MC9S12C128 and MC9S12GC128 FCS Full Chip Simulation Implementation Note Covers MC9S12C and MC9S12GC Family

1 Introduction

1.1 Overview of MC9S12C/ MC9S12GC Family

The MC9S12C-Family / MC9S12GC-Family are 48/52/80 pin Flash-based MCU families, which deliver the power and flexibility of the 16-bit core to a whole new range of cost and space sensitive, general purpose industrial and automotive network applications. All MC9S12C-Family / MC9S12GC-Family members feature standard on-chip peripherals including a 16-bit central processing unit (CPU12), up to 128K bytes of Flash EEPROM, up to 4K bytes of RAM, an asynchronous serial communications interface (SCI), a serial peripheral interface (SPI), an 8-channel 16-bit timer module (TIM), a 6-channel 8-bit pulse width modulator (PWM), an 8-channel, 10-bit analog-to-digital converter (ADC). The devices are available in 48-, 52-, and 80- pin QFP packages, with the 80-pin version pin compatible to the HCS12 A, B, and D Family derivatives.

1.2 MC9S12C/ MC9S12GC 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, TIM ...). In the section Simulation Summary the supported I/O devices are listed for each supported derivative of MC9S12C/GC -Family.

HC(S) 12 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 MC9S12C/ MC9S12GC -Family. It details the features that are simulated, not simulated, or simulated with exceptions.

1.4 References

Item	Description
MC9S12C128V1.pdf	MC9S12C and MC9S12GC family Data Sheet
Debugger_HC12.pdf	HC(S) 12 Debugger Manual.

2 Simulation Summary

The following table gives a summary of peripheral options of MC9S12C /GC Family members. Each derivative for C 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.

Derivative →		**MC9S12C128**	**MC9S12C96**	**MC9S12C64**	**MC9S12C32**	**MC9S12GC128**	**MC9S12GC64**	**MC9S12GC32**	**MC9S12GC16**
Peripheral Module	**Module Version**								
PIM (port integration module)	PIM9C32	1							
MMC (memory mapping control)	MMCV4 [1) 2)]	1							
MEBI (external bus interface)	MEBIV3 [6)]	1							
DBG (debug module)	DBGV1 [2)]	1							
CRG (clock and reset generator)	CRGV4	1							
TIM (timer)	TIM16B8CV1	1							
ATD (analog-to-digital converter)	ATD10B8CV2	1							
SCI (serial communications interface)	S12SCIV2	1							
SPI (serial peripheral interface)	SPIV3 [3)]	1							
INT (interrupt module)	INTV1 [4)]	1							
CAN (controller area network)	S12MSCANV2 [3)]	1				-	-	-	-
PWM (pulse width modulator)	PWM8B6CV1	1							
FTS (Flash)	S12FTS128K1V1	1	-	-	-	1	-	-	-
FTS (Flash)	S12FTS96KV1	-	1	-	-	-	-	-	-
FTS (Flash)	S12FTS64KV4	-	-	1	-	-	1	-	-
FTS (Flash)	S12FTS32KV1	-	-	-	1	-	-	1	-
FTS (Flash)	S12FTS16KV1	-	-	-	-	-	-	-	1
BDM (background debug module)	S12XBDMV4 [2) 5)]	1							
OSC (pierce oscillator)	OSCV2 [2) 5)]	1							

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) MEBI available only in 80 pin package.

3 Peripheral Modules – Simulation Details

3.1.1 Implementation of PIM module (PIM9C32)

Major functionalities of GPIO are implemented. GPIO functionalities like reduced output drive, wired-or functions are not implemented. Static routing is implemented only for peripherals PWM and TIM. 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 and TIM are implemented.

The ports that are available are PortA, PortB, PortE, PortT, PortS, PortM, PortP, and PortJ.

Each port can be accessed by using the template mentioned below:

To Access all the bits of the port use PIM.PORT<x>Val.

To Access a particular bit of the port use PIM.PORT<x>Pin<y>.

Where x is the port name and y represents the bit position (Least significant bit is 0).

Consider the example of PORT B:

To Access all 8 bits of port B use the virtual register PIM.PORTBVal.

To Access individual bits of port B use the virtual registers below:

PIM.PORTBPin0 - Least significant bit.

PIM.PORTBPin1

PIM.PORTBPin2

PIM.PORTBPin3

PIM.PORTBPin4

PIM.PORTBPin5

PIM.PORTBPin6

PIM.PORTBPin7 - Most significant bit.

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, RDRJ and RDRAD) functionalities are not implemented but read/write to same has been provided.

All Wired-Or Mode Registers (WOMS and WOMM) functionalities 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.

The Slew rate Registers (SRCR) is not implemented but read/write to same has been provided.

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	PTA7	PTA6	PTA5	PTA4	PTA3	PTA2	PTA1	PTA0
	W								
0x0001 PORTB	R	PTB7	PTB6	PTB5	PTB4	PTB3	PTB2	PTB1	PTB0
	W								
0x0002 DDRA	R	DDRA7	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0
	W								
0x0003 DDRB	R	DDRB7	DDRB6	DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0
	W								
0x0004-07 RESERVED	R	0	0	0	0	0	0	0	0
	W								
0x0008 PORTE	R	PTE7	PTE6	PTE5	PTE4	PTE3	PTE2	PTE1	PTE0
	W								
0x0009 DDRE	R	DDRE7	DDRE6	DDRE5	DDRE4	DDRE3	DDRE2	DDRE1	DDRE0
	W								
0x000A PEAR	R	NOACCE	0	PIPOE	NECLK	LSTRE	RDWE	0	0
	W								
0x000B MODE	R	MODC	MODB	MODA	0	IVIS	0	EMK	EME
	W								
0x000C DDRE	R	PUPKE	0	0	PUPEE	0	0	PUPBE	PUPAE
	W								
0x000D RDRIV	R	RDPK	0	0	RDPE	0	0	RDPB	RDPA
	W								

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x000E	R	0	0	0	0	0	0	0	ESTR
EBICTL	W								
0x000F	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	SRRK	0	0	SRRE	SRRD	SRRC	SRRB	SRRA
SRCR	W								
0x0032	R	PTK7	PTK6	PTK5	PTK4	PTK3	PTK2	PTK1	PTK0
PORTK	W								
0x0033	R	DDRK7	DDRK6	DDRK5	DDRK4	DDRK3	DDRK2	DDRK1	DDRK0
DDRK	W								
0x0200	R	PTT7	PTT6	PTT5	PTT4	PTT3	PTT2	PTT1	PTT0
PTT	W								
0x0201	R	PTIT7	PTIT6	PTIT5	PTIT4	PTIT3	PTIT2	PTIT1	PTIT0
PTIT	W								
0x0202	R	DDRT7	DDRT6	DDRT5	DDRT4	DDRT3	DDRT2	DDRT1	DDRT0
DDRT	W								
0x0203	R	RDRT7	RDRT6	RDRT5	RDRT4	RDRT3	RDRT2	RDRT1	RDRT0
RDRT	W								
0x0204	R	PERT7	PERT6	PERT5	PERT4	PERT3	PERT2	PERT1	PERT0
PERT	W								
0x0205	R	PPST7	PPST6	PPST5	PPST4	PPST3	PPST2	PPST1	PPST0
PPST	W								
0x0206	R	0	0	0	0	0	0	0	0
RESERVED	W								
0x0207	R	0	0	0	MODRR4	MODRR3	MODRR2	MODRR1	MODRR0
MODRR	W								
0x0208	R	0	0	0	0	PTS3	PTS2	PTS1	PTS0
PTS	W								

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0209	R	0	0	0	0	PTIS3	PTIS2	PTIS1	PTIS0
PTIS	W								
0x020A	R	0	0	0	0	DDRS3	DDRS2	DDRS1	DDRS0
DDRS	W								
0x020B	R	0	0	0	0	RDRS3	RDRS2	RDRS1	RDRS0
RDRS	W								
0x020C	R	0	0	0	0	PERS3	PERS2	PERS1	PERS0
PERS	W								
0x020D	R	0	0	0	0	PPSS3	PPSS2	PPSS1	PPSS0
PPSS	W								
0x020E	R	0	0	0	0	WOMS3	WOMS2	WOMS1	WOMS0
WOMS	W								
0x020F	R	0	0	0	0	0	0	0	0
RESERVED	W								
0x0210	R	0	0	PTM5	PTM4	PTM3	PTM2	PTM1	PTM0
PTM	W								
0x0211	R	0	0	PTIM5	PTIM4	PTIM3	PTIM2	PTIM1	PTIM0
PTIM	W								
0x0212	R	0	0	DDRM5	DDRM4	DDRM3	DDRM2	DDRM1	DDRM0
DDRM	W								
0x0213	R	0	0	RDRM5	RDRM4	RDRM3	RDRM2	RDRM1	RDRM0
RDRM	W								
0x0214	R	0	0	PERM5	PERM4	PERM3	PERM2	PERM1	PERM0
PERM	W								
0x0215	R	0	0	PPSM5	PPSM4	PPSM3	PPSM2	PPSM1	PPSM0
PPSM	W								
0x0216	R	0	0	WOMM5	WOMM4	WOMM3	WOMM2	WOMM1	WOMM0
WOMM	W								
0x0217	R	0	0	0	0	0	0	0	0
RESERVED	W								

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0218 PTP	R	PTP7	PTP6	PTP5	PTP4	PTP3	PTP2	PTP1	PTP0
	W								
0x0219 PTIP	R	PTIP7	PTIP6	PTIP5	PTIP4	PTIP3	PTIP2	PTIP1	PTIP0
	W								
0x021A DDRP	R	DDRP7	DDRP6	DDRP5	DDRP4	DDRP3	DDRP2	DDRP1	DDRP0
	W								
0x021B RDRP	R	RDRP7	RDRP6	RDRP5	RDRP4	RDRP3	RDRP2	RDRP1	RDRP0
	W								
0x021C PERP	R	PERP7	PERP6	PERP5	PERP4	PERP3	PERP2	PERP1	PERP0
	W								
0x021D PPSP	R	PPSP7	PPSP6	PPSP5	PPSP4	PPSP3	PPSP2	PPSP1	PPSP0
	W								
0x021E PIEP	R	PIEP7	PIEP6	PIEP5	PIEP4	PIEP3	PIEP2	PIEP1	PIEP0
	W								
0x021F PIFP	R	PIFP7	PIFP6	PIFP5	PIFP4	PIFP3	PIFP2	PIFP1	PIFP0
	W								
0x0220-027 RESERVED	R	0	0	0	0	0	0	0	0
	W								
0x0228 PTJ	R	PTJ7	PTJ6	0	0	0	0	0	0
	W								
0x0229 PTIJ	R	PTIJ7	PTIJ6	0	0	0	0	0	0
	W								
0x022A DDRJ	R	DDRJ7	DDRJ6	0	0	0	0	0	0
	W								
0x022B RDRJ	R	RDRJ7	RDRJ6	0	0	0	0	0	0
	W								
0x022C PERJ	R	PERJ7	PERJ6	0	0	0	0	0	0
	W								
0x022D PPSJ	R	PPSJ7	PPSJ6	0	0	0	0	0	0
	W								

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x022E PIEJ	R	PIEJ7	PIEJ6	0	0	0	0	0	0
	W								
0x022F PIFJ	R	PIFJ7	PIFJ6	0	0	0	0	0	0
	W								
0x0230 PTAD	R	PTAD7	PTAD6	PTAD5	PTAD4	PTAD3	PTAD2	PTAD1	PTAD0
	W								
0x0231 PTIAD	R	PTIAD7	PTIAD6	PTIAD5	PTIAD4	PTIAD3	PTIAD2	PTIAD1	PTIAD0
	W								
0x0232 DDRAD	R	DDRAD7	DDRAD6	DDRAD5	DDRAD4	DDRAD3	DDRAD2	DDRAD1	DDRAD0
	W								
0x0233 RDRAD	R	RDRAD7	RDRAD6	RDRAD5	RDRAD4	RDRAD3	RDRAD2	RDRAD1	RDRAD0
	W								
0x0234 PERAD	R	PERAD7	PERAD6	PERAD5	PERAD4	PERAD3	PERAD2	PERAD1	PERAD0
	W								
0x0235 PPSAD	R	PPSAD7	PPSAD6	PPSAD5	PPSAD4	PPSAD3	PPSAD2	PPSAD1	PPSAD0
	W								
0x0236-3F RESERVED	R	0	0	0	0	0	0	0	0
	W								

= Not simulated

= Only Read / Write simulated, but no functionality

= Unimplemented or reserved (on Hardware and Simulation)

3.2 Memory Mapping Control (MMCV4)

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.

Program Page Index Register (PPAGE)

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) and 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 /		Bit 7	6	5	4	3	2	1	Bit 0
0x0010	R	RAM15	RAM14	RAM13	RAM12	RAM12	0	0	RAMHAL
INITRM	W								
0x0011	R	0	REG14	REG13	REG12	REG11	0	0	0
INITRG	W								
0x0012	R	0	EE14	EE13	EE12	EE11	0	0	EEON
INITEE	W								
0x0013	R	0	0	0	0	EXSTR1	EXSTR0	ROMHM	ROMON
MISC	W								

Register Offset /		Bit 7	6	5	4	3	2	1	Bit 0
0x0014	R	0	0	0	0	0	0	0	0
MTSTO	W								
0x0017	R	0	0	0	0	0	0	0	0
MTST1	W								
0x001C	R	REG_SW0	EEP_SW1	0	EEP_SW0	0	RAM_SW2	RAM_SW1	RAM_SW0
MEMSIZ0	W								
0x001D	R	ROM_SW1	ROM_SW0	0	0	0	0	PAG_SW1	PAG_SW0
MEMSIZ1	W								
0x0030	R	0	0	PIX5	PIX4	PIX3	PIX2	PIX1	PIX0
PPAGE	W								
0x0031	R	0	0	0	0	0	0	0	0
Reserved	W								

= Not simulated

= Only Read / Write simulated, but no functionality

= Unimplemented or reserved (on Hardware and Simulation)

3.3 Clocks and Reset Generator (S12CRGV4)

3.3.1 Implementation

The primitive version of CRG which has a Synthesizer Divider, and a Reference Divider to generate various PLL output frequencies and that makes the “Clock Factor” to be a function of Synthesizer and Reference. The S12XCRGV6 version is the enhanced version of its predecessor. They have an improved reference register [5 bits] for generating the PLL output frequencies. Also, prescalar values selected for RTI timeout can be binary or decimal.

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

3.3.2 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.

3.3.3 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

3.3.4 Register Details (CRGV4)

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 SYNR	R	0	0	SYNDIV[5:0]					
	W								
0x0001 REFDV	R	0	0	0	0	REFDIV[3:0]			
	W								
0x0002 CTFLG	R	0	0	0	0	0	0	0	0
	W								
0x0003 CRGFLG	R	RTIF	PORF	LVRF[A)]	LOCKIF	LOCK	TRACK	SCMIF	SCM
	W								
0x0004 CRGINT	R	RTIE	0	0	LOCKIE	0	0	SCMIE	0
	W								

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

= Not simulated

= Only Read / Write simulated, but no functionality

= Unimplemented or reserved (on Hardware and Simulation)

3.4 Analog-to-Digital Converter (ATD10B8CV2)

3.4.1 Implementation

ATD_10B8C is a primitive analog to digital converter with 8/10 bit resolution with 16 channels 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.

3.4.2 Extensions

PADx

The analog inputs channels are reachable separately through the object pool. They are implemented as Non Memory Mapped ATD virtual registers called PAD0 to PAD15. For the ATD 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>

3.4.3 Restrictions

The unimplemented hardware features of ATD10B8CV2 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.

3.4.4 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	0	0	0	0	0
[ATDCTL0]	W								
0x0001	R	0	0	0	0	0	0	0	0
[ATDCTL1]	W								
0x0002	R	ADPU	AFFC	AWAI	ETRIGLE	ETRIGP	ETRIGE	ASCIE	ASCIF
[ATDCTL2]	W								

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0003 [ATDCTL3]	R	0	S8C	S4C	S2C	S1C	FIFO	FRZ1	FRZ0
	W								
0x0004 [ATDCTL4]	R	SRES8	SMP1	SMP0	PRS[4:0]				
	W								
0x0005 [ATDCTL5]	R	DJM	DSGN	SCAN	MULT	0	CC	CB	CA
	W								
0x0006 [ATDSTAT0]	R	SCF	0	ETORF	FIFOR	0	CC2	CC1	CC0
	W								
0x0007 Unimplemented	R	0	0	0	0	0	0	0	0
	W								
0x0008 ATDTEST0	R	U	U	U	U	U	U	U	U
	W								
0x0009 ATDTEST1	R	U	U	0	0	0	0	0	SC
	W								
0x000A Unimplemented	R	0	0	0	0	0	0	0	0
	W								
0x000B [ATDSTAT1]	R	CCF7	CCF6	CCF5	CCF4	CCF3	CCF2	CCF1	CCF0
	W								
0x000C Unimplemented	R	0	0	0	0	0	0	0	0
	W								
0x000D [ATDDIEN]	R	IEN7	IEN6	IEN5	IEN4	IEN3	IEN2	IEN1	IEN0
	W								
0x000E Unimplemntd	R	0	0	0	0	0	0	0	0
	W								
0x000F [PORTAD]	R	PTAD7	PTAD6	PTAD5	PTAD4	PTAD3	PTAD2	PTAD1	PTAD0
	W								

Left Justified, ATD Conversion Result Register

0x0010,12, 14,16,18, 1A,1C,1E ATDDR0H -7H	R	Bit 9 MSB	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2
	W	Bit 7 MSB	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

Register Offset / Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0011,13, 15,17,19, 1B,1D,1F	R	Bit 1	Bit 0	0	0	0	0	0	0
ATDDR0L -7L	W	U	U	0	0	0	0	0	0

Right justified, ATD Conversion Result Register

0x0010,12, 14,16,18, 1A,1C,1E	R	0	0	0	0	0	Bit 9 MSB	Bit 8	0
ATDDR0H -7H	W	0	0	0	0	0	0	0	0

0x0011,13, 15,17,19, 1B,1D,1F	R	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ATDDR0L -7L	W	Bit 7 MSB	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

= Not simulated

= Only Read / Write simulated, but no functionality

= Unimplemented or reserved (on Hardware and Simulation)

3.5 Timer (TIM16B8CV1)

3.5.1 Implementation

The basic timer consists of a 16-bit, software-programmable counter driven by a seven-stage programmable prescaler. This 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 timer contains 8 complete input capture/output compare 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.

3.5.2 Extensions

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

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

3.5.3 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 pre-scalar clock is always used as timer counter clock as the simulator handles only pre-scalar 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.

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

0x000F TFLG2	R	TOF	0	0	0	0	0	0	0
	W								
0x0010-0x001F	R	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
	W								
TCxH -TCxL	R	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
	W								
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 PACNTH	R	PACNT15	PACNT14	PACNT13	PACNT12	PACNT11	PACNT10	PACNT9	PACNT8
	W								
0x0023 PACNTL	R	PACNT7	PACNT6	PACNT5	PACNT4	PACNT3	PACNT2	PACNT1	PACNT0
	W								
0x0024-2F RESERVED	R								
	W								

= Not simulated

= Only Read / Write simulated, but no functionality

= Unimplemented or reserved (on Hardware and Simulation)

3.6 Serial Communication Interface (S12SCIV2)

3.6.1 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 following functionalities have been implemented in all the above mentioned versions of SCI.

3.6.2 Extension

The SCI module also relies on additional simulated functionality such as virtual registers for input and output to simulate the external pins TXD and RXD.

3.6.3 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

3.6.4 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 SCIBDH	R	IREN	TNP1	TNP0	SBR12	SBR11	SBR10	SBR9	SBR8
	W								
0x0001 SCIBDL	R	SBR7	SBR6	SBR5	SBR4	SBR3	SBR2	SBR1	SBR0
	W								
0x0002 SCICR1	R	LOOPS	SCISWAI	RSRC	M	WAKE	ILT	PE	PT
	W								
0x0003 SCICR2	R	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
	W								
0x0004 SCISR1	R	TDRE	TC	RDRF	IDLE	OR	NF	FE	PF
	W								
0x0005 SCISR2	R	0	0	0	TXPOL	RXPOL	BRK13	TXDIR	RAF
	W								
0x0006 SCIDRH	R	R8	T8	0	0	0	0	0	0
	W								
0x0007 SCIDRL	R	R7	R6	R5	R4	R3	R2	R1	R0
	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)

3.7 Pulse Width Modulator (S12PWM8B6CV1)

3.7.1 Implementation

The pulse width modulation (PWM) definition is based on the HC12PWMdefinitions. The PWM8B6CV1 module contains the basic features from the HC11 with some of the enhancements incorporated on the HC12, that is center aligned output mode and four available clock sources. The PWM8B6CV1 module has six channels with independent control of left and center aligned outputs on each channel. Each of the six PWM 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 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.

3.7.2 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.

3.7.3 Restrictions

The following is the list of constraints:

The PFRZ bit in register PWMCTL is not implemented . Refer to the table below.

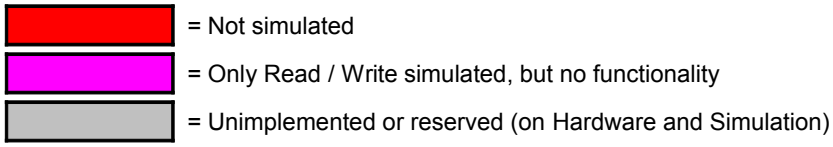
Note: Writing the PMSDN register bit manipulating instructions MUST be avoided because it may give incorrect results. Always write whole byte in this register.

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.

3.7.4 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	0	0	PWME5	PWME4	PWME3	PWME2	PWME1	PWME0
	W								
0x0001 [PWMPOL]	R	0	0	PPOL5	PPOL4	PPOL3	PPOL2	PPOL1	PPOL0
	W								
0x0002 [PWMCLK]	R	0	0	PCLK5	PCLK4	PCLK3	PCLK2	PCLK1	PCLK0
	W								
0x0003 [PWMPRCLK]	R	0	PCKB2	PCKB1	PCKB0	0	PCKA2	PCKA1	PCKA0
	W								
0x0004 [PWMCAE]	R	0	0	CAE5	CAE4	CAE3	CAE2	CAE1	CAE0
	W								
0x0005 [PWMCTL]	R	0	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	Bit7	6	5	4	3	2	1	Bit0
	W								
0x000A-0B PWMSCNTA-B	R	0	0	0	0	0	0	0	0
	W								
0x000C-11 [PWMCNT0-5]	R	0	0	0	0	0	0	0	0
	W								
0x0012-17 PWMPER0-5	R	Bit7	6	5	4	3	2	1	Bit0
	W								
0x0018-1D PWMDTY0-7	R	Bit7	6	5	4	3	2	1	Bit0
	W								
0x001E [PWMSDN]	R	PWMIF	PWMIE	0	PWMLVL	0	PWM5IN	PWM5INL	PWM5ENA
	W			PWMRSTRT					


= Not simulated
= Only Read / Write simulated, but no functionality
= Unimplemented or reserved (on Hardware and Simulation)