Freescale Semiconductor
Application Note

Document Number:AN4526
Rev. 1.0, March 2013

Kinetis 100 MHz Rev 1.x to 120 MHz Migration Guide

1 Purpose and overview

This document describes the details of migrating from Kinetis 100 MHz Revision 1.x microcontrollers to Kinetis 120/150 MHz microcontrollers. Migrating between the two devices within the same family could require hardware and/or software changes in some cases. Changes that might be required are described in this document.

Contents

1.1 Part numbering and mask set information

The table below lists all of the Kinetis 100 MHz rev 1.x mask sets that have been produced as of the writing of this document.

Table 1. Kinetis 100 MHz Rev 1.x mask sets

Revision	Mask Set	Part Number Example
1.0	0M33Z	PK10N512VMD100
1.1	0N30D	N/A
1.2	1N30D / 2N30D (functionally identical)	PK10N512VMD100
1.4	4N30D	MK10DN512ZVMD10 ('Z' character: INITIAL Production mask set)

The table below lists the Kinetis 120 and 150 MHz masks sets that have been produced as of the writing of this document.

Table 2. Kinetis 120 and 150 MHz mask sets

Revision	Mask Set	Part Number Example
1.0	0N96B	PK60FN1M0VMJ12
1.1	1N96B	PK60FN1M0VMJ12
1.3	3N96B	MK60FN1M0VMJ12

This document is primarily focused on migration between 100 MHz and 120/150MHz devices within the same Kinetis family. For example, this document will help if you are migrating from an MK60DN512VMD10 to a PK60FN1M0VMD12, but if you are migrating from a K10 to a K60 device, this document might still have some helpful information but it is not focused on addressing the changes in functionality between the Kinetis families.

1.2 About this document

This document describes migration between Kinetis 100 MHz and Kinetis 120 and 150 MHz devices. The 120 and 150 MHz devices are speed grade options for the same devices. There are no functional or feature differences between the speed grades with the exception of the core frequency and resulting performance increase. For simplicity, the document will refer to these devices as Kinetis 120 MHz devices. The information applies to the 150 MHz devices as well, but they will not be specifically referenced.

This document is divided up into five major sections – new modules, updated modules, enhanced modules, modules with additional instantiations, and unchanged modules.

The new modules section provides a quick overview of modules that are completely new for the 120 MHz devices. These are modules that don't have any functional equivalent available in the 100 MHz microcontrollers. If any of these new modules will be used in your application, then you'll require software changes to use the new module. Most of the modules will also require hardware changes to use them with the exception of the LMEM and FPU.

The updated modules section outlines the modules that have been updated to use newer versions. The overall functionality provided will be similar, however, changes will be required in software. Hardware changes may be required in order to utilize new features.

The enhanced modules section outlines the existing modules which have undergone minor changes. Software and hardware changes may be required in order to utilize new features.

The modules with additional instantiations sections describes modules where the modules themselves don't have changes, but there are more instantiations of the module included in the microcontroller.

The unchanged modules section lists modules that are the same between the 100 MHz and 120 MHz devices.

Additionally, a color coding scheme has been used throughout this document where:

- GREEN: Designates new additions,
- YELLOW: Designates changes, and
- RED: Designates removals

2 New Modules

The Kinetis 120 MHz platforms adds several modules that are not available on the 100 MHz devices. The following sections give an overview of the features of these modules. Applications that want to take advantage of these new modules will require software changes and in some cases hardware changes to take advantage of the new functionality.

Availability of these modules can depend on the specific Kinetis device that you are using. Please refer to the reference manual for your Kinetis device for information on which features are available.

2.1 NAND Flash Controller (NFC)

The NAND flash controller (NFC) interfaces to standard NAND flash memory devices. It is composed of various control logic units and a 9 KB SRAM buffer. The NFC provides a glueless interface to 8- and 16-bit NAND flash devices with page sizes of 512 bytes, 2 KB, 4 KB, and 8 KB.

2.2 DDR Memory Controller (DDRMC)

Interface to store and retrieve data from an external SDRAM. Supports glue-less interface to 16-bit or 8-bit DDR1, DDR2, or LPDDR SDRAM memories.

2.3 USB High Speed OTG Controller (USBHS)

The USB high speed OTG controller (USBHS) is a USB 2.0-compliant serial interface engine for implementing a USB interface. The registers and data structures are based on the Enhanced Host Controller Interface Specification for Universal Serial Bus (EHCI) from Intel Corporation. The USBHS module can act as a host, a device, or an On-The-Go negotiable host/device on the USB bus. The USBHS controller interfaces to the processor's core. The controller is programmable to support host or device operations under firmware control. To operate the USB HS OTG controller, connect the USB HS controller to an external ULPI PHY/transceiver via the ULPI interface.

2.4 Graphical LCD Controller (LCDC)

Provides display data for external gray-scale or color LCD panels. It supports black-and-white, gray-scale, passive-matrix color (CSTN), and active-matrix color (TFT) LCD panels. The maximum panel resolution supported is 800x600 with up to 24 bpp color.

2.5 MCU DryIce

The MCU DryIce module includes a 32-byte secure memory that is asynchronously erased on any tamper detect. In addition it can optionally force a system reset and/or invalidate the Real Time Clock.

2.6 Local Memory Controller (LMEM)

The local memory controller is a combination of the system RAM controller that is used on all Kinetis devices and a system cache controller that is new to the 120 MHz Kinetis devices. The LMEM manages simultaneous accesses to system RAM by multiple master peripherals and core and also controls cache which improves system performance by providing single-cycle access to the instruction and data pipelines.

2.7 Floating Point Unit (FPU)

The Kinetis 120 MHz devices add in the Cortex M4 core's optional floating point unit. The single precision FPU is compliant to the *IEEE Standard for Floating-Point Arithmetic* (IEEE 754).

3 Updated Modules

3.1 Flash Memory Module (FTFL to FTFE)

In order to support the higher density of flash available on the 120 MHz Kinetis devices, the flash block has been changed. The 100 MHz devices use the FTFL module and support up to 512 KB of flash, while the 120 MHz devices use the FTFE module and support up to 1 MB of flash.

In addition to supporting larger flash arrays, the 120 MHz devices also support a larger FlexRAM. 100 MHz devices have up to 4 KB of FlexRAM, while the 120 MHz devices have up to 16 KB of FlexRAM.

3.1.1 Memory map comparison

The FTFE and FTFL use different module acronyms, but other than that the memory map remains the same. For example, on a 100 MHz device there is an FTFL_FSTAT register. The FTFL_FSTAT register becomes the FTFE_FSTAT register on a 120 MHz device. All of the register addresses and fields are identical only the preceding acronym changes.

3.1.2 Software impact

The biggest change when moving from the FTFL to the FTFE module is that the programming size for the flash changes. The FTFL module uses a program longword command (FCMD = 0x6) to program 32 bits at a time, while the FTFE module uses a program phrase command (FCMD = 0x7) to program 64 bits at a time. Any application that modifies the flash contents will need to be changed to use the new program size.

Another change is that the default size of the flash protection regions increases. For example, there are 32 P-flash protection regions on both the FTFE and FTFL where the total P-flash size is even distributed across the 32 regions. For an FTFE device with twice as much P-flash, the size of the P-flash protection regions will double. The D-flash and EEPROM protection region sizes also increase accordingly.

3.1.3 Hardware impact

No hardware impact.

3.2 Power Management Controller (PMC)

The primary duties of the Power Management Controller (PMC) are: control the regulator, control the POR and LVD circuits, and provide voltage and current sources for the MCU. For Kinetis 120 MHz devices the I/O retention control is now in the PMC. The functionality of the PMC is linked closely with the Low Leakage Wakeup Unit (LLWU), the System Mode Controller (SMC), and the new Reset Control Module (RCM).

3.2.1 Memory map comparison

The register map and names have remained the same. The figure below shows the only bit level changes to the memory map.

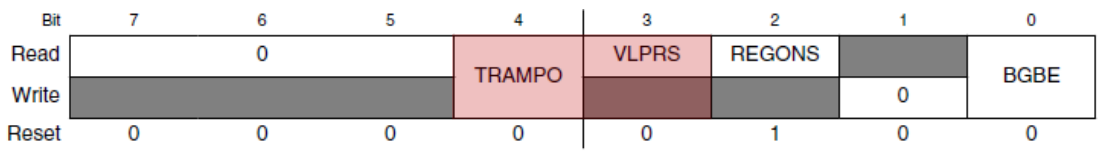


Figure 1. PMC_REGSC—Kinetis 100 MHz Rev. 1.x

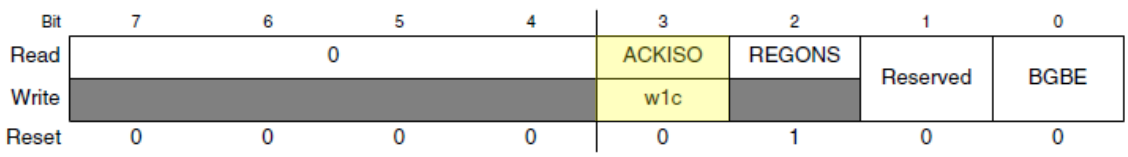


Figure 2. PMC_REGSC—Kinetis 120 MHz

Removed bit/field names:

- TRAMPO
- VLPRS

Changed bit/field names:

- ACKISO - Moved from LLWU to PMC

3.2.2 Software impact

The PMC register names are the same; however, some important bit changes have been made.

TRAMPO: The TRAMPO bit has been removed. There is a new bit in the SMC—RAM2PO—that controls the RAM power in VLLS2. The new RAM2PO bit has different functionality, however.

VLPRS: The VLPRS bit has been removed. Software only needs to check the Run Regulator Status and the SMC_PMSTAT register when entering VLPR mode. The REGONS bit will clear when the regulator is in stop regulation or transition to/from it and the SMC_PMSTAT will read 04 when the MCU power mode is in VLPR.

ACKISO: The PMC has the acknowledge to clear the hold on the state of the I/O pins and oscillator module. This bit used to reside in the LLWU. This bit must be cleared upon recovery from VLLSx modes to release the hold on the I/O and oscillator modules.

3.2.3 Hardware impact

Bandgap Enable (BGEN): The new bit has hardware interaction impact. If BGEN is set, then the bandgap is turned on. The operation of this bit may be needed by other circuitry like the VREF module or ADC.

ACKISO: The ACKISO bit is an important bit to consider when working with the low power modes of operation. The MCU recovers from VLLSx through the reset flow. Typically the port I/O, module initialization of timers, communications modules, and the oscillator should be initialized prior to acknowledging the release of the I/O.

3.3 Mode Controller Version 1 to Version 2

The primary duty of the Mode Controller (MC) and System Mode Controller (SMC) has been to control the entry into and exit from each power mode. The module has a new name in newer Kinetis devices. The MC is now the SMC. The functionality of the SMC has always been linked closely with the Power Management Controller (PMC), the Low Leakage Wakeup Unit (LLWU), and the new Reset Control Module (RCM).

3.3.1 Memory Map Comparison

There are two new registers in the SMC module. They are SMC_VLLSCTRL and SMC_PMSTAT. The two SRS registers that are in the MC have been moved to the RCM. The changes to this register will be discussed in the RCM section.

Address: MC_PMPROT is 4007_E000h base + 2h offset = 4007_E002h

Bit	7	6	5	4	3	2	1	0
Read	0		AVLP	ALLS	0	AVLLS3	AVLLS2	AVLLS1
Write								
Reset	0	0	0	0	0	0	0	0

Figure 3. MC_PMPROT

Address: SMC_PMPROT is 4007_E000h base + 0h offset = 4007_E000h

Bit	7	6	5	4	3	2	1	0
Read	0		AVLP	0	ALLS	0	AVLLS	0
Write								
Reset	0	0	0	0	0	0	0	0

Figure 4. SMC_PMPROT

Changed bit/field names:

- ALLS—Moved to bit 3
- AVLLS3—AVLLS
- AVLLS2—AVLLS
- AVLLS1—AVLLS

Address: MC_PMCTRL is 4007_E000h base + 3h offset = 4007_E003h

Bit	7	6	5	4	3	2	1	0
Read	LPWUI	RUNM		0		LPLLSM		
Write								
Reset	0	0	0	0	0	0	0	0

Figure 5. MC_PMCTRL

Address: SMC_PMCTRL is 4007_E000h base + 1h offset = 4007_E001h

Bit	7	6	5	4	3	2	1	0
Read	LPWUI	RUNM		0	STOPA	STOPM		
Write								
Reset	0	0	0	0	0	0	0	0

Figure 6. SMC_PMCTRL

Changed bit/field names:

- STOPA—New bit in the SMC
- LPLLSM—VLLSM, moves to new register SMC_VLLSSCTRL
- STOPM—New bit field in the SMC

Address: SMC_VLLSCTRL is 4007_E000h base + 2h offset = 4007_E002h

Bit	7	6	5	4	3	2	1	0
Read	0		0	RAM2PO	0	VLLSM		
Write								
Reset	0	0	0	0	0	0	1	1

Figure 7. SMC_VLLCTRL

Changed bit/field names:

- RAM2PO—New bit in the SMC
- VLLSM—New bit field in the SMC

Address: SMC_PMSTAT is 4007_E000h base + 3h offset = 4007_E003h

Bit	7	6	5	4	3	2	1	0
Read	0	PMSTAT						
Write								
Reset	0	0	0	0	0	0	0	1

Figure 8. SCM_PMSTAT

Changed bit/field names:

PMSTAT—New bit field in the SMC

3.3.2 Software impact

General impact: Any register reference to the MC register will need to change the RCM and SMC register names. The SRS registers are in the RCM. The prefix for the mode control functions changes from MC to SMC.

SMC_PMPROT: The SMC_PMPROT register is still a one-time-write after reset type of register. The protection of entry into the VLLSx low-power modes is controlled by one bit in the SMC. Software will need to write to the PMPROT in the initialization code to allow only the desired low-power modes. The address changes of the registers and the location change of the ALLS bit is handled by the header file definitions for the SMC.

SMC_PMCTRL and SMC_VLLSCTRL:

The SMC_PMCTRL and SMC_VLLSCTRL register control mode entry and exit. Software will need to change the values written to these registers to accomplish the functions done in the MC_PMCTRL register.

A read of the new bit STOPA in the SMC_PMCTRL register will indicate that an interrupt or reset occurred during a stop entry sequence and the SMC can abort the transition early and return to Run mode without completely entering the stop mode. See the reference manual SMC functional description for more information.

The new STOPM bit field has the following meaning:

2–0 STOPM	Stop Mode Control When written, this field controls entry into the selected stop mode when sleep-now or sleep-on-exit mode is entered with SLEEPDEEP=1 . Writes to this field are blocked if the protection level has not been enabled using the PMPROT register. After any system reset, this field is cleared by hardware on any successful write to the PMPROT register. **NOTE:** When set to VLLSx, the VLLSM bits in the VLLSCTRL register is used to further select the particular VLLS sub-mode which will be entered. **NOTE:** 000 Normal stop (STOP) 001 Reserved 010 Very low power stop (VLPS) 011 Low leakage stop (LLS) 100 Very low leakage stop (VLLSx) 101 Reserved 110 Reseved 111 Reserved

Low-Power Mode Entry Change: If a low-power mode of Stop, VLPS, or LLS is desired, the PMPROT bit must first be set and the a write to STOPM is needed. If one of the VLLSx modes is desired, one additional write to the VLLSM bits in the SMC_VLLSCTRL is needed.

2–0 VLLSM	VLLS Mode Control This field controls which VLLS sub-mode to enter if STOPM=VLLS. 000 Reserved 001 VLLS1 010 VLLS2 011 VLLS3 100 Reserved 101 Reserved 110 Reserved 111 Reserved

SMC_PMSTAT

A read of this new register in the SMC will give you the current low-power mode.

3.3.3 Hardware impact

RAM power in VLLS2 mode

A new bit in the SCM_VLLSCTRL register—RAM2PO—controls whether the RAM partition 2 is powered or not while in VLLS2 low-power mode.

VLPR and VLPW now supported

With the SMC the mode VLPR and VLPW are now supported. The way you enter the modes is the same between revisions.

Debug in low-power modes

Please note that the SMC allows debug operation in Run, Wait, VLPR, or VLPW in the same way as in the MC. The debugger handles attempts to enter Stop and VLPS by entering an emulated stop state. See the reference manual SMC functional description for more information.

3.4 Reset Controller Module (RCM)

The RCM has been updated; however, not all of the functions that are in this module are new. The SRS registers that were in the MC previously are now in the RCM. The SRS register has been revised.

3.4.1 Memory map comparison

Bit	7	6	5	4	3	2	1	0
Read	POR	PIN	COP	0		LOC	LVD	WAKEUP
Write								
Reset	1	0	0	0	0	0	1	0

Bit	7	6	5	4	3	2	1	0
Read	0					SW	LOCKUP	JTAG
Write								
Reset	0	0	0	0	0	0	0	0

Figure 9. SRS Register High and Low (MC_SRSx)

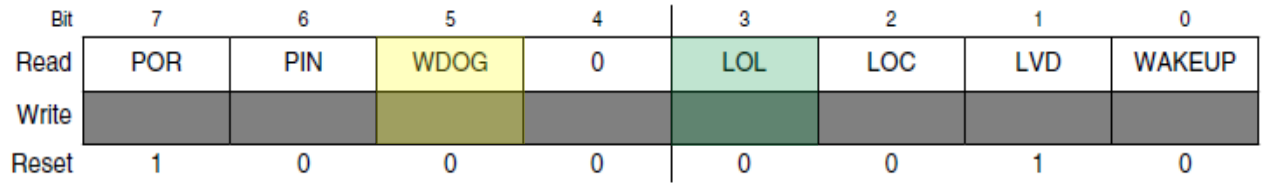


Figure 10. SRS Register 0 (RCM_SRS0)

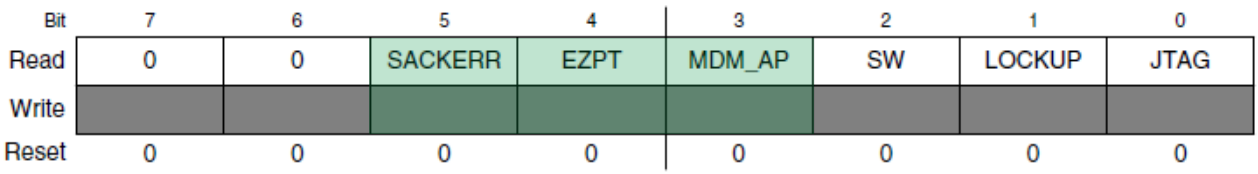


Figure 11. SRS Register 1 (RCM_SRS1)

Changed bit/field names:

- COP → WDOG - field name change

New bit/field names:

- LOL
- SACKERR
- EZPT
- MDM_AP

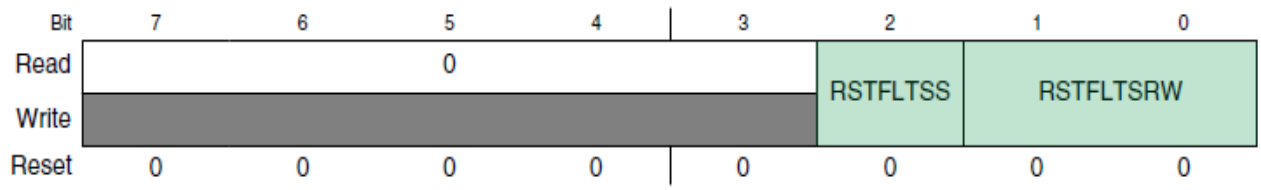


Figure 12. Reset Pin Filter Control Register (RCM_RPFC)—New register

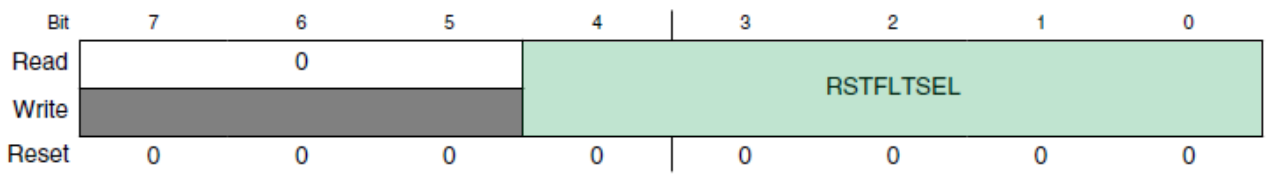


Figure 13. Reset Pin Filter Width Register (RCM_RFPW)—New register

3.4.2 Software impact

Any references to MC_SRSH and MC_SRSL will need to be changed to the new register names. Software that decoded the reset information in these two registers will need to account for the new bits added in each of the two registers.

If you want to use the new digital filter function of the RCM, please refer to the reference manual sections Reset and Boot and Reset Control Module.

3.4.3 Hardware impact

There are four additional sources of reset that have been added to the RCM.

The Reset pin input can be digitally filtered with either the LPO clock or the bus clock. The bus clock filter value is selectable in the RCM_RFPW register.

3.5 LLWU

The Low Leakage Wakeup Unit (LLWU) controls the exit from LLS and VLLSx power modes. The functionality of the LLWU is linked closely with the Power Management Controller (PMC), the System Mode Controller (SMC), and the new Reset Control Module (RCM).

3.5.1 Memory map comparison

There are three new registers in the LLWU module. They are the LLWU_FILT1, LLWU_FILT2, and LLWU_RST registers. One register has been removed from Rev. 2 of the LLWU, the LLWU_CS.

Address: LLWU_ME is 4007_C000h base + 4h offset = 4007_C004h

Bit	7	6	5	4	3	2	1	0
Read Write	WUME7	WUME6	WUME5	WUME4	WUME3	WUME2	WUME1	WUME0
Reset	0	0	0	0	0	0	0	0

Figure 14. LLWU_ME Register

Changed bit/field names:

WUME7—Connected to RTC Seconds enable

Address: LLWU_F3 is 4007_C000h base + 7h offset = 4007_C007h

Bit	7	6	5	4	3	2	1	0
Read	MWUF7	MWUF6	MWUF5	MWUF4	MWUF3	MWUF2	MWUF1	MWUF0
Write								
Reset	0	0	0	0	0	0	0	0

Figure 15. LLWU_F3 Register

Changed bit/field names:

WUMF7—Connected to RTC Seconds interrupt

The RTC seconds flag is connected to bit 7 of the corresponding LLWU flag register.

Address: LLWU_CS is 4007_C000h base + 8h offset = 4007_C008h

Bit	7	6	5	4	3	2	1	0
Read	ACKISO	0					FLTEP	FLTR
Write	w1c					1		
Reset	0	0	0	0	0	1	0	0

Figure 16. LLWU_CS—Rev. 1 removed

Changed bit/field names:

- ACKISO—Moved to bit 3 of the PMC_REGSC register
- FLTEP—Filter functionality changed

Address: LLWU_FILT1 is 4007_C000h base + 8h offset = 4007_C008h

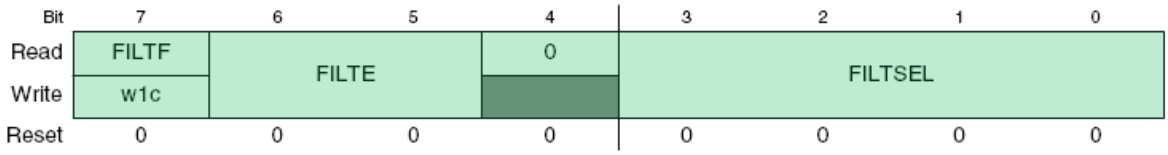


Figure 17. LLWU_FILT1—New register Rev. 2.x

Address: LLWU_FILT2 is 4007_C000h base + 9h offset = 4007_C009h

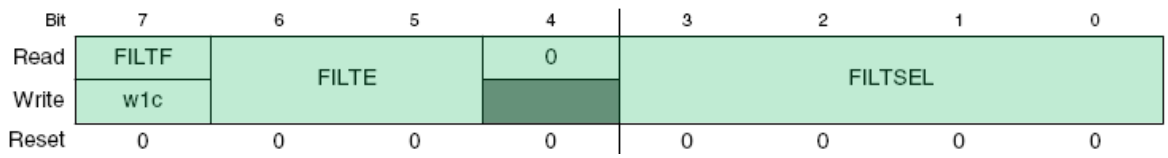


Figure 18. LLWU_FILT2—New register Rev. 2.x

Address: LLWU_RST is 4007_C000h base + Ah offset = 4007_C00Ah

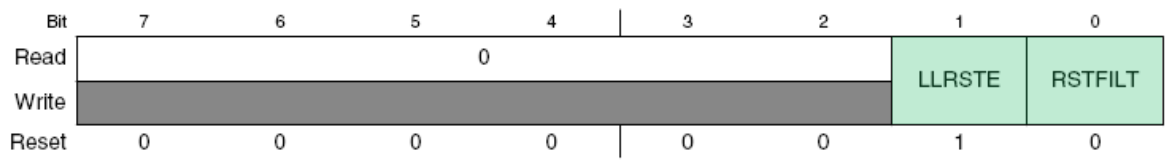


Figure 19. LLWU_RST—New register Rev. 2.x

The functionality of the LLRSTE and RSTFILT bits is enabled when there is no dedicated Reset pin. Kinetis parts have a dedicated Reset pin; therefore, these bits do not change the operation of the Reset pin.

3.5.2 Software impact

LLWU Module Enable and Flag

Not obvious in the register description for Rev. 2 of the LLWU is the fact that the RTC Seconds module interrupt is connected to bit 7 of the LLWU_ME register [WUME7]. The RTC seconds flag is connected to bit 7 [MWUF7] of the corresponding LLWU flag register LLWU_F3.

LLWU pin and reset filters

Register LLWU_FILT1, LLWU_FILT2, and LLWU_RST control the filter functionality of the LLWU. See the reference manual LLWU register definition and functional description for more information.

ACKISO move

The ACKISO bit is now in the PMC module registers. Software will need to re-target the write-1-to-clear operation to the new register. This is an important bit for users who are waking up from VLLSx modes through the reset flow. A read of the ACKISO bit indicates whether the I/O pads and the system oscillator(s) are in a latched state. Care should be used in not clearing this bit too early. You should re-initialize the I/O and the oscillator before writing the ACKISO or there might be a glitch on the pins or in the oscillator startup.

3.5.3 Hardware impact

LLWU_M7IF → RTC seconds: In Rev. 2 of the LLWU, Module Wakeup bit 7 is connected to the output of the RTC seconds interrupt. This allows the RTC seconds output to wake up the MCU from LLS and VLLSx low-power modes.

LLWU pin filter function: The functionality of the pin filter in the LLWU module has changed. In the original LLWU, all of the LLWU inputs are fed into one pin filter circuitry resulting in a single wakeup flag. In the new revision, only two preselected LLWU inputs are fed into two pin filter circuits resulting in two possible filtered pin wakeup flags.

The newer version of the LLWU has filter and pin enable for the reset pin to wake up the MCU from LLS and VLLSx low power modes. For devices like Kinetis that have a dedicated Reset pin, the wakeup from LLS and VLLSx modes is always enabled and these bits do not enable or disable the Reset pin as a wakeup source in low-power modes.

3.6 RNG-B to RNG-A

Kinetis 100 MHz 1.x versions use random number generator version B (RNG-B) while the latest versions of the Kinetis family use random number generator version A (RNG-A). While these two versions are radically different, the migration is surprisingly simple. This section details the differences between the two versions and describes the necessary changes to your Kinetis setup to ensure a smooth transition from RNG-B to RNG-A.

3.6.1 RNG-A vs. RNG-B

The RNG-B variant is a cryptographically strong random number generator with three distinctive features:

- National Institute of Standards and Technology (NIST)-approved pseudo-random number generator (http://csrc.nist.gov)
- Inclusion of the key generation algorithm defined in the Digital Signature Standard (http://www.itl.nist.gov/fipspubs/fip186.htm)
- Integrated entropy sources capable of providing the PRNG with entropy for its seed

RNG-B uses a true random number generator module (TRNG) to add entropy to a register from which the pseudo-random number generator is seeded. See below for a detailed block diagram of the RNG-B module.

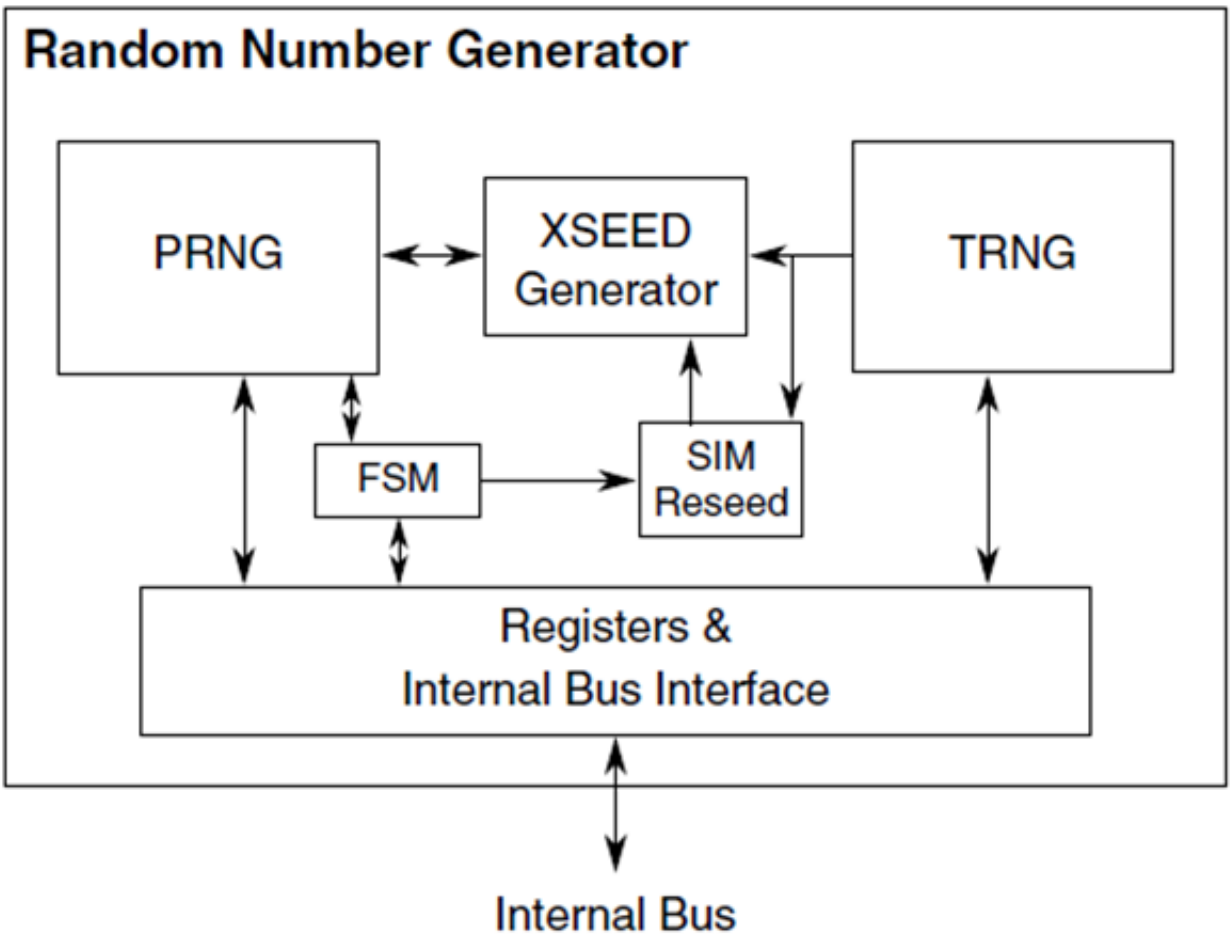


Figure 20. Random number generator

The RNG-A variant is simply a pseudo-random number generator. This module is less bulky and less complex than the RNG-B variant. The RNG-A block diagram is shown in the block diagram below.

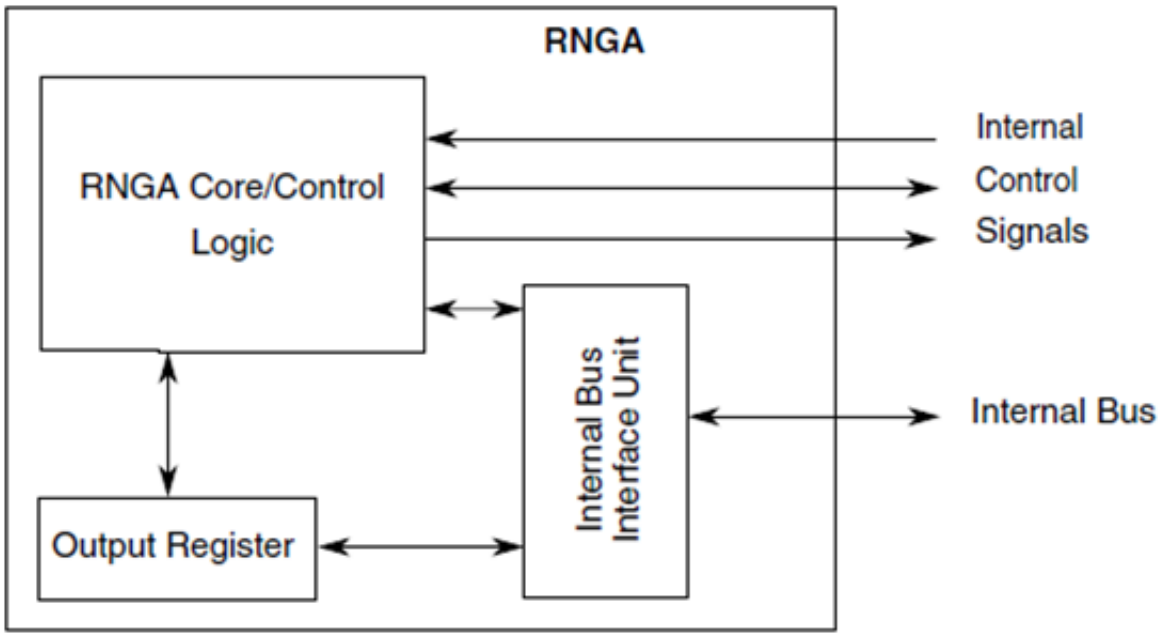


Figure 21. RNG-A block diagram

There is no known cryptographic proof showing that this is a secure method of generating random data. Therefore, it is highly recommended that the random data produced by this module be used as an input seed to a NIST-approved (based on DES or SHA-1) pseudo-random number generator as defined in NIST Fips Pub 186-2 Appendix 3 and NIST Fips Pub SP 800-90.

Though these modules are very different internally, operationally they are very similar. The RNG-A module does not allow for manual seeding. Thus, only the automatic operation algorithms will be compared here. The two different initialization and operation modes are displayed below.

Table 3. Initialization and operation algorithms comparison

RNG-A	RNG-B (Automatic)	RNG-B (Manual)
1. Reset/initialize 2. Write to the RNGA Control Register and set the Interrupt Mask (INTM), High Assurance (HA), and GO bits. 3. Poll the RNGA Status Register for RNGA Output Register level. 4. Read available random data from the RNGA Output Register. 5. Repeat steps 3 and 4 as needed.	1. Reset/initialize. 2. Write to the RNG_CR to set up the RNGB for automatic seeding and the desired functionality. 3. Wait for interrupt to indicate completion of first seed. 4. Poll RNG_SR for FIFO level. 5. Read available random data from output FIFO. 6. Repeat steps 4 and 5 as needed. Automatic seeding occurs when necessary and is transparent to operation.	1. Reset/initialize. 2. Write to the RNG_CR to set up the desired functionality. 3. Write to RNG_CMD register to run self-test or seed generation. 4. Wait for interrupt to indicate completion of the requested operation(s). 5. Repeat steps 3 and 4 if seed generation is not complete. 6. Poll RNG_SR for FIFO level. 7. Read available random data from output FIFO. 8. Repeat steps 6 and 7 as needed, until 2^{20} words have been generated. 9. Write to RNG_CMD to run seed mode. 10. Repeat steps 4 through 9.

3.6.2 Memory map comparison

The RNG-A and RNG-B modules share all of the registers that are present in the RNG-A module with the exception of the entropy register. Therefore, migration to the RNG-A module requires that you remove some code and change the values written to the registers that are shared.

Though they may share registers, the registers are not located in the same memory locations. Therefore, it is important to update your system with the latest Freescale-provided header files for your system to work correctly.

The changes in the memory map locations are displayed in the figure below. Changes in the reset values for these registers are discussed in the specific register comparisons.

Memory map comparison				
	RNG-B		RNG-A	
	Location	**Name**	**Location**	**Name**
Control Register	400A_0008	RNG_CR	400A_0000	RNG_CR
Status Register	400A_000C	RNG_SR	400A_0004	RNG_SR
Output Register	400A_0014	RNG_OUT	400A_000C	RNG_OR
Entropy Register	N/A	N/A	400A_0008	RNG_ER
Command Register	400A_0004	RNG_CMD	N/A	N/A
Error Status Register	400A_0010	RNG_ESR	N/A	N/A
Version Register	400A_0000	RNG_VER	N/A	N/A

In addition to the changes in location, there have also been changes to the register structures and some reset values. The control register structure of RNG-A and RNG-B is shown below. Note that the register structures are completely different.

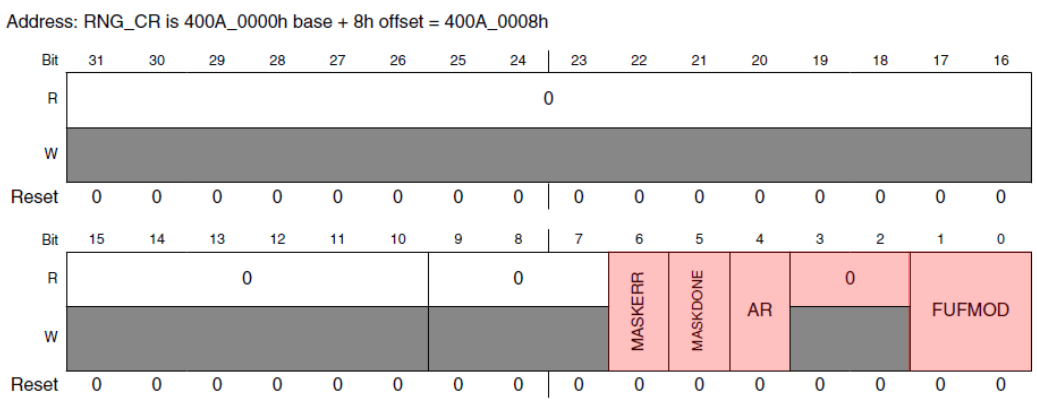


Figure 22. RNG-B module RNG_CR

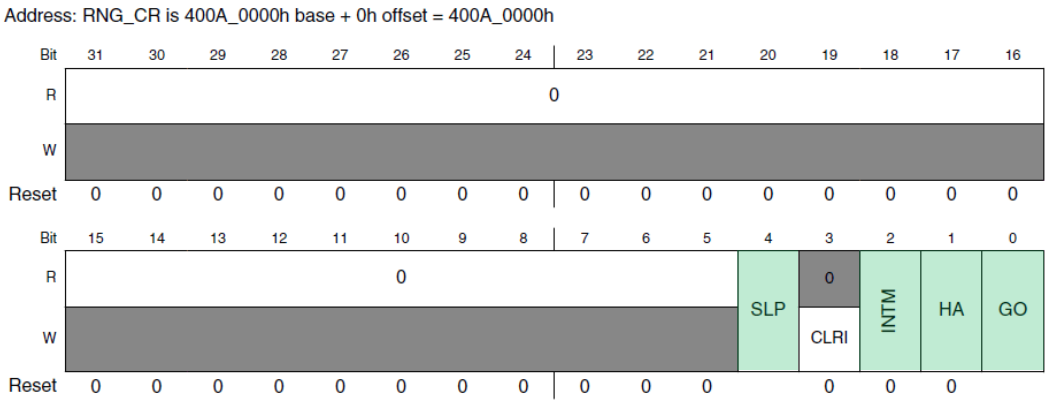


Figure 23. RNG-A module RNG_CR

Deleted bit fields:

- MASKERR
- MASKDONE
- AR
- FUFMOD

Added bit fields:

- SLP: Setting this bit puts the RNG module in Sleep mode; clearing this bit awakens it.
- CLRI: Setting this bit clears the error interrupt flag.
- INTM: Setting this bit enables RNG interrupts; clearing it disables them.
- HA: Enables the security violation bit in the RNG register. Reads of the RNG register while this bit is set are not permitted.
- GO: RNG register is loaded with random data.

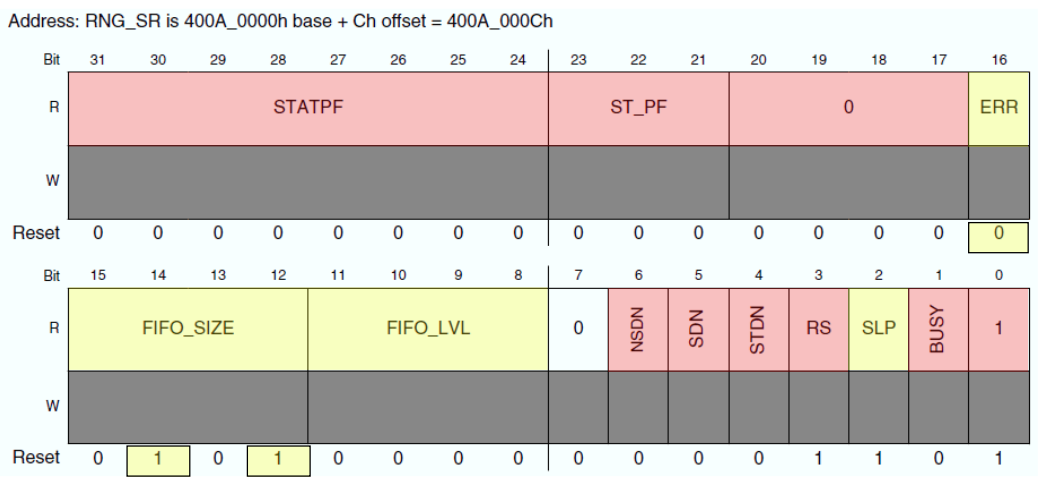


Figure 24. RNG-B module RNG_SR

Address: RNG_ESR is 400A_0000h base + 10h offset = 400A_0010h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0															
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0											FUFE	SATE	STE	OSCE	LFE
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 25. RNG-B module RNG_ESR

Address: RNG_SR is 400A_0000h base + 4h offset = 400A_0004h

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0								OREG_SIZE							
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	OREG_LVL								0			SLP	ERRI	ORU	LRS	SECV
W																
Reset	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0

Figure 26. RNG-A module RNG_SR

Added bit fields:

- SECV: Signals that a security violation has occurred if set. No security violation has occurred if cleared.
- LRS: When set, the last read status bit indicates that the last read was performed while the output register was empty (underflow condition).

Changed bit fields:

- OREG_SIZE: This is an 8-bit integer field that indicates the size of the output register and has been renamed from FIFO_SIZE in the RNG-B variant.
- OREG_LVL: This bit indicates that a random word is available in the output register. Only two values are possible (0b00000001 or 0b00000000). This bit was renamed from FIFO_LVL in the RNG-B variant.
- SLP: Indicates that the RNG module is in Sleep mode when set. This bit was moved from bit 2 (in the RNG-B variant) to bit 4.
- ERRI: When set, this bit indicates that the output register was read while empty. This bit was moved from bit 16 (in the RNG-B variant) to bit 3.
- ORU: When set, this bit indicates that the output register was read while empty since the last read of the status register. This bit was moved from bit 4 of the ESR register (in the RNG-B variant) to bit 2.

Address: RNG_OUT is 400A_0000h base + 14h offset = 400A_0014h

Bit	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
R	RANDOUT																															
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 27. RNG-B module RNG_OUT

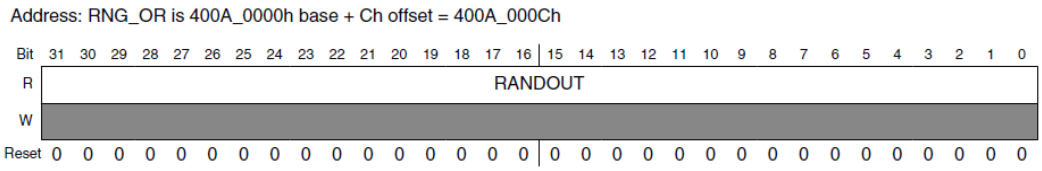


Figure 28. RNG-A module RNG_OR

The output register is simply a 32-bit register that holds the resultant random number once a random number has been generated and, thus, the structure has not been changed. The register has been renamed as shown below.

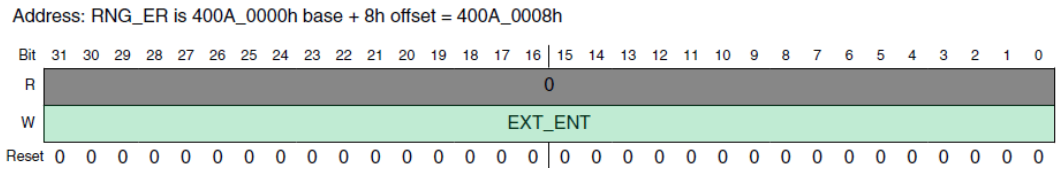


Figure 29. RNG-A module RNG_ER

The RNG-A module allows you to add entropy to the random number generation process. This is done by writing a value to the write-only entropy register. Sources of entropy that may feed the entropy registers are:

- Current time (highest precision possible)
- Mouse and/or keyboard motions
- Other random number generators

3.6.3 Software impact

Freescale Semiconductor will provide customers affected by this migration with new header and linker files, making the migration to the new RNG-A module as seamless as possible. It is important to remember (as previously mentioned in this document) that the RNG-A algorithm does not produce a cryptographically strong random number. If your current software setup relies on a cryptographically strong random number generator for your cryptographic algorithm, you will need NIST-approved support software to generate a cryptographically strong random number to feed into your current software solution. In addition, it is recommended that you use the entropy register of the RNG-A module to add extra entropy to your random number generation process.

There are two general cases for RNG-B to RNG-A migration: either a manually seeded RNG-B algorithm is being implemented or an automatically seeded RNG-B algorithm is being implemented. The manually seeded RNG-B implementation is examined first below.

3.6.4 Impact when using manual seeding algorithm

If you are using manual seeding in your code, you will need to rewrite your code based on the automatic seeding algorithm of RNG-A, as RNG-A provides no manual seeding option. The two algorithms are presented below in a side-by-side comparison highlighting the similarities in the algorithms

RNG-B Manual Algorithm	RNG-A Algorithm
Reset/Initialize	Reset/Initialize
Write to the RNG_CR to setup the desired functionality	Write to the RNGA_CR and set the Interrupt Mask (INTM), High Assurance (HA), and GO bits

Table continues on the next page...

RNG-B Manual Algorithm	RNG-A Algorithm
Write to RNG_CMD register to run self-test or seed generation.	
Wait for interrupt to indicate completion of the requested operation(s).	
Repeat steps 3-4 if seed generation is not complete.	
Poll RNG_SR for FIFO level.	Poll RNG_SR for Output register level.
Read available random data from output FIFO.	Read the available data from the Output Register.
Repeat steps 6 and 7 as needed, until 2^{20} words have been generated.	
Write to RNG_CMD to run seed mode	
Repeat steps 4 through 9.	Repeat steps 3 and 4.

As can be seen from the comparison above, when switching to the RNG-A algorithm, steps 3, 4, 5, and 9 may be deleted from your code project. If you will not be using the Freescale-provided header and linker files, remember that, when writing to the RNGA_CR register, you will need to update the value that is written to the CR register. Also, when polling the SR register, you will to poll bits 8–15 (instead of just 8–11).

3.6.5 Impact when using automatic seeding algorithm

The algorithmic difference between RNG-A and RNG-B are minimal when implementing the automatic seeding functionality. Observe the chart comparison below.

RNG-B Manual Algorithm	RNG-A Manual Algorithm
Reset/initialize	Reset/initialize
Write to the RNG_CR to set up the desired functionality	Write to the RNGA_CR and set the Interrupt Mask (INTM), High Assurance (HA), and GO bits
Wait for interrupt to indicate completion of first seed.	
Poll RNG_SR for FIFO level.	Poll RNG_SR for output register level.
Read available random data from output FIFO.	Read the available data from the output register.
Repeat steps 4 through 9.	Repeat steps 3 and 4.

If you are using the automatic seeding algorithm, you will simply need to remove your code related to the RNG-B seeding interrupt, adjusting the value written to the control register to be the proper value for RNG-A, and adjusting your code to read from the correct output level register. To simplify the output register reads, it is recommended that you insert a define statement converting the output register name of RNG-B to the output register name of the RNG-A output register.

```
(#define RNG_OUT    RNG_OR).
```

3.6.6 Hardware impact

No hardware impact.

3.7 SSI to SAI

The Kinetis 100 MHz Rev. 1.x devices used the SSI module to provide I2S capability, but newer Kinetis devices use the SAI module. While the modules have similar functionality, there are many differences.

In addition to changing from the SSI to the SAI, the 120MHz devices have up to two instantiations of the SAI while the 100MHz devices only had one.

3.7.1 Features

Even though the register map and bit definitions of the SAI module are quite different from the SSI module, the main features are still the same. SAI supports full duplex synchronous serial interfaces with frame sync such as I2S, TDM, AC97, codec, and DSP interfaces. However, in the movement from the SSI to the SAI module, some features have been added and some have been removed:

Following is a list of what has been removed on the SAI module:

- No option to disable transmit and receive frame sync separately. Once TE or RE enabled, frame sync generates, it can only be disabled by clearing TE or RE.
- No option to output oversampling clock on RX bit clock.
- No separate bits to control whether SAI operates under I2S mode or network mode; it all depends on how you configure the frame sync size.
- No longer supports gated clock mode.
- No hardware support for AC97, though AC97 can still be supported with software if configured for 13 or more words per frame.
- No support for separate start-of-frame flag and start-of-last-unmasked-word-in-frame-flag. These have been replaced with single flag to represent the start of a word in a frame.
- No option to disable TX or RX FIFOs. They are always enabled.
- No support for separate left/right FIFOs. Instead there are two separate pins for TX and RX, each associated with its own FIFO.
- No option to select MSB aligned or LSB aligned
- No bits to represent FIFO entry counts. Instead, a write and read FIFO pointer is used to represent current FIFO entry and can be used to tell whether the FIFO is full or empty.

Below is a list of what is new on the SAI module

- Support for Stop mode operation
- Support for Debug mode operation
- Support for 32-bit transfers
- Added option to enable and disable bit clock separately
- Added frame sync error flag
- Added TX FIFO empty or RX FIFO full flag to allow trigger interrupt and DMA, apart from the TX and RX FIFO watermark trigger
- One additional TX and RX data channel
- SAI master clock select and divide registers relocated to SAI's own memory space. (For Kinetis Rev. 1.x, these were located in the SIM module.)

3.7.2 Memory map comparison

Table 7 provides a memory map comparison of the SSI and SAI modules. Since the memory map and the registers' intended functions have changed significantly in some cases, the table uses the following conventions:

1. Registers that implement similar functions are listed in the same row; for example, I2S0_TX0 corresponds to I2S0_TDR0.
2. If there is an SSI module register whose function needs to be implemented with more than one register, that one register map is merged so that it corresponds to all those related registers in the SAI module; for example, I2S0_TCR's function needs to be implemented with I2S0_TCR2, I2S0_TCR3, and I2S0_TCR4.
3. Some of the registers in either the SSI or SAI module do not have a related register in the other module; for example, there is no hardware acceleration support for AC97 on SAI and no related AC97 registers on SAI. In these cases, "N/A" in the associated row indicates that the register is not available.
4. For the SSI module, master clock generation needs to configure two registers in the SIM module, while SAI has pulled related registers into its own memory space.

Table 7. Memory map comparison

	SSI			SAI	
I2S transmit data register 0	4002_F000	I2S0_TX0	SAI transmit data register 0	4002_F020	I2S0_TDR0
I2S transmit data register 1	4002_F004	I2S0_TX1	SAI transmit data register 1	4002_F024	I2S0_TDR1
I2S receive data register 0	4002_F008	I2S0_RX0	SAI receive data register 0	4002_F0A0	I2S0_RDR0
I2S receive data register 1	4002_F00C	I2S0_RX1	SAI receive data register 1	4002_F0A4	I2S0_RDR1
I2S control register	4002_F010	I2S0_CR	SAI transmit control register	4002_F000	I2S0_TCSR
			SAI receive control register	4002_F080	I2S0_RCSR
			SAI transmit configuration 2 register	4002_F008	I2S0_TCR2
			SAI receive configuration 2 register	4002_F088	I2S0_RCR2
			SAI transmit configuration 3 register	4002_F00C	I2S0_TCR3
			SAI receive configuration 3 register	4002_F08C	I2S0_RCR3
I2S interrupt status register	4002_F014	I2S0_ISR	SAI transmit control register	4002_F000	I2S0_TCSR
			SAI receive control register	4002_F080	I2S0_RCSR
I2S interrupt enable register	4002_F018	I2S0_IER	SAI transmit control register	4002_F000	I2S0_TCSR
			SAI receive control register	4002_F080	I2S0_RCSR

Table continues on the next page...

Table 7. Memory map comparison (continued)

	SSI			SAI	
I2S transmit configuration register	4002_F01C	I2S0_TCR	SAI transmit configuration 2 register	4002_F008	I2S0_TCR2
			SAI transmit configuration 3 register	4002_F00C	I2S0_TCR3
			SAI transmit configuration 4 register	4002_F010	I2S0_TCR4
I2S receive configuration register	4002_F020	I2S0_RCR	SAI receive configuration 2 register	4002_F088	I2S0_RCR2
			SAI receive configuration 3 register	4002_F08C	I2S0_RCR3
			SAI receive configuration 4 register	4002_F090	I2S0_RCR4
I2S transmit clock control register	4002_F024	I2S0_TCCR	SAI transmit configuration 2 register	4002_F008	I2S0_TCR2
			SAI transmit configuration 4 register	4002_F010	I2S0_TCR4
			SAI transmit configuration 5 register	4002_F014	I2S0_TCR5
I2S receive clock control register	4002_F028	I2S0_RCCR	SAI receive configuration 2 register	4002_F088	I2S0_RCR2
			SAI receive configuration 4 register	4002_F090	I2S0_RCR4
			SAI receive configuration 5 register	4002_F094	I2S0_RCR5

Table continues on the next page...

Table 7. Memory map comparison (continued)

	SSI			SAI	
I2S FIFO control/ status register	4002_F02C	I2S0_FCSR	SAI transmit configuration 1 register	4002_F004	I2S0_TCR1
			SAI receive configuration 1 register	4002_F084	I2S0_RCR1
			SAI transmit FIFO 0 register	4002_F040	I2S0_TFR0
			SAI transmit FIFO 1 register	4002_F044	I2S0_TFR1
			SAI receive FIFO 0 register	4002_F0C0	I2S0_RFR0
			SAI receive FIFO 1 register	4002_F0C4	I2S0_RFR1
I2S0 transmit time slot mask register	4002_F048	I2S0_TMSK	SAI transmit mask register	4002_F060	I2S0_TMR
I2S0 receive time slot mask register	4002_F04C	I2S0_RMSK	SAI receive mask register	4002_F0E0	I2S0_RMR
I2S AC97 command data register	4002_F040	I2S0_ACDAT	N/A	N/A	N/A
I2S AC97 tag register	4002_F044	I2S0_ATAG	N/A	N/A	N/A
I2S AC97 Channel status register	4002_F050	I2S0_ACCST	N/A	N/A	N/A
I2S AC97 channel enable register	4002_F054	I2S0_ACCEN	N/A	N/A	N/A
I2S AC97 channel disable register	4002_F058	I2S0_ACCDIS	N/A	N/A	N/A
System option register 2	4004_8004	SIM_SOPT2	SAI MCLK control register	4002_F100	I2S0_MCR
System clock divider register 2	4004_8048	SIM_CLKDIV2	SAI MCLK divider register	4002_F014	I2S0_MDR

3.7.2.1 Control register

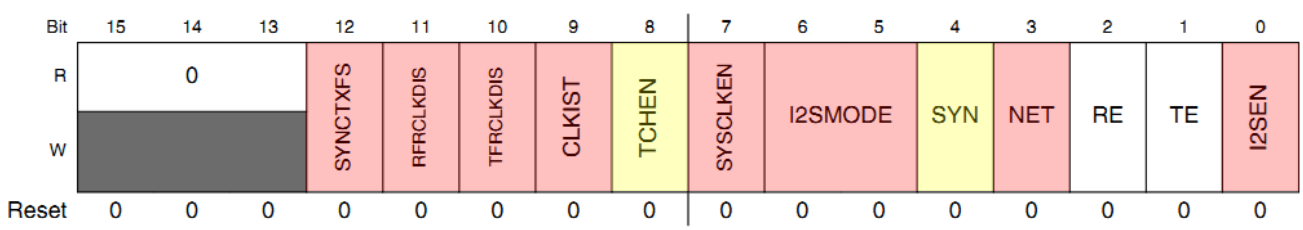


Figure 30. I2S0_CR–SSI

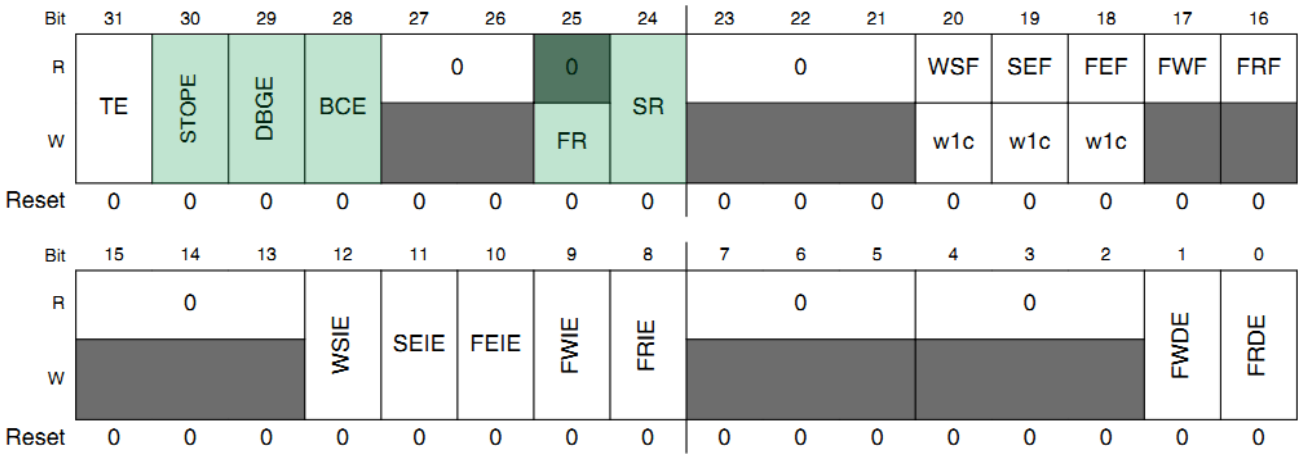


Figure 31. I2S0_TCSR–SAI

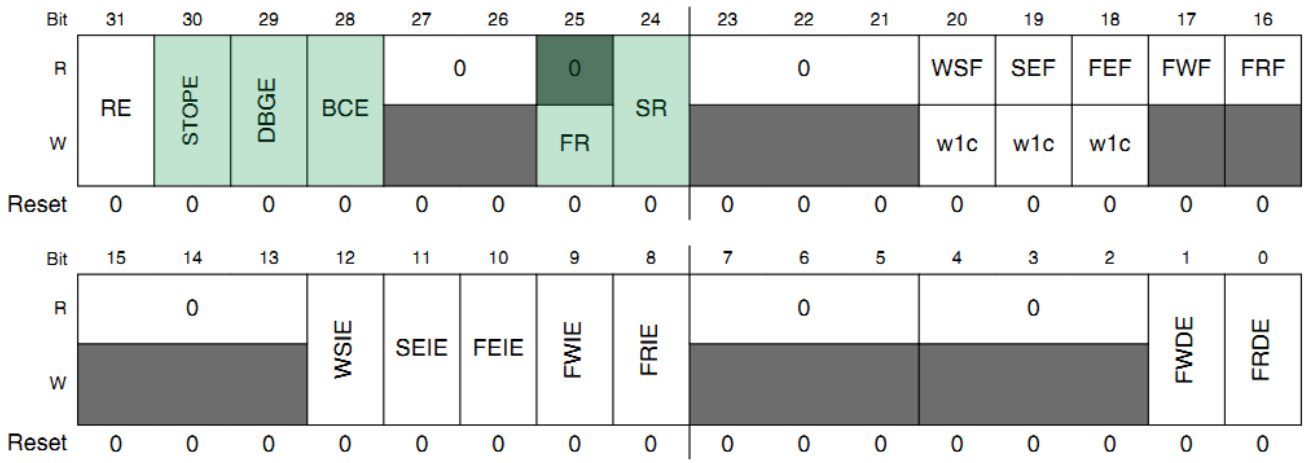


Figure 32. I2S0_RCSR–SAI

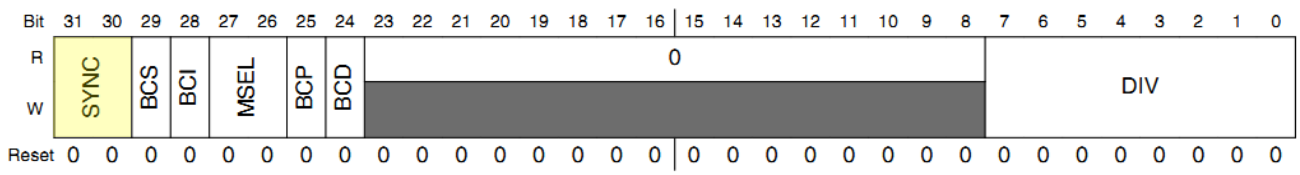


Figure 33. I2S0_TCR2 and I2S0_RCR2–SAI

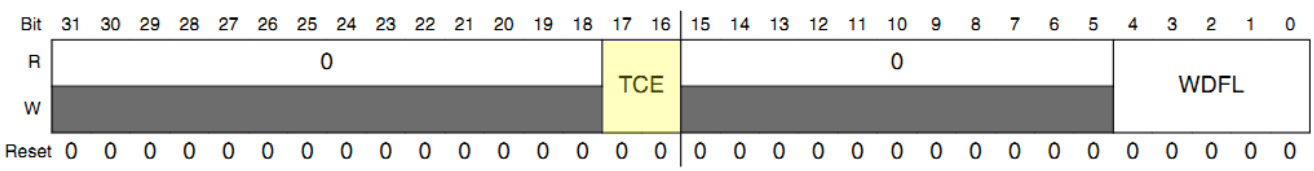


Figure 34. I2S0_TCR3–SAI

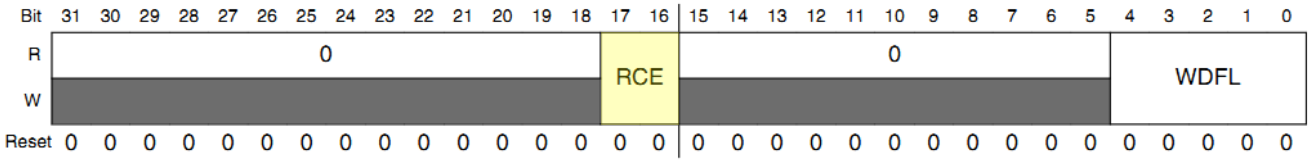


Figure 35. I2S0_RCR3–SAI

New bits/fields added:

- **STOPE:** Stop Enable
- **DBGE:** Debug Enable
- **FR:** FIFO Reset
- **SR:**Software Reset

Changed bits/fields name:

- TCHEN ⇢ TCE and RCE
- SYN ⇢ SYNC

Removed bits/fields:

- **SYNCTXFS:** CR[TE] latch with FS occurrence
- **RFRCLKDIS:** Receive Frame Sync Disable
- **TFRCLKDIS:** Transmit Frame Sync Disable
- **CLKIST:** Clock Idle State during I2S Gated Clock Mode
- **SYSCLKEN:** Oversampling Clock Enable
- **I2SMODE:** I2S Mode Select
- **NET:** Network Mode I2SEN: I2S enable

3.7.2.2 Interrupt and status enable register

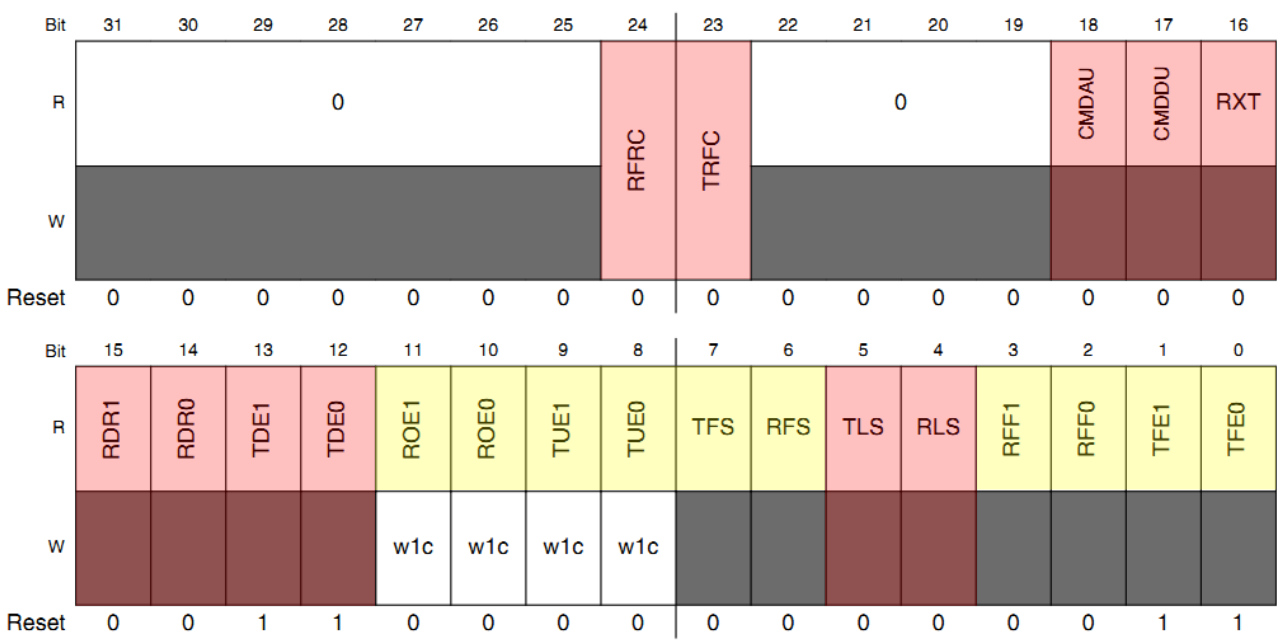


Figure 36. I2S0_ISR—SSI

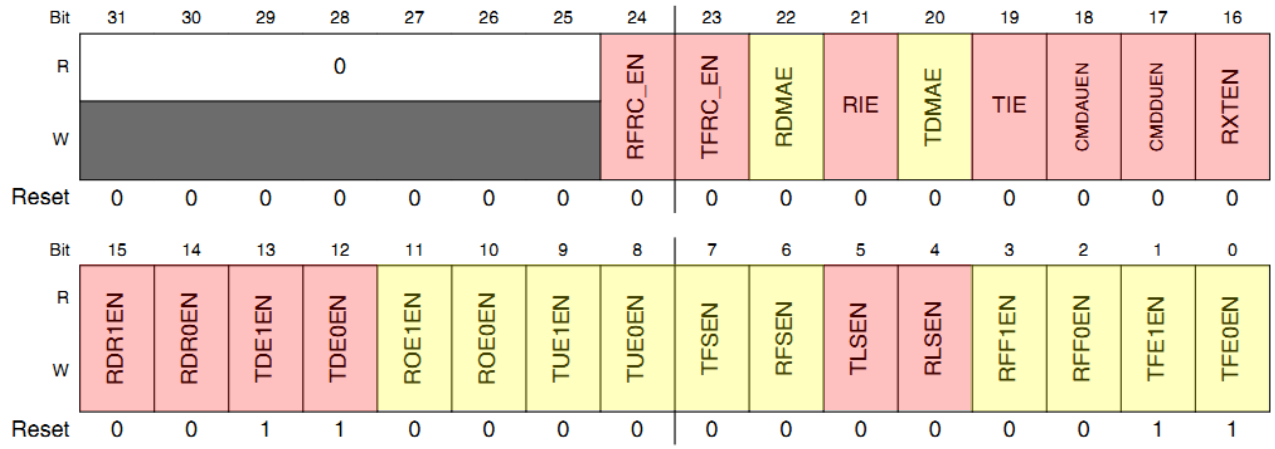


Figure 37. I2S0_IER—SSI

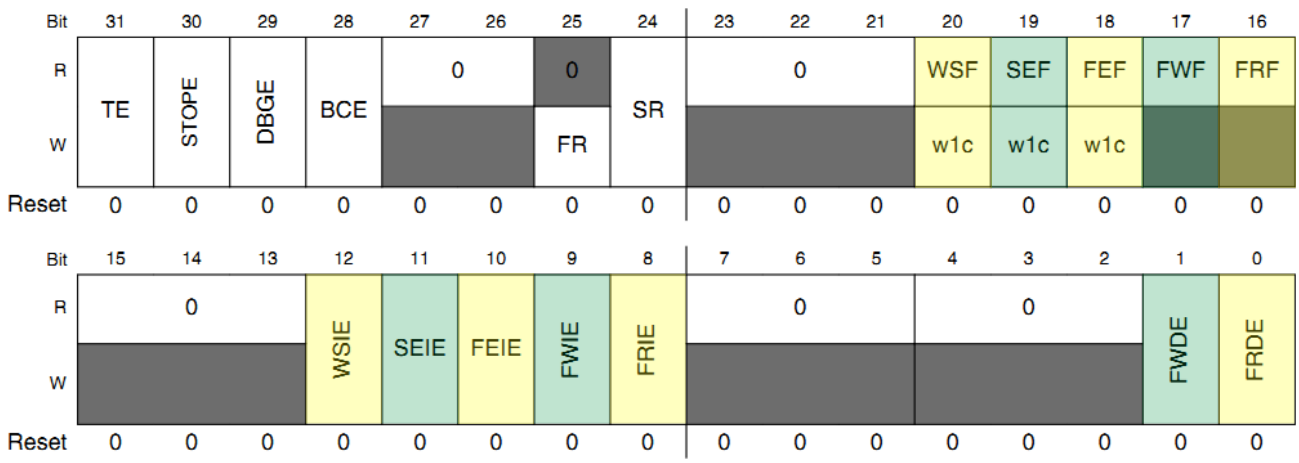


Figure 38. I2S0_TCSR—SAI

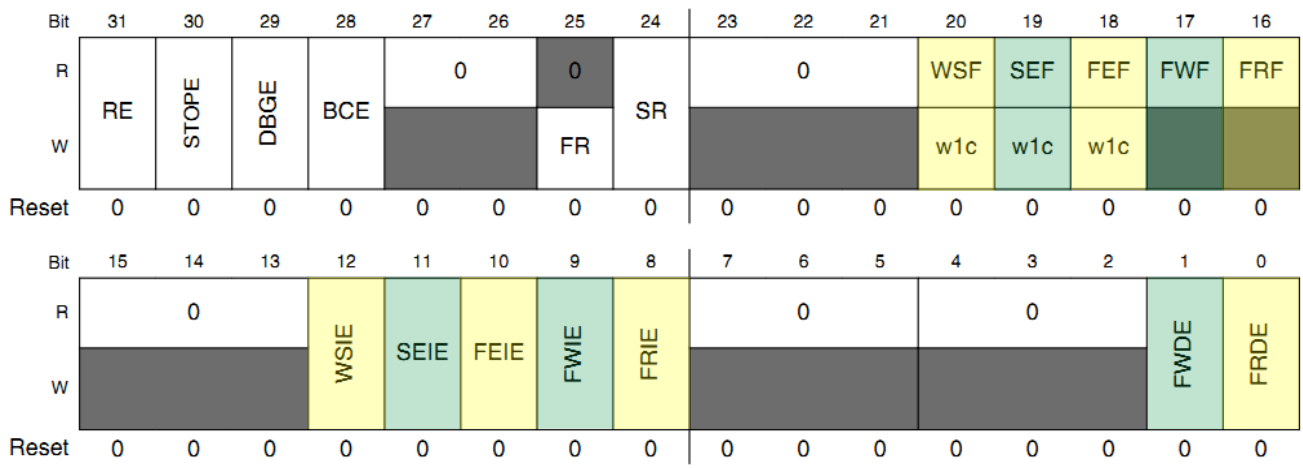


Figure 39. I2S0_RCSR—SAI

New bits/fields added:

- **SEF:** Sync Error Flag
- **FWF:** FIFO Warning Flag
- **SEIE:** Sync Error Interrupt Enable

- **FWIE:** FIFO Warning Interrupt Enable
- **FWDE:** FIFO Warning DMA Enable

Changed bits/fields name:

- ROE1 & ROE0 → RCSR[FEF] (Receive FIFO Error, overrun)
- TUE1 & TUE0 → TCSR[FEF] (Transmit FIFO Error, underrun)
- TFS → TCSR[WSF] (Transmit Frame Sync)
- RFS → RCSR[WSF] (Receive Frame Sync)
- RFF1 & RFF0 → RCSR[FRF] (Receive FIFO Request, received data greater than watermark)
- TFE1 & TFE0 → TCSR[FRF] (Transmit FIFO Request, transmit data less than watermark)
- RDMAE → RCSR[FRDE] (Receive FIFO Request DMA Enable)
- TDMAE → TCSR[FRDE] (Transmit FIFO Request DMA Enable)
- ROE1EN & ROE0EN → RCSR[FEIE]
- TUE1EN & TUE0EN → TCSR[FEIE]
- TFSEN → TCSR[WSIE]
- RFSEN → RCSR[WSIE]
- RFF1EN and RFF0EN → RCSR[FRIE]
- TFE1EN and TFE0EN → TCSR[FRIE]

Removed bits/fields:

- **TFRC and RFRC:** Transmit and Receive Frame Complete
- **TLS and RLS:** Transmit and Receive Last Time Slot
- **RDR1 and RDR0:** Receive Data Ready
- **TDE1 and TDE0:** Transmit Data Empty
- **CMDAU:** Command Address Register Updated
- **CMDDU:** Command Data Register Updated
- **RXT:** Receive Tag Updated

Associated interrupt enable bits:

- TFRCEN and RFRCEN,
- TLSEN and RLSEN,
- RDR1EN and RDR0EN,
- TDE1EN, and
- TDE0E, CMDAUEN, CMDDUEN, RXTEN, RIE, TIE

3.7.2.3 Transmit and receive configuration register

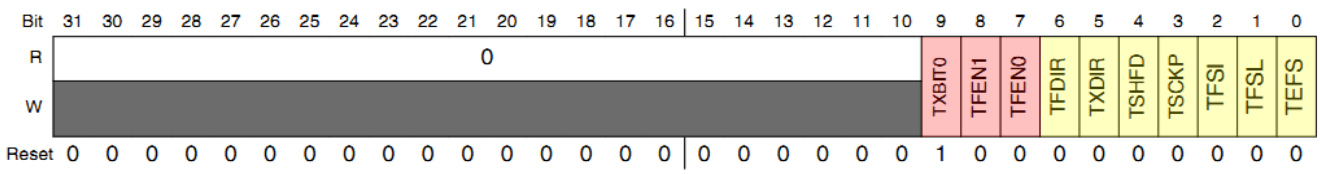


Figure 40. I2S0_TCR—SSI

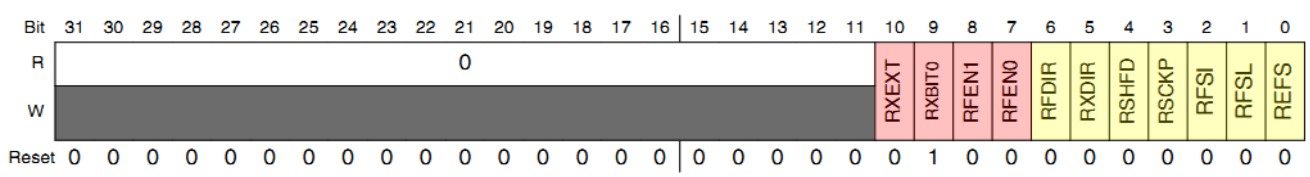


Figure 41. I2S0_RCR—SSI

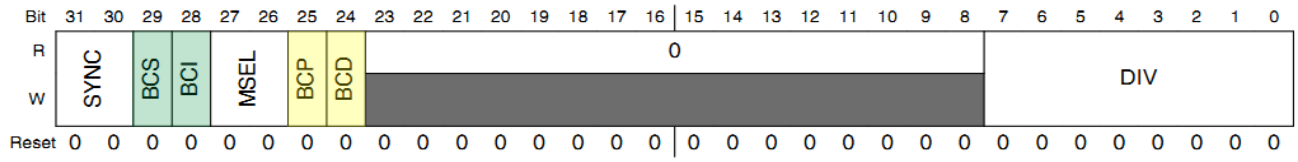


Figure 42. I2S0_TCR2 and I2S0_RCR2—SAI

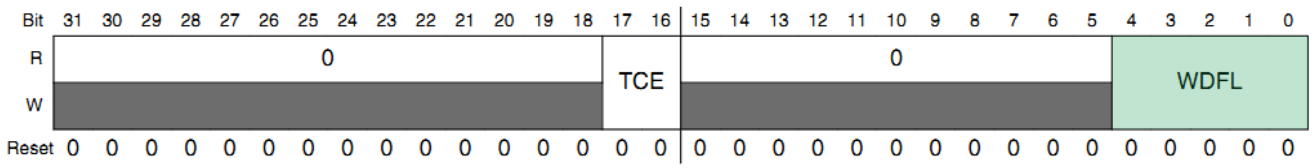


Figure 43. I2S0_TCR3—SAI

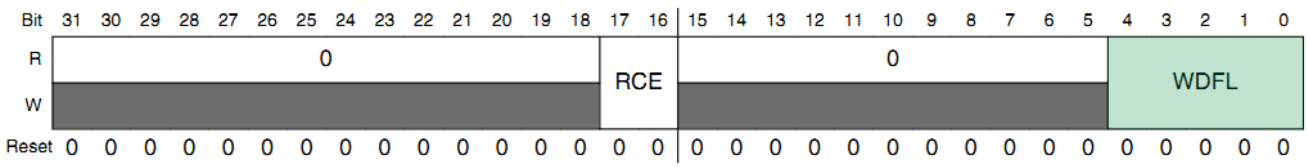


Figure 44. I2S0_RCR3—SAI

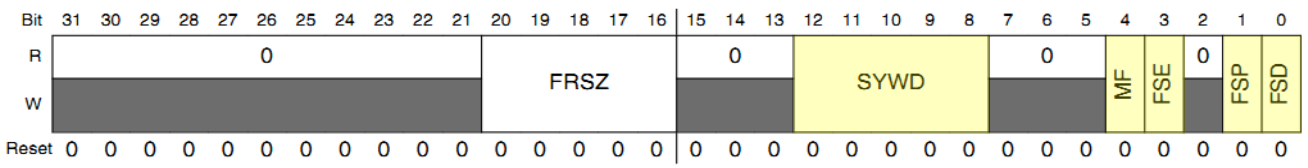


Figure 45. I2S0_TCR4 and I2S0_RCR4—SAI

New bits/fields added:

- **BCS:** Bit Clock Swap
- **BCI:** Bit Clock Input
- **WDFL:** Configures which word start of word flag is set

Changed bits/fields name:

- TSCKP and RSCKP → BCP (Bit Clock Polarity)
- TXDIR and RXDIR → BCD (Bit Clock Direction)
- TFSL and RFSL → SYWD (Frame sync length)
- TSHFD and RSHFD → MF (MSB or LSB transmit first)
- TEFS and REFS → FSE (Frame Sync Early)
- TFSI and RFSI → FSP (Frame Sync Polarity)
- TFDIR and RFDIR → FSD (Frame Sync Direction)

Removed bits/fields:

- **TXBIT0 and RXBIT0:** MSB aligned or LSB aligned
- **TFEN0 and TFEN1:** Transmit FIFO Enable
- **RXEXT:** Receive Sign Extension
- **RFEN0 and RFEN1:** Receive FIFO Enable

3.7.2.4 Transmit and receive clock configuration register

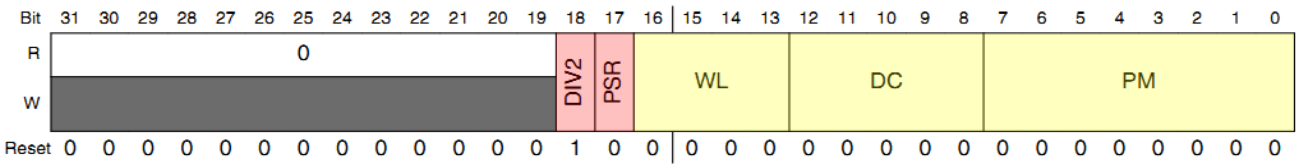


Figure 46. I2S0_TCCR and I2S0_RCCR—SSI

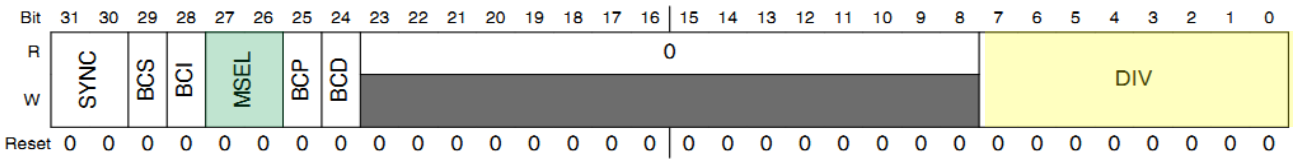


Figure 47. I2S0_TCR2 and I2S0_RCR2—SAI

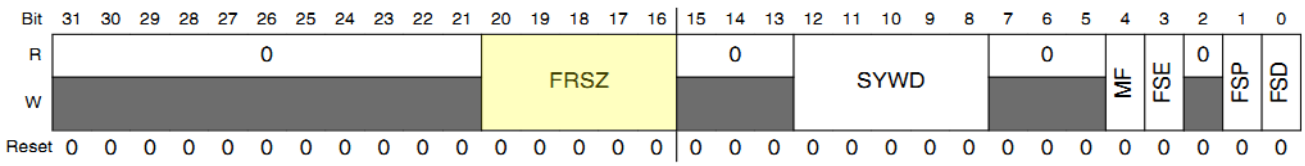


Figure 48. I2S0_TCR4 and I2S0_RCR4—SAI

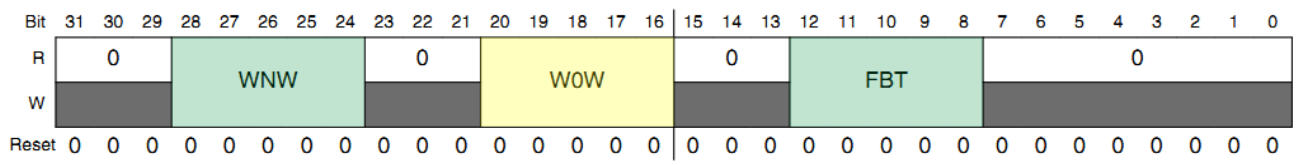


Figure 49. I2S0_TCR5 and I2S0_RCR5—SAI

New bits/fields added:

- MSEL: Master Clock Select before it is divided down to bit clock
- WNW: Configures number of bits of each word, except the first word in a frame
- FBT: Configures the bit index of the first bit transmitted in each word in a frame

Changed bits/fields name:

- DC → FRSZ (number of words in each frame)
- WL → W0W (number of bits in each word)
- PM → DIV (bit clock prescaler)

Removed bits/fields:

- DIV2: Whether or not to divide by 2
- PSR: Prescaler divide by 8

3.7.2.5 FIFO control and status register

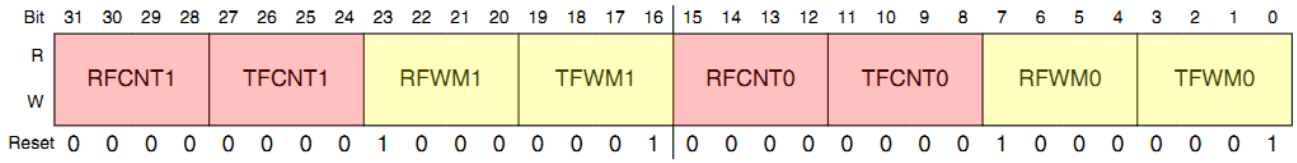


Figure 50. I2S0_FCSR—SSI

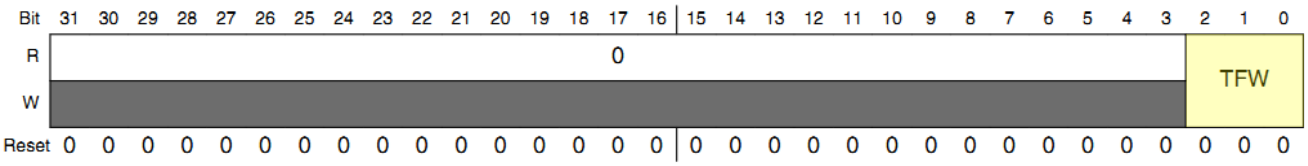


Figure 51. I2S0_TCR1—SAI

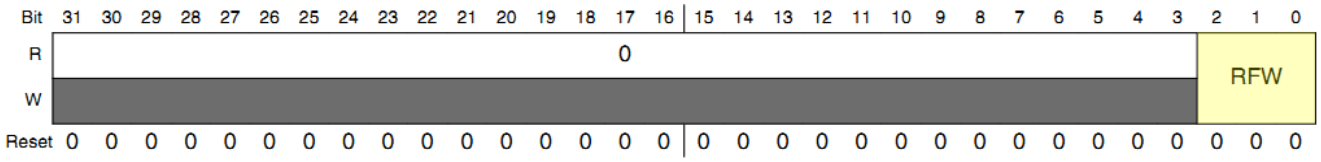


Figure 52. I2S0_RCR1—SAI

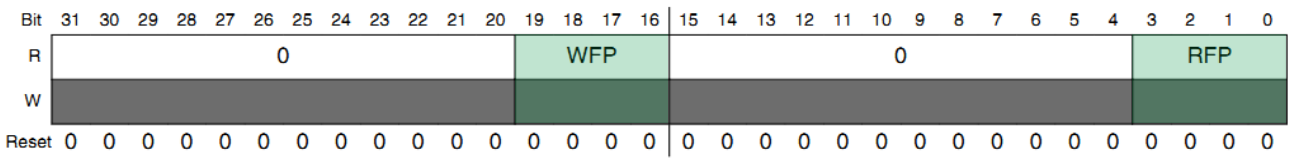


Figure 53. I2S0_TFR0, I2S0_TFR1, I2S0_RFR0, and I2S0_RFR1—SAI

New bits/fields added:

- **WFP:** Write FIFO Pointer
- **RFP:** Read FIFO Pointer

Changed bits/fields name:

- RFWM0 and RFWM1 → RFW (Receive FIFO Watermark)
- TFWM0 and TFWM1 → TFW (Transmit FIFO Watermark)

Removed bits/fields:

- RFCNT0 and RFCNT1: Receive FIFO Counter
- TFCNT0 and TFCNT1: Transmit FIFO Counter

3.7.2.6 Master clock generation register

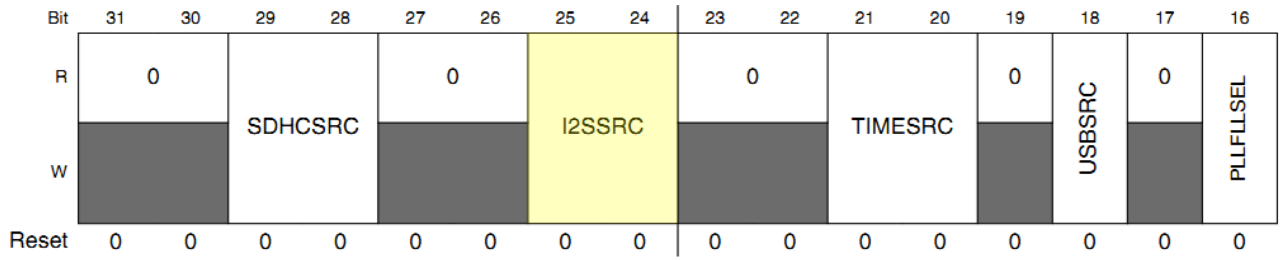


Figure 54. SIM_SOPT2—SSI

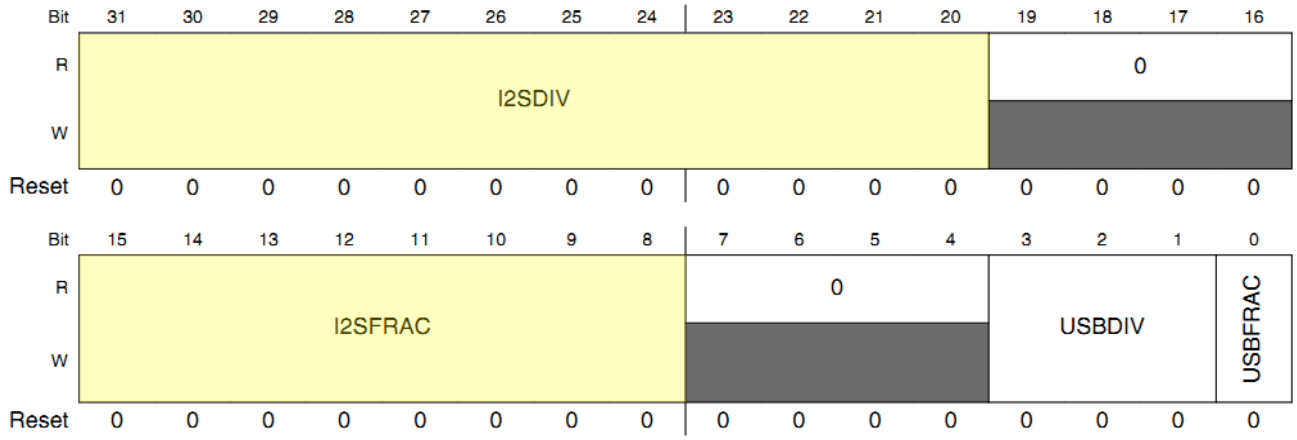


Figure 55. SIM_CLKDIV2—SSI

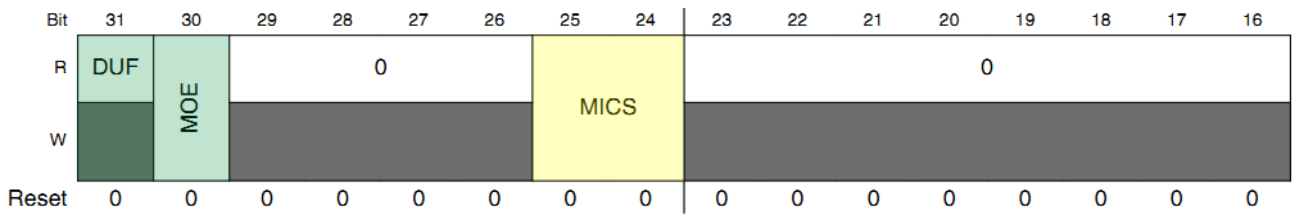


Figure 56. I2S0_MCR—SAI

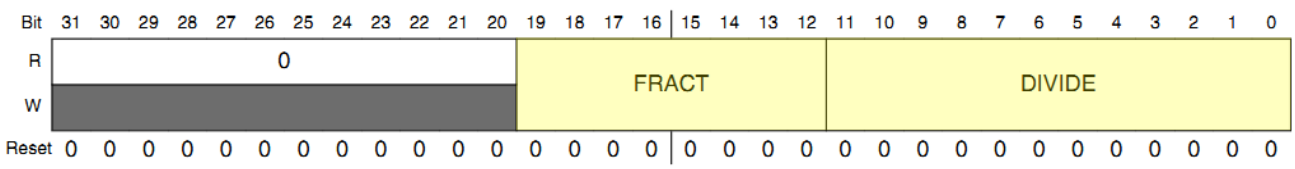


Figure 57. I2S0_MDR—SAI

New bits/fields added:

- DUF: Divide Update Flag
- MOE: Master Clock Output Enable

Changed bits/fields name:

- I2SSRC → MICS (Master Clock Source Select)
- I2SFRAC → FRACT (Clock Divider Fraction)

3.7.3 Software impact

There is not much change in software configuration for the SAI module. Bit clock and frame sync generation, their direction and polarity selection, frame size, and word size setting are the same as for the SSI module. Only the bit naming is different.

With the option to trigger DMA with a TX FIFO empty or RX FIFO full condition, it is easier to program the DMA controller for automatic data transfer by simply programming the total transfer size to be the same as the FIFO depth for the TX and RX FIFO. If instead the FIFO watermark condition is used to trigger DMA, take care to program the DMA controller with a total transfer size less than (FIFO depth – FIFO watermark); otherwise, either TX FIFO is pushed with more data than it can hold, or the RX FIFO is pulled with more data that has been received.

Also note that there is no longer any FIFO count information; instead we have write and read FIFO pointer, so the actual FIFO entry needs to be calculated in your code. Also you need to include logic to determine whether the FIFO is empty or full, and to stop writing data when the FIFO is full or reading data when the FIFO is empty.

3.7.4 Hardware impact

There is not much hardware change from the SSI to the SAI. There is one more data channel for TX and RX, so there are I2S0_TXD0 and I2S0_TXD1, as well as I2S0_RXD0 and I2S0_RXD1. It makes no difference whether you use channel 0 or 1 for transmit or receive data. Since each channel has its own FIFO, each channel can send or receive same or different data and do this at the same time they share the same frame sync and bit clock.

3.8 TSI Version 1 to TSI Version 2

This section addresses the specific differences between the TSI module version 1 and version 2. The TSI module has been simplified in the new version to simplify configuration. The current ranges for the oscillator remain the same, but the number of possible configuration values has been reduced by half. Delta voltage configuration was removed; now, a constant delta voltage is used. Threshold registers used to generate an interrupt when capacitance increases have been removed for each channel; only one register was kept for the low-power wake-up channel, exclusively. Linked to this change, the status register for all these thresholds was also removed; now the threshold is only used to wake up. The changes in software needed to adjust for these differences are detailed in the sections below.

3.8.1 Memory map comparison

	Kinetis Rev. 100 MHz 1.x (TSI 1.0)		Kinetis 120 MHz (TSI 2.0)	
	Location	**Name**	**Location**	**Name**
General Control and Status Register	4004_5000	TSI0_GENCS	4004_5000	TSI0_GENCS
SCAN control register	4004_5004	TSI0_SCANC	4004_5004	TSI0_SCANC
Pin enable register	4004_5008	TSI0_PEN	4004_5008	TSI0_PEN
Status Register	4004_500C	TSI0_STATUS	N/A	N/A
Counter Registers	4004_5100–4004_511C	TSI0_CNTRn	4004_5100–4004_511C	TSI0_CNTRn
Channel threshold register(s)	4004_5120–4004_515C	TSI0_THRESHLDn	4004_5120	TSI0_THRESHOLD
Wake-up counter	N/A	N/A	4004_500C	TSI_WUCNTR

- **GENCS**: No change, but the out-of-range setting is now only operational in low-power modes.

- **SCANC: CAPTRM** removed (internal cap is now fixed to 1 pF), **REFCHRG** and **EXTCHRG** have been scaled down from 5 bits to 4 bits. Change involves reducing the amount of possible current values from 32 to 16. Current range is now from 2 µA to 32 µA in increments of 2. **DELVOL** eliminated, delta voltage value is now fixed. AMCLKS clock sources changed: bus clock reference was changed for **LPOCLK**. **AMCLKSDIV** was also removed because it was only practical with bus clock as reference. Instead of dividing down the clock source, now a slower, lower power source can be used. This reduces overall power consumption of the module. If a slow source is needed, use **LPOCLOCK** and adjust with **AMPSC**. If a faster source is needed, use **MCGIRCLK** or **OSCERCLK**.

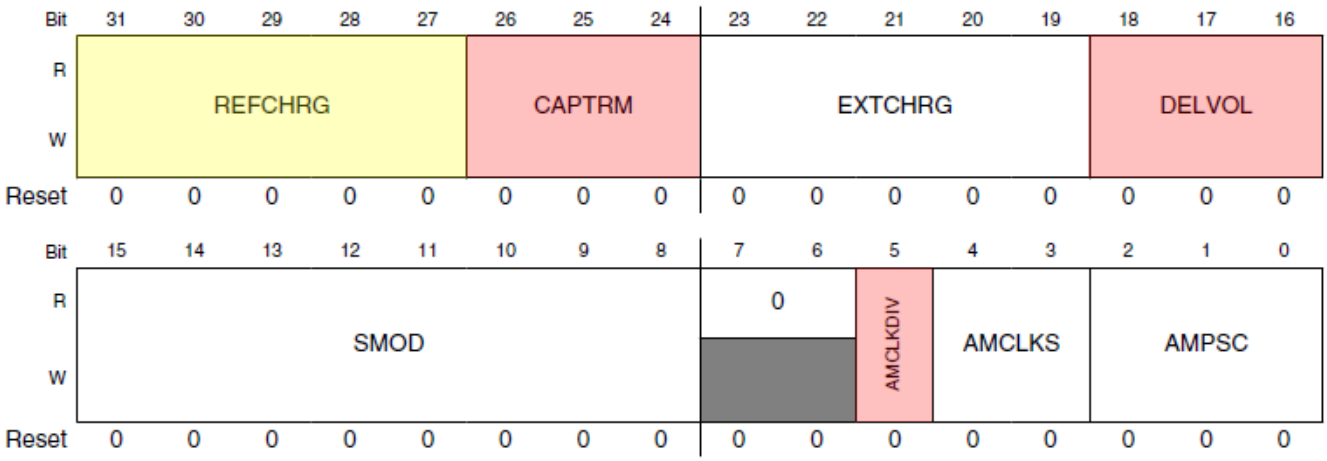


Figure 58. TSI_SCANC—TSI Rev. 1

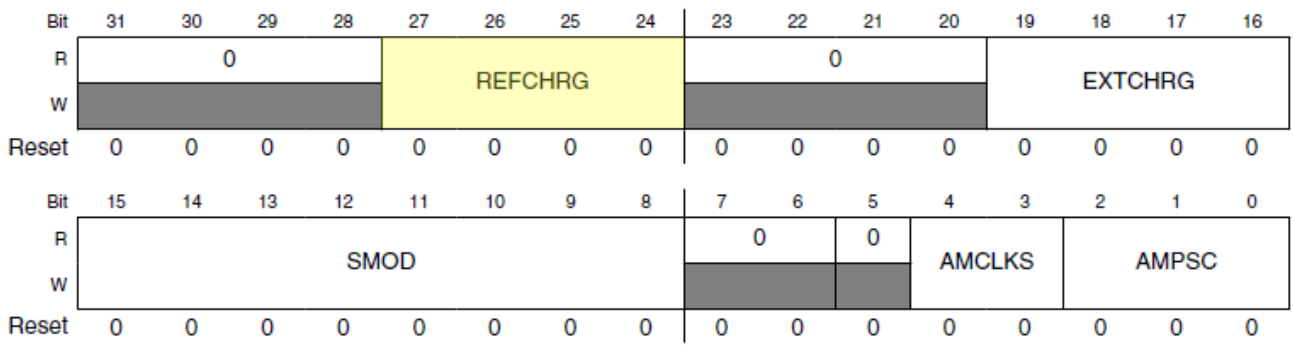
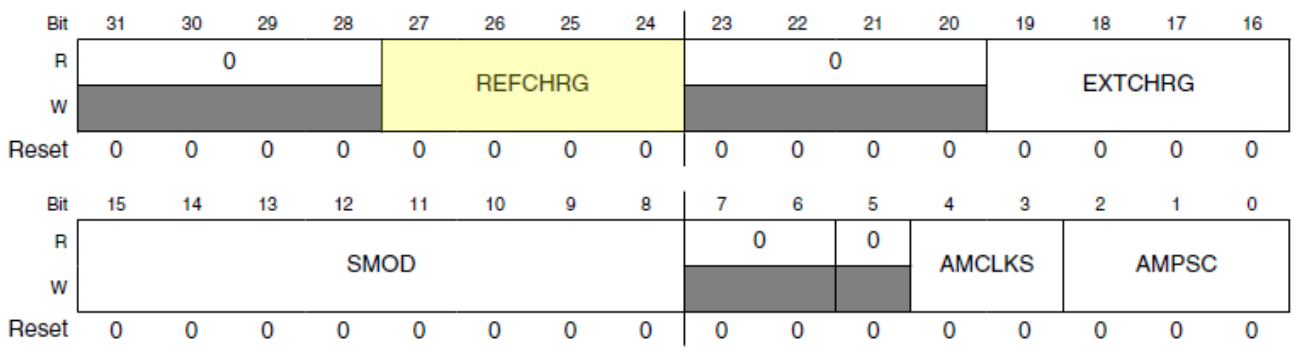


Figure 59. TSI_SCANC—TSI Rev. 2

- **TSI_PEN:** No change
- **TSI_WUCNTR:** New register, stores the value of the counter for the low power electrode upon wakeup.



- **TSI_STATUS:** Removed

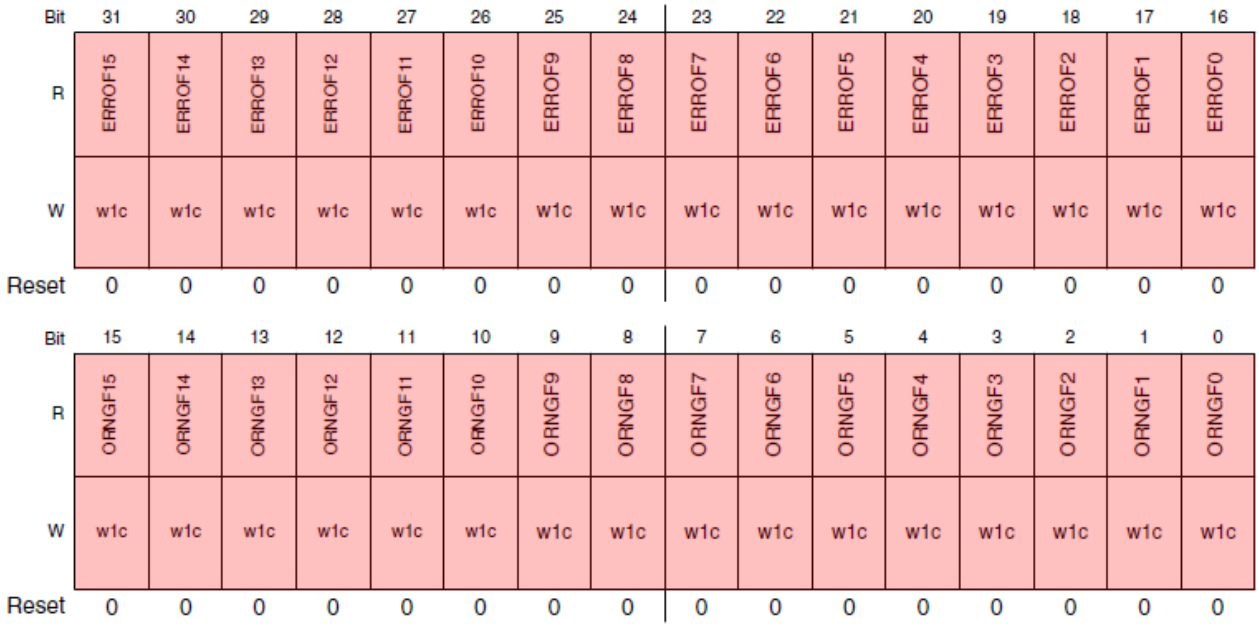


Figure 60. TSI_STATUS (completely removed)

- **TSI_CNTR:** No change
- **TSI_THRESHLDn:** Used to be one threshold register for each electrode; now there is a single threshold register for the selected wakeup electrode because the out-of-range function now only operates in low-power modes.

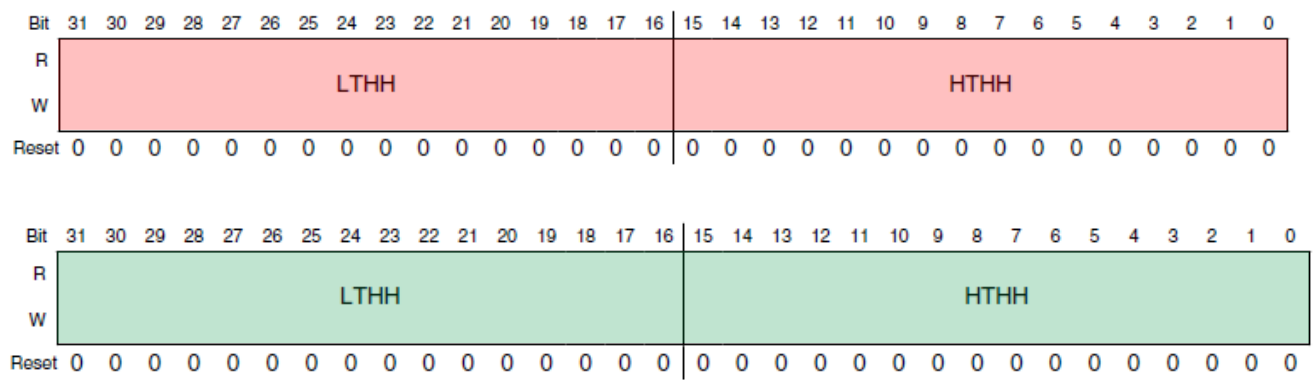


Figure 61. Single TSI_THRESHOLD for wakeup instead of separate registers for each channel

3.8.2 Software impact

- No delta voltages register because there is no difference between internal delta voltage and the external, so the frequency relationship is not affected. The highest voltage that can be output has been selected as a fixed voltage to increase EMC tolerance. As the change in delta voltages does not affect the relationship between oscillators and thus, the counters, this change does not affect software other than the need to remove the code that configured this register.
- Less current for external and internal oscillators. Touch sensing does not require the level of granularity that was in the first TSI version, so the number of values has been scaled down to 16. It is important to note that this change caused the EXTCHRG and REFCHRG registers to go from 5 to 4 bits, meaning that it is necessary to scale down the value. Make sure to take care of this modification. If the value used previously is now not valid, choose either the next value up or down depending on convenience.

- The change of bus clock to LPOCLK for timing reference in AMCLKS means that the prescaler (AMPSC) and modulo (SMOD) need to be adjusted to the LPOCLK, which is a 1 kHz reference. If the other two clock sources were used (MCGIRCLK or OCSERCLK), no change is needed.
- Only one threshold register instead of one for each counter. Consider that the wakeup source can wake up from EOSF (end of scan) or out of range. Because out of range is only available in the low leakage modes (LLS and VLLSx), it is recommended that any of these be used to wake up with TSI. If another mode is needed (like Stop, Low-power Stop, Wait, and so on) and it is needed to wake-up with TSI, then it is important to have a periodic wakeup to check the TSI status. The periodic wakeup can be an external source like RTC or MTIM, or can be the TSI module by configuring the scan time to be very slow (for example, every two or three seconds). This will cause reaction time to be slow, but also lower power.
- Remove references to TSI_STATUS, as it is no longer available.
- Remove references to the numbered TSI_THRESHLDn registers; remember now only one TSI_THRESHOLD register, with the OURGF flag and interrupt, is used for wakeup.
- If the first counter value upon wakeup is needed (from the configured wake-up electrode), use TSI_WUCNTR to read this value. If not, wait for next scan and read directly from the counter registers.

3.8.3 Hardware impact

There is no hardware impact. The signaling and measurement algorithm are the same.

4 Enhanced Modules

4.1 System Integration Module (SIM)

4.1.1 Impacted registers

The SIM's memory map only adds a few registers, but there are many changes to fields within the existing registers as well. There are additions associated with the control of new modules and additional instantiations of modules and new functionality. The function of some fields is slightly modified. Some control functions have been relocated to other modules, and therefore the SIM bits previously associated with these functions are removed.

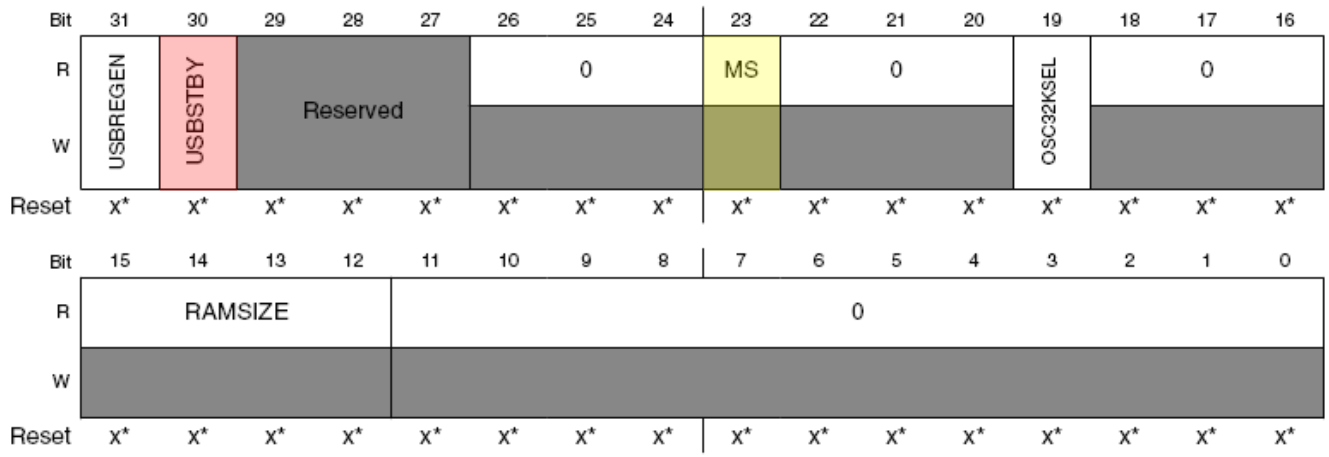
The table below shows the memory map differences between the original SIM from the Kinetis 100 MHz Rev. 1.x devices and the SIM on the 120 MHz devices. The following sections provide information on the field changes within the registers.

Table 9. Memory map comparison

Location	Kinetis 100 MHz Rev. 1.x SIM	Kinetis 120 MHz SIM
0x40047004	N/A	SOPT1 Configuration Register (SIM_SOPT1CFG)
0x40048064	N/A	System Clock Divider Register 3 (SIM_CLKDIV3)
0x40048068	N/A	System Clock Divider Register 4 (SIM_CLKDIV4)
0x4004806C	N/A	Misc Control Register (SIM_MCR)

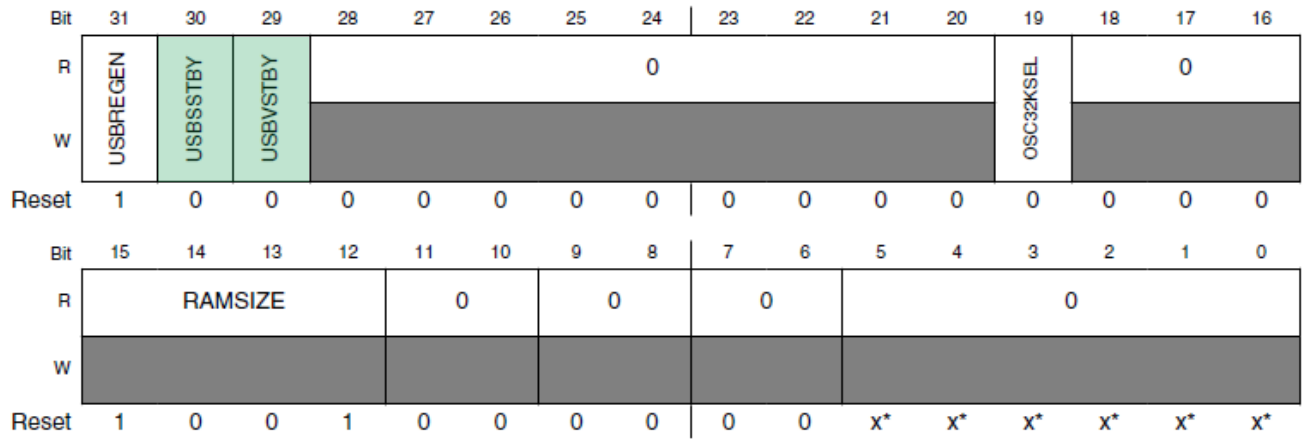
4.1.1.1 SIM SOPT register changes

The changes to the SIM system options registers (SOPT) are shown below.



* Notes:
- x = Undefined at reset.

Figure 62. SIM_SOPT1 register—Kinetis 100 MHz Rev. 1.x



* Notes:
- x = Undefined at reset.

Figure 63. SIM_SOPT1 register—Kinetis 120 MHz

New bits/fields added:

- USBSSTBY
- USBVSTBY

Remove bit/field names:

- USBSTBY

Changed bit/field names:

- MS → EZP_MS moved to the RCM_MR Register

Address: RCM_MR is 4007_F000h base + 7h offset = 4007_F007h

Bit	7	6	5	4	3	2	1	0
Read	0						EZP_MS	0
Write								
Reset	0	0	0	0	0	0	0	0

Figure 64. Mode register (RCM_MR)—Kinetis 120 MHz

Address: SIM_SOPT1CFG is 4004_7000h base + 4h offset = 4004_7004h

Bit	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
R	0					USSWE	UVSWE	URWE	0								0						0		0								
W																																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 65. SIM_SOPT1CFG—Kinetis 120 MHz

SIM_SOPT1CFG is a new register added to control the operation of the USB regulator. Refer to the USB section for more details.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	0		SDHCSRC		0		I2SSRC		0		TIMESRC		0	USBSRC	0	PLLFLLSEL
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0			TRACECLKSEL	CMTUARTPAD	0	FBSL		0							MCGCLKSEL
W																
Reset	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

Figure 66. SIM_SOPT2—Kinetis 100 MHz Rev. 1.x

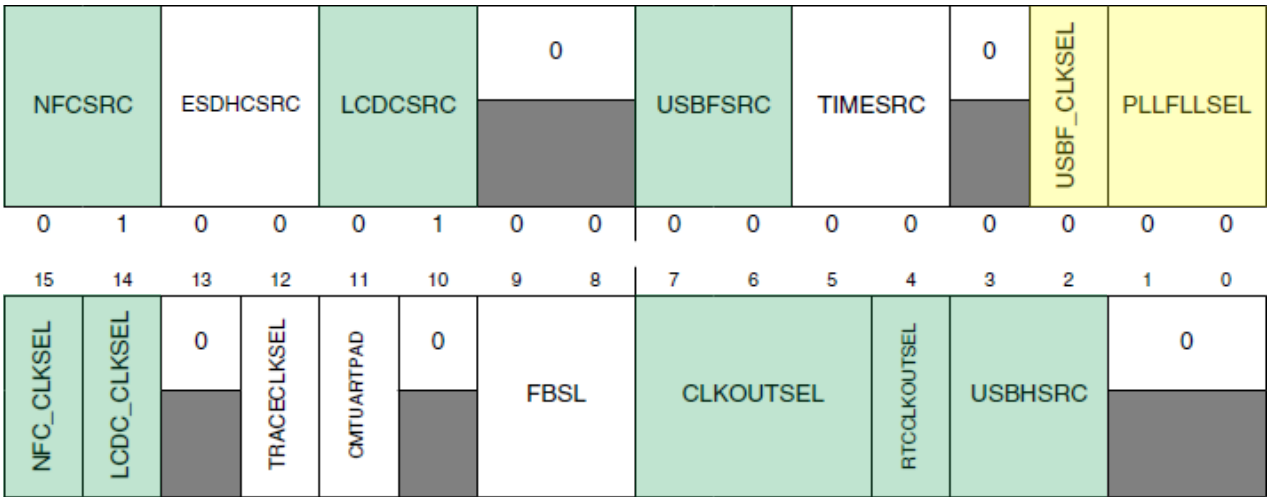


Figure 67. SIM_SOPT2—Kinetis 120 MHz

New bits/fields added:

- NFCSRC - selects the clock divider source for the new NFC module
- LCDCSRC - selects the clock divider source for the new LCDC module
- USBFSRC - clock select for the new USBFS fractional divider
- NFC_CLKSEL - selects between the NFC clock divider and EXTAL1 clock
- LCDC_CLKSEL - selects between the LCDC clock divider and EXTAL1 clock
- CLKOUTSEL - selects the internal clock to output on the CLKOUT pin
- RTCCLKOUTTSEL - Selects the clock to output on the RTC_CLKOUT pin (32 kHz or 1 Hz)
- USBHSRC - selects the clock divider source for the new USB HS module

Remove bit/field names:

- I2SSRC
- MCGCLKSEL

Changed bit/field names:

- USBSRC → USBFSRC bit field renamed to differentiate between USB FS and USB HS
- PLLFLLSEL → PLLFLLSEL field expanded to two bits to allow for selection of new internal clocks. The encodings are compatible between revisions, so code changes are not required to keep the same clock selection.

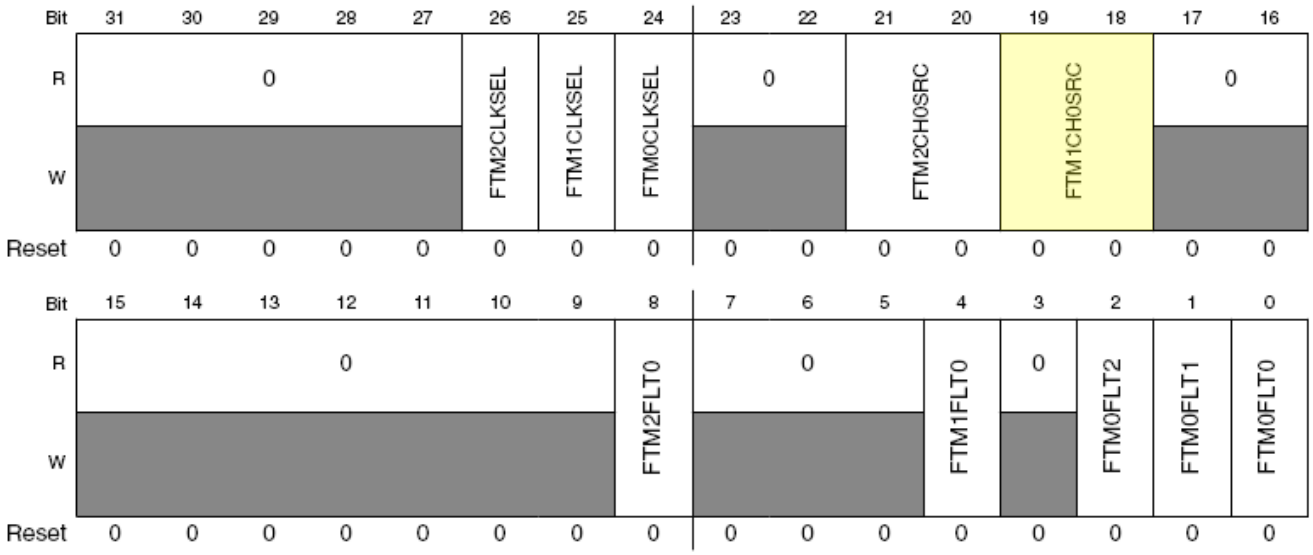


Figure 68. SIM_SOPT4—Kinetis 100 MHz Rev. 1.x

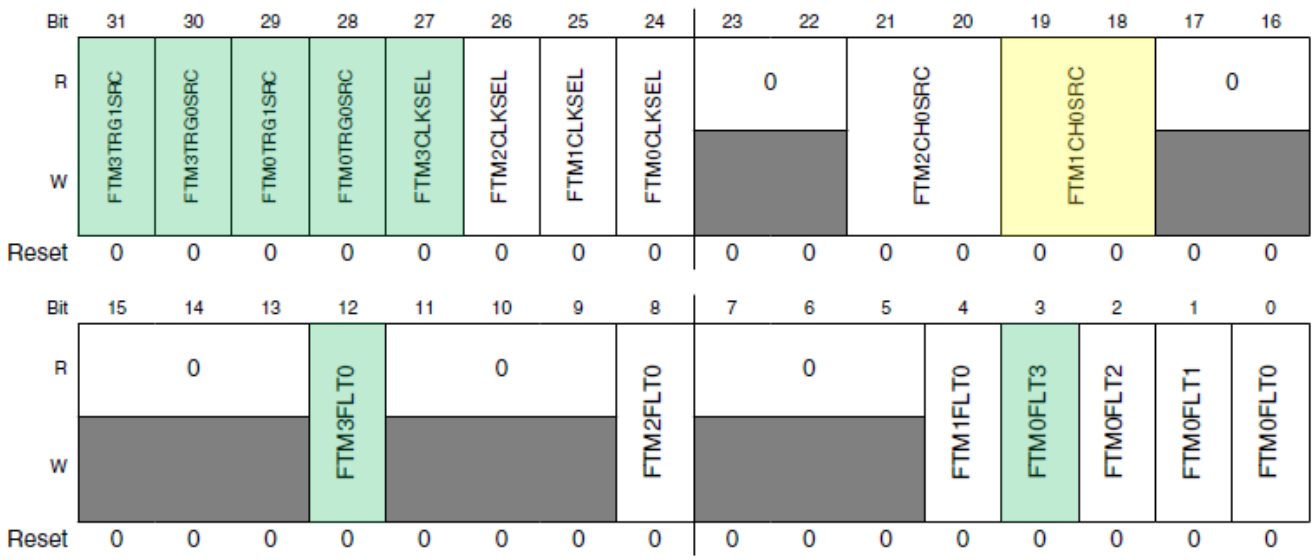


Figure 69. SIM_SOPT4—Kinetis 120 MHz

New bits/fields added:

- FTM3TRG1SRC, FTM3TRG0SRC, FTM0TRG1SRC, and FTM0TRG0SRC - new hardware trigger source selects
- FTM3CLKSEL - Clock select for new FTM instantiation
- FTM3FLT0 - Fault select for new FTM instantiation
- FTM0FLT3 - FTM0 fault 3 selection

Changed bit/field names:

- FTM1CH0SRC → Added option for the USB Start-of-Frame Pulse

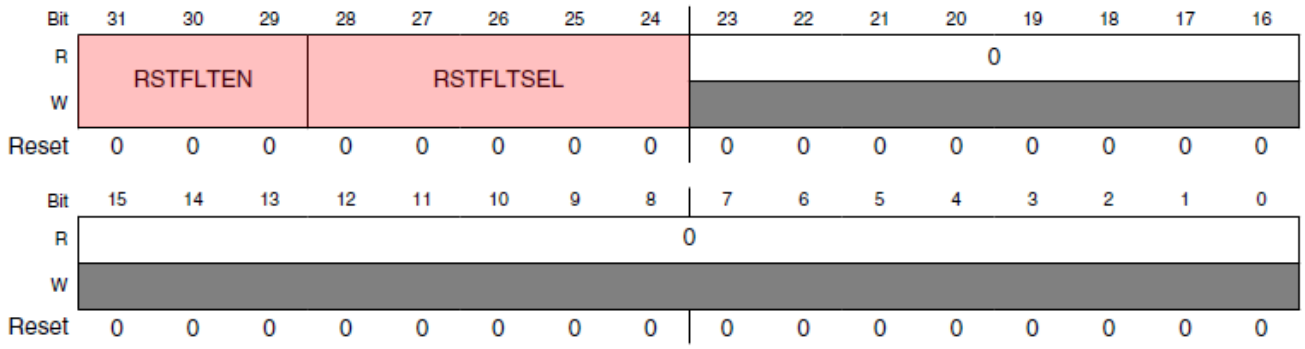


Figure 70. SIM_SOPT6—Kinetis 100 MHz Rev. 1.x

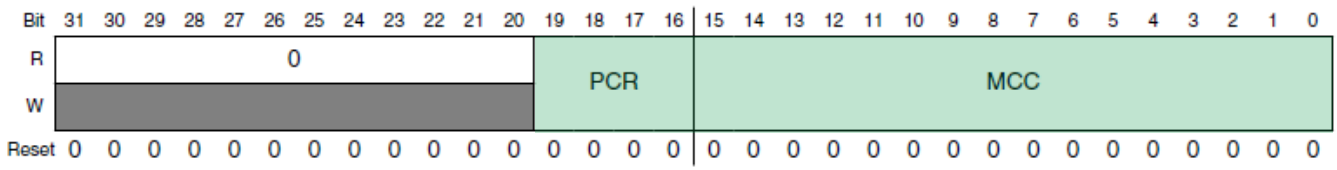


Figure 71. SIM_SOPT6—Kinetis 120 MHz

New bits/fields added:

- PCR and MCC -These fields control hold times for sharing of pins between the NFC and FlexBus

Remove bit/field names:

- RSTFLTEN and RSTFLTSEL - the reset pin filter functionality has moved to the RCM
- MCGCLKSEL

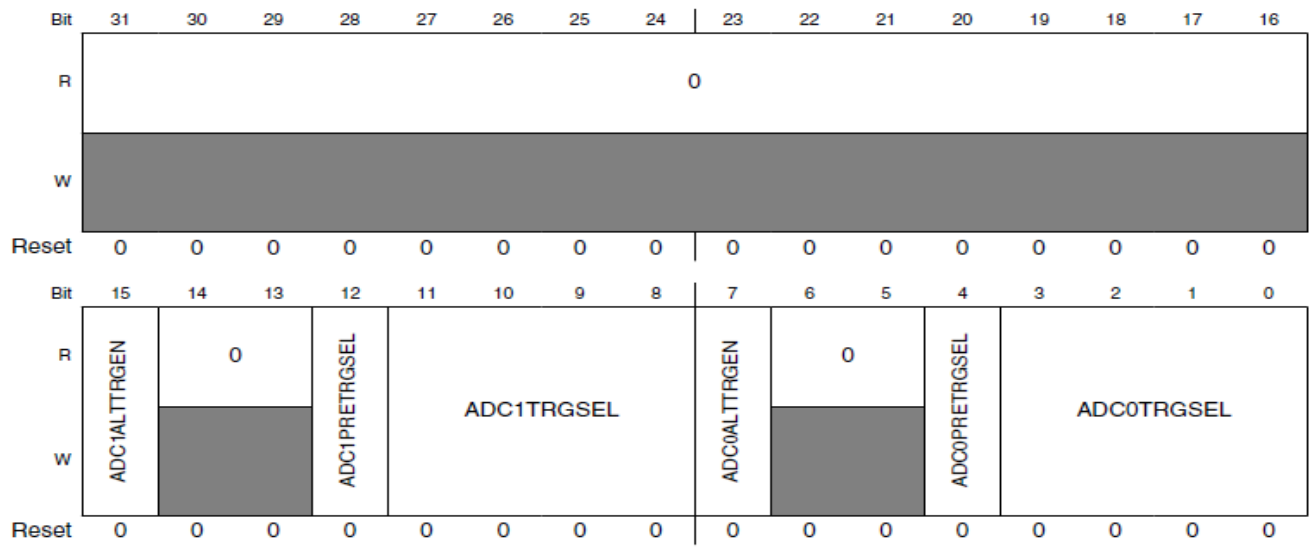


Figure 72. SIM_SOPT7—Kinetis 100 MHz Rev. 1.x

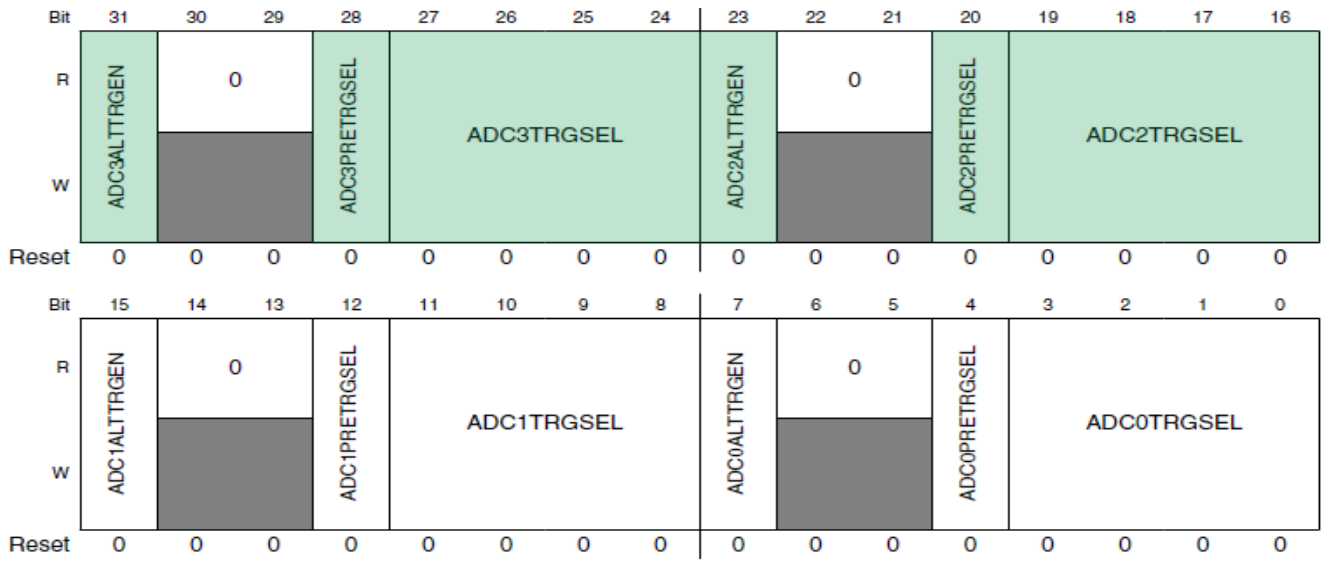


Figure 73. SIM_SOPT7—Kinetis 120 MHz

New bits/fields added:

- ADC3ALTTRGEN, ADC3PRETRGSEL, ADC3TRGSEL - bits added to control new ADC instantiation
- ADC2ALTTRGEN, ADC2PRETRGSEL, ADC2TRGSEL - bits added to control new ADC instantiation

4.1.1.2 SIM SCGC register changes

The SIM module has a number of system clock gating control registers. New bits have been added to many of the registers to add clock gating control bits for new modules that were not present on the Kinetis 100 MHz Rev 1.x devices. The clock gating registers are backwards compatible, so changes to the clock gating are only required to use new modules and/or new module instantiations. Code does not need to be modified to enable the clock gate for modules that were available on the Kinetis 100 MHz Rev. 1.x MCUs.

4.1.1.3 SIM CLKDIV register changes

The SIM module also adds clock dividers for new modules including the USB HS, LCDC, and NFC. The fields to control each these clock dividers can be found in SIM CLKDIV*n* registers. These fields are all associated with the addition of new modules, so the changes to these registers are only a concern if using one of the new modules.

The only previously existing module with new bits in the CLKDIV registers is the trace clock. Because of the higher CPU frequency, a trace clock divider has been added to the 120 MHz Kinetis devices. By default the trace clock will be divided by two.

4.1.1.4 SIM miscellaneous control register

The last register for the SIM module is the addition of a miscellaneous control register (MCR). This module has fields for:

- Trace clock enable/disable
- ULPI clock direction
- PDB loop mode
- LCD display start/stop
- DDR RCR reset status
- DDR RCR reset request
- DDR pad configuration
- DDR DQS enable
- DDR I/O enable/disable

- DDR self refresh status
- DDR self refresh enable

4.1.2 Software impact

Because the SIM is responsible for control and configuration of many module features, many applications might be impacted by the changes to the SIM module. Although some applications might not require any software changes, the SIM is one of the areas most likely to require software updates. The exact changes needed will depend heavily on the modules that are being used and the desired configuration and clocking of those modules. In order to use new modules, initialization of the SIM clock gating and any other control fields associated with the new module will be required.

Operating the Kinetis MCU with DDR memory in low power modes, VLPS, VLPR, STOP, LLS and VLLSx requires SIM_MCR and DDR memory controller settings to minimize the power being used by the DDR controller and memory. The bits in the SIM_MCR register that control the pads and analog circuit configuration should be properly set to minimize leakage current while in the low power modes. The external DDR memory device should also be placed in a DDR low power mode as well. See the SIM_MCR register description and the DDR memory controller section in the Reference Manual for details of these modes.

4.1.3 Hardware impact

The SIM isn't directly associated with external signals, so the SIM changes do not require hardware changes. However, some of the modules impacted by the SIM changes might require hardware updates.

4.2 Direct Memory Access Controller (eDMA)

4.2.1 Features

The Kinetis 100 MHz devices support 16 channels on the EDMA. Due to the increased on-chip integration for the 120 MHz devices, the number of DMA channels has been increased to 32.

4.2.2 Impacted registers

In order to support the new DMA channels there are some changes to the eDMA's memory map. All of the changes are additions. Any of the existing bits that were previously in the 100 MHz Kinetis devices remain the same.

The table below shows the new eDMA transfer control descriptor (TCD) registers that have been added to the eDMA in order to support the increased number of channels. The following section lists register fields that have been added.

Table 10. Memory map comparison

Register	16-channel eDMA Locations	32-channel eDMA Locations
DMA_TCD*n*_SADDR	0x4000_9000 + (32d x *n*), where *n* = 0d to 15d	0x4000_9000 + (32d x *n*), where *n* = 0d to 31d
DMA_TCD*n*_SOFF	0x4000_9004 + (32d x *n*), where *n* = 0d to 15d	0x4000_9004 + (32d x *n*), where *n* = 0d to 31d

Table continues on the next page...

Table 10. Memory map comparison (continued)

Register	16-channel eDMA Locations	32-channel eDMA Locations
DMA_TCD*n*_ATTR	0x4000_9006 + (32d x *n*), where *n* = 0d to 15d	0x4000_9006 + (32d x *n*), where *n* = 0d to 31d
DMA_TCD*n*_NBYTES	0x4000_9008 + (32d x *n*), where *n* = 0d to 15d	0x4000_9008 + (32d x *n*), where *n* = 0d to 31d
DMA_TCD*n*_SLAST	0x4000_900C + (32d x *n*), where *n* = 0d to 15d	0x4000_900C + (32d x *n*), where *n* = 0d to 31d
DMA_TCD*n*_DADDR	0x4000_9010 + (32d x *n*), where *n* = 0d to 15d	0x4000_9010 + (32d x *n*), where *n* = 0d to 31d
DMA_TCD*n*_DOFF	0x4000_9014 + (32d x *n*), where *n* = 0d to 15d	0x4000_9014 + (32d x *n*), where *n* = 0d to 31d
DMA_TCD*n*_CITER	0x4000_9016 + (32d x *n*), where *n* = 0d to 15d	0x4000_9016 + (32d x *n*), where *n* = 0d to 31d
DMA_TCD*n*_DLASTSGA	0x4000_9018 + (32d x *n*), where *n* = 0d to 15d	0x4000_9018 + (32d x *n*), where *n* = 0d to 31d
DMA_TCD*n*_CSR	0x4000_901C + (32d x *n*), where *n* = 0d to 15d	0x4000_901C + (32d x *n*), where *n* = 0d to 31d
DMA_TCD*n*_BITER	0x4000_901E + (32d x *n*), where *n* = 0d to 15d	0x4000_901E + (32d x *n*), where *n* = 0d to 31d

4.2.2.1 DMA_ERQ

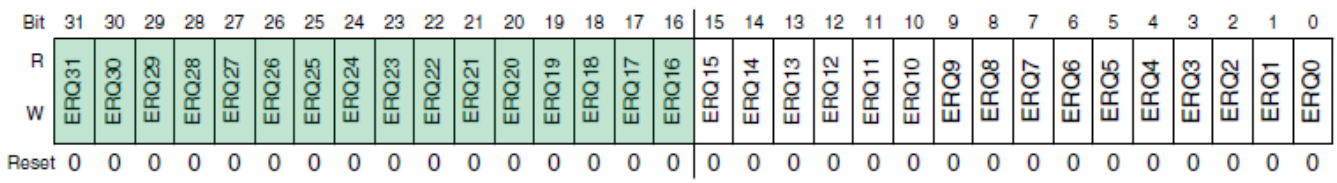


Figure 74. DMA_ERQ—Kinetis 120 MHz

New bits/fields added:

- ERQ31-ERQ16

4.2.2.2 DMA_EEI

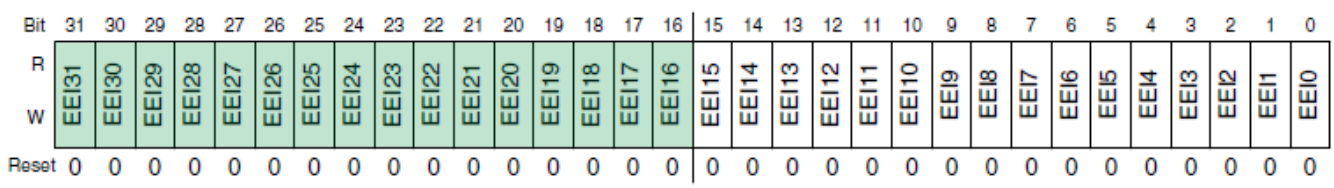
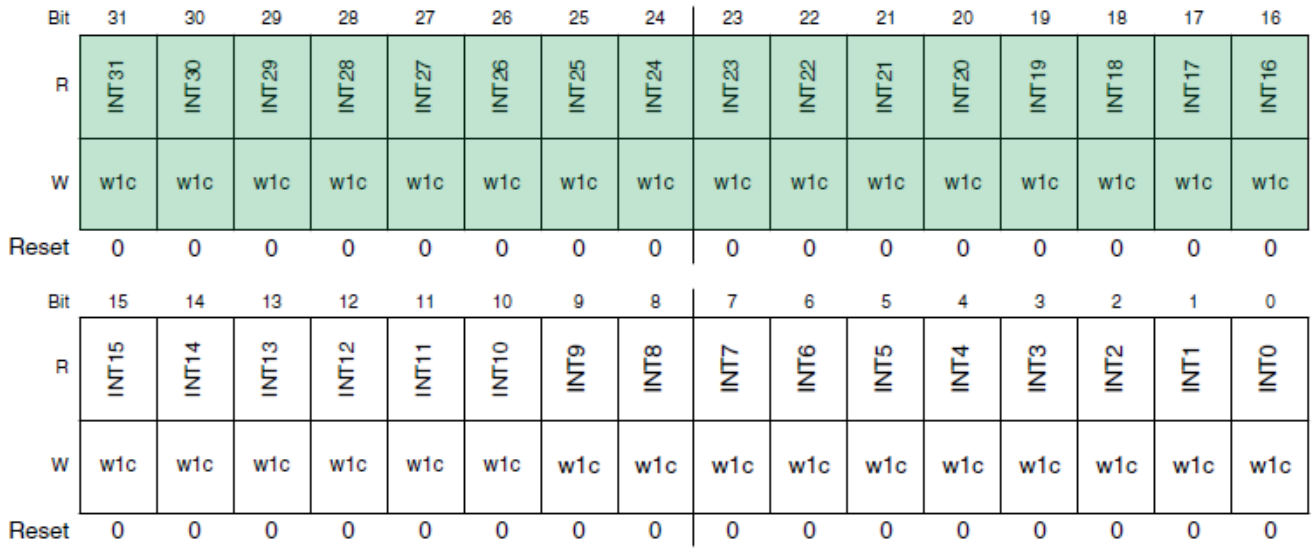


Figure 75. DMA_EEI—Kinetis 120 MHz

New bits/fields added:

- EEI31-EEI16

4.2.2.3 DMA_INT



Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	INT31	INT30	INT29	INT28	INT27	INT26	INT25	INT24	INT23	INT22	INT21	INT20	INT19	INT18	INT17	INT16
W	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

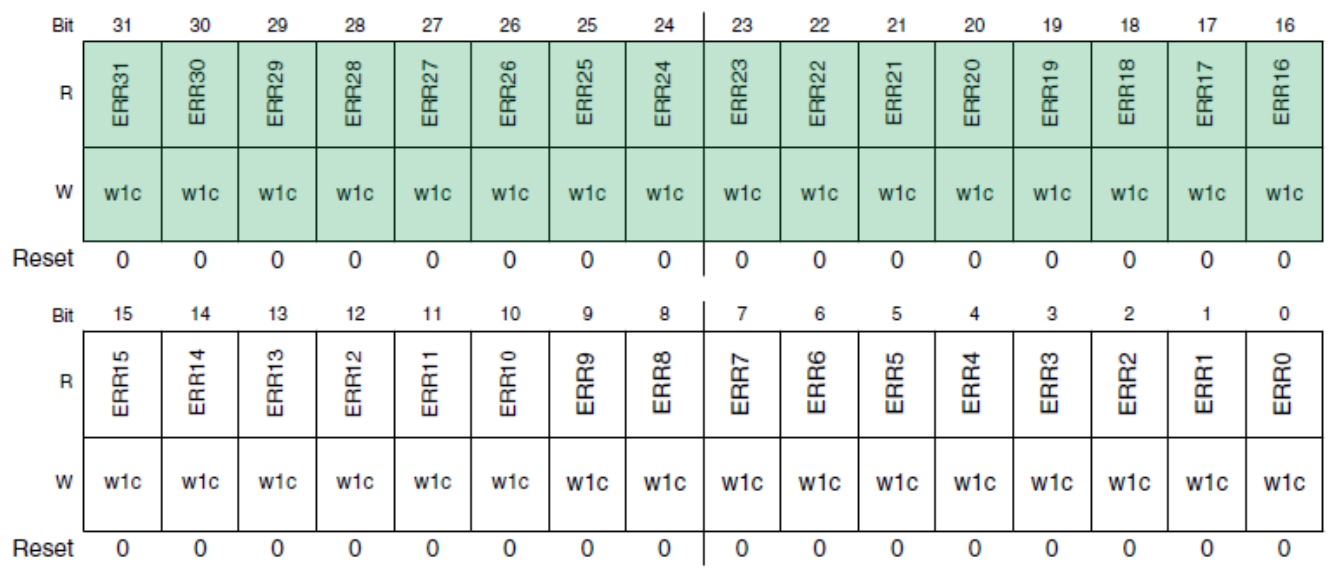
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	INT15	INT14	INT13	INT12	INT11	INT10	INT9	INT8	INT7	INT6	INT5	INT4	INT3	INT2	INT1	INT0
W	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 76. DMA_INT—Kinetis 120 MHz

New bits/fields added:

- INT31-INT16

4.2.2.4 DMA_ERR



Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
R	ERR31	ERR30	ERR29	ERR28	ERR27	ERR26	ERR25	ERR24	ERR23	ERR22	ERR21	ERR20	ERR19	ERR18	ERR17	ERR16
W	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	ERR15	ERR14	ERR13	ERR12	ERR11	ERR10	ERR9	ERR8	ERR7	ERR6	ERR5	ERR4	ERR3	ERR2	ERR1	ERR0
W	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 77. DMA_ERR—Kinetis 120 MHz

New bits/fields added:

- ERR31-ERR16

4.2.2.5 DMA_HRS

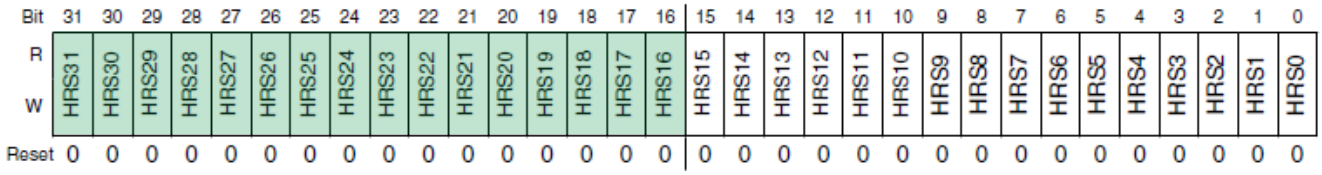


Figure 78. DMA_HRS—Kinetis 120 MHz

New bits/fields added:

- HRS31-HRS16

4.2.3 Software impact

Existing code will run on the eDMA with no changes. If you want to take advantage of the new channels, then software updates will be needed.

4.2.4 Hardware impact

No hardware impact.

4.3 Multi-Clock Generator (MCG)

4.3.1 Features

The MCG on the Kinetis 120 MHz products has a number of changes. Most importantly the PLL supports higher frequencies. There are also two PLLs included within the MCG instead of just one.

New features of the MCG include:

- Maximum PLL output frequency increases to 120MHz or 150MHz (max depends on part number speed grade)
- Second PLL has been added. Either PLL0 or PLL1 can provide the system level clocks (MCGOUTCLK).
- Only PLL1 can be used as the clock source for the DDR module.
- A second OSC has been added
  - Either of the OSCs can be the input to either of the PLLs
  - Only OSC0 or the RTC OSC can be the input to the FLL
  - Additional OSC control outputs have been added to the MCG for control of the second OSC (HGO1, RANGE1, EREFS1)
- PLL input reference range is now 8-16 MHz
- MCG registers can only be written when in privileged mode. Registers can be read in any mode.

4.3.2 Impacted registers

The MCG adds several new registers, and the MCG_ATC register has been renamed MCG_SC. In addition, many of the register field and bit names have been updated to match the naming convention used by the other modules.

The memory map changes are shown in the table below:

Table 11. MCG memory map comparison

Register	Location	Kinetis 100 MHz Rev. 1.x	Kinetis 120 MHz
MCG Status and Control Register	0x4006_4008	MCG_ATC	MCG_SC
MCG Control 7 Register	0x4006_400C	N/A	MCG_C7
MCG Control 8 Register	0x4006_400D	N/A	MCG_C8
MCG Control 10 Register	0x4006_400F	N/A	MCG_C10
MCG Control 11 Register	0x4006_4010	N/A	MCG_C11
MCG Control 12 Register	0x4006_4011	N/A	MCG_C12
MCG Status 2 Register	0x4006_4012	N/A	MCG_S2

4.3.2.1 MCG_C2 Register

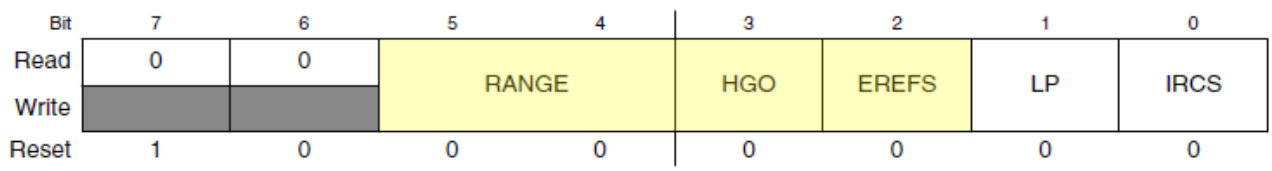


Figure 79. MCG_C2—Kinetis 100 MHz Rev. 1.x

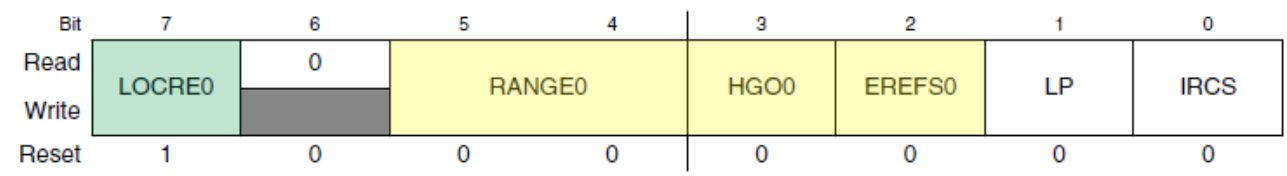


Figure 80. MCG_C2—Kinetis 120 MHz

New bits/fields added:

- LOCRE0

Changed bit/field names:

- RANGE → RANGE0
- HGO → HGO0
- EREFS → EREFS0

4.3.2.2 MCG_C5 Register

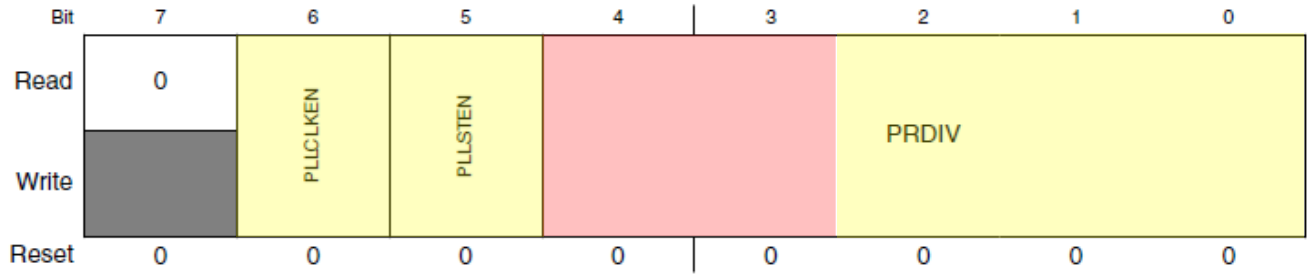


Figure 81. MCG_C5—Kinetis 100 MHz Rev. 1.x

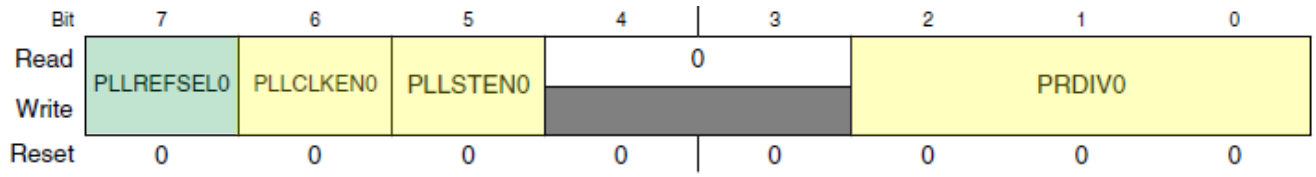


Figure 82. MCG_C5—Kinetis 120 MHz

New bits/fields added:

- PLLREFSEL0 - used to select between OSC0 and OSC1 as the input to PLL0

Changed bit/field names:

- PLLCLKEN → PLLCLKEN0
- PLLSTEN → PLLSTEN 0
- PRDIV → PRDIV 0 - the valid range is now 1-8 instead of 1-25

4.3.2.3 MCG_C6

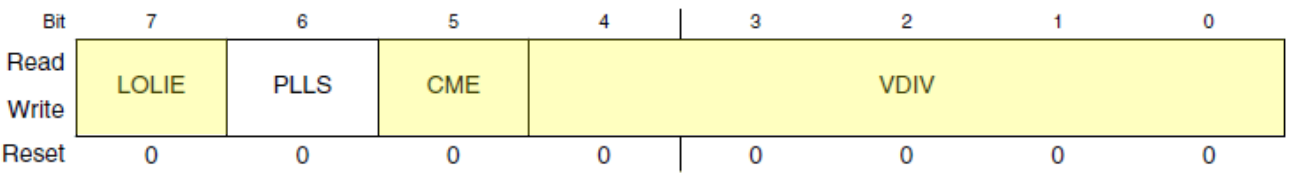


Figure 83. MCG_C6—Kinetis 100 MHz Rev. 1.x

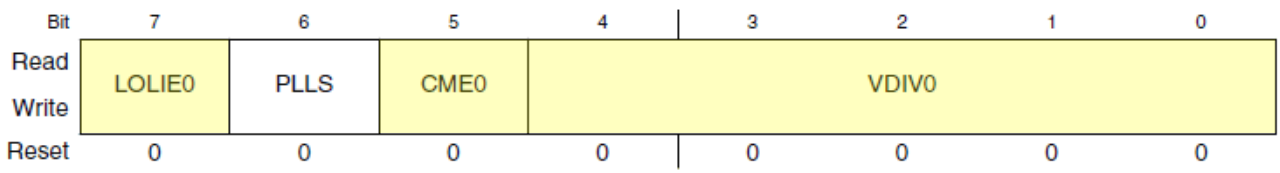


Figure 84. MCG_C6—Kinetis 120 MHz

Changed bit/field names:

- LOLIE → LOLIE0
- CME → CME0
- VDIV → VDIV0 - the range of valid VCO divides changes to 16-47 instead of 24-55

4.3.2.4 MCG_S Register

Figure 85. MCG_S—Kinetis 100 MHz Rev. 1.x

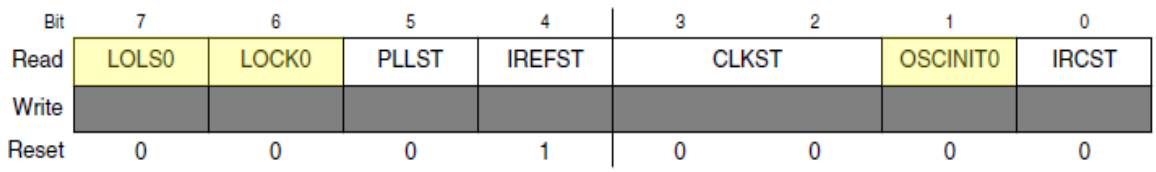


Figure 86. MCG_S—Kinetis 120 MHz

Changed bit/field names:

- LOLS → LOLS0
- LOCK → LOCK0
- OSCINIT → OSCINIT0

4.3.2.5 MCG_ATC Register

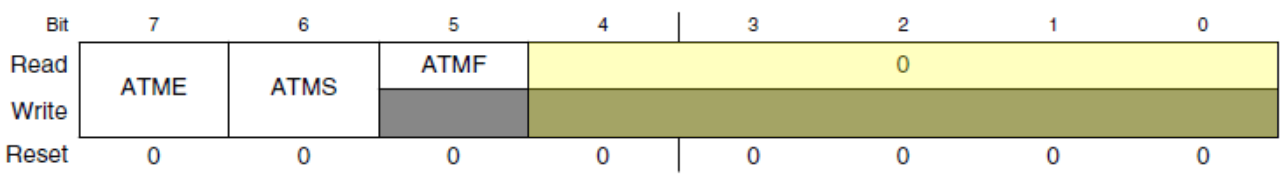


Figure 87. MCG_ATC—Kinetis 100 MHz Rev. 1.x

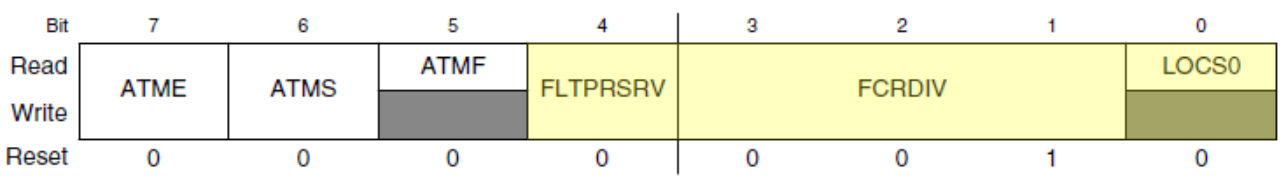


Figure 88. MCG_SC—Kinetis 120 MHz

Register name is changed to MCG_SC

New bits/fields added:

- FLTPRSRV
- FCRDIV
- LOCS0

4.3.2.6 MCG_C7 Register

Address: MCG_C7 is 4006_4000h base + Ch offset = 4006_400Ch

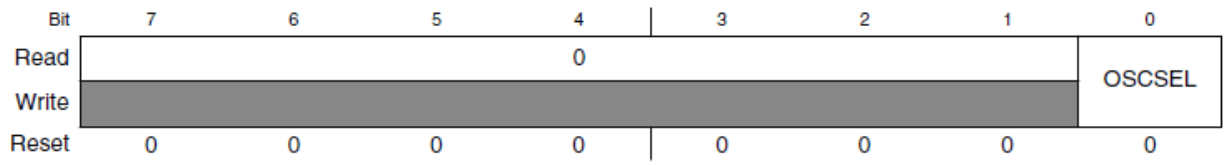


Figure 89. MCG_C7—Kinetis 120 MHz

New register

New bits/fields added:

- OSCSEL

4.3.2.7 MCG_C8 Register

MCG_C8 is a new register with control and status bits for the new RTC clock monitor. This is a new feature that has been added.

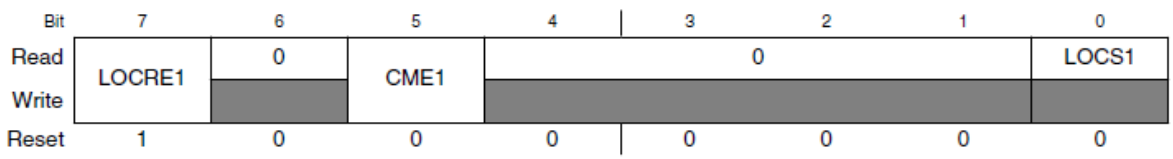


Figure 90. MCG_C8—Kinetis 120 MHz

New bits/fields added:

- LOLCRE1
- CME1
- LOCS1

4.3.2.8 MCG_C10 Register

The MCG_C10 is a new register used for control of OSC1. Most of the register fields are similar to the MCG_C2 register.

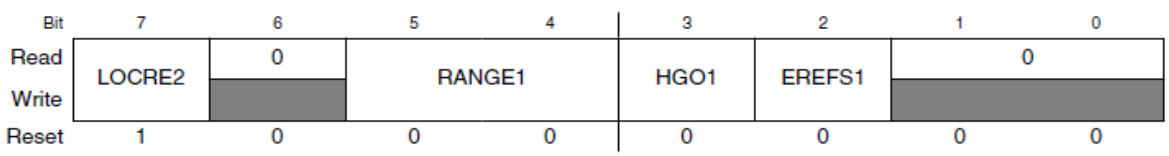


Figure 91. MCG_C10—Kinetis 120 MHz

New bits/fields added:

- LOCRE2
- RANGE1
- HGO1
- EREFS1

4.3.2.9 MCG_C11 Register

The MCG_C11 is a new register used for control of PLL1. Most of the register fields are similar to the MCG_C5 register that is used to control PLL0.

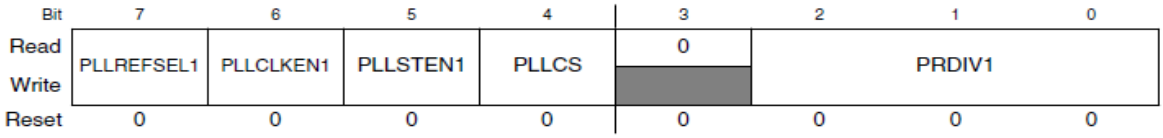


Figure 92. MCG_C11—Kinetis 120 MHz

New bits/fields added:

- PLLREFSEL1
- PLLCLKEN1
- PLLSTEN1
- PLLCS
- PRDIV1

4.3.2.10 MCG_C12 Register

The MCG_C12 is a new register used for control of PLL1. Most of the register fields are similar to the MCG_C6 register that is used to control PLL0.

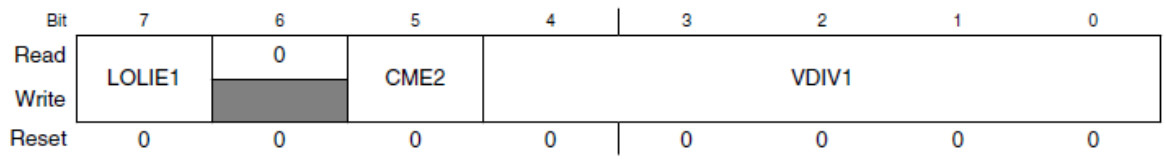


Figure 93. MCG_C12—Kinetis 120 MHz

New bits/fields added:

- LOLIE1
- CME2
- VDIV1

4.3.2.11 MCG_S2 Register

The MCG_S2 is a new register used for status of PLL1 and OSC1. Most of the register fields are similar to the MCG_S register that is used to for status of PLL0 and OSC0.

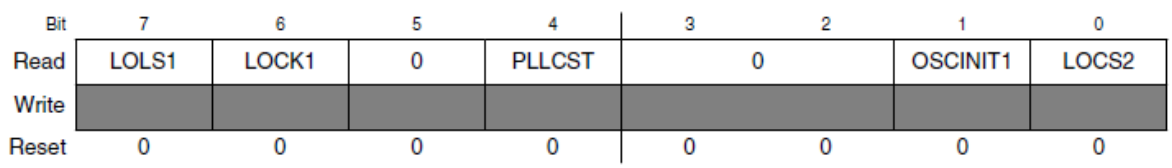


Figure 94. MCG_S2—Kinetis 120 MHz

New bits/fields added:

- LOLS1
- LOCK1
- PLLCST

- OSCINIT1
- LOCS2

4.3.3 Software impact

Any system that is using the PLL will require software changes associated with the updates to the MCG to adjust for the new PLL input range, divider ranges, and increased output frequency. Software will also need to take into account the name changes associated with the registers and bit fields. The software will need to be changed to match the new names included in the new device header file.

If the RTC is being used as the reference clock for the FLL, or as the system clock, the selection of this clock is no longer made in the SIM_SOPT1; it is now in the MCG_C7 register.

Software will need to be added to make use of any new functionality. If the system makes use of the processor user mode, any MCG register configuration will need to be performed in privileged mode.

4.3.4 Hardware impact

Because the PLL input reference range is now 8-16 MHz, systems that were using input clocks lower than 8 MHz will need to change the input clock frequency if either of the PLLs are going to be used.

The change associated with the RTC oscillator selection as the DFLL reference clock has added the ability to also use OCS0 with a high frequency crystal to provide a separate clock to several on-chip peripherals (OSC0ERCLK).

A second oscillator has also been added (OSC1). OSC1 can be used as the reference clock source for one or both of the PLLs, but there are some restrictions on its use. The following items should be taken into account if you are considering using OSC1 as the main clock input on your board:

- Either OSC0 or OSC1 can be used as the reference clock source for the PLL being used as MCGCLKOUT.
- Only OSC0 or the RTC OSC can be used as the FLL reference clock so one of those clock sources MUST be available to the MCG if using FEE, FBE, or BLPE clock modes or when transitioning from the reset default FEI mode to PBE mode (or any “external” clocked MCG mode). You must transition through FBE mode to enter PEE mode and use the PLL as the MCGCLKOUT source.
- If OSC1 is to be used it must be configured in MCG_C10 and enabled in OSC1_CR.
- If OSC1 is the only external clock source then the only MCG clock modes available are FEI, FBI and BLPI.

4.4 Real Time Clock (RTC)

4.4.1 Features

The following new features have been included:

- RTC_CLKOUT signal now has the option to be either 1 Hz or 32 kHz in all packages

  Use case/Improvement: The originally available option of 1 Hz allows the ability to accurately generate a 1 Hz clock even if the input crystal frequency deviates from the 32.768 kHz due to on-board variations. The 32.768 kHz option allows the input crystal frequency to be passed straight through and out to RTC_CLKOUT. This allows for other external devices, which require this clock, to use it as an input while leveraging the RTC oscillator and reducing the system BOM cost.

- RTC_WAKEUP pin is now implemented to provide an external alarm event signal in some MAPBGA packages

Use case/Improvement: This is an active low open drain pin that asserts when the MCU is powered down and the main RTC interrupt asserts. Note that this pin can be used to measure the RTC trigger output that monitors the MCU VDD level.

- RTC seconds interrupt has been added

  Use case/Improvement: This is a dedicated interrupt that asserts every second. Note that it is an edge-triggered interrupt, so there is no flag to clear. This means the interrupt can be serviced without accessing the RTC registers, which decreases VBAT power consumption. The seconds interrupt can generate a wakeup from any low-power mode (Stop/VLPS/LLS/VLLS). For example, if the RTC Seconds signal is connected to the PDB, the PDB trigger can receive the RTC Seconds trigger input forcing ADC conversions in Run mode (where PDB is enabled).

In addition to the changes listed above the 120 MHz devices implement a secure version of the RTC with additional features specifically targetted at creating secure systems. These features are:

- 64-bit monotonic counter that cannot be exhausted or return to any previous value, once it has been initialized.
- Tamper time seconds register that records when the time was invalidated.

4.4.2 Impacted registers

The RTC module has several new registers. There are also new fields added within some of the existing registers. All of the existing registers and fields remain the same.

The table below lists the new registers that have been added. The following sections provide details on register fields that have been added to existing registers.

Table 12. RTC memory map comparison

Register	Location	Kinetis 100 MHz Rev 1.x	Kinetis 120 MHz
Tamper Time Seconds Register	0x4003_D020	N/A	RTC_TTSR
Monotonic Enable Register	0x4003_D024	N/A	RTC_MER
Monotonic Counter Low Register	0x4003_D028	N/A	RTC_MCLR
Monotonic Counter High Register	0x4003_D02C	N/A	RTC_MCHR

4.4.2.1 RTC_SR Register

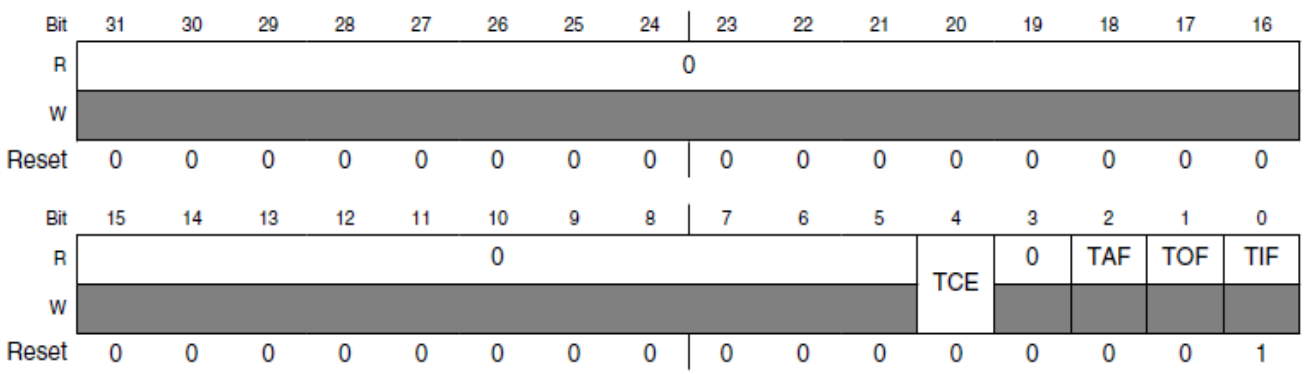


Figure 95. RTC_SR—Kinetis 100 MHz Rev. 1.x

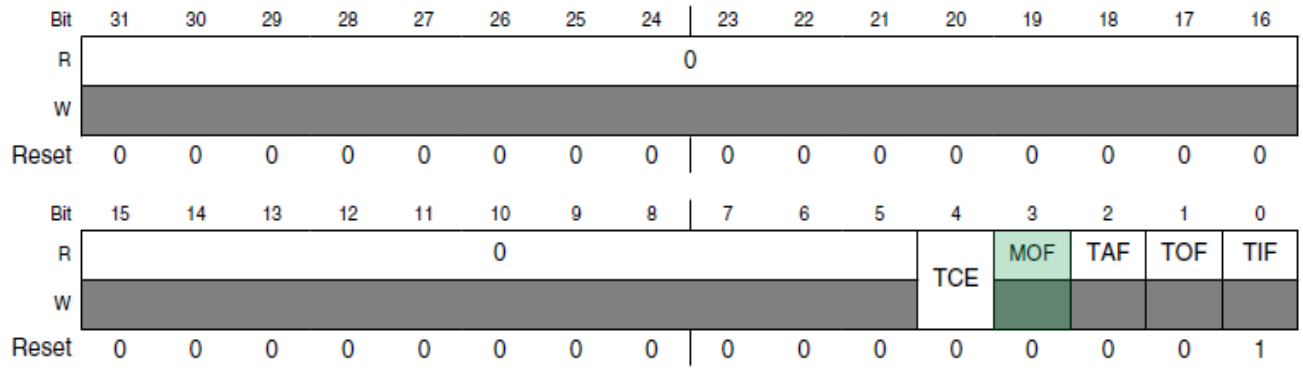


Figure 96. RTC_SR—Kinetis 120 MHz

New bits/fields added:

- MOF - monotonic overflow flag

4.4.2.2 RTC_LR Register

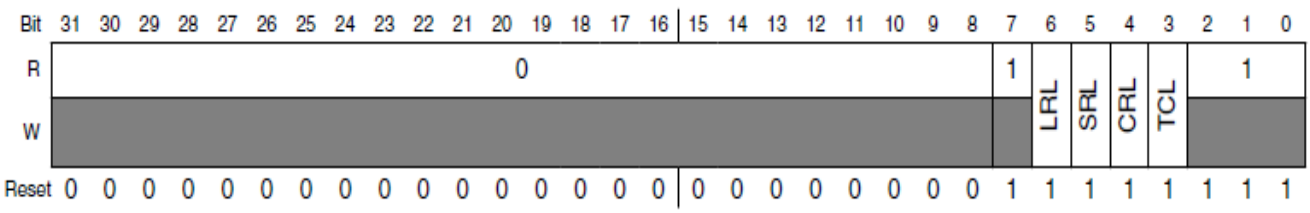


Figure 97. RTC_LR—Kinetis 100 MHz Rev. 1.x

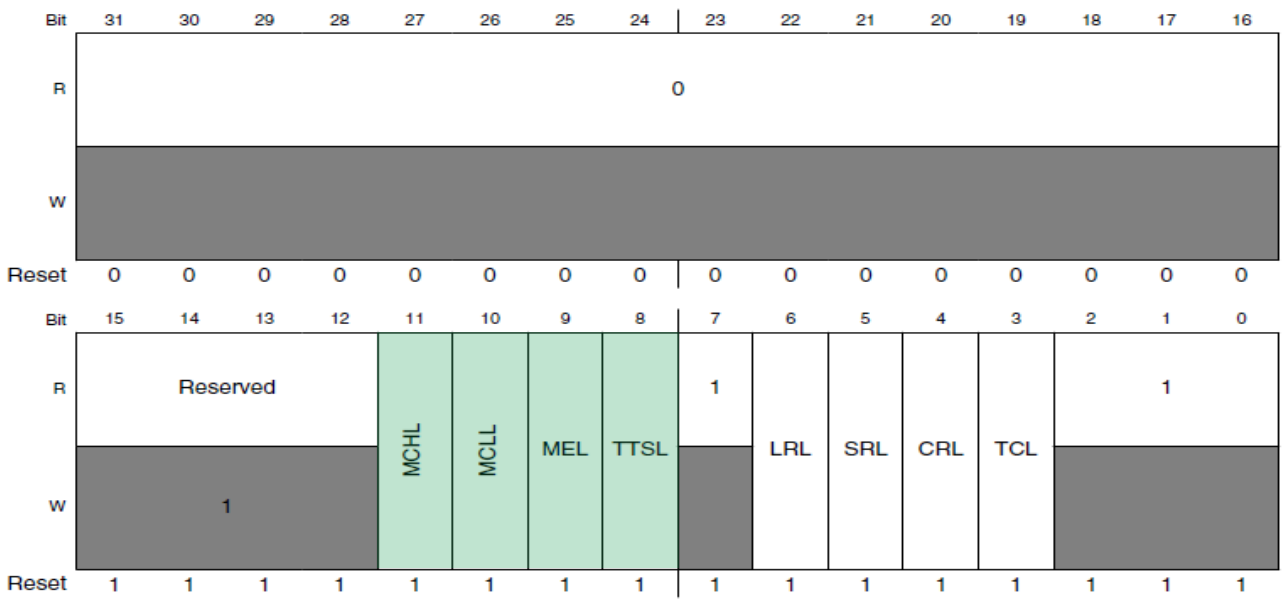


Figure 98. RTC_LR—Kinetis 120 MHz

New bits/fields added:

- MCHL - monotonic counter high lock
- MCLL - monotonic counter low lock

- MEL - monotonic enable lock
- TTSL - tamper time seconds lock

4.4.2.3 RTC_IER Register

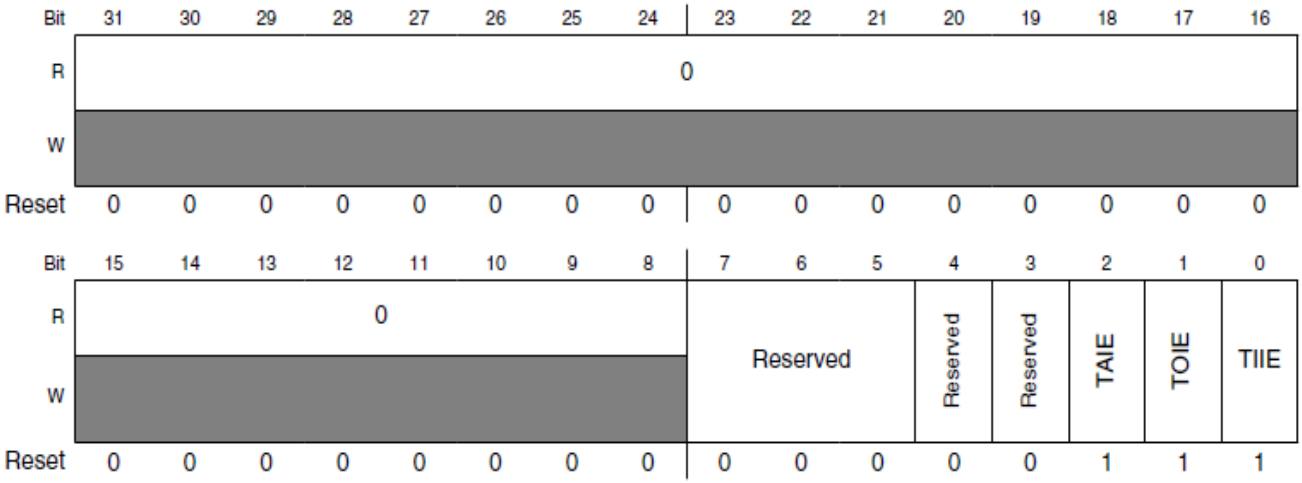


Figure 99. RTC_IER—Kinetis 100 MHz Rev. 1.x

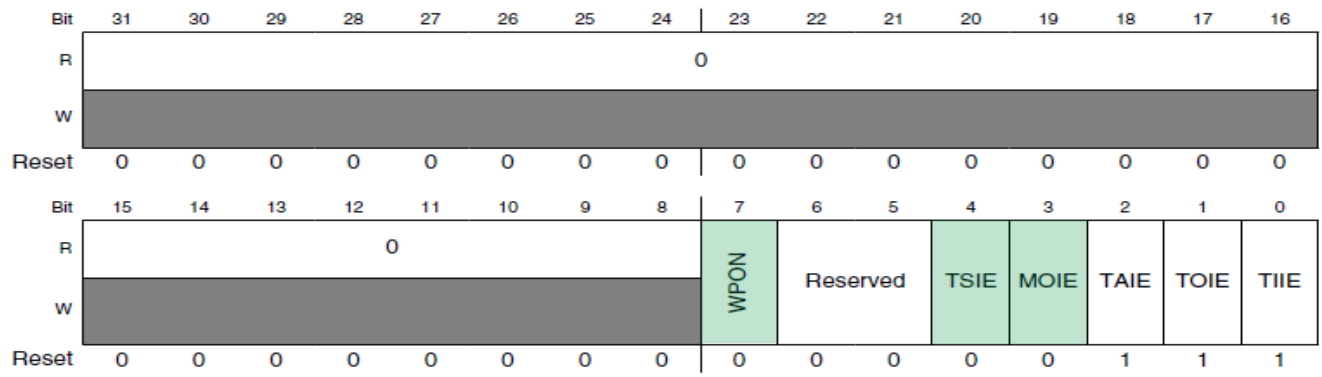


Figure 100. RTC_IER—Kinetis 120 MHz

New bits/fields added:

- WPON - enables assertion of the RTC_WAKEUP pin
- TSIE - time seconds interrupt enable
- MOIE - monotonic counter overflow interrupt enable

4.4.2.4 RTC_WAR Register

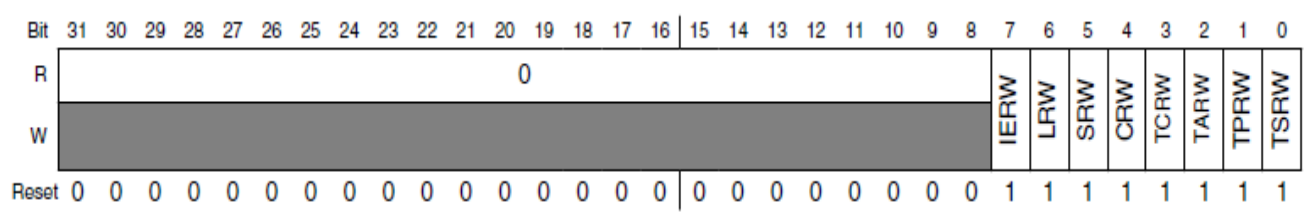


Figure 101. RTC_WAR—Kinetis 100 MHz Rev. 1.x

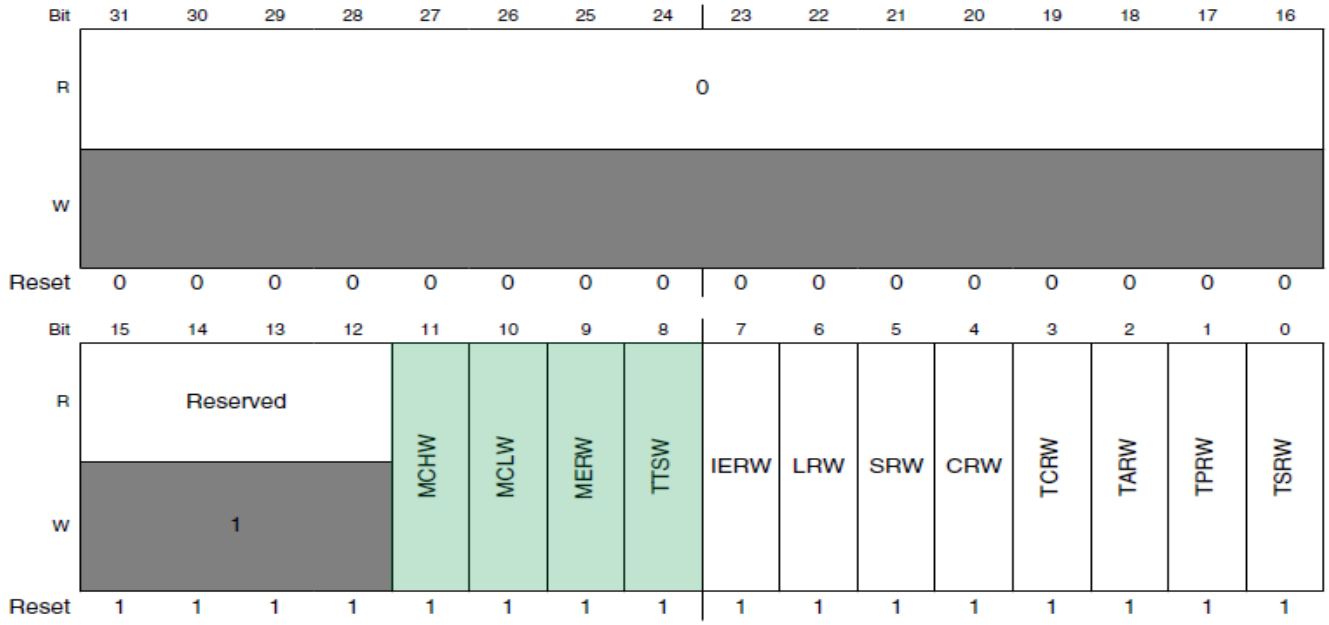


Figure 102. RTC_WAR—Kinetis 120 MHz

New bits/fields added:

- MCHW - monotonic counter high write
- MCLW - monotonic counter low write
- MERW - monotonic enable register write
- TTSW - tamper time seconds write

4.4.2.5 RTC_RAR Register

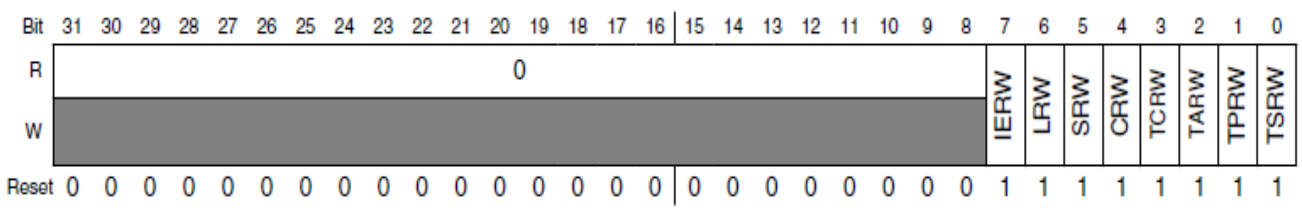


Figure 103. RTC_RAR—Kinetis 100 MHz Rev. 1.x

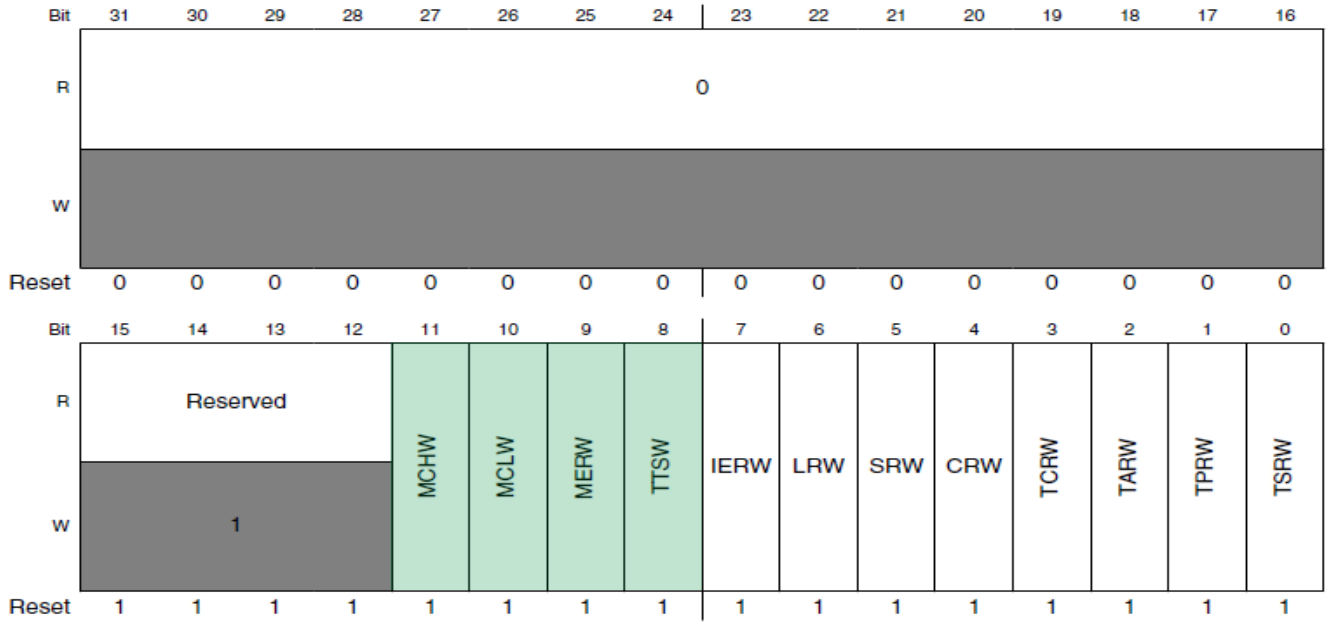


Figure 104. RTC_RAR—Kinetis 120 MHz

New bits/fields added:

- MCHR - monotonic counter high read
- MCLR - monotonic counter low read
- MERR - monotonic enable register read
- TTSR - tamper time seconds read

4.4.3 Software impact

No software changes are required in order to execute existing code. To take advantage of additional features, some software changes would apply.

4.4.4 Hardware impact

No hardware changes are required. To take advantage of additional features, some hardware changes would apply.

4.5 Programmable Delay Block (PDB)

4.5.1 Features

The PDB has several new features that have been added. New features include:

- Three additional Pulse-Out n Delay registers (PDB0_PO1DLY, PDB0_PO2DLY, and PDB0_PO3DLY) are added. These registers are used to set sample window for CMP1, CMP2, and CMP3 modules.
- Two new sets of ADC channel trigger configuration registers are added (PDB0_CHnC1, PDB0_CHnS, PDB0_CHnDLY0, PDB0_CHnDLY1). The two new channels can be used as triggers for ADC2 and ADC3.
- PDB Counter Register(PDBx_CNT) value will pause when the processor is in debug mode.

4.5.2 Impacted registers

The PDB adds several new registers. The registers are all additional instantiations of existing registers. The field definitions within the registers are the same there are just more copies of the registers to allow for control of more channels.

The new registers locations are shown in the table below:

Table 13. PDB memory map comparison

Register	Location	Kinetis 100 MHz Rev. 1.x	Kinetis 120 MHz
Channel n Control Register 1	0x4003_6060	N/A	PDB0_CH2C1
Channel n Status Register	0x4003_6064	N/A	PDB0_CH2S
Channel n Delay 0 Register	0x4003_6068	N/A	PDB0_CH2DLY0
Channel n Delay 1 Register	0x4003_606C	N/A	PDB0_CH2DLY1
Channel n Control Register 1	0x4003_6088	N/A	PDB0_CH3C1
Channel n Status Register	0x4003_608C	N/A	PDB0_CH3S
Channel n Delay 0 Register	0x4003_6090	N/A	PDB0_CH3DLY0
Channel n Delay 1 Register	0x4003_6094	N/A	PDB0_CH3DLY1
Pulse-Out n Delay Register	0x4003_6198	N/A	PDB0_PO1DLY
Pulse-Out n Delay Register	0x4003_619C	N/A	PDB0_PO2DLY
Pulse-Out n Delay Register	0x4003_61A0	N/A	PDB0_PO3DLY

4.5.3 Software impact

The additional PDB0_PO1DLY and PDB0_PO2DLY registers have been added so that you can have different window periods for CMP0, CMP1, and CMP2 outputs. If CMP0, CMP1, and CMP2 are configured for window mode, then software changes will be needed to configure the appropriate PDB0_PO*n*DLY register to configure the window instead of using the PDB0_PO1DLY register for all of the comparators.

Software changes are also required if the new PDB channels will be used to trigger ADC2 and ADC3.

4.5.4 Hardware impact

No hardware impact.

4.6 Flexible Timer Module (FTM)

4.6.1 Features

The Kinetis 120 MHz devices add a fourth FTM instantiation with 8 channels (FTM3).

The FTM module adds a new hardware synchronization trigger feature that allows FTM0 and/or FTM3 to be triggered by a match on FTM1 or FTM2. This new features makes it possible to connect to 16-bit FlexTimers, thus allowing you to increase PWM period with less dead time than using the timer overflow flag (TOF) to connect both modules. These hardware triggers are configured using new bits in the SIM_SOPT4 register. Refer to the SIM section for more information.

There is also an option to use the new USB start of frame pulse as an input capture trigger for FTM1 channel 0. This bit can be used to synchronize the MCU clock with the USB clock which is useful for some USB audio applications. Synchronization with the USB start of frame is controlled through the SIM_SOPT4[FTM1CH0SRC] bit. Refer to the SIM and USB sections for more information.

4.6.2 Impacted registers

There is a new set of FTM registers added for the new instantiation (FTM3). The new FTM features are all handled with changes to SIM registers; the FTM registers and fields are unchanged.

4.6.3 Software impact

No software changes are required to execute existing code. To take advantage of additional features or the new FTM3 module, some software changes would apply.

4.6.4 Hardware impact

No hardware impact.

4.7 Low-power timer (LPTMR)

4.7.1 Features

The LPTMR has been updated to always ensure valid data when reading the LPTMR_CNR value.

4.7.2 Impacted register

LPTMR_CNR

4.7.3 Software impact

On each read of the LPTMR counter register, software must first write to the LPTMR counter register with any value. This will synchronize and register the current value of the LPTMR counter register into a temporary register. The contents of the temporary register are returned on each read of the LPTMR counter register.

4.7.4 Hardware impact

No hardware impact.

4.8 Universal asynchronous receiver/transmitters (UART)

4.8.1 Features

The new version of the UART adds support for the CEA709.1-B (LON) protocol on UART0. This protocol is commonly used in building automation and home networking applications. The features across the UART instantiations have not changed except for the addition of CEA709.1-B support.

4.8.2 Impacted Registers

There are a number of new registers added to the UART0 memory map to support new CEA709.1-B (LON) features. The new features and control of those features are all implemented as additions to the memory map, so previously existing registers and bits are unchanged.

The table below shows the memory map differences between the original UART and the enhanced version that includes LON capability.

Table 14. Memory map comparison

Location	Original UART	Enhanced UART
0x4006A021	N/A	UART CEA709.1-B Control Register 6 (UART0_C6)
0x4006A022	N/A	UART CEA709.1-B Packet Cycle Time Counter High (UART0_PCTH)
0x4006A023	N/A	UART CEA709.1-B Packet Cycle Time Counter Low (UART0_PCTL)
0x4006A024	N/A	UART CEA709.1-B Beta1 Timer (UART0_B1T)
0x4006A025	N/A	UART CEA709.1-B Secondary Delay Timer High (UART0_SDTH)
0x4006A026	N/A	UART CEA709.1-B Secondary Delay Timer Low (UART0_SDTL)
0x4006A027	N/A	UART CEA709.1-B Preamble (UART0_PRE)
0x4006A028	N/A	UART CEA709.1-B Transmit Packet Length (UART0_TPL)
0x4006A029	N/A	UART CEA709.1-B Interrupt Enable Register (UART0_IE)
0x4006A02A	N/A	UART CEA709.1-B WBASE (UART0_WB)
0x4006A02B	N/A	UART CEA709.1-B Status Register (UART0_S3)
0x4006A02C	N/A	UART CEA709.1-B Status Register (UART0_S4)
0x4006A02D	N/A	UART CEA709.1-B Received Packet Length (UART0_RPL)

Table continues on the next page...

Table 14. Memory map comparison (continued)

Location	Original UART	Enhanced UART
0x4006A02E	N/A	UART CEA709.1-B Received Preamble Length (UART0_RPREL)
0x4006A02F	N/A	UART CEA709.1-B Collision Pulse Width (UART0_CPW)
0x4006A030	N/A	UART CEA709.1-B Receive Indeterminate Time (UART0_RIDT)
0x4006A031	N/A	UART CEA709.1-B Transmit Indeterminate Time (UART0_TIDT)

4.8.3 Software impact

If your system does not need to take advantage of the new CEA709.1-B features, no software changes are required.

4.8.4 Hardware impact

If your system does not need to take advantage of the new CEA709.1-B features, no hardware changes are required.

4.9 Universal Serial Bus (USB)

4.9.1 Features

The following new features have been included:

- USB regulator Standby mode protection mechanism
- Start-of-frame (SOF) output on USB_SOF_OUT pin
- USB peripheral adds adjustable frame register in Host mode

4.9.1.1 USB regulator Standby mode

One of the features of the USB voltage regulator output Vout33 is to serve as the MCU's main power.

When Standby mode is enabled, the regulator limits maximum current load to 1 mA. The Kinetis 100 MHz Rev. 1.x devices permit the configuration of the USB VREG into Standby mode while the MCU is in Run mode, which means that there won't be enough current for the MCU to execute instructions.

Newer Kinetis devices implement a protection mechanism that controls when the USB voltage regulator is placed in Standby mode.

4.9.1.2 Start-of-frame output pin

A new signal has been added that can reflect the value of the SOF pin that can be used like a startup event for synchronization purposes (for audio, data loggers, and so on).

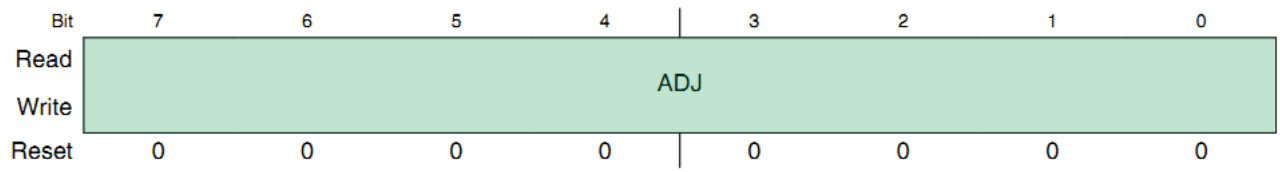
4.9.1.3 Adjustable frames in Host mode

SOF is normally generated every 12,000 12 MHz clock cycles. This new feature can adjust this by –128 to +127 to compensate for inaccuracies in the USB 48 MHz clock.

4.9.2 Impacted register

Functionality of the USB voltage regulator can be controlled via the SIM_SOPT1CFG Register, a new register, and the SIM_SOPT1 Register, where bits and bit fields have been updated in order to support the new features. Refer to the SIM section more more details.

The USB has one new register: USBx_USBFRMADJUST.



USBx_USBFRMADJUST field descriptions

Figure 105. USBx USBFRMADJUST

4.9.3 Software impact

4.9.3.1 USB regulator Standby mode

Two new bits have been added into the SIM_SOPT1 register to control the USB VREG Standby mode.

USBSSTBY—USB voltage regulator in Standby mode during Stop, VLPS, LLS, and VLLS modes.

USBVSTBY—USB voltage regulator in Standby mode during VLPR and VLPW modes.

The new SIM_SOPT1CFG register needs to allow writes prior to enabling the USB voltage regulator settings in SIM_SOPT1 (USBSSTB, USBVSTB, USBREGEN bits).

4.9.3.2 Adjustable frames in Host mode

Write twos complement number to the new USBx_USBFRMADJUST register to adjust the period of USB frame.

4.9.4 Hardware impact

No hardware changes required to be compatible with previous versions.

4.9.4.1 Start-of-frame output pin

The Pin Multiplexing section of this document provides more information about this pin.

4.10 Comparator

4.10.1 Features added and changed

Change 1

A 12-bit DAC1 output has been added as part of the CMP0 input channel 4 (IN4) input.

The table below shows that in Kinetis 120 MHz silicon, CMP0 input channel 4 is now connected to both the CMP0_IN4 external pin and the 12-bit DAC1 output. When CMP0 channel 4 is selected as an input, only one of the two signals should be enabled to avoid signal conflicts.

CMP0 inputs	Kinetis 100 MHz Rev.1.x	Kinetis 120 MHz
IN4	CMP0_IN4	12b DAC1 reference/CMP0_IN4

Change 2

The CMP0_IN4 input has been relocated to a different MCU pin.

The Pin Multiplexing section of this document shows that the CMP0_IN4 input has been moved to the new location on pin L4 for K60 144 MAPBGA Kinetis devices or pin 39 for K60 LQFP Kinetis devices. This new location may be different among different Kinetis family devices and package types. For the exact new pin location of the CMP0_IN4 input, please refer to the corresponding Kinetis reference manual and look for the Signal Multiplexing and Pin Assignments table.

Change 3

In addition to the functional changes described above the Kinetis 120 MHz devices add an extra instantiation of the comparator. The 120 MHz devices have up to four comparators where the 100 MHz devices have up to three.

4.10.2 Impacted registers

Several register fields have been removed from the CMP registers.

4.10.2.1 CMPx_SCR

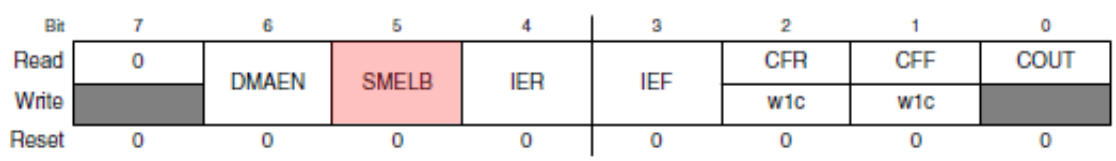


Figure 106. CMPx_SCR—Kinetis 100 MHz Rev. 1.x

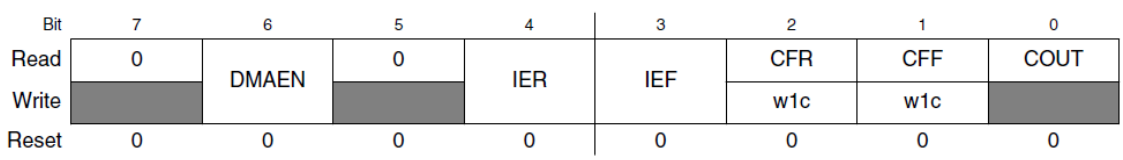


Figure 107. CMPx_SCR—Kinetis 120 MHz

Removed bit/field names:

- SMELB

4.10.2.2 CMPx_MUXCR

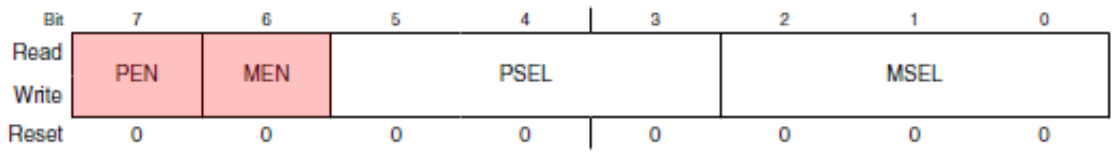


Figure 108. CMPx_MUXCR—Kinetis 100 MHz Rev. 1.x

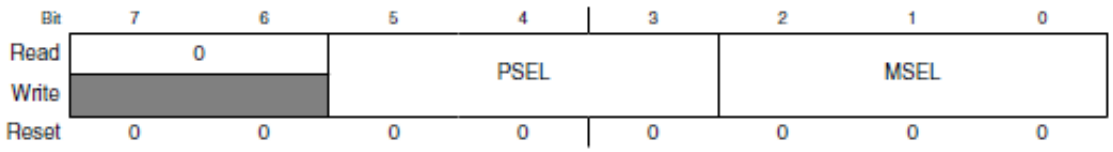


Figure 109. CMPx_MUXCR—Kinetis 120 MHz

Removed bit/field names:

- PEN
- MEN

4.10.3 Software impact

Change 1

The CMP0 IN4 input channel is now multiplexed with both the 12-bit DAC1 reference and CMP0_IN4 pin. Only one of the two signals should be enabled when CMP0 IN4 is selected. For example, if the CMP0 IN4 is intended for CMP0_IN4 pin, make sure you disable the 12-bit DAC1 output by DAC1_C0[DACEN] = 0. If CMP0 IN4 is intended for the 12-bit DAC1 output, the external source that drives the CMP0_IN4 pin should be disconnected.

Change 2

In newer Kinetis devices, PORTC_PCR_10[MUX] = 0 no longer selects CMP0_IN4 as an input. The new CMP0_IN4 input has been relocated to an analog-dedicated pin where all the multiplexed functions on this pin are analog functions and not digital GPIO. Such an analog-only pin does not require that you specify the pin function via a Pin Control Register. However, it is recommended that only one of the multiplexed analog functions should be enabled at the same pin.

4.10.4 Hardware impact

If the CMP0_IN4 signal is being used, then hardware changes are required to reroute the signal to the new pin location for the CMP0_IN4 input. When the CMP0_IN4 pin is used, the corresponding multiplexed analog functions on the same pin should not be used and should be disabled.

4.11 Periodic interrupt timers (PIT)

4.11.1 Features

The following new feature has been included:

- The ability for the peripheral to concatenate timer channels via the Chain Mode bit, PIT_TCTRLn[CHN].

  Use case/Improvement: This provides a means of chaining a timer to a previous timer. For example, if this bit is set for channel 2, then timer 2 is chained to timer 1, thus creating a 64-bit counter.

4.11.2 Impacted registers

Although the overall PIT memory map has not changed, it does have a new register bit field added to the Timer Control Register.

The affected registers are summarized below.

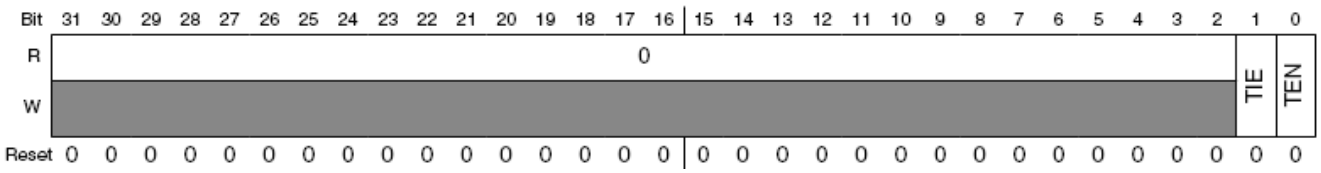


Figure 110. PIT_TCTRLn Register - Orignal

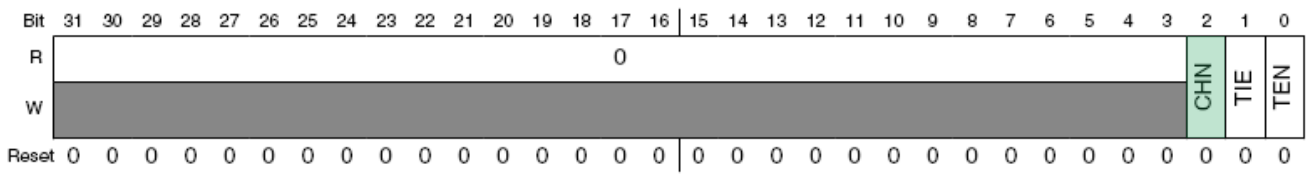


Figure 111. PIT_TCTRLn Register - Enhanced

New bits/fields added: CHN

4.11.3 Software impact

No software changes are required to execute existing code on the new revision of the PIT. To take advantage of additional features, some software changes would apply.

4.11.4 Hardware impact

No hardware impact.

4.12 Ethernet MAC (ENET)

4.12.1 Features

The following new features have been included:

1. Compliant with AMD magic packet detection

   Use case/Improvement: The previous ENET version decodes a magic packet if the frame is formed with a synchronization stream (six consecutive 0xFF bytes) followed by sequence of six consecutive unicast MAC addresses of the node to be awakened. These 42 bytes must not have anything between them and might be placed anywhere in the Ethernet frame. Then it can be mounted over Ethernet, UDP, IP, TCP, or any other protocol.

   The new version is compliant with AMD magic packet detection, which requires a frame formed with a synchronization stream (six consecutive 0xFF bytes), followed by sequence of sixteen consecutive unicast MAC addresses of the node to be awakened. These 102 bytes might be placed anywhere in the Ethernet frame.

2. Addition of VLAN Priority bits to the enhanced receive buffer descriptor

   Use case/Improvement: When using enhanced buffer descriptors, the RxBD[VPCP] (VLAN priority code point) detects the VLAN frame priority level in three bits (only valid when the RxBD[L] bit is set). This value can be used to prioritize different classes of traffic (for example, voice, video, or data) where a value of 7 is the highest priority VLAN frame.

3. Addition of buffer descriptor endianess option

   Use case/Improvement: When writing the buffer descriptors fields, regardless of the enhanced or legacy mode, an endianess software conversion from big endian (register endianess) to little endian is required in previous and new MAC-NET versions. When reading the buffer descriptor fields, a little endian to big endian conversion is also required.

   However, ENET_ECR[DBSWP] allows managing buffer descriptors in ARM Cortex-M4 native little endian, performing the endianess conversion in hardware.

4.12.2 Impacted registers

The ENET adds two new register fields, ENET_ECR[DBSWP] and enhanced RxBD[VPCP], and changed functionality of ENET_ECR[MAGICEN].

There are no memory map changes but mentioned registers are included for reference as shown in the table below.

Memory map comparison				
	Orignal ENET		Enhanced ENET	
	Location	Name	Location	Name
Ethernet Control Register	400C_0024	ENET_ECR	400C_0024	ENET_ECR
Enhanced Rx Buffer Descriptor	N/A	Enhanced RxBD	N/A	Enhanced RxBD

4.12.2.1 ENET_ECR Register

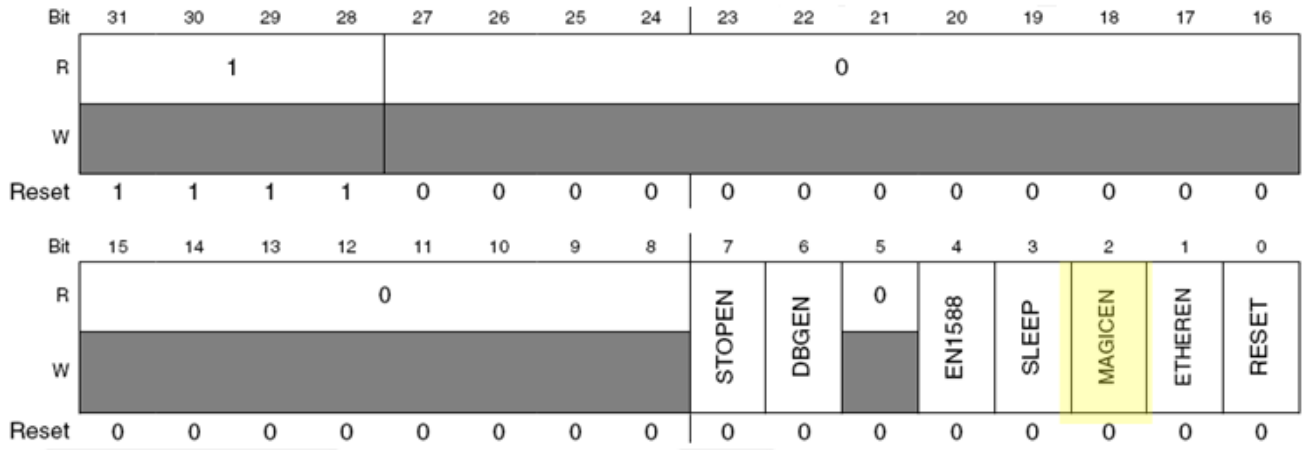


Figure 112. ENET_ECR - Original

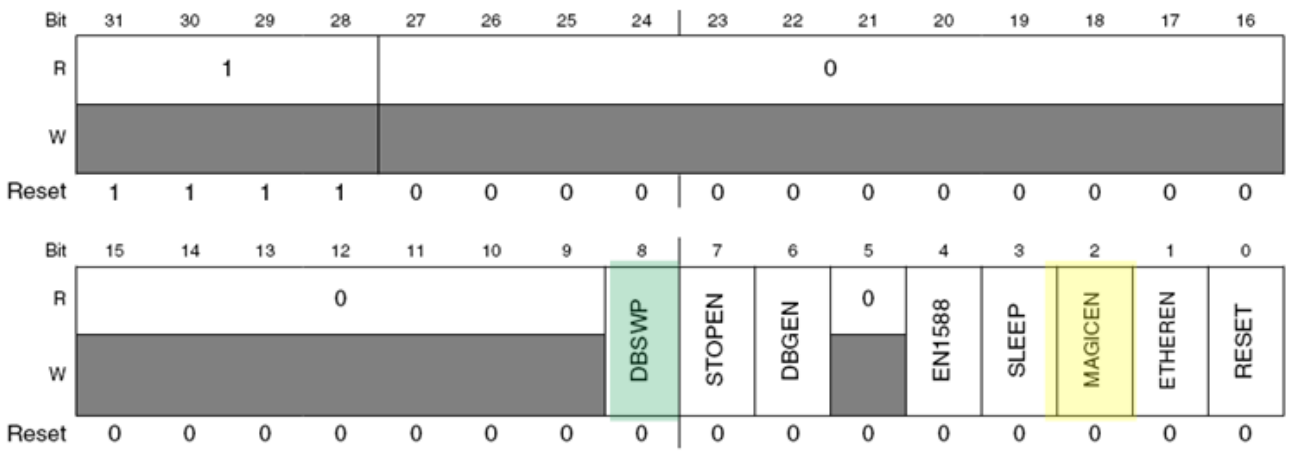


Figure 113. ENET_ECR - Enhanced

New bits/fields added: DBSWP

New bits/fields changed: MAGICEN

4.12.2.2 Enhanced Rx Buffer Descriptor (RxBD)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Offset + 0	E	RO1	W	RO2	L	—	—	M	BC	MC	LG	NO	—	CR	OV	TR
Offset + 2	Data length															
Offset + 4	Rx data buffer pointer - A[31:16]															
Offset + 6	Rx data buffer pointer - A[15:0]															
Offset + 8	ME	—	—	—	—	PE	CE	UC	INT	—	—	—	—	—	—	—
Offset + A	—	—	—	—	—	—	—	—	—	—	ICE	PCR	—	VLAN	IPV6	FRAG
Offset + C	Header length					—	—	—	Protocol type							
Offset + E	Payload checksum															
Offset + 10	BDU	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Offset + 12	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Offset + 14	1588 timestamp [31:16]															
Offset + 16	1588 timestamp [15:0]															
Offset + 18	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Offset + 1A	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Offset + 1C	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Offset + 1E	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Figure 114. RxBD - original

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Offset + 0	E	RO1	W	RO2	L	—	—	M	BC	MC	LG	NO	—	CR	OV	TR
Offset + 2	Data length															
Offset + 4	Rx data buffer pointer - A[31:16]															
Offset + 6	Rx data buffer pointer - A[15:0]															
Offset + 8	ME	—	—	—	—	PE	CE	UC	INT	—	—	—	—	—	—	—
Offset + A	VPCP			—	—	—	—	—	—	—	ICE	PCR	—	VLAN	IPV6	FRAG
Offset + C	Header length					—	—	—	Protocol type							
Offset + E	Payload checksum															
Offset + 10	BDU	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Offset + 12	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Offset + 14	1588 timestamp [31:16]															
Offset + 16	1588 timestamp [15:0]															
Offset + 18	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Offset + 1A	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Offset + 1C	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Offset + 1E	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Figure 115. RxBD - Enhanced

New bits/fields added: VPCP

4.12.3 Software impact

The default configuration/operation of the ENET has not been changed. All the new features make the ENET compliant with a standard or offload work to the application, giving it to the hardware. Two of the three features are backwards-compatible with previous implementations and don't require any upgrade in software. The addition of buffer descriptor (BD) endianess is turned off by default in the enhanced ENET and the corresponding bit is reserved the original ENET. A change is only required if it is explicitly enabled.

- Compliant with AMD magic packet detection

  The new version is compliant with AMD magic packet detection. It requires a frame formed with a synchronization stream (six consecutive 0xFF bytes) followed by sequence of sixteen consecutive unicast MAC addresses of the node to be awakened. These 102 bytes might be placed anywhere in the Ethernet frame. Six consecutive unicast MAC addresses no longer wake up the MCU.

- Addition of VLAN Priority bits to the enhanced receive buffer descriptor

  The application no longer needs to parse the VLAN packet to get the frame priority. It can be taken from the enhanced RxBD[VPCP] field. Note that enhanced buffer descriptor mode must be enabled.

- Addition of buffer descriptor endianess option
  Before implementing the buffer descriptor endianess feature, a legacy buffer descriptor is exposited in the following C-language structure.

```
typedef struct
{
   uint16_t status;       /* control and status */
   uint16_t length;       /* transfer length */
   uint8_t *data;         /* buffer address */
} NBUF;
```

All the elements of the structure must be written in big endian; a 16-bit (status and length) and a 32-bit (pointer to data) endianess conversion is required in software when accessing elements of the structure that describe a buffer descriptor. The same applies for read accesses.

For the enhanced ENET, if buffer the descriptor endianess feature is enabled, then the following C-language structure describes a buffer descriptor.

```
typedef struct
{
   uint16_t length;       /* transfer length */
   uint16_t status;       /* control and status */
   uint8_t *data;         /* buffer address */
} NBUF;
```

Then, all the elements of the structure can be accessed in native little endian.

Note the change between length and status elements because the hardware endianess is applied in 32-bit accesses and not in 16-bit accesses.

Legacy buffer descriptors are used in this example instead of enhanced BD for simplicity, but the same apply to both modes.

4.12.4 Hardware impact

There are no hardware changes required to be compatible with previous versions.

4.13 Voltage reference (VREF)

4.13.1 Features

The features for the VREF module are unchanged.

4.13.2 Impacted register

The memory map and fields for the VREF module remain the same. The only thing that changes is the encoding for the VREF_SC[MODE_LV] register.

The table below shows how the encodings for the MODE_LV field have changed.

Table 17. VREF_SC[MODE_LV] encoding

MODE_LV	Kinetis 100 MHz rev 1.x	Kinetis 120 MHz
00	Bandgap on only	Bandgap on only
01	Reserved	Tight-regulation buffer enabled
10	Tight-regulation buffer enabled	Reserved

Table continues on the next page...

Table 17. VREF_SC[MODE_LV] encoding (continued)

MODE_LV	Kinetis 100 MHz rev 1.x	Kinetis 120 MHz
11	Reserved	Reserved

4.13.3 Software impact

All applications that are using the VREF module will require changes to account for the reorganized encoding of the VREF_SC[MODE_LV] field.

4.13.4 Hardware impact

No hardware impact.

4.14 Miscellaneous Control Module (MCM)

4.14.1 Features

The MCM module features are unchanged, but there is one register that has changed.

4.14.2 Impacted registers

The MCM_SRAMAP register has changed. The table below shows how the register naming has changed.

Table 18. MCM memory map comparison

Register	Location	Kinetis 100 MHz Rev. 1.x	Kinetis 120 MHz
MCM control register	0xE008_000C	MCM_SRAMAP	MCM_CR

4.14.2.1 MCM_CR

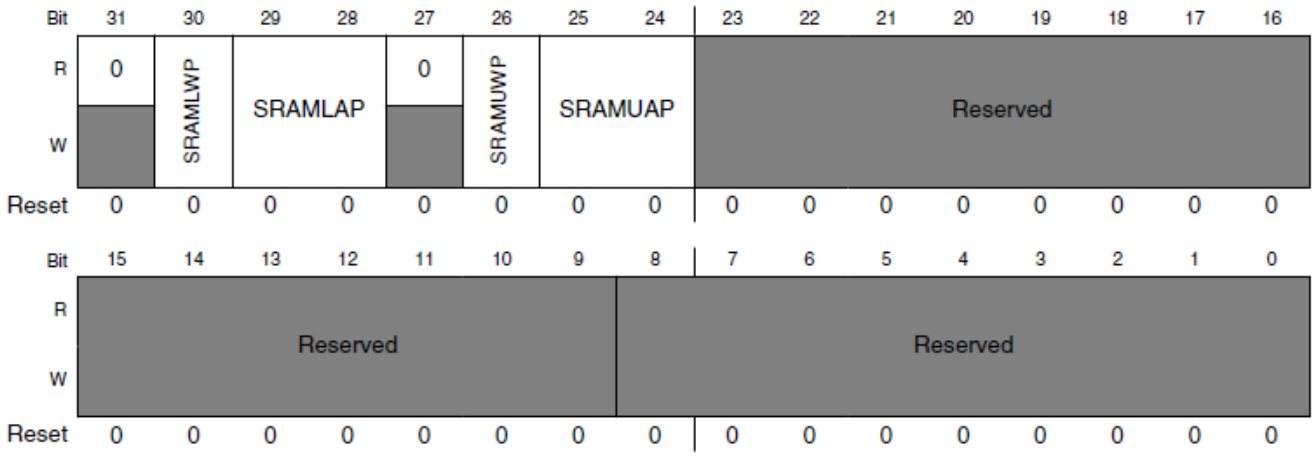
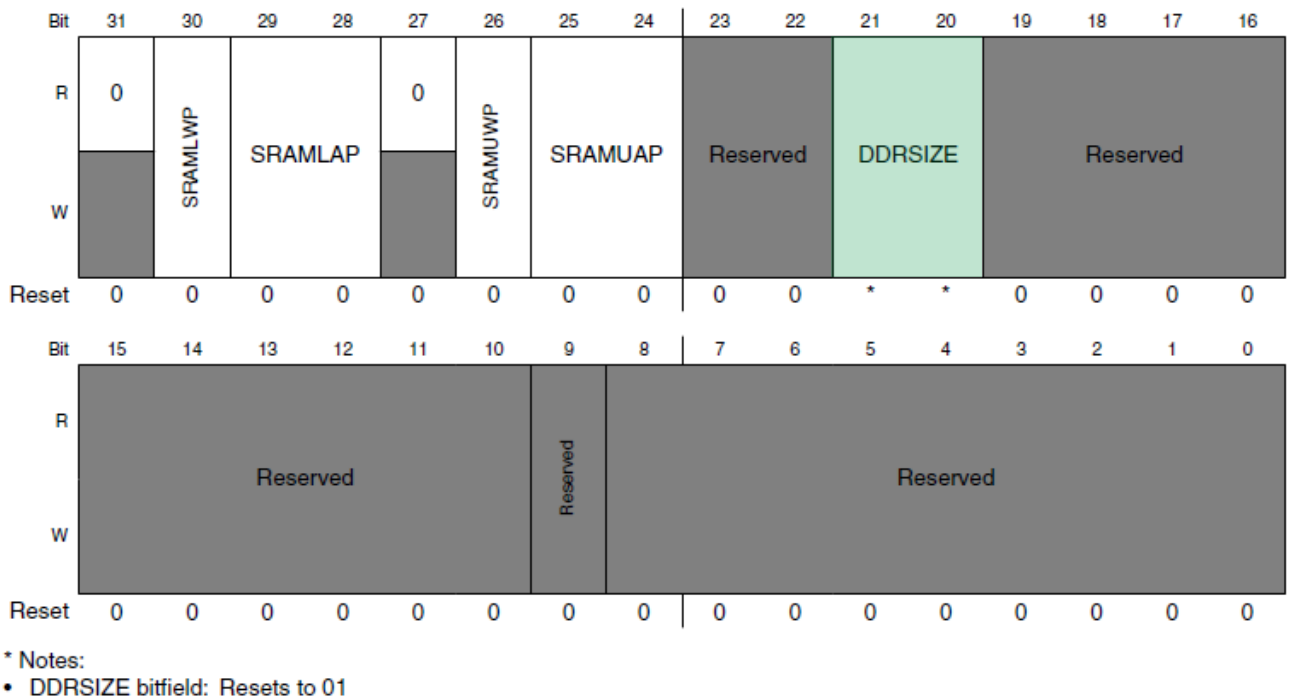


Figure 116. MCM_SRAMP—Kinetis 100 MHz Rev. 1.x



* Notes:
- DDRSIZE bitfield: Resets to 01

Figure 117. MCM_CR—Kinetis 120 MHz

Added bits/fields:
- DDRSIZE

4.14.3 Software impact

There are no software changes required to execute existing code. If using an updated header file, then any references to the MCM_SRAMAP would need to be changed to MCM_CR. If using DDR, then the MCM_CR[DDR_SIZE] field should be programmed appropriately.

4.14.4 Hardware impact

No hardware impact.

4.15 External Watchdog Monitor (EWM)

4.15.1 Features

The EWM module adds the ability to generate an interrupt request when the $\overline{\text{EWM\_OUT}}$ signal is asserted.

4.15.2 Impacted registers

The EWM module adds one bit to allow for enabling the new interrupt feature.

4.15.2.1 EWM_CTRL

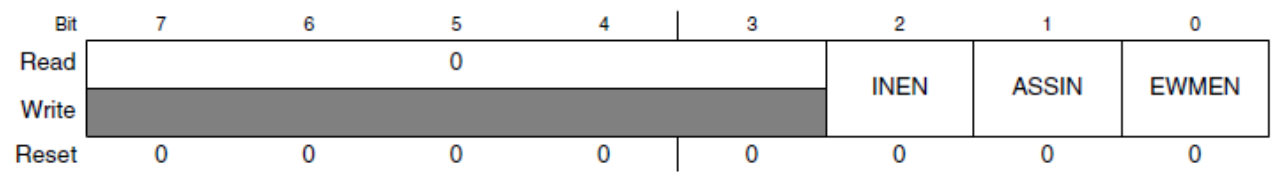


Figure 118. EWM_CTRL—Kinetis 100 MHz Rev. 1.x

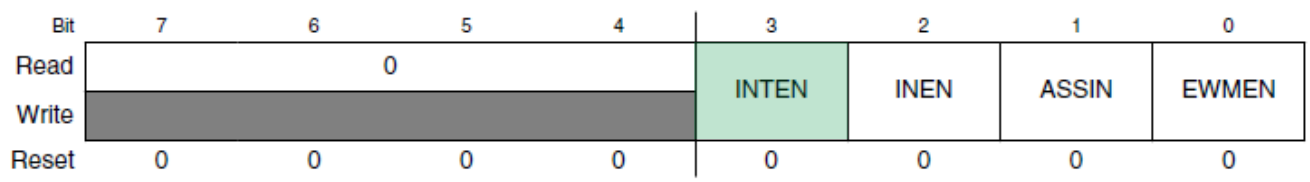


Figure 119. EWM_CTRL—Kinetis 120 MHz

Added bits/fields:

- INTEN

4.15.3 Software impact

There are no software changes required to execute existing code. Software changes are required to use new features.

4.15.4 Hardware impact

No hardware impact.

4.16 Watchdog timer (WDOG)

4.16.1 Features

The features of the WDOG module are unchanged, but there is one change to the registers.

4.16.2 Impacted registers

The WDOG module has one register bit that has been removed. Previously, the WDOG_STCTRLH's STNDBYEN and STOPEN bit were both used to control the operation of the watchdog in STOP mode. To avoid confusion the redundant STNDBYEN bit has been removed. Now the STNDBYEN bit is an always '1' reserved bit.

4.16.2.1 WDOG_STCTRLH

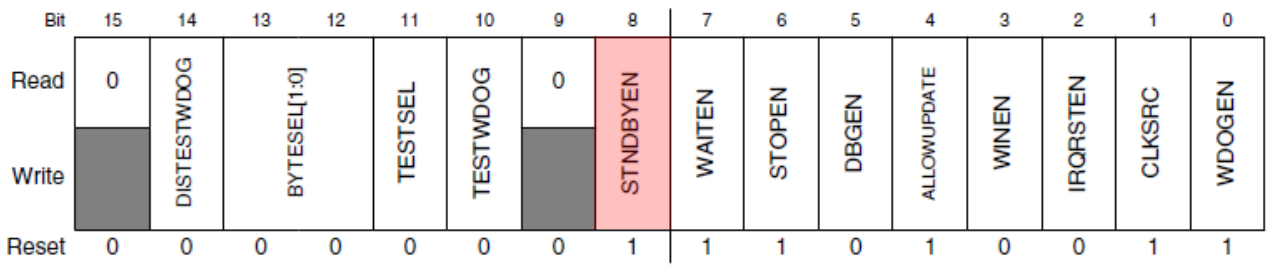


Figure 120. WDOG_STCTRLH—Kinetis 100 MHz Rev. 1.x

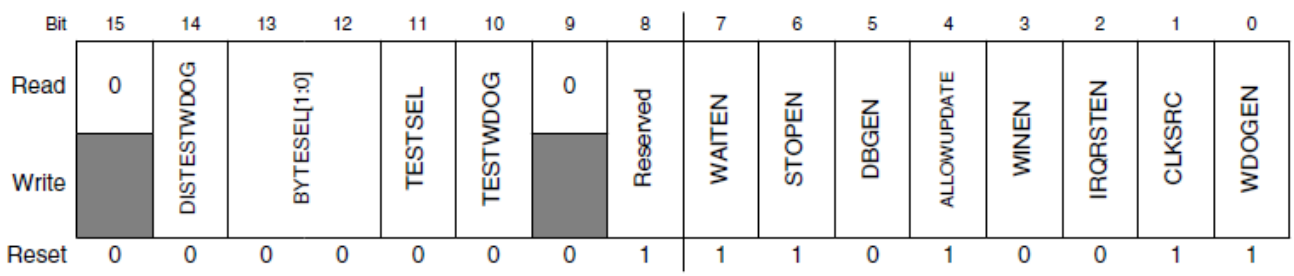


Figure 121. WDOG_STCTRLH—Kinetis 120 MHz

Removed bits/fields:

- STNDBYEN

4.16.3 Software impact

The WDOG_STCTRLH[STNDBYEN] bit has been removed from the register and Freescale provided header files. If any references were made to this bit in software it should be removed.

4.16.4 Hardware impact

No hardware impact.

5 Modules with New Instantiations

5.1 Direct Memory Access Multiplexer (DMAMUX)

Due to the increased number of DMA channels (32 instead of 16), a second DMA mux has been added. The second DMA mux selects the DMA request inputs for DMA channels 16-31.

All of the DMA sources on DMAMUX0 are unchanged, but to allow as much flexibility as possible some of the sources are duplicated on DMAMUX1.

5.2 Oscillator (OSC)

A second oscillator has been added (OSC1). The new OSC pin functions are muxed over PTE24 and PTE25, so if the second OSC is used it will impact hardware if any of the functions on PTE24 and PTE25 were being used.

OSC1 can be used as an input to the MCG, so the addition of the second OSC does affect the MCG module. Refer to the MCG section for more information.

5.3 Analog-to-Digital Converter (ADC)

The number of ADCs and PGAs increases from two to four.

The change to the number of ADCs impacts the PDB module as additional channels are required for the new ADC instantiations. Refer to the PDB section for more information.

The SIM module also adds additional bits for configuration and control of the new ADC modules. Refer to the SIM section for more information.

5.4 General purpose input/output (GPIO)

The Kinetis 120 MHz devices are available in larger packages than the 100MHz devices (256BGA package option). Because of the increased number of pins an additional GPIO port has been added for the 120 MHz devices (PORTF).

6 Unchanged Modules

The table below lists the modules that do not have any significant functional changes

Module	Comments
NMI	
XBS	The number of slave and master ports have been increased, but functionality is the same.
MPU	
EzPort	
MMCAU	
FlexBus	
CRC	
DAC	
CMT	
USBDCD	
FlexCAN	
DSPI	
I2C	
SDHC	

7 Appendices

7.1 Pin Multiplexing

The table below shows pin multiplexing differences between the Kinetis 100 MHz Rev 1.x devices and Kinetis 120 MHz devices where they are available in the same package. Some of the Kinetis 120 MHz devices are also available in a 256 MAPBGA package. This section is focused on making it easier to reuse hardware between revisions, so the new package and the pins that are added are not covered here because a new board design would be required to use the larger package.

The table uses the following conventions:

- **(+)**—added from Kinetis 100 MHz Rev. 1.x to Kinetis 120 MHz
- **(-)**—removed from Kinetis 100 MHz Rev. 1.x to Kinetis 120 MHz
- **Shadedregion**—changed from Kinetis 100 MHz Rev. 1.x to Kinetis 120 MHz

K60 144 MAPBGA	K60 144 LQFP	ALT0	ALT1	ALT2	ALT3	ALT4	ALT5	ALT6	ALT7
L5	–	(+) RTC_WAKEUP_b							
D3	1								(+) RTC_CLKOUT
D2	2								(+) SPI1_SIN

Table continues on the next page...

E4	4								(+) SPI1_SOUT
E2	8							(+) FTM3_CH0	
E1	9							(-) I2S0_CLKIN (+) FTM3_CH1	(+) USB_SOF_OUT
F4	10					I2S0_RXD →I2S0_RXD0		(+) FTM3_CH2	
F3	11	(+) ADC2_SE16		(+) I2S0_RXD1				(+) FTM3_CH3	
F2	12	(+) ADC2_SE17		(+) I2S0_TXD1				(+) FTM3_CH4	
F1	13					I2S0_TXD →I2S0_TXD0		(+) FTM3_CH5	
G4	14	(+) ADC3_SE16						(+) FTM3_CH6	
G3	15	(+) ADC3_SE17						(+) FTM3_CH7	
J1	23	ADC0_DP1 → PGA2_DP/ ADC2_DP0/ ADC3_DP3/ ADC0_DP1							
J2	24	ADC0_DM1 → PGA2_DM/ ADC2_DM0/ ADC3_DM3/ ADC0_DM1							

Table continues on the next page...

K1	25	ADC1_DP1 → PGA3_DP/ ADC3_DP0/ ADC2_DP3/ ADC1_DP1							
K2	26	ADC1_DM1 → PGA3_DM/ ADC3_DM0/ ADC2_DM3/ ADC1_DM1							
L4	39	DAC1_OUT/ CMP2_IN3/ ADC1_SE23 → DAC1_OUT/ CMP0_IN4/ CMP2_IN3/ ADC1_SE23							
M4	45	ADC0_SE17 → ADC0_SE17/ EXTAL1				(+) I2S1_TX_FS			(+) I2S1_RXD1
K5	46	ADC0_SE18 → ADC0_SE18/XTAL1				(+) I2S1_TX_BCLK			(+) I2S1_TXD1
K4	47	(+) ADC3_SE5b		(+) ENET_1588_CLKIN		(-) ENET_1588_CLKIN (+) I2S1_TXD0			
J4	48	(+) ADC3_SE4b				(+) I2S1_MCLK			
H4	49	(+) ADC3_SE7a							

Table continues on the next page...

J5	50			UART0_CTS_b → UART0_CTS_b/ UART0_COL_b					
M8	55			(+) USB_CLKIN				I2S0_RX_BCLK → I2S0_TX_BCLK	
J7	58	(+) ADC3_SE6a		(+) ULPI_CLK		(+) I2S1_RXD0	(+) CLKOUT		
J8	59			(+) ULPI_DIR		(+) I2S1_RX_BCLK			
K8	60			(+) ULPI_NXT		(+) I2S1_RX_FS			
L8	61	(+) ADC3_SE5a		(+) ULPI_STP					
M9	62	(+) ADC3_SE4a		(+) ULPI_DATA0					
L9	63	(+) ADC3_SE15		(+) ULPI_DATA1					
K9	64							I2S0_TXD → I2S0_TXD0	
L10	66	(+) CMP3_IN0						I2S0_TX_BCLK → I2S0_RX_BCLK	(+) I2S0_TXD1
L11	67	(+) CMP3_IN1						I2S0_RXD → I2S0_RXD0	
K10	68	(+) CMP3_IN2			UART0_CTS_b →UART0_CTS_b/ UART0_COL_b				(+) I2S0_RXD1
K11	69								(-) I2S0_CLKIN
M12	72	EXTAL → EXTAL0							

Table continues on the next page...

M11	73	XTAL → XTAL0						LPT0ALT1 → LPTMR0_ALT1	
K12	75	(+) CMP3_IN4		(+) ULPI_DATA2					
J12	76	(+) CMP3_IN5		(+) ULPI_DATA3					
J11	77	(+) ADC2_SE15		(+) ULPI_DATA4					
J10	78	(+) ADC2_SE14		(+) ULPI_DATA5					
H12	79	(+) ADC2_SE13		(+) ULPI_DATA6					
H11	80	(+) ADC2_SE12		(+) ULPI_DATA7					
H10	81	ADC0_SE8/ ADC1_SE8/ TSI0_CH0 → ADC0_SE8/ ADC1_SE8/ ADC2_SE8/ ADC3_SE8/ TSI0_CH0							
H9	82	ADC0_SE9/ ADC1_SE9/ TSI0_CH6 → ADC0_SE9/ ADC1_SE9/ ADC2_SE9/ ADC3_SE9/ TSI0_CH6							

Table continues on the next page...

G11	84				UART0_CTS_b → UART0_CTS_b/ UART0_COL_b				
E12	91					(+) I2S1_TX_BCLK			
E11	92					(+) I2S1_TX_FS			
E10	95					(+) I2S1_TXD0			
E9	96					(+) I2S1_TXD1			
D10	99	(+) ADC2_SE4a					FB_AD31 → FB_AD31/ NFC_DATA15		
D9	100	(+) ADC2_SE5a					FB_AD30 → FB_AD310 NFC_DATA14		
C12	101						FB_AD29 → FB_AD29/ NFC_DATA13		
C11	102						FB_AD28 → FB_AD28/ NFC_DATA12	(+) CMP3_OUT1	
B12	103					(-) I2S0_TXD	FB_AD14 → FB_AD14/ NFC_DATA11	(+) I2S0_TXD1	
B11	104						FB_AD13 → FB_AD13/ NFC_DATA10	(+) I2S0_TXD0	
A12	105						FB_AD12 → FB_AD12/ NFC_DATA9	(+) I2S0_TX_FS	

Table continues on the next page...

A11	106						FB_CLKOUT → CLKOUT	(+) I2S0_TX_BCLK	
A9	109						FB_AD11 → FB_AD11/ NFC_DATA8		(+) I2S1_TX_BCLK
D8	110				(+) LPTMR0_ALT2	LPTMR0_ALT2 → I2S0_RXD0	FB_AD10 → FB_AD10/ NFC_DATA7		(+) I2S1_TX_FS
C8	111					(+) I2S0_RX_BCLK	FB_AD9 → FB_AD9/ NFC_DATA6	(+) I2S0_MCLK	
B8	112				(+) USB_SOF_OUT	(+) I2S0_RX_FS	FB_AD8 → FB_AD8/ NFC_DATA5		
A8	113				(-) I2S0_MCLK (+) FTM3_CH4	I2S0_CLKIN → I2S0_MCLK	FB_AD7 → FB_AD7/ NFC_DATA4		
D7	114				(+) FTM3_CH5		FB_AD6 → FB_AD6/ NFC_DATA3		
C7	115	(-) CMP0_IN4			(+) FTM3_CH6		FB_AD5 → FB_AD5/ NFC_DATA2	(+) I2S1_MCLK	
B7	116				(+) FTM3_CH7	I2S0_RXD → I2S0_RXD1	FB_RW_b → FB_RW_b/ NFC_WE		
A6	123							(+) NFC_RB	
D5	124							(+) NFC_CE0_b	
C5	125							(+) NFC_CE1_b	
A5	127					(+) FTM3_CH0		(+) I2S1_RXD1	
D4	128					(+) FTM3_CH1		(+) I2S1_RXD0	

Table continues on the next page...

C4	129					(+) FTM3_CH 2		(+) I2S1_RX_ FS	
B4	130					(+) FTM3_CH 3		(+) I2S1_RX_ BCLK	
A4	131						FB_AD2 → FB_AD2/ NFC_DAT A1		
A3	132				UART0_C TS_b → UART0_C TS_b/ UART0_C OL_b		FB_AD1 → FB_AD1/ NFC_DAT A0		
C9	137							FB_A16 → FB_A16/ NFC_CLE	
B9	138							FB_A17 → FB_A17/ NFC_ALE	
B3	139							FB_A18 → FB_A18/ NFC_RE	
B1	141				(+) FTM3_FLT 0				

7.2 Memory Map

7.2.1 Memory map—Kinetis 100 MHz Rev. 1.x

System 32-bit Address Range	Destination Slave	Access
0x0000_0000–0x0FFF_FFFF	Program flash and read-only data (Includes exception vectors in first 1024 bytes)	All masters
0x1000_0000–0x13FF_FFFF	• For MK60DN256ZVLQ10: Reserved • For MK60DX256ZVLQ10: FlexNVM • For MK60DN512ZVLQ10: Reserved • For MK60DN256ZVMD10: Reserved • For MK60DX256ZVMD10: FlexNVM • For MK60DN512ZVMD10: Reserved	All masters
0x1400_0000–0x17FF_FFFF	For devices with FlexNVM: FlexRAM For devices with program flash only: Programming acceleration RAM	All masters
0x1800_0000–0x1FFF_FFFF	SRAM_L: Lower SRAM (ICODE/DCODE)	All masters
0x2000_0000–0x200F_FFFF	SRAM_U: Upper SRAM bitband region	All masters
0x2010_0000–0x21FF_FFFF	Reserved	–
0x2200_0000–0x23FF_FFFF	Aliased to SRAM_U bitband	Cortex-M4 core only
0x2400_0000–0x3FFF_FFFF	Reserved	–
0x4000_0000–0x4007_FFFF	Bitband region for peripheral bridge 0 (AIPS-Lite0)	Cortex-M4 core & DMA/EzPort
0x4008_0000–0x400F_EFFF	Bitband region for peripheral bridge 1 (AIPS-Lite1)	Cortex-M4 core & DMA/EzPort
0x400F_F000–0x400F_FFFF	Bitband region for general purpose input/output (GPIO)	Cortex-M4 core & DMA/EzPort
0x4010_0000–0x41FF_FFFF	Reserved	–
0x4200_0000–0x43FF_FFFF	Aliased to peripheral bridge (AIPS-Lite) and general purpose input/output (GPIO) bitband	Cortex-M4 core only
0x4400_0000–0x5FFF_FFFF	Reserved	–
0x6000_0000–0x7FFF_FFFF	FlexBus (External Memory - Write-back)	All masters
0x8000_0000–0x9FFF_FFFF	FlexBus (External Memory - Write-through)	All masters
0xA000_0000–0xDFFF_FFFF	FlexBus (External Peripheral - Not executable)	All masters
0xE000_0000–0xE00F_FFFF	Private peripherals	Cortex-M4 core only
0xE010_0000–0xFFFF_FFFF	Reserved	–

7.2.2 Memory map—Kinetis 120 MHz

System 32-bit Address Range	Destination Slave	Access	Slave Port
0x0000_0000–0x07FF_FFFF	Program flash and read-only data (Includes exception vectors in first 1024 bytes)	All masters	S0
0x0800_0000–0x0FFF_FFFF	DRAM Controller (Aliased Area)	Cortex-M4 core (M0) only	S5
0x1000_0000–0x13FF_FFFF	FlexNVM	All masters	S0
0x1000_0000–0x13FF_FFFF	Reserved	–	–
0x1400_0000–0x17FF_FFFF	For devices with FlexNVM: FlexRAM	All masters	S0
0x1400_0000–0x17FF_FFFF	For devices with program flash only: Programming acceleration RAM	–	S0
0x1800_0000–0x1BFF_FFFF	FlexBus (Aliased Area). 0x1800_0000-0x1BFF_FFFF are mapped to the same access space of 0x9800_0000-0x9BFF_FFFF.	Cortex-M4 core (M0) only	–
0x1C00_0000–0x1FFF_FFFF	SRAM_L: Lower SRAM (ICODE/DCODE)	All masters	–
0x2000_0000–0x200F_FFFF	SRAM_U: Upper SRAM bitband region	All masters	–
0x2010_0000–0x21FF_FFFF	Reserved	–	–
0x2200_0000–0x23FF_FFFF	Aliased to TCMU SRAM bitband	Cortex-M4 core only	–
0x2400_0000–0x3FFF_FFFF	Reserved	–	–
0x4000_0000–0x4007_FFFF	Bitband region for AIPS0	Cortex-M4 core & DMA/EzPort	S2
0x4008_0000–0x400F_EFFF	Bitband region for AIPS1	Cortex-M4 core & DMA/EzPort	S3
0x400F_F000–0x400F_FFFF	Bitband region for GPIO	Cortex-M4 core & DMA/EzPort	S3
0x4010_0000–0x41FF_FFFF	Reserved	–	–
0x4200_0000–0x43FF_FFFF	Aliased to AIPS and GPIO bitband	Cortex-M4 core only	–
0x4400_0000–0x5FFF_FFFF	Reserved	–	–
0x6000_0000–0x6FFF_FFFF	Flexbus (External memory - Write-back)	All masters	S4
0x7000_0000–0x7FFF_FFFF	DRAM Controller - Write-back	Cortex-M4 core (M1), eDMA(M2)	S5, S6, and S7
0x7000_0000–0x7FFF_FFFF	DRAM Controller	LCD (M4)	S5, S6, and S7
0x7000_0000–0x7FFF_FFFF	DRAM Controller	LCD(M5), eSDHC/NFC (M3), ENET(M7), USB (M6)	S5
0x8000_0000–0x8FFF_FFFF	DRAM Controller - Write-through	Cortex-M4 core (M1), eDMA(M2)	
0x8000_0000–0x8FFF_FFFF	DRAM Controller	LCD (M4)	S5, S6, and S7
0x8000_0000–0x8FFF_FFFF	DRAM Controller	LCD (M5), eSDHC/NFC (M3), ENET(M7), USB (M6)	S5, S6, and S7
0x9000_0000–0x9FFF_FFFF	FlexBus (External memory - Write-through)	All masters	S4
0xA000_0000–0xDFFF_FFFF	FlexBus (External peripheral - not executable)	All masters	S4
0xE000_0000–0xE00F_FFFF	Private Peripherals	Cortex-M4 core only	–
0xE010_0000–0xFFFF_FFFF	Reserved	–	–

How to Reach Us:

Home Page:
www.freescale.com

Web Support:
http://www.freescale.com/support

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
+1-800-521-6274 or +1-480-768-2130
www.freescale.com/support

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.freescale.com/support

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor China Ltd.
Exchange Building 23F
No. 118 Jianguo Road
Chaoyang District
Beijing 100022
China
+86 10 5879 8000
support.asia@freescale.com

Document Number: AN4526
Rev. 1.0, March 2013

freescale