Freescale Semiconductor
Application Note

Document Number: AN3622
Rev. 0, 03/2008

Comparison of the S12XS CRG Module with S12P CPMU Module

by: Steven McLaughlin
Applications Engineer
Microcontroller Solutions Group, Scotland

1 Introduction

This application note highlights and explains the differences that exist between the clock, reset, and power management unit (CPMU) module of the S12P devices, and the clocks and reset generator (CRG) module of the S12XS. The intended audience of this text is customers who have knowledge of the CRG module of the S12XS family and are thinking of migrating their applications to the S12P family of microcontrollers. As the clocking scheme on previous members of the S12 family is similar in many respects to the S12XS, this document will also be of interest when converting an existing S12 design to S12P.

Contents

2 CPMU Overview

Figure 1 features the block diagram of the CPMU module. Unlike the CRG, this module integrates the voltage regulator, autonomous periodical interrupt (API), and high temperature sensor blocks.

It also includes an integrated RC-based reference oscillator (IRC1M), which by default is set at 1 MHz.

On the S12XS, the default mode on startup (from reset or full stop mode) is operation from the external reference gated by the clock quality checker. The PLL can then be selected and configured to obtain a higher bus clock.

On the S12P, the default mode on startup (from reset or full stop mode) is operation from the PLL using the internal reference (IRC1M). Startup from an external oscillator is not automatic and can be enabled under software control. This is discussed in detail in Section 3, “CPMU Clocking Scheme.”

The CPMU contains an adaptive spike filter that can prevent spurious oscillator noise reaching the PLL and help prevent loss of lock events (use of the spike filter is available only for external clock modes and for certain oscillator and bus configurations).

The S12XS external oscillator can be configured for loop-controlled Pierce, full-swing Pierce, or external clock input, whereas the S12P external oscillator supports only a loop-controlled Pierce configuration.

The BDM clock on the S12P, by default, is 8 MHz. Figure 1 illustrates the BDM clock being derived from the PLL, which can be used with either the internal or external reference. The BDM clock source on the S12XS is derived by the bus clock dependent upon the oscillator or the PLL.

Note: While debugging, background debug mode (BDM) may experience delays due to the ramping and lock, as well as switching between clock run modes.

Both devices have different operating frequencies, as stated in Table 1.

Table 1. Frequency Comparison

	Maximum bus frequency (MHz)	Maximum VCO frequency (MHz)
S12XS	40	120
S12P	32	64

Unlike the CRG module, the CPMU module also includes a CPMU protection register (CPMUPROT) that is designed to protect the clock configuration registers from accidental overwrite. The registers protected are: CPMUSYNR, CPMUREFDIV, CPMUPLL, CPMUIRCTRIMH/L, and CPMUOSC.

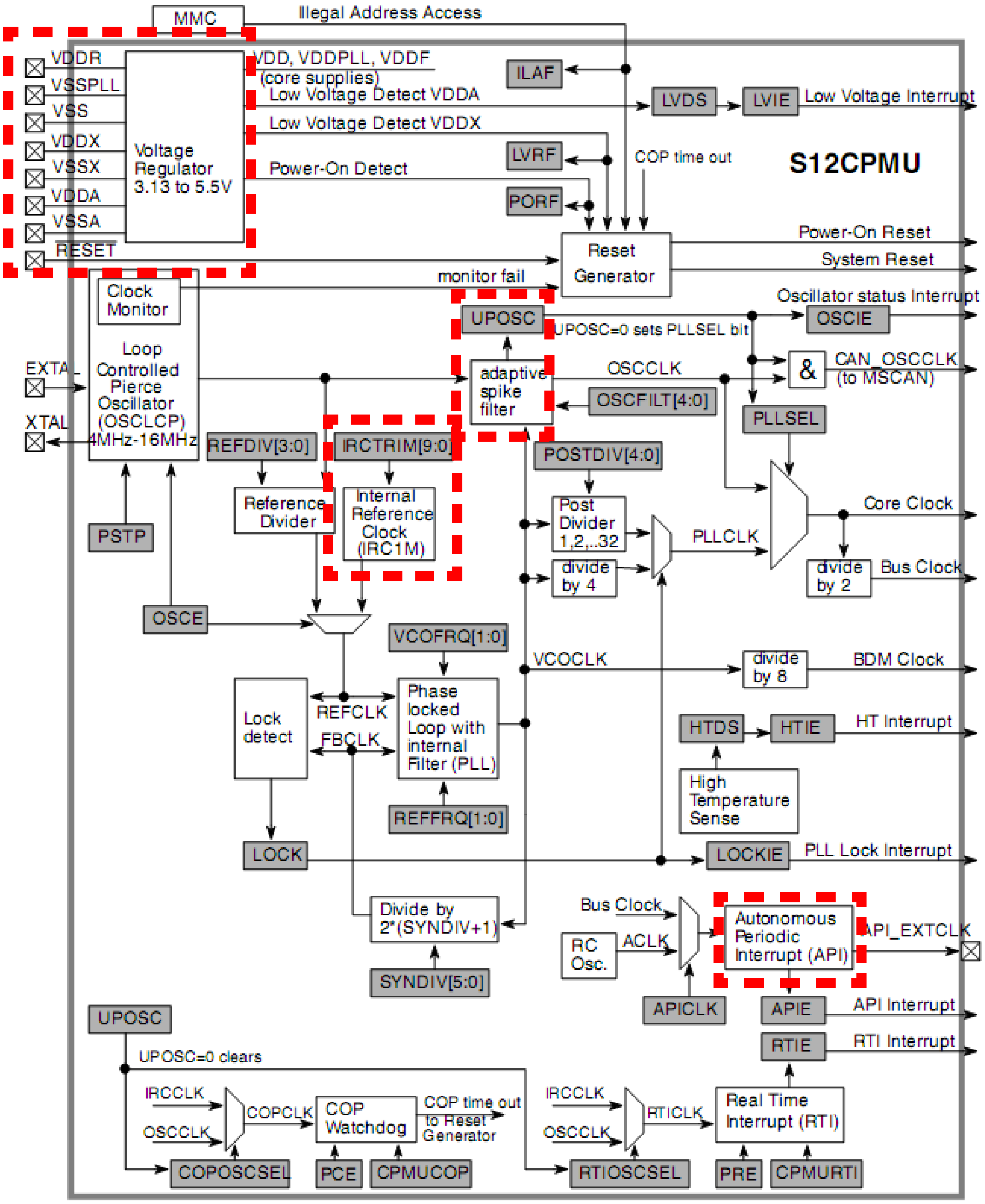


Figure 1. S12CPMU Block Diagram, Highlighting Major Changes from CRG Module

2.1 CPMU Internal Reference Clock (IRC1M)

The CPMU IRC1M provides a stable internal reference option for the PLL. The IRC1M will be factory trimmed before shipping, but can be re-trimmed for accuracy under specific operating conditions via the CPMUIRCTRML/H registers.

The internal reference of the CPMU offers a number of potential advantages, including:

- Design cost reductions for applications that do not require an external oscillator at all.

- PLL stability with the IRC1M reference that is sufficient for LIN communications.
- Faster startup time — the startup time of the internal reference clock (typically 50 µs) is significantly faster than that of an external oscillator (typically milliseconds). This can avoid the overhead of an application that is waiting for an external oscillator to start.
- Quick reaction to handling noise (if the adaptive spike filter bits have been enabled) in comparison to the S12XS CRG.

Where higher clock accuracy is required, for instance in CAN nodes, an external loop-controlled Pierce oscillator is included (the CAN module clock is directly sourced only from the external clock reference).

The majority of other designs, where such high clock accuracy is not required, can be easily achieved using only the internal reference.

2.2 CPMU Adaptive Spike Filter

The adaptive spike filter can be used with an external reference to filter noise, thus avoiding possible loss of lock situations. This is performed by a digital hardware filter and is valid only for certain combinations of oscillator and f_{VCO}. The filter is also responsible for qualifying the PLL.

To make use of the filter, select an external reference and PLL configuration where the VCOCLK frequency is eight times the oscillator frequency. The register OSCFILT contains filter bits that are used to configure the adaptive spike filter. The value of OSCFILT is calculated from the ratio of VCOCLK to OSCCLK and divided by 2. In PBE mode (discussed in Section 3, “CPMU Clocking Scheme”), when the PLL is not selected, the source of the bus clock is still enabled and will remain locked, to provide an appropriate f_{VCO} for the filter.

3 CPMU Clocking Scheme

The clocking scheme of this module differs from the CRG module primarily due to the inclusion of the 1 MHz internal reference clock (IRC1M).

The module has three system clock configurations which are listed and explained in Table 2. The following sections explain each of these operating modes in greater detail.

Table 2. CPMU Operating Modes

Operating Mode	Int Ref	Ext Ref	PLL On	Osc On
PLL Engaged Internal (PEI)	Yes	No	Yes	No
PLL Engaged External (PEE)	No	Yes	Yes	Yes
PLL Bypassed External (PBE)	No	Yes	No[1]	Yes

[1] The PLL is enabled but not used in the generation of the bus clock.

The default mode of operation is PEI mode. This mode is entered from system resets/power-on resets (PORs), and the default bus frequency is 8 MHz as described in Section 3.1.1, “PLL Engaged Internal Mode (PEI).”

3.1 Clock Modes

Initial startup of the CPMU module requires that the voltage regulator is in full performance mode, and the PLL and internal reference clock are on. The device will always start up from reset or full stop using the internal reference to generate the system clocks.

The S12P CPMU clock modes are described as follows:

- PEI: system clocks are derived from PLL — PLL reference clock = internal reference
- PEE: system clocks derived from PLL — PLL reference clock = external reference
- PBE: system clocks derived from external reference

The clock modes on the S12P CPMU have commonalities with the S12XS CRG. Fast wake is similar to PEI in that the system clocks start up using the PLLVCO clock and the oscillator does not start automatically. The key difference is that on the S12P the PLL has the internal reference clock (IRC1M), so it will ramp up and lock.

PEI replaces self clock mode (present on the CRG), so that in the event that the external oscillator fails, you are automatically put into PEI mode from PEE mode. In this event, the device will run with default settings.

The default mode on the S12X (under normal conditions) uses the external oscillator to derive the system clocks. This clock mode is equivalent to PBE mode on the S12P. The key difference is that this is not the default mode.

The typical operational mode on the S12X is with the PLL deriving the system clocks with the external oscillator as the reference clock. This is equivalent to PEE on the S12P.

Figure 2 illustrates the available clock modes, and how to move between them.

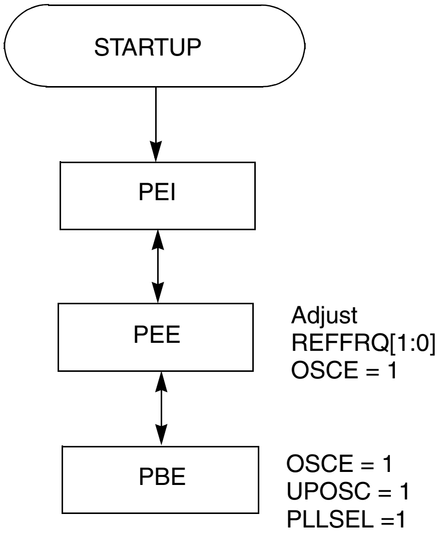


Figure 2. CPMU Clock Modes

The CPMU will start in PEI mode. To enter PEE mode, appropriate adjustment to the PLL divider bits (REFFEQ[1:0]) is required, and the OSCE bit should be set. Both PEI and PEE modes have a bus frequency dependent on the PLL. After the UPOSC bit has been set, PBE mode is entered by clearing PLLSEL, and the bus frequency is derived directly from the oscillator. By setting PLLSEL it is possible to return to PEE mode; similarly, clearing OSCE will return the run mode to PEI.

3.1.1 PLL Engaged Internal Mode (PEI)

PEI mode is entered upon startup, as shown in Figure 2. The bus clock is based on the PLL clock and the PLL reference clock is internally generated. The default PLL configuration is for 64 MHz VCOCLK with POSTDIV set to 0x03, resulting in a PLL clock of 16 MHz and thus a bus clock of 8 MHz.

In this mode, the clock sources for COP and RTI are based on the internal reference clock generator (IRC1M).

NOTE

The default bus frequency may vary depending on the setting of the trim registers.

3.1.2 PLL Engaged External Mode (PEE)

To enter PEE mode from PEI mode:

1. Configure the PLL for the desired bus frequency, via the CPMUSYNR, CPMUREFDIV, and CPMUPOSTDIV registers.
2. Enable the external oscillator bit (OSCE in CPMUOSC register).

For PEE mode, the bus clock is derived from the PLL, which is supplied from an external reference. The accuracy of the system clocks will depend on the oscillator and oscillator circuitry used. The module also contains an adaptive spike filter, which prevents noise spikes from appearing on the system clock signals. Filter bits (OSCFILT of the CPMUOSC register) are used to configure the adaptive spike filter, and these are calculated from the ratio of VCOCLK to OSCCLK and divided by 2.

For example, when using a 4 MHz crystal (OSCCLK) and synthesizing a VCOCLK of 64 MHz, then OSCFILT must be set to 8 (in other words, 16 divided by 2).

The adaptive spike filter uses the VCO clock as a reference to continuously qualify the external oscillator clock. When using the oscillator as reference, the PLL locking to the reference acts as a clock quality checker, because stable oscillation will lead to the lock detector detecting lock condition. After it is locked, a fixed frequency and phase will have been achieved; therefore the VCOCLK can be used to sample the oscillator clock. Note: VCOCLK should be higher than the oscillator clock.

Because the adaptive spike filter uses the VCOCLK (from PLL) to continuously qualify the external oscillator, losing PLL lock (LOCK=0) means losing the oscillator status information as well (UPOSC=0).

Within PEE mode, the COP and RTI can be generated from either an internal or external reference, depending on their respective clock select bits.

3.1.3 PLL Bypassed External Mode (PBE)

PBE mode can be entered directly from PEE mode as follows:

1. Make sure the REFDIV bits are suitable to divide down the oscillator frequency if necessary — it could result in the PLL not locking if REFDIV[1:0] is incorrect.
2. Enable the external oscillator bit (OSCE in CPMUOSC register).
3. Wait for the UPOSC bit to set, indicating the oscillator starting up.
4. Select the oscillator as the bus clock (PLLSEL = 0).

For PBE mode, the bus clock is derived from the external reference. The main difference from PEE is that PLLSEL is cleared to allow system clocks derived from the OSCCLK. The PLL is still active to provide the VCOCLK to the adaptive spike filter, which qualifies the status of the external oscillator clock. The operation of the adaptive spike filter is described in Section 3.1.2, “PLL Engaged External Mode (PEE).”

The adaptive spike filter is responsible for the state of the UPOSC bit. When the UPOSC bit is set, this qualifies the PLL and subsequently clears PLLSEL, which derives the system clocks from the OSCCLK, in other words from the external reference.

Within PBE mode, the COP and RTI can be generated from either an internal or external reference, depending upon their respective clock select bits.

Because the adaptive spike filter uses the VCOCLK (from PLL) to continuously qualify the external oscillator, losing PLL lock (LOCK=0) means losing the oscillator status information as well (UPOSC=0).

The impact of losing the oscillator status in PBE mode is as follows:

- The MSCAN module, which can be configured to run on the oscillator clock, may need to be re-configured.
- PLLSEL is set automatically and the bus clock is switched back to the PLL clock.

Application software needs to be prepared to deal with the impact of losing the oscillator status at any time.

Table 3 below summarizes and compares using the clock modules on both the S12XS CRG and the S12P CPMU.

Table 3. Comparison of Operating the CRG and CPMU With and Without the PLL

	CRG Module	CPMU Module
Running with PLL	1. Device starts up on external oscillator. 2. Configure PLL registers. 3. Check PLL for lock. 4. Select PLL clock source.	1. Device starts on internal oscillator in PEI mode. 2. *(Optional) Configure PLL registers if necessary.* 3. *(Optional) Set OSCE bit to enable external oscillator in PEE mode.* 4. Check for lock.

Table 3. Comparison of Operating the CRG and CPMU With and Without the PLL

	CRG Module	CPMU Module
Running with external oscillator and no PLL	1. Device starts up on the external oscillator. 2. Select oscillator as clock source.	1. It is good practice to configure PLL registers appropriately. 2. Set OSCE bit to enable external oscillator in PEE mode. 3. Wait for UPOSC to set. 4. Select oscillator as clock.

3.2 Operating Modes

Run, wait, stop, and pseudo-stop are compatible between the CRG and CPMU. They all have the same function. However, on the CPMU, run mode has three different operating modes, as described in Section 3.1, “Clock Modes.”

3.2.1 Wait Mode

On the CPMU module, wait mode functions in the same way as run mode. Wait is compatible — however, on the S12X the IPLL is powered down, whereas it is not on the S12P. It is possible on the S12P CPMU to configure this action to occur in software, if required.

3.2.2 Stop Modes

Stop mode is entered by executing the CPU STOP instruction. When in stop mode, the voltage regulator is in reduced power mode, the API is available, and the PLL and internal reference are off. The stop modes on CPMU and CRG are similar in that the PSTP bit will determine whether full stop mode or pseudo-stop mode will be engaged. However, unlike the CRG, the CPMU has to take account of the state of OSCE to determine whether full stop or pseudo-stop is being used. Table 4 outlines the requirements for each stop mode, and how to maintain capability between both modules.

Table 4. Conditions for Full Stop and Pseudo-Stop Modes on CPMU

	PSTP	OSCE
Full Stop	0	0
Full Stop	0	1
Full Stop	1	0
Pseudo-Stop	1	1

It should also be noted that the CPMU module does not have fast wakeup or self-clock mode functionality, which are available on the CRG module. However, the fast wakeup from stop on an S12XS device is comparable to wakeup from stop on the S12P, at approximately 50 µs.

3.2.2.1 Stop in PEI Mode

Table 5 highlights the state of the module before, during, and after stop modes have been initiated from PEI mode.

Table 5. STOP for PEI

	Before / After Stop Mode	Stop Mode
IRC1M	On	Off
PLL	On	Off
REFCLK	IRC1M	Off
OSCLCP	Off	Off
Bus Clock	PLL-Based	Off

Table 5 shows the state of the various reference and system clocks in PEI mode. During full stop and pseudo-stop modes, the references, and therefore the system clocks, are disabled.

The timing diagram in Figure 3 explains the stop mode in PEI. The diagram shows normal running of the PLLCLK, which is then disturbed with the STOP instruction. Wakeup from stop mode on the CPMU module is comparable to the CRG module in that any interrupt will bring the device out of stop mode. Recovery from stop, t_{stp_rec}, is typically 50 µs. After the PLL begins to recover, it will take time to lock, t_{Lock}, which is defined as 150 + (256 / f_{ref}). After it is locked, the PLLCLK will return to its previous running frequency.

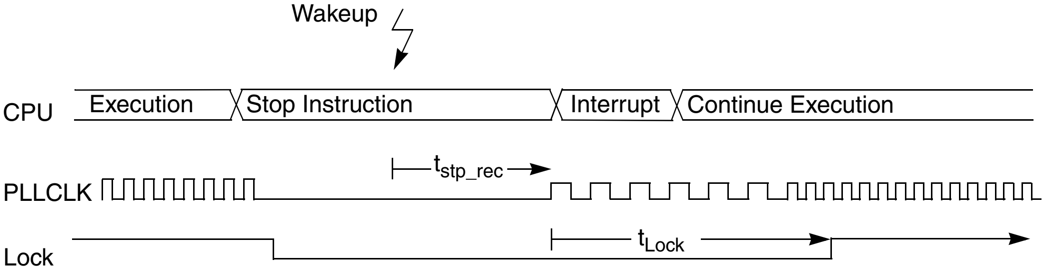


Figure 3. Stop Mode Using the PLL as Bus Clock

3.2.2.2 Stop in PEE Mode

Table 6 shows the state of the various reference and system clocks in PEE mode. During full stop the system clocks and references are disabled, and in pseudo-stop the only reference clock available is the external (oscillator). This allows the COP and RTI to function in pseudo-stop mode, similar to the CRG module.

Table 6. Stop for PEE

	Before / After Stop Mode	Stop Mode
IRC1M	On	Off

Table 6. Stop for PEE (continued)

	Before / After Stop Mode	Stop Mode
PLL	On	Off
REFCLK	OSCLCP-Based	Off
OSCLCP	On	Off; On if PSTP = 1
Bus Clock	PLL-Based	Off

3.2.2.3 Stop in PBE Mode

Table 7 and Table 8 represent stop modes on PBE for full stop and pseudo-stop modes, respectively. For PBE mode starting up, there will be a short period of time in which the oscillator is not qualified; therefore, the bus clock will be derived from the PLL (UPOSC = 0). In full stop mode, all system clocks are disabled. For pseudo-stop mode the oscillator will remain on if required (Table 8).

Table 7. Full Stop Mode for PBE

	Before Stop Mode	Stop Mode	After Stop Mode (UPOSC = 0)	After Stop Mode (UPOSC = 1)
IRC1M	On	Off	On	On
PLL	On	Off	On	On
REFCLK	OSCLCP-based	Off	OSCLCP-based	OSCLCP-based
OSCLCP	On	Off	On	On
Bus Clock	OSCLCP-based	Off	PLL-based	OSCLCP-based

Table 8. Pseudo-Stop Mode for PBE

	Before / After Stop Mode	Stop Mode
IRC1M	On	Off
PLL	On	Off
REFCLK	OSCLCP-based	Off
OSCLCP	On	On
Bus Clock	OSCLCP-based	Off

3.2.3 Reset Properties

The reset functionality on the S12P is similar to the S12XS, in that upon detection of any reset, an internal circuit drives the reset pin low for a number of clock cycles. The difference between the two is in the timing for which the reset pin is held low; 512 PLL clock cycles for S12P and 128 system clock cycles for the S12XS. The reset generator will then wait for a further 256 PLL clock cycles on the S12P, compared to 64 system clock cycles on the S12XS, and then sample the reset pin to determine the originating source.

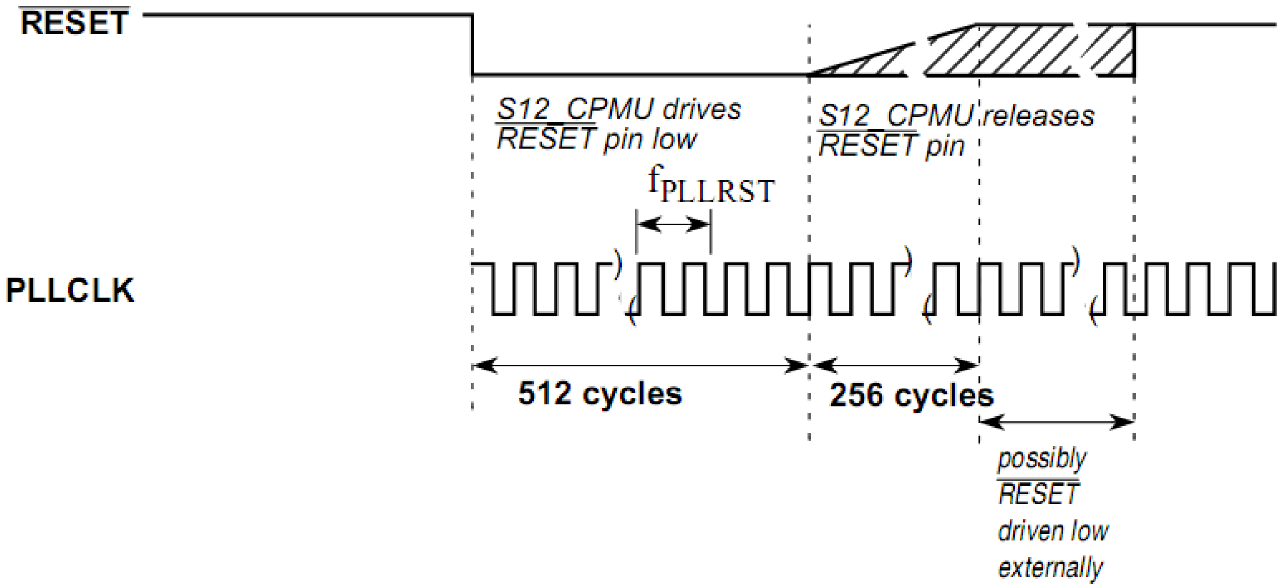

RESET
S12_CPMU drives RESET pin low
S12_CPMU releases RESET pin
f_PLLRST
PLLCLK
512 cycles
256 cycles
possibly RESET driven low externally

Figure 4. Reset Timing on the S12P

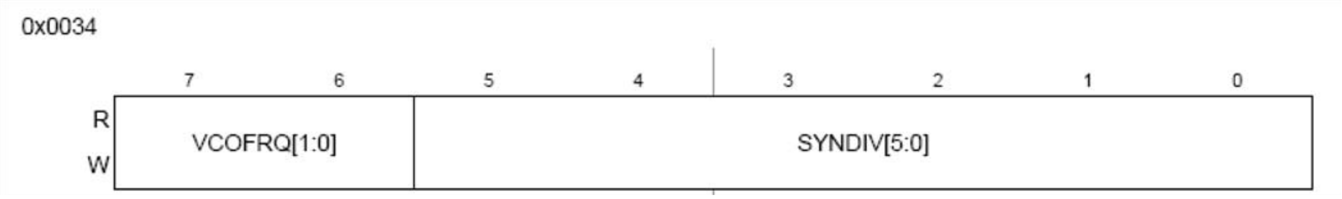
Both devices have a clock monitor reset, but it operates differently on each. On the S12XS, the clock monitor must be enabled and will eventually enter self-clock mode. Then it will begin a clock quality-check sequence to determine when valid oscillations occur to allow the device to switch back to the oscillator clock.

The S12P will generate a clock monitor reset (only with the oscillator enabled) if a loss of oscillation or the oscillator frequency is less than clock monitor failure-assert frequency (typically 400 kHz). The device will revert to PEI mode. In stop mode, the external reference and clock monitor are not available.

4 Register Differences

This section of the application note will concentrate upon the registers common to both the CRG and CPMU, and indicate the differences with individual bits to maintain compatibility. The CPMU module consists of more registers than the CRG module mainly because the CPMU module has the API and VREG functionality to consider, whereas these are separate from the CRG module on the S12XS. Therefore, to maintain device compatibility API and VREG code of the S12XS will have to be adjusted to suit S12P applications.

4.1 Synthesizer Register



0x0034

	7	6	5	4	3	2	1	0
R W	VCOFRQ[1:0]		SYNDIV[5:0]					

Figure 5. SYNR/CPMUSYNR Register

Table 9. Synthesizer Register Details Between CRG and CPMU Modules

	CRG Module	**CPMU Module**
Register Name	SYNR	CPMUSYNR
Register Address	0x0034	0x0034
Register State upon Reset	0x00	0x5F
Bit Changes on CPMU Register from CRG Module	None	

This register controls the multiplication factor of the PLL and selects the VCO frequency range. Between the two modules there is no difference with the individual bits. However, a difference exists with writing to the registers. To write anytime on the CPMUSYNDIV register, PLLSEL = 1, PROT=0; else write has no effect.

The CRG module can be written anytime, except when PLLSEL=1. It should be noted that writing to the CPMUSYNR register clears the LOCK (CPMUFLG register) status bit and a write to the SYNR register initializes the LOCK bit (CRGFLG register).

4.2 Reference Divider Register

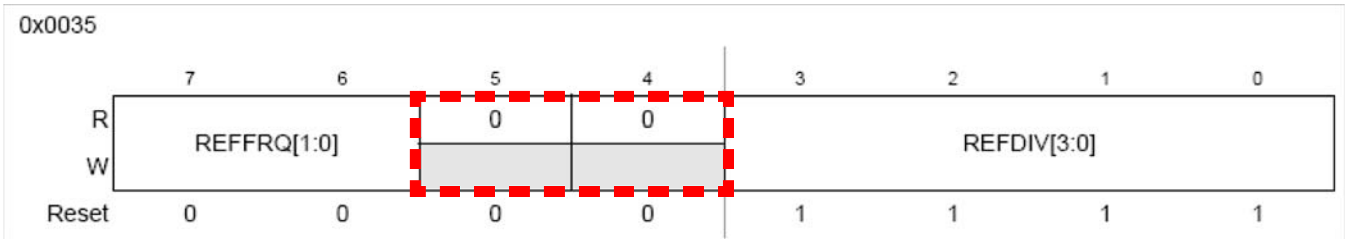


Figure 6. CPMU Reference Divider Register (CPMUREFDIV)

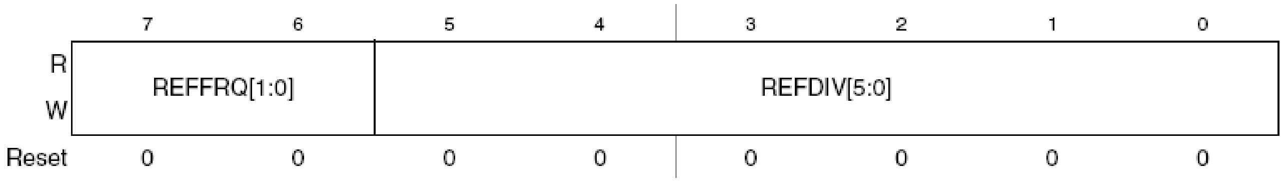


Figure 7. CRG Reference Divider Register (REFDIV)

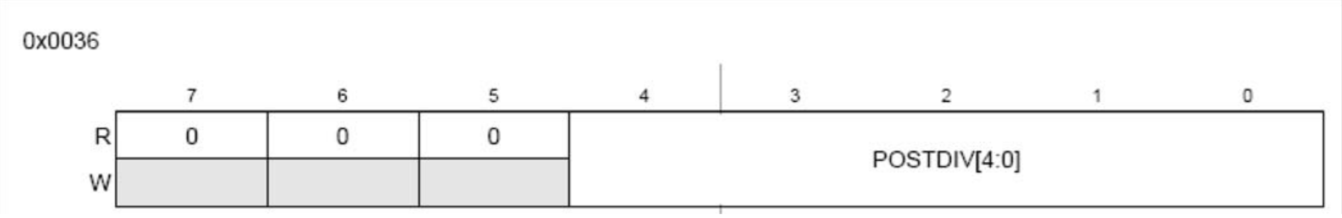
Table 10. Reference Divider Register Details Between CRG and CPMU Modules

	CRG Module	CPMU Module
Register Name	REFDIV	CPMUREFDIV
Register Address	0x0035	0x0035
Register State upon Reset	0x00	0x0F
Bit Changes on CPMU Register from CRG Module	Bits 4 and 5 are read only and REFDIV has less bits.	

This register provides a finer granularity for the PLL multiplier in steps, resulting in more choices for PLL operating frequencies.

The major differences between the registers are that the REFDIV bits, although they exist in both registers, are smaller in value in the CPMU module. This difference should be taken into account when migrating from S12XS to S12P applications.

4.3 Post Divider Register



0x0036

	7	6	5	4	3	2	1	0
R	0	0	0	POSTDIV[4:0]				
W								

Figure 8. Post Divider Register

Table 11. Post Divider Register Details between CRG and CPMU Modules

	CRG Module	CPMU Module
Register Name	POSTDIV	CPMUPOSTDIV
Register Address	0x0036	0x0036
Register State upon Reset	0x00	0x03
Bit Changes on CPMU Register from CRG Module	None.	

The POSTDIV register controls the frequency ratio between the VCOCLK and the PLLCLK. The register can be read anytime but can be written only when PLLSEL=1 in the CPMU.

The CRG module's POSTDIV register can be written anytime, except when PLLSEL=1.

The equations to find f_{PLL} also differ between the two modules.

$$f_{PLL(CRG)} = \frac{f_{VCO}}{(2xPOSTDIV)} \qquad \textbf{\textit{Eqn. 1}}$$

Because the CPMU module can run with an internal reference, a PLL-derived bus frequency can be generated with or without the PLL being locked. The automatic switch to Equation 3 below ensures that the bus clock remains in a safe range while the PLL is not locked.

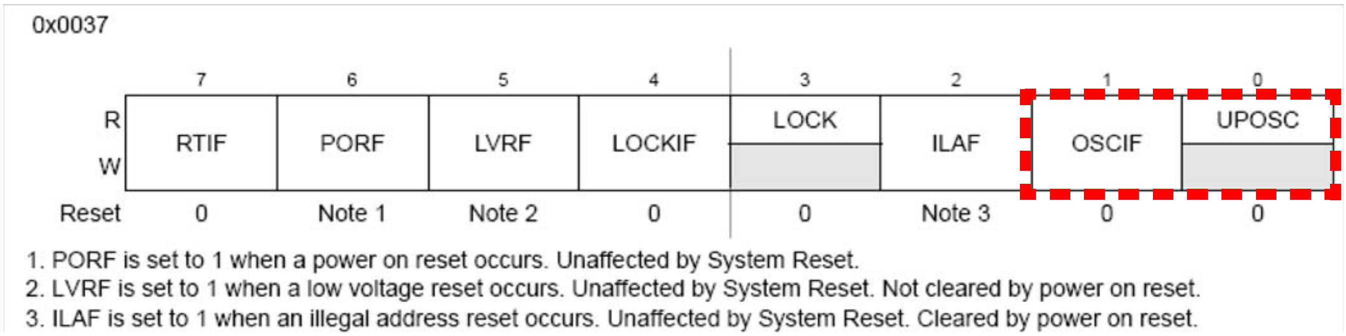
Lock = 1

$$f_{PLL(CPMU)} = \frac{f_{VCO}}{(1 + POSTDIV)} \qquad \textbf{\textit{Eqn. 2}}$$

Lock = 0

$$f_{PLL(CPMU)} = \frac{f_{VCO}}{4} \qquad \textbf{\textit{Eqn. 3}}$$

4.4 Flags Register

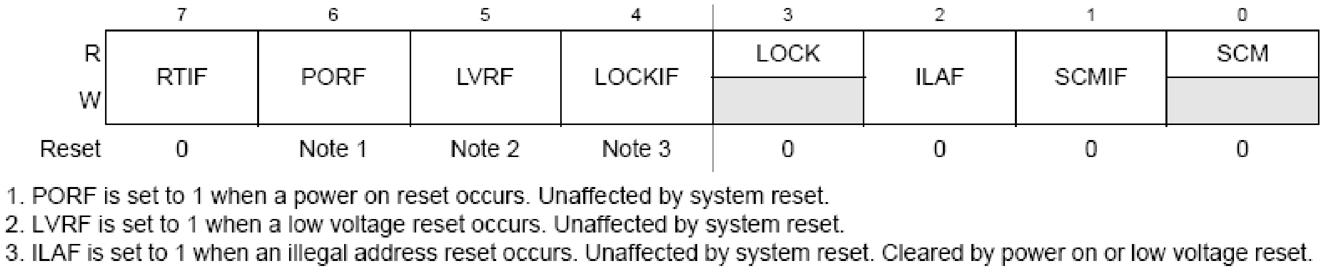


0x0037

	7	6	5	4	3	2	1	0
R	RTIF	PORF	LVRF	LOCKIF	LOCK	ILAF	OSCIF	UPOSC
W								
Reset	0	Note 1	Note 2	0	0	Note 3	0	0

1. PORF is set to 1 when a power on reset occurs. Unaffected by System Reset.
2. LVRF is set to 1 when a low voltage reset occurs. Unaffected by System Reset. Not cleared by power on reset.
3. ILAF is set to 1 when an illegal address reset occurs. Unaffected by System Reset. Cleared by power on reset.

Figure 9. CPMU Flags Register (CPMUFLG)



	7	6	5	4	3	2	1	0
R	RTIF	PORF	LVRF	LOCKIF	LOCK	ILAF	SCMIF	SCM
W								
Reset	0	Note 1	Note 2	Note 3	0	0	0	0

1. PORF is set to 1 when a power on reset occurs. Unaffected by system reset.
2. LVRF is set to 1 when a low voltage reset occurs. Unaffected by system reset.
3. ILAF is set to 1 when an illegal address reset occurs. Unaffected by system reset. Cleared by power on or low voltage reset.

Figure 10. CRG Flags Register (CRGFLG)

Table 12. Flags Register Details Between CRG and CPMU Modules

	CRG Module	CPMU Module
Register Name	CRGFLG	CPMUFLG
Register Address	0x0037	0x0037
Register State upon Reset	Dependent upon type of reset – see notes below register diagram	Dependent upon type of reset – see notes below register diagram
Bit Changes on CPMU Register from CRG Module	Bit 0 and 1. As there is no SCM on CPMU, these bits have been altered.	

The CPMU module does not have a self-clock mode (SCM), which forms part of an operational mode on the CRG module. For compatibility, this must be taken into account and dealt with appropriately.

On the CPMU, the OSCIF is used to indicate a bit change in UPOSC.

4.5 Interrupt Enable Register

0x0038

	7	6	5	4	3	2	1	0
R	RTIE	0	0	LOCKIE	0	0	OSCIE	0
W								
Reset	0	0	0	0	0	0	0	0

Figure 11. CPMU Interrupt Enable Register (CPMUINT)

	7	6	5	4	3	2	1	0
R	RTIE	0	0	LOCKIE	0	0	SCMIE	0
W								
Reset	0	0	0	0	0	0	0	0

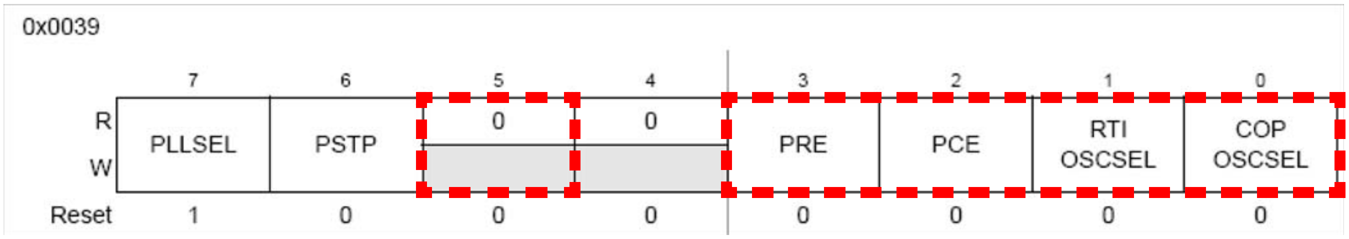
Figure 12. CRG Interrupt Enable Register (CRGINT)

Table 13. Interrupt Register Details between CRG and CPMU Modules

	CRG Module	CPMU Module
Register Name	CRGINT	CPMUINT
Register Address	0x0038	0x0038
Register State upon Reset	0x00	0x00
Bit Changes on CPMU Register from CRG Module	Bit 1. As there is no SCM on the CPMU, register bit 1 has been altered.	

This register enables interrupt requests specific to the module. In similar fashion to the flags register, bit 1 of the CPMUINT register differs from the CRG register due to the fact that there is no SCM on the CPMU. This bit is able to detect oscillator corruption and enable a flag (given the appropriate interrupt flag is activated). Bits 4 and 7 function in the same manner for both the CPMU and CRG registers.

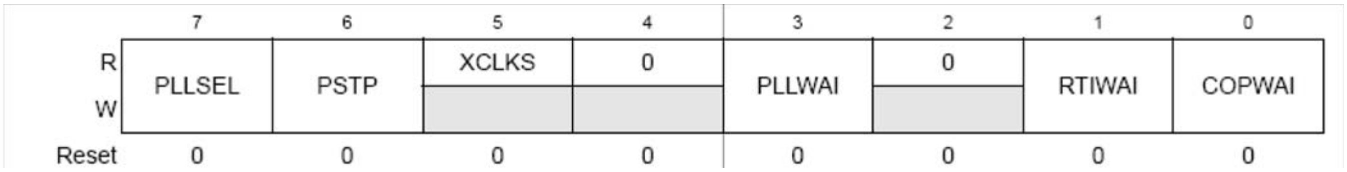
4.6 Clock Select Registers



0x0039

	7	6	5	4	3	2	1	0
R	PLLSEL	PSTP	0	0	PRE	PCE	RTI OSCSEL	COP OSCSEL
W	PLLSEL	PSTP			PRE	PCE	RTI OSCSEL	COP OSCSEL
Reset	1	0	0	0	0	0	0	0

Figure 13. CPMU Clock Select Register (CPMUCLKS)



	7	6	5	4	3	2	1	0
R	PLLSEL	PSTP	XCLKS	0	PLLWAI	0	RTIWAI	COPWAI
W	PLLSEL	PSTP			PLLWAI		RTIWAI	COPWAI
Reset	0	0	0	0	0	0	0	0

Figure 14. CRG Clock Select Register (CLKSEL)

Table 14. Clock Select Register Details Between CRG and CPMU modules

	CRG Module	CPMU Module
Register Name	CLKSEL	CPMUCLKS
Register Address	0x0039	0x0039
Register State upon Reset	0x00	0x80
Bit Changes on CPMU Register from CRG Module	Bits 0, 1, 2, 3, and 5. Significant changes due to inclusion of internal reference and difference on WAIT instruction within CPMU.	

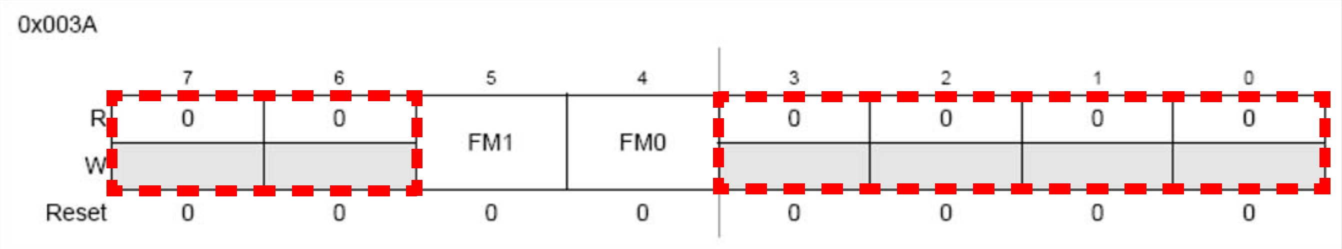
These registers are responsible for selecting the clock. Bits 6 and 7 maintain compatibility between the S12XS and S12P, because they have the same functions. When bit 7 is set the system clocks will be derived from the oscillator; when bit 7 is cleared the system clocks will be derived from the PLL. Bit 6 controls the oscillator in stop mode — setting it maintains the oscillator in stop/pseudo-stop mode, and clearing it disables the oscillator in stop mode.

The changes between the two registers are mainly due to the differences of the wait property from the CPMU module, plus the addition of the internal reference. On this module, wait mode has the same properties as run mode.

On the CRG module, in wait mode bits 3, 1, and 0 are used to adjust the properties of the PLL, RTI, and COP respectively.

On the CPMU module, bits 3 and 2 are used to configure the RTI and COP respectively while the module is running in pseudo-stop, and bits 1 and 0 are used to select either internal or external reference for the RTI and COP respectively.

4.7 PLL Control Register



0x003A

	7	6	5	4	3	2	1	0
R	0	0	FM1	FM0	0	0	0	0
W								
Reset	0	0	0	0	0	0	0	0

Figure 15. CPMU PLL Control Register (CPMUPLL)

	7	6	5	4	3	2	1	0
R W	CME	PLLON	FM1	FM0	FSTWKP	PRE	PCE	SCME
Reset	1	1	0	0	0	0	0	1

Figure 16. CRG IPLL Control Register (PLLCTL)

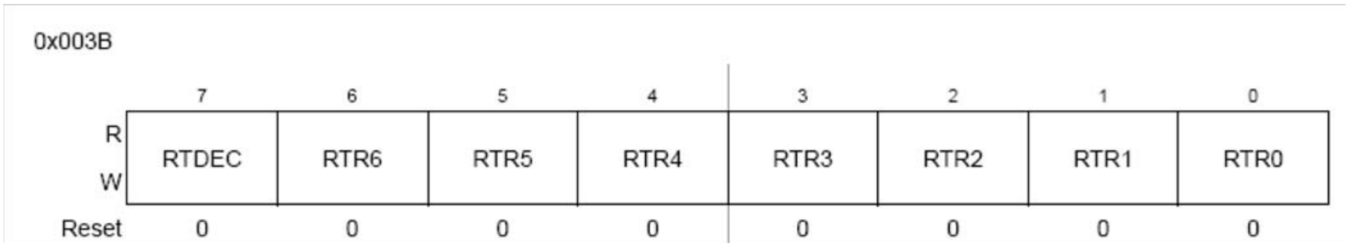
Table 15. PLL Control Register Details Between CRG and CPMU Modules

	CRG Module	**CPMU Module**
Register Name	PLLCTL	CPMUPLL
Register Address	0x003A	0x003A
Register State upon Reset	0xC1	0x00
Bit Changes on CPMU Register from CRG Module	Bits 7, 6, 3, 2, 1, and 0. Significant changes due to differing features from CRG.	

Bits 5 and 4 are compatible between the two registers and have the same functionality in that they are used as the PLL frequency modulation enable bits for VCOCLK. On the CRG module they are write anytime except when PLLSEL=1, but on the CPMU module they can be written to only when PROT=0 and PLLSEL=1.

Bits 7, 6, 3, 2, 1, and 0 are not used on the CPMU register. Bits 2 and 1 on the CRG register are respectively compatible with bits 3 and 2 on the CPMUCLKS register (see Section 4.6, “Clock Select Registers”). The remaining bits on the CRG do not have compatible functionality on the CPMU register, so they are excluded.

4.8 RTI Control Register



0x003B

	7	6	5	4	3	2	1	0
R W	RTDEC	RTR6	RTR5	RTR4	RTR3	RTR2	RTR1	RTR0
Reset	0	0	0	0	0	0	0	0

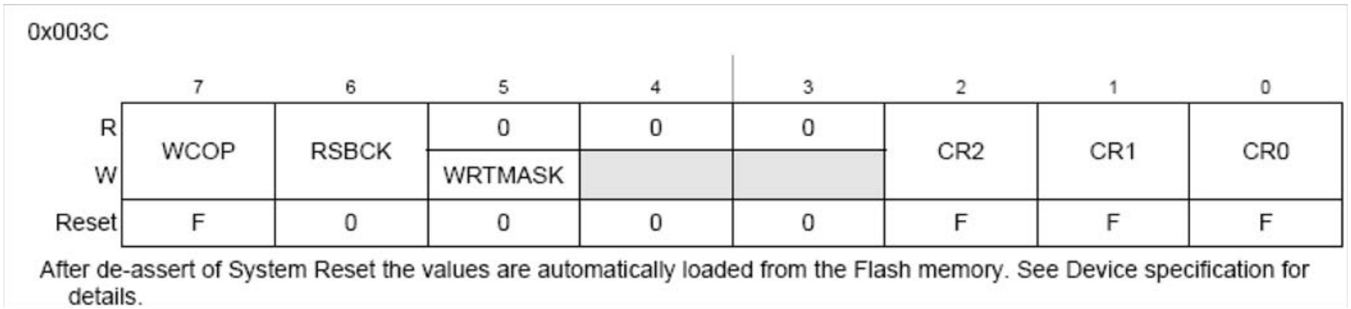
Figure 17. RTI Control Register

Table 16. RTI Control Register Details Between CRG and CPMU Modules

	CRG Module	**CPMU Module**
Register Name	RTICTL	CPMURTI
Register Address	0x003B	0x003B
Register State upon Reset	0x00	0x00
Bit Changes on CPMU Register from CRG Module	None	

These registers are compatible between the CRG and CPMU modules, and there are no changes between them except the register name.

4.9 COP Control Register



0x003C

	7	6	5	4	3	2	1	0
R	WCOP	RSBCK	0	0	0	CR2	CR1	CR0
W			WRTMASK					
Reset	F	0	0	0	0	F	F	F

After de-assert of System Reset the values are automatically loaded from the Flash memory. See Device specification for details.

Figure 18. COP Control Register

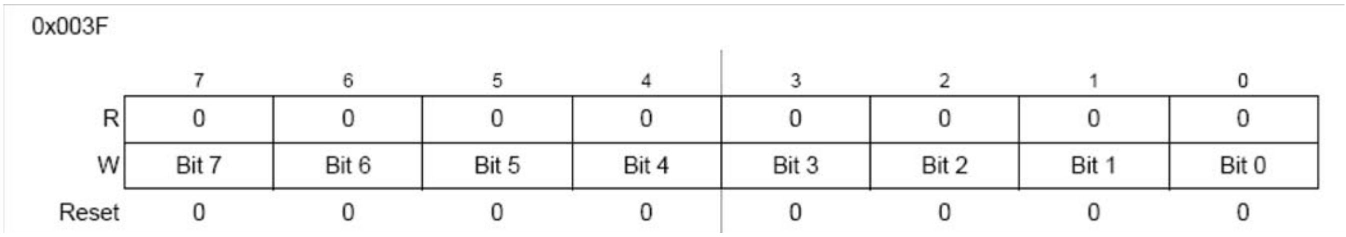
Table 17. COP Control Register Details Between CRG and CPMU Modules

	CRG Module	CPMU Module
Register Name	COPCTL	CPMUCOP
Register Address	0x003C	0x003C
Register State upon Reset	0x00	From flash
Bit Changes on CPMU Register from CRG Module	None	

These registers are responsible for controlling the COP watchdog. The COP control registers are similar, with the bits being compatible between the CPMU and CRG modules. However, the register names differ.

The major functionality difference with the CPMU module is the option to run the COP with either the internal or external reference, which is dependent upon the setting of COPOSCSEL (CPMUCLKS register).

4.10 COP Timer/Arm Reset Register



0x003F

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	0
W	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Reset	0	0	0	0	0	0	0	0

Figure 19. COP Timer Arm/Reset Register

Table 18. COP Timer/Arm Control Register Details Between CRG and CPMU Modules

	CRG Module	CPMU Module
Register Name	ARMCOP	CPMUARMCOP
Register Address	0x003F	0x003F
Register State upon Reset	0x00	0x00
Bit Changes on CPMU Register from CRG Module	None	

The ARMCOP and CPMUARMCOP registers are compatible, and there are no changes between them except the register name.

5 Conclusion

This application note has highlighted the functionality of the CPMU module, inclusive of the new clocking scheme, internal reference, API, and VREG. The main objective was to explain what it would take to alter an application, which previously used the CRG as a clock module, to use the CPMU as the source of system clocks. Furthermore, detailed analysis of the registers used to configure the CPMU was also considered in this application note, giving a side-by side comparison with equivalent CRG registers.

Appendix A CPMU Expected Behaviors

Table 19 outlines some scenarios in which the MCU might find itself, and provides an explanation of the expected behaviors from the CPMU.

Table 19. CPMU Plausible Scenarios and Actions

	CRG	CPMU
Switching from Internal to External/External to Internal Reference	Not applicable.	Directly done by altering OSCE bit. If going to an external reference also requires the UPOSC bit to be set, then switching will have occurred. Bus is on PLL for both cases. The REFDIV bits must be adjusted before the OSCE bit. When OSCE is set, the PLL will lose lock and the bus clock will switch to $f_{VCO}/4$ until the PLL has locked again.
Loss of Lock	A lock-interrupt flag is generated and is cleared when LOCKIF is set. User software should handle this situation.	If bus clock is derived from PLL, nothing happens (clock switches frequency to $f_{VCO}/4$ at least). The lock interrupt lets the user know that the device is not running with the target frequency.
Severe Oscillator Fault / Failure	Clock monitor fail event occurs and places part in SCM (note various bits must be set to allow this operation). A clock quality check may also be performed.	Clock monitor will perform a reset and the device will restart in PEI mode.
Oscillator Noise	Clock monitor fail event occurs and places part in SCM (note various bits must be set to allow this operation). A clock quality check may also be performed.	Will automatically recover. Occasional spikes are filtered; OSCFILT[4:0] bits should be configured appropriately. If UPOSC is lost, bus clock is switched back to PLL immediately — in other words, PBE would switch back to PEE and PEE would switch back to PEI.

THIS PAGE IS INTENTIONALLY BLANK

How to Reach Us:

Home Page:
www.freescale.com

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

USA/Europe or Locations Not Listed:
Freescale Semiconductor, Inc.
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 Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including “Typicals”, must be validated for each customer application by customer’s technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

RoHS-compliant and/or Pb-free versions of Freescale products have the functionality and electrical characteristics as their non-RoHS-compliant and/or non-Pb-free counterparts. For further information, see http://www.freescale.com or contact your Freescale sales representative.

For information on Freescale’s Environmental Products program, go to http://www.freescale.com/epp.