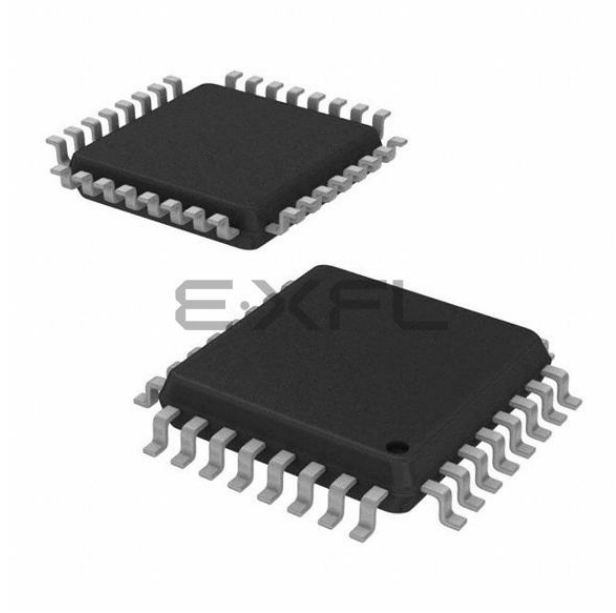

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	12V1
Core Size	16-Bit
Speed	25MHz
Connectivity	IrDA, LINbus, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	16
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	128 x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	3.13V ~ 5.5V
Data Converters	A/D 2x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LQFP
Supplier Device Package	32-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/s9s12vr16f0mlcr

Table 2-7. ECLKCTL Register Field Descriptions

Field	Description
7 NECLK	**No ECLK** — Disable ECLK output This bit controls the availability of a free-running clock on the ECLK pin. This clock has a fixed rate equivalent to the internal bus clock. 1 ECLK disabled 0 ECLK enabled

2.3.7 PIM Miscellaneous Register (PIMMISC)

Address 0x001D — Access: User read/write[1]

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

Figure 2-5. PIM Miscellaneous Register (PIMMISC)

[1] Read: Anytime
Write: Anytime

Table 2-8. PIMMISC Register Field Descriptions

Field	Description
7 OCPE	**Over-Current Protection Enable**— Activate over-current detector on PP2 Refer to Section 2.5.3, “Over-Current Protection on EVDD” 1 PP2 over-current detector enabled 0 PP2 over-current detector disabled

2.3.8 IRQ Control Register (IRQCR)[1]

Address 0x001E (S12VR64/48) — Access: User read/write[1]

	7	6	5	4	3	2	1	0
R	IRQE	IRQEN	0	0	0	0	0	0
W								
Reset	0	0	0	0	0	0	0	0

Figure 2-6. IRQ Control Register (IRQCR)

[1] Read: Anytime
Write:
IRQE: Once in normal mode, anytime in special mode
IRQEN: Anytime

1. This register is applicable for S12VR64 and S12VR48 in 48 pin package

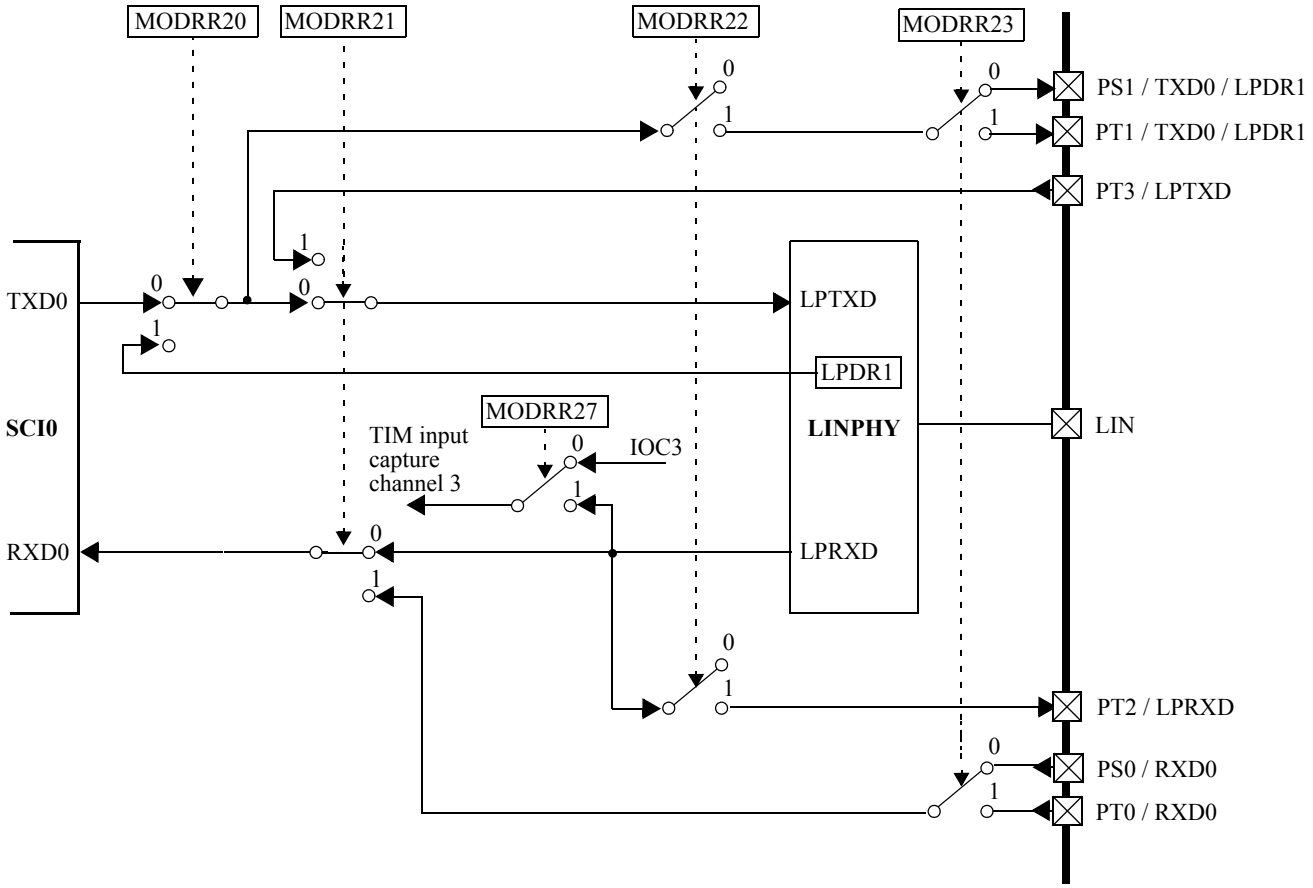

Figure 2-32. SCI0-to-LINPHY Routing Options Illustration

Table 2-24. Preferred Interface Configurations

MODRR2[3:0]	Signal Routing	Description
0000	TXD0 → LPTXD RXD0 ← LPRXD	Default setting: SCI0 connects to LINPHY, interface internal only
0001	LPDR1 → LPTXD RXD0 ← LPRXD	Direct control setting: LPDR[LPDR1] register bit controls LPTXD, interface internal only

NOTE

When enabling the resistor paths to ground by setting PTAL[PTAENL]=1 or by changing PTAL[PTAL1:PTAL0], a settling time of t_{UNC_HVI} + two bus cycles must be considered to let internal nodes be loaded with correct values.

Table 2-36. HVI pin connected to ADC channel

PTAL[PTAL1]	PTAL[PTAL0]	HVI pin connected to ADC[1]
0	0	HVI0
0	1	HVI1
1	0	HVI2
1	1	HVI3

1 Refer to device overview section for channel assignment

2.3.46 Port AD Interrupt Enable Register (PIE1AD)

Address 0x027D (S12VR64/48) Access: User read/write[1]

	7	6	5	4	3	2	1	0
R	0	0	PIE1AD5	PIE1AD4	PIE1AD3	PIE1AD2	PIE1AD1	PIE1AD0
W								
Reset	0	0	0	0	0	0	0	0

Figure 2-66. Port AD Interrupt Enable Register (PIE1AD - S12VR64/48)

1 Read: Anytime
Write: Anytime

Read: Anytime.

Address 0x027D (S12VR32/16) Access: User read/write[1]

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	PIE1AD1	PIE1AD0
W								
Reset	0	0	0	0	0	0	0	0

Figure 2-67. Port AD Interrupt Enable Register (PIE1AD - S12VR32/16)

1 Read: Anytime
Write: Anytime

Table 2-46. PIE1AD Register Field Descriptions

Field	Description
5-0 PIE1AD	**Pin Interrupt Enable register 1 port AD —** This bit enables or disables the edge sensitive pin interrupt on the associated pin. An interrupt can be generated if the pin is operating in input or output mode when in use with the general-purpose or related peripheral function. For wakeup from stop mode this bit must be set to allow activating the RC oscillator. 1 Interrupt is enabled 0 Interrupt is disabled (interrupt flag masked)

Table 3-5. DIRECT Field Descriptions

Field	Description
7–0 DP[15:8]	**Direct Page Index Bits 15–8** — These bits are used by the CPU when performing accesses using the direct addressing mode. These register bits form bits [15:8] of the local address (see Figure 3-6).

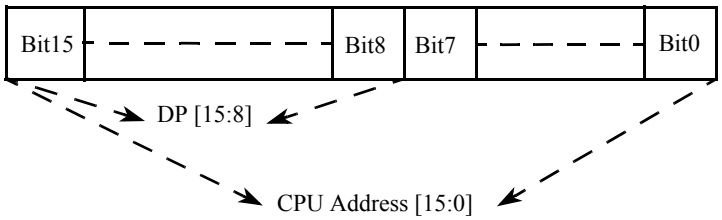

Figure 3-6. DIRECT Address Mapping

Example 3-1. This example demonstrates usage of the Direct Addressing Mode

```
MOVB     #$04,DIRECT      ;Set DIRECT register to 0x04. From this point on, all memory
                          ;accesses using direct addressing mode will be in the local
                          ;address range from 0x0400 to 0x04FF.
LDY      <$12             ;Load the Y index register from 0x0412 (direct access).
```

3.3.2.3 MMC Control Register (MMCCTL1)

Address: 0x0013

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

= Unimplemented or Reserved

Figure 3-7. MMC Control Register (MMCCTL1)

Read: Anytime.

Write: Anytime.

The NVMRES bit maps 16k of internal NVM resources (see Section FTMRG) to the global address space 0x04000 to 0x07FFF.

Table 3-6. MODE Field Descriptions

Field	Description
0 NVMRES	Map internal NVM resources into the global memory map Write: Anytime This bit maps internal NVM resources into the global address space. 0 Program flash is mapped to the global address range from 0x04000 to 0x07FFF. 1 NVM resources are mapped to the global address range from 0x04000 to 0x07FFF.

4.1.3 S12CPMU_UHV_V8 Block Diagram

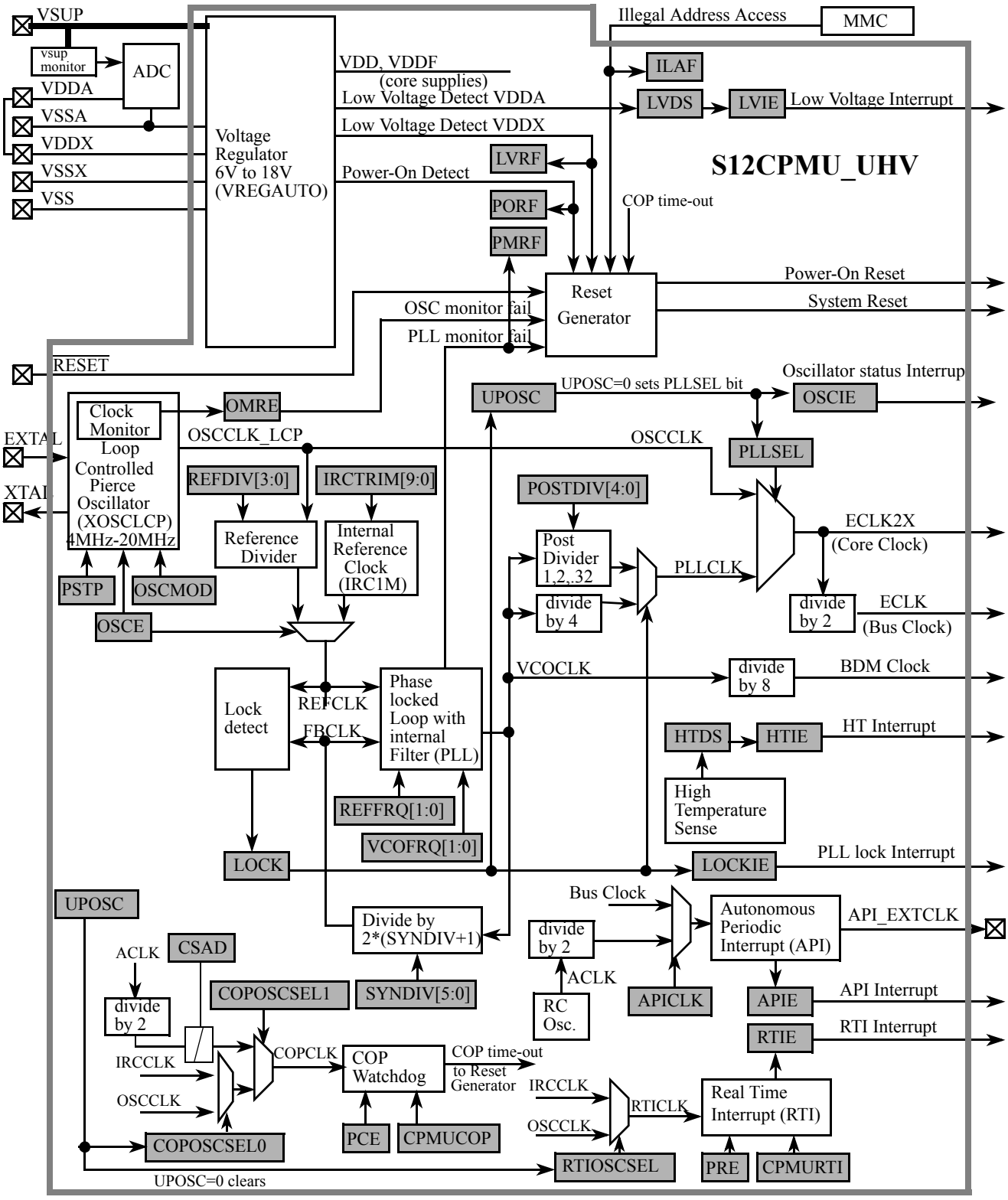

Figure 4-1. Block diagram of S12CPMU_UHV_V8

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x02F3	CPMUACLKTR	R	ACLKTR5	ACLKTR4	ACLKTR3	ACLKTR2	ACLKTR1	ACLKTR0	0	0
		W								
0x02F4	CPMUAPIRH	R	APIR15	APIR14	APIR13	APIR12	APIR11	APIR10	APIR9	APIR8
		W								
0x02F5	CPMUAPIRL	R	APIR7	APIR6	APIR5	APIR4	APIR3	APIR2	APIR1	APIR0
		W								
0x02F6	RESERVED CPMUTEST3	R	0	0	0	0	0	0	0	0
		W								
0x02F7	CPMUHTTR	R	HTOE	0	0	0	HTTR3	HTTR2	HTTR1	HTTR0
		W								
0x02F8	CPMU IRCTRIMH	R	TCTRIM[4:0]					0	IRCTRIM[9:8]	
		W								
0x02F9	CPMU IRCTRIML	R	IRCTRIM[7:0]							
		W								
0x02FA	CPMUOSC	R	OSCE	0	0	0	0	0	0	0
		W								
0x02FB	CPMUPROT	R	0	0	0	0	0	0	0	PROT
		W								
0x02FC	RESERVED CPMUTEST2	R	0	0	0	0	0	0	0	0
		W								
0x02FD	RESERVED	R	0	0	0	0	0	0	0	0
		W								
0x02FE	CPMUOSC2	R	0	0	0	0	0	0	**OMRE**	**OSCMOD**
		W								

= Unimplemented or Reserved

Figure 4-3. CPMU Register Summary[1]

[1] Registers in **bold underlined** are only available on S12VR32/16. On S12VR64/48 these locations read 0 and write is not implemented.

Table 4-4. CPMUFLG Field Descriptions

Field	Description
7 RTIF	**Real Time Interrupt Flag** — RTIF is set to 1 at the end of the RTI period. This flag can only be cleared by writing a 1. Writing a 0 has no effect. If enabled (RTIE=1), RTIF causes an interrupt request. 0 RTI time-out has not yet occurred. 1 RTI time-out has occurred.
6 PORF	**Power on Reset Flag** — PORF is set to 1 when a power on reset occurs. This flag can only be cleared by writing a 1. Writing a 0 has no effect. 0 Power on reset has not occurred. 1 Power on reset has occurred.
5 LVRF	**Low Voltage Reset Flag** — LVRF is set to 1 when a low voltage reset occurs. This flag can only be cleared by writing a 1. Writing a 0 has no effect. 0 Low voltage reset has not occurred. 1 Low voltage reset has occurred.
4 LOCKIF	**PLL Lock Interrupt Flag** — LOCKIF is set to 1 when LOCK status bit changes. This flag can only be cleared by writing a 1. Writing a 0 has no effect.If enabled (LOCKIE=1), LOCKIF causes an interrupt request. 0 No change in LOCK bit. 1 LOCK bit has changed.
3 LOCK	**Lock Status Bit** — LOCK reflects the current state of PLL lock condition. Writes have no effect.While PLL is unlocked (LOCK=0) f_{PLL} is f_{VCO} / 4 to protect the system from high core clock frequencies during the PLL stabilization time tlock. 0 VCOCLK is not within the desired tolerance of the target frequency. $f_{PLL} = f_{VCO}/4$. 1 VCOCLK is within the desired tolerance of the target frequency. $f_{PLL} = f_{VCO}/(POSTDIV+1)$.
2 ILAF	**Illegal Address Reset Flag** — ILAF is set to 1 when an illegal address reset occurs.Refer to MMC chapter for details.This flag can only be cleared by writing a 1. Writing a 0 has no effect. 0 Illegal address reset has not occurred. 1 Illegal address reset has occurred.
1 OSCIF	**Oscillator Interrupt Flag** — OSCIF is set to 1 when UPOSC status bit changes. This flag can only be cleared by writing a 1. Writing a 0 has no effect.If enabled (OSCIE=1), OSCIF causes an interrupt request. 0 No change in UPOSC bit. 1 UPOSC bit has changed.
0 UPOSC	**Oscillator Status Bit** — UPOSC reflects the status of the oscillator. Writes have no effect. Entering Full Stop Mode UPOSC is cleared. 0 The oscillator is off or oscillation is not qualified by the PLL. 1 The oscillator is qualified by the PLL.

Table 4-6. CPMUCLKS Descriptions

Field	Description
7 PLLSEL	**PLL Select Bit** This bit selects the PLLCLK as source of the System Clocks (Core Clock and Bus Clock). PLLSEL can only be set to 0, if UPOSC=1. UPOSC= 0 sets the PLLSEL bit. Entering Full Stop Mode sets the PLLSEL bit. 0 System clocks are derived from OSCCLK if oscillator is up (UPOSC=1, $f_{bus} = f_{osc} / 2$). 1 System clocks are derived from PLLCLK, $f_{bus} = f_{PLL} / 2$.
6 PSTP	**Pseudo Stop Bit** This bit controls the functionality of the oscillator during Stop Mode. 0 Oscillator is disabled in Stop Mode (Full Stop Mode). 1 Oscillator continues to run in Stop Mode (Pseudo Stop Mode), option to run RTI and COP. **Note:** Pseudo Stop Mode allows for faster STOP recovery and reduces the mechanical stress and aging of the resonator in case of frequent STOP conditions at the expense of a slightly increased power consumption. **Note:** When starting up the external oscillator (either by programming OSCE bit to 1 or on exit from Full Stop Mode with OSCE bit already 1) the software must wait for a minimum time equivalent to the startup-time of the external oscillator t_{UPOSC} before entering Pseudo Stop Mode.
5 CSAD	**COP in Stop Mode ACLK Disable** — If this bit is set the ACLK for the COP in Stop Mode is disabled. Hence the COP is static while in Stop Mode and continues to operate after exit from Stop Mode. **For CSAD = 1 and COP is running on ACLK (COPOSCSEL1 = 1) the following applies:** Due to clock domain crossing synchronization there is a latency time of 2 ACLK cycles to enter Stop Mode. **After exit from STOP mode (when interrupt service routine is entered) the software has to wait for 2 ACLK cycles before it is allowed to enter Stop mode again (STOP instruction). It is absolutely forbidden to enter Stop Mode before this time of 2 ACLK cycles has elapsed.** 0 COP running in Stop Mode (ACLK for COP enabled in Stop Mode). 1 COP stopped in Stop Mode (ACLK for COP disabled in Stop Mode)
4 COP OSCSEL1	**COP Clock Select 1** — COPOSCSEL0 and COPOSCSEL1 combined determine the clock source to the COP (see also Table 4-7). If COPOSCSEL1 = 1, COPOSCSEL0 has no effect regarding clock select and changing the COPOSCSEL0 bit does not re-start the COP time-out period. COPOSCSEL1 selects the clock source to the COP to be either ACLK (derived from trimmable internal RC-Oscillator) or clock selected via COPOSCSEL0 (IRCCLK or OSCCLK). Changing the COPOSCSEL1 bit re-starts the COP time-out period. COPOSCSEL1 can be set independent from value of UPOSC. UPOSC= 0 does not clear the COPOSCSEL1 bit. 0 COP clock source defined by COPOSCSEL0 1 COP clock source is ACLK derived from a trimmable internal RC-Oscillator
3 PRE	**RTI Enable During Pseudo Stop Bit** — PRE enables the RTI during Pseudo Stop Mode. 0 RTI stops running during Pseudo Stop Mode. 1 RTI continues running during Pseudo Stop Mode if RTIOSCSEL=1. **Note:** If PRE=0 or RTIOSCSEL=0 then the RTI will go static while Stop Mode is active. The RTI counter will not be reset.
2 PCE	**COP Enable During Pseudo Stop Bit** — PCE enables the COP during Pseudo Stop Mode. 0 COP stops running during Pseudo Stop Mode 1 COP continues running during Pseudo Stop Mode if COPOSCSEL=1 **Note:** If PCE=0 or COPOSCSEL=0 then the COP will go static while Stop Mode is active. The COP counter will not be reset.

Table 4-15. COP Watchdog Rates if COPOSCSEL1=1.

CR2	CR1	CR0	COPCLK Cycles to time-out (COPCLK is ACLK divided by 2)
0	0	0	COP disabled
0	0	1	2^{7}
0	1	0	2^{9}
0	1	1	2^{11}
1	0	0	2^{13}
1	0	1	2^{15}
1	1	0	2^{16}
1	1	1	2^{17}

4.3.2.22 S12CPMU_UHV_V8 Protection Register (CPMUPROT)

This register protects the clock configuration registers from accidental overwrite:

CPMUSYNR, CPMUREFDIV, CPMUCLKS, CPMUPLL, CPMUIRCTRIMH/L, CPMUOSC and CPMUOSC2

0x02FB

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

Figure 4-31. S12CPMU_UHV_V8 Protection Register (CPMUPROT)

Read: Anytime

Write: Anytime

Field	Description
PROT	**Clock Configuration Registers Protection Bit** — This bit protects the clock configuration registers from accidental overwrite (see list of protected registers above): Writing 0x26 to the CPMUPROT register clears the PROT bit, other write accesses set the PROT bit. 0 Protection of clock configuration registers is disabled. 1 Protection of clock configuration registers is enabled. (see list of protected registers above).

4.4 Functional Description

4.4.1 Phase Locked Loop with Internal Filter (PLL)

The PLL is used to generate a high speed PLLCLK based on a low frequency REFCLK.

The REFCLK is by default the IRCCLK which is trimmed to f_{IRC1M_TRIM}=1MHz.

If using the oscillator (OSCE=1) REFCLK will be based on OSCCLK. For increased flexibility, OSCCLK can be divided in a range of 1 to 16 to generate the reference frequency REFCLK using the REFDIV[3:0] bits. Based on the SYNDIV[5:0] bits the PLL generates the VCOCLK by multiplying the reference clock by a 2, 4, 6,... 126, 128. Based on the POSTDIV[4:0] bits the VCOCLK can be divided in a range of 1,2, 3, 4, 5, 6,... to 32 to generate the PLLCLK.

If oscillator is enabled (OSCE=1) $$f_{REF} = \frac{f_{OSC}}{(REFDIV + 1)}$$

If oscillator is disabled (OSCE=0) $$f_{REF} = f_{IRC1M}$$

$$f_{VCO} = 2 \times f_{REF} \times (SYNDIV + 1)$$

If PLL is locked (LOCK=1) $$f_{PLL} = \frac{f_{VCO}}{(POSTDIV + 1)}$$

If PLL is not locked (LOCK=0) $$f_{PLL} = \frac{f_{VCO}}{4}$$

If PLL is selected (PLLSEL=1) $$f_{bus} = \frac{f_{PLL}}{2}$$

NOTE

Although it is possible to set the dividers to command a very high clock frequency, do not exceed the specified bus frequency limit for the MCU.

Address: 0x0021

	7	6	5	4	3	2	1	0
R	TBF	0	0	0	0	SSF2	SSF1	SSF0
W								
Reset	—	0	0	0	0	0	0	0
POR	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 6-4. Debug Status Register (DBGSR)

Read: Anytime

Write: Never

Table 6-5. DBGSR Field Descriptions

Field	Description
7 TBF	**Trace Buffer Full** — The TBF bit indicates that the trace buffer has stored 64 or more lines of data since it was last armed. If this bit is set, then all 64 lines will be valid data, regardless of the value of DBGCNT bits. The TBF bit is cleared when ARM in DBGC1 is written to a one. The TBF is cleared by the power on reset initialization. Other system generated resets have no affect on this bit This bit is also visible at DBGCNT[7]
2–0 SSF[2:0]	**State Sequencer Flag Bits** — The SSF bits indicate in which state the State Sequencer is currently in. During a debug session on each transition to a new state these bits are updated. If the debug session is ended by software clearing the ARM bit, then these bits retain their value to reflect the last state of the state sequencer before disarming. If a debug session is ended by an internal event, then the state sequencer returns to state0 and these bits are cleared to indicate that state0 was entered during the session. On arming the module the state sequencer enters state1 and these bits are forced to SSF[2:0] = 001. See Table 6-6**.**

Table 6-6. SSF[2:0] — State Sequence Flag Bit Encoding

SSF[2:0]	Current State
000	State0 (disarmed)
001	State1
010	State2
011	State3
100	Final State
101,110,111	Reserved

6.3.2.7.4 Debug Match Flag Register (DBGMFR)

Address: 0x0027

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	MC2	MC1	MC0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 6-12. Debug Match Flag Register (DBGMFR)

Read: If COMRV[1:0] = 11

Write: Never

DBGMFR is visible at 0x0027 only with COMRV[1:0] = 11. It features 3 flag bits each mapped directly to a channel. Should a match occur on the channel during the debug session, then the corresponding flag is set and remains set until the next time the module is armed by writing to the ARM bit. Thus the contents are retained after a debug session for evaluation purposes. These flags cannot be cleared by software, they are cleared only when arming the module. A set flag does not inhibit the setting of other flags. Once a flag is set, further comparator matches on the same channel in the same session have no affect on that flag.

6.3.2.8 Comparator Register Descriptions

Each comparator has a bank of registers that are visible through an 8-byte window in the DBG module register address map. Comparator A consists of 8 register bytes (3 address bus compare registers, two data bus compare registers, two data bus mask registers and a control register). Comparator B consists of four register bytes (three address bus compare registers and a control register). Comparator C consists of four register bytes (three address bus compare registers and a control register).

Each set of comparator registers can be accessed using the COMRV bits in the DBGC1 register. Unimplemented registers (e.g. Comparator B data bus and data bus masking) read as zero and cannot be written. The control register for comparator B differs from those of comparators A and C.

Table 6-21. Comparator Register Layout

0x0028	CONTROL	Read/Write	Comparators A,B and C
0x0029	ADDRESS HIGH	Read/Write	Comparators A,B and C
0x002A	ADDRESS MEDIUM	Read/Write	Comparators A,B and C
0x002B	ADDRESS LOW	Read/Write	Comparators A,B and C
0x002C	DATA HIGH COMPARATOR	Read/Write	Comparator A only
0x002D	DATA LOW COMPARATOR	Read/Write	Comparator A only
0x002E	DATA HIGH MASK	Read/Write	Comparator A only
0x002F	DATA LOW MASK	Read/Write	Comparator A only

6.3.2.8.1 Debug Comparator Control Register (DBGXCTL)

The contents of this register bits 7 and 6 differ depending upon which comparator registers are visible in the 8-byte window of the DBG module register address map.

8.3.2.12.2 Right Justified Result Data (DJM=1)

Module Base +
0x0010 = ATDDR0, 0x0012 = ATDDR1, 0x0014 = ATDDR2, 0x0016 = ATDDR3
0x0018 = ATDDR4, 0x001A = ATDDR5

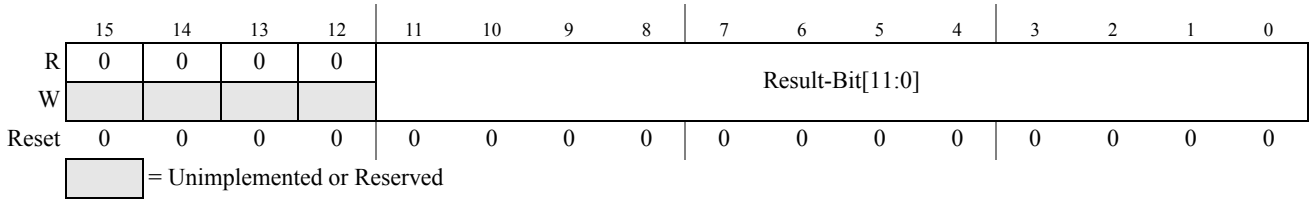

Figure 8-15. Right justified ATD conversion result register (ATDDR*n*)

Table 8-22 shows how depending on the A/D resolution the conversion result is transferred to the ATD result registers for right justified data. Compare is always done using all 12 bits of both the conversion result and the compare value in ATDDRn.

Table 8-22. Conversion result mapping to ATDDRn

A/D resolution	DJM	conversion result mapping to ATDDR*n*
8-bit data	1	Result-Bit[7:0] = result, Result-Bit[11:8]=0000
10-bit data	1	Result-Bit[9:0] = result, Result-Bit[11:10]=00

10.3.2.4 SCI Alternative Control Register 1 (SCIACR1)

	7	6	5	4	3	2	1	0
R	RXEDGIE	0	0	0	0	0	BERRIE	BKDIE
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 10-7. SCI Alternative Control Register 1 (SCIACR1)

Read: Anytime, if AMAP = 1

Write: Anytime, if AMAP = 1

Table 10-6. SCIACR1 Field Descriptions

Field	Description
7 RXEDGIE	**Receive Input Active Edge Interrupt Enable** — RXEDGIE enables the receive input active edge interrupt flag, RXEDGIF, to generate interrupt requests. 0 RXEDGIF interrupt requests disabled 1 RXEDGIF interrupt requests enabled
1 BERRIE	**Bit Error Interrupt Enable** — BERRIE enables the bit error interrupt flag, BERRIF, to generate interrupt requests. 0 BERRIF interrupt requests disabled 1 BERRIF interrupt requests enabled
0 BKDIE	**Break Detect Interrupt Enable** — BKDIE enables the break detect interrupt flag, BKDIF, to generate interrupt requests. 0 BKDIF interrupt requests disabled 1 BKDIF interrupt requests enabled

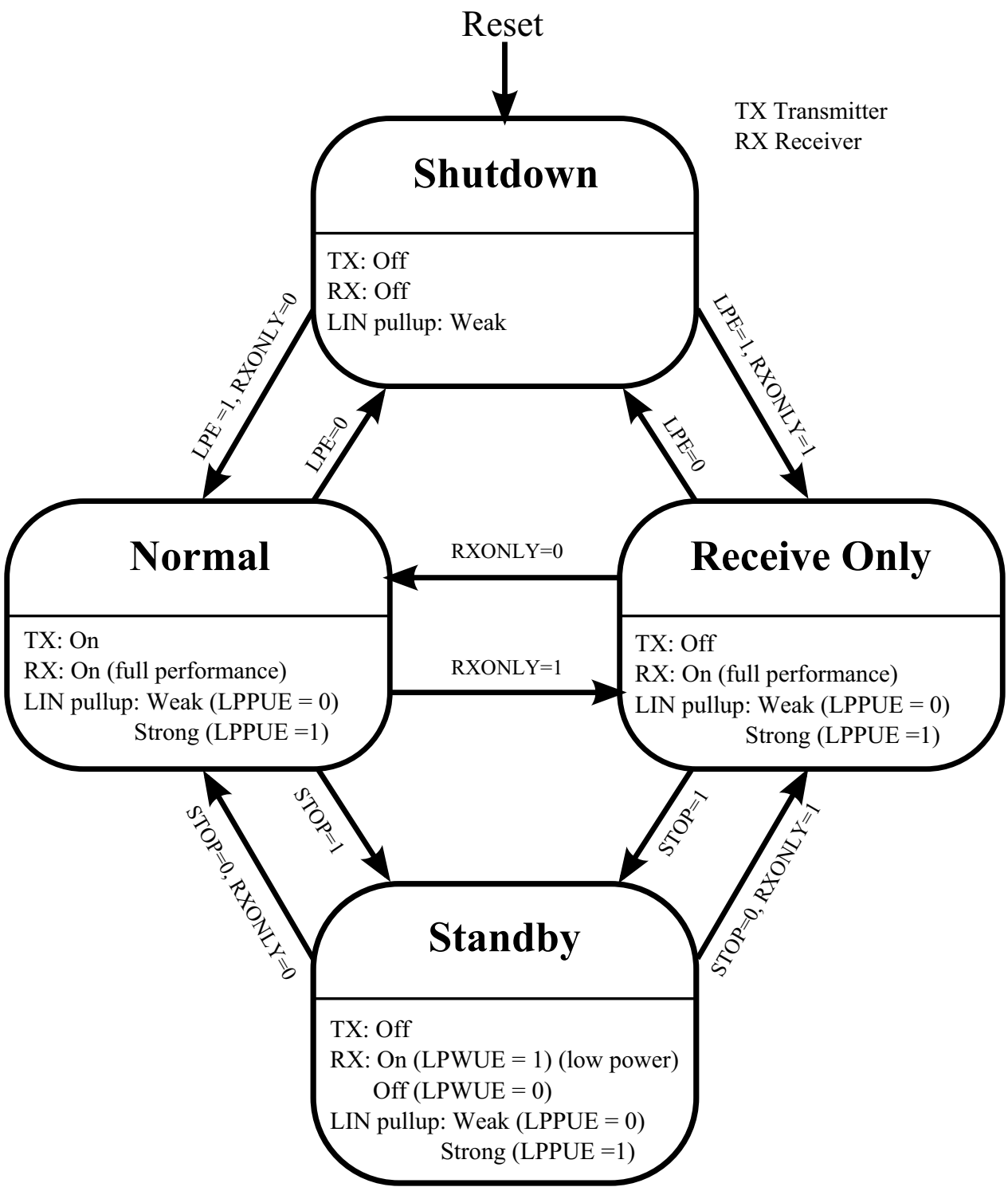

Figure 16-11. LIN Physical Layer Mode Transitions

18.1.3 Block Diagram

The block diagram of the Flash module is shown in Figure 18-1.

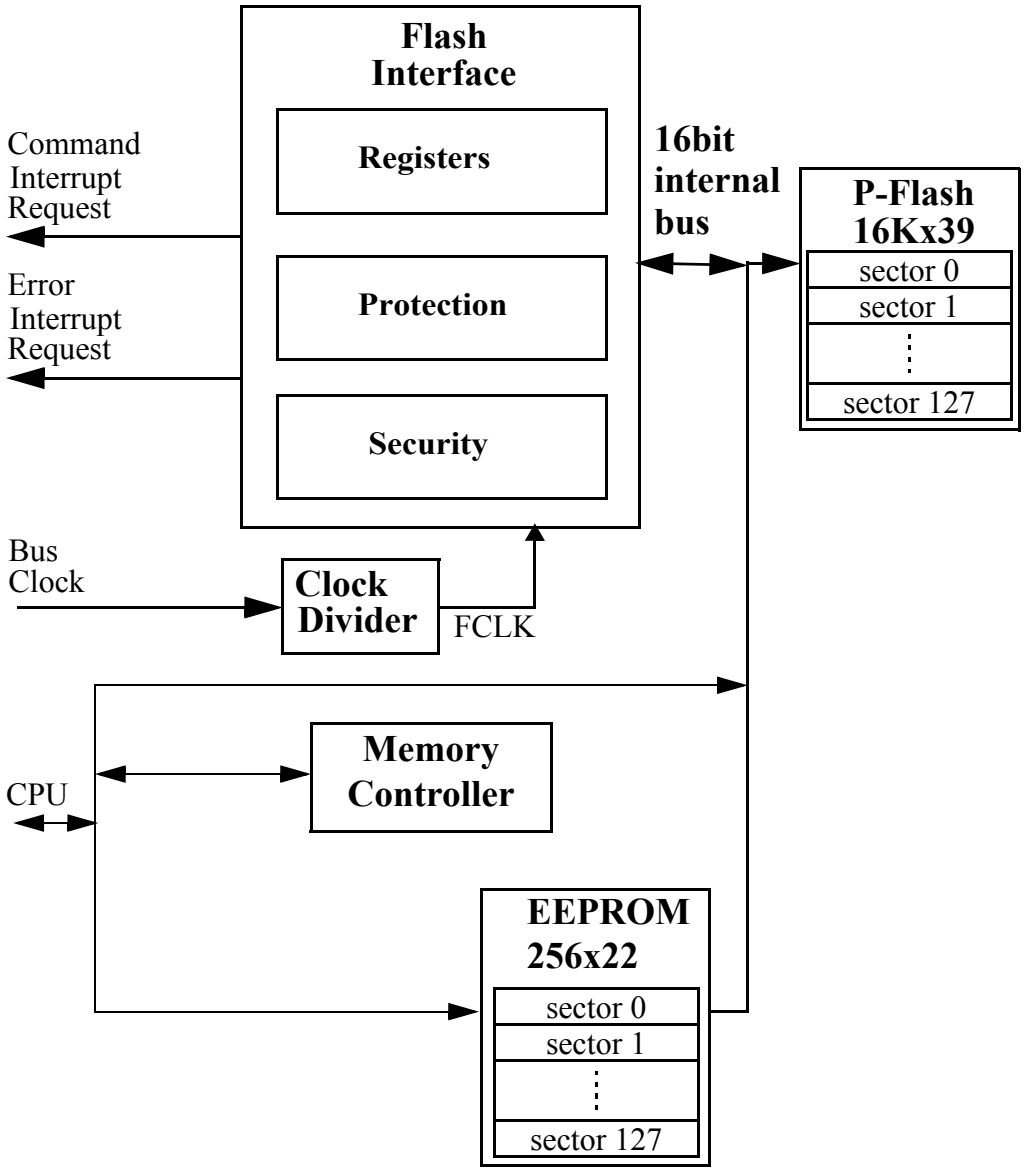

Figure 18-1. FTMRG64K512 Block Diagram

18.2 External Signal Description

The Flash module contains no signals that connect off-chip.

Table A-11. Peripheral Configurations for Run & Wait Current Measurement

Peripheral	Configuration
SCI	continuously transmit data (0x55) at speed of 19200 baud
SPI	configured to master mode, continuously transmit data (0x55) at 1Mbit/s
PWM	configured to toggle its pins at the rate of 40kHz
ADC	the peripheral is configured to operate at its maximum specified frequency and to continuously convert voltages on all input channels in sequence.
DBG	the module is enabled and the comparators are configured to trigger in outside range.The range covers all the code executed by the core.
TIM	the peripheral is configured to output compare mode, pulse accumulator and modulus counter enabled.
COP & RTI	enabled
HSDRV 1 & 2	module is enabled but output driver disabled
LSDRV 1 & 2	module is enabled but output driver disabled
BATS	enabled
LINPHY	connected to SCI and continuously transmit data (0x55) at speed of 19200 baud

Table A-12. Run and Wait Current Characteristics

Conditions are: V_{SUP}=V_{SUPHS}=18V, T_A=105°C, see Table A-10 and Table A-9						
Num	**Rating**	**Symbol**	**Min**	**Typ**	**Max**	**Unit**
1	Run Current	I_{SUPR}		15	22	mA
2	Wait Current	I_{SUPW}		10	15	mA