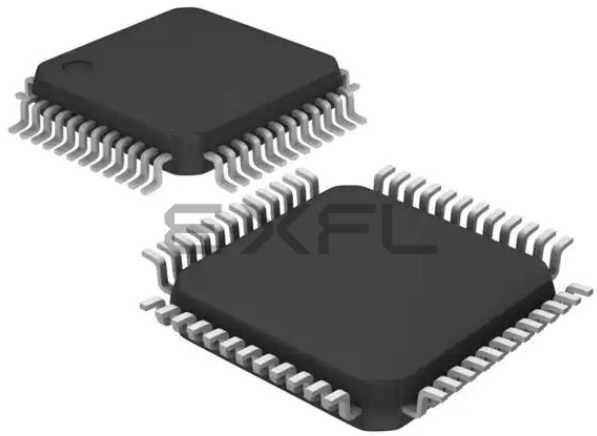

Welcome to E-XFL.COM

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	CANbus, IrDA, LINbus, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	40
Program Memory Size	240KB (240K x 8)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	11K x 8
Voltage - Supply (Vcc/Vdd)	3.13V ~ 5.5V
Data Converters	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-LQFP
Supplier Device Package	48-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/s9s12g240f0clf

1.8.2 S12GNA16 and S12GNA32

1.8.2.1 Pinout 48-Pin LQFP/QFN

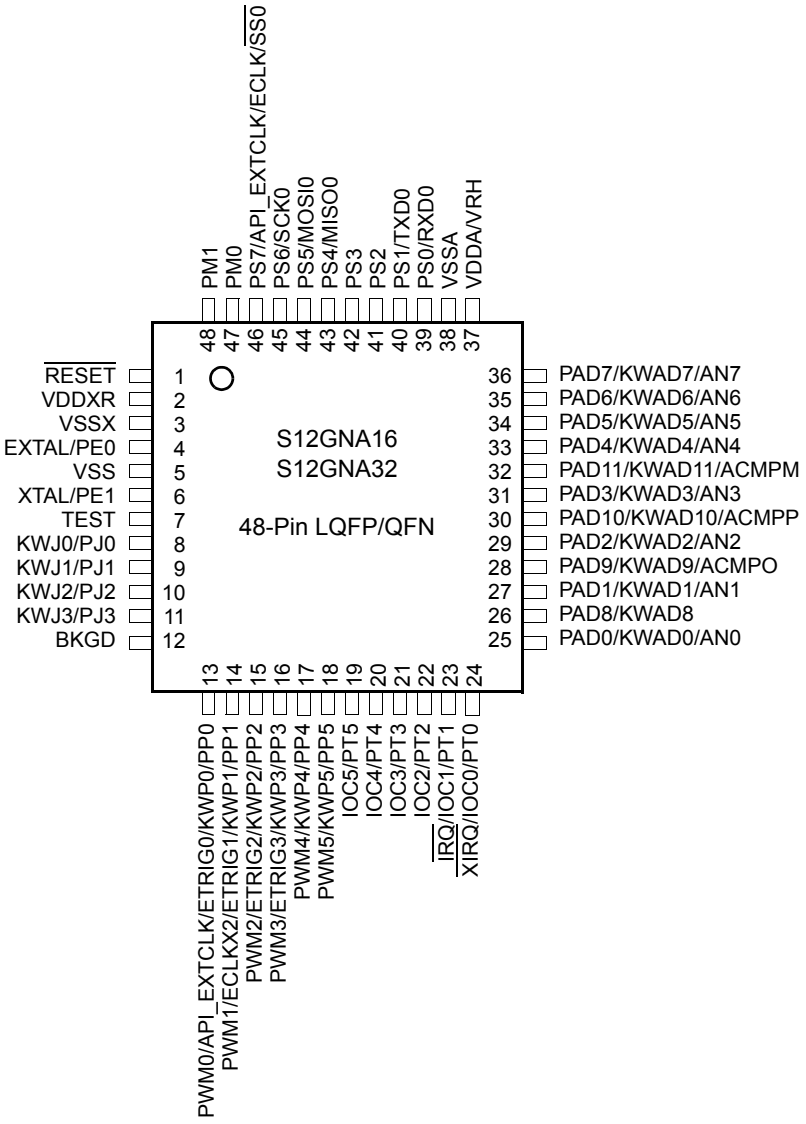

Figure 1-6. 48-Pin LQFP/QFN Pinout for S12GNA16 and S12GNA32

Table 1-11. 48-Pin LQFP/QFN Pinout for S12GNA16 and S12GNA32

	Function <----lowest-----PRIORITY-----highest---->					Power Supply	Internal Pull Resistor	
Package Pin	Pin	2nd Func.	3rd Func.	4th Func	5th Func		CTRL	Reset State
1	RESET	—	—	—	—	V_{DDX}	PULLUP	

Table 1-22. 100-Pin LQFP Pinout for S12G96 and S12G128

	Function <----lowest-----PRIORITY-----highest---->				Power Supply	Internal Pull Resistor	
Package Pin	Pin	2nd Func.	3rd Func.	4th Func.		CTRL	Reset State
28	PB3	—	—	—	V_{DDX}	PUCR/PUPBE	Disabled
29	PP0	KWP0	ETRIG0	PWM0	V_{DDX}	PERP/PPSP	Disabled
30	PP1	KWP1	ETRIG1	PWM1	V_{DDX}	PERP/PPSP	Disabled
31	PP2	KWP2	ETRIG2	PWM2	V_{DDX}	PERP/PPSP	Disabled
32	PP3	KWP3	ETRIG3	PWM3	V_{DDX}	PERP/PPSP	Disabled
33	PP4	KWP4	PWM4	—	V_{DDX}	PERP/PPSP	Disabled
34	PP5	KWP5	PWM5	—	V_{DDX}	PERP/PPSP	Disabled
35	PP6	KWP6	PWM6	—	V_{DDX}	PERP/PPSP	Disabled
36	PP7	KWP7	PWM7	—	V_{DDX}	PERP/PPSP	Disabled
37	VDDX3	—	—	—	—	—	—
38	VSSX3	—	—	—	—	—	—
39	PT7	IOC7	—	—	V_{DDX}	PERT/PPST	Disabled
40	PT6	IOC6	—	—	V_{DDX}	PERT/PPST	Disabled
41	PT5	IOC5	—	—	V_{DDX}	PERT/PPST	Disabled
42	PT4	IOC4	—	—	V_{DDX}	PERT/PPST	Disabled
43	PT3	IOC3	—	—	V_{DDX}	PERT/PPST	Disabled
44	PT2	IOC2	—	—	V_{DDX}	PERT/PPST	Disabled
45	PT1	IOC1	—	—	V_{DDX}	PERT/PPST	Disabled
46	PT0	IOC0	—	—	V_{DDX}	PERT/PPST	Disabled
47	PB4	$\overline{\text{IRQ}}$	—	—	V_{DDX}	PUCR/PUPBE	Disabled
48	PB5	$\overline{\text{XIRQ}}$	—	—	V_{DDX}	PUCR/PUPBE	Disabled
49	PB6	—	—	—	V_{DDX}	PUCR/PUPBE	Disabled
50	PB7	—	—	—	V_{DDX}	PUCR/PUPBE	Disabled
51	PC0	—	—	—	V_{DDA}	PUCR/PUPCE	Disabled
52	PC1	—	—	—	V_{DDA}	PUCR/PUPCE	Disabled
53	PC2	—	—	—	V_{DDA}	PUCR/PUPCE	Disabled
54	PC3	—	—	—	V_{DDA}	PUCR/PUPCE	Disabled
55	PAD0	KWAD0	AN0	—	V_{DDA}	PER1AD/PPS1AD	Disabled
56	PAD8	KWAD8	AN8	—	V_{DDA}	PER0AD/PPS0AD	Disabled

Table 2-15. Port J Pins PJ7-0 (continued)

PJ1	• Except 20 TSSOP and 32 LQFP: The SPI1 MOSI signal is mapped to this pin when used with the SPI function. Depending on the configuration of the enabled SPI1 the I/O state is forced to be input or output. • 48 LQFP: The TIM channel 6 signal is mapped to this pin when used with the timer function. The TIM forces the I/O state to be an output for a timer port associated with an enabled output. • Except 20 TSSOP and 32 LQFP: Pin interrupts can be generated if enabled in input or output mode. • Signal priority: 48 LQFP: MOSI1 > IOC6 > GPO 64/100 LQFP: MOSI1 > GPO
PJ0	• Except 20 TSSOP and 32 LQFP: The SPI1 MISO signal is mapped to this pin when used with the SPI function. Depending on the configuration of the enabled SPI1 the I/O state is forced to be input or output. • 48 LQFP: The PWM channel 6 signal is mapped to this pin when used with the PWM function. The enabled PWM channel forces the I/O state to be an output. • Except 20 TSSOP and 32 LQFP: Pin interrupts can be generated if enabled in input or output mode. • Signal priority: 48 LQFP: MISO1 > PWM6 > GPO 64/100 LQFP: MISO1 > GPO

2.3.12 Pins AD15-0

NOTE

The following sources contribute to enable the input buffers on port AD:

- Digital input enable register bits set for each individual pin in ADC
- External trigger function of ADC enabled on ADC channel
- ADC channels routed to port C freeing up pins
- Digital input enable register set bit in and ACMP

Taking the availability of the different sources on each pin into account the following logic equation must be true to activate the digital input buffer for general-purpose input use:

IBEx = ((ATDDIENH/L[IENx]=1) OR (ATDCTL1[ETRIGSEL]=0 AND ATDCTL2[ETRIGE]=1) OR (PRR1[PRR1AN]=1)) AND (ACDIEN=1) ***Eqn. 2-1***

Table 2-57. WOMM Register Field Descriptions

Field	Description
3-0 WOMM	**Port M wired-or mode**—Enable open-drain functionality on output pin This bit configures an output pin as wired-or (open-drain) or push-pull. In wired-or mode a logic “0” is driven active-low while a logic “1” remains undriven. This allows a multipoint connection of several serial modules. The bit has no influence on pins used as input. 1 Output buffer operates as open-drain output. 0 Output buffer operates as push-pull output.

2.4.3.33 Package Code Register (PKGCR)

Address 0x0257 Access: User read/write[1]

	7	6	5	4	3	2	1	0
R	APICLKS7	0	0	0	0	PKGCR2	PKGCR1	PKGCR0
W								
Reset	0	0	0	0	0	F	F	F

After deassert of system reset the values are automatically loaded from the Flash memory. See device specification for details.

Figure 2-34. Package Code Register (PKGCR)

1 Read: Anytime
Write:
APICLKS7: Anytime
PKGCR2-0: Once in normal mode, anytime in special mode

Table 2-58. PKGCR Register Field Descriptions

Field	Description
7 APICLKS7	**Pin Routing Register API_EXTCLK** —Select PS7 as API_EXTCLK output When set to 1 the API_EXTCLK output will be routed to PS7. The default pin will be disconnected in all packages except 20 TSSOP, which has no default location for API_EXTCLK. See Table 2-59 for more details.
2-0 PKGCR	**Package Code Register** —Select package in use Those bits are preset by factory and reflect the package in use. See Table 2-60 for code definition. The bits can be modified once after reset to allow software development for a different package. In any other application it is recommended to re-write the actual package code once after reset to lock the register from inadvertent changes during operation. Writing reserved codes or codes of larger packages than the given device is offered in are illegal. In these cases the code will be converted to PKGCR[2:0]=0b111 and select the maximum available package option for the given device. Codes writes of smaller packages than the given device is offered in are not restricted. Depending on the package selection the input buffers of non-bonded pins are disabled to avoid shoot-through current. Also a predefined signal routing will take effect. Refer also to Section 2.6.5, “Emulation of Smaller Packages”.

If the X bit maskable interrupt request is used to wake-up the MCU with the X bit in the CCR set, the associated ISR is not called. The CPU then resumes program execution with the instruction following the WAI or STOP instruction. This features works following the same rules like any interrupt request, that is care must be taken that the X interrupt request used for wake-up remains active at least until the system begins execution of the instruction following the WAI or STOP instruction; otherwise, wake-up may not occur.

Table 7-5. Hardware Commands (continued)

Command	Opcode (hex)	Data	Description
READ_BD_BYTE	E4	16-bit address 16-bit data out	Read from memory with standard BDM firmware lookup table in map. Odd address data on low byte; even address data on high byte.
READ_BD_WORD	EC	16-bit address 16-bit data out	Read from memory with standard BDM firmware lookup table in map. Must be aligned access.
READ_BYTE	E0	16-bit address 16-bit data out	Read from memory with standard BDM firmware lookup table out of map. Odd address data on low byte; even address data on high byte.
READ_WORD	E8	16-bit address 16-bit data out	Read from memory with standard BDM firmware lookup table out of map. Must be aligned access.
WRITE_BD_BYTE	C4	16-bit address 16-bit data in	Write to memory with standard BDM firmware lookup table in map. Odd address data on low byte; even address data on high byte.
WRITE_BD_WORD	CC	16-bit address 16-bit data in	Write to memory with standard BDM firmware lookup table in map. Must be aligned access.
WRITE_BYTE	C0	16-bit address 16-bit data in	Write to memory with standard BDM firmware lookup table out of map. Odd address data on low byte; even address data on high byte.
WRITE_WORD	C8	16-bit address 16-bit data in	Write to memory with standard BDM firmware lookup table out of map. Must be aligned access.

NOTE:
If enabled, ACK will occur when data is ready for transmission for all BDM READ commands and will occur after the write is complete for all BDM WRITE commands.

7.4.4 Standard BDM Firmware Commands

BDM firmware commands are used to access and manipulate CPU resources. The system must be in active BDM to execute standard BDM firmware commands, see Section 7.4.2, “Enabling and Activating BDM”. Normal instruction execution is suspended while the CPU executes the firmware located in the standard BDM firmware lookup table. The hardware command BACKGROUND is the usual way to activate BDM.

As the system enters active BDM, the standard BDM firmware lookup table and BDM registers become visible in the on-chip memory map at 0x3_FF00–0x3_FFFF, and the CPU begins executing the standard BDM firmware. The standard BDM firmware watches for serial commands and executes them as they are received.

The firmware commands are shown in Table 7-6.

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. Set 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.

= Unimplemented or Reserved

Figure 10-7. S12CPMU Flags Register (CPMUFLG)

Read: Anytime

Write: Refer to each bit for individual write conditions

Table 10-3. 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) fPLL is fVCO / 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.

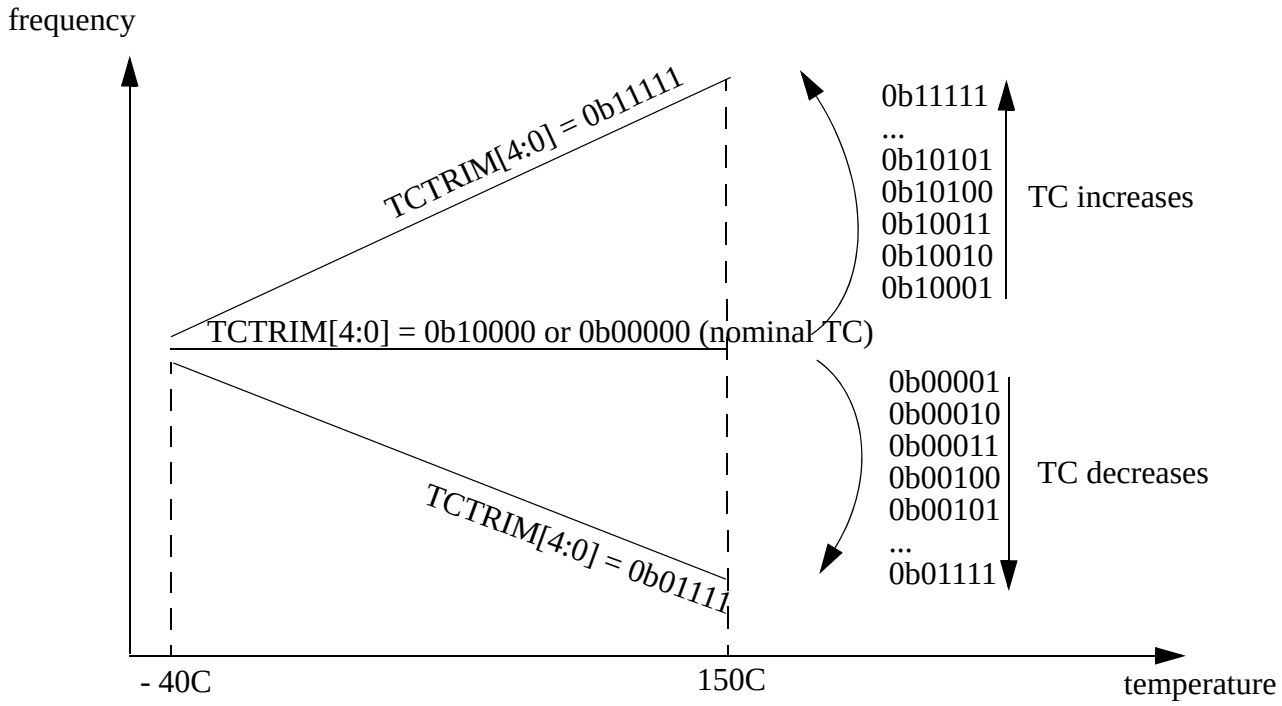

Figure 10-26. Influence of TCTRIM[4:0] on the Temperature Coefficient

NOTE

The frequency is not necessarily linear with the temperature (in most cases it will not be). The above diagram is meant only to give the direction (positive or negative) of the variation of the TC, relative to the nominal TC.

Setting TCTRIM[4:0] to 0b00000 or 0b10000 does not mean that the temperature coefficient will be zero. These two combinations basically switch off the TC compensation module, which results in the nominal TC of the IRC1M.

TCTRIM[4:0]	IRC1M indicative relative TC variation	IRC1M indicative frequency drift for relative TC variation
00000	0 (nominal TC of the IRC)	0%
00001	-0.27%	-0.5%
00010	-0.54%	-0.9%
00011	-0.81%	-1.3%
00100	-1.08%	-1.7%
00101	-1.35%	-2.0%
00110	-1.63%	-2.2%

Table 12-5. External Trigger Channel Select Coding

ETRIGSEL	ETRIGCH3	ETRIGCH2	ETRIGCH1	ETRIGCH0	External trigger source is
0	0	0	0	0	AN0
0	0	0	0	1	AN1
0	0	0	1	0	AN2
0	0	0	1	1	AN3
0	0	1	0	0	AN4
0	0	1	0	1	AN5
0	0	1	1	0	AN6
0	0	1	1	1	AN7
0	1	0	0	0	AN7
0	1	0	0	1	AN7
0	1	0	1	0	AN7
0	1	0	1	1	AN7
0	1	1	0	0	AN7
0	1	1	0	1	AN7
0	1	1	1	0	AN7
0	1	1	1	1	AN7
1	0	0	0	0	ETRIG0[1]
1	0	0	0	1	ETRIG1[1]
1	0	0	1	0	ETRIG2[1]
1	0	0	1	1	ETRIG3[1]
1	0	1	X	X	Reserved
1	1	X	X	X	Reserved

[1] Only if ETRIG3-0 input option is available (see device specification), else ETRISEL is ignored, that means external trigger source is still on one of the AD channels selected by ETRIGCH3-0

12.3.2.3 ATD Control Register 2 (ATDCTL2)

Writes to this register will abort current conversion sequence.

Module Base + 0x0002

	7	6	5	4	3	2	1	0
R	0	AFFC	Reserved	ETRIGLE	ETRIGP	ETRIGE	ASCIE	ACMPIE
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 12-5. ATD Control Register 2 (ATDCTL2)

Read: Anytime

Write: Anytime

message in its RxBG (wrong identifier, transmission errors, etc.) the actual contents of the buffer will be over-written by the next message. The buffer will then not be shifted into the FIFO.

When the MSCAN module is transmitting, the MSCAN receives its own transmitted messages into the background receive buffer, RxBG, but does not shift it into the receiver FIFO, generate a receive interrupt, or acknowledge its own messages on the CAN bus. The exception to this rule is in loopback mode (see Section 18.3.2.2, “MSCAN Control Register 1 (CANCTL1)”) where the MSCAN treats its own messages exactly like all other incoming messages. The MSCAN receives its own transmitted messages in the event that it loses arbitration. If arbitration is lost, the MSCAN must be prepared to become a receiver.

An overrun condition occurs when all receive message buffers in the FIFO are filled with correctly received messages with accepted identifiers and another message is correctly received from the CAN bus with an accepted identifier. The latter message is discarded and an error interrupt with overrun indication is generated if enabled (see Section 18.4.7.5, “Error Interrupt”). The MSCAN remains able to transmit messages while the receiver FIFO is being filled, but all incoming messages are discarded. As soon as a receive buffer in the FIFO is available again, new valid messages will be accepted.

18.4.3 Identifier Acceptance Filter

The MSCAN identifier acceptance registers (see Section 18.3.2.12, “MSCAN Identifier Acceptance Control Register (CANIDAC)”) define the acceptable patterns of the standard or extended identifier (ID[10:0] or ID[28:0]). Any of these bits can be marked ‘don’t care’ in the MSCAN identifier mask registers (see Section 18.3.2.18, “MSCAN Identifier Mask Registers (CANIDMR0–CANIDMR7)”).

A filter hit is indicated to the application software by a set receive buffer full flag (RXF = 1) and three bits in the CANIDAC register (see Section 18.3.2.12, “MSCAN Identifier Acceptance Control Register (CANIDAC)”). These identifier hit flags (IDHIT[2:0]) clearly identify the filter section that caused the acceptance. They simplify the application software’s task to identify the cause of the receiver interrupt. If more than one hit occurs (two or more filters match), the lower hit has priority.

A very flexible programmable generic identifier acceptance filter has been introduced to reduce the CPU interrupt loading. The filter is programmable to operate in four different modes:

- Two identifier acceptance filters, each to be applied to:
  - The full 29 bits of the extended identifier and to the following bits of the CAN 2.0B frame:
    - Remote transmission request (RTR)
    - Identifier extension (IDE)
    - Substitute remote request (SRR)
  - The 11 bits of the standard identifier plus the RTR and IDE bits of the CAN 2.0A/B messages. This mode implements two filters for a full length CAN 2.0B compliant extended identifier. Although this mode can be used for standard identifiers, it is recommended to use the four or eight identifier acceptance filters.
    Figure 18-40 shows how the first 32-bit filter bank (CANIDAR0–CANIDAR3, CANIDMR0–CANIDMR3) produces a filter 0 hit. Similarly, the second filter bank (CANIDAR4–CANIDAR7, CANIDMR4–CANIDMR7) produces a filter 1 hit.
- Four identifier acceptance filters, each to be applied to:

Table 21-3. SPICR2 Field Descriptions

Field	Description
6 XFRW	**Transfer Width —** This bit is used for selecting the data transfer width. If 8-bit transfer width is selected, SPIDRL becomes the dedicated data register and SPIDRH is unused. If 16-bit transfer width is selected, SPIDRH and SPIDRL form a 16-bit data register. Please refer to Section 21.3.2.4, “SPI Status Register (SPISR) for information about transmit/receive data handling and the interrupt flag clearing mechanism. In master mode, a change of this bit will abort a transmission in progress and force the SPI system into idle state. 0 8-bit Transfer Width (n = 8)[1] 1 16-bit Transfer Width (n = 16)[1]
4 MODFEN	**Mode Fault Enable Bit** — This bit allows the MODF failure to be detected. If the SPI is in master mode and MODFEN is cleared, then the $\overline{SS}$ port pin is not used by the SPI. In slave mode, the $\overline{SS}$ is available only as an input regardless of the value of MODFEN. For an overview on the impact of the MODFEN bit on the $\overline{SS}$ port pin configuration, refer to Table 21-2. In master mode, a change of this bit will abort a transmission in progress and force the SPI system into idle state. 0 $\overline{SS}$ port pin is not used by the SPI. 1 $\overline{SS}$ port pin with MODF feature.
3 BIDIROE	**Output Enable in the Bidirectional Mode of Operation** — This bit controls the MOSI and MISO output buffer of the SPI, when in bidirectional mode of operation (SPC0 is set). In master mode, this bit controls the output buffer of the MOSI port, in slave mode it controls the output buffer of the MISO port. In master mode, with SPC0 set, a change of this bit will abort a transmission in progress and force the SPI into idle state. 0 Output buffer disabled. 1 Output buffer enabled.
1 SPISWAI	**SPI Stop in Wait Mode Bit** — This bit is used for power conservation while in wait mode. 0 SPI clock operates normally in wait mode. 1 Stop SPI clock generation when in wait mode.
0 SPC0	**Serial Pin Control Bit 0** — This bit enables bidirectional pin configurations as shown in Table 21-4. In master mode, a change of this bit will abort a transmission in progress and force the SPI system into idle state.

[1] n is used later in this document as a placeholder for the selected transfer width.

Table 21-4. Bidirectional Pin Configurations

Pin Mode	SPC0	BIDIROE	MISO	MOSI
Master Mode of Operation				
Normal	0	X	Master In	Master Out
Bidirectional	1	0	MISO not used by SPI	Master In
		1		Master I/O
Slave Mode of Operation				
Normal	0	X	Slave Out	Slave In
Bidirectional	1	0	Slave In	MOSI not used by SPI
		1	Slave I/O	

25.4.4.2.1 Define FCCOB Contents

The FCCOB parameter fields must be loaded with all required parameters for the Flash command being executed. Access to the FCCOB parameter fields is controlled via the CCOBIX bits in the FCCOBIX register (see Section 25.3.2.3).

The contents of the FCCOB parameter fields are transferred to the Memory Controller when the user clears the CCIF command completion flag in the FSTAT register (writing 1 clears the CCIF to 0). The CCIF flag will remain clear until the Flash command has completed. Upon completion, the Memory Controller will return CCIF to 1 and the FCCOB register will be used to communicate any results. The flow for a generic command write sequence is shown in Figure 25-26.

The user code stored in the P-Flash memory must have a method of receiving the backdoor keys from an external stimulus. This external stimulus would typically be through one of the on-chip serial ports.

If the KEYEN[1:0] bits are in the enabled state (see Section 25.3.2.2), the MCU can be unsecured by the backdoor key access sequence described below:

1. Follow the command sequence for the Verify Backdoor Access Key command as explained in Section 25.4.6.11
2. If the Verify Backdoor Access Key command is successful, the MCU is unsecured and the SEC[1:0] bits in the FSEC register are forced to the unsecure state of 10

The Verify Backdoor Access Key command is monitored by the Memory Controller and an illegal key will prohibit future use of the Verify Backdoor Access Key command. A reset of the MCU is the only method to re-enable the Verify Backdoor Access Key command. The security as defined in the Flash security byte (0x3_FF0F) is not changed by using the Verify Backdoor Access Key command sequence. The backdoor keys stored in addresses 0x3_FF00-0x3_FF07 are unaffected by the Verify Backdoor Access Key command sequence. The Verify Backdoor Access Key command sequence has no effect on the program and erase protections defined in the Flash protection register, FPROT.

After the backdoor keys have been correctly matched, the MCU will be unsecured. After the MCU is unsecured, the sector containing the Flash security byte can be erased and the Flash security byte can be reprogrammed to the unsecure state, if desired. In the unsecure state, the user has full control of the contents of the backdoor keys by programming addresses 0x3_FF00-0x3_FF07 in the Flash configuration field.

25.5.2 Unsecuring the MCU in Special Single Chip Mode using BDM

A secured MCU can be unsecured in special single chip mode by using the following method to erase the P-Flash and EEPROM memory:

1. Reset the MCU into special single chip mode
2. Delay while the BDM executes the Erase Verify All Blocks command write sequence to check if the P-Flash and EEPROM memories are erased
3. Send BDM commands to disable protection in the P-Flash and EEPROM memory
4. Execute the Erase All Blocks command write sequence to erase the P-Flash and EEPROM memory. Alternatively the Unsecure Flash command can be executed, if so the steps 5 and 6 below are skeeped.
5. After the CCIF flag sets to indicate that the Erase All Blocks operation has completed, reset the MCU into special single chip mode
6. Delay while the BDM executes the Erase Verify All Blocks command write sequence to verify that the P-Flash and EEPROM memory are erased

If the P-Flash and EEPROM memory are verified as erased, the MCU will be unsecured. All BDM commands will now be enabled and the Flash security byte may be programmed to the unsecure state by continuing with the following steps:

7. Send BDM commands to execute the Program P-Flash command write sequence to program the Flash security byte to the unsecured state

Table 26-61. Erase Verify EEPROM Section Command Error Handling

Register	Error Bit	Error Condition
FSTAT	ACCERR	Set if CCOBIX[2:0] != 010 at command launch
		Set if command not available in current mode (see Table 26-27)
		Set if an invalid global address [17:0] is supplied
		Set if a misaligned word address is supplied (global address [0] != 0)
		Set if the requested section breaches the end of the EEPROM block
	FPVIOL	None
	MGSTAT1	Set if any errors have been encountered during the read or if blank check failed.
	MGSTAT0	Set if any non-correctable errors have been encountered during the read or if blank check failed.

26.4.6.15 Program EEPROM Command

The Program EEPROM operation programs one to four previously erased words in the EEPROM block. The Program EEPROM operation will confirm that the targeted location(s) were successfully programmed upon completion.

CAUTION

A Flash word must be in the erased state before being programmed. Cumulative programming of bits within a Flash word is not allowed.

Table 26-62. Program EEPROM Command FCCOB Requirements

CCOBIX[2:0]	FCCOB Parameters	
000	0x11	Global address [17:16] to identify the EEPROM block
001	Global address [15:0] of word to be programmed	
010	Word 0 program value	
011	Word 1 program value, if desired	
100	Word 2 program value, if desired	
101	Word 3 program value, if desired	

Upon clearing CCIF to launch the Program EEPROM command, the user-supplied words will be transferred to the Memory Controller and be programmed if the area is unprotected. The CCOBIX index value at Program EEPROM command launch determines how many words will be programmed in the EEPROM block. The CCIF flag is set when the operation has completed.

The user code stored in the P-Flash memory must have a method of receiving the backdoor keys from an external stimulus. This external stimulus would typically be through one of the on-chip serial ports.

If the KEYEN[1:0] bits are in the enabled state (see Section 26.3.2.2), the MCU can be unsecured by the backdoor key access sequence described below:

1. Follow the command sequence for the Verify Backdoor Access Key command as explained in Section 26.4.6.11
2. If the Verify Backdoor Access Key command is successful, the MCU is unsecured and the SEC[1:0] bits in the FSEC register are forced to the unsecure state of 10

The Verify Backdoor Access Key command is monitored by the Memory Controller and an illegal key will prohibit future use of the Verify Backdoor Access Key command. A reset of the MCU is the only method to re-enable the Verify Backdoor Access Key command. The security as defined in the Flash security byte (0x3_FF0F) is not changed by using the Verify Backdoor Access Key command sequence. The backdoor keys stored in addresses 0x3_FF00-0x3_FF07 are unaffected by the Verify Backdoor Access Key command sequence. The Verify Backdoor Access Key command sequence has no effect on the program and erase protections defined in the Flash protection register, FPROT.

After the backdoor keys have been correctly matched, the MCU will be unsecured. After the MCU is unsecured, the sector containing the Flash security byte can be erased and the Flash security byte can be reprogrammed to the unsecure state, if desired. In the unsecure state, the user has full control of the contents of the backdoor keys by programming addresses 0x3_FF00-0x3_FF07 in the Flash configuration field.

26.5.2 Unsecuring the MCU in Special Single Chip Mode using BDM

A secured MCU can be unsecured in special single chip mode by using the following method to erase the P-Flash and EEPROM memory:

1. Reset the MCU into special single chip mode
2. Delay while the BDM executes the Erase Verify All Blocks command write sequence to check if the P-Flash and EEPROM memories are erased
3. Send BDM commands to disable protection in the P-Flash and EEPROM memory
4. Execute the Erase All Blocks command write sequence to erase the P-Flash and EEPROM memory. Alternatively the Unsecure Flash command can be executed, if so the steps 5 and 6 below are skeeped.
5. After the CCIF flag sets to indicate that the Erase All Blocks operation has completed, reset the MCU into special single chip mode
6. Delay while the BDM executes the Erase Verify All Blocks command write sequence to verify that the P-Flash and EEPROM memory are erased

If the P-Flash and EEPROM memory are verified as erased, the MCU will be unsecured. All BDM commands will now be enabled and the Flash security byte may be programmed to the unsecure state by continuing with the following steps:

7. Send BDM commands to execute the Program P-Flash command write sequence to program the Flash security byte to the unsecured state

Offset Module Base + 0x000E

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

= Unimplemented or Reserved

Figure 27-20. Flash Reserved3 Register (FRSV3)

All bits in the FRSV3 register read 0 and are not writable.

27.3.2.15 Flash Reserved4 Register (FRSV4)

This Flash register is reserved for factory testing.

Offset Module Base + 0x000F

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

= Unimplemented or Reserved

Figure 27-21. Flash Reserved4 Register (FRSV4)

All bits in the FRSV4 register read 0 and are not writable.

27.3.2.16 Flash Option Register (FOPT)

The FOPT register is the Flash option register.

Offset Module Base + 0x0010

	7	6	5	4	3	2	1	0
R	NV[7:0]							
W								
Reset	F[1]	F[1]	F[1]	F[1]	F[1]	F[1]	F[1]	F[1]

= Unimplemented or Reserved

Figure 27-22. Flash Option Register (FOPT)

1 Loaded from IFR Flash configuration field, during reset sequence.

All bits in the FOPT register are readable but are not writable.

During the reset sequence, the FOPT register is loaded from the Flash nonvolatile byte in the Flash configuration field at global address 0x3_FF0E located in P-Flash memory (see Table 27-4) as indicated by reset condition F in Figure 27-22. If a double bit fault is detected while reading the P-Flash phrase containing the Flash nonvolatile byte during the reset sequence, all bits in the FOPT register will be set.

Table 29-29. EEPROM Commands

FCMD	Command	Function on EEPROM Memory
0x08	Erase All Blocks	Erase all EEPROM (and P-Flash) blocks. An erase of all Flash blocks is only possible when the FPLDIS, FPHDIS, and FPOPEN bits in the FPROT register and the DPOPEN bit in the DFPROT register are set prior to launching the command.
0x09	Erase Flash Block	Erase a EEPROM (or P-Flash) block. An erase of the full EEPROM block is only possible when DPOPEN bit in the DFPROT register is set prior to launching the command.
0x0B	Unsecure Flash	Supports a method of releasing MCU security by erasing all EEPROM (and P-Flash) blocks and verifying that all EEPROM (and P-Flash) blocks are erased.
0x0D	Set User Margin Level	Specifies a user margin read level for the EEPROM block.
0x0E	Set Field Margin Level	Specifies a field margin read level for the EEPROM block (special modes only).
0x10	Erase Verify EEPROM Section	Verify that a given number of words starting at the address provided are erased.
0x11	Program EEPROM	Program up to four words in the EEPROM block.
0x12	Erase EEPROM Sector	Erase all bytes in a sector of the EEPROM block.

29.4.5 Allowed Simultaneous P-Flash and EEPROM Operations

Only the operations marked ‘OK’ in Table 29-30 are permitted to be run simultaneously on the Program Flash and EEPROM blocks. Some operations cannot be executed simultaneously because certain hardware resources are shared by the two memories. The priority has been placed on permitting Program Flash reads while program and erase operations execute on the EEPROM, providing read (P-Flash) while write (EEPROM) functionality.

Table 29-30. Allowed P-Flash and EEPROM Simultaneous Operations

	EEPROM				
Program Flash	**Read**	**Margin Read[1]**	**Program**	**Sector Erase**	**Mass Erase[2]**
Read		OK	OK	OK	
Margin Read[1]					
Program					
Sector Erase					
Mass Erase[2]					OK

1 A ‘Margin Read’ is any read after executing the margin setting commands ‘Set User Margin Level’ or ‘Set Field Margin Level’ with anything but the ‘normal’ level specified. See the Note on margin settings in Section 29.4.6.12 and Section 29.4.6.13.

2 The ‘Mass Erase’ operations are commands ‘Erase All Blocks’ and ‘Erase Flash Block’

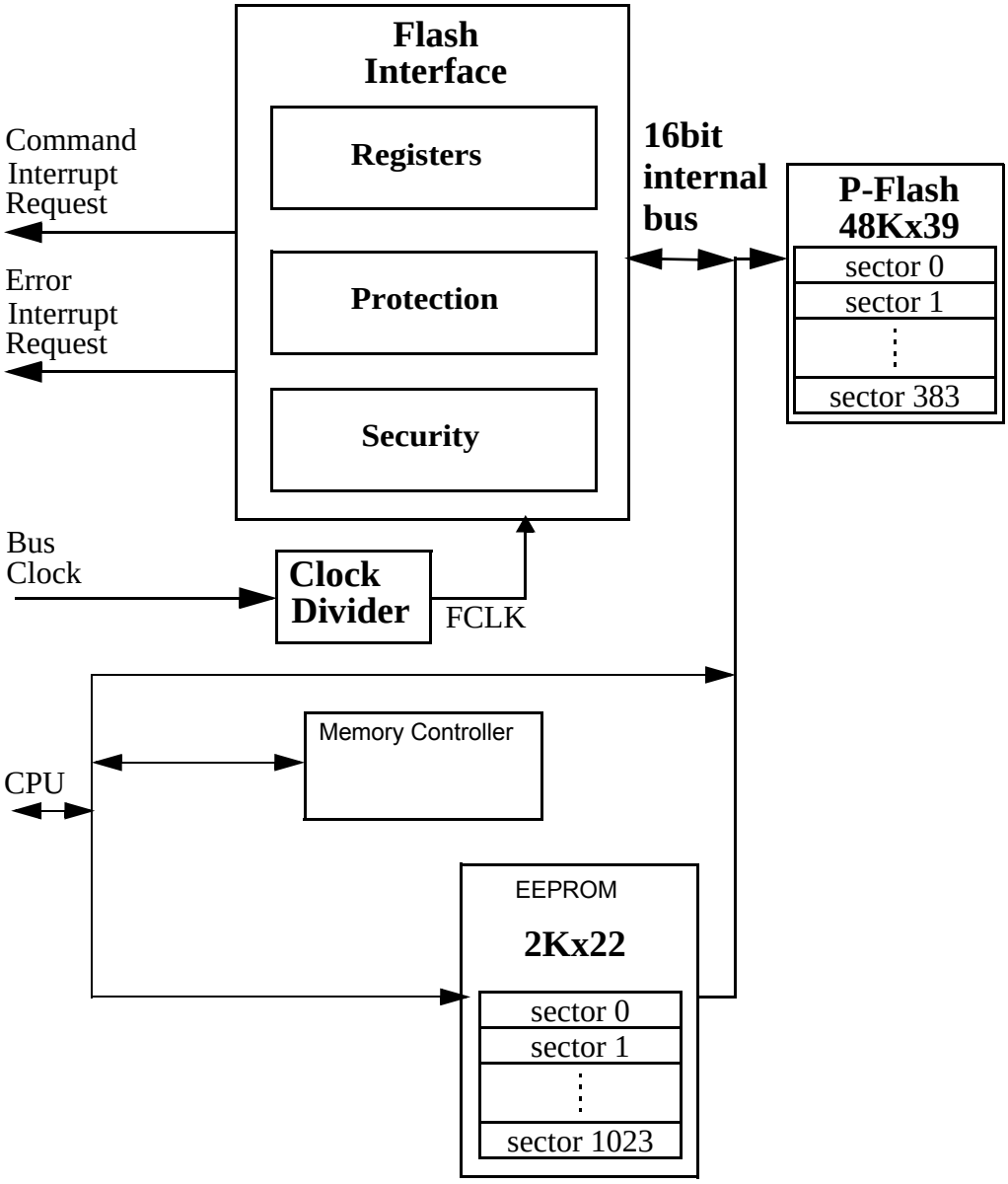

Figure 30-1. FTMRG192K2 Block Diagram

30.2 External Signal Description

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

Table 30-15. FSTAT Field Descriptions (continued)

Field	Description
3 MGBUSY	**Memory Controller Busy Flag** — The MGBUSY flag reflects the active state of the Memory Controller. 0 Memory Controller is idle 1 Memory Controller is busy executing a Flash command (CCIF = 0)
2 RSVD	**Reserved Bit** — This bit is reserved and always reads 0.
1–0 MGSTAT[1:0]	**Memory Controller Command Completion Status Flag** — One or more MGSTAT flag bits are set if an error is detected during execution of a Flash command or during the Flash reset sequence. See Section 30.4.6, “Flash Command Description,” and Section 30.6, “Initialization” for details.

30.3.2.8 Flash Error Status Register (FERSTAT)

The FERSTAT register reflects the error status of internal Flash operations.

Offset Module Base + 0x0007

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	DFDIF	SFDIF
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 30-12. Flash Error Status Register (FERSTAT)

All flags in the FERSTAT register are readable and only writable to clear the flag.

Table 30-16. FERSTAT Field Descriptions

Field	Description
1 DFDIF	**Double Bit Fault Detect Interrupt Flag** — The setting of the DFDIF flag indicates that a double bit fault was detected in the stored parity and data bits during a Flash array read operation or that a Flash array read operation returning invalid data was attempted on a Flash block that was under a Flash command operation.[1] The DFDIF flag is cleared by writing a 1 to DFDIF. Writing a 0 to DFDIF has no effect on DFDIF.[2] 0 No double bit fault detected 1 Double bit fault detected or a Flash array read operation returning invalid data was attempted while command running
0 SFDIF	**Single Bit Fault Detect Interrupt Flag** — With the IGNSF bit in the FCNFG register clear, the SFDIF flag indicates that a single bit fault was detected in the stored parity and data bits during a Flash array read operation or that a Flash array read operation returning invalid data was attempted on a Flash block that was under a Flash command operation.[1] The SFDIF flag is cleared by writing a 1 to SFDIF. Writing a 0 to SFDIF has no effect on SFDIF. 0 No single bit fault detected 1 Single bit fault detected and corrected or a Flash array read operation returning invalid data was attempted while command running

1 The single bit fault and double bit fault flags are mutually exclusive for parity errors (an ECC fault occurrence can be either single fault or double fault but never both). A simultaneous access collision (Flash array read operation returning invalid data attempted while command running) is indicated when both SFDIF and DFDIF flags are high.

2 There is a one cycle delay in storing the ECC DFDIF and SFDIF fault flags in this register. At least one NOP is required after a flash memory read before checking FERSTAT for the occurrence of ECC errors.

Table 31-49. Erase P-Flash Sector Command Error Handling

Register	Error Bit	Error Condition
FSTAT	ACCERR	Set if CCOBIX[2:0] != 001 at command launch
		Set if command not available in current mode (see Table 31-27)
		Set if an invalid global address [17:16] is supplied see Table 31-3)
		Set if a misaligned phrase address is supplied (global address [2:0] != 000)
	FPVIOL	Set if the selected P-Flash sector is protected
	MGSTAT1	Set if any errors have been encountered during the verify operation
	MGSTAT0	Set if any non-correctable errors have been encountered during the verify operation

31.4.6.10 Unsecure Flash Command

The Unsecure Flash command will erase the entire P-Flash and EEPROM memory space and, if the erase is successful, will release security.

Table 31-50. Unsecure Flash Command FCCOB Requirements

CCOBIX[2:0]	FCCOB Parameters	
000	0x0B	Not required

Upon clearing CCIF to launch the Unsecure Flash command, the Memory Controller will erase the entire P-Flash and EEPROM memory space and verify that it is erased. If the Memory Controller verifies that the entire Flash memory space was properly erased, security will be released. If the erase verify is not successful, the Unsecure Flash operation sets MGSTAT1 and terminates without changing the security state. During the execution of this command (CCIF=0) the user must not write to any Flash module register. The CCIF flag is set after the Unsecure Flash operation has completed.

Table 31-51. Unsecure Flash Command Error Handling

Register	Error Bit	Error Condition
FSTAT	ACCERR	Set if CCOBIX[2:0] != 000 at command launch
		Set if command not available in current mode (see Table 31-27)
	FPVIOL	Set if any area of the P-Flash or EEPROM memory is protected
	MGSTAT1	Set if any errors have been encountered during the verify operation
	MGSTAT0	Set if any non-correctable errors have been encountered during the verify operation

31.4.6.11 Verify Backdoor Access Key Command

The Verify Backdoor Access Key command will only execute if it is enabled by the KEYEN bits in the FSEC register (see Table 31-10). The Verify Backdoor Access Key command releases security if user-supplied keys match those stored in the Flash security bytes of the Flash configuration field (see