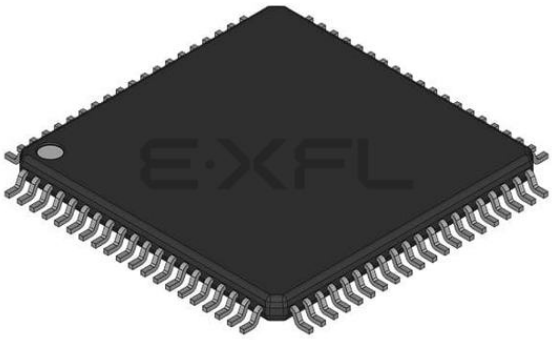


Welcome to E-XFL.COM

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	HCS12
Core Size	16-Bit
Speed	25MHz
Connectivity	CANbus, EBI/EMI, SCI, SPI
Peripherals	POR, PWM, WDT
Number of I/O	60
Program Memory Size	96KB (96K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	2.35V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	80-QFP
Supplier Device Package	80-QFP (14x14)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc9s12c96cfue

0x00E0–0x00FF PWM (Pulse Width Modulator) (continued)

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
$00F6	PWMPER4	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
$00F7	PWMPER5	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
$00F8	PWMDTY0	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
$00F9	PWMDTY1	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
$00FA	PWMDTY2	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
$00FB	PWMDTY3	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
$00FC	PWMDTY4	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
$00FD	PWMDTY5	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
$00FE	Reserved	Read:	PWMIF	PWMIE	0	PWMLVL	0	PWM5IN	PWM5INL	PWM5ENA
		Write:			PWMRSTRT					
$00FF	Reserved	Read:	0	0	0	0	0	0	0	0
		Write:								

0x0110–0x013F Reserved

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0110–0x003F	Reserved	Read:	0	0	0	0	0	0	0	0
		Write:								

0x0140–0x017F CAN (Scalable Controller Area Network — MSCAN)[1]

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0140	CANCTL0	Read:	RXFRM	RXACT	CSWAI	SYNCH	TIME	WUPE	SLPRQ	INITRQ
		Write:								
0x0141	CANCTL1	Read:	CANE	CLKSRC	LOOPB	LISTEN	0	WUPM	SLPAK	INITAK
		Write:								
0x0142	CANBTR0	Read:	SJW1	SJW0	BRP5	BRP4	BRP3	BRP2	BRP1	BRP0
		Write:								
0x0143	CANBTR1	Read:	SAMP	TSEG22	TSEG21	TSEG20	TSEG13	TSEG12	TSEG11	TSEG10
		Write:								
0x0144	CANRFLG	Read:	WUPIF	CSCIF	RSTAT1	RSTAT0	TSTAT1	TSTAT0	OVRIF	RXF
		Write:								
0x0145	CANRIER	Read:	WUPIE	CSCIE	RSTATE1	RSTATE0	TSTATE1	TSTATE0	OVRIE	RXFIE
		Write:								

1.2.3 Part ID Assignments

The part ID is located in two 8-bit registers PARTIDH and PARTIDL (addresses 0x001A and ox001B after reset). The read-only value is a unique part ID for each revision of the chip. Table 1-3 shows the assigned part ID numbers for production mask sets.

Table 1-3. Assigned Part ID Numbers

Device	Mask Set Number	Part ID[1]
MC9S12C32	1L45J	$3300
MC9S12C32	2L45J	$3302
MC9S12C32	1M34C	$3311
MC9S12GC16	2L45J	$3302
MC9S12GC32	2L45J	$3302
MC9S12GC32	1M34C	$3311
MC9S12C64,MC9S12C96,MC9S12C128	2L09S	$3102
MC9S12GC64,MC9S12GC96,MC9S12GC128	2L09S	$3102
MC9S12C64,MC9S12C96,MC9S12C128	0M66G	$3103
MC9S12GC64,MC9S12GC96,MC9S12GC128	0M66G	$3103

1. The coding is as follows:
   Bit 15–12: Major family identifier
   Bit 11–8: Minor family identifier
   Bit 7–4: Major mask set revision number including FAB transfers
   Bit 3–0: Minor — non full — mask set revision

The device memory sizes are located in two 8-bit registers MEMSIZ0 and MEMSIZ1 (addresses 0x001C and 0x001D after reset). Table 1-4 shows the read-only values of these registers. Refer to Module Mapping and Control (MMC) Block Guide for further details.

Table 1-4. Memory Size Registers

Device	Register Name	Value
MC9S12GC16	MEMSIZ0	$00
	MEMSIZ1	$80
MC9S12C32, MC9S12GC32	MEMSIZ0	$00
	MEMSIZ1	$80
MC9S12C64, MC9S12GC64	MEMSIZ0	$01
	MEMSIZ1	$C0
MC9S12C96,MC9S12GC96	MEMSIZ0	$01
	MEMSIZ1	$C0
MC9S12C128, MC9S12GC128	MEMSIZ0	$01
	MEMSIZ1	$C0

1.7.4 VREGEN

The VREGEN input mentioned in the VREG section is device internal, connected internally to V_{DDR}.

1.7.5 V_{DD1}, V_{DD2}, V_{SS1}, V_{SS2}

In the 80-pin QFP package versions, both internal V_{DD} and V_{SS} of the 2.5V domain are bonded out on 2 sides of the device as two pin pairs (V_{DD1}, V_{SS1} & V_{DD2}, V_{SS2}). V_{DD1} and V_{DD2} are connected together internally. V_{SS1} and V_{SS2} are connected together internally. The extra pin pair enables systems using the 80-pin package to employ better supply routing and further decoupling.

1.7.6 Clock Reset Generator And VREG Interface

The low voltage reset feature uses the low voltage reset signal from the VREG module as an input to the CRG module. When the regulator output voltage supply to the internal chip logic falls below a specified threshold the LVR signal from the VREG module causes the CRG module to generate a reset.

NOTE

If the voltage regulator is shut down by connecting V_{DDR} to ground then the LVRF flag in the CRG flags register (CRGFLG) is undefined.

1.7.7 Analog-to-Digital Converter

In the 48- and 52-pin package versions, the V_{RL} pad is bonded internally to the V_{SSA} pin.

1.7.8 MODRR Register Port T And Port P Mapping

The MODRR register within the PIM allows for mapping of PWM channels to port T in the absence of port P pins for the low pin count packages. For the 80QFP package option it is recommended not to use MODRR since this is intended to support PWM channel availability in low pin count packages. Note that when mapping PWM channels to port T in an 80QFP option, the associated PWM channels are then mapped to both port P and port T. .

1.7.9 Port AD Dependency On PIM And ATD Registers

The port AD pins interface to the PIM module. However, the port pin digital state can be read from either the PORTAD register in the ATD register map or from the PTAD register in the PIM register map.

In order to read a digital pin value from PORTAD the corresponding ATDDIEN bit must be set and the corresponding DDRDA bit cleared. If the corresponding ATDDIEN bit is cleared then the pin is configured as an analog input and the PORTAD bit reads back as "1".

In order to read a digital pin value from PTAD, the corresponding DDRAD bit must be cleared, to configure the pin as an input.

Furthermore in order to use a port AD pin as an analog input, the corresponding DDRAD bit must be cleared to configure the pin as an input

2.3.2.6.5 Port AD Pull Device Enable Register (PERAD)

Module Base + 0x0034

	7	6	5	4	3	2	1	0
R W	PERAD7	PERAD6	PERAD5	PERAD4	PERAD3	PERAD2	PERAD1	PERAD0
Reset	0	0	0	0	0	0	0	0

Figure 2-44. Port AD Pull Device Enable Register (PERAD)

Read: Anytime.

Write: Anytime.

Table 2-36. PERAD Field Descriptions

Field	Description
7–0 PERAD[7:0]	**Pull Device Enable Port AD** — This register configures whether a pull-up or a pull-down device is activated, if the port is used as input. This bit has no effect if the port is used as output. Out of reset no pull device is enabled. It is not possible to enable pull devices when a associated ATD channel is enabled simultaneously. 0 Pull-up or pull-down device is disabled. 1 Either a pull-up or pull-down device is enabled.

2.3.2.6.6 Port AD Polarity Select Register (PPSAD)

Module Base + 0x0035

	7	6	5	4	3	2	1	0
R W	PPSAD7	PPSAD6	PPSAD5	PPSAD4	PPSAD3	PPSAD2	PPSAD1	PPSAD0
Reset	0	0	0	0	0	0	0	0

Figure 2-45. Port AD Polarity Select Register (PPSAD)

Read: Anytime.

Write: Anytime.

Table 2-37. PPSAD Field Descriptions

Field	Description
7–0 PPSAD[7:0]	**Pull Select Port AD** — This register selects whether a pull-down or a pull-up device is connected to the pin. 0 A pull-up device is connected to the associated port AD pin, if enabled by the associated bit in register PERAD and if the port is used as input. 1 A pull-down device is connected to the associated port AD pin, if enabled by the associated bit in register PERAD and if the port is used as input.

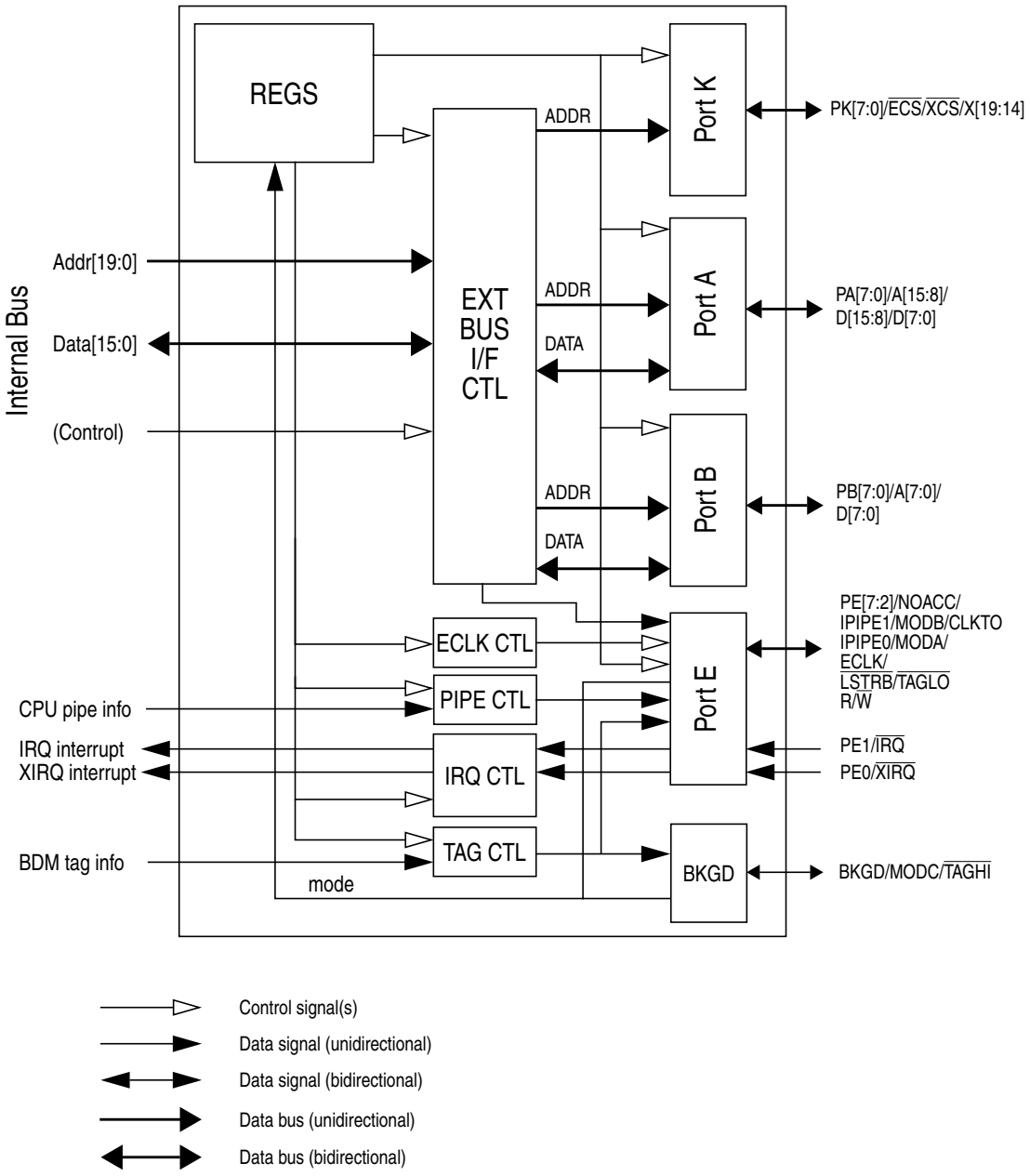


Figure 4-1. MEBI Block Diagram

6.3.2 Register Descriptions

Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0xFF00 Reserved	R	X	X	X	X	X	X	0	0
	W								
0xFF01 BDMSTS	R	ENBDM	BDMACT	ENTAG	SDV	TRACE	CLKSW	UNSEC	0
	W								
0xFF02 Reserved	R	X	X	X	X	X	X	X	X
	W								
0xFF03 Reserved	R	X	X	X	X	X	X	X	X
	W								
0xFF04 Reserved	R	X	X	X	X	X	X	X	X
	W								
0xFF05 Reserved	R	X	X	X	X	X	X	X	X
	W								
0xFF06 BDMCCR	R	CCR7	CCR6	CCR5	CCR4	CCR3	CCR2	CCR1	CCR0
	W								
0xFF07 BDMINR	R	0	REG14	REG13	REG12	REG11	0	0	0
	W								
0xFF08 Reserved	R	0	0	0	0	0	0	0	0
	W								
0xFF09 Reserved	R	0	0	0	0	0	0	0	0
	W								
0xFF0A Reserved	R	X	X	X	X	X	X	X	X
	W								
0xFF0B Reserved	R	X	X	X	X	X	X	X	X
	W								

= Unimplemented, Reserved
X = Indeterminate
= Implemented (do not alter)
0 = Always read zero

Figure 6-2. BDM Register Summary

7.4.2.6 Capture Modes

The DBG in DBG mode can operate in four capture modes. These modes are described in the following subsections.

7.4.2.6.1 Normal Mode

In normal mode, the DBG module uses comparator A and B as triggering devices. Change-of-flow information or data will be stored depending on TRG in DBGSC.

7.4.2.6.2 Loop1 Mode

The intent of loop1 mode is to prevent the trace buffer from being filled entirely with duplicate information from a looping construct such as delays using the DBNE instruction or polling loops using BRSET/BRCLR instructions. Immediately after address information is placed in the trace buffer, the DBG module writes this value into the C comparator and the C comparator is placed in ignore address mode. This will prevent duplicate address entries in the trace buffer resulting from repeated bit-conditional branches. Comparator C will be cleared when the ARM bit is set in loop1 mode to prevent the previous contents of the register from interfering with loop1 mode operation. Breakpoints based on comparator C are disabled.

Loop1 mode only inhibits duplicate source address entries that would typically be stored in most tight looping constructs. It will not inhibit repeated entries of destination addresses or vector addresses, because repeated entries of these would most likely indicate a bug in the user's code that the DBG module is designed to help find.

NOTE

In certain very tight loops, the source address will have already been fetched again before the C comparator is updated. This results in the source address being stored twice before further duplicate entries are suppressed. This condition occurs with branch-on-bit instructions when the branch is fetched by the first P-cycle of the branch or with loop-construct instructions in which the branch is fetched with the first or second P cycle. See examples below:

```
LOOP   INCX                    ; 1-byte instruction fetched by 1st P-cycle of BRCLR
       BRCLR CMPTMP,#$0c,LOOP  ; the BRCLR instruction also will be fetched by 1st P-cycle of BRCLR

LOOP2  BRN   *                 ; 2-byte instruction fetched by 1st P-cycle of DBNE
       NOP                     ; 1-byte instruction fetched by 2nd P-cycle of DBNE
       DBNE  A,LOOP2           ; this instruction also fetched by 2nd P-cycle of DBNE
```

NOTE

Loop1 mode does not support paged memory, and inhibits duplicate entries in the trace buffer based solely on the CPU address. There is a remote possibility of an erroneous address match if program flow alternates between paged and unpaged memory space.

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x001E	ATDDR7H	R	BIT 9 MSB BIT 7 MSB	BIT 8 BIT 6	BIT 7 BIT 5	BIT 6 BIT 4	BIT 5 BIT 3	BIT 4 BIT 2	BIT 3 BIT 1	BIT 2 BIT 0
		W								
0x001F	ATDDR7L	R	BIT 1 u	BIT 0 u	0 0	0 0	0 0	0 0	0 0	0 0
		W								
Right Justified Result Data										
0x0010	ATDDR0H	R	0 0	0 0	0 0	0 0	0 0	0 0	BIT 9 MSB 0	BIT 8 0
		W								
0x0011	ATDDR0L	R	BIT 7 BIT 7 MSB	BIT 6 BIT 6	BIT 5 BIT 5	BIT 4 BIT 4	BIT 3 BIT 3	BIT 2 BIT 2	BIT 1 BIT 1	BIT 0 BIT 0
		W								
0x0012	ATDDR1H	R	0 0	0 0	0 0	0 0	0 0	0 0	BIT 9 MSB 0	BIT 8 0
		W								
0x0013	ATDDR1L	R	BIT 7 BIT 7 MSB	BIT 6 BIT 6	BIT 5 BIT 5	BIT 4 BIT 4	BIT 3 BIT 3	BIT 2 BIT 2	BIT 1 BIT 1	BIT 0 BIT 0
		W								
0x0014	ATDDR2H	R	0 0	0 0	0 0	0 0	0 0	0 0	BIT 9 MSB 0	BIT 8 0
		W								
0x0015	ATDDR2L	R	BIT 7 BIT 7 MSB	BIT 6 BIT 6	BIT 5 BIT 5	BIT 4 BIT 4	BIT 3 BIT 3	BIT 2 BIT 2	BIT 1 BIT 1	BIT 0 BIT 0
		W								
0x0016	ATDDR3H	R	0 0	0 0	0 0	0 0	0 0	0 0	BIT 9 MSB 0	BIT 8 0
		W								
0x0017	ATDDR3L	R	BIT 7 BIT 7 MSB	BIT 6 BIT 6	BIT 5 BIT 5	BIT 4 BIT 4	BIT 3 BIT 3	BIT 2 BIT 2	BIT 1 BIT 1	BIT 0 BIT 0
		W								
0x0018	ATDDR4H	R	0 0	0 0	0 0	0 0	0 0	0 0	BIT 9 MSB 0	BIT 8 0
		W								
0x0019	ATDDR4L	R	BIT 7 BIT 7 MSB	BIT 6 BIT 6	BIT 5 BIT 5	BIT 4 BIT 4	BIT 3 BIT 3	BIT 2 BIT 2	BIT 1 BIT 1	BIT 0 BIT 0
		W								
0x001A	ATDDR5H	R	0 0	0 0	0 0	0 0	0 0	0 0	BIT 9 MSB 0	BIT 8 0
		W								
0x001B	ATDDR5L	R	BIT 7 BIT 7 MSB	BIT 6 BIT 6	BIT 5 BIT 5	BIT 4 BIT 4	BIT 3 BIT 3	BIT 2 BIT 2	BIT 1 BIT 1	BIT 0 BIT 0
		W								

= Unimplemented or Reserved

Figure 8-2. ATD Register Summary (Sheet 3 of 4)

Table 9-2. CRGFLG Field Descriptions (continued)

Field	Description
1 SCMIF	**Self-Clock Mode Interrupt Flag** — SCMIF is set to 1 when SCM status bit changes. This flag can only be cleared by writing a 1. Writing a 0 has no effect. If enabled (SCMIE=1), SCMIF causes an interrupt request. 0 No change in SCM bit. 1 SCM bit has changed.
0 SCM	**Self-Clock Mode Status Bit** — SCM reflects the current clocking mode. Writes have no effect. 0 MCU is operating normally with OSCCLK available. 1 MCU is operating in self-clock mode with OSCCLK in an unknown state. All clocks are derived from PLLCLK running at its minimum frequency f_{SCM}.

9.3.2.5 CRG Interrupt Enable Register (CRGINT)

This register enables CRG interrupt requests.

Module Base + 0x0004

	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

= Unimplemented or Reserved

Figure 9-8. CRG Interrupt Enable Register (CRGINT)

Read: anytime

Write: anytime

Table 9-3. CRGINT Field Descriptions

Field	Description
7 RTIE	**Real-Time Interrupt Enable Bit** 0 Interrupt requests from RTI are disabled. 1 Interrupt will be requested whenever RTIF is set.
4 LOCKIE	**Lock Interrupt Enable Bit** 0 LOCK interrupt requests are disabled. 1 Interrupt will be requested whenever LOCKIF is set.
1 SCMIE	**Self-Clock Mode Interrupt Enable Bit** 0 SCM interrupt requests are disabled. 1 Interrupt will be requested whenever SCMIF is set.

If the bus clock is generated from a PLL, it is recommended to select the oscillator clock rather than the bus clock due to jitter considerations, especially at the faster CAN bus rates.

For microcontrollers without a clock and reset generator (CRG), CANCLK is driven from the crystal oscillator (oscillator clock).

A programmable prescaler generates the time quanta (Tq) clock from CANCLK. A time quantum is the atomic unit of time handled by the MSCAN.

Eqn. 10-2

$$Tq = \frac{f_{CANCLK}}{(\text{Prescaler value})}$$

A bit time is subdivided into three segments as described in the Bosch CAN specification. (see Figure 10-43):

- SYNC_SEG: This segment has a fixed length of one time quantum. Signal edges are expected to happen within this section.
- Time Segment 1: This segment includes the PROP_SEG and the PHASE_SEG1 of the CAN standard. It can be programmed by setting the parameter TSEG1 to consist of 4 to 16 time quanta.
- Time Segment 2: This segment represents the PHASE_SEG2 of the CAN standard. It can be programmed by setting the TSEG2 parameter to be 2 to 8 time quanta long.

Eqn. 10-3

$$\text{Bit Rate} = \frac{f_{Tq}}{(\text{number of Time Quanta})}$$

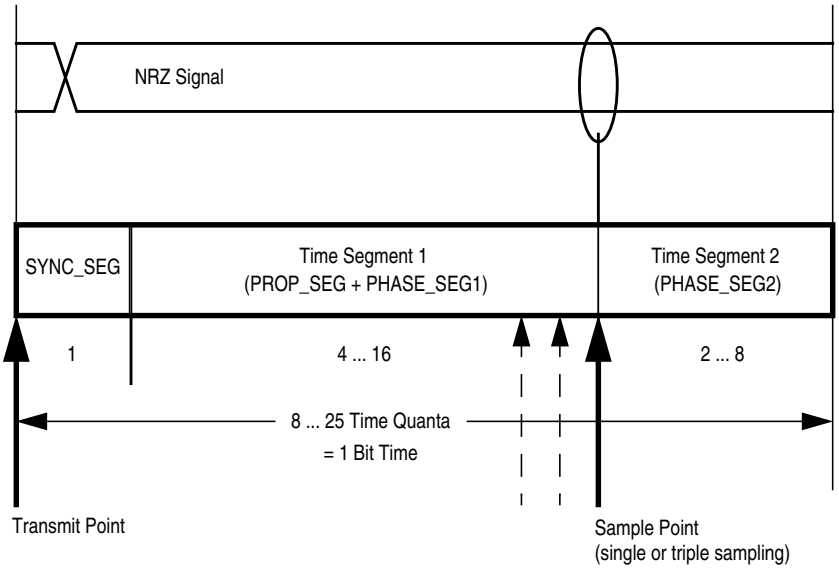


Figure 10-43. Segments within the Bit Time

12.3.2.9 PWM Scale A Register (PWMSCLA)

PWMSCLA is the programmable scale value used in scaling clock A to generate clock SA. Clock SA is generated by taking clock A, dividing it by the value in the PWMSCLA register and dividing that by two.

Clock SA = Clock A / (2 * PWMSCLA)

NOTE

When PWMSCLA = 0x0000, PWMSCLA value is considered a full scale value of 256. Clock A is thus divided by 512.

Any value written to this register will cause the scale counter to load the new scale value (PWMSCLA).

Module Base + 0x0008

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

Figure 12-11. PWM Scale A Register (PWMSCLA)

Read: anytime

Write: anytime (causes the scale counter to load the PWMSCLA value)

12.3.2.10 PWM Scale B Register (PWMSCLB)

PWMSCLB is the programmable scale value used in scaling clock B to generate clock SB. Clock SB is generated by taking clock B, dividing it by the value in the PWMSCLB register and dividing that by two.

Clock SB = Clock B / (2 * PWMSCLB)

NOTE

When PWMSCLB = 0x0000, PWMSCLB value is considered a full scale value of 256. Clock B is thus divided by 512.

Any value written to this register will cause the scale counter to load the new scale value (PWMSCLB).

Module Base + 0x0009

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

Figure 12-12. PWM Scale B Register (PWMSCLB)

Read: anytime

Write: anytime (causes the scale counter to load the PWMSCLB value).

12.3.2.11 Reserved Registers (PWMSCNTx)

The registers PWMSCNTA and PWMSCNTB are reserved for factory testing of the PWM module and are not available in normal modes.

Module Base + 0x000A

	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 12-13. Reserved Register (PWMSCNTA)

Module Base + 0x000B

	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 12-14. Reserved Register (PWMSCNTB)

Read: always read 0x0000 in normal modes

Write: unimplemented in normal modes

NOTE

Writing to these registers when in special modes can alter the PWM functionality.

Module Base + 0x001B

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

Figure 12-30. PWM Channel Duty Registers (PWMDTY3)

Module Base + 0x001C

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

Figure 12-31. PWM Channel Duty Registers (PWMDTY4)

Module Base + 0x001D

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

Figure 12-32. PWM Channel Duty Registers (PWMDTY5)

Read: anytime

Write: anytime

12.3.2.15 PWM Shutdown Register (PWMSDN)

The PWMSDN register provides for the shutdown functionality of the PWM module in the emergency cases.

Module Base + 0x00E

	7	6	5	4	3	2	1	0
R	PWMIF	PWMIE	0	PWMLVL	0	PWM5IN	PWM5INL	PWM5ENA
W			PWMRSTRT					
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 12-33. PWM Shutdown Register (PWMSDN)

Read: anytime

Write: anytime

In Figure 13-16, a large burst of noise is perceived as the beginning of a start bit, although the test sample at RT5 is high. The RT5 sample sets the noise flag. Although this is a worst-case misalignment of perceived bit time, the data samples RT8, RT9, and RT10 are within the bit time and data recovery is successful.

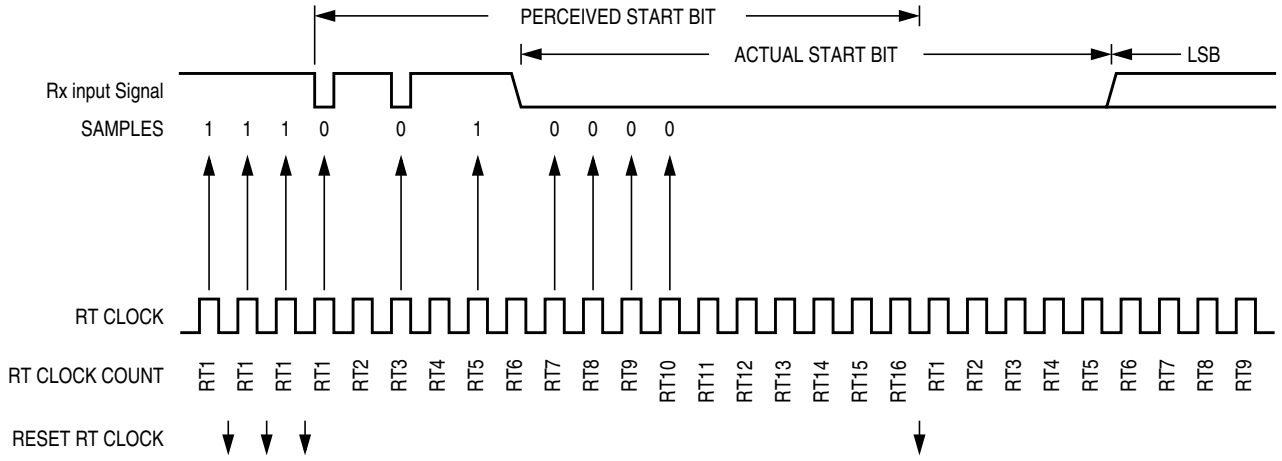


Figure 13-16. Start Bit Search Example 3

Figure 13-17 shows the effect of noise early in the start bit time. Although this noise does not affect proper synchronization with the start bit time, it does set the noise flag.

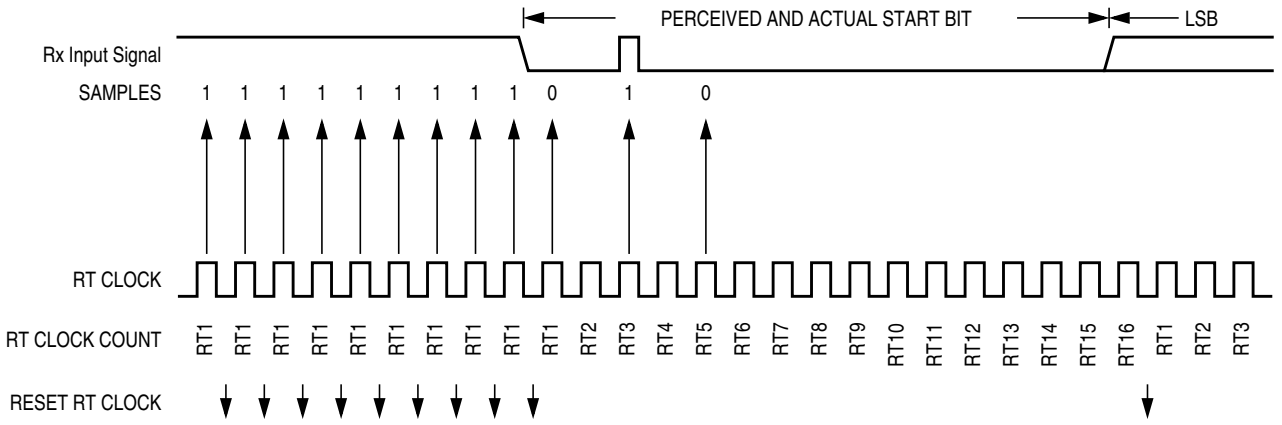


Figure 13-17. Start Bit Search Example 4

18.3.2.4 Flash Configuration Register (FCNFG)

The FCNFG register enables the Flash interrupts and gates the security backdoor key writes.

Module Base + 0x0003

	7	6	5	4	3	2	1	0
R	CBEIE	CCIE	KEYACC	0	0	0	0	0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 18-7. Flash Configuration Register (FCNFG)

CBEIE, CCIE, and KEYACC are readable and writable while remaining bits read 0 and are not writable. KEYACC is only writable if the KEYEN bit in the FSEC register is set to the enabled state (see Section 18.3.2.2).

Table 18-7. FCNFG Field Descriptions

Field	Description
7 CBEIE	**Command Buffer Empty Interrupt Enable** — The CBEIE bit enables the interrupts in case of an empty command buffer in the Flash module. 0 Command Buffer Empty interrupts disabled 1 An interrupt will be requested whenever the CBEIF flag is set (see Section 18.3.2.6)
6 CCIE	**Command Complete Interrupt Enable** — The CCIE bit enables the interrupts in case of all commands being completed in the Flash module. 0 Command Complete interrupts disabled 1 An interrupt will be requested whenever the CCIF flag is set (see Section 18.3.2.6)
5 KEYACC	**Enable Security Key Writing**. 0 Flash writes are interpreted as the start of a command write sequence 1 Writes to the Flash array are interpreted as a backdoor key while reads of the Flash array return invalid data

18.3.2.5 Flash Protection Register (FPROT)

The FPROT register defines which Flash sectors are protected against program or erase.

Module Base + 0x0004

	7	6	5	4	3	2	1	0
R	FPOPEN	NV6	FPHDIS	FPHS1	FPHS0	FPLDIS	FPLS1	FPLS0
W								
Reset	F	F	F	F	F	F	F	F

Figure 18-8. Flash Protection Register (FPROT)

The FPROT register is readable in normal and special modes. FPOPEN can only be written from a 1 to a 0. FPLS[1:0] can be written anytime until FPLDIS is cleared. FPHS[1:0] can be written anytime until

19.1.3 Modes of Operation

See Section 19.4.2, “Operating Modes” for a description of the Flash module operating modes. For program and erase operations, refer to Section 19.4.1, “Flash Command Operations”.

19.1.4 Block Diagram

Figure 19-1Figure 19-2 shows a block diagram of the FTS128K1FTS64K module.

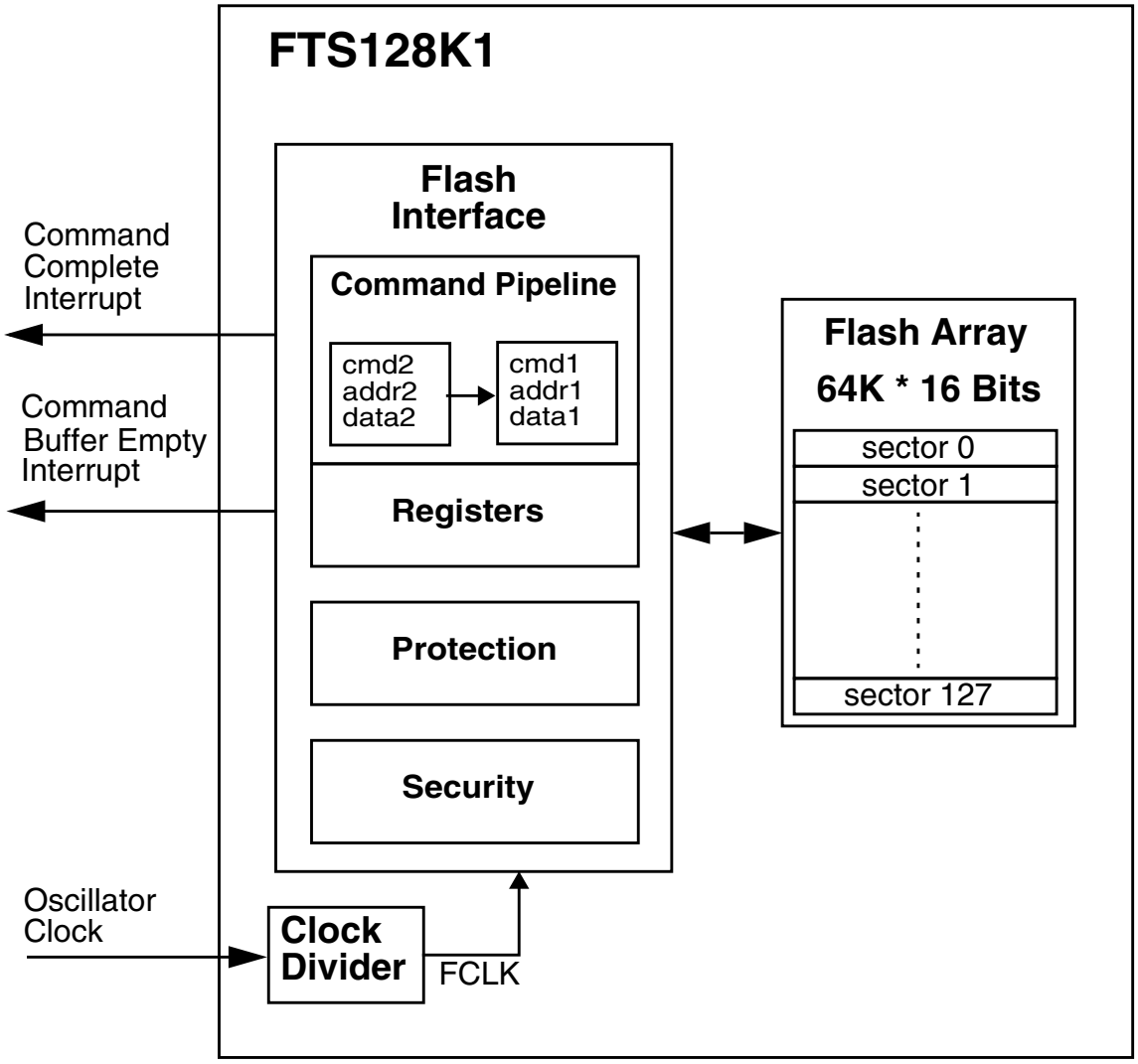


Figure 19-1. FTS128K1 Block Diagram

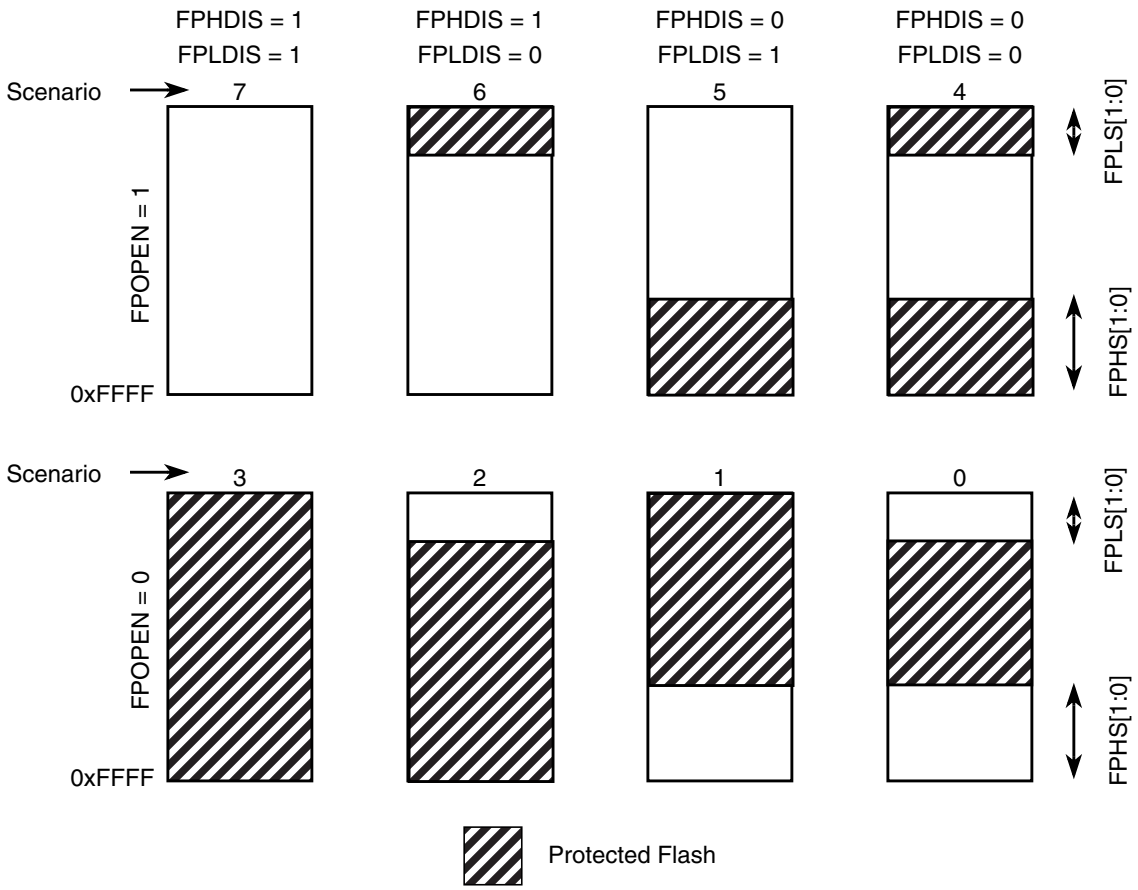


Figure 19-11. Flash Protection Scenarios

19.3.2.5.1 Flash Protection Restrictions

The general guideline is that protection can only be added, not removed. All valid transitions between Flash protection scenarios are specified in Table 19-13. Any attempt to write an invalid scenario to the FPROT register will be ignored and the FPROT register will remain unchanged. The contents of the FPROT register reflect the active protection scenario.

Table 19-13. Flash Protection Scenario Transitions

From Protection Scenario	To Protection Scenario[1]							
	0	1	2	3	4	5	6	7
0	X	X	X	X				
1		X		X				
2			X	X				
3				X				
4				X	X			
5			X	X	X	X		

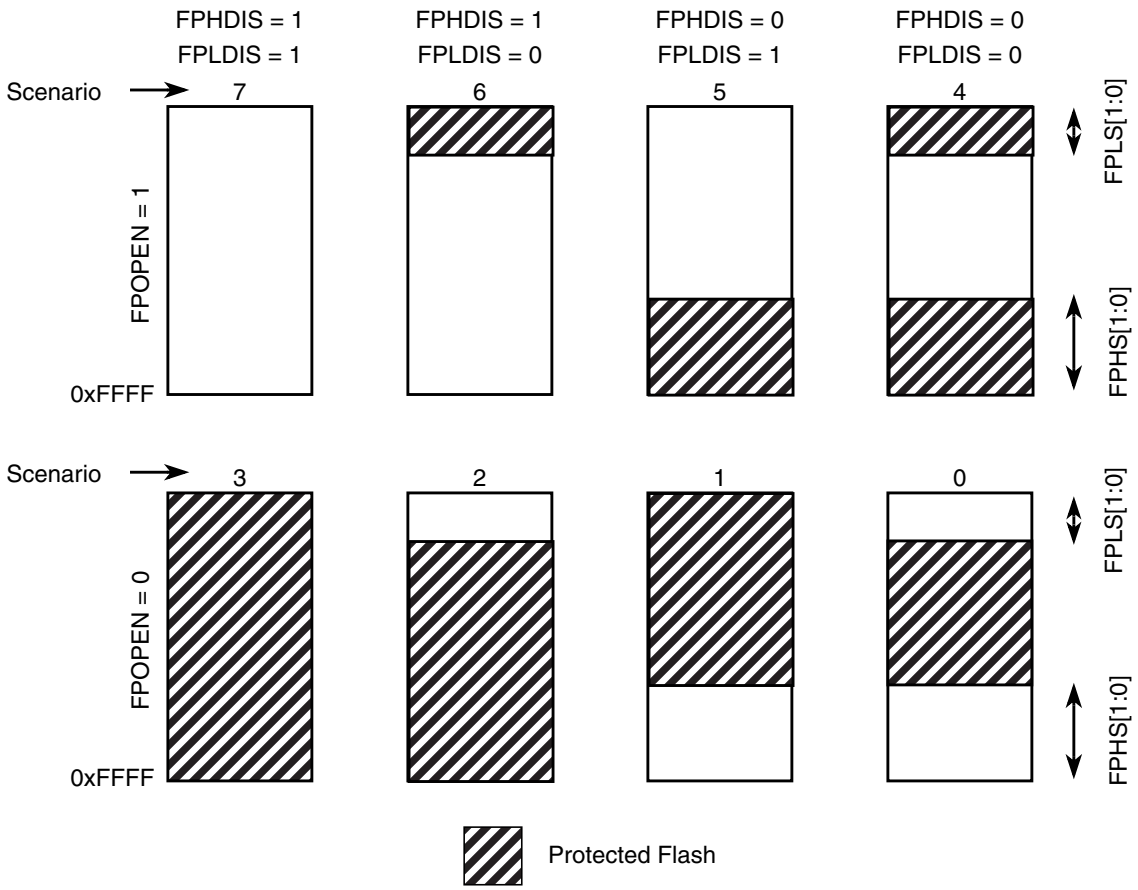


Figure 20-11. Flash Protection Scenarios

20.3.2.5.1 Flash Protection Restrictions

The general guideline is that protection can only be added, not removed. All valid transitions between Flash protection scenarios are specified in Table 20-13. Any attempt to write an invalid scenario to the FPROT register will be ignored and the FPROT register will remain unchanged. The contents of the FPROT register reflect the active protection scenario.

Table 20-13. Flash Protection Scenario Transitions

From Protection Scenario	To Protection Scenario[1]							
	0	1	2	3	4	5	6	7
0	X	X	X	X				
1		X		X				
2			X	X				
3				X				
4				X	X			
5			X	X	X	X		

A.5.2 NVM Reliability

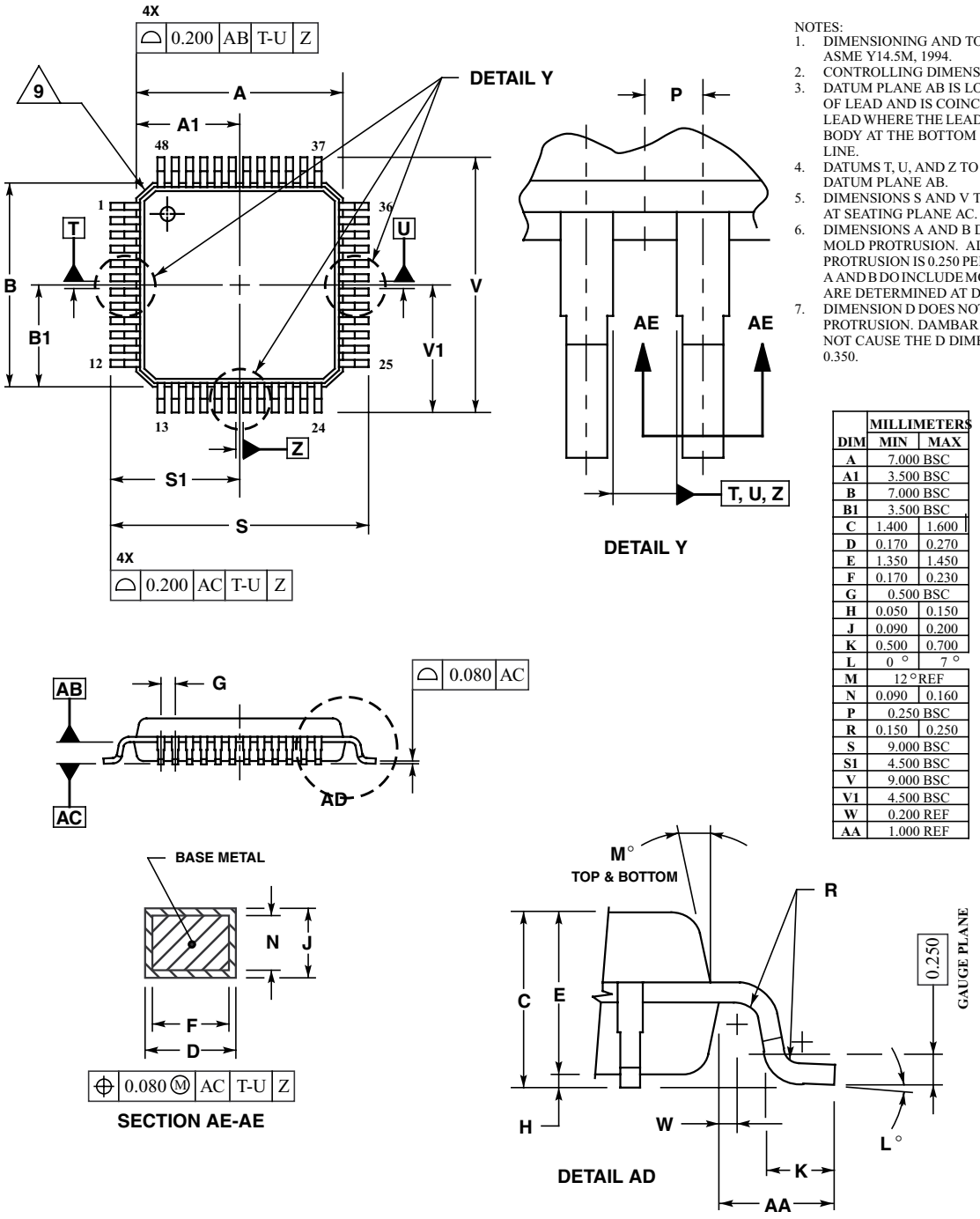
The reliability of the NVM blocks is guaranteed by stress test during qualification, constant process monitors and burn-in to screen early life failures. The program/erase cycle count on the sector is incremented every time a sector or mass erase event is executed.

Table A-19. NVM Reliability Characteristics[1]

Conditions are shown in Table A-4. unless otherwise noted							
Num	**C**	**Rating**	**Symbol**	**Min**	**Typ**	**Max**	**Unit**
Flash Reliability Characteristics							
1	C	Data retention after 10,000 program/erase cycles at an average junction temperature of $T_{Javg} \leq 85°C$	t_{FLRET}	15	100[2]	—	Years
2	C	Data retention with <100 program/erase cycles at an average junction temperature $T_{Javg} \leq 85°C$		20	100[2]	—	
3	C	Number of program/erase cycles ($-40°C \leq T_J \leq 0°C$)	n_{FL}	10,000	—	—	Cycles
4	C	Number of program/erase cycles ($0°C \leq T_J \leq 140°C$)		10,000	100,000[3]	—	

1. T_{Javg} will not exeed 85°C considering a typical temperature profile over the lifetime of a consumer, industrial or automotive application.
2. Typical data retention values are based on intrinsic capability of the technology measured at high temperature and de-rated to 25°C using the Arrhenius equation. For additional information on how Freescale defines Typical Data Retention, please refer to Engineering Bulletin EB618.
3. Spec table quotes typical endurance evaluated at 25°C for this product family, typical endurance at various temperature can be estimated using the graph below. For additional information on how Freescale defines Typical Endurance, please refer to Engineering Bulletin EB619.

C.1.3 48-Pin LQFP Package



NOTES:
1. DIMENSIONING AND TOLERANCING PER ASME Y14.5M, 1994.
2. CONTROLLING DIMENSION: MILLIMETER.
3. DATUM PLANE AB IS LOCATED AT BOTTOM OF LEAD AND IS COINCIDENT WITH THE LEAD WHERE THE LEAD EXITS THE PLASTIC BODY AT THE BOTTOM OF THE PARTING LINE.
4. DATUMS T, U, AND Z TO BE DETERMINED AT DATUM PLANE AB.
5. DIMENSIONS S AND V TO BE DETERMINED AT SEATING PLANE AC.
6. DIMENSIONS A AND B DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.250 PER SIDE. DIMENSIONS A AND B DO INCLUDE MOLD MISMATCH AND ARE DETERMINED AT DATUM PLANE AB.
7. DIMENSION D DOES NOT INCLUDE DAMBAR PROTRUSION. DAMBAR PROTRUSION SHALL NOT CAUSE THE D DIMENSION TO EXCEED 0.350.

	MILLIMETERS	
DIM	MIN	MAX
A	7.000 BSC	
A1	3.500 BSC	
B	7.000 BSC	
B1	3.500 BSC	
C	1.400	1.600
D	0.170	0.270
E	1.350	1.450
F	0.170	0.230
G	0.500 BSC	
H	0.050	0.150
J	0.090	0.200
K	0.500	0.700
L	0 °	7 °
M	12 °REF	
N	0.090	0.160
P	0.250 BSC	
R	0.150	0.250
S	9.000 BSC	
S1	4.500 BSC	
V	9.000 BSC	
V1	4.500 BSC	
W	0.200 REF	
AA	1.000 REF	

Figure C-3. 48-Pin LQFP Mechanical Dimensions (Case no. 932-03 issue F)