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	Obsolete
Core Processor	HCS12
Core Size	16-Bit
Speed	25MHz
Connectivity	CANbus, EBI/EMI, I²C, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	91
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	12K x 8
Voltage - Supply (Vcc/Vdd)	2.35V ~ 5.5V
Data Converters	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	112-LQFP
Supplier Device Package	112-LQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mcs12kg256cpve

2.7.1 Flash Reset Sequence . . . 113
2.7.2 Reset While Flash Command Active . . . 114
2.8 Interrupts . . . 114
2.8.1 Description of Flash Interrupt Operation . . . 114

Chapter 3
256 Kbyte ECC Flash Module (S12FTS256K2ECCV2)

3.1 Introduction . . . 117
3.1.1 Glossary . . . 118
3.1.2 Features . . . 118
3.1.3 Modes of Operation . . . 118
3.1.4 Block Diagram . . . 118
3.2 External Signal Description . . . 119
3.3 Memory Map and Register Definition . . . 120
3.3.1 Module Memory Map . . . 120
3.3.2 Register Descriptions . . . 124
3.4 Functional Description . . . 139
3.4.1 Flash Command Operations (NVM User Mode) . . . 139
3.5 Operating Modes . . . 155
3.5.1 Wait Mode . . . 155
3.5.2 Stop Mode . . . 155
3.5.3 Background Debug Mode . . . 155
3.6 Flash Module Security . . . 156
3.6.1 Unsecuring the MCU using Backdoor Key Access . . . 156
3.6.2 Unsecuring the Flash Module in Special Single-Chip Mode using BDM . . . 157
3.7 Resets . . . 157
3.7.1 Flash Reset Sequence . . . 157
3.7.2 Reset While Flash Command Active . . . 158
3.8 Interrupts . . . 158
3.8.1 Description of Flash Interrupt Operation . . . 158

Chapter 4
4 Kbyte EEPROM Module (S12EETS4KV2)

4.1 Introduction . . . 161
4.1.1 Glossary . . . 161
4.1.2 Features . . . 161
4.1.3 Modes of Operation . . . 162
4.1.4 Block Diagram . . . 162
4.2 External Signal Description . . . 162
4.3 Memory Map and Register Definition . . . 162
4.3.1 Module Memory Map . . . 162
4.3.2 Register Descriptions . . . 165
4.4 Functional Description . . . 173
4.4.1 Program and Erase Operation . . . 173

Except for the sector erase abort command, a buffered command will wait for the active operation to be completed before being launched. The sector erase abort command is launched when the CBEIF flag is cleared as part of a sector erase abort command write sequence. After a command is launched, the completion of the command operation is indicated by the setting of the CCIF flag. The CCIF flag only sets when all active and buffered commands have been completed.

2.4.1.3 Valid Flash Commands

Table 2-21 summarizes the valid Flash commands along with the effects of the commands on the Flash block.

Table 2-21. Valid Flash Command Description

FCMDB	NVM Command	Function on Flash Memory
0x05	Erase Verify	Verify all memory bytes in the Flash block are erased. If the Flash block is erased, the BLANK flag in the FSTAT register will set upon command completion.
0x06	Data Compress	Compress data from a selected portion of the Flash block. The resulting signature is stored in the FDATA register.
0x20	Program	Program a word (two bytes) in the Flash block.
0x40	Sector Erase	Erase all memory bytes in a sector of the Flash block.
0x41	Mass Erase	Erase all memory bytes in the Flash block. A mass erase of the full Flash block is only possible when FPLDIS, FPHDIS, and FPOPEN bits in the FPROT register are set prior to launching the command.
0x47	Sector Erase Abort	Abort the sector erase operation. The sector erase operation will terminate according to a set procedure. The Flash sector must not be considered erased if the ACCERR flag is set upon command completion.

A Flash word must be in the erased state before being programmed. Cumulative programming of bits within a Flash word is not allowed and will result in invalid data stored.

2.4.1.4 Illegal Flash Operations

The ACCERR flag will be set during the command write sequence if any of the following illegal steps are performed, causing the command write sequence to immediately abort:

1. Writing to a Flash address before initializing the FCLKDIV register.
2. Writing to a Flash address in the range 0x8000–0xBFFF when the PPAGE register does not select a 16 Kbyte page in the Flash block selected by the BKSEL bit in the FCNFG register.
3. Writing to a Flash address in the range 0x4000–0x7FFF or 0xC000–0xFFFF with the BKSEL bit in the FCNFG register not selecting Flash block 0.
4. Writing a byte or misaligned word to a valid Flash address.
5. Starting a command write sequence while a data compress operation is active.
6. Starting a command write sequence while a sector erase abort operation is active.
7. Writing a second word to a Flash address in the same command write sequence.
8. Writing to any Flash register other than FCMD after writing a word to a Flash address.
9. Writing a second command to the FCMD register in the same command write sequence.
10. Writing an invalid command to the FCMD register.
11. When security is enabled, writing a command other than mass erase to the FCMD register when the write originates from a non-secure memory location or from the Background Debug Mode.
12. Writing to any Flash register other than FSTAT (to clear CBEIF) after writing to the FCMD register.
13. Writing a 0 to the CBEIF flag in the FSTAT register to abort a command write sequence.

The ACCERR flag will not be set if any Flash register is read during a valid command write sequence.

The ACCERR flag will also be set if any of the following events occur:

1. Launching the sector erase abort command while a sector erase operation is active which results in the early termination of the sector erase operation (see Section 2.4.1.3.6, “Sector Erase Abort Command”)
2. A double bit fault is detected in any of the following Flash operations:
   a) Array read
   b) Erase Verify
   c) Data Compress
   d) Reset Sequence Array Read (Configuration Field)
3. The MCU enters stop mode and a program or erase operation is in progress. The operation is aborted immediately and any pending command is purged (see Section 2.5.2, “Stop Mode”).

If the Flash memory is read during execution of an algorithm (i.e., CCIF flag in the FSTAT register is low), the read operation will return invalid data and the ACCERR flag will not be set.

If the ACCERR flag is set in the FSTAT register, the user must clear the ACCERR flag before starting another command write sequence (see Section 2.3.2.7, “Flash Status Register (FSTAT)”).

3.3.2 Register Descriptions

Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0000 FCLKDIV	R	FDIVLD	PRDIV8	FDIV5	FDIV4	FDIV3	FDIV2	FDIV1	FDIV0
	W								
0x0001 FSEC	R	KEYEN		RNV5	RNV4	RNV3	RNV2	SEC	
	W								
0x0002 FTSTMOD	R	0	0	0	WRALL	0	0	0	0
	W								
0x0003 FCNFG	R	CBEIE	CCIE	KEYACC	0	0	0	0	BKSEL
	W								
0x0004 FPROT	R	FPOPEN	RNV6	FPHDIS	FPHS		FPLDIS	FPLS	
	W								
0x0005 FSTAT	R	CBEIF	CCIF	PVIOL	ACCERR	0	BLANK	0	0
	W								
0x0006 FCMD	R	0	CMDB						
	W								
0x0007 FCTL	R	NV7	NV6	NV5	NV4	NV3	NV2	NV1	NV0
	W								
0x0008 FADDRHI	R	FADDRHI							
	W								
0x0009 FADDRLO	R	FADDRLO							
	W								
0x000A FDATAHI	R	FDATAHI							
	W								
0x000B FDATALO	R	FDATALO							
	W								
0x000C RESERVED1	R	0	0	0	0	0	0	0	0
	W								

= Unimplemented or Reserved

Figure 3-3. FTS256K2ECC Register Summary

5.3.6.3 Port J Data Direction Register (DDRJ)

Module Base + 0x002A

	7	6	5	4	3	2	1	0
R	DDRJ7	DDRJ6	0	0	0	0	DDRJ1	DDRJ0
W								
Reset	0	0	—	—	—	—	0	0

= Unimplemented or Reserved

Figure 5-41. Port J Data Direction Register (DDRJ)

Read: Anytime. Write: Anytime.

This register configures each port J pin as either input or output.

If enable, CAN4 forces the I/O state to be an output on PJ7 (TXCAN4) and an input on pin PJ6 (RXCAN4). If CAN4 is disabled, the IIC takes control of the I/O if enabled. In these cases the data direction bits will not change.

The DDRJ bits revert to controlling the I/O direction of a pin when the associated peripheral module is disabled.

Table 5-36. Field Descriptions

Field	Description
7, 6, 1, 0 DDRJ[7:6] DDRJ[1:0]	**Data Direction Port J** 0 Associated pin is configured as input. 1 Associated pin is configured as output.

5.3.6.4 Port J Reduced Drive Register (RDRJ)

Module Base + 0x002B

	7	6	5	4	3	2	1	0
R	RDRJ7	RDRJ6	0	0	0	0	RDRJ1	RDRJ0
W								
Reset	0	0	—	—	—	—	0	0

= Unimplemented or Reserved

Figure 5-42. Port J Reduced Drive Register (RDRJ)

Read: Anytime. Write: Anytime.

This register configures the drive strength of each port J output pin as either full or reduced. If the port is used as input this bit is ignored.

Table 5-37. RDRJ Field Descriptions

Field	Description
7, 6, 1, 0 RDRJ[7:6] RDRJ[1:0]	**Reduced Drive Port J** 0 Full drive strength at output. 1 Associated pin drives at about 1/6 of the full drive strength.

Chapter 6
Clocks and Reset Generator (CRGV4) Block Description

6.1 Introduction

This specification describes the function of the clocks and reset generator (CRGV4).

6.1.1 Features

The main features of this block are:

- Phase-locked loop (PLL) frequency multiplier
  - Reference divider
  - Automatic bandwidth control mode for low-jitter operation
  - Automatic frequency lock detector
  - CPU interrupt on entry or exit from locked condition
  - Self-clock mode in absence of reference clock
- System clock generator
  - Clock quality check
  - Clock switch for either oscillator- or PLL-based system clocks
  - User selectable disabling of clocks during wait mode for reduced power consumption
- Computer operating properly (COP) watchdog timer with time-out clear window
- System reset generation from the following possible sources:
  - Power-on reset
  - Low voltage reset
    Refer to the device overview section for availability of this feature.
  - COP reset
  - Loss of clock reset
  - External pin reset
- Real-time interrupt (RTI)

8.3.2.6 ATD Control Register 5 (ATDCTL5)

This register selects the type of conversion sequence and the analog input channels sampled. Writes to this register will abort current conversion sequence and start a new conversion sequence.

Module Base + 0x0005

	7	6	5	4	3	2	1	0
R	DJM	DSGN	SCAN	MULT	0	CC	CB	CA
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 8-8. ATD Control Register 5 (ATDCTL5)

Read: Anytime

Write: Anytime

Table 8-13. ATDCTL5 Field Descriptions

Field	Description
7 DJM	**Result Register Data Justification** — This bit controls justification of conversion data in the result registers. See Section 8.3.2.13, “ATD Conversion Result Registers (ATDDRx),” for details. 0 Left justified data in the result registers 1 Right justified data in the result registers
6 DSGN	**Result Register Data Signed or Unsigned Representation** — This bit selects between signed and unsigned conversion data representation in the result registers. Signed data is represented as 2’s complement. Signed data is not available in right justification. See Section 8.3.2.13, “ATD Conversion Result Registers (ATDDRx),” for details. 0 Unsigned data representation in the result registers 1 Signed data representation in the result registers Table 8-14 summarizes the result data formats available and how they are set up using the control bits. Table 8-15 illustrates the difference between the signed and unsigned, left justified output codes for an input signal range between 0 and 5.12 Volts.
5 SCAN	**Continuous Conversion Sequence Mode** — This bit selects whether conversion sequences are performed continuously or only once. 0 Single conversion sequence 1 Continuous conversion sequences (scan mode)
4 MULT	**Multi-Channel Sample Mode** — When MULT is 0, the ATD sequence controller samples only from the specified analog input channel for an entire conversion sequence. The analog channel is selected by channel selection code (control bits CC/CB/CA located in ATDCTL5). When MULT is 1, the ATD sequence controller samples across channels. The number of channels sampled is determined by the sequence length value (S8C, S4C, S2C, S1C). The first analog channel examined is determined by channel selection code (CC, CB, CA control bits); subsequent channels sampled in the sequence are determined by incrementing the channel selection code. 0 Sample only one channel 1 Sample across several channels
2–0 CC, CB, CA	**Analog Input Channel Select Code** — These bits select the analog input channel(s) whose signals are sampled and converted to digital codes. Table 8-16 lists the coding used to select the various analog input channels. In the case of single channel scans (MULT = 0), this selection code specified the channel examined. In the case of multi-channel scans (MULT = 1), this selection code represents the first channel to be examined in the conversion sequence. Subsequent channels are determined by incrementing channel selection code; selection codes that reach the maximum value wrap around to the minimum value.

10.3.2.4 MSCAN Bus Timing Register 1 (CANBTR1)

The CANBTR1 register configures various CAN bus timing parameters of the MSCAN module.

Module Base + 0x0003

	7	6	5	4	3	2	1	0
R W	SAMP	TSEG22	TSEG21	TSEG20	TSEG13	TSEG12	TSEG11	TSEG10
Reset:	0	0	0	0	0	0	0	0

Figure 10-7. MSCAN Bus Timing Register 1 (CANBTR1)

Read: Anytime
Write: Anytime in initialization mode (INITRQ = 1 and INITAK = 1)

Table 10-6. CANBTR1 Register Field Descriptions

Field	Description
7 SAMP	**Sampling** — This bit determines the number of CAN bus samples taken per bit time. 0 One sample per bit. 1 Three samples per bit[1]. If SAMP = 0, the resulting bit value is equal to the value of the single bit positioned at the sample point. If SAMP = 1, the resulting bit value is determined by using majority rule on the three total samples. For higher bit rates, it is recommended that only one sample is taken per bit time (SAMP = 0).
6:4 TSEG2[2:0]	**Time Segment 2** — Time segments within the bit time fix the number of clock cycles per bit time and the location of the sample point (see Figure 10-43). Time segment 2 (TSEG2) values are programmable as shown in Table 10-7.
3:0 TSEG1[3:0]	**Time Segment 1** — Time segments within the bit time fix the number of clock cycles per bit time and the location of the sample point (see Figure 10-43). Time segment 1 (TSEG1) values are programmable as shown in Table 10-8.

1 In this case, PHASE_SEG1 must be at least 2 time quanta (Tq).

Table 10-7. Time Segment 2 Values

TSEG22	TSEG21	TSEG20	Time Segment 2
0	0	0	1 Tq clock cycle[1]
0	0	1	2 Tq clock cycles
:	:	:	:
1	1	0	7 Tq clock cycles
1	1	1	8 Tq clock cycles

1 This setting is not valid. Please refer to Table 10-34 for valid settings.

Table 10-8. Time Segment 1 Values

TSEG13	TSEG12	TSEG11	TSEG10	Time segment 1
0	0	0	0	1 Tq clock cycle[1]
0	0	0	1	2 Tq clock cycles[1]
0	0	1	0	3 Tq clock cycles[1]
0	0	1	1	4 Tq clock cycles
:	:	:	:	:
1	1	1	0	15 Tq clock cycles
1	1	1	1	16 Tq clock cycles

1 This setting is not valid. Please refer to Table 10-34 for valid settings.

The bit time is determined by the oscillator frequency, the baud rate prescaler, and the number of time quanta (Tq) clock cycles per bit (as shown in Table 10-7 and Table 10-8).

Eqn. 10-1

$$\text{Bit Time} = \frac{(\text{Prescaler value})}{f_{CANCLK}} \bullet (1 + \text{TimeSegment1} + \text{TimeSegment2})$$

10.3.2.5 MSCAN Receiver Flag Register (CANRFLG)

A flag can be cleared only by software (writing a 1 to the corresponding bit position) when the condition which caused the setting is no longer valid. Every flag has an associated interrupt enable bit in the CANRIER register.

Module Base + 0x0004

	7	6	5	4	3	2	1	0
R	WUPIF	CSCIF	RSTAT1	RSTAT0	TSTAT1	TSTAT0	OVRIF	RXF
W								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 10-8. MSCAN Receiver Flag Register (CANRFLG)

NOTE

The CANRFLG register is held in the reset state[1] when the initialization mode is active (INITRQ = 1 and INITAK = 1). This register is writable again as soon as the initialization mode is exited (INITRQ = 0 and INITAK = 0).

Read: Anytime
Write: Anytime when out of initialization mode, except RSTAT[1:0] and TSTAT[1:0] flags which are read-only; write of 1 clears flag; write of 0 is ignored.

1. The RSTAT[1:0], TSTAT[1:0] bits are not affected by initialization mode.

10.4.2.1 Message Transmit Background

Modern application layer software is built upon two fundamental assumptions:

- Any CAN node is able to send out a stream of scheduled messages without releasing the CAN bus between the two messages. Such nodes arbitrate for the CAN bus immediately after sending the previous message and only release the CAN bus in case of lost arbitration.
- The internal message queue within any CAN node is organized such that the highest priority message is sent out first, if more than one message is ready to be sent.

The behavior described in the bullets above cannot be achieved with a single transmit buffer. That buffer must be reloaded immediately after the previous message is sent. This loading process lasts a finite amount of time and must be completed within the inter-frame sequence (IFS) to be able to send an uninterrupted stream of messages. Even if this is feasible for limited CAN bus speeds, it requires that the CPU reacts with short latencies to the transmit interrupt.

A double buffer scheme de-couples the reloading of the transmit buffer from the actual message sending and, therefore, reduces the reactiveness requirements of the CPU. Problems can arise if the sending of a message is finished while the CPU re-loads the second buffer. No buffer would then be ready for transmission, and the CAN bus would be released.

At least three transmit buffers are required to meet the first of the above requirements under all circumstances. The MSCAN has three transmit buffers.

The second requirement calls for some sort of internal prioritization which the MSCAN implements with the "local priority" concept described in Section 10.4.2.2, "Transmit Structures."

10.4.2.2 Transmit Structures

The MSCAN triple transmit buffer scheme optimizes real-time performance by allowing multiple messages to be set up in advance. The three buffers are arranged as shown in Figure 10-38.

All three buffers have a 13-byte data structure similar to the outline of the receive buffers (see Section 10.3.3, "Programmer's Model of Message Storage"). An additional Section 10.3.3.4, "Transmit Buffer Priority Register (TBPR) contains an 8-bit local priority field (PRIO) (see Section 10.3.3.4, "Transmit Buffer Priority Register (TBPR)"). The remaining two bytes are used for time stamping of a message, if required (see Section 10.3.3.5, "Time Stamp Register (TSRH–TSRL)").

To transmit a message, the CPU must identify an available transmit buffer, which is indicated by a set transmitter buffer empty (TXEx) flag (see Section 10.3.2.7, "MSCAN Transmitter Flag Register (CANTFLG)"). If a transmit buffer is available, the CPU must set a pointer to this buffer by writing to the CANTBSEL register (see Section 10.3.2.11, "MSCAN Transmit Buffer Selection Register (CANTBSEL)"). This makes the respective buffer accessible within the CANTXFG address space (see Section 10.3.3, "Programmer's Model of Message Storage"). The algorithmic feature associated with the CANTBSEL register simplifies the transmit buffer selection. In addition, this scheme makes the handler software simpler because only one address area is applicable for the transmit process, and the required address space is minimized.

The CPU then stores the identifier, the control bits, and the data content into one of the transmit buffers. Finally, the buffer is flagged as ready for transmission by clearing the associated TXE flag.

Table 14-15. TSCR2 Field Descriptions

Field	Description
7 TOI	**Timer Overflow Interrupt Enable** 0 Interrupt inhibited. 1 Hardware interrupt requested when TOF flag set.
3 TCRE	**Timer Counter Reset Enable** — This bit allows the timer counter to be reset by a successful output compare 7 event. This mode of operation is similar to an up-counting modulus counter. 0 Counter reset inhibited and counter free runs. 1 Counter reset by a successful output compare 7. **Note:** If TC7 = 0x0000 and TCRE = 1, TCNT will stay at 0x0000 continuously. If TC7 = 0xFFFF and TCRE = 1, TOF will never be set when TCNT is reset from 0xFFFF to 0x0000. **Note:** TCRE=1 and TC7!=0, the TCNT cycle period will be TC7 x "prescaler counter width" + "1 Bus Clock", for a more detail explanation please refer to Section 14.4.3, “Output Compare
2 PR[2:0]	**Timer Prescaler Select** — These three bits select the frequency of the timer prescaler clock derived from the Bus Clock as shown in Table 14-16.

Table 14-16. Timer Clock Selection

PR2	PR1	PR0	Timer Clock
0	0	0	Bus Clock / 1
0	0	1	Bus Clock / 2
0	1	0	Bus Clock / 4
0	1	1	Bus Clock / 8
1	0	0	Bus Clock / 16
1	0	1	Bus Clock / 32
1	1	0	Bus Clock / 64
1	1	1	Bus Clock / 128

NOTE

The newly selected prescale factor will not take effect until the next synchronized edge where all prescale counter stages equal zero.

14.3.2.12 Main Timer Interrupt Flag 1 (TFLG1)

Module Base + 0x000E

	7	6	5	4	3	2	1	0
R W	C7F	C6F	C5F	C4F	C3F	C2F	C1F	C0F
Reset	0	0	0	0	0	0	0	0

Figure 14-20. Main Timer Interrupt Flag 1 (TFLG1)

Read: Anytime

16.3.2.1 BDM Status Register (BDMSTS)

0xFF01

	7	6	5	4	3	2	1	0
R	ENBDM	BDMACT	ENTAG	SDV	TRACE	CLKSW	UNSEC	0
W								
Reset:								
Special single-chip mode:	1[1]	1	0	0	0	0	0[2]	0
Special peripheral mode:	0	1	0	0	0	0	0	0
All other modes:	0	1	0	0	0	0	0	0
	0	0	0	0	0	0	0	0

= Unimplemented or Reserved = Implemented (do not alter)

Figure 16-3. BDM Status Register (BDMSTS)

Note:

1 ENBDM is read as "1" by a debugging environment in Special single-chip mode when the device is not secured or secured but fully erased (Flash and EEPROM).This is because the ENBDM bit is set by the standard firmware before a BDM command can be fully transmitted and executed.

2 UNSEC is read as "1" by a debugging environment in Special single-chip mode when the device is secured and fully erased, else it is "0" and can only be read if not secure (see also bit description).

Read: All modes through BDM operation

Write: All modes but subject to the following:

- BDMACT can only be set by BDM hardware upon entry into BDM. It can only be cleared by the standard BDM firmware lookup table upon exit from BDM active mode.
- CLKSW can only be written via BDM hardware or standard BDM firmware write commands.
- All other bits, while writable via BDM hardware or standard BDM firmware write commands, should only be altered by the BDM hardware or standard firmware lookup table as part of BDM command execution.
- ENBDM should only be set via a BDM hardware command if the BDM firmware commands are needed. (This does not apply in special single-chip mode).

16.4.8 Hardware Handshake Abort Procedure

The abort procedure is based on the SYNC command. In order to abort a command, which had not issued the corresponding ACK pulse, the host controller should generate a low pulse in the BKGD pin by driving it low for at least 128 serial clock cycles and then driving it high for one serial clock cycle, providing a speedup pulse. By detecting this long low pulse in the BKGD pin, the target executes the SYNC protocol, see Section 16.4.9, “SYNC — Request Timed Reference Pulse,” and assumes that the pending command and therefore the related ACK pulse, are being aborted. Therefore, after the SYNC protocol has been completed the host is free to issue new BDM commands.

Although it is not recommended, the host could abort a pending BDM command by issuing a low pulse in the BKGD pin shorter than 128 serial clock cycles, which will not be interpreted as the SYNC command. The ACK is actually aborted when a falling edge is perceived by the target in the BKGD pin. The short abort pulse should have at least 4 clock cycles keeping the BKGD pin low, in order to allow the falling edge to be detected by the target. In this case, the target will not execute the SYNC protocol but the pending command will be aborted along with the ACK pulse. The potential problem with this abort procedure is when there is a conflict between the ACK pulse and the short abort pulse. In this case, the target may not perceive the abort pulse. The worst case is when the pending command is a read command (i.e., READ_BYTE). If the abort pulse is not perceived by the target the host will attempt to send a new command after the abort pulse was issued, while the target expects the host to retrieve the accessed memory byte. In this case, host and target will run out of synchronism. However, if the command to be aborted is not a read command the short abort pulse could be used. After a command is aborted the target assumes the next falling edge, after the abort pulse, is the first bit of a new BDM command.

NOTE

The details about the short abort pulse are being provided only as a reference for the reader to better understand the BDM internal behavior. It is not recommended that this procedure be used in a real application.

Because the host knows the target serial clock frequency, the SYNC command (used to abort a command) does not need to consider the lower possible target frequency. In this case, the host could issue a SYNC very close to the 128 serial clock cycles length. Providing a small overhead on the pulse length in order to assure the SYNC pulse will not be misinterpreted by the target. See Section 16.4.9, “SYNC — Request Timed Reference Pulse.”

Figure 16-12 shows a SYNC command being issued after a READ_BYTE, which aborts the READ_BYTE command. Note that, after the command is aborted a new command could be issued by the host computer.

NOTE

Figure 16-12 does not represent the signals in a true timing scale

If an interrupt is pending when a TRACE1 command is issued, the interrupt stacking operation occurs but no user instruction is executed. Upon return to standard BDM firmware execution, the program counter points to the first instruction in the interrupt service routine.

16.4.11 Instruction Tagging

The instruction queue and cycle-by-cycle CPU activity are reconstructible in real time or from trace history that is captured by a logic analyzer. However, the reconstructed queue cannot be used to stop the CPU at a specific instruction. This is because execution already has begun by the time an operation is visible outside the system. A separate instruction tagging mechanism is provided for this purpose.

The tag follows program information as it advances through the instruction queue. When a tagged instruction reaches the head of the queue, the CPU enters active BDM rather than executing the instruction.

NOTE

Tagging is disabled when BDM becomes active and BDM serial commands are not processed while tagging is active.

Executing the BDM TAGGO command configures two system pins for tagging. The $\overline{\text{TAGLO}}$ signal shares a pin with the $\overline{\text{LSTRB}}$ signal, and the $\overline{\text{TAGHI}}$ signal shares a pin with the BKGD signal.

Table 16-7 shows the functions of the two tagging pins. The pins operate independently, that is the state of one pin does not affect the function of the other. The presence of logic level 0 on either pin at the fall of the external clock (ECLK) performs the indicated function. High tagging is allowed in all modes. Low tagging is allowed only when low strobe is enabled (LSTRB is allowed only in wide expanded modes and emulation expanded narrow mode).

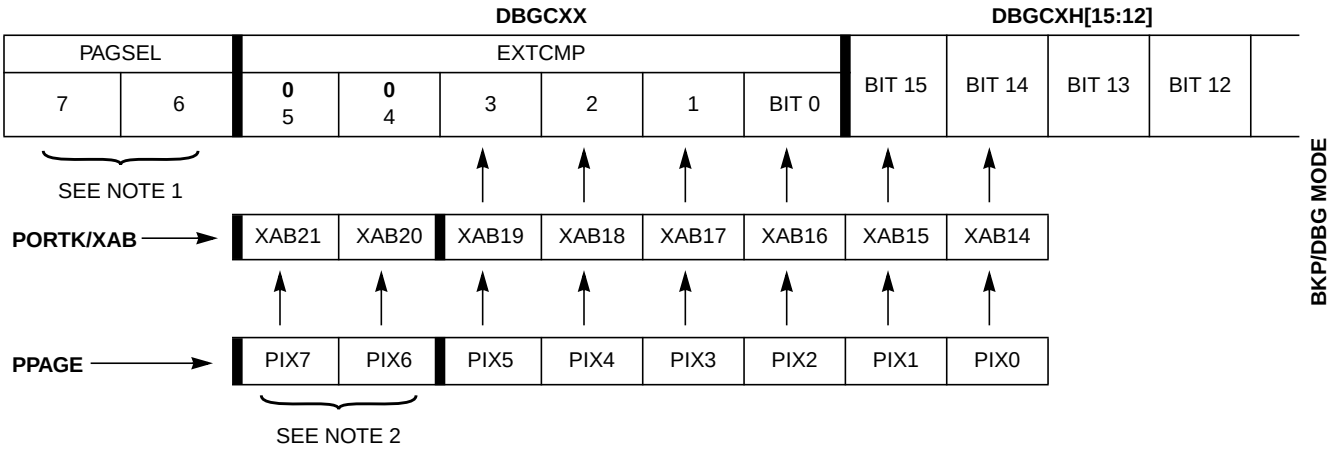
Table 16-7. Tag Pin Function

$\overline{\text{TAGHI}}$	$\overline{\text{TAGLO}}$	Tag
1	1	No tag
1	0	Low byte
0	1	High byte
0	0	Both bytes

16.4.12 Serial Communication Time-Out

The host initiates a host-to-target serial transmission by generating a falling edge on the BKGD pin. If BKGD is kept low for more than 128 target clock cycles, the target understands that a SYNC command was issued. In this case, the target will keep waiting for a rising edge on BKGD in order to answer the SYNC request pulse. If the rising edge is not detected, the target will keep waiting forever without any time-out limit.

Consider now the case where the host returns BKGD to logic one before 128 cycles. This is interpreted as a valid bit transmission, and not as a SYNC request. The target will keep waiting for another falling edge marking the start of a new bit. If, however, a new falling edge is not detected by the target within 512 clock cycles since the last falling edge, a time-out occurs and the current command is discarded without affecting memory or the operating mode of the MCU. This is referred to as a soft-reset.



NOTES:
1. In BKP and DBG mode, PAGSEL selects the type of paging as shown in Table 17-11.
2. Current HCS12 implementations are limited to six PPAGE bits, PIX[5:0]. Therefore, EXTCMP[5:4] = 00.

Figure 17-10. Comparator C Extended Comparison in BKP/DBG Mode

17.3.2.6 Debug Comparator C Register (DBGCC)

Module Base + 0x0026
Starting address location affected by INITRG register setting.

	15	14	13	12	11	10	9	8
R	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 17-11. Debug Comparator C Register High (DBGCCH)

Module Base + 0x0027
Starting address location affected by INITRG register setting.

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

= Unimplemented or Reserved

Figure 17-12. Debug Comparator C Register Low (DBGCCL)

Table 19-7. MODE Field Descriptions

Field	Description
7:5 MOD[C:A]	**Mode Select Bits** — These bits indicate the current operating mode. If MODA = 1, then MODC, MODB, and MODA are write never. If MODC = MODA = 0, then MODC, MODB, and MODA are writable with the exception that you cannot change to or from special peripheral mode If MODC = 1, MODB = 0, and MODA = 0, then MODC is write never. MODB and MODA are write once, except that you cannot change to special peripheral mode. From normal single-chip, only normal expanded narrow and normal expanded wide modes are available. See Table 19-8 and Table 19-16.
3 IVIS	**Internal Visibility (for both read and write accesses)** — This bit determines whether internal accesses generate a bus cycle that is visible on the external bus. Normal: write once Emulation: write never Special: write anytime 0 No visibility of internal bus operations on external bus. 1 Internal bus operations are visible on external bus.
1 EMK	**Emulate Port K** Normal: write once Emulation: write never Special: write anytime 0 PORTK and DDRK are in the memory map so port K can be used for general-purpose I/O. 1 If in any expanded mode, PORTK and DDRK are removed from the memory map. In single-chip modes, PORTK and DDRK are always in the map regardless of the state of this bit. In special peripheral mode, PORTK and DDRK are never in the map regardless of the state of this bit.
0 EME	**Emulate Port E** Normal and Emulation: write never Special: write anytime 0 PORTE and DDRE are in the memory map so port E can be used for general-purpose I/O. 1 If in any expanded mode or special peripheral mode, PORTE and DDRE are removed from the memory map. Removing the registers from the map allows the user to emulate the function of these registers externally. In single-chip modes, PORTE and DDRE are always in the map regardless of the state of this bit.

A.6 NVM, Flash and EEPROM

NOTE

Unless otherwise noted the abbreviation NVM (Non Volatile Memory) is used for both Flash and EEPROM.

A.6.1 NVM Timing

The time base for all NVM program or erase operations is derived from the oscillator. A minimum oscillator frequency f_{NVMOSC} is required for performing program or erase operations. The NVM modules do not have any means to monitor the frequency and will not prevent program or erase operation at frequencies above or below the specified minimum. Attempting to program or erase the NVM modules at a lower frequency a full program or erase transition is not assured.

The Flash and EEPROM program and erase operations are timed using a clock derived from the oscillator using the FCLKDIV and ECLKDIV registers respectively. The frequency of this clock must be set within the limits specified as f_{NVMOP}.

The minimum program and erase times shown in Table A-16 are calculated for maximum f_{NVMOP} and maximum f_{bus}. The maximum times are calculated for minimum f_{NVMOP} and a f_{bus} of 2MHz.

A.6.1.1 Single Word Programming

The programming time for single word programming is dependant on the bus frequency as a well as on the frequency f_{NVMOP} and can be calculated according to the following formula.

$$t_{swpgm} = 9 \cdot \frac{1}{f_{NVMOP}} + 25 \cdot \frac{1}{f_{bus}}$$

A.6.1.2 Row Programming

Flash programming where up to 64 words in a row can be programmed consecutively by keeping the command pipeline filled. The time to program a consecutive word can be calculated as:

$$t_{bwpgm} = 4 \cdot \frac{1}{f_{NVMOP}} + 9 \cdot \frac{1}{f_{bus}}$$

The time to program a whole row is:

$$t_{brpgm} = t_{swpgm} + 63 \cdot t_{bwpgm}$$

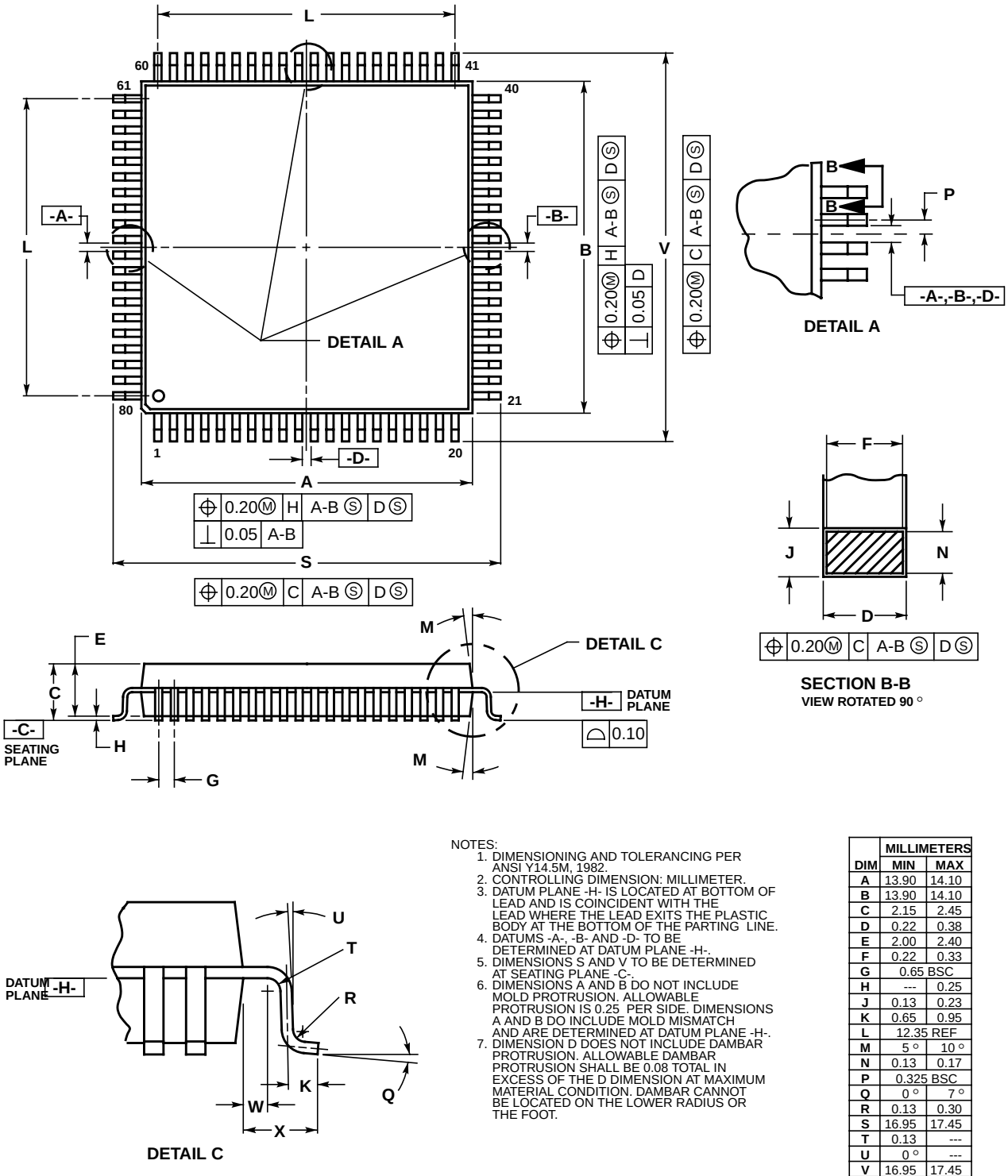
Row programming is more than 2 times faster than single word programming.

Table A-24. Expanded Bus Timing Characteristics

Conditions are shown in Table A-4 unless otherwise noted, C_{LOAD} = 50pF							
Num	**C**	**Rating**	**Symbol**	**Min**	**Typ**	**Max**	**Unit**
1	P	Frequency of operation (E-clock)	f_o	0	—	25.0	MHz
2	P	Cycle time	t_{cyc}	40	—	—	ns
3	D	Pulse width, E low	PW_{EL}	17	—	—	ns
4	D	Pulse width, E high[1]	PW_{EH}	17	—	—	ns
5	D	Address delay time	t_{AD}	—	—	8	ns
6	D	Address valid time to E rise (PW_{EL}–t_{AD})	t_{AV}	11	—	—	ns
7	D	Muxed address hold time	t_{MAH}	2	—	—	ns
8	D	Address hold to data valid	t_{AHDS}	7	—	—	ns
9	D	Data hold to address	t_{DHA}	2	—	—	ns
10	D	Read data setup time	t_{DSR}	13	—	—	ns
11	D	Read data hold time	t_{DHR}	0	—	—	ns
12	D	Write data delay time	t_{DDW}	—	—	7	ns
13	D	Write data hold time	t_{DHW}	2	—	—	ns
14	D	Write data setup time[1] (PW_{EH}–t_{DDW})	t_{DSW}	10	—	—	ns
15	D	Address access time[1] (t_{cyc}–t_{AD}–t_{DSR})	t_{ACCA}	19	—	—	ns
16	D	E high access time[1] (PW_{EH}–t_{DSR})	t_{ACCE}	4	—	—	ns
17	D	Non-multiplexed address delay time	t_{NAD}	—	—	7	ns
18	D	Non-muxed address valid to E rise (PW_{EL}–t_{NAD})	t_{NAV}	10	—	—	ns
19	D	Non-multiplexed address hold time	t_{NAH}	2	—	—	ns
20	D	Chip select delay time	t_{CSD}	—	—	16	ns
21	D	Chip select access time[1] (t_{cyc}–t_{CSD}–t_{DSR})	t_{ACCS}	11	—	—	ns
22	D	Chip select hold time	t_{CSH}	2	—	—	ns
23	D	Chip select negated time	t_{CSN}	8	—	—	ns
24	D	Read/write delay time	t_{RWD}	—	—	7	ns
25	D	Read/write valid time to E rise (PW_{EL}–t_{RWD})	t_{RWV}	10	—	—	ns
26	D	Read/write hold time	t_{RWH}	2	—	—	ns
27	D	Low strobe delay time	t_{LSD}	—	—	7	ns
28	D	Low strobe valid time to E rise (PW_{EL}–t_{LSD})	t_{LSV}	10	—	—	ns
29	D	Low strobe hold time	t_{LSH}	2	—	—	ns
30	D	NOACC strobe delay time	t_{NOD}	—	—	7	ns
31	D	NOACC valid time to E rise (PW_{EL}–t_{LSD})	t_{NOV}	10	—	—	ns
32	D	NOACC hold time	t_{NOH}	2	—	—	ns
33	D	PIPO0 delay time	t_{P0D}	2	—	7	ns
34	D	PIPO0 valid time to E rise (PW_{EL}–t_{P0D})	t_{P0V}	10	—		ns
35	D	PIPO1 delay time[1] (PW_{EH}-t_{P1V})	t_{P1D}	2	—	7	ns
36	D	PIPO1 valid time to E fall	t_{P1V}	10	—	—	ns

1 Affected by clock stretch: add N x t_{cyc} where N=0,1,2 or 3, depending on the number of clock stretches.

C.2 80-Pin QFP Package



NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: MILLIMETER.
3. DATUM PLANE -H- 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 -A-, -B- AND -D- TO BE DETERMINED AT DATUM PLANE -H-.
5. DIMENSIONS S AND V TO BE DETERMINED AT SEATING PLANE -C-.
6. DIMENSIONS A AND B DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.25 PER SIDE. DIMENSIONS A AND B DO INCLUDE MOLD MISMATCH AND ARE DETERMINED AT DATUM PLANE -H-.
7. DIMENSION D DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.08 TOTAL IN EXCESS OF THE D DIMENSION AT MAXIMUM MATERIAL CONDITION. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE FOOT.

	MILLIMETERS	
DIM	MIN	MAX
A	13.90	14.10
B	13.90	14.10
C	2.15	2.45
D	0.22	0.38
E	2.00	2.40
F	0.22	0.33
G	0.65 BSC	
H	---	0.25
J	0.13	0.23
K	0.65	0.95
L	12.35 REF	
M	5 °	10 °
N	0.13	0.17
P	0.325 BSC	
Q	0 °	7 °
R	0.13	0.30
S	16.95	17.45
T	0.13	---
U	0 °	---
V	16.95	17.45
W	0.35	0.45
X	1.6 REF	

Figure C-2. 80-Pin QFP Mechanical Dimensions (Case no. 841B)