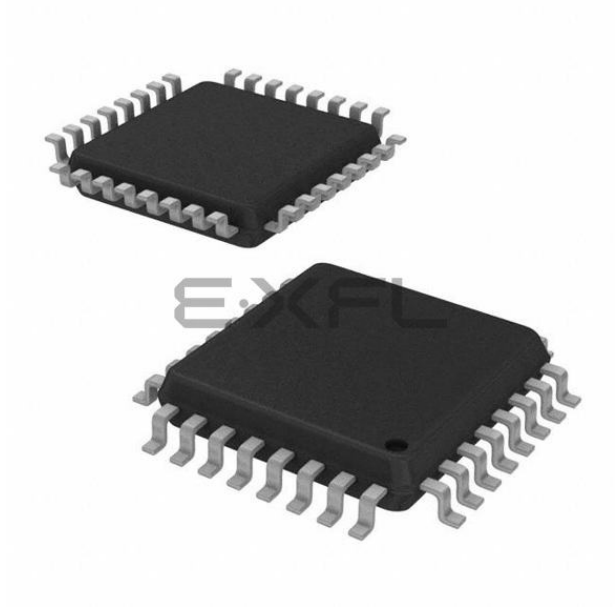

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	12V1
Core Size	16-Bit
Speed	25MHz
Connectivity	IrDA, LINbus, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	16
Program Memory Size	48KB (48K x 8)
Program Memory Type	FLASH
EEPROM Size	512 x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	3.13V ~ 5.5V
Data Converters	A/D 2x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LQFP
Supplier Device Package	32-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/s9s12vr48af0mlc

Chapter 1
Device Overview MC9S12VR-Family

Revision History

Version Number	Revision Date	Description of Changes
Rev 3.4	30-January-2014	• Corrected Figure 1-4 32LQFP pinout pin 5
Rev 4.0 Initial Draft	3-February-2014	• Added MC9S12VR16 & MC9S12VR32 to Table 1-1 • Added MC9S12VR16 & MC9S12VR32 maskset N11N to Table 1-3 • Added CPMU differences between masksets N11N & N59H
Draft A	2-April-2014	• Added 32QFN package
Draft B	24-February-2015	• Included feedback from shared review

1.1 Introduction

The MC9S12VR-Family is an optimized automotive 16-bit microcontroller product line focused on low-cost, high-performance, and low pin-count. This family integrates an S12 microcontroller with a LIN Physical interface, a 5V regulator system to supply the microcontroller, and analog blocks to control other elements of the system which operate at vehicle battery level (e.g. relay drivers, high-side driver outputs, wake up inputs). The MC9S12VR-Family is targeted at generic automotive applications requiring single node LIN communications. Typical examples of these applications include window lift modules, seat modules and sun-roof modules to name a few.

The MC9S12VR-Family uses many of the same features found on the MC9S12G family, including error correction code (ECC) on flash memory, EEPROM for diagnostic or data storage, a fast analog-to-digital converter (ADC) and a frequency modulated phase locked loop (IPLL) that improves the EMC performance. The MC9S12VR-Family delivers an optimized solution with the integration of several key system components into a single device, optimizing system architecture and achieving significant space savings. The MC9S12VR-Family delivers all the advantages and efficiencies of a 16-bit MCU while retaining the low cost, power consumption, EMC, and code-size efficiency advantages currently enjoyed by users of NXP's existing 8-bit and 16-bit MCU families. Like the MC9S12XS family, the MC9S12VR-Family runs 16-bit wide accesses without wait states for all peripherals and memories. Misaligned single cycle 16-bit RAM access is not supported. The MC9S12VR-Family is available in 32-pin and 48-pin LQFP. In addition to the I/O ports available in each module, further I/O ports are available with interrupt capability allowing wake-up from stop or wait modes.

The MC9S12VR-Family is a general-purpose family of devices created with relay based motor control in mind and is suitable for a range of applications, including:

- Window lift modules
- Door modules
- Seat controllers

CAUTION

The ADC's reference voltage V_{RH} must must remain at a constant level throughout the conversion process.

The reference voltage V_{BG} is measured under the conditions shown in Table 1-15.. The value stored in the IFR is the average of eight consecutive conversions at T_J=150°C and eight consecutive conversions at T_J=-40°C.

Table 1-15. Measurement Conditions for V_{BG} ADC Conversion[1]

Description	Symbol	Value	Unit
Regulator supply voltage	V_{SUP}	5	V
I/O supply voltage	V_{DDX}	5	V
Analog supply voltage	V_{DDA}	5	V
ADC clock	f_{adcclk}	6.25	MHz
ADC sample time	t_{smp}	4	ADC clock cycles
Bus frequency	f_{bus}	25	MHz
Junction temperature	T_j	-40 .. 150	°C
Code execution	from RAM		

1 Voltage Regulator bypassed , VDDX and VDDA supplied from tester

2.3.33 Port P Interrupt Flag Register (PIFP)

Address 0x025F (S12VR64/48) Access: User read/write[1]

	7	6	5	4	3	2	1	0
R	OCIF	0	PIFP5	PIFP4	PIFP3	PIFP2	PIFP1	PIFP0
W								
Reset	0	0	0	0	0	0	0	0

Figure 2-47. Port P Interrupt Flag Register (PIFP - S12VR64/48)

[1] Read: Anytime
Write: Anytime, write 1 to clear

Address 0x025F (S12VR32/16) Access: User read/write[1]

	7	6	5	4	3	2	1	0
R	OCIF	0	0	0	0	PIFP2	PIFP1	0
W								
Reset	0	0	0	0	0	0	0	0

Figure 2-48. Port P Interrupt Flag Register (PIFP - S12VR32/16)

[1] Read: Anytime
Write: Anytime, write 1 to clear

Table 2-32. PIFP Register Field Descriptions

Field	Description
7 OCIF	**Over-Current Interrupt Flag** register port P — This flag asserts if an over-current condition is detected on PP2 (Section 2.4.4.3, “Over-Current Interrupt and Protection”). 1 PP2 Over-current event occurred 0 No PP2 over-current event occurred
5-0 PIFP	**Pin Interrupt Flag** register port P — This flag asserts after a valid active edge was detected on the related pin (Section 2.4.4, “Interrupts”). This can be a rising or a falling edge based on the state of the polarity select register. An interrupt will occur if the associated interrupt enable bit is set. 1 Active edge on the associated bit has occurred 0 No active edge occurred

4.6.1 Description of Interrupt Operation

4.6.1.1 Real Time Interrupt (RTI)

The clock source for the RTI is either IRCCLK or OSCCLK depending on the setting of the RTIOSCSEL bit. In Stop Mode with PSTP=1 (Pseudo Stop Mode), RTIOSCSEL=1 and PRE=1 the RTI continues to run, else the RTI counter halts in Stop Mode.

The RTI can be used to generate hardware interrupts at a fixed periodic rate. If enabled (by setting RTIE=1), this interrupt will occur at the rate selected by the CPMURTI register. At the end of the RTI time-out period the RTIF flag is set to one and a new RTI time-out period starts immediately.

A write to the CPMURTI register restarts the RTI time-out period.

4.6.1.2 PLL Lock Interrupt

The S12CPMU_UHV_V8 generates a PLL Lock interrupt when the lock condition (LOCK status bit) of the PLL changes, either from a locked state to an unlocked state or vice versa. Lock interrupts are locally disabled by setting the LOCKIE bit to zero. The PLL Lock interrupt flag (LOCKIF) is set to1 when the lock condition has changed, and is cleared to 0 by writing a 1 to the LOCKIF bit.

4.6.1.3 Oscillator Status Interrupt

When the OSCE bit is 0, then UPOSC stays 0. When OSCE=1 the UPOSC bit is set after the LOCK bit is set.

Upon detection of a status change (UPOSC) the OSCIF flag is set. Going into Full Stop Mode or disabling the oscillator can also cause a status change of UPOSC.

Any change in PLL configuration or any other event which causes the PLL lock status to be cleared leads to a loss of the oscillator status information as well (UPOSC=0).

Oscillator status change interrupts are locally enabled with the OSCIE bit.

NOTE

Loosing the oscillator status (UPOSC=0) affects the clock configuration of the system[1]. This needs to be dealt with in application software.

4.6.1.4 Low-Voltage Interrupt (LVI)

In FPM the input voltage VDDA is monitored. Whenever VDDA drops below level $V_{LVIA,}$ the status bit LVDS is set to 1. When VDDA rises above level V_{LVID} the status bit LVDS is cleared to 0. An interrupt, indicated by flag LVIF = 1, is triggered by any change of the status bit LVDS if interrupt enable bit LVIE = 1.

1. For details please refer to “4.4.6 System Clock Configurations”

Table 6-28. DBGADH Field Descriptions

Field	Description
7–0 Bits[15:8]	**Comparator Data High Compare Bits**— The Comparator data high compare bits control whether the selected comparator compares the data bus bits [15:8] to a logic one or logic zero. The comparator data compare bits are only used in comparison if the corresponding data mask bit is logic 1. This register is available only for comparator A. Data bus comparisons are only performed if the TAG bit in DBGACTL is clear. 0 Compare corresponding data bit to a logic zero 1 Compare corresponding data bit to a logic one

6.3.2.8.6 Debug Comparator Data Low Register (DBGADL)

Address: 0x002D

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

Figure 6-20. Debug Comparator Data Low Register (DBGADL)

Read: If COMRV[1:0] = 00

Write: If COMRV[1:0] = 00 and DBG not armed.

Table 6-29. DBGADL Field Descriptions

Field	Description
7–0 Bits[7:0]	**Comparator Data Low Compare Bits** — The Comparator data low compare bits control whether the selected comparator compares the data bus bits [7:0] to a logic one or logic zero. The comparator data compare bits are only used in comparison if the corresponding data mask bit is logic 1. This register is available only for comparator A. Data bus comparisons are only performed if the TAG bit in DBGACTL is clear 0 Compare corresponding data bit to a logic zero 1 Compare corresponding data bit to a logic one

6.3.2.8.7 Debug Comparator Data High Mask Register (DBGADHM)

Address: 0x002E

	7	6	5	4	3	2	1	0
R W	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
Reset	0	0	0	0	0	0	0	0

Figure 6-21. Debug Comparator Data High Mask Register (DBGADHM)

Read: If COMRV[1:0] = 00

Write: If COMRV[1:0] = 00 and DBG not armed.

requested immediately. TRIG breakpoints are possible with a single write to DBGC1, setting ARM and TRIG simultaneously.

6.4.7.3 Breakpoint Priorities

If a TRIG trigger occurs after Begin aligned tracing has already started, then the TRIG no longer has an effect. When the associated tracing session is complete, the breakpoint occurs. Similarly if a TRIG is followed by a subsequent comparator channel match, it has no effect, since tracing has already started.

If a forced SWI breakpoint coincides with a BGND in user code with BDM enabled, then the BDM is activated by the BGND and the breakpoint to SWI is suppressed.

6.4.7.3.1 DBG Breakpoint Priorities And BDM Interfacing

Breakpoint operation is dependent on the state of the BDM module. If the BDM module is active, the CPU is executing out of BDM firmware, thus comparator matches and associated breakpoints are disabled. In addition, while executing a BDM TRACE command, tagging into BDM is disabled. If BDM is not active, the breakpoint gives priority to BDM requests over SWI requests if the breakpoint happens to coincide with a SWI instruction in user code. On returning from BDM, the SWI from user code gets executed.

Table 6-43. Breakpoint Mapping Summary

DBGBRK	BDM Bit (DBGC1[4])	BDM Enabled	BDM Active	Breakpoint Mapping
0	X	X	X	No Breakpoint
1	0	X	0	Breakpoint to SWI
X	X	1	1	No Breakpoint
1	1	0	X	Breakpoint to SWI
1	1	1	0	Breakpoint to BDM

BDM cannot be entered from a breakpoint unless the ENABLE bit is set in the BDM. If entry to BDM via a BGND instruction is attempted and the ENABLE bit in the BDM is cleared, the CPU actually executes the BDM firmware code, checks the ENABLE and returns if ENABLE is not set. If not serviced by the monitor then the breakpoint is re-asserted when the BDM returns to normal CPU flow.

If the comparator register contents coincide with the SWI/BDM vector address then an SWI in user code could coincide with a DBG breakpoint. The CPU ensures that BDM requests have a higher priority than SWI requests. Returning from the BDM/SWI service routine care must be taken to avoid a repeated breakpoint at the same address.

Should a tagged or forced breakpoint coincide with a BGND in user code, then the instruction that follows the BGND instruction is the first instruction executed when normal program execution resumes.

Table 7-3. IVBR Field Descriptions

Field	Description
7–0 IVB_ADDR[7:0]	**Interrupt Vector Base Address Bits** — These bits represent the upper byte of all vector addresses. Out of reset these bits are set to 0xFF (that means vectors are located at 0xFF80–0xFFFE) to ensure compatibility to HCS12. **Note:** A system reset will initialize the interrupt vector base register with "0xFF" before it is used to determine the reset vector address. Therefore, changing the IVBR has no effect on the location of the three reset vectors (0xFFFA–0xFFFE). **Note:** If the BDM is active (that means the CPU is in the process of executing BDM firmware code), the contents of IVBR are ignored and the upper byte of the vector address is fixed as "0xFF". This is done to enable handling of all non-maskable interrupts in the BDM firmware.

7.4 Functional Description

The INT module processes all exception requests to be serviced by the CPU module. These exceptions include interrupt vector requests and reset vector requests. Each of these exception types and their overall priority level is discussed in the subsections below.

7.4.1 S12S Exception Requests

The CPU handles both reset requests and interrupt requests. A priority decoder is used to evaluate the priority of pending interrupt requests.

7.4.2 Interrupt Prioritization

The INT module contains a priority decoder to determine the priority for all interrupt requests pending for the CPU. If more than one interrupt request is pending, the interrupt request with the higher vector address wins the prioritization.

The following conditions must be met for an I bit maskable interrupt request to be processed.

1. The local interrupt enabled bit in the peripheral module must be set.
2. The I bit in the condition code register (CCR) of the CPU must be cleared.
3. There is no SWI, TRAP, or X bit maskable request pending.

NOTE

All non I bit maskable interrupt requests always have higher priority than the I bit maskable interrupt requests. If the X bit in the CCR is cleared, it is possible to interrupt an I bit maskable interrupt by an X bit maskable interrupt. It is possible to nest non maskable interrupt requests, for example by nesting SWI or TRAP calls.

Since an interrupt vector is only supplied at the time when the CPU requests it, it is possible that a higher priority interrupt request could override the original interrupt request that caused the CPU to request the vector. In this case, the CPU will receive the highest priority vector and the system will process this interrupt request first, before the original interrupt request is processed.

7.5 Initialization/Application Information

7.5.1 Initialization

After system reset, software should:

1. Initialize the interrupt vector base register if the interrupt vector table is not located at the default location (0xFF80–0xFFF9).
2. Enable I bit maskable interrupts by clearing the I bit in the CCR.
3. Enable the X bit maskable interrupt by clearing the X bit in the CCR.

7.5.2 Interrupt Nesting

The interrupt request scheme makes it possible to nest I bit maskable interrupt requests handled by the CPU.

- I bit maskable interrupt requests can be interrupted by an interrupt request with a higher priority.

I bit maskable interrupt requests cannot be interrupted by other I bit maskable interrupt requests per default. In order to make an interrupt service routine (ISR) interruptible, the ISR must explicitly clear the I bit in the CCR (CLI). After clearing the I bit, other I bit maskable interrupt requests can interrupt the current ISR.

An ISR of an interruptible I bit maskable interrupt request could basically look like this:

1. Service interrupt, that is clear interrupt flags, copy data, etc.
2. Clear I bit in the CCR by executing the instruction CLI (thus allowing other I bit maskable interrupt requests)
3. Process data
4. Return from interrupt by executing the instruction RTI

7.5.3 Wake Up from Stop or Wait Mode

7.5.3.1 CPU Wake Up from Stop or Wait Mode

Every I bit maskable interrupt request is capable of waking the MCU from wait mode.

Since bus and core clocks are disabled in stop mode, only interrupt requests that can be generated without these clocks can wake the MCU from stop mode. These are listed in the device overview interrupt vector table.

To determine whether an I bit maskable interrupts is qualified to wake-up the CPU or not, the same conditions as in normal run mode are applied during stop or wait mode:

- If the I bit in the CCR is set, all I bit maskable interrupts are masked from waking-up the MCU.

The X bit maskable interrupt request can wake up the MCU from stop or wait mode at anytime, even if the X bit in CCR is set[1].

8.3.2.9 ATD Status Register 2 (ATDSTAT2)

This read-only register contains the Conversion Complete Flags CCF[5:0].

Module Base + 0x000A

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	0	0	0	CCF[5:0]					
W																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 8-11. ATD Status Register 2 (ATDSTAT2)

Read: Anytime

Write: Anytime (for details see Table 8-18 below)

Table 8-18. ATDSTAT2 Field Descriptions

Field	Description
5–0 CCF[5:0]	**Conversion Complete Flag *n* (*n*= 5, 4, 3, 2, 1, 0) (*n conversion number, NOT channel number!)***— A conversion complete flag is set at the end of each conversion in a sequence. The flags are associated with the conversion position in a sequence (and also the result register number). Therefore in non-fifo mode, CCF[4] is set when the fifth conversion in a sequence is complete and the result is available in result register ATDDR4; CCF[5] is set when the sixth conversion in a sequence is complete and the result is available in ATDDR5, and so forth. If automatic compare of conversion results is enabled (CMPE[*n*]=1 in ATDCMPE), the conversion complete flag is only set if comparison with ATDDR*n* is true. If ACMPIE=1 a compare interrupt will be requested. In this case, as the ATDDR*n* result register is used to hold the compare value, the result will not be stored there at the end of the conversion but is lost. A flag CCF[*n*] is cleared when one of the following occurs: A) Write to ATDCTL5 (a new conversion sequence is started) B) If AFFC=0, write “1” to CCF[*n*] C) If AFFC=1 and CMPE[*n*]=0, read of result register ATDDR*n* D) If AFFC=1 and CMPE[*n*]=1, write to result register ATDDR*n* In case of a concurrent set and clear on CCF[*n*]: The clearing by method A) will overwrite the set. The clearing by methods B) or C) or D) will be overwritten by the set. 0 Conversion number *n* not completed or successfully compared 1 If (CMPE[*n*]=0): Conversion number *n* has completed. Result is ready in ATDDR*n*. If (CMPE[*n*]=1): Compare for conversion result number *n* with compare value in ATDDR*n*, using compare operator CMPGT[*n*] is true. (No result available in ATDDR*n*)

On the front end of the PWM timer, the clock is enabled to the PWM circuit by the PWMEx bit being high. There is an edge-synchronizing circuit to guarantee that the clock will only be enabled or disabled at an edge. When the channel is disabled (PWMEx = 0), the counter for the channel does not count.

9.4.2.2 PWM Polarity

Each channel has a polarity bit to allow starting a waveform cycle with a high or low signal. This is shown on the block diagram Figure 9-16 as a mux select of either the Q output or the $\overline{Q}$ output of the PWM output flip flop. When one of the bits in the PWMPOL register is set, the associated PWM channel output is high at the beginning of the waveform, then goes low when the duty count is reached. Conversely, if the polarity bit is zero, the output starts low and then goes high when the duty count is reached.

9.4.2.3 PWM Period and Duty

Dedicated period and duty registers exist for each channel and are double buffered so that if they change while the channel is enabled, the change will NOT take effect until one of the following occurs:

- The effective period ends
- The counter is written (counter resets to $00)
- The channel is disabled

In this way, the output of the PWM will always be either the old waveform or the new waveform, not some variation in between. If the channel is not enabled, then writes to the period and duty registers will go directly to the latches as well as the buffer.

A change in duty or period can be forced into effect “immediately” by writing the new value to the duty and/or period registers and then writing to the counter. This forces the counter to reset and the new duty and/or period values to be latched. In addition, since the counter is readable, it is possible to know where the count is with respect to the duty value and software can be used to make adjustments

NOTE

When forcing a new period or duty into effect immediately, an irregular PWM cycle can occur.

Depending on the polarity bit, the duty registers will contain the count of either the high time or the low time.

9.4.2.4 PWM Timer Counters

Each channel has a dedicated 8-bit up/down counter which runs at the rate of the selected clock source (see Section 9.4.1, “PWM Clock Select” for the available clock sources and rates). The counter compares to two registers, a duty register and a period register as shown in Figure 9-16. When the PWM counter matches the duty register, the output flip-flop changes state, causing the PWM waveform to also change state. A match between the PWM counter and the period register behaves differently depending on what output mode is selected as shown in Figure 9-16 and described in Section 9.4.2.5, “Left Aligned Outputs” and Section 9.4.2.6, “Center Aligned Outputs”.

Figure 10-17 shows two cases of break detect. In trace RXD_1 the break symbol starts with the start bit, while in RXD_2 the break starts in the middle of a transmission. If BRKDFE = 1, in RXD_1 case there will be no byte transferred to the receive buffer and the RDRF flag will not be modified. Also no framing error or parity error will be flagged from this transfer. In RXD_2 case, however the break signal starts later during the transmission. At the expected stop bit position the byte received so far will be transferred to the receive buffer, the receive data register full flag will be set, a framing error and if enabled and appropriate a parity error will be set. Once the break is detected the BRKDIF flag will be set.

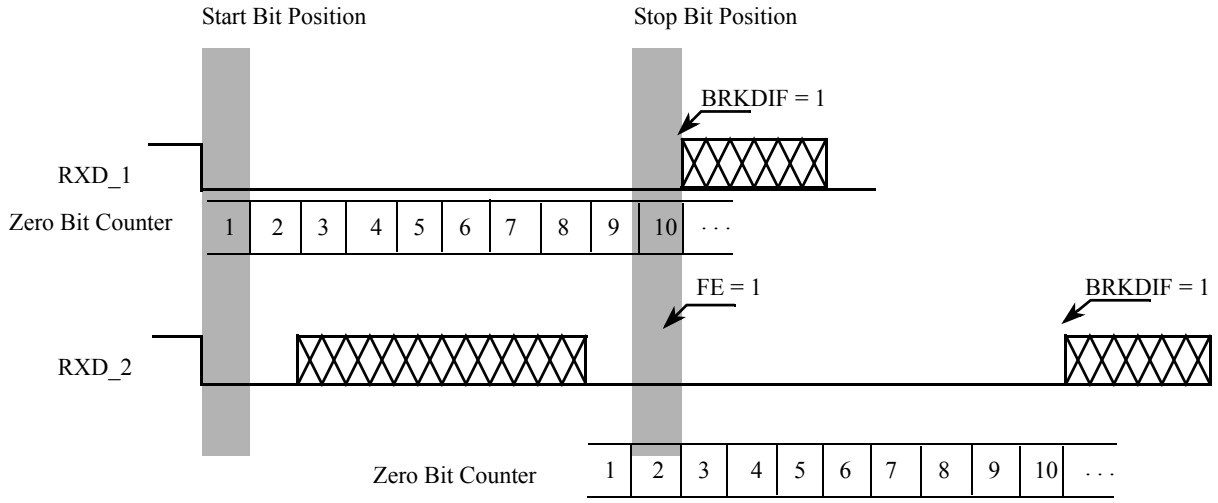

Figure 10-17. Break Detection if BRKDFE = 1 (M = 0)

10.4.5.4 Idle Characters

An idle character (or preamble) contains all logic 1s and has no start, stop, or parity bit. Idle character length depends on the M bit in SCI control register 1 (SCICR1). The preamble is a synchronizing idle character that begins the first transmission initiated after writing the TE bit from 0 to 1.

If the TE bit is cleared during a transmission, the TXD pin becomes idle after completion of the transmission in progress. Clearing and then setting the TE bit during a transmission queues an idle character to be sent after the frame currently being transmitted.

NOTE

When queueing an idle character, return the TE bit to logic 1 before the stop bit of the current frame shifts out through the TXD pin. Setting TE after the stop bit appears on TXD causes data previously written to the SCI data register to be lost. Toggle the TE bit for a queued idle character while the TDRE flag is set and immediately before writing the next byte to the SCI data register.

If the TE bit is clear and the transmission is complete, the SCI is not the master of the TXD pin

10.4.6.5 Baud Rate Tolerance

A transmitting device may be operating at a baud rate below or above the receiver baud rate. Accumulated bit time misalignment can cause one of the three stop bit data samples (RT8, RT9, and RT10) to fall outside the actual stop bit. A noise error will occur if the RT8, RT9, and RT10 samples are not all the same logical values. A framing error will occur if the receiver clock is misaligned in such a way that the majority of the RT8, RT9, and RT10 stop bit samples are a logic zero.

As the receiver samples an incoming frame, it re-synchronizes the RT clock on any valid falling edge within the frame. Re synchronization within frames will correct a misalignment between transmitter bit times and receiver bit times.

10.4.6.5.1 Slow Data Tolerance

Figure 10-28 shows how much a slow received frame can be misaligned without causing a noise error or a framing error. The slow stop bit begins at RT8 instead of RT1 but arrives in time for the stop bit data samples at RT8, RT9, and RT10.

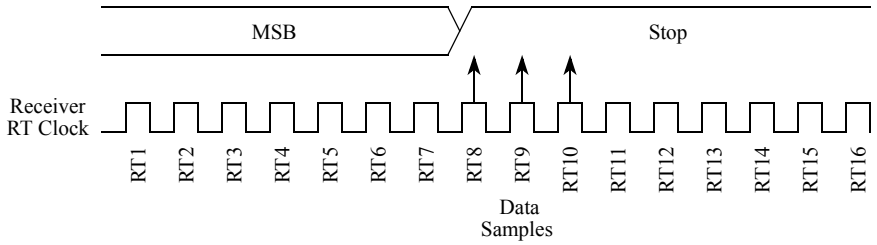

Figure 10-28. Slow Data

Let’s take RTr as receiver RT clock and RTt as transmitter RT clock.

For an 8-bit data character, it takes the receiver 9 bit times x 16 RTr cycles +7 RTr cycles = 151 RTr cycles to start data sampling of the stop bit.

With the misaligned character shown in Figure 10-28, the receiver counts 151 RTr cycles at the point when the count of the transmitting device is 9 bit times x 16 RTt cycles = 144 RTt cycles.

The maximum percent difference between the receiver count and the transmitter count of a slow 8-bit data character with no errors is:

((151 – 144) / 151) x 100 = 4.63%

For a 9-bit data character, it takes the receiver 10 bit times x 16 RTr cycles + 7 RTr cycles = 167 RTr cycles to start data sampling of the stop bit.

With the misaligned character shown in Figure 10-28, the receiver counts 167 RTr cycles at the point when the count of the transmitting device is 10 bit times x 16 RTt cycles = 160 RTt cycles.

The maximum percent difference between the receiver count and the transmitter count of a slow 9-bit character with no errors is:

((167 – 160) / 167) X 100 = 4.19%

NOTE

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

12.3.2.10 Main Timer Interrupt Flag 1 (TFLG1)

	7	6	5	4	3	2	1	0
R W	RESERVED	RESERVED	RESERVED	RESERVED	C3F	C2F	C1F	C0F
Reset	0	0	0	0	0	0	0	0

Figure 12-16. Main Timer Interrupt Flag 1 (TFLG1)

Read: Anytime

Write: Used in the clearing mechanism (set bits cause corresponding bits to be cleared). Writing a zero will not affect current status of the bit.

Table 12-13. TRLG1 Field Descriptions

Note: Writing to unavailable bits has no effect. Reading from unavailable bits return a zero.

Field	Description
3:0 C[3:0]F	**Input Capture/Output Compare Channel "x" Flag** — These flags are set when an input capture or output compare event occurs. Clearing requires writing a one to the corresponding flag bit while TEN is set to one. **Note:** When TFFCA bit in TSCR register is set, a read from an input capture or a write into an output compare channel (0x0010–0x001F) will cause the corresponding channel flag CxF to be cleared.

12.3.2.11 Main Timer Interrupt Flag 2 (TFLG2)

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

Unimplemented or Reserved

Figure 12-17. Main Timer Interrupt Flag 2 (TFLG2)

TFLG2 indicates when interrupt conditions have occurred. To clear a bit in the flag register, write the bit to one while TEN bit of TSCR1 .

Read: Anytime

Write: Used in clearing mechanism (set bits cause corresponding bits to be cleared).

Any access to TCNT will clear TFLG2 register if the TFFCA bit in TSCR register is set.

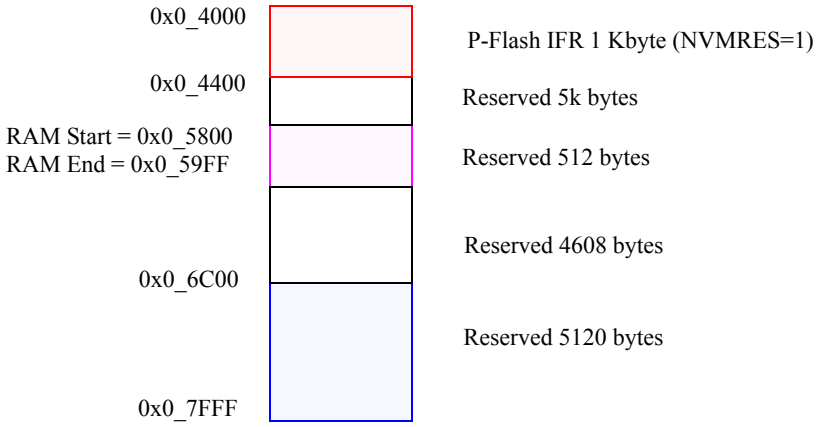

Figure 19-3. Memory Controller Resource Memory Map (NVMRES=1)

19.3.2 Register Descriptions

The Flash module contains a set of 20 control and status registers located between Flash module base + 0x0000 and 0x0013.

In the case of the writable registers, the write accesses are forbidden during Fash command execution (for more detail, see Caution note in Section 19.3 Memory Map and Registers).

A summary of the Flash module registers is given in Figure 19-4 with detailed descriptions in the following subsections.

Address & Name		7	6	5	4	3	2	1	0
FCLKDIV	R	FDIVLD	FDIVLCK	FDIV5	FDIV4	FDIV3	FDIV2	FDIV1	FDIV0
	W								
FSEC	R	KEYEN1	KEYEN0	RNV5	RNV4	RNV3	RNV2	SEC1	SEC0
	W								
FCCOBIX	R	0	0	0	0	0	CCOBIX2	CCOBIX1	CCOBIX0
	W								

Figure 19-4. FTMRG32K128 Register Summary

CCIE, IGNSF, FDFD, and FSFD bits are readable and writable while remaining bits read 0 and are not writable.

Table 19-13. FCNFG Field Descriptions

Field	Description
7 CCIE	**Command Complete Interrupt Enable** — The CCIE bit controls interrupt generation when a Flash command has completed. 0 Command complete interrupt disabled 1 An interrupt will be requested whenever the CCIF flag in the FSTAT register is set (see Section 19.3.2.7 Flash Status Register (FSTAT))
4 IGNSF	**Ignore Single Bit Fault** — The IGNSF controls single bit fault reporting in the FERSTAT register (see Section 19.3.2.8 Flash Error Status Register (FERSTAT)). 0 All single bit faults detected during array reads are reported 1 Single bit faults detected during array reads are not reported and the single bit fault interrupt will not be generated
1 FDFD	**Force Double Bit Fault Detect** — The FDFD bit allows the user to simulate a double bit fault during Flash array read operations and check the associated interrupt routine. The FDFD bit is cleared by writing a 0 to FDFD. 0 Flash array read operations will set the DFDIF flag in the FERSTAT register only if a double bit fault is detected 1 Any Flash array read operation will force the DFDIF flag in the FERSTAT register to be set (see Section 19.3.2.7 Flash Status Register (FSTAT)) and an interrupt will be generated as long as the DFDIE interrupt enable in the FERCNFG register is set (see Section 19.3.2.6 Flash Error Configuration Register (FERCNFG))
0 FSFD	**Force Single Bit Fault Detect** — The FSFD bit allows the user to simulate a single bit fault during Flash array read operations and check the associated interrupt routine. The FSFD bit is cleared by writing a 0 to FSFD. 0 Flash array read operations will set the SFDIF flag in the FERSTAT register only if a single bit fault is detected 1 Flash array read operation will force the SFDIF flag in the FERSTAT register to be set (see Section 19.3.2.7 Flash Status Register (FSTAT)) and an interrupt will be generated as long as the SFDIE interrupt enable in the FERCNFG register is set (see Section 19.3.2.6 Flash Error Configuration Register (FERCNFG))

19.3.2.6 Flash Error Configuration Register (FERCNFG)

The FERCNFG register enables the Flash error interrupts for the FERSTAT flags.

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

= Unimplemented or Reserved

Figure 19-10. Flash Error Configuration Register (FERCNFG)

All assigned bits in the FERCNFG register are readable and writable.

Upon clearing CCIF to launch the Set Field Margin Level command, the Memory Controller will set the

Table 19-57. Set Field Margin Level Command FCCOB Requirements

CCOBIX[2:0]	FCCOB Parameters	
000	0x0E	Flash block selection code [1:0]. See Table 19-34
001	Margin level setting.	

field margin level for the targeted block and then set the CCIF flag.

NOTE

When the EEPROM block is targeted, the EEPROM field margin levels are applied only to the EEPROM reads. However, when the P-Flash block is targeted, the P-Flash field margin levels are applied to both P-Flash and EEPROM reads. It is not possible to apply field margin levels to the P-Flash block only.

Valid margin level settings for the Set Field Margin Level command are defined in Table 19-58.

Table 19-58. Valid Set Field Margin Level Settings

CCOB (CCOBIX=001)	Level Description
0x0000	Return to Normal Level
0x0001	User Margin-1 Level[1]
0x0002	User Margin-0 Level[2]
0x0003	Field Margin-1 Level[1]
0x0004	Field Margin-0 Level[2]

1 Read margin to the erased state

2 Read margin to the programmed state

Table 19-59. Set Field Margin Level 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 19-27)
		Set if an invalid FlashBlockSelectionCode[1:0] is supplied (See Table 19-34)
		Set if an invalid margin level setting is supplied
	FPVIOL	None
	MGSTAT1	None
	MGSTAT0	None

Supply voltage V_{DDA} =5.12 V, -40°C < T_J < 150°C. V_{REF} = V_{RH} - V_{RL} = 5.12V. f_{ATDCLK} = 8.0MHz The values are tested to be valid with no PortAD output drivers switching simultaneous with conversions.							
Num	**Rating**		**Symbol**	**Min**	**Typ**	**Max**	**Unit**
1	Resolution	10-Bit	LSB		5		mV
2	Differential Nonlinearity	10-Bit	DNL	-1	±0.5	1	counts
3	Integral Nonlinearity	10-Bit	INL	-2	±1	2	counts
4	Absolute Error	10-Bit	AE	-3	±2	3	counts
5	Resolution	8-Bit	LSB		20		mV
6	Differential Nonlinearity	8-Bit	DNL	-0.5	±0.3	0.5	counts
7	Integral Nonlinearity	8-Bit	INL	-1	±0.5	1	counts
8	Absolute Error	8-Bit	AE	-1.5	±1	1.5	counts

Appendix D
HSDRV Electrical Specifications

This section provides electrical parametric and ratings for the S12HSDRV1CV3 on S12VR32 and S12HSDRV2 on S12VR64. The open-load detection feature is only available on S12VR32. The ratings below which refer to open-load detection feature are only valid for S12VR32.

D.1 Operating Characteristics

Table D-1. Operating Characteristics - HSDRV

Num	Ratings	Symbol	Min	Typ	Max	Unit
1	High Voltage Supply for the high-side drivers.	V_{SUPHS}	7	–	42	V
2	VSUP_HS in case of being connected to VDDX	V_{SUPHS_X}	4.5	–	5.5	V

D.2 Static Characteristics

Table D-2. Static Characteristics - HSDRV (Junction Temperature From -40°C To +150°C)

Characteristics noted under conditions 7V ≤ VSUPHS ≤ 18 V unless otherwise noted. Typical values noted reflect the approximate parameter mean at T_J = 25°C[1] under nominal conditions unless otherwise noted.

Num	Ratings	Symbol	Min	Typ	Max	Unit
1	Output Drain-to-Source On Resistance T_J = 150°C, $I_{PHS0/1}$ = -50 mA	$R_{DS(ON)}$	–	–	18.0	Ω
2	Over-current Threshold. The threshold is valid for each HS-driver output. **Note:** **The high-side driver is NOT intended to switch capacitive loads. A significant capacitive load on HS0/1 would induce a current when the high-side driver gate is turned on. This current will be sensed by the over-current circuitry and eventually lead to an immediate over-current shut down. In such cases of capacitive loads you can leverage the over current masking feature or handle it by software.**	I_{OCTHSX}	90	120	150	mA
3	Nominal Current for continuous operation. This value is valid for each HS-driver output.	I_{NOMHSX}	–	–	50	mA

NOTES:

1. DIMENSIONS ARE IN MILLIMETERS.

2. INTERPRET DIMENSIONS AND TOLERANCES PER ASME Y14.5–1994.

3. DATUMS A, B, AND D TO BE DETERMINED AT DATUM PLANE H.

4. DIMENSIONS TO BE DETERMINED AT SEATING PLANE DATUM C.

5. DIMENSION DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL NOT CAUSE THE LEAD WIDTH TO EXCEED THE MAXIMUM DIMENSION BY MORE THAN 0.08 MM. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE FOOT. MINIMUM SPACE BETWEEN PROTRUSION AND ADJACENT LEAD OR PROTRUSION: 0.07 MM.

6. DIMENSIONS DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.25 MM PER SIDE. DIMENSIONS ARE MAXIMUM PLASTIC BODY SIZE DIMENSIONS INCLUDING MOLD MISMATCH.

7. EXACT SHAPE OF EACH CORNER IS OPTIONAL.

8. THESE DIMENSIONS APPLY TO THE FLAT SECTION OF THE LEAD BETWEEN 0.1 MM AND 0.25 MM FROM THE LEAD TIP.

MECHANICAL OUTLINE	PRINT VERSION NOT TO SCALE

TITLE:	DOCUMENT NO: 98ASH70029A	REV: D
LOW PROFILE QUAD FLAT PACK (LQFP) 32 LEAD, 0.8 PITCH (7 X 7 X 1.4)	CASE NUMBER: 873A-03	19 MAY 2005
	STANDARD: JEDEC MS-026 BBA	