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	HC08
Core Size	8-Bit
Speed	8MHz
Connectivity	I²C, IRSCI, SPI
Peripherals	LCD, LVD, POR, PWM
Number of I/O	48
Program Memory Size	24KB (24K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	768 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 6x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	80-LQFP
Supplier Device Package	80-FQFP (12x12)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc908lj24cpke

MC68HC908LJ24
MC68HC908LK24

Data Sheet

To provide the most up-to-date information, the revision of our documents on the World Wide Web will be the most current. Your printed copy may be an earlier revision. To verify you have the latest information available, refer to:

http://freescale.com

The following revision history table summarizes changes contained in this document. For your convenience, the page number designators have been linked to the appropriate location.

Section 21. Computer Operating Properly (COP)

Section 22. Low-Voltage Inhibit (LVI)

Figure	Title	Page

- Specific features of the MC68HC908LJ24 in 64-pin packages are:
  - 40 general-purpose I/Os only
  - High current 15-mA sink capability on 22 pins
  - 4/3 backplanes and static with maximum 26 or 27 frontplanes LCD driver

Features of the CPU08 include the following:

- Enhanced HC05 programming model
- Extensive loop control functions
- 16 addressing modes (eight more than the HC05)
- 16-bit Index register and stack pointer
- Memory-to-memory data transfers
- Fast 8 × 8 multiply instruction
- Fast 16/8 divide instruction
- Binary-coded decimal (BCD) instructions
- Optimization for controller applications
- Efficient C language support

1.4 MCU Block Diagram

Figure 1-1 shows the structure of the MC68HC908LJ24.

NOTE: *The location of the stack is arbitrary and may be relocated anywhere in RAM. Moving the SP out of page 0 ($0000 to $00FF) frees direct address (page 0) space. For correct operation, the stack pointer must point only to RAM locations.*

6.4.4 Program Counter

The program counter is a 16-bit register that contains the address of the next instruction or operand to be fetched.

Normally, the program counter automatically increments to the next sequential memory location every time an instruction or operand is fetched. Jump, branch, and interrupt operations load the program counter with an address other than that of the next sequential location.

During reset, the program counter is loaded with the reset vector address located at $FFFE and $FFFF. The vector address is the address of the first instruction to be executed after exiting the reset state.

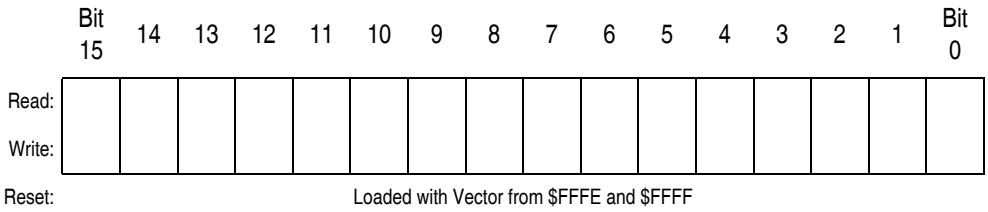

Figure 6-5. Program Counter (PC)

6.4.5 Condition Code Register

The 8-bit condition code register contains the interrupt mask and five flags that indicate the results of the instruction just executed. Bits 6 and 5 are set permanently to logic 1. The following paragraphs describe the functions of the condition code register.

	Bit 7	6	5	4	3	2	1	Bit 0
Read: Write:	V	1	1	H	I	N	Z	C
Reset:	X	1	1	X	1	X	X	X

X = Indeterminate

Figure 6-6. Condition Code Register (CCR)

V — Overflow Flag

The CPU sets the overflow flag when a two's complement overflow occurs. The signed branch instructions BGT, BGE, BLE, and BLT use the overflow flag.

1 = Overflow
0 = No overflow

H — Half-Carry Flag

The CPU sets the half-carry flag when a carry occurs between accumulator bits 3 and 4 during an add-without-carry (ADD) or add-with-carry (ADC) operation. The half-carry flag is required for binary-coded decimal (BCD) arithmetic operations. The DAA instruction uses the states of the H and C flags to determine the appropriate correction factor.

1 = Carry between bits 3 and 4
0 = No carry between bits 3 and 4

9.8.2 SIM Reset Status Register

This register contains six flags that show the source of the last reset provided all previous reset status bits have been cleared. Clear the SIM reset status register by reading it. A power-on reset sets the POR bit and clears all other bits in the register.

Address: $FE01

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	POR	PIN	COP	ILOP	ILAD	0	LVI	0
Write:								
Reset:	1	0	0	0	0	0	0	0

= Unimplemented

Figure 9-21. SIM Reset Status Register (SRSR)

POR — Power-On Reset Bit
1 = Last reset caused by POR circuit
0 = Read of SRSR

PIN — External Reset Bit
1 = Last reset caused by external reset pin ($\overline{\text{RST}}$)
0 = POR or read of SRSR

COP — Computer Operating Properly Reset Bit
1 = Last reset caused by COP counter
0 = POR or read of SRSR

ILOP — Illegal Opcode Reset Bit
1 = Last reset caused by an illegal opcode
0 = POR or read of SRSR

ILAD — Illegal Address Reset Bit (opcode fetches only)
1 = Last reset caused by an opcode fetch from an illegal address
0 = POR or read of SRSR

LVI — Low-Voltage Inhibit Reset Bit
1 = Last reset caused by the LVI circuit
0 = POR or read of SRSR

Section 10. Monitor ROM (MON)

10.1 Contents

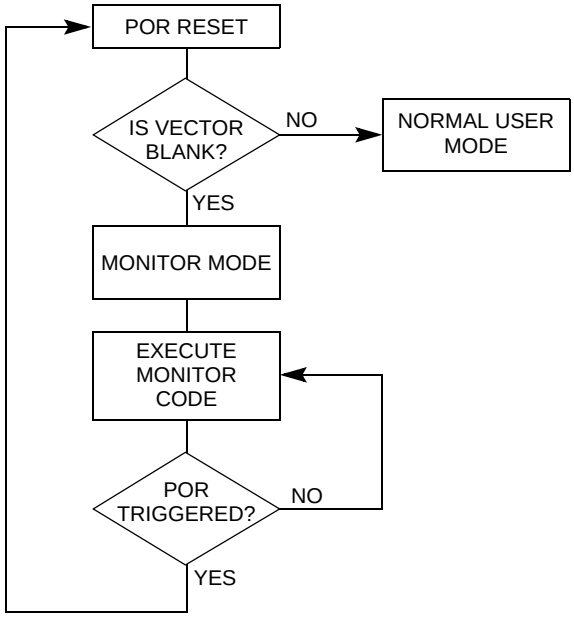

Figure 10-2. Low-Voltage Monitor Mode Entry Flowchart

In monitor mode, the MCU uses different vectors for reset, SWI (software interrupt), and break interrupt than those for user mode. The alternate vectors are in the $FE page instead of the $FF page and allow code execution from the internal monitor firmware instead of user code.

NOTE: *Exiting monitor mode after it has been initiated by having a blank reset vector requires a power-on reset (POR). Pulling $\overline{RST}$ low will not exit monitor mode in this situation.*

Table 10-2 summarizes the differences between user mode and monitor mode vectors.

Table 10-2. Mode Differences (Vectors)

Modes	Functions					
	Reset Vector High	Reset Vector Low	Break Vector High	Break Vector Low	SWI Vector High	SWI Vector Low
User	$FFFE	$FFFF	$FFFC	$FFFD	$FFFC	$FFFD
Monitor	$FEFE	$FEFF	$FEFC	$FEFD	$FEFC	$FEFD

10.6.6 MON_LDRNGE

In monitor mode, LDRNGE is used to load the data array in RAM with data from a range of FLASH locations.

Table 10-16. ICP_LDRNGE Routine

Routine Name	MON_LDRNGE
Routine Description	Loads data from a range of locations, in monitor mode
Calling Address	$FF24
Stack Used	11 bytes
Data Block Format	Bus speed Data size Starting address (high byte) Starting address (low byte) Data 1 : Data N

The MON_LDRNGE routine is designed to be used in monitor mode. It performs the same function as the LDRNGE routine (see **10.6.3 LDRNGE**), except that MON_LDRNGE returns to the main program via an SWI instruction. After a MON_LDRNGE call, the SWI instruction will return the control back to the monitor code.

12.10.9 Hour Register (HRR)

This read/write register contains the current value of the hour counter. This register can be read at any time without affecting the counter count. Writing to this register loads the value to the hour counter and the counter continues to count from this new value.

The hour counter rolls over to 0 ($00) after reaching 23 ($17). Writing a value other than 0 to 23 to this register has no effect.

Address: $0049

Read:	0	0	0	HR4	HR3	HR2	HR1	HR0
Write:								
Reset:	0	0	0	U	U	U	U	U

= Unimplemented

This register is write-protected; unprotect by a write sequence to RTCWE[1:0] in RTCCOMR.

Figure 12-15. Hour Register (HRR)

12.10.10 Day Register (DAYR)

This read/write register contains the current value of the day-of-month counter. This register can be read at any time without affecting the counter count. Writing to this register loads the value to the day counter and the counter continues to count from this new value.

The day counter rolls over to 1 ($01) after reaching 28 ($1B), 29 ($1C), 30 ($1D), or 31 ($1E), depending on the value in the month and year registers. Writing a value that is not valid for the month and year to this register has no effect.

Address: $004A

Read:	0	0	0	DAY4	DAY3	DAY2	DAY1	DAY0
Write:								
Reset:	0	0	0	U	U	U	U	U

= Unimplemented

This register is write-protected; unprotect by a write sequence to RTCWE[1:0] in RTCCOMR.

Figure 12-16. Day Register (DAYR)

13.7.2.3 Break Characters

Writing a logic 1 to the send break bit, SBK, in SCC2 loads the transmit shift register with a break character. A break character contains all logic 0s and has no start, stop, or parity bit. Break character length depends on the M bit in SCC1. As long as SBK is at logic 1, transmitter logic continuously loads break characters into the transmit shift register. After software clears the SBK bit, the shift register finishes transmitting the last break character and then transmits at least one logic 1. The automatic logic 1 at the end of a break character guarantees the recognition of the start bit of the next character.

The SCI recognizes a break character when a start bit is followed by eight or nine logic 0 data bits and a logic 0 where the stop bit should be.

Receiving a break character has the following effects on SCI registers:

- Sets the framing error bit (FE) in SCS1
- Sets the SCI receiver full bit (SCRF) in SCS1
- Clears the SCI data register (SCDR)
- Clears the R8 bit in SCC3
- Sets the break flag bit (BKF) in SCS2
- May set the overrun (OR), noise flag (NF), parity error (PE), or reception in progress flag (RPF) bits

13.7.2.4 Idle Characters

An idle character contains all logic 1s and has no start, stop, or parity bit. Idle character length depends on the M bit in SCC1. The preamble is a synchronizing idle character that begins every transmission.

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 character currently being transmitted.

14.5 Functional Description

Figure 14-2 shows the structure of the SPI module.

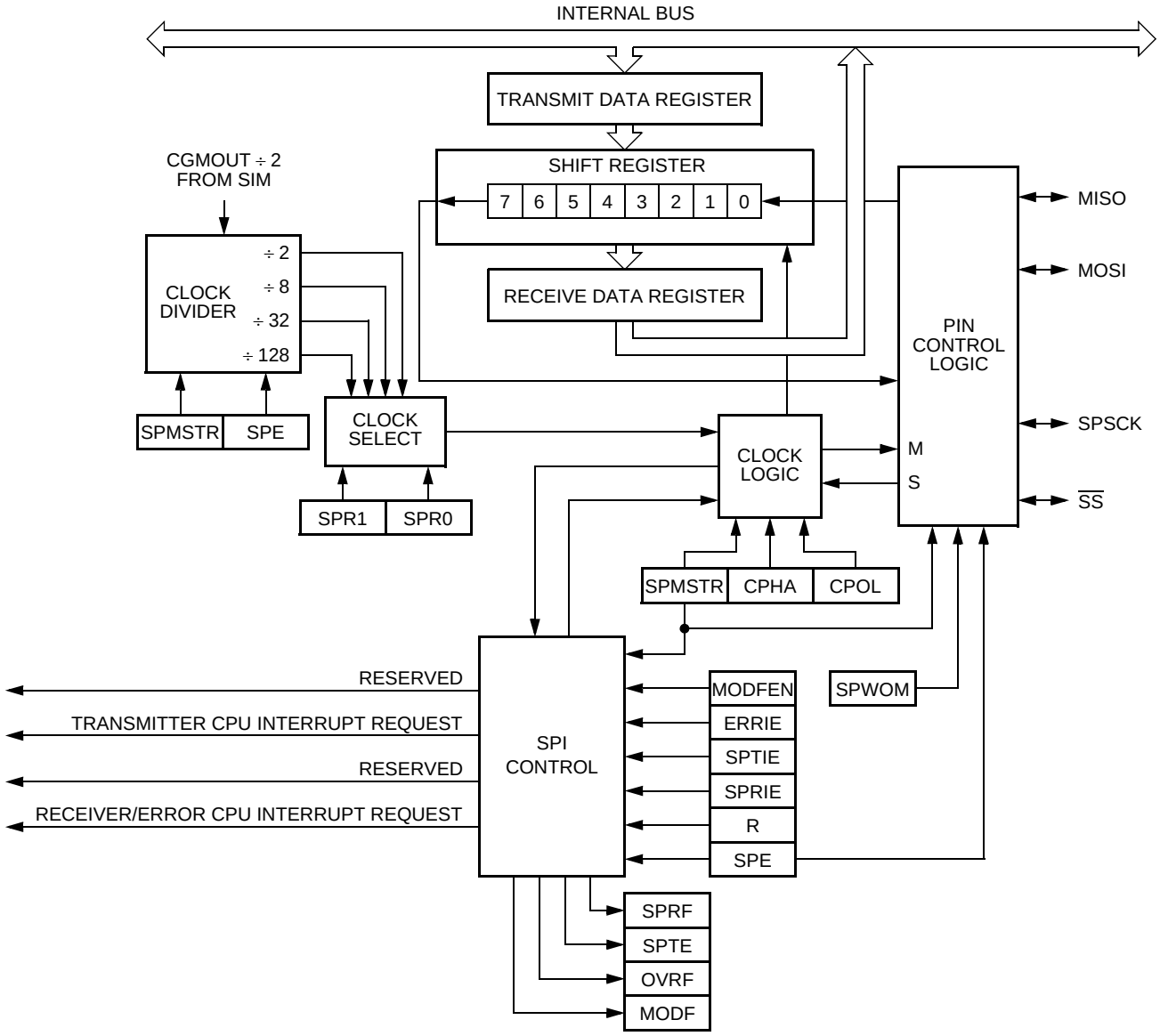

Figure 14-2. SPI Module Block Diagram

The SPI module allows full-duplex, synchronous, serial communication between the MCU and peripheral devices, including other MCUs. Software can poll the SPI status flags or SPI operation can be interrupt-driven.

The following paragraphs describe the operation of the SPI module.

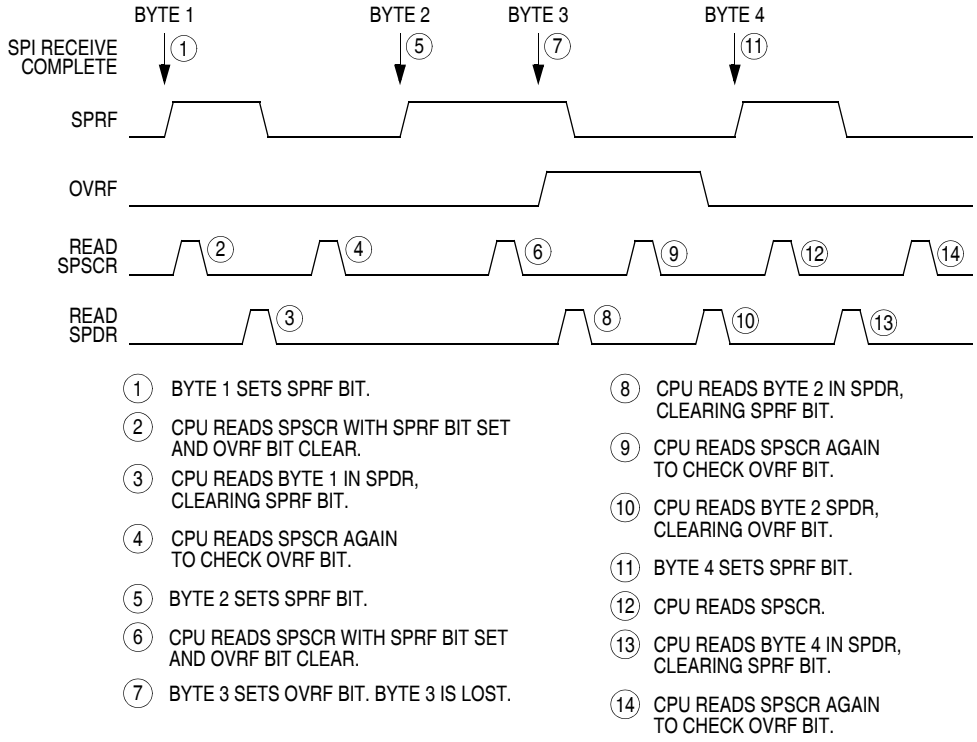

Figure 14-10. Clearing SPRF When OVRF Interrupt Is Not Enabled

14.8.2 Mode Fault Error

Setting the SPMSTR bit selects master mode and configures the SPSCK and MOSI pins as outputs and the MISO pin as an input. Clearing SPMSTR selects slave mode and configures the SPSCK and MOSI pins as inputs and the MISO pin as an output. The mode fault bit, MODF, becomes set any time the state of the slave select pin, $\overline{SS}$, is inconsistent with the mode selected by SPMSTR.

To prevent SPI pin contention and damage to the MCU, a mode fault error occurs if:

- The $\overline{SS}$ pin of a slave SPI goes high during a transmission
- The $\overline{SS}$ pin of a master SPI goes low at any time

For the MODF flag to be set, the mode fault error enable bit (MODFEN) must be set. Clearing the MODFEN bit does not clear the MODF flag but does prevent MODF from being set again after MODF is cleared.

14.12 SPI During Break Interrupts

The system integration module (SIM) controls whether status bits in other modules can be cleared during the break state. The BCFE bit in the SIM break flag control register (SBFCR) enables software to clear status bits during the break state. (See **Section 9. System Integration Module (SIM)**.)

To allow software to clear status bits during a break interrupt, write a logic 1 to the BCFE bit. If a status bit is cleared during the break state, it remains cleared when the MCU exits the break state.

To protect status bits during the break state, write a logic 0 to the BCFE bit. With BCFE at logic 0 (its default state), software can read and write I/O registers during the break state without affecting status bits. Some status bits have a 2-step read/write clearing procedure. If software does the first step on such a bit before the break, the bit cannot change during the break state as long as BCFE is at logic 0. After the break, doing the second step clears the status bit.

Since the SPTE bit cannot be cleared during a break with the BCFE bit cleared, a write to the transmit data register in break mode does not initiate a transmission nor is this data transferred into the shift register. Therefore, a write to the SPDR in break mode with the BCFE bit cleared has no effect.

14.13 I/O Signals

The SPI module has five I/O pins and shares four of them with a parallel I/O port. They are:

- MISO — Data received
- MOSI — Data transmitted
- SPSCK — Serial clock
- $\overline{SS}$ — Slave select
- CGND — Clock ground (internally connected to V_{SS})

REPSEN — Repeated Start Enable

This bit is set to enable repeated START signal to be generated when in master mode transfer (MMAST = 1). The REPSEN bit is cleared by hardware after the completion of repeated START signal or when the MMAST bit is cleared. Reset clears this bit.

1 = Repeated START signal will be generated if MMAST bit is set

0 = No repeated START signal will be generated

15.5.3 Multi-Master IIC Master Control Register (MIMCR)

Address: $006A

	Bit 7	6	5	4	3	2	1	Bit 0
Read:	MMALIF	MMNAKIF	MMBB	MMAST	MMRW	MMBR2	MMBR1	MMBR0
Write:	0	0						
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 15-4. Multi-Master IIC Master Control Register (MIMCR)

MMALIF — Multi-Master Arbitration Lost Interrupt Flag

This flag is set when software attempt to set MMAST but the MMBB has been set by detecting the start condition on the lines or when the MMIIC is transmitting a "1" to SDA line but detected a "0" from SDA line in master mode – an arbitration loss. This bit generates an interrupt request to the CPU if the MMIEN bit in MMCR is also set. This bit is cleared by writing "0" to it or by reset.

1 = Lost arbitration in master mode

0 = No arbitration lost

MMNAKIF — No Acknowledge Interrupt Flag

This flag is only set in master mode (MMAST = 1) when there is no acknowledge bit detected after one data byte or calling address is transferred. This flag also clears MMAST. MMNAKIF generates an interrupt request to CPU if the MMIEN bit in MMCR is also set. This bit is cleared by writing "0" to it or by reset.

1 = No acknowledge bit detected

0 = Acknowledge bit detected

16.5 Interrupts

When the AIEN bit is set, the ADC module is capable of generating a CPU interrupt after each ADC conversion. A CPU interrupt is generated if the COCO bit is at logic 0. The COCO bit is not used as a conversion complete flag when interrupts are enabled. The interrupt vector is defined in **Table 2-1 . Vector Addresses**.

16.6 Low-Power Modes

The STOP and WAIT instructions put the MCU in low power-consumption standby modes.

16.6.1 Wait Mode

The ADC continues normal operation in wait mode. Any enabled CPU interrupt request from the ADC can bring the MCU out of wait mode. If the ADC is not required to bring the MCU out of wait mode, power down the ADC by setting the ADCH[4:0] bits to logic 1's before executing the WAIT instruction.

16.6.2 Stop Mode

The ADC module is inactive after the execution of a STOP instruction. Any pending conversion is aborted. ADC conversions resume when the MCU exits stop mode. Allow one conversion cycle to stabilize the analog circuitry before attempting a new ADC conversion after exiting stop mode.

16.7 I/O Signals

The ADC module has ten channels, six channels are shared with port A and port B I/O pins; two channels are the ADC voltage reference inputs, V_{REFH} and V_{REFL}; one channel is the V_{LCD} input; and one channel is the 1.2V bandgap reference voltage.

Section 19. External Interrupt (IRQ)

19.1 Contents

19.2 Introduction

The external interrupt (IRQ) module provides a maskable interrupt input.

19.3 Features

Features of the IRQ module include the following:

- A dedicated external interrupt pin ($\overline{\text{IRQ}}$)
- IRQ interrupt control bits
- Hysteresis buffer
- Programmable edge-only or edge and level interrupt sensitivity
- Automatic interrupt acknowledge
- Internal pullup resistor

24.18 FLASH Memory Characteristics

Table 24-16. FLASH Memory Electrical Characteristics

Characteristic	Symbol	Min.	Max.	Unit
RAM data retention voltage	V_{RDR}	1.3	—	V
Number of rows per page		2		Rows
Number of bytes per page		128		Bytes
Read bus clock frequency	f_{read}[1]	32k	8M	Hz
Page erase time	t_{erase}[2]	1	—	ms
Mass erase time	t_{merase}[3]	4	—	ms
PGM/ERASE to HVEN setup time	t_{nvs}	10	—	µs
High-voltage hold time	t_{nvh}	5	—	µs
High-voltage hold time (mass erase)	t_{nvhl}	100	—	µs
Program hold time	t_{pgs}	5	—	µs
Program time	t_{prog}	30	40	µs
Address/data setup time	t_{ads}	—	30	ns
Address/data hold time	t_{adh}	—	30	ns
Recovery time	t_{rcv}[4]	1	—	µs
Cumulative HV period	t_{hv}[5]	—	25	ms
Row erase endurance[6]	—	10k	—	Cycles
Row program endurance[7]	—	10k	—	Cycles
Data retention time[8]	—	10	—	Years

Notes:
1. f_{read} is defined as the frequency range for which the FLASH memory can be read.
2. If the page erase time is longer than t_{erase} (Min.), there is no erase-disturb, but it reduces the endurance of the FLASH memory.
3. If the mass erase time is longer than t_{merase} (Min.), there is no erase-disturb, but is reduces the endurance of the FLASH memory.
4. It is defined as the time it needs before the FLASH can be read after turning off the high voltage charge pump, by clearing HVEN to logic 0.
5. t_{hv} is the cumulative high voltage programming time to the same row before next erase, and the same address can not be programmed twice before next erase.
6. The minimum row endurance value specifies each row of the FLASH memory is guaranteed to work for at least this many erase/program cycles.
7. The minimum row endurance value specifies each row of the FLASH memory is guaranteed to work for at least this many erase/program cycle.
8. The FLASH is guaranteed to retain data over the entire operating temperature range for at least the minimum time specified.