Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

XC164LM

16-Bit Single-Chip Microcontroller with C166SV2 Core

Microcontrollers

Never stop thinking

Edition 2007-03

Published by
Infineon Technologies AG
81726 Munich, Germany

Legal Disclaimer

The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics. With respect to any examples or hints given herein, any typical values stated herein and/or any information regarding the application of the device, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation, warranties of non-infringement of intellectual property rights of any third party.

Information

For further information on technology, delivery terms and conditions and prices, please contact the nearest Infineon Technologies Office (**www.infineon.com**).

Warnings

Due to technical requirements, components may contain dangerous substances. For information on the types in question, please contact the nearest Infineon Technologies Office.

Infineon Technologies components may be used in life-support devices or systems only with the express written approval of Infineon Technologies, if a failure of such components can reasonably be expected to cause the failure of that life-support device or system or to affect the safety or effectiveness of that device or system. Life support devices or systems are intended to be implanted in the human body or to support and/or maintain and sustain and/or protect human life. If they fail, it is reasonable to assume that the health of the user or other persons may be endangered.

XC164LM

16-Bit Single-Chip Microcontroller with C166SV2 Core

Microcontrollers

Never stop thinking

Table of Contents

XC164LM
Revision History: V1.2, 2007-03

Previous Version(s):
V1.1, 2006-08
V1.0, 2005-11

Page	Subjects (major changes since last revision)
6	Design steps of the derivatives differentiated.
47	Power consumption of the derivatives differentiated.
48	Figure 9 adapted.
49	Figure 11 adapted.
56	Packages of the derivatives differentiated.
57	Thermal resistances of the derivatives differentiated.
all	“Preliminary” removed

We Listen to Your Comments

Any information within this document that you feel is wrong, unclear or missing at all? Your feedback will help us to continuously improve the quality of this document. Please send your proposal (including a reference to this document) to:

mcdocu.comments@infineon.com

1 Summary of Features

For a quick overview or reference, the XC164LM's properties are listed here in a condensed way.

- High Performance 16-bit CPU with 5-Stage Pipeline
  - 25 ns Instruction Cycle Time at 40 MHz CPU Clock (Single-Cycle Execution)
  - 1-Cycle Multiplication (16 × 16 bit), Background Division (32 / 16 bit) in 21 Cycles
  - 1-Cycle Multiply-and-Accumulate (MAC) Instructions
  - Enhanced Boolean Bit Manipulation Facilities
  - Zero-Cycle Jump Execution
  - Additional Instructions to Support HLL and Operating Systems
  - Register-Based Design with Multiple Variable Register Banks
  - Fast Context Switching Support with Two Additional Local Register Banks
  - 16 Mbytes Total Linear Address Space for Code and Data
  - 1024 Bytes On-Chip Special Function Register Area (C166 Family Compatible)
- 16-Priority-Level Interrupt System with up to 63 Sources, Sample-Rate down to 50 ns
- 8-Channel Interrupt-Driven Single-Cycle Data Transfer Facilities via Peripheral Event Controller (PEC), 24-Bit Pointers Cover Total Address Space
- Clock Generation via on-chip PLL (factors 1:0.15 … 1:10), or via Prescaler (factors 1:1 … 60:1)
- On-Chip Memory Modules
  - 2 Kbytes On-Chip Dual-Port RAM (DPRAM)
  - 0/2/4 Kbytes[1)] On-Chip Data SRAM (DSRAM)
  - 2 Kbytes On-Chip Program/Data SRAM (PSRAM)
  - 32/64/128[1)] Kbytes On-Chip Program Memory (Flash Memory)
- On-Chip Peripheral Modules
  - 16-Channel General Purpose Capture/Compare Unit (CAPCOM2)
  - Multi-Functional General Purpose Timer Unit with 5 Timers
  - Two Synchronous/Asynchronous Serial Channels (USARTs)
  - Two High-Speed-Synchronous Serial Channels
  - On-Chip Real Time Clock, Driven by the Main Oscillator
- Idle, Sleep, and Power Down Modes with Flexible Power Management
- Programmable Watchdog Timer and Oscillator Watchdog
- Up to 47 General Purpose I/O Lines, partly with Selectable Input Thresholds and Hysteresis
- On-Chip Bootstrap Loader

1) Depends on the respective derivative. See **Table 1 "XC164LM Derivative Synopsis" on Page 6**.

- On-Chip Debug Support via JTAG Interface
- 64-Pin Green LQFP Package for the -16F derivatives, 0.5 mm (19.7 mil) pitch (RoHS compliant)
- 64-Pin TQFP Package for the -4F/8F derivatives, 0.5 mm (19.7 mil) pitch (RoHS compliant)

Ordering Information

The ordering code for Infineon microcontrollers provides an exact reference to the required product. This ordering code identifies:

- the derivative itself, i.e. its function set, the temperature range, and the supply voltage
- the package and the type of delivery.

For the available ordering codes for the XC164LM please refer to your responsible sales representative or your local distributor.

This document describes several derivatives of the XC164LM group. **Table 1** enumerates these derivatives and summarizes the differences. As this document refers to all of these derivatives, some descriptions may not apply to a specific product.

For simplicity all versions are referred to by the term **XC164LM** throughout this document.

Table 1 XC164LM Derivative Synopsis

Derivative[1)]	Temp. Range	Program Memory	On-Chip RAM	Interfaces
SAF-XC164LM-16F40F SAF-XC164LM-16F20F	-40 to 85 °C	128 Kbytes Flash	2 Kbytes DPRAM, 4 Kbytes DSRAM, 2 Kbytes PSRAM	ASC0, ASC1, SSC0, SSC1
SAF-XC164LM-8F40F SAF-XC164LM-8F20F	-40 to 85 °C	64 Kbytes Flash	2 Kbytes DPRAM, 2 Kbytes DSRAM, 2 Kbytes PSRAM	ASC0, ASC1, SSC0, SSC1,
SAF-XC164LM-4F40F SAF-XC164LM-4F20F	-40 to 85 °C	32 Kbytes Flash	2 Kbytes DPRAM, 2 Kbytes PSRAM	ASC0, ASC1, SSC0, SSC1

1) This Data Sheet is valid for:
devices starting with and including design step BA for the -16F derivatives, and for
devices starting with and including design step AA for -4F/8F derivatives.

2 General Device Information

The XC164LM derivatives are high-performance members of the Infineon XC166 Family of full featured single-chip CMOS microcontrollers. These devices extend the functionality and performance of the C166 Family in terms of instructions (MAC unit), peripherals, and speed. They combine high CPU performance (up to 40 million instructions per second) with high peripheral functionality and enhanced IO-capabilities. They also provide clock generation via PLL and various on-chip memory modules such as program Flash, program RAM, and data RAM.

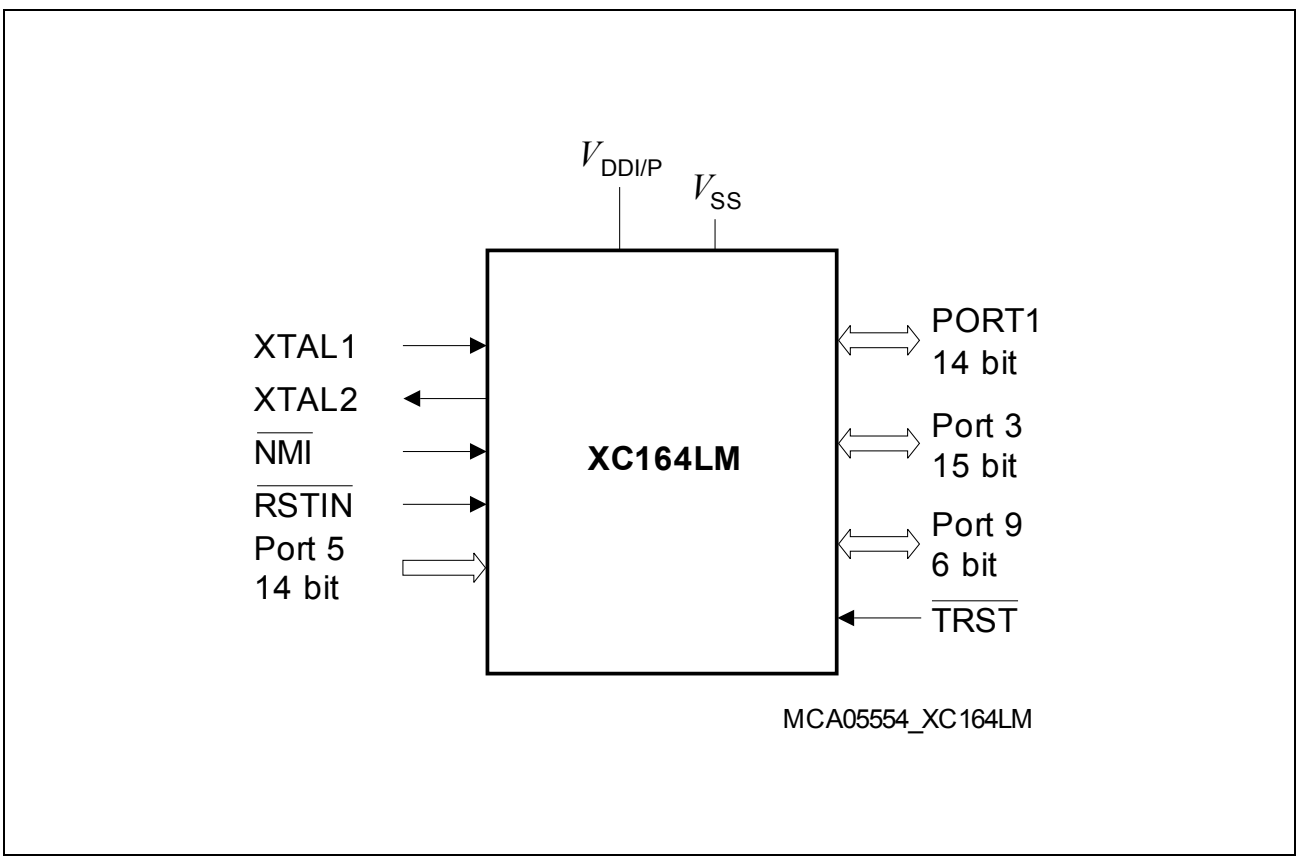

Figure 1 Logic Symbol

2.1 Pin Configuration and Definition

The pins of the XC164LM are described in detail in **Table 2**, including all their alternate functions. **Figure 2** summarizes all pins in a condensed way, showing their location on the 4 sides of the package. E* marks pins to be used as alternate external interrupt inputs.

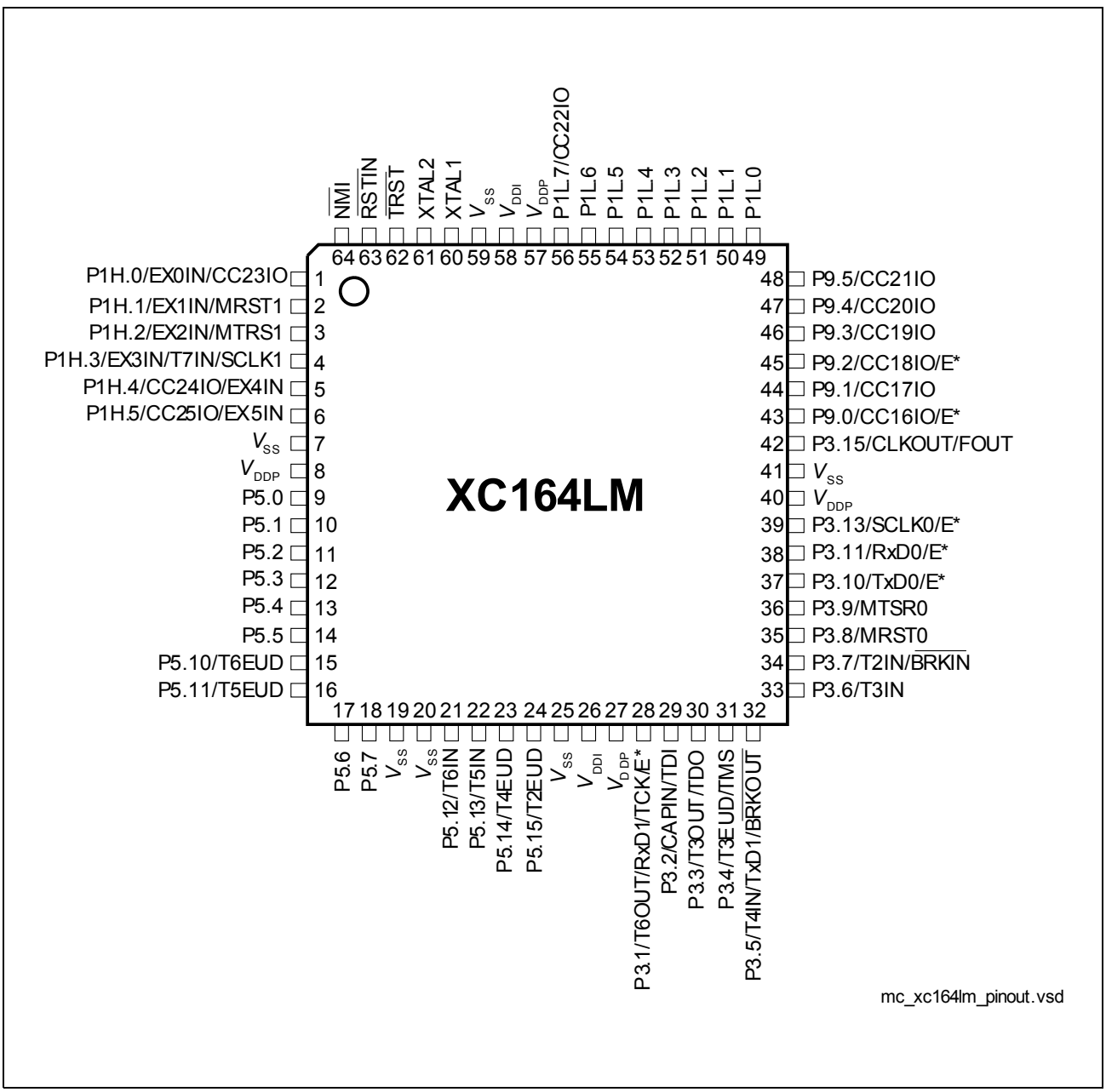

Figure 2 Pin Configuration (top view)

Table 2 Pin Definitions and Functions

Sym-bol	Pin Num.	Input Outp.	Function
$\overline{\text{RSTIN}}$	63	I	Reset Input with Schmitt-Trigger characteristics. A low-level at this pin while the oscillator is running resets the XC164LM. A spike filter suppresses input pulses < 10 ns. Input pulses > 100 ns safely pass the filter. The minimum duration for a safe recognition should be 100 ns + 2 CPU clock cycles. *Note: The reset duration must be sufficient to let the hardware configuration signals settle. External circuitry must guarantee low-level at the $\overline{\text{RSTIN}}$ pin at least until both power supply voltages have reached the operating range.*
$\overline{\text{NMI}}$	64	I	Non-Maskable Interrupt Input. A high to low transition at this pin causes the CPU to vector to the NMI trap routine. When the PWRDN (power down) instruction is executed, the $\overline{\text{NMI}}$ pin must be low in order to force the XC164LM into power down mode. If $\overline{\text{NMI}}$ is high, when PWRDN is executed, the part will continue to run in normal mode. If not used, pin $\overline{\text{NMI}}$ should be pulled high externally.
Port 9	43-48	IO	Port 9 is a 6-bit bidirectional I/O port. Each pin can be programmed for input (output driver in high-impedance state) or output (configurable as push/pull or open drain driver). The input threshold of Port 9 is selectable (standard or special). The following Port 9 pins also serve for alternate functions: CC16IO: (CAPCOM2) CC16 Capture Inp./Compare Outp.,
P9.0	43	I/O I	EX5IN: (Fast External Interrupt 5) Input (alternate pin A) CC17IO: (CAPCOM2) CC17 Capture Inp./Compare Outp.,
P9.1	44	I/O	CC18IO: (CAPCOM2) CC18 Capture Inp./Compare Outp.,
P9.2	45	I/O I	EX4IN: (Fast External Interrupt 4) Input (alternate pin A) CC19IO: (CAPCOM2) CC19 Capture Inp./Compare Outp.,
P9.3	46	I/O	CC20IO: (CAPCOM2) CC20 Capture Inp./Compare Outp.
P9.4	47	I/O	CC21IO: (CAPCOM2) CC21 Capture Inp./Compare Outp.
P9.5	48	I/O	*Note: At the end of an external reset P9.4 and P9.5 also may input startup configuration values*

Table 2 Pin Definitions and Functions (cont'd)

Sym-bol	Pin Num.	Input Outp.	Function
Port 5	9-18, 21-24	I	Port 5 is a 14-bit input-only port. Some pins of Port 5 also serve as timer inputs:
P5.10	15	I	T6EUD: GPT2 Timer T6 Ext. Up/Down Control Input
P5.11	16	I	T5EUD: GPT2 Timer T5 Ext. Up/Down Control Input
P5.12	21	I	T6IN: GPT2 Timer T6 Count/Gate Input
P5.13	22	I	T5IN: GPT2 Timer T5 Count/Gate Input
P5.14	23	I	T4EUD: GPT1 Timer T4 Ext. Up/Down Control Input
P5.15	24	I	T2EUD: GPT1 Timer T2 Ext. Up/Down Control Input
$\overline{\text{TRST}}$	62	I	Test-System Reset Input. For normal system operation, pin $\overline{\text{TRST}}$ should be held low. A high level at this pin at the rising edge of $\overline{\text{RSTIN}}$ enables the hardware configuration and activates the XC164LM's debug system. In this case, pin $\overline{\text{TRST}}$ must be driven low once to reset the debug system.

Table 2 Pin Definitions and Functions (cont'd)

Sym-bol	Pin Num.	Input Outp.	Function
Port 3	28-39, 42	IO	Port 3 is a 13-bit bidirectional I/O port. Each pin can be programmed for input (output driver in high-impedance state) or output (configurable as push/pull or open drain driver). The input threshold of Port 3 is selectable (standard or special).The following Port 3 pins also serve for alternate functions:
P3.1	28	O	T6OUT: [GPT2] Timer T6 Toggle Latch Output,
		I/O	RxD1: [ASC1] Data Input (Async.) or Inp./Outp. (Sync.),
		I	EX1IN: [Fast External Interrupt 1] Input (alternate pin A),
		I	TCK: [Debug System] JTAG Clock Input
P3.2	29	I	CAPIN: [GPT2] Register CAPREL Capture Input,
		I	TDI: [Debug System] JTAG Data In
P3.3	30	O	T3OUT: [GPT1] Timer T3 Toggle Latch Output,
		O	TDO: [Debug System] JTAG Data Out
P3.4	31	I	T3EUD: [GPT1] Timer T3 External Up/Down Control Input,
		I	TMS: [Debug System] JTAG Test Mode Selection
P3.5	32	I	T4IN: [GPT1] Timer T4 Count/Gate/Reload/Capture Inp.
		O	TxD1: [ASC0] Clock/Data Output (Async./Sync.),
		O	BRKOUT: [Debug System] Break Out
P3.6	33	I	T3IN: [GPT1] Timer T3 Count/Gate Input
P3.7	34	I	T2IN: [GPT1] Timer T2 Count/Gate/Reload/Capture Inp.
		I	BRKIN: [Debug System] Break In
P3.8	35	I/O	MRST0: [SSC0] Master-Receive/Slave-Transmit In/Out.
P3.9	36	I/O	MTSR0: [SSC0] Master-Transmit/Slave-Receive Out/In.
P3.10	37	O	TxD0: [ASC0] Clock/Data Output (Async./Sync.),
		I	EX2IN: [Fast External Interrupt 2] Input (alternate pin B)
P3.11	38	I/O	RxD0: [ASC0] Data Input (Async.) or Inp./Outp. (Sync.),
		I	EX2IN: [Fast External Interrupt 2] Input (alternate pin A)
P3.13	39	I/O	SCLK0: [SSC0] Master Clock Output / Slave Clock Input.,
		I	EX3IN: [Fast External Interrupt 3] Input (alternate pin A)
P3.15	42	O	CLKOUT: System Clock Output (= CPU Clock),
		O	FOUT: Programmable Frequency Output

Table 2 Pin Definitions and Functions (cont'd)

Sym-bol	Pin Num.	Input Outp.	Function
PORT1	1-6, 49-56	IO	PORT1 consists of one 8-bit and one 6-bit bidirectional I/O port P1L and P1H. Each pin can be programmed for input (output driver in high-impedance state) or output. The following PORT1 pins also serve for alt. functions:
P1L.7	56	I/O	CC22IO: [CAPCOM2] CC22 Capture Inp./Compare Outp.
P1H.0	1	I	EX0IN: [Fast External Interrupt 0] Input (default pin),
		I/O	CC23IO: [CAPCOM2] CC23 Capture Inp./Compare Outp.
P1H.1	2	I	EX1IN: [Fast External Interrupt 1] Input (default pin),
		I/O	MRST1: [SSC1] Master-Receive/Slave-Transmit In/Out.
P1H.2	3	I	EX2IN: [Fast External Interrupt 2] Input (default pin),
		I/O	MTSR1: [SSC1] Master-Transmit/Slave-Receive Out/Inp.
P1H.3	3	I	T7IN: [CAPCOM2] Timer T7 Count Input,
		I/O	SCLK1: [SSC1] Master Clock Output / Slave Clock Input,
		I	EX3IN: [Fast External Interrupt 3] Input (default pin),
P1H.4	5	I/O	CC24IO: [CAPCOM2] CC24 Capture Inp./Compare Outp.,
		I	EX4IN: [Fast External Interrupt 4] Input (default pin)
P1H.5	6	I/O	CC25IO: [CAPCOM2] CC25 Capture Inp./Compare Outp.,
		I	EX5IN: [Fast External Interrupt 5] Input (default pin) *Note: At the end of an external reset P1H.4 and P1H.5 also may input startup configuration values*
XTAL2	61	O	XTAL2: Output of the oscillator amplifier circuit
XTAL1	60	I	XTAL1: Input to the oscillator amplifier and input to the internal clock generator To clock the device from an external source, drive XTAL1, while leaving XTAL2 unconnected. Minimum and maximum high/low and rise/fall times specified in the AC Characteristics must be observed. *Note: Input pin XTAL1 belongs to the core voltage domain. Therefore, input voltages must be within the range defined for V_{DDI}.*
V_{DDI}	26, 58	–	Digital Core Supply Voltage (On-Chip Modules): +2.5 V during normal operation and idle mode. Please refer to the **Operating Condition Parameters**

Table 2 Pin Definitions and Functions (cont'd)

Sym-bol	Pin Num.	Input Outp.	Function
V_{DDP}	8, 27, 40, 57	–	Digital Pad Supply Voltage (Pin Output Drivers): +5 V during normal operation and idle mode. Please refer to the **Operating Condition Parameters**
V_{SS}	7, 25, 41, 59	–	**Digital Ground** Connect decoupling capacitors to adjacent V_{DD}/V_{SS} pin pairs as close as possible to the pins. All V_{SS} pins must be connected to the ground-line or ground-plane.

3 Functional Description

The architecture of the XC164LM combines advantages of RISC, CISC, and DSP processors with an advanced peripheral subsystem in a very well-balanced way. In addition, the on-chip memory blocks allow the design of compact systems-on-silicon with maximum performance (computing, control, communication).

The on-chip memory blocks (program code-memory and SRAM, dual-port RAM, data SRAM) and the set of generic peripherals are connected to the CPU via separate buses. Another bus, the LXBus, connects additional on-chip resources (see **Figure 3**).

This bus structure enhances the overall system performance by enabling the concurrent operation of several subsystems of the XC164LM.

The following block diagram gives an overview of the different on-chip components and of the advanced, high bandwidth internal bus structure of the XC164LM.

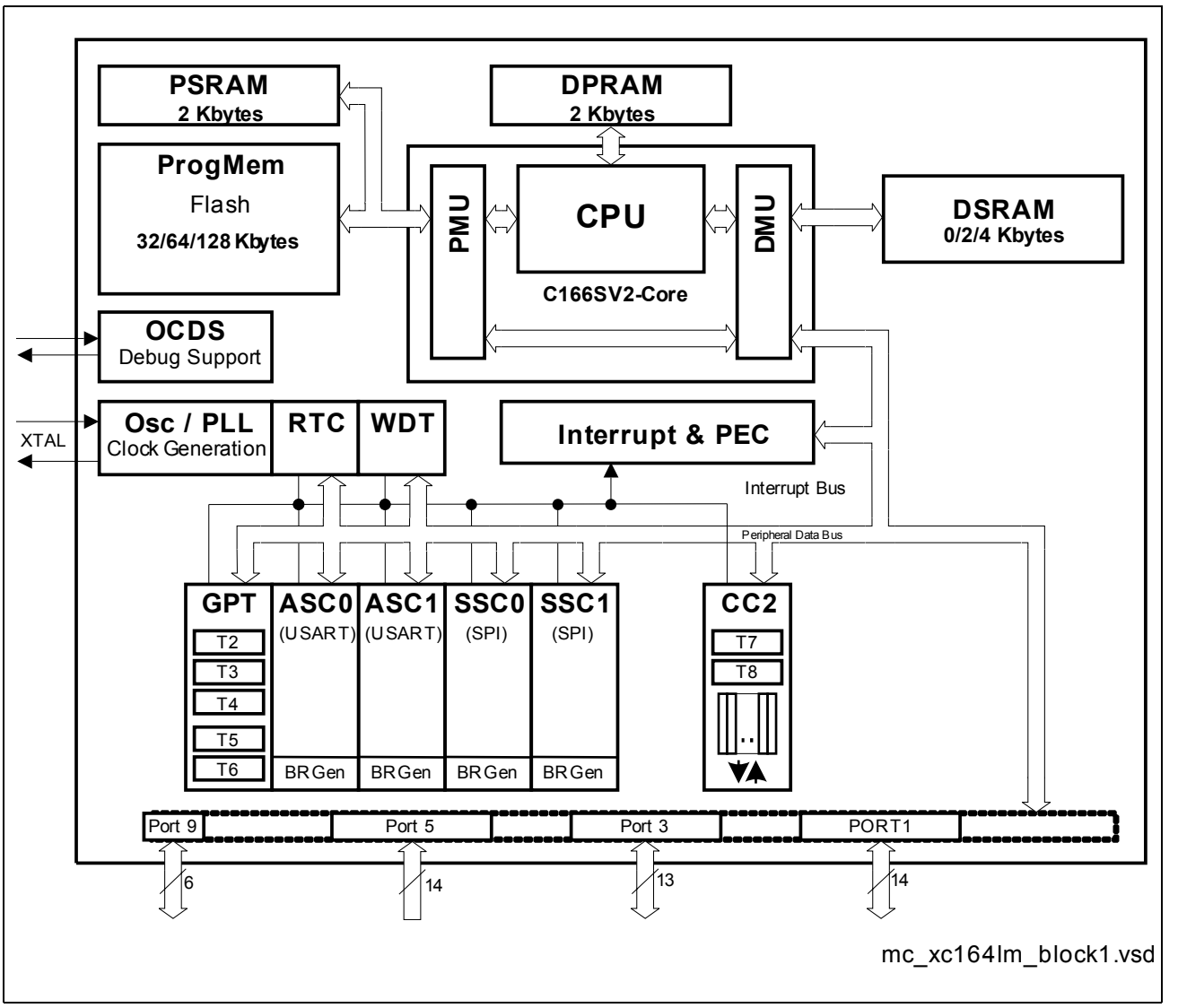

Figure 3 Block Diagram

3.1 Memory Subsystem and Organization

The memory space of the XC164LM is configured in a von Neumann architecture, which means that all internal and external resources, such as code memory, data memory, registers and I/O ports, are organized within the same linear address space. This common memory space includes 16 Mbytes and is arranged as 256 segments of 64 Kbytes each, where each segment consists of four data pages of 16 Kbytes each. The entire memory space can be accessed byte wise or word wise. Portions of the on-chip DPRAM and the register spaces (E/SFR) have additionally been made directly bit addressable.

The internal data memory areas and the Special Function Register areas (SFR and ESFR) are mapped into segment 0, the system segment.

The Program Management Unit (PMU) handles all code fetches and, therefore, controls accesses to the program memories, such as Flash memory and PSRAM.

The Data Management Unit (DMU) handles all data transfers and, therefore, controls accesses to the DSRAM and the on-chip peripherals.

Both units (PMU and DMU) are connected via the high-speed system bus to exchange data. This is required if operands are read from program memory or code or data is written to the PSRAM. The system bus allows concurrent two-way communication for maximum transfer performance.

32/64/128 Kbytes of on-chip Flash memory[1] store code or constant data. The on-chip Flash memory is organized as four 8-Kbyte sectors and up to three 32-Kbyte sectors. Each sector can be separately write protected[2], erased and programmed (in blocks of 128 Bytes). The complete Flash area can be read-protected. A password sequence temporarily unlocks protected areas. The Flash module combines very fast 64-bit one-cycle read accesses with protected and efficient writing algorithms for programming and erasing. Thus, program execution out of the internal Flash results in maximum performance. Dynamic error correction provides extremely high read data security for all read accesses.
Programming typically takes 2 ms per 128-byte block (5 ms max.), erasing a sector typically takes 200 ms (500 ms max.).

2 Kbytes of on-chip Program SRAM (PSRAM) are provided to store user code or data. The PSRAM is accessed via the PMU and is therefore optimized for code fetches.

0/2/4 Kbytes[1] of on-chip Data SRAM (DSRAM) are provided as a storage for general user data. The DSRAM is accessed via the DMU and is therefore optimized for data accesses. DSRAM is not available in the XC164LM-4F derivatives.

2 Kbytes of on-chip Dual-Port RAM (DPRAM) are provided as a storage for user defined variables, for the system stack, general purpose register banks. A register bank

1) Depends on the respective derivative. See **Table 1 "XC164LM Derivative Synopsis" on Page 6**.

2) Each two 8-Kbyte sectors are combined for write-protection purposes.

can consist of up to 16 word wide (R0 to R15) and/or byte wide (RL0, RH0, …, RL7, RH7) so-called General Purpose Registers (GPRs).
The upper 256 bytes of the DPRAM are directly bit addressable. When used by a GPR, any location in the DPRAM is bit addressable.

1024 bytes (2 × 512 bytes) of the address space are reserved for the Special Function Register areas (SFR space and ESFR space). SFRs are word wide registers which are used for controlling and monitoring functions of the different on-chip units. Unused SFR addresses are reserved for future members of the XC166 Family. Therefore, they should either not be accessed, or written with zeros, to ensure upward compatibility.

Table 3 XC164LM Memory Map

Address Area	Start Loc.	End Loc.	Area Size[1)]	Notes
Flash register space	$FF'F000_H$	$FF'FFFF_H$	4 Kbytes	2)
Reserved (Acc. trap)	$F8'0000_H$	$FF'FFFF_H$	508 Kbytes	–
Reserved for PSRAM	$E0'0800_H$	$F7'FFFF_H$	< 1.5 Mbytes	Minus PSRAM
Program SRAM	$E0'0000_H$	$E0'07FF_H$	2 Kbytes	–
Reserved for pr. mem.	$C2'0000_H$	$DF'FFFF_H$	< 2 Mbytes	Minus Flash
Program Flash	$C0'0000_H$	$C1'FFFF_H$	128 Kbytes	XC164LM-16F
	$C0'0000_H$	$C0'FFFF_H$	64 Kbytes	XC164LM-8F
	$C0'0000_H$	$C0'7FFF_H$	32 Kbytes	XC164LM-4F
Reserved	$20'0000_H$	$BF'FFFF_H$	< 10 Mbytes	
Reserved	$01'0000_H$	$1F'FFFF_H$	< 2 Mbytes	Minus segment 0
SFR area	$00'FE00_H$	$00'FFFF_H$	0.5 Kbyte	–
Dual-Port RAM	$00'F600_H$	$00'FDFF_H$	2 Kbytes	–
Reserved for DPRAM	$00'F200_H$	$00'F5FF_H$	1 Kbyte	–
ESFR area	$00'F000_H$	$00'F1FF_H$	0.5 Kbyte	–
XSFR area	$00'E000_H$	$00'EFFF_H$	4 Kbytes	–
Reserved	$00'D000_H$	$00'DFFF_H$	6 Kbytes	–
Data SRAM	$00'C000_H$	$00'CFFF_H$	4 Kbytes	3)
Reserved for DSRAM	$00'8000_H$	$00'BFFF_H$	16 Kbytes	–
Reserved	$00'0000_H$	$00'7FFF_H$	32 Kbytes	–

1) The areas marked with “<” are slightly smaller than indicated, see column “Notes”.

2) Not defined register locations return a trap code ($1E9B_H$).

3) Depends on the respective derivative. See **Table 1 “XC164LM Derivative Synopsis” on Page 6**.

3.2 Central Processing Unit (CPU)

The main core of the CPU consists of a 5-stage execution pipeline with a 2-stage instruction-fetch pipeline, a 16-bit arithmetic and logic unit (ALU), a 32-bit/40-bit multiply and accumulate unit (MAC), a register-file providing three register banks, and dedicated SFRs. The ALU features a multiply and divide unit, a bit-mask generator, and a barrel shifter.

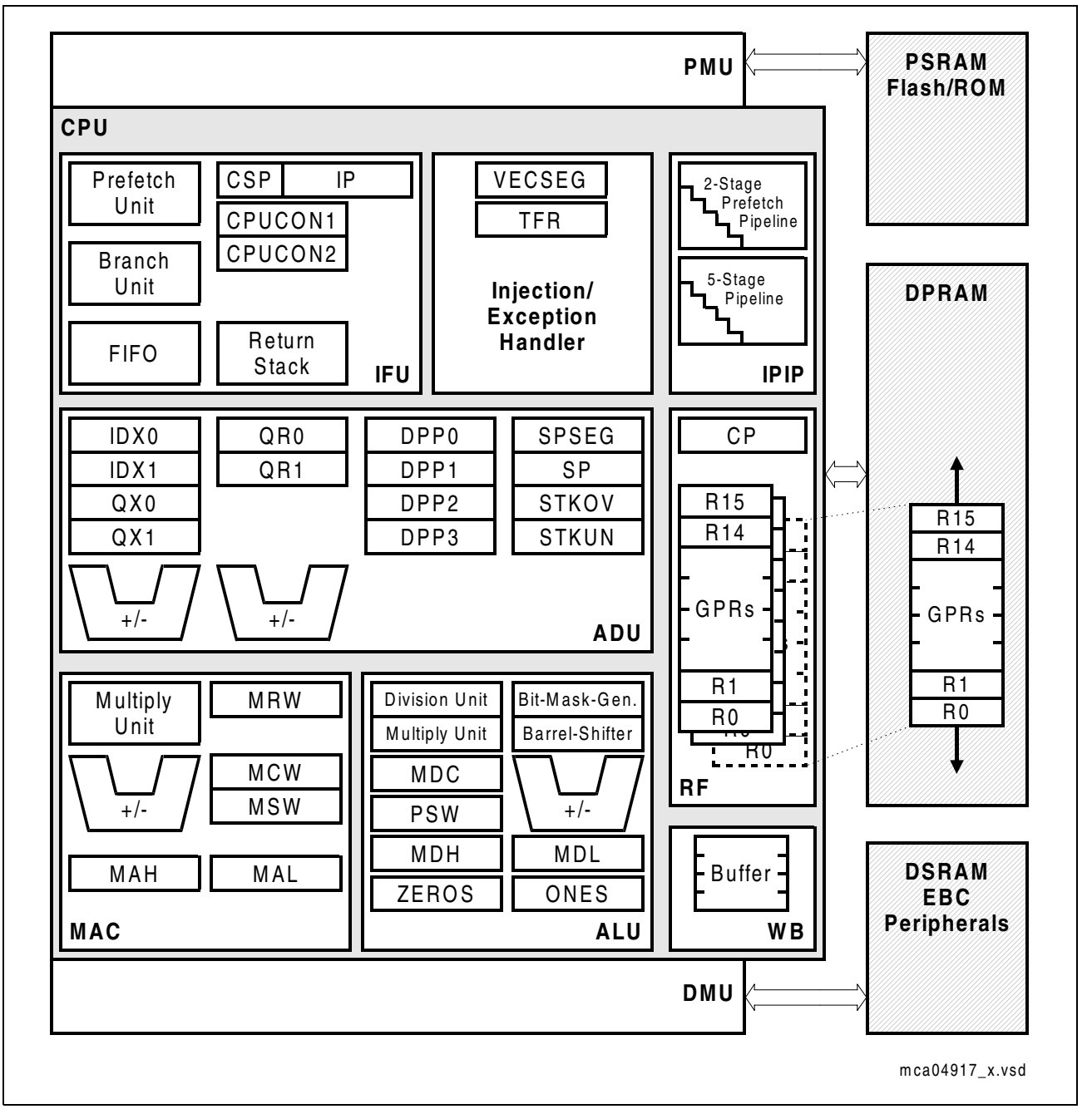

Figure 4 CPU Block Diagram

Based on these hardware provisions, most of the XC164LM's instructions can be executed in just one machine cycle which requires 25 ns at 40 MHz CPU clock. For

example, shift and rotate instructions are always processed during one machine cycle independent of the number of bits to be shifted. Also multiplication and most MAC instructions execute in one single cycle. All multiple-cycle instructions have been optimized so that they can be executed very fast as well: for example, a 32-/16-bit division is started within 4 cycles, while the remaining 15 cycles are executed in the background. Another pipeline optimization, the branch target prediction, allows eliminating the execution time of branch instructions if the prediction was correct.

The CPU has a register context consisting of up to three register banks with 16 word wide GPRs each at its disposal. One of these register banks is physically allocated within the on-chip DPRAM area. A Context Pointer (CP) register determines the base address of the active register bank to be accessed by the CPU at any time. The number of register banks is only restricted by the available internal RAM space. For easy parameter passing, a register bank may overlap others.

A system stack of up to 32 Kwords is provided as a storage for temporary data. The system stack can be allocated to any location within the address space (preferably in the on-chip RAM area), and it is accessed by the CPU via the stack pointer (SP) register. Two separate SFRs, STKOV and STKUN, are implicitly compared against the stack pointer value upon each stack access for the detection of a stack overflow or underflow.

The high performance offered by the hardware implementation of the CPU can efficiently be utilized by a programmer via the highly efficient XC164LM instruction set which includes the following instruction classes:

- Standard Arithmetic Instructions
- DSP-Oriented Arithmetic Instructions
- Logical Instructions
- Boolean Bit Manipulation Instructions
- Compare and Loop Control Instructions
- Shift and Rotate Instructions
- Prioritize Instruction
- Data Movement Instructions
- System Stack Instructions
- Jump and Call Instructions
- Return Instructions
- System Control Instructions
- Miscellaneous Instructions

The basic instruction length is either 2 or 4 bytes. Possible operand types are bits, bytes and words. A variety of direct, indirect or immediate addressing modes are provided to specify the required operands.

3.3 Interrupt System

With an interrupt response time of typically 8 CPU clocks (in case of internal program execution), the XC164LM is capable of reacting very fast to the occurrence of non-deterministic events.

The architecture of the XC164LM supports several mechanisms for fast and flexible response to service requests that can be generated from various sources internal or external to the microcontroller. Any of these interrupt requests can be programmed to being serviced by the Interrupt Controller or by the Peripheral Event Controller (PEC).

In contrast to a standard interrupt service where the current program execution is suspended and a branch to the interrupt vector table is performed, just one cycle is ‘stolen’ from the current CPU activity to perform a PEC service. A PEC service implies a single byte or word data transfer between any two memory locations with an additional increment of either the PEC source, or the destination pointer, or both. An individual PEC transfer counter is implicitly decremented for each PEC service except when performing in the continuous transfer mode. When this counter reaches zero, a standard interrupt is performed to the corresponding source related vector location. PEC services are very well suited, for example, for supporting the transmission or reception of blocks of data. The XC164LM has 8 PEC channels each of which offers such fast interrupt-driven data transfer capabilities.

A separate control register which contains an interrupt request flag, an interrupt enable flag and an interrupt priority bit field exists for each of the possible interrupt nodes. Via its related register, each node can be programmed to one of sixteen interrupt priority levels. Once having been accepted by the CPU, an interrupt service can only be interrupted by a higher prioritized service request. For the standard interrupt processing, each of the possible interrupt nodes has a dedicated vector location.

Fast external interrupt inputs are provided to service external interrupts with high precision requirements. These fast interrupt inputs feature programmable edge detection (rising edge, falling edge, or both edges).

Software interrupts are supported by means of the ‘TRAP’ instruction in combination with an individual trap (interrupt) number.

Table 4 shows all of the possible XC164LM interrupt sources and the corresponding hardware-related interrupt flags, vectors, vector locations and trap (interrupt) numbers.

Note: Interrupt nodes which are not assigned to peripherals (unassigned nodes), may be used to generate software controlled interrupt requests by setting the respective interrupt request bit (xIR).

Table 4 XC164LM Interrupt Nodes

Source of Interrupt or PEC Service Request	Control Register	Vector Location[1)]	Trap Number
EX0IN	CC1_CC8IC	xx'0060$_H$	18$_H$ / 24$_D$
EX1IN	CC1_CC9IC	xx'0064$_H$	19$_H$ / 25$_D$
EX2IN	CC1_CC10IC	xx'0068$_H$	1A$_H$ / 26$_D$
EX3IN	CC1_CC11IC	xx'006C$_H$	1B$_H$ / 27$_D$
EX4IN	CC1_CC12IC	xx'0070$_H$	1C$_H$ / 28$_D$
EX5IN	CC1_CC13IC	xx'0074$_H$	1D$_H$ / 29$_D$
CAPCOM Register 16	CC2_CC16IC	xx'00C0$_H$	30$_H$ / 48$_D$
CAPCOM Register 17	CC2_CC17IC	xx'00C4$_H$	31$_H$ / 49$_D$
CAPCOM Register 18	CC2_CC18IC	xx'00C8$_H$	32$_H$ / 50$_D$
CAPCOM Register 19	CC2_CC19IC	xx'00CC$_H$	33$_H$ / 51$_D$
CAPCOM Register 20	CC2_CC20IC	xx'00D0$_H$	34$_H$ / 52$_D$
CAPCOM Register 21	CC2_CC21IC	xx'00D4$_H$	35$_H$ / 53$_D$
CAPCOM Register 22	CC2_CC22IC	xx'00D8$_H$	36$_H$ / 54$_D$
CAPCOM Register 23	CC2_CC23IC	xx'00DC$_H$	37$_H$ / 55$_D$
CAPCOM Register 24	CC2_CC24IC	xx'00E0$_H$	38$_H$ / 56$_D$
CAPCOM Register 25	CC2_CC25IC	xx'00E4$_H$	39$_H$ / 57$_D$
CAPCOM Register 26	CC2_CC26IC	xx'00E8$_H$	3A$_H$ / 58$_D$
CAPCOM Register 27	CC2_CC27IC	xx'00EC$_H$	3B$_H$ / 59$_D$
CAPCOM Register 28	CC2_CC28IC	xx'00F0$_H$	3C$_H$ / 60$_D$
CAPCOM Register 29	CC2_CC29IC	xx'0110$_H$	44$_H$ / 68$_D$
CAPCOM Register 30	CC2_CC30IC	xx'0114$_H$	45$_H$ / 69$_D$
CAPCOM Register 31	CC2_CC31IC	xx'0118$_H$	46$_H$ / 70$_D$
CAPCOM Timer 7	CC2_T7IC	xx'00F4$_H$	3D$_H$ / 61$_D$
CAPCOM Timer 8	CC2_T8IC	xx'00F8$_H$	3E$_H$ / 62$_D$
GPT1 Timer 2	GPT12E_T2IC	xx'0088$_H$	22$_H$ / 34$_D$
GPT1 Timer 3	GPT12E_T3IC	xx'008C$_H$	23$_H$ / 35$_D$
GPT1 Timer 4	GPT12E_T4IC	xx'0090$_H$	24$_H$ / 36$_D$
GPT2 Timer 5	GPT12E_T5IC	xx'0094$_H$	25$_H$ / 37$_D$
GPT2 Timer 6	GPT12E_T6IC	xx'0098$_H$	26$_H$ / 38$_D$

Table 4 XC164LM Interrupt Nodes (cont'd)

Source of Interrupt or PEC Service Request	Control Register	Vector Location[1)]	Trap Number
GPT2 CAPREL Register	GPT12E_CRIC	xx'009C$_H$	27$_H$ / 39$_D$
ASC0 Transmit	ASC0_TIC	xx'00A8$_H$	2A$_H$ / 42$_D$
ASC0 Transmit Buffer	ASC0_TBIC	xx'011C$_H$	47$_H$ / 71$_D$
ASC0 Receive	ASC0_RIC	xx'00AC$_H$	2B$_H$ / 43$_D$
ASC0 Error	ASC0_EIC	xx'00B0$_H$	2C$_H$ / 44$_D$
ASC0 Autobaud	ASC0_ABIC	xx'017C$_H$	5F$_H$ / 95$_D$
SSC0 Transmit	SSC0_TIC	xx'00B4$_H$	2D$_H$ / 45$_D$
SSC0 Receive	SSC0_RIC	xx'00B8$_H$	2E$_H$ / 46$_D$
SSC0 Error	SSC0_EIC	xx'00BC$_H$	2F$_H$ / 47$_D$
PLL/OWD	PLLIC	xx'010C$_H$	43$_H$ / 67$_D$
ASC1 Transmit	ASC1_TIC	xx'0120$_H$	48$_H$ / 72$_D$
ASC1 Transmit Buffer	ASC1_TBIC	xx'0178$_H$	5E$_H$ / 94$_D$
ASC1 Receive	ASC1_RIC	xx'0124$_H$	49$_H$ / 73$_D$
ASC1 Error	ASC1_EIC	xx'0128$_H$	4A$_H$ / 74$_D$
ASC1 Autobaud	ASC1_ABIC	xx'0108$_H$	42$_H$ / 66$_D$
End of PEC Subchannel	EOPIC	xx'0130$_H$	4C$_H$ / 76$_D$
SSC1 Transmit	SSC1_TIC	xx'0144$_H$	51$_H$ / 81$_D$
SSC1 Receive	SSC1_RIC	xx'0148$_H$	52$_H$ / 82$_D$
SSC1 Error	SSC1_EIC	xx'014C$_H$	53$_H$ / 83$_D$
RTC	RTC_IC	xx'0174$_H$	5D$_H$ / 93$_D$
Unassigned node	–	xx'0040$_H$	10$_H$ / 16$_D$
Unassigned node	–	xx'0044$_H$	11$_H$ / 17$_D$
Unassigned node	–	xx'0048$_H$	12$_H$ / 18$_D$
Unassigned node	–	xx'004C$_H$	13$_H$ / 19$_D$
Unassigned node	–	xx'0050$_H$	14$_H$ / 20$_D$
Unassigned node	–	xx'0054$_H$	15$_H$ / 21$_D$
Unassigned node	–	xx'0058$_H$	16$_H$ / 22$_D$
Unassigned node	–	xx'005C$_H$	17$_H$ / 23$_D$
Unassigned node	–	xx'0078$_H$	1E$_H$ / 30$_D$
Unassigned node	–	xx'007C$_H$	1F$_H$ / 31$_D$

Table 4 XC164LM Interrupt Nodes (cont'd)

Source of Interrupt or PEC Service Request	Control Register	Vector Location[1)]	Trap Number
Unassigned node	–	xx'0080$_H$	20$_H$ / 32$_D$
Unassigned node	–	xx'0084$_H$	21$_H$ / 33$_D$
Unassigned node	–	xx'00A0$_H$	28$_H$ / 40$_D$
Unassigned node	–	xx'00A4$_H$	29$_H$ / 41$_D$
Unassigned node	–	xx'00FC$_H$	3F$_H$ / 63$_D$
Unassigned node	–	xx'0100$_H$	40$_H$ / 64$_D$
Unassigned node	–	xx'0104$_H$	41$_H$ / 65$_D$
Unassigned node	–	xx'012C$_H$	4B$_H$ / 75$_D$
Unassigned node	–	xx'0134$_H$	4D$_H$ / 77$_D$
Unassigned node	–	xx'0138$_H$	4E$_H$ / 78$_D$
Unassigned node	–	xx'013C$_H$	4F$_H$ / 79$_D$
Unassigned node	–	xx'0140$_H$	50$_H$ / 80$_D$
Unassigned node	–	xx'0150$_H$	54$_H$ / 84$_D$
Unassigned node	–	xx'0154$_H$	55$_H$ / 85$_D$
Unassigned node	–	xx'0158$_H$	56$_H$ / 86$_D$
Unassigned node	–	xx'015C$_H$	57$_H$ / 87$_D$
Unassigned node	–	xx'0160$_H$	58$_H$ / 88$_D$
Unassigned node	–	xx'0164$_H$	59$_H$ / 89$_D$
Unassigned node	–	xx'0168$_H$	5A$_H$ / 90$_D$
Unassigned node	–	xx'016C$_H$	5B$_H$ / 91$_D$
Unassigned node	–	xx'0170$_H$	5C$_H$ / 92$_D$

1) Register VECSEG defines the segment where the vector table is located to.
Bitfield VECSC in register CPUCON1 defines the distance between two adjacent vectors. This table represents the default setting, with a distance of 4 (two words) between two vectors.

The XC164LM also provides an excellent mechanism to identify and to process exceptions or error conditions that arise during run-time, so-called ‘Hardware Traps’. Hardware traps cause immediate non-maskable system reaction which is similar to a standard interrupt service (branching to a dedicated vector table location). The occurrence of a hardware trap is additionally signified by an individual bit in the trap flag register (TFR). Except when another higher prioritized trap service is in progress, a hardware trap will interrupt any actual program execution. In turn, hardware trap services can normally not be interrupted by standard or PEC interrupts.

Table 5 shows all of the possible exceptions or error conditions that can arise during run-time:

Table 5 Hardware Trap Summary

Exception Condition	Trap Flag	Trap Vector	Vector Location[1)]	Trap Number	Trap Priority
Reset Functions:	–				
• Hardware Reset		RESET	xx’0000$_H$	00$_H$	III
• Software Reset		RESET	xx’0000$_H$	00$_H$	III
• W-dog Timer Overflow		RESET	xx’0000$_H$	00$_H$	III
Class A Hardware Traps:					
• Non-Maskable Interrupt	NMI	NMITRAP	xx’0008$_H$	02$_H$	II
• Stack Overflow	STKOF	STOTRAP	xx’0010$_H$	04$_H$	II
• Stack Underflow	STKUF	STUTRAP	xx’0018$_H$	06$_H$	II
• Software Break	SOFTBRK	SBRKTRAP	xx’0020$_H$	08$_H$	II
Class B Hardware Traps:					
• Undefined Opcode	UNDOPC	BTRAP	xx’0028$_H$	0A$_H$	I
• PMI Access Error	PACER	BTRAP	xx’0028$_H$	0A$_H$	I
• Protected Instruction Fault	PRTFLT	BTRAP	xx’0028$_H$	0A$_H$	I
• Illegal Word Operand Access	ILLOPA	BTRAP	xx’0028$_H$	0A$_H$	I
Reserved	–	–	[2C$_H$ - 3C$_H$]	[0B$_H$ - 0F$_H$]	–
Software Traps • TRAP Instruction	–	–	Any [xx’0000$_H$ - xx’01FC$_H$] in steps of 4$_H$	Any [00$_H$ - 7F$_H$]	Current CPU Priority

1) Register VECSEG defines the segment where the vector table is located to.
Bitfield VECSC in register CPUCON1 defines the distance between two adjacent vectors. This table represents the default setting, with a distance of 4 (two words) between two vectors.