Data Sheet, V1.0, Oct. 2002

# XC167CI

## 16-Bit Single-Chip Microcontroller

## Preliminary




Microcontrollers

Infineon technologies

Never stop thinking.

**Edition 2002-10**

**Published by Infineon Technologies AG,**
**St.-Martin-Strasse 53,**
**D-81541 München, Germany**

**© Infineon Technologies AG 2002.**
**All Rights Reserved.**

**Attention please!**

The information herein is given to describe certain components and shall not be considered as warranted characteristics.

Terms of delivery and rights to technical change reserved.

We hereby disclaim any and all warranties, including but not limited to warranties of non-infringement, regarding circuits, descriptions and charts stated herein.

Infineon Technologies is an approved CECC manufacturer.

**Information**

For further information on technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies Office in Germany or our Infineon Technologies Representatives worldwide (see address list).

**Warnings**

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

Infineon Technologies Components may only be used in life-support devices or systems 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.

# XC167CI

## 16-Bit Single-Chip Microcontroller

## Preliminary

Preliminary

Microcontrollers

Infineon technologies

Never stop thinking.

**XC167**
**Preliminary**
**Revision History: 2002-10** V1.0

Previous Version: ---

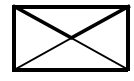
| Page | Subjects (major changes since last revision) |
|---|---|
| | |
| | |
| | |

Controller Area Network (CAN): License of Robert Bosch GmbH

**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**



**Preliminary**

**16-Bit Single-Chip Microcontroller** **XC167**
**XC166 Family**

**XC167**

# 1 Summary of Features

- 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 77 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)
  - 4 Kbytes On-Chip Data SRAM (DSRAM)
  - 2 Kbytes On-Chip Program/Data SRAM (PSRAM)
  - 128 Kbytes On-Chip Program Memory (Flash Memory)
- On-Chip Peripheral Modules
  - 16-Channel A/D Converter with Programmable Resolution (10-bit or 8-bit) and Conversion Time (down to 2.95 µs or 2.55 µs)
  - Two 16-Channel General Purpose Capture/Compare Units (32 Input/Output Pins)
  - Capture/Compare Unit for flexible PWM Signal Generation (CAPCOM6) (3/6 Capture/Compare Channels and 1 Compare Channel)
  - Multi-Functional General Purpose Timer Unit with 5 Timers
  - Two Synchronous/Asynchronous Serial Channels (USARTs)
  - Two High-Speed-Synchronous Serial Channels
  - On-Chip TwinCAN Interface (Rev. 2.0B active) with 32 Message Objects (Full CAN/Basic CAN) on Two CAN Nodes, and Gateway Functionality
  - IIC Bus Interface (10-bit addressing, 400 kbit/s) with 3 Channels (multiplexed)
  - On-Chip Real Time Clock, Driven by Dedicated Oscillator
- Idle, Sleep, and Power Down Modes with Flexible Power Management
- Programmable Watchdog Timer and Oscillator Watchdog

- Up to 12 Mbytes External Address Space for Code and Data
  - Programmable External Bus Characteristics for Different Address Ranges
  - Multiplexed or Demultiplexed External Address/Data Buses
  - Selectable Address Bus Width
  - 16-Bit or 8-Bit Data Bus Width
  - Five Programmable Chip-Select Signals
  - Hold- and Hold-Acknowledge Bus Arbitration Support
- Up to 103 General Purpose I/O Lines,
  partly with Selectable Input Thresholds and Hysteresis
- On-Chip Bootstrap Loader
- Supported by a Large Range of Development Tools like C-Compilers, Macro-Assembler Packages, Emulators, Evaluation Boards, HLL-Debuggers, Simulators, Logic Analyzer Disassemblers, Programming Boards
- On-Chip Debug Support via JTAG Interface
- 144-Pin TQFP Package, 0.5 mm (19.7 mil) pitch

## 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 XC167 please refer to the **“Product Catalog Microcontrollers”**, which summarizes all available microcontroller variants.

*Note: The ordering codes for Mask-ROM versions are defined for each product after verification of the respective ROM code.*

This document describes several derivatives of the XC167 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 **XC167** throughout this document.

**Table 1 XC167 Derivative Synopsis**

| Derivative[1)] | Program Memory | On-Chip RAM | Interfaces |
|---|---|---|---|
| SAK-XC167CI-16F40F,<br>SAK-XC167CI-16F20F | 128 Kbytes Flash | 2 Kbytes DPRAM,<br>4 Kbytes DSRAM,<br>2 Kbytes PSRAM | ASC0, ASC1,<br>SSC0, SSC1,<br>CAN0, CAN1,<br>IIC |
| SAF-XC167CI-16F40F,<br>SAF-XC167CI-16F20F | 128 Kbytes Flash | 2 Kbytes DPRAM,<br>4 Kbytes DSRAM,<br>2 Kbytes PSRAM | ASC0, ASC1,<br>SSC0, SSC1,<br>CAN0, CAN1,<br>IIC |
| SAB-XC167CI-16F40F,<br>SAB-XC167CI-16F20F<br>(on request) | 128 Kbytes Flash | 2 Kbytes DPRAM,<br>4 Kbytes DSRAM,<br>2 Kbytes PSRAM | ASC0, ASC1,<br>SSC0, SSC1,<br>CAN0, CAN1,<br>IIC |

1) This Data Sheet is valid for devices starting with and including design step AC.

# 2 General Device Information

## 2.1 Introduction

The XC167 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.

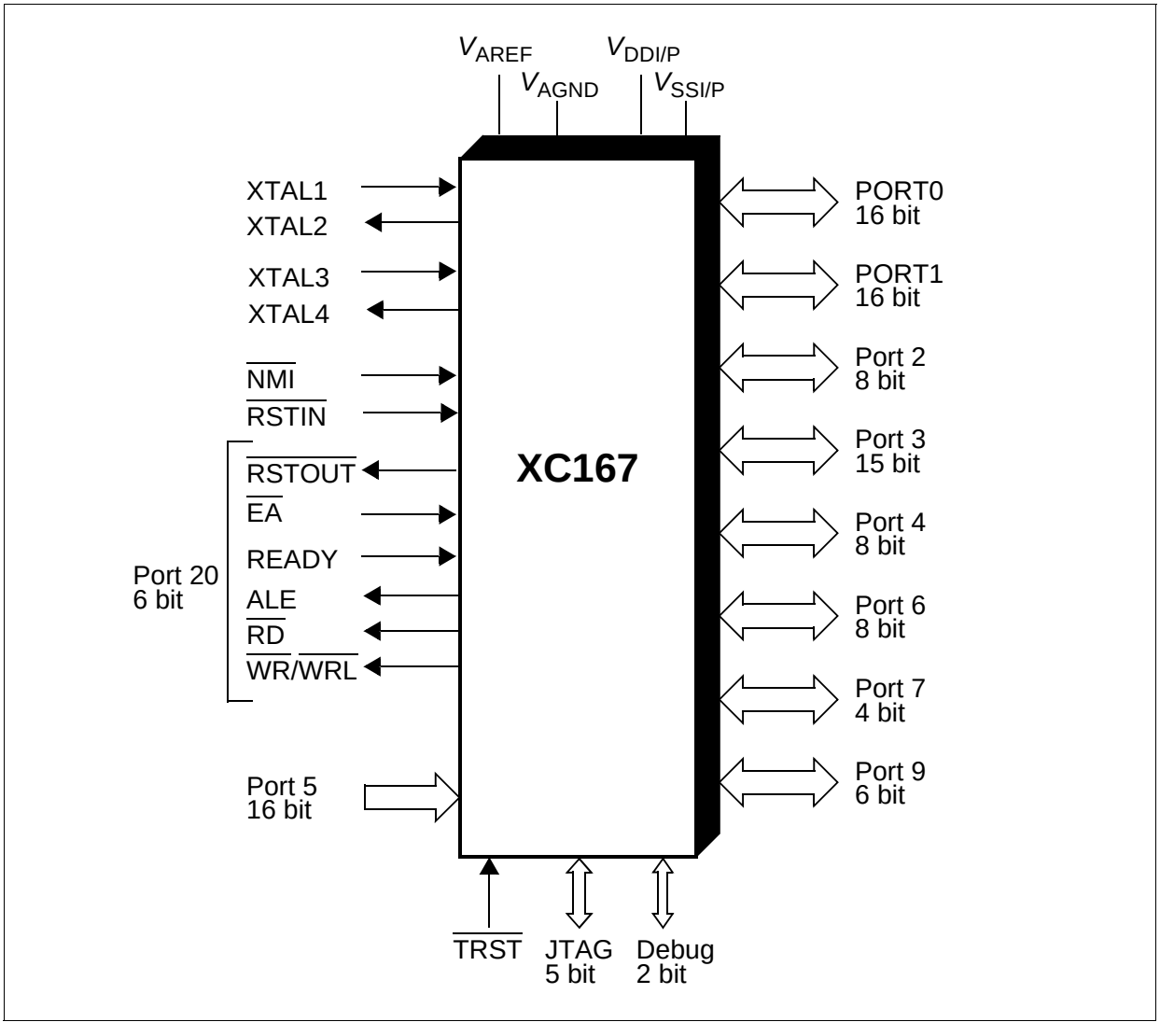



**Figure 1 Logic Symbol**

## 2.2 Pin Configuration and Definition

The pins of the XC167 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*) and C*) mark pins to be used as alternate external interrupt inputs, C*) marks pins that can have CAN interface lines assigned to them.

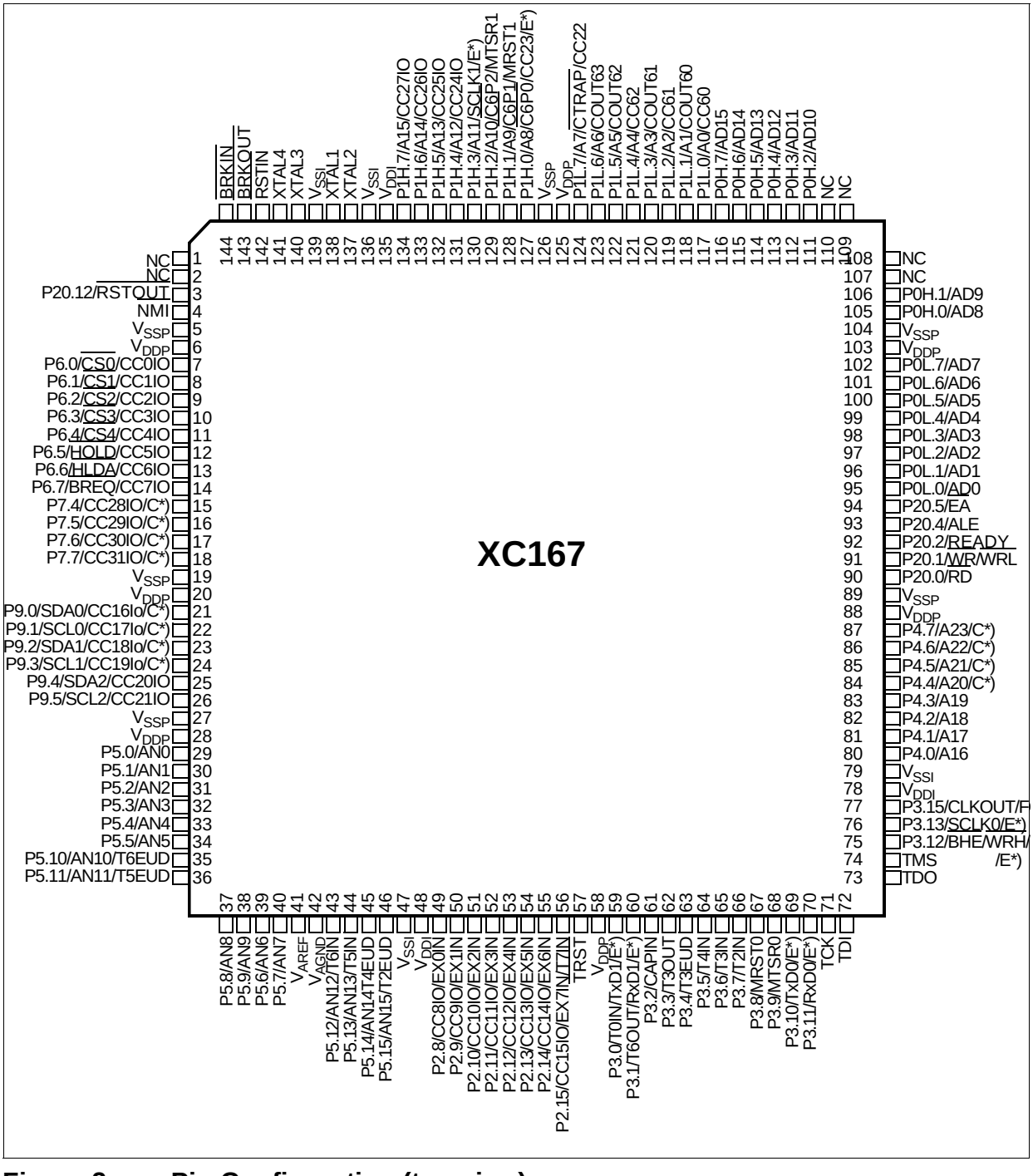



**Figure 2 Pin Configuration (top view)**

**Table 2 Pin Definitions and Functions**

| Symbol | Pin Num. | Input Outp. | Function |
|---|---|---|---|
| P20.12 | 3 | IO | For details, please refer to the description of **P20**. |
| NMI | 4 | 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 NMI pin must be low in order to force the XC167 into power down mode. If NMI is high, when PWRDN is executed, the part will continue to run in normal mode.<br>If not used, pin NMI should be pulled high externally. |
| **P6** | | IO | Port 6 is an 8-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 6 is selectable (standard or special).<br>The Port 6 pins also serve for alternate functions: |
| P6.0 | 7 | O | CS0 Chip Select 0 Output, |
| | | IO | CC0IO CAPCOM1: CC0 Capture Inp./Compare Output |
| P6.1 | 8 | O | CS1 Chip Select 1 Output, |
| | | IO | CC1IO CAPCOM1: CC1 Capture Inp./Compare Output |
| P6.2 | 9 | O | CS2 Chip Select 2 Output, |
| | | IO | CC2IO CAPCOM1: CC2 Capture Inp./Compare Output |
| P6.3 | 10 | O | CS3 Chip Select 3 Output, |
| | | IO | CC3IO CAPCOM1: CC3 Capture Inp./Compare Output |
| P6.4 | 11 | O | CS4 Chip Select 4 Output, |
| | | IO | CC4IO CAPCOM1: CC4 Capture Inp./Compare Output |
| P6.5 | 12 | I | HOLD External Master Hold Request Input, |
| | | IO | CC5IO CAPCOM1: CC5 Capture Inp./Compare Output |
| P6.6 | 13 | I/O | HLDA Hold Acknowledge Output (master mode) or Input (slave mode), |
| | | IO | CC6IO CAPCOM1: CC6 Capture Inp./Compare Output |
| P6.7 | 14 | O | BREQ Bus Request Output, |
| | | IO | CC7IO CAPCOM1: CC7 Capture Inp./Compare Output |

**Table 2 Pin Definitions and Functions** (cont'd)

| Symbol | Pin Num. | Input Outp. | Function |
|---|---|---|---|
| **P7** | | IO | Port 7 is a 4-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 7 is selectable (standard or special).<br>Port 7 pins provide inputs/outputs for CAPCOM2 and serial interface lines.[1)] |
| P7.4 | 15 | I/O | CC28IO CAPCOM2: CC28 Capture Inp./Compare Outp., |
| | | I | CAN2_RxD CAN Node 2 Receive Data Input, |
| | | I | EX7IN Fast External Interrupt 7 Input (alternate pin B) |
| P7.5 | 16 | I/O | CC29IO CAPCOM2: CC29 Capture Inp./Compare Outp., |
| | | O | CAN2_TxD CAN Node 2 Transmit Data Output, |
| | | I | EX6IN Fast External Interrupt 6 Input (alternate pin B) |
| P7.6 | 17 | I/O | CC30IO CAPCOM2: CC30 Capture Inp./Compare Outp., |
| | | I | CAN1_RxD CAN Node 1 Receive Data Input, |
| | | I | EX7IN Fast External Interrupt 7 Input (alternate pin A) |
| P7.7 | 18 | I/O | CC31IO CAPCOM2: CC31 Capture Inp./Compare Outp., |
| | | O | CAN1_TxD CAN Node 1 Transmit Data Output, |
| | | I | EX6IN Fast External Interrupt 6 Input (alternate pin A) |

**Table 2 Pin Definitions and Functions** (cont'd)

| Symbol | Pin Num. | Input Outp. | Function |
|---|---|---|---|
| **P9** | | 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).<br>The following Port 9 pins also serve for alternate functions:[1)] |
| P9.0 | 21 | I/O<br>I<br>I/O | CC16IO CAPCOM2: CC16 Capture Inp./Compare Outp.,<br>CAN2_RxD CAN Node 2 Receive Data Input,<br>SDA0 IIC Bus Data Line 0 |
| P9.1 | 22 | I/O<br>O<br>I/O | CC17IO CAPCOM2: CC17 Capture Inp./Compare Outp.,<br>CAN2_TxD CAN Node 2 Transmit Data Output,<br>SCL0 IIC Bus Clock Line 0 |
| P9.2 | 23 | I/O<br>I<br>I/O | CC18IO CAPCOM2: CC18 Capture Inp./Compare Outp.,<br>CAN1_RxD CAN Node 1 Receive Data Input,<br>SDA1 IIC Bus Data Line 1 |
| P9.3 | 24 | I/O<br>O<br>I/O | CC19IO CAPCOM2: CC19 Capture Inp./Compare Outp.,<br>CAN1_TxD CAN Node 1 Transmit Data Output,<br>SCL1 IIC Bus Clock Line 1 |
| P9.4 | 25 | I/O<br>I/O | CC20IO CAPCOM2: CC20 Capture Inp./Compare Outp.,<br>SDA2 IIC Bus Data Line 2 |
| P9.5 | 26 | I/O<br>I/O | CC21IO CAPCOM2: CC21 Capture Inp./Compare Outp.,<br>SCL2 IIC Bus Clock Line 2 |

**Table 2 Pin Definitions and Functions** (cont'd)

| Symbol | Pin Num. | Input Outp. | Function | | |
|---|---|---|---|---|---|
| **P5** | | I | Port 5 is a 16-bit input-only port.<br>The pins of Port 5 also serve as analog input channels for the A/D converter, or they serve as timer inputs: | | |
| P5.0 | 29 | I | AN0 | | |
| P5.1 | 30 | I | AN1 | | |
| P5.2 | 31 | I | AN2 | | |
| P5.3 | 32 | I | AN3 | | |
| P5.4 | 33 | I | AN4 | | |
| P5.5 | 34 | I | AN5 | | |
| P5.10 | 35 | I | AN10, | T6EUD | GPT2 Timer T6 Ext. Up/Down Ctrl. Inp. |
| P5.11 | 36 | I | AN11, | T5EUD | GPT2 Timer T5 Ext. Up/Down Ctrl. Inp. |
| P5.8 | 37 | I | AN8 | | |
| P5.9 | 38 | I | AN9 | | |
| P5.6 | 39 | I | AN6 | | |
| P5.7 | 40 | I | AN7 | | |
| P5.12 | 43 | I | AN12, | T6IN | GPT2 Timer T6 Count/Gate Input |
| P5.13 | 44 | I | AN13, | T5IN | GPT2 Timer T5 Count/Gate Input |
| P5.14 | 45 | I | AN14, | T4EUD | GPT1 Timer T4 Ext. Up/Down Ctrl. Inp. |
| P5.15 | 46 | I | AN15, | T2EUD | GPT1 Timer T2 Ext. Up/Down Ctrl. Inp. |

**Table 2 Pin Definitions and Functions** (cont'd)

| Symbol | Pin Num. | Input Outp. | Function |
|---|---|---|---|
| **P2** | | IO | Port 2 is an 8-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 2 is selectable (standard or special).<br>The following Port 2 pins also serve for alternate functions: |
| P2.8 | 49 | I/O<br>I | CC8IO CAPCOM1: CC8 Capture Inp./Compare Output,<br>EX0IN Fast External Interrupt 0 Input (default pin) |
| P2.9 | 50 | I/O<br>I | CC9IO CAPCOM1: CC9 Capture Inp./Compare Output,<br>EX1IN Fast External Interrupt 1 Input (default pin) |
| P2.10 | 51 | I/O<br>I | CC10IO CAPCOM1: CC10 Capture Inp./Compare Outp.,<br>EX2IN Fast External Interrupt 2 Input (default pin) |
| P2.11 | 52 | I/O<br>I | CC11IO CAPCOM1: CC11 Capture Inp./Compare Outp.,<br>EX3IN Fast External Interrupt 3 Input (default pin) |
| P2.12 | 53 | I/O<br>I | CC12IO CAPCOM1: CC12 Capture Inp./Compare Outp.,<br>EX4IN Fast External Interrupt 4 Input (default pin) |
| P2.13 | 54 | I/O<br>I | CC13IO CAPCOM1: CC13 Capture Inp./Compare Outp.,<br>EX5IN Fast External Interrupt 5 Input (default pin) |
| P2.14 | 55 | I/O<br>I | CC14IO CAPCOM1: CC14 Capture Inp./Compare Outp.,<br>EX6IN Fast External Interrupt 6 Input (default pin) |
| P2.15 | 56 | I/O<br>I<br>I | CC15IO CAPCOM1: CC15 Capture Inp./Compare Outp.,<br>EX7IN Fast External Interrupt 7 Input (default pin),<br>T7IN CAPCOM2: Timer T7 Count Input |
| $\overline{\text{TRST}}$ | 57 | I | Test-System Reset Input. A high level at this pin activates the XC167's debug system.<br>*Note: For normal system operation, pin $\overline{TRST}$ should be held low.* |

**Table 2 Pin Definitions and Functions** (cont'd)

| Symbol | Pin Num. | Input Outp. | Function | |
|---|---|---|---|---|
| **P3** | | IO | Port 3 is a 15-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).<br>The following Port 3 pins also serve for alternate functions: | |
| P3.0 | 59 | I | T0IN | CAPCOM1 Timer T0 Count Input, |
| | | O | TxD1 | ASC1 Clock/Data Output (Async./Sync), |
| | | I | EX1IN | Fast External Interrupt 1 Input (alternate pin B) |
| P3.1 | 60 | 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) |
| P3.2 | 61 | I | CAPIN | GPT2 Register CAPREL Capture Input |
| P3.3 | 62 | O | T3OUT | GPT1 Timer T3 Toggle Latch Output |
| P3.4 | 63 | I | T3EUD | GPT1 Timer T3 External Up/Down Control Input |
| P3.5 | 64 | I | T4IN | GPT1 Timer T4 Count/Gate/Reload/Capture Inp |
| P3.6 | 65 | I | T3IN | GPT1 Timer T3 Count/Gate Input |
| P3.7 | 66 | I | T2IN | GPT1 Timer T2 Count/Gate/Reload/Capture Inp |
| P3.8 | 67 | I/O | MRST0 | SSC0 Master-Receive/Slave-Transmit In/Out. |
| P3.9 | 68 | I/O | MTSR0 | SSC0 Master-Transmit/Slave-Receive Out/In. |
| P3.10 | 69 | O | TxD0 | ASC0 Clock/Data Output (Async./Sync.), |
| | | I | EX2IN | Fast External Interrupt 2 Input (alternate pin B) |
| P3.11 | 70 | I/O | RxD0 | ASC0 Data Input (Async.) or Inp./Outp. (Sync.), |
| | | I | EX2IN | Fast External Interrupt 2 Input (alternate pin A) |
| P3.12 | 75 | O | $\overline{\text{BHE}}$ | External Memory High Byte Enable Signal, |
| | | O | $\overline{\text{WRH}}$ | External Memory High Byte Write Strobe, |
| | | I | EX3IN | Fast External Interrupt 3 Input (alternate pin B) |
| P3.13 | 76 | I/O | SCLK0 | SSC0 Master Clock Output / Slave Clock Input., |
| | | I | EX3IN | Fast External Interrupt 3 Input (alternate pin A) |
| P3.15 | 77 | O | CLKOUT | Master Clock Output, |
| | | O | FOUT | Programmable Frequency Output |
| TCK | 71 | I | Debug System: JTAG Clock Input | |
| TDI | 72 | I | Debug System: JTAG Data In | |
| TDO | 73 | O | Debug System: JTAG Data Out | |
| TMS | 74 | I | Debug System: JTAG Test Mode Selection | |

**Table 2 Pin Definitions and Functions** (cont'd)

| Symbol | Pin Num. | Input Outp. | Function | |
|---|---|---|---|---|
| **P4** | | IO | Port 4 is an 8-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 4 is selectable (standard or special).<br>Port 4 can be used to output the segment address lines, the optional chip select lines, and for serial interface lines:[1)] | |
| P4.0 | 80 | O | A16 | Least Significant Segment Address Line |
| P4.1 | 81 | O | A17 | Segment Address Line |
| P4.2 | 82 | O | A18 | Segment Address Line |
| P4.3 | 83 | O | A19 | Segment Address Line |
| P4.4 | 84 | O | A20 | Segment Address Line, |
| | | I | CAN2_RxD | CAN Node 2 Receive Data Input, |
| | | I | EX5IN | Fast External Interrupt 5 Input (alternate pin B) |
| P4.5 | 85 | O | A21 | Segment Address Line, |
| | | I | CAN1_RxD | CAN Node 1 Receive Data Input, |
| | | I | EX4IN | Fast External Interrupt 4 Input (alternate pin B) |
| P4.6 | 86 | O | A22 | Segment Address Line, |
| | | O | CAN1_TxD | CAN Node 1 Transmit Data Output, |
| | | I | EX5IN | Fast External Interrupt 5 Input (alternate pin A) |
| P4.7 | 87 | O | A23 | Most Significant Segment Address Line, |
| | | I | CAN1_RxD | CAN Node 1 Receive Data Input, |
| | | O | CAN2_TxD | CAN Node 2 Transmit Data Output, |
| | | I | EX4IN | Fast External Interrupt 4 Input (alternate pin A) |

**Table 2 Pin Definitions and Functions** (cont'd)

| Symbol | Pin Num. | Input Outp. | Function | |
|---|---|---|---|---|
| **P20** | | IO | Port 20 is a 6-bit bidirectional I/O port. Each pin can be programmed for input (output driver in high-impedance state) or output. The input threshold of Port 20 is selectable (standard or special).<br>The following Port 20 pins also serve for alternate functions: | |
| P20.0 | 90 | O | $\overline{RD}$ | External Memory Read Strobe, activated for every external instruction or data read access. |
| P20.1 | 91 | O | $\overline{WR}$/$\overline{WRL}$ | External Memory Write Strobe.<br>In $\overline{WR}$-mode this pin is activated for every external data write access.<br>In $\overline{WRL}$-mode this pin is activated for low byte data write accesses on a 16-bit bus, and for every data write access on an 8-bit bus. |
| P20.2 | 92 | I | READY | READY Input. When the READY function is enabled, memory cycle time waitstates can be forced via this pin during an external access. |
| P20.4 | 93 | O | ALE | Address Latch Enable Output.<br>Can be used for latching the address into external memory or an address latch in the multiplexed bus modes. |
| P20.5 | 94 | I | $\overline{EA}$ | External Access Enable pin.<br>**A low level** at this pin during and after Reset forces the XC167 to latch the configuration from PORT0 and pin $\overline{RD}$, and to begin instruction execution out of external memory.<br>**A high level** forces the XC167 to latch the configuration from pins $\overline{RD}$, ALE, and $\overline{WR}$, and to begin instruction execution out of the internal program memory. "ROMless" versions must have this pin tied to '0'. |
| P20.12 | 3 | O | $\overline{RSTOUT}$ | Internal Reset Indication Output.<br>**Is activated** asynchronously with an external hardware reset. It may also be activated (selectable) synchronously with an internal software or watchdog reset.<br>**Is deactivated** upon the execution of the EINIT instruction, optionally at the end of reset, or at any time (before EINIT) via user software. |
| | | | *Note: Port 20 pins may input configuration values (see $\overline{EA}$).* | |

**Table 2 Pin Definitions and Functions** (cont'd)

| Symbol | Pin Num. | Input Outp. | Function |
|---|---|---|---|
| **PORT0** | | IO | PORT0 consists of the two 8-bit bidirectional I/O ports P0L and P0H. Each pin can be programmed for input (output driver in high-impedance state) or output.<br>In case of an external bus configuration, PORT0 serves as the address (A) and address/data (AD) bus in multiplexed bus modes and as the data (D) bus in demultiplexed bus modes.<br>**Demultiplexed bus modes:** |
| P0L.0-7 | 95 - 102 | | Data Path Width: 8-bit 16-bit<br>P0L.0 – P0L.7: D0 – D7 D0 - D7 |
| P0H.0-1 | 105 - 106 | | P0H.0 – P0H.7: I/O D8 - D15<br>**Multiplexed bus modes:** |
| P0H.2-7 | 111 - 116 | | Data Path Width: 8-bit 16-bit<br>P0L.0 – P0L.7: AD0 – AD7 AD0 - AD7<br>P0H.0 – P0H.7: A8 - A15 AD8 - AD15<br>*Note: At the end of an external reset ($\overline{EA}$ = '0') PORT0 also may input configuration values.* |
| **PORT1** | | IO | PORT1 consists of the two 8-bit bidirectional I/O ports P1L and P1H. It is bit-wise programmable for input or output via direction bits. For a pin configured as input, the output driver is put into high-impedance state. PORT1 is used as the 16-bit address bus (A) in demultiplexed bus modes (also after switching from a demultiplexed to a multiplexed bus mode).<br>The following PORT1 pins also serve for alt. functions: |
| P1L.0 | 117 | I/O | CC60 CAPCOM6: Input / Output of Channel 0 |
| P1L.1 | 118 | O | COUT60 CAPCOM6: Output of Channel 0 |
| P1L.2 | 119 | I/O | CC61 CAPCOM6: Input / Output of Channel 1 |
| P1L.3 | 120 | O | COUT61 CAPCOM6: Output of Channel 1 |
| P1L.4 | 121 | I/O | CC62 CAPCOM6: Input / Output of Channel 2 |
| P1L.5 | 122 | O | COUT62 CAPCOM6: Output of Channel 2 |
| P1L.6 | 123 | O | $\overline{COUT63}$ Output of 10-bit Compare Channel |
| P1L.7 | 124 | I | $\overline{CTRAP}$ CAPCOM6: Trap Input<br>$\overline{CTRAP}$ is an input pin with an internal pullup resistor. A low level on this pin switches the CAPCOM6 compare outputs to the logic level defined by software (if enabled). |
| P1H | … | I/O | CC22IO CAPCOM2: CC22 Capture Inp./Compare Outp.<br>…continued… |

**Table 2 Pin Definitions and Functions** (cont'd)

| Symbol | Pin Num. | Input Outp. | Function |
|---|---|---|---|
| **PORT1** (cont'd) | | IO | …continued… |
| P1H.0 | 127 | I | $\overline{\text{CC6POS0}}$ CAPCOM6: Position 0 Input, |
| | | I | EX0IN Fast External Interrupt 0 Input (alternate pin B), |
| | | I/O | $\overline{\text{CC23IO}}$ CAPCOM2: CC23 Capture Inp./Compare Outp. |
| P1H.1 | 128 | I | $\overline{\text{CC6POS1}}$ CAPCOM6: Position 1 Input, |
| | | I/O | $\overline{\text{MRST1}}$ SSC1 Master-Receive/Slave-Transmit In/Out. |
| P1H.2 | 129 | I | $\overline{\text{CC6POS2}}$ CAPCOM6: Position 2 Input, |
| | | I/O | MTSR1 SSC1 Master-Transmit/Slave-Receive Out/Inp. |
| P1H.3 | 130 | I/O | SCLK1 SSC1 Master Clock Output / Slave Clock Input, |
| | | I | EX0IN Fast External Interrupt 0 Input (alternate pin A) |
| P1H.4 | 131 | I/O | CC24IO CAPCOM2: CC24 Capture Inp./Compare Outp. |
| P1H.5 | 132 | I/O | CC25IO CAPCOM2: CC25 Capture Inp./Compare Outp. |
| P1H.6 | 133 | I/O | CC26IO CAPCOM2: CC26 Capture Inp./Compare Outp. |
| P1H.7 | 134 | I/O | CC27IO CAPCOM2: CC27 Capture Inp./Compare Outp. |
| XTAL2 | 137 | O | XTAL2: Output of the main oscillator amplifier circuit |
| XTAL1 | 138 | I | XTAL1: Input to the main oscillator amplifier and input to the internal clock generator<br>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. |
| XTAL3 | 140 | I | XTAL3: Input to the auxiliary (32-kHz) oscillator amplifier |
| XTAL4 | 141 | O | XTAL4: Output of the auxiliary (32-kHz) oscillator amplifier circuit<br>To clock the device from an external source, drive XTAL3, while leaving XTAL4 unconnected. Minimum and maximum high/low and rise/fall times specified in the AC Characteristics must be observed. |

**Table 2 Pin Definitions and Functions** (cont'd)

| Symbol | Pin Num. | Input Outp. | Function |
|---|---|---|---|
| $\overline{\text{RSTIN}}$ | 142 | I | Reset Input with Schmitt-Trigger characteristics. A low level at this pin while the oscillator is running resets the XC167. An internal pullup resistor permits power-on reset using only a capacitor connected to $V_{SS}$.<br>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.<br>*Note: To let the reset configuration of PORT0 settle and to let the PLL lock a reset duration of ca. 1 ms is recommended.* |
| $\overline{\text{BRK OUT}}$ | 143 | O | Debug System: Break Out |
| $\overline{\text{BRKIN}}$ | 144 | I | Debug System: Break In |
| NC | 1, 2, 107 - 110 | - | No connection.<br>It is recommended not to connect these pins to the PCB. |
| $V_{AREF}$ | 41 | - | Reference voltage for the A/D converter. |
| $V_{AGND}$ | 42 | - | Reference ground for the A/D converter. |
| $V_{DDI}$ | 48, 78, 135 | - | Digital Core Supply Voltage (On-Chip Modules):<br>+2.5 V during normal operation and idle mode.<br>Please refer to the **Operating Conditions** |
| $V_{DDP}$ | 6, 20, 28, 58, 88, 103, 125 | - | Digital Pad Supply Voltage (Pin Output Drivers):<br>+5 V during normal operation and idle mode.<br>Please refer to the **Operating Conditions** |
| $V_{SSI}$ | 47, 79, 136, 139 | - | **Digital Ground.**<br>Connect decoupling capacitors to adjacent $V_{DD}/V_{SS}$ pin pairs as close as possible to the pins.<br>All $V_{SS}$ pins must be connected to the ground-line or ground-plane. |
| $V_{SSP}$ | 5, 19, 27, 89, 104, 126 | - | |

1) The CAN interface lines are assigned to ports P4, P7, and P9 under software control.

# 3 Functional Description

The architecture of the XC167 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 resoures as well as external resources (see **Figure 3**).

This bus structure enhances the overall system performance by enabling the concurrent operation of several subsystems of the XC167.

The following block diagram gives an overview of the different on-chip components and of the advanced, high bandwidth internal bus structure of the XC167.

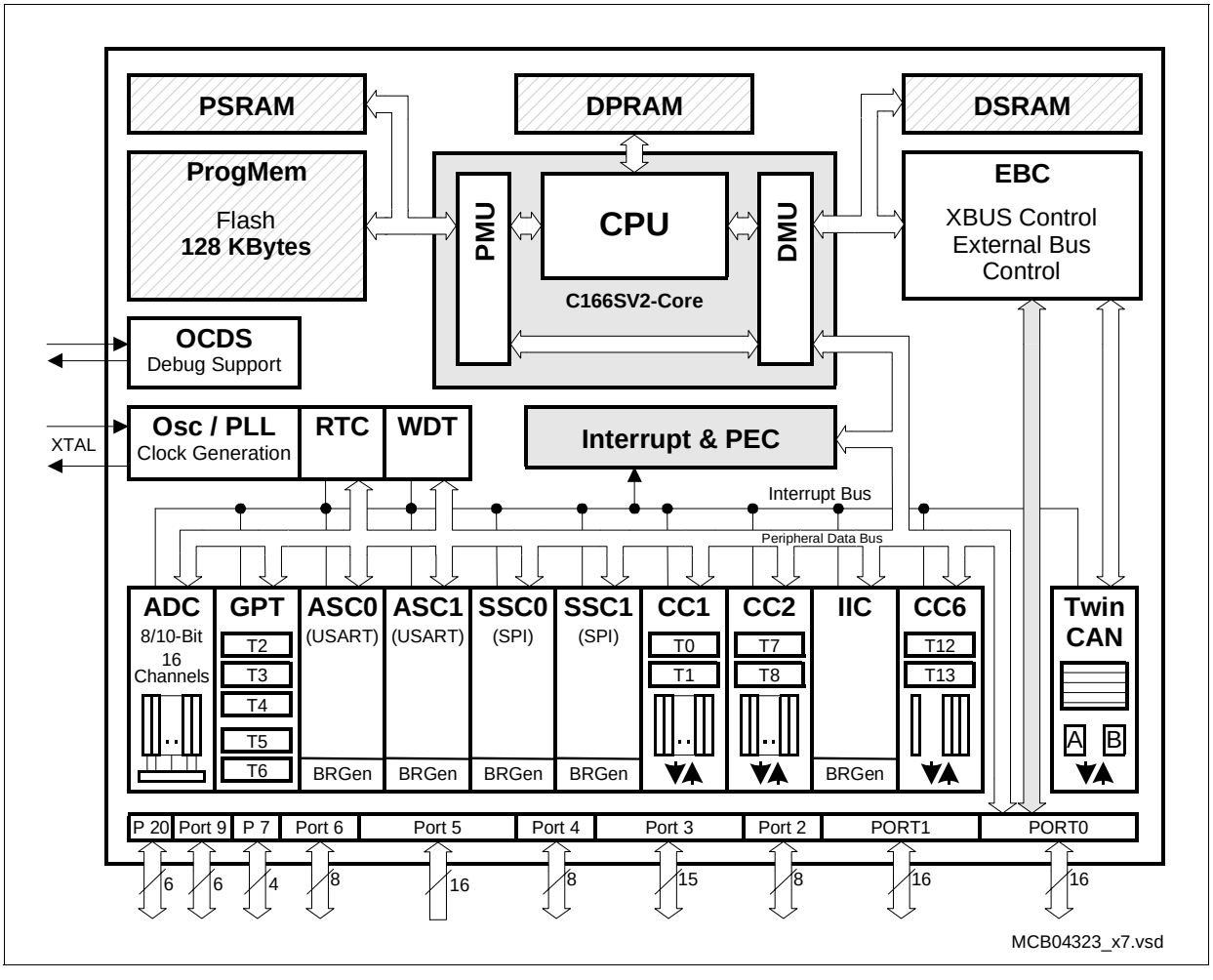



**Figure 3 Block Diagram**

## 3.1 Memory Subsystem and Organization

The memory space of the XC167 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 bytewise or wordwise. Portions of the on-chip DPRAM and the register spaces (E/SFR) have additionally been made directly bitaddressable.

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, code or data is written to the PSRAM, code is fetched from external memory, or data is read from or written to external resources, including peripherals on the LXbus (such as TwinCAN). The system bus allows concurrent two-way communication for maximum transfer performance.

**128 Kbytes of on-chip Flash memory** store code or constant data. The on-chip Flash memory is organized as four 8-Kbyte sectors, one 32-Kbyte sector, and one 64-Kbyte sector. Each sector can be separately write protected[1)], 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.

**4 Kbytes 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.

**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 can consist of up to 16 wordwide (R0 to R15) and/or bytewide (RL0, RH0, …, RL7, RH7)

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

so-called General Purpose Registers (GPRs).
The upper 256 bytes of the DPRAM are directly bitaddressable. When used by a GPR, any location in the DPRAM is bitaddressable.

**1024 bytes (2 × 512 bytes)** of the address space are reserved for the Special Function Register areas (SFR space and ESFR space). SFRs are wordwide 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.

In order to meet the needs of designs where more memory is required than is provided on chip, up to 12 Mbytes (approximately, see **Table 3**) of external RAM and/or ROM can be connected to the microcontroller. The External Bus Interface also provides access to external peripherals.

**Table 3 XC167 Memory Map[1)]**

| Address Area | Start Loc. | End Loc. | Area Size[2)] | Notes |
|---|---|---|---|---|
| Flash register space | $FF'F000_H$ | $FF'FFFF_H$ | 4 Kbytes | 3) |
| *Reserved (Acc. trap)* | $F8'0000_H$ | $FF'EFFF_H$ | <0.5 Mbytes | Minus Flash regs |
| *Reserved for PSRAM* | $E0'0800_H$ | $F7'FFFF_H$ | <1.5 Mbytes | Minus PSRAM |
| Program SRAM | $E0'0000_H$ | $E0'07FF_H$ | 2 Kbytes | Maximum |
| *Reserved for pr. mem.* | $C2'0000_H$ | $DF'FFFF_H$ | < 2 Mbytes | Minus Flash |
| Program Flash | $C0'0000_H$ | $C1'FFFF_H$ | 128 Kbytes | |
| *Reserved* | $BF'0000_H$ | $BF'FFFF_H$ | 64 Kbytes | |
| External memory area | $40'0000_H$ | $BE'FFFF_H$ | < 8 Mbytes | Minus res. seg. |
| External IO area[4)] | $20'0800_H$ | $3F'FFFF_H$ | < 2 Mbytes | Minus TwinCAN |
| TwinCAN registers | $20'0000_H$ | $20'07FF_H$ | 2 Kbytes | |
| External memory area | $01'0000_H$ | $1F'FFFF_H$ | < 2 Mbytes | Minus segment 0 |
| Data RAMs and SFRs | $00'8000_H$ | $00'FFFF_H$ | 32 Kbytes | Partly used |
| External memory area | $00'0000_H$ | $00'7FFF_H$ | 32 Kbytes | |

1) Accesses to the shaded areas generate external bus accesses.

2) The areas marked with "<" are slightly smaller than indicated, see column "Notes".

3) Not defined register locations return a trap code.

4) Several pipeline optimizations are not active within the external IO area. This is necessary to control external peripherals properly.

## 3.2 External Bus Controller

All of the external memory accesses are performed by a particular on-chip External Bus Controller (EBC). It can be programmed either to Single Chip Mode when no external memory is required, or to one of four different external memory access modes[1)], which are as follows:

– 16 … 24-bit Addresses, 16-bit Data, Demultiplexed
– 16 … 24-bit Addresses, 16-bit Data, Multiplexed
– 16 … 24-bit Addresses, 8-bit Data, Multiplexed
– 16 … 24-bit Addresses, 8-bit Data, Demultiplexed

In the demultiplexed bus modes, addresses are output on PORT1 and data is input/ output on PORT0 or P0L, respectively. In the multiplexed bus modes both addresses and data use PORT0 for input/output. The high order address (segment) lines use Port 4. The number of active segment address lines is selectable, restricting the external address space to 8 Mbytes … 64 Kbytes. This is required when interface lines are assigned to Port 4.

Up to 5 external $\overline{\text{CS}}$ signals (4 windows plus default) can be generated in order to save external glue logic. External modules can directly be connected to the common address/ data bus and their individual select lines.

Access to very slow memories or modules with varying access times is supported via a particular ‘Ready’ function. The active level of the control input signal is selectable.

A $\overline{\text{HOLD}}$/$\overline{\text{HLDA}}$ protocol is available for bus arbitration and allows the sharing of external resources with other bus masters. The bus arbitration is enabled by software. After enabling, pins P6.7 … P6.5 ($\overline{\text{BREQ}}$, $\overline{\text{HLDA}}$, $\overline{\text{HOLD}}$) are automatically controlled by the EBC. In Master Mode (default after reset) the $\overline{\text{HLDA}}$ pin is an output. In Slave Mode pin $\overline{\text{HLDA}}$ is switched to input. This allows the direct connection of the slave controller to another master controller without glue logic.

Important timing characteristics of the external bus interface have been made programmable (via registers TCONCSx/FCONCSx) to allow the user the adaption of a wide range of different types of memories and external peripherals.

In addition, up to 4 independent address windows may be defined (via registers ADDRSELx) which control the access to different resources with different bus characteristics. These address windows are arranged hierarchically where window 4 overrides window 3, and window 2 overrides window 1. All accesses to locations not covered by these 4 address windows are controlled by TCONCS0/FCONCS0. The currently active window can generate a chip select signal.

The external bus timing is related to the rising edge of the reference clock output CLKOUT. The external bus protocol is compatible with that of the standard C166 Family.

1) Bus modes are switched dynamically if several address windows with different mode settings are used.

The EBC also controls accesses to resources connected to the on-chip LXBus. The LXBus is an internal representation of the external bus and allows accessing integrated peripherals and modules in the same way as external components.

The TwinCAN module is connected and accessed via the LXBus.

## 3.3 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.

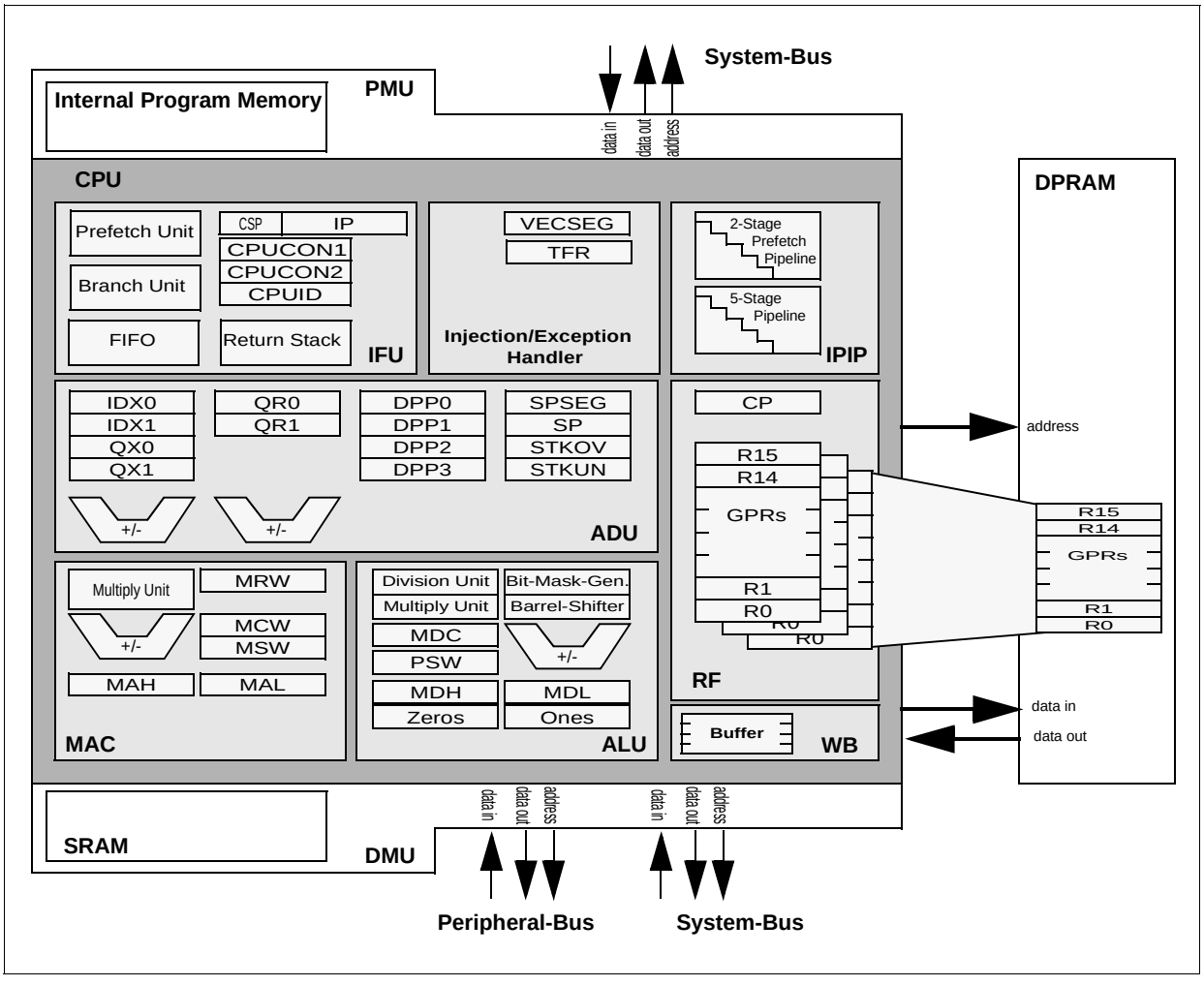



**Figure 4 CPU Block Diagram**

Based on these hardware provisions, most of the XC167'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 wordwide 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 XC167 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.

## 3.4 Interrupt System

With an interrupt response time of typically 8 CPU clocks (in case of internal program execution), the XC167 is capable of reacting very fast to the occurrence of non-deterministic events.

The architecture of the XC167 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 XC167 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 bitfield 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 XC167 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 XC167 Interrupt Nodes**

| Source of Interrupt or PEC Service Request | Control Register | Vector Location[1)] | Trap Number |
|---|---|---|---|
| CAPCOM Register 0 | CC1_CC0IC | xx'0040$_H$ | 10$_H$ / 16$_D$ |
| CAPCOM Register 1 | CC1_CC1IC | xx'0044$_H$ | 11$_H$ / 17$_D$ |
| CAPCOM Register 2 | CC1_CC2IC | xx'0048$_H$ | 12$_H$ / 18$_D$ |
| CAPCOM Register 3 | CC1_CC3IC | xx'004C$_H$ | 13$_H$ / 19$_D$ |
| CAPCOM Register 4 | CC1_CC4IC | xx'0050$_H$ | 14$_H$ / 20$_D$ |
| CAPCOM Register 5 | CC1_CC5IC | xx'0054$_H$ | 15$_H$ / 21$_D$ |
| CAPCOM Register 6 | CC1_CC6IC | xx'0058$_H$ | 16$_H$ / 22$_D$ |
| CAPCOM Register 7 | CC1_CC7IC | xx'005C$_H$ | 17$_H$ / 23$_D$ |
| CAPCOM Register 8 | CC1_CC8IC | xx'0060$_H$ | 18$_H$ / 24$_D$ |
| CAPCOM Register 9 | CC1_CC9IC | xx'0064$_H$ | 19$_H$ / 25$_D$ |
| CAPCOM Register 10 | CC1_CC10IC | xx'0068$_H$ | 1A$_H$ / 26$_D$ |
| CAPCOM Register 11 | CC1_CC11IC | xx'006C$_H$ | 1B$_H$ / 27$_D$ |
| CAPCOM Register 12 | CC1_CC12IC | xx'0070$_H$ | 1C$_H$ / 28$_D$ |
| CAPCOM Register 13 | CC1_CC13IC | xx'0074$_H$ | 1D$_H$ / 29$_D$ |
| CAPCOM Register 14 | CC1_CC14IC | xx'0078$_H$ | 1E$_H$ / 30$_D$ |
| CAPCOM Register 15 | CC1_CC15IC | xx'007C$_H$ | 1F$_H$ / 31$_D$ |
| 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'00E0$_H$ | 3C$_H$ / 60$_D$ |
| CAPCOM Register 29 | CC2_CC29IC | xx'0110$_H$ | 44$_H$ / 68$_D$ |

**Table 4 XC167 Interrupt Nodes** (cont'd)

| Source of Interrupt or PEC Service Request | Control Register | Vector Location[1)] | Trap Number |
|---|---|---|---|
| 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 0 | CC1_T0IC | xx'0080$_H$ | 20$_H$ / 32$_D$ |
| CAPCOM Timer 1 | CC1_T1IC | xx'0084$_H$ | 21$_H$ / 33$_D$ |
| 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$_D$ |
| GPT2 CAPREL Reg. | GPT12E_CRIC | xx'009C$_H$ | 27$_H$ / 39$_D$ |
| A/D Conversion Compl. | ADC_CIC | xx'00A0$_H$ | 28$_H$ / 40$_D$ |
| A/D Overrun Error | ADC_EIC | xx'00A4$_H$ | 29$_H$ / 41$_D$ |
| 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$_D$ |
| IIC Data Transfer Event | IIC_DTIC | xx'0100$_H$ | 40$_H$ / 64$_D$ |
| IIC Protocol Event | IIC_PEIC | xx'0104$_H$ | 41$_H$ / 65$_D$ |
| 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$_D$ |

**Table 4 XC167 Interrupt Nodes** (cont'd)

| Source of Interrupt or PEC Service Request | Control Register | Vector Location[1)] | Trap Number |
|---|---|---|---|
| End of PEC Subch. | EOPIC | xx'0130$_H$ | 4C$_H$ / 76$_D$ |
| CAPCOM6 Timer T12 | CCU6_T12IC | xx'0134$_H$ | 4D$_H$ / 77$_D$ |
| CAPCOM6 Timer T13 | CCU6_T13IC | xx'0138$_H$ | 4E$_H$ / 78$_D$ |
| CAPCOM6 Emergency | CCU6_EIC | xx'013C$_H$ | 4F$_H$ / 79$_D$ |
| CAPCOM6 | CCU6_IC | xx'0140$_H$ | 50$_H$ / 80$_D$ |
| 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$_D$ |
| CAN0 | CAN_0IC | xx'0150$_H$ | 54$_H$ / 84$_D$ |
| CAN1 | CAN_1IC | xx'0154$_H$ | 55$_H$ / 85$_D$ |
| CAN2 | CAN_2IC | xx'0158$_H$ | 56$_H$ / 86$_D$ |
| CAN3 | CAN_3IC | xx'015C$_H$ | 57$_H$ / 87$_D$ |
| CAN4 | CAN_4IC | xx'0164$_H$ | 59$_H$ / 89$_D$ |
| CAN5 | CAN_5IC | xx'0168$_H$ | 5A$_H$ / 90$_D$ |
| CAN6 | CAN_6IC | xx'016C$_H$ | 5B$_H$ / 91$_D$ |
| CAN7 | CAN_7IC | xx'0170$_H$ | 5C$_H$ / 92$_D$ |
| RTC | RTC_IC | xx'0174$_H$ | 5D$_H$ / 93$_D$ |
| Unassigned node | --- | xx'012C$_H$ | 4B$_H$ / 75$_D$ |
| Unassigned node | --- | xx'00FC$_H$ | 3F$_H$ / 63$_D$ |
| Unassigned node | --- | xx'0160$_H$ | 58$_H$ / 88$_D$ |

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 XC167 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 occurence 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<br>– 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.

## 3.5 On-Chip Debug Support (OCDS)

The On-Chip Debug Support system provides a broad range of debug and emulation features built into the XC167. The user software running on the XC167 can thus be debugged within the target system environment.

The OCDS is controlled by an external debugging device via the debug interface, consisting of the IEEE-1149-conforming JTAG port and a break interface. The debugger controls the OCDS via a set of dedicated registers accessible via the JTAG interface. Additionally, the OCDS system can be controlled by the CPU, e.g. by a monitor program. An injection interface allows the execution of OCDS-generated instructions by the CPU.

Multiple breakpoints can be triggered by on-chip hardware, by software, or by an external trigger input. Single stepping is supported as well as the injection of arbitrary instructions and read/write access to the complete internal address space. A breakpoint trigger can be answered with a CPU-halt, a monitor call, a data transfer, or/and the activation of an external signal.

Tracing data can be obtained via the JTAG interface or via the external bus interface for increased performance.

The debug interface uses a set of 6 interface signals (4 JTAG lines, 2 break lines) to communicate with external circuitry. These interface signals use dedicated pins.

Complete system emulation is supported by the New Emulation Technology (NET) interface. Via this full-featured emulation interface (including internal buses, control, status, and pad signals) the XC167 chip can be connected to a NET carrier chip.

The use of the XC167 production chip together with the carrier chip provides superior emulation behavior, because the emulation system shows exactly the same functionality as the production chip (use of the identical silicon).

## 3.6 Capture/Compare Units (CAPCOM1/2)

The CAPCOM units support generation and control of timing sequences on up to 32 channels with a maximum resolution of 1 system clock cycle (8 cycles in staggered mode). The CAPCOM units are typically used to handle high speed I/O tasks such as pulse and waveform generation, pulse width modulation (PMW), Digital to Analog (D/A) conversion, software timing, or time recording relative to external events.

Four 16-bit timers (T0/T1, T7/T8) with reload registers provide two independent time bases for each capture/compare register array.

The input clock for the timers is programmable to several prescaled values of the internal system clock, or may be derived from an overflow/underflow of timer T6 in module GPT2. This provides a wide range of variation for the timer period and resolution and allows precise adjustments to the application specific requirements. In addition, external count inputs for CAPCOM timers T0 and T7 allow event scheduling for the capture/compare registers relative to external events.

Both of the two capture/compare register arrays contain 16 dual purpose capture/compare registers, each of which may be individually allocated to either CAPCOM timer T0 or T1 (T7 or T8, respectively), and programmed for capture or compare function.
All registers of each module have each one port pin associated with it which serves as an input pin for triggering the capture function, or as an output pin to indicate the occurrence of a compare event.

**Table 6 Compare Modes (CAPCOM1/2)**

| Compare Modes | Function |
|---|---|
| Mode 0 | Interrupt-only compare mode;<br>several compare interrupts per timer period are possible |
| Mode 1 | Pin toggles on each compare match;<br>several compare events per timer period are possible |
| Mode 2 | Interrupt-only compare mode;<br>only one compare interrupt per timer period is generated |
| Mode 3 | Pin set ‘1’ on match; pin reset ‘0’ on compare timer overflow;<br>only one compare event per timer period is generated |
| Double Register Mode | Two registers operate on one pin;<br>pin toggles on each compare match;<br>several compare events per timer period are possible |
| Single Event Mode | Generates single edges or pulses;<br>can be used with any compare mode |

When a capture/compare register has been selected for capture mode, the current contents of the allocated timer will be latched (‘captured’) into the capture/compare register in response to an external event at the port pin which is associated with this register. In addition, a specific interrupt request for this capture/compare register is generated. Either a positive, a negative, or both a positive and a negative external signal transition at the pin can be selected as the triggering event.

The contents of all registers which have been selected for one of the five compare modes are continuously compared with the contents of the allocated timers.

When a match occurs between the timer value and the value in a capture/compare register, specific actions will be taken based on the selected compare mode.

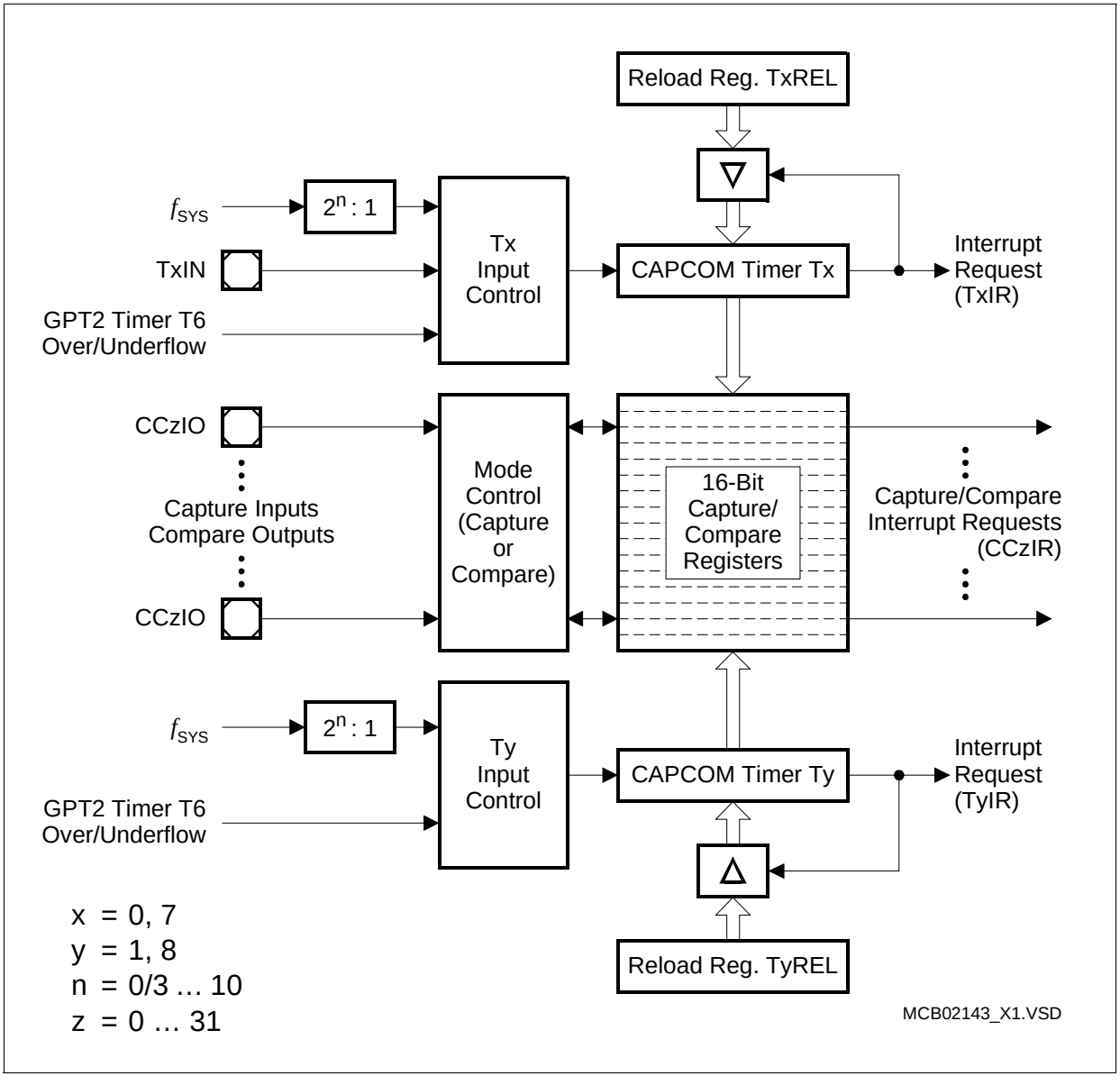
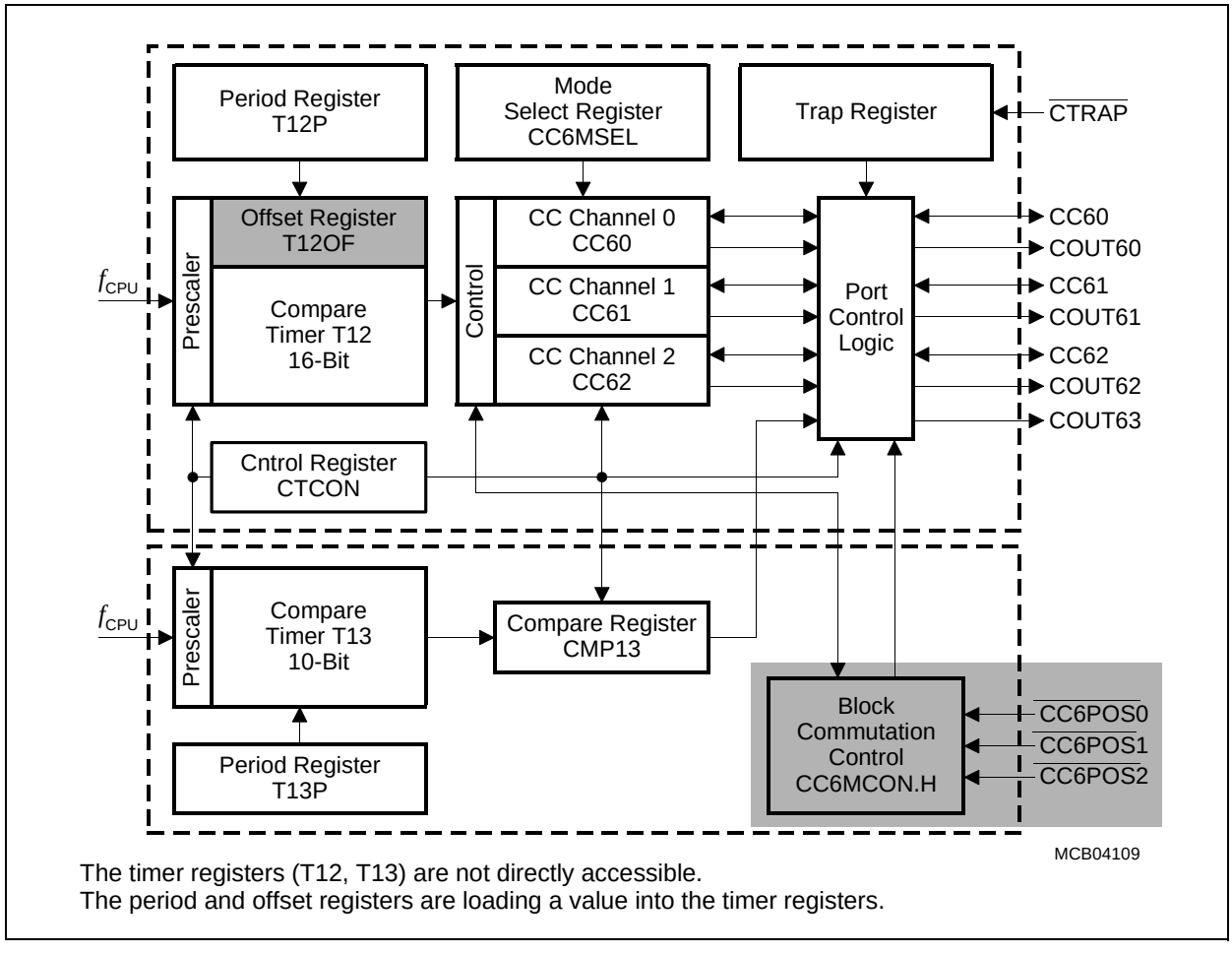



**Figure 5 CAPCOM1/2 Unit Block Diagram**

## 3.7 The Capture/Compare Unit CAPCOM6

The CAPCOM6 unit supports generation and control of timing sequences on up to three 16-bit capture/compare channels plus one independent 10-bit compare channel.
In compare mode the CAPCOM6 unit provides two output signals per channel which have inverted polarity and non-overlapping pulse transitions (deadtime control). The compare channel can generate a single PWM output signal and is further used to modulate the capture/compare output signals.
In capture mode the contents of compare timer T12 is stored in the capture registers upon a signal transition at pins CCx.

Compare timers T12 (16-bit) and T13 (10-bit) are free running timers which are clocked by the prescaled system clock.




The timer registers (T12, T13) are not directly accessible.
The period and offset registers are loading a value into the timer registers.

**Figure 6 CAPCOM6 Block Diagram**

For motor control applications both subunits may generate versatile multichannel PWM signals which are basically either controlled by compare timer T12 or by a typical hall sensor pattern at the interrupt inputs (block commutation).

## 3.8 General Purpose Timer (GPT12E) Unit

The GPT12E unit represents a very flexible multifunctional timer/counter structure which may be used for many different time related tasks such as event timing and counting, pulse width and duty cycle measurements, pulse generation, or pulse multiplication.

The GPT12E unit incorporates five 16-bit timers which are organized in two separate modules, GPT1 and GPT2. Each timer in each module may operate independently in a number of different modes, or may be concatenated with another timer of the same module.

Each of the three timers T2, T3, T4 of **module GPT1** can be configured individually for one of four basic modes of operation, which are Timer, Gated Timer, Counter, and Incremental Interface Mode. In Timer Mode, the input clock for a timer is derived from the system clock, divided by a programmable prescaler, while Counter Mode allows a timer to be clocked in reference to external events.
Pulse width or duty cycle measurement is supported in Gated Timer Mode, where the operation of a timer is controlled by the ‘gate’ level on an external input pin. For these purposes, each timer has one associated port pin (TxIN) which serves as gate or clock input. The maximum resolution of the timers in module GPT1 is 4 system clock cycles.

The count direction (up/down) for each timer is programmable by software or may additionally be altered dynamically by an external signal on a port pin (TxEUD) to facilitate e.g. position tracking.

In Incremental Interface Mode the GPT1 timers (T2, T3, T4) can be directly connected to the incremental position sensor signals A and B via their respective inputs TxIN and TxEUD. Direction and count signals are internally derived from these two input signals, so the contents of the respective timer Tx corresponds to the sensor position. The third position sensor signal TOP0 can be connected to an interrupt input.

Timer T3 has an output toggle latch (T3OTL) which changes its state on each timer over-flow/underflow. The state of this latch may be output on pin T3OUT e.g. for time out monitoring of external hardware components. It may also be used internally to clock timers T2 and T4 for measuring long time periods with high resolution.

In addition to their basic operating modes, timers T2 and T4 may be configured as reload or capture registers for timer T3. When used as capture or reload registers, timers T2 and T4 are stopped. The contents of timer T3 is captured into T2 or T4 in response to a signal at their associated input pins (TxIN). Timer T3 is reloaded with the contents of T2 or T4 triggered either by an external signal or by a selectable state transition of its toggle latch T3OTL. When both T2 and T4 are configured to alternately reload T3 on opposite state transitions of T3OTL with the low and high times of a PWM signal, this signal can be constantly generated without software intervention.

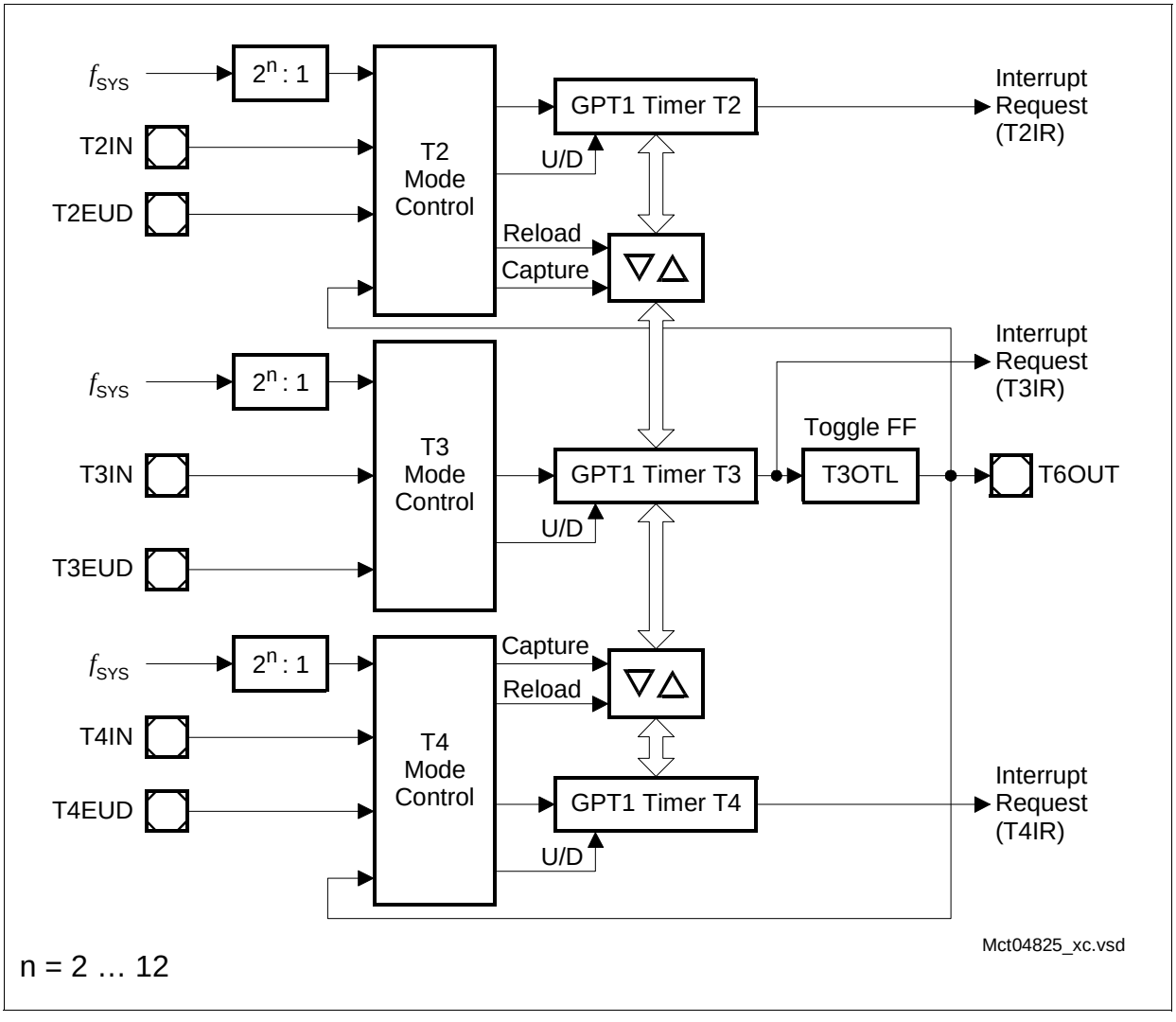



**Figure 7 Block Diagram of GPT1**

With its maximum resolution of 2 system clock cycles, the **GPT2 module** provides precise event control and time measurement. It includes two timers (T5, T6) and a capture/reload register (CAPREL). Both timers can be clocked with an input clock which is derived from the CPU clock via a programmable prescaler or with external signals. The count direction (up/down) for each timer is programmable by software or may additionally be altered dynamically by an external signal on a port pin (TxEUD). Concatenation of the timers is supported via the output toggle latch (T6OTL) of timer T6, which changes its state on each timer overflow/underflow.

The state of this latch may be used to clock timer T5, and/or it may be output on pin T6OUT. The overflows/underflows of timer T6 can additionally be used to clock the CAPCOM1/2 timers, and to cause a reload from the CAPREL register.

The CAPREL register may capture the contents of timer T5 based on an external signal transition on the corresponding port pin (CAPIN), and timer T5 may optionally be cleared

after the capture procedure. This allows the XC167 to measure absolute time differences or to perform pulse multiplication without software overhead.

The capture trigger (timer T5 to CAPREL) may also be generated upon transitions of GPT1 timer T3's inputs T3IN and/or T3EUD. This is especially advantageous when T3 operates in Incremental Interface Mode.

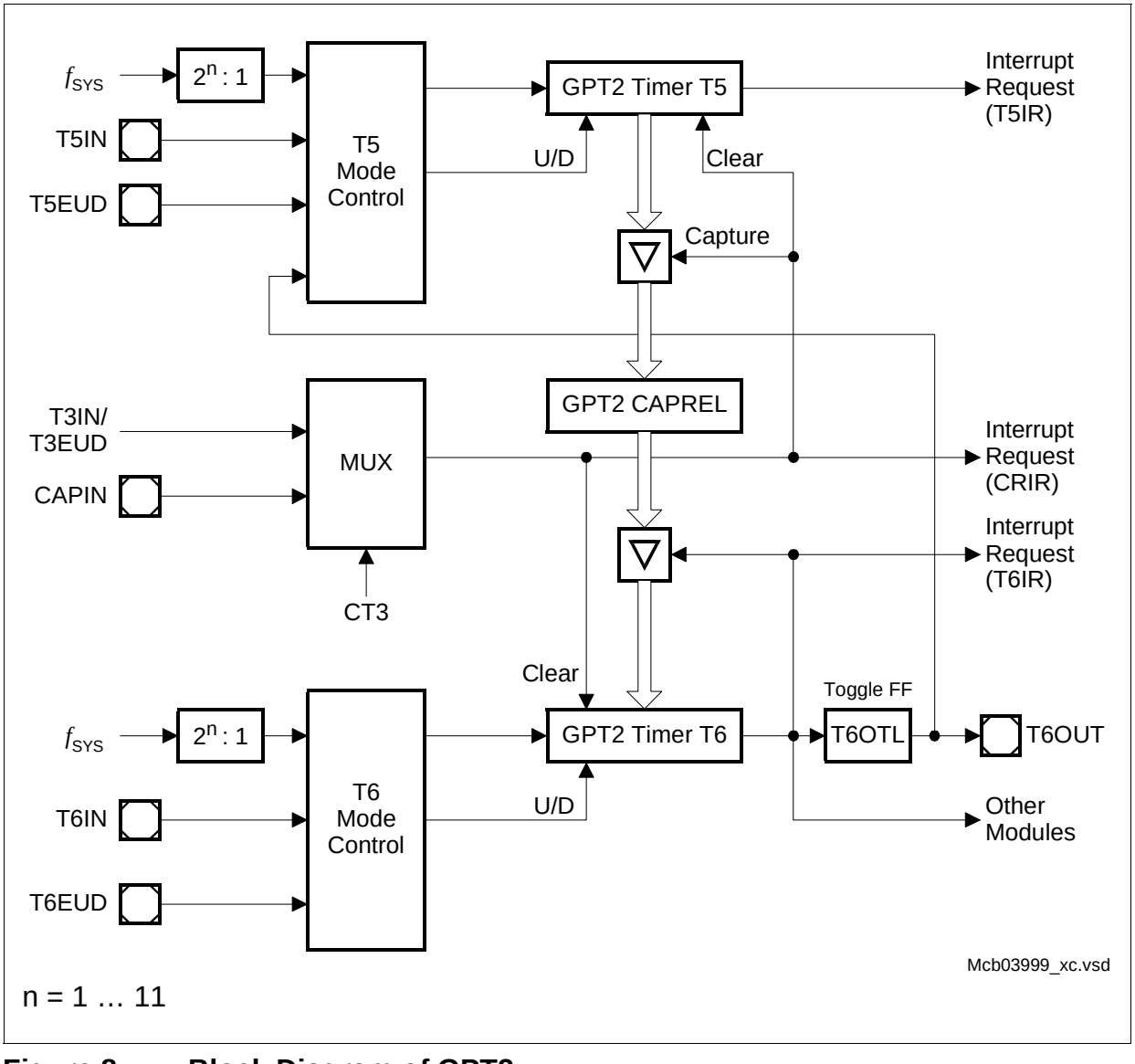



**Figure 8 Block Diagram of GPT2**

## 3.9 Real Time Clock

The Real Time Clock (RTC) module of the XC167 is directly clocked via a separate clock driver with the on-chip auxiliary oscillator frequency ($f_{RTC} = f_{OSCa}$). It is therefore independent from the selected clock generation mode of the XC167.

The RTC basically consists of a chain of divider blocks:

- a selectable 8:1 divider (on - off)
- the reloadable 16-bit timer T14
- the 32-bit RTC timer block (accessible via registers RTCH and RTCL), made of:
  - a reloadable 10-bit timer
  - a reloadable 6-bit timer
  - a reloadable 6-bit timer
  - a reloadable 10-bit timer

All timers count up. Each timer can generate an interrupt request. All requests are combined to a common node request. Additionally, T14 can generate a separate node request.

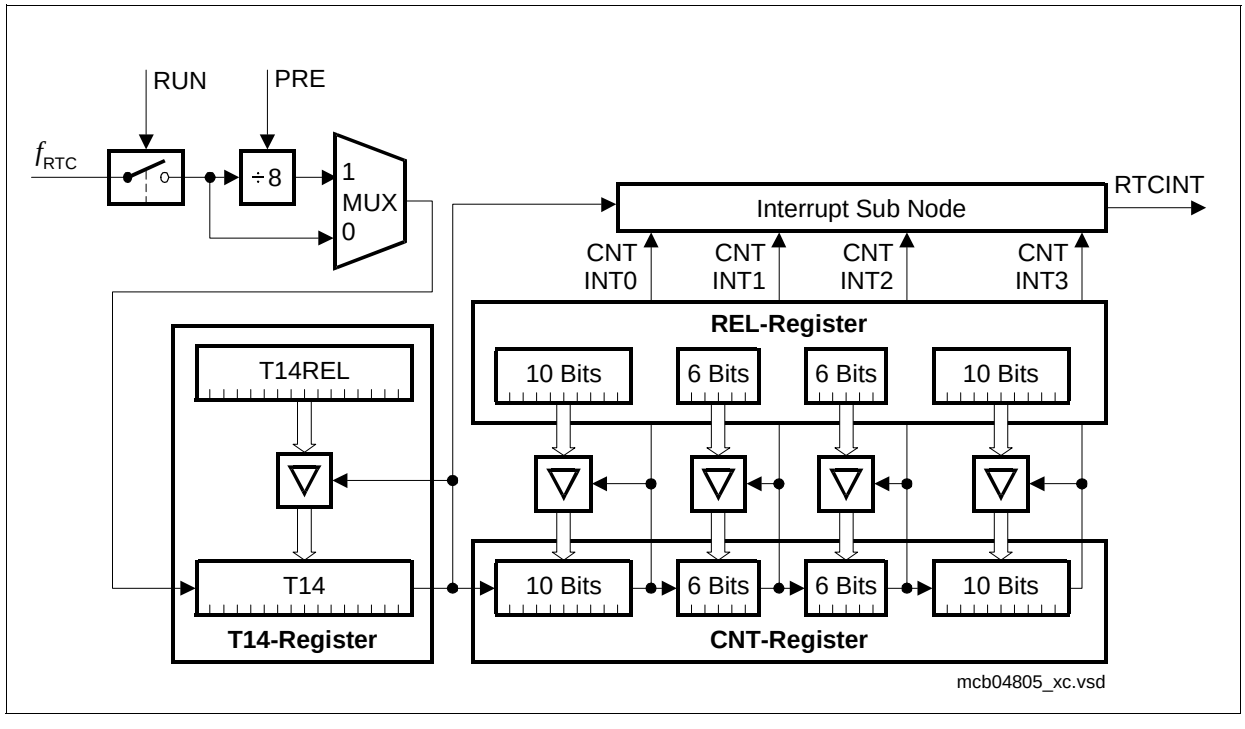



**Figure 9 RTC Block Diagram**

*Note: The registers associated with the RTC are not affected by a reset in order to maintain the correct system time even when intermediate resets are executed.*

The RTC module can be used for different purposes:

- System clock to determine the current time and date,
  optionally during idle mode, sleep mode, and power down mode
- Cyclic time based interrupt, to provide a system time tick independent of CPU frequency and other resources, e.g. to wake up regularly from idle mode.
- 48-bit timer for long term measurements (maximum timespan is >100 years).
- Alarm interrupt for wake-up on a defined time

## 3.10 A/D Converter

For analog signal measurement, a 10-bit A/D converter with 16 multiplexed input channels and a sample and hold circuit has been integrated on-chip. It uses the method of successive approximation. The sample time (for loading the capacitors) and the conversion time is programmable (in two modes) and can thus be adjusted to the external circuitry. The A/D converter can also operate in 8-bit conversion mode, where the conversion time is further reduced.

Overrun error detection/protection is provided for the conversion result register (ADDAT): either an interrupt request will be generated when the result of a previous conversion has not been read from the result register at the time the next conversion is complete, or the next conversion is suspended in such a case until the previous result has been read.

For applications which require less analog input channels, the remaining channel inputs can be used as digital input port pins.

The A/D converter of the XC167 supports four different conversion modes. In the standard Single Channel conversion mode, the analog level on a specified channel is sampled once and converted to a digital result. In the Single Channel Continuous mode, the analog level on a specified channel is repeatedly sampled and converted without software intervention. In the Auto Scan mode, the analog levels on a prespecified number of channels are sequentially sampled and converted. In the Auto Scan Continuous mode, the prespecified channels are repeatedly sampled and converted. In addition, the conversion of a specific channel can be inserted (injected) into a running sequence without disturbing this sequence. This is called Channel Injection Mode.

The Peripheral Event Controller (PEC) may be used to automatically store the conversion results into a table in memory for later evaluation, without requiring the overhead of entering and exiting interrupt routines for each data transfer.

After each reset and also during normal operation the ADC automatically performs calibration cycles. This automatic self-calibration constantly adjusts the converter to changing operating conditions (e.g. temperature) and compensates process variations.

These calibration cycles are part of the conversion cycle, so they do not affect the normal operation of the A/D converter.

In order to decouple analog inputs from digital noise and to avoid input trigger noise those pins used for analog input can be disconnected from the digital IO or input stages under software control. This can be selected for each pin separately via register P5DIDIS (Port 5 Digital Input Disable).

The Auto-Power-Down feature of the A/D converter minimizes the power consumption when no conversion is in progress.

## 3.11 Asynchronous/Synchronous Serial Interfaces (ASC0/ASC1)

The Asynchronous/Synchronous Serial Interfaces ASC0/ASC1 (USARTs) provide serial communication with other microcontrollers, processors, terminals or external peripheral components. They are upward compatible with the serial ports of the Infineon 8-bit microcontroller families and support full-duplex asynchronous communication and half-duplex synchronous communication. A dedicated baud rate generator with a fractional divider precisely generates all standard baud rates without oscillator tuning. For transmission, reception, error handling, and baudrate detection 5 separate interrupt vectors are provided.

In asynchronous mode, 8- or 9-bit data frames (with optional parity bit) are transmitted or received, preceded by a start bit and terminated by one or two stop bits. For multiprocessor communication, a mechanism to distinguish address from data bytes has been included (8-bit data plus wake-up bit mode). IrDA data transmissions up to 115.2 kbit/s with fixed or programmable IrDA pulse width are supported.

In synchronous mode, bytes (8 bits) are transmitted or received synchronously to a shift clock which is generated by the ASC0/1. The LSB is always shifted first.

In both modes, transmission and reception of data is FIFO-buffered. An autobaud detection unit allows to detect asynchronous data frames with its baudrate and mode with automatic initialization of the baudrate generator and the mode control bits.

A number of optional hardware error detection capabilities has been included to increase the reliability of data transfers. A parity bit can automatically be generated on transmission or be checked on reception. Framing error detection allows to recognize data frames with missing stop bits. An overrun error will be generated, if the last character received has not been read out of the receive buffer register at the time the reception of a new character is complete.

### Summary of Features

- Full-duplex asynchronous operating modes
  - 8- or 9-bit data frames, LSB first, one or two stop bits, parity generation/checking
  - Baudrate from 2.5 Mbit/s to 0.6 bit/s (@ 40 MHz)
  - Multiprocessor mode for automatic address/data byte detection
  - Support for IrDA data transmission/reception up to max. 115.2 kbit/s (@ 40 MHz)
  - Loop-back capability
  - Auto baudrate detection
- Half-duplex 8-bit synchronous operating mode at 5 Mbit/s to 406.9 bit/s (@ 40 MHz)
- Buffered transmitter/receiver with FIFO support (8 entries per direction)
- Loop-back option available for testing purposes
- Interrupt generation on transmitter buffer empty condition, last bit transmitted condition, receive buffer full condition, error condition (frame, parity, overrun error), start and end of an autobaud detection

## 3.12 High Speed Synchronous Serial Channels (SSC0/SSC1)

The High Speed Synchronous Serial Channels SSC0/SSC1 support full-duplex and half-duplex synchronous communication. It may be configured so it interfaces with serially linked peripheral components, full SPI functionality is supported.

A dedicated baud rate generator allows to set up all standard baud rates without oscillator tuning. For transmission, reception and error handling three separate interrupt vectors are provided.

The SSC transmits or receives characters of 2 … 16 bits length synchronously to a shift clock which can be generated by the SSC (master mode) or by an external master (slave mode). The SSC can start shifting with the LSB or with the MSB and allows the selection of shifting and latching clock edges as well as the clock polarity.

A number of optional hardware error detection capabilities has been included to increase the reliability of data transfers. Transmit error and receive error supervise the correct handling of the data buffer. Phase error and baudrate error detect incorrect serial data.

### Summary of Features

- Master or Slave mode operation
- Full-duplex or Half-duplex transfers
- Baudrate generation from 20 Mbit/s to 305.18 bit/s (@ 40 MHz)
- Flexible data format
  - Programmable number of data bits: 2 to 16 bits
  - Programmable shift direction: LSB-first or MSB-first
  - Programmable clock polarity: idle low or idle high
  - Programmable clock/data phase: data shift with leading or trailing clock edge
- Loop back option available for testing purposes
- Interrupt generation on transmitter buffer empty condition, receive buffer full condition, error condition (receive, phase, baudrate, transmit error)
- Three pin interface with flexible SSC pin configuration

## 3.13 TwinCAN Module

The integrated TwinCAN module handles the completely autonomous transmission and reception of CAN frames in accordance with the CAN specification V2.0 part B (active), i.e. the on-chip TwinCAN module can receive and transmit standard frames with 11-bit identifiers as well as extended frames with 29-bit identifiers.

Two Full-CAN nodes share the TwinCAN module's resources to optimize the CAN bus traffic handling and to minimize the CPU load. The module provides up to 32 message objects, which can be assigned to one of the CAN nodes and can be combined to FIFO-structures. Each object provides separate masks for acceptance filtering.

The flexible combination of Full-CAN functionality and FIFO architecture reduces the efforts to fulfill the real-time requirements of complex embedded control applications. Improved CAN bus monitoring functionality as well as the number of message objects permit precise and comfortable CAN bus traffic handling.

Gateway functionality allows automatic data exchange between two separate CAN bus systems, which reduces CPU load and improves the real time behavior of the entire system.

The bit timing for both CAN nodes is derived from the master clock and is programmable up to a data rate of 1 Mbit/s. Each CAN node uses two pins of Port 4, Port 7, or Port 9 to interface to an external bus transceiver. The interface pins are assigned via software.

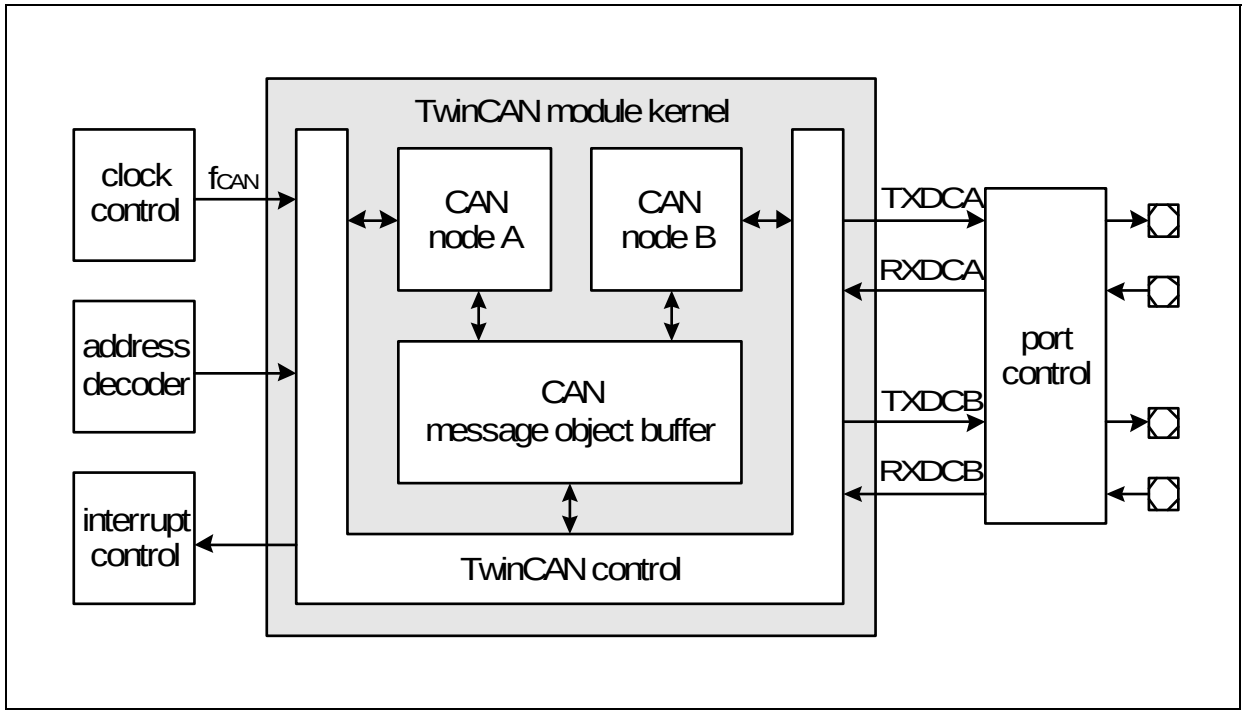



**Figure 10 TwinCAN Module Block Diagram**

**Summary of Features**

- CAN functionality according to CAN specification V2.0 B active.
- Data transfer rate up to 1 Mbit/s
- Flexible and powerful message transfer control and error handling capabilities
- Full-CAN functionality and Basic CAN functionality for each message object
- 32 flexible message objects
  - Assignment to one of the two CAN nodes
  - Configuration as transmit object or receive object
  - Concatenation to a 2-, 4-, 8-, 16-, or 32-message buffer with FIFO algorithm
  - Handling of frames with 11-bit or 29-bit identifiers
  - Individual programmable acceptance mask register for filtering for each object
  - Monitoring via a frame counter
  - Configuration for Remote Monitoring Mode
- Up to eight individually programmable interrupt nodes can be used
- CAN Analyzer Mode for bus monitoring is implemented

*Note: When a CAN node has the interface lines assigned to Port 4, the segment address output on Port 4 must be limited. $\overline{CS}$ lines can be used to increase the total amount of addressable external memory.*

## 3.14 IIC Bus Module

The integrated IIC Bus Module handles the transmission and reception of frames over the two-line IIC bus in accordance with the IIC Bus specification. The IIC Module can operate in slave mode, in master mode or in multi-master mode. It can receive and transmit data using 7-bit or 10-bit addressing. Up to 4 send/receive data bytes can be stored in the extended buffers.

Several physical interfaces (port pins) can be established under software control. Data can be transferred at speeds up to 400 kbit/sec.

Two interrupt nodes dedicated to the IIC module allow efficient interrupt service and also support operation via PEC transfers.

*Note: The port pins associated with the IIC interfaces must be switched to open drain mode, as required by the IIC specification.*

## 3.15 Watchdog Timer

The Watchdog Timer represents one of the fail-safe mechanisms which have been implemented to prevent the controller from malfunctioning for longer periods of time.

The Watchdog Timer is always enabled after a reset of the chip, and can be disabled until the EINIT instruction has been executed (compatible mode), or it can be disabled and enabled at any time by executing instructions DISWDT and ENWDT (enhanced mode). Thus, the chip's start-up procedure is always monitored. The software has to be designed to restart the Watchdog Timer before it overflows. If, due to hardware or software related failures, the software fails to do so, the Watchdog Timer overflows and generates an internal hardware reset and pulls the $\overline{\text{RSTOUT}}$ pin low in order to allow external hardware components to be reset.

The Watchdog Timer is a 16-bit timer, clocked with the system clock divided by 2/4/128/256. The high byte of the Watchdog Timer register can be set to a prespecified reload value (stored in WDTREL) in order to allow further variation of the monitored time interval. Each time it is serviced by the application software, the high byte of the Watchdog Timer is reloaded and the low byte is cleared. Thus, time intervals between 13 µs and 419 ms can be monitored (@ 40 MHz).
The default Watchdog Timer interval after reset is 3.28 ms (@ 40 MHz).

## 3.16 Clock Generation

The Clock Generation Unit uses a programmable on-chip PLL with multiple prescalers to generate the clock signals for the XC167 with high flexibility. The master clock $f_{MC}$ is the reference clock signal, and is used for TwinCAN and is output to the external system. The CPU clock $f_{CPU}$ and the system clock $f_{SYS}$ are derived from the master clock either directly (1:1) or via a 2:1 prescaler ($f_{SYS} = f_{CPU} = f_{MC} / 2$). See also **Section 5.1**.

The on-chip oscillator can drive an external crystal or accepts an external clock signal. The oscillator clock frequency can be multiplied by the on-chip PLL (by a programmable factor) or can be divided by a programmable prescaler factor.

If the bypass mode is used (direct drive or prescaler) the PLL can deliver an independent clock to monitor the clock signal generated by the on-chip oscillator. This PLL clock is independent from the XTAL1 clock. When the expected oscillator clock transitions are missing the Oscillator Watchdog (OWD) activates the PLL Unlock / OWD interrupt node and supplies the CPU with an emergency clock, the PLL clock signal. Under these circumstances the PLL will oscillate with its basic frequency.

**The oscillator watchdog can be disabled** by switching the PLL off. This reduces power consumption, but also no interrupt request will be generated in case of a missing oscillator clock.

*Note: At the end of an external reset ($\overline{EA}$ = '0') the oscillator watchdog may be disabled via hardware by (externally) pulling the $\overline{RD}$ line low upon a reset, similar to the standard reset configuration.*

## 3.17 Parallel Ports

The XC167 provides up to 103 I/O lines which are organized into nine input/output ports and one input port. All port lines are bit-addressable, and all input/output lines are individually (bit-wise) programmable as inputs or outputs via direction registers. The I/O ports are true bidirectional ports which are switched to high impedance state when configured as inputs. The output drivers of some I/O ports can be configured (pin by pin) for push/pull operation or open-drain operation via control registers. During the internal reset, all port pins are configured as inputs (except for pin $\overline{RSTOUT}$).

The edge characteristics (shape) and driver characteristics (output current) of the port drivers can be selected via registers POCONx.

The input threshold of some ports is selectable (TTL or CMOS like), where the special CMOS like input threshold reduces noise sensitivity due to the input hysteresis. The input threshold may be selected individually for each byte of the respective ports.

All port lines have programmable alternate input or output functions associated with them. All port lines that are not used for these alternate functions may be used as general purpose IO lines.

**Table 7 Summary of the XC167's Parallel Ports**

| Port | Control | Alternate Functions |
|---|---|---|
| **PORT0** | Pad drivers | Address/Data lines or data lines[1)] |
| **PORT1** | Pad drivers | Address lines[2)] |
| | | Capture inputs or compare outputs,<br>Serial interface lines |
| **Port 2** | Pad drivers,<br>Open drain,<br>Input threshold | Capture inputs or compare outputs,<br>Timer control signal,<br>Fast external interrupt inputs |
| **Port 3** | Pad drivers,<br>Open drain,<br>Input threshold | Timer control signals, serial interface lines,<br>Optional bus control signal $\overline{BHE}$/$\overline{WRH}$,<br>System clock output CLKOUT (or FOUT) |
| **Port 4** | Pad drivers,<br>Open drain,<br>Input threshold | Segment address lines[3)] |
| | | CAN interface lines[4)] |
| **Port 5** | --- | Analog input channels to the A/D converter,<br>Timer control signals |
| **Port 6** | Open drain,<br>Input threshold | Capture inputs or compare outputs,<br>Bus arbitration signals $\overline{BREQ}$, $\overline{HLDA}$, $\overline{HOLD}$,<br>Optional chip select signals |
| **Port 7** | Open drain,<br>Input threshold | Capture inputs or compare outputs,<br>CAN interface lines[4)] |
| **Port 9** | Pad drivers,<br>Open drain,<br>Input threshold | Capture inputs or compare outputs |
| | | CAN interface lines[4)],<br>IIC bus interface lines[4)] |
| **Port 20** | Pad drivers,<br>Open drain | Bus control signals $\overline{RD}$, $\overline{WR}$/$\overline{WRL}$, READY, ALE,<br>External access enable pin $\overline{EA}$,<br>Reset indication output $\overline{RSTOUT}$ |

1) For multiplexed bus cycles.

2) For demultiplexed bus cycles.

3) For more than 64 Kbytes of external resources.

4) Can be assigned by software.

## 3.18 Power Management

The XC167 provides several means to control the power it consumes either at a given time or averaged over a certain timespan. Three mechanisms can be used (partly in parallel):

- **Power Saving Modes** switch the XC167 into a special operating mode (control via instructions).
  Idle Mode stops the CPU while the peripherals can continue to operate.
  Sleep Mode and Power Down Mode stop all clock signals and all operation (RTC may optionally continue running). Sleep Mode can be terminated by external interrupt signals.
- **Clock Generation Management** controls the distribution and the frequency of internal and external clock signals. While the clock signals for currently inactive parts of logic are disabled automatically, the user can reduce the XC167's CPU clock frequency which drastically reduces the consumed power.
  External circuitry can be controlled via the programmable frequency output FOUT.
- **Peripheral Management** permits temporary disabling of peripheral modules (control via register SYSCON3). Each peripheral can separately be disabled/enabled.

The on-chip RTC supports intermittend operation of the XC167 by generating cyclic wake-up signals. This offers full performance to quickly react on action requests while the intermittend sleep phases greatly reduce the average power consumption of the system.

## 3.19 Instruction Set Summary

Table 8 lists the instructions of the XC167 in a condensed way.
The various addressing modes that can be used with a specific instruction, the operation of the instructions, parameters for conditional execution of instructions, and the opcodes for each instruction can be found in the **“Instruction Set Manual”**.

This document also provides a detailled description of each instruction.

**Table 8 Instruction Set Summary**

| Mnemonic | Description | Bytes |
|---|---|---|
| ADD(B) | Add word (byte) operands | 2 / 4 |
| ADDC(B) | Add word (byte) operands with Carry | 2 / 4 |
| SUB(B) | Subtract word (byte) operands | 2 / 4 |
| SUBC(B) | Subtract word (byte) operands with Carry | 2 / 4 |
| MUL(U) | (Un)Signed multiply direct GPR by direct GPR (16-16-bit) | 2 |
| DIV(U) | (Un)Signed divide register MDL by direct GPR (16-/16-bit) | 2 |
| DIVL(U) | (Un)Signed long divide reg. MD by direct GPR (32-/16-bit) | 2 |
| CPL(B) | Complement direct word (byte) GPR | 2 |
| NEG(B) | Negate direct word (byte) GPR | 2 |
| AND(B) | Bitwise AND, (word/byte operands) | 2 / 4 |
| (X)OR(B) | Bitwise (exclusive)OR, (word/byte operands) | 2 / 4 |
| BCLR / BSET | Clear/Set direct bit | 2 |
| BMOV(N) | Move (negated) direct bit to direct bit | 4 |
| BAND / BOR / BXOR | AND/OR/XOR direct bit with direct bit | 4 |
| BCMP | Compare direct bit to direct bit | 4 |
| BFLDH / BFLDL | Bitwise modify masked high/low byte of bit-addressable direct word memory with immediate data | 4 |
| CMP(B) | Compare word (byte) operands | 2 / 4 |
| CMPD1/2 | Compare word data to GPR and decrement GPR by 1/2 | 2 / 4 |
| CMPI1/2 | Compare word data to GPR and increment GPR by 1/2 | 2 / 4 |
| PRIOR | Determine number of shift cycles to normalize direct word GPR and store result in direct word GPR | 2 |
| SHL / SHR | Shift left/right direct word GPR | 2 |
| ROL / ROR | Rotate left/right direct word GPR | 2 |
| ASHR | Arithmetic (sign bit) shift right direct word GPR | 2 |
| MOV(B) | Move word (byte) data | 2 / 4 |
| MOVBS/Z | Move byte operand to word op. with sign/zero extension | 2 / 4 |

**Table 8 Instruction Set Summary** (cont'd)

| Mnemonic | Description | Bytes |
|---|---|---|
| JMPA/I/R | Jump absolute/indirect/relative if condition is met | 4 |
| JMPS | Jump absolute to a code segment | 4 |
| JB(C) | Jump relative if direct bit is set (and clear bit) | 4 |
| JNB(S) | Jump relative if direct bit is not set (and set bit) | 4 |
| CALLA/I/R | Call absolute/indirect/relative subroutine if condition is met | 4 |
| CALLS | Call absolute subroutine in any code segment | 4 |
| PCALL | Push direct word register onto system stack and call absolute subroutine | 4 |
| TRAP | Call interrupt service routine via immediate trap number | 2 |
| PUSH / POP | Push/pop direct word register onto/from system stack | 2 |
| SCXT | Push direct word register onto system stack and update register with word operand | 4 |
| RET(P) | Return from intra-segment subroutine (and pop direct word register from system stack) | 2 |
| RETS | Return from inter-segment subroutine | 2 |
| RETI | Return from interrupt service subroutine | 2 |
| SBRK | Software Break | 2 |
| SRST | Software Reset | 4 |
| IDLE | Enter Idle Mode | 4 |
| PWRDN | Enter Power Down Mode (supposes $\overline{\text{NMI}}$-pin being low) | 4 |
| SRVWDT | Service Watchdog Timer | 4 |
| DISWDT/ENWDT | Disable/Enable Watchdog Timer | 4 |
| EINIT | Signify End-of-Initialization on RSTOUT-pin | 4 |
| ATOMIC | Begin ATOMIC sequence | 2 |
| EXTR | Begin EXTended Register sequence | 2 |
| EXTP(R) | Begin EXTended Page (and Register) sequence | 2 / 4 |
| EXTS(R) | Begin EXTended Segment (and Register) sequence | 2 / 4 |
| NOP | Null operation | 2 |
| CoMUL / CoMAC | Multiply (and accumulate) | 4 |
| CoADD / CoSUB | Add / Subtract | 4 |
| Co(A)SHR/CoSHL | (Arithmetic) Shift right / Shift left | 4 |
| CoLOAD/STORE | Load accumulator / Store MAC register | 4 |
| CoCMP/MAX/MIN | Compare (maximum/minimum) | 4 |
| CoABS / CoRND | Absolute value / Round accumulator | 4 |
| CoMOV/NEG/NOP | Data move / Negate accumulator / Null operation | 4 |

# 4 Electrical Parameters

## 4.1 Absolute Maximum Ratings

**Table 9 Absolute Maximum Rating Parameters**

| Parameter | Symbol | Limit Values | | Unit | Notes |
|---|---|---|---|---|---|
| | | min. | max. | | |
| Storage temperature | $T_{ST}$ | -65 | 150 | °C | – |
| Junction temperature | $T_J$ | -40 | 150 | °C | under bias |
| Voltage on $V_{DDI}$ pins with respect to ground ($V_{SS}$) | $V_{DDI}$ | -0.5 | 3.25 | V | – |
| Voltage on $V_{DDP}$ pins with respect to ground ($V_{SS}$) | $V_{DDP}$ | -0.5 | 6.2 | V | – |
| Voltage on any pin with respect to ground ($V_{SS}$) | $V_{IN}$ | -0.5 | $V_{DDP}$ + 0.5 | V | – |
| Input current on any pin during overload condition | – | -10 | 10 | mA | – |
| Absolute sum of all input currents during overload condition | – | – | \|100\| | mA | – |

*Note: Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.*
*During absolute maximum rating overload conditions ($V_{IN} > V_{DDP}$ or $V_{IN} < V_{SS}$) the voltage on $V_{DDP}$ pins with respect to ground ($V_{SS}$) must not exceed the values defined by the absolute maximum ratings.*

## 4.2 Package Properties

**Table 10 Package Parameters** (P-TQFP-144-19)

| Parameter | Symbol | Limit Values | | Unit | Notes |
|---|---|---|---|---|---|
| | | min. | max. | | |
| Power dissipation | $P_{DISS}$ | – | 0.8 | W | – |
| Thermal Resistance | $R_{THA}$ | – | 32 | K/W | Chip-Ambient |

## 4.3 Operating Conditions

The following operating conditions must not be exceeded to ensure correct operation of the XC167. All parameters specified in the following sections refer to these operating conditions, unless otherwise noticed.

**Table 11 Operating Condition Parameters**

| Parameter | Symbol | Limit Values | | Unit | Notes |
|---|---|---|---|---|---|
| | | min. | max. | | |
| Digital supply voltage for the core | $V_{DDI}$ | 2.35 | 2.7 | V | Active mode, $f_{CPUmax}$ = 40 MHz |
| Digital supply voltage for IO pads | $V_{DDP}$ | 4.75 | 5.25 | V | Active mode |
| Supply Voltage Difference | $\Delta V_{DD}$ | -0.5 | – | V | $V_{DDP}$ - $V_{DDI}$ [1] |
| Digital ground voltage | $V_{SS}$ | 0 | | V | Reference voltage |
| Overload current | $I_{OV}$ | -5 | 5 | mA | Per IO pin [2][3] |
| | | -2 | 5 | mA | Per analog input pin [2][3] |
| Overload current coupling factor for analog inputs [4] | $K_{OVA}$ | – | $1.0 \times 10^{-4}$ | – | $I_{OV} > 0$ |
| | | – | $1.5 \times 10^{-3}$ | – | $I_{OV} < 0$ |
| Overload current coupling factor for digital I/O pins [4] | $K_{OVD}$ | – | $5.0 \times 10^{-3}$ | – | $I_{OV} > 0$ |
| | | – | $1.0 \times 10^{-2}$ | – | $I_{OV} < 0$ |
| Absolute sum of overload currents | $\Sigma\lvert I_{OV}\rvert$ | – | 50 | mA | [3] |
| External Load Capacitance | $C_L$ | – | 50 | pF | Pin drivers in **default** mode [5] |
| Ambient temperature | $T_A$ | 0 | 70 | °C | SAB-XC167… |
| | | -40 | 85 | °C | SAF-XC167… |
| | | -40 | 125 | °C | SAK-XC167… |

1) This limitation must be fulfilled under all operating conditions including power-ramp-up, power-ramp-down, and power-save modes.

2) Overload conditions occur if the standard operating conditions are exceeded, i.e. the voltage on any pin exceeds the specified range: $V_{OV} > V_{DDP} + 0.5$ V ($I_{OV} > 0$) or $V_{OV} < V_{SS} - 0.5$ V ($I_{OV} < 0$). The absolute sum of input overload currents on all pins may not exceed **50 mA**. The supply voltages must remain within the specified limits.
Proper operation is not guaranteed if overload conditions occur on functional pins such as XTAL1, $\overline{RD}$, $\overline{WR}$, etc.

3) Not 100% tested, guaranteed by design and characterization.

4) An overload current ($I_{OV}$) through a pin injects a certain error current ($I_{INJ}$) into the adjacent pins. This error current adds to the respective pin's leakage current ($I_{OZ}$). The amount of error current depends on the overload current and is defined by the overload coupling factor $K_{OV}$. The polarity of the injected error current is inverse compared to the polarity of the overload current that produces it.
The total current through a pin is $|I_{TOT}| = |I_{OZ}| + (|I_{OV}| \times K_{OV})$. The additional error current may distort the input voltage on analog inputs.

5) The timing is valid for pin drivers operating in default current mode (selected after reset). Reducing the output current may lead to increased delays or reduced driving capability ($C_L$).

## 4.4 Parameter Interpretation

The parameters listed in the following partly represent the characteristics of the XC167 and partly its demands on the system. To aid in interpreting the parameters right, when evaluating them for a design, they are marked in column “Symbol”:

**CC** (**C**ontroller **C**haracteristics):
The logic of the XC167 will provide signals with the respective characteristics.

**SR** (**S**ystem **R**equirement):
The external system must provide signals with the respective characteristics to the XC167.

## 4.5 DC Parameters

**DC Characteristics**
(Operating Conditions apply)[1)]

| Parameter | Symbol | | Limit Values | | Unit | Test Condition |
|---|---|---|---|---|---|---|
| | | | min. | max. | | |
| Input low voltage TTL (all except XTAL1, XTAL3) | $V_{IL}$ | SR | -0.5 | $0.2 \times V_{DDP}$ - 0.1 | V | – |
| Input low voltage for XTAL1, XTAL3 | $V_{ILC}$ | SR | -0.5 | $0.3 \times V_{DDI}$ | V | – |
| Input low voltage (Special Threshold) | $V_{ILS}$ | SR | -0.5 | 0.45 $\times V_{DDP}$ | V | – |
| Input high voltage TTL (all except XTAL1, XTAL3) | $V_{IH}$ | SR | $0.2 \times V_{DDP}$ + 0.9 | $V_{DDP}$ + 0.5 | V | – |
| Input high voltage XTAL1, XTAL3 | $V_{IHC}$ | SR | 0.7 $\times V_{DDI}$ | $V_{DDI}$ + 0.5 | V | – |
| Input high voltage (Special Threshold) | $V_{IHS}$ | SR | $0.8 \times V_{DDP}$ - 0.2 | $V_{DDP}$ + 0.5 | V | – |
| Input Hysteresis (Special Threshold) | HYS | | 0.04 $\times V_{DDP}$ | – | V | $V_{DDP}$ in [V], Series resistance = 0 Ω |
| Output low voltage | $V_{OL}$ | CC | – | 1.0 | V | $I_{OL} \le I_{OLmax}$[2)] |
| | | | – | 0.45 | V | $I_{OL} \le I_{OLnom}$[2) 3)] |
| Output high voltage[4)] | $V_{OH}$ | CC | $V_{DDP}$ - 1.0 | – | V | $I_{OH} \ge I_{OHmax}$[2)] |
| | | | $V_{DDP}$ - 0.45 | – | V | $I_{OH} \ge I_{OHnom}$[2) 3)] |
| Input leakage current (Port 5)[5)] | $I_{OZ1}$ | CC | – | ±300 | nA | 0 V < $V_{IN}$ < $V_{DDP}$ |
| Input leakage current (all other)[5)] | $I_{OZ2}$ | CC | – | ±500 | nA | 0.45 V < $V_{IN}$ < $V_{DDP}$ |
| Configuration pull-up current[6)] | $I_{CPUH}$[7)] | | – | -10 | µA | $V_{IN} = V_{IHmin}$ |
| | $I_{CPUL}$[8)] | | -100 | – | µA | $V_{IN} = V_{ILmax}$ |
| Configuration pull-down current[9)] | $I_{CPDL}$[7)] | | – | 10 | µA | $V_{IN} = V_{ILmax}$ |
| | $I_{CPDH}$[8)] | | 120 | – | µA | $V_{IN} = V_{IHmin}$ |

**DC Characteristics** (cont'd)
(Operating Conditions apply)[1)]

| Parameter | Symbol | Limit Values | | Unit | Test Condition |
|---|---|---|---|---|---|
| | | min. | max. | | |
| Level inactive hold current[10)] | $I_{LHI}$[7)] | – | -10 | µA | $V_{OUT}$ = $0.5 \times V_{DDP}$ |
| Level active hold current[10)] | $I_{LHA}$[8)] | -100 | – | µA | $V_{OUT}$ = 0.45 V |
| XTAL1, XTAL3 input current | $I_{IL}$ CC | – | ±20 | µA | 0 V < $V_{IN}$ < $V_{DDI}$ |
| Pin capacitance[11)] (digital inputs/outputs) | $C_{IO}$ CC | – | 10 | pF | |

1) Keeping signal levels within the limits specified in this table, ensures operation without overload conditions. For signal levels outside these specifications, also refer to the specification of the overload current $I_{OV}$.

2) The maximum deliverable output current of a port driver depends on the selected output driver mode, see **Table 12**, **Current Limits for Port Output Drivers**. The limit for pin groups must be respected.

3) As a rule, with decreasing output current the output levels approach the respective supply level ($V_{OL} \rightarrow V_{SS}$, $V_{OH} \rightarrow V_{DDP}$). However, only the levels for nominal output currents are guaranteed.

4) This specification is not valid for outputs which are switched to open drain mode. In this case the respective output will float and the voltage results from the external circuitry.

5) An additional error current ($I_{INJ}$) will flow if an overload current flows through an adjacent pin. Please refer to the definition of the overload coupling factor $K_{OV}$.

6) This specification is valid during Reset for configuration on $\overline{RD}$, $\overline{WR}$, $\overline{EA}$, PORT0.
The pull-ups on $\overline{RD}$ and $\overline{WR}$ ($\overline{WRL}$/$\overline{WRH}$) are also active during bus hold.

7) The maximum current may be drawn while the respective signal line remains inactive.

8) The minimum current must be drawn to drive the respective signal line active.

9) This specification is valid during Reset for configuration on ALE.
The pull-down on ALE is also active during bus hold.

10) This specification is valid during Reset for pins P6.4-0, which can act as $\overline{CS}$ outputs.
The pull-ups on $\overline{CS}$ outputs are also active during bus hold.

11) Not 100% tested, guaranteed by design and characterization.

**Table 12 Current Limits for Port Output Drivers**

| Port Output Driver Mode | Maximum Output Current ($I_{OLmax}$, $-I_{OHmax}$)[1] | Nominal Output Current ($I_{OLnom}$, $-I_{OHnom}$) |
|---|---|---|
| **Strong driver** | 10 mA | 2.5 mA |
| **Medium driver** | 4.0 mA | 1.0 mA |
| **Weak driver** | 0.5 mA | 0.1 mA |

1) An output current above $|I_{OXnom}|$ may be drawn from up to three pins at the same time.
For any group of 16 neighboring port output pins the total output current in each direction ($\Sigma I_{OL}$ and $\Sigma -I_{OH}$) must remain below 50 mA.

## Power Consumption XC167

(Operating Conditions apply)

| Parameter | Symbol | Limit Values | | Unit | Test Condition |
|---|---|---|---|---|---|
| | | min. | max. | | |
| Power supply current (active) with all peripherals active | $I_{DDI}$ | – | $15 + 2.6 \times f_{CPU}$ | mA | 1) $f_{CPU}$ in [MHz]2) |
| Pad supply current | $I_{DDP}$ | – | 5 | mA | 3) |
| Idle mode supply current with all peripherals active | $I_{IDX}$ | – | $15 + 1.2 \times f_{CPU}$ | mA | $f_{CPU}$ in [MHz]2) |
| Sleep and Power-down mode supply current with RTC disabled4) | $I_{PDO}$5) | – | 20 | µA | $V_{DDI}$=$V_{DDImax}$6) $T_J \leq 25$ °C |

1) During Flash programming or erase operations the supply current is increased by max. 5 mA.

2) The supply current is a function of the operating frequency. This dependency is illustrated in **Figure 11**. These parameters are tested at $V_{DDImax}$ and maximum CPU clock frequency with all outputs disconnected and all inputs at $V_{IL}$ or $V_{IH}$.

3) The pad supply voltage pins ($V_{DDP}$) mainly provides the current consumed by the pin output drivers. A small amount of current is consumed even though no outputs are driven, because the drivers' input stages are switched and also the Flash module draws some power from the $V_{DDP}$ supply.

4) If the RTC remains active in Sleep or Power-down mode the supply current is increased by max. 100 µA.

5) This parameter is determined mainly by the transistor leakage currents. This current heavily depends on the junction temperature (see **Figure 12**). The junction temperature $T_J$ is the same as the ambient temperature $T_A$ if no current flows through the port output drivers. Otherwise, the resulting temperature difference must be taken into account.

6) This parameter is tested including leakage currents. All inputs (including pins configured as inputs) at 0 V to 0.1 V or at $V_{DDP}$ - 0.1 V to $V_{DDP}$, all outputs (including pins configured as outputs) disconnected.

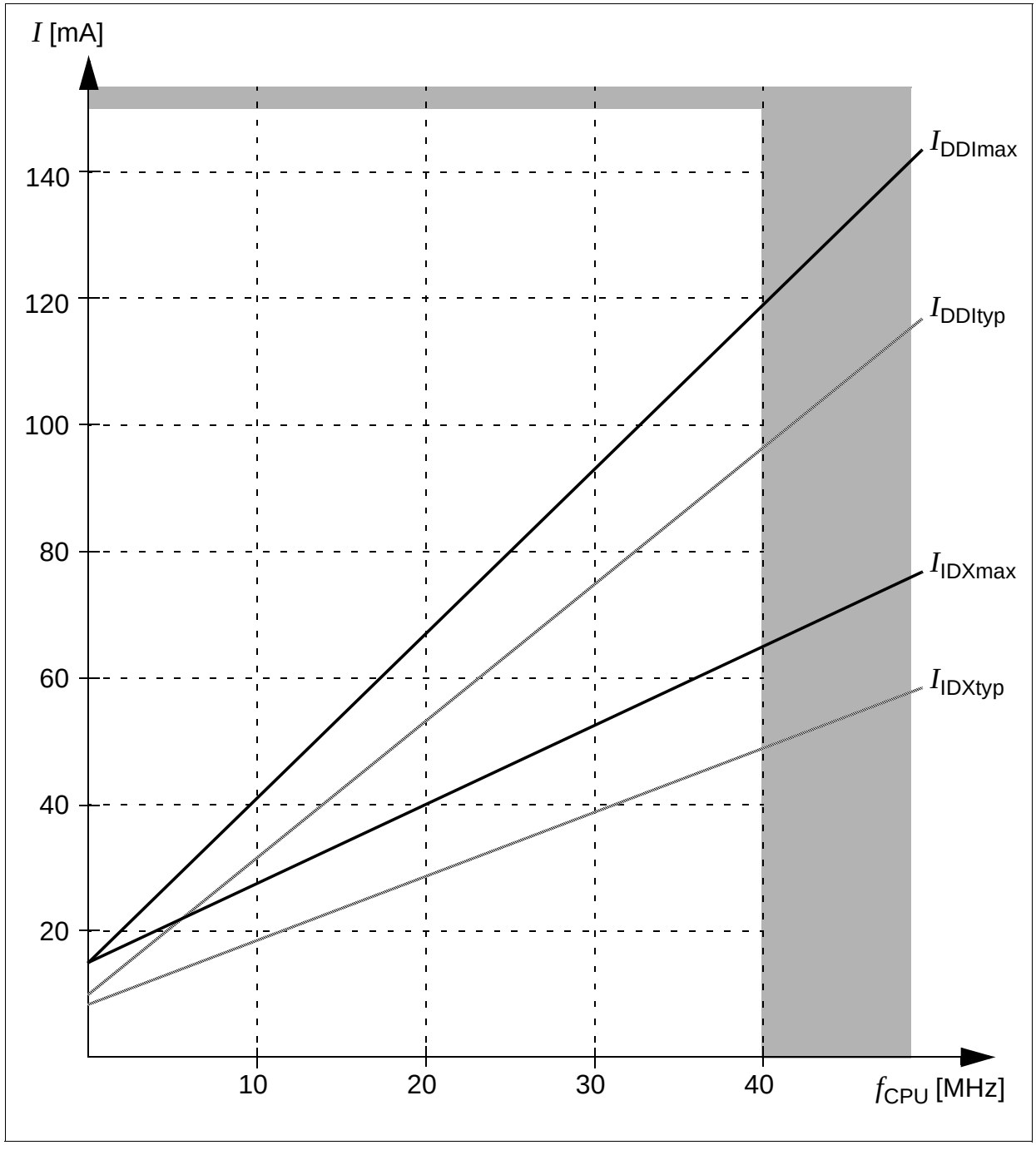



**Figure 11 Supply/Idle Current as a Function of Operating Frequency**

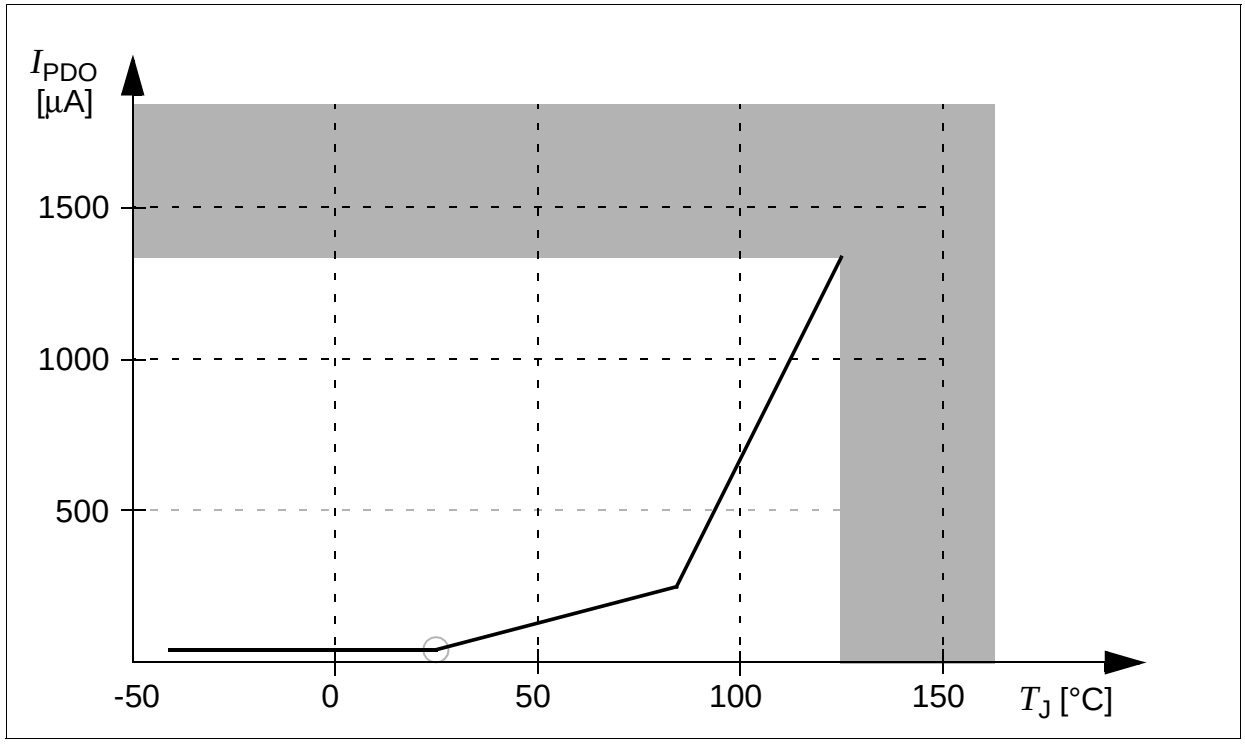



**Figure 12 Sleep and Power Down Supply Current as a Function of Temperature**

## 4.6 A/D Converter Characteristics

**Table 13 A/D Converter Characteristics**
(Operating Conditions apply)

| Parameter | Symbol | | Limit Values | | Unit | Test Condition |
|---|---|---|---|---|---|---|
| | | | min. | max. | | |
| Analog reference supply | $V_{AREF}$ | SR | 4.5 | $V_{DDP}$ + 0.1 | V | 1) |
| Analog reference ground | $V_{AGND}$ | SR | $V_{SS}$ - 0.1 | $V_{SS}$ + 0.1 | V | |
| Analog input voltage range | $V_{AIN}$ | SR | $V_{AGND}$ | $V_{AREF}$ | V | 2) |
| Basic clock frequency | $f_{BC}$ | | 0.5 | 20 | MHz | 3) |
| Conversion time[4)] | $t_{C10}$ | CC | – | $48\times t_{BC}$ + $t_S$+$6\times t_{SYS}$ | – | 10-bit conv. $t_{SYS} = 1 / f_{SYS}$ |
| | $t_{C8}$ | CC | – | $40\times t_{BC}$ + $t_S$+$6\times t_{SYS}$ | – | 8-bit conv. $t_{SYS} = 1 / f_{SYS}$ |
| Calibration time after reset | $t_{CAL}$ | CC | 484 | 11,696 | $t_{BC}$ | 5) |
| Total unadjusted error | TUE | CC | – | ±2 | LSB | 1) |
| Total capacitance of an analog input | $C_{AINT}$ | CC | – | 15 | pF | 6) |
| Switched capacitance of an analog input | $C_{AINS}$ | CC | – | 10 | pF | 6) |
| Resistance of the analog input path | $R_{AIN}$ | CC | – | 2 | kΩ | 6) |
| Total capacitance of the reference input | $C_{AREFT}$ | CC | – | 20 | pF | 6) |
| Switched capacitance of the reference input | $C_{AREFS}$ | CC | – | 15 | pF | 6) |
| Resistance of the reference input path | $R_{AREF}$ | CC | – | 1 | kΩ | 6) |

1) TUE is tested at $V_{AREF} = V_{DDP} + 0.1$ V, $V_{AGND} = 0$ V. It is guaranteed by design for all other voltages within the defined voltage range.
If the analog reference supply voltage drops below 4.5 V (i.e. $V_{AREF} \geq 4.0$ V) or exceeds the power supply voltage by up to 0.2 V (i.e. $V_{AREF} = V_{DDP} + 0.2$ V) the maximum TUE is increased to ±3 LSB. This range is not 100% tested.
The specified TUE is guaranteed only, if the absolute sum of input overload currents on Port 5 pins (see $I_{OV}$ specification) does not exceed 10 mA, and if $V_{AREF}$ and $V_{AGND}$ remain stable during the respective period of time. During the reset calibration sequence the maximum TUE may be ±4 LSB.

2) $V_{AIN}$ may exceed $V_{AGND}$ or $V_{AREF}$ up to the absolute maximum ratings. However, the conversion result in these cases will be $X000_H$ or $X3FF_H$, respectively.

3) The limit values for $f_{BC}$ must not be exceeded when selecting the peripheral frequency and the ADCTC setting.

4) This parameter includes the sample time $t_S$, the time for determining the digital result and the time to load the result register with the conversion result.
Values for the basic clock $t_{BC}$ depend on programming and can be taken from **Table 14**.

5) The actual duration of the reset calibration depends on the noise on the reference signal. Conversions executed during the reset calibration increase the calibration time. The TUE for those conversions may be increased.

6) Not 100% tested, guaranteed by design and characterization.
The given parameter values cover the complete operating range. Under relaxed operating conditions (temperature, supply voltage) reduced values can be used for calculations. At room temperature and nominal supply voltage the following typical values can be used:
$C_{AINTtyp}$ = 12 pF, $C_{AINStyp}$ = 7 pF, $R_{AINtyp}$ = 1.5 kΩ, $C_{AREFTtyp}$ = 15 pF, $C_{AREFStyp}$ = 13 pF, $R_{AREFtyp}$ = 0.7 kΩ.

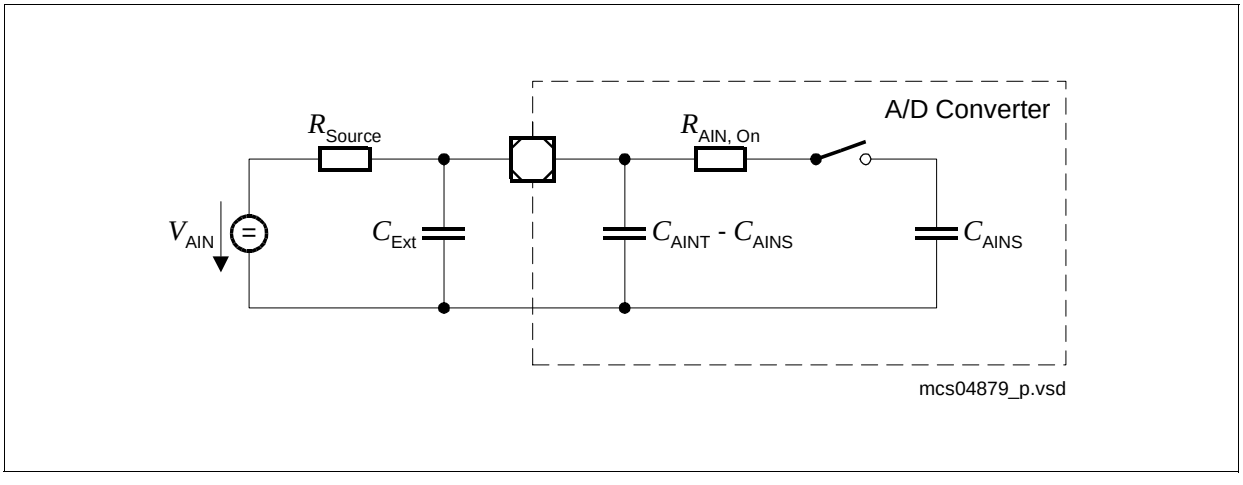



**Figure 13 Equivalent Circuitry for Analog Inputs**

Sample time and conversion time of the XC167's A/D Converter are programmable. In compatibility mode, the above timing can be calculated using **Table 14**.
The limit values for $f_{BC}$ must not be exceeded when selecting ADCTC.

**Table 14 A/D Converter Computation Table[1)]**

| ADCON.15\|14 (ADCTC) | A/D Converter Basic Clock $f_{BC}$ | ADCON.13\|12 (ADSTC) | Sample time $t_S$ |
|---|---|---|---|
| 00 | $f_{SYS}$ / 4 | 00 | $t_{BC} \times 8$ |
| 01 | $f_{SYS}$ / 2 | 01 | $t_{BC} \times 16$ |
| 10 | $f_{SYS}$ / 16 | 10 | $t_{BC} \times 32$ |
| 11 | $f_{SYS}$ / 8 | 11 | $t_{BC} \times 64$ |

1) These selections are available in compatibility mode. An improved mechanism to control the ADC input clock can be selected.

**Converter Timing Example:**

Assumptions: $f_{SYS}$ = 40 MHz (i.e. $t_{SYS}$ = 25 ns), ADCTC = '01', ADSTC = '00'.
Basic clock $f_{BC}$ = $f_{SYS}$ / 2 = 20 MHz, i.e. $t_{BC}$ = 50 ns.
Sample time $t_S$ = $t_{BC} \times 8$ = 400 ns.
Conversion 8-bit $t_{C8}$ = $t_S + 40 \times t_{BC} + 6 \times t_{SYS}$ = (400 + 2000 + 150) ns = 2.55 µs.
Conversion 10-bit $t_{C10}$ = $t_S + 48 \times t_{BC} + 6 \times t_{SYS}$ = (400 + 2400 + 150) ns = 2.95 µs.

# 5 Timing Parameters

## 5.1 Definition of Internal Timing

The internal operation of the XC167 is controlled by the internal master clock $f_{MC}$.

The master clock signal $f_{MC}$ can be generated from the oscillator clock signal $f_{OSC}$ via different mechanisms. The duration of master clock periods (TCMs) and their variation (and also the derived external timing) depend on the used mechanism to generate $f_{MC}$. This influence must be regarded when calculating the timings for the XC167.

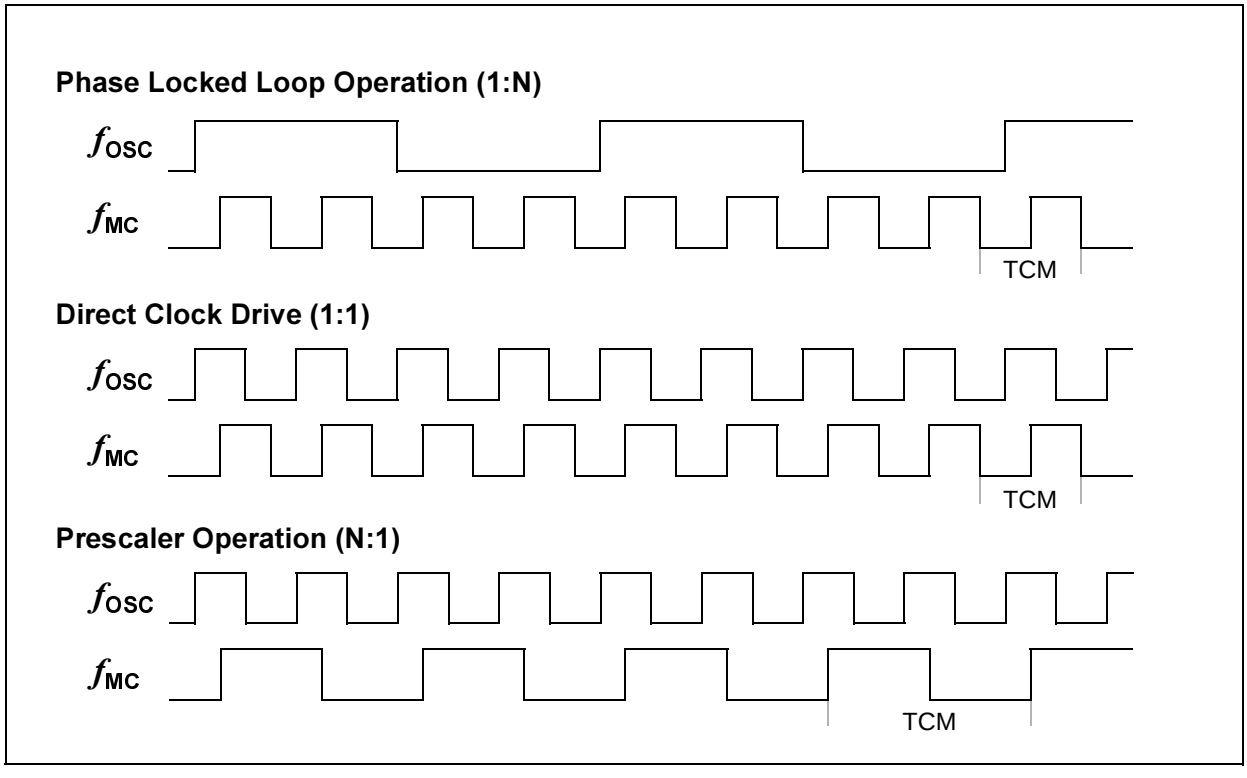



**Figure 14 Generation Mechanisms for the Master Clock**

*Note: The example for PLL operation shown in **Figure 14** refers to a PLL factor of 1:4, the example for prescaler operation refers to a divider factor of 2:1.*

The used mechanism to generate the master clock is selected by register PLLCON.

CPU and EBC are clocked with the CPU clock signal $f_{CPU}$. The CPU clock can have the same frequency as the master clock ($f_{CPU} = f_{MC}$) or can be the master clock divided by two: $f_{CPU} = f_{MC} / 2$. This factor is selected by bit CPSYS in register SYSCON1.

The specification of the external timing (AC Characteristics) depends on the period of the CPU clock, called “TCP”.

The other peripherals are supplied with the system clock signal $f_{SYS}$ which has the same frequency as the CPU clock signal $f_{CPU}$.

### Bypass Operation

When bypass operation is configured (PLLCTRL = $0x_B$) the master clock is derived from the internal oscillator (input clock signal XTAL1) through the input- and output-prescalers:

$f_{MC} = f_{OSC}$ / ((PLLIDIV+1)×(PLLODIV+1)).

If both divider factors are selected as '1' (PLLIDIV = PLLODIV = '0') the frequency of $f_{MC}$ directly follows the frequency of $f_{OSC}$ so the high and low time of $f_{MC}$ is defined by the duty cycle of the input clock $f_{OSC}$.

The lowest master clock frequency is achieved by selecting the maximum values for both divider factors:

$f_{MC} = f_{OSC}$ / ((3+1)×(14+1)) = $f_{OSC}$ / 60.

### Phase Locked Loop (PLL)

When PLL operation is configured (PLLCTRL = $11_B$) the on-chip phase locked loop is enabled and provides the master clock. The PLL multiplies the input frequency by the factor **F** ($f_{MC} = f_{OSC} \times$ **F**) which results from the input divider, the multiplication factor, and the output divider (**F** = PLLMUL+1 / (PLLIDIV+1 × PLLODIV+1)). The PLL circuit synchronizes the master clock to the input clock. This synchronization is done smoothly, i.e. the master clock frequency does not change abruptly.

Due to this adaptation to the input clock the frequency of $f_{MC}$ is constantly adjusted so it is locked to $f_{OSC}$. The slight variation causes a jitter of $f_{MC}$ which also affects the duration of individual TCMs.

The timing listed in the AC Characteristics refers to TCPs. Because $f_{CPU}$ is derived from $f_{MC}$, the timing must be calculated using the minimum TCP possible under the respective circumstances.
The actual minimum value for TCP depends on the jitter of the PLL. As the PLL is constantly adjusting its output frequency so it corresponds to the applied input frequency (crystal or oscillator) the relative deviation for periods of more than one TCP is lower than for one single TCP (see formula and **Figure 15**).

This is especially important for bus cycles using waitstates and e.g. for the operation of timers, serial interfaces, etc. For all slower operations and longer periods (e.g. pulse train generation or measurement, lower baudrates, etc.) the deviation caused by the PLL jitter is negligible.

The value of the accumulated PLL jitter depends on the number of consecutive VCO output cycles within the respective timeframe. The VCO output clock is divided by the output prescaler (K = PLLODIV+1) to generate the master clock signal $f_{MC}$. Therefore, the number of VCO cycles can be represented as $K \times N$, where $N$ is the number of consecutive $f_{MC}$ cycles (TCM).

For a period of $N \times$ TCM the accumulated PLL jitter is defined by the corresponding deviation $D_N$:

$D_N$ [ns] = $\pm(1.5 + 6.32 \times N / f_{MC})$; $f_{MC}$ in [MHz], $N$ = number of consecutive TCMs.

So, for a period of 3 TCMs @ 20 MHz and K = 12: $D_3 = \pm(1.5 + 6.32 \times 3 / 20) = 2.448$ ns.

This formula is applicable for $K \times N < 95$. For longer periods the $K \times N = 95$ value can be used. This steady value can be approximated by: $D_{Nmax}$ [ns] = $\pm(1.5 + 600 / (K \times f_{MC}))$.

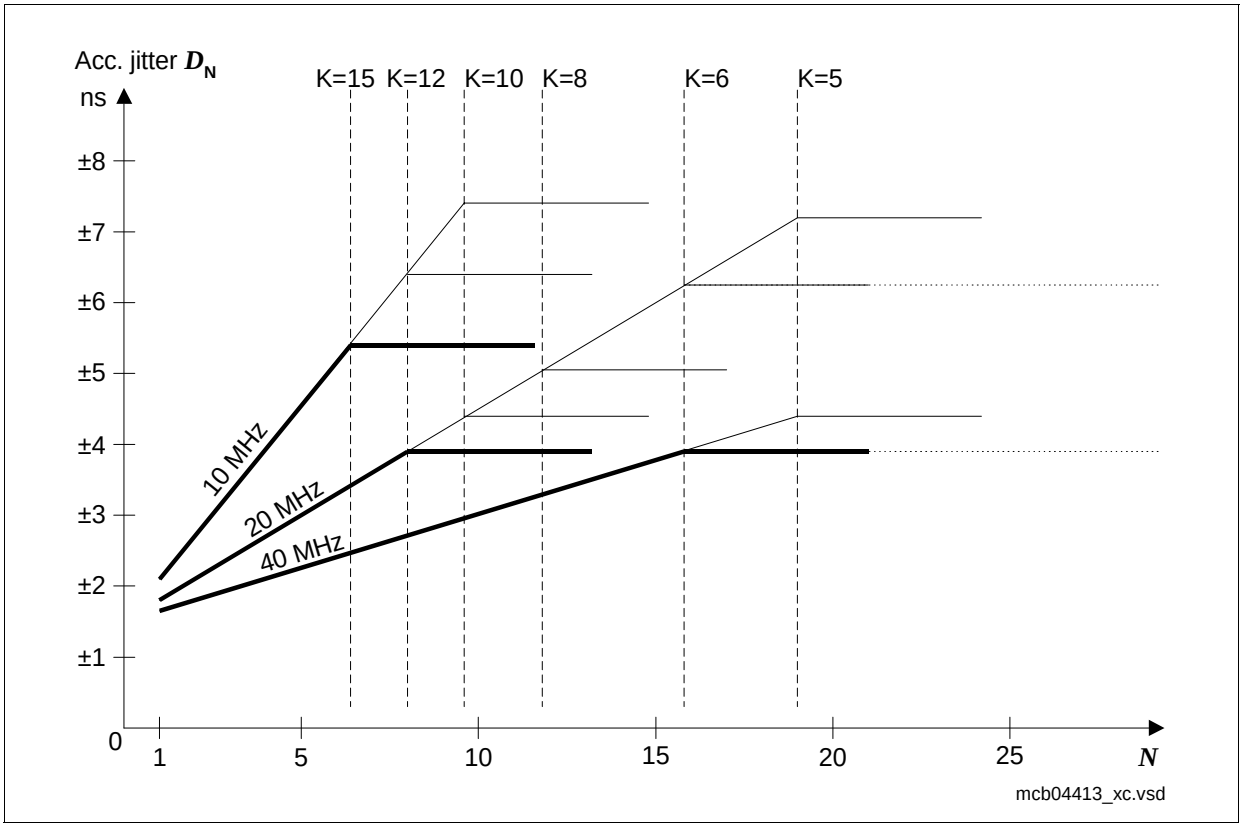



**Figure 15 Approximated Accumulated PLL Jitter**

*Note: The bold lines indicate the minimum accumulated jitter which can be achieved by selecting the maximum possible output prescaler factor K.*

Different frequency bands can be selected for the VCO, so the operation of the PLL can be adjusted to a wide range of input and output frequencies:

**Table 15 VCO Bands for PLL Operation**

| PLLCON.PLLVB | VCO Frequency Range | Base Frequency Range |
|---|---|---|
| 00 | 100 … 150 MHz | 20 … 80 MHz |
| 01 | 150 … 200 MHz | 40 … 130 MHz |
| 10 | 200 … 250 MHz | 60 … 180 MHz |
| 11 | Reserved | |

## 5.2 External Clock Drive XTAL1

**Table 16 External Clock Drive Characteristics**
(Operating Conditions apply)

| Parameter | Symbol | | Limit Values | | Unit |
|---|---|---|---|---|---|
| | | | min. | max. | |
| Oscillator period | $t_{OSC}$ | SR | 20 | 250[1] | ns |
| High time[2] | $t_1$ | SR | 6 | – | ns |
| Low time[2] | $t_2$ | SR | 6 | – | ns |
| Rise time[2] | $t_3$ | SR | – | 8 | ns |
| Fall time[2] | $t_4$ | SR | – | 8 | ns |

1) The maximum limit is only relevant for PLL operation to ensure the minimum input frequency for the PLL.

2) The clock input signal must reach the defined levels $V_{ILC}$ and $V_{IHC}$.

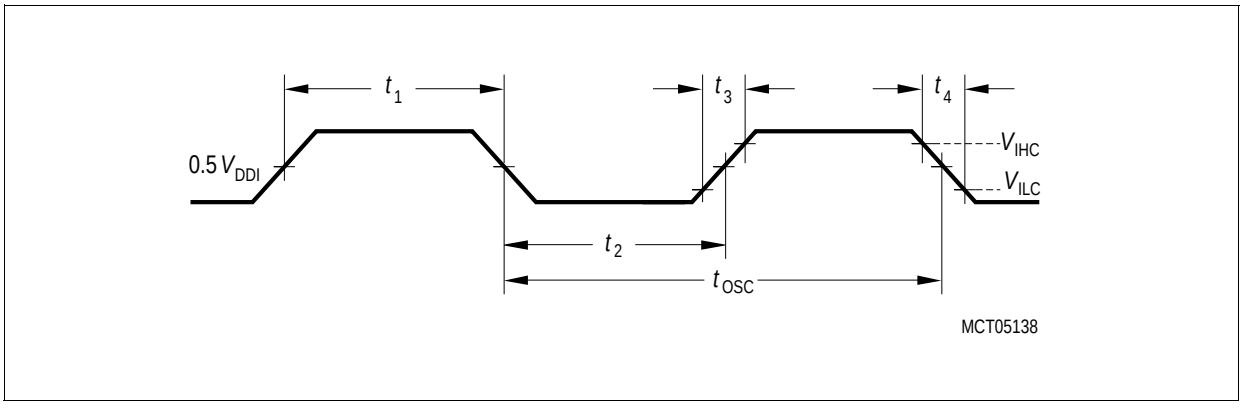



**Figure 16 External Clock Drive XTAL1**

*Note: If the on-chip oscillator is used together with a crystal or a ceramic resonator, the oscillator frequency is limited to a range of 4 MHz to 16 MHz.*
*It is strongly recommended to measure the oscillation allowance (negative resistance) in the final target system (layout) to determine the optimum parameters for the oscillator operation. Please refer to the limits specified by the crystal supplier.*
*When driven by an external clock signal it will accept the specified frequency range. Operation at lower input frequencies is possible but is guaranteed by design only (not 100% tested).*

## 5.3 Testing Waveforms

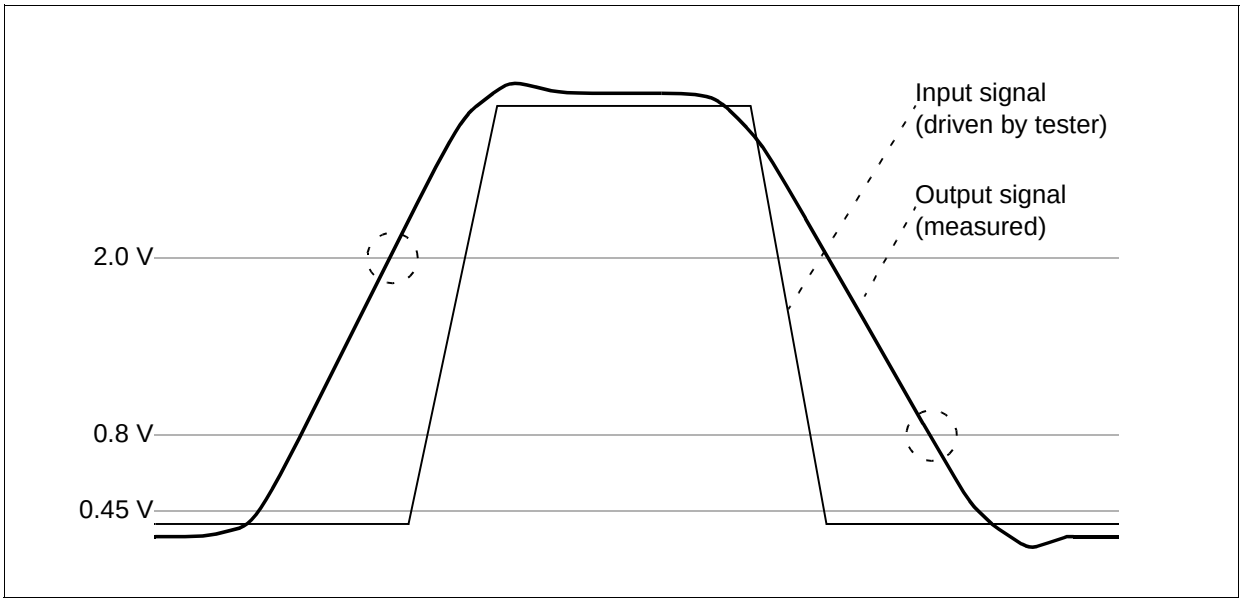
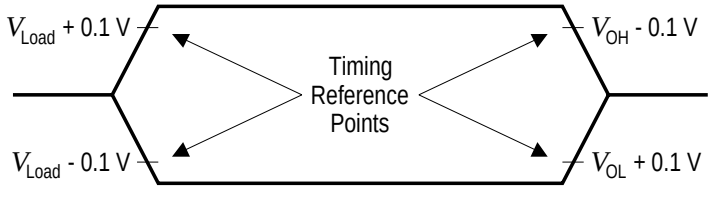



**Figure 17 Input Output Waveforms**




For timing purposes a port pin is no longer floating when a 100 mV change from load voltage occurs, but begins to float when a 100 mV change from the loaded $V_{OH}$ / $V_{OL}$ level occurs ($I_{OH}$ / $I_{OL}$ = 20 mA).

MCA00763

**Figure 18 Float Waveforms**

## 5.4 AC Characteristics

**Table 17 CLKOUT Reference Signal**

| Parameter | Symbol | | Limits | | Unit |
|---|---|---|---|---|---|
| | | | min. | max. | |
| CLKOUT cycle time | $tc_5$ | CC | 40/30/25[1)] | | ns |
| CLKOUT high time | $tc_6$ | CC | 8 | – | ns |
| CLKOUT low time | $tc_7$ | CC | 6 | – | ns |
| CLKOUT rise time | $tc_8$ | CC | – | 4 | ns |
| CLKOUT fall time | $tc_9$ | CC | – | 4 | ns |

1) The CLKOUT cycle time is influenced by the PLL jitter (given values apply to $f_{CPU}$ = 25/33/40 MHz). For longer periods the relative deviation decreases (see PLL deviation formula).

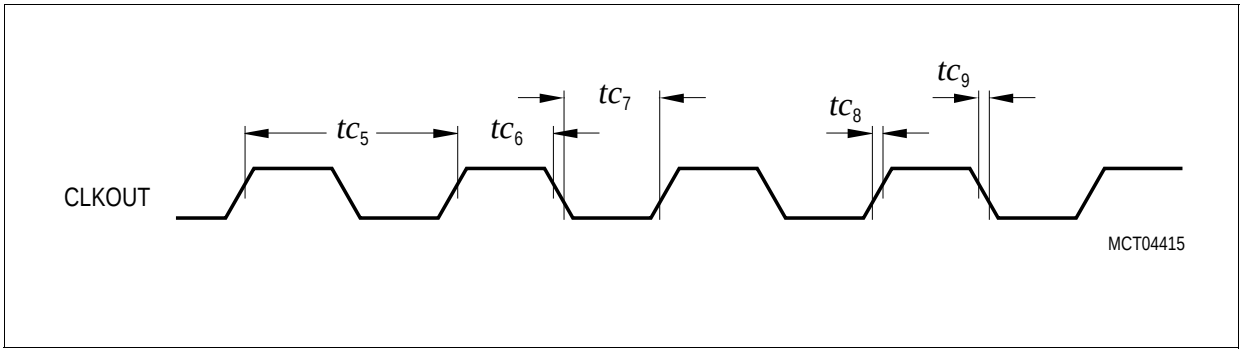



**Figure 19 CLKOUT Signal Timing**

### Variable Memory Cycles

External bus cycles of the XC167 are executed in five subsequent cycle phases (AB, C, D, E, F). The duration of each cycle phase is programmable (via the TCONCSx registers) to adapt the external bus cycles to the respective external module (memory, peripheral, etc.).

The duration of the access phase can optionally be controlled by the external module via the READY handshake input.

This table provides a summary of the phases and the respective choices for their duration.

**Table 18 Programmable Bus Cycle Phases** (see timing diagrams)

| Bus Cycle Phase | Parameter | Valid Values | Unit |
|---|---|---|---|
| Address setup phase, the standard duration of this phase (1 … 2 TCP) can be extended by 0 … 3 TCP if the address window is changed | $tp_{AB}$ | 1 … 2 (5) | TCP |
| Command delay phase | $tp_{C}$ | 0 … 3 | TCP |
| Write Data setup / MUX Tristate phase | $tp_{D}$ | 0 … 1 | TCP |
| Access phase | $tp_{E}$ | 1 … 32 | TCP |
| Address / Write Data hold phase | $tp_{F}$ | 0 … 3 | TCP |

*Note: The bandwidth of a parameter (minimum and maximum value) covers the whole operating range (temperature, voltage) as well as process variations. Within a given device, however, this bandwidth is smaller than the specified range. This is also due to interdependencies between certain parameters. Some of these interdependencies are described in additional notes (see standard timing).*

**Table 19 External Bus Cycle Timing** (Operating Conditions apply)

| Parameter | Symbol | | Limits | | Unit |
|---|---|---|---|---|---|
| | | | min. | max. | |
| Output valid delay for: RD, WR(L/H) | $tc_{10}$ | CC | 1 | 11 | ns |
| Output valid delay for: A23…A16, BHE, ALE | $tc_{11}$ | CC | -1 | 5 | ns |
| Output valid delay for: A15…A0 (on PORT1) | $tc_{12}$ | CC | 1 | 12 | ns |
| Output valid delay for: A15…A0 (on PORT0) | $tc_{13}$ | CC | 3 | 14 | ns |
| Output valid delay for: CS | $tc_{14}$ | CC | 1 | 12 | ns |
| Output valid delay for: D15…D0 (write data, mux-mode) | $tc_{15}$ | CC | -1 | 6 | ns |
| Output valid delay for: D15…D0 (write data, demux-mode) | $tc_{16}$ | CC | 3 | 17 | ns |
| Output hold time for: RD, WR(L/H) | $tc_{20}$ | CC | -3 | 3 | ns |
| Output hold time for: A23…A16, BHE, ALE | $tc_{21}$ | CC | 0 | 8 | ns |
| Output hold time for: A15…A0 (on PORT0) | $tc_{23}$ | CC | 1 | 13 | ns |
| Output hold time for: CS | $tc_{24}$ | CC | -3 | 3 | ns |
| Output hold time for: D15…D0 (write data) | $tc_{25}$ | CC | 1 | 13 | ns |
| Input setup time for: READY, D15…D0 (read data) | $tc_{30}$ | SR | 24 | – | ns |
| Input hold time READY, D15…D0 (read data)[1)] | $tc_{31}$ | SR | -5 | – | ns |

1) Read data are latched with the same (internal) clock edge that triggers the address change and the rising edge of RD. Therefore address changes before the end of RD have no impact on (demultiplexed) read cycles. Read data can be removed after the rising edge of RD.

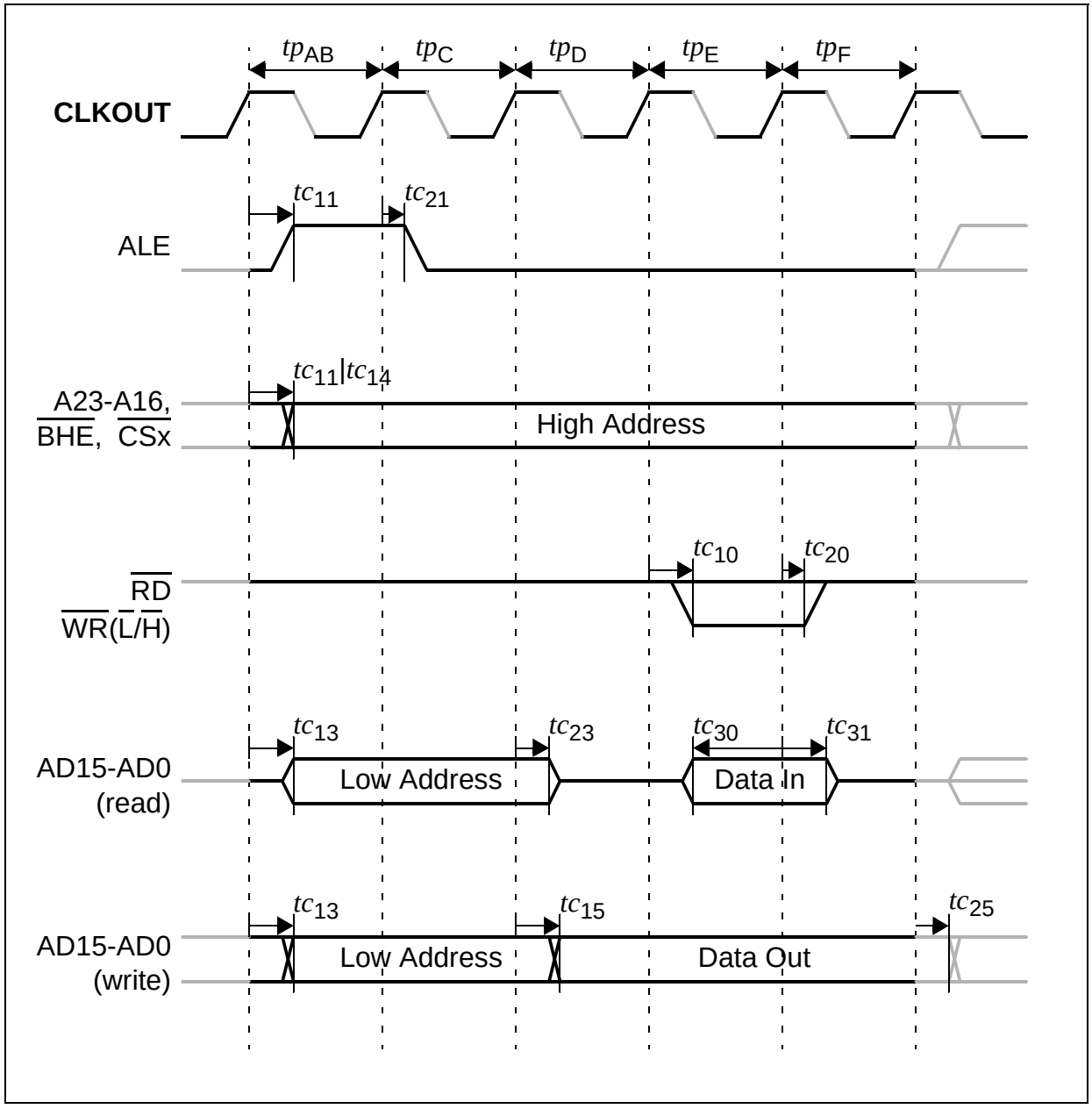



**Figure 20 Multiplexed Bus Cycle**

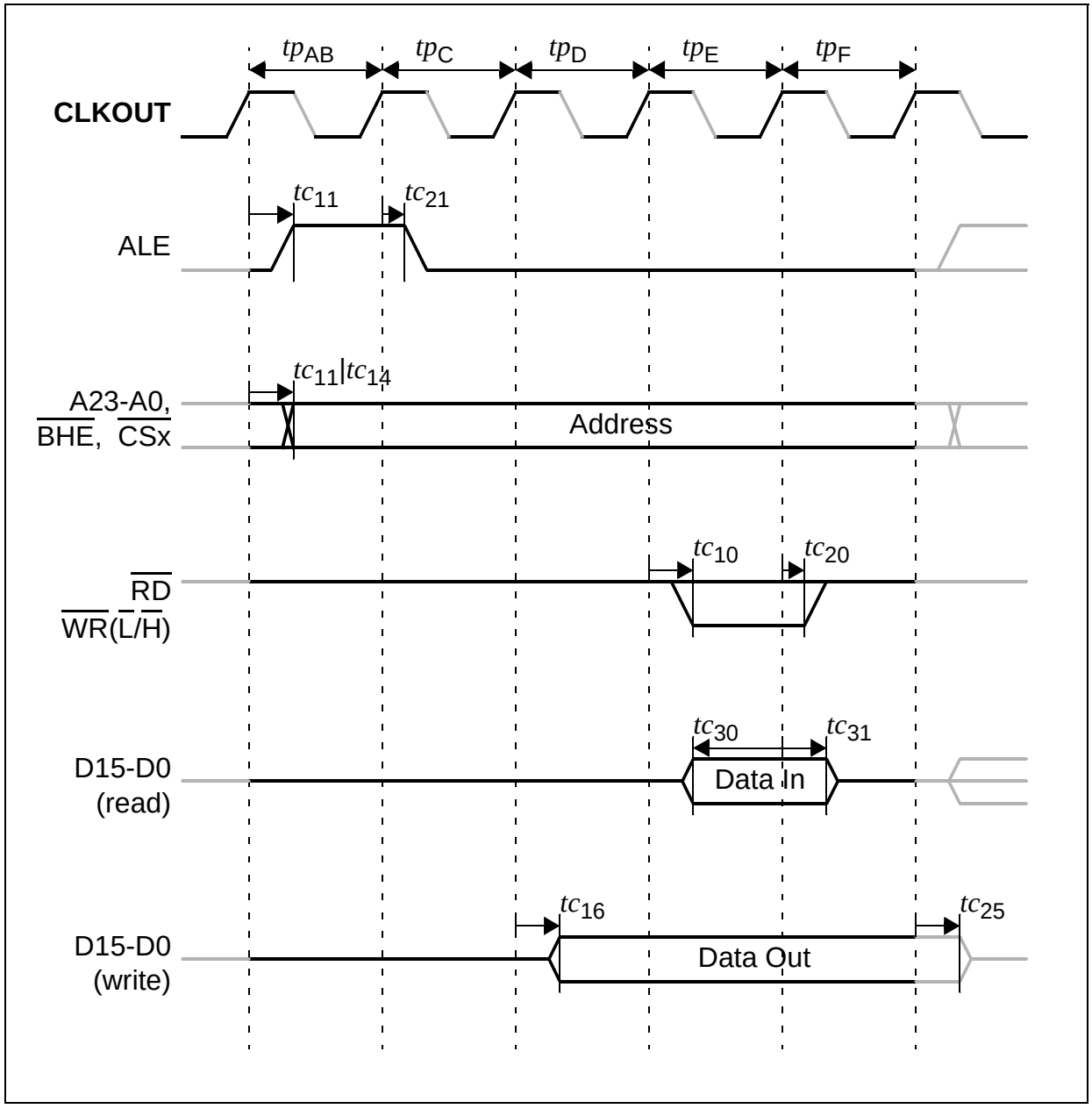



**Figure 21 Demultiplexed Bus Cycle**

### Bus Cycle Control via READY Input

The duration of an external bus cycle can be controlled by the external circuitry via the READY input signal. The polarity of this input signal can be selected.

**Synchronous** READY permits the shortest possible bus cycle but requires the input signal to be synchronous to the reference signal CLKOUT.

**Asynchronous** READY puts no timing constraints on the input signal but incurs one waitstate minimum due to the additional synchronization stage. The minimum duration of an asynchronous READY signal to be safely synchronized must be one CLKOUT period plus the input setup time.

An active READY signal can be deactivated in response to the trailing (rising) edge of the corresponding command ($\overline{RD}$ or $\overline{WR}$).

If the next following bus cycle is READY-controlled, an active READY signal must be disabled before the first valid sample point for the next bus cycle. This sample point depends on the programmed phases of the next following cycle.

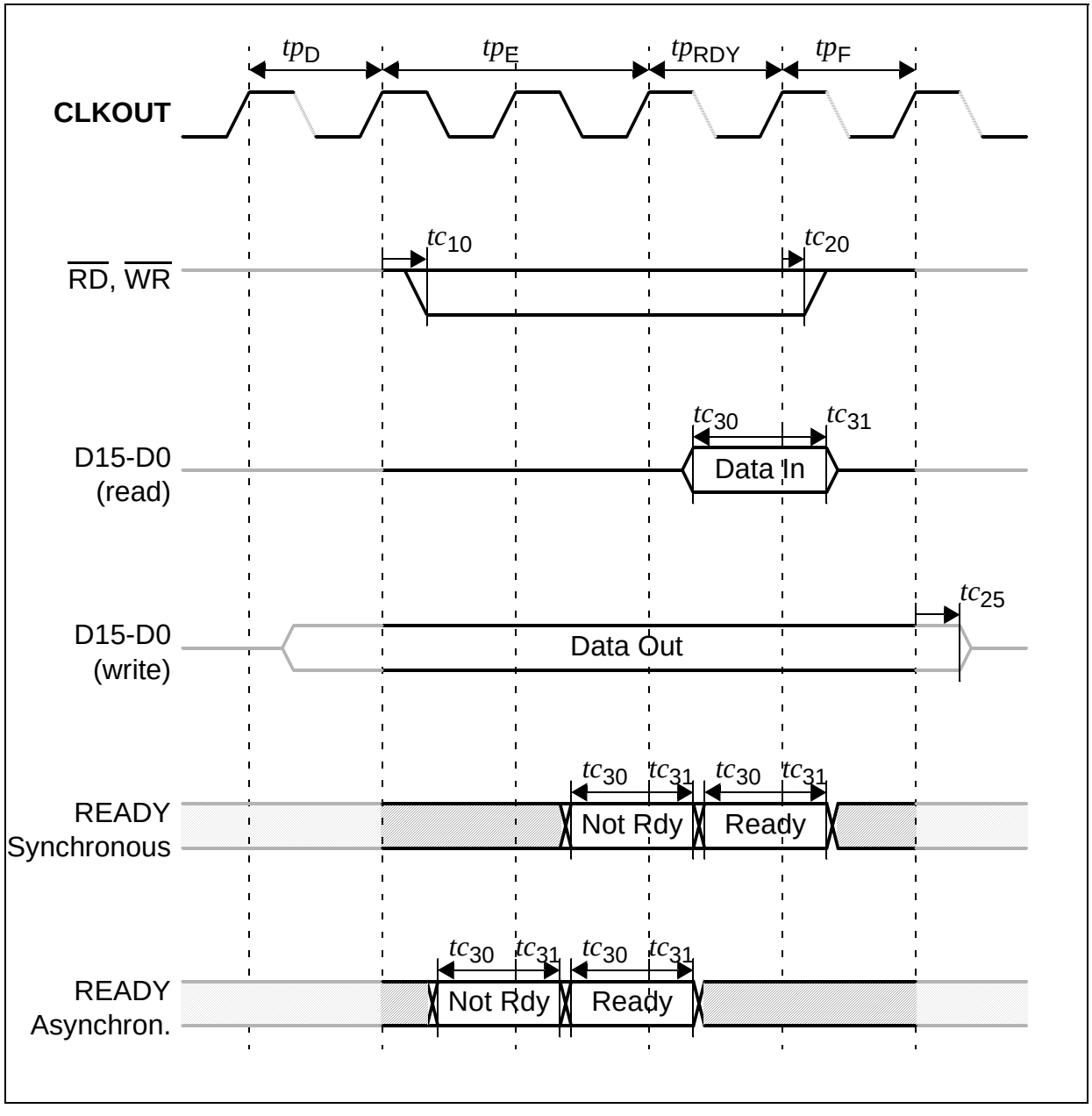



**Figure 22 READY Timing**

*Note: If the READY input is sampled inactive at the indicated sampling point (“Not Rdy”) a READY-controlled waitstate is inserted ($tp_{RDY}$), sampling the READY input active at the indicated sampling point (“Ready”) terminates the currently running bus cycle. Note the different sampling points for synchronous and asynchronous READY. This example uses one mandatory waitstate (see $tp_E$) before the READY input is evaluated.*

## External Bus Arbitration

**Table 20 Bus Arbitration Timing** (Operating Conditions apply)

| Parameter | Symbol | | Limits | | Unit |
|---|---|---|---|---|---|
| | | | min. | max. | |
| Input setup time for:<br>HOLD input | $tc_{40}$ | SR | 15 | – | ns |
| Output delay rising edge for:<br>HLDA, BREQ | $tc_{41}$ | CC | 1 | 6 | ns |
| Output delay falling edge for:<br>HLDA | $tc_{42}$ | CC | 1 | 10 | ns |

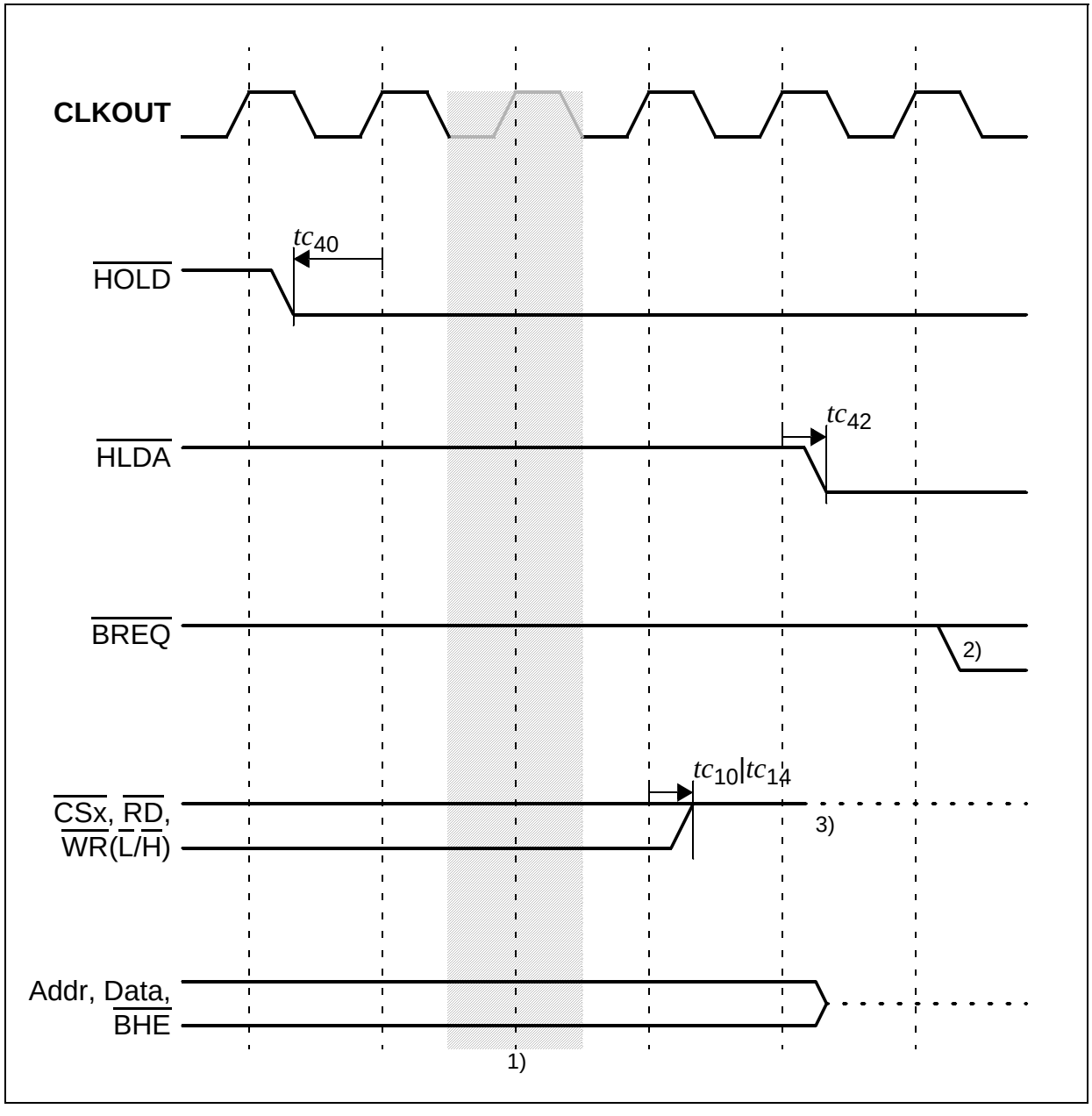



**Figure 23 External Bus Arbitration, Releasing the Bus**

**Notes**
1) The XC167 will complete the currently running bus cycle before granting bus access.
2) This is the first possibility for BREQ to get active.
3) The control outputs will be resistive high (pullup) after being driven inactive (ALE will be low).

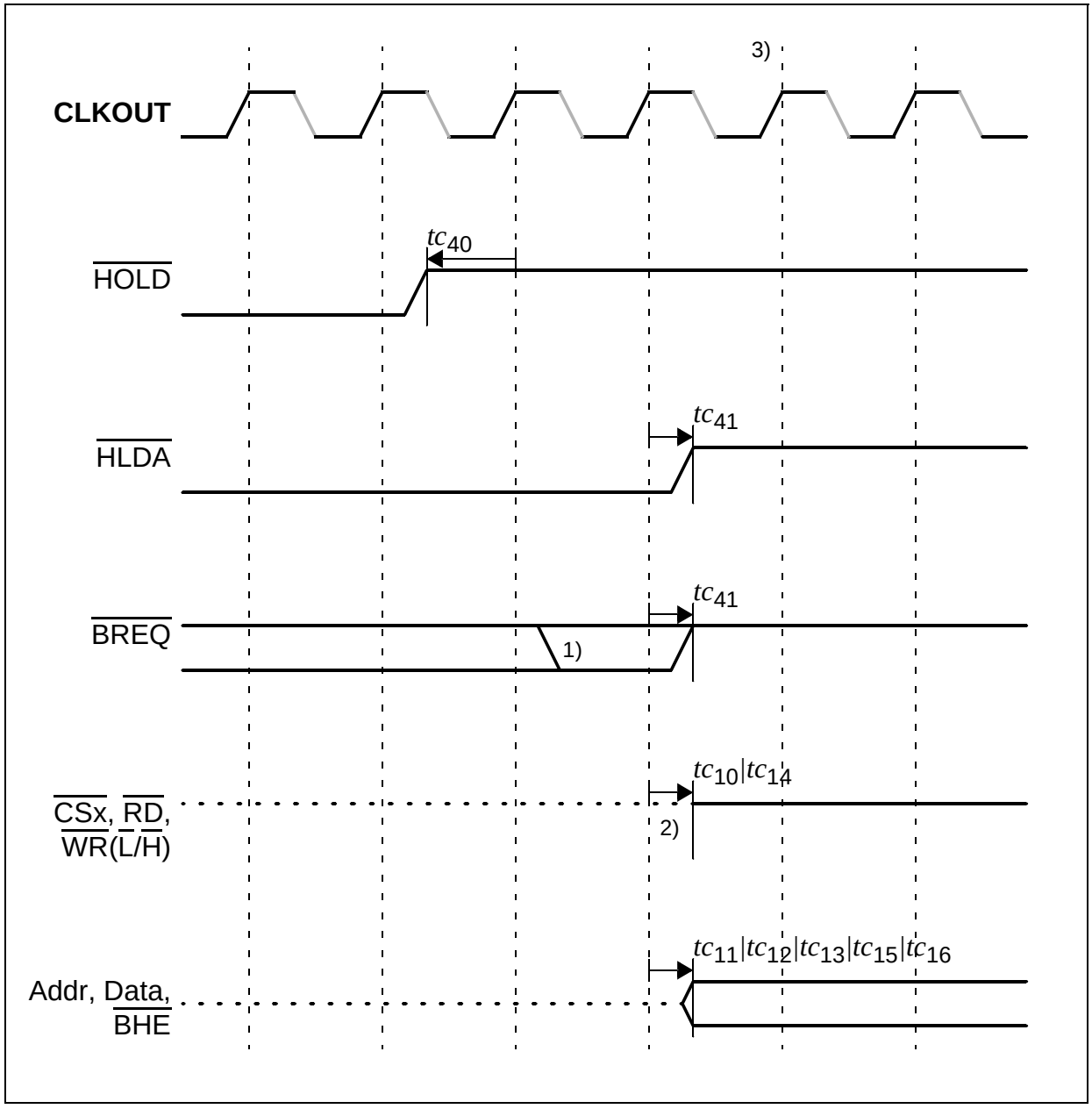



**Figure 24 External Bus Arbitration, (Regaining the Bus)**

**Notes**

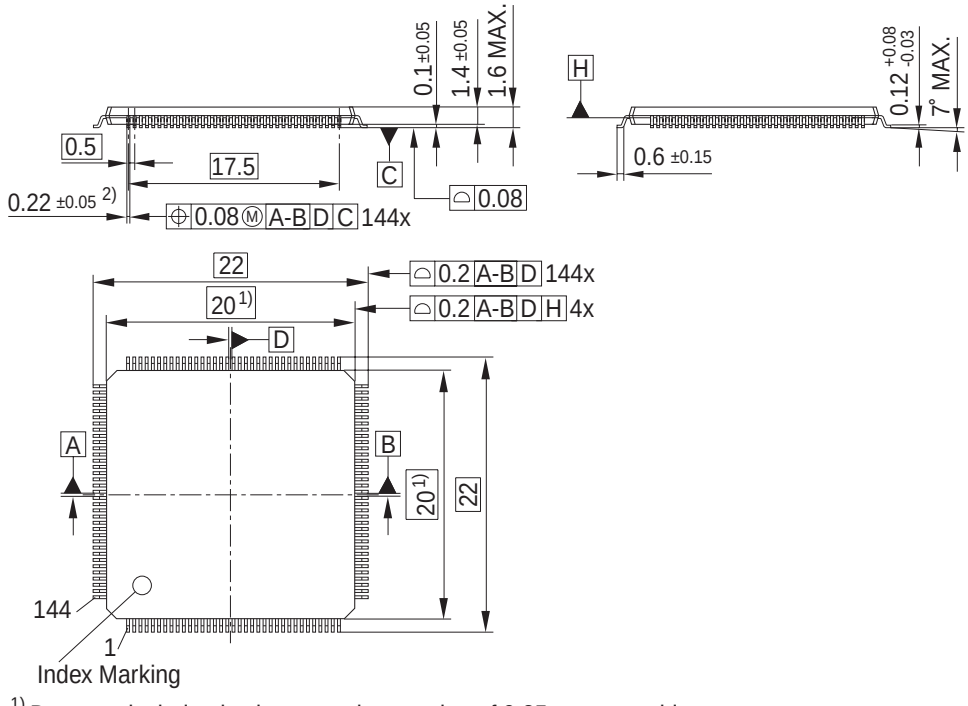
1) This is the last chance for $\overline{\text{BREQ}}$ to trigger the indicated regain-sequence.
Even if $\overline{\text{BREQ}}$ is activated earlier, the regain-sequence is initiated by HOLD going high.
Please note that HOLD may also be deactivated without the XC167 requesting the bus.

2) The control outputs will be resistive high (pullup) before being driven inactive (ALE will be low).

3) The next XC167 driven bus cycle may start here.

# 6 Packaging

**P-TQFP-144-19**
(Plastic Metric Quad Flat Package)




1) Does not include plastic or metal protrusion of 0.25 max. per side
2) Does not include dambar protrusion of 0.08 max. per side

**Figure 25 Package Outlines P-TQFP-144-19**

**Sorts of Packing**

Package outlines for tubes, trays etc. are contained in our Data Book “Package Information”.

SMD = Surface Mounted Device

Dimensions in mn

http://www.infineon.com

Published by Infineon Technologies AG