

# SIEMENS

## Microcomputer Components

16-Bit CMOS Single-Chip Microcontroller

# C161RI

Preliminary 05.98 Preliminary

[http://www.siemens.de/  
semiconductor/](http://www.siemens.de/semiconductor/)

<b>C161RI</b>	
<b>Revision History:                    1998-05-01 Preliminary</b>	
Previous Releases:                    1998-01 Advance Information 1997-12 Advance Information	
<b>Page</b>	<b>Subjects</b>
7	XTAL pin numbers (MQFP) corrected.
34	$V_{DDMIN}$ corrected, special threshold parameters added ( $V_{ILS}$ , $V_{IHS}$ , HYS).
35, 37	Specification of $I_{IDO}$ improved.
41	ADCTC value in converter timing example timing corrected.

### 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@siemens-scg.com](mailto:mcdocu.comments@siemens-scg.com)**



#### Edition 1998-05-01

**Published by Siemens AG,  
Bereich Halbleiter, Marketing-  
Kommunikation, Balanstraße 73,  
81541 München**

© Siemens AG 1998.  
All Rights Reserved.

#### Attention please!

As far as patents or other rights of third parties are concerned, liability is only assumed for components, not for applications, processes and circuits implemented within components or assemblies.

The information describes the type of component and shall not be considered as assured characteristics.

Terms of delivery and rights to change design reserved.

For questions on technology, delivery and prices please contact the Semiconductor Group Offices in Germany or the Siemens Companies and Representatives worldwide (see address list).

Due to technical requirements components may contain dangerous substances. For information on the types in question please contact your nearest Siemens Office, Semiconductor Group.

Siemens AG is an approved CECC manufacturer.

#### Packing

Please use the recycling operators known to you. We can also help you – get in touch with your nearest sales office. By agreement we will take packing material back, if it is sorted. You must bear the costs of transport.

For packing material that is returned to us unsorted or which we are not obliged to accept, we shall have to invoice you for any costs incurred.

#### Components used in life-support devices or systems must be expressly authorized for such purpose!

Critical components<sup>1</sup> of the Semiconductor Group of Siemens AG, may only be used in life-support devices or systems<sup>2</sup> with the express written approval of the Semiconductor Group of Siemens AG.

- 1 A critical component is a component used in a life-support device or system whose failure can reasonably be expected to cause the failure of that life-support device or system, or to affect its safety or effectiveness of that device or system.
- 2 Life support devices or systems are intended (a) to be implanted in the human body, or (b) to support and/or maintain and sustain human life. If they fail, it is reasonable to assume that the health of the user may be endangered.

## C166-Family of High-Performance CMOS 16-Bit Microcontrollers

## C161RI

### Preliminary

### C161RI 16-Bit Microcontroller

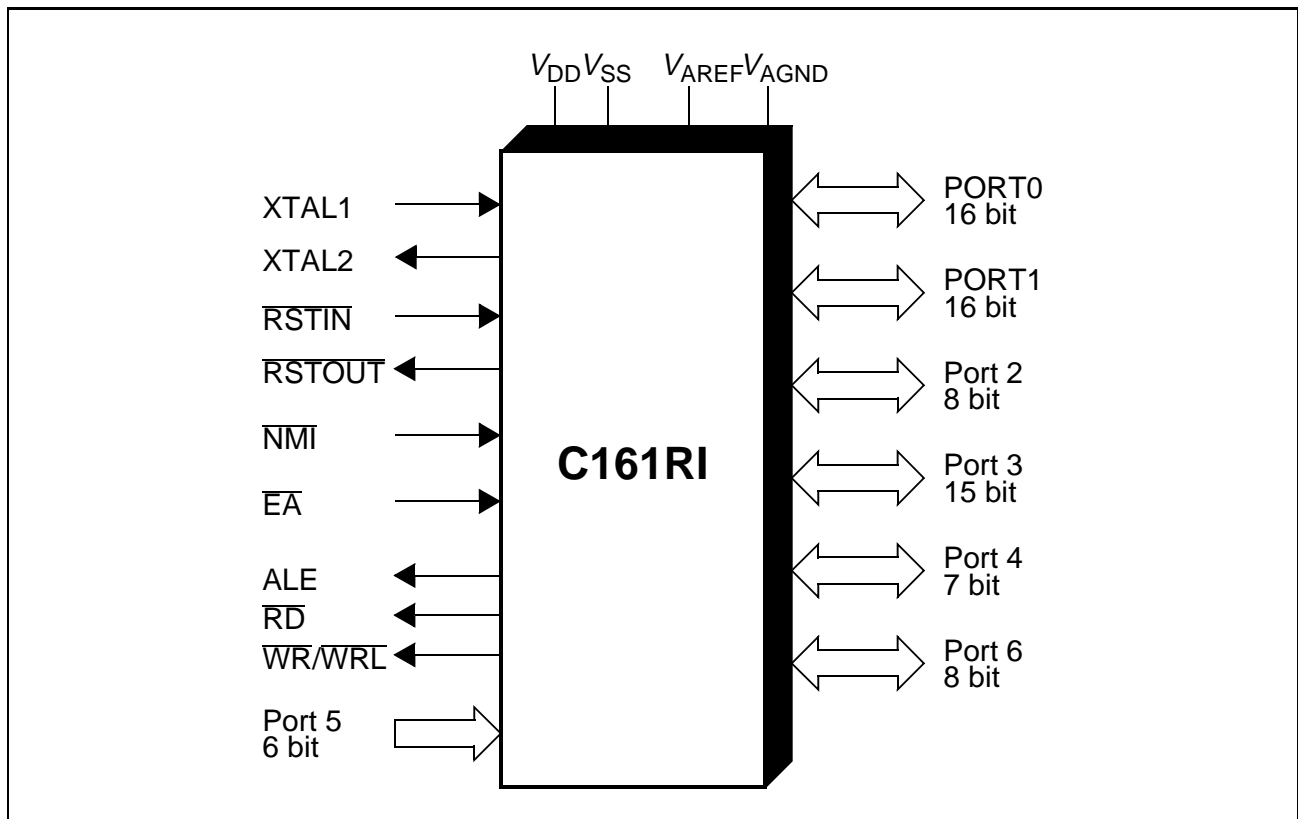
- High Performance 16-bit CPU with 4-Stage Pipeline
- 125 ns Instruction Cycle Time at 16 MHz CPU Clock
- 625 ns Multiplication ( $16 \times 16$  bits), 1.25  $\mu$ s Division (32 / 16 bit)
- Enhanced Boolean Bit Manipulation Facilities
- Additional Instructions to Support HLL and Operating Systems
- Register-Based Design with Multiple Variable Register Banks
- Single-Cycle Context Switching Support
- Clock Generation via Prescaler or via Direct Clock Input
- Up to 8 MBytes Linear Address Space for Code and Data
- 1 KByte On-Chip Internal RAM (IRAM)
- 2 KBytes On-Chip Extension RAM (XRAM)
- Programmable External Bus Characteristics for Different Address Ranges
- 8-Bit or 16-Bit External Data Bus
- Multiplexed or Demultiplexed External Address/Data Bus
- 5 Programmable Chip-Select Signals
- 1024 Bytes On-Chip Special Function Register Area
- 8-Channel Interrupt-Driven Single-Cycle Data Transfer Facilities via Peripheral Event Controller (PEC)
- 16-Priority-Level Interrupt System, 11 External Interrupts
- 4-Channel 8-bit A/D Converter, conversion time down to 7.625  $\mu$ s
- 2 Multi-Functional General Purpose Timer Units with five 16-bit Timers
- Synchronous/Asynchronous Serial Channel (USART)
- High-Speed Synchronous Serial Channel
- I<sup>2</sup>C Bus Interface (10-bit Addressing, 400 KHz) with 2 Channels (multiplexed)
- Up to 76 General Purpose I/O Lines
- Programmable Watchdog Timer
- On-Chip Real Time Clock
- Idle and Power Down Modes with Flexible Power Management
- Ambient temperature range – 40 to 85 °C
- 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 Bootstraploader
- 100-Pin MQFP / TQFP Package

This document describes the **SAB-C161RI-LM**, the **SAB-C161RI-LF**, the **SAF-C161RI-LM** and the **SAB-C161RI-LF**.

For simplicity all versions are referred to by the term **C161RI** throughout this document.

## Introduction

The C161RI is a new derivative of the Siemens C166 Family of 16-bit single-chip CMOS microcontrollers. It combines high CPU performance (up to 8 million instructions per second) with high peripheral functionality and enhanced IO-capabilities. The C161RI derivative is especially suited for cost sensitive applications.



**Figure 1**  
**Logic Symbol**

## Ordering Information

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

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

For the available ordering codes for the C161RI please refer to the “**Product Information Microcontrollers**”, which summarizes all available microcontroller variants.

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

Pin Configuration MQFP Package

(top view)

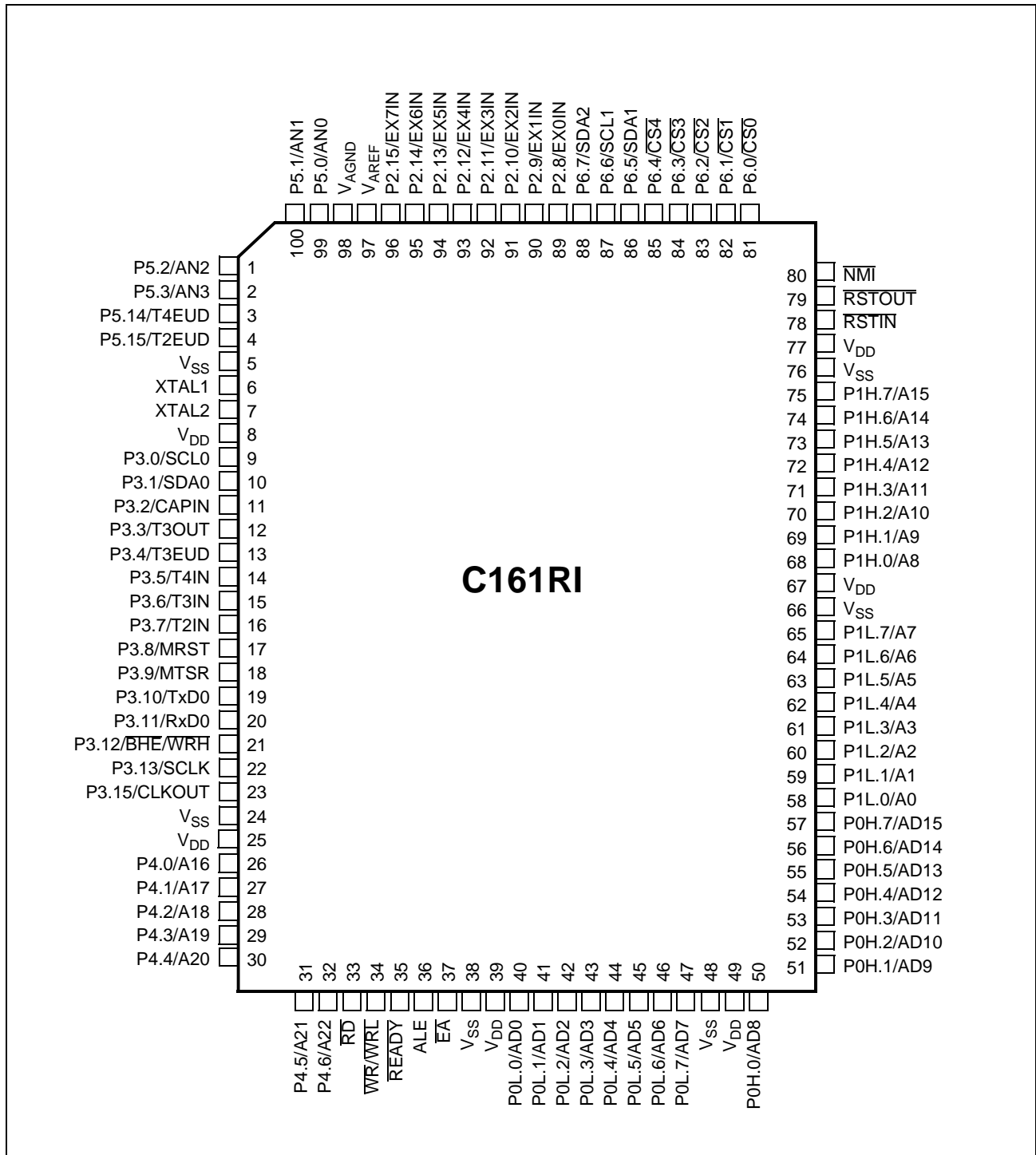


Figure 2

Pin Configuration TQFP Package  
(top view)

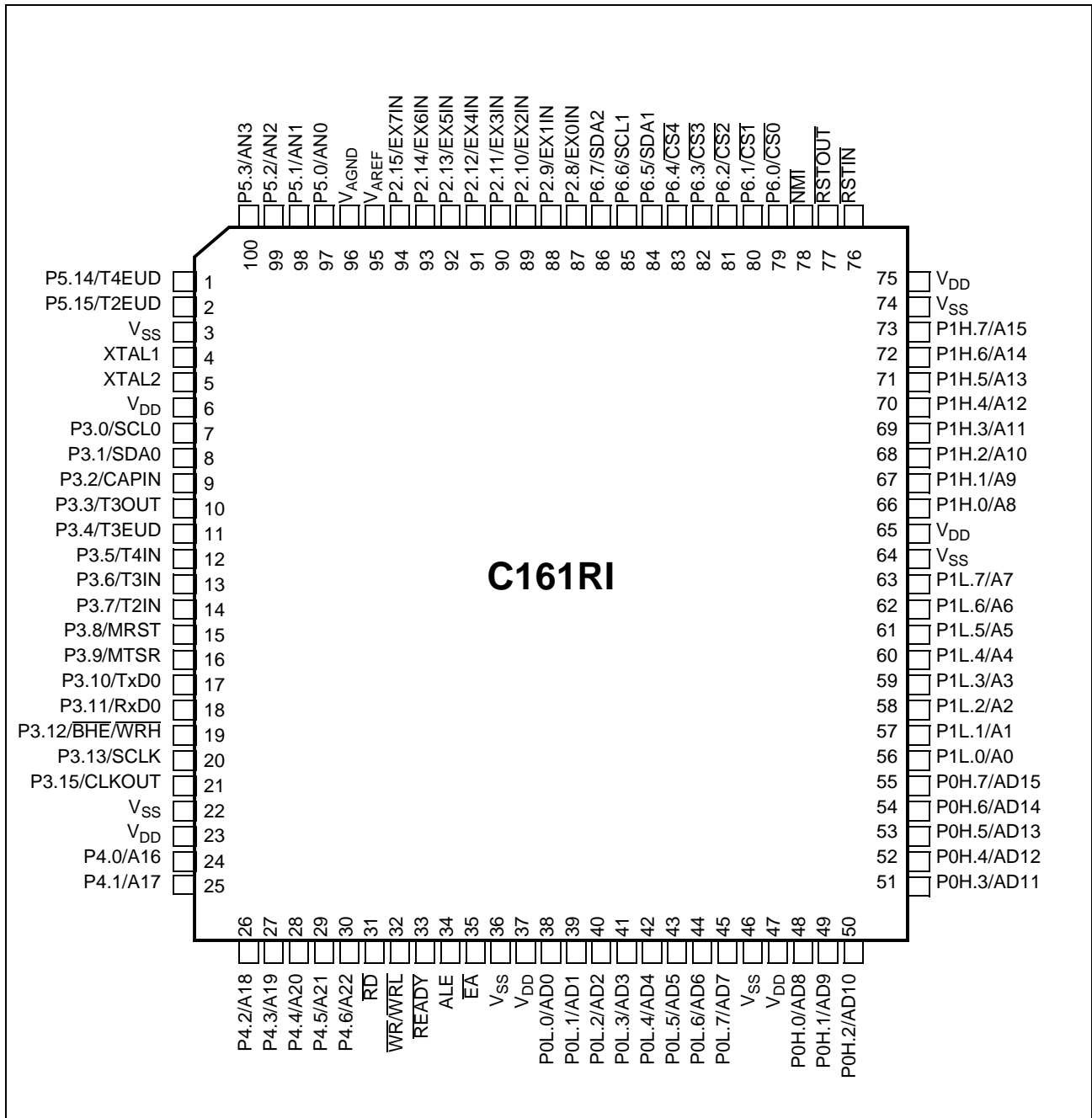


Figure 3

## Pin Definitions and Functions

Symbol	Pin No. TQFP	Pin No. MQFP	Input Outp	Function
P5.0 – P5.3, P5.14 – P5.15	97 – 100, 1 – 2	99 – 2, 3 – 4	I I I I	Port 5 is a 6-bit input-only port with Schmitt-Trigger characteristics. The pins of Port 5 also serve as the (up to 4) analog input channels for the A/D converter, where P5.x equals ANx (Analog input channel x, x=0...3). The following pins of Port 5 also serve as timer inputs: P5.14 T4EUD GPT1 Timer T4 Ext.Up/Down Ctrl.Input P5.15 T2EUD GPT1 Timer T5 Ext.Up/Down Ctrl.Input
XTAL1	4	6	I	XTAL1: Input to the oscillator amplifier and input to the internal clock generator
XTAL2	5	7	O	XTAL2: Output of the oscillator amplifier circuit. 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.
P3.0 – P3.13, P3.15	7 – 20, 21	9 – 22, 23	I/O I/O I/O	Port 3 is a 15-bit bidirectional I/O port. 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. The following Port 3 pins also serve for alternate functions:
	7	9	I/O	P3.0 SCL0 I <sup>2</sup> C Bus Clock Line 0
	8	10	I/O	P3.1 SDA0 I <sup>2</sup> C Bus Data Line 0
	9	11	I	P3.2 CAPIN GPT2 Register CAPREL Capture Input
	10	12	O	P3.3 T3OUT GPT1 Timer T3 Toggle Latch Output
	11	13	I	P3.4 T3EUD GPT1 Timer T3 Ext.Up/Dwn Ctrl.Input
	12	14	I	P3.5 T4IN GPT1 Timer T4 Input for Count/Gate/Reload/Capture
	13	15	I	P3.6 T3IN GPT1 Timer T3 Count/Gate Input
	14	16	I	P3.7 T2IN GPT1 Timer T2 Input for Count/Gate/Reload/Capture
	15	17	I/O	P3.8 MRST SSC Master-Rec./Slave-Transmit I/O
	16	18	I/O	P3.9 MTSR SSC Master-Transmit/Slave-Rec. O/I
	17	19	O	P3.10 TxD0 ASC0 Clock/Data Output (Asyn./Syn.)
	18	20	I/O	P3.11 RxD0 ASC0 Data Input (Asyn.) or I/O (Syn.)
	19	21	O	P3.12 BHE Ext. Memory High Byte Enable Signal,
			O	WRH Ext. Memory High Byte Write Strobe
	20	22	I/O	P3.13 SCLK SSC Master Clock Outp./Slave Cl. Inp.
	21	23	O	P3.15 CLKOUT System Clock Output (=CPU Clock)
				<b>Note:</b> Pins P3.0 and P3.1 are open drain outputs only.

## Pin Definitions and Functions (cont'd)

Symbol	Pin No. TQFP	Pin No. MQFP	Input Outp	Function
P4.0 – P4.6	24 – 30	26 - 32	I/O I/O	Port 4 is a 7-bit bidirectional I/O port. 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. In case of an external bus configuration, Port 4 can be used to output the segment address lines: P4.0      A16      Least Significant Segment Addr. Line ...      ...      ... P4.6      A22      Most Significant Segment Addr. Line
$\overline{RD}$	31	33	O	External Memory Read Strobe. $\overline{RD}$ is activated for every external instruction or data read access.
$\overline{WR}/\overline{WRL}$	32	34	O	External Memory Write Strobe. In $\overline{WR}$ -mode this pin is activated for every external data write access. 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. See WRCFG in register SYSCON for mode selection.
READY	33	35	I	Ready Input. When the Ready function is enabled, a high level at this pin during an external memory access will force the insertion of memory cycle time waitstates until the pin returns to a low level.
ALE	34	36	O	Address Latch Enable Output. Can be used for latching the address into external memory or an address latch in the multiplexed bus modes.
$\overline{EA}$	35	37	I	External Access Enable pin. A low level at this pin during and after Reset forces the C161RI to begin instruction execution out of external memory. A high level forces execution out of the internal ROM. The C161RI must have this pin tied to '0'. <b>Note:</b> This pin is expected to be used to accept the programming voltage for OTP versions of the C161RI.



## Pin Definitions and Functions (cont'd)

Symbol	Pin No. TQFP	Pin No. MQFP	Input Outp	Function																		
PORT0: P0L.0 – P0L.7, P0H.0 - P0H.7	38 – 45, 48 – 55	40 – 47, 50 – 57	I/O	<p>PORT0 consists of the two 8-bit bidirectional I/O ports P0L and P0H. 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.</p> <p>In case of external bus configurations, 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.</p> <p><b>Demultiplexed bus modes:</b></p> <table> <tr> <td>Data Path Width:</td> <td>8-bit</td> <td>16-bit</td> </tr> <tr> <td>P0L.0 – P0L.7:</td> <td>D0 – D7</td> <td>D0 - D7</td> </tr> <tr> <td>P0H.0 – P0H.7:</td> <td>I/O</td> <td>D8 - D15</td> </tr> </table> <p><b>Multiplexed bus modes:</b></p> <table> <tr> <td>Data Path Width:</td> <td>8-bit</td> <td>16-bit</td> </tr> <tr> <td>P0L.0 – P0L.7:</td> <td>AD0 – AD7</td> <td>AD0 - AD7</td> </tr> <tr> <td>P0H.0 – P0H.7:</td> <td>A8 - A15</td> <td>AD8 - AD15</td> </tr> </table>	Data Path Width:	8-bit	16-bit	P0L.0 – P0L.7:	D0 – D7	D0 - D7	P0H.0 – P0H.7:	I/O	D8 - D15	Data Path Width:	8-bit	16-bit	P0L.0 – P0L.7:	AD0 – AD7	AD0 - AD7	P0H.0 – P0H.7:	A8 - A15	AD8 - AD15
Data Path Width:	8-bit	16-bit																				
P0L.0 – P0L.7:	D0 – D7	D0 - D7																				
P0H.0 – P0H.7:	I/O	D8 - D15																				
Data Path Width:	8-bit	16-bit																				
P0L.0 – P0L.7:	AD0 – AD7	AD0 - AD7																				
P0H.0 – P0H.7:	A8 - A15	AD8 - AD15																				
PORT1: P1L.0 – P1L.7, P1H.0 - P1H.7	56 – 63, 66 – 73	58 - 65, 68 - 75	I/O	<p>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 and also after switching from a demultiplexed bus mode to a multiplexed bus mode.</p>																		
RSTIN	76	78	I	<p>Reset Input with Schmitt-Trigger characteristics. A low level at this pin for a specified duration while the oscillator is running resets the C161RI. An internal pullup resistor permits power-on reset using only a capacitor connected to <math>V_{SS}</math>.</p> <p>In bidirectional reset mode (enabled by setting bit BDRSTEN in register SYSCON) the <math>\overline{RSTIN}</math> line is internally pulled low for the duration of the internal reset sequence upon a software reset, a WDT reset and a hardware reset. <sup>1)</sup></p>																		
RSTOUT	77	79	O	<p>Internal Reset Indication Output. This pin is set to a low level when the part is executing either a hardware-, a software- or a watchdog timer reset. <math>\overline{RSTOUT}</math> remains low until the EINIT (end of initialization) instruction is executed.</p>																		
NMI	78	80	I	<p>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 <math>\overline{NMI}</math> pin must be low in order to force the C161RI to go into power down mode. If <math>\overline{NMI}</math> is high, when PWRDN is executed, the part will continue to run in normal mode.</p> <p>If not used, pin <math>\overline{NMI}</math> should be pulled high externally.</p>																		

## Pin Definitions and Functions (cont'd)

Symbol	Pin No. TQFP	Pin No. MQFP	Input Outp	Function
P6.0 – P6.7	79 – 86	81 – 88	I/O I/O	Port 6 is an 8-bit bidirectional I/O port. 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. The Port 6 pins also serve for alternate functions: P6.0 $\overline{CS0}$ Chip Select 0 Output ...     ...     ... P6.4 $\overline{CS4}$ Chip Select 4 Output P6.5     SDA1     I <sup>2</sup> C Bus Data Line 1 P6.6     SCL1     I <sup>2</sup> C Bus Clock Line 1 P6.7     SDA2     I <sup>2</sup> C Bus Data Line 2 <b>Note:</b> Pins P6.5-P6.7 are open drain outputs only.
P2.8 – P2.15	87 – 94	89 – 96	I/O I/O	Port 2 is an 8-bit bidirectional I/O port. 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. The following Port 2 pins also serve for alternate functions: P2.8     EX0IN     Fast External Interrupt 0 Input ...     ...     ... P2.15     EX7IN     Fast External Interrupt 7 Input
V <sub>AREF</sub>	95	97	-	Reference voltage for the A/D converter.
V <sub>AGND</sub>	96	98	-	Reference ground for the A/D converter.
V <sub>DD</sub>	6, 23, 37, 47, 65, 75	8, 25, 39, 49, 67, 77	-	Digital Supply Voltage: + 5 V during normal operation and idle mode. ≥ 2.5 V during power down mode
V <sub>SS</sub>	3, 22, 36, 46, 64, 74	5, 24, 38, 48, 66, 76	-	Digital Ground.

1) The following behavior differences must be observed when the bidirectional reset is active:

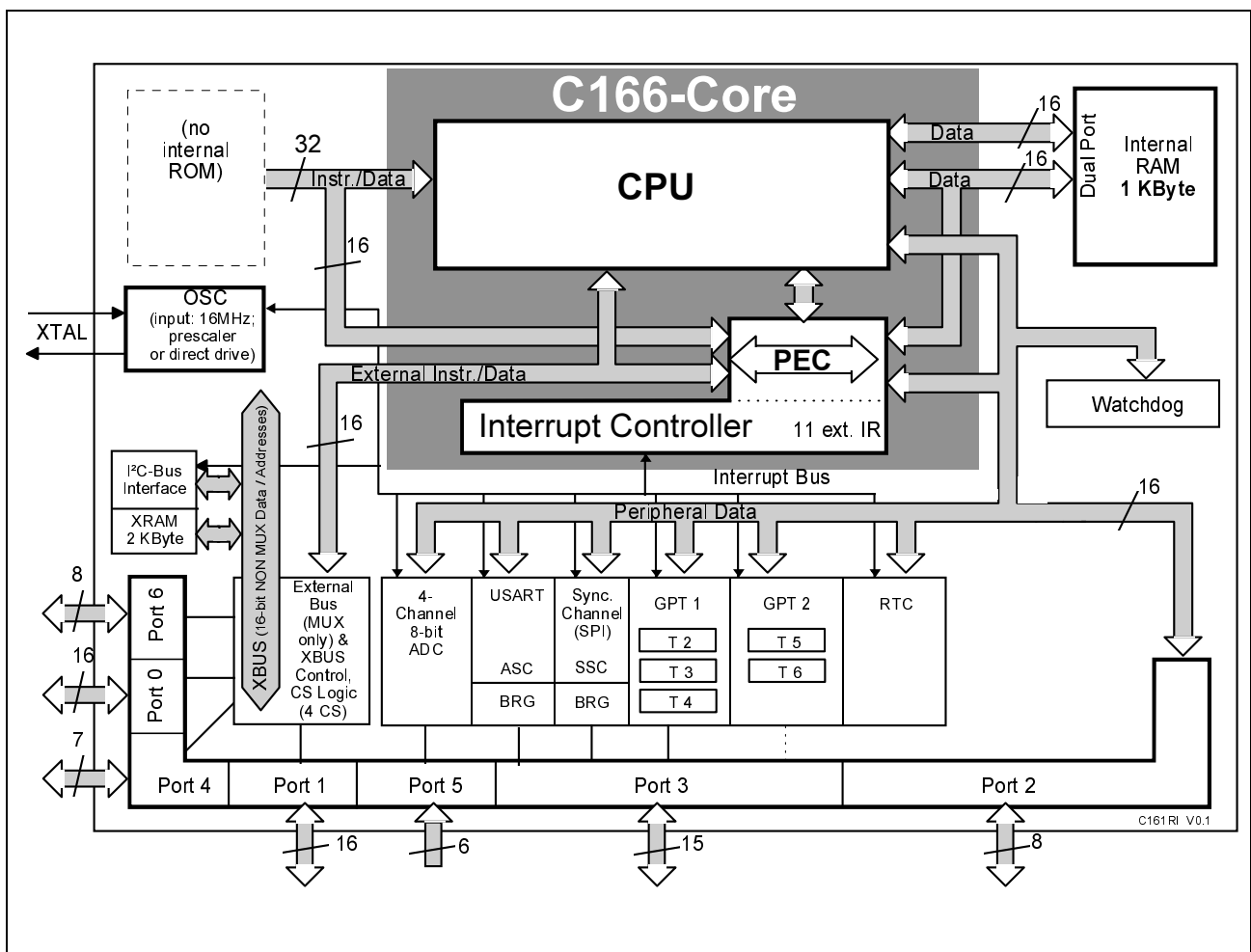
- Bit BDRSTEN in register SYSCON cannot be changed after EINIT.
- After a reset bit BDRSTEN is cleared.
- The reset indication flags always indicate a long hardware reset.
- The PORT0 configuration is treated like on a hardware reset. Especially the bootstrap loader may be activated when P0L.4 is low.
- Pin  $\overline{RSTIN}$  may only be connected to external reset devices with an open drain output driver.
- A short hardware reset is extended to the duration of the internal reset sequence.

**Functional Description**

The C161RI is a low cost downgrade of the high performance microcontroller C167CR with OTP or internal ROM, reduced peripheral functionality and a high performance Capture Compare Unit with an additional functionality.

The architecture of the C161RI combines advantages of both RISC and CISC processors and of advanced peripheral subsystems in a very well-balanced way. The following block diagram gives an overview of the different on-chip components and of the advanced, high bandwidth internal bus structure of the C161RI.

**Note:** All time specifications refer to a CPU clock of 16 MHz (see definition in the AC Characteristics section).



**Figure 4**  
**Block Diagram**

### Memory Organization

The memory space of the C161RI is configured in a Von Neumann architecture which means that code memory, data memory, registers and I/O ports are organized within the same linear address space which includes 16 MBytes. The entire memory space can be accessed bitwise or wordwise. Particular portions of the on-chip memory have additionally been made directly bitaddressable.

1 KByte of on-chip Internal RAM is provided as a storage for user defined variables, for the system stack, general purpose register banks and even for code. A register bank can consist of up to 16 wordwise (R0 to R15) and/or bitwise (RL0, RH0, ..., RL7, RH7) so-called General Purpose Registers (GPRs).

1024 bytes ( $2 \times 512$  bytes) of the address space are reserved for the Special Function Register areas (SFR space and ESFR space). SFRs are wordwise 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 C166 family.

In order to meet the needs of designs where more memory is required than is provided on chip, up to 8 MBytes of external RAM and/or ROM can be connected to the microcontroller.

### 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, which are as follows:

- 16-/18-/20-/23-bit Addresses, 16-bit Data, Demultiplexed
- 16-/18-/20-/23-bit Addresses, 16-bit Data, Multiplexed
- 16-/18-/20-/23-bit Addresses, 8-bit Data, Multiplexed
- 16-/18-/20-/23-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.

Important timing characteristics of the external bus interface (Memory Cycle Time, Memory Tri-State Time, Length of ALE and Read Write Delay) have been made programmable 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 register pairs ADDRSELx / BUSCONx) which allow to access different resources with different bus characteristics. These address windows are arranged hierarchically where BUSCON4 overrides BUSCON3 and BUSCON2 overrides BUSCON1. All accesses to locations not covered by these 4 address windows are controlled by BUSCON0.

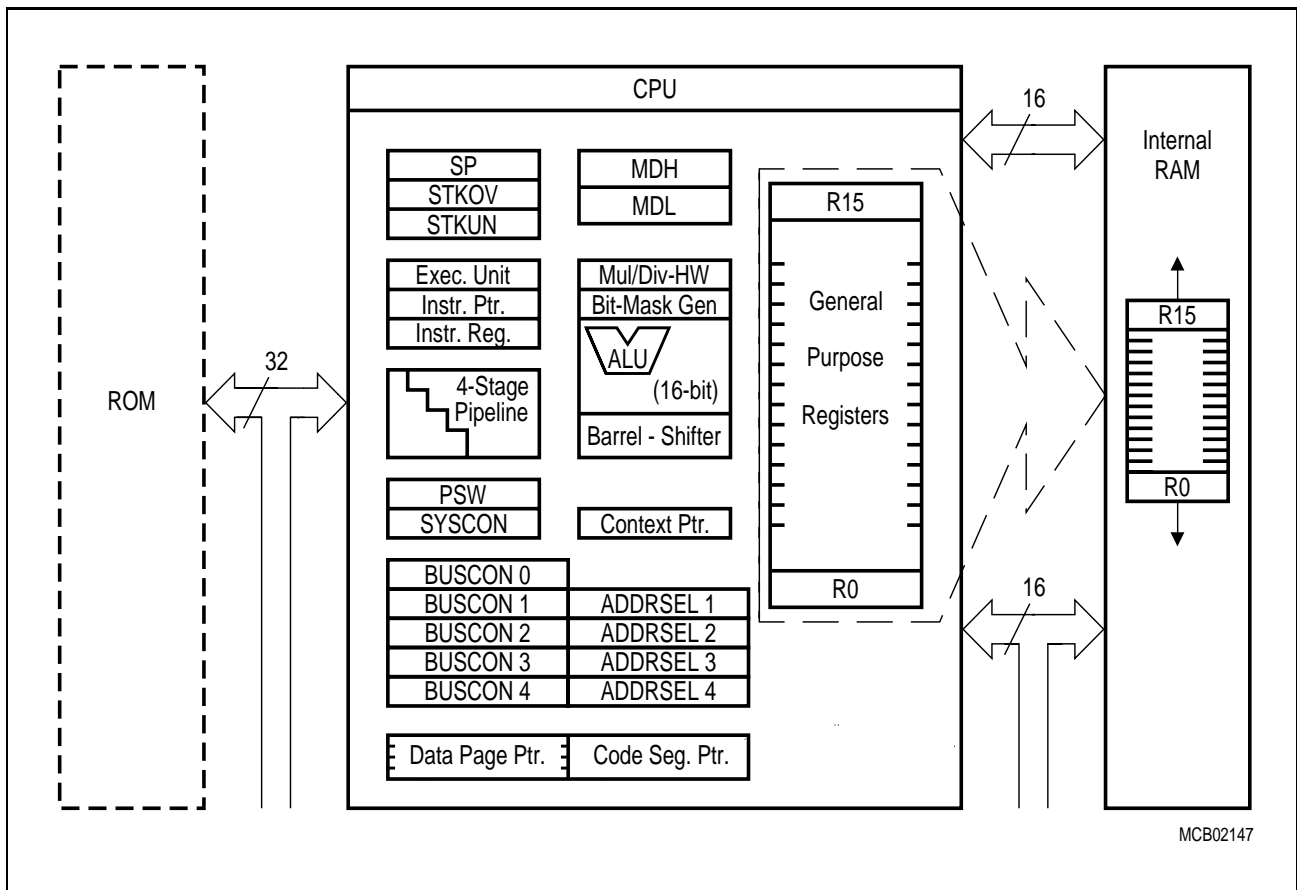
Up to 5 external  $\overline{CS}$  signals (4 windows plus default) can be generated in order to save external glue logic. Access to very slow memories is supported via a particular 'Ready' function.

For applications which require less than 8 MBytes of external memory space, this address space can be restricted to 1 MByte, 256 KByte or to 64 KByte. In this case Port 4 outputs four, two or no address lines at all. It outputs all 7 address lines, if an address space of 8 MBytes is used.

**Central Processing Unit (CPU)**

The main core of the CPU consists of a 4-stage instruction pipeline, a 16-bit arithmetic and logic unit (ALU) and dedicated SFRs. Additional hardware has been spent for a separate multiply and divide unit, a bit-mask generator and a barrel shifter.

Based on these hardware provisions, most of the C161RI's instructions can be executed in just one machine cycle which requires 125 ns at 16 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. All multiple-cycle instructions have been optimized so that they can be executed very fast as well: branches in 2 cycles, a  $16 \times 16$  bit multiplication in 5 cycles and a 32-/16 bit division in 10 cycles. Another pipeline optimization, the so-called 'Jump Cache', allows reducing the execution time of repeatedly performed jumps in a loop from 2 cycles to 1 cycle.



MCB02147

**Figure 5  
CPU Block Diagram**

The CPU disposes of an actual register context consisting of up to 16 wordwide GPRs which are physically allocated within the on-chip RAM area. A Context Pointer (CP) register determines the base address of the active register bank to be accessed by the CPU at a 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 1024 bytes is provided as a storage for temporary data. The system stack is allocated 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 C161RI instruction set which includes the following instruction classes:

- 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.

## Interrupt System

With an interrupt response time within a range from just 315 ns to 750 ns (in case of internal program execution), the C161RI is capable of reacting very fast to the occurrence of non-deterministic events.

The architecture of the C161RI 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. 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 C161RI 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 sources. Via its related register, each source 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 sources 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.



The following table shows all of the possible C161RI interrupt sources and the corresponding hardware-related interrupt flags, vectors, vector locations and trap (interrupt) numbers:

Source of Interrupt or PEC Service Request	Request Flag	Enable Flag	Interrupt Vector	Vector Location	Trap Number
External Interrupt 0	CC8IR	CC8IE	CC8INT	00'0060 <sub>H</sub>	18 <sub>H</sub>
External Interrupt 1	CC9IR	CC9IE	CC9INT	00'0064 <sub>H</sub>	19 <sub>H</sub>
External Interrupt 2	CC10IR	CC10IE	CC10INT	00'0068 <sub>H</sub>	1A <sub>H</sub>
External Interrupt 3	CC11IR	CC11IE	CC11INT	00'006C <sub>H</sub>	1B <sub>H</sub>
External Interrupt 4	CC12IR	CC12IE	CC12INT	00'0070 <sub>H</sub>	1C <sub>H</sub>
External Interrupt 5	CC13IR	CC13IE	CC13INT	00'0074 <sub>H</sub>	1D <sub>H</sub>
External Interrupt 6	CC14IR	CC14IE	CC14INT	00'0078 <sub>H</sub>	1E <sub>H</sub>
External Interrupt 7	CC15IR	CC15IE	CC15INT	00'007C <sub>H</sub>	1F <sub>H</sub>
GPT1 Timer 2	T2IR	T2IE	T2INT	00'0088 <sub>H</sub>	22 <sub>H</sub>
GPT1 Timer 3	T3IR	T3IE	T3INT	00'008C <sub>H</sub>	23 <sub>H</sub>
GPT1 Timer 4	T4IR	T4IE	T4INT	00'0090 <sub>H</sub>	24 <sub>H</sub>
GPT2 Timer 5	T5IR	T5IE	T5INT	00'0094 <sub>H</sub>	25 <sub>H</sub>
GPT2 Timer 6	T6IR	T6IE	T6INT	00'0098 <sub>H</sub>	26 <sub>H</sub>
GPT2 CAPREL Register	CRIR	CRIE	CRINT	00'009C <sub>H</sub>	27 <sub>H</sub>
A/D Conversion Complete	ADCIR	ADCIE	ADCINT	00'00A0 <sub>H</sub>	28 <sub>H</sub>
A/D Overrun Error	ADEIR	ADEIE	ADEINT	00'00A4 <sub>H</sub>	29 <sub>H</sub>
ASC0 Transmit	S0TIR	S0TIE	S0TINT	00'00A8 <sub>H</sub>	2A <sub>H</sub>
ASC0 Transmit Buffer	S0TBIR	S0TBIE	S0TBINT	00'011C <sub>H</sub>	47 <sub>H</sub>
ASC0 Receive	S0RIR	S0RIE	S0RINT	00'00AC <sub>H</sub>	2B <sub>H</sub>
ASC0 Error	S0EIR	S0EIE	S0EINT	00'00B0 <sub>H</sub>	2C <sub>H</sub>
SSC Transmit	SCTIR	SCTIE	SCTINT	00'00B4 <sub>H</sub>	2D <sub>H</sub>
SSC Receive	SCRIR	SCRIE	SCRINT	00'00B8 <sub>H</sub>	2E <sub>H</sub>
SSC Error	SCEIR	SCEIE	SCEINT	00'00BC <sub>H</sub>	2F <sub>H</sub>
I <sup>2</sup> C Data Transfer Event	XP0IR	XP0IE	XP0INT	00'0100 <sub>H</sub>	40 <sub>H</sub>
I <sup>2</sup> C Protocol Event	XP1IR	XP1IE	XP1INT	00'0104 <sub>H</sub>	41 <sub>H</sub>
X-Peripheral Node 2	XP2IR	XP2IE	XP2INT	00'0108 <sub>H</sub>	42 <sub>H</sub>
PLL Unlock / RTC	XP3IR	XP3IE	XP3INT	00'010C <sub>H</sub>	43 <sub>H</sub>

The C161RI also provides an excellent mechanism to identify and to process exceptions or error conditions that arise during run-time, so-called 'Hardware Traps'. Hardware traps cause immediate non-maskable system reaction which is similar to a standard interrupt service (branching to a dedicated vector table location). The occurrence of a hardware trap is additionally signified by an individual bit in the trap flag register (TFR). Except when another higher prioritized trap service is in progress, a hardware trap will interrupt any actual program execution. In turn, hardware trap services can normally not be interrupted by standard or PEC interrupts.

The following table shows all of the possible exceptions or error conditions that can arise during run-time:

Exception Condition	Trap Flag	Trap Vector	Vector Location	Trap Number	Trap Priority
Reset Functions:					
Hardware Reset		RESET	00'0000 <sub>H</sub>	00 <sub>H</sub>	III
Software Reset		RESET	00'0000 <sub>H</sub>	00 <sub>H</sub>	III
Watchdog Timer Overflow		RESET	00'0000 <sub>H</sub>	00 <sub>H</sub>	III
Class A Hardware Traps:					
Non-Maskable Interrupt	NMI	NMITRAP	00'0008 <sub>H</sub>	02 <sub>H</sub>	II
Stack Overflow	STKOF	STOTRAP	00'0010 <sub>H</sub>	04 <sub>H</sub>	II
Stack Underflow	STKUF	STUTRAP	00'0018 <sub>H</sub>	06 <sub>H</sub>	II
Class B Hardware Traps:					
Undefined Opcode	UNDOPC	BTRAP	00'0028 <sub>H</sub>	0A <sub>H</sub>	I
Protected Instruction Fault	PRTFLT	BTRAP	00'0028 <sub>H</sub>	0A <sub>H</sub>	I
Illegal Word Operand Access	ILLOPA	BTRAP	00'0028 <sub>H</sub>	0A <sub>H</sub>	I
Illegal Instruction Access	ILLINA	BTRAP	00'0028 <sub>H</sub>	0A <sub>H</sub>	I
Illegal External Bus Access	ILLBUS	BTRAP	00'0028 <sub>H</sub>	0A <sub>H</sub>	I
Reserved			[2C <sub>H</sub> – 3C <sub>H</sub> ]	[0B <sub>H</sub> – 0F <sub>H</sub> ]	
Software Traps			Any	Any	Current CPU Priority
TRAP Instruction			[00'0000 <sub>H</sub> – 00'01FC <sub>H</sub> ] in steps of 4 <sub>H</sub>	[00 <sub>H</sub> – 7F <sub>H</sub> ]	

## General Purpose Timer (GPT) Unit

The GPT 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 GPT 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 CPU 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 500 ns (@ 16 MHz CPU clock).

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 a port pin (T3OUT) e.g. for time out monitoring of external hardware components, or may be used internally to clock timers T2 and T4 for measuring long time periods with high resolution.

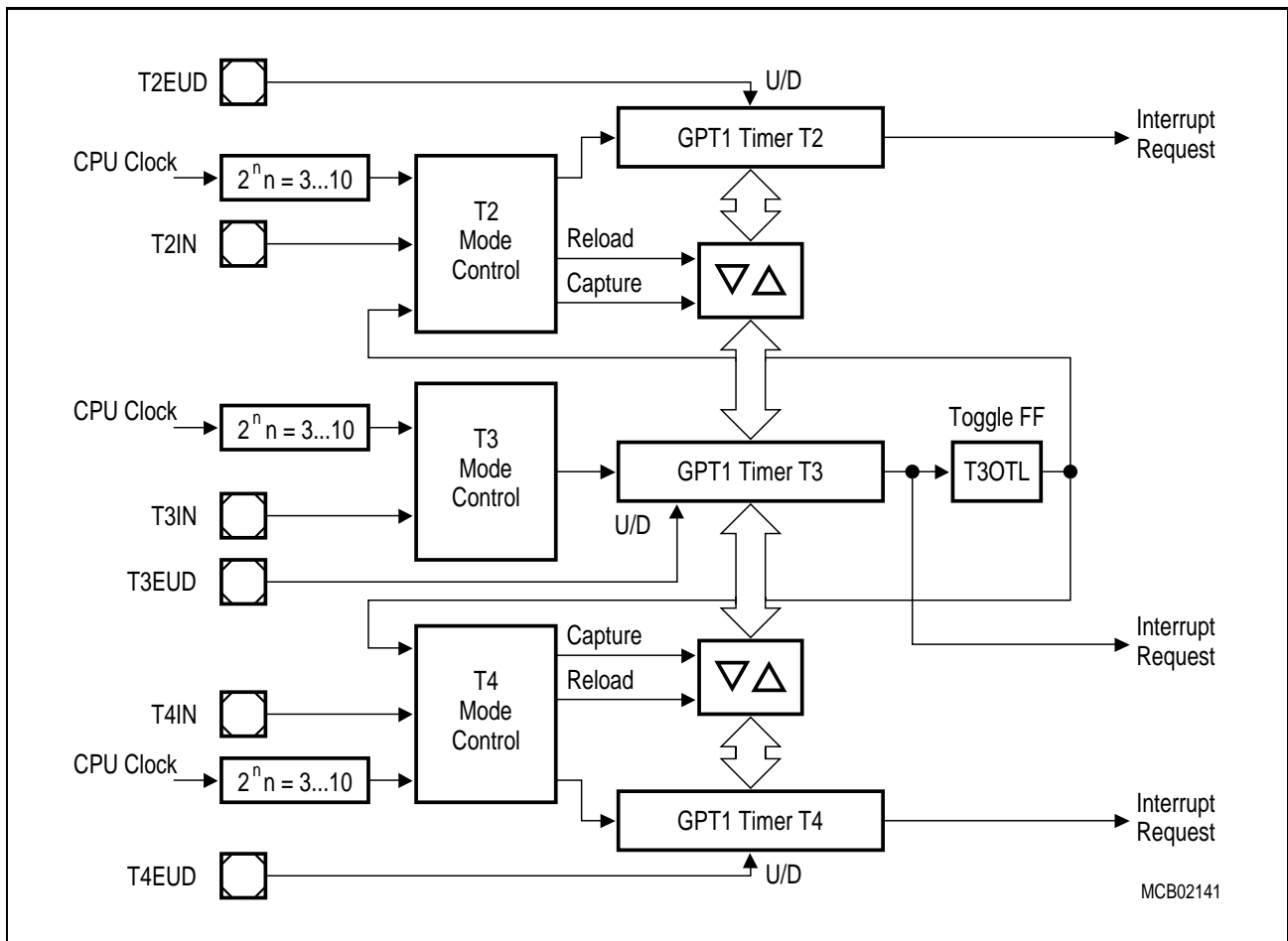
In addition to their basic operating modes, timers T2 and T4 may be configured as reload or capture registers for timer T3. When used as capture or reload registers, timers T2 and T4 are stopped. The contents of timer T3 are captured into T2 or T4 in response to a signal at their associated input pins (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.

With its maximum resolution of 250 ns (@ 16 MHz), 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. The count direction (up/down) for each timer is programmable by software. Concatenation of the timers is supported via the output toggle latch (T6OTL) of timer T6, which changes its state on each timer overflow/underflow.

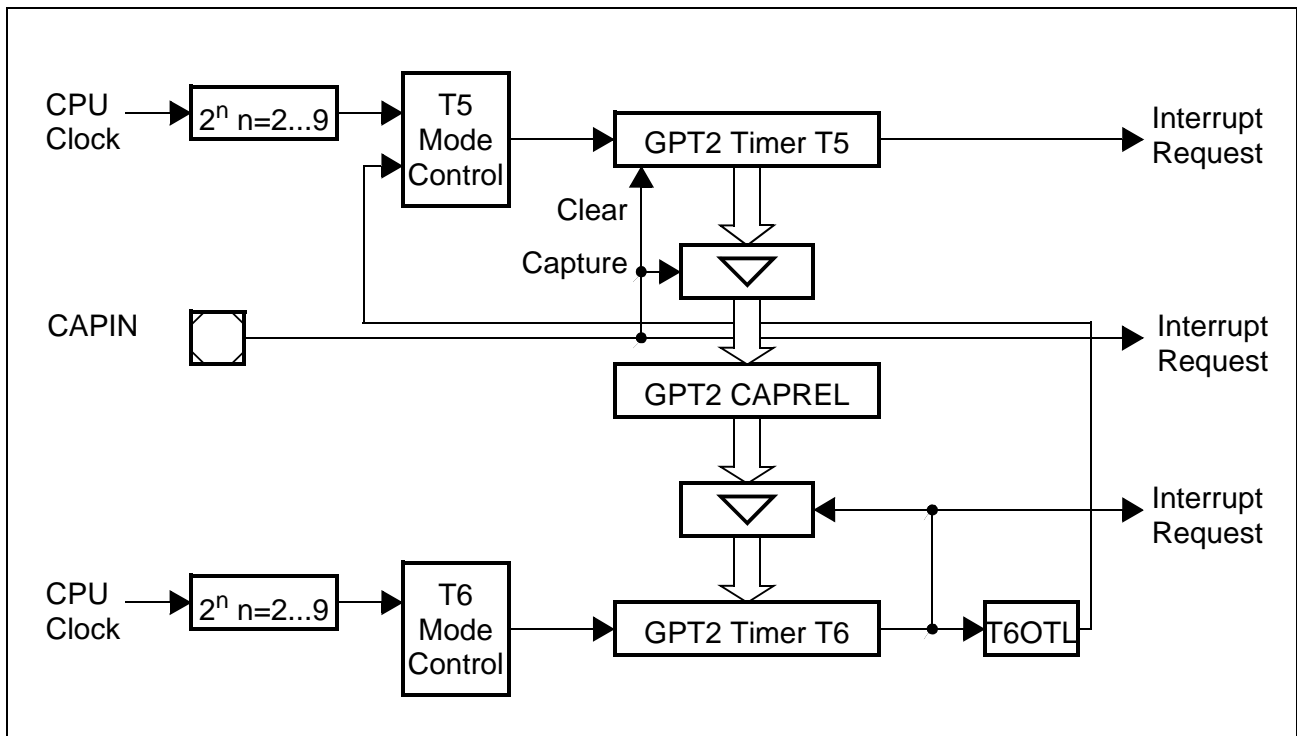
The state of this latch may be used to clock timer T5. The overflows/underflows of timer T6 can additionally be used to cause a reload from the CAPREL register. The CAPREL register may capture the contents of timer T5 based on an external signal transition on the corresponding port pin

(CAPIN), and timer T5 may optionally be cleared after the capture procedure. This allows absolute time differences to be measured or pulse multiplication to be performed 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 6**  
**Block Diagram of GPT1**



**Figure 7**  
**Block Diagram of GPT2**

**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 only be disabled in the time interval until the EINIT (end of initialization) instruction has been executed. Thus, the chip's start-up procedure is always monitored. The software has to be designed to service 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 RSTOUT pin low in order to allow external hardware components to be reset.

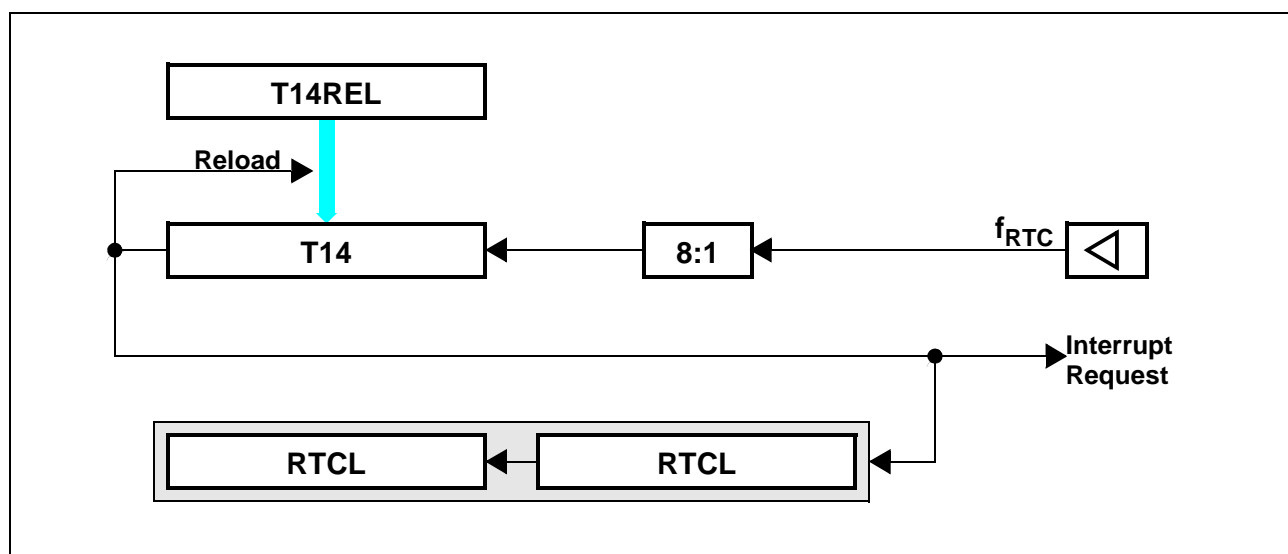
The Watchdog Timer is a 16-bit timer, clocked with the system clock divided either by 2 or by 128. The high byte of the Watchdog Timer register can be set to a prespecified reload value (stored in WDTREL) in order to allow further variation of the monitored time interval. Each time it is serviced by the application software, the high byte of the Watchdog Timer is reloaded. Thus, time intervals between 31 μs and 525 ms can be monitored (@ 16 MHz). The default Watchdog Timer interval after reset is 8.2 ms (@ 16 MHz).

## Real Time Clock

The Real Time Clock (RTC) module of the C161RI consists of a chain of 3 divider blocks, a fixed 8-bit divider, the reloadable 16-bit timer T14 and the 32-bit RTC timer (accessible via registers RTCH and RTCL). The RTC module is directly clocked with the on-chip oscillator frequency divided by 32 via a separate clock driver and is therefore independent from the selected clock generation mode of the C161RI. All timers count up.

The RTC module can be used for different purposes:

- System clock to determine the current time and date
- Cyclic time based interrupt
- 48-bit timer for long term measurements



**Figure 7-1**  
**RTC Block Diagram**

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

### A/D Converter

For analog signal measurement, an 8-bit A/D converter with 4 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 and can so be adjusted to the external circuitry.

Overflow error detection is provided for the conversion result register (ADDAT): 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.

For applications which require less than 4 analog input channels, the remaining channel inputs can be used as digital input port pins.

The A/D converter of the C161RI supports two 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.

The 8-bit result can be left-aligned or right-aligned within a 10-bit result area.

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.

### I<sup>2</sup>C Module

The integrated I<sup>2</sup>C Bus Module handles the transmission and reception of frames over the two-line I<sup>2</sup>C bus in accordance with the I<sup>2</sup>C Bus specification. The on-chip I<sup>2</sup>C Module can receive and transmit data using 7-bit or 10-bit addressing and it can operate in slave mode, in master mode or in multi-master mode.

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 I<sup>2</sup>C module allow efficient interrupt service and also support operation via PEC transfers.

**Note:** The port pins associated with the I<sup>2</sup>C interfaces feature open drain drivers only, as required by the I<sup>2</sup>C specification.

## Serial Channels

Serial communication with other microcontrollers, processors, terminals or external peripheral components is provided by two serial interfaces with different functionality, an Asynchronous/Synchronous Serial Channel (**ASC0**) and a High-Speed Synchronous Serial Channel (**SSC**).

**The ASC0** is upward compatible with the serial ports of the Siemens 8-bit microcontroller families and supports full-duplex asynchronous communication at up to 500 KBaud and half-duplex synchronous communication at up to 2 MBaud @ 16 MHz CPU clock.

A dedicated baud rate generator allows to set up all standard baud rates without oscillator tuning. For transmission, reception and error handling 4 separate interrupt vectors are provided. In asynchronous mode, 8- or 9-bit data frames 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).

In synchronous mode, the ASC0 transmits or receives bytes (8 bits) synchronously to a shift clock which is generated by the ASC0. The ASC0 always shifts the LSB first. A loop back option is available for testing purposes.

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.

**The SSC** supports full-duplex synchronous communication at up to 4 Mbaud @ 16 MHz CPU clock. It may be configured so it interfaces with serially linked peripheral components. A dedicated baud rate generator allows to set up all standard baud rates without oscillator tuning. For transmission, reception and error handling 3 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 and receive error supervise the correct handling of the data buffer. Phase and baudrate error detect incorrect serial data.



## Parallel Ports

The C161RI provides up to 76 IO lines which are organized into six 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 three IO ports can be configured (pin by pin) for push/pull operation or open-drain operation via control registers. The other IO ports operate in push/pull mode, except for the I<sup>2</sup>C interface pins which are open drain pins only. During the internal reset, all port pins are configured as inputs.

All port lines have programmable alternate input or output functions associated with them.

PORT0 and PORT1 may be used as address and data lines when accessing external memory, while Port 4 outputs the additional segment address bits A22/19/17...A16 in systems where segmentation is enabled to access more than 64 KBytes of memory.

Port 3 includes alternate functions of timers, serial interfaces, the optional bus control signal  $\overline{\text{BHE}}$  and the system clock output (CLKOUT).

Port 5 is used for the analog input channels to the A/D converter or timer control signals.

Port 6 provides the optional chip select signals and interface lines for the I<sup>2</sup>C module.

All port lines that are not used for these alternate functions may be used as general purpose IO lines.

### Instruction Set Summary

The table below lists the instructions of the C161RI 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 “**C166 Family Instruction Set Manual**”.

This document also provides a detailed description of each instruction.

### 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
OR(B)	Bitwise OR, (word/byte operands)	2 / 4
XOR(B)	Bitwise XOR, (word/byte operands)	2 / 4
BCLR	Clear direct bit	2
BSET	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/L	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

## Instruction Set Summary (cont'd)

Mnemonic	Description	Bytes
MOV(B)	Move word (byte) data	2 / 4
MOVBS	Move byte operand to word operand with sign extension	2 / 4
MOVZ	Move byte operand to word operand. with zero extension	2 / 4
JMPA, JMPI, JMPR	Jump absolute/indirect/relative if condition is met	4
JMPS	Jump absolute to a code segment	4
J(N)B	Jump relative if direct bit is (not) set	4
JBC	Jump relative and clear bit if direct bit is set	4
JNBS	Jump relative and set bit if direct bit is not set	4
CALLA, CALLI, CALLR	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	Return from intra-segment subroutine	2
RETS	Return from inter-segment subroutine	2
RETP	Return from intra-segment subroutine and pop direct word register from system stack	2
RETI	Return from interrupt service subroutine	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	Disable 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

## Special Function Registers Overview

The following table lists all SFRs which are implemented in the C161RI in alphabetical order.

**Bit-addressable** SFRs are marked with the letter “b” in column “Name”. SFRs within the **Extended SFR-Space** (ESFRs) are marked with the letter “E” in column “Physical Address”. Registers within on-chip X-Peripherals (I<sup>2</sup>C) are marked with the letter “X” in column “Physical Address”.

An SFR can be specified via its individual mnemonic name. Depending on the selected addressing mode, an SFR can be accessed via its physical address (using the Data Page Pointers), or via its short 8-bit address (without using the Data Page Pointers).

Name	Physical Address	8-Bit Address	Description	Reset Value
<b>ADCIC</b>	b FF98 <sub>H</sub>	CC <sub>H</sub>	A/D Converter End of Conversion Interrupt Control Register	0000 <sub>H</sub>
<b>ADCON</b>	b FFA0 <sub>H</sub>	D0 <sub>H</sub>	A/D Converter Control Register	0000 <sub>H</sub>
<b>ADDAT</b>	FEA0 <sub>H</sub>	50 <sub>H</sub>	A/D Converter Result Register	0000 <sub>H</sub>
<b>ADDRSEL1</b>	FE18 <sub>H</sub>	0C <sub>H</sub>	Address Select Register 1	0000 <sub>H</sub>
<b>ADDRSEL2</b>	FE1A <sub>H</sub>	0D <sub>H</sub>	Address Select Register 2	0000 <sub>H</sub>
<b>ADDRSEL3</b>	FE1C <sub>H</sub>	0E <sub>H</sub>	Address Select Register 3	0000 <sub>H</sub>
<b>ADDRSEL4</b>	FE1E <sub>H</sub>	0F <sub>H</sub>	Address Select Register 4	0000 <sub>H</sub>
<b>ADEIC</b>	b FF9A <sub>H</sub>	CD <sub>H</sub>	A/D Converter Overrun Error Interrupt Control Register	0000 <sub>H</sub>
<b>BUSCON0</b>	b FF0C <sub>H</sub>	86 <sub>H</sub>	Bus Configuration Register 0	0000 <sub>H</sub>
<b>BUSCON1</b>	b FF14 <sub>H</sub>	8A <sub>H</sub>	Bus Configuration Register 1	0000 <sub>H</sub>
<b>BUSCON2</b>	b FF16 <sub>H</sub>	8B <sub>H</sub>	Bus Configuration Register 2	0000 <sub>H</sub>
<b>BUSCON3</b>	b FF18 <sub>H</sub>	8C <sub>H</sub>	Bus Configuration Register 3	0000 <sub>H</sub>
<b>BUSCON4</b>	b FF1A <sub>H</sub>	8D <sub>H</sub>	Bus Configuration Register 4	0000 <sub>H</sub>
<b>CAPREL</b>	FE4A <sub>H</sub>	25 <sub>H</sub>	GPT2 Capture/Reload Register	0000 <sub>H</sub>
<b>CC8IC</b>	b FF88 <sub>H</sub>	C4 <sub>H</sub>	External Interrupt 0 Control Register	0000 <sub>H</sub>
<b>CC9IC</b>	b FF8A <sub>H</sub>	C5 <sub>H</sub>	External Interrupt 1 Control Register	0000 <sub>H</sub>
<b>CC10IC</b>	b FF8C <sub>H</sub>	C6 <sub>H</sub>	External Interrupt 2 Control Register	0000 <sub>H</sub>
<b>CC11IC</b>	b FF8E <sub>H</sub>	C7 <sub>H</sub>	External Interrupt 3 Control Register	0000 <sub>H</sub>
<b>CC12IC</b>	b FF90 <sub>H</sub>	C8 <sub>H</sub>	External Interrupt 4 Control Register	0000 <sub>H</sub>
<b>CC13IC</b>	b FF92 <sub>H</sub>	C9 <sub>H</sub>	External Interrupt 5 Control Register	0000 <sub>H</sub>
<b>CC14IC</b>	b FF94 <sub>H</sub>	CA <sub>H</sub>	External Interrupt 6 Control Register	0000 <sub>H</sub>
<b>CC15IC</b>	b FF96 <sub>H</sub>	CB <sub>H</sub>	External Interrupt 7 Control Register	0000 <sub>H</sub>
<b>CP</b>	FE10 <sub>H</sub>	08 <sub>H</sub>	CPU Context Pointer Register	FC00 <sub>H</sub>

Name		Physical Address	8-Bit Address	Description	Reset Value
<b>CRIC</b>	<b>b</b>	FF6A <sub>H</sub>	B5 <sub>H</sub>	GPT2 CAPREL Interrupt Control Register	0000 <sub>H</sub>
<b>CSP</b>		FE08 <sub>H</sub>	04 <sub>H</sub>	CPU Code Segment Pointer Register (8 bits, not directly writeable)	0000 <sub>H</sub>
<b>DP0L</b>	<b>b</b>	F100 <sub>H</sub>	<b>E</b> 80 <sub>H</sub>	P0L Direction Control Register	00 <sub>H</sub>
<b>DP0H</b>	<b>b</b>	F102 <sub>H</sub>	<b>E</b> 81 <sub>H</sub>	P0H Direction Control Register	00 <sub>H</sub>
<b>DP1L</b>	<b>b</b>	F104 <sub>H</sub>	<b>E</b> 82 <sub>H</sub>	P1L Direction Control Register	00 <sub>H</sub>
<b>DP1H</b>	<b>b</b>	F106 <sub>H</sub>	<b>E</b> 83 <sub>H</sub>	P1H Direction Control Register	00 <sub>H</sub>
<b>DP2</b>	<b>b</b>	FFC2 <sub>H</sub>	E1 <sub>H</sub>	Port 2 Direction Control Register	0000 <sub>H</sub>
<b>DP3</b>	<b>b</b>	FFC6 <sub>H</sub>	E3 <sub>H</sub>	Port 3 Direction Control Register	0000 <sub>H</sub>
<b>DP4</b>	<b>b</b>	FFCA <sub>H</sub>	E5 <sub>H</sub>	Port 4 Direction Control Register	00 <sub>H</sub>
<b>DP6</b>	<b>b</b>	FFCE <sub>H</sub>	E7 <sub>H</sub>	Port 6 Direction Control Register	00 <sub>H</sub>
<b>DPP0</b>		FE00 <sub>H</sub>	00 <sub>H</sub>	CPU Data Page Pointer 0 Register (10 bits)	0000 <sub>H</sub>
<b>DPP1</b>		FE02 <sub>H</sub>	01 <sub>H</sub>	CPU Data Page Pointer 1 Register (10 bits)	0001 <sub>H</sub>
<b>DPP2</b>		FE04 <sub>H</sub>	02 <sub>H</sub>	CPU Data Page Pointer 2 Register (10 bits)	0002 <sub>H</sub>
<b>DPP3</b>		FE06 <sub>H</sub>	03 <sub>H</sub>	CPU Data Page Pointer 3 Register (10 bits)	0003 <sub>H</sub>
<b>EXICON</b>	<b>b</b>	F1C0 <sub>H</sub>	<b>E</b> E0 <sub>H</sub>	External Interrupt Control Register	0000 <sub>H</sub>
<b>ICADR</b>		ED06 <sub>H</sub>	<b>X</b> ---	I <sup>2</sup> C Address Register	0XXX <sub>H</sub>
<b>ICCFG</b>		ED00 <sub>H</sub>	<b>X</b> ---	I <sup>2</sup> C Configuration Register	XX00 <sub>H</sub>
<b>ICCON</b>		ED02 <sub>H</sub>	<b>X</b> ---	I <sup>2</sup> C Control Register	0000 <sub>H</sub>
<b>ICRTB</b>		ED08 <sub>H</sub>	<b>X</b> ---	I <sup>2</sup> C Receive/Transmit Buffer	XX <sub>H</sub>
<b>ICST</b>		ED04 <sub>H</sub>	<b>X</b> ---	I <sup>2</sup> C Status Register	0000 <sub>H</sub>
<b>IDCHIP</b>		F07C <sub>H</sub>	<b>E</b> 3E <sub>H</sub>	Identifier	09XX <sub>H</sub>
<b>IDMANUF</b>		F07E <sub>H</sub>	<b>E</b> 3F <sub>H</sub>	Identifier	1820 <sub>H</sub>
<b>IDMEM</b>		F07A <sub>H</sub>	<b>E</b> 3D <sub>H</sub>	Identifier	0000 <sub>H</sub>
<b>IDPROG</b>		F078 <sub>H</sub>	<b>E</b> 3C <sub>H</sub>	Identifier	0000 <sub>H</sub>
<b>ISNC</b>	<b>b</b>	F1DE <sub>H</sub>	<b>E</b> EF <sub>H</sub>	Interrupt Subnode Control Register	0000 <sub>H</sub>
<b>MDC</b>	<b>b</b>	FF0E <sub>H</sub>	87 <sub>H</sub>	CPU Multiply Divide Control Register	0000 <sub>H</sub>
<b>MDH</b>		FE0C <sub>H</sub>	06 <sub>H</sub>	CPU Multiply Divide Register – High Word	0000 <sub>H</sub>
<b>MDL</b>		FE0E <sub>H</sub>	07 <sub>H</sub>	CPU Multiply Divide Register – Low Word	0000 <sub>H</sub>
<b>ODP2</b>	<b>b</b>	F1C2 <sub>H</sub>	<b>E</b> E1 <sub>H</sub>	Port 2 Open Drain Control Register	0000 <sub>H</sub>
<b>ODP3</b>	<b>b</b>	F1C6 <sub>H</sub>	<b>E</b> E3 <sub>H</sub>	Port 3 Open Drain Control Register	0000 <sub>H</sub>
<b>ODP6</b>	<b>b</b>	F1CE <sub>H</sub>	<b>E</b> E7 <sub>H</sub>	Port 6 Open Drain Control Register	00 <sub>H</sub>
<b>ONES</b>	<b>b</b>	FF1E <sub>H</sub>	8F <sub>H</sub>	Constant Value 1's Register (read only)	FFFF <sub>H</sub>

Name	Physical Address	8-Bit Address	Description	Reset Value
<b>P0L</b>	<b>b</b> FF00 <sub>H</sub>	80 <sub>H</sub>	Port 0 Low Register (Lower half of PORT0)	00 <sub>H</sub>
<b>P0H</b>	<b>b</b> FF02 <sub>H</sub>	81 <sub>H</sub>	Port 0 High Register (Upper half of PORT0)	00 <sub>H</sub>
<b>P1L</b>	<b>b</b> FF04 <sub>H</sub>	82 <sub>H</sub>	Port 1 Low Register (Lower half of PORT1)	00 <sub>H</sub>
<b>P1H</b>	<b>b</b> FF06 <sub>H</sub>	83 <sub>H</sub>	Port 1 High Register (Upper half of PORT1)	00 <sub>H</sub>
<b>P2</b>	<b>b</b> FFC0 <sub>H</sub>	E0 <sub>H</sub>	Port 2 Register	0000 <sub>H</sub>
<b>P3</b>	<b>b</b> FFC4 <sub>H</sub>	E2 <sub>H</sub>	Port 3 Register	0000 <sub>H</sub>
<b>P4</b>	<b>b</b> FFC8 <sub>H</sub>	E4 <sub>H</sub>	Port 4 Register (7 bits)	00 <sub>H</sub>
<b>P5</b>	<b>b</b> FFA2 <sub>H</sub>	D1 <sub>H</sub>	Port 5 Register (read only)	XXXX <sub>H</sub>
<b>P5DIDIS</b>	<b>b</b> FFA4 <sub>H</sub>	D2 <sub>H</sub>	Port 5 Digital Input Disable Register	0000 <sub>H</sub>
<b>P6</b>	<b>b</b> FFCC <sub>H</sub>	E6 <sub>H</sub>	Port 6 Register (8 bits)	00 <sub>H</sub>
<b>PECC0</b>	FEC0 <sub>H</sub>	60 <sub>H</sub>	PEC Channel 0 Control Register	0000 <sub>H</sub>
<b>PECC1</b>	FEC2 <sub>H</sub>	61 <sub>H</sub>	PEC Channel 1 Control Register	0000 <sub>H</sub>
<b>PECC2</b>	FEC4 <sub>H</sub>	62 <sub>H</sub>	PEC Channel 2 Control Register	0000 <sub>H</sub>
<b>PECC3</b>	FEC6 <sub>H</sub>	63 <sub>H</sub>	PEC Channel 3 Control Register	0000 <sub>H</sub>
<b>PECC4</b>	FEC8 <sub>H</sub>	64 <sub>H</sub>	PEC Channel 4 Control Register	0000 <sub>H</sub>
<b>PECC5</b>	FECA <sub>H</sub>	65 <sub>H</sub>	PEC Channel 5 Control Register	0000 <sub>H</sub>
<b>PECC6</b>	FECC <sub>H</sub>	66 <sub>H</sub>	PEC Channel 6 Control Register	0000 <sub>H</sub>
<b>PECC7</b>	FECE <sub>H</sub>	67 <sub>H</sub>	PEC Channel 7 Control Register	0000 <sub>H</sub>
<b>PSW</b>	<b>b</b> FF10 <sub>H</sub>	88 <sub>H</sub>	CPU Program Status Word	0000 <sub>H</sub>
<b>RP0H</b>	<b>b</b> F108 <sub>H</sub>	<b>E</b> 84 <sub>H</sub>	System Startup Configuration Register (Rd. only)	XX <sub>H</sub>
<b>RTCH</b>	F0D6 <sub>H</sub>	<b>E</b> 6B <sub>H</sub>	RTC High Register	no
<b>RTCL</b>	F0D4 <sub>H</sub>	<b>E</b> 6A <sub>H</sub>	RTC Low Register	no
<b>S0BG</b>	FEB4 <sub>H</sub>	5A <sub>H</sub>	Serial Channel 0 Baud Rate Generator Reload Register	0000 <sub>H</sub>
<b>S0CON</b>	<b>b</b> FFB0 <sub>H</sub>	D8 <sub>H</sub>	Serial Channel 0 Control Register	0000 <sub>H</sub>
<b>S0EIC</b>	<b>b</b> FF70 <sub>H</sub>	B8 <sub>H</sub>	Serial Channel 0 Error Interrupt Control Register	0000 <sub>H</sub>
<b>S0RBUF</b>	FEB2 <sub>H</sub>	59 <sub>H</sub>	Serial Channel 0 Receive Buffer Register (read only)	XXXX <sub>H</sub>
<b>S0RIC</b>	<b>b</b> FF6E <sub>H</sub>	B7 <sub>H</sub>	Serial Channel 0 Receive Interrupt Control Register	0000 <sub>H</sub>
<b>S0TBIC</b>	<b>b</b> F19C <sub>H</sub>	<b>E</b> CE <sub>H</sub>	Serial Channel 0 Transmit Buffer Interrupt Control Register	0000 <sub>H</sub>
<b>S0TBUF</b>	FEB0 <sub>H</sub>	58 <sub>H</sub>	Serial Channel 0 Transmit Buffer Register	0000 <sub>H</sub>

Name	Physical Address	8-Bit Address	Description	Reset Value
<b>S0TIC</b>	<b>b</b> FF6C <sub>H</sub>	B6 <sub>H</sub>	Serial Channel 0 Transmit Interrupt Control Register	0000 <sub>H</sub>
<b>SP</b>	FE12 <sub>H</sub>	09 <sub>H</sub>	CPU System Stack Pointer Register	FC00 <sub>H</sub>
<b>SSCBR</b>	F0B4 <sub>H</sub>	<b>E</b> 5A <sub>H</sub>	SSC Baudrate Register	0000 <sub>H</sub>
<b>SSCON</b>	<b>b</b> FFB2 <sub>H</sub>	D9 <sub>H</sub>	SSC Control Register	0000 <sub>H</sub>
<b>SSCEIC</b>	<b>b</b> FF76 <sub>H</sub>	BB <sub>H</sub>	SSC Error Interrupt Control Register	0000 <sub>H</sub>
<b>SSCRB</b>	F0B2 <sub>H</sub>	<b>E</b> 59 <sub>H</sub>	SSC Receive Buffer (read only)	XXXX <sub>H</sub>
<b>SSCRIC</b>	<b>b</b> FF74 <sub>H</sub>	BA <sub>H</sub>	SSC Receive Interrupt Control Register	0000 <sub>H</sub>
<b>SSCTB</b>	F0B0 <sub>H</sub>	<b>E</b> 58 <sub>H</sub>	SSC Transmit Buffer (write only)	0000 <sub>H</sub>
<b>SSCTIC</b>	<b>b</b> FF72 <sub>H</sub>	B9 <sub>H</sub>	SSC Transmit Interrupt Control Register	0000 <sub>H</sub>
<b>STKOV</b>	FE14 <sub>H</sub>	0A <sub>H</sub>	CPU Stack Overflow Pointer Register	FA00 <sub>H</sub>
<b>STKUN</b>	FE16 <sub>H</sub>	0B <sub>H</sub>	CPU Stack Underflow Pointer Register	FC00 <sub>H</sub>
<b>SYSCON</b>	<b>b</b> FF12 <sub>H</sub>	89 <sub>H</sub>	CPU System Configuration Register	0XX0 <sub>H</sub> <sup>1)</sup>
<b>SYSCON2</b>	<b>b</b> F1D0 <sub>H</sub>	<b>E</b> E8 <sub>H</sub>	CPU System Configuration Register 2	0000 <sub>H</sub>
<b>SYSCON3</b>	<b>b</b> F1D4 <sub>H</sub>	<b>E</b> EA <sub>H</sub>	CPU System Configuration Register 3	0000 <sub>H</sub>
<b>T14</b>	F0D2 <sub>H</sub>	<b>E</b> 69 <sub>H</sub>	RTC Timer 14 Register	no
<b>T14REL</b>	F0D0 <sub>H</sub>	<b>E</b> 68 <sub>H</sub>	RTC Timer 14 Reload Register	no
<b>T2</b>	FE40 <sub>H</sub>	20 <sub>H</sub>	GPT1 Timer 2 Register	0000 <sub>H</sub>
<b>T2CON</b>	<b>b</b> FF40 <sub>H</sub>	A0 <sub>H</sub>	GPT1 Timer 2 Control Register	0000 <sub>H</sub>
<b>T2IC</b>	<b>b</b> FF60 <sub>H</sub>	B0 <sub>H</sub>	GPT1 Timer 2 Interrupt Control Register	0000 <sub>H</sub>
<b>T3</b>	FE42 <sub>H</sub>	21 <sub>H</sub>	GPT1 Timer 3 Register	0000 <sub>H</sub>
<b>T3CON</b>	<b>b</b> FF42 <sub>H</sub>	A1 <sub>H</sub>	GPT1 Timer 3 Control Register	0000 <sub>H</sub>
<b>T3IC</b>	<b>b</b> FF62 <sub>H</sub>	B1 <sub>H</sub>	GPT1 Timer 3 Interrupt Control Register	0000 <sub>H</sub>
<b>T4</b>	FE44 <sub>H</sub>	22 <sub>H</sub>	GPT1 Timer 4 Register	0000 <sub>H</sub>
<b>T4CON</b>	<b>b</b> FF44 <sub>H</sub>	A2 <sub>H</sub>	GPT1 Timer 4 Control Register	0000 <sub>H</sub>
<b>T4IC</b>	<b>b</b> FF64 <sub>H</sub>	B2 <sub>H</sub>	GPT1 Timer 4 Interrupt Control Register	0000 <sub>H</sub>
<b>T5</b>	FE46 <sub>H</sub>	23 <sub>H</sub>	GPT2 Timer 5 Register	0000 <sub>H</sub>
<b>T5CON</b>	<b>b</b> FF46 <sub>H</sub>	A3 <sub>H</sub>	GPT2 Timer 5 Control Register	0000 <sub>H</sub>
<b>T5IC</b>	<b>b</b> FF66 <sub>H</sub>	B3 <sub>H</sub>	GPT2 Timer 5 Interrupt Control Register	0000 <sub>H</sub>
<b>T6</b>	FE48 <sub>H</sub>	24 <sub>H</sub>	GPT2 Timer 6 Register	0000 <sub>H</sub>
<b>T6CON</b>	<b>b</b> FF48 <sub>H</sub>	A4 <sub>H</sub>	GPT2 Timer 6 Control Register	0000 <sub>H</sub>
<b>T6IC</b>	<b>b</b> FF68 <sub>H</sub>	B4 <sub>H</sub>	GPT2 Timer 6 Interrupt Control Register	0000 <sub>H</sub>
<b>TFR</b>	<b>b</b> FFAC <sub>H</sub>	D6 <sub>H</sub>	Trap Flag Register	0000 <sub>H</sub>

Name	Physical Address	8-Bit Address	Description	Reset Value
<b>WDT</b>	FEAE <sub>H</sub>	57 <sub>H</sub>	Watchdog Timer Register (read only)	0000 <sub>H</sub>
<b>WDTCON</b>	<b>b</b> FFAE <sub>H</sub>	D7 <sub>H</sub>	Watchdog Timer Control Register	00XX <sub>H</sub> <sup>2)</sup>
<b>XP0IC</b>	<b>b</b> F186 <sub>H</sub>	<b>E</b> C3 <sub>H</sub>	I <sup>2</sup> C Data Interrupt Control Register	0000 <sub>H</sub>
<b>XP1IC</b>	<b>b</b> F18E <sub>H</sub>	<b>E</b> C7 <sub>H</sub>	I <sup>2</sup> C Protocol Interrupt Control Register	0000 <sub>H</sub>
<b>XP2IC</b>	<b>b</b> F196 <sub>H</sub>	<b>E</b> CB <sub>H</sub>	X-Peripheral 2 Interrupt Control Register	0000 <sub>H</sub>
<b>XP3IC</b>	<b>b</b> F19E <sub>H</sub>	<b>E</b> CF <sub>H</sub>	RTC Interrupt Control Register	0000 <sub>H</sub>
<b>ZEROS</b>	<b>b</b> FF1C <sub>H</sub>	8E <sub>H</sub>	Constant Value 0's Register (read only)	0000 <sub>H</sub>

- 1) The system configuration is selected during reset.
- 2) The reset value depends on the indicated reset source.



## Absolute Maximum Ratings

Ambient temperature under bias ( $T_A$ ):

SAB-C161RI ..... 0 to + 70 °C

SAF-C161RI ..... – 40 to + 85 °C

Storage temperature ( $T_{ST}$ )..... – 65 to + 150 °C

Voltage on  $V_{DD}$  pins with respect to ground ( $V_{SS}$ ) ..... – 0.5 to + 6.5 V

Voltage on any pin with respect to ground ( $V_{SS}$ ) ..... – 0.5 to  $V_{DD} + 0.5$  V

Input current on any pin during overload condition ..... – 10 to + 10 mA

Absolute sum of all input currents during overload condition ..... |100 mA|

Power dissipation..... 1.5 W

**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 ( $V_{IN} > V_{DD}$  or  $V_{IN} < V_{SS}$ ) the voltage on  $V_{DD}$  pins with respect to ground ( $V_{SS}$ ) must not exceed the values defined by the absolute maximum ratings.

## Parameter Interpretation

The parameters listed in the following partly represent the characteristics of the C161RI 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 (Controller Characteristics):**

The logic of the C161RI will provide signals with the respective timing characteristics.

**SR (System Requirement):**

The external system must provide signals with the respective timing characteristics to the C161RI.

## DC Characteristics

$V_{DD} = 4.5 - 5.5 \text{ V}; \quad V_{SS} = 0 \text{ V}; \quad f_{CPU} = 20 \text{ MHz}$

$T_A = 0 \text{ to } +70 \text{ }^\circ\text{C}$  for SAB-C161RI

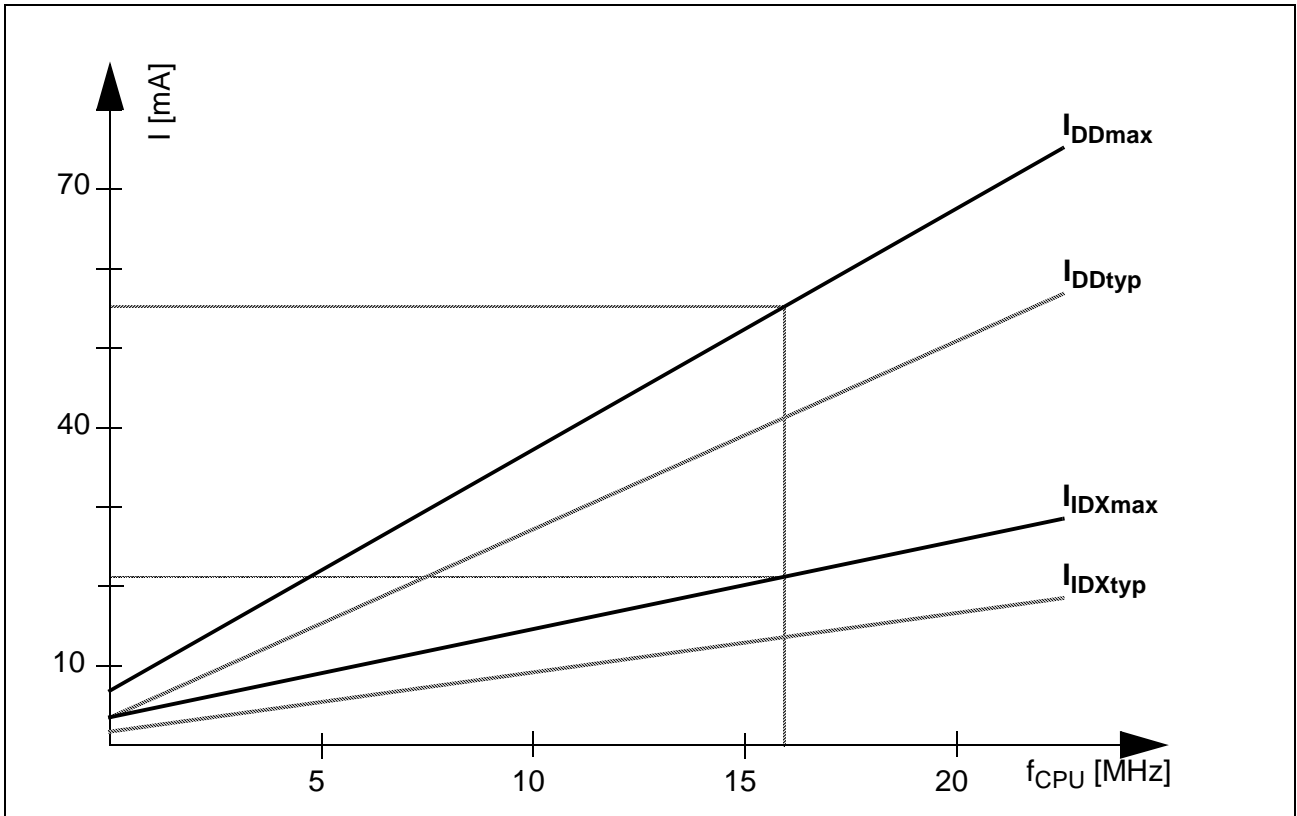
$T_A = -40 \text{ to } +85 \text{ }^\circ\text{C}$  for SAF-C161RI

Parameter	Symbol	Limit Values		Unit	Test Condition
		min.	max.		
Input low voltage (P3.0, P3.1, P6.5, P6.6, P6.7)	$V_{IL1}$ SR	-0.5	$0.3 V_{DD}$	V	-
Input low voltage (TTL)	$V_{IL}$ SR	-0.5	$0.2 V_{DD}$ -0.1	V	-
Input low voltage (Special Threshold)	$V_{ILS}$ SR	-0.5	2.0	V	-
Input high voltage $\overline{RSTIN}$	$V_{IH1}$ SR	$0.6 V_{DD}$	$V_{DD} + 0.5$	V	-
Input high voltage XTAL1, P3.0, P3.1, P6.5, P6.6, P6.7	$V_{IH2}$ SR	$0.7 V_{DD}$	$V_{DD} + 0.5$	V	-
Input high voltage (TTL)	$V_{IH}$ SR	$0.2 V_{DD}$ +0.9	$V_{DD} + 0.5$	V	-
Input high voltage (Special Threshold)	$V_{IHS}$ SR	$0.8 V_{DD}$ -0.2	$V_{DD} + 0.5$	V	-
Input Hysteresis (Special Threshold)	<i>HYS</i>	400	-	mV	-
Output low voltage (PORT0, PORT1, Port 4, $\overline{RD}$ , $\overline{WR}$ , BHE, CLKOUT, $\overline{RSTOUT}$ )	$V_{OL}$ CC	-	0.45	V	$I_{OL} = 2.4 \text{ mA}$
Output low voltage (P3.0, P3.1, P6.5, P6.6, P6.7)	$V_{OL2}$ CC	-	0.4	V	$I_{OL2} = 3 \text{ mA}$
Output low voltage (all other outputs)	$V_{OL1}$ CC	-	0.45	V	$I_{OL1} = 1.6 \text{ mA}$
Output high voltage (PORT0, PORT1, Port 4, $\overline{RD}$ , $\overline{WR}$ , BHE, CLKOUT, $\overline{RSTOUT}$ )	$V_{OH}$ CC	$0.9 V_{DD}$ 2.4	-	V	$I_{OH} = -500 \mu\text{A}$ $I_{OH} = -2.4 \text{ mA}$
Output high voltage <sup>1)</sup> (all other outputs)	$V_{OH1}$ CC	$0.9 V_{DD}$ 2.4	-	V V	$I_{OH} = -250 \mu\text{A}$ $I_{OH} = -1.6 \text{ mA}$
Input leakage current (Port 5)	$I_{OZ1}$ CC	-	$\pm 200$	nA	$0.45 \text{ V} < V_{IN} < V_{DD}$
Input leakage current (all other)	$I_{OZ2}$ CC	-	$\pm 500$	nA	$0.45 \text{ V} < V_{IN} < V_{DD}$
Overload current	$I_{OV}$ SR	-	$\pm 5$	mA	5) 8)
$\overline{RSTIN}$ pullup resistor	$R_{RST}$ CC	50	250	k $\Omega$	-
Read/Write inactive current <sup>4)</sup>	$I_{RWH}$ <sup>2)</sup>	-	-40	$\mu\text{A}$	$V_{OUT} = 2.4 \text{ V}$
Read/Write active current <sup>4)</sup>	$I_{RWL}$ <sup>3)</sup>	-500	-	$\mu\text{A}$	$V_{OUT} = V_{OLmax}$

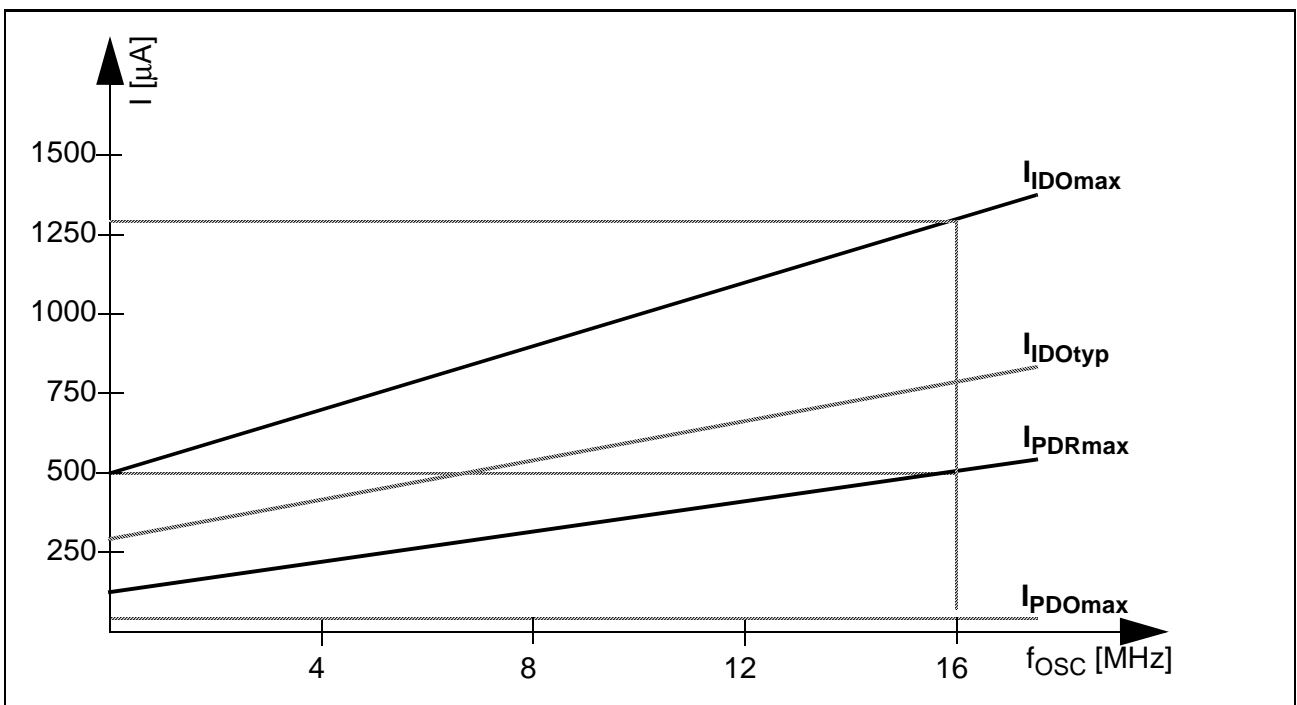
Parameter	Symbol	Limit Values		Unit	Test Condition
		min.	max.		
ALE inactive current <sup>4)</sup>	$I_{ALEL}$ <sup>2)</sup>	–	40	μA	$V_{OUT} = V_{OLmax}$
ALE active current <sup>4)</sup>	$I_{ALEH}$ <sup>3)</sup>	500	–	μA	$V_{OUT} = 2.4 V$
Port 6 inactive current <sup>4)</sup>	$I_{P6H}$ <sup>2)</sup>	–	– 40	μA	$V_{OUT} = 2.4 V$
Port 6 active current <sup>4)</sup>	$I_{P6L}$ <sup>3)</sup>	– 500	–	μA	$V_{OUT} = V_{OL1max}$
PORT0 configuration current <sup>4)</sup>	$I_{POH}$ <sup>2)</sup>	–	– 10	μA	$V_{IN} = V_{IHmin}$
	$I_{POL}$ <sup>3)</sup>	– 100	–	μA	$V_{IN} = V_{ILmax}$
XTAL1 input current	$I_{IL}$ <sub>CC</sub>	–	± 20	μA	$0 V < V_{IN} < V_{DD}$
Pin capacitance <sup>5)</sup> (digital inputs/outputs)	$C_{IO}$ <sub>CC</sub>	–	10	pF	$f = 1 MHz$ $T_A = 25 °C$
Power supply current (active) with all peripherals active	$I_{DD}$	–	7 + $3 \times f_{CPU}$	mA	$\overline{RSTIN} = V_{IL2}$ $f_{CPU}$ in [MHz] <sup>6)</sup>
Idle mode supply current with all peripherals active	$I_{IDX}$	–	3 + $1.1 \times f_{CPU}$	mA	$\overline{RSTIN} = V_{IH1}$ $f_{CPU}$ in [MHz] <sup>6)</sup>
Idle mode supply current with all peripherals deactivated, PLL off, SDD factor = 32	$I_{IDO}$	–	500 + $50 \times f_{OSC}$ <sup>9)</sup>	μA	$\overline{RSTIN} = V_{IH1}$ $f_{OSC}$ in [MHz] <sup>6)</sup>
Power-down mode supply current with RTC running	$I_{PDR}$	–	100 + $25 \times f_{OSC}$ <sup>9)</sup>	μA	$V_{DD} = 5.5 V$ $f_{OSC}$ in [MHz] <sup>7)</sup>
Power-down mode supply current with RTC disabled	$I_{PDO}$	–	50	μA	$V_{DD} = 5.5 V$ <sup>7)</sup>

## Notes

- 1) 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.
- 2) The maximum current may be drawn while the respective signal line remains inactive.
- 3) The minimum current must be drawn in order to drive the respective signal line active.
- 4) This specification is only valid during Reset, or during Adapt-mode.
- 5) Not 100% tested, guaranteed by design characterization.
- 6) The supply current is a function of the operating frequency. This dependency is illustrated in the figure below. These parameters are tested at  $V_{DDmax}$  and 20 MHz CPU clock with all outputs disconnected and all inputs at  $V_{IL}$  or  $V_{IH}$ . The oscillator also contributes to the total supply current. The given values refer to the worst case, i.e.  $I_{PDRmax}$ . For lower oscillator frequencies the respective supply current can be reduced accordingly.
- 7) This parameter is tested including leakage currents. All inputs (including pins configured as inputs) at 0 V to 0.1 V or at  $V_{DD} - 0.1$  V to  $V_{DD}$ ,  $V_{REF} = 0$  V, all outputs (including pins configured as outputs) disconnected.
- 8) Overload conditions under operating conditions occur if the voltage on the respective pin exceeds the specified operating range (i.e.  $V_{OV} > V_{DD} + 0.5$  V or  $V_{OV} < V_{SS} - 0.5$  V). The absolute sum of input overload currents on all port pins may not exceed **50 mA**. The supply voltage ( $V_{DD}$  and  $V_{SS}$ ) must remain within the specified limits.
- 9) This parameter is determined mainly by the current consumed by the oscillator. This current, however, is influenced by the external oscillator circuitry (crystal, capacitors). The values given for  $I_{PDR}$  refer to a typical circuitry and may change in case of a not optimized external oscillator circuitry. A typical value for  $I_{PDR}$  at room temperature and  $f_{CPU} = 16$  MHz is 300  $\mu$ A.



**Figure 8**  
Supply/Idle Current as a Function of Operating Frequency



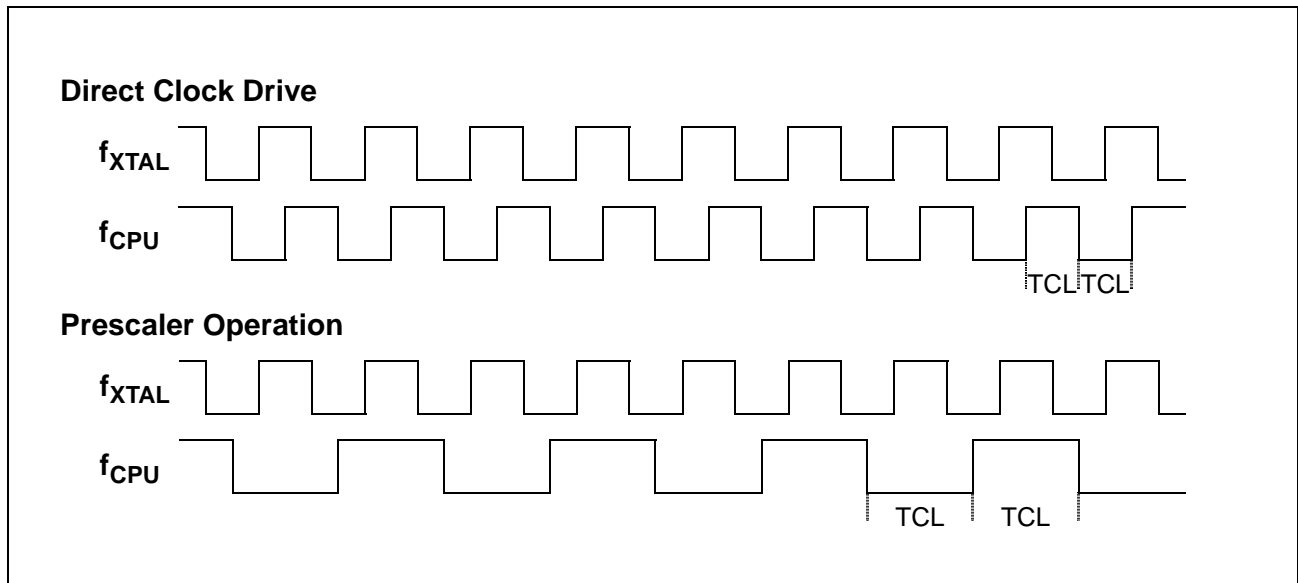
**Figure 9**  
Power Down Supply Current as a Function of Oscillator Frequency

## AC Characteristics

### Definition of Internal Timing

The internal operation of the C161RI is controlled by the internal CPU clock  $f_{CPU}$ . Both edges of the CPU clock can trigger internal (e.g. pipeline) or external (e.g. bus cycles) operations.

The specification of the external timing (AC Characteristics) therefore depends on the time between two consecutive edges of the CPU clock, called "TCL" (see figure below).



**Figure 10**  
**Generation Mechanisms for the CPU Clock**

The CPU clock signal can be generated via different mechanisms. The duration of TCLs and their variation (and also the derived external timing) depends on the used mechanism to generate  $f_{CPU}$ . This influence must be regarded when calculating the timings for the C161RI.

The used mechanism to generate the CPU clock is selected during reset via the logic levels on pins P0.15-13 (P0H.7-5).

The table below associates the combinations of these three bits with the respective clock generation mode.

### C161RI Clock Generation Modes

P0.15-13 (P0H.7-5)	CPU Frequency $f_{\text{CPU}} = f_{\text{XTAL}} \times F$	Notes
1 1 1	Reserved	Default configuration without pull-downs
1 1 0	Reserved	
1 0 1	Reserved	
1 0 0	Reserved	
0 1 1	$f_{\text{XTAL}} \times 1$	Direct drive <sup>1)</sup>
0 1 0	Reserved	
0 0 1	$f_{\text{XTAL}} / 2$	CPU clock via prescaler
0 0 0	Reserved	

<sup>1)</sup> The maximum frequency depends on the duty cycle of the external clock signal.

### Prescaler Operation

When pins P0.15-13 (P0H.7-5) equal '001' during reset the CPU clock is derived from the internal oscillator (input clock signal) by a 2:1 prescaler.

The frequency of  $f_{\text{CPU}}$  is half the frequency of  $f_{\text{XTAL}}$  and the high and low time of  $f_{\text{CPU}}$  (i.e. the duration of an individual TCL) is defined by the period of the input clock  $f_{\text{XTAL}}$ .

The timings listed in the AC Characteristics that refer to TCLs therefore can be calculated using the period of  $f_{\text{XTAL}}$  for any TCL.

### Direct Drive

When pins P0.15-13 (P0H.7-5) equal '011' during reset the on-chip phase locked loop is disabled and the CPU clock is directly driven from the internal oscillator with the input clock signal.

The frequency of  $f_{\text{CPU}}$  directly follows the frequency of  $f_{\text{XTAL}}$  so the high and low time of  $f_{\text{CPU}}$  (i.e. the duration of an individual TCL) is defined by the duty cycle of the input clock  $f_{\text{XTAL}}$ .

The timings listed below that refer to TCLs therefore must be calculated using the minimum TCL that is possible under the respective circumstances. This minimum value can be calculated via the following formula:

$$\text{TCL}_{\text{min}} = 1/f_{\text{XTAL}} \times \text{DC}_{\text{min}} \quad (\text{DC} = \text{duty cycle})$$

For two consecutive TCLs the deviation caused by the duty cycle of  $f_{\text{XTAL}}$  is compensated so the duration of 2TCL is always  $1/f_{\text{XTAL}}$ . The minimum value  $\text{TCL}_{\text{min}}$  therefore has to be used only once for timings that require an odd number of TCLs (1,3,...). Timings that require an even number of TCLs (2,4,...) may use the formula  $2\text{TCL} = 1/f_{\text{XTAL}}$ .

**Note:** The address float timings in Multiplexed bus mode ( $t_{11}$  and  $t_{45}$ ) use the maximum duration of TCL ( $\text{TCL}_{\text{max}} = 1/f_{\text{XTAL}} \times \text{DC}_{\text{max}}$ ) instead of  $\text{TCL}_{\text{min}}$ .

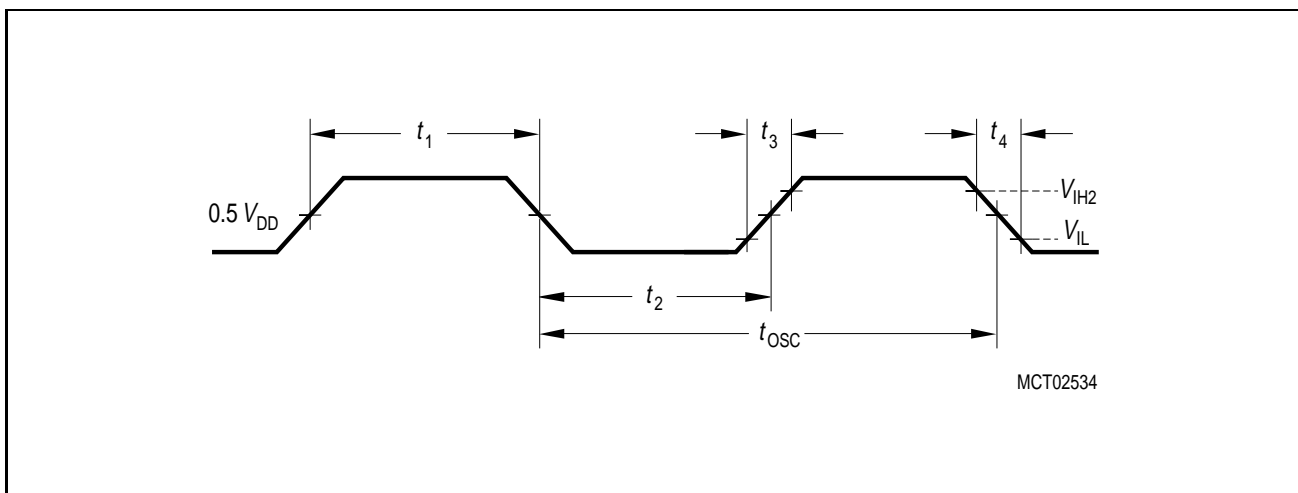
## AC Characteristics

### External Clock Drive XTAL1

$V_{DD} = 4.5 - 5.5 \text{ V}; \quad V_{SS} = 0 \text{ V}$   
 $T_A = 0 \text{ to } +70 \text{ }^\circ\text{C}$  for SAB-C161RI  
 $T_A = -40 \text{ to } +85 \text{ }^\circ\text{C}$  for SAF-C161RI

Parameter	Symbol	Direct Drive 1:1		Prescaler 2:1		Unit
		min.	max.	min.	max.	
Oscillator period	$t_{OSC}$ SR	62	8000	31	4000	ns
High time	$t_1$ SR	25 <sup>1) 2)</sup>	—	6	—	ns
Low time	$t_2$ SR	25 <sup>1) 2)</sup>	—	6	—	ns
Rise time	$t_3$ SR	—	10 <sup>1)</sup>	—	6 <sup>1)</sup>	ns
Fall time	$t_4$ SR	—	10 <sup>1)</sup>	—	6 <sup>1)</sup>	ns

- 1) The clock input signal must reach the defined levels  $V_{IL}$  and  $V_{IH2}$ .
- 2) The specified minimum low and high times allow a duty cycle range of 40...60% at 16 MHz.



**Figure 11**  
**External Clock Drive XTAL1**



## A/D Converter Characteristics

$V_{DD} = 4.5 - 5.5 \text{ V}; \quad V_{SS} = 0 \text{ V}$   
 $T_A = 0 \text{ to } +70 \text{ }^\circ\text{C}$  for SAB-C161RI  
 $T_A = -40 \text{ to } +85 \text{ }^\circ\text{C}$  for SAF-C161RI  
 $4.0 \text{ V} \leq V_{AREF} \leq V_{DD} + 0.1 \text{ V}; \quad V_{SS} - 0.1 \text{ V} \leq V_{AGND} \leq V_{SS} + 0.2 \text{ V}$

Parameter	Symbol	Limit Values		Unit	Test Condition
		min.	max.		
Analog input voltage range	$V_{AIN}$ SR	$V_{AGND}$	$V_{AREF}$	V	1)
Basic clock frequency	$f_{BC}$	0.5	4	MHz	2)
Sample time	$t_S$ CC	–	$6 t_{BC}$		$t_{BC} = 1 / f_{BC}$
Conversion time	$t_C$ CC	–	$30 t_{BC} + 2 t_{CPU}$		3) $t_{CPU} = 1 / f_{CPU}$
Total unadjusted error	TUE CC	–	$\pm 2$	LSB	4)
Internal resistance of reference voltage source	$R_{AREF}$ SR	–	$t_{BC} / 125 - 0.25$	k $\Omega$	$t_{BC}$ in [ns] <sup>5) 6)</sup>
Internal resistance of analog source	$R_{ASRC}$ SR	–	$t_S / 750 - 0.25$	k $\Omega$	$t_S$ in [ns] <sup>6) 7)</sup>
ADC input capacitance	$C_{AIN}$ CC	–	50	pF	6)

The conversion time of the C161RI's A/D Converter is programmable. The table below should be used to calculate the above timings.

The limit values for  $f_{BC}$  must not be exceeded when selecting ADCTC.

ADCON.15 14 (ADCTC)	A/D Converter Basic Clock $f_{BC}$ <sup>2)</sup>
00	$f_{CPU} / 2$
01	$f_{CPU} / 4$
10	$f_{CPU} / 8$
11	$f_{CPU} / 16$

### Converter Timing Example:

Assumptions:  $f_{CPU} = 16 \text{ MHz}$  (i.e.  $t_{CPU} = 62.5 \text{ ns}$ ), ADCTC = '01'.

Basic clock  $f_{BC} = f_{CPU} / 4 = 4 \text{ MHz}$ , i.e.  $t_{BC} = 250 \text{ ns}$ .

Sample time  $t_S = t_{BC} \times 6 = 1500 \text{ ns}$ .

Conversion time  $t_C = 30 t_{BC} + 2 t_{CPU} = (7500 + 125) \text{ ns} = 7.625 \mu\text{s}$ .

**Notes**

- 1)  $V_{AIN}$  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.
- 2) The limit values for  $f_{BC}$  must not be exceeded when selecting the CPU frequency and the ADCTC setting.
- 3) 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 the table above.  
This parameter depends on the ADC control logic. It is not a real maximum value, but rather a fixum.
- 4) TUE is tested at  $V_{AREF} = 5.0\text{ V}$ ,  $V_{AGND} = 0\text{ V}$ ,  $V_{DD} = 4.9\text{ V}$ . It is guaranteed by design for all other voltages within the defined voltage range.  
The specified TUE is guaranteed only if an overload condition (see  $I_{OV}$  specification) occurs on maximum 2 not selected analog input pins and the absolute sum of input overload currents on all analog input pins does not exceed 10 mA.
- 5) During the conversion the ADC's capacitance must be repeatedly charged or discharged. The internal resistance of the reference voltage source must allow the capacitance to reach its respective voltage level within each conversion step. The maximum internal resistance results from the programmed conversion timing.
- 6) Not 100% tested, guaranteed by design.
- 7) During the sample time the input capacitance  $C_I$  can be charged/discharged by the external source. The internal resistance of the analog source must allow the capacitance to reach its final voltage level within  $t_S$ . After the end of the sample time  $t_S$ , changes of the analog input voltage have no effect on the conversion result. Values for the sample time  $t_S$  depend on programming and can be taken from the table above.

Testing Waveforms

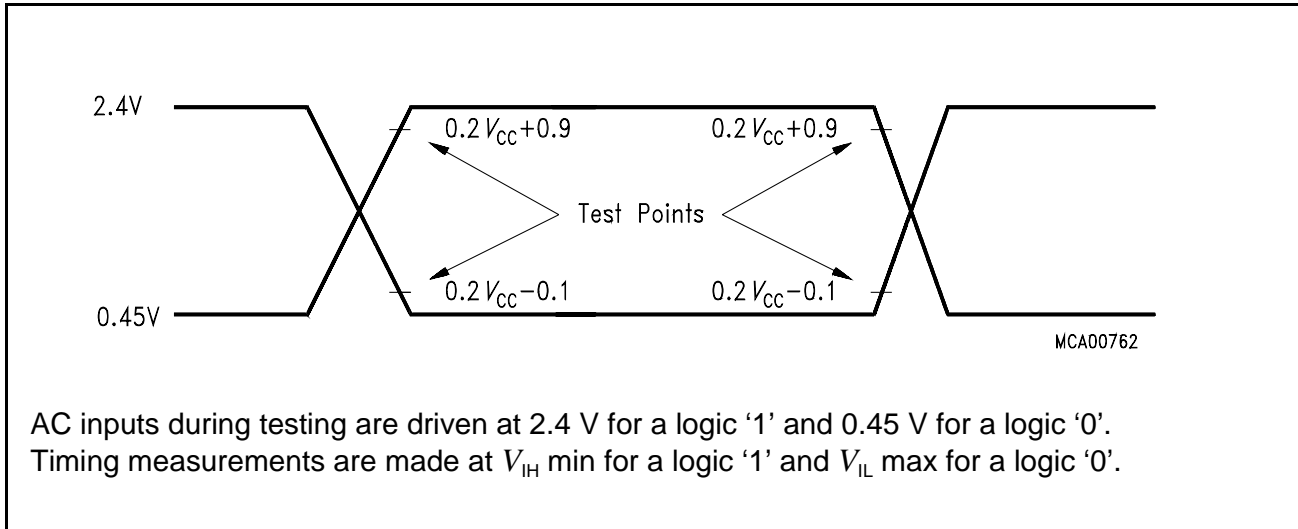


Figure 12  
Input Output Waveforms

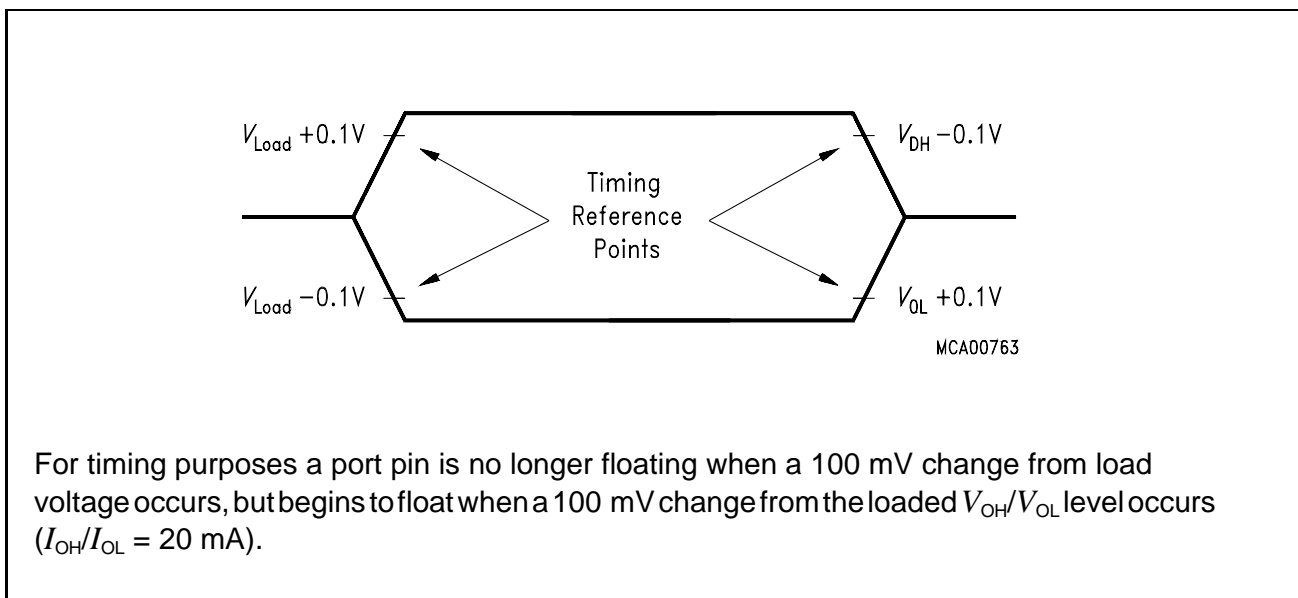


Figure 13  
Float Waveforms

## Memory Cycle Variables

The timing tables below use three variables which are derived from the BUSCONx registers and represent the special characteristics of the programmed memory cycle. The following table describes, how these variables are to be computed.

Description	Symbol	Values
ALE Extension	$t_A$	$TCL \times \langle ALECTL \rangle$
Memory Cycle Time Waitstates	$t_C$	$2TCL \times (15 - \langle MCTC \rangle)$
Memory Tristate Time	$t_F$	$2TCL \times (1 - \langle MTTC \rangle)$

## AC Characteristics

### Multiplexed Bus

$V_{DD} = 4.5 - 5.5 \text{ V}; \quad V_{SS} = 0 \text{ V}$

$T_A = 0 \text{ to } +70 \text{ }^\circ\text{C}$  for SAB-C161RI

$T_A = -40 \text{ to } +85 \text{ }^\circ\text{C}$  for SAF-C161RI

$C_L = 100 \text{ pF}$

ALE cycle time =  $6 \text{ TCL} + 2t_A + t_C + t_F$  (186 ns at 16 MHz CPU clock without waitstates)

Parameter	Symbol	Max. CPU Clock = 16 MHz	Variable CPU Clock 1/2TCL = 1 to 16 MHz		Unit	
			min.	max.		min.
ALE high time	$t_5$ CC	$21 + t_A$	–	$TCL - 10 + t_A$	–	ns
Address setup to ALE	$t_6$ CC	$15 + t_A$	–	$TCL - 16 + t_A$	–	ns
Address hold after ALE	$t_7$ CC	$21 + t_A$	–	$TCL - 10 + t_A$	–	ns
ALE falling edge to $\overline{RD}$ , $\overline{WR}$ (with RW-delay)	$t_8$ CC	$21 + t_A$	–	$TCL - 10 + t_A$	–	ns
ALE falling edge to $\overline{RD}$ , $\overline{WR}$ (no RW-delay)	$t_9$ CC	$-10 + t_A$	–	$-10 + t_A$	–	ns
Address float after $\overline{RD}$ , $\overline{WR}$ (with RW-delay)	$t_{10}$ CC	–	6	–	6	ns
Address float after $\overline{RD}$ , $\overline{WR}$ (no RW-delay)	$t_{11}$ CC	–	37	–	$TCL + 6$	ns
$\overline{RD}$ , $\overline{WR}$ low time (with RW-delay)	$t_{12}$ CC	$53 + t_C$	–	$2TCL - 10 + t_C$	–	ns
$\overline{RD}$ , $\overline{WR}$ low time (no RW-delay)	$t_{13}$ CC	$84 + t_C$	–	$3TCL - 10 + t_C$	–	ns

Parameter	Symbol	Max. CPU Clock = 16 MHz		Variable CPU Clock 1/2TCL = 1 to 16 MHz		Unit
		min.	max.	min.	max.	
$\overline{RD}$ to valid data in (with RW-delay)	$t_{14}$ SR	–	$43 + t_C$	–	$2TCL - 20 + t_C$	ns
$\overline{RD}$ to valid data in (no RW-delay)	$t_{15}$ SR	–	$74 + t_C$	–	$3TCL - 20 + t_C$	ns
ALE low to valid data in	$t_{16}$ SR	–	$74 + t_A + t_C$	–	$3TCL - 20 + t_A + t_C$	ns
Address to valid data in	$t_{17}$ SR	–	$95 + 2t_A + t_C$	–	$4TCL - 30 + 2t_A + t_C$	ns
Data hold after $\overline{RD}$ rising edge	$t_{18}$ SR	0	–	0	–	ns
Data float after $\overline{RD}$	$t_{19}$ SR	–	$49 + t_F$	–	$2TCL - 14 + t_F$	ns
Data valid to $\overline{WR}$	$t_{22}$ CC	$43 + t_C$	–	$2TCL - 20 + t_C$	–	ns
Data hold after $\overline{WR}$	$t_{23}$ CC	$49 + t_F$	–	$2TCL - 14 + t_F$	–	ns
ALE rising edge after $\overline{RD}$ , $\overline{WR}$	$t_{25}$ CC	$49 + t_F$	–	$2TCL - 14 + t_F$	–	ns
Address hold after $\overline{RD}$ , $\overline{WR}$	$t_{27}$ CC	$49 + t_F$	–	$2TCL - 14 + t_F$	–	ns
ALE falling edge to $\overline{CS}$	$t_{38}$ CC	$-4 - t_A$	$10 - t_A$	$-4 - t_A$	$10 - t_A$	ns
$\overline{CS}$ low to Valid Data In	$t_{39}$ SR	–	$74 + t_C + 2t_A$	–	$3TCL - 20 + t_C + 2t_A$	ns
$\overline{CS}$ hold after $\overline{RD}$ , $\overline{WR}$	$t_{40}$ CC	$80 + t_F$	–	$3TCL - 14 + t_F$	–	ns
ALE fall. edge to $\overline{RdCS}$ , $\overline{WrCS}$ (with RW delay)	$t_{42}$ CC	$27 + t_A$	–	$TCL - 4 + t_A$	–	ns
ALE fall. edge to $\overline{RdCS}$ , $\overline{WrCS}$ (no RW delay)	$t_{43}$ CC	$-4 + t_A$	–	$-4 + t_A$	–	ns
Address float after $\overline{RdCS}$ , $\overline{WrCS}$ (with RW delay)	$t_{44}$ CC	–	0	–	0	ns
Address float after $\overline{RdCS}$ , $\overline{WrCS}$ (no RW delay)	$t_{45}$ CC	–	31	–	TCL	ns
$\overline{RdCS}$ to Valid Data In (with RW delay)	$t_{46}$ SR	–	$39 + t_C$	–	$2TCL - 24 + t_C$	ns

Parameter	Symbol	Max. CPU Clock = 16 MHz		Variable CPU Clock 1/2TCL = 1 to 16 MHz		Unit
		min.	max.	min.	max.	
RdCS to Valid Data In (no RW delay)	$t_{47}$ SR	–	$70 + t_C$	–	$3TCL - 24 + t_C$	ns
RdCS, WrCS Low Time (with RW delay)	$t_{48}$ CC	$53 + t_C$	–	$2TCL - 10 + t_C$	–	ns
RdCS, WrCS Low Time (no RW delay)	$t_{49}$ CC	$84 + t_C$	–	$3TCL - 10 + t_C$	–	ns
Data valid to $\overline{WrCS}$	$t_{50}$ CC	$49 + t_C$	–	$2TCL - 14 + t_C$	–	ns
Data hold after $\overline{RdCS}$	$t_{51}$ SR	0	–	0	–	ns
Data float after $\overline{RdCS}$	$t_{52}$ SR	–	$43 + t_F$	–	$2TCL - 20 + t_F$	ns
Address hold after $\overline{RdCS}$ , $\overline{WrCS}$	$t_{54}$ CC	$43 + t_F$	–	$2TCL - 20 + t_F$	–	ns
Data hold after $\overline{WrCS}$	$t_{56}$ CC	$43 + t_F$	–	$2TCL - 20 + t_F$	–	ns

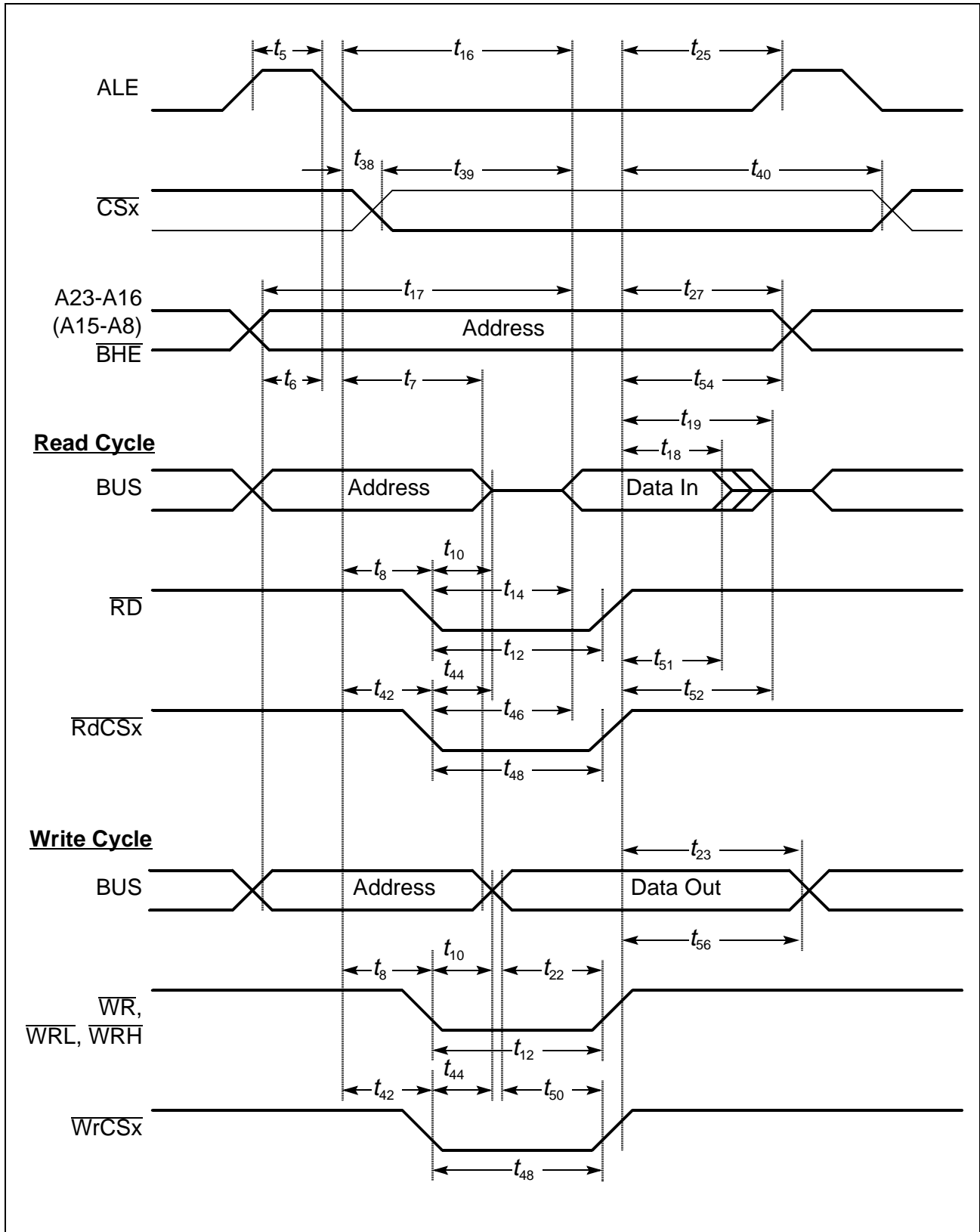


Figure 14-1  
External Memory Cycle: Multiplexed Bus, With Read/Write Delay, Normal ALE

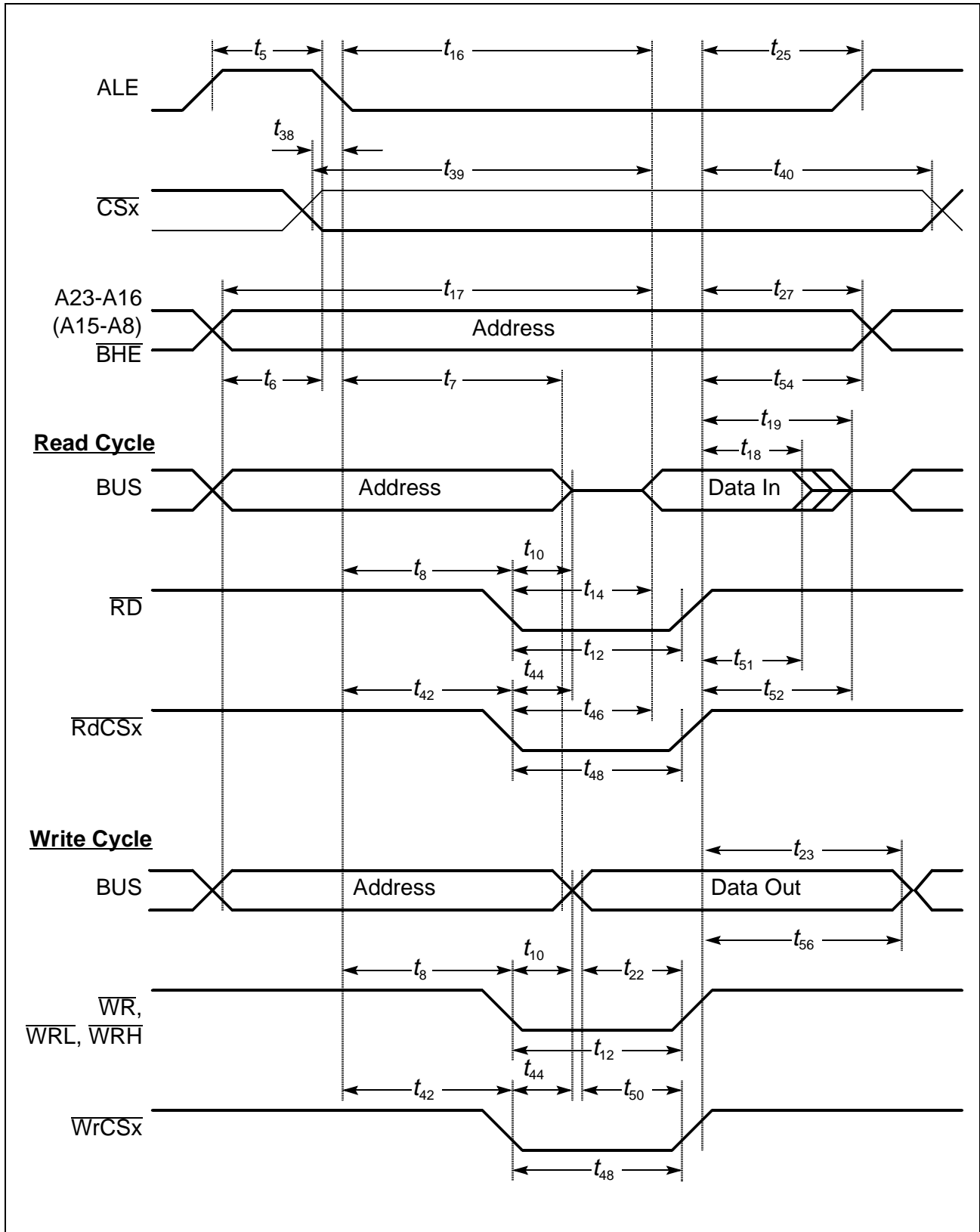
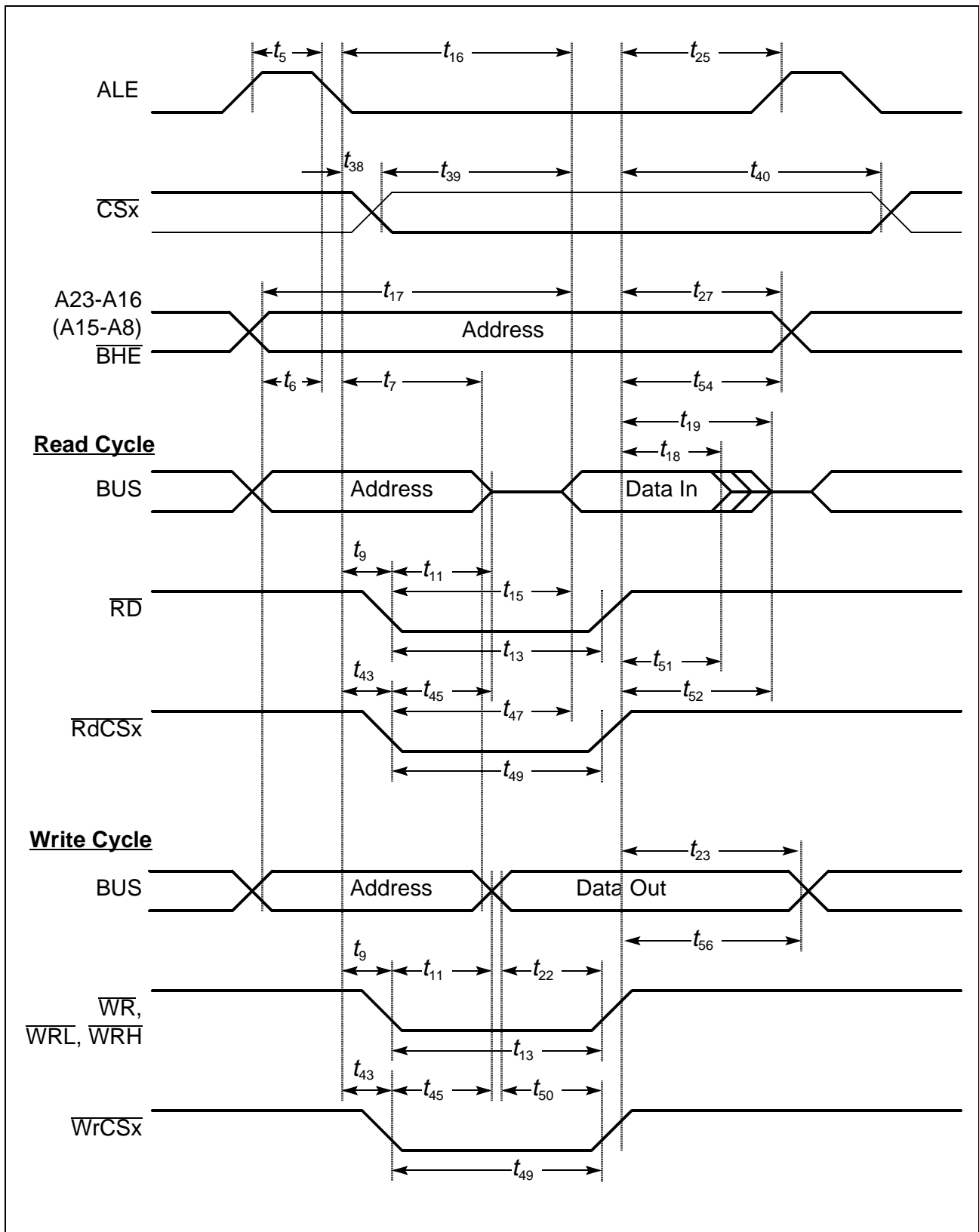


Figure 14-2  
External Memory Cycle: Multiplexed Bus, With Read/Write Delay, Extended ALE





**Figure 14-3**  
**External Memory Cycle: Multiplexed Bus, No Read/Write Delay, Normal ALE**

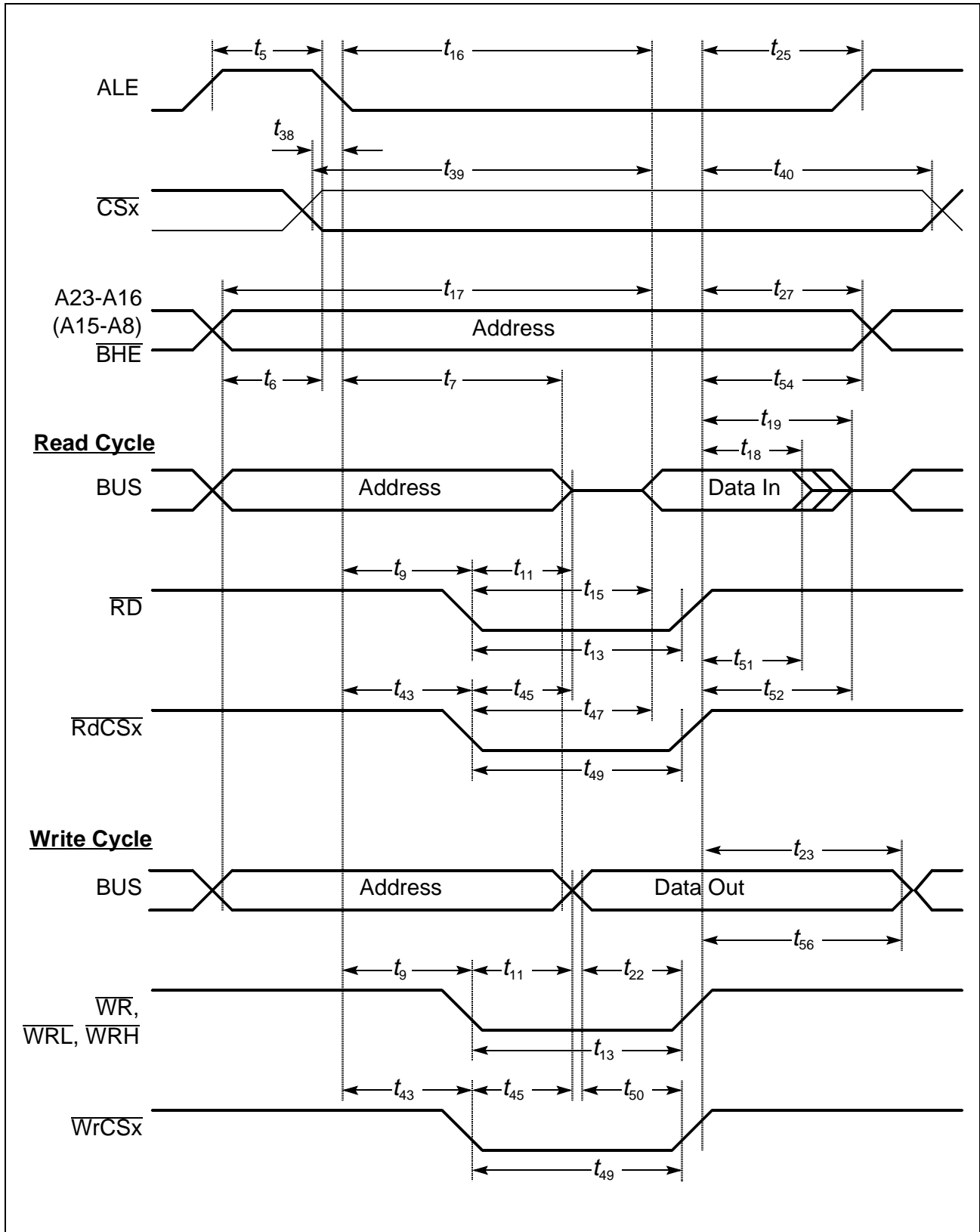


Figure 14-4  
External Memory Cycle: Multiplexed Bus, No Read/Write Delay, Extended ALE

## AC Characteristics Demultiplexed Bus

$V_{DD} = 4.5 - 5.5 \text{ V}; \quad V_{SS} = 0 \text{ V}$

$T_A = 0 \text{ to } +70 \text{ }^\circ\text{C}$  for SAB-C161RI

$T_A = -40 \text{ to } +85 \text{ }^\circ\text{C}$  for SAF-C161RI

$C_L = 100 \text{ pF}$

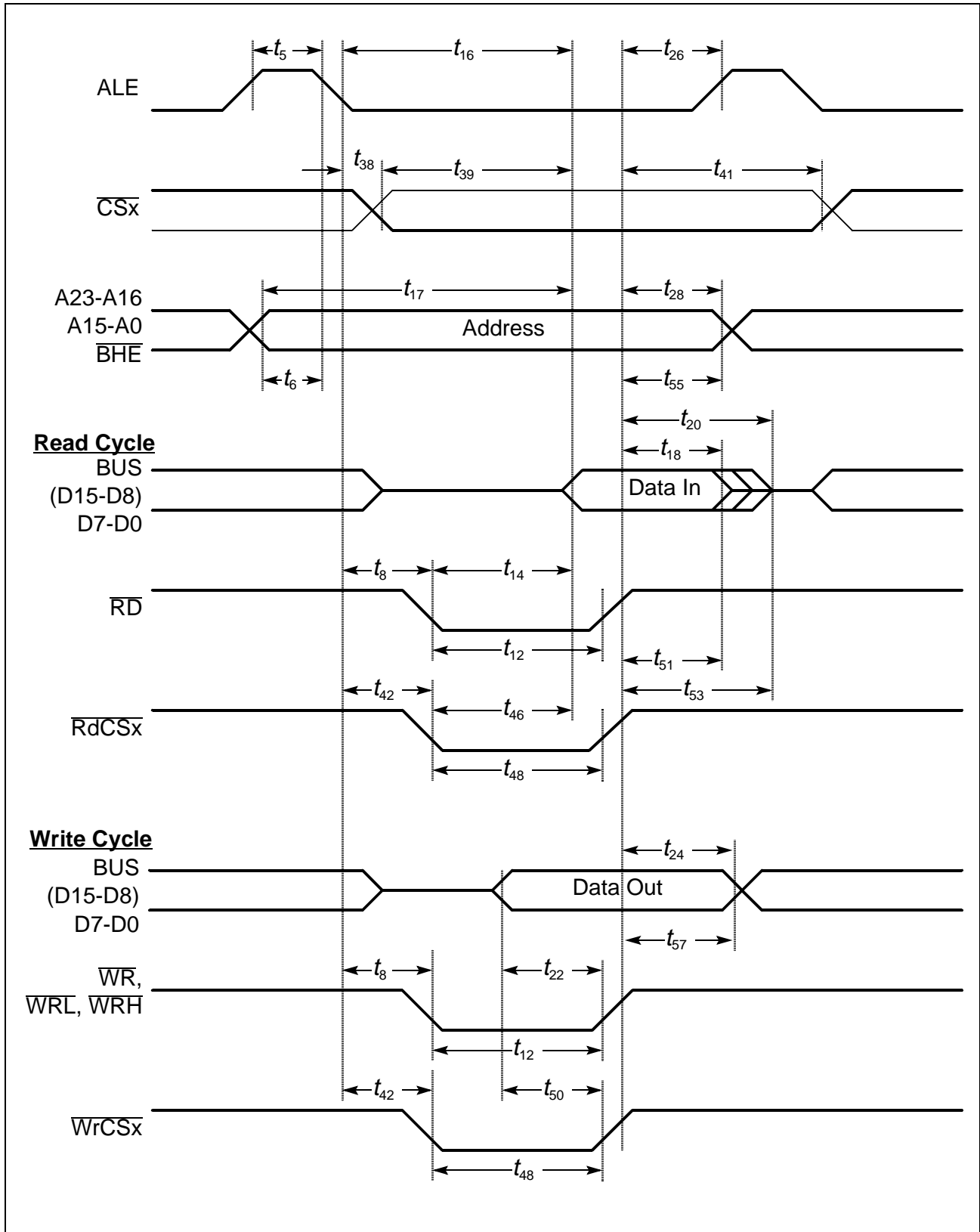
ALE cycle time =  $4 \text{ TCL} + 2t_A + t_C + t_F$  (125 ns at 16 MHz CPU clock without waitstates)

Parameter	Symbol	Max. CPU Clock = 16 MHz		Variable CPU Clock 1/2TCL = 1 to 16 MHz		Unit
		min.	max.	min.	max.	
ALE high time	$t_5$ CC	$21 + t_A$	–	$\text{TCL} - 10 + t_A$	–	ns
Address setup to ALE	$t_6$ CC	$15 + t_A$	–	$\text{TCL} - 16 + t_A$	–	ns
ALE falling edge to $\overline{\text{RD}}$ , $\overline{\text{WR}}$ (with RW-delay)	$t_8$ CC	$21 + t_A$	–	$\text{TCL} - 10 + t_A$	–	ns
ALE falling edge to $\overline{\text{RD}}$ , $\overline{\text{WR}}$ (no RW-delay)	$t_9$ CC	$-10 + t_A$	–	$-10 + t_A$	–	ns
$\overline{\text{RD}}$ , $\overline{\text{WR}}$ low time (with RW-delay)	$t_{12}$ CC	$53 + t_C$	–	$2\text{TCL} - 10 + t_C$	–	ns
$\overline{\text{RD}}$ , $\overline{\text{WR}}$ low time (no RW-delay)	$t_{13}$ CC	$84 + t_C$	–	$3\text{TCL} - 10 + t_C$	–	ns
$\overline{\text{RD}}$ to valid data in (with RW-delay)	$t_{14}$ SR	–	$43 + t_C$	–	$2\text{TCL} - 20 + t_C$	ns
$\overline{\text{RD}}$ to valid data in (no RW-delay)	$t_{15}$ SR	–	$74 + t_C$	–	$3\text{TCL} - 20 + t_C$	ns
ALE low to valid data in	$t_{16}$ SR	–	$74 + t_A + t_C$	–	$3\text{TCL} - 20 + t_A + t_C$	ns
Address to valid data in	$t_{17}$ SR	–	$95 + 2t_A + t_C$	–	$4\text{TCL} - 30 + 2t_A + t_C$	ns
Data hold after $\overline{\text{RD}}$ rising edge	$t_{18}$ SR	0	–	0	–	ns
Data float after $\overline{\text{RD}}$ rising edge (with RW-delay <sup>1)</sup> )	$t_{20}$ SR	–	$49 + 2t_A + t_F$	–	$2\text{TCL} - 14 + 2t_A + t_F$	ns
Data float after $\overline{\text{RD}}$ rising edge (no RW-delay <sup>1)</sup> )	$t_{21}$ SR	–	$21 + 2t_A + t_F$	–	$\text{TCL} - 10 + 2t_A + t_F$	ns
Data valid to $\overline{\text{WR}}$	$t_{22}$ CC	$43 + t_C$	–	$2\text{TCL} - 20 + t_C$	–	ns
Data hold after $\overline{\text{WR}}$	$t_{24}$ CC	$21 + t_F$	–	$\text{TCL} - 10 + t_F$	–	ns
ALE rising edge after $\overline{\text{RD}}$ , $\overline{\text{WR}}$	$t_{26}$ CC	$-10 + t_F$	–	$-10 + t_F$	–	ns

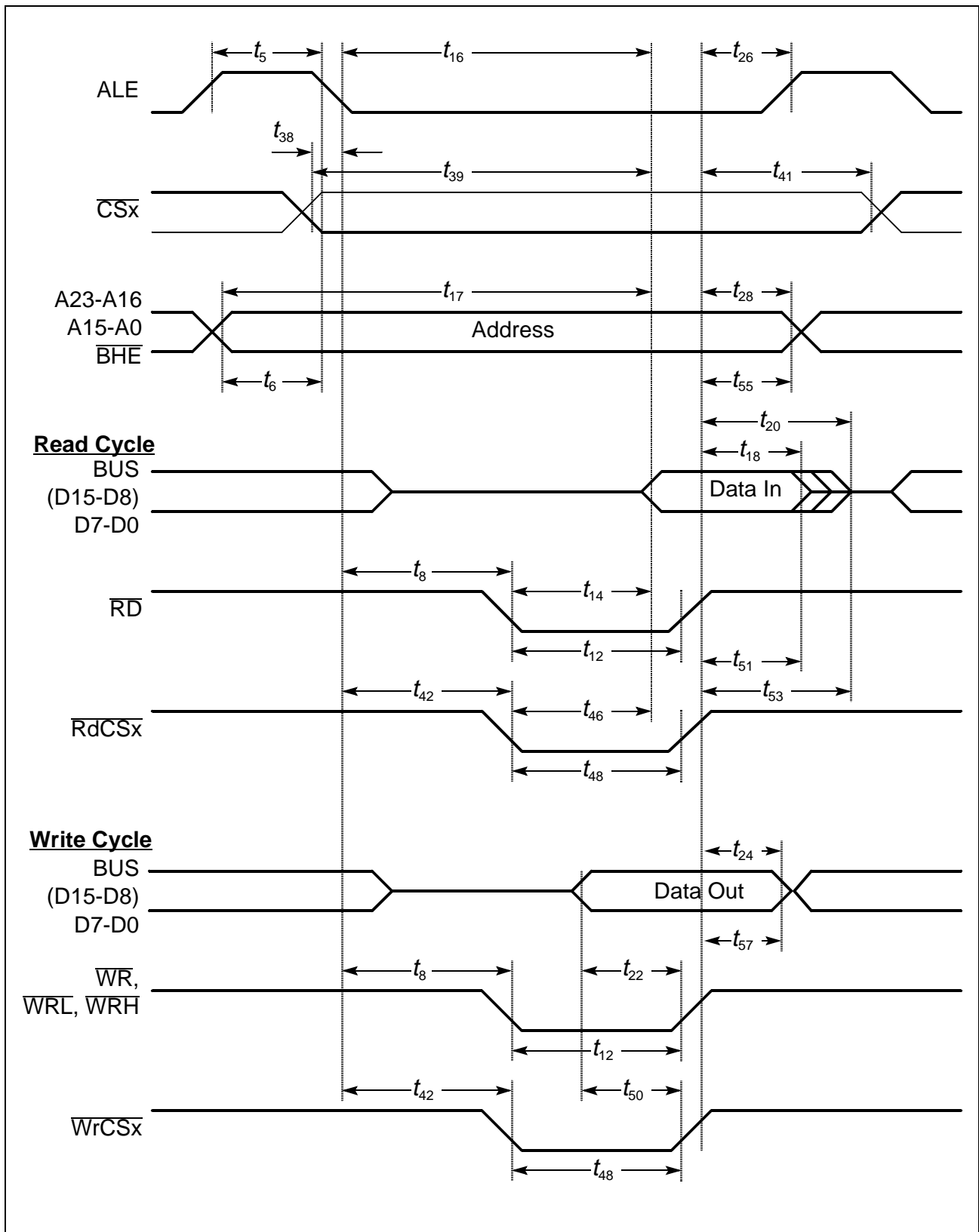
Parameter	Symbol	Max. CPU Clock = 16 MHz		Variable CPU Clock 1/2TCL = 1 to 16 MHz		Unit
		min.	max.	min.	max.	
Address hold after $\overline{WR}$ <sup>2)</sup>	$t_{28}$ CC	$0 + t_F$	–	$0 + t_F$	–	ns
ALE falling edge to $\overline{CS}$	$t_{38}$ CC	$-4 - t_A$	$10 - t_A$	$-4 - t_A$	$10 - t_A$	ns
$\overline{CS}$ low to Valid Data In	$t_{39}$ SR	–	$74 + t_C + 2t_A$	–	$3TCL - 20 + t_C + 2t_A$	ns
$\overline{CS}$ hold after $\overline{RD}$ , $\overline{WR}$	$t_{41}$ CC	$17 + t_F$	–	$TCL - 14 + t_F$	–	ns
ALE falling edge to $\overline{RdCS}$ , $\overline{WrCS}$ (with RW-delay)	$t_{42}$ CC	$27 + t_A$	–	$TCL - 4 + t_A$	–	ns
ALE falling edge to $\overline{RdCS}$ , $\overline{WrCS}$ (no RW-delay)	$t_{43}$ CC	$-4 + t_A$	–	$-4 + t_A$	–	ns
$\overline{RdCS}$ to Valid Data In (with RW-delay)	$t_{46}$ SR	–	$39 + t_C$	–	$2TCL - 24 + t_C$	ns
$\overline{RdCS}$ to Valid Data In (no RW-delay)	$t_{47}$ SR	–	$70 + t_C$	–	$3TCL - 24 + t_C$	ns
$\overline{RdCS}$ , $\overline{WrCS}$ Low Time (with RW-delay)	$t_{48}$ CC	$53 + t_C$	–	$2TCL - 10 + t_C$	–	ns
$\overline{RdCS}$ , $\overline{WrCS}$ Low Time (no RW-delay)	$t_{49}$ CC	$84 + t_C$	–	$3TCL - 10 + t_C$	–	ns
Data valid to $\overline{WrCS}$	$t_{50}$ CC	$49 + t_C$	–	$2TCL - 14 + t_C$	–	ns
Data hold after $\overline{RdCS}$	$t_{51}$ SR	0	–	0	–	ns
Data float after $\overline{RdCS}$ (with RW-delay)	$t_{53}$ SR	–	$43 + t_F$	–	$2TCL - 20 + t_F$	ns
Data float after $\overline{RdCS}$ (no RW-delay)	$t_{68}$ SR	–	$11 + t_F$	–	$TCL - 20 + t_F$	ns
Address hold after $\overline{RdCS}$ , $\overline{WrCS}$	$t_{55}$ CC	$-10 + t_F$	–	$-10 + t_F$	–	ns
Data hold after $\overline{WrCS}$	$t_{57}$ CC	$17 + t_F$	–	$TCL - 14 + t_F$	–	ns

<sup>1)</sup> RW-delay and  $t_A$  refer to the next following bus cycle.

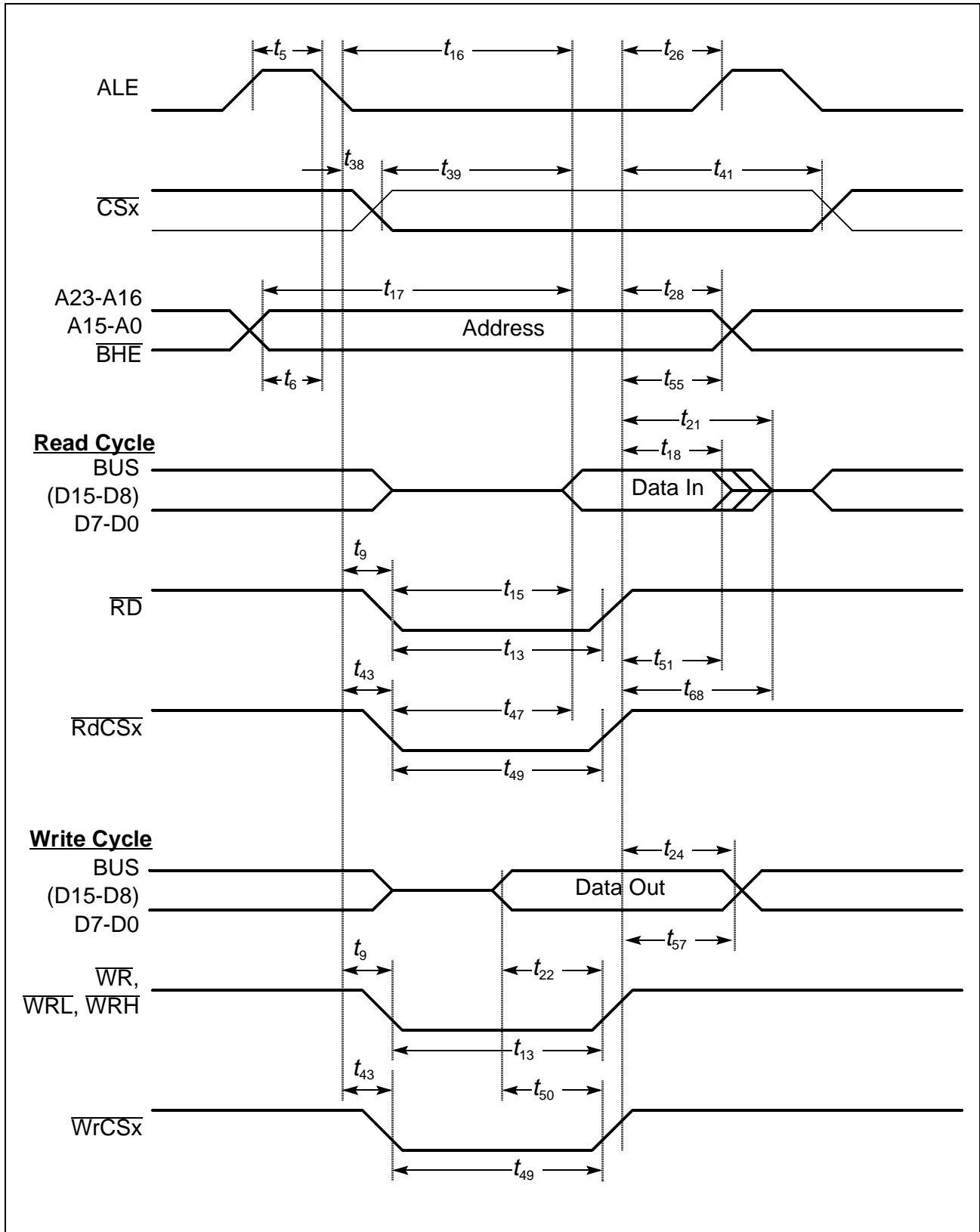
<sup>2)</sup> It is guaranteed by design that read data are latched before the address changes.



**Figure 15-1**  
**External Memory Cycle: Demultiplexed Bus, With Read/Write Delay, Normal ALE**



**Figure 15-2**  
**External Memory Cycle: Demultiplexed Bus, With Read/Write Delay, Extended ALE**



**Figure 15-3**  
**External Memory Cycle: Demultiplexed Bus, No Read/Write Delay, Normal ALE**

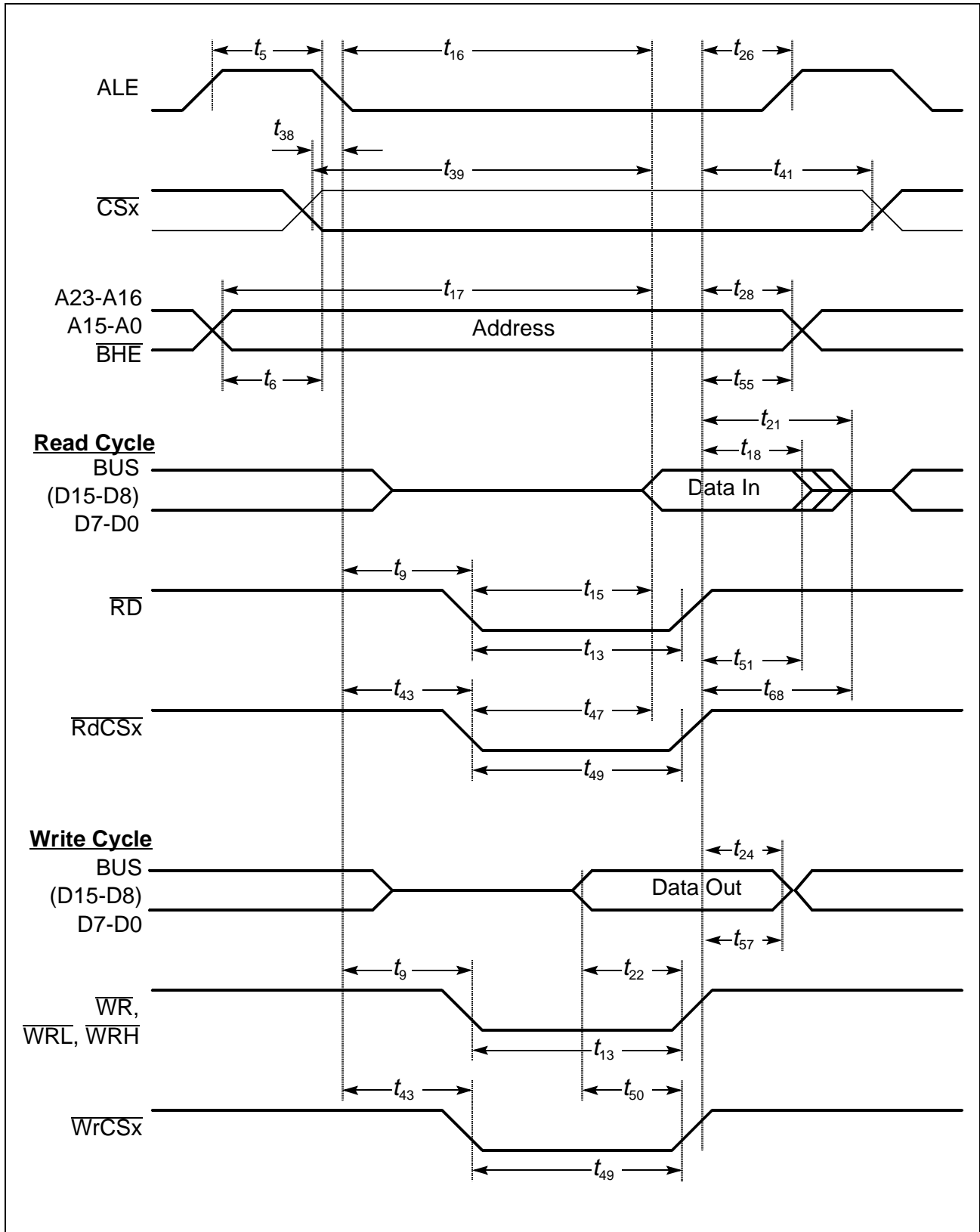


Figure 15-4  
External Memory Cycle: Demultiplexed Bus, No Read/Write Delay, Extended ALE



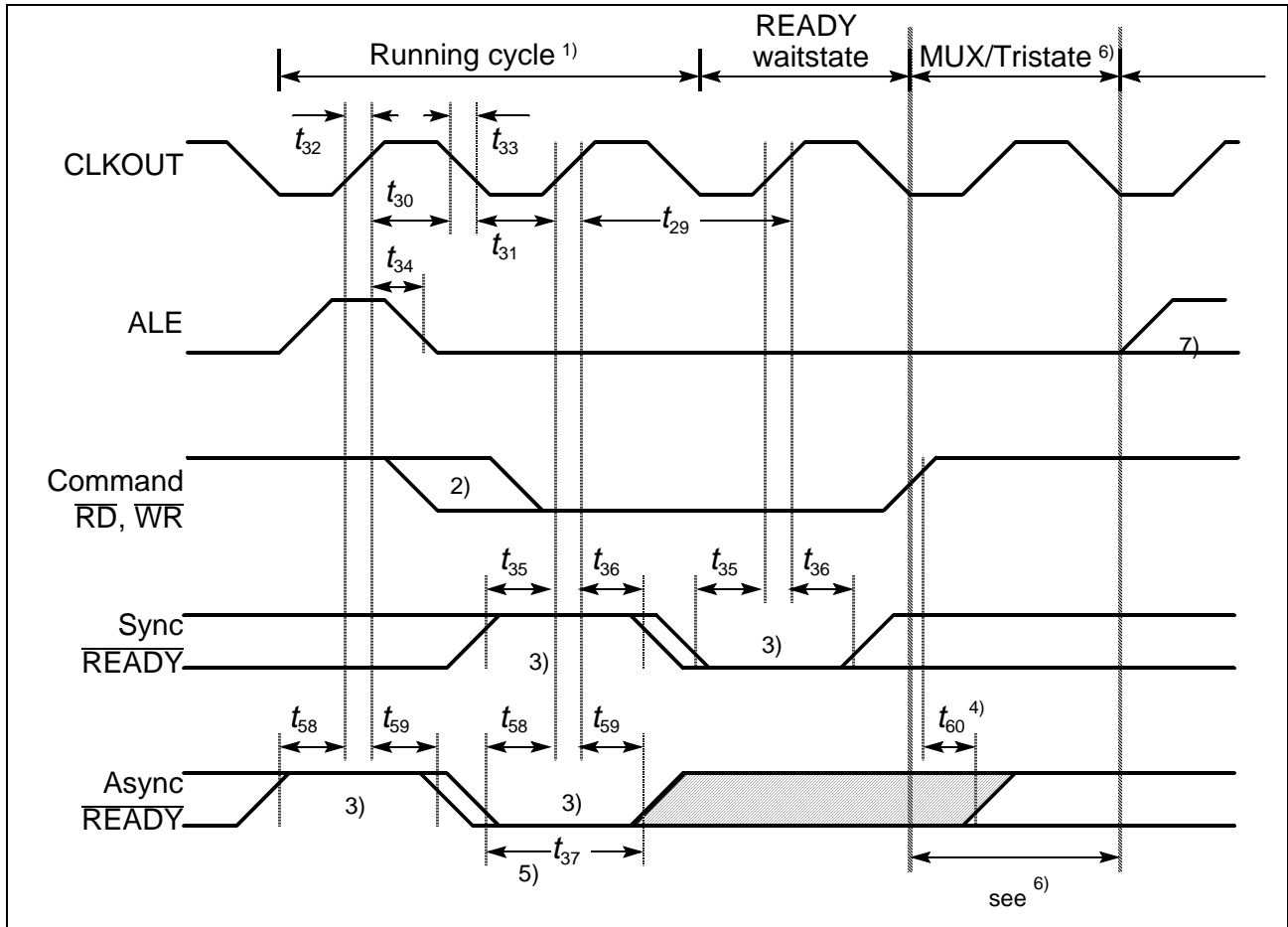
## AC Characteristics CLKOUT and $\overline{\text{READY}}$

$V_{DD} = 4.5 - 5.5 \text{ V}; \quad V_{SS} = 0 \text{ V}$   
 $T_A = 0 \text{ to } +70 \text{ }^\circ\text{C}$  for SAB-C161RI  
 $T_A = -40 \text{ to } +85 \text{ }^\circ\text{C}$  for SAF-C161RI  
 $C_L = 100 \text{ pF}$

Parameter	Symbol	Max. CPU Clock = 16 MHz		Variable CPU Clock 1/2TCL = 1 to 16 MHz		Unit
		min.	max.	min.	max.	
CLKOUT cycle time	$t_{29}$ CC	62	62	2TCL	2TCL	ns
CLKOUT high time	$t_{30}$ CC	25	–	TCL – 6	–	ns
CLKOUT low time	$t_{31}$ CC	21	–	TCL – 10	–	ns
CLKOUT rise time	$t_{32}$ CC	–	4	–	4	ns
CLKOUT fall time	$t_{33}$ CC	–	4	–	4	ns
CLKOUT rising edge to ALE falling edge	$t_{34}$ CC	$0 + t_A$	$10 + t_A$	$0 + t_A$	$10 + t_A$	ns
Synchronous $\overline{\text{READY}}$ setup time to CLKOUT	$t_{35}$ SR	14	–	14	–	ns
Synchronous $\overline{\text{READY}}$ hold time after CLKOUT	$t_{36}$ SR	4	–	4	–	ns
Asynchronous $\overline{\text{READY}}$ low time	$t_{37}$ SR	76	–	2TCL + 14	–	ns
Asynchronous $\overline{\text{READY}}$ setup time <sup>1)</sup>	$t_{58}$ SR	14	–	14	–	ns
Asynchronous $\overline{\text{READY}}$ hold time <sup>1)</sup>	$t_{59}$ SR	4	–	4	–	ns
Async. $\overline{\text{READY}}$ hold time after $\overline{\text{RD}}$ , $\overline{\text{WR}}$ high (Demultiplexed Bus) <sup>2)</sup>	$t_{60}$ SR	0	$1 + 2t_A + t_C + t_F$ <sup>2)</sup>	0	$\text{TCL} - 30 + 2t_A + t_C + t_F$ <sup>2)</sup>	ns

### Notes

- 1) These timings are given for test purposes only, in order to assure recognition at a specific clock edge.
- 2) Demultiplexed bus is the worst case. For multiplexed bus 2TCL are to be added to the maximum values. This adds even more time for deactivating  $\overline{\text{READY}}$ .  
The  $2t_A$  and  $t_C$  refer to the next following bus cycle,  $t_F$  refers to the current bus cycle.



**Figure 16**  
**CLKOUT and READY**

**Notes**

- 1) Cycle as programmed, including MCTC waitstates (Example shows 0 MCTC WS).
- 2) The leading edge of the respective command depends on RW-delay.
- 3)  $\overline{\text{READY}}$  sampled HIGH at this sampling point generates a READY controlled waitstate,  $\overline{\text{READY}}$  sampled LOW at this sampling point terminates the currently running bus cycle.
- 4)  $\overline{\text{READY}}$  may be deactivated in response to the trailing (rising) edge of the corresponding command ( $\overline{\text{RD}}$  or  $\overline{\text{WR}}$ ).
- 5) If the Asynchronous  $\overline{\text{READY}}$  signal does not fulfill the indicated setup and hold times with respect to CLKOUT (eg. because CLKOUT is not enabled), it must fulfill  $t_{37}$  in order to be safely synchronized. This is guaranteed, if  $\overline{\text{READY}}$  is removed in reponse to the command (see Note 4).
- 6) Multiplexed bus modes have a MUX waitstate added after a bus cycle, and an additional MTTC waitstate may be inserted here.  
For a multiplexed bus with MTTC waitstate this delay is 2 CLKOUT cycles, for a demultiplexed bus without MTTC waitstate this delay is zero.
- 7) The next external bus cycle may start here.

Package Outlines

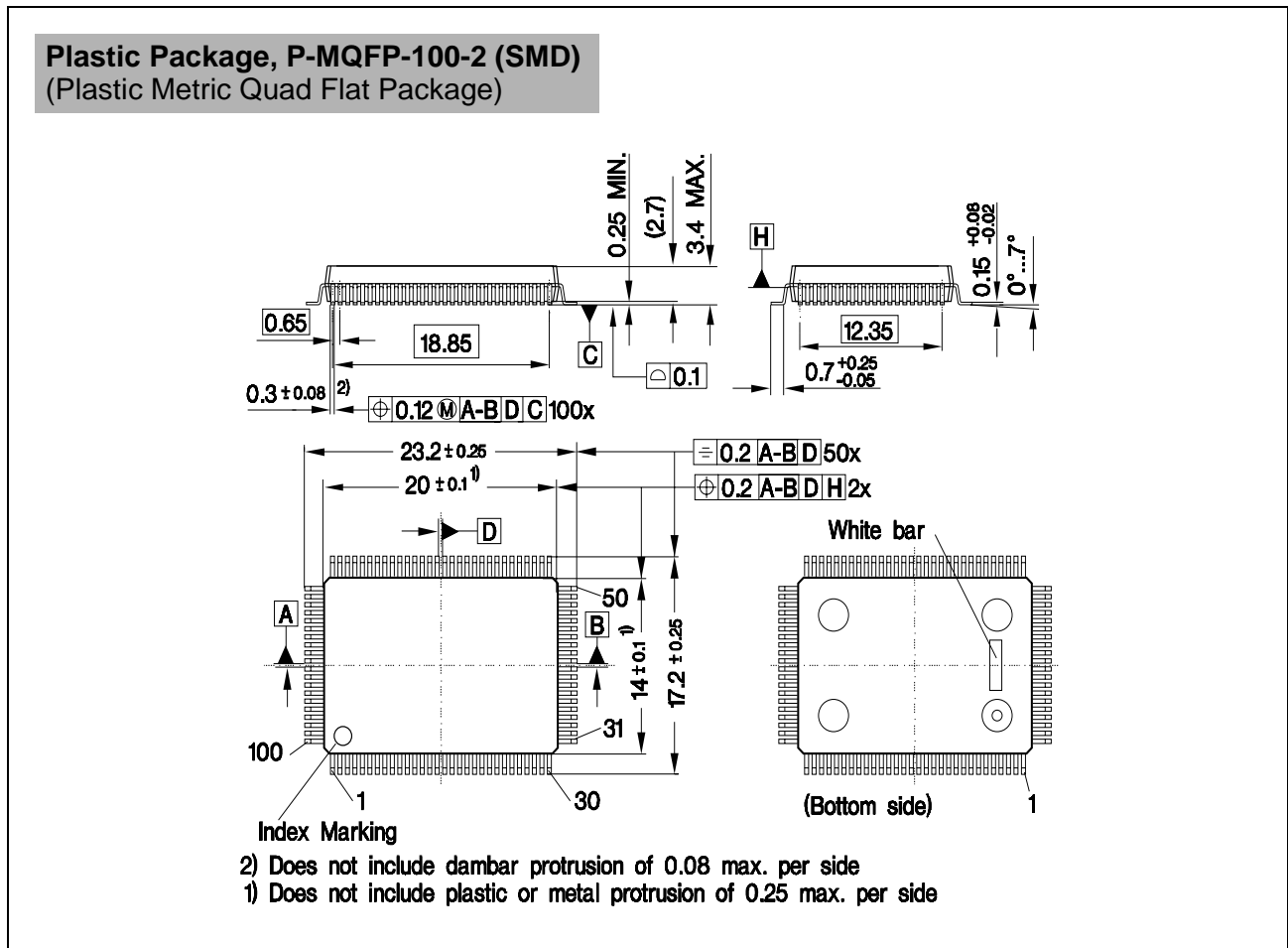


Figure 17

Sorts of Packing

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

SMD = Surface Mounted Device

Dimensions in mm

Package Outlines (cont'd)

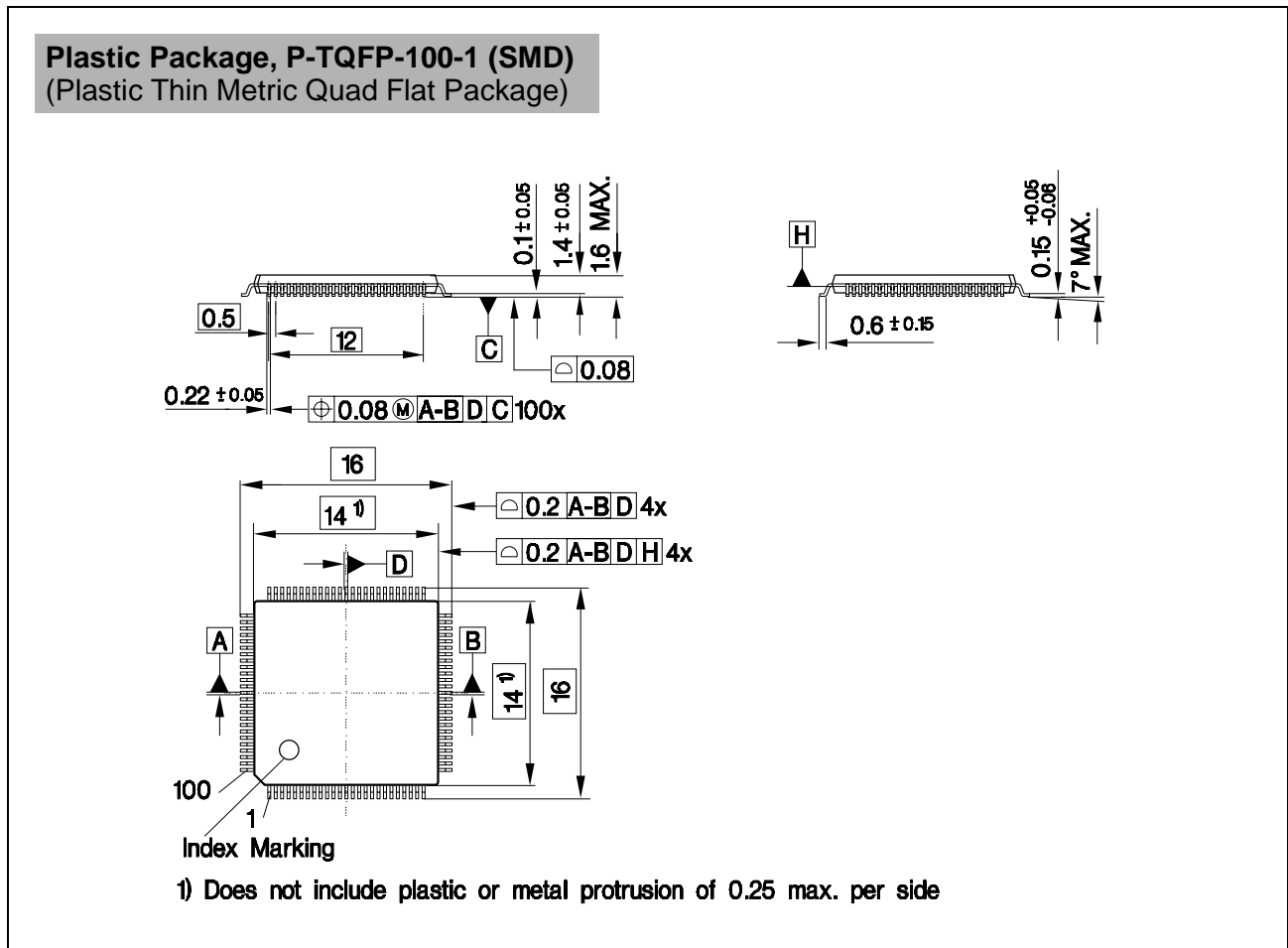


Figure 18

**Sorts of Packing**

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

SMD = Surface Mounted Device

Dimensions in mm