SIEMENS

Microcomputer Components

16-Bit CMOS Microcontroller

C163-16F

Data Sheet 10.97 Preliminary

C163-16F Revision History:	Version 10.97 Preliminary
Previous Releases:	Feature List 03.97 (Advance Information) Feature List 05.96 (Advance Information)

Page	Subjects (changes since last revision)

Edition 1997-10-01

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

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[1] of the Semiconductor Group of Siemens AG, may only be used in life-support devices or systems[2] 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.

C16x-Family of High-Performance CMOS 16-Bit Microcontrollers

Preliminary

C163-16F 16-Bit Microcontroller

- High Performance 16-bit CPU with 4-Stage Pipeline
- 80 ns Instruction Cycle Time at 25-MHz CPU Clock
- 400 ns Multiplication (16 × 16 bits), 800 ns 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 on-chip PLL or via direct or prescaled clock input
- Up to 16 MBytes Linear Address Space for Code and Data
- 1 KByte On-Chip Internal RAM (IRAM)
- 128 KByte On-Chip Flash EPROM with 0 Waitstate Operation
- 5 V Flash Programming Voltage
- Minimum 1000 Program/Erase Cycles
- Minimum 10 Years Data Retention
- Programmable External Bus Characteristics for Different Address Ranges
- 8-Bit or 16-Bit External Data Bus
- Multiplexed or Demultiplexed External Address/Data Buses
- Five Programmable Chip-Select Signals (Latched/Unlatched)
- Hold- and Hold-Acknowledge Bus Arbitration Support
- 1024 Bytes On-Chip Special Function Register Area
- Idle and Power Down Modes
- 8-Channel Interrupt-Driven Single-Cycle Data Transfer Facilities via Peripheral Event Controller (PEC)
- 16-Priority-Level Interrupt System with 20 Sources, Sample-Rate down to 40 ns
- Two Multi-Functional General Purpose Timer Units with 5 Timers
- Two Serial Channels (Synchronous/Asynchronous and Synchronous)
- Programmable Watchdog Timer
- Oscillator Watchdog
- Up to 77 General Purpose I/O Lines
- Boot Support Mechanism
- Supported by a Wealth of Development Tools like C-Compilers, Macro-Assembler Packages, Emulators, Evaluation Boards, HLL-Debuggers, Simulators, Logic Analyzer Disassemblers, Programming Boards
- 100-Pin TQFP Package (Thin QFP)

This document describes the **SAB-C163-16FF** and the **SAB-C163-16F25F**.
For simplicity all versions are referred to by the term **C163-16F** throughout this document.

Introduction

The C163-16F is a new derivative of the Siemens C166 family of 16-bit single-chip CMOS microcontrollers. It combines high CPU performance (up to 12.5 million instructions per second) with high peripheral functionality and enhanced IO-capabilities.

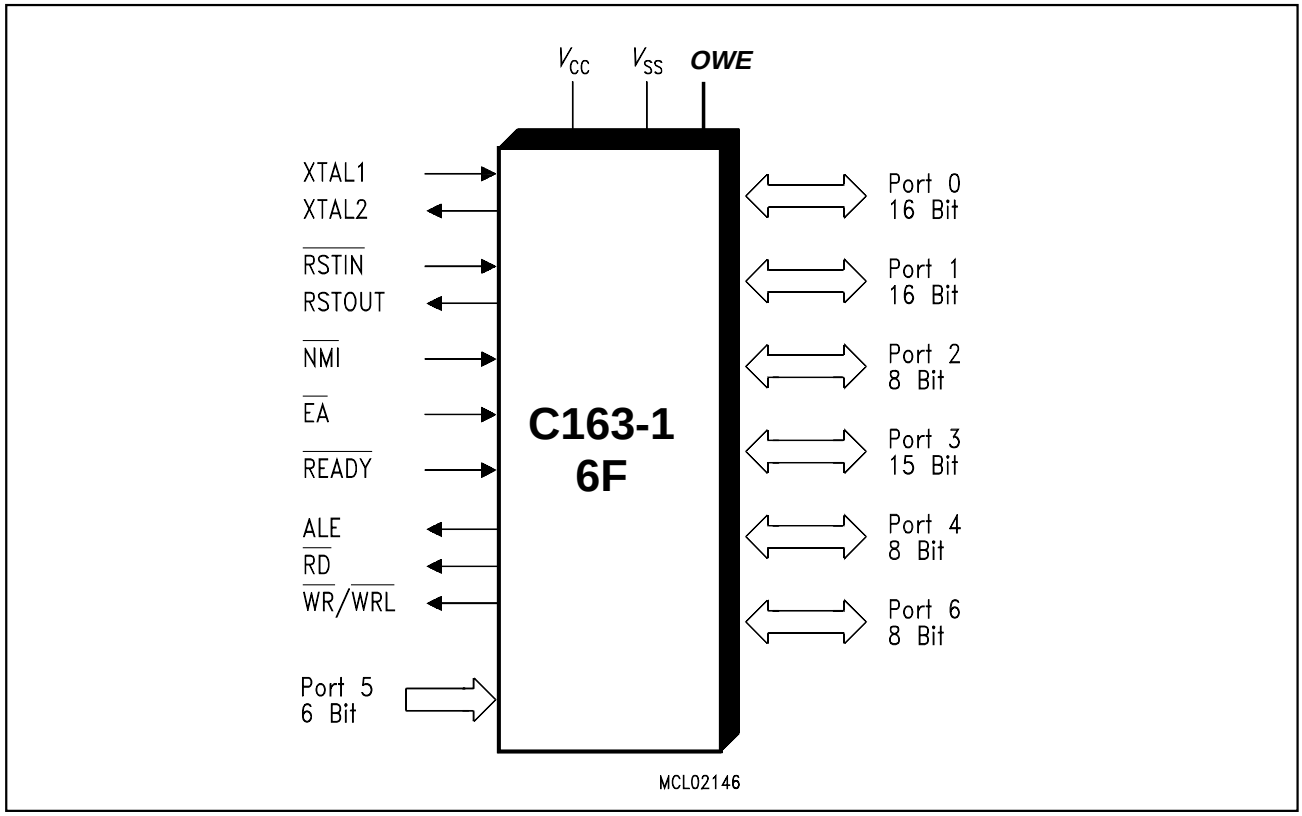


Figure 1
Logic Symbol

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 C163-16F 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.

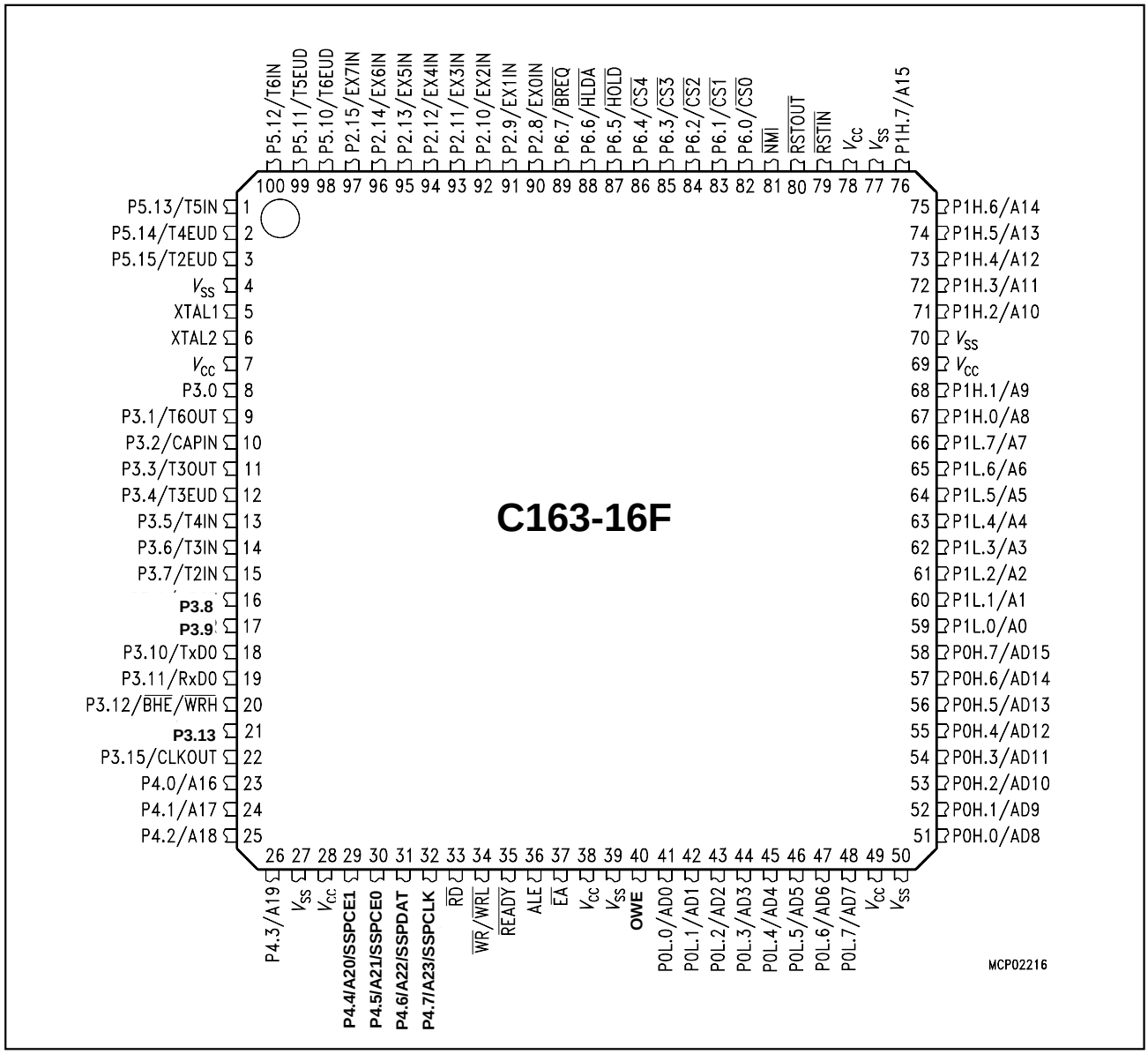


Figure 2
Pin Configuration TQFP Package (top view)

Pin Definitions and Functions

Symbol	Pin Number	Input (I) Output (O)	Function		
P5.10 – P5.15	98-100 1 - 3	I I	Port 5 is a 6-bit input-only port with Schmitt-Trigger characteristics. The pins of Port 5 also serve as timer inputs:		
	98	I	P5.10	T6EUD	GPT2 Timer T6 Ext.Up/Down Ctrl.Input
	99	I	P5.11	T5EUD	GPT2 Timer T5 Ext.Up/Down Ctrl.Input
	100	I	P5.12	T6IN	GPT2 Timer T6 Count Input
	1	I	P5.13	T5IN	GPT2 Timer T5 Count Input
	2	I	P5.14	T4EUD	GPT1 Timer T4 Ext.Up/Down Ctrl.Input
	3	I	P5.15	T2EUD	GPT1 Timer T2 Ext.Up/Down Ctrl.Input
XTAL1	5	I	XTAL1: Input to the oscillator amplifier and input to the internal clock generator.		
XTAL2	6	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	8 – 21, 22	I/O I/O I/O	Port 3 is a 15-bit (P3.14 is missing) 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. Port 3 outputs can be configured as push/ pull or open drain drivers. The following Port 3 pins also serve for alternate functions:		
	9	O	P3.1	T6OUT	GPT2 Timer T6 Toggle Latch Output
	10	I	P3.2	CAPIN	GPT2 Register CAPREL Capture Input
	11	O	P3.3	T3OUT	GPT1 Timer T3 Toggle Latch Output
	12	I	P3.4	T3EUD	GPT1 Timer T3 Ext.Up/Down Ctrl.Input
	13	I	P3.5	T4IN	GPT1 Timer T4 Input for Count/Gate/Reload/Capture
	14	I	P3.6	T3IN	GPT1 Timer T3 Count/Gate Input
	15	I	P3.7	T2IN	GPT1 Timer T2 Input for Count/Gate/Reload/Capture
	18	O	P3.10	T×D0	ASC0 Clock/Data Output (Asyn./Syn.)
	19	I/O	P3.11	R×D0	ASC0 Data Input (Asyn.) or I/O (Syn.)
	20	O	P3.12	$\overline{\text{BHE}}$	Ext. Memory High Byte Enable Signal,
		O		$\overline{\text{WRH}}$	Ext. Memory High Byte Write Strobe
	22	O	P3.15	CLKOUT	System Clock Output (=CPU Clock)

Pin Definitions and Functions (cont'd)

Symbol	Pin Number	Input (I) Output (O)	Function		
P4.0 – P4.7	23 - 26, 29 - 32	I/O	Port 4 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. In case of an external bus configuration, Port 4 can be used to output the segment address lines:		
	23	O	P4.0	A16	Least Significant Segment Addr. Line
	...	...	...	...	...
	26	O	P4.3	A19	Segment Address Line
	29	O	P4.4	A20	Segment Address Line,
		O		SSPCE1	SSP Chip Enable Line 1
	30	O	P4.5	A21	Segment Address Line,
		O		SSPCE0	SSP Chip Enable Line 0
	31	O	P4.6	A22	Segment Address Line,
		I/O		SSPDAT	SSP Data Input/Output Line
	32	O	P4.7	A23	Most Significant Segment Addr. Line
		O		SSPCLK	SSP Clock Output Line
$\overline{RD}$	33	O	External Memory Read Strobe. $\overline{RD}$ is activated for every external instruction or data read access.		
$\overline{WR}$/ $\overline{WRL}$	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.		
$\overline{READY}$	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	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.		
$\overline{EA}$	37	I	External Access Enable pin. A low level at this pin during and after Reset forces the C163-16F to begin instruction execution out of external memory. A high level forces execution out of the internal ROM. The C163-16F must have this pin tied to '0'.		

Pin Definitions and Functions (cont'd)

Symbol	Pin Number	Input (I) Output (O)	Function
PORT0: P0L.0 – P0L.7, P0H.0 - P0H.7	41 – 48 51 – 58	I/O	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. In case of an external bus configuration, PORT0 serves as the address (A) and address/data (AD) bus in multiplexed bus modes and as the data (D) bus in demultiplexed bus modes. **Demultiplexed bus modes:** Data Path Width: 8-bit 16-bit P0L.0 – P0L.7: D0 – D7 D0 - D7 P0H.0 – P0H.7: I/O D8 - D15 **Multiplexed bus modes:** 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	59 - 66 67 - 68, 71 - 76	I/O	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.
$\overline{\text{RSTIN}}$	79	I	Reset Input with Schmitt-Trigger characteristics. A low level at this pin for a specified duration while the oscillator is running resets the C163-16F. An internal pullup resistor permits power-on reset using only a capacitor connected to V_{SS}.
$\overline{\text{RSTOUT}}$	80	O	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. $\overline{\text{RSTOUT}}$ remains low until the EINIT (end of initialization) instruction is executed.
$\overline{\text{NMI}}$	81	I	Non-Maskable Interrupt Input. A high to low transition at this pin causes the CPU to vector to the NMI trap routine. When the PWRDN (power down) instruction is executed, the $\overline{\text{NMI}}$ pin must be low in order to force the C163-16F to go into power down mode. If $\overline{\text{NMI}}$ is high, when PWRDN is executed, the part will continue to run in normal mode. If not used, pin $\overline{\text{NMI}}$ should be pulled high externally.

Pin Definitions and Functions (cont'd)

Symbol	Pin Number	Input (I) Output (O)	Function
P6.0 – P6.7	82 - 89	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. Port 6 outputs can be configured as push/ pull or open drain drivers. The following 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 $\overline{HOLD}$ External Master Hold Request Input P6.6 $\overline{HLDA}$ Hold Acknowledge Output or Input (Master mode: O, Slave mode: I) P6.7 $\overline{BREQ}$ Bus Request Output
	82	O	
	...	...	
	86	O	
	87	I	
	88	I/O	
	89	O	
P2.8 – P2.15	90 - 97	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. Port 2 outputs can be configured as push/ pull or open drain drivers. 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
	90	I	
	...	...	
	97	I	
OWE	40	-	Oscillator Watchdog Enable. This pin enables the oscillator watchdog when high or disables it when low e.g. for testing purposes. An internal pullup device holds this input high if nothing is driving it. The oscillator watchdog operation is also controlled via software.
V_{CC}	7, 28, 38, 49, 69, 78	-	Digital Supply Voltage: + 5 V during normal operation and idle mode. ≥ 2.5 V during power down mode
V_{SS}	4, 27, 39, 50, 70, 77	-	Digital Ground.

Functional Description

The architecture of the C163-16F 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 C163-16F.

Note: All time specifications refer to a CPU clock of 25 MHz
(see definition in the AC Characteristics section).

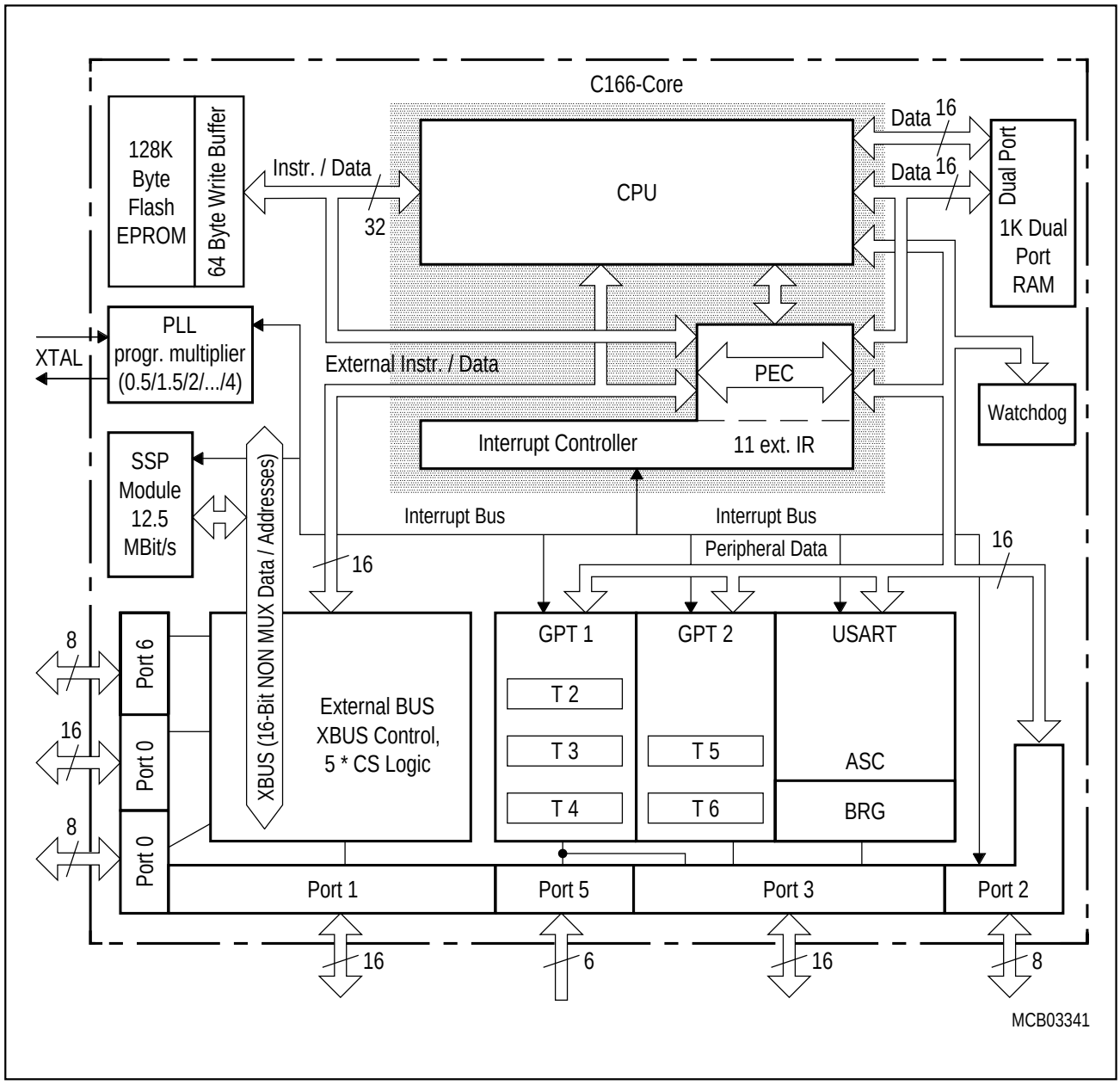


Figure 3
Block Diagram

Memory Organization

The memory space of the C163-16F 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 bytewise or wordwise. Particular portions of the on-chip memory have additionally been made directly bit addressable.

The C163-16F incorporates 128 KByte of on-chip Flash memory, organized as four 32 KByte sectors. Each sector can be separately erased and programmed. A detailled description of the Flash memory can be found later in this data sheet.

1 KByte of on-chip 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 wordwide (R0 to R15) and/or bytewide (RL0, RH0, …, RL7, RH7) so-called General Purpose Registers (GPRs).

1024 bytes (2 * 512 bytes) of the address space are reserved for the Special Function Register areas (SFR space and ESFR space). SFRs are wordwide registers which are used for controlling and monitoring functions of the different on-chip units. Unused SFR addresses are reserved for other/future members of the C16x family.

In order to meet the needs of designs where more memory is required than is provided on chip, up to 16 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-/24-bit Addresses, 16-bit Data, Demultiplexed

– 16-/18-/20-/24-bit Addresses, 16-bit Data, Multiplexed

– 16-/18-/20-/24-bit Addresses, 8-bit Data, Multiplexed

– 16-/18-/20-/24-bit Addresses, 8-bit Data, Demultiplexed

In the demultiplexed bus modes, addresses are output on PORT1 and data is input/output on PORT0 or P0L, respectively. In the multiplexed bus modes both addresses and data use PORT0 for input/output.

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. The C163-16F offers the possibility to switch the $\overline{CS}$ outputs to an unlatched mode. In this mode the internal filter logic is switched off and the $\overline{CS}$ signals are directly generated from the address. The unlatched $\overline{CS}$ mode is enabled by setting CSCFG (SYSCON.6).

Access to very slow memories is supported via a particular ‘Ready’ function.

A $\overline{HOLD}$/$\overline{HLDA}$ protocol is available for bus arbitration and allows to share external resources with other bus masters. The bus arbitration is enabled by setting bit HLDEN in register SYSCON. After setting HLDEN once, pins P6.7...P6.5 ($\overline{BREQ}$, $\overline{HLDA}$, $\overline{HOLD}$) are automatically controlled by the EBC. In Master Mode (default after reset) the $\overline{HLDA}$ pin is an output.

By setting bit DP6.7 to ‘1’ the Slave Mode is selected where pin $\overline{HLDA}$ is switched to input. This allows to directly connect the slave controller to another master controller without glue logic.

For applications which require less than 16 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 8 address lines, if an address space of 16 MBytes is used.

Note: When the on-chip SSP Module is to be used the segment address output on Port 4 must be limited to 4 bits (i.e. A19...A16) in order to enable the alternate function of the SSP interface pins.

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 C163-16F's instructions can be executed in just one machine cycle which requires 80 ns at 25-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 × 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.

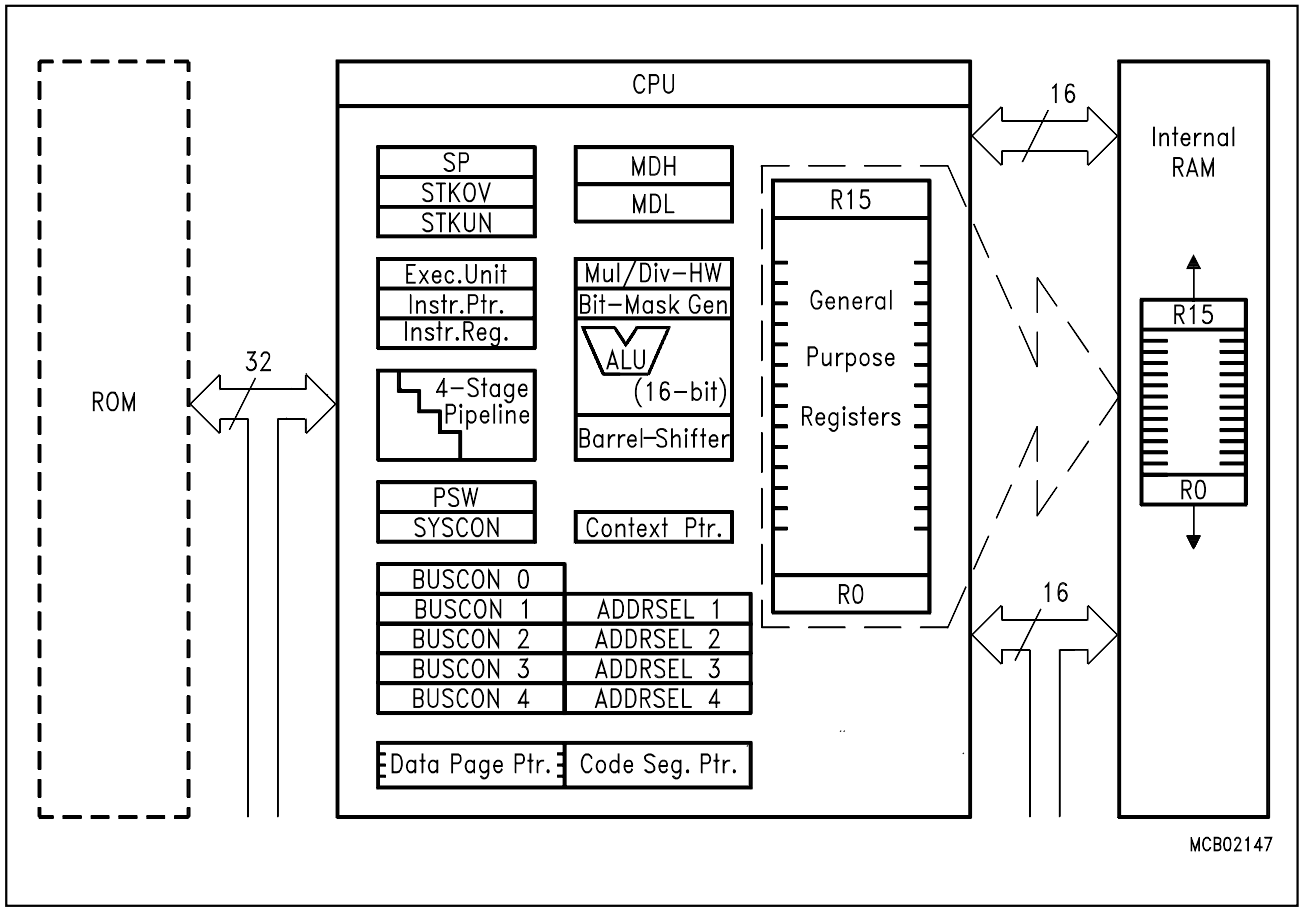


Figure 4
CPU Block Diagram

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 C163-16F 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 200 ns to 480 ns (in case of internal program execution), the C163-16F is capable of reacting very fast to the occurence of non-deterministic events.

The architecture of the C163-16F 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 implicity 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 C163-16F 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 C163-16F 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	$00'0060_H$	18_H
External Interrupt 1	CC9IR	CC9IE	CC9INT	$00'0064_H$	19_H
External Interrupt 2	CC10IR	CC10IE	CC10INT	$00'0068_H$	$1A_H$
External Interrupt 3	CC11IR	CC11IE	CC11INT	$00'006C_H$	$1B_H$
External Interrupt 4	CC12IR	CC12IE	CC12INT	$00'0070_H$	$1C_H$
External Interrupt 5	CC13IR	CC13IE	CC13INT	$00'0074_H$	$1D_H$
External Interrupt 6	CC14IR	CC14IE	CC14INT	$00'0078_H$	$1E_H$
External Interrupt 7	CC15IR	CC15IE	CC15INT	$00'007C_H$	$1F_H$
GPT1 Timer 2	T2IR	T2IE	T2INT	$00'0088_H$	22_H
GPT1 Timer 3	T3IR	T3IE	T3INT	$00'008C_H$	23_H
GPT1 Timer 4	T4IR	T4IE	T4INT	$00'0090_H$	24_H
GPT2 Timer 5	T5IR	T5IE	T5INT	$00'0094_H$	25_H
GPT2 Timer 6	T6IR	T6IE	T6INT	$00'0098_H$	26_H
GPT2 CAPREL Register	CRIR	CRIE	CRINT	$00'009C_H$	27_H
ASC0 Transmit	S0TIR	S0TIE	S0TINT	$00'00A8_H$	$2A_H$
ASC0 Transmit Buffer	S0TBIR	S0TBIE	S0TBINT	$00'011C_H$	47_H
ASC0 Receive	S0RIR	S0RIE	S0RINT	$00'00AC_H$	$2B_H$
ASC0 Error	S0EIR	S0EIE	S0EINT	$00'00B0_H$	$2C_H$
SSP Interrupt	XP1IR	XP1IE	XP1INT	$00'0104_H$	41_H
PLL Unlock / OWD	XP3IR	XP3IE	XP3INT	$00'010C_H$	43_H

The C163-16F also provides an excellent mechanism to identify and to process exceptions or error conditions that arise during run-time, so-called ‘Hardware Traps’. Hardware traps cause immediate non-maskable system reaction which is similar to a standard interrupt service (branching to a dedicated vector table location). The occurence of a hardware trap is additionally signified by an individual bit in the trap flag register (TFR). Except when another higher prioritized trap service is in progress, a hardware trap will interrupt any actual program execution. In turn, hardware trap services can normally not be interrupted by standard or PEC interrupts.

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	$00'0000_H$	00_H	III
Software Reset		RESET	$00'0000_H$	00_H	III
Watchdog Timer Overflow		RESET	$00'0000_H$	00_H	III
Class A Hardware Traps:					
Non-Maskable Interrupt	NMI	NMITRAP	$00'0008_H$	02_H	II
Stack Overflow	STKOF	STOTRAP	$00'0010_H$	04_H	II
Stack Underflow	STKUF	STUTRAP	$00'0018_H$	06_H	II
Class B Hardware Traps:					
Undefined Opcode	UNDOPC	BTRAP	$00'0028_H$	$0A_H$	I
Protected Instruction Fault	PRTFLT	BTRAP	$00'0028_H$	$0A_H$	I
Illegal Word Operand Access	ILLOPA	BTRAP	$00'0028_H$	$0A_H$	I
Illegal Instruction Access	ILLINA	BTRAP	$00'0028_H$	$0A_H$	I
Illegal External Bus Access	ILLBUS	BTRAP	$00'0028_H$	$0A_H$	I
Reserved			$[2C_H – 3C_H]$	$[0B_H – 0F_H]$	
Software Traps TRAP Instruction			Any $[00'0000_H – 00'01FC_H]$ in steps of 4_H	Any $[00_H – 7F_H]$	Current CPU Priority

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 three basic modes of operation, which are Timer, Gated Timer, and Counter 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 320 ns (@ 25-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.

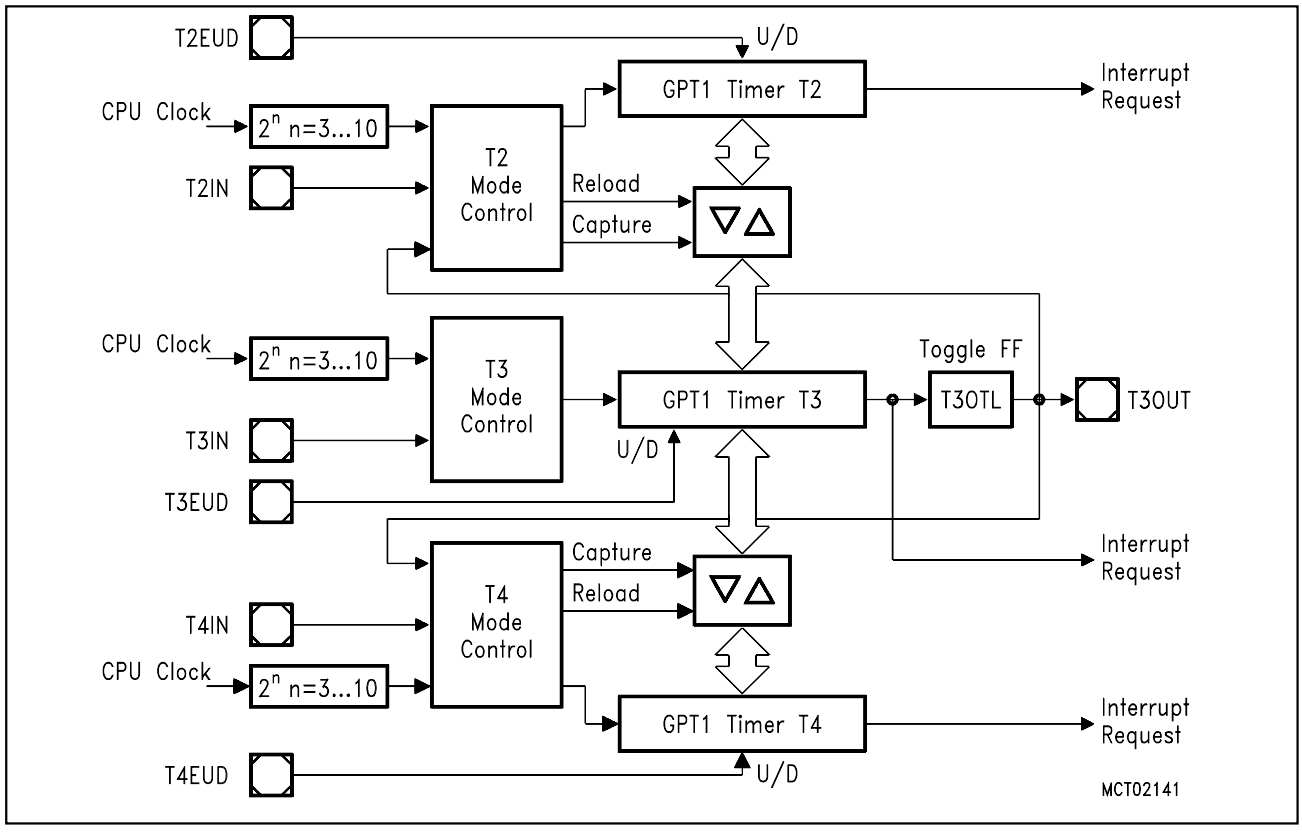


Figure 5
Block Diagram of GPT1

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 port a 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 is captured into T2 or T4 in response to a signal at their associated input pins (TxIN). Timer T3 is reloaded with the contents of T2 or T4 triggered either by an external signal or by a selectable state transition of its toggle latch T3OTL. When both T2 and T4 are configured to alternately reload T3 on opposite state transitions of T3OTL with the low and high times of a PWM signal, this signal can be constantly generated without software intervention.

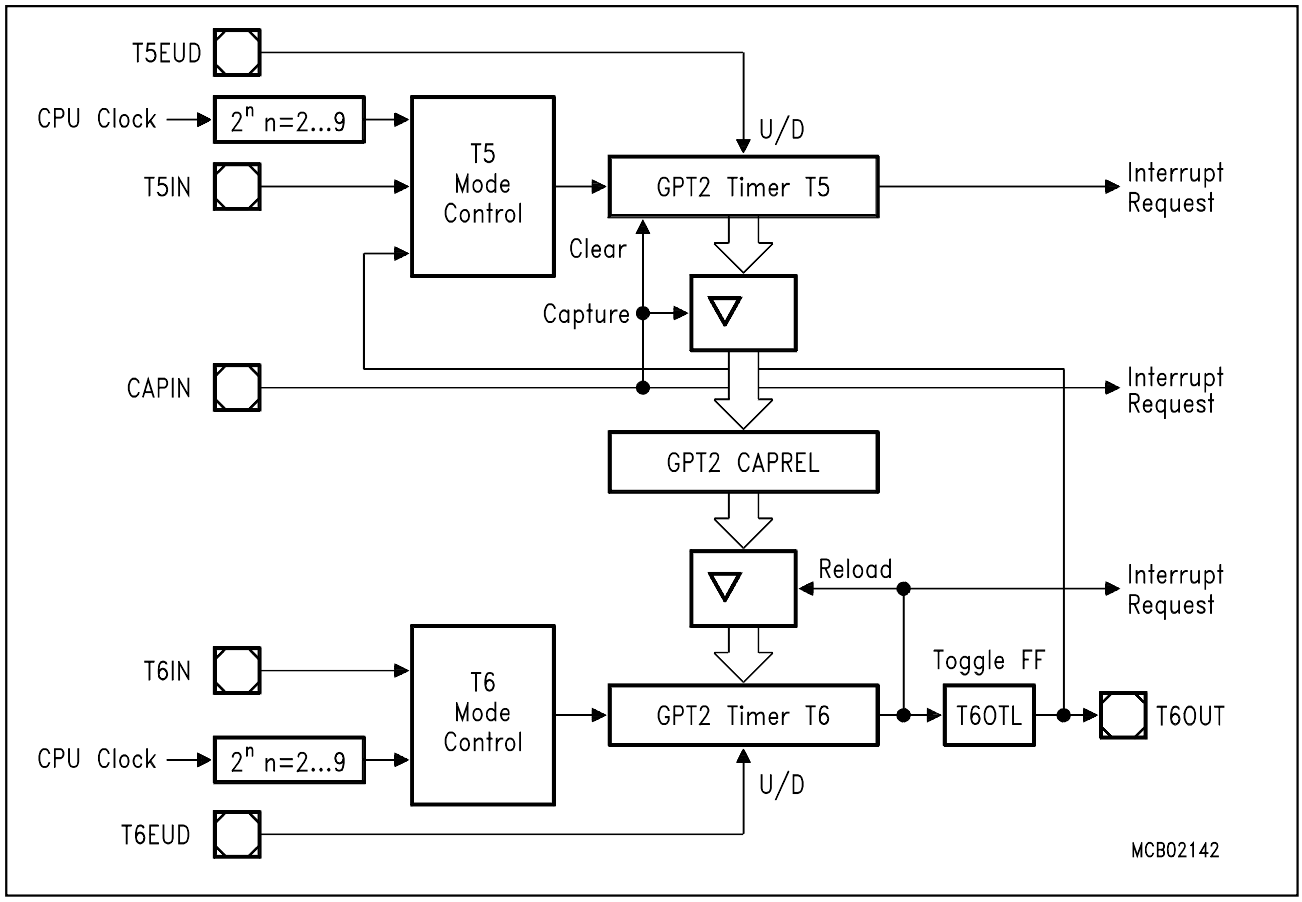


Figure 6
Block Diagram of GPT2

With its maximum resolution of 160 ns (@ 25 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 or with external signals. The count direction (up/down) for each timer is programmable by software or may additionally be altered dynamically by an external signal on a port pin (TxEUD). Timer T6 has an output toggle latch (T6OTL) which changes its state on each timer overflow/underflow. Concatenation of the timers is supported via T6OTL.

The state of this latch may be used to clock timer T5, or it may be output on a port pin (T6OUT). The overflows/underflows of timer T6 can additionally be used to clock the CAPCOM timers T0 or T1, and to cause a reload from the CAPREL register. The CAPREL register may capture the contents of timer T5 based on an external signal transition on the corresponding port pin (CAPIN), and timer T5 may optionally be cleared after the capture procedure. This allows absolute time differences to be measured or pulse multiplication to be performed without software overhead.

Parallel Ports

The C163-16F provides up to 77 I/O 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 I/O ports can be configured (pin by pin) for push/pull operation or open-drain operation via control registers. During the internal reset, all port pins are configured as inputs.

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 A23/19/17...A16 in systems where segmentation is enabled to access more than 64 KBytes of memory. Port 6 provides optional bus arbitration signals ($\overline{\text{BREQ}}$, $\overline{\text{HLDA}}$, $\overline{\text{HOLD}}$) and chip select signals. 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 timer control signals. All port lines that are not used for these alternate functions may be used as general purpose I/O lines.

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 Synchronous Serial Port (SSP).

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 625 KBaud and half-duplex synchronous communication at up to 2.5 MBaud @ 20 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 SSP transmits 1...3 bytes or receives 1 byte after sending 1...3 bytes synchronously to a shift clock which is generated by the SSP. The SSP can start shifting with the LSB or with the MSB and allows to select shifting and latching clock edges as well as the clock polarity. Up to two chip select lines may be activated in order to direct data transfers to one or both of two peripheral devices.
One general interrupt vector is provided for the SSP.

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 $\overline{\text{RSTOUT}}$ pin low in order to allow external hardware components to be reset.

The Watchdog Timer is a 16-bit timer, clocked with the system clock divided 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 20 s and 336 ms can be monitored (@ 25 MHz). The default Watchdog Timer interval after reset is 5.24 ms (@ 25 MHz).

Oscillator Watchdog

The Oscillator Watchdog (OWD) monitors the clock signal generated by the on-chip oscillator (either with a crystal or via external clock drive). For this operation the PLL provides a clock signal which is used to supervise transitions on the oscillator clock. This PLL clock is independent from the XTAL1 clock. When the expected oscillator clock transitions are missing the OWD activates the PLL Unlock / OWD interrupt node and supplies the CPU with the PLL clock signal. Under these circumstances the PLL will oscillate with its basic frequency.

The oscillator watchdog of the C163-16F can be disabled in two ways:

- Provide a low level on pin OWE, or
- Set bit OWDDIS (SYSCON.4).

If either condition is true the on-chip oscillator watchdog is disabled and the CPU clock is always fed from the oscillator input. After reset the oscillator watchdog is enabled and active.

The operation of the OWD is controlled according to the table below:

	OWDDIS = '0'	**OWDDIS = '1'**
OWE = '0', ie. $V_{OWE} = V_{IL}$	Disabled	Disabled
OWE = '1', ie. $V_{IH} \leq V_{OWE} < 13.2V$	**OWD active**	Disabled

Note: The CPU clock source is only switched back to the oscillator clock after a hardware reset.

Flash Memory Overview

The on-chip Flash module of the C163-16F has a capacity of 128 KByte and combines the advantages of a very fast read access of 32 bit in one machine cycle with protected but simple writing algorithms for programming and erase. Read accesses of code and data are possible in any addressing mode, thus realizing the highest CPU performance with fetch of double word instructions in a single cycle. Based on the Flash cell concept (split gate) special algorithms for over/ under-programming or erase, with verify operations, are not necessary. For optimized programming efficiency, a burst (paging) mode is offered which allows to load up to 64 Bytes into an assembly buffer with normal CPU timing before this buffer is stored into the Flash with a write command. The algorithms for the program and erase operations are automatically controlled by the internal command state machine.

Note: Erased Flash memory cells contain all ‘0’s, contrary to standard EPROMs.

The C163-16F Flash module is a 1 Mbit, 5 Volt-only Flash memory organized as 32K Doublewords of 32 bit each. The physical structure of the Flash array allows simultaneous access to 64 Byte for write operations.

The lower 32 KBytes of the on-chip Flash memory of the C163-16F can be mapped to either segment 0 ($00'0000_H$ to $00'7FFF_H$) or segment 1 ($01'0000_H$ to $01'7FFF_H$) during the initialization phase to allow external memory to be used for additional system flexibility. The upper 96 KBytes of the on-chip Flash memory are assigned to locations $01'8000_H$ to $02'FFFF_H$.

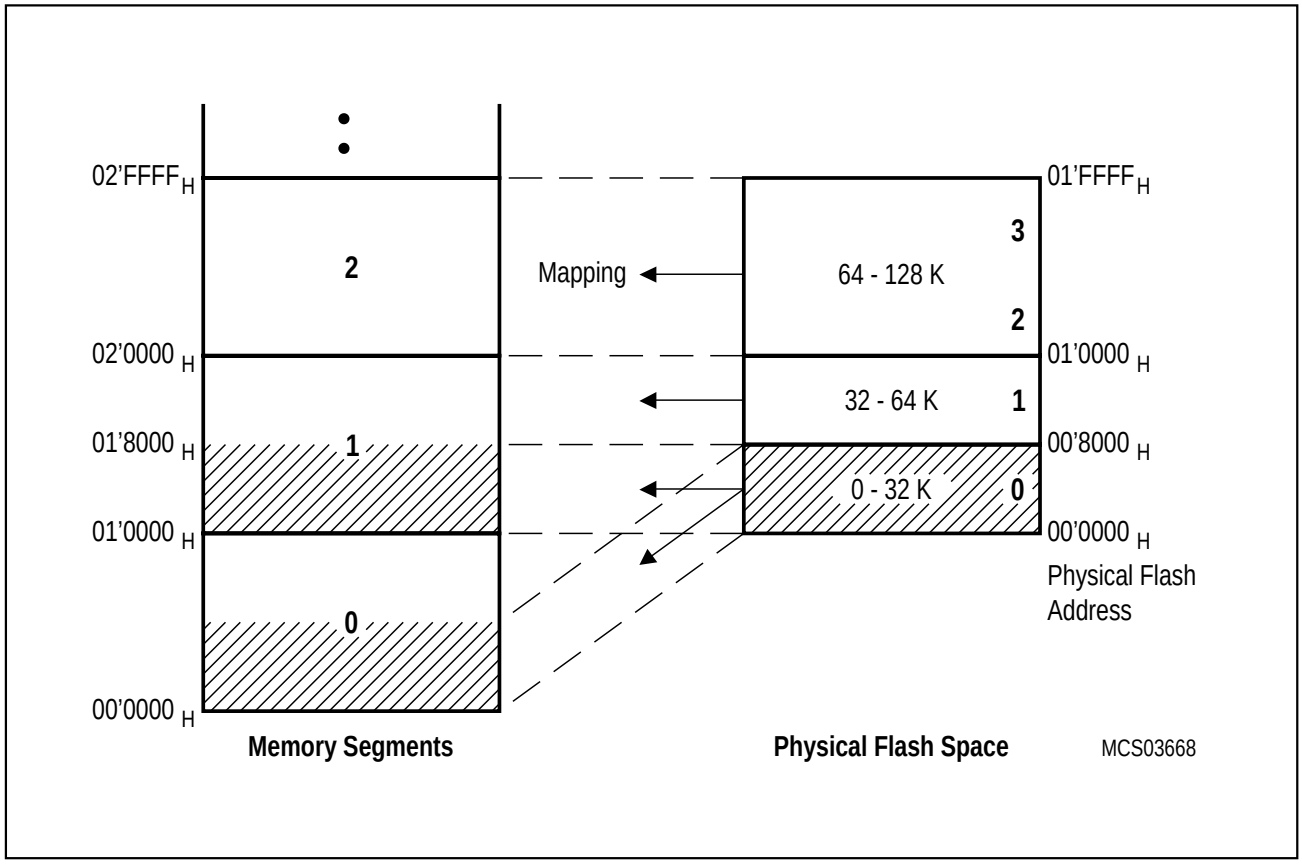


Figure 7
Mapping of the On-chip Flash Module Sectors

In standard mode (the normal operating mode) the Flash memory appears like the standard on-chip ROM of C167 devices with the same timing and functionality. Instruction fetches and data operand reads are performed with all addressing modes of the C16x instruction set.

Programming and erasing is controlled via special command sequences. This avoids inadvertent destruction of the Flash contents at a reasonably low software overhead. Command sequences consist of subsequent write (or read) accesses to virtual locations within the Flash space. These virtual locations are defined by special addresses (see command sequence table) and require register-indirect addressing.

Flash Memory Configuration

Upon reset the default memory configuration of the C163-16F is determined by the state of its $\overline{EA}$ pin. When $\overline{EA}$ is high the startup code is fetched from the on-chip Flash memory, when $\overline{EA}$ is low the internal Flash is disabled and the startup code is fetched from external memory.

In order to access the on-chip Flash memory after booting from external memory the internal Flash must be enabled via software by setting bit ROMEN in register SYSCON. The lower 32 KBytes of the Flash memory can be mapped to segment 0 or to segment 1, controlled by bit ROMS1 in register SYSCON. Mapping to segment 1 preserves the external memory containing the startup code, while mapping to segment 0 replaces the lower 32 KBytes of the external memory with on-chip Flash memory. In this case a valid vector table must be provided in the Flash memory. As the on-chip Flash memory covers more than segment 0 segmentation should be enabled (by clearing bit SGTDIS in register SYSCON) in order to ensure correct stack handling when branching to the upper segments.

Whenever the internal memory configuration of the C163-16F is changed (enable, disable, mapping) the following procedure must be used to ensure correct operation:

- Configure the internal Flash as required
- Execute an inter-segment branch (JMPS, CALLS, RETS)
- Reload all four DPP registers

Note: Instructions that configure the internal Flash may only be executed from internal RAM or from external memory, **not** from the Flash itself.
Register SYSCON can only be modified **before** the execution of the EINIT instruction.

Boot Support

The C163-16F's boot support provides a mechanism to load the startup code and/or the Flash programming routines from a remote code source e.g. via a serial interface without requiring additional external memory. This allows for firmware updates of the Flash memory for program and/or data values. When the boot support is enabled (by configuring pin P0L.4 low during reset) the C163-16F will start code execution at location $02'0000_H$ rather than at location $00'0000_H$.

Note: The boot support mechanism is not supported by the A-step devices.

Flash Operating Modes

For the operation of the on-chip Flash module basically three operating modes can be distinguished:

In Standard Read Mode the Flash memory appears like a standard ROM allowing all code and data accesses in any addressing mode without waitstates. Standard read mode is entered:

- after the deactivation of CPU reset (max. 100 s after reset state is finished)
- after a successful erase operation
- after a successful programming operation
- when a command sequence error is detected
- after a “reset to read” command

Note: Standard read mode is indicated by status bit BUSY=’0’.

In Burst Mode a programming operation is prepared by writing to the Flash assembly buffer. Burst mode begins after the “Enter Burst Mode” command sequence and ends after the “Store Burst” command sequence. Burst mode allows the assembly (writing) of 32 words (=64 bytes) at standard CPU speed, which are the programmed in a single self-timed programming cycle.

Burst mode is only left after the “Store Burst” command sequence, if the buffer was filled with exactly 32 words. If more or less than 32 words have been written the “Store Burst” command will not be executed and a burst error is indicated instead (BUER=’1’).

Note: During burst mode standard read accesses can still be executed. However, the code to fill the buffer must be executed from locations outside the Flash memory (e.g. RAM or external memory).

In Command Mode the C163-16F executes a Flash command (erase sector, program buffer, reset state machine, etc.) which has been defined by a previous command sequence. During command mode (indicated by bit BUSY=’1’) no other Flash operations/accesses are possible except for reading the Flash status.

General rules for command sequences:

- code must be executed from locations outside the Flash memory
- all addresses must point into the active Flash space
- only register-indirect addressing is possible
- pauses between command cycles are allowed

Note: Carefully check the addresses used during command sequences. When using DPPs or EXT instructions the resulting address must be within the active Flash space. For the special addresses (see table below) bits A15...A1 are regarded. A sector address must point to the first (lowest) location within the target sector.

Command Sequences

The table below summarizes the implemented command sequences and describes how to execute them (please note that register-indirect addressing is required):

Command Sequence Definitions

Cycle	Reset to read mode	Enter burst mode	Load burst data	Store burst buffer	Erase sector	Read flash status	Clear status
1	A = $AAAA_H$ D = $xxF0_H$	A = $AAAA_H$ D = $xx50_H$	A = $A0F2_H$ D = WDAT	A = $AAAA_H$ D = $xxAA_H$	A = $AAAA_H$ D = $xxAA_H$	A = $AAAA_H$ D = $xxFA_H$	A = $AAAA_H$ D = $xxF5_H$
2		A = WLOC D = WDAT	no read	A = 5554_H D = $xx55_H$	A = 5554_H D = $xx55_H$	A = SLOC D = status	
3		no read		A = $AAAA_H$ D = $xxA0_H$	A = $AAAA_H$ D = $xx80_H$		
4				A = WLOC D = WDAT	A = 5554_H D = $xxAA_H$		
5					A = $AAAA_H$ D = $xx55_H$		
6					A = SLOC D = $xx30_H$		

Notes:

WLOC is the burst base address to which the 64-Byte buffer shall be written, e.g. $01'ABC0_H$.
WDAT is the data word which shall be stored in the buffer.
SLOC is the first location within the target sector, e.g. $01'8000_H$ for sector 1.
no read: the first cycle after an “Enter burst mode” or “Load burst data” command sequence will return dummy data if a read access to the Flash area is executed.

The locations WLOC and SLOC refer to the first (lowest) location within the respective block, i.e. a 64-Byte buffer block for WLOC and a 32-KByte sector for SLOC.
All bits that represent the offset in this block must be ‘0’, the higher address bits identify the block.

The first word of programming data is written to the buffer with the “Enter burst mode” command, the last word is written with the “Store burst buffer” command. The medium 30 words are written with “Load burst data” commands. Note that WLOC is the same for a complete programming sequence as the buffer address is incremented internally.

The “Read Flash status” command sequence may be executed during command mode in order to check the BUSY bit of the Flash module.

Caution:

Writing to a Flash page (space for the 64-Byte buffer) **more than once** before erasing may destroy data stored in neighbour cells! This is especially important for programming algorithms that do not write to sequential locations.

The Flash Status Register FSR reflects the overall and also sector specific status of the Flash module. Therefore the register address of the FSR is the sector address SLOC (as defined above) where bit SE is sector specific and all other bits are identical within each sector.

FSR (*Sector Address*) **Reset Value: 0000_H**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SE	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**BUER**	**SQER**	**VPER**	**OPER**	**BRST**	**ERASE**	**PROG**	**BUSY**
r	-	-	-	-	-	-	-	r(w)	r(w)	r(w)	r(w)	r	r	r	r

Bit	Function
BUSY	**Flash Busy** (Summarizes the single busy bits) 0: Ready. Flash command execution is completed, module is in std. read mode. 1: Busy. Embedded algorithms for command execution are in progress.
PROG	**Programming State** 0: There is no programming operation in progress. 1: Flash busy with programming operation (store burst in operation).
ERASE	**Erase State** 0: There is no erase operation in progress. 1: Flash busy with erase operation.
BRST	**Burst Mode** 0: Flash not in burst mode. 1: Flash in burst mode, i.e. assembly register being filled. **Note:** Burst and read mode may occur concurrently.
OPER	**Operation Error** (Cleared via “Clear status” command) 0: Flash operation successfully terminated or currently running. 1: Flash array operation not successfully terminated, error in flash operation.
VPER	**Voltage Error** (Cleared via “Clear status” command) 0: No problem of programming voltage during Flash array operation. 1: Flash array operation not successful because of a prog. voltage problem.
SQER	**Command Sequence Error** (Cleared via “Clear status” command) 0: No command sequence error detected. 1: State machine operation aborted because of an illegal command sequence.
BUER	**Burst Error** (Cleared via “Clear status” command) 0: Burst operation successfully terminated or currently running. 1: Overflow or underload condition in burst mode detected.
SE	**Sector Erased** (Sector specific status bit) 0: Sector has not been erased since the last reset. 1: Sector has been successfully erased.

Note: The status register is a read-only register. Only the four error flags in the FSR are affected with the “clear status” command, indicated by “r(w)”.

Operation Control and Error Handling

Command execution is started with the last command of the respective command sequence and is indicated by the respective state flag (PROG for programming, ERASE for erasing) as well as by the summarizing BUSY flag. While polling BUSY is sufficient to detect the end of a command execution it is recommended to check the error flags afterwards so an aborted command can be detected.

The command execution should therefore use the following general structure:

- Write command sequence to Flash module
- Ensure correct sequence by checking bit SQER
- Poll BUSY to determine the command termination
- Check error flags OPER, VPER, BUER (whatever is appropriate)
- If error: clear flags via “Clear status” or “Reset” and act upon it (e.g. with a retry operation)

The table below gives examples of software actions to be taken after a specific error has been detected:

Software Reactions to Error Conditions

Detected Error	Fault Condition	Software Reaction
SQER Sequence Error	Wrong command/sector address, wrong command code, illegal command sequence	Check address or code and repeat with correct values.
OPER Operation Error	Aborted programming or erase operation due to SW or WDT reset	Repeat Flash operation.
VPER Voltage Error	Power supply failure	Compare data. Erase sector, if data is faulty. Repeat Flash operation. Note that previous blocks must be reprogrammed after a sector erase.
BUER Burst Error	Burst buffer overrun, burst buffer underload	Repeat load sequence with correct word count.

Reset Processing

Upon a CPU reset the Flash module resets its state machine and enters the standard read mode after the internal voltages have stabilized. The internal voltages need to ramp up (e.g. after power down) or to ramp down (e.g. after an interrupted programming or erase operation). This power stabilization phase is completed after maximum 100 s. The reset condition of CPU and Flash module is lengthened until power has stabilized.

Note: The lengthened reset condition is not reflected via pin $\overline{\text{RSTIN}}$ in bidirectional reset mode. The reset lengthening is disabled in case of an external start after reset.

External Host Mode Programming

In addition to the method described above the C163-16F's Flash module may also programmed by external programming devices.

External Host Mode (EHM) is enabled by selecting emulation mode (P0L.0='0') and also pulling low pin P0L.5. Pins P0L.5...0 represent '01 1110' in this case.

In External Host Mode the signals to control a programming cycle are generated by an external host using the C163-16F's bus interface. The external host provides the data to be programmed. The C163-16F itself is switched off and its Flash module can be accessed like standalone memory.

The following port pins represent the interface to the C163-16F's Flash module in EHM:

External Host Mode Interface Signals

Signal	Pin	Description
ADDR	P4.0, P1H.7 - P1L.1	Physical Flash word address (address line A0 is not evaluated).
DATA	P0H.7 - P0L.0	Word to be written or read
$\overline{RD}$	$\overline{RD}$	Read cycle control
$\overline{WR}$	$\overline{WR}$	Programming cycle control
$\overline{CE}$	P3.9	Flash enable signal
ALE	ALE	Address latch enable (indicates begin of bus cycle and is used for synchronization)
$\overline{RSTOUT}$	$\overline{RSTOUT}$	Must be held high (pullup resistor)

The access cycles generated by the external host must fulfill the timing requirements shown in the timing diagrams above.

Note: EHM is a varity of the emulation mode where pin P0.15 (P0L.7) is inverted during the reset configuration. This influences the selected clock generation mode.
For EHM operation direct drive or prescaler mode must be configured. If the on-chip oscillator is not supplied with a clock signal the oscillator watchdog must not be disabled, so the PLL can provide the clock signal instead.

Signal ALE is used for internal synchronization and for the state machine. Therefore the following rules must be obeyed:

- Min. 1 dummy read cycle after each command sequence.
- Min. 20 dummy read cycles after each program or erase command.
- Min. 2 dummy read cycles **before** each read status command.

The figure below shows typical external 100 ns access cycles (f_{CPU} = 20 MHz). Please note that $\overline{CE}$ must be activated at least 120 s before the 1st access cycle.

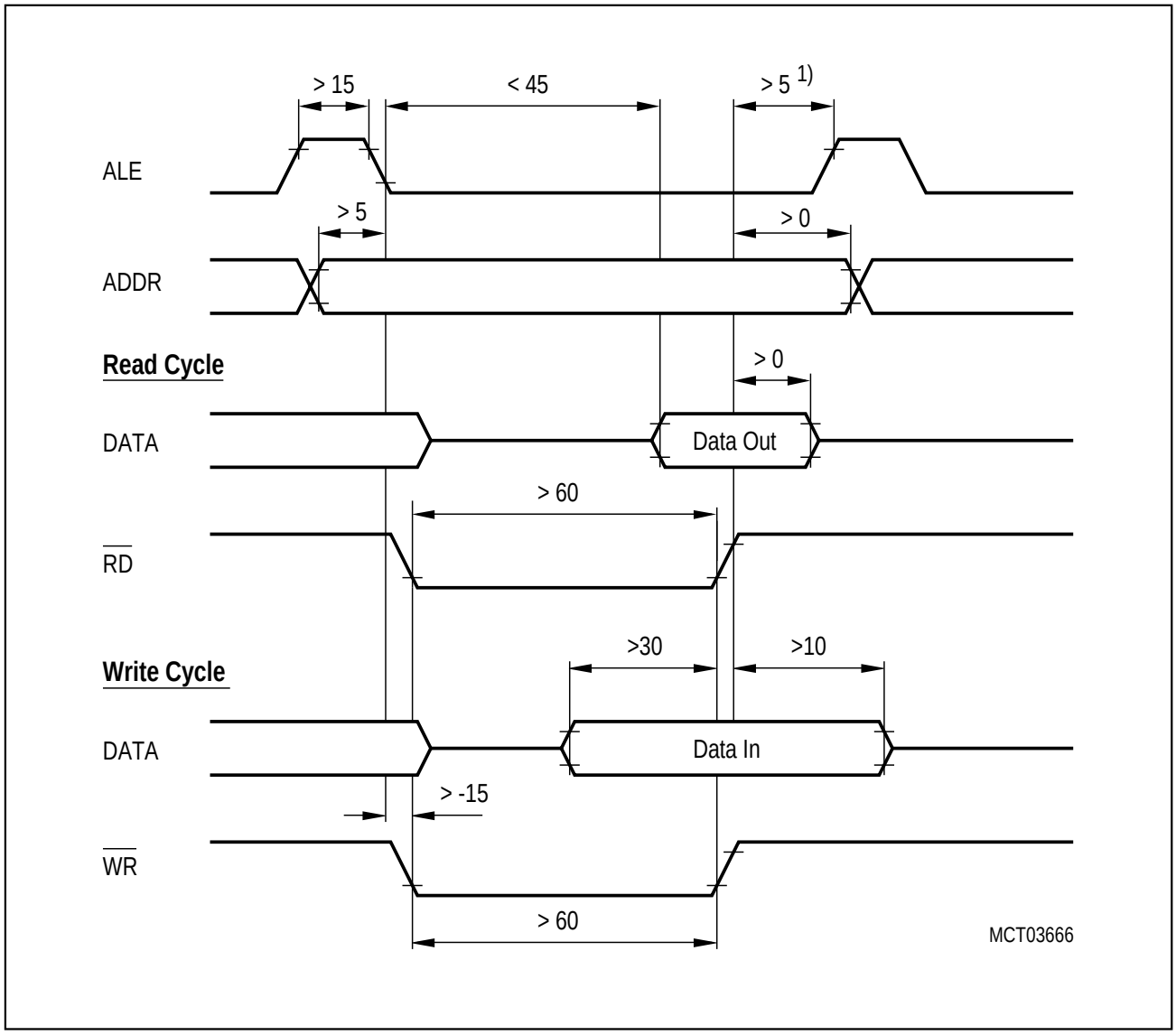


Figure 8
Asynchronous Flash Access Cycle in External Host Mode

1) This requirement exceeds the respective requirement for the standard XBUS.

Instruction Set Summary

The table below lists the instructions of the C163-16F 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 **“C16x Family Instruction Set Manual”**.

This document also provides a detailled 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
MOVBZ	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 und 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 C163-16F 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 (SSP) 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).

Special Function Registers Overview

Name	Physical Address	8-Bit Address	Description	Reset Value
ADDRSEL1	$FE18_H$	$0C_H$	Address Select Register 1	0000_H
ADDRSEL2	$FE1A_H$	$0D_H$	Address Select Register 2	0000_H
ADDRSEL3	$FE1C_H$	$0E_H$	Address Select Register 3	0000_H
ADDRSEL4	$FE1E_H$	$0F_H$	Address Select Register 4	0000_H
BUSCON0 b	$FF0C_H$	86_H	Bus Configuration Register 0	$0XX0_H$
BUSCON1 b	$FF14_H$	$8A_H$	Bus Configuration Register 1	0000_H
BUSCON2 b	$FF16_H$	$8B_H$	Bus Configuration Register 2	0000_H
BUSCON3 b	$FF18_H$	$8C_H$	Bus Configuration Register 3	0000_H
BUSCON4 b	$FF1A_H$	$8D_H$	Bus Configuration Register 4	0000_H
CAPREL	$FE4A_H$	25_H	GPT2 Capture/Reload Register	0000_H
CC8IC b	$FF88_H$	$C4_H$	EX0IN Interrupt Control Register	0000_H
CC9IC b	$FF8A_H$	$C5_H$	EX1IN Interrupt Control Register	0000_H
CC10IC b	$FF8C_H$	$C6_H$	EX2IN Interrupt Control Register	0000_H
CC11IC b	$FF8E_H$	$C7_H$	EX3IN Interrupt Control Register	0000_H
CC12IC b	$FF90_H$	$C8_H$	EX4IN Interrupt Control Register	0000_H
CC13IC b	$FF92_H$	$C9_H$	EX5IN Interrupt Control Register	0000_H
CC14IC b	$FF94_H$	CA_H	EX6IN Interrupt Control Register	0000_H
CC15IC b	$FF96_H$	CB_H	EX7IN Interrupt Control Register	0000_H
CP	$FE10_H$	08_H	CPU Context Pointer Register	$FC00_H$
CRIC b	$FF6A_H$	$B5_H$	GPT2 CAPREL Interrupt Control Register	0000_H
CSP	$FE08_H$	04_H	CPU Code Segment Pointer Register (read only)	0000_H

Special Function Registers Overview (cont'd)

Name		Physical Address		8-Bit Address	Description	Reset Value
DP0L	**b**	$F100_H$	**E**	80_H	P0L Direction Control Register	00_H
DP0H	**b**	$F102_H$	**E**	81_H	P0H Direction Control Register	00_H
DP1L	**b**	$F104_H$	**E**	82_H	P1L Direction Control Register	00_H
DP1H	**b**	$F106_H$	**E**	83_H	P1H Direction Control Register	00_H
DP2	**b**	$FFC2_H$		$E1_H$	Port 2 Direction Control Register	0000_H
DP3	**b**	$FFC6_H$		$E3_H$	Port 3 Direction Control Register	0000_H
DP4	**b**	$FFCA_H$		$E5_H$	Port 4 Direction Control Register	00_H
DP6	**b**	$FFCE_H$		$E7_H$	Port 6 Direction Control Register	00_H
DPP0		$FE00_H$		00_H	CPU Data Page Pointer 0 Register (10 bits)	0000_H
DPP1		$FE02_H$		01_H	CPU Data Page Pointer 1 Register (10 bits)	0001_H
DPP2		$FE04_H$		02_H	CPU Data Page Pointer 2 Register (10 bits)	0002_H
DPP3		$FE06_H$		03_H	CPU Data Page Pointer 3 Register (10 bits)	0003_H
EXICON	**b**	$F1C0_H$	**E**	$E0_H$	External Interrupt Control Register	0000_H
IDCHIP		$F07C_H$	**E**	$3E_H$	Identifier	$08XX_H$
IDMANUF		$F07E_H$	**E**	$3F_H$	Identifier	1820_H
IDMEM		$F07A_H$	**E**	$3D_H$	Identifier	3020_H
IDPROG		$F078_H$	**E**	$3C_H$	Identifier	4040_H
MDC	**b**	$FF0E_H$		87_H	CPU Multiply Divide Control Register	0000_H
MDH		$FE0C_H$		06_H	CPU Multiply Divide Register – High Word	0000_H
MDL		$FE0E_H$		07_H	CPU Multiply Divide Register – Low Word	0000_H
ODP2	**b**	$F1C2_H$	**E**	$E1_H$	Port 2 Open Drain Control Register	0000_H
ODP3	**b**	$F1C6_H$	**E**	$E3_H$	Port 3 Open Drain Control Register	0000_H
ODP6	**b**	$F1CE_H$	**E**	$E7_H$	Port 6 Open Drain Control Register	00_H
ONES		$FF1E_H$		$8F_H$	Constant Value 1's Register (read only)	$FFFF_H$
P0L	**b**	$FF00_H$		80_H	Port 0 Low Register (Lower half of PORT0)	00_H
P0H	**b**	$FF02_H$		81_H	Port 0 High Register (Upper half of PORT0)	00_H
P1L	**b**	$FF04_H$		82_H	Port 1 Low Register (Lower half of PORT1)	00_H
P1H	**b**	$FF06_H$		83_H	Port 1 High Register (Upper half of PORT1)	00_H
P2	**b**	$FFC0_H$		$E0_H$	Port 2 Register	0000_H

Special Function Registers Overview (cont'd)

Name		Physical Address		8-Bit Address	Description	Reset Value
P3	**b**	$FFC4_H$		$E2_H$	Port 3 Register	0000_H
P4	**b**	$FFC8_H$		$E4_H$	Port 4 Register (8 bits)	00_H
P5	**b**	$FFA2_H$		$D1_H$	Port 5 Register (read only)	$XXXX_H$
P6	**b**	$FFCC_H$		$E6_H$	Port 6 Register (8 bits)	00_H
PECC0		$FEC0_H$		60_H	PEC Channel 0 Control Register	0000_H
PECC1		$FEC2_H$		61_H	PEC Channel 1 Control Register	0000_H
PECC2		$FEC4_H$		62_H	PEC Channel 2 Control Register	0000_H
PECC3		$FEC6_H$		63_H	PEC Channel 3 Control Register	0000_H
PECC4		$FEC8_H$		64_H	PEC Channel 4 Control Register	0000_H
PECC5		$FECA_H$		65_H	PEC Channel 5 Control Register	0000_H
PECC6		$FECC_H$		66_H	PEC Channel 6 Control Register	0000_H
PECC7		$FECE_H$		67_H	PEC Channel 7 Control Register	0000_H
PSW	**b**	$FF10_H$		88_H	CPU Program Status Word	0000_H
RP0H	**b**	$F108_H$	**E**	84_H	System Startup Configuration Register (Rd. only)	XX_H
S0BG		$FEB4_H$		$5A_H$	Serial Channel 0 Baud Rate Generator Reload Register	0000_H
S0CON	**b**	$FFB0_H$		$D8_H$	Serial Channel 0 Control Register	0000_H
S0EIC	**b**	$FF70_H$		$B8_H$	Serial Channel 0 Error Interrupt Control Register	0000_H
S0RBUF		$FEB2_H$		59_H	Serial Channel 0 Receive Buffer Register (read only)	XX_H
S0RIC	**b**	$FF6E_H$		$B7_H$	Serial Channel 0 Receive Interrupt Control Register	0000_H
S0TBIC	**b**	$F19C_H$	**E**	CE_H	Serial Channel 0 Transmit Buffer Interrupt Control Register	0000_H
S0TBUF		$FEB0_H$		58_H	Serial Channel 0 Transmit Buffer Register (write only)	00_H
S0TIC	**b**	$FF6C_H$		$B6_H$	Serial Channel 0 Transmit Interrupt Control Register	0000_H
SP		$FE12_H$		09_H	CPU System Stack Pointer Register	$FC00_H$
SSPCON0		$EF00_H$	**X**	---	SSP Control Register 0	0000_H
SSPCON1		$EF02_H$	**X**	---	SSP Control Register 1	0000_H

Special Function Registers Overview (cont'd)

Name		Physical Address		8-Bit Address	Description	Reset Value
SSPRTB		$EF04_H$	**X**	---	SSP Receive/Transmit Buffer	$XXXX_H$
SSPTBH		$EF06_H$	**X**	---	SSP Transmit Buffer High	$XXXX_H$
STKOV		$FE14_H$		$0A_H$	CPU Stack Overflow Pointer Register	$FA00_H$
STKUN		$FE16_H$		$0B_H$	CPU Stack Underflow Pointer Register	$FC00_H$
SYSCON	**b**	$FF12_H$		89_H	CPU System Configuration Register	$0XX0_H$ [1]
T2		$FE40_H$		20_H	GPT1 Timer 2 Register	0000_H
T2CON	**b**	$FF40_H$		$A0_H$	GPT1 Timer 2 Control Register	0000_H
T2IC	**b**	$FF60_H$		$B0_H$	GPT1 Timer 2 Interrupt Control Register	0000_H
T3		$FE42_H$		21_H	GPT1 Timer 3 Register	0000_H
T3CON	**b**	$FF42_H$		$A1_H$	GPT1 Timer 3 Control Register	0000_H
T3IC	**b**	$FF62_H$		$B1_H$	GPT1 Timer 3 Interrupt Control Register	0000_H
T4		$FE44_H$		22_H	GPT1 Timer 4 Register	0000_H
T4CON	**b**	$FF44_H$		$A2_H$	GPT1 Timer 4 Control Register	0000_H
T4IC	**b**	$FF64_H$		$B2_H$	GPT1 Timer 4 Interrupt Control Register	0000_H
T5		$FE46_H$		23_H	GPT2 Timer 5 Register	0000_H
T5CON	**b**	$FF46_H$		$A3_H$	GPT2 Timer 5 Control Register	0000_H
T5IC	**b**	$FF66_H$		$B3_H$	GPT2 Timer 5 Interrupt Control Register	0000_H
T6		$FE48_H$		24_H	GPT2 Timer 6 Register	0000_H
T6CON	**b**	$FF48_H$		$A4_H$	GPT2 Timer 6 Control Register	0000_H
T6IC	**b**	$FF68_H$		$B4_H$	GPT2 Timer 6 Interrupt Control Register	0000_H
TFR	**b**	$FFAC_H$		$D6_H$	Trap Flag Register	0000_H
WDT		$FEAE_H$		57_H	Watchdog Timer Register (read only)	0000_H
WDTCON		$FFAE_H$		$D7_H$	Watchdog Timer Control Register	$000X_H$ [2]
XP1IC	**b**	$F18E_H$	**E**	$C7_H$	SSP Interrupt Control Register	0000_H
XP3IC	**b**	$F19E_H$	**E**	CF_H	PLL/OWD Interrupt Control Register	0000_H
ZEROS	**b**	$FF1C_H$		$8E_H$	Constant Value 0's Register (read only)	0000_H

[1] The system configuration is selected during reset.

[2] Bit WDTR indicates a watchdog timer triggered reset.

Absolute Maximum Ratings

Ambient temperature under bias (T_A):
SAB-C163-16F-LF, SAB-C163-16F-L25F ..... 0 to + 70 °C
Storage temperature (T_{ST}) ..... – 65 to + 150 °C
Voltage on V_{CC} 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_{CC} + 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 overload conditions (V_{IN}>V_{CC} or V_{IN}<V_{SS}) the voltage on 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 C163-16F and partly its demands on the system. To aid in interpreting the parameters right, when evaluating them for a design, they are marked in column “Symbol”:

CC (**C**ontroller **C**haracteristics):
The logic of the C163-16F will provide signals with the respective timing characteristics.

SR (**S**ystem **R**equirement):
The external system must provide signals with the respective timing characteristics to the C163-16F.

DC Characteristics

V_{CC} = 5 V ± 10 %; V_{SS} = 0 V
T_A = 0 to +70 ˚C for SAB-C163-16F-LF, SAB-C163-16F-L25F

Parameter	Symbol		Limit Values		Unit	Test Condition
			min.	max.		
Input low voltage	V_{IL}	SR	– 0.5	0.2 V_{CC} – 0.1	V	–
Input high voltage (all except $\overline{\text{RSTIN}}$ and XTAL1)	V_{IH}	SR	0.2 V_{CC} + 0.9	V_{CC} + 0.5	V	–
Input high voltage $\overline{\text{RSTIN}}$	V_{IH1}	SR	0.6 V_{CC}	V_{CC} + 0.5	V	–
Input high voltage XTAL1	V_{IH2}	SR	0.7 V_{CC}	V_{CC} + 0.5	V	–
Output low voltage (PORT0, PORT1, Port 4, ALE, $\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{BHE}}$, CLKOUT, $\overline{\text{RSTOUT}}$)	V_{OL}	CC	–	0.45	V	I_{OL} = 2.4 mA
Output low voltage (all other outputs)	V_{OL1}	CC	–	0.45	V	I_{OL1} = 1.6 mA
Output high voltage (PORT0, PORT1, Port 4, ALE, $\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{BHE}}$, CLKOUT, $\overline{\text{RSTOUT}}$)	V_{OH}	CC	0.9 V_{CC} 2.4	–	V V	I_{OH} = – 500 A I_{OH} = – 2.4 mA
Output high voltage [1] (all other outputs)	V_{OH1}	CC	0.9 V_{CC} 2.4	–	V V	I_{OH} = – 250 A I_{OH} = – 1.6 mA
Input leakage current (Port 5)	I_{OZ1}	CC	–	± 200	nA	0 V < V_{IN} < V_{CC}
Input leakage current (all other)	I_{OZ2}	CC	–	± 500	nA	0 V < V_{IN} < V_{CC}
Overload current	I_{OV}	SR	–	± 5	mA	[2] [3]
$\overline{\text{RSTIN}}$ pullup resistor	R_{RST}	CC	50	150	kΩ	–
Read/Write inactive current [4]	I_{RWH}	[5]	–	– 40	A	V_{OUT} = 2.4 V
Read/Write active current [4]	I_{RWL}	[6]	-500	–	A	V_{OUT} = V_{OLmax}
ALE inactive current [4]	I_{ALEL}	[5]	–	40	A	V_{OUT} = V_{OLmax}
ALE active current [4]	I_{ALEH}	[6]	500	–	A	V_{OUT} = 2.4 V
Port 6 inactive current [4]	I_{P6H}	[5]	–	-40	A	V_{OUT} = 2.4 V
Port 6 active current [4]	I_{P6L}	[6]	-500	–	A	V_{OUT} = V_{OL1max}
PORT0 configuration current [4]	I_{P0H}	[5]	–	– 10	A	V_{IN} = V_{IHmin}
	I_{P0L}	[6]	-100	–	A	V_{IN} = V_{ILmax}
XTAL1 input current	I_{IL}	CC	–	± 20	A	0 V < V_{IN} < V_{CC}
Pin capacitance [3] (digital inputs/outputs)	C_{IO}	CC	–	10	pF	f = 1 MHz T_A = 25 ˚C

Parameter	Symbol	Limit Values		Unit	Test Condition
		min.	max.		
Power supply current	I_{CC}	–	10 + 3.5 * f_{CPU}	mA	$\overline{RSTIN}$ = V_{IL2} f_{CPU} in [MHz] 7)
Idle mode supply current	I_{ID}	–	2 + 1.1 * f_{CPU}	mA	$\overline{RSTIN}$ = V_{IH1} f_{CPU} in [MHz] 7)
Power-down mode supply current	I_{PD}	–	50	A	V_{CC} = 5.5 V 8)

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) Overload conditions occur if the standard operatings conditions are exceeded, i.e. the voltage on any pin exceeds the specified range (i.e. $V_{OV} > V_{CC}$+0.5V, except pin OWE, or $V_{OV} < V_{SS}$-0.5V). The absolute sum of input overload currents on all port pins may not exceed **50 mA**. The supply voltage must remain within the specified limits.

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

4) This specification is only valid during Reset, or during Hold- or Adapt-mode. Port 6 pins are only affected, if they are used for $\overline{CS}$ output and the open drain function is not enabled.

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

6) The minimum current must be drawn in order to drive the respective signal line active.

7) The supply current is a function of the operating frequency. This dependency is illustrated in the figure below. These parameters are tested at V_{CCmax} and maximum CPU clock with all outputs disconnected and all inputs at V_{IL} or V_{IH}.
Program execution out of the internal Flash memory adds 20 mA (typically) to the current consumption in active mode (see also figure below).

8) This parameter is tested including leakage currents. All inputs (including pins configured as inputs) at 0 V to 0.1 V or at V_{CC} – 0.1 V to V_{CC}, V_{REF} = 0 V, all outputs (including pins configured as outputs) disconnected.

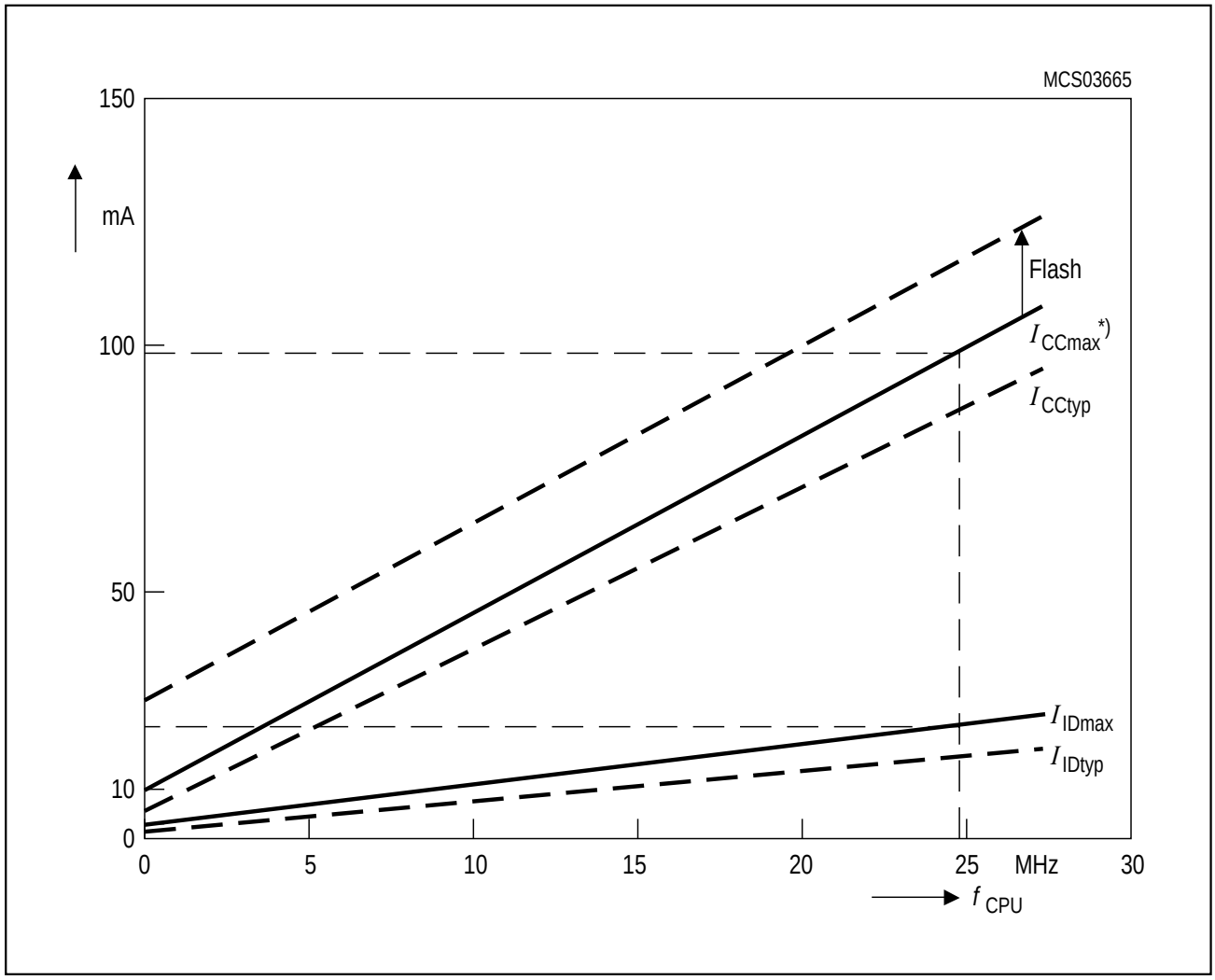
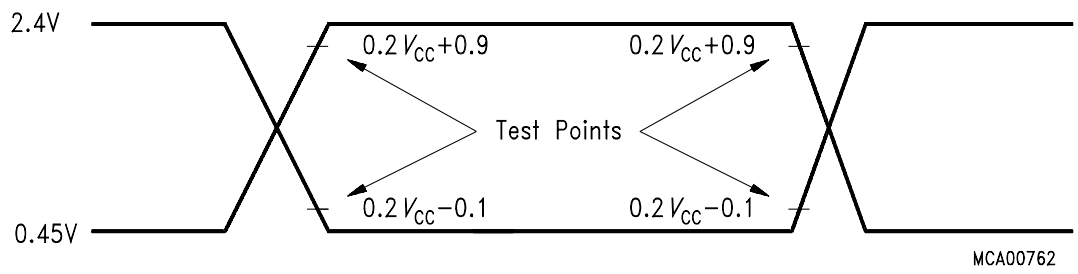


Figure 9
Supply/Idle Current as a Function of Operating Frequency

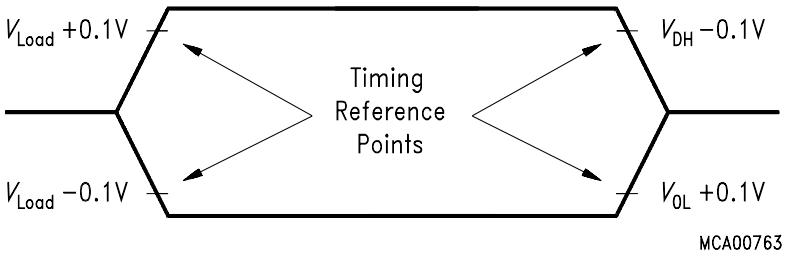
Note: *) The indicated I_{CCmax} is tested under reset conditions. Program execution out of the internal Flash memory adds 20 mA (typically) to the current consumption in active mode.

Testing Waveforms



AC inputs during testing are driven at 2.4 V for a logic ‘1’ and 0.4 V for a logic ‘0’. Timing measurements are made at V_{IH} min for a logic ‘1’ and V_{IL} max for a logic ‘0’.

Figure 10
Input Output Waveforms



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

Figure 11
Float Waveforms

AC Characteristics
Definition of Internal Timing

The internal operation of the C163-16F 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).

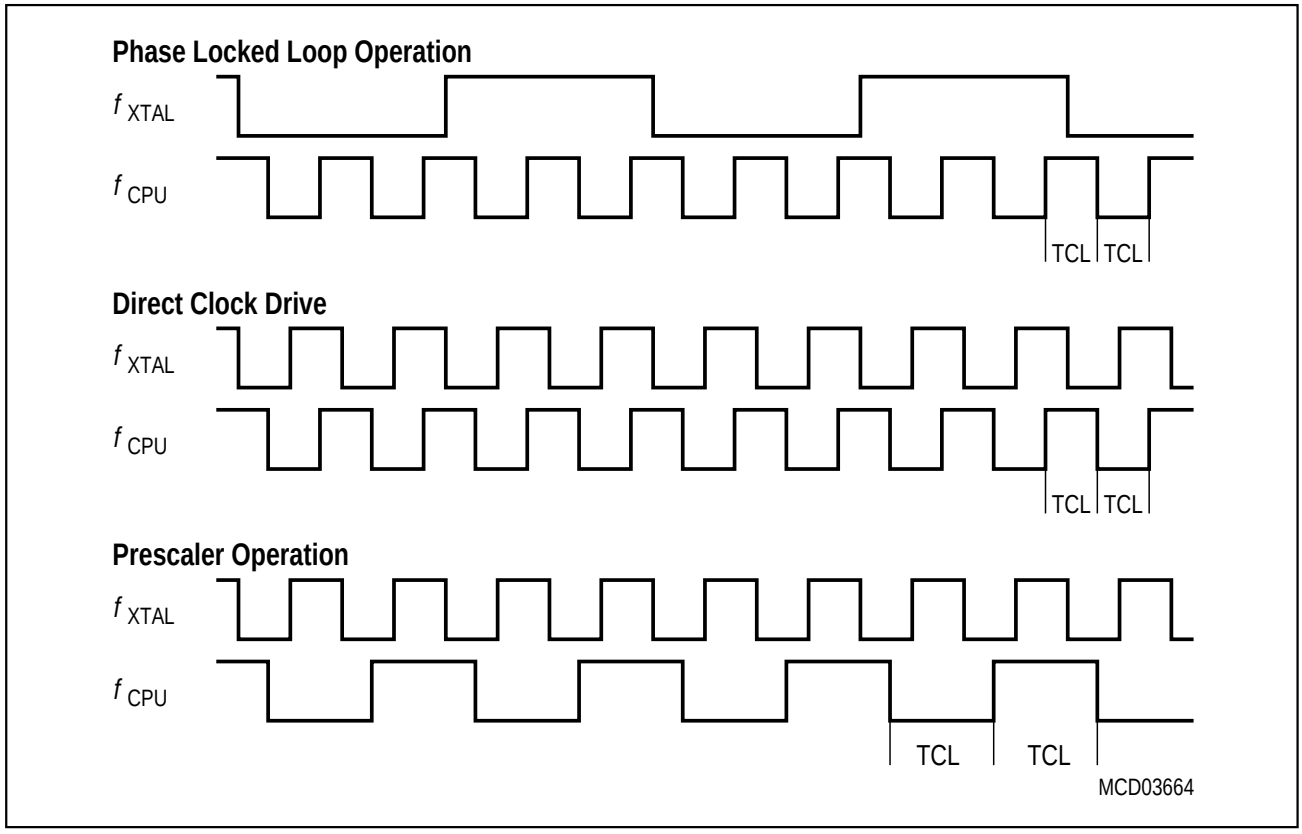


Figure 12
Generation Mechanisms for the CPU Clock

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 C163-16F.

Note: The example for PLL operation shown in the figure above refers to a PLL factor of 4.

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.

C163-16F Clock Generation Modes

P0.15-13 (P0H.7-5)			CPU Frequency $f_{CPU} = f_{XTAL} * F$	External Clock Input Range [1]	Notes
1	1	1	$f_{XTAL} * 4$	2.5 to 6.25 MHz	Default configuration
1	1	0	$f_{XTAL} * 3$	3.33 to 8.33 MHz	
1	0	1	$f_{XTAL} * 2$	5 to 12.5 MHz	
1	0	0	$f_{XTAL} * 5$	2 to 5 MHz	
0	1	1	$f_{XTAL} * 1$	1 to 25 MHz	Direct drive [2]
0	1	0	$f_{XTAL} * 1.5$	6.66 to 16.6 MHz	
0	0	1	$f_{XTAL} / 2$	2 to 50 MHz	CPU clock via prescaler
0	0	0	$f_{XTAL} * 2.5$	4 to 10 MHz	

[1] The external clock input range refers to a CPU clock range of 10...25 MHz.
[2] The maximum 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_{CPU} is half the frequency of f_{XTAL} and the high and low time of f_{CPU} (i.e. the duration of an individual TCL) is defined by the period of the input clock f_{XTAL}.

The timings listed in the AC Characteristics that refer to TCLs therefore can be calculated using the period of f_{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_{CPU} directly follows the frequency of f_{XTAL} so the high and low time of f_{CPU} (i.e. the duration of an individual TCL) is defined by the duty cycle of the input clock f_{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:

$$TCL_{min} = 1/f_{XTAL} * DC_{min} \qquad \text{(DC = duty cycle)}$$

For two consecutive TCLs the deviation caused by the duty cycle of f_{XTAL} is compensated so the duration of 2TCL is always $1/f_{XTAL}$. The minimum value TCL_{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 $2TCL = 1/f_{XTAL}$.

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

Phase Locked Loop

For all other combinations of pins P0.15-13 (P0H.7-5) during reset the on-chip phase locked loop is enabled and provides the CPU clock (see table above). The PLL multiplies the input frequency by the factor **F** which is selected via the combination of pins P0.15-13 (i.e. $f_{CPU} = f_{XTAL} * \mathbf{F}$). With every **F**'th transition of f_{XTAL} the PLL circuit synchronizes the CPU clock to the input clock. This synchronization is done smoothely, i.e. the CPU clock frequency does not change abruptly.

Due to this adaptation to the input clock the frequency of f_{CPU} is constantly adjusted so it is locked to f_{XTAL}. The slight variation causes a jitter of f_{CPU} which also effects the duration of individual TCLs.

The timings listed in the AC Characteristics that refer to TCLs therefore must be calculated using the minimum TCL that is possible under the respective circumstances.

The actual minimum value for TCL depends on the jitter of the PLL. As the PLL is constantly adjusting its output frequency so it corresponds to the applied input frequency (crystal or oscillator) the relative deviation for periods of more than one TCL is lower than for one single TCL (see formula and figure below).
For a period of $\boldsymbol{N}$ * TCL the minimum value is computed using the corresponding deviation $D_{\boldsymbol{N}}$:

$$TCL_{min} = TCL_{NOM} * (1 - D_{\boldsymbol{N}} / 100)$$

$D_{\boldsymbol{N}} = \pm(4 - \boldsymbol{N}/15)$ [%],
where $\boldsymbol{N}$ = number of consecutive TCLs
and $1 \le \boldsymbol{N} \le 40$.

So for a period of 3 TCLs (i.e. $\boldsymbol{N} = 3$): $D_{\boldsymbol{3}} = 4 - \boldsymbol{3}/15 = 3.8\%$,
and $(3TCL)_{min} = 3TCL_{NOM} * (1 - 3.8 / 100) = 3TCL_{NOM} * 0.962$ (57.72 nsec @ f_{CPU} = 25 MHz).

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

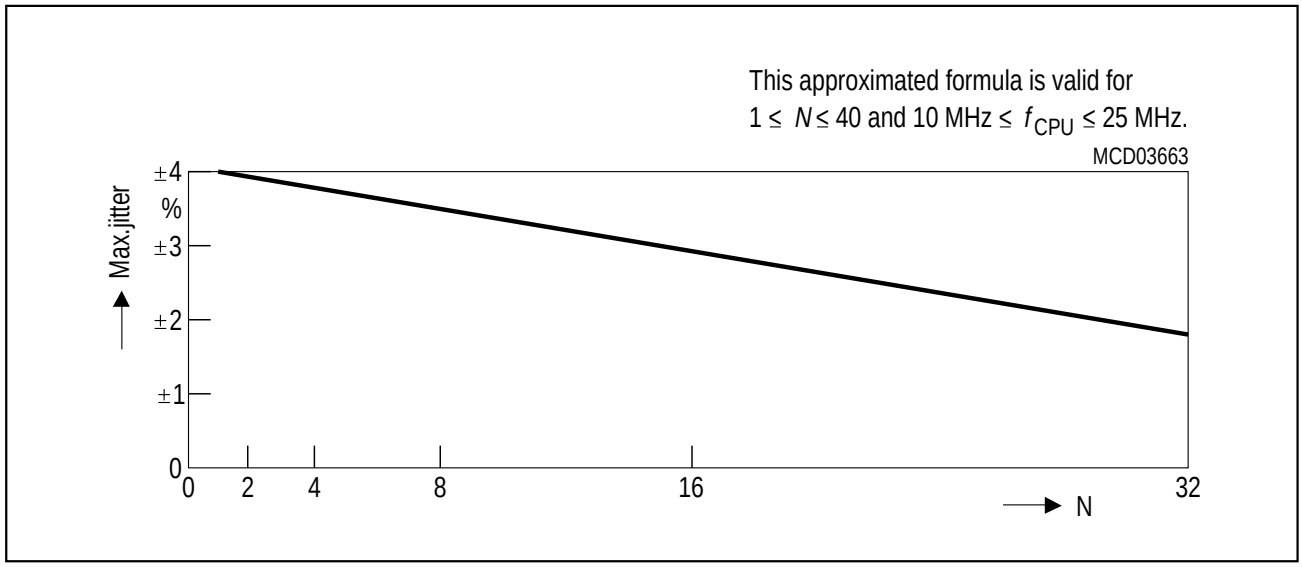


Figure 13
Approximated Maximum PLL Jitter

AC Characteristics
External Clock Drive XTAL1

V_{CC} = 5 V ± 10 %; V_{SS} = 0 V
T_A = 0 to +70 ˚C for SAB-C163-16F-LF, SAB-C163-16F-L25F

Parameter	Symbol		Direct Drive 1:1		Prescaler 2:1		PLL 1:N		Unit
			min.	max.	min.	max.	min.	max.	
Oscillator period	t_{OSC}	SR	40	1000	20	500	60 [1]	500 [1]	ns
High time	t_1	SR	18 [2]	–	6	–	10	–	ns
Low time	t_2	SR	18 [2]	–	6	–	10	–	ns
Rise time	t_3	SR	–	10 [2]	–	6 [2]	–	10 [2]	ns
Fall time	t_4	SR	–	10 [2]	–	6 [2]	–	10 [2]	ns

1) The minimum and maximum oscillator periods for PLL operation depend on the selected CPU clock generation mode. Please see respective table above.
2) The clock input signal must reach the defined levels V_{IL} and V_{IH2}.

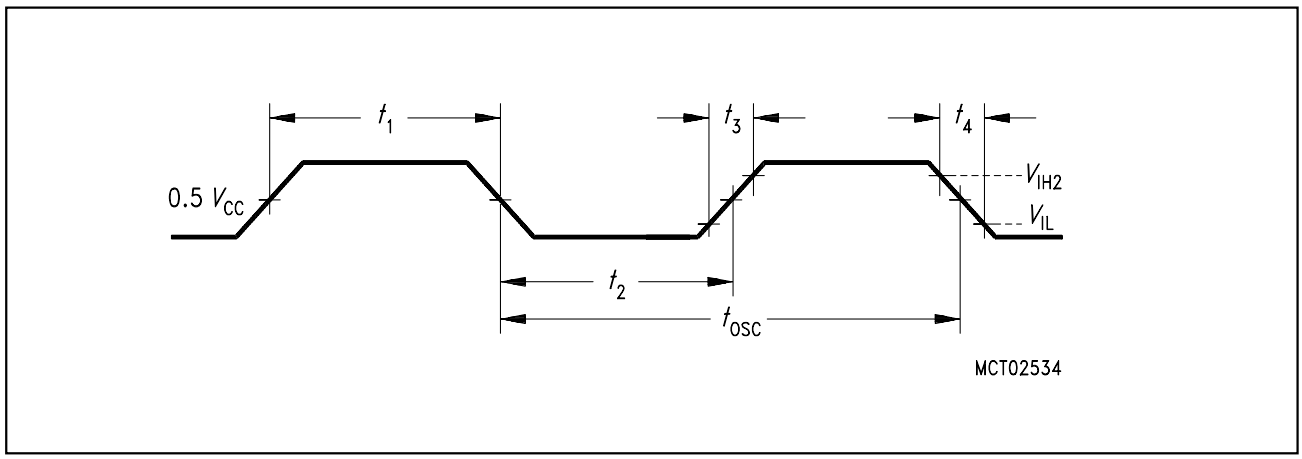


Figure 14
External Clock Drive XTAL1

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 * <ALECTL>
Memory Cycle Time Waitstates	t_C	2TCL * (15 - <MCTC>)
Memory Tristate Time	t_F	2TCL * (1 - <MTTC>)

AC Characteristics
Multiplexed Bus

V_{CC} = 5 V ± 10 %; V_{SS} = 0 V
T_A = 0 to +70 ˚C for SAB-C163-16F-LF, SAB-C163-16F-L25F
C_L (for PORT0, PORT1, Port 4, ALE, $\overline{RD}$, $\overline{WR}$, $\overline{BHE}$, CLKOUT) = 100 pF
C_L (for Port 6, $\overline{CS}$) = 100 pF
ALE cycle time = 6 TCL + 2t_A + t_C + t_F (120 ns at 25 MHz CPU clock without waitstates)

Parameter	Symbol		Max. CPU Clock = 25 MHz		Variable CPU Clock 1 / 2TCL = 1 to 25 MHz		Unit
			min.	max.	min.	max.	
ALE high time	t_5	CC	10 + t_A	–	TCL - 10 + t_A	–	ns
Address setup to ALE	t_6	CC	4 + t_A	–	TCL - 16 + t_A	–	ns
Address hold after ALE	t_7	CC	10 + t_A	–	TCL - 10 + t_A	–	ns
ALE falling edge to $\overline{RD}$, $\overline{WR}$ (with RW-delay)	t_8	CC	10 + 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	–	26	–	TCL + 6	ns
$\overline{RD}$, $\overline{WR}$ low time (with RW-delay)	t_{12}	CC	30 + t_C	–	2TCL - 10 + t_C	–	ns
$\overline{RD}$, $\overline{WR}$ low time (no RW-delay)	t_{13}	CC	50 + t_C	–	3TCL - 10 + t_C	–	ns
$\overline{RD}$ to valid data in (with RW-delay)	t_{14}	SR	–	20 + t_C	–	2TCL - 20 + t_C	ns
$\overline{RD}$ to valid data in (no RW-delay)	t_{15}	SR	–	40 + t_C	–	3TCL - 20 + t_C	ns
ALE low to valid data in	t_{16}	SR	–	40 + t_A + t_C	–	3TCL - 20 + t_A + t_C	ns
Address to valid data in	t_{17}	SR	–	50 + 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	–	26 + t_F	–	2TCL - 14 + t_F	ns

Parameter	Symbol		Max. CPU Clock = 25 MHz		Variable CPU Clock 1 / 2TCL = 1 to 25 MHz		Unit
			min.	max.	min.	max.	
Data valid to $\overline{WR}$	t_{22}	CC	20 + t_C	–	2TCL - 20 + t_C	–	ns
Data hold after $\overline{WR}$	t_{23}	CC	26 + t_F	–	2TCL - 14 + t_F	–	ns
ALE rising edge after $\overline{RD}$, $\overline{WR}$	t_{25}	CC	26 + t_F	–	2TCL - 14 + t_F	–	ns
Address hold after $\overline{RD}$, $\overline{WR}$	t_{27}	CC	26 + 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	–	40 + t_C + 2t_A	–	3TCL - 20 + t_C+2t_A	ns
$\overline{CS}$ hold after $\overline{RD}$, $\overline{WR}$	t_{40}	CC	46 + t_F	–	3TCL - 14 + t_F	–	ns
ALE fall. edge to $\overline{RdCS}$, $\overline{WrCS}$ (with RW delay)	t_{42}	CC	16 + 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	–	20	–	TCL	ns
$\overline{RdCS}$ to Valid Data In (with RW delay)	t_{46}	SR	–	16 + t_C	–	2TCL - 24 + t_C	ns
$\overline{RdCS}$ to Valid Data In (no RW delay)	t_{47}	SR	–	36 + t_C	–	3TCL - 24 + t_C	ns
$\overline{RdCS}$, $\overline{WrCS}$ Low Time (with RW delay)	t_{48}	CC	30 + t_C	–	2TCL - 10 + t_C	–	ns
$\overline{RdCS}$, $\overline{WrCS}$ Low Time (no RW delay)	t_{49}	CC	50 + t_C	–	3TCL - 10 + t_C	–	ns
Data valid to $\overline{WrCS}$	t_{50}	CC	26 + 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	–	20 + t_F	–	2TCL - 20 + t_F	ns

Parameter	Symbol		Max. CPU Clock = 25 MHz		Variable CPU Clock 1 / 2TCL = 1 to 25 MHz		Unit
			min.	max.	min.	max.	
Address hold after $\overline{\text{RdCS}}$, $\overline{\text{WrCS}}$	t_{54}	CC	20 + t_F	–	2TCL - 20 + t_F	–	ns
Data hold after $\overline{\text{WrCS}}$	t_{56}	CC	20 + t_F	–	2TCL - 20 + t_F	–	ns

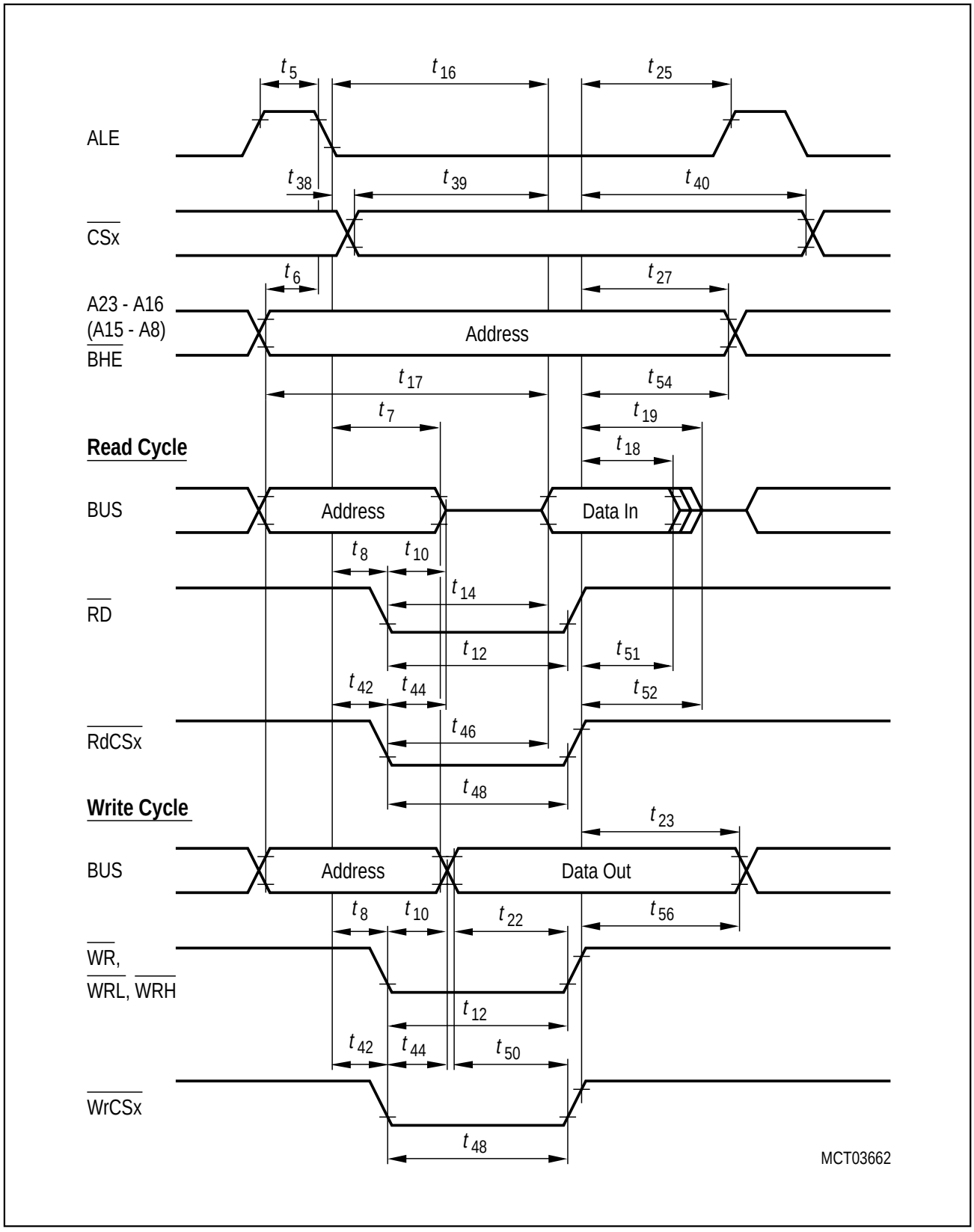


Figure 15-1
External Memory Cycle: Multiplexed Bus, With Read/Write Delay, Normal ALE

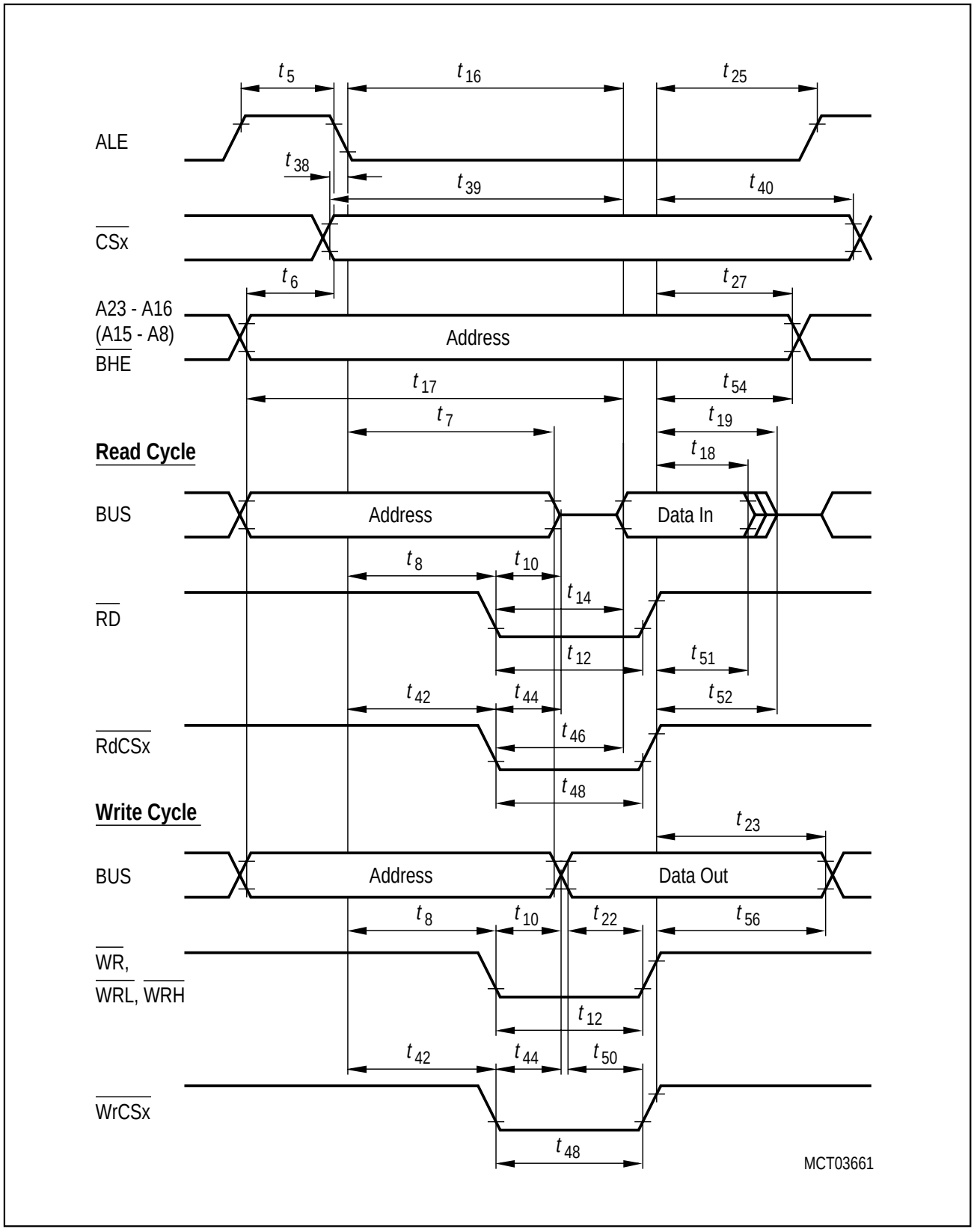


Figure 15-2
External Memory Cycle: Multiplexed Bus, With Read/Write Delay, Extended ALE

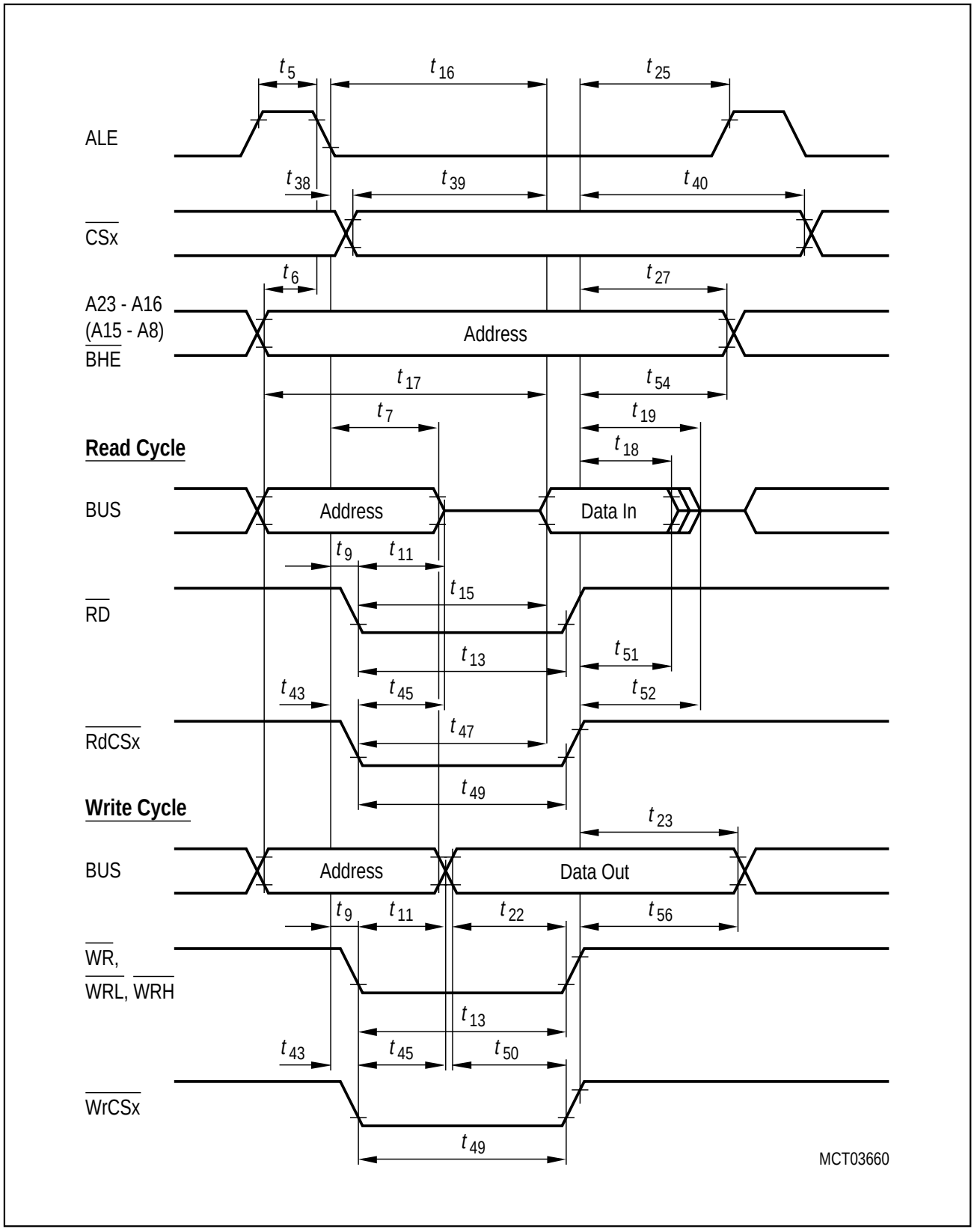


Figure 15-3
External Memory Cycle: Multiplexed Bus, No Read/Write Delay, Normal ALE

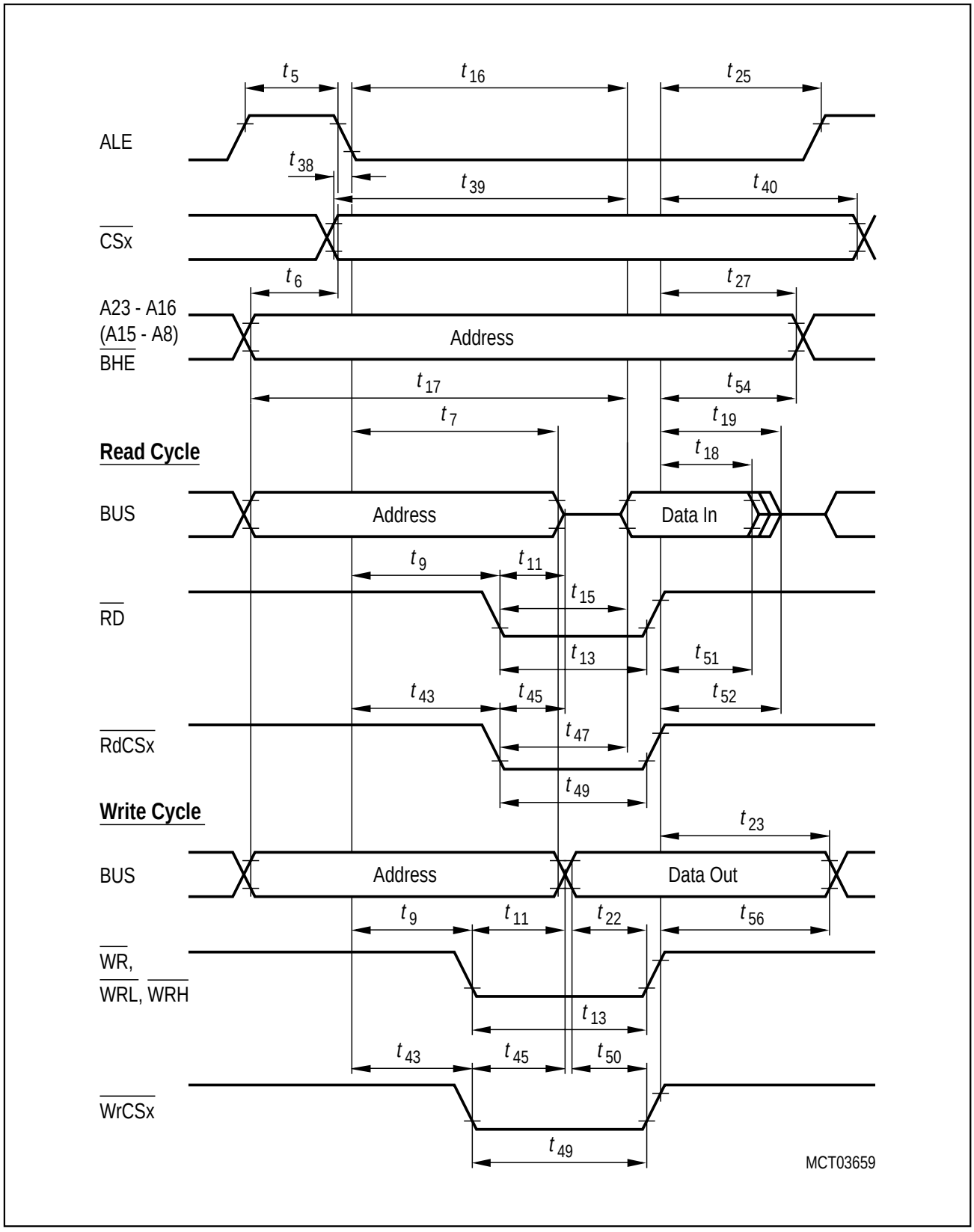


Figure 15-4
External Memory Cycle: Multiplexed Bus, No Read/Write Delay, Extended ALE

AC Characteristics
Demultiplexed Bus

V_{CC} = 5 V ± 10 %; V_{SS} = 0 V
T_A = 0 to +70 ˚C for SAB-C163-16F-LF, SAB-C163-16F-L25F
C_L (for PORT0, PORT1, Port 4, ALE, $\overline{RD}$, $\overline{WR}$, $\overline{BHE}$, CLKOUT) = 100 pF
C_L (for Port 6, $\overline{CS}$) = 100 pF
ALE cycle time = 4 TCL + 2t_A + t_C + t_F (80 ns at 25 MHz CPU clock without waitstates)

Parameter	Symbol		Max. CPU Clock = 25 MHz		Variable CPU Clock 1 / 2TCL = 1 to 25 MHz		Unit
			min.	max.	min.	max.	
ALE high time	t_5	CC	10 + t_A	–	TCL - 10 + t_A	–	ns
Address setup to ALE	t_6	CC	4 + t_A	–	TCL - 16 + t_A	–	ns
ALE falling edge to $\overline{RD}$, $\overline{WR}$ (with RW-delay)	t_8	CC	10 + 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
$\overline{RD}$, $\overline{WR}$ low time (with RW-delay)	t_{12}	CC	30 + t_C	–	2TCL - 10 + t_C	–	ns
$\overline{RD}$, $\overline{WR}$ low time (no RW-delay)	t_{13}	CC	50 + t_C	–	3TCL - 10 + t_C	–	ns
$\overline{RD}$ to valid data in (with RW-delay)	t_{14}	SR	–	20 + t_C	–	2TCL - 20 + t_C	ns
$\overline{RD}$ to valid data in (no RW-delay)	t_{15}	SR	–	40 + t_C	–	3TCL - 20 + t_C	ns
ALE low to valid data in	t_{16}	SR	–	40 + t_A + t_C	–	3TCL - 20 + t_A + t_C	ns
Address to valid data in	t_{17}	SR	–	50 + 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 $\overline{RD}$ rising edge (with RW-delay [1])	t_{20}	SR	–	26 + t_F	–	2TCL - 14 + 2t_A + t_F [1]	ns
Data float after $\overline{RD}$ rising edge (no RW-delay [1])	t_{21}	SR	–	10 + t_F	–	TCL - 10 + 2t_A + t_F [1]	ns
Data valid to $\overline{WR}$	t_{22}	CC	20 + t_C	–	2TCL - 20 + t_C	–	ns
Data hold after $\overline{WR}$	t_{24}	CC	10 + t_F	–	TCL - 10 + t_F	–	ns

Parameter	Symbol		Max. CPU Clock = 25 MHz		Variable CPU Clock 1 / 2TCL = 1 to 25 MHz		Unit
			min.	max.	min.	max.	
ALE rising edge after $\overline{RD}$, $\overline{WR}$	t_{26}	CC	-10 + t_F	–	-10 + t_F	–	ns
Address hold after $\overline{WR}$ 2)	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	–	40 + t_C + 2t_A	–	3TCL - 20 + t_C+2t_A	ns
$\overline{CS}$ hold after $\overline{RD}$, $\overline{WR}$	t_{41}	CC	6 + t_F	–	TCL - 14 + t_F	–	ns
ALE falling edge to $\overline{RdCS}$, $\overline{WrCS}$ (with RW-delay)	t_{42}	CC	16 + 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	–	16 + t_C	–	2TCL - 24 + t_C	ns
$\overline{RdCS}$ to Valid Data In (no RW-delay)	t_{47}	SR	–	36 + t_C	–	3TCL - 24 + t_C	ns
$\overline{RdCS}$, $\overline{WrCS}$ Low Time (with RW-delay)	t_{48}	CC	30 + t_C	–	2TCL - 10 + t_C	–	ns
$\overline{RdCS}$, $\overline{WrCS}$ Low Time (no RW-delay)	t_{49}	CC	50 + t_C	–	3TCL - 10 + t_C	–	ns
Data valid to $\overline{WrCS}$	t_{50}	CC	26 + 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	–	20 + t_F	–	2TCL - 20 + t_F	ns
Data float after $\overline{RdCS}$ (no RW-delay)	t_{68}	SR	–	0 + t_F	–	TCL - 20 + t_F	ns
Address hold after $\overline{RdCS}$, $\overline{WrCS}$	t_{55}	CC	-6 + t_F	–	-6 + t_F	–	ns
Data hold after $\overline{WrCS}$	t_{57}	CC	6 + t_F	–	TCL - 14 + t_F	–	ns

1) RW-delay and t_A refer to the next following bus cycle.

2) Read data are latched with the same clock edge that triggers the address change and the rising $\overline{RD}$ edge. Therefore address changes before the end of $\overline{RD}$ have no impact on read cycles.

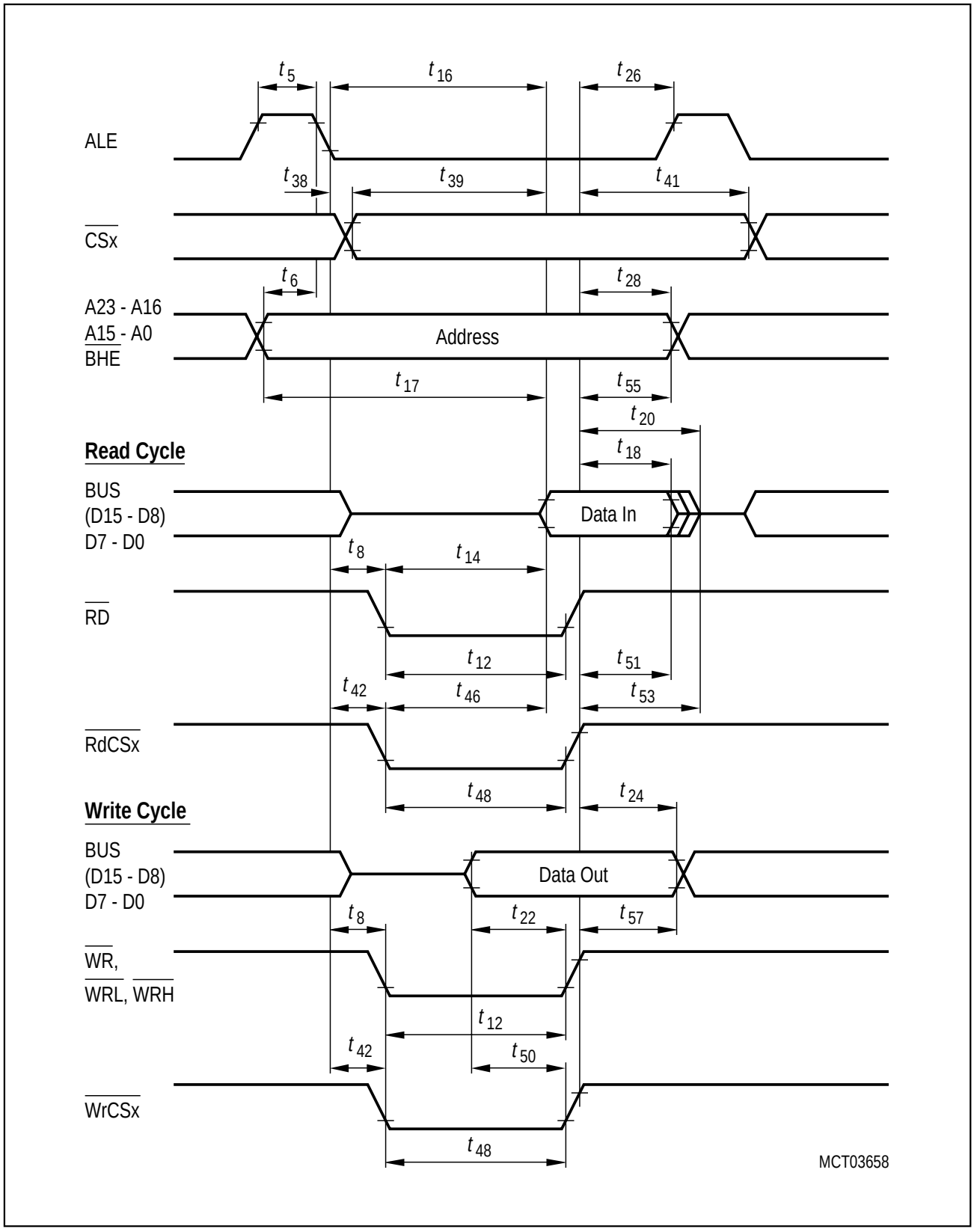


Figure 16-1
External Memory Cycle: Demultiplexed Bus, With Read/Write Delay, Normal ALE

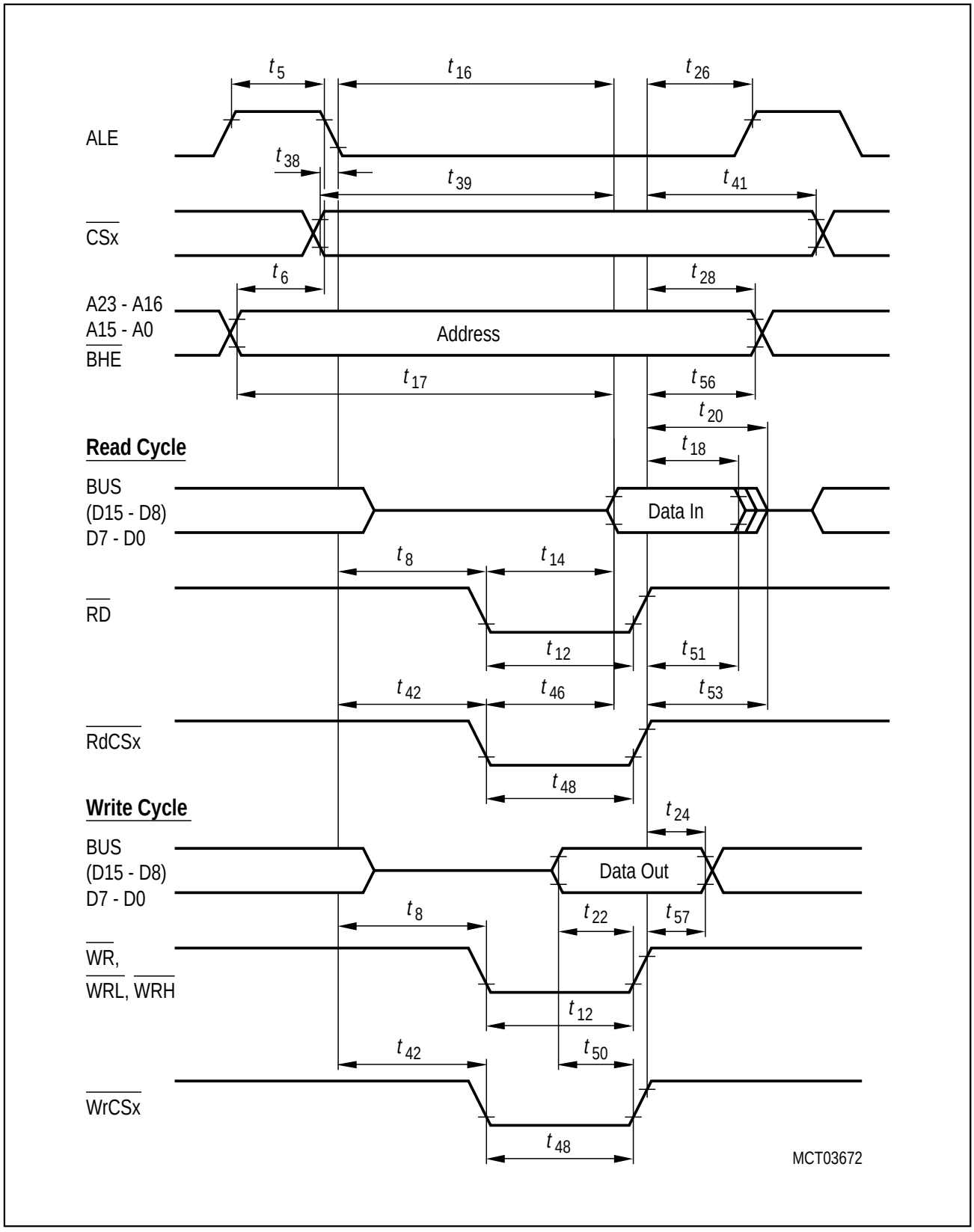


Figure 16-2
External Memory Cycle: Demultiplexed Bus, With Read/Write Delay, Extended ALE

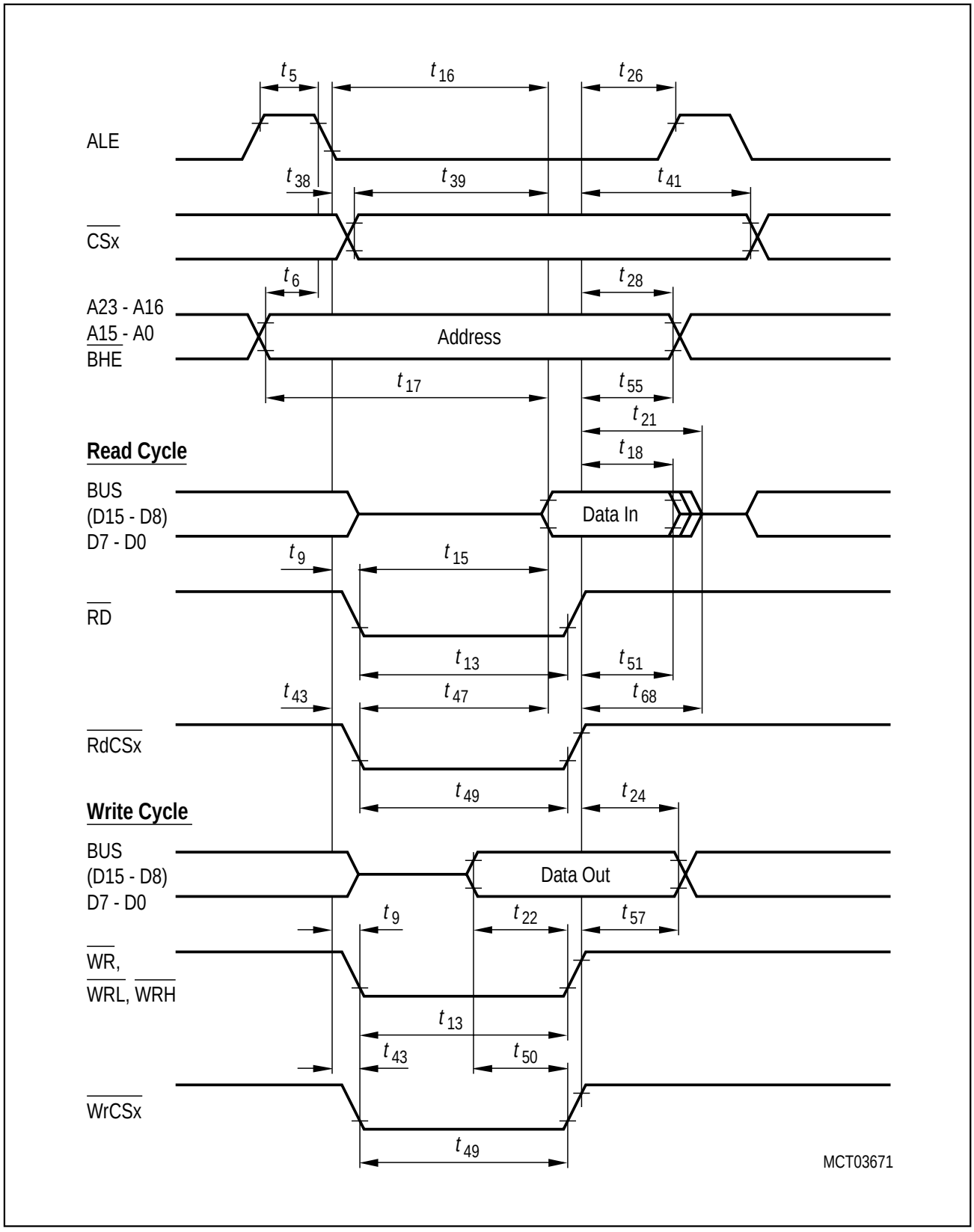


Figure 16-3
External Memory Cycle: Demultiplexed Bus, No Read/Write Delay, Normal ALE

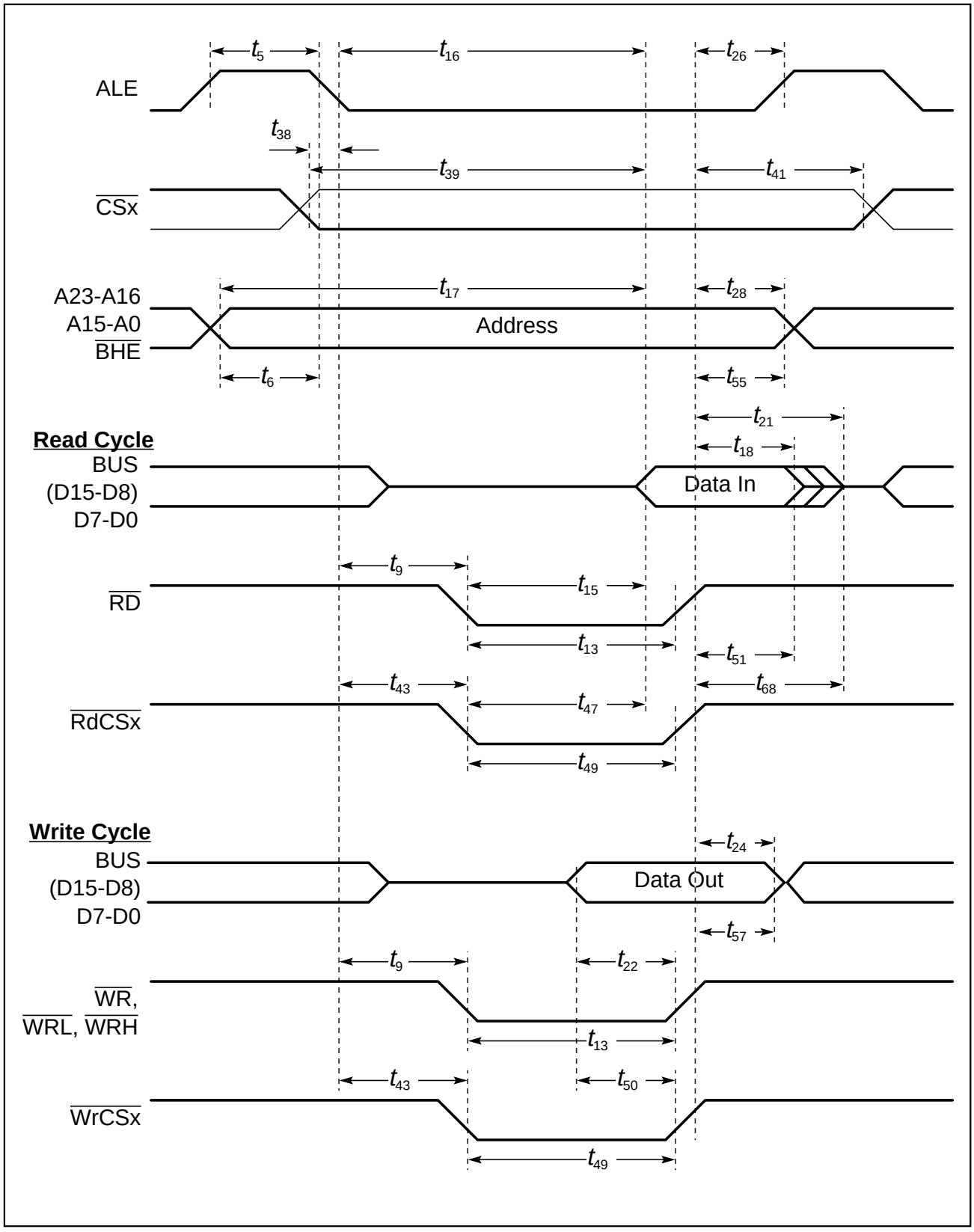


Figure 16-4
External Memory Cycle: Demultiplexed Bus, No Read/Write Delay, Extended ALE

AC Characteristics
Unlatched Chip Select

The C163-16F provides the standard latched $\overline{CS}$ mode known from other C166-Family derivatives and optinally an unlatched $\overline{CS}$ mode. In this case the $\overline{CS}$ signals are directly derived from the address and are output without latching. Unlatched $\overline{CS}$ mode is selected with bit CSCFG = '1' (SYSCON.6).

The table and the figure below describe the $\overline{CS}$ timing in unlatched mode.

Parameter	Symbol	Max. CPU Clock = 25 MHz		Variable CPU Clock 1 / 2TCL = 1 to 25 MHz		Unit
		min.	max.	min.	max.	
Address to $\overline{CS}$ (unlatched $\overline{CS}$ mode)	t_{69} CC	0	5	0	5	ns
$\overline{CS}$ hold after $\overline{WR}$	t_{70} CC	0	15	0	15	ns

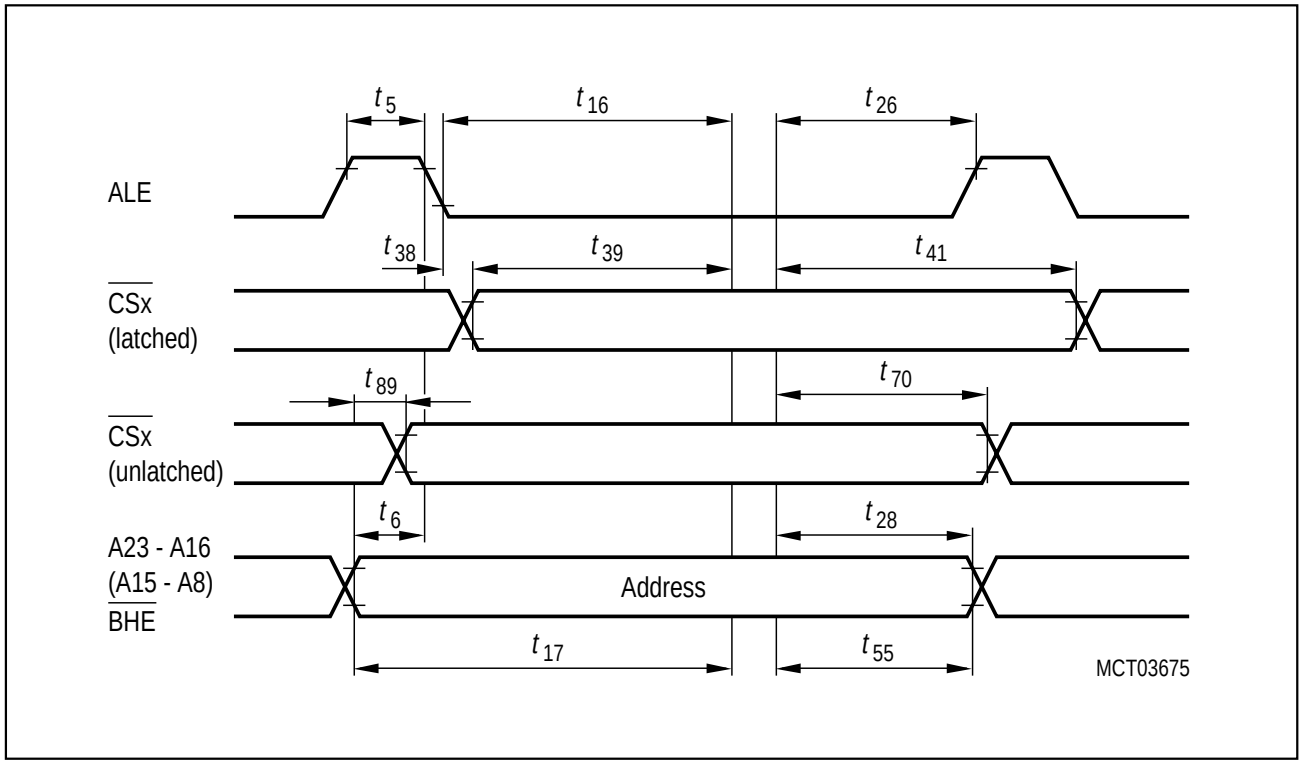


Figure 17
External Memory Cycle: Unlatched $\overline{CS}$ Signal

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

V_{CC} = 5 V ± 10 %; V_{SS} = 0 V
T_A = 0 to +70 ˚C for SAB-C163-16F-LF, SAB-C163-16F-L25F
C_L (for PORT0, PORT1, Port 4, ALE, $\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{BHE}}$, CLKOUT) = 100 pF
C_L (for Port 6, $\overline{\text{CS}}$) = 100 pF

Parameter	Symbol		Max. CPU Clock = 25 MHz		Variable CPU Clock 1 / 2TCL = 1 to 25 MHz		Unit
			min.	max.	min.	max.	
CLKOUT cycle time	t_{29}	CC	40	40	2TCL	2TCL	ns
CLKOUT high time	t_{30}	CC	14	–	TCL – 6	–	ns
CLKOUT low time	t_{31}	CC	10	–	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	54	–	2TCL + 14	–	ns
Asynchronous $\overline{\text{READY}}$ setup time [1]	t_{58}	SR	14	–	14	–	ns
Asynchronous $\overline{\text{READY}}$ hold time [1]	t_{59}	SR	4	–	4	–	ns
Async. $\overline{\text{READY}}$ hold time after $\overline{\text{RD}}$, $\overline{\text{WR}}$ high (Demultiplexed Bus) [2]	t_{60}	SR	0	0 + 2t_A + t_C + t_F [2]	0	TCL - 20 + 2t_A + t_C + t_F [2]	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.

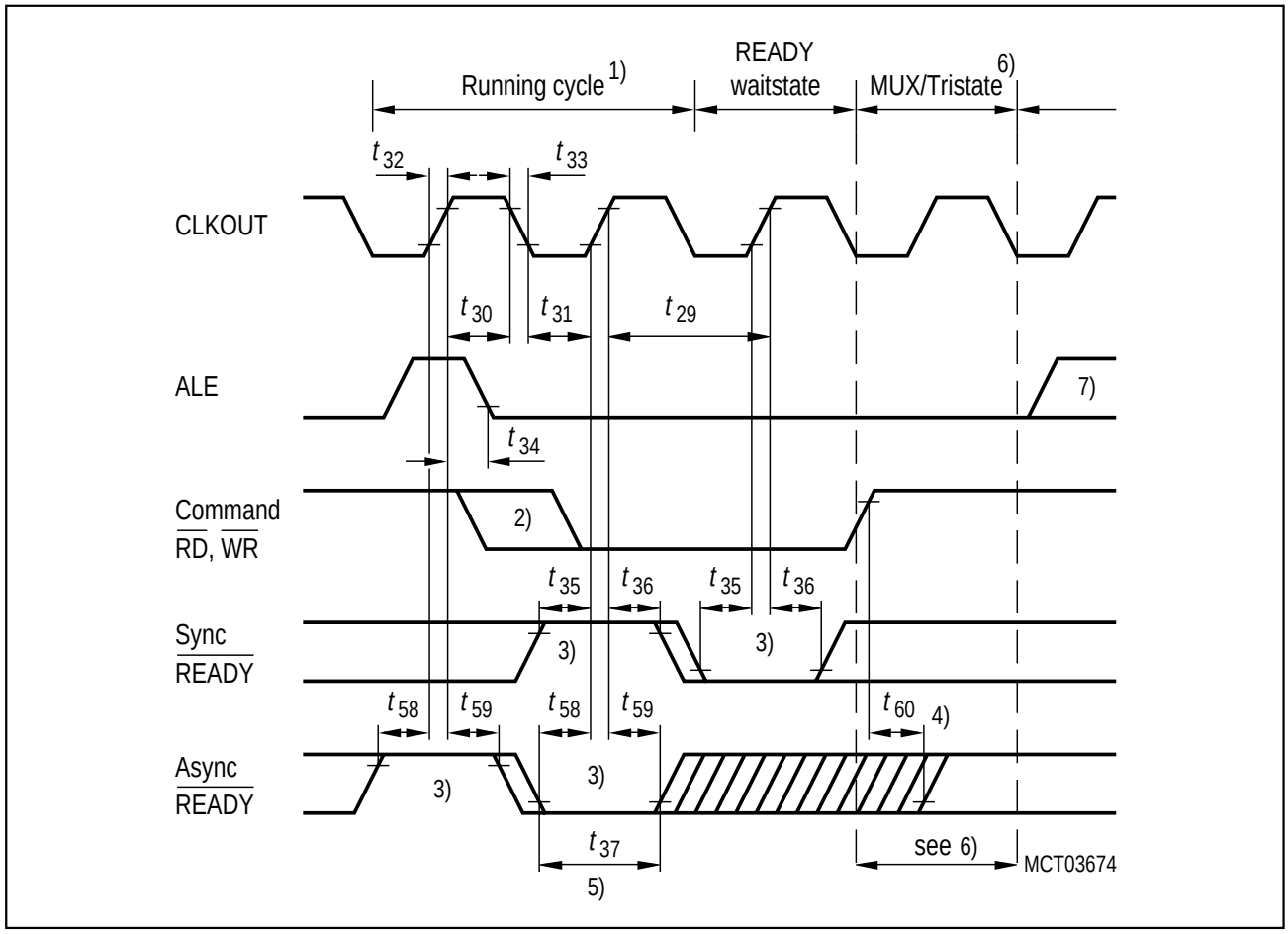


Figure 18
CLKOUT and $\overline{\text{READY}}$

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 (e.g. 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 response 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.

AC Characteristics
External Bus Arbitration

V_{CC} = 5 V ± 10 %; V_{SS} = 0 V
T_A = 0 to +70 ˚C for SAB-C163-16F-LF, SAB-C163-16F-L25F
C_L (for PORT0, PORT1, Port 4, ALE, $\overline{RD}$, $\overline{WR}$, $\overline{BHE}$, CLKOUT) = 100 pF
C_L (for Port 6, $\overline{CS}$) = 100 pF

Parameter	Symbol		Max. CPU Clock = 25 MHz		Variable CPU Clock 1 / 2TCL = 1 to 25 MHz		Unit
			min.	max.	min.	max.	
$\overline{HOLD}$ input setup time to CLKOUT	t_{61}	SR	20	–	20	–	ns
CLKOUT to $\overline{HLDA}$ high or $\overline{BREQ}$ low delay	t_{62}	CC	–	20	–	20	ns
CLKOUT to $\overline{HLDA}$ low or $\overline{BREQ}$ high delay	t_{63}	CC	–	20	–	20	ns
$\overline{CSx}$ release	t_{64}	CC	–	20	–	20	ns
$\overline{CSx}$ drive	t_{65}	CC	-4	24	-4	24	ns
Other signals release	t_{66}	CC	–	20	–	20	ns
Other signals drive	t_{67}	CC	-4	24	-4	24	ns

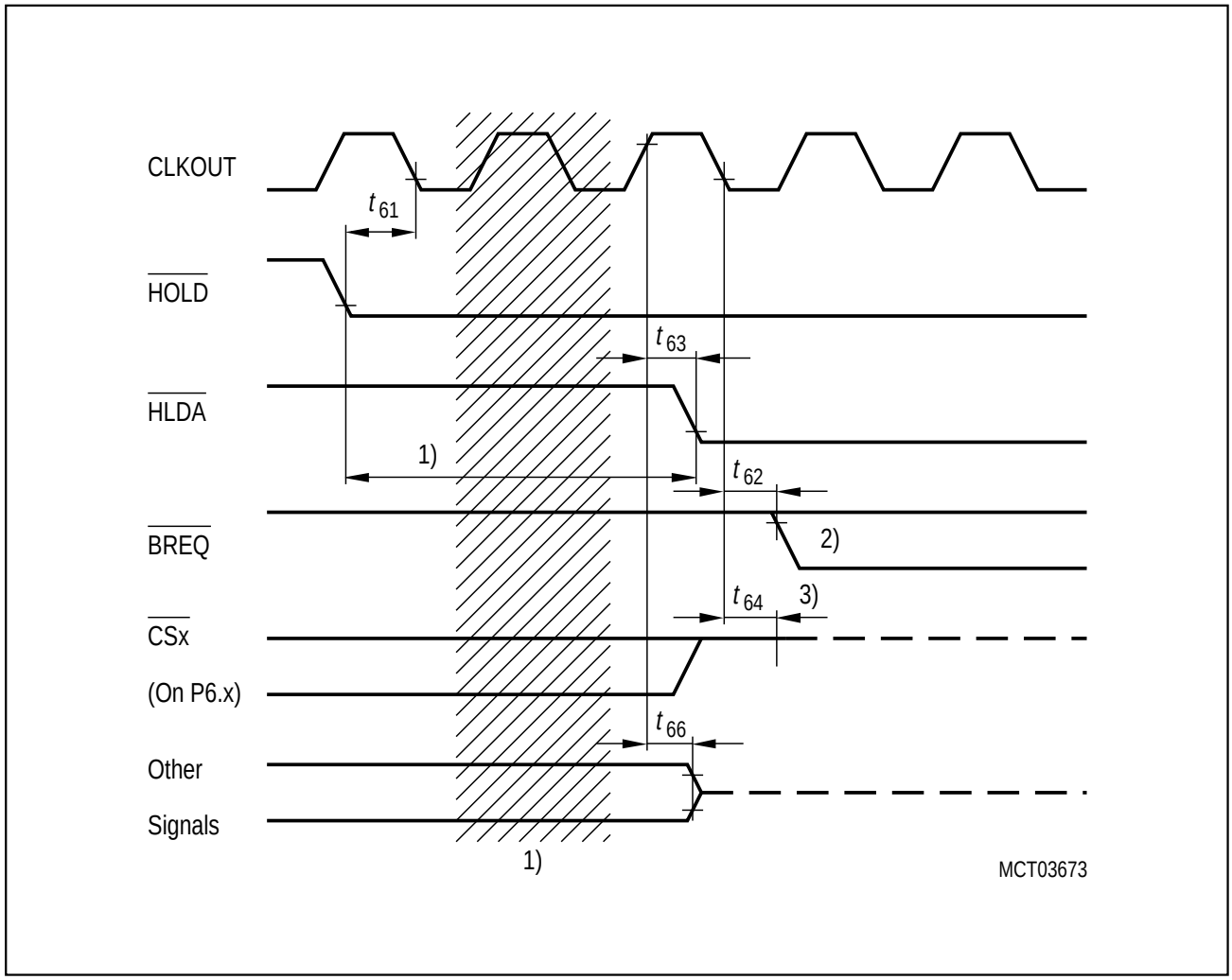


Figure 19
External Bus Arbitration, Releasing the Bus

Notes

1) The C163-16F will complete the currently running bus cycle before granting bus access.

2) This is the first possibility for $\overline{\text{BREQ}}$ to get active.

3) The $\overline{\text{CS}}$ outputs will be resistive high (pullup) after t_{64}.

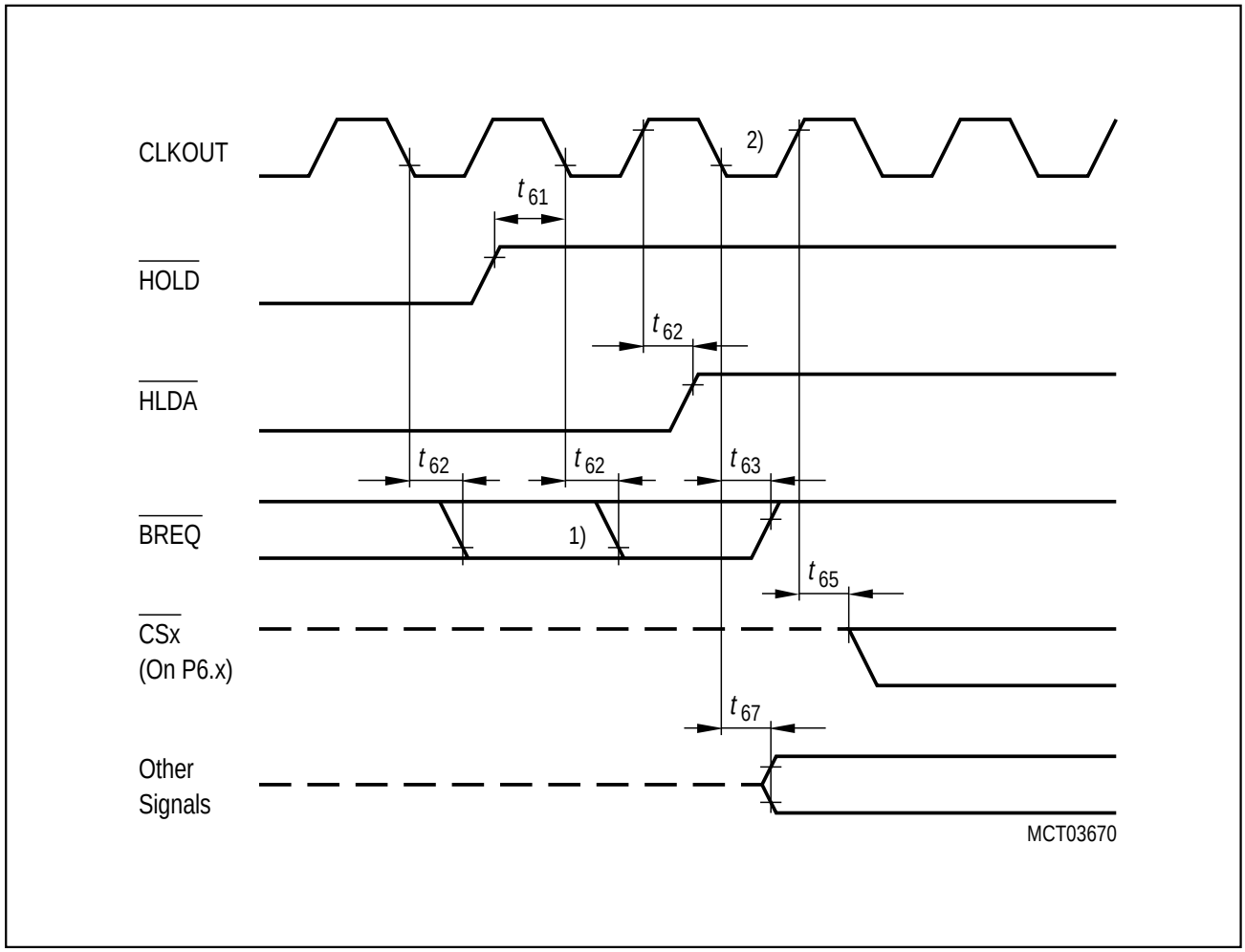


Figure 20
External Bus Arbitration, (Regaining the Bus)

Notes

1) This is the last chance for $\overline{\text{BREQ}}$ to trigger the indicated regain-sequence.
Even if $\overline{\text{BREQ}}$ is activated earlier, the regain-sequence is initiated by $\overline{\text{HOLD}}$ going high.
Please note that $\overline{\text{HOLD}}$ may also be deactivated without the C163-16F requesting the bus.

2) The next C163-16F driven bus cycle may start here.

AC Characteristics
Synchronous Serial Port Timing

V_{CC} = 5 V ± 10 %; V_{SS} = 0 V
T_A = 0 to +70 ˚C for SAB-C163-16F-LF, SAB-C163-16F-L25F
C_L = 100 pF

Parameter	Symbol		Max. Baudrate = 12.5 / 10 MBd		Variable Baudrate = 0.5 to 12.5 MBd		Unit
			min.	max.	min.	max.	
SSP clock cycle time	t_{200}	CC	80 / 100	80 / 100	4 TCL	512 TCL	ns
SSP clock high time	t_{201}	CC	30 / 40	– / –	t_{200}/2 - 10	–	ns
SSP clock low time	t_{202}	CC	30 / 40	– / –	t_{200}/2 - 10	–	ns
SSP clock rise time	t_{203}	CC	– / –	6 / 6	–	6	ns
SSP clock fall time	t_{204}	CC	– / –	6 / 6	–	6	ns
CE active before shift edge	t_{205}	CC	30 / 40	– / –	t_{200}/2 - 10	–	ns
CE inactive after latch edge	t_{206}	CC	70 / 90	90 / 110	t_{200} - 10	t_{200} + 10	ns
Write data valid after shift edge	t_{207}	CC	– / –	10 / 10	–	10	ns
Write data hold after shift edge	t_{208}	CC	0 / 0	– / –	0	–	ns
Write data hold after latch edge	t_{209}	CC	34 / 44	46 / 56	t_{200}/2 - 6	t_{200}/2 + 6	ns
Read data active after latch edge	t_{210}	SR	50 / 60	– / –	t_{200}/2 + 10	–	ns
Read data setup time before latch edge	t_{211}	SR	20 / 20	– / –	20	–	ns
Read data hold time after latch edge	t_{212}	SR	0 / 0	– / –	0	–	ns

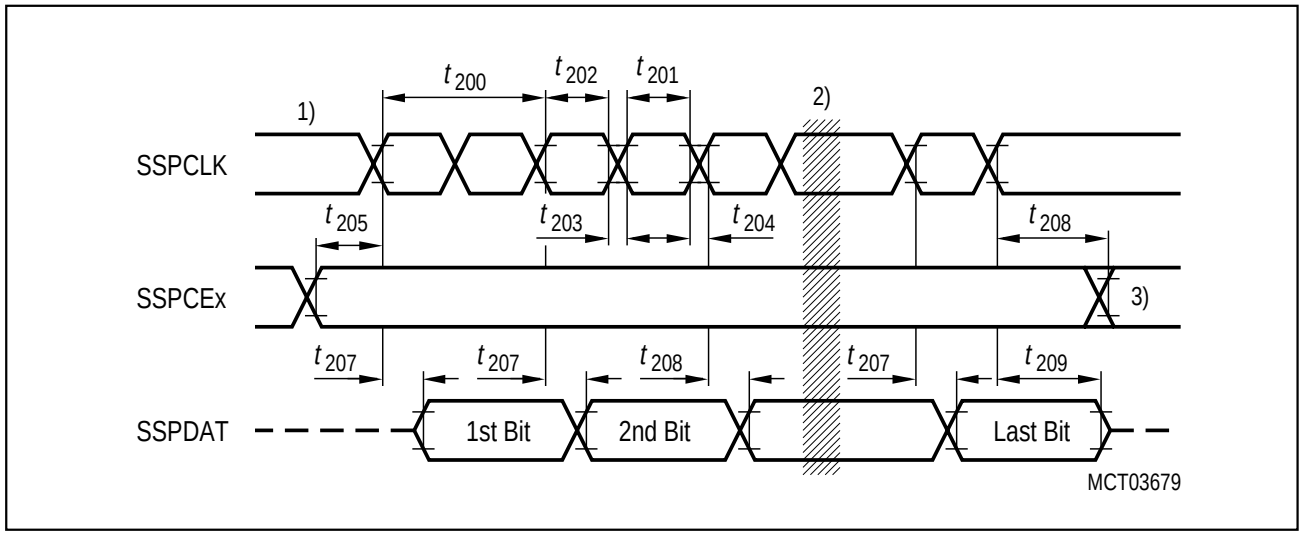


Figure 21
SSP Write Timing

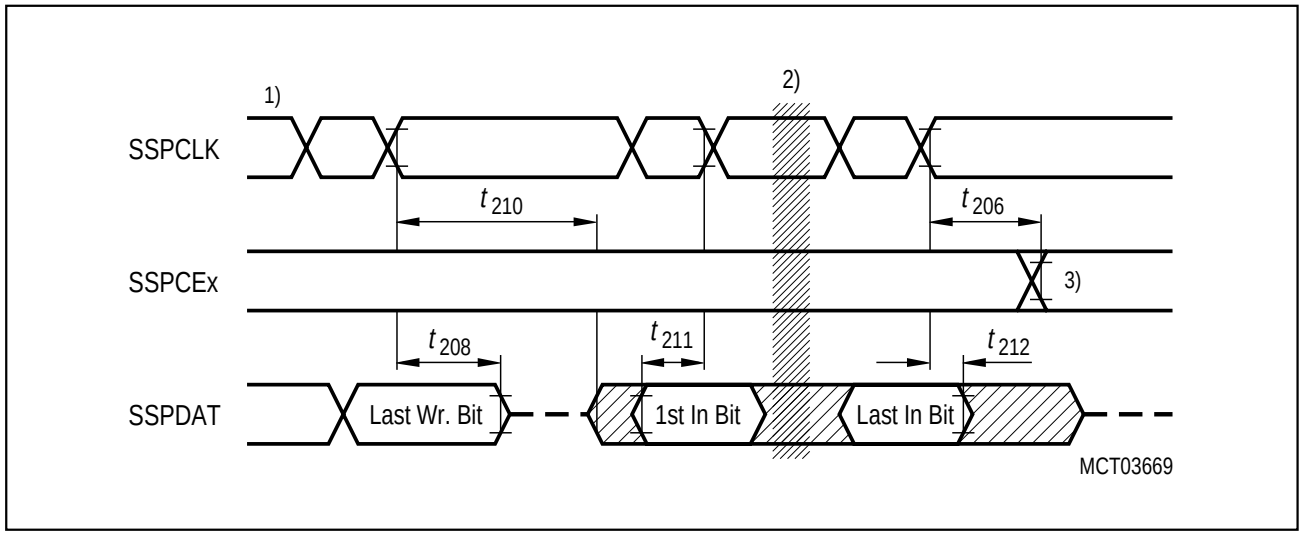


Figure 22
SSP Read Timing

Notes

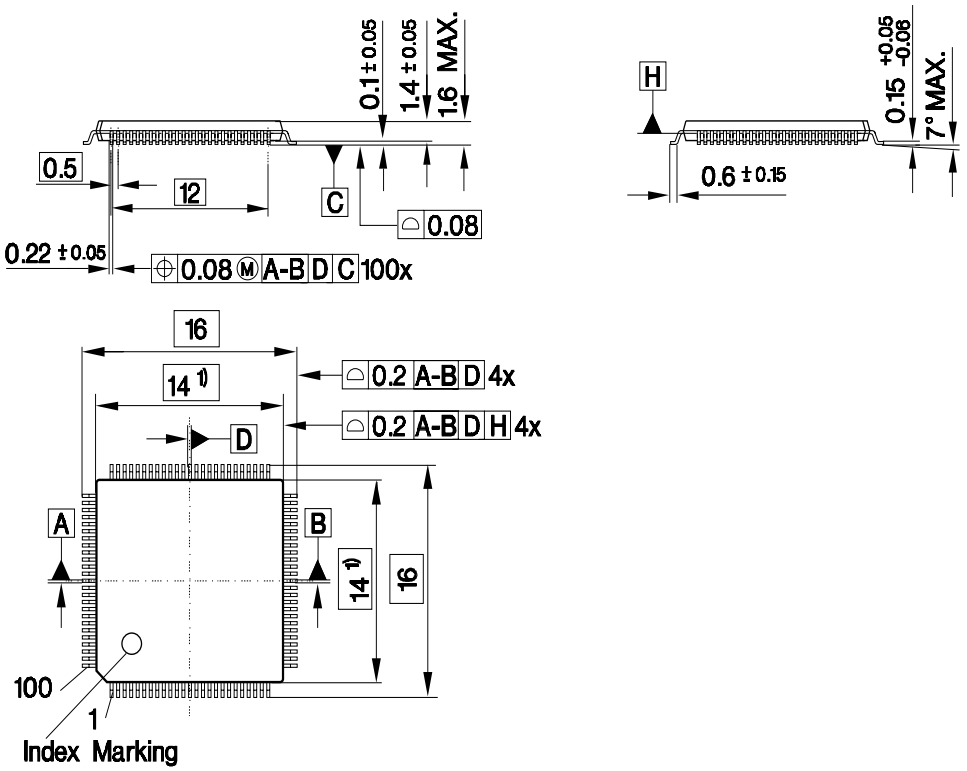
1) The transition of shift and latch edge of SSPCLK is programmable. This figure uses the falling edge as shift edge (drawn bold).

2) The bit timing is repeated for all bits to be transmitted or received.

3) The active level of the chip enable lines is programmable. This figure uses an active low CE (drawn bold). At the end of a transmission or reception the CE signal is disabled in single transfer mode. In continuous transfer mode it remains active.

Package Outline

Plastic Package, P-TQFP-100-3 (SMD)
(Plastic Thin Metric Quad Flat Package)



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

Figure 23

Sorts of Packing
Package outlines for tubes, trays etc. are contained in our Data Book “Package Information”.
SMD = Surface Mounted Device

Dimensions in mm