Infineon Technologies - C161CSLFCABXUMA1 Datasheet

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	C166
Core Size	16-Bit
Speed	25MHz
Connectivity	CANbus, EBI/EMI, I²C, SPI, UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	93
Program Memory Size	-
Program Memory Type	ROMless
EEPROM Size	-
RAM Size	10K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 12x10b
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	128-LQFP
Supplier Device Package	PG-TQFP-128-2
Purchase URL	https://www.e-xfl.com/product-detail/infineon-technologies/c161cslfcabxuma1

Revision History: **2001-01** V3.0

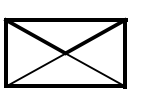
Previous Version: 2000-08 V2.0 (intermediate version)
1999-03 (Advance Information)

Page	Subjects (major changes since last revision)[1)]
All	Converted to Infineon layout
2	Derivative Synopsis Table updated
4, 6, 10, 18	Programmable Interface Routing introduced
27, 28	GPT block diagrams updated
29	RTC description improved
35	OWD description improved
39ff	RSTCON and SDLM registers added
51	Description of input/output voltage and hysteresis improved
53	Separate table for power consumption
57	Clock generation mode table updated
60	External clock drive specification improved
62	Reset calibration time specified, definition of V_{AREF} improved
63	Programmable sample time introduced
65ff	Timing tables updated to 25 MHz

1) Changes refer to version 1999-03.

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

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



C161CS/JC/JI

- High Performance 16-bit CPU with 4-Stage Pipeline
  - 80 ns Instruction Cycle Time at 25 MHz CPU Clock
  - 400 ns Multiplication (16 × 16 bit), 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
  - 16 MBytes Total Linear Address Space for Code and Data
  - 1024 Bytes On-Chip Special Function Register Area
- 16-Priority-Level Interrupt System with 59 Sources, Sample-Rate down to 40 ns
- 8-Channel Interrupt-Driven Single-Cycle Data Transfer Facilities via Peripheral Event Controller (PEC)
- Clock Generation via on-chip PLL (factors 1:1.5/2/2.5/3/4/5), via prescaler or via direct clock input
  - Additional 32 kHz Oscillator
- On-Chip Memory Modules
  - 2 KBytes On-Chip Internal RAM (IRAM)
  - 8 KBytes On-Chip Extension RAM (XRAM)
  - 256 KBytes On-Chip Mask ROM
- On-Chip Peripheral Modules
  - 12-Channel 10-bit A/D Converter with Programmable Conversion Time down to 7.8 µs
  - Two 16-Channel Capture/Compare Units (eight IO lines each)
  - Two Multi-Functional General Purpose Timer Units with 5 Timers
  - Two Asynchronous/Synchronous Serial Channels
  - High-Speed Synchronous Serial Channel (SPI)
  - On-Chip CAN Interface (Rev. 2.0B active, Full CAN / Basic CAN) with 15 Message Objects (**C161CS 2x, C161JC 1x**)
  - Serial Data Link Module (SDLM), compliant with J1850, supporting Class 2 (**C161JC/JI**)
  - IIC Bus Interface (10-bit Addressing, 400 kHz) with 2 Channels (multiplexed)
  - On-Chip Real Time Clock
- Up to 16 MBytes External Address Space for Code and Data
  - Programmable External Bus Characteristics for Different Address Ranges
  - Multiplexed or Demultiplexed External Address/Data Buses with 8-Bit or 16-Bit Data Bus Width
  - Five Programmable Chip-Select Signals
  - Hold- and Hold-Acknowledge Bus Arbitration Support

Table 2 Pin Definitions and Functions (cont'd)

Symbol	Pin No.	Input Outp.	Function
PORT1		IO	PORT1 consists of the two 8-bit bidirectional I/O ports P1L and P1H. It is bit-wise programmable for input or output via direction bits. For a pin configured as input, the output driver is put into high-impedance state. PORT1 is used as the 16-bit address bus (A) in demultiplexed bus modes and also after switching from a demultiplexed bus mode to a multiplexed bus mode.
P1L.0-7	103-110		
P1H.0-7	113-120		
			The following PORT1 pins also serve for alternate functions:
P1H.4	117	I/O	CC24IO CAPCOM2: CC24 Capture Inp./Compare Outp.
P1H.5	118	I/O	CC25IO CAPCOM2: CC25 Capture Inp./Compare Outp.
P1H.6	119	I/O	CC26IO CAPCOM2: CC26 Capture Inp./Compare Outp.
P1H.7	120	I/O	CC27IO CAPCOM2: CC27 Capture Inp./Compare Outp.
XTAL2	123	O	XTAL2: Output of the oscillator amplifier circuit.
XTAL1	124	I	XTAL1: Input to the oscillator amplifier and input to the internal clock generator To clock the device from an external source, drive XTAL1, while leaving XTAL2 unconnected. Minimum and maximum high/low and rise/fall times specified in the AC Characteristics must be observed.
XTAL3	126	I	XTAL3: Input to the 32-kHz oscillator amplifier and input to the internal clock generator
XTAL4	127	O	XTAL4: Output of the oscillator amplifier circuit. To clock the device from an external source, drive XTAL3, while leaving XTAL4 unconnected. Minimum and maximum high/low and rise/fall times specified in the AC Characteristics must be observed.

Interrupt System

With an interrupt response time within a range from just 5 to 12 CPU clocks (in case of internal program execution), the C161CS/JC/JI is capable of reacting very fast to the occurrence of non-deterministic events.

The architecture of the C161CS/JC/JI 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 C161CS/JC/JI 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.

Table 3 shows all of the possible C161CS/JC/JI interrupt sources and the corresponding hardware-related interrupt flags, vectors, vector locations and trap (interrupt) numbers.

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

The C161CS/JC/JI also provides an excellent mechanism to identify and to process exceptions or error conditions that arise during run-time, so-called ‘Hardware Traps’. Hardware traps cause immediate non-maskable system reaction which is similar to a standard interrupt service (branching to a dedicated vector table location). The occurence of a hardware trap is additionally signified by an individual bit in the trap flag register (TFR). Except when another higher prioritized trap service is in progress, a hardware trap will interrupt any actual program execution. In turn, hardware trap services can normally not be interrupted by standard or PEC interrupts.

Table 4 shows all of the possible exceptions or error conditions that can arise during run-time:

Table 4 Hardware Trap Summary

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
W-dog 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

after the capture procedure. This allows the C161CS/JC/JI to measure absolute time differences or to perform pulse multiplication without software overhead.

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

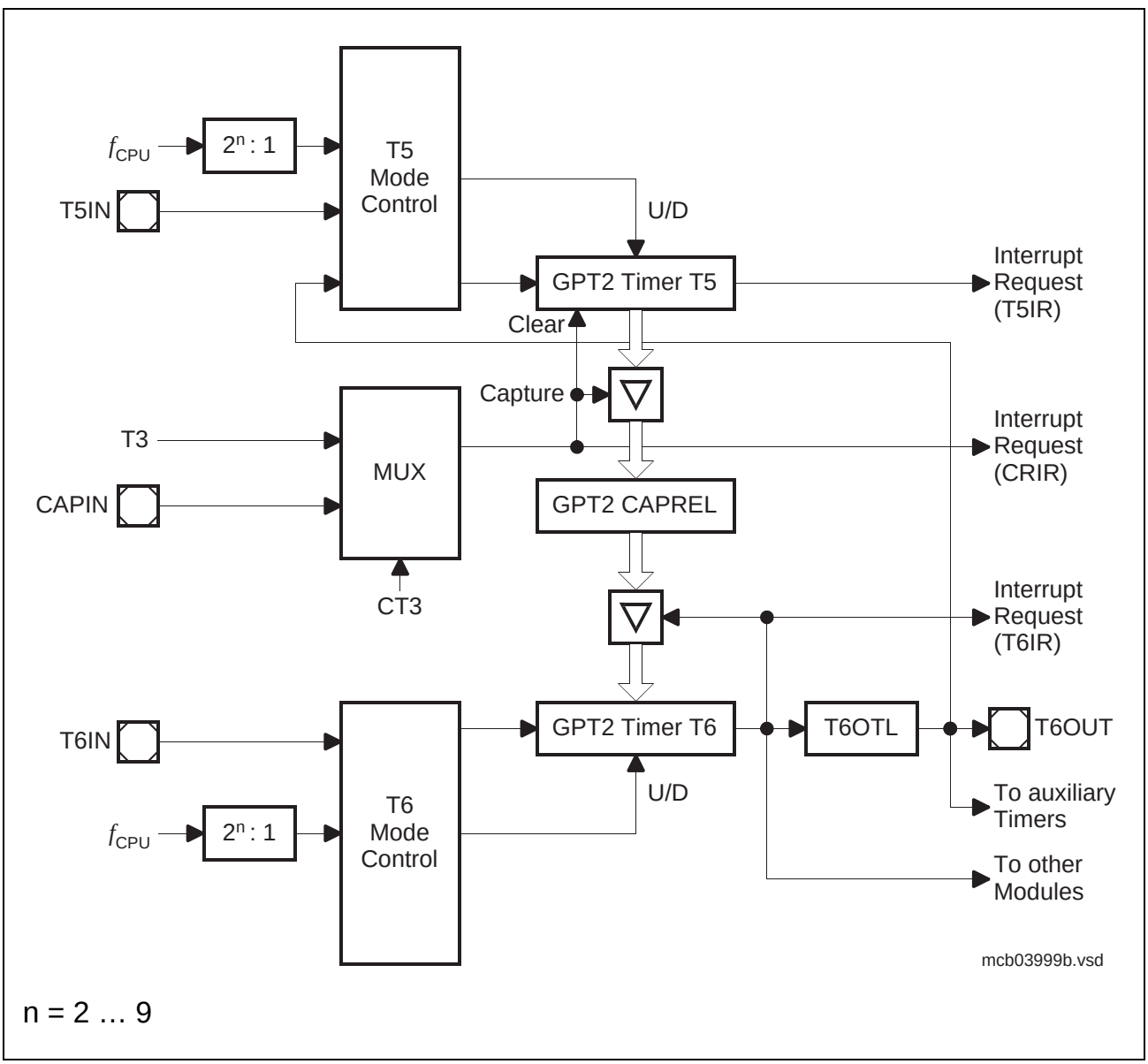


Figure 7 Block Diagram of GPT2

Serial Channels

Serial communication with other microcontrollers, processors, terminals or external peripheral components is provided by three serial interfaces with different functionality, two Asynchronous/Synchronous Serial Channels (**ASC0**/**ASC1**) and a High-Speed Synchronous Serial Channel (**SSC**).

The ASC0 is upward compatible with the serial ports of the Infineon 8-bit microcontroller families and supports full-duplex asynchronous communication at up to 781 kBaud and half-duplex synchronous communication at up to 3.1 MBaud (@ 25 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 ASC1 is function compatible with the ASC0, except that its registers are not bit-addressable (XBUS peripheral) and it provides only three interrupt vectors.

The SSC supports full-duplex synchronous communication at up to 6.25 MBaud (@ 25 MHz CPU clock). It may be configured so it interfaces with serially linked peripheral components. A dedicated baud rate generator allows to set up all standard baud rates without oscillator tuning. For transmission, reception and error handling three separate interrupt vectors are provided.
The SSC transmits or receives characters of 2 … 16 bits length synchronously to a shift clock which can be generated by the SSC (master mode) or by an external master (slave mode). The SSC can start shifting with the LSB or with the MSB and allows the selection of shifting and latching clock edges as well as the clock polarity.
A number of optional hardware error detection capabilities has been included to increase the reliability of data transfers. Transmit and receive error supervise the correct handling of the data buffer. Phase and baudrate error detect incorrect serial data.

Table 7 C161CS/JC/JI Registers, Ordered by Name (cont'd)

Name	Physical Address	8-Bit Addr.	Description	Reset Value
C1PCIR	$EF02_H$ **X**	---	CAN1 Port Control / Interrupt Register	$XXXX_H$
C1LARn	$EFn4_H$ **X**	---	CAN1 Lower Arbitration Reg. (msg. **n**)	$UUUU_H$
C1LGML	$EF0A_H$ **X**	---	CAN1 Lower Global Mask Long	$UUUU_H$
C1LMLM	$EF0E_H$ **X**	---	CAN1 Lower Mask of Last Message	$UUUU_H$
C1MCFGn	$EFn6_H$ **X**	---	CAN1 Message Config. Reg. (msg. **n**)	UU_H
C1MCRn	$EFn0_H$ **X**	---	CAN1 Message Control Reg. (msg. **n**)	$UUUU_H$
C1UARn	$EFn2_H$ **X**	---	CAN1 Upper Arbitration Reg. (msg. **n**)	$UUUU_H$
C1UGML	$EF08_H$ **X**	---	CAN1 Upper Global Mask Long	$UUUU_H$
C1UMLM	$EF0C_H$ **X**	---	CAN1 Upper Mask of Last Message	$UUUU_H$
C2BTR	$EE04_H$ **X**	---	CAN2 Bit Timing Register	$UUUU_H$
C2CSR	$EE00_H$ **X**	---	CAN2 Control / Status Register	$XX01_H$
C2GMS	$EE06_H$ **X**	---	CAN2 Global Mask Short	$UFUU_H$
C2PCIR	$EE02_H$ **X**	---	CAN2 Port Control / Interrupt Register	$XXXX_H$
C2LARn	$EEn4_H$ **X**	---	CAN2 Lower Arbitration Reg. (msg. **n**)	$UUUU_H$
C2LGML	$EE0A_H$ **X**	---	CAN2 Lower Global Mask Long	$UUUU_H$
C2LMLM	$EE0E_H$ **X**	---	CAN2 Lower Mask of Last Message	$UUUU_H$
C2MCFGn	$EEn6_H$ **X**	---	CAN2 Message Config. Reg. (msg. **n**)	UU_H
C2MCRn	$EEn0_H$ **X**	---	CAN2 Message Control Reg. (msg. **n**)	$UUUU_H$
C2UARn	$EEn2_H$ **X**	---	CAN2 Upper Arbitration Reg. (msg. **n**)	$UUUU_H$
C2UGML	$EE08_H$ **X**	---	CAN2 Upper Global Mask Long	$UUUU_H$
C2UMLM	$EE0C_H$ **X**	---	CAN2 Upper Mask of Last Message	$UUUU_H$
CAPREL	$FE4A_H$	25_H	GPT2 Capture/Reload Register	0000_H
CC0	$FE80_H$	40_H	CAPCOM Register 0	0000_H
CC0IC **b**	$FF78_H$	BC_H	CAPCOM Register 0 Interrupt Ctrl. Reg.	0000_H
CC1	$FE82_H$	41_H	CAPCOM Register 1	0000_H
CC10	$FE94_H$	$4A_H$	CAPCOM Register 10	0000_H
CC10IC **b**	$FF8C_H$	$C6_H$	CAPCOM Reg. 10 Interrupt Ctrl. Reg.	0000_H
CC11	$FE96_H$	$4B_H$	CAPCOM Register 11	0000_H
CC11IC **b**	$FF8E_H$	$C7_H$	CAPCOM Reg. 11 Interrupt Ctrl. Reg.	0000_H
CC12	$FE98_H$	$4C_H$	CAPCOM Register 12	0000_H
CC12IC **b**	$FF90_H$	$C8_H$	CAPCOM Reg. 12 Interrupt Ctrl. Reg.	0000_H
CC13	$FE9A_H$	$4D_H$	CAPCOM Register 13	0000_H

Absolute Maximum Ratings

Table 8 Absolute Maximum Rating Parameters

Parameter	Symbol	Limit Values		Unit	Notes
		min.	max.		
Storage temperature	T_{ST}	-65	150	°C	–
Junction temperature	T_J	-40	150	°C	under bias
Voltage on V_{DD} pins with respect to ground (V_{SS})	V_{DD}	-0.5	6.5	V	–
Voltage on any pin with respect to ground (V_{SS})	V_{IN}	-0.5	V_{DD} + 0.5	V	–
Input current on any pin during overload condition	–	-10	10	mA	–
Absolute sum of all input currents during overload condition	–	–	|100|	mA	–
Power dissipation	P_{DISS}	–	1.5	W	–

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

Parameter Interpretation

The parameters listed in the following partly represent the characteristics of the C161CS/JC/JI 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 C161CS/JC/JI 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 C161CS/JC/JI.

DC Characteristics

(Operating Conditions apply)[1)]

Parameter	Symbol		Limit Values		Unit	Test Condition
			min.	max.		
Input low voltage (TTL, all except XTAL1, XTAL3, Port 9)	V_{IL}	SR	-0.5	0.2 V_{DD} - 0.1	V	–
Input low voltage XTAL1, XTAL3, Port 9	V_{IL2}	SR	-0.5	0.3 V_{DD}	V	–
Input low voltage (Special Threshold)	V_{ILS}	SR	-0.5	2.0	V	–
Input high voltage (TTL, all except $\overline{\text{RSTIN}}$, XTAL1, XTAL3, Port 9)	V_{IH}	SR	0.2 V_{DD} + 0.9	V_{DD} + 0.5	V	–
Input high voltage $\overline{\text{RSTIN}}$ (when operated as input)	V_{IH1}	SR	0.6 V_{DD}	V_{DD} + 0.5	V	–
Input high voltage XTAL1, XTAL3, Port 9	V_{IH2}	SR	0.7 V_{DD}	V_{DD} + 0.5	V	–
Input high voltage (Special Threshold)	V_{IHS}	SR	0.8 V_{DD} - 0.2	V_{DD} + 0.5	V	–
Input Hysteresis (Special Threshold)	HYS		400	–	mV	Series resistance = 0 Ω
Output low voltage (PORT0, PORT1, Port 4, ALE, $\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{BHE}}$, CLKOUT, $\overline{\text{RSTOUT}}$, $\overline{\text{RSTIN}}$[2)])	V_{OL}	CC	–	0.45	V	I_{OL} = 2.4 mA[3)] I_{OL} = 0.5 mA[4)]
Output low voltage (Port 9)	V_{OL9}	CC	–	0.4	V	I_{OL} = 3.0 mA

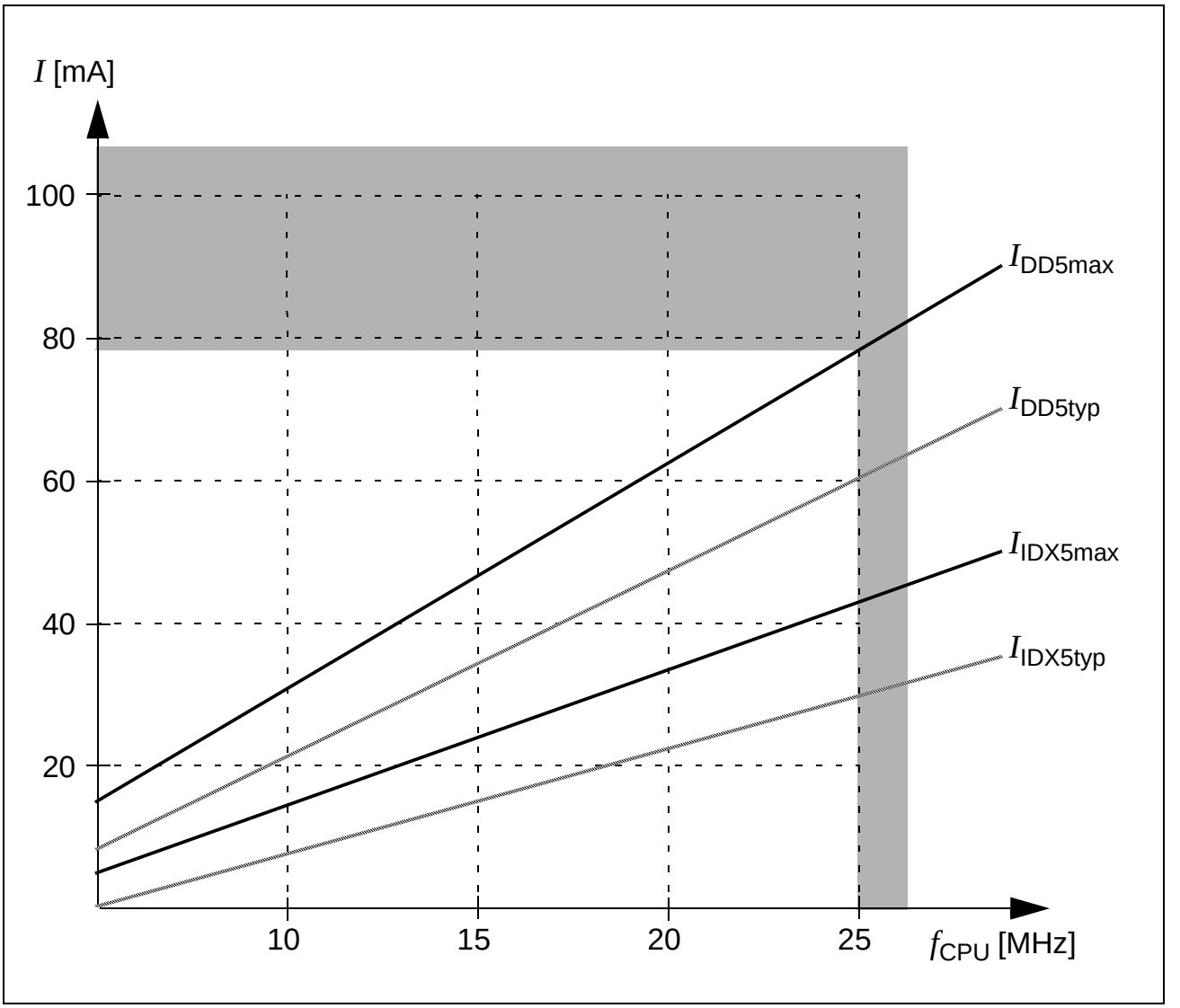


Figure 10 Supply/Idle Current as a Function of Operating Frequency

Direct Drive

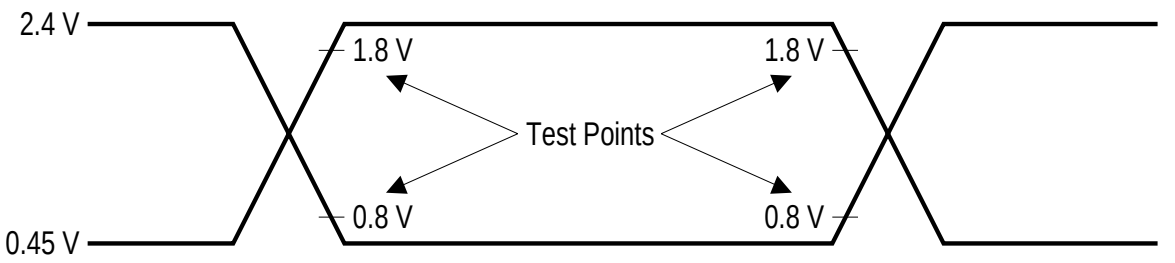
When direct drive is configured (CLKCFG = 011_B) the on-chip phase locked loop is disabled and the CPU clock is directly driven from the internal oscillator with the input clock signal.
The frequency of f_{CPU} directly follows the frequency of f_{OSC} so the high and low time of f_{CPU} (i.e. the duration of an individual TCL) is defined by the duty cycle of the input clock f_{OSC}.
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:

$TCL_{min} = 1/f_{OSC} \times DC_{min}$ (DC = duty cycle)

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

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

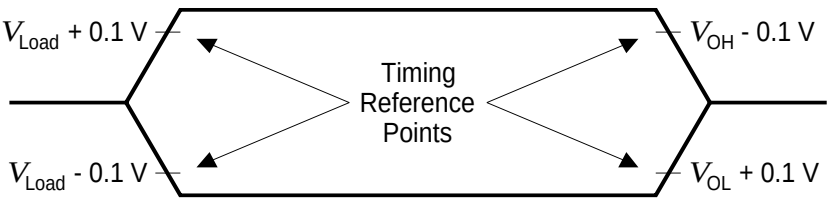
Testing Waveforms



AC inputs during testing are driven at 2.4 V for a logic ‘1’ and 0.45 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’.

MCA04414

Figure 14 Input Output Waveforms



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

MCA00763

Figure 15 Float Waveforms

Memory Cycle Variables

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

Table 15 Memory Cycle Variables

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>)

Note: Please respect the maximum operating frequency of the respective derivative.

AC Characteristics

Multiplexed Bus

(Operating Conditions apply)

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

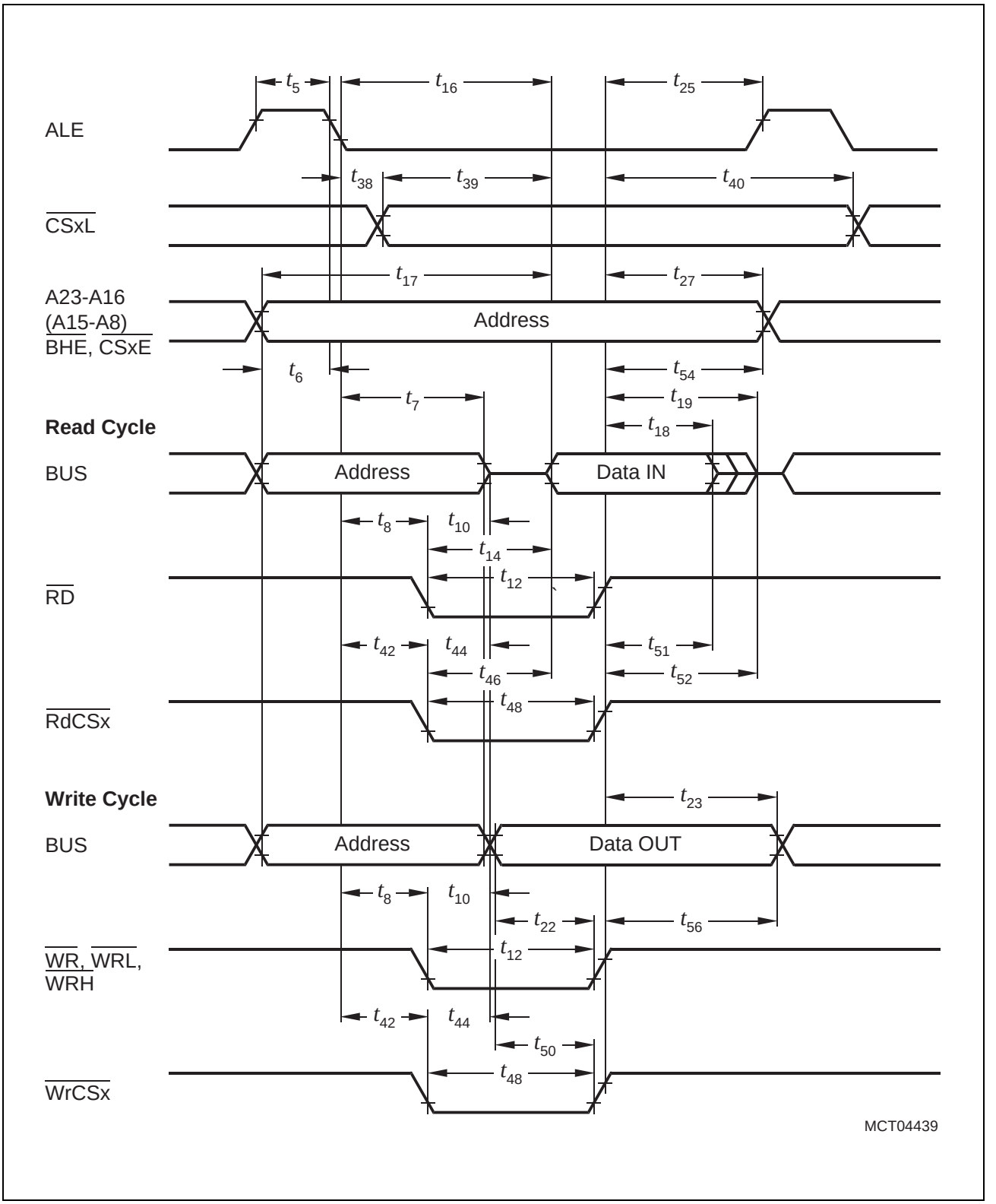


Figure 16 External Memory Cycle: Multiplexed Bus, With Read/Write Delay, Normal ALE

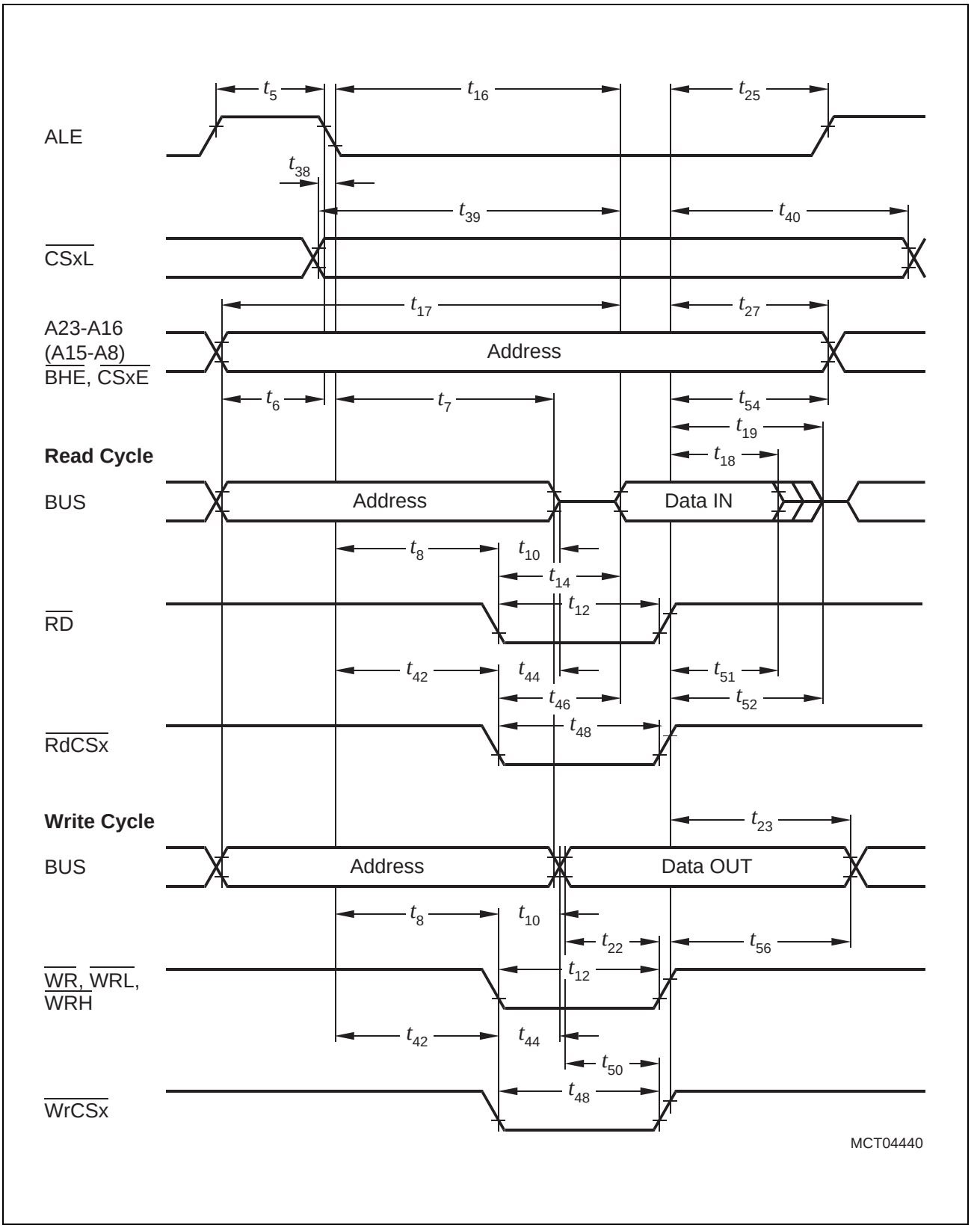


**Figure 17 External Memory Cycle:
Multiplexed Bus, With Read/Write Delay, Extended ALE**

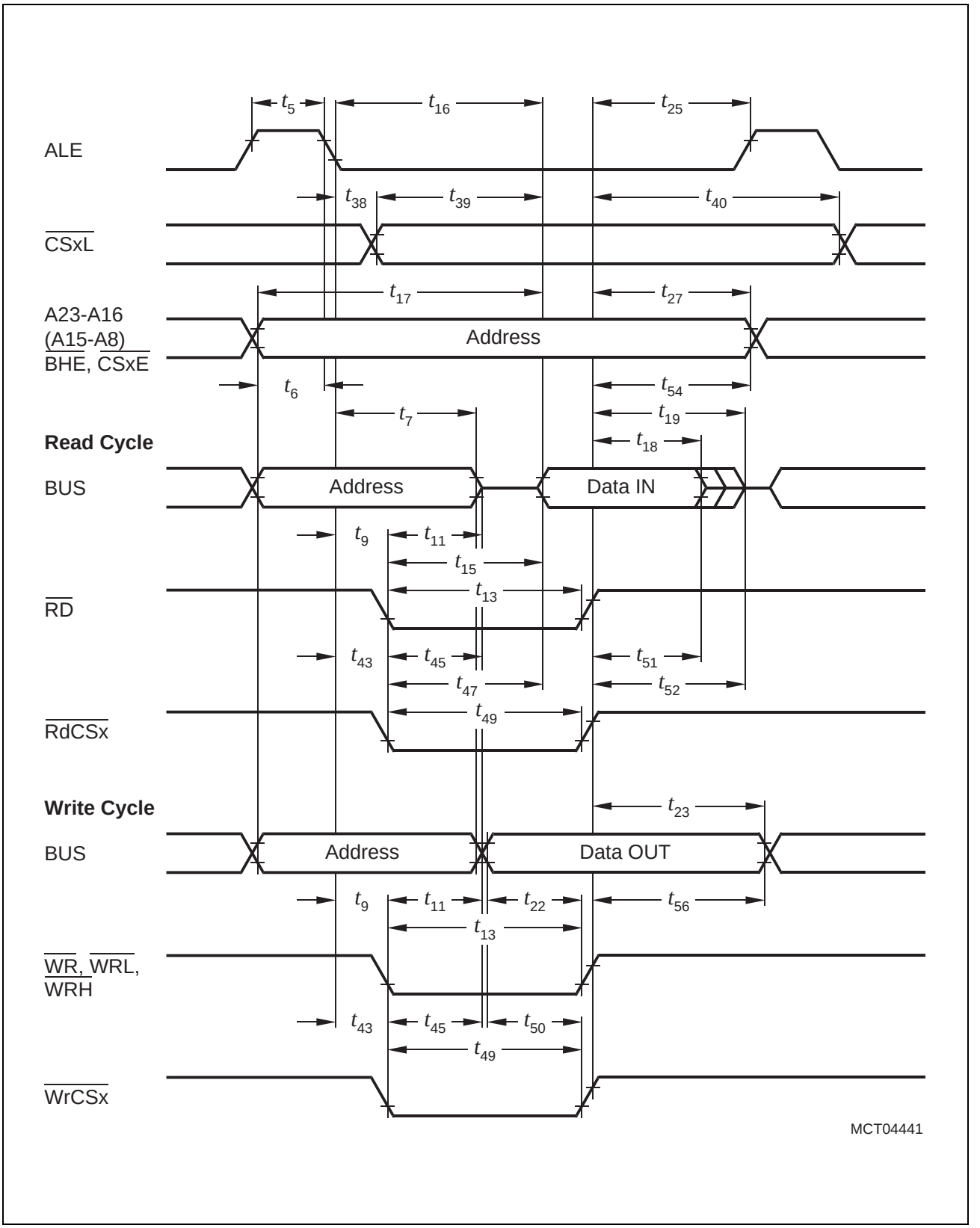


**Figure 18 External Memory Cycle:
Multiplexed Bus, No Read/Write Delay, Normal ALE**

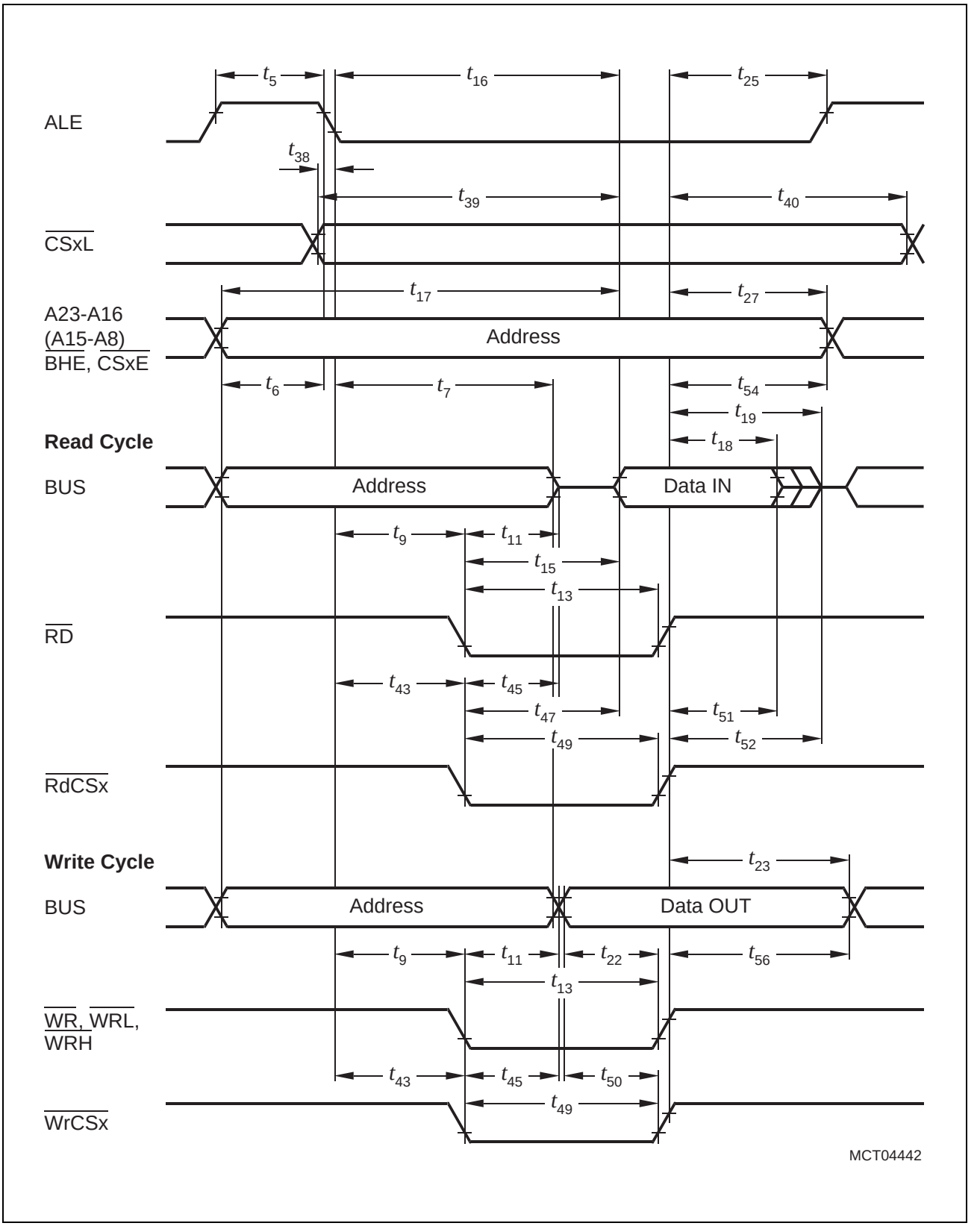


**Figure 19 External Memory Cycle:
Multiplexed Bus, No Read/Write Delay, Extended ALE**

Demultiplexed Bus (cont'd)
(Operating Conditions apply)
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.	
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
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 $\overline{CS}$[3]	t_{38}	CC	-4 - t_A	10 - t_A	-4 - t_A	10 - t_A	ns
$\overline{CS}$ low to Valid Data In[3]	t_{39}	SR	–	40 + t_C + $2t_A$	–	3TCL - 20 + t_C + $2t_A$	ns
$\overline{CS}$ hold after $\overline{RD}$, $\overline{WR}$[3]	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)[1]	t_{53}	SR	–	20 + t_F	–	2TCL - 20 + $2t_A$ + t_F[1]	ns

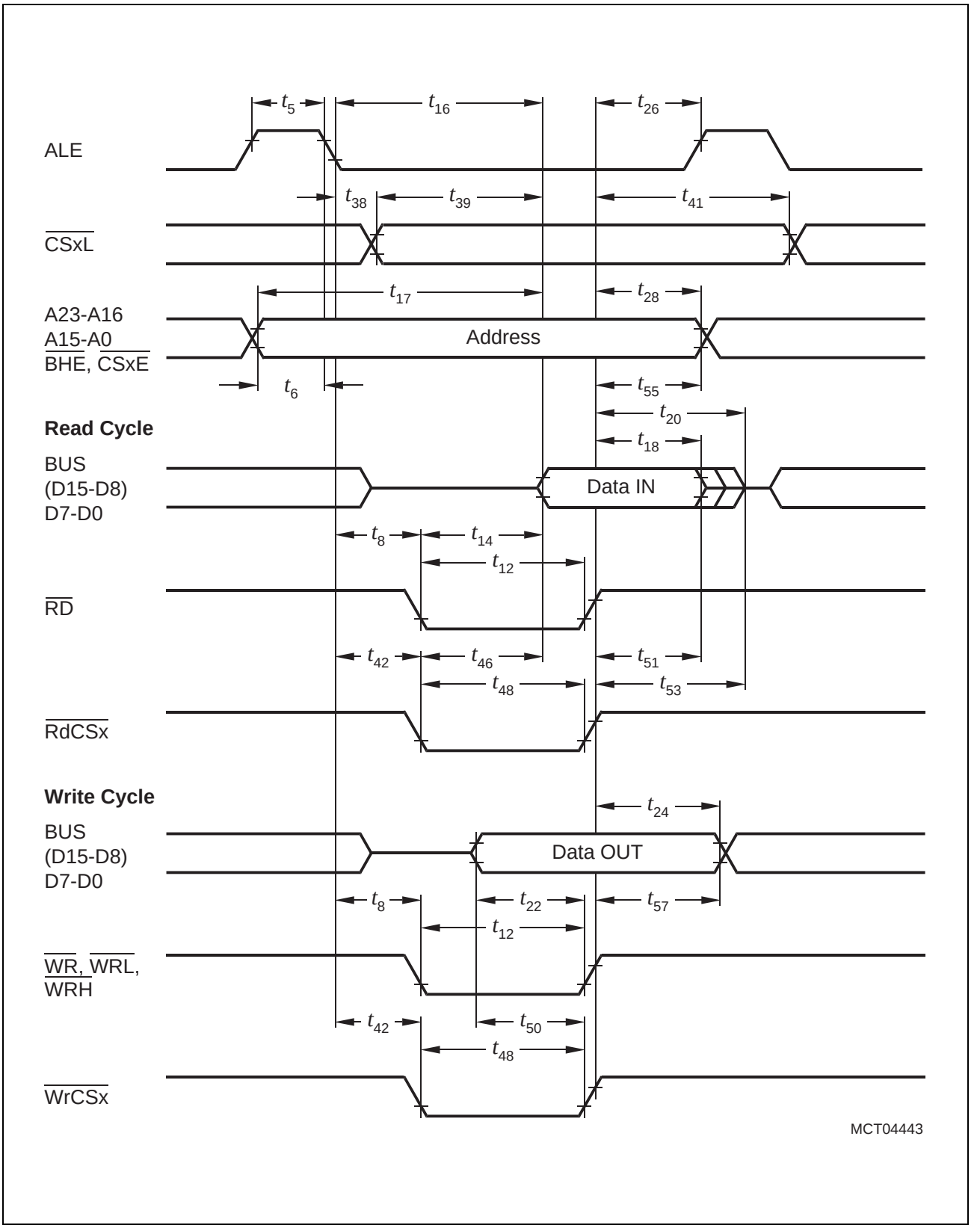


**Figure 20 External Memory Cycle:
Demultiplexed Bus, With Read/Write Delay, Normal ALE**