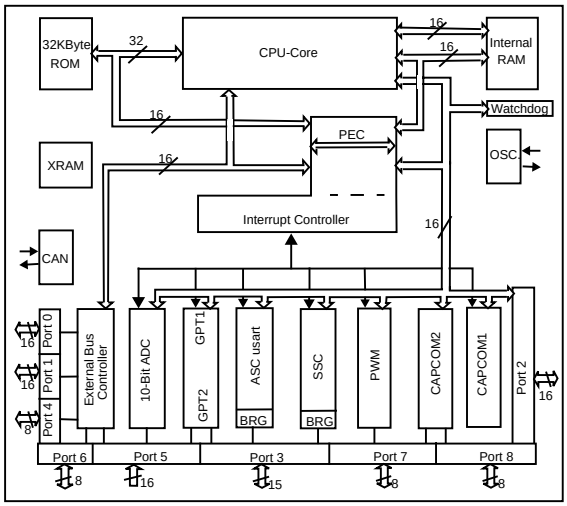
SGS-THOMSON MICROELECTRONICS

PRELIMINARY DATA

# ST10C167

16-bit MCU with 32KByte ROM

- High performance CPU
  - 16-bit CPU with 4-stage pipeline.
  - 80ns instruction cycle time at 25MHz CPU clock.
  - 400ns 16 x 16-bit multiplication.
  - 800ns 32 / 16-bit division.
  - Enhanced boolean bit manip facilities.
  - Additional instructions to support HLL and operating systems.
  - Single-cycle context switching support.
- Memory organization
  - 32K Bytes on-chip ROM memory.
  - Up to 16 MBytes linear address space for code and data (5 MByte with CAN).
  - 2KByte on-chip internal RAM (IRAM).
  - 2KByte on-chip extension RAM (XRAM).
- Fast and flexible bus
  - 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.
  - Hold-acknowledge bus arbitration support.
- Interrupt
  - 8-channel peripheral event controller for single cycle, interrupt driven data transfer.
  - 16-priority-level interrupt system with 56 sources, sample-rate down to 40 ns.
- Timers
  - Two multi-functional general purpose timer units with 5 timers.
  - Two 16-channel capture/compare units.
- A/D converter
  - 16-channel 10-bit.
  - 7.76 s conversion time
- Fail-safe protection
  - Programmable watchdog timer.
  - Oscillator Watchdog.




- On-chip CAN 2.0b interface
- On-chip bootstrap loader
- Clock generation
  - On-chip PLL.
  - Direct or prescaled clock input.
- Up to 111 general purpose I/O lines
  - Individually programmable as input, output or special function.
- Programmable drive strength
- Programmable threshold (hysteresis)
- Idle and power down modes
  - Idle Current <70mA.
  - Power down supply current <100 A.
- 4-Channel PWM Unit
- Serial channels
  - Synchronous/async serial channel
  - High-speed synchronous channel.
- Development support
  - C-compilers, macro-assembler packages, emulators, evaluation boards, HLL-debuggers, simulators, logic analyzer disassemblers, programming boards.
- Package
  - 144-Pin PQFP Package.

# 1 Introduction

The ST10C167 is a derivative of the SGS THOMSON ST10 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 I/O capabilities. It also provides on-chip high-speed RAM and clock generation via PLL.

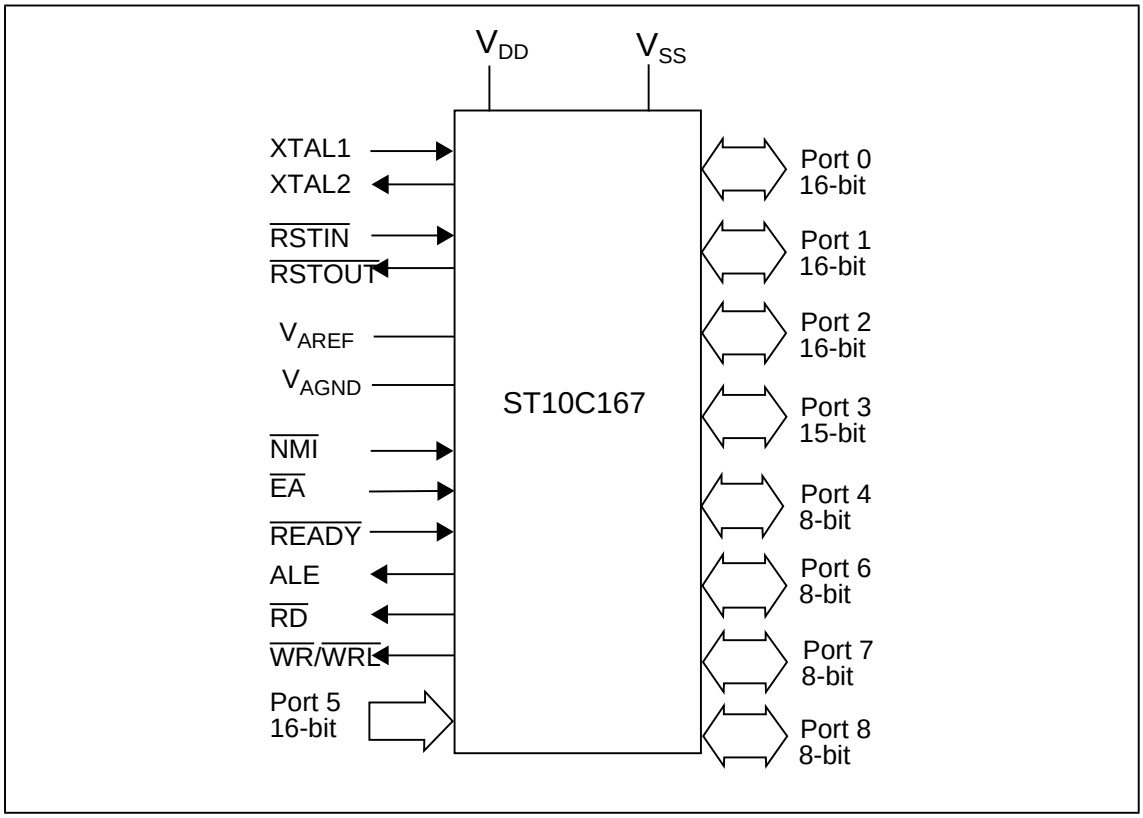



**Figure 1 Logic symbol**

# 2 Pin Data

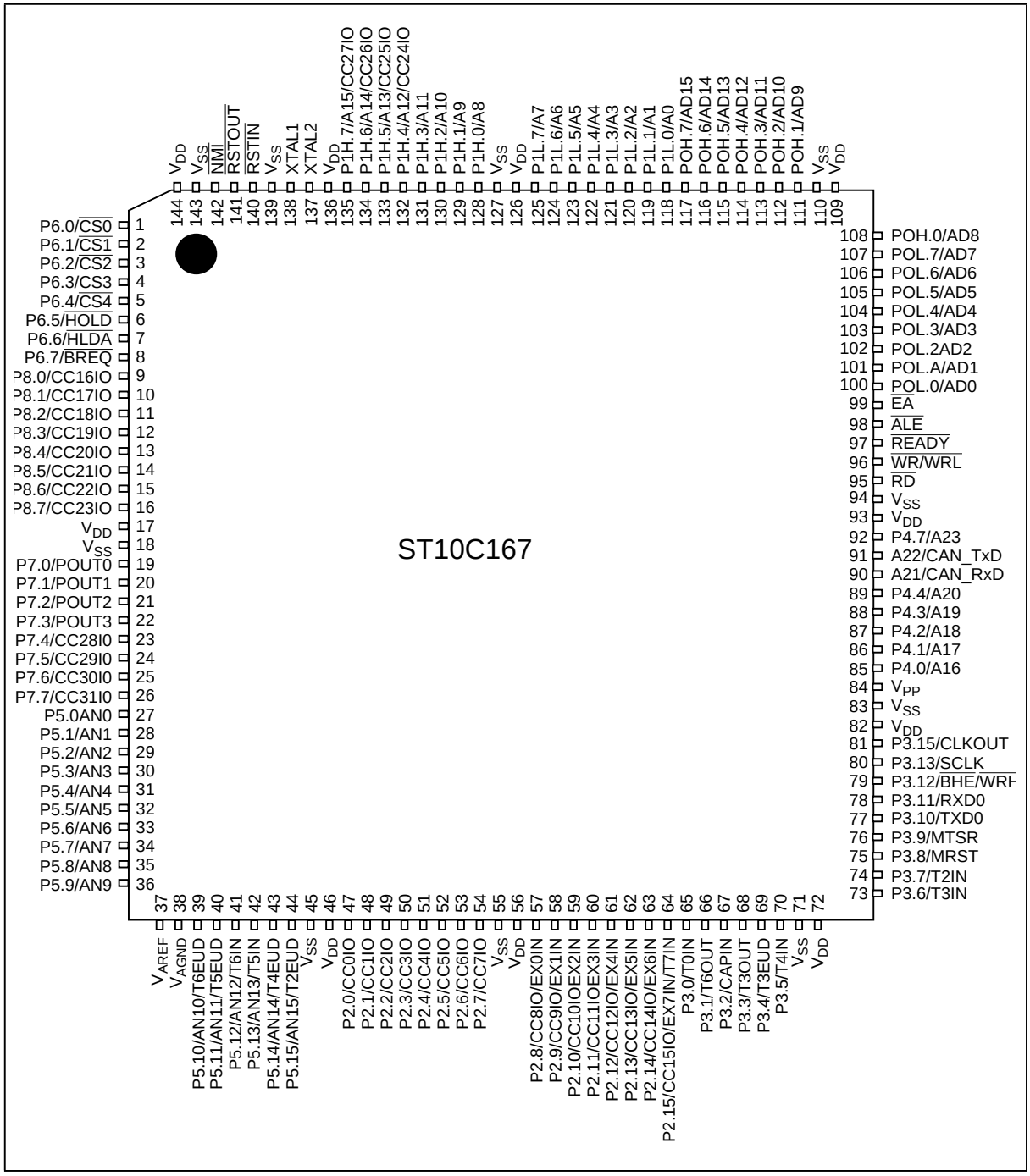



**Figure 2 Pin Configuration (top view)**

## 2 Pin Data

| Symbol | Pin Number | Input (I) Output(O) | Function |
|---|---|---|---|
| P6.0 –P6.7 | 1 - 8 | 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.<br>The following Port 6 pins also serve for alternate functions: |
| | 1 | O | P6.0$\overline{CS0}$Chip Select 0 Output |
| | ... | ... | ..... |
| | 5 | O | P6.4$\overline{CS4}$Chip Select 4 Output |
| | 6 | I | P6.5$\overline{HOLD}$External Master Hold Request Input |
| | 7 | O | P6.6$\overline{HLDA}$Hold Acknowledge Output |
| | 8 | O | P6.7$\overline{BREQ}$Bus Request Output |
| P8.0 –P8.7 | 9 - 16 | I/O | Port 8 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 8 outputs can be configured as push/pull or open drain drivers. The input threshold of Port 8 is selectable (TTL or special).<br>The following Port 8 pins also serve for alternate functions: |
| | 9 | I/O | P8.0CC16IOCAPCOM2: CC16 Cap.-In/Comp.Out |
| | ... | ... | ..... |
| | 16 | I/O | P8.7CC23IOCAPCOM2: CC23 Cap.-In/Comp.Out |

**Table 1 Pin description**

## 2 Pin Data

| Symbol | Pin Number | Input (I) Output(O) | Function |
|---|---|---|---|
| P7.0 –P7.7 | 19 -26 | I/O | Port 7 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 7 outputs can be configured as push/pull or open drain drivers. The input threshold of Port 7 is selectable (TTL or special).<br>The following Port 7 pins also serve for alternate functions: |
| | 19 | O | P7.0POUT0PWM Channel 0 Output |
| | ... | ... | ..... |
| | 22 | O | P7.3POUT3PWM Channel 3 Output |
| | 23 | I/O | P7.4CC28IOCAPCOM2: CC28 Cap.-In/Comp.Out |
| | ... | ... | ..... |
| | 26 | I/O | P7.7CC31IOCAPCOM2: CC31 Cap.-In/Comp.Out |
| P5.0 –P5.15 | 27-36<br>39-44 | I<br>I | Port 5 is a 16-bit input-only port with Schmitt-Trigger characteristics. The pins of Port 5 also serve as the (up to 16) analog input channels for the A/D converter, where P5.x equals ANx (Analog input channel x), or they serve as timer inputs: |
| | 39 | I | P5.10T6EUDGPT2 Timer T6 Ext.Up/Down Ctrl.Input |
| | 40 | I | P5.11T5EUDGPT2 Timer T5 Ext.Up/Down Ctrl.Input |
| | 41 | I | P5.12T6INGPT2 Timer T6 Count Input |
| | 42 | I | P5.13T5INGPT2 Timer T5 Count Input |
| | 43 | I | P5.14T4EUDGPT1 Timer T4 Ext.Up/Down Ctrl.Input |
| | 44 | I | P5.15T2EUDGPT1 Timer T2 Ext.Up/Down Ctrl.Input |

**Table 1 Pin description (Continued)**

## 2 Pin Data

| Symbol | Pin Number | Input (I) Output(O) | Function |
|---|---|---|---|
| P2.0 –P2.15 | 47-54<br>57-64 | I/O | Port 2 is a 16-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 input threshold of Port 2 is selectable (TTL or special).<br>The following Port 2 pins also serve for alternate functions: |
| | 47 | I/O | P2.0CC0IOCAPCOM: CC0 Cap.-In/Comp.Out |
| | ... | ... | ..... |
| | 54 | I/O | P2.7CC7IOCAPCOM: CC7 Cap.-In/Comp.Out |
| | 57 | I/O | P2.8CC8IOCAPCOM: CC8 Cap.-In/Comp.Out, |
| | | I | EX0INFast External Interrupt 0 Input |
| | ... | ... | ..... |
| | 64 | I/O | P2.15CC15IOCAPCOM: CC15 Cap.-In/Comp.Out, |
| | | I | EX7INFast External Interrupt 7 Input |
| | | I | T7INCAPCOM2 Timer T7 Count Input |

**Table 1 Pin description (Continued)**

## 2 Pin Data

| Symbol | Pin Number | Input (I) Output(O) | Function |
|---|---|---|---|
| P3.0-P3.13, P3.15 | 65-70, 73-0, 81 | 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 input threshold of Port 3 is selectable (TTL or special). The following Port 3 pins also serve for alternate functions: |
| | 65 | I | P3.0T0INCAPCOM Timer T0 Count Input |
| | 66 | O | P3.1T6OUTGPT2 Timer T6 Toggle Latch Output |
| | 67 | I | P3.2CAPINGPT2 Register CAPREL Capture Input |
| | 68 | O | P3.3T3OUTGPT1 Timer T3 Toggle Latch Output |
| | 69 | I | P3.4T3EUDGPT1 Timer T3 Ext.Up/Down Ctrl.Input |
| | 70 | I | P3.5T4INGPT1 Timer T4 Input for Count/Gate/Reload/Capture |
| | 73 | I | P3.6T3INGPT1 Timer T3 Count/Gate Input |
| | 74 | I | P3.7T2INGPT1 Timer T2 Input for Count/Gate/Reload/Capture |
| | 75 | I/O | P3.8MRSTSSC Master-Rec./Slave-Transmit I/O |
| | 76 | I/O | P3.9MTSRSSC Master-Transmit/Slave-Rec. O/I |
| | 77 | I/O | P3.10TxD0ASC0 Clock/Data Output (Asyn./Syn.) |
| | 78 | O | P3.11RxD0ASC0 Data Input (Asyn.) or I/O (Syn.) |
| | 79 | O | P3.12$\overline{BHE}$Ext. Memory High Byte Enable Signal, $\overline{WRH}$Ext. Memory High Byte Write Strobe |
| | 80 | I/O | P3.13SCLKSSC Master Clock Outp./Slave Cl. Inp. |
| | 81 | O | P3.15CLKOUTSystem Clock Output (=CPU Clock) |

**Table 1 Pin description (Continued)**

## 2 Pin Data

| Symbol | Pin Number | Input (I) Output(O) | Function |
|---|---|---|---|
| P4.0 –P4.7 | 85-92 | 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.<br>In case of an external bus configuration, Port 4 can be used to output the segment address lines: |
| | 85 | O | P4.0A16Least Significant Segment Addr. Line |
| | 90 | O | P4.5A21Segment Address Line, |
| | | I | CAN_RxDCAN Receive Data Input |
| | 91 | O | P4.6A22Segment Address Line, CAN_TxD |
| | | O | CAN Transmit Data Output |
| | 92 | O | P4.7A23Most Significant Segment Addr. Line |
| RD | 95 | O | External Memory Read Strobe. $\overline{RD}$ is activated for every external instruction or data read access. |
| WR/WRL | 96 | O | External Memory Write Strobe. In $\overline{WR}$-mode this pin is activated for every external data write access. In $\overline{WRL}$-mode this pin is activated for low byte data write accesses on a 16-bit bus, and for every data write access on an 8-bit bus. See WRCFG in register SYSCON for mode selection. |
| READY/ $\overline{READY}$ | 97 | I | Ready Input. The active level is programmable. When the Ready function is enabled, the selected inactive level at this pin during an external memory access will force the insertion of memory cycle time waitstates until the pin returns to the selected active level. |
| ALE | 98 | O | Address Latch Enable Output. Can be used for latching the address into external memory or an address latch in the multiplexed bus modes. |
| EA | 99 | I | External Access Enable pin. A low level at this pin during and after Reset forces the ST10C167 to begin instruction execution out of external memory. A high level forces execution out of the internal Flash Memory. |

**Table 1 Pin description (Continued)**

## 2 Pin Data

| Symbol | Pin Number | Input (I) Output(O) | Function |
|---|---|---|---|
| PORT0:<br>P0L.0-<br>P0L.7,<br>P0H.0-<br>P0H.7 | <br>100-<br>107,<br>108,<br>111-<br>117 | 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.<br>In case of an external bus configuration, PORT0 serves as the address (A) and address/data (AD) bus in multiplexed bus modes and as the data (D) bus in demultiplexed bus modes.<br>**Demultiplexed bus modes:**<br>Data Path Width:8-bit16-bit<br>P0L.0 – P0L.7:D0 – D7D0 - D7<br>P0H.0 – P0H.7:I/OD8 - D15<br>**Multiplexed bus modes:**<br>Data Path Width:8-bit16-bit<br>P0L.0 – P0L.7:AD0 – AD7AD0 - AD7<br>P0H.0 – P0H.7:A8 - A15AD8 - AD15 |
| PORT1:<br>P1L.0-<br>P1L.7,<br>P1H.0-<br>P1H.7 | <br>118-<br>125<br>128-<br>135 | 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.<br>The following PORT1 pins also serve for alternate functions: |
| | 132 | I | P1H.4CC24IOCAPCOM2: CC24 Capture Input |
| | 133 | I | P1H.5CC25IOCAPCOM2: CC25 Capture Input |
| | 134 | I | P1H.6CC26IOCAPCOM2: CC26 Capture Input |
| | 135 | I | P1H.7CC27IOCAPCOM2: CC27 Capture Input |
| XTAL1 | 138 | I | XTAL1:Input to the oscillator amplifier and input to the internal clock generator |
| XTAL2 | 137 | O | XTAL2:Output of the oscillator amplifier circuit.<br>To clock the device from an external source, drive XTAL1, while leaving XTAL2 unconnected. Minimum and maximum high/low and rise/fall times specified in the AC Characteristics must be observed. |

**Table 1 Pin description (Continued)**

## 2 Pin Data

| Symbol | Pin Number | Input (I) Output(O) | Function |
|---|---|---|---|
| RSTIN | 140 | I | Reset Input with Schmitt-Trigger characteristics. A low level at this pin for a specified duration while the oscillator is running resets the ST10C167. An internal pullup resistor permits power-on reset using only a capacitor connected to $V_{SS}$.<br>In bidirectional reset mode (enabled by setting bit BDRSTEN in SYSCON register), the RSTIN line is pulled low for the duration of the internal reset sequence. |
| RSTOUT | 141 | 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. |
| NMI | 142 | I | Non-Maskable Interrupt Input. A high to low transition at this pin causes the CPU to vector to the NMI trap routine. If bit PWDCFG = '0' in SYSCON register, when the PWRDN (power down) instruction is executed, the $\overline{\text{NMI}}$ pin must be low in order to force the ST10C167 to go into power down mode. If $\overline{\text{NMI}}$ is high and PWDCFG ='0', when PWRDN is executed, the part will continue to run in normal mode.<br>If not used, pin $\overline{\text{NMI}}$ should be pulled high externally. |
| $V_{AREF}$ | 37 | - | Reference voltage for the A/D converter. |
| $V_{AGND}$ | 38 | - | Reference ground for the A/D converter. |
| $V_{PP}$/RPD | 84 | - | Flash programming voltage. This pin accepts the programming voltage for ST10F167 derivatives with on-chip flash memory.<br>It is used also as the timing pin for the return from powerdown circuit and power-up asynchronous reset. |
| $V_{DD}$ | 17, 46, 56, 72, 82, 93, 109, 126, 136, 144 | - | Digital Supply Voltage:<br>+ 5 V during normal operation and idle mode.<br>≥ 2.5 V during power down mode |

**Table 1 Pin description (Continued)**

## 2 Pin Data

| Symbol | Pin Number | Input (I) Output(O) | Function |
|---|---|---|---|
| $V_{SS}$ | 18, 45, 55, 71, 83, 94, 110, 127, 139, 143 | - | Digital Ground. |

**Table 1 Pin description (Continued)**

# 3 Functional Description

The architecture of the ST10C167 combines advantages of both RISC and CISC processors and an advanced peripheral subsystem. The block diagram gives an overview of the different on-chip components and the high bandwidth internal bus structure of the ST10C167.

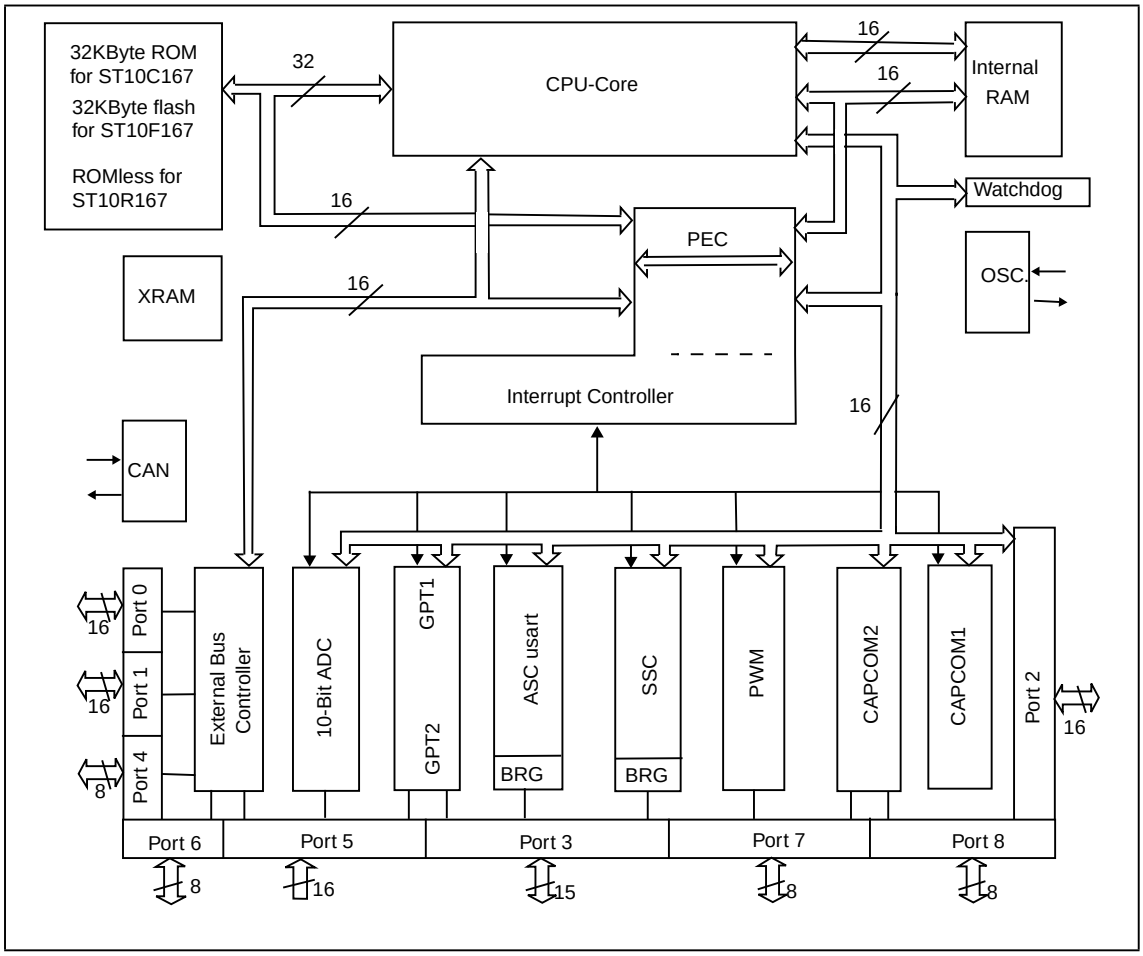



**Figure 3 Block diagram**

# 4 Memory Organization

The memory space of the ST10C167 is configured in a Von Neumann architecture. Code memory, data memory, registers and I/O ports are organized within the same linear address space of 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.

**ROM:** 32KBytes of on-chip ROM.

**IRAM:** 2KByte of on-chip internal RAM (dual-port) is provided as a storage for data, system stack, general purpose register banks and code. The register bank can consist of up to 16 wordwide (R0 to R15) and/or bytewide (RL0, RH0, …, RL7, RH7) general purpose registers.

**XRAM:** 2KByte of on-chip extension RAM (single port XRAM) is provided as a storage for data, user stack and code. The XRAM is connected to the internal XBUS and is accessed like an external memory in 16-bit demultiplexed bus-mode without waitstate or read/write delay (80 ns access @ 25 MHz CPU clock). Byte and word access is allowed.
The XRAM address range is 00’E000h - 00’E7FFh if the XRAM is enabled (XPEN bit 2 of SYSCON register). As the XRAM appears like external memory, it cannot be used for the ST10C167’s system stack or register banks. The XRAM is not provided for single bit storage and therefore is not bit addressable. If bit XRAMEN is cleared, then any access in the address range 00’E000h - 00’E7FFh will be directed to external memory interface, using the BUSCONx register corresponding to address matching ADDRSELx register.

**SFR/ESFR:** 1024 bytes (2 * 512 bytes) of address space is reserved for the special function register areas. SFRs are wordwide registers which are used for controlling and monitoring functions of the different on-chip units.

**CAN:** Address range 00’EF00h - 00’EFFFh is reserved for the CAN Module access. The CAN is enabled by setting XPEN bit 2 of the SYSCON register. Accesses to the CAN Module use demultiplexed addresses and a 16-bit data bus (byte accesses are possible). Two waitstates give an access time of 160 ns @25 Mhz CPU clock. No tristate waitstate is used.

*Note* *If the CAN module is used, Port 4 can not be programmed to output all 8 segment address lines. Thus, only 4 segment address lines can be used, reducing the external memory space to 5 MBytes (1 MByte per $\overline{CS}$ line)*

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.

# 5 Central Processing Unit (CPU)

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

Most of the ST10C167's instructions can be executed in one instruction cycle which requires 80ns at 25MHz CPU clock. For example, shift and rotate instructions are processed in one instruction cycle independent of the number of bits to be shifted. Multiple-cycle instructions have been optimized: branches are carried out in 2 cycles, 16 x 16 bit multiplication in 5 cycles and a 32/16 bit division in 10 cycles.The jump cache reduces the execution time of repeatedly performed jumps in a loop, from 2 cycles to 1 cycle.

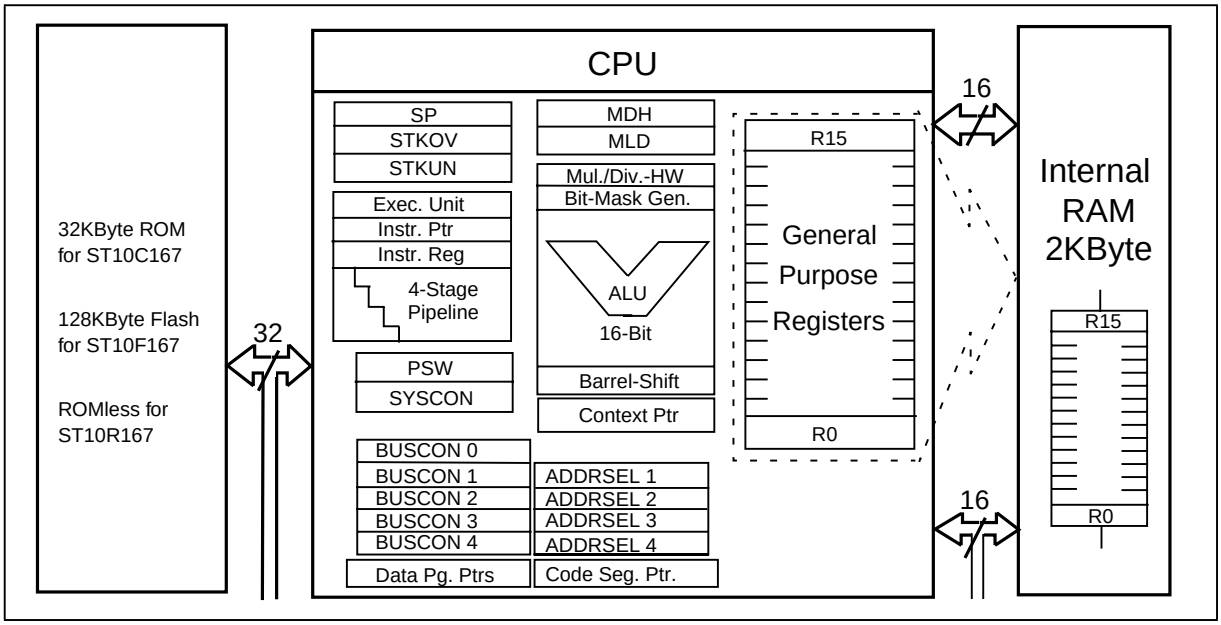



**Figure 4 CPU block diagram**

The CPU uses an actual register context consisting of up to 16 wordwide GPRs 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. 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.

# 6 External Bus Controller

All of the external memory accesses are performed by the on-chip external bus controller. The EBC can be programmed to single chip mode when no external memory is required, or to one of four different external memory access modes:

- 16-/18-/20-/24-bit addresses16-bit data, demultiplexed
- 16-/18-/20-/24-bit addresses16-bit data, multiplexed
- 16-/18-/20-/24-bit addresses8-bit data, multiplexed
- 16-/18-/20-/24-bit addresses8-bit data, demultiplexed

In 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.

Timing characteristics of the external bus interface (memory cycle time, memory tri-state time, length of ale and read write delay) are programmable giving the choice of a wide range of memories and external peripherals. Up to 4 independent address windows may be defined (using register pairs ADDRSELx / BUSCONx) to access different resources and bus characteristics. These address windows are arranged hierarchically where BUSCON4 overrides BUSCON3 and BUSCON2 overrides BUSCON1. All accesses to locations not covered by these 4 address windows are controlled by BUSCON0. Up to 5 external $\overline{CS}$ signals (4 windows plus default) can be generated in order to save external glue logic. Access to very slow memories is supported by a ‘Ready’ function.

A $\overline{HOLD}$/$\overline{HLDA}$ protocol is available for bus arbitration which shares 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 directly connects the slave controller to another master controller without glue logic.

For applications which require less external memory space, the address space can be restricted to 1 MByte, 256 KByte or to 64 KByte. Port 4 outputs all 8 address lines if an address space of 16 MBytes is used, otherwise four, two or no address lines.

Chip select timing can be made programmable. By default (after reset), the CSx lines change half a CPU clock cycle after the rising edge of ALE. With the CSCFG bit set in the SYSCON register the CSx lines change with the rising edge of ALE.

The active level of the READY pin can be set by bit RDYPOL in the BUSCONx registers. When the READY function is enabled for a specific address window, each bus cycle within the window must be terminated with the active level defined by bit RDYPOL in the associated BUSCON register.

# 7 Interrupt System

The interrupt response time for internal program execution is from 200 ns to 480ns.

The ST10C167 architecture 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 serviced by the Interrupt Controller or by the Peripheral Event Controller (PEC).

In contrast to a standard interrupt service where the current program execution is suspended and a branch to the interrupt vector table is performed, just one cycle is ‘stolen’ from the current CPU activity to perform a PEC service. A PEC service implies a single byte or word data transfer between any two memory locations with an additional increment of either the PEC source or the destination pointer. An individual PEC transfer counter is implicitly decremented for each PEC service except when performing in the continuous transfer mode. When this counter reaches zero, a standard interrupt is performed to the corresponding source related vector location. PEC services are very well suited, for example, for supporting the transmission or reception of blocks of data. The ST10C167 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 ST10C167 interrupt sources and the corresponding hardware-related interrupt flags, vectors, vector locations and trap (interrupt) numbers:

## 7 Interrupt System

| Source of Interrupt or PEC Service Request | Request Flag | Enable Flag | Interrupt Vector | Vector Location | Trap Number |
|---|---|---|---|---|---|
| CAPCOM Register 0 | CC0IR | CC0IE | CC0INT | 00’0040h | 10h |
| CAPCOM Register 1 | CC1IR | CC1IE | CC1INT | 00’0044h | 11h |
| CAPCOM Register 2 | CC2IR | CC2IE | CC2INT | 00’0048h | 12h |
| CAPCOM Register 3 | CC3IR | CC3IE | CC3INT | 00’004Ch | 13h |
| CAPCOM Register 4 | CC4IR | CC4IE | CC4INT | 00’0050h | 14h |
| CAPCOM Register 5 | CC5IR | CC5IE | CC5INT | 00’0054h | 15h |
| CAPCOM Register 6 | CC6IR | CC6IE | CC6INT | 00’0058h | 16h |
| CAPCOM Register 7 | CC7IR | CC7IE | CC7INT | 00’005Ch | 17h |
| CAPCOM Register 8 | CC8IR | CC8IE | CC8INT | 00’0060h | 18h |
| CAPCOM Register 9 | CC9IR | CC9IE | CC9INT | 00’0064h | 19h |
| CAPCOM Register 10 | CC10IR | CC10IE | CC10INT | 00’0068h | 1Ah |
| CAPCOM Register 11 | CC11IR | CC11IE | CC11INT | 00’006Ch | 1Bh |
| CAPCOM Register 12 | CC12IR | CC12IE | CC12INT | 00’0070h | 1Ch |
| CAPCOM Register 13 | CC13IR | CC13IE | CC13INT | 00’0074h | 1Dh |
| CAPCOM Register 14 | CC14IR | CC14IE | CC14INT | 00’0078h | 1Eh |
| CAPCOM Register 15 | CC15IR | CC15IE | CC15INT | 00’007Ch | 1Fh |
| CAPCOM Register 16 | CC16IR | CC16IE | CC16INT | 00’00C0h | 30h |
| CAPCOM Register 17 | CC17IR | CC17IE | CC17INT | 00’00C4h | 31h |
| CAPCOM Register 18 | CC18IR | CC18IE | CC18INT | 00’00C8h | 32h |
| CAPCOM Register 19 | CC19IR | CC19IE | CC19INT | 00’00CCh | 33h |
| CAPCOM Register 20 | CC20IR | CC20IE | CC20INT | 00’00D0h | 34h |
| CAPCOM Register 21 | CC21IR | CC21IE | CC21INT | 00’00D4h | 35h |

**Table 2 Interrupt sources**

## 7 Interrupt System

| Source of Interrupt or PEC Service Request | Request Flag | Enable Flag | Interrupt Vector | Vector Location | Trap Number |
|---|---|---|---|---|---|
| CAPCOM Register 22 | CC22IR | CC22IE | CC22INT | 00’00D8h | 36h |
| CAPCOM Register 23 | CC23IR | CC23IE | CC23INT | 00’00DCh | 37h |
| CAPCOM Register 24 | CC24IR | CC24IE | CC24INT | 00’00E0h | 38h |
| CAPCOM Register 25 | CC25IR | CC25IE | CC25INT | 00’00E4h | 39h |
| CAPCOM Register 26 | CC26IR | CC26IE | CC26INT | 00’00E8h | 3Ah |
| CAPCOM Register 27 | CC27IR | CC27IE | CC27INT | 00’00ECh | 3Bh |
| CAPCOM Register 28 | CC28IR | CC28IE | CC28INT | 00’00E0h | 3Ch |
| CAPCOM Register 29 | CC29IR | CC29IE | CC29INT | 00’0110h | 44h |
| CAPCOM Register 30 | CC30IR | CC30IE | CC30INT | 00’0114h | 45h |
| CAPCOM Register 31 | CC31IR | CC31IE | CC31INT | 00’0118h | 46h |
| CAPCOM Timer 0 | T0IR | T0IE | T0INT | 00’0080h | 20h |
| CAPCOM Timer 1 | T1IR | T1IE | T1INT | 00’0084h | 21h |
| CAPCOM Timer 7 | T7IR | T7IE | T7INT | 00’00F4h | 3Dh |
| CAPCOM Timer 8 | T8IR | T8IE | T8INT | 00’00F8h | 3Eh |
| GPT1 Timer 2 | T2IR | T2IE | T2INT | 00’0088h | 22h |
| GPT1 Timer 3 | T3IR | T3IE | T3INT | 00’008Ch | 23h |
| GPT1 Timer 4 | T4IR | T4IE | T4INT | 00’0090h | 24h |
| GPT2 Timer 5 | T5IR | T5IE | T5INT | 00’0094h | 25h |
| GPT2 Timer 6 | T6IR | T6IE | T6INT | 00’0098h | 26h |
| GPT2 CAPREL Register | CRIR | CRIE | CRINT | 00’009Ch | 27h |
| A/D Conversion Complete | ADCIR | ADCIE | ADCINT | 00’00A0h | 28h |
| A/D Overrun Error | ADEIR | ADEIE | ADEINT | 00’00A4h | 29h |
| ASC0 Transmit | S0TIR | S0TIE | S0TINT | 00’00A8h | 2Ah |

**Table 2 Interrupt sources (Continued)**

## 7 Interrupt System

| Source of Interrupt or PEC Service Request | Request Flag | Enable Flag | Interrupt Vector | Vector Location | Trap Number |
|---|---|---|---|---|---|
| ASC0 Transmit Buffer | S0TBIR | S0TBIE | S0TBINT | 00’011Ch | 47h |
| ASC0 Receive | S0RIR | S0RIE | S0RINT | 00’00ACh | 2Bh |
| ASC0 Error | S0EIR | S0EIE | S0EINT | 00’00B0h | 2Ch |
| SSC Transmit | SCTIR | SCTIE | SCTINT | 00’00B4h | 2Dh |
| SSC Receive | SCRIR | SCRIE | SCRINT | 00’00B8h | 2Eh |
| SSC Error | SCEIR | SCEIE | SCEINT | 00’00BCh | 2Fh |
| PWM Channel 0...3 | PWMIR | PWMIE | PWMINT | 00’00FCh | 3Fh |
| CAN Interface | XP0IR | XP0IE | XP0INT | 00’0100h | 40h |
| X-Peripheral Node | XP1IR | XP1IE | XP1INT | 00’0104h | 41h |
| X-Peripheral Node | XP2IR | XP2IE | XP2INT | 00’0108h | 42h |
| PLL Unlock | XP3IR | XP3IE | XP3INT | 00’010Ch | 43h |

**Table 2 Interrupt sources (Continued)**

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

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

# 7 Interrupt System

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

**Table 3 Exceptions or error conditions that can arise during run-time**

# 8 Capture/Compare (CAPCOM) Units

The ST10C167 has two 16 channel CAPCOM units. They support generation and control of timing sequences on up to 32 channels with a maximum resolution of 320ns at 25MHz CPU clock. The CAPCOM units are typically used to handle high speed I/O tasks such as pulse and waveform generation, pulse width modulation (PMW), Digital to Analog (D/A) conversion, software timing, or time recording relative to external events.

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

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

Each of the two capture/compare register arrays contain 16 dual purpose capture/compare registers, each of which may be individually allocated to either CAPCOM timer T0 or T1 (T7 or T8, respectively), and programmed for capture or compare functions. Each register has one associated port pin which serves as an input pin for triggering the capture function, or as an output pin (except for CC24...CC27) to indicate the occurrence of a compare event.

When a capture/compare register has been selected for capture mode, the current contents of the allocated timer will be latched (captured) into the capture/compare register in response to an external event at the port pin which is associated with this register. In addition, a specific interrupt request for this capture/compare register is generated. Either a positive, a negative, or both a positive and a negative external signal transition at the pin can be selected as the triggering event. The contents of all registers which have been selected for one of the five compare modes are continuously compared with the contents of the allocated timers. When a match occurs between the timer value and the value in a capture/compare register, specific actions will be taken based on the selected compare mode.

## 8 Capture/Compare (CAPCOM) Units

| Compare Modes | Function |
|---|---|
| Mode 0 | Interrupt-only compare mode;<br>several compare interrupts per timer period are possible |
| Mode 1 | Pin toggles on each compare match;<br>several compare events per timer period are possible |
| Mode 2 | Interrupt-only compare mode;<br>only one compare interrupt per timer period is generated |
| Mode 3 | Pin set ‘1’ on match; pin reset ‘0’ on compare time overflow;<br>only one compare event per timer period is generated |
| Double Register Mode | Two registers operate on one pin; pin toggles on each compare match;<br>several compare events per timer period are possible. |

**Table 4 Compare modes**

The input frequencies $f_{TX}$ for Tx are determined as a function of the CPU clocks. The formulas are detailed in the user manual. The timer input frequencies, resolution and periods which result from the selected pre-scaler option in TxI when using a 25MHz CPU clock are listed in the table below. The numbers for the timer periods are based on a reload value of $0000_H$. Note that some numbers may be rounded to 3 significant figures.

| **$f_{CPU}$ = 25MHz** | **Timer Input Selection TxI** | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | $000_B$ | $001_B$ | $010_B$ | $011_B$ | $100_B$ | $101_B$ | $110_B$ | $111_B$ |
| **Pre-scaler for $f_{CPU}$** | 8 | 16 | 32 | 64 | 128 | 256 | 512 | 1024 |
| **Input Frequency** | 3.125 MHz | 1.56 MHz | 781 kHz | 391 kHz | 195 kHz | 97.7 kHz | 48.8 kHz | 24.4 kHz |
| **Resolution** | 320ns | 640ns | 1.28 s | 2.56 s | 5.12 s | 10.24 s | 20.48 s | 40.96 s |
| **Period** | 21.0 ms | 41.9 ms | 83.9 ms | 167 ms | 336 ms | 671 ms | 1.34 s | 2.68 s |

**Table 5 CAPCOM timer input frequencies, resolution and periods**

# 9 General Purpose Timer Unit

The GPT unit is a flexible multifunctional timer/counter structure which is used for time related tasks such as event timing and counting, pulse width and duty cycle measurements, pulse generation, or pulse multiplication. The GPT unit contains five 16-bit timers organized into two separate modules GPT1 and GPT2. Each timer in each module may operate independently in several different modes, or may be concatenated with another timer of the same module.

## 9.1 GPT1

Each of the three timers T2, T3, T4 of the GPT1 module can be configured individually for one of four basic modes of operation: **timer**, **gated timer**, **counter mode and incremental interface mode**. In timer mode, the input clock for a timer is derived from the CPU clock, divided by a programmable prescaler. In counter mode, the timer is 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.

Table 6 lists the timer input frequencies, resolution and periods for each pre-scaler option at 25 MHz CPU clock. This also applies to the Gated Timer Mode of T3 and to the auxiliary timers T2 and T4 in Timer and Gated Timer Mode.

| **$f_{CPU}$ = 25MHz** | **Timer Input Selection T2I / T3I / T4I** | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | $000_B$ | $001_B$ | $010_B$ | $011_B$ | $100_B$ | $101_B$ | $110_B$ | $111_B$ |
| Pre-scaler factor | 8 | 16 | 32 | 64 | 128 | 256 | 512 | 1024 |
| Input Freq | 3.125 MHz | 1.563 MHz | 781.3 kHz | 390.6 kHz | 195.3 kHz | 97.66 kHz | 48.83 kHz | 24.41 kHz |
| Resolu-tion | 320 ns | 640 ns | 128 ns | 2.56 s | 5.12 s | 10.24 s | 20.48 s | 40.96 s |
| Period | 21.0 ms | 41.9 ms | 83.9 ms | 167 ms | 336 ms | 671 ms | 1.34 s | 2.68 s |

**Table 6 GPT1 timer input frequencies, resolution and periods**

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).

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

Timer T3 has output toggle latches (TxOTL) which changes state on each timer over-flow/ underflow. The state of this latch may be output on port pins (TxOUT) 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.

## 9.2 GPT2

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

The state of this latch may be used to clock timer T5, 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.

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 advantageous when T3 operates in Incremental Interface Mode.

Table 7 lists the timer input frequencies, resolution and periods for each pre-scaler option at 25 MHz CPU clock. This also applies to the Gated Timer Mode of T6 and to the auxiliary timer T5 in Timer and Gated Timer Mode.

| $f_{CPU}$ = 25MHz | Timer Input Selection T5I / T6I | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 000B | 001B | 010B | 011B | 100B | 101B | 110B | 111B |
| Pre-scaler factor | 4 | 8 | 16 | 32 | 64 | 128 | 256 | 512 |
| Input Freq | 6.25 MHz | 3.125 MHz | 1.563 MHz | 781.3 kHz | 390.6 kHz | 195.3 kHz | 97.66 kHz | 48.83 kHz |
| Resolution | 160ns | 320 ns | 640 ns | 128 ns | 2.56 s | 5.12 s | 10.24 s | 20.48 s |
| Period | 10.49 ms | 21.0 ms | 41.9 ms | 83.9 ms | 167 ms | 336 ms | 671 ms | 1.34 s |

**Table 7 GPT2 timer input frequencies, resolution and period**

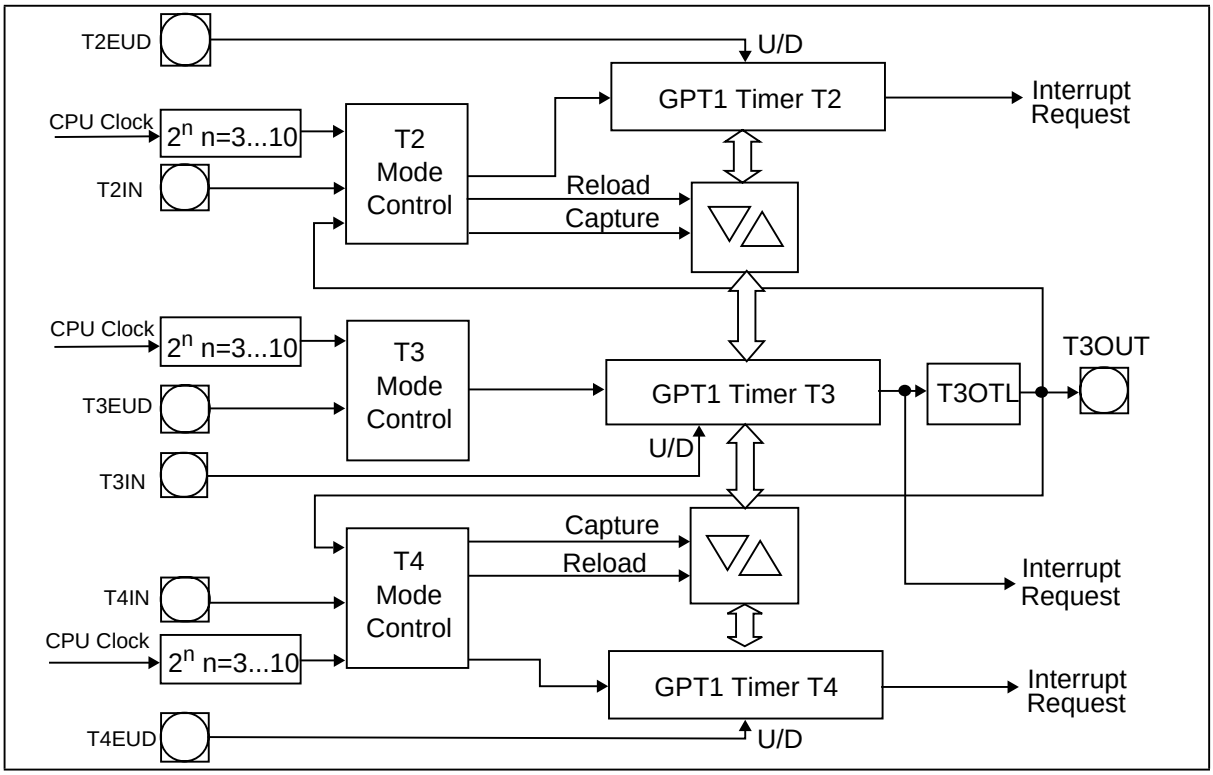



**Figure 5 Block diagram of GPT1**

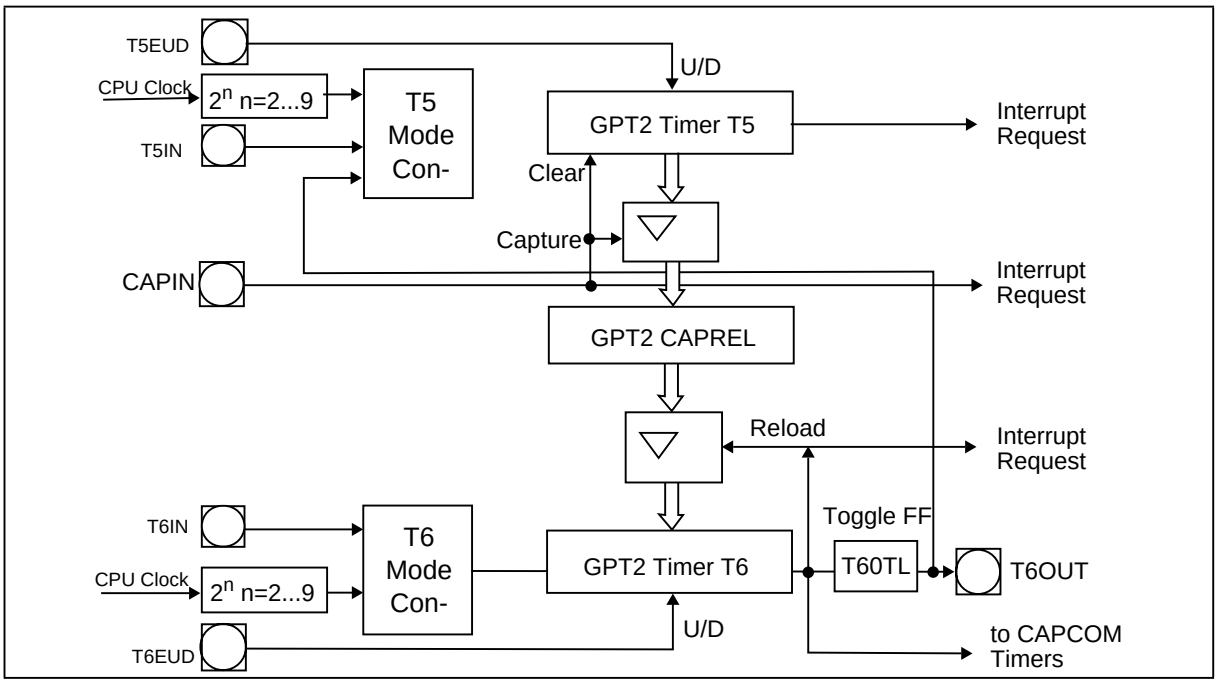



**Figure 6 Block diagram of GPT2**

# 10 PWM Module

The pulse width modulation module can generate up to four PWM output signals using edge-aligned or centre-aligned PWM. In addition, the PWM module can generate PWM burst signals and single shot outputs. The table below shows the PWM frequencies for different resolutions. The level of the output signals is selectable and the PWM module can generate interrupt requests.

| **Mode 0** | **Resolution** | **8-bit** | **10-bit** | **12-bit** | **14-bit** | **16-bit** |
|---|---|---|---|---|---|---|
| CPU Clock/1 | 40ns | 97.66 KHz | 24.41KHz | 6.104KHz | 1.526KHz | 0.381KHz |
| CPU Clock/64 | 2.56ns | 1.526KHz | 381.5 Hz | 95.37Hz | 23.84Hz | 5.96Hz |
| **Mode 1** | **Resolution** | **8-bit** | **10-bit** | **12-bit** | **14-bit** | **16-bit** |
| CPU Clock/1 | 40ns | 48.82KHz | 12.20KHz | 3.05KHz | 762.9Hz | 190.7Hz |
| CPU Clock/64 | 2.56ns | 762.9Hz | 190.7 Hz | 47.68Hz | 11.92Hz | 0Hz |

**Table 8 PWM unit frequencies and resolution at 25MHz CPU clock**

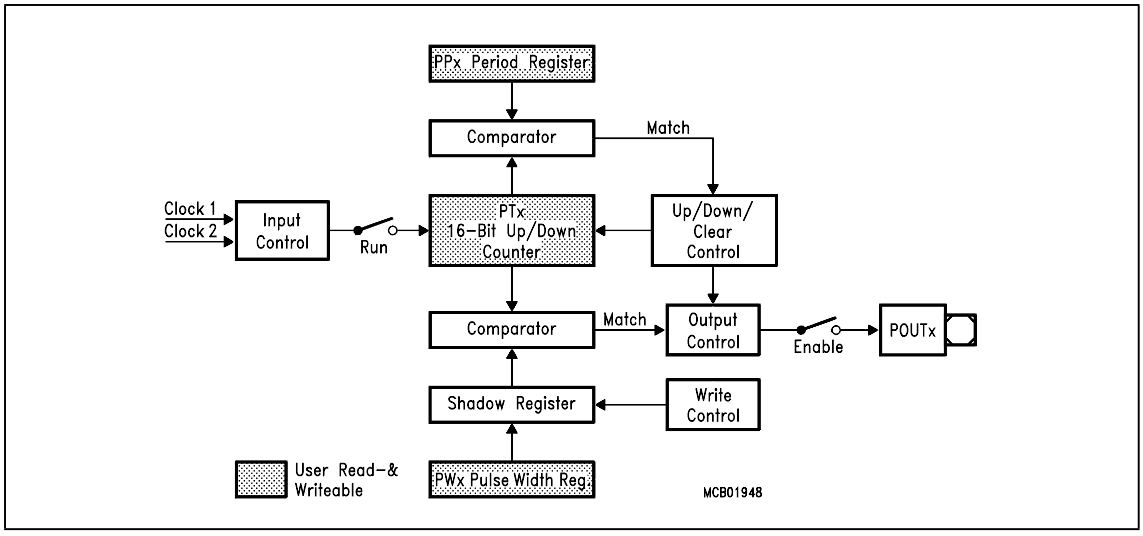



**Figure 7 Block diagram of PWM module**

# 11 Parallel Ports

The ST10C167 provides up to 111 I/O lines organized into eight 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 five 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.

The input threshold of Port 2, Port 3, Port 7 and Port 8 is selectable (TTL or CMOS like), where the special CMOS like input threshold reduces noise sensitivity due to the input hysteresis. The input threshold may be selected individually for each byte of the respective ports.

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

The standard output drivers of the ST10C167 have a drive capability of 8 mA. But in order to reduce chip consumption and also noise generated by level transition of output pins, the ST10C167 offers programmable output drivers on Port 2, Port 3, Port 7 and Port 8 that can be switched by software from 8 mA strength to 4 mA strength. The high byte of the PICON register is used to select the output buffer strength for each byte of the indicated ports, i.e. the 8-bit port P7 and P8 are controlled by one bit each while ports P2 and P3 are controlled by two bits each.

Port 2, Port 8 and Port 7 are associated with the capture inputs or compare outputs of the CAPCOM units and/or with the outputs of the PWM module.
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 the analog input channels to the A/D converter or timer control signals.

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

# 12 A/D converter

A10-bit A/D converter with 16 multiplexed input channels and a sample and hold circuit is integrated on-chip. The sample time (for loading the capacitors) and the conversion time is programmable and can be adjusted to the external circuitry.

Overrun error detection/protection is controlled by the ADDAT register. Either an interrupt request is generated when the result of a previous conversion has not been read from the result register at the time the next conversion is complete, or the next conversion is suspended until the previous result has been read. For applications which require less than 16 analog input channels, the remaining channel inputs can be used as digital input port pins.

The A/D converter of the ST10C167 supports four different conversion modes:

**Single channel conversion mode** the analog level on a specified channel is sampled once and converted to a digital result.

**Single channel continuous mode** the analog level on a specified channel is repeatedly sampled and converted without software intervention.

**Auto scan mode** the analog levels on a pre-specified number of channels are sequentially sampled and converted.

**Auto scan continuous mode** the number of pre-specified channels is repeatedly sampled and converted.

Channel Injection Mode injects a channel into a running sequence without disturbing this sequence. The peripheral event controller stores the conversion results in memory without entering and exiting interrupt routines for each data transfer.

The following table shows the ADC unit conversion clock, sample clock and complete conversion times.

| ADCTC | Conversion clock tcc | ADSTC | Sample clock tsc | Complete conversion |
|---|---|---|---|---|
| 00 | 0.48 s | 00 | 0.48 s | 7.76 s |
| 01 | reserved | 01 | reserved | |
| 10 | 1.92 s | 10 | 7.68 s | 42.32 s |
| 11 | 0.96 s | 11 | 7.68 s | 28.88 s |

**Table 9 ADC sample clock and complete conversion times**

The A/D converter provides automatic offset and linearity self calibration. The calibration operation is performed in two ways:

- A full calibration sequence is performed after a reset and lasts 1.6 ms minimum (@ 25MHz CPU clock). During this time, the ADBSY flag is set to indicate the operation. Normal conversion can be performed during this time. The duration of the calibration sequence is then extended by the time consumed by the conversions.

*Note* *After a power-on reset, the total unadjusted error (TUE) of the ADC might be worse than +-2 LSB (max. +-4 LSB). During the full calibration sequence, the TUE is constantly improved until at the end of the cycle, TUE is within the specified limits of +-2 LSB.*

- One calibration cycle is performed after each conversion: each calibration cycle takes 4 ADC clock cycles. These operation cycles ensure constant updating of the ADC's accuracy, compensating changing operating conditions.

# 13 Serial Channels

Serial communication with other microcontrollers, processors, terminals or external peripheral components is provided by two serial interfaces: the asynchronous/synchronous serial channel (ASC0) and the high-speed synchronous serial channel (SSC). Two dedicated baud rate generators set up all standard baud rates without oscillator tuning. For transmission, reception and erroneous reception, 3 separate interrupt vectors are provided for each serial channel.

## ASCO

Supports full-duplex asynchronous communication up to 781.25 KBaud and half-duplex synchronous communication up to 5MBaud @ 25MHz system clock. For asynchronous operation, the Baud rate generator provides a clock with 16 times the rate of the established Baud rate. The table below lists various commonly used baud rates together with the required reload values and the deviation errors compared to the intended baudrate.

| **S0BRS = '0', $f_{CPU}$ = 25MHz** | | | **S0BRS = '1', $f_{CPU}$ = 25MHz** | | |
|---|---|---|---|---|---|
| **Baud Rate (Baud)** | **Deviation Error** | **Reload Value** | **Baud Rate (Baud)** | **Deviation Error** | **Reload Value** |
| 781250 | 0.0% | $0000_H$ | 520833 | 0.0% | $0000_H$ |

**Table 10 Commonly used baud rates by reload value and deviation errors**

## 13 Serial Channels

| S0BRS = '0', $f_{CPU}$ = 25MHz | | | S0BRS = '1', $f_{CPU}$ = 25MHz | | |
|---|---|---|---|---|---|
| **Baud Rate (Baud)** | **Deviation Error** | **Reload Value** | **Baud Rate (Baud)** | **Deviation Error** | **Reload Value** |
| 56000 | +7.3%/ -0.4% | $000C_H$ / $000D_H$ | 56000 | +3.3%/ -7.0% | 0008H / 0009H |
| 38400 | +1.7%/ -3.1% | $0013_H$ / $0014_H$ | 38400 | +4.3%/ -3.1% | 000CH / 000DH |
| 19200 | +1.7%/ -0.8% | $0027_H$ / $0028_H$ | 19200 | +0.5%/ -3.1% | 001AH / 001BH |
| 9600 | +0.5%/ -0.8% | $0050_H$/ $0051_H$ | 9600 | +0.5%/ -1.4% | 0035H / 0036H |
| 4800 | +0.5%/ -0.1% | $00A1_H$ / $00A2_H$ | 4800 | +0.5%/ -0.5% | 006BH / 006CH |
| 2400 | +0.2%/ -0.1% | $0144_H$ / $0145_H$ | 2400 | +0.0%/ -0.5% | 00D8H / 00D9H |
| 1200 | +0.0%/ -0.1% | $028A_H$ / $028B_H$ | 1200 | +0.0%/ -0.2% | 01B1H / 01B2H |
| 600 | +0.0%/ -0.1% | $0515_H$ / $0516_H$ | 600 | +0.0%/ -0.1% | 0363H / 0364H |
| 95 | +0.4%/ 0.4% | $1FFF_H$ / $1FFF_H$ | 75 | +0.0%/ 0.0% | 1B1FH / 1B20H |
| | | | 63 | +0.9%/ 0.9% | 1FFFH / 1FFFH |

**Table 10 Commonly used baud rates by reload value and deviation errors**

*Note* *The deviation errors given in the table above are rounded. Using a baudrate crystal will provide correct baudrates without deviation errors.*

For synchronous operation, the Baud rate generator provides a clock with 4 times the rate of the established Baud rate.

### High Speed Synchronous Serial Channel (SSC)

The High-Speed Synchronous Serial Interface SSC provides flexible high-speed serial communication between the ST10C167 and other microcontrollers, microprocessors or external peripherals.

The SSC supports full-duplex and half-duplex synchronous communication; The serial clock signal can be generated by the SSC itself (master mode) or be received from an external master (slave mode). Data width, shift direction, clock polarity and phase are programmable. This allows communication with SPI-compatible devices. Transmission and reception of data is double-buffered. A 16-bit baud rate generator provides the SSC with a separate serial clock signal. The serial channel SSC has its own dedicated 16-bit baud rate generator with 16-bit reload capability, allowing baud rate generation independent from the timers.

## 13 Serial Channels

SSCBR is the dual-function Baud Rate Generator/Reload register. The table below lists some possible baud rates against the required reload values and the resulting bit times for a 25MHz CPU clock.

| Baud Rate | Bit Time | Reload Value |
|---|---|---|
| Reserved use a reload value > 0. | --- | $0000_H$ |
| 5MBaud | 200ns | $0001_H$ |
| 3.3MBaud | 303ns | $0002_H$ |
| 2.5MBaud | 400ns | $0004_H$ |
| 2MBaud | 500ns | $0005_H$ |
| 1MBaud | 1µs | $000B_H$ |
| 100KBaud | 10µs | $007C_H$ |
| 10KBaud | 100µs | $04E1_H$ |
| 1KBaud | 1ms | $30D3_H$ |
| 190.7Baud | 5.2ms | $FFFF_H$ |

**Table 11 Synchronous baud rate and reload values**

# 14 CAN Module

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

Provides full CAN functionality on up to 15 message objects. Message object 15 can be configured for basic CAN functionality. Both modes provide separate masks for acceptance filtering, allowing a number of identifiers in full CAN mode to be accepted and disregarding a number of identifiers in basic CAN mode. All message objects can be updated independentl from other objects and are equipped for the maximum message length of 8 bytes.

The bit timing is derived from the XCLK and is programmable up to a data rate of 1 MBaud. The CAN module uses two pins to interface to a bus transceiver.

# 15 Watchdog Timer

The Watchdog Timer is a fail-safe mechanism which prevents the microcontroller from malfunctioning for long 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. Therefore, the chip's start-up procedure is always monitored. The software must 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. It pulls the $\overline{\text{RSTOUT}}$ pin low in order to allow external hardware components to be reset.

The Watchdog Timer is 16-bit, clocked with the system clock divided by 2 or 128. The high byte of the watchdog timer register can be set to a pre-specified reload value (stored in WDTREL). Each time it is serviced by the application software, the high byte of the watchdog timer is reloaded. *For security, rewrite WDTCON each time before the watchdog timer is serviced*

The table below shows the watchdog time range for 25MHz CPU clock.

| **Reload value in WDTREL** | **Prescaler for $f_{CPU}$** | |
|---|---|---|
| | **2 (WDTIN = '0')** | **128 (WDTIN = '1')** |
| $FF_H$ | 20.48 s | 1.31 ms |
| $00_H$ | 5.24 ms | 336 ms |

# 16 Instruction Set Summary

The table below lists the instructions of the ST10C167. The various addressing modes, instruction operation, parameters for conditional execution of instructions, opcodes and a detailed description of each instruction can be found in the “ST10 Family Programming Manual”.

| 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 |

**Table 12 Instruction set summary**

## 16 Instruction Set Summary

| Mnemonic | Description | Bytes |
|---|---|---|
| **CMPD1/2** | Compare word data to GPR and decrement GPR by 1/2 | 2 / 4 |
| **CMPI1/2** | Compare word data to GPR and increment GPR by 1/2 | 2 / 4 |
| **PRIOR** | Determine number of shift cycles to normalize direct word GPR and store result in direct word GPR | 2 |
| **SHL / SHR** | Shift left/right direct word GPR | 2 |
| **ROL / ROR** | Rotate left/right direct word GPR | 2 |
| **ASHR** | Arithmetic (sign bit) shift right direct word GPR | 2 |
| **MOV(B)** | Move word (byte) data | 2 / 4 |
| **MOVBS** | 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 and update register with word operand | 4 |
| **RET** | Return from intra-segment subroutine | 2 |
| **RETS** | Return from inter-segment subroutine | 2 |

**Table 12 Instruction set summary**

## 16 Instruction Set Summary

| Mnemonic | Description | Bytes |
|---|---|---|
| **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 |

**Table 12 Instruction set summary**

# 17 System Reset

The internal system reset function is invoked either by asserting a hardware reset signal on pin $\overline{\text{RSTIN}}$ (Hardware Reset Input), by the execution of the SRST instruction (Software Reset) or by an overflow of the watchdog timer. Whenever one of these conditions occurs, the microcontroller is reset into its predefined default state. The following type of reset are implemented on the ST10C167:

**Asynchronous hardware reset:** Asynchronous reset does not require a stabilized clock signal on XTAL1, as it is not internally resynchronized. It immediately resets the microcontroller into its default reset state. This asynchronous reset is required upon power-up of the chip and may be used during catastrophic situations. The rising edge of the RSTIN pin is internally resynchronized before exiting the reset condition. Therefore, only the entry of the this hardware reset is asynchronous.

**Synchronous hardware reset (warm reset):** A warm synchronous hardware reset is triggered when the reset input signal $\overline{\text{RSTIN}}$ is latched low and Vpp pin is high. The I/Os are immediately (asynchronously) set in high impedance, $\overline{\text{RSTOUT}}$ is driven low. After $\overline{\text{RSTIN}}$ negation is detected, a short transition period elapses, during which pending internal hold states are cancelled and any current internal access cycles are completed, external bus cycles are aborted. Then, the internal reset sequence starts for 1024 TCL (512 CPU clock cycles). During this reset sequence, if bit BDRSTEN was previously set by software (bit 5 in SYSCON register), $\overline{\text{RSTIN}}$ pin is driven low and internal reset signal is asserted to reset the microcontroller in its default state. Note that after all reset sequence, bit BDRSTEN is cleared. After the reset sequence has been completed, the $\overline{\text{RSTIN}}$ input is sampled. When the reset input signal is active at that time the internal reset condition is prolonged until $\overline{\text{RSTIN}}$ becomes inactive.

**Software reset:** The reset sequence can be triggered at any time by the protected instruction SRST (software reset). This instruction can be executed deliberately within a program, e.g. to leave bootstrap loader mode, or on a hardware trap that reveals a system failure. As for a synchronous hardware reset, the reset sequence lasts 1024 TCL (512 CPU clock cycles), and drives the $\overline{\text{RSTIN}}$ pin low.

**Watchdog timer reset:** When the watchdog timer is not disabled during the initialization or serviced regularly during program execution is will overflow and trigger the reset sequence. Unlike hardware and software resets, the watchdog reset completes a running external bus cycle if this bus cycle does not use $\overline{\text{READY}}$, or if $\overline{\text{READY}}$ is sampled active (low) after the programmed waitstates. When $\overline{\text{READY}}$ is sampled inactive (high) after the programmed waitstates the running external bus cycle is aborted. Then the internal reset sequence is started. The watchdog reset cannot occur while the ST10C167 is in bootstrap loader mode.

**Bidirectional reset:** This feature is enabled by bit 3 of the SYSCON register. The bidirectional reset makes the watchdog timer reset and software reset externally visible. It is active for the duration of an internal reset sequences caused by a watchdog timer reset and

software reset. This means that the bidirectional reset transforms an internal watchdog timer reset or software reset into an external hardware reset with a minimum duration of 1024 TCL. The consequence is that during a watchdog timer reset or software reset, the behavior of the C167CR-4RM is equal to an external hardware reset.

# 18 Power Reduction Modes

Two different power reduction modes are implemented on the ST10C167.

**Idle mode:** The CPU is stopped, while the peripherals continue their operation. Idle mode can be terminated by any reset or interrupt request.

**Power Down mode:** Clocking of all internal blocks is stopped, the contents of the internal RAM, however, are preserved through the voltage supplied via the $V_{CC}$ pins. The watchdog timer is stopped

Two different power down modes are implemented:

**Protected power down mode:** This is used in conjunction with an external power failure signal. The microcontroller enters the $\overline{\text{NMI}}$ trap routine which saves the internal state into RAM. After the internal state has been saved, the trap routine may set a flag or write a certain bit patterns into specific RAM locations and then execute the PWRDN instruction. If the $\overline{\text{NMI}}$ pin is still low at this time, power down mode will be entered, otherwise program execution continues. During power down the voltage at the $V_{CC}$ pins can be lowered to 2.5V conserving the contents of the internal RAM.The initialization routine (executed upon reset) can check the identification flag or bit pattern within RAM to determine whether the controller was initially switched on, or whether it was properly restarted from power down mode.

**Interruptible power down mode:** When power down mode is entered, the CPU and peripheral clocks are frozen, and the oscillator and PLL are stopped. To exit power down mode with external interrupt, an EXxIN pin has to be asserted for at least 40 ns (x = 7...0). This signal enables the internal oscillator and PLL circuitry, and turns on the weak pull-down. If the Interrupt was enabled before entering power down mode, the device executes the interrupt service routine, and then resumes execution after the PWRDN instruction. If the interrupt was disabled, the device executes the instruction following PWRDN instruction, and the Interrupt Request Flag remains set until it is cleared by software.

# 19 Special Function Register Overview

The following table lists all SFRs which are implemented in the ST10C167 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”.

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

| Name | Physical address | 8-bit address | Description | Reset value |
|---|---|---|---|---|
| **ADCICb** | FF98h | CCh | A/D converter end of conversion interrupt control reg | 0000h |
| **ADCONb** | FFA0h | D0h | A/D Converter Control Register | 0000h |
| **ADDAT** | FEA0h | 50h | A/D Converter Result Register | 0000h |
| **ADDAT2** | F0A0h**E** | 50h | A/D Converter 2 Result Register | 0000h |
| **ADDRSEL1** | FE18h | 0Ch | Address Select Register 1 | 0000h |
| **ADDRSEL2** | FE1Ah | 0Dh | Address Select Register 2 | 0000h |
| **ADDRSEL3** | FE1Ch | 0Eh | Address Select Register 3 | 0000h |
| **ADDRSEL4** | FE1Eh | 0Fh | Address Select Register 4 | 0000h |
| **ADEICb** | FF9Ah | CDh | A/D converter overrun error interrupt control reg | 0000h |
| **BUSCON0b** | FF0Ch | 86h | Bus Configuration Register 0 | 0XX0h |
| **BUSCON1b** | FF14h | 8Ah | Bus Configuration Register 1 | 0000h |
| **BUSCON2b** | FF16h | 8Bh | Bus Configuration Register 2 | 0000h |
| **BUSCON3b** | FF18h | 8Ch | Bus Configuration Register 3 | 0000h |
| **BUSCON4b** | FF1Ah | 8Dh | Bus Configuration Register 4 | 0000h |
| **CAPREL** | FE4Ah | 25h | GPT2 Capture/Reload Register | 0000h |
| **CC8ICb** | FF88h | C4h | EX0IN Interrupt Control Register | 0000h |

**Table 13 Special function registers listed by name**

## 19 Special Function Register Overview

| Name | Physical address | 8-bit address | Description | Reset value |
|---|---|---|---|---|
| **CC0** | FE80h | 40h | CAPCOM Register 0 | 0000h |
| **CC0ICb** | FF78h | BCh | CAPCOM Register 0 Interrupt Control Reg | 0000h |
| **CC1** | FE82h | 41h | CAPCOM Register 1 | 0000h |
| **CC1ICb** | FF7Ah | BDh | CAPCOM Register 1 Interrupt Control Reg | 0000h |
| **CC2** | FE84h | 42h | CAPCOM Register 2 | 0000h |
| **CC2ICb** | FF7Ch | BEh | CAPCOM Register 2 Interrupt Control Reg | 0000h |
| **CC3** | FE86h | 43h | CAPCOM Register 3 | 0000h |
| **CC3ICb** | FF7Eh | BFh | CAPCOM Register 3 Interrupt Control Reg | 0000h |
| **CC4** | FE88h | 44h | CAPCOM Register 4 | 0000h |
| **CC4ICb** | FF80h | C0h | CAPCOM Register 4 Interrupt Control Reg | 0000h |
| **CC5** | FE8Ah | 45h | CAPCOM Register 5 | 0000h |
| **CC5ICb** | FF82h | C1h | CAPCOM Register 5 Interrupt Control Reg | 0000h |
| **CC6** | FE8Ch | 46h | CAPCOM Register 6 | 0000h |
| **CC6ICb** | FF84h | C2h | CAPCOM Register 6 Interrupt Control Reg | 0000h |
| **CC7** | FE8Eh | 47h | CAPCOM Register 7 | 0000h |
| **CC7ICb** | FF86h | C3h | CAPCOM Register 7 Interrupt Control Reg | 0000h |
| **CC8** | FE90h | 48h | CAPCOM Register 8 | 0000h |
| **CC8ICb** | FF88h | C4h | CAPCOM Register 8 Interrupt Control Reg | 0000h |
| **CC9** | FE92h | 49h | CAPCOM Register 9 | 0000h |
| **CC9ICb** | FF8Ah | C5h | CAPCOM Register 9 Interrupt Control Reg | 0000h |
| **CC10** | FE94h | 4Ah | CAPCOM Register 10 | 0000h |
| **CC10ICb** | FF8Ch | C6h | CAPCOM Register 10 Interrupt Control Reg | 0000h |
| **CC11** | FE96h | 4Bh | CAPCOM Register 11 | 0000h |
| **CC11ICb** | FF8Eh | C7h | CAPCOM Register 11 Interrupt Control Reg | 0000h |

**Table 13 Special function registers listed by name**

## 19 Special Function Register Overview

| Name | Physical address | 8-bit address | Description | Reset value |
|---|---|---|---|---|
| **CC12** | FE98h | 4Ch | CAPCOM Register 12 | 0000h |
| **CC12ICb** | FF90h | C8h | CAPCOM Register 12 Interrupt Control Reg | 0000h |
| **CC13** | FE9Ah | 4Dh | CAPCOM Register 13 | 0000h |
| **CC13ICb** | FF92h | C9h | CAPCOM Register 13 Interrupt Control Reg | 0000h |
| **CC14** | FE9Ch | 4Eh | CAPCOM Register 14 | 0000h |
| **CC14ICb** | FF94h | CAh | CAPCOM Register 14 Interrupt Control Reg | 0000h |
| **CC15** | FE9Eh | 4Fh | CAPCOM Register 15 | 0000h |
| **CC15ICb** | FF96h | CBh | CAPCOM Register 15 Interrupt Control Reg | 0000h |
| **CC16** | FE60h | 30h | CAPCOM Register 16 | 0000h |
| **CC16ICb** | F160h**E** | B0h | CAPCOM Register 16 Interrupt Control Reg | 0000h |
| **CC17** | FE62h | 31h | CAPCOM Register 17 | 0000h |
| **CC17ICb** | F162h**E** | B1h | CAPCOM Register 17 Interrupt Control Reg | 0000h |
| **CC18** | FE64h | 32h | CAPCOM Register 18 | 0000h |
| **CC18ICb** | F164h**E** | B2h | CAPCOM Register 18 Interrupt Control Reg | 0000h |
| **CC19** | FE66h | 33h | CAPCOM Register 19 | 0000h |
| **CC19ICb** | F166h**E** | B3h | CAPCOM Register 19 Interrupt Control Reg | 0000h |
| **CC20** | FE68h | 34h | CAPCOM Register 20 | 0000h |
| **CC20ICb** | F168h**E** | B4h | CAPCOM Register 20 Interrupt Control Reg | 0000h |
| **CC21** | FE6Ah | 35h | CAPCOM Register 21 | 0000h |
| **CC21ICb** | F16Ah**E** | B5h | CAPCOM Register 21 Interrupt Control Reg | 0000h |
| **CC22** | FE6Ch | 36h | CAPCOM Register 22 | 0000h |
| **CC22ICb** | F16Ch**E** | B6h | CAPCOM Register 22 Interrupt Control Reg | 0000h |
| **CC23** | FE6Eh | 37h | CAPCOM Register 23 | 0000h |
| **CC23ICb** | F16Eh**E** | B7h | CAPCOM Register 23 Interrupt Control Reg | 0000h |

**Table 13 Special function registers listed by name**

## 19 Special Function Register Overview

| Name | Physical address | 8-bit address | Description | Reset value |
|---|---|---|---|---|
| **CC24** | FE70h | 38h | CAPCOM Register 24 | 0000h |
| **CC24ICb** | F170h**E** | B8h | CAPCOM Register 24 Interrupt Control Reg | 0000h |
| **CC25** | FE72h | 39h | CAPCOM Register 25 | 0000h |
| **CC25ICb** | F172h**E** | B9h | CAPCOM Register 25 Interrupt Control Reg | 0000h |
| **CC26** | FE74h | 3Ah | CAPCOM Register 26 | 0000h |
| **CC26ICb** | F174h**E** | BAh | CAPCOM Register 26 Interrupt Control Reg | 0000h |
| **CC27** | FE76h | 3Bh | CAPCOM Register 27 | 0000h |
| **CC27ICb** | F176h**E** | BBh | CAPCOM Register 27 Interrupt Control Reg | 0000h |
| **CC28** | FE78h | 3Ch | CAPCOM Register 28 | 0000h |
| **CC28ICb** | F178h**E** | BCh | CAPCOM Register 28 Interrupt Control Reg | 0000h |
| **CC29** | FE7Ah | 3Dh | CAPCOM Register 29 | 0000h |
| **CC29ICb** | F184h**E** | C2h | CAPCOM Register 29 Interrupt Control Reg | 0000h |
| **CC30** | FE7Ch | 3Eh | CAPCOM Register 30 | 0000h |
| **CC30ICb** | F18Ch**E** | C6h | CAPCOM Register 30 Interrupt Control Reg | 0000h |
| **CC31** | FE7Eh | 3Fh | CAPCOM Register 31 | 0000h |
| **CC31ICb** | F194h**E** | CAh | CAPCOM Register 31 Interrupt Control Reg | 0000h |
| **CCM0b** | FF52h | A9h | CAPCOM Mode Control Register 0 | 0000h |
| **CCM1b** | FF54h | AAh | CAPCOM Mode Control Register 1 | 0000h |
| **CCM2b** | FF56h | ABh | CAPCOM Mode Control Register 2 | 0000h |
| **CCM3b** | FF58h | ACh | CAPCOM Mode Control Register 3 | 0000h |
| **CCM4b** | FF22h | 91h | CAPCOM Mode Control Register 4 | 0000h |
| **CCM5b** | FF24h | 92h | CAPCOM Mode Control Register 5 | 0000h |
| **CCM6b** | FF26h | 93h | CAPCOM Mode Control Register 6 | 0000h |
| **CCM7b** | FF28h | 94h | CAPCOM Mode Control Register 7 | 0000h |

**Table 13 Special function registers listed by name**

## 19 Special Function Register Overview

| Name | Physical address | 8-bit address | Description | Reset value |
|---|---|---|---|---|
| **CP** | FE10h | 08h | CPU Context Pointer Register | FC00h |
| **CRICb** | FF6Ah | B5h | GPT2 CAPREL Interrupt Control Register | 0000h |
| **CSP** | FE08h | 04h | CPU Code Segment Pointer Reg (read only) | 0000h |
| **DP0Lb** | $F100h_E$ | 80h | P0L Direction Control Register | 00h |
| **DP0Hb** | $F102h_E$ | 81h | P0h Direction Control Register | 00h |
| **DP1Lb** | $F104h_E$ | 82h | P1L Direction Control Register | 00h |
| **DP1Hb** | $F106h_E$ | 83h | P1h Direction Control Register | 00h |
| **DP2b** | FFC2h | E1h | Port 2 Direction Control Register | 0000h |
| **DP3b** | FFC6h | E3h | Port 3 Direction Control Register | 0000h |
| **DP4b** | FFCAh | E5h | Port 4 Direction Control Register | 00h |
| **DP6b** | FFCEh | E7h | Port 6 Direction Control Register | 00h |
| **DP7b** | FFD2h | E9h | Port 7 Direction Control Register | 00h |
| **DP8b** | FFD6h | EBh | Port 8 Direction Control Register | 00h |
| **DPP0** | FE00h | 00h | CPU Data Page Pointer 0 Register (10 bits) | 0000h |
| **DPP1** | FE02h | 01h | CPU Data Page Pointer 1 Register (10 bits) | 0001h |
| **DPP2** | FE04h | 02h | CPU Data Page Pointer 2 Register (10 bits) | 0002h |
| **DPP3** | FE06h | 03h | CPU Data Page Pointer 3 Register (10 bits) | 0003h |
| **EXICONb** | $F1C0h_E$ | E0h | External Interrupt Control Register | 0000h |
| **IDCHIP** | $F07Ch_E$ | 3Eh | Device Identifier Register | 0A7h |
| **IDMANUF** | $F07Eh_E$ | 3Fh | Manufacturer Identifier Register | 0020h |
| **IDMEM** | $F07Ah_E$ | 3Dh | On-chip Memory Identifier Register | 3020h |
| **IDPROG** | $F078h_E$ | 3Ch | Programming Voltage Identifier Register | 9A40h |
| **MDCb** | FF0Eh | 87h | CPU Multiply Divide Control Register | 0000h |

**Table 13 Special function registers listed by name**

## 19 Special Function Register Overview

| Name | Physical address | 8-bit address | Description | Reset value |
|---|---|---|---|---|
| **MDH** | FE0Ch | 06h | CPU Multiply Divide Register – High Word | 0000h |
| **MDL** | FE0Eh | 07h | CPU Multiply Divide Register – Low Word | 0000h |
| **ODP2b** | F1C2h**E** | E1h | Port 2 Open Drain Control Register | 0000h |
| **ODP3b** | F1C6h**E** | E3h | Port 3 Open Drain Control Register | 0000h |
| **ODP6b** | F1CEh**E** | E7h | Port 6 Open Drain Control Register | 00h |
| **ODP7b** | F1D2h**E** | E9h | Port 7 Open Drain Control Register | 00h |
| **ODP8b** | F1D6h**E** | EBh | Port 8 Open Drain Control Register | 00h |
| **ONES** | FF1Eh | 8Fh | Constant Value 1's Register (read only) | FFFFh |
| **P0Lb** | FF00h | 80h | Port 0 Low Register (Lower half of PORT0) | 00h |
| **P0Hb** | FF02h | 81h | Port 0 High Register (Upper half of PORT0) | 00h |
| **P1Lb** | FF04h | 82h | Port 1 Low Register (Lower half of PORT1) | 00h |
| **P1Hb** | FF06h | 83h | Port 1 High Register (Upper half of PORT1) | 00h |
| **P2b** | FFC0h | E0h | Port 2 Register | 0000h |
| **P3b** | FFC4h | E2h | Port 3 Register | 0000h |
| **P4b** | FFC8h | E4h | Port 4 Register (8 bits) | 00h |
| **P5b** | FFA2h | D1h | Port 5 Register (read only) | XXXXh |
| **P6b** | FFCCh | E6h | Port 6 Register (8 bits) | 00h |
| **P7b** | FFD0h | E8h | Port 7 Register (8 bits) | 00h |
| **P8b** | FFD4h | EAh | Port 8 Register (8 bits) | 00h |
| **PECC0** | FEC0h | 60h | PEC Channel 0 Control Register | 0000h |
| **PECC1** | FEC2h | 61h | PEC Channel 1 Control Register | 0000h |
| **PECC2** | FEC4h | 62h | PEC Channel 2 Control Register | 0000h |
| **PECC3** | FEC6h | 63h | PEC Channel 3 Control Register | 0000h |

**Table 13 Special function registers listed by name**

## 19 Special Function Register Overview

| Name | Physical address | 8-bit address | Description | Reset value |
|---|---|---|---|---|
| **PECC4** | FEC8h | 64h | PEC Channel 4 Control Register | 0000h |
| **PECC5** | FECAh | 65h | PEC Channel 5 Control Register | 0000h |
| **PECC6** | FECCh | 66h | PEC Channel 6 Control Register | 0000h |
| **PECC7** | FECEh | 67h | PEC Channel 7 Control Register | 0000h |
| **PICON** | F1C4h**E** | E2h | Port Input Threshold Control Register | 0000h |
| **PP0** | F038h**E** | 1Ch | PWM Module Period Register 0 | 0000h |
| **PP1** | F03Ah**E** | 1Dh | PWM Module Period Register 1 | 0000h |
| **PP2** | F03Ch**E** | 1Eh | PWM Module Period Register 2 | 0000h |
| **PP3** | F03Eh**E** | 1Fh | PWM Module Period Register 3 | 0000h |
| **PSWb** | FF10h | 88h | CPU Program Status Word | 0000h |
| **PT0** | F030h**E** | 18h | PWM Module Up/Down Counter 0 | 0000h |
| **PT1** | F032h**E** | 19h | PWM Module Up/Down Counter 1 | 0000h |
| **PT2** | F034h**E** | 1Ah | PWM Module Up/Down Counter 2 | 0000h |
| **PT3** | F036h**E** | 1Bh | PWM Module Up/Down Counter 3 | 0000h |
| **PW0** | FE30h | 18h | PWM Module Pulse Width Register 0 | 0000h |
| **PW1** | FE32h | 19h | PWM Module Pulse Width Register 1 | 0000h |
| **PW2** | FE34h | 1Ah | PWM Module Pulse Width Register 2 | 0000h |
| **PW3** | FE36h | 1Bh | PWM Module Pulse Width Register 3 | 0000h |
| **PWMCON0b** | FF30h | 98h | PWM Module Control Register 0 | 0000h |
| **PWMCON1b** | FF32h | 99h | PWM Module Control Register 1 | 0000h |
| **PWMICb** | F17Eh**E** | BFh | PWM Module Interrupt Control Register | 0000h |
| **RP0Hb** | $F108h_{\mathbf{E}}$ | 84h | System Start-up configuration reg (read only) | XXh |
| **S0BG** | FEB4h | 5Ah | Serial Channel 0 baud rate generator reload register | 0000h |

**Table 13 Special function registers listed by name**

## 19 Special Function Register Overview

| Name | Physical address | 8-bit address | Description | Reset value |
|---|---|---|---|---|
| **S0CONb** | FFB0h | D8h | Serial Channel 0 Control Register | 0000h |
| **S0EICb** | FF70h | B8h | Serial Channel 0 Error Interrupt Control Reg | 0000h |
| **S0RBUF** | FEB2h | 59h | Serial Channel 0 receive buffer reg (read only) | XXh |
| **S0RICb** | FF6Eh | B7h | Serial Channel 0 receive interrupt control reg | 0000h |
| **S0TBICb** | F19Ch**E** | CEh | Serial Channel 0 transmit buffer interrupt control reg | 0000h |
| **S0TBUF** | FEB0h | 58h | Serial Channel 0 transmit buffer register (write only) | 00h |
| **S0TICb** | FF6Ch | B6h | Serial Channel 0 transmit interrupt control reg | 0000h |
| **SP** | FE12h | 09h | CPU System Stack Pointer Register | FC00h |
| **SSCBR** | F0B4h**E** | 5Ah | SSC Baudrate Register | 0000h |
| **SSCCONb** | FFB2h | D9h | SSC Control Register | 0000h |
| **SSCEICb** | FF76h | BBh | SSC Error Interrupt Control Register | 0000h |
| **SSCRB** | F0B2h**E** | 59h | SSC Receive Buffer (read only) | XXXXh |
| **SSCRICb** | FF74h | BAh | SSC Receive Interrupt Control Register | 0000h |
| **SSCTB** | F0B0h**E** | 58h | SSC Transmit Buffer (write only) | 0000h |
| **SSCTICb** | FF72h | B9h | SSC Transmit Interrupt Control Register | 0000h |
| **STKOV** | FE14h | 0Ah | CPU Stack Overflow Pointer Register | FA00h |
| **STKUN** | FE16h | 0Bh | CPU Stack Underflow Pointer Register | FC00h |
| **SYSCONb** | FF12h | 89h | CPU System Configuration Register | 0xx0h[1)] |
| **T0** | FE50h | 28h | CAPCOM Timer 0 Register | 0000h |
| **T01CONb** | FF50h | A8h | CAPCOM Timer 0 and Timer 1 Control Reg | 0000h |
| **T0ICb** | FF9Ch | CEh | CAPCOM Timer 0 Interrupt Control Register | 0000h |
| **T0REL** | FE54h | 2Ah | CAPCOM Timer 0 Reload Register | 0000h |

**Table 13 Special function registers listed by name**

## 19 Special Function Register Overview

| Name | Physical address | 8-bit address | Description | Reset value |
|---|---|---|---|---|
| **T1** | FE52h | 29h | CAPCOM Timer 1 Register | 0000h |
| **T1ICb** | FF9Eh | CFh | CAPCOM Timer 1 Interrupt Control Register | 0000h |
| **T1REL** | FE56h | 2Bh | CAPCOM Timer 1 Reload Register | 0000h |
| **T2** | FE40h | 20h | GPT1 Timer 2 Register | 0000h |
| **T2CONb** | FF40h | A0h | GPT1 Timer 2 Control Register | 0000h |
| **T2ICb** | FF60h | B0h | GPT1 Timer 2 Interrupt Control Register | 0000h |
| **T3** | FE42h | 21h | GPT1 Timer 3 Register | 0000h |
| **T3CONb** | FF42h | A1h | GPT1 Timer 3 Control Register | 0000h |
| **T3ICb** | FF62h | B1h | GPT1 Timer 3 Interrupt Control Register | 0000h |
| **T4** | FE44h | 22h | GPT1 Timer 4 Register | 0000h |
| **T4CONb** | FF44h | A2h | GPT1 Timer 4 Control Register | 0000h |
| **T4ICb** | FF64h | B2h | GPT1 Timer 4 Interrupt Control Register | 0000h |
| **T5** | FE46h | 23h | GPT2 Timer 5 Register | 0000h |
| **T5CONb** | FF46h | A3h | GPT2 Timer 5 Control Register | 0000h |
| **T5ICb** | FF66h | B3h | GPT2 Timer 5 Interrupt Control Register | 0000h |
| **T6** | FE48h | 24h | GPT2 Timer 6 Register | 0000h |
| **T6CONb** | FF48h | A4h | GPT2 Timer 6 Control Register | 0000h |
| **T6ICb** | FF68h | B4h | GPT2 Timer 6 Interrupt Control Register | 0000h |
| **T7** | F050h**E** | 28h | CAPCOM Timer 7 Register | 0000h |
| **T78CONb** | FF20h | 90h | CAPCOM Timer 7 and 8 Control Register | 0000h |
| **T7ICb** | F17Ah**E** | BEh | CAPCOM Timer 7 Interrupt Control Register | 0000h |
| **T7REL** | F054h**E** | 2Ah | CAPCOM Timer 7 Reload Register | 0000h |
| **T8** | F052h**E** | 29h | CAPCOM Timer 8 Register | 0000h |
| **T8ICb** | F17Ch**E** | BFh | CAPCOM Timer 8 Interrupt Control Register | 0000h |

**Table 13 Special function registers listed by name**

## 19 Special Function Register Overview

| Name | Physical address | 8-bit address | Description | Reset value |
|---|---|---|---|---|
| **T8REL** | F056h**E** | 2Bh | CAPCOM Timer 8 Reload Register | 0000h |
| **TFRb** | FFACh | D6h | Trap Flag Register | 0000h |
| **WDT** | FEAEh | 57h | Watchdog Timer Register (read only) | 0000h |
| **WDTCON** | FFAEh | D7h | Watchdog Timer Control Register | 000xh[2)] |
| **XP0ICb** | F186h**E** | C3h | CAN Module Interrupt Control Register | 0000h |
| **XP1ICb** | F18Eh**E** | C7h | X-Peripheral 1 Interrupt Control Register | 0000h |
| **XP2ICb** | F196h**E** | CBh | X-Peripheral 2 Interrupt Control Register | 0000h |
| **XP3ICb** | F19Eh$_{\mathbf{E}}$ | CFh | PLL unlock Interrupt Control Register | 0000h |
| **ZEROSb** | FF1Ch | 8Eh | Constant Value 0's Register (read only) | 0000h |

**Table 13 Special function registers listed by name**

The Interrupt Control Registers XPnIC control interrupt requests from integrated X-Bus peripherals. Nodes, where no X-Peripherals are connected, may be used to generate software controlled interrupt requests by setting the respective XPnIR bit.

1 The system configuration is selected during reset.

2 Bit WDTR indicates a watchdog timer triggered reset.

# 20 Electrical Characteristics

## 20.1 Absolute maximum ratings

- Ambient temperature under bias (TA): -40 to +125 ˚C
- Storage temperature (TST):– 65 to +150 ˚C
- Voltage on VDD pins with respect to ground (VSS): 0.5 to +6.5 V
- Voltage on any pin with respect to ground (VSS): –0.3 toVDD +0.3 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 ($V_{IN}$>$V_{DD}$ 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.*

## 20.2 Parameter interpretation

The parameters listed in the following tables represent the characteristics of the ST10C167 and its demands on the system.

Where the ST10C167 logic provides signals with their respective timing characteristics, the symbol “CC” for Controller Characteristics, is included in the “Symbol” column.

Where the external system must provide signals with their respective timing characteristics to the ST10C167, the symbol “SR” for System Requirement, is included in the “Symbol” column.

# 20.3 DC characteristics

$V_{DD}$ = 5 V 10%,$V_{SS}$ = 0 V, $f_{CPU}$ = 25 MHz, Reset active, $T_A$ = -40 to +125 ˚C

| Parameter | Symbol | Limit Values | | Unit | Test Condition |
|---|---|---|---|---|---|
| | | min. | max. | | |
| Input low voltage | $V_{IL}$**SR** | – 0.5 | 0.2 $V_{DD}$ – 0.1 | V | – |
| Input low voltage (special threshold) | $V_{ILS}$**SR** | – 0.5 | 2.0 | V | – |
| Input high voltage (all except $\overline{\text{RSTIN}}$ and XTAL1) | $V_{IH}$**SR** | 0.2 $V_{DD}$ + 0.9 | $V_{DD}$ + 0.5 | V | – |
| Input high voltage $\overline{\text{RSTIN}}$ | $V_{IH1}$**SR** | 0.6 $V_{DD}$ | $V_{DD}$ + 0.5 | V | – |
| Input high voltage XTAL1 | $V_{IH2}$**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 | – |
| 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 $V_{DD}$ 2.4 | – | V | $I_{OH}$ = – 500 A<br>$I_{OH}$ = –2.4 mA |
| Output high voltage [1] (all other outputs) | $V_{OH1}$**CC** | 0.9 $V_{DD}$ 2.4 | – | V<br>V | $I_{OH}$ = – 250 A<br>$I_{OH}$ = – 1.6 mA |
| Input leakage current (Port 5) | $I_{OZ1}$**CC** | – | 0.5 | A | 0 V < $V_{IN}$ < $V_{DD}$ |
| Input leakage current (all other) | $I_{OZ2}$**CC** | – | 1 | A | 0 V < $V_{IN}$ < $V_{DD}$ |

**Table 14 DC characteristics**

| Parameter | Symbol | Limit Values | | Unit | Test Condition |
|---|---|---|---|---|---|
| | | min. | max. | | |
| Overload current | $I_{OV}$**SR** | – | 5 | mA | 5) 8) |
| $\overline{\text{RSTIN}}$ pull-up resistor 5) | $R_{RST}$**CC** | 50 | 250 | KOhm | – |
| Read/Write inactive current 4) | $I_{RWH}$ 2) | – | -40 | A | $V_{OUT}$ = 2.4 V |
| Read/Write active current 4) | $I_{RWL}$ 3) | -500 | – | A | $V_{OUT}$ = $V_{OLmax}$ |
| ALE inactive current 4) | $I_{ALEL}$ 2) | 40 | – | A | $V_{OUT}$ = $V_{OLmax}$ |
| ALE active current 4) | $I_{ALEH}$ 3) | – | 500 | A | $V_{OUT}$ = 2.4 V |
| Port 6 inactive current 4) | $I_{P6H}$ 2) | – | -40 | A | $V_{OUT}$ = 2.4 V |
| Port 6 active current 4) | $I_{P6L}$ 3) | -500 | – | A | $V_{OUT}$ = $V_{OL1max}$ |
| PORT0 configuration current 4) | $I_{P0H}$ 2) | – | -10 | A | $V_{IN}$ = $V_{IHmin}$ |
| | $I_{P0L}$ 3) | -100 | – | A | $V_{IN}$ = $V_{ILmax}$ |
| XTAL1 input current | $I_{IL}$ **CC** | – | 20 | A | 0 V < $V_{IN}$ < $V_{DD}$ |
| Pin capacitance 5) (digital inputs/outputs) | $C_{IO}$**CC** | – | 10 | pF | f = 1 MHz<br>$T_A$ = 25 °C |
| Power supply current | $I_{CC}$ | – | 20 + 5 * $f_{CPU}$ | mA | $\overline{\text{RSTIN}}$ = $V_{IL2}$<br>$f_{CPU}$ in [MHz] 6) |
| Idle mode supply current | $I_{ID}$ | – | 20 + 2 * $f_{CPU}$ | mA | $\overline{\text{RSTIN}}$ = $V_{IH1}$<br>$f_{CPU}$ in [MHz] 6) |
| Power-down mode supply current | $I_{PD}$ | – | 100 | A | $V_{DD}$ = 5.5 V 7) |

**Table 14 DC characteristics**

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

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

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

4 This specification is only valid during Reset, or during 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 Not 100% tested, guaranteed by design characterization.

6 The supply current is a function of the operating frequency. This dependency is illustrated in the figure below.
These parameters are tested at $V_{DDmax}$ and 20 MHz CPU clock with all outputs disconnected and all inputs at $V_{IL}$ or $V_{IH}$.

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

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

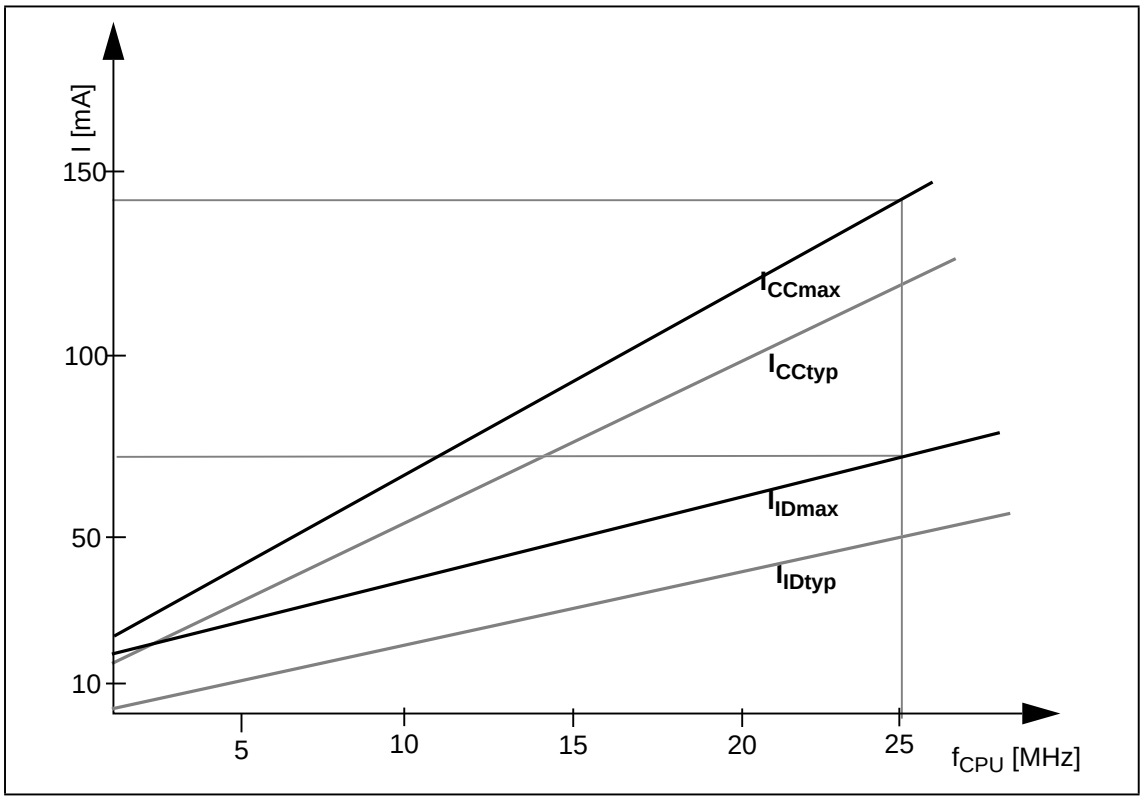



**Figure 8 Supply/idle current as a function of operating frequency**

## 20.3.1 A/D converter characteristics

$V_{DD}$ = 5 V ± 10%, $V_{SS}$ = 0 V, $T_A$ = -40 to +125 °C
4.0 V ≤ $V_{AREF}$ ≤ $V_{DD}$+0.1 V, $V_{SS}$-0.1 V≤ $V_{AGND}$ ≤ $V_{SS}$+0.2 V

| Parameter | Symbol | Limit Values | | Unit | Test Condition |
|---|---|---|---|---|---|
| | | min. | max. | | |
| Analog input voltage range | $V_{AIN}$**SR** | $V_{AGND}$ | $V_{AREF}$ | V | 1) |
| Sample time | $t_S$**CC** | – | 2 $t_{SC}$ | | 2) 4) |
| Conversion time | $t_C$**CC** | – | 14 $t_{CC}$ + $t_S$ + 4TCL | | 3) 4) |
| Total unadjusted error | TUE**CC** | – | 2 | LSB | 5) |
| Internal resistance of reference voltage source | $R_{AREF}$**SR** | – | $t_{CC}$ /165 - 0.25 | KOhm | $t_{CC}$ in [ns] 6) 7) |
| Internal resistance of analog source | $R_{ASRC}$**SR** | – | $t_S$ / 330 - 0.25 | KOhm | $t_S$ in [ns] 2) 7) |
| ADC input capacitance | $C_{AIN}$**CC** | – | 33 | pF | 7) |

**Table 15 A/D converter characteristics**

Sample time and conversion time of the ST10C167's ADC are programmable. The table below should be used to calculate the above timings.

| ADCON.15\|14 (ADCTC) | Conversion clock $t_{CC}$ | ADCON.13\|12 (ADSTC) | Sample clock $t_{SC}$ |
|---|---|---|---|
| 00 | TCL * 24 | 00 | $t_{CC}$ |
| 01 | Reserved, do not use | 01 | $t_{CC}$ * 2 |
| 10 | TCL * 96 | 10 | $t_{CC}$ * 4 |
| 11 | TCL * 48 | 11 | $t_{CC}$ * 8 |

1 $V_{AIN}$ may exceed $V_{AGND}$ or $V_{AREF}$ up to the absolute maximum ratings. However, the conversion result in these cases will be $X000_H$ or $X3FF_H$, respectively.

2 During the sample time the input capacitance $C_I$ can be charged/discharged by the external source. The internal resistance of the analog source must allow the capacitance to reach its final voltage level within $t_S$. After the end of the sample time $t_S$, changes of the analog input voltage have no effect on the conversion result.
Values for the sample clock $t_{SC}$ depend on programming and can be taken from the table above.

3 This parameter includes the sample time $t_S$, the time for determining the digital result and the time to load the result register with the conversion result.
Values for the conversion clock $t_{CC}$ depend on programming and can be taken from the table above.

4 This parameter is fixed by ADC control logic.

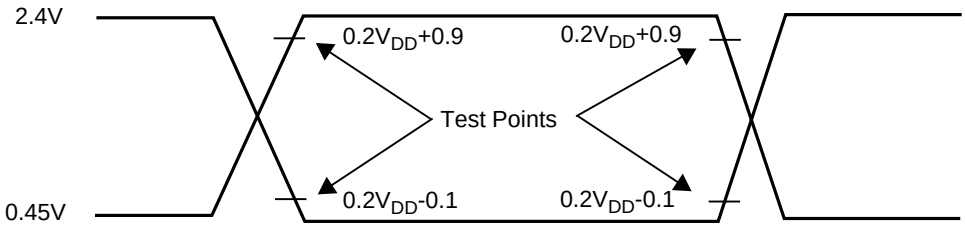
5 TUE is tested at $V_{AREF}$=5.0V, $V_{AGND}$=0V, $V_{CC}$=4.9V. It is guaranteed by design characterization for all other voltages within the defined voltage range.
The specified TUE is guaranteed only if an overload condition (see $I_{OV}$ specification) occurs on maximum 2 not selected analog input pins and the absolute sum of input overload currents on all analog input pins does not exceed 10 mA.
During the reset calibration sequence the maximum TUE may be 4 LSB

6 During the conversion the ADC's capacitance must be repeatedly charged or discharged. The internal resistance of the reference voltage source must allow the capacitance to reach its respective voltage level within $t_{CC}$. The maximum internal resistance results from the programmed conversion timing.

7 Not 100% tested, guaranteed by design characterization.

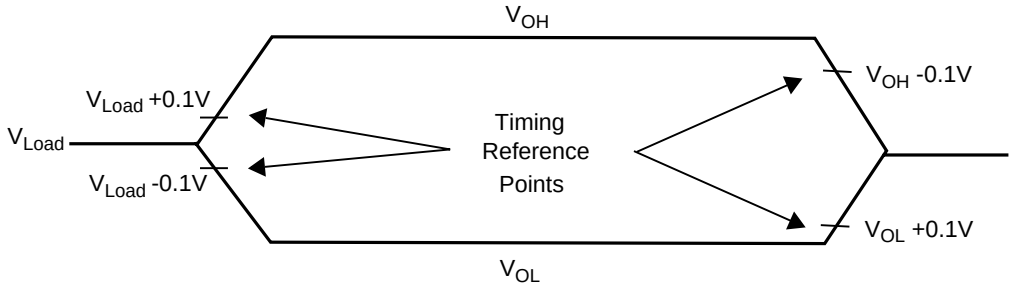
# 20.4 AC characteristics

## Test 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 VIH min for a logic '1' and VIL max for a logic '0'.

**Figure 9 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 VOH/VOL level occurs (IOH/IOL = 20 mA).

**Figure 10 Float waveforms**

## 20.4.1 Definition of internal timing

The internal operation of the ST10C167 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).

The CPU clock signal can be generated by 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 ST10C167.

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

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

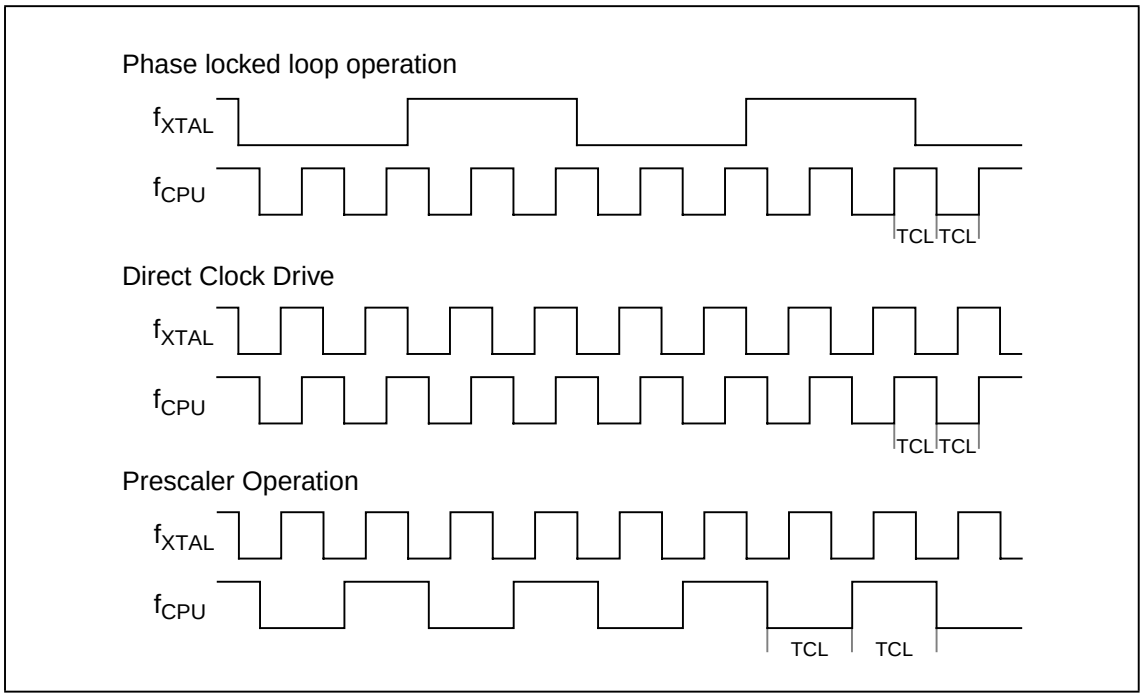



**Figure 11 Generation mechanisms for the CPU clock**

## 20.4.2 Clock generation modes

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

| 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.

## 20.4.3 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.

Note that if the bit OWDDIS in SYSCON register is cleared, the PLL is running on its free-running frequency and delivers the clock signal for the Oscillator Watchdog. If bit OWDDIS is set, then the PLL is switched off.

## 20.4.4 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 by the following formula:

$$TCL_{min} = 1/f_{XTAL} * DC_{min}$$

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 $TCL_{min}$.*

Note that if the bit OWDDIS in SYSCON register is cleared, the PLL is running on its free-running frequency and delivers the clock signal for the Oscillator Watchdog. If bit OWDDIS is set, then the PLL is switched off.

## 20.4.5 Oscillator watchdog (OWD)

When the clock option selected is direct drive or direct drive with prescaler, in order to provide a fail safe mechanism in case of a loss of the external clock, an oscillator watchdog is implemented as an additional functionality of the PLL circuitry. This oscillator watchdog operates as follows:

After a reset, the Oscillator Watchdog is enabled by default. To disable the OWD, the bit OWDDIS (bit 4 of SYSCON register) must be set.

When the OWD is enabled, the PLL is running on its free-running frequency, and increment the Oscillator Watchdog counter. On each transition of XTAL1 pin, the Oscillator Watchdog is cleared. If an external clock failure occurs, then the Oscillator Watchdog counter overflows (after 16 PLL clock cycles). The CPU clock signal will be switched to the PLL free-running clock signal, and the Oscillator Watchdog Interrupt Request (XP3INT) is flagged. The CPU

clock will not switch back to the external clock even if a valid external clock exits on XTAL1 pin. Only a hardware reset can switch the CPU clock source back to direct clock input.

When the OWD is disabled, the CPU clock is always fed from the oscillator input and the PLL is switched off to decrease power supply current.

## 20.4.6 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. $f_{CPU}$ = $f_{XTAL}$ * F). With every F'th transition of $f_{XTAL}$ the PLL circuit synchronizes the CPU clock to the input clock. This synchronization is done smoothly, 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 *N* * TCL the minimum value is computed using the corresponding deviation $D_N$:

$$\mathrm{TCL}_{min} = \mathrm{TCL}_{NOM} * (1 - |\mathrm{D}_N| / 100)$$

$$\mathrm{D}_N = \pm(4 - N/15)[\%]$$

where ***N*** = number of consecutive TCLs and $1 \leq N \leq 40$. So for a period of 3 TCLs (i.e. N = 3):

$$\begin{aligned} \mathrm{D}_3 &= 4 - 3/15 \\ &= 3.8\% \\ 3\mathrm{TCL}_{min} &= 3\mathrm{TCL}_{NOM} \times (1 - 3.8/100) \\ &= \mathrm{TCL}_{NOM} \times 0.962 \\ &(57.72\,\mathrm{nsec}@\mathrm{f}_{CPU} = 25\,\mathrm{MHz}) \end{aligned}$$

This is especially important for bus cycles using waitstates and e.g. for the operation of timers, serial interfaces, etc. For all slower operations and longer periods (e.g. pulse train

generation or measurement, lower baudrates, etc.) the deviation caused by the PLL jitter is negligible.

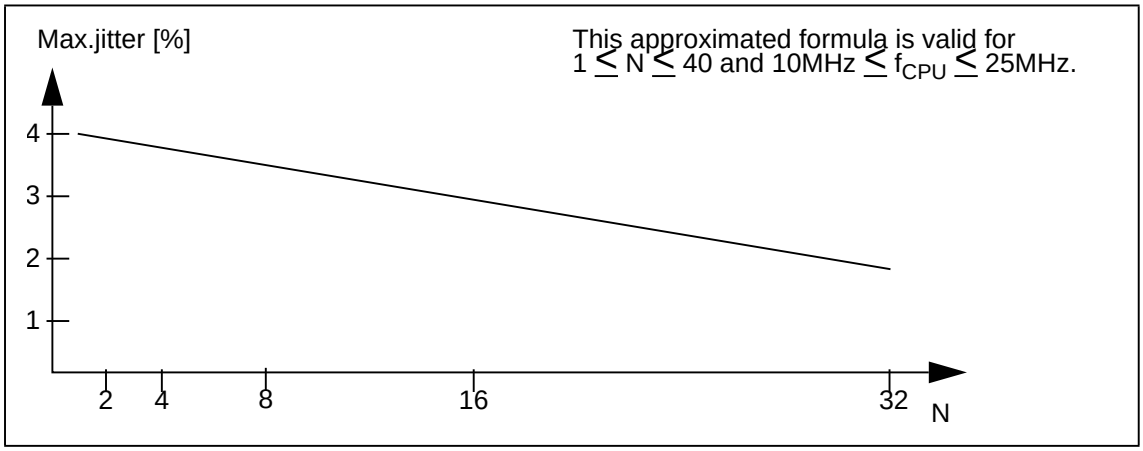



**Figure 12 Approximated maximum PLL jitter**

## 20.4.7 Memory cycle variables

The 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>) |

## 20.4.8 External clock drive XTAL1

$V_{DD}$ = 5 V 10%,$V_{SS}$ = 0 V, $T_A$ = -40 to +125 ˚C

| Parameter | Symbol | $f_{CPU} = f_{XTAL}$ | | $f_{CPU} = f_{XTAL} / 2$ | | $f_{CPU} = f_{XTAL} * N$ N = 1.5/2,/2.5/3/4/5 | | Unit |
|---|---|---|---|---|---|---|---|---|
| | | min | max | min | max | min | max | |
| Oscillator period | $t_{OSC}$**SR** | 40[1)] | 1000 | 20[2)] | 500 | 40 * N | 100 * N | ns |
| High time | $t_1$**SR** | 18[3)] | – | 6[3)] | – | 10[3)] | – | ns |
| Low time | $t_2$**SR** | 18[3)] | – | 6[3)] | – | 10[3)] | – | ns |
| Rise time | $t_3$**SR** | – | 10[3)] | – | 6[3)] | – | 10[3)] | ns |
| Fall time | $t_4$**SR** | – | 10[3)] | – | 6[3)] | – | 10[3)] | ns |

1 Theoretical minimum. The real minimum value depends on the duty cycle of the input clock signal.

2 25 MHz is the maximum input frequency when using an external crystal oscillator; however, 50 MHz can be applied with an external clock source.

3 The input clock signal must reach the defined levels VIL and VIH2.

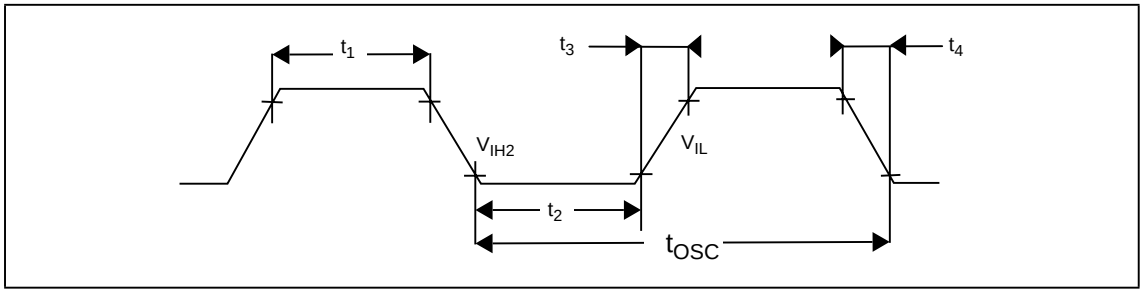



**Figure 13 External clock drive XTAL1**

## 20.4.9 Multiplexed bus

$V_{DD}$ = 5 V 10%,$V_{SS}$ = 0 V, $T_A$ = -40 to +125 °C
$C_L$ (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 25MHz | | 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 |

**Table 16 Multiplexed bus characteristics**

| Parameter | Symbol | Max. CPU Clock = 25 MHz | | Variable CPU Clock 1/2TCL = 1 to 25MHz | | Unit |
|---|---|---|---|---|---|---|
| | | min. | max. | min. | max. | |
| $\overline{RD}$ 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/Unlatched $\overline{CS}$ to valid data in | $t_{17}$SR | – | 50 + 2$t_A$ + $t_C$ | – | 4TCL - 30 + 2$t_A$ + $t_C$ | 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 |
| 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/Unlatched $\overline{CS}$ hold after $\overline{RD}$, $\overline{WR}$ | $t_{27}$CC | 26 + $t_F$ | – | 2TCL - 14 + $t_F$ | – | ns |
| ALE falling edge to Latched $\overline{CS}$ | $t_{38}$CC | -4 - $t_A$ | 10 - $t_A$ | -4 - $t_A$ | 10 - $t_A$ | ns |
| Latched $\overline{CS}$ low to Valid Data In | $t_{39}$SR | – | 40 + $t_C$ + 2$t_A$ | – | 3TCL - 20 + $t_C$ + 2$t_A$ | ns |
| Latched $\overline{CS}$ 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 |

**Table 16 Multiplexed bus characteristics**

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

**Table 16 Multiplexed bus characteristics**

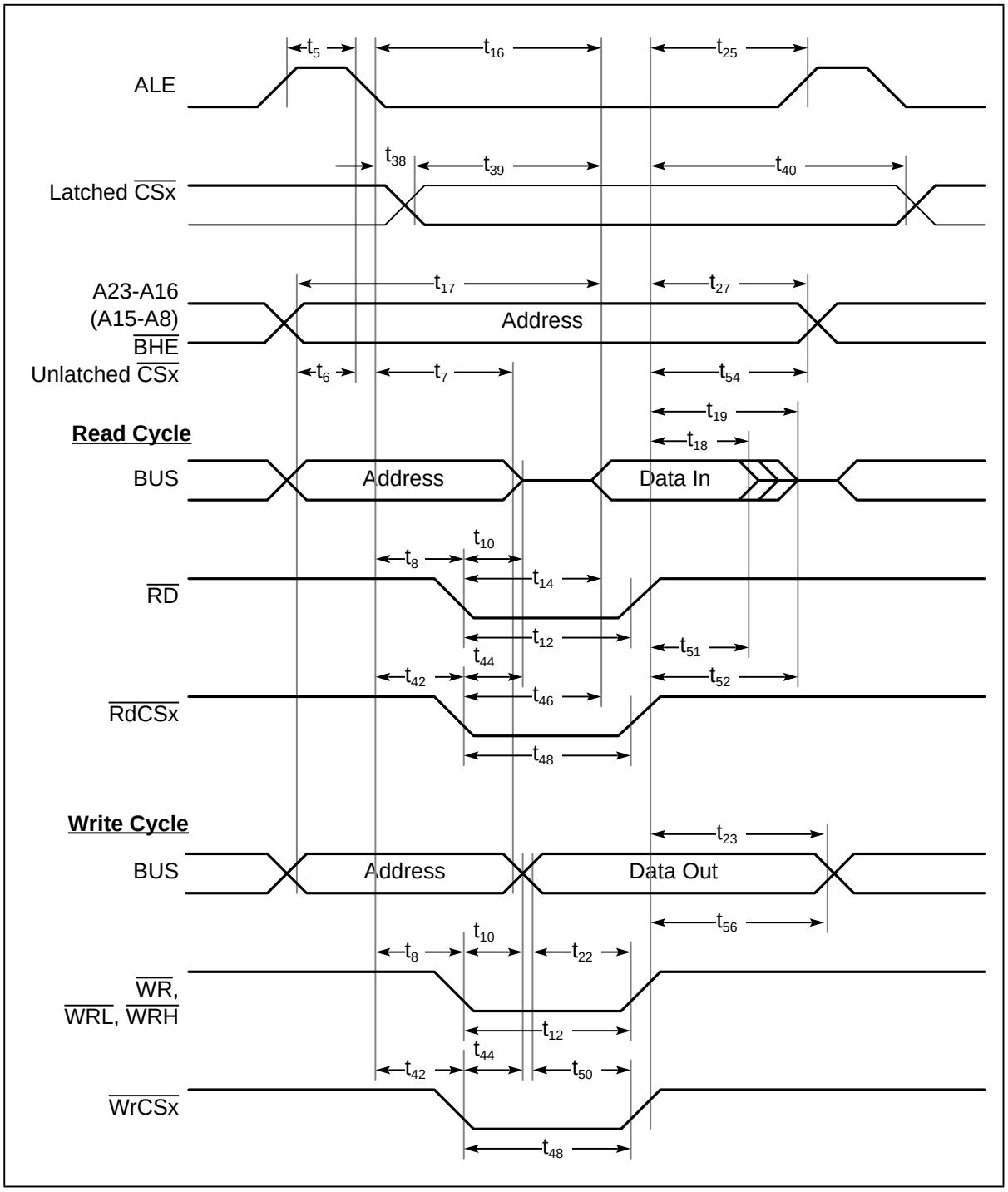



**Figure 14 External memory cycle:multiplexed bus, with read/write delay, normal ALE**

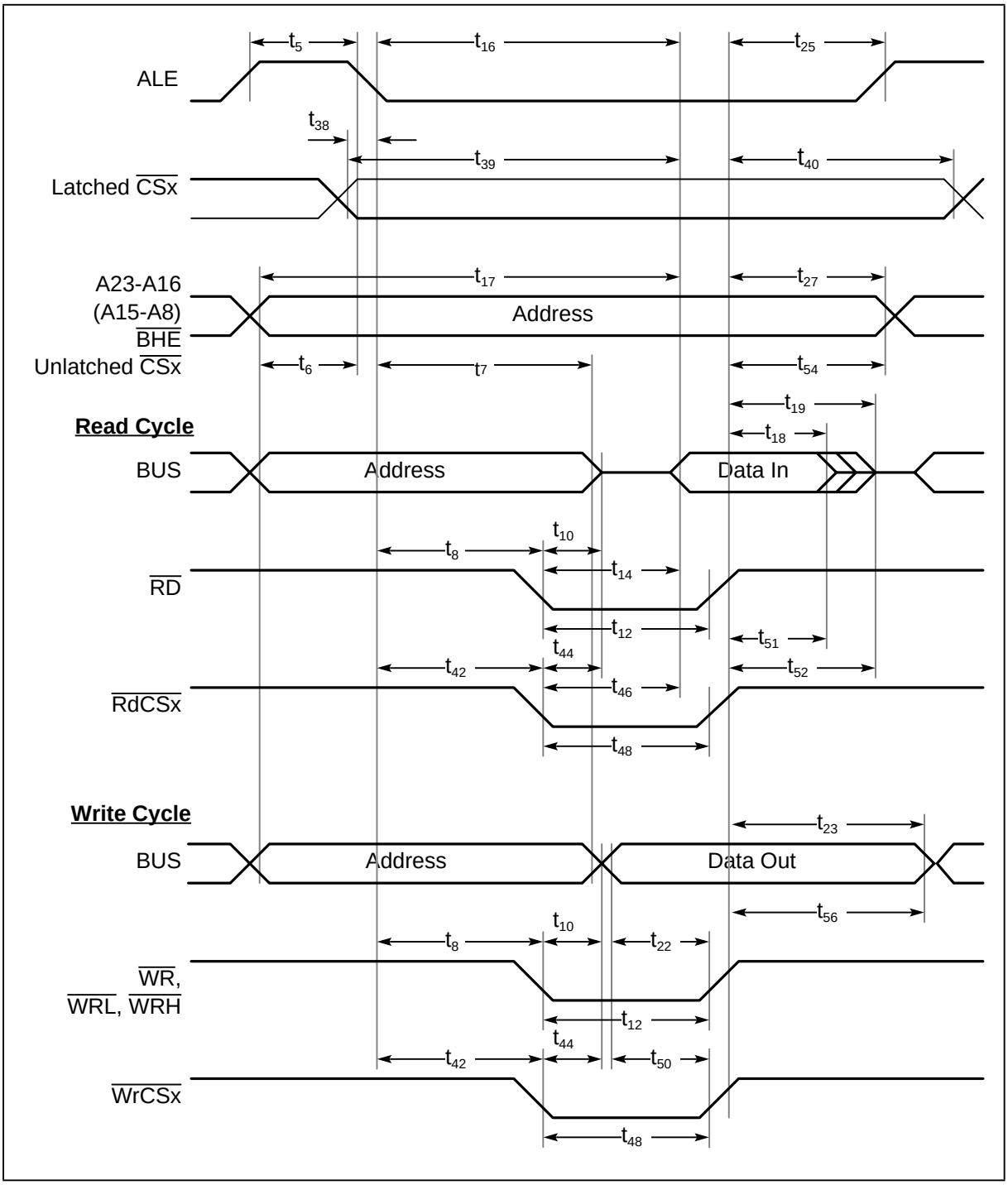



**Figure 15 External memory cycle:multiplexed bus, with read/write delay, ext'd ALE**

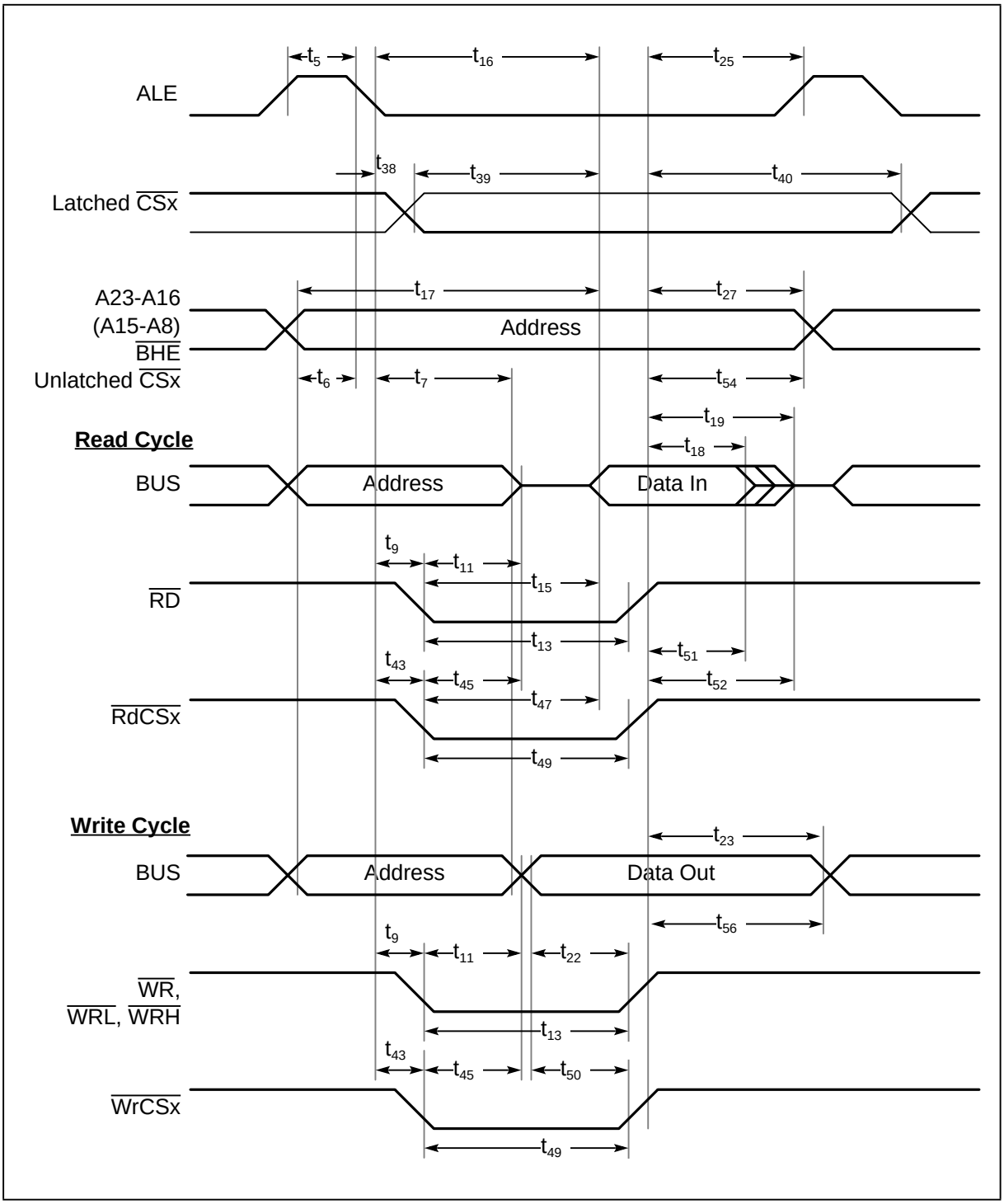



**Figure 16 External memory cycle:multiplexed bus, no read/write delay, normal ALE**

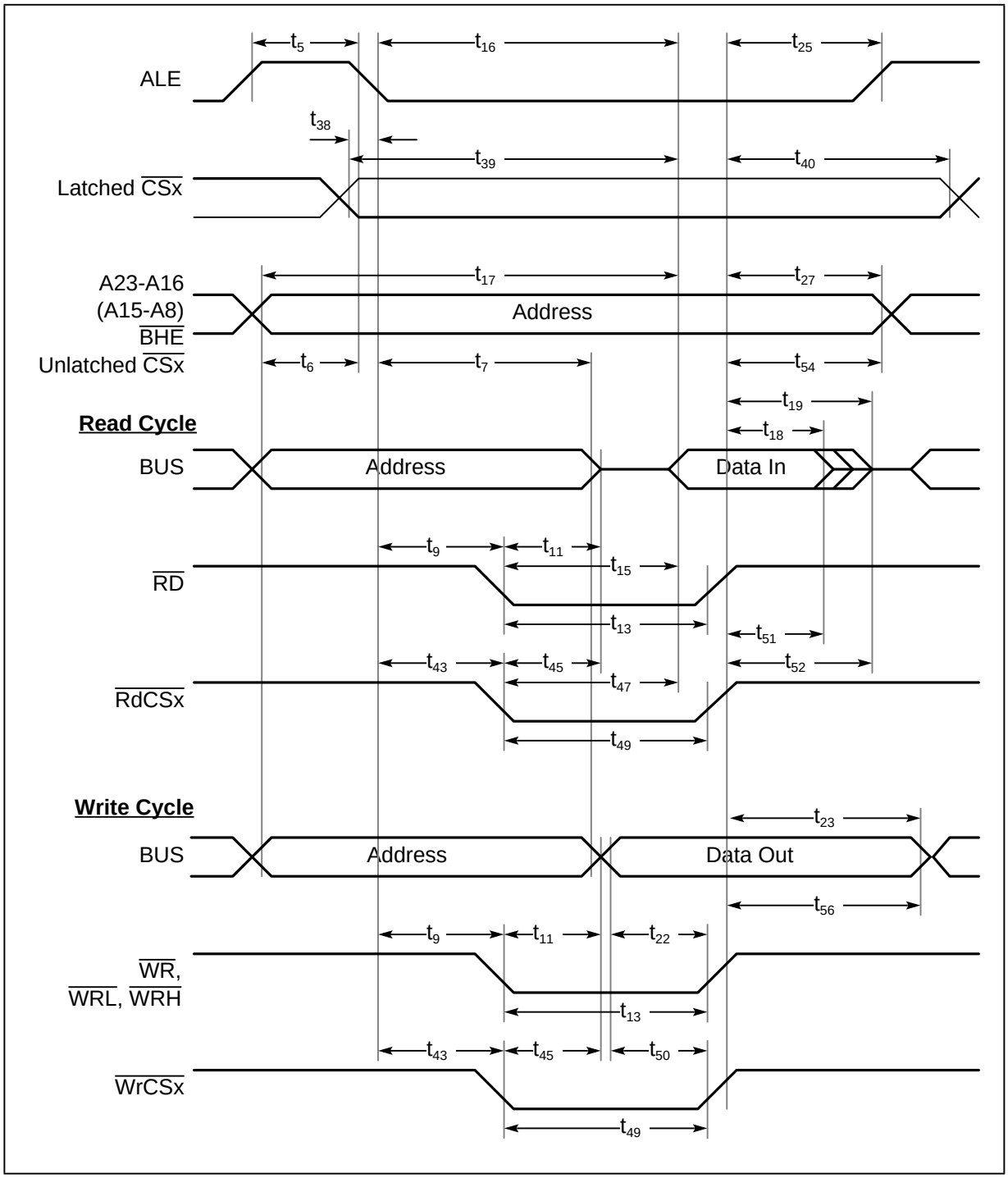



**Figure 17 External memory cycle:multiplexed bus, no read/write delay, extended ALE**

## 20.4.10 Demultiplexed bus

$V_{DD}$ = 5 V 10%,$V_{SS}$ = 0 V, $T_A$ = -40 to +125 °C

$C_L$ (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 = 25MHz | | Variable CPU Clock 1/2TCL = 1 to 25MHz | | 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/Unlatched $\overline{CS}$ 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 |

**Table 17 Demultiplexed bus characteristics**

| Parameter | Symbol | Max. CPU Clock = 25MHz | | Variable CPU Clock 1/2TCL = 1 to 25MHz | | Unit |
|---|---|---|---|---|---|---|
| | | min. | max. | min. | max. | |
| Data float after $\overline{RD}$ rising edge (with RW-delay[1)]) | $t_{20}$SR | – | 26 + $t_F$ | – | 2TCL - 14 + $t_F$ + $2t_A$[1)] | ns |
| Data float after $\overline{RD}$ rising edge (no RW-delay[1)]) | $t_{21}$SR | – | 10 + $t_F$ | – | TCL - 10 + $t_F$ + $2t_A$[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 |
| ALE rising edge after $\overline{RD}$, $\overline{WR}$ | $t_{26}$CC | -10 + $t_F$ | – | -10 + $t_F$ | – | ns |
| Address/Unlatched $\overline{CS}$ hold after $\overline{RD}$, $\overline{WR}$[2)] | $t_{28}$CC | 0 + $t_F$ | – | 0 + $t_F$ | – | ns |
| ALE falling edge to Latched $\overline{CS}$ | $t_{38}$CC | -4 - $t_A$ | 10 - $t_A$ | -4 - $t_A$ | 10 - $t_A$ | ns |
| Latched $\overline{CS}$ low to Valid Data In | $t_{39}$SR | – | 40 + $t_C$+ $2t_A$ | – | 3TCL - 20 + $t_C$ + $2t_A$ | ns |
| Latched $\overline{CS}$ 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 |

**Table 17 Demultiplexed bus characteristics**

| Parameter | Symbol | Max. CPU Clock = 25MHz | | Variable CPU Clock 1/2TCL = 1 to 25MHz | | Unit |
|---|---|---|---|---|---|---|
| | | min. | max. | min. | max. | |
| $\overline{RdCS}$, $\overline{WrCS}$ 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 | -10 + $t_F$ | – | -10 + $t_F$ | – | ns |
| Data hold after $\overline{WrCS}$ | $t_{57}$CC | 6 + $t_F$ | – | TCL - 14 + $t_F$ | – | ns |

**Table 17 Demultiplexed bus characteristics**

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

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

.

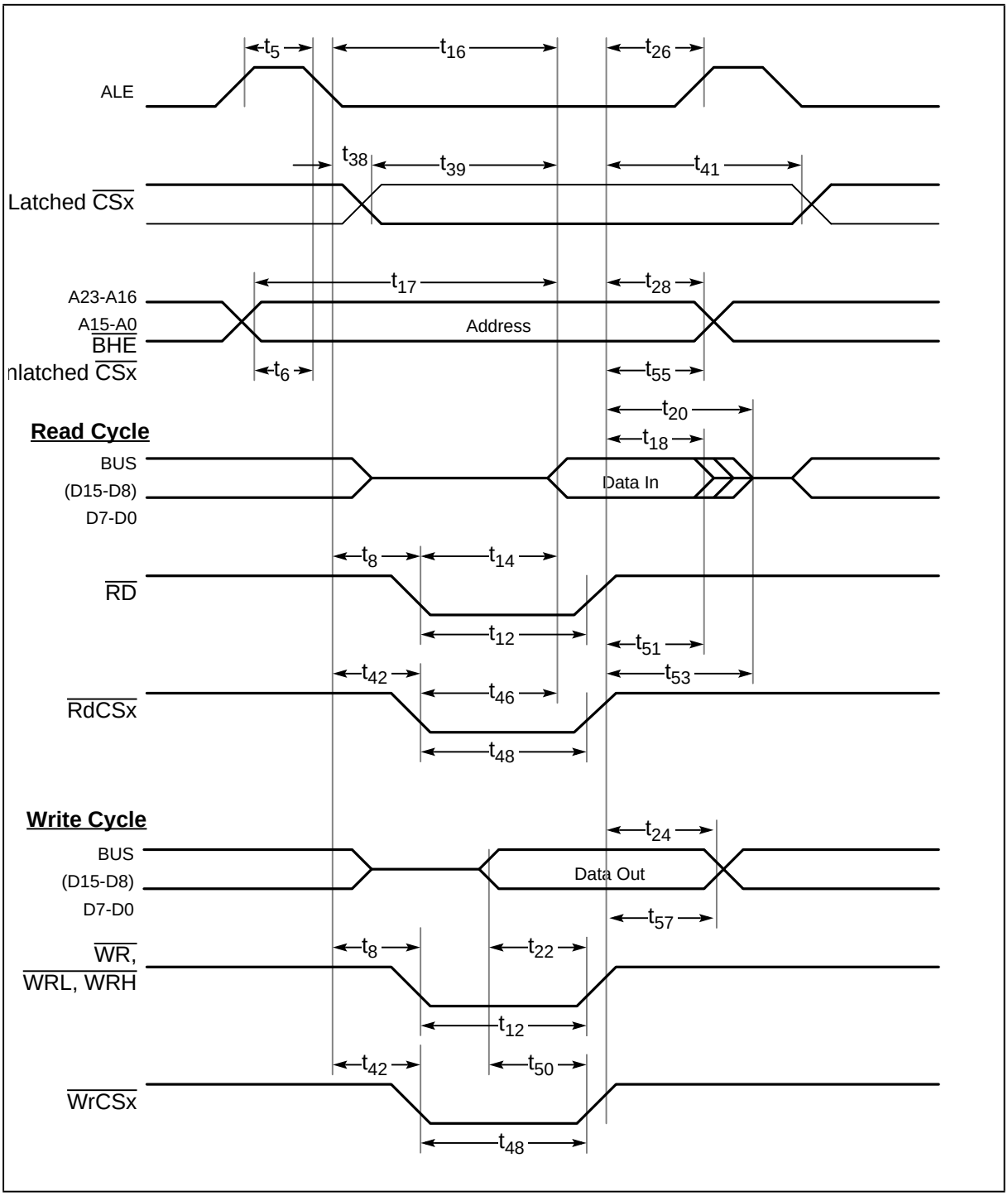



**Figure 18 External memory cycle: demultip bus, with read/write delay, normal ALE**

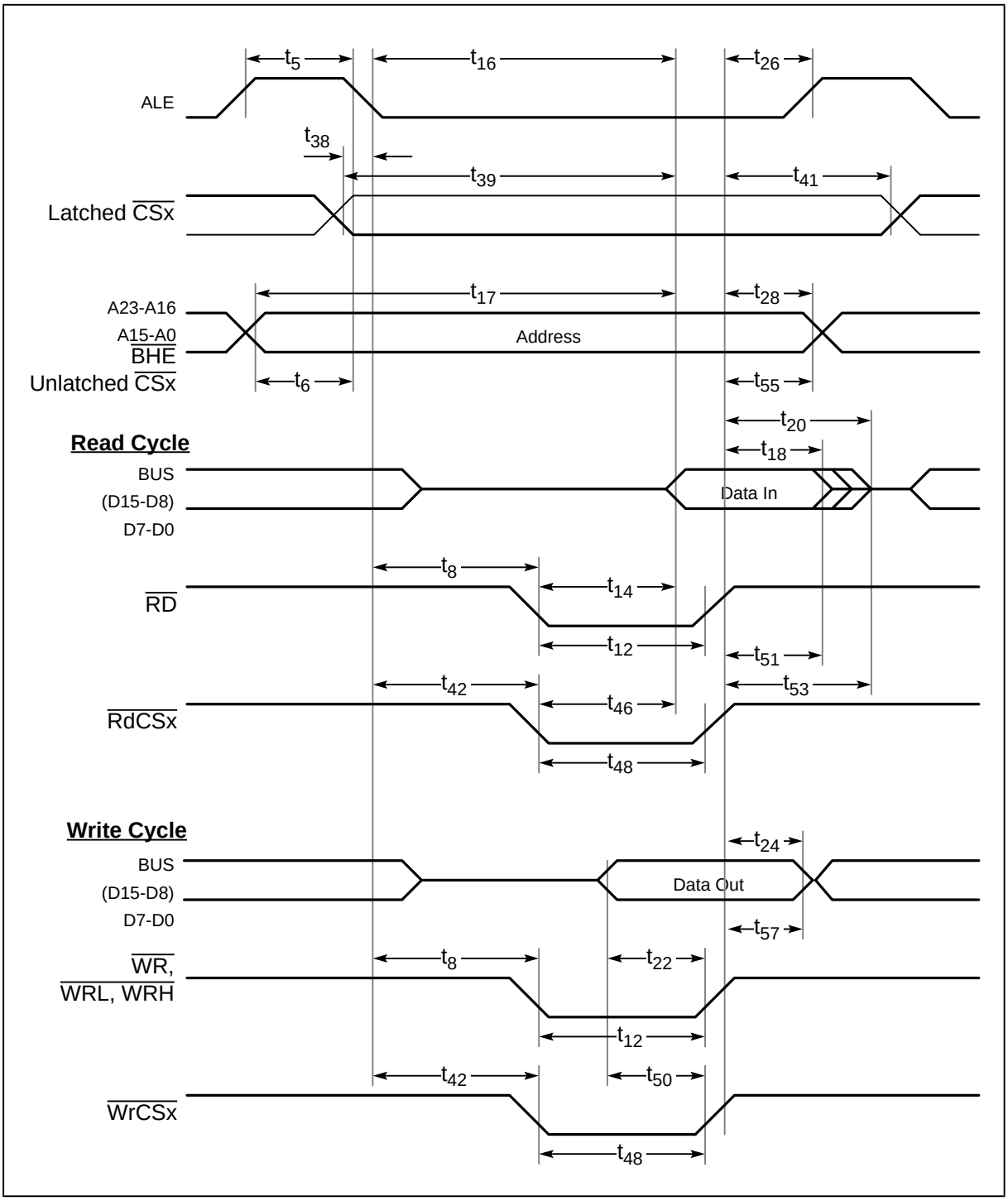



**Figure 19 External memory cycle:demultiplexed bus, with read/write delay, ext'd ALE**

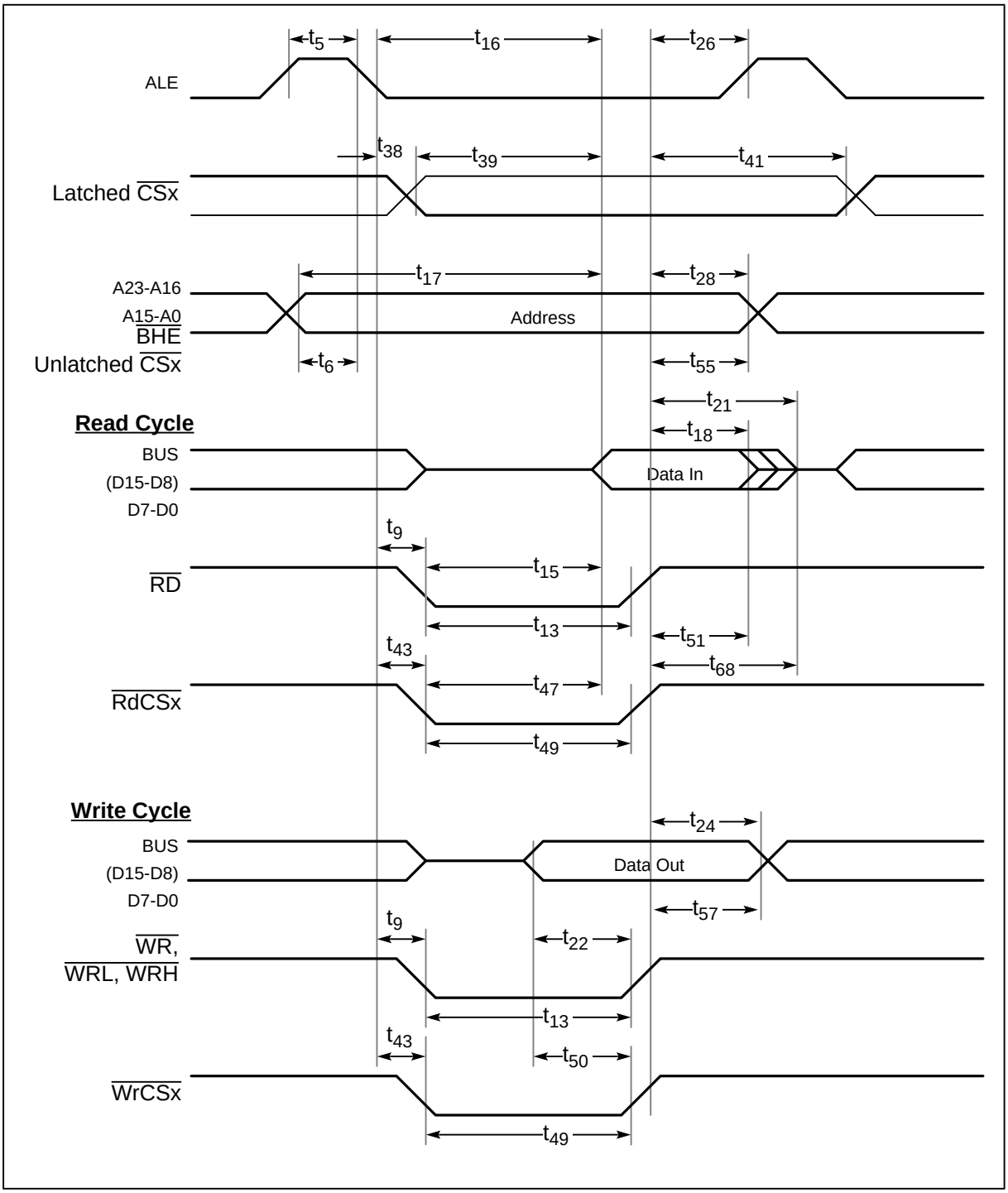



**Figure 20 External memory cycle: demultip bus, no read/write delay, normal ALE**

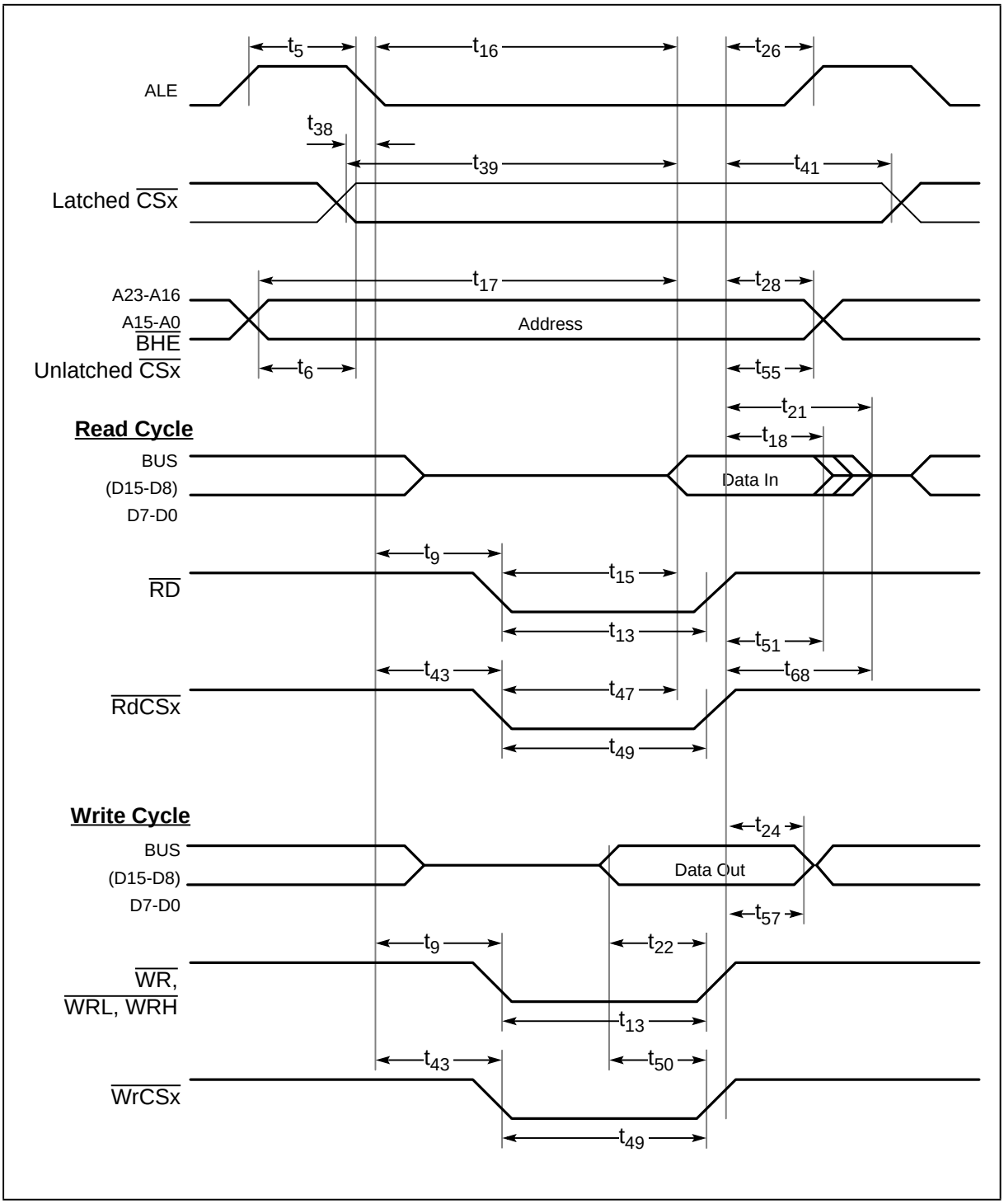



**Figure 21 External memory cycle:demultiplexed bus, no read/write delay, ext'd ALE**

## 20.4.11 CLKOUT and $\overline{\text{READY}}$

$V_{DD}$ = 5 V 10%,$V_{SS}$ = 0 V, $T_A$ = -40 to +125 °C
$C_L$ (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 = 25MHz | | Variable CPU Clock 1/2TCL = 1 to 25MHz | | 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 $\overline{RD}$, $\overline{WR}$ high (Demultiplexed Bus) [2] | $t_{60}$SR | 0 | 0 + 2$t_A$ + $t_C$ + $t_F$ [2] | 0 | TCL - 20 + 2$t_A$ + $t_C$ + $t_F$ [2] | ns |

**Table 18 CLKOUT and $\overline{\text{READY}}$ characteristics**

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 READY.
The $2t_A$ and $t_C$ refer to the next following bus cycle, $t_F$ refers to the current bus cycle.

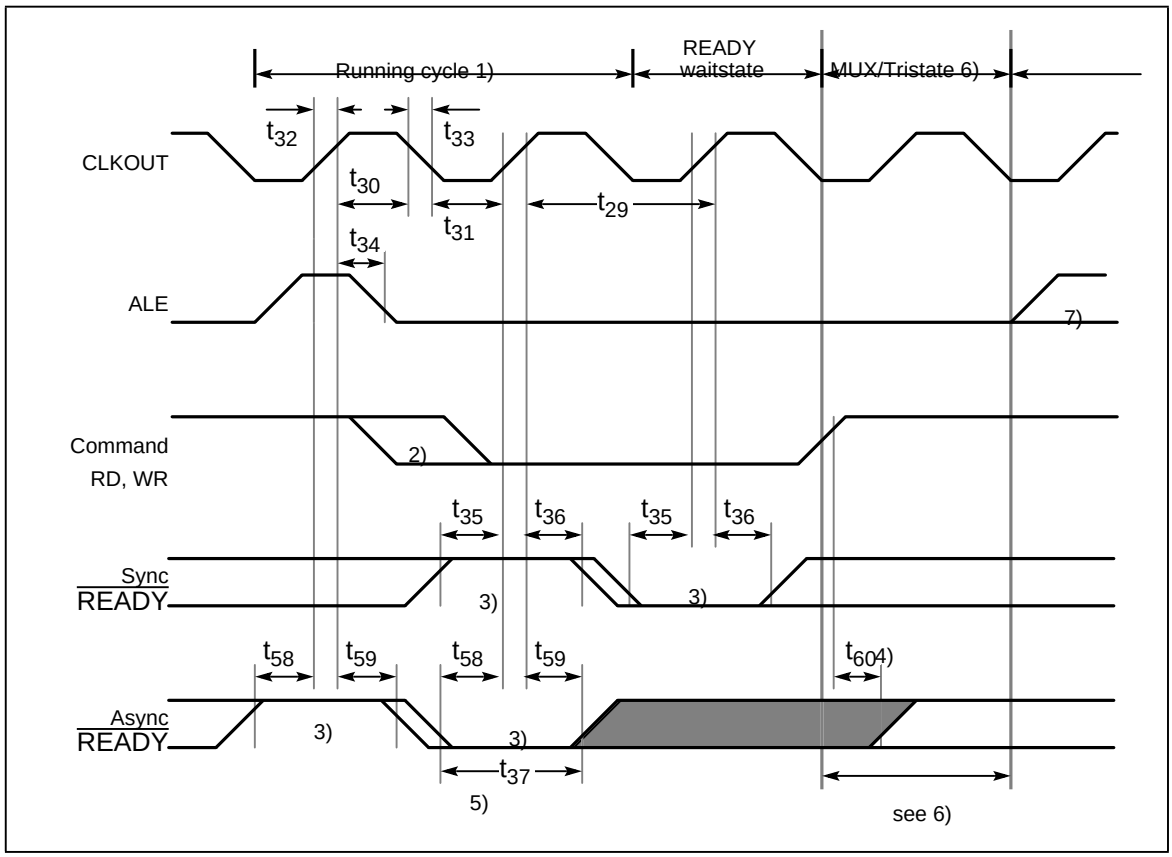



**Figure 22 CLKOUT and $\overline{\text{READY}}$**

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 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.

## 20.4.12 External bus arbitration

$V_{DD}$ = 5 V 10%,$V_{SS}$ = 0 V, $T_A$ = -40 to +125 °C
$C_L$ (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 = 25MHz | | Variable CPU Clock 1/2TCL = 1 to 25MHz | | Unit |
|---|---|---|---|---|---|---|
| | | min. | max. | min. | max. | |
| $\overline{HOLD}$ input setup time to CLKOUT | $t_{61SR}$ | 20 | – | 20 | – | ns |
| CLKOUT to $\overline{HLDA}$ high or $\overline{BREQ}$ low delay | $t_{62CC}$ | – | 20 | – | 20 | ns |
| CLKOUT to $\overline{HLDA}$ low or $\overline{BREQ}$ high delay | $t_{63CC}$ | – | 20 | – | 20 | ns |
| $\overline{CSx}$ release | $t_{64CC}$ | – | 20 | – | 20 | ns |
| $\overline{CSx}$ drive | $t_{65CC}$ | -4 | 24 | -4 | 24 | ns |
| Other signals release | $t_{66CC}$ | – | 20 | – | 20 | ns |
| Other signals drive | $t_{67CC}$ | -4 | 24 | -4 | 24 | ns |

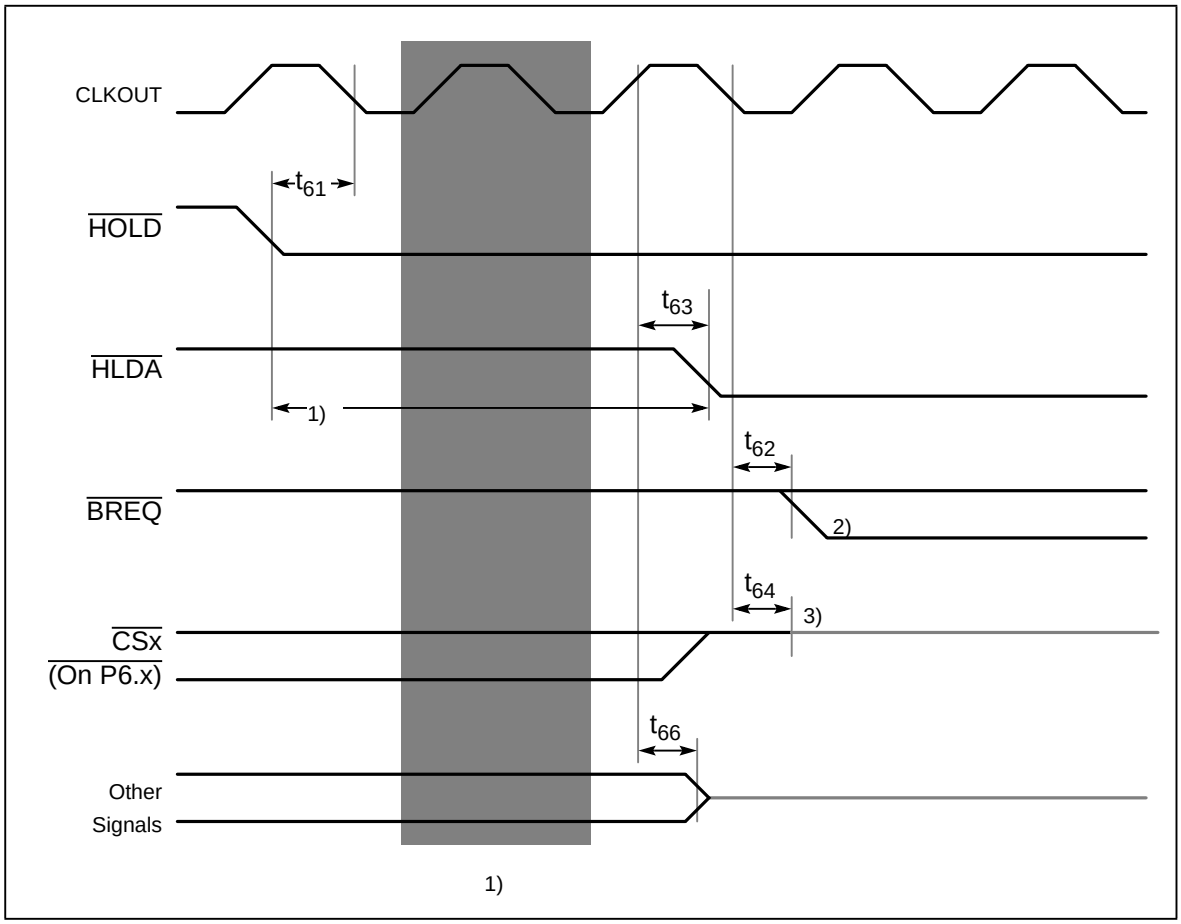



**Figure 23 External bus arbitration, releasing the bus**

1 The ST10C167 will complete the currently running bus cycle before granting bus access.

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

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

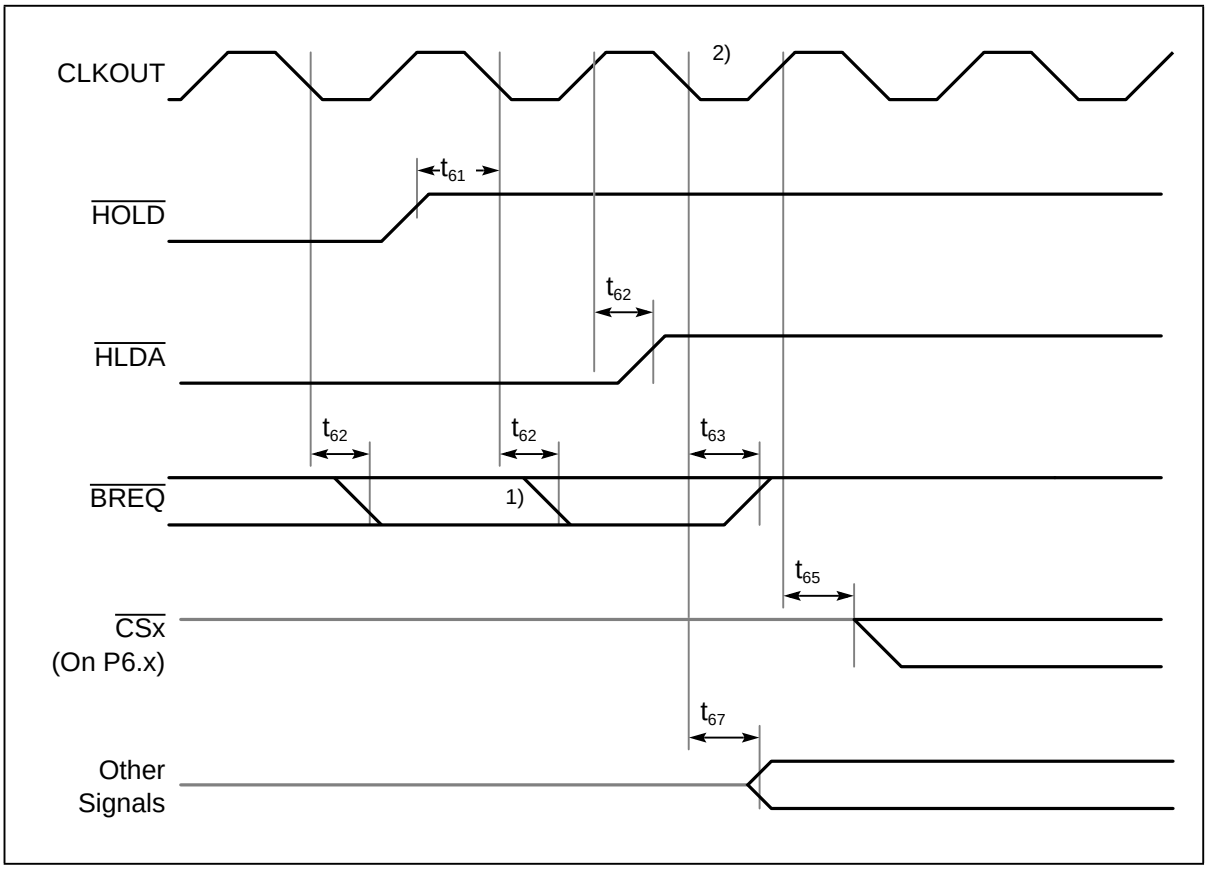


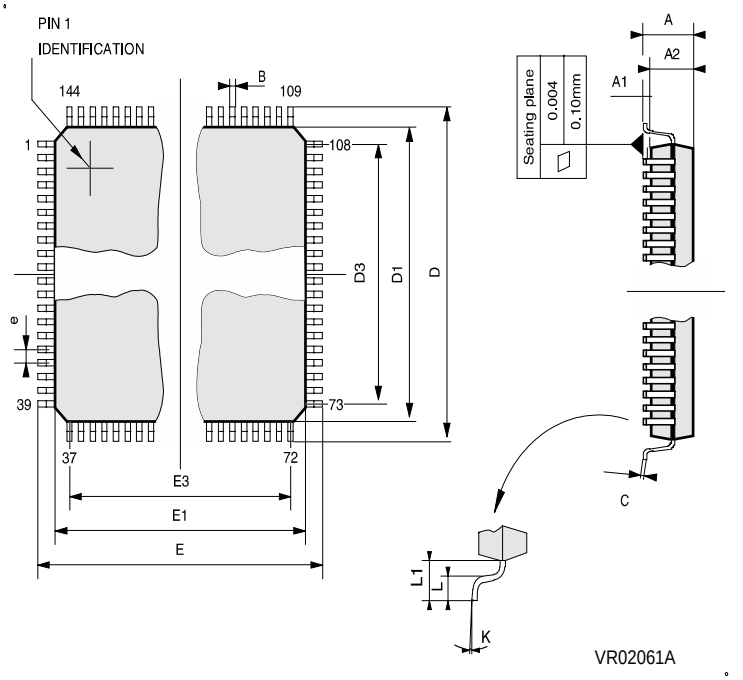

**Figure 24 External bus arbitration, (regaining the bus)**

1 This is the last chance for BREQ to trigger the indicated regain-sequence. Even if BREQ is activated earlier, the regain-sequence is initiated by HOLD going high. Please note that HOLD may also be disactivated without the ST10C167 requesting the bus.

2 The next ST10C167 driven bus cycle may start here.

# 21 Package Mechanical Data

**Figure 25 Package Outline PQFP144 (28 x 28 mm)**




| Dim | mm | | | inches | | |
|---|---|---|---|---|---|---|
| | min | ty | max | min | ty | max |
| A | | | 4.07 | | | 0.106 |
| A1 | 0.25 | | | 0.010 | | |
| A2 | 3.17 | 3.42 | 3.67 | 0.125 | 0.315 | 0.144 |
| B | 0.22 | | 0.38 | 0.009 | | 0.015 |
| C | 0.13 | | 0.23 | 0.005 | | 0.009 |
| D | 30.95 | 31.20 | 31.45 | 1.129 | 1.228 | 1.238 |
| D1 | 27.90 | 28.00 | 28.10 | 1.098 | 1.120 | 1.106 |
| D3 | | 22.75 | | | 0.896 | |
| e | | 0.65 | | | 0.02- | |
| E | 30.95 | 31.20 | 31.45 | 1.219 | 1.228 | 1.238 |
| E1 | 27.90 | 28.00 | 28.10 | 1.098 | 1.102 | 1.106 |
| E3 | | 22.75 | | | 0.896 | |
| L | 0.65 | 0.80 | 0.95 | 0.026 | 0.031 | 0.037 |
| L1 | | 1.60 | | | 0.063 | |
| K | 0°(min), 7°(max) | | | | | |
| Number of Pins | | | | | | |
| N1 | 144 | | | | | |

# 22 Ordering Information

| Salestype | Temperature range | Package |
|---|---|---|
| ST10C167-Q3 | -40°C to 125°C | PQFP144 (28 x 28 mm) |