

MC68307 MC68307V

Technical Summary Integrated Multiple-Bus Processor

The MC68307 is an integrated processor combining a static 68EC000 processor with multiple interchip bus interfaces. The MC68307 is designed to provide optimal integration and performance for applications such as digital cordless telephones, portable measuring equipment, and point-of-sale terminals. By providing 3.3 V, static operation in a small package, the MC68307 delivers cost-effective performance to handheld, battery-powered applications.

The MC68307 (shown in Figure 1) contains a static EC000 core processor, multiple bus interfaces, a serial channel, two timers, and common system glue logic. The multiple bus interfaces include: dynamic 68000 bus, 8051 bus, and Motorola bus (M-bus) or I²C bus¹. The dynamically sized 68000 bus allows 16-bit performance out of static random access memory (SRAM) while still providing a low-cost interface to an 8-bit read-only memory (ROM). The 8051 bus interfaces gluelessly to 8051-type devices and allows the reuse of application-specific integrated circuits (ASICs) designed for this industry standard bus. The M-bus is an industry standard 2-wire interface which provides efficient communications with peripherals such as EEPROM, analog/digital (A/D) converters, and liquid crystal display (LCD) drivers. Thus, the MC68307 interfaces gluelessly to boot ROM, SRAM, 8051 devices, M-bus devices, and memory-mapped peripherals. The MC68307 also incorporates a slave mode which allows the EC000 core to be turned off, providing a 3.3-V static, low-power multi-function peripheral for higher performance M68000 family processors.

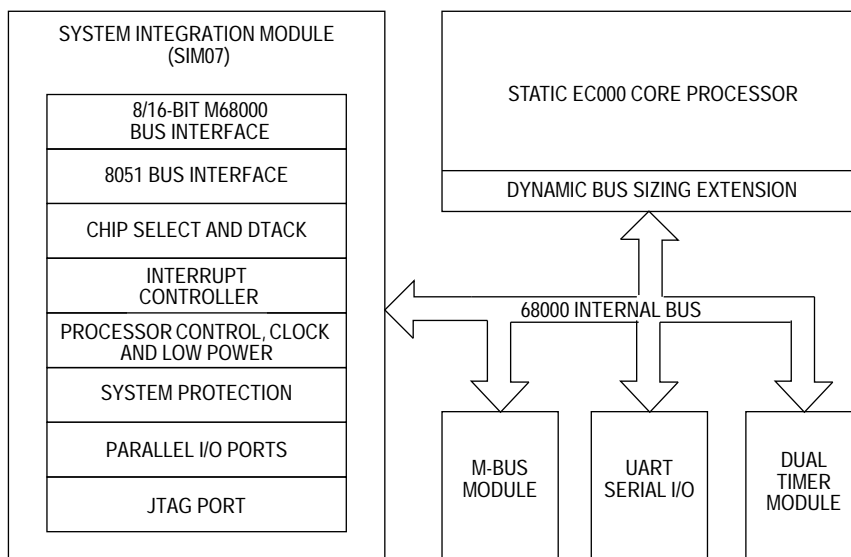


Figure 1. MC68307 Block Diagram

¹. I²C bus is a proprietary Philips interface bus.

This document contains information on a product under development. Motorola reserves the right to change or discontinue this product without notice.



The main features of the MC68307 include:

- Static EC000 Core Processor—Identical to MC68EC000 Microprocessor
 - Full compatibility with MC68000 and MC68EC000
 - 24-bit address bus, for 16-Mbyte off-chip address space
 - 16-bit on-chip data bus for MC68000 bus operations
 - Static design allows processor clock to be stopped providing dramatic power savings
 - 2.4 MIPS performance at 16.67-MHz processor clock
- External M68000 Bus Interface with Dynamic Bus Sizing for 8-bit and 16-bit Data Ports
- External 8-Bit Data Bus Interface (8051-Compatible)
- M-Bus Module
 - Provides interchip bus interface for EEPROMs, LCD controllers, A/D converters, etc.
 - Compatible with industry-standard I²C bus
 - Master or slave operation modes, supports multiple masters
 - Automatic interrupt generation with programmable level
 - Software-programmable clock frequency
 - Data rates from 4–100 Kbit/s above 3.0-MHz system clock
- Universal Asynchronous Receiver/Transmitter (UART) Module
 - Flexible baud rate generator
 - Based on MC68681 Dual Universal Asynchronous Receiver/Transmitter (DUART) programming model
 - 5 Mbits/s maximum transfer rate at 16.67-MHz system clock
 - Automatic interrupt generation with programmable level
 - Modem control signals available (CTS,RTS)
- Timer Module
 - Dual channel 16-bit general purpose counter/timer
 - Multimode operation, independent capture/compare registers
 - Automatic interrupt generation with programmable level
 - Third 16-bit timer configured as a software watchdog
 - 60-ns resolution at 16.67-MHz system clock
 - Each timer has an input and an output pin
- System Integration Module (SIM07), Incorporating Many Functions Typically Relegated to External Programmable Array Logic (PALs), Transistor-Transistor Logic (TTL), and ASICs, such as:
 - System configuration, programmable address mapping
 - System protection by hardware watchdog logic
 - Power-down mode control, programmable processor clock driver
 - Four programmable chip selects with wait state generation logic
 - Three simple peripheral chip selects
 - Parallel input/output ports, some with interrupt capability
 - Programmed interrupt vector response for on-chip peripheral modules
 - IEEE 1149.1 boundary scan test access port (JTAG)
- Operation from DC to 16.67 MHz (Processor Clock)
- Operating Voltages of 3.3V ± 0.3V and 5V ± 0.5V
- Compact 100-Lead Quad Flat Pack (QFP) Package

M68300 FAMILY

The MC68307 is one of a series of components in Motorola's M68300 family. Other members of the family include the MC68302, MC68306, MC68330, MC68331, MC68332, MC68F333, MC68334, MC68340, MC68341, MC68349, and MC68360.

ORGANIZATION

The M68300 family of integrated processors and controllers is built on an M68000 core processor and a selection of intelligent peripherals appropriate for a set of applications. Common system glue logic such as address decoding, wait state insertion, interrupt prioritization, and watchdog timing is also included.

Each member of the M68300 family is distinguished by its selection of on-chip peripherals. Peripherals are chosen to address specific applications but are often useful in a wide variety of applications. The peripherals may be highly sophisticated timing or protocol engines that have their own processors, or they may be more traditional peripheral functions, such as UARTs and timers.

ADVANTAGES

By incorporating so many major features into a single M68300 family chip, a system designer can realize significant savings in design time, power consumption, cost, board space, pin count, and programming. The equivalent functionality can easily require 20 separate components. Each component might have 16–64 pins, totaling over 350 connections. Most of these connections require interconnects or are duplications. Each connection is a candidate for a bad solder joint or misrouted trace. Each component is another part to qualify, purchase, inventory, and maintain. Each component requires a share of the printed circuit board. Each component draws power, which is often used to drive large buffers to get the signal to another chip. The cumulative power consumption of all the components must be available from the power supply. The signals between the central processing unit (CPU) and a peripheral might not be compatible nor run from the same clock, requiring time delays or other special design considerations.

In an M68300 family component, the major functions and glue logic are all properly connected internally, timed with the same clock, fully tested, and uniformly documented. Only essential signals are brought out to pins. The primary package is the surface-mount plastic QFP for the smallest possible footprint.

MC68307 ARCHITECTURE

To improve total system throughput and reduce part count, board size and cost of system implementation, the MC68307 integrates a powerful processor, intelligent peripheral modules, and typical system interface logic. These functions include the SIM07, timers, UART, M-bus interface, and 8051-compatible bus interface.

The EC000 processor core communicates with these modules via an internal bus, providing the opportunity for fully synchronized communication between all modules and allowing interrupts to be handled in parallel with data transfers, greatly improving system performance.

STATIC EC000 CORE

The EC000 core is a core implementation of the MC68000 32-bit microprocessor architecture. The features of the EC000 core processor include:

- Low power, static HCMOS implementation
- 24-bit address bus, 16-bit data bus
- Seventeen 32-bit data and address registers
- 56 powerful instruction types that support high level development languages
- 14 addressing modes and five main data types
- Seven priority levels for interrupt control

The EC000 core is completely upward user code-compatible with all other members of the M68000 microprocessor families and thus has access to a broad base of established real-time kernels, operating systems, languages, applications, and development tools.

EC000 Core Programming Model

The EC000 core offers sixteen 32-bit registers and a 32-bit program counter (see Figure 2). The first eight registers (D7–D0) are used as data registers for byte (8-bit), word (16-bit) and long-word (32-bit) operations. Because the use of the data registers will affect the condition code register (indicating negative number, carry, and overflow conditions) they are primarily used for data manipulation. The second set of seven registers (A6–A0) and the user stack pointer (USP) may be used as software stack pointers and base address registers. These registers can be used for word and long-word operations and do not affect the condition code register. All of the registers (D7–D0 and A6–A0) may be used as index registers.

In supervisor mode, the upper byte of the status register (SR) and the supervisor stack pointer (SSP) are also available to the programmer. These registers are shown in Figure 3.

The SR (refer to Figure 4) contains the interrupt mask (seven levels available) as well as the following condition codes: extend (X), negative (N), zero (Z), overflow (V), and carry (C). Additional status bits indicate whether the processor is in trace mode (T-bit) and in supervisor or user state (S-bit).

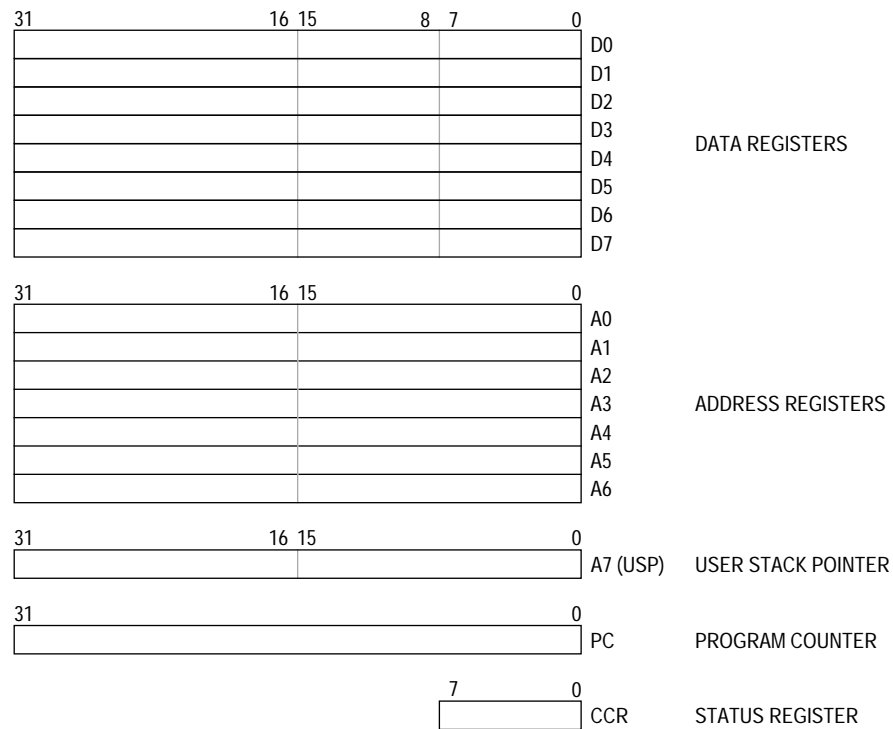


Figure 2. User Programming Model

Figure 3. Supervisor Programming Model Supplement

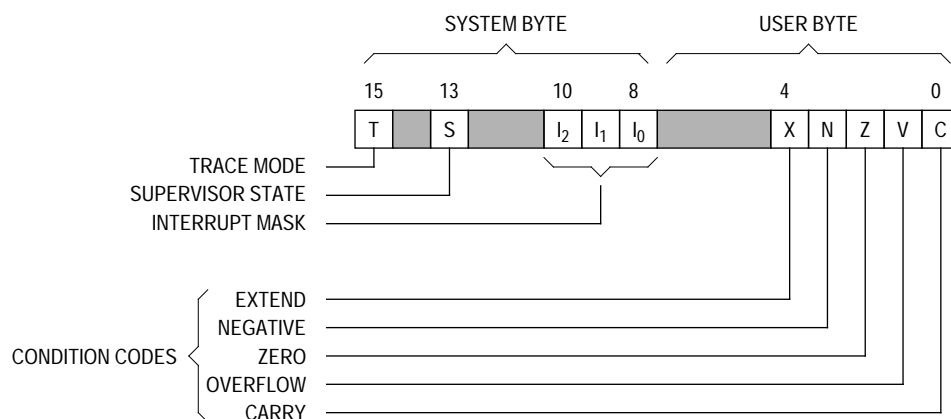


Figure 4. Status Register

Data Types and Addressing Modes

Five basic data types are supported:

- 1.) Bits
- 2.) Binary coded decimal (BCD) digits (4 bits)
- 3.) Bytes (8 bits)
- 4.) Words (16 bits)
- 5.) Long words (32 bits)

In addition, operations on other data types such as memory addresses, status word data, etc. are provided in the instruction set.

The 14 addressing modes listed in Table 1 include six basic types:

- 1.) Register direct
- 2.) Register indirect
- 3.) Absolute
- 4.) Program counter relative
- 5.) Immediate
- 6.) Implied

Included in the register indirect addressing modes is the capability to perform postincrementing, predecrementing, offsetting, and indexing. The program counter relative mode can also be modified via indexing and offsetting.

Instruction Set Overview

The EC000 core instruction set is listed in Table 2. The instruction set facilitates ease of programming by supporting high-level languages. Each instruction, with few exceptions, operates on bytes, words, and long-words, and most instructions can use any of the 14 addressing modes. Combining instruction types, data types, and addressing modes, over 1000 useful instructions are provided. These instructions include signed and unsigned, multiply and divide, quick arithmetic operations, BCD arithmetic, and expanded operations (through traps).

Table 1. Addressing Modes

| Addressing modes | Syntax |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------|
| Register direct addressing Data register direct Address register direct | Dn An |
| Absolute data addressing Absolute short Absolute long | xxx.W xxx.L |
| Program counter relative addressing Relative with offset Relative with index offset | d ₁₆ (PC) d ₈ (PC, Xn) |
| Register indirect addressing register Register indirect Postincrement register indirect Predecrement register indirect Register indirect with offset Indexed register indirect with offset | (An) (An)+ -(An) d ₁₆ (An) d ₈ (An, Xn) |
| Immediate data addressing Immediate Quick immediate | #xxx #1–#8 |
| Implied addressing Implied register | SR/USP/SP/PC |
| Legend: Dn = Data Register An = Address Register Xn = Address or Data Register Used as Index Register SR = Status Register PC = Program Counter SP = Stack Pointer USP = User Stack Pointer <> = Effective Address d ₈ = 8-Bit Offset (Displacement) d ₁₆ = 16-Bit Offset (Displacement) #xxx = Immediate Data | |

Table 2. Instruction Set

| Mnemonic | Description | Mnemonic | Description |
|-----------------|-------------------------------------------|-----------------|---------------------------------|
| ABCD | Add decimal with extend | MOVEM | Move multiple registers |
| ADD | Add | MOVEP | Move peripheral data |
| ADDA | Add address | MOVEQ | Move quick |
| ADDQ | Add quick | MOVE from SR | Move from status register |
| ADDI | Add immediate | MOVE to SR | Move to status register |
| ADDX | Add with extend | MOVE to CCR | Move to condition codes |
| AND | Logical AND | MOVE USP | Move user stack pointer |
| ANDI | AND immediate | MULS | Signed multiply |
| ANDI to CCR | AND immediate to condition codes | MULU | Unsigned multiply |
| ANDI to SR | AND immediate to status register | NBCD | Negate decimal with extend |
| ASL | Arithmetic shift left | NEG | Negate |
| ASR | Arithmetic shift right | NEGX | Negate with extend |
| Bcc | Branch conditionally | NOP | No operation |
| BCHG | Bit test and change | NOT | Ones complement |
| BCLR | Bit test and clear | OR | Logical OR |
| BRA | Branch always | ORI | OR immediate |
| BSET | Bit test and set | ORI to CCR | OR immediate to condition codes |
| BSR | Branch to subroutined set | ORI to SR | OR immediate to status register |
| BTST | Bit test | PEA | Push effective address |
| CHK | Check register against bounds | RESET | Reset external devices |
| CLR | Clear operand | ROL | Rotate left without extend |
| CMP | Compare | ROR | Rotate right without extend |
| CMPA | Compare address | ROXL | Rotate left with extend |
| CMPM | Compare memory | ROXR | Rotate right with extend |
| CMPI | Compare immediate | RTE | Return from exception |
| DBcc | Test cond, decrement and branch | RTR | Return and restore |
| DIVS | Signed divide | RTS | Return from subroutine |
| DIVU | Unsigned divide | SBCD | Subtract decimal with extend |
| EOR | Exclusive OR | Scc | Set conditional |
| EORI | Exclusive OR immediate | STOP | Stop |
| EORI to CCR | Exclusive OR immediate to condition codes | SUB | Subtract |
| EORI to SR | Exclusive OR immediate to status register | SUBA | Subtract address |
| EXG | Exchange registers | SUBI | Subtract immediate |
| EXT | Sign extend | SUBQ | Subtract quick |
| JMP | Jump | SUBX | Subtract with extend |
| JSR | Jump to subroutine | SWAP | Swap data register halves |
| LEA | Load effective address | TAS | Test and set operand |
| LINK | Link stack | TRAP | Trap |
| LSL | Logical shift left | TRAPV | Trap on overflow |
| LSR | Logical shift right | TST | Test |
| MOVE | Move | UNLK | Unlink |
| MOVEA | Move address | — | — |

SYSTEM INTEGRATION MODULE

The MC68307 system integration module (SIM07) consists of several functions that control the system start-up, initialization, configuration, and the external bus with a minimum of external devices.

The SIM07 features include:

- System configuration
- Oscillator & clock dividers
- Reset control, power-down mode control
- Chip-selects and wait states
- External bus interfaces, 68000 and 8051-compatible
- Parallel input/outputs with interrupt capability
- Interrupt configuration/response
- Software watchdog
- JTAG test access port

System Configuration

The MC68307 system configuration logic consists of a module base address register (MBAR) and a system control register (SCR) which together allow the user to configure operation of the following functions:

- Base address and address space of internal peripheral registers
- Low-power (stand-by) modes
- Hardware watchdog for system protection
- 8051-compatible bus
- Peripheral chip selects
- Data bus size control for chip selected address ranges

Chip Select Logic and Dynamic Bus Sizing

The MC68307 provides four programmable chip-select signals ($\overline{CS3}$ – $\overline{CS0}$). For a given chip-select block, the user may choose whether the chip-select allows read-only, write-only, or both read and write accesses, whether the chip-select should match only one function code value or all values, whether a \overline{DTACK} is automatically generated for this chip-select, and after how many wait states (from zero to six) the \overline{DTACK} will be generated. Each of the chip selects includes a dynamic bus-sizing extension to the basic 68000 bus which allows the system designer to mix 16-bit and 8-bit contiguous address memory devices (RAM, ROM) on a 16-bit data bus system.

An additional feature of $\overline{CS2}$ allows the user to opt either to use the programmable chip select $\overline{CS2}$ or to use four peripheral chip selects ($\overline{CS2A}$, $\overline{CS2B}$, $\overline{CS2C}$, and $\overline{CS2D}$). When the four peripheral chip selects are enabled, each one selects a 16-Kbyte block within the programmed range of $\overline{CS2}$.

External Bus Interface

The external bus interface handles the transfer of information between the internal EC000 core and the memory, peripherals, or other processing elements in the external address space. It consists of a 68000 bus interface and an 8051-compatible bus interface. The external 68000 bus provides up to 24 address lines and 16 data lines. Each bus access can appear externally either as a 68000 bus cycle (either 16-bit or 8-bit dynamic data bus width) or as an 8-bit wide 8051-compatible bus cycle (multiplexing 8 bits of address and data) with the appropriate sets of control signals.

Parallel General-Purpose I/O Ports

The MC68307 supports two general-purpose I/O ports, port A (8-bits) and port B (16-bits), whose pins can be configured as general-purpose I/O pins or as dedicated peripheral interface pins for the on-chip modules.

Each port pin can be independently programmed as general-purpose I/O pins, even when other pins related to the same on-chip peripheral are used as dedicated pins. Even if all the pins for a particular peripheral are configured as general-purpose I/O, the peripheral will still operate normally (although this is only useful in the case of the timer module). Power consumption may be reduced by turning off unused modules.

Interrupt Controller

The interrupt controller supports interrupts from three sources. The first source is an external, nonmaskable interrupt input on the $\overline{\text{TRQ}}7$ signal, which always causes an interrupt priority level 7 request to the EC000 core. Assuming no other source is programmed as a level 7 source, this input will always obtain the immediate attention of the core.

The second source is an external interrupt received through the 8-channel latched interrupt port ($\overline{\text{INT}}8\text{--}\overline{\text{INT}}1$). Each $\overline{\text{INT}}x$ signal can be programmed with an interrupt priority level, and each can have pending interrupts cleared independently of the others.

The third source of interrupts is the on-chip peripherals. The interrupt controller allows the user to assign the interrupt priority level each of the four on-chip peripherals will use, and to determine a particular vector number to be presented when the respective module receives an interrupt acknowledge from the processor via the interrupt controller logic.

Software Watchdog

A software watchdog timer is used to protect against system failures by providing a means to escape from unexpected input conditions, external events, or programming errors. Once started, the software watchdog timer must be cleared by software on a regular basis so that it never reaches its time-out value. Upon reaching the time-out value, the assumption is made that a system failure has occurred, and the software watchdog logic resets the MC68307.

Low-Power Stop Logic

Various options for power-saving are available: turning off unused peripherals, reducing processor clock speed, disabling the processor altogether or a combination of these.

A wake-up from power-down can be achieved by causing an interrupt at the interrupt controller logic which runs throughout the period of processor power-down. Any interrupt will cause a wake-up of the EC000 core followed by processing of that interrupt.

The on-chip peripherals can initiate a wake-up; for example, the timer can be set to wake-up after a certain elapsed time, or number of external events, or the UART can cause a wake-up on receiving serial data.

The clocks provided to the various internal modules can all be disabled to further reduce power consumption. In the case of the UART, its clock is restarted automatically by a transition on the RxD pin, so that incoming data is clocked in. When the data has been completely received, an interrupt from the UART wakes-up the processor core. If the other on-chip peripherals (the timer and M-bus) are required to cause a wake-up, then their clocks should not be disabled in this manner.

JTAG Test Access Port

To aid in system diagnostics the MC68307 includes dedicated user-accessible test logic that is fully compliant with the IEEE 1149.1 standard for boundary scan testability, often referred to as JTAG (joint test action group).

SIM07 Programming Model

The SIM07 programming model is listed in Tables 3–7. The FC (function code) column in each table indicates whether a register is restricted to supervisor access (S) or programmable to exist in either supervisor or user space (S/U). With the exception of the system configuration registers (listed in Table 3), the address column of each table contains the offset from the base address (MBASE) contained in the MBAR.

Table 3. SIM07 System Configuration Registers

| Address | FC | Register Name |
|----------|----|-------------------------------------|
| \$0000F0 | — | Reserved—No external bus access |
| \$0000F2 | S | Module Base Address Register (MBAR) |
| \$0000F4 | S | System Control Register (SCR) |
| \$0000F6 | S | System Control Register (SCR) |
| \$0000F8 | — | Reserved—No external bus access |
| \$0000FA | — | Reserved—No external bus access |
| \$0000FC | — | Reserved—No external bus access |
| \$0000FE | — | Reserved—No external bus access |

Table 4. SIM07 Chip Select Registers

| Address | FC | Register Name |
|-------------|-----|-------------------|
| MBASE+\$040 | S/U | Base register 0 |
| MBASE+\$042 | S/U | Option register 0 |
| MBASE+\$044 | S/U | Base register 1 |
| MBASE+\$046 | S/U | Option register 1 |
| MBASE+\$048 | S/U | Base register 2 |
| MBASE+\$04A | S/U | Option register 2 |
| MBASE+\$04C | S/U | Base register 3 |
| MBASE+\$04E | S/U | Option register 3 |

Table 5. SIM07 External Bus Interface Registers

| Address | FC | Register Name | |
|-------------|-----|----------------------------------------|----------------------------------------|
| MBASE+\$011 | S/U | Do not access byte \$010 | Port A control register (PACNT) |
| MBASE+\$013 | S/U | Do not access byte \$012 | Port A data direction register (PADDR) |
| MBASE+\$015 | S/U | Do not access byte \$014 | Port A data register (PADAT) |
| MBASE+\$016 | S/U | Port B control register (PBCNT) | |
| MBASE+\$018 | S/U | Port B data direction register (PBDDR) | |
| MBASE+\$01A | S/U | Port B data register (PBDAT) | |

Table 6. SIM07 Interrupt Controller Registers

| Address | FC | Register Name | |
|-------------|-----|----------------------------------------------|-----------------------------------------------|
| MBASE+\$020 | S/U | Latched interrupt control register 1 (LICR1) | |
| MBASE+\$022 | S/U | Latched interrupt control register 2 (LICR2) | |
| MBASE+\$024 | S/U | Peripheral interrupt control register (PICR) | |
| MBASE+\$027 | S/U | Do not access byte \$026 | Programmable interrupt vector register (PIVR) |

Table 7. SIM07 Software Watchdog Registers

| Address | FC | Register Name |
|-------------|-----|-----------------------------------|
| MBASE+\$12A | S/U | Watchdog reference register (WRR) |
| MBASE+\$12C | S/U | Watchdog counter register (WCR) |

DUAL TIMER MODULE

The MC68307 includes two independent, identical, general-purpose timers. Each general-purpose timer block contains a free-running 16-bit timer which can be used in various modes, to capture the timer value with an external event, to trigger an external event or interrupt when the timer reaches a set value, or to count external events. Each has an 8-bit prescaler to allow programmable clock input frequency derived from the system clock (divided by 1 or by 16) or external count input. The output pins (one per timer) have a variety of programmable modes and the output signal can be an active-low pulse or a toggle of the current output. The features of the 16-bit timer include:

- Maximum period of 16 seconds (at 16.67 MHz)
- 60-ns resolution (at 16.67 MHz)
- Programmable sources for the clock input, including external clock
- Input capture capability with programmable trigger edge on input pins
- Output compare with programmable mode for the output pins
- Two timers externally cascadeable to form a 32-bit timer
- Free-run and restart modes

Dual Timer Programming Model

Table 8 shows the programming model for the dual timer module. The FC (function code) column indicates whether a register is restricted to supervisor access (S) or programmable to exist in either supervisor or user space (S/U). The address column contains the offset from the base address (MBASE) contained in the SIM07 MBAR.

Table 8. Dual Timer Module Registers

| Address | FC | Register Name |
|-------------|-----|----------------------------------------------------------|
| MBASE+\$120 | S/U | Timer mode register 1 (TMR1) |
| MBASE+\$122 | S/U | Timer reference register 1 (TTR1) |
| MBASE+\$124 | S/U | Timer capture register 1 (TCR1) |
| MBASE+\$126 | S/U | Timer counter 1 (TCN1) |
| MBASE+\$129 | S/U | Do not access byte \$128 Timer event register 1 (TER1) |
| MBASE+\$130 | S/U | Timer mode register 2 (TMR2) |
| MBASE+\$132 | S/U | Timer reference register 2 (TRR1) |
| MBASE+\$134 | S/U | Timer capture register 2 (TCR2) |
| MBASE+\$136 | S/U | Timer counter 2 (TCN2) |
| MBASE+\$139 | S/U | Do not access byte \$138 Timer event register 2 (TER2) |

M-BUS INTERFACE MODULE

The M-bus is a two-wire, bidirectional serial bus which provides a simple and efficient means of data exchange between devices; it is fully compatible with the I²C bus standard. The maximum data rate is limited to 100 kbit/s at 16.67-MHz system clock speed. The maximum communication length and the number of devices that can be connected are limited by a maximum bus capacitance of 400 pF. The serial bit clock frequency of the M-bus is programmable and ranges from 3830 Hz to 757 kHz for a 16.67-MHz internal operating frequency.

The M-bus system is a true multimaster bus including collision detection and arbitration to prevent data corruption (when two or more masters intend to control the bus simultaneously). The M-bus system uses the SDA and SCL signals for data transfer. All devices connected to the M-bus interface must have open-drain or open-collector output; a logic AND function is exercised in both lines with pull-up resistors.

The features of the M-bus include:

- Fully compatible with I²C bus standard
- Multimaster operation
- Software programmable for one of 32 different serial clock frequencies
- Software selectable acknowledge bit
- Interrupt driven byte-by-byte data transfer
- Arbitration-lost driven interrupt with automatic mode switching from master to slave
- Calling address identification interrupt
- Generate/detect the START or STOP signal
- Repeated START signal generation
- Generate/recognize the acknowledge bit
- Bus busy detection

M-Bus Programming Model

The programming model for the M-bus module is listed in Table 9. The FC (function code) column indicates whether a register is restricted to supervisor access (S) or programmable to exist in either supervisor or user space (S/U). The address column contains the offset from the base address (MBASE) contained in the SIM07 MBAR.

Table 9. M-Bus Module Registers

| Address | FC | Register Name | |
|-------------|-----|--------------------------|-----------------------------------------|
| MBASE+\$141 | S/U | Do not access byte \$140 | M-bus address register (MADR) |
| MBASE+\$143 | S/U | Do not access byte \$142 | M-bus frequency divider register (MFDR) |
| MBASE+\$145 | S/U | Do not access byte \$144 | M-bus control register (MBCR) |
| MBASE+\$147 | S/U | Do not access byte \$146 | M-bus status register (MBSR) |
| MBASE+\$149 | S/U | Do not access byte \$148 | M-bus data I/O register (MBDR) |

UART MODULE

The UART module in the MC68307 is based on the MC68681 DUART, which is part of the M68000 family of peripherals which directly interfaces to the MC68000 processor via an asynchronous bus structure. The UART module consists of internal control logic, timing and baud-rate generator logic, interrupt control logic, and the serial communications channel. Only one serial channel is implemented for the MC68307.

Clocking is provided by the MC68307 system clock, via a programmable prescaler. This allows various baud rates to be chosen. Modem support is provided with request-to-send ($\overline{\text{RTS}}$) and clear-to-send ($\overline{\text{CTS}}$) signals available. The serial port can sustain data rates of 5Mbits/s.

The features of the UART include:

- Full-duplex asynchronous/synchronous receiver/transmitter channels
- Maximum data transfer: 1X clock—5 Mbits/s, 16X clock—625 Kbits/s
- Quadruple-buffered receiver data registers
- Double-buffered transmitter data registers
- Programmable baud rate for serial channel
 - User defined rate derived from a programmable timer
- Programmable data format
 - Five to eight data bits plus parity
 - Odd, even, no parity, or force parity
 - One, one and one-half, or two stop bits programmable in 1/16 bit increments
- Programmable channel modes for diagnostics
 - Normal (full duplex)/automatic echo/local loopback/remote loopback
- Automatic wake-up mode for multidrop applications
- Versatile interrupt system
 - Single interrupt output with eight maskable interrupting conditions
 - Interrupt vector output on interrupt acknowledge
- Parity, framing, and overrun error detection
- False-start bit detection
- Line-break detection and generation
- Detects break which originates in the middle of a character
- Interrupt or poll on start/stop break

UART Programming Model

The programming model for the UART module is listed in Table 10. The FC (function code) column indicates whether a register is restricted to supervisor access (S) or programmable to exist in either supervisor or user space (S/U). The address column contains the offset from the base address (MBASE) contained in the SIM07 MBAR.

Table 10. UART Module Registers

| Address | FC | Register Name | |
|-------------|-----|--------------------------|------------------------------------------------|
| MBASE+\$101 | S/U | Do not access byte \$100 | UART mode register (UMR1,UMR2) |
| MBASE+\$103 | S/U | Do not access byte \$102 | UART status/clock select register (USR,UCSR) |
| MBASE+\$105 | S/U | Do not access byte \$104 | UART command register (UCR) |
| MBASE+\$107 | S/U | Do not access byte \$106 | (read) UART receive buffer (UTB, URB) |
| MBASE+\$107 | S/U | Do not access byte \$106 | (write) UART transmit buffer (UTB, URB) |
| MBASE+\$109 | S/U | Do not access byte \$108 | (read) UART CTS change register (UCCR) |
| MBASE+\$109 | S/U | Do not access byte \$108 | (write) UART auxiliary control register (UACR) |
| MBASE+\$10B | S/U | Do not access byte \$10A | (read) UART interrupt status register (UISR) |
| MBASE+\$10B | S/U | Do not access byte \$10A | (write) UART interrupt mask register (UIMR) |
| MBASE+\$10D | S/U | Do not access byte \$10C | Baud rate gen prescaler msb (UBG1) |
| MBASE+\$10F | S/U | Do not access byte \$10E | Baud rate gen prescaler lsb (UBG2) |
| MBASE+\$119 | S/U | Do not access byte \$118 | UART interrupt vector register (UIVR) |
| MBASE+\$11B | S/U | Do not access byte \$11A | UART CTS unlatched input port (UCP) |
| MBASE+\$11D | S/U | Do not access byte \$11C | UART RTS output bit set cmd (URBS) |
| MBASE+\$11F | S/U | Do not access byte \$11E | UART RTS output bit reset cmd (URBR) |

EXTERNAL SIGNAL DESCRIPTIONS

Figure 5 shows the MC68307 input and output signals in their respective functional groups. Table 11 briefly describes each of the MC68307 signals.

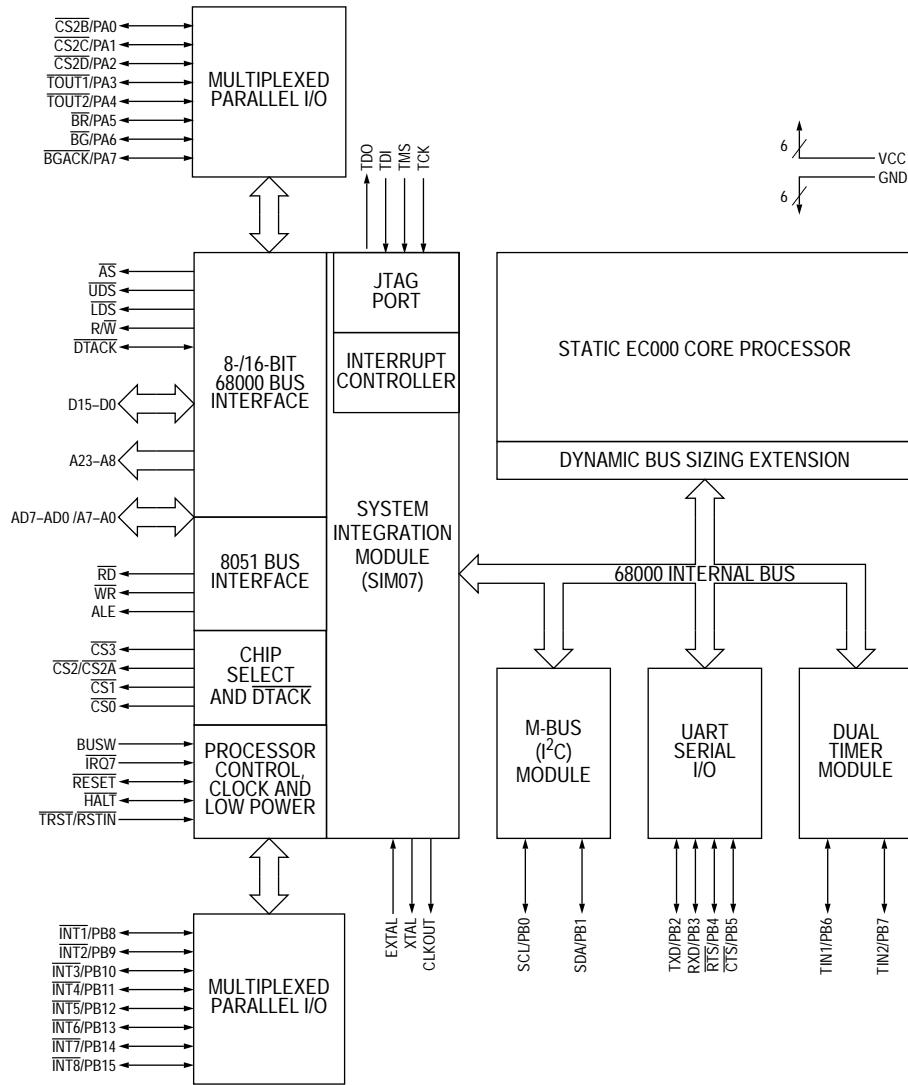


Figure 5. MC68307 Detailed Block Diagram

Table 11. Signal index

| Mnemonic | Description | Configuration |
|-------------------------|-------------------------------------------------|----------------------|
| D15-D0 | Data bus | Bidirectional |
| A23-A8 | Address bus out | Output |
| AD7-AD0/A7-A0 | Multiplexed 8051 address/data/Address bus out | Bidirectional |
| AS | Address strobe | Output |
| UDS | Upper data strobe | Output |
| LDS | Lower data strobe | Output |
| R/ \overline{W} | Read/write | Output |
| DTACK | Data acknowledge | Bidirectional |
| HALT | System halt | Bidirectional |
| RESET | System reset | Bidirectional |
| $\overline{TRST}/RSTIN$ | Power-on reset | Input |
| CS0 | Chip select 0 (ROM) | Output |
| CS1 | Chip select 1 (RAM) | Output |
| CS2/CS2A | Chip select 2 (peripherals) | Output |
| CS3 | Chip select 3 (8051) | Output |
| ALE | Address latch enable (8051) | Output |
| RD | 8051-bus read | Output |
| WR | 8051-bus write | Output |
| EXTAL | External clock/crystal in | Input |
| XTAL | External crystal | Output |
| CLKOUT | Clock to system | Output |
| BUSW | Initial data bus width for CS0 | Input |
| $\overline{CS2B}/PA0$ | Chip select 2B / I/O port A bit 0 | Bidirectional |
| $\overline{CS2C}/PA1$ | Chip select 2C / I/O port A bit 1 | Bidirectional |
| $\overline{CS2D}/PA2$ | Chip select 2D / I/O port A bit 2 | Bidirectional |
| $\overline{TOUT1}/PA3$ | Timer 1 output / I/O port A bit 3 | Bidirectional |
| $\overline{TOUT2}/PA4$ | Timer 2 output / I/O port A bit 4 | Bidirectional |
| $\overline{BR}/PA5$ | Bus request input / I/O port A bit 5 | Bidirectional |
| $\overline{BG}/PA6$ | Bus grant output / I/O port A bit 6 | Bidirectional |
| $\overline{BGACK}/PA7$ | Bus grant acknowledge output / I/O port A bit 7 | Bidirectional |
| IRQ7 | Interrupt level 7 | Input |
| SCL/PB0 | Serial M-bus clock / port B bit 0 | Bidirectional |
| SDA/PB1 | Serial M-bus data / port B bit 1 | Bidirectional |
| TxD/PB2 | UART transmit data / port B bit 2 | Bidirectional |
| RxD/PB3 | UART receive data / port B bit 3 | Bidirectional |
| $\overline{RTS}/PB4$ | Request-to-send / port B bit 4 | Bidirectional |
| $\overline{CTS}/PB5$ | Clear-to-send / port B bit 5 | Bidirectional |
| TIN1/PB6 | Timer 1 input / port B bit 6 | Bidirectional |
| TIN2/PB7 | Timer 2 input / port B bit 7 | Bidirectional |
| $\overline{INT1}/PB8$ | Interrupt in 1 / port B bit 8 | Bidirectional |
| $\overline{INT2}/PB9$ | Interrupt in 2 / port B bit 9 | Bidirectional |
| $\overline{INT3}/PB10$ | Interrupt in 3 / port B bit 10 | Bidirectional |
| $\overline{INT4}/PB11$ | Interrupt in 4 / port B bit 11 | Bidirectional |
| $\overline{INT5}/PB12$ | Interrupt in 5 / port B bit 12 | Bidirectional |
| $\overline{INT6}/PB13$ | Interrupt in 6 / port B bit 13 | Bidirectional |
| $\overline{INT7}/PB14$ | Interrupt in 7 / port B bit 14 | Bidirectional |
| $\overline{INT8}/PB15$ | Interrupt in 8 / port B bit 15 | Bidirectional |

ELECTRICAL CHARACTERISTICS

PRELIMINARY DC ELECTRICAL SPECIFICATIONS

| Characteristic | Symbol | Min | Max | Unit |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|------------------|-------------------------|---------|
| Input high voltage (except clock) | V_{IH} | 2.0 | V_{CC} | V |
| Input low voltage | V_{IL} | GND | 0.8 | V |
| Clock input high voltage | V_{IHC} | $0.7 V_{CC}$ | $V_{CC} + 0.3$ | V |
| Input leakage current @5.25V (all input-only pins) ^a | I_{IN} | - 2.5 | 2.5 | μA |
| Three-state (off state) input current @2.4V/0.4V | I_{TSI} | — | 20 | μA |
| Output high voltage (I_{OH} = rated maximum) | V_{OH} | $V_{CC} - 0.75$ | — | V |
| Output low voltage (I_{OL} = rated maximum) | V_{OL} | — | 0.5 | V |
| Current dissipation $V_{CC} = 5.0V \pm 0.5V^b$ $f_{EXT} = 16.67MHz$ $V_{CC} = 3.3V \pm 0.3V^b$ $f_{EXT} = 8MHz$ Low power STOP mode $V_{CC} = 5.0V \pm 0.5V$ $f_{EXT} = 16.67MHz$ $V_{CC} = 3.3V \pm 0.3V$ $f_{EXT} = 8MHz$ | I_D | — — — — | 30 TBD TBD TBD | mA |
| Power dissipation $V_{CC} = 5.0V \pm 0.5V$ $f_{EXT} = 16.67MHz$ $V_{CC} = 3.3V \pm 0.3V$ $f_{EXT} = 8MHz$ | P_D | — — | 0.26 TBD | W |
| Input capacitance ^c All input-only pins All I/O pins | C_{IN} | — — | 10 20 | pF |
| Load capacitance ^c All output pins (except SCL and SDA) SCL, SDA | C_L | — — | 100 400 | pF |

a. Not including internal pull-up or pull-down.

b. Currents listed are with no loading.

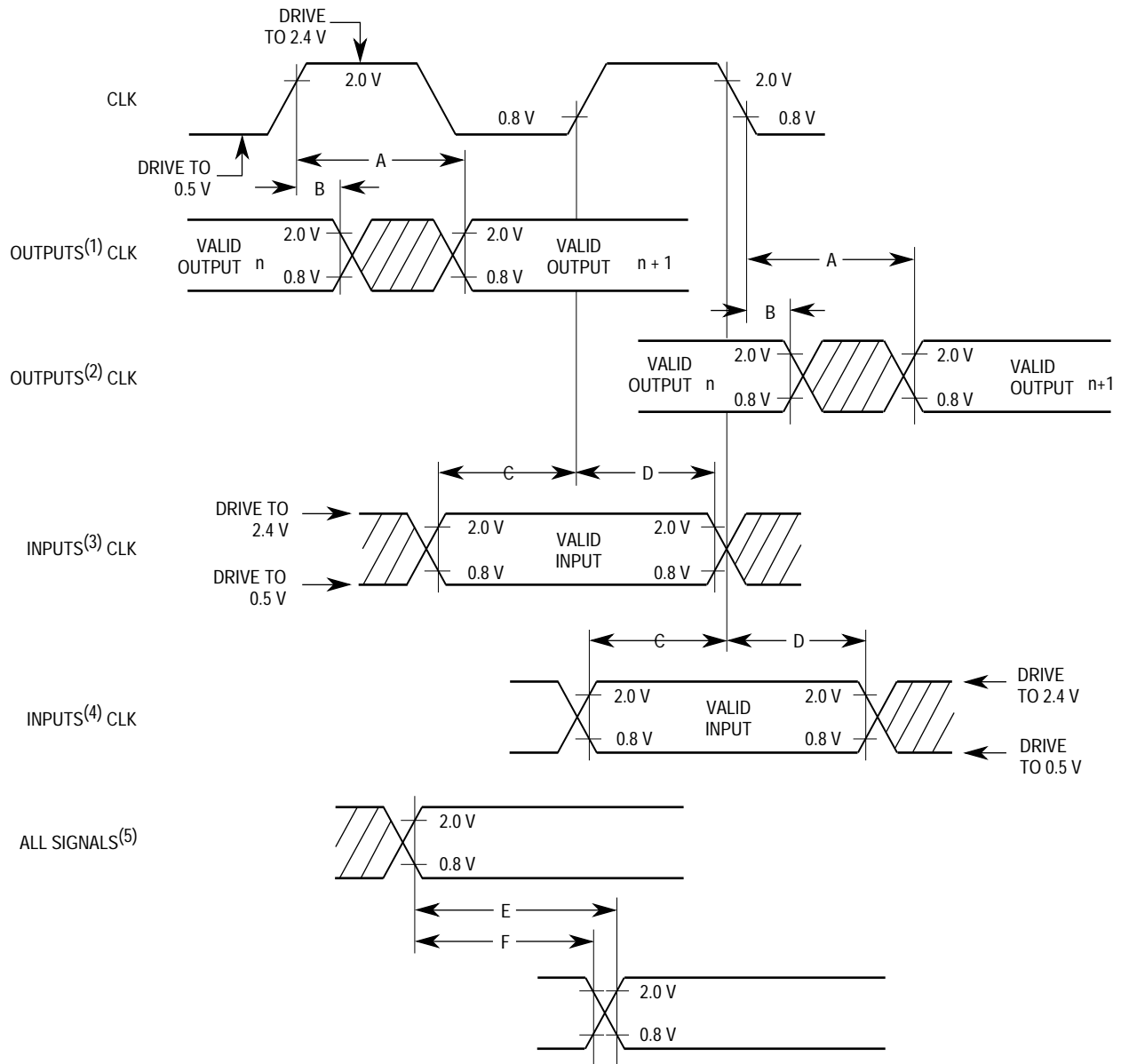
c. Capacitance is periodically sampled rather than 100% tested.

AC ELECTRICAL SPECIFICATION DEFINITIONS

The AC specifications presented consist of output delays, input setup and hold times and signal skew times. All signals are specified relative to an appropriate edge of the clock and possibly to one or more other signals.

The measurement of the AC specifications is defined by the waveforms shown in Figure 6. To test the parameters guaranteed by Motorola, inputs must be driven to the voltage levels specified in the figure. Outputs are specified with minimum and/or maximum limits, as appropriate, and are measured as shown. Inputs are specified with minimum setup and hold times, and are measured as shown. Finally, the measurement for signal-to-signal specifications are shown.

Note that the testing levels used to verify conformance to the AC specifications does not affect the guaranteed DC operation of the device as specified in the DC electrical characteristics.



NOTES:

1. This output timing is applicable to all parameters specified relative to the rising edge of the clock.
2. This output timing is applicable to all parameters specified relative to the falling edge of the clock.
3. This input timing is applicable to all parameters specified relative to the rising edge of the clock.
4. This input timing is applicable to all parameters specified relative to the falling edge of the clock.
5. This timing is applicable to all parameters specified relative to the assertion/negation of another signal.

LEGEND:

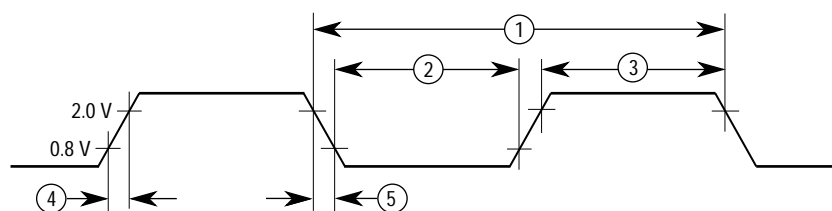
- A. Maximum output delay specification.
- B. Minimum output hold time.
- C. Minimum input setup time specification.
- D. Minimum input hold time specification.
- E. Signal valid to signal valid specification (maximum or minimum).
- F. Signal valid to signal invalid specification (maximum or minimum).

Figure 6. Drive Levels and Test Points for AC Specifications

PRELIMINARY AC ELECTRICAL SPECIFICATIONS—CONTROL TIMING

(See Figure 7)

| Num | Characteristic | 3.3 V | | 5 V | | Unit |
|-----|--------------------------|-------|------|-----------|-------|------|
| | | 8 MHz | | 16.67 MHz | | |
| | | Min | Max | Min | Max | |
| | Frequency of operation | 0.0 | 8.33 | 0.0 | 16.67 | MHz |
| 1 | Cycle time | 120 | — | 60 | — | ns |
| 2,3 | Clock pulse width | 54 | — | 27 | — | ns |
| 4,5 | Clock rise and fall time | — | 5 | — | 5 | ns |



NOTE: Timing measurements are referenced to and from a low voltage of 0.8 V and a high voltage of 2.0 V, unless otherwise noted. The voltage swing through this range should start outside and pass through the range such that the rise or fall will be linear between 0.8 V and 2.0 V.

Figure 7. Clock Timing

PRELIMINARY AC TIMING SPECIFICATIONS

($V_{CC} = 5.0V \pm 0.5V$ or $3.3Vdc \pm 0.3V$; $GND = 0Vdc$; $T_A = T_L$ to T_H) (See Figures 8–10)

| Num | Characteristic | 3.3V | | 5V | | Unit |
|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|-----|-----------|-----|------|
| | | 8.33 MHz | | 16.67 MHz | | |
| | | Min | Max | Min | Max | |
| 6 | Clock low to address valid | — | 60 | — | 30 | ns |
| 7 | Clock high to address, data bus high impedance (maximum) | — | 100 | — | 50 | ns |
| 8 | Clock high to address (minimum) | 0 | — | 0 | — | ns |
| 9 ^a | Clock high to \overline{AS} , \overline{CSx} , \overline{LDS} , \overline{UDS} asserted | 3 | 60 | 3 | 30 | ns |
| 11 ^b | Address valid to \overline{AS} , \overline{CSx} , \overline{LDS} , \overline{UDS} asserted (read) / \overline{AS} , \overline{CSx} asserted (write) | 30 | — | 15 | — | ns |
| 12 ^a | Clock low to \overline{AS} , \overline{CSx} , \overline{LDS} , \overline{UDS} negated | 3 | 60 | 3 | 30 | ns |
| 13 ^b | \overline{AS} , \overline{CSx} , \overline{LDS} , \overline{UDS} negated to address, FC invalid | 30 | — | 15 | — | ns |
| 14 ^b | \overline{AS} , \overline{CSx} , (and \overline{LDS} , \overline{UDS} read) width asserted | 240 | — | 120 | — | ns |
| 14A ^b | \overline{LDS} , \overline{UDS} width asserted | 100 | — | 50 | — | ns |
| 15 ^b | \overline{AS} , \overline{CSx} , \overline{LDS} , \overline{UDS} width negated | 120 | — | 60 | — | ns |
| 17 ^c | \overline{AS} , \overline{CSx} , \overline{LDS} , \overline{UDS} negated to R/\overline{W} invalid | 30 | — | 15 | — | ns |
| 18 ^a | Clock high to R/\overline{W} high (read) | 0 | 60 | 0 | 30 | ns |
| 20 | Clock high to R/\overline{W} low (write) | 0 | 60 | 0 | 30 | ns |
| 20A ^c | \overline{AS} , \overline{CSx} , asserted to R/\overline{W} low (write) | — | 20 | — | 10 | ns |
| 21 ^b | Address valid to R/\overline{W} low (write) | 0 | — | 0 | — | ns |
| 22 ^c | R/\overline{W} low to \overline{LDS} , \overline{UDS} asserted (write) | 60 | — | 30 | — | ns |
| 23 | Clock low to data-out valid (write) | — | 60 | — | 30 | ns |
| 25 ^b | \overline{AS} , \overline{CSx} , \overline{LDS} , \overline{UDS} negated to data-out invalid (write) | 30 | — | 15 | — | ns |
| 26 ^b | Data-out valid to \overline{LDS} , \overline{UDS} asserted (write) | 30 | — | 15 | — | ns |
| 27 ^d | Data-in valid to clock low (setup time on read) | 10 | — | 5 | — | ns |
| 28 ^b | \overline{AS} , \overline{CSx} , \overline{LDS} , \overline{UDS} negated to \overline{DTACK} negated (asynchronous hold) | 0 | 220 | 0 | 110 | ns |
| 29 | \overline{AS} , \overline{CSx} , \overline{LDS} , \overline{UDS} negated to data-in invalid (hold time on read) | 0 | — | 0 | — | ns |
| 29A | \overline{AS} , \overline{CSx} , \overline{LDS} , \overline{UDS} negated to data-in high impedance | — | 180 | — | 90 | ns |
| 30 | \overline{AS} , \overline{CSx} , \overline{LDS} , \overline{UDS} negated to \overline{BR} negated | 0 | — | 0 | — | ns |
| 31 | \overline{DTACK} asserted to data-in valid (setup time) | — | 100 | — | 50 | ns |
| 32 | \overline{HALT} and \overline{RESET} input transition time | 0 | 300 | 0 | 150 | ns |
| 33 | Clock high to \overline{BG} asserted | 0 | 40 | 0 | 20 | ns |
| 34 | Clock high to \overline{BG} negated | 0 | 40 | 0 | 20 | ns |
| 35 | \overline{BR} asserted to \overline{BG} asserted | 1.5 | 3.5 | 1.5 | 3.5 | Clks |
| 36 | \overline{BR} negated to \overline{BG} negated | 1.5 | 3.5 | 1.5 | 3.5 | Clks |
| 37 | \overline{BGACK} asserted to \overline{BG} asserted | 1.5 | 3.5 | 1.5 | 3.5 | Clks |
| 38 | \overline{BG} asserted to control, address, data bus high impedance (\overline{AS} , \overline{CSx} negated) | — | 100 | — | 50 | ns |
| 39 | \overline{BG} width negated | 1.5 | — | 1.5 | — | Clks |
| 46 | \overline{BGACK} width low | 1.5 | — | 1.5 | — | Clks |
| 47 ^d | Asynchronous input setup time | 10 | — | 5 | — | ns |
| 53 | Data-out hold from clock high | 0 | — | 0 | — | ns |

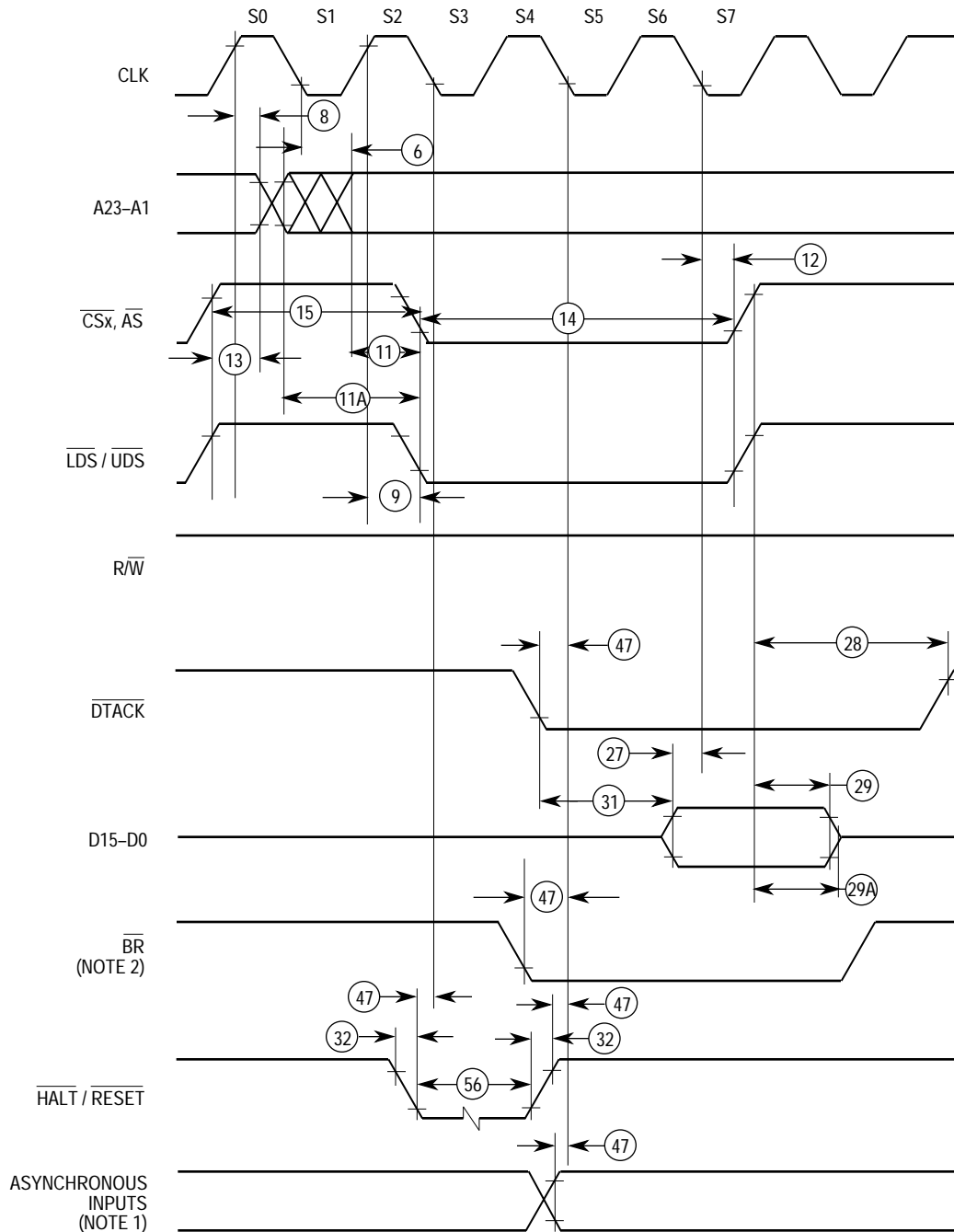
PRELIMINARY AC TIMING SPECIFICATIONS

($V_{CC} = 5.0V \pm 0.5V$ or $3.3Vdc \pm 0.3V$; $GND = 0Vdc$; $T_A = T_L$ to T_H) (See Figures 8–10)

| Num | Characteristic | 3.3V | | 5V | | Unit |
|-----------------|------------------------------------------------|----------|-----|-----------|-----|------|
| | | 8.33 MHz | | 16.67 MHz | | |
| | | Min | Max | Min | Max | |
| 55 | R/W asserted to data bus impedance change | 40 | — | 20 | — | ns |
| 56 ^e | HALT/RESET pulse width | 10 | — | 10 | — | Clks |
| 57 | BGACK negated to AS, CSx, LDS, UDS, R/W driven | 1.5 | — | 1.5 | — | Clks |
| 58 | BR negated to AS, CSx, LDS, UDS, R/W driven | 1.5 | — | 1.5 | — | Clks |

NOTES:

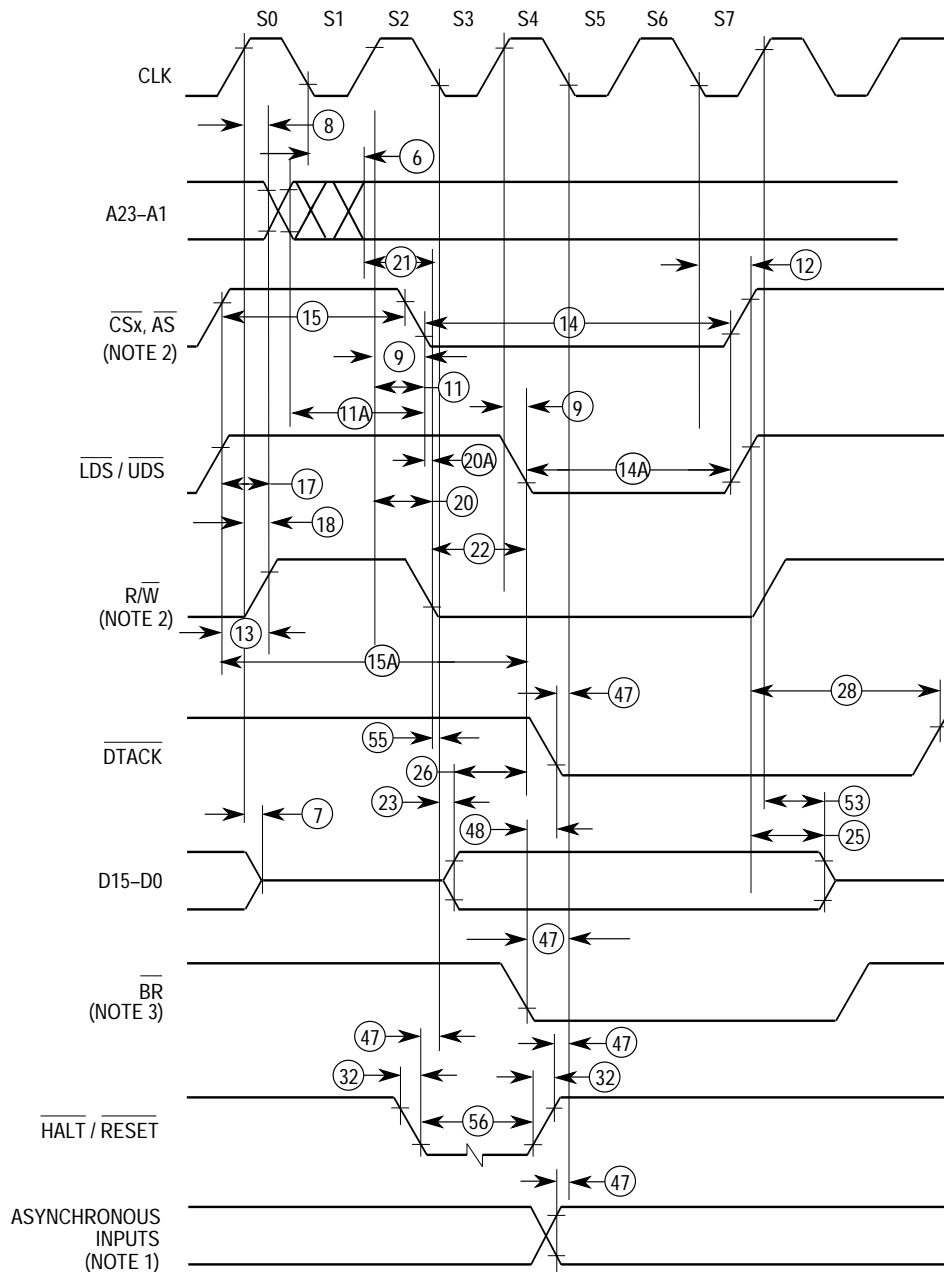
- For a loading capacitance of less than or equal to 50 pF, subtract 5 ns from the value given in the maximum columns.
- Actual value depends on clock period.
- When \overline{AS} , \overline{CSx} and R/\overline{W} are equally loaded ($\pm 20\%$), subtract 5 ns from the values given in these columns.
- If the asynchronous input setup time (#47) requirement is satisfied for \overline{DTACK} , the \overline{DTACK} asserted to data setup time (#31) requirement can be ignored. The data must only satisfy the data-in to clock low setup time (#27) for the following clock cycle.
- For power-up, the MC68307 must be held in the reset state for 128 clock cycles after CLK and V_{CC} become stable to allow stabilization of on-chip circuitry. After the system is powered up, #56 refers to the minimum pulse width required to reset the controller.



NOTES:

1. Setup time (#47) for asynchronous inputs ($\overline{\text{HALT}}$, $\overline{\text{RESET}}$, $\overline{\text{BR}}$, $\overline{\text{BGACK}}$, $\overline{\text{DTACK}}$) guarantees their recognition at the next falling edge of the clock.
2. $\overline{\text{BR}}$ need fall at this time only to ensure being recognized at the end of the bus cycle.

Figure 8. Read Cycle Timing Diagram



NOTES:

1. Setup time (#47) for asynchronous inputs ($\overline{\text{HALT}}$, $\overline{\text{RESET}}$, $\overline{\text{BR}}$, $\overline{\text{BGACK}}$, $\overline{\text{DTACK}}$) guarantees their recognition at the next falling edge of the clock.
2. Because of loading variations, $\overline{\text{R/W}}$ may be valid after $\overline{\text{AS}}$ even though both are initiated by the rising edge of S2 (specification #20A).
3. $\overline{\text{BR}}$ need fall at this time only to ensure being recognized at the end of the bus cycle.

Figure 9. Write Cycle Timing Diagram

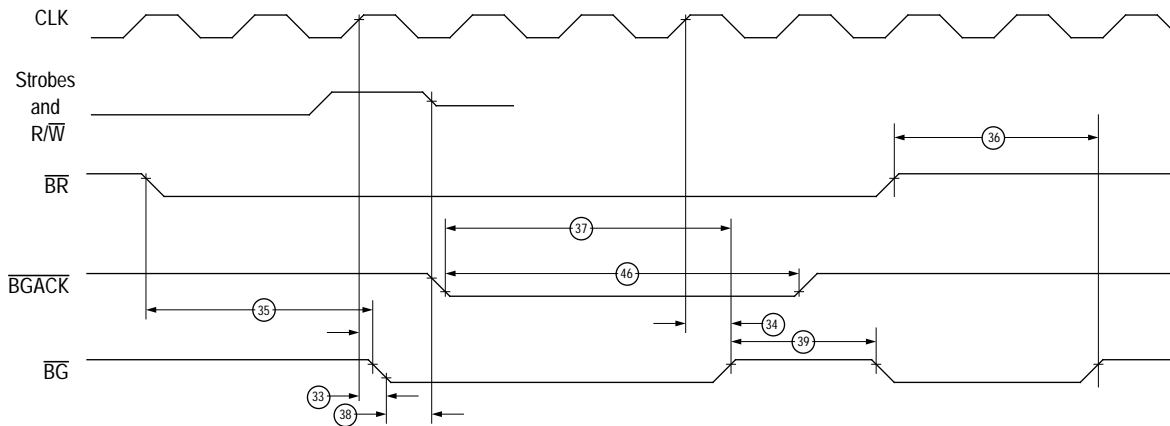


Figure 10. Bus Arbitration Timing

PRELIMINARY 8051 BUS INTERFACE MODULE AC ELECTRICAL SPECIFICATIONS

($V_{CC} = 5.0V \pm 0.5V$ or $3.3Vdc \pm 0.3V$; $GND = 0Vdc$; $T_A = T_L$ to T_H) (See Figures 11 and 12)

| Symbol | Characteristic | 3.3V | | 5V | | Unit |
|------------------|-------------------------------------------------------|----------------------------|----------------------------|----------------------------|----------------------------|------|
| | | 8 MHz | | 16.67 MHz | | |
| | | Min | Max | Min | Max | |
| t _{cyc} | Cycle time | 120 | — | 60 | — | ns |
| TLHLL | ALE pulse width | 2 x t _{cyc} – 40 | — | 2 x t _{cyc} – 40 | — | ns |
| TAVLL | Address valid to ALE low | t _{cyc} – 40 | — | t _{cyc} – 40 | — | ns |
| TLLAX | Address hold after ALE low | t _{cyc} – 35 | — | t _{cyc} – 35 | — | ns |
| TRLRH | \overline{RD} pulse width ^a | 5 x t _{cyc} | — | 5 x t _{cyc} | —8051 | ns |
| TWLWH | \overline{WR} pulse width ⁽¹⁾ | 5 x t _{cyc} | — | 5 x t _{cyc} | — | ns |
| TRLDV | \overline{RD} low to valid data in ⁽¹⁾ | — | 5 x t _{cyc} – 165 | — | 5 x t _{cyc} – 165 | ns |
| TRHDX | Data hold after \overline{RD} | 0 | — | 0 | — | ns |
| TRHDZ | Data float after \overline{RD} | — | 0.5 x t _{cyc} | — | 0.5 x t _{cyc} | ns |
| TLLDV | ALE low to valid data in ⁽¹⁾ | — | 8 x t _{cyc} – 150 | — | 8 x t _{cyc} – 150 | ns |
| TAVDV | Address to valid data in ⁽¹⁾ | — | 9 x t _{cyc} – 165 | — | 9 x t _{cyc} – 165 | ns |
| TLLWL | ALE low to \overline{RD} or \overline{WR} low | 3 x t _{cyc} – 50 | 3 x t _{cyc} + 50 | 3 x t _{cyc} – 50 | 3 x t _{cyc} + 50 | ns |
| TAVWL | Address to \overline{RD} low or \overline{WR} low | 4 x t _{cyc} – 130 | — | 4 x t _{cyc} – 130 | — | ns |
| TQVWX | Data valid to \overline{WR} transition | t _{cyc} – 60 | — | t _{cyc} – 60 | — | ns |
| TQVWH | Data valid to \overline{WR} high ⁽¹⁾ | 7 x t _{cyc} – 150 | — | 7 x t _{cyc} – 150 | — | ns |
| TWHQX | Data held after \overline{WR} | t _{cyc} – 50 | — | t _{cyc} – 50 | — | ns |
| TRLAZ | \overline{RD} low to address float | — | — | — | — | ns |
| TWHLH | \overline{RD} or \overline{WR} high to ALE high | t _{cyc} – 40 | t _{cyc} + 50 | t _{cyc} – 40 | t _{cyc} + 50 | ns |

NOTE:

a. Wait states can be added.

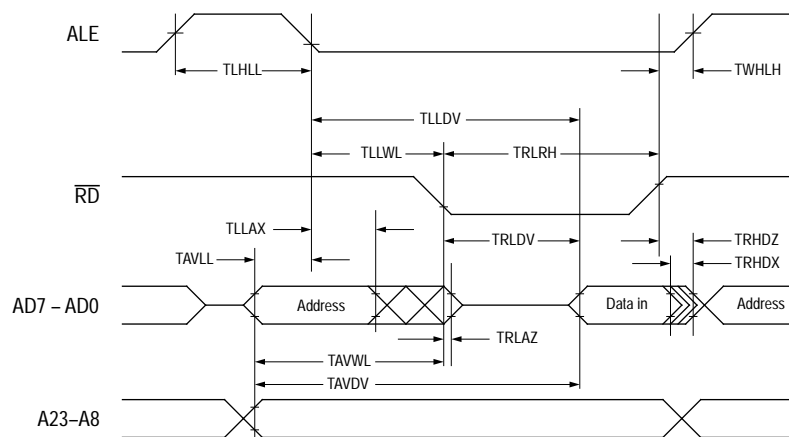


Figure 11. External Dat3a Memory Read Cycle

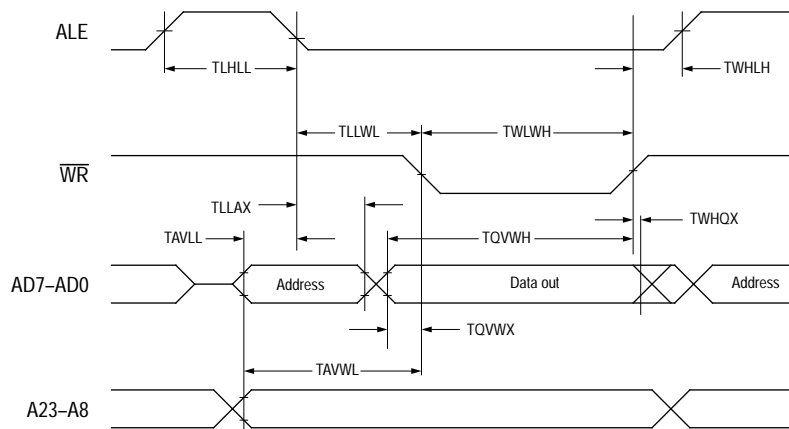


Figure 12. External Data Memory Write Cycle

PRELIMINARY IEEE 1149.1 ELECTRICAL SPECIFICATIONS

($V_{CC} = 5.0V \pm 0.5V$ or $3.3Vdc \pm 0.3V$; $GND = 0Vdc$; $T_A = T_L$ to T_H) (See Figures 13–15)

| Num | Characteristic | 3.3V | | 5V | | Unit |
|-----|-----------------------------------------|-------|------|-----------|------|------|
| | | 8 MHz | | 16.67 MHz | | |
| | | Min | Max | Min | Max | |
| | TCK frequency of operation | 0 | 10.0 | 0 | 10.0 | MHz |
| 1 | TCK cycle time | 100 | — | 100 | — | ns |
| 2 | TCK clock pulse width measured at 1.5 V | 45 | — | 45 | — | ns |
| 3 | TCK rise and fall times | 0 | 5 | 0 | 5 | ns |
| 6 | Boundary scan input data setup time | 15 | — | 15 | — | ns |
| 7 | Boundary scan input data hold time | 15 | — | 15 | — | ns |
| 8 | TCK low to output data valid | 0 | 80 | 0 | 80 | ns |
| 9 | TCK low to output high impedance | 0 | 80 | 0 | 80 | ns |
| 10 | TMS, TDI data setup time | 15 | — | 15 | — | ns |
| 11 | TMS, TDI data hold time | 15 | — | 15 | — | ns |
| 12 | TCK low to TDO data valid | 0 | 30 | 0 | 30 | ns |
| 13 | TCK low to TDO high impedance | 0 | 30 | 0 | 30 | ns |
| 14 | TRST width low | 80 | — | 80 | — | ns |

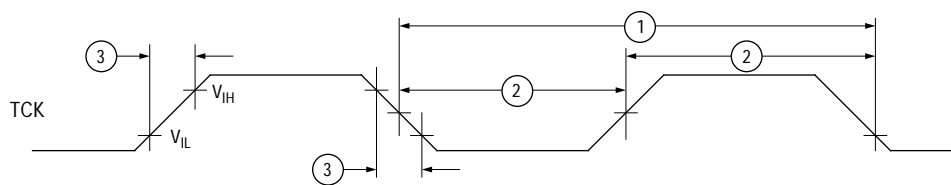


Figure 13. Test Clock Input Timing Diagram

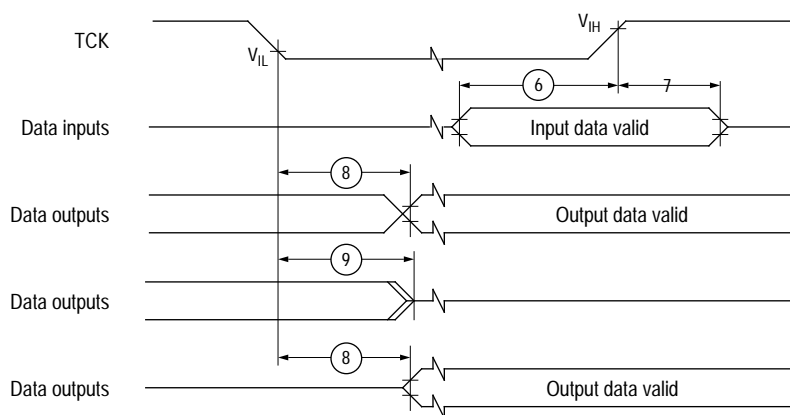


Figure 14. Boundary Scan Timing Diagram

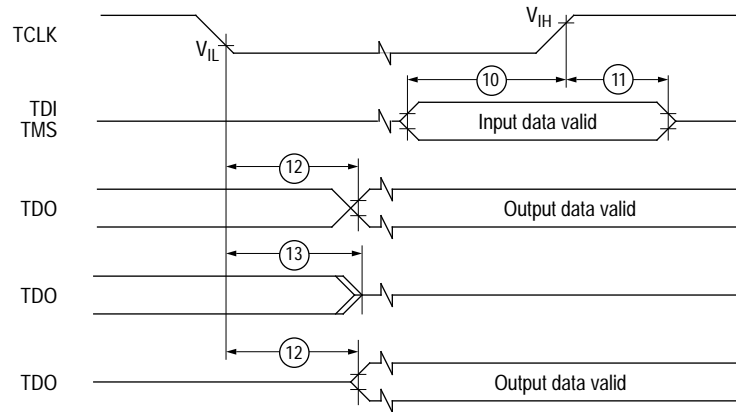


Figure 15. Test Access Port Timing Diagram

PRELIMINARY TIMER MODULE ELECTRICAL SPECIFICATIONS

($V_{CC} = 5.0V \pm 0.5V$ or $3.3Vdc \pm 0.3V$; $GND = 0Vdc$; $T_A = T_L$ to T_H) (See Figure 16)

| Num | Characteristic | 3.3V | | 5V | | Unit |
|------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------|-------|-----|-----------|-----|------|
| | | 8 MHz | | 16.67 MHz | | |
| | | Min | Max | Min | Max | |
| 1 | Timer input capture pulse width | 100 | — | 50 | — | ns |
| 2 | TINclock low pulse width | 100 | — | 50 | — | ns |
| 3 | TIN clock high pulse width and input capture high pulse width | 2 | — | 2 | — | Clk |
| 4 | TIN clock cycle time | 3 | — | 3 | — | Clk |
| 5 | Clock high to TOUT valid | — | 70 | — | 35 | ns |
| NOTE: The TIN specifications do not apply to the use of TIN1 as a baud rate generator input clock. In such a case, specifications may be used. | | | | | | |

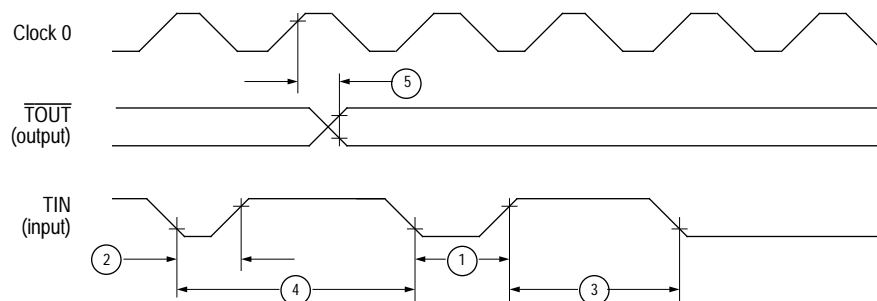


Figure 16. Timer Module Timing Diagram

PRELIMINARY UART ELECTRICAL SPECIFICATIONS

($V_{CC} = 5.0V \pm 0.5V$ or $3.3Vdc \pm 0.3V$; $GND = 0Vdc$; $T_A = T_L$ to T_H) (See Figures 17 and 18)

| Num | Characteristic | 3.3V | | 5V | | Unit |
|-----|----------------------------------|-------|-----|-----------|-----|------|
| | | 8 MHz | | 16.67 MHz | | |
| | | Min | Max | Min | Max | |
| 1 | TxD output valid from TxC low | — | 700 | — | 350 | ns |
| 2 | RxD data setup time to RxC high | 480 | — | 240 | — | ns |
| 3 | RxD data hold time from RxC high | 400 | — | 200 | — | ns |

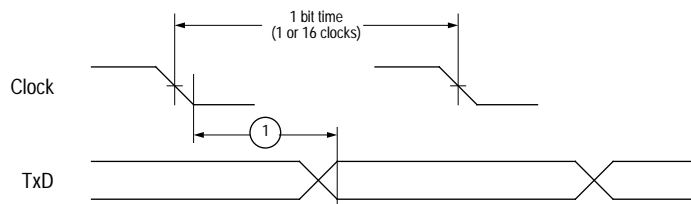


Figure 17. Transmitter Timing

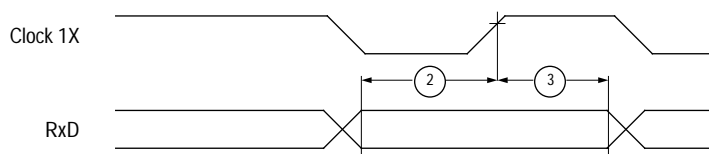


Figure 18. Receiver Timing

PRELIMINARY M-BUS INTERFACE INPUT SIGNAL TIMING

($V_{CC} = 5.0V \pm 0.5V$ or $3.3Vdc \pm 0.3V$; $GND = 0Vdc$; $T_A = T_L$ to T_H) (See Figure 19)

| Num | Characteristic | 3.3V | | 5V | | Unit |
|-----|----------------------------------------------------------------|-------|-----|-----------|-----|------|
| | | 8 MHz | | 16.67 MHz | | |
| | | Min | Max | Min | Max | |
| 1 | Start condition hold time | 2 | — | 2 | — | Clk |
| 2 | Clock low period | 4.7 | — | 4.7 | — | Clk |
| 3 | SDA/SCL rise time | — | 2 | — | 1 | μs |
| 4 | Data hold time | 0 | — | 0 | — | Clk |
| 5 | SDA/SCL fall time | — | 600 | — | 300 | ns |
| 6 | Clock high period | 4 | — | 4 | — | Clk |
| 7 | Data setup time | 500 | — | 250 | — | μs |
| 8 | Start condition setup time (for repeated start condition only) | 2 | — | 2 | — | Clk |
| 9 | Stop condition setup time | 2 | — | 2 | — | Clk |

PRELIMINARY M-BUS INTERFACE OUTPUT SIGNAL TIMING

($V_{CC} = 5.0V \pm 0.5V$ or $3.3Vdc \pm 0.3V$; $GND = 0Vdc$; $T_A = T_L$ to T_H) (See Figure 19)

| Num | Characteristic | 3.3V | | 5V | | Unit |
|-----|----------------------------------------------------------------|----------------|-----|----------------|-----|------|
| | | 8 MHz | | 16.67 MHz | | |
| | | Min | Max | Min | Max | |
| 1 | Start condition hold time | 8 | — | 8 | — | Clk |
| 2 | Clock low period | 11 | — | 11 | — | Clk |
| 3 | SDA/SCL rise time | — | 2 | — | 1 | μs |
| 4 | Data hold time | 0 | 2 | 0 | 2 | Clk |
| 5 | SDA/SCL fall time | — | 600 | — | 300 | ns |
| 6 | Clock high period | 11 | — | 11 | — | Clk |
| 7 | Data setup time | (Spec 2) x Clk | — | (Spec 2) x Clk | — | ns |
| 8 | Start condition setup time (for repeated start condition only) | 20 | — | 10 | — | Clk |
| 9 | Stop condition setup time | 20 | — | 10 | — | Clk |

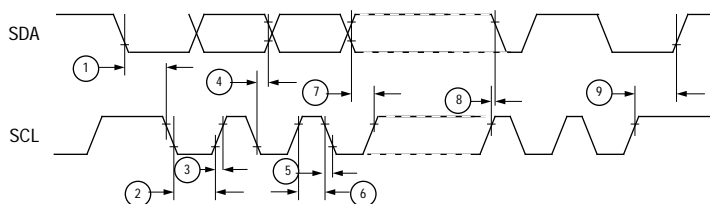


Figure 19. M-Bus Interface Input/Output Signal Timing

MECHANICAL DATA

The MC68307 is available in a 100-lead QFP package (FG suffix). Figure 20 shows the MC68307 pinout. Figure 21 shows the case drawing for the MC68307.

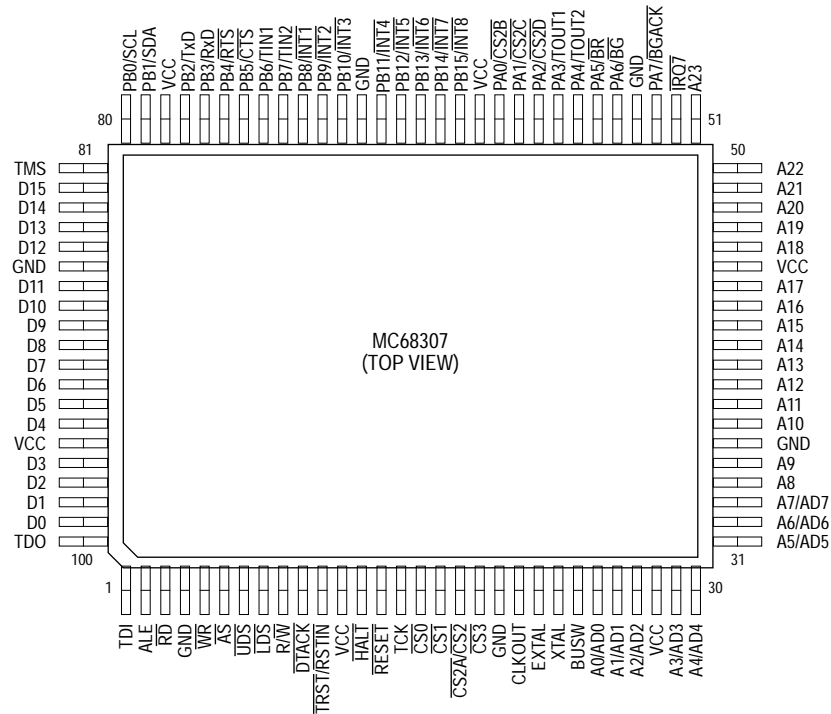


Figure 20. MC68307 FG Suffix—Pinout

MC68307FG
CASE 842B-01

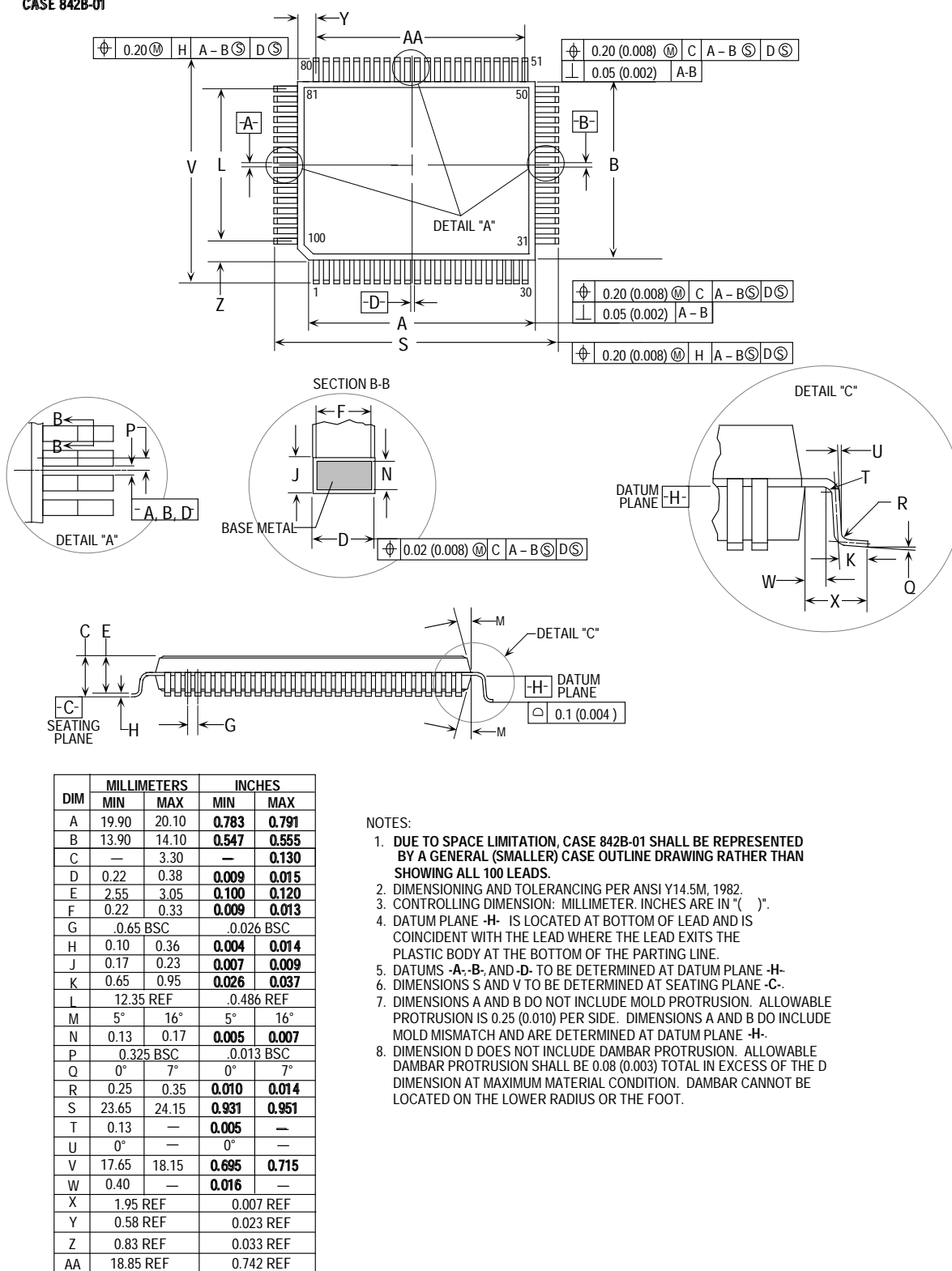



Figure 21. MC68307 FG Suffix—Package Dimensions

MORE INFORMATION

The documents listed in the following table contain detailed information on the MC68307. These documents may be obtained from the Literature Distribution Centers at the addresses listed below.

Table 12. Documentation

| Document Title | Order Number | Contents |
|----------------------------------------------------|--------------|----------------------------------------------------------------------|
| <i>M68300 Integrated Processor Family</i> | BR1114/D | M68300 Family Overview |
| <i>MC68307 User's Manual</i> | MC68307UM/AD | Detailed information for design |
| <i>M68000 Family Programmer's Reference Manual</i> | M68000PM/AD | M68000 Family Instruction Set |
| <i>The 68K Source</i> | BR729/D | Independent vendor listing supporting software and development tools |

Motorola reserves the right to make changes without further notice to any products herein. Motorola makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Motorola assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters can and do vary in different applications. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. Motorola does not convey any license under its patent rights nor the rights of others. Motorola products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Motorola product could create a situation where personal injury or death may occur. Should Buyer purchase or use Motorola products for any such unintended or unauthorized application, Buyer shall indemnify and hold Motorola and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Motorola was negligent regarding the design or manufacture of the part. Motorola and  are registered trademarks of Motorola, Inc. Motorola, Inc. is an Equal Opportunity/Affirmative Action Employer.

Literature Distribution Centers:

USA: Motorola Literature Distribution; P.O. Box 20912, Arizona 85036.

EUROPE: Motorola Ltd.; European Literature Centre; 88 Tanners Drive, Blakelands, Milton Keynes, MK14 5BP, England.

JAPAN: Nippon Motorola Ltd.; 4-32-1, Nishi-Gotanda, Shinagawa-ku, Tokyo 141 Japan.

ASIA-PACIFIC: Motorola Semiconductors H.K. Ltd.; Silicon Harbour Center, No. 2 Dai King Street, Tai Po Industrial Estate, Tai Po, N.T., Hong Kong.

**MOTOROLA**