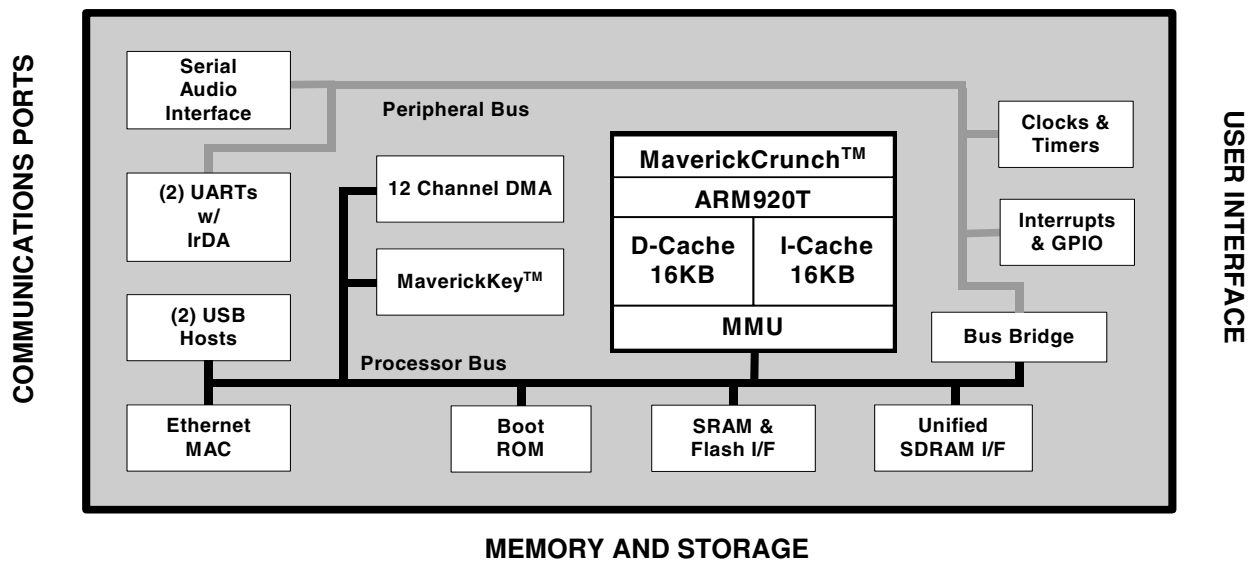


FEATURES

- 200-MHz ARM920T Processor
 - 16-kbyte Instruction Cache
 - 16-kbyte Data Cache
 - Linux®, Microsoft® Windows® CE-enabled MMU
 - 100-MHz System Bus
- MaverickCrunch™ Math Engine
 - Floating point, Integer and Signal Processing Instructions
 - Optimized for digital music compression and decompression algorithms.
 - Hardware interlocks allow in-line coding.
- MaverickKey™ IDs
 - 32-bit unique ID can be used for DRM-compliant, 128-bit random ID.
- Integrated Peripheral Interfaces
 - 16-bit SDRAM Interface (up to 4 banks)
 - 16-bit SRAM / FLASH / ROM
 - Serial EEPROM Interface
 - 1/10/100 Mbps Ethernet MAC
 - Two UARTs
 - Two-port USB 2.0 Full-speed Host (OHCI) (12 Mbits per second)
 - IrDA Interface
 - ADC
 - Serial Peripheral Interface (SPI) Port

**High-speed ARM9
System-on-chip Processor
with MaverickCrunch**

- 6-channel Serial Audio Interface (I²S)
- 2-channel, Low-cost Serial Audio Interface (AC'97)
- Internal Peripherals
 - 12 Direct Memory Access (DMA) Channels
 - Real-time Clock with Software Trim
 - Dual PLL controls all clock domains.
 - Watchdog Timer
 - Two General-purpose 16-bit Timers
 - One General-purpose 32-bit Timer
 - One 40-bit Debug Timer
- Interrupt Controller
- Boot ROM
- Package
 - 208-pin LQFP



OVERVIEW

The EP9302 is an ARM920T-based system-on-a-chip design with a large peripheral set targeted to a variety of applications:

- Industrial controls
- Digital media servers
- Integrated home media gateways
- Digital audio jukeboxes
- Streaming audio players
- Set-top boxes
- Point-of-sale terminals
- Thin clients
- Biometric security systems
- GPS & fleet management systems
- Educational toys
- Industrial computers
- Industrial hand-held devices
- Voting machines
- Medical equipment

The EP9302 is one of a series of ARM920T-based devices. Other members of the family have different peripheral sets, coprocessors and package configurations.

The ARM920T microprocessor core with separate 16 kbyte, 64-way set-associative instruction and data caches is augmented by the MaverickCrunch™ co-processor enabling faster than real-time compression of audio CDs.

The MaverickKey™ unique hardware programmed IDs are a solution to the growing concern over secure web content and commerce. With Internet security playing an important role in the delivery of digital media such as books or music, traditional software methods are quickly becoming unreliable. The MaverickKey unique IDs provide OEMs with a method of utilizing specific hardware IDs such as those assigned for SDMI (Secure Digital Music Initiative) or any other authentication mechanism.

A high-performance 1/10/100 Mbps Ethernet media access controller (EMAC) is included along with external interfaces to SPI, AC'97, and I²S audio. A two-port USB 2.0 Full-speed Host (OHCI) (12 Mbits per second), two UARTs, and a analog voltage measurement analog-to-digital converter (ADC) are included as well.

The EP9302 is a high-performance, low-power RISC-based, single-chip computer built around an ARM920T microprocessor core with a maximum operating clock rate of 200 MHz (184 MHz for industrial conditions). The ARM core operates from a 1.8 V supply, while the I/O operates at 3.3 V with power usage between 100 mW and 750 mW (dependent on speed).

Table A. Change History

| Revision | Date | Changes |
|----------|------------|--|
| PP1 | June 2004 | Initial Release. |
| PP2 | July 2004 | Update AC data. Add ADC data. |
| PP3 | March 2005 | Update electrical specs with most-current characterization data. |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |

Table of Contents

| | |
|--|-----------|
| Processor Core - ARM920T | 6 |
| MaverickCrunch™ Math Engine | 6 |
| MaverickKey™ Unique ID | 6 |
| General Purpose Memory Interface (SDRAM, SRAM, ROM, Flash) | 6 |
| Ethernet Media Access Controller (MAC) | 7 |
| Serial Interfaces (SPI, I2S, and AC '97) | 7 |
| 12-bit Analog-to-digital Converter (ADC) | 7 |
| Universal Asynchronous Receiver/Transmitters (UARTs) | 8 |
| Dual-port USB Host | 8 |
| Two-wire Interface | 8 |
| Real-Time Clock with Software Trim | 8 |
| PLL and Clocking | 9 |
| Timers | 9 |
| Interrupt Controller | 9 |
| Dual LED Drivers | 9 |
| General Purpose Input/Output (GPIO) | 9 |
| Reset and Power Management | 10 |
| Hardware Debug Interface | 10 |
| 12-Channel DMA Controller | 10 |
| Internal Boot ROM | 10 |
| Electrical Specifications | 11 |
| Absolute Maximum Ratings | 11 |
| Recommended Operating Conditions | 11 |
| DC Characteristics | 12 |
| Timings | 13 |
| Memory Interface | 14 |
| Ethernet MAC Interface | 27 |
| Audio Interface | 29 |
| AC'97 | 33 |
| ADC | 34 |
| JTAG | 35 |
| 208 Pin LQFP Package Outline | 36 |
| 208 Pin LQFP Pinout | 37 |
| Acronyms and Abbreviations | 41 |
| Units of Measurement | 41 |
| Ordering Information | 42 |

List of Figures

| | |
|---|----|
| Figure 1. Timing Diagram Drawing Key | 13 |
| Figure 2. SDRAM Load Mode Register Cycle Timing Measurement | 14 |
| Figure 3. SDRAM Burst Read Cycle Timing Measurement | 15 |
| Figure 4. SDRAM Burst Write Cycle Timing Measurement | 16 |
| Figure 5. SDRAM Auto Refresh Cycle Timing Measurement | 17 |
| Figure 6. Static Memory Multiple Word Read 8-bit Cycle Timing Measurement | 18 |
| Figure 7. Static Memory Multiple Word Write 8-bit Cycle Timing Measurement | 19 |
| Figure 8. Static Memory Multiple Word Read 16-bit Cycle Timing Measurement | 20 |
| Figure 9. Static Memory Multiple Word Write 16-bit Cycle Timing Measurement | 21 |
| Figure 10. Static Memory Burst Read Cycle Timing Measurement | 22 |
| Figure 11. Static Memory Burst Write Cycle Timing Measurement | 23 |
| Figure 12. Static Memory Single Read Wait Cycle Timing Measurement | 24 |
| Figure 13. Static Memory Single Write Wait Cycle Timing Measurement | 25 |
| Figure 14. Static Memory Turnaround Cycle Timing Measurement | 26 |
| Figure 15. Ethernet MAC Timing Measurement | 28 |
| Figure 16. <i>T</i> /Single Transfer Timing Measurement | 30 |
| Figure 17. Microwire Frame Format, Single Transfer | 30 |
| Figure 18. SPI Format with SPH=1 Timing Measurement | 31 |
| Figure 19. Inter-IC Sound (I2S) Timing Measurement | 32 |
| Figure 20. AC '97 Configuration Timing Measurement | 33 |
| Figure 21. ADC Transfer Function | 34 |
| Figure 22. JTAG Timing Measurement | 35 |

List of Tables

| | |
|---|----|
| Table A. Change History | 2 |
| Table B. General Purpose Memory Interface Pin Assignments | 6 |
| Table C. Ethernet Media Access Controller Pin Assignments | 7 |
| Table D. Audio Interfaces Pin Assignment | 7 |
| Table E. 12-bit Analog-to-Digital Converter Pin Assignments | 7 |
| Table F. Universal Asynchronous Receiver/Transmitters Pin Assignments | 8 |
| Table G. Dual Port USB Host Pin Assignments | 8 |
| Table H. Two-Wire Port with EEPROM Support Pin Assignments | 8 |
| Table I. Real-Time Clock with Pin Assignments | 8 |
| Table J. PLL and Clocking Pin Assignments | 9 |
| Table K. Interrupt Controller Pin Assignment | 9 |
| Table L. Dual LED Pin Assignments | 9 |
| Table M. General Purpose Input/Output Pin Assignment | 9 |
| Table N. Reset and Power Management Pin Assignments | 10 |
| Table O. Hardware Debug Interface | 10 |
| Table P. Pin List in Numerical Order by Pin Number | 37 |
| Table Q. Pin Descriptions | 39 |
| Table R. Pin Multiplex Usage Information | 40 |

Processor Core - ARM920T

The ARM920T is a Harvard architecture processor with separate 16-kbyte instruction and data caches with an 8-word line length but a unified memory. The processor utilizes a five-stage pipeline consisting of fetch, decode, execute, memory, and write stages. Key features include:

- ARM (32-bit) and Thumb (16-bit compressed) instruction sets
- 32-bit Advanced Micro-Controller Bus Architecture (AMBA)
- 16 kbyte Instruction Cache with lockdown
- 16 kbyte Data Cache (programmable write-through or write-back) with lockdown
- MMU for Linux[®], Microsoft[®] Windows[®] CE and other operating systems
- Translation Look Aside Buffers with 64 Data and 64 Instruction Entries
- Programmable Page Sizes of 1 Mbyte, 64 kbyte, 4 kbyte, and 1 kbyte
- Independent lockdown of TLB Entries

MaverickCrunch[™] Math Engine

The MaverickCrunch Engine is a mixed-mode coprocessor designed primarily to accelerate the math processing required to rapidly encode digital audio formats. It accelerates single and double precision integer and floating point operations plus an integer multiply-accumulate (MAC) instruction that is considerably faster than the ARM920T's native MAC instruction. The ARM920T coprocessor interface is utilized thereby sharing its memory interface and instruction stream. Hardware forwarding and interlock allows the ARM to handle looping and addressing while MaverickCrunch handles computation. Features include:

- IEEE-754 single and double precision floating point
- 32 / 64-bit integer
- Add / multiply / compare
- Integer MAC 32-bit input with 72-bit accumulate
- Integer Shifts
- Floating point to/from integer conversion
- Sixteen 64-bit register files
- Four 72-bit accumulators

MaverickKey[™] Unique ID

MaverickKey unique hardware programmed IDs are a solution to the growing concern over secure web content and commerce. With Internet security playing an important role in the delivery of digital media such as books or music, traditional software methods are quickly becoming unreliable. The MaverickKey unique IDs

provide OEMs with a method of utilizing specific hardware IDs such as those assigned for SDMI (Secure Digital Music Initiative) or any other authentication mechanism.

Both a specific 32-bit ID as well as a 128-bit random ID are programmed into the EP9302 through the use of laser probing technology. These IDs can then be used to match secure copyrighted content with the ID of the target device the EP9302 is powering, and then deliver the copyrighted information over a secure connection. In addition, secure transactions can benefit by also matching device IDs to server IDs. MaverickKey IDs provide a level of hardware security required for today's Internet appliances.

General Purpose Memory Interface (SDRAM, SRAM, ROM, Flash)

The EP9302 features a unified memory address model where all memory devices are accessed over a common address/data bus. Memory accesses are performed via the Processor bus. The SRAM memory controller supports 8 and 16-bit devices and accommodates an internal boot ROM concurrently with 16-bit SDRAM memory.

- 1 to 4 banks of 16-bit ,100-MHz SDRAM
- Address and data bus shared between SDRAM, SRAM, ROM, and FLASH memory
- NOR FLASH memory supported

Table B. General Purpose Memory Interface Pin Assignments

| Pin Mnemonic | Pin Description |
|-----------------------|-----------------------------------|
| SDCLK | SDRAM Clock |
| SDCLKEN | SDRAM Clock Enable |
| SDCSn[3:0] | SDRAM Chip Selects 3-0 |
| RASn | SDRAM RAS |
| CASn | SDRAM CAS |
| SDWEn | SDRAM Write Enable |
| CSn[7:6] and CSn[3:0] | Chip Selects 7, 6, 3, 2, 1, 0 |
| AD[25:0] | Address Bus 25-0 |
| DA[15:0] | Data Bus 15-0 |
| DQMn[1:0] | SDRAM Output Enables / Data Masks |
| WRn | SRAM Write Strobe |
| RDn | SRAM Read / OE Strobe |
| WAITn | SRAM Wait Input |

Ethernet Media Access Controller (MAC)

The MAC subsystem is compliant with the ISO/TEC 802.3 topology for a single shared medium with several stations. Multiple MII-compliant PHYs are supported. Features include:

- Supports 1/10/100 Mbps transfer rates for home / small-business / large-business applications
- Interfaces to an off-chip PHY through industry-standard Media-independent Interface (MII)

Table C. Ethernet Media Access Controller Pin Assignments

| Pin Mnemonic | Pin Description |
|--------------|-----------------------|
| MDC | Management Data Clock |
| MDIO | Management Data I / O |
| RXCLK | Receive Clock |
| MIIRXD[3:0] | Receive Data |
| RXDVAL | Receive Data Valid |
| RXERR | Receive Data Error |
| TXCLK | Transmit Clock |
| MIITXD[3:0] | Transmit Data |
| TXEN | Transmit Enable |
| TXERR | Transmit Error |
| CRS | Carrier Sense |
| CLD | Collision Detect |

Serial Interfaces (SPI, I²S, and AC '97)

The Serial Peripheral Interface (SPI) port can be configured as a master or a slave, supporting the National Semiconductor[®], Motorola[®], and Texas Instruments[®] signaling protocols.

The AC'97 port supports multiple codecs for multichannel audio output with a single stereo input. The I²S port supports stereo 24-bit audio.

These ports are multiplexed so that the I²S port will take over either the AC'97 pins or the SPI pins.

- Normal Mode: One SPI Port and one AC'97 Port
- I²S on SSP Mode: One AC'97 Port and one I²S Port
- I²S on AC'97 Mode: One SPI Port and one I²S Port

Note: I²S may not be output on AC'97 and SSP ports at the same time.

Table D. Audio Interfaces Pin Assignment

| Pin Name | Normal Mode | I ² S on SSP Mode | I ² S on AC'97 Mode |
|----------|---------------------|------------------------------|--------------------------------|
| | Pin Description | Pin Description | Pin Description |
| SCLK1 | SPI Bit Clock | I2S Serial Clock | SPI Bit Clock |
| SFRM1 | SPI Frame Clock | I2S Frame Clock | SPI Frame Clock |
| SSPRX1 | SPI Serial Input | I2S Serial Input | SPI Serial Input |
| SSPTX1 | SPI Serial Output | I2S Serial Output | SPI Serial Output |
| | | (No I2S Master Clock) | |
| ARSTn | AC'97 Reset | AC'97 Reset | I2S Master Clock |
| ABITCLK | AC'97 Bit Clock | AC'97 Bit Clock | I2S Serial Clock |
| ASYN | AC'97 Frame Clock | AC'97 Frame Clock | I2S Frame Clock |
| ASDI | AC'97 Serial Input | AC'97 Serial Input | I2S Serial Input |
| ASDO | AC'97 Serial Output | AC'97 Serial Output | I2S Serial Output |

12-bit Analog-to-digital Converter (ADC)

The ADC block consists of a 12-bit analog-to-digital converter with a analog input multiplexer. The multiplexer can select to measure battery voltage and other miscellaneous voltages on the external measurement pins. Features include:

- 5 external pins for ADC measurement
- Measurement pin input range: 0 to 3.3 V.
- ADC-conversion-complete interrupt signal

Table E. 12-bit Analog-to-Digital Converter Pin Assignments

| Pin Mnemonic | Pin Description |
|-----------------------|-----------------------------------|
| ADC[0] (Ym, pin 135) | External Analog Measurement Input |
| ADC[1] (sXp, pin 134) | External Analog Measurement Input |
| ADC[2] (sXm, pin 133) | External Analog Measurement Input |
| ADC[3] (sYp, pin 132) | External Analog Measurement Input |
| ADC[4] (sYm, pin 131) | External Analog Measurement Input |

Universal Asynchronous Receiver/Transmitters (UARTs)

Two 16550-compatible UARTs are supplied. One provides asynchronous HDLC (High-level Data Link Control) protocol support for full duplex transmit and receive. The HDLC receiver handles framing, address matching, CRC checking, control-octet transparency, and optionally passes the CRC to the host at the end of the packet. The HDLC transmitter handles framing, CRC generation, and control-octet transparency. The host must assemble the frame in memory before transmission. The HDLC receiver and transmitter use the UART FIFOs to buffer the data streams. The second UART provides IrDA[®] compatibility.

- UART1 supports modem bit rates up to 115.2 kbps, supports HDLC and includes a 16 byte FIFO for receive and a 16 byte FIFO for transmit. Interrupts are generated on Rx, Tx and modem status change.
- UART2 contains an IrDA encoder operating at either the slow (up to 115 kbps), medium (0.576 or 1.152 Mbps), or fast (4 Mbps) IR data rates. It also has a 16 byte FIFO for receive and a 16 byte FIFO for transmit.

Table F. Universal Asynchronous Receiver/Transmitters Pin Assignments

| Pin Mnemonic | Pin Name - Description |
|-------------------------------------|--|
| TXD0 | UART1 Transmit |
| RXD0 | UART1 Receive |
| CTS _n | UART1 Clear To Send / Transmit Enable |
| DSR _n / DCD _n | UART1 Data Set Ready / Data Carrier Detect |
| DTR _n | UART1 Data Terminal Ready |
| RTS _n | UART1 Ready To Send |
| EGPIO[0] / RI | UART1 Ring Indicator |
| TXD1 / SIROUT | UART2 Transmit / IrDA Output |
| RXD1 / SIRIN | UART2 Receive / IrDA Input |

Dual-port USB Host

The USB Open Host Controller Interface (Open HCI) provides full-speed serial communications ports at a baud rate of 12 Mbits/sec. Up to 127 USB devices (printer, mouse, camera, keyboard, etc.) and USB hubs can be connected to the USB host in the USB “tiered-star” topology.

This includes the following feature:

- Compliance with the USB 2.0 specification
- Compliance with the Open HCI Rev 1.0 specification

- Supports both low-speed (1.5 Mbps) and full-speed (12 Mbps) USB device connections
- Root HUB integrated with 2 downstream USB ports
- Transceiver buffers integrated, over-current protection on ports
- Supports power management
- Operates as a master on the bus

The Open HCI host controller initializes the master DMA transfer with the AHB bus:

- Fetches endpoint descriptors and transfer descriptors
- Accesses endpoint data from system memory
- Accesses the HC communication area
- Writes status and retire transfer descriptor

Table G. Dual Port USB Host Pin Assignments

| Pin Mnemonic | Pin Name - Description |
|--------------|------------------------|
| USBp[2,0] | USB Positive signals |
| USBm[2,0] | USB Negative Signals |

Note: USBm[1] and USBp[1] are not bonded out.

Two-wire Interface

The two-wire interface provides communication and control for synchronous-serial-driven devices.

Table H. Two-Wire Port with EEPROM Support Pin Assignments

| Pin Mnemonic | Pin Name - Description | Alternative Usage |
|--------------|--------------------------|---------------------|
| EECLK | Two-wire Interface Clock | General Purpose I/O |
| EEDATA | Two-Wire Interface Data | General Purpose I/O |

Real-Time Clock with Software Trim

The software trim feature on the real time clock (RTC) provides software controlled digital compensation of the 32.768 KHz input clock. This compensation is accurate to ± 1.24 sec/month.

Note: A real time clock must be connected to RTCXTALI or the EP9302 device will not boot.

Table I. Real-Time Clock with Pin Assignments

| Pin Mnemonic | Pin Name - Description |
|--------------|-----------------------------------|
| RTCXTALI | Real-Time Clock Oscillator Input |
| RTCXTALO | Real-Time Clock Oscillator Output |

PLL and Clocking

The Processor and the Peripheral Clocks operate from a single 14.7456 MHz crystal.

The Real Time Clock operates from a 32.768 KHz external oscillator.

Table J. PLL and Clocking Pin Assignments

| Pin Mnemonic | Pin Name - Description |
|--------------|------------------------|
| XTALI | Main Oscillator Input |
| XTALO | Main Oscillator Output |
| VDD_PLL | Main Oscillator Power |
| GND_PLL | Main Oscillator Ground |

Timers

The Watchdog Timer ensures proper operation by requiring periodic attention to prevent a reset-on-time-out.

Two 16-bit timers operate as free running down-counters or as periodic timers for fixed interval interrupts and have a range of 0.03 ms to 4.27 seconds.

One 32-bit timer, plus a 6-bit prescale counter, has a range of 0.03 μ s to 73.3 hours.

One 40-bit debug timer, plus 6-bit prescale counter, has a range of 1.0 μ s to 12.7 days.

Interrupt Controller

The interrupt controller allows up to 54 interrupts to generate an Interrupt Request (IRQ) or Fast Interrupt Request (FIQ) signal to the processor core. Thirty-two hardware priority assignments are provided for assisting IRQ vectoring, and two levels are provided for FIQ vectoring. This allows time critical interrupts to be processed in the shortest time possible. Internal interrupts may be programmed as active high or active low level sensitive inputs. GPIO pins programmed as interrupts may be programmed as active high level sensitive, active low level sensitive, rising edge triggered, falling edge triggered, or combined rising/falling edge triggered.

- Supports 54 interrupts from a variety of sources (such as UARTs, GPIO and ADC)
- Routes interrupt sources to either the ARM920T's IRQ or FIQ (Fast IRQ) inputs
- Three dedicated off-chip interrupt lines operate as active high level sensitive interrupts
- Any of the 19 GPIO lines maybe configured to generate interrupts

- Software supported priority mask for all FIQs and IRQs

Table K. External Interrupt Controller Pin Assignment

| Pin Mnemonic | Pin Name - Description |
|---------------------|-----------------------------|
| INT[3] and INT[1:0] | External Interrupts 2, 1, 0 |

Note: INT[2] is not bonded out.

Dual LED Drivers

Two pins are assigned specifically to drive external LEDs.

Table L. Dual LED Pin Assignments

| Pin Mnemonic | Pin Name - Description | Alternative Usage |
|--------------|------------------------|---------------------|
| GRLED | Green LED | General Purpose I/O |
| REDLED | Red LED | General Purpose I/O |

General Purpose Input/Output (GPIO)

The 16 EGPIO and the 3 FGPIO pins may each be configured individually as an output, an input, or an interrupt input.

There are 10 pins that may alternatively be used as input, output, or open-drain pins, but do not support interrupts. These pins are:

- Ethernet MDIO
- Both LED Outputs
- EEPROM Clock and Data
- HGPIO[5:2]
- CGPIO[0]

6 pins may alternatively be used as inputs only:

- CTS_n, DSR_n / DCD_n
- 3 Interrupt Lines

2 pins may alternatively be used as outputs only:

- RTS_n
- ARST_n

Table M. General Purpose Input/Output Pin Assignment

| Pin Mnemonic | Pin Name - Description |
|--------------|--|
| EGPIO[15:0] | Expanded General Purpose Input / Output Pins with Interrupts |
| FGPIO[3:1] | Expanded General Purpose Input / Output Pins with Interrupts |

Reset and Power Management

The chip may be reset through the PRSTn pin or through the open drain common reset pin, RSTOn.

Clocks are managed on a peripheral-by-peripheral basis and may be turned off to conserve power.

The processor clock is dynamically adjustable from 0 to 200 MHz (184 MHz for industrial conditions).

Table N. Reset and Power Management Pin Assignments

| Pin Mnemonic | Pin Name - Description |
|--------------|--|
| PRSTn | Power On Reset |
| RSTOn | User Reset In/Out – Open Drain – Preserves Real Time Clock value |

Hardware Debug Interface

The JTAG interface allows use of ARM's Multi-ICE or other in-circuit emulators.

Table O. Hardware Debug Interface

| Pin Mnemonic | Pin Name - Description |
|--------------|------------------------|
| TCK | JTAG Clock |
| TDI | JTAG Data In |
| TDO | JTAG Data Out |
| TMS | JTAG Test Mode Select |
| TRSTn | JTAG Port Reset |

12-Channel DMA Controller

The DMA module contains 12 separate DMA channels. Ten of these may be used for peripheral-to-memory or memory-to-peripheral access. Two of these are dedicated to memory-to-memory transfers. Each DMA channel is connected to the 16-bit DMA request bus.

The request bus is a collection of requests, Serial Audio and UARTs. Each DMA channel can be used independently or dedicated to any request signal. For each DMA channel, source and destination addressing can be independently programmed to increment, decrement, or stay at the same value. All DMA addresses are physical, not virtual addresses.

Internal Boot ROM

The Internal 16-kbyte ROM allows booting from FLASH memory, SPI or UART. Consult the EP9301 User's Guide for operational details.

Electrical Specifications

Absolute Maximum Ratings

(All grounds = 0 V, all voltages with respect to 0 V)

| Parameter | Symbol | Min | Max | Unit |
|--|---------|------|----------|------|
| Power Supplies | RVDD | - | 3.96 | V |
| | CVDD | - | 2.16 | V |
| | VDD_PLL | - | 2.16 | V |
| | VDD_ADC | - | 3.96 | V |
| Total Power Dissipation (Note 1) | | - | 2 | W |
| Input Current per Pin, DC (Except supply pins) | | - | ±10 | mA |
| Output current per pin, DC | | - | ±50 | mA |
| Digital Input voltage (Note 2) | | -0.3 | RVDD+0.3 | V |
| Storage temperature | | -40 | +125 | °C |

- Note: 1. Includes all power generated by AC and/or DC output loading.
 2. The power supply pins are at maximum values listed in "Recommended Operating Conditions", below.
 3. At ambient temperatures above 70° C, total power dissipation must be limited to less than 2.5 Watts.

WARNING: Operation beyond these limits may result in permanent damage to the device.
 Normal operation is not guaranteed at these extremes.

Recommended Operating Conditions

(All grounds = 0 V, all voltages with respect to 0 V)

| Parameter | Symbol | Min | Typ | Max | Unit |
|--|----------------|------|------|------|------|
| Power Supplies | RVDD | 3.0 | 3.3 | 3.6 | V |
| | CVDD | 1.65 | 1.80 | 1.94 | V |
| | VDD_PLL | 1.65 | 1.80 | 1.94 | V |
| | VDD_ADC | 3.0 | 3.3 | 3.6 | V |
| Operating Ambient Temperature - Commercial | T _A | 0 | +25 | +70 | °C |
| Operating Ambient Temperature - Industrial | T _A | -40 | +25 | +85 | °C |
| Processor Clock Speed - Commercial | FCLK | - | - | 200 | MHz |
| Processor Clock Speed - Industrial | FCLK | - | - | 184 | MHz |
| System Clock Speed - Commercial | HCLK | - | - | 100 | MHz |
| System Clock Speed - Industrial | HCLK | - | - | 92 | MHz |

DC Characteristics

($T_A = 0$ to 70°C ; $CVDD = VDD_PLL = 1.8$; $RVDD = 3.3\text{ V}$;
All grounds = 0 V ; all voltages with respect to 0 V unless otherwise noted)

| Parameter | | | Symbol | Min | Max | Unit |
|----------------------------|--------------------------|----------|----------|--------------------|--------------------|---------------|
| High level output voltage | $I_{out} = -4\text{ mA}$ | (Note 4) | V_{oh} | $0.85 \times RVDD$ | - | V |
| Low level output voltage | $I_{out} = 4\text{ mA}$ | | V_{ol} | - | $0.15 \times RVDD$ | V |
| High level input voltage | | (Note 5) | V_{ih} | $0.65 \times RVDD$ | $VDD + 0.3$ | V |
| Low level input voltage | | (Note 5) | V_{il} | -0.3 | $0.35 \times RVDD$ | V |
| High level leakage current | $V_{in} = 3.3\text{ V}$ | (Note 5) | I_{ih} | - | 10 | μA |
| Low level leakage current | $V_{in} = 0$ | (Note 5) | I_{il} | - | -10 | μA |

| Parameter | | Min | Typ | Max | Unit |
|--------------------------------------|----------------------|-----|-----|-----|------|
| Power Supply Pins (Outputs Unloaded) | | | | | |
| Power Supply Current: | CVDD / VDD_PLL Total | - | 180 | 230 | mA |
| | RVDD | - | 45 | 80 | mA |
| Low-Power Mode Supply Current | CVDD / VDD_PLL Total | - | 2 | 3.5 | mA |
| | RVDD | - | 1.0 | 2 | mA |

Note: 4. For open drain pins, high level output voltage is dependent on the external load.

5. All inputs that do not include internal pull-ups or pull-downs, must be externally driven for proper operation (See [Table Q on page 39](#)). If an input is not driven, it should be tied to power or ground, depending on the particular function. If an I/O pin is not driven and programmed as an input, it should be tied to power or ground through its own resistor.

Timings

Timing Diagram Conventions

This data sheet contains one or more timing diagrams. The following key explains the components used in these diagrams. Any variations are clearly labelled when they occur. Therefore, no additional meaning should be attached unless specifically stated.

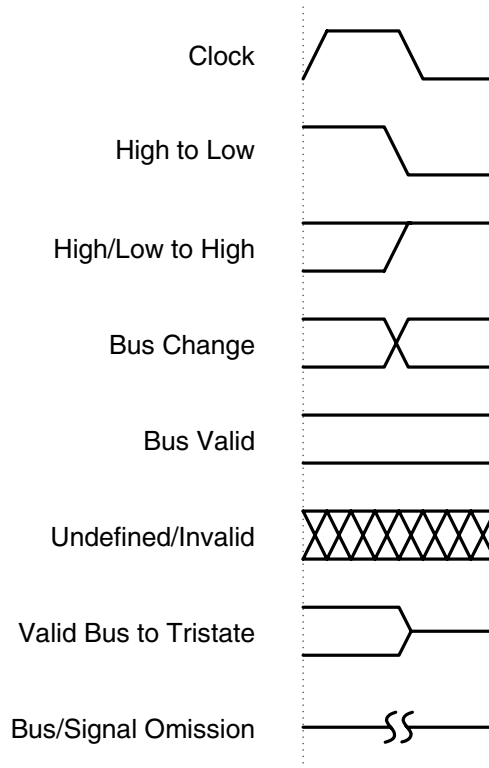


Figure 1. Timing Diagram Drawing Key

Timing Conditions

Unless specified otherwise, the following conditions are true for all timing measurements.

- $T_A = 0$ to 70°C
- $CVDD = VDD_PLL = 1.8\text{V}$
- $RVDD = 3.3\text{V}$
- All grounds = 0V
- Logic 0 = 0V , Logic 1 = 3.3V
- Output loading = 50pF
- Timing reference levels = 1.5V
- The Processor Bus Clock (HCLK) is programmable and is set by the user. The frequency is typically between 33MHz and 100MHz (92MHz for industrial conditions).

Memory Interface

Figure 2 through Figure 5 define the timings associated with all phases of the SDRAM. The following table contains the values for the timings of each of the SDRAM modes.

| Parameter | Symbol | Min | Typ | Max | Unit |
|---|-----------------|-----|------------------|-----|------|
| SDCLK high time | t_{clk_high} | - | $(t_{HCLK}) / 2$ | - | ns |
| SDCLK low time | t_{clk_low} | - | $(t_{HCLK}) / 2$ | - | ns |
| SDCLK rise/fall time | t_{clkrf} | - | 2 | 4 | ns |
| Signal delay from SDCLK rising edge time | t_d | - | - | 8 | ns |
| Signal hold from SDCLK rising edge time | t_h | 1 | - | - | ns |
| DQMn delay from SDCLK rising edge time | t_{DQd} | - | - | 8 | ns |
| DQMn hold from SDCLK rising edge time | t_{DQh} | 1 | - | - | ns |
| DA valid setup to SDCLK rising edge time | t_{DA_s} | 2 | - | - | ns |
| DA valid hold from SDCLK rising edge time | t_{DA_h} | 3 | - | - | ns |

SDRAM Load Mode Register Cycle

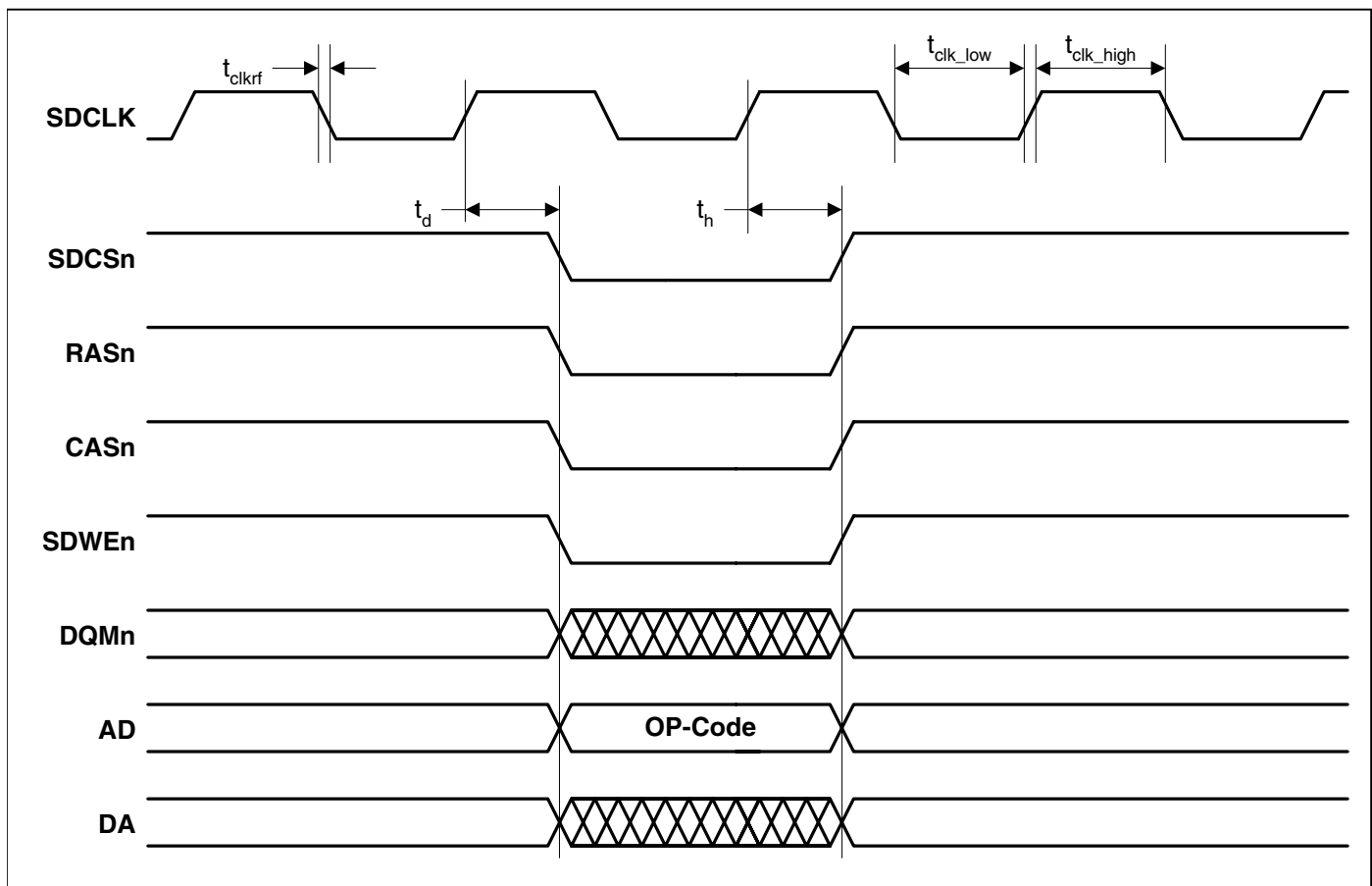
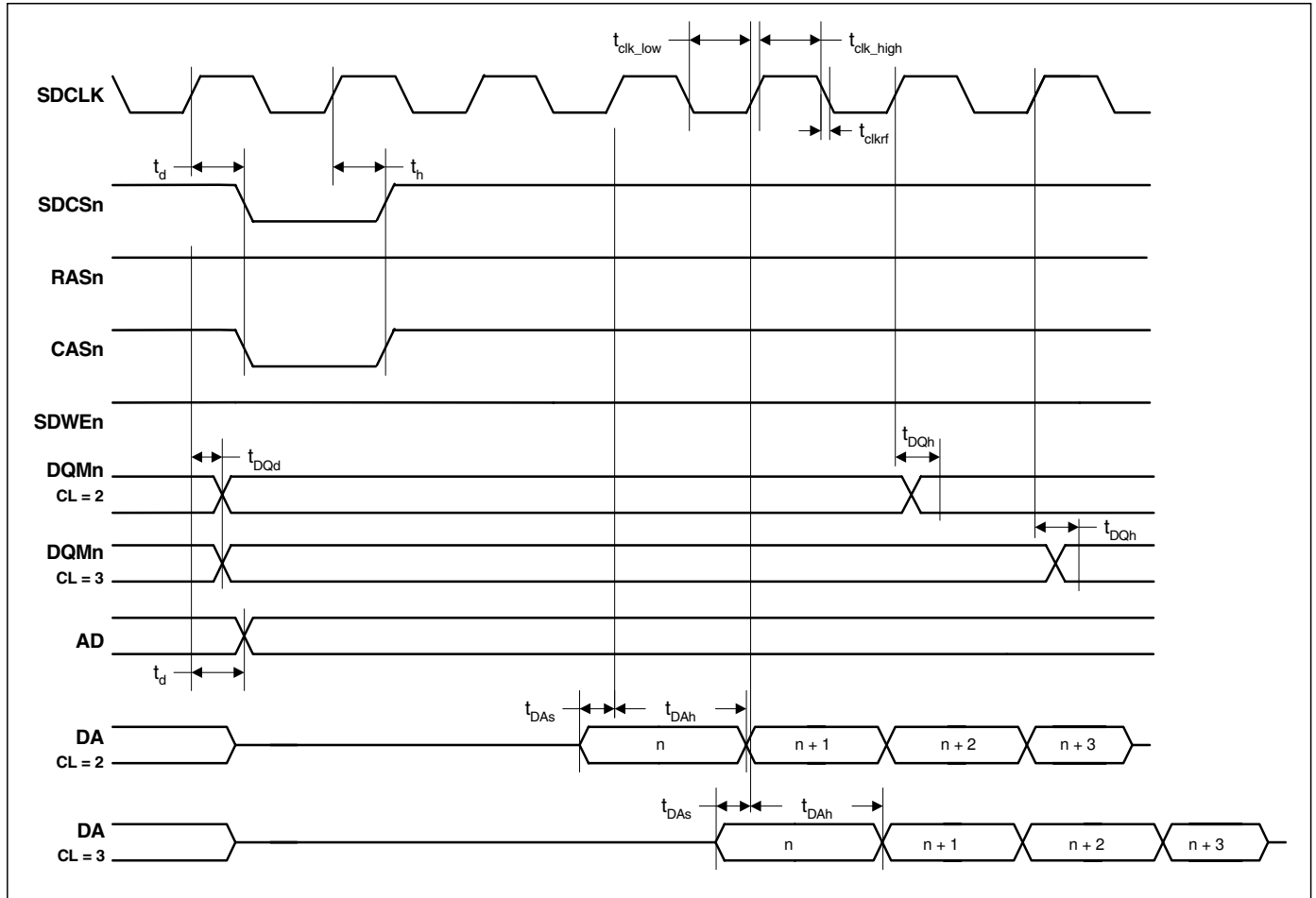


Figure 2. SDRAM Load Mode Register Cycle Timing Measurement

SDRAM Burst Read Cycle

Figure 3. SDRAM Burst Read Cycle Timing Measurement

SDRAM Burst Write Cycle

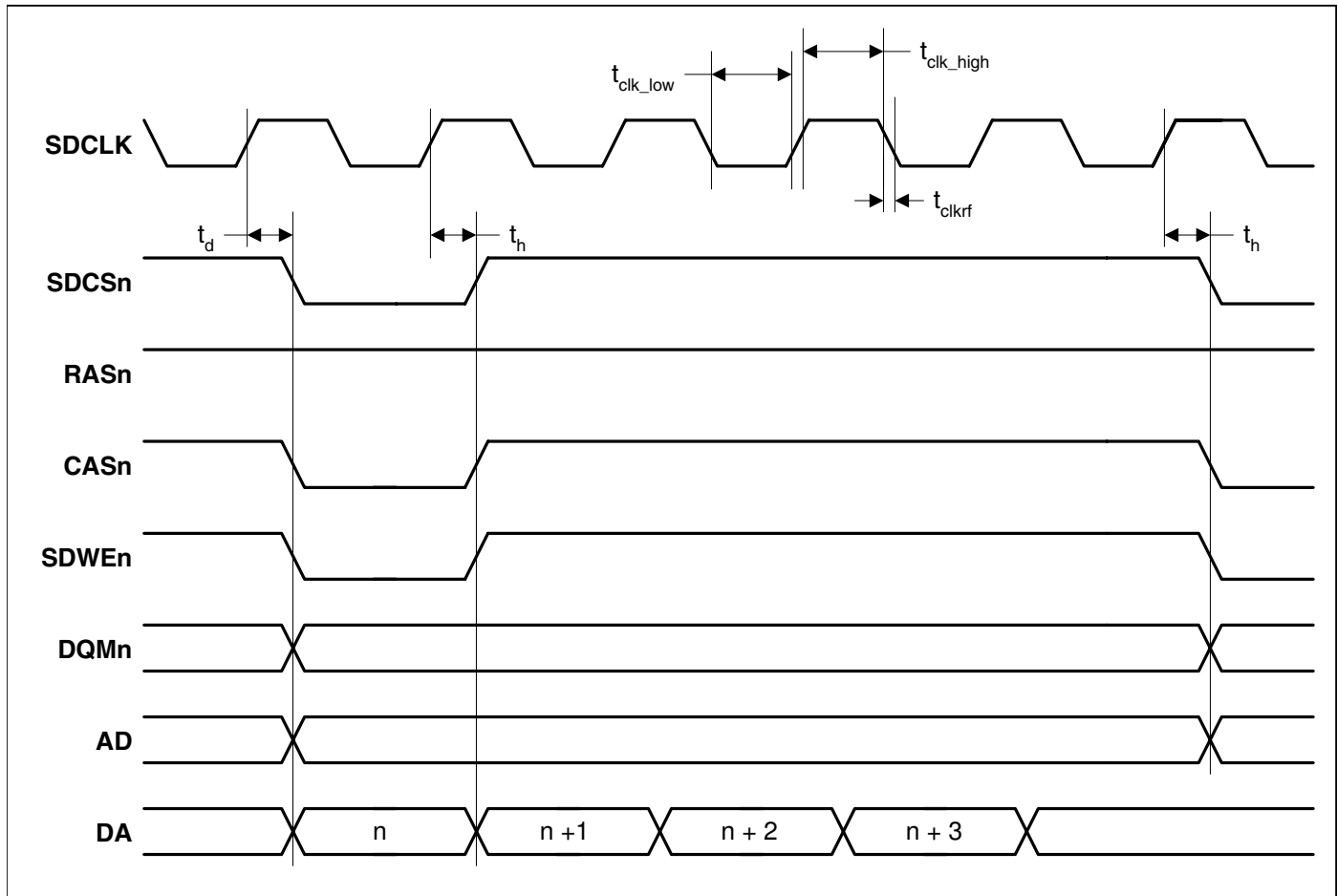
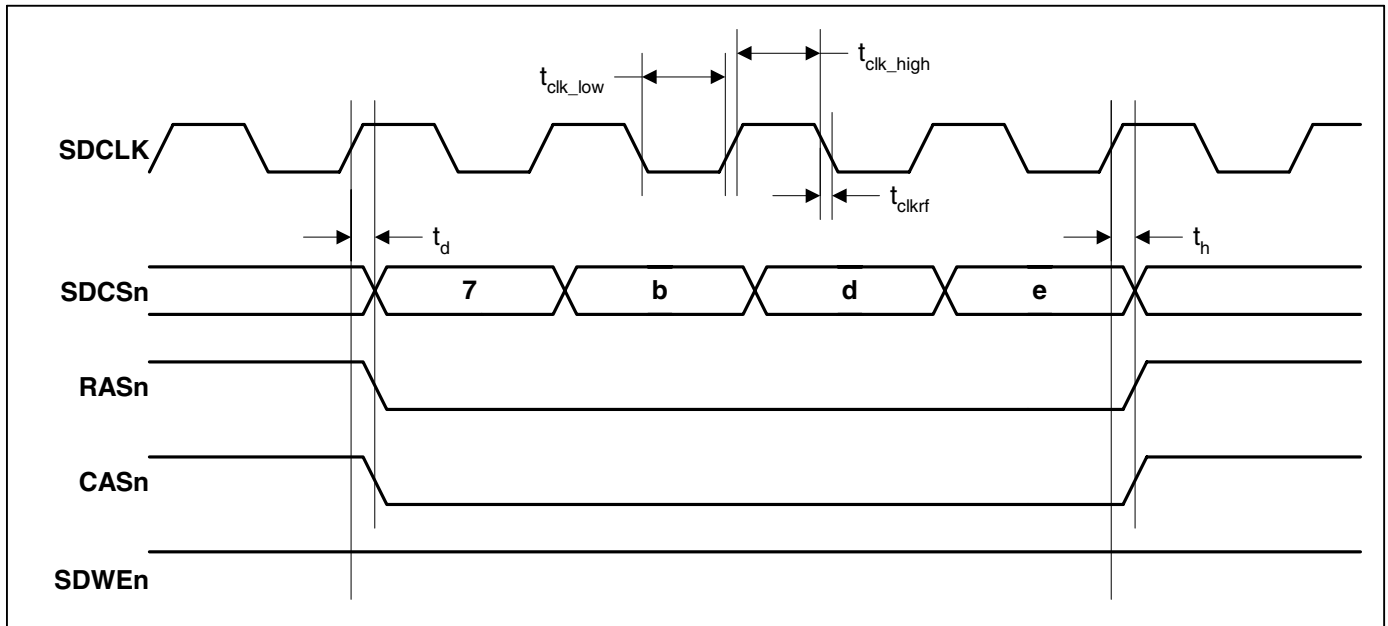


Figure 4. SDRAM Burst Write Cycle Timing Measurement

SDRAM Auto Refresh Cycle


Note: Chip select shown as bus to illustrate multiple devices being put into auto refresh in one access

Figure 5. SDRAM Auto Refresh Cycle Timing Measurement

Static Memory 32-bit Read on 8-bit External Bus

| Parameter | Symbol | Min | Typ | Max | Unit |
|---------------------------------------|-------------|-----------------|---------------------------------------|-----|------|
| AD setup to CSn assert time | t_{ADs} | t_{HCLK} | - | - | ns |
| CSn assert to Address transition time | t_{AD1} | - | $t_{HCLK} \times (WST1 + 1)$ | - | ns |
| Address assert time | t_{AD2} | - | $t_{HCLK} \times (WST1 + 1)$ | - | ns |
| AD transition to CSn deassert time | t_{AD3} | - | $t_{HCLK} \times (WST1 + 2)$ | - | ns |
| AD hold from CSn deassert time | t_{ADh} | t_{HCLK} | - | - | ns |
| RDn assert time | t_{RDpwL} | - | $t_{HCLK} \times (4 \times WST1 + 5)$ | - | ns |
| CSn to RDn delay time | t_{RDd} | - | - | 3 | ns |
| CSn assert to DQMn assert delay time | t_{DQMd} | - | - | 1 | ns |
| DA setup to AD transition time | t_{DAs1} | 15 | - | - | ns |
| DA setup to RDn deassert time | t_{DAs2} | $t_{HCLK} + 12$ | - | - | ns |
| DA hold from AD transition time | t_{DAh1} | 0 | - | - | ns |
| DA hold from RDn deassert time | t_{DAh2} | 0 | - | - | ns |

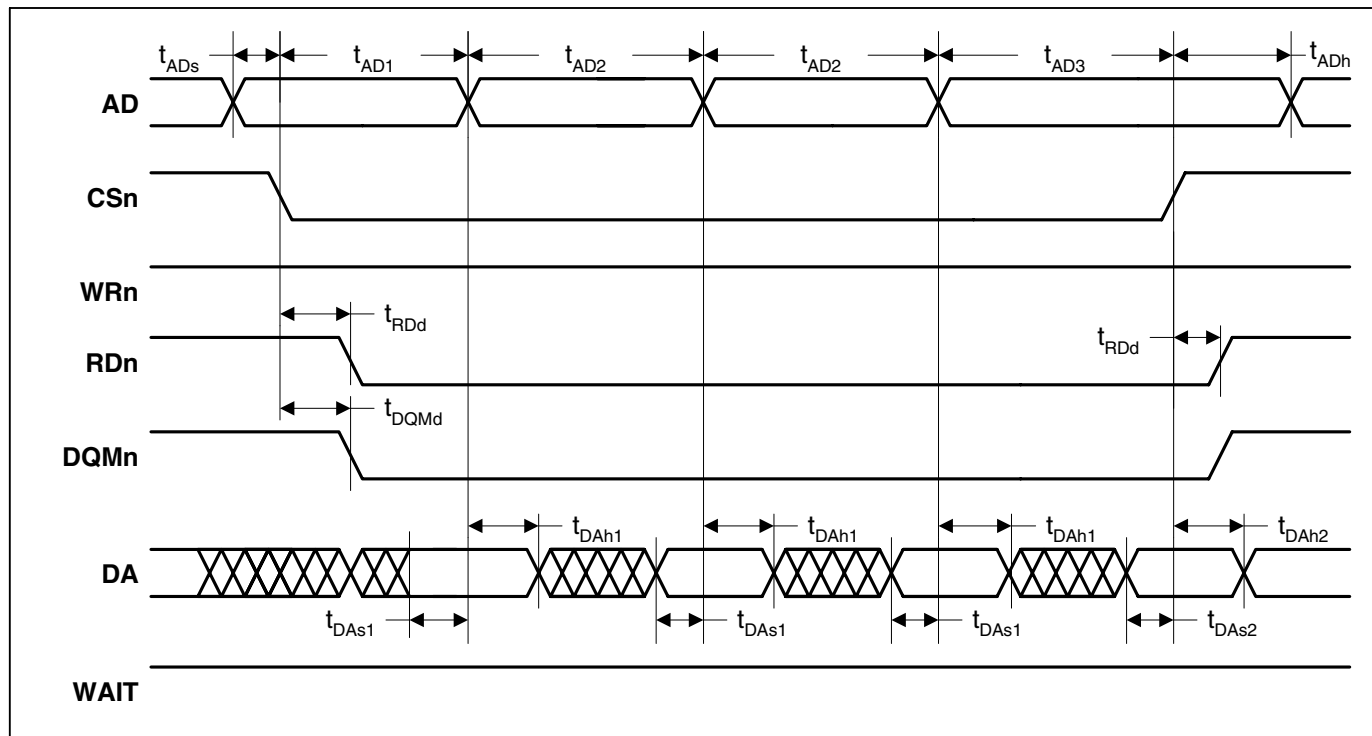
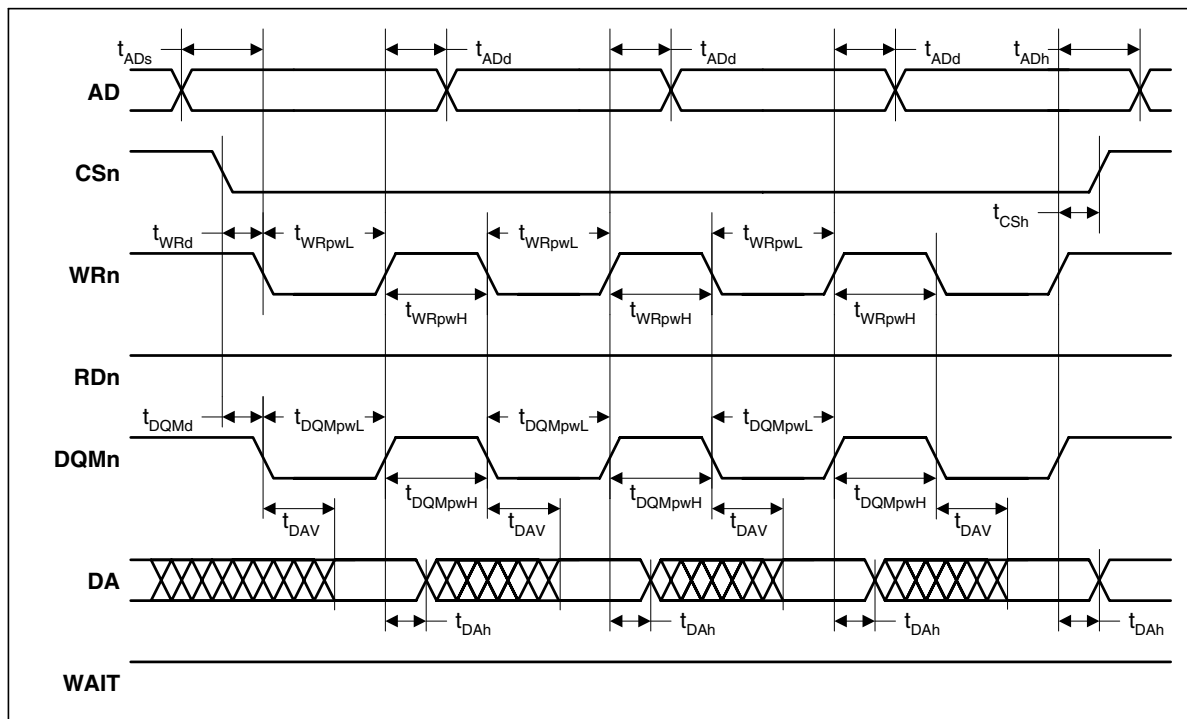


Figure 6. Static Memory Multiple Word Read 8-bit Cycle Timing Measurement

Static Memory 32-bit Write on 8-bit External Bus

| Parameter | Symbol | Min | Typ | Max | Unit |
|---|--------------|---------------------|------------------------------|----------------------------|------|
| AD setup to WRn assert time | t_{ADs} | $t_{HCLK} - 3$ | - | - | ns |
| WRn/DQMn deassert to AD transition time | t_{ADd} | - | - | $t_{HCLK} + 6$ | ns |
| AD hold from WRn deassert time | t_{ADh} | $t_{HCLK} \times 2$ | - | - | ns |
| CSn hold from WRn deassert time | t_{CSH} | 7 | - | - | ns |
| CSn to WRn assert delay time | t_{WRd} | - | - | 2 | ns |
| WRn assert time | t_{WRpwL} | - | $t_{HCLK} \times (WST1 + 1)$ | - | ns |
| WRn deassert time | t_{WRpwH} | - | $t_{HCLK} \times 2$ | $(t_{HCLK} \times 2) + 14$ | ns |
| CSn to DQMn assert delay time | t_{DQMd} | - | - | 1 | ns |
| DQMn assert time | t_{DQMpwL} | - | $t_{HCLK} \times (WST1 + 1)$ | - | ns |
| DQMn deassert time | t_{DQMpwH} | - | - | $(t_{HCLK} \times 2) + 7$ | ns |
| WRn / DQMn deassert to DA transition time | t_{DAh} | t_{HCLK} | - | - | ns |
| WRn / DQMn assert to DA valid time | t_{DAV} | - | - | 8 | ns |


Figure 7. Static Memory Multiple Word Write 8-bit Cycle Timing Measurement

Static Memory 32-bit Read on 16-bit External Bus

| Parameter | Symbol | Min | Typ | Max | Unit |
|--------------------------------------|-------------|-----------------|---|-----|------|
| AD setup to CSn assert time | t_{ADs} | t_{HCLK} | - | - | ns |
| CSn assert to AD transition time | t_{ADd1} | - | $t_{HCLK} \times (WST1 + 1)$ | - | ns |
| AD transition to CSn deassert time | t_{ADd2} | - | $t_{HCLK} \times (WST1 + 2)$ | - | ns |
| AD hold from CSn deassert time | t_{ADh} | t_{HCLK} | - | - | ns |
| RDn assert time | t_{RDpwL} | - | $t_{HCLK} \times ((2 \times WST1) + 3)$ | - | ns |
| CSn to RDn delay time | t_{RDd} | - | - | 3 | ns |
| CSn assert to DQMn assert delay time | t_{DQMd} | - | - | 1 | ns |
| DA setup to AD transition time | t_{DAs1} | 15 | - | - | ns |
| DA to RDn deassert time | t_{DAh1} | $t_{HCLK} + 12$ | - | - | ns |
| DA hold from AD transition time | t_{DAh2} | 0 | - | - | ns |
| DA hold from RDn deassert time | t_{DAh2} | 0 | - | - | ns |

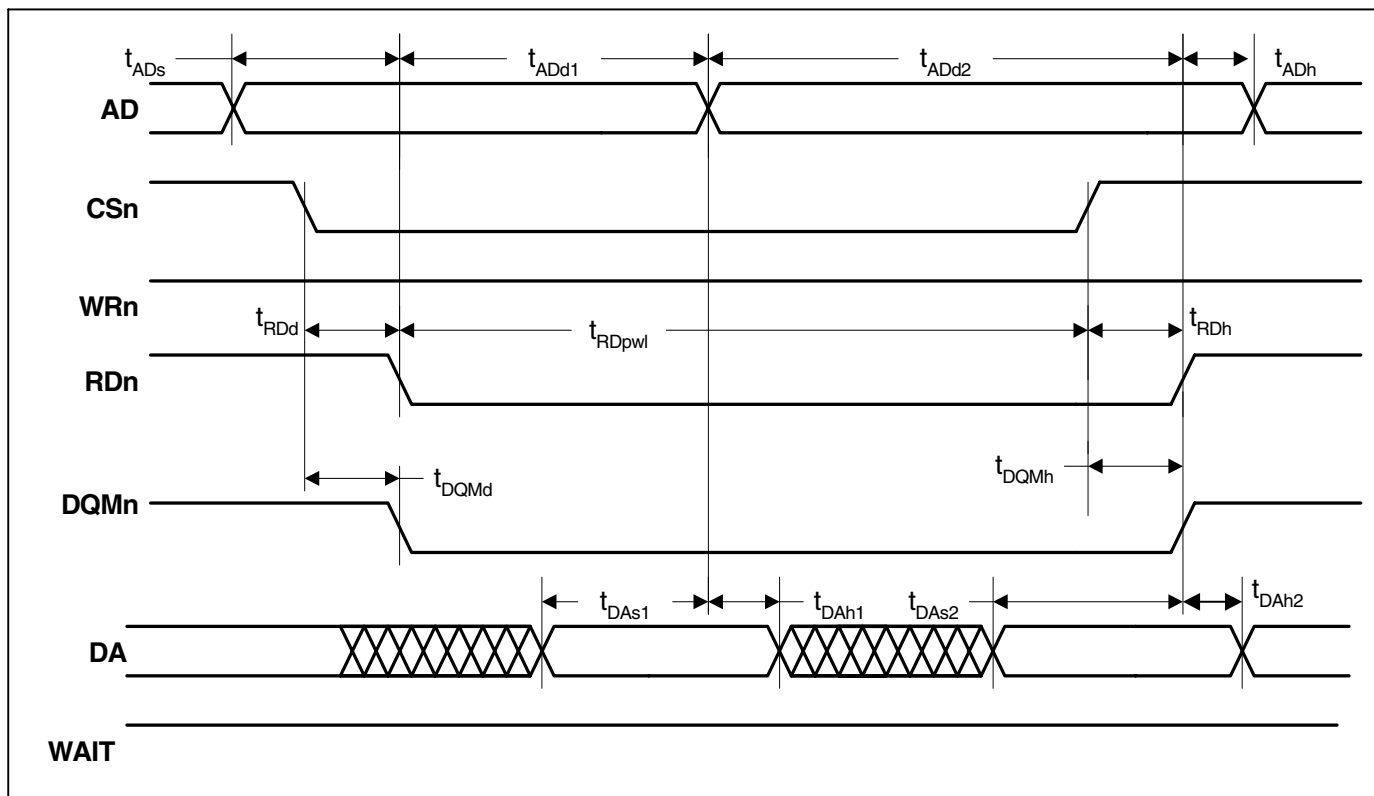
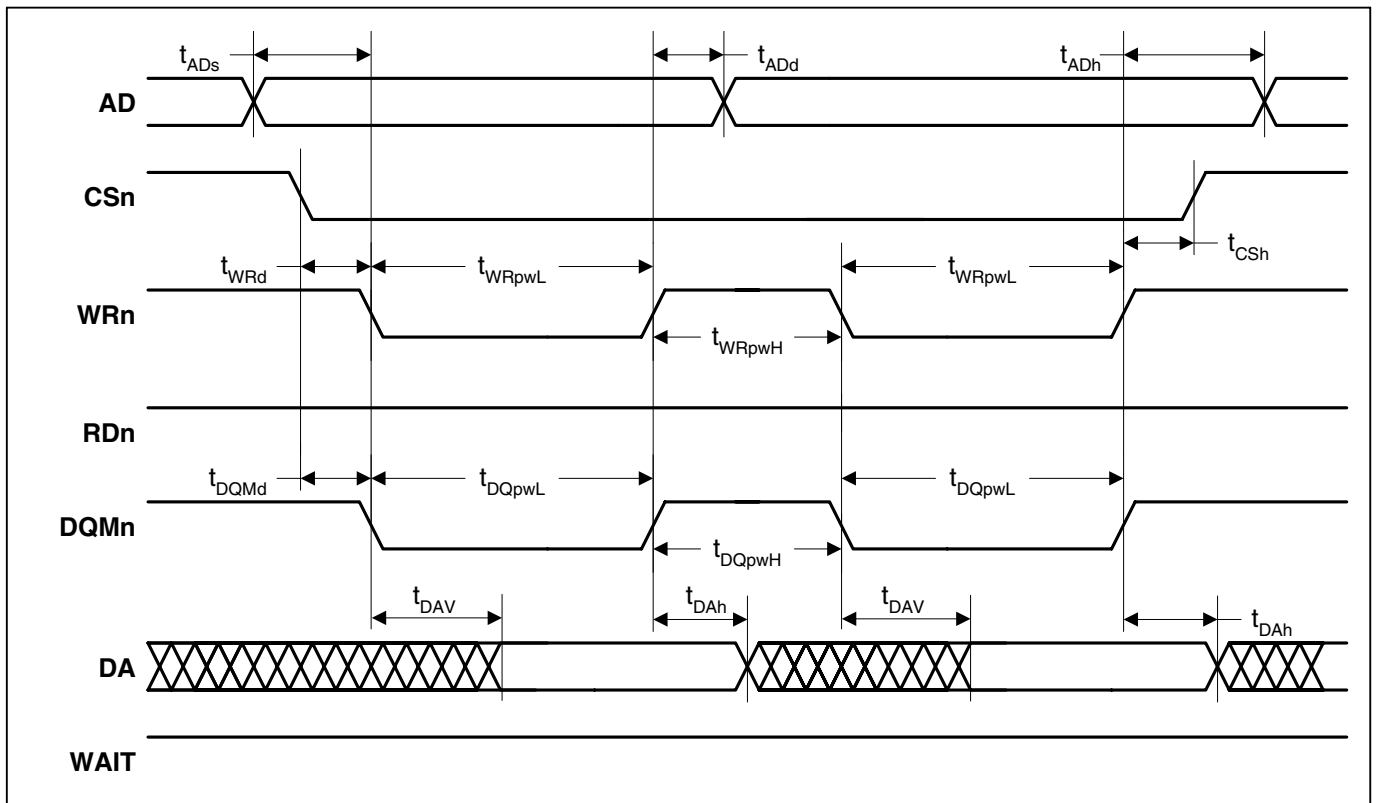


Figure 8. Static Memory Multiple Word Read 16-bit Cycle Timing Measurement

Static Memory 32-bit Write on 16-bit External Bus

| Parameter | Symbol | Min | Typ | Max | Unit |
|---|--------------|---------------------|------------------------------|----------------------------|------|
| AD setup to WRn assert time | t_{ADs} | $t_{HCLK} - 3$ | - | - | ns |
| WRn/DQMn deassert to AD transition time | t_{ADd} | - | - | $t_{HCLK} + 6$ | ns |
| AD hold from WRn deassert time | t_{ADh} | $t_{HCLK} \times 2$ | - | - | ns |
| CSn hold from WRn deassert time | t_{CSH} | 7 | - | - | ns |
| CSn to WRn assert delay time | t_{WRd} | - | - | 2 | ns |
| WRn assert time | t_{WRpwL} | - | $t_{HCLK} \times (WST1 + 1)$ | - | ns |
| WRn deassert time | t_{WRpwH} | - | - | $(t_{HCLK} \times 2) + 14$ | ns |
| CSn to DQMn assert delay time | t_{DQMd} | - | - | 1 | ns |
| DQMn assert time | t_{DQMpwL} | - | $t_{HCLK} \times (WST1 + 1)$ | - | ns |
| DQMn deassert time | t_{DQMpwH} | - | - | $(t_{HCLK} \times 2) + 7$ | ns |
| WRn / DQMn deassert to DA transition time | t_{DAh1} | t_{HCLK} | - | - | ns |
| WRn / DQMn assert to DA valid time | t_{DAV} | - | - | 8 | ns |


Figure 9. Static Memory Multiple Word Write 16-bit Cycle Timing Measurement

Static Memory Burst Read Cycle

| Parameter | Symbol | Min | Typ | Max | Unit |
|---|------------|-----------------|------------------------------|-----|------|
| CSn assert to Address 1 transition time | t_{ADd1} | - | $t_{HCLK} \times (WST1 + 1)$ | - | ns |
| Address assert time | t_{ADd2} | - | $t_{HCLK} \times (WST2 + 1)$ | - | ns |
| AD transition to CSn deassert time | t_{ADd3} | - | $t_{HCLK} \times (WST1 + 2)$ | - | ns |
| AD hold from CSn deassert time | t_{ADh} | t_{HCLK} | - | - | ns |
| CSn to RDn delay time | t_{RDd} | - | - | 3 | ns |
| CSn to DQMn assert delay time | t_{DQMd} | - | - | 1 | ns |
| DA setup to AD transition time | t_{DAs1} | 15 | - | - | ns |
| DA setup to CSn deassert time | t_{DAs2} | $t_{HCLK} + 12$ | - | - | ns |
| DA hold from AD transition time | t_{DAh1} | 0 | - | - | ns |
| DA hold from RDn deassert time | t_{DAh2} | 0 | - | - | ns |

Note: These characteristics are valid when the Page Mode Enable (Burst Mode) bit is set. See the User's Guide for details.

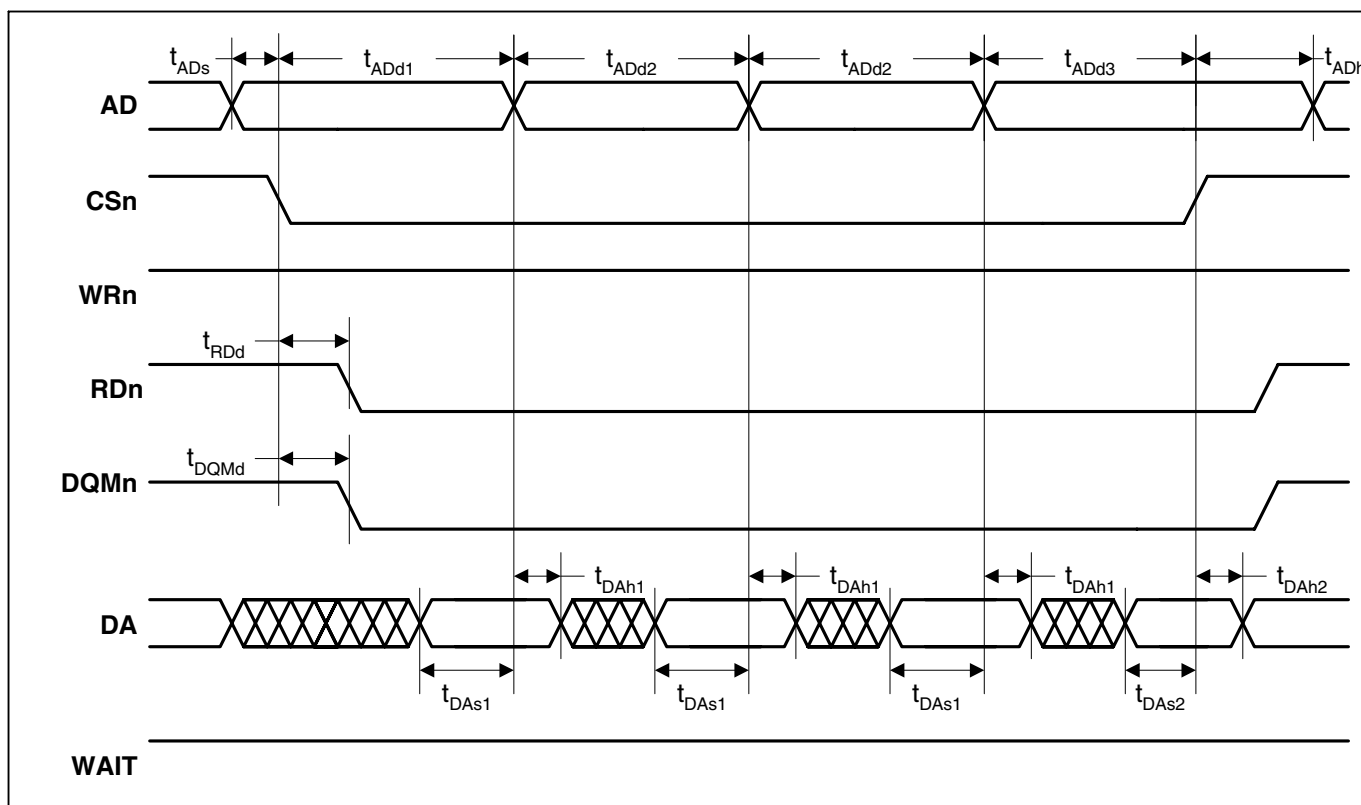


Figure 10. Static Memory Burst Read Cycle Timing Measurement

Static Memory Burst Write Cycle

| Parameter | Symbol | Min | Typ | Max | Unit |
|---|-------------|---------------------|-------------------------------|----------------------------|------|
| AD setup to WRn assert time | t_{ADs} | $t_{HCLK} - 3$ | | | ns |
| AD hold from WRn deassert time | t_{ADh} | $t_{HCLK} \times 2$ | | | ns |
| WRN/DQMn deassert to AD transition time | t_{ADd} | | | $t_{HCLK} + 6$ | ns |
| CSn hold from WRn deassert time | t_{CSH} | 7 | | | ns |
| CSn to WRn assert delay time | t_{WRd} | | | 2 | ns |
| CSn to DQMn assert delay time | t_{DQMd} | | | 1 | ns |
| DQMn assert time | t_{DQpWL} | | $t_{HCLK} \times (WST1 + 1)$ | | ns |
| DQMn deassert time | t_{DQpWH} | | | $(t_{HCLK} \times 2) + 14$ | ns |
| WRn assert time | t_{WRpWL} | | $t_{HCLK} \times (WST1 + 11)$ | | ns |
| WRn deassert time | t_{WRpWH} | | | $(t_{HCLK} \times 2) + 7$ | ns |
| WRn/DQMn deassert to DA transition time | t_{DAh} | t_{HCLK} | | | ns |
| WRn/DQMn assert to DA valid time | t_{DAv} | | | 8 | ns |

Note: These characteristics are valid when the Page Mode Enable (Burst Mode) bit is set. See the User's Guide for details.

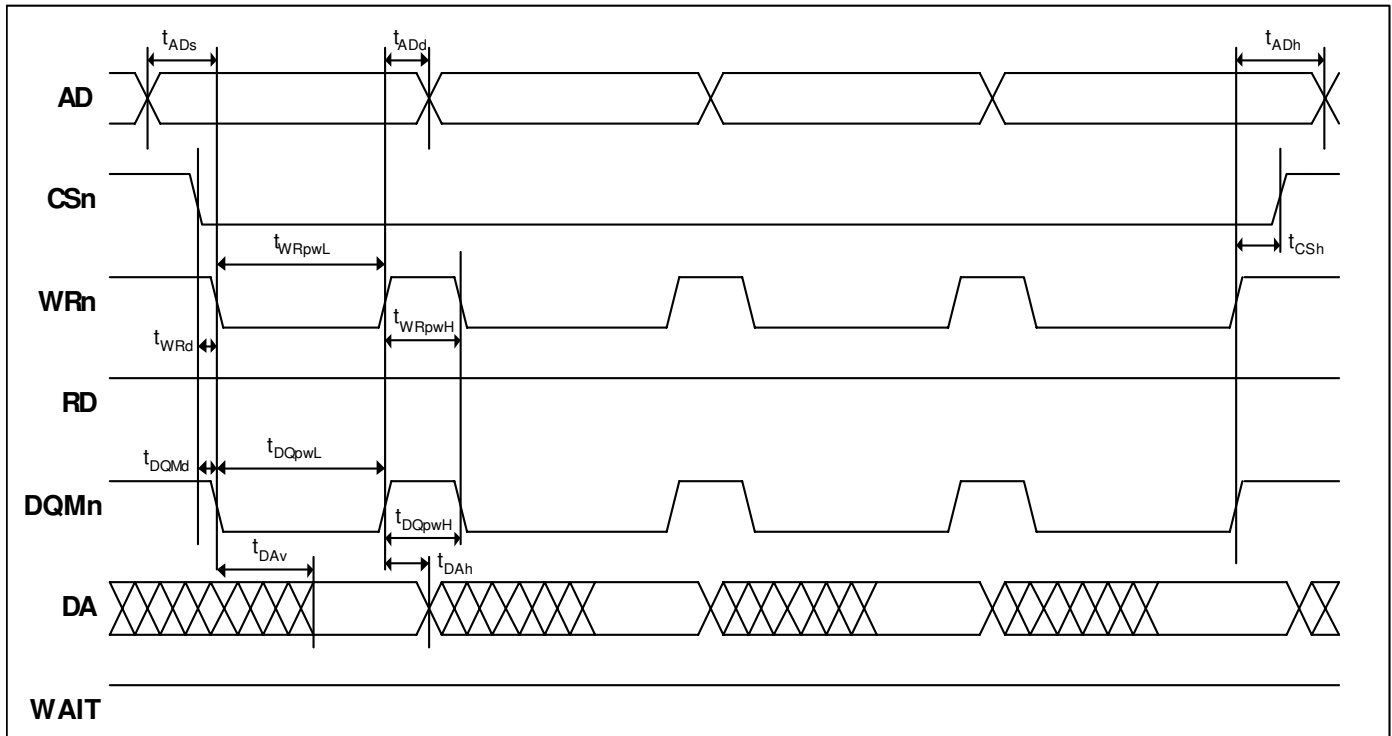


Figure 11. Static Memory Burst Write Cycle Timing Measurement

Static Memory Single Read Wait Cycle

| Parameter | Symbol | Min | Typ | Max | Unit |
|---------------------------------|--------------|---------------------|-----|----------------------------|------|
| CSn assert to WAIT time | t_{WAITd} | - | - | $t_{HCLK} \times (WST1-2)$ | ns |
| WAIT assert time | t_{WAITpw} | $t_{HCLK} \times 2$ | - | $t_{HCLK} \times 510$ | ns |
| WAIT to CSn deassert delay time | t_{CSnd} | $t_{HCLK} \times 3$ | - | $t_{HCLK} \times 5$ | ns |

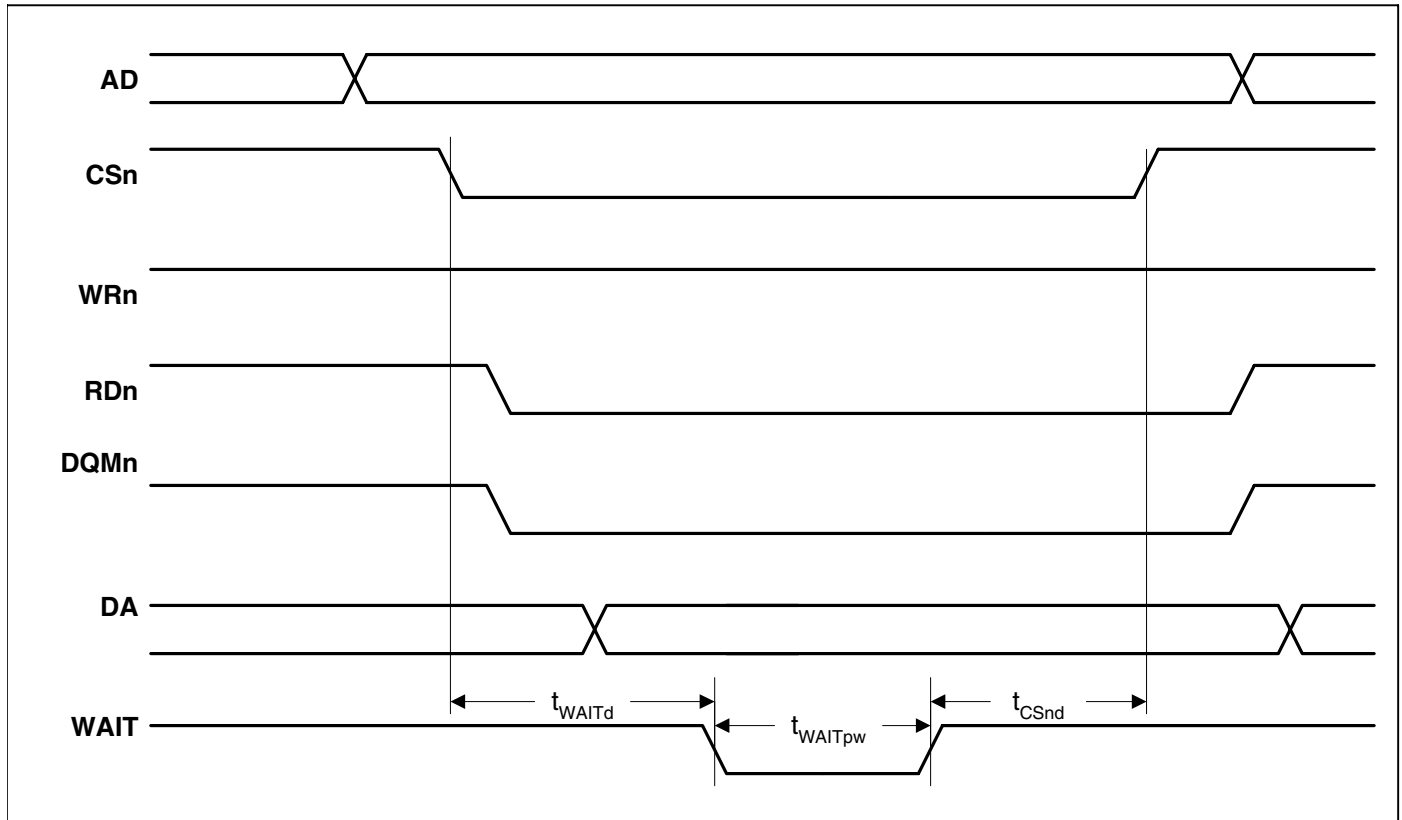
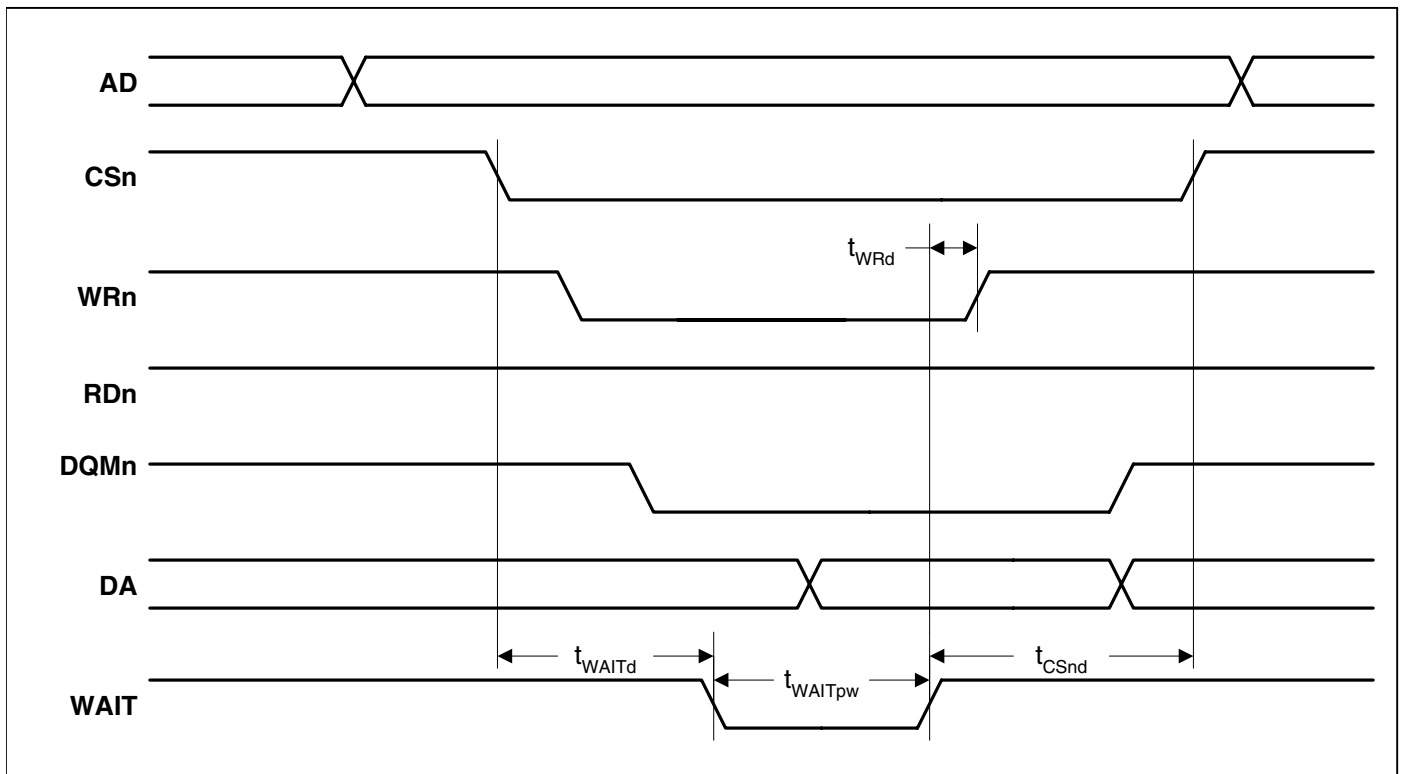


Figure 12. Static Memory Single Read Wait Cycle Timing Measurement

Static Memory Single Write Wait Cycle

| Parameter | Symbol | Min | Typ | Max | Unit |
|---------------------------------|--------------|---------------------|-----|----------------------------|------|
| WAIT to WRn deassert delay time | t_{WRd} | $t_{HCLK} \times 2$ | - | $t_{HCLK} \times 4$ | ns |
| CSn assert to WAIT time | t_{WAITd} | - | - | $t_{HCLK} \times (WST1-2)$ | ns |
| WAIT assert time | t_{WAITpw} | $t_{HCLK} \times 2$ | - | $t_{HCLK} \times 510$ | ns |
| WAIT to CSn deassert delay time | t_{CSnd} | $t_{HCLK} \times 3$ | - | $t_{HCLK} \times 5$ | ns |


Figure 13. Static Memory Single Write Wait Cycle Timing Measurement

Static Memory Turnaround Cycle

| Parameter | Symbol | Min | Typ | Max | Unit |
|-----------------------------------|-------------|-----|----------------------------|-----|------|
| CSnX deassert to CSnY assert time | t_{BTcyc} | - | $t_{HCLK} \times (IDCY+1)$ | - | ns |

- Notes: 1. X and Y represent any two chip select numbers.
 2. IDCY occurs on read-to-write and write-to-read.
 3. IDCY is honored when going from a asynchronous device (CSx) to a synchronous device (/SDCSy).

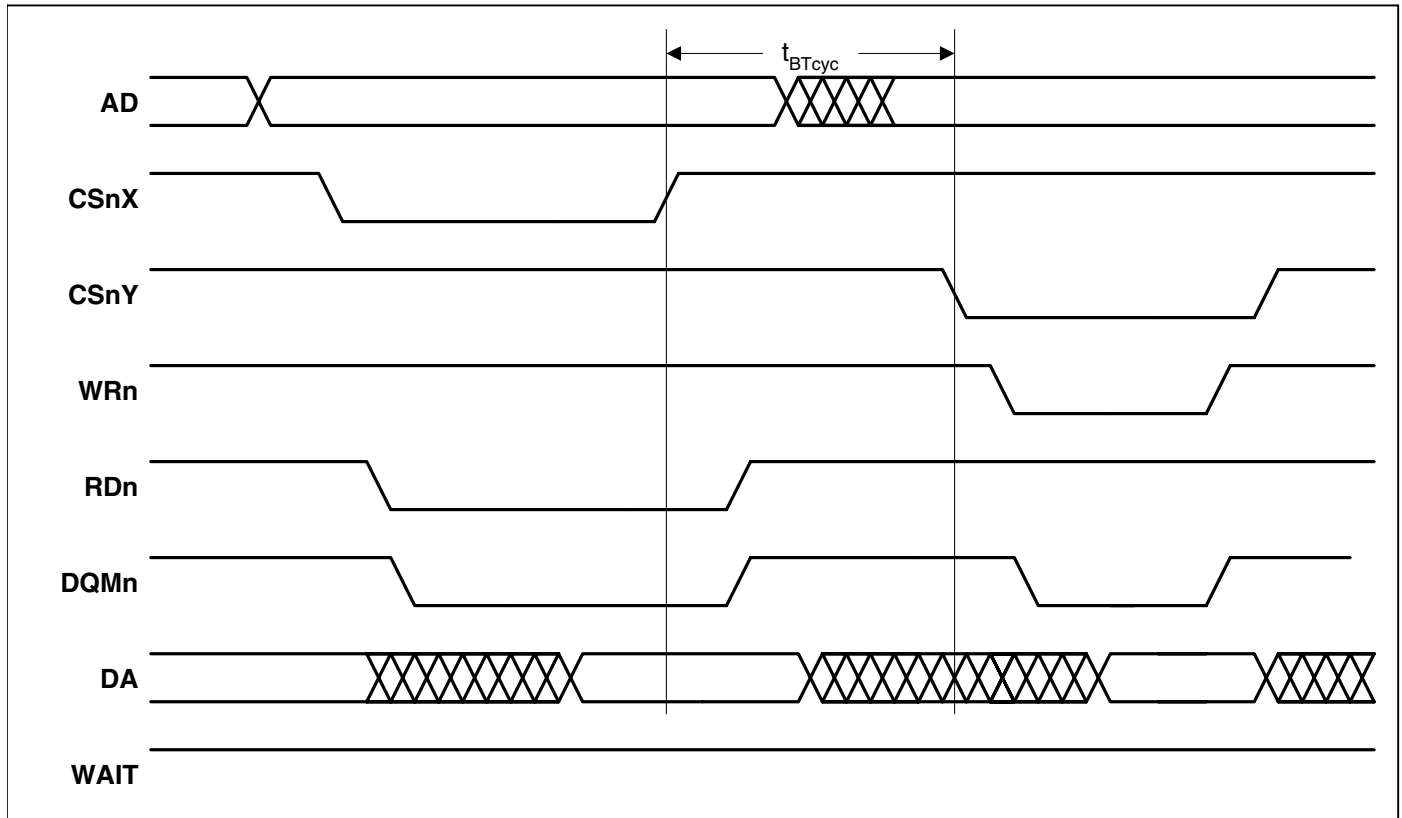


Figure 14. Static Memory Turnaround Cycle Timing Measurement

Ethernet MAC Interface

| Parameter | Symbol | Min | | Typ | | Max | | Unit |
|--|-----------------|--------------|---------------|--------------|---------------|--------------|---------------|------|
| | | 10 Mbit mode | 100 Mbit mode | 10 Mbit mode | 100 Mbit mode | 10 Mbit mode | 100 Mbit mode | |
| TXCLK cycle time | t_{TX_per} | - | - | 400 | 40 | - | - | ns |
| TXCLK high time | t_{TX_high} | 140 | 14 | 200 | 20 | 260 | 26 | ns |
| TXCLK low time | t_{TX_low} | 140 | 14 | 200 | 20 | 260 | 26 | ns |
| TXCLK to signal transition delay time | t_{TXd} | 0 | 0 | 10 | 10 | 25 | 25 | ns |
| TXCLK rise/fall time | t_{TXrf} | - | - | - | - | 5 | 5 | ns |
| RXCLK cycle time | t_{RX_per} | - | - | 400 | 40 | - | - | ns |
| RXCLK high time | t_{RX_high} | 140 | 14 | 200 | 20 | 260 | 26 | ns |
| RXCLK low time | t_{RX_low} | 140 | 14 | 200 | 20 | 260 | 26 | ns |
| RXDVAL / RXERR setup time | t_{RXs} | 10 | 10 | - | - | - | - | ns |
| RXDVAL / RXERR hold time | t_{RXh} | 10 | 10 | - | - | - | - | ns |
| RXCLK rise/fall time | t_{RXrf} | - | - | - | - | 5 | 5 | ns |
| MDC cycle time | t_{MDC_per} | - | - | 400 | 400 | - | - | ns |
| MDC high time | t_{MDC_high} | 160 | 160 | - | - | - | - | ns |
| MDC low time | t_{MDC_low} | 160 | 160 | - | - | - | - | ns |
| MDC rise/fall time | t_{MDCrf} | - | - | - | - | 5 | 5 | ns |
| MDIO setup time (STA sourced) | t_{MDIOs} | 10 | 10 | - | - | - | - | ns |
| MDIO hold time (STA sourced) | t_{MDIOh} | 10 | 10 | - | - | - | - | ns |
| MDC to MDIO signal transition delay time (PHY sourced) | t_{MDIOd} | - | - | - | - | 300 | 300 | ns |

STA - Station - Any device that contains an IEEE 802.11 conforming Medium Access Control (MAC) and physical layer (PHY) interface to the wireless medium.

PHY - Ethernet physical layer interface.

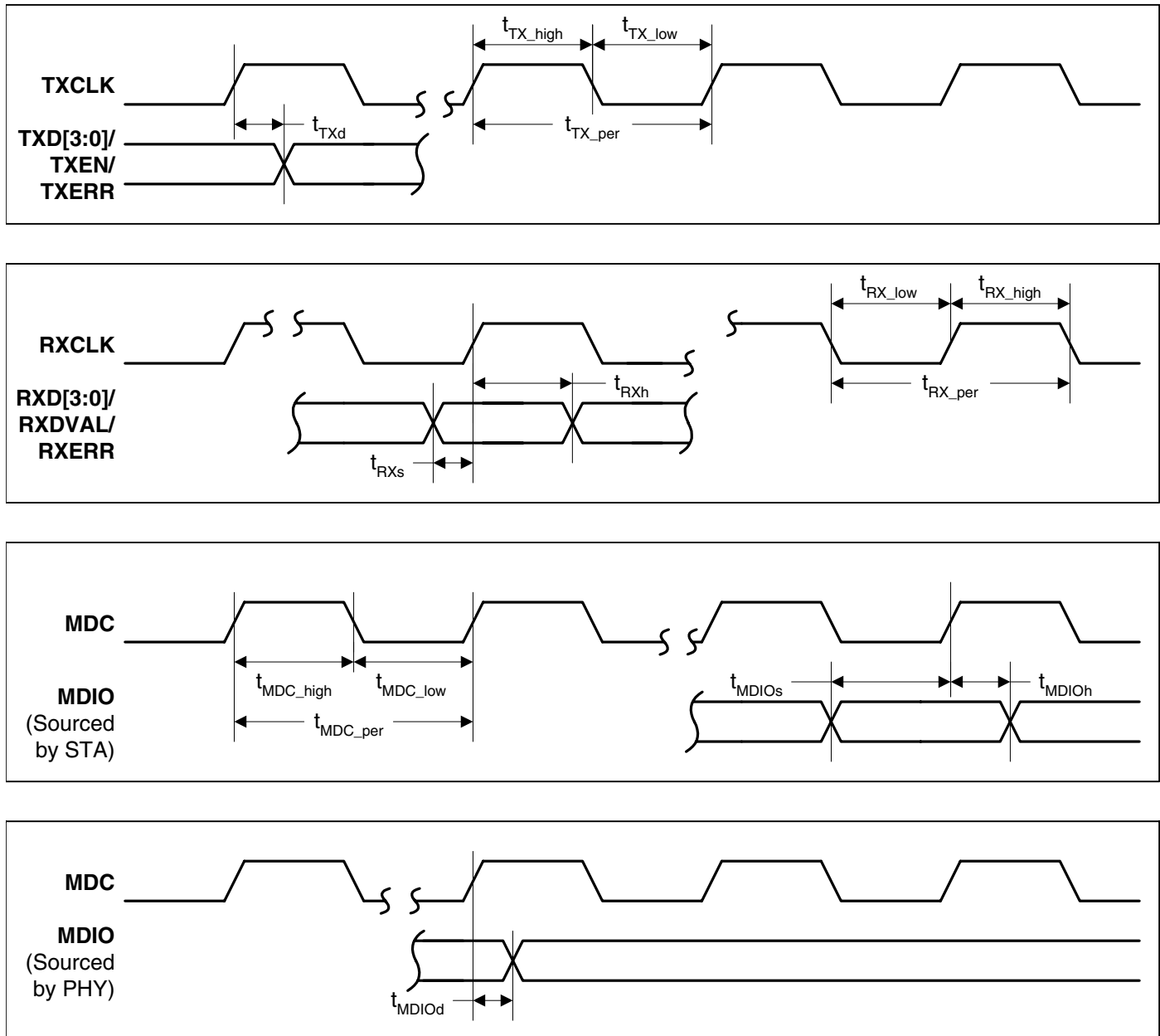


Figure 15. Ethernet MAC Timing Measurement

Audio Interface

The following table contains the values for the timings of each of the SPI modes.

| Parameter | Symbol | Min | Typ | Max | Unit |
|-----------------------------------|------------------------|-----|------------------------------|-----|------|
| SCLK cycle time | $t_{\text{clk_per}}$ | - | $t_{\text{spix_clk}}$ | - | ns |
| SCLK high time | $t_{\text{clk_high}}$ | - | $(t_{\text{spix_clk}}) / 2$ | - | ns |
| SCLK low time | $t_{\text{clk_low}}$ | - | $(t_{\text{spix_clk}}) / 2$ | - | ns |
| SCLK rise/fall time | t_{clkrf} | 1 | - | 8 | ns |
| Data from master valid delay time | t_{DMd} | - | - | 3 | ns |
| Data from master setup time | t_{DMs} | 20 | - | - | ns |
| Data from master hold time | t_{DMh} | 40 | - | - | ns |
| Data from slave setup time | t_{DSs} | 20 | - | - | ns |
| Data from slave hold time | t_{DSH} | 40 | - | - | ns |

Note: The $t_{\text{spix_clk}}$ is programmable by the user.

Texas Instruments' Synchronous Serial Format

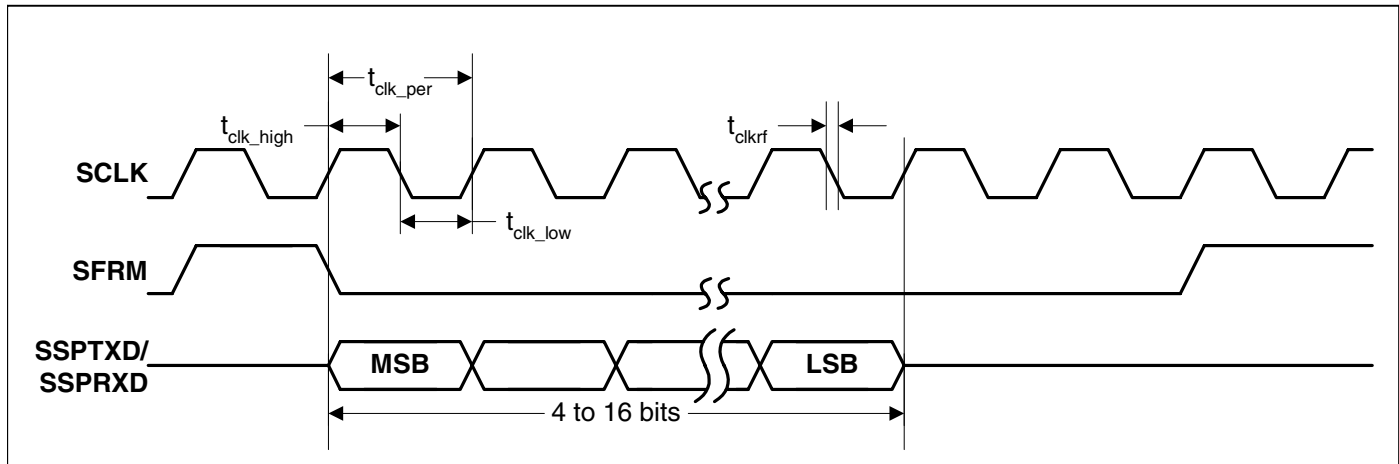


Figure 16. *T*/Single Transfer Timing Measurement

Microwire

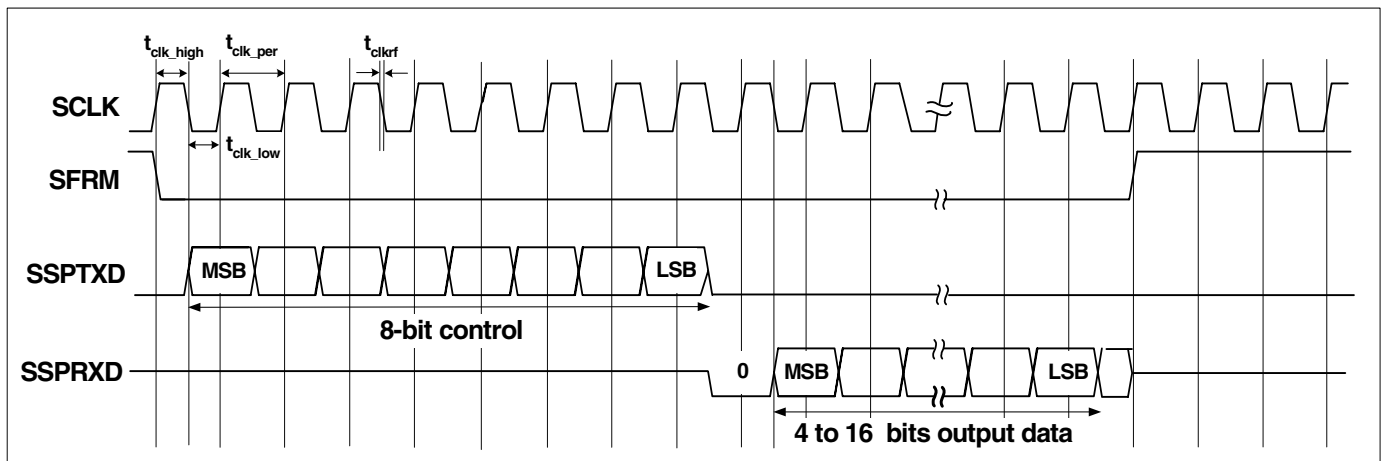
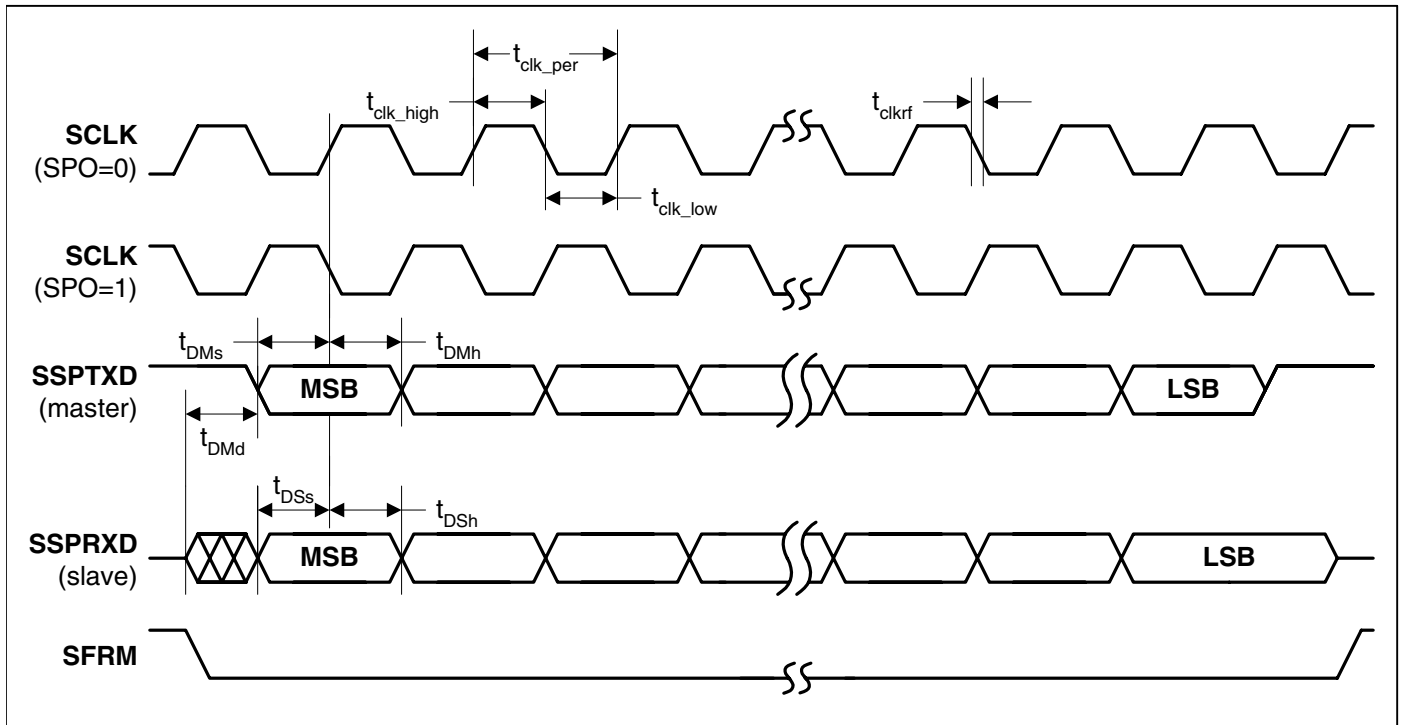


Figure 17. Microwire Frame Format, Single Transfer

Motorola SPI

Figure 18. SPI Format with SPH=1 Timing Measurement

Inter-IC Sound - I²S

| Parameter | Symbol | Min | Typ | Max | Unit |
|--|-----------------|-----|----------------------|-----|------|
| SCLK cycle time | t_{clk_per} | - | t_{i2s_clk} | - | ns |
| SCLK high time | t_{clk_high} | - | $(t_{i2s_clk}) / 2$ | - | ns |
| SCLK low time | t_{clk_low} | - | $(t_{i2s_clk}) / 2$ | - | ns |
| SCLK rise/fall time | t_{clkrf} | 1 | 4 | 8 | ns |
| SCLK to LRCLK assert delay time | t_{LRd} | - | - | 3 | ns |
| Hold between SCLK assert then LRCLK deassert or Hold between LRCLK deassert then SCLK assert | t_{LRh} | 0 | - | - | ns |
| SDI to SCLK deassert setup time | t_{SDIs} | 12 | - | - | ns |
| SDI from SCLK deassert hold time | t_{SDIh} | 0 | - | - | ns |
| SCLK assert to SDO delay time | t_{SDOd} | - | - | 9 | ns |
| SDO from SCLK assert hold time | t_{SDOh} | 1 | - | - | ns |

Note: t_{i2s_clk} is programmable by the user.

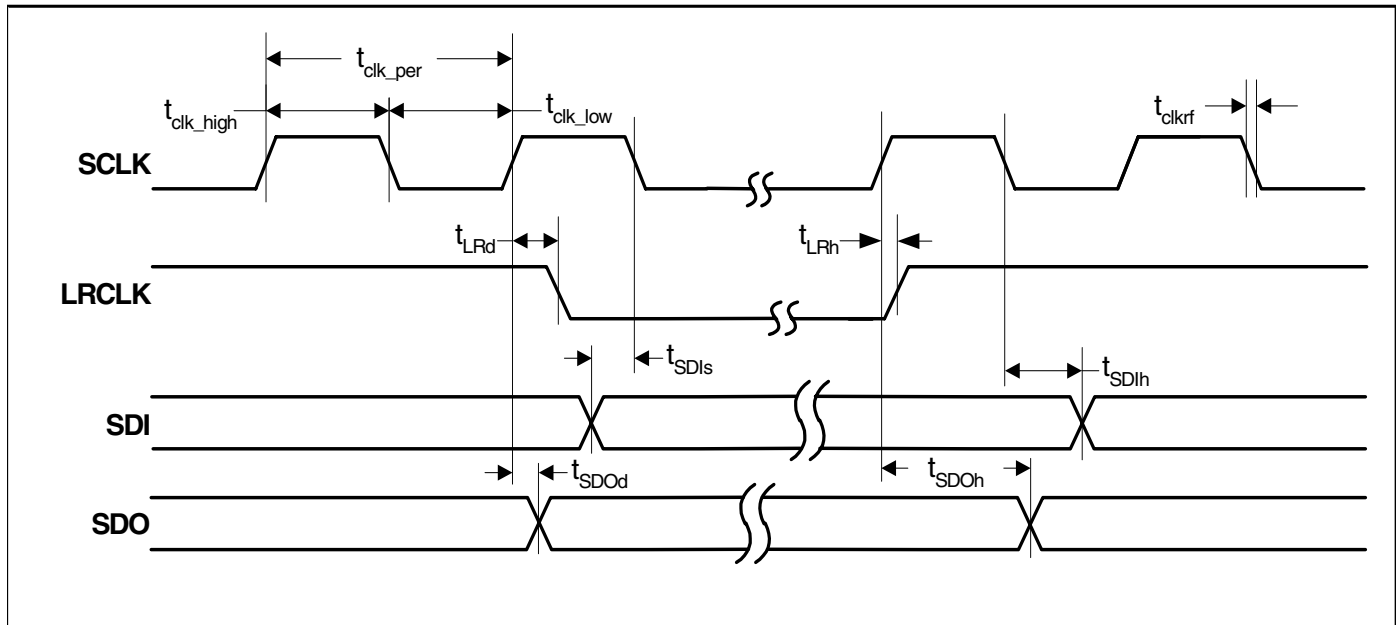
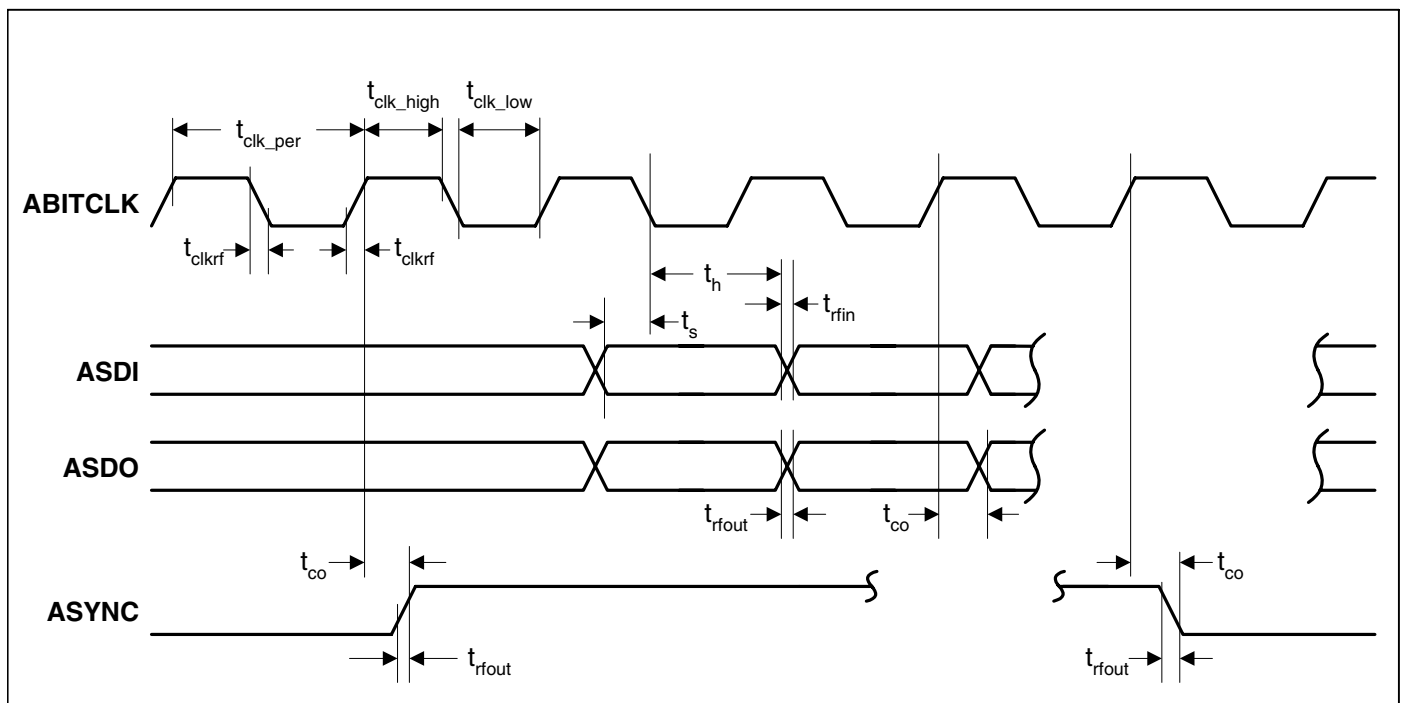


Figure 19. Inter-IC Sound (I²S) Timing Measurement

AC'97

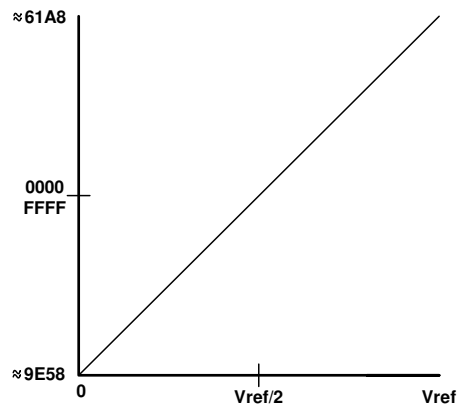
| Parameter | Symbol | Min | Typ | Max | Unit |
|---|-----------------|-----|------|-----|------|
| ABITCLK input cycle time | t_{clk_per} | - | 81.4 | - | ns |
| ABITCLK input high time | t_{clk_high} | 36 | - | 45 | ns |
| ABITCLK input low time | t_{clk_low} | 36 | - | 45 | ns |
| ABITCLK input rise/fall time | t_{clkrf} | 2 | - | 6 | ns |
| ASDI setup to ABITCLK falling | t_s | 10 | - | - | ns |
| ASDI hold after ABITCLK falling | t_h | 10 | - | - | ns |
| ASDI input rise/fall time | t_{rfin} | 2 | - | 6 | ns |
| ABITCLK rising to ASDO / ASYNC valid, $C_L = 55$ pF | t_{co} | 2 | - | 15 | ns |
| ASYNC / ASDO rise/fall time, $C_L = 55$ pF | t_{rfout} | 2 | - | 6 | ns |


Figure 20. AC '97 Configuration Timing Measurement

ADC

| Parameter | Comment | Value | Units |
|------------------------------|---|--------------------------|--|
| Resolution | No missing codes Range of 0 to 3.3 V | 50K counts (approximate) | |
| Integral non-linearity | | 0.01% | |
| Offset error | | ±15 | mV |
| Full scale error | | 0.2% | |
| Maximum sample rate | ADIV = 0 ADIV = 1 | 3750 925 | Samples per second Samples per second |
| Channel switch settling time | ADIV = 0 ADIV = 1 | 500 2 | µs ms |
| Noise (RMS) - typical | | 120 | µV |

Note: ADIV refers to bit 16 in the KeyTchClkDiv register.
 ADIV = 0 means the input clock to the ADC module is equal to the external 14.7456 MHz clock divided by 4.
 ADIV = 1 means the input clock to the ADC module is equal to the external 14.7456 MHz clock divided by 16.



**A/D Converter Transfer Function
(approximately ±25,000 counts)**

Figure 21. ADC Transfer Function

Using the ADC:

This ADC has a state-machine based conversion engine that automates the conversion process. The initiator for a conversion is the read access of the TSXYResult register by the CPU. The data returned from reading this register contains the result as well as the status bit indicating the state of the ADC. However, this peripheral requires a delay between each successful conversion and the issue of the next conversion command, or else the returned value of successive samples may not reflect the analog input. Since the state of the ADC state machine is returned through the same channel used to initiate the conversion process, there must be a delay inserted after every complete conversion. Note that reading TSXYResult during a conversion will not affect the result of the ongoing process.

The following is a recommended procedure for safely polling the ADC from software:

1. Read the TSXYResult register into a local variable to initiate a conversion.
2. If the value of bit 31 of the local variable is '0' then repeat step 1.
3. Delay long enough to meet the maximum sample rate as shown above.
4. Mask the local variable with 0xFFFF to remove extraneous data.
5. If signed mode is used, do a sign extend of the lower halfword.
6. Return the sampled value.

JTAG

| Parameter | Symbol | Min | Max | Units |
|--|-----------------|-----|-----|-------|
| TCK clock period | t_{clk_per} | 100 | - | ns |
| TCK clock high time | t_{clk_high} | 50 | - | ns |
| TCK clock low time | t_{clk_low} | 50 | - | ns |
| TMS / TDI to clock rising setup time | t_{JP_s} | 20 | - | ns |
| Clock rising to TMS / TDI hold time | t_{JP_h} | 45 | - | ns |
| JTAG port clock to output | $t_{JP_{co}}$ | - | 30 | ns |
| JTAG port high impedance to valid output | $t_{JP_{zx}}$ | - | 30 | ns |
| JTAG port valid output to high impedance | $t_{JP_{xz}}$ | - | 30 | ns |

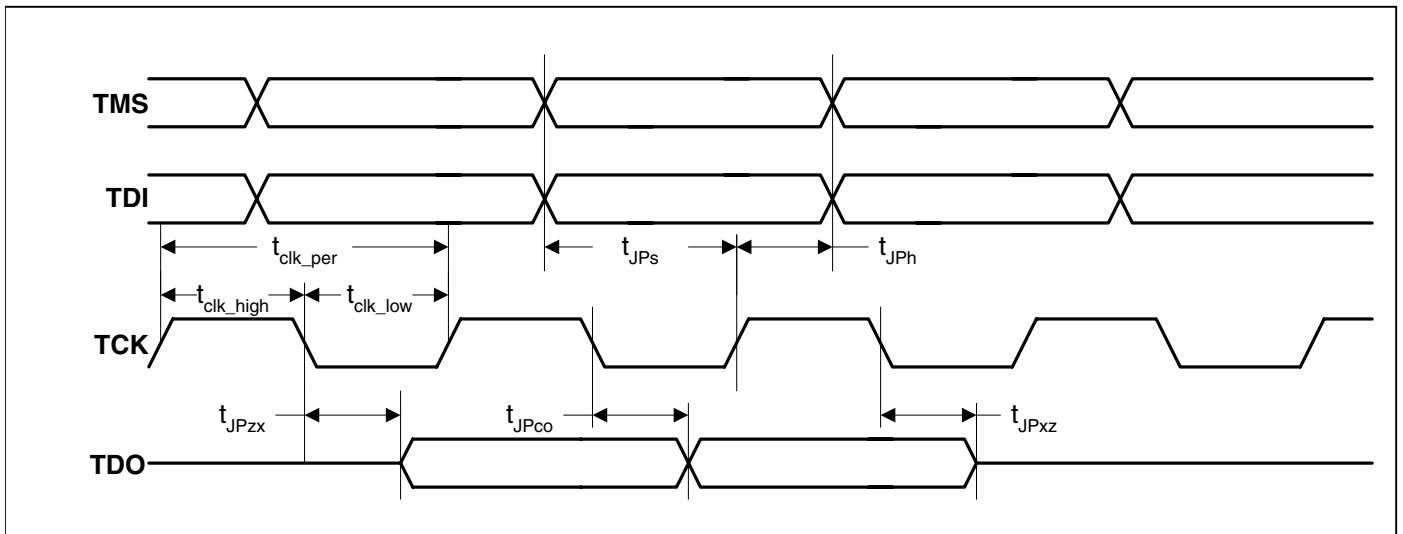
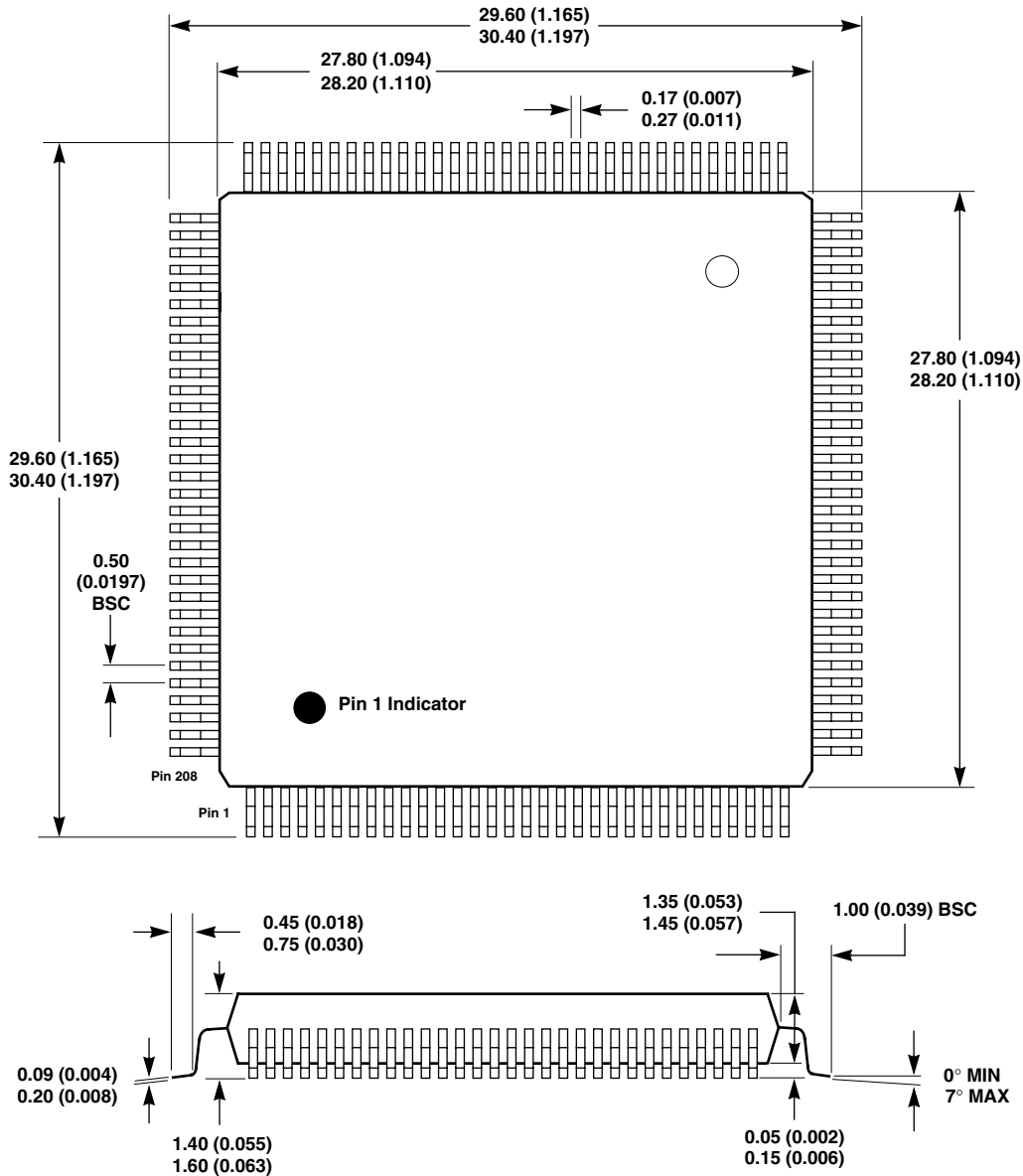


Figure 22. JTAG Timing Measurement

208 Pin LQFP Package Outline

208-Pin LQFP (28 × 28 × 1.40-mm Body)



NOTES:

- 1) Dimensions are in millimeters, and controlling dimension is millimeter.
- 2) Package body dimensions do not include mold protrusion, which is 0.25 mm (0.010 in).
- 3) Pin 1 identification may be either ink dot or dimple.
- 4) Package top dimensions can be smaller than bottom dimensions by 0.20 mm (0.008 in).
- 5) The 'lead width with plating' dimension does not include a total allowable dambar protrusion of 0.08 mm (at maximum material condition).
- 6) Ejector pin marks in molding are present on every package.
- 7) Drawing above does not reflect exact package pin count.

208 Pin LQFP Pinout

The following table shows the 208 pin LQFP pinout.

- VDD_core is CVDD.
- VDD_ring is RVDD.
- NC means that the pin is not connected.

Pin List

The following Low-Profile Quad Flat Pack (LQFP) pin assignment table is sorted in order of pin.

Table P. Pin List in Numerical Order by Pin Number

| Pin Number | Pin Name | Pin Number | Pin Name | Pin Number | Pin Name | Pin Number | Pin Name | Pin Number | Pin Name | Pin Number | Pin Name |
|------------|----------|------------|----------|------------|----------|------------|-----------|------------|-----------|------------|-----------|
| 1 | CSn[7] | 36 | AD[5] | 71 | AD[9] | 106 | USBp[0] | 141 | EGPIO[10] | 176 | TXEN |
| 2 | CSn[6] | 37 | DA[12] | 72 | DA[1] | 107 | ABITCLK | 142 | EGPIO[9] | 177 | MIITXD[0] |
| 3 | CSn[3] | 38 | AD[4] | 73 | AD[8] | 108 | CTSn | 143 | EGPIO[8] | 178 | MIITXD[1] |
| 4 | CSn[2] | 39 | DA[11] | 74 | DA[0] | 109 | RXD[0] | 144 | EGPIO[7] | 179 | MIITXD[2] |
| 5 | CSn[1] | 40 | AD[3] | 75 | DSRn | 110 | RXD[1] | 145 | EGPIO[6] | 180 | MIITXD[3] |
| 6 | AD[25] | 41 | vdd_ring | 76 | DTRn | 111 | vdd_ring | 146 | EGPIO[5] | 181 | TXCLK |
| 7 | vdd_ring | 42 | gnd_ring | 77 | TCK | 112 | gnd_ring | 147 | EGPIO[4] | 182 | RXERR |
| 8 | gnd_ring | 43 | DA[10] | 78 | TDI | 113 | TXD[0] | 148 | EGPIO[3] | 183 | RXDVAL |
| 9 | AD[24] | 44 | AD[2] | 79 | TDO | 114 | TXD[1] | 149 | gnd_ring | 184 | MIIRXD[0] |
| 10 | SDCLK | 45 | DA[9] | 80 | TMS | 115 | CGPIO[0] | 150 | vdd_ring | 185 | MIIRXD[1] |
| 11 | AD[23] | 46 | AD[1] | 81 | vdd_ring | 116 | gnd_core | 151 | EGPIO[2] | 186 | MIIRXD[2] |
| 12 | vdd_core | 47 | DA[8] | 82 | gnd_ring | 117 | PLL_GND | 152 | EGPIO[1] | 187 | gnd_ring |
| 13 | gnd_core | 48 | AD[0] | 83 | BOOT[1] | 118 | XTALI | 153 | EGPIO[0] | 188 | vdd_ring |
| 14 | SDWEn | 49 | vdd_ring | 84 | BOOT[0] | 119 | XTALO | 154 | ARSTn | 189 | MIIRXD[3] |
| 15 | SDCSn[3] | 50 | gnd_ring | 85 | gnd_ring | 120 | PLL_VDD | 155 | TRSTn | 190 | RXCLK |
| 16 | SDCSn[2] | 51 | NC | 86 | NC | 121 | vdd_core | 156 | ASDI | 191 | MDIO |
| 17 | SDCSn[1] | 52 | NC | 87 | EECLK | 122 | gnd_ring | 157 | USBm[2] | 192 | MDC |
| 18 | SDCSn[0] | 53 | vdd_ring | 88 | EEDAT | 123 | vdd_ring | 158 | USBp[2] | 193 | RDn |
| 19 | vdd_ring | 54 | gnd_ring | 89 | ASYNC | 124 | RSTOn | 159 | WAITn | 194 | WRn |
| 20 | gnd_ring | 55 | AD[15] | 90 | vdd_core | 125 | PRSTn | 160 | EGPIO[15] | 195 | AD[16] |
| 21 | RASn | 56 | DA[7] | 91 | gnd_core | 126 | CSn[0] | 161 | gnd_ring | 196 | AD[17] |
| 22 | CASn | 57 | vdd_core | 92 | ASDO | 127 | gnd_core | 162 | vdd_ring | 197 | gnd_core |
| 23 | DQMn[1] | 58 | gnd_core | 93 | SCLK1 | 128 | vdd_core | 163 | EGPIO[14] | 198 | vdd_core |
| 24 | DQMn[0] | 59 | AD[14] | 94 | SFRM1 | 129 | gnd_ring | 164 | EGPIO[13] | 199 | HGPIO[2] |
| 25 | AD[22] | 60 | DA[6] | 95 | SSPRX1 | 130 | vdd_ring | 165 | EGPIO[12] | 200 | HGPIO[3] |
| 26 | AD[21] | 61 | AD[13] | 96 | SSPTX1 | 131 | ADC[4] | 166 | gnd_core | 201 | HGPIO[4] |
| 27 | vdd_ring | 62 | DA[5] | 97 | GRLED | 132 | ADC[3] | 167 | vdd_core | 202 | HGPIO[5] |
| 28 | gnd_ring | 63 | AD[12] | 98 | RDLED | 133 | ADC[2] | 168 | FGPIO[3] | 203 | gnd_ring |
| 29 | DA[15] | 64 | DA[4] | 99 | vdd_ring | 134 | ADC[1] | 169 | FGPIO[2] | 204 | vdd_ring |
| 30 | AD[7] | 65 | AD[11] | 100 | gnd_ring | 135 | ADC[0] | 170 | FGPIO[1] | 205 | AD[18] |
| 31 | DA[14] | 66 | vdd_ring | 101 | INT[3] | 136 | ADC_VDD | 171 | gnd_ring | 206 | AD[19] |
| 32 | AD[6] | 67 | gnd_ring | 102 | INT[1] | 137 | RTCXTALI | 172 | vdd_ring | 207 | AD[20] |
| 33 | DA[13] | 68 | DA[3] | 103 | INT[0] | 138 | RTCXTALO | 173 | CLD | 208 | SDCLKEN |
| 34 | vdd_core | 69 | AD[10] | 104 | RTSn | 139 | ADC_GND | 174 | CRS | | |
| 35 | gnd_core | 70 | DA[2] | 105 | USBm[0] | 140 | EGPIO[11] | 175 | TXERR | | |

The following section focuses on the EP9302 pin signals from two viewpoints - the pin usage and pad characteristics, and the pin multiplexing usage. The first table ([Table Q](#)) is a summary of all the EP9302 pin signals. The second table ([Table R](#)) illustrates the pin signal multiplexing and configuration options.

[Table Q](#) is a summary of the EP9302 pin signals, which illustrates the pad type and pad pull type (if any). The symbols used in the table are defined as follows. (Note: A blank box means Not Applicable (NA) or, for Pull Type, No Pull (NP).)

Under the Pad Type column:

- A - Analog pad
- P - Power pad
- G - Ground pad
- I - Pin is an input only
- I/O - Pin is input/output
- 4mA - Pin is a 4mA output driver
- 8mA - Pin is an 8mA output driver
- 12mA - Pin is an 12mA output driver

See the text description for additional information about bi-directional pins.

Under the Pull Type Column:

- PU - Resistor is a pull up to the RVDD supply
- PD - Resistor is a pull down to the RGND supply

Table Q. Pin Descriptions

| Pin Name | Block | Pad Type | Pull Type | Description |
|------------------|--------|----------|-----------|--------------------------------------|
| TCK | JTAG | I | PD | JTAG clock in |
| TDI | JTAG | I | PD | JTAG data in |
| TDO | JTAG | 4ma | | JTAG data out |
| TMS | JTAG | I | PD | JTAG test mode select |
| TRSTn | JTAG | I | PD | JTAG reset |
| BOOT[1:0] | System | I | PD | Boot mode select in |
| XTALI | PLL | A | | Main oscillator input |
| XTALO | PLL | A | | Main oscillator output |
| VDD_PLL | PLL | P | | Main oscillator power, 1.8V |
| GND_PLL | PLL | G | | Main oscillator ground |
| RTCXTALI | RTC | A | | RTC oscillator input |
| RTCXTALO | RTC | A | | RTC oscillator output |
| WRn | EBUS | 4ma | | SRAM Write strobe out |
| RDn | EBUS | 4ma | | SRAM Read / OE strobe out |
| WAITn | EBUS | I | PU | SRAM Wait in |
| AD[25:0] | EBUS | 8ma | | Shared Address bus out |
| DA[15:0] | EBUS | 8ma | PU | Shared Data bus in/out |
| CSn[3:0] | EBUS | 4ma | PU | Chip select out |
| CSn[7:6] | EBUS | 4ma | PU | Chip select out |
| DQMn[1:0] | EBUS | 8ma | | Shared data mask out |
| SDCLK | SDRAM | 8ma | | SDRAM clock out |
| SDCLKEN | SDRAM | 8ma | | SDRAM clock enable out |
| SDCSn[3:0] | SDRAM | 4ma | | SDRAM chip selects out |
| RASn | SDRAM | 8ma | | SDRAM RAS out |
| CASn | SDRAM | 8ma | | SDRAM CAS out |
| SDWEn | SDRAM | 8ma | | SDRAM write enable out |
| ADC[4:0] | ADC | A | | External Analog Measurement Input |
| VDD_ADC | ADC | P | | ADC power, 3.3V |
| GND_ADC | ADC | G | | ADC ground |
| USBp[2, 0] | USB | A | | USB positive signals |
| USBm[2, 0] | USB | A | | USB negative signals |
| TXD0 | UART1 | 4ma | | Transmit out |
| RXD0 | UART1 | I | PU | Receive in |
| CTS _n | UART1 | I | PU | Clear to send / transmit enable |
| DSR _n | UART1 | I | PU | Data set ready / Data Carrier Detect |
| DTR _n | UART1 | 4ma | | Data Terminal Ready output |
| RTS _n | UART1 | 4ma | | Ready to send |
| TXD1 | UART2 | 4ma | | Transmit / IrDA output |
| RXD1 | UART2 | I | PU | Receive / IrDA input |

Table Q. Pin Descriptions (Continued)

| Pin Name | Block | Pad Type | Pull Type | Description |
|---------------------|--------|----------|-----------|-----------------------------------|
| MDC | EMAC | 4ma | | Management data clock |
| MDIO | EMAC | 4ma | PU | Management data input/output |
| RXCLK | EMAC | I | PD | Receive clock in |
| MIIRXD[3:0] | EMAC | I | PD | Receive data in |
| RXDVAL | EMAC | I | PD | Receive data valid |
| RXERR | EMAC | I | PD | Receive data error |
| TXCLK | EMAC | I | PU | Transmit clock in |
| MIITXD[3:0] | EMAC | I | PD | Transmit data out |
| TXEN | EMAC | 4ma | PD | Transmit enable |
| TXERR | EMAC | 4ma | PD | Transmit error |
| CRS | EMAC | I | PD | Carrier sense |
| CLD | EMAC | I | PU | Collision detect |
| GRLED | LED | 12ma | | Green LED |
| RDLED | LED | 12ma | | Red LED |
| EECLK | EEPROM | 4ma | PU | EEPROM / Two-wire Interface clock |
| EEDAT | EEPROM | 4ma | PU | EEPROM / Two-wire Interface data |
| ABITCLK | AC97 | 8ma | PD | AC97 bit clock |
| ASYNC | AC97 | 8ma | PD | AC97 frame sync |
| ASDI | AC97 | I | PD | AC97 Primary input |
| ASDO | AC97 | 8ma | PU | AC97 output |
| ARST _n | AC97 | 8ma | | AC97 reset |
| SCLK1 | SPI1 | 8ma | PD | SPI bit clock |
| SFRM1 | SPI1 | 8ma | PD | SPI Frame Clock |
| SSPRX1 | SPI1 | I | PD | SPI input |
| SSPTX1 | SPI1 | 8ma | | SPI output |
| INT[3], INT[1:0] | INT | I | PD | External interrupts |
| PRST _n | Syscon | I | PU | Power on reset |
| RSTO _n | Syscon | 4ma | | User Reset in out - open drain |
| SLA[1:0] | EEPROM | 4ma | | Flash programming voltage control |
| EGPIO[15:0] | GPIO | I/O, 4ma | PU | Enhanced GPIO |
| FGPIO[3:1] | GPIO | I/O, 8ma | PU | GPIO on Port F |
| HGPIO[5:2] | GPIO | I/O, 8ma | PU | GPIO on Port H |
| CGPIO[0] | GPIO | I/O, 8ma | PU | GPIO on Port C |
| CVDD | Power | P | | Digital power, 1.8V |
| RVDD | Power | P | | Digital power, 3.3V |
| CGND | Ground | G | | Digital ground |
| RGND | Ground | G | | Digital ground |

Table R illustrates the pin signal multiplexing and configuration options.

Table R. Pin Multiplex Usage Information

| Physical Pin Name | Description | Multiplex signal name |
|-------------------|-------------------------|-----------------------|
| EGPIO[0] | Ring Indicator Input | RI |
| EGPIO[1] | 1Hz clock monitor | CLK1HZ |
| EGPIO[3] | HDLC Clock | HDLCLK1 |
| EGPIO[4] | I2S Transmit Data 1 | SDO1 |
| EGPIO[5] | I2S Receive Data 1 | SDI1 |
| EGPIO[6] | I2S Transmit Data 2 | SDO2 |
| EGPIO[7] | DMA Request 0 | DREQ0 |
| EGPIO[8] | DMA Acknowledge 0 | DACK0 |
| EGPIO[9] | DMA EOT 0 | DEOT0 |
| EGPIO[10] | DMA Request 1 | DREQ1 |
| EGPIO[11] | DMA Acknowledge 1 | DACK1 |
| EGPIO[12] | DMA EOT 1 | DEOT1 |
| EGPIO[13] | I2S Receive Data 2 | SDI2 |
| EGPIO[14] | PWM1 Output | PWMOUT1 |
| EGPIO[15] | Device active / present | DASP |
| ABITCLK | I2S Serial clock | SCLK |
| ASYNC | I2S Frame Clock | LRCK |
| ASDO | I2S Transmit Data 0 | SDO0 |
| ASDI | I2S Receive Data 0 | SDI0 |
| ARSTn | I2S Master clock | MCLK |
| SCLK1 | I2S Serial clock | SCLK |
| SFRM1 | I2S Frame Clock | LRCK |
| SSPTX1 | I2S Transmit Data 0 | SDO0 |
| SSPRX1 | I2S Receive Data 0 | SDI0 |

Acronyms and Abbreviations

The following tables list abbreviations and acronyms used in this data sheet.

| Term | Definition |
|------------------|---|
| ADC | Analog-to-Digital Converter |
| ALT | Alternative |
| AMBA | Advanced Micro-controller Bus Architecture |
| ATAPI | ATA Packet Interface |
| CODEC | COder / DECoder |
| CRC | Cyclic Redundancy Check |
| DAC | Digital-to-Analog Converter |
| DMA | Direct-Memory Access |
| EBUS | External Memory Bus |
| EEPROM | Electrically Erasable Programmable Read Only Memory |
| EMAC | Ethernet Media Access Controller |
| FIFO | First In / First Out |
| FIQ | Fast Interrupt Request |
| FLASH | Flash memory |
| GPIO | General Purpose I/O |
| HDLC | High-level Data Link Control |
| I/F | Interface |
| I ² S | Inter-IC Sound |
| IC | Integrated Circuit |
| ICE | In-Circuit Emulator |
| IDE | Integrated Drive Electronics |
| IEEE | Institute of Electronics and Electrical Engineers |
| IrDA | Infrared Data Association |
| IRQ | Standard Interrupt Request |
| ISO | International Standards Organization |
| JTAG | Joint Test Action Group |
| LFSR | Linear Feedback Shift Register |
| MII | Media-independent Interface |
| MMU | Memory Management Unit |

| Term | Definition |
|-------|--|
| OHCI | Open Host Controller Interface |
| PHY | Ethernet PHYSical layer interface |
| PIO | Programmed I/O |
| RISC | Reduced Instruction Set Computer |
| SDMI | Secure Digital Music Initiative |
| SDRAM | Synchronous Dynamic RAM |
| SPI | Serial Peripheral Interface |
| SRAM | Static Random Access Memory |
| STA | Station - Any device that contains an IEEE 802.11 conforming Medium Access Control (MAC) and physical layer (PHY) interface to the wireless medium |
| TFT | Thin Film Transistor |
| TLB | Translation Lookaside Buffer |
| USB | Universal Serial Bus |

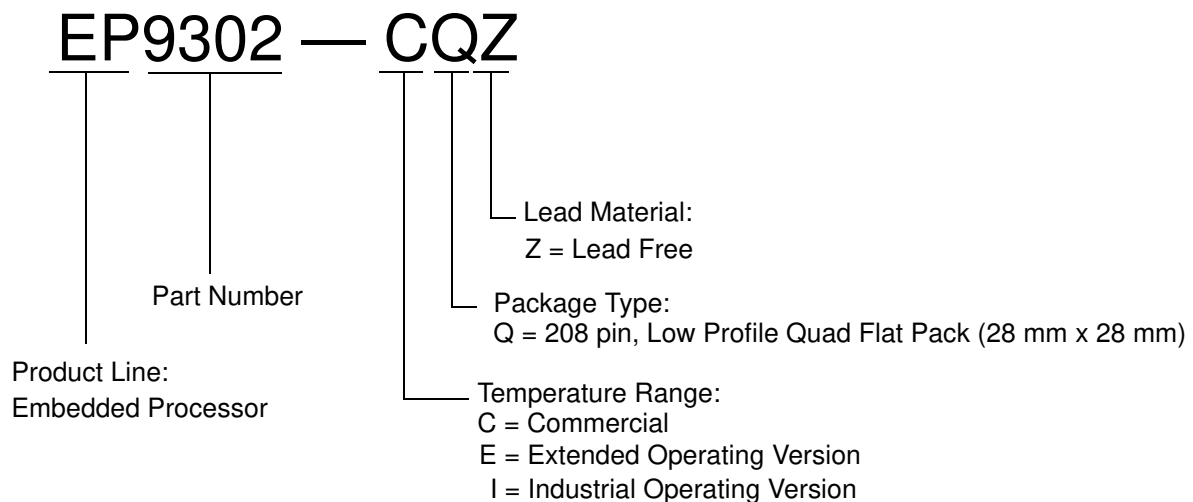
Units of Measurement

| Symbol | Unit of Measure |
|--------|---|
| °C | degree Celsius |
| Hz | Hertz = cycle per second |
| kbps | kilobits per second |
| kbyte | kilobyte |
| kHz | kiloHertz = 1000 Hz |
| Mbps | Megabits per second |
| MHz | MegaHertz = 1,000 KiloHertz |
| μA | microAmpere = 10 ⁻⁶ Ampere |
| μs | microsecond = 1,000 nanoseconds = 10 ⁻⁶ seconds |
| mA | milliAmpere = 10 ⁻³ Ampere |
| ms | millisecond = 1,000 microseconds = 10 ⁻³ seconds |
| mW | milliWatt = 10 ⁻³ Watts |
| ns | nanosecond = 10 ⁻⁹ seconds |
| pF | picoFarad = 10 ⁻¹² Farads |
| V | Volt |
| W | Watt |

Ordering Information

The order numbers for the device are:

| | | | |
|------------|------------------|--------------|-----------|
| EP9302-CQ | 0 °C to +70 °C | 208-pin LQFP | |
| EP9302-CQZ | 0 °C to +70 °C | 208-pin LQFP | Lead Free |
| EP9302-IQ | -40 °C to +85 °C | 208-pin LQFP | |
| EP9302-IQZ | -40 °C to +85 °C | 208-pin LQFP | Lead Free |



Note: Go to the Cirrus Logic Internet site at <http://www.cirrus.com> to find contact information for your local sales representative.

Contacting Cirrus Logic Support

For all product questions and inquiries contact a Cirrus Logic Sales Representative.

To find one nearest you go to www.cirrus.com

IMPORTANT NOTICE

"Preliminary" product information describes products that are in production, but for which full characterization data is not yet available. Cirrus Logic, Inc. and its subsidiaries ("Cirrus") believe that the information contained in this document is accurate and reliable. However, the information is subject to change without notice and is provided "AS IS" without warranty of any kind (express or implied). Customers are advised to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgment, including those pertaining to warranty, indemnification, and limitation of liability. No responsibility is assumed by Cirrus for the use of this information, including use of this information as the basis for manufacture or sale of any items, or for infringement of patents or other rights of third parties. This document is the property of Cirrus and by furnishing this information, Cirrus grants no license, express or implied under any patents, mask work rights, copyrights, trademarks, trade secrets or other intellectual property rights. Cirrus owns the copyrights associated with the information contained herein and gives consent for copies to be made of the information only for use within your organization with respect to Cirrus integrated circuits or other products of Cirrus. This consent does not extend to other copying such as copying for general distribution, advertising or promotional purposes, or for creating any work for resale.

CERTAIN APPLICATIONS USING SEMICONDUCTOR PRODUCTS MAY INVOLVE POTENTIAL RISKS OF DEATH, PERSONAL INJURY, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE ("CRITICAL APPLICATIONS"). CIRRUS PRODUCTS ARE NOT DESIGNED, AUTHORIZED OR WARRANTED FOR USE IN AIRCRAFT SYSTEMS, MILITARY APPLICATIONS, PRODUCTS SURGICALLY IMPLANTED INTO THE BODY, AUTOMOTIVE SAFETY OR SECURITY DEVICES, LIFE SUPPORT PRODUCTS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF CIRRUS PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK AND CIRRUS DISCLAIMS AND MAKES NO WARRANTY, EXPRESS, STATUTORY OR IMPLIED, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR PARTICULAR PURPOSE, WITH REGARD TO ANY CIRRUS PRODUCT THAT IS USED IN SUCH A MANNER. IF THE CUSTOMER OR CUSTOMER'S CUSTOMER USES OR PERMITS THE USE OF CIRRUS PRODUCTS IN CRITICAL APPLICATIONS, CUSTOMER AGREES, BY SUCH USE, TO FULLY INDEMNIFY CIRRUS, ITS OFFICERS, DIRECTORS, EMPLOYEES, DISTRIBUTORS AND OTHER AGENTS FROM ANY AND ALL LIABILITY, INCLUDING ATTORNEYS' FEES AND COSTS, THAT MAY RESULT FROM OR ARISE IN CONNECTION WITH THESE USES.

Cirrus Logic, Cirrus, MaverickCrunch, MaverickKey, and the Cirrus Logic logo designs are trademarks of Cirrus Logic, Inc. All other brand and product names in this document may be trademarks or service marks of their respective owners.

Microsoft and Windows are registered trademarks of Microsoft Corporation.

Microwire is a trademark of National Semiconductor Corp. National Semiconductor is a registered trademark of National Semiconductor Corp.

Texas Instruments is a registered trademark of Texas Instruments, Inc.

Motorola and SPI are registered trademarks of Motorola, Inc.

LINUX is a registered trademark of Linus Torvalds.