PM7800

PALADIN-10

Digital Correction Signal Processor

Data Sheet

PROPRIETARY AND CONFIDENTIAL
Released
Issue 3: July, 2001

PUBLIC REVISION HISTORY

Issue No.	Issue Date	ECN Number	Details of Change
Issue 1	January, 2001	N/A	Document created.
Issue 2	March, 2001	N/A	Modified voltage tolerances for VDD and VDDI. Modified CPUMODE2 connection (tied to VDD rather than GND). Minor editorial changes.
Issue 3	July, 2001	N/A	• Removed I2C references throughout to reflect the fact that I2C is not supported. • Changed Figure 3-1 on page 7: changed GND to VSS and VCC to VDD to match labels used in pin descriptions. • Text edits in Section 5.1 on page 16. • Modified procedure for Programming the WAIT_N Register (Section 5.1.2 on page 23). • Inserted text and diagram regarding relative timing of SPI inputs in Section 5.3 on page 26. • Added section (Section 5.4.2 on page 31) describing bug affecting Power_Attenuator_Delay and Power_Correction_Delay circuits. Modified Figure 5-25 on page 33 to reflect this issue. • Revision number incremented to 1; i.e., rev_code changed to 178000CD. Affects Section 5.6 on page 34 and Table 6-3 on page 41. • Changed VT+ and VT- values to 2.2V min and 0.8V max, respectively, in DC Characteristics. Also, changed VTH to 0.87V typ. • IIH text edit in Table 8-1 on page 51. • added IDDOP (VDDI) and IDDOP (VDDO) in DC Characteristics. • Changed notes at start of AC Timing. • Removed Relative Timing Diagrams and table for serial interface in AC Timing. • Changed min/max loading from 0/40pF to 15/30pF on toe, tdp, toz, and trp, in Table 9-2 on page 54; toe, in Table 9-3 on page 55; and toe, top, and toz in Table 9-7 on page 59.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 Introduction

This document describes the features, functionality, and physical characteristics of the PM7800 Digital Correction Signal Processor, which forms part of the PALADIN-10 system.

1.1 Definitions

The following table defines the terms and abbreviations used in this data sheet.

Table 1-1 Definitions

Abbreviation	Definition
ACPCE	Adaptive Control Processing Compensation Engine
ADC	Analog to Digital Converter
AQM	Analog Quadrature Modulator
BGA	Super Ball Grid Array, the type of package used by this chip
BIST	Built-In Self-Test
BTS	Base Transceiver Station (Node B in WCDMA)
CMOS	Complementary Metal Oxide Semiconductor
CPU	Central Processing Unit - in this context, the CPU is the ACPCE and the terms are used interchangeably
DAC	Digital to Analog Converter
DCSP	Digital Correction Signal Processor
DQM	Digital Quadrature Modulator
DSP	Digital-Signal Processor
EDGE	Enhanced Data Rates for GSM Evolution - 3rd generation GSM (Global System for Mobile Communications)
FIFO	First-In, First-Out - a queuing mechanism
GPIO	General-Purpose Input-Output pin
GSM	Global System for Mobile Communications
NC	No Connect, indicates an unused pin
RAM	Random-Access Memory
RRME	Radio Resource Management Entity
WCDMA	Wideband Code Division Multiple Access

1.2 PALADIN Technologies

PALADIN is a family of distortion elimination chip technologies that enables the development of higher performance, higher efficiency 2G and 3G base stations that utilize fewer and less expensive components. Based on proprietary Digital Signal Processing (DSP) architectures and techniques, PALADIN products feature:

- Digital Adaptive Predistortion for wideband amplifier linearization;
- Digital Correction for Analog Quadrature Modulation (AQM) distortion; and
- Constant-Gain Mode to facilitate operation at higher efficiency.

Emerging 3G wireless services require high capacity radio networks to deliver the high volume multi-media data traffic that is central to the new "wireless internet" paradigm. High spectral efficiency, the ability to maximize the data-carrying capacity of a limited amount of licensed spectrum, will be key to the success of 3G radio networks. However, spectral efficiency is fundamentally limited by distortion in the analog transmitter and power amplifier components of the Base Transceiver Station (BTS).

PALADIN products eliminate distortion in the transmitter chain and power amplifier using fully digital methods, permitting designers to replace many expensive and difficult to manufacture analog IF and RF sub-systems, components which are often required to control distortion and aggregate signals in many existing BTS designs.

In particular, PALADIN can transform an inexpensive, simple, narrow-band Class AB power amplifier into a wide-band, multi-carrier capable, high efficiency, digitally controlled amplifier unit which can replace the expensive, low efficiency, feedforward-based multi-carrier power amplifiers (MCPA) commonly used in many current 3G BTS designs. Furthermore, PALADIN opens the door to the development of new BTS architectures that could feature "standard sockets" for multiple air interfaces, multiple transmitter/amplifier "hot swap" redundancy, multiple amplifier efficiency management, and smart antenna transceivers.

1.3 PALADIN-10

PALADIN-10 is a wideband transceiver linearizer and distortion elimination chip solution for multi-carrier high power radio transmitter applications. Based on high speed digital signal processing (DSP), PALADIN-10 provides digital adaptive predistortion and preconditioning for complex modulation signals with instantaneous bandwidths up to 10MHz. PALADIN-10 consists of (1) the PM7800 Digital Correction Signal Processor (DCSP) chip and (2) the Adaptive Control Processor Compensation Engine (APCE) running on an industry-standard programmable DSP[1].

1. TMS320C54xx

1.3.1 System Diagram

Figure 1-1 shows a simplified system diagram of PALADIN-10 operating in the transmitter chain of a typical multi-carrier base transceiver subsystem (BTS).

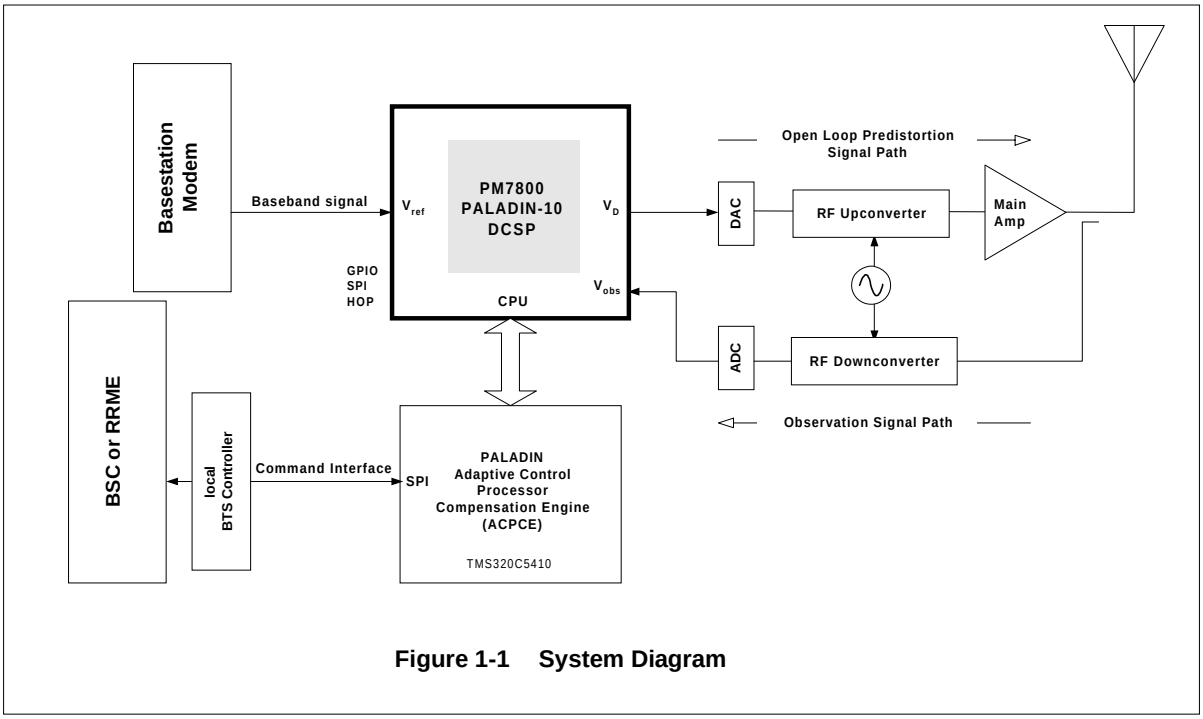


Figure 1-1 System Diagram

The PM7800 DCSP chip, also called the PALADIN predistortion kernel, is responsible for all the real-time operations on the complex modulation baseband signal, from the base station modem, that predistort and precondition it such that the modified signal at the main or power amplifier (PA) will cancel out the distortions due to non-linearities in the PA and transmitter chain. The PM7800 DCSP chip supports all 3G and 2G air interfaces, including WCDMA, cdma2000 and IS-95, and GSM/EDGE. The PM7800 DCSP chip is the very high speed “hardware” digital signal processor component of the PALADIN-10 system.

The PALADIN ACPCE, also called the PALADIN predistortion firmware, is responsible for carrying out all the nonreal-time complex computations necessary to generate predictive parameters used inside the PM7800 DCSP. The PALADIN ACPCE is also responsible for control functions and acts as the Master, taking commands from the base station controller (BSC) or some radio resource management entity (RRME) and controlling the operation and functioning of the PM7800 DCSP chip. The ACPCE runs on an external programmable DSP processor chip (TMS320C5410), also referred to as the CPU. The ACPCE is fully software upgradeable via a

serial interface from the BSC or RRME. The current version of the ACPCE supports the WCDMA air interface standard only. Versions supporting the other 2G and 3G air standards will be provided as software updates. The PALADIN-10 ACPCE is the highly complex adaptive "software" digital signal processor component of the PALADIN-10 system.

1.3.2 Signal Flow

The signal flow through the PALADIN-10 system is as follows:

In the forward direction, the composite (digitally combined single or multi-carrier complex baseband) signal from the base station modem enters the PM7800 through the V_{ref} port, where it is interpolated to the correct rate, predistorted by the Digital Correction Signal (DCSP) Processor core using parameters provided by the Adaptive Control Processor Compensation Engine (ACPCE), and exits through the V_D port. The predistorted signal output at the V_D port is converted to analog, up-converted, amplified, and applied to the antenna. This is called the open-loop predistortion signal path, which is sometimes referred to as the transmit path or upconversion path.

In the reverse direction, a small portion of the amplified signal is downconverted, re-sampled and re-digitized, and enters the PM7800 through the V_{obs} port. This is called the observation or sampling signal path, and is sometimes referred to as the downconversion path. The PM7800 concurrently captures the baseband signal (V_{ref}) and observed signal (V_{obs}) for analysis by the Adaptive Control Processor Compensation Engine (running on a TMS320C5410 DSP). The CPU analyzes the captured data to monitor system performance, and adjusts the internal parameters of the PM7800 DCSP accordingly.

1.4 PM7800 DCSP

The PM7800 DCSP chip is the very high speed "hardware" digital signal processor component of the PALADIN-10 system. The remainder of this document describes the PM7800 DCSP chip only.

The ACPCE is described in a companion document: PMC-2002272, "PALADIN-10/PALADIN-15 ACPCE Firmware User Manual."

1.4.1 Features

- predistortion kernel for linearization of power-amplifiers in wireless base-stations
- up to 80MHz data-rate
- interpolated up-conversion (1:N, where 1<=N<=10) of baseband input to the data-rate
- generic 16-bit microprocessor bus interface for configuration, control, and monitoring
- SPI serial interface for update of power and carrier values
- 48 general-purpose IO pins, eight of which are edge-triggered interrupt sources
- standard five-signal IEEE 1149.1 JTAG test port for boundary scan board test purposes
- low-power 1.8V CMOS core logic with 3.3V CMOS/TTL compatible digital inputs and digital outputs
- Industrial temperature range (-40C to +85C)

1.4.2 Interfaces

The major interfaces to the PM7800 DCSP include:

- Baseband interface to the base station modem (V_{ref})
- Digital IF or Digital Baseband interface to DAC(s) and RF Upconverter (V_D)
- Digital IF or Digital Baseband interface from ADC(s) and RF Downconverter (V_{obs})
- GPIO and Serial (SPI) interface for auxiliary monitor and control
- CPU interface to ACPCE processor (external C54xx DSP)

Note: The command interface to the BSC or RRME is through an SPI (serial) interface on the CPU (TMS320C5410). The command interface is separately detailed in the companion document: PMC-2002272, “PALADIN-10/PALADIN-15 ACPCE Firmware User Manual.”

1.4.3 Applications

- WCDMA Base Transceiver Subsystems (BTS)
- CDMA2000 BTS (requires firmware upgrade)
- GSM/TDMA/EDGE BTS (requires firmware upgrade)

1.4.4 References

1. TMS320VC5410 Fixed-Point Digital Signal Processor (Texas Instruments)
2. Section 8: Synchronous Serial Peripheral Interface, M68HC11 Reference Manual (Motorola)
3. PMC-2002272, “PALADIN-10/PALADIN-15 ACPCE Firmware User Manual.”

2 Block Diagram

A block diagram of the PM7800 DCSP, showing the major interfaces is shown below as:

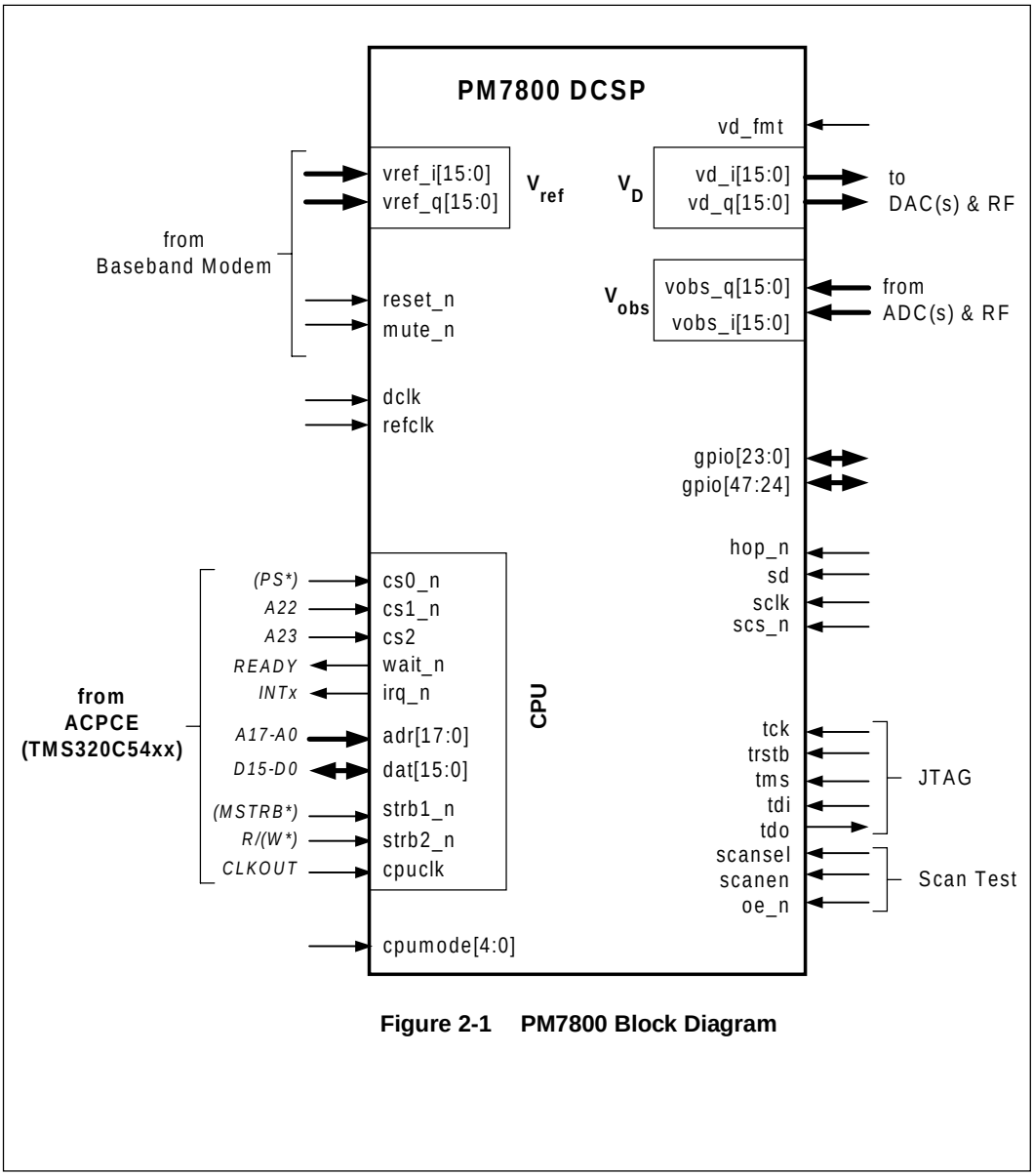
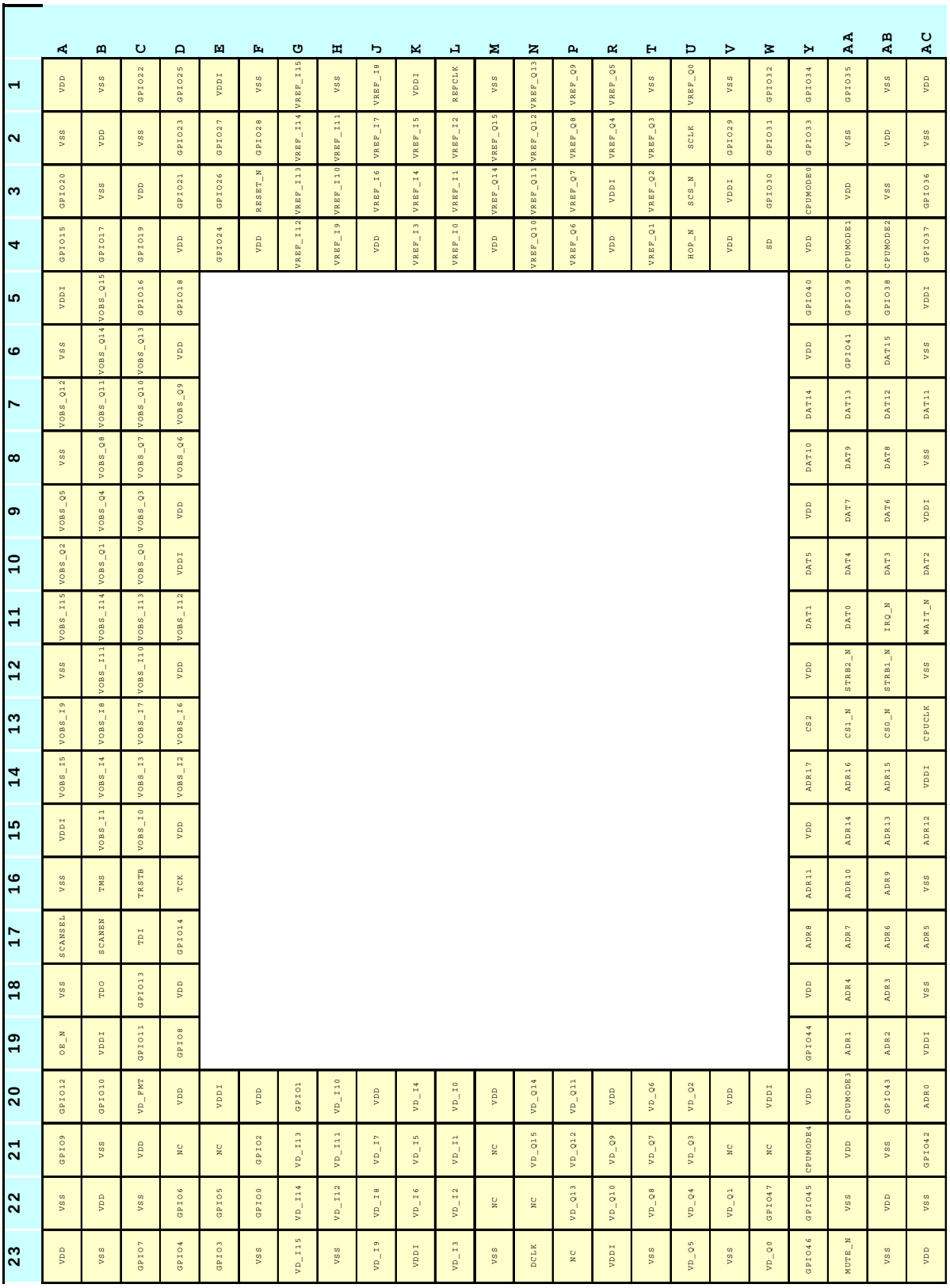


Figure 2-1 PM7800 Block Diagram

3 Pin Diagram

The PM7800 is packaged in a 304-pin SBGA with a body size of 31mm x 31mm. The following pin diagram shows the pinout from the ball-side of the package. Note that, for readability, the aspect ratio of the diagram has been changed.

Figure 3-1 Pin Diagram (Bottom View)



	A	B	C	D	E	F	G	H	J	K	L	M	N	P	R	T	U	V	W	Y	AA	AB	AC
1	VDD	VSS	GPIO22	GPIO25	VDDI	VSS	VREF_I15	VSS	VREF_I8	VDDI	REFCLK	VSS	VREF_Q13	VREF_Q9	VREF_Q5	VSS	VREF_Q0	VSS	GPIO32	GPIO34	GPIO35	VSS	VDD
2	VSS	VDD	VSS	GPIO23	GPIO27	GPIO28	VREF_I14	VREF_I11	VREF_I7	VREF_I5	VREF_I2	VREF_Q15	VREF_Q12	VREF_Q8	VREF_Q4	VREF_Q3	SCLK	GPIO29	GPIO31	GPIO33	VSS	VDD	VSS
3	GPIO20	VSS	VDD	GPIO21	GPIO26	RESET_N	VREF_I13	VREF_I10	VREF_I6	VREF_I4	VREF_I1	VREF_Q14	VREF_Q11	VREF_Q7	VDDI	VREF_Q2	SCS_N	VDDI	GPIO30	CPUMODE0	VDD	VSS	GPIO36
4	GPIO15	GPIO17	GPIO19	VDD	GPIO24	VDD	VREF_I12	VREF_I9	VDD	VREF_I3	VREF_I0	VDD	VREF_Q10	VREF_Q6	VDD	VREF_Q1	HOP_N	VDD	SD	VDD	CPUMODE1	CPUMODE2	GPIO37
5	VDDI	VOBS_Q15	GPIO16	GPIO18																GPIO40	GPIO39	GPIO38	VDDI
6	VSS	VOBS_Q14	VOBS_Q13	VDD																VDD	GPIO41	DAT15	VSS
7	VOBS_Q12	VOBS_Q11	VOBS_Q10	VOBS_Q9																DAT14	DAT13	DAT12	DAT11
8	VSS	VOBS_Q8	VOBS_Q7	VOBS_Q6																DAT10	DAT9	DAT8	VSS
9	VOBS_Q5	VOBS_Q4	VOBS_Q3	VDD																VDD	DAT7	DAT6	VDDI
10	VOBS_Q2	VOBS_Q1	VOBS_Q0	VDDI																DAT5	DAT4	DAT3	DAT2
11	VOBS_I15	VOBS_I14	VOBS_I13	VOBS_I12																DAT1	DAT0	IRQ_N	WAIT_N
12	VSS	VOBS_I11	VOBS_I10	VDD																VDD	STRB2_N	STRB1_N	VSS
13	VOBS_I9	VOBS_I8	VOBS_I7	VOBS_I6																CS2	CS1_N	CS0_N	CPUCLK
14	VOBS_I5	VOBS_I4	VOBS_I3	VOBS_I2																ADR17	ADR16	ADR15	VDDI
15	VDDI	VOBS_I1	VOBS_I0	VDD																VDD	ADR14	ADR13	ADR12
16	VSS	TMS	TRSTB	TCK																ADR11	ADR10	ADR9	VSS
17	SCANSEL	SCANEN	TDI	GPIO14																ADR8	ADR7	ADR6	ADR5
18	VSS	TDO	GPIO13	VDD																VDD	ADR4	ADR3	VSS
19	OE_N	VDDI	GPIO11	GPIO8																GPIO44	ADR1	ADR2	VDDI
20	GPIO12	GPIO10	VD_PMT	VDD	VDDI	VDD	GPIO1	VD_I10	VDD	VD_I4	VD_I0	VDD	VD_Q14	VD_Q11	VDD	VD_Q6	VD_Q2	VDD	VDDI	VDD	CPUMODE3	GPIO43	ADR0
21	GPIO9	VSS	VDD	NC	NC	GPIO2	VD_I13	VD_I11	VD_I7	VD_I5	VD_I1	NC	VD_Q15	VD_Q12	VD_Q9	VD_Q7	VD_Q3	NC	NC	CPUMODE4	VDD	VSS	GPIO42
22	VSS	VDD	VSS	GPIO6	GPIO5	GPIO0	VD_I14	VD_I12	VD_I8	VD_I6	VD_I2	NC	NC	VD_Q13	VD_Q10	VD_Q8	VD_Q4	VD_Q1	GPIO47	GPIO45	VSS	VDD	VSS
23	VDD	VSS	GPIO7	GPIO4	GPIO3	VSS	VD_I15	VSS	VD_I9	VDDI	VD_I3	VSS	DCLK	NC	VDDI	VSS	VD_Q5	VSS	VD_Q0	GPIO46	MUTE_N	VSS	VDD

4 Pin Description

Table 4-1 Pin Description and Cell Types

Pin Name	Type	Pin #	Default	Functional Description
CPU Interface (47)				
STRB1_N	I	AB12	Hi-Z	This signal is either used as an access strobe or as a write strobe, depending on the setting of the CPUMODE0 pin. If the CPUMODE0 pin is connected to VSS, this signal is treated as an access strobe, if the CPUMODE0 pin is connected to VDD, then this strobe is treated as a WRITE_N strobe.
STRB2_N	I	AA12	Hi-Z	This signal is either used as an Read/Write_N cycle type indication or as a read strobe, depending on the setting of the CPUMODE0 pin. If the CPUMODE0 pin is connected to VSS, this signal is treated as an Read/Write_N indication, if the CPUMODE0 pin is connected to VDD, then this strobe is treated as a READ_N strobe.
CS0_N	I	AB13	Hi-Z	This signal is an active-low chip select. All three chip selects must be asserted for the chip to be accessed.
CS1_N	I	AA13	Hi-Z	This signal is an active-low chip select. All three chip selects must be asserted for the chip to be accessed.
CS2	I	Y13	Hi-Z	This signal is an active-high chip select. All three chip selects must be asserted for the chip to be accessed.
ADR[17:0]	I	Y14, AA14, AB14, AA15, AB15, AC15, Y16, AA16, AB16, Y17, AA17, AB17, AC17, AA18, AB18, AB19, AA19, AC20	Hi-Z	This is the eighteen-bit word-address bus. The bus addresses 256k 16-bit words (512Kbytes) in the PM7800.

Table 4-1 Pin Description and Cell Types

Pin Name	Type	Pin #	Default	Functional Description
DAT[15:0]	IO	AB6, Y7, AA7, AB7, AC7, Y8, AA8, AB8, AA9, AB9, Y10, AA10, AB10, AC10, Y11, AA11	Hi-Z	This is the 16-bit data bus.
WAIT_N	O	AC11	Output High	This is the external wait signal, required when DCLK on the PM7800 is slow. This output has programmable timing. See *Programming the WAIT_N Register* section 5.1.2.
IRQ_N	O	AB11	Output High	This is an active-low interrupt request pin.
CPUCLK	I	AC13	Hi-Z	CPU Clock Input. This is a Schmitt-trigger input.
CPUMODE0	I	Y3	Hi-Z	This pin configures the strobe-signalling mode of the PM7800. When this pin is connected to VSS, access strobe and R/W# indication signalling is used. When this pin is connected to VDD, read strobe and write strobe signaling is used. See *CPU Interface* section 5.1.
CPUMODE1	I	AA4	Hi-Z	If this pin is connected to VDD, the WAIT circuit deactivates the internal version of the CPU write strobe for a programmable number of CPUCLK cycles. See *CPU Interface* section 5.1.
CPUMODE2	I	AB4	Hi-Z	This input is reserved and must be connected to VDD.
CPUMODE3	I	AA20	Hi-Z	This pin, when connected to VDD, selects a mode of operation that can be used to support processors, e.g. ARM, that require the read strobe to be sampled on the falling edge of CPUCLK and the write strobe to be sampled on the rising edge. See *CPU Interface* section 5.1.
CPUMODE4	I	Y21	Hi-Z	This pin, when connected to VDD, enables additional delay on the WAIT_N output to provide more hold time. See *CPU Interface* section 5.1.
Data Interfaces (98)				
DCLK	I	N23	Hi-Z	Data clock. This is a Schmitt-trigger input.
REFCLK	I	L1	Hi-Z	VREF clock. This is a Schmitt-trigger input. VREF is synchronous to this clock in 2-clock systems.

Table 4-1 Pin Description and Cell Types

Pin Name	Type	Pin #	Default	Functional Description
VREF_I[15:0]	IO	G1, G2, G3, G4, H2, H3, H4, J1, J2, J3, K2, K3, K4, L2, L3, L4	Hi-Z	In normal modes, this bus is the I-channel of the baseband input signal. In scan test mode, this bus is an output used for SCANOUT[31:16].
VREF_Q[15:0]	IO	M2, M3, N1, N2, N3, N4, P1, P2, P3, P4, R1, R2, T2, T3, T4, U1	Hi-Z	In normal modes, this bus is the Q-channel of baseband input signal. In scan test mode, this bus is an output used for SCANOUT[15:0].
VOBS_I[15:0]	I	A11, B11, C11, D11, B12, C12, A13, B13, C13, D13, A14, B14, C14, D14, B15, C15	Hi-Z	Input: I-channel of observed signal from power amplifier.

Table 4-1 Pin Description and Cell Types

Pin Name	Type	Pin #	Default	Functional Description
VOBS_Q[15:0]	I	B5, B6, C6, A7, B7, C7, D7, B8, C8, D8, A9, B9, C9, A10, B10, C10	Hi-Z	Input: Q-channel of observed signal from power amplifier.
VD_I[15:0]	IO	G23, G22, G21, H22, H21, H20, J23, J22, J21, K22, K21, K20, L23, L22, L21, L20	Hi-Z	In normal modes, this bus is the I-channel of output signal to the DAC. In scan test mode, this bus is an input used for SCANIN[31:16].
VD_Q[15:0]	IO	N21, N20, P22, P21, P20, R22, R21, T22, T21, T20, U23, U22, U21, U20, V22, W23	Hi-Z	In normal modes, this bus is the Q-channel of output signal to the DAC. In scan test mode, this bus is an input used for SCANIN[15:0].
Serial Port (4)				
SD	I	W4	Hi-Z	Serial Data input for the serial port. If unused, this pin must be connected to VSS.
SCLK	I	U2	Hi-Z	Clock input for the serial port - Schmitt trigger. If unused, this pin must be connected to VSS.

Table 4-1 Pin Description and Cell Types

Pin Name	Type	Pin #	Default	Functional Description
SCS_N	I	U3	Hi-Z	Chip Select for the serial port. This is a Schmitt-trigger input. If unused, this pin must be connected to VSS.
HOP_N	I	U4	Hi-Z	This input can be used to activate the new carrier and power values. This is a Schmitt-trigger input. If unused, this pin must be connected to VSS.
JTAG and other test pins (8)				
TCK	I	D16	Hi-Z	JTAG clock. In normal operation, this pin must be connected to VSS.
TRSTB	I	C16	Hi-Z pulled high	JTAG reset signal - active low. This is a Schmitt-trigger input with internal pull-up resistor. In normal mode (non-JTAG) this pin must be connected to RESET_N.
TMS	I	B16	Hi-Z pulled high	JTAG test mode select input with internal pull-up resistor. In normal operation, this pin must be driven high or left unconnected.
TDI	I	C17	Hi-Z pulled high	JTAG test data input with internal pull-up resistor. In normal operation, this pin must be driven high or left unconnected.
TDO	O	B18	Hi-Z	Tri-state output for JTAG test data. This is the only pin unaffected by OE_N.
SCANSEL	I	A17	Hi-Z	Scan MUX select. When high, this signal selects Scan test mode. This pin must be connected to VSS in normal operation.
SCANEN	I	B17	Hi-Z	When high, this signal enables the Scan chain. This pin must be connected to VSS in normal operation.
OE_N	I	A19	Hi-Z pulled low	Forces all pins except TDO to high impedance. This is a Schmitt-trigger input with internal pull-down resistor. In normal operation, this pin must be driven low or left unconnected.
Miscellaneous (50)				
RESET_N	I	F3	Hi-Z pulled high	System reset signal - active low. This is a Schmitt-trigger input with internal pull-up resistor.
MUTE_N	I	AA23	Hi-Z pulled high	Mute input - active-low. When low, this signal forces VD to zero. This is a Schmitt-trigger input with internal pull-up resistor.
VD_FMT	I	C20	Hi-Z	Configuration pin that sets the data format of the VD_I and VD_Q outputs: 0 -> offset-binary 1 -> 2s-complement.

Table 4-1 Pin Description and Cell Types

Pin Name	Type	Pin #	Default	Functional Description
GPIO[47:32]	IO	W22, Y23, Y22, Y19, AB20, AC21, AA6, Y5, AA5, AB5, AC4, AC3, AA1, Y1, Y2, W1	Hi-Z	General-purpose IO pins.
GPIO[31:28]	IO	W2, W3, V2, F2	Hi-Z	General-purpose IO pins with rising-edge interrupt capability (see Interrupt_Enable2 register).
GPIO[27:24]	IO	E2, E3, D1, E4	Hi-Z	General-purpose IO pins with falling-edge interrupt capability (see Interrupt_Enable2 register).
GPIO[23:18]	IO	D2, C1, D3, A3, C4, D5	Hi-Z	General-purpose IO pins.
GPIO[17]	IO	B4	Hi-Z	General-purpose IO pins or Power_Attenuator_Pulse output (see GPIO_Secondary_Function_Select registers).
GPIO[16]	IO	C5	Hi-Z	General-purpose IO pins or Watchdog_Timer_Output (see GPIO_Secondary_Function_Select registers).
GPIO[15:0]	IO	A4, D17, C18, A20, C19, B20, A21, D19, C23, D22, E22, D23, E23, F21, G20, F22	Hi-Z	General-purpose IO pins or Power_Attenuator_Output[15:0] (see GPIO_Secondary_Function_Select registers).
Power Supply (84)				

Table 4-1 Pin Description and Cell Types

Pin Name	Type	Pin #	Default	Functional Description
VSS	P	A2 A6 A8 A12 A16 A18 A22 B1 B3 B21 B23 C2 C22 F1 F23 H1 H23 M1 M23 T1 T23 V1 V23 AA2 AA22 AB1 AB3 AB21 AB23 AC2 AC6 AC8 AC12 AC16 AC18 AC22		Ground supply.

Table 4-1 Pin Description and Cell Types

Pin Name	Type	Pin #	Default	Functional Description
VDD	P	A1 A23 B2 B22 C3 C21 D4 D6 D9 D12 D15 D18 D20 F4 F20 J4 J20 M4 M20 R4 R20 V4 V20 Y4 Y6 Y9 Y12 Y15 Y18 Y20 AA3 AA21 AB2 AB22 AC1 AC23		3.3V IO voltage supply.
VDDI	P	E20, K23, R23, W20, AC19, AC14 AC9, AC5, V3, R3, K1, E1, A5, D10, A15, B19		1.8V Core voltage supply.

5 Functional Description

5.1 CPU Interface

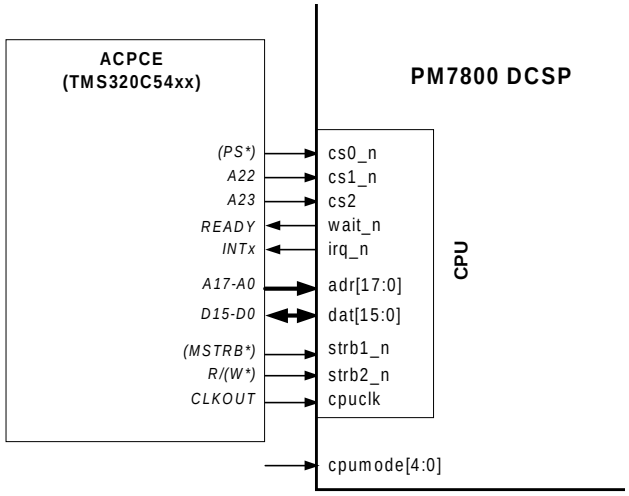


Figure 5-1 CPU Interface

The CPU interface is designed to “look” like the interface to a slow asynchronous SRAM device, with some added functionality to reduce glue logic. The CPU interface is primarily designed for a TI TMS320C54xx General Purpose DSP processor, but some flexibility is provided to support other processors.

Internally, the CPU access cycle is synchronized to the DCLK to allow a simultaneous CPU write to, and data path read from, the same memory location - this is to provide fail-safe operation of the internal dual-port RAM. This requirement significantly slows down the access speed of the CPU interface, especially at EDGE sample clock rates in the 13MHz range. The slow access speed can result in a longer access time than what can be generated by the internal wait-state machine of the processor when the processor is fast and DCLK is slow. To generate longer CPU access cycles, a wait signal (WAIT_N) is generated by the PM7800. The WAIT_N signal is generated by the wait circuit which is clocked by CPUCLK.

The PM7800 needs to detect cycle-to-cycle transitions on the CPU bus. Detecting the cycle-to-cycle change is complicated because the cycle-inactive indication (deassertion of the strobes) between cycles can be short for write-to-write cycles, and short or non-existent for read-to-read cycle transitions. To simplify the design and stay within synchronous design practices, cycle-to-

cycle transition-detection is achieved by the direct approach of sampling the control signal(s) inactive, or by the indirect approach of sampling the control strobes a programmable number of clock cycles after the WAIT_N signal is deasserted - if the strobes are still active, then the start of another cycle is assumed.

The operation of the wait circuit - clocked by CPUCLK - is as follows: sample control strobes active, assert WAIT_N for Assert_Wait_Cycles (register 0x0_0010 bits 6:0) CPUCLK's, negate WAIT_N for Negate_Wait_Cycles (register 0x0_0010 bits 11:8) CPUCLK's, then sample control strobes again. Another cycle has started if the control strobes are still active. The TI processor's internal wait-state circuit continues to count while WAIT_N is asserted. This results in the wait-state circuitry re-aligning itself with the TI cycle at each transition since the TI cycle will end as soon as it detects WAIT_N negated. Other processors' internal wait-state circuits may halt while WAIT_N is asserted: in this case, the Assert_Wait_Cycles and Negate_Wait_Cycles registers must be programmed so that they match the length of the processor accesses.

To further support the case where the de-assertion period of the strobe signal between back-to-back writes is too short to be reliably detected by DCLK, the PM7800 must be programmed to temporarily deassert its internal version of the write strobe: CPUMODE1, when tied to VDD, selects this operation. The Fake_End_Length register (register 0x0_0010 bits 15:12) sets the duration, in CPUCLK's, for which the write strobe is deasserted.

The C54xx interface has an access strobe (either I/O or memory), an area indication (either DS# or PS# for data or program memory or IS# for I/O space select), a R/W# indication, a data bus and an address bus. To support C54xx processors, STRB1_N and STRB2_N must be connected to the access strobe and R/W# indicator respectively, and CPUMODE0 must be tied to VSS. Processors with read and write strobes can be supported by tying CPUMODE0 to VDD, and connecting the write strobe to STRB1_N and the read strobe to STRB2_N.

The CPUMODE2 pin is reserved and should be tied to VDD.

The CPUMODE3 pin, when tied to VDD, selects a mode of operation that can be used to support processors, e.g. ARM, that require the read strobe to be sampled on the falling edge of CPUCLK and the write strobe to be sampled on the rising edge.

The PM7800 can be configured to provide additional hold time from CPUCLK before negating WAIT_N - this may be required to support processors, e.g. ARM, that require longer hold times on WAIT_N. CPUMODE4, when tied to VDD, selects additional hold time.

The following table describes how to connect the PM7800 to a C5410 processor.

Table 5-1 Connection to C54xx Processor

Signal (PM7800)	Connect To (TMS320C5410)	Description
MSTRB#	STRB1_N	This signal is the active-low program memory access strobe.
R/W#	STRB2_N	This signal is the Read/Write# cycle type indication.
PS#	CS0_N.	This signal is the program space access strobe.
A22 -> A21	CS1_N, and CS2.	Two of the upper address bits can be connected to the two other chip select inputs to provide basic address decoding of the program memory space.
READY	WAIT_N	When the WAIT_N signal is asserted low, the TI chip extends the current cycle until the WAIT_N signal negated to a high state.
INTx	IRQ_N	INTx. is one of the four interrupt inputs.
A[17:00]	ADR[17:00]	The TI bus is word addressed, as is the PM7800.
D[15:00]	DAT[15:00]	Data is connected straight across.
CLKOUT	CPUCLK	The master clock output for the TI processor. Maximum CPU clock frequency 125 MHz.
CPUMODE0	0	The CPU signalling style is access strobe and read/write# indication.
CPUMODE1	1	The WAIT circuit forces the internal version of the CPU write strobe inactive for a programmable number of CPUCLK cycles.
CPUMODE2	1	Reserved.
CPUMODE3	0	All inputs sampled on rising edge of CPUCLK.
CPUMODE4	0	No additional delay on WAIT_N.

5.1.1 Functional Timing

The following timing diagrams give functional timing for the CPU interface. Because this interface is asynchronous to DCLK, AC timing parameters (t_{is}, t_{ih}, t_{oe}, t_{op}, t_{oz}, t_w) are shown here for clarity but are defined in *AC Timing* section 9.

Figure 5-2 Read Cycle (CPUMODE0 = 0)

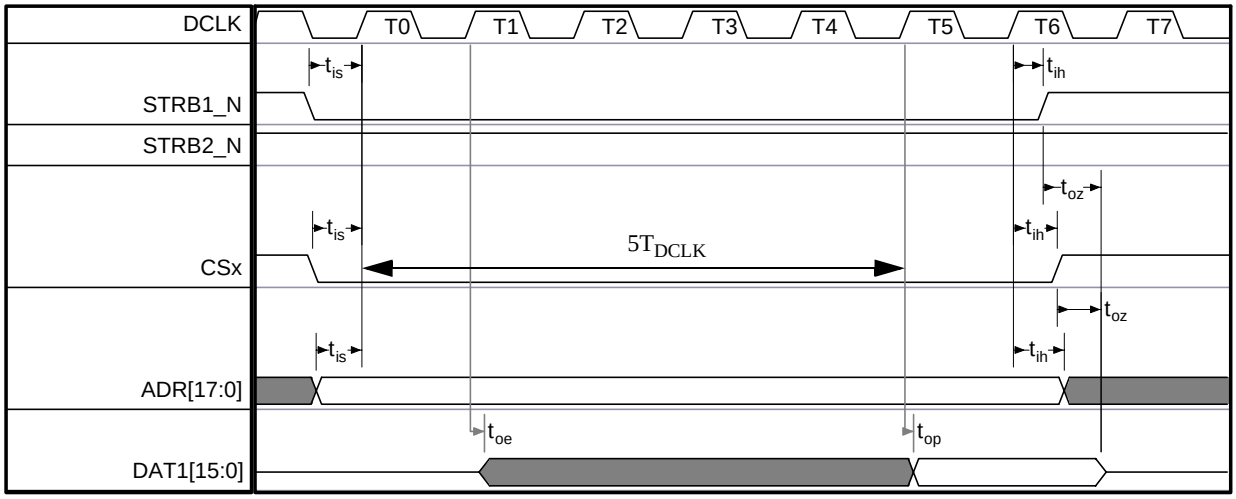


Figure 5-3 Read Cycle (CPUMODE0 = 1)

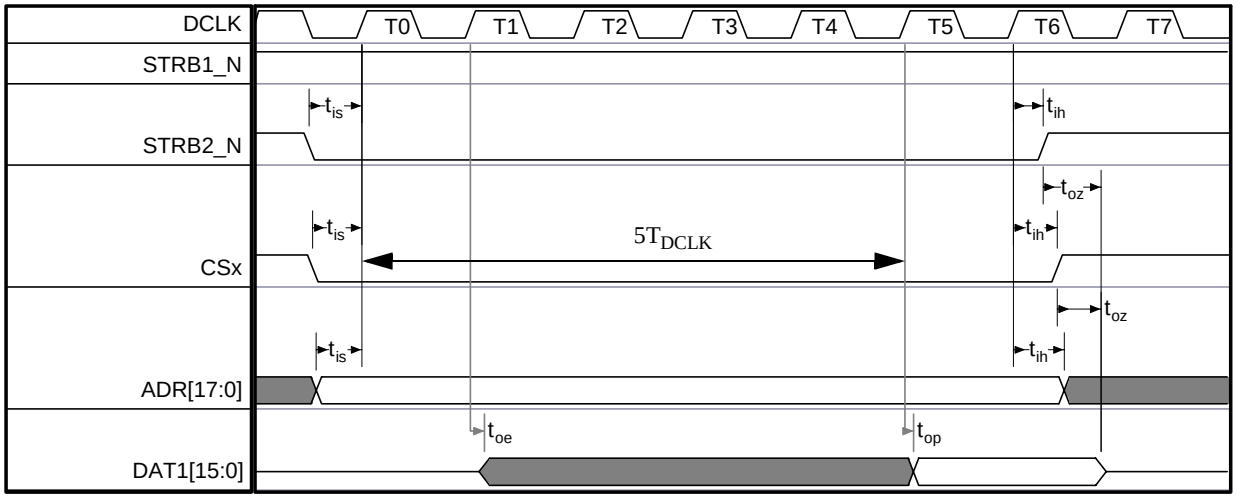


Figure 5-4 Back-to-Back Read Cycles (CPUMODE0 = 0)

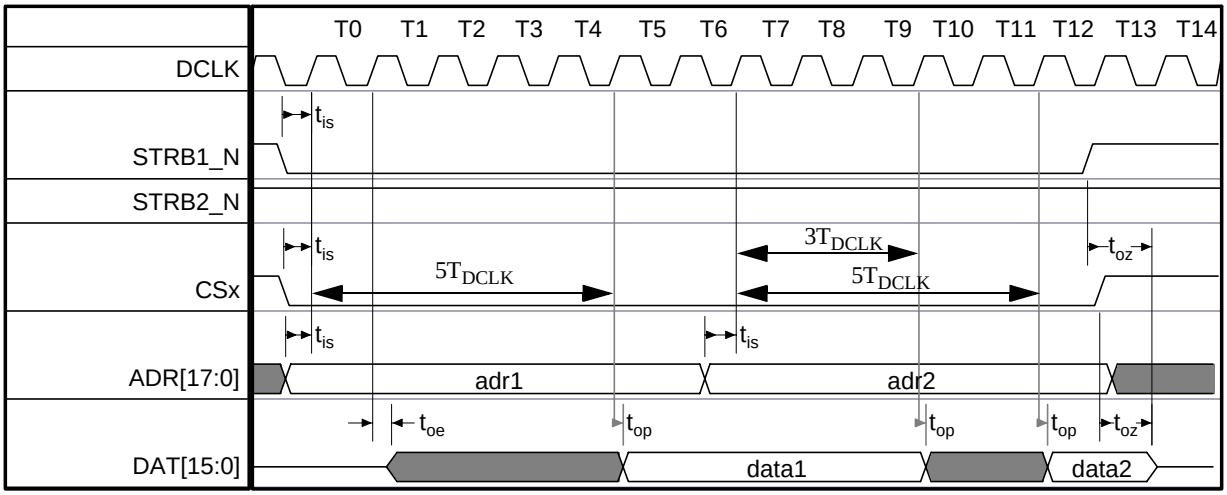


Figure 5-5 Back-to-Back Read Cycles (CPUMODE0 = 1)

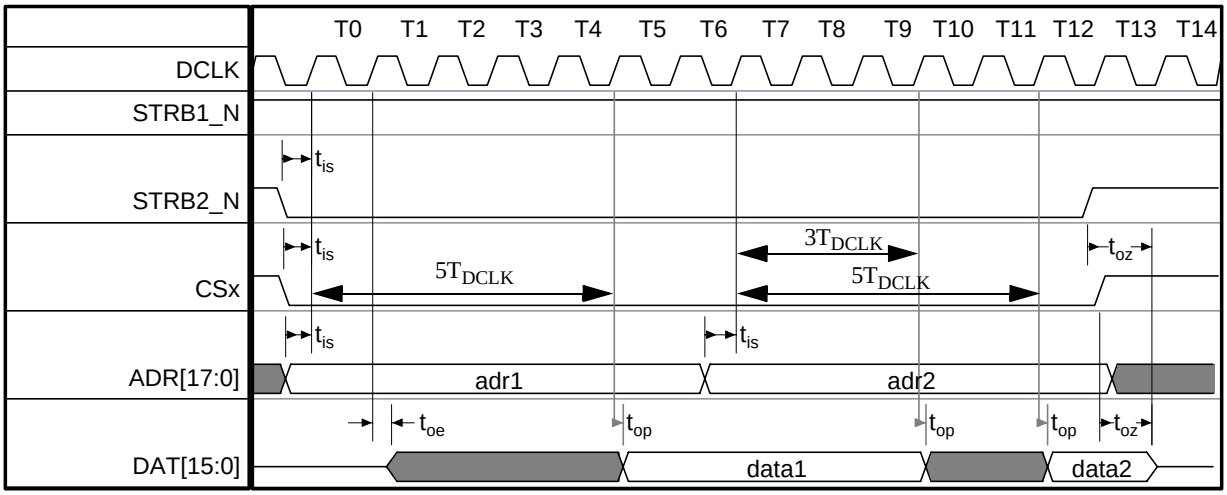


Figure 5-6 Back-to-Back Write Cycles (CPUMODE0 = 0)

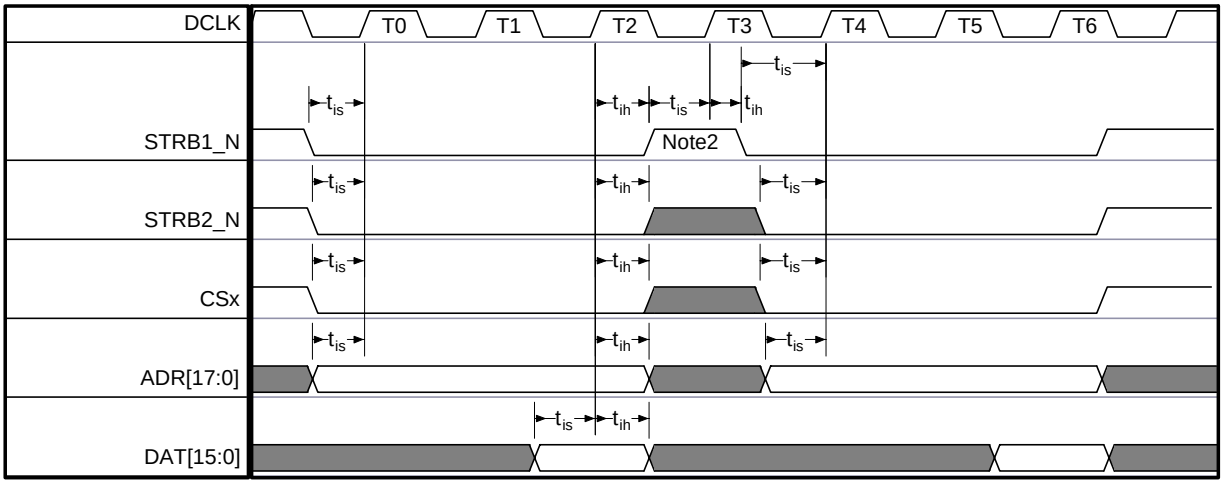
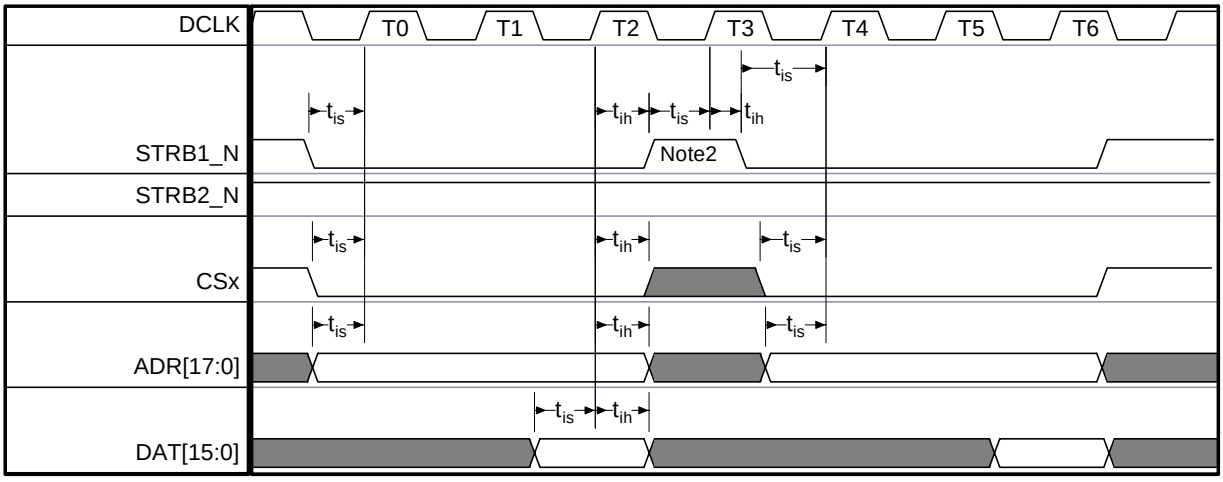


Figure 5-7 Back-to-Back Write Cycles (CPUMODE0 = 1)



Note:

1. The system designer need not worry about violating t_{is} and t_{ih} - all inputs are treated as asynchronous signals. Setup and hold times are specified here to identify the clock edge on which the signal is sampled.
2. If deassertion of the strobe cannot be detected by DCLK, CPUMODE1 must be tied to VDD to program the PM7800 to temporarily deactivate the internal write strobe - See *Functional Description* section 5. .

Figure 5-8 WAIT_N Timing - Reads (CPUMODE0 = 0)

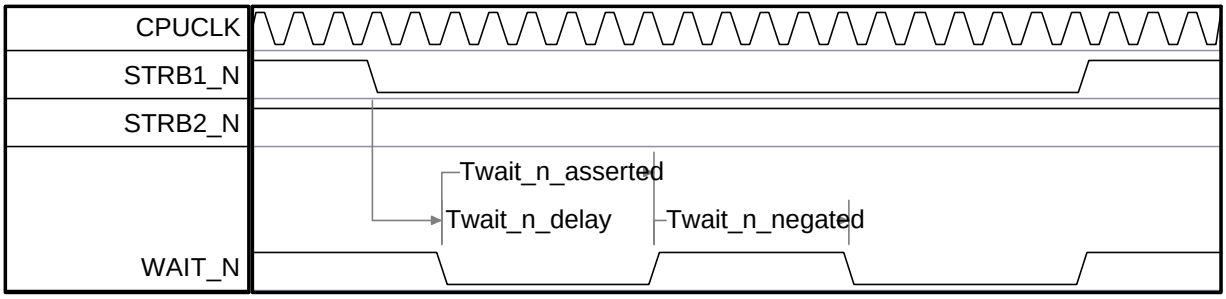


Figure 5-9 WAIT_N Timing - Reads (CPUMODE0 = 1)

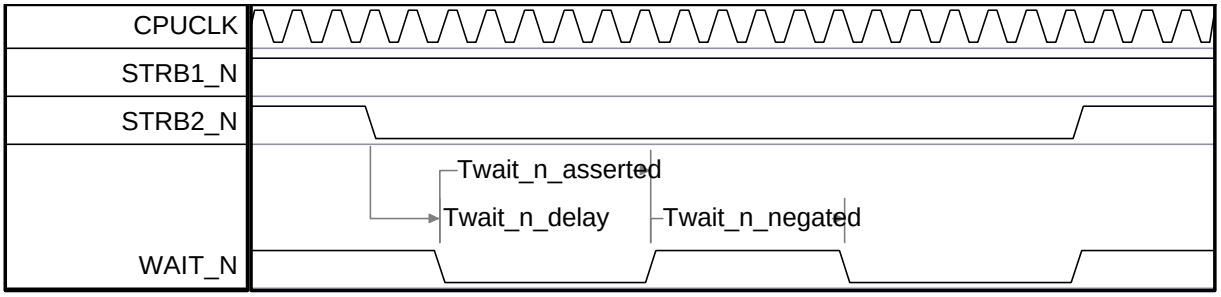


Figure 5-10 WAIT_N Timing - Writes (CPUMODE0 = 0)

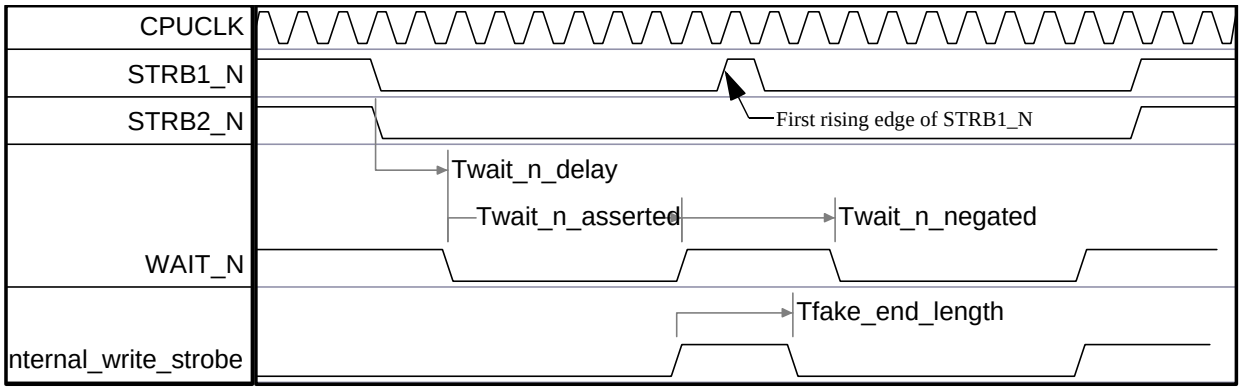


Figure 5-11 WAIT_N Timing - Writes (CPUMODE0 = 1)

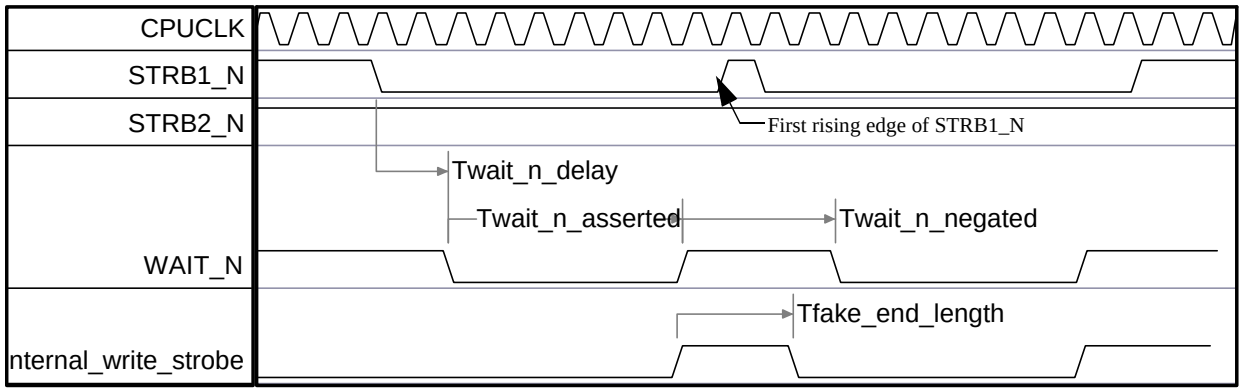


Table 5-2 Functional Timing for WAIT_N

Parameter	min	Description
$T_{wait\_n\_delay}$	$2T_{CPUCLK}$	STRBx to WAIT_N delay
$T_{wait\_n\_asserted}$	(Assert_Wait_Cycles[6:0] + 1)T_{CPUCLK}	WAIT_N pulse width
$T_{wait\_n\_negated}$	(Negate_Wait_Cycles[3:0] + 1) T_{CPUCLK}	WAIT_N negated width
$T_{fake\_end\_length}$	(Fake_End_Length[3:0] + 1) T_{CPUCLK}	Internal fake_end length for back-to-back writes.

5.1.2 Programming the WAIT_N Register

The following are guidelines for programming the WAIT_N register:

Table 5-3 Programming the WAIT_N Register

Formula	Comment
Assert_Wait_Cycles = round-up {6Tdclk /Tcpuclk -1}	Must satisfy PM7800 read-access time of 6 dclk periods
Negate_Wait_Cycles = round-up {3 + 32 / T_{cpuclk}}	Must be long enough to allow the C54 to end the current cycle, and to flush the cycle_active out of the cpuclk synchronizers.
internal_c54_wait > **Negate_Wait_Cycles** where internal_c54_wait is the number of wait cycles programmed in the C54xx processor.	The C54 must be programmed such that it will not start another cycle before WAIT_N can be reasserted.
Fake_End_Length = round-up{max{[(T_{dclk} + 2ns) / T_{cpuclk} - 1] : [1.5 + 32 / T_{cpuclk}]}}	Must be long enough for the PM7800 to sample its internal fake_wait_inactive with dclk. Must also be long enough that fake_wait_inactive is held until past the rising edge of STRB1_N (shown in Figure 5-10 on page 22 and Figure 5-11 on page 23).

5.2 Data Interfaces

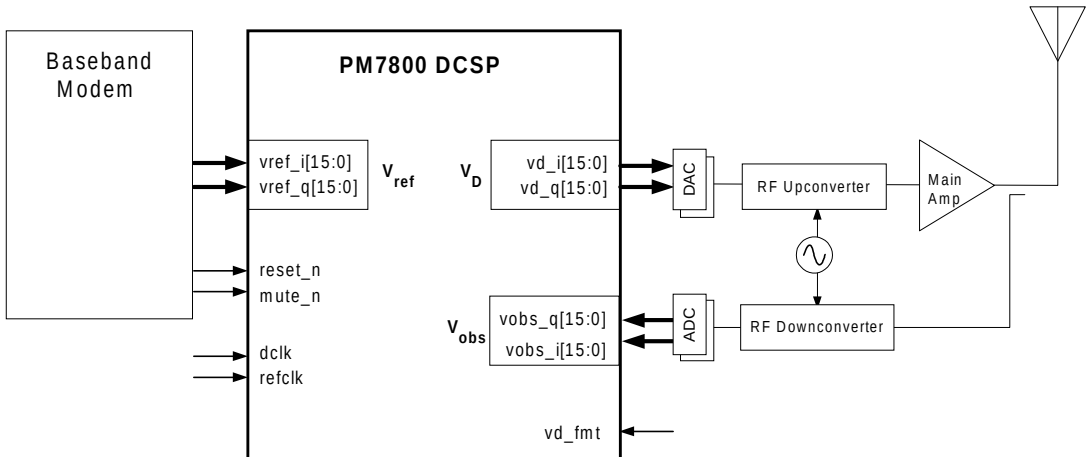


Figure 5-12 Data Interfaces

5.2.1 VREF Interface and FIFO

The VREF interface is used to input the reference signal in either offset-binary or two's-complement notation. In the case of offset-binary format, the VREF_Data_Format bit in the Mode register must be set to zero to instruct the PM7800 to convert the data to two's-complement format which is used throughout the chip. There are three modes of operation for the VREF interface:

1. VREF is clocked in by REFCLK and upsampled to the rate of the PM7800 clock, DCLK, where $f_{DCLK} = N * f_{REFCLK}$. In this case, a FIFO is used to handle the skew between REFCLK and DCLK, and to provide a steady supply of data to the interpolator. Set the Bypass_FIFO and Interpolator_Bypass bits to zero (default) in this case.
2. VREF is clocked in by REFCLK which is at the same rate as the PM7800 clock, DCLK, that is, $f_{DCLK} = f_{REFCLK}$. In this case, a FIFO is used to handle the skew between REFCLK and DCLK. The interpolator can be bypassed to reduce latency. Set the Bypass_FIFO bit to zero (default) and the Interpolator_Bypass bit to one in this case.
3. VREF is at the same rate as and synchronous to DCLK. In this case, the FIFO and interpolator can be bypassed to reduce system latency by setting the Bypass_FIFO and Interpolator_Bypass bits to one in the Mode register. The REFCLK input is ignored and should be connected to VSS.

5.2.2 VOBS Interface

The VOBS interface is used to input the observed signal from the ADC in either offset-binary or two's-complement notation. In the case of offset-binary format, the VOBS_Data_Format bit in the Mode register must be set to zero to instruct the PM7800 to convert the data to two's-complement format.

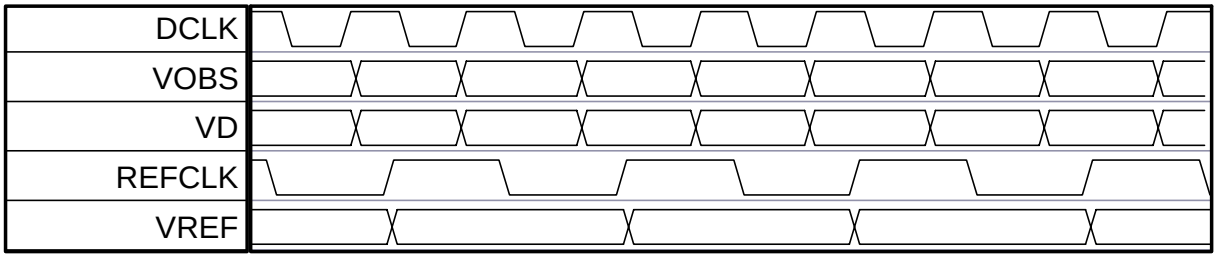
5.2.3 VD Interface

The VD interface is used to output the predistorted signal to the DAC in either offset-binary or two's-complement notation. The VD_FMT pin selects offset-binary format when low, and two's-complement format when high.

5.2.4 Dual-Clock System

Use the dual-clock system when VREF is synchronous to REFCLK (usually when VREF is upsampled internally to the DCLK rate). T_{REFCLK} must be an exact integer multiple of T_{DCLK}, i.e. $T_{REFCLK} = nT_{DCLK}$, where n = 1,2,...,10. REFCLK must be created from DCLK. In this case, a FIFO and interpolator are used to upsample VREF to the DCLK rate. REFCLK must be created from DCLK such that there is no varying phase shift that may cause under-run or over-run, i.e. skew may exist between DCLK and REFCLK, but this skew must not vary over time by more than one DCLK period.

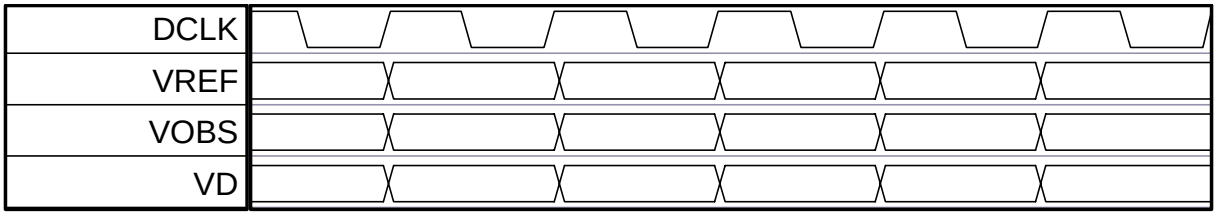
Figure 5-13 Dual-Clock System timing



5.2.5 Single-Clock System

Use the single-clock system when all signals are synchronous to DCLK and the data-rate of VREF is the same as the output data-rate of VD. In this case, connect the REFCLK pin to VSS, and bypass the FIFO and interpolator to reduce latency through the chip.

Figure 5-14 Single-Clock System timing



5.3 Serial Interface

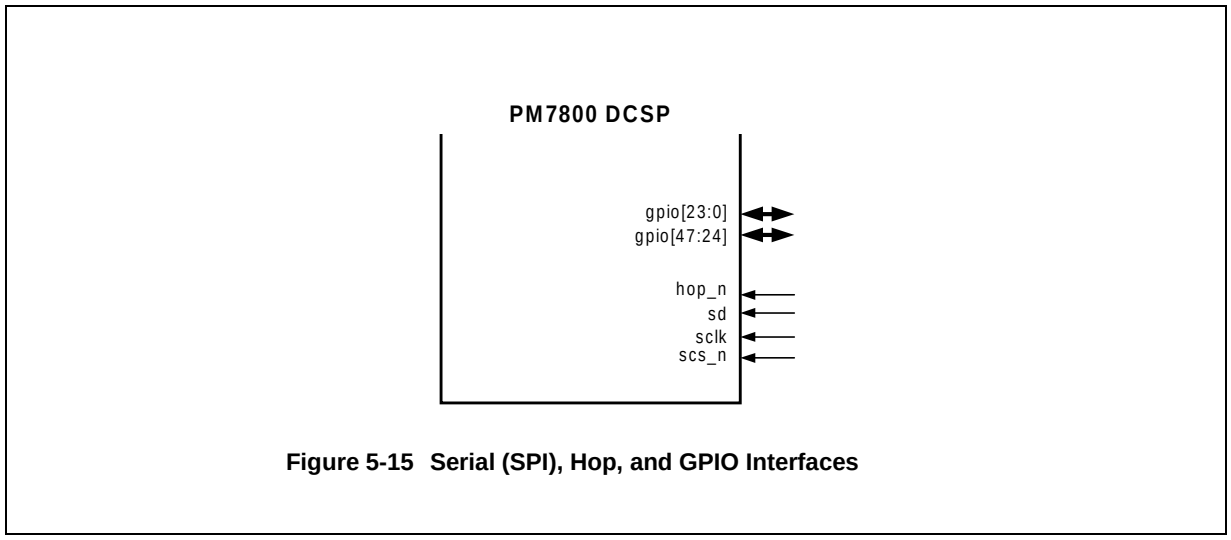


Figure 5-15 Serial (SPI), Hop, and GPIO Interfaces

The serial interface is used to input the next_power, next_carrier values, hop_stream command, and general-purpose serial_word register. Also included in this interface is the HOP_N pin which can be used to delay the effect of the hop_stream to a more precise time.

The PM7800 supports the SPI specification included in the M68HC11 Reference Manual (section 8). Only the modes where CPHA = 1 is supported. CPOL may be 0 or 1, and the spi_clock_polarity bit of the serial_mode register (0x0_0013 bit 2) is used to select the active clock edge. The PM7800 does not have a data output pin (MISO). The PM7800's SCS_N pin is equivalent to the /SS pin, and it can be tied to ground to reduce system pin requirements.

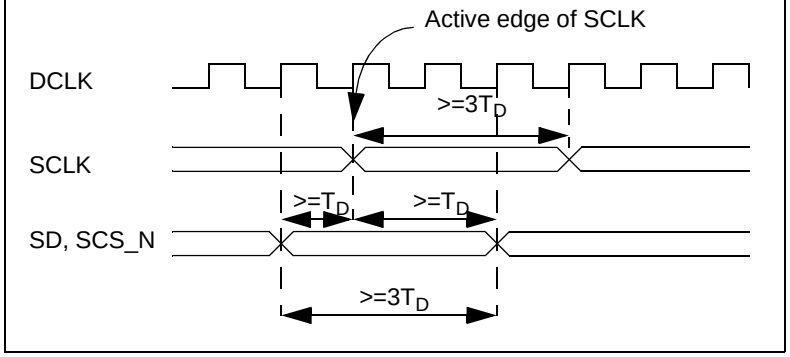
A transfer can be reset by deactivating SCS_N. If SCS_N is tied low (always active) then another method must be used to reset the state machine. One way is to disable the serial interface by writing a zero to the serial_interface_enable bit in the Serial_Mode register. Another way is to set reg_serial_address[4:0] = 00000b (this is the default setting) and stream twenty-four 1's into the serial port: at some stage during the stream of 1's, the idle state will be reached and the interface

will be held waiting for the first 0 of the address. Note that if the transfer is aborted after some data has been sent, that data will already be shifted into the respective data register.

Relative timing of the SPI input signals is shown in Figure 5-16. Note that SCLK, SD, SCS_N can be asynchronous to DCLK: Figure 5-16 shows timing for the inputs relative to the DCLK edges on which they are detected. Notes:

- SCLK, SD, SCS_N are debounced using DCLK, therefore they must be present for two consecutive DCLK edges to be properly detected. It is recommended that the minimum pulse-width be 3 T_{DCLK}, i.e. $T_{SCLK} >= 6T_{DCLK}$.
- SD and SCS_N are sampled by DCLK on the active edge of SCLK. Therefore, because of the asynchronous relationship, it is recommended that SD and SCS_N be valid for one DCLK either side of the active SCLK edge.

Figure 5-16 Relative Timing of SPI Inputs



5.3.1 Serial Operation

All signal pins in the serial interface are synchronized to dclk and debounced to reject glitches shorter than a dclk period - see *AC Timing*. There are four serial streams recognized by this interface:

1. carrier_stream consists of three bytes as follows:

- five address bits corresponding to reg_serial_address[4:0]
- two bits = 00b, that indicate a carrier_stream
- a r/w bit that must be zero
- seven don’t-care bits

• nine bits of the next_carrier value - in the order of msb to lsb

Figure 5-17 SPI carrier_stream (spi_clock_polarity = 0, falling edge of SCLK is active)

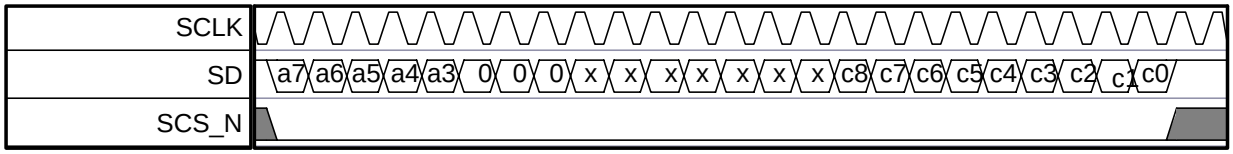
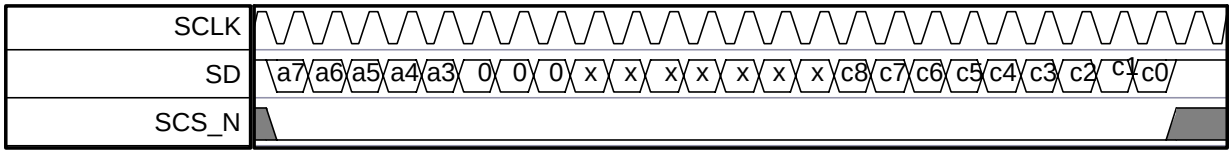


Figure 5-18 SPI carrier_stream (spi_clock_polarity = 1, rising edge of SCLK is active)



2. power_stream consists of two bytes as follows:

• five address bits corresponding to reg_serial_address[4:0]
• two bits = 01b, that indicate a power_stream
• a r/w bit that must be zero
• eight bits of the next_power value - in the order of msb to lsb

Figure 5-19 SPI power_stream (spi_clock_polarity = 0, falling edge of SCLK is active)

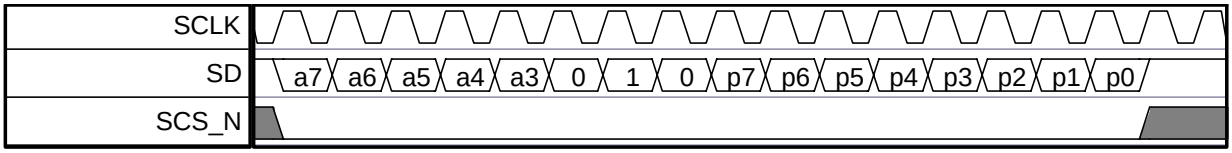
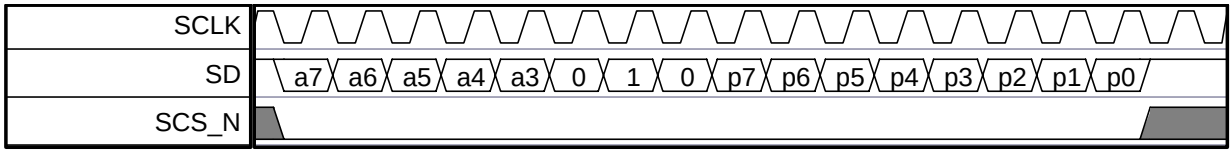


Figure 5-20 SPI power_stream (spi_clock_polarity = 1, rising edge of SCLK is active)



3. word_stream consists of three bytes as follows:

- five address bits corresponding to reg_serial_address[4:0]
- two bits = 10b, that indicate a word_stream
- a r/w bit that must be zero
- sixteen bits of the serial_word value - in the order of msb to lsb

Figure 5-21 SPI word_stream (spi_clock_polarity = 0, falling edge of SCLK is active)

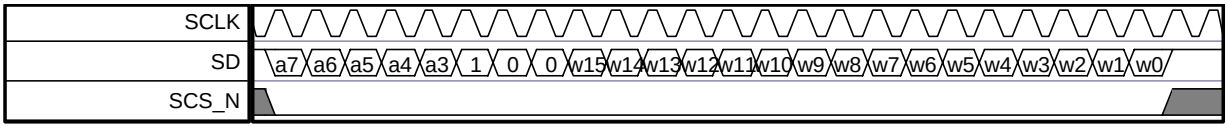
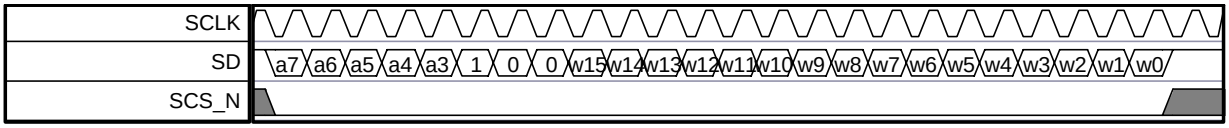


Figure 5-22 SPI word_stream (spi_clock_polarity = 1, rising edge of SCLK is active)



4. hop_stream consists of one byte as follows:

- five address bits corresponding to reg_serial_address[4:0]
- two bits = 11b, that indicate a hop_stream
- a r/w bit that must be zero

Figure 5-23 SPI hop_stream (spi_clock_polarity = 0, falling edge of SCLK is active)

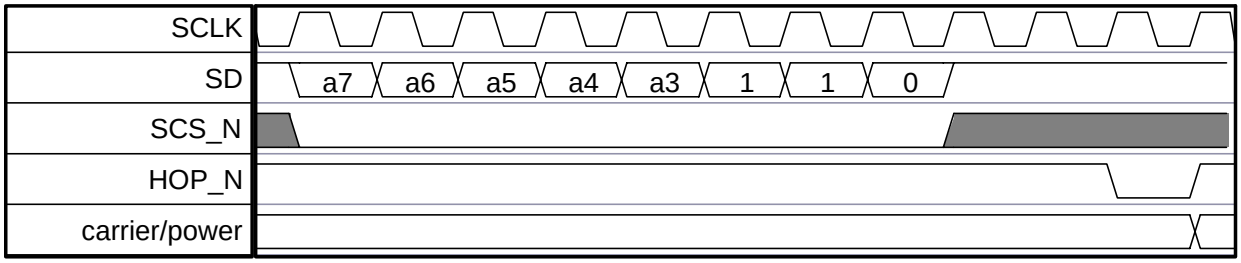


Figure 5-23 shows how HOP_N can be used to delay the new carrier/power values.

Figure 5-24 SPI hop_stream (spi_clock_polarity = 1, rising edge of SCLK is active)

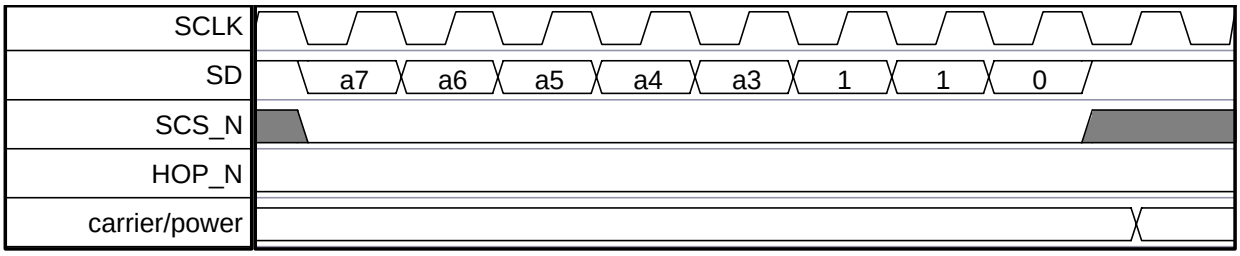


Figure 5-24 shows a case where HOP_N is tied active so that the new carrier/power values take effect immediately after the Hop_Stream.

5.4 Hop Generation

The hop_state_machine generates the Hop pulse that propagates throughout the chip and activates the new carrier and power values. The Hop pulse is generated upon a write to the Hop bit in the Control register, or on the later of HOP_N and a Hop_Stream. There are three methods of operation:

1. The Hop bit in the Control register controls the activation point.
   The CPU writes to the next_carrier and/or next_power registers, then writes 1 to the Hop bit in the Control register.
2. HOP_N controls the activation point.
   Carrier_Stream and Power_Stream are immediately followed by a Hop_Stream. Then HOP_N is applied so that the new values are activated at the correct time. This is a likely scenario for Multi_Band_Mode (e.g. single-carrier EDGE systems). This is illustrated in Figure 5-23.
3. Hop_Stream controls the activation point.
   HOP_N is held active by connecting it to VSS, thereby reducing system pin requirements. The Hop_Stream is used to activated new values at the correct time. This is illustrated in Figure 5-24. Note that the final bit of the Hop_Stream can be delayed to the desired time.

5.4.1 Power and Carrier Values and Hop

There are two parameters that are used throughout the device and must be explained here:

- *Power* - Power selection by base station. This is an 8-bit integer value and its interpretation is configured by the Power_Attenuation_Table, the Power_Correction_Gain_Table, and the Carrier_Power_Map.
- *Carrier* - Carrier (or frequency band) selection by base station. This is a 9-bit integer value and its interpretation is configured by the Carrier_Correction_Gain_Table and the Carrier_Power_Map.

The Power and Carrier values can be input through the serial interface, or directly programmed by the CPU. In either case, the values entered are Next_Power and Next_Carrier, and they become active Power and Carrier values after the next Hop occurs (see Section 5.4, *Hop Generation*, on page 30).

The new Power value propagates to the Power_Attenuation_Table after a delay controlled by the Power_Attenuator_Delay register. The new Power and Carrier values propagate to all other areas of the device after a delay controlled by the Power_Correction_Delay register. These two programmable delays are designed to allow some control over the effects of power changes throughout the system.

5.4.2 Bug that affects Power_Attenuator_Delay and Power_Correction_Delay

The Power_Attenuator_Delay and Power_Correction_Delay circuits were intended to operate so that:

1. if Power_Attenuator_Delay > Power_Correction_Delay, the Power_Attenuator output on GPIO[15:0] and the pulse output on GPIO17 are delayed with respect to the power_correction_gain value applied inside the chip.
2. if Power_Attenuator_Delay < Power_Correction_Delay, the Power_Attenuator output on GPIO[15:0] and the pulse output on GPIO17 are earlier than the power_correction_gain value applied inside the chip.

The bug has two ramifications:

- The relationship between these two delays must be such that Power_Attenuator_Delay >= Power_Correction_Delay + 1, i.e. (2) above is not possible. We recommend that Power_Correction_Delay is programmed to its default state of zero.
- The Power_Attenuator output on GPIO[15:0] is not delayed but the pulse on GPIO17 is. Therefore, the external attenuator device must use the pulse to latch the new Power_Attenuator value.

See Section 5.5.1, *Power Attenuator Outputs*, on page 33.

5.5 GPIO Pins

These pins are general-purpose input-outputs. They are controlled and monitored through the PM7800 register set. Some of the pins have secondary functions.

- GPIO[15:0] can be configured to output the value from the Power_Attenuator_Table.
- GPIO16 can be configured to output an active-low pulse of $32T_{DCLK}$ duration when the Watchdog Timer expires.
- GPIO17 can be configured to output a $5T_{DCLK}$ pulse when a Hop occurs. The pulse is positioned so that its leading edge is concurrent with changes in the value of the Power_Attenuator_Table output at GPIO[15:0] - see *Functional Timing*. The polarity of the pulse is selected by the Atten_Pulse_Polarity bit in the Mode register.
- GPIO[23:18] have no secondary function.
- GPIO[27:24] can be configured as falling-edge triggered interrupt sources. The interrupts are enabled using the GPIO_Interrupt_Enable bits in the Interrupt_Enable2 register. The interrupts are monitored and cleared in the Interrupt_Status_Clear register.
- GPIO[31:28] can be configured as rising-edge triggered interrupt sources. The interrupts are enabled using the GPIO_Interrupt_Enable bits in the Interrupt_Enable2 register. The interrupts are monitored and cleared in the Interrupt_Status_Clear register.
- GPIO[47:32] have no secondary function.

5.5.1 Power Attenuator Outputs

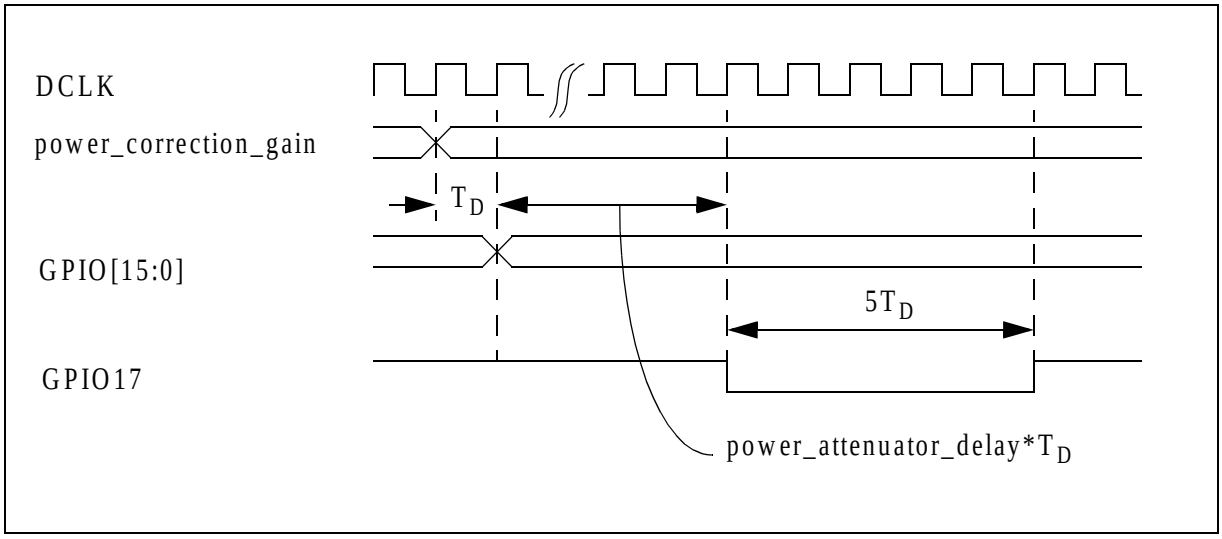


Figure 5-25 Power Attenuator Outputs - shown with active-low pulse on GPIO17

5.5.2 Watchdog Timer Output

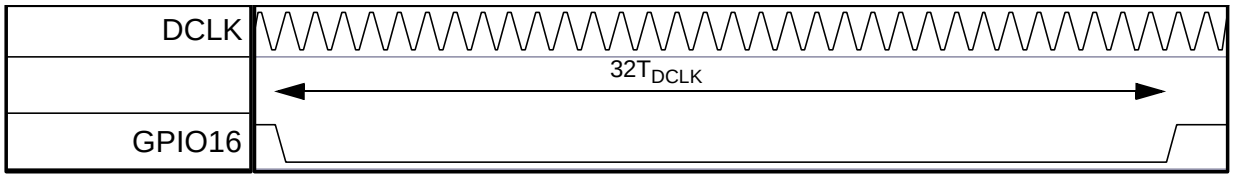


Figure 5-26 Watchdog Timer Output

5.6 JTAG Test Access Port

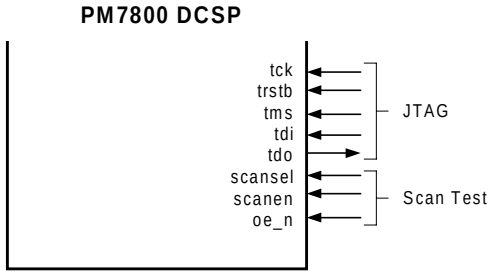


Figure 5-27 JTAG and Scan Test Interfaces

The JTAG Test Access Port block provides JTAG support for boundary scan. The standard JTAG EXTEST, SAMPLE, BYPASS, IDCODE and STCTEST instructions are supported. The PM7800 identification code is 178000CD hexadecimal.

The JTAG port is described in detail in *Test Features* section 6.

6 Test Features

The PM7800 has four test modes: Full Scan, RAMBIST, OE_N Pin, and JTAG Boundary Scan.

6.1 Full Scan

The core logic is tested using full scan. The following table describes the full-scan interface.

Table 6-1 Full-Scan Interface

Full-Scan Signal	Pin	Description
SCANSEL	SCANSEL	Selects full-scan mode: selects RESET_N to directly control all flop sets and resets; disables RAM BIST; configures the SCANIN and SCANOUT IO pads.
SCANEN	SCANEN	Enables the scan chain.
SCANIN[55:32]	GPIO[47:24]	SCANIN bus.
SCANIN[31:16]	VD_Q[15:0]	SCANIN bus.
SCANIN[15:0]	VD_I[15:0]	SCANIN bus.
SCANOUT[55:32]	GPIO[23:0]	SCANOUT bus.
SCANOUT[31:16]	VREF_Q[15:0]	SCANOUT bus.
SCANOUT[15:0]	VREF_I[15:0]	SCANOUT bus.

6.2 RAMBIST

These registers are for RAMBIST. They must remain in their default states during regular operation.

0x0_00F0	BIST_Mode				
Name	**Type**	**Size**	**Bits**	**Description**	**Reset**
BISTTEST[7:0]	RW	8	7:0	BIST test pattern bus which is replicated over the RAM data bus. This bus is the seed used by the BIST to generate patterns during the bist test. It is also used in exercising the backdoor feature by providing a method to shift in/out data.	0
BISTEN	RW	1	8	This bit activates BIST testing.	0

0x0_00F0	BIST_Mode				
Name	**Type**	**Size**	**Bits**	**Description**	**Reset**
BISTSIDE	RW	1	9	BISTSIDE is used to select between port A and port B when performing a BIST test. Since Predistorter1 contains dual-port RAM, two BIST tests must be run, first with this bit = 0, and then with this bit = 1. 0: port A 1: port B	0
BISTMODE[2:0]	RW	3	12:10	BISTMODE configures the BIST in several different modes: 000: bistmode_hold 001: bistmode_shift 010: bistmode_bist 011: bistmode_run 100: bistmode_reset 101 ~ 111: Reserved.	0

0x0_00F2	BIST_Result1				
Name	**Type**	**Size**	**Bits**	**Description**	**Reset**
BISTEND	RO	1	0	The BISTEND bit indicates the BIST test sequence has completed. When the BIST test is complete, bistend transitions from low to high.	0
BISTRESULT	RO	1	1	The BISTRESULT bit indicates whether all RAMs have passed or failed the BIST test. If BISTRESULT transitions from high to low, a RAM has experienced a compare failure (i.e. at least one of the bisterror bits have been asserted). Once a compare failure is detected, BISTRESULT transitions to a low state, and stays in that state for the remainder of the test.	0
BISTERROR[13:0]	RO	14	15:2	BISTERROR bus to indicate compare failure on a RAM word. Each bit corresponds to one of the RAMs in the device. The BISTERROR bit associated with each RAM toggles high each time a compare failure is detected.	0

0x0_00F4	BIST_Result2				
Name	**Type**	**Size**	**Bits**	**Description**	**Reset**
BISTERROR[20:14]	RO	7	6:0	BISTERROR bus to indicate compare failure on a RAM word. Each bit corresponds to one of the RAMs in the device. The BISTERROR bit associated with each RAM toggles high each time a compare failure is detected.	0

6.3 OE_N Pin

The OE_N pin, when high, deactivates all pins except TDO. TDO is a tri-state output driven during boundary-scan testing only (see below); applying an active-low pulse to TRSTB ensures that TDO is inactive.

6.4 JTAG Boundary Scan, IEEE 1149.1

The PM7800 supports the IEEE Boundary Scan Specification as described in the IEEE 1149.1 standards. The PM7800 JTAG Test Access Port (TAP) allows access to the TAP controller and the four TAP registers: instruction, bypass, device identification and boundary scan. Using the TAP, device input logic levels can be read, device outputs can be forced, the device can be identified and the device scan path can be bypassed.

The Test Access Port (TAP) consists of the five standard pins, TRSTB, TCK, TMS, TDI and TDO used to control the TAP controller and the boundary scan registers. The TRSTB input is the active-low reset signal used to reset the TAP controller. TCK is the test clock used to sample data on input, TDI and to output data on output, TDO. The TMS input is used to direct the TAP controller through its states.

The boundary-can architecture is shown below.

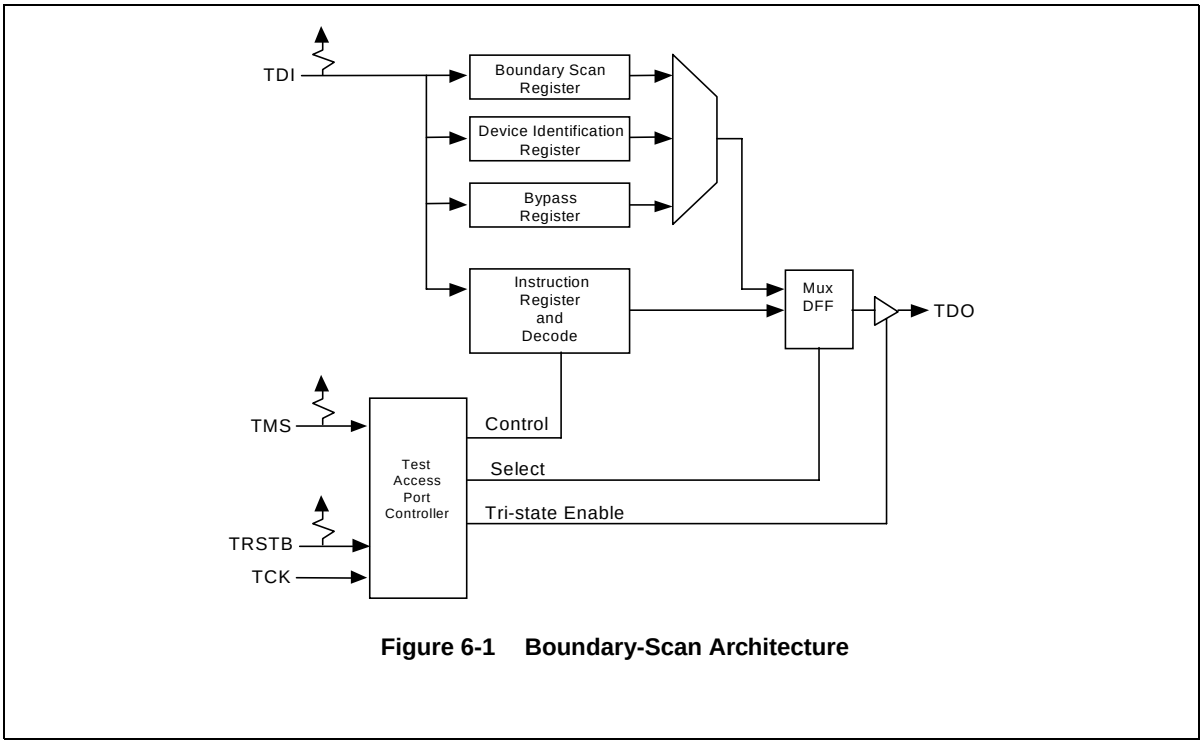


Figure 6-1 Boundary-Scan Architecture

The boundary scan architecture consists of a TAP controller, an instruction register with instruction decode, a bypass register, a device identification register and a boundary scan register. The TAP controller interprets the TMS input and generates control signals to load the instruction and data registers. The instruction register with instruction decode block is used to select the test to be executed and/or the register to be accessed. The bypass register offers a single-bit delay from primary input, TDI to primary output, TDO. The device identification register contains the device identification code.

The boundary scan register allows testing of board inter-connectivity. The boundary scan register consists of a shift register place in series with device inputs and outputs. Using the boundary scan register, all digital inputs can be sampled and shifted out on primary output, TDO. In addition, patterns can be shifted in on primary input, TDI and forced onto all digital outputs.

6.4.1 TAP Controller

The TAP controller is a synchronous finite state machine clocked by the rising edge of primary input, TCK. All state transitions are controlled using primary input, TMS. The finite state machine is described below.

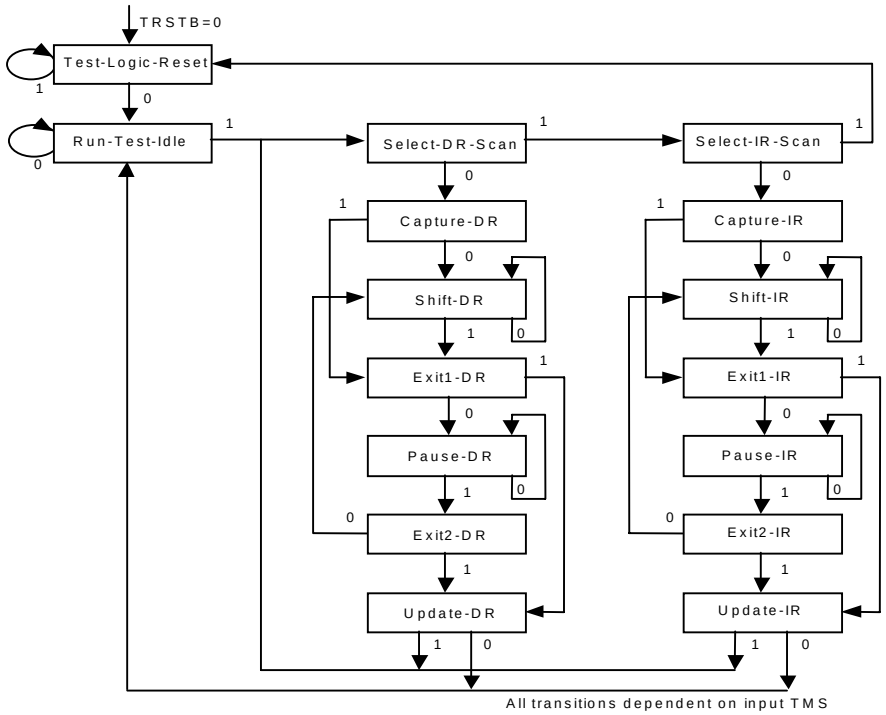


Figure 6-2 TAP Controller Finite State Machine

Test-Logic-Reset

The test logic reset state is used to disable the TAP logic when the device is in normal mode operation. The state is entered asynchronously by asserting input, TRSTB. The state is entered synchronously regardless of the current TAP controller state by forcing input, TMS high for 5 TCK clock cycles. While in this state, the instruction register is set to the IDCODE instruction.

Run-Test-Idle

The run test/idle state is used to execute tests.

Capture-DR

The capture data register state is used to load parallel data into the test data registers selected by the current instruction. If the selected register does not allow parallel loads or no loading is required by the current instruction, the test register maintains its value. Loading occurs on the rising edge of TCK.

Shift-DR

The shift data register state is used to shift the selected test data registers by one stage. Shifting is from MSB to LSB and occurs on the rising edge of TCK.

Update-DR

The update data register state is used to load a test register's parallel output latch. In general, the output latches are used to control the device. For example, for the EXTEST instruction, the boundary scan test register's parallel output latches are used to control the device's outputs. The parallel output latches are updated on the falling edge of TCK.

Capture-IR

The capture instruction register state is used to load the instruction register with a fixed instruction. The load occurs on the rising edge of TCK.

Shift-IR

The shift instruction register state is used to shift both the instruction register and the selected test data registers by one stage. Shifting is from MSB to LSB and occurs on the rising edge of TCK.

Update-IR

The update instruction register state is used to load a new instruction into the instruction register. The new instruction must be scanned in using the Shift-IR state. The load occurs on the falling edge of TCK.

The Pause-DR and Pause-IR states are provided to allow shifting through the test data and/or instruction registers to be momentarily paused.

6.4.2 Registers

6.4.2.1 Instruction Register

Length = 3 bits

Table 6-2 Instruction Register

Instructions	Selected Register	Instruction Codes, IR[2:0]
EXTEST	Boundary Scan	000
IDCODE	Identification	001
SAMPLE	Boundary Scan	010
BYPASS	Bypass	011
BYPASS	Bypass	100
STCTEST	Boundary Scan	101
BYPASS	Bypass	110
BYPASS	Bypass	111

6.4.2.2 Identification register

Length = 32 bits

Table 6-3 Identification Register

Length	32 bits
Version number	1h
Part Number	7800h
Manufacturer's identification code	0CDh
Device identification	178000CDh

6.4.2.3 Boundary Scan Register

```
Name                             Register Bit  Cell Type  Device ID
-------------------------------- ------------  ---------  ---------
OEB_GPIO7                         333           OUT_CELL   L
```

GPIO7	332	IO_CELL	L
OEB_GPIO6	331	OUT_CELL	L
GPIO6	330	IO_CELL	L
OEB_GPIO4	329	OUT_CELL	L
GPIO4	328	IO_CELL	H
OEB_GPIO5	327	OUT_CELL	H
GPIO5	326	IO_CELL	H
OEB_GPIO2	325	OUT_CELL	H
GPIO2	324	IO_CELL	L
OEB_GPIO1	323	OUT_CELL	L
GPIO1	322	IO_CELL	L
OEB_GPIO3	321	OUT_CELL	L
GPIO3	320	IO_CELL	L
OEB_GPIO0	319	OUT_CELL	L
GPIO0	318	IO_CELL	L
OEB_VD_I13	317	OUT_CELL	L
VD_I13	316	IO_CELL	L
OEB_VD_I10	315	OUT_CELL	L
VD_I10	314	IO_CELL	L
OEB_VD_I14	313	OUT_CELL	L
VD_I14	312	IO_CELL	L
OEB_VD_I11	311	OUT_CELL	L
VD_I11	310	IO_CELL	L
OEB_VD_I15	309	OUT_CELL	H
VD_I15	308	IO_CELL	H
OEB_VD_I12	307	OUT_CELL	L
VD_I12	306	IO_CELL	L
OEB_VD_I7	305	OUT_CELL	H
VD_I7	304	IO_CELL	H
OEB_VD_I8	303	OUT_CELL	L
VD_I8	302	IO_CELL	H
OEB_VD_I4	301	OUT_CELL	-
VD_I4	300	IO_CELL	-
OEB_VD_I9	299	OUT_CELL	-
VD_I9	298	IO_CELL	-
OEB_VD_I5	297	OUT_CELL	-
VD_I5	296	IO_CELL	-
OEB_VD_I6	295	OUT_CELL	-
VD_I6	294	IO_CELL	-
OEB_VD_I0	293	OUT_CELL	-
VD_I0	292	IO_CELL	-
OEB_VD_I1	291	OUT_CELL	-
VD_I1	290	IO_CELL	-
OEB_VD_I2	289	OUT_CELL	-
VD_I2	288	IO_CELL	-
OEB_VD_I3	287	OUT_CELL	-
VD_I3	286	IO_CELL	-
DCLK	285	IN_CELL	-
OEB_VD_Q15	284	OUT_CELL	-
VD_Q15	283	IO_CELL	-
OEB_VD_Q14	282	OUT_CELL	-
VD_Q14	281	IO_CELL	-
OEB_VD_Q13	280	OUT_CELL	-
VD_Q13	279	IO_CELL	-
OEB_VD_Q12	278	OUT_CELL	-
VD_Q12	277	IO_CELL	-
OEB_VD_Q11	276	OUT_CELL	-
VD_Q11	275	IO_CELL	-
OEB_VD_Q10	274	OUT_CELL	-
VD_Q10	273	IO_CELL	-
OEB_VD_Q9	272	OUT_CELL	-
VD_Q9	271	IO_CELL	-
OEB_VD_Q8	270	OUT_CELL	-
VD_Q8	269	IO_CELL	-
OEB_VD_Q7	268	OUT_CELL	-
VD_Q7	267	IO_CELL	-
OEB_VD_Q5	266	OUT_CELL	-
VD_Q5	265	IO_CELL	-
OEB_VD_Q4	264	OUT_CELL	-
VD_Q4	263	IO_CELL	-
OEB_VD_Q6	262	OUT_CELL	-
VD_Q6	261	IO_CELL	-
OEB_VD_Q3	260	OUT_CELL	-
VD_Q3	259	IO_CELL	-
OEB_VD_Q1	258	OUT_CELL	-
VD_Q1	257	IO_CELL	-
OEB_VD_Q0	256	OUT_CELL	-

VD_Q0	255	IO_CELL	-
OEB_VD_Q2	254	OUT_CELL	-
VD_Q2	253	IO_CELL	-
OEB_GPIO47	252	OUT_CELL	-
GPIO47	251	IO_CELL	-
OEB_GPIO46	250	OUT_CELL	-
GPIO46	249	IO_CELL	-
OEB_GPIO45	248	OUT_CELL	-
GPIO45	247	IO_CELL	-
MUTE_N	246	IN_CELL	-
CPUMODE4	245	IN_CELL	-
CPUMODE3	244	IN_CELL	-
OEB_GPIO44	243	OUT_CELL	-
GPIO44	242	IO_CELL	-
OEB_GPIO42	241	OUT_CELL	-
GPIO42	240	IO_CELL	-
OEB_GPIO43	239	OUT_CELL	-
GPIO43	238	IO_CELL	-
ADR1	237	IN_CELL	-
ADR0	236	IN_CELL	-
ADR2	235	IN_CELL	-
ADR4	234	IN_CELL	-
ADR8	233	IN_CELL	-
ADR3	232	IN_CELL	-
ADR7	231	IN_CELL	-
ADR11	230	IN_CELL	-
ADR6	229	IN_CELL	-
ADR10	228	IN_CELL	-
ADR5	227	IN_CELL	-
ADR9	226	IN_CELL	-
ADR14	225	IN_CELL	-
ADR13	224	IN_CELL	-
ADR17	223	IN_CELL	-
ADR12	222	IN_CELL	-
ADR16	221	IN_CELL	-
ADR15	220	IN_CELL	-
CS2	219	IN_CELL	-
CS1_N	218	IN_CELL	-
CS0_N	217	IN_CELL	-
CPUCLK	216	IN_CELL	-
STRB1_N	215	IN_CELL	-
STRB2_N	214	IN_CELL	-
OEB_WAIT_N	213	OUT_CELL	-
WAIT_N	212	OUT_CELL	-
OEB_IRQ_N	211	OUT_CELL	-
IRQ_N	210	OUT_CELL	-
OEB_DAT0	209	OUT_CELL	-
DAT0	208	IO_CELL	-
OEB_DAT1	207	OUT_CELL	-
DAT1	206	IO_CELL	-
OEB_DAT2	205	OUT_CELL	-
DAT2	204	IO_CELL	-
OEB_DAT3	203	OUT_CELL	-
DAT3	202	IO_CELL	-
OEB_DAT4	201	OUT_CELL	-
DAT4	200	IO_CELL	-
OEB_DAT5	199	OUT_CELL	-
DAT5	198	IO_CELL	-
OEB_DAT6	197	OUT_CELL	-
DAT6	196	IO_CELL	-
OEB_DAT7	195	OUT_CELL	-
DAT7	194	IO_CELL	-
OEB_DAT8	193	OUT_CELL	-
DAT8	192	IO_CELL	-
OEB_DAT9	191	OUT_CELL	-
DAT9	190	IO_CELL	-
OEB_DAT11	189	OUT_CELL	-
DAT11	188	IO_CELL	-
OEB_DAT12	187	OUT_CELL	-
DAT12	186	IO_CELL	-
OEB_DAT10	185	OUT_CELL	-
DAT10	184	IO_CELL	-
OEB_DAT13	183	OUT_CELL	-
DAT13	182	IO_CELL	-
OEB_DAT15	181	OUT_CELL	-
DAT15	180	IO_CELL	-
OEB_DAT14	179	OUT_CELL	-

DAT14	178	IO_CELL	-
OEB_GPIO41	177	OUT_CELL	-
GPIO41	176	IO_CELL	-
OEB_GPIO38	175	OUT_CELL	-
GPIO38	174	IO_CELL	-
OEB_GPIO37	173	OUT_CELL	-
GPIO37	172	IO_CELL	-
OEB_GPIO39	171	OUT_CELL	-
GPIO39	170	IO_CELL	-
VSS	169	IN_CELL	-
OEB_GPIO36	168	OUT_CELL	-
GPIO36	167	IO_CELL	-
OEB_GPIO40	166	OUT_CELL	-
GPIO40	165	IO_CELL	-
CPUMODE1	164	IN_CELL	-
CPUMODE0	163	IN_CELL	-
OEB_SD	162	OUT_CELL	-
SD	161	IO_CELL	-
OEB_GPIO35	160	OUT_CELL	-
GPIO35	159	IO_CELL	-
OEB_GPIO33	158	OUT_CELL	-
GPIO33	157	IO_CELL	-
OEB_GPIO30	156	OUT_CELL	-
GPIO30	155	IO_CELL	-
OEB_GPIO34	154	OUT_CELL	-
GPIO34	153	IO_CELL	-
OEB_GPIO31	152	OUT_CELL	-
GPIO31	151	IO_CELL	-
HOP_N	150	IN_CELL	-
OEB_GPIO32	149	OUT_CELL	-
GPIO32	148	IO_CELL	-
OEB_GPIO29	147	OUT_CELL	-
GPIO29	146	IO_CELL	-
SCS_N	145	IN_CELL	-
OEB_VREF_Q1	144	OUT_CELL	-
VREF_Q1	143	IO_CELL	-
SCLK	142	IN_CELL	-
OEB_VREF_Q2	141	OUT_CELL	-
VREF_Q2	140	IO_CELL	-
OEB_VREF_Q0	139	OUT_CELL	-
VREF_Q0	138	IO_CELL	-
OEB_VREF_Q3	137	OUT_CELL	-
VREF_Q3	136	IO_CELL	-
OEB_VREF_Q4	135	OUT_CELL	-
VREF_Q4	134	IO_CELL	-
OEB_VREF_Q6	133	OUT_CELL	-
VREF_Q6	132	IO_CELL	-
OEB_VREF_Q5	131	OUT_CELL	-
VREF_Q5	130	IO_CELL	-
OEB_VREF_Q7	129	OUT_CELL	-
VREF_Q7	128	IO_CELL	-
OEB_VREF_Q8	127	OUT_CELL	-
VREF_Q8	126	IO_CELL	-
OEB_VREF_Q9	125	OUT_CELL	-
VREF_Q9	124	IO_CELL	-
OEB_VREF_Q10	123	OUT_CELL	-
VREF_Q10	122	IO_CELL	-
OEB_VREF_Q11	121	OUT_CELL	-
VREF_Q11	120	IO_CELL	-
OEB_VREF_Q12	119	OUT_CELL	-
VREF_Q12	118	IO_CELL	-
OEB_VREF_Q13	117	OUT_CELL	-
VREF_Q13	116	IO_CELL	-
OEB_VREF_Q15	115	OUT_CELL	-
VREF_Q15	114	IO_CELL	-
OEB_VREF_Q14	113	OUT_CELL	-
VREF_Q14	112	IO_CELL	-
REFCLK	111	IN_CELL	-
OEB_VREF_I2	110	OUT_CELL	-
VREF_I2	109	IO_CELL	-
OEB_VREF_I1	108	OUT_CELL	-
VREF_I1	107	IO_CELL	-
OEB_VREF_I0	106	OUT_CELL	-
VREF_I0	105	IO_CELL	-
OEB_VREF_I5	104	OUT_CELL	-
VREF_I5	103	IO_CELL	-
OEB_VREF_I4	102	OUT_CELL	-

VREF_I4	101	IO_CELL	-
OEB_VREF_I8	100	OUT_CELL	-
VREF_I8	99	IO_CELL	-
OEB_VREF_I3	98	OUT_CELL	-
VREF_I3	97	IO_CELL	-
OEB_VREF_I7	96	OUT_CELL	-
VREF_I7	95	IO_CELL	-
OEB_VREF_I6	94	OUT_CELL	-
VREF_I6	93	IO_CELL	-
OEB_VREF_I11	92	OUT_CELL	-
VREF_I11	91	IO_CELL	-
OEB_VREF_I10	90	OUT_CELL	-
VREF_I10	89	IO_CELL	-
OEB_VREF_I15	88	OUT_CELL	-
VREF_I15	87	IO_CELL	-
OEB_VREF_I14	86	OUT_CELL	-
VREF_I14	85	IO_CELL	-
OEB_VREF_I9	84	OUT_CELL	-
VREF_I9	83	IO_CELL	-
OEB_VREF_I13	82	OUT_CELL	-
VREF_I13	81	IO_CELL	-
OEB_GPIO28	80	OUT_CELL	-
GPIO28	79	IO_CELL	-
OEB_VREF_I12	78	OUT_CELL	-
VREF_I12	77	IO_CELL	-
RESET_N	76	IN_CELL	-
OEB_GPIO27	75	OUT_CELL	-
GPIO27	74	IO_CELL	-
OEB_GPIO25	73	OUT_CELL	-
GPIO25	72	IO_CELL	-
OEB_GPIO26	71	OUT_CELL	-
GPIO26	70	IO_CELL	-
OEB_GPIO23	69	OUT_CELL	-
GPIO23	68	IO_CELL	-
OEB_GPIO22	67	OUT_CELL	-
GPIO22	66	IO_CELL	-
OEB_GPIO24	65	OUT_CELL	-
GPIO24	64	IO_CELL	-
OEB_GPIO21	63	OUT_CELL	-
GPIO21	62	IO_CELL	-
OEB_GPIO19	61	OUT_CELL	-
GPIO19	60	IO_CELL	-
OEB_GPIO18	59	OUT_CELL	-
GPIO18	58	IO_CELL	-
OEB_GPIO20	57	OUT_CELL	-
GPIO20	56	IO_CELL	-
OEB_GPIO17	55	OUT_CELL	-
GPIO17	54	IO_CELL	-
OEB_GPIO16	53	OUT_CELL	-
GPIO16	52	IO_CELL	-
OEB_GPIO15	51	OUT_CELL	-
GPIO15	50	IO_CELL	-
VOBS_Q15	49	IN_CELL	-
VOBS_Q13	48	IN_CELL	-
VOBS_Q9	47	IN_CELL	-
VOBS_Q14	46	IN_CELL	-
VOBS_Q10	45	IN_CELL	-
VOBS_Q6	44	IN_CELL	-
VOBS_Q11	43	IN_CELL	-
VOBS_Q7	42	IN_CELL	-
VOBS_Q12	41	IN_CELL	-
VOBS_Q8	40	IN_CELL	-
VOBS_Q3	39	IN_CELL	-
VOBS_Q4	38	IN_CELL	-
VOBS_Q5	37	IN_CELL	-
VOBS_Q0	36	IN_CELL	-
VOBS_Q1	35	IN_CELL	-
VOBS_Q2	34	IN_CELL	-
VOBS_I12	33	IN_CELL	-
VOBS_I13	32	IN_CELL	-
VOBS_I14	31	IN_CELL	-
VOBS_I15	30	IN_CELL	-
VOBS_I11	29	IN_CELL	-
VOBS_I10	28	IN_CELL	-
VOBS_I9	27	IN_CELL	-
VOBS_I8	26	IN_CELL	-
VOBS_I7	25	IN_CELL	-

VOBS_I6	24	IN_CELL	-
VOBS_I5	23	IN_CELL	-
VOBS_I4	22	IN_CELL	-
VOBS_I3	21	IN_CELL	-
VOBS_I2	20	IN_CELL	-
VOBS_I1	19	IN_CELL	-
VOBS_I0	18	IN_CELL	-
SCANSEL	17	IN_CELL	-
SCANEN	16	IN_CELL	-
OE_N	15	IN_CELL	-
OEB_GPIO14	14	OUT_CELL	-
GPIO14	13	IO_CELL	-
OEB_GPIO13	12	OUT_CELL	-
GPIO13	11	IO_CELL	-
OEB_GPIO12	10	OUT_CELL	-
GPIO12	9	IO_CELL	-
OEB_GPIO11	8	OUT_CELL	-
GPIO11	7	IO_CELL	-
OEB_GPIO10	6	OUT_CELL	-
GPIO10	5	IO_CELL	-
OEB_GPIO9	4	OUT_CELL	-
GPIO9	3	IO_CELL	-
OEB_GPIO8	2	OUT_CELL	-
GPIO8	1	IO_CELL	-
VD_FMT	0	IN_CELL	-

6.4.3 Instructions

The following is an description of the standard instructions. Each instruction selects an serial test data register path between input, TDI and output, TDO.

BYPASS

The bypass instruction shifts data from input, TDI to output, TDO with one TCK clock period delay. The instruction is used to bypass the device.

EXTEST

The external test instruction allows testing of the interconnection to other devices. When the current instruction is the EXTEST instruction, the boundary scan register is place between input, TDI and output, TDO. Primary device inputs can be sampled by loading the boundary scan register using the Capture-DR state. The sampled values can then be viewed by shifting the boundary scan register using the Shift-DR state. Primary device outputs can be controlled by loading patterns shifted in through input TDI into the boundary scan register using the Update-DR state.

SAMPLE

The sample instruction samples all the device inputs and outputs. For this instruction, the boundary scan register is placed between TDI and TDO. Primary device inputs and outputs can be sampled by loading the boundary scan register using the Capture-DR state. The sampled values can then be viewed by shifting the boundary scan register using the Shift-DR state.

IDCODE

The identification instruction is used to connect the identification register between TDI and TDO. The device's identification code can then be shifted out using the Shift-DR state.

STCTEST

The single transport chain instruction is used to test out the TAP controller and the boundary scan register during production test. When this instruction is the current instruction, the boundary scan register is connected between TDI and TDO. During the Capture-DR state, the device identification code is loaded into the boundary scan register. The code can then be shifted out output, TDO using the Shift-DR state.

6.4.4 Boundary Scan Cells

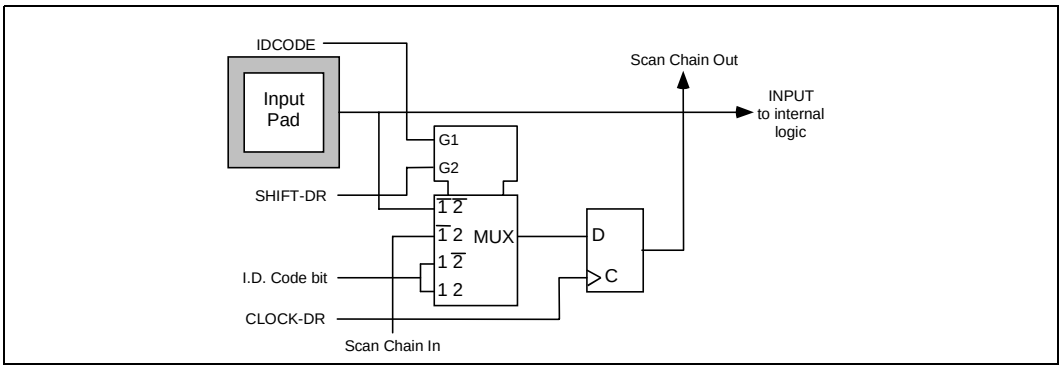


Table 6-4 Input Observation Cell (IN_CELL)

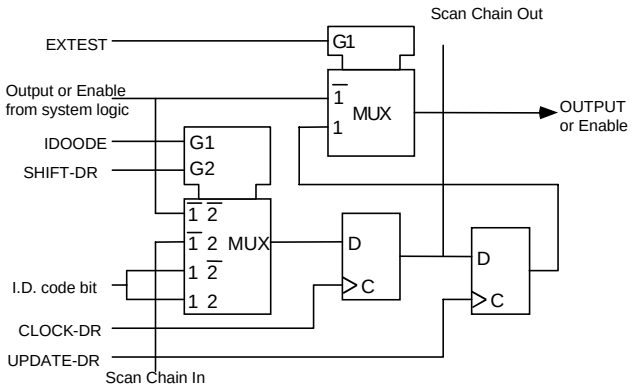


Figure 6-3 Output Cell (OUT_CELL)

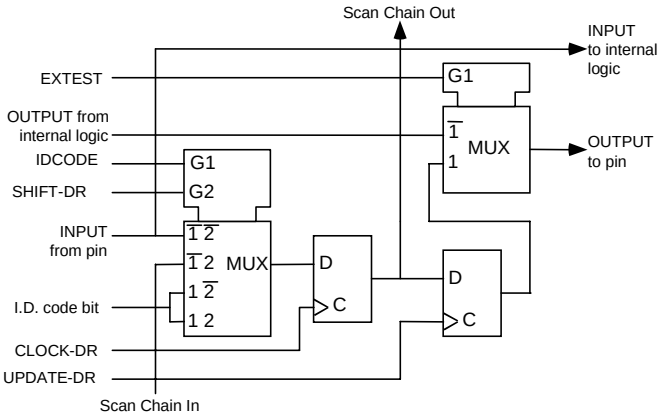


Figure 6-4 Bi-directional Cell (IO_CELL)

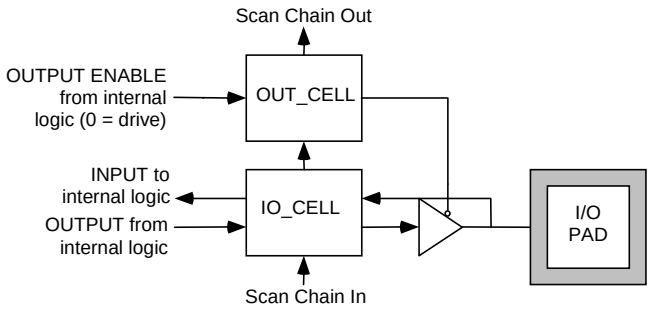


Figure 6-5 Layout of Output Enable and Bidirectional Cells

7 Absolute Maximum Ratings

Maximum rating are the worst case limits that the device can withstand without sustaining permanent damage. They are not indicative of normal mode operation conditions.

Table 7-1 Absolute Maximum Ratings

Parameter	Symbol	Value	Units
Junction Temperature	T_J (Absolute)	+150	°C
Junction Temperature under Bias (Operation) [1]	T_J (Operation)	-40 to +125	°C
Junction Temperature under Bias (Long-Term) [2]	T_J (Long-Term)	-40 to +105	°C
Storage Temperature	T_{ST}	-40 to +125	°C
Supply Voltage [3,4]	V_{VDDI}	-0.3 to + 3.6	V_{DC}
Supply Voltage [3,4]	V_{VDD}	-0.3 to + 6.0	V_{DC}
Voltage on Any Pin	V_{IN}	-0.3 to 6.0	V_{DC}
Static Discharge Voltage		±1000	V
Latch-Up Current		±100	mA
DC Input Current	I_{IN}	±20	mA
Lead Temperature		+230	°C

Notes on Power Supplies:

1. Correct operation is not guaranteed outside these limits.
2. Long-term operation outside these limits will reduce reliability.
3. VDD must power up before VDDI.
4. VDD must not drop below VDDI except when VDDI is not powered.

8 DC Characteristics

T_J = -40°C to +125°C, V_{VDD} = 3.3 V ± 5%, V_{VDDI} = 1.8V ± 5%

(Typical Conditions: T_J = 25°C, V_{VDD} = 3.3 V, V_{VDDI} = 1.8V)

Table 8-1 D.C.Characteristics

Symbol	Parameter	Min	Typ	Max	Units	Conditions
VDD	Pin Power Supply	3.15	3.3	3.45	Volts	Tj = -40°C to 125°C
VDDI	Core Power Supply	1.71	1.8	1.89	Volts	Tj = -40°C to 125°C
VIL	Input Low Voltage	0		0.8	Volts	
VIH	Input High Voltage	2.0			Volts	
VOL	Output or Bidirectional Low Voltage	0	0.1	0.4	Volts	VDD = min IOL = 4mA for • VREF_I[15:0], VREF_Q[15:0] (outputs in scan test mode only) IOL = 12mA for • IRQ_N • VD_I[15:0], VD_Q[15:0] • SD • TDO • GPIO[47:0] IOL = 15mA for • DAT[15:0] • WAIT_N
VOH	Output or Bidirectional High Voltage	2.4			Volts	VDD = min IOH = -4mA for • VREF_I[15:0], VREF_Q[15:0] (outputs in scan test mode only) IOH = -12mA for • IRQ_N • VD_I[15:0], VD_Q[15:0] • SD • TDO • GPIO[47:0] IOH = -15mA for • DAT[15:0] • WAIT_N
VT+	Schmitt Input High Threshold Voltage	2.2			Volts	
VT-	Schmitt Input Low Threshold Voltage			0.8	Volts	
VTH	Schmitt Input Hysteresis Voltage		0.87		Volts	

Table 8-1 D.C.Characteristics

Symbol	Parameter	Min	Typ	Max	Units	Conditions
IIL	Input Low Leak Current	-10		+10	µA	VIL = GND. Note 1
IIH	Input High Leak Current	-10		+10	µA	VIH = VDD. Note 1
IILPD	Input Low Leak Current for input with pull-down.	-10		+10	µA	VIL = GND. Note 1
IIHPD	Input High Leak Current for input with pull-down.	+50		+400	µA	VIH = VDD. Note 1
IILPU	Input Low Leak Current for input with pull-up.	-300		-10	µA	VIL = GND. Note 1
IIHPU	Input High Leak Current for input with pull-up.	-10		+10	µA	VIH = VDD. Note 1
CIN	Input Capacitance		8		pF	
COUT	Output Capacitance		8		pF	
CIO	Bidirectional Capacitance		8		pF	
IDDOP (VDDI)	Core Operating Current for PM7800		230 (VDDI = 1.80V)	560 (VDDI = 1.89V)	mA	DCLK = 80MHz
IDDOP (VDDO)	Nominal IO Operating Current for PM7800		132		mA	VDDO = 3.3V DCLK = 80MHz C_L = 20pF

Notes on D.C. Characteristics:

1. Positive currents flow into the device (sinking), negative currents flow out of the device (sourcing).

9 AC Timing

T_J = -40°C to +125°C, V_{VDD} = 3.3 V ± 5%, V_{VDDI} = 1.8V ± 5%

Notes on Input Timing:

1. When a set-up time is specified between an input and a clock, the set-up time is the time in nanoseconds from the 1.4 Volt point of the input to the 1.4 Volt point of the clock.
2. When a hold time is specified between an input and a clock, the hold time is the time in nanoseconds from the 1.4 Volt point of the clock to the 1.4 Volt point of the input.
3. It is recommended that the transition time on all clock inputs is less than 15 ns.

Notes on Output Timing:

1. Output propagation delay time is the time in nanoseconds from the 1.4 Volt point of the reference signal to the 1.4 Volt point of the output.
1. Output tristate delay is the time in nanoseconds from the 1.4 Volt point of the reference signal to the point where the total current delivered through the output is less than or equal to the leakage current.

9.1 RESET_N

Figure 9-1 RESET_N

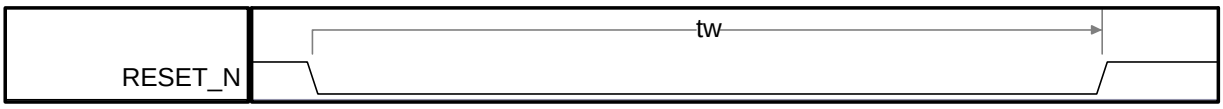


Table 9-1 RESET_N

parameter	min	max	units	description	notes
t_w	100		ns	RESET_N pulse width.	

9.2 CPU Interface

Figure 9-2 CPU Interface - DCLK Timing

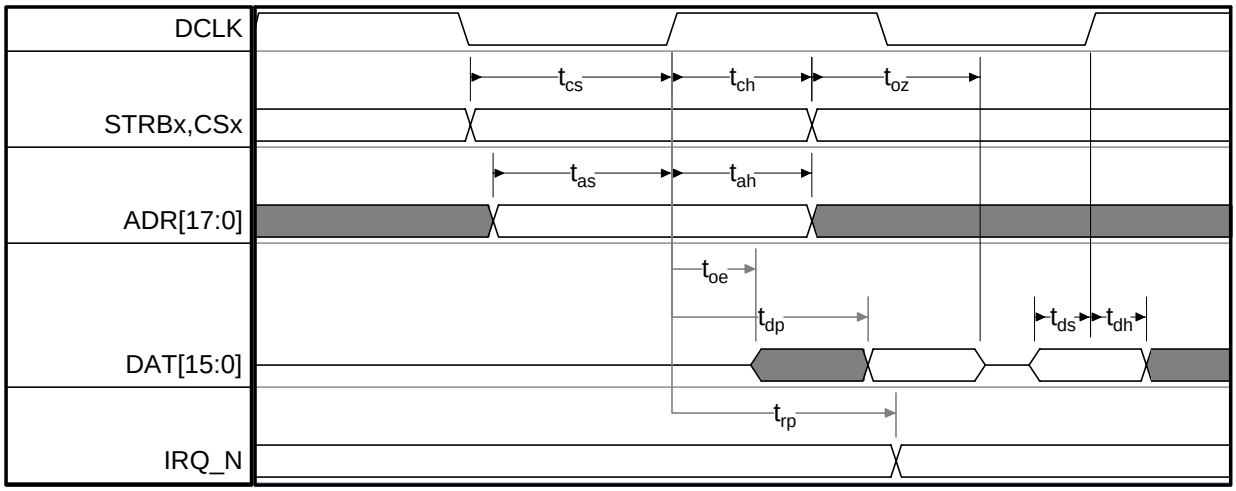


Table 9-2 CPU Interface - DCLK Timing

parameter	min	max	units	description	notes
T_{DCLK}	12.5		ns	DCLK Period.	
t_{cs}	3		ns	STRB1_N, STRB2_N, CS0_N, CS1_N, CS2 setup time to DCLK.	1
t_{ch}	2		ns	STRB1_N, STRB2_N, CS0_N, CS1_N, CS2 hold time from DCLK.	1
t_{as}	5		ns	ADR setup time to DCLK.	1
t_{ah}	0		ns	ADR hold time from DCLK.	1
t_{ds}	5		ns	DAT input-setup time to DCLK.	1
t_{dh}	0		ns	DAT input-hold time from DCLK.	1
t_{oe}		12	ns	DAT driven delay from DCLK, 30pF load.	
t_{dp}		12	ns	DAT propagation delay from DCLK, 30pF load.	
t_{oz}		12	ns	DAT released delay from STRBx, CSx, 30pF load.	
t_{rp}		12	ns	IRQ_N propagation delay from DCLK, 30pF load.	

1. The system designer need not worry about violating setup and hold times - all inputs are treated as asynchronous signals. Setup and hold times are specified here to identify the clock edge on which the signal is sampled.

Figure 9-3 CPU Interface - CPUCLK Timing

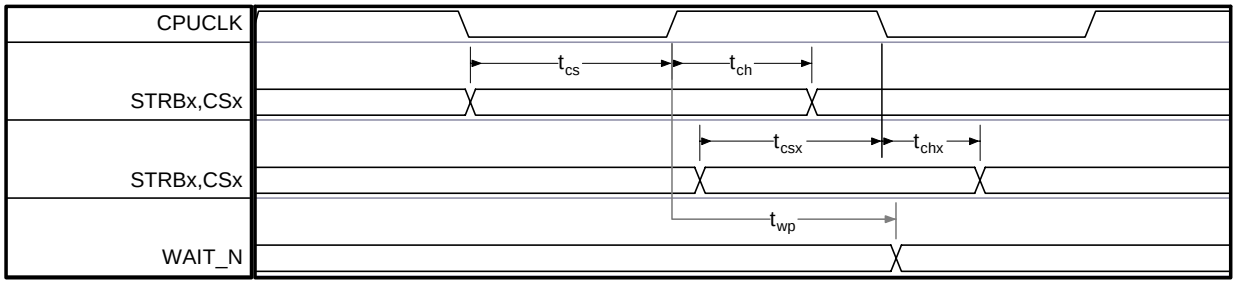


Table 9-3 CPU Interface - CPUCLK Timing

parameter	min	max	units	description	notes
T_{CPUCLK}	8		ns	CPUCLK Period.	
t_{cs}	3		ns	STRB1_N, STRB2_N, CS0_N, CS1_N, CS2 setup time to positive edge CPUCLK.	
t_{ch}	2		ns	STRB1_N, STRB2_N, CS0_N, CS1_N, CS2 hold time from posedge CPUCLK.	
t_{csx}	3		ns	STRB1_N, STRB2_N, CS0_N, CS1_N, CS2 setup time to negative edge CPUCLK (for reads operation with CPUMODE3 = 1).	
t_{chx}	2		ns	STRB1_N, STRB2_N, CS0_N, CS1_N, CS2 hold time from negative edge CPUCLK (for reads operation with CPUMODE3 = 1).	
t_{wp}	2 (15pF) 5 (15pF)	10 (30pF) 16 (30pF)	ns ns	WAIT_N propagation delay from CPUCLK (CPUMODE4 = 0). WAIT_N propagation delay from CPUCLK (CPUMODE4 = 1).	

9.3 VREF, VOBS, VD Interfaces

9.3.1 Dual-Clock System

Use the dual-clock system when VREF is synchronous to REFCLK (usually when VREF is upsampled internally to the DCLK rate). In this case, a FIFO and interpolator are used to upsample VREF to the DCLK rate. REFCLK must be created from DCLK such that there is no varying phase shift that may cause under-run or over-run.

Figure 9-4 Dual-Clock System timing

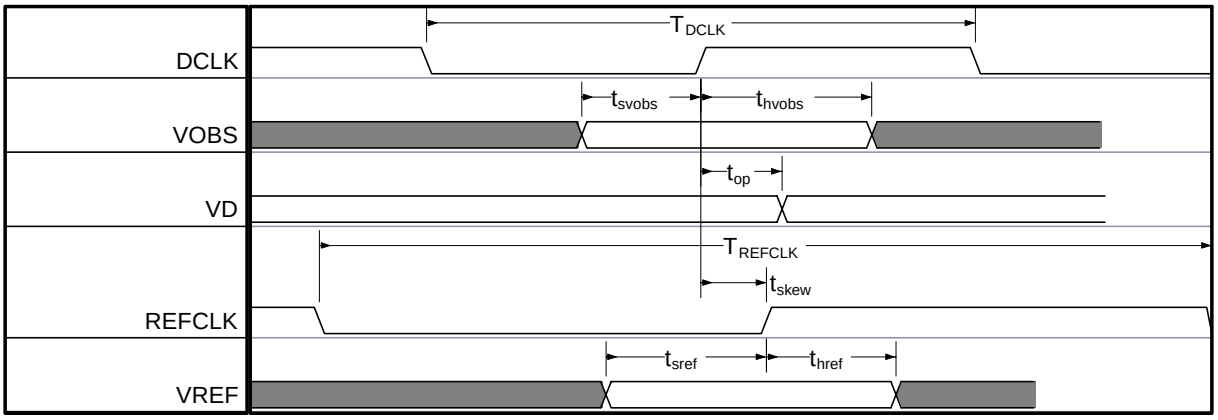


Table 9-4 Dual-Clock System timing

parameter	min	max	units	description	notes
T_{DCLK}	12.5		ns	DCLK period.	
T_{REFCLK}	12.5		ns	REFCLK period.	1
t_{skew}	0	T_{DCLK}	ns	DCLK to REFCLK skew.	2
t_{svobs}	3		ns	VOBS setup time to DCLK.	
t_{hvobs}	0		ns	VOBS hold time from DCLK.	
t_{op}	1.8 (15pF)	7.8 (30pF)	ns	Output delay from DCLK.	
t_{sref}	2		ns	VREF setup time to REFCLK.	
t_{href}	0		ns	VREF hold time from REFCLK.	

Note:

1. T_{REFCLK} must be an exact integer multiple of T_{DCLK}, i.e. $T_{REFCLK} = nT_{DCLK}$, where n = 1,2,...,10. REFCLK must be created from DCLK.

2. Skew may exist between DCLK and REFCLK, but this skew must not vary over time by more than one DCLK period.

9.3.2 Single-Clock System

Use the single-clock system when all signals are synchronous to DCLK and the data-rate of VREF is the same as the output data-rate of VD. In this case, connect the REFCLK pin to VSS, and bypass the FIFO and interpolator to reduce latency through the chip.

Figure 9-5 Single-Clock System timing

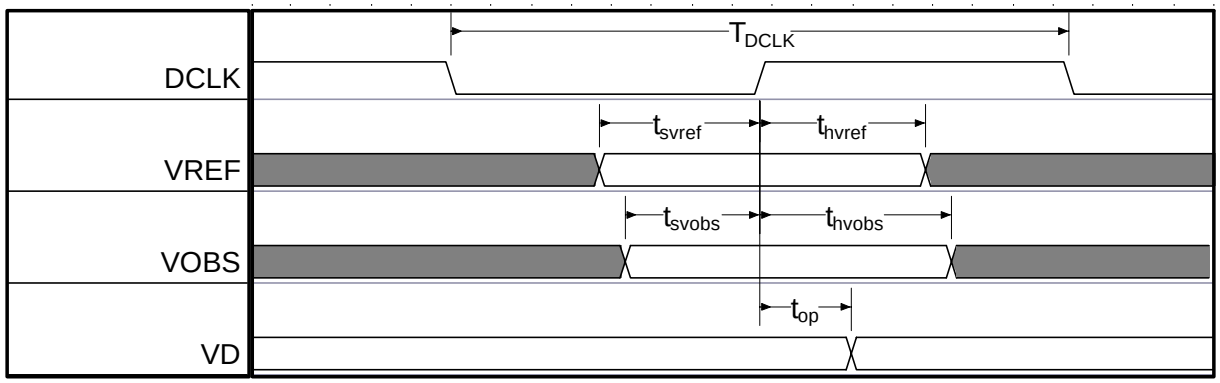


Table 9-5 Single-Clock System timing

parameter	min	max	units	description	notes
T_{DCLK}	12.5		ns	DCLK period.	
t_{svref}	3		ns	VREF setup time to DCLK.	
t_{hvref}	0		ns	VREF hold time from DCLK.	
t_{svobs}	3		ns	VOBS setup time to DCLK.	
t_{hvobs}	0		ns	VOBS hold time from DCLK.	
t_{op}	1.8 (15pF)	7.8 (30pF)	ns	VD propagation delay from DCLK.	

9.4 Serial Interface

9.4.1 Serial Interface AC Timing

All serial inputs are debounced and synchronized to dclk. The minimum pulse width for all serial input signals is $2T_{DCLK}$ + setup and hold.

Figure 9-6 AC Timing for Serial Inputs (SCLK, SCS_N, SD, HOP_N)

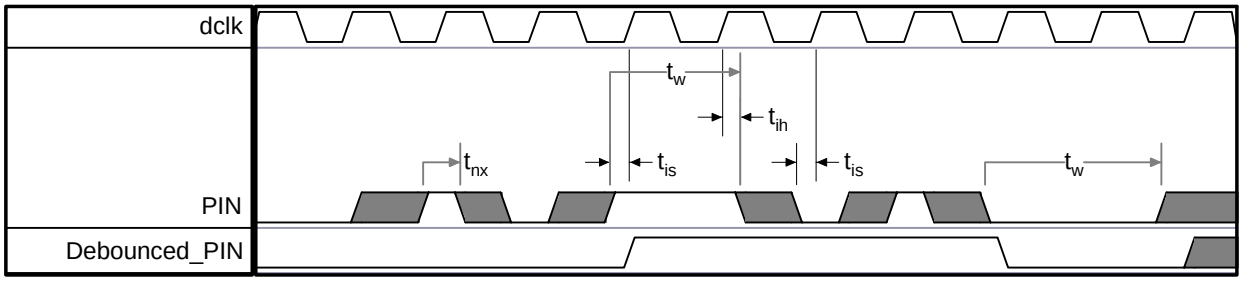


Table 9-6 AC Timing for Serial Inputs (SCLK, SCS_N, SD, HOP_N)

parameter	min	max	units	description	notes
t_{is}	0		ns	Setup time to DCLK.	1
t_{ih}	3		ns	Hold time from DCLK.	1
t_w	$2T_D + t_{is} + t_{ih}$		ns	Pulse width for input.	2
t_{nx}		$T_D - t_{is} - t_{ih}$	ns	Noise exclusion period.	3

Note:

1. The system designer need not worry about violating t_{is} and t_{ih} - all inputs are treated as asynchronous signals. Setup and hold times are specified here to identify the clock edge on which the signal is sampled.
2. The input signal must be stable for at least two rising edges of DCLK.
3. All serial inputs are debounced on rising and falling edges. The period of noise (bounce) must be smaller than $T_D - t_{is} - t_{ih}$.

9.5 GPIO

Figure 9-7 GPIO

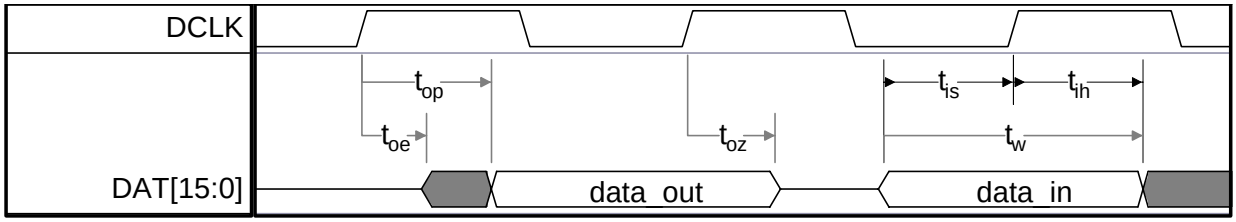


Table 9-7 GPIO

parameter	min	max	units	description	notes
t_{is}	3		ns	GPIO input-setup time to DCLK.	1
t_{ih}	2		ns	GPIO input-hold time from DCLK.	1
t_w	$2T_{DCLK} + t_{is} + t_{ih}$		ns	GPIO[31:24] signal width.	2
t_{oe}	2 (15pF)	14 (30pF)	ns	GPIO output-driven delay from DCLK.	
t_{op}	2 (15pF)	14 (30pF)	ns	GPIO output-propagation delay from DCLK.	
t_{oz}	2 (15pF)	14 (30pF)	ns	GPIO output-released delay from DCLK.	

1. The system designer need not worry about violating t_{is} and t_{ih} - all inputs are treated as asynchronous signals. Setup and hold times are specified here to identify the clock edge on which the signal is sampled.
2. GPIO[31:24] are debounced. The input signal must be stable for at least two rising edges of DCLK. The period of noise (bounce) must be smaller than $T_D - t_{is} - t_{ih}$.

9.6 JTAG Interface

Figure 9-8 JTAG Interface

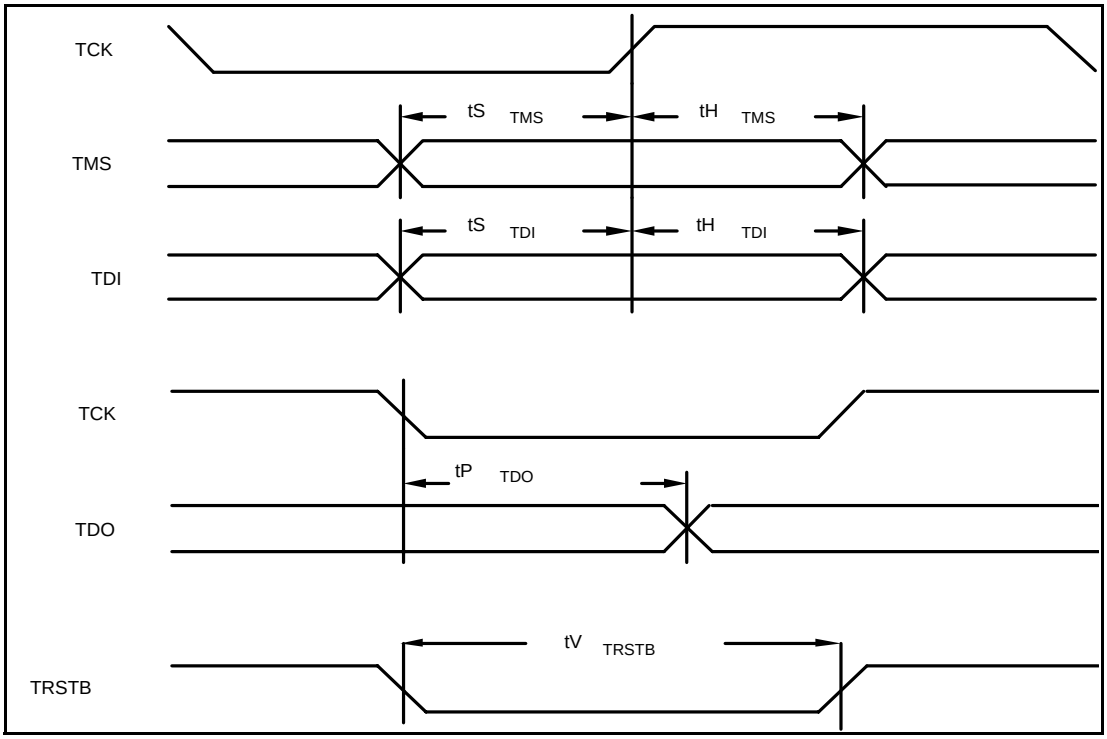
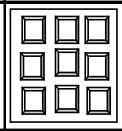
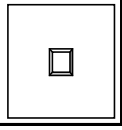


Table 9-8 JTAG Interface

parameter	min	max	units	description	notes
		5	MHz	TCK Frequency	
	40	60	%	TCK Duty Cycle	
tS_{TMS}	50		ns	TMS Set-up time to TCK	
tH_{TMS}	50		ns	TMS Hold time to TCK	
tS_{TDI}	50		ns	TDI Set-up time to TCK	
tH_{TDI}	50		ns	TDI Hold time to TCK	
tP_{TDO}	2	50	ns	TCK Low to TDO Valid	
tV_{TRSTB}	100		ns	TRSTB Pulse Width	

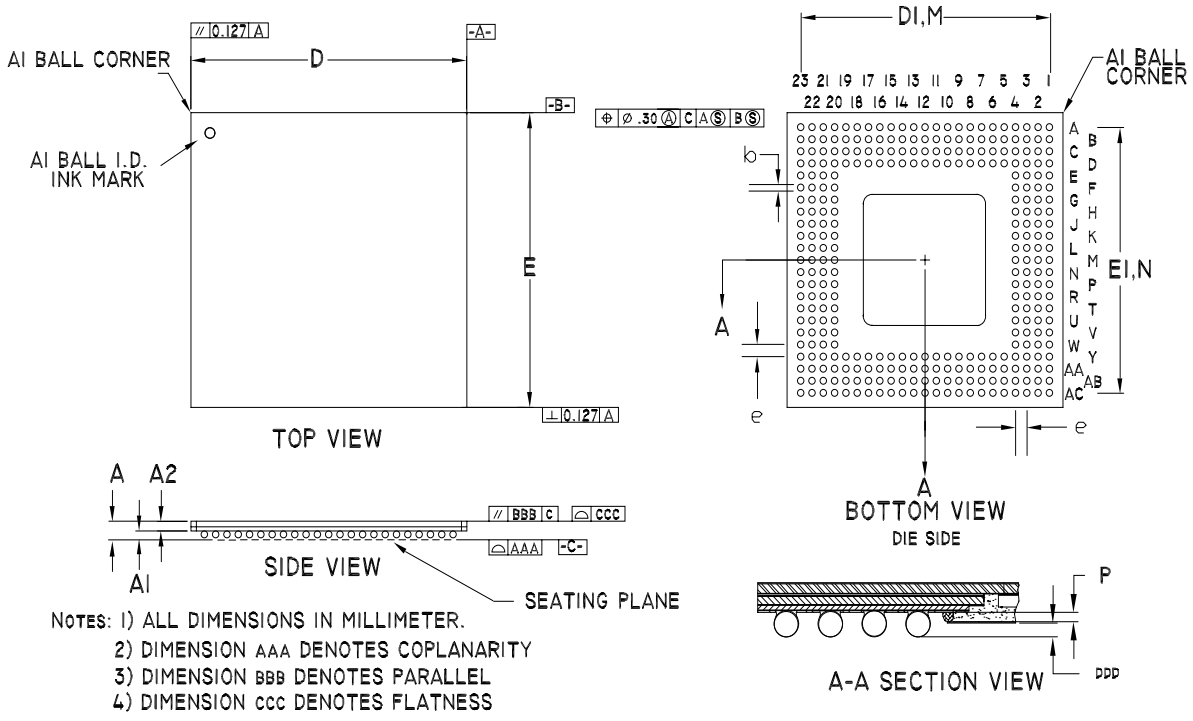
10 Thermal

		Forced Air (Linear Feet per Minute)					
Theta JA @ 1.6W		Conv	100	200	300	400	500
Dense Board		23.4	20.7	18.9	17.8	17.3	17.1
JEDEC Board		15.7	14.0	12.8	12.0	11.4	10.9

Notes:

1. Dense Board is defined as a 3S3P board and consists of a 3x3 array of device PM7800 located as close to each other as board design rules allow. All PM7800 devices are assumed to be dissipating 1.6Watts. Theta-J_A listed is for the device in the middle of the array.
2. JEDEC Board: Theta-J_A is the measured value for a single thermal device in the same package on a 2S2P board following EIA/JESD 51-3.

11 Mechanical



NOTES: 1) ALL DIMENSIONS IN MILLIMETER.
2) DIMENSION AAA DENOTES COPLANARITY
3) DIMENSION BBB DENOTES PARALLEL
4) DIMENSION CCC DENOTES FLATNESS

PACKAGE TYPE: 304 PIN THERMAL BALL GRID ARRAY															
BODY SIZE: 31 x 31 x 1.45 MM															
DIM.	A	A1	A2	D	D1	E	E1	M,N	e	b	AAA	BBB	CCC	DDD	P
MIN.	1.41	0.56	0.85	30.90	27.84	30.90	27.84			0.60				0.15	0.20
NOM.	1.54	0.63	0.91	31.00	27.94	31.00	27.94	23x23	1.27	0.75				0.33	0.30
MAX.	1.67	0.70	0.97	31.10	28.04	31.10	28.04			0.90	0.15	0.15	0.20	0.50	0.35

CONTACTING PMC-SIERRA, INC.

PMC-Sierra, Inc.
105-8555 Baxter Place,
Burnaby, BC
Canada V5A 4V7

Tel: (604) 415-6000

Fax: (604) 415-6200

Document Information: document@pmc-sierra.com
Corporation Information: info@pmc-sierra.com
Application Information: apps@pmc-sierra.com
(604) 415-4533

Web Site: http://www.pmc-sierra.com

PMC-2001646 (R3) Issue date: July, 2001