MSM6100™ Mobile Station Modem

Device Specification (Advance Information)

80-V4689-1 Rev. D

March 14, 2003

QUALCOMM Proprietary

QUALCOMM Incorporated
5775 Morehouse Drive
San Diego, CA 92121-1714
U.S.A.

QUALCOMM Incorporated
5775 Morehouse Drive
San Diego, CA 92121-1714
U.S.A.

Submit Technical Questions at:
https://support.cdmatech.com

MSM6100™ Device Specification (Advance Information)

80-V4689-1 Rev. D

Contents

Figures

Tables

Preface

About this Technical Manual

This manual provides hardware interface and programming information for the MSM6100 Mobile Station Modem. The manual is divided into the following chapters:

Chapter 1: Introduction	This chapter introduces users to the MSM6100 device's basic features and functions.
Chapter 2: Pin Descriptions	This chapter lists each MSM6100 device pin and its function within the device. The pinout for the 341-ball CSP package is listed by functional grouping.
Chapter 3: Electrical Specifications	This chapter specifies the recommended operating conditions, DC voltage characteristics, I/O timing, and power estimations for the MSM6100 device. Timing diagrams are also included.
Chapter 4: Interface Descriptions	This chapter details each subsystem or block within the MSM6100 device and how the subsystem or block interfaces to external peripherals.
Chapter 5: Mechanical Dimensions	This chapter provides package dimensions and land pattern for the 341-ball CSP.

Usage Conventions

The following table defines terms commonly used throughout this manual.

Table 1 Usage Conventions

Term	Definition
AAGC	Audio Automatic Gain Control
ADC	Analog-to-digital converter
AEC	Acoustic Echo Cancellation
AGC	Automatic Gain Control
AMPS	Advanced Mobile Phone System (analog IS-95)
ASIC	Application Specific Integrated Circuit
BER	Bit error rate
BPF	Bandpass Filter
Bps	Bits per second
CAGC	CDMA AGC
CDMA	Code-Division Multiple Access
CELP	Code Excited Linear Prediction
CMX™	Compact Media Extension. Trademarked by QUALCOMM.
CODEC	Coder-Decoder
CR	Command Register
CSM	Cell Site Modem
CSP	Chip Scale Package
CRC	Cyclic Redundancy Code
DBR	Data Burst Randomization
DFM	Digital Frequency Modulation
DSP	Digital Signal Processor
DTMF	Dual-Tone Multiple-Frequency
EBI	External Bus Interface
ESEC	Ear Seal Echo Cancellation
ETM	Embedded Trace Macrocell
EVRC	Enhanced Variable Rate CODEC
F-QPCH	Forward Quick Paging Channel

Table 1 Usage Conventions (Continued)

Term	Definition
FIFO	First-In First-Out
FIR	Finite Impulse Response
FM	Frequency Modulation
FOCC	Forward Control Channel
FVC	Forward Voice Channel
GPIO	General-purpose Input/Output
HPF	Highpass Filter
ICD	In-Circuit Debugger
ICE	In-Circuit Emulation, or In-circuit emulator
IF	Intermediate Frequency
IR	Instruction Register
JPEG	Joint Photographic Experts Group
JTAG	Joint Test Action Group (ANSI/ICEEE Std. 1149.1–1990)
kbps	Kilobits per Second
LCD	Liquid Crystal Display
LNA	Low Noise Amplifier
LPF	Lowpass Filter
LSByte or LSBit	Defines whether the LSB is the least significant bit or least significant byte. All instances of LSB used in this manual are assumed to be LSByte, unless otherwise specified.
MMC	Multimedia Card
MPEG	Moving Pictures Experts Group
MSByte or MSBit	Defines whether the MSB is the most significant bit or most significant byte. All instances of MSB used in this manual are assumed to be MSByte, unless otherwise specified.
MSM™	Mobile Station Modem. Trademarked by QUALCOMM.
NMI	Non-maskable Interrupt
NS	Noise Suppression
PA	Power Amplifier
PCM	Pulse Coded Modulation
PDE	Position Determination Entity
PCS	Personal Communications Service

Table 1 Usage Conventions (Continued)

Term	Definition
PDM	Pulse Density Modulation
PN	Pseudo-Random Noise
PNG	Portable Network Graphics
PSRAM	Pseudo Static Random Access Memory
QCELP	QUALCOMM Code Excited Linear Prediction
RC	Resistance-Capacitance
RF	Radio Frequency
RLP	Radio Link Protocol
RRM	Reduced Rate Mode
RSSI	Receive Signal Strength Indicator
RVC	Reverse Voice Channel
Rx	Receive
SAT	Supervisory Audio Tone
SCH	Supplemental Channel
SDRAM	Synchronous Dynamic Random Access Memory
SMV	Selectable Mode Vocoder
SNR	Signal to Noise Ratio
Sps	Symbols Per Second (or Samples Per Second)
SER	Symbol Error Rate
TAP	Test Access Port
TCXO	Temperature-Controlled Crystal Oscillator
Tx	Transmit
UART	Universal Asynchronous Receiver Transmitter
UHF	Ultra High Frequency
WBD	Wide Band Data
ZIF	Zero Intermediate Frequency

Special Marks

The following table defines special marks used in this manual.

Table 2 Special Marks

Mark	Definition
[ ]	Brackets ([ ]) sometimes follow a pin, register, or bit name. These brackets enclose a range of numbers. For example, GPIO_INT [7:0] may indicate a range that is 8 bits in length, or DATA[7:0] may refer to all eight DATA pins.
_N	A suffix of "_N" indicates an active low signal. For example, RESIN_N.
0x0000	Hexadecimal numbers are identified with an x in the number, for example, 0x0000. All numbers are decimal (base 10) unless otherwise specified. Non-obvious binary numbers have the term "binary" enclosed in parentheses at the end of the number, for example, 0011 (binary).
|	A vertical bar in the outside margin of the page indicates a change or revision has occurred since the last release of the document.

Revision History

Table 3 Revision History

Revision	Date	Description
A	July 25, 2002	Initial release
B	August 16, 2002	■ Changed occurrences of FBGA throughout document to CSP. ■ Added several items to the acronym list in Table 1: CSP, CMX, ETM, MPEG, JPEG, PSRAM, PNG, MLZ, RSP3, PDE, MMC, EVRC, F-QPCH, SCH, SMV, EBI, SDRAM, DSP, ZIF ■ Removed MICE acronym from Table 1. ■ Added ETM module to MSM6100 device in Figure 1-1. ■ Removed redundant description of features in Section 1.2.1. ■ Fixed typos in subsystem descriptions in Section 1.2.4 and Section 1.2.6. ■ Added P1, P2, P3, and P4 parameter and description to Table 2-1. ■ Removed part of heading description for Alternate Functions column in Table 2-2. ■ Added P1, P2, P3, and P4 parameters to the Pad_Type column in Table 2-2. ■ Made corrections to various pad types in Table 2-2. ■ Modified and clarified various pin functions in Table 2-2. ■ Changed reference type from _X to [X] in Table 2-3. ■ Changed pin names of E23 and R23 in Table 2-5. ■ Replaced all occurrences of lub_n with ub_n in Chapter 3. ■ Replaced all occurrences of AVSS with GND in Chapter 3. ■ Added note regarding pad voltages to Table 3-2. ■ Added “(Ground Voltage)” to note b of Table 3-4 and Table 3-5. ■ Added text to note c of Table 3-6 explaining unavailability of information for Chapter 4. ■ Removed “internal” reference to part number in description of line 3 of Table 5-1.

Table 3 Revision History (Continued)

Revision	Date	Description
C	September 25, 2002	■ Added new Chapter 4
D	March 14, 2003	■ Changed technical support to https://support.cdmatech.com. ■ Changed nomenclature from 285-pin to 341-pin. ■ Section 1.2.6 48 to 64 voices (spec change). ■ Table 2-2; RTCK, VDD_C, VDD_P1, GND pins, “necessary to connect one ball” deleted. ■ Table 2-2; VDD_C corrected typo on voltage from 2.60V to 1.867V. ■ Table 2-2: deleted drive strength from unused voltage setting. ■ Updated EBI1 and EBI2 sections in Chapter 3 and 4. ■ Changed High/Low-Level output voltage specifications for DC characteristics in Table 3-4. ■ Added one-fourth-rated High/Low-Level output voltage specifications in Table 3-4 and Table 3-5. ■ Added rated drive strength footnote to Table 3-4 and Table 3-5. ■ Added Section 3.2.1, Measurement Conditions. ■ Added Section 3.2.2, Bus Rise and Fall Times. ■ Added Section 3.2.3, Clock Jitter. ■ Deleted paragraph in Section 3.2.5, replaced by new Section 3.2.1. ■ Figure 4-2. Changed capacitors from 1000pF to 0.01µF ■ Figure 4-4. Switched MICOUTN and MICOUTP. ■ Figure 4-5. Corrected MICOUTN to MICOUTP. ■ Section 4.8.1, added “NAND is accessed through a special controller” to fourth bullet point, and changed seventh bullet point’s 1.8V to 1.867V. ■ Corrected labels in Figure 4-48 to MSM6100 usage. ■ Figure 4-54. Corrected GPIO_FUNCSEL to GPIO_CFG:FUNC_SEL and GPIO_OE to GPIO_PAGE. ■ Corrected GPIO numbers in Table 4-27, Figure 4-55, and Figure 4-56. ■ Section 4.10.4.4: changed all mentions of “EIA/CEA” to CEA (including Figure 4-57), and corrected the title of the interface document. ■ Added Section 4.13.5. ■ Added Sections 4.13.6.1 to 4.13.6.9. ■ Corrected MCN in Figure 5-1 from 2815 to 4400.

1 Overview

1.1 Application Description

QUALCOMM CDMA Technologies (QCT) strives to constantly improve the indispensable communication tools our customers use every day. QCT creates state-of-the-art chipsets, system software, development tools and products—such as the Wireless Internet Launchpad™ suite of hardware cores and software. This enables the most advanced wireless multimedia applications available on a wireless device while continually reducing complexity, cost, and board-space requirements.

Third-generation (3G) mobile products are becoming personal information tools for work and entertainment, taking advantage of multimedia content delivered through high-speed data access. 3G form factors such as smart phones, PDAs, handheld computers or other devices will benefit from the MSM6100™ chipset's multimedia support for MP3 players, audio e-mail, still image e-mail, video e-mail, image capture, voice capture, karaoke, 2D/3G games, advanced gaming with position location, and a whole host of applications using the core Wireless Internet Launchpad features.

QCT's MSM6100 solution, part of QCT's MSM6xxx Mobile Station Modem (MSM™) family of chipsets and system software uses QCT's revolutionary radioOne™ Zero Intermediate Frequency (ZIF), or direct conversion, architecture. It is optimized to support voice and multimedia data applications while enabling CDMA2000 1X network benefits. The MSM6100 solution provides a seamless migration path from 2G to 3G services and applications, and increases voice capacity for CDMA2000 1X networks. It will also enable CDMA developers to quickly develop 3G CDMA2000 1X handsets that exceed the specifications of mobile stations for worldwide cdmaOne™ and 3G 1xMC systems, including those based on IS-95A/B and IS-2000 standards.

The MSM6100 chipset solution consists of the MSM6100 baseband processor, direct conversion RFL6000^{TM} and RFR6000^{TM} receive devices, the direct conversion RFT6100^{TM} transmit device, PM6050^{TM} power management device and a compatible power amplifier device. These devices perform all of the signal processing and power management in the subscriber unit.

The 3G CDMA2000 1X MSM6100 chipset and system software features radioOne direct conversion architecture and incorporates a low-power, high-performance RISC microprocessor core featuring the ARM926EJ-S™ CPU and Jazelle™ accelerator circuit for advanced Java applications from ARM® Limited. The MSM6100 solution integrates two low-power, high-performance QDSP4000™ digital signal processor (DSP) cores. Use of the ARM926EJ-S™ CPU and QDSP4000 DSP eliminates the need for the multimedia companion processor(s) normally required for video-based applications, playing MP3 music files and MIDI synthesizer/CMX functions.

The MSM6100 chipset and system software incorporates the advanced feature set of QCT's Wireless Internet Launchpad™ suite of technologies, integrated MPEG-4 video decoding/encoding, MP3 audio decoding, a 2D/3D graphics accelerator for advanced gaming applications, a Compact Media Extension (CMX™)/MIDI synthesizer, a digital camera interface, an enhanced LCD interface, and JPEG encoding/decoding.

The MSM6100 solution supports QUALCOMM's gpsOne™ position location technology, including standalone mode in which the handset can act as a GPS receiver. The gpsOne solution, featuring SnapTrack™ technology, offers robust data availability under the most challenging conditions, whether in concrete-and-steel high-rises, convention centers, shopping malls, or urban canyons. Using a hybrid approach that utilizes signals from both the GPS satellite constellation and from CDMA cell sites, the gpsOne solution enhances location services availability, accelerates the location determination process and provides better accuracy for callers, whether during emergency situations or while using GPS-enabled commercial applications. The MSM6100 solution also supports the Wireless Internet Launchpad's VectorOne™ compass capability.

The MSM6100 chipset reduces radio bill-of-materials (BOM) by the introduction of RadioOne RF devices. System BOM is further reduced by supporting interfaces to next-generation memories architectures such as; NAND FLASH, Psuedo SRAM (PSRAM), Page and Burst mode NOR FLASH and low power SDRAM (LP-SDRAM).

QCT provides a complete software suite, Dual-Mode Subscriber (DMSS) software, for building handsets around the MSM6100 chipset. DMSS software is designed to run on a Subscriber Unit Reference (SURF) phone platform, an optional development platform optimized to assist in evaluating, testing and debugging DMSS software.

The MSM6100 device is offered in a 341-ball, 0.5mm pitch Chip Scale Package (CSP) production package. Additionally, the MSM6100 solution supports QUALCOMM's Binary Run-time Environment for Wireless™ (BREW™) applications development platform.

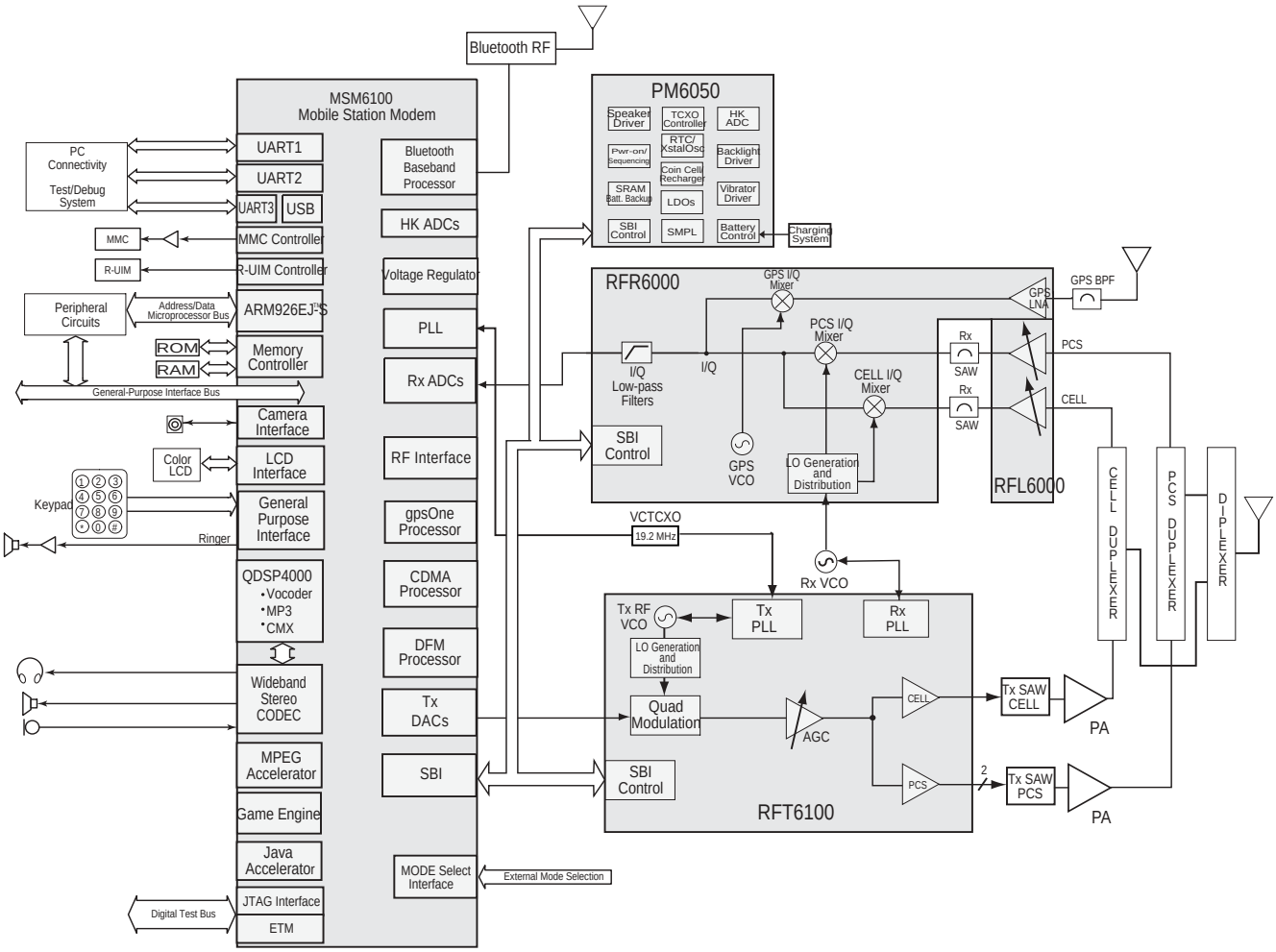


Figure 1-1 MSM6100 Device Functional Block Diagram

1.2 MSM6100 Features

1.2.1 MSM6100 General Features

- MSM6100 system solution consists of the MSM6100, RFL6000, RFR6000, RFT6100, PM6050™, DMSS system software and SURF6100™
- High-performance ARM926EJ-S microprocessor core
- Java hardware acceleration for faster Java-based games and other applets
- Integrated Bluetooth™ baseband processor for wireless connectivity to peripherals
- Integrated wideband stereo codec for digital audio applications

- Direct interface to digital camera module
- Tri-Mode (CDMA cellular, CDMA PCS, AMPS cellular) operation
- gpsOne position location capabilities
- Vocoder support (EVRC, 13K QCELP, SMV)
- Advanced 13*13 mm, 0.5mm pitch, 341-pin CSP packaging technology

1.2.2 IS-2000 1X Features Supported by the MSM6100 Device

- Fast 800 Hz forward power control
- Quasi-Orthogonal functions
- Supplemental channel (SCH) support
- CDMA2000 1X Forward Quick Paging channel (F-QPCH)
- Convolutional and turbo codes on SCH
- Radio Link Protocol (RLP3)

1.2.3 MSM6100 Audio Processing Features

- Integrated wideband stereo voice CODEC
  - 16-bit DAC with typical 88 dB Dynamic Range
  - Supports sampling rates up to 48 kHz on the speaker path and 16 kHz on the microphone path
  - Supports summing of an external device's stereo single-ended analog signal
  - Supports summing of I2S digital audio signal
  - Supports Headset switch press detection
- Voice Recognition (PureVoiceVR™), including speaker-independent digit dialing
- Acoustic echo cancellation
- Audio AGC
- Internal Vocoder supporting 13kbps Pure Voice QCELP, SMV, and EVRC
  - Standard MIDI with 16 Voices

1.2.4 MSM6100 Microprocessor Subsystem

- Industry standard ARM926EJ-S™ embedded microprocessor subsystem
  - 16KB Instruction and 16KB Data cache
  - Instruction set compatible with ARM926EJ-S
  - ARM v5TEJ instructions
  - Higher performance 5 stage pipeline, Harvard cached architecture
  - Higher internal CPU clock rate with on-chip cache
- Java hardware acceleration
- Enhanced memory support
  - Dual memory buses separating the memory subsystem from low-speed peripherals such as LCD panels
  - DMA support for LCD refresh
  - 1.867 V or 2.6 V memory interface support (excluding EBI1)
  - Page and burst mode NOR FLASH or SRAM
  - Burst Mode is supported on all four chip selects
  - NAND FLASH memory interface
  - Boot from NAND
  - Low-power SDRAM (LP-SDRAM) interface
  - Multimedia Card (MMC) support
- Internal watchdog and sleep timers

1.2.5 Supported Interface Features

- Universal Serial Bus (USB)
- Removable Universal Identity Module (R-UIM) card interface
- Three universal asynchronous receiver transmitter (UART) serial ports
- R-UIM controller (via second or third UART)
- Parallel LCD interface
- General-purpose I/O pins

1.2.6 Supported Multimedia Features

- Integrated MPEG-4 decoder/encoder for streaming Video-on-Demand (VOD) and video mail applications
- Integrated JPEG decoder/encoder for camera-based applications
- Integrated 2D/3D graphics for more realistic game play
- CMX (text, picture, and MIDI streaming)
  - 64 Voices
  - 44 kHz Sampling
  - 128 kb wavetable
  - Morphing Player and Free A.T.
  - PNG decoder
  - Pitch Bend Range Support
  - MLZ Decoder
  - Integrated PNG/SAF A.T.

1.2.7 gpsOne™ Technology

gpsOne technology merges Global Positioning System (GPS) satellite and network information to provide a high-availability solution that offers industry-leading accuracy and performance. This solution performs under the most challenging conditions and provides a platform for both location-based applications and FCC compliance.

When a request for position location is issued, the handset takes pilot phase measurements and requests assistance from the PDE. The PDE sends the assistance information to the handset. The handset/mobile unit measures the GPS pseudoranges and pilot phase (again) and provides data to the PDE. The PDE then calculates position location and returns results to the requesting entity.

This hybrid method uses both GPS satellite pseudoranges and pilot offsets. The strengths of this solution are:

- High-sensitivity solution works in all terrain, indoors, outdoors, urban, rural
- High availability is enabled by using both satellite and pilot information

Hence, while network solutions do poorly in rural areas and areas of poor cell geometry/density, and while pure GPS solutions do not work indoors, the QUALCOMM gpsOne solution combines information from both network and satellites to calculate a fix under all conditions. This is further enabled using assistance provided by the PDE that enhances the sensitivity of the gpsOne receiver by approximately 20 to 30 dB.

As opposed to network solutions that require equipment at each cell site, the gpsOne solution packs a complete GPS receiver into every MSM6100 chipset. This means that each handset is capable of position location without requiring expensive cell site equipment. This solution not only meets the FCC E911 mandate under the most challenging conditions, but also provides a ubiquitous platform for location-based applications. gpsOne and radioOne technologies will enable consumer-priced, position-capable handsets.

1.2.8 radioOne™ Technology

The MSM6100 device interfaces directly with QCT's new radioOne RF ASICs. radioOne is a revolutionary technology for CDMA transceivers that uses Zero Intermediate Frequency (ZIF), or direct conversion, architecture for the wireless handset market. This direct conversion eliminates the need for large IF Surface Acoustic Wave (SAW) filters and additional IF circuitry, which significantly reduces the handset BOM parts count, facilitating multiband and multimode handsets that can be produced in smaller form factors. radioOne technology also incorporates the frequency synthesis and passive elements used in converting baseband signals to and from RF. A single external local oscillator is used for the CDMA receiver, which will provide the capabilities needed to operate on systems around the world and will simplify the procurement of parts and the cost of designing CDMA handsets.

2 Pin Descriptions

2.1 I/O Description Parameters

Table 2-1 I/O Description (Pad Type) Parameters

Symbol	Description
Type	
I	CMOS input
IS	Input with Schmitt trigger
O	Output
Z	High-Z output
B	Bidirectional
BS	Bidirectional with Schmitt trigger
P1	Tied to pad group 1 that uses supply voltage V_{DD_P1}.
P2	Tied to pad group 2 that uses supply voltage V_{DD_P2}.
P3	Tied to pad group 3 that uses supply voltage V_{DD_P3}.
P4	Tied to pad group 4 that uses supply voltage V_{DD_P4}.
V	Power/ground
Special Circuitry	
PU	Contains an internal pull-up device
PD	Contains an internal pull-down device
NP	Contains no internal pull
K	Contains an internal weak keeper device (Keepers cannot drive external buses.)
A	Analog pad

Table 2-1 I/O Description (Pad Type) Parameters (Continued)

Symbol	Description
HV	Digital input where input voltage level may reach up to TBD.
n-m	Variable drive strength pins. The number *n* is the drive strength when the PAD_CTL register bit is clear (0). The number *m* is the drive strength when the PAD_CTL register bit is set (1).
OD	Open drain
HLV	The I/O can be configured as 2.60V or 1.8V.

2.2 Pin Names and Pinouts

Pins with an 'a' in their pin number may be found by referring to their alternate function's pin number.

Table 2-2 Pin Names and Pinouts

pin #	Native Mode				ETM Mode		Volt	Pad_ Type	2.6V Dr mA	1.867V Dr mA	Description
	Main Function	Dir-Pol	Alternate Functions {SURF Functions}	Dir-Pol	{SURF Functions}	Dir-Pol					
Clocks, Resets, and Mode control [12 dedicated pins]											
A18	TCXO	I						IA-PD			TCXO (temperature compensated crystal oscillator) clock input
AA6	XTAL48_IN	I						IA			48MHz crystal oscillator input. For USB.
Y7	XTAL48_OUT	O						OA			48MHz crystal oscillator feedback output
T21	SLEEP_XTAL_IN	I						IA			Low-power sleep controller crystal oscillator input.
T24	SLEEP_XTAL_OUT	O						OA			Low-power sleep controller crystal oscillator output.
V24	RESIN_N	I						IS, P4			Hardware reset input to system.
N21	RESOUT_N	O						O, P4	3-5	1.5-2.5	Reset output generated by synchronized RESIN_N and by wdog_reset. The rising edge of RESOUT_N is delayed from the rising edge of RESIN_N by a few microprocessor clocks.
T2	RESOUT_N_EMI1	O						O, P1		1.5-2.5	Reset output generated by synchronized RESIN_N and by wdog_reset. The rising edge of RESOUT_N is delayed from the rising edge of RESIN_N by a few microprocessor clocks. Same voltage level as the first external memory interface. Connected to pad_group1.
A17	WDOG_EN	I						IS-PU, P4			Watchdog timer enable input. WDOG_STB is behind GPIO[20].
N20 ***R20***	MODE[1] MODE[0]	I						IS-PD, P4			Determines operating mode of the chip. MODE[1] - test mode selection MODE[0] - ETM/Native_N MODE[1:0] & WDOG_EN 000 - Native, ARM Jtag, WDOG disabled 001 - Native, ARM Jtag, WDOG enabled 010 - ETM, ARM Jtag, WDOG disabled 011 - ETM, ARM Jtag, WDOG enabled
B20	BOOT_MODE	I						IS, P4			Determines boot mode of the chip. 0 - boot from NOR FLASH in EBI1 1 - boot from NAND FLASH in EBI2
Microprocessor/Memory Interface1 [48 dedicated pins]											
P1	LB1_N	O	SDRAM1_DQM[0]	O				O, P1		4.5-10	Low byte enable signal for byte access of 16-bit memory.
L1	UB1_N	O	SDRAM1_DQM[1]	O				O, P1		4.5-10	Upper byte enable signal for byte access of 16-bit memory.
J5	WE1_N	O	SDRAM1_WE_N	O				O, P1		4.5-10	Write enable signal
P5	OE1_N	O		O				O, P1		4.5-10	Output enable signal (effectively read enable signal)

Table 2-2 Pin Names and Pinouts (Continued)

pin #	Native Mode				ETM Mode		Volt	Pad_ Type	2.6V Dr mA	1.867V Dr mA	Description
	Main Function	Dir-Pol	Alternate Functions {SURF Functions}	Dir-Pol	{SURF Functions}	Dir-Pol					
a	A1[24]	O	GPIO[32] SDRAM1_A[0]	B				BS-PD, P1		4.5-10	Peripheral address bus.
a	A1[23]	O	GPIO[31]	B				BS-PD, P1		4.5-10	
A4	A1[22]	O	SDRAM1_D[31]	B				B-K, P1		4.5-10	
G1	A1[21]	O	SDRAM1_D[30]	B				B-K, P1		4.5-10	A1[23] is SDRAM1_DQM[2] (output) for 32-bit SDRAM.
J2	A1[20]	O	SDRAM1_D[29]	B				B-K, P1		4.5-10	
G2	A1[19]	O	SDRAM1_D[28]	B				B-K, P1		4.5-10	
J1	A1[18]	O	SDRAM1_D[27]	B				B-K, P1		4.5-10	
M2	A1[17]	O	SDRAM1_D[26]	B				B-K, P1		4.5-10	
B4	A1[16]	O	SDRAM1_D[25]	B				B-K, P1		4.5-10	
D6	A1[15]	O	SDRAM1_D[24]	B				B-K, P1		4.5-10	
D5	A1[14]	O	SDRAM1_A[14]	O				O, P1		4.5-10	
D1	A1[13]	O	SDRAM1_A[13]	O				O, P1		4.5-10	
D2	A1[12]	O	SDRAM1_A[12]	O				O, P1		4.5-10	
E1	A1[11]	O	SDRAM1_A[11]	O				O, P1		4.5-10	
F5	A1[10]	O	SDRAM1_A[10]	O				O, P1		4.5-10	
F2	A1[9]	O	SDRAM1_A[9]	O				O, P1		4.5-10	
F4	A1[8]	O	SDRAM1_A[8]	O				O, P1		4.5-10	
M1	A1[7]	O	SDRAM1_A[7]	O				O, P1		4.5-10	
M4	A1[6]	O	SDRAM1_A[6]	O				O, P1		4.5-10	
M5	A1[5]	O	SDRAM1_A[5]	O				O, P1		4.5-10	
N2	A1[4]	O	SDRAM1_A[4]	O				O, P1		4.5-10	
R1	A1[3]	O	SDRAM1_A[3]	O				O, P1		4.5-10	
N4	A1[2]	O	SDRAM1_A[2]	O				O, P1		4.5-10	
N5	A1[1]	O	SDRAM1_A[1]	O				O, P1		4.5-10	
B3 *E5* *E6* *G5* *H4* *H2* *K1* *L5* *E4* *C2* *G4* *F1* *H5* *H1* *K2* *L4*	D1[15] D1[14] D1[13] D1[12] D1[11] D1[10] D1[9] D1[8] D1[7] D1[6] D1[5] D1[4] D1[3] D1[2] D1[1] D1[0]	B						B-K, P1		4.5-10	Peripheral data bus
P4	RAM1_CS_N[0]	O	SDRAM1_CS_N[0]					O, P1		4.5-10	SRAM chip select
a	RAM1_CS_N[1]	O	GPIO[77]	B				BS-PU, P1		4.5-10	SDRAM chip select or RAM1[1] chip select SDRAM1_CS_N[1] (output)
P2	ROM1_CS_N[0]	O						O, P1		4.5-10	ROM chip select
a	ROM1_CS_N[1]	O	GPIO[34]	B				BS-PU, P1		4.5-10	ROM chip select
R2	ROM1_CLK	B	SDRAM1_CLK	B				BS-NP, P1		4.5-10	SDRAM clock or Burst mode FLASH clock. Power on as output.
R4	SDRAM1_CLK_EN	O						O, P1		4.5-10	SDRAM clock Enable. Alternate is a GPO (General Purpose Output) controlled by the EBI1_PSRAM_CRE register.
R5	ROM1_ADV_N	O	SDRAM1_RAS_N	O				O, P1		4.5-10	Burst mode FLASH's address valid signal. SDRAM row address strobe.
K4	ROM1_WAIT_N	I	SDRAM1_CAS_N	O				BS-PU, P1		4.5-10	Burst FLASH ready signal. SDRAM column address strobe
Other SDRAM signals 0 dedicated pins]											
a	SDRAM1_CLK	O	ROM1_CLK	O				O, P1		4.5-10	
a	SDRAM1_RAS_N	O	ROM1_ADV_N	O				O, P1		4.5-10	
a	SDRAM1_CAS_N	O	ROM1_WAIT_N	I				BS-PU, P1		4.5-10	

Table 2-2 Pin Names and Pinouts (Continued)

pin #	Native Mode				ETM Mode		Volt	Pad_ Type	2.6V Dr mA	1.867V Dr mA	Description
	Main Function	Dir-Pol	Alternate Functions {SURF Functions}	Dir-Pol	{SURF Functions}	Dir-Pol					
Microprocessor/Memory Interface2 [39 dedicated pins]											
AD4	LB2_N	O	A2[0] NAND1_ALE	O O			HLV	O, P2	3-5	1.5-2.5	Low byte enable signal for byte access of 16-bit memory.
AD9	UB2_N	O	NAND1_CLE	O			HLV	O, P2	3-5	1.5-2.5	Upper byte enable signal for byte access of 16-bit memory.
Y10	WE2_N	O					HLV	O, P2	3-5	1.5-2.5	Write enable signal.
AC12	OE2_N	O	NAND1_RE_N	O			HLV	O, P2	3-5	1.5-2.5	Output enable signal (effectively read enable signal)
a	A2[20]	O	GPIO[78]	B			HLV	BS-PD, P2	3-5	1.5-2.5	Second peripheral address bus
T5	A2[19]	O					HLV	BS-PD, P2	3-5	1.5-2.5	
U4	A2[18]	O					HLV	BS-PD, P2	3-5	1.5-2.5	
V1	A2[17]	O					HLV	O, P2	3-5	1.5-2.5	
U2	A2[16]	O					HLV	B-K, P2	3-5	1.5-2.5	
V2	A2[15]	O					HLV	B-K, P2	3-5	1.5-2.5	
W1	A2[14]	O					HLV	B-K, P2	3-5	1.5-2.5	
U5	A2[13]	O					HLV	B-K, P2	3-5	1.5-2.5	
V5	A2[12]	O					HLV	B-K, P2	3-5	1.5-2.5	
W2	A2[11]	O					HLV	B-K, P2	3-5	1.5-2.5	
V4	A2[10]	O					HLV	B-K, P2	3-5	1.5-2.5	
Y1	A2[9]	O					HLV	B-K, P2	3-5	1.5-2.5	
AA1	A2[8]	O					HLV	B-K, P2	3-5	1.5-2.5	
Y2	A2[7]	O					HLV	B-K, P2	3-5	1.5-2.5	
W4	A2[6]	O					HLV	B-K, P2	3-5	1.5-2.5	
Y4	A2[5]	O					HLV	B-K, P2	3-5	1.5-2.5	
AA2	A2[4]	O					HLV	B-K, P2	3-5	1.5-2.5	
W5	A2[3]	O					HLV	B-K, P2	3-5	1.5-2.5	
AB2	A2[2]	O					HLV	B-K, P2	3-5	1.5-2.5	
AC3	A2[1]	O					HLV	B-K, P2	3-5	1.5-2.5	
a	A2[0]	O	LB2_N	O			HLV	O, P2	3-5	1.5-2.5	
Y6 ***AC4*** ***AC5*** ***AA7*** ***AD5*** ***AC6*** ***Y8*** ***AD6*** ***AC7*** ***AA8*** ***AD7*** ***AC8*** ***Y9*** ***AD8*** ***AC9*** ***AA9***	D2[15] D2[14] D2[13] D2[12] D2[11] D2[10] D2[9] D2[8] D2[7] D2[6] D2[5] D2[4] D2[3] D2[2] D2[1] D2[0]	B					HLV	B-K, P2	3-5	1.5-2.5	Second peripheral data bus
a	LCD2_CS_N	O	GPIO[38]	B			HLV	BS-PU, P2	3-5	1.5-2.5	Peripheral chip-select.
a	LCD2_EN	O	GPIO[37]	B			HLV	BS-PD, P2	3-5	1.5-2.5	Peripheral chip-select.
a	GP2_CS_N[1]	O	GPIO[36]	B			HLV	BS-PU, P2	3-5	1.5-2.5	Peripheral chip-select.
a	GP2_CS_N[0]	O	GPIO[35]	B			HLV	BS-PU, P2	3-5	1.5-2.5	Peripheral chip-select.
a	ROM2_CS_N	O	GPIO[33]	B			HLV	BS-PU, P2	3-5	1.5-2.5	Peripheral chip-select.
a	RAM2_CS_N	O	GPIO[67]	B			HLV	BS-PU, P2	3-5	1.5-2.5	Peripheral chip-select.

Table 2-2 Pin Names and Pinouts (Continued)

pin #	Native Mode				ETM Mode		Volt	Pad_ Type	2.6V Dr mA	1.867V Dr mA	Description
	Main Function	Dir-Pol	Alternate Functions {SURF Functions}	Dir-Pol	{SURF Functions}	Dir-Pol					
General Purpose I/O and Interrupt Pins [82 dedicated pins]											
D21	GPIO[81]	B	RINGER	O				BS-PD, P4	3-5	1.5-2.5	
E20	GPIO[80]	B	GP_MN GP_CLK	O O				BS-PD, P4	3-5	1.5-2.5	
M21	GPIO[79]	B	FM_LNA_RANGE	O				BS-PU, P4	3-5	1.5-2.5	
U1	GPIO[78]	B	A2[20]	O			HLV	BS-PD, P2	3-5	1.5-2.5	Peripheral address bus.
T4	GPIO[77]	B	RAM1_CS_N[1]	O				BS-PU, P1		4.5-10	Peripheral chip-select. SDRAM1_CS_N[1]
E8	GPIO[76]	B	SDRAM1_DQM[3]	O			HLV	B-PD, P3	9-20	4.5-10	These pins are used as data pins for SDRAM.
E9	GPIO[75]	B	SDRAM1_D[23]	B-K			HLV	B-K, P3	9-20	4.5-10	Connect the Vdd of these pads to VDD_P3.
D8	GPIO[74]	B	SDRAM1_D[22]	B-K			HLV	B-K, P3	9-20	4.5-10	
B6	GPIO[73]	B	SDRAM1_D[21]	B-K			HLV	B-K, P3	9-20	4.5-10	Short VDD_P3 to VDD_P1.
E7	GPIO[72]	B	SDRAM1_D[20]	B-K			HLV	B-K, P3	9-20	4.5-10	
A6	GPIO[71]	B	SDRAM1_D[19]	B-K			HLV	B-K, P3	9-20	4.5-10	
D7	GPIO[70]	B	SDRAM1_D[18]	B-K			HLV	B-K, P3	9-20	4.5-10	
B8	GPIO[69]	B	SDRAM1_D[17]	B-K			HLV	B-K, P3	9-20	4.5-10	
A7	GPIO[68]	B	SDRAM1_D[16]	B-K			HLV	B-K, P3	9-20	4.5-10	
AC11	GPIO[67]	B	RAM2_CS_N	O			HLV	BS-PU, P2	3-5	1.5-2.5	Peripheral chip select output.
A10	GPIO[66]	B	{BACKLIGHT}	O-L	ETM_TRACECLK	O		BS-PD, P4	3-6	1.5-3	
A12	GPIO[65]	B	{TCXO_EN}		ETM_TRACESYNC	O		BS-PU, P4	3-6	1.5-3	
D12	GPIO[64]	B	{MSM_DP_RI}	O-L	ETM_PIPESTAT2	O		BS-PD, P4	3-6	1.5-3	
E12	GPIO[63]	B	KEYSENSE1_N {KYPD_3}	I-PU I-PU	ETM_PIPESTAT1	O		BS-PU, P4	3-6	1.5-3	
A13	GPIO[62]	B	KEYSENSE0_N {KYPD_1}	I-PU I-PU	ETM_PIPESTAT0	O		BS-PU, P4	3-6	1.5-3	
B13	GPIO[61]	B	CAMIF_DATA7	I	ETM_TRACE_PKT15	O		B-K, P4	3-6	1.5-3	When running ETM at 16-bit normal mode, they are the upper byte. When running ETM at 8-bit demux mode, they are the data pins for the second connector.
D13	GPIO[60]	B	CAMIF_DATA6	I	ETM_TRACE_PKT14	O		B-K, P4	3-6	1.5-3	
E13	GPIO[59]	B	CAMIF_DATA5	I	ETM_TRACE_PKT13	O		B-K, P4	3-6	1.5-3	
A14	GPIO[58]	B	CAMIF_DATA4	I	ETM_TRACE_PKT12	O		B-K, P4	3-6	1.5-3	
B14	GPIO[57]	B	CAMIF_DATA3	I	ETM_TRACE_PKT11	O		B-K, P4	3-6	1.5-3	
D14	GPIO[56]	B	CAMIF_DATA2	I	ETM_TRACE_PKT10	O		B-K, P4	3-6	1.5-3	
E14	GPIO[55]	B	CAMIF_DATA1	I	ETM_TRACE_PKT9	O		B-K, P4	3-6	1.5-3	
A15	GPIO[54]	B	CAMIF_DATA0	I	ETM_TRACE_PKT8	O		B-K, P4	3-6	1.5-3	
B9	GPIO[53]	B	{KYPD_17}	O-H	ETM_TRACE_PKT7	O		BS-PU, P4	3-6	1.5-3	
D9	GPIO[52]	B	{KYPD_15}	O-H	ETM_TRACE_PKT6	O		BS-PU, P4	3-6	1.5-3	
A8	GPIO[51]	B	{KYPD_13}	O-H	ETM_TRACE_PKT5	O		BS-PU, P4	3-6	1.5-3	
D10	GPIO[50]	B	{KYPD_11}	O-H	ETM_TRACE_PKT4	O		BS-PU, P4	3-6	1.5-3	
E10	GPIO[49]	B	{KYPD_9}	O-H	ETM_TRACE_PKT3	O		BS-PU, P4	3-6	1.5-3	
A11	GPIO[48]	B	KEYSENSE4_N {ON_SW_SENSE}	O-H	ETM_TRACE_PKT2	O		BS-PU, P4	3-6	1.5-3	
B11	GPIO[47]	B	KEYSENSE3_N {KYPD_7}	O-H	ETM_TRACE_PKT1	O		BS-PU, P4	3-6	1.5-3	
D11	GPIO[46]	B	KEYSENSE2_N {KYPD_5}	O-H	ETM_TRACE_PKT0	O		BS-PU, P4	3-6	1.5-3	
B15	GPIO[45]	B	{KYPD_MEMO}	O-L	ETM_TRACESYNC_B	O		BS-PD, P4	3-6	1.5-3	On the ETM mode, there four pins are used in the 8-bit demux mode. These pins are not used in ETM 16-bit normal mode.
A16	GPIO[44]	B	{MSM_DP_DTR}	I-H	ETM_PIPESTAT0B	O		BS-PU, P4	3-6	1.5-3	
E15	GPIO[43]	B	{VEXT_SENSE}	I-L	ETM_PIPESTAT1B	O		BS-PD, P4	3-6	1.5-3	
D15	GPIO[42]	B	{PS_HOLD}	O	ETM_PIPESTAT2B	O		BS-PD, P4	3-6	1.5-3	
AA12	GPIO[41]	B			{ETM_GPIO_IRQ_SRC}	I		PS-PD, P4	3-5	1.5-2.5	
AD14	GPIO[40]	B	{PM_INT_O}	O	{ETM_GPIO_CS_N}	O		BS-PU, P4	3-5	1.5-2.5	
AA14	GPIO[39]	B	{MSM_DP_DCD\}	O-L	{KEYSENSE_IRQ_SRC}	I		BS-PU, P4	3-5	1.5-2.5	

Table 2-2 Pin Names and Pinouts (Continued)

pin #	Native Mode				ETM Mode		Volt	Pad_ Type	2.6V Dr mA	1.867V Dr mA	Description
	Main Function	Dir-Pol	Alternate Functions {SURF Functions}	Dir-Pol	{SURF Functions}	Dir-Pol					
AA10	GPIO[38]	B	LCD2_CS_N	O			HLV	BS-PU, P2	3-5	1.5-2.5	
AC10	GPIO[37]	B	LCD2_EN	O			HLV	BS-PD, P2	3-5	1.5-2.5	
Y11	GPIO[36]	B	GP2_CS_N[1]	O			HLV	BS-PU, P2	3-5	1.5-2.5	
AA11	GPIO[35]	B	GP2_CS_N[0]	O			HLV	BS-PU, P2	3-5	1.5-2.5	
T1	GPIO[34]	B	ROM1_CS_N[1]	O				BS-PU, P1		4.5-10	
AD11	GPIO[33]	B	ROM2_CS_N	O			HLV	BS-PU, P2	3-5	1.5-2.5	Power up as ROM2_CS_N. If NAND FLASH is present, connect it to this pin.
A5	GPIO[32]	B	A1[24]	O				BS-PD, P1		4.5-10	SDRAM1_A1[0] (output) when doing a SDRAM access
B5	GPIO[31]	B	A1[23]	O				BS-PD, P1		4.5-10	SDRAM1_DQM[2]
AD12	GPIO[30]	B	NAND2_FLASH_READY	I			HLV	PS-PD, P2	3-5	1.5-2.5	Configured as NAND2_FLASH_READY at power-on
E18	GPIO[29]	B	UART2_DP_RX_DATA {UIM_PWR_EN}	I O				BS-PD, P4	3-5	1.5-2.5	
B22	GPIO[28]	B	UART2_DP_TX_DATA {UIM_DATA}	O B				BS-PU HV, P4	3-5	1.5-2.5	
L21	GPIO[27]	B	I2C_SCL	B				BS-PU, P4	3-5	1.5-2.5	I2C_SCL needs pull up.
L24	GPIO[26]	B	I2C_SDA	B				BS-PU, P4	3-5	1.5-2.5	I2C_SDA needs pull up.
E17	GPIO[25]	B	UART3_RFR_N UIM2_CLK	O O				BS-PU, P4	3-5	1.5-2.5	
A20	GPIO[24]	B	UART3_DP_TX_DATA {UIM2_DATA}	O O				BS-PU HV, P4	3-5	1.5-2.5	
J21	GPIO[23]	B	USB_TX_VPO	B				BS-PU, P4	3-5	1.5-2.5	
J23	GPIO[22]	B	USB_TX_VMO	B				BS-PD, P4	3-5	1.5-2.5	
J24	GPIO[21]	B	USB_TX_OE_N	O-PU				BS-PU, P4	3-5	1.5-2.5	
H24	GPIO[20]	B	USB_RX_DATA	I-PD				BS-HK, P4	3-5	1.5-2.5	
K23	GPIO[19]	B	USB_RX_VMI MMC_DATA_SRC1	I-PD B				BS-HK P4	3-5	1.5-2.5	
K21	GPIO[18]	B	USB_RX_VPI MMC_CLK_OUT1	I-PU O				BS-HK, P4	3-5	1.5-2.5	
N24	GPIO[17]	B	TX_ON	O				BS-PD, P4	3-5	1.5-2.5	
P23	GPIO[16]	B	GP_PDM[1]	O-PD				BS-PD, P4	3-5	1.5-2.5	
N23	GPIO[15]	B	GP_PDM[2]	O-PD				BS-PD, P4	3-5	1.5-2.5	
E19	GPIO[14]	B	UART2_RFR_N UIM_CLK	O O				BS-PU, P4	3-5	1.5-2.5	
D20	GPIO[13]	B	UART2_CTS_N {UIM_RESET}	I O				BS-PU, P4	3-5	1.5-2.5	
D18	GPIO[12]	B	UART3_CTS_N {UIM2_RESET}	I O				BS-PU, P4	3-5	1.5-2.5	
A21	GPIO[11]	B	UART3_DP_RX_DATA {UIM2_PWR_EN}	I O				BS-PD, P4	3-5	1.5-2.5	
P24	GPIO[10]	B	GP_PDM[0]	O-PD				BS-PD, P4	3-5	1.5-2.5	
A19	GPIO[9]	B	MMC_DATA_SRC0 BT_DATA	B B				BS-PU HV, P4	3-5	1.5-2.5	
E16	GPIO[8]	B	MMC_CMD	B				BS-PU HV, P4	3-5	1.5-2.5	
B16	GPIO[7]	B	MMC_CLK_OUT0 BT_CLK	O I				BS-PU, P4	3-5	1.5-2.5	
J20	GPIO[6]	B	AUX_PCM_CLK BT_SBST	B O				BS-PD, P4	3-5	1.5-2.5	
H21	GPIO[5]	B	AUX_PCM_DOUT BT_SBCK	B B				BS-PU, P4	3-5	1.5-2.5	
F21	GPIO[4]	B	AUX_PCM_DIN BT_SBDT	B B				BS-PD, P4	3-5	1.5-2.5	
G24	GPIO[3]	B	AUX_PCM_SYNC BT_TX_RX_N	B O				BS-PD, P4	3-5	1.5-2.5	
K20	GPIO[2]	B	CAMIF_VSYNC WDOG_STB, {USB_SUSPEND}	B O O				BS-PD, P4	3-5	1.5-2.5	
K24	GPIO[1]	B	CAMIF_HSYNC WDOG_STB	B O				BS-PD, P4	3-5	1.5-2.5	WDOG_STB is duplicated. WDOG_STB is for debug purposes.
L20	GPIO[0]	B	CAMIF_PCLK	I				BS-PD, P4	3-5	1.5-2.5	

Table 2-2 Pin Names and Pinouts (Continued)

pin #	Native Mode				ETM Mode		Volt	Pad_ Type	2.6V Dr mA	1.867V Dr mA	Description
	Main Function	Dir-Pol	Alternate Functions {SURF Functions}	Dir-Pol	{SURF Functions}	Dir-Pol					
User Interface [0 dedicated pins]											
a	KEYSENSE4_N	I-PU	GPIO[48]								Can be used to sense key contact closure when connected to an external keypad. These pins require an active-low level-sensitive input signal and may cause an interrupt to the microprocessor. So possible implementation is detecting KEYSENSE interrupt and sending out drive signals via GPIOs, then we can decide which keypad was pressed.
a	KEYSENSE3_N	I-PU	GPIO[47]								
a	KEYSENSE2_N	I-PU	GPIO[46]								
a	KENSENSE1_N	I-PU	GPIO[63]								
a	KEYSENSE0_N	I-PU	GPIO[62]								
a	RINGER	O	GPIO[81]						5	2.5	DTMF tone generator circuit output which selects the pitch and cadence of the subscriber's ring. This output drives the second transducer.
AUX CODEC Interface [0 dedicated pins]											
a	AUX_PCM_CLK	B	GPIO[6]						3-5	1.5-2.5	PCM clock for auxiliary CODEC port
a	AUX_PCM_DOUT	B	GPIO[5]						3-5	1.5-2.5	PCM data output for auxiliary CODEC port
a	AUX_PCM_DIN	I	GPIO[4]								PCM data input for auxiliary CODEC port
a	AUX_PCM_SYNC	B	GPIO[3]						3-5	1.5-2.5	PCM data strobe for auxiliary CODEC port
Internal CODEC Signals [20 dedicated pins]											
AD19	MIC1P							IA			Mic 1 input (+)
AC18	MIC1N							IA			Mic 1 input (-)
Y18	MIC2P							IA			Mic 2 input (+)
Y17	MIC2N							IA			Mic 2 input (-)
AC17	AUXIP							IA			Auxiliary input (+)
AD18	AUXIN							IA			Auxiliary input (-)
AA18	MICOUTP							OA			Mic output to external Tx high pass filter (+)
AC19	MICOUTN							OA			Mic output to external Tx high pass filter (-)
AC20	MICINP							IA			Mic input from external Tx high pass filter (+)
AA19	MICINN							IA			Mic input from external Tx high pass filter (-)
AD20	MICIFBP							IA			Mic amp feedback from external Tx high pass filter (+)
AD21	MICIFBN							IA			Mic amp feedback from external Tx high pass filter (-)
AA16	EAR1OP							OA			Earphone 1 amplifier output (+)
AD16	EAR1ON							OA			Earphone 1 amplifier output (-)
AD17	EAR2O		HPH_L					OA			Earphone 2 amplifier output (Stereo headphone left output)
Y16	AUXOP							OA			Auxiliary output (+) To carkit, PMIC2, or external speaker
AA17	AUXON							OA			Auxiliary output (-) To carkit, PMIC2, or external speaker
Y19	MICBIAS							OA			Microphone bias supply output, no decoupling capacitors.
Y20	CCOMP							IA			External decoupling capacitor input for CODEC voltage reference.
AD15	HPH_R							OA			Stereo headphone right output.
UART [4 dedicated pins]											
H23	RFR_N	O						O, P4	3	1.5	UART ready-for-receive signal.
E21	CTS_N	I						IS-PU, P4			UART clear-to-send signal
D23	DP_RX_DATA	I						IS-PD, P4			UART receive serial data input.
F20	DP_TX_DATA	O	NAND_BOOT_ERR	O				O, P4	3-5	1.5-2.5	UART transmit serial data output. Boot up as NAND_BOOT_ERR when boots from NAND FLASH.
a	UART2_RFR_N	O	GPIO [14]					P4	3-5	1.5-2.5	2nd UART ready-for-receive signal
a	UART2_CTS_N	I	GPIO [13]					P4	3-5	1.5-2.5	2nd UART clear-to-send signal
a	UART2_DP_RX _DATA	I	GPIO [29]					P4	3-5	1.5-2.5	2nd UART receive serial data input
a	UART2_DP_TX _DATA	O	GPIO [28]					P4	3-5	1.5-2.5	2nd UART transmit serial data output
a	UART3_RFR_N	O	GPIO [25]					P4	3-5	1.5-2.5	3rd UART ready-for-receive signal
a	UART3_CTS_N	I	GPIO [12]					P4	3-5	1.5-2.5	3rd UART clear-to-send signal
a	UART3_DP_RX _DATA	I	GPIO [11]					P4	3-5	1.5-2.5	3rd UART receive serial data input
a	UART3_DP_TX _DATA	O	GPIO [24]					P4	3-5	1.5-2.5	3rd UART transmit serial data output

Table 2-2 Pin Names and Pinouts (Continued)

pin #	Native Mode				ETM Mode		Volt	Pad_ Type	2.6V Dr mA	1.867V Dr mA	Description
	Main Function	Dir-Pol	Alternate Functions {SURF Functions}	Dir-Pol	{SURF Functions}	Dir-Pol					
USB Transceiver Interface [0 dedicated pins]											
a	USB_TX_VPO	O	GPIO[23]					BS-PU, P4	3-5	1.5-2.5	differential output (+)
a	USB_TX_VMO	O	GPIO[22]					BS-PU, P4	3-5	1.5-2.5	differential output (-)
a	USB_TX_OE_N	O	GPIO[21]					BS-PU, P4	3-5	1.5-2.5	An active low output used to enable or disable the D+ and D- pins of the transceiver.
a	USB_RX_DATA	I-HK	GPIO[20]					BS-HK, P4	2	1	single ended input from USB transceiver
a	USB_RX_VMI	I-HK	GPIO[19]					BS-HK, P4	2	1	Gated version of D-. It is used to detect SE0 (-)
a	USB_RX_VPI	I-HK	GPIO[18]					BS-HK, P4	2	1	Gated version of D+. It is used to detect SE0 (+).
a	USB_SUSPEND	O	GPIO[2] for testing					O	3	1.5	indicating suspending state
a	USB_PRG_SPEED	O	any GPIO					O	3	1.5	USB speed selection
MMC Interface [0 dedicated pins]											
a	MMC_DATA_SRC1	B	GPIO[19]					BS-HK, P4	5	2.5	MMC data. Has to be high-drive for speed reason.
	MMC_DATA_SRC0		GPIO[9]					BS-PU, P4	5	2.5	
a	MMC_CMD	B	GPIO[8]					BS-PU, P4	5	2.5	MMC command. Has to be high-drive for speed reason.
a	MMC_CLK_OUT0	O	GPIO[7]					BS-PU, P4	5	2.5	MMC clock. Has to be high-drive for speed reason.
	MMC_CLK_OUT1		GPIO[18]								
SBI [3 dedicated pins]											
T23	SBST	O						O, P4	2-5	1-2.5	I3Q serial bus start/stop.
U24	SBDT	B						B-K, P4	2-5	1-2.5	I3Q serial bus data. (Keeper is option for I3C.)
T20	SBCK	B						B-NP, P4	2-5	1-2.5	I3Q serial bus clock.
I2C [0 dedicated pins]											
a	I2C_SDA	B	GPIO[26]					BS-PU, P4	3-5	1.5-2.5	I2C serial bus data.
a	I2C_SCL	B	GPIO[27]					BS-PU, P4	3-5	1.5-2.5	I2C serial bus clock.
Bluetooth [0 dedicated pins]											
a	BT_CLK	I	GPIO[7]						5	2.5	12 Mhz Bluetooth Clock
a	BT_SBST	O	GPIO[6]						5	2.5	BT SBI strobe signal
a	BT_SBCK	O	GPIO[5]						5	2.5	BT SBI clock
a	BT_SBDT	B	GPIO[4]						5	2.5	BT SBI input/output data
a	BT_TX_RX_N	O	GPIO[3]						5	2.5	Tx/Rx strobe
a	BT_DATA	B	GPIO[9]						5	2.5	Bluetooth Rx/Tx data
ITU656-like Camera[0 dedicated pins]											
a	CAMIF_DATA7	I	GPIO[61]								Data from the camera module.
a	CAMIF_DATA6	I	GPIO[60]								
a	CAMIF_DATA5	I	GPIO[59]]								
a	CAMIF_DATA4	I	GPIO[58]								
	CAMIF_DATA3	I	GPIO[57]								
a	CAMIF_DATA2	I	GPIO[56]								
a	CAMIF_DATA1	I	GPIO[55]								
a	CAMIF_DATA0	I	GPIO[54]								
a	CAMIF_VSYNC	B	GPIO[2]						3-5	1.5-2.5	Input vertical reference signal for ITU656-like interface. Intended for camera application.
a	CAMIF_HSYNC	B	GPIO[1]						3-5	1.5-2.5	Input horizontal reference signal for ITU656-like interface. Intended for camera application.
a	CAMIF_PCLK	B	GPIO[0]						3-5	1.5-2.5	Input pixel clock for ITU656-like interface. Intended for camera application.
RF RX Interface [4 dedicated pins]											
Y23	I_P	I						IA			Differential analog I signal (+)
Y24	I_N	I						IA			Differential analog I signal (-)
W24	Q_P	I						IA			Differential analog Q signal (+)
W23	Q_N	I						IA			Differential analog Q signal (-)
a	FM_LNA_RANGE	Z	GPIO[79]						2-5	1-2.5	Digital (wired-OR) outputs from the receive AGC circuit which can be used to alter the low-noise amplifier characteristics or other stages in the subscriber receive signal path.

Table 2-2 Pin Names and Pinouts (Continued)

pin #	Native Mode				ETM Mode		Volt	Pad_ Type	2.6V Dr mA	1.867V Dr mA	Description
	Main Function	Dir-Pol	Alternate Functions {SURF Functions}	Dir-Pol	{SURF Functions}	Dir-Pol					
RF TX Interface [10 dedicated pins]											
H20	I_OUT	O						OA			Non-inverted current mode output from the I transmit DAC
G20	I_OUT_N	O						OA			Inverted current mode output from the I transmit DAC
D24	Q_OUT	O						OA			Non-inverted current mode output from the Q transmit DAC
E24	Q_OUT_N	O						OA			Inverted current mode output from the Q transmit DAC
E23	DAC_REF	I						IA			Input reference to set the gain of the I&Q transmit DACs.
C23 ***F24***	PA_ON[1] PA_ON[0]	O O						O, P4	5	2.5	Control signal that controls the power amplifier. PA_ON is high only when the RF power amplifier is needed for transmission. Supports up to two power amplifiers. There is a design requirement that both PA's are powered off on the hardware reset.
a	TX_ON	O	GPIO[17]					O, P4	5	2.5	
P21 ***P20***	PA_R1 PA_R0	O O						Z, P4	2	1	Digital (wired-OR) outputs from the transmit AGC circuit which can be used to alter the subscriber transmit power amplifier characteristics.
R24	TX_AGC_ADJ	O						Z-PD, P4	1-5	0.5-2.5	PDM output from the transmit AGC circuit to control the transmit output power.
PA control DAC signals [1 dedicated pin]											
M23	Reserved							OA			
Other RF Interface [2 dedicated pins]											
Y14	SYNTH_LOCK	I						IS, P4			Indicates the lock status of the frequency synthesizer.
L23	TRK_LO_ADJ	Z						Z, P4	1-5	0.5-2.5	PDM output from the frequency tracking circuit which sets the subscriber VHF and UHF frequencies.
General Purpose ADC Signal [7 dedicated pins]											
AA21 ***AB23*** ***W20*** ***AA23*** ***Y21*** ***W21*** ***V20***	HKADC_REF HKAIN[5] HKAIN[4] HKAIN[3] HKAIN[2] HKAIN[1] HKAIN[0]							IA			Analog mux input channels to the on-chip house-keeping ADC.
Miscellaneous I/O [0 dedicated pins]											
a	GP_PDM[2]	O-PD	GPIO[15]					BS-PD, P4	1-5	0.5-2.5	General purpose PDM. Clocked at TCXO/4 rate. Both microprocessor and DSP control the PDM.
a	GP_PDM[1]	O-PD	GPIO[16]					BS-PD, P4	1-5	0.5-2.5	
a	GP_PDM[0]	O-PD	GPIO[10]					BS-PD, P4	1-5	0.5-2.5	
a	GP_MN	O	GPIO[80]					BS-PD, P4	1-5	0.5-2.5	General purpose processor-controlled M/N counter. Clocked at TCXO/4 rate.
JTAG Pins [6 dedicated pins]											
AD13	TRST_N	IS-PU						BS-PU, P4			JTAG reset. On the handset design, this pin is connected to resin, thus pull up.
Y13	TCK	IS-PU						BS-PU, P4			JTAG clock input
AC13	TMS	IS-PU						BS-PU, P4			JTAG mode select input
AC14	TDI	IS-PU						BS-PU, P4			JTAG data input
AA13	TDO	Z						Z, P4	5	2.5	JTAG data output
K13	RTCK	O						P4	5	2.5	Internally bonded to K14. Return clock for JTAG Debugger.
K14	RTCK	O						P4	5	2.5	Internally bonded to K13, only necessary to connect one ball. Return clock for JTAG Debugger.
Power/Ground [51 dedicated pins]											
M15 ***N15*** ***U23*** ***AA4*** ***D16*** ***Y12*** ***A9*** ***K5***	VDD_C VDD_C VDD_C VDD_C VDD_C VDD_C VDD_C VDD_C										Digital VDD inputs 1.876V nominal. M15 and N15 are internally bonded together.
N10 ***P10*** ***G23*** ***D19*** ***AD10*** ***B10*** ***J4***	GND GND GND GND GND GND GND										Ground. N10 and P10 are internally bonded together.

Table 2-2 Pin Names and Pinouts (Continued)

pin #	Native Mode				ETM Mode		Volt	Pad_ Type	2.6V Dr mA	1.867V Dr mA	Description
	Main Function	Dir-Pol	Alternate Functions {SURF Functions}	Dir-Pol	{SURF Functions}	Dir-Pol					
K11	VDD_P1										Vdd for Pad group 1. 1.8V nominal. K11 and K12 are internally bonded together.
K12	VDD_P1										
E2	VDD_P1										
N1	VDD_P1										
L10	GND										Ground. L10 and M10 are internally bonded together
M10	GND										
D4	GND										
L2	GND										
Y5	VDD_P2										Vdd for pad group 2. Digital VDD for pad I/O (2.60V nominal)
AA5	GND										Ground
B7	VDD_P3										Vdd for pad group 3 Digital VDD for pad I/O
B19	VDD_P4										Vdd for pad group 4.
B12	VDD_P4										
R21	VDD_P4										
B21	GND										Ground
E11	GND										
R23	GND										
AA20	VDD_A										Analog VDD for the CODEC
AC21	GND										Ground
Y15	VDD_A										Analog VDD for the earphone amplifier
AC16	GND										Ground
G21	VDD_A										Analog VDD for the integrated TX DAC
F23	GND										Ground
B17	VDD_A										Analog VDD for the digital clock generation PLL
D17	GND										Ground
B18	GND										Ground
AA24	GND										Ground
AC22	GND_RET										Analog return ground for the CODEC. Decoupling capacitor is connected between CCOMP and GND_RET.
V23	VDD_A										Analog VDD for the baseband sigma-delta modulator.
U21	VDD_A										
V21	GND										Ground
U20	GND										
M24	VDD_A										Analog VDD.
M20	GND										Ground
AC15	VDD_A										Analog VDD for Stereo DAC.
AA15	GND										Ground

2.3 341-Ball CSP Pinout for MSM6100 Device (Top View)

Table 2-3 MSM6100 Pinouts (1)

	1	2	3	4	5	6	7	8	
A	N/C	N/C	N/C	A1[22]	GPIO32 (A1[24])	GPIO71 (D1[19])	GPIO68 (D1[16])	GPIO51 (APKT5)	A
B	N/C	N/C	D1[15]	A1[16]	GPIO31 (A1[23])	GPIO73 (D1[21])	VDD_P3	GPIO69 (D1[17])	B
C	N/C	D1[6]							C
D	A1[13]	A1[12]		GND	A1[14]	A1[15]	GPIO70 (D1[18])	GPIO74 (D1[22])	D
E	A1[11]	VDD_P1		D1[7]	D1[14]	D1[13]	GPIO72 (D1[20])	GPIO76 (DQM3)	E
F	D1[4]	A1[9]		A1[8]	A1[10]				F
G	A1[21]	A1[19]		D1[5]	D1[12]				G
H	D1[2]	D1[10]		D1[11]	D1[3]				H
J	A1[18]	A1[20]		GND	WE1_N				J
K	D1[9]	D1[1]		ROM1_WAIT_N	VDD_C				K
L	UB1_N	GND		D1[0]	D1[8]				L
M	A1[7]	A1[17]		A1[6]	A1[5]				M
N	VDD_P1	A1[4]		A1[2]	A1[1]				N
P	LB1_N	ROM1_CS_N[0]		RAM1_CS_N[0]	OE1_N				P
R	A1[3]	ROM1_CLK		SDRAM1 _CLK_EN	ROM1_ADV_N				R
T	GPIO34 (ROM1_CS_N[1])	RESOUT_ N_EMI1		GPIO77 (RAM1_CS_N[1])	A2[19]				T
U	GPIO78 (A2[20])	A2[16]		A2[18]	A2[13]				U
V	A2[17]	A2[15]		A2[10]	A2[12]				V
W	A2[14]	A2[11]		A2[6]	A2[3]				W
Y	A2[9]	A2[7]		A2[5]	VDD_P2	D2[15]	XTAL48_OUT	D2[9]	Y
AA	A2[8]	A2[4]		VDD_C	GND	XTAL48_IN	D2[12]	D2[6]	AA
AB	N/C	A2[2]							AB
AC	N/C	N/C	A2[1]	D2[14]	D2[13]	D2[10]	D2[7]	D2[4]	AC
AD	N/C	N/C	N/C	LB2_N	D2[11]	D2[8]	D2[5]	D2[2]	AD
	1	2	3	4	5	6	7	8	

Table 2-4 MSM6100 Pinouts (2)

	9	10	11	12	13	14	15	16	
A	VDD_C	GPIO66(ECLK)	GPIO48(APKT2)	GPIO65(SYNCA)	GPIO62(APS0)	GPIO58(BPKT4)	GPIO54(BPKT0)	GPIO44(BPS0)	A
B	GPIO53(APKT7)	GND	GPIO47(APKT1)	VDD_P4	GPIO61(BPKT7)	GPIO57(BPKT3)	GPIO45(SYNCB)	GPIO7 (MMC_CLK)	B
C									C
D	GPIO52(APKT6)	GPIO50(APKT4)	GPIO46(APKT0)	GPIO64(APS2)	GPIO60(BPKT6)	GPIO56(BPKT2)	GPIO42(BPS2)	VDD_C	D
E	GPIO75 (D1[23])	GPIO49(APKT3)	GND	GPIO63(APS1)	GPIO59(BPKT5)	GPIO55(BPKT1)	GPIO43(BPS1)	GPIO8 (MMC_CMD)	E
F									F
G									G
H									H
J	N/C								J
K		N/C	VDD_P1		RTCK		N/C		K
L		GND	N/C	N/C	N/C	N/C	N/C		L
M			N/C	N/C	N/C	N/C	VDD_C		M
N		GND	N/C	N/C	N/C	N/C			N
P			N/C	N/C	N/C	N/C	N/C		P
R		N/C	N/C	N/C	N/C	N/C	N/C		R
T									T
U									U
V									V
W									W
Y	D2[3]	WE2_N	GPIO36 (GP2_CS_N[1])	VDD_C	TCK	SYNTH_LOCK	VDD_A	AUXOP	Y
AA	D2[0]	GPIO38 (LCD2_CS_N)	GPIO35 (GP2_CS_N[0])	GPIO41	TDO	GPIO39	GND	EAR1OP	AA
AB									AB
AC	D2[1]	GPIO37 (LCD2_EN)	GPIO67 (RAM2_CS_N)	OE2_N	TMS	TDI	VDD_A	GND	AC
AD	UB2_N	GND	GPIO33 (ROM2_CS_N)	GPIO30 (NANDF_READY)	TRST_N	GPIO40	HPH_R	EAR1ON	AD
	9	10	11	12	13	14	15	16	

Table 2-5 MSM6100 Pinouts (3)

	17	18	19	20	21	22	23	24	
A	WDOG_EN	TCXO	GPIO9 (MMC_DATA)	GPIO24(UART3 _DP_TX_DATA)	GPIO11(UART3 _DP_RX_DATA)	N/C	N/C	N/C	A
B	VDD_A	GND	VDD_P4	BOOT_MODE	GND	GPIO28(UART2 _DP_TX_DATA	N/C	N/C	B
C							PA_ON[1]	N/C	C
D	GND	GPIO12(UART3 _CTS_N)	GND	GPIO13(UART2 _CTS_N)	GPIO81		DP_RX_DATA	Q_OUT	D
E	GPIO25(UART3 _RFR_N)	GPIO29(UART2 _DP_RX_DATA)	GPIO14(UART2 _RFR_N)	GPIO80	CTS_N		DAC_REF	Q_OUT_N	E
F				DP_TX_DATA	GPIO4 (PCM_DIN)		GND	PA_ON[0]	F
G				I_OUT_N	VDD_A		GND	GPIO3 (PCM_SYNC)	G
H				I_OUT	GPIO5 (PCM_DOUT)		RFR_N	GPIO20 (USB_RX_DATA)	H
J				GPIO6 (PCM_CLK)	GPIO23 (USB_TX_VPO)		GPIO22 (USB_TX_VMO)	GPIO21 (USB_TX_OE_N)	J
K				GPIO2	GPIO18 (USB_RX_VPI)		GPIO19 (USB_RX_VMI)	GPIO1	K
L				GPIO0	GPIO27 (I2CSCL)		TRK_LO_ADJ	GPIO26 (I2C_SDA)	L
M				GND	GPIO79 (FM_LNA _RANGE)		N/C (Reserved)	VDD_A	M
N				MODE[1]	RESOUT_N		GPIO15 (GP_PDM2)	GPIO17 (TX_ON)	N
P				PA_R0	PA_R1		GPIO16 (GP_PDM1)	GPIO10 (GP_PDM0)	P
R				MODE[0]	VDD_P4		GND	TX_AGC_ADJ	R
T				SBCK	SLEEP _XTAL_IN		SBST	SLEEP _XTAL_OUT	T
U				GND	VDD_A		VDD_C	SBDT	U
V				HKAIN[0]	GND		VDD_A	RESIN_N	V
W				HKAIN[4]	HKAIN[1]		Q_N	Q_P	W
Y	MIC2N	MIC2P	MICBIAS	CCOMP	HKAIN[2]		I_P	I_N	Y
AA	AUXON	MICOUTP	MICINN	VDD_A	HKADC_REF		HKAIN[3]	GND	AA
AB							HKAIN[5]	N/C	AB
AC	AUXIP	MIC1N	MICOUTN	MICINP	GND	GND_RET	N/C	N/C	AC
AD	EAR2O	AUXIN	MIC1P	MICFBP	MICFBN	N/C	N/C	N/C	AD
	17	18	19	20	21	22	23	24	

3 Electrical Specifications

3.1 DC Electrical Specifications

3.1.1 Absolute Maximum Ratings

Operating the MSM6100 device under conditions that exceed those listed in Table 3-1 may result in damage to the device. Absolute maximum ratings are limiting values, and are considered individually, while all other parameters are within their specified operating ranges. Functional operation of the MSM6100 device under any of the conditions in Table 3-1 is not implied. Exposure to Absolute Maximum Ratings for extended periods of time may affect the device's reliability.

Table 3-1 Absolute Maximum Ratings

Symbol	Parameter	Min	Max	Units
T_S	Storage temperature	–65	+150	°C
T_J	Junction temperature	—	TBD	°C
V_I	Voltage on any input or output pin	–0.3	V_{DD} + 0.3	V
V_{DD}	Supply voltage	TBD	TBD	V
I_{IN}	Latchup current	TBD	TBD	mA
V_{ESD}	Electrostatic discharge voltage (Human Body Model)	–2000	+2000	V
V_{ESD}	Electrostatic discharge voltage (Charge Device Model)	TBD	TBD	V

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.1.2 Recommended Operating Conditions

Table 3-2 Recommended Operating Conditions

Parameter	Symbol	Min	Typ	Max	Units
Operating temperature (case)	T_C	–30		85	°C
Supply voltage for internal (core)	V_{DD_C}	1.77	1.867	1.96	V
Supply voltage for analog core	V_{DD_A}	2.50	2.60	2.70	V
Supply voltage for pad1	V_{DD_P1}	1.77	1.867	1.96	V
Supply voltage for pad2 (configurable) [a]	V_{DD_P2}	1.77 2.50	1.867 2.60	1.96 2.70	V V
Supply voltage for pad3 (configurable) [b]	V_{DD_P3}	1.77 2.50	1.867 2.60	1.96 2.70	V V
Supply voltage for pad4 (configurable) [c]	V_{DD_P4}	1.77 2.50	1.867 2.60	1.96 2.70	V V

[a] This voltage must match the memory supply voltage. It is a dual voltage pin and can be either 1.867 VDC nominal or 2.60 VDC nominal.

[b] Same.

[c] Same.

NOTE V_{DD_P3} must be tied to V_{DD_P1} when 32 bit SDRAM is used. This is not a requirement when 16 bit SDRAM is used.

Table 3-3 Thermal Resistance

Parameter	Description	Typ	Units
θ_{JA}	Junction to still air	TBD	°C/W
θ_{JC}	Junction to case	TBD	°C/W

3.1.3 DC Characteristics

Table 3-4 DC Characteristics (for VDD_P0 = 1.867)

Parameter	Description	Min	Max	Units	Notes
V_{IH}	High-level input voltage, CMOS/Schmitt	0.65 • V_{DD_P}	V_{DD_P} + 0.3	Volts	—
V_{IL}	Low-level input voltage, CMOS/Schmitt	–0.3	0.35 • V_{DD_P}	Volts	—
V_{SHYS}	Schmitt Hysteresis Voltage	100	—	mV	
I_{IH}	Input high leakage current	—	1	µA	a, b
I_{IL}	Input low leakage current	–1	—	µA	a, b
I_{IHPD}	Input high leakage current with pull-down	3	30	µA	a, b

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-4 DC Characteristics (for VDD_P0 = 1.867)

Parameter	Description	Min	Max	Units	Notes
I_{ILPU}	Input low leakage current with pull-up	–30	–3	µA	b,c
I_{OZH}	High-level, three-state leakage current	—	1	µA	a
I_{OZL}	Low-level, three-state leakage current	–1	—	µA	b
I_{OZHPD}	High-level, three-state leakage current with pull down	3	30	µA	a,c
I_{OZLPU}	Low-level, three-state leakage current with pull up	–30	–3	µA	b,c
I_{OZHKP}	High-level, three-state leakage current with keeper	–15	–3	µA	a,c
I_{OZLKP}	Low-level, three-state leakage current with keeper	3	15	µA	b, c
V_{OH}	High-level output voltage, CMOS, when driving pin at rated drive strength	V_{DD_P} – 0.3	V_{DD_P}	Volts	d, e
V_{OL}	Low-level output voltage, CMOS, when driving pin at rated drive strength	0.0	0.3	Volts	d, e
V_{OHQ}	High-level output voltage, CMOS, when driving pin at a one-fourth of rated drive strength	V_{DD_P} – 0.1	V_{DD_P}	Volts	d, e
V_{OLQ}	Low-level output voltage, CMOS, when driving pin at one-fourth of rated drive strength	0.0	0.1	Volts	d, e
C_{IN}	Input capacitance	—	10	pF	—
I_{ISL}	Sleep xtal input leakage	TBD	TBD	µA	—
I_{IUXTAL}	USB xtal input leakage	TBD	TBD	µA	—
I_{IHVKP}	High voltage input leakage with keeper	–3	—	µA	—
I_{ILVKP}	Low voltage input leakage with keeper	—	3	µA	—

[a] Pin voltage = V_{DD_P} max. For keeper pins, pin voltage = V_{DD_P} max – 0.45 volts.
[b] Pin voltage = Vss and V_{DD_P} = V_{DD_P} max. For keeper pins, pin voltage = 0.45 volts and V_{DD_P} = V_{DD_P} max.
[c] Refer to Table 2-2 for pull-up, pull-down, and keeper information on pins.
[d] Refer to Table 2-2 for I_{OH} and I_{OL} rated drive strength (current capacity) for output pins (at V_{DD_P} = V_{DD_P} min).
[e] The rated drive strength is found in Table 2-2, under the column titled “1.867V Dr mA”.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-5 DC Characteristics (for VDD_P0 = 2.60)

Parameter	Description	Min	Max	Units	Notes
V_{IH}	High-level input voltage, CMOS/Schmitt	0.65 • V_{DD_P}	V_{DD_P} + 0.3	Volts	—
V_{IL}	Low-level input voltage, CMOS/Schmitt	–0.3	0.35 • V_{DD_P}	Volts	—
V_{SHYS}	Schmitt Hysteresis Voltage	150	—	mV	
I_{IH}	Input high leakage current	—	1	µA	a, b
I_{IL}	Input low leakage current	–1	—	µA	a, b
I_{IHPD}	Input high leakage current with pull-down	10	60	µA	a, b
I_{ILPU}	Input low leakage current with pull-up	–60	–10	µA	b,c
I_{OZH}	High-level, three-state leakage current	—	1	µA	a
I_{OZL}	Low-level, three-state leakage current	–1	—	µA	b
I_{OZHPD}	High-level, three-state leakage current with pull down	10	60	µA	a,c
I_{OZLPU}	Low-level, three-state leakage current with pull up	–60	–10	µA	b,c
I_{OZHKP}	High-level, three-state leakage current with keeper	–25	–5	µA	a,c
I_{OZLKP}	Low-level, three-state leakage current with keeper	5	25	µA	b, c
V_{OH}	High-level output voltage, CMOS, when driving pin at rated drive strength	V_{DD_P} – 0.45	V_{DD_P}	Volts	d, e
V_{OL}	Low-level output voltage, CMOS, when driving pin at rated drive strength	0.0	0.45	Volts	d, e
V_{OHQ}	High-level output voltage, CMOS, when driving pin at a one-fourth of rated drive strength	V_{DD_P} – 0.15	V_{DD_P}	Volts	d, e
V_{OLQ}	Low-level output voltage, CMOS, when driving pin at one-fourth of rated drive strength	0.0	0.15	Volts	d, e
C_{IN}	Input capacitance	—	10	pF	—
I_{ISL}	Sleep xtal input leakage	TBD	TBD	µA	—
I_{IUXTAL}	USB xtal input leakage	TBD	TBD	µA	—
I_{IHVKP}	High voltage input leakage with keeper	–3	—	µA	—
I_{ILVKP}	Low voltage input leakage with keeper	—	3	µA	—

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

[a] Pin voltage = V_{DD_P} max. For keeper pins, pin voltage = V_{DD_P} max – 0.45 volts.
[b] Pin voltage = Vss and V_{DD_P} = V_{DD_P} max. For keeper pins, pin voltage = 0.45 volts and V_{DD_P} = V_{DD_P} max.
[c] Refer to Table 2-2 for pins having pull-ups, pull-downs, and keepers.
[d] Refer to Table 2-2 for I_{OH} and I_{OL} rated drive strength (current capacity) for output pins (at V_{DD_P} = V_{DD_P} min).
[e] The rated drive strength is found in Table 2-2, under the column titled “2.6V Dr mA”.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.1.4 GPADC Specifications

Table 3-6 GPADC Performance Specification

Parameter	Min	Typ	Max	Units	Comments/Conditions
Resolution	—	8 Bits	—	—	
DNL	—	—	± 0.75	LSB	a
INL	—	—	± 1.5	LSB	a
Full-Scale Error	—	—	TBD	LSB	a,b
Offset Error	—	—	TBD	LSB	a
Internal Reference Voltage	—	—	± 3	%	0.62 V, 1.24 V, 1.86 V
Channel Isolation	—	50	—	dB	at DC
Full-Scale Input Range	GND	—	V_{RT} (Vref)	—	V_{RT} is variable 0.62 V to V_{DD_A}
3 dB Input Bandwidth	—	2.65	—	MHz	source resistance = 50 Ω
Input Serial Resistance	—	TBD	—	kΩ	When input is selected. c
Input Capacitance	—	12	—	pF	c
Powerdown to Wakeup	—	—	5	uS	
Throughput Rate	20.8	—	170.7	kHz	
Leakage Current	—	—	TBD	nA	

a Specifications are only valid for the following two sets of conditions:
1. V_{RT} (Vref) levels from 0.62V to V_{DD_A}, with an input clock frequency range of 300 kHz to 309.375 kHz.

or

2. V_{RT} equals V_{DD_A} with an input frequency range of 300 kHz to 1.2375 MHz.

b Only valid for a fixed external reference.

c See Chapter 4, for more information (Not available until Preliminary Device Specification).

Note: For an acquisition time of 3 clk cycles (CLK period), the CLK period should be greater than 7• τ, where τ = (input resistance + source resistance) • input capacitance.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.1.5 CODEC Specifications

Table 3-7 Microphone Interface Requirements

Parameter		Test Conditions	Min	Typ	Max	Unit
V_{IO}	Input offset voltage at MIC1, MIC2 and AUX inputs	Over recommended ranges of supply voltage and free-air temperature	–5		+5	mV
C_I	Input capacitance at MIC1, MIC2 and AUX inputs			5		pF
	Input DC common mode voltage		0.9	1.0	1.1	V
V_{mbias}	Microphone bias supply voltage	Open circuit DC voltage	1.69	1.8	1.91	V
	MBIAS output DC source current	1.69 kΩ 1% resistive load	1	1.07		mA
	MICMUTE attenuation	+3 dBm0 analog input level 1.02 kHz sine-wave	80			dB
Z_{in}	Input impedance, MIC1, MIC2 and AUX inputs	Fully differential	61	72	83	kΩ

Table 3-8 Speaker Interface Requirements

Parameter		Test Conditions	Min	Typ	Max	Unit
P_{O1}	EAR_AMP1 output power (rms)	Differential, 32 Ω load, PCMI = +3 dBm0, 1.02 kHz sine-wave		70		mW
P_{O2}	EAR_AMP2 output power (rms)	Single ended, 16 Ω load, PCMI = +3 dBm0, 1.02 kHz sine-wave		21.6		mW
P_{O3}	EAR_AMP3 output power (rms)	Single ended, 16 Ω load, PCMI = +3 dBm0, 1.02 kHz sine-wave		21.6		mW
P_{O4}	AUX_AMP output power (rms)	Differential, 600 Ω load, PCMI = +3 dBm0, 1.02 kHz sine-wave		2.30		mW
P_{O5}	AUX Amp output power (rms)	Single ended, 600 Ω load, PCMI = +3 dBm0, 1.02 kHz sine-wave (AUXOP, AUXON)		0.58		mW
	Output DC offset voltage between EAR1OP and EAR1ON, AUXOP and AUXON	Fully differential	–50		50	mV
	Output common mode voltage, EAR1OP, EAR1ON, EAR2O, AUXOP, AUXON	Measured at each output pin with respect to GNDGND: Vdd = 2.5 V TO 2.7 V	1.125	1.25	1.375	V
Z_{OUT1}	Differential Output Impedance	At 1.02 kHz, for outputs EAR1 and AUX amp			1	Ω
Z_{OUT2}	Single-Ended Output Impedance	At 1.02 kHz, for output EAR2 and AUX amps			0.5	Ω
THD	Total harmonic distortion +Noise (voice)	AV_{DD}= 2.5 V, 13-bit mode, PCMI = +3 dBm0, 498 Hz sine- wave, 32 Ω load for EAR1 and EAR2 amps, 600 Ω load for differential line outputs.			4	%

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-8 Speaker Interface Requirements (Continued)

Parameter		Test Conditions	Min	Typ	Max	Unit
THD	Total harmonic distortion + Noise (audio)	AV_{DD}= 2.5 V, 16-bit mode, PCMI = +3 dBm0, 1.02 kHz and 5 kHz sine waves, 32 Ω load for EAR2 amps, 600 Ω load for single-ended line outputs.		0.05	0.1	%
	EARMUTE attenuation	PCMI = +3 dBm0, 1.02 kHz sine-wave	80			dB

Notes:

- +3 dBm0 level corresponds to 0 dB full-scale sine wave.
- RXPGA = 0 dB

Table 3-9 Transmit VOICE Path Level Translation and Linearity, MIC AMP2 Enabled

Parameter	Test Conditions	Min	Typ	Max	Unit
Transmit reference-signal level (0 dBm0)	Differential analog input		57.3		mV_{rms}
Overload-signal level (+3 dBm0)	Differential analog input		229		mV_{pp}
Overload-signal level (+3 dBm0) at the analog modulator input (MICFBN, MICFBP)	Differential analog input		3.626		V_{pp}
Absolute gain error **	0 dBm0 analog input level, 1.02 kHz sine-wave	–1		1	dB
Gain error relative to gain at -10 dBm0	Analog input level from +3 dBm0 to -30 dBm0	–0.5		0.5	dB
Gain error relative to gain at -10 dBm0	Analog input level from -31 dBm0 to -45 dBm0	–1		1	dB
Gain error relative to gain at -10 dBm0	Analog input level from -46 dBm0 to -55 dBm0	–1.2		1.2	dB

Notes:

- These specifications apply to all three Tx path inputs: Mic inputs 1 and 2 and Aux In.
- The total transmit channel gain in this default configuration is +24 dB (microphone amplifier 1 is set to +6 dB, the external gain is set to +18 dB, and the TXPGA is set to 0 dB).
- Fs = Sampling rate, 8 kHz or 16 kHz

** This spec must be applicable to all microphone amplifier gain settings (i.e., -2 dB, +6 dB, +8 dB, +16 dB)

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-10 Transmit VOICE Path Level Translation and Linearity, MIC AMP2 Bypassed

Parameter	Test Conditions	Min	Typ	Max	Unit
Transmit reference-signal level (0 dBm0)	Differential analog input		455		mV_{rms}
Overload-signal level (+3 dBm0)	Differential analog input		1.82		V_{pp}
Overload-signal level (+3 dBm0) at the analog modulator input (MICFBN, MICFBP)	Differential analog input		3.626		V_{pp}
Absolute gain error **	0 dBm0 analog input level, 1.02 kHz sine-wave	–1		1	dB
Gain error relative to gain at –10 dBm0	Analog Input Level from +3 dBm0 to –30 dBm0	–0.5		0.5	dB
Gain error relative to gain at –10 dBm0	Analog Input Level from –31 dBm0 to –45 dBm0	–1		1	dB
Gain error relative to gain at –10 dBm0	Analog Input Level from –46 dBm0 to –55 dBm0	–1.2		1.2	dB

Notes:
- These specifications apply to all three Tx path inputs: Mic inputs 1 and 2 and Aux In.
- The total transmit channel gain in this default configuration is +6 dB (microphone amplifier 1 is set to +6 dB and the TXPGA is set to 0 dB).
- Fs = Sampling rate, 8 kHz or 16 kHz

** This spec must be applicable to all microphone amplifier gain settings (i.e., -2 dB, +6 dB, +8 dB, +16 dB)

Table 3-11 Transmit VOICE Path Frequency Response and Image Rejection, Digital Transmit Slope Filter Disabled, Mic Amp 2 Bypassed

Parameter	Test Conditions	Min	Typ	Max	Unit
Gain relative to input signal gain at 1.02 kHz, digital Tx highpass filter disabled.	Frequency <0.0125xFs Hz	–0.5		0.5	dB
	Frequency = 0.025xFs Hz	–0.5		0.5	dB
	Frequency = 0.0375xFs Hz to 0.375xFs Hz	–0.5		0.5	dB
	Frequency = 0.425xFs Hz	–1.5		0	dB
	Frequency = 0.4975xFs Hz			–14	dB
	Frequency = 0.575xFs Hz			–35	dB
	Frequency = 0.9975xFs Hz			–47	dB
Gain relative to input signal gain at 1.02 kHz, digital Tx highpass filter enabled.	Frequency < 0.0125xFs Hz			–15	dB
	Frequency = 0.025xFs Hz			–5	dB

Notes:
- The Tx path input level is 0 dBm0
- Frequencies 0.575xFs Hz and 0.9975xFs Hz are used to determine image rejection performance.
- Fs = Sampling rate, 8 kHz or 16 kHz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-12 Transmit VOICE Path Frequency Response and Image Rejection, Digital Transmit Slope Filter Enabled, Mic Amp 2 Bypassed

Parameter	Test Conditions	Min	Typ	Max	Unit
Gain relative to input signal gain at 1.02 kHz, with slope filter selected, digital Tx highpass filter enabled	Frequency = 0.0125xFs Hz			–27	dB
	Frequency = 0.025xFs Hz			–8	dB
	Frequency = 0.03125xFs Hz			–4	dB
	Frequency = 0.0375xFs Hz		–1.80		dB
	Frequency = 0.05xFs Hz		–1.50		dB
	Frequency = 0.0625xFs Hz		–1.30		dB
	Frequency = 0.075xFs Hz		–1.1		dB
	Frequency = 0.0875xFs Hz		–0.8		dB
	Frequency = 0.1xFs Hz		–0.57		dB
	Frequency = 0.1125xFs Hz		–0.25		dB
	Frequency = 0.1275xFs Hz		0		dB
	Frequency = 0.1875xFs Hz		1.8		dB
	Frequency = 0.2475xFs Hz		4.0		dB
	Frequency = 0.3125xFs Hz		6.5		dB
	Frequency = 0.375xFs Hz		7.6		dB
	Frequency = 0.3875xFs Hz		7.7		dB
	Frequency = 0.4125xFs Hz		8.0		dB
	Frequency = 0.4375xFs Hz		6.48		dB
	Frequency = 0.4975xFs Hz			–13	dB
	Frequency = 0.5625xFs Hz			–35	dB
	Frequency = 0.625xFs Hz			–45	dB
	Frequency = 0.9975xFs Hz			–50	dB

Notes:

- The passband tolerance is ± 0.25 dB from 300 Hz to 3500 Hz.
- The Tx path input level is 0 dBm0
- Frequencies 0.5625xFs, 0.625xFs and 0.9975xFs Hz are used to determine image rejection performance.
- Fs = Sampling rate, 8k Hz or 16k Hz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-13 Transmit VOICE Path Idle Channel Noise and Distortion (8K)

Parameter	Test Conditions	Min	Typ	Max	Unit
Transmit noise	TXPGA gain = 0 dB, external gain = +18 dB, microphone amplifier 1 gain = +6 dB		10	15	μVrms
Transmit signal-to-THD+N ratio with 1020Hz sine-wave input	Analog Input Level at +3 dBm0	35			dB
	Analog Input Level at 0 dBm0	50			dB
	Analog Input Level at –5 dBm0	50			dB
	Analog Input Level at –10 dBm0	46			dB
	Analog Input Level at –20 dBm0	45			dB
	Analog Input Level at –30 dBm0	40			dB
	Analog Input Level at –40 dBm0	30			dB
	Analog Input Level at –45 dBm0	25			dB

Notes:
- Specifications must be met with and without Tx Slope Filter enabled.
- Specifications must be met for all inputs MIC1, MIC2 and AUX.
- C-message weighted for 8K sampling rate
- Fs = Sampling rate, 8 kHz

Table 3-14 Transmit VOICE Path Idle Channel Noise and Distortion (16K)

Parameter	Test Conditions	Min	Typ	Max	Unit
Transmit noise	TXPGA gain = 0 dB, external gain = +18 dB, microphone amplifier 1 gain = +6 dB		12	22	μVrms
Transmit signal-to-THD+N ratio with 1020 Hz sine-wave input	Analog Input Level at +3 dBm0	35			dB
	Analog Input Level at 0 dBm0	46			dB
	Analog Input Level at –5 dBm0	46			dB
	Analog Input Level at –10 dBm0	42			dB
	Analog Input Level at –20 dBm0	41			dB
	Analog Input Level at –30 dBm0	36			dB
	Analog Input Level at –40 dBm0	27			dB
	Analog Input Level at –45 dBm0	21			dB

Notes:
- Specifications must be met with and without Tx Slope Filter enabled.
- Specifications must be met for all inputs MIC1, MIC2 and AUX.
- A-weighted for 16K sampling rate
- Fs = Sampling rate, 16 kHz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-15 Receive VOICE Path Level Translation and Linearity, EAR_AMP1 Selected

Parameter	Test Conditions	Min	Typ	Max	Unit
Receive reference-signal level (0 dBm0)	PCMI = 0 dBm0, 1.02 kHz sine-wave		1.06		V_{rms}
Overload-signal level (+3 dBm0)	PCMI = +3 dBm0, 1.02 kHz sine-wave		4.24		V_{pp}
Absolute gain error	PCMI = 0 dBm0, 1.02 kHz sine-wave	–1		+1	dB
Gain error relative to gain at –10 dBm0	PCMI = +3 dBm0 to –40 dBm0	–0.5		+0.5	dB
Gain error relative to gain at –10 dBm0	PCMI = –41 dBm0 to –50 dBm0	–1		+1	dB
Gain error relative to gain at –10 dBm0	PCMI = –51 dBm0 to –55 dBm0	–1.2		+1.2	dB

Notes:
- RXPGA = 0 dB
- Output measured differentially between EAR1ON and EAR1OP
- +3 dBm0 level corresponds 0 dB full-scale sine-wave
- Loaded condition (32 ohm)
- 13-bit mode
- Fs = Sampling rate = 8 kHz or 16 kHz

Table 3-16 Receive VOICE Path Level Translation and Linearity, EAR_AMP2 Selected

Parameter	Test Conditions	Min	Typ	Max	Unit
Receive reference-signal level (0 dBm0)	PCMI = 0 dBm0, 1.02 kHz sine-wave		416.5		mV_{rms}
Overload-signal level (+3 dBm0)	PCMI = +3 dBm0, 1.02 kHz sine-wave		1.664		V_{pp}
Absolute gain error	PCMI = 0 dBm0, 1.02 kHz sine-wave	–1		+1	dB
Gain error relative to gain at –10 dBm0	PCMI = +3 dBm0 to –40 dBm0	–0.5		+0.5	dB
Gain error relative to gain at –10 dBm0	PCMI = –41 dBm0 to –50 dBm0	–1		+1	dB
Gain error relative to gain at –10 dBm0	PCMI = –51 dBm0 to –55 dBm0	–1.2		+1.2	dB

Notes:
- RXPGA = 0 dB
- Output measured single-ended between EAR20 and GND
- Output measured single-ended with 32 Ω load and DC blocking capacitor (220 uF) connected between EAR20 and GND
- +3 dBm0 level corresponds to 0 dB full-scale sine wave.
- 13-bit mode
- Fs = Sampling rate = 8K or 16K Hz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-17 Receive VOICE Path Level Translation and Linearity, EAR_AMP3 Selected

Parameter	Test Conditions	Min	Typ	Max	Unit
Receive reference-signal level (0 dBm0)	PCMI = 0 dBm0, 1.02 kHz sine-wave		416.5		mV_{rms}
Overload-signal level (+3 dBm0)	PCMI = +3 dBm0, 1.02 kHz sine-wave		1.664		V_{pp}
Absolute gain error	PCMI = 0 dBm0, 1.02 kHz sine-wave	–1		+1	dB
Gain error relative to gain at –10 dBm0	PCMI = +3 dBm0 to –40 dBm0	–0.5		+0.5	dB
Gain error relative to gain at –10 dBm0	PCMI = –41 dBm0 to –50 dBm0	–1		+1	dB
Gain error relative to gain at –10 dBm0	PCMI = –51 dBm0 to –55 dBm0	–1.2		+1.2	dB

Notes:
- RXPGA = 0 dB
- Output measured single-ended between EAR20 and GND
- Output measured single-ended with 32 Ω load and DC blocking capacitor (220 uF) connected between EAR20 and GND
- +3 dBm0 level corresponds to 0 dB full-scale sine wave.
- 13-bit mode
- Fs = Sampling rate = 8K or 16K Hz

Table 3-18 Receive VOICE Path Level Translation and Linearity, AUX_AMP Selected

Parameter	Test Conditions	Min	Typ	Max	Unit
Receive reference-signal level (0 dBm0)	PCMI = 0 dBm0, 1.02 kHz sine-wave		831		mV_{rms}
Overload-signal level (+3 dBm0)	PCMI = +3 dBm0, 1.02 kHz sine-wave		3.32		V_{pp}
Absolute gain error	PCMI = 0 dBm0, 1.02 kHz sine-wave	–1		+1	dB
Gain error relative to gain at –10 dBm0	PCMI = +3 dBm0 to –40 dBm0	–0.5		+0.5	dB
Gain error relative to gain at –10 dBm0	PCMI = –41 dBm0 to –50 dBm0	–1		+1	dB
Gain error relative to gain at –10 dBm0	PCMI = –51 dBm0 to –55 dBm0	–1.2		+1.2	dB

Notes:
- RXPGA = 0 dB
- Output measured differentially between AUXOP and AUXON.
- +3 dBm0 level corresponds to 0 dB Full-scale sine wave.
- loaded condition (600 ohm)
- 13-bit mode
- Fs = Sampling rate = 8K or 16K Hz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-19 Receive VOICE Path Frequency Response and Image Rejection

Parameter	Test Conditions	Min	Typ	Max	Unit
Gain relative to input signal gain at 1.02 kHz, digital Rx highpass filter disabled.	Frequency <0.0125xFs Hz	–0.5		0.5	dB
	Frequency = 0.025xFs Hz	–0.5		0.5	dB
	Frequency = 0.0375xFs Hz to 0.375xFs Hz	–0.5		0.5	dB
	Frequency = 0.425xFs Hz	–1.5		0	dB
	Frequency = 0.4975xFs Hz			–14	dB
	Frequency = 0.575xFs Hz			–46	dB
	Frequency = 0.9975xFs Hz			–50	dB
Gain relative to input signal gain at 1.02 kHz, digital Rx highpass filter enabled.	Frequency < 0.0125xFs Hz			–15	dB
	Frequency = 0.025xFs Hz			–5	dB

Notes:

- RXPGA = 0 dB
- This specification applies to EAR_AMP1, EAR_AMP2, AUXOP, AUXON outputs
- Unloaded condition.
- Frequencies 0.4975xFs, 0.575xFs Hz and 0.9975xFs Hz are used to determine image rejection performance.
- Fs = Sampling rate = 8K or 16K Hz

Table 3-20 Receive VOICE Path Idle Channel Noise and Distortion, EAR_AMP1 Selected

Parameter	Test Conditions	Min	Typ	Max	Unit
Receive noise	PCMIN = “0000000000000”		150	200	μV_{rms}
Receive signal-to-THD+N ratio with 1020 Hz sine-wave input	PCMI = +3 dBm0	29			dB
	PCMI = 0 dBm0	50			dB
	PCMI = –5 dBm0	47			dB
	PCMI = –10 dBm0	46			dB
	PCMI = –20 dBm0	42			dB
	PCMI = –30 dBm0	40			dB
	PCMI = –40 dBm0	30			dB
	PCMI = –45 dBm0	25			dB
Intermodulation distortion (2 tone method)	PCMI = 498 Hz and 2.02 kHz equal amplitude tones, composite peak level equivalent to 0 dBm0 sine-wave	50			dB

Notes:

- RXPGA = 0 dB
- Output measured differentially between EAR1ON and EAR1OP
- +3 dBm0 level corresponds 0 dB full-scale sine-wave
- Loaded condition (32 ohm)
- 13-bit mode
- A-weighted
- Fs = Sampling rate, 8 kHz or 16 kHz
- Measurement Bandwidth = 100 Hz to 20 kHz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-21 Receive VOICE Path Idle Channel Noise and Distortion, EAR_AMP2 Selected

Parameter	Test Conditions	Min	Typ	Max	Unit
Receive noise	PCMI = “0000000000000”			106	μV_{rms}
Receive signal-to- THD+N ratio with 1020Hz sine-wave input	Output Level at +3 dBm0	26			dB
	Output Level at 0 dBm0	45			dB
	Output Level at –5 dBm0	44			dB
	Output Level at –10 dBm0	42			dB
	Output Level at –20 dBm0	39			dB
	Output Level at –30 dBm0	37			dB
	Output Level at –40 dBm0	27			dB
	Output Level at –45 dBm0	22			dB
Intermodulation distortion (2 tone method)	PCMI = 498 Hz and 2.02 kHz equal amplitude tones, composite peak level equivalent to 0 dBm0 sine wave.	50			dB

Notes:
- RXPGA = 0 dB
- Output measured single-ended between EAR2O and GND
- Output measured single-ended with 32 Ω load and DC blocking capacitor (220 uF) connected between EAR2O and GND
- +3 dBm0 level corresponds 0 dB full-scale sine-wave
- 13-bit mode
- A-weighted
- Fs = Sampling rate, 8 kHz or 16 kHz
- Measurement Bandwidth = 100 Hz to 20 kHz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-22 Receive VOICE Path Idle Channel Noise and Distortion, EAR_AMP3 Selected

Parameter	Test Conditions	Min	Typ	Max	Unit
Receive noise	PCMI = “0000000000000”			106	μV_{rms}
Receive signal-to- THD+N ratio with 1020Hz sine-wave input	Output Level at +3 dBm0	26			dB
	Output Level at 0 dBm0	45			dB
	Output Level at –5 dBm0	44			dB
	Output Level at –10 dBm0	42			dB
	Output Level at –20 dBm0	39			dB
	Output Level at –30 dBm0	37			dB
	Output Level at –40 dBm0	27			dB
	Output Level at –45 dBm0	22			dB
Intermodulation distortion (2 tone method)	PCMI = 498 Hz and 2.02 kHz equal amplitude tones, composite peak level equivalent to 0 dBm0 sine wave.	50			dB

Notes:
- RXPGA = 0 dB
- Output measured single-ended between EAR2O and GND
- Output measured single-ended with 32 Ω load and DC blocking capacitor (220 uF) connected between EAR2O and GND
- +3 dBm0 level corresponds 0 dB full-scale sine-wave
- 13-bit mode
- A-weighted
- Fs = Sampling rate, 8 kHz or 16 kHz
- Measurement Bandwidth = 100 Hz to 20 kHz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-23 Receive VOICE Path Idle Channel Noise And Distortion, AUX_AMP Selected

Parameter	Test Conditions	Min	Typ	Max	Unit
Receive noise	PCMIN = "0000000000000000"		150	200	μV_{rms}
Receive signal-to-THD+N ratio with 1020 Hz sine-wave input	PCMI = +3 dBm0	29			dB
	PCMI = 0 dBm0	50			dB
	PCMI = –5 dBm0	47			dB
	PCMI = –10 dBm0	46			dB
	PCMI = –20 dBm0	42			dB
	PCMI = –30 dBm0	40			dB
	PCMI = –40 dBm0	30			dB
	PCMI = –45 dBm0	25			dB
Intermodulation distortion (2 tone method)	PCMI = 498 Hz and 2.02 kHz equal amplitude tones, composite peak level equivalent to 0 dBm0 sine wave.	50			dB

Notes:

- RXPGA = 0 dB
- Output measured differentially between AUXOP and AUXON
- +3 dBm0 level corresponds 0 dB full-scale sine-wave
- Loaded condition (600 ohm)
- 13-bit mode
- A-weighted
- Fs = Sampling rate, 8 kHz or 16 kHz
- Measurement Bandwidth = 100 Hz to 20 kHz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-24 Receive AUDIO Path Level Translation and Linearity, EAR_AMP2 Selected

Parameter	Test Conditions	Min	Typ	Max	Unit
Receive reference-signal level (0 dBm0)	PCMI = 0 dBm0, 1.02 kHz sine-wave		416.5		mV_{rms}
Overload-signal level (+3 dBm0)	PCMI = +3 dBm0, 1.02 kHz sine-wave		1.664		V_{pp}
Absolute gain error	PCMI = 0 dBm0, 1.02 kHz sine-wave	–1		+1	dB
Gain error relative to gain at –10 dBm0	PCMI = +3 dBm0 to –40 dBm0	–0.5		+0.5	dB
Gain error relative to gain at –10 dBm0	PCMI = –41 dBm0 to –50 dBm0	–1		+1	dB
Gain error relative to gain at –10 dBm0	PCMI = –51 dBm0 to –55 dBm0	–1.2		+1.2	dB

Notes:
- RXPGA = 0 dB
- Output measured single-ended between EAR20 and GND
- Output measured single-ended with 32 Ω load and DC blocking capacitor (220 uF) connected between EAR20 and GND
- +3 dBm0 level corresponds to 0 dB full-scale sine wave.
- 16-bit mode
- Fs = Sampling rate = 16K or 22.05K or 24K or 32K or 44.1K or 48K Hz

Table 3-25 Receive AUDIO Path Level Translation and Linearity, EAR_AMP3 Selected

Parameter	Test Conditions	Min	Typ	Max	Unit
Receive reference-signal level (0 dBm0)	PCMI = 0 dBm0, 1.02 kHz sine-wave		416.5		mV_{rms}
Overload-signal level (+3 dBm0)	PCMI = +3 dBm0, 1.02 kHz sine-wave		1.664		V_{pp}
Absolute gain error	PCMI = 0 dBm0, 1.02 kHz sine-wave	–1		+1	dB
Gain error relative to gain at –10 dBm0	PCMI = +3 dBm0 to –40 dBm0	–0.5		+0.5	dB
Gain error relative to gain at –10 dBm0	PCMI = –41 dBm0 to –50 dBm0	–1		+1	dB
Gain error relative to gain at –10 dBm0	PCMI = –51 dBm0 to –55 dBm0	–1.2		+1.2	dB

Notes:
- RXPGA = 0 dB
- Output measured single-ended between EAR20 and GND
- Output measured single-ended with 32 Ω load and DC blocking capacitor (220 uF) connected between EAR20 and GND
- +3 dBm0 level corresponds to 0 dB full-scale sine wave.
- 16-bit mode
- Fs = Sampling rate = 16K or 22.05K or 24K or 32K or 44.1K or 48K Hz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-26 Receive AUDIO Path Level Translation and Linearity, AUX_AMP (L or R) Selected and Configured Single-Ended

Parameter	Test Conditions	Min	Typ	Max	Unit
Receive reference-signal level (0 dBm0)	PCMI = 0 dBm0, 1.02 kHz sine-wave		416.5		mV_{rms}
Overload-signal level (+3 dBm0)	PCMI = +3 dBm0, 1.02 kHz sine-wave		1.664		V_{pp}
Absolute gain error	PCMI = 0 dBm0, 1.02 kHz sine-wave	–1		+1	dB
Gain error relative to gain at –10 dBm0	PCMI = +3 dBm0 to –40 dBm0	–0.5		+0.5	dB
Gain error relative to gain at –10 dBm0	PCMI = –41 dBm0 to –50 dBm0	–1		+1	dB
Gain error relative to gain at –10 dBm0	PCMI = –51 dBm0 to –55 dBm0	–1.2		+1.2	dB

Notes:
- RXPGA = 0 dB
- Output measured single-ended between AUXOP and GND, AUXON and GND.
- Output measured single-ended with 600Ω load and DC blocking capacitor (15 uF) connected between AUXOP and GND; AUXON and GND
- +3 dBm0 level corresponds to 0 dB full-scale sine wave.
- 16-bit mode
- Fs = Sampling rate = 16K or 22.05K or 24K or 32K or 44.1K or 48K Hz

Table 3-27 Receive AUDIO Path Level Translation and Linearity, AUX_AMP Selected and Configured Differential

Parameter	Test Conditions	Min	Typ	Max	Unit
Receive reference-signal level (0 dBm0)	PCMI = 0 dBm0, 1.02 kHz sine-wave		831		mV_{rms}
Overload-signal level (+3 dBm0)	PCMI = +3 dBm0, 1.02 kHz sine-wave		3.32		V_{pp}
Absolute gain error	PCMI = 0 dBm0, 1.02 kHz sine-wave	–1		+1	dB
Gain error relative to gain at –10 dBm0	PCMI = +3 dBm0 to –40 dBm0	–0.5		+0.5	dB
Gain error relative to gain at –10 dBm0	PCMI = –41 dBm0 to –50 dBm0	–1		+1	dB
Gain error relative to gain at –10 dBm0	PCMI = –51 dBm0 to –55 dBm0	–1.2		+1.2	dB

Notes:
- RXPGA = 0 dB
- Output measured differentially with 600Ω load connected between AUXOP and AUXON
- +3 dBm0 level corresponds to 0 dB full-scale sine wave.
- 16-bit mode
- Fs = Sampling rate = 16K or 22.05K or 24K or 32K or 44.1K or 48K Hz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-28 Receive Path AUDIO Frequency Response and Image Rejection

Parameter	Test Conditions	Min	Typ	Max	Unit
Gain relative to input signal gain at 1.02 kHz, digital Rx highpass filter disabled.	Frequency <0.0125xFs Hz	–0.5		0.5	dB
	Frequency = 0.025xFs Hz	–0.5		0.5	dB
	Frequency = 0.0375xFs Hz to 0.375xFs Hz	–0.5		0.5	dB
	Frequency = 0.425xFs Hz	–1.5		0	dB
	Frequency = 0.4975xFs Hz			–14	dB
	Frequency = 0.575xFs Hz			–35	dB
	Frequency = 0.9975xFs Hz			–47	dB
Gain relative to input signal gain at 1.02 kHz, digital Rx highpass filter enabled.	Frequency < 0.0125xFs Hz			–15	dB
	Frequency = 0.025xFs Hz			–5	dB

Notes:
- RXPGA = 0 dB
- This specification applies to EAR_AMP2, AUX_AMP outputs
- Unloaded condition.
- Frequencies 0.4975xFs, 0.575xFs Hz and 0.9975xFs Hz are used to determine image rejection performance.
- Fs = Sampling rate = 16K or 22.05K or 24K or 32K or 44.1K or 48K Hz

Table 3-29 Receive AUDIO Path Idle Channel Noise and Distortion, EAR_AMP2 Selected

Parameter	Test Conditions	Min	Typ	Max	Unit
Receive noise	PCMI = “0000000000000000”		23.42	26.28	μV_{rms}
Total Harmonic Distortion + Noise (THD+N)	Output Level at +3 dBm0, 1.02 kHz and 5 kHz, Vdd=2.5 V		–66	–60	dB
	Output Level at –57 dBm0, 1.02 kHz and 5 kHz, Vdd=2.5 V		–28		dB

Notes:
- RXPGA = 0 dB
- Output measured single-ended between EAR2O and GND
- Output measured single-ended with 16 Ω load and DC blocking capacitor (450 uF) connected between HPh_L and GND
- +3 dBm0 level corresponds 0 dB full-scale sine-wave
- 16-bit mode
- A-weighted
- Fs = Sampling rate, 16K or 22.05K or 24K or 32K or 44.1K or 48K Hz
- Measurement Bandwidth = 20 Hz to 20 kHz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-30 Receive AUDIO Path Idle Channel Noise and Distortion, EAR_AMP3 Selected

Parameter	Test Conditions	Min	Typ	Max	Unit
Receive noise	PCMI = “0000000000000000”		23.42	26.28	μV_{rms}
Total Harmonic Distortion + Noise (THD+N)	Output Level at +3 dBm0, 1.02 kHz and 5 kHz, Vdd=2.5 V		–66	–60	dB
	Output Level at –57 dBm0, 1.02 kHz and 5 kHz, Vdd=2.5 V		–28		dB

Notes:

- RXPGA = 0 dB
- Output measured single-ended between EAR2O and GND
- Output measured single-ended with 16 Ω load and DC blocking capacitor (450 μF) connected between HPh_R and GND
- +3 dBm0 level corresponds 0 dB full-scale sine-wave
- 16-bit mode
- A-weighted
- Fs = Sampling rate, 16K or 22.05K or 24K or 32K or 44.1K or 48K Hz
- Measurement Bandwidth = 20 Hz to 20 kHz

Table 3-31 Receive AUDIO Path Idle Channel Noise and Distortion, AUX_AMP (R or L) Selected and Configured Single-Ended

Parameter	Test Conditions	Min	Typ	Max	Unit
Receive noise	PCMI = “0000000000000000”		26		μV_{rms}
Total Harmonic Distortion + Noise (THD+N)	Output Level at +3 dBm0, 1.02 kHz and 5 kHz, Vdd=2.5 V		–66	–60	dB
	Output Level at –57 dBm0, 1.02 kHz and 5 kHz, Vdd=2.5 V		–25		dB

Notes:

- RXPGA = 0 dB
- Output measured single-end with 600 Ω load and DC blocking capacitor (15 μF) connected between AUXO (L or R) and GND.
- +3 dBm0 level corresponds to 0 dB full-scale sine wave.
- Loaded condition (600 ohm)
- 16-bit mode
- A-weighted
- Fs = Sampling rate, 16K or 22.05K or 24K or 32K or 44.1K or 48K Hz
- Measurement Bandwidth = 20 Hz to 20 kHz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-32 Receive AUDIO Path Idle Channel Noise and Distortion, AUX_AMP Selected and Configured Differential

Parameter	Test Conditions	Min	Typ	Max	Unit
Receive noise	PCMI = "0000000000000000"		46.73	52.0	μV_{rms}
Total Harmonic Distortion + Noise (THD+N)	Output Level at +3 dBm0, 1.02 kHz and 5 kHz, Vdd=2.5 V		–66	–60	dB
	Output Level at –57 dBm0, 1.02 kHz and 5 kHz, Vdd=2.5 V		–28		dB

Notes:
- RXPGA = 0 dB
- Output measured differentially between AUXOP and AUXON.
- +3 dBm0 level corresponds to 0 dB full-scale sine wave.
- Loaded condition (600 ohm)
- 16-bit mode
- A-weighted
- Fs = Sampling rate, 16K or 22.05K or 24K or 32K or 44.1K or 48K Hz
- Measurement Bandwidth = 20 Hz to 20 kHz

Table 3-33 AUXIP (Line_In_L) Input to EAR_AMP1 Output Selected

Parameter	Test Conditions	Min	Typ	Max	Unit
Overload-signal level *	AUXIP (Line_In_L) = 1.5 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB		4.24		V_{pp}
Output Referred Noise * (A-weighted)	AUXIP (Line_In_L) = 0 V_{pp}, AUX PGA = +5.5dB			106	μV_{rms}
Total Harmonic Distortion + Noise (THD+N) * (A-weighted)	AUXIP (Line_In_L) = 1.5 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB, Vdd=2.5 V			-28	dB
Absolute gain error	AUXIP (Line_In_L) = 1.06 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB	–0.5		+0.5	dB

Notes:
1. * Levels should scale for AUX PGA gains = +2.5dB, -0.5dB, -3.5dB, -6.5dB respectively.
2. Output measured differentially between EARN and EARP, loaded condition (32 Ω)
3. Left and Right DAC channels are in mute condition
4. Measurement Bandwidth = 20 Hz to 20 kHz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-34 AUXIP (Line_In_L) Input to EAR_AMP2 Output Selected

Parameter	Test Conditions	Min	Typ	Max	Unit
Overload-signal level *	AUXIP (Line_In_L) = 1.5 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB		1.664		V_{pp}
Output Referred Noise * (A-weighted)	AUXIP (Line_In_L) = 0 $V_{pp,}$ AUX PGA = +5.5dB			42	μV_{rms}
Total Harmonic Distortion + Noise (THD+N) * (A-weighted)	AUXIP (Line_In_L) = 1.5 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB, Vdd=2.5 V			-66	dB
Absolute gain error	AUXIP (Line_In_L) = 1.06 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB	–0.5		+0.5	dB

Notes:

1. * Levels should scale for AUX PGA gains = +2.5dB, -0.5dB, -3.5dB, -6.5dB respectively.
2. Output measured single-ended with 16 Ω load and DC blocking capacitor (450 uF) connected between HPH_L and GND
3. Left and Right DAC channels are in mute condition
4. Measurement Bandwidth = 20 Hz to 20 kHz

Table 3-35 AUXIP (Line_In_L) Input to AUX_AMP Output Selected and Configured Differential

Parameter	Test Conditions	Min	Typ	Max	Unit
Overload-signal level *	AUXIP (Line_In_L) = 1.5 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB		3.32		V_{pp}
Output Referred Noise * (A-weighted)	AUXIP (Line_In_L) = 0 $V_{pp,}$ AUX PGA = +5.5dB			84	μV_{rms}
Total Harmonic Distortion + Noise (THD+N) * (A-weighted)	AUXIP (Line_In_L) = 1.5 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB, Vdd=2.5 V			-66	dB
Absolute gain error	AUXIP (Line_In_L) = 1.06 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB	–0.5		+0.5	dB

Notes:

1. * Levels should scale for AUX PGA gains = +2.5dB, -0.5dB, -3.5dB, -6.5dB respectively.
2. Output measured differentially between AUXOP and AUXON
3. Left and Right DAC channels are in mute condition
4. Measurement Bandwidth = 20 Hz to 20 kHz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-36 AUXIP (Line_In_L) Input to AUX_AMP Output Selected and Configured Single-Ended

Parameter	Test Conditions	Min	Typ	Max	Unit
Overload-signal level *	AUXIP (Line_In_L) = 1.5 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB		1.664		V_{pp}
Output Referred Noise * (A-weighted)	AUXIP (Line_In_L) = 0 $V_{pp,}$ AUX PGA = +5.5dB			42	μV_{rms}
Total Harmonic Distortion + Noise (THD+N) * (A-weighted)	AUXIP (Line_In_L) = 1.5 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB, Vdd=2.5 V			-66	dB
Absolute gain error	AUXIP (Line_In_L) = 1.06 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB	–0.5		+0.5	dB

Notes:

1. * Levels should scale for AUX PGA gains = +2.5dB, -0.5dB, -3.5dB, -6.5dB respectively.
2. Output measured single-ended with 600Ω load and DC blocking capacitor (15 uF) connected between AUXOP and GND
3. Left and Right DAC channels are in mute condition
4. Measurement Bandwidth = 20 Hz to 20 kHz

Table 3-37 AUXIN (Line_In_R) Input to EAR_AMP3 Output Selected

Parameter	Test Conditions	Min	Typ	Max	Unit
Overload-signal level *	AUXIN (Line_In_R) = 1.5 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB		1.664		V_{pp}
Output Referred Noise * (A-weighted)	AUXIN (Line_In_R) = 0 $V_{pp,}$ AUX PGA = +5.5dB			42	μV_{rms}
Total Harmonic Distortion + Noise (THD+N) * (A-weighted)	AUXIN (Line_In_R) = 1.5 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB, Vdd=2.5 V			-66	dB
Absolute gain error	AUXIN (Line_In_R) = 1.06 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB	–0.5		+0.5	dB

Notes:

1. * Levels should scale for AUX PGA gains = +2.5dB, -0.5dB, -3.5dB, -6.5dB respectively.
2. Output measured single-ended with 16 Ω load and DC blocking capacitor (450 uF) connected between HPH_R and GND
3. Left and Right DAC channels are in mute condition
4. Measurement Bandwidth = 20 Hz to 20 kHz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-38 AUXIN (Line_In_R) Input to AUX_AMP Output Selected and Configured Differential

Parameter	Test Conditions	Min	Typ	Max	Unit
Overload-signal level *	AUXIN (Line_In_R) = 1.5 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB		3.32		V_{pp}
Output Referred Noise * (A-weighted)	AUXIN (Line_In_R) = 0 V_{pp}, AUX PGA = +5.5dB			84	μV_{rms}
Total Harmonic Distortion + Noise (THD+N) * (A-weighted)	AUXIN (Line_In_R) = 1.5 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB, Vdd=2.5 V			-66	dB
Absolute gain error	AUXIN (Line_In_R) = 1.06 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB	–0.5		+0.5	dB

Notes:
1. * Levels should scale for AUX PGA gains = +2.5dB, -0.5dB, -3.5dB, -6.5dB respectively.
2. Output measured differentially between AUXOP and AUXON
3. Left and Right DAC channels are in mute condition
4. Measurement Bandwidth = 20 Hz to 20 kHz

Table 3-39 AUXIN (Line_In_R) Input to AUX_AMP Output Selected and Configured Single-Ended

Parameter	Test Conditions	Min	Typ	Max	Unit
Overload-signal level *	AUXIN (Line_In_R) = 1.5 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB		1.664		V_{pp}
Output Referred Noise * (A-weighted)	AUXIN (Line_In_R) = 0 V_{pp}, AUX PGA = +5.5dB			42	μV_{rms}
Total Harmonic Distortion + Noise (THD+N) * (A-weighted)	AUXIN (Line_In_R) = 1.5 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB, Vdd=2.5 V			-66	dB
Absolute gain error	AUXIN (Line_In_R) = 1.06 V_{pp}, 1.02 kHz sine-wave, AUX PGA = +5.5dB	–0.5		+0.5	dB

Notes:
1. * Levels should scale for AUX PGA gains = +2.5dB, -0.5dB, -3.5dB, -6.5dB respectively.
2. Output measured single-ended with 600Ω load and DC blocking capacitor (15 uF) connected between AUXON and GND
3. Left and Right DAC channels are in mute condition
4. Measurement Bandwidth = 20 Hz to 20 kHz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-40 Power Supply Rejection and Crosstalk Attenuation

Parameter	Test Conditions	Min	Typ	Max	Unit
Supply voltage rejection, transmit channel	Analog Input level = 0 Vpp, AV_{DD} = 2.5V_{dc} + 100mV_{rms},Frequency = 0 - 30 kHz	45			dB
Supply voltage rejection, receive channel	PCMIN = “0000000000000”. AV_{DD} = 2.5 V_{dc} + 100 mV_{rms} , Frequency = 0 - 30 kHz	45			dB
Crosstalk attenuation, transmit-to-receive (differential outputs) with sidetone disabled	0 dBm0 Analog Input Level. Frequency = 0.0375xFs - 0.425xFs Hz. Measured differentially at Rx Path output	70			dB
Crosstalk attenuation, receive-to-transmit	PCMI = 0 dBm0, Frequency = 0.0375xFs - 0.425xFs Hz. Measured at PCMO, EAR/HPh_L/HPh_R/AUXOP/AUXON amp unloaded	70			dB
Interchannel isolation, Left to Right channels	Left channel input = 0 dBm0, Right channel = all zeros input. Frequency = 20 - 20 kHz. Measured HPh_R output	70			dB
Interchannel isolation, Right to Left channels	Right channel input = 0 dBm0, Left channel = all zeros input. Frequency = 20 - 20 kHz. Measured HPh_L output	70			dB

Notes:

1. RXPGA = 0 dB.
2. Applies to all three Tx path inputs and all four Rx Path outputs.
3. Fs = Sampling rate = 8 K or 16 kHz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.1.6 Power Consumption

These values are an estimation of operating currents for the nominal VDD_P and VDD_C voltages of 1.867 V and 2.6 V. Separately shown are the operating currents for VDD_A. This information should be used as a general guideline for system design. CDMA modes assume that the subscriber unit is operating in compliance with the CDMA specifications of IS-95-B. FM modes assume that the subscriber unit is operating in compliance with the AMPS specifications of IS-95-A.

Table 3-41 MSM6100 Device Power Supply Current

Symbol	Operating Mode	Average		Units	Notes
		VDD_P = TBD V VDD_C = TBD V	VDD_P = TBD V VDD_C = TBD V		
IDD1	CDMA RxTx	TBD	TBD	mA	a, b, c, d
IDD2	CDMA Rx	TBD	TBD	mA	a, c, d
IDD3	CDMA SLEEP	TBD	TBD	µA	a, e
IDD4	FM RxTx	TBD	TBD	mA	a
IDD5	FM IDLE	TBD	TBD	mA	a

[a] Values shown in the Average columns represent a time average of the observed data from recent testing conducted by QUALCOMM over a limited number of devices at Room Temperature (25 °C). See Figure 3-1 for illustrations of "Average" power supply current.

[b] The vocoder is in EVRC mode and the UART clock regime is turned off.

[c] The current consumption for the integrated Analog portion of the device is not included in the table.

[d] The TCXO frequency is 19.2 MHz.

[e] TCXO is disabled.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

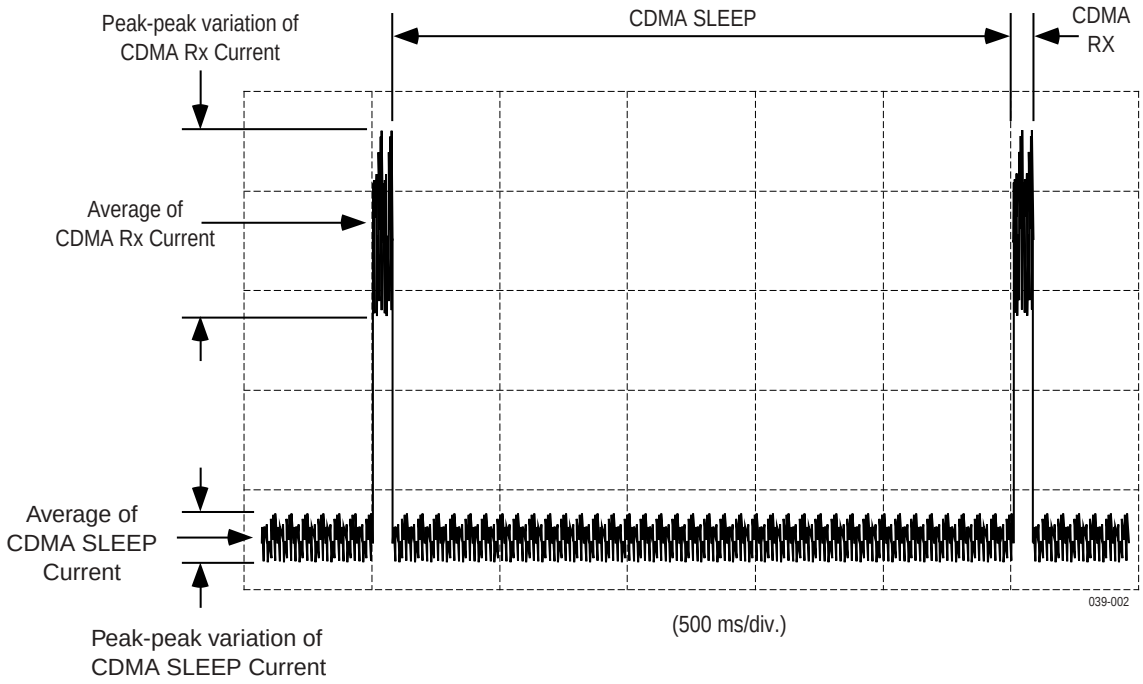


Figure 3-1 Power Supply Current versus Time in Slotted Paging Mode

Table 3-42 VDD_A Power Consumption

Symbol	Operating Mode	Average VDD_A = 2.60 V	Units	Notes
IDDA1	CDMA RxTx	TBD	mA	a,
IDDA2	CDMA Rx	TBD	mA	a,
IDDA3	CDMA SLEEP	TBD	µA	a,
IDDA4	FM RxTx	TBD	mA	a,
IDDA5	FM IDLE	TBD	mA	a,

[a] Values shown in the Average column represent a time average of the observed data from recent testing conducted by QUALCOMM over a limited number of devices at Room Temperature (25 °C). See Figure 3-1 for illustrations of "Average" power supply current.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-43 Continuous Power Consumption per Voltage Rail

Rail	Typical Continuous Maximum		Units	Notes
	TBD V	TBD V		
VDD_C	TBD	TBD	mA	a
VDD_P	TBD	TBD	mA	a
VDD_A	TBD	TBD	mA	a, b

a Values shown in the **Typical Continuous Maximum** columns represent a time average of the observed data from recent testing conducted by QUALCOMM over a limited number of devices at Room Temperature (25 °C).

b Add TBD mA for driving earpiece speaker to maximum level.

NOTE This table is intended to assist with power supply requirements.

3.2 Timing Characteristics

Two distinct memory configurations are targeted; 1.) NAND + SDRAM and 2.) Burst Flash + PSRAM.

NOTE Burst Flash + SDRAM *cannot* be supported.

3.2.1 Measurement Conditions

The tester has an actively terminated load. This makes rise and fall transitions very quick (essentially mimicking a no-load condition). ***For this reason, the bus rise or fall time must be added to parameters that start timing at the MSM6100 and terminate at the memory device.*** One of these parameters is the chip select to data valid on a read cycle. An example of this is t(csacc) in EBI2 asynchronous memory timing.

3.2.2 Bus Rise and Fall Times

This section will be added in a future release of this document.

3.2.3 Clock Jitter

The MSM6100 device has some MCLK jitter, due to PLL jitter and the divider circuitry. This jitter affects all timing parameters, and must be added to, or subtracted from, every measured timing value in this chapter. (in other words, jitter is not included in any of the values listed).

An MCLK jitter specification will be included in a future release of this document.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.4 TCXO Timing

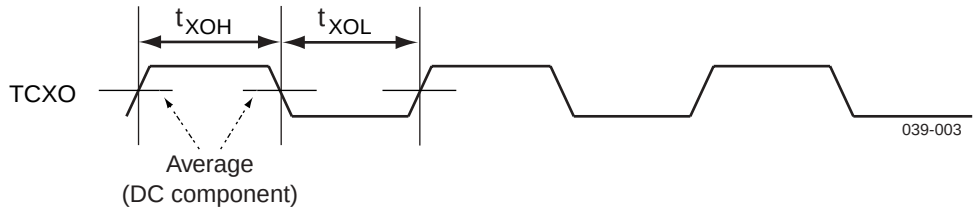


Figure 3-2 TCXO Timing Parameters

Table 3-44 TCXO Timing Parameters

Symbol	Parameter	Min	Typ	Max	Units
t_{XOH}	TCXO logic high	TBD	—	—	ns
t_{XOL}	TCXO logic low	TBD	—	—	ns
T	Clock period	—	52	—	ns
1/T	Frequency (19.2 MHz must be used.)	—	19.2	—	MHz

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.5 MCLK Timing

Table 3-45 MCLK Timing Parameters

Symbol	Parameter	Min	Max	Units
MCLK	ARM Microprocessor clock	—	[a]	ns

[a] The maximum MCLK frequency supported is the maximum frequency defined by the latest release of MSM6100 software. MCLK is a fractional multiple of the TCXO frequency created by the TCXO PLL. The control of this circuitry is determined by feature #defines within the DMSS6100 code. Do not alter the PLL configuration within these #defines. QCT tests the MSM6100 and the latest commercial release of system software to verify operation.

Note: MCLK clock period is equal to T. The MCLK frequency may be the same as the TCXO frequency. MCLK is derived from TCXO within the MSM6100 device.

3.2.6 PCM Interface

Table 3-46 provides timing values for Figure 3-3, Figure 3-4, and Figure 3-5.

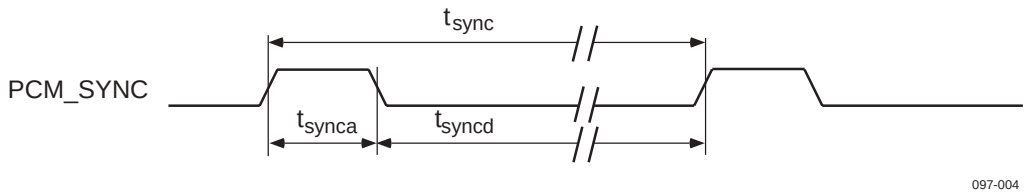


Figure 3-3 PCM_SYNC Timing

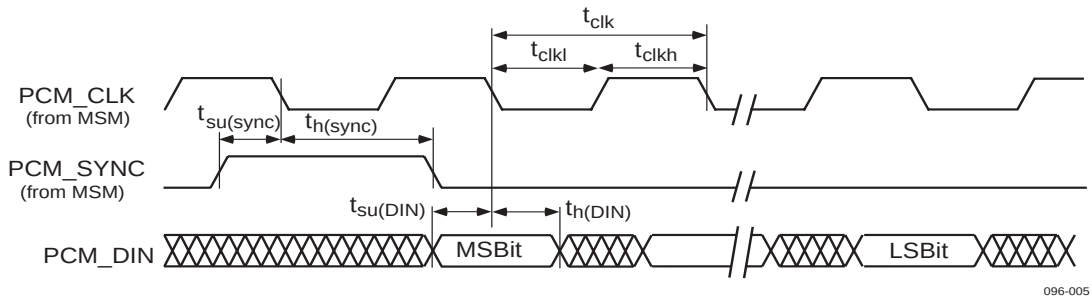


Figure 3-4 PCM_CODEC to MSM6100 Timing

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

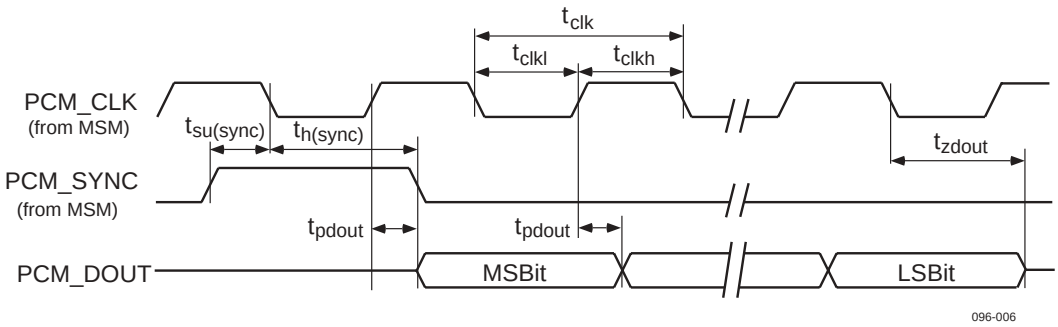


Figure 3-5 MSM6100 to PCM_CODEC Timing

Table 3-46 PCM_CODEC Timing Parameters

Parameter	Description	Min	Typical	Max	Units	Notes
t_{sync}	PCM_SYNC cycle time (PCM_SYNC_DIR=1)	—	125	—	µs	a
	PCM_SYNC cycle time (PCM_SYNC_DIR=0)	—	125	—	µs	
t_{synca}	PCM_SYNC “asserted” time (PCM_SYNC_DIR=1)	400	500	—	ns	1
	PCM_SYNC “asserted” time (PCM_SYNC_DIR=0)	—	—	—	ns	
t_{syncd}	PCM_SYNC “deasserted” time (PCM_SYNC_DIR=1)	—	124.5	—	µs	1
	PCM_SYNC “deasserted” time (PCM_SYNC_DIR=0)	—	—	—	µs	
t_{clk}	PCM_CLK cycle time (PCM_CLK_DIR=1)	400	500	—	ns	1
	PCM_CLK cycle time (PCM_CLK_DIR=0)	—	—	—	ns	
t_{clkh}	PCM_CLK high time (PCM_CLK_DIR=1)	200	250	—	ns	1, b
	PCM_CLK high time (PCM_CLK_DIR=0)	—	—	—	ns	
t_{clkl}	PCM_CLK low time (PCM_CLK_DIR=1)	200	250	—	ns	1, 2
	PCM_CLK low time (PCM_CLK_DIR=0)	—	—	—	ns	
$t_{su(sync)}$	PCM_SYNC setup time to PCM_CLK falling (PCM_SYNC_DIR = 1, PCM_CLK_DIR = 1)	—	150	—	ns	
	PCM_SYNC setup time to PCM_CLK falling (PCM_SYNC_DIR = 0, PCM_CLK_DIR = 0)	—	—	—	ns	
$t_{h(sync)}$	PCM_SYNC hold time after PCM_CLK falling (PCM_SYNC_DIR = 1, PCM_CLK_DIR = 1)	—	350	—	ns	
	PCM_SYNC hold time after PCM_CLK falling (PCM_SYNC_DIR = 0, PCM_CLK_DIR = 0)	—	—	—	ns	
$t_{su(din)}$	PCM_DIN setup time to PCM_CLK falling	50	—	—	ns	
$t_{h(din)}$	PCM_DIN hold time after PCM_CLK falling	10	—	—	ns	
t_{pdout}	Delay from PCM_CLK rising to PCM_DOUT valid	—	—	50	ns	
t_{zdout}	Delay from PCM_CLK falling to PCM_DOUT HIGH-Z	—	16	—	ns	

[a] This value assumes that CODEC_CTL is not being used to override the CDMA CODEC clock and sync operation.

[b] t_{clkh} and t_{clkl} are independent of PCM_CLK_SENSE.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.7 Auxiliary PCM Interface

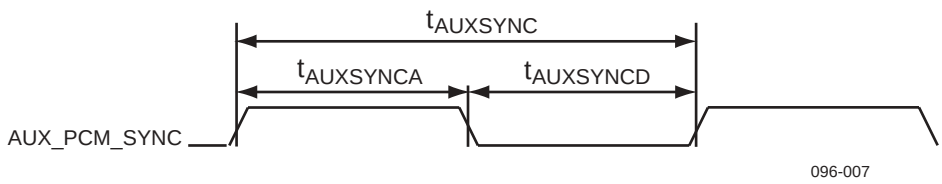


Figure 3-6 AUX_PCM_SYNC Timing

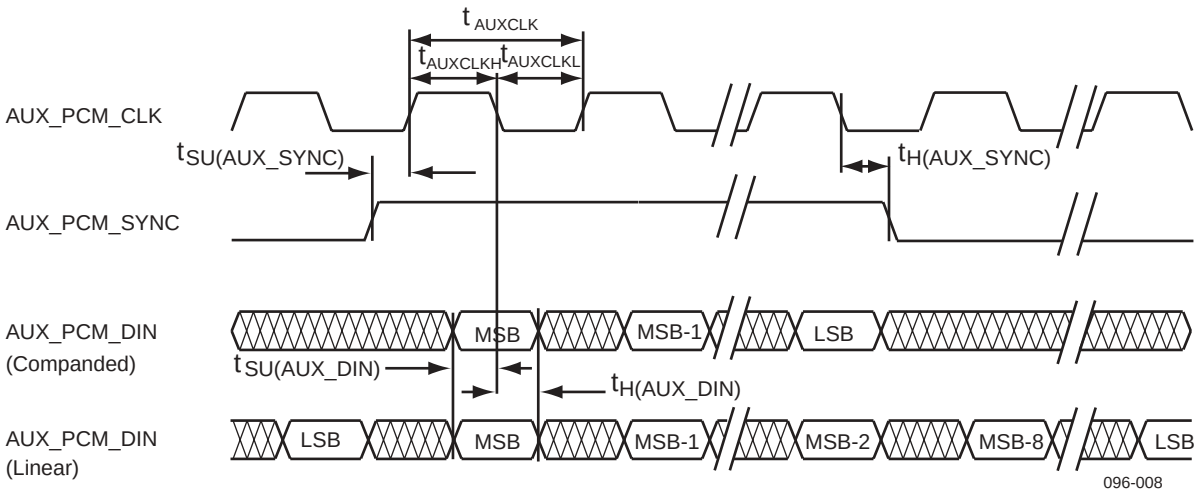


Figure 3-7 CODEC to MSM6100 Timing via AUX_CODEC (MSM6100 Receiving)

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

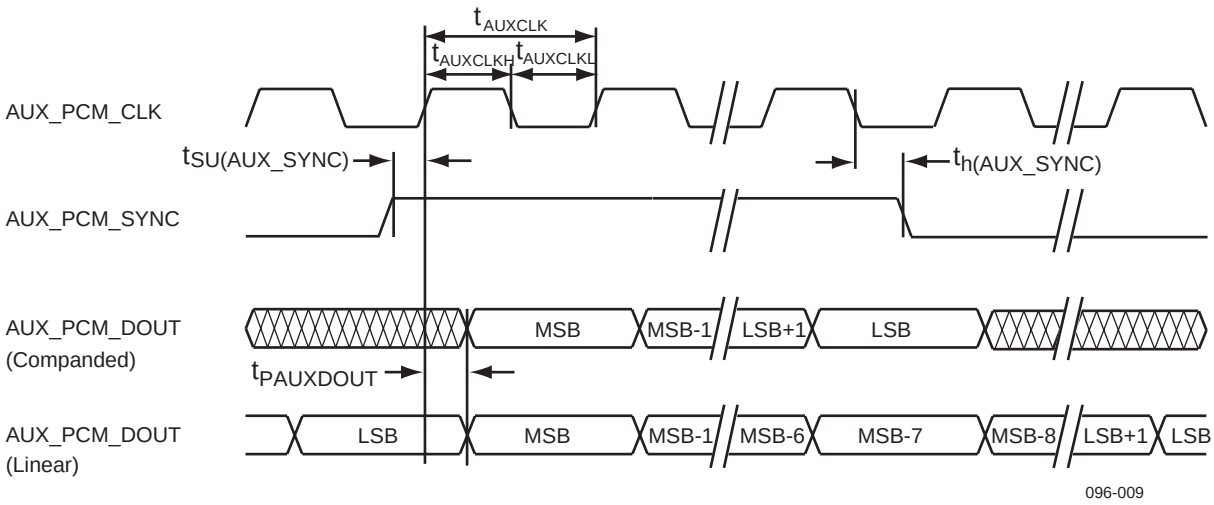


Figure 3-8 MSM6100 to CODEC Timing via AUX_CODEC (MSM6100 Transmitting)

Table 3-47 AUX_CODEC Timing Parameters

Parameter	Description	Min	Typical	Max	Units	Notes
$t_{AUXSYNC}$	AUX_PCM_SYNC cycle time	—	125	—	µs	a
$t_{AUXSYNCA}$	AUX_PCM_SYNC “asserted” time	62.4	62.5	—	µs	a
$t_{AUXSYNCD}$	AUX_PCM_SYNC “deasserted” time	62.4	62.5	—	µs	a
t_{AUXCLK}	AUX_PCM_CLK cycle time	—	7.8	—	µs	a
$t_{AUXCLKH}$	AUX_PCM_CLK high time	3.8	3.9	—	µs	a
$t_{AUXCLKL}$	AUX_PCM_CLK low time	3.8	3.9	—	µs	a
$t_{SU(AUX\_SYNC)}$	AUX_PCM_SYNC setup time to AUX_PCM_CLK rising	1.95	—	—	µs	
$t_{H(AUX\_SYNC)}$	AUX_PCM_SYNC hold time after AUX_PCM_CLK rising	1.95	—	—	µs	
$t_{SU(AUX\_DIN)}$	AUX_PCM_DIN setup time to AUX_PCM_CLK falling	70	—	—	ns	
$t_{H(AUX\_DIN)}$	AUX_PCM_DIN hold time after AUX_PCM_CLK falling	20	—	—	ns	
$t_{PAUXDOUT}$	Propagation delay from AUX_PCM_CLK rising to AUX_PCM_DOUT valid	—	—	50	ns	

[a] This value assumes that CODEC_CTL is not being used to override the CDMA CODEC clock and sync operation.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.8 External Bus Interface 1 (EBI1)

HCLK is shown for reference only and is not available for direct observation. Additionally, cs1_n is used to indicate any of the following pins:

Table 3-48 EBI1 Chip Select Registers Association With Pins

Chip Select/Pin Name	AHB Register
rom1_cs_n0	EBI1_CS0_CFG
rom1_cs_n1(gpio34)	EBI1_CS1_CFG
ram1_cs_n0	EBI1_CS2_CFG
ram1_cs_n1(gpio77)	EBI1_CS3_CFG

Table 3-49 EBI1 Wait States

Wait	Description
$m_{w(rd)}$	DELTA READ - 1
$n_{w(rd)}$	WAIT READ
$m_{w(wr)}$	DELTA WRITE - 1
$n_{w(wr)}$	WAIT WRITE
h_w	HOLD WAIT - 1

Table 3-50 EBI1 Minimum Access Configuration Register Settings (Page Mode Off)

Chip Select	Register	Page Size (30:28)	PRECH_CYC (27:24)	RECOVERY (23:20)	HOLD_WAIT (19:16)	DELTA_WR (15:12)	DELTA_RD (10:8)	WAIT_WR (7:4)	WAIT_RD (3:0)
rom1_cs_n0	EBI1_CS0_CFG0	0	0	0	1	1	1	0	0
rom1_cs_n1	EBI1_CS1_CFG0	0	0	0	1	1	1	0	0
ram1_cs_n0	EBI1_CS2_CFG0	0	0	0	1	1	1	0	0
ram1_cs_n1	EBI1_CS3_CFG0	0	0	0	1	1	1	0	0

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.8.1 EBI1 External Memory Interface Timing

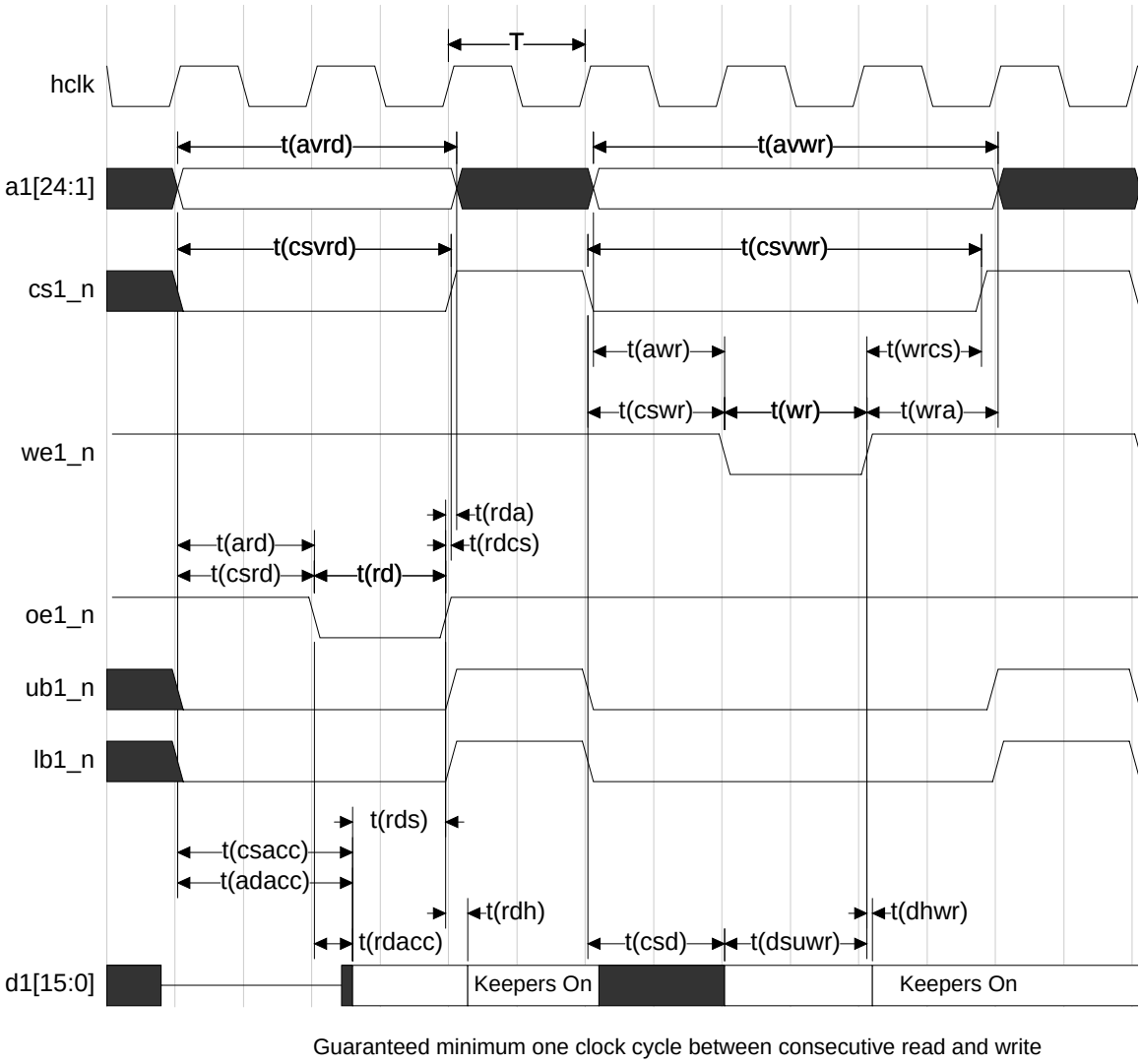
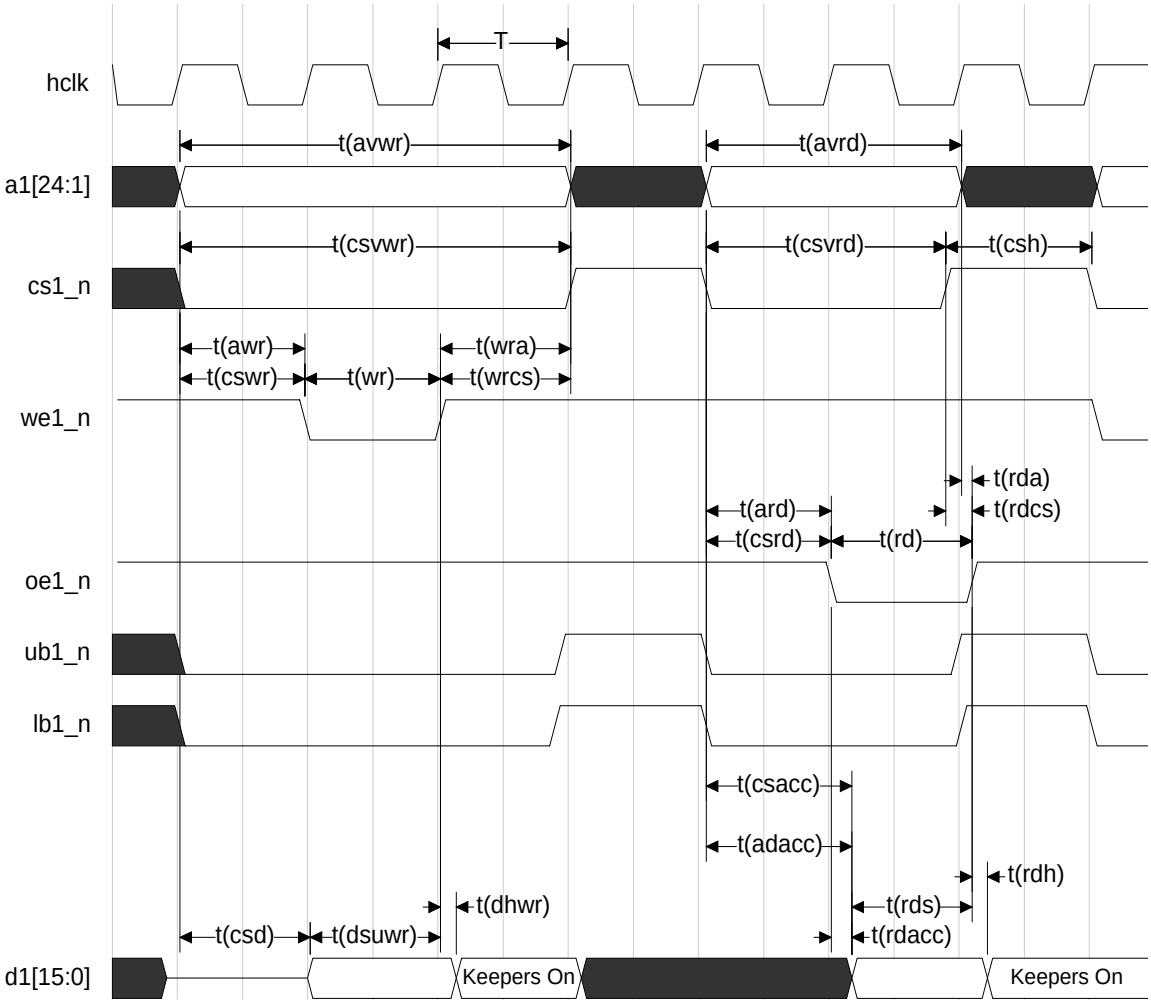


Figure 3-9 16-bit Read Followed by 16-Bit Write Access (Async)

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.



Guaranteed minimum one clock cycle between consecutive write and read access to EB

Figure 3-10 16-bit Write Followed by 16-Bit Read Access (Async)

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-51 EBI1 16-bit Asynchronous Write Timing

Parameter		Equation	Notes
t_{avwr}	Address valid to address invalid	$3T + (m_{w(wr)} + n_{w(wr)} + h_{w(wr)})T - \xi_2$	take worst case among all pins.
t_{csvwr}	Chip select active	$3T + (m_{w(wr)} + n_{w(wr)} + h_{w(wr)})T - \varepsilon_1$	for a non-bus-sized, non-sequential access
t_{dsuwr}	Write data setup	$T + (m_{w(wr)} + n_{w(wr)})T - \upsilon_1$	
t_{dhwr}	Write data hold	σ_1	by design, keepers will hold data, spec T in user manual
t_{awr}	Address valid to write active	$T - \omega_1$	take worst case of a normal write or a bus-sized write
t_{cswr}	Chip select active to write active	$T - \omega_2$	
t_{wr}	Write active	$T + (m_{w(wr)} + n_{w(wr)})T - \zeta_1$	
t_{csd}	Chip select active to data valid	$T - \iota$	will determine allowable trdh
t_{wra}	Write inactive to address invalid	$T + (h_w)T - \mu_1$	
t_{wrcs}	Write inactive to chip select inactive	$T + (h_w)T - \mu_2$	
t_{acs}	Address Valid to chip select active	$T + (r_w)T - \theta_2$	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-52 EBI1 16-bit Asynchronous Read Timing

Parameter		Equation	Notes
t_{avrd}	Address valid to address invalid	$2T + (m_{w(rd)} + n_{w(rd)})T - \xi_1$	
t_{csvrd}	Chip select active	$2T + (m_{w(rd)} + n_{w(rd)})T - \varepsilon_2$	for a non-bus-sized, non-sequential access
t_{rds}	Read data setup	χ_1	
t_{rdh}	Read data hold	α_1	Max value also limited by t_{csd}.
t_{ard}	Address valid to read active	$T - \lambda_1$	take worst case of a normal read or a bus-sized read
t_{csrd}	Chip select active to read active	$T - \lambda_2$	
t_{rd}	Read active	$T + (m_{w(rd)} + n_{w(rd)})T - \gamma$	
t_{rda}	Read inactive to address invalid	η_1	
t_{rdcs}	Read inactive to chip select inactive	η_2	
t_{acs}	Address valid to chip select active	$(r_w)T - \theta_1$	

Table 3-53 EBI1 Timing Parameters Where V_{DD_P1} = 1.8V

Symbol	Parameter	Min	Mean	Max	Std Dev
$\alpha 1$	TRDH	-6.580	-4.677	-3.300	0.673
$\alpha 2$	TRDH1	-4.290	-3.002	-2.120	0.446
$\alpha 3$	TRDH2	-6.040	-4.354	-3.100	0.603
$\chi 1$	TRDS	3.720	5.358	7.610	0.765
$\chi 2$	TRDS1	3.810	5.328	7.340	0.735
$\chi 3$	TRDS2	0.170	0.384	0.720	0.100
$\xi 1$	TAVRD	2.600	3.662	5.080	0.522
$\xi 2$	TAVWR	0.120	0.303	0.670	0.083
$\varepsilon 1$	TCSVWR	-0.290	-0.131	-0.030	0.042
$\varepsilon 2$	TCSVRD	-0.470	0.345	1.330	0.464
γ	TRD	-0.700	-0.388	-0.240	0.086

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-53 EBI1 Timing Parameters Where V_{DD_P1} = 1.8V (Continued)

Symbol	Parameter	Min	Mean	Max	Std Dev
η1	TRDA	-2.280	-0.877	-0.010	0.472
η2	TRDCS	0.300	0.510	0.840	0.099
ι	TCSD	0.220	0.389	0.710	0.069
κ1	TBSWRH	-0.460	0.349	1.370	0.437
λ1	TARD	0.200	0.375	0.680	0.069
λ2	TCSRD	0.170	0.407	0.700	0.092
λ3	TARD	-0.580	0.243	1.370	0.461
μ1	TWRA	-0.250	-0.018	0.140	0.071
μ2	TWRCS	-0.960	-0.039	0.760	0.432
π1	TABSWR	-0.280	-0.033	0.120	0.076
θ1	TACS	-0.900	-0.576	-0.300	0.127
θ2	TACS	0.430	0.551	0.850	0.067
σ1	TDHWR	-0.770	-0.496	-0.050	0.112
σ2	TDHBSWR2	0.320	0.642	0.890	0.098
σ3	TDHBSWR	0.300	0.599	0.770	0.083
υ1	TDSUWR	-0.710	-0.448	-0.260	0.073
υ2	TDSUBSWR1	-0.770	0.040	0.830	0.426
υ3	TDSUBSWR2	-0.740	0.021	0.820	0.420
ω1	TAWR	0.070	0.331	0.740	0.110
ω2	TCSWR	0.150	0.363	0.760	0.098
ω3	TAWR	0.710	0.958	1.400	0.132
ψ1	TABSRD1	-0.060	0.143	0.320	0.056
ψ2	TABSRD2	-0.400	-0.118	0.090	0.071
ζ1	TWR	-0.900	-0.507	-0.290	0.100
ζ2	TBSWR1	-0.910	-0.494	-0.280	0.107
ζ3	TBSWR2	-33.860	-33.513	-33.330	0.087

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.8.2 EBI1 Bus Sized Access Timing

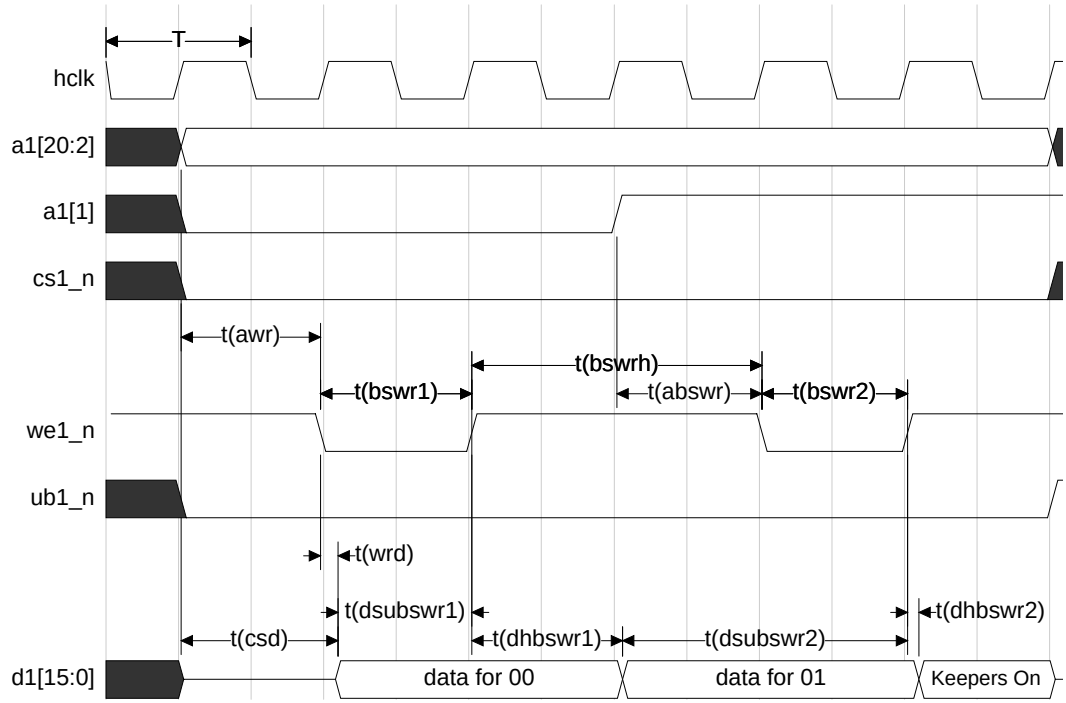


Figure 3-11 Word Bus Sized Write Access to 16-Bit Memory (Async)

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

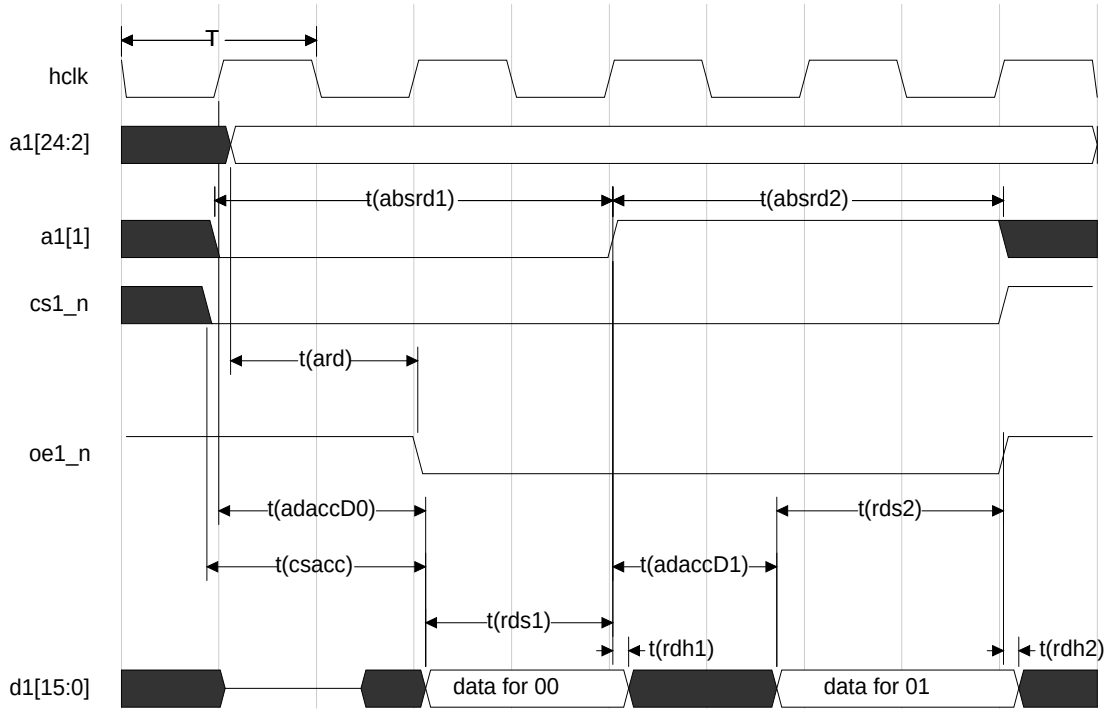


Figure 3-12 Word (32-Bit) Read Access From 16 Bit Memory (Async)

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-54 EBI1 32-bit Bus Sized Write Timing

Parameter		Equation	Notes
t_{bswr1}	Pulse width of first write active	$T + (m_{w(wr)} + n_{w(wr)})T - \zeta_2$	
t_{bswr2}	Pulse width of second write active	$T + n_{w(wr)}T - \zeta_3$	
t_{bswrh}	Pulse width of write inactive between bus-sized writes	$2T + h_wT - \kappa_1$	
t_{awr}	Address valid to first write active	$T - \omega_3$	take worst case among all address pins and a normal write case
t_{abswr}	Address[1] valid to the second write active	$T - \pi_1$	
$t_{dsubswr1}$	Bus-sized write data setup for first write	$T + (m_{w(wr)} + n_{w(wr)})T - \upsilon_2$	
$t_{dsubswr2}$ or $t_{dsubswr4}$	Bus-sized write data setup for the last write	$2T + n_wT - \upsilon_3$	take worst case among the 2nd, or 3rd, or 4th writes.
$t_{dhbswr1}$	Bus-sized write data hold (first write)	$T + h_{w(wr)}T - \sigma_3$	worst case among all write cycles except the last.
$t_{dhbswr2}$ or $t_{dhbswr4}$	Bus-sized write data hold (last write)	σ_2	Last cycle. Time when drivers turn-off and keepers turn-on.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-55 EBI1 32-bit Bus Sized Read Timing

Parameter		Equation	Notes
t_{ard}	Address valid to first read active	$T - \lambda_3$	take worst case among all address pins and a normal read case
t_{absrd1}	Address valid to end of first read	$2T + (m_{w(rd)} + n_{w(rd)})T - \psi_1$	
t_{absrd2}	Address valid to end of the second read	$2T + n_{w(rd)}T - \psi_2$	
t_{rds1}	Bus-sized read data setup for the first read	χ_2	take worst case among all reads and normal read
t_{rdh1}	Bus-sized read data hold for the first read	α_2	take worst case among all reads and normal read. Max value also limited by $t_{csd.}$
t_{rds2}	Bus-sized read data setup for the second read	χ_3	take worst case among all reads and normal read
t_{rdh2}	Bus-sized read data hold for the second read	α_3	take worst case among all reads and normal read. Max value also limited by $t_{csd.}$

3.2.8.3 EBI1 SDRAM Timing

MSM Configuration:

- VREG = 1.4V
- V_{DD_P1} = 1.8V
- EBI1_CLK_POL = 1 (Bit # 9 in EBI1_CFG)

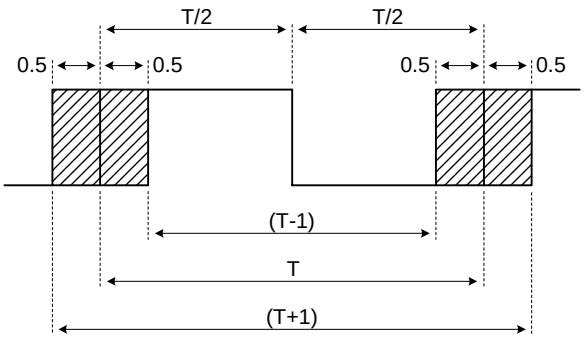


Figure 3-13 SDRAM Clock for T = 13.62 ns

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

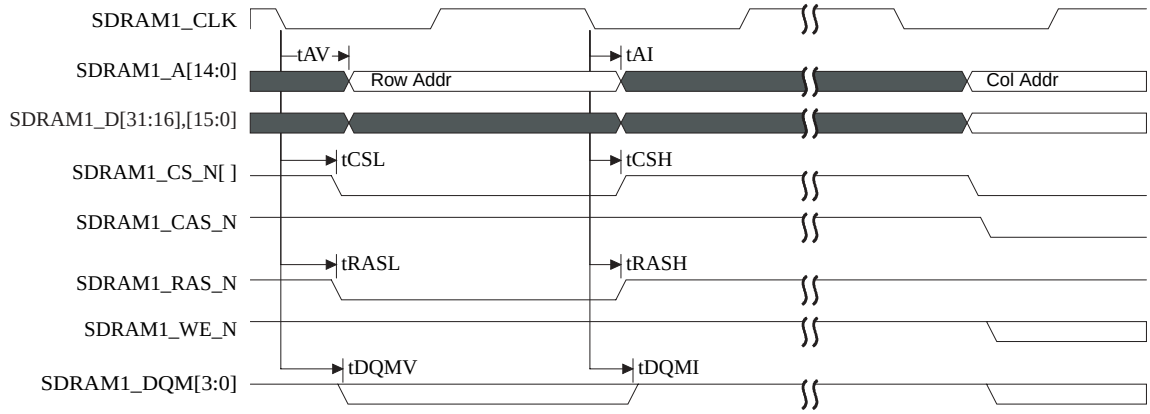


Figure 3-14 SDRAM RAS Timing

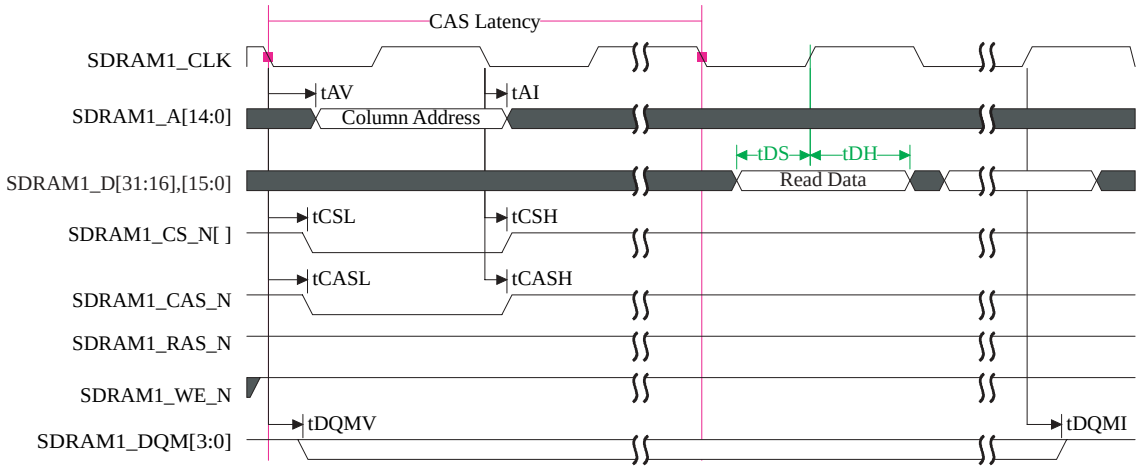


Figure 3-15 SDRAM Read Access Timing

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

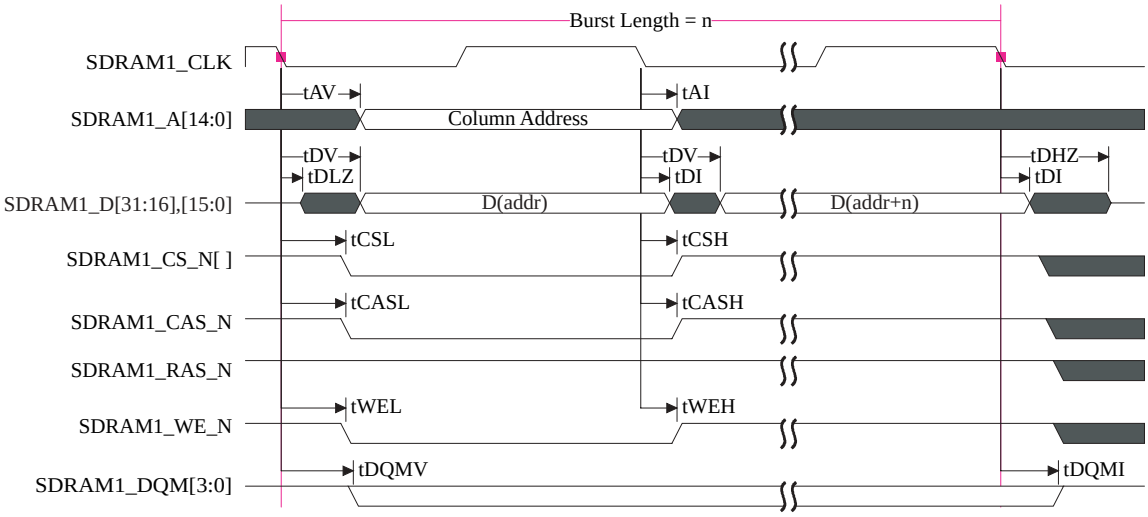


Figure 3-16 SDRAM Write Access Timing

3.2.8.3.1 SDRAM Controller - AC Characteristics

For any timing analysis, the measurement point for all signals is V_{IH} or V_{IL} (for rising or falling edges respectively).

All output timing parameters represent the point where the output signal transition begins (additional delays due to signal rise/fall times will have to be accounted for separately for a specific bus load).

All timing parameters are reported positive in the direction indicated on the diagram, while negative values indicate the opposite direction.

Table 3-56 SDRAM Controller - AC Characteristics

Symbol	Description	Min.	Max.	Unit	Note
t_{AV}	Clock to Column Address Valid	—	1.5	ns	1,3
t_{AI}	Clock to Column Address Invalid	-1.5	—	ns	1,3
t_{CSL}	Clock to Chip-Select Low	—	1.5	ns	1,3
t_{CSH}	Clock to Chip-Select High	-1.5	—	ns	1,3
t_{RASL}	Clock to RAS Low	—	1.5	ns	1,3
t_{RASH}	Clock to RAS High	-1.5	—	ns	1,3
t_{CASL}	Clock to CAS Low	—	1.5	ns	1,3
t_{CASH}	Clock to CAS High	-1.5	—	ns	1,3
t_{DQMV}	Clock to DQM Valid	—	1.5	ns	1,3

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-56 SDRAM Controller - AC Characteristics (Continued)

Symbol	Description	Min.	Max.	Unit	Note
t_{DQMI}	Clock to DQM Invalid	-1.5	—	ns	1,3
Read Cycle					
t_{DS}	Read Data Setup Time	1.0	—	ns	2
t_{DH}	Read Data Hold Time	3.0	—	ns	2
Write Cycle					
t_{DV}	Clock to Data Valid	—	1.5	ns	1
t_{DI}	Clock to Data Invalid	-1.5	—	ns	1
t_{DHZ}	Clock to Data Bus in Low Impedance	TBD	—	ns	1
t_{DHZ}	Clock to Data Bus in High Impedance	—	TBD	ns	1
t_{WEL}	Clock to Write Enable Low	—	1.5	ns	1
t_{WEH}	Clock to Write Enable High	-1.5	—	ns	1

[1] This parameter is an MSM output driving an external device input.
[2] This parameter is an external device output drivin.g an MSM input.
[3] This parameter is common to both Read and Write.

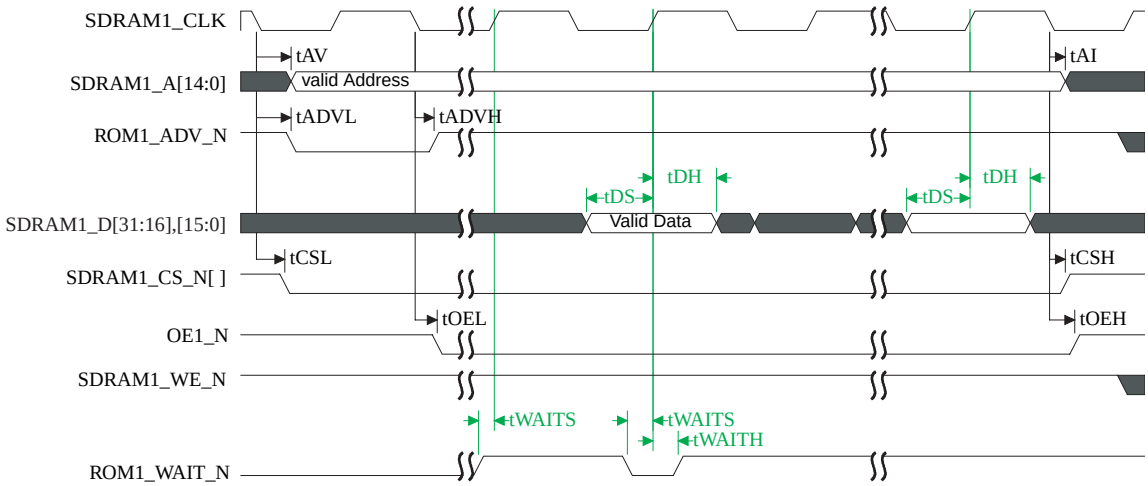


Figure 3-17 Burst Memory Read Timing

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

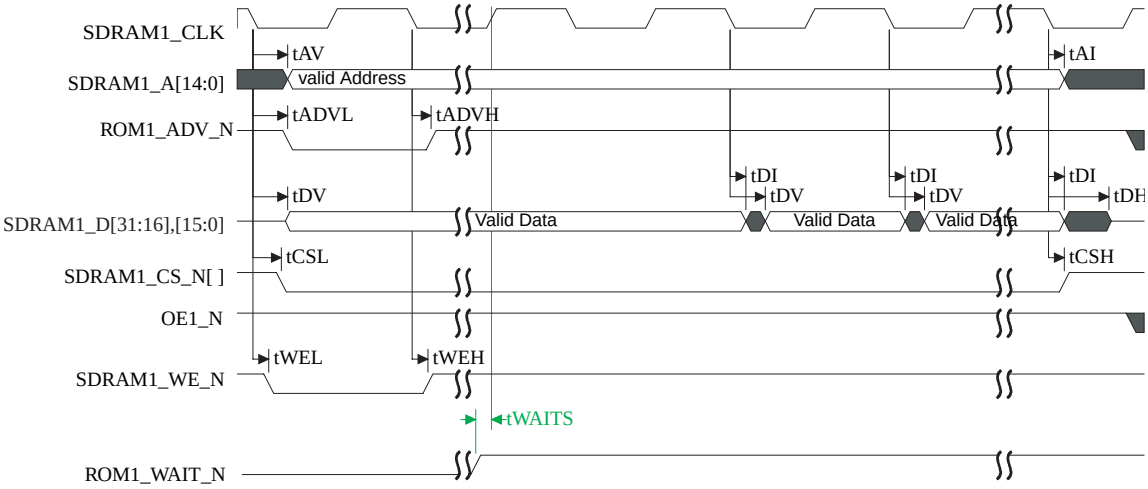


Figure 3-18 Burst Memory Write Timing

3.2.8.3.2 Burst Memory Controller - AC Characteristics

The measurement point for all input signals is V_{IH} or V_{IL} (for rising or falling edges respectively).

For output signals, all parameters represent the point where the output signal transition begins (additional delays due to rise/fall times will have to be accounted for separately for a specific bus load).

All timing parameters are reported as positive in the direction indicated on the diagram, while negative values indicate the opposite direction.

Table 3-57 Burst Memory Controller - AC Characteristics

Symbol	Description	Min.	Max.	Unit	Note
t_{AV}	Clock to Address Valid		1.5	ns	1,3
t_{AI}	Clock to Address Invalid	-1.5		ns	1,3
t_{CSL}	Clock to Chip-Select Low		1.5	ns	1,3
t_{CSH}	Clock to Chip-Select High	-1.5		ns	1,3
t_{ADVL}	Clock to ADV# Low		1.5	ns	1,3
t_{ADVH}	Clock to ADV# High	-1.5		ns	1,3
Read Cycle					
t_{OEL}	Clock to OE# Low		1.5	ns	1,3
t_{OEH}	Clock to OE# High	-1.5		ns	1,3
t_{DS}	Read Data Setup Time	1.0	—	ns	2
t_{DH}	Read Data Hold Time	3.0	—	ns	2

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-57 Burst Memory Controller - AC Characteristics

Symbol	Description	Min.	Max.	Unit	Note
t_{WAITS}	WAIT Signal Setup Time w.r.t. Clock	1.0	—	ns	2
t_{WAITH}	WAIT Signal Hold Time w.r.t. Clock	3.0	—	ns	2
Write Cycle					
t_{DV}	Clock to Data Valid		TBD	ns	1
t_{DI}	Clock to Data Invalid	TBD		ns	1
t_{DHZ}	Clock to Data Bus in Low Impedance	TBD		ns	1
t_{DHZ}	Clock to Data Bus in High Impedance		TBD	ns	1
t_{WEL}	Clock to Write Enable Low		TBD	ns	1
t_{WEH}	Clock to Write Enable High	TBD		ns	1

[1] This parameter is an MSM output driving an external device input.
[2] This parameter is an external device output drivin.g an MSM input.
[3] This parameter is common to both Read and Write.

3.2.8.3.3 Example Calculations

Determine the address setup time measured from the rising edge of the clock input to the SDRAM (t_{ADDR_SETUP}):

$$\begin{aligned} t_{ADDR\_SETUP} &= (\text{Minimum Half Clock Period}) - t_{AV}\,(\text{max}) \\ &= (T/2 - 0.5) - 1.5 \\ &\approx 4.8\ \text{ns} \end{aligned}$$

Determine the address hold time measured from the rising edge of the clock input to the SDRAM (t_{ADDR_HOLD}):

$$\begin{aligned} t_{ADDR\_HOLD} &= (\text{Minimum Half Clock Period}) + t_{AI}\,(\text{min}) \\ &= (T/2 - 0.5) + (-1.5) \\ &\approx 4.8\ \text{ns} \end{aligned}$$

NOTE These calculations do not account for the delays due to slow signal transition time or signal flight time between the MSM and the memory device. In a real system level timing analysis these effects must also be considered for a particular bus load.

3.2.8.4 EBI1 Page Mode Timing

cs_n is used to indicate any of the following pins:

- ROM1_CS_N[0]
- ROM1_CS_N[1]
- RAM1_CS_N[0]
- RAM1_CS_N[1]

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

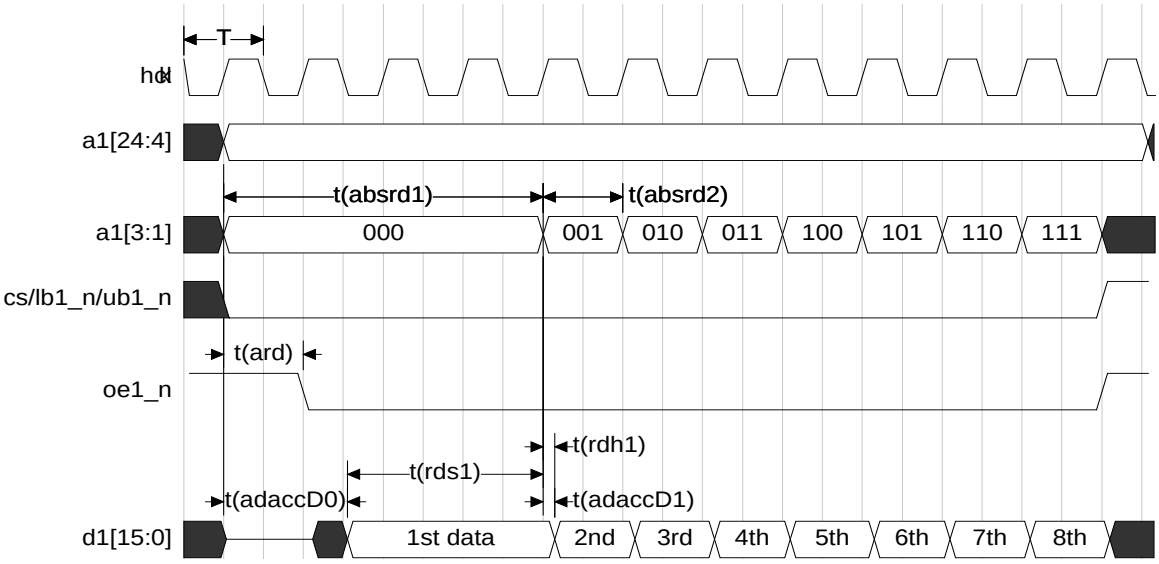


Figure 3-19 Page (Page Size = 8 HWORDS) Mode Read Timing for a 4-Word (32-bit) Burst

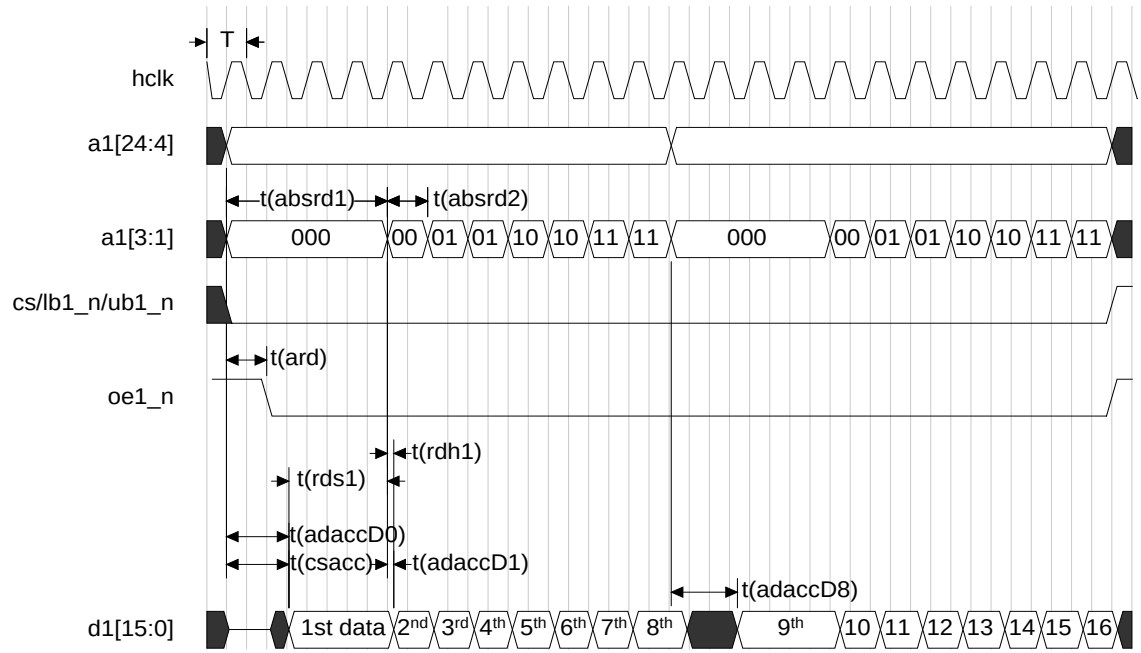


Figure 3-20 Page (Page Size = 8 HWORDS) Mode Read Timing for an 8-Word (32-Bit) Burst

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.9 External Bus Interface 2 (EBI2)

3.2.9.1 External Bus Interface 2 (EBI1) Memory Controller

EBI2 is supports only asynchronous devices.

3.2.9.1.1 Wait State Definition

Table 3-58 EBI2 Wait States

Wait	Description (corresponding field on configuration register)
$m_{w(rd)}$	DELTA READ
$n_{w(rd)}$	WAIT READ - 1
$m_{w(wr)}$	DELTA WRITE
$n_{w(wr)}$	WAIT WRITE - 1
$h_{w(rd)}$	HOLD READ WAIT
$h_{w(wr)}$	HOLD WRITE WAIT - 1
r_w	RECOVERY CYCLE or CHIP SELECT SETUP (the greater of the two)

3.2.9.1.2 Timing Parameters

Table 3-59 EBI2 Timing Parameters Where V_{DD_P2}= 2.5V

Symbol	Parameter	Min	Max
ι	TCSD	-1.400	1.440
α1	TRDH	-9.550	-4.830
α2	TRDH1	-10.030	-4.640
α3	TRDH2	-9.480	-4.800
α3	TRDH4	-8.990	-4.770
χ1	TRDS	5.240	10.180
χ2	TRDS1	5.290	10.850
χ3	TRDS2	5.100	10.150
χ3	TRDS4	5.140	9.700

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-59 EBI2 Timing Parameters Where V_{DD_P2}= 2.5V (Continued)

Symbol	Parameter	Min	Max
ε1	TCSVWR	-0.540	0.950
ε2	TCSVRD	-0.220	1.010
γ	TRD	-0.340	0.180
η1	TRDA	-1.330	1.960
η2	TRDCS	-2.730	-0.470
κ1	TBSWRH	-0.430	-0.150
λ1	TARD	0.680	1.580
λ2	TCSRD	0.270	2.770
λ3	TARD	0.670	1.510
μ1	TWRA	1.040	2.440
μ2	TWRCS	-2.460	-0.120
π1	TABSWR	0.610	2.560
θ1	TACS	-1.430	1.040
θ2	TACS	-1.390	2.010
σ1	TDHWR	-0.220	0.640
σ2	TDHBSWR2	-0.150	0.500
σ2	TDHBSWR4	-0.130	0.510
σ3	TDHBSWR	-2.350	-0.270
υ1	TDSUWR	0.290	1.520
υ2	TDSUBSWR1	0.360	1.510
υ3	TDSUBSWR2	0.510	2.600
υ3	TDSUBSWR4	1.090	2.260
ω1	TAWR	0.380	2.290
ω2	TCSWR	0.070	2.890
ω3	TAWR	0.370	1.540
ξ1	TAVRD	-0.250	3.120
ξ2	TAVWR	1.350	4.450
ψ1	TABSRD1	-1.450	0.200
ψ2	TABSRD2	0.760	2.580

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-59 EBI2 Timing Parameters Where V_{DD_P2}= 2.5V (Continued)

Symbol	Parameter	Min	Max
ψ2	TABSRD4	1.290	2.600
ζ1	TWR	-0.510	-0.020
ζ2	TBSWR1	-0.480	-0.070
ζ3	TBSWR2	-0.430	-0.070

Table 3-60 EBI2 Timing Parameters Where V_{DD_P2}= 1.77 V

Symbol	Parameter	Min	Max
ι	TCSD	-1.370	0.830
α1	TRDH	-8.420	-4.430
α2	TRDH1	-8.770	-4.280
α3	TRDH2	-8.090	-4.380
α3	TRDH4	-7.730	-4.370
χ1	TRDS	4.900	9.180
χ2	TRDS1	4.940	9.730
χ3	TRDS2	4.780	8.870
χ3	TRDS4	4.780	8.580
ε1	TCSVWR	0.150	1.050
ε2	TCSVRD	0.190	0.980
γ	TRD	-0.380	0.120
η1	TRDA	-1.360	1.480
η2	TRDCS	-2.670	-0.500
κ1	TBSWRH	-0.460	-0.160
λ1	TARD	0.690	1.620
λ2	TCSRD	0.280	2.230
λ3	TARD	0.680	1.660
μ1	TWRA	0.920	1.870
μ2	TWRCS	-2.370	-0.160
π1	TABSWR	0.620	2.610
θ1	TACS	-0.800	1.050

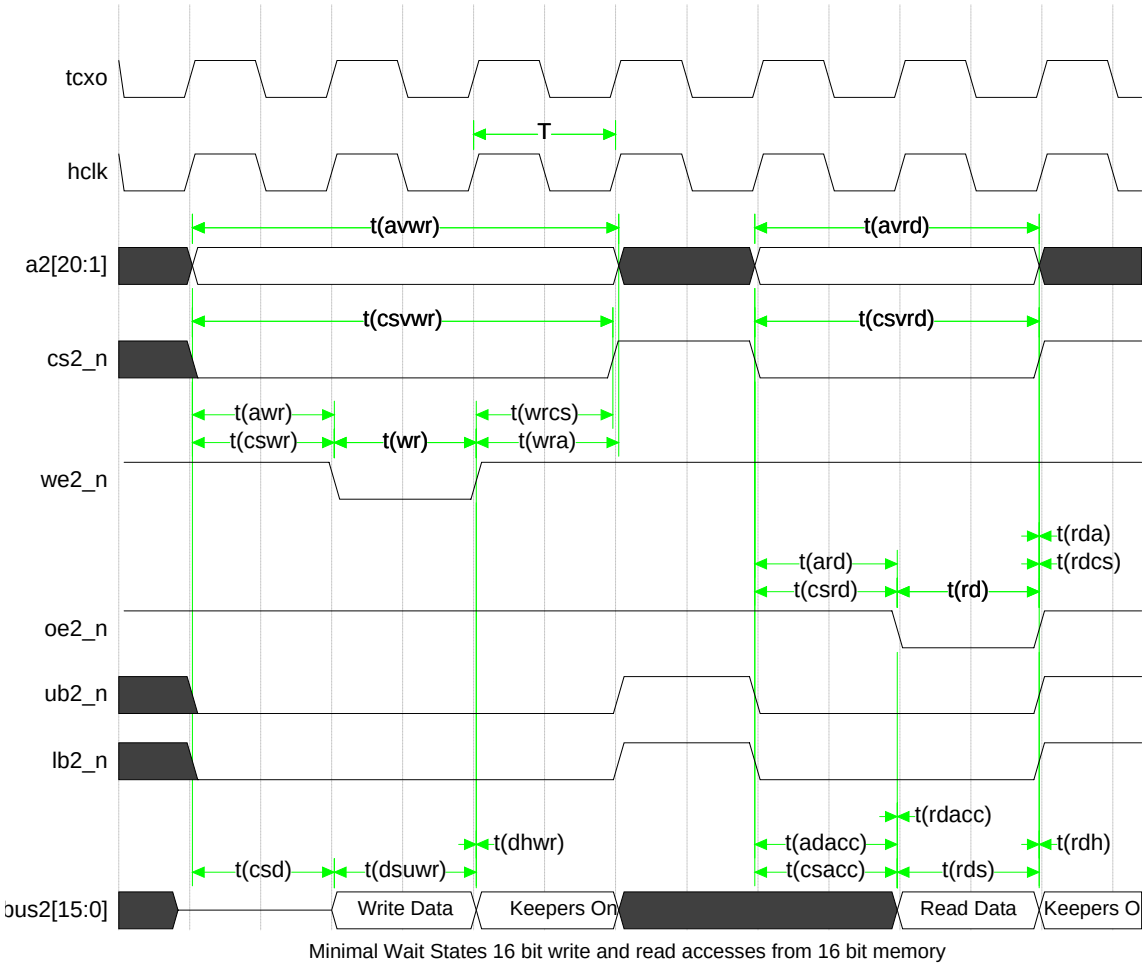
NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-60 EBI2 Timing Parameters Where V_{DD_P2}= 1.77 V (Continued)

Symbol	Parameter	Min	Max
θ2	TACS	-1.010	1.850
σ1	TDHWR	0.290	0.860
σ2	TDHBSWR2	0.300	0.680
σ2	TDHBSWR4	0.350	0.680
σ3	TDHBSWR	-2.290	-0.240
υ1	TDSUWR	0.220	1.420
υ2	TDSUBSWR1	0.280	1.370
υ3	TDSUBSWR2	0.500	2.660
υ3	TDSUBSWR4	1.100	2.270
ω1	TAWR	0.380	2.110
ω2	TCSWR	0.070	2.070
ω3	TAWR	0.360	1.550
ξ1	TAVRD	-0.230	2.650
ξ2	TAVWR	1.230	3.700
ψ1	TABSRD1	-1.430	0.290
ψ2	TABSRD2	0.780	2.610
ψ2	TABSRD4	1.300	2.630
ζ1	TWR	-0.520	-0.080
ζ2	TBSWR1	-0.490	-0.100
ζ3	TBSWR2	-0.510	-0.130

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.9.1.3 EBI2 Native Timing



Minimal Wait States 16 bit write and read accesses from 16 bit memory

Figure 3-21 EBI2 Minimal Wait States 16-bit Write and Read Accesses From 16-Bit Memory

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

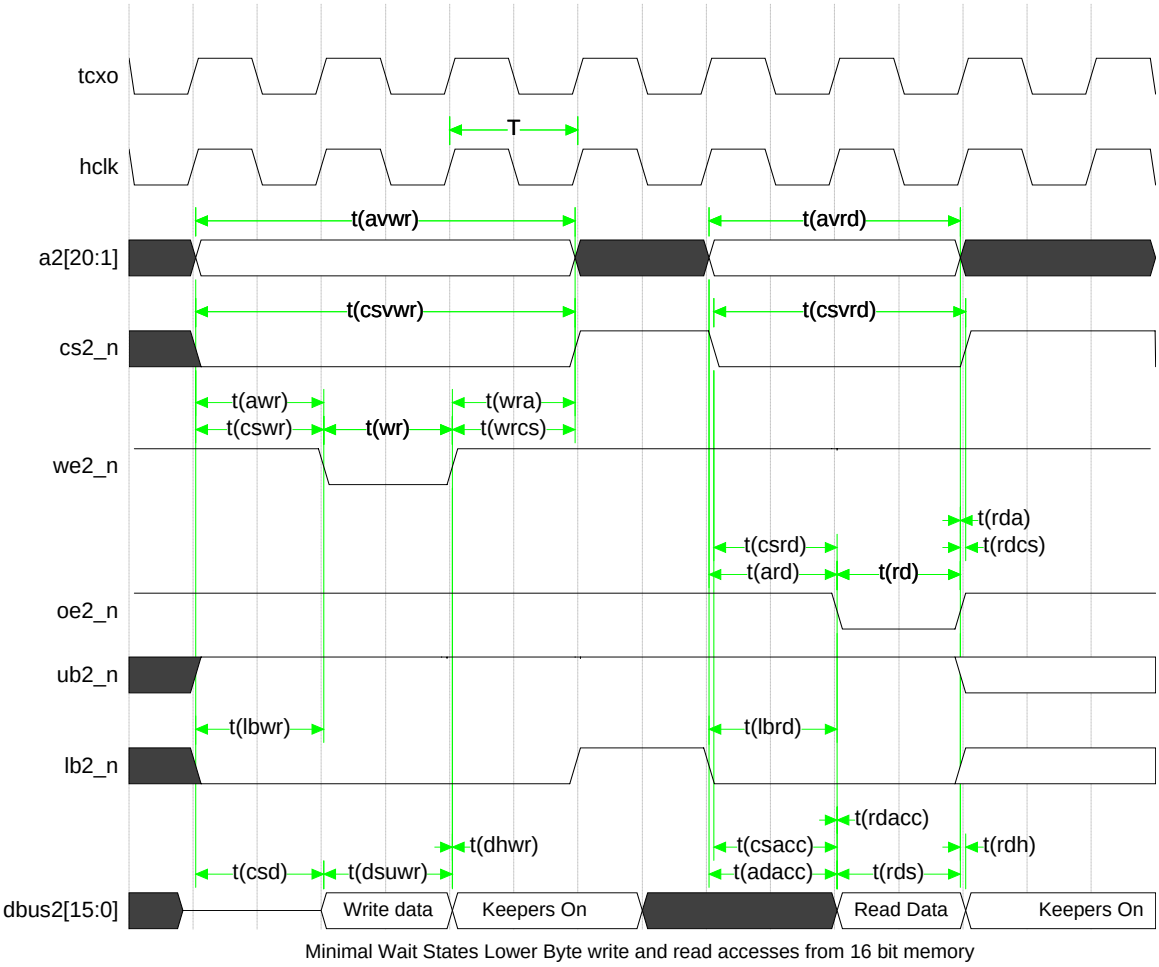


Figure 3-22 EBI2 Minimal Wait States Lower Byte Write and Read Accesses From 16-Bit Memory

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

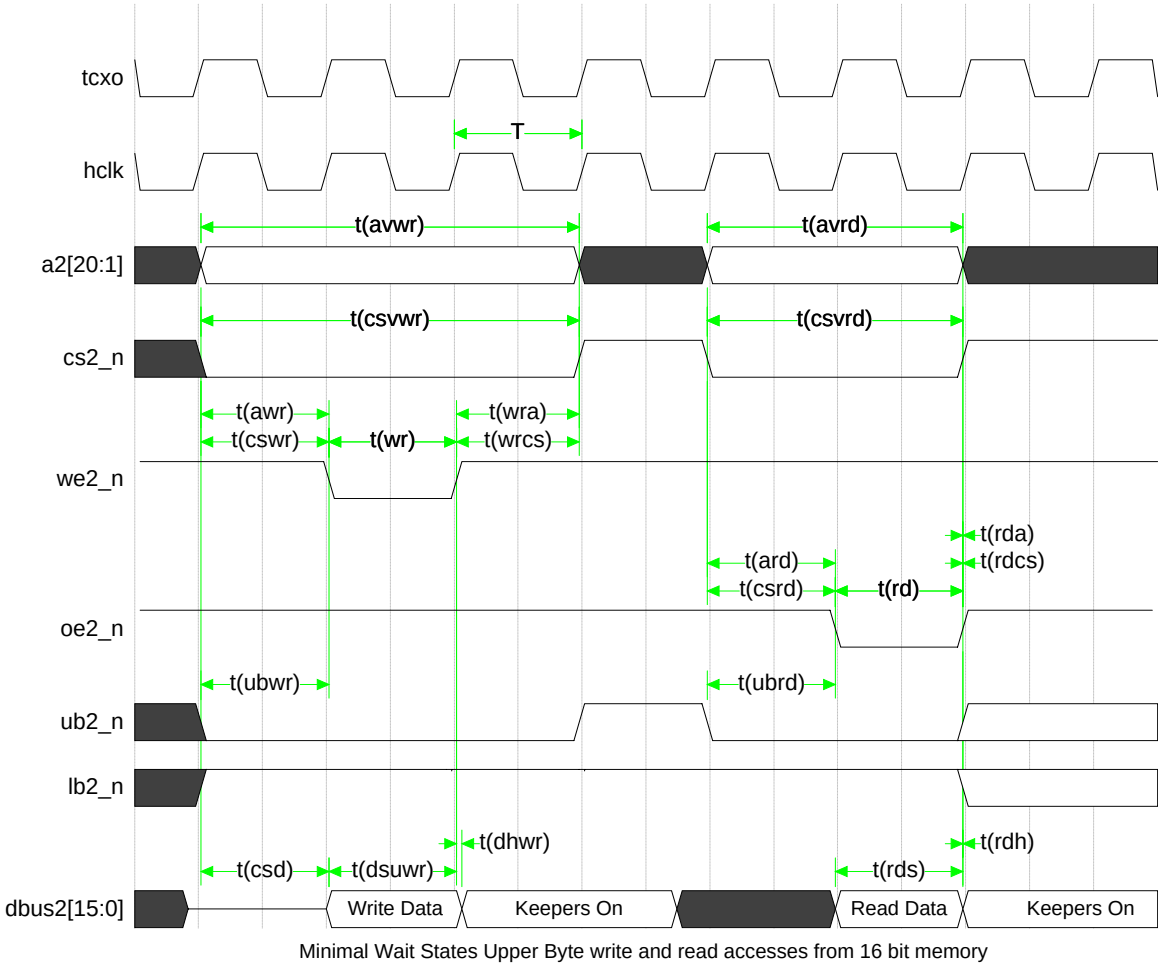


Figure 3-23 EBI2 Minimal Wait States Upper Byte Write and Read Accesses From 16-Bit Memory

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

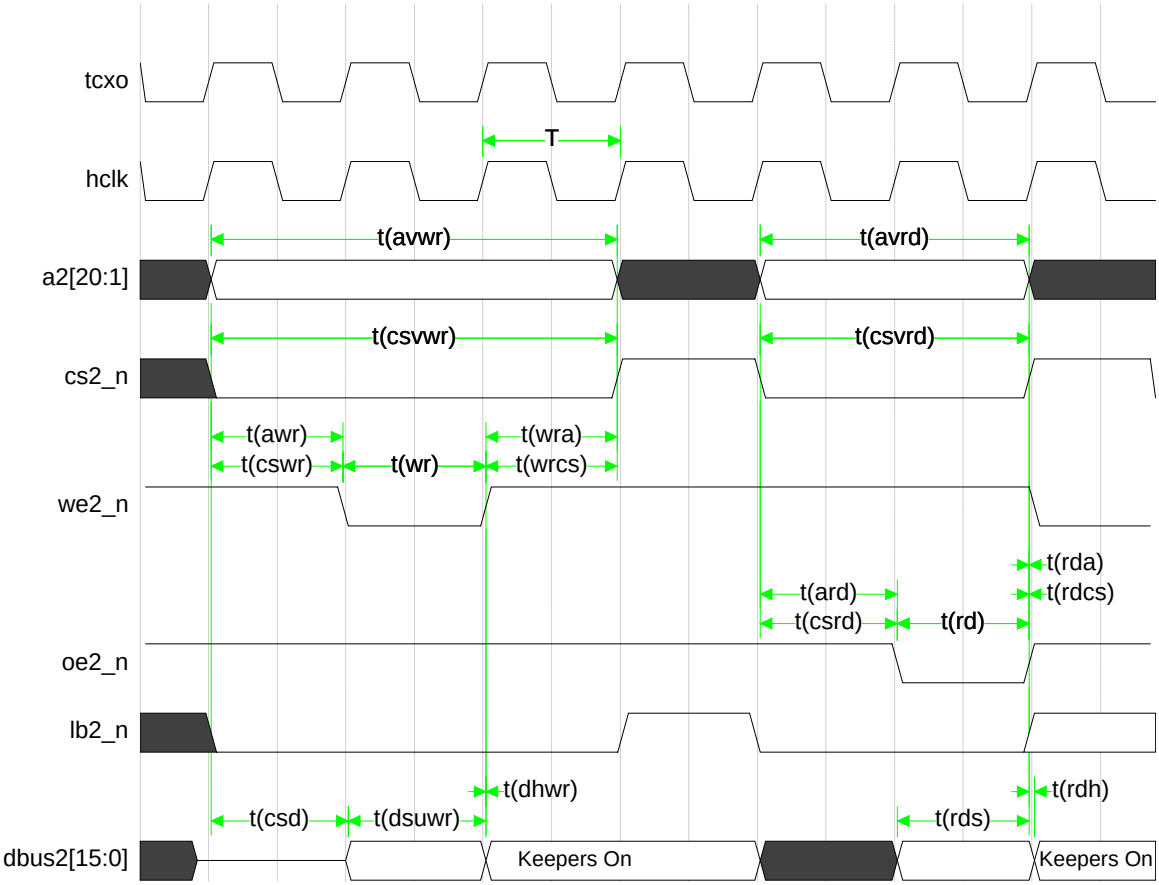


Figure 3-24 EBI2 Minimal Wait States 8-bit Write and Read Accesses (Even Address) From 8-Bit Memory

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

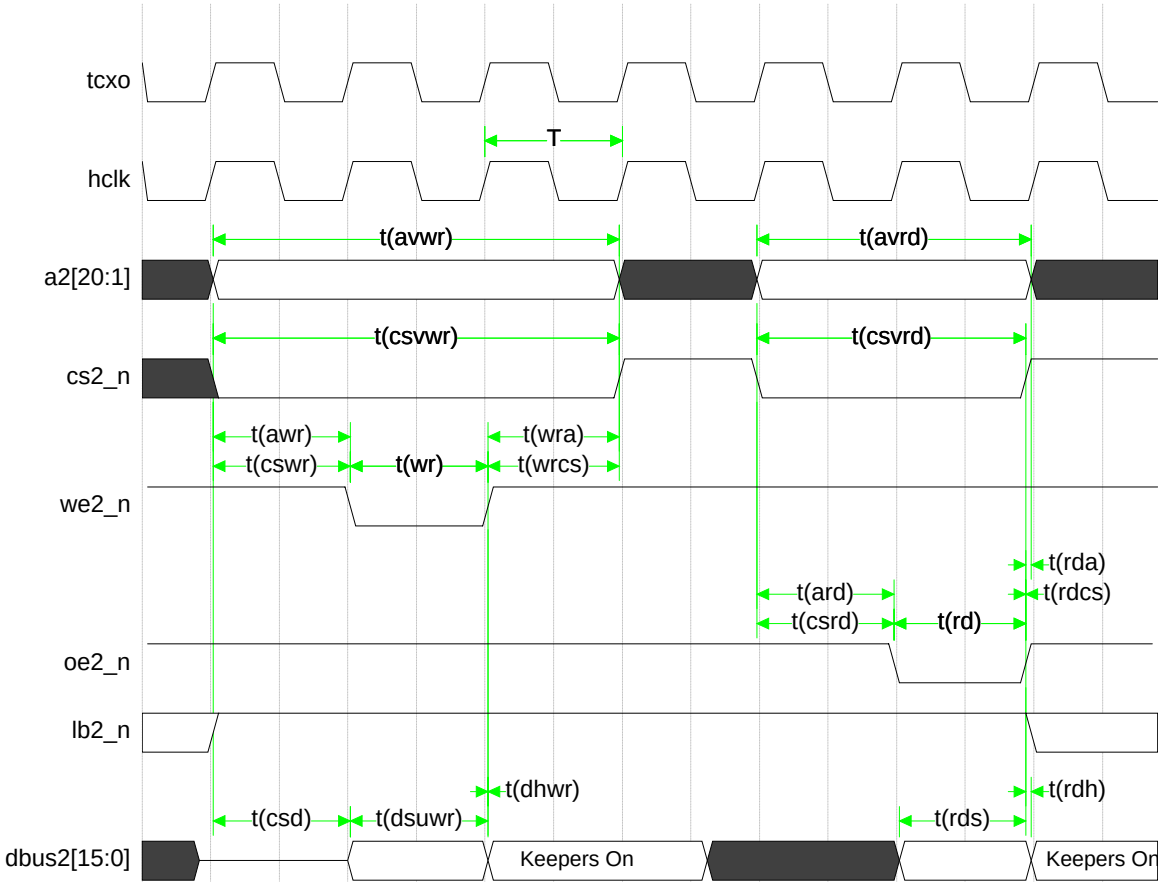


Figure 3-25 EBI2 Minimal Wait States 8-Bit Write and Read Accesses (Odd Address) From 8-Bit Memory

Table 3-61 EBI2 Native Mode Write Timing

Parameter		Equation	Notes
t_{avwr}	Address valid to address invalid	$3T + (m_{w(wr)} + n_{w(wr)} + h_{w(wr)} + r_{w(wr)})T - \xi_2$	Take worst case among all pins.
t_{csvwr}	Chip select active	$3T + (m_{w(wr)} + n_{w(wr)} + h_{w(wr)})T - \varepsilon_1$	For a non-bus-sized, non-sequential access
t_{dsuwr}	Write data setup	$T + (m_{w(wr)} + n_{w(wr)})T - \upsilon_1$	
t_{dhwr}	Write data hold	σ_1	By design, keepers will hold data, spec T in user manual.
t_{awr}	Address valid to write active	$T + (h_{w(wr)})T - \omega_1$	Take worst case of a normal write or a bus-sized write.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-61 EBI2 Native Mode Write Timing (Continued)

Parameter		Equation	Notes
t_{cswr}	Chip select active to write active	$T - \omega_2$	
t_{wr}	Write active	$T + (m_{w(wr)} + n_{w(wr)})T - \zeta_1$	
t_{csd}	Chip select active to data valid	$T - \iota$	Will determine allowable trdh.
t_{wra}	Write inactive to address invalid	$T + (h_{w(wr)})T - \mu_1$	
t_{wrcs}	Write inactive to chip select inactive	$T + (h_{w(wr)})T - \mu_2$	
t_{acs}	Address Valid to chip select active	$(r_w)T - \theta_2$	

Table 3-62 EBI2 Native Mode Read Timing

Parameter		Equation	Notes
t_{avrd}	Address valid to address invalid	$2T + (m_{w(rd)} + n_{w(rd)} + r_w + h_{w(rd)})T - \xi_1$	
t_{csvrd}	Chip select active	$2T + (m_{w(rd)} + n_{w(rd)} + h_{w(rd)})T - \varepsilon_2$	For a non-bus-sized, non-sequential access
t_{rds}	Read data setup	$h_{w(rd)} + \chi_1$	
t_{rdh}	Read data hold	$h_{w(rd)} + \alpha_1$	Max value also limited by t_{csd}.
t_{ard}	Address valid to read active	$T + (r_w)T - \lambda_1$	Take worst case of a normal read or a bus-sized read.
t_{csrd}	Chip select active to read active	$T - \lambda_2$	
t_{rd}	Read active	$T + (m_{w(rd)} + n_{w(rd)})T - \gamma$	
t_{rda}	Read inactive to address invalid	$(h_{w(rd)})T - \eta_1$	
t_{rdcs}	Read inactive to chip select inactive	$(h_{w(rd)})T - \eta_2$	
t_{acs}	Address valid to chip select active	$(r_w)T - \theta_1$	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.9.1.4 EBI2 Bus Sized Access Timing

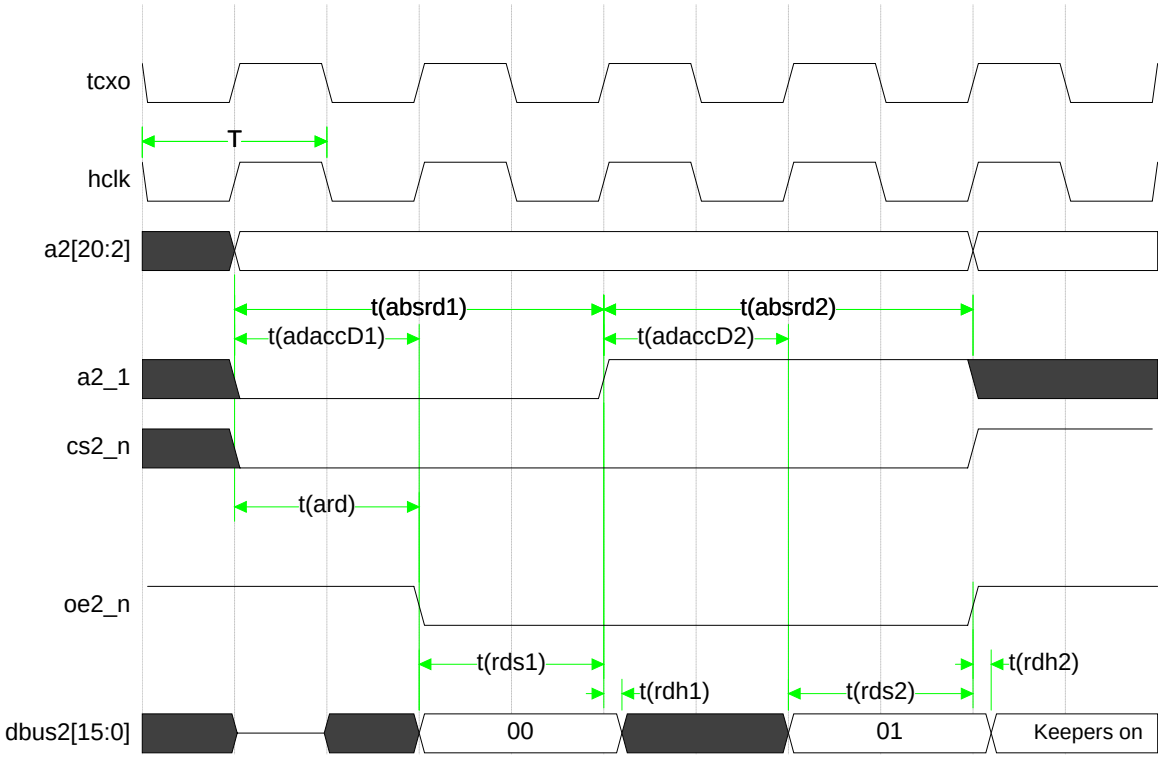


Figure 3-26 EBI2 Word Read Access From 16-Bit Memory

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

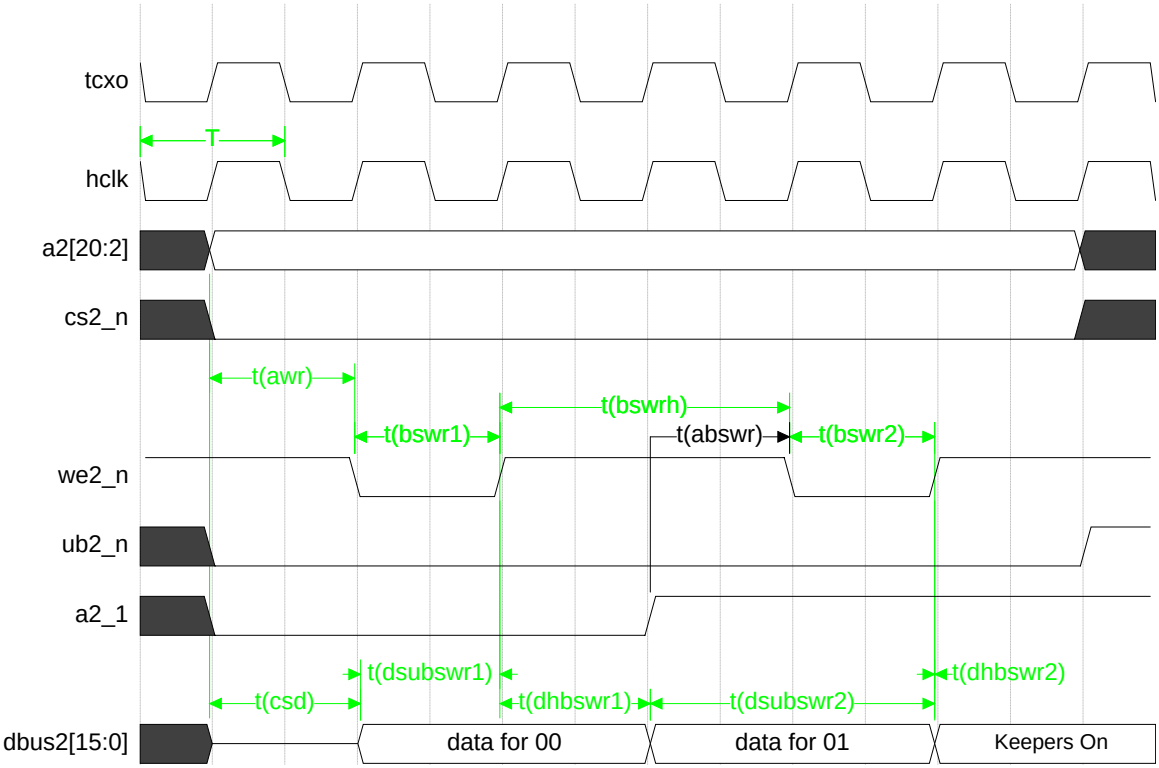


Figure 3-27 EBI2 Word Bus Sized Write Access to 16-Bit Memory

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

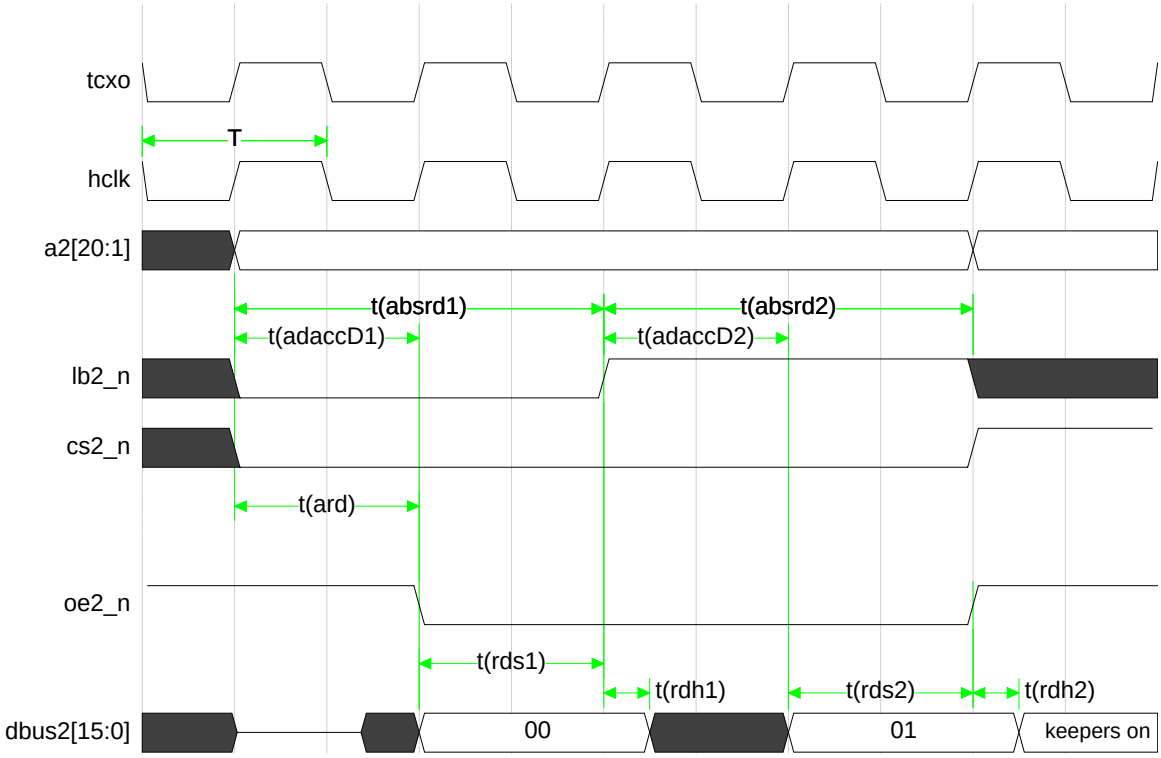


Figure 3-28 EBI2 Half Word Read Access From 8-Bit Memory

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

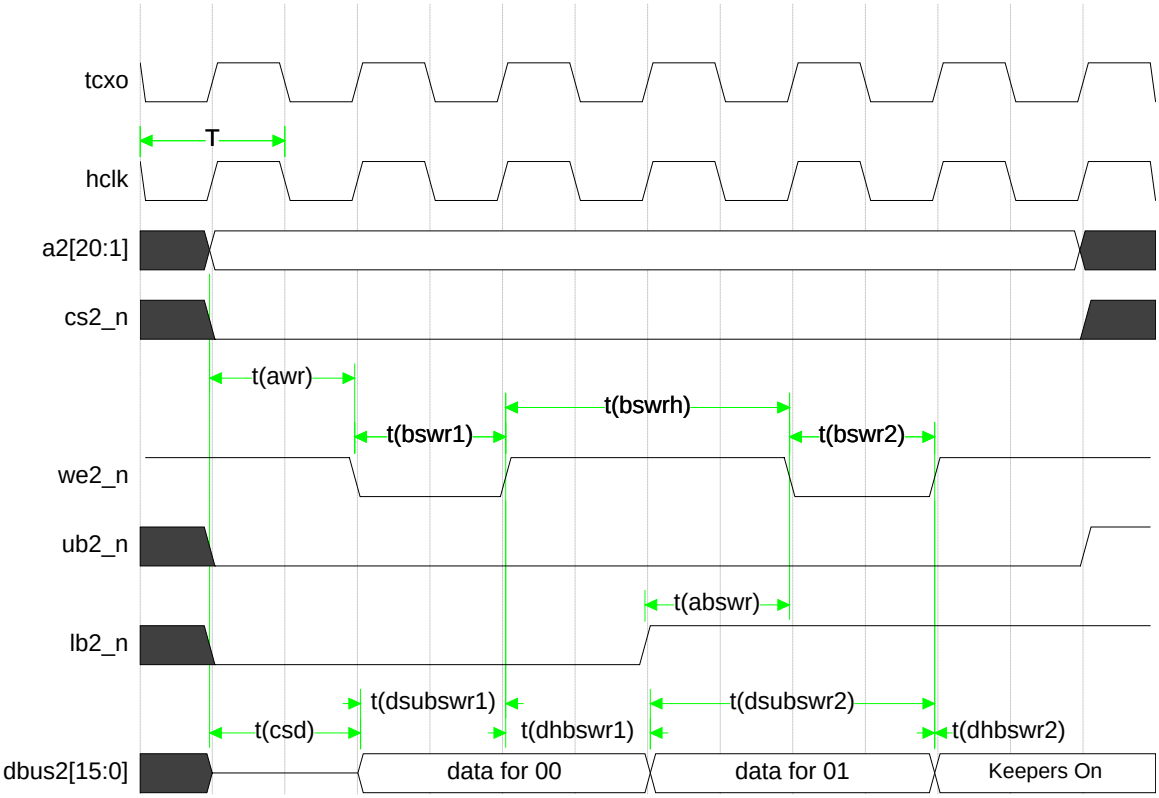


Figure 3-29 EBI2 Half Word Bus Sized Write Access to 8-Bit Memory

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

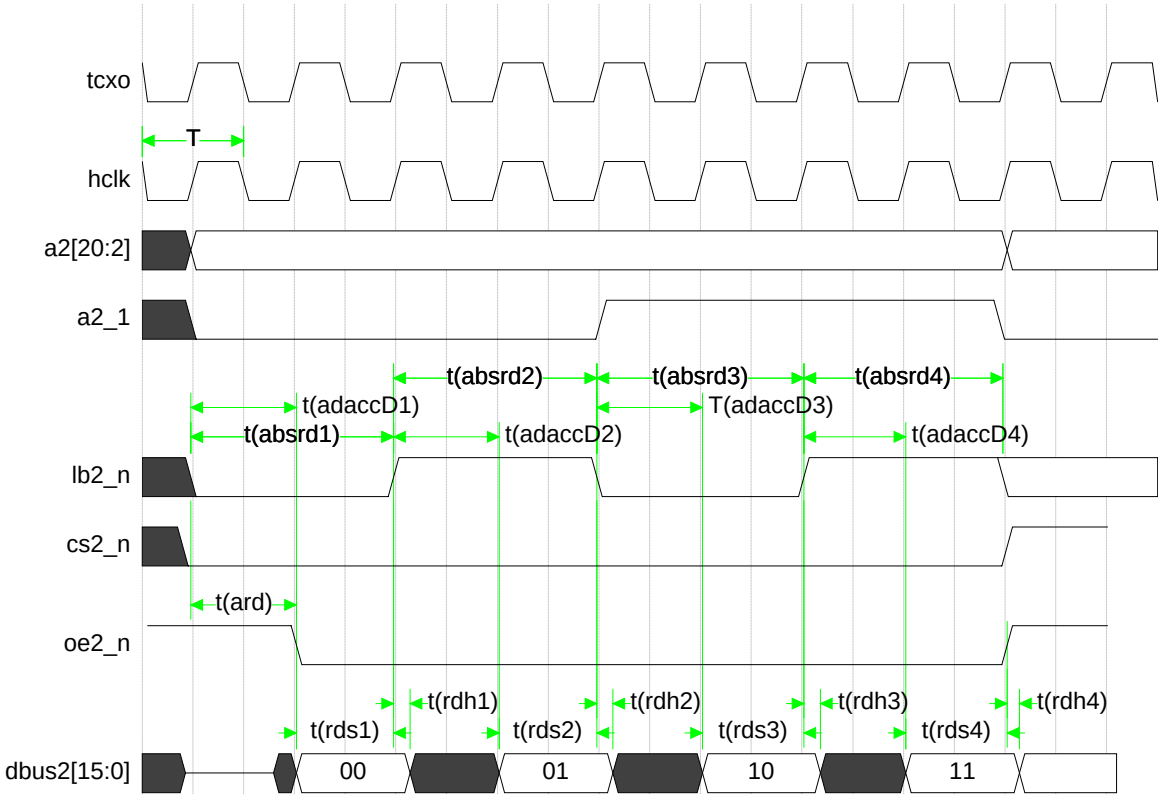


Figure 3-30 EBI2 Word Read Access From 8-Bit Memory

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

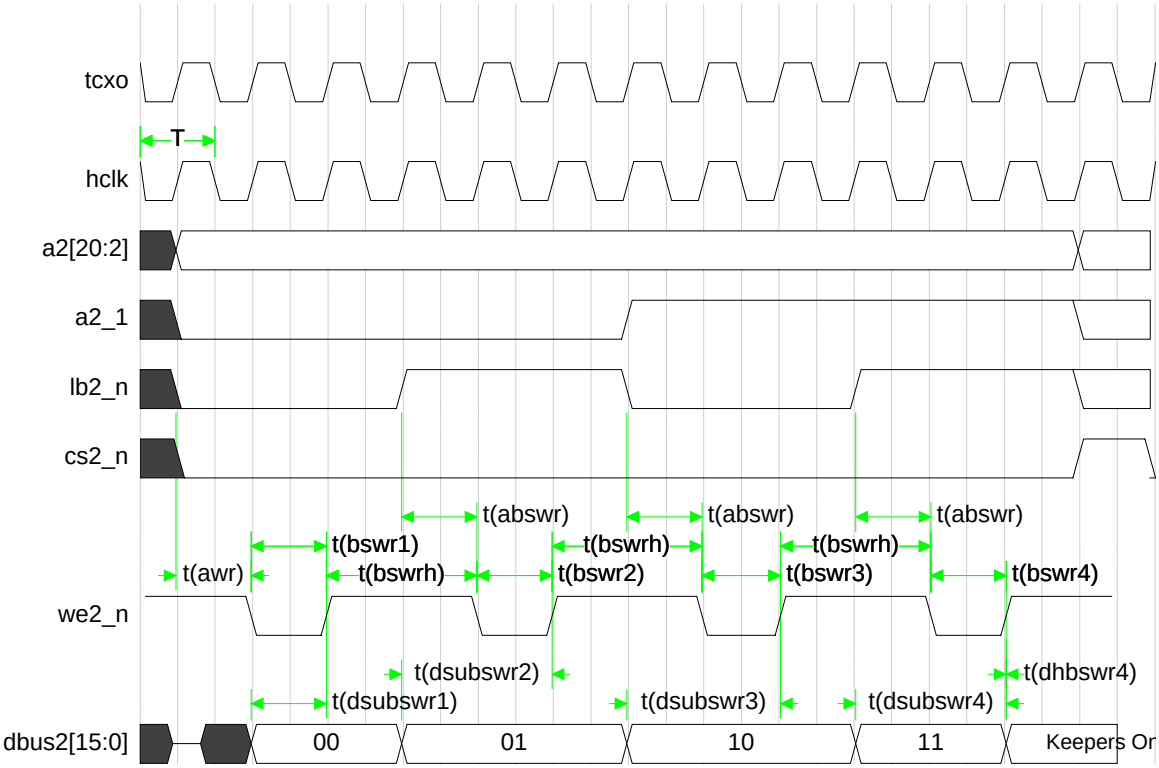


Figure 3-31 EBI2 Word Write Access to 8-Bit Memory

Table 3-63 EBI2 Bus Sized Write Timing

Parameter		Equation	Notes
t_{bswr1}	Pulse width of first write active	$T + (m_{w(wr)} + n_{w(wr)})T - \zeta_2$	
t_{bswr2} t_{bswr3} t_{bswr4}	Pulse width of second, third(32x8 only), fourth(32x8 only) write active	$T + n_{w(wr)}T - \zeta_3$	
t_{bswrh}	Pulse width of write inactive between bus-sized writes	$2T + h_wT - \kappa_1$	
t_{awr}	Address valid to first write active	$(r_w)\ T + T - \omega_3$	Take worst case among all address pins and a normal write case
t_{abswr}	Address[1] valid to the second write active	$T - \pi_1$	
$t_{dsubswr1}$	Bus-sized write data setup for first write	$T + (m_{w(wr)} + n_{w(wr)})T - \upsilon_2$	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-63 EBI2 Bus Sized Write Timing (Continued)

Parameter		Equation	Notes
$t_{dsubswr2}$ or $t_{dsubswr4}$	Bus-sized write data setup for the last write	$2T + n_wT - \upsilon_3$	Take worst case among the 2nd, or 3rd, or 4th writes.
$t_{dhbswr1}$	Bus-sized write data hold (first write)	$T + (h_{w(wr)})T - \sigma_3$	Worst case among all write cycles except the last.
$t_{dhbswr2}$ or $t_{dhbswr4}$	Bus-sized write data hold (last write)	σ_2	Last cycle. Time when drivers turn-off and keepers turn-on.

Table 3-64 EBI2 Bus Sized Read Timing

Parameter		Equation	Notes
t_{ard}	Address valid to first read active	$T + (r_w)T - \lambda_3$	Take worst case among all address pins and a normal read case
t_{absrd1}	Address valid to end of first read	$2T + (m_{w(rd)} + n_{w(rd)} + r_{w(rd)} + h_{w(rd)})T - \psi_1$	
t_{absrd2} t_{absrd3} t_{absrd4}	Address valid to end of the second, third(32x8 only), and fourth(32x8 only) reads	$2T + (n_{w(rd)} + h_{w(rd)})T - \psi_2$	
t_{rds1}	Bus-sized read data setup for the first read	$h_{w(rd)} - \chi_2$	Take worst case among all reads and normal read
t_{rdh1}	Bus-sized read data hold for the first read	$h_{w(rd)} - \alpha_2$	Take worst case among all reads and normal read. Max value also limited by $t_{csd.}$
t_{rds2} t_{rds4}	Bus-sized read data setup for the second/fourth read	$h_{w(rd)} - \chi_3$	Take worst case among all reads and normal read
t_{rdh2} t_{rdh4}	Bus-sized read data hold for the second/fourth read	$h_{w(rd)} - \alpha_3$	Take worst case among all reads and normal read. Max value also limited by $t_{csd.}$

3.2.9.2 Parallel LCD Interface

The MSM6100 supports the following types of parallel LCD devices:

- Memory-mapped or port-mapped
- Motorola or Intel interface timing
- 16 or 8-bit data

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.9.2.1 Wait State Definition

Table 3-65 LCD Wait States

Wait	Description (Corresponding Field on Configuration Register)
$n_{w(rd)}$	WAIT READ - 1 (for INTEL) WAIT READ (for MOTOROLA)
$n_{w(wr)}$	WAIT WRITE - 1 (for INTEL) WAIT WRITE (for MOTOROLA)
$h_{w(rd)}$	HOLD READ WAIT
$h_{w(wr)}$	HOLD WRITE WAIT - 1 (for INTEL)
r_w	RECOVERY CYCLE or CS_SETUP (whatever is greater)
$s_{w(rd)}$	LCD_E_SETUP_READ
$s_{w(wr)}$	LCD_E_SETUP_WRITE
$p_{w(rd)}$	LCD_E_HIGH_READ
$p_{w(wr)}$	LCD_E_HIGH_WRITE

3.2.9.2.2 Intel Interface Timing Parameters

Table 3-66 Intel-Type Parallel LCD Timing Parameters Where V_{DD_P2} = 2.5V

Symbol	Parameter	Min	Max
$\alpha 1$	TRDH	-8.470	-4.960
$\alpha 2$	TRDH1	-8.850	-4.730
$\alpha 3$	TRDH2	-8.270	-4.920
$\alpha 3$	TRDH4	-7.900	-4.860
$\chi 1$	TRDS	5.380	9.010
$\chi 2$	TRDS1	5.390	9.560
$\chi 3$	TRDS2	5.230	8.930
$\chi 3$	TRDS4	5.270	8.500
$\varepsilon 1$	TCSVWR	0.090	0.730
$\varepsilon 2$	TCSVRD	0.130	0.780
γ	TRD	-0.280	0.040

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-66 Intel-Type Parallel LCD Timing Parameters Where V_{DD_P2} = 2.5V (Continued)

Symbol	Parameter	Min	Max
η1	TRDA	-0.380	1.520
η2	TRDCS	-2.630	-1.300
ι	TCSD	0.190	0.880
κ1	TBSWRH	0.150	0.400
λ1	TARD	0.700	1.430
λ2	TCSRD	1.240	2.150
λ3	TARD	0.690	1.390
μ1	TWRA	1.080	1.890
μ2	TWRCS	-2.140	-1.040
π1	TABSWR	0.610	2.360
θ1	TACS	-0.980	-0.300
θ2	TACS	-0.980	-0.320
σ1	TDHWR	0.100	0.510
σ2	TDHBSWR2	0.110	0.540
σ2	TDHBSWR4	0.120	0.550
σ3	TDHBSWR	-2.070	-0.790
υ1	TDSUWR	0.320	1.020
υ2	TDSUBSWR1	0.360	1.160
υ3	TDSUBSWR2	1.020	2.330
υ3	TDSUBSWR4	1.100	2.000
ω1	TAWR	0.390	1.340
ω2	TCSWR	1.090	2.010
ω3	TAWR	0.400	1.410
ξ1	TAVRD	0.440	2.540
ξ2	TAVWR	1.410	2.680
ψ1	TABSRD1	-1.300	0.040
ψ2	TABSRD2	0.780	2.430
ψ2	TABSRD4	1.310	2.410
ζ1	TWR	-0.430	-0.140

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-66 Intel-Type Parallel LCD Timing Parameters Where V_{DD_P2} = 2.5V (Continued)

Symbol	Parameter	Min	Max
ζ2	TBSWR1	-0.400	-0.100
ζ3	TBSWR2	-0.370	-0.120

Table 3-67 Intel-Type Parallel LCD Timing Parameters Where V_{DD_P2} = 1.77V

Symbol	Parameter	Min	Max
α1	TRDH	-8.140	-4.530
α2	TRDH1	-8.580	-4.350
α3	TRDH2	-7.860	-4.470
α3	TRDH4	-7.510	-4.480
χ1	TRDS	5.010	8.920
χ2	TRDS1	5.010	9.560
χ3	TRDS2	4.890	8.670
χ3	TRDS4	4.880	8.390
ε1	TCSVWR	0.170	0.920
ε2	TCSVRD	0.140	0.970
γ	TRD	-0.400	0.020
η1	TRDA	-0.480	1.380
η2	TRDCS	-2.900	-1.320
ι	TCSD	0.300	0.910
κ1	TBSWRH	0.160	0.510
λ1	TARD	0.680	1.780
λ2	TCSRD	1.240	2.330
λ3	TARD	0.690	1.780
μ1	TWRA	0.920	1.720
μ2	TWRCS	-2.450	-1.110
π1	TABSWR	0.620	2.690
θ1	TACS	-0.960	-0.280
θ2	TACS	-1.180	-0.280
σ1	TDHWR	0.320	0.660
σ2	TDHBSWR2	0.310	0.700

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-67 Intel-Type Parallel LCD Timing Parameters Where V_{DD_P2} = 1.77V

Symbol	Parameter	Min	Max
σ2	TDHBSWR4	0.360	0.680
σ3	TDHBSWR	-2.250	-0.820
υ1	TDSUWR	0.290	1.130
υ2	TDSUBSWR1	0.280	1.300
υ3	TDSUBSWR2	1.050	2.660
υ3	TDSUBSWR4	1.090	2.310
ω1	TAWR	0.410	1.720
ω2	TCSWR	1.080	2.190
ω3	TAWR	0.380	1.680
ξ1	TAVRD	0.350	2.630
ξ2	TAVWR	1.270	2.830
ψ1	TABSRD1	-1.480	0.120
ψ2	TABSRD2	0.790	2.710
ψ2	TABSRD4	1.340	2.690
ζ1	TWR	-0.480	-0.150
ζ2	TBSWR1	-0.470	-0.100
ζ3	TBSWR2	-0.460	-0.150

3.2.9.2.3 Intel LCD Timing Diagrams and Equations

Table 3-68 Parallel LCD Native Mode Write Timing

Parameter		Equation	Notes
t_{avwr}	Address valid to address invalid	$3T + (n_{w(wr)} + h_{w(wr)} + r_{w(wr)})T - \xi_2$	Rake worst case among all pins.
t_{csvwr}	Chip select active	$3T + (n_{w(wr)} + h_{w(wr)})T - \varepsilon_1$	For a non-bus-sized, non-sequential access
t_{dsuwr}	Write data setup	$T + (n_{w(wr)})T - \upsilon_1$	
t_{dhwr}	Write data hold	σ_1	By design, keepers will hold data, spec T in user manual.
t_{awr}	Address valid to write active	$T + (h_{w(wr)})T - \omega_1$	Rake worst case of a normal write or a bus-sized write.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

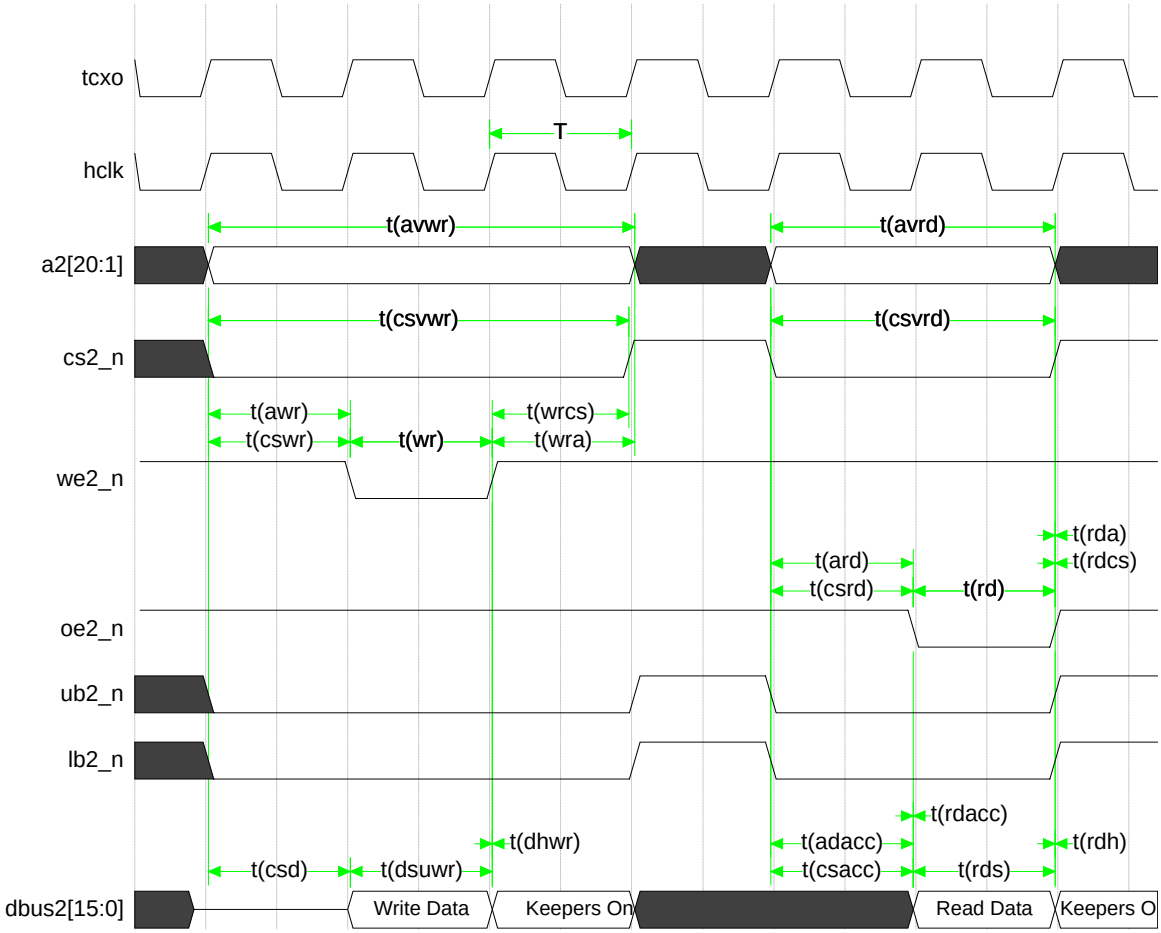
Table 3-68 Parallel LCD Native Mode Write Timing

Parameter		Equation	Notes
t_{cswr}	Chip select active to write active	$T - \omega_2$	
t_{wr}	Write active	$T + (n_{w(wr)})T - \zeta_1$	
t_{csd}	Chip select active to data valid	$T - \iota$	Will determine allowable trdh.
t_{wra}	Write inactive to address invalid	$T + (h_{w(wr)})T - \mu_1$	
t_{wrcs}	Write inactive to chip select inactive	$T + (h_{w(wr)})T - \mu_2$	
t_{acs}	Address Valid to chip select active	$(r_w)T - \theta_2$	

Table 3-69 PLCD Native Mode Read Timing

Parameter		Equation	Notes
t_{avrd}	Address valid to address invalid	$2T + (n_{w(rd)} + r_w + h_{w(rd)})T - \xi_1$	
t_{csvrd}	Chip select active	$2T + (n_{w(rd)} + h_{w(rd)})T - \varepsilon_2$	For a non-bus-sized, non-sequential access
t_{rds}	Read data setup	$h_{w(rd)} - \chi_1$	
t_{rdh}	Read data hold	$h_{w(rd)} - \alpha_1$	Max value also limited by t_{csd}.
t_{ard}	Address valid to read active	$T + (r_w)T - \lambda_1$	take worst case of a normal read or a bus-sized read
t_{csrd}	Chip select active to read active	$T - \lambda_2$	
t_{rd}	Read active	$T + (n_{w(rd)})T - \gamma$	
t_{rda}	Read inactive to address invalid	$(h_{w(rd)})T - \eta_1$	
t_{rdcs}	Read inactive to chip select inactive	$(h_{w(rd)})T - \eta_2$	
t_{acs}	Address valid to chip select active	$(r_w)T - \theta_1$	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.



Minimal Wait States 16 bit write and read accesses from 16 bit memory

Figure 3-32 Intel LCD Interface NATIVE Write and Read Timing

Table 3-70 LCD Bus Sized Write Timing

Parameter		Equation	Notes
t_{bswr1}	Pulse width of first write active	$T + (n_{w(wr)})T - \zeta_2$	
t_{bswr2}	Pulse width of second write active	$T + n_{w(wr)}T - \zeta_3$	
t_{bswrh}	Pulse width of write inactive between bus-sized writes	$2T + h_wT - \kappa_1$	
t_{awr}	Address valid to first write active	$(r_w)\ T + T - \omega_3$	Take worst case among all address pins and a normal write case.
t_{abswr}	Address[1] valid to the second write active	$T - \pi_1$	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-70 LCD Bus Sized Write Timing (Continued)

Parameter		Equation	Notes
$t_{dsubswr1}$	Bus-sized write data setup for first write	$T + (n_{w(wr)})T - \upsilon_2$	
$t_{dsubswr2}$ or $t_{dsubswr4}$	Bus-sized write data setup for the last write	$2T + n_wT - \upsilon_3$	Take worst case among the 2nd, or 3rd, or 4th writes.
$t_{dhbswr1}$	Bus-sized write data hold (first write)	$T + (h_{w(wr)})T - \sigma_3$	Worst case among all write cycles except the last.
$t_{dhbswr2}$ or $t_{dhbswr4}$	Bus-sized write data hold (last write)	σ_2	Last cycle. Time when drivers turn-off and keepers turn-on.

Table 3-71 LCD Bus Sized Read Timing

Parameter		Equation	Notes
t_{ard}	Address valid to first read active	$T + (r_w)T - \lambda_3$	Take worst case among all address pins and a normal read case.
t_{absrd1}	Address valid to end of first read	$2T + (n_{w(rd)} + r_{w(rd)} + h_{w(rd)})T - \psi_1$	
t_{absrd2}	Address valid to end of the second read	$2T + (n_{w(rd)} + h_{w(rd)})T - \psi_2$	
t_{rds1}	Bus-sized read data setup for the first read	$h_{w(rd)} - \chi_2$	Take worst case among all reads and normal read.
t_{rdh1}	Bus-sized read data hold for the first read	$h_{w(rd)} - \alpha_2$	Take worst case among all reads and normal read. β_2 is also α_2 (max) Max value also limited by $t_{csd.}$
t_{rds2} t_{rds4}	Bus-sized read data setup for the second/fourth read	$h_{w(rd)} - \chi_3$	Take worst case among all reads and normal read
t_{rdh2} t_{rdh4}	Bus-sized read data hold for the second/fourth read	$h_{w(rd)} - \alpha_3$	Take worst case among all reads and normal read. β_3 is also α_3 (max). Max value also limited by $t_{csd.}$

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

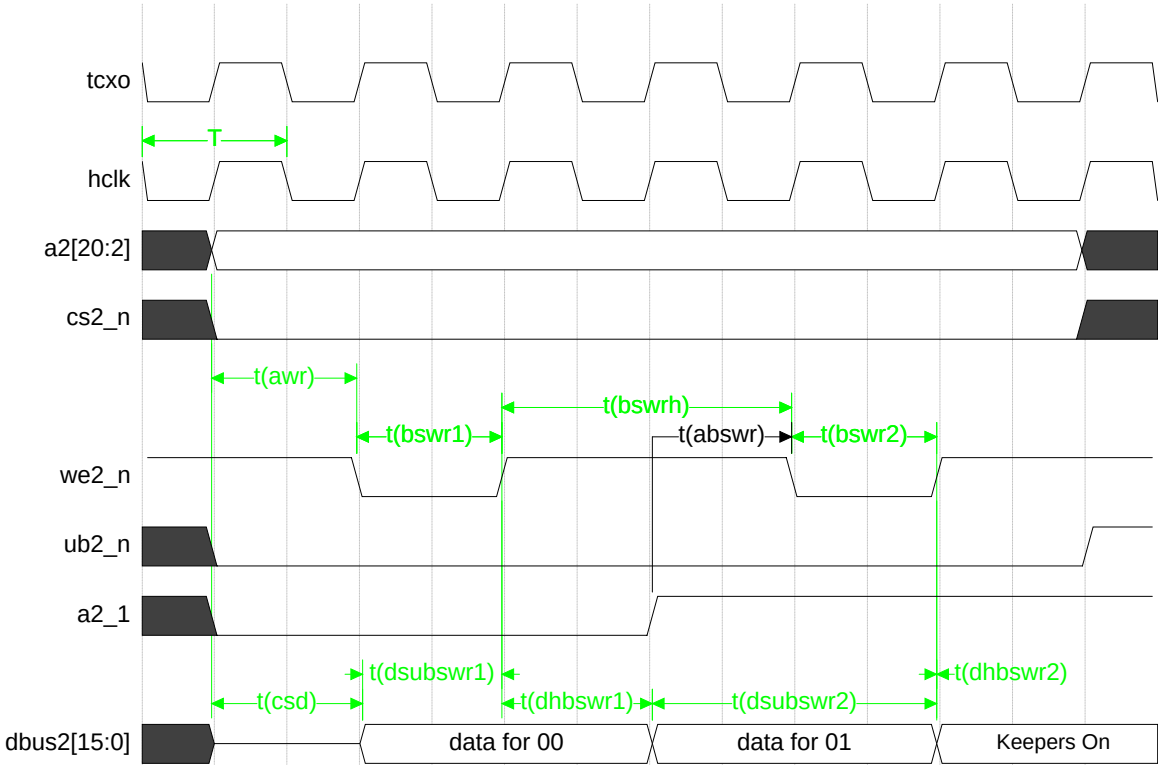


Figure 3-33 Intel LCD Interface Bus Sized 32x16 Write Timing

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

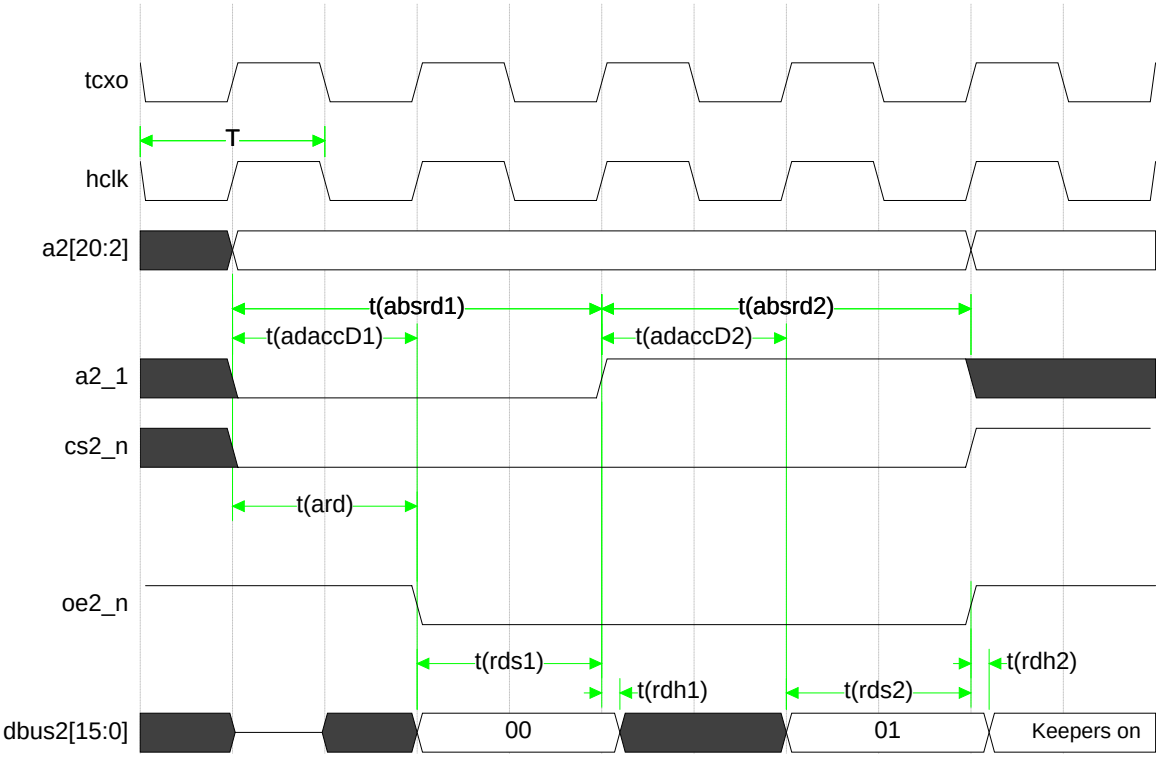


Figure 3-34 Intel LCD Interface Bus Sized 32x16 Read Timing

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

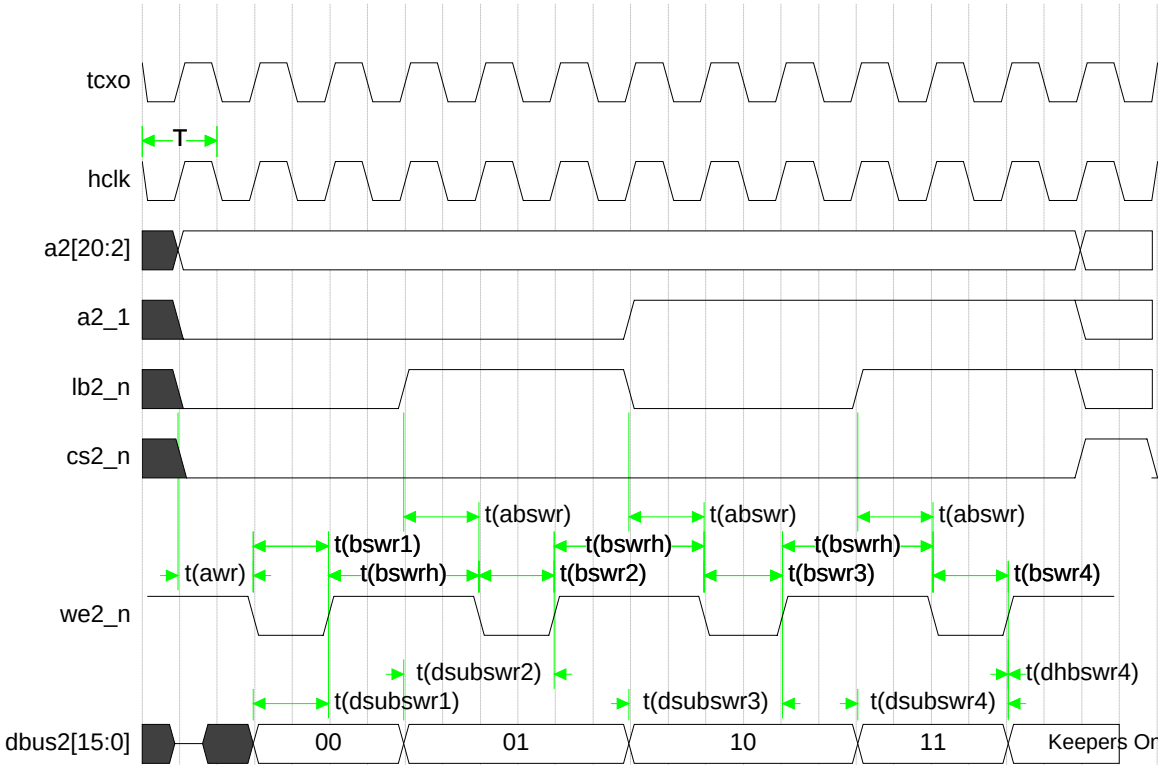


Figure 3-35 Intel LCD Interface Bus Sized 32x8 Write Timing

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

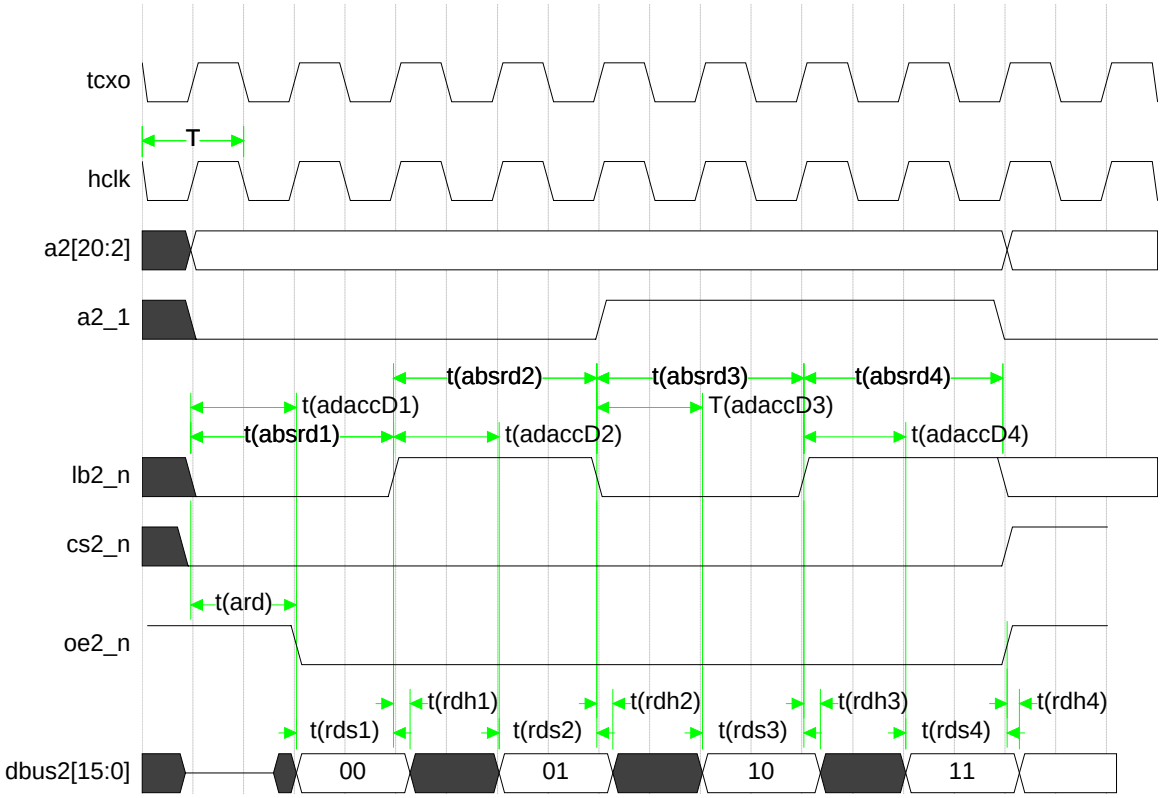


Figure 3-36 Intel LCD Interface Bus Sized 32x8 Read Timing

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

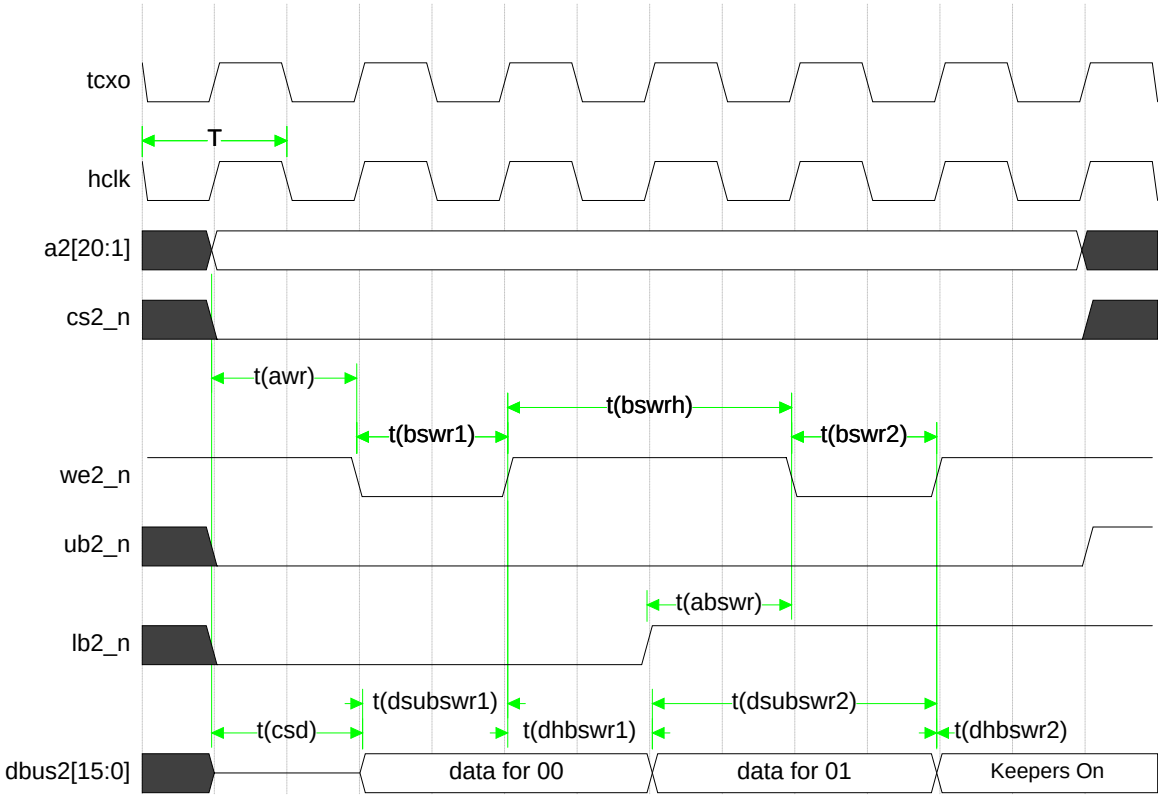


Figure 3-37 Intel LCD Interface Bus Sized 16x8 Write Timing

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

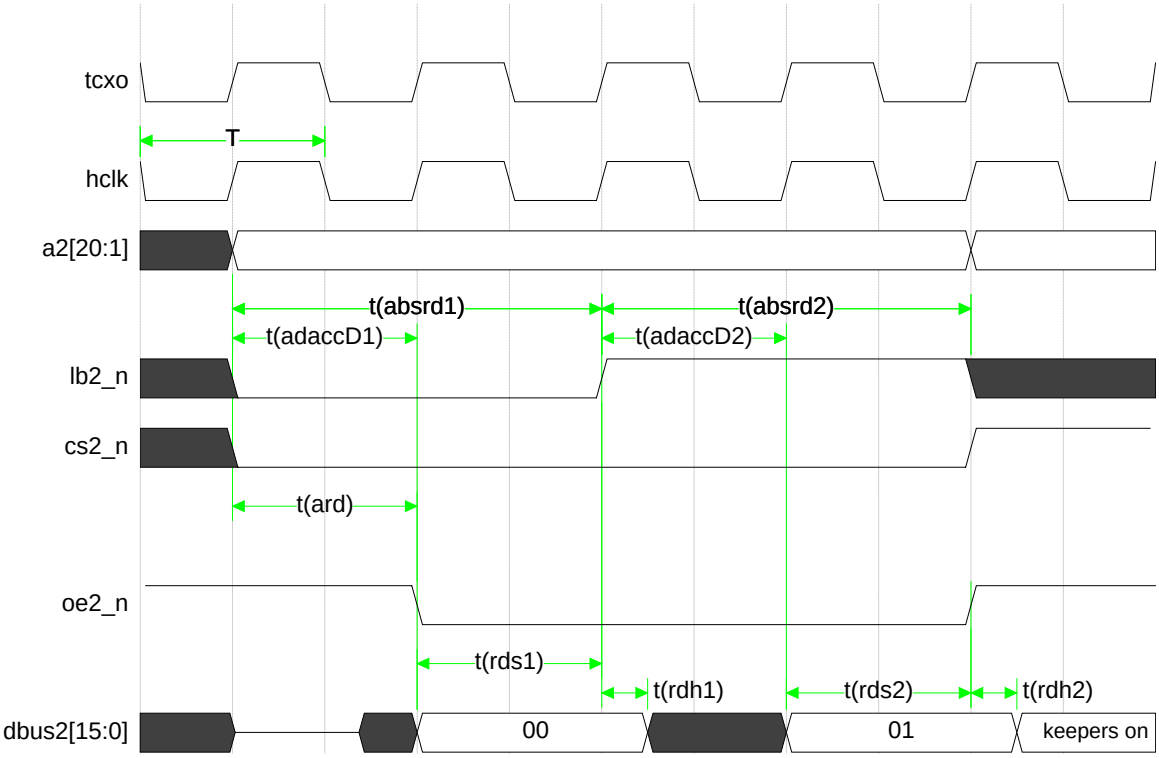


Figure 3-38 Intel LCD Interface Bus Sized 16x8 Read Timing

3.2.9.2.4 Motorola Interface Timing Parameters

Table 3-72 Motorola-Type Parallel LCD Timing Parameters Where V_{DD_P2} = 2.5 V

Symbol	Parameter	Min	Max
$\alpha 1$	TRDDH	-10.120	-5.890
$\alpha 2$	TRDDH1	-8.300	-4.790
$\alpha 3$	TRDDH2	-9.970	-5.820
$\alpha 3$	TRDDH4	-9.780	-5.800
$\chi 1$	TRDDS	6.170	10.500
$\chi 2$	TRDDS1	5.370	9.030
$\chi 3$	TRDDS2	6.150	10.660
$\varepsilon 1$	TCSVWR	-0.790	-0.020
$\varepsilon 2$	TCSVRD	-0.780	-0.120
$\gamma 1$	TEHRD	0.020	0.780

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-72 Motorola-Type Parallel LCD Timing Parameters Where V_{DD_P2} = 2.5 V

Symbol	Parameter	Min	Max
γ2	TBSEI	-0.630	0.020
η1	TERSRD	-0.960	3.250
η2	TECSRD	-1.130	-0.450
ι	TCSD	0.170	0.800
λ1	TRSERD	-1.090	0.470
λ2	TCSERD	-0.570	0.330
μ1	TERSWR	1.630	3.230
μ2	TECSWR	-1.040	-0.290
θ1	TCSRW	1.060	1.930
θ2	TWRCS	-2.300	-0.950
θ3	TRWE	-2.160	-0.930
θ4	TERW	0.530	1.530
σ1	TWRDH	0.240	1.090
σ2	TDHBSWR1	-0.840	-0.120
σ3	TDHBSWR2	0.260	1.590
σ3	TDHBSWR4	0.650	1.650
ω1	TRSEWR	-1.130	0.440
ω2	TCSEWR	-0.590	0.240
ξ1	TRSVWR	1.360	3.670
ξ2	TRSVRD	0.100	3.650
ζ1	TEHWR	0.120	0.800
ζ2	TBSEI	-0.540	-0.040

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-73 Motorola-Type Parallel LCD Timing Parameters Where V_{DD_P2} = 1.77V

Symbol	Parameter	Min	Max
α1	TRDDH	-9.960	-5.510
α2	TRDDH1	-8.000	-4.410
α3	TRDDH2	-9.690	-5.400
α3	TRDDH4	-9.890	-5.390
χ1	TRDDS	5.870	10.540
χ2	TRDDS1	4.960	9.020
χ3	TRDDS2	5.780	10.530
ε1	TCSVWR	-0.930	-0.050
ε2	TCSVRD	-0.950	-0.140
γ1	TEHRD	0.040	0.830
γ2	TBSEI	-0.750	0.000
η1	TERSRD	-1.080	3.260
η2	TECSRD	-1.250	-0.460
ι	TCSD	0.230	0.900
λ1	TRSERD	-1.150	0.380
λ2	TCSERD	-0.630	0.320
μ1	TERSWR	1.520	3.260
μ2	TECSWR	-1.180	-0.380
θ1	TCSRW	1.050	2.200
θ2	TWRCS	-2.570	-0.990
θ3	TRWE	-2.350	-0.900
θ4	TERW	0.510	1.670
σ1	TWRDH	0.060	1.060
σ2	TDHBSWR1	-0.960	-0.150
σ3	TDHBSWR2	0.130	1.310
σ3	TDHBSWR4	0.420	1.330
ω1	TRSEWR	-1.150	0.360
ω2	TCSEWR	-0.630	0.270
ξ1	TRSVWR	1.270	3.650

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-73 Motorola-Type Parallel LCD Timing Parameters Where V_{DD_P2} = 1.77V

Symbol	Parameter	Min	Max
$\xi 2$	TRSVRD	0.180	3.530
$\zeta 1$	TEHWR	0.120	0.840
$\zeta 2$	TBSEI	-0.630	-0.050

3.2.9.2.5 Motorola LCD Timing Diagrams and Equations

Table 3-74 LCD Interface NATIVE Read Timing, Motorola Type

Parameter		Equation	Notes
t_{csvrd}	Chip select active	$(n_{w(rd)} + 1)T - \varepsilon_2$	
t_{rsvrd}	Address/RS valid	$(n_{w(rd)} + 1 + r_w)T - \xi_2$	
t_{rdds}	Read data setup	$-\{(1 + n_{w(rd)} - s_{w(rd)} - p_{w(rd)}) * T - \chi_1\}$	Read data setup and hold time is measured with respect to the lcd_e falling edge, despite the fact that ARM may be latching the data later in some cases.
t_{rddh}	Read data hold	$(1 + n_{w(rd)} - s_{w(rd)} - p_{w(rd)}) * T - \alpha_1$	Read data setup and hold time is measured with respect to the lcd_e falling edge, despite the fact that ARM may be latching the data later in some cases.
t_{cserd}	Chip select active to LCD_E active	$(s_{w(rd)})T - \lambda_2$	
t_{rserd}	RS active to LCD_E active	$(s_{w(rd)} + r_w)T - \lambda_1$	
t_{ehrd}	LCD_E active	$(p_{w(rd)})T - \gamma_1$	
t_{ecsrd}	LCD_E inactive to chip select inactive	$(1 + n_{w(rd)} - s_{w(rd)} - p_{w(rd)})T - \eta_2$	
t_{ersrd}	LCD_E inactive to address/RS invalid	$(1 + n_{w(rd)} - s_{w(rd)} - p_{w(rd)})T - \eta_1$	
t_{rwcsrd}	RW/ high to chip select active	T	Guarantee by design. No measurement.
t_{csrwrd}	Chip select inactive to RW/ low	T	Guarantee by design. No measurement.
t_{eacc}	LCD_E access time	$p_{w(rd)} * T$	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

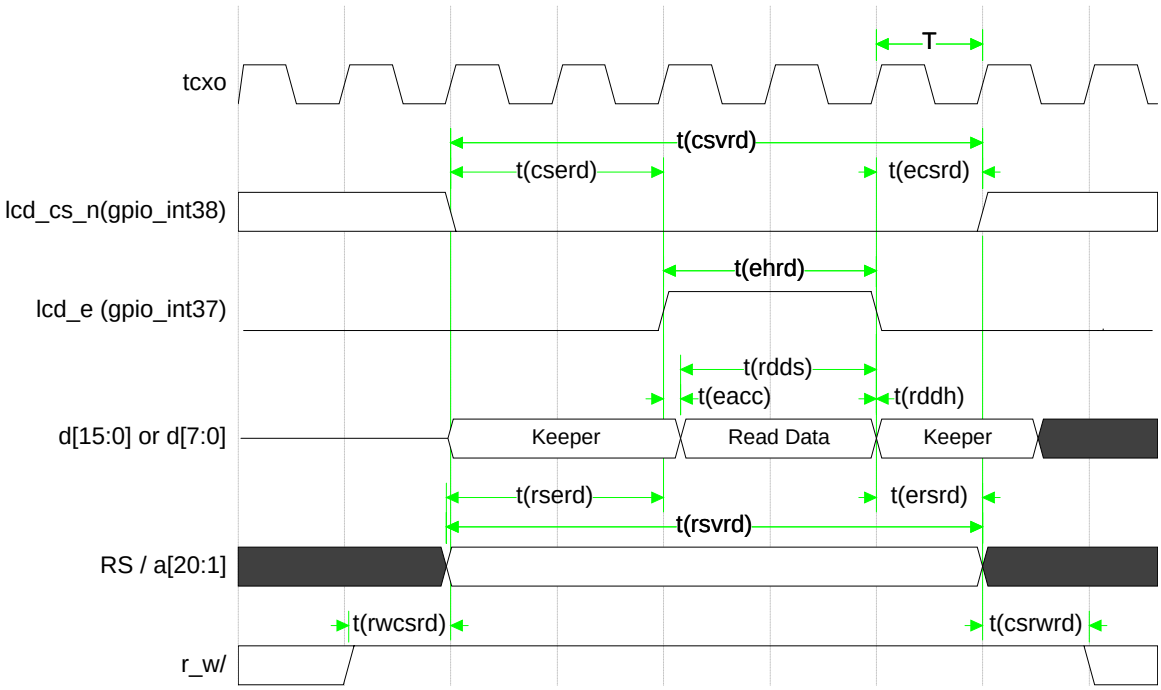


Figure 3-39 Motorola LCD Interface NATIVE Read Timing

Table 3-75 PLCD Interface Write Timing, Motorola Type

Parameter		Equation	Notes
t_{csvwr}	Chip select active for write	$(n_{w(wr)} + 1)T - \varepsilon_1$	
t_{rsvwr}	Chip select active for write	$(n_{w(wr)} + 1 + r_w)T - \xi_1$	
t_{wrdh}	Write data hold	$(n_{w(wr)} - s_{w(wr)} - p_{w(wr)}) * T - \sigma_1$	By design, keepers will hold data, spec T in user manual.
t_{csewr}	Chip select active to LCD_E active	$(s_{w(wr)})T - \omega_2$	
t_{rsewr}	RS/Address valid to LCD_E active	$(s_{w(wr)} + r_w)T - \omega_1$	
t_{ehwr}	LCD_E active	$(p_{w(wr)})T - \zeta_1$	
t_{csd}	Chip select active to data valid	$T - \iota$	
t_{ecswr}	LCD_E inactive to chip select inactive	$(1 + n_{w(wr)} - s_{w(wr)} - p_{w(wr)})T - \mu_2$	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-75 PLCD Interface Write Timing, Motorola Type

Parameter		Equation	Notes
t_{erswr}	LCD_E inactive to RS/address invalid	$(1 + n_{w(wr)} - s_{w(wr)} - p_{w(wr)})T - \mu_1$	
t_{csrw}	Chip select active to RW/ low	$T - \theta_1$	
t_{rwe}	RW/ low to LCD_E active	$(s_{w(wr)} - 1)\ T - \theta_3$	
t_{erw}	LCD_E inactive to RW/ high	$(n_{w(wr)} - s_{w(wr)} - p_{w(wr)})\ T - \theta_4$	
t_{rwcs}	RW/ high to chip select inactive	$T - \theta_2$	

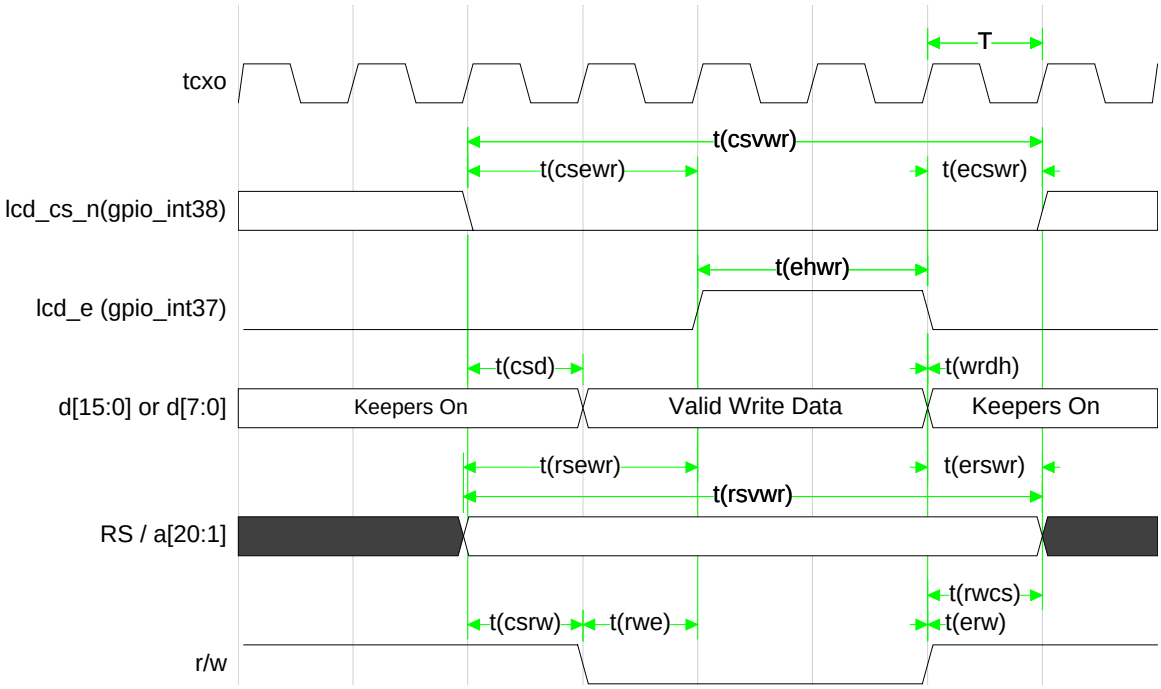


Figure 3-40 Motorola LCD Interface NATIVE Write Timing

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-76 LCD Interface BUS SIZED Read Timing, Motorola Type

Parameter		Equation	Notes
t_{csvrd}	Chip select active	$BP * (n_{w(rd)} + 1)T - \varepsilon_2$	BP = 2 for 32x16, 16x8 BP = 4 for 32x8
t_{rsvrd}	Address/RS valid	$BP * (n_{w(rd)} + 1 + r_w)T - \xi_2$	BP = 2 for 32x16, 16x8 BP = 4 for 32x8
t_{rdds1} (32x8 and 32x16) or t_{rdds2} (32x8) or t_{rdds3} (32x8)	Read data setup. All Bus Size phases but last.	$-\{(1 + n_{w(rd)} - s_{w(rd)} - p_{w(rd)}) * T - \chi_2\}$	Read data setup and hold time is measured with respect to the lcd_e falling edge, despite the fact that ARM may be latching the data later in some cases. Measure worst case of all.
t_{rddh1} (32x8 and 32x16) or t_{rddh2} (32x8) or t_{rddh3} (32x8)	Read data hold. All Bus Size phases but last.	$(1 + n_{w(rd)} - s_{w(rd)} - p_{w(rd)}) * T - \alpha_2$	Read data setup and hold time is measured with respect to the lcd_e falling edge, despite the fact that ARM may be latching the data later in some cases. Measure worst case of all.
t_{rdds2} (32x16) or t_{rdds4} (32x8)	Read data setup. Last Bus Size phase.	$-\{(1 + n_{w(rd)} - s_{w(rd)} - p_{w(rd)}) * T - \chi_3\}$	Read data setup and hold time is measured with respect to the lcd_e falling edge, despite the fact that ARM may be latching the data later in some cases. Measure worst case of all.
t_{rddh2} (32x16) or t_{rddh4} (32x8)	Read data hold. Last Bus Size phase	$(1 + n_{w(rd)} - s_{w(rd)} - p_{w(rd)}) * T - \alpha_3$	Read data setup and hold time is measured with respect to the lcd_e falling edge, despite the fact that ARM may be latching the data later in some cases. Measure worst case of all.
t_{cserd}	Chip select active to LCD_E active	$(s_{w(rd)})T - \lambda_2$	
t_{rserd}	RS active to LCD_E active	$(s_{w(rd)} + r_w)T - \lambda_1$	
t_{ehrd}	LCD_E active	$(p_{w(rd)})T - \gamma_1$	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-76 LCD Interface BUS SIZED Read Timing, Motorola Type

Parameter		Equation	Notes
t_{bsei}	LCD_E inactive between bus sized accesses	$(1 + n_{w(rd)} - p_{w(rd)}) * T - \gamma_2$	
t_{ecsrd}	LCD_E inactive to chip select inactive	$(1 + n_{w(rd)} - s_{w(rd)} - p_{w(rd)})T - \eta_2$	
t_{ersrd}	LCD_E inactive to address/RS invalid	$(1 + n_{w(rd)} - s_{w(rd)} - p_{w(rd)})T - \eta_1$	
t_{rwcs}	RW/ high to chip select active	T	Guarantee by design. No measurement.
t_{csrw}	Chip select inactive to RW/ low	T	Guarantee by design. No measurement.
t_{eacc}	LCD_E access time	$p_{w(rd)} * T$	

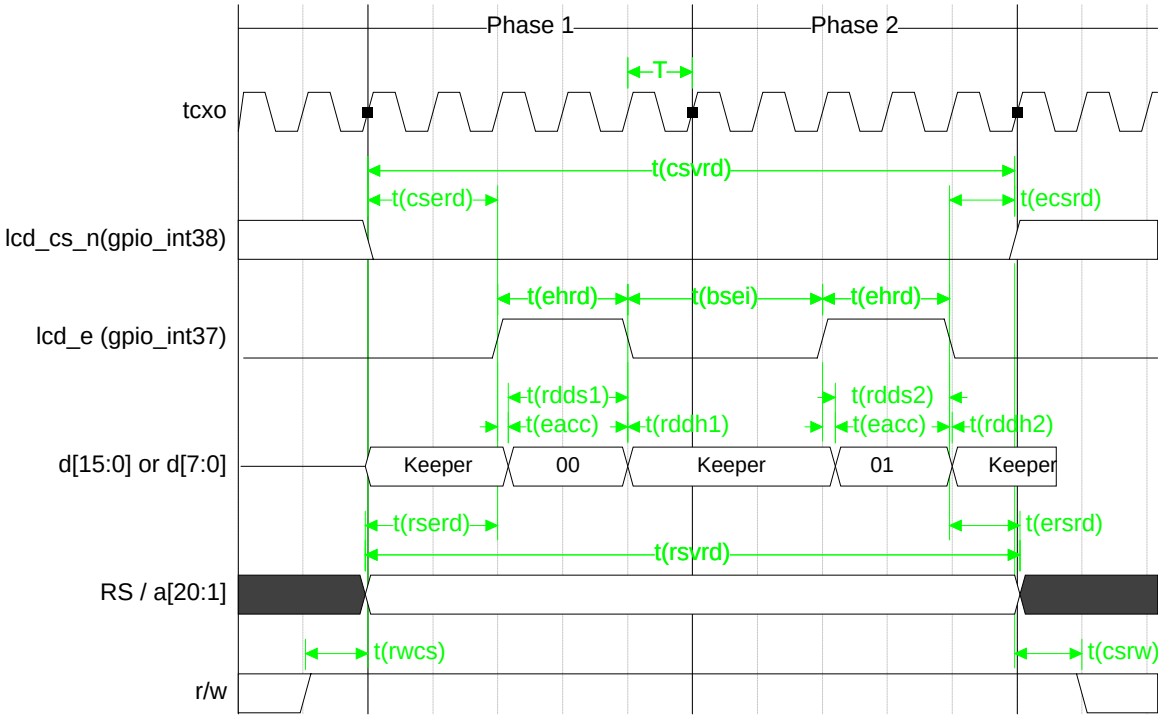


Figure 3-41 Motorola LCD Interface Bus Sized 32x16 or 16x8 Read Timing

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

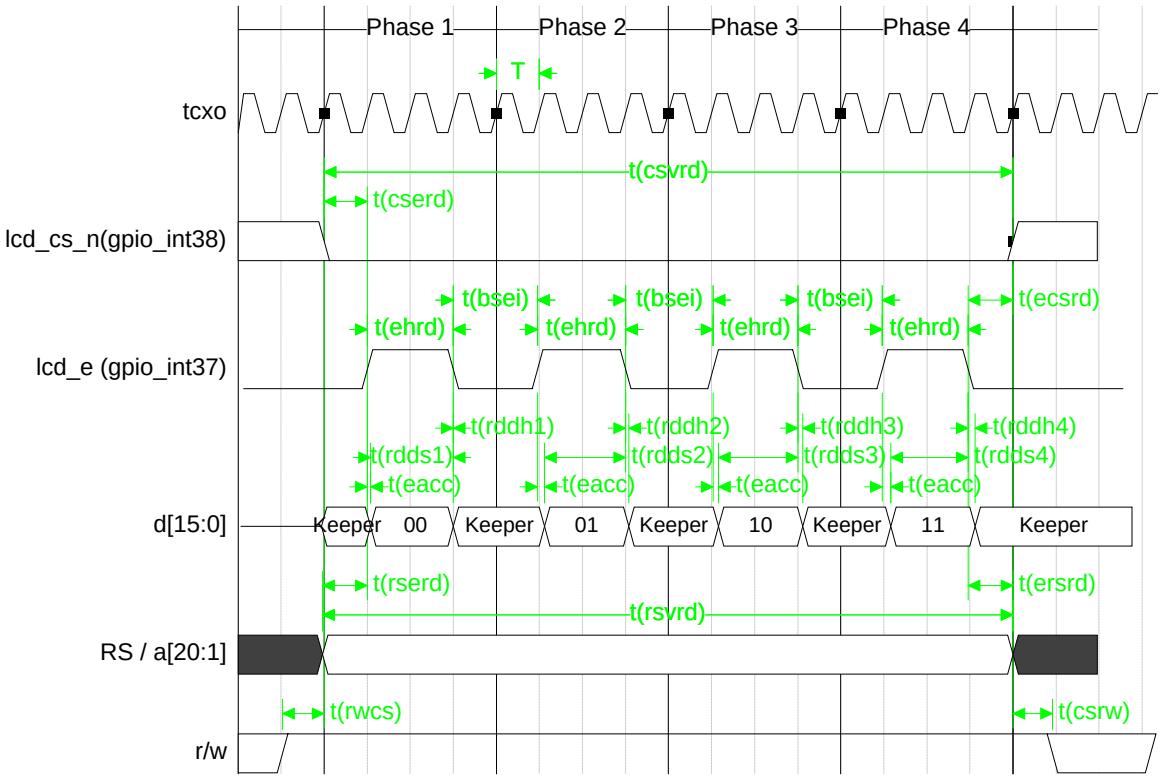


Figure 3-42 Motorola LCD Interface Bus Sized 32x8 Read Timing

Table 3-77 LCD Interface BUS SIZED Write Timing, Motorola Type

Parameter		Equation	Notes
t_{csvwr}	Chip select active for write	BP * ($n_{w(wr)}$ + 1)T - ε_1	BP = 2 for 32x16, 16x8 BP = 4 for 32x8
t_{rsvwr}	RS/address valid	BP * ($n_{w(wr)}$ + 1 + r_w)T - ξ_1	BP = 2 for 32x16, 16x8 BP = 4 for 32x8
$t_{dhbswr1}$ (32x16 and 32x8) or $t_{dhbswr2}$ (32x8) or $t_{dhbswr3}$ (32x8)	Write data hold. All Bus Size phases but last.	(1 + $n_{w(wr)}$ - $s_{w(wr)}$ - $p_{w(wr)}$) * T + σ_2	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-77 LCD Interface BUS SIZED Write Timing, Motorola Type (Continued)

Parameter		Equation	Notes
$t_{dhbswr2}$ (32x16) or $t_{dhbswr4}$ (32x8)	Write data hold. Last Bus phase.	$(n_{w(wr)} - s_{w(wr)} - p_{w(wr)}) * T + \sigma_3$	by design, keepers will hold data, spec T in user manual
t_{csewr}	Chip select active to LCD_E active	$(s_{w(wr)})T - \omega_2$	
t_{rsewr}	RS/Address valid to LCD_E active	$(s_{w(wr)} + r_w)T - \omega_1$	
t_{ehwr}	LCD_E active	$(p_{w(wr)})T - \zeta_1$	
t_{bsei}	LCD_E inactive between bus sized accesses	$(1 + n_{w(rd)} - p_{w(rd)}) * T - \zeta_2$	
t_{csd}	Chip select active to data valid	$T - \iota$	
t_{ecswr}	LCD_E inactive to chip select inactive	$(1 + n_{w(wr)} - s_{w(wr)} - p_{w(wr)})T - \mu_2$	
t_{erswr}	LCD_E inactive to RS/address invalid	$(1 + n_{w(wr)} - s_{w(wr)} - p_{w(wr)})T - \mu_1$	
t_{csrw}	Chip select active to RW/ low	$T - \theta_1$	
t_{rwe}	RW/ low to LCD_E active	$(s_{w(wr)} - 1)T - \theta_3$	
t_{erw}	LCD_E inactive to RW/ high	$(n_{w(wr)} - s_{w(wr)} - p_{w(wr)})T - \theta_4$	
t_{rwcs}	RW/ high to chip select inactive	$T - \theta_2$	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

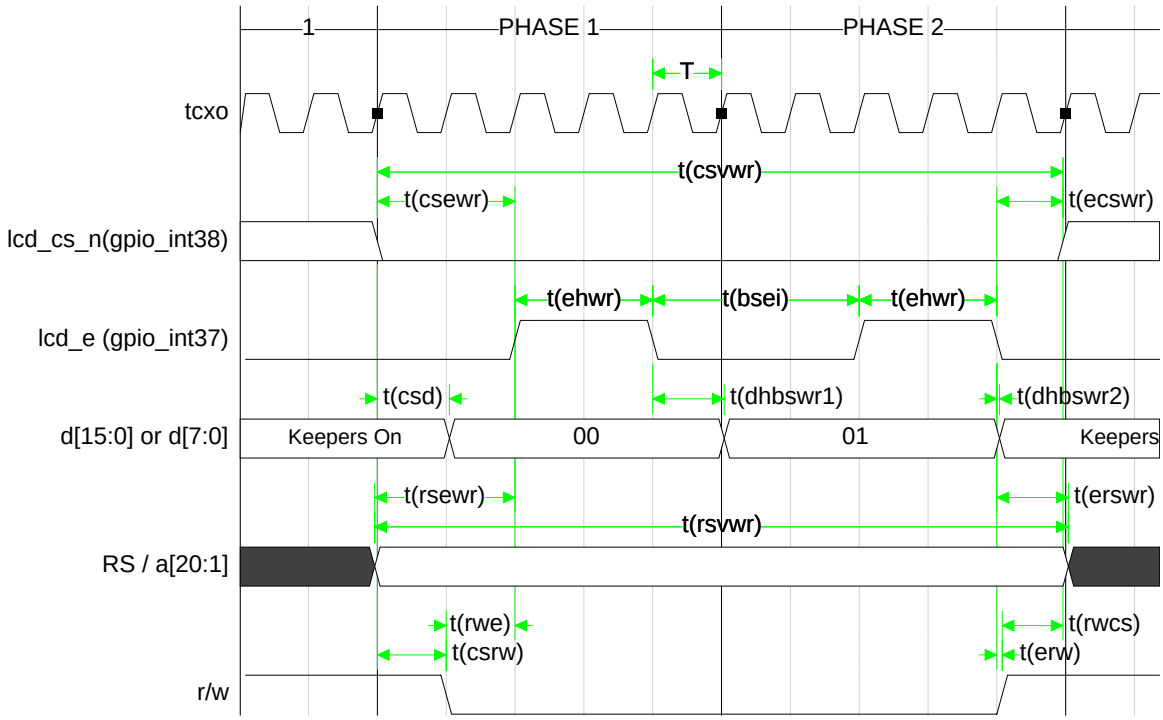


Figure 3-43 Motorola LCD Interface Bus Sized 32x16 or 16x8 Write Timing

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

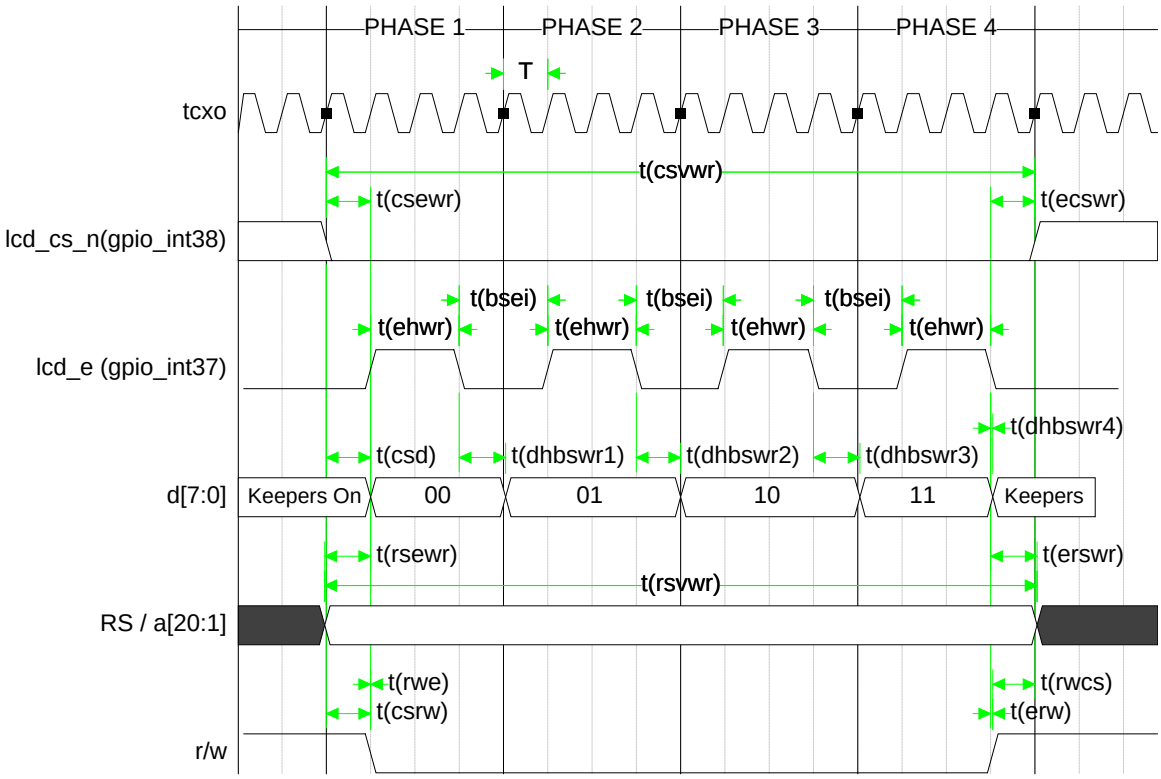


Figure 3-44 Motorola LCD Interface Bus Sized 32x8 Write Timing

3.2.9.3 External NAND Interface

The NAND flash memory is connected to rom2_cs_n. The NAND flash controller shares the external bus interface 2 (EBI2) bus with the EBI2 external memory controller.

3.2.9.3.1 Wait State Definition

Table 3-78 NAND Wait States Description

Symbol	Description	Notes
T	period of internal HCLK	affected by register MICRO_CLK_DIV; MICRO_CLOCK_SOURCE 0/1 default setting is TCXO div 1 characterization setting: T= 40ns, TCXO div 1
$n_{w(wr)}$	number of wait states used for write	$n_{w(wr)}$=NAND_FLASH_CFG[4:2]

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-78 NAND Wait States Description

Symbol	Description	Notes
$n_{w(rd)}$	number of wait states used for read	$n_{w(rd)}$=NAND_FLASH_CFG[4:2]
r_n	number of recovery wait states from previous memory (when the previous memory access was a read)	r_n= 2 if Recovery Value < 3 Recovery Value if Recovery Value ≥ 3 If a NAND access comes immediately after a read to an EBI2 chip select controlled by the EBI2 external memory controller (RAM2_CS_N, GP2_CS_N0, GP2_CS_N1, or LCD_CS_N), the NAND controller may need to provide enough time for that memory to get off the EBI2 bus (stop driving the bus). That time is referred to as “recovery time.” Each configuration register for the chip selects mentioned above has a RECOVERY wait state field that indicates to the next access (the NAND access in this context) the required amount of recovery time. Thus, Recovery Value can be one of the following register values, providing the NAND access immediately follows an XMEM read access: RAM2_CFG0[27:24] GP0_CFG0[27:24] GP1_CFG[27:24] LCD_CFG0[27:24]

3.2.9.3.2 NAND Timing Parameters

Table 3-79 NAND Access Timing Parameters Where V_{DD_P2} = 2.5V

Symbol	Parameter	Min	Max
$\alpha 1$	TIDRDH	-5.550	-3.480
$\alpha 1$	TRDH	-5.520	-3.470
$\alpha 2$	TRDH	-7.380	-4.580
$\chi 1$	TIDRDS	3.710	6.290
$\chi 1$	TRDS	3.900	6.400
$\chi 2$	TRDS	5.190	8.360
$\varepsilon 1$	TCSVWR	0.080	0.860
$\varepsilon 2$	TCSVRD	0.190	0.600
$\varepsilon 3$	TCSVWR	-0.010	0.670
$\varepsilon 4$	TCSVRD	0.090	0.570

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-79 NAND Access Timing Parameters Where V_{DD_P2} = 2.5V (Continued)

Symbol	Parameter	Min	Max
ε5	TIDCSVRD	0.040	0.560
η2	TIDRDCS	1.200	2.450
η2	TRDCS	1.240	2.250
η3	TRDCS	1.250	2.270
γ1	TRD	-0.320	-0.020
γ2	TRD	-0.340	-0.030
γ3	TRC	-0.200	0.090
γ4	TREH	-0.040	0.260
γ5	TIDRD	-0.400	-0.090
γ6	TIDRD	-0.190	0.040
γ7	TIDREH	-0.100	0.260
ι1	TCSD	0.210	0.600
ι2	TCSD	-1.910	-0.760
λ2	TCSRD	1.490	2.860
λ3	TCSRD	1.510	3.020
λ4	TIDCSRD	1.490	2.950
μ2	TWRCS	-1.430	0.120
μ3	TWRCS	-1.520	0.060
ω2	TCSWR	0.690	2.300
ω3	TCSWR	0.660	2.650
π1	TWHR	0.450	2.130
π3	TAR1	0.700	1.390
ψ1	TALESETUPAW	-3.470	-1.570
ψ2	TCLESETUPCW	-2.950	-1.400
ψ3	TCLS	0.270	2.250
ψ4	TALS	0.840	2.130
ψ4	TCLH	-1.860	1.190
ψ5	TALS	-0.530	2.640
ψ6	TALH	-1.210	0.770

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-79 NAND Access Timing Parameters Where V_{DD_P2} = 2.5V (Continued)

Symbol	Parameter	Min	Max
σ1	TDHWR	0.270	0.920
σ2	TDHWR	-1.310	0.760
υ1	TDSUWR	0.480	2.310
υ2	TDSUWR	0.140	3.040
ζ1	TWR	-1.000	-0.150
ζ2	TWR	-1.030	-0.110
ζ3	TWC	-0.320	0.190
ζ4	TWH	-0.050	0.890

Table 3-80 NAND Access Timing Parameters Where V_{DD_P2} = 1.77V

Symbol	Parameter	Min	Max
α1	TIDRDH	-6.360	-3.780
α1	TRDH	-6.220	-3.730
α2	TRDH	-7.770	-4.760
χ1	TIDRDS	3.940	6.760
χ1	TRDS	4.210	6.840
χ2	TRDS	5.410	8.870
ε1	TCSVWR	0.120	0.880
ε2	TCSVRD	0.190	0.700
ε3	TCSVWR	-0.010	0.750
ε4	TCSVRD	0.070	0.580
ε5	TIDCSVRD	0.040	0.720
η2	TIDRDCS	1.190	2.420
η2	TRDCS	1.210	2.240
η3	TRDCS	1.240	2.280
γ1	TRD	-0.400	-0.020
γ2	TRD	-0.410	-0.040
γ3	TRC	-0.220	0.080
γ4	TREH	-0.030	0.330

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-80 NAND Access Timing Parameters Where V_{DD_P2} = 1.77V (Continued)

Symbol	Parameter	Min	Max
γ5	TIDRD	-0.450	-0.090
γ6	TIDRD	-0.260	-0.010
γ7	TIDREH	-0.060	0.370
ι1	TCSD	0.290	0.980
ι2	TCSD	-2.280	-0.810
λ2	TCSRD	1.510	2.930
λ3	TCSRD	1.520	2.970
λ4	TIDCSRD	1.510	2.930
μ2	TWRCS	-1.440	0.100
μ3	TWRCS	-1.560	0.030
ω2	TCSWR	0.680	2.240
ω3	TCSWR	0.640	2.670
π1	TWHR	0.400	2.120
π3	TAR1	0.700	1.360
ψ1	TALESETUPAW	-3.290	-1.470
ψ2	TCLESETUPCW	-2.790	-1.390
ψ3	TCLS	0.230	2.170
ψ4	TALS	0.790	2.080
ψ4	TCLH	-1.880	1.210
ψ5	TALS	-0.500	2.620
ψ6	TALH	-1.200	0.750
σ1	TDHWR	0.250	0.940
σ2	TDHWR	-1.300	0.780
υ1	TDSUWR	0.670	2.600
υ2	TDSUWR	0.280	3.390
ζ1	TWR	-1.050	-0.150
ζ2	TWR	-1.010	-0.130
ζ3	TWC	-0.300	0.130
ζ4	TWH	-0.040	0.860

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.9.3.3 NAND Access Timing for Single Accesses Interleaved with EBI2 XMEM Controller Accesses

The MSM 6100 EBI2 arbiter allows NAND memory access only when the sole request present is that of the NAND controller. In other words, the NAND controller has the lowest priority of all requesters to EBI2, and it only gets the grant when all other requesters are idle. Under that condition, it is very possible to have single NAND accesses preceded and followed by EBI2 external memory interface accesses.

The characterization timings shown on the following sections referred to that specific condition.

3.2.9.3.4 Command Write Timing (0 and 1 Wait States)

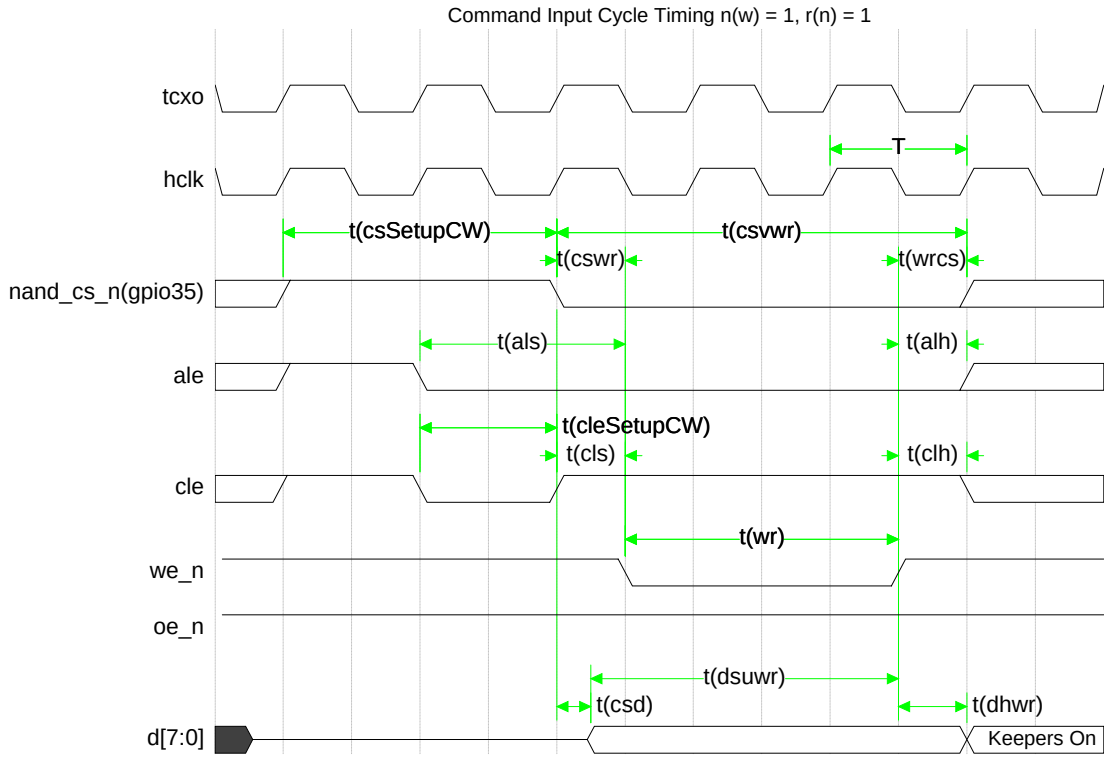


Figure 3-45 NAND Command Write Timing with 3 Access Cycles (Minimum Access Cycles + 1 Access WS) and 1 Recovery Cycle

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-81 NAND Command Write Timing (0 and 1 Access Wait States)

Parameter		Equation	Notes
t_{csvwr}	Chip select active	$2T + (n_{w(wr)})T - \varepsilon_1$	
t_{dsuwr}	Write data setup	$1.5T + (n_{w(wr)})T - \upsilon_1$	
t_{dhwr}	Write data hold	$0.5T - \sigma_1$	
t_{cswr}	Chip select active to write active	$0.5T - \omega_2$	
t_{wr}	Write active	$(1 + n_{w(wr)})T - \zeta_1$	
t_{csd}	Chip select active to data valid	i_1	
t_{wrcs}	Write inactive to chip select inactive	$0.5T - \mu_2$	
$t_{csSetupCW}$	End of EBI2 XMEM controller access to NAND chip select active.	$(r_n)T$	Minimum of 2 cycles guaranteed by design.
$t_{cleSetupCW}$	Start of NAND controller access to NAND control latch enable active.	$T - \psi_2$	
t_{cls}	Command latch enable setup time to write active.	$0.5T - \psi_3$	
t_{als}	Address latch enable setup time to write active.	$1.5*T - \psi_5$	Guaranteed by design. No measurement done.
t_{clh}	Command latch enable hold time from write inactive.	$0.5T - \psi_4$	
t_{alh}	Address latch enable hold time from write inactive.	$0.5T - \psi_6$	

NOTE ale (lb_n) and cle (ub_n) are LOW most of the time for accesses done by the EBI2 XMEM controller. However, if the XMEM controller access is a BYTE access to a 16-bit memory, then either lb_n or ub_n will be HIGH during that access. If that BYTE access precedes a NAND memory access, then either ale_n or cle_n setup, from the time the EBI2 XMEM access finishes to the time the NAND access, we_n asserts needs to be measured as required. Similarly, if the NAND access is followed by an EBI2 XMEM controller access that requires either recovery time or cs_setup wait states, then both ale (lb_n) and cle (ub_n) signals will be HIGH right after the end of the NAND access. In that case, the hold time from the NAND access we_n signal deasserting to the ale and cle signals rising needs to be measured.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.9.3.5 Command Write Timing (2 and 3 Wait States)

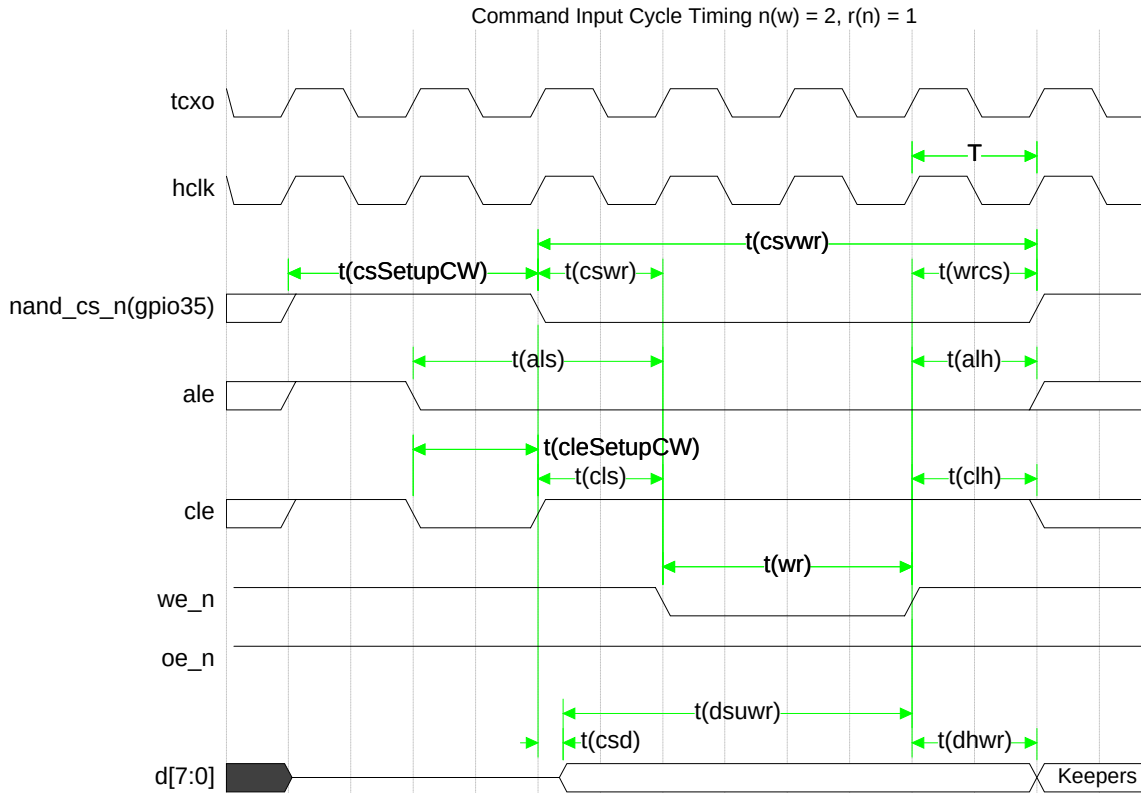


Figure 3-46 NAND Command Write Timing with 4 Access Cycles (Minimum Access Cycles + 2 Access WS) and 1 Recovery Cycle

Table 3-82 NAND Command Write Timing (2 and 3 Access Wait States)

Parameter		Equation	Notes
t_{csvwr}	Chip select active	$2T + (n_{w(wr)})T - \varepsilon_1$	
t_{dsuwr}	Write data setup	$T + (n_{w(wr)})T - \upsilon_1$	
t_{dhwr}	Write data hold	$T - \sigma_1$	
t_{cswr}	Chip select active to write active	$T - \omega_2$	
t_{wr}	Write active	$(n_{w(wr)})T - \zeta_1$	
t_{csd}	Chip select active to data valid	i_1	
t_{wrcs}	Write inactive to chip select inactive	$T - \mu_2$	
$t_{csSetupCW}$	End of EBI2 XMEM controller access to NAND chip select active.	(r_n)	Minimum of 2 cycles guaranteed by design.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-82 NAND Command Write Timing (2 and 3 Access Wait States) (Continued)

Parameter		Equation	Notes
$t_{cleSetupCW}$	Start of NAND controller access to NAND control latch enable active.	$T - \psi_2$	
t_{cls}	Command latch enable setup time to write active.	$T - \psi_3$	
t_{als}	Address latch enable setup time to write active.	$2T - \psi_5$	Guaranteed by Design. No measurement done
t_{clh}	Command latch enable hold time from write inactive.	$T - \psi_4$	
t_{alh}	Address latch enable hold time from write inactive.	$T - \psi_6$	

NOTE ale (lb_n) and cle (ub_n) are LOW most of the time for accesses done by the EBI2 XMEM controller. However, if the XMEM controller access is a BYTE access to a 16-bit memory, then either lb_n or ub_n will be HIGH during that access. If that BYTE access precedes a NAND memory access, then either ale_n or cle_n setup, from the time the EBI2 XMEM access finishes to the time the NAND access we_n asserts, needs to be measured as required. Similarly, if the NAND access is followed by an EBI2 XMEM controller access that requires either recovery time or cs_setup wait states, then both ale (lb_n) and cle (ub_n) signals will be HIGH right after the end of the NAND access. In that case, the hold time from the NAND access we_n signal deasserting to the ale and cle signals rising needs to be measured.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.9.3.6 Address Write Timing (0 and 1 Wait States)

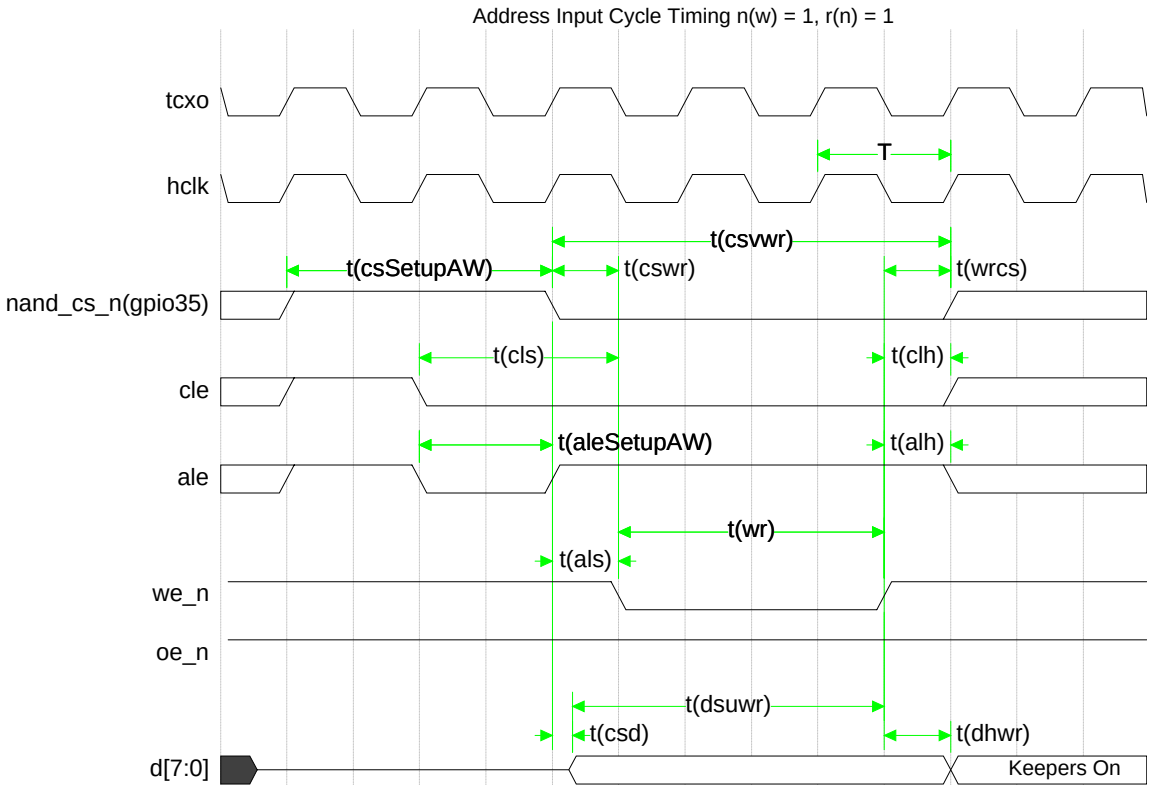


Figure 3-47 NAND Address Write Timing with 3 Access Cycles (Minimum Access Cycles + 1 Access WS) and 1 Recovery Cycle

Table 3-83 NAND Address Write Timing (0 and 1 Wait States)

Parameter		Equation	Notes
t_{csvwr}	Chip select active	$2T + (n_{w(wr)})T - \varepsilon_1$	
t_{dsuwr}	Write data setup	$1.5T + (n_{w(wr)})T - \upsilon_1$	
t_{dhwr}	Write data hold	$0.5T - \sigma_1$	
t_{cswr}	Chip select active to write active	$0.5T - \omega_2$	
t_{wr}	Write active	$(1 + n_{w(wr)})T - \zeta_1$	
t_{csd}	Chip select active to data valid	i_1	
t_{wrcs}	Write inactive to chip select inactive	$0.5T - \mu_2$	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-83 NAND Address Write Timing (0 and 1 Wait States) (Continued)

Parameter		Equation	Notes
$t_{csSetupAW}$	End of EBI2 XMEM controller access to NAND chip select active.	$(r_n)T$	Minimum of 2 cycles guaranteed by design.
$t_{aleSetupAW}$	Start of NAND controller access to NAND address latch enable active.	$T - y_1$	
t_{als}	Address latch enable setup time to write active.	$0.5T - \psi_5$	
t_{cls}	Command latch enable setup time to write active.	$1.5*T - \psi_3$	Guaranteed by design. No measurement done.
t_{alh}	Address latch enable hold time from write inactive.	$0.5T - \psi_6$	
t_{clh}	Command latch enable hold time from write inactive.	$0.5T - \psi_4$	

NOTE ale (lb_n) and cle (ub_n) are LOW most of the time for accesses done by the EBI2 XMEM controller. However, if the XMEM controller access is a BYTE access to a 16-bit memory, then either lb_n or ub_n will be HIGH during that access. If that BYTE access precedes a NAND memory access, then either ale_n or cle_n setup, from the time the EBI2 XMEM access finishes to the time the NAND access we_n asserts, needs to be measured as required. Similarly, if the NAND access is followed by an EBI2 XMEM controller access that requires either recovery time or cs_setup wait states, then both ale (lb_n) and cle (ub_n) signals will be HIGH right after the end of the NAND access. In that case, the hold time from the NAND access we_n signal deasserting to the ale and cle signals rising needs to be measured.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.9.3.7 Address Write Timing (2 and 3 Wait States)

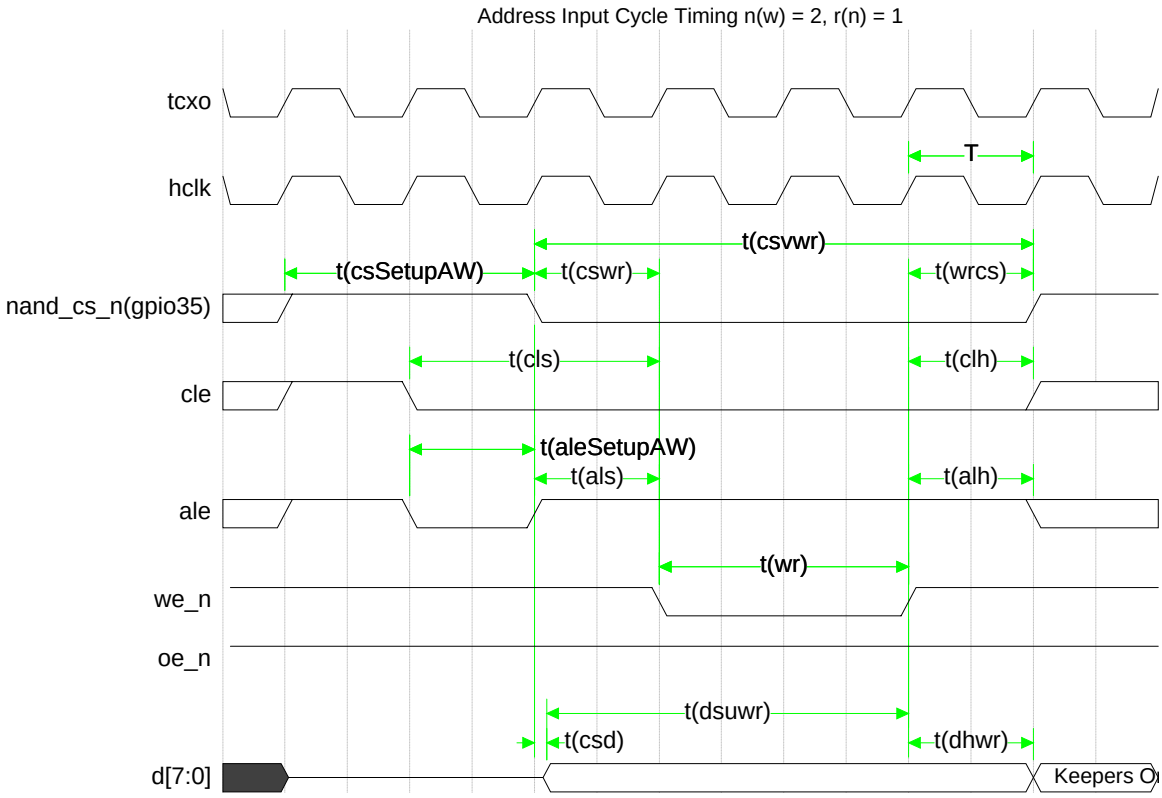


Figure 3-48 NAND Address Write Timing with 4 Access Cycles (Minimum Access Cycles + 2 Access WS) and 1 Recovery Cycle

Table 3-84 NAND Address Write Timing (2 and 3 Access Wait States)

Parameter		Equation	Notes
t_{csvwr}	Chip select active	$2T + (n_{w(wr)})T - \varepsilon_1$	
t_{dsuwr}	Write data setup	$T + (n_{w(wr)})T - \upsilon_1$	
t_{dhwr}	Write data hold	$T - \sigma_1$	
t_{cswr}	Chip select active to write active	$T - \omega_2$	
t_{wr}	Write active	$(n_{w(wr)})T - \zeta_1$	
t_{csd}	Chip select active to data valid	i_1	
t_{wrcs}	Write inactive to chip select inactive	$T - \mu_2$	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-84 NAND Address Write Timing (2 and 3 Access Wait States)

Parameter		Equation	Notes
$t_{csSetupAW}$	End of EBI2 XMEM controller access to NAND chip select active.	$(r_n)T$	Minimum of 2 cycles guaranteed by design.
$t_{aleSetupAW}$	Start of NAND controller access to NAND control latch enable active.	$(r_n)T - \psi_1$	
t_{als}	Address latch enable setup time to write active.	$T - \psi_5$	
t_{cls}	Command latch enable setup time to write active.	$2T - \psi_3$	Guaranteed by design. No measurement done.
t_{alh}	Address latch enable hold time from write inactive.	$T - \psi_6$	
t_{clh}	Command latch enable hold time from write inactive.	$T - \psi_4$	

NOTE ale (lb_n) and cle (ub_n) are LOW most of the time for accesses done by the EBI2 XMEM controller. However, if the XMEM controller access is a BYTE access to a 16-bit memory, then either lb_n or ub_n will be HIGH during that access. If that BYTE access precedes a NAND memory access, then either ale_n or cle_n setup, from the time the EBI2 XMEM access finishes to the time the NAND access we_n asserts, needs to be measured as required. Similarly, if the NAND access is followed by an EBI2 XMEM controller access that requires either recovery time or cs_setup wait states, then both ale (lb_n) and cle (ub_n) signals will be HIGH right after the end of the NAND access. In that case, the hold time from the NAND access we_n signal deasserting to the ale and cle signals rising needs to be measured.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.9.3.8 Status Read Cycle Timing (All Wait States)

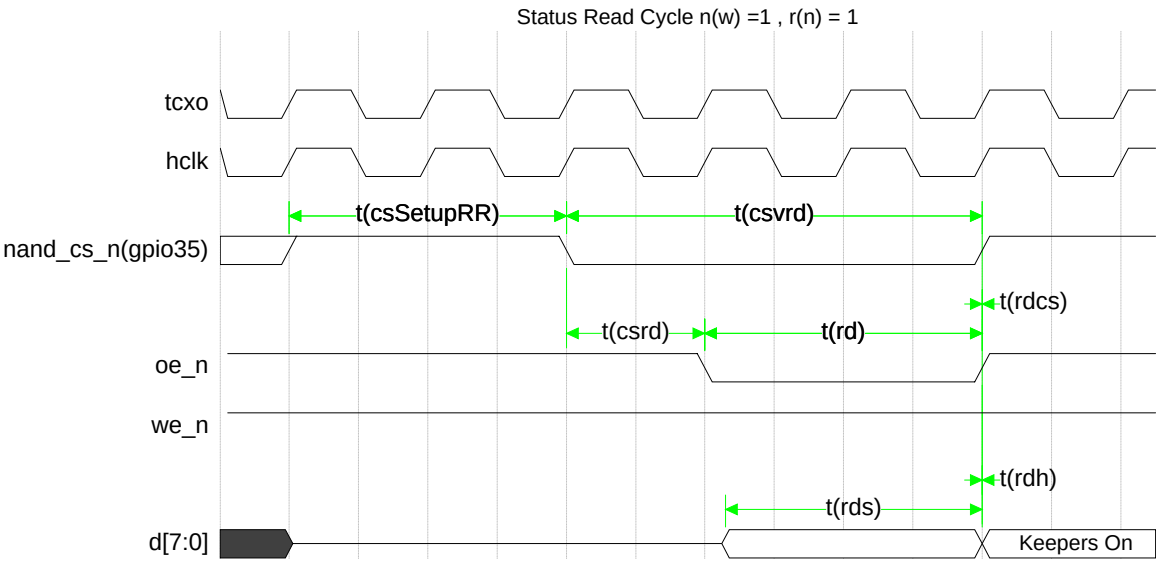


Figure 3-49 NAND Command Data Read Timing with 3 Access Cycles (Minimum Access Cycles + 1 Access WS) and 1 Recovery Cycle

Table 3-85 NAND Command Data Read Timing (All Accesses)

Parameter		Equation	Notes
t_{csvrd}	Chip select active	$2T + (n_{w(rd)})T - \varepsilon_2$	
t_{rds}	Read data setup	χ_1	
t_{rdh}	Read data hold	α_1	Max value also limited by $t_{csd.}$
t_{csrd}	Chip select active to read active	$T - \lambda_2$	
t_{rd}	Read active	$(1 + n_{w(rd)})T - \gamma_1$	
t_{rdcs}	Read inactive to chip select inactive	η_2	
$t_{csSetupRR}$	End of EBI2 XMEM controller access to NAND chip select active.	$(r_n)T$	Minimum of 2 cycles guaranteed by design.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.9.3.9 ID Read Cycle Timing (All Wait States)

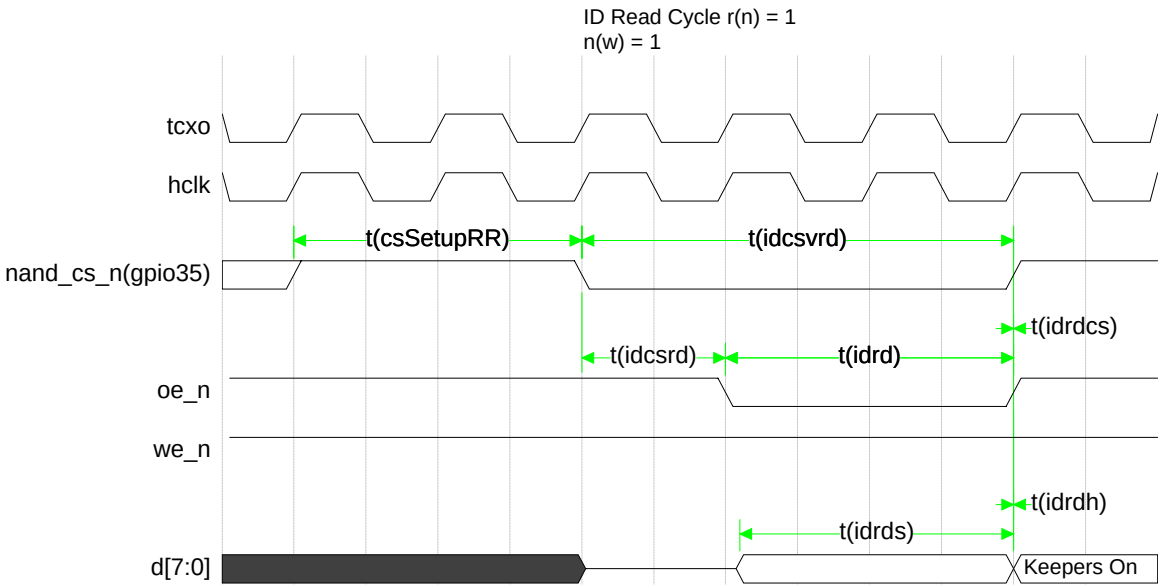


Figure 3-50 NAND ID Data Read Timing with 3 Access Cycles (Minimum Access Cycles + 1 Access WS) and 1 Recovery Cycle

Table 3-86 NAND Command Data Read Timing (All Accesses)

Parameter		Equation	Notes
$t_{idcsvrd}$	Chip select active for ID Read Access	$2T + (n_{w(rd)})T - \varepsilon_5$	
t_{idrds}	Read data setup for ID Read Access	χ_1	
t_{idrdh}	Read data hold for ID Read Access	α_1	Max value also limited by $t_{csd.}$
t_{idcsrd}	Chip select active to read active for ID Read Access	$T - \lambda_4$	
t_{idrd}	Read active for ID Read Access	$(1 + n_{w(rd)})*T - \gamma_5$	
t_{idrdcs}	Read inactive to chip select inactive for ID Read Access	η_2	
$t_{csSetupRR}$	End of EBI2 XMEM controller access to NAND chip select active.	$(r_n)T$	Minimum of 2 cycles guaranteed by design.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.9.3.10 Data Write Timing (0 and 1 Wait States)

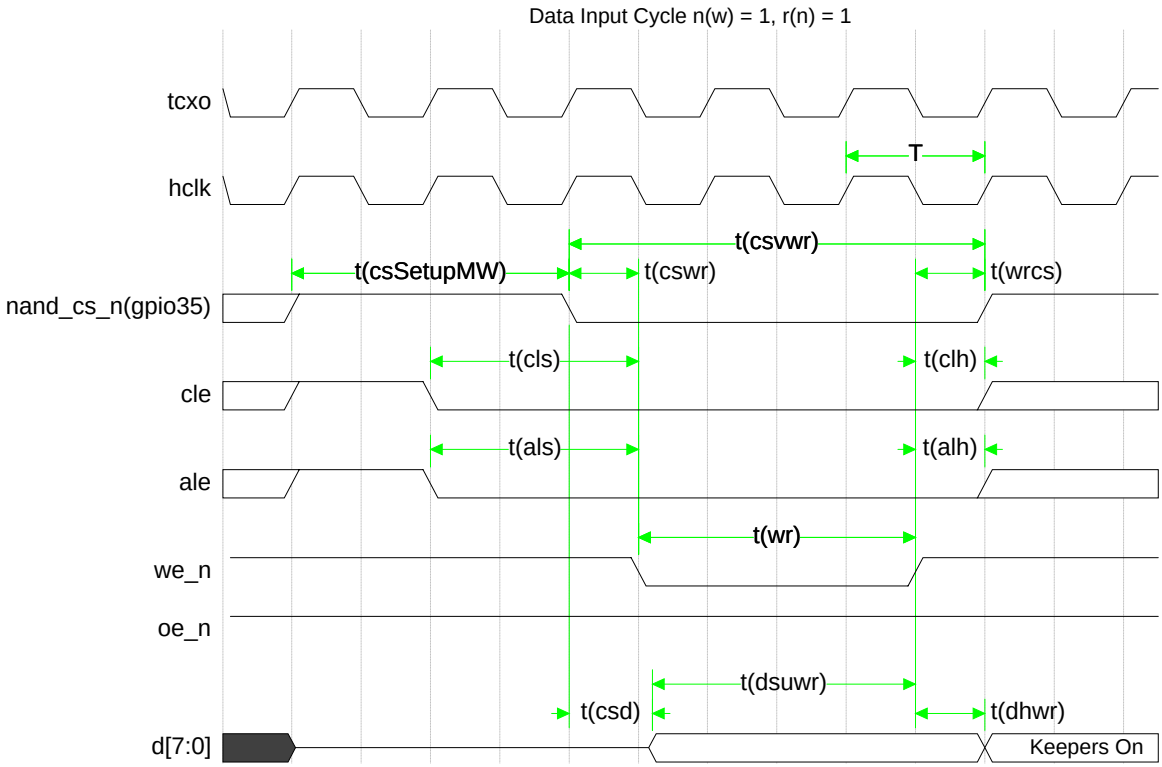


Figure 3-51 NAND Data Write Timing with 3 Access Cycles (Minimum Access Cycles + 1 Access WS) and 1 Recovery Cycle

Table 3-87 NAND Data Write Timing (0 and 1 Access Wait States)

Parameter		Equation	Notes
t_{csvwr}	Chip select active	$2T + (n_{w(wr)})T - \varepsilon_3$	
t_{dsuwr}	Write data setup	$(1 + n_{w(wr)})T - \upsilon_2$	
t_{dhwr}	Write data hold	$0.5T - \sigma_2$	
t_{cswr}	Chip select active to write active	$0.5T - \omega_3$	
t_{wr}	Write active	$(1 + n_{w(wr)})T - \zeta_2$	
t_{csd}	Chip select active to data valid	$0.5T - i_2$	
t_{wrcs}	Write inactive to chip select inactive	$0.5T - \mu_3$	
$t_{csSetupMW}$	End of EBI2 XMEM controller access to NAND chip select active.	$(r_n)T$	Minimum of 2 cycles guaranteed by design.
t_{cls}	Command latch enable setup time to write active.	$1.5*T - \psi_3$	Guaranteed by design. No measurement done

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-87 NAND Data Write Timing (0 and 1 Access Wait States)

Parameter		Equation	Notes
t_{als}	Address latch enable setup time to write active.	1.5*T - ψ_5	Guaranteed by design. No measurement done
t_{clh}	Command latch enable hold time from write inactive.	0.5T - ψ_4	
t_{alh}	Address latch enable hold time from write inactive.	0.5T - ψ_6	

NOTE ale (lb_n) and cle (ub_n) are LOW most of the time for accesses done by the EBI2 XMEM controller. However, if the XMEM controller access is a BYTE access to a 16-bit memory, then either lb_n or ub_n will be HIGH during that access. If that BYTE access precedes a NAND memory access, then either ale_n or cle_n setup, from the time the EBI2 XMEM access finishes to the time the NAND access we_n asserts, needs to be measured as required. Similarly, if the NAND access is followed by an EBI2 XMEM controller access that requires either recovery time or cs_setup wait states, then both ale (lb_n) and cle (ub_n) signals will be HIGH right after the end of the NAND access. In that case, the hold time from the NAND access we_n signal deasserting to the ale and cle signals rising needs to be measured.

3.2.9.3.11 Data Write Timing (2 and 3 Wait States)

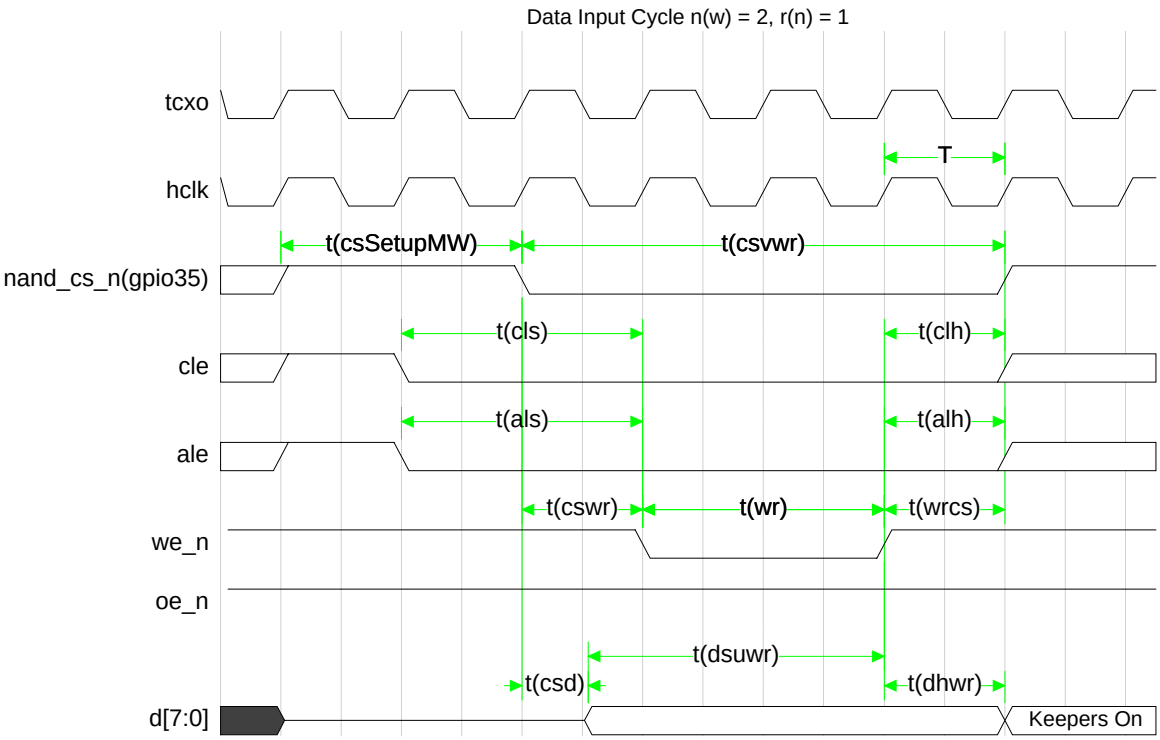


Figure 3-52 NAND Data Write Timing with 4 Access Cycles (Minimum Access Cycles + 2 Access WS) and 1 Recovery Cycle

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-88 NAND Data Write Timing (2 and 3 Access Wait States)

Parameter		Equation	Notes
t_{csvwr}	Chip select active	$2T + (n_{w(wr)})T - \varepsilon_3$	
t_{dsuwr}	Write data setup	$0.5T + (n_{w(wr)})T - \upsilon_2$	
t_{dhwr}	Write data hold	$T - \sigma_2$	
t_{cswr}	Chip select active to write active	$T - \omega_3$	
t_{wr}	Write active	$(n_{w(wr)})T - \zeta_2$	
t_{csd}	Chip select active to data valid	$0.5T - i_2$	
t_{wrcs}	Write inactive to chip select inactive	$T - \mu_3$	
$t_{csSetupMW}$	End of EBI2 XMEM controller access to NAND chip select active.	$(r_n)T$	Minimum of 2 cycles guaranteed by design.
t_{cls}	Command latch enable setup time to write active.	$2T - \psi_3$	Guaranteed by design. No measurement done
t_{als}	Address latch enable setup time to write active.	$2T - \psi_5$	Guaranteed by design. No measurement done
t_{clh}	Command latch enable hold time from write inactive.	$T - \psi_4$	
t_{alh}	Command latch enable hold time from write inactive.	$T - \psi_6$	

NOTE ale (lb_n) and cle (ub_n) are LOW most of the time for accesses done by the EBI2 XMEM controller. However, if the XMEM controller access is a BYTE access to a 16-bit memory, then either lb_n or ub_n will be HIGH during that access. If that BYTE access precedes a NAND memory access, then either ale_n or cle_n setup, from the time the EBI2 XMEM access finishes to the time the NAND access we_n asserts, needs to be measured as required. Similarly, if the NAND access is followed by an EBI2 XMEM controller access that requires either recovery time or cs_setup wait states, then both ale (lb_n) and cle (ub_n) signals will be HIGH right after the end of the NAND access. In that case, the hold time from the NAND access we_n signal deasserting to the ale and cle signals rising needs to be measured.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.9.3.12 Memory Data Read Timing (All Accesses)

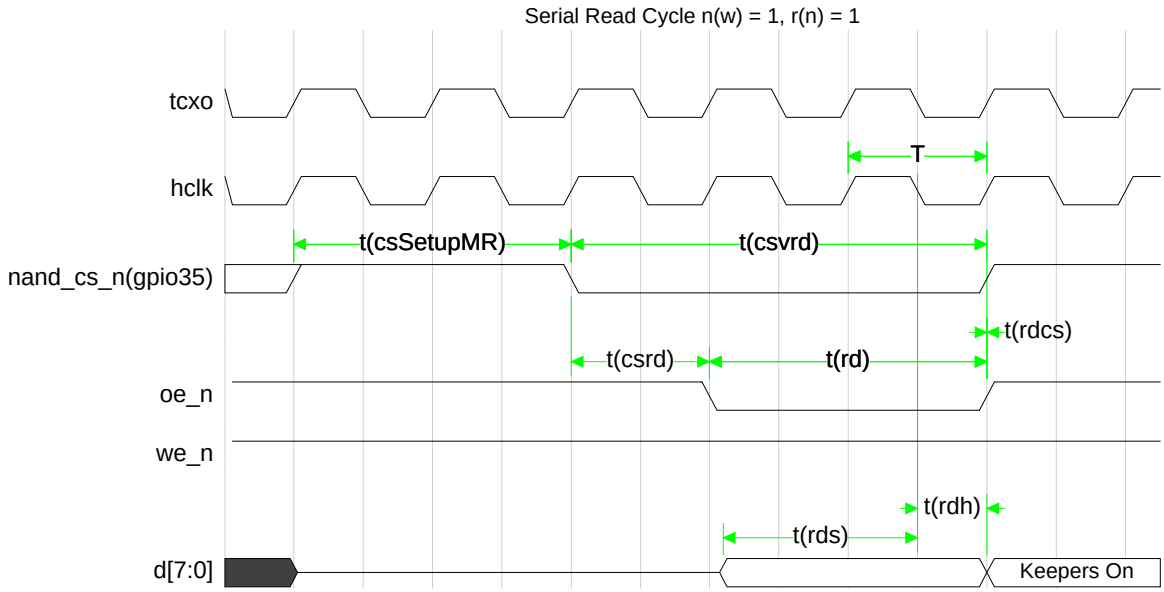


Figure 3-53 NAND Memory Data Read Timing (All Accesses)

Table 3-89 NAND Memory Data Read Timing (All Accesses)

Parameter		Equation	Notes
t_{csvrd}	Chip select active	$2T + (n_{w(rd)})T - \varepsilon_4$	
t_{rds}	Read data setup	$0.5T + \chi_2$	
t_{rdh}	Read data hold	$-0.5T + \alpha_2$	Max value also limited by t_{csd}.
t_{csrd}	Chip select active to read active	$T - \lambda_3$	
t_{rd}	Read active	$(1 + n_{w(rd)})T - \gamma_2$	
t_{rdcs}	Read inactive to chip select inactive	η_3	
$t_{csSetupMR}$	End of EBI2 XMEM controller access to NAND chip select active.	$(r_n)T$	Minimum of 2 cycles guaranteed by design.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.9.3.13 NAND Access Timing for Consecutive NAND Accesses Case

If the MSM 6100 External Bus Interface 2 (EBI2) has no requests other than those from the NAND controller, the NAND controller can do consecutive accesses to the NAND memory.

The characterization timings shown on the following sections refer to the specific condition of consecutive NAND accesses that occur when all the ARM, the DSP, and the GRP buses have no EBI2 access requests.

3.2.9.3.14 Consecutive Address Access Timing Sequence (0 and 1 Wait States)

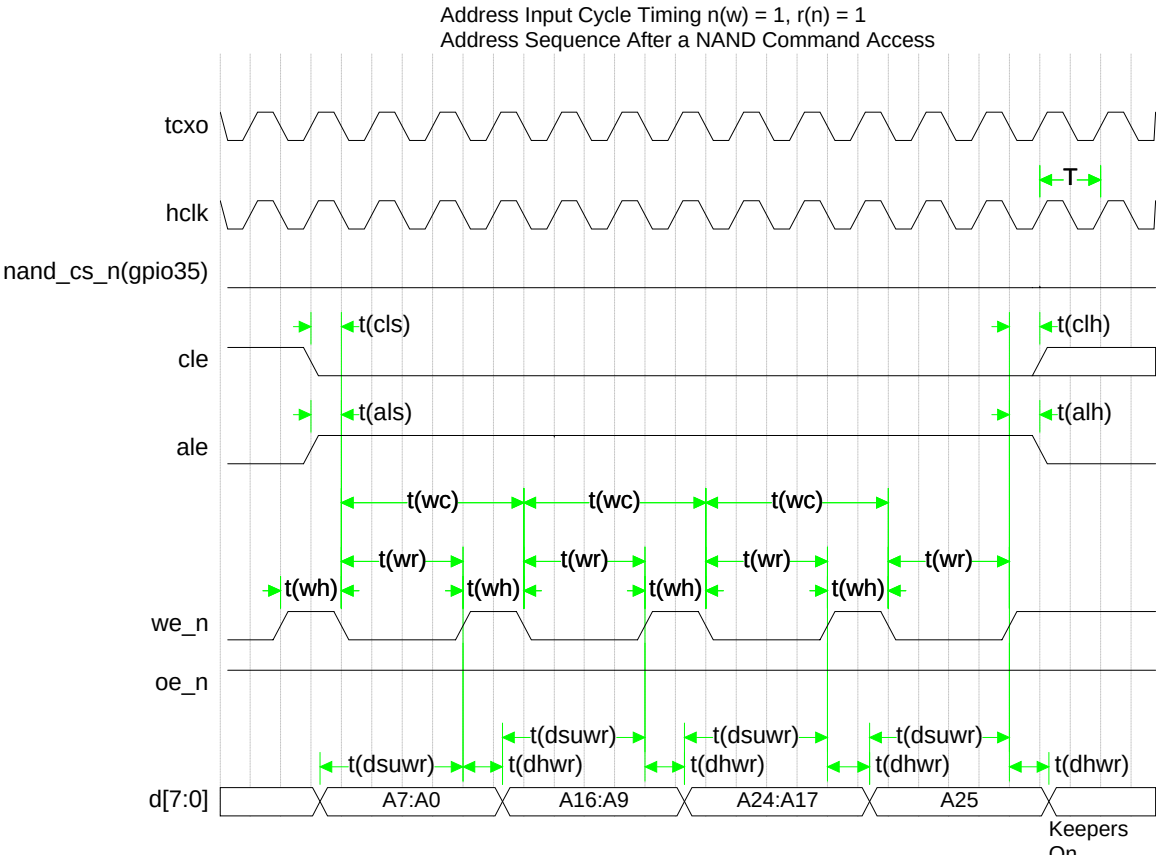


Figure 3-54 Consecutive NAND Address Write Timing with 3 Access Cycles (Minimum Access Cycles + 1 Access WS)

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-90 Consecutive NAND Address Write Timing (0 and 1 Wait States) after a NAND Command Access

Parameter		Equation	Notes
t_{wc}	Write Cycle Time	$2T + (n_{w(wr)})T - \zeta_3$	
t_{dsuwr}	Write data setup	$1.5T + (n_{w(wr)})T - \upsilon_2$	
t_{dhwr}	Write data hold	$0.5T + \sigma_2$	
t_{wr}	Write active	$(1 + n_{w(wr)})T - \zeta_2$	
t_{wh}	Write high	$T - \zeta_4$	
t_{cls}	Command latch enable setup time to write active.	$0.5T - \psi_3$	
t_{als}	Address latch enable setup time to write active.	$0.5T - \psi_5$	
t_{clh}	Command latch enable hold time from write inactive.	$0.5T - \psi_4$	
t_{alh}	Address latch enable hold time from write inactive.	$0.5T - \psi_6$	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.9.3.15 Consecutive Address Access Timing Sequence (2 and 3 Wait States)

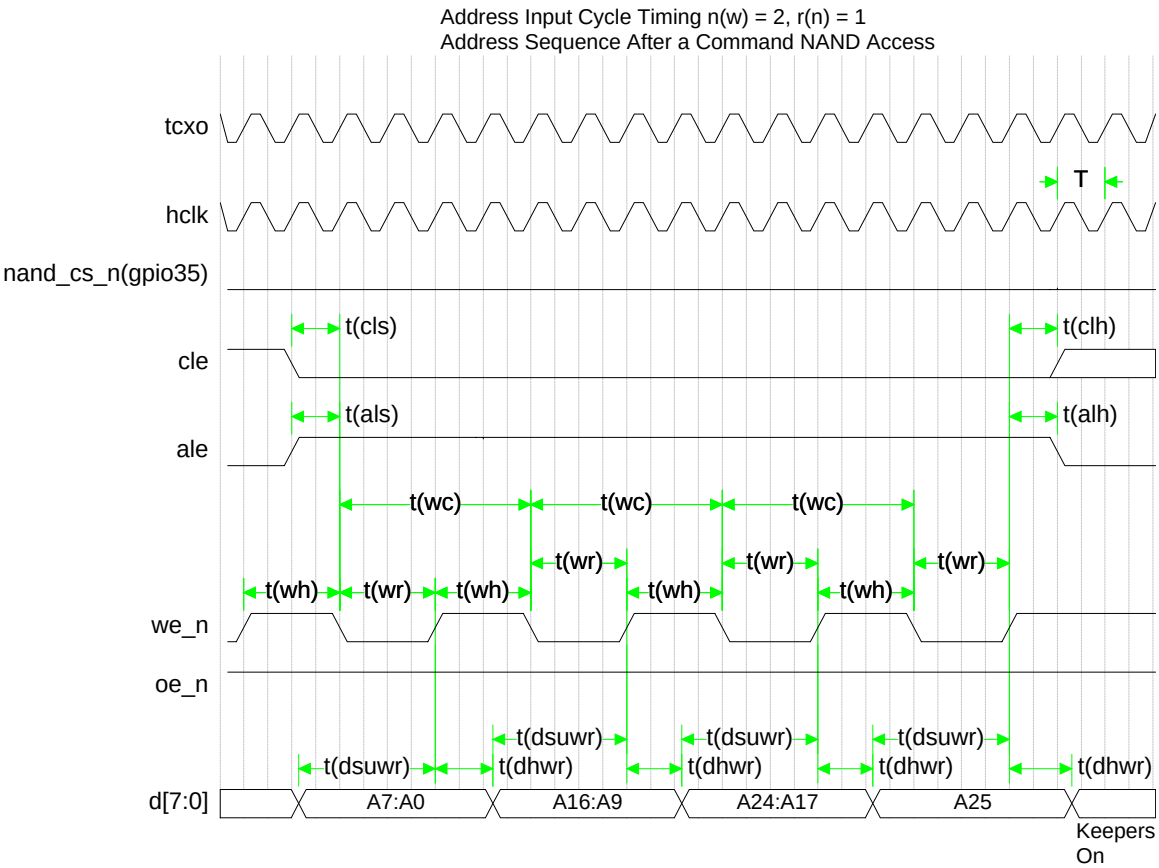


Figure 3-55 Consecutive NAND Address Write Timing with 4 Access Cycles (Minimum Access Cycles + 2 Access WS)

Table 3-91 Consecutive NAND Address Write Timing (2 and 3 Wait States) after a NAND Command Access

Parameter		Equation	Notes
t_{wc}	Write Cycle Time	$2T + (n_{w(wr)})T - \zeta_3$	
t_{dsuwr}	Write data setup	$T + (n_{w(wr)})T - \upsilon_2$	
t_{dhwr}	Write data hold	$T + \sigma_2$	
t_{wr}	Write active	$(n_{w(wr)})T - \zeta_2$	
t_{wh}	Write high	$2T - \zeta_4$	
t_{cls}	Command latch enable setup time to write active.	$T - \psi_3$	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-91 Consecutive NAND Address Write Timing (2 and 3 Wait States) after a NAND Command Access

Parameter		Equation	Notes
t_{als}	Command latch enable setup time to write active.	T - ψ_5	
t_{clh}	Command latch enable setup time to write active.	T - ψ_4	
t_{alh}	Command latch enable setup time to write active.	T - ψ_6	

3.2.9.3.16 Consecutive Data Write Access Timing Sequence (0 and 1 Wait States)

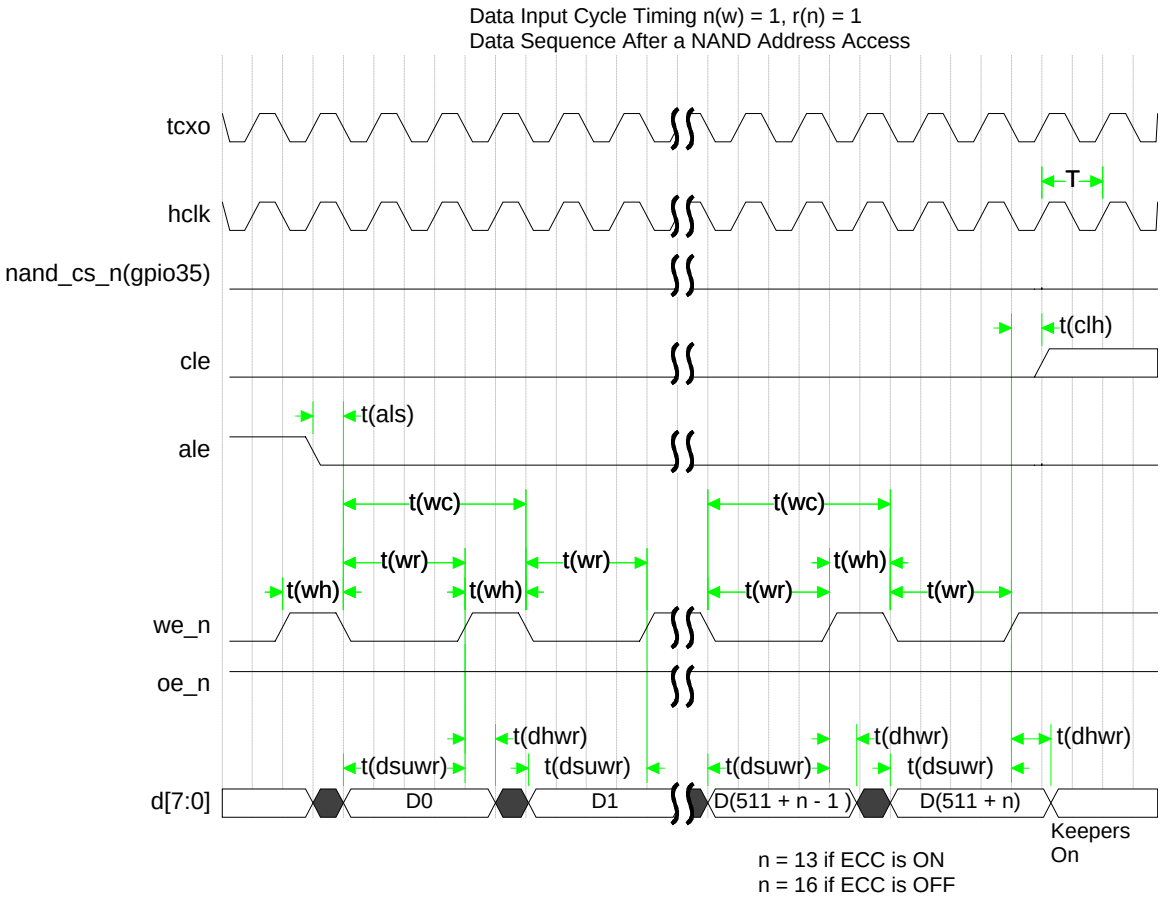


Figure 3-56 Consecutive NAND Data Write Timing with 3 Access Cycles (Minimum Access Cycles + 1 Access WS)

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-92 Consecutive NAND Data Write Timing (0 and 1 Wait States) after a NAND Address Access

Parameter		Equation	Notes
t_{wc}	Write Cycle Time	$2T + (n_{w(wr)})T - \zeta_3$	
t_{dsuwr}	Write data setup	$(1 + n_{w(wr)})T - \upsilon_2$	
t_{dhwr}	Write data hold	$0.5T + \sigma_2$	
t_{wr}	Write active	$(1 + n_{w(wr)})T - \zeta_2$	
t_{wh}	Write high	$T - \zeta_4$	
t_{als}	Address latch enable setup time to write active.	$0.5T - \psi_5$	
t_{clh}	Command latch enable hold time from write inactive.	$0.5T - \psi_4$	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.9.3.17 Consecutive Data Write Access Timing Sequence (2 and 3 Wait States)

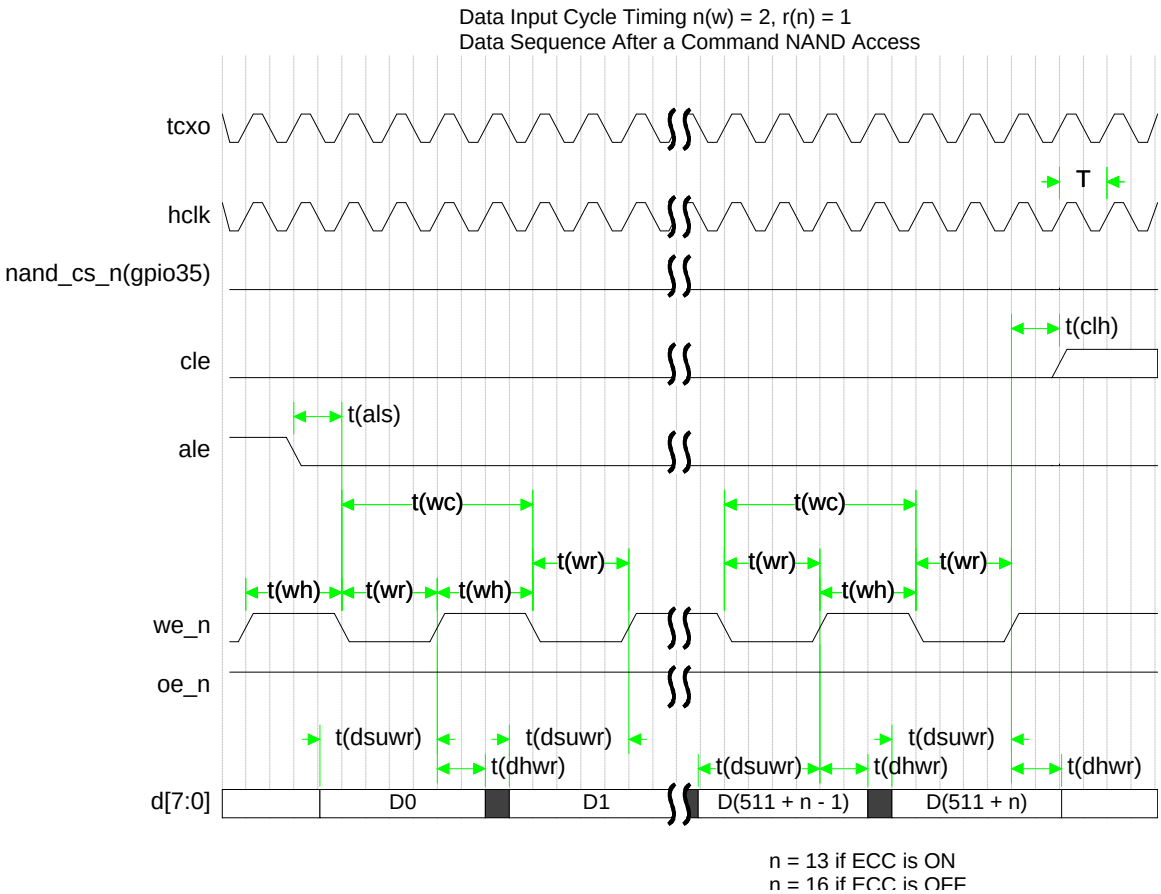


Figure 3-57 Consecutive NAND Command Write Timing with 4 Access Cycles (Minimum Access Cycles + 2 Access WS)

Table 3-93 Consecutive NAND Data Write Timing (2 and 3 Wait States) after a NAND Address Access.

Parameter		Equation	Notes
t_{wc}	Write Cycle Time	$2T + (n_{w(wr)})T - \zeta_3$	
t_{dsuwr}	Write data setup	$0.5T + (n_{w(wr)})T - \upsilon_2$	
t_{dhwr}	Write data hold	$T + \sigma_2$	
t_{wr}	Write active	$(n_{w(wr)})T - \zeta_2$	
t_{wh}	Write high	$2T - \zeta_4$	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-93 Consecutive NAND Data Write Timing (2 and 3 Wait States) after a NAND Address Access.

Parameter		Equation	Notes
t_{als}	Command latch enable setup time to write active.	T - ψ_5	
t_{clh}	Command latch enable setup time to write active.	T - ψ_4	

3.2.9.3.18 Consecutive Data Read Access Timing Sequence (All Accesses)

After Waiting for R/B_N to go high

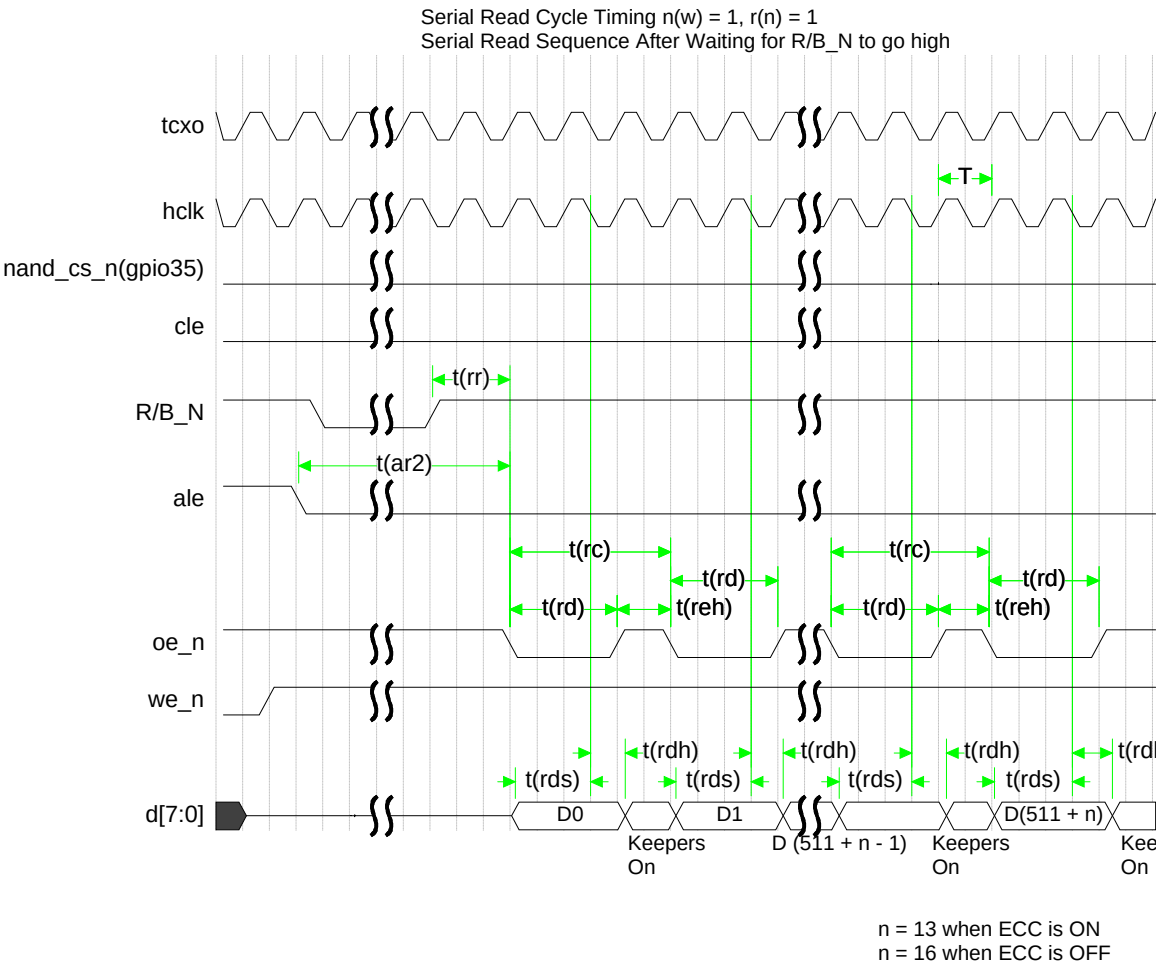


Figure 3-58 Consecutive NAND Data Read Timing with 3 Access Cycles (Minimum Access Cycles + 1 Access WS) After Waiting for R/B_N to Go High

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-94 Consecutive NAND Data Read Timing (All Wait States) After Waiting for R/B_N to Go High

Parameter		Equation	Notes
t_{rc}	Read Cycle Time	$2T + (n_{w(rd)})T - \gamma_3$	
t_{rds}	Read data setup	$0.5T + \chi_1$	
t_{rdh}	Read data hold	$-0.5T + \alpha_1$	Max value also limited by $t_{csd.}$
t_{rd}	Read active	$(1 + n_{w(rd)})T - \gamma_2$	
t_{reh}	Read high	$T - \gamma_4$	
t_{ar2}	ALE Inactive to read active	tr (Memory Cell Array Access Time) + 3.5T	Guaranteed by design. Nothing measured.
t_{rr}	R/B_N high to read active	3T	Guaranteed by design. Nothing measured.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.9.3.19 Status Command Sequence (0 and 1 Wait States)

Assuming XMEM Access Before Command Starts

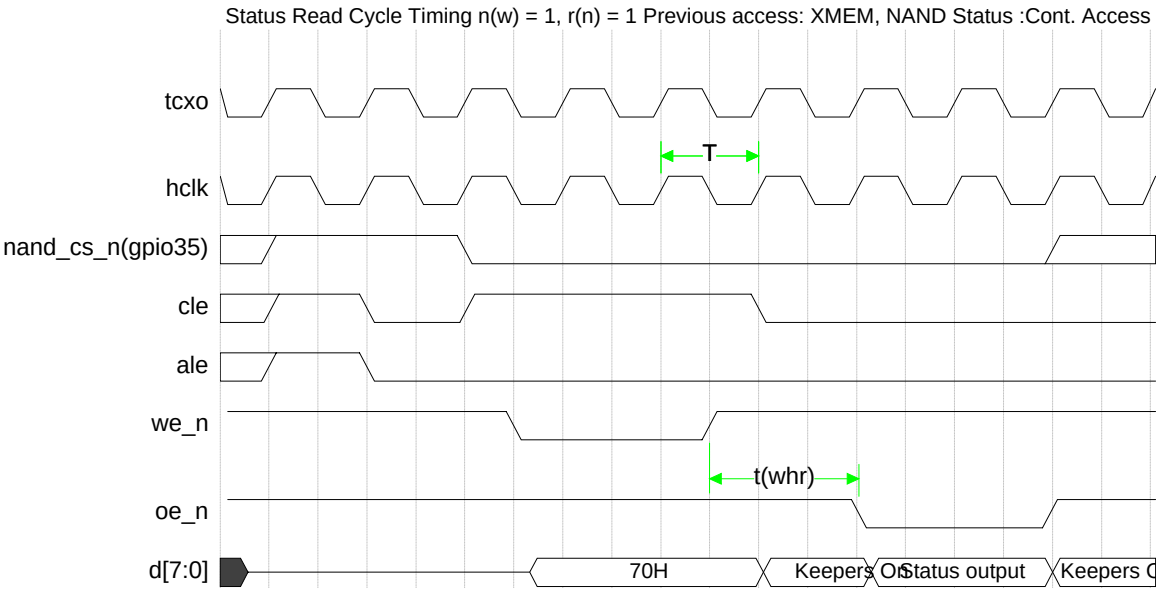


Figure 3-59 Continuous Status Command Sequence With 3 Access Cycles (Minimum Access Cycles + 1 Access WS)

Table 3-95 Continuous Status Command Sequence With First Access Following an XMEM Access (0 and 1 Wait States)

Parameter		Equation	Notes
t_{whr}	Write inactive to read active	1.5T - π_1	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.9.3.20 Status Command Sequence (2 and 3 Wait States)

Assuming XMEM Access Before Command Starts

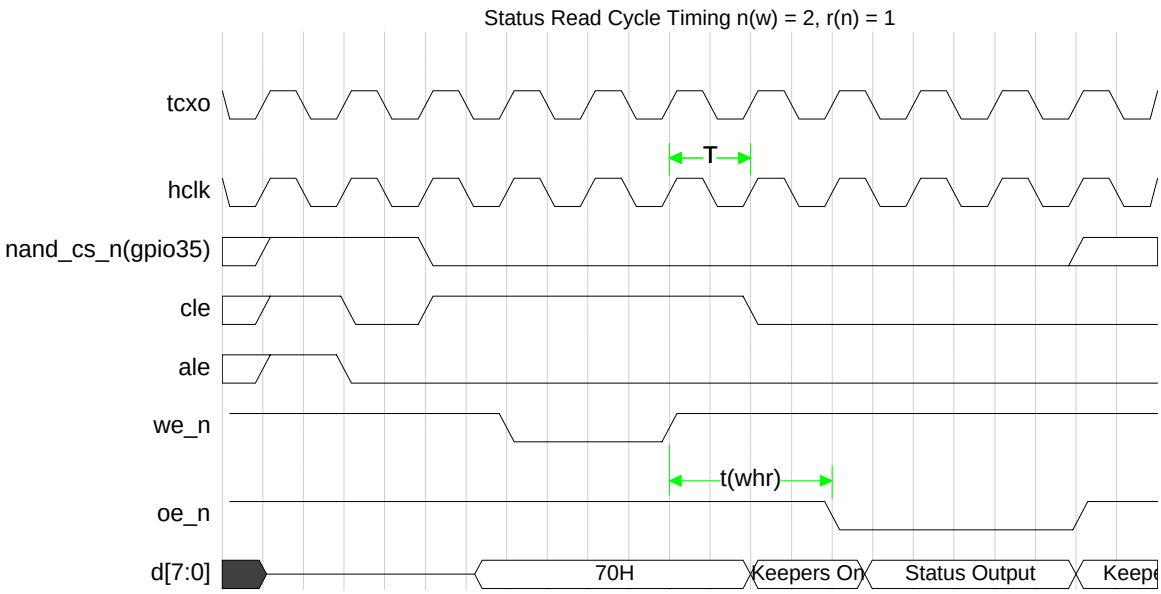


Figure 3-60 Continuous Status Command Sequence With 4 Access Cycles (Minimum Access Cycles + 2 Access WS)

Table 3-96 Continuous Status Command Sequence With First Access Following an XMEM Access (2 and 3 Wait States)

Parameter		Equation	Notes
t_{wrh}	Write inactive to chip select inactive	$2T - \pi_1$	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.9.3.21 ID Command Sequence (0 and 1 Wait States)

Assuming XMEM Access Before Command Starts

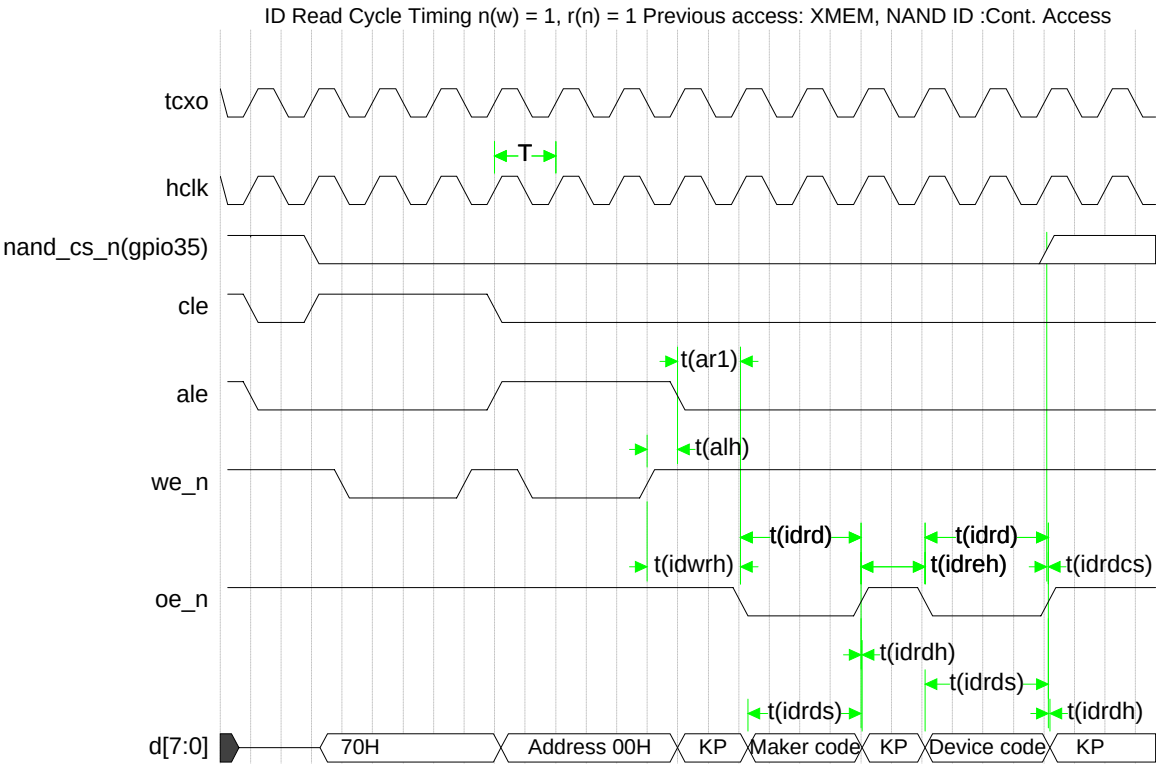


Figure 3-61 Continuous ID Command Sequence With First Access Following an XMEM Access(0 and 1 Wait States)

Table 3-97 Continuous ID Command Sequence With First Access Following an XMEM Access (0 and 1 Wait States)

Parameter		Equation	Notes
t_{alh}	Address latch enable hold time from write inactive.	0.5T - ψ_6	
t_{ar1}	Address latch enable low to read active.	T - π_3	
t_{idwrh}	Write inactive to read active for id read sequence	1.5T - π_2	
t_{idrds}	Read data setup for ID Read Access	χ_1	
t_{idrdh}	Read data hold for ID Read Access	α_1	Max value also limited by $t_{csd.}$

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-97 Continuous ID Command Sequence With First Access Following an XMEM Access (0 and 1 Wait States) (Continued)

Parameter		Equation	Notes
t_{idrd}	Read active for ID Read Access	$(1 + n_{w(rd)})T - \gamma_6$	
t_{idreh}	Read high for ID Read Access	$T - \gamma_7$	
t_{idrdcs}	Read inactive to chip select inactive for ID Read Access	η_2	

3.2.9.3.22 ID Command Sequence (2 and 3 Wait States)

Assuming XMEM Access Before Command Starts

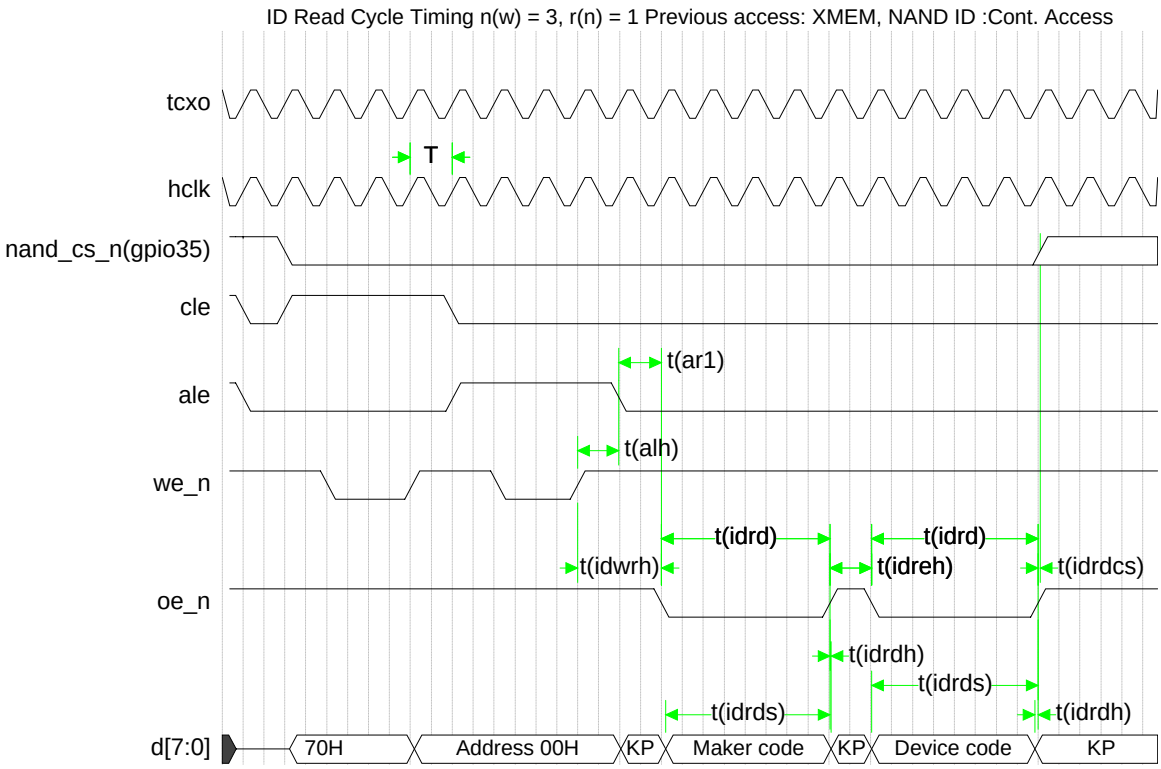


Figure 3-62 Continuous ID Command Sequence With First Access Following an XMEM Access (2 and 3 Wait States)

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

Table 3-98 Continuous ID Command Sequence With First Access Following an XMEM Access (2 and 3 Wait States)

Parameter		Equation	Notes
t_{alh}	Address latch enable hold time from write inactive.	$T - \psi_6$	
t_{ar1}	Address latch enable low to read active.	$T - \pi_3$	
t_{idwhr}	Write inactive to read active for id read sequence	$2T - \pi_2$	
t_{idrds}	Read data setup for ID Read Access	χ_1	
t_{idrdh}	Read data hold for ID Read Access	α_1	Max value also limited by $t_{csd.}$
t_{idrd}	Read active for ID Read Access	$(1 + n_{w(rd)})T - \gamma_6$	
t_{idreh}	Read high for ID Read Access	$T - \gamma_7$	
t_{idrdcs}	Read inactive to chip select inactive for ID Read Access	η_2	

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

3.2.10 JTAG Timing

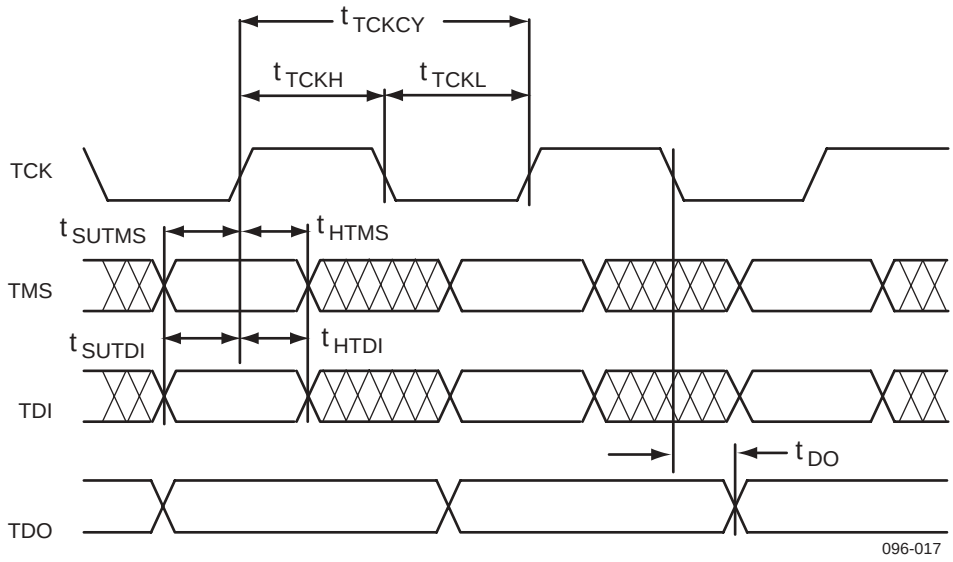


Figure 3-63 JTAG Interface Timing

Table 3-99 JTAG Interface Timing

Parameter	Description	Min	Typical	Max	Units
t_{TCKCY}	TCK period	100	—	—	ns
t_{TCKH}	TCK pulse width high	40	—	—	ns
t_{TCKL}	TCK pulse width low	40	—	—	ns
t_{SUTMS}	TMS Input Set-up Time	25	—	—	ns
t_{HTMS}	TMS Input Hold Time	25	—	—	ns
t_{SUTDI}	TDI Input Set-up Time	25	—	—	ns
t_{HTDI}	TDI Input Hold Time	25	—	—	ns
t_{DO}	TDO Data Output Delay	—	—	70	ns

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

NOTE: Specifications listed in this chapter are target specifications for the MSM6100 device and are subject to change without notice.

4 MSM6100 Interface Descriptions

Since the MSM6100 device is the central interface device of the subscriber unit, it sends and receives information to and from most of the other internal components of the phone. This chapter discusses the interfaces to peripheral devices supported by the MSM6100 device. The sections of this chapter are:

4.1 Overview

This chapter contains information needed to design the MSM6100 device into a subscriber unit application and describes some of the internal blocks of the device necessary for complete understanding of the various interfaces. This chapter also discusses the interfaces to the major blocks of the MSM6100 device, as shown in Figure 4-1.

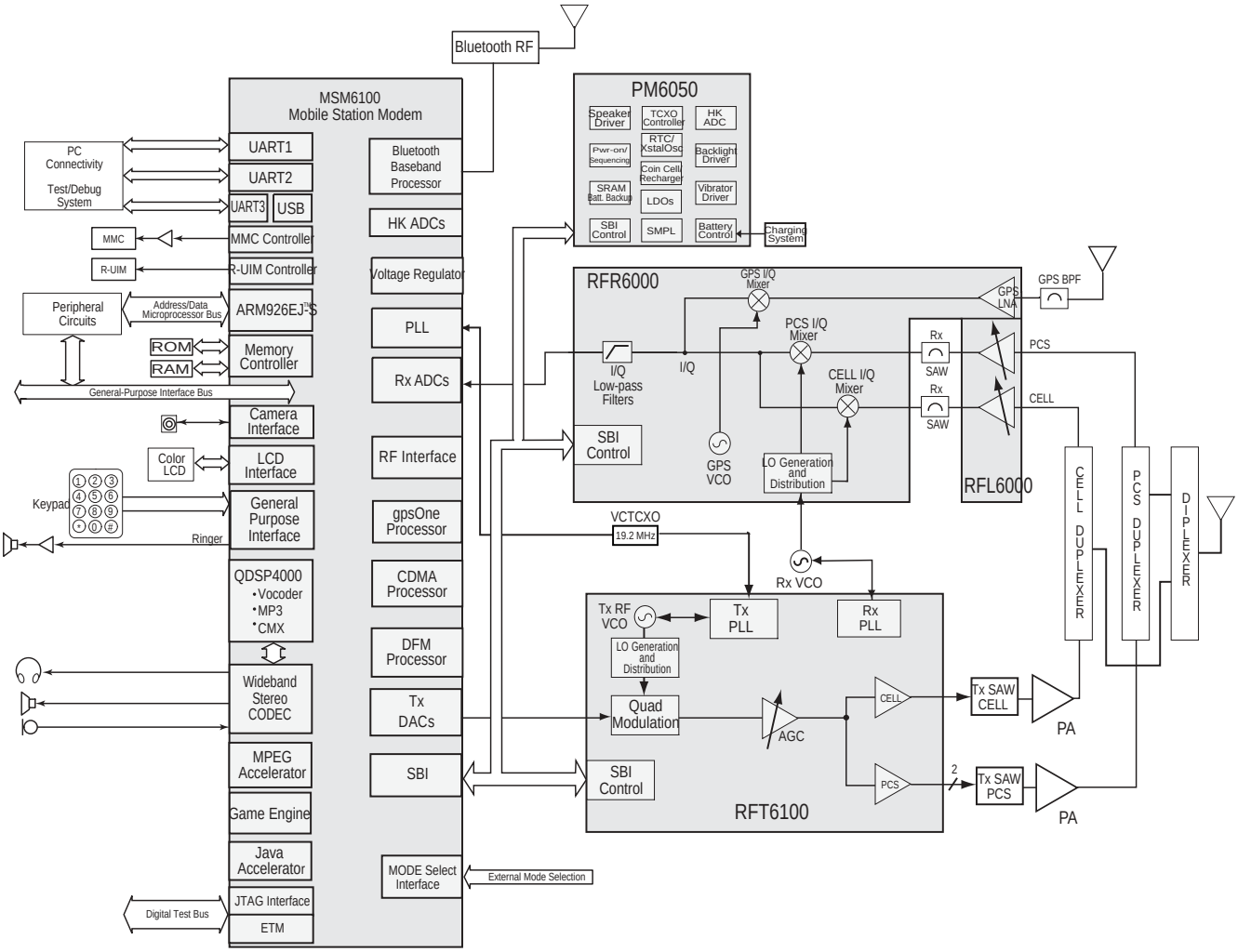


Figure 4-1 MSM6100 Device Functional Block Diagram

4.2 MSM6100 Mobile Station Modem ASIC Overview

CDMA Subsystem

The CDMA Subsystem performs the digital IS-95-A/B and IS-2000 signal processing. Its components include:

- Searcher engine
- Demodulating fingers
- Combining block
- Frame deinterleaver
- Viterbi decoder
- Reverse link subsystem
- Turbo decoder

On the forward-link traffic channel the CDMA subsystem searches, demodulates, and decodes incoming Pilot, Sync, Paging, and Traffic Channel information. It extracts low bit-rate packet data from the forward-link traffic channel and sends the packet data to the vocoder for processing. For the reverse link, the CDMA subsystem processes the packet data from the vocoder and modulates the reverse traffic channel.

Digital FM Subsystem

The MSM6100 device also supports Advanced Mobile Phone System (AMPS) cellular operations. The Digital FM (DFM) processor performs digital signal processing operations for the mobile station using the AMPS cellular standard.

RF Interface

The RF interface communicates with the mobile station's external RF circuits. Signals to these circuits control signal gain in the Rx and Tx signal path, control DC offset errors, and maintain the system's frequency reference.

Stereo Wideband CODEC

The MSM6100 device integrates a stereo wideband voice/audio codec into the mobile station modem (MSM). The codec supports two differential microphone inputs, one differential earphone output, one single-ended earphone output, and a differential auxiliary interface on two single-ended line outputs. The codec integrates the microphone and earphone amplifiers into the MSM6100 device, reducing the external component count to just a few passive components. The microphone (Tx) audio path consists of a two-stage amplifier with the gain of the second stage set externally. The Rx/Tx paths are designed to meet the ITU-G.712 requirements for digital transmission systems.

Vocoder Subsystem

The MSM6100 device's QDSP4000 supports EVRC and QCELP13K vocoders along with DFM base-band audio processing. In addition, the QDSP4000 has modules to support the following audio functions: DTMF tone generation, DTMF tone detection, Tx/Rx volume controls, Tx/Rx automatic gain control (AGC), Rx Automatic Volume Control (AVC), EarSeal Echo Canceller (ESEC), Acoustic Echo Canceller (AEC), Noise Suppression (NS), Supervisory Audio Tone (SAT) transponding in DFM mode, and programmable, 13-tap, Type-I, FIR, Tx/Rx compensation filters. The MSM6100 device's integrated ARM926EJ-S processor downloads the firmware into the QDSP4000 and configures QDSP4000 to support the desired functionality.

ARM Microprocessor Subsystem

The MSM6100 device uses an embedded ARM926EJ-S microprocessor. This microprocessor, through the system software, controls most of the functionality for the MSM device, including control of the external peripherals such as the keypad, LCD, RAM, ROM, and EEPROM devices. Through a generic serial bus interface (SBI) the ARM926EJ-S configures and controls the functionality of the RFL6000, RFR6000, RFT6100. and PM6xxx devices.

UART

The MSM6100 device employs three UARTs. UART1 has dedicated pins while UART2 and UART3 share multiplexed pins.

USB

The MSM6100 device integrates a universal serial bus (USB) controller that supports both unidirectional and bidirectional transceiver interfaces. The USB controller acts as a USB peripheral communicating with the USB host.

User Interface

The MSM6100 device user interface comprises digital connections to the subscriber unit ringer transducer, keypad, and LCD.

General-Purpose Interface Bus

The MSM6100 device has general-purpose bidirectional input/output pins. Some of the GPIO pins have alternate functions supported on them. The alternate functions include USB interface, additional RAM, ROM, general-purpose chip selects, parallel LCD interface, and a UART interface. The function of these pins is documented in the various software releases.

Mode Select and JTAG Interfaces

The mode pins to the MSM6100 device determine the overall operating mode of the ASIC. The options under the control of the mode inputs are Native mode, which is the normal subscriber unit operation, ETM mode, which enable the built-in trace mode, and test mode for factory testing.

The MSM6100 device meets the intent of ANSI/IEEE 1149.1A-1993 feature list. The JTAG interface can be used to test digital interconnects between devices within the mobile station during manufacture.

Camera Interface

The camera interface allows MSM6100 to be connected to an external camera with no external logic.

4.3 RF Interface

Precise power control of each mobile station is vital for CDMA system performance. It is the mobile station's responsibility to accurately estimate the power spectral density within the 1.2288 MHz bandwidth of the frequency channel used for CDMA demodulation. In this section, the power spectral density measurement is called the mean Rx power estimation, or the received signal strength indication (RSSI). The mobile station needs to estimate mean Rx power because it uses the RSSI to determine its mean Tx output power. The mean Tx output power of the Reverse Traffic Channel is specified by the following equation:

mean Tx output power (dBm) = – mean Rx input power (dBm)
+ offset power (–73 for cellular, –76 for PCS)
+ the sum of all access probe corrections
+ the sum of all closed loop power control corrections
+ a few other control parameters

The CDMA Tx AGC block controls the mobile station's mean Tx output power, as specified by the equation shown above. The mean Rx input power used by the Tx AGC block, is a filtered version of the RSSI created by the Rx AGC block. The offset power is determined by linearizer calibration procedures and corresponding software interpretation. Closed loop power control corrections are summed by the Tx AGC block circuits.

The Tx AGC block also controls the operating point of a multistate (high efficiency) power amplifier (PA) circuit. The CDMA Tx AGC block supports switching between up to four different PA operating points with programmable levels and temporal hysteresis.

Figure 4-2 shows the high-level functional block diagram for the MSM6100 RF interface pins, when used with radioOne™ RF ICs. These figures are only applicable to the functionality of the MSM6100 RF interface pins. All radioOne ICs are highly integrated and fulfill specific functions. The chipset includes:

- RFL6000 Dual LNA IC
- RFR6000 RF-to-Baseband Receiver IC
- RFT6100 Baseband-to-RF Transmitter IC
- PM6xxx Power Management IC
- MSM6100 Digital Processor IC

The RFL6000 IC is a standalone chip with two Low Noise Amplifiers (LNAs) which serves as front end low noise amplifiers prior to the RFR6000 chip and the MSM6100 device. It supports all modes of operation.

The RFR6000 IC provides the Zero-IF receiver signal path, from RF to analog baseband, for multi-band, multi-mode handsets.

The RFT6100 device is a Baseband-to-RF transmitter IC. All radioOne ICs are highly integrated and fulfill specific functions. Functional requirements are partitioned between the ICs to yield complete, optimal transceiver implementations.

The PM6xxx device is a custom mixed signal device containing all of the circuitry required to support the battery charging/monitoring and voltage regulation for a CDMA/AMPS or other mobile standard handset. It also contains extensive support for user interface devices.

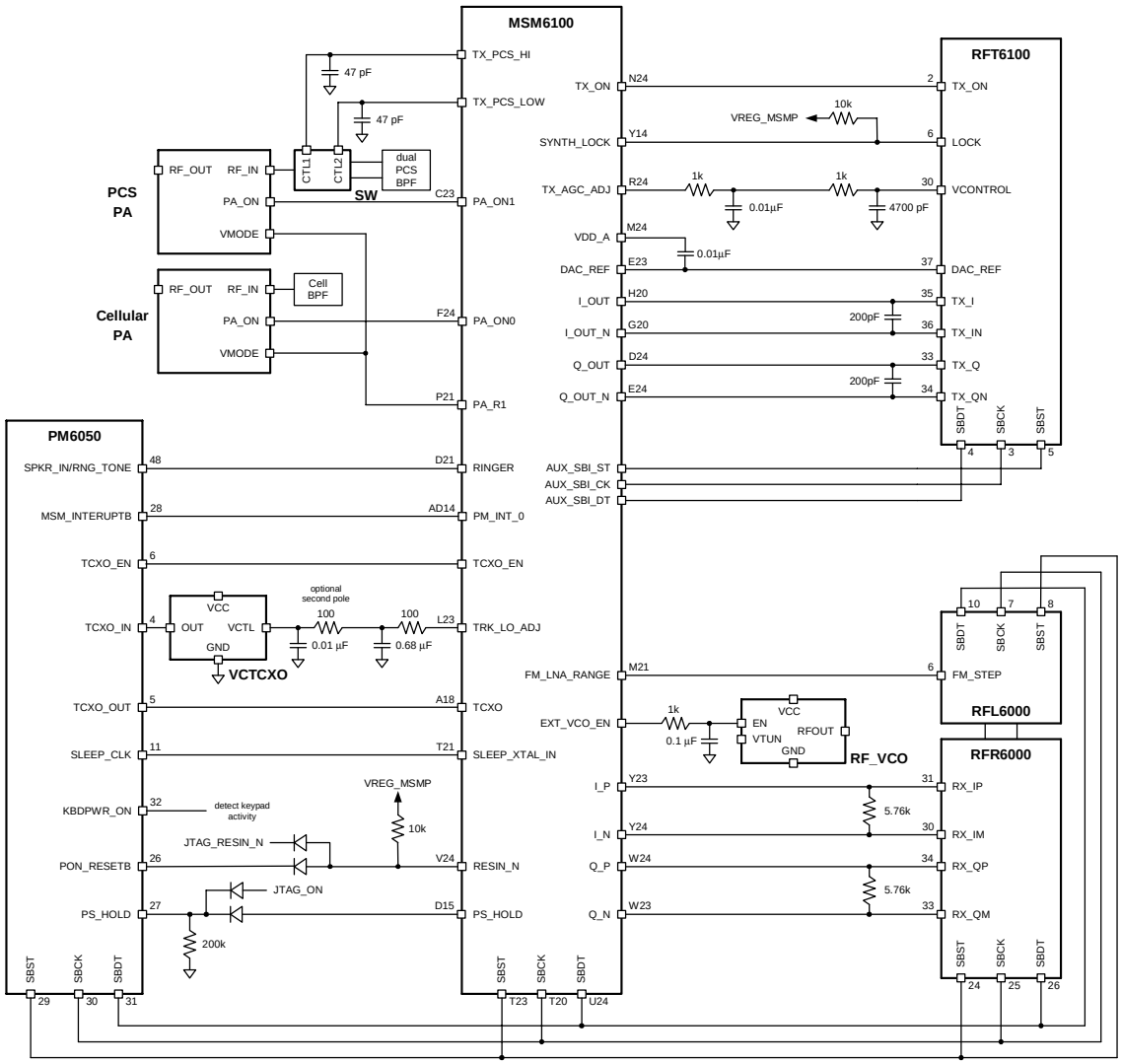


Figure 4-2 Interface Connections for the MSM6100 Chipset

Transmit Signal Paths

The MSM device provides I and Q differential baseband signals (FM or CDMA) to the RFT6100 IC via the proprietary analog interface. The analog input signals are amplified and applied to the upconverter mixers.

Q_OUT Transmit quadrature-phase analog output, differential positive

Q_OUT_N Transmit quadrature-phase analog output, differential negative

I_OUT Transmit in-phase analog output, differential positive

I_OUT_N Transmit in-phase analog output, differential negative

DAC_REF Reference input to MSM Tx data DACs

TX_ON Puncture control from MSM device: H = Tx on, L = Tx off

EXT_VCO_EN represents a pin used to control the RX power state of the VCO oscillator. If this pin is high, the VCO is enabled. If this pin is low, the VCO is disabled. The VCO must be disabled during GPS acquisition.

TX_AGC_ADJ is a PDM output pin used to control the attenuation (-gain in dB) of the Tx AGC amplifier. This signal is used as part of the CDMA and Digital FM Tx AGC loop.

PA_ON[1:0] are used to control the power state of the power amplifier (PA). PA_ON transitions high (active), a programmable amount of time before the beginning of a transmission. PA_ON remains high a short time after the end of the transmission. PA_ON0 is used for the cellular band, and PA_ON1 is used for the PCS band.

PA_R[1:0] are pins that can be used to control the operating point of the PA circuit.

TX_PCS_HIGH is a control SPDT (single pole double throw) switch used to choose PCS high-band (PCS channel > 600). TX_PCS_HI complements TX_PCS_LOW. Only used for US PCS handsets with split-band TX SAWs.

TX_PCS_LOW is a control SPDT (single pole double throw) switch used to choose PCS low-band (PCS channel < = 600). TX_PCS_LOW complements TX_PCS_HI. Only used for US PCS handsets with split-band TX SAWs.

Receive Signal Paths

The RFL6000 outputs drive the RF ports of the quadrature RFR6000 RF-to-baseband downconverters (a dedicated downconverter for each band). The downconverted baseband outputs are multiplexed and routed to lowpass filters. The filter outputs are buffered and passed on to the MSM device for further processing.

I_IN Receiver in-phase analog input (negative), a differential signal that complements I_IP.

I_IP Receiver in-phase analog input (positive), a differential signal that complements I_IN.

Q_IN Receiver quadrature analog input (negative), a differential signal that complements Q_IP.

Q_IP Receiver quadrature analog input (positive), a differential signal that complements Q_IN.

FM_LNA_RANGE is a digital output produced by the receive AGC subsystem. The FM_LNA_RANGE is designed to change states at programmable thresholds of receive signal strength in FM mode. This allows the subscriber unit to alter the receive signal path by bypassing or enabling the receive RF components at these thresholds. For the MSM6000 device, this pin requires a pull-up resistor to be connected to RFL6000 FM_STEP input. No pull-up is required for the MSM6100 device.

Others

TCXO is the primary 19.2 MHz clock source used by various blocks of the MSM6100 device, such as the ARM926EJ-S, ringer, UARTs, general-purpose PDMs, and the Digital FM circuits. TCXO can be used as a Vocoder clock source for EVRC support. TCXO is also used by the MSM6100 device to produce CHIPX8, CHIPX16, and CHIPX32.

TRK_LO_ADJ is a PDM output from the frequency tracking circuit which adjusts the subscriber frequencies. TRK_LO_ADJ is a PDM output from the frequency tracking subsystem which adjusts the subscriber units VCTCXO.

SYNTH_LOCK is an input pin that indicates the status of the Tx and Rx synthesizers. If the MSM6100 SYNTH_LOCK pin is high, the active synthesizer must be in-lock. If a Tx synthesizer is out-of-lock, SYNTH_LOCK is low to disable the PA_ON signal. LOCK is an open-drain output pin from the RFT6100, connect SYNTH_LOCK and LOCK directly together. Also connect a resistor from the SYNTH_LOCK to the MSM6100 device's positive supply.

Digital Interface

All control and status commands are communicated through the MSM-compatible 3-line Serial Bus Interface (SBI), allowing efficient initialization, control of device operating modes and parameters, verification of programmed parameters, and status reports. The MSM's SBI controller is the Master while the RF ICs are the Slaves.

SBST Serial Bus Interface (SBI) Strobe

SBCK SBI Clock

SBDT SBI Data

AUX_SBI_ST SBI bus for RFT6100 serial bus strobe

AUX_SBI_CK SBI bus for RFT6100 serial clock

AUX_SBI_DT SBI bus for RFT6100 serial bus data

NOTE Please refer to the *radioOne Zero-IF Chipset Design Guidelines* document, 93-V2685-3, for further discussions on the MSM6100 RF Interface.

4.4 Stereo Wideband CODEC

The MSM6100 device integrates a wideband audio CODEC into the Mobile Station Modem. The wideband codec allows the MSM device to support stereo music/ringer Melody applications in addition to the 8 kHz voice band applications on the forward link. In the audio transmit path, the device operates as 13-bit linear converter with software selectable 8 kHz and 16 kHz sampling rate. In the audio receive path, the device operates as a software selectable 13-bit or 16-bit linear converter with software selectable 8 kHz, 16 kHz, 22.05 kHz, 24 kHz, 32 kHz, 44.1 kHz, or 48 kHz sampling rate. Through software, the Rx path can be configured as either a mono or stereo output.

The integrated CODEC contains all of the required conversion and amplification stages for the audio front end. The CODEC operates as a 13-bit linear CODEC with the transmit (Tx) and receive (Rx) filters designed to meet ITU-T G.712 requirements. The CODEC includes a programmable sidetone path for summing a portion of the Tx audio into the Rx path. An on-chip Voltage/Current reference is provided to generate the precise voltages and currents required by the CODEC. This circuit requires a single capacitor of 0.1 uF to be connected between the CCOMP and GND (U14) pins. (See Figure 4-3.) The on-chip Voltage reference also provides a microphone bias voltage required for electret condenser microphones typically used in handset applications. The MICBIAS output pin is designed to provide 1.8 Volts DC while delivering as much as 1 mA of current. Audio decoder summing and headset switch detection are included.

The CODEC interface includes the amplification stages for both the microphone and earphone. The interface supports two differential microphone inputs and a differential auxiliary input, each of which can be configured as single-ended if desired. In addition, the interface supports one differential earphone output, one single-ended earphone output, and one differential auxiliary output or two single-ended line outputs.

The CODEC is configured through the QDSP4000 Command types and is not directly controlled by the microprocessor. The CODEC Configuration command is sent to the QDSP4000 and then the QDSP4000 executes the command and configures the CODEC. Data is exchanged between the codec interface and the QDSP4000 through its DMA interface. The QDSP4000 uses the Ex_DMA_4 channel for reading data from the codec and uses the Ex_DMA_5 channel to transfer data to the codec.

4.4.1 Transmit Path Processing

The microphone interface consists of two differential microphone inputs, one differential auxiliary input and a two-stage audio amplifier. The microphone input is selected by the CODEC Configuration command (MIC_SEL). Only one of the inputs is active at any given time and the other inputs are powered down.

The gain for the first stage amplifier can be set to either –2 db, +6 dB, +8 dB or +16 dB by the CODEC Configuration command (MIC_AMP1_GAIN). The outputs of the first stage are the output pins MICOUTP and MICOUTN.

The gain of the second stage amplifier is set externally. Additional filtering for the microphone can be designed into the external gain circuit in order to enhance the audio performance of the transmit channel. The second stage amplifier can also be bypassed internally by setting CODEC Configuration command (MIC_AMP2_BYP). The MICINP and MICINN are the inputs to the second stage amplifier and MICFBN and MICFBP are the feedback outputs. Figure 4-3 shows an example of an external circuit with a gain of 18 dB. In addition to the gain stage, the circuit contains a highpass filter that suppresses low frequencies. For an implementation of this circuit, refer to the MSM6100 Phone Reference Design document.

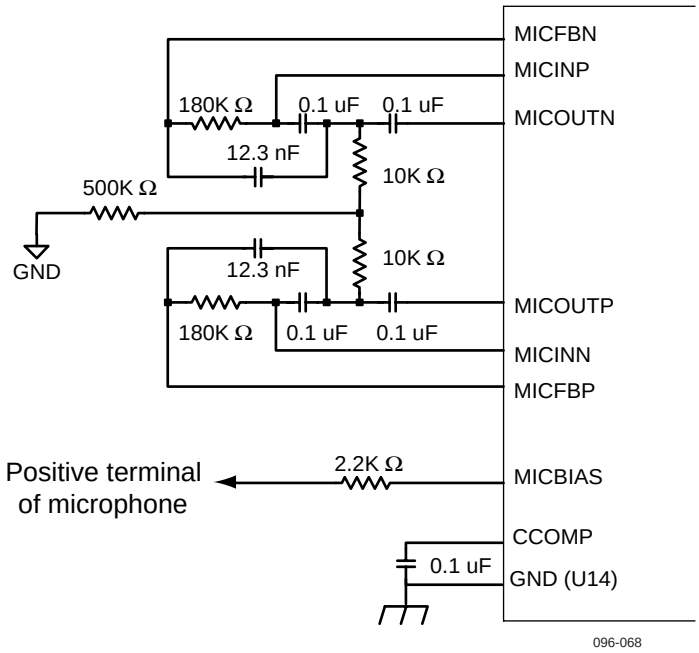


Figure 4-3 Example of an External Filter Network

The second stage mic amplifier (MICAMP2) is configured into a gain and filter stage. This is a high pass 2nd order Butterworth filter response. See Figure 4-4.

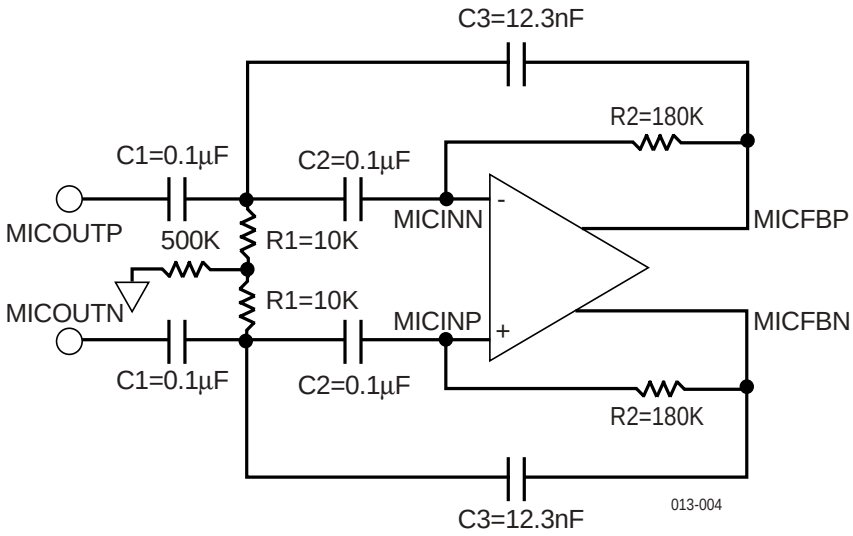


Figure 4-4 Equivalent Circuit for External Filter Network

Use the following equations to determine the values to R1, R2, C1, C2 and C3.

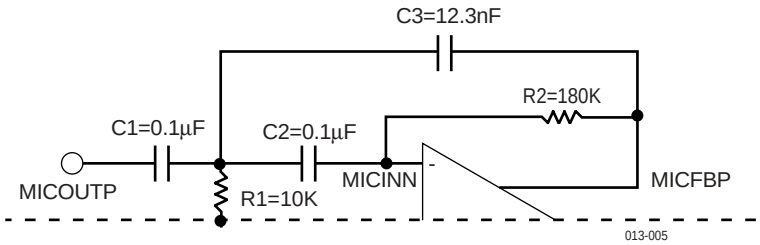


Figure 4-5 Equivalent circuit for MICAMP2

$|Ho| = C1 / C3$ => Eq1

$R1 = |Ho| / [2*\pi*fn*Q*C*\{2*|Ho|+1\}]$ where $C = C1 = C2$ => Eq2

$R2 = (Q*\{2*|Ho|+1\}) / (2*\pi*fn*C)$ => Eq3

For example:

If Q = 0.707

fn = 105.5 Hz

|Ho| = 8.13 = 18.2dB

C = C1 = C2 = 0.1µF

Using Eq1: C3 = 12.3nF

Using Eq2: R1 = 10K ohm

Using Eq3: R2 = 180K ohm

The transmit data from the microphone input is digitally filtered with an ITU G.712 compliant filter. The filter attenuates the input signals outside the 3400 Hz baseband and decimates the data rate to 8 kHz.

The MSM6100 device has two optional digital filters on the Tx path prior to the vocoder, a slope filter, and a highpass filter. The slope filter, Figure 4-6, is designed to provide pre-emphasis for the high frequency audio prior to the vocoder.

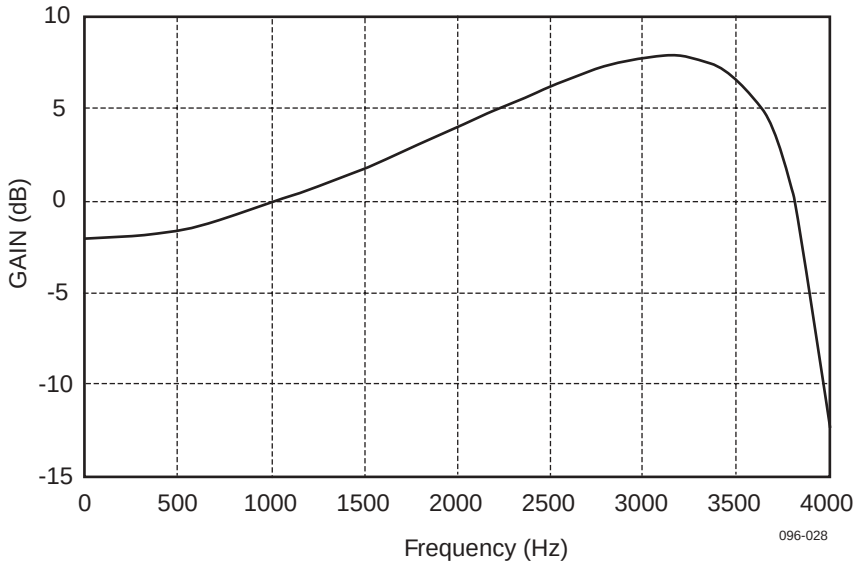


Figure 4-6 Slope Filter

The highpass filter, Figure 4-7, provides at least 30 dB of attenuation below 120 Hz.

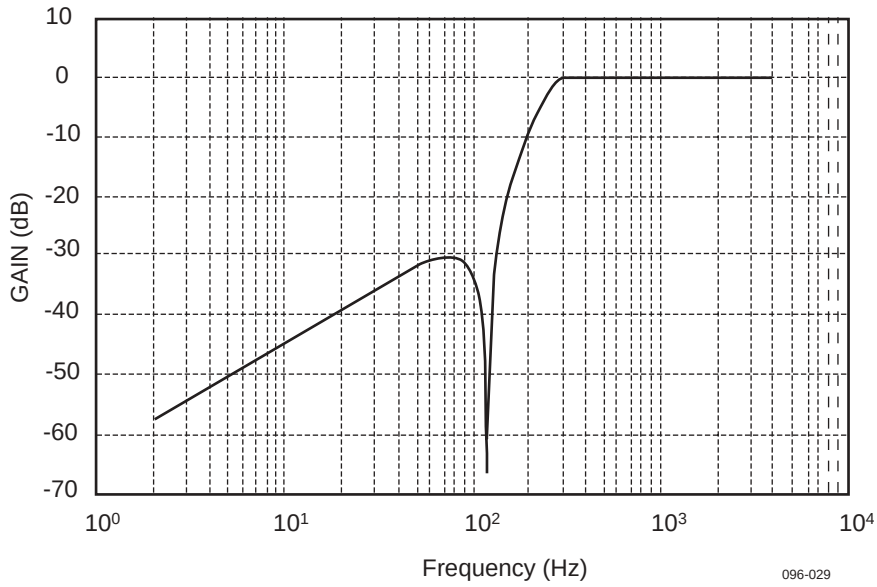


Figure 4-7 Highpass Filter

Both filters are individually enabled by the CODEC configuration command (TX_HPF_DIS_N and TX_SLOPE_FILT_DIS_N).

The Tx audio path contains a programmable gain stage, with a range of +12 dB to –84 dB, after the Analog-to-Digital conversion and prior to the QDSP4000. The QDSP4000 DMA parameter CodecTxGain sets the Tx gain. A programmed value of 0x4000 is unity gain and a programmed value of 0x0000 mutes the Tx audio data. The gain calculation for CodecTxGain is:

Gain = 20*LOG(CodecTxGain/16384)

4.4.2 Receive Path Processing

The MSM6100 device includes the capability of adding a portion of the Tx audio into the receive path. This sidetone is added with a programmable gain stage, with a range of 0 dB to –96 dB, controlled by the QDSP4000 DMA parameter CodecSTGain. A programmed value of 0x4000 is –12 dB of gain and a programmed value of 0x0000 mutes the sidetone. The gain calculation for CodecSTGain is:

Gain = 20*LOG(CodecSTGain /16384) –12 dB

A user selectable highpass filter is available for rejection of low frequency noise. This filter provides at least 30 dB of attenuation below 120 Hz. This filter is identical to the Tx highpass filter with the frequency response shown in Figure 4-8. Selection of this filter is accomplished by sending the CODEC configuration command (RX_HPF_DIS_N).

The Rx audio path contains a programmable gain stage, with a range of +15 dB to –81 dB, after the QDSP4000 and prior to the Digital-to-Analog conversion. The QDSP4000 DMA parameter CodecRxGain sets the Rx gain. A programmed value of 0x4000 is +3 dB of gain and a programmed value of 0x0000 mutes the Rx audio data. The gain calculation for CodecRxGain is:

Gain = 20*LOG(CodecRxGain /16384) +3 dB

The receive path is digitally filtered with an ITU G.712 compliant filter. The filter response has a flat passband out to 3400 Hz and offers attenuation of at least 14 dB at 3.98 kHz to allow adequate image rejection.

The receive path can be directed to earphone1 (EAR1OP, EAR1ON) or earphone2 (EAR2) amplifiers, or the auxiliary output for voice (AUXOP, AUXON). The outputs, earphone1 and Auxiliary out, are differential outputs. Earphone2 (EAR2O) and earphone3 are single-ended output stages designed to deliver stereo outputs. Selection of the active amplifier is accomplished by sending the QDSP4000 CODEC configuration command (AMP_SEL). The earphone amplifiers that are not selected are disabled and the output is in a high-Z state.

4.4.3 Microphone and Earphone Interface

The MSM6100 device's microphone and earphone is designed to interface directly to the handsets microphone and earphone or to the connector of a headset. Figure 4-8 illustrates a typical differential interface used in handset applications. The MICBIAS output pin is designed to provide 1 mA of current at 1.8 Volts DC. The output power for the differential EAR1 output is typically 70 mW for a full-scale +3 dBm0 sine wave into a 32 OHM speaker.

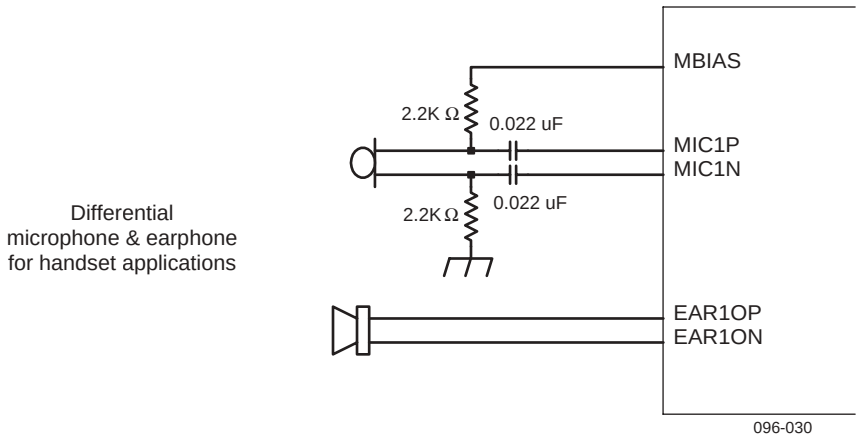


Figure 4-8 Typical Handset Interface

Figure 4-9 is an example of a single-ended microphone and earphone application found in typical headset applications. The output power for the single-ended EAR2 output is typically 10.8 mW for a full-scale +3 dBm0 sine wave into a 32 OHM speaker.

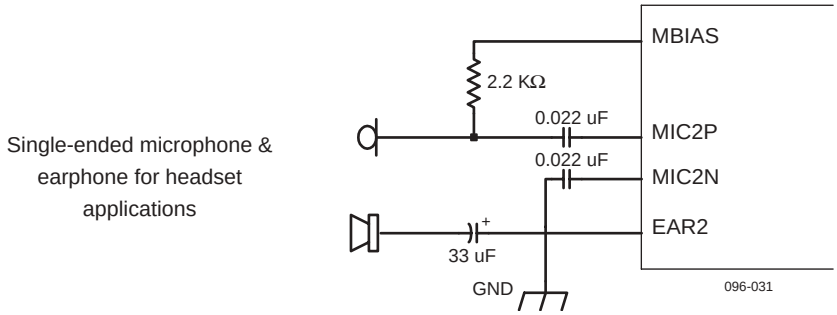


Figure 4-9 Typical Handset Application

4.4.4 Auxiliary I/O Interface

The MSM6100 device also provides an analog auxiliary interface for car-kit applications. The output power for the auxiliary output is typically 2.3 mW for a full-scale +3 dBm0 sine wave into a 600 OHM load.

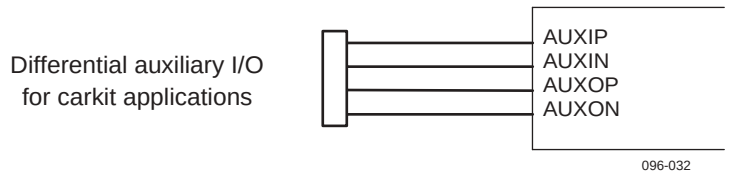


Figure 4-10 Typical Analog Car-kit Application

4.4.5 Digital PCM Interface

The Auxiliary PCM Interface enables communication with an external CODEC that provides the hardware support for hands free applications. Both mu-law and A-law CODECs are supported by the Aux PCM interface. The interface does not support linear CODECs.

The auxiliary CODEC port operates with standard long-sync timing and a 128 kHz clock. The AUX_PCM_SYNC runs at 8 kHz with 50% duty cycle. Most mu-law and A-law CODEC's support the 128 kHz AUX_PCM_CLK bit clock. The Aux CODEC port also supports 2.048 MHz PCM data and sync timing for A-law and mu-law CODECs that match the sync timing.

The ONES_DETECT state machine is clocked by CLKOUT of the internal ARM926EJ-S microprocessor. The logic is controlled by CODEC_CTL:ONES_POLARITY with a single output of WEB_MISC_RD:ONES_DETECT. In addition, this logic can also generate an AUX_PCM_DIN_INT interrupt. This interrupt is asserted when the value on the AUX_PCM_DIN pin matches that of the ONES_POLARITY bit. To enable or disable this interrupt, respectively set (1) or clear (0) AUX_PCM_DIN_INT_EN in either IRQ_MASK_0 or FIQ_MASK_0. If ONES_POLARITY=0, then a high on AUX_DIN guarantees ONES_DETECT=0. If ONES_POLARITY=1, then a low on AUX_DIN guarantees ONES_DETECT=1.

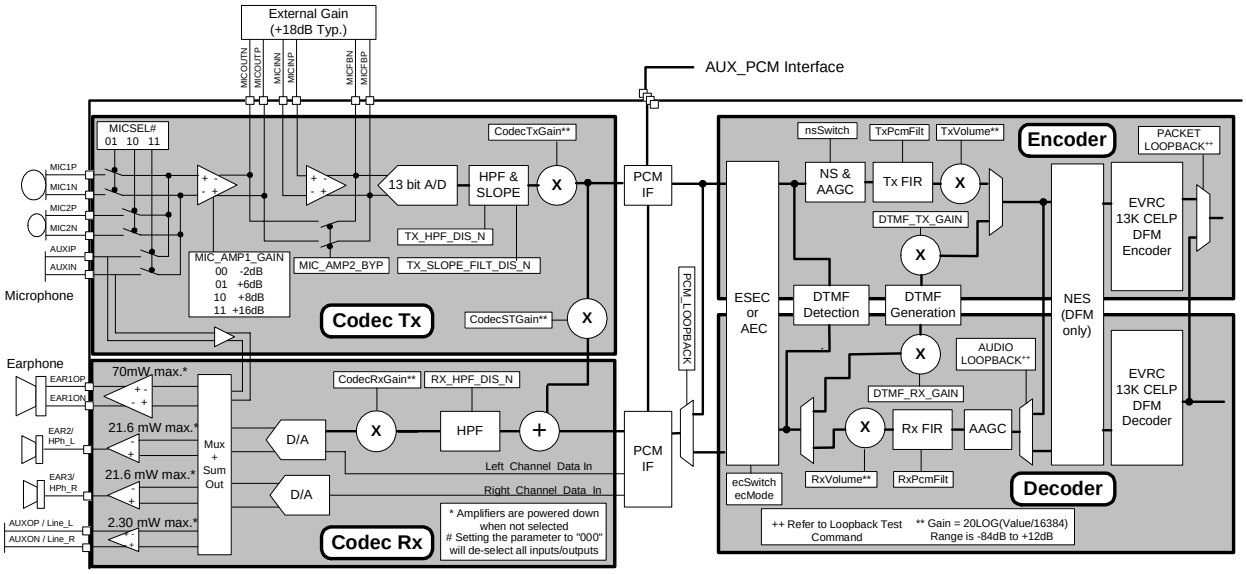


Figure 4-11 Auxiliary/Primary PCM Interface

The Primary PCM interface is supported on the Auxiliary PCM interface by clearing bit 4 in the CODEC_CTL register. The primary PCM interface is disabled at powerup or when RESIN_N asserts.

4.4.6 Primary PCM Operation

The PCM interfaces can be configured and controlled by either direct register access through the CODEC_CTL register, or by the QDSP4000 CODEC Configuration Command. Using the CODEC Configuration command is the preferred way of setting the PCM interface and this section will only discuss the PCM interface operation in terms of the CODEC Configuration Command.

The PCM_CLK pin, the clock for receive (PCM_DIN) and transmit (PCM_DOUT) CODEC PCM data, has a programmable direction defined by PCM_CLK_DIR. When clear (0), the PCM_CLK pin becomes an input (PCM_CLK comes from the PCM_CLK pin). When set (1), the PCM_CLK pin becomes an output (PCM_CLK comes from the clock and sync generator). The PCM_SYNC pin also has a programmable direction, defined by the PCM_SYNC_DIR in the same register. When clear (0), the PCM_SYNC pin becomes an input (PCM_SYNC comes from the PCM_SYNC pin; PCM_SYNC must be latched in). When set (1), the PCM_SYNC pin becomes an output (PCM_SYNC comes from the clock and sync generator). The polarity of PCM_CLK and PCM_SYNC can be inverted by setting (1) the PCM_CLK_SENSE. When PCM_CLK_SENSE is set (1), the CODEC interface latches the PCM_SYNC on the rising edge of the PCM_CLK.

4.5 ARM Microprocessor and Peripherals and AHB

The MSM6100 device contains an embedded ARM926EJ-S microprocessor and supporting peripherals. The support peripherals include the Memory and Peripheral Interface Controller, Sleep Controller, Clock Controller, Watchdog Controller, and Watchdog circuit.

The ARM926EJ-S used in the MSM6100 device is a high-performance, low-power microprocessor. Some of the features of the ARM microprocessor include a 3-stage pipelined RISC architecture, both 32-bit ARM and 16-bit THUMB instruction sets, a 32-bit address bus, and a 32-bit internal data bus. For more information on using the ARM926EJ-S please refer to the Advanced RISC Machines publication ARM926EJ-S Data Sheet (document number ARM DDI 0029E.) The ARM processor is configured for little-endian mode and supports a 16-bit external data bus.

4.5.1 AHB System

MSM6100 must support several types of voice, video, and data features. This requires significant processing power as well as memory access bandwidth. The MSM6100 AHB system is leveraged from the MSM 6050 AHB system with modifications to increase performance.

To reach the higher performance goals, MSM6100 uses the ARM926EJ-S microprocessor, which has a target speed of 150 MHz. The ARM926EJ-S brings increased processing power by using a Harvard Cache Architecture that provides a processor subsystem which includes ARM9EJ-S integer core, a Memory Management Unit (MMU), separate instruction and data AMBA AHB bus interfaces, separate instructions and data caches, and separate instruction and data tightly-coupled memory (TCM) interfaces. Not all the features of the ARM926EJ-S are used in MSM6100.

Also, the MSM6100 makes use of a Multi-Layer Advanced High-Performance Bus (AHB) system, which is expected to run at some divide ratio of the ARM processor operating frequency, with 75 MHz being the target speed. The Multi-Layer Bus Structure is required as the QDSP4 processors and new blocks on the system (e.g., GRP) require access to external memory and peripherals. Since we want to minimize any access bandwidth hit on the processor, two additional AHB buses were added to the system to parallelize accesses to system peripherals.

The main AHB bus (AMBA_AHB) is controlled by the microprocessor. In the case of the ARM926EJ-S, the bus will be shared between both data and instruction Bus Interface Units (BIU). The microprocessor is used on what is called the Single Layer AHB configuration (since data and instruction BIUs share a common AHB bus), so glue logic is added to the macro to take care of the BIU master arbitration.

A second AHB bus (DSP_AHB) exists to allow the aDSP and mDSP processors access to external peripherals by means of the DME interfaces. The DME interfaces are masters of the DSP AHB bus which can access external memories and peripherals.

A third AHB bus (GRP_AHB) is required by the Graphics Processor (GRP) added to MSM6100. The GRP is a master of its own AHB bus (GRP AHB) and also has access to the external memory and peripherals.

MSM6100 introduces a second memory interface to support parallel accesses to memory. This feature should help increase the memory access bandwidth. This is important, since it is planned to allow additional masters other than the ARM to have access to external devices. Thus, two memory interfaces are available in MSM6100 to support accesses to asynchronous NOR FLASH and SRAM, asynchronous page mode NOR FLASH, pseudo-RAM, burst mode FLASH, SDRAM and NAND FLASH. The use of page mode, burst mode, and SDRAM type of memories should help to increase the performance of the system.

4.5.1.1 MSM6100 Implementation

4.5.1.1.1 Requirements

- ARM926EJ-S integration
- ARM926EJ-S operating at a maximum frequency of 150 MHz
- Multi-Layer AHB System operating at a maximum frequency of 75 MHz
- Two external memory interfaces with arbitration for the Multi-Layer AHB system and memory controllers
- Support for external memory and peripherals access for the QDSP4 processors and the graphics processor
- Support for SDRAM and BURST mode flash
- Boot-up from NAND flash device
- Debugging capabilities at target speeds

4.5.1.1.2 Architecture

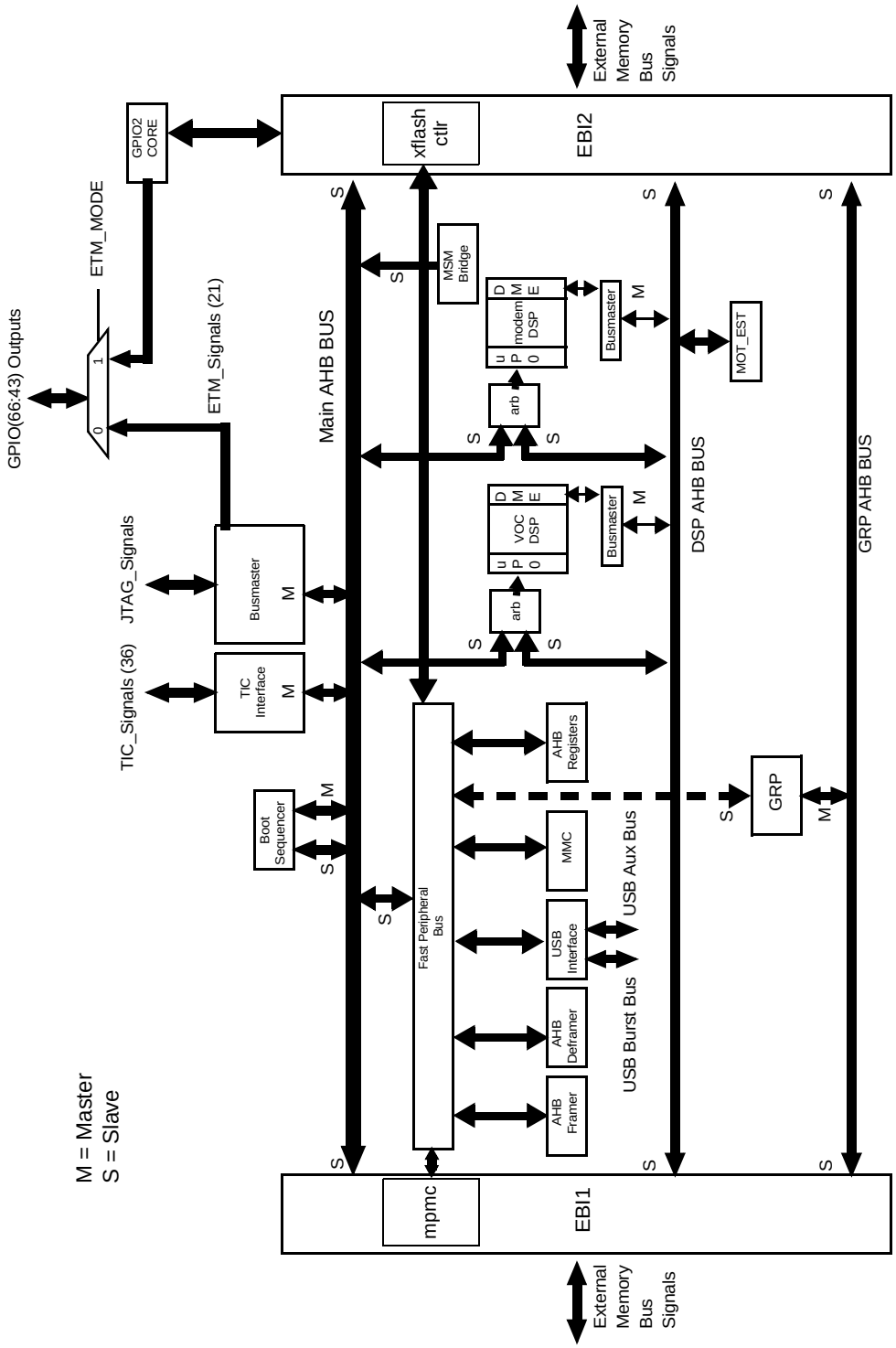


Figure 4-12 AHB System Architecture

4.5.2 ARM926EJ-S

4.5.2.1 Features

The ARM926EJ-S is a single-clock, high-performance, Java-enabled, synthesizable core. Its features include:

- Harvard caches
  - 16KB Instruction and 16KB Data caches, each 4-way associative supported on MSM6100
- Memory management unit (MMU)
- Harvard AHB interface
- ARM and THUMB modes supported
- Tightly coupled memory (TCM) is NOT supported on MSM6100
- Jazelle ARM9EJ-S core (Java support)
- Embedded Trace Macrocell (ETM) support
- Low Power Features

4.5.2.2 Burst Access

The ARM926EJ-S supports burst transfers to its slaves, specifically for cache operations. Table 4-1 shows the burst transfer types supported:

Table 4-1 ARM Burst Transfer Support

HBURST[2:0]	Description	Operation
SINGLE	Single transfer of word, half-word, or byte	■ Single transfer of word. ■ NC instruction fetches. ■ Page-table walk read. ■ Continuation of a burst that either lost grant or received a split/retry response.
INCR4	4-word incrementing burst	■ Half-line cache writeback. ■ Instruction prefetch (if enabled). ■ 4-word burst NCNB, NCB, WT, WB write.
INCR8	8-word incrementing burst	■ Full line cache writeback. ■ 8-word burst NCNB, NCB, WT, WB write.
WRAP8	8-word wrapping burst	■ Cache line fill.

Burst transfers coupled with the use of page mode, burst mode, and SDRAM memories help to increase system performance.

4.5.3 MAIN AHB Bus Masters

The MAIN AHB bus has 4 masters: TIC (Test Interface Controller), BOOT sequencer, ARM926EJ-S processor's data port, ARM926EJ-S processor's instruction port.

4.5.3.1 Features

- Access to all defined memory regions except Motion Estimator.
- An Arbiter is required for the 4 masters to access the bus. The masters with their priority from the highest to the lowest are: TIC, BOOT sequencer, ARM926EJ-S processor's data port, ARM926EJ-S processor's instruction port, with ARM926EJ-S processor's instruction port being the default master.

4.5.4 DSP AHB Bus Masters

The DSP AHB Bus has two masters: the DME interfaces of the QDSP4 processors.

4.5.4.1 Features

- Access to both EBI1 and EBI2 interfaces and Motion Estimator.
- An arbiter is required for the two DSPs to access the bus, which has a fixed priority mechanism with the ADSP being the lower priority master, but also the default one.

4.5.5 GRP AHB Bus Masters

The graphics processor (GRP) is a piece of dedicated hardware for graphics acceleration purposes. It requires access to external memories and external LCD devices. The GRP includes an AHB master to perform the required transfers to and from external devices. Since a significant bandwidth requirement is expected for the device, a dedicated AHB bus has been assigned to the GRP.

4.5.5.1 Features

- Access to both EBI1 and EBI2 interfaces
- INCR4 Burst transfer support

4.5.6 Memory Map and Memory Map Decoder

The memory map indicates how the different slaves are assigned memory address ranges on the system. The memory map decoder or decoder in an AMBA system is used to perform a centralized address decoding function. This decoding generates select lines to each of the system bus slaves, indicating that a read or write access to the slave is required.

The MSM6100 allows booting from the ROM1(0) memory (NOR Flash) or from the internal Boot SRAM used for when booting from NAND Flash. Also, the ARM926EJ-S processor allows placement of the exception vectors at 0XFFFF0000, in addition to the traditional 0x00000000 location. RAM1(0) memory can be used to run code and store the exception table (usually when SDRAM is connected to it). Based on the BOOT_MODE pin value, which indicates whether the chip should boot from NAND or NOR flash, the decoder maps the memory regions differently. When booting from NOR flash, the 2 ROM1 regions (ROM1(0) and ROM1(1)) are mapped to the lowest addresses, so that a NOR flash can be used at ROM1(0) from which the ARM can directly execute when the chip is released from reset. When booting from NAND flash, the 2 RAM1 regions (RAM1(0) and RAM1(1)) are mapped to the lowest addresses, so that an SDRAM can be used at RAM1(0) to which the ARM processor can move the entire DMSS software from the NAND flash. The decoder must properly decode accesses under the different configurations allowed by these options.

4.5.6.1 MSM6100 Memory Map

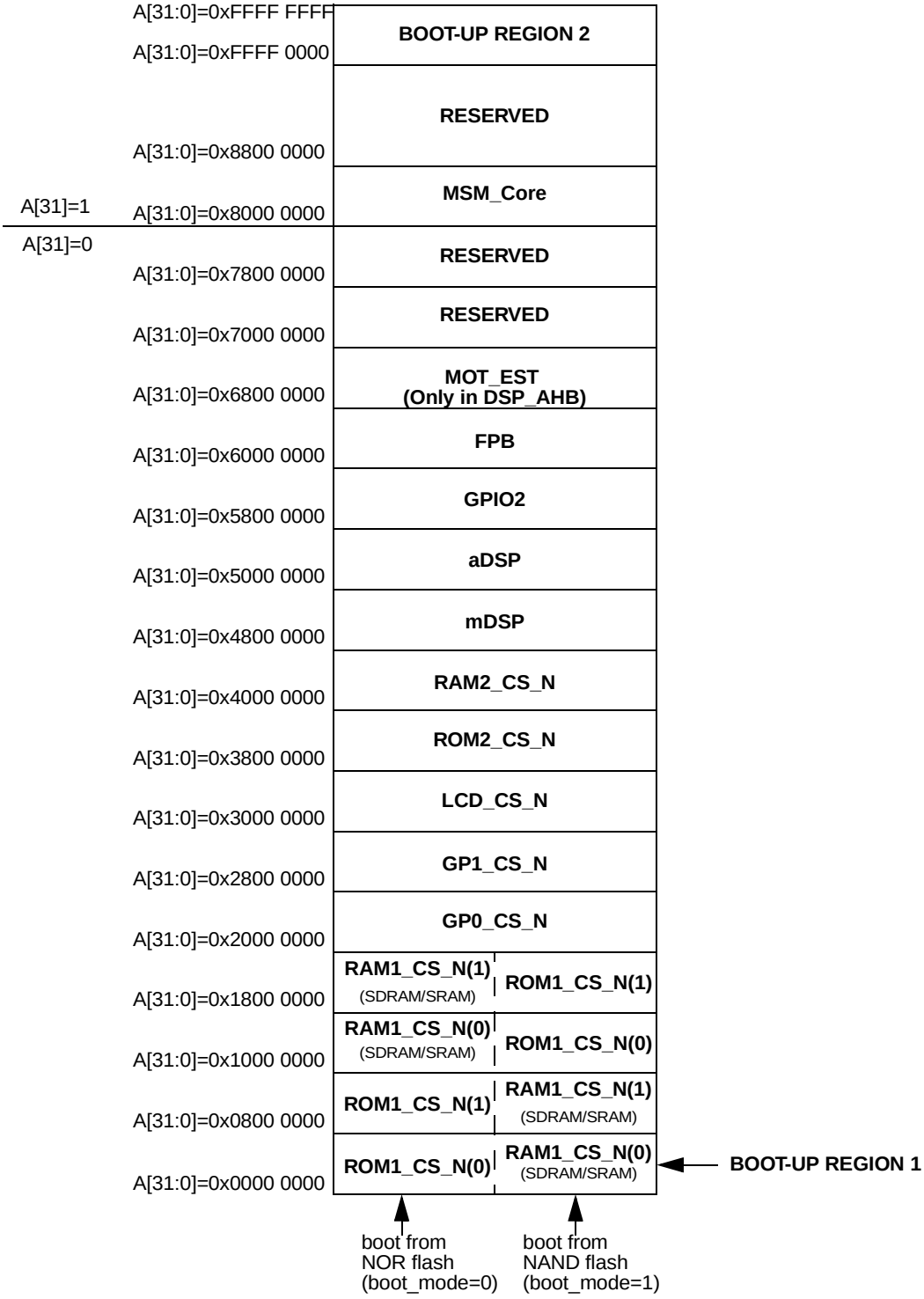


Figure 4-13 MSM6100 Memory Map

Table 4-2 MSM6100 Decoder HSEL Signals Description

HSEL Signal	Address Range (boot_mode)	Description	SIze
HSEL_ROM1_0	0x0000 0000 - 0x07FF FFFF(NOR) 0x1000 0000 - 0x17FF FFFF(NAND)	Select for EBI1 ROM1(0) memory. (boot-up region 1)	128 MBytes
HSEL_ROM1_1	0x0800 0000 - 0x0FFF FFFF(NOR) 0x1800 0000 - 0x1FFF FFFF(NAND)	Select for EBI1 ROM1(1) memory.	128 MBytes
HSEL_RAM1_0	0x1000 0000 - 0x17FF FFFF(NOR) 0x0000 0000 - 0x07FF FFFF(NAND)	Select for EBI1 RAM1(0) memory.	128 MBytes
HSEL_RAM1_1	0x1800 0000 - 0x1FFF FFFF(NOR) 0x0800 0000 - 0x0FFF FFFF(NAND)	Select for EBI1 RAM1(1) memory.	128 MBytes
HSEL_GP0	0x2000 0000 - 0x27FF FFFF	Select for EBI2 general purpose peripheral 0.	128 MBytes
HSEL_GP1	0x2800 0000 - 0x2FFF FFFF	Select for EBI2 general purpose peripheral 1.	128 MBytes
HSEL_LCD	0x3000 0000 - 0x37FF FFFF	Select for EBI2 LCD memory.	128 MBytes
HSEL_ROM2	0x3800 0000 - 0x3FFF FFFF	Select for EBI2 ROM2 memory.	128 MBytes
HSEL_RAM2	0x4000 0000 - 0x47FF FFFF	Select for EBI2 RAM2 memory.	128 MBytes
HSEL_mDSP	0x4800 0000 - 0x4FFF FFFF	Select for modem QDSP4 (memories and registers).	128 MBytes
HSEL_aDSP	0x5000 0000 - 0x57FF FFFF	Select for applications QDSP4 (memories and registers).	128 MBytes
HSEL_GPIO2	0x5800 0000 - 0x5FFF FFFF	Select for GPIO block 2 (the one implementing the gpio's used on ETM).	128 MBytes
HSEL_FPB	0x6000 0000 - 0x67FF FFFF	Select for FPB peripherals. Further decoding is done by this slave interface. FPB peripherals include: ■ AHB Registers ■ Framer ■ Deframer ■ MMC ■ USB ■ GRP ■ MPM Controller ■ NAND Controller The FPB Section on this chapter describes the FPB decoding.	128 MBytes
HSEL_MOT_EST	0x6800 0000 - 0x6FFF FFFF	Select for Motion Estimator (Only accessed by DSP_AHB).	128 MBytes
RESERVED	0x7000 0000 - 0x77FF FFFF	Reserved.	128 MBytes
RESERVED	0x7800 0000 - 0x7FFF FFFF	Reserved.	128 MBytes
HSEL_MSM	0x8000 0000 - 0x87FF FFFF	Select for MSM bus peripherals.	128 MBytes
RESERVED	0x8800 0000 - 0xFFFE FFFF	Undefined.	~1920MBytes
HSEL_BOOT2	0xFFFF0000 - 0xFFFFFFFF	Select for on-chip boot SRAM. (boot-up region 2)	64 KBytes

4.5.7 Fast Peripheral Bridge

The number of AHB slaves on the MSM6100 chip would constrain the main bus speed, so we introduced a Fast peripheral Bridge (FPB) to off load the main AHB bus. This bridge is intended for slaves clocked by the AHB clock. In other words, only fast peripherals are supposed to connect to this bridge. Having a FPB connect to these peripherals also saves power, since the FBPB peripherals do not see the main AHB bus activity (i.e., accesses to external memory and to the MSM bus).

The FPB is a slave of the main AHB bus and a master to the slaves attached to it. The FPB is AMBA AHB compliant so that any AMBA compliant slave can attach to it. MSM6100 allocates 128 MB address space to the FPB. This memory space is evenly divided into 16 regions so that the address space for each FPB slave is 8 MB. In MSM6100, there are 8 slaves behind the FPB, therefore 8 of the address regions are unused. These slaves are:

- Framer
- Deframer
- USB
- Graphic Processor (GRP)
- AHB Registers
- MMC
- MPM Controller
- NAND Controller

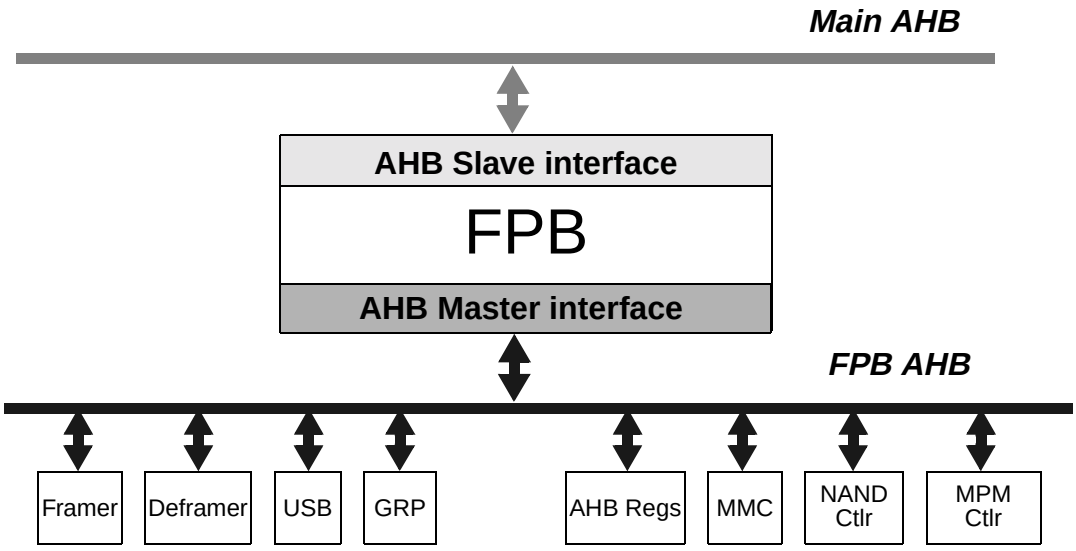


Figure 4-14 Fast Peripheral Bridge

4.5.7.1 Requirements

- AMBA AHB compliant slave and master interface
- Off load the main AHB bus as much as possible
- Minimize the additional wait state

4.5.7.2 FPB Address Decoding

The FPB performs further address decoding for the slaves connected to it. The whole address space designated to the FPB (128 MBytes) is evenly divided into 16 sub-regions of 8MBytes each. In MSM6100, 9 sub-regions are used and are defined as follows.

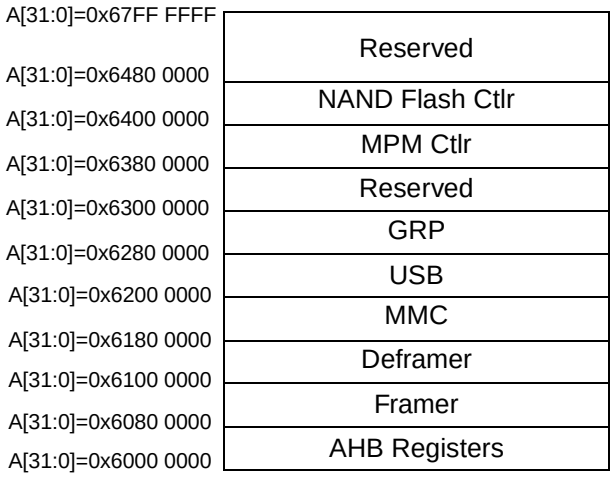


Figure 4-15 Fast Peripheral Bus Memory Region Map

4.5.8 ARM Clock and Power Management

The MSM6100 device provides three ways of managing clock and power for the ARM microprocessor. The MSM6100 device has a powerdown control unit, a clock scale control unit, and a clock source select circuit.

The registers that are used for clock and power management are MICRO_CLK_SOURCE0, MICRO_CLK_SOURCE1, SWITCH_CLK, MICRO_CLK_OFF, and MICRO_CLK_DIV.

Entry to powerdown mode is specified by two successive writes to the powerdown bit of the MICRO_CLK_OFF register (write a 1 followed by a 0). This prevents accidental entry to powerdown mode. Upon entering powerdown the clock to the microprocessor is gated off and the ARM retains its state. The microprocessor remains in powerdown state until an unmasked interrupt is asserted on nIRQ or nFIQ.

Wakeup is initiated by any unmasked interrupt to the microprocessor (nFIQ, nIRQ) from the interrupt controller.

On initial powerup, RESIN_N is asserted. The microprocessor starts on deassertion of RESOUT_N. On initial powerup, RESIN_N must be of sufficient duration for the external VCTCXO to stabilize. The same is true for the Watchdog timeout duration.

The clock scale unit allows software to change the clock rate of the processor. Clock rates can be scaled down by factors of 1, 2, 3, 4, 8, 16, 32, or 64 to allow for reduced power consumption. When exiting power down, the clock rate specified by MICRO_CLK_DIV is used. After reset, a divide-by-1 rate is used. The clock rate is not changed upon an interrupt assertion. Therefore clock rates must not be set to a low rate when waiting for a critical interrupt. Setting the clock scale unit for slow clock rates results in long latencies for the interrupt service routines.

In slotted-paging mode, software shuts down the TCXO when the MSM6100 device enters SLEEP mode. During the SLEEP interval, the SLEEP oscillator remains active and drives the MSM6100 device's internal SLEEP controller. The microprocessor is restarted at the end of the SLEEP interval as the phone wakes up.

4.5.9 Reset and Pause

The Reset Pause block is responsible for:

- Generating a synchronous reset for the ARM as well as the other AMBA (Advanced Microcontroller Bus Architecture) peripherals
- Providing status information concerning the source of the last reset (pin or watchdog generated)
- Providing a software-controllable mechanism for stalling the ARM for a specified amount of time

This block generates reset signals for the ARM and the system. It also controls the clock to the ARM microprocessor providing the ability to reset or pause the ARM. Software can use the pause mechanism to halt the microprocessor for a specified amount of time.

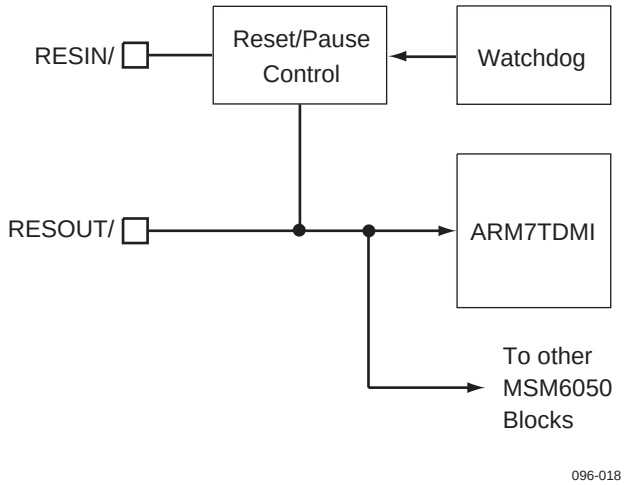


Figure 4-16 Reset Generation

This block combines the two reset signals, power-on reset (RESIN_N) and watchdog reset, and provides a synchronously deasserting reset pulse to the rest of the chip. HRESET_N is guaranteed to be asserted for a minimum of 2 CLK pulses regardless of the duration of the RESIN_N pulse. The source of the last reset is stored in bit 0 of register RESET_STATUS.

This block also contains the circuits to pause the ARM. This can be done by deasserting HREADY which stalls the microprocessor until HREADY asserts.

Software pauses the processor by writing to a 16-bit counter through register PAUSE_TIMER. This action immediately deasserts HREADY for 4 cycles. ARM is paused for 4 + PAUSE_TIMER periods. For example, if TCXO is a 19.2 MHz clock (52 ns period), then writing the value 10 to PAUSE_TIMER pauses the microprocessor for 52 ns*4 + 10*52ns = 728 μs.

This timer is meant for introducing small delays in the system software. Note that if the microprocessor clock frequency is sufficiently low, delays introduced by this timer can exceed the watchdog timer duration resulting in a watchdog reset of the system.

4.5.10 Watchdog Timer

The watchdog timer is a 21-bit counter running on the sleep controller clock regime that enables the Mobile Station to recover from unexpected hardware or software anomalies. Unless the microprocessor periodically resets the watchdog timer, the watchdog timer resets the Mobile Station. The watchdog timer is disabled and reset by grounding the WDOG_EN pin. Asserting the RESIN_N pin also resets the watchdog timer. The watchdog timer is enabled by leaving the WDOG_EN pin unconnected (it has an internal pull-up) or by connecting it to V_{DD_P}. The watchdog timer is disabled by hardware as soon as the ARM processor enters debug mode (indicated by the DBGACK pin of the processor transitioning from low to high). To re-enable the watchdog timer, apply a system reset to clear the state of the internal signal wdog_disable.

The watchdog circuit pulses the signal WATCHDOG_EXPIRED when the watchdog timer has expired. In NATIVE mode, this signal is combined with the RESIN_N pin to generate RESOUT_N and the internal reset for the MSM device.

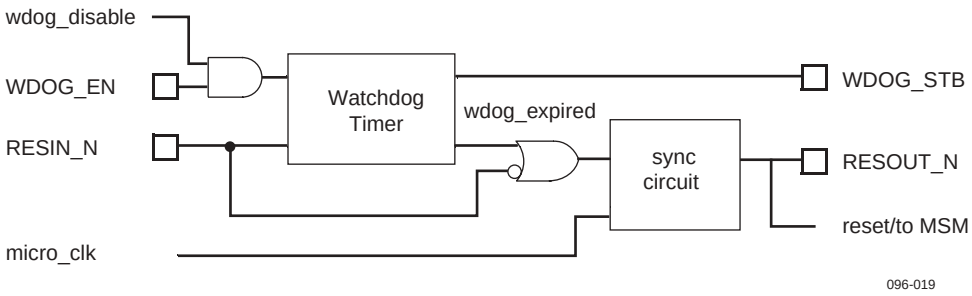


Figure 4-17 Watchdog Timer Configuration

4.5.10.1 Sleep Mode

In sleep mode, the microprocessor cannot reset the watchdog timer. The sleep controller contains an auto-kicker that resets the Watchdog Timer every 1024 TCXO/4 clock cycles (~200 μs) when SLEEP_XTAL_EN = 0. When the sleep crystal is used (SLEEP_XTAL_EN =1), the auto-kicker resets the watchdog every 32 sleep clock cycles. The output of the sleep controller auto-kicker is also routed to output via the WDOGSTB pin. To enable the auto-kicker, the microprocessor must write a sequence of 1, 0 to the SLEEP_CTL:AUTO_KICK_ARM bit. When the sleep counter reaches its terminal count, the auto-kicker circuit is disabled.

4.5.10.2 Non-sleep Mode

In non-sleep mode, the microprocessor must reset the watchdog timer at least once every 213 ms. If the microprocessor does not reset the watchdog timer in this time, the watchdog timer expires and asserts an internal RESIN_N signal to the system. The WDOG_EXPIRED signal lasts between 53.3 ms to 106 ms (53.3 ms for a 4.8 MHz clock rate and 53.3 to 106 ms when the sleep crystal is used). To reset the watchdog timer, the microprocessor must write a sequence of 1, 0, 1 to the SLEEP_CTL:WATCH_DOG bit.

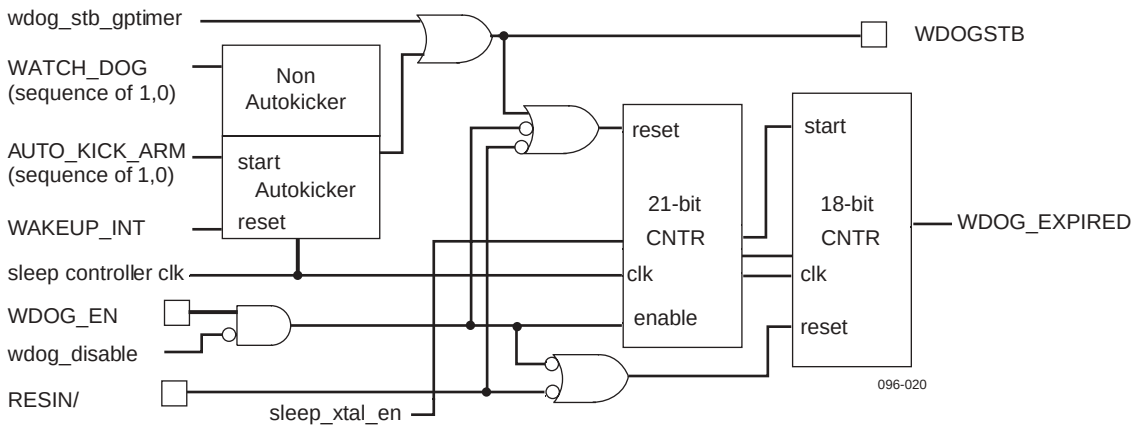


Figure 4-18 Watchdog Timer Block Diagram

4.5.10.3 General-purpose Timer Operation

When the microprocessor is powered down after programming a general-purpose timer delay that exceeds the watchdog timer interval, the general-purpose timer autokicker must be enabled to prevent the watchdog from resetting the ARM microprocessor. The autokicker is disabled when the general-purpose timer reaches its terminal count.

4.5.11 MSM6100 Reset Scheme Summary

Figure 4-19 depicts the reset scheme of MSM6100.

- The sync logic synchronizes the trailing edge of resin_n to the HCLK. This logic resides in the reset_pause block in the amba_ahb.
- When the boot_mode pin is 0, the resin_n pin can directly reset the ARM processor.
- When the boot_mode pin is 1, the ARM processor's reset is under the control of the boot sequencer. During the time the boot sequencer holds ARM in reset, it also keeps the watchdog disabled.
- Both the resin_n pin and the watchdog (wdog_expire signal) can cause the MSM to reset.
- When the mode[1] pin is 1 (test mode), the ARM processor is held in reset.

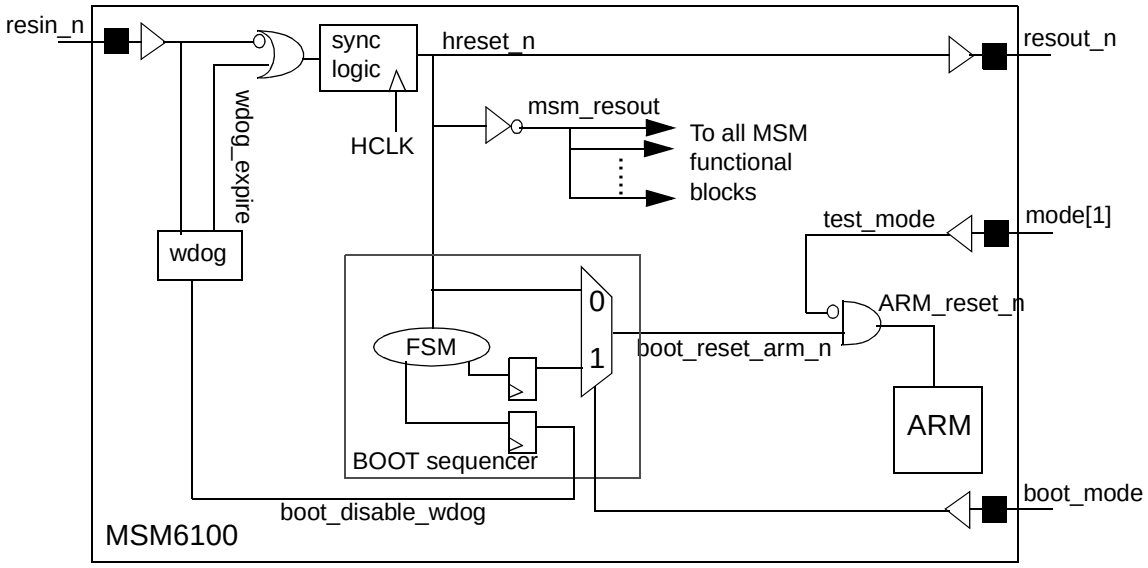


Figure 4-19 MSM6100 Reset Scheme

4.5.12 Interrupt Controller

The interrupt controller has a two-level hierarchy: level 1 and level 2. Some of the interrupt signals are connected to the level 1 interrupt controller directly while others are connected to the level 2 interrupt controllers, whose outputs are then fed to the level 1 interrupt controller. There is only one level 1 interrupt controller. The level 1 interrupt controller is located within the WEB module whose outputs, IRQ and FIQ, are fed to the ARM9 processor. Multiple Level 2 interrupt controllers are used and distributed into modules where the interrupts are generated. One exception is that one of the GPIO (GPIO 0:38, 67:81) level 2 interrupt controllers is located within the same file as the level 1 interrupt controller. This section describes only the level 1 interrupt controller and the level 2 interrupt controller for GPIO group 1 (GPIO 0:38, 67:81) interrupts. Figure 4-20 is the block diagram of the level 1 interrupt controller and GPIO group 1's level 2 interrupt controller.

The level 1 interrupt controller is made up of multiple interrupt bit slices, one for each interrupt input. Each level 1 interrupt bit slice consists of one interrupt status register and two AND gates. The interrupt status register is set asynchronously whenever the interrupt signal is active and cleared synchronously when the CLEAR signal is active. Notice that the interrupt input has higher priority than the clear signal. Therefore, this register can't be cleared unless the interrupt is cleared. The output of this register is ANDed with IRQ and FIQ interrupt enable signals to generated the masked-irq-interrupt and masked-fiq-interrupt signals, respectively (see Figure 4-22). The outputs of all the interrupt slices are ORed together to generate the IRQ and FIQ signal (see Figure 4-20). The output of the interrupt status register can be read by the ARM9 processor directly and the interrupt clear, and IRQ and FIQ interrupt enables are controlled by ARM9 on per slice bases.

Each level 2 interrupt controller is unique to each module. The GPIO group 2 level interrupt controller is very similar in design as the level 1 interrupt controller. It consists of multiple interrupt bit slices, one for each interrupt input. Each bit slice consists of one interrupt status register and a mask AND gate (see Figure 4-21). The interrupt status register is set asynchronously whenever the interrupt signal is active and cleared synchronously when the CLEAR signal is active. The output of this register is ANDed with the GPIO interrupt enable signal to form the masked interrupt signal. The outputs of all the bit slices are ORed together to generate the gpio_group1_interrupt that is fed to the level 1 interrupt controller (see Figure 4-20). The output of the interrupt status register can be read by the ARM9 process directly and the interrupt clear, and IRQ and FIQ interrupt enables are controlled by ARM9 on per slice bases. The GPIO interrupt polarity and edge/level trigger is also programmable by ARM9.

4.5.12.1 Architecture

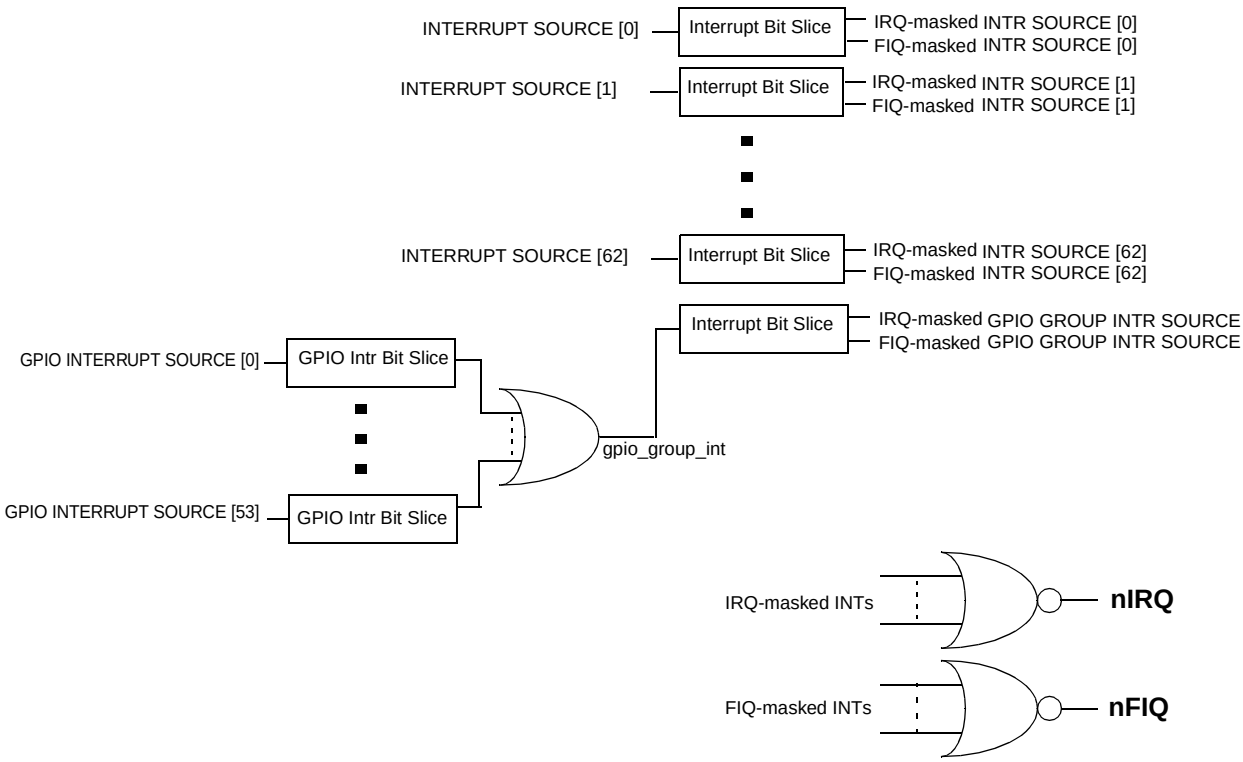


Figure 4-20 Interrupt Controller

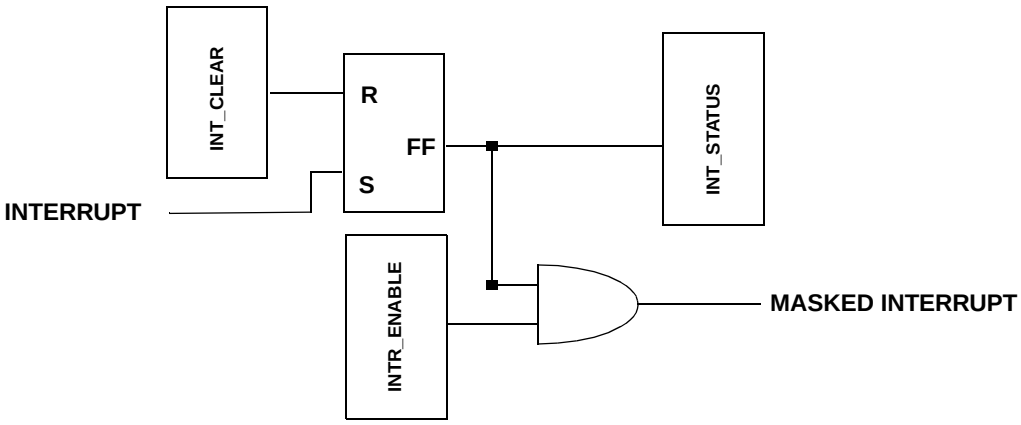


Figure 4-21 GPIO Group 1 Level 2 Interrupt Controller Bit Slice

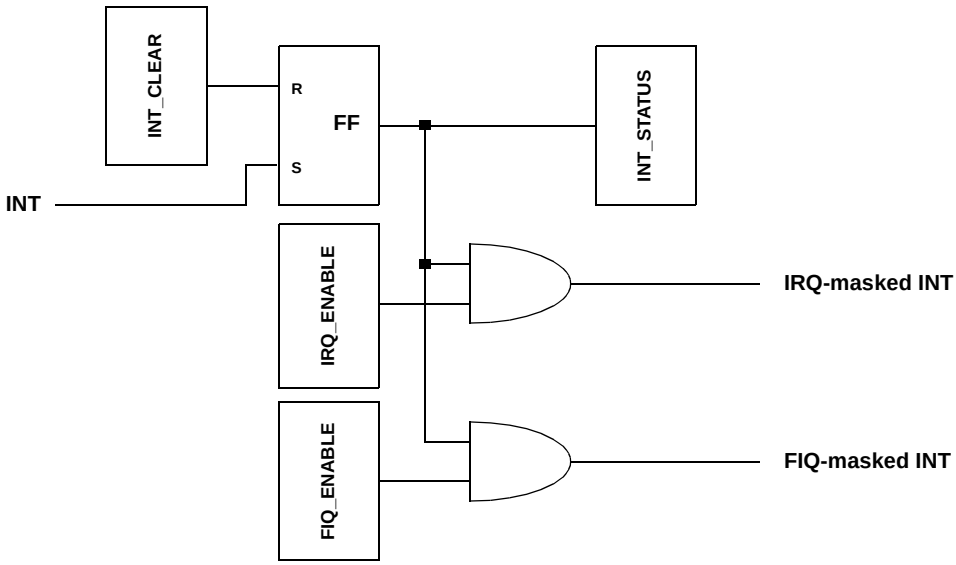


Figure 4-22 Level 1 Interrupt Bit Slice Design

4.5.12.2 Interrupt Sources

There are 64 level 1 interrupt sources, as listed in the following table:

Bit	Name
1	RTC_ROLL_INT
2	OS_TIME_TICK3_INT
3	MDSP_INT[2]

Bit	Name
4	MDSP_INT[1]
5	MDSP_INT[0]
6	SAMP_RAM_INT
7	RXCHIPX8_ENABLE_STB
8	AUX_CODEC_ONE_INT
9	DEM_SEARCH_DMA_REQ
10	DEMOD_INT
11	DEM_SEARCHER_DONE_INT
12	DEC_DONE_STROBE
13	TX_FR_INT
14	TX_1_25MS_INT
15	DFM_TX_WBD_INT
16	DFM_RX_WBD_INT
17	FM_RF_SLEEP_INT
18	GSM_MICRO_IRQ
19	GSM_GSAC0_DONE_IRQ
20	GSM_GSAC1_DONE_IRQ
21	GSM_TIME_INT2
22	GSM_TIME_INT1
23	GSM_SLEEP_FEE_INT
24	GSM_WAKEUP_INT
25	SLPCTL_TIME_TICK2_INT
26	SLPCTL_SLEEP_FEE_INT
27	SLPCTL_WAKEUP_INT
28	TIME_INT2
29	TIME_INT1
30	TIME_TICK_INT
31	26MS_INT
32	GPTIMER_INT
33	GPIO_GROUP_INT

Bit	Name
34	GPIO2_GROUP_INT
35	RESERVED_1[0]
36	RESERVED_1[1]
37	RESERVED_1[2]
38	RESERVED_1[3]
39	RESERVED_1[4]
40	RESERVED_1[5]
41	RESERVED_1[6]
42	RESERVED_1[7]
43	RESERVED_1[8]
44	BT_WAKEUP
45	ADSP_INT[2]
46	ADSP_INT[1]
47	ADSP_INT[0]
48	MDDI_INT
49	GRP_INT
50	NAND_WR_ER_DONE_INT
51	NAND_OP_DONE_INT
52	MMCC_ERR_INT
53	MMCC_INT
54	KEYSENSE_GROUP_INT
55	USB_HDR_INT
56	USB_INT
57	SBI_INT
58	I2CC_INT
59	UART3_DP_RX_DATA_INT
60	UART2_DP_RX_DATA_INT
61	DP_RX_DATA_INT
62	UART3_INT

Bit	Name
63	UART2_INT
64	UART_INTR

There are 54 interrupt sources to the GPIO group 2 level 2 interrupt controller.

Bit	Name	Description
54:1	GPIO_INT[31:0]	GPIO0 to GPIO38, and GPIO67 to GPIO81 interrupts

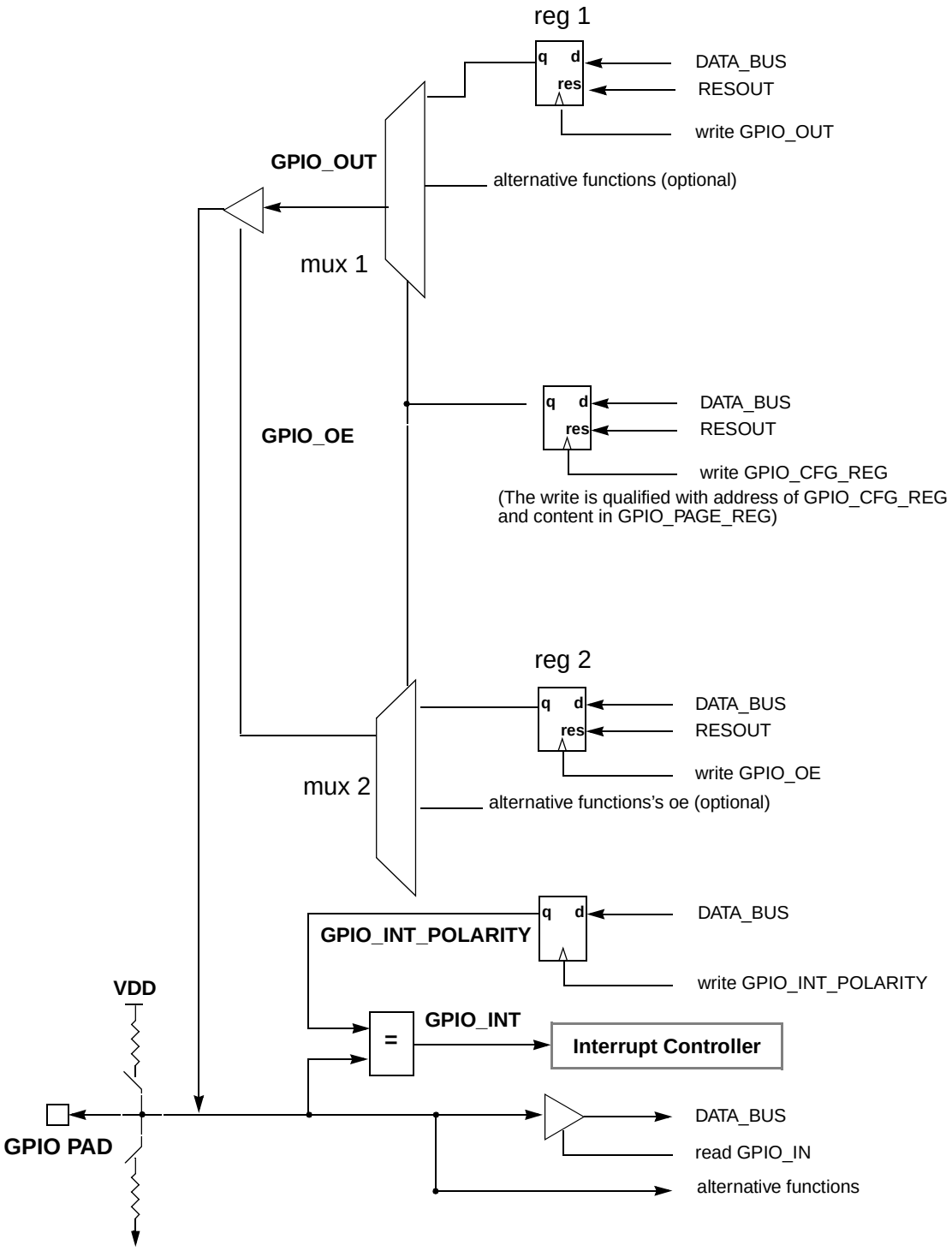


Figure 4-23 GPIO Structure

4.6 Boot Methodology

4.6.1 Overview

Traditionally, in MSM chips upon power-on-reset the microprocessor starts executing the boot code directly from the NOR flash. While booting from the NOR flash is still supported, MSM6100 now supports booting from the NAND flash. However, since the NAND flash is not suitable for random location accesses, extra hardware was added to accommodate booting up from the NAND flash. A boot_mode input pin is used to indicate to the MSM hardware whether to boot from the NOR flash or the NAND flash.

The recommended configuration for booting from the NAND flash is to attach a NAND flash device to the EBI2, and an SDRAM device to the EBI1. The NAND flash device contains one block (8K bytes) of boot code in the first 16 pages, and the entire DMSS software. After power-on reset, the MSM hardware (i.e., a boot sequencer) automatically loads the boot code from NAND flash to an on-chip boot SRAM, and then releases ARM to execute from this boot SRAM. By executing the boot code, the ARM processor transfers the entire DMSS software to the SDRAM, and then branches to the SDRAM to execute the DMSS and completes the boot-up process.

For booting from the NOR flash, simply attach a NOR flash, which contains the entire DMSS, to the lowest address region (ROM1_CS_N0) of EBI1.

4.6.2 Pin Requirements

To support MSM6100's boot-up scheme, three pins are required: BOOT_MODE, RESIN_N and DP_TX_DATA. The BOOT_MODE pin determines whether the MSM should boot up from the NOR flash (0) or the NAND flash (1). It is a dedicated pin and its value is not allowed to change while the MSM is active. The RESIN_N pin is the power-on reset pin of the MSM. A synchronizer synchronizes the "deassertion" of RESIN_N to HCLK to create the HRESET_N signal. The HRESET_N signal is then used to reset the internal logic of the MSM, except maybe the ARM processor. When booting from the NOR flash, HRESET_N directly resets the ARM. When booting from the NAND flash, the reset of the ARM is controlled by the boot sequencer. The DP_TX_DATA pin is shared by the UART and the boot sequencer. When booting from NOR flash, the UART uses this pin as a data output pin. When booting from NAND flash, the boot sequencer uses this pin to report the status of boot-up to the outside world, until software switches the DP_TX_DATA pin to be used by UART by writing a 0 to bit 11 (UART_NAND_SEL) of the WEB_MISC_WR register. (This register bit powers on to be 1.)

Table 4-3 Pin Requirements

Pin name	Direction	Value	Functions
BOOT_MODE	IN	'0'	1. Selects to boot from the NOR flash on EBI1. 2. Forces the ARM926 processor to start executing code from location 0x00000000, which is aliased to the NOR flash. 3. Forces the DP_TX_DATA pin to be used by the UART block as the data output pin.
		'1'	1. Selects to boot from the NAND flash on EBI2. 2. Forces the ARM926 processor to start executing code from location 0xFFFF0000, which is aliased to the on-chip boot SRAM. 3. Forces the DP_TX_DATA pin to be used by the boot sequencer to report boot-up status right after power-on reset. Software can later program the DP_TX_DATA pin to be used by UART after the boot-up process completes.
RESIN_N	IN	active low	When asserted (='0'), it puts the MSM hardware in power-on reset (boot) mode. The deassertion of this signal initiates the boot-up process.
DP_TX_DATA	OUT	'0'	When this pin remains at '0' (not toggling), it indicates that the boot sequencer is still in the process of loading the first block of data (ARM's boot code) from the NAND flash, and has not encountered an error condition yet. This is also the power-on value of this pin if the boot_mode pin is '1'.
		'1'	When this pin remains at '1' (not toggling), it indicates that the boot sequencer has finished loading the first block of data (ARM's boot code) from the NAND flash to the on-chip boot SRAM without errors and ARM has been released from reset and the watchdog released from disable.
		HCLK/2	When this pin toggles at HCLK/2 frequency, it indicates that the boot sequencer has aborted its operation due to a page-read error condition detected by the NAND flash controller right after reading a page. Usually it is due to uncorrectable ECC errors in the page. It also indicates that the ARM has been released from reset and the watchdog released from disable.
		HCLK/4	When this pin toggles at HCLK/4 frequency, it indicates that the boot sequencer has aborted its operation due to receiving an AHB ERROR response when accessing the NAND flash controller. It also indicates that the ARM has been released from reset and the watchdog released from disable.
		HCLK/8	When this pin toggles at HCLK/8 frequency, it indicates that the boot sequencer has aborted its operation due to not receiving the "op_done_irq" interrupt signal from the NAND flash controller for 8K TCXO cycles after the boot sequencer activates a NAND flash device reset or a page read. For TCXO running at 20 MHz, 8K cycles takes 409550 ns. It also indicates that the ARM has been released from reset and the watchdog released from disable.

4.6.3 Hardware Support

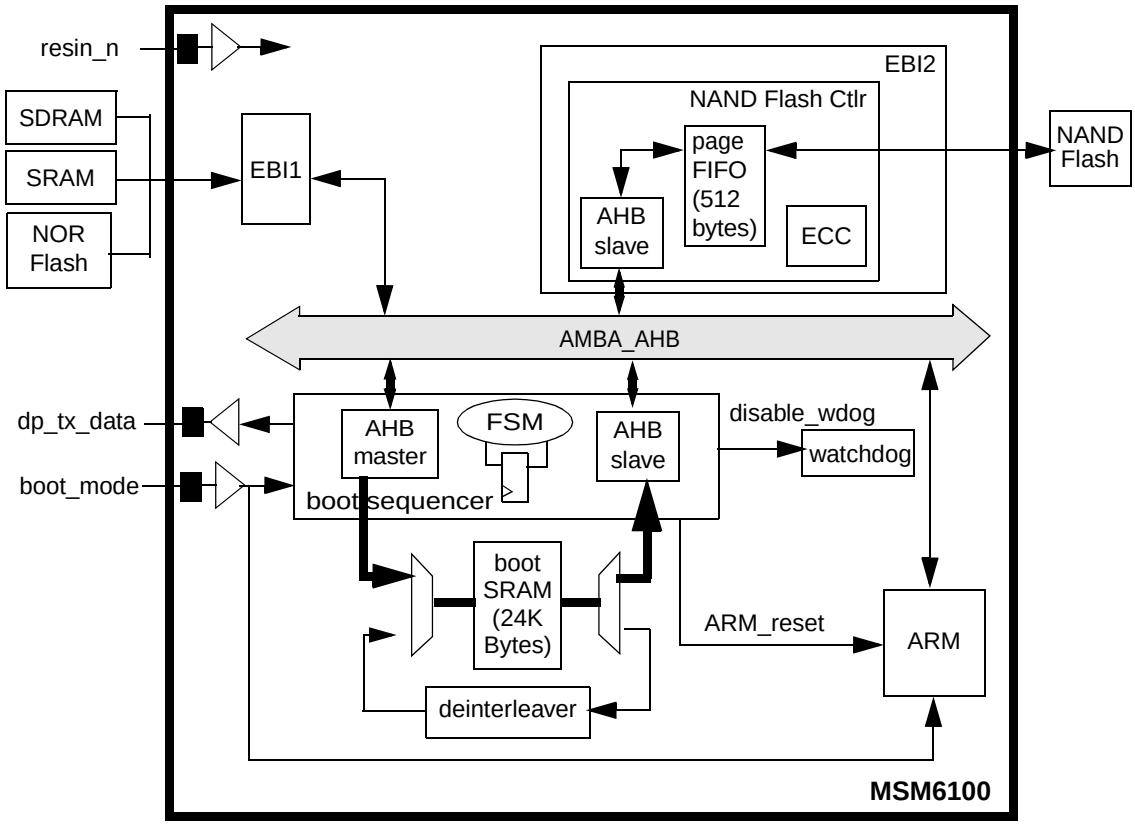


Figure 4-24 Hardware Support for MSM Boot-up

4.6.4 The Sequence of Events of Booting From the NOR Flash

1. When the power-on-reset (RESIN_N) is asserted, the hardware enters and remains in its boot mode, where the boot sequencer has control of the boot SRAM.
2. Upon the release of power-on-reset, if the BOOT_MODE pin has value ‘0’ (selects to boot from the NOR flash), then the boot sequencer initiates the boot-up process by releasing the ARM926 processor from reset immediately. The ARM926 processor then starts executing the boot code from location 0x00000000, which is aliased to be the boot code in the NOR flash.
3. By executing the boot code, the ARM926 processor initializes the entire chip for functional operations. The ARM926 processor can program the MODEM_RAM_SELECT register bits 1:0 to release the 24K-byte SRAM to the deinterleaver, MDSP or SRCH4, as well as program bit 5 to be 0 to disable the decoding of address 0xFFFF0000.
4. From this point on, the ARM926 processor continues to execute from the NOR flash until the power-on-reset is asserted again.

4.6.5 The Sequence of Events of Booting From the NAND Flash

1. When the power-on-reset (RESIN_N) is asserted, the hardware enters and remains in its boot mode, where the boot sequencer has control of the boot SRAM.
2. Upon the release of power-on-reset, if the BOOT_MODE pin has value ‘1’ (selects to boot from the NAND flash), then the boot sequencer wakes up and activates the NAND flash controller to fetch the first block (16 pages or 8K bytes) of data from the NAND flash through the EBI2 into the boot SRAM. This block of data is the ARM926 processor boot code. This is done on a page-by-page basis under the control of the boot sequencer. When the transfer is completed successfully, the on-chip boot SRAM contains ECC-corrected boot code.

   If the NAND flash controller reports any read error (usually ECC-uncorrectable errors) with any page fetched, the boot sequencer drives a toggling signal of HCLK/2 frequency onto the DP_TX_DATA pin, releases the ARM926 processor from reset and the watchdog from disable, and aborts the boot-up process.

   If the boot sequencer does not receive an interrupt signal from the NAND flash controller 8K TCXO cycles (409550 ns) after activating it to perform a reset or page-read command to the NAND flash device, the boot sequencer drives a toggling signal of HCLK/4 frequency onto the DP_TX_DATA pin, releases the ARM926 processor from reset and the watchdog from disable, and aborts the boot-up process.

   If the boot sequencer receives an AHB ERROR response when accessing the NAND flash controller, the boot sequencer drives a toggling signal of HCLK/8 frequency onto the DP_TX_DATA pin, releases the ARM926 processor from reset and the watchdog from disable, and aborts the boot-up process.

   Otherwise, the boot sequencer advances to the next step.
3. The boot sequencer then releases the ARM926 processor from reset and removes the disable on the watchdog at the same time. The ARM processor then starts executing from location 0xFFFF0000 which is aliased to the boot code in the boot SRAM.
4. By executing the boot code, the ARM926 processor initializes the entire chip for functional operations, including configuring the SDRAM/SRAM controller into the normal functional mode.
5. The ARM926 processor then starts to transfer the entire DMSS (Dual Mode Subscriber System) software from the NAND flash to external SDRAM/SRAM, which starts at memory location 0x00000000. The data transfer is done by the ARM926 processor itself, which reads from NAND controller’s page FIFO and writes to SDRAM/SRAM controller through the AHB bus.

6. When the transfer completes, the ARM926 processor branches to location 0x00000000 and starts executing from the external SDRAM/SRAM. The ARM926 processor then releases the on-chip boot SRAM to another functional block (MDSP, SRCH4 or deinterleaver). The ARM926 processor also configures the AHB decoder to disable the mapping of location 0xFFFF0000.
7. The boot-up process is now completed. From this point on, the ARM926 processor executes the DMSS software from the external SDRAM/SRAM, and the boot sequencer remains idle until the power-on-reset is asserted, which starts the boot-up process again.

4.6.6 Other Requirements

- Every time the power-on-reset is asserted then deasserted, the whole boot-up process is always activated.
- TCXO (20 MHz) is the only clock available to MSM internal hardware during the boot-up process, and is used as the source of HCLK.
- Only NAND flash devices with a page size of 512 bytes (with ECC enabled) or 528 bytes (with ECC disabled) are supported. NAND flash devices with a page size of 256 bytes are not supported.
- It is a requirement on the NAND flash vendors that the defects in the first block of data in the NAND flash be within ECC's correction capability (1 bit per ECC codeword). In current NAND flash controller design, each ECC codeword is 128 bytes long, so each page contains 4 codewords. The ECC correction capability is therefore 4 bit errors per page evenly distributed into the 4 codewords.
- During the fetching of the boot code from the NAND flash to the boot SRAM, it is not allowed for any AHB bus master (such as TIC) to access the boot SRAM starting at address 0xFFFF0000. If this happens, the boot sequencer's behavior would be unpredictable and data in the boot SRAM may be corrupted. This should not be a problem for software, since the ARM processor is always released from reset when the boot sequencer has already finished boot code loading (if boot from NAND flash) or when the boot sequencer is not performing boot code loading at all (if boot from NOR flash).

The MMU is not turned on until the entire application software has been transferred to SDRAM/SRAM.

4.7 External Bus Interface 1

EBI1 is a high performance bus that supports a wide variety of memories. With a new higher performance ARM9 microprocessor in the MSM6100, the system memory bandwidth becomes a bottleneck when running memory-intensive applications such as video and graphics. Also, LCD accesses consume a significant portion of memory bandwidth.

Keeping this in mind the MSM6100 was designed to provide two distinct memory interfaces. EBI1 was targeted for supporting high speed synchronous memory devices. EBI2 was targeted towards supporting slower asynchronous devices such as LCD, NAND Flash, SRAM, NOR Flash etc.

The ARM926EJS microprocessor is a cached processor and all accesses to external memory will be burst accesses of 4 or 8 32-bit words when the memory region is declared to be cacheable/bufferable. To take advantage of this higher performance microprocessor, the data from the memories have to satisfy the requirements for these burst accesses.

Using the traditional asynchronous memories that have been used in the phones results in access times of around 70-90 ns access times: very poor performance. Page memories would improve accesses to 30-40 ns, however, this is still not sufficient for new applications.

The MSM6100 is meant to provide a new flexible memory interface to support high speed synchronous memories operating at much higher frequencies than was possible with asynchronous memories. The two new kinds of memories that are targeted are the Synchronous Burst NOR Flash device and Synchronous DRAM devices. Also, the memory interface design can support the traditional page mode and asynchronous memory devices for a low-power phone design.

4.7.1 Features

- 16-bit static and dynamic memory interface
- 32-bit dynamic memory interface
- 24 bits of address for static memory devices which can support up to 32MB on each chip select
- Synchronous burst memories supported (Burst NOR, Burst PSRAM)
- Synchronous DRAM memories supported
- Supports only low-power memories at 1.8V I/O power supply voltage
- Byte-addressable memory supporting 8-bit, 16-bit and 32-bit accesses
- Page mode memory support
- Pseudo SRAM (PSRAM) memory support

4.7.2 Memories Supported on EBI1

The following memories will be supported on EBI1:

- 16-bit asynchronous NOR flash
- 16-bit asynchronous SRAM
- 16-bit asynchronous page mode NOR flash
- 16-bit asynchronous page mode PSRAM (Toshiba), asynchronous FCRAM

- 16-bit synchronous burst mode flash (AMD, Micron, etc.)
- 16-bit synchronous PSRAM (similar timing as burst flash)
- 16 and 32-bit synchronous DRAM (low power SDRAM)

The following combinations of memories cannot be supported simultaneously.

- Synchronous burst mode flash + SDRAM

Following combinations of Static and Dynamic Memory are also NOT supported:

- Async/Page NOR Flash + SDRAM
- SRAM + SDRAM

4.7.3 EBI1 System

The EBI1 system includes two controllers, each targeting different memory types:

- The multi-port memory controller (MPMC), also referred to as the PL172 core, is primarily intended to support low power synchronous DRAM.
- The external memory controller (XMEMC) is primarily intended to support synchronous burst mode devices.

Only one memory controller can be active at any time. Which memory controller remains active depends on the memories that need to be supported on the EBI1 bus. The memory controller switching should occur only once after power-up during the initialization phase. Section 4.7.3.1 discusses the procedure for switching memory controllers after reset. Figure 4-25 shows the hierarchy of the EBI1 block. The EBI1 core receives the AHB buses from the ARM, DSP, and the GRP processors.

The EBI1 core operates at the bus clock (HCLK). This means that any synchronous memories that reside on the EBI1 bus (external to the MSM) must also be capable of operating at this frequency. For example, if HCLK = 50 MHz then the SDRAM/burst flash must be capable of operating at 50 MHz as well.

Both memory controllers support byte (8-bit), half-word (16-bit), and word (32-bit) accesses to external memory. However, the XMEM controller only supports a 16-bit memory interface. The MPMC controller supports both 16 and 32 bit SDRAM interface.

Figure 4-25 shows the hierarchy of the EBI1 block. The EBI1 core receives the AHB buses from the ARM, DSP, and the GRP processors.

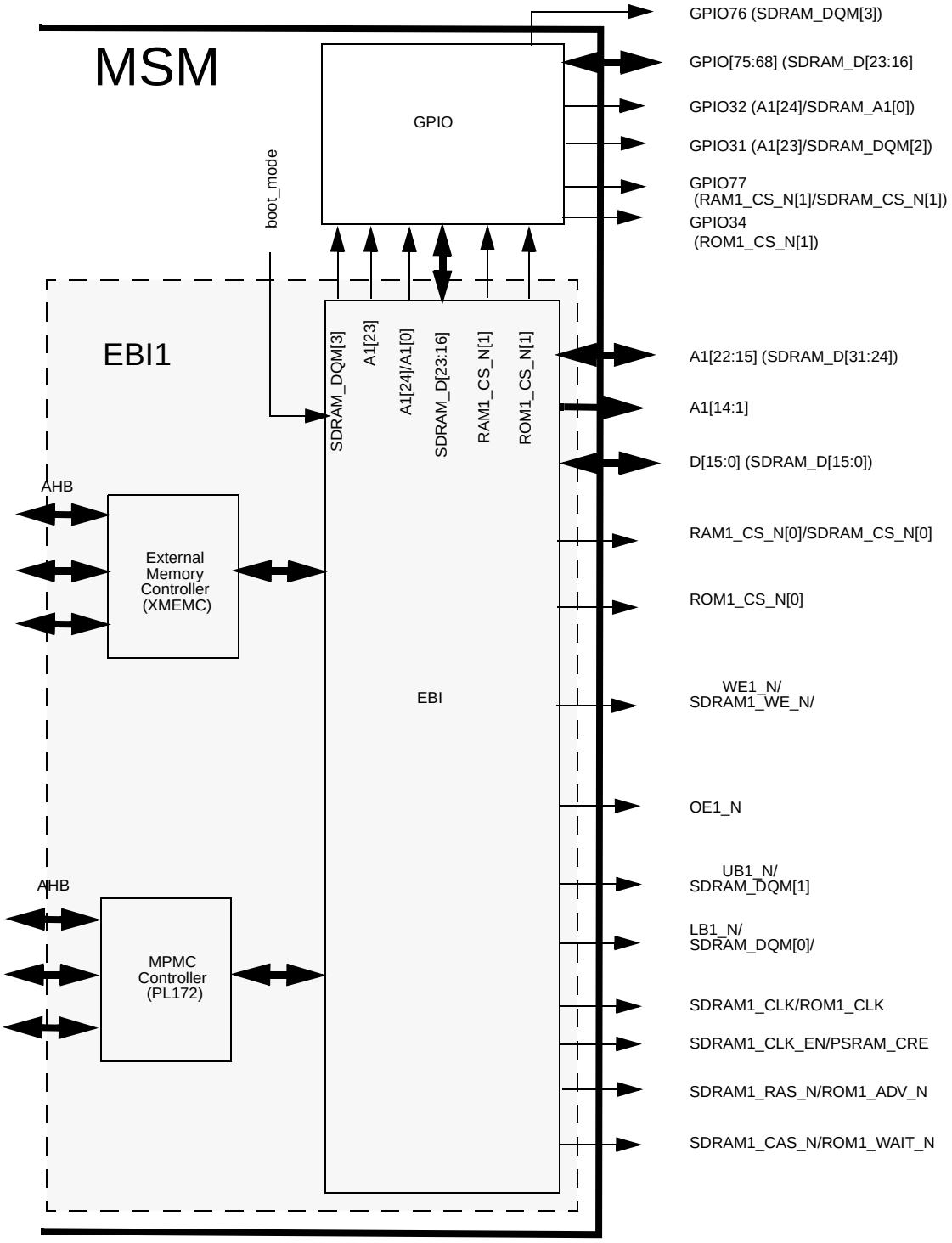


Figure 4-25 EBI1 Hierarchy and Connections to External Memory Pins

The memory controllers support a wide range of memory devices to enable maximum flexibility in designing a MSM6100-based phone targeting both a low-power solution and a high-performance solution.

Table 4-4 Memories Supported on EBI1 by the Two Memory Controllers

Memory Type	Max Size	MPMC	XMEMC	Chip Selects which support this memory	Comments
Async NOR Flash	32MB	Yes	Yes	all EBI1 chip selects	-
Page NOR Flash	32MB	Yes	Yes	all EBI1 chip selects	MPMC only supports a page size of 4 16-bit words
Sync Burst NOR Flash	32MB	No	Yes	all EBI1 chip selects	MPMC does not support Synchronous Burst NOR Flash timing. This is why SDRAM cannot be used in conjunction with Burst NOR Flash
Async/Page PSRAM	32 MB	No	Yes	all EBI1 chip selects	The MPMC does not support any special timing requirement of PSRAM. If PSRAM has the same timing as SRAM then MPMC should be able to support that
Sync PSRAM	32 MB	No	Yes	all EBI1 chip selects	XMEMC can support Sync. PSRAM if it has the same timing as a Burst NOR Flash device
Synchronous DRAM	64 MB	Yes	No	Only on RAM1_CS_N0 and RAM1_CS_N1 (gpio77)	SDRAM is supported only on the RAM chip selects
Async SRAM	32 MB	Yes	Yes	all EBI1 chip selects	-

Based on the different memories that can be supported by the two memory controllers the next table shows the different memory combinations that can be supported. The different configurations are rated based on the cost, power, and performance considerations. The phone designer must choose the memory configuration that is best suited for his system.

Even though the EBI1 in the MSM6100 supports all the configurations mentioned in the table below there are two primary 16-bit configurations that are targeted. These are indicated by the shaded regions in Table 4-5.

Table 4-5 Supported EBI1 Memory Configurations

BOOT MODE pin	Configuration	Active Controller	Comments	Cost/Power/ Performance
1	NAND (EBI2) + 16-bit SDRAM	MPMC	NAND (EBI2) used for code storage	low cost high power high performance
1	NAND (EBI2) + 32-bit SDRAM	MPMC	NAND (EBI2) used for code storage	medium cost highest power highest performance
1	NAND (EBI2) + Async PSRAM	XMEMC	NAND (EBI2) used for code storage	low cost medium power low performance
1	NAND (EBI2) + Sync PSRAM*	MPMC	NAND (EBI2) used for code storage	medium cost medium power high performance
0	Sync NOR + Async/Page PSRAM	XMEMC	NOR used for code storage and execution	medium cost medium power medium performance
0	Sync NOR + Sync PSRAM*	XMEMC	NOR used for code storage and execution	medium cost medium power high performance
0	Async NOR + SDRAM	MPMC	NOR used for code storage and for execution	medium cost high power medium performance
0	Sync NOR + Async SRAM	XMEMC	NOR used for code storage and execution	high cost low power medium performance
0	Async NOR + Async SRAM	XMEMC	NOR used for code storage and execution	high cost low power low performance

* Synchronous PSRAM memories can be supported if the timing is similar to the burst NOR Flash.

The NAND + SDRAM configuration is expected to be the cheapest configuration available to a phone designer with the advantage of high performance. However, the power consumption associated with this configuration could impact the battery life.

Alternatively, the synchronous burst NOR + PSRAM configuration provides a lower power consumption at the expense of cost and some performance loss because of the asynchronous PSRAM. If a synchronous PSRAM memory is used instead that would provide a high performance similar to the NAND+SDRAM combination.

Figure 4-26 below shows a few example memory configurations below

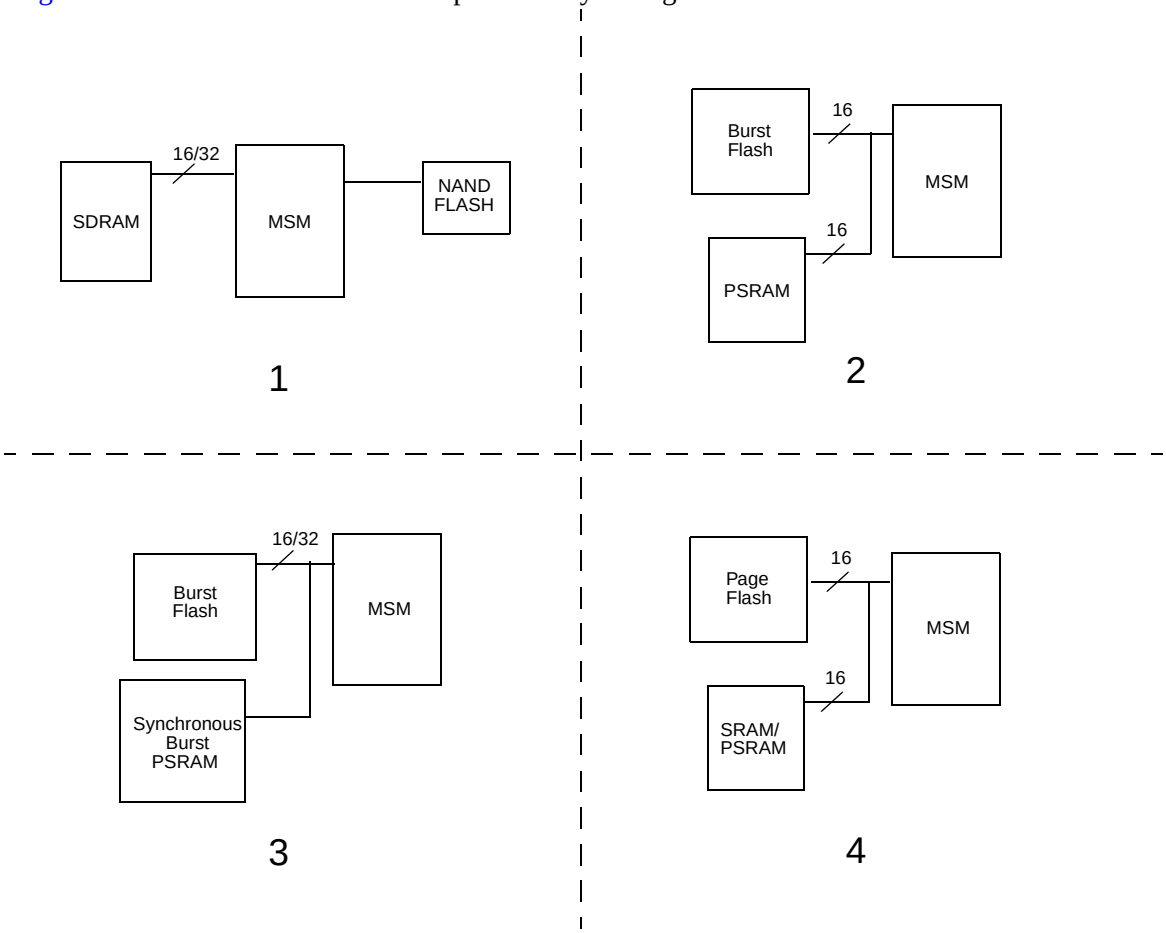


Figure 4-26 Different Memory Configurations on the EBI1 Bus

4.7.3.1 Boot-up Configuration and Memory Controller Switching

- On power up, the XMEM controller is always selected.
  - To switch to the MPMC, software will have to write to the EBI1_MEM_CTLR_SEL_CMD register.
  - To switch back to the XMEMC, software will have to write to the EBI1_MEM_CTLR_SEL_CMD register.
  - Software can determine which memory controller is active by reading the EBI1_MEM_CTLR_SEL_STATUS register.

4.7.3.2 Chip Selects

EBI1 provides support for the following four chip selects:

- ROM1_CS_N0
- ROM1_CS_N1 (GPIO34)
- RAM1_CS_N0/SDRAM1_CS_N0
- RAM1_CS_N1/SDRAM1_CS_N1 (GPIO77)

Table 4-6 below indicates which registers are responsible for configuring the parameters corresponding to a particular chip select based on the memory controller that is active.

Table 4-6 Register Names Associated With the EBI1 Chip Selects

Chip Select Pin Name	XMEMC	MPMC
ROM1_CS_N0	EBI1_CS0_CFG	MPMC_ST_CONFIG0
ROM1_CS_N1	EBI1_CS1_CFG	MPMC_ST_CONFIG1
RAM1_CS_N0	EBI1_CS2_CFG	MPMC_ST_CONFIG2
RAM1_CS_N1	EBI1_CS3_CFG	MPMC_ST_CONFIG3
SDRAM1_CS_N0	N/A	MPMC_DY_CONFIG0
SDRAM1_CS_N1	N/A	MPMC_DY_CONFIG1

4.7.3.3 Software Initialization Prior to Memory Controller Switching

Prior to switching from the XMEMC to the MPMC the following registers must be initialized:

1. MPMC_CONTROL
2. MPMC_CONFIG
3. MPMC_STATIC_CONFIG (0,1,2,3)

4.7.4 External Memory Controller (XMEMC)

The external memory controller is targeted mainly towards supporting static memories. The important enhancements are the support for burst mode memories such as the Intel StrataFlash, and AMD burst flash which power up in the asynchronous/page mode and then can be configured to operate in the burst (synchronous) mode. Some of the important features of the XMEMC are:

- 1 WS MINIMUM READ access

- The fastest read access by the controller to an asynchronous memory will require 2 clock cycles. Software has to ensure that the value programmed into the "DELTA_RD/DELTA_WR" bits of the CSx_CFG register has to be greater than or equal to 0x01.
- 2 WS MINIMUM WRITE access
- The fastest write access by the controller to an asynchronous memory will require 3 clock cycles. Software has to ensure that the value programmed into the "HOLD_WAIT" bits of the CSx_CFG register has to be greater than or equal to 0x01.
- 8-bit devices not supported; only 16-bit memory devices will be supported by this memory controller.
- Supporting synchronous burst memories requires the following new pins as well:
  - ROM1_ADV_N
  - ROM1_CLK
  - ROM1_WAIT_N
- To support low power operation of PSRAM the following pin can be used.
  - SDRAM1_CLK_EN(PSRAM_CRE)

4.7.4.1 Priorities

The three masters accessing the XMEM controller are the ARM, DSP, and the GRP processors. GRP is always the lowest priority port in accesses to the XMEM controller. There is a programmable priority scheme that is supported between the ARM and DSP. Because of the real time requirements of the DSP it is normally expected that the DSP will be programmed to have higher priority in accesses to external memory.

4.7.4.2 DELTA and WAIT Cycles Operation

The delta cycles are the additional cycles inserted at the beginning of an access. The delta cycles are inserted in the first access when a particular chip select is enabled for page or synchronous burst operation or when the access crosses a page boundary. The WAIT cycles are inserted in all accesses to the memory.

Upon power-up, all accesses to any chip select will incur the following number of clock cycles:

(DELTA + WAIT + 1) cycles for reads and

(DELTA + WAIT + HOLD + 1) cycles for writes

Once a chip select is enabled for page mode read operation the first access to the memory will always incur (DELTA + WAIT + 1) cycles. Subsequent sequential accesses within the same page will incur (WAIT + 1) cycles. For page mode write operation, all accesses will also incur the HOLD cycles.

Example 1:

For a 50 MHz HCLK 70/25 ns page flash device, the following conditions must be satisfied (note that in reality the on-chip delay and the board routing delay must be taken into consideration, however we choose to ignore that for all examples here):

- DELTA + WAIT + 1 > 80 ns
- WAIT + 1 > 25 ns

So, we can select a WAIT = 1 and DELTA = 2 to give us 80 ns for the first access and 40 ns for the sequential access within the page.

Example 2:

For a 54 MHz HCLK, 70/13.5 ns burst flash device configuration, the following conditions must be satisfied:

- DELTA + WAIT + 1 > 70 ns
- WAIT + 1 > 13.5 ns

So, we can select a WAIT = 0 and DELTA = 3 to give us 74 ns for the first access and 18.5 ns for the sequential access within the page. There are two types of burst flash devices which are supported: Intel StrataFlash and the AMD parts which DOES NOT have multiplexed address/data. The AMD part does not support a WAIT = 1 setting. However, the Intel part allows 2 cycles for subsequent accesses.

NOTE It is recommended that the WAIT values be set to 0 and the DELTA values be used for all asynchronous non-page memories.

4.7.4.3 HOLD Cycles Operation

The hold cycles are the extra cycles inserted after every write access. A minimum value of 1 HOLD value must be programmed for every write access, including those for page mode writes.

The data from the MSM is driven until the WE_N strobe is low and once it goes high, the data is not driven. The keepers are expected to retain the value if no one else drives it. When a hold cycle value of 1 is programmed then the CS_N stays active for 1 extra clock cycle. The data is held with respect to WE_N.

4.7.4.4 Recovery Cycles Operation

The recovery cycles are required to ensure that there is no contention on the databus. This value can be obtained from the memory datasheet.

Example 1:

For a 50 MHz HCLK and a flash devices with 30 ns recovery time the condition required to be met is (RECOVERY > 30 ns). So, we need 2 cycles programmed for the RECOVERY to ensure that there is no bus contention if another read or a write access happens.

Recovery cycles are inserted for the following situations:

1. Consecutive access to two different chip selects with the first access being a read.
2. Read access followed by a write access to the same chip select.

4.7.4.5 Precharge Cycles Operation

The precharge cycles, as the name implies, is used by any pseudostatic SRAM to ensure that the precharge time is met. These cause extra cycles to be inserted between consecutive read or write access to the same memory. The precharge cycles are NOT inserted for bus-sized and page accesses.

When the memory is configured for page/burst mode read or write operation the precharge cycles are inserted only when the access goes across page boundaries.

- For a read-write or write-read access to the same chip select there is one arbitration clock cycle inserted between accesses which should provide the necessary precharge time.
- For a read-write access, recovery cycles will ensure that enough cycles are inserted between accesses.
- For a write-read access to the same chip select, and HOLD=1, two precharge cycles (1 arbitration cycle + 1 hold) will be inserted. More precharge cycles will require increasing the value for HOLD.

NOTE Note that precharge cycles also get inserted for accesses to burst memory devices when going across page boundaries. If this is not required make sure that the PRECHARGE value for that chip select is set to 0.

4.7.4.6 AVD Recovery Cycles Operation

The AVD recovery cycles apply *only* when a synchronous burst mode device is used in synchronous or asynchronous mode.

The address valid (AVD) recovery cycles have to be inserted to ensure that the AVD_N signal stays high for the specified duration before going low (active). There is a 1 clock cycle arbitration penalty which will guarantee that AVD_N stays high for at least 1 clock cycle between accesses to different chip selects or between a read and write access.

On power-up, AVD_N will be asserted low for all accesses to burst memory. This is required to ensure that the memory will operate in the asynchronous/page mode. Once the memory is configured to be a burst memory, AVD_N has to pulse during the access.

To ensure that the AVD_N stays high for the required amount of time before going low when doing a burst access, insert the appropriate AVD recovery cycles.

Program the AVD_RECOVERY field to at least "1" if CS*x* is a burst memory that uses AVD_N but is operating in the page Mode. For example, if CS0 is a burst memory configured to operate in page Mode and CS1 is a burst memory configured to operate in burst mode, then the AVD recovery cycles will ensure that after an asynchronous read access to CS0, a synchronous burst read to CS1 will have 1 recovery cycle inserted to ensure AVD_N high for 1 clock.

However, if CS0 is a page memory that does not required AVD_N, program CS0_AVD_RECOVERY to 0 and configure CS0_AVD_STATE to be 1.

Accesses to different chip selects or between read-write/write-read accesses always incurs a 1-cycle arbitration penalty. This guarantees that there is a minimum of 1 cycle AVD recovery inherent in the design. It is recommended to program this bit field to "00" since the maximum frequency of operation is around 75 MHz which guarantees a recovery of at least 13 ns. For example, the Micron burst flash device defines this parameter as $\mathbf{t_{vph}}$ = 10 ns which is easily met.

4.7.4.7 Page Memory Operation

Page memory devices typically require a long initial access time for completing the first access in a page. Subsequent accesses within the page are much faster. Most of the memories that support page mode operation usually only support a PAGE READ operation. However, MSM6100 supports a PAGE WRITE operation as well.

4.7.4.8 Page Memory Initialization

Every chip select has a bit CS*x*_PAGE_RD_ENA and CS*x*_PAGE_WR_ENA in the EBI1_CS*x*_CFG1 register that must be set to '1' if operating in page mode (independent control for reads and writes).

Also, the DELTA and WAIT must be programmed appropriately for operating in page mode.

- Page read operations can have WAIT_RD = 0 which implies that subsequent read operations within the page would incur only 1 HCLK cycle.
- Page write operations CANNOT have WAIT_WR = 0. The minimum value that can be programmed is WAIT_WR = 1 when CS*x*_PAGE_WR_ENA is set to 1. Since the minimum value for HOLD is at least 1. So, the fastest page write access that can be supported would incur a total of 3 cycles (WAIT_WR + 1 + HOLD).

4.7.4.9 Synchronous Burst Memory Operation

Burst memories are like page memories in that they provide faster access times for accesses within a page. However, burst memories use a clock to begin their accesses (page memories detect an address change to begin their access) and this enables them to operate at much higher frequencies. The following new pins will be needed for supporting burst mode memories:

- ROM1_ADV_N
  - This pin will indicate to the burst memory that the address is valid and must be captured by the memory.
- ROM1_WAIT_N
  - This pin is an input to the MSM device. The burst memory device will assert this pin low if it wants to assert waitstates for any particular access. This pin must be asserted one clock cycle prior to the actual access.
- ROM1_CLK
  - Provides the clock to the burst memory device.

The controller uses the DELTA and WAIT settings as well for the burst memory device. Note that the ROM1_WAIT_N pin is used to delay a particular access. However, programming a DELTA = 3, WAIT = 0 implies that the first access to a burst memory will be take DELTA+WAIT+1 irrespective of the value on ROM1_WAIT_N. It checks for the wait on the pin only after the initial DELTA and WAIT counters have expired.

4.7.4.10 Burst Memory Initialization

All burst mode devices will require the following to be configured for the memory during software initialization:

- Rising edge of the CLK has to be the active edge for the memory.
- Synchronous programming (write) to burst flash will not be supported. All writes to burst flash must be asynchronous. This will require that BURST_WR_ENA must be set to 0 for all burst flash devices.
- RDY/WAIT_N must be configured to arrive one clock earlier than the data.
- Can only support 0 waitstates subsequent operation. A value of WAIT_RD = 0 ***must be programmed for all burst devices***.
- The burst memory ***must*** be configured when the cache is ***disabled***.
- The burst memory ***must not*** be used for execution/reading while being configured. For example, if CS0 is currently a burst flash operating in asynchronous mode then the microprocessor has to be executing out of a different chip select first. The burst flash can then be configured to switch to the synchronous mode as described in the example below.

Example 1:

For a 50 MHz HCLK, 70/13.5 ns AMD burst flash device (AM29BDS640G) the Burst Mode Configuration Register must be programmed first. Some of the parameters such as WRAP operation should be configured based on software requirements. The AMD burst flash requires the following sequence of writes to set the Burst Mode Configuration Register:

1. Write to address = 0x555 with data = 0xAA. This implies that the ARM microprocessor must actually write to ADDR = 0xAAA since the A0 bit the Burst memory corresponds to HADDR[1].
2. Write to address = 0x2AA with data = 0x55. This corresponds to ADDR = 0x554.
3. Write to address = 0x(CR)555 with data = C0. CR corresponds to the configuration register bits(A19-A12) of the burst memory device. These bits correspond to ADDR bits [A20-A13] from ARM's point of view. Therefore, this offset in the address bits must be taken into account when configuring the burst memory device. Based on the settings below we will write to ADDR = 0x46AAA.
   - A19 = 0, Synchronous Read Enabled
   - A18 = 0, RDY active one clock cycle before data
   - A17 = 1, Burst starts and data is output on the rising edge of CLK
   - [A16:A15] = 00, Continuous burst. If the WRAP8_MODE feature is used then set to [10]. This is described in more detail in Section 4.7.4.11.
   - [A14:A12] = 011, configures the wait state settings for the initial access. Since we ***only*** support the handshake option we have to set 5 clock cycles for 50 MHz operation as specified in the datasheet.

In addition to configuring the burst flash we also need to program the configuration registers in the XMEM controller. We have to make sure that the following conditions are satisfied.

- DELTA_RD + WAIT_RD + 1 > 70 ns
- WAIT_RD + 1 > 13.5 ns

WAIT_RD = 0 implies the clock can be as fast as 1/13.5 ns = 74 MHz assuming that the burst flash can support it; the AMD part mentioned above supports a maximum speed of 54 MHz only. We have to program the delta waitstates correctly now. Looking at the AMD datasheet we find that the "RDY" pin is controlled by the burst flash indicating when the data will actually be available. Since we support the handshake mode the delta cycles need to be programmed to a value ≥ 1. The sequence of steps to configure CS0 to be a burst flash operating in synchronous mode is as follows:

1. Set DELTA_RD = 2, and WAIT_RD = 0 for correct operation by writing to the EBI1_CS1_CFG0 register. Since we use the "HANDSHAKING" mode of the burst memory we rely on the RDY/ROM1_WAIT_N to go high to indicate valid data.
2. Set READ_DATA_PIPELINE_ENA = 0 for 50 MHz operation. For higher frequency operation it may be required to set this bit to 1. Refer to Section 4.7.4.13 for a description of this functionality.
3. Next, the AMD burst flash must be configured as described in the datasheet to meet the conditions listed previously. Note that since the memory is 16-bit wide all accesses to the memory are 16-bit (half word) accesses. The sequence of operations are:

```
LDR  R3, =EBI1_CS0_BASE
```

```
LDR  R4, =0xAAA
LDR  R5, =0xAA
STRH R5, [R3,R4]

LDR  R3, =EBI1_CS0_BASE
LDR  R4, =0x554
LDR  R5, =0x55
STRH R5, [R3,R4]

LDR  R3, =EBI1_CS0_BASE
LDR  R4, =0x46AAA
LDR  R5, =0xC0
STRH R5, [R3,R4]
```

4. Now, set the CS0_BURST_RD_ENA bit to “1” in the EBI1_CS0_CFG1 register.

Example 2:

For a 50 MHz HCLK, and a 54 MHz Micron Synchronous Burst Flash MT28F322D20 (80 ns/17 ns) we have to program the Read Configuration Register (RCR) first. This Micron part requires the following sequence of writes to put it into burst mode.

1. Write to address = RCD (Read Configuration Data) with data = 0x60. The RCD value based on the parameters listed below implies that 0x21CF must be presented on the address bus to the burst memory. Taking into account the offset in the address required by the ARM the value of the ADDR = 0x439E
   - A15 = 0, Synchronous Burst Access Mode
   - A14 = 0, Reserved
   - [A13:A11] = 100, Latency Counter = 4 for 50 MHz operation
   - A10 = 0, Reserved
   - A9 = 0, Hold Data Out for one clock only
   - A8 = 1, WAIT_N is asserted one clock cycle before data is available
   - A7 = 1, Data is always addressed linearly
   - A6 = 1, Rising edge of the clock is the active edge
   - [A5:A4] = 00, Reserved
   - A3 = 1, Burst does not wrap within the burst length.
   - [A2:A0] = 111, Continuous burst. The micron burst memory does not support a Burst Length of 16 words hence the WRAP8_MODE feature CANNOT be used.
2. Write to address = RCD (in this case: 0x439E) with data = 0x03

Now, we need to also configure the DELTA_RD and WAIT_RD parameters for the burst memory.

- Since the clock period is 20 ns and the Burst Flash has an access time of 17 ns there is approximately a 3 ns margin here which should account for any external board delays. A value of WAIT_RD = 0 can thus be programmed. This is ***only*** an example. The system designer must do the analysis to make sure that the timing can be met from the AC characterization data in the device specification document.
- For the 54 MHz part the DELTA_RD = Latency Configuration Code + 1 = 5

The sequence of steps to configure CS0 to be a burst flash operating in synchronous mode is as follows:

1. Set DELTA_RD = 5, and WAIT_RD = 0 for correct operation by writing to the EBI1_CS1_CFG0 register.
2. Set READ_DATA_PIPELINE_ENA = 0 for 50 MHz operation. For higher frequency operation it may be required to set this bit to 1. Refer to Section 4.7.4.13 for a description of this functionality.
3. Next, the Micron burst flash must be configured as described in the datasheet to meet the conditions listed previously. Note that since the memory is 16-bit wide all accesses to the memory are 16-bit (half word) accesses. The sequence of operations are:

```
LDR  R3, =EBI1_CS0_BASE
LDR  R4, =0x439E
LDR  R5, =0x60
STRH R5, [R3,R4]

LDR  R3, =EBI1_CS0_BASE
LDR  R4, =0x439E
LDR  R5, =0x03
STRH R5, [R3,R4]
```

4. Now, set the CS0_BURST_RD_ENA bit to “1” in the EBI1_CS0_CFG1 register.

```
PICTURE SHOWING BURST FLASH and BURST PSRAM HOOKED UP
```

4.7.4.11 WRAP8 Burst Optimization

There are two different modes of supporting AHB burst transactions.

1. Optimizing for WRAP8 bursts in memories with page size of 16 HWORDs (16-bit words)
2. Optimizing for INCR bursts

In the WRAP8 mode of operation, the memory will keep wrapping at the 8-word (16-hword) boundaries. This ensures that only the first access in a cache linefill will be a long access and all the subsequent accesses can be single cycle accesses. In the INCR mode of operation the memory will keep supplying the data continuously until the controller terminates the burst. XMEMC will use the RDY (WAIT_N) signal provided by the burst memory to insert extra wait cycles when required.

Set the WRAP8_MODE bit for a particular chip select when the following conditions are satisfied:

1. Chip select is hooked up to a burst device
2. If the burst device supports a page size of 16 hwords or greater.

When the burst flash is being configured then the burst memory must be set to the "16-hword wraparound" mode instead of the "continuous burst" mode.

NOTE Although setting this bit to 1 optimizes accesses for the ARM, the DSP and the GRP do not perform WRAP8 operations and will suffer a hit in their performance. Therefore, we recommend setting this bit to 1 only after analyzing the system.

Step 3 of the initialization sequence described in the previous section must be modified as follows to support WRAP8 optimization. The WRAP8_MODE bit of EBI1_CS1_CFG1 must also be set to 1.

```
LDR  R3, =EBI1_CS1_BASE
LDR  R4, =0xAAA
LDR  R5, =0xAA
STRH R5, [R3,R4]

LDR  R3, =EBI1_CS1_BASE
LDR  R4, =0x554
LDR  R5, =0x55
STRH R5, [R3,R4]

LDR  R3, =EBI1_CS1_BASE
LDR  R4, =0x64AAA ; <--------- Configure AMD part for WRAP8 operation
LDR  R5, =0xC0
STRH R5, [R3,R4]
```

4.7.4.12 Burst Memory Program Operation

The program and erase operation of the burst memory ***must be performed in asynchronous mode***. For example, the AMD burst flash supports "synchronous" program operation as well. However, MSM6100 will not support it. Software will have to ensure that the CSx_BURST_WR_ENA is set to 0.

Other requirements for AMD program/erase operations:

- Set PWRSAVE_CLK_DIS to 1

  This ensures that the ROM1_CLK pin going to the burst memory will be kept toggling only during read operations to the AMD memory after it is configured for burst operation

4.7.4.13 READ_DATA_PIPELINE_ENA

In the engineering sample, the speed of the burst memory interface was limited because of the read data path. The data coming back from the synchronous burst memory after a read had to propagate all the way back to the masters (ARM, DSP, or GRP) before it was registered. In the MSM6100 we still provide this mode of operation. However, we provide a new software configuration bit which will break this critical path, as shown in Figure 4-27.

It is recommended that the READ_DATA_PIPELINE_ENA bit should be set to 1 for 50 MHz operation. For higher speeds it is recommended to not use this feature because of the “inherent” pipeline that is added by skewing the ROM1_CLK(and FB_CLK).

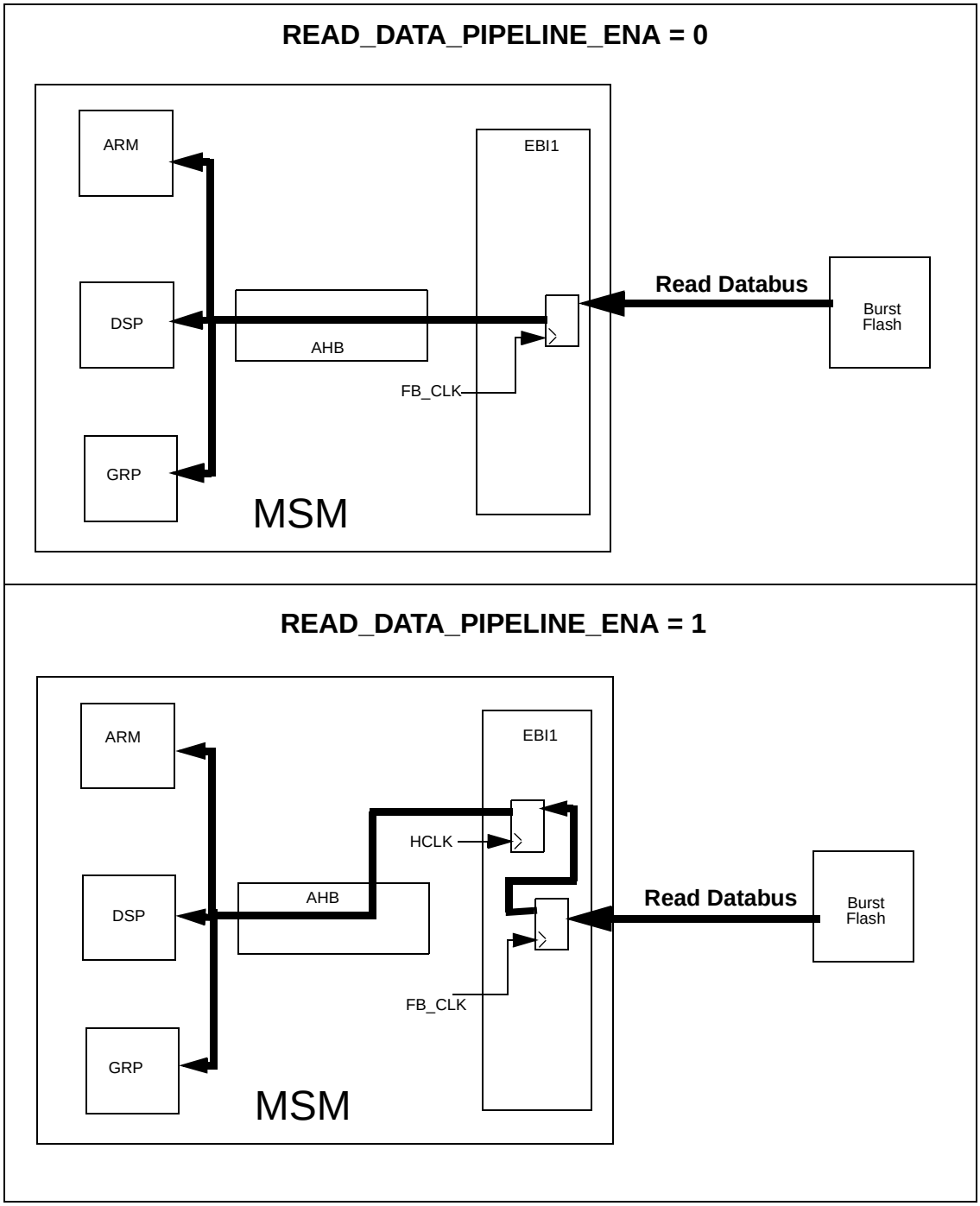


Figure 4-27 Breaking the Critical Path on the Read Data For Burst Memory Access

4.7.4.14 PWRSAVE_CLK_DIS Feature

When burst flash is used, power can be saved by preventing the ROM1_CLK pin from toggling when no accesses are being done to the burst memory. This is achieved by setting the PWRSAVE_CLK_DIS to high which ensures that the ROM1_CLK (burst clock) coming out of EBI1 will be toggling only during a valid access.

Once the ARM begins execution out of the burst flash (in synchronous mode) it can always go into sleep/low-power mode and upon wake-up can continue execution out of the burst flash. It is not necessary to switch to asynchronous mode prior to going into sleep.

4.7.4.15 Low Power Mode Control Using the EBI1_PSRAM_CRE Register

When the XMEM controller is active then the SDRAM1_CLK_EN pin can be used by software as a general purpose output (GPO). The EBI1_PSRAM_CRE register bit 0 can be used to control the pin through software.

This is very useful for connecting to pseudo SRAMs that support a deep power down or deep sleep mode using an extra pin. For example:

- cs pin on the Toshiba PSRAM
- zz pin on the Cypress PSRAM
- cre pin on the Micron PSRAM
- mode pin on the NEC Mobile RAM
- zz pin on the Samsung UtRAM

This pin can also be used for any PSRAMs that operate in the synchronous mode, and to issue commands to the RAM to program the mode register as described in Section 4.7.4.9 for the Micron CellularRAM.

4.7.5 ARM's Multi-Port Memory Controller (MPMC) or PL172

ARM's MPMC controller has to be used when SDRAM memory needs to be used in the system. In addition to SDRAM, it also supports asynchronous memories such as SRAM, and page mode NOR flash (page size = 4 hwords).

The MPMC supports 4 static memory chip selects and 4 dynamic memory chip selects. However, only 2 dynamic chip selects will be used. The multiplexing is shown in the Figure 4-25 which shows the SDRAM chip selects being shared with the regular RAM chip selects. Software can configure each of the RAM chip selects to be either static or dynamic by writing to the EBI1_MPMC_STDY_SEL register.

4.7.5.1 Power Down Mode Support Using SDRAM1_CLK_EN Pin

Even though two SDRAM chip selects are supported, only one CKE (SDRAM1_CLK_EN) pin is brought out. This implies that both chip selects have to enter POWERDOWN at the same time: the SDRAM on one chip select cannot be in a low power mode while the SDRAM on the other chip select remains active.

4.7.5.2 16-bit SDRAM Initialization Requirements

- The address bit 0 (SDRAM_A1[0]) is being shared with a1[24] which comes out on GPIO32. This GPIO must be configured correctly prior to beginning the SDRAM initialization.
- The MPMC supports only the clock delayed mode in the sense that the clock going to the external memory can be delayed by software to meet the timing. The other option would be command delayed where the address/data/control signals going out to the external memory are delayed. However, this mode is NOT supported.
- The MPMC also supports the external clock going to the SDRAM (referred to as the MPMCCLK in the ARM document) to be either 1/2 or 2 times the HCLK. However, we will support only one mode: MPMCCLK = HCLK.
- The SDRAM initialization ***must be done prior to the cache initialization.*** So, cache must be disabled while doing the SDRAM initialization
- The SDRAM initialization ***must be done only after the memory controller has been switched to the MPMC.***
- If the static memory such as flash or SRAM is being used then the ***initialization must be done prior to the memory controller switch.***

4.7.5.3 16-bit SDRAM Initialization Sequence

A sample software initialization sequence is provided below when operating HCLK at 75 MHz. This is provided for reference only. For more details refer to the PL172 TRM.

```
// Initialize EBI1 CS2 (RAM1_CS_N0) to be SDRAM
MEM_WRITE (EBI1_MPMC_STDY_SEL,  0x01);

// Initialize the MPMC controller registers first

MEM_WRITE (MPMC_CONTROL,    0x01);
MEM_WRITE (MPMC_ST_CONFIG0,   0x81);
MEM_WRITE (MPMC_ST_CONFIG1,   0x81);
MEM_WRITE (MPMC_ST_CONFIG2,   0x81);
MEM_WRITE (MPMC_ST_CONFIG3,   0x81);
MEM_WRITE (MPMC_CONFIG,    0x00);

MEM_WRITE (MPMC_ST_W_TWEN0,   0x00);
MEM_WRITE (MPMC_ST_W_TWEN1,   0x00);
MEM_WRITE (MPMC_ST_W_TWEN2,   0x00);
MEM_WRITE (MPMC_ST_W_TWEN3,   0x00);
```

```
MEM_WRITE (MPMC_ST_W_TOEN0,   0x00);
MEM_WRITE (MPMC_ST_W_TOEN1,   0x00);
MEM_WRITE (MPMC_ST_W_TOEN2,   0x00);
MEM_WRITE (MPMC_ST_W_TOEN3,   0x00);

MEM_WRITE (MPMC_ST_W_TRD0,   0x0A);
MEM_WRITE (MPMC_ST_W_TRD1,   0x0A);
MEM_WRITE (MPMC_ST_W_TRD2,   0x0A);
MEM_WRITE (MPMC_ST_W_TRD3,   0x0A);

MEM_WRITE (MPMC_ST_W_TWR0,   0x07);
MEM_WRITE (MPMC_ST_W_TWR1,   0x07);
MEM_WRITE (MPMC_ST_W_TWR2,   0x07);
MEM_WRITE (MPMC_ST_W_TWR3,   0x07);

// Switch to the MPMC controller

MEM_WRITE (EBI1_MEM_CTLR_SEL_CMD,  0x00);

// Initialize GPIO32 which is A[0] for the SDRAM

MEM_WRITE (GPIO_PAGE,     0x20);
MEM_WRITE (GPIO_CFG,     0x04);

// Initialize GPIO77 which is the SDRAM_CS_N

MEM_WRITE (GPIO_PAGE,     0x4D);
MEM_WRITE (GPIO_CFG,     0x04);

//-------------------------------------------------------------------
// Start doing SDRAM initialization
//-------------------------------------------------------------------

// Initialize the timing registers for the Dynamic Controller

MEM_WRITE (MPMC_DY_TRP,      0x02);
MEM_WRITE (MPMC_DY_TRAS,      0x04);
MEM_WRITE (MPMC_DY_TSREX,      0x06);
MEM_WRITE (MPMC_DY_TAPR,      0x04);

MEM_WRITE (MPMC_DY_TDAL,      0x06);
MEM_WRITE (MPMC_DY_TWR,      0x01);
MEM_WRITE (MPMC_DY_TRC,      0x06);
MEM_WRITE (MPMC_DY_TRFC,      0x06);

MEM_WRITE (MPMC_DY_TXSR,      0x06);
MEM_WRITE (MPMC_DY_TRRD,      0x01);
MEM_WRITE (MPMC_DY_TMRD,      0x01);

// Insert some delay first
for (i=0; i<20; i++);
```

```
// Insert a NOP command first

MEM_WRITE (MPMC_DY_CNTL,      0x0183);

// Issue a PALL command

MEM_WRITE (MPMC_DY_CNTL,      0x0103);

// Insert some delay again
for (i=0; i<4; i++);

// Insert a number of refresh cycles
MEM_WRITE (MPMC_DY_REF,      0x02);


// Insert more delay
for (i=0; i<8; i++);

// Number of HCLKs between refresh cycles. Since we are
// operating at 75 MHz the value is 75*16 HCLK cycles. The
// hex value of 75 is 0x4D. Program a value slightly
// smaller to be safe.
MEM_WRITE (MPMC_DY_REF,      0x40);

// Initialize the RAS/CAS register
MEM_WRITE (MPMC_DY_RAS_CAS1,     0x0202);
// Initialize the Config. register
MEM_WRITE (MPMC_DY_CONFIG1,     0x14C01488);

// Initialize the MPMC Dynamic Control register to MODE
MEM_WRITE (MPMC_DY_CNTL,      0x83);

// Program the MODE register in the SDRAM by reading from
// the address below

ReadValue = MEM_READ (EBI1_CS2_BASE + 0x8C00);

// Program the Extended MODE register by reading from the
// address below
ReadValue = MEM_READ (EBI1_CS2_BASE + 0x800000);

// Initialize the MPMC Dynamic Control register to NORMAL
MEM_WRITE (MPMC_DY_CNTL,      0x03);

// Enable the buffers by writing to the config register
MEM_WRITE (MPMC_DY_CONFIG1,     0x14C81488);

// SDRAM Initialization done
```

4.7.5.4 32-bit SDRAM Initialization Requirements

In addition to the things that are required for the 16-bit SDRAM initialization the following steps must be done as well:

- The GPIOs corresponding to DQM[2], DQM[3], SDRAM_D1[23:16] must be configured.

4.7.5.5 32-bit SDRAM Initialization Sequence

```
// Initialize CS2 to be SDRAM
MEM_WRITE (EBI1_MPMC_STDY_SEL,  0x0);

// Initialize the MPMC controller registers first

MEM_WRITE (MPMC_CONTROL,    0x09);
MEM_WRITE (MPMC_ST_CONFIG0,   0x81);
MEM_WRITE (MPMC_ST_CONFIG1,   0x81);
MEM_WRITE (MPMC_ST_CONFIG2,   0x81);
MEM_WRITE (MPMC_ST_CONFIG3,   0x81);
MEM_WRITE (MPMC_CONFIG,    0x00);

MEM_WRITE (MPMC_ST_W_TWEN0,   0x00);
MEM_WRITE (MPMC_ST_W_TWEN1,   0x00);
MEM_WRITE (MPMC_ST_W_TWEN2,   0x00);
MEM_WRITE (MPMC_ST_W_TWEN3,   0x00);

MEM_WRITE (MPMC_ST_W_TOEN0,   0x00);
MEM_WRITE (MPMC_ST_W_TOEN1,   0x00);
MEM_WRITE (MPMC_ST_W_TOEN2,   0x00);
MEM_WRITE (MPMC_ST_W_TOEN3,   0x00);

MEM_WRITE (MPMC_ST_W_TRD0,   0x0A);
MEM_WRITE (MPMC_ST_W_TRD1,   0x0A);
MEM_WRITE (MPMC_ST_W_TRD2,   0x0A);
MEM_WRITE (MPMC_ST_W_TRD3,   0x0A);

MEM_WRITE (MPMC_ST_W_TWR0,   0x07);
MEM_WRITE (MPMC_ST_W_TWR1,   0x07);
MEM_WRITE (MPMC_ST_W_TWR2,   0x07);
MEM_WRITE (MPMC_ST_W_TWR3,   0x07);

// Switch to the MPMC controller

MEM_WRITE (EBI1_MEM_CTLR_SEL_CMD,  0x00);

// Initialize GPIO32 which is A[0] for the SDRAM

MEM_WRITE (GPIO_PAGE,     0x20);
MEM_WRITE (GPIO_CFG,     0x04);

// Initialize GPIO77 which is the SDRAM_CS_N
```

```
MEM_WRITE (GPIO_PAGE,     0x4D);
MEM_WRITE (GPIO_CFG,     0x04);

// Initialize the GPIOs needed for 32-bit SDRAM support

MEM_WRITE (GPIO_PAGE,     68);
MEM_WRITE (GPIO_CFG,    0x08);

MEM_WRITE (GPIO_PAGE,     69);
MEM_WRITE (GPIO_CFG,    0x08);

MEM_WRITE (GPIO_PAGE,     70);
MEM_WRITE (GPIO_CFG,    0x08);

MEM_WRITE (GPIO_PAGE,     71);
MEM_WRITE (GPIO_CFG,    0x08);

MEM_WRITE (GPIO_PAGE,     72);
MEM_WRITE (GPIO_CFG,    0x08);

MEM_WRITE (GPIO_PAGE,     73);
MEM_WRITE (GPIO_CFG,    0x08);

MEM_WRITE (GPIO_PAGE,     74);
MEM_WRITE (GPIO_CFG,    0x08);

MEM_WRITE (GPIO_PAGE,     75);
MEM_WRITE (GPIO_CFG,    0x08);

MEM_WRITE (GPIO_PAGE,     76);
MEM_WRITE (GPIO_CFG,    0x08);

// Initialize GPIO31 for supporting dqm2

MEM_WRITE (GPIO_PAGE,     31);
MEM_WRITE (GPIO_CFG,    0x04);

//-------------------------------------------------------------------
// Start doing SDRAM initialization
//-------------------------------------------------------------------

// Initialize the timing registers for the Dynamic Controller

MEM_WRITE (MPMC_DY_TRP,      0x02);
MEM_WRITE (MPMC_DY_TRAS,      0x04);
MEM_WRITE (MPMC_DY_TSREX,      0x06);
MEM_WRITE (MPMC_DY_TAPR,      0x04);

MEM_WRITE (MPMC_DY_TDAL,      0x06);
MEM_WRITE (MPMC_DY_TWR,      0x01);
MEM_WRITE (MPMC_DY_TRC,      0x06);
MEM_WRITE (MPMC_DY_TRFC,      0x06);
```

```
MEM_WRITE (MPMC_DY_TXSR,      0x06);
MEM_WRITE (MPMC_DY_TRRD,      0x01);
MEM_WRITE (MPMC_DY_TMRD,      0x01);
// Insert some delay first
for (i=0; i<20; i++);

// Insert a NOP command first

MEM_WRITE (MPMC_DY_CNTL,      0x0183);

// Issue a PALL command

MEM_WRITE (MPMC_DY_CNTL,      0x0103);

// Insert some delay again
for (i=0; i<4; i++);

// Insert a number of refresh cycles
MEM_WRITE (MPMC_DY_REF,      0x02);

// Insert more delay
for (i=0; i<8; i++);

// Number of HCLKs between refresh cycles. Since we are
// operating at 75 MHz the value is 75*16 HCLK cycles. The
// hex value of 75 is 0x4D. Program a value slightly
// smaller to be safe.
MEM_WRITE (MPMC_DY_REF,      0x40);

// Initialize the RAS/CAS register
MEM_WRITE (MPMC_DY_RAS_CAS1,     0x0202);

// Initialize the configuration register
MEM_WRITE (MPMC_DY_CONFIG1,     0x14C05488);

// Initialize the MPMC Dynamic Control register to MODE
MEM_WRITE (MPMC_DY_CNTL,      0x83);

// Program the MODE register in the SDRAM by reading from
// the address below

ReadValue = MEM_READ (0x18011000);

// Program the Extended MODE register by reading from the
// address below
ReadValue = MEM_READ (0x18800000);

// Initialize the MPMC Dynamic Control register to NORMAL
MEM_WRITE (MPMC_DY_CNTL,      0x03);

// Enable the buffers by writing to the config register
```

```
MEM_WRITE (MPMC_DY_CONFIG1,     0x14C85488);

// SDRAM Initialization done
```

4.7.6 EBI1 Clock Skew Block

The synchronous memories such as SDRAM and burst NOR flash require a clock to be fed into the memory. This clock is provided by the MSM and is needed to ensure that the clock going out to the memory is skewed (delayed) to meet the setup/hold time requirements on the control/address/data signals with respect to the clock.

The EBI1 clock (pin name: ROM1_CLK/SDRAM1_CLK) coming out of the MSM goes through the path shown in Figure 4-28. The ROM1_CLK clock going out to the external memory is a delayed version of the on-chip clk (HCLK) due to the external loading on the clock line and the delays due to on-chip routing and pads.

For reads from external synchronous memories the read data coming back into the MSM will be “registered” using a fed-back clock (FB_CLK). Both the ROM1_CLK/SDRAM1_CLK and the FB_CLK have the same source, HCLK. However, they follow different paths. One clock is used by the external memories. The other is used by the MSM to capture the data. This flexibility allows us to tune the delay settings to account for delays on the read data path.

For example, if the HCLK period = 13 ns (75 MHz), and memory access time = 6 ns then assuming that FB_CLK has same phase as CLK at memory input, then this leaves about 7 ns for propagation delay all the way to the memory. However, if the FB_CLK is delayed further by about 6 ns then the read data can propagate back to the MSM in clock period (13 ns), which is much more relaxed.

The EBI1 clock skew block applies to both controllers in EBI1: XMEMC and MPMC.

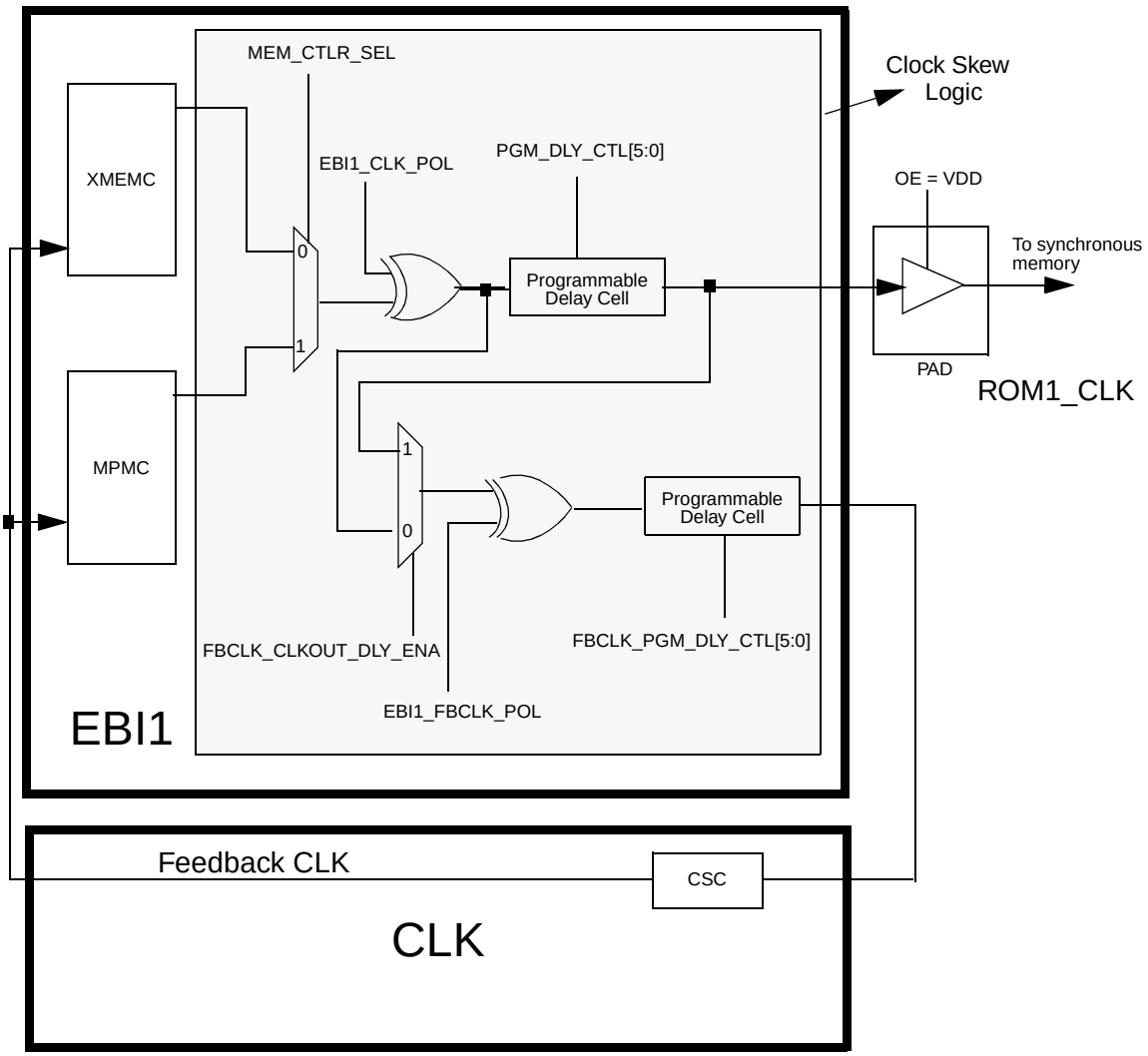


Figure 4-28 EBI1 External Clock Generation

4.7.7 Interface Timing Diagrams

All timing diagrams shown here are only for the XMEM controller. Please refer to the PL172 document for any timing diagrams of the SDRAM.

4.7.7.1 Asynchronous Access

The following figures shows the timing for the accesses to different types of memories. Figure 4-29 shows 16-bit (half word) read and write accesses. Note that the fastest 16-bit read access will require 2 clock cycles and the fastest write access will incur 3 clock cycles. Read/write accesses faster than the ones shown in the diagram are not supported.

Also, between every read and write access there is one idle clock cycle which is incurred because of the arbitration penalty. This cycle ensures a minimum recovery of 1 cycle. Note that this extra cycle is not inserted for consecutive (masters doing a sequential access) reads or writes to the same chip select.

Figure 4-30 shows 32-bit (word) read and write accesses: in this case, a higher number of waitstates (DELTA and WAIT) are programmed for each access. Note that for writes the HOLD must always be programmed to be at least 1.

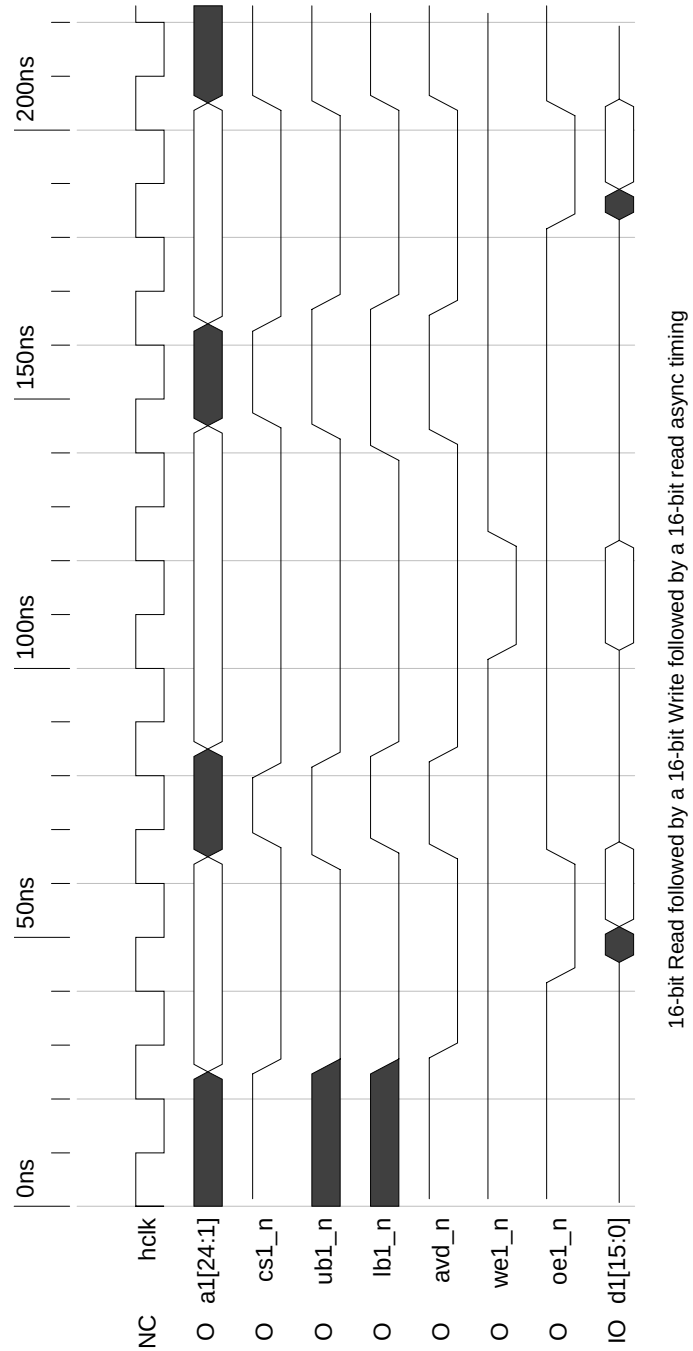


Figure 4-29 16-bit Read, 16-bit Write, 16-bit Read Async Timing

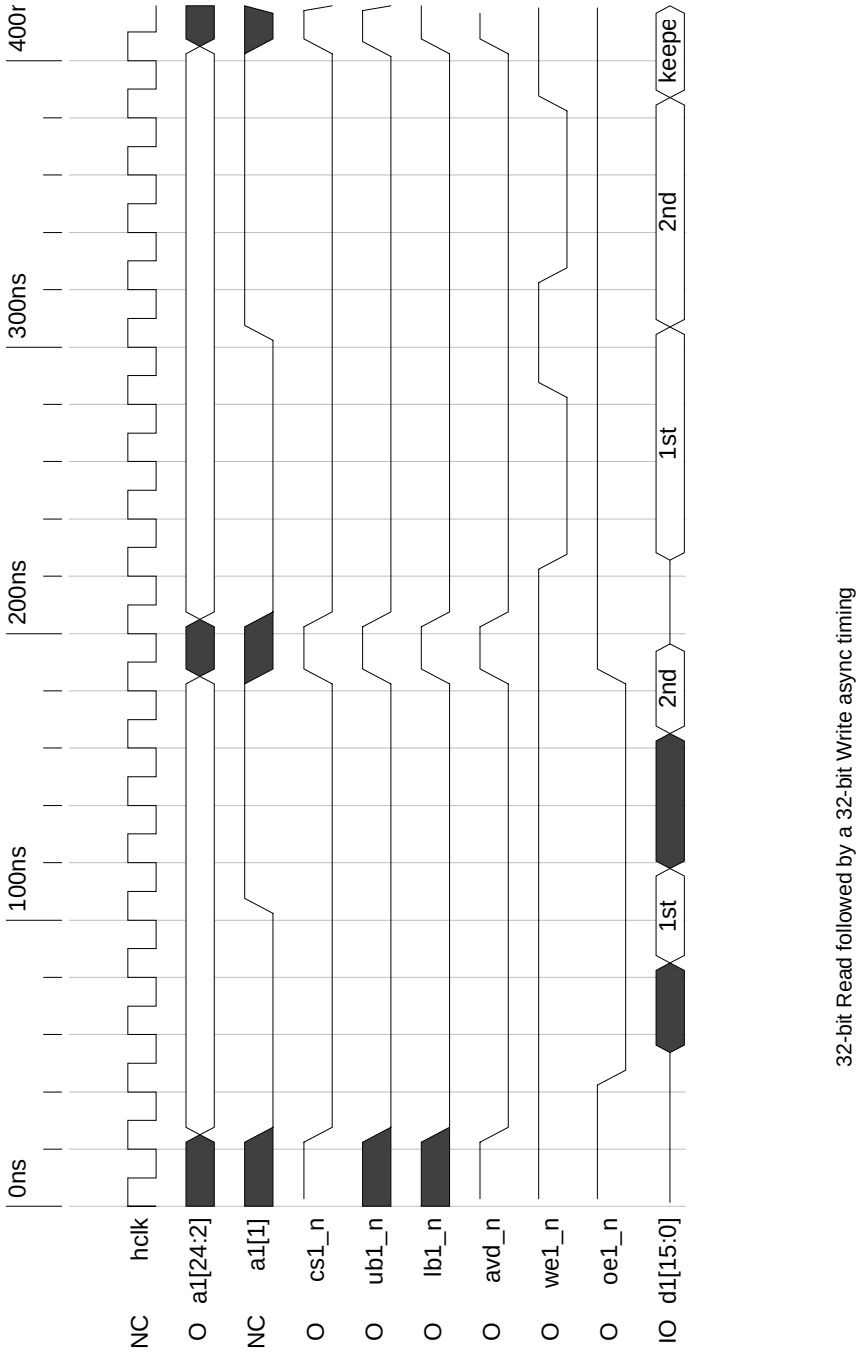


Figure 4-30 32-bit Read Followed by a 32-bit Write Async Timing

4.7.7.2 Page Memory Access Timing

The MSM6100 supports page mode read and page mode write operations.

Figure 4-31 shows a page mode read operation where extra waitstates are inserted when accessing a new page. The DELTA cycles are programmed to be 4 cycles and the WAIT are programmed to 0. This implies 5 cycles for the first access and 1 cycle for subsequent accesses within the page. When the ARM (or any master) tries to do a burst access (INCR4, INCR, WRAP8 etc.) on the AHB bus and the access goes across the page boundary, then the extra DELTA cycles are inserted once again.

Typically, page devices have page access times of around 20-25 ns; so the WAIT setting is normally expected to be at least 1 (2 cycle page access which equals around 40 ns @50 MHz AHB clock) or greater. Figure 4-32 shows a page mode read operation, however, the WAIT_RD setting is 1 now which allows two clock cycles for every page access. This is the most likely setting expected, based on the current page mode devices.

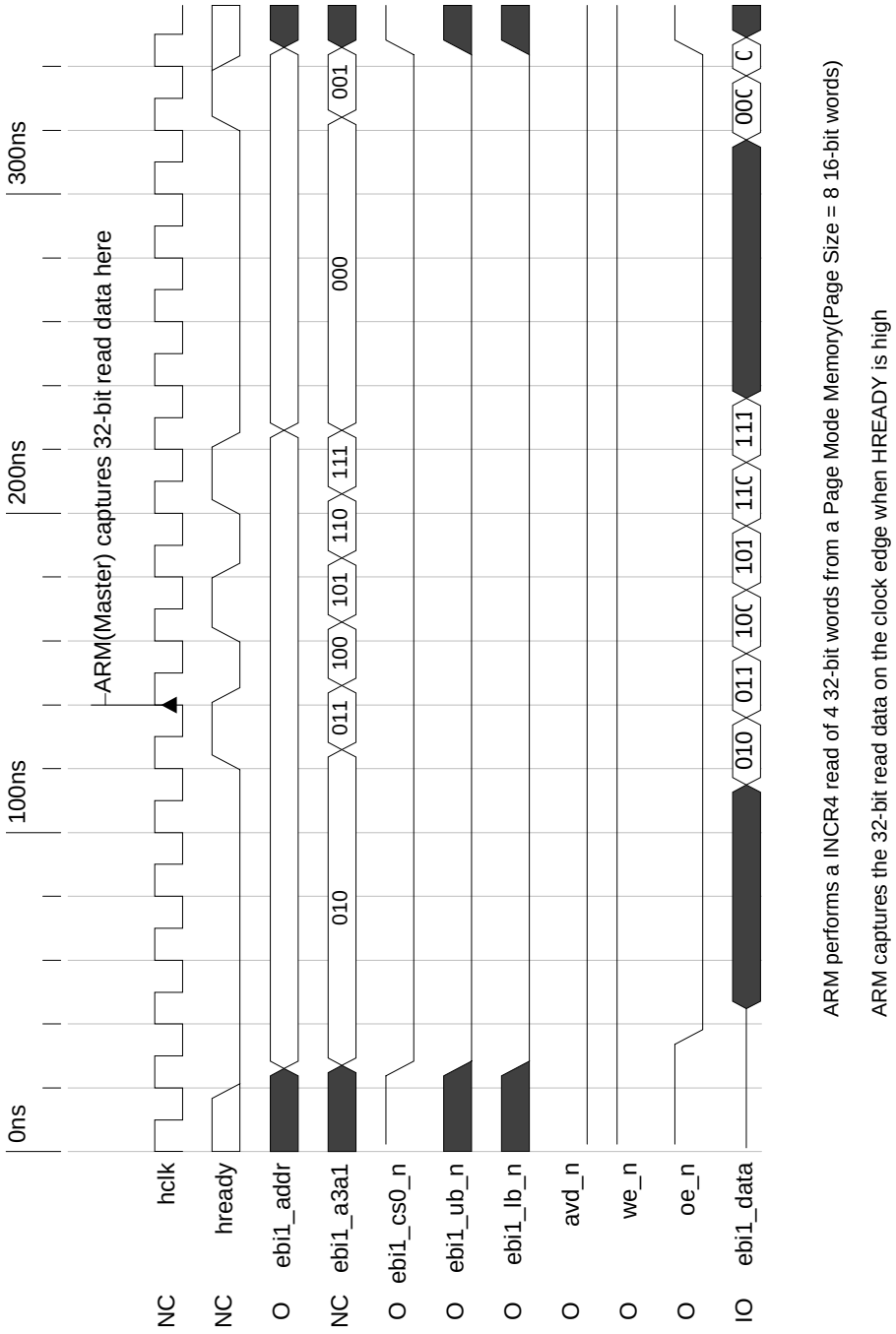
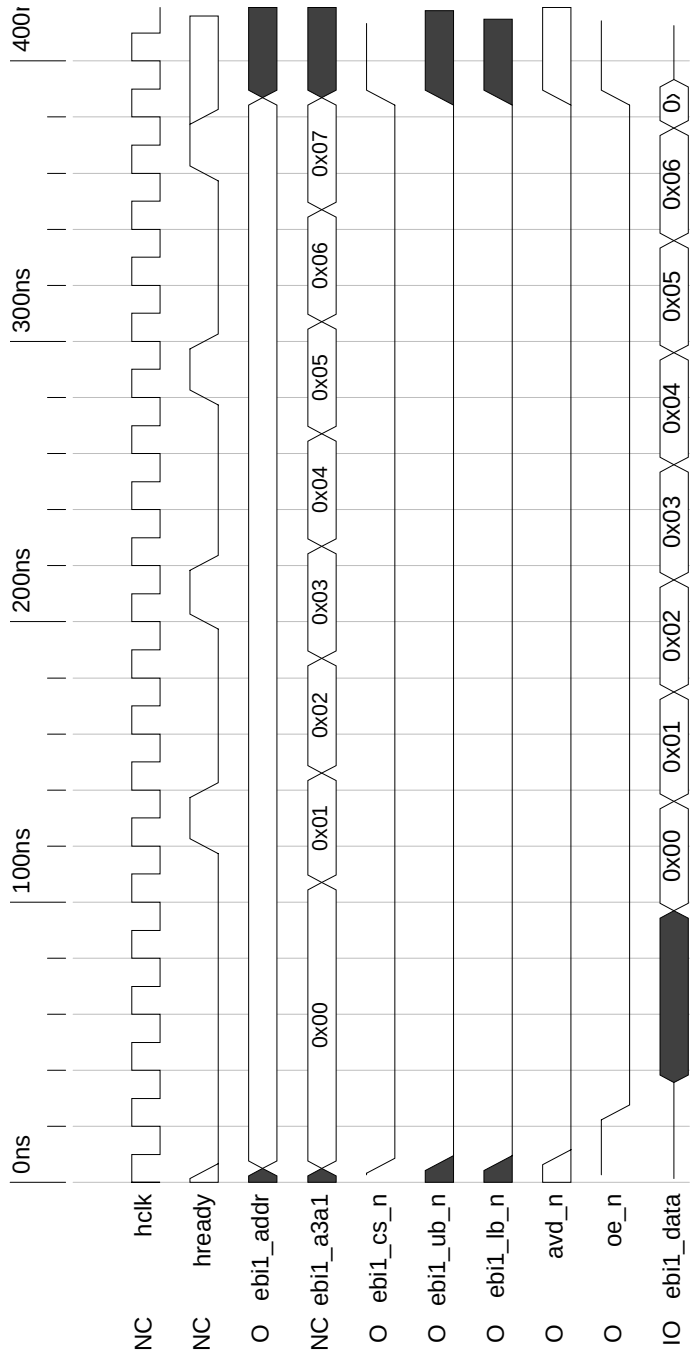


Figure 4-31 Page Read Access With DELTA = 4, WAIT = 0



ARM performs a INCR4 burst of 4 32-bit words from a Page Mode Memory(Page Size = 8 16-bit words)

Access begins at a page boundary and ends within the page

Figure 4-32 Page Read Access With DELTA =3, WAIT = 1

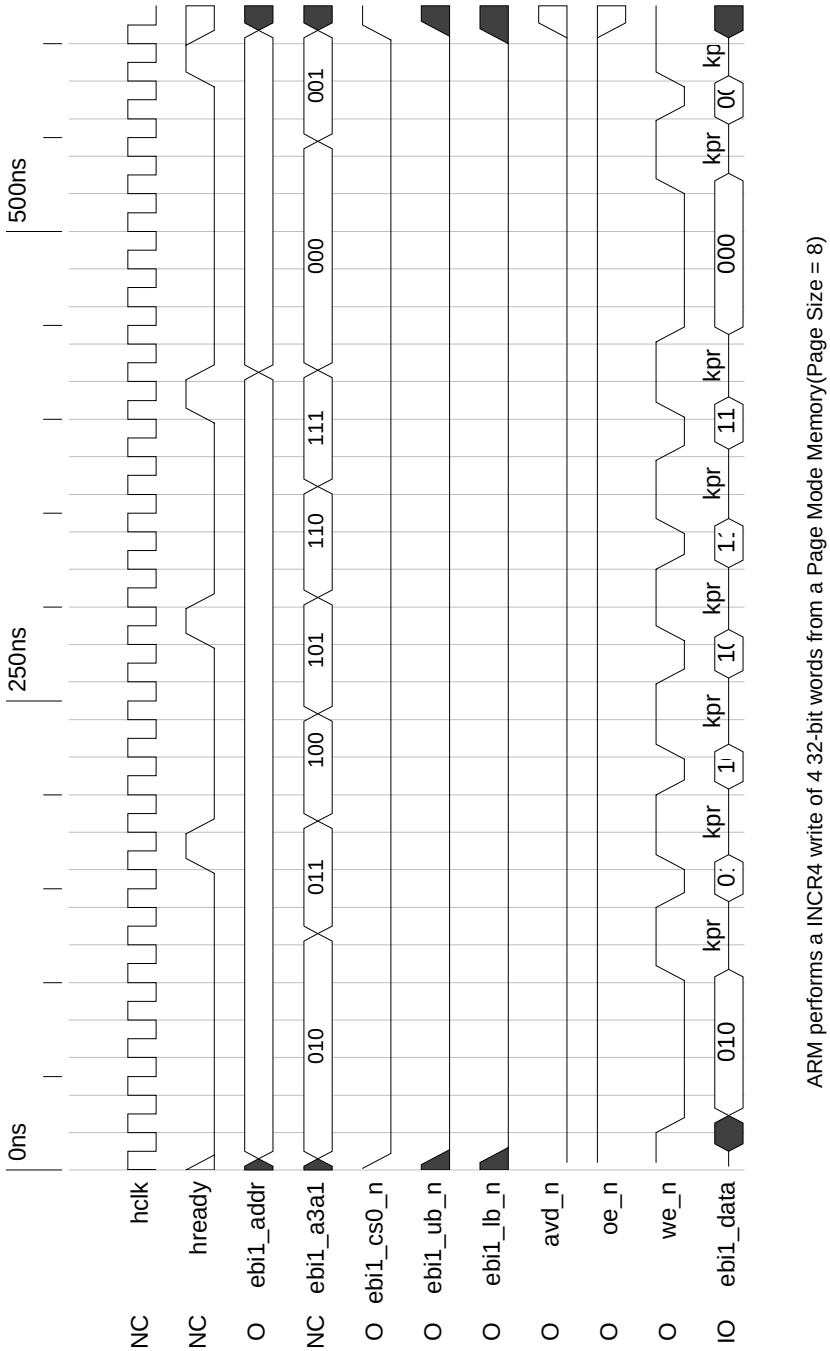


Figure 4-33 Page Write Access With DELTA =4, WAIT = 0

4.7.7.3 Burst Memory Access Timing

The MSM6100 supports both burst read and write timing.

Figure 4-34 shows a WRAP8 access to a burst memory when READ_DATA_PIPELINE_ENA = 0. Note that as described in Section 4.7.4.13 there exists a critical path from the FB_CLK (similar to ROM1_CLK) domain to the HCLK domain.

Figure 4-35 shows the same WRAP8 operation with READ_DATA_PIPELINE_ENA =1. Now the critical path is eliminated. However, as can seen when measuring the total access time for the burst operation, it consumes 1 extra cycle (24 cycles when READ_DATA_PIPELINE_ENA = 0 and 25 cycles when set to 1).

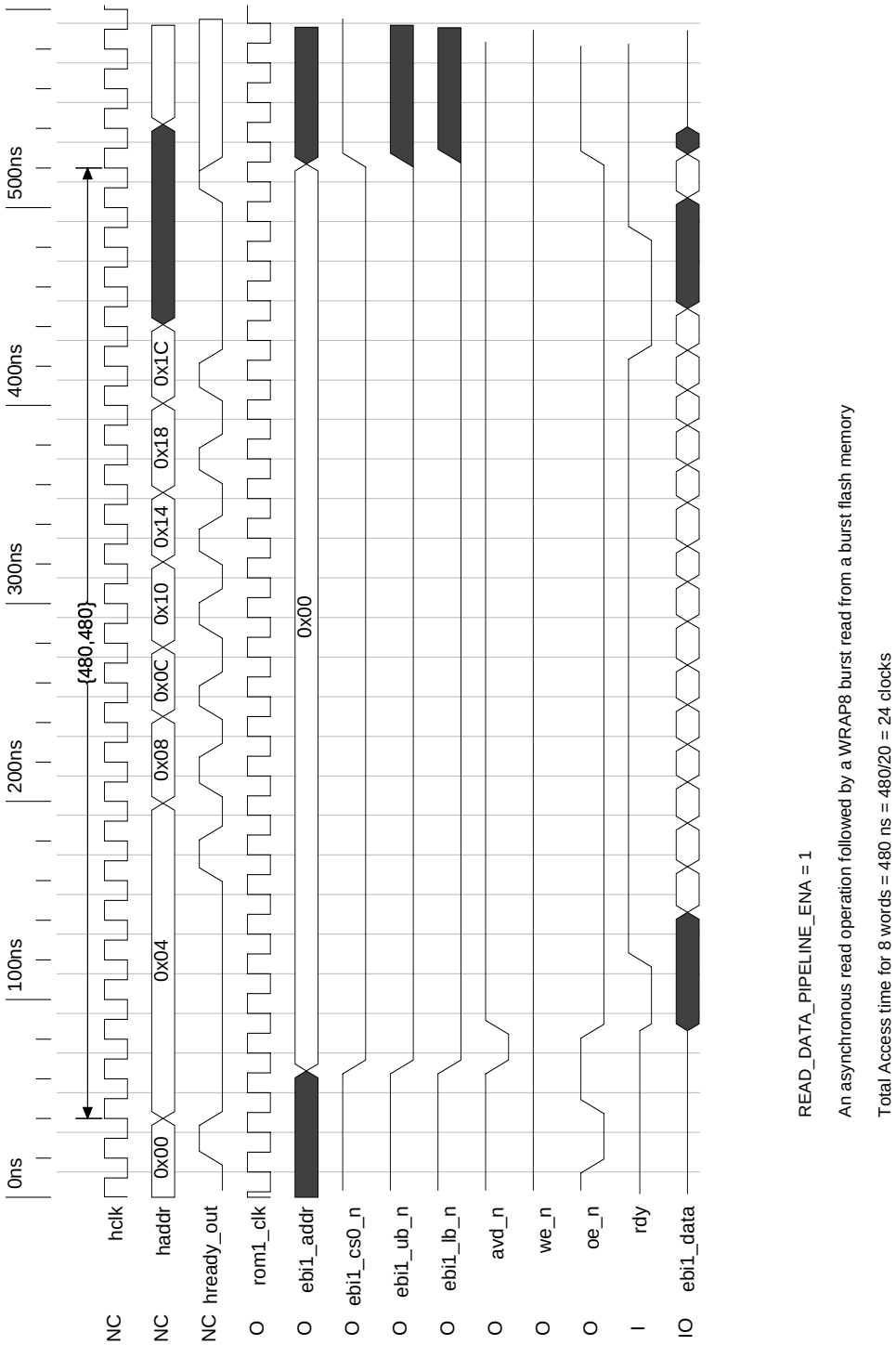


Figure 4-34 Burst Memory Read Timing When READ_DATA_PIPELINE_ENA = 0

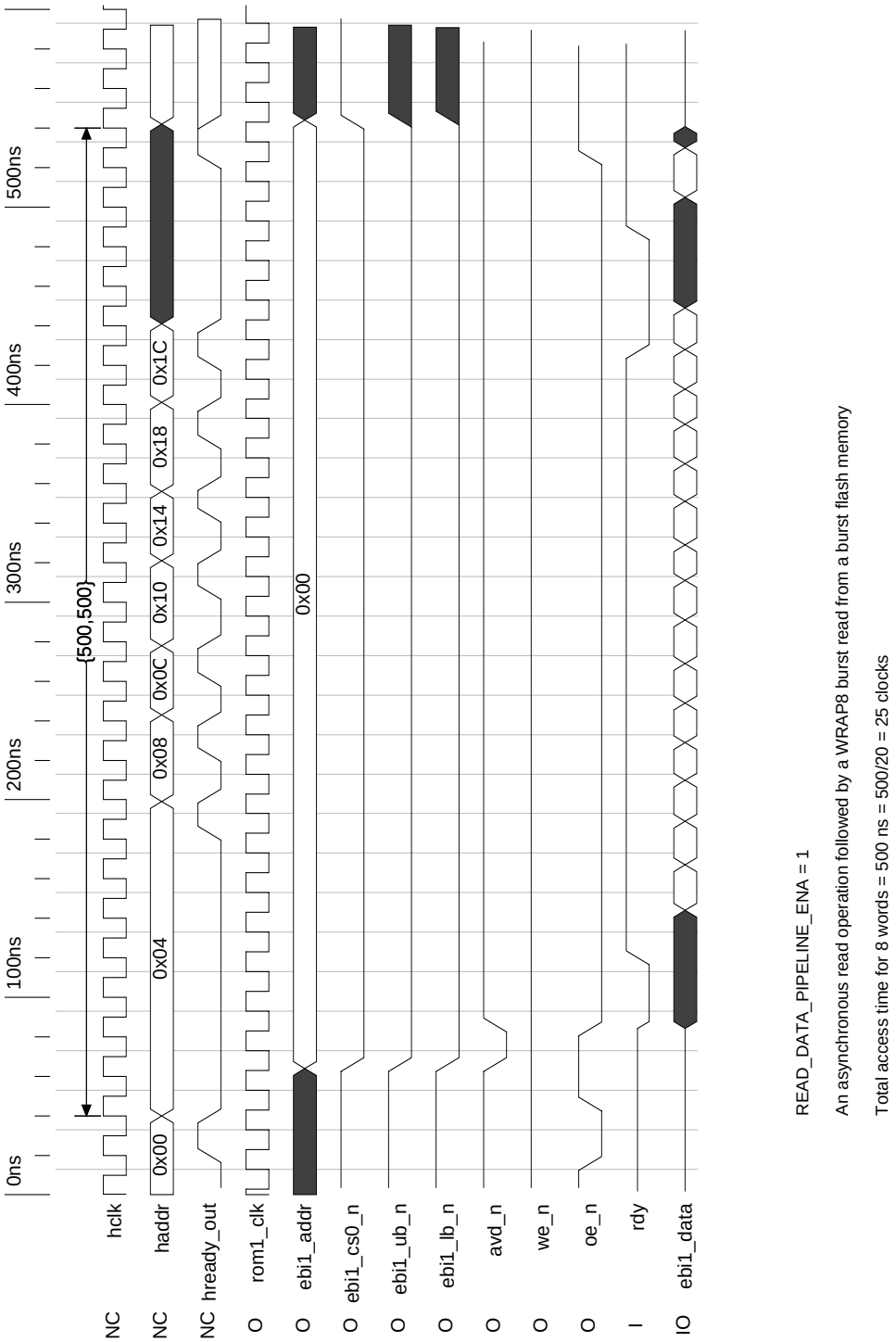


Figure 4-35 Burst Memory Read Timing When READ_DATA_PIPELINE_ENA = 1

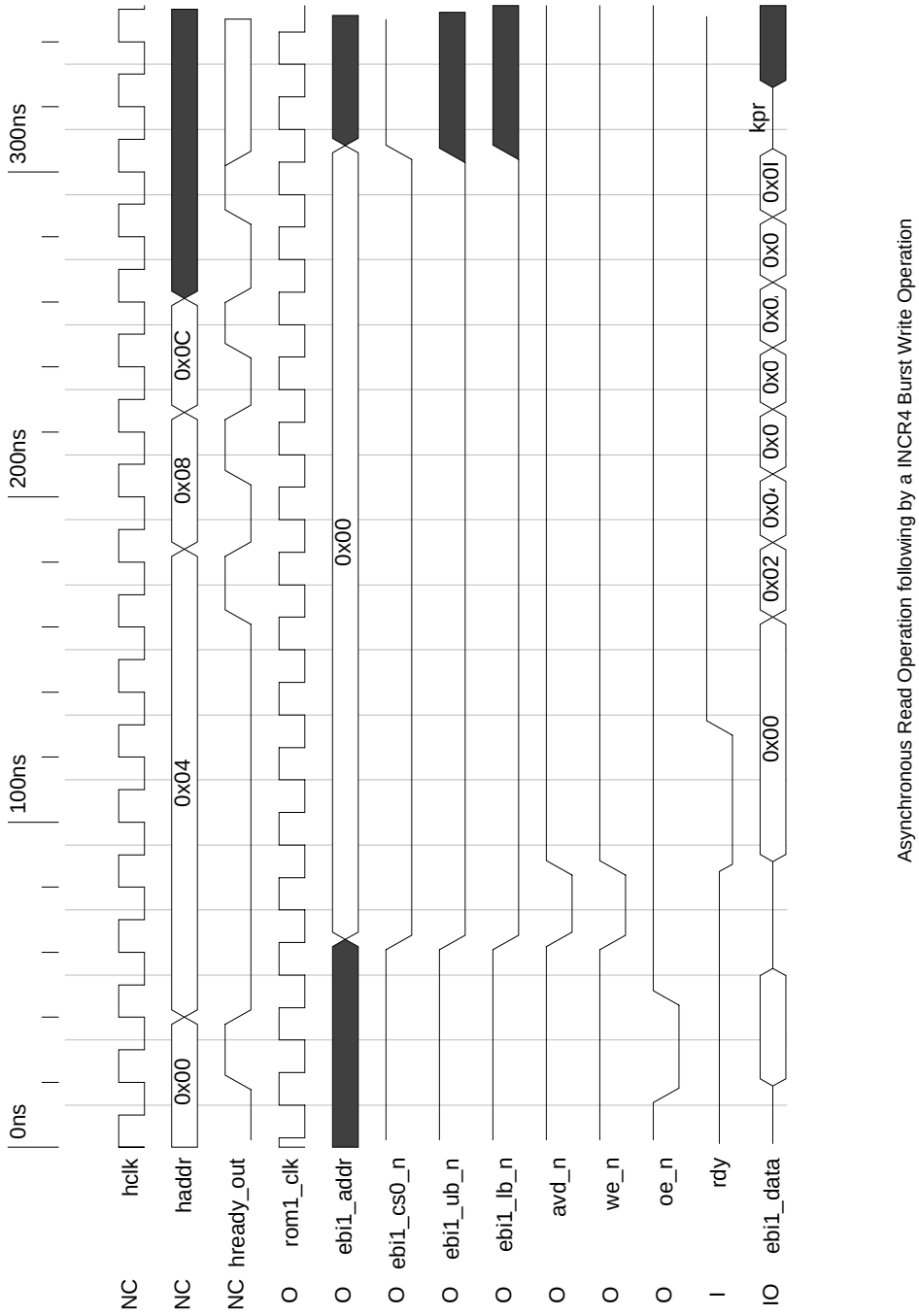


Figure 4-36 Burst Write Operation to a Synchronous PSRAM Device

4.8 External Bus Interface 2

The External Bus Interface 2 (EBI2) is one of the two MSM6100 external memory interfaces. It is targeted to be the interface for slow peripheral devices (i.e., LCD) as well as the NAND Flash memory.

4.8.1 Features

- Support for asynchronous FLASH and SRAM (16-bit and 8-bit). Page mode is NOT supported. Programmable wait states for access, hold, and recovery for all chip selects.
- Interface support for byte addressable 16-bit devices (ub_n and lb_n signals).
- 2 Mbytes of memory per chip select.
- Support for 8-bit wide NAND Flash
- Support for parallel LCD interfaces, port mapped (Intel and Motorola timing) or memory mapped (16-bit and 8-bit)
- Support for system WORD, HWORD, and BYTE accesses to 16-bit and 8-bit wide devices on all chip selects (bus sizing supported for all interfaces with the exception of NAND)
- Support for memory accesses (with exception of NAND) by ARM, DSPs, and GRP. NAND is accessed through a special controller.
- Support for any generic external peripheral whose interface timing is the same as that of asynchronous memories.
- SW controllable write protect feature (against user mode writes).
- Support for accesses to the external SURF CPLD emulating the GPIO2 core for ETM mode.
- Support for an EBI2 pin interface for either 1.867 V or 2.6 V pad power supply voltage.

4.8.2 Chip Selects

Table 4-7 EBI2 Chip Selects

Chip Select	Memory Types Supported
ROM2_CS_N	Asynchronous Flash, Asynchronous SRAM and PSRAM, and NAND Flash.
RAM2_CS_N GP2_CS_N[0] GP2_CS_N[1]	Asynchronous SRAM and PSRAM, asynchronous FLASH, and any other device with asynchronous memory type of interface.

Table 4-7 EBI2 Chip Selects (Continued)

Chip Select	Memory Types Supported
LCD_CS_N	Parallel LCD. Port map devices with either Motorola or Intel type of interface timing. Memory mapped LCD devices with asynchronous memory type of interface.
GPIO2_CS_N	ETM Mode ONLY. For access to GPIO2 emulation CPLD on SURF.

4.8.3 Access Types

In the MSM6100 AHB system, three different transfer sizes can be attempted: WORD, HWORD, and BYTE transfers. If the external memory is being accessed, the memory bus width can be equal to, less than, or greater than the transfer size. Table 4-8 shows the transfers supported for each chip select.

Table 4-8 Access Types by Chip Select

Chip Select	32x16	16	8ubx16	8lbx16	32x8	16x8	8
ROM2_CS_N (XMEM Controller)	Y	Y	Y	Y	Y	Y	Y
ROM2_CS_N (NAND Controller)	N	N	N	N	N	N	Y
RAM2_CS_N	Y	Y	Y	Y	Y	Y	Y
GP2_CS0_N	Y	Y	Y	Y	Y	Y	Y
GP2_CS1_N	Y	Y	Y	Y	Y	Y	Y
LCD_CS_N	Y	Y	N	N	Y	Y	Y
GPIO2_CS_N (ETM Mode Only)	N	Y	N	N	N	N	N

WORD = 32-bit data
HWORD = 16-bit data

4.8.4 EBI2 System

In order to support all the device types mentioned the interface requires three different controllers:

- External memory controller (EBI2_XMEM): Used to access NOR, SRAM, LCD, and generic asynchronous interface devices.
- Nand Controller: Used to access NAND Flash devices.

- Gpio2_bridge: Used to access the external CPLD emulation the internal gpio2_core during ETM mode, or to access the internal gpio2_core itself during NATIVE/TIC modes.

The interface also allows accesses by multiple requesters. An arbiter controls the access of the requesters to the controllers and to the external memory interface pins. For EBI2 the requesters are:

- AMBA AHB Bus: It can access the external memory controller and the gpio2_bridge. The masters on this bus attempting accesses to EBI2 is the ARM processor only.
- DSP AHB Bus: It can access the external memory controller only. The masters on this bus attempting accesses to EBI2 are the ADSP DME and the MDSP DME.
- GRP AHB Bus: It can access the external memory controller only. The masters on this bus attempting accesses to EBI2 is the GRP only.
- NAND Controller: This is a different type of requester than the previous three. The NAND controller initiates requests to the EBI2 after being configured by the AMBA_AHB bus through the FPB interface.

Arbitration is required to allow the access of multiple requesters to the external memory interface. The arbiter uses a fixed priority scheme that can be programmed into two priority modes as shown below. Arbitration requires polling the different requests placed by the different requesters and granting the access to that request with the higher priority. Granting an access requires controlling the datapath from the bus to the controller and from the controller to the pins.

Table 4-9 EBI2 Priority Options

Option	Priority 1 (Higher)	Priority 2	Priority 3	Priority 4 (Lower)
Mode 0 EBI2_CFG[0] = 0 Power up default	ARM	DSP	GRP	NAND
Mode 1 EBI2_CFG[0] = 1	DSP	ARM	GRP	NAND

4.8.5 EBI2 External Memory Controller

4.8.5.1 FLASH and SRAM Interface

This interface provides access to asynchronous FLASH and SRAM as well as to any memory-mapped peripheral that requires similar timing.

4.8.5.1.1 Features

- Support of bsizing operation to allow WORD, HWORD, and BYTE accesses to 16-bit and 8-bit memories.

- Support for recovery/turnover wait states (RECOVERY) insertion to the beginning of the current chip select access when the previous access was a read to a different chip select, or a when the current access is a write and the previous access was a read to the same chip select. The number of RECOVERY cycles is that required by the previous memory accessed.
- Support for address to chip select assertion setup wait states (CS_SETUP) at the beginning of the access. This feature helps support some types of pseudorams on the market. When accesses to the chip select using this features are consecutive, this feature also provides the high time for chip select required by the mentioned pseudorams. In general, in can be used to provide extra setup time between address and chip select assertion if needed.
- Since the RECOVERY and CS_SETUP wait states are applied at the beginning of the access, only the field with the greatest number of wait states takes effect.
- Support for WAIT_RD and WAIT_WR wait states to control access length (how long CS_N, WE_N and OE_N assert). Both WAIT_RD and WAIT_WR waits states require a minimum programmed value of 1. That is WAIT_RD(min) = 1, WAIT_WR(min) = 1.
- Support for DELTA_RD and DELTA_WR wait states for additional access waits states (similar to WAIT-RD and WAIT_WR). However, this type of wait states only applies to single (non-bsized accesses) and to the first access of a bus sized transfer only.
- Support for hold time between the we_n signal rising and the address/cs_n/data signals changing (HOLD_WR wait states). A minimum of 1 hold wait state is required (HOLD_WR = 1) for proper operation of the interface.
- Support for hold time between the oe_n signal rising and the address/cs_n signals changing (HOLD_RD wait states). Required by specific types of peripherals. Minimum value required (and most common value to use) is HOLD_RD = 0.

4.8.5.1.2 Timing

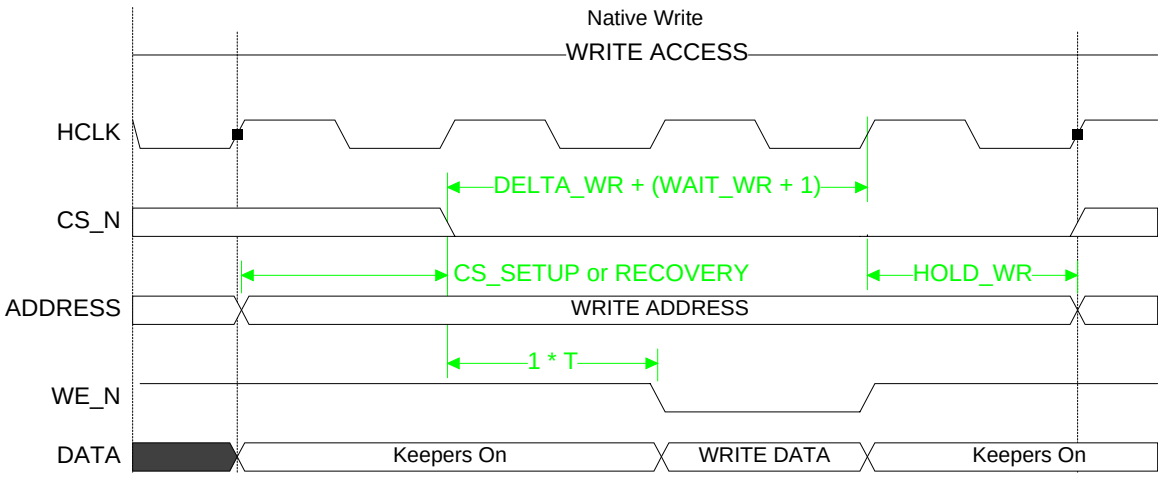


Figure 4-37 Native Write Timing

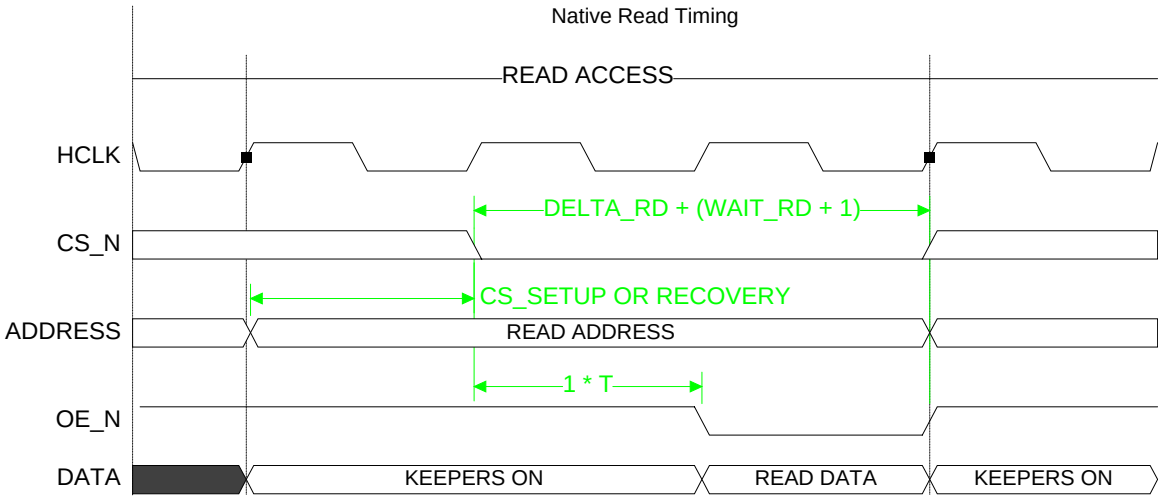


Figure 4-38 Native Read Timing

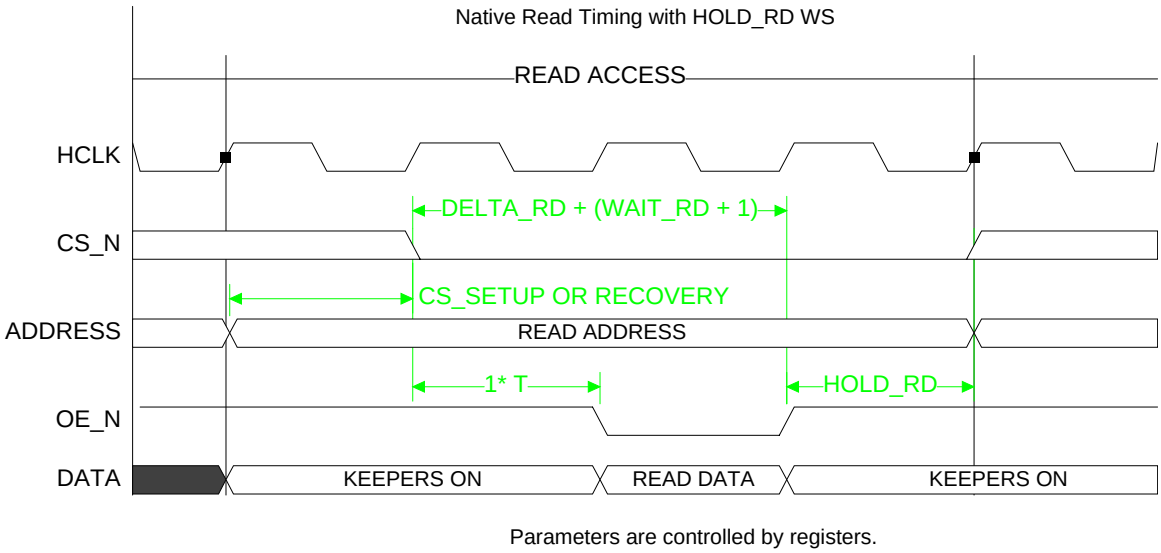


Figure 4-39 Native Read With HOLD_RD Wait State

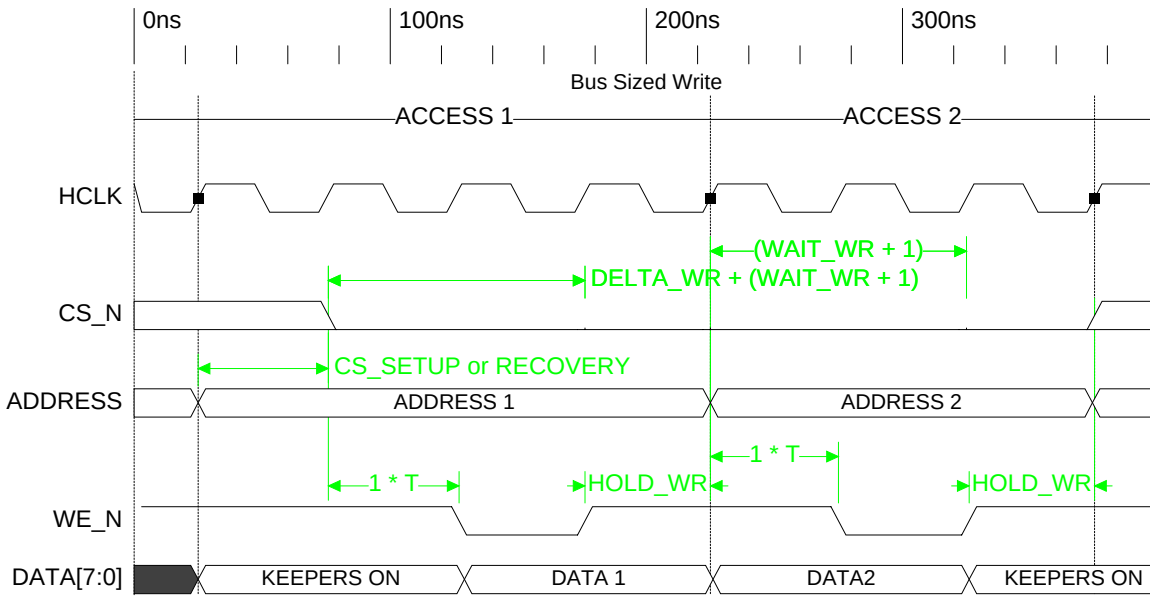


Figure 4-40 Bus-Sized Write Timing

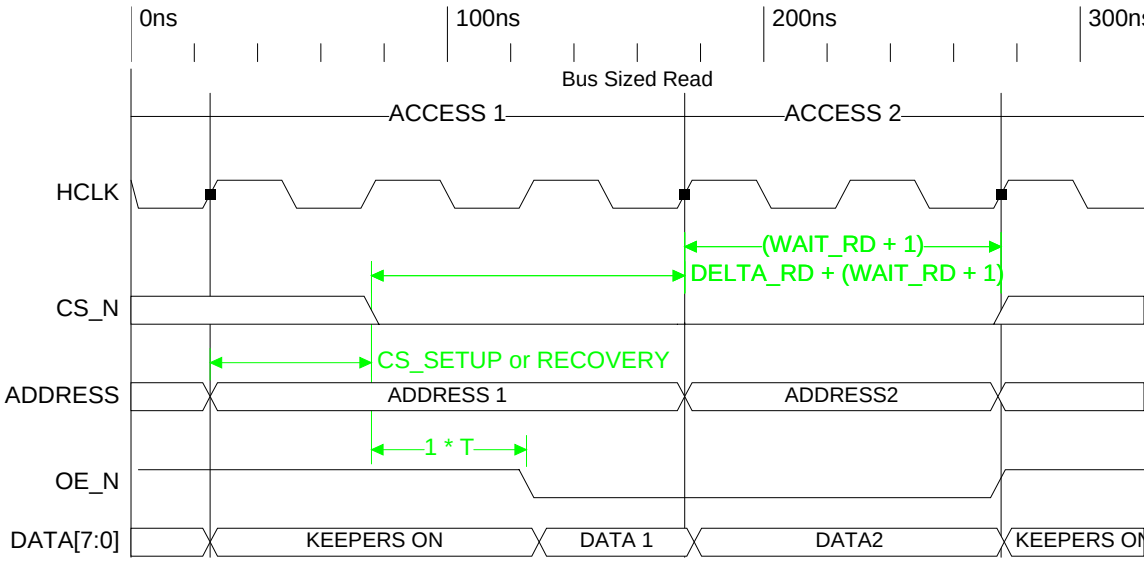


Figure 4-41 Bus-Sized Read Timing

4.8.5.1.3 Minimum Programmed Values Requirement

The table below shows the minimum programmed values requirements on interface configuration 0 registers.

Table 4-10 EBI2 XMEM CFG 0 Registers

REGISTERS	CS_SETUP	RECOVERY	HOLD_WR	HOLD_RD	DELTA_WR	DELTA_RD	WAIT_WR	WAIT_RD
GP0_CFG0	0	0	1	0	0	0	1	1
GP1_CFG0								
RAM2_CFG0								
ROM2_CFG0								

4.8.5.2 LCD Interface

The LCD controller on the MSM6100 is upgraded to offer more flexibility than previous MSM LCD controller designs. In the MSM6100 there is an assigned chip select, LCD_CS_N, that supports memory- mapped and port-mapped LCD interfaces with either the Motorola (6800) or the Intel (8080) parallel interface types of timing. An additional LCD_EN signal is required for Motorola's interface. The memory-mapped or port-mapped (Intel type) parallel LCD device could be connected to any chip select in EBI2. Bus sizing capability was added in the MSM6100 to optimize LCD accesses. Since EBI2 supports both 8-bit and 16-bit devices, most of the external LCD controllers are supported.

4.8.5.2.1 Features

- Support for both memory-mapped and port-mapped parallel interfaces.
- Support for both Motorola and Intel types of port mapped parallel LCD interfaces.
- Support of a bsizing operation to allow WORD and HWORD accesses to parallel 16-bit data parallel interfaces, and to allow WORD, HWORD, and BYTE access to parallel 8-bit data parallel interfaces. This allows the optimization of software for more efficient data transfers to and from a LCD.
- Support for access times of up to 2000 ns (at 75 MHz AHB bus speed).
- Separate configuration control for write and read operations.
- For Motorola Interface mode, support lcd_enable high times of up to 1000 ns (at 75 MHz AHB bus speed).
- For Intel Interface mode, support for a hold time from WR_N deasserting to CS_N/RS/ADDRESS changing and from OE_N deasserting to CS_N/RS/ADDRESS changing.
- For Intel Interface mode, support for setup time from RS/ADDRESS stable to CS_N/OE_N/WE_N asserting.

4.8.5.2.2 Timing

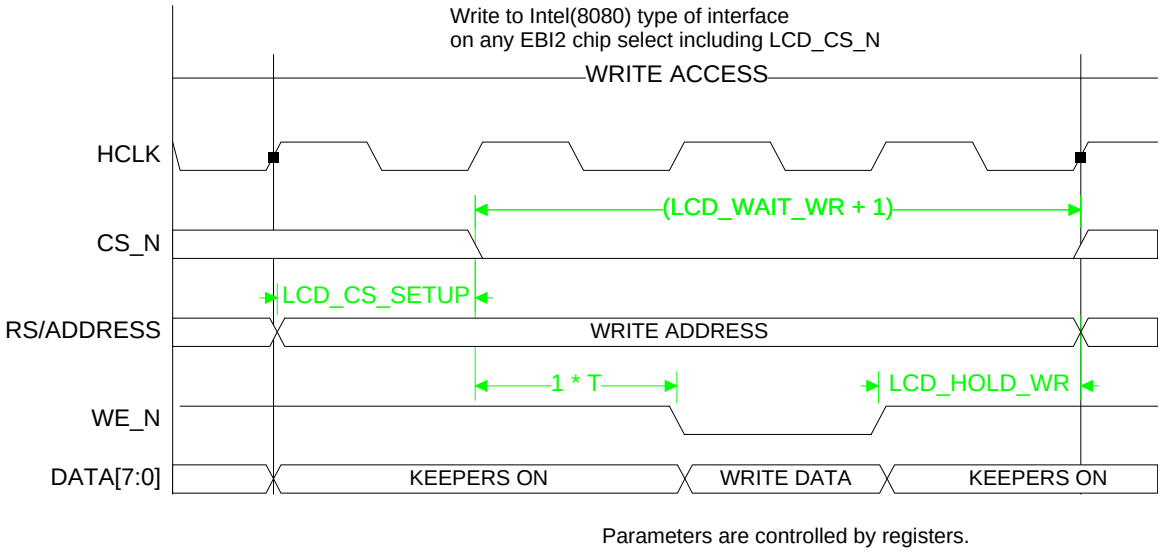


Figure 4-42 Timing For Parallel LCD Write to Intel(8080)-Type Interface

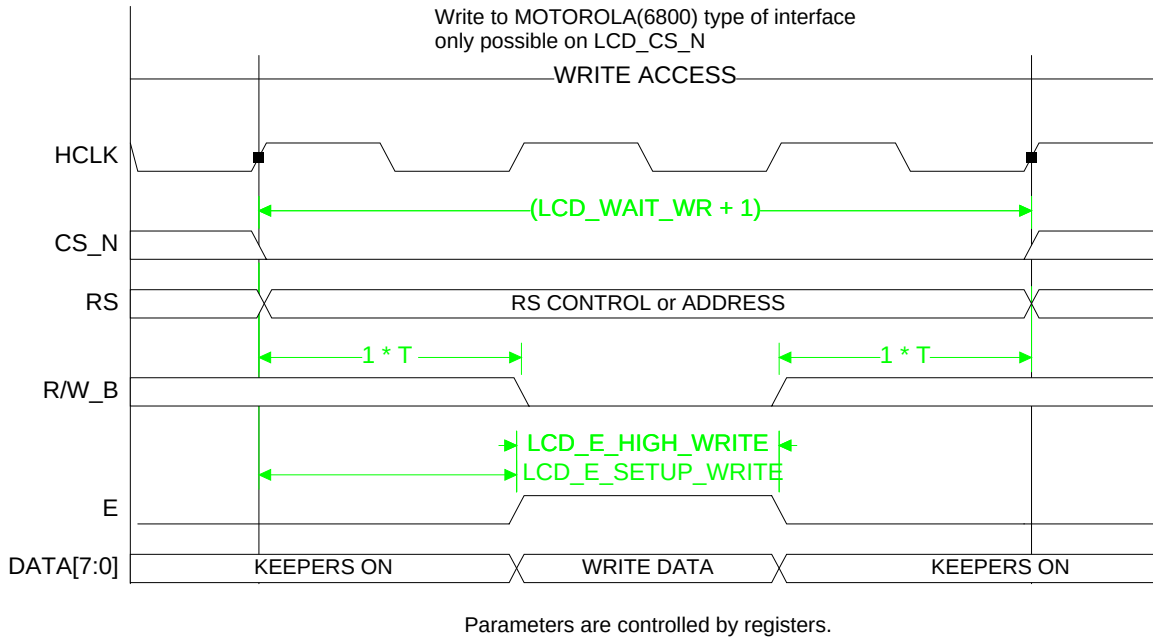


Figure 4-43 Timing For Parallel LCD Write to Motorola (6800)-Type Interface

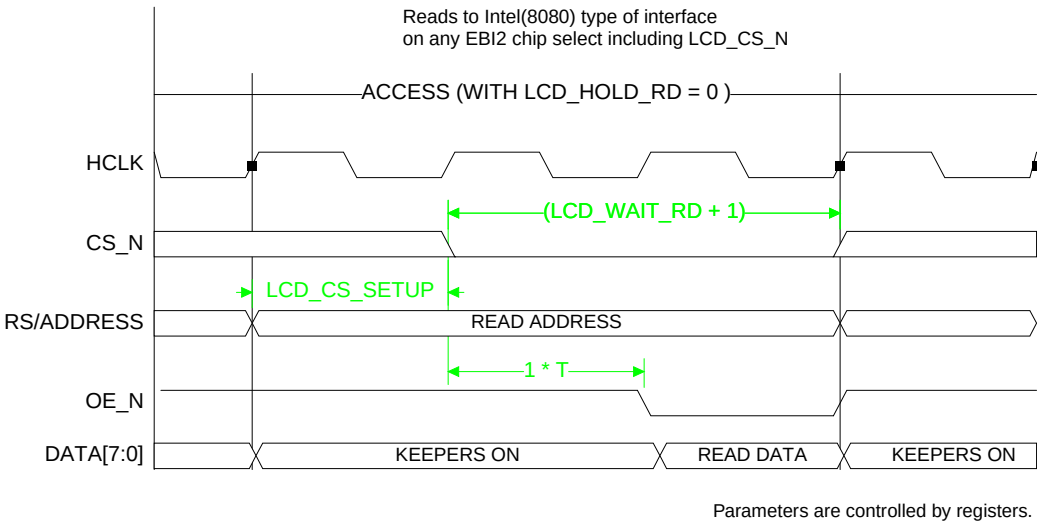


Figure 4-44 Timing for Parallel LCD Read to Intel (8080)-type Interface, Zero HOLD_RD WS Shown.

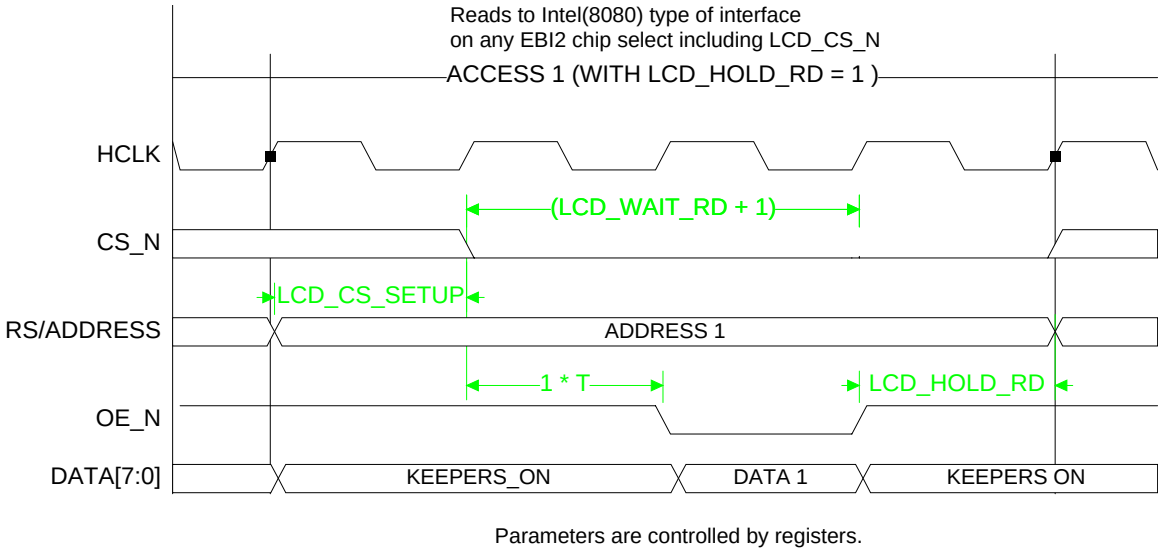


Figure 4-45 Timing for Parallel LCD Read to Intel (8080)-Type Interface, One HOLD_RD WS Shown.

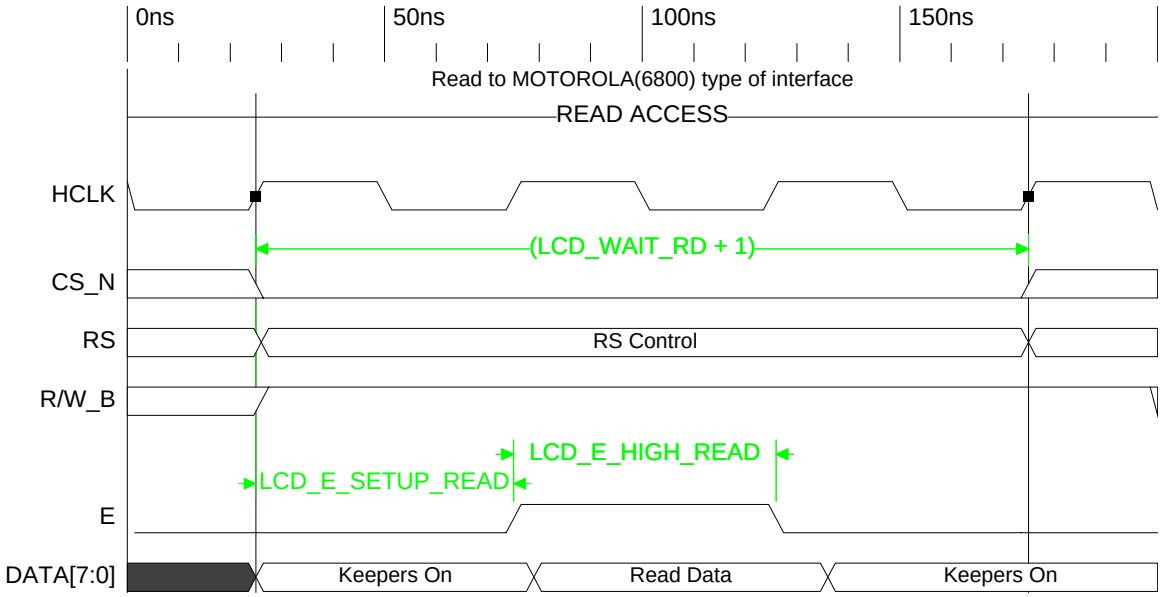
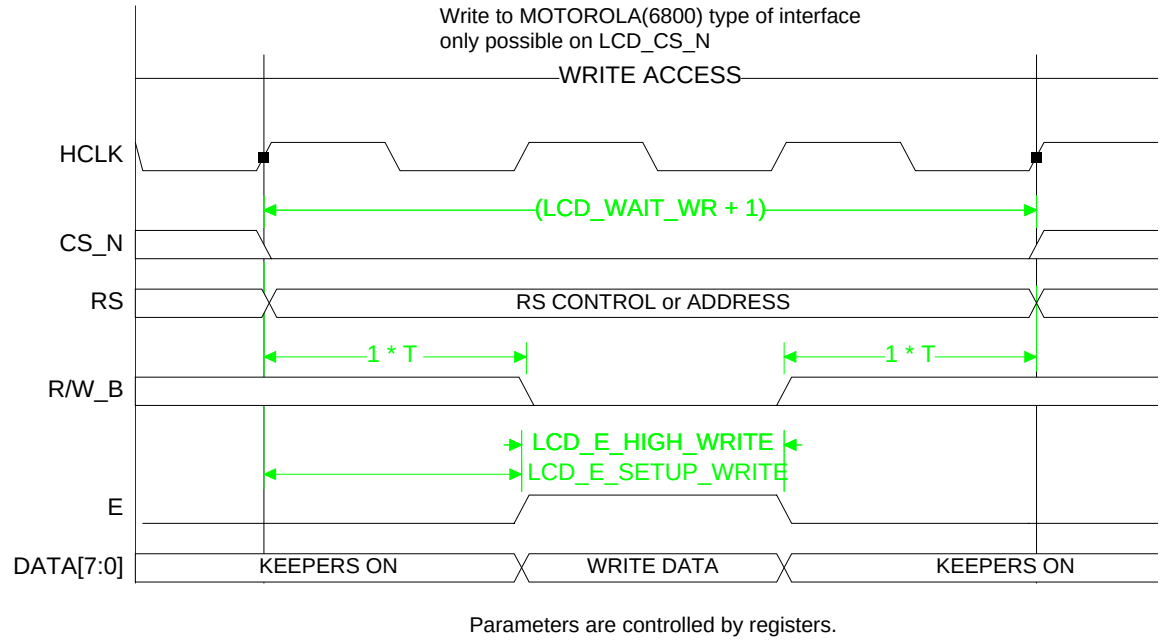


Figure 4-46 Timing for Parallel LCD Read to Motorola(6800)-Type interface



Parameters are controlled by registers.

Figure 4-47 Timing For Parallel LCD Bus-Sized Write to Motorola(6800)-type Interface

4.8.5.2.3 Minimum Programmed Values Requirement

The table below shows the minimum programmed values requirement on the LCD configuration register.

Motorola Interface

Table 4-11 LCD_CFG0 Register

REGISTER	CS_SETUP	RECOVERY	HOLD_WR	HOLD_RD	WAIT_WR	WAIT_RD
LCD_CFG0	0	0	0	0	2*	2*

Table 4-12 LCD_CFG1 Register

REGISTER	LCD INTERFACE TYPE	LCD E SETUP_WR	LCD E SETUP_RD	LCD E HIGH_WR	LCD E HIGH_RD
LCD_CFG1	1	1	1	1	1

- WAIT_RD should always be ≥ LCD_E_SETUP_RD + LCD_E_HIGH_RD.
- WAIT_WR should always be ≥ LCD_E_SETUP_WR + LCD_E_HIGH_WR.

HOLD_WR, HOLD_RD, and CS_SETUP do not apply to Motorola interface accesses; however, for correct operation in this mode, program all these fields to 0.

Intel Interface

Table 4-13 LCD_CFG0 Register

REGISTER	CS_SETUP	RECOVERY	HOLD_WR	HOLD_RD	WAIT_WR	WAIT_RD
LCD_CFG0	0	0	1	0	1	1

Table 4-14 LCD_CFG1 Register

REGISTER	LCD INTERFACE TYPE	LCD E SETUP_WR	LCD E SETUP_RD	LCD E HIGH_WR	LCD E HIGH_RD
LCD_CFG1	0		0	0	0

LCD_E_SETUP_WR, LCD_E_SETUP_RD, LCD_E_HIGH_WR, and LCD_E_HIGH_WR do not apply to Intel interface accesses.

4.8.6 NAND Flash Memory Interface

4.8.6.1 Overview

The MSM6100 device supports NAND Flash memory to improve data storage capability.

Here is an overview for the MSM6100 NAND Flash controller.

- Generic NAND flash memory interface supports 8-bit NAND Flash devices from manufacturers such as Samsung, Toshiba, etc.
- NAND Interface uses the EBI2 ports. (Shares these ports with other external devices, i.e., external SRAM.)
- Interface performs one page at a time data transfers to/from NAND device. No Sequential page accesses supported.
- Interface has an internal SRAM buffer for one data page (512 bytes) with the ARM processor directly accessing this internal buffer memory through the AHB bus through 32-bit word accesses.
- Error correction coding (ECC): Interface automatically corrects a maximum of four bits of error in one page; and it is also able to detect uncorrectable bit errors and ECC self errors. ECC function can be disabled by the MSM processor.

4.8.6.2 Supported Commands

The MSM6100 NAND Flash controller supports the following command as shown in Table 4-15.

Table 4-15 NAND Flash Controller Commands

	First Command	Second Command
Page Write	80	10H
Page Read	00	—
Flag Read (Spare Byte Reads)	50	—
Reset	FF	—
Auto Block Erase	60	D0
Status Read	70	—
ID Read	90	—

Table 4-16 shows the sequence of NAND memory accesses generated by each command. Each command sequence is triggered by a write to the NAND_FLASH_CMD register where each command has a corresponding value to be programmed into the register.The command register is NAND_FLASH_CMD. Address is 0X6400 0300.

Table 4-16 Command Sequences Generated by Writes to the NAND_FLASH_CMD Register

			Command Structure			
Bit [2:0]	**Name of Command**	**Description**	**Command**	**Address**	**Data**	**Command**
000	Software reset NAND flash controller	Ongoing operation will abort, NAND controller will be forced into idle, and the configure register is reset to power up default value	---	---	---	---
001	page_read	Transfer one page of data from the NAND flash to the SRAM buffer. The NAND flash page address is provided by the NAND_FLASH_ADDR register	00H	4 Byte Writes _____ 1st: [A7~A0] 2nd: [A16~A9] 3rd: [A24~A17] 4th: [A32~A25] A32='0'	512 + n Byte Reads n=13 for ECC on n=16 for ECC off _____ Byte [0] Byte [1] .... Byte[511] ... .. Byte[511 + n -1] Byte[511 + n]]	

Table 4-16 Command Sequences Generated by Writes to the NAND_FLASH_CMD Register

			Command Structure			
Bit [2:0]	**Name of Command**	**Description**	**Command**	**Address**	**Data**	**Command**
010	flag_read	This command reads one byte from the spare area of the NAND flash. The page address and byte address are all provided by the NAND_FLASH_ADDR register. Some NAND flash makers (e.g., SAMSUNG) use the spare area to mask bad blocks upon shipping. After a flag_read operation, a reset operation is suggested before issuing a page_write command.	50H	4 Byte Writes _____ 1st: [A7~A0] 2nd: [A16~A9] 3rd: [A24~A17] 4th: [A32~A25] A32='0'	1 Byte Read _____ Byte[Address]	---
011	page_write	Transfer one page of data from the SRAM buffer to the NAND flash. The page address is provided by the NAND_FLASH_ADDR register.	80H	4 Byte Writes _____ 1st: [A7~A0] 2nd: [A16~A9] 3rd: [A24~A17] 4th: [A32~A25]	[512 + n Byte Writes n=13 for ECC on n=16 for ECC off _____ Byte [0] Byte [1] .... Byte[511] ... .. Byte[511 + n -1] Byte[511 + n]]	10H
100	block_erase	Erase one block of NAND flash memory. The block address is provided by the NAND_FLASH_ADDR register	60H	3 Byte Writes _____ [A16~A9] [A24~A17] [A32~A25]		D0H

Table 4-16 Command Sequences Generated by Writes to the NAND_FLASH_CMD Register

			Command Structure			
Bit [2:0]	**Name of Command**	**Description**	**Command**	**Address**	**Data**	**Command**
101	ID_fetch	Read the NAND flash maker ID and device ID. (This ID information can be used to determine the actual size of the NAND device.)	90H	1 Byte Writes Address:(00)H	2 Byte Reads 1st Byte for maker Id 2nd Byte for device ID	---
110	status_check	Check the NAND flash device status.	70H	---	1 Byte Read Device Status Byte	---
111	reset NAND flash memory	This reset operation is suggested upon power-up. This reset operation is also suggested after a soft-reset command aborts an ongoing operation.	FFH	---	---	---

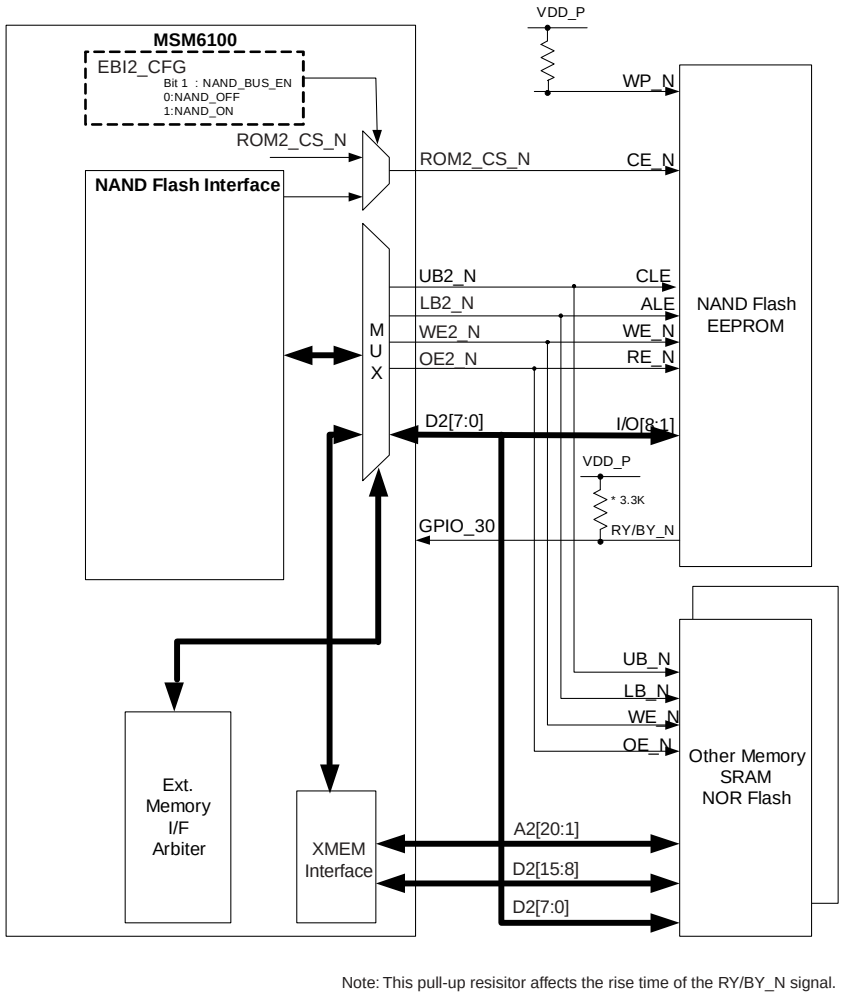
Example: Definition of bit HEX data bit assignment Serial data input: 80

Table 4-17 Example Data Assignment

I/O7	I/O6	I/O5	I/O4	I/O3	I/O2	I/O1	I/O0
1	0	0	0	0	0	0	0

4.8.6.3 NAND Flash Controller Block Diagram

The interface between the MSM6100 device and NAND Flash shown in Figure 4-48.



Note: This pull-up resisitor affects the rise time of the RY/BY_N signal.

Figure 4-48 Interface Connection NAND Flash and MSM6100 Device

4.8.6.4 Configuration Registers

Table 4-18 Chip Select Configuration Registers for EBI2 External Memory Controller (LCD)

Chip Select	Configuration Register
ROM2_CS_N (NAND Enabled)	NAND_FLASH_CFG

4.8.6.5 Access Description

NAND FLASH chip select is asserted by:

2 + WAIT_STATE_SEL AHB cycles. The WAIT_STATE_SEL field affects both read and writes.

4.8.6.5.1 Wait States for NAND Interface

The MSM6100 allows 8 wait state settings to support different NAND/System speed requirements.

The wait state modes are defined as follows:

0x6400 031C **NAND_FLASH_CFG**

Type: Write
Clock: HCLK
Bits: 10:0

Bit	Name	Description
4:2	WAIT_STATE_SEL	Configuration NAND flash interface signal duration based on a different AHB clock frequency. Values 0 to 3 represent the number of wait states added. Values 4 to 7 represent the wait state mode. 000: No wait state, 2 HCLK cycles per access. 001: 1 wait state, (default), 3 HCLK cycles per access. 010: 2 wait states, 4HCLK cycles per access. 011: 3 wait states, 5HCLK cycles per access. 100: wait states mode 4, 6 HCLK cycles per access. 101: wait states mode 5, 9HCLK cycles per access. 110: wait states mode 6, 17 HCLK cycles per access. 111: wait states mode 7, 32 HCLK cycles per access.

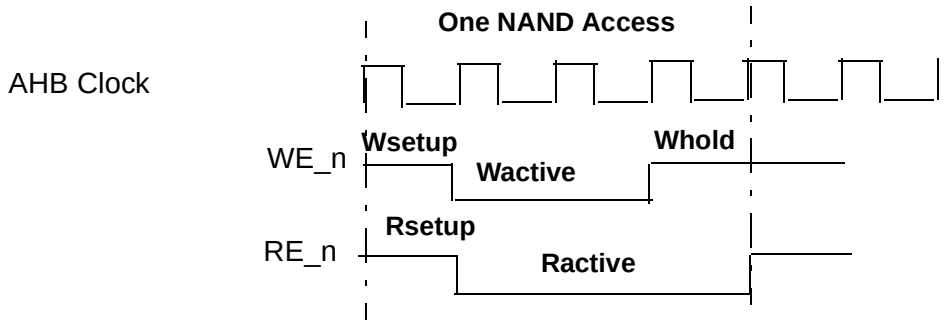


Figure 4-49 Basic NAND parameter

Table 4-19 Wait State Table for Writes

WAIT STATES	Wsetup	Wactive	Whold	Total
0	0.5	1	0.5	2
1	0.5	2	0.5	3
2	1	2	1	4
3	1	3	1	5
4	1	3	2	6
5	1	5	3	9
6	1	8	8	17
7	1	16	15	32

Table 4-20 Wait State Table for Reads

WAIT STATES	Rsetup	Ractive	Total
0	1	1	2
1	1	2	3
2	1	3	4
3	1	4	5
4	2	4	6
5	4	5	9
6	9	8	17
7	17	15	32

4.9 Mode Select and Emulation Considerations

4.9.1 Mode Selection Inputs

The MSM6100 device can be put into eight different operating modes. The mode is selected by the MODE[1:0] and WDOG_EN inputs. The different modes are listed in Table 4-21 along with the binary value applied to MODE[1:0].

Table 4-21 Operating Modes

MODE[1,0], WDOG_EN	Operating Mode
000	Native, ARM JTAG, WDOG disabled
001	Native, ARM JTAG, WDOG enabled

Table 4-21 Operating Modes

MODE[1,0], WDOG_EN	Operating Mode
010	ETM, ARM JTAG, WDOG disabled
011	ETM, ARM JTAG, WDOG enabled
100	Reserved
101	Reserved
110	Native, MSM JTAG, WDOG disabled
111	Reserved

4.9.2 NATIVE Mode

NATIVE mode is default operating mode for the MSM6100 device. When the MSM6100 device operates in NATIVE mode, it uses the internal microprocessor. NATIVE mode is the default mode when MODE[1:0] pins are left unconnected, and the WDOG_EN pin is either a ‘0’ or ‘1’.

4.9.3 MSM6100 ETM Mode

Target speeds for the processor core of MSM6100 are higher than previous MSM cores, an Embedded Trace Macrocell (ETM) provided by ARM is implemented in the MSM6100 device to enable trace capability when the device reach its target system speed.

4.9.3.1 Embedded Trace Macrocell (ETM) Overview

The ETM9 Embedded Trace Macrocell is a block that provides instruction and data trace for the ARM9 family of microprocessors. The use of ETM in a system allows real-time debugging.

The ETM consists of two parts:

- A trace port: The trace port broadcasts trace information (either instruction trace or data trace).
- Triggering facilities: Controlling the ETM to filter and control trace operations.

ETM is designed to be connected directly to the ARM core it is tracing. The trace port connects to a software debugger which will configure the ETM and process the information the ETM provides.

An example debugging environment is shown below.

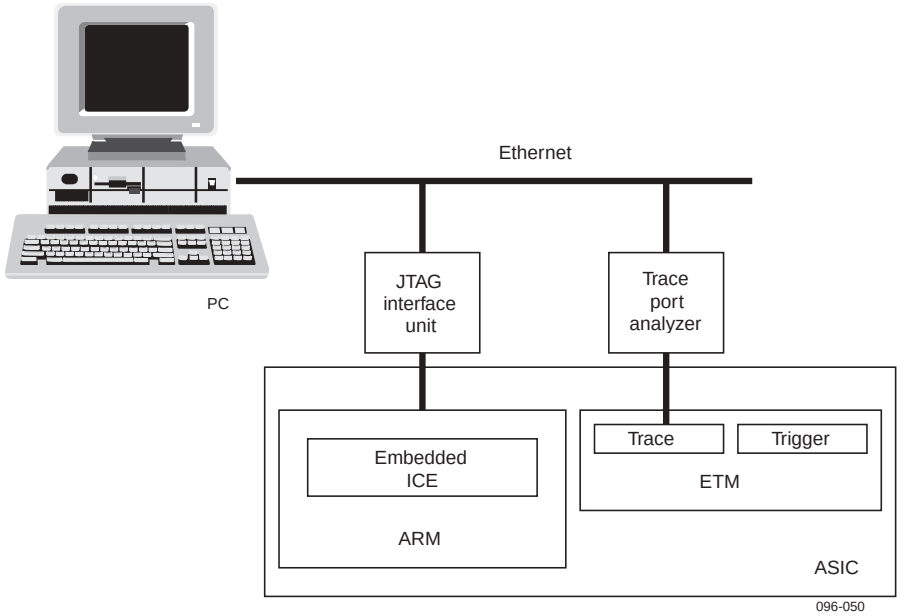


Figure 4-50 Example Debugging Environment

4.9.3.2 ETM Architecture

The architecture of the ETM is shown in Figure 4-51. A trace packet can be 4, 8, or 16 bits, with the bigger packets allowing higher trace data bandwidth.

The trace port comprises:

- PIPESTAT[2:0]: Three pipeline status pins.
- TRACEPKT: An n-pin trace packet port, where n can be 4, 8, or 16 pins.
- TRACESYNC: A trace packet synchronization pin.
- TRACECLK: A clk signal with the same frequency as the processor clock.

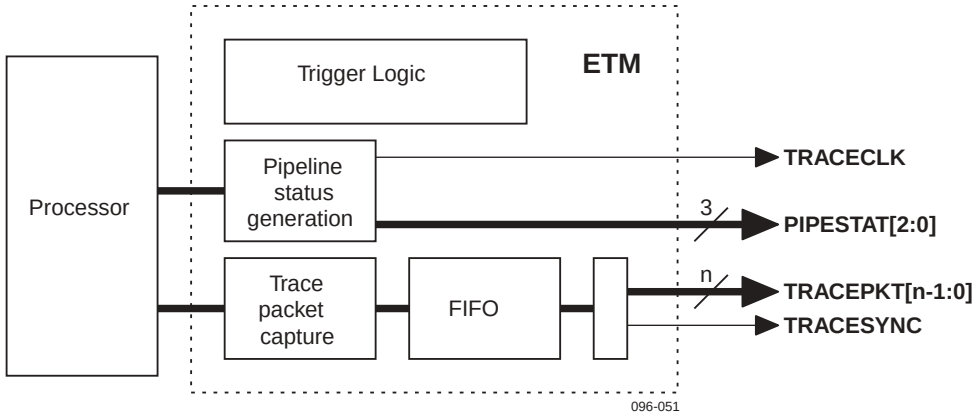


Figure 4-51 ETM Architecture

In addition to the ETM modes supported in previous MSMs which uses ETM7, MSM6100 (which uses ETM9) also implements a wide (demultiplexed) trace port, this is achieved by clocking the trace port at half the processor operating frequency and routing trace port outputs to pairs of output pins.

MSM6100 ETM supports features include:

- 16-bit normal mode when running the ARM at low speed
- 8-bit normal mode when running the ARM at low speed and use fewer pins
- 8-bit demux mode when running the ARM at target speed
- both full-rate and half-rate clocking modes (when half-rate clocking is used, the trace data signals are sampled by the TPA on both rising and falling edges of the trace clock)

Depending on the port size and port mode chosen, the number of pins required by the trace port may be 13, 21 or 25.

4.9.3.3 ETM Implementation

The MSM6100 device implements the maximum trace port size (16 bits) and the maximum FIFO size.

For ETM debugging, the MSM6100 device is set into a dedicated debug mode called ETM mode. This mode is set through the MSM6100 device wdog_en and mode pins. To set ETM mode, WDOG_EN = X(don't care) and MODE[1:0] = 0b01. In ETM mode, the MSM device operates on native mode but with selected GPIO pins dedicated to the ETM trace port. See Table 4-22.

Table 4-22 GPIO Pins Used in ETM Mode

Pins Used	ETM Signal
GPIO66	ETM_TRACECLK
GPIO65	ETM_TRACESYNC

Table 4-22 GPIO Pins Used in ETM Mode (Continued)

Pins Used	ETM Signal
GPIO64	ETM_PIPESTAT2
GPIO63	ETM_PIPESTAT1
GPIO62	ETM_PIPESTAT0
GPIO61	ETM_TRACE_PKT15
GPIO60	ETM_TRACE_PKT14
GPIO59	ETM_TRACE_PKT13
GPIO58	ETM_TRACE_PKT12
GPIO57	ETM_TRACE_PKT11
GPIO56	ETM_TRACE_PKT10
GPIO55	ETM_TRACE_PKT9
GPIO54	ETM_TRACE_PKT8
GPIO53	ETM_TRACE_PKT7
GPIO52	ETM_TRACE_PKT6
GPIO51	ETM_TRACE_PKT5
GPIO50	ETM_TRACE_PKT4
GPIO49	ETM_TRACE_PKT3
GPIO48	ETM_TRACE_PKT2
GPIO47	ETM_TRACE_PKT1
GPIO46	ETM_TRACE_PKT0
GPIO45	ETM_TRACESYNC_B
GPIO44	ETM_PIPESTAT0B
GPIO43	ETM_PIPESTAT1B
GPIO42	ETM_PIPESTAT2B
GPIO41	ETM_GPIO2_GROUP_INT
GPIO40	ETM_GPIO2_CS_N
GPIO39	ETM_KEYSENSE_INT

ETM_GPIO2_GROUP_INT, ETM_GPIO2_CS_N and ETM_GPIO2_CS_N are for emulation on SURF boards in ETM mode.

4.9.4 JTAG Emulation

For a more detailed description of this topic please refer to the ARM Application Note 28: “The ARM926EJ-S Debug Architecture” (ARM DAI 0028A). A few important points from that document are reproduced here for a quick overview.

EmbeddedICE is a JTAG-based debugging environment for ARM microprocessors which provides an interface between the source-level symbolic debugger on a host and an ARM microprocessor embedded in any ASIC. The ARM Debug Architecture uses a protocol converter box to allow the debugger to talk via a JTAG port directly to the microprocessor. In effect the scan chains in the microprocessor that are required for test are re-used for debugging.

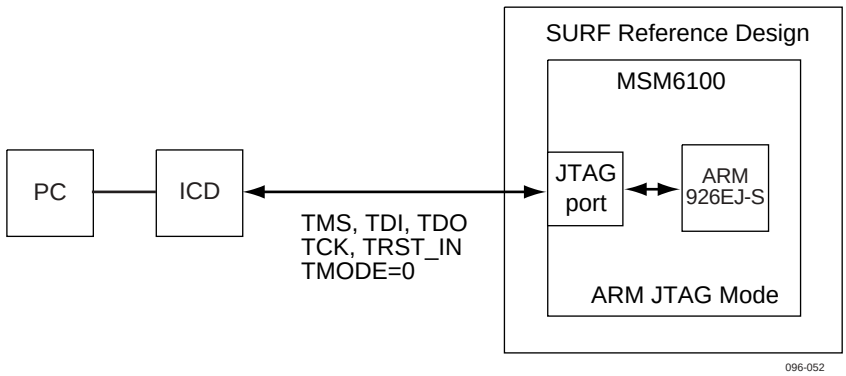


Figure 4-52 ARM JTAG Setup Example

There are effectively two scan chains around the ARM microprocessor: a scan chain around the whole periphery of the microprocessor and a subset of the first scan chain, covering only the databus and the breakpoint.

The shorter scan chain on the databus allows instructions and data to be inserted into the microprocessor without the overhead of clocking the data around the entire periphery of the ARM926EJ-S processor. In addition to the scan chains, the ARM926EJ-S Debug Architecture uses a macrocell called the EmbeddedICE macrocell. The EmbeddedICE macrocell provides on-chip debug support for the ARM926EJ-S. The EmbeddedICE macrocell consists of two real-time watchpoint registers, together with a control and status register. Execution is halted when a match occurs between the values programmed into the EmbeddedICE macrocell and the values currently appearing on the address bus, databus and some control signals.

Native Mode debugging uses the ARM JTAG port and the embedded ICE BREAKER unit to control the operation of the ARM processor. This unit contains two hardware break points, if more break points are required then the FLASH must be replaced (shadowed) with SRAM.

4.10 UART, R-UIM, and USB Interfaces

4.10.1 UART1

The Universal Asynchronous Receiver Transmitter (UART) communicates with serial data that conforms to RS-232 interface protocol. The UART (Figure 4-53) can be used as a serial data port in Mobile Station testing and debugging with a properly written, user-defined download program. If the Mobile Station uses EEPROM or Flash memory, the serial data port can be used to load and/or upgrade the system software. The serial data port can also be used to run Mobile Station diagnostic tests during the manufacturing process.

The serial data port is a UART channel. The UART processes both the transmit and the receive data, and interrupt control circuits, a clock generator, a bit rate generator (BRG), and a microprocessor interface.

The UART has a 512-byte transmit (Tx) FIFO and a 512-byte receive (Rx) FIFO. The UART features hardware handshaking, programmable data sizes (5, 6, 7, or 8 bits), programmable stop bits (0.563, 1.000, 1.563, and 2.000), and odd, even, space, or no parity. The UART operates at a 230.4 kbps maximum bit rate.

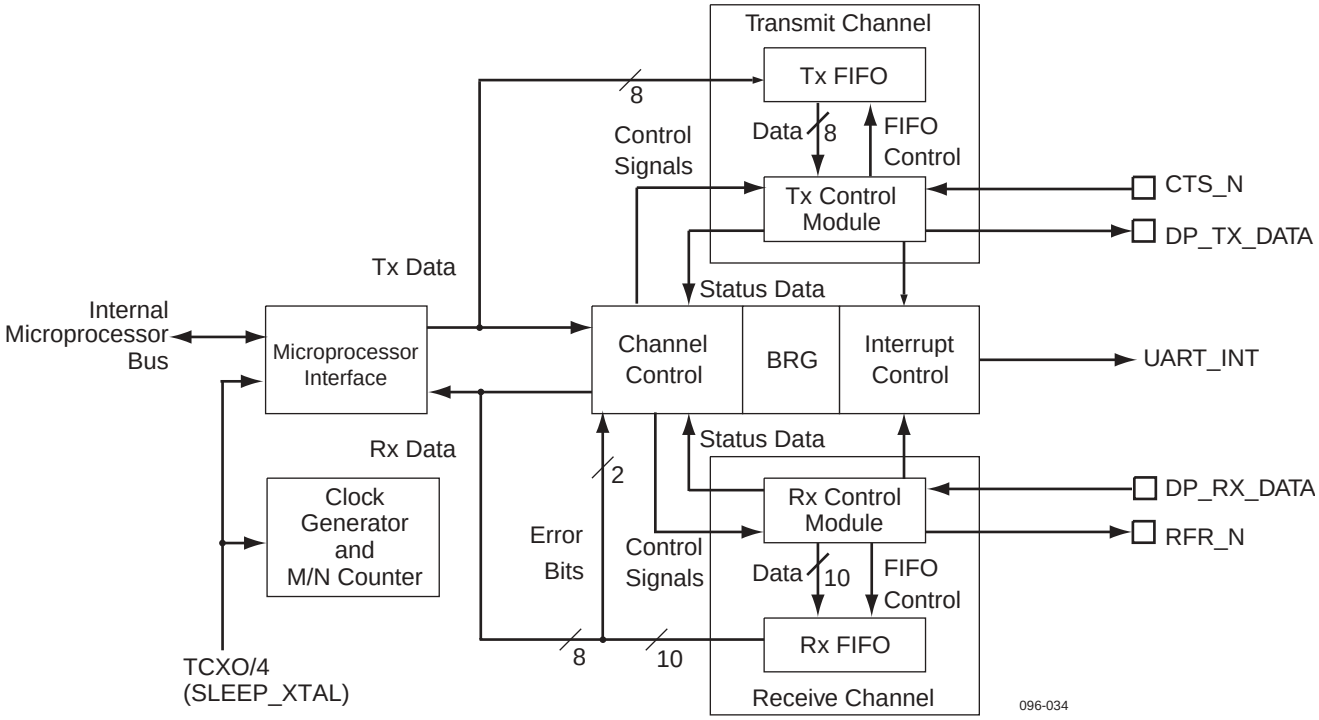


Figure 4-53 UART Block Diagram

4.10.1.1 UART1 Transmit Cycle

The UART Tx channel contains a 512-byte Tx FIFO and a Tx control module. The Tx FIFO accepts parallel data from the microprocessor. The Tx control module reads data from the Tx FIFO and sends it out serially, adding a start bit, optional parity bits, and stop bit(s).

The Tx FIFO accepts characters from the microprocessor. The TXRDY bit in the Status Register (SR) sets (1) whenever the Tx FIFO has space available. The Tx channel asserts an interrupt (TXLEV) when the Tx FIFO has fewer than the number of characters programmed in the Transmit FIFO Watermark Register (TFWR).

Idling or disabling the Tx channel holds the DP_TX_DATA pin in a marking state (high). Enabling the Tx channel, with a character waiting in the Tx FIFO, loads that character into the Tx shift register. Next, a start bit is transmitted (Tx low) for one bit time, followed by each bit in the character, LSBit first, then the optional parity bit followed by stop bits (Tx high). The number of stop bits is programmable.

If after the previous transmission the Tx FIFO is not empty, the Tx control module begins another transmission by loading the next Tx FIFO character into the Tx shift register. If after the previous transmission the Tx FIFO is empty, the Status Register's (SR) TXEMT (Tx empty) bit is set (1). Setting TXEMT indicates an underrun in the Tx channel. The TXEMT bit clears (0) once the transmit shift register has a new character from the Tx FIFO.

Setting (1) Command Register, CR:UART_TX_DIS, disables the Tx channel. When disabled, the Tx channel continues transmitting any character in the Tx shift register until the register is empty. When transmission ends, the DP_TX_DATA pin goes into a marking state (high). Setting (1) CR:UART_TX_EN reenables the Tx channel.

Setting (1) MR1:CTS_CTC enables the Clear-To-Send (CTS_N) control signal. The Tx channel checks the CTS_N input before transmitting a character. If CTS_N is high, Tx channel stops transmitting and continues marking. If CTS_N is low, transmission begins or continues. If CTS_N goes high in the middle of character transmission, the Tx channel waits for a completed transmission before entering the marking state. The Tx channel can generate a programmable interrupt whenever CTS_N changes states.

If the CR issues a 'Start Break' command after the Tx channel transmits all the characters in the Tx FIFO and shift register, the Tx channel forces the DP_TX_DATA pin low. Until the CR issues a 'Stop Break' command, the DP_TX_DATA pin remains low.

When the CR issues a 'Reset Transmitter' command, it causes the Tx channel to cease transmitting. The DP_TX_DATA pin enters a marking state and flushes the Tx FIFO.

4.10.1.2 UART1 Receive Cycle

The receive channel contains a 512-byte Rx FIFO and a receive (Rx) control module. Each byte in the Rx FIFO has two bits of corresponding status information. The DP_RX_DATA pin inputs the serial data. The Rx control module converts the serial data into parallel data and loads it into the Rx FIFO.

If the Rx FIFO is not full, the control module writes the received character to the Rx FIFO. When the Rx control module writes a received character to an empty Rx FIFO, the RXRDY bit in the status register (SR) is set (1). If the Rx FIFO becomes full, the RXFULL bit sets (1). The RXFULL bit clears (0), when a character is read from the Rx FIFO. The RXRDY bit clears (0) when the Rx FIFO becomes empty once again. When the Rx FIFO has more characters than what was programmed in the Receive FIFO Watermark Register (RFWR), the Rx channel asserts a RXLEV interrupt. If a character is waiting in the Rx FIFO for a programmed time period (STALE_TIMEOUT), a RXSTALE interrupt occurs. The Rx channel asserts an RXHUNT interrupt whenever the Rx channel receives a character that matches the value found in the Hunt Character Register (HCR).

Two status bits are associated with each 8-bit data word in the Rx FIFO. One status bit defines the received break condition, the other status bit is the logical-OR of a parity error or framing error. MR2:ERROR_MODE determines the character error mode or the block error mode. In character error mode, the SR's error bits apply only to the byte waiting to be read from the Rx FIFO. In block mode, the SR's error bits are the logical-OR of all incoming error bits, since the Command Register (CR) last issued a 'Reset Error Status' command. Whether in character error mode or block mode, reading the status register has no effect on the Rx FIFO.

If the Rx FIFO is full and the Rx channel receives a new character, that character remains in the Rx shift register until a position becomes available in the Rx FIFO. Any additional incoming characters are lost, until space becomes available in the Rx FIFO. When characters are lost, an overrun error occurs and SR:OVERRUN is set (1). Once SR:OVERRUN is set (1), it remains set until the CR issues a 'Reset Error Status' command. An overrun error does not affect the Rx FIFO's contents.

Setting CRUART_RX_DIS disables the receive channel. Incoming characters are now ignored, but characters already in the Rx FIFO remain unchanged and can be read by the microprocessor. Having the CR issue a 'Reset Receiver' command flushes the Rx FIFO and clears (0) the status bits. The receive channel remains disabled until it is reenabled by setting (1) CR:UART_RX_EN.

Setting MR1:RFR_CTL enables the 'Automatic Ready-For-Receiving' (RFR_N) mode. In this mode, the RFR_N pin goes high when the Rx FIFO level is the same or greater than the value programmed in MR1AUTO_RFR_LEVEL. When the Rx FIFO level falls below the programmed RFR_N level, the RFR_N pin returns to a low state. This feature prevents overruns by connecting the channel's RFR_N pin to a transmitting device's CTS_N pin.

4.10.1.3 UART1 Interrupt Control

The Tx channel asserts DELTA_CTS and TXLEV interrupts while the Rx channel asserts RXLEV, RXSTALE, RXBREAK, and RXHUNT. The Interrupt Status Register (ISR) shows each interrupt's status bit independent of the Interrupt Mask Register's IMR bit state. The Mask Interrupt Status Register (MISR) returns the Bitwise AND of the ISR and IMR registers. The six UART interrupts generated by the ISR and IMR registers are described below. IMR bit 6 indicates the current state of the CTS input.

1. DELTA_CTS indicates that the CTS_N pin has changed state from high to low, or from low to high. The CURRENT_CTS (bit 6 in the ISR register) holds the actual value of the CTS_N pin. Setting (1) DELTA_CTS (bit 5 in the IMR register) enables the interrupt. The CR channel command Reset CTS_N clears (0) the DELTA_CTS interrupt status bit.
2. RXLEV indicates that the Rx FIFO has more characters than the programmed watermark threshold. To enable this interrupt, the RXLEV (IMR, bit 4) must be set (1) and an appropriate watermark threshold must be written to the RFWR. If the amount of characters read from the Rx FIFO is less than or equal to the programmed RFWR value, the RXLEV interrupt clears (0).
3. RXSTALE (IMR register, bit 3) shows that there are one or more characters in the Rx FIFO (but less than the level defined in RFWR) and they have been there longer than the timeout value specified by the IPR register. To enable RXSTALE, RXSTALE (IMR bit 3) must be set (1) and an appropriate timeout value must be written to the IPR register. This timeout value is determined by programming STALE_TIMEOUT, bits [10:7, 4:0]. The STALE_TIME_OUT value specifies how many character times must elapse before a Stale Character timeout is generated. In this instance, a character time is defined as 10 times the bit duration. The RXSTALE interrupt status bit clears each time a character is read from the Rx FIFO. The stale character interrupt duration is measured from either the first or the last arriving character in the Rx FIFO as programmed by the RXSTALE_LAST, bit 5 in IPR. Clearing (0) the RXSTALE_LAST bit measures the Rx Stale Character Timeout duration from the first arriving character in the Rx FIFO. Setting (1) this bit measures the duration from the last arriving character in the Rx FIFO. When RXSTALE_LAST is cleared (0), the value programmed in STALE_TIMEOUT is usually greater than the value programmed in RFWR; if the value is not greater, then STALE_TIMEOUT expires before the RFWR setpoint.

   RXBREAK indicates that the Rx break-detect circuit detected the beginning of a new break condition or the end of an existing break condition. Setting (1) the RXBREAK (IMR register, bit 2) enables the interrupt. The CR channel command "Reset Break" Change clears the RXBREAK interrupt status bit. A detected break is an all-zeros character with an invalid stop bit. The end of a break is a marking condition (1) at least 1/2 bit width. A detected break occupies one position in the Rx FIFO[1].

[1] The DP_RX_DATA pin has an internal pull-down. As such, a "no connect" on this pin is perceived as a break.

4. RXHUNT indicates that the RX FIFO has detected the special hunt character. To enable the RXHUNT interrupt, an appropriate hunt character must be written to the Hunt Character Register (HCR) and the RXHUNT (IMR register, bit 1) must be set (1). The CR channel command Reset Error Status clears the RXHUNT interrupt status bit.
5. TXLEV shows that the Tx FIFO has the same or fewer number of characters than the number programmed in the Tx watermark threshold. To enable this interrupt, TXLEV (IMR register, bit 0) must be set (1) and an appropriate number must be written to the Transmit FIFO Watermark Register (TFWR). The TXLEV interrupt status bit clears (0) after enough characters have been written into the Tx FIFO to bring the amount of characters equal to or more than the TFWR value.

4.10.1.4 Clock Generator

The clock generator generates the UART clock. The clock comes from a dedicated M/N counter running off TCXO or TCXO/4. The selection is made via MSM_CLK_SRC_SEL: UART_CLK_SRC_SEL. This counter, defined by the MREG_MSB, NREG_MSB, DREG_MSB, and MND_LSB registers, must be initialized before using the UART. Using the recommended M/N value, the clock runs at 1.8432 MHz based on TCXO/4, or 7.3728 MHz based on TCXO. Writing the value 0 (zero) to the M registers cause the M/N counter to output a 0 Hz clock, effectively turning off the clock. This puts the UART into its power save mode. When the clock is disabled, none of the UART registers are accessible except for the M/N counter registers.

4.10.1.5 Bit Rate Generator

The Bit Rate Generator (BRG) generates enables to the Tx and Rx channels that are 16X the nominal bit rate. The BRG selects one of 16 possible bit rates defined in Clock Select Register and sends the selected bit rate to the Tx channel (CSR bits [3:0]) and Rx channel (CSR bits [7:4]). Table 4-23 assumes a system clock frequency of 1.8432 MHz. Table 4-24 assumes a system clock frequency of 7.372 MHz. The bit rates are generated by dividing the UART system clock.

Table 4-23 Clock Select Register Bit Rates (Based on 1.8432 MHz Input)

CSR[7:4] CSR[3:0]	Bit Rate (bit/sec)	CSR[7:4] CSR[3:0]	Bit Rate (bit/sec)
0000	75	1000	7.2 k
0001	150	1001	9.6 k
0010	300	1010	14.4 k
0011	600	1011	19.2 k
0100	1.2 k	1100	28.8 k
0101	2.4 k	1101	38.4 k
0110	3.6 k	1110	57.6 k
0111	4.8 k	1111	115.2 k

Table 4-24 Clock Select Register Bit Rates (Based on 7.3728 MHz Input)

CSR[7:4] CSR[3:0]	Bit Rate (bit/sec)	CSR[7:4] CSR[3:0]	Bit Rate (bit/sec)
0000	300	1000	28.8 k
0001	600	1001	38.4 k
0010	1.2 k	1010	57.6 k
0011	2.4 k	1011	76.8 k
0100	4.8 k	1100	115.2 k
0101	9.6 k	1101	153.6 k
0110	14.4 k	1110	230.4 k
0111	19.2 k	1111	—

4.10.1.6 Microprocessor Interface

All communication between the microprocessor and the UART goes through the microprocessor interface. The microprocessor interface synchronizes the data and command signals to the UART. This interface runs off the general-purpose TCXO (clock) as shown in Figure 4-53, to minimize wait states. After data aligns with the UART clock, the microprocessor interface presents the data on an internal bus to the other related circuits.

4.10.2 UART2 and UART3

The MSM6100 device contains a second and third UART which share multiplexed pins as shown in Figure 4-54.

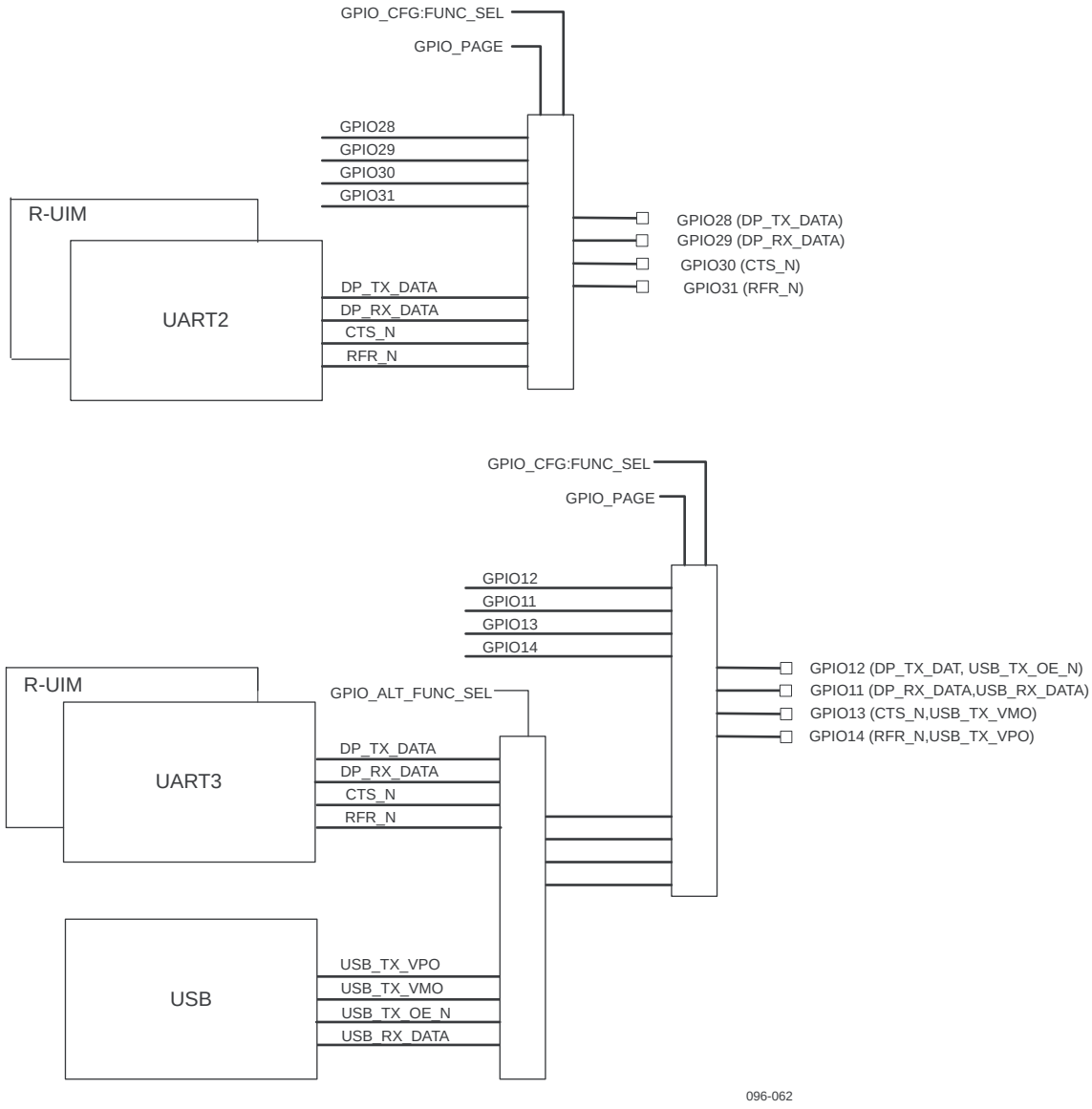


Figure 4-54 Multiplexing Arrangement for UART2 and UART3

4.10.3 R-UIM

The Removable User Identity Module (R-UIM) is a smart card for CDMA cellular applications. R-UIM provides personal authentication information which allows the mobile station or handset to be connected with the network. The R-UIM card enables handset independence for the user. The R-UIM card can be inserted into any CDMA R-UIM equipped handset, allowing the user to receive or make calls and receive other subscribed services from any R-UIM equipped handset.

When CDMA was initially introduced in North America, it had to coexist with the already mature AMPS networks, and to ensure its compatibility, CDMA networks adopted concepts such as MIN (Mobile Identification Number) and ESN (Electronic Serial Number) as unique IDs within each mobile station. Unlike other digital cellular standards (specifically, GSM), CDMA handsets did not implement a SIM card approach to providing subscribers with a portable module of identity independent of the used handset.

Standardization efforts for providing R-UIM capability to CDMA subscribers are aimed at providing personal mobility and roaming capability. It is anticipated that the use of the R-UIM will make it much easier for CDMA subscribers and mobile operators to access and provide value-added services such as wireless Internet and information services plus future e-commerce opportunities.

4.10.3.1 Implementation

The MSM6100 device supplies UIM_CLK, UIM_IO, and UIM_RESET to R-UIM reader. UIM_RESET can be implemented by using a GPIO bit under software control. UIM_CLK and UIM_DATA are hidden behind the UART2 and UART3 interfaces. A brief summary of the functional mode mux at pins is shown in Table 4-25 and Table 4-26.

Table 4-25 GPIO Pins (UART2)

Primary Pin	UIM	UART2
GPIO14	UIM_CLK	RFR_N
GPIO13	—	CTS_N
GPIO29	—	DP_RX_DATA
GPIO28	UIM_DATA	DP_TX_DATA

Table 4-26 GPIO Pins (UART3)

Primary Pin	UIM	UART3	USB
GPIO25	UIM_CLK	RFR_N	TX_VPO
GPIO12	—	CTS_N	TX_VMO
GPIO11	—	DP_RX_DATA	RX_DATA
GPIO24	UIM_DATA	DP_TX_DATA	TX_OE_N

4.10.4 USB Interface

The MSM6100 device contains a Universal Serial Bus (USB) interface in order to provide an efficient interconnect between the mobile phone and a personal computer (PC). The USB interface of the MSM6100 is compliant with the USB 2.0 Specification as a full-speed (12 Mbps) peripheral. The USB 2.0 Specification also allows a peripheral to operate at low-speed (1.5 Mbps) and hi-speed (480 Mbps). The MSM6100 does not support either the low-speed or the hi-speed options.

The MSM6100 USB interface can be used to transfer the following types of data between a phone and a host:

- USB status and control, for enumeration, interrupts, etc.
- circuit-switched or packet switched data from an over-the-air connection
- audio data from an over-the-air connection
- file data to and from memory, or a storage media interface
- diagnostic data when the phone is in diagnostic mode

The MSM6100 has ten different USB endpoints to support these various types of control and data transfers. An external USB transceiver and 48 MHz resonator or crystal oscillator are required to implement the USB interface.

This section describes the functionality and configuration of the USB interface controller within the MSM6100 device. Section 4.13.4 describes the functionality and configuration of the USB clock circuit.

Consult the USB Specification Revision 2.0 (see www.usb.org/developers) for more information on USB usage and requirements.

4.10.4.1 Connecting to Transceivers

As shown in Table 4-27, the MSM6100 has six signal pins assigned for the USB interface. These pins are multiplexed with GPIO[9:14], and allow the implementation of either a seven wire, five wire or three wire interface to a USB transceiver. The seven wire interface requires a GPIO in addition to GPIO[9:14], in order to implement the SUSPEND signal.

The three wire interface requires that the transceiver also connect to the I2C bus of the MSM6100. The I2C pins on the MSM6100 device are not dedicated to only the transceiver, but can also connect with other I2C devices on the phone. Since the I2C pins are not dedicated to the transceiver, the interface is sometimes referred to as a three wire interface, even though the total number of pins required is still five.

The connections between the MSM6100 device and the three types of transceivers are also shown in Table 4-27.

Table 4-27 Transceiver Connections

MSM6100 Pin		Seven Wire Interface	Five Wire Interface	Three Wire Interface
Primary function	USB function			
GPIO23	USB_TX_VPO	VPO	VP	DAT
GPIO22	USB_TX_VMO	VMO	VM	SE0
GPIO21	USB_TX_OE_N	OE_N	OE_N	OE_TP_N
GPIO20	USB_RX_DATA	RCV	RCV	
GPIO19	USB_RX_VMI	VMI		
GPIO18	USB_RX_VPI	VPI		
other GPIO	SUSPEND	SUSPEND	SUSPEND	
GPIO26	I2C_SDA			I2C_SDA
GPIO27	I2C_SCL			I2C_SCL

The MSM6100 pins are configured for either GPIO or USB operation with the following registers:

- GPIO_FUNC_SEL_0
- WEB_MISC_WR

The five wire interface allows GPIO[9:10] to be used for other purposes, while the three wire interface allows GPIO[9:11] to be used for other purposes. The mode of these pins is programmed as follows:

seven wire interface

- USB_DEVICE_CONTROL : XCVR_TYPE = 0
- USB_DEVICE_CONTROL : USB_OTG_XCVR = 0

five wire interface

- USB_DEVICE_CONTROL : XCVR_TYPE = 1
- USB_DEVICE_CONTROL : USB_OTG_XCVR = 0

three wire interface

- USB_DEVICE_CONTROL : XCVR_TYPE = X
- USB_DEVICE_CONTROL : USB_OTG_XCVR = 1

4.10.4.2 Seven Wire Transceiver

Figure 4-55 shows the connections between the MSM6100 device and a seven wire transceiver.

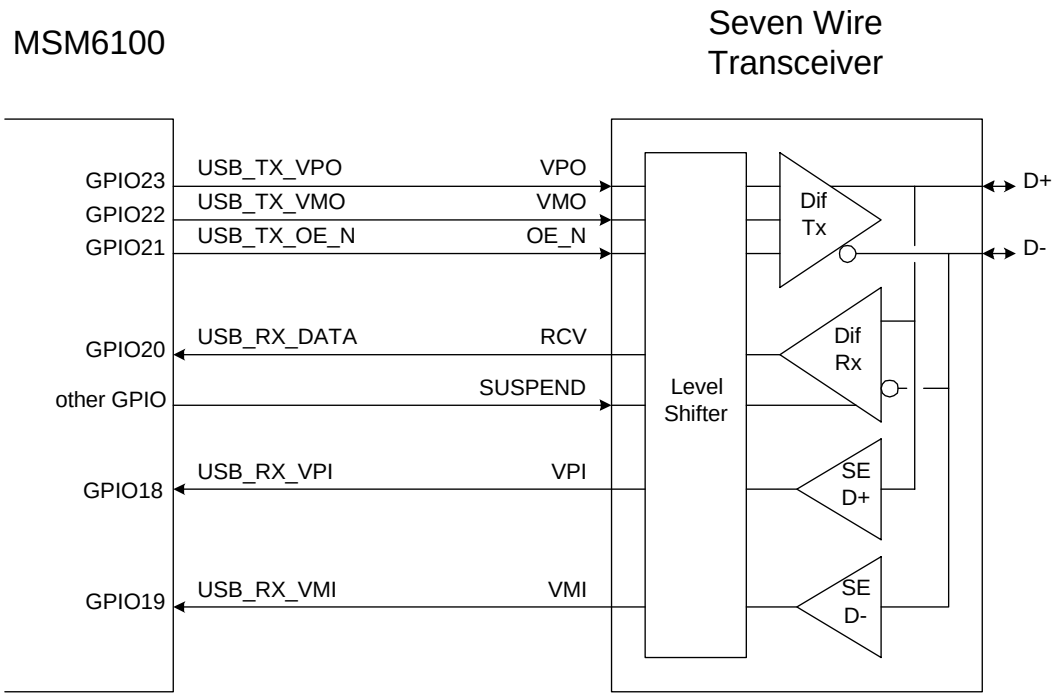


Figure 4-55 Seven Wire Transceiver

4.10.4.3 Five Wire Transceiver

Figure 4-56 shows the connections between the MSM6100 device and a five wire transceiver.

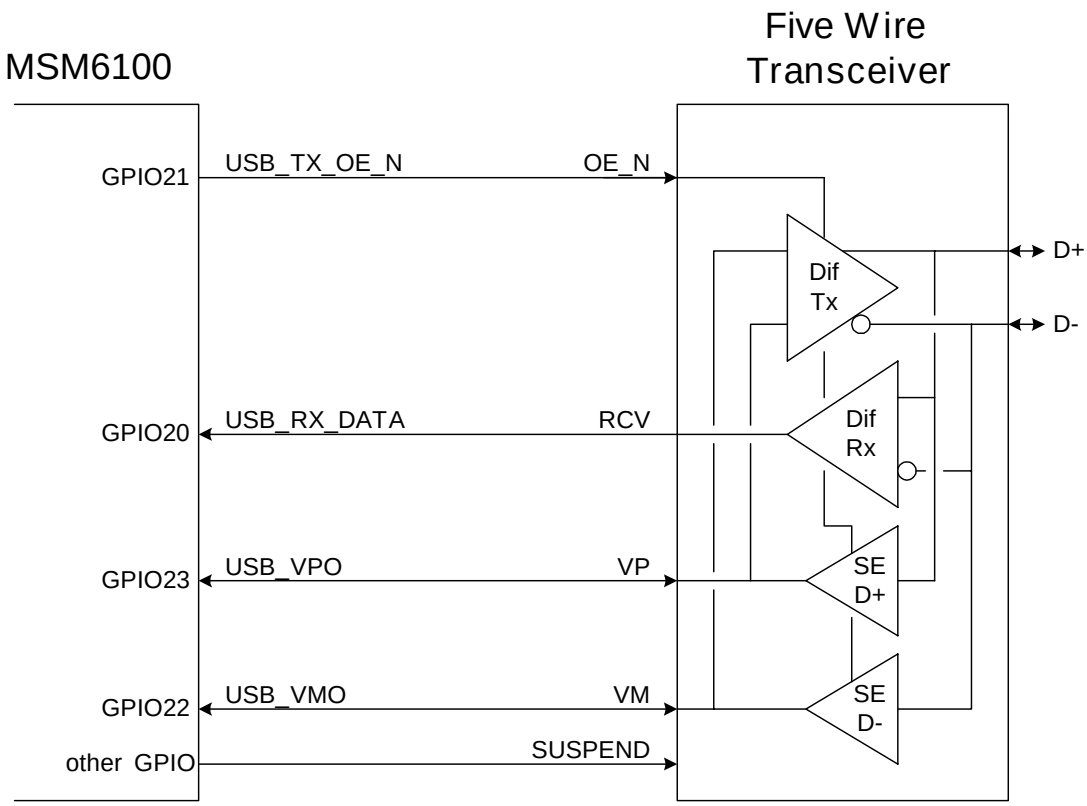


Figure 4-56 Five Wire Transceiver

4.10.4.4 Three Wire Transceiver

Figure 4-57 shows both the basic and optional connections between the MSM6100 device and a three wire transceiver.

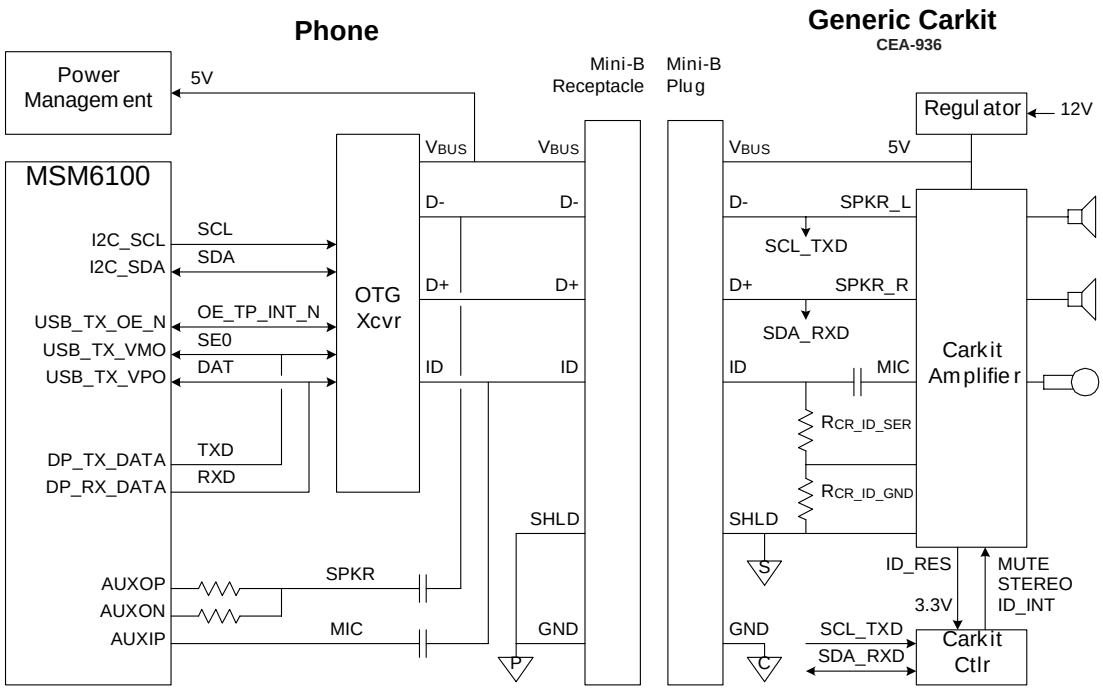


Figure 4-57 Three Wire Transceiver

The three wire transceiver is more commonly known as an On-The-Go (OTG) Transceiver. OTG Transceivers are available from several manufacturers, and implement the features called out in the following two specifications:

- On-The-Go Supplement to the USB 2.0 Specification, www.usb.org

CEA-936 Mini USB Analog Carkit Interface, www.ce.org

The basic connection between the MSM6100 and the OTG transceiver consists of I2C_SCL, I2C_DAT, DAT, SE0 and OE_TP_INT/. These connections allow an MSM6100 based phone to connect to a PC through a mini-B receptacle.

The TXD and RXD connections allow an MSM6100 based phone to connect to an RS232 accessory through a mini-B receptacle. CEA-936 allows a phone to supply power to such an accessory through the VBUS pin of the mini-B receptacle.

The SPKR and MIC connections allow an MSM6100-based phone to connect to a generic analog carkit through the mini-B receptacle, as specified in CEA-936.

4.10.4.5 Endpoint Configuration

The MSM6100 USB interface supports ten endpoints. Each endpoint has its own FIFO, except for endpoint 0, which has two FIFO's. Table 4-28 shows the configuration of the endpoints.

Table 4-28 USB Endpoint Configurations

Endpoint Number	FIFO Number	Endpoint Type	Direction	Used For	Max Packet Size (bytes)	Number of Banks	FIFO Size (bytes)
0	0	Control	IN	Control	16	1	16
0	8	Control	OUT	Status	16	1	16
1	1	Interrupt	IN	Notification	16	1	16
2	2	Bulk	IN	MMC	64	4	256
3	3	Bulk	IN	Not used	64	2	128
4	4	Isochronous	IN	Voice PCM	16	2	32
5	5	Bulk	OUT	MMC	64	4	256
6	6	Bulk	OUT	Not used	64	2	128
7	7	Isochronous	OUT	Voice PCM	16	2	32
10	10	Bulk	IN	Burst Data, Diagnostics	64	1	1024
11	11	Bulk	OUT	Burst Data, Diagnostics	64	1	1024

As shown in Table 4-28, some endpoint FIFO's have multiple banks. For endpoints 0 to 7, each bank is able to store one packet, of size MaxPacketSize. This allows multiple packets to be queued up at once by either the USB controller for OUT transfers or by the MSM processor for IN transfers.

Endpoints 10 and 11 only have one bank associated with them. The USB controller and MSM processor are able to read or write multiple packets to these FIFO's at one time. In the software register maps, FIFO's 10 and 11 are sometimes referred to as HDR (or high data rate) FIFO's. In order to avoid confusing this name with the HDR air interface protocol that was developed by QUALCOMM, these FIFO's are now referred to as Burst FIFO's in much of the literature.

In accordance with USB terminology, the IN endpoints are used to transfer data to the USB host, and the OUT endpoints are used to receive data from the USB host. Endpoints 3 and 6 were used for data and diagnostics in earlier MSM devices that did not have Burst FIFOs. These endpoints are still present in current MSM devices for legacy reasons.

Endpoints 4 and 7 are used for the isochronous voice PCM interface with the vocoder.

4.11 User Interface

The User Interface of the MSM6100 device consists of:

- Keypad[4:0]
- Ringer
- Auxiliary CODEC Interface
- I2C
- I2S

4.11.1 Keypad Interface

The KEYSENSE[4:0] pins can be used to connect a matrix keypad to the MSM6100 device. The KEYSENSE[4:0] inputs assert a KEYSENSE_INT if any of the pins are pulled low. Figure 4-58 shows the circuits associated with KEYSENSE[4:0].

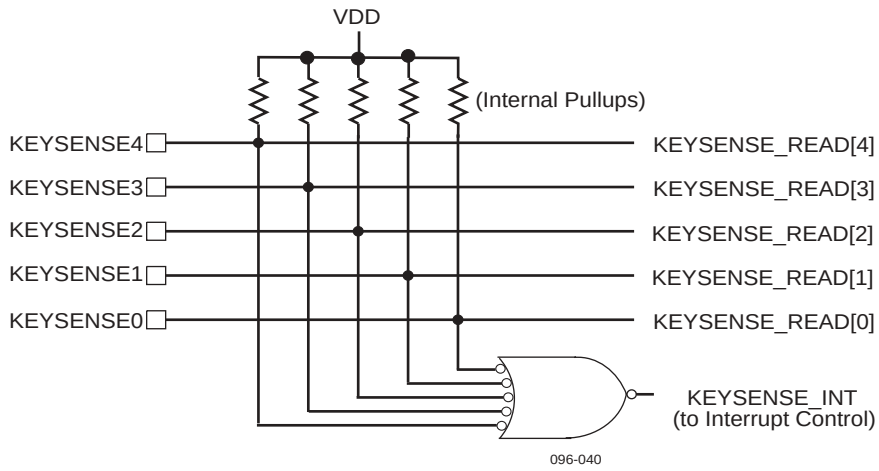


Figure 4-58 KEYSENSE[4:0] Circuits

4.11.2 Ringer

The Ringer generation circuit is programmed to output single tones or DTMF tone pairs on the RINGER pin. The Ringer generation circuit produces and sums two different user-programmable frequencies. Single tones can be generated by programming two tones of the same frequency. The Ringer function is disabled when RESOUT is asserted.

Figure 4-80 is the Ringer generation circuit block diagram. Each M/N counter is clocked by a 300 kHz clock derived from a 19.2 MHz TCXO. The Ringer generation circuit also controls the envelope for the ring output waveform. The signal on the RINGER pin is a digital pulse stream. The RINGER output generally requires an external booster circuit, such as the one shown in Figure 4-80, to drive a sound transducer. The equivalent circuit is integrated into the PM6XXX.

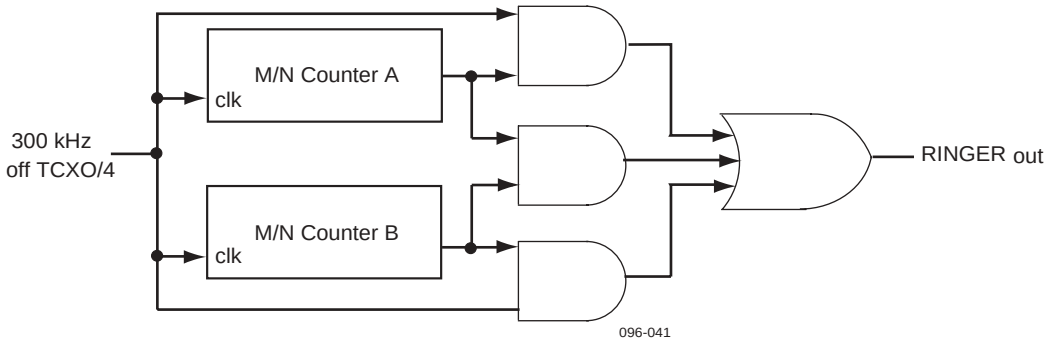


Figure 4-59 Ringer Generation Circuit

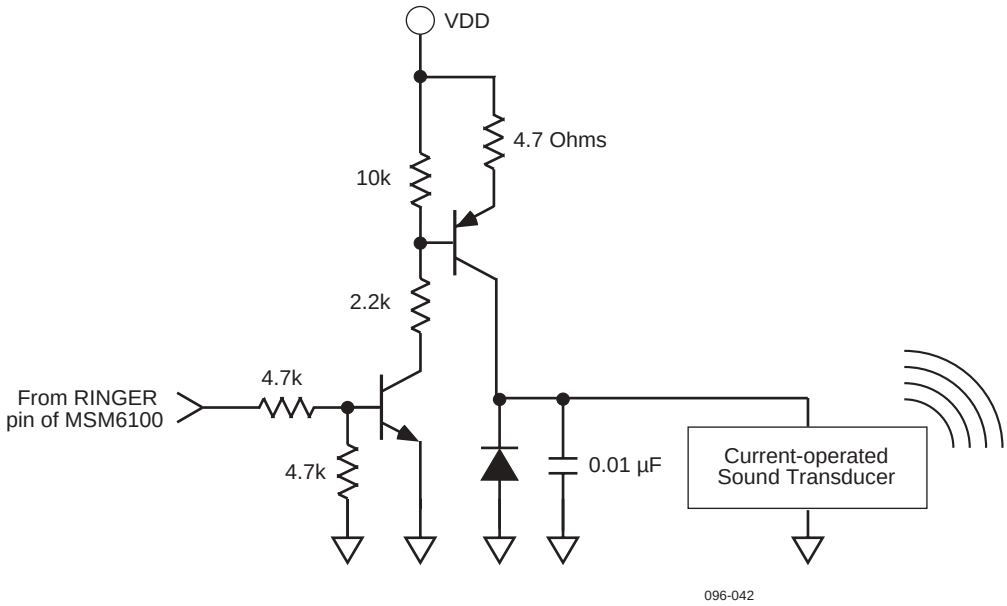


Figure 4-60 External Driver Circuit Example

To generate DTMF tones on the RINGER output, M, N, and D values must be written into the two identical M/N counters. M is a 6-bit value, N is a 13-bit value, and D is a 12-bit value. The 6-bit M value (RINGER_MN_A_MDIV:RINGER_MN_A_MDIV) is written along with the enable for the counter (RINGER_MN_A_MDIV:RINGER_MN_A_EN) to the A counter M value register. RINGER_MN_A_NDIV is written to the A counter N-value register as the one's-complement of the N minus M value. The D value in RINGER_MN_A_DUTY is written to the A counter D value register. These register allocations are identical for the B counter. Table 4-29 lists the decimal values for M and N which are used to program the M/N counters for DTMF tones. The frequencies produced are accurate to better than 1.5% of their specified values for a TCXO of 19.2 MHz.

Table 4-29 Standard DTMF Frequencies and Ringer Programming Values

DTMF Frequency (Hz)	M/N (dec)	M (hex)	N (hex)	1's complement of (N minus M) (hex)	D Duty-Cycle (50% of N) (hex)
350	7 / 6000	07	1770	0896	BB8
440	7 / 4772	07	12A4	0D62	952
480	8 / 5000	08	1388	0C7F	9C4
620	16 / 7742	10	1E3E	01D1	F1F
697	17 / 7317	11	1C95	037B	E4A
770	20 / 7792	14	1E70	01A3	F38
852	12 / 4225	0C	1081	0F8A	840
941	9 / 2869	09	0B35	14D3	5A9
1209	32 / 7940	20	1F04	011B	F82
1336	6 / 1347	6	0543	0AC2	2A1
1477	26 / 5281	1A	14A1	0B78	A50
1633	23 / 4225	17	1081	0F95	840

The DTMF frequency is found by multiplying the ratio M/N (decimal equivalent values) by the clock frequency of 300 kHz. The one's-complement of (N minus M) value is the actual value for programming the N register to get the desired N value. The D value acts as a loudness control. Maximum RINGER loudness is set by using a D value which is 50% of the value in the N column. Softer ringing is achieved by decreasing or increasing the D value.

Each counter (A and B) is enabled by setting (1) bit 6 of the M register. For counter A, this is RINGER_MN_A_EN in the RINGER_MN_A_MDIV register, for counter B it is RINGER_MN_B_EN in the RINGER_MN_B_MDIV register. Setting bit 6 of the M register, resets both counters (CNTR_A and CNTR_B) to start counting at the same time. Both counters can be programmed to generate the same or different frequencies. Clearing (0) bit 6 disables both counters. These M register bits have an edge detect for a zero-to-one transition. To generate a single tone, the counters are phase-locked by clearing bit 6 of both M registers every time the counters are programmed. Bit 6 must be cleared (0) before the N and D registers are programmed and remain cleared for at least 1 ms. The last step of the programming sequence is to set each bit 6 along with its corresponding M value. This triggers the synchronization mechanisms. The standard DTMF keypad tone combinations are listed in Table 4-30.

Table 4-30 Keypad Frequencies

Keypad	Counter A Frequency (Hz)	Counter B Frequency (Hz)
1	697	1209
2	697	1336
3	697	1477
4	770	1209
5	770	1336
6	770	1477
7	852	1209
8	852	1336
9	852	1477
*	941	1209
0	941	1336
#	941	1477
A	697	1633
B	770	1633
C	852	1633
D	941	1633

4.11.3 M/N Counter

GP_MN is a general-purpose M/N Counter. The nominal output frequency is (M/N) x TCXO/4. Figure 4-61 shows the block diagram of the GP_MN M/N counter.

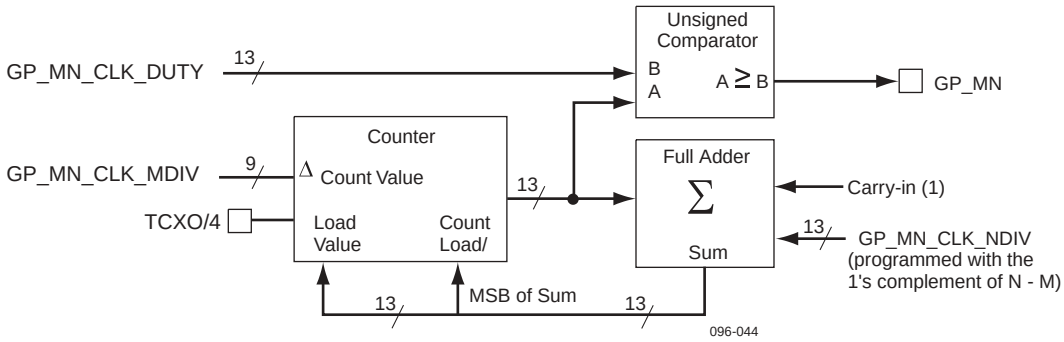


Figure 4-61 GP_MN Block Diagram

To create an output frequency, *f*, with a given duty cycle, *D*, the following sequence must be used to initialize the GP_MN M/N counter:

1. Solve for decimal values for M and N. The value of M cannot exceed the value of N.
2. Convert the decimal values for M and N into a 9-bit value for M and an 13-bit value for N.
3. Write the 9-bit M value into GP_MN_CLK_MDIV.
4. Write the 13-bit result of the one's complement value of N minus M into GP_MN_CLK_NDIV. In other words, GP_MN_CLK_NDIV = 0x1FFF – (N – M).
5. Write the desired 13-bit duty cycle value, *D*, into GP_MN_CLK_DUTY. A 50% duty cycle output waveform is generated when: GP_MN_CLK_DUTY = N/2.

GP_MN can also be used as a programmable digital output. Table 4-31 shows a set of values and the resulting GP_MN output state.

Table 4-31 Using GP_MN as a Digital Output

GP_MN Output State	GP_MN_CLK_MDIV	GP_MN_CLK_NDIV	GP_MN_CLK_DUTY
LOW	0x00	0x1000	0x1FFF
HIGH	0x00	0x1000	0

4.11.4 I2C

The I2C Controller (I2CC) provides an interface between the MSM6100 device and the industry standard I2C serial bus. The controller is an I2C compliant, master-only device. The I2C controller can be used to interface to such devices as Stereo CODEC (control registers only), external LCD controllers, and external touch screen panels.

4.11.4.1 Implementation

The serial data is connected to the GPIO3 and the serial clock is connected to GPIO2 of the MSM6100 device.

Table 4-32 I2C Pins

Primary Pin	I2C Signals
GPIO3	I2C_SDA
GPIO2	I2C_SCL

4.11.5 I2S

The stereo music data receiver consists of an I2S slave which receives the stereo data and sends the data to the QDSP4000. The I2S interface consists of three signals: serial clock (SCK), word select (WS), and serial data (SD). It won't be necessary to require the transmitter and receiver to have the same word length because the extra LSBs will all be ignored.

4.11.5.1 Implementation

Table 4-33 I2S Pins

Primary Pin	Alternate Pins	I2S Signals
GPIO27	AUX_PCM_CLK	I2S_SCK
GPIO26	AUX_PCM_SYNC	I2S_WS
GPIO25	AUX_PCM_DIN	I2S_MASTER_CLK
GPIO24	AUX_PCM_DOUT	I2S_SD

4.11.6 MultiMediaCard Controller

The MSM6100 device contains a MultiMediaCard (MMC) controller that provides a link between the ARM busmaster and the MultiMediaCard bus. The MSM6100 device's MMC interface is designed to be compliant with the MultiMediaCard Association Specifications, version 1.4. The MSM6100 device's MMC Interface can support MMC (2.5 Mbps data transfer) and SD (10 Mbps data transfer) interface specifications. The MSM6100 device's MMC interface provides three communication lines: Command (MMC_CMD), Data (MMC_DATA), and Clock (MMC_CLK) interfaces. Pull-up resistors are required for CMD and DAT lines to implement the MMC interface.

This section is intended to describe only the functionality and configuration of the MSM6100 device's interface controller within the MSM6100 device. Consult the MultiMediaCard Specifications, version 1.4 (available from www.mmca.org) for more information on MMC usage and requirements.

4.11.6.1 Features

The MSM6100 device supports the following MMC features:

- Byte Addressable Memory
- Single/Multiple Block Transfer
- Stream Data Transfer

NOTE Software currently does not support Stream Mode. It is recommended to use multiple block transfers instead of stream mode. Stream mode transfers do not include CRC checking and must use a slower clock to avoid buffer overruns in the MMC card.

4.11.6.2 Bus Topology

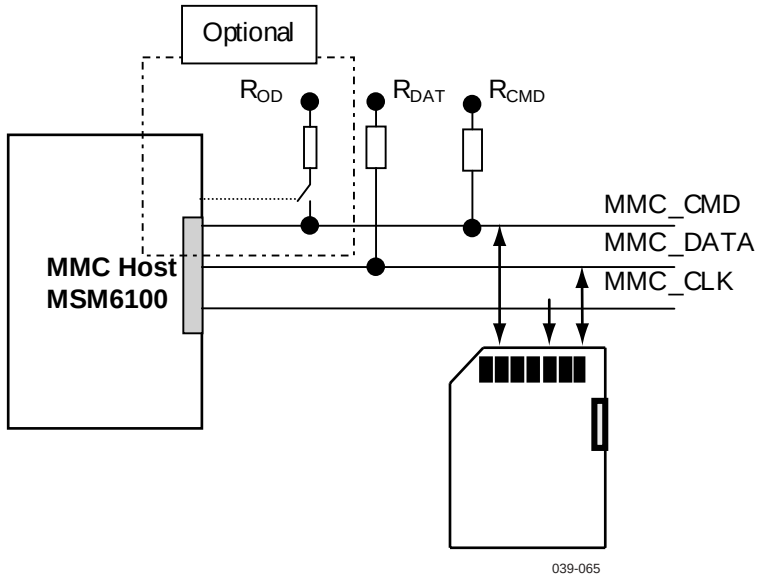


Figure 4-62 MMC Bus Circuitry Diagram

The MSM6100 device's MMC operates in two modes:

- Data Transfer
- Initialization

Resistors are needed on the CMD and DAT bus lines. Consult the MultiMediaCard Product Manual for specifications.

MMC Bus Configuration Data Transfer

R_{DAT} and R_{CMD} are pull-up resistors protecting the CMD and the DAT lines against bus floating when no card is inserted or when all card drivers are in hi-impedance mode.

MMC Bus Configuration Initialization

The R_{OD} is switched on and off by the host synchronously to the open-drain (Initialization) and push-pull (Data Transfer) mode transitions. This function is not implemented in the MSM6100 device.

A constant current source can replace the R_{OD} by achieving better performance (constant slopes for the signal rising and falling edges). If the host does not allow the switchable R_{OD} implementation, a fix RMCD can be used. Consequently the maximum operating frequency in the open-drain mode has to be reduced in this case.

R_{OD} can be replaced by R_{CMD} only. This configuration requires careful consideration for choice of R_{CMD}. If R_{CMD} is too high, then Initialization of the MMC card may fail due to lack of current draw.

4.11.6.3 Initialization Procedures

The MSM6100 device's MMC controller initialization procedure requires the following steps:

- Reset of the MMC controller
- MMC controller configuration
- Idle Mode (Card Identification)

4.11.6.4 MMC Reset

After Reset, software must start the MultiMediaCard clock for at least (7 MASTER_CLK + 4 MMCCLK) cycles to clear all registers. MASTER_CLK is fixed by hardware. MMCC_CLK should be set to less than 400 kHz (i.e., MMCC_CLKRATE = 0x06) during initialization.

Reset can be accomplished by the following procedure:

1. Set MMCC_CLKRATE = 0x0, now MASTER_CLK = MMCCLK.
2. Start the clock for at least 11 MASTER_CLK cycles (7 + 4)
3. Set MMCC_CLKRATE = 0x6 to continue initialization.

4.12 HKADC

The MSM6100 device has an on-chip 8-bit analog-to-digital converter (HKADC) which is intended to digitize DC signals corresponding to analog parameters such as battery voltage, temperature, and RF power levels. The general functional diagram of the MSM6100 device's HKADC is shown in Figure 4-63.

The MSM6100 device has seven analog input pins (HKADC[6:0]) which are multiplexed to the input of the internal HKADC. An eighth analog input is connected to an internally generated 1.826 Volt source. The analog multiplexer is switched by three control bits, MUX_SEL_N found in the HKADC_CNTL2 SBI register. Conversion must be done for the HKADC to complete initialization. This requires that a dummy conversion be done prior to using HKADC.

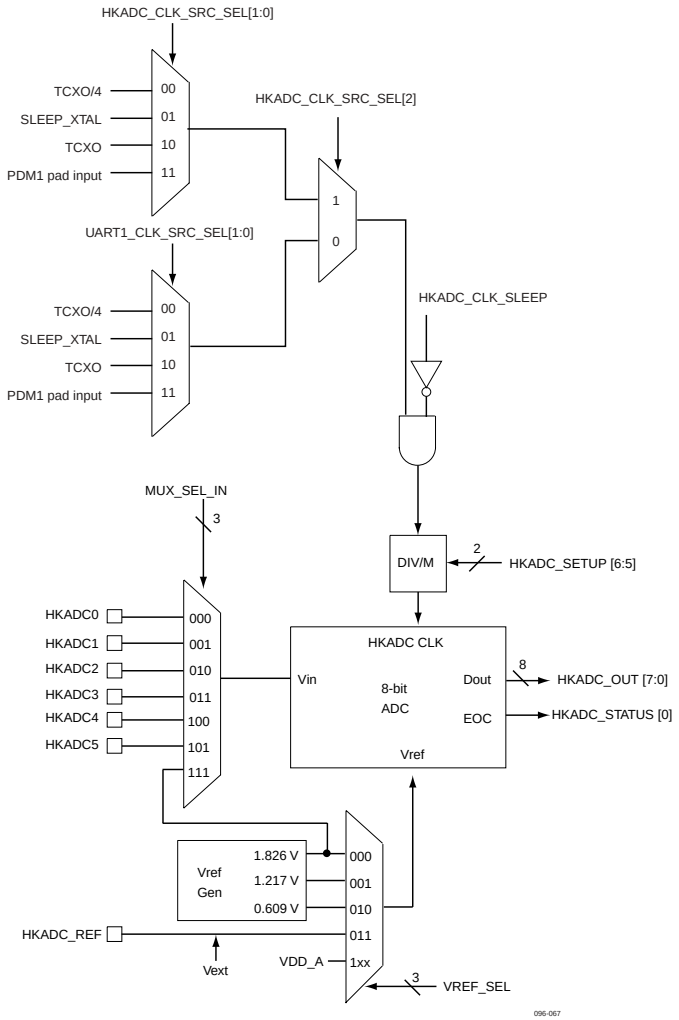


Figure 4-63 HKADC Configuration

4.12.1 Analog Input Voltage Range

The input voltage range of the ADC on the MSM6100 device is programmable to three fixed ranges as well as two other ranges determined by off-chip voltage sources. The analog input range is set by selecting the voltage reference of the ADC (Vref) via the HKADC_VRT_SEL[2:0] control bits found in the HKADC_CONFIG (analog SBI) register.

In the case of HKADC_VRT_SEL = 011, the reference voltage to the HKADC is input on the HKADC6 pin. The maximum voltage that can be applied to the HKADC6 pin is V_{DD_A}. The minimum voltage that can be applied is 0.609 Volts. In the case of HKADC_VRT_SEL = 1xx, the reference voltage to the HKADC is V_{DD_A}.

The range of voltage for V_{DD_A} is from 2.5 to 2.7 Volts.

Table 4-34 shows the transfer function of the ADC under five different Vref conditions.

Table 4-34 MSM6100 ADC Transfer Function

Input Voltage Range and Vref set by VREF_SEL					8-bit Digital Output Result HKADC_OUT[7:0]
000	001	010	011	1xx	
Vref=1.826 V	Vref=1.217 V	Vref=0.609 V	Vref= Vext	Vref=VDD_A	
>1.826	>1.217	>0.609	>Vext	>VDD_A	0xFF
1.826	1.217	0.609	(255/255)Vext	(255/255)VDD_A	0xFF
1.819	1.212	0.607	(254/255) Vext	(254/255) VDD_A	0xFE
1.382	0.921	0.461	(193/255) Vext	(193/255) VDD_A	0xC1
1.375	0.916	0.459	(192/255) Vext	(192/255) VDD_A	0xC0
1.368	0.912	0.456	(191/255) Vext	(191/255) VDD_A	0xBF
0.924	0.616	0.308	(129/255) Vext	(129/255) VDD_A	0x81
0.917	0.611	0.306	(128/255) Vext	(128/255) VDD_A	0x80
0.909	0.606	0.303	(127/255) Vext	(127/255) VDD_A	0x7F
0.465	0.310	0.155	(65/255) Vext	(65/255) VDD_A	0x41
0.458	0.305	0.153	(64/255) Vext	(64/255) VDD_A	0x40
0.451	0.301	0.150	(63/255) Vext	(63/255) VDD_A	0x3F
0.007	0.005	0.002	(1/255) Vext	(1/255) VDD_A	0x01
0.000	0.000	0.000	0.000	0.000	0x00

4.12.2 HKADC Operation

Four clock sources are shown in Figure 4-63 for the HKADC. A clock source based on the frequency of TCXO-divided-by-four is the default clock source for the HKADC. The alternate clock sources can be selected by MSM_CLK_SRC_SEL, bits [10:8] of the MSM_CLK_SRCSEL3 register. The sleep oscillator frequency is 32.768 kHz.

The analog-to-digital conversion process of the HKADC requires 16 cycles of the HKADC Clk.

The general HKADC conversion process is shown in Figure 4-64. The HKADC must be enabled and powered up for at least 5 microseconds prior to the beginning of any conversion. Once the power-up requirement has been met, the first step in the conversion process is to switch the analog input multiplexer to the desired channel by writing the HKADC_MUXSEL[7:5] bits in the HKADC_CNTL2 (analog SBI) register. This takes one HKADC clock cycle to accomplish before a conversion is initiated.

A conversion is initiated by writing to the HKADC_CONV_START register. The analog signal level to be converted is acquired during the next three HKADC Clk cycles. This time is required to charge the internal sampling capacitor which holds the DC level for the remainder of the conversion process. Eight clock cycles are required after signal acquisition to determine the 8-bit final result. There are four clock cycles of latency at the end of the conversion process to allow the result to propagate to the HKADC_OUT[7:0] register. When HKADC_STATUS[0] sets (1), the 8-bit result is available for reading from the HKADC_OUT[7:0] register.

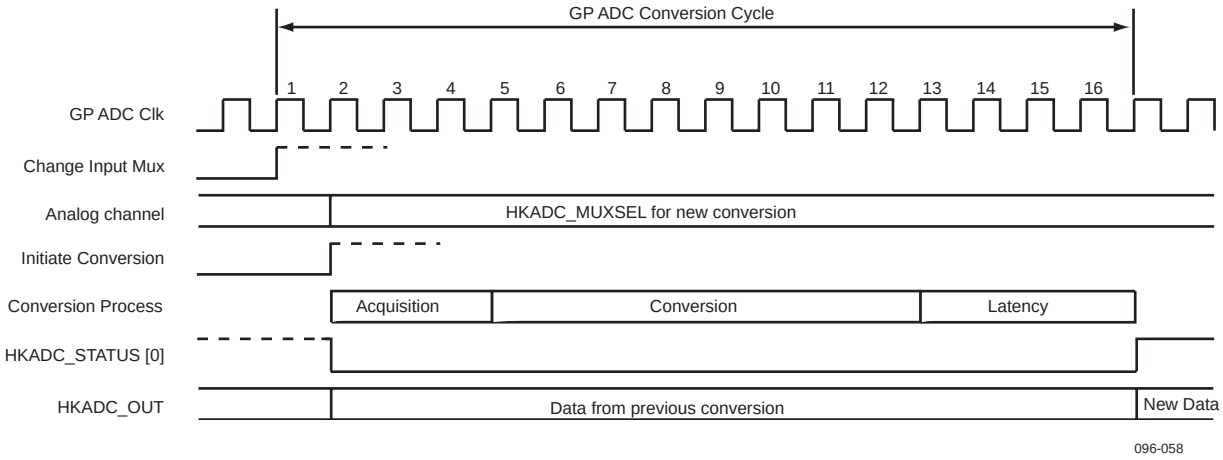


Figure 4-64 General HKADC Conversion Process

4.12.3 HKADC Conversion Time

The conversion time of the HKADC is directly related to the frequency of the HKADC Clk which depends upon the frequency of the clock source selected and the divide factor programmed. Since the MSM6100 device is designed for TCXO frequency of 19.2 MHz, Table 4-35 is arranged to show the conversion time of the HKADC as it relates to the TCXO frequency.

When one of the internally generated Vref sources (1.826, 1.217, or 0.609 Volts) is selected for the HKADC, the maximum frequency for the HKADC Clk signal is 300 kHz. To limit the HKADC Clk to 300 kHz when using the TCXO/4 as the clock source, the divide ratio must be set to 16 (DIV[1:0] = 11). When using TCXO as the clock source, the divide ratio must be set to 16 and the internally generated Vref source cannot be used.

When one of the off-chip Vref sources (V_{DD_A} or Vext) is selected for the HKADC, the maximum frequency for the HKADC Clk is 2.0 MHz. Table 4-35 shows the settings of DIV[1:0] which is used when V_{DD_A} or Vext are used for the Vref of the HKADC. (V_{DD_A} and Vext must have source resistances of 50 Ohms or less).

Table 4-35 ADC Timing for TCXO/4 Clock Source

ADC Clock Divider		TCXO = 19.20 MHz	
DIV[1:0]	Divide ratio	ADC Clock Frequency	Conversion Time
11	16	300 kHz	53.33 µs
10	8	600 kHz	26.67 µs
01	4	1.2 MHz	13.33 µs
00	2	—	—

NOTE For DIV[1:0] = 10 or 01, the HKADC Vref must be input on HKADC6 or V_{DD_A} from a voltage source with source resistance no greater than 50 Ohms.

Table 4-36 ADC Timing for TCXO Clock Source

ADC Clock Divider		TCXO = 19.20 MHz	
DIV[1:0]	Divide ratio	ADC Clock Frequency	Conversion Time
11	16	1.2 MHz	13.33 µs
10	8	—	—
01	4	—	—
00	2	—	—

NOTE The HKADC Vref must be input on HKADC6 or V_{DD_A} from a voltage source with source resistance no greater than 50 Ohms. Internally generated Vref sources (1.817, 1.212, or 0.607 Volts) cannot be used.

4.12.4 HKADC Analog Interface Considerations

For HKADC[6:0] input pins that are not selected by the HKADC input multiplexer, the impedance is very high and the MSM6100 device's pins present essentially no load to external circuits connected to these pins. Figure 4-65 illustrates the equivalent circuit of the HKADC input and the circuits that are designed to drive the HKADC[6:0] pins. The input multiplexer selects an input pin by closing S1. When a conversion is initiated, S2 stays closed until the end of the acquisition interval (three HKADC Clk cycles). After acquisition, S2 opens and the DC voltage to be converted is held on the sampling capacitor, Cs.

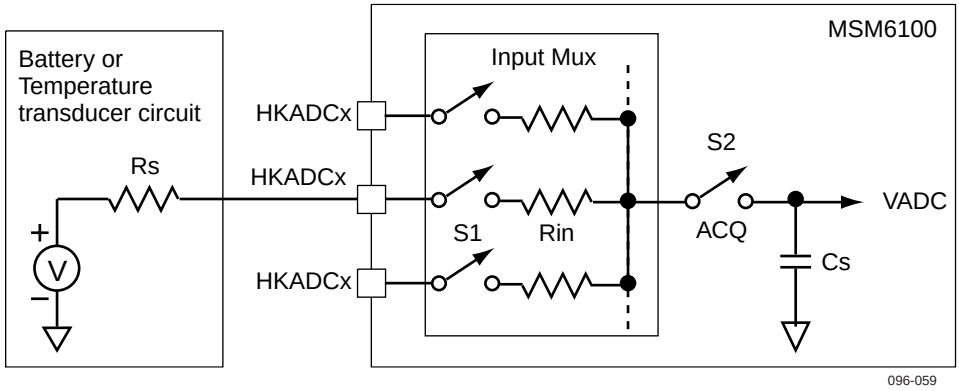


Figure 4-65 Equivalent Circuits for HKADC Input and External Voltage Sources

During the acquisition interval, the external circuit connected to the selected HKADC pin must supply enough current to charge Cs through the 5 kOhm (maximum) on-resistance of the input multiplexer. For accurate conversion results, the voltage on Cs must settle to within 0.25% of its final value (~1/2 LSB) in acquisition interval. Cs is approximately 12 pF.

The relationship that must be met is given by:

$7 \times (Rs + Rin) \times Cs < 3 / f_{ADC\ Clk}$

This relationship is used to determine the maximum source resistance of the external circuits driving the HKADC input pin. Examples of this relationship are shown in Table 4-37.

Table 4-37 Recommended Rs Maximum Values

fADC Clk	Rs max.	CLK Conditions	Vref Conditions
300 kHz	100kOhm	TCXO = 19.2 MHz	Internal 1.826, 1.217, or 0.609 Vref is selected.
300 kHz	100 kOhms	TCXO = 19.2 MHz	VDD_A or Vext selected for Vref. (Source resistance of VDD_A or Vext is < 50 Ohms)
600 kHz	50 kOhms	TCXO = 19.2 MHz	VDD_A or Vext selected for Vref. (Source resistance of VDD_A or Vext is < 50 Ohms)
1.2 MHz	25 kOhms	TCXO = 19.2 MHz	VDD_A or Vext selected for Vref. (Source resistance of VDD_A or Vext is < 50 Ohms)

4.13 Clock Regime

The MSM6100 device derives all of its internal clock sources from three clock inputs, TCXO, SLEEP_XTAL, and 48XTAL. The TCXO clock input supports 19.2 MHz. An integrated PLL and M/N counter is used to create the required clock sources when the TCXO frequency is 19.2 MHz. The SLEEP_XTAL can support a 32.768 kHz clock source to drive the sleep controller during periods when most of the MSM6100 device is powered down and the TCXO is disabled. A 48 MHz crystal oscillator is used to generate the clock source for the USB interface and is not required for any other subsystem of the MSM6100 device. It is recommended that the USB clock source be shut down when the USB interface is not being used.

4.13.1 TCXO

The MSM6100 device integrates a phase-locked loop and an M/N counter to derive CHIPX16 and CHIPX8 from the TCXO clock input.

4.13.2 CODEC PLL

A separate fractional PLL will be available and can be used to generate the jitter-sensitive clocks for the wideband audio codec. The supersets of frequencies that need to be generated by the codec PLL are:

- 11.2896 MHz: Required for the noise shaper in the codec, will be used to generate the 5.6448 MHz clock also required by the codec interface.
- 12.288 MHz: Required for the noise shaper in the codec, will be used to generate the 8.192 MHz, 6.144 MHz, 4.096 MHz, 2.048 MHz, and 1.024 MHz clocks also required by the codec interface.

When the SLEEP_XTAL frequency is 4.096 MHz, the CODEC PLL can be bypassed by setting CODECPLL_CTL:CPLL_CLKSEL and CODECPLL_CTL:CPLL_OUTSEL allowing the CODEC to be driven directly from the SLEP_XTAL. The CODEC PLL can also be used to multiply the SLEEP_XTAL frequency in order to generate a clock source for both the ARM926EJ-S and the QDSP4000. Setting (1) CODECPLL_CTL:CPLL_CLKSEL selects the SLEEP_XTAL as the input of the CODEC PLL. The CODECPLL_MULT values must be chosen to create the correct VCO center frequency selected by CODECPLL_MODE:CPLL_EXTRAC.

4.13.3 SLEEP Crystal Circuit for 32.768 kHz

The MSM6100 device contains a SLEEP oscillator circuit used as a clock source when other clocks are disabled to conserve power. Bits 11:10 of MSM_CLK_SLEEPOSC control the gain of the internal inverter in the MSM6100 device's SLEEP oscillator circuit. Bits 11:10 are collectively called SLEEP_OSC_GAIN. Table 4-38 shows the gain settings of the inverter.

Table 4-38 SLEEP Oscillator Inverter Relative Gain Settings

SLEEP_OSC_GAIN	Gain
00	Minimum
01	
10	Default
11	Maximum

Bit 9 of MSM_CLK_SLEEPOSC, called SLEEP_OSC_RF_BYPASS controls the feedback resistance between the input and output of the internal inverter. Set (1) SLEEP_OSC_RF_BYPASS to disable the feedback path. Clear (0) SLEEP_OSC_RF_BYPASS to enable the feedback path. Bit 8 of MSM_CLK_SLEEPOSC, called SLEEP_OSC_RD_BYPASS controls the series resistance between the output of the internal inverter, and the SLEEP_XTAL_OUT pin. Set (1) this bit to disable series resistance between the inverter's output and the SLEEP_XTAL_OUT pin. Clear (0) this bit to enable series resistance between the inverter's output and the SLEEP_XTAL_OUT pin

Figure 4-66 shows an example 32.768 kHz SLEEP passive crystal circuit configuration with load capacitors. The inverter's gain is at its lowest setting. S1 is closed, enabling the inverter feedback path. S2 is open, enabling the series resistance between the inverter's output and SLEEP_XTAL_OUT.

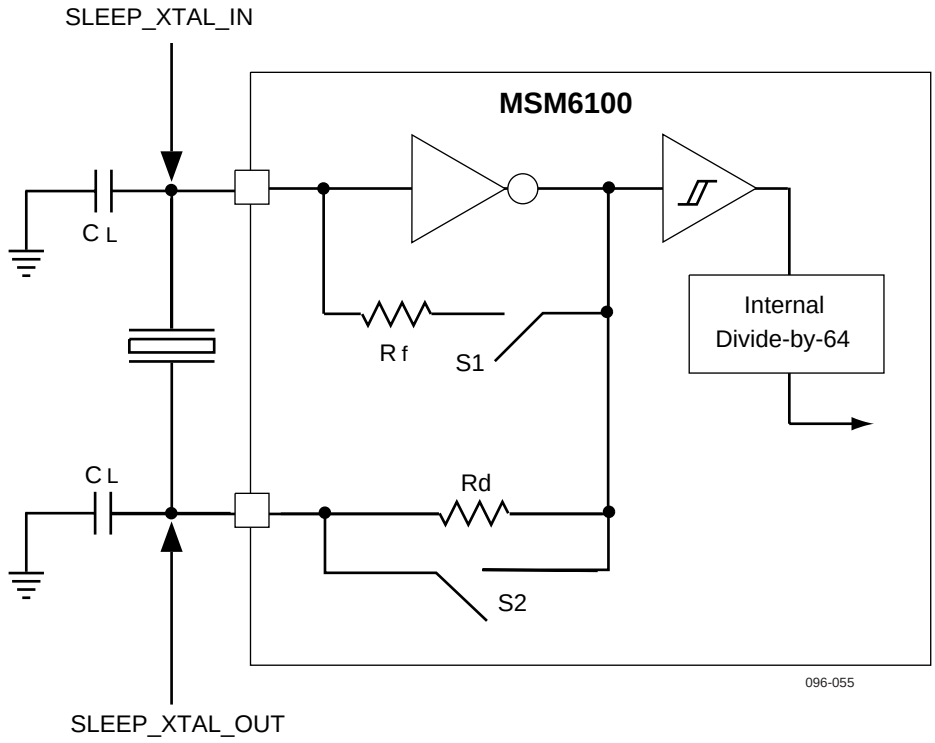


Figure 4-66 SLEEP Oscillator Circuit with Passive Crystal

The SLEEP oscillator circuit also supports an external, passive oscillator circuit. To configure the SLEEP oscillator circuit for an external passive oscillator, the gain should be set to maximum, in order to produce fast transitions at both the output and input of the inverter.

To reduce power consumption while the SLEEP oscillator is not being used, both S1 and S2 should be set to open. Opening S1 disables the inverter's feedback path, and reduces power consumption. Opening S2 disables the low impedance path between the inverter and the SLEEP_XTAL_OUT pin.

If the active oscillator has an open-drain output, care must be taken in choosing the pull-up resistor. The pull-up resistor must be less than 20 kOhms. If the output of the external oscillator circuit drives both high and low, the pull-up resistor is not required.

4.13.4 USB Crystal Circuit for 48 MHz

Figure 4-67 shows the architecture of the USB oscillator in the MSM6100 device.

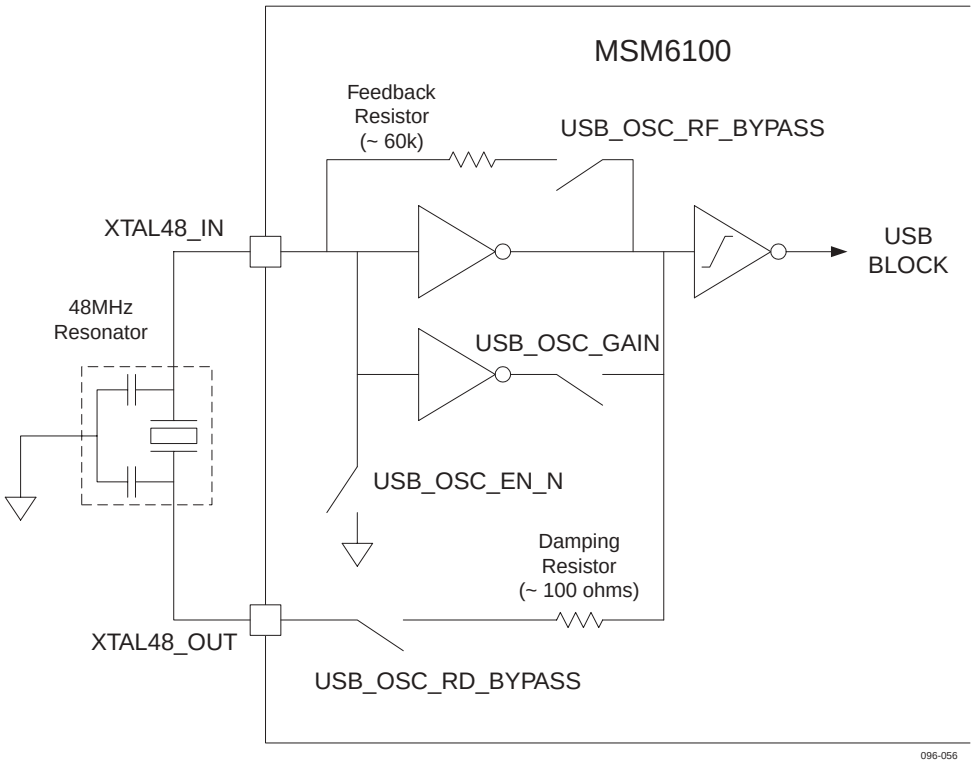


Figure 4-67 USB Oscillator Circuit

The oscillator can work with either a crystal or a resonator. Resonators tend to be smaller, and cheaper, while crystals tend to have tighter frequency tolerances. The above diagram shows a resonator being connected. Most resonators have internal bypassing caps as shown above.

The USB specification allows a peripheral to have an operating frequency of 12 Mbps +/– 2500 ppm. This tolerance is achievable with resonators. However, the isochronous endpoints of the MSM6100 USB interface have a limitation that requires the frequency of the oscillator to be less than 48 MHz +500 ppm. This may require the use of a crystal instead of a less expensive resonator. All other endpoints of the MSM6100 USB interface work over the full range of 12 Mbps +/– 2500 ppm.

The USB oscillator includes an internal feedback resistor. The feedback resistor switch is closed when the following bit is zero, i.e.,:

- MSM_CLK_USBOSC: USB_OSC_RF_BYPASS = 0
  (feedback resistor switch closed)

During times when the USB oscillator is not being used, the feedback resistor should be set to open in order to reduce current consumption.

The USB oscillator includes a second gain stage. In practice, the second gain stage has little effect on performance, because of the damping resistor which is always in series with these two gain stages.

The second gain stage switch is closed when the following bit is one:

- MSM_CLK_USBOSC: USB_OSC_GAIN = 1
  (oscillator gain switch closed)

The USB oscillator includes a switch that can enable or disable the oscillator. This switch is closed and the oscillator is disabled when the following bit is one:

- MSM_CLK_USBOSC: USB_OSC_EN_N = 1
  (oscillator enable switch closed)

The USB oscillator includes an internal damping resistor. When the damping resistor switch is open, the oscillator is disabled. The damping resistor switch is closed when the following bit is one:

- MSM_CLK_USBOSC: USB_OSC_RD_BYPASS = 1
  (damping resistor switch closed)

When the oscillator is not being used, it can be disabled by closing the oscillator enable switch. However, if the feedback resistor switch also remains closed, then a constant current will flow from the output of the oscillator inverter through the oscillator switch to ground. Thus, in order to conserve power, it is recommended that the switches be set as follows when the oscillator is not being used:

- MSM_CLK_USBOSC : USB_OSC_RF_BYPASS = 1
  (feedback resistor switch open)
- MSM_CLK_USBOSC : USB_OSC_EN_N = 1
  (oscillator enable switch closed)

4.13.5 Clock

The clock block is a single-phase clock generator which is responsible for providing all clocks required by internal cores and external interfaces. Figure 4-68 shows the clock block architecture. Via a frequency synthesizer constructed using two fractional phase-locked loops (PLLs), all required CDMA, DFM, GPS, and miscellaneous peripheral clocks are generated from the two mandatory external oscillators, TCXO and sleep XO. The only supported TCXO oscillator frequency is 19.2 MHz. The sleep XO (SLEEP_XTAL) can range from 30 kHz to 60 kHz (32 kHz target). An optional external USB oscillator (USB XO or USB_XTAL) at 48 MHz may also be provided to support USB functionality.

The MSM6100 device's PLLs are used to generate a couple high frequency, less-jittery clock sources. All other clocks are derived from these clock sources or from the external oscillators via divider circuits and M/N:D counters.

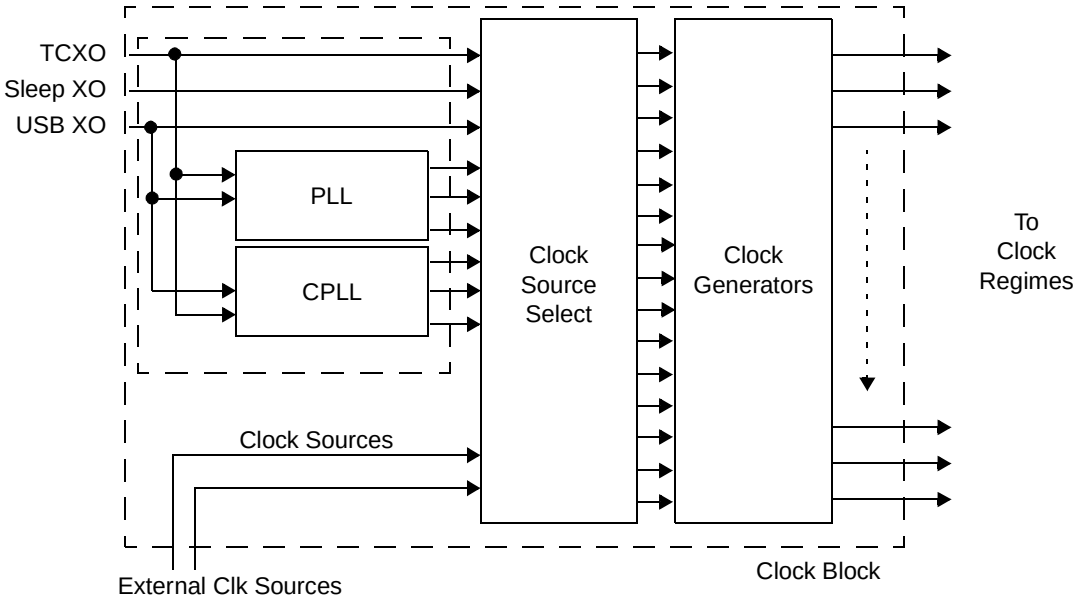


Figure 4-68 Clock Block Architecture

The clock generation architecture is designed to support two different clock generation methodologies. In the first scenario, when sleep XO and TCXO are the only external oscillators, all the internal clocks, except during Sleep/QPCH off-line processing, are generated from TCXO.This approach is thought to have advantages in the phone area and Bill of Materials since a USB XO is not required for USB functionality.

The second scenario occurs when a USB XO is also present with the sleep XO and TCXO. In this case, TCXO is only powered up during an active call, and the USB XO is used to generate all non-CDMA clocks, such as PCM, GPS, and USB clocks. This second approach is thought to have advantages over the first case in power savings by only requiring TCXO to be active when absolutely necessary. Due to the temperature-compensation circuity and large array of drivers on the TCXO clock line, TCXO consumes much more power than a typical oscillator circuit. It also allows software to employ a simpler clock sourcing algorithm by always having a clock source available which can be used to generate all internal clocks, especially for peripheral clocks such as USB, GPS, or codec/PCM. The disadvantage to this approach, of course, is the additional cost, in terms of board area and component cost, of adding another external oscillator to the handset.

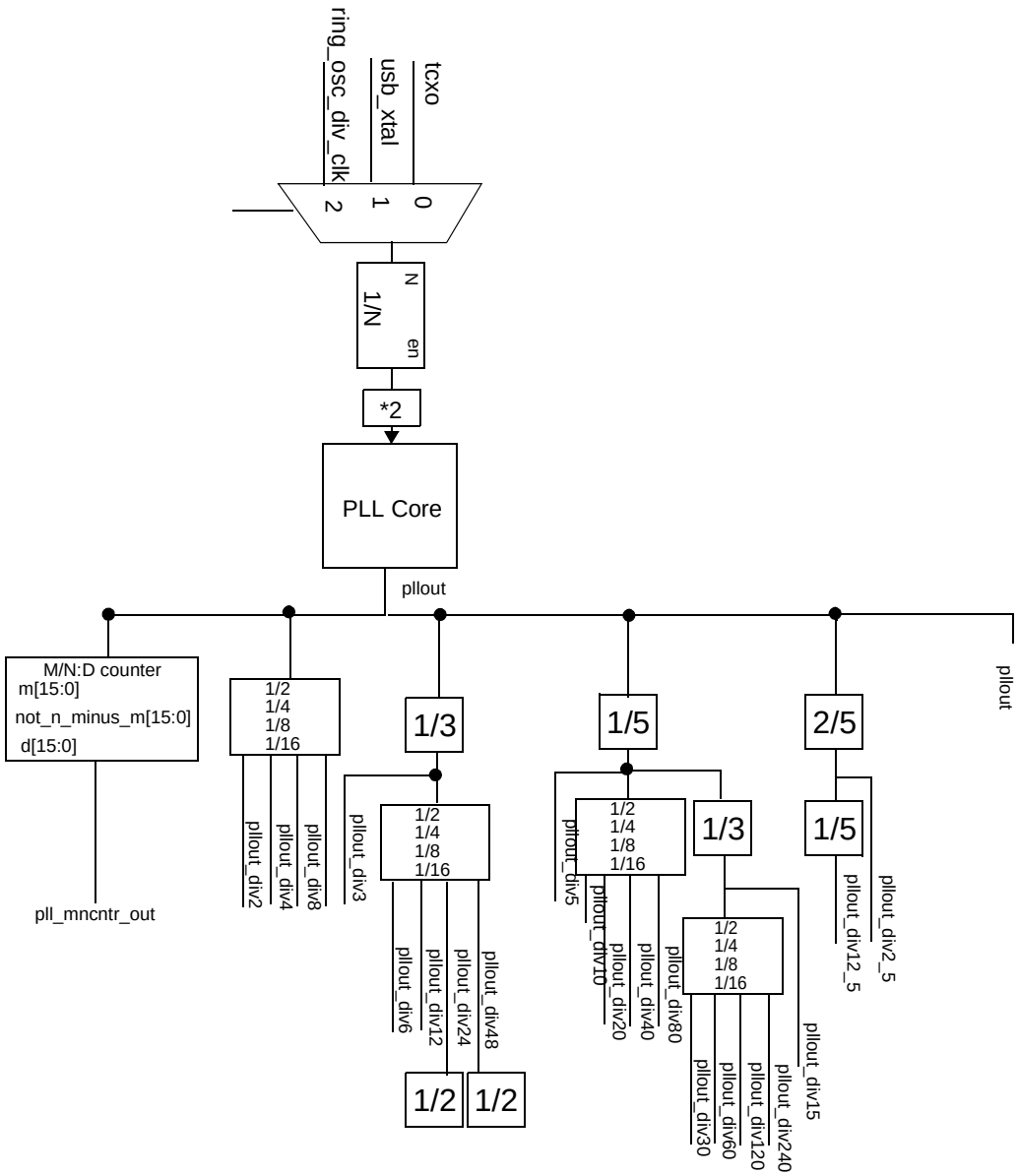


Figure 4-69 Clock Source Generation From PLL Core Block Diagram

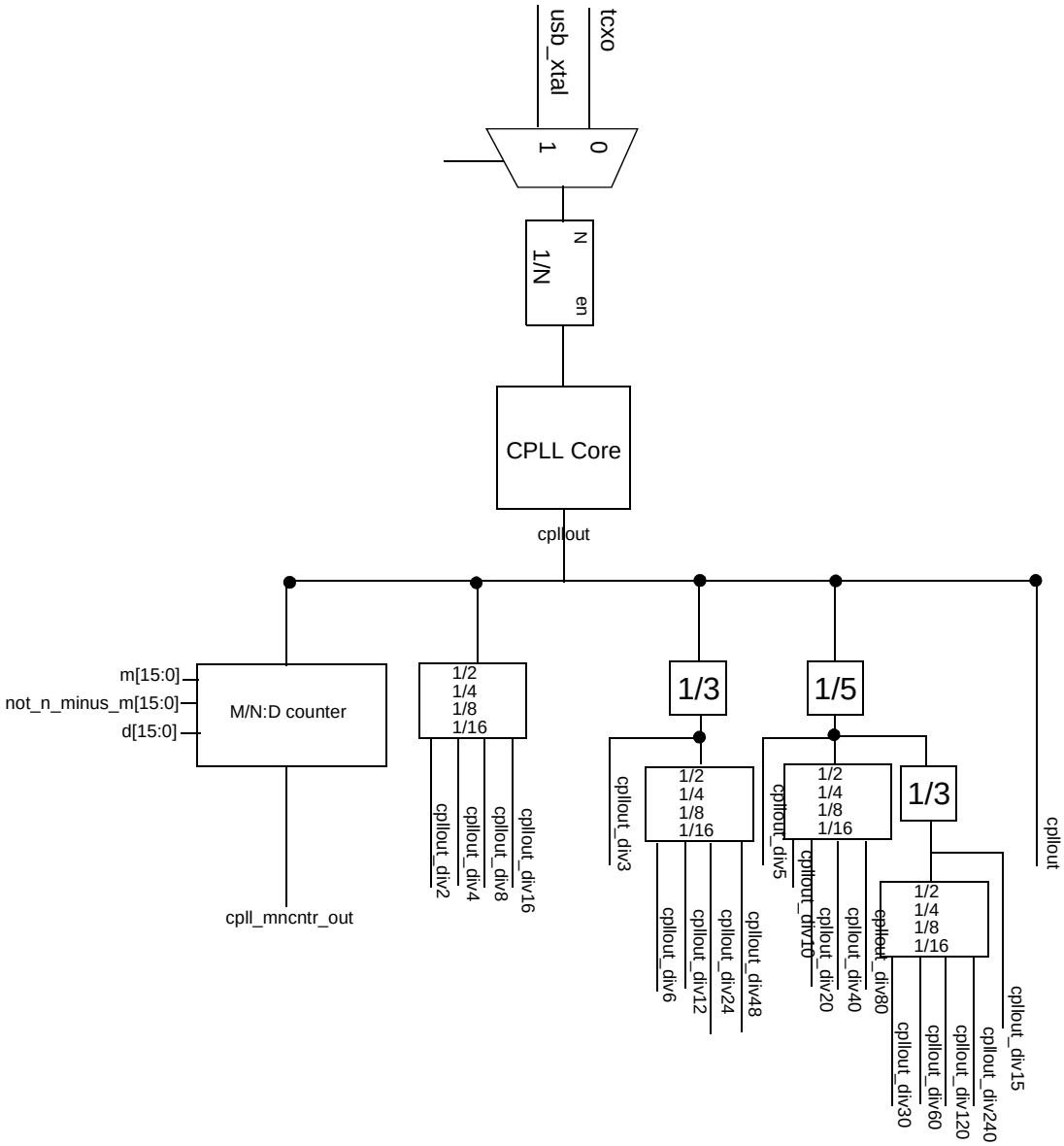


Figure 4-70 Clock Source Generation From CPLL Core Block Diagram

4.13.6 Subsystem Clock Regimes

Each clock regime is separately enabled, initialized and reset by the MSM_CLK_CTL registers.

Table 4-40 lists the register bit selection for MSM_CLK_SRCSEL, the subsystem that it controls and the clock source options. Table 4-39 defines the clock origins.

Table 4-39 Descriptions of Clock Origins

Clock	Description
CHIPX16	Output of the M/N counter and TCXO PLL, path depends on TCXO frequency
CHIPX8	Output of the M/N counter and TCXO PLL, path depends on TCXO frequency
PLLOUT	Output of the TCXO PLL
GPS	Output of the TCXO PLL
TCXO/4	TCXO input divided by 4
TCXO	TCXO input
Sleep	Sleep Oscillator Output
48XTAL	48 MHz Oscillator Output

Table 4-40 MSM6100 Device Clock Regimes

Name	Subsystem	Clock Origin	CLK SRC DIV
PDM	TRK_LO_ADJ, TX_AGC_ADJ	pll -> chipx16, pll -> gps_chipx16, tcxo	
USB	USB	usb_xtal	
CODEC	Internal Audio Wideband CODEC	cpllout_div* cpll_mncntr_out, pllout_div*, ext_test_clk	1 1/2 1/3 1/4
aDSP	Vocoder QDSP4	tcxo, pllout_div*, cpllout, ext_test_clk, sleep_xtal, usb_xtal	1 M/N
GPS	GPS	gps_chipx8, gps_chipx16, gps_chipx32	
DEMOD	DEMOD	tcxo -> chipx8	
CDMA RX	Searcher, Combiner, FFE	tcxo -> chipx8	
mDSP	Demod DSP	tcxo, chipx8, pllout_div*, sleep_xtal, usb_xtal, cpllout, ring_osc, ext_test_clk	1 M/N
Decoder	Decoder (Deinterleaver)	tcxo, usb, chipx8, cpllout, pllout_div*, ring_osc	1 1/2 1/3 1/4
CDMA TX	Modulator, Interleaver	tcxo -> chipx16	
SLEEP	Sleep Controller, Watchdog	tcxo -> chipx8	
DFM	DFM M/N, micro interfaces, PDMs: TRK_LO_ADJ	tcxo	
UART1	UART1, HKADC	tcxo, tcxo/4 sleep_xtal, GPIO10	
UART2	UART2	tcxo, tcxo/4 sleep_xtal, GPIO11	

Table 4-40 MSM6100 Device Clock Regimes (Continued)

Name	Subsystem	Clock Origin	CLK SRC DIV
UART3	UART3	tcxo, tcxo/4 sleep_xtal, GPIO12	
GENERAL	Ringer, Timetick, PDM1,PDM2, gp_mn	tcxo/4 sleep_xtal	
ARM	ARM Core, AHB	TCXO, pllout_div* gpio, sleep_xtal, Ring Osc, usb_xtal	1 1/2 1/3 1/4 1/8 1/16 1/32 1/64
SDAC	Vocoder, aux-codec stereo DAC interface, I2S	tcxo -> SDAC M/N Counter, 1.024 MHz, ext clk src, icodec_clk / 8	
Ext CODEC CLK	Aux codec PCM core clock.	pll -> 4.096 MHz, cpll -> 4.096 MHz, bt_clk -> M/N cnt, sdac_clk	

4.13.6.1 ARM Processor and AMBA Clock Selection

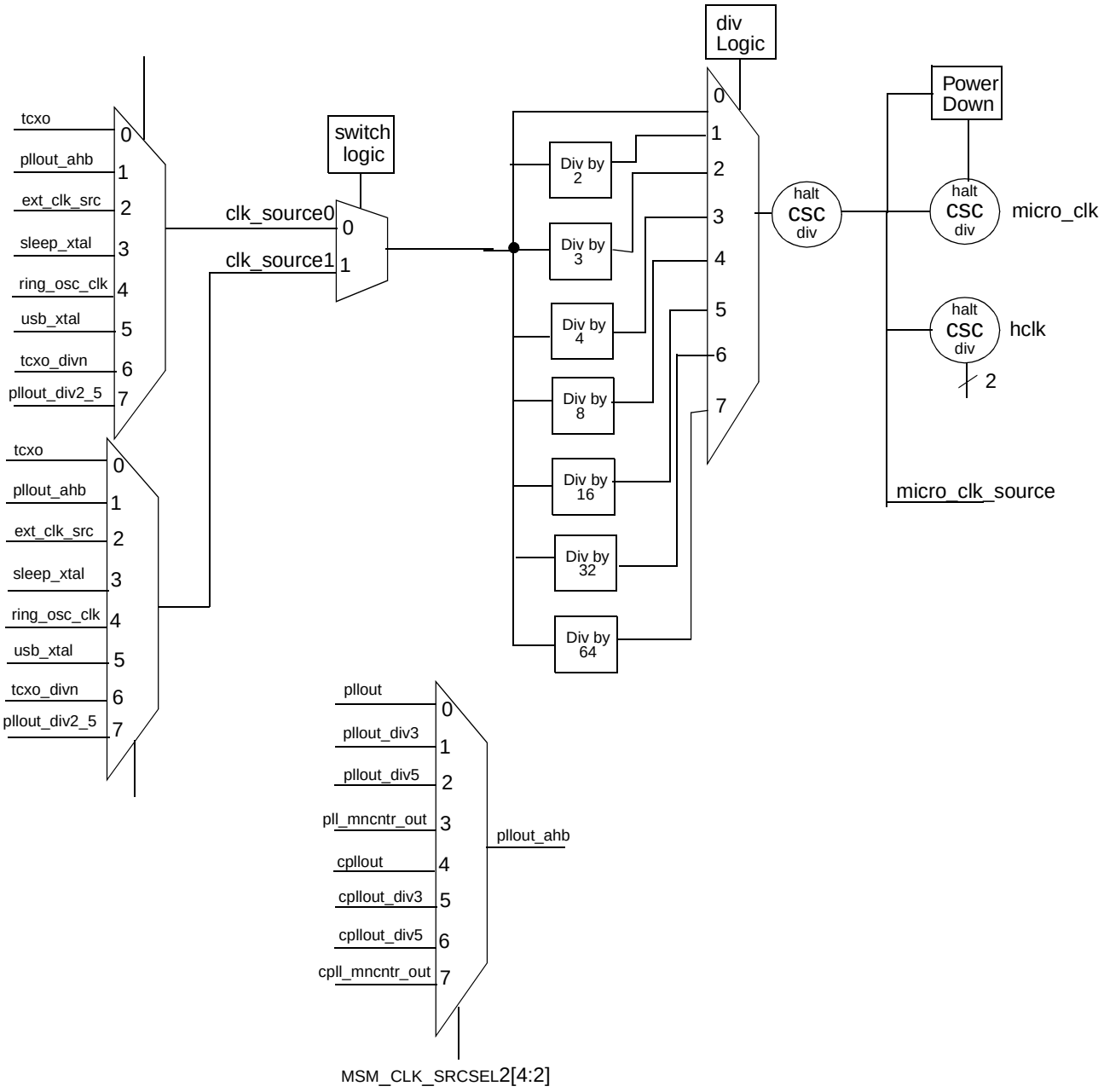


Figure 4-71 ARM Clock Selection

4.13.6.2 External stereo DAC clock selection

As shown in Figure 4-71 and Figure 4-72, the clock selection circuitry for the external stereo DAC is similar to previous MSMXXX devices.

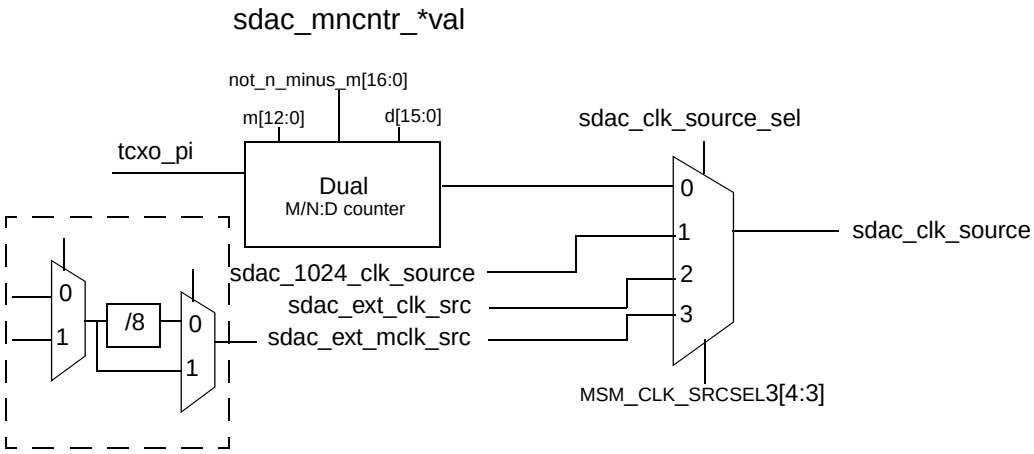


Figure 4-72 External Stereo Codec Clock Selection Block Diagram

NOTE The sdac_clk cannot be the sdac_1024_clk_source as long as the ecodec_clk is using the sdac_clk as its source. This would create a loop where the clock is never driven.

4.13.6.3 External (Auxiliary) Codec Clock Selection

The ecodec_clk will be a 4.096-MHz clock that is used by an off-chip, eternal audio codec. As shown in Figure 4-73, it has four potential sources. Notice as well that the SDAC 1.024 MHz clock is the external codec clock source divided down by 4.

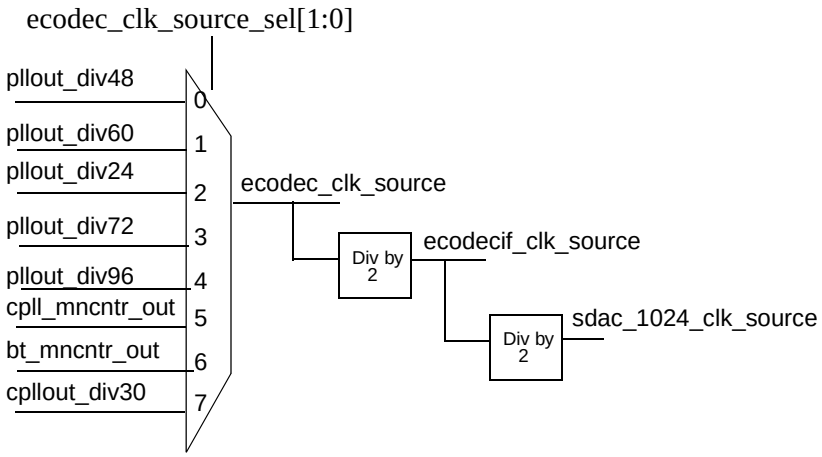


Figure 4-73 Ecodec Clock Selection

4.13.6.4 Internal Wideband Codec Clock Selection

The internal wideband audio codec requires a variety of clock frequencies for different applications. To obtain all the required clocks independent of VCO frequency, a selection scheme with optional M/N counter is used as shown in Figure 4-74.

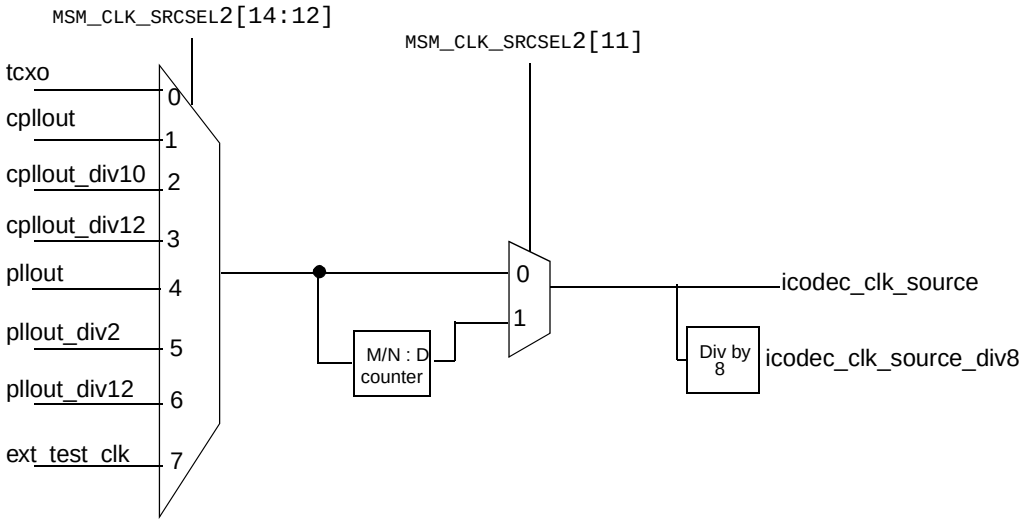


Figure 4-74 Internal Wide Bank Codec Clock Selection Block Diagram

4.13.6.5 SBI Clock Source Selection

As shown in Figure 4-75, the SBI clock source is chosen by the microprocessor. Ideally it is 5 to 10 MHz, but depending on limitations of the analog chips, this may need to be more flexible. So, a dual edge M/N counter is also included as an input to the selection mux.

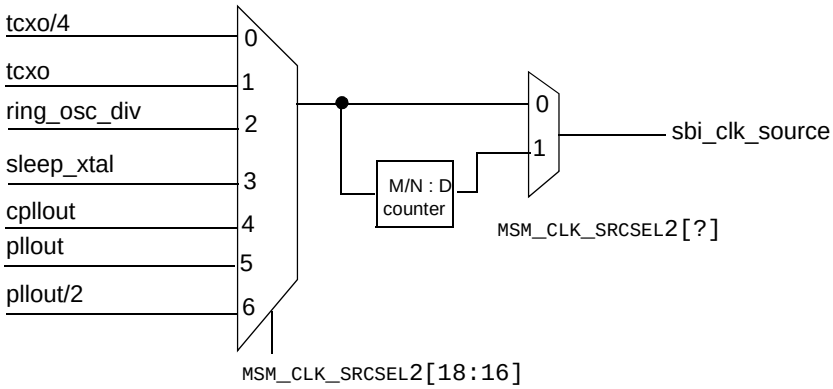


Figure 4-75 SBI Clock Source Selection

4.13.6.6 Multimedia Card and Controller Clock Selection

As shown in Figure 4-76, the multimedia card controller clock can be chosen to be one of four sources with an optional divide ratio of 1-4. The multimedia card clock is fed back to the clock block from the controller. It can be chosen to be the internal version of the mmc_div_clk_in or the external version. The external version will be fed from the pad and thus is a better representation of the actual clock seen by the external card.

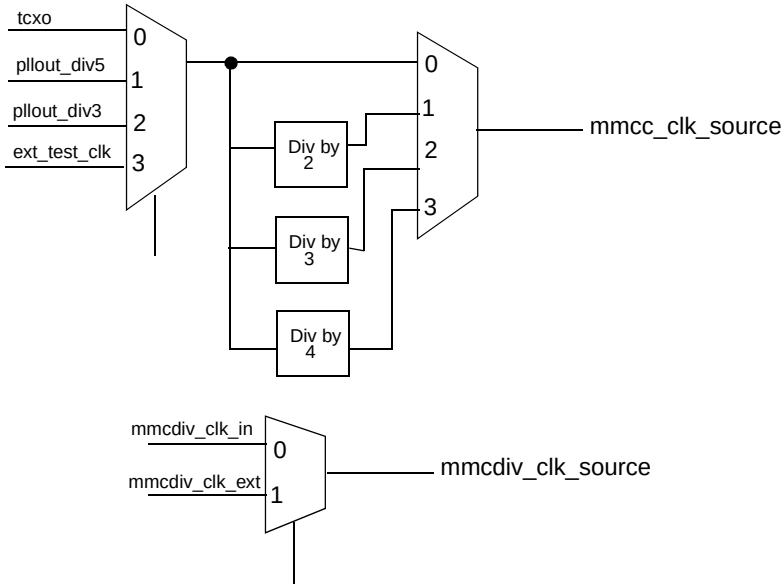


Figure 4-76 MMC Clock Selection

4.13.6.7 Camera Interface Clock Selection

As shown in Figure 4-77, the camera interface clock will have an input from an external clock source from a camera and will produce the desired camera interface clock in the CAMIF block itself.

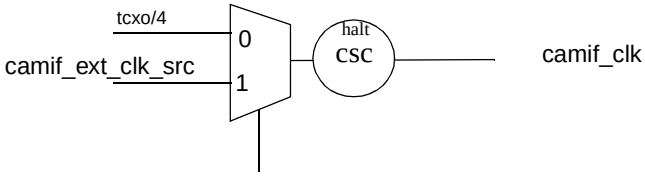


Figure 4-77 Camera Interface Clock Selection

4.13.6.8 USB Clock Selection

As shown in Figure 4-78, the USB 48 MHz clock source can be chosen to be one of several sources. In the past, the xtal_48_in pin of the MSM was fed to an internal oscillator and then used as the only source of the USB 48 MHz clock. The various options in the MSM6100 allow the phone to have USB capability without an external crystal. This could potentially save phone area, bill of material, and power.

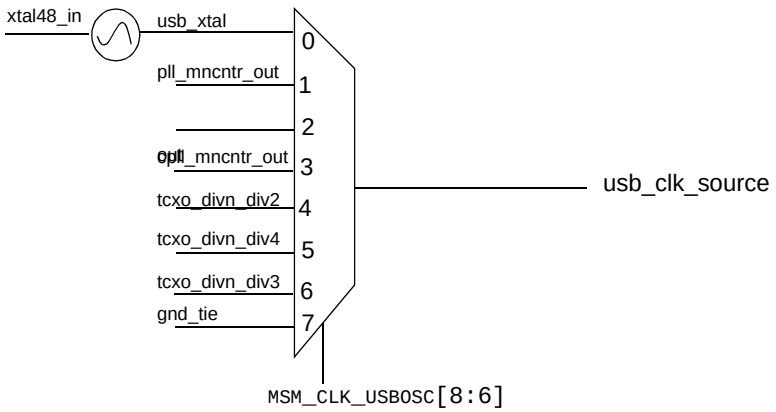


Figure 4-78 USB 48 MHz Clock Selection

4.13.6.9 I²C Clock Selection

As shown in Figure 4-79, the I²C clock is tied to the microprocessor clock frequency, but has a 6-bit modulo-N counter to divide it down to an acceptable frequency. The I²C clock must lie between.

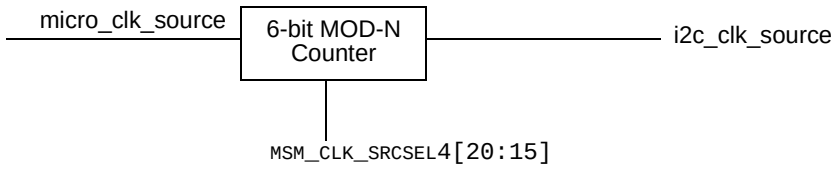


Figure 4-79 I²C Clock Selection

4.14 JTAG Interface

The JTAG Interface on the MSM6100 device aids in Mobile Station board-level testing, and debugging.

Referring to the IEEE 1149.1A-93 manual: IEEE Standard Test Access Port and Boundary-Scan Architecture, the compliance clause defines how compliance with this standard is “switched on” or “switched off.”

In the MSM6100 device, the TAP selection is done as part of the MODE selection for the device. For normal operation and to select the ARM JTAG TAP, the MODE 1:0 pins are left unconnected (they are pulled low internally). To disable the Watchdog, the WDOG_EN pin must be pulled low externally; the WDOG_EN pin is pulled high internally to enable the Watch_Dog for normal operation. To select the MSM6100 device scan chain TAP, the MODE 1:0 pins must be pulled high externally.

NOTE This implementation does not conform to the IEEE 1149.1-90 standard. This implementation meets the intent of the IEEE 1149.1A-93.

It is important to note that the tap controller which is not selected is held in the TEST_LOGIC_RESET state. Since the JTAG port is shared, the TMS and TCK pins are not seen by both TAP controllers. To guarantee that both TAP controllers are in the TEST RESET STATE, the TRST_N pin must be pulled low externally during the power-on sequence.

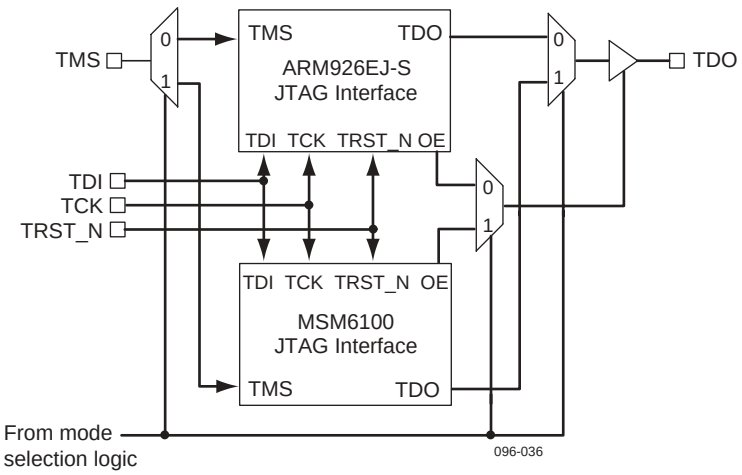


Figure 4-80 MSM6100 Device JTAG Interface

The JTAG interface allows test instructions and test data to be shifted into the MSM6100 device, and the test results shifted out in a serial format.

There are four main functional elements in the JTAG Interface: a Test Access Port (TAP), a TAP controller, an Instruction Register, and a group of Test Data Registers. Figure 4-81 is a block diagram of the JTAG Interface.

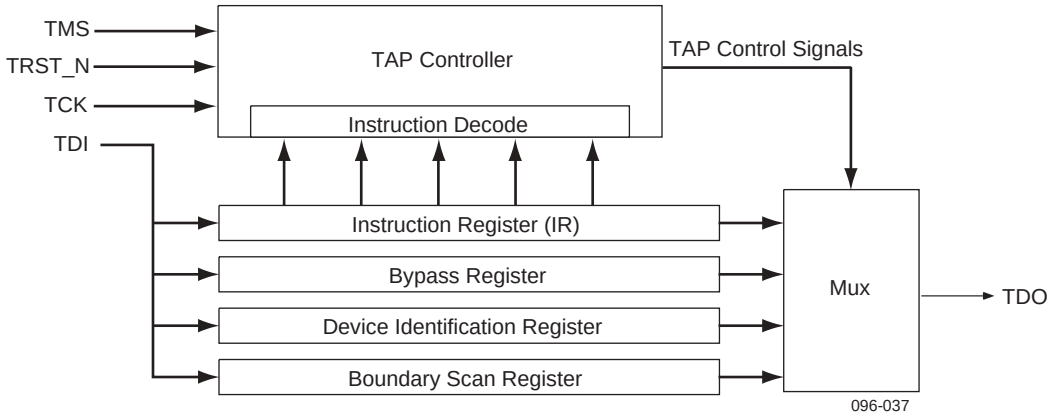


Figure 4-81 JTAG Interface Block Diagram

4.14.1 Test Access Port

The JTAG Interface is accessed through the Test Access Port (TAP). The TAP includes the following input and output (I/O) pins:

- TCK (test clock input)
- TMS (test mode select input)
- TDI (test data input)
- TDO (test data output)
- TRST_N (test reset input)

The TAP dedicates all its I/O pins to the JTAG interface, these pins have no other use on the MSM6100 device.

TCK is a user-defined clock input for the JTAG Interface. TCK is independent of the system clock, it supplies a clock input to the serial test path between TDI and TDO. TCK also permits shifting of test data concurrent with MSM6100 device system operation. The independent TCK clock ensures that test data is shifted into and out of the TAP without changing the state of the MSM6100 device.

The TAP controller decodes the TMS input to set up test operations. TMS is sampled on the rising edge of TCK. When unconnected, the TMS input is a logic high (via an internal pull-up) to ensure that the MSM6100 device continues operating without interference from the test logic.

TDI is the TAP input for serial test instructions or test data. TDI is sampled on the rising edge of TCK. TDI and TDO are the input and output ports for serial test data through the MSM6100 device. Data propagates from TDI to TDO without inversion. When unconnected, the TDI input is a logic high (via an internal pull-up). This ensures that open-circuit faults in the board-level, and the serial test data path forces a defined logic value to be shifted into the TAP.

TDO outputs the TAP serial test data. To ensure race-free operation, changes on TDO occur on the falling edge of TCK, while changes on the TAP inputs (TMS and TDI) occur on the rising edge of TCK. TDO is three-stated unless the data scanning is in progress.

The TRST_N input asynchronously resets the TAP controller when driven to a logic low. When TRST_N is low, all other test logic is asynchronously initialized. When unconnected, the TRST_N input is a logic low (via an internal pull-down).

4.14.2 TAP Controller

The TAP controller is a synchronous finite state machine that responds to changes in the TMS and TCK signals The TAP controller controls a sequence of operations. Its TAP state machine has 16 states, allowing both data and instructions to be shifted. There are states for capturing, shifting, and updating data and instructions, a state for running tests, and a reset state.

The TAP Controller initializes when it is in the Test-Logic-Reset state. There are two ways to get the TAP Controller into the Test-Logic-Reset state:

- Applying a logic low to TRST_N brings the TAP Controller asynchronously into the Test Logic Reset state.
- Holding the TMS high, while allowing at least five rising edges of TCK to occur, brings the TAP Controller synchronously into the Test Logic Reset state. This is the worst case time to reach the Test Logic Reset state from any other state in the TAP controller.

Once in the Test-Logic-Reset state, all test logic disables and normal MSM6100 device operations occur.

For board-level designs where JTAG testing is not used, TRST_N must be a logic low (externally pulled low by a low value resistor or tied to VSS). Before using the MSM6100 device in a Mobile Station application, TRST_N must be low at power-up; or the TAP Controller must be in the Test-Logic-Reset state.

4.14.3 Data Registers

There are three Data Register types included in the MSM6100 device JTAG Interface: Device Identification Register, Bypass Register, and Boundary Scan Register.

4.14.3.1 Device Identification Register

The Device Identification Register is a 32-bit register holding the device's Manufacturer Identity Code, Part Number, and Version data. The upper nibble of the Device ID register (the "version" field) is supplied by the lower nibble of the HARDWARE_REVISION_NUMBER register. The LSBit is the Device ID register's start bit and is set to a logic 1. The Manufacturer Identity Code (bits [11:1[) is a manufacturer coding scheme administered by JEDEC (Joint Electron Device Engineering Council). QUALCOMM's manufacturing code is 0x070. The Part Number (Bits [27:12]) is a QUALCOMM-generated number (0x2e for the MSM6100 device). The Version number (Bits [31:28]) is a QUALCOMM-generated number (0x1 for the production version). Selecting the Device Identification register, loads its value into the shift register on the rising edge of TCK in the Capture-DR state. The Device Identification register does not have a parallel output. Figure 4-82 shows the structure of the Device Identification Register.

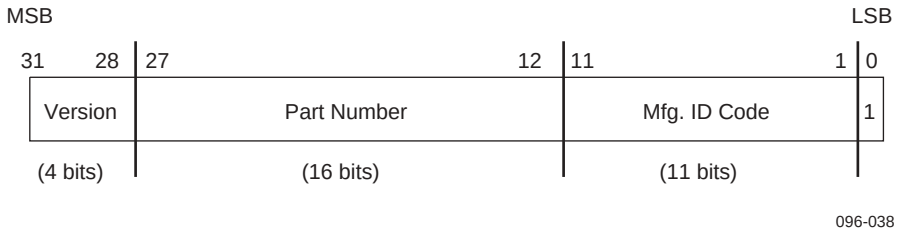


Figure 4-82 Data Structure for Device Identification Register

4.14.3.2 Bypass Register

The Bypass Register is a single-shift register stage located between TDI and TDO. The Bypass Register is used to route boundary scan data directly from TDI to TDO without going through the entire Boundary Scan Register. Bypassing the Boundary Scan Register allows more rapid movement of test data to and from other components on a board. Selecting the Bypass Register, loads the shift register with a logic low on the rising edge of TCK; following entry into the Capture-DR state. The Bypass Register does not have a parallel output.

4.14.3.3 Boundary Scan Register

The Boundary Scan Register allows board interconnection testing to detect typical defects, like opens and shorts. The Boundary Scan Register connects between each digital I/O pin and the MSM6100 device's internal circuits; to allow I/O access when testing system logic, or to sample system I/O signals. The Boundary Scan Register also handles parallel input and outputs.

4.14.3.4 Boundary Scan Cells

The Boundary Scan Register contains cells that access digital signals (Table 4-41) at the MSM6100 device. Figure 4-83 shows a typical schematic diagram of a single boundary scan cell and its control signals.

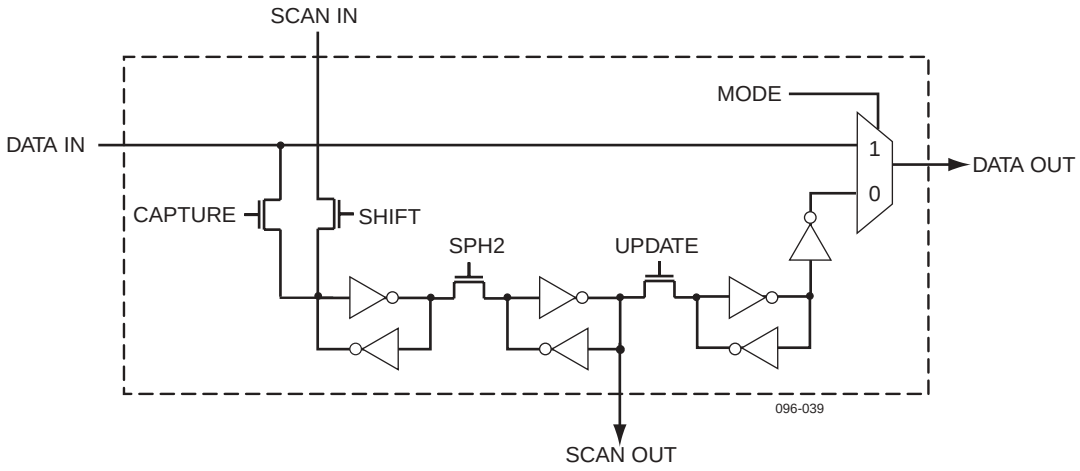


Figure 4-83 Typical Boundary Scan Cell

4.14.4 JTAG Instructions

The Instruction Register (IR) holds JTAG instructions that were shifted into the JTAG Interface. A JTAG instruction shifts though TDI, LSB first. In the IR, the JTAG instruction is used to select the test to be performed and/or the test data register to be accessed.

The IR is a 7-bit shift register having a serial input and parallel latched output. The parallel output holds the current instruction and is updated on the falling edge of TCK (when in the Update-IR state) or asynchronously upon entry into the Test-Logic-Reset state.

The instruction register allows instructions to be serially entered into the test logic during a instruction register scan cycle. Instructions loaded in the instruction register are decoded so that each instruction selects a test data register that operates during the current instruction. All other test data registers are controlled, so that they do not interfere with MSM6100 device operation, or with the selected test data register operations.

The supported instructions and their corresponding codes are listed in Table 4-41.

Table 4-41 Boundary Scan Cell Control Signals

Cell Signal	Description
DATA IN	Data input coming from either MSM6100 device circuits (pin_OUT or pin_OE cells) or from the pin (pin_IN cell).
DATA OUT	An output going to either MSM6100 device circuits (pin_IN cell) or to the pin (pin_OUT or pin_OE cells).
SCAN IN	SCAN IN from previous cell's SCAN OUT or the TDI pin if it is the first cell in the sequence.
SCAN OUT	SCAN OUT to the SCAN IN of the next cell or the TDO pin if it is the last cell in the sequence.

Table 4-41 Boundary Scan Cell Control Signals

Cell Signal	Description
MODE	The MODE signal from the TAP Controller. A logic low selects the Boundary Scan test logic and a logic high selects normal MSM6100 device data flow.
CAPTURE, UPDATE, SHIFT	Control signals from the TAP Controller.
SPH2	A clock signal generated from TCK.

4.14.4.1 BYPASS

The BYPASS instruction selects the Bypass Register to connect between TDI and TDO. Selecting the Bypass Register, loads the 1-bit shift register with a logic low on the rising edge of TCK, following entry into the Capture-DR state. Holding TDI high, and completing an instruction scan cycle, executes the BYPASS instruction and sets up the Bypass Register. When the TDI input is unconnected, it is pulled to a logic high. If there was an open circuit fault condition in the serial board level test data path, the Bypass Register is selected following an instruction scan cycle. The BYPASS instruction has no effect on any other test data register.

4.14.4.2 SAMPLE/PRELOAD

The SAMPLE/PRELOAD instruction selects the Boundary Scan Register. This register connects between TDI and TDO in the Shift-DR state. The SAMPLE/PRELOAD instruction has no effect on any other test data register. Using SAMPLE/PRELOAD allows two functions to be performed:

- SAMPLE allows a snapshot of the normal MSM6100 device I/O operation to be taken and examined without causing interference. The snapshot is taken on the rising edge of TCK in the Capture-DR state. The data is viewed by shifting the snapshot data out via the TDO output during the Shift-DR state.
- PRELOAD allows data values to be loaded onto the Boundary Scan Register's latched parallel outputs, before selecting other Boundary Scan instructions. The shift register's value latches on the falling edge of TCK in the Update-DR state. For example, before selecting the EXTEST instruction, data loads onto the latched parallel outputs using PRELOAD. As soon as EXTEST becomes the current instruction, the pre-loaded data updates through the system output pins.

4.14.4.3 EXTEST

The EXTEST instruction selects only the Boundary Scan Register to connect TDI and TDO in the Shift-DR state. The EXTEST instruction has no effect on any other test data register. EXTEST allows circuits that are external to the MSM6100 device to be tested (typically the board interconnect). Boundary Scan Register cells at the output pins apply the test stimulus, while those at input pins capture test results. The EXTEST instruction also tests components that cannot do a Boundary Scan. Holding TDI low, and completing an instruction scan cycle, executes the EXTEST instruction and sets up the Boundary Scan Register.

Executing the EXTEST instruction (falling edge of TCK in the Update-DR state), updates the Boundary Scan Register's latched parallel output into the MSM6100 device and then out (via the normal system output pins). The update operation is done using the input and output Boundary Scan Register cells. Data is captured into the Boundary Scan Register on the TCK rising edge during the Capture-DR state. When the Shift-DR state is initiated, on the TCK rising edge, the data shifts from the Boundary Scan Register out via TDO.

4.14.4.4 IDCODE

This instruction selects only the Device Identification Register to connect between TDI and TDO. Selecting the IDCODE instruction, loads the manufacturer's identification code into the Device Identification Register on the rising edge of TCK, following entry into the Capture-DR state.

The IDCODE instruction loads into the IR parallel outputs during the Test Logic Reset state, on the falling edge of TCK. This loading occurs during normal TAP controller clocking. The IDCODE also loads into the IR parallel outputs on the falling edge of TRST_N.

4.14.4.5 JTAG Selection

The following diagrams show the pin configurations on the MSM6100 device for using the mode pins to select between the ARM TAP controller and the MSM6100 TAP controller. When configured to use the ARM interface, an ICD is expected to be connected and when selected for MSM6100 device, a BSDL test can be performed.

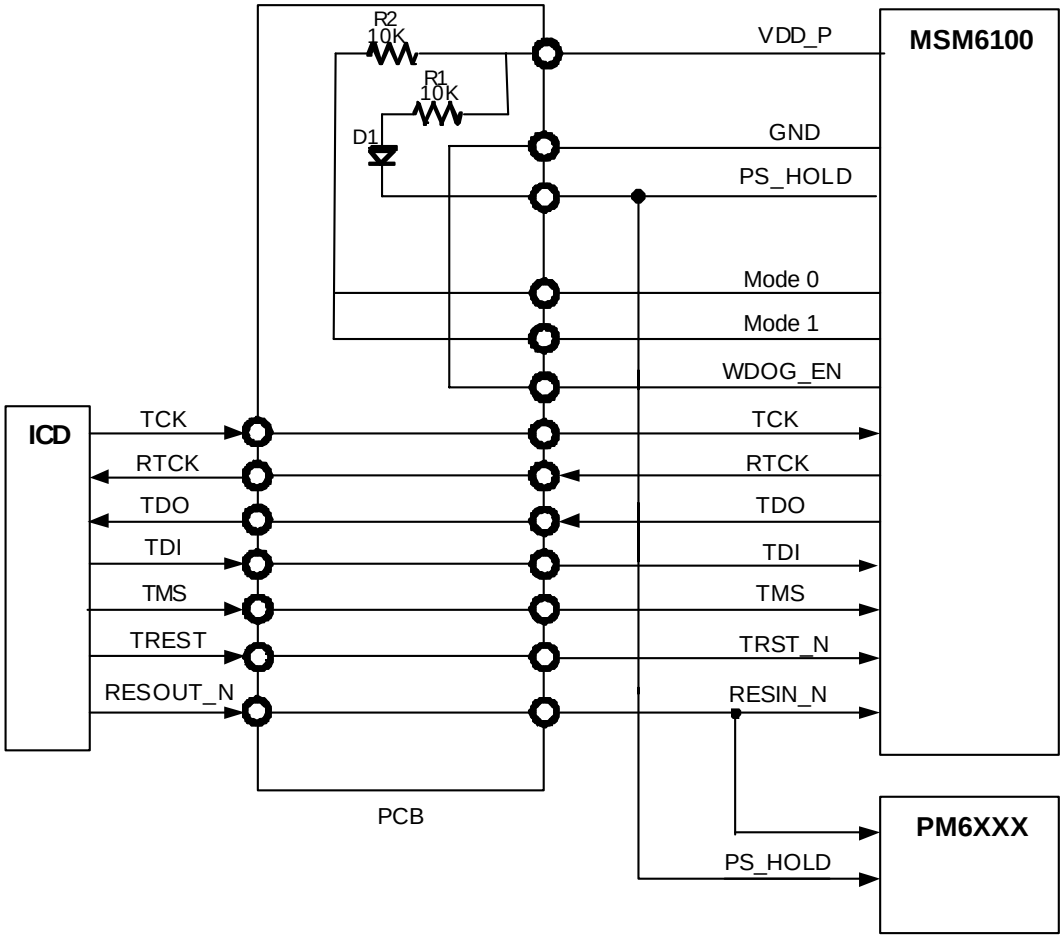


Figure 4-84 JTAG Connections for Using the MSM6100 BSDL Scan Chain WDOG Disabled

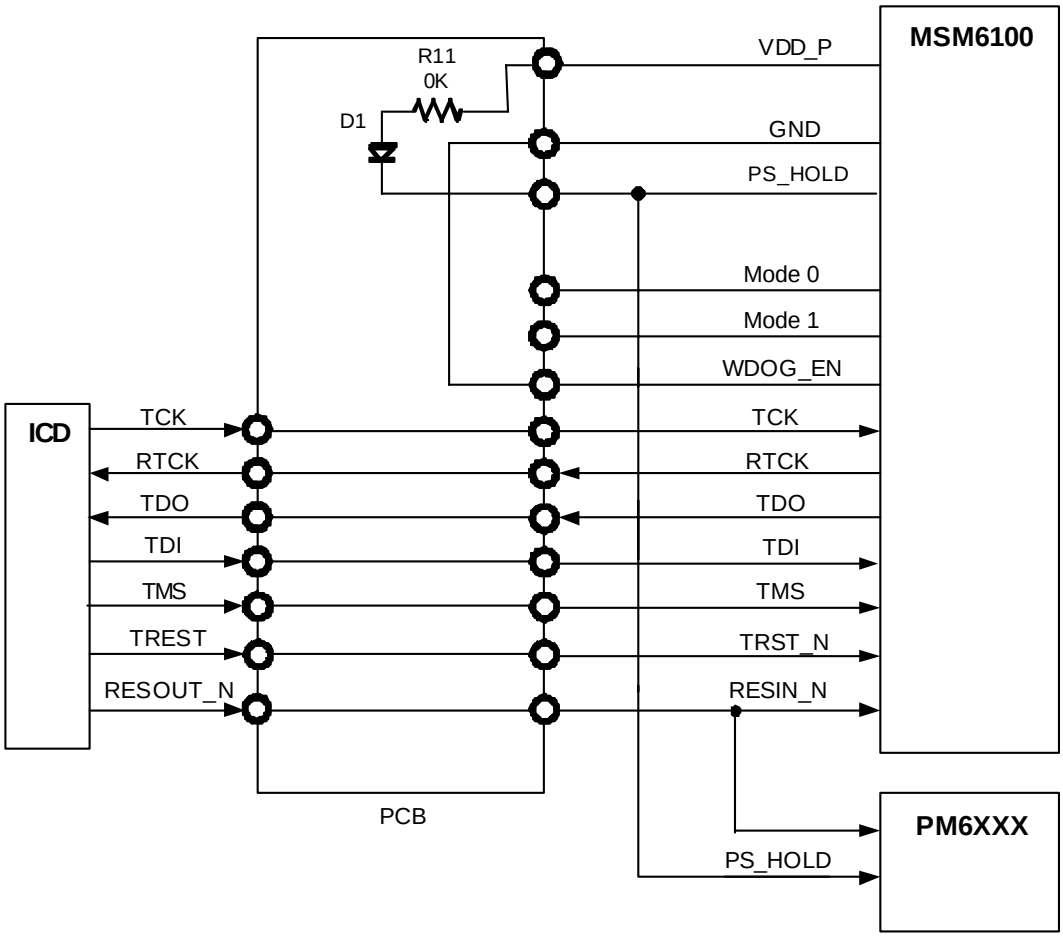


Figure 4-85 Connections for Using the MSM6100 ARM JTAG WDOG Disabled

4.15 Camera Interface

4.15.1 CAMIF Control

Timing Generation

The timing generation provides horizontal and vertical reference signals to cameras that operate in PPS mode. In all other modes, it requires these as reference signals inputs. The timing generation is implemented as a couple of counters, with count comparison circuitry that kicks off scheduled events, such as setting hsync high whenever the pixel counter equals 12, or setting vsync low whenever the line counter equals 187.

Synchronization

The camif_cntl module also accepts horizontal and vertical reference signals—whether they are physical or embedded in the data stream—from any cameras that provide them. The CAMIF locks onto these reference signals before beginning data collection. The CAMIF can also monitor these reference signals to ensure that synchronization is maintained.

Windowing

Most image sensor formats are slightly larger than a standard format. For example, the Sanyo sensor's format is 360x288, which is larger than CIF's 352x288. The CAMIF is programmable to allow data collection to begin and end at arbitrary points within the frame. In this example, the CAMIF could ignore four columns of pixels on the left and right sides of the image so that the DSP does not have to process the extra data.

Subsampling Support

The controller generates write strobes for storing the camera data in to the output buffer. When the CAMIF is set not to subsample, the CAMIF collects all data under the window described above. When the CAMIF is set to subsample, then it can selectively ignore data along all three axes.

To ignore pixels within a line, the CAMIF uses a 16-bit shift register to store the pixel culling pattern. A one at the shift register output indicates to the output buffer that it must store the current byte. A zero indicates to the output buffer that it should ignore the byte. The output is fed back into a register input so the pattern can repeat every 16 bytes. For RGB support, the shift register can be reduced to 12 bits. The shift register is reloaded with the pattern at the beginning of every line.

To ignore a line, the CAMIF uses a four-bit shift register. This register loads at the beginning of every frame and shifts after every line. A zero on the output tells the output buffer to ignore the current line. The output is fed back to the input, to cause the pattern to repeat every fourth line.

To ignore frames, the CAMIF uses a counter. When the count is zero, the output buffer stores the image. The counter increments every frame and resets when it reaches a programmable value.

4.15.2 Output Buffer

The output buffer provides a short-term storage area for camera data. Its purpose is to de-couple the camera's timing from the DSP's DMA interface and to account for any latency in the DSP's DMA servicing schedule.

The maximum rate at which the CAMIF can operate is dictated by speed at which the DMA will empty the output buffer. The CAMIF operates synchronously to the pclk provided by the image sensor, so this limitation places a similar limitation on the image sensors that can be supported for a given DSP clock frequency. The maximum supported pclk rate is one third that of the DSP clock frequency.

4.15.3 Supported Interface Detail

All of the cameras that CAMIF supports differ from one another. This section of the document is provided to show the MSM device's interconnection to the individual cameras, and to list some features from each of these cameras. This section is only included for reference purposes.

4.15.3.1 Sanyo iGT99263 CCD Camera Module

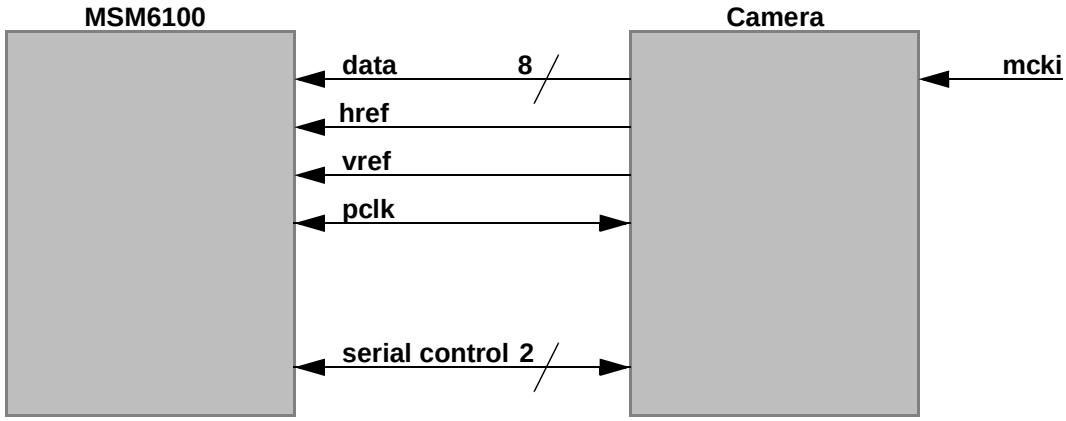


Figure 4-86 MSM6100 Device to Sanyo Camera Module Interconnect

Features and notes

- APS mode
  - Not a ITU-656 interface
  - href and vref required
  - pclk can be configured in either direction, but it is only supported as an input to the MSM
- Data formats
  - YUV (UYV) 4:2:2 and RBG (of unknown format)
  - 360 x 288 (CIF is 352 x 288)
  - 50 fps, 858 pclks per line, 315 lines per frame, 13.5 MHz pclk
- I^2C control bus

4.15.3.2 Hitachi HAM49001 CMOS Camera Module

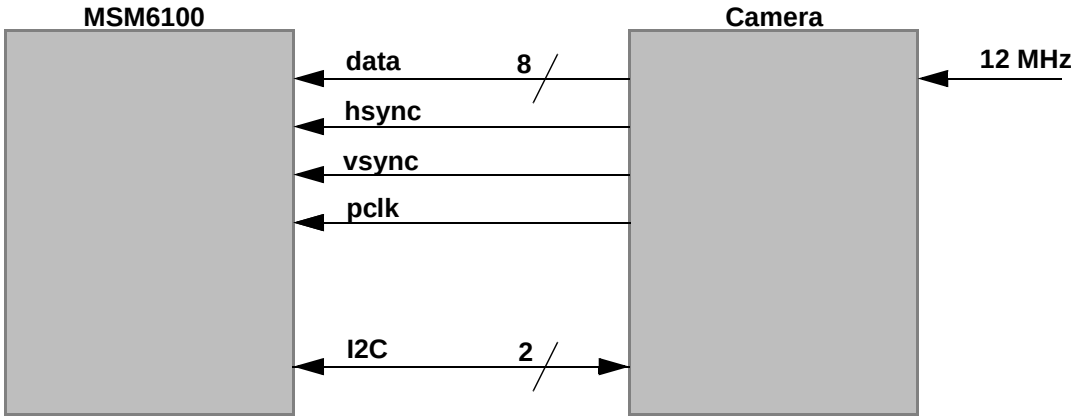


Figure 4-87 MSM6100 Device to Hitachi Camera Module Interconnect

Features and notes

- APS mode
  - Not a ITU-656 interface
  - href and vref required
- Data formats
  - YUV (UYV) 4:2:2
  - RBG (of unknown format)
- CIF or QCIF
- 30, 15, 7.5, 3.75 fps
- I^2C control bus

4.15.3.3 Matsushita (Panasonic)

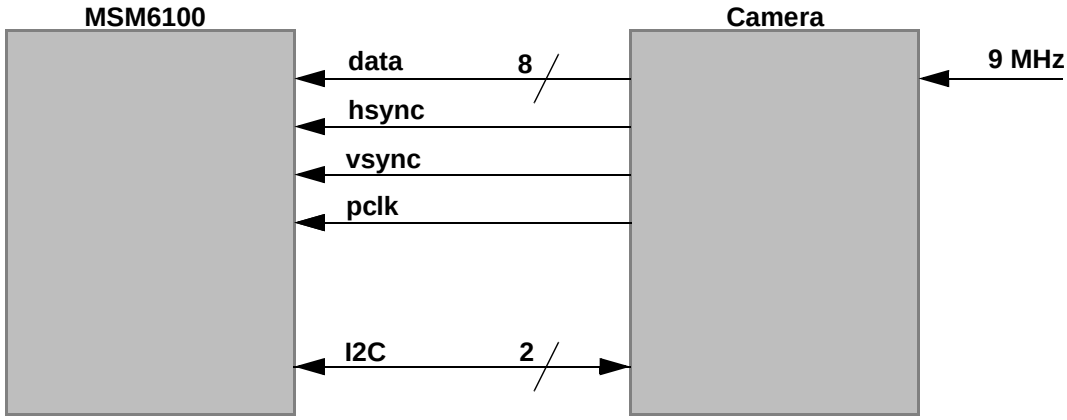


Figure 4-88 MSM6100 Device to Matsushita Camera Module Interconnect

Features and notes

- APS mode
  - Not a ITU-656 interface
  - hsync and vsync required
- Data formats
  - YUV
  - weird ordering (Y1, U1, Y2, V1) ⇒ need for endian support
- CIF
- 15 fps (30 fps possible, according to the datasheet)
- I^2C control bus

4.15.3.4 Toshiba TC90A70F CMOS Camera Module

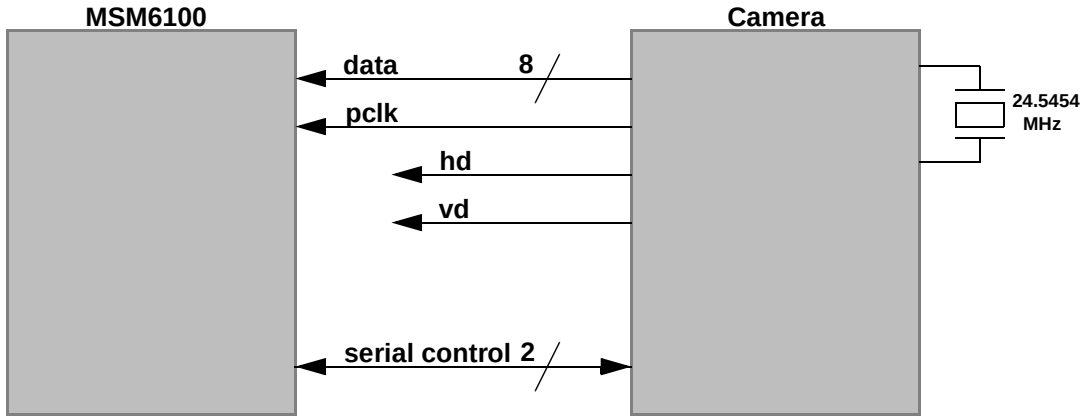


Figure 4-89 MSM6100 Device to Toshiba Camera Module Interconnect

Features and notes

- ELS or APS mode
  - ITU-656-like interface
  - hd and vd are optional
  - undocumented SOF, SOL, EOL, EOF codes
- YUV 4:2:2
- 30, 15, 7.5, 3.75 fps
- VGA, QVGA, QQVGA, CIF, QCIF
- Serial control via I^2C type bus

4.15.3.5 Sharp LZ0P3800 CMOS Image Sensor

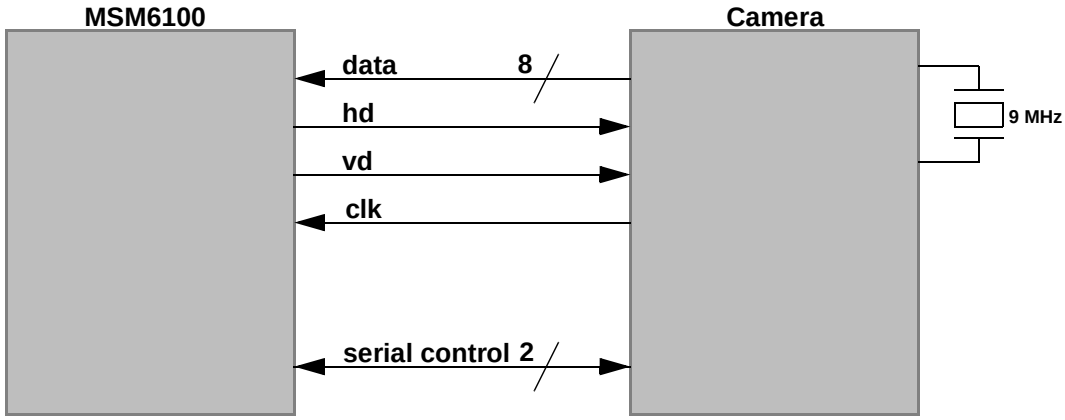


Figure 4-90 MSM6100 Device to Sharp Image Sensor Interconnect

Features and notes:

- PPS mode
  - Not ITU656
  - Timing generation
    - Camera has crystal and provides clock to MSM
    - MSM divides clock and provides HD and VD timing signals
    - Camera spits data out and MSM knows when to capture based upon HD and VD
- Serial Control
  - Not I^2C
  - Must be synchronized to VD as there is a lock-out time where you cannot control camera
- Data format
  - Raw RG GB (Bayer matrix)
  - Requires post processing

4.15.3.6 OmniVision OV7630 and OV6630 CMOS Cameras

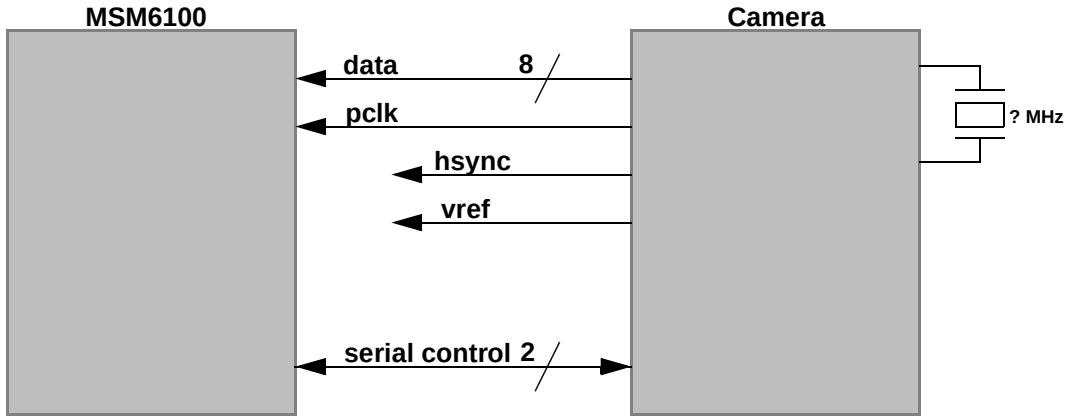


Figure 4-91 MSM to OmniVision Camera Interconnect

Features and notes

- ITU-656-like interface
  - href and vsync are optional?
- Data formats
  - CbYCr 4:2:2
  - Raw RBG (sensor format)
- Serial control via I^2C-type bus

4.15.3.7 IC Media ICM102A CIF CMOS Camera

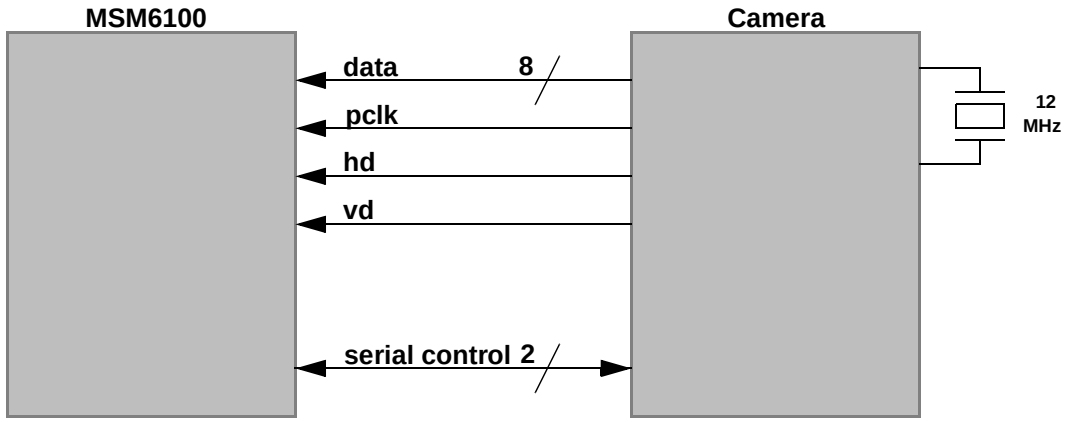


Figure 4-92 MSM6100 Device to IC Media Camera Interconnect

Features and notes

- Not a ITU-656-like interface
- RGB Bayer matrix output
- 30, 15, 10, 6, 5, 3, 2, 1 fps
- CIF
- Serial control via I^2C-type bus

4.15.3.8 Sharp CMOS Camera Module

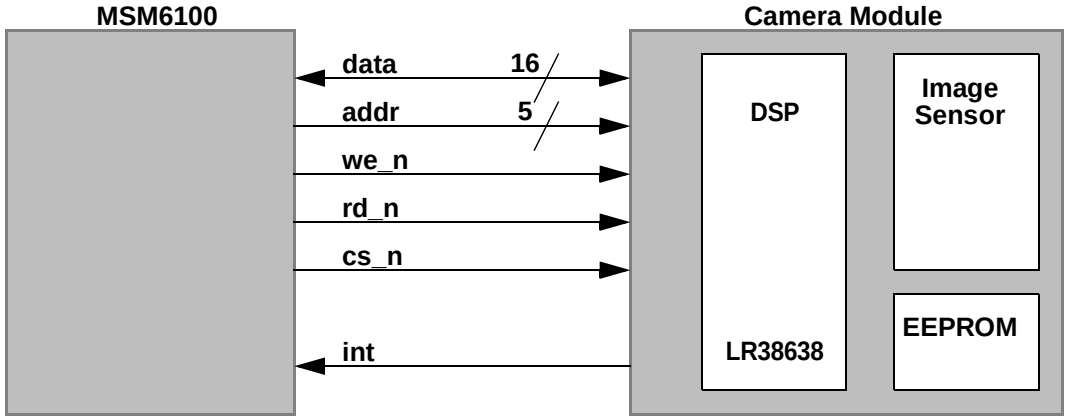


Figure 4-93 MSM6100 Device to Sharp Camera Module Interconnect

Features and notes

- SRAM style interface
  - Beyond scope of CAMIF
  - Included here for completeness
- Data format:
  - Image sensor is RG GB.
  - DSP does some type of formatting
- Approximate 1 ms interrupt interval
  - QCIF (176 x 144)
  - 15 fps
  - Average of an interrupt every other line.

4.15.3.9 Atmel AT76C401 CMOS Imager

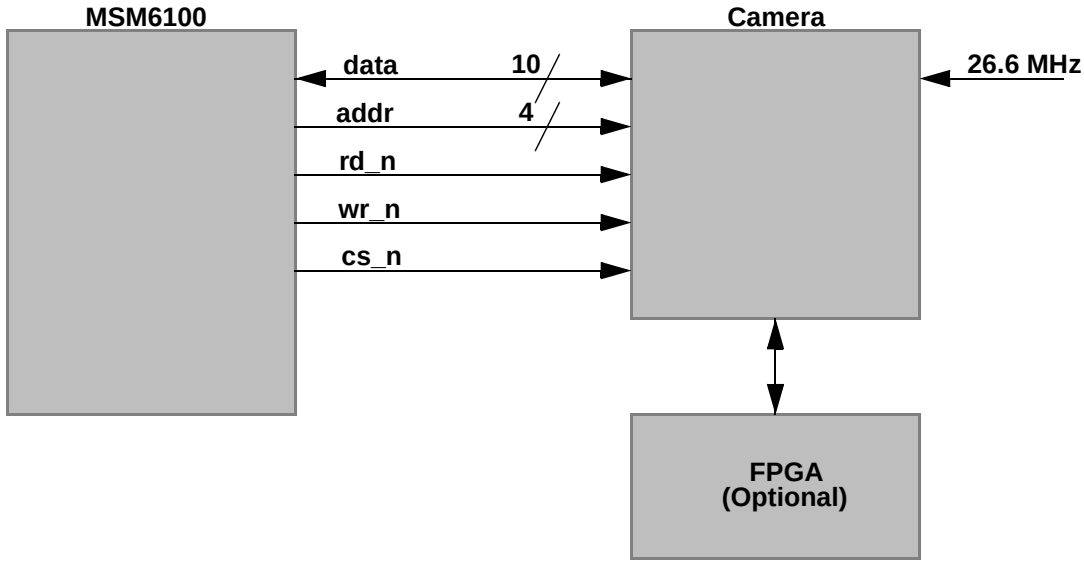


Figure 4-94 MSM6100 Device to Atmel Imager

Features and notes

- SRAM style interface
  - Beyond scope of CAMIF
  - Included here for completeness
- Data format is RGB
- Interrupt interval unknown

Optional FPGA for non-standard formats like QCIF

5 Mechanical Dimensions

5.1 341-Ball CSP Package Outline

The 341-Ball CSP Package Outline is specified in QUALCOMM document *BGA User Guide*, 80-V2560-1.

5.2 341-Ball CSP Land Pattern

The 341-Ball CSP Land Pattern is specified in QUALCOMM document *BGA User Guide*, 80-V2560-1.

5.3 Part Marking

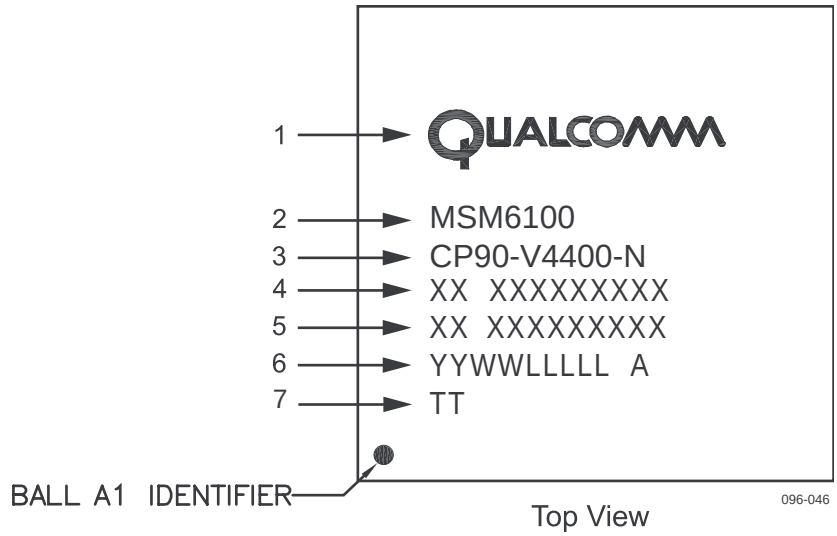


Figure 5-1 Part Marking

Table 5-1 Marking Descriptions

Line	Description
1	QUALCOMM logo
2	QUALCOMM product number
3	QUALCOMM part number N = device version
4	Traceability Number
5	Traceability Number
6	Assembly Encapsulation Date and Site Codes: YYWW = assembly date code LLLLL = assembly lot code A = assembly site code
7	Test Codes: TT = test location