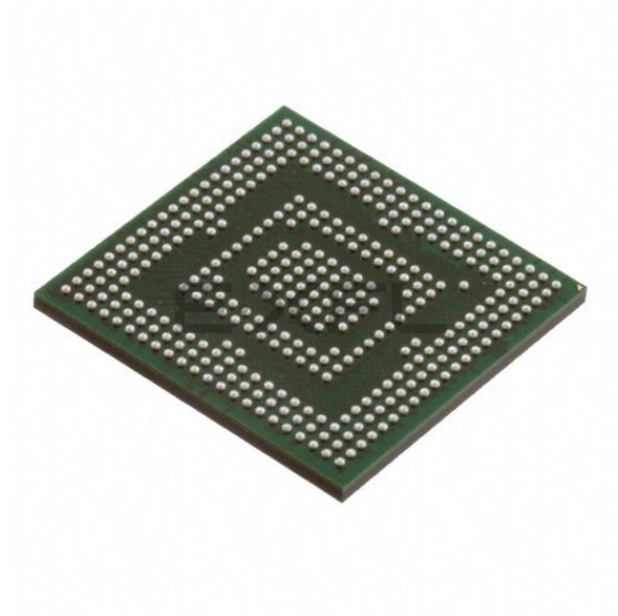


Welcome to E-XFL.COM

Understanding Embedded - DSP (Digital Signal Processors)

Embedded - DSP (Digital Signal Processors) are specialized microprocessors designed to perform complex mathematical computations on digital signals in real-time. Unlike general-purpose processors, DSPs are optimized for high-speed numeric processing tasks, making them ideal for applications that require efficient and precise manipulation of digital data. These processors are fundamental in converting and processing signals in various forms, including audio, video, and communication signals, ensuring that data is accurately interpreted and utilized in embedded systems.

Applications of Embedded - DSP (Digital Signal Processors)

Details	
Product Status	Active
Type	Dual Core
Interface	CAN, EBI/EMI, Ethernet, I²C, SPI, SPORT, UART/USART, USB OTG
Clock Rate	400MHz
Non-Volatile Memory	ROM (64kB)
On-Chip RAM	552kB
Voltage - I/O	1.8V, 3.3V
Voltage - Core	1.25V
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	349-LFBGA, CSPBGA
Supplier Device Package	349-CSPBGA (19x19)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/adsp-bf606kbcz-4

GENERAL DESCRIPTION

The ADSP-BF60x processors are members of the Blackfin family of products, incorporating the Analog Devices/Intel Micro Signal Architecture (MSA). Blackfin processors combine a dual-MAC state-of-the-art signal processing engine, the advantages of a clean, orthogonal RISC-like microprocessor instruction set, and single-instruction, multiple-data (SIMD) multimedia capabilities into a single instruction-set architecture.

The processors offer performance up to 500 MHz, as well as low static power consumption. Produced with a low-power and low-voltage design methodology, they provide world-class power management and performance.

By integrating a rich set of industry-leading system peripherals and memory (shown in Table 1), Blackfin processors are the platform of choice for next-generation applications that require RISC-like programmability, multimedia support, and leading-edge signal processing in one integrated package. These applications span a wide array of markets, from automotive systems to embedded industrial, instrumentation and power/motor control applications.

Table 1. Processor Comparison

Processor Feature	ADSP-BF606	ADSP-BF607	ADSP-BF608	ADSP-BF609
Up/Down/Rotary Counters	1			
Timer/Counters with PWM	8			
3-Phase PWM Units (4-pair)	2			
SPORTs	3			
SPIs	2			
USB OTG	1			
Parallel Peripheral Interface	3			
Removable Storage Interface	1			
CAN	1			
TWI	2			
UART	2			
ADC Control Module (ACM)	1			
Link Ports	4			
Ethernet MAC (IEEE 1588)	2			
Pixel Compositor (PIXC)	No		1	1
Pipelined Vision Processor (PVP) Video Resolution[1]	No		VGA	HD
Maximum PVP Line Buffer Size	N/A		640	1280
GPIOs	112			

Table 1. Processor Comparison (Continued)

Processor Feature		ADSP-BF606	ADSP-BF607	ADSP-BF608	ADSP-BF609
Memory (bytes, per core)	L1 Instruction SRAM	64K			
	L1 Instruction SRAM/Cache	16K			
	L1 Data SRAM	32K			
	L1 Data SRAM/Cache	32K			
	L1 Scratchpad	4K			
	L2 Data SRAM	128K	256K		
	L2 Boot ROM	32K			
Maximum Speed Grade (MHz)[2]		400	500		
Maximum SYSCLK (MHz)		250			
Package Options		349-Ball CSP_BGA			

[1] VGA is 640 × 480 pixels per frame. HD is 1280 × 960 pixels per frame.
[2] Maximum speed grade is not available with every possible SYSCLK selection.

BLACKFIN PROCESSOR CORE

As shown in Figure 1, the processor integrates two Blackfin processor cores. Each core, shown in Figure 2, contains two 16-bit multipliers, two 40-bit accumulators, two 40-bit ALUs, four video ALUs, and a 40-bit shifter. The computation units process 8-, 16-, or 32-bit data from the register file.

The compute register file contains eight 32-bit registers. When performing compute operations on 16-bit operand data, the register file operates as 16 independent 16-bit registers. All operands for compute operations come from the multiported register file and instruction constant fields.

Each MAC can perform a 16-bit by 16-bit multiply in each cycle, accumulating the results into the 40-bit accumulators. Signed and unsigned formats, rounding, and saturation are supported.

The ALUs perform a traditional set of arithmetic and logical operations on 16-bit or 32-bit data. In addition, many special instructions are included to accelerate various signal processing tasks. These include bit operations such as field extract and population count, modulo 2^{32} multiply, divide primitives, saturation and rounding, and sign/exponent detection. The set of video instructions include byte alignment and packing operations, 16-bit and 8-bit adds with clipping, 8-bit average operations, and 8-bit subtract/absolute value/accumulate (SAA) operations. Also provided are the compare/select and vector search instructions.

For certain instructions, two 16-bit ALU operations can be performed simultaneously on register pairs (a 16-bit high half and 16-bit low half of a compute register). If the second ALU is used, quad 16-bit operations are possible.

a very small final memory size. The instruction set also provides fully featured multifunction instructions that allow the programmer to use many of the processor core resources in a single instruction. Coupled with many features more often seen on microcontrollers, this instruction set is very efficient when compiling C and C++ source code. In addition, the architecture supports both user (algorithm/application code) and supervisor (O/S kernel, device drivers, debuggers, ISRs) modes of operation, allowing multiple levels of access to core processor resources.

The assembly language, which takes advantage of the processor's unique architecture, offers the following advantages:

- Seamlessly integrated DSP/MCU features are optimized for both 8-bit and 16-bit operations.
- A multi-issue load/store modified-Harvard architecture, which supports two 16-bit MAC or four 8-bit ALU + two load/store + two pointer updates per cycle.
- All registers, I/O, and memory are mapped into a unified 4G byte memory space, providing a simplified programming model.
- Control of all asynchronous and synchronous events to the processor is handled by two subsystems: the Core Event Controller (CEC) and the System Event Controller (SEC).
- Microcontroller features, such as arbitrary bit and bit-field manipulation, insertion, and extraction; integer operations on 8-, 16-, and 32-bit data-types; and separate user and supervisor stack pointers.
- Code density enhancements, which include intermixing of 16-bit and 32-bit instructions (no mode switching, no code segregation). Frequently used instructions are encoded in 16 bits.

PROCESSOR INFRASTRUCTURE

The following sections provide information on the primary infrastructure components of the ADSP-BF609 processor.

DMA Controllers

The processor uses Direct Memory Access (DMA) to transfer data within memory spaces or between a memory space and a peripheral. The processor can specify data transfer operations and return to normal processing while the fully integrated DMA controller carries out the data transfers independent of processor activity.

DMA transfers can occur between memory and a peripheral or between one memory and another memory. Each Memory-to-memory DMA stream uses two channels, where one channel is the source channel, and the second is the destination channel.

All DMAs can transport data to and from all on-chip and off-chip memories. Programs can use two types of DMA transfers, descriptor-based or register-based. Register-based DMA allows the processor to directly program DMA control registers to initiate a DMA transfer. On completion, the control registers may be automatically updated with their original setup values for continuous transfer. Descriptor-based DMA transfers require a set of parameters stored within memory to initiate a DMA sequence. Descriptor-based DMA transfers allow multiple DMA sequences to be chained together and a DMA channel can be programmed to automatically set up and start another DMA transfer after the current sequence completes.

The DMA controller supports the following DMA operations.

- A single linear buffer that stops on completion.
- A linear buffer with negative, positive or zero stride length.
- A circular, auto-refreshing buffer that interrupts when each buffer becomes full.
- A similar buffer that interrupts on fractional buffers (for example, 1/2, 1/4).
- 1D DMA – uses a set of identical ping-pong buffers defined by a linked ring of two-word descriptor sets, each containing a link pointer and an address.
- 1D DMA – uses a linked list of 4 word descriptor sets containing a link pointer, an address, a length, and a configuration.
- 2D DMA – uses an array of one-word descriptor sets, specifying only the base DMA address.
- 2D DMA – uses a linked list of multi-word descriptor sets, specifying everything.

CRC Protection

The two CRC protection modules allow system software to periodically calculate the signature of code and/or data in memory, the content of memory-mapped registers, or communication message objects. Dedicated hardware circuitry compares the signature with pre calculated values and triggers appropriate fault events.

For example, every 100 ms the system software might initiate the signature calculation of the entire memory contents and compare these contents with expected, pre calculated values. If a mismatch occurs, a fault condition can be generated (via the processor core or the trigger routing unit).

The CRC is a hardware module based on a CRC32 engine that computes the CRC value of the 32-bit data words presented to it. Data is provided by the source channel of the memory-to-memory DMA (in memory scan mode) and is optionally forwarded to the destination channel (memory transfer mode). The main features of the CRC peripheral are:

- Memory scan mode
- Memory transfer mode
- Data verify mode
- Data fill mode
- User-programmable CRC32 polynomial
- Bit/byte mirroring option (endianness)
- Fault/error interrupt mechanisms
- 1D and 2D fill block to initialize array with constants.
- 32-bit CRC signature of a block of a memory or MMR block.

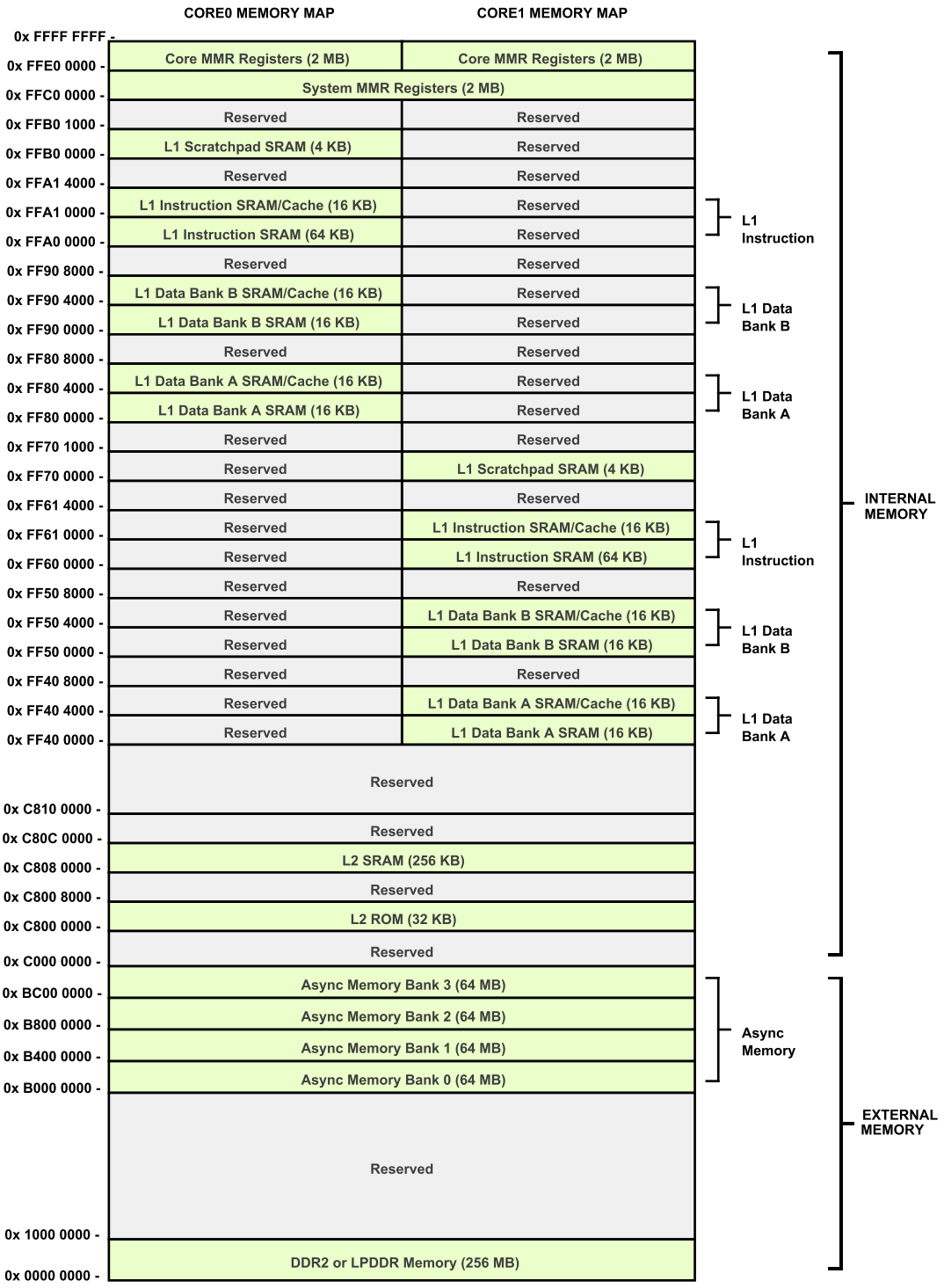


Figure 4. ADSP-BF607/ADSP-BF608/ADSP-BF609 Internal/External Memory Map

- Support for remote frames.
- Active or passive network support.
- CAN wakeup from hibernation mode (lowest static power consumption mode).
- Interrupts, including: TX complete, RX complete, error and global.

An additional crystal is not required to supply the CAN clock, as the CAN clock is derived from a system clock through a programmable divider.

10/100 Ethernet MAC

The processor can directly connect to a network by way of an embedded fast Ethernet media access controller (MAC) that supports both 10-BaseT (10M bits/sec) and 100-BaseT (100M bits/sec) operation. The 10/100 Ethernet MAC peripheral on the processor is fully compliant to the IEEE 802.3-2002 standard and it provides programmable features designed to minimize supervision, bus use, or message processing by the rest of the processor system.

Some standard features are:

- Support and RMII protocols for external PHYs
- Full duplex and half duplex modes
- Media access management (in half-duplex operation)
- Flow control
- Station management: generation of MDC/MDIO frames for read-write access to PHY registers

Some advanced features are:

- Automatic checksum computation of IP header and IP payload fields of RX frames
- Independent 32-bit descriptor-driven receive and transmit DMA channels
- Frame status delivery to memory through DMA, including frame completion semaphores for efficient buffer queue management in software
- TX DMA support for separate descriptors for MAC header and payload to eliminate buffer copy operations
- Convenient frame alignment modes
- 47 MAC management statistics counters with selectable clear-on-read behavior and programmable interrupts on half maximum value
- Advanced power management
- Magic packet detection and wakeup frame filtering
- Support for 802.3Q tagged VLAN frames
- Programmable MDC clock rate and preamble suppression

IEEE 1588 Support

The IEEE 1588 standard is a precision clock synchronization protocol for networked measurement and control systems. The processor includes hardware support for IEEE 1588 with an integrated precision time protocol synchronization engine (PTP_TSYNC). This engine provides hardware assisted time stamping to improve the accuracy of clock synchronization between PTP nodes. The main features of the engine are:

- Support for both IEEE 1588-2002 and IEEE 1588-2008 protocol standards
- Hardware assisted time stamping capable of up to 12.5 ns resolution
- Lock adjustment
- Automatic detection of IPv4 and IPv6 packets, as well as PTP messages
- Multiple input clock sources (SCLK0, RMII clock, external clock)
- Programmable pulse per second (PPS) output
- Auxiliary snapshot to time stamp external events

USB 2.0 On-the-Go Dual-Role Device Controller

The USB 2.0 OTG dual-role device controller provides a low-cost connectivity solution for the growing adoption of this bus standard in industrial applications, as well as consumer mobile devices such as cell phones, digital still cameras, and MP3 players. The USB 2.0 controller allows these devices to transfer data using a point-to-point USB connection without the need for a PC host. The module can operate in a traditional USB peripheral-only mode as well as the host mode presented in the On-the-Go (OTG) supplement to the USB 2.0 specification.

The USB clock (USB_CLKIN) is provided through a dedicated external crystal or crystal oscillator.

The USB On-the-Go dual-role device controller includes a Phase Locked Loop with programmable multipliers to generate the necessary internal clocking frequency for USB.

POWER AND CLOCK MANAGEMENT

The processor provides four operating modes, each with a different performance/power profile. When configured for a 0 V internal supply voltage (V_{DD_INT}), the processor enters the hibernate state. Control of clocking to each of the processor peripherals also reduces power consumption. See Table 5 for a summary of the power settings for each mode.

Crystal Oscillator (SYS_XTAL)

The processor can be clocked by an external crystal (Figure 6), a sine wave input, or a buffered, shaped clock derived from an external clock oscillator. If an external clock is used, it should be a TTL compatible signal and must not be halted, changed, or operated below the specified frequency during normal operation. This signal is connected to the processor's SYS_CLKIN pin. When an external clock is used, the SYS_XTAL pin must be left unconnected. Alternatively, because the processor includes an on-chip oscillator circuit, an external crystal may be used.

For fundamental frequency operation, use the circuit shown in Figure 6. A parallel-resonant, fundamental frequency, microprocessor grade crystal is connected across the SYS_CLKIN and XTAL pins. The on-chip resistance between SYS_CLKIN and the XTAL pin is in the 500 kΩ range. Further parallel resistors are typically not recommended.

Table 3. Clock Dividers

Clock Source	Divider
CCLK (core clock)	By 4
SYSCLK (System clock)	By 2
SCLK0 (system clock for PVP, all peripherals not covered by SCLK1)	None
SCLK1 (system clock for SPORTS, SPI, ACM)	None
DCLK (LPDDR/DDR2 clock)	By 2
OCLK (output clock)	Programmable
CLKBUF	None, direct from SYS_CLKIN

Power Management

As shown in Table 4, the processor supports five different power domains, which maximizes flexibility while maintaining compliance with industry standards and conventions. There are no sequencing requirements for the various power domains, but all domains must be powered according to the appropriate Specifications table for processor operating conditions; even if the feature/peripheral is not used.

Table 4. Power Domains

Power Domain	V_{DD} Range
All internal logic	V_{DD_INT}
DDR2/LPDDR	V_{DD_DMC}
USB	V_{DD_USB}
Thermal diode	V_{DD_TD}
All other I/O (includes SYS, JTAG, and Ports pins)	V_{DD_EXT}

The dynamic power management feature of the processor allows the processor's core clock frequency (f_{CCLK}) to be dynamically controlled.

The power dissipated by a processor is largely a function of its clock frequency and the square of the operating voltage. For example, reducing the clock frequency by 25% results in a 25% reduction in dynamic power dissipation.

Full-On Operating Mode—Maximum Performance

In the full-on mode, the PLL is enabled and is not bypassed, providing capability for maximum operational frequency. This is the power-up default execution state in which maximum performance can be achieved. The processor cores and all enabled peripherals run at full speed.

Active Operating Mode—Moderate Dynamic Power Savings

In the active mode, the PLL is enabled but bypassed. Because the PLL is bypassed, the processor's core clocks and system clocks run at the input clock (SYS_CLKIN) frequency. DMA access is available to appropriately configured L1 memories.

For more information about PLL controls, see the "Dynamic Power Management" chapter in the *ADSP-BF60x Blackfin Processor Hardware Reference.*

See Table 5 for a summary of the power settings for each mode.

Table 5. Power Settings

Mode/State	PLL	PLL Bypassed	f_{CCLK}	f_{SYSCLK}, f_{DCLK}, f_{SCLK0}, f_{SCLK1}	Core Power
Full On	Enabled	No	Enabled	Enabled	On
Active	Enabled/ Disabled	Yes	Enabled	Enabled	On
Deep Sleep	Disabled	—	Disabled	Disabled	On
Hibernate	Disabled	—	Disabled	Disabled	Off

Deep Sleep Operating Mode—Maximum Dynamic Power Savings

The deep sleep mode maximizes dynamic power savings by disabling the clocks to the processor core and to all synchronous peripherals. Asynchronous peripherals may still be running but cannot access internal resources or external memory.

Hibernate State—Maximum Static Power Savings

The hibernate state maximizes static power savings by disabling the voltage and clocks to the processor cores and to all of the peripherals. This setting signals the external voltage regulator supplying the V_{DD_INT} pins to shut off using the SYS_EXTWAKE signal, which provides the lowest static power dissipation. Any critical information stored internally (for example, memory contents, register contents, and other information) must be written to a non-volatile storage device prior to removing power if the processor state is to be preserved.

Since the V_{DD_EXT} pins can still be supplied in this mode, all of the external pins three-state, unless otherwise specified. This allows other devices that may be connected to the processor to still have power applied without drawing unwanted current.

Reset Control Unit

Reset is the initial state of the whole processor or one of the cores and is the result of a hardware or software triggered event. In this state, all control registers are set to their default values and functional units are idle. Exiting a full system reset starts with Core-0 only being ready to boot. Exiting a Core-n only reset starts with this Core-n being ready to boot.

The Reset Control Unit (RCU) controls how all the functional units enter and exit reset. Differences in functional requirements and clocking constraints define how reset signals are generated. Programs must guarantee that none of the reset functions puts the system into an undefined state or causes resources to stall. This is particularly important when only one of the cores is reset (programs must ensure that there is no pending system activity involving the core that is being reset).

349-BALL CSP_BGA SIGNAL DESCRIPTIONS

The processors' pin definitions are shown in the table. The columns in this table provide the following information:

- Signal Name: The Signal Name column in the table includes the Signal Name for every pin.
- Description: The Description column in the table provides a verbose (descriptive) name for the signal.
- Port: The General-Purpose I/O Port column in the table shows whether or not the signal is multiplexed with other signals on a general-purpose I/O port pin.
- Pin Name: The Pin Name column in the table identifies the name of the package pin (at power-on reset) on which the signal is located (if a single function pin) or is multiplexed (if a general-purpose I/O pin).

Table 7. ADSP-BF60x 349-Ball CSP_BGA Signal Descriptions

Signal Name	Description	Port	Pin Name
ACM0_A0	ACM0 Address 0	F	PF_14
ACM0_A1	ACM0 Address 1	F	PF_15
ACM0_A2	ACM0 Address 2	F	PF_12
ACM0_A3	ACM0 Address 3	F	PF_13
ACM0_A4	ACM0 Address 4	F	PF_10
ACM0_CLK	ACM0 Clock	E	PE_04
ACM0_FS	ACM0 Frame Sync	E	PE_03
ACM0_T0	ACM0 External Trigger 0	E	PE_08
ACM0_T1	ACM0 External Trigger 1	G	PG_05
CAN0_RX	CAN0 Receive	G	PG_04
CAN0_TX	CAN0 Transmit	G	PG_01
CNT0_DG	CNT0 Count Down and Gate	G	PG_12
CNT0_UD	CNT0 Count Up and Direction	G	PG_11
CNT0_ZM	CNT0 Count Zero Marker	G	PG_07
DMC0_A00	DMC Address 0	Not Muxed	DMC0_A00
DMC0_A01	DMC Address 1	Not Muxed	DMC0_A01
DMC0_A02	DMC Address 2	Not Muxed	DMC0_A02
DMC0_A03	DMC Address 3	Not Muxed	DMC0_A03
DMC0_A04	DMC Address 4	Not Muxed	DMC0_A04
DMC0_A05	DMC Address 5	Not Muxed	DMC0_A05
DMC0_A06	DMC Address 6	Not Muxed	DMC0_A06
DMC0_A07	DMC Address 7	Not Muxed	DMC0_A07
DMC0_A08	DMC Address 8	Not Muxed	DMC0_A08
DMC0_A09	DMC Address 9	Not Muxed	DMC0_A09
DMC0_A10	DMC Address 10	Not Muxed	DMC0_A10
DMC0_A11	DMC Address 11	Not Muxed	DMC0_A11
DMC0_A12	DMC Address 12	Not Muxed	DMC0_A12
DMC0_A13	DMC Address 13	Not Muxed	DMC0_A13
DMC0_BA0	DMC Bank Address Input 0	Not Muxed	DMC0_BA0
DMC0_BA1	DMC Bank Address Input 1	Not Muxed	DMC0_BA1
DMC0_BA2	DMC Bank Address Input 2	Not Muxed	DMC0_BA2
$\overline{\text{DMC0\_CAS}}$	DMC Column Address Strobe	Not Muxed	$\overline{\text{DMC0\_CAS}}$
DMC0_CK	DMC Clock	Not Muxed	DMC0_CK
DMC0_CKE	DMC Clock Enable	Not Muxed	DMC0_CKE
$\overline{\text{DMC0\_CK}}$	DMC Clock (complement)	Not Muxed	$\overline{\text{DMC0\_CK}}$
$\overline{\text{DMC0\_CS0}}$	DMC Chip Select 0	Not Muxed	$\overline{\text{DMC0\_CS0}}$

Table 14. Signal Multiplexing for Port G

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function Input Tap
PG_00	ETH1_RXD0	PWM1_BH	RSI0_D2	
PG_01	SPT2_AFS	TM0_TMR2	CAN0_TX	
PG_02	ETH1_TXD1	PWM1_AL	RSI0_D1	
PG_03	ETH1_TXD0	PWM1_AH	RSI0_D0	
PG_04	SPT2_ACLK	TM0_TMR1	CAN0_RX	TM0_ACI2
PG_05	ETH1_TXEN	RSI0_CMD	PWM1_SYNC	ACM0_T1
PG_06	ETH1_REFCLK	RSI0_CLK	SPT2_BTDV	$\overline{\text{PWM1\_TRIP0}}$
PG_07	SPT2_BFS	TM0_TMR5		CNT0_ZM
PG_08	SPT2_AD1	TM0_TMR3		$\overline{\text{PWM1\_TRIP1}}$
PG_09	SPT2_AD0	TM0_TMR4		
PG_10	$\overline{\text{UART1\_RTS}}$	SPT2_BCLK		
PG_11	SPT2_BD1	TM0_TMR6		CNT0_UD
PG_12	SPT2_BD0	TM0_TMR7		CNT0_DG
PG_13	$\overline{\text{UART1\_CTS}}$			TM0_CLK
PG_14	$\overline{\text{UART1\_RX}}$	$\overline{\text{SYS\_IDLE1}}$		TM0_ACI1
PG_15	$\overline{\text{UART1\_TX}}$	$\overline{\text{SYS\_IDLE0}}$	$\overline{\text{SYS\_SLEEP}}$	TM0_ACI4

Table 15. ADSP-BF60x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
DMC0_A11	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Address 11. Notes: No notes.
DMC0_A12	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Address 12. Notes: No notes.
DMC0_A13	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Address 13. Notes: No notes.
DMC0_BA0	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Bank Address Input 0. Notes: No notes.
DMC0_BA1	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Bank Address Input 1. Notes: No notes.
DMC0_BA2	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Bank Address Input 2. Notes: For LPDDR, leave unconnected.
$\overline{\text{DMC0\_CAS}}$	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Column Address Strobe. Notes: No notes.
DMC0_CK	I/O	C	none	none	L	none	L	VDD_DMC	Desc: DMC0 Clock. Notes: No notes.
$\overline{\text{DMC0\_CK}}$	I/O	C	none	none	L	none	L	VDD_DMC	Desc: DMC0 Clock (complement). Notes: No notes.
DMC0_CKE	I/O	B	none	none	L	none	L	VDD_DMC	Desc: DMC0 Clock enable. Notes: No notes.
$\overline{\text{DMC0\_CS0}}$	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Chip Select 0. Notes: No notes.
DMC0_DQ00	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 0. Notes: No notes.
DMC0_DQ01	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 1. Notes: No notes.
DMC0_DQ02	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 2. Notes: No notes.
DMC0_DQ03	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 3. Notes: No notes.
DMC0_DQ04	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 4. Notes: No notes.
DMC0_DQ05	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 5. Notes: No notes.
DMC0_DQ06	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 6. Notes: No notes.
DMC0_DQ07	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 7. Notes: No notes.
DMC0_DQ08	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 8. Notes: No notes.
DMC0_DQ09	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 9. Notes: No notes.
DMC0_DQ10	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 10. Notes: No notes.
DMC0_DQ11	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 11. Notes: No notes.
DMC0_DQ12	I/O	B	none	none	none	none	none	VDD_DMC	Desc: DMC0 Data 12. Notes: No notes.

Table 15. ADSP-BF60x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Hiber Term	Hiber Drive	Power Domain	Description and Notes
SMC0_D06	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 6. Notes: No notes.
SMC0_D07	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 7. Notes: No notes.
SMC0_D08	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 8. Notes: No notes.
SMC0_D09	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 9. Notes: No notes.
SMC0_D10	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 10. Notes: No notes.
SMC0_D11	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 11. Notes: No notes.
SMC0_D12	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 12. Notes: No notes.
SMC0_D13	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 13. Notes: No notes.
SMC0_D14	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 14. Notes: No notes.
SMC0_D15	I/O	A	wk	wk	none	wk	none	VDD_EXT	Desc: SMC0 Data 15. Notes: No notes.
SYS_BMODE0	I/O	na	none	none	none	none	none	VDD_EXT	Desc: SYS Boot Mode Control 0. Notes: No notes.
SYS_BMODE1	I/O	na	none	none	none	none	none	VDD_EXT	Desc: SYS Boot Mode Control 1. Notes: No notes.
SYS_BMODE2	I/O	na	none	none	none	none	none	VDD_EXT	Desc: SYS Boot Mode Control 2. Notes: No notes.
SYS_CLKIN	a	na	none	none	none	none	none	VDD_EXT	Desc: SYS Clock Input/Crystal Input. Notes: Active during reset.
SYS_CLKOUT	I/O	A	none	none	L	none	none	VDD_EXT	Desc: SYS Processor Clock Output. Notes: No notes.
SYS_EXTWAKE	I/O	A	none	none	H	none	L	VDD_EXT	Desc: SYS External Wake Control. Notes: Drives low during hibernate and high all other times.
SYS_FAULT	I/O	A	none	none	none	none	none	VDD_EXT	Desc: SYS Fault. Notes: Open source, requires an external pull-down resistor.
$\overline{\text{SYS\_FAULT}}$	I/O	A	none	none	none	none	none	VDD_EXT	Desc: SYS Complementary Fault. Notes: Open drain, requires an external pull-up resistor.
$\overline{\text{SYS\_HWRST}}$	I/O	na	none	none	none	none	none	VDD_EXT	Desc: SYS Processor Hardware Reset Control. Notes: Active during reset.
$\overline{\text{SYS\_NMI\_RESOUT}}$	I/O	A	none	none	L	none	none	VDD_EXT	Desc: SYS Reset Output | SYS Non-maskable Interrupt. Notes: Requires an external pull-up resistor.

TIMING SPECIFICATIONS

Specifications are subject to change without notice.

Clock and Reset Timing

Table 26 and Figure 10 describe clock and reset operations. Per the CCLK, SYSCLK, SCLK0, SCLK1, DCLK, and OCLK timing specifications in Table 17 on Page 53, combinations of SYS_CLKIN and clock multipliers must not select clock rates in excess of the processor's maximum instruction rate.

Table 26. Clock and Reset Timing

Parameter		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Timing Requirements				
f_{CKIN}	SYS_CLKIN Frequency (using a crystal)[1, 2, 3]	20	50	MHz
f_{CKIN}	SYS_CLKIN Frequency (using a crystal oscillator)[1, 2, 3]	20	60	MHz
t_{CKINL}	SYS_CLKIN Low Pulse[1]	6.67		ns
t_{CKINH}	SYS_CLKIN High Pulse[1]	6.67		ns
t_{WRST}	$\overline{\text{SYS\_HWRST}}$ Asserted Pulse Width Low[4]	11 × t_{CKIN}		ns

[1] Applies to PLL bypass mode and PLL non bypass mode.
[2] The t_{CKIN} period (see Figure 10) equals $1/f_{CKIN}$.
[3] If the CGU_CTL.DF bit is set, the minimum f_{CKIN} specification is 40 MHz.
[4] Applies after power-up sequence is complete. See Table 27 and Figure 11 for power-up reset timing.

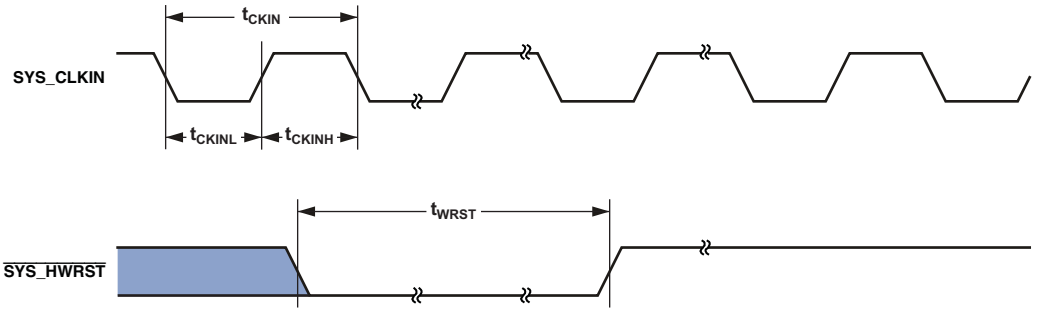


Figure 10. Clock and Reset Timing

Asynchronous Flash Read

Table 29. Asynchronous Flash Read

Parameter		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Switching Characteristics				
t_{AMSADV}	SMC0_Ax (Address)/$\overline{\text{SMC0\_AMSx}}$ Assertion Before SMC0_NORDV Low[1]	PREST × t_{SCLK0} – 2		ns
t_{WADV}	SMC0_NORDV Active Low Width[2]	RST × t_{SCLK0} – 2		ns
$t_{DADVARE}$	$\overline{\text{SMC0\_ARE}}$ Low Delay From SMC0_NORDV High[3]	PREAT × t_{SCLK0} – 2		ns
t_{HARE}	Output[4] Hold After $\overline{\text{SMC0\_ARE}}$ High[5]	RHT × t_{SCLK0} – 2		ns
t_{WARE}[6]	$\overline{\text{SMC0\_ARE}}$ Active Low Width[7]	RAT × t_{SCLK0} – 2		ns

[1] PREST value set using the SMC_BxETIM.PREST bits.
[2] RST value set using the SMC_BxTIM.RST bits.
[3] PREAT value set using the SMC_BxETIM.PREAT bits.
[4] Output signals are SMC0_Ax, $\overline{\text{SMC0\_AMS}}$, $\overline{\text{SMC0\_AOE}}$.
[5] RHT value set using the SMC_BxTIM.RHT bits.
[6] SMC0_BxCTL.ARDYEN bit = 0.
[7] RAT value set using the SMC_BxTIM.RAT bits.

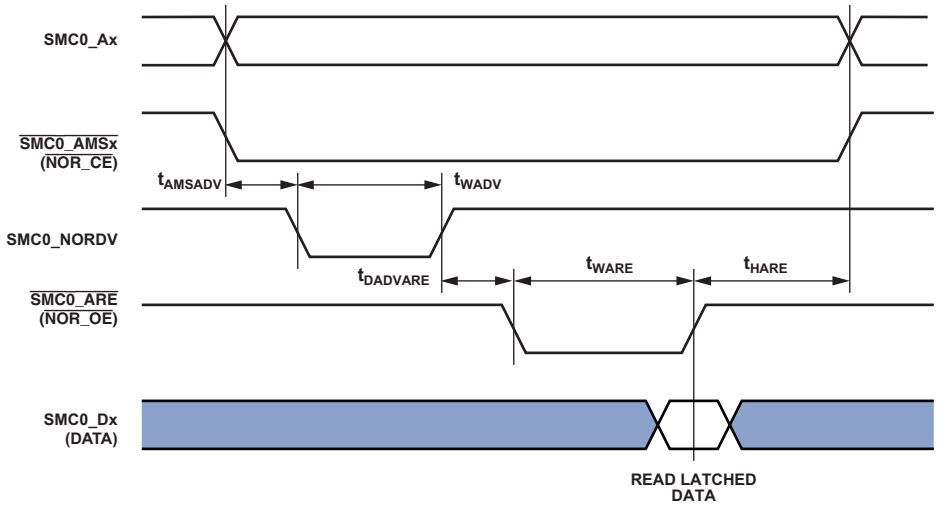
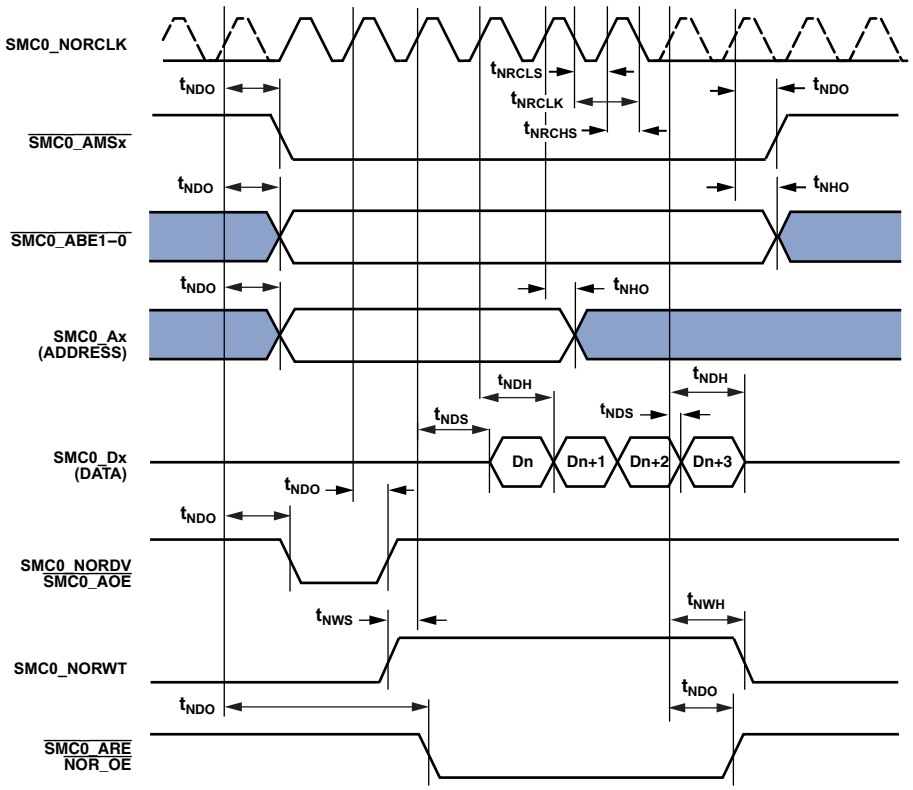


Figure 13. Asynchronous Flash Read



NOTE: SMC0_NORCLK dotted line represents a free running version of SMC0_NORCLK that is not visible on the SMC0_NORCLK pin.

Figure 15. Synchronous Burst AC Interface Timing

Table 43. Enhanced Parallel Peripheral Interface—External Clock

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Timing Requirements						
t_{PCLKW}	EPPI_CLK Width[1]	$(0.5 \times t_{PCLKEXT}) - 1.25$		$(0.5 \times t_{PCLKEXT}) - 1.25$		ns
t_{PCLK}	EPPI_CLK Period[1]	$t_{PCLKEXT} - 1.25$		$t_{PCLKEXT} - 1.25$		ns
t_{SFSPE}	External FS Setup Before EPPI_CLK	2		2		ns
t_{HFSPE}	External FS Hold After EPPI_CLK	3.7		3.7		ns
t_{SDRPE}	Receive Data Setup Before EPPI_CLK	2		2		ns
t_{HDRPE}	Receive Data Hold After EPPI_CLK	3.7		3.7		ns
Switching Characteristics						
t_{DFSPE}	Internal FS Delay After EPPI_CLK		20.1		15.3	ns
t_{HOFSPE}	Internal FS Hold After EPPI_CLK	2.4		2.4		ns
t_{DDTPE}	Transmit Data Delay After EPPI_CLK		20.1		15.3	ns
t_{HDTPE}	Transmit Data Hold After EPPI_CLK	2.4		2.4		ns

[1] This specification indicates the minimum instantaneous width or period that can be tolerated due to duty cycle variation or jitter on the external EPPI_CLK. For the external EPPI_CLK ideal maximum frequency see the $f_{PCLKEXT}$ specification in Table 17 on Page 53.

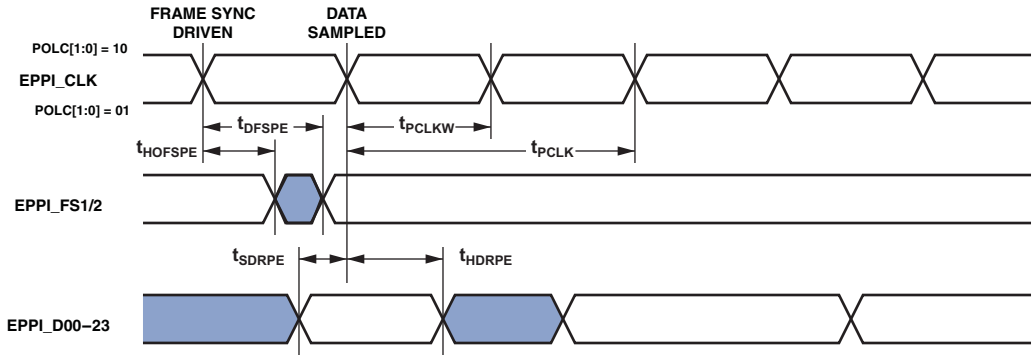


Figure 30. PPI External Clock GP Receive Mode with Internal Frame Sync Timing

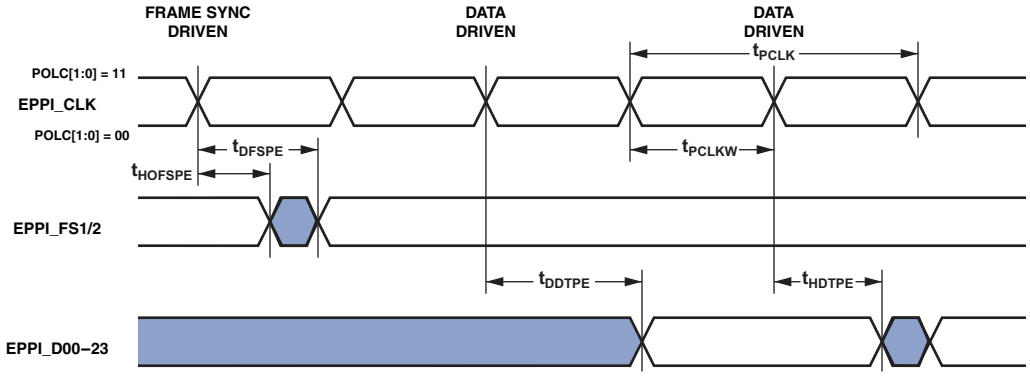


Figure 31. PPI External Clock GP Transmit Mode with Internal Frame Sync Timing

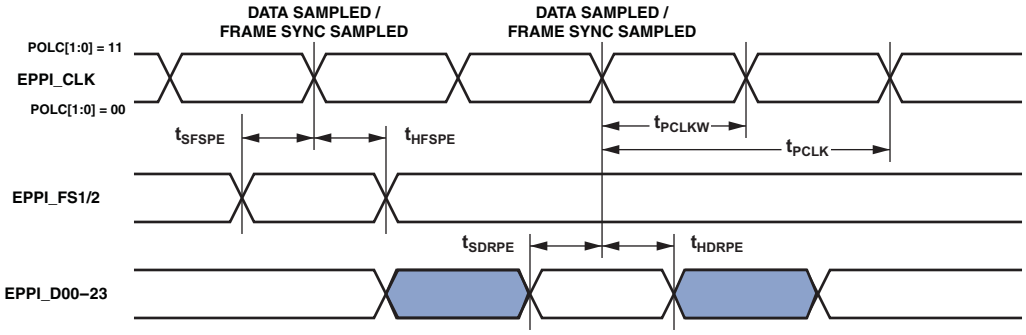


Figure 32. PPI External Clock GP Receive Mode with External Frame Sync Timing

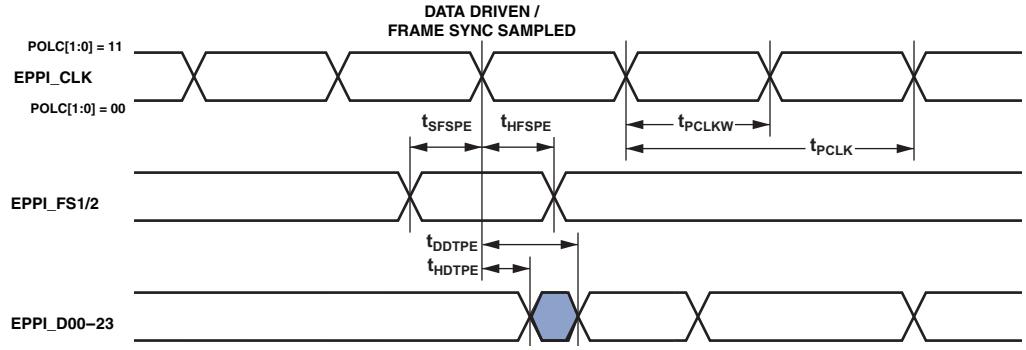
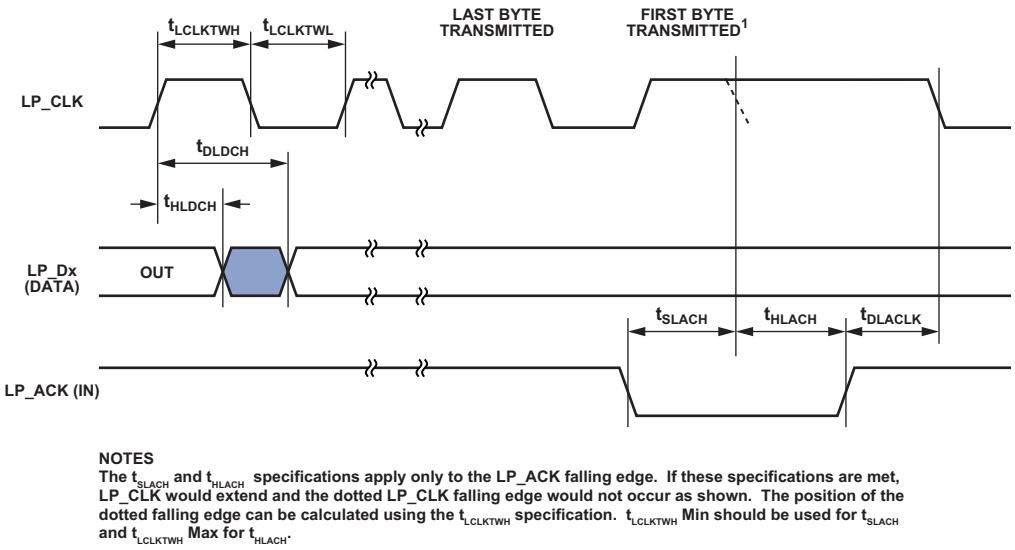


Figure 33. PPI External Clock GP Transmit Mode with External Frame Sync Timing

Table 45. Link Ports—Transmit

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Timing Requirements						
t_{SLACH}	LP_ACK Setup Before LP_CLK Low	$2 \times t_{SCLK0} + 17.5$		$2 \times t_{SCLK0} + 13.5$		ns
t_{HLACH}	LP_ACK Hold After LP_CLK Low	0		0		ns
Switching Characteristics						
t_{DLDCH}	Data Delay After LP_CLK High		2.5		2.5	ns
t_{HLDCH}	Data Hold After LP_CLK High	–1.5		–1.5		ns
$t_{LCLKTWL}$[1]	LP_CLK Width Low	$0.4 \times t_{LCLKTPROG}$	$0.6 \times t_{LCLKTPROG}$	$0.4 \times t_{LCLKTPROG}$	$0.6 \times t_{LCLKTPROG}$	ns
$t_{LCLKTWH}$[1]	LP_CLK Width High	$0.4 \times t_{LCLKTPROG}$	$0.6 \times t_{LCLKTPROG}$	$0.4 \times t_{LCLKTPROG}$	$0.6 \times t_{LCLKTPROG}$	ns
t_{LCLKTW}[1]	LP_CLK Period	$t_{LCLKTPROG} - 1.2$		$t_{LCLKTPROG} - 1.2$		ns
t_{DLACLK}	LP_CLK Low Delay After LP_ACK High	$t_{SCLK0} + 4$	$(2 \times t_{SCLK0}) + t_{LCLK} + 10$	$t_{SCLK0} + 4$	$(2 \times t_{SCLK0}) + t_{LCLK} + 10$	ns

[1] See Table 17 on Page 53 in Clock Related Operating Conditions for details on the minimum period that may be programmed for $t_{LCLKTPROG}$.



NOTES
The t_{SLACH} and t_{HLACH} specifications apply only to the LP_ACK falling edge. If these specifications are met, LP_CLK would extend and the dotted LP_CLK falling edge would not occur as shown. The position of the dotted falling edge can be calculated using the $t_{LCLKTWH}$ specification. $t_{LCLKTWH}$ Min should be used for t_{SLACH} and $t_{LCLKTWH}$ Max for t_{HLACH}.

Figure 35. Link Ports—Transmit

Serial Peripheral Interface (SPI) Port—Master Timing

Table 51 and Figure 40 describe SPI port master operations.

When internally generated, the programmed SPI clock ($f_{SPICLKPROG}$) frequency in MHz is set by the following equation where BAUD is a field in the SPI_CLK register that can be set from 0 to 65535:

$$f_{SPICLKPROG} = \frac{f_{SCLK1}}{(BAUD + 1)}$$

$$t_{SPICLKPROG} = \frac{1}{f_{SPICLKPROG}}$$

Note that:

- In dual mode data transmit the SPI_MISO signal is also an output.
- In quad mode data transmit the SPI_MISO, SPI_D2, and SPI_D3 signals are also outputs.
- In dual mode data receive the SPI_MOSI signal is also an input.
- In quad mode data receive the SPI_MOSI, SPI_D2, and SPI_D3 signals are also inputs.
- To add additional frame delays see the documentation for the SPI_DLY register in the hardware reference manual.

Table 51. Serial Peripheral Interface (SPI) Port—Master Timing

Parameter		V_{DD_EXT} 1.8 V Nominal Min	V_{DD_EXT} 1.8 V Nominal Max	V_{DD_EXT} 3.3 V Nominal Min	V_{DD_EXT} 3.3 V Nominal Max	Unit
Timing Requirements						
t_{SSPIDM}	Data Input Valid to SPI_CLK Edge (Data Input Setup)	4.6		3.2		ns
t_{HSPIDM}	SPI_CLK Sampling Edge to Data Input Invalid	1.3		1.3		ns
Switching Characteristics						
t_{SDSCIM}	$\overline{SPI\_SEL}$ low to First SPI_CLK Edge	$0.5 \times t_{SCLK1} - 2$		$0.5 \times t_{SCLK1} - 2$		ns
t_{SPICHM}	SPI_CLK High Period[1]	$0.5 \times t_{SPICLKPROG} - 1.5$		$0.5 \times t_{SPICLKPROG} - 1.5$		ns
t_{SPICLM}	SPI_CLK Low Period[1]	$0.5 \times t_{SPICLKPROG} - 1.5$		$0.5 \times t_{SPICLKPROG} - 1.5$		ns
t_{SPICLK}	SPI_CLK Period[1]	$t_{SPICLKPROG} - 1.5$		$t_{SPICLKPROG} - 1.5$		ns
t_{HDSM}	Last SPI_CLK Edge to $\overline{SPI\_SEL}$ High	$(0.5 \times t_{SCLK1}) - 1.5$		$(0.5 \times t_{SCLK1}) - 1.5$		ns
t_{SPITDM}	Sequential Transfer Delay	$t_{SCLK1} - 1.5$		$t_{SCLK1} - 1.5$		ns
$t_{DDSPIDM}$	SPI_CLK Edge to Data Out Valid (Data Out Delay)		2.6		2.6	ns
$t_{HDSPIDM}$	SPI_CLK Edge to Data Out Invalid (Data Out Hold)	–1		–1		ns

[1] See Table 17 on Page 53 in Clock Related Operating Conditions for details on the minimum period that may be programmed for $t_{SPICLKPROG}$.

Serial Peripheral Interface (SPI) Port—SPI_RDY Timing

SPI_RDY is used to provide flow control. The CPOL and CPHA bits are set in SPI_CTL, while LEADX, LAGX, and STOP are in SPI_DLY.

Table 56. SPI Port—SPI_RDY Timing

Parameter		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Timing Requirements				
$t_{SRDYSCKM0}$	Minimum Setup Time for SPI_RDY De-assertion in Master Mode Before Last SPI_CLK Edge of Valid Data Transfer to Block Subsequent Transfer with CPHA = 0	$(2.5 + 1.5 \times BAUD^{1}) \times t_{SCLK1} + 17.5$		ns
$t_{SRDYSCKM1}$	Minimum Setup Time for SPI_RDY De-assertion in Master Mode Before Last SPI_CLK Edge of Valid Data Transfer to Block Subsequent Transfer with CPHA = 1	$(1.5 + BAUD^{1}) \times t_{SCLK1} + 17.5$		ns
Switching Characteristic				
$t_{SRDYSCKM}$	Time Between Assertion of SPI_RDY by Slave and First Edge of SPI_CLK for New SPI Transfer with CPHA = 0 and BAUD = 0 (STOP, LEADX, LAGX = 0)	$3 \times t_{SCLK1}$	$4 \times t_{SCLK1} + 17.5$	ns
	Time Between Assertion of SPI_RDY by Slave and First Edge of SPI_CLK for New SPI Transfer with CPHA = 0 and BAUD ≥ 1 (STOP, LEADX, LAGX = 0)	$(4 + 1.5 \times BAUD^{1}) \times t_{SCLK1}$	$(5 + 1.5 \times BAUD^{1}) \times t_{SCLK1} + 17.5$	ns
	Time Between Assertion of SPI_RDY by Slave and First Edge of SPI_CLK for New SPI Transfer with CPHA = 1 (STOP, LEADX, LAGX = 0)	$(3 + 0.5 \times BAUD^{1}) \times t_{SCLK1}$	$(4 + 0.5 \times BAUD^{1}) \times t_{SCLK1} + 17.5$	ns

[1] BAUD value set using the SPI_CLK.BAUD bits.

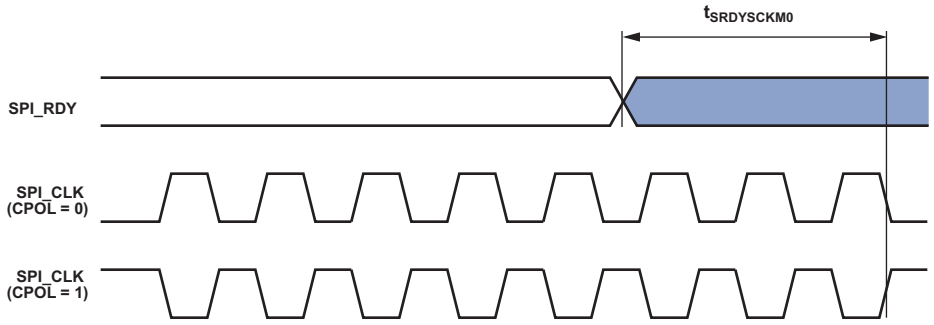


Figure 46. SPI_RDY Setup Before SPI_CLK with CPHA = 0

General-Purpose Port Timing

Table 57 and Figure 49 describe general-purpose port operations.

Table 57. General-Purpose Port Timing

Parameter		V_{DD_EXT} 1.8 V/3.3 V Nominal Min	Max	Unit
Timing Requirement				
t_{WFI}	General-Purpose Port Pin Input Pulse Width	$2 \times t_{SCLK0} - 1.5$		ns

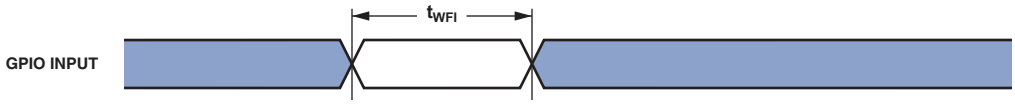


Figure 49. General-Purpose Port Timing

Timer Cycle Timing

Table 58 and Figure 50 describe timer expired operations. The input signal is asynchronous in "width capture mode" and "external clock mode" and has an ideal maximum input frequency of (f_{SCLK0}/4) MHz. The Period Value (VALUE) is the timer period assigned in the TMx_TMRn_PER register and can range from 2 to $2^{32} - 1$.

Table 58. Timer Cycle Timing

Parameter		V_{DD_EXT} 1.8 V Nominal Min	Max	V_{DD_EXT} 3.3 V Nominal Min	Max	Unit
Timing Requirements						
t_{WL}	Timer Pulse Width Input Low[1]	$2 \times t_{SCLK0} - 1.5$		$2 \times t_{SCLK0} - 1.5$		ns
t_{WH}	Timer Pulse Width Input High[1]	$2 \times t_{SCLK0} - 1.5$		$2 \times t_{SCLK0} - 1.5$		ns
Switching Characteristics						
t_{HTO}	Timer Pulse Width Output	$t_{SCLK0} \times VALUE - 1.5$		$t_{SCLK0} \times VALUE - 1.5$		ns

[1] This specification indicates the minimum instantaneous width that can be tolerated due to duty cycle variation or jitter for TMx signals in width capture and external clock modes. The ideal maximum frequency for TMx signals is listed in Timer Cycle Timing on this page.

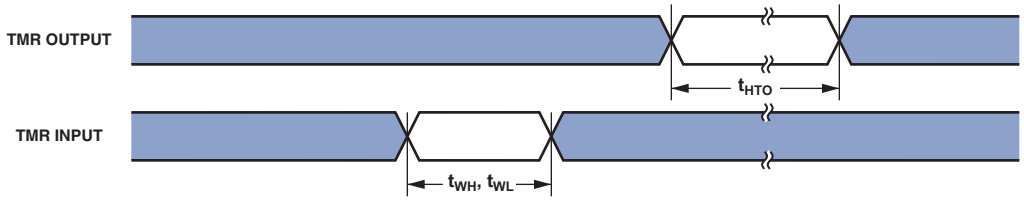


Figure 50. Timer Cycle Timing

Universal Asynchronous Receiver-Transmitter (UART) Ports—Receive and Transmit Timing

The UART ports receive and transmit operations are described in the *ADSP-BF60x Blackfin Processor Hardware Reference Manual*.

CAN Interface

The CAN interface timing is described in the *ADSP-BF60x Blackfin Processor Hardware Reference Manual*.

Universal Serial Bus (USB) On-The-Go—Receive and Transmit Timing

Table 62 describes the USB On-The-Go receive and transmit operations.

Table 62. USB On-The-Go—Receive and Transmit Timing

Parameter		V_{DD_USB} 3.3 V Nominal Min	Max	Unit
Timing Requirements				
f_{USBS}	USB_XI Frequency	48	48	MHz
fs_{USB}	USB_XI Clock Frequency Stability	–50	+50	ppm

The time t_{DECAY} is calculated with test loads C_L and I_L, and with ΔV equal to 0.25 V for V_{DDEXT}/V_{DDMEM} (nominal) = 2.5 V/3.3 V and 0.15 V for V_{DDEXT}/V_{DDMEM} (nominal) = 1.8V.

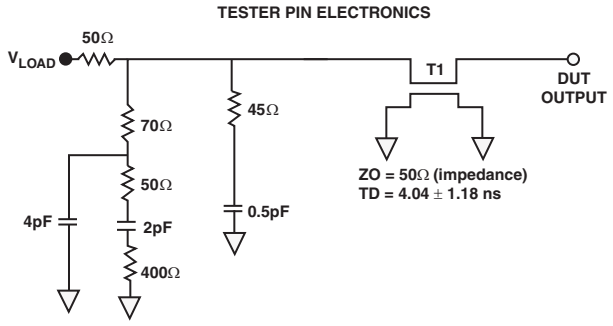
The time $t_{DIS\_MEASURED}$ is the interval from when the reference signal switches, to when the output voltage decays ΔV from the measured output high or output low voltage.

Example System Hold Time Calculation

To determine the data output hold time in a particular system, first calculate t_{DECAY} using the equation given above. Choose ΔV to be the difference between the processor's output voltage and the input threshold for the device requiring the hold time. C_L is the total bus capacitance (per data line), and I_L is the total leakage or three-state current (per data line). The hold time is t_{DECAY} plus the various output disable times as specified in the Timing Specifications on Page 60.

Capacitive Loading

Output delays and holds are based on standard capacitive loads of an average of 6 pF on all pins (see Figure 67). V_{LOAD} is equal to $(V_{DD\_EXT})/2$.



NOTES:
THE WORST CASE TRANSMISSION LINE DELAY IS SHOWN AND CAN BE USED FOR THE OUTPUT TIMING ANALYSIS TO REFELECT THE TRANSMISSION LINE EFFECT AND MUST BE CONSIDERED. THE TRANSMISSION LINE (TD) IS FOR LOAD ONLY AND DOES NOT AFFECT THE DATA SHEET TIMING SPECIFICATIONS.

ANALOG DEVICES RECOMMENDS USING THE IBIS MODEL TIMING FOR A GIVEN SYSTEM REQUIREMENT. IF NECESSARY, A SYSTEM MAY INCORPORATE EXTERNAL DRIVERS TO COMPENSATE FOR ANY TIMING DIFFERENCES.

Figure 67. Equivalent Device Loading for AC Measurements (Includes All Fixtures)

The graphs of Figure 68 through Figure 70 show how output rise and fall times vary with capacitance. The delay and hold specifications given should be derated by a factor derived from these figures. The graphs in these figures may not be linear outside the ranges shown.

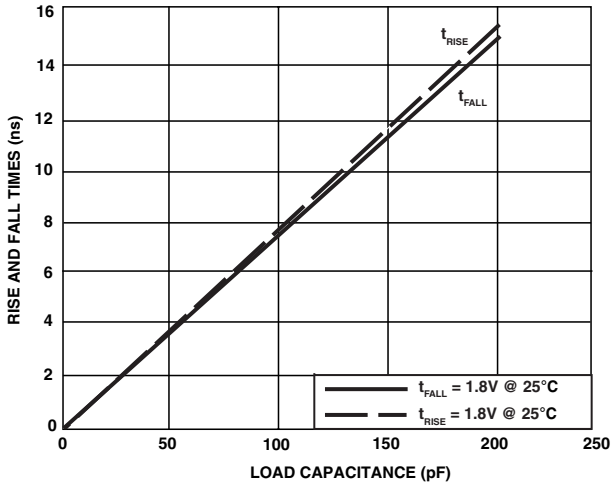


Figure 68. Driver Type A Typical Rise and Fall Times (10%-90%) vs. Load Capacitance (V_{DD_EXT} = 1.8 V)

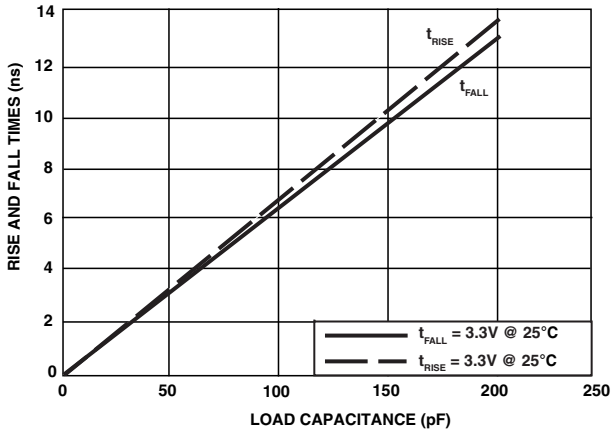


Figure 69. Driver Type A Typical Rise and Fall Times (10%-90%) vs. Load Capacitance (V_{DD_EXT} = 3.3 V)

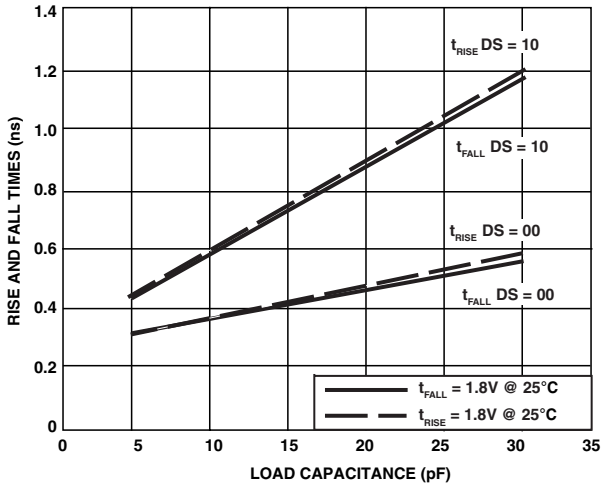


Figure 70. Driver Type B & C Typical Rise and Fall Times (10%-90%) vs. Load Capacitance (V_{DD_DMC} = 1.8 V)