CY28551

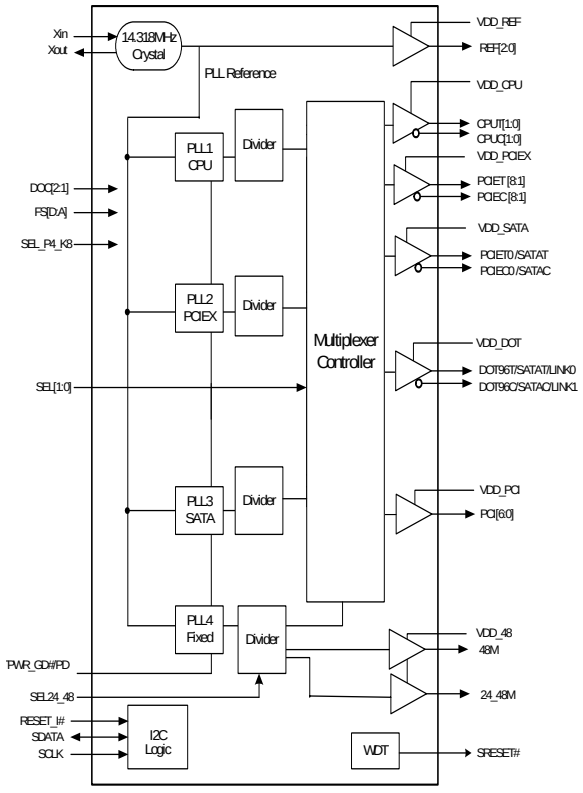
Universal Clock Generator for Intel, VIA, and SIS®

Features

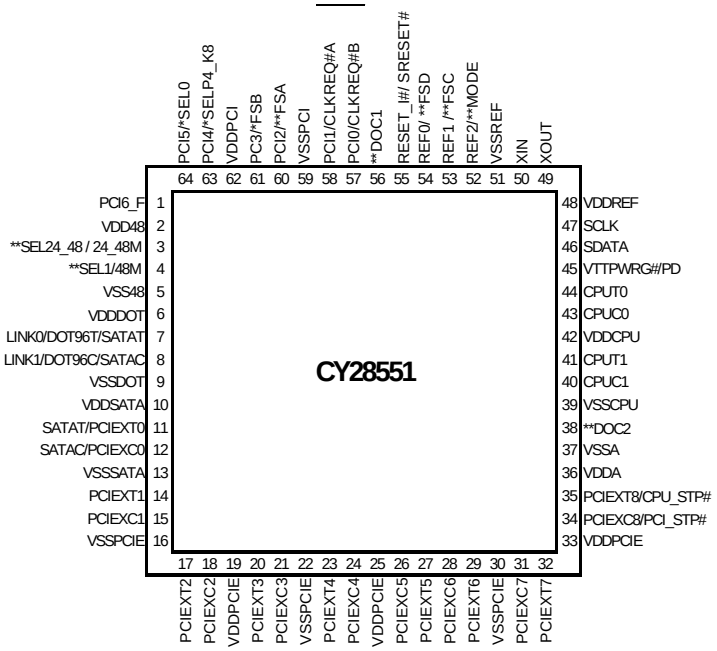
- Compliant to Intel® CK505
- Selectable CPU clock buffer type for Intel P4 or K8 selection
- Selectable CPU frequencies
- Universal clock to support Intel, SiS and VIA platform
- 0.7V Differential CPU clock for Intel CPU
- 3.3V Differential CPU clock for AMD K8
- 100 MHz differential SRC clocks
- 96 MHz differential dot clock
- 133 MHz Link clock
- 48 MHz USB clock
- 33 MHz PCI clocks
- Dynamic Frequency Control
- Dial-A-Frequency®
- WatchDog Timer
- Two Independent Overclocking PLLs
- Low-voltage frequency select input
- I^2C support with readback capabilities
- Ideal Lexmark Spread Spectrum profile for maximum electromagnetic interference (EMI) reduction
- 3.3V Power supply
- 64-pin QFN package

CPU	SRC	SATA	PCI	REF	LINK	DOT96	24_48M	48M
x 2	x 8	x1	x 7	x 3	x2	x 1	x1	x 1

Block Diagram



Pin Configuration



*** Indicates internal pull up**
**** indicates internal pull down**

Pin Description

Pin No.	Name	Type	Description
1	PCI6_F	O	Free running 33 MHz clock output. Intel Type-3A output buffer
2	VDD48	PWR	3.3V power supply for outputs.
3	**SEL24_48#/24_48M	I/O, PD	3.3V tolerant input for 24 MHz, 48 MHz selection/24_48MHz clock output. Internal 150k pull down 1 = 24 MHz, 0 = 48 MHz Intel Type-3A output buffer
4	**SEL1/48MHz	I/O, PD	3.3V tolerant input for output selection/48MHz clock output. Refer to *Table 1* for selection options Internal 150k pull down
5	VSS48	GND	Ground for outputs
6	VDDDOT	PWR	3.3V Power supply for outputs
7,8	LINK0/DOT96T/SATAT LINK1/DOT96C/SATAC	O, SE/DiF	Link output for VIA and SIS, differential 96 MHz clock output and 100 MHz differential clock. The output is selected by SEL[1:0]
9	VSSDOT	GND	Ground for outputs
10	VDDSATA	PWR	3.3V Power supply for outputs
11,12	PCIEX0[T/C]/SATA[T/C]	O, DIF	Differential SRC clock output/Differential SATA SRC clock output Intel Type-SR output buffer
13	VSSSATA	GND	Ground for outputs
14,15	PCIEX[T/C]1	O, DIF	100 MHz Differential serial reference clock. Intel Type-SR output buffer
16	VSSPCIE	GND	Ground for outputs
17,18	PCIEX[T/C]2	O, DIF	100 MHz Differential serial reference clock. Intel Type-SR output buffer
19	VDDPCIE	PWR	3.3V power supply for outputs.
20,21	PCIEX[T/C]3	O, DIF	100 MHz Differential serial reference clock. Intel Type-SR output buffer
22	VSSPCIE	GND	Ground for outputs
23,24	PCIEX[T/C]4	O, DIF	100 MHz Differential serial reference clock. Intel Type-SR output buffer
25	VDDPCIE	PWR	3.3V power supply for outputs
26,27,28,29	PCIEX[T/C][5:6]	O, DIF	100 MHz Differential Serial reference clock. Intel Type-SR output buffer
30	VSSPCIE	GND	Ground for outputs
31,32	PCIEX[T/C]7	O, DIF	100 MHz Differential Serial reference clock. Intel Type-SR output buffer
33	VDDPCIE	PWR	3.3V power supply for outputs
34,35	PCIEXT8/CPU_STOP# PCIEXC8/PCI_STOP#	I/O, DIF	3.3V-tolerant input for stopping PCI and SRC outputs/3.3V-tolerant input for stopping CPU outputs/100-MHz Differential serial reference clocks. The two multifunction pins are selected by MODE. Default PCIEX8 Intel Type-SR output buffer
36	VDDA	PWR	3.3V Power supply for PLL.
37	VSSA	GND	Ground for PLL.
38	**DOC2	I, PD	Dynamic Over Clocking pin 0 = normal, 1 = Frequency will be changed depend on DOC register. Internal 150k pull-down.
39	VSSCPU	GND	Ground for outputs.
40,41	CPU[T/C]1	O, DIF	Differential CPU clock output. Intel Type-SR output buffer.
42	VDDCPU	PWR	3.3V Power supply for outputs+
43, 44	CPU[T/C]0	O, DIF	Differential CPU clock output. Intel Type-SR output buffer.

Pin Description (continued)

Pin No.	Name	Type	Description
45	VTT_PWRGD#/PD	I	3.3V LVTTL input. This pin is a level-sensitive strobe used to latch the HW strapping pin inputs. After asserting VTT_PWRGD# (active LOW), this pin becomes a real-time input for asserting power-down (active HIGH).
46	SDATA	I/O	SMBus compatible SDATA
47	SCLK	I	SMBus compatible SCLOCK.
48	VDDREF	PWR	3.3V Power supply for outputs
49	XOUT	O	14.318 MHz Crystal Output
50	XIN	I	14.318 MHz Crystal Input
51	VSSREF	GND	Ground for outputs
52	REF2	O, SE	14.318 MHz REF clock output. Intel Type-5 output buffer
53	**FSC/REF1	I/O,PD, SE	3.3V tolerant input for CPU frequency selection/14.318 MHz REF clock output Internal 150k pull down Intel Type-5 output buffer Refer to DC Electrical Specifications table for Vil_FS and Vih_FS specifications
54	**FSD/REF0	I/O,PD, SE	3.3V tolerant input for CPU frequency selection/14.318 MHz REF clock output Internal 150k pull down Intel Type-5 output buffer Refer to DC Electrical Specifications table for Vil_FS and Vih_FS specifications
55	RESET_I#/SRESET#	I/O, OD	3.3V tolerant input for reset all of registers to default setting 3.3V LVTTL output for watchdog reset signal
56	**DOC1	I, PD	Dynamic Over Clocking pin 0 = normal; 1 = Frequency will be changed depend on DOC register. Internal 150k pull-down
57	PCI0/**CLKREQ#B	I/O,SE, PD	33 MHz clock output/Output enable control for PCIEX4; 5 via I2C register Default is PCI0 0 = Selected PCIEXs are enabled, 1 = Selected PCIEXs are disabled. Internal 150k pull down Intel Type-3A output buffer
58	PCI1/**CLKREQ#A	I/O,SE, PD	33 MHz clock output/Output enable control for PCIEX6, 7via I2C register. Default is PCI1 0 = Selected PCIEXs are enabled, 1 = Selected PCIEXs are disabled. Internal 150k pull down Intel Type-3A output buffer
59	VSSPCI	GND	Ground for outputs.
60	**FSA/PCI2	I/O, PD	3.3V tolerant input for CPU frequency selection/33 MHz clock output. Internal 150k pull down Intel Type-3A output buffer Refer to DC Electrical Specifications table for Vil_FS and Vih_FS specifications
61	*FSB/PCI3	I/O, PU	3.3V tolerant input for CPU frequency selection/33 MHz clock output. Internal 150k pull up Intel Type-3A output buffer Refer to DC Electrical Specifications table for Vil_FS and Vih_FS specifications
62	VDDPCI	PWR	3.3V power supply for outputs.
63	*SELP4_K8/PCI3	I/O, PU	3.3V tolerant input for CPU clock output buffer type selection/33 MHz clock output. Internal 150k pull up Intel Type-3A output buffer Refer to DC Electrical Specifications table for Vil_FS and Vih_FS specifications 0 = K8 CPU buffer type, 1 = P4 CPU buffer type.
64	*SEL0/PCI5	I/O, PU	3.3V tolerant input for output selection/33 MHz clock output. Refer to *Table 1* for selection options. Internal 150k pull up

Table 1. Frequency Select Table

FSD	FSC	FSB	FSA	Frequency Table (ROM)												
FSEL3	FSEL2	FSEL1	FSEL0	CPU0	CPU1	SRC	LINK	PCI	CPU VCO	CPU PLL Gear Constant (G)	CPU M	CPU N	PCIE VCO	SRC PLL Gear Constant	PCIE M	PCIE N
0	0	0	0	266.6666667	266.6666667	100	66.6667	33.3333	800	80	60	200	800	30	60	200
0	0	0	1	133.3333333	133.3333333	100	66.6667	33.3333	800	40	60	200	800	30	60	200
0	0	1	0	200	200	100	66.6667	33.3333	800	60	60	200	800	30	60	200
0	0	1	1	166.6666667	166.6666667	100	66.6667	33.3333	666.6666667	60	63	175	800	30	60	200
0	1	0	0	333.3333333	333.3333333	100	66.6667	33.3333	666.6666667	120	63	175	800	30	60	200
0	1	0	1	100	100	100	66.6667	33.3333	800	30	60	200	800	30	60	200
0	1	1	0	400	400	100	66.6667	33.3333	800	120	60	200	800	30	60	200
0	1	1	1	200	250	100	66.6667	33.3333	1000	60	60	250	800	30	60	200
1	0	0	0	266.6666667	266.6666667	100	133.3333	33.3333	800	80	60	200	800	30	60	200
1	0	0	1	133.3333333	133.3333333	100	133.3333	33.3333	800	40	60	200	800	30	60	200
1	0	1	0	200	200	100	133.3333	33.3333	800	60	60	200	800	30	60	200
1	0	1	1	166.6666667	166.6666667	100	133.3333	33.3333	666.6666667	60	63	175	800	30	60	200
1	1	0	0	333.3333333	333.3333333	100	133.3333	33.3333	666.6666667	120	63	175	800	30	60	200
1	1	0	1	100	100	100	133.3333	33.3333	800	30	60	200	800	30	60	200
1	1	1	0	400	400	100	133.3333	33.3333	800	120	60	200	800	30	60	200
1	1	1	1	200	250	100	133.3333	33.3333	1000	60	60	250	800	30	60	200

Frequency Select Pins (FS[D:A])

To achieve host clock frequency selection, apply the appropriate logic levels to FS_A, FS_B, FS_C, and FS_D inputs prior to VTT_PWRGD# assertion (as seen by the clock synthesizer). When VTT_PWRGD# is sampled LOW by the clock chip (indicating processor VTT voltage is stable), the clock chip samples the FS_A, FS_B, FS_C, and FS_D input values. For all logic levels of FS_A, FS_B, FS_C, FS_D, and FS_E, VTT_PWRGD# employs a one-shot functionality, in that once a valid LOW on VTT_PWRGD# has been sampled, all further VTT_PWRGD#, FS_A, FS_B, FS_C, and FS_D transitions will be ignored, except in test mode.

Serial Data Interface

To enhance the flexibility and function of the clock synthesizer, a two-signal serial interface is provided. Through the Serial Data Interface, various device functions, such as individual clock output buffers, can be individually enabled or disabled. The registers associated with the Serial Data Interface initialize to their default setting upon power-up, and therefore use of this interface is optional. Clock device register changes are normally made upon system initialization, if any are required. The interface cannot be used during system operation for power management functions.

Data Protocol

The clock driver serial protocol accepts byte write, byte read, block write, and block read operations from the controller. For block write/read operation, the bytes must be accessed in sequential order from lowest to highest byte (most significant bit first) with the ability to stop after any complete byte has been transferred. For byte write and byte read operations, the system controller can access individually indexed bytes. The offset of the indexed byte is encoded in the command code, as described in *Table 2*.

The block write and block read protocol is outlined in *Table 3*, while *Table 4* outlines the corresponding byte write and byte read protocol. The slave receiver address is 11010010 (D2h).

Table 2. Command Code Definition

Bit	Description
7	0 = Block read or block write operation, 1 = Byte read or byte write operation
(6:0)	Byte offset for byte read or byte write operation. For block read or block write operations, these bits should be '0000000'

Table 3. Block Read and Block Write Protocol

Block Write Protocol		Block Read Protocol	
Bit	**Description**	**Bit**	**Description**
1	Start	1	Start
8:2	Slave address – 7 bits	8:2	Slave address – 7 bits
9	Write	9	Write
10	Acknowledge from slave	10	Acknowledge from slave
18:11	Command Code – 8 bits	18:11	Command Code – 8 bits
19	Acknowledge from slave	19	Acknowledge from slave
27:20	Byte Count – 8 bits (Skip this step if I^2C_EN bit set)	20	Repeat start
28	Acknowledge from slave	27:21	Slave address – 7 bits
36:29	Data byte 1 – 8 bits	28	Read = 1
37	Acknowledge from slave	29	Acknowledge from slave
45:38	Data byte 2 – 8 bits	37:30	Byte Count from slave – 8 bits
46	Acknowledge from slave	38	Acknowledge
....	Data Byte/Slave Acknowledges	46:39	Data byte 1 from slave – 8 bits
....	Data Byte N – 8 bits	47	Acknowledge
....	Acknowledge from slave	55:48	Data byte 2 from slave – 8 bits
....	Stop	56	Acknowledge
		....	Data bytes from slave/Acknowledge
		....	Data Byte N from slave – 8 bits
		....	NOT Acknowledge
		....	Stop

Table 4. Byte Read and Byte Write Protocol

Byte Write Protocol		Byte Read Protocol	
Bit	**Description**	**Bit**	**Description**
1	Start	1	Start
8:2	Slave address – 7 bits	8:2	Slave address – 7 bits
9	Write	9	Write
10	Acknowledge from slave	10	Acknowledge from slave
18:11	Command Code – 8 bits	18:11	Command Code – 8 bits
19	Acknowledge from slave	19	Acknowledge from slave
27:20	Data byte – 8 bits	20	Repeated start
28	Acknowledge from slave	27:21	Slave address – 7 bits
29	Stop	28	Read
		29	Acknowledge from slave
		37:30	Data from slave – 8 bits
		38	NOT Acknowledge
		39	Stop

Control Registers

Byte 0: Control Register 0

Bit	@Pup	Type	Name	Description
7	1	R/W	PCIEX[T/C]7	PCIEX[T/C]7 Output Enable 0 = Disable (Tri-state), 1 = Enable
6	1	R/W	PCIEX[T/C]6	PCIEX[T/C]6 Output Enable 0 = Disable (Tri-state), 1 = Enable
5	1	R/W	PCIEX[T/C]5	PCIEX[T/C]5 Output Enable 0 = Disable (Tri-state), 1 = Enable
4	1	R/W	PCIEX[T/C]4	PCIEX[T/C]4 Output Enable 0 = Disable (Tri-state), 1 = Enable
3	1	R/W	PCIEX[T/C]3	PCIEX[T/C]3 Output Enable 0 = Disable (Tri-state), 1 = Enable
2	1	R/W	PCIEX[T/C]2	PCIEX[T/C]2 Output Enable 0 = Disable (Tri-state), 1 = Enable
1	1	R/W	PCIEX[T/C]1	PCIEX[T/C]1 Output Enable 0 = Disable (Tri-state), 1 = Enable
0	1	R/W	SATA/PCIEX[T/C]0	SATA/PCIEX[T/C]0 Output Enable 0 = Disable (Tri-state), 1 = Enable

Byte 1: Control Register 1

Bit	@Pup	Type	Name	Description
7	1	R/W	SATA/DOT96]	SATA/DOT96Output Enable 0 = Disable (Tri-state), 1 = Enable
6	1	R/W	24_48M	24_48M Output Enable 0 = Disable, 1 = Enable
5	1	R/W	48M	48M Output Enable 0 = Disable, 1 = Enable
4	1	R/W	REF2	REF2 Output Enable 0 = Disable, 1 = Enable
3	1	R/W	REF1	REF1 Output Enable 0 = Disable, 1 = Enable
2	1	R/W	REF0	REF0 Output Enable 0 = Disable, 1 = Enable
1	1	R/W	CPU[T/C]1	CPU[T/C]1 Output Enable 0 = Disable (Tri-state), 1 = Enable
0	1	R/W	CPU[T/C]0	CPU[T/C]0 Output Enable 0 = Disable (Tri-state), 1 = Enable

Byte 2: Control Register 2

Bit	@Pup	Type	Name	Description
7	1	R/W	Reserved	Reserved
6	1	R/W	PCI6_F	PCI6_F Output Enable 0 = Disable, 1 = Enable
5	1	R/W	PCI5	PCI5 Output Enable 0 = Disable, 1 = Enable
4	1	R/W	PCI4	PCI4 Output Enable 0 = Disable, 1 = Enable

Byte 2: Control Register 2 (continued)

Bit	@Pup	Type	Name	Description
3	1	R/W	PCI3	PCI3 Output Enable 0 = Disable, 1 = Enable
2	1	R/W	PCI2	PCI2 Output Enable 0 = Disable, 1 = Enable
1	1	R/W	PCI1	PCI1 Output Enable 0 = Disable, 1 = Enable
0	1	R/W	PCI0	PCI0 Output Enable 0 = Disable, 1 = Enable

Byte 3: Control Register 3

Bit	@Pup	Type	Name	Description
7	1	R/W	LINK1	LINK1 Output Enable 0 = Disable, 1 = Enable
6	1	R/W	LINK0	LINKI0 Output Enable 0 = Disable, 1 = Enable
5	1	R/W	PCIEX[T/C]8	PCIEX[T/C]8 Output Enable 0 = Disable (Tri-state), 1 = Enable
4	1	R/W	Reserved	Reserved
3	0	R/W	Reserved	Reserved
2	1	R/W	PCI	33 MHz Output Drive Strength 0 = 2x, 1 = 1x
1	1	R/W	REF	REF Output Drive Strength 0 = 2x, 1 = 1x
0	0	R/W	48M, 24_48M	48 MHz and 24_48M Output Drive Strength 0 = 2x, 1 = 1x

Byte 4: Control Register 4

Bit	@Pup	Type	Name	Description
7	0	R/W	CPU1	Allow control of CPU1 with assertion of CPU_STP# 0 = Free Running 1 = Stopped with CPU_STP#
6	0	R/W	CPU0	Allow control of CPU0 with assertion of CPU_STP# 0 = Free Running 1 = Stopped with CPU_STP#
5	0	R/W	PCI6_F	Allow control of PCI6_F with assertion of PCI_STP# 0 = Free Running 1 = Stopped with PCI_STP#
4	0	R/W	PCIEX	Allow control of PCIEX with assertion of PCI_STP# 0 = Free Running 1 = Stopped with PCI_STP#
3	0	R/W	FSEL_D	SW Frequency selection bits. See *Table 1*.
2	0	R/W	FSEL_C	
1	0	R/W	FSEL_B	
0	0	R/W	FSEL_A	

Byte 5: Control Register 5

Bit	@Pup	Type	Name	Description
7	0	R/W	CPU_SS1	CPU (PLL1) Spread Spectrum Selection 00: –0.5% (peak to peak) 01: ±0.25% (peak to peak) 10: –1.0% (peak to peak) 11: ±0.5% (peak to peak)
6	0	R/W	CPU_SS0	
5	0	R/W	CPU_SS_OFF	PLL1 (CPUPLL) Spread Spectrum Enable 0 = Spread off, 1 = Spread on
4	0	R/W	PCIE_SS0	PLL2 (PCIEPLL) Spread Spectrum Selection 0: –0.5% (peak to peak) 0: –1.0% (peak to peak)
3	0	R/W	PCIE_SS_OFF	PLL2 (PCIEPLL) Spread Spectrum Enable 0 = SRC spread off, 1 = SRC spread on
2	0	R/W	SATA_SS_OFF	PLL3 (SATAPLL) Spread Spectrum Enable 0 = Spread off, 1 = Spread on
1	HW	R/W	SEL24_48	24M/48 MHz output selection 0 = 48 MHz, 1 = 24 MHz
0	1	R/W	Reserved	Reserved

Byte 6: Control Register 6

Bit	@Pup	Type	Name	Description
7	0	R/W	SW_RESET	Software Reset. When set, the device asserts a reset signal on SRESET# upon completion of the block/word/byte write that set it. After asserting and deasserting the SRESET# this bit will self clear (set to 0).
6	0	R/W	Reserved	Reserved
5	0	R/W	FIX_LINK_PCI	LINK and PCI clock source selection 0 = PLL2(SRCPLL), 1 = PLL (SATAPLL)
4	HW	R	FSD	FSD Reflects the value of the FSD pin sampled on power up. 0 = FSD was low during VTT_PWRGD# assertion.
3	HW	R	FSC	FSC Reflects the value of the FSC pin sampled on power up. 0 = FSC was low during VTT_PWRGD# assertion.
2	HW	R	FSB	FSB Reflects the value of the FSB pin sampled on power up. 0 = FSB was LOW during VTT_PWRGD# assertion
1	HW	R	FSA	FSA Reflects the value of the FSA pin sampled on power up. 0 = FSA was LOW during VTT_PWRGD# assertion
0	HW	R	POWERGOOD	Power Status bit: 0 = Internal power or Internal resets are NOT valid 1 = Internal power and Internal resets are valid Read only Bit 7 sets to 0 when Bit 7 = 0

Byte 7: Vendor ID

Bit	@Pup	Type	Name	Description
7	0	R	Revision Code Bit 3	Revision Code Bit 3
6	0	R	Revision Code Bit 2	Revision Code Bit 2
5	1	R	Revision Code Bit 1	Revision Code Bit 1
4	0	R	Revision Code Bit 0	Revision Code Bit 0
3	1	R	Vendor ID Bit 3	Vendor ID Bit 3
2	0	R	Vendor ID Bit 2	Vendor ID Bit 2

Byte 7: Vendor ID (continued)

Bit	@Pup	Type	Name	Description
1	0	R	Vendor ID Bit 1	Vendor ID Bit 1
0	0	R	Vendor ID Bit 0	Vendor ID Bit 0

Byte 8: Control Register 8

Bit	@Pup	Type	Name	Description
7	0	R/W	CR1_PCIEX7	PCIEX[T/C75 CLKREQ#A Control 1 = PCIEX [T/C]5 stoppable by CLKREQ#B pin 0 = Free running
6	0	R/W	CR1_PCIEX6	PCIEX[T/C]6 CLKREQ#A Control 1 = PCIEX [T/C]4 stoppable by CLKREQ#B pin 0 = Free running
5	0	R/W	CR1_PCIEX5	PCIEX[T/C]5 CLKREQ#B Control 1 = PCIEX [T/C]5 stoppable by CLKREQ#B pin 0 = Free running
4	0	R/W	CR1_PCIEX4	PCIEX[T/C]4 CLKREQ#B Control 1 = PCIEX [T/C]4 stoppable by CLKREQ#B pin 0 = Free running
3	0	R/W	RESERVED	RESERVED, Set = 0
2	0	R/W	RESERVED	RESERVED, Set = 0
1	0	R/W	RESERVED	RESERVED, Set = 0
0	0	R/W	RESERVED	RESERVED, Set = 0

Byte 9: Control Register 9

Bit	@Pup	Type	Name	Description
7	0	R/W	DF3_N8	The DF3_N[8:0] configures CPU frequency for Dynamic Frequency. DOC[1:2] =11
6	0	R/W	DF2_N8	The DF2_N[8:0] configures CPU frequency for Dynamic Frequency. DOC[1:2] =10
5	0	R/W	DF1_N8	The DF1_N[8:0] configures CPU frequency for Dynamic Frequency. DOC[1:2] =01
4	0	R/W	RESERVED	RESERVED, Set = 0
3	0	R/W	RESERVED	RESERVED, Set = 0
2	1	R/W	SMSW_Bypass	Smooth switch Bypass 0 = Activate SMSW block 1 = Bypass and deactivate SMSW block.
1	0	R/W	SMSW_SEL	Smooth switch select 0 = Select CPU_PLL 1 = Select SRC_PLL
0	0	R/W	RESERVED	RESERVED, Set = 0

Byte 10: Control Register 10

Bit	@Pup	Type	Name	Description
7	0	R/W	DF1_N7	The DF1_N[8:0] configures CPU frequency for Dynamic Frequency. DOC[1:2] =01.
6	0	R/W	DF1_N6	
5	0	R/W	DF1_N5	
4	0	R/W	DF1_N4	
3	0	R/W	DF1_N3	
2	0	R/W	DF1_N2	
1	0	R/W	DF1_N1	
0	0	R/W	DF1_N0	

Byte 11: Control Register 11

Bit	@Pup	Type	Name	Description
7	0	R/W	DF2_N7	The DF2_N[8:0] configures CPU frequency for Dynamic Frequency. DOC[1:2] =10
6	0	R/W	DF2_N6	
5	0	R/W	DF2_N5	
4	0	R/W	DF2_N4	
3	0	R/W	DF2_N3	
2	0	R/W	DF2_N2	
1	0	R/W	DF2_N1	
0	0	R/W	DF2_N0	

Byte 12: Control Register 12

Bit	@Pup	Type	Name	Description
7	0	R/W	DF3_N7	The DF3_N[8:0] configures CPU frequency for Dynamic Frequency. DOC[1:2] =11
6	0	R/W	DF3_N6	
5	0	R/W	DF3_N5	
4	0	R/W	DF3_N4	
3	0	R/W	DF3_N3	
2	0	R/W	DF3_N2	
1	0	R/W	DF3_N1	
0	0	R/W	DF3_N0	

Byte 13: Control Register 13

Bit	@Pup	Type	Name	Description
7	0	R/W	Recovery_Frequency	This bit allows selection of the frequency setting to which the clock will be restored once the system is rebooted 0 = Use HW settings 1 = Recovery N[8:0]
6	0	R/W	Timer_SEL	Timer_SEL selects the WD reset function at SRESET pin when WD times out. 0 = Reset and Reload Recovery_Frequency 1 = Only Reset

Byte 13: Control Register 13

Bit	@Pup	Type	Name	Description
5	1	R/W	Time_Scale	Time_Scale allows selection of WD time scale 0 = 294 ms, 1 = 2.34 s
4	0	R/W	WD_Alarm	WD_Alarm is set to '1' when the watchdog times out. It is reset to '0' when the system clears the WD_TIMER time stamp
3	0	R/W	WD_TIMER2	Watchdog timer time stamp selection 000: Reserved (test mode) 001: 1 * Time_Scale 010: 2 * Time_Scale 011: 3 * Time_Scale 100: 4 * Time_Scale 101: 5 * Time_Scale 110: 6 * Time_Scale 111: 7 * Time_Scale
2	0	R/W	WD_TIMER1	
1	0	R/W	WD_TIMER0	
0	0	R/W	WD_EN	Watchdog timer enable. When the bit is asserted, Watchdog timer is triggered and time stamp of WD_Timer is loaded 0 = Disable, 1 = Enable

Byte 14: Control Register 14

Bit	@Pup	Type	Name	Description
7	0	R/W	CPU_DAF_N7	If Prog_CPU_EN is set, the values programmed in CPU_DAF_N[8:0] and CPU_DAF_M[6:0] will be used to determine the CPU output frequency. The setting of the FS_Override bit determines the frequency ratio for CPU and other output clocks. When it is cleared, the same frequency ratio stated in the Latched FS[E:A] register will be used. When it is set, the frequency ratio stated in the FSEL[3:0] register will be used
6	0	R/W	CPU_DAF_N6	
5	0	R/W	CPU_DAF_N5	
4	0	R/W	CPU_DAF_N4	
3	0	R/W	CPU_DAF_N3	
2	0	R/W	CPU_DAF_N2	
1	0	R/W	CPU_DAF_N1	
0	0	R/W	CPU_DAF_N0	

Byte 15: Control Register 15

Bit	@Pup	Type	Name	Description
7	0	R/W	CPU_DAF_N8	If Prog_CPU_EN is set, the values programmed in CPU_DAF_N[8:0] and CPU_DAF_M[6:0] will be used to determine the CPU output frequency. The setting of the FS_Override bit determines the frequency ratio for CPU and other output clocks. When it is cleared, the same frequency ratio stated in the Latched FS[E:A] register will be used. When it is set, the frequency ratio stated in the FSEL[3:0] register will be used.
6	0	R/W	CPU_DAF_M6	
5	0	R/W	CPU_DAF_M5	
4	0	R/W	CPU_DAF_M4	
3	0	R/W	CPU_DAF_M3	
2	0	R/W	CPU_DAF_M2	
1	0	R/W	CPU_DAF_M1	
0	0	R/W	CPU_DAF_M0	

Byte 16: Control Register 16

Bit	@Pup	Type	Name	Description
7	0	R/W	PCIE_DAF_N7	The PCIE_DAF_N[8:0] configures the PCIE frequency for Dial-A-Frequency
6	0	R/W	PCIE_DAF_N6	
5	0	R/W	PCIE_DAF_N5	
4	0	R/W	PCIE_DAF_N4	
3	0	R/W	PCIE_DAF_N3	
2	0	R/W	PCIE_DAF_N2	
1	0	R/W	PCIE_DAF_N1	
0	0	R/W	PCIE_DAF_N0	

Byte 17: Control Register 17

Bit	@Pup	Type	Name	Description
7	0	R/W	Recovery N7	Watchdog Recovery Bit
6	0	R/W	Recovery N6	Watchdog Recovery Bit
5	0	R/W	Recovery N5	Watchdog Recovery Bit
4	0	R/W	Recovery N4	Watchdog Recovery Bit
3	0	R/W	Recovery N3	Watchdog Recovery Bit
2	0	R/W	Recovery N2	Watchdog Recovery Bit
1	0	R/W	Recovery N1	Watchdog Recovery Bit
0	0	R/W	Recovery N0	Watchdog Recovery Bit

Byte 18: Control Register 18

Bit	@Pup	Type	Name	Description
7	0	R/W	PCIE_N8	PCI-E Dial-A-Frequency Bit N8
6	0	R/W	FS[D:A]	FS_Override 0 = Select operating frequency by FS(D:A) input pins 1 = Select operating frequency by FSEL_(3:0) settings
5	0	R/W	DF_EN	Dynamic Frequency for CPU Frequency Enable 0 = Disable, 1 = Enable
4	0	R/W	RESET_I_EN	RESET_I# Enable 0 = Disable, 1 = Enable
3	0	R/W	Prog_PCIE_EN	Programmable SRC Frequency Enable 0 = Disable, 1 = Enabled
2	0	R/W	Prog_CPU_EN	Programmable CPU Frequency Enable 0 = Disable, 1 = Enable
1	0	R/W	Watchdog Autorecovery	Watchdog Autorecovery Mode 0 = Disable (Manual), 1= Enable (Auto)
0	0	R/W	Recovery N8	Watchdog Recovery Bit

Table 5. Crystal Recommendations

Frequency (Fund)	Cut	Loading	Load Cap	Drive (max.)	Shunt Cap (max.)	Motional (max.)	Tolerance (max.)	Stability (max.)	Aging (max.)
14.31818 MHz	AT	Parallel	20 pF	0.1 mW	5 pF	0.016 pF	35 ppm	30 ppm	5 ppm

Crystal Recommendations

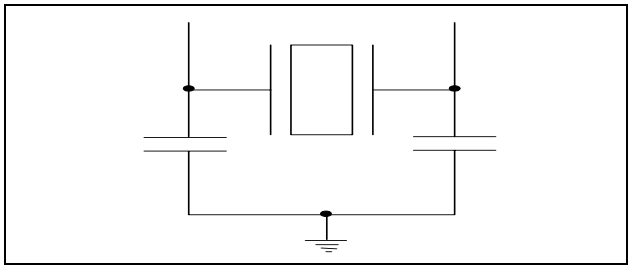
The CY28551 requires a parallel resonance crystal. Substituting a series resonance crystal will cause the CY28551 to operate at the wrong frequency and violate the ppm specification. For most applications there is a 300-ppm frequency shift between series and parallel crystals due to incorrect loading.

Crystal Loading

Crystal loading plays a critical role in achieving low ppm performance. To realize low ppm performance, the total capacitance the crystal will see must be considered to calculate the appropriate capacitive loading (CL).

Figure 1 shows a typical crystal configuration using the two trim capacitors. An important clarification for the following discussion is that the trim capacitors are in series with the crystal not parallel. It is a common misconception that load capacitors are in parallel with the crystal and should be approximately equal to the load capacitance of the crystal. This is not true.

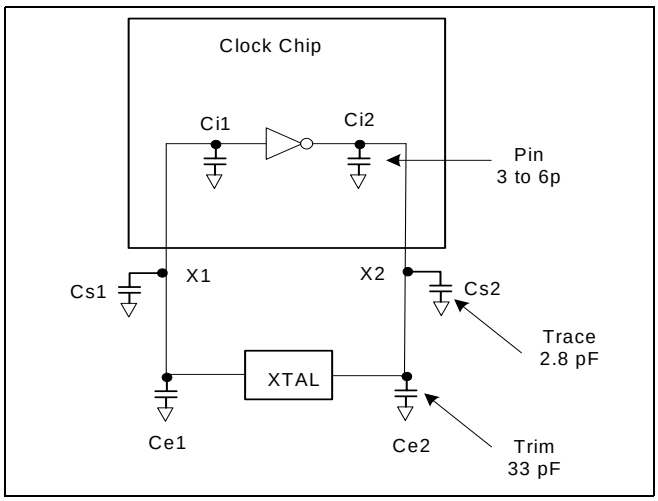
Figure 1. Crystal Capacitive Clarification



Calculating Load Capacitors

In addition to the standard external trim capacitors, trace capacitance and pin capacitance must also be considered to correctly calculate crystal loading. As mentioned previously, the capacitance on each side of the crystal is in series with the crystal. This means the total capacitance on each side of the crystal must be twice the specified crystal load capacitance (CL). While the capacitance on each side of the crystal is in series with the crystal, trim capacitors (Ce1,Ce2) should be calculated to provide equal capacitive loading on both sides.

Figure 2. Crystal Loading Example



Use the following formulas to calculate the trim capacitor values for Ce1 and Ce2.

Load Capacitance (each side)

$$Ce = 2 * CL - (Cs + Ci)$$

Total Capacitance (as seen by the crystal)

$$CLe = \frac{1}{\left(\frac{1}{Ce1 + Cs1 + Ci1} + \frac{1}{Ce2 + Cs2 + Ci2}\right)}$$

CL .....Crystal load capacitance

CLe ..... Actual loading seen by crystal using standard value trim capacitors

Ce ..... External trim capacitors

Cs .....Stray capacitance (terraced)

Ci ..... Internal capacitance (lead frame, bond wires etc.)

Multifunction Pin Selection

In the CY28551, some of the pins can provide different types of frequency, depending on the SEL[1:0] HW strapping pin setting, to support different chipset vendors. The configuration is shown as follows:

SEL[1:0]	LINK/DOT/SATA	SATA/PCIE	Platform
00	LINK	SATA	SIS
01	DOT	SATA	Intel W/Gfx
10	LINK	PCIEX	VIA
11	SATA	PCIEX	Intel

Dynamic Frequency

Dynamic Frequency – Dynamic Frequency (DF) is a technique used to increase CPU frequency or SRC frequency dynamically from any starting value. The user selects the starting point, either by HW, FSEL, or DAF, then enables DF. After that, DF will dynamically change as determined by DF-N registers and the M value of frequency table.

DF Pin – There are two pins to be used on Dynamic Frequency (DF). When used as DF, these two pins will map to four DF-N registers that correspond to different “N” values for Dynamic Frequency. Any time there is a change in DF, it should load the new value.

DOC[2:1]	DOC N register
00	Original Frequency
01	DF1_N
10	DF2_N
11	DF3_N

DF_EN bit – This bit enables the DF mode. By default, it is not set. When set, the operating frequency is determined by DF[2:0] pins. Default = 0, (No DF)

Dial-A-Frequency (CPU & PCIEX)

This feature allows users to overclock their systems by slowly stepping up the CPU or SRC frequency. When the programmable output frequency feature is enabled, the CPU and SRC frequencies are determined by the following equation:

Fcpu = G * N/M or Fcpu = G2 * N, where G2 = G/M.

‘N’ and ‘M’ are the values programmed in Programmable Frequency Select N-Value Register and M-Value Register, respectively. ‘G’ stands for the PLL Gear Constant, which is determined by the programmed value of FS[E:A]. See *Table 1* for the Gear Constant for each Frequency selection. The PCI Express only allows user control of the N register; the M value is fixed and documented in *Table 1*.

In this mode, the user writes the desired N and M value into the DAF I2C registers. The user cannot change only the M value and must change both the M and the N values at the same time, if they require a change to the M value. The user may change only the N value if required.

Associated Register Bits

CPU_DAF Enable – This bit enables CPU DAF mode. By default, it is not set. When set, the operating frequency is determined by the values entered into the CPU_DAF_N register. **Note:** The CPU_DAF_N and M register must contain valid values before CPU_DAF is set. Default = 0, (No DAF).

CPU_DAF_N – There are nine bits (for 512 values) to linearly change the CPU frequency (limited by VCO range). Default = 0, (0000). The allowable values for N are detailed in the frequency select table (*Table 1*).

CPU_DAF_M – There are 7 bits (for 128 values) to linearly change the CPU frequency (limited by VCO range). Default = 0. The allowable values for M are detailed in the frequency select table (*Table 1*).

SRC_DAF Enable – This bit enables SRC DAF mode. By default, it is not set. When set, the operating frequency is determined by the values entered into the SRC_DAF_N register. **Note:** The SRC_DAF_N register must contain valid values before SRC_DAF is set. Default = 0, (No DAF).

SRC_DAF_N – There are nine bits (for 512 values) to linearly change the CPU frequency (limited by VCO range). Default = 0, (0000). The allowable values for N are detailed in the frequency select table (*Table 1*).

Recovery – The recovery mechanism during CPU DAF, when the system locks up and the watchdog timer is enabled, is determined by the “Watchdog Recovery Mode” and “Watchdog Autorecovery Enable” bits. The possible recovery methods are: (A) Auto, (B) Manual (by Recovery N), (C) HW, and (D) No recovery, just send reset signal.

There is no recovery mode for SRC Dial-a-Frequency.

Software Frequency Select

This mode allows the user to select the CPU output frequencies using the Software Frequency select bits in the SMBUS register.

FSEL – There are four bits (for 16 combinations) to select predetermined CPU frequencies from a table. The table selections are detailed in *Table 1*.

FS_Override – This bit allows the CPU frequency to be selected from HW or FSEL settings. By default, this bit is not set and the CPU frequency is selected by HW. When this bit is set, the CPU frequency is selected by the FSEL bits. Default = 0.

Recovery – The recovery mechanism during FSEL when the system locks up is determined by the “Watchdog Recovery Mode” and “Watchdog Autorecovery Enable” bits. The only possible recovery method is to use Hardware Settings. Auto recovery or manual recovery can cause a wrong output frequency because the output divider may have changed with the selected CPU frequency and these recovery methods will not recover the original output divider setting.

Smooth Switching

The device contains one smooth switch circuit, which is shared by the CPU PLL and SRC PLL. The smooth switch circuit ensures that when the output frequency changes by overclocking, the transition from the old frequency to the new frequency is a slow, smooth transition containing no glitches. The rate of change of output frequency when using the smooth switch circuit is less than 1 MHz/0.667 µs. The frequency overshoot and undershoot will be less than 2%.

The smooth switch circuit can be assigned auto or manual mode. In auto mode, the clock generator will assign smooth switch automatically when the PLL will perform overclocking. For manual mode, the smooth switch circuit can be assigned to either PLL via SMBUS. By default the smooth switch circuit is set to auto mode. Either PLL can still be overclocked when it does not have control of the smooth switch circuit, but it is not guaranteed to transition to the new frequency without large frequency glitches.

Do not enable overclocking and change the N values of both PLLs in the same SMBUS block write and use smooth switch mechanism on spread spectrum on/off.

Watchdog Timer

The Watchdog timer is used in the system in conjunction with overclocking. It is used to provide a reset to a system that has hung up due to overclocking the CPU and the Front side bus. The watchdog is enabled by the user and if the system completes its checkpoints, the system will clear the timer. However, when the timer runs out, there will be a reset pulse generated on the SRESET# pin for 20 ms that is used to reset the system.

When the Watchdog is enabled (WD_EN = 1) the Watchdog timer will start counting down from a value of Watchdog_timer * time scale. If the Watchdog timer reaches 0 before the WD_EN bit is cleared then it will assert the SRESET# signal and set the Watchdog Alarm bit to '1'.

To use the watchdog, the SRESET# pin must be enabled by sampling SRESET_EN pin LOW by VTTPWRGD# assertion during system boot up.

If at any point during the Watchdog timer countdown the time stamp or Watchdog timer bits are changed, the timer will reset and start counting down from the new value.

After the Reset pulse, the watchdog will stay inactive until either:

1. A new time stamp or watchdog timer value is loaded.
2. The WD_EN bit is cleared and then set again.

Watchdog Register Bits

The following register bits are associated with the Watchdog timer:

Watchdog Enable – This bit (by default) is not set, which disables the Watchdog. When set, the Watchdog is enabled. Also, when there is a transition from LOW to HIGH, the timer reloads. Default = 0, disable

Watchdog Timer – There are three bits (for seven combinations) to select the timer value. Default = 000, the value '000' is a reserved test mode.

Watchdog Alarm – This bit is a flag and when it is set, it indicates that the timer has expired. This bit is not set by default. When the bit is set, the user is allowed to clear. Default = 0.

Watchdog Time Scale – This bit selects the multiplier. When this bit is not set, the multiplier will be 250 ms. When set (by default), the multiplier will be 3s. Default = 1

Watchdog Reset Mode – This selects the Watchdog Reset Mode. When this bit is not set (by default), the Watchdog will send a reset pulse and reload the recovery frequency depending on the Watchdog Recovery Mode setting. When set, it sends a reset pulse. Default = 0, Reset & Recover Frequency.

Watchdog Recovery Mode – This bit selects the location to recover from. One option is to recover from the HW settings (already stored in SMBUS registers for readback capability) and the second is to recover from a register called "Recovery N". Default = 0 (Recover from the HW setting)

Watchdog Autorecovery Enable – This bit is set by default and the recovered values are automatically written into the "Watchdog Recovery Register" and reloaded by the Watchdog function. When this bit is not set, the user is allowed to write to the "Watchdog Recovery Register". The value stored in the "Watchdog Recovery Register" will be used for recovery. Default = 1, Autorecovery.

Watchdog Recovery Register – This is a nine-bit register to store the watchdog N recovery value. This value can be written by the Autorecovery or User depending on the state of the "Watchdog Autorecovery Enable bit".

Watchdog Recovery Modes

There are three operating modes that require Watchdog recovery. The modes are Dial-A-Frequency (DAF), Dynamic Clocking (DF), or Frequency Select. There are four different recovery modes; the following sections list the operating mode and the recovery mode associated with it.

Recover to Hardware M, N, O

When this recovery mode is selected, in the event of a Watchdog timeout, the original M, N, and O values that were latched by the HW FSEL pins at chip boot-up will be reloaded.

Autorecovery

When this recovery mode is selected, in the event of a Watchdog timeout, the M and N values stored in the Recovery M and N registers will be reloaded. The current values of M and N will be latched into the internal recovery M and N registers by the WD_EN bit being set.

Manual Recovery

When this recovery mode is selected, in the event of a Watchdog timeout, the N value as programmed by the user in the N recovery register, and the M value that is stored in the Recovery M register (not accessible by the user), will be restored. The current M value will be latched to M recovery register by the WD_EN bit being set.

No Recovery

If no recovery mode is selected, in the event of a Watchdog time out, the device will assert the SRESET# and keep the current values of M and N

Software Reset

Software reset is a reset function that is used to send out a pulse from the SRESET# pin. It is controlled by the SW_RESET enable register bit. Upon completion of the byte/word/block write in which the SW_RESET bit was set, the device will send a RESET pulse on the SRESET# pin. The duration of the SRESET# pulse will be the same as the duration of the SRESET# pulse after a Watchdog timer time out.

After the SRESET# pulse is asserted the SW_RESET bit will be automatically cleared by the device.

PD Clarification

The VTT_PWRGD#/PD pin is a dual-function pin. During initial power up, the pin functions as VTT_PWRGD#. Once

VTT_PWRGD# has been sampled low by the clock chip, the pin assumes PD functionality. The PD pin is an asynchronous active HIGH input used to shut off all clocks cleanly prior to shutting off power to the device. This signal must be synchronized internal to the device prior to powering down the clock synthesizer. PD is also an asynchronous input for powering up the system. When PD is asserted HIGH, all clocks must be driven to a LOW value and held prior to turning off the VCOs and the crystal oscillator

PD Assertion

When PD is sampled HIGH by two consecutive rising edges of CPUC, all single-ended outputs must be held LOW on their next HIGH-to-LOW transition and differential clocks must be held HIGH or tri-stated (depending on the state of the control register drive mode bit) on the next “Diff clock#” HIGH-to-LOW transition within 4 clock periods. When the SMBus PD drive mode bit corresponding to the differential (CPU, SRC, and DOT) clock output of interest is programmed to '0', the clock output must be held with “Diff clock” pin driven HIGH at 2 x Iref, and “Diff clock#” tri-state. If the control register PD drive mode bit corresponding to the output of interest is programmed to ‘1’, then both the “Diff clock” and the “Diff clock#” are tri-state. Note *Figure 3* shows CPUT = 133 MHz and PD drive mode = '1' for all differential outputs. This diagram and description is applicable to valid CPU frequencies 100, 133, 166, and 200 MHz. In the event that PD mode is desired as the initial power-on state, PD must be asserted HIGH in less than 10 µs after asserting VTT_PWRGD#.

PD Deassertion

The power-up latency must be less than 1.8 ms. This is the time from the deassertion of the PD pin or the ramping of the power supply until the time that stable clocks are output from the clock chip. All differential outputs stopped in a tri-state condition resulting from power down must be driven HIGH in less than 300 µs of PD deassertion to a voltage greater than 200 mV. After the clock chip's internal PLL is powered up and locked, all outputs are to be enabled within a few clock cycles of each other. *Figure 4* is an example showing the relationship of clocks coming up. Unfortunately, we can not show all possible combinations; designers need to ensure that from the first active clock output to the last takes no more than two full PCI clock cycles.

Figure 3. PD Assertion Timing Waveform

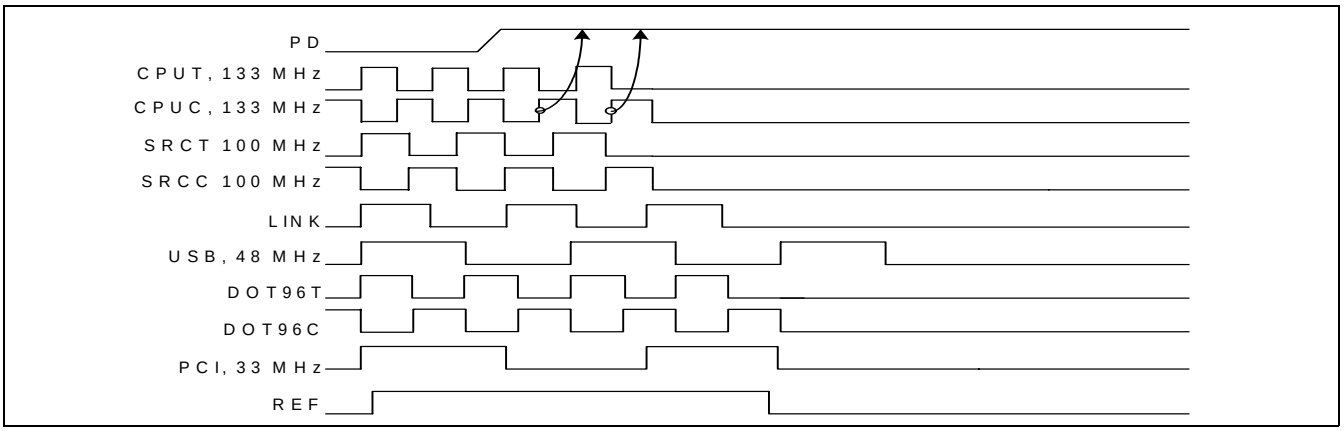
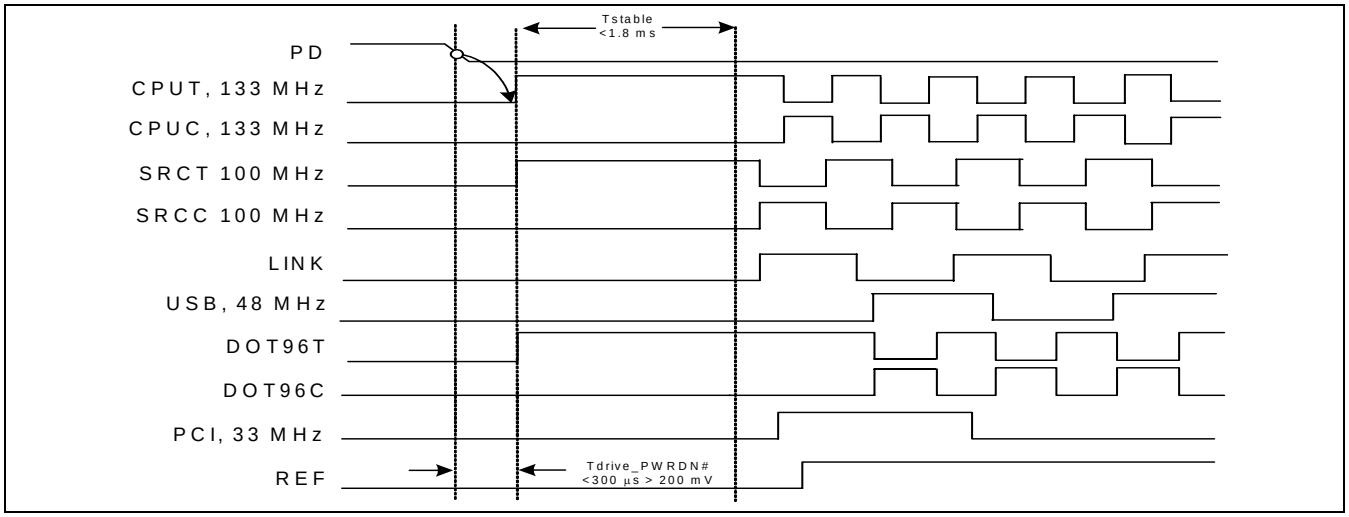


Figure 4. PD Deassertion Timing Waveform



CPU_STP# Clarification

The CPU_STP# signal is an active LOW input used for cleanly stopping and starting the CPU outputs while the rest of the clock generator continues to function. Note that the assertion and deassertion of this signal is absolutely asynchronous.

CPU_STP# Assertion

The CPU_STP# signal is an active LOW input used for synchronous stopping and starting of the CPU output clocks while the rest of the clock generator continues to function. When the CPU_STP# pin is asserted, all CPU outputs that are set with the SMBus configuration to be stoppable via assertion of CPU_STP# will be stopped after being sampled by 2 to 6 rising edges of the internal CPUC clock. The final state of the stopped CPU clock is LOW due to tri-state; both CPUT and CPUC outputs will not be driven.

Figure 5. CPU_STP# Assertion Timing Waveform

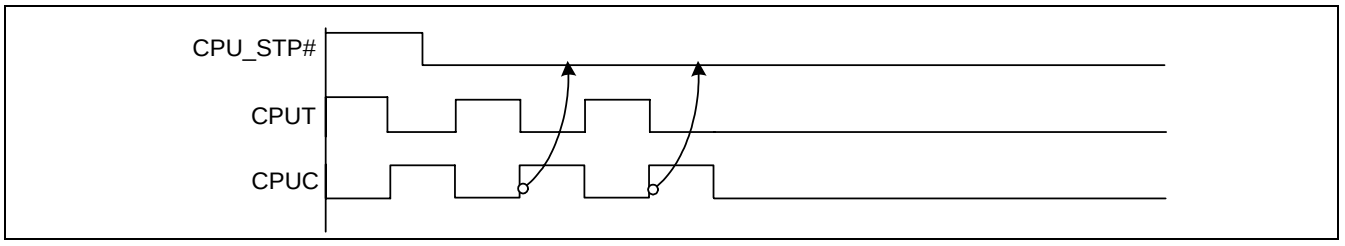
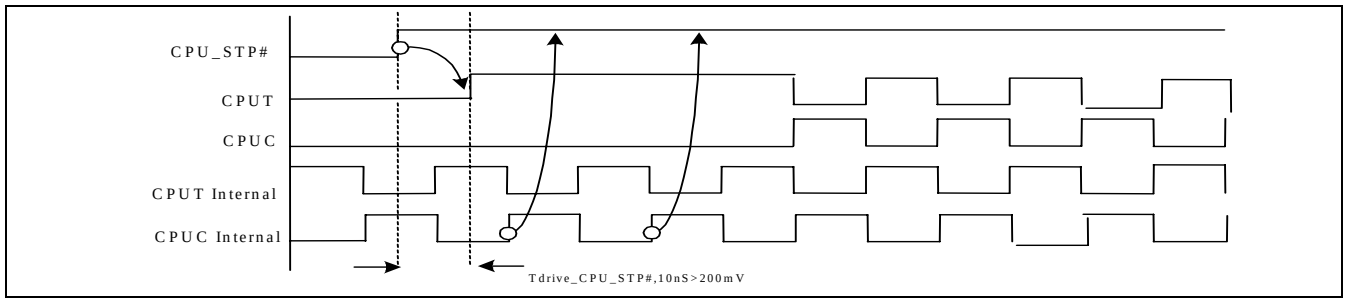


Figure 6. CPU_STP# Deassertion



CPU_STP# Deassertion

The deassertion of the CPU_STP# signal will cause all CPU outputs that were stopped to resume normal operation in a synchronous manner, synchronous manner meaning that no

short or stretched clock pulses will be produced when the clock resumes. The maximum latency from the deassertion to active outputs is between 2 and 6 CPU clock periods (2 clocks are shown). If the control register tri-state bit corresponding to the output of interest is programmed to '1', then the stopped CPU outputs will be driven HIGH within 10 ns of CPU_Stop# deassertion to a voltage greater than 200 mV.

PCI_STP# Clarification

The PCI_STP# signal is an active LOW input used for cleanly stopping and starting the PCI and PCIEX outputs while the rest of the clock generator continues to function. The PCIF and PCIEX clocks are special in that they can be programmed to ignore PCI_STP# by setting the register bit corresponding to the output of interest to free running. Outputs set to free running will ignore the PCI_STP# pin.

PCI_STP# Assertion

The impact of asserting the PCI_STP# signal is as follows. The clock chip is to sample the PCI_STP# signal on a rising edge of PCIF clock. After detecting the PCI_STP# assertion LOW, all PCI and stoppable PCIF clocks will latch LOW on their next HIGH-to-LOW transition. After the PCI clocks are latched LOW, the stoppable PCIEX clocks will latch to LOW due to tri-state, as shown in *Figure 7*. The one PCI clock latency shown is critical to system functionality; any violation of this may result in system failure. The Tsu_pci_stp# is the setup time required by the clock generator to correctly sample the PCI_STP# assertion. This time is 10 ns minimum.

PCI_STP# Deassertion

The deassertion of the PCI_STP# signal functions as follows. The deassertion of the PCI_STP# signal is to be sampled on the rising edge of the PCIF free running clock domain. After detecting PCI_STP# deassertion, all PCI, stoppable PCIF and stoppable PCIEX clocks will resume in a glitch-free manner. The PCI and PCIEX clock resume latency should exactly match the 1 PCI clock latency required for PCI_STP# entry. The stoppable PCIEX clocks must be driven HIGH within 15 ns of PCI_STP# deassertion. *Figure 8* shows the appropriate relationship. The Tsu_cpu_stp# is the setup time required by the clock generator to correctly sample the PCI_STP# deassertion. This time is 10 ns minimum.

Figure 7. PCI_STP# Assertion

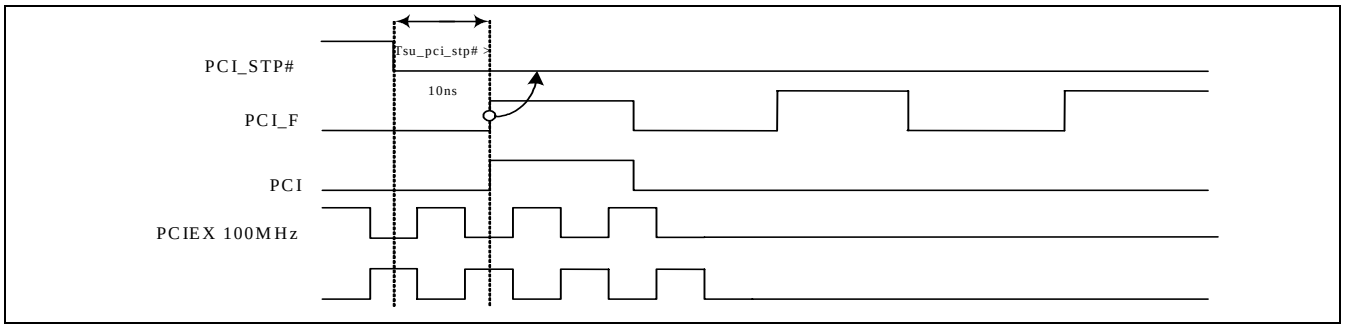
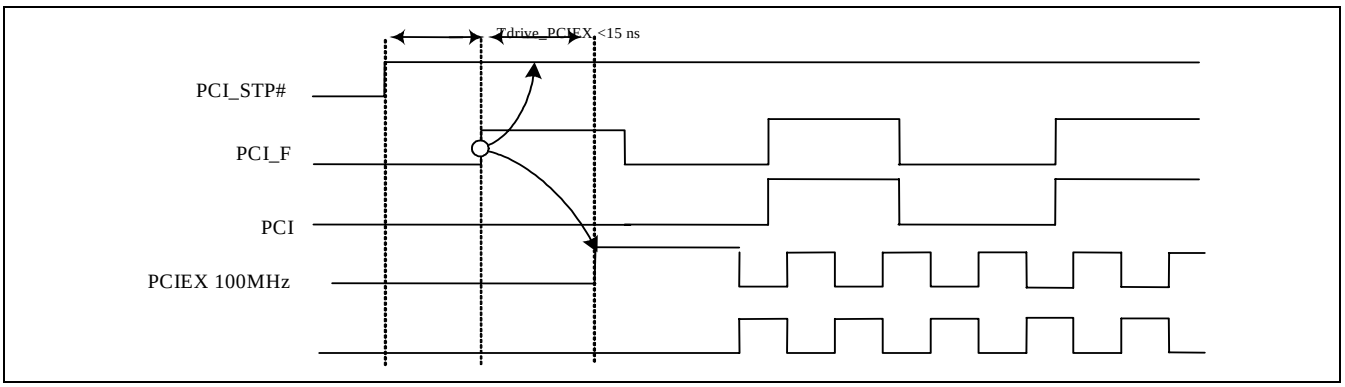


Figure 8. PCI_STP# Deassertion



CLKREQ# Clarification

The CLKREQ# signals are active LOW inputs used to cleanly stop and start selected SRC outputs. The outputs controlled by CLKREQ# are determined by the settings in register bytes 10 and 11. The CLKREQ# signal is a debounced signal in that its state must remain unchanged during two consecutive rising edges of DIFC to be recognized as a valid assertion or deassertion. (The assertion and deassertion of this signal is absolutely asynchronous.)

CLKREQ# Assertion

All differential outputs that were stopped will resume normal operation in a glitch-free manner. The maximum latency from the deassertion to active outputs is between 2 and 6 PCIEX clock periods (2 clocks are shown) with all CLKREQ# outputs resuming simultaneously. If the CLKREQ# drive mode is tri-state, all stopped PCIEX outputs must be driven HIGH within 10 ns of CLKREQ# deassertion to a voltage greater than 200 mV.

CLKREQ# Deassertion

The impact of asserting the CLKREQ# pins is that all DIF outputs that are set in the control registers to stoppable via assertion of CLKREQ# are to be stopped after their next transition. When the control register CLKREQ# drive mode bit is programmed to '0', the final state of all stopped PCIEX signals is PCIEXT clock = HIGH and PCIEXC = LOW. There will be no change to the output drive current values. SRCT will be driven HIGH with a current value equal 6 x Iref. When the control register CLKREQ# drive mode bit is programmed to '1', the final state of all stopped DIF signals is LOW; both PCIEXT clock and PCIEXC clock outputs will not be driven.

Figure 9. CLKREQ# Deassertion

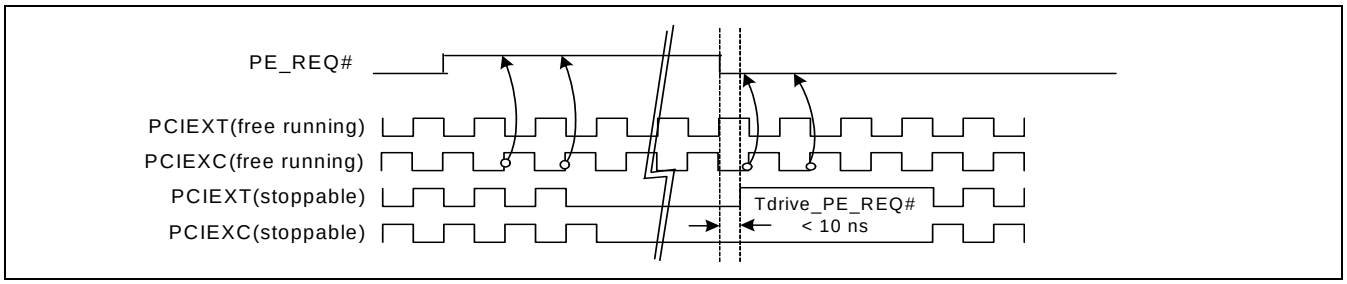


Figure 10. VTT_PWRGD# Timing Diagram

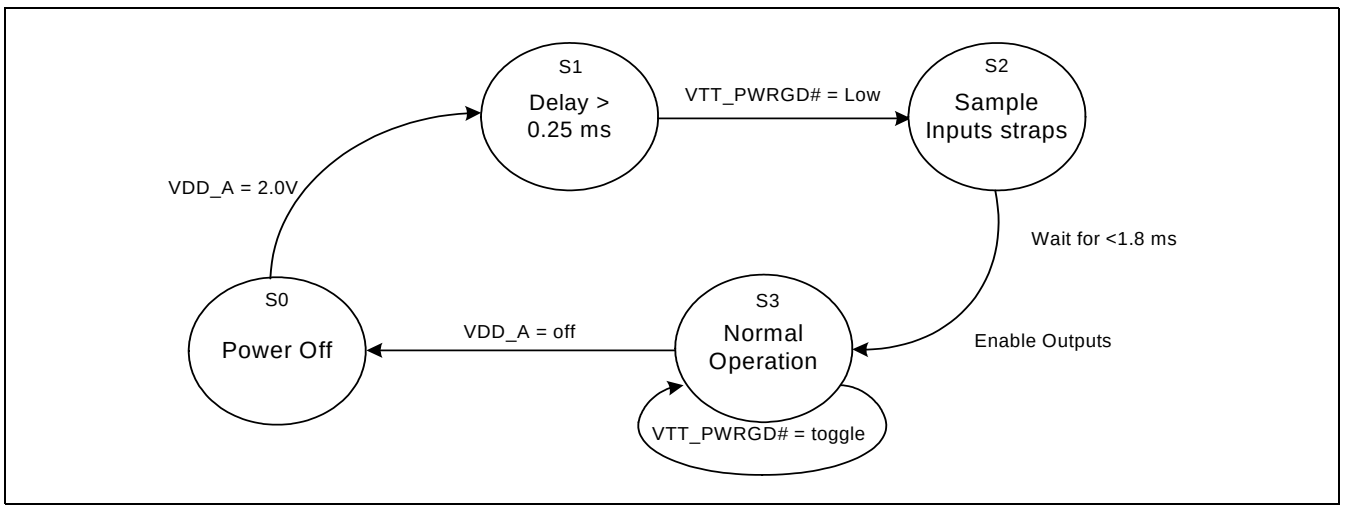
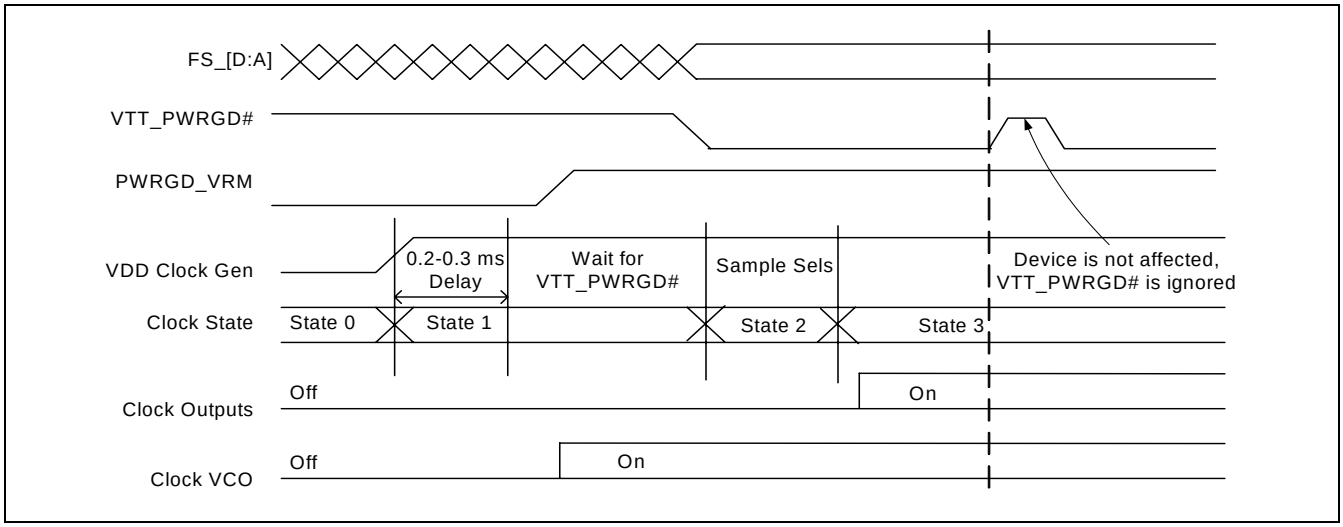


Figure 11. VTT_PWRGD# Timing Diagram



Absolute Maximum Conditions

Parameter	Description	Condition	Min.	Max.	Unit
V_{DD}	Core Supply Voltage		–0.5	4.6	V
V_{DD_A}	Analog Supply Voltage		–0.5	4.6	V
V_{IN}	Input Voltage	Relative to V_{SS}	–0.5	V_{DD} + 0.5	VDC
T_S	Temperature, Storage	Non-functional	–65	150	°C
T_A	Temperature, Operating Ambient	Functional	0	70	°C
T_J	Temperature, Junction	Functional	–	150	°C
$Ø_{JC}$	Dissipation, Junction to Case	Mil-STD-883E Method 1012.1	–	20	°C/W
$Ø_{JA}$	Dissipation, Junction to Ambient	JEDEC (JESD 51)	–	60	°C/W
ESD_{HBM}	ESD Protection (Human Body Model)	MIL-STD-883, Method 3015	2000	–	V
UL-94	Flammability Rating	At 1/8 in.	V–0		
MSL	Moisture Sensitivity Level		1		

Multiple Supplies: The voltage on any input or I/O pin cannot exceed the power pin during power-up. Power supply sequencing is NOT required.

DC Electrical Specifications

Parameter	Description	Condition	Min.	Max.	Unit
All V_{DD}s	3.3V Operating Voltage	3.3 ± 5%	3.135	3.465	V
V_{ILI2C}	Input Low Voltage	SDATA, SCLK	–	1.0	V
V_{IHI2C}	Input High Voltage	SDATA, SCLK	2.2	–	V
V_{IL_FS}	FS_[A:D] Input Low Voltage		V_{SS} – 0.3	0.35	V
V_{IH_FS}	FS_[A:D] Input High Voltage		0.7	V_{DD} + 0.5	V
V_{IL}	3.3V Input Low Voltage		V_{SS} – 0.3	0.8	V
V_{IH}	3.3V Input High Voltage		2.0	V_{DD} + 0.3	V
I_{IL}	Input Low Leakage Current	Except internal pull-up resistors, $0 < V_{IN} < V_{DD}$	–5	–	µA
I_{IH}	Input High Leakage Current	Except internal pull-down resistors, $0 < V_{IN} < V_{DD}$	–	5	µA
V_{OL}	3.3V Output Low Voltage	I_{OL} = 1 mA	–	0.4	V
V_{OH}	3.3V Output High Voltage	I_{OH} = –1 mA	2.4	–	V
I_{OZ}	High-impedance Output Current		–10	10	µA
C_{IN}	Input Pin Capacitance		3	5	pF
C_{OUT}	Output Pin Capacitance		3	5	pF
L_{IN}	Pin Inductance		–	7	nH
V_{XIH}	Xin High Voltage		$0.7V_{DD}$	V_{DD}	V
V_{XIL}	Xin Low Voltage		0	$0.3V_{DD}$	V
$I_{DD3.3V}$	Dynamic Supply Current	At max. load and freq. per *Figure 14*	–	500	mA
$I_{PT3.3V}$	Power-down Supply Current	PD asserted, Outputs Tri-state	–	12	mA

AC Electrical Specifications

Parameter	Description	Condition	Min.	Max.	Unit
Crystal					
T_{DC}	XIN Duty Cycle	The device will operate reliably with input duty cycles up to 30/70 but the REF clock duty cycle will not be within specification	47.5	52.5	%
T_{PERIOD}	XIN Period	When XIN is driven from an external clock source	69.841	71.0	ns
T_R/T_F	XIN Rise and Fall Times	Measured between $0.3V_{DD}$ and $0.7V_{DD}$	–	10.0	ns
T_{CCJ}	XIN Cycle to Cycle Jitter	As an average over 1-µs duration	–	500	ps
L_{ACC}	Long-term Accuracy	Over 150 ms	–	30	ppm
CPU at 0.7V (SSC refers to –0.5% spread spectrum)					
T_{DC}	CPUT and CPUC Duty Cycle	Measured at crossing point V_{OX} @ 0.1s	45	55	%
T_{PERIOD}	100 MHz CPUT and CPUC Period	Measured at crossing point V_{OX} @ 0.1s	9.99900	10.0100	ns
T_{PERIOD}	133 MHz CPUT and CPUC Period	Measured at crossing point V_{OX} @ 0.1s	7.49925	7.50075	ns
T_{PERIOD}	166 MHz CPUT and CPUC Period	Measured at crossing point V_{OX} @ 0.1s	5.99940	6.00060	ns
T_{PERIOD}	200 MHz CPUT and CPUC Period	Measured at crossing point V_{OX} @ 0.1s	4.99950	5.00050	ns
T_{PERIOD}	266 MHz CPUT and CPUC Period	Measured at crossing point V_{OX} @ 0.1s	3.74963	3.75038	ns
T_{PERIOD}	333 MHz CPUT and CPUC Period	Measured at crossing point V_{OX} @ 0.1s	2.99970	3.00030	ns
T_{PERIOD}	400 MHz CPUT and CPUC Period	Measured at crossing point V_{OX} @ 0.1s	2.49975	2.50025	ns
$T_{PERIODSS}$	100 MHz CPUT and CPUC Period, SSC	Measured at crossing point V_{OX} @ 0.1s	9.99900	10.0100	ns
$T_{PERIODSS}$	133 MHz CPUT and CPUC Period, SSC	Measured at crossing point V_{OX} @ 0.1s	7.49925	7.50075	ns
$T_{PERIODSS}$	166 MHz CPUT and CPUC Period, SSC	Measured at crossing point V_{OX} @ 0.1s	5.99940	6.00060	ns
$T_{PERIODSS}$	200 MHz CPUT and CPUC Period, SSC	Measured at crossing point V_{OX} @ 0.1s	4.99950	5.00050	ns
$T_{PERIODSS}$	266 MHz CPUT and CPUC Period, SSC	Measured at crossing point V_{OX} @ 0.1s	3.74963	3.75038	ns
$T_{PERIODSS}$	333 MHz CPUT and CPUC Period, SSC	Measured at crossing point V_{OX} @ 0.1s	2.99970	3.00030	ns
$T_{PERIODSS}$	400 MHz CPUT and CPUC Period, SSC	Measured at crossing point V_{OX} @ 0.1s	2.49975	2.50025	ns
$T_{PERIODAbs}$	100 MHz CPUT and CPUC Absolute period	Measured at crossing point V_{OX} @ 1 clock	9.91400	10.0860	ns
$T_{PERIODAbs}$	133 MHz CPUT and CPUC Absolute Period	Measured at crossing point V_{OX} @ 1 clock	7.41425	7.58575	ns
$T_{PERIODAbs}$	166 MHz CPUT and CPUC Absolute Period	Measured at crossing point V_{OX} @ 1 clock	5.91440	6.08560	ns
$T_{PERIODAbs}$	200 MHz CPUT and CPUC Absolute Period	Measured at crossing point V_{OX} @ 1 clock	4.91450	5.08550	ns
$T_{PERIODAbs}$	266 MHz CPUT and CPUC Absolute Period	Measured at crossing point V_{OX} @ 1 clock	3.66463	3.83538	ns
$T_{PERIODAbs}$	333 MHz CPUT and CPUC Absolute Period	Measured at crossing point V_{OX} @ 1 clock	2.91470	3.08530	ns
$T_{PERIODAbs}$	400 MHz CPUT and CPUC Absolute Period	Measured at crossing point V_{OX} @ 1 clock	2.41475	2.58525	ns
$T_{PERIODSSAbs}$	100- MHz CPUT and CPUC Absolute Period, SSC	Measured at crossing point V_{OX} @ 1 clock	9.91400	10.1363	ns
$T_{PERIODSSAbs}$	133 MHz CPUT and CPUC Absolute Period, SSC	Measured at crossing point V_{OX} @ 1 clock	7.41425	7.62345	ns
$T_{PERIODSSAbs}$	166 MHz CPUT and CPUC Absolute Period, SSC	Measured at crossing point V_{OX} @ 1 clock	5.91440	6.11576	ns

AC Electrical Specifications (continued)

Parameter	Description	Condition	Min.	Max.	Unit
$T_{PERIODSSAbs}$	200 MHz CPUT and CPUC Absolute Period, SSC	Measured at crossing point V_{OX} @ 1 clock	4.91450	5.11063	ns
$T_{PERIODSSAbs}$	266 MHz CPUT and CPUC Absolute Period, SSC	Measured at crossing point V_{OX} @ 1 clock	3.66463	3.85422	ns
$T_{PERIODSSAbs}$	333 MHz CPUT and CPUC Absolute Period, SSC	Measured at crossing point V_{OX} @ 1 clock	2.91470	3.10038	ns
$T_{PERIODSSAbs}$	400 MHz CPUT and CPUC Absolute Period, SSC	Measured at crossing point V_{OX} @ 1 clock	2.41475	2.59782	ns
T_{SKEW}	CPU0 to CPU1	Measured at crossing point V_{OX}	–	100	ps
T_{CCJ}	CPUT/C Cycle to Cycle	Measured at crossing point V_{OX}	–	85	ps
L_{ACC}	Long Term Accuracy	Measured at crossing point V_{OX}	–	100	ppm
T_R/T_F	CPUT and CPUC Rise and Fall Times	Measured differentially from ±150 mV	2.5	8	V/ns
T_{RFM}	Rise/Fall Matching	Measured single-endedly from ±75 mV	–	*20*	%
$V_{\_max}$	Max Output Voltage	Math averages *Figure 14*	–	1.15	V
$V_{\_min}$	Min Output Voltage	Math averages *Figure 14*	–0.3	–	V
V_{OX}	Crossing Point Voltage at 0.7V Swing		300	550	mV
CPU at 3.3V (SSC refers to –0.5% spread spectrum)					
T_R	Output Rise Edge Rate	Measured @ K8 test load using VOCM ± 400 mV, 0.850V to 1.650V	2	7	V/ns
T_F	Output Fall Edge Rate	Measured @ K8test load using VOCM ± 400 mV, 1.650V to 0.850V	2	7	V/ns
V_{DIFF}	Differential Voltage	Measured @ K8 test load (single-ended)	0.4	2.3	V
Δ_{DIFF}	Change in VDIFF_DC Magnitude	Measured @ K8 test load (single-ended)	–150	150	mV
V_{CM}	Common Mode Voltage	Measured @ K8 test load (single-ended)	1.05	1.45	V
ΔV_{CM}	Change in VCM	Measured @ K8 test load (single-ended)	–200	200	mV
T_{DC}	Duty Cycle	Measured at V_{OX}	45	53	%
T_{CYC}	Jitter, Cycle to Cycle	Measured at V_{OX}	0	200	ps
T_{ACCUM}	Jitter, Accumulated	Measured at V_{OX}	–1000	1000	ps
PCIEX					
T_{DC}	PCIEXT and PCIEXC Duty Cycle	Measured at crossing point V_{OX}	45	55	%
T_{PERIOD}	100 MHz PCIEXT and PCIEXC Period	Measured at crossing point V_{OX} @ 0.1s	9.99900	10.0010	ns
$T_{PERIODSS}$	100 MHz PCIEXT and PCIEXC Period, SSC	Measured at crossing point V_{OX} @ 0.1s	9.99900	10.0010	ns
$T_{PERIODAbs}$	100 MHz PCIEXT and PCIEXC Absolute Period	Measured at crossing point V_{OX} @ 1 clock	9.87400	10.1260	ns
$T_{PERIODSSAbs}$	100 MHz PCIEXT and PCIEXC Absolute Period, SSC	Measured at crossing point V_{OX} @ 1 clock	9.87400	10.1763	ns
T_{SKEW}	Any PCIEXT/C to PCIEXT/C Clock Skew	Measured at crossing point V_{OX}	–	250	ps
T_{CCJ}	PCIEXT/C Cycle to Cycle Jitter	Measured at crossing point V_{OX}	–	125	ps
L_{ACC}	PCIEXT/C Long Term Accuracy	Measured at crossing point V_{OX}	–	100	ppm
T_R/T_F	PCIEXT and PCIEXC Rise and Fall Times	Measured differentially from ±150 mV	2.5	8	V/ns
T_{RFM}	Rise/Fall Matching	Measured single-endedly from ±75 mV	–	20	%
$V_{\_max}$	Max Output Voltage	Math averages, see *Figure 14*	–	1.15	V
$V_{\_min}$	Min Output Voltage	Math averages, see *Figure 14*	–0.3	–	V

AC Electrical Specifications (continued)

Parameter	Description	Condition	Min.	Max.	Unit
V_{OX}	Crossing Point Voltage at 0.7V Swing		300	550	mV
DOT					
T_{DC}	DOT96T and DOT96C Duty Cycle	Measured at crossing point V_{OX}	45	55	%
T_{PERIOD}	DOT96T and DOT96C Period	Measured at crossing point V_{OX} @ 0.1s	10.4156	10.4177	ns
$T_{PERIODAbs}$	DOT96T and DOT96C Absolute Period	Measured at crossing point V_{OX} @ 0.1s	10.1656	10.6677	ns
T_{CCJ}	DOT96T/C Cycle to Cycle Jitter	Measured at crossing point V_{OX}	–	*250*	ps
L_{ACC}	DOT96T/C Long Term Accuracy	Measured at crossing point V_{OX}	–	300	ppm
T_{LTJ}	Long Term Jitter	Measurement taken from cross point V_{OX} @ 10 µs	–	700	ps
T_R/T_F	DOT96T and DOT96C Rise and Fall Time	Measured differentially from ±150 mV	2.5	8	V/ns
T_{RFM}	Rise/Fall Matching	Measured single-endedly from ±75 mV	–	20	%
V_{max}	Max Output Voltage	Math averages, *see Figure 14*	–	1.15	V
V_{min}	Min Output Voltage	Math averages, see *Figure 14*	–0.3	–	V
V_{OX}	Crossing Point Voltage at 0.7V Swing		300	550	mV
LINK - 133 MHz					
T_{DC}	LINK (133 MHz) Duty Cycle	Measurement at 1.5V	45	55	%
T_{PERIOD}	Spread Disabled LINK (133 MHz) Period	Measurement at 1.5V	7.50	7.56	ns
$T_{PERIODSS}$	Spread Enabled LINK (133 MHz) Period, SSC	Measurement at 1.5V	7.50	7.56	ns
T_R/T_F	LINK (133 MHz) Rising and Falling Edge Rate	Measured between 0.8V and 2.0V	1.0	4.0	V/ns
T_{CCJ}	LINK (133 MHz) Cycle-to-cycle Jitter	Measurement at 1.5V	–	250	ps
T_{SKEW}	Any LINK Clock Skew	Measurement at 1.5V	–	175	ps
L_{ACC}	LINK (133 MHz) Long Term Accuracy	Measured at crossing point V_{OX}	–	300	ppm
LINK - 66 MHz					
T_{DC}	LINK (66 MHz) Duty Cycle	Measurement at 1.5V	45	55	%
T_{PERIOD}	Spread Disabled LINK (66 MHz) Period	Measurement at 1.5V	14.9955	15.0045	ns
$T_{PERIODSS}$	Spread Enabled LINK (66 MHz) Period, SSC	Measurement at 1.5V	14.9955	15.0045	ns
T_R/T_F	LINK (66 MHz) Rising and Falling Edge Rate	Measured between 0.8V and 2.0V	1.0	4.0	V/ns
T_{CCJ}	LINK (66 MHz) Cycle-to-cycle Jitter	Measurement at 1.5V	–	250	ps
T_{SKEW}	Any LINK Clock Skew	Measurement at 1.5V	–	175	ps
L_{ACC}	LINK (66 MHz) Long Term Accuracy	Measured at crossing point V_{OX}	–	300	ppm
PCI					
T_{DC}	PCI Duty Cycle	Measurement at 1.5V	45	55	%
T_{PERIOD}	Spread Disabled PCIF/PCI Period	Measurement at 1.5V	29.99100	30.00900	ns
$T_{PERIODSS}$	Spread Enabled PCIF/PCI Period, SSC	Measurement at 1.5V	29.9910	30.15980	ns
$T_{PERIODAbs}$	Spread Disabled PCIF/PCI Period	Measurement at 1.5V	29.49100	30.50900	ns
$T_{PERIODSSAbs}$	Spread Enabled PCIF/PCI Period, SSC	Measurement at 1.5V	29.49100	30.65980	ns
T_{HIGH}	PCIF and PCI High Time	Measurement at 2.4V	12.0	–	ns
T_{LOW}	PCIF and PCI Low Time	Measurement at 0.4V	12.0	–	ns
T_R/T_F	PCIF and PCI Rising and Falling Edge Rate	Measured between 0.8V and 2.0V	1.0	4.0	V/ns

AC Electrical Specifications (continued)

Parameter	Description	Condition	Min.	Max.	Unit
T_{SKEW}	Any PCI Clock to Any PCI Clock Skew	Measurement at 1.5V	–	250	ps
T_{CCJ}	PCIF and PCI Cycle to Cycle Jitter	Measurement at 1.5V	–	*500*	ps
USB					
T_{DC}	Duty Cycle	Measurement at 1.5V In High Drive mode	45	55	%
T_{PERIOD}	Period	Measurement at 1.5V	20.83125	20.83542	ns
$T_{PERIODAbs}$	Absolute Period	Measurement at 1.5V	20.48125	21.18542	ns
T_{HIGH}	USB High Time	Measurement at 2.4V	8.094	*10.5*	ns
T_{LOW}	USB Low Time	Measurement at 0.4V	7.694	*10.5*	ns
T_R/T_F	Rising and Falling Edge Rate	Measured between 0.8V and 2.0V	1.0	2.0	V/ns
T_{CCJ}	Cycle-to-cycle Jitter	Measurement at 1.5V	–	350	ps
L_{ACC}	48M Long Term Accuracy	Measured at crossing point V_{OX}	–	100	ppm
T_{LTJ}	Long Term Jitter	Measurement taken from cross point V_{OX} @ 1 µs	–	1.0	ns
24M					
T_{DC}	Duty Cycle	Measurement at 1.5V In High Drive mode	45	55	%
T_{PERIOD}	Period	Measurement at 1.5V	41.6646	41.6688	ns
T_{HIGH}	USB High Time	Measurement at 2.4V	18.8323	*18.8323*	ns
T_{LOW}	USB Low Time	Measurement at 0.4V	18.8323	*18.8323*	ns
T_R/T_F	Rising and Falling Edge Rate	Measured between 0.8V and 2.0V	1.0	4.0	V/ns
T_{CCJ}	Cycle-to-cycle Jitter	Measurement at 1.5V	–	350	ps
L_{ACC}	48M Long Term Accuracy	Measured at crossing point V_{OX}	–	100	ppm
T_{LTJ}	Long Term jitter	Measurement taken from cross point V_{OX} @ 1 µs	–	1.0	ns
REF					
T_{DC}	REF Duty Cycle	Measurement at 1.5V	45	55	%
T_{PERIOD}	REF Period	Measurement at 1.5V	69.8203	69.8622	ns
$T_{PERIODAbs}$	REF Absolute Period	Measurement at 1.5V	68.82033	70.86224	ns
T_R/T_F	REF Rise and Fall Times Edge rate	Measured between 0.8V and 2.0V	1.0	4.0	V/ns
T_{CCJ}	REF Cycle-to-cycle Jitter	Measurement at 1.5V	–	1000	ps
T_{SKEW}	REF Clock to REF Clock	Measurement at 1.5V	–	500	ps
L_{ACC}	Long Term Accuracy	Measurement at 1.5V	–	300	ppm
ENABLE/DISABLE and SET-UP					
T_{STABLE}	Clock Stabilization from Power-up		–	1.8	ms

Test and Measurement Set-up

For PCI/USB and 24M Single-ended Signals and Reference

Figure 12 and *Figure 13* show the test load configurations for the single-ended PCI, USB, 24M, and REF output signals

Figure 12. Single-ended Load Configuration.

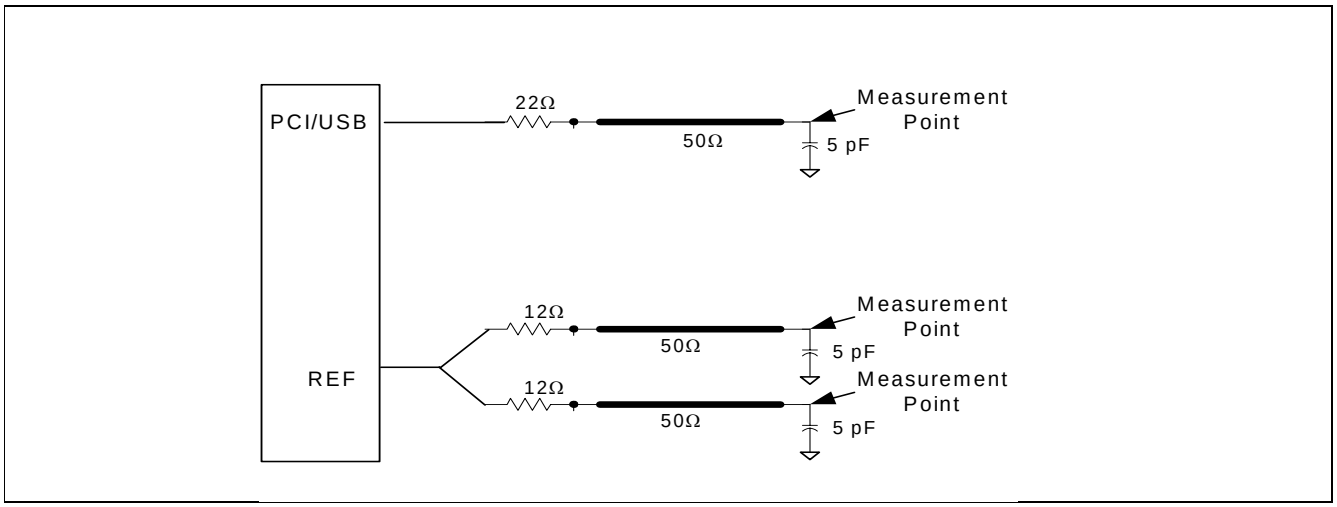
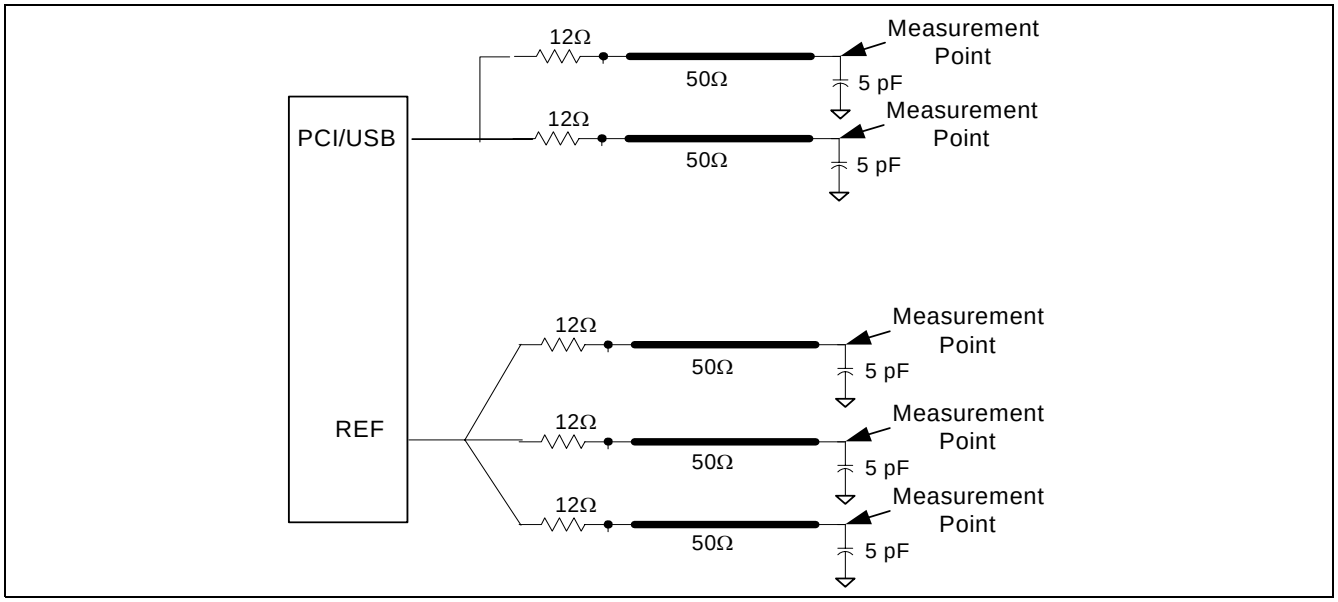


Figure 13. Single-ended Load Configuration HIGH DRIVE OPTION



The following diagrams show the test load configuration for the differential CPU and PCIEX outputs.

Figure 14. Differential Load Configuration for 0.7V Push Pull Clock

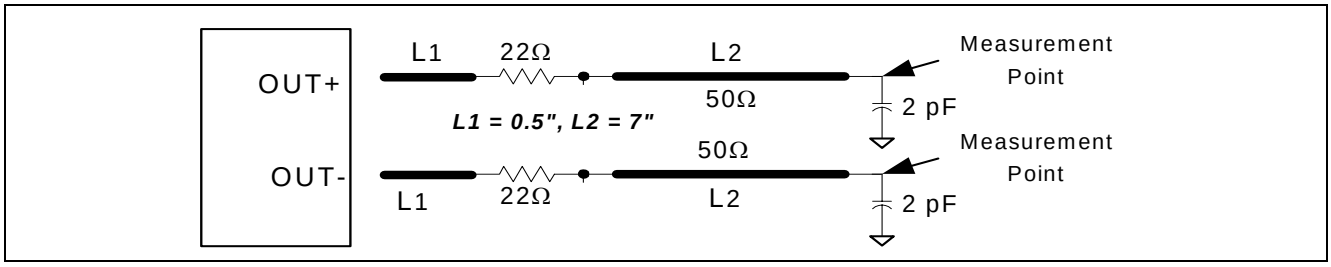


Figure 15. Differential Load Configuration for 0.7 Push Pull Clock

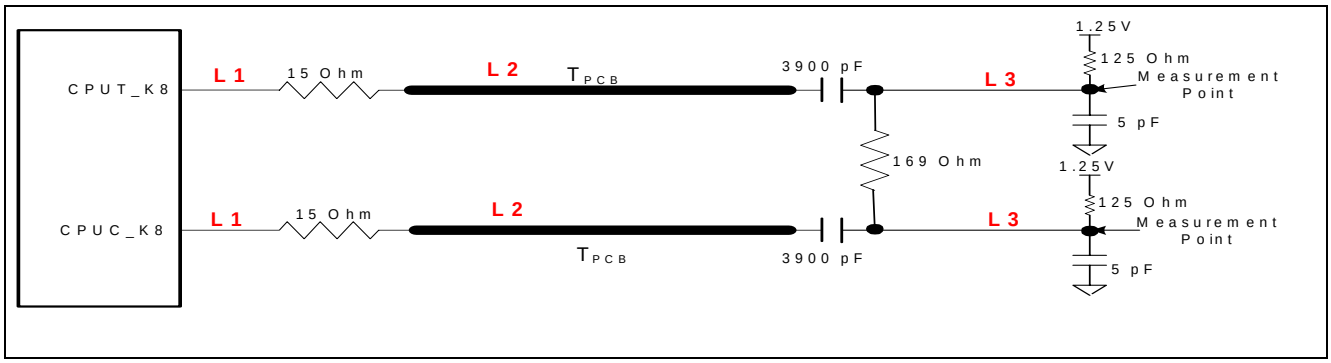


Figure 16. Differential Measurement for Differential Output Signals (for AC Parameters Measurement)

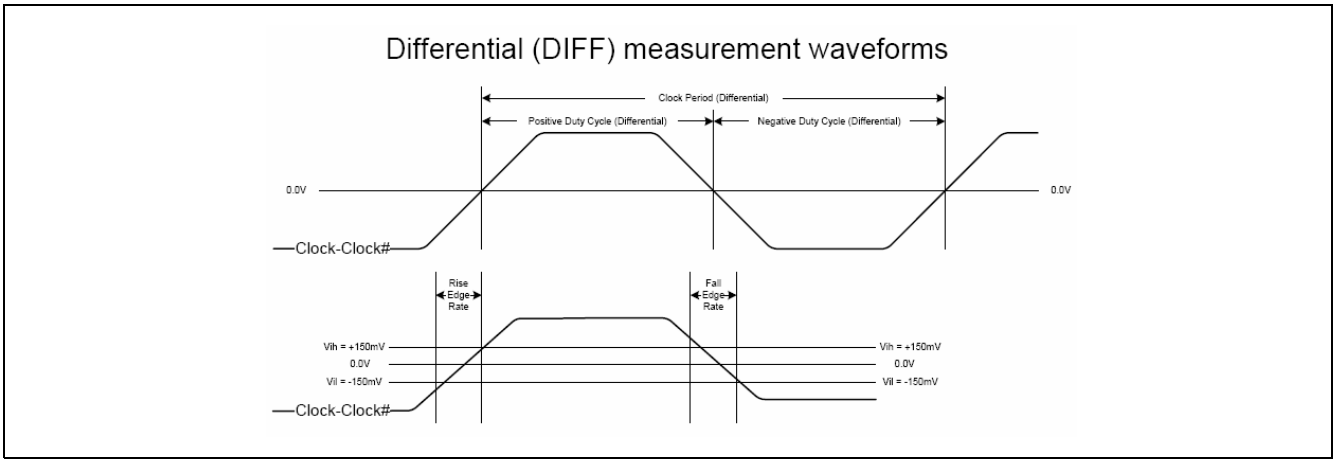


Figure 17. Single-ended Measurement for Differential Output Signals (for AC Parameters Measurement)

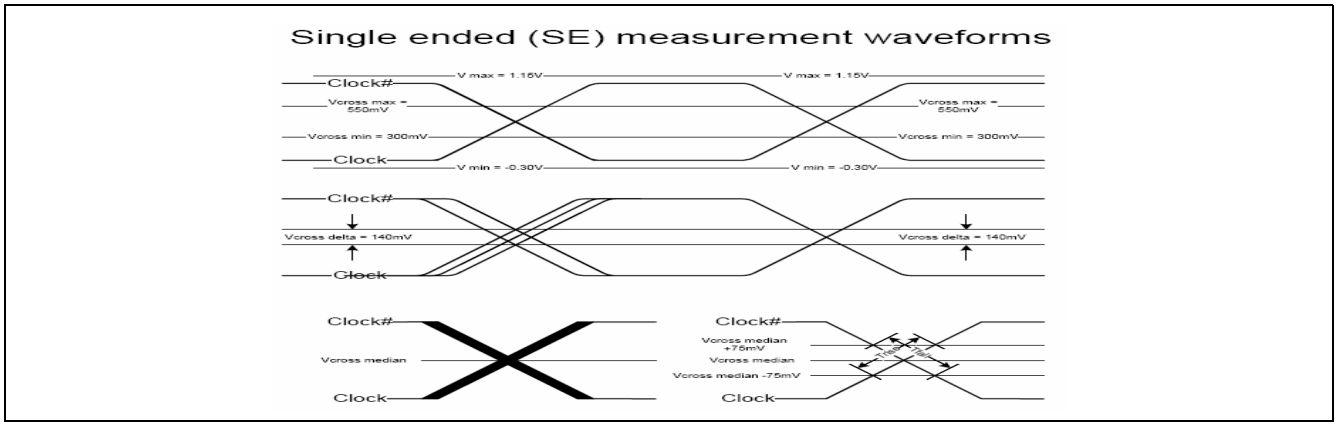
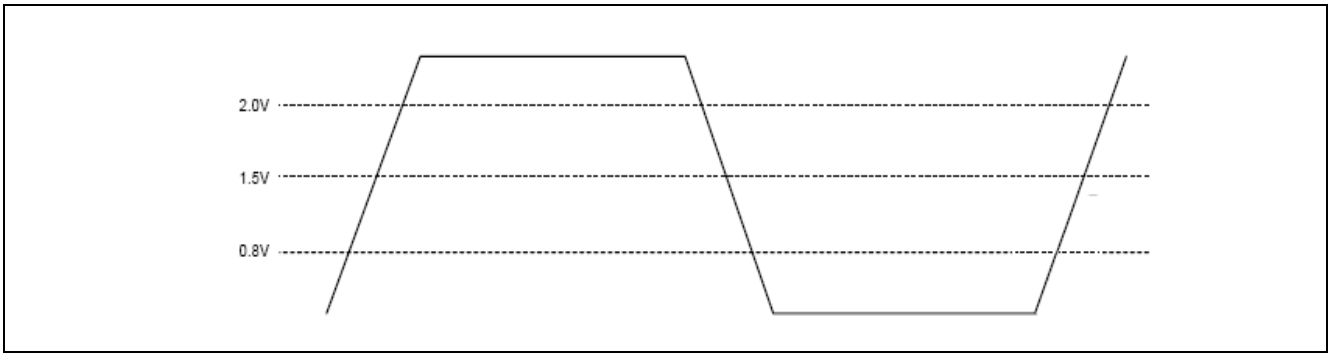


Figure 18. Single-ended Output Signals (for AC Parameters Measurement)

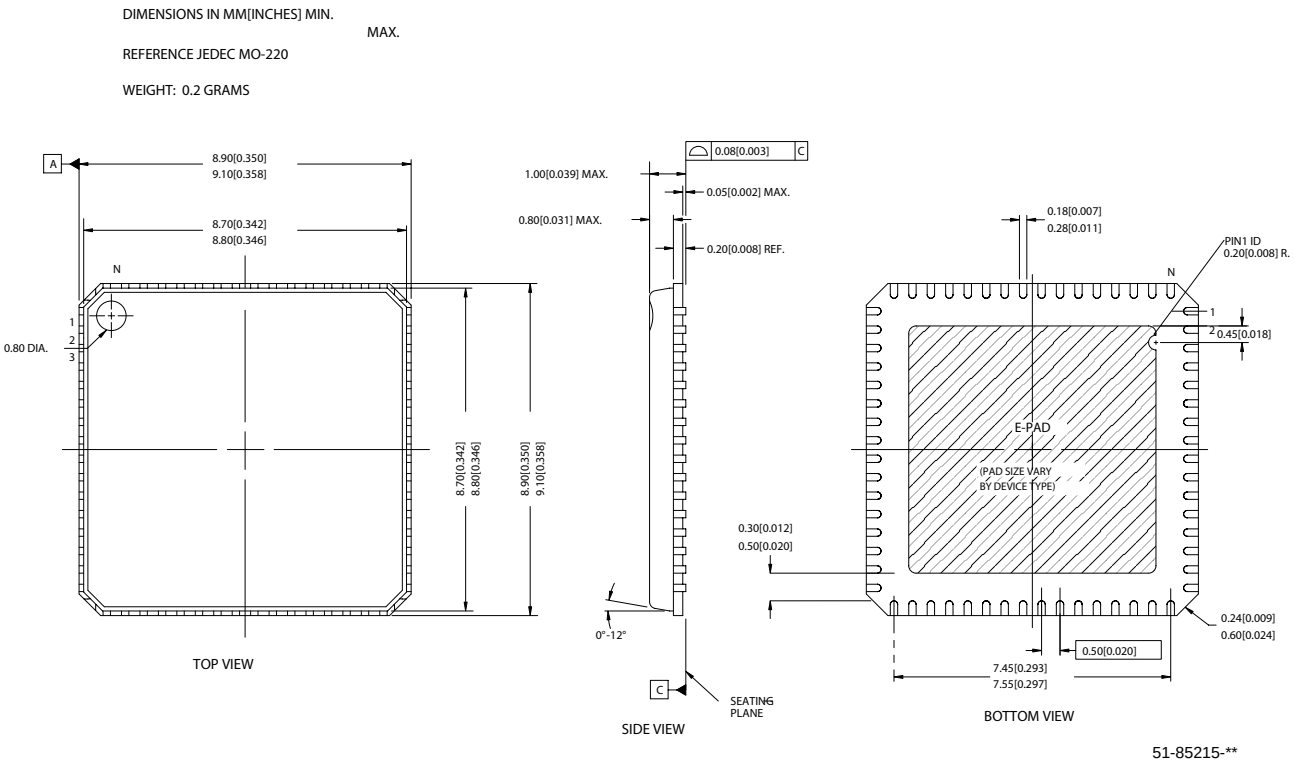


Ordering Information

Part Number	Package Type	Product Flow
Lead-free		
CY28551LFXC	64-pin QFN	Commercial, 0° to 85°C
CY28551LFXCT	64-pin QFN – Tape and Reel	Commercial, 0° to 85°C

Package Diagram

Figure 19. 64-Lead QFN 9 x 9 mm (Punch Version) LF64A



Document History Page

Document Title: CY28551 Universal Clock Generator for Intel, VIA, and SIS® Document Number: 001-05675				
REV.	**ECN NO.**	**Issue Date**	**Orig. of Change**	**Description of Change**
**	409135	See ECN	HGS	New Data Sheet
*A	417501	See ECN	HGS	Register alignment changed: -BYTE 9 contains DF3_N8,DF2_N8,DF1_N8 -BYTE 10 contains DF1_N<7:0> -BYTE 11 contains DF2_N<7:0> -BYTE 12 contains DF3_N<7:0> Add POWERGOOD status bit at BYTE6 [0] Add CPU_STP#, PCI_STP# and CLKREQ# description
*B	460105	See ECN	RGL	Minor change – To post on web
*C	491596	See ECN	HGS	1. Change 48M/24_48M driving strength to high by default 2. Change Revision ID (BYTE7[7:4]) to 0010