AMD Xilinx - XCV300E-7PQ240C Datasheet

Welcome to E-XFL.COM

Understanding Embedded - FPGAs (Field Programmable Gate Array)

Embedded - FPGAs, or Field Programmable Gate Arrays, are advanced integrated circuits that offer unparalleled flexibility and performance for digital systems. Unlike traditional fixed-function logic devices, FPGAs can be programmed and reprogrammed to execute a wide array of logical operations, enabling customized functionality tailored to specific applications. This reprogrammability allows developers to iterate designs quickly and implement complex functions without the need for custom hardware.

Applications of Embedded - FPGAs

The versatility of Embedded - FPGAs makes them indispensable in numerous fields. In telecommunications,

Details	
Product Status	Obsolete
Number of LABs/CLBs	1536
Number of Logic Elements/Cells	6912
Total RAM Bits	131072
Number of I/O	158
Number of Gates	411955
Voltage - Supply	1.71V ~ 1.89V
Mounting Type	Surface Mount
Operating Temperature	0°C ~ 85°C (TJ)
Package / Case	240-BFQFP
Supplier Device Package	240-PQFP (32x32)
Purchase URL	https://www.e-xfl.com/product-detail/xilinx/xcv300e-7pq240c

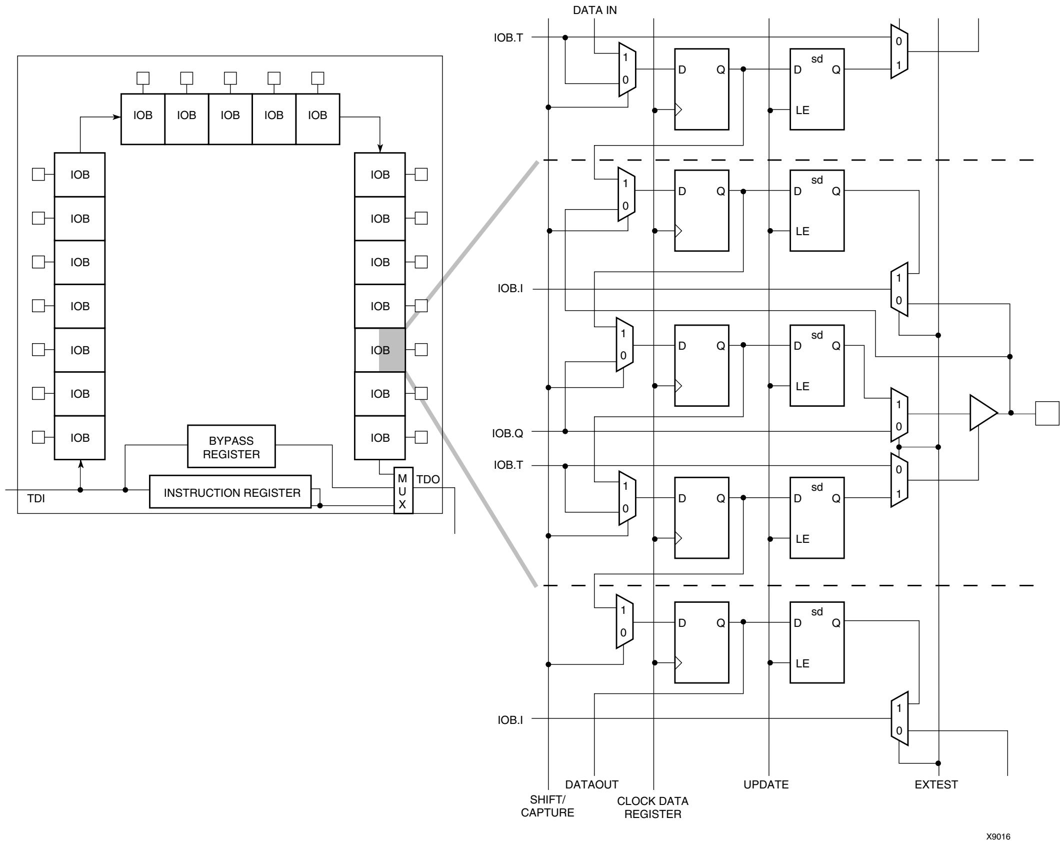

Figure 11: **Virtex-E Family Boundary Scan Logic**

Instruction Set

The Virtex-E series Boundary Scan instruction set also includes instructions to configure the device and read back configuration data (CFG_IN, CFG_OUT, and JSTART). The complete instruction set is coded as shown in Table 6..

Table 6: **Boundary Scan Instructions**

Boundary Scan Command	Binary Code(4:0)	Description
EXTEST	00000	Enables Boundary Scan EXTEST operation
SAMPLE/ PRELOAD	00001	Enables Boundary Scan SAMPLE/PRELOAD operation
USER1	00010	Access user-defined register 1
USER2	00011	Access user-defined register 2
CFG_OUT	00100	Access the configuration bus for read operations.

Table 6: **Boundary Scan Instructions *(Continued)***

Boundary Scan Command	Binary Code(4:0)	Description
CFG_IN	00101	Access the configuration bus for write operations.
INTEST	00111	Enables Boundary Scan INTEST operation
USERCODE	01000	Enables shifting out USER code
IDCODE	01001	Enables shifting out of ID Code
HIGHZ	01010	3-states output pins while enabling the Bypass Register
JSTART	01100	Clock the start-up sequence when StartupClk is TCK
BYPASS	11111	Enables BYPASS
RESERVED	All other codes	Xilinx reserved instructions

For in-circuit debugging, an optional download and read-back cable is available. This cable connects the FPGA in the target system to a PC or workstation. After downloading the design into the FPGA, the designer can single-step the logic, readback the contents of the flip-flops, and so observe the internal logic state. Simple modifications can be down-loaded into the system in a matter of minutes.

Configuration

Virtex-E devices are configured by loading configuration data into the internal configuration memory. Note that attempting to load an incorrect bitstream causes configuration to fail and can damage the device.

Some of the pins used for configuration are dedicated pins, while others can be re-used as general purpose inputs and outputs once configuration is complete.

The following are dedicated pins:

- Mode pins (M2, M1, M0)
- Configuration clock pin (CCLK)
- $\overline{\text{PROGRAM}}$ pin
- DONE pin
- Boundary Scan pins (TDI, TDO, TMS, TCK)

Depending on the configuration mode chosen, CCLK can be an output generated by the FPGA, or can be generated externally and provided to the FPGA as an input. The $\overline{\text{PROGRAM}}$ pin must be pulled High prior to reconfiguration.

Note that some configuration pins can act as outputs. For correct operation, these pins require a V_{CCO} of 3.3 V or 2.5 V. At 3.3 V the pins operate as LVTTL, and at 2.5 V they operate as LVCMOS. All affected pins fall in banks 2 or 3. The configuration pins needed for SelectMap (CS, Write) are located in bank 1.

Configuration Modes

Virtex-E supports the following four configuration modes.

- Slave-serial mode
- Master-serial mode
- SelectMAP mode
- Boundary Scan mode (JTAG)

The Configuration mode pins (M2, M1, M0) select among these configuration modes with the option in each case of having the IOB pins either pulled up or left floating prior to configuration. The selection codes are listed in Table 8.

Configuration through the Boundary Scan port is always available, independent of the mode selection. Selecting the Boundary Scan mode simply turns off the other modes. The three mode pins have internal pull-up resistors, and default to a logic High if left unconnected. However, it is recommended to drive the configuration mode pins externally.

Table 8: **Configuration Codes**

Configuration Mode	M2(1)	M1	M0	CCLK Direction	Data Width	Serial D_{out}	Configuration Pull-ups(1)
Master-serial mode	0	0	0	Out	1	Yes	No
Boundary Scan mode	1	0	1	N/A	1	No	No
SelectMAP mode	1	1	0	In	8	No	No
Slave-serial mode	1	1	1	In	1	Yes	No
Master-serial mode	1	0	0	Out	1	Yes	Yes
Boundary Scan mode	0	0	1	N/A	1	No	Yes
SelectMAP mode	0	1	0	In	8	No	Yes
Slave-serial mode	0	1	1	In	1	Yes	Yes

Notes:
1. M2 is sampled continuously from power up until the end of the configuration. Toggling M2 while INIT is being held externally Low can cause the configuration pull-up settings to change.

Table 9 lists the total number of bits required to configure each device.

Table 9: **Virtex-E Bitstream Lengths**

Device	# of Configuration Bits
XCV50E	630,048
XCV100E	863,840
XCV200E	1,442,016
XCV300E	1, 875,648
XCV400E	2,693,440
XCV600E	3,961,632
XCV1000E	6,587,520
XCV1600E	8,308,992
XCV2000E	10,159,648
XCV2600E	12,922,336
XCV3200E	16,283,712

Slave-Serial Mode

In slave-serial mode, the FPGA receives configuration data in bit-serial form from a serial PROM or other source of serial configuration data. The serial bitstream must be set up at the DIN input pin a short time before each rising edge of an externally generated CCLK.

For more detailed information on serial PROMs, see the PROM data sheet at **http://www.xilinx.com/bvdocs/publications/ds026.pdf**.

Multiple FPGAs can be daisy-chained for configuration from a single source. After a particular FPGA has been configured, the data for the next device is routed to the DOUT pin. The maximum capacity for a single LOUT/DOUT write is 2^{20}-1 (1,048,575) 32-bit words, or 33,554,4000 bits. The data on the DOUT pin changes on the rising edge of CCLK.

The change of DOUT on the rising edge of CCLK differs from previous families, but does not cause a problem for mixed configuration chains. This change was made to improve serial configuration rates for Virtex and Virtex-E only chains.

Figure 13 shows a full master/slave system. A Virtex-E device in slave-serial mode should be connected as shown in the right-most device.

Slave-serial mode is selected by applying <111> or <011> to the mode pins (M2, M1, M0). A weak pull-up on the mode pins makes slave serial the default mode if the pins are left unconnected. However, it is recommended to drive the configuration mode pins externally. Figure 14 shows slave-serial mode programming switching characteristics.

Table 10 provides more detail about the characteristics shown in Figure 14. Configuration must be delayed until the $\overline{\text{INIT}}$ pins of all daisy-chained FPGAs are High.

Table 10: **Master/Slave Serial Mode Programming Switching**

	Description	Figure References	Symbol	Values	Units
CCLK	DIN setup/hold, slave mode	1/2	T_{DCC}/T_{CCD}	5.0 / 0.0	ns, min
	DIN setup/hold, master mode	1/2	T_{DSCK}/T_{CKDS}	5.0 / 0.0	ns, min
	DOUT	3	T_{CCO}	12.0	ns, max
	High time	4	T_{CCH}	5.0	ns, min
	Low time	5	T_{CCL}	5.0	ns, min
	Maximum Frequency		F_{CC}	66	MHz, max
	Frequency Tolerance, master mode with respect to nominal			+45% –30%	

Configuration through the TAP uses the CFG_IN instruction. This instruction allows data input on TDI to be converted into data packets for the internal configuration bus.

The following steps are required to configure the FPGA through the Boundary Scan port (when using TCK as a start-up clock).

1. Load the CFG_IN instruction into the Boundary Scan instruction register (IR).
2. Enter the Shift-DR (SDR) state.
3. Shift a configuration bitstream into TDI.
4. Return to Run-Test-Idle (RTI).
5. Load the JSTART instruction into IR.
6. Enter the SDR state.
7. Clock TCK through the startup sequence.
8. Return to RTI.

Configuration and readback via the TAP is always available. The Boundary Scan mode is selected by a <101> or <001> on the mode pins (M2, M1, M0). For details on TAP characteristics, refer to XAPP139.

Configuration Sequence

The configuration of Virtex-E devices is a three-phase process. First, the configuration memory is cleared. Next, configuration data is loaded into the memory, and finally, the logic is activated by a start-up process.

Configuration is automatically initiated on power-up unless it is delayed by the user, as described below. The configuration process can also be initiated by asserting $\overline{\text{PROGRAM}}$. The end of the memory-clearing phase is signalled by $\overline{\text{INIT}}$ going High, and the completion of the entire process is signalled by DONE going High.

The power-up timing of configuration signals is shown in Figure 20.

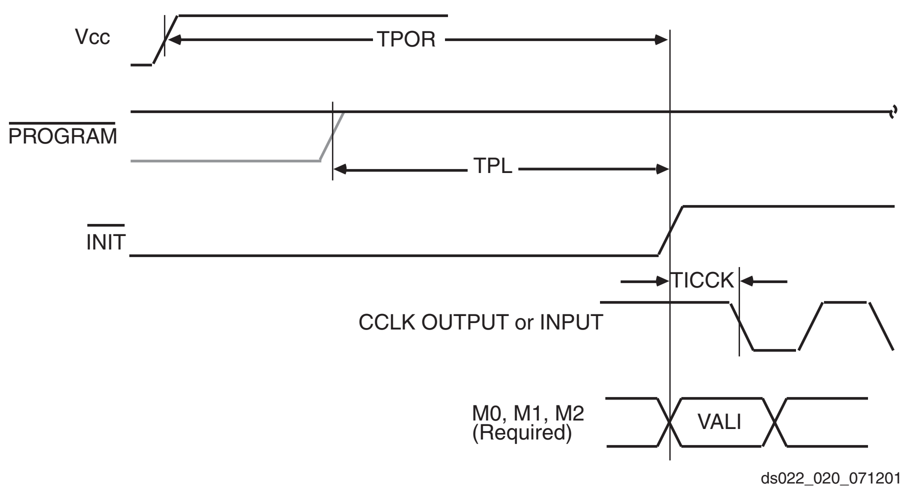

Figure 20: **Power-Up Timing Configuration Signals**

The corresponding timing characteristics are listed in Table 12.

Table 12: **Power-up Timing Characteristics**

Description	Symbol	Value	Units
Power-on Reset[1]	T_{POR}	2.0	ms, max
Program Latency	T_{PL}	100.0	µs, max
CCLK (output) Delay	T_{ICCK}	0.5	µs, min
		4.0	µs, max
Program Pulse Width	$T_{PROGRAM}$	300	ns, min

Notes:

1. T_{POR} delay is the initialization time required after V_{CCINT} and V_{CCO} in Bank 2 reach the recommended operating voltage.

Delaying Configuration

$\overline{\text{INIT}}$ can be held Low using an open-drain driver. An open-drain is required since $\overline{\text{INIT}}$ is a bidirectional open-drain pin that is held Low by the FPGA while the configuration memory is being cleared. Extending the time that the pin is Low causes the configuration sequencer to wait. Thus, configuration is delayed by preventing entry into the phase where data is loaded.

Start-Up Sequence

The default Start-up sequence is that one CCLK cycle after DONE goes High, the global 3-state signal (GTS) is released. This permits device outputs to turn on as necessary.

One CCLK cycle later, the Global Set/Reset (GSR) and Global Write Enable (GWE) signals are released. This permits

DLL Properties

Properties provide access to some of the Virtex-E series DLL features, (for example, clock division and duty cycle correction).

Duty Cycle Correction Property

The 1x clock outputs, CLK0, CLK90, CLK180, and CLK270, use the duty-cycle corrected default, exhibiting a 50/50 duty cycle. The DUTY_CYCLE_CORRECTION property (by default TRUE) controls this feature. To deactivate the DLL duty-cycle correction for the 1x clock outputs, attach the DUTY_CYCLE_CORRECTION=FALSE property to the DLL symbol.

Clock Divide Property

The CLKDV_DIVIDE property specifies how the signal on the CLKDV pin is frequency divided with respect to the CLK0 pin. The values allowed for this property are 1.5, 2, 2.5, 3, 4, 5, 8, or 16; the default value is 2.

Startup Delay Property

This property, STARTUP_WAIT, takes on a value of TRUE or FALSE (the default value). When TRUE the device configuration DONE signal waits until the DLL locks before going to High.

Virtex-E DLL Location Constraints

As shown in Figure 26, there are four additional DLLs in the Virtex-E devices, for a total of eight per Virtex-E device. These DLLs are located in silicon, at the top and bottom of the two innermost block SelectRAM columns. The location constraint LOC, attached to the DLL symbol with the identifier DLL0S, DLL0P, DLL1S, DLL1P, DLL2S, DLL2P, DLL3S, or DLL3P, controls the DLL location.

The LOC property uses the following form:

LOC = DLL0P

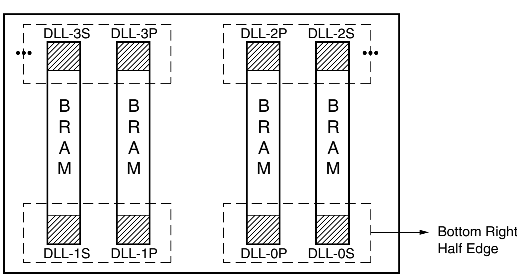

Figure 26: **Virtex Series DLLs**

Design Factors

Use the following design considerations to avoid pitfalls and improve success designing with Xilinx devices.

Input Clock

The output clock signal of a DLL, essentially a delayed version of the input clock signal, reflects any instability on the input clock in the output waveform. For this reason the quality of the DLL input clock relates directly to the quality of the output clock waveforms generated by the DLL. The DLL input clock requirements are specified in the data sheet.

In most systems a crystal oscillator generates the system clock. The DLL can be used with any commercially available quartz crystal oscillator. For example, most crystal oscillators produce an output waveform with a frequency tolerance of 100 PPM, meaning 0.01 percent change in the clock period. The DLL operates reliably on an input waveform with a frequency drift of up to 1 ns — orders of magnitude in excess of that needed to support any crystal oscillator in the industry. However, the cycle-to-cycle jitter must be kept to less than 300 ps in the low frequencies and 150 ps for the high frequencies.

Input Clock Changes

Changing the period of the input clock beyond the maximum drift amount requires a manual reset of the CLKDLL. Failure to reset the DLL produces an unreliable lock signal and output clock.

It is possible to stop the input clock with little impact to the DLL. Stopping the clock should be limited to less than 100 µs to keep device cooling to a minimum. The clock should be stopped during a Low phase, and when restored the full High period should be seen. During this time, LOCKED stays High and remains High when the clock is restored.

When the clock is stopped, one to four more clocks are still observed as the delay line is flushed. When the clock is restarted, the output clocks are not observed for one to four clocks as the delay line is filled. The most common case is two or three clocks.

In a similar manner, a phase shift of the input clock is also possible. The phase shift propagates to the output one to four clocks after the original shift, with no disruption to the CLKDLL control.

Output Clocks

As mentioned earlier in the DLL pin descriptions, some restrictions apply regarding the connectivity of the output pins. The DLL clock outputs can drive an OBUF, a global clock buffer BUFG, or they can route directly to destination clock pins. The only BUFGs that the DLL clock outputs can drive are the two on the same edge of the device (top or bottom). In addition, the CLK2X output of the secondary DLL can connect directly to the CLKIN of the primary DLL in the same quadrant.

Do not use the DLL output clock signals until after activation of the LOCKED signal. Prior to the activation of the LOCKED signal, the DLL output clocks are not valid and can exhibit glitches, spikes, or other spurious movement.

Useful Application Examples

The Virtex-E DLL can be used in a variety of creative and useful applications. The following examples show some of the more common applications. The Verilog and VHDL example files are available at:

ftp://ftp.xilinx.com/pub/applications/xapp/xapp132.zip

Standard Usage

The circuit shown in Figure 27 resembles the BUFGDLL macro implemented to provide access to the RST and LOCKED pins of the CLKDLL.

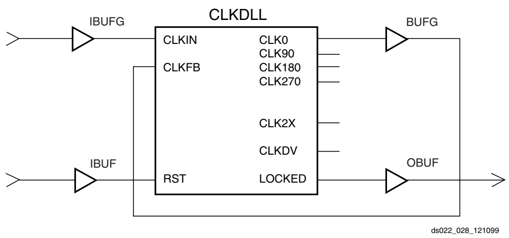

Figure 27: **Standard DLL Implementation**

Board Level Deskew of Multiple Non-Virtex-E Devices

The circuit shown in Figure 28 can be used to deskew a system clock between a Virtex-E chip and other non-Virtex-E chips on the same board. This application is commonly used when the Virtex-E device is used in conjunction with other standard products such as SRAM or DRAM devices. While designing the board level route, ensure that the return net delay to the source equals the delay to the other chips involved.

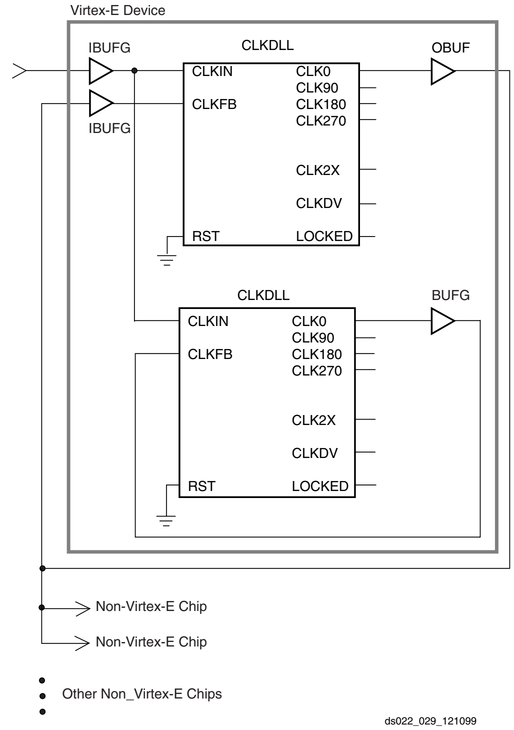

Figure 28: **DLL Deskew of Board Level Clock**

Board-level deskew is not required for low-fanout clock networks. It is recommended for systems that have fanout limitations on the clock network, or if the clock distribution chip cannot handle the load.

Do not use the DLL output clock signals until after activation of the LOCKED signal. Prior to the activation of the LOCKED signal, the DLL output clocks are not valid and can exhibit glitches, spikes, or other spurious movement.

The dll_mirror_1 files in the xapp132.zip file show the VHDL and Verilog implementation of this circuit.

Deskew of Clock and Its 2x Multiple

The circuit shown in Figure 29 implements a 2x clock multiplier and also uses the CLK0 clock output with a zero ns skew between registers on the same chip. Alternatively, a clock divider circuit can be implemented using similar connections.

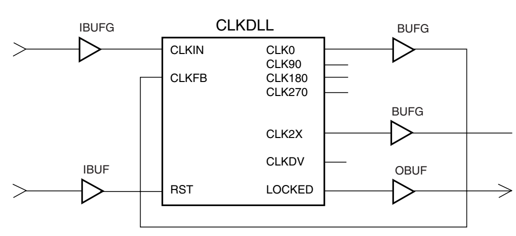

Figure 29: **DLL Deskew of Clock and 2x Multiple**

Fundamentals

Modern bus applications, pioneered by the largest and most influential companies in the digital electronics industry, are commonly introduced with a new I/O standard tailored specifically to the needs of that application. The bus I/O standards provide specifications to other vendors who create products designed to interface with these applications. Each standard often has its own specifications for current, voltage, I/O buffering, and termination techniques.

The ability to provide the flexibility and time-to-market advantages of programmable logic is increasingly dependent on the capability of the programmable logic device to support an ever increasing variety of I/O standards

The SelectI/O resources feature highly configurable input and output buffers which provide support for a wide variety of I/O standards. As shown in Table 18, each buffer type can support a variety of voltage requirements.

Table 18: **Virtex-E Supported I/O Standards**

I/O Standard	Output V_{CCO}	Input V_{CCO}	Input V_{REF}	Board Termination Voltage (V_{TT})
LVTTL	3.3	3.3	N/A	N/A
LVCMOS2	2.5	2.5	N/A	N/A
LVCMOS18	1.8	1.8	N/A	N/A
SSTL3 I & II	3.3	N/A	1.50	1.50
SSTL2 I & II	2.5	N/A	1.25	1.25
GTL	N/A	N/A	0.80	1.20
GTL+	N/A	N/A	1.0	1.50
HSTL I	1.5	N/A	0.75	0.75
HSTL III & IV	1.5	N/A	0.90	1.50
CTT	3.3	N/A	1.50	1.50
AGP-2X	3.3	N/A	1.32	N/A
PCI33_3	3.3	3.3	N/A	N/A
PCI66_3	3.3	3.3	N/A	N/A
BLVDS & LVDS	2.5	N/A	N/A	N/A
LVPECL	3.3	N/A	N/A	N/A

Overview of Supported I/O Standards

This section provides a brief overview of the I/O standards supported by all Virtex-E devices.

While most I/O standards specify a range of allowed voltages, this document records typical voltage values only. Detailed information on each specification can be found on the Electronic Industry Alliance Jedec website at:

http://www.jedec.org

LVTTL — Low-Voltage TTL

The Low-Voltage TTL, or LVTTL standard is a general purpose EIA/JESDSA standard for 3.3V applications that uses an LVTTL input buffer and a Push-Pull output buffer. This standard requires a 3.3V output source voltage (V_{CCO}), but does not require the use of a reference voltage (V_{REF}) or a termination voltage (V_{TT}).

LVCMOS2 — Low-Voltage CMOS for 2.5 Volts

The Low-Voltage CMOS for 2.5 Volts or lower, or LVCMOS2 standard is an extension of the LVCMOS standard (JESD 8.-5) used for general purpose 2.5V applications. This standard requires a 2.5V output source voltage (V_{CCO}), but does not require the use of a reference voltage (V_{REF}) or a board termination voltage (V_{TT}).

LVCMOS18 — 1.8 V Low Voltage CMOS

This standard is an extension of the LVCMOS standard. It is used in general purpose 1.8 V applications. The use of a reference voltage (V_{REF}) or a board termination voltage (V_{TT}) is not required.

PCI — Peripheral Component Interface

The Peripheral Component Interface, or PCI standard specifies support for both 33 MHz and 66 MHz PCI bus applications. It uses a LVTTL input buffer and a Push-Pull output buffer. This standard does not require the use of a reference voltage (V_{REF}) or a board termination voltage (V_{TT}), however, it does require a 3.3V output source voltage (V_{CCO}).

GTL — Gunning Transceiver Logic Terminated

The Gunning Transceiver Logic, or GTL standard is a high-speed bus standard (JESD8.3) invented by Xerox. Xilinx has implemented the terminated variation for this standard. This standard requires a differential amplifier input buffer and a Open Drain output buffer.

GTL+ — Gunning Transceiver Logic Plus

The Gunning Transceiver Logic Plus, or GTL+ standard is a high-speed bus standard (JESD8.3) first used by the Pentium Pro processor.

HSTL — High-Speed Transceiver Logic

The High-Speed Transceiver Logic, or HSTL standard is a general purpose high-speed, 1.5V bus standard sponsored by IBM (EIA/JESD 8-6). This standard has four variations or classes. SelectI/O devices support Class I, III, and IV. This

CLB Switching Characteristics

Delays originating at F/G inputs vary slightly according to the input used, see Figure 2. The values listed below are worst-case. Precise values are provided by the timing analyzer.

Description	Symbol	Speed Grade[(1)] Min	-8	-7	-6	Units
Combinatorial Delays						
4-input function: F/G inputs to X/Y outputs	T_{ILO}	0.19	0.40	0.42	0.47	ns, max
5-input function: F/G inputs to F5 output	T_{IF5}	0.36	0.76	0.8	0.9	ns, max
5-input function: F/G inputs to X output	T_{IF5X}	0.35	0.74	0.8	0.9	ns, max
6-input function: F/G inputs to Y output via F6 MUX	T_{IF6Y}	0.35	0.74	0.9	1.0	ns, max
6-input function: F5IN input to Y output	T_{F5INY}	0.04	0.11	0.20	0.22	ns, max
Incremental delay routing through transparent latch to XQ/YQ outputs	T_{IFNCTL}	0.27	0.63	0.7	0.8	ns, max
BY input to YB output	T_{BYYB}	0.19	0.38	0.46	0.51	ns, max
Sequential Delays						
FF Clock CLK to XQ/YQ outputs	T_{CKO}	0.34	0.78	0.9	1.0	ns, max
Latch Clock CLK to XQ/YQ outputs	T_{CKLO}	0.40	0.77	0.9	1.0	ns, max
Setup and Hold Times before/after Clock CLK						
4-input function: F/G Inputs	T_{ICK} / T_{CKI}	0.39 / 0	0.9 / 0	1.0 / 0	1.1 / 0	ns, min
5-input function: F/G inputs	T_{IF5CK} / T_{CKIF5}	0.55 / 0	1.3 / 0	1.4 / 0	1.5 / 0	ns, min
6-input function: F5IN input	T_{F5INCK} / T_{CKF5IN}	0.27 / 0	0.6 / 0	0.8 / 0	0.8 / 0	ns, min
6-input function: F/G inputs via F6 MUX	T_{IF6CK} / T_{CKIF6}	0.58 / 0	1.3 / 0	1.5 / 0	1.6 / 0	ns, min
BX/BY inputs	T_{DICK} / T_{CKDI}	0.25 / 0	0.6 / 0	0.7 / 0	0.8 / 0	ns, min
CE input	T_{CECK} / T_{CKCE}	0.28 / 0	0.55 / 0	0.7 / 0	0.7 / 0	ns, min
SR/BY inputs (synchronous)	T_{RCK} / T_{CKR}	0.24 / 0	0.46 / 0	0.52 / 0	0.6 / 0	ns, min
Clock CLK						
Minimum Pulse Width, High	T_{CH}	0.56	1.2	1.3	1.4	ns, min
Minimum Pulse Width, Low	T_{CL}	0.56	1.2	1.3	1.4	ns, min
Set/Reset						
Minimum Pulse Width, SR/BY inputs	T_{RPW}	0.94	1.9	2.1	2.4	ns, min
Delay from SR/BY inputs to XQ/YQ outputs (asynchronous)	T_{RQ}	0.39	0.8	0.9	1.0	ns, max
Toggle Frequency (MHz) (for export control)	F_{TOG}	-	416	400	357	MHz

Notes:
1. A Zero "0" Hold Time listing indicates no hold time or a negative hold time. Negative values can not be guaranteed "best-case", but if a "0" is listed, there is no positive hold time.

DLL Clock Tolerance, Jitter, and Phase Information

All DLL output jitter and phase specifications determined through statistical measurement at the package pins using a clock mirror configuration and matched drivers.

			CLKDLLHF		CLKDLL		
Description	**Symbol**	**F_{CLKIN}**	**Min**	**Max**	**Min**	**Max**	**Units**
Input Clock Period Tolerance	T_{IPTOL}		-	1.0	-	1.0	ns
Input Clock Jitter Tolerance (Cycle to Cycle)	T_{IJITCC}		-	± 150	-	± 300	ps
Time Required for DLL to Acquire Lock(6)	T_{LOCK}	> 60 MHz	-	20	-	20	µs
		50 - 60 MHz	-	-	-	25	µs
		40 - 50 MHz	-	-	-	50	µs
		30 - 40 MHz	-	-	-	90	µs
		25 - 30 MHz	-	-	-	120	µs
Output Jitter (cycle-to-cycle) for any DLL Clock Output(1)	T_{OJITCC}			± 60		± 60	ps
Phase Offset between CLKIN and CLKO(2)	T_{PHIO}			± 100		± 100	ps
Phase Offset between Clock Outputs on the DLL(3)	T_{PHOO}			± 140		± 140	ps
Maximum Phase Difference between CLKIN and CLKO(4)	T_{PHIOM}			± 160		± 160	ps
Maximum Phase Difference between Clock Outputs on the DLL(5)	T_{PHOOM}			± 200		± 200	ps

Notes:

1. **Output Jitter** is cycle-to-cycle jitter measured on the DLL output clock and is based on a maximum tap delay resolution, *excluding* input clock jitter.
2. **Phase Offset between CLKIN and CLKO** is the worst-case fixed time difference between rising edges of CLKIN and CLKO, *excluding* Output Jitter and input clock jitter.
3. **Phase Offset between Clock Outputs on the DLL** is the worst-case fixed time difference between rising edges of any two DLL outputs, *excluding* Output Jitter and input clock jitter.
4. **Maximum Phase Difference between CLKIN an CLKO** is the sum of Output Jitter and Phase Offset between CLKIN and CLKO, or the greatest difference between CLKIN and CLKO rising edges due to DLL alone (*excluding* input clock jitter).
5. **Maximum Phase DIfference between Clock Outputs on the DLL** is the sum of Output JItter and Phase Offset between any DLL clock outputs, or the greatest difference between any two DLL output rising edges sue to DLL alone (*excluding* input clock jitter).
6. Add 30% to the value for industrial grade parts.

Table 4: **CS144 — XCV50E, XCV100E, XCV200E**

Bank	Pin Description	Pin #
1	VCCO	A13
1	VCCO	D7
2	VCCO	B12
3	VCCO	G11
3	VCCO	M13
4	VCCO	N13
5	VCCO	N1
5	VCCO	N7
6	VCCO	M2
7	VCCO	B2
7	VCCO	G2
NA	GND	A1
NA	GND	B9
NA	GND	B11
NA	GND	C7
NA	GND	D5
NA	GND	E4
NA	GND	E11
NA	GND	F1
NA	GND	G10
NA	GND	J1
NA	GND	J12
NA	GND	L3
NA	GND	L5
NA	GND	L7
NA	GND	L9
NA	GND	N12

Notes:
1. V_{REF} or I/O option only in the XCV200E; otherwise, I/O option only.
2. V_{REF} or I/O option only in the XCV100E, 200E; otherwise, I/O option only.

CS144 Differential Pin Pairs

Virtex-E devices have differential pin pairs that can also provide other functions when not used as a differential pair. A √ in the AO column indicates that the pin pair can be used as an asynchronous output for all devices provided in this package. Pairs with a note number in the AO column are device dependent. They can have asynchronous outputs if the pin pair are in the same CLB row and column in the device. Numbers in this column refer to footnotes that indicate which devices have pin pairs than can be asynchronous outputs. The Other Functions column indicates alternative function(s) not available when the pair is used as a differential pair or differential clock.

Table 5: **CS144 Differential Pin Pair Summary XCV50E, XCV100E, XCV200E**

Pair	Bank	P Pin	N Pin	AO	Other Functions
Global Differential Clock					
0	4	K7	N8	NA	IO_DLL_L18P
1	5	M7	M6	NA	IO_DLL_L18N
2	1	A7	B7	NA	IO_DLL_L2P
3	0	A6	C6	NA	IO_DLL_L2N
IO LVDS Total Pairs: 30, Asynchronous Output Pairs: 18					
0	0	A4	B4	√	VREF
1	0	A5	B5	√	-
2	1	B7	C6	NA	IO_LVDS_DLL
3	1	D8	C8	√	-
4	1	D9	C9	√	VREF
5	1	D10	C10	√	CS, WRITE
6	2	C11	C12	√	DIN, D0
7	2	D13	E10	1	D1, VREF
8	2	E12	E13	√	D2
9	2	F10	F11	1	D3, VREF
10	3	F13	G13	NA	-
11	3	H12	H11	1	D4, VREF
12	3	H10	J13	√	D5
13	3	J11	J10	1	D6, VREF
14	3	K10	L13	√	INIT
15	4	L11	M11	√	-
16	4	N10	K9	√	VREF
17	4	N9	K8	√	-

Table 7: **PQ240 Differential Pin Pair Summary XCV50E, XCV100E, XCV200E, XCV300E, XCV400E**

Pair	Bank	P Pin	N Pin	AO	Other Functions
48	6	P56	P57	√	-
49	6	P52	P53	2	-
50	6	P49	P50	3	VREF
51	6	P46	P47	4	VREF
52	6	P41	P42	√	-
53	6	P38	P39	2	-
54	6	P35	P36	4	VREF
55	6	P33	P34	5	VREF
56	7	P27	P28	√	-
57	7	P23	P24	4	VREF
58	7	P20	P21	2	-
59	7	P17	P18	√	-
60	7	P12	P13	4	VREF
61	7	P9	P10	3	VREF
62	7	P6	P7	2	-
63	7	P4	P5	6	VREF

Notes:
1. AO in the XCV50E.
2. AO in the XCV50E, 100E, 200E, 300E.
3. AO in the XCV50E, 200E, 300E, 400E.
4. AO in the XCV50E, 300E, 400E.
5. AO in the XCV100E, 200E, 400E.
6. AO in the XCV100E, 400E.
7. AO in the XCV50E, 200E, 400E.
8. AO in the XCV100E.

HQ240 High-Heat Quad Flat-Pack Packages

XCV600E and XCV1000E devices in High-heat dissipation Quad Flat-pack packages have footprint compatibility. Pins labeled I0_VREF can be used as either in all parts unless device-dependent as indicated in the footnotes. If the pin is not used as V_{REF}, it can be used as general I/O. Immediately following Table 8, see Table 9 for Differential Pair information.

Table 8: **HQ240 — XCV600E, XCV1000E**

Pin #	Pin Description	Bank
P240	VCCO	7
P239	TCK	NA
P238	IO	0
P237	IO_L0N	0
P236	IO_VREF_L0P	0
P235	IO_L1N_YY	0
P234	IO_L1P_YY	0
P233	GND	NA
P232	VCCO	0
P231	IO_VREF	0
P230	IO_VREF	0
P229	IO_VREF_L2N_YY	0
P228	IO_L2P_YY	0
P227	GND	NA
P226	VCCO	0
P225	VCCINT	NA
P224	IO_L3N_YY	0
P223	IO_L3P_YY	0
P222	IO_VREF	0[1]
P221	IO_L4N_Y	0
P220	IO_L4P_Y	0
P219	GND	NA
P218	IO_VREF_L5N_Y	0
P217	IO_L5P_Y	0
P216	IO_VREF	0
P215	IO_LVDS_DLL_L6N	0
P214	VCCINT	NA
P213	GCK3	0
P212	VCCO	0
P211	GND	NA

Table 12: **BG432 — XCV300E, XCV400E, XCV600E**

Bank	Pin Description	Pin #
7	IO_L132P_Y	G28
7	IO_L133N	E31
7	IO_L133P	E30
7	IO_L134N_Y	F29
7	IO_VREF_L134P_Y	F28
7	IO_L135N_Y	D31
7	IO_L135P_Y	D30
7	IO_L136N	E29
7	IO_L136P	E28
2	CCLK	D4
3	DONE	AH4
NA	DXN	AH27
NA	DXP	AK29
NA	M0	AH28
NA	M1	AH29
NA	M2	AJ28
NA	PROGRAM	AH3
NA	TCK	D28
NA	TDI	B3
2	TDO	C4
NA	TMS	D29
NA	VCCINT	A10
NA	VCCINT	A17
NA	VCCINT	B23
NA	VCCINT	B26
NA	VCCINT	C7
NA	VCCINT	C14
NA	VCCINT	C19
NA	VCCINT	F1
NA	VCCINT	F30
NA	VCCINT	K3
NA	VCCINT	K29
NA	VCCINT	N2
NA	VCCINT	N29

Table 12: **BG432 — XCV300E, XCV400E, XCV600E**

Bank	Pin Description	Pin #
NA	VCCINT	T1
NA	VCCINT	T29
NA	VCCINT	W2
NA	VCCINT	W31
NA	VCCINT	AB2
NA	VCCINT	AB30
NA	VCCINT	AE29
NA	VCCINT	AF1
NA	VCCINT	AH8
NA	VCCINT	AH24
NA	VCCINT	AJ10
NA	VCCINT	AJ16
NA	VCCINT	AK22
NA	VCCINT	AK13
NA	VCCINT	AK19
0	VCCO	A21
0	VCCO	C29
0	VCCO	D21
1	VCCO	A1
1	VCCO	A11
1	VCCO	D11
2	VCCO	C3
2	VCCO	L4
2	VCCO	L1
3	VCCO	AA1
3	VCCO	AA4
3	VCCO	AJ3
4	VCCO	AH11
4	VCCO	AL1
4	VCCO	AL11
5	VCCO	AH21
5	VCCO	AL21
5	VCCO	AJ29
6	VCCO	AA28
6	VCCO	AA31

Table 15: **BG560 Differential Pin Pair Summary XCV400E, XCV600E, XCV1000E, XCV1600E, XCV2000E**

Pair	Bank	P Pin	N Pin	AO	Other Functions
47	2	F4	C1	14	-
48	2	G5	E3	15	VREF
49	2	D2	G4	16	-
50	2	H5	E2	15	-
51	2	H4	G3	√	VREF
52	2	J5	F1	17	VREF
53	2	J4	H3	14	-
54	2	K5	H2	18	VREF
55	2	J3	K4	19	-
56	2	L5	K3	√	D1
57	2	L4	K2	√	D2
58	2	M5	L3	17	-
59	2	L1	M4	14	-
60	2	N5	M2	15	VREF
61	2	N4	N3	16	-
62	2	N2	P5	15	-
63	2	P4	P3	√	D3
64	2	P2	R5	17	-
65	2	R4	R3	14	-
66	2	R1	T4	18	VREF
67	2	T5	T3	19	VREF
68	2	T2	U3	√	-
69	3	U1	U2	19	VREF
70	3	V2	V4	18	VREF
71	3	V5	V3	14	-
72	3	W1	W3	17	-
73	3	W4	W5	√	VREF
74	3	Y3	Y4	15	-
75	3	AA1	Y5	16	-
76	3	AA3	AA4	15	VREF
77	3	AB3	AA5	14	-

Table 15: **BG560 Differential Pin Pair Summary XCV400E, XCV600E, XCV1000E, XCV1600E, XCV2000E**

Pair	Bank	P Pin	N Pin	AO	Other Functions
78	3	AC1	AB4	17	-
79	3	AC3	AB5	√	D5
80	3	AC4	AD3	√	VREF
81	3	AE1	AC5	4	-
82	3	AD4	AF1	18	VREF
83	3	AF2	AD5	14	-
84	3	AG2	AE4	20	VREF
85	3	AH1	AE5	√	VREF
86	3	AF4	AJ1	15	-
87	3	AJ2	AF5	14	-
88	3	AG4	AK2	15	VREF
89	3	AJ3	AG5	14	-
90	3	AL1	AH4	14	VREF
91	3	AJ4	AH5	√	INIT
92	4	AL4	AJ6	√	-
93	4	AK5	AN3	8	VREF
94	4	AL5	AJ7	√	-
95	4	AM4	AM5	√	VREF
96	4	AK7	AL6	3	-
97	4	AM6	AN6	√	-
98	4	AL7	AJ9	√	VREF
99	4	AN7	AL8	9	VREF
100	4	AM8	AJ10	7	-
101	4	AL9	AM9	7	VREF
102	4	AK10	AN9	2	-
103	4	AL10	AM10	√	VREF
104	4	AL11	AJ12	√	-
105	4	AN11	AK12	8	-
106	4	AL12	AM12	√	-
107	4	AK13	AL13	√	VREF
108	4	AM13	AN13	3	-

Table 16: **FG256 Package — XCV50E, XCV100E, XCV200E, XCV300E**

Bank	Pin Description	Pin #
4	IO_L43P_Y	P12
4	IO_VREF_L43N_Y	R13[2]
4	IO_L44P_YY	N12
4	IO_L44N_YY	T13
4	IO_VREF_L45P_YY	T12
4	IO_L45N_YY	P11
4	IO_L46P_Y	R12
4	IO_L46N_Y	N11
4	IO_VREF_L47P_YY	T11[1]
4	IO_L47N_YY	M11
4	IO_L48P_YY	R11
4	IO_L48N_YY	T10
4	IO_L49P_Y	R10
4	IO_L49N_Y	M10
4	IO_VREF_L50P_Y	P9
4	IO_L50N_Y	T9
4	IO_L51P_Y	N10
4	IO_L51N_Y	R9
4	IO_LVDS_DLL_L52P	N9
5	GCK1	R8
5	IO	N7
5	IO	T7
5	IO_LVDS_DLL_L52N	T8
5	IO_L53P_Y	R7
5	IO_VREF_L53N_Y	P8
5	IO_L54P_Y	P7
5	IO_L54N_Y	T6
5	IO_L55P_YY	M7
5	IO_L55N_YY	R6
5	IO_L56P_YY	P6
5	IO_VREF_L56N_YY	R5[1]
5	IO_L57P_Y	N6
5	IO_L57N_Y	T5
5	IO_L58P_YY	M6
5	IO_VREF_L58N_YY	T4
5	IO_L59P_YY	T3
5	IO_L59N_YY	P5
5	IO_VREF_L60P_Y	T2[2]
5	IO_L60N_Y	N5
6	IO_L61N_YY	M3
6	IO_L61P_YY	R1
6	IO_L62N	M4
6	IO_VREF_L62P	N2[2]
6	IO_L63N_YY	L5
6	IO_L63P_YY	P1
6	IO_VREF_L64N_Y	N1
6	IO_L64P_Y	L3
6	IO_L65N	M2
6	IO_L65P	L4
6	IO_VREF_L66N_Y	M1[1]
6	IO_L66P_Y	K4
6	IO_L67N_YY	L2
6	IO_L67P_YY	L1
6	IO_L68N	K3
6	IO_L68P	K1
6	IO_L69N_YY	K2
6	IO_L69P_YY	K5
6	IO_VREF_L70N_Y	J3
6	IO_L70P_Y	J1
6	IO_L71N	J4
6	IO_L71P	H1
6	IO	J2
7	IO	C2
7	IO_L72N_YY	G1
7	IO_L72P_YY	H4
7	IO_L73N	G5
7	IO_L73P	H2

Table 20: **FG676 — XCV400E, XCV600E**

Bank	Pin Description	Pin #
NA	NC	L2
NA	NC	F6
NA	NC	F25
NA	NC	F21
NA	NC	F2
NA	NC	C26
NA	NC	C25
NA	NC	C2
NA	NC	C1
NA	NC	B6
NA	NC	B26
NA	NC	B24
NA	NC	B21
NA	NC	B16
NA	NC	B11
NA	NC	B1
NA	NC	AF25
NA	NC	AF24
NA	NC	AF2
NA	NC	AE6
NA	NC	AE3
NA	NC	AE26
NA	NC	AE24
NA	NC	AE21
NA	NC	AE16
NA	NC	AE14
NA	NC	AE11
NA	NC	AE1
NA	NC	AD25
NA	NC	AD2
NA	NC	AD1
NA	NC	AA6
NA	NC	AA25
NA	NC	AA21
NA	NC	AA2
NA	NC	A3
NA	NC	A25

Table 20: **FG676 — XCV400E, XCV600E**

Bank	Pin Description	Pin #
NA	NC	A2
NA	NC	A15
NA	VCCINT	G7
NA	VCCINT	G20
NA	VCCINT	H8
NA	VCCINT	H19
NA	VCCINT	J9
NA	VCCINT	J10
NA	VCCINT	J11
NA	VCCINT	J16
NA	VCCINT	J17
NA	VCCINT	J18
NA	VCCINT	K9
NA	VCCINT	K18
NA	VCCINT	L9
NA	VCCINT	L18
NA	VCCINT	T9
NA	VCCINT	T18
NA	VCCINT	U9
NA	VCCINT	U18
NA	VCCINT	V9
NA	VCCINT	V10
NA	VCCINT	V11
NA	VCCINT	V16
NA	VCCINT	V17
NA	VCCINT	V18
NA	VCCINT	Y7
NA	VCCINT	Y20
NA	VCCINT	W8
NA	VCCINT	W19
0	VCCO	J13
0	VCCO	J12
0	VCCO	H9
0	VCCO	H12
0	VCCO	H11

Table 24: **FG860 — XCV1000E, XCV1600E, XCV2000E**

Bank	Pin Description	Pin #
0	IO_VREF_L27N_YY	D27
0	IO_L27P_YY	B25
0	IO_L28N_Y	A25
0	IO_L28P_Y	D26
0	IO_L29N_Y	A24
0	IO_L29P_Y	E25
0	IO_L30N_YY	D25
0	IO_L30P_YY	B24
0	IO_VREF_L31N_YY	E24
0	IO_L31P_YY	A23
0	IO_L32N_Y	C23
0	IO_L32P_Y	E23
0	IO_VREF_L33N_Y	B23[1]
0	IO_L33P_Y	D23
0	IO_LVDS_DLL_L34N	A22
1	GCK2	B22
1	IO	A14
1	IO	A20
1	IO	B11
1	IO	B13
1	IO	C8
1	IO	C18
1	IO	C21
1	IO	D7
1	IO	D10
1	IO	D15
1	IO	D17
1	IO	E20
1	IO_LVDS_DLL_L34P	D22
1	IO_L35N_Y	D21
1	IO_VREF_L35P_Y	B21[1]
1	IO_L36N_Y	D20
1	IO_L36P_Y	A21
1	IO_L37N_YY	C20
1	IO_VREF_L37P_YY	D19
1	IO_L38N_YY	B20

Table 24: **FG860 — XCV1000E, XCV1600E, XCV2000E**

Bank	Pin Description	Pin #
1	IO_L38P_YY	E19
1	IO_L39N_Y	D18
1	IO_L39P_Y	A19
1	IO_L40N_Y	E18
1	IO_L40P_Y	C19
1	IO_L41N_YY	B19
1	IO_VREF_L41P_YY	E17
1	IO_L42N_YY	A18
1	IO_L42P_YY	D16
1	IO_L43N_Y	E16
1	IO_L43P_Y	B18
1	IO_L44N_Y	F16
1	IO_L44P_Y	A17
1	IO_L45N_YY	C17
1	IO_VREF_L45P_YY	E15
1	IO_L46N_YY	B17
1	IO_L46P_YY	D14
1	IO_L47N_Y	A16
1	IO_L47P_Y	E14
1	IO_L48N_Y	C16
1	IO_L48P_Y	D13
1	IO_L49N_Y	B16
1	IO_L49P_Y	D12
1	IO_L50N_Y	A15
1	IO_L50P_Y	E12
1	IO_L51N_YY	C15
1	IO_L51P_YY	C11
1	IO_L52N_YY	B15
1	IO_VREF_L52P_YY	D11
1	IO_L53N_Y	E11
1	IO_L53P_Y	C14
1	IO_L54N_Y	C10
1	IO_L54P_Y	B14
1	IO_L55N_YY	A13
1	IO_VREF_L55P_YY	E10
1	IO_L56N_YY	C13
1	IO_L56P_YY	C9

Table 24: **FG860 — XCV1000E, XCV1600E, XCV2000E**

Bank	Pin Description	Pin #
1	IO_L57N_Y	D9
1	IO_VREF_L57P_Y	A12[2]
1	IO_L58N_Y	E9
1	IO_L58P_Y	C12
1	IO_L59N_YY	B12
1	IO_VREF_L59P_YY	D8
1	IO_L60N_YY	A11
1	IO_L60P_YY	E8
1	IO_L61N_Y	C7
1	IO_L61P_Y	A10
1	IO_L62N_Y	C6
1	IO_L62P_Y	B10
1	IO_L63N_YY	A9
1	IO_VREF_L63P_YY	B9
1	IO_L64N_YY	A8
1	IO_L64P_YY	E7
1	IO_L65N_Y	B8
1	IO_L65P_Y	C5
1	IO_L66N_Y	A7
1	IO_VREF_L66P_Y	A6
1	IO_L67N_Y	B7
1	IO_L67P_Y	D6
1	IO_L68N_Y	A5
1	IO_L68P_Y	C4
1	IO_WRITE_L69N_YY	B6
1	IO_CS_L69P_YY	E6
2	IO	H2
2	IO	H3
2	IO	J1
2	IO	K5
2	IO	M2
2	IO	N1
2	IO	R5
2	IO	U1
2	IO	U4
2	IO	W3

Table 24: **FG860 — XCV1000E, XCV1600E, XCV2000E**

Bank	Pin Description	Pin #
2	IO	Y3
2	IO	AA3
2	IO_DOUT_BUSY_L70P_YY	F5
2	IO_DIN_D0_L70N_YY	D2
2	IO_L71P_Y	E4
2	IO_L71N_Y	E2
2	IO_L72P_Y	D3
2	IO_L72N_Y	F2
2	IO_VREF_L73P_Y	E1
2	IO_L73N_Y	F4
2	IO_L74P	G2
2	IO_L74N	E3
2	IO_L75P_Y	F1
2	IO_L75N_Y	G5
2	IO_VREF_L76P_Y	G1
2	IO_L76N_Y	F3
2	IO_L77P_YY	G4
2	IO_L77N_YY	H1
2	IO_L78P_Y	J2
2	IO_L78N_Y	G3
2	IO_L79P_Y	H5
2	IO_L79N_Y	K2
2	IO_VREF_L80P_YY	H4
2	IO_L80N_YY	K1
2	IO_L81P_YY	L2
2	IO_L81N_YY	L3
2	IO_VREF_L82P_Y	L1[2]
2	IO_L82N_Y	J5
2	IO_L83P_Y	J4
2	IO_L83N_Y	M3
2	IO_VREF_L84P_YY	J3
2	IO_L84N_YY	M1
2	IO_L85P_YY	N2
2	IO_L85N_YY	K4
2	IO_L86P_Y	N3
2	IO_L86N_Y	K3
2	IO_VREF_L87P_YY	L5

Table 26: **FG900 — XCV600E, XCV1000E, XCV1600E**

Bank	Pin Description	Pin #
2	IO	D29[5]
2	IO	G26[4]
2	IO	H24[4]
2	IO	H25[4]
2	IO	H28[5]
2	IO	J25[4]
2	IO	J27[5]
2	IO	K30[4]
2	IO	M24[4]
2	IO	M25[4]
2	IO	N20
2	IO	N23[4]
2	IO	P26[5]
2	IO	P27[5]
2	IO	P30[4]
2	IO	R30
2	IO_DOUT_BUSY_L70P_YY	J22
2	IO_DIN_D0_L70N_YY	E27
2	IO_L71P	C29[4]
2	IO_L71N	D28[3]
2	IO_L72P_Y	G25
2	IO_L72N_Y	E25
2	IO_VREF_L73P_YY	E28[1]
2	IO_L73N_YY	C30
2	IO_L74P_Y	K22[4]
2	IO_L74N_Y	F27[3]
2	IO_L75P_YY	D30
2	IO_L75N_YY	J23
2	IO_VREF_L76P_Y	L21
2	IO_L76N_Y	F28
2	IO_L77P_YY	G28
2	IO_L77N_YY	E30
2	IO_L78P_YY	G27
2	IO_L78N_YY	E29
2	IO_L79P	K23
2	IO_L79N	H26
2	IO_VREF_L80P_YY	F30

Table 26: **FG900 — XCV600E, XCV1000E, XCV1600E**

Bank	Pin Description	Pin #
2	IO_L80N_YY	L22
2	IO_L81P_YY	H27
2	IO_L81N_YY	G29
2	IO_L82P	G30
2	IO_L82N	M21
2	IO_L83P_YY	J24
2	IO_L83N_YY	J26
2	IO_VREF_L84P_YY	H30
2	IO_L84N_YY	L23
2	IO_L85P_YY	K26[4]
2	IO_L85N_YY	J28[3]
2	IO_L86P_YY	J29
2	IO_L86N_YY	K24
2	IO_L87P_YY	K27[4]
2	IO_VREF_L87N_YY	J30
2	IO_D1_L88P	M22
2	IO_D2_L88N	K29
2	IO_L89P_YY	K28[3]
2	IO_L89N_YY	L25[4]
2	IO_L90P	N21
2	IO_L90N	K25
2	IO_L91P_YY	L24
2	IO_L91N_YY	L27
2	IO_L92P_Y	L29[4]
2	IO_L92N_Y	M23[4]
2	IO_L93P_YY	L26
2	IO_L93N_YY	L28
2	IO_VREF_L94P	L30[1]
2	IO_L94N	M27
2	IO_L95P_YY	M26
2	IO_L95N_YY	M29
2	IO_L96P_YY	N29
2	IO_L96N_YY	M30
2	IO_L97P	N25
2	IO_L97N	N27
2	IO_VREF_L98P_YY	N30
2	IO_D3_L98N_YY	P21

Table 27: **FG900 Differential Pin Pair Summary XCV600E, XCV1000E, XCV1600E**

Pair	Bank	P Pin	N Pin	AO	Other Functions
188	5	AA12	AJ12	√	VREF
189	5	AB12	AE11	√	-
190	5	AK12	Y13	2	-
191	5	AG11	AF11	2	-
192	5	AH11	AJ11	2	-
193	5	AE12	AG10	4	-
194	5	AD12	AK11	√	-
195	5	AJ10	AC12	√	VREF
196	5	AK10	AD11	4	-
197	5	AJ9	AE9	4	-
198	5	AH10	AF9	√	VREF
199	5	AH9	AK9	√	-
200	5	AF8	AB11	2	-
201	5	AC11	AG8	2	-
202	5	AK8	AF7	√	VREF
203	5	AG7	AK7	√	-
204	5	AJ7	AD10	1	-
205	5	AH6	AC10	1	-
206	5	AD9	AG6	√	VREF
207	5	AB10	AJ5	√	-
208	5	AD8	AK5	2	-
209	5	AC9	AJ4	2	VREF
210	5	AG5	AK4	2	-
211	5	AH5	AG3	4	-
212	6	AC6	AF3	√	-
213	6	AG2	AH2	NA	-
214	6	AE4	AB9	1	-
215	6	AH1	AE3	4	VREF
216	6	AD6	AB8	3	-
217	6	AA10	AG1	4	-
218	6	AD4	AA9	1	VREF
219	6	AD2	AD5	√	-
220	6	AF2	AD3	4	-
221	6	AA7	AA8	1	-
222	6	Y9	AF1	√	VREF
223	6	AC4	AB6	√	-
224	6	W8	AE1	2	-
225	6	AB4	Y8	4	-
226	6	W9	AB3	4	VREF
227	6	W10	AA5	4	-
228	6	V10	AB1	4	-
229	6	AC1	Y7	4	VREF
230	6	AA3	V11	NA	-
231	6	U10	AA2	4	-
232	6	AA6	W7	1	-
233	6	Y4	Y6	4	-
234	6	V7	AA1	3	-
235	6	Y2	Y3	4	-
236	6	W5	Y5	1	VREF
237	6	W6	W4	√	-
238	6	W2	V6	4	-
239	6	V4	U9	1	-
240	6	T8	AB2	√	VREF
241	6	W1	U5	√	-
242	6	T9	Y1	2	-
243	6	U3	T7	4	-
244	6	V2	T5	4	VREF
245	6	T6	R9	4	-
246	6	U2	T4	4	VREF
247	7	R10	T1	NA	
248	7	R6	R5	4	-
249	7	R4	R8	4	VREF
250	7	R3	R7	4	-
251	7	P6	P10	4	VREF
252	7	P2	P5	4	-
253	7	P4	P7	2	-
254	7	R2	N4	√	-
255	7	P1	N7	√	VREF

Table 28: **FG1156 — XCV1000E, XCV1600E, XCV2000E, XCV2600E, XCV3200E**

Bank	Pin Description	Pin #
0	IO_L6P_YY	H10[5]
0	IO_L7N_Y	D7
0	IO_L7P_Y	B5
0	IO_L8N_Y	K12
0	IO_L8P_Y	E8
0	IO_L9N	B6[4]
0	IO_L9P	F9[5]
0	IO_L10N_YY	G10
0	IO_L10P_YY	C7
0	IO_VREF_L11N_YY	D8
0	IO_L11P_YY	B7
0	IO_L12N	H11[4]
0	IO_L12P	C8[5]
0	IO_L13N_Y	E9
0	IO_L13P_Y	B8
0	IO_VREF_L14N_Y	K13[2]
0	IO_L14P_Y	G11
0	IO_L15N	A8[4]
0	IO_L15P	F10[5]
0	IO_L16N_YY	C9
0	IO_L16P_YY	H12
0	IO_VREF_L17N_YY	D10
0	IO_L17P_YY	A9
0	IO_L18N_Y	F11
0	IO_L18P_Y	A10
0	IO_L19N_Y	K14
0	IO_L19P_Y	C10
0	IO_VREF_L20N_YY	H13
0	IO_L20P_YY	G12
0	IO_L21N_YY	A11
0	IO_L21P_YY	B11
0	IO_L22N_Y	E12
0	IO_L22P_Y	D11
0	IO_L23N_Y	G13

Table 28: **FG1156 — XCV1000E, XCV1600E, XCV2000E, XCV2600E, XCV3200E**

Bank	Pin Description	Pin #
0	IO_L23P_Y	C12
0	IO_L24N_Y	K15
0	IO_L24P_Y	A12
0	IO_L25N_Y	B12
0	IO_L25P_Y	H14
0	IO_L26N_YY	D12
0	IO_L26P_YY	F13
0	IO_VREF_L27N_YY	A13
0	IO_L27P_YY	B13
0	IO_L28N_YY	J15[4]
0	IO_L28P_YY	G14[5]
0	IO_L29N_Y	C13
0	IO_L29P_Y	F14
0	IO_L30N_Y	H15
0	IO_L30P_Y	D13
0	IO_L31N	A14[4]
0	IO_L31P	K16[5]
0	IO_L32N_YY	E14
0	IO_L32P_YY	B14
0	IO_VREF_L33N_YY	G15
0	IO_L33P_YY	D14
0	IO_L34N	J16[4]
0	IO_L34P	D15[5]
0	IO_L35N_Y	F15
0	IO_L35P_Y	B15
0	IO_L36N_Y	A15
0	IO_L36P_Y	E15
0	IO_L37N	G16[4]
0	IO_L37P	A16[5]
0	IO_L38N_YY	F16
0	IO_L38P_YY	J17
0	IO_VREF_L39N_YY	C16
0	IO_L39P_YY	B16
0	IO_L40N_Y	H17