AMD Xilinx - XCV600E-6BG432I 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	3456
Number of Logic Elements/Cells	15552
Total RAM Bits	294912
Number of I/O	316
Number of Gates	985882
Voltage - Supply	1.71V ~ 1.89V
Mounting Type	Surface Mount
Operating Temperature	-40°C ~ 100°C (TJ)
Package / Case	432-LBGA Exposed Pad, Metal
Supplier Device Package	432-MBGA (40x40)
Purchase URL	https://www.e-xfl.com/product-detail/xilinx/xcv600e-6bg432i

The DLL also operates as a clock mirror. By driving the output from a DLL off-chip and then back on again, the DLL can be used to deskew a board level clock among multiple devices.

To guarantee that the system clock is operating correctly prior to the FPGA starting up after configuration, the DLL can delay the completion of the configuration process until after it has achieved lock. For more information about DLL functionality, see the Design Consideration section of the data sheet.

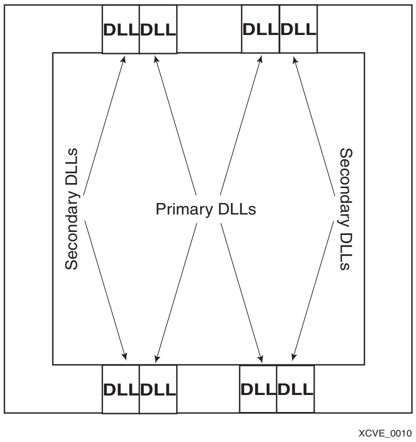


Figure 10: **DLL Locations**

Boundary Scan

Virtex-E devices support all the mandatory Boundary Scan instructions specified in the IEEE standard 1149.1. A Test Access Port (TAP) and registers are provided that implement the EXTEST, INTEST, SAMPLE/PRELOAD, BYPASS, IDCODE, USERCODE, and HIGHZ instructions. The TAP also supports two internal scan chains and configuration/readback of the device.

The JTAG input pins (TDI, TMS, TCK) do not have a V_{CCO} requirement and operate with either 2.5 V or 3.3 V input signalling levels. The output pin (TDO) is sourced from the V_{CCO} in bank 2, and for proper operation of LVTTL 3.3 V levels, the bank should be supplied with 3.3 V.

Boundary Scan operation is independent of individual IOB configurations, and unaffected by package type. All IOBs, including un-bonded ones, are treated as independent 3-state bidirectional pins in a single scan chain. Retention of the bidirectional test capability after configuration facilitates the testing of external interconnections, provided the user design or application is turned off.

Table 6 lists the Boundary Scan instructions supported in Virtex-E FPGAs. Internal signals can be captured during EXTEST by connecting them to un-bonded or unused IOBs. They can also be connected to the unused outputs of IOBs defined as unidirectional input pins.

Before the device is configured, all instructions except USER1 and USER2 are available. After configuration, all instructions are available. During configuration, it is recommended that those operations using the Boundary Scan register (SAMPLE/PRELOAD, INTEST, EXTEST) not be performed.

In addition to the test instructions outlined above, the Boundary Scan circuitry can be used to configure the FPGA, and also to read back the configuration data.

Figure 11 is a diagram of the Virtex-E Series Boundary Scan logic. It includes three bits of Data Register per IOB, the IEEE 1149.1 Test Access Port controller, and the Instruction Register with decodes.

Development System

Virtex-E FPGAs are supported by the Xilinx Foundation and Alliance Series CAE tools. The basic methodology for Virtex-E design consists of three interrelated steps: design entry, implementation, and verification. Industry-standard tools are used for design entry and simulation (for example, Synopsys FPGA Express), while Xilinx provides proprietary architecture-specific tools for implementation.

The Xilinx development system is integrated under the Xilinx Design Manager (XDM™) software, providing designers with a common user interface regardless of their choice of entry and verification tools. The XDM software simplifies the selection of implementation options with pull-down menus and on-line help.

Application programs ranging from schematic capture to Placement and Routing (PAR) can be accessed through the XDM software. The program command sequence is generated prior to execution, and stored for documentation.

Several advanced software features facilitate Virtex-E design. RPMs, for example, are schematic-based macros with relative location constraints to guide their placement. They help ensure optimal implementation of common functions.

For HDL design entry, the Xilinx FPGA Foundation development system provides interfaces to the following synthesis design environments.

- Synopsys (FPGA Compiler, FPGA Express)
- Exemplar (Spectrum)
- Synplicity (Synplify)

For schematic design entry, the Xilinx FPGA Foundation and Alliance development system provides interfaces to the following schematic-capture design environments.

- Mentor Graphics V8 (Design Architect, QuickSim II)
- Viewlogic Systems (Viewdraw)

Third-party vendors support many other environments.

A standard interface-file specification, Electronic Design Interchange Format (EDIF), simplifies file transfers into and out of the development system.

Virtex-E FPGAs are supported by a unified library of standard functions. This library contains over 400 primitives and macros, ranging from 2-input AND gates to 16-bit accumulators, and includes arithmetic functions, comparators, counters, data registers, decoders, encoders, I/O functions, latches, Boolean functions, multiplexers, shift registers, and barrel shifters.

The “soft macro” portion of the library contains detailed descriptions of common logic functions, but does not contain any partitioning or placement information. The performance of these macros depends, therefore, on the partitioning and placement obtained during implementation.

RPMs, on the other hand, do contain predetermined partitioning and placement information that permits optimal implementation of these functions. Users can create their own library of soft macros or RPMs based on the macros and primitives in the standard library.

The design environment supports hierarchical design entry, with high-level schematics that comprise major functional blocks, while lower-level schematics define the logic in these blocks. These hierarchical design elements are automatically combined by the implementation tools. Different design entry tools can be combined within a hierarchical design, thus allowing the most convenient entry method to be used for each portion of the design.

Design Implementation

The place-and-route tools (PAR) automatically provide the implementation flow described in this section. The partitioner takes the EDIF net list for the design and maps the logic into the architectural resources of the FPGA (CLBs and IOBs, for example). The placer then determines the best locations for these blocks based on their interconnections and the desired performance. Finally, the router interconnects the blocks.

The PAR algorithms support fully automatic implementation of most designs. For demanding applications, however, the user can exercise various degrees of control over the process. User partitioning, placement, and routing information is optionally specified during the design-entry process. The implementation of highly structured designs can benefit greatly from basic floor planning.

The implementation software incorporates Timing Wizard® timing-driven placement and routing. Designers specify timing requirements along entire paths during design entry. The timing path analysis routines in PAR then recognize these user-specified requirements and accommodate them.

Timing requirements are entered on a schematic in a form directly relating to the system requirements, such as the targeted clock frequency, or the maximum allowable delay between two registers. In this way, the overall performance of the system along entire signal paths is automatically tailored to user-generated specifications. Specific timing information for individual nets is unnecessary.

Design Verification

In addition to conventional software simulation, FPGA users can use in-circuit debugging techniques. Because Xilinx devices are infinitely reprogrammable, designs can be verified in real time without the need for extensive sets of software simulation vectors.

The development system supports both software simulation and in-circuit debugging techniques. For simulation, the system extracts the post-layout timing information from the design database, and back-annotates this information into the net list for use by the simulator. Alternatively, the user can verify timing-critical portions of the design using the TRCE® static timing analyzer.

Data Output Bus—DO[A|B]<#:0>

The data out bus reflects the contents of the memory cells referenced by the address bus at the last active clock edge. During a write operation, the data out bus reflects the data in bus. The width of this bus equals the width of the port. The allowed widths appear in Table 15.

Inverting Control Pins

The four control pins (CLK, EN, WE and RST) for each port have independent inversion control as a configuration option.

Address Mapping

Each port accesses the same set of 4096 memory cells using an addressing scheme dependent on the width of the port.

The physical RAM location addressed for a particular width are described in the following formula (of interest only when the two ports use different aspect ratios).

$$\text{Start} = ((\text{ADDR}_{\text{port}} + 1) * \text{Width}_{\text{port}}) - 1$$

$$\text{End} = \text{ADDR}_{\text{port}} * \text{Width}_{\text{port}}$$

Conflict Resolution

The block SelectRAM+ memory is a true dual-read/write port RAM that allows simultaneous access of the same memory cell from both ports. When one port writes to a given memory cell, the other port must not address that memory cell (for a write or a read) within the clock-to-clock setup window. The following lists specifics of port and memory cell write conflict resolution.

- If both ports write to the same memory cell simultaneously, violating the clock-to-clock setup requirement, consider the data stored as invalid.
- If one port attempts a read of the same memory cell the other simultaneously writes, violating the clock-to-clock setup requirement, the following occurs.
  - The write succeeds
  - The data out on the writing port accurately reflects the data written.
  - The data out on the reading port is invalid.

Conflicts do not cause any physical damage.

Table 16 shows low order address mapping for each port width.

Table 16: **Port Address Mapping**

Port Width	**Port Addresses**																
1	4095...	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
2	2047...	07		06		05		04		03		02		01		00	
4	1023...	03				02				01				00			
8	511...	01								00							
16	255...	00															

Creating Larger RAM Structures

The block SelectRAM+ columns have specialized routing to allow cascading blocks together with minimal routing delays. This achieves wider or deeper RAM structures with a smaller timing penalty than when using normal routing channels.

Location Constraints

Block SelectRAM+ instances can have LOC properties attached to them to constrain the placement. The block SelectRAM+ placement locations are separate from the CLB location naming convention, allowing the LOC properties to transfer easily from array to array.

The LOC properties use the following form.

LOC = RAMB4_R#C#

RAMB4_R0C0 is the upper left RAMB4 location on the device.

Single Port Timing

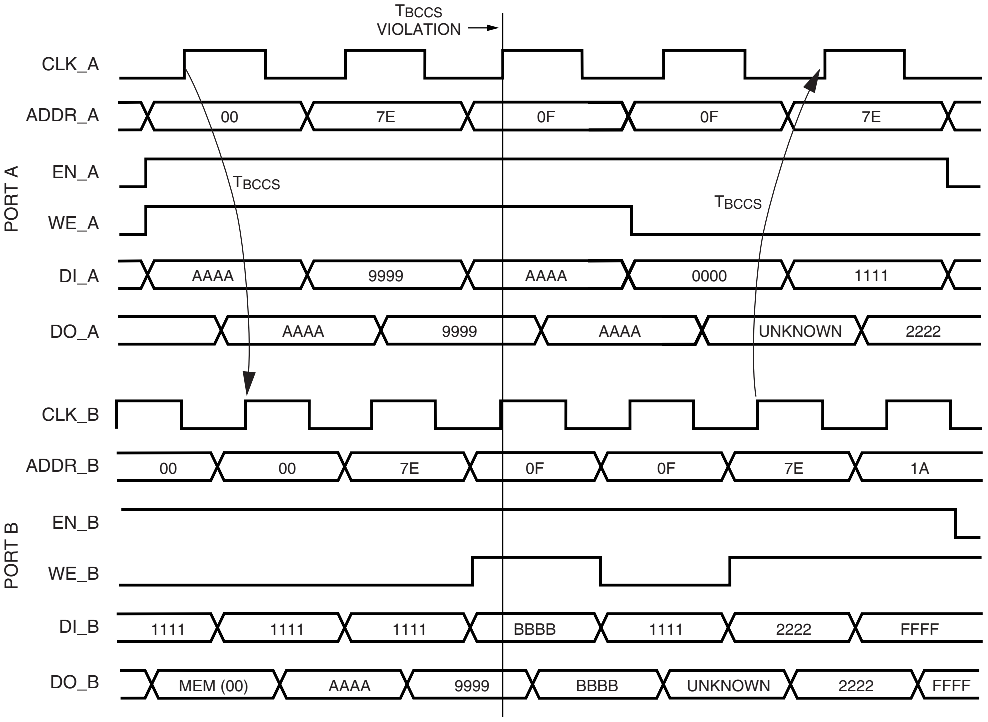
Figure 33 shows a timing diagram for a single port of a block SelectRAM+ memory. The block SelectRAM+ AC switching characteristics are specified in the data sheet. The block SelectRAM+ memory is initially disabled.

At the first rising edge of the CLK pin, the ADDR, DI, EN, WE, and RST pins are sampled. The EN pin is High and the WE pin is Low indicating a read operation. The DO bus contains the contents of the memory location, 0x00, as indicated by the ADDR bus.

At the second rising edge of the CLK pin, the ADDR, DI, EN, WR, and RST pins are sampled again. The EN and WE pins are High indicating a write operation. The DO bus mirrors the DI bus. The DI bus is written to the memory location 0x0F.

At the third rising edge of the CLK pin, the ADDR, DI, EN, WR, and RST pins are sampled again. The EN pin is High and the WE pin is Low indicating a read operation. The DO bus contains the contents of the memory location 0x7E as indicated by the ADDR bus.

At the fourth rising edge of the CLK pin, the ADDR, DI, EN, WR, and RST pins are sampled again. The EN pin is Low



ds022_035_121399

Figure 34: **Timing Diagram for a True Dual-port Read/Write Block SelectRAM+ Memory**

At the third rising edge of CLKA, the T_{BCCS} parameter is violated with two writes to memory location 0x0F. The DOA and DOB buses reflect the contents of the DIA and DIB buses, but the stored value at 0x0F is invalid.

At the fourth rising edge of CLKA, a read operation is performed at memory location 0x0F and invalid data is present on the DOA bus. Port B also executes a read operation to memory location 0x0F and also reads invalid data.

At the fifth rising edge of CLKA a read operation is performed that does not violate the T_{BCCS} parameter to the previous write of 0x7E by Port B. THe DOA bus reflects the recently written value by Port B.

Initialization

The block SelectRAM+ memory can initialize during the device configuration sequence. The 16 initialization properties of 64 hex values each (a total of 4096 bits) set the initialization of each RAM. These properties appear in Table 17. Any initialization properties not explicitly set configure as zeros. Partial initialization strings pad with zeros. Initialization strings greater than 64 hex values generate an error. The RAMs can be simulated with the initialization values using generics in VHDL simulators and parameters in Verilog simulators.

Initialization in VHDL and Synopsys

The block SelectRAM+ structures can be initialized in VHDL for both simulation and synthesis for inclusion in the EDIF output file. The simulation of the VHDL code uses a generic to pass the initialization. Synopsys FPGA compiler does not presently support generics. The initialization values instead attach as attributes to the RAM by a built-in Synopsys dc_script. The translate_off statement stops synthesis translation of the generic statements. The following code illustrates a module that employs these techniques.

Table 17: **RAM Initialization Properties**

Property	Memory Cells
INIT_00	255 to 0
INIT_01	511 to 256
INIT_02	767 to 512
INIT_03	1023 to 768
INIT_04	1279 to 1024
INIT_05	1535 to 1280
INIT_06	1791 to 2047
INIT_07	2047 to 1792
INIT_08	2303 to 2048
INIT_09	2559 to 2304
INIT_0a	2815 to 2560
INIT_0b	3071 to 2816
INIT_0c	3327 to 3072
INIT_0d	3583 to 3328
INIT_0e	3839 to 3584
INIT_0f	4095 to 3840

Initialization in Verilog and Synopsys

The block SelectRAM+ structures can be initialized in Verilog for both simulation and synthesis for inclusion in the EDIF output file. The simulation of the Verilog code uses a defparam to pass the initialization. The Synopsys FPGA compiler does not presently support defparam. The initialization values instead attach as attributes to the RAM by a built-in Synopsys dc_script. The translate_off statement stops synthesis translation of the defparam statements. The following code illustrates a module that employs these techniques.

Design Examples

Creating a 32-bit Single-Port RAM

The true dual-read/write port functionality of the block SelectRAM+ memory allows a single port, 128 deep by 32-bit wide RAM to be created using a single block SelectRAM+ cell as shown in Figure 35.

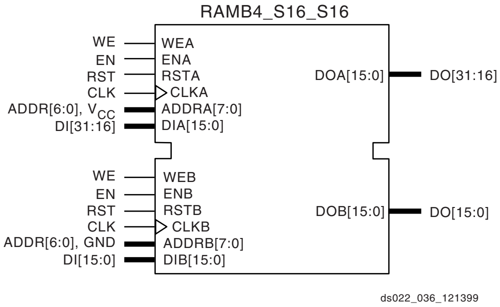


Figure 35: **Single Port 128 x 32 RAM**

Interleaving the memory space, setting the LSB of the address bus of Port A to 1 (V_{CC}), and the LSB of the address bus of Port B to 0 (GND), allows a 32-bit wide single port RAM to be created.

Creating Two Single-Port RAMs

The true dual-read/write port functionality of the block SelectRAM+ memory allows a single RAM to be split into two single port memories of 2K bits each as shown in Figure 36.

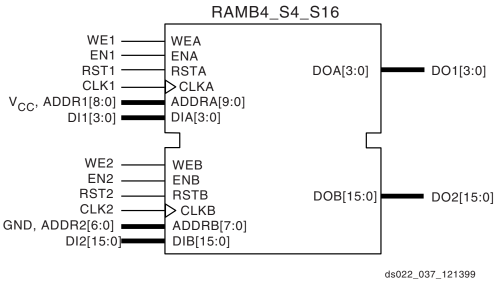


Figure 36: **512 x 4 RAM and 128 x 16 RAM**

In this example, a 512K x 4 RAM (Port A) and a 128 x 16 RAM (Port B) are created out of a single block SelectRAM+. The address space for the RAM is split by fixing the MSB of Port A to 1 (V_{CC}) for the upper 2K bits and the MSB of Port B to 0 (GND) for the lower 2K bits.

Block Memory Generation

The CoreGen program generates memory structures using the block SelectRAM+ features. This program outputs VHDL or Verilog simulation code templates and an EDIF file for inclusion in a design.

Virtex-E Pin-to-Pin Output Parameter Guidelines

All devices are 100% functionally tested. Listed below are representative values for typical pin locations and normal clock loading. Values are expressed in nanoseconds unless otherwise noted.

Global Clock Input to Output Delay for LVTTL, 12 mA, Fast Slew Rate, *with* DLL

Description[1]	Symbol	Device	Speed Grade[2, 3]				Units
			Min	-8	-7	-6	
LVTTL Global Clock Input to Output Delay using Output Flip-flop, 12 mA, Fast Slew Rate, *with* DLL. For data *output* with different standards, adjust the delays with the values shown in **IOB Output Switching Characteristics Standard Adjustments**, page 10.	$T_{ICKOFDLL}$	XCV50E	1.0	3.1	3.1	3.1	ns
		XCV100E	1.0	3.1	3.1	3.1	ns
		XCV200E	1.0	3.1	3.1	3.1	ns
		XCV300E	1.0	3.1	3.1	3.1	ns
		XCV400E	1.0	3.1	3.1	3.1	ns
		XCV600E	1.0	3.1	3.1	3.1	ns
		XCV1000E	1.0	3.1	3.1	3.1	ns
		XCV1600E	1.0	3.1	3.1	3.1	ns
		XCV2000E	1.0	3.1	3.1	3.1	ns
		XCV2600E	1.0	3.1	3.1	3.1	ns
		XCV3200E	1.0	3.1	3.1	3.1	ns

Notes:

1. Listed above are representative values where one global clock input drives one vertical clock line in each accessible column, and where all accessible IOB and CLB flip-flops are clocked by the global clock net.
2. Output timing is measured at 50% V_{CC} threshold with 35 pF external capacitive load. For other I/O standards and different loads, see Table 3 and Table 4.
3. DLL output jitter is already included in the timing calculation.

Global Clock Input to Output Delay for LVTTL, 12 mA, Fast Slew Rate, *without* DLL

Description[1]	Symbol	Device	Speed Grade[2]				Units
			Min	**-8**	**-7**	**-6**	
LVTTL Global Clock Input to Output Delay using Output Flip-flop, 12 mA, Fast Slew Rate, *without* DLL. For data *output* with different standards, adjust the delays with the values shown in **IOB Output Switching Characteristics Standard Adjustments**, page 10.	T_{ICKOF}	XCV50E	1.5	4.2	4.4	4.6	ns
		XCV100E	1.5	4.2	4.4	4.6	ns
		XCV200E	1.5	4.3	4.5	4.7	ns
		XCV300E	1.5	4.3	4.5	4.7	ns
		XCV400E	1.5	4.4	4.6	4.8	ns
		XCV600E	1.6	4.5	4.7	4.9	ns
		XCV1000E	1.7	4.6	4.8	5.0	ns
		XCV1600E	1.8	4.7	4.9	5.1	ns
		XCV2000E	1.8	4.8	5.0	5.2	ns
		XCV2600E	2.0	5.0	5.2	5.4	ns
		XCV3200E	2.2	5.2	5.4	5.6	ns

Notes:

1. Listed above are representative values where one global clock input drives one vertical clock line in each accessible column, and where all accessible IOB and CLB flip-flops are clocked by the global clock net.
2. Output timing is measured at 50% V_{CC} threshold with 35 pF external capacitive load. For other I/O standards and different loads, see Table 3 and Table 4.

Global Clock Set-Up and Hold for LVTTL Standard, *without* DLL

Description[1]	Symbol	Device	Speed Grade[2, 3] Min	-8	-7	-6	Units
Input Setup and Hold Time Relative to Global Clock Input Signal for LVTTL Standard. For data input with different standards, adjust the setup time delay by the values shown in **IOB Input Switching Characteristics Standard Adjustments**, page 8.							
Full Delay Global Clock and IFF, without DLL	T_{PSFD}/T_{PHFD}	XCV50E	1.8 / 0	1.8 / 0	1.8 / 0	1.8 / 0	ns
		XCV100E	1.8 / 0	1.8 / 0	1.8 / 0	1.8 / 0	ns
		XCV200E	1.9 / 0	1.9 / 0	1.9 / 0	1.9 / 0	ns
		XCV300E	2.0 / 0	2.0 / 0	2.0 / 0	2.0 / 0	ns
		XCV400E	2.0 / 0	2.0 / 0	2.0 / 0	2.0 / 0	ns
		XCV600E	2.1 / 0	2.1 / 0	2.1 / 0	2.1 / 0	ns
		XCV1000E	2.3 / 0	2.3 / 0	2.3 / 0	2.3 / 0	ns
		XCV1600E	2.5 / 0	2.5 / 0	2.5 / 0	2.5 / 0	ns
		XCV2000E	2.5 / 0	2.5 / 0	2.5 / 0	2.5 / 0	ns
		XCV2600E	2.7 / 0	2.7 / 0	2.7 / 0	2.7 / 0	ns
		XCV3200E	2.8 / 0	2.8 / 0	2.8 / 0	2.8 / 0	ns

Notes:

1. IFF = Input Flip-Flop or Latch
2. Setup time is measured relative to the Global Clock input signal with the fastest route and the lightest load. Hold time is measured relative to the Global Clock input signal with the slowest route and heaviest load.
3. 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.

Table 10: **BG352 — XCV100E, XCV200E, XCV300E**

Bank	Pin Description	Pin #
0	IO	C15
0	IO	B15[1]
0	IO_LVDS_DLL_L9N	A15
0	GCK3	D14
1	GCK2	B14
1	IO_LVDS_DLL_L9P	A13
1	IO	B13[1]
1	IO_L10N	C13
1	IO_L10P	A12
1	IO_L11N_Y	B12
1	IO_VREF_1_L11P_Y	C12
1	IO_L12N_Y	A11
1	IO_L12P_Y	B11
1	IO	B10[1]
1	IO_L13N	C11
1	IO_L13P	D11
1	IO	A9[1]
1	IO_L14N_YY	B9
1	IO_L14P_YY	C10
1	IO_L15N_YY	B8
1	IO_VREF_1_L15P_YY	C9
1	IO_L16N _Y	D9
1	IO_L16P _Y	A7
1	IO	B7
1	IO	C8[1]
1	IO	D8[1]
1	IO_L17N_YY	A6
1	IO_VREF_1_L17P_YY	B6
1	IO_L18N_YY	C7
1	IO_L18P_YY	A4
1	IO	B5[1]
1	IO_L19N_YY	C6
1	IO_VREF_1_L19P_YY	D6[2]
1	IO	B4
1	IO	C5[1]
1	IO	A3[1]
1	IO_WRITE_L20N_YY	D5
1	IO_CS_L20P_YY	C4
2	IO_DOUT_BUSY_L21P_YY	E4
2	IO_DIN_D0_L21N_YY	D3
2	IO	C2[1]
2	IO	E3[1]
2	IO	F4
2	IO_VREF_2_L22P_YY	D2[2]
2	IO_L22N_YY	C1
2	IO	D1[1]
2	IO_L23P_YY	G4
2	IO_L23N_YY	F3
2	IO_VREF_2_L24P_Y	E2
2	IO_L24N_Y	F2
2	IO	G3[1]
2	IO	G2[1]
2	IO_L25P	F1
2	IO_L25N	J4
2	IO	H3
2	IO_VREF_2_L26P _Y	H2
2	IO_D1_L26N _Y	G1
2	IO_D2_L27P_YY	J3
2	IO_L27N_YY	J2
2	IO	K3[1]
2	IO_L28P	J1
2	IO_L28N	L4
2	IO	K2[1]
2	IO_L29P_YY	L3
2	IO_L29N_YY	L2
2	IO_VREF_2_L30P _Y	M4

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

Bank	Pin Description	Pin #
6	IO	AA30
6	IO	AC30
6	IO	AD29
6	IO	U31
6	IO	W28
6	IO_L103N_YY	AJ30
6	IO_L103P_YY	AH30
6	IO_L104N	AG28
6	IO_L104P	AH31
6	IO_L105N_Y	AG29
6	IO_L105P_Y	AG30
6	IO_VREF_L106N_Y	AF28
6	IO_L106P_Y	AG31
6	IO_L107N	AF29
6	IO_L107P	AF30
6	IO_L108N_Y	AE28
6	IO_L108P_Y	AF31
6	IO_VREF_L109N_YY	AE30
6	IO_L109P_YY	AD28
6	IO_L110N_Y	AD30
6	IO_L110P_Y	AD31
6	IO_VREF_L111N_Y	AC28[1]
6	IO_L111P_Y	AC29
6	IO_VREF_L112N_YY	AB28
6	IO_L112P_YY	AB29
6	IO_L113N_YY	AB31
6	IO_L113P_YY	AA29
6	IO_L114N_Y	Y28
6	IO_L114P_Y	Y29
6	IO_L115N_Y	Y30
6	IO_L115P_Y	Y31
6	IO_L116N_Y	W29
6	IO_L116P_Y	W30
6	IO_VREF_L117N_YY	V28
6	IO_L117P_YY	V29
6	IO_L118N_Y	V30
6	IO_L118P_Y	U29
6	IO_VREF_L119N_Y	U28[2]
6	IO_L119P_Y	U30
6	IO	T30
7	IO	C30
7	IO	H29
7	IO	H31
7	IO	L29
7	IO	M31
7	IO	R28
7	IO_L120N_YY	T31
7	IO_L120P_YY	R29
7	IO_L121N_Y	R30
7	IO_VREF_L121P_Y	R31[2]
7	IO_L122N_Y	P29
7	IO_L122P_Y	P28
7	IO_L123N_YY	P30
7	IO_VREF_L123P_YY	N30
7	IO_L124N_Y	N28
7	IO_L124P_Y	N31
7	IO_L125N_Y	M29
7	IO_L125P_Y	M28
7	IO_L126N_Y	M30
7	IO_L126P_Y	L30
7	IO_L127N_YY	K31
7	IO_L127P_YY	K30
7	IO_L128N_YY	K28
7	IO_VREF_L128P_YY	J30
7	IO_L129N_Y	J29
7	IO_VREF_L129P_Y	J28[1]
7	IO_L130N_Y	H30
7	IO_L130P_Y	G30
7	IO_L131N_YY	H28
7	IO_VREF_L131P_YY	F31
7	IO_L132N_Y	G29

Table 22: **FG680 - XCV600E, XCV1000E, XCV1600E, XCV2000E**

Bank	Pin Description	Pin #
4	IO_VREF_L132P_YY	AV8
4	IO_L132N_YY	AU9
4	IO_L133P_Y	AW8
4	IO_L133N_Y	AT10
4	IO_VREF_L134P_Y	AV9[3]
4	IO_L134N_Y	AU10
4	IO_L135P_YY	AW9
4	IO_L135N_YY	AT11
4	IO_VREF_L136P_YY	AV10
4	IO_L136N_YY	AU11
4	IO_L137P_Y	AW10
4	IO_L137N_Y	AU12
4	IO_L138P_Y	AV11
4	IO_L138N_Y	AT13
4	IO_VREF_L139P_YY	AW11
4	IO_L139N_YY	AU13
4	IO_L140P_YY	AT14
4	IO_L140N_YY	AV12
4	IO_L141P_Y	AU14
4	IO_L141N_Y	AW12
4	IO_L142P_Y	AT15
4	IO_L142N_Y	AV13
4	IO_L143P_YY	AU15
4	IO_L143N_YY	AW13
4	IO_VREF_L144P_YY	AV14[1]
4	IO_L144N_YY	AT16
4	IO_L145P_Y	AW14
4	IO_L145N_Y	AU16
4	IO_L146P_Y	AV15
4	IO_L146N_Y	AR17
4	IO_L147P_YY	AW15
4	IO_L147N_YY	AT17
4	IO_VREF_L148P_YY	AU17
4	IO_L148N_YY	AV16
4	IO_L149P_Y	AR18
4	IO_L149N_Y	AW16
4	IO_L150P_Y	AT18
4	IO_L150N_Y	AV17
4	IO_L151P_YY	AU18
4	IO_L151N_YY	AW17
4	IO_VREF_L152P_YY	AT19
4	IO_L152N_YY	AV18
4	IO_L153P_Y	AU19
4	IO_L153N_Y	AW18
4	IO_VREF_L154P	AU21[2]
4	IO_L154N	AV19
4	IO_LVDS_DLL_L155P	AT21
5	GCK1	AU22
5	IO	AT34
5	IO	AW20
5	IO_LVDS_DLL_L155N	AT22
5	IO_VREF_L156P_Y	AV20[2]
5	IO_L156N_Y	AR22
5	IO_L157P_YY	AV23
5	IO_VREF_L157N_YY	AW21
5	IO_L158P_YY	AU23
5	IO_L158N_YY	AV21
5	IO_L159P_Y	AT23
5	IO_L159N_Y	AW22
5	IO_L160P_Y	AR23
5	IO_L160N_Y	AV22
5	IO_L161P_YY	AV24
5	IO_VREF_L161N_YY	AW23
5	IO_L162P_YY	AW24
5	IO_L162N_YY	AU24
5	IO_L163P_Y	AW25
5	IO_L163N_Y	AT24
5	IO_L164P_Y	AV25
5	IO_L164N_Y	AU25
5	IO_L165P_YY	AW26
5	IO_VREF_L165N_YY	AT25[1]

Table 23: **FG680 Differential Pin Pair Summary XCV600E, XCV1000E, XCV1600E, XCV2000E**

Pair	Bank	P Pin	N Pin	AO	Other Functions
52	1	B8	A8	√	VREF
53	1	A7	D9	√	-
54	1	B7	C8	3	-
55	1	A6	D8	3	-
56	1	B6	C7	√	VREF
57	1	A5	D7	√	-
58	1	B5	C6	5	VREF
59	1	A4	D6	5	-
60	1	D5	B4	√	CS
61	2	E3	C2	√	DIN, D0
62	2	D3	F3	6	-
63	2	D2	G4	4	VREF
64	2	G3	E2	4	-
65	2	H4	E1	6	VREF
66	2	H3	F2	√	-
67	2	J4	F1	4	-
68	2	J3	G2	6	-
69	2	G1	K4	√	VREF
70	2	H2	K3	√	-
71	2	H1	L4	7	VREF
72	2	J2	L3	4	-
73	2	J1	M3	√	VREF
74	2	K2	N4	√	-
75	2	K1	N3	4	-
76	2	L2	P4	√	D1
77	2	P3	L1	√	D2
78	2	R4	M2	6	-
79	2	R3	M1	4	-
80	2	T4	N2	4	-
81	2	N1	T3	6	VREF
82	2	P2	U5	√	-
83	2	P1	U4	4	-
84	2	R2	U3	6	-
85	2	V5	R1	√	D3
86	2	V4	T2	√	-
87	2	V3	T1	7	-
88	2	W4	U2	4	-
89	2	W3	U1	√	VREF
90	2	AA3	V2	√	-
91	2	AA4	V1	4	VREF
92	2	AB2	W2	√	-
93	3	AB4	W1	4	VREF
94	3	AB5	Y2	√	-
95	3	AC2	Y1	√	VREF
96	3	AC3	AA1	4	-
97	3	AC4	AA2	7	-
98	3	AC5	AB1	√	-
99	3	AD3	AC1	√	VREF
100	3	AD1	AD4	6	-
101	3	AD2	AE3	4	-
102	3	AE1	AE4	√	-
103	3	AE2	AF3	6	VREF
104	3	AF4	AF1	4	-
105	3	AG3	AF2	4	-
106	3	AG4	AG1	6	-
107	3	AH3	AG2	√	D5
108	3	AH1	AJ2	√	VREF
109	3	AH2	AJ3	4	-
110	3	AJ1	AJ4	√	-
111	3	AK1	AK3	√	VREF
112	3	AK2	AK4	4	-
113	3	AL1	AL2	7	VREF
114	3	AM1	AL3	√	-
115	3	AM2	AL4	√	VREF
116	3	AM3	AN1	6	-
117	3	AM4	AP1	4	-
118	3	AN2	AP2	√	-
119	3	AN3	AR1	6	VREF

FG860 Fine-Pitch Ball Grid Array Package

XCV1000E, XCV1600E, and XCV2000E devices in the FG680 fine-pitch Ball Grid Array package 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 24, see Table 25 for Differential Pair information.

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

Bank	Pin Description	Pin #
0	GCK3	C22
0	IO	A26
0	IO	B31
0	IO	B34
0	IO	C24
0	IO	C29
0	IO	C34
0	IO	D24
0	IO	D36
0	IO	D40
0	IO	E26
0	IO	E28
0	IO	E35
0	IO_L0N_Y	A38
0	IO_L0P_Y	D38
0	IO_L1N_Y	B37
0	IO_L1P_Y	E37
0	IO_VREF_L2N_Y	A37
0	IO_L2P_Y	C39
0	IO_L3N_Y	B36
0	IO_L3P_Y	C38
0	IO_L4N_YY	A36
0	IO_L4P_YY	B35
0	IO_VREF_L5N_YY	A35
0	IO_L5P_YY	D37
0	IO_L6N_Y	C37
0	IO_L6P_Y	A34
0	IO_L7N_Y	E36
0	IO_L7P_Y	B33
0	IO_L8N_YY	A33
0	IO_L8P_YY	C32
0	IO_VREF_L9N_YY	C36
0	IO_L9P_YY	B32
0	IO_L10N_Y	A32
0	IO_L10P_Y	D35
0	IO_VREF_L11N_Y	C31[2]
0	IO_L11P_Y	C35
0	IO_L12N_YY	E34
0	IO_L12P_YY	A31
0	IO_VREF_L13N_YY	D34
0	IO_L13P_YY	C30
0	IO_L14N_Y	B30
0	IO_L14P_Y	E33
0	IO_L15N_Y	A30
0	IO_L15P_Y	D33
0	IO_VREF_L16N_YY	C33
0	IO_L16P_YY	B29
0	IO_L17N_YY	E32
0	IO_L17P_YY	A29
0	IO_L18N_Y	D32
0	IO_L18P_Y	C28
0	IO_L19N_Y	E31
0	IO_L19P_Y	B28
0	IO_L20N_Y	D31
0	IO_L20P_Y	A28
0	IO_L21N_Y	D30
0	IO_L21P_Y	C27
0	IO_L22N_YY	E29
0	IO_L22P_YY	B27
0	IO_VREF_L23N_YY	D29
0	IO_L23P_YY	A27
0	IO_L24N_Y	C26
0	IO_L24P_Y	D28
0	IO_L25N_Y	B26
0	IO_L25P_Y	F27
0	IO_L26N_YY	E27
0	IO_L26P_YY	C25

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

Bank	Pin Description	Pin #
2	IO_D1_L87N_YY	P2
2	IO_D2_L88P_YY	P3
2	IO_L88N_YY	L4
2	IO_L89P_Y	P1
2	IO_L89N_Y	R2
2	IO_L90P_Y	M5
2	IO_L90N_Y	R3
2	IO_L91P_Y	M4
2	IO_L91N_Y	R1
2	IO_L92P	N4
2	IO_L92N	T2
2	IO_L93P_Y	P5
2	IO_L93N_Y	T3
2	IO_VREF_L94P_Y	P4
2	IO_L94N_Y	T1
2	IO_L95P_YY	U2
2	IO_L95N_YY	R4
2	IO_L96P_Y	U3
2	IO_L96N_Y	T5
2	IO_L97P_Y	T4
2	IO_L97N_Y	V2
2	IO_VREF_L98P_YY	U5
2	IO_D3_L98N_YY	V3
2	IO_L99P_YY	V1
2	IO_L99N_YY	V5
2	IO_L100P_Y	W2
2	IO_L100N_Y	V4
2	IO_L101P_Y	W5
2	IO_L101N_Y	W1
2	IO_VREF_L102P_YY	Y2
2	IO_L102N_YY	W4
2	IO_L103P_YY	Y1
2	IO_L103N_YY	Y5
2	IO_VREF_L104P_Y	AA1[1]
2	IO_L104N_Y	Y4
2	IO_L105P_YY	AA4
2	IO_L105N_YY	AA2
3	IO	AB4
3	IO	AC2
3	IO	AD1
3	IO	AE3
3	IO	AF4
3	IO	AH5
3	IO	AJ2
3	IO	AL1
3	IO	AM3
3	IO	AP3
3	IO	AR5
3	IO	AU4
3	IO	AB2
3	IO_L106P_Y	AB3
3	IO_VREF_L106N_Y	AC4[1]
3	IO_L107P_YY	AB1
3	IO_L107N_YY	AC5
3	IO_L108P_YY	AD4
3	IO_VREF_L108N_YY	AC3
3	IO_L109P_Y	AC1
3	IO_L109N_Y	AD5
3	IO_L110P_Y	AE4
3	IO_L110N_Y	AD3
3	IO_L111P_YY	AE5
3	IO_L111N_YY	AD2
3	IO_D4_L112P_YY	AE1
3	IO_VREF_L112N_YY	AF5
3	IO_L113P_Y	AE2
3	IO_L113N_Y	AG4
3	IO_L114P_Y	AG5
3	IO_L114N_Y	AF1
3	IO_L115P_YY	AH4
3	IO_L115N_YY	AF2
3	IO_L116P_Y	AF3
3	IO_VREF_L116N_Y	AJ4
3	IO_L117P_Y	AG1

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

Bank	Pin Description	Pin #
3	IO_L117N_Y	AJ5
3	IO_L118P	AG2
3	IO_L118N	AK4
3	IO_L119P_Y	AG3
3	IO_L119N_Y	AL4
3	IO_L120P_Y	AH1
3	IO_L120N_Y	AL5
3	IO_L121P_Y	AH2
3	IO_L121N_Y	AM4
3	IO_L122P_YY	AH3
3	IO_D5_L122N_YY	AM5
3	IO_D6_L123P_YY	AJ1
3	IO_VREF_L123N_YY	AN3
3	IO_L124P_Y	AN4
3	IO_L124N_Y	AJ3
3	IO_L125P_YY	AN5
3	IO_L125N_YY	AK1
3	IO_L126P_YY	AK2
3	IO_VREF_L126N_YY	AP4
3	IO_L127P_Y	AK3
3	IO_L127N_Y	AP5
3	IO_L128P_Y	AR3
3	IO_VREF_L128N_Y	AL2[2]
3	IO_L129P_YY	AR4
3	IO_L129N_YY	AL3
3	IO_L130P_YY	AM1
3	IO_VREF_L130N_YY	AT3
3	IO_L131P_Y	AM2
3	IO_L131N_Y	AT4
3	IO_L132P_Y	AT5
3	IO_L132N_Y	AN1
3	IO_L133P_YY	AU3
3	IO_L133N_YY	AN2
3	IO_L134P_Y	AP1
3	IO_VREF_L134N_Y	AP2
3	IO_L135P_Y	AR1
3	IO_L135N_Y	AV3

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

Bank	Pin Description	Pin #
3	IO_L136P	AR2
3	IO_L136N	AT1
3	IO_L137P_Y	AV4
3	IO_VREF_L137N_Y	AT2
3	IO_L138P_Y	AU1
3	IO_L138N_Y	AU5
3	IO_L139P_Y	AU2
3	IO_L139N_Y	AW3
3	IO_D7_L140P_YY	AV1
3	IO_INIT_L140N_YY	AW5
4	GCK0	BA22
4	IO	AV17
4	IO	AY11
4	IO	AY12
4	IO	AY13
4	IO	AY14
4	IO	BA8
4	IO	BA17
4	IO	BA19
4	IO	BA20
4	IO	BA21
4	IO	BB9
4	IO	BB18
4	IO_L141P_YY	AV6
4	IO_L141N_YY	BA4
4	IO_L142P_Y	AY4
4	IO_L142N_Y	BA5
4	IO_L143P_Y	AW6
4	IO_L143N_Y	BB5
4	IO_VREF_L144P_Y	BA6
4	IO_L144N_Y	AY5
4	IO_L145P_Y	BB6
4	IO_L145N_Y	AY6
4	IO_L146P_YY	BA7
4	IO_L146N_YY	AV7
4	IO_VREF_L147P_YY	BB7

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

Bank	Pin Description	Pin #
6	IO	AC5[4]
6	IO	AD1[4]
6	IO	AE5[5]
6	IO_L212N_YY	AF3
6	IO_L212P_YY	AC6
6	IO_L213N	AH2[4]
6	IO_L213P	AG2[3]
6	IO_L214N	AB9
6	IO_L214P	AE4
6	IO_VREF_L215N_YY	AE3[1]
6	IO_L215P_YY	AH1
6	IO_L216N_Y	AB8[4]
6	IO_L216P_Y	AD6[3]
6	IO_L217N_YY	AG1
6	IO_L217P_YY	AA10
6	IO_VREF_L218N	AA9
6	IO_L218P	AD4
6	IO_L219N_YY	AD5
6	IO_L219P_YY	AD2
6	IO_L220N_YY	AD3
6	IO_L220P_YY	AF2
6	IO_L221N	AA8
6	IO_L221P	AA7
6	IO_VREF_L222N_YY	AF1
6	IO_L222P_YY	Y9
6	IO_L223N_YY	AB6
6	IO_L223P_YY	AC4
6	IO_L224N	AE1
6	IO_L224P	W8
6	IO_L225N_YY	Y8
6	IO_L225P_YY	AB4
6	IO_VREF_L226N_YY	AB3
6	IO_L226P_YY	W9
6	IO_L227N_YY	AA5[4]
6	IO_L227P_YY	W10[3]
6	IO_L228N_YY	AB1
6	IO_L228P_YY	V10

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

Bank	Pin Description	Pin #
6	IO_L229N_YY	Y7[4]
6	IO_VREF_L229P_YY	AC1
6	IO_L230N	V11
6	IO_L230P	AA3
6	IO_L231N_YY	AA2[3]
6	IO_L231P_YY	U10[4]
6	IO_L232N	W7
6	IO_L232P	AA6
6	IO_L233N_YY	Y6
6	IO_L233P_YY	Y4
6	IO_L234N_Y	AA1[4]
6	IO_L234P_Y	V7[4]
6	IO_L235N_YY	Y3
6	IO_L235P_YY	Y2
6	IO_VREF_L236N	Y5[1]
6	IO_L236P	W5
6	IO_L237N_YY	W4
6	IO_L237P_YY	W6
6	IO_L238N_YY	V6
6	IO_L238P_YY	W2
6	IO_L239N	U9
6	IO_L239P	V4
6	IO_VREF_L240N_YY	AB2
6	IO_L240P_YY	T8
6	IO_L241N_YY	U5
6	IO_L241P_YY	W1
6	IO_L242N	Y1
6	IO_L242P	T9
6	IO_L243N_YY	T7
6	IO_L243P_YY	U3
6	IO_VREF_L244N_YY	T5
6	IO_L244P_YY	V2
6	IO_L245N_YY	R9[4]
6	IO_L245P_YY	T6[3]
6	IO_VREF_L246N_YY	T4[2]
6	IO_L246P_YY	U2
6	IO_L247N	T1

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

Bank	Pin Description	Pin #
4	IO_L178N_YY	AL28
4	IO_L179P_YY	AE24[4]
4	IO_L179N_YY	AN28[5]
4	IO_L180P_Y	AJ27
4	IO_L180N_Y	AH26
4	IO_L181P_Y	AG25
4	IO_L181N_Y	AK27
4	IO_L182P	AM28[4]
4	IO_L182N	AF24[5]
4	IO_L183P_YY	AJ26
4	IO_L183N_YY	AP27
4	IO_VREF_L184P_YY	AK26
4	IO_L184N_YY	AN27
4	IO_L185P	AE23[4]
4	IO_L185N	AM27[5]
4	IO_L186P_Y	AL26
4	IO_L186N_Y	AP26
4	IO_VREF_L187P_Y	AN26[2]
4	IO_L187N_Y	AJ25
4	IO_L188P	AG24[4]
4	IO_L188N	AP25[5]
4	IO_L189P_YY	AF23
4	IO_L189N_YY	AM26
4	IO_VREF_L190P_YY	AJ24
4	IO_L190N_YY	AN25
4	IO_L191P_Y	AE22
4	IO_L191N_Y	AM25
4	IO_L192P_Y	AK24
4	IO_L192N_Y	AH23
4	IO_VREF_L193P_YY	AF22
4	IO_L193N_YY	AP24
4	IO_L194P_YY	AL24
4	IO_L194N_YY	AK23
4	IO_L195P_Y	AG22
4	IO_L195N_Y	AN23
4	IO_L196P_Y	AP23
4	IO_L196N_Y	AM23
4	IO_L197P_Y	AH22
4	IO_L197N_Y	AP22
4	IO_L198P_Y	AL23
4	IO_L198N_Y	AF21
4	IO_L199P_YY	AL22
4	IO_L199N_YY	AJ22
4	IO_VREF_L200P_YY	AK22
4	IO_L200N_YY	AM22
4	IO_L201P_YY	AG21[4]
4	IO_L201N_YY	AJ21[5]
4	IO_L202P_Y	AP21
4	IO_L202N_Y	AE20
4	IO_L203P_Y	AH21
4	IO_L203N_Y	AL21
4	IO_L204P	AN21[4]
4	IO_L204N	AF20[5]
4	IO_L205P_YY	AK21
4	IO_L205N_YY	AP20
4	IO_VREF_L206P_YY	AE19
4	IO_L206N_YY	AN20
4	IO_L207P_Y	AG20[4]
4	IO_L207N_Y	AL20[5]
4	IO_L208P_Y	AH20
4	IO_L208N_Y	AK20
4	IO_L209P_Y	AN19
4	IO_L209N_Y	AJ20
4	IO_L210P	AF19[4]
4	IO_L210N	AP19[5]
4	IO_L211P_YY	AM19
4	IO_L211N_YY	AH19
4	IO_VREF_L212P_YY	AJ19

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

Bank	Pin Description	Pin #
NA	VCCO_7	K5
NA	VCCO_7	F1
NA	VCCO_7	T11
NA	VCCO_7	T12
NA	VCCO_7	R11
NA	VCCO_7	R12
NA	VCCO_7	P3
NA	VCCO_7	P11
NA	VCCO_7	P12
NA	VCCO_7	N11
NA	GND	K32
NA	GND	R4
NA	GND	AN1
NA	GND	AM11
NA	GND	AK5
NA	GND	AH28
NA	GND	AD32
NA	GND	AA20
NA	GND	Y20
NA	GND	W19
NA	GND	V19
NA	GND	U20
NA	GND	T20
NA	GND	R19
NA	GND	P19
NA	GND	H8
NA	GND	F12
NA	GND	C2
NA	GND	B1
NA	GND	A7
NA	GND	AP1
NA	GND	AN2
NA	GND	AM15
NA	GND	AK17
NA	GND	AH34
NA	GND	AC6
NA	GND	AA21
NA	GND	Y21
NA	GND	W20
NA	GND	V20
NA	GND	U21
NA	GND	T21
NA	GND	R20
NA	GND	P20
NA	GND	H16
NA	GND	F23
NA	GND	C3
NA	GND	B2
NA	GND	A28
NA	GND	AP34
NA	GND	AM3
NA	GND	AL31
NA	GND	AH7
NA	GND	AD3
NA	GND	AA19
NA	GND	Y19
NA	GND	W18
NA	GND	V18
NA	GND	U19
NA	GND	T19
NA	GND	R18
NA	GND	P18
NA	GND	J26
NA	GND	F6
NA	GND	C1
NA	GND	C34
NA	GND	A3

Date	Version	Revision
4/2/01	2.0	- Updated numerous values in **Virtex-E Switching Characteristics** tables. - Changed pinout table footnotes from "V_{REF} option only" to "V_{REF} or I/O option only" to improve clarity. - Converted file to modularized format. See the **Virtex-E Data Sheet** section.
7/26/01	2.1	- Changed pinout table footnotes from "V_{REF} or I/O option only" to "V_{REF} or I/O option only; otherwise I/O only" to improve clarity. - Changed designation for pin pair 300 in Table 29 from AO to footnote 9.
10/25/01	2.2	- Changed Table 29 to clarify which devices in the FG1156 package can use each pin pair as an asynchronous output. - Updated references to the XCV3200E device in the FG1156 package.
11/15/01	2.3	- Fixed cosmetic error.
07/17/02	2.4	- Added “VREF” to the description for pin B15 in Table 12. - Changed designation for pin pair 129 in Table 15 from AO to “AO in the XCV1000E, 1600E, 2000E“. - Data sheet designation upgraded from Preliminary to Production.
03/14/03	2.5	- Removed the Virtex-E XCV300E section under **Pinout Differences Between Virtex and Virtex-E Families** (and revised Table 1), since these differences do not exist.

Virtex-E Data Sheet

The Virtex-E Data Sheet contains the following modules:

- DS022-1, Virtex-E 1.8V FPGAs:
Introduction and Ordering Information (Module 1)
- DS022-2, Virtex-E 1.8V FPGAs:
Functional Description (Module 2)
- DS022-3, Virtex-E 1.8V FPGAs:
DC and Switching Characteristics (Module 3)
- DS022-4, Virtex-E 1.8V FPGAs:
Pinout Tables (Module 4)