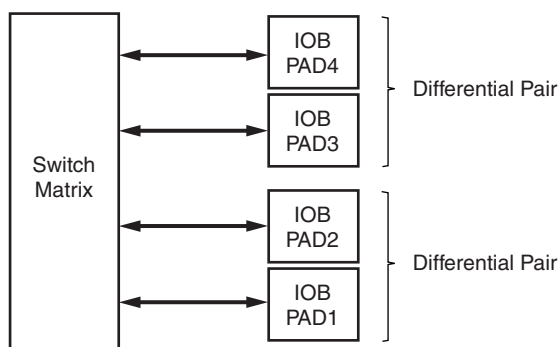


## Detailed Description

### Input/Output Blocks (IOBs)

Virtex-II I/O blocks (IOBs) are provided in groups of two or four on the perimeter of each device. Each IOB can be used as input and/or output for single-ended I/Os. Two IOBs can be used as a differential pair. A differential pair is always connected to the same switch matrix, as shown in **Figure 1**.

IOB blocks are designed for high performances I/Os, supporting 19 single-ended standards, as well as differential signaling with LVDS, LDT, Bus LVDS, and LVPECL.



DS031\_30\_101600

Figure 1: Virtex-II Input/Output Tile

Note: Differential I/Os must use the same clock.

### Supported I/O Standards

Virtex-II IOB blocks feature SelectI/O-Ultra inputs and outputs that support a wide variety of I/O signaling standards. In addition to the internal supply voltage ( $V_{CCINT} = 1.5V$ ), output driver supply voltage ( $V_{CCO}$ ) is dependent on the I/O standard (see **Table 1**). An auxiliary supply voltage ( $V_{CCAUX} = 3.3V$ ) is required, regardless of the I/O standard used. For exact supply voltage absolute maximum ratings, see **DC Input and Output Levels** in [Data Sheet Module 3](#).

Table 1: Supported Single-Ended I/O Standards

I/O Standard	Output $V_{CCO}$	Input $V_{CCO}$	Input $V_{REF}$	Board Termination Voltage ( $V_{TT}$ )
LVTTTL	3.3	3.3	N/R <sup>(3)</sup>	N/R
LVC MOS33	3.3	3.3	N/R	N/R
LVC MOS25	2.5	2.5	N/R	N/R
LVC MOS18	1.8	1.8	N/R	N/R
LVC MOS15	1.5	1.5	N/R	N/R
PCI33_3	3.3	3.3	N/R	N/R
PCI66_3	3.3	3.3	N/R	N/R
PCI-X	3.3	3.3	N/R	N/R
GTL	Note (1)	Note (1)	0.8	1.2
GTLP	Note (1)	Note (1)	1.0	1.5
HSTL_I	1.5	N/R	0.75	0.75
HSTL_II	1.5	N/R	0.75	0.75
HSTL_III	1.5	N/R	0.9	1.5
HSTL_IV	1.5	N/R	0.9	1.5
HSTL_I_18	1.8	N/R	0.9	0.9
HSTL_II_18	1.8	N/R	0.9	0.9
HSTL_III_18	1.8	N/R	1.1	1.8
HSTL_IV_18	1.8	N/R	1.1	1.8
SSTL18_I <sup>(2)</sup>	1.8	N/R	0.9	0.9
SSTL18_II	1.8	N/R	0.9	0.9
SSTL2_I	2.5	N/R	1.25	1.25
SSTL2_II	2.5	N/R	1.25	1.25
SSTL3_I	3.3	N/R	1.5	1.5
SSTL3_II	3.3	N/R	1.5	1.5
AGP-2X/AGP	3.3	N/R	1.32	N/R

#### Notes:

- $V_{CCO}$  of GTL or GTLP should not be lower than the termination voltage or the voltage seen at the I/O pad.
- SSTL18\_I is not a JEDEC-supported standard.
- N/R = no requirement.

Table 2: Supported Differential Signal I/O Standards

I/O Standard	Output V <sub>CCO</sub>	Input V <sub>CCO</sub>	Input V <sub>REF</sub>	Output V <sub>OD</sub>
LVPECL_33	3.3	N/R <sup>(1)</sup>	N/R	490 mV to 1.22V
LDT_25	2.5	N/R	N/R	0.430 - 0.670
LVDS_33	3.3	N/R	N/R	0.250 - 0.400
LVDS_25	2.5	N/R	N/R	0.250 - 0.400
LVDSEXT_33	3.3	N/R	N/R	0.330 - 0.700
LVDSEXT_25	2.5	N/R	N/R	0.330 - 0.700
BLVDS_25	2.5	N/R	N/R	0.250 - 0.450
ULVDS_25	2.5	N/R	N/R	0.430 - 0.670

**Notes:**

1. N/R = no requirement.

All of the user IOBs have fixed-clamp diodes to V<sub>CCO</sub> and to ground. As outputs, these IOBs are not compatible or compliant with 5V I/O standards. As inputs, these IOBs are not normally 5V tolerant, but can be used with 5V I/O standards when external current-limiting resistors are used. For more details, see the “5V Tolerant I/Os” Tech Topic at [www.xilinx.com](http://www.xilinx.com).

Table 3 lists supported I/O standards with Digitally Controlled Impedance. See **Digitally Controlled Impedance (DCI)**, page 8.

Table 3: Supported DCI I/O Standards

I/O Standard	Output V <sub>CCO</sub>	Input V <sub>CCO</sub>	Input V <sub>REF</sub>	Termination Type
LVDCI_33 <sup>(1)</sup>	3.3	3.3	N/R <sup>(4)</sup>	Series
LVDCI_DV2_33 <sup>(1)</sup>	3.3	3.3	N/R	Series
LVDCI_25 <sup>(1)</sup>	2.5	2.5	N/R	Series
LVDCI_DV2_25 <sup>(1)</sup>	2.5	2.5	N/R	Series
LVDCI_18 <sup>(1)</sup>	1.8	1.8	N/R	Series
LVDCI_DV2_18 <sup>(1)</sup>	1.8	1.8	N/R	Series
LVDCI_15 <sup>(1)</sup>	1.5	1.5	N/R	Series
LVDCI_DV2_15 <sup>(1)</sup>	1.5	1.5	N/R	Series
GTL_DCI	1.2	1.2	0.8	Single
GTLP_DCI	1.5	1.5	1.0	Single
HSTL_I_DCI	1.5	1.5	0.75	Split
HSTL_II_DCI	1.5	1.5	0.75	Split
HSTL_III_DCI	1.5	1.5	0.9	Single
HSTL_IV_DCI	1.5	1.5	0.9	Single
HSTL_I_DCI_18	1.8	1.8	0.9	Split
HSTL_II_DCI_18	1.8	1.8	0.9	Split
HSTL_III_DCI_18	1.8	1.8	1.1	Single
HSTL_IV_DCI_18	1.8	1.8	1.1	Single
SSTL18_I_DCI <sup>(3)</sup>	1.8	1.8	0.9	Split
SSTL18_II_DCI	1.8	1.8	0.9	Split
SSTL2_I_DCI <sup>(2)</sup>	2.5	2.5	1.25	Split
SSTL2_II_DCI <sup>(2)</sup>	2.5	2.5	1.25	Split
SSTL3_I_DCI <sup>(2)</sup>	3.3	3.3	1.5	Split
SSTL3_II_DCI <sup>(2)</sup>	3.3	3.3	1.5	Split
LVDS_33_DCI	3.3	3.3	N/R	Split
LVDS_25_DCI	2.5	2.5	N/R	Split
LVDSEXT_33_DCI	3.3	3.3	N/R	Split
LVDSEXT_25_DCI	2.5	2.5	N/R	Split

**Notes:**

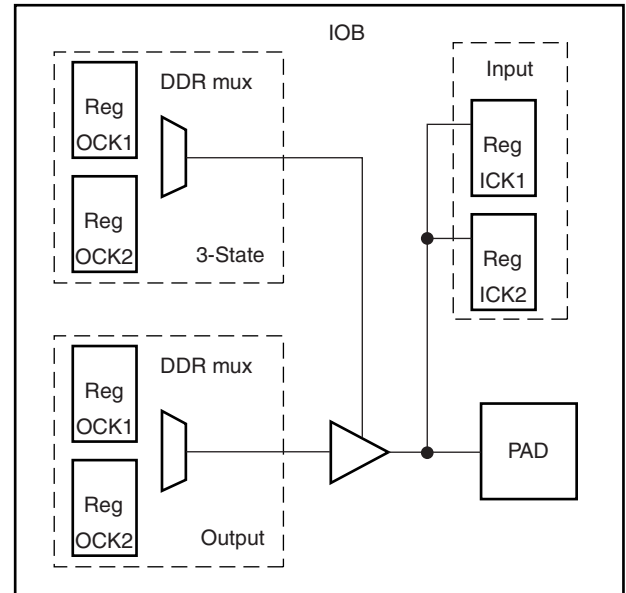
1. LVDCI\_XX and LVDCI\_DV2\_XX are LVCMOS controlled impedance buffers, matching the reference resistors or half of the reference resistors.
2. These are SSTL compatible.
3. SSTL18\_I is not a JEDEC-supported standard.
4. N/R = no requirement.

## Logic Resources

IOB blocks include six storage elements, as shown in **Figure 2**.

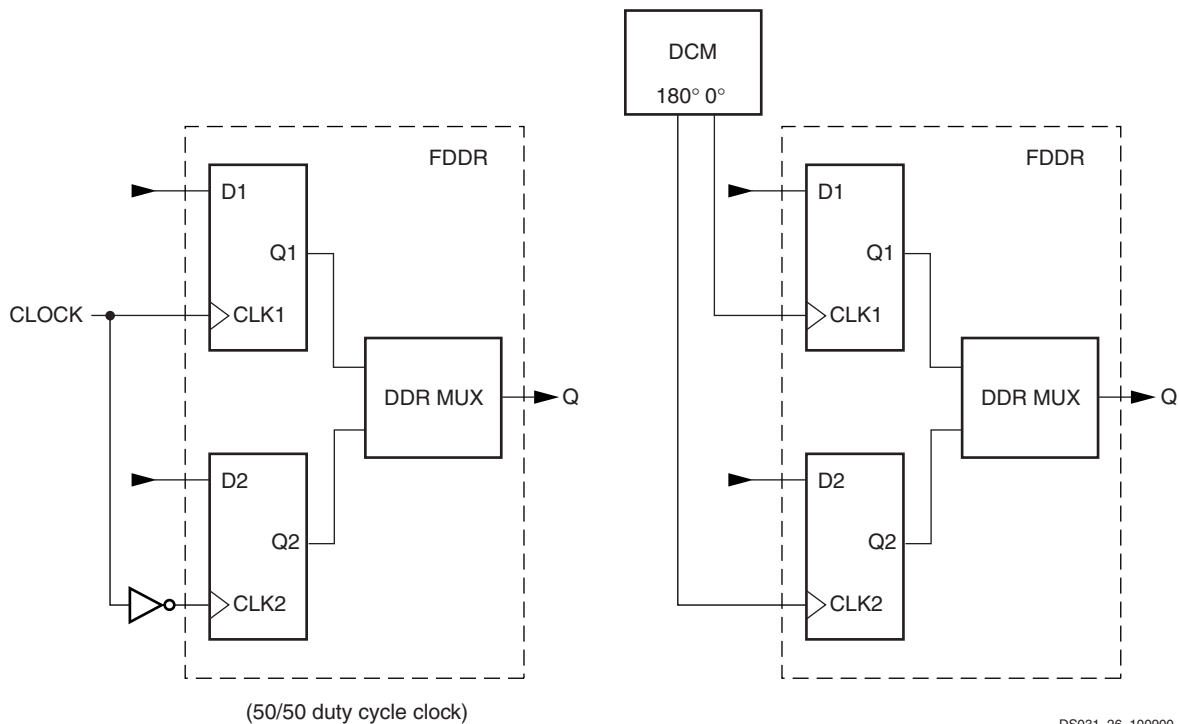
Each storage element can be configured either as an edge-triggered D-type flip-flop or as a level-sensitive latch. On the input, output, and 3-state path, one or two DDR registers can be used.

Double data rate is directly accomplished by the two registers on each path, clocked by the rising edges (or falling edges) from two different clock nets. The two clock signals are generated by the DCM and must be 180 degrees out of phase, as shown in **Figure 3**. There are two input, output, and 3-state data signals, each being alternately clocked out.



DS031\_29\_100900

Figure 2: Virtex-II IOB Block



DS031\_26\_100900

Figure 3: Double Data Rate Registers

For each storage element, the SRHIGH, SRLOW, INIT0, and INIT1 attributes are independent. Synchronous or asynchronous set / reset is consistent in an IOB block.

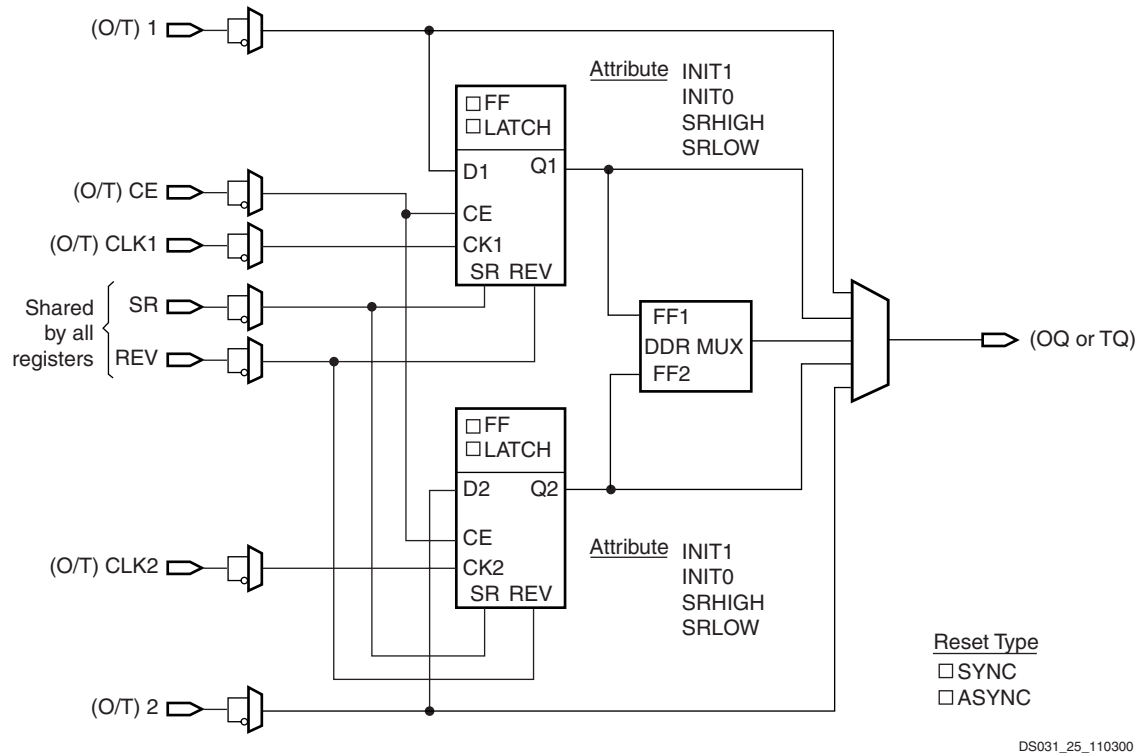
All the control signals have independent polarity. Any inverter placed on a control input is automatically absorbed.

Each register or latch (independent of all other registers or latches) (see **Figure 4**) can be configured as follows:

- No set or reset

- Synchronous set
- Synchronous reset
- Synchronous set and reset
- Asynchronous set (preset)
- Asynchronous reset (clear)
- Asynchronous set and reset (preset and clear)

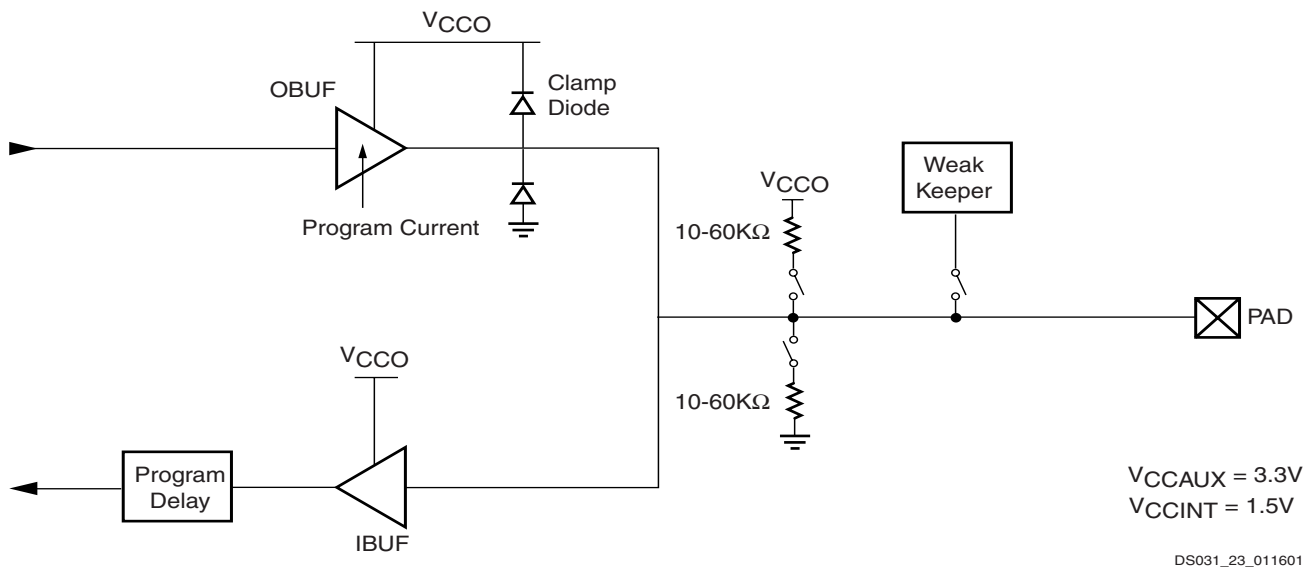
The synchronous reset overrides a set, and an asynchronous clear overrides a preset.



### Input/Output Individual Options

Each device pad has optional pull-up and pull-down in all SelectI/O-Ultra configurations. Each device pad has optional weak-keeper in LVTTTL, LVCMOS, and PCI SelectI/O-Ultra configurations, as illustrated in Figure 5.

Values of the optional pull-up and pull-down resistors are in the range 10 - 60 K $\Omega$ , which is the specification for  $V_{CCO}$  when operating at 3.3V (from 3.0 to 3.6V only). The clamp diode is always present, even when power is not.



The optional weak-keeper circuit is connected to each user I/O pad. When selected, the circuit monitors the voltage on

the pad and weakly drives the pin High or Low. If the pin is connected to a multiple-source signal, the weak-keeper

holds the signal in its last state if all drivers are disabled. Maintaining a valid logic level in this way eliminates bus chatter. An enabled pull-up or pull-down overrides the weak-keeper circuit.

LVTTL sinks and sources current up to 24 mA. The current is programmable for LVTTL and LVCMOS SelectI/O-Ultra standards (see Table 4). Drive-strength and slew-rate controls for each output driver, minimize bus transients. For LVDCI and LVDCI\_DV2 standards, drive strength and slew-rate controls are not available.

Table 4: LVTTL and LVCMOS Programmable Currents (Sink and Source)

SelectI/O-Ultra	Programmable Current (Worst-Case Guaranteed Minimum)						
LVTTL	2 mA	4 mA	6 mA	8 mA	12 mA	16 mA	24 mA
LVCMOS33	2 mA	4 mA	6 mA	8 mA	12 mA	16 mA	24 mA
LVCMOS25	2 mA	4 mA	6 mA	8 mA	12 mA	16 mA	24 mA
LVCMOS18	2 mA	4 mA	6 mA	8 mA	12 mA	16 mA	n/a
LVCMOS15	2 mA	4 mA	6 mA	8 mA	12 mA	16 mA	n/a

Figure 6 shows the SSTL2, SSTL3, and HSTL configurations. HSTL can sink current up to 48 mA. (HSTL IV)

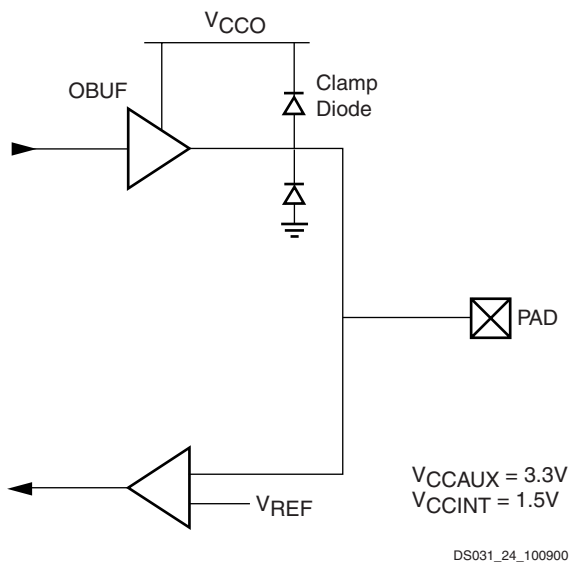


Figure 6: SSTL or HSTL SelectI/O-Ultra Standards

All pads are protected against damage from electrostatic discharge (ESD) and from over-voltage transients. Virtex-II uses two memory cells to control the configuration of an I/O as an input. This is to reduce the probability of an I/O configured as an input from flipping to an output when subjected to a single event upset (SEU) in space applications.

Prior to configuration, all outputs not involved in configuration are forced into their high-impedance state. The pull-down resistors and the weak-keeper circuits are inactive. The dedicated pin HSWAP\_EN controls the pull-up resistors prior to configuration. By default, HSWAP\_EN is set high, which disables the pull-up resistors on user I/O pins. When HSWAP\_EN is set low, the pull-up resistors are activated on user I/O pins.

All Virtex-II IOBs support IEEE 1149.1 compatible boundary scan testing.

### Input Path

The Virtex-II IOB input path routes input signals directly to internal logic and / or through an optional input flip-flop or latch, or through the DDR input registers. An optional delay element at the D-input of the storage element eliminates pad-to-pad hold time. The delay is matched to the internal clock-distribution delay of the Virtex-II device, and when used, assures that the pad-to-pad hold time is zero.

Each input buffer can be configured to conform to any of the low-voltage signaling standards supported. In some of these standards the input buffer utilizes a user-supplied threshold voltage,  $V_{REF}$ . The need to supply  $V_{REF}$  imposes constraints on which standards can be used in the same bank. See I/O banking description.

### Output Path

The output path includes a 3-state output buffer that drives the output signal onto the pad. The output and / or the 3-state signal can be routed to the buffer directly from the internal logic or through an output / 3-state flip-flop or latch, or through the DDR output / 3-state registers.

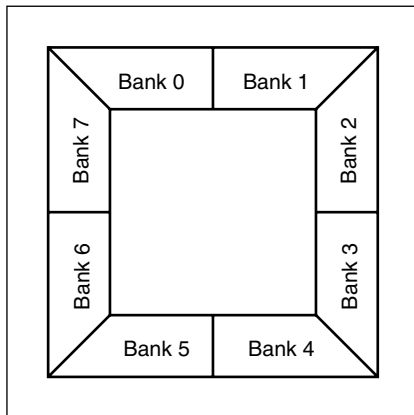
Each output driver can be individually programmed for a wide range of low-voltage signaling standards. In most signaling standards, the output High voltage depends on an externally supplied  $V_{CCO}$  voltage. The need to supply  $V_{CCO}$  imposes constraints on which standards can be used in the same bank. See I/O banking description.

### I/O Banking

Some of the I/O standards described above require  $V_{CCO}$  and  $V_{REF}$  voltages. These voltages are externally supplied and connected to device pins that serve groups of IOB blocks, called banks. Consequently, restrictions exist about which I/O standards can be combined within a given bank.

Eight I/O banks result from dividing each edge of the FPGA into two banks, as shown in Figure 7 and Figure 8. Each bank has multiple  $V_{CCO}$  pins, all of which must be con-

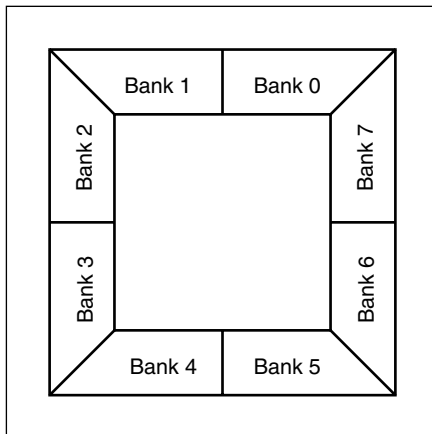
nected to the same voltage. This voltage is determined by the output standards in use.



ug002\_c2\_014\_112900

Figure 7: Virtex-II I/O Banks: Top View for Wire-Bond Packages (CS, FG, & BG)

Some input standards require a user-supplied threshold voltage ( $V_{REF}$ ), and certain user-I/O pins are automatically configured as  $V_{REF}$  inputs. Approximately one in six of the I/O pins in the bank assume this role.



ds031\_66\_112900

Figure 8: Virtex-II I/O Banks: Top View for Flip-Chip Packages (FF & BF)

$V_{REF}$  pins within a bank are interconnected internally, and consequently only one  $V_{REF}$  voltage can be used within each bank. However, for correct operation, all  $V_{REF}$  pins in the bank must be connected to the external reference voltage source.

The  $V_{CCO}$  and the  $V_{REF}$  pins for each bank appear in the device pinout tables. Within a given package, the number of  $V_{REF}$  and  $V_{CCO}$  pins can vary depending on the size of device. In larger devices, more I/O pins convert to  $V_{REF}$  pins. Since these are always a superset of the  $V_{REF}$  pins used for smaller devices, it is possible to design a PCB that permits migration to a larger device if necessary.

All  $V_{REF}$  pins for the largest device anticipated must be connected to the  $V_{REF}$  voltage and not used for I/O. In smaller devices, some  $V_{CCO}$  pins used in larger devices do not connect within the package. These unconnected pins can be left unconnected externally, or, if necessary, they can be connected to  $V_{CCO}$  to permit migration to a larger device.

### Rules for Combining I/O Standards in the Same Bank

The following rules must be obeyed to combine different input, output, and bi-directional standards in the same bank:

- Combining output standards only.** Output standards with the same output  $V_{CCO}$  requirement can be combined in the same bank.  
Compatible example:  
SSTL2\_I and LVDS\_25\_DCI outputs  
Incompatible example:  
SSTL2\_I (output  $V_{CCO} = 2.5V$ ) and  
LVCMOS33 (output  $V_{CCO} = 3.3V$ ) outputs
- Combining input standards only.** Input standards with the same input  $V_{CCO}$  and input  $V_{REF}$  requirements can be combined in the same bank.  
Compatible example:  
LVCMOS15 and HSTL\_IV inputs  
Incompatible example:  
LVCMOS15 (input  $V_{CCO} = 1.5V$ ) and  
LVCMOS18 (input  $V_{CCO} = 1.8V$ ) inputs  
Incompatible example:  
HSTL\_I\_DCI\_18 ( $V_{REF} = 0.9V$ ) and  
HSTL\_IV\_DCI\_18 ( $V_{REF} = 1.1V$ ) inputs
- Combining input standards and output standards.** Input standards and output standards with the same input  $V_{CCO}$  and output  $V_{CCO}$  requirement can be combined in the same bank.  
Compatible example:  
LVDS\_25 output and HSTL\_I input  
Incompatible example:  
LVDS\_25 output (output  $V_{CCO} = 2.5V$ ) and  
HSTL\_I\_DCI\_18 input (input  $V_{CCO} = 1.8V$ )
- Combining bi-directional standards with input or output standards.** When combining bi-directional I/O with other standards, make sure the bi-directional standard can meet rules 1 through 3 above.
- Additional rules for combining DCI I/O standards.**
  - No more than one Single Termination type (input or output) is allowed in the same bank.  
Incompatible example:  
HSTL\_IV\_DCI input and HSTL\_III\_DCI input
  - No more than one Split Termination type (input or output) is allowed in the same bank.  
Incompatible example:  
HSTL\_I\_DCI input and HSTL\_II\_DCI input

The implementation tools will enforce these design rules.

Table 5 summarizes all standards and voltage supplies.

**Table 5: Summary of Voltage Supply Requirements for All Input and Output Standards**

I/O Standard	V <sub>CCO</sub>		V <sub>REF</sub>	Termination Type	
	Output	Input	Input	Output	Input
LVDS_33	3.3	N/R	N/R <sup>(1)</sup>	N/R	N/R
LVDSEXT_33			N/R	N/R	N/R
LVPECL_33			N/R	N/R	N/R
SSTL3_I			1.5	N/R	N/R
SSTL3_II			1.5	N/R	N/R
AGP			1.32	N/R	N/R
LVTTTL		3.3	N/R	N/R	N/R
LVC MOS33			N/R	N/R	N/R
LVDCI_33			N/R	Series	N/R
LVDCI_DV2_33			N/R	Series	N/R
PCI33_3			N/R	N/R	N/R
PCI66_3			N/R	N/R	N/R
PCIX			N/R	N/R	N/R
LVDS_33_DCI			N/R	N/R	Split
LVDSEXT_33_DCI			N/R	N/R	Split
SSTL3_I_DCI			1.5	N/R	Split
SSTL3_II_DCI			1.5	Split	Split
LVDS_25	2.5	N/R	N/R	N/R	N/R
LVDSEXT_25			N/R	N/R	N/R
LDT_25			N/R	N/R	N/R
ULVDS_25			N/R	N/R	N/R
BLVDS_25			N/R	N/R	N/R
SSTL2_I			1.25	N/R	N/R
SSTL2_II			1.25	N/R	N/R
LVC MOS25		2.5	N/R	N/R	N/R
LVDCI_25			N/R	Series	N/R
LVDCI_DV2_25			N/R	Series	N/R
LVDS_25_DCI			N/R	N/R	Split
LVDSEXT_25_DCI			N/R	N/R	Split
SSTL2_I_DCI			1.25	N/R	Split
SSTL2_II_DCI			1.25	Split	Split

**Table 5: Summary of Voltage Supply Requirements for All Input and Output Standards (Continued)**

I/O Standard	V <sub>CCO</sub>		V <sub>REF</sub>	Termination Type	
	Output	Input	Input	Output	Input
HSTL_III_18	1.8	N/R	1.1	N/R	N/R
HSTL_IV_18			1.1	N/R	N/R
HSTL_I_18			0.9	N/R	N/R
HSTL_II_18			0.9	N/R	N/R
SSTL18_I			0.9	N/R	N/R
SSTL18_II			0.9	N/R	N/R
LVC MOS18		1.8	N/R	N/R	N/R
LVDCI_18			N/R	Series	N/R
LVDCI_DV2_18			N/R	Series	N/R
HSTL_III_DCI_18			1.1	N/R	Single
HSTL_IV_DCI_18			1.1	Single	Single
HSTL_I_DCI_18			0.9	N/R	Split
HSTL_II_DCI_18			0.9	Split	Split
SSTL18_I_DCI			0.9	N/R	Split
SSTL18_II_DCI			0.9	Split	Split
HSTL_III	1.5	N/R	0.9	N/R	N/R
HSTL_IV			0.9	N/R	N/R
HSTL_I			0.75	N/R	N/R
HSTL_II			0.75	N/R	N/R
LVC MOS15		1.5	N/R	N/R	N/R
LVDCI_15			N/R	Series	N/R
LVDCI_DV2_15			N/R	Series	N/R
GTLP_DCI			1	Single	Single
HSTL_III_DCI			0.9	N/R	Single
HSTL_IV_DCI			0.9	Single	Single
HSTL_I_DCI			0.75	N/R	Split
HSTL_II_DCI			0.75	Split	Split
GTLP_DCI	1.2	1.2	0.8	Single	Single
GTLP	N/R	N/R	1	N/R	N/R
GTLP			0.8	N/R	N/R

**Notes:**

1. N/R = no requirement.

## Digitally Controlled Impedance (DCI)

Today's chip output signals with fast edge rates require termination to prevent reflections and maintain signal integrity. High pin count packages (especially ball grid arrays) can not accommodate external termination resistors.

Virtex-II XCITE DCI provides controlled impedance drivers and on-chip termination for single-ended and differential I/Os. This eliminates the need for external resistors, and improves signal integrity. The DCI feature can be used on any IOB by selecting one of the DCI I/O standards.

When applied to inputs, DCI provides input parallel termination. When applied to outputs, DCI provides controlled impedance drivers (series termination) or output parallel termination.

DCI operates independently on each I/O bank. When a DCI I/O standard is used in a particular I/O bank, external reference resistors must be connected to two dual-function pins on the bank. These resistors, voltage reference of N transistor (VRN) and the voltage reference of P transistor (VRP) are shown in **Figure 9**.

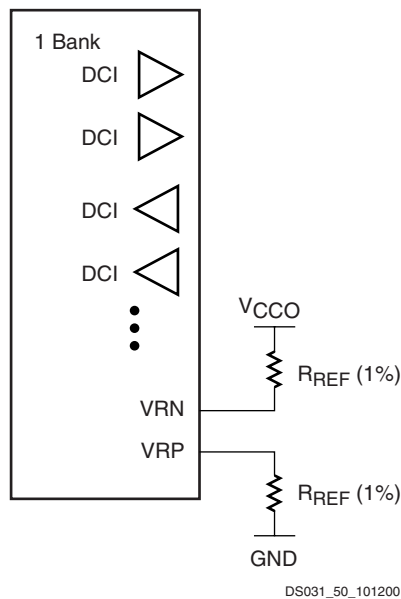


Figure 9: DCI in a Virtex-II Bank

When used with a terminated I/O standard, the value of resistors are specified by the standard (typically 50 Ω). When used with a controlled impedance driver, the resistors set the output impedance of the driver within the specified range (25 Ω to 100 Ω). For all series and parallel terminations listed in **Table 6** and **Table 7**, the reference resistors must have the same value for any given bank. One percent resistors are recommended.

The DCI system adjusts the I/O impedance to match the two external reference resistors, or half of the reference resistors, and compensates for impedance changes due to voltage and/or temperature fluctuations. The adjustment is done by turning parallel transistors in the IOB on or off.

## Controlled Impedance Drivers (Series Termination)

DCI can be used to provide a buffer with a controlled output impedance. It is desirable for this output impedance to match the transmission line impedance (Z). Virtex-II input buffers also support LVDCI and LVDCI\_DV2 I/O standards.

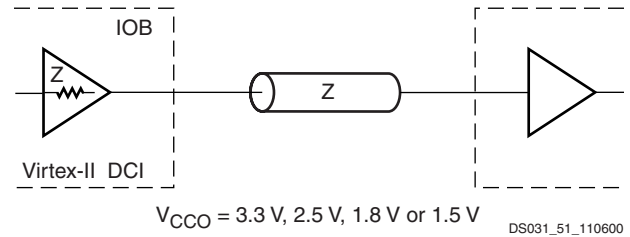


Figure 10: Internal Series Termination

Table 6: SelectI/O-Ultra Controlled Impedance Buffers

V <sub>CCO</sub>	DCI	DCI Half Impedance
3.3 V	LVDCI_33	LVDCI_DV2_33
2.5 V	LVDCI_25	LVDCI_DV2_25
1.8 V	LVDCI_18	LVDCI_DV2_18
1.5 V	LVDCI_15	LVDCI_DV2_15

## Controlled Impedance Drivers (Parallel Termination)

DCI also provides on-chip termination for SSTL3, SSTL2, HSTL (Class I, II, III, or IV), and GTL/GTLP receivers or transmitters on bidirectional lines.

**Table 7** lists the on-chip parallel terminations available in Virtex-II devices. V<sub>CCO</sub> must be set according to **Table 3**. Note that there is a V<sub>CCO</sub> requirement for GTL\_DCI and GTLP\_DCI, due to the on-chip termination resistor.

Table 7: SelectI/O-Ultra Buffers With On-Chip Parallel Termination

I/O Standard	External Termination	On-Chip Termination
SSTL3 Class I	SSTL3_I	SSTL3_I_DCI <sup>(1)</sup>
SSTL3 Class II	SSTL3_II	SSTL3_II_DCI <sup>(1)</sup>
SSTL2 Class I	SSTL2_I	SSTL2_I_DCI <sup>(1)</sup>
SSTL2 Class II	SSTL2_II	SSTL2_II_DCI <sup>(1)</sup>
HSTL Class I	HSTL_I	HSTL_I_DCI
HSTL Class II	HSTL_II	HSTL_II_DCI
HSTL Class III	HSTL_III	HSTL_III_DCI
HSTL Class IV	HSTL_IV	HSTL_IV_DCI
GTL	GTL	GTL_DCI
GTLP	GTLP	GTLP_DCI

### Notes:

1. SSTL Compatible

Figure 11 provides examples illustrating the use of the HSTL\_I\_DCI, HSTL\_II\_DCI, HSTL\_III\_DCI, and HSTL\_IV\_DCI I/O standards. For a complete list, see the *Virtex-II User Guide*.

	HSTL_I	HSTL_II	HSTL_III	HSTL_IV
Conventional				
DCI Transmit Conventional Receive				
Conventional Transmit DCI Receive				
DCI Transmit DCI Receive				
Bidirectional	N/A		N/A	
Reference Resistor	$VRN = VRP = R = Z_0$	$VRN = VRP = R = Z_0$	$VRN = VRP = R = Z_0$	$VRN = VRP = R = Z_0$
Recommended $Z_0^{(1)}$	50 $\Omega$	50 $\Omega$	50 $\Omega$	50 $\Omega$

Note:

1.  $Z_0$  is the recommended PCB trace impedance.

DS031\_65a\_100201

Figure 11: HSTL DCI Usage Examples

Figure 12 provides examples illustrating the use of the SSTL2\_I\_DCI, SSTL2\_II\_DCI, SSTL3\_I\_DCI, and SSTL3\_II\_DCI I/O standards. For a complete list, see the Virtex-II User Guide.

	SSTL2_I	SSTL2_II	SSTL3_I	SSTL3_II
Conventional				
DCI Transmit Conventional Receive				
Conventional Transmit DCI Receive				
DCI Transmit DCI Receive				
Bidirectional	N/A		N/A	
Reference Resistor	$VRN = VRP = R = Z_0$	$VRN = VRP = R = Z_0$	$VRN = VRP = R = Z_0$	$VRN = VRP = R = Z_0$
Recommended $Z_0^{(2)}$	50 Ω	50 Ω	50 Ω	50 Ω

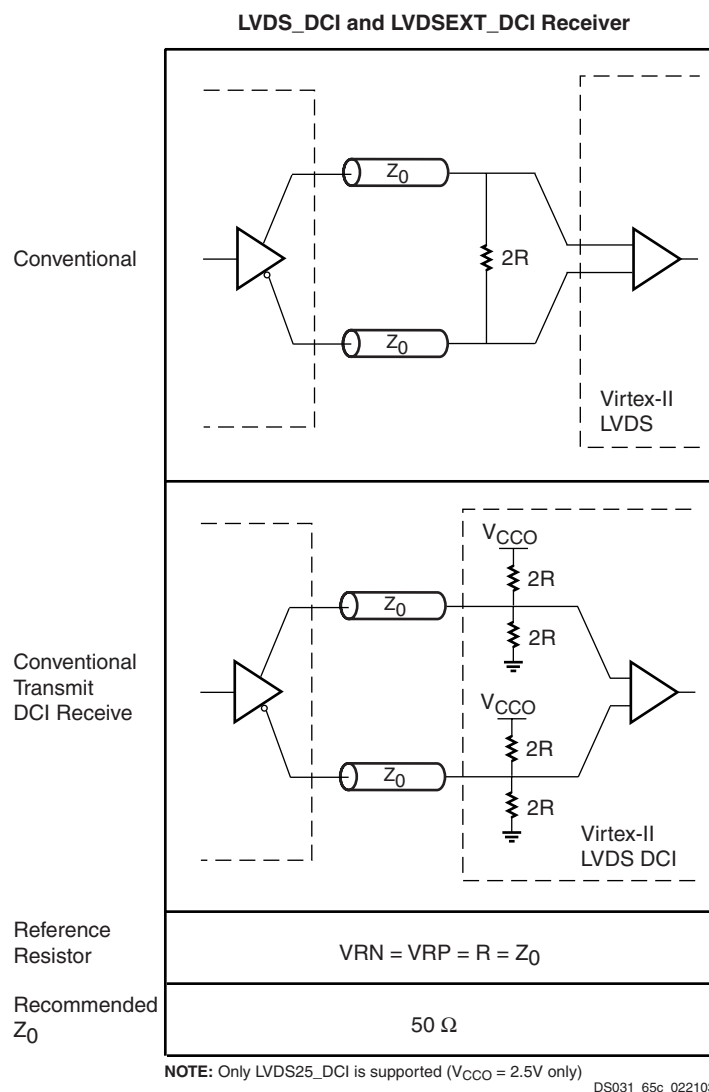
Notes:

1. The SSTL-compatible 25Ω series resistor is accounted for in the DCI buffer, and it is not DCI controlled.
2.  $Z_0$  is the recommended PCB trace impedance.

DS031\_65b\_112502

Figure 12: SSTL DCI Usage Examples

Figure 13 provides examples illustrating the use of the LVDS\_DCI and LVDSEXT\_DCI I/O standards. For a complete list, see the *Virtex-II User Guide*.



NOTE: Only LVDS25\_DCI is supported ( $V_{CCO} = 2.5V$  only)

Figure 13: LVDS DCI Usage Examples

## Configurable Logic Blocks (CLBs)

The Virtex-II configurable logic blocks (CLB) are organized in an array and are used to build combinatorial and synchronous logic designs. Each CLB element is tied to a switch matrix to access the general routing matrix, as shown in **Figure 14**. A CLB element comprises 4 similar slices, with fast local feedback within the CLB. The four slices are split in two columns of two slices with two independent carry logic chains and one common shift chain.

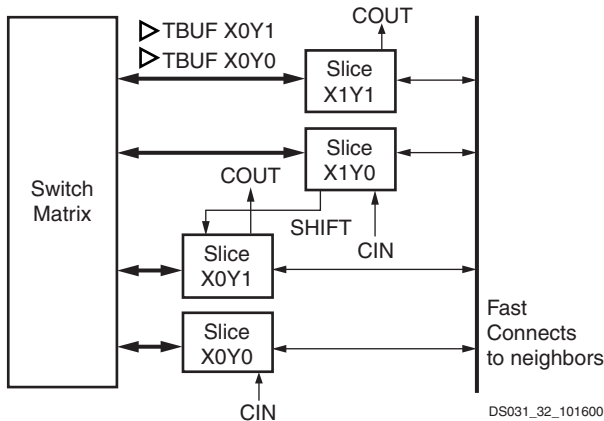


Figure 14: Virtex-II CLB Element

## Slice Description

Each slice includes two 4-input function generators, carry logic, arithmetic logic gates, wide function multiplexers and two storage elements. As shown in **Figure 15**, each 4-input function generator is programmable as a 4-input LUT, 16 bits of distributed SelectRAM memory, or a 16-bit variable-tap shift register element.

The output from the function generator in each slice drives both the slice output and the D input of the storage element. **Figure 16** shows a more detailed view of a single slice.

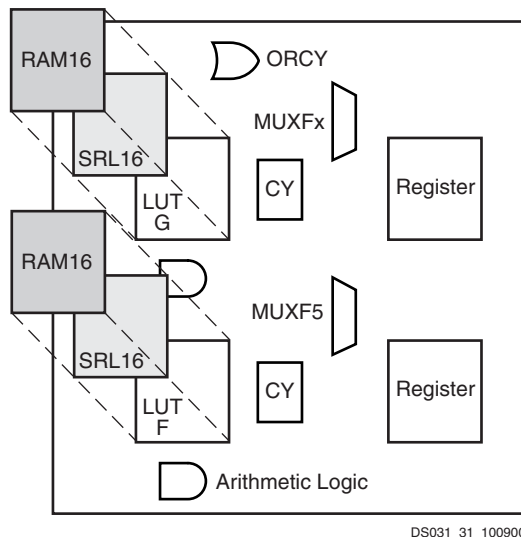


Figure 15: Virtex-II Slice Configuration

## Configurations

### Look-Up Table

Virtex-II function generators are implemented as 4-input look-up tables (LUTs). Four independent inputs are provided to each of the two function generators in a slice (F and G). These function generators are each capable of implementing any arbitrarily defined boolean function of four inputs. The propagation delay is therefore independent of the function implemented. Signals from the function generators can exit the slice (X or Y output), can input the XOR dedicated gate (see arithmetic logic), or input the carry-logic multiplexer (see fast look-ahead carry logic), or feed the D input of the storage element, or go to the MUXF5 (not shown in **Figure 16**).

In addition to the basic LUTs, the Virtex-II slice contains logic (MUXF5 and MUXFX multiplexers) that combines function generators to provide any function of five, six, seven, or eight inputs. The MUXFX are either MUXF6, MUXF7 or MUXF8 according to the slice considered in the CLB. Selected functions up to nine inputs (MUXF5 multiplexer) can be implemented in one slice. The MUXFX can also be a MUXF6, MUXF7, or MUXF8 multiplexers to map any functions of six, seven, or eight inputs and selected wide logic functions.

### Register/Latch

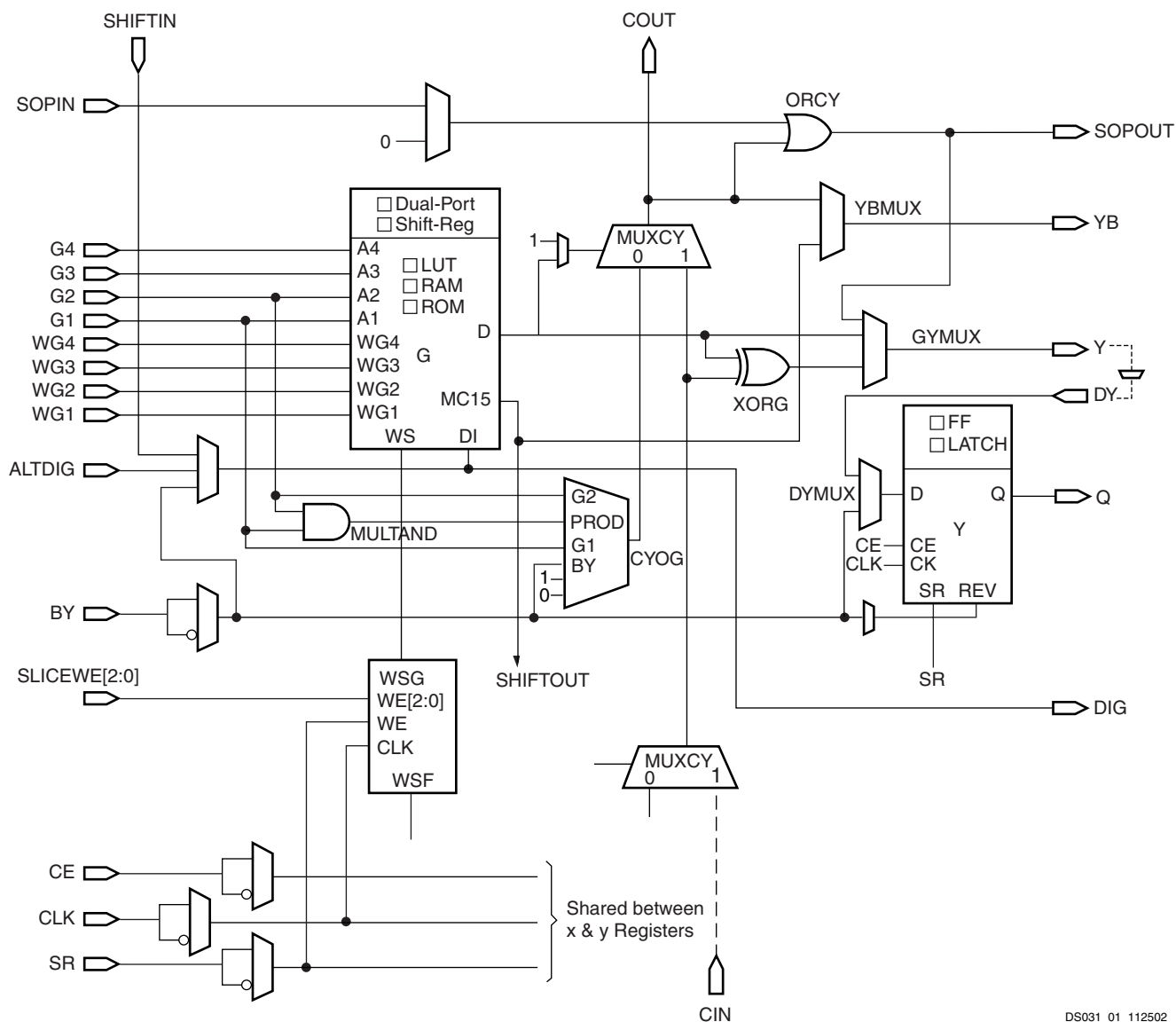
The storage elements in a Virtex-II slice can be configured either as edge-triggered D-type flip-flops or as level-sensitive latches. The D input can be directly driven by the X or Y output via the DX or DY input, or by the slice inputs bypassing the function generators via the BX or BY input. The clock enable signal (CE) is active High by default. If left unconnected, the clock enable for that storage element defaults to the active state.

In addition to clock (CK) and clock enable (CE) signals, each slice has set and reset signals (SR and BY slice inputs). SR forces the storage element into the state specified by the attribute SRHIGH or SRLOW. SRHIGH forces a logic "1" when SR is asserted. SRLOW forces a logic "0". When SR is used, a second input (BY) forces the storage element into the opposite state. The reset condition is predominant over the set condition. (See **Figure 17**.)

The initial state after configuration or global initial state is defined by a separate INIT0 and INIT1 attribute. By default, setting the SRLOW attribute sets INIT0, and setting the SRHIGH attribute sets INIT1.

For each slice, set and reset can be set to be synchronous or asynchronous. Virtex-II devices also have the ability to set INIT0 and INIT1 independent of SRHIGH and SRLOW.

The control signals clock (CLK), clock enable (CE) and set/reset (SR) are common to both storage elements in one slice. All of the control signals have independent polarity. Any inverter placed on a control input is automatically absorbed.



DS031\_01\_112502

Figure 16: Virtex-II Slice (Top Half)

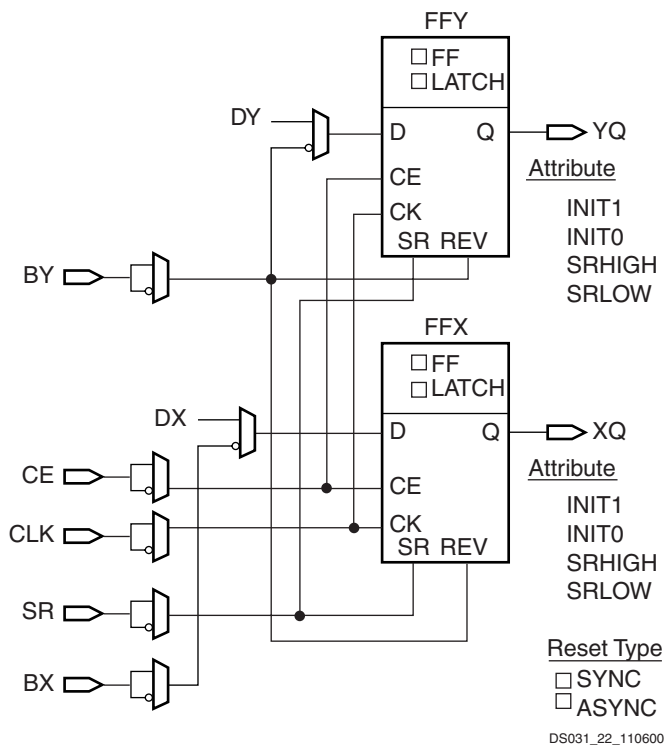


Figure 17: Register / Latch Configuration in a Slice

The set and reset functionality of a register or a latch can be configured as follows:

- No set or reset
- Synchronous set
- Synchronous reset
- Synchronous set and reset
- Asynchronous set (preset)
- Asynchronous reset (clear)
- Asynchronous set and reset (preset and clear)

The synchronous reset has precedence over a set, and an asynchronous clear has precedence over a preset.

### Distributed SelectRAM Memory

Each function generator (LUT) can implement a 16 x 1-bit synchronous RAM resource called a distributed SelectRAM element. The SelectRAM elements are configurable within a CLB to implement the following:

- Single-Port 16 x 8 bit RAM
- Single-Port 32 x 4 bit RAM
- Single-Port 64 x 2 bit RAM
- Single-Port 128 x 1 bit RAM
- Dual-Port 16 x 4 bit RAM
- Dual-Port 32 x 2 bit RAM
- Dual-Port 64 x 1 bit RAM

Distributed SelectRAM memory modules are synchronous (write) resources. The combinatorial read access time is extremely fast, while the synchronous write simplifies high-speed designs. A synchronous read can be implemented with a storage element in the same slice. The distributed SelectRAM memory and the storage element share the same clock input. A Write Enable (WE) input is active High, and is driven by the SR input.

Table 8 shows the number of LUTs (2 per slice) occupied by each distributed SelectRAM configuration.

Table 8: Distributed SelectRAM Configurations

RAM	Number of LUTs
16 x 1S	1
16 x 1D	2
32 x 1S	2
32 x 1D	4
64 x 1S	4
64 x 1D	8
128 x 1S	8

#### Notes:

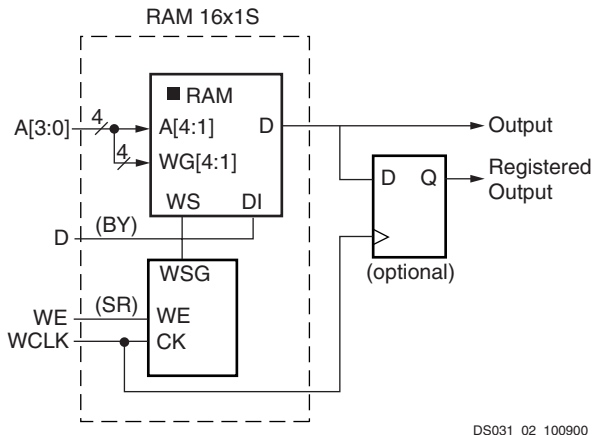
1. S = single-port configuration; D = dual-port configuration

For single-port configurations, distributed SelectRAM memory has one address port for synchronous writes and asynchronous reads.

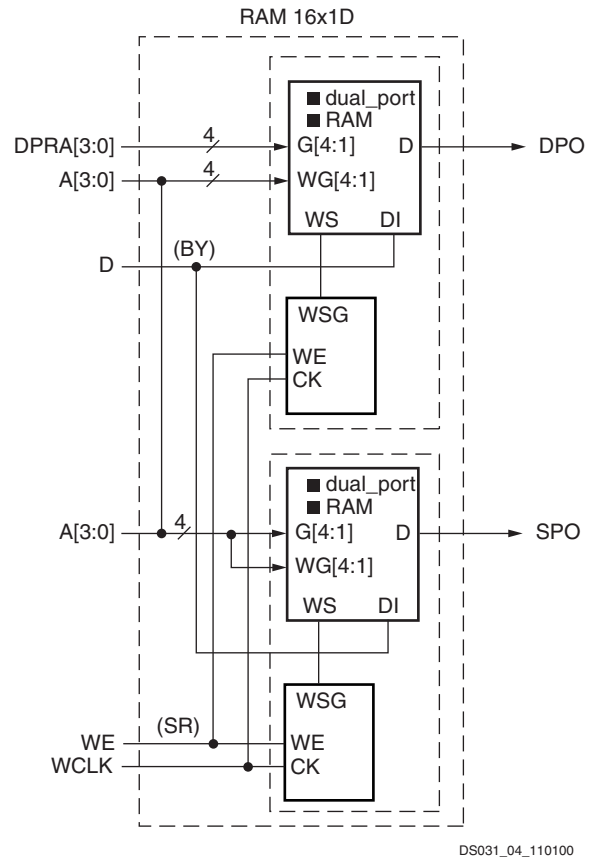
For dual-port configurations, distributed SelectRAM memory has one port for synchronous writes and asynchronous reads and another port for asynchronous reads. The function generator (LUT) has separated read address inputs (A1, A2, A3, A4) and write address inputs (WG1/WF1, WG2/WF2, WG3/WF3, WG4/WF4).

In single-port mode, read and write addresses share the same address bus. In dual-port mode, one function generator (R/W port) is connected with shared read and write addresses. The second function generator has the A inputs (read) connected to the second read-only port address and the W inputs (write) shared with the first read/write port address.

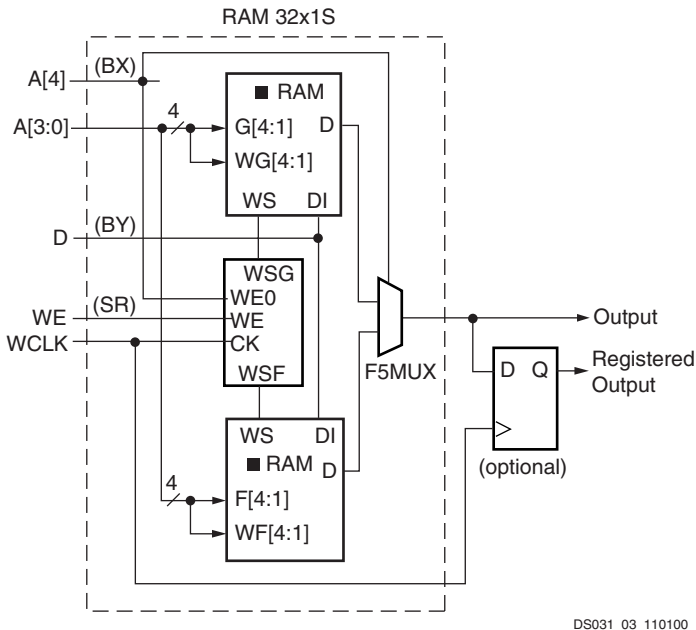
Figure 18, Figure 19, and Figure 20 illustrate various example configurations.



**Figure 18: Distributed SelectRAM (RAM16x1S)**



**Figure 20: Dual-Port Distributed SelectRAM (RAM16x1D)**



**Figure 19: Single-Port Distributed SelectRAM (RAM32x1S)**

Similar to the RAM configuration, each function generator (LUT) can implement a 16 x 1-bit ROM. Five configurations are available: ROM16x1, ROM32x1, ROM64x1, ROM128x1, and ROM256x1. The ROM elements are cascadable to implement wider or/and deeper ROM. ROM contents are loaded at configuration. Table 9 shows the number of LUTs occupied by each configuration.

Table 9: ROM Configuration

ROM	Number of LUTs
16 x 1	1
32 x 1	2
64 x 1	4
128 x 1	8 (1 CLB)
256 x 1	16 (2 CLBs)

## Shift Registers

Each function generator can also be configured as a 16-bit shift register. The write operation is synchronous with a clock input (CLK) and an optional clock enable, as shown in **Figure 21**. A dynamic read access is performed through the 4-bit address bus, A[3:0]. The configurable 16-bit shift register cannot be set or reset. The read is asynchronous, however the storage element or flip-flop is available to implement a synchronous read. The storage element should always be used with a constant address. For example, when building an 8-bit shift register and configuring the addresses to point to the 7th bit, the 8th bit can be the flip-flop. The overall system performance is improved by using the superior clock-to-out of the flip-flops.

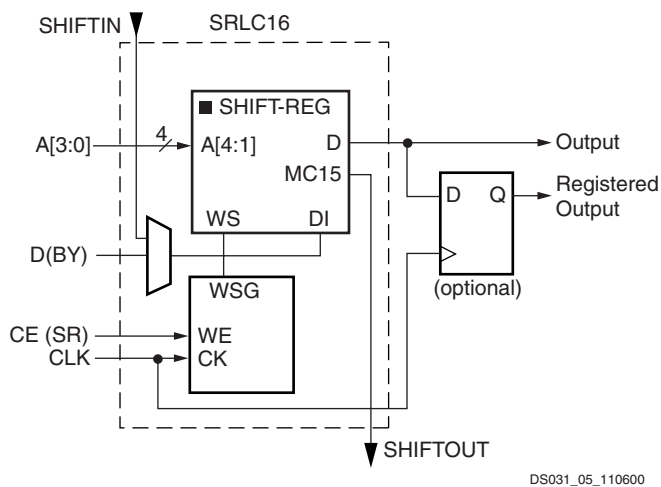


Figure 21: Shift Register Configurations

An additional dedicated connection between shift registers allows connecting the last bit of one shift register to the first bit of the next, without using the ordinary LUT output. (See **Figure 22**.) Longer shift registers can be built with dynamic access to any bit in the chain. The shift register chaining and the MUXF5, MUXF6, and MUXF7 multiplexers allow up to a 128-bit shift register with addressable access to be implemented in one CLB.

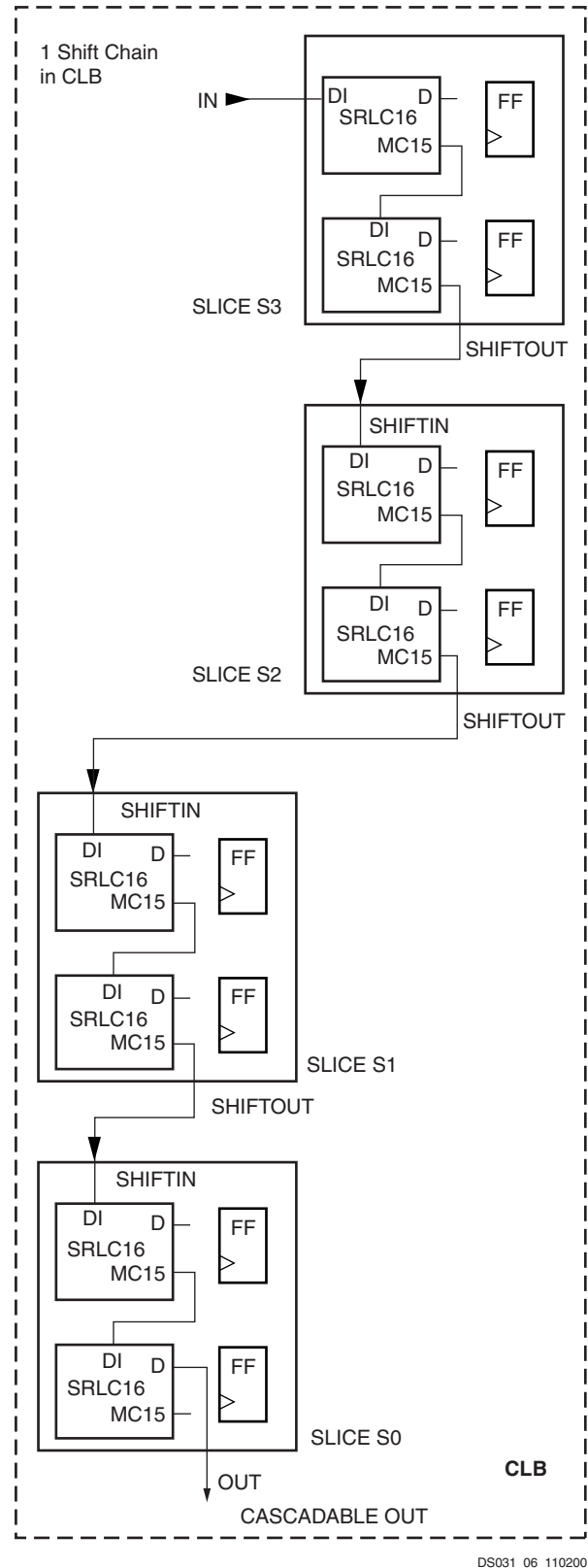


Figure 22: Cascadable Shift Register

## Multiplexers

Virtex-II function generators and associated multiplexers can implement the following:

- 4:1 multiplexer in one slice
- 8:1 multiplexer in two slices
- 16:1 multiplexer in one CLB element (4 slices)
- 32:1 multiplexer in two CLB elements (8 slices)

Each Virtex-II slice has one MUXF5 multiplexer and one MUXFX multiplexer. The MUXFX multiplexer implements the MUXF6, MUXF7, or MUXF8, as shown in **Figure 23**. Each CLB element has two MUXF6 multiplexers, one MUXF7 multiplexer and one MUXF8 multiplexer. Examples of multiplexers are shown in the *Virtex-II User Guide*. Any LUT can implement a 2:1 multiplexer.

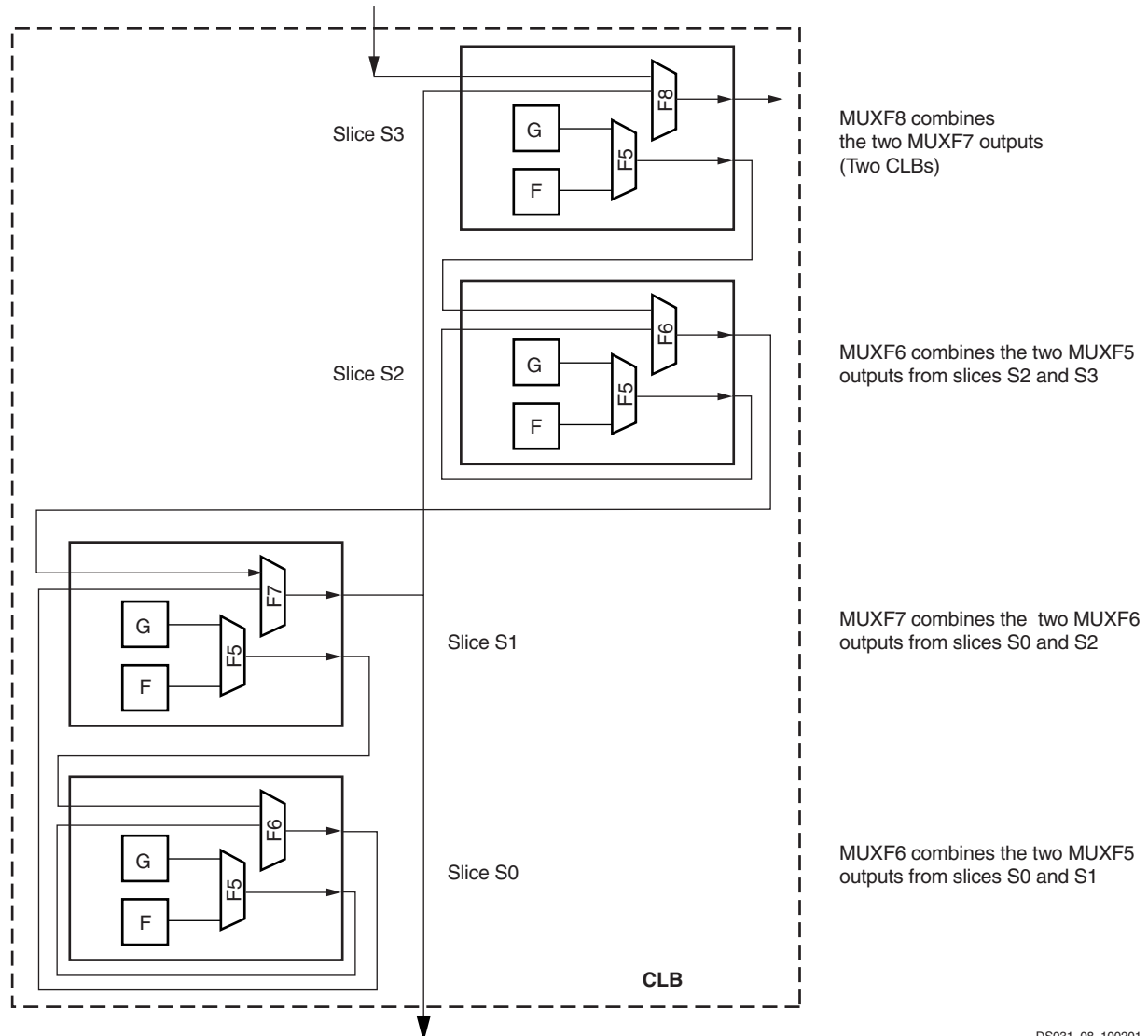


Figure 23: MUXF5 and MUXFX multiplexers

## Fast Lookahead Carry Logic

Dedicated carry logic provides fast arithmetic addition and subtraction. The Virtex-II CLB has two separate carry chains, as shown in the **Figure 24**.

The height of the carry chains is two bits per slice. The carry chain in the Virtex-II device is running upward. The dedicated carry path and carry multiplexer (MUXCY) can also

be used to cascade function generators for implementing wide logic functions.

## Arithmetic Logic

The arithmetic logic includes an XOR gate that allows a 2-bit full adder to be implemented within a slice. In addition, a dedicated AND (MULT\_AND) gate (shown in **Figure 16**) improves the efficiency of multiplier implementation.

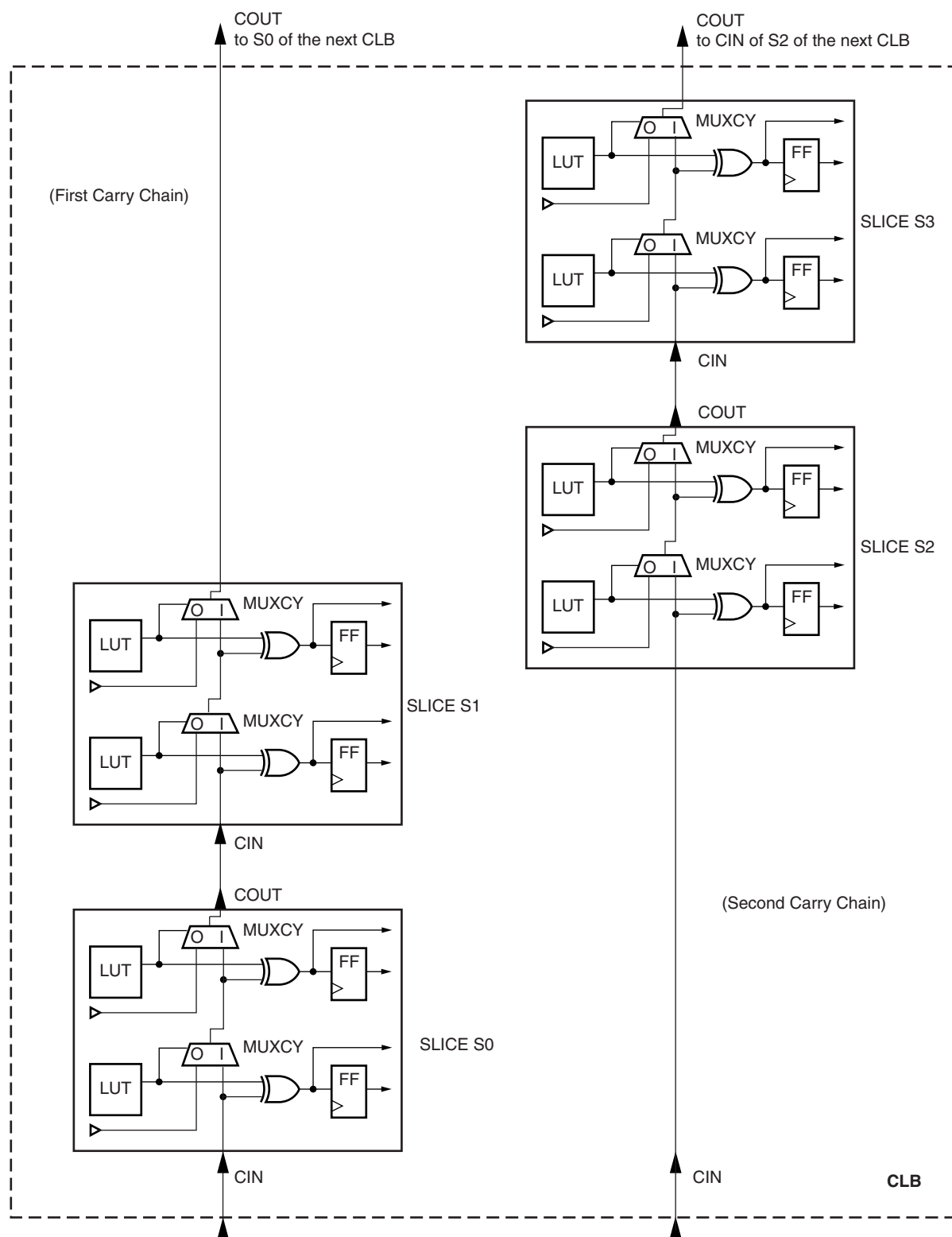


Figure 24: Fast Carry Logic Path

## Sum of Products

Each Virtex-II slice has a dedicated OR gate named ORCY, ORing together outputs from the slices carryout and the ORCY from an adjacent slice. The ORCY gate with the dedicated Sum of Products (SOP) chain are designed for implementing large, flexible SOP chains. One input of each ORCY is connected through the fast SOP chain to the output of the previous ORCY in the same slice row. The second input is connected to the output of the top MUXCY in the same slice, as shown in Figure 25.

LUTs and MUXCYs can implement large AND gates or other combinatorial logic functions. Figure 26 illustrates LUT and MUXCY resources configured as a 16-input AND gate.

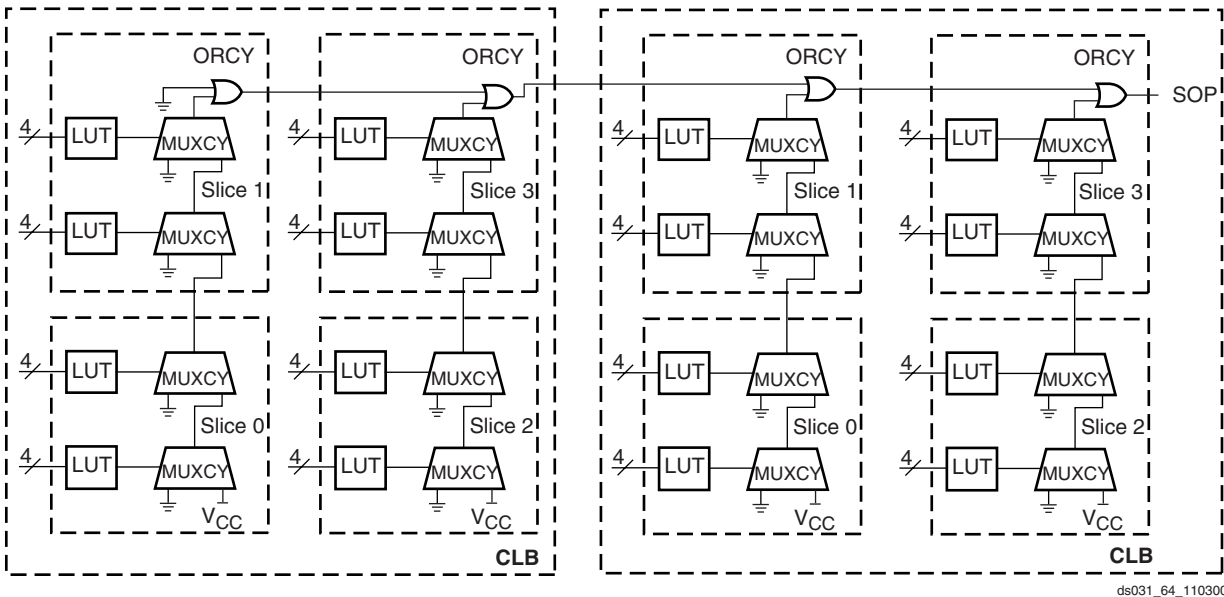


Figure 25: Horizontal Cascade Chain

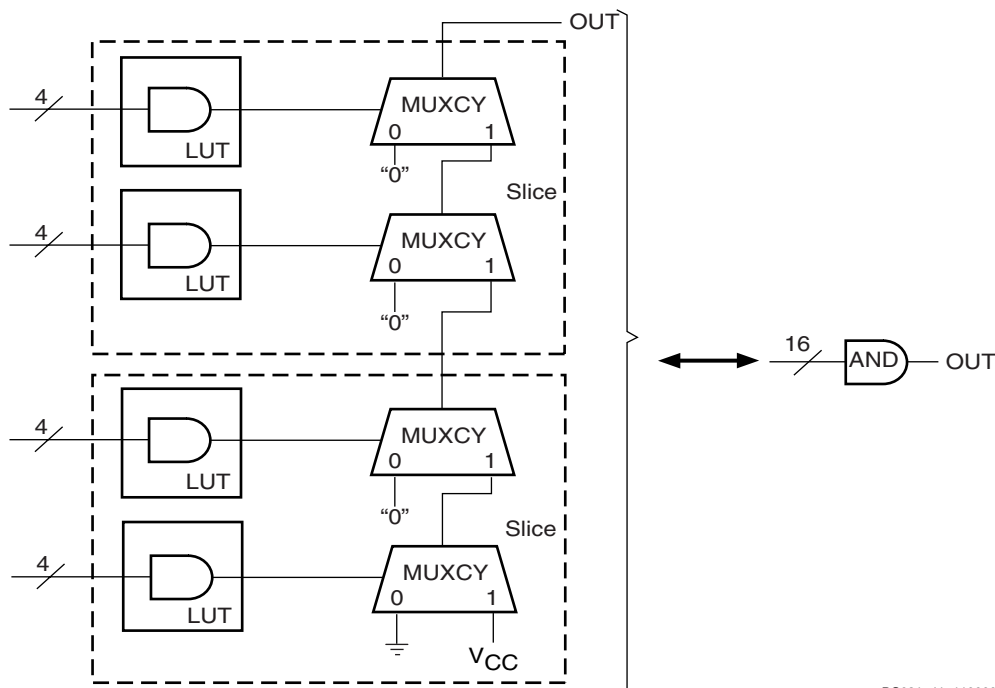


Figure 26: Wide-Input AND Gate (16 Inputs)

## 3-State Buffers

### Introduction

Each Virtex-II CLB contains two 3-state drivers (TBUFs) that can drive on-chip busses. Each 3-state buffer has its own 3-state control pin and its own input pin.

Each of the four slices have access to the two 3-state buffers through the switch matrix, as shown in Figure 27. TBUFs in neighboring CLBs can access slice outputs by direct connects. The outputs of the 3-state buffers drive horizontal routing resources used to implement 3-state busses.

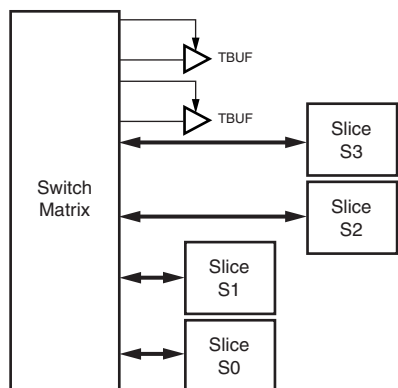


Figure 27: Virtex-II 3-State Buffers

The 3-state buffer logic is implemented using AND-OR logic rather than 3-state drivers, so that timing is more predictable and less load dependant especially with larger devices.

### Locations / Organization

Four horizontal routing resources per CLB are provided for on-chip 3-state busses. Each 3-state buffer has access alternately to two horizontal lines, which can be partitioned as shown in Figure 28. The switch matrices corresponding to SelectRAM memory and multiplier or I/O blocks are skipped.

### Number of 3-State Buffers

Table 10 shows the number of 3-state buffers available in each Virtex-II device. The number of 3-state buffers is twice the number of CLB elements.

Table 10: Virtex-II 3-State Buffers

Device	3-State Buffers per Row	Total Number of 3-State Buffers
XC2V40	16	128
XC2V80	16	256
XC2V250	32	768
XC2V500	48	1,536
XC2V1000	64	2,560
XC2V1500	80	3,840
XC2V2000	96	5,376
XC2V3000	112	7,168
XC2V4000	144	11,520
XC2V6000	176	16,896
XC2V8000	208	23,296

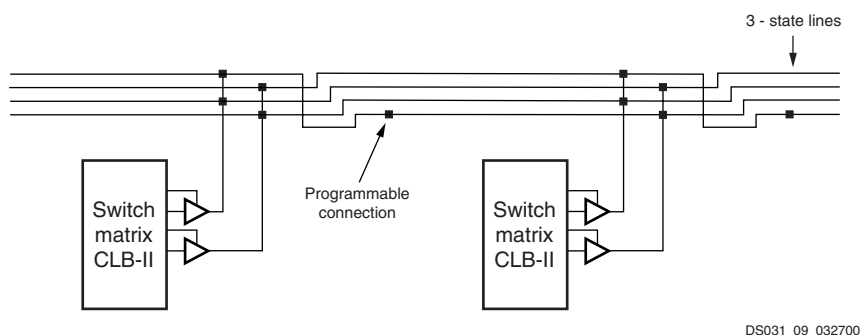


Figure 28: 3-State Buffer Connection to Horizontal Lines

### CLB/Slice Configurations

Table 11 summarizes the logic resources in one CLB. All of the CLBs are identical and each CLB or slice can be implemented in one of the configurations listed. Table 12 shows the available resources in all CLBs.

Table 11: Logic Resources in One CLB

Slices	LUTs	Flip-Flops	MULT_ANDs	Arithmetic & Carry-Chains	SOP Chains	Distributed SelectRAM	Shift Registers	TBUF
4	8	8	8	2	2	128 bits	128 bits	2

Table 12: Virtex-II Logic Resources Available in All CLBs

Device	CLB Array: Row x Column	Number of Slices	Number of LUTs	Max Distributed SelectRAM or Shift Register (bits)	Number of Flip-Flops	Number of Carry-Chains <sup>(1)</sup>	Number of SOP Chains <sup>(1)</sup>
XC2V40	8 x 8	256	512	8,192	512	16	16
XC2V80	16 x 8	512	1,024	16,384	1,024	16	32
XC2V250	24 x 16	1,536	3,072	49,152	3,072	32	48
XC2V500	32 x 24	3,072	6,144	98,304	6,144	48	64
XC2V1000	40 x 32	5,120	10,240	163,840	10,240	64	80
XC2V1500	48 x 40	7,680	15,360	245,760	15,360	80	96
XC2V2000	56 x 48	10,752	21,504	344,064	21,504	96	112
XC2V3000	64 x 56	14,336	28,672	458,752	28,672	112	128
XC2V4000	80 x 72	23,040	46,080	737,280	46,080	144	160
XC2V6000	96 x 88	33,792	67,584	1,081,344	67,584	176	192
XC2V8000	112 x 104	46,592	93,184	1,490,944	93,184	208	224

**Notes:**

1. The carry-chains and SOP chains can be split or cascaded.

## 18 Kbit Block SelectRAM Resources

### Introduction

Virtex-II devices incorporate large amounts of 18 Kbit block SelectRAM. These complement the distributed SelectRAM resources that provide shallow RAM structures implemented in CLBs. Each Virtex-II block SelectRAM is an 18 Kbit true dual-port RAM with two independently clocked and independently controlled synchronous ports that access a common storage area. Both ports are functionally identical. CLK, EN, WE, and SSR polarities are defined through configuration.

Each port has the following types of inputs: Clock and Clock Enable, Write Enable, Set/Reset, and Address, as well as separate Data/parity data inputs (for write) and Data/parity data outputs (for read).

Operation is synchronous; the block SelectRAM behaves like a register. Control, address and data inputs must (and need only) be valid during the set-up time window prior to a

rising (or falling, a configuration option) clock edge. Data outputs change as a result of the same clock edge.

### Configuration

The Virtex-II block SelectRAM supports various configurations, including single- and dual-port RAM and various data/address aspect ratios. Supported memory configurations for single- and dual-port modes are shown in Table 13.

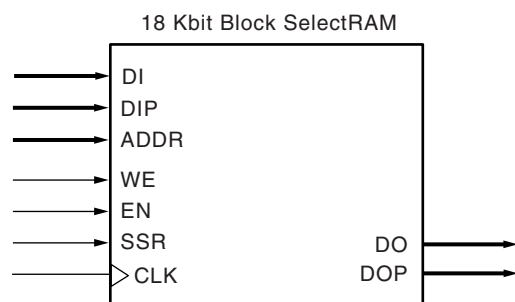
Table 13: Dual- and Single-Port Configurations

16K x 1 bit	2K x 9 bits
8K x 2 bits	1K x 18 bits
4K x 4 bits	512 x 36 bits

## Single-Port Configuration

As a single-port RAM, the block SelectRAM has access to the 18 Kbit memory locations in any of the 2K x 9-bit, 1K x 18-bit, or 512 x 36-bit configurations and to 16 Kbit memory locations in any of the 16K x 1-bit, 8K x 2-bit, or 4K x 4-bit configurations. The advantage of the 9-bit, 18-bit and 36-bit widths is the ability to store a parity bit for each eight bits. Parity bits must be generated or checked externally in user logic. In such cases, the width is viewed as 8 + 1, 16 + 2, or 32 + 4. These extra parity bits are stored and behave exactly as the other bits, including the timing parameters. Video applications can use the 9-bit ratio of Virtex-II block SelectRAM memory to advantage.

Each block SelectRAM cell is a fully synchronous memory as illustrated in Figure 29. Input data bus and output data bus widths are identical.



DS031\_10\_071602

Figure 29: 18 Kbit Block SelectRAM Memory in Single-Port Mode

Table 14: Dual-Port Mode Configurations

Port A	16K x 1	16K x 1	16K x 1	16K x 1	16K x 1	16K x 1
Port B	16K x 1	8K x 2	4K x 4	2K x 9	1K x 18	512 x 36
Port A	8K x 2	8K x 2	8K x 2	8K x 2	8K x 2	
Port B	8K x 2	4K x 4	2K x 9	1K x 18	512 x 36	
Port A	4K x 4	4K x 4	4K x 4	4K x 4		
Port B	4K x 4	2K x 9	1K x 18	512 x 36		
Port A	2K x 9	2K x 9	2K x 9			
Port B	2K x 9	1K x 18	512 x 36			
Port A	1K x 18	1K x 18				
Port B	1K x 18	512 x 36				
Port A	512 x 36					
Port B	512 x 36					

If both ports are configured in either 2K x 9-bit, 1K x 18-bit, or 512 x 36-bit configurations, the 18 Kbit block is accessible from port A or B. If both ports are configured in either 16K x 1-bit, 8K x 2-bit, or 4K x 4-bit configurations, the

## Dual-Port Configuration

As a dual-port RAM, each port of block SelectRAM has access to a common 18 Kbit memory resource. These are fully synchronous ports with independent control signals for each port. The data widths of the two ports can be configured independently, providing built-in bus-width conversion.

Table 14 illustrates the different configurations available on ports A & B.

16 K-bit block is accessible from Port A or Port B. All other configurations result in one port having access to an 18 Kbit memory block and the other port having access to a 16 K-bit subset of the memory block equal to 16 Kbits.

Each block SelectRAM cell is a fully synchronous memory, as illustrated in Figure 30. The two ports have independent inputs and outputs and are independently clocked.

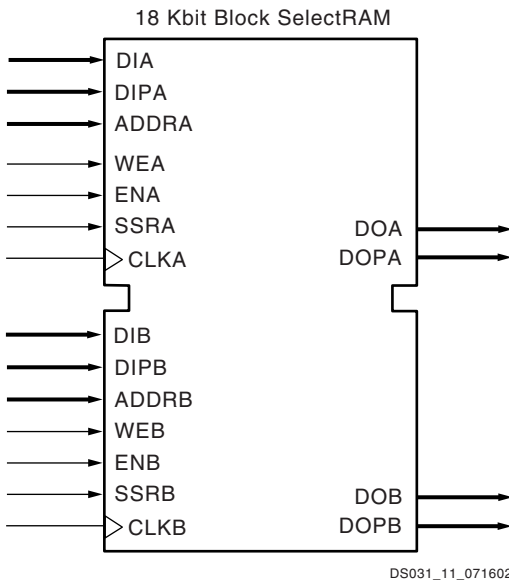


Figure 30: 18 Kbit Block SelectRAM in Dual-Port Mode

### Port Aspect Ratios

Table 15 shows the depth and the width aspect ratios for the 18 Kbit block SelectRAM. Virtex-II block SelectRAM also includes dedicated routing resources to provide an efficient interface with CLBs, block SelectRAM, and multipliers.

Table 15: 18 Kbit Block SelectRAM Port Aspect Ratio

Width	Depth	Address Bus	Data Bus	Parity Bus
1	16,384	ADDR[13:0]	DATA[0]	N/A
2	8,192	ADDR[12:0]	DATA[1:0]	N/A
4	4,096	ADDR[11:0]	DATA[3:0]	N/A
9	2,048	ADDR[10:0]	DATA[7:0]	Parity[0]
18	1,024	ADDR[9:0]	DATA[15:0]	Parity[1:0]
36	512	ADDR[8:0]	DATA[31:0]	Parity[3:0]

### Read/Write Operations

The Virtex-II block SelectRAM read operation is fully synchronous. An address is presented, and the read operation is enabled by control signals WEA and WEB in addition to ENA or ENB. Then, depending on clock polarity, a rising or falling clock edge causes the stored data to be loaded into output registers.

The write operation is also fully synchronous. Data and address are presented, and the write operation is enabled by control signals WEA or WEB in addition to ENA or ENB. Then, again depending on the clock input mode, a rising or falling clock edge causes the data to be loaded into the memory cell addressed.

A write operation performs a simultaneous read operation. Three different options are available, selected by configuration:

#### 1. "WRITE\_FIRST"

The "WRITE\_FIRST" option is a transparent mode. The same clock edge that writes the data input (DI) into the memory also transfers DI into the output registers DO as shown in Figure 31.

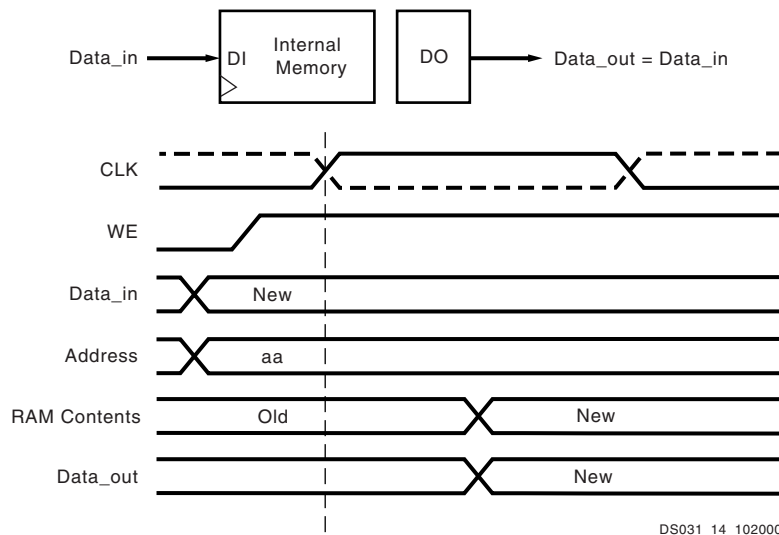


Figure 31: WRITE\_FIRST Mode

## 2. "READ\_FIRST"

The "READ\_FIRST" option is a read-before-write mode.

The same clock edge that writes data input (DI) into the memory also transfers the prior content of the memory cell addressed into the data output registers DO, as shown in Figure 32.

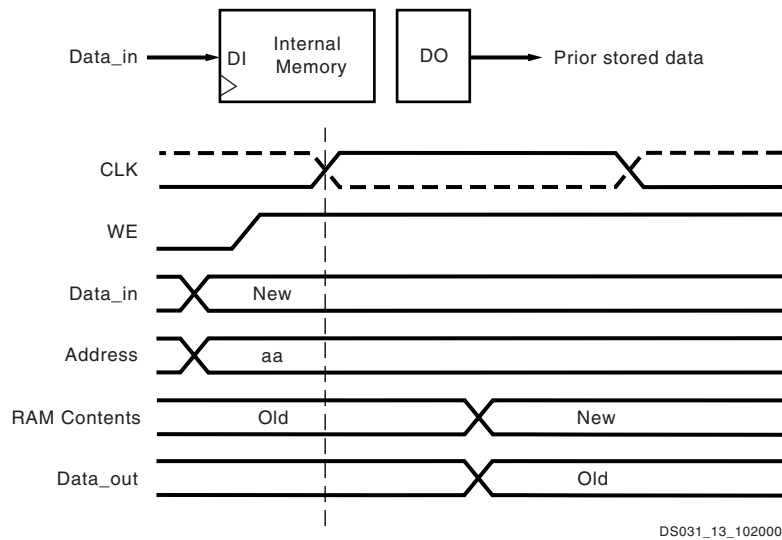


Figure 32: READ\_FIRST Mode

## 3. "NO\_CHANGE"

The "NO\_CHANGE" option maintains the content of the output registers, regardless of the write operation. The clock edge during the write mode has no effect on the content of the data output register DO. When the port is configured as "NO\_CHANGE", only a read operation loads a new value in the output register DO, as shown in Figure 33.

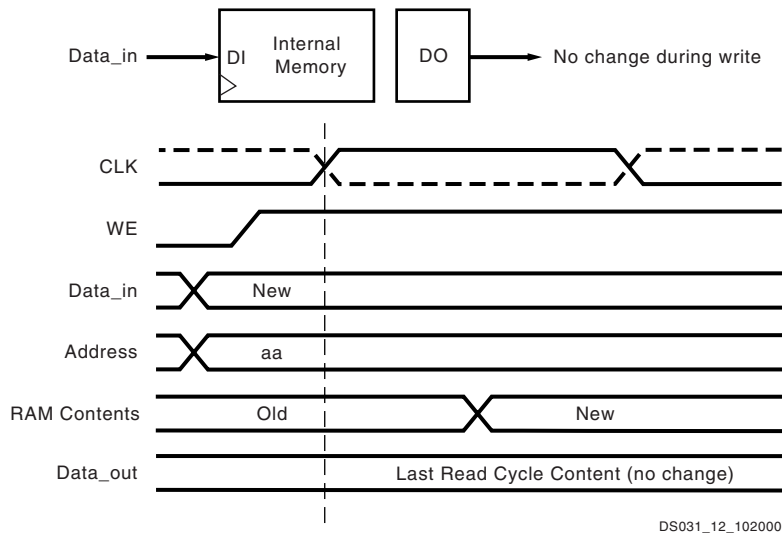


Figure 33: NO\_CHANGE Mode

## Control Pins and Attributes

Virtex-II SelectRAM memory has two independent ports with the control signals described in Table 16. All control inputs including the clock have an optional inversion.

Table 16: Control Functions

Control Signal	Function
CLK	Read and Write Clock
EN	Enable affects Read, Write, Set, Reset
WE	Write Enable
SSR	Set DO register to SRVAL (attribute)

Initial memory content is determined by the INIT\_xx attributes. Separate attributes determine the output register value after device configuration (INIT) and SSR is asserted (SRVAL). Both attributes (INIT\_B and SRVAL) are available for each port when a block SelectRAM resource is configured as dual-port RAM.

## Locations

Virtex-II SelectRAM memory blocks are located in either four or six columns. The number of blocks per column depends of the device array size and is equivalent to the

number of CLBs in a column divided by four. Column locations are shown in Table 17.

Table 17: SelectRAM Memory Floor Plan

Device	Columns	SelectRAM Blocks	
		Per Column	Total
XC2V40	2	2	4
XC2V80	2	4	8
XC2V250	4	6	24
XC2V500	4	8	32
XC2V1000	4	10	40
XC2V1500	4	12	48
XC2V2000	4	14	56
XC2V3000	6	16	96
XC2V4000	6	20	120
XC2V6000	6	24	144
XC2V8000	6	28	168

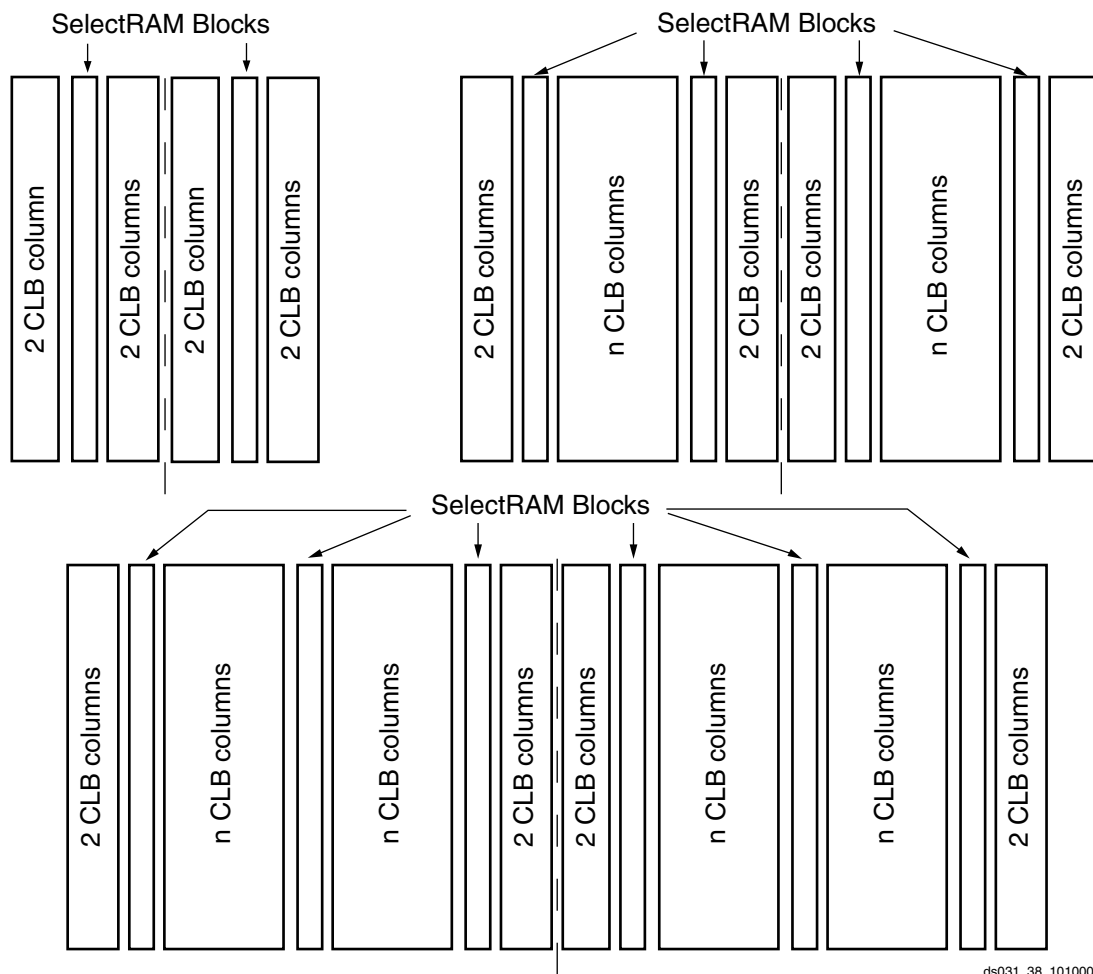


Figure 34: Block SelectRAM (2-column, 4-column, and 6-column)

## Total Amount of SelectRAM Memory

Table 18 shows the amount of block SelectRAM memory available for each Virtex-II device. The 18 Kbit SelectRAM blocks are cascadable to implement deeper or wider single- or dual-port memory resources.

Table 18: Virtex-II SelectRAM Memory Available

Device	Total SelectRAM Memory		
	Blocks	in Kbits	in Bits
XC2V40	4	72	73,728
XC2V80	8	144	147,456
XC2V250	24	432	442,368
XC2V500	32	576	589,824
XC2V1000	40	720	737,280
XC2V1500	48	864	884,736
XC2V2000	56	1,008	1,032,192
XC2V3000	96	1,728	1,769,472
XC2V4000	120	2,160	2,211,840
XC2V6000	144	2,592	2,654,208
XC2V8000	168	3,024	3,096,576

## 18-Bit x 18-Bit Multipliers

### Introduction

A Virtex-II multiplier block is an 18-bit by 18-bit 2's complement signed multiplier. Virtex-II devices incorporate many embedded multiplier blocks. These multipliers can be associated with an 18 Kbit block SelectRAM resource or can be used independently. They are optimized for high-speed operations and have a lower power consumption compared to an 18-bit x 18-bit multiplier in slices.

Each SelectRAM memory and multiplier block is tied to four switch matrices, as shown in Figure 35.

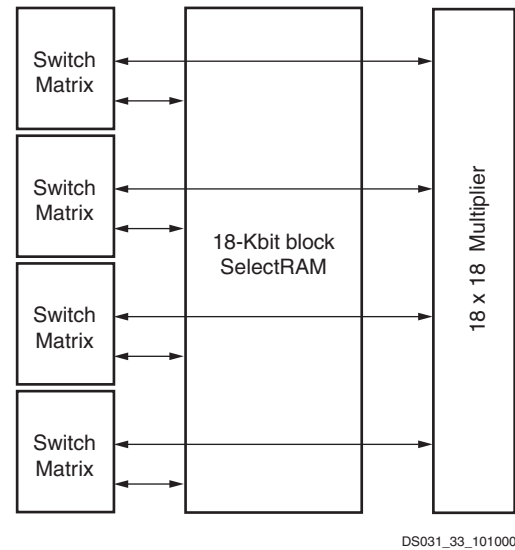


Figure 35: SelectRAM and Multiplier Blocks

### Association With Block SelectRAM Memory

The interconnect is designed to allow SelectRAM memory and multiplier blocks to be used at the same time, but some interconnect is shared between the SelectRAM and the multiplier. Thus, SelectRAM memory can be used only up to 18 bits wide when the multiplier is used, because the multiplier shares inputs with the upper data bits of the SelectRAM memory.

This sharing of the interconnect is optimized for an 18-bit-wide block SelectRAM resource feeding the multiplier. The use of SelectRAM memory and the multiplier with an accumulator in LUTs allows for implementation of a digital signal processor (DSP) multiplier-accumulator (MAC) function, which is commonly used in finite and infinite impulse response (FIR and IIR) digital filters.

### Configuration

The multiplier block is an 18-bit by 18-bit signed multiplier (2's complement). Both A and B are 18-bit-wide inputs, and the output is 36 bits. Figure 36 shows a multiplier block.

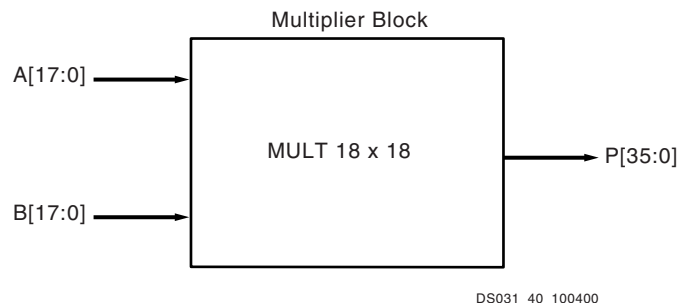


Figure 36: Multiplier Block

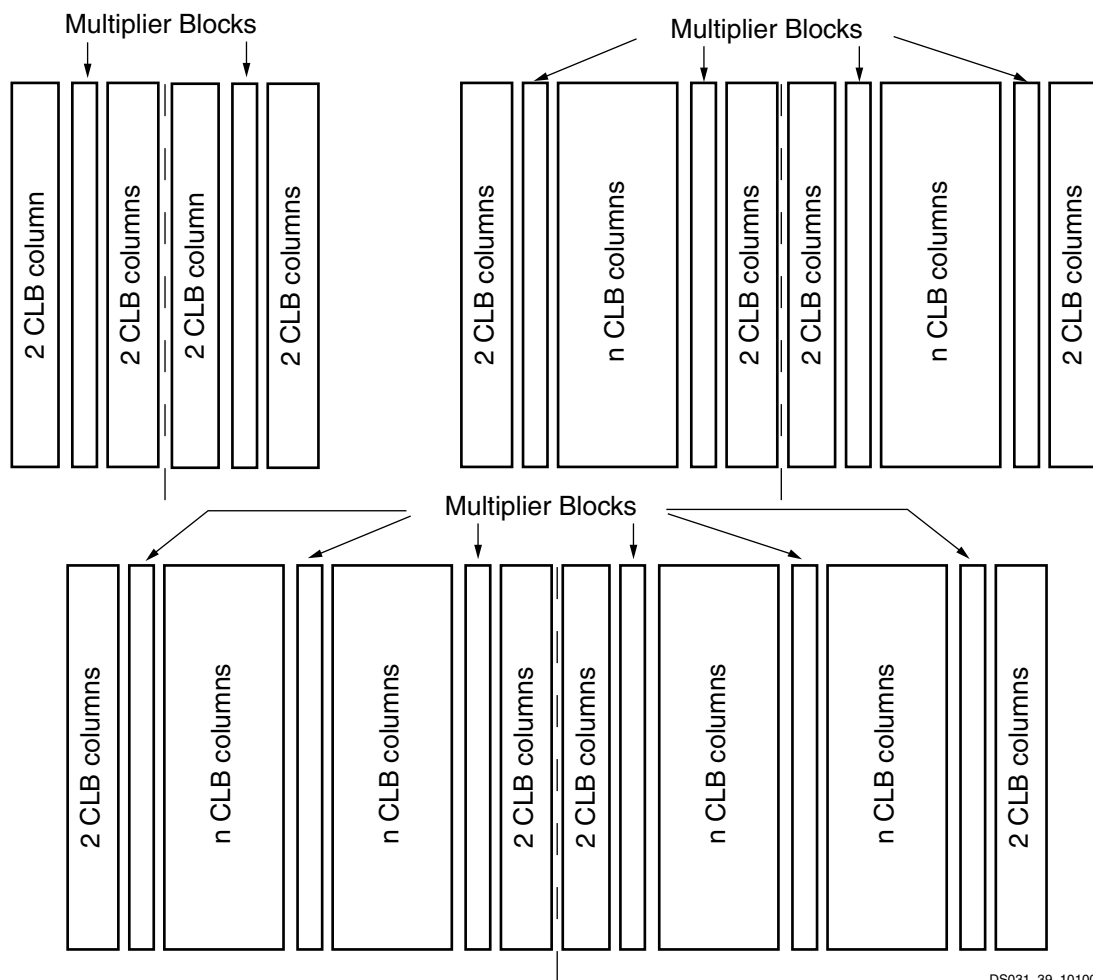
## Locations / Organization

Multiplier organization is identical to the 18 Kbit SelectRAM organization, because each multiplier is associated with an 18 Kbit block SelectRAM resource.

In addition to the built-in multiplier blocks, the CLB elements have dedicated logic to implement efficient multipliers in logic. (Refer to **Configurable Logic Blocks (CLBs)**).

Table 19: Multiplier Floor Plan

Device	Columns	Multipliers	
		Per Column	Total
XC2V40	2	2	4
XC2V80	2	4	8
XC2V250	4	6	24
XC2V500	4	8	32
XC2V1000	4	10	40
XC2V1500	4	12	48
XC2V2000	4	14	56
XC2V3000	6	16	96
XC2V4000	6	20	120
XC2V6000	6	24	144
XC2V8000	6	28	168



DS031\_39\_101000

Figure 37: Multipliers (2-column, 4-column, and 6-column)

## Global Clock Multiplexer Buffers

Virtex-II devices have 16 clock input pins that can also be used as regular user I/Os. Eight clock pads are on the top edge of the device, in the middle of the array, and eight are on the bottom edge, as illustrated in Figure 38.

The global clock multiplexer buffer represents the input to dedicated low-skew clock tree distribution in Virtex-II devices. Like the clock pads, eight global clock multiplexer buffers are on the top edge of the device and eight are on the bottom edge.

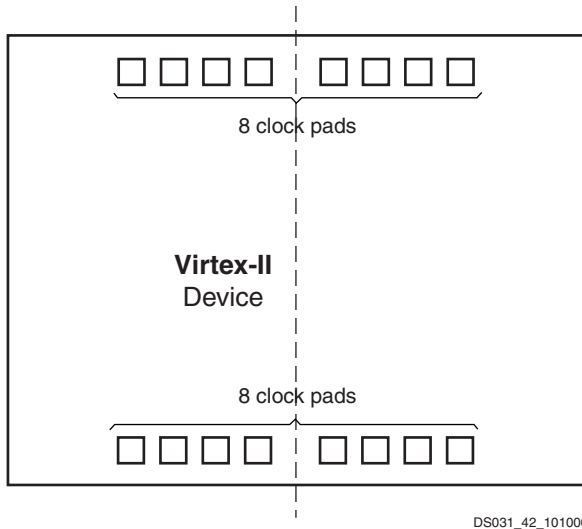


Figure 38: Virtex-II Clock Pads

Each global clock buffer can either be driven by the clock pad to distribute a clock directly to the device, or driven by the Digital Clock Manager (DCM), discussed in **Digital Clock Manager (DCM)**, page 30. Each global clock buffer

can also be driven by local interconnects. The DCM has clock output(s) that can be connected to global clock buffer inputs, as shown in Figure 39.

Global clock buffers are used to distribute the clock to some or all synchronous logic elements (such as registers in CLBs and IOBs, and SelectRAM blocks).

Eight global clocks can be used in each quadrant of the Virtex-II device. Designers should consider the clock distribution detail of the device prior to pin-locking and floorplanning (see the Virtex-II User Guide).

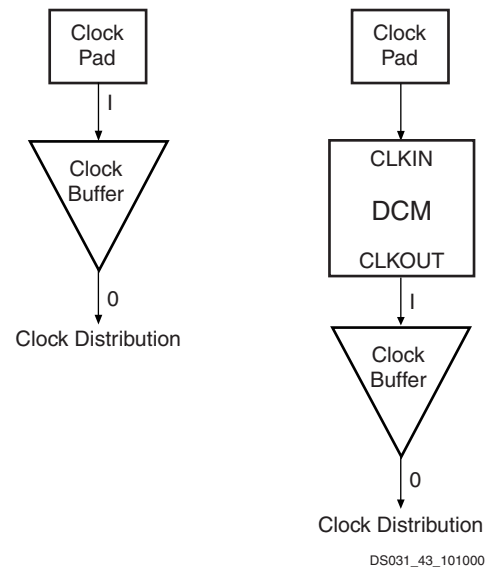


Figure 39: Virtex-II Clock Distribution Configurations

Figure 40 shows clock distribution in Virtex-II devices.

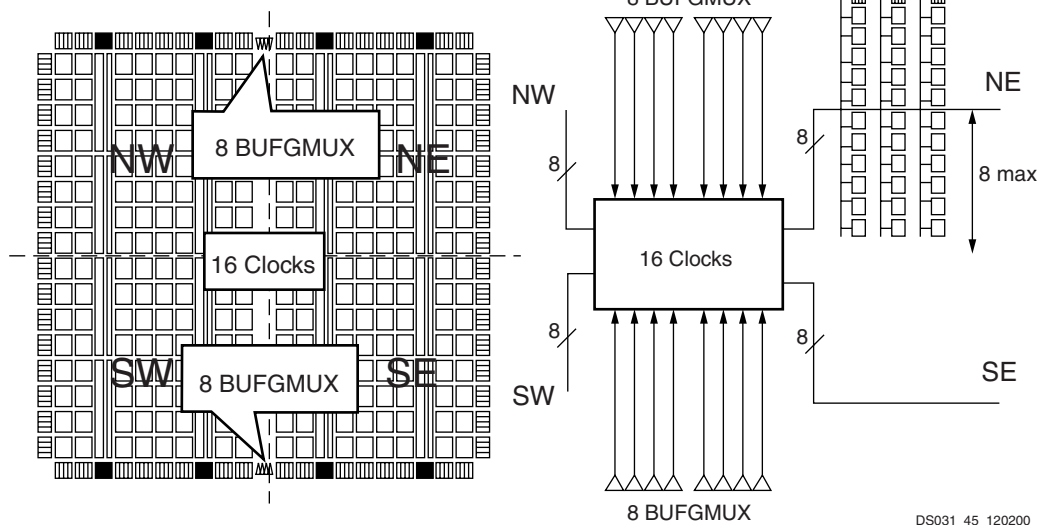


Figure 40: Virtex-II Clock Distribution

In each quadrant, up to eight clocks are organized in clock rows. A clock row supports up to 16 CLB rows (eight up and eight down). For the largest devices a new clock row is added, as necessary.

To reduce power consumption, any unused clock branches remain static.

Global clocks are driven by dedicated clock buffers (BUFG), which can also be used to gate the clock (BUFGCE) or to multiplex between two independent clock inputs (BUFGMUX).

The most common configuration option of this element is as a buffer. A BUFG function in this (global buffer) mode, is shown in Figure 41.

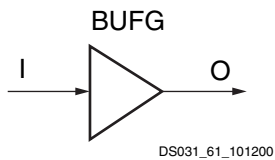


Figure 41: Virtex-II BUFG Function

The Virtex-II global clock buffer BUFG can also be configured as a clock enable/disable circuit (Figure 42), as well as a two-input clock multiplexer (Figure 43). A functional description of these two options is provided below. Each of them can be used in either of two modes, selected by configuration: rising clock edge or falling clock edge.

This section describes the rising clock edge option. For the opposite option, falling clock edge, just change all "rising" references to "falling" and all "High" references to "Low", except for the description of the CE or S levels. The rising clock edge option uses the BUFGCE and BUFGMUX primitives. The falling clock edge option uses the BUFGCE\_1 and BUFGMUX\_1 primitives.

## BUFGCE

If the CE input is active (High) prior to the incoming rising clock edge, this Low-to-High-to-Low clock pulse passes through the clock buffer. Any level change of CE during the incoming clock High time has no effect.

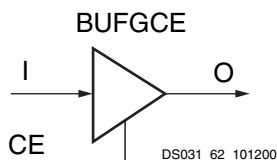


Figure 42: Virtex-II BUFGCE Function

If the CE input is inactive (Low) prior to the incoming rising clock edge, the following clock pulse does not pass through the clock buffer, and the output stays Low. Any level change of CE during the incoming clock High time has no effect. CE must not change during a short setup window just prior to the rising clock edge on the BUFGCE input I. Violating this setup time requirement can result in an undefined runt pulse output.

## BUFGMUX

BUFGMUX can switch between two unrelated, even asynchronous clocks. Basically, a Low on S selects the I0 input, a High on S selects the I1 input. Switching from one clock to the other is done in such a way that the output High and Low time is never shorter than the shortest High or Low time of either input clock. As long as the presently selected clock is High, any level change of S has no effect.

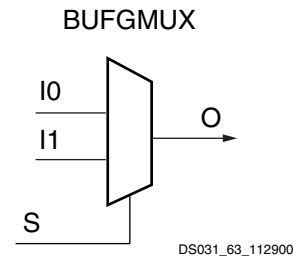


Figure 43: Virtex-II BUFGMUX Function

If the presently selected clock is Low while S changes, or if it goes Low after S has changed, the output is kept Low until the other ("to-be-selected") clock has made a transition from High to Low. At that instant, the new clock starts driving the output.

The two clock inputs can be asynchronous with regard to each other, and the S input can change at any time, except for a short setup time prior to the rising edge of the presently selected clock; that is, prior to the rising edge of the BUFGMUX output O. Violating this setup time requirement can result in an undefined runt pulse output.

All Virtex-II devices have 16 global clock multiplexer buffers.

Figure 44 shows a switchover from CLK0 to CLK1.

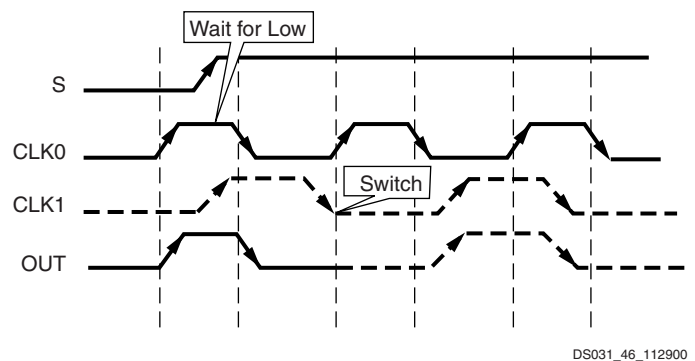


Figure 44: Clock Multiplexer Waveform Diagram

- The current clock is CLK0.
- S is activated High.
- If CLK0 is currently High, the multiplexer waits for CLK0 to go Low.
- Once CLK0 is Low, the multiplexer output stays Low until CLK1 transitions High to Low.
- When CLK1 transitions from High to Low, the output switches to CLK1.
- No glitches or short pulses can appear on the output.

## Local Clocking

In addition to global clocks, there are local clock resources in the Virtex-II devices. There are more than 72 local clocks in the Virtex-II family. These resources can be used for many different applications, including but not limited to

## Digital Clock Manager (DCM)

The Virtex-II DCM offers a wide range of powerful clock management features.

- **Clock De-skew:** The DCM generates new system clocks (either internally or externally to the FPGA), which are phase-aligned to the input clock, thus eliminating clock distribution delays.
- **Frequency Synthesis:** The DCM generates a wide range of output clock frequencies, performing very flexible clock multiplication and division.
- **Phase Shifting:** The DCM provides both coarse phase shifting and fine-grained phase shifting with dynamic phase shift control.

The DCM utilizes fully digital delay lines allowing robust high-precision control of clock phase and frequency. It also utilizes fully digital feedback systems, operating dynamically to compensate for temperature and voltage variations during operation.

Up to four of the nine DCM clock outputs can drive inputs to global clock buffers or global clock multiplexer buffers simultaneously (see Figure 45). All DCM clock outputs can simultaneously drive general routing resources, including routes to output buffers.

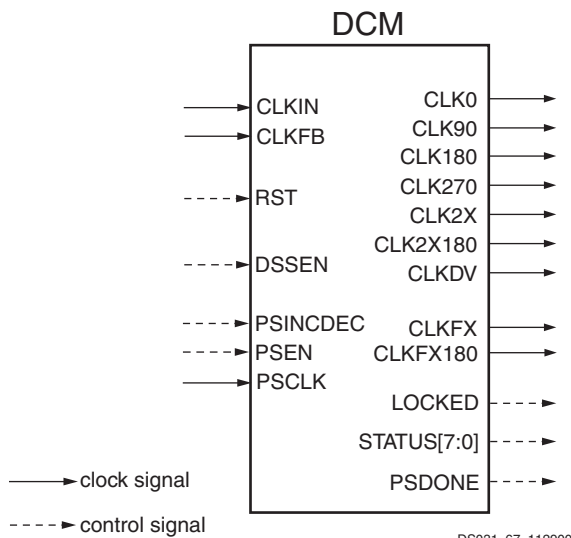


Figure 45: Digital Clock Manager

The DCM can be configured to delay the completion of the Virtex-II configuration process until after the DCM has achieved lock. This guarantees that the chip does not begin operating until after the system clocks generated by the DCM have stabilized.

memory interfaces. For example, even using only the left and right I/O banks, Virtex-II FPGAs can support up to 50 local clocks for DDR SDRAM. These interfaces can operate beyond 200 MHz on Virtex-II devices.

The DCM has the following general control signals:

- RST input pin: resets the entire DCM
- LOCKED output pin: asserted High when all enabled DCM circuits have locked.
- STATUS output pins (active High): shown in Table 20.

Table 20: DCM Status Pins

Status Pin	Function
0	Phase Shift Overflow
1	CLKIN Stopped
2	CLKFX Stopped
3	N/A
4	N/A
5	N/A
6	N/A
7	N/A

## Clock De-Skew

The DCM de-skews the output clocks relative to the input clock by automatically adjusting a digital delay line. Additional delay is introduced so that clock edges arrive at internal registers and block RAMs simultaneously with the clock edges arriving at the input clock pad. Alternatively, external clocks, which are also de-skewed relative to the input clock, can be generated for board-level routing. All DCM output clocks are phase-aligned to CLK0 and, therefore, are also phase-aligned to the input clock.

To achieve clock de-skew, the CLKFB input must be connected, and its source must be either CLK0 or CLK2X. Note that CLKFB must always be connected, unless only the CLKFX or CLKFX180 outputs are used and de-skew is not required.

## Frequency Synthesis

The DCM provides flexible methods for generating new clock frequencies. Each method has a different operating frequency range and different AC characteristics. The CLK2X and CLK2X180 outputs double the clock frequency. The CLKDV output creates divided output clocks with division options of 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5, 5.5, 6, 6.5, 7, 7.5, 8, 9, 10, 11, 12, 13, 14, 15, and 16.

The CLKFX and CLKFX180 outputs can be used to produce clocks at the following frequency:

$$\text{FREQ}_{\text{CLKFX}} = (\text{M}/\text{D}) * \text{FREQ}_{\text{CLKIN}}$$

where M and D are two integers. Specifications for M and D are provided under **DCM Timing Parameters** in [Data Sheet Module 3](#). By default, M=4 and D=1, which results in a clock output frequency four times faster than the clock input frequency (CLKIN).

CLK2X180 is phase shifted 180 degrees relative to CLK2X. CLKFX180 is phase shifted 180 degrees relative to CLKFX. All frequency synthesis outputs automatically have 50/50 duty cycles (with the exception of the CLKDV output when performing a non-integer divide in high-frequency mode).

Note that CLK2X and CLK2X180 are not available in high-frequency mode.

## Phase Shifting

The DCM provides additional control over clock skew through either coarse or fine-grained phase shifting. The CLK0, CLK90, CLK180, and CLK270 outputs are each phase shifted by ¼ of the input clock period relative to each other, providing coarse phase control. Note that CLK90 and CLK270 are not available in high-frequency mode.

Fine-phase adjustment affects all nine DCM output clocks. When activated, the phase shift between the rising edges of CLKIN and CLKFB is a specified fraction of the input clock period.

In variable mode, the PHASE\_SHIFT value can also be dynamically incremented or decremented as determined by PSINCDEC synchronously to PSCLK, when the PSEN input is active. [Figure 46](#) illustrates the effects of fine-phase shifting. For more information on DCM features, see the *Virtex-II User Guide*.

[Table 21](#) lists fine-phase shifting control pins, when used in variable mode.

Table 21: Fine-Phase Shifting Control Pins

Control Pin	Direction	Function
PSINCDEC	in	Increment or decrement
PSEN	in	Enable ± phase shift
PSCLK	in	Clock for phase shift
PSDONE	out	Active when completed

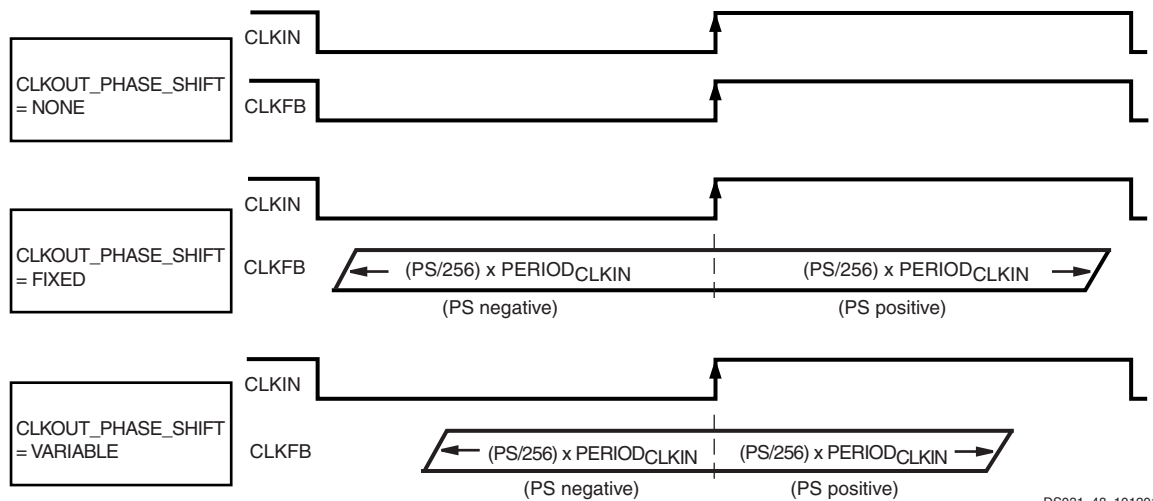


Figure 46: Fine-Phase Shifting Effects

Two separate components of the phase shift range must be understood:

- PHASE\_SHIFT attribute range
- FINE\_SHIFT\_RANGE DCM timing parameter range

The PHASE\_SHIFT attribute is the numerator in the following equation:

$$\text{Phase Shift (ns)} = (\text{PHASE\_SHIFT}/256) \times \text{PERIOD}_{CLKIN}$$

The full range of this attribute is always -255 to +255, but its practical range varies with CLKIN frequency, as constrained by the FINE\_SHIFT\_RANGE component, which represents the total delay achievable by the phase shift delay line. Total delay is a function of the number of delay taps used in the circuit. Across process, voltage, and temperature, this abso-

lute range is guaranteed to be as specified under **DCM Timing Parameters** in [Data Sheet Module 3](#).

Absolute range (fixed mode) =  $\pm \text{FINE\_SHIFT\_RANGE}$

Absolute range (variable mode) =  $\pm \text{FINE\_SHIFT\_RANGE}/2$

The reason for the difference between fixed and variable modes is as follows. For variable mode to allow symmetric, dynamic sweeps from -255/256 to +255/256, the DCM sets the "zero phase skew" point as the middle of the delay line, thus dividing the total delay line range in half. In fixed mode, since the PHASE\_SHIFT value never changes after configuration, the entire delay line is available for insertion into either the CLKIN or CLKFB path (to create either positive or negative skew).

Taking both of these components into consideration, the following are some usage examples:

- If  $PERIOD_{CLKIN} = 2 * FINE\_SHIFT\_RANGE$ , then  $PHASE\_SHIFT$  in fixed mode is limited to  $\pm 128$ , and in variable mode it is limited to  $\pm 64$ .
- If  $PERIOD_{CLKIN} = FINE\_SHIFT\_RANGE$ , then  $PHASE\_SHIFT$  in fixed mode is limited to  $\pm 255$ , and in variable mode it is limited to  $\pm 128$ .
- If  $PERIOD_{CLKIN} \leq 0.5 * FINE\_SHIFT\_RANGE$ , then  $PHASE\_SHIFT$  is limited to  $\pm 255$  in either mode.

## Operating Modes

The frequency ranges of DCM input and output clocks depend on the operating mode specified, either low-frequency mode or high-frequency mode, according to [Table 22](#). (For actual values, see **Virtex-II Switching Characteristics** in [Data Sheet Module 3](#)). The CLK2X,

CLK2X180, CLK90, and CLK270 outputs are not available in high-frequency mode.

High or low-frequency mode is selected by an attribute.

Table 22: DCM Frequency Ranges

Output Clock	Low-Frequency Mode		High-Frequency Mode	
	CLKIN Input	CLK Output	CLKIN Input	CLK Output
CLK0, CLK180	CLKIN_FREQ_DLL_LF	CLKOUT_FREQ_1X_LF	CLKIN_FREQ_DLL_HF	CLKOUT_FREQ_1X_HF
CLK90, CLK270	CLKIN_FREQ_DLL_LF	CLKOUT_FREQ_1X_LF	NA	NA
CLK2X, CLK2X180	CLKIN_FREQ_DLL_LF	CLKOUT_FREQ_2X_LF	NA	NA
CLKDV	CLKIN_FREQ_DLL_LF	CLKOUT_FREQ_DV_LF	CLKIN_FREQ_DLL_HF	CLKOUT_FREQ_DV_HF
CLKFX, CLKFX180	CLKIN_FREQ_FX_LF	CLKOUT_FREQ_FX_LF	CLKIN_FREQ_FX_HF	CLKOUT_FREQ_FX_HF

## Locations/Organization

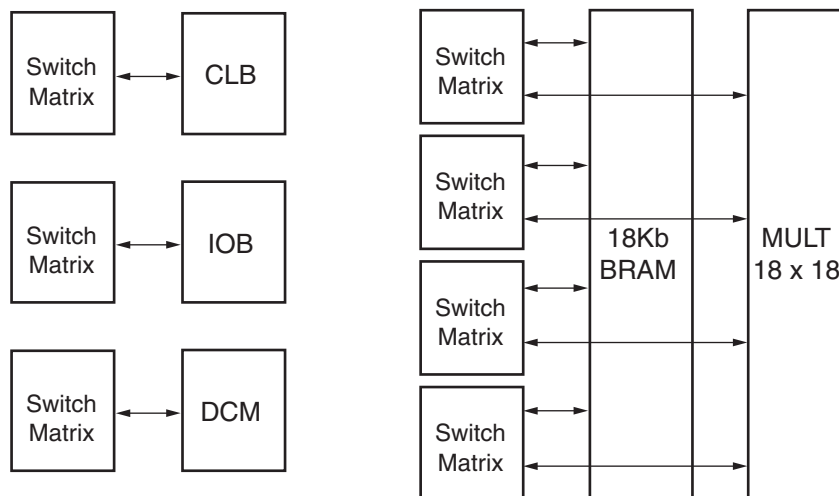
Virtex-II DCMs are placed on the top and bottom of each block RAM and multiplier column. The number of DCMs depends on the device size, as shown in [Table 23](#).

Table 23: DCM Organization

Device	Columns	DCMs
XC2V40	2	4
XC2V80	2	4
XC2V250	4	8
XC2V500	4	8
XC2V1000	4	8
XC2V1500	4	8
XC2V2000	4	8
XC2V3000	6	12
XC2V4000	6	12
XC2V6000	6	12
XC2V8000	6	12

## Active Interconnect Technology

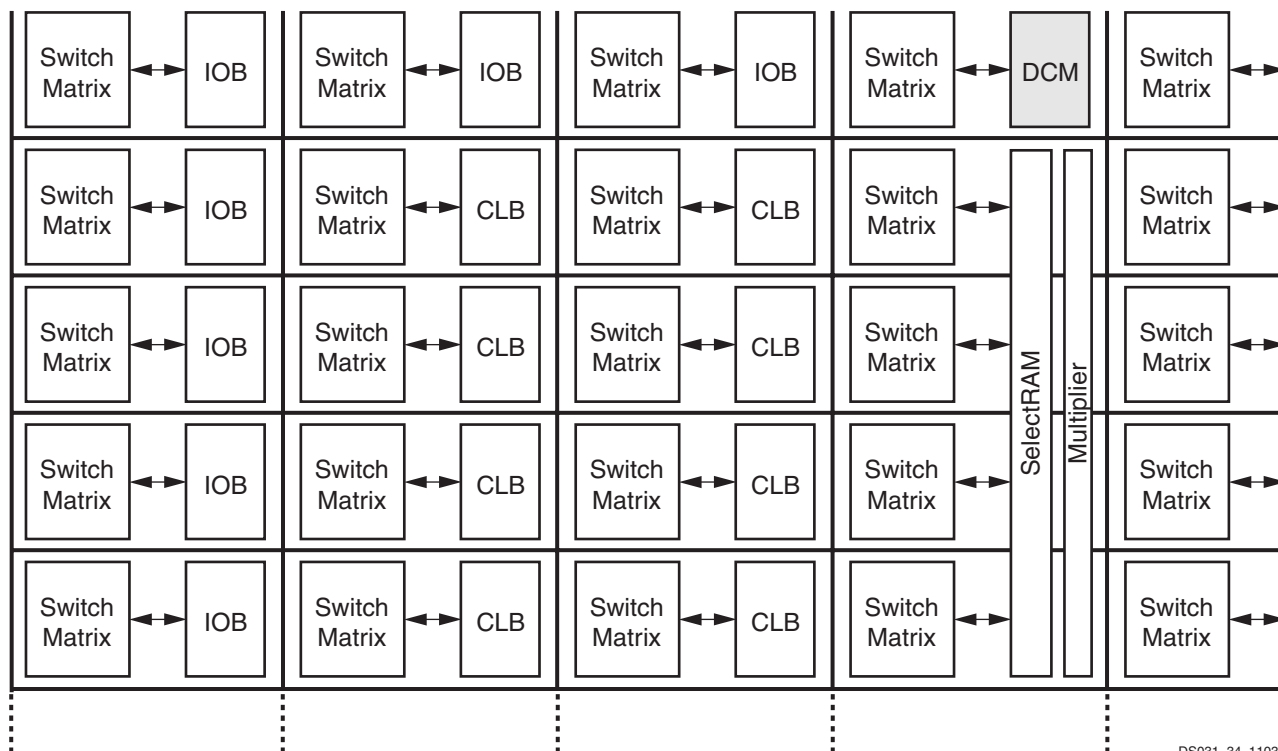
Local and global Virtex-II routing resources are optimized for speed and timing predictability, as well as to facilitate IP cores implementation. Virtex-II Active Interconnect Technology is a fully buffered programmable routing matrix. All routing resources are segmented to offer the advantages of a hierarchical solution. Virtex-II logic features like CLBs, IOBs, block RAM, multipliers, and DCMs are all connected to an identical switch matrix for access to global routing resources, as shown in Figure 47.



DS031\_55\_101000

Figure 47: Active Interconnect Technology

Each Virtex-II device can be represented as an array of switch matrixes with logic blocks attached, as illustrated in Figure 48.



DS031\_34\_110300

Figure 48: Routing Resources

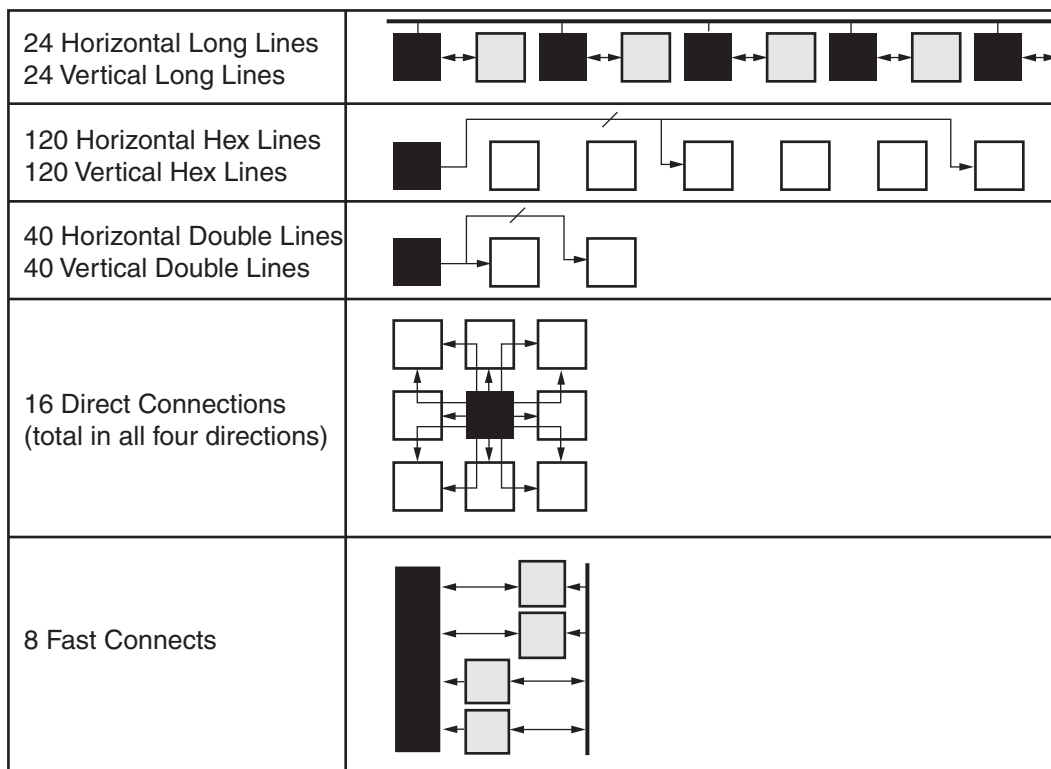
Place-and-route software takes advantage of this regular array to deliver optimum system performance and fast compile times. The segmented routing resources are essential to guarantee IP cores portability and to efficiently handle an

incremental design flow that is based on modular implementations. Total design time is reduced due to fewer and shorter design iterations.

## Hierarchical Routing Resources

Most Virtex-II signals are routed using the global routing resources, which are located in horizontal and vertical routing channels between each switch matrix.

As shown in Figure 49, Virtex-II has fully buffered programmable interconnections, with a number of resources counted between any two adjacent switch matrix rows or columns. Fanout has minimal impact on the performance of each net.



DS031\_60\_110200

Figure 49: Hierarchical Routing Resources

- The long lines are bidirectional wires that distribute signals across the device. Vertical and horizontal long lines span the full height and width of the device.
- The hex lines route signals to every third or sixth block away in all four directions. Organized in a staggered pattern, hex lines can only be driven from one end. Hex-line signals can be accessed either at the endpoints or at the midpoint (three blocks from the source).
- The double lines route signals to every first or second block away in all four directions. Organized in a staggered pattern, double lines can be driven only at their endpoints. Double-line signals can be accessed either at the endpoints or at the midpoint (one block from the source).
- The direct connect lines route signals to neighboring blocks: vertically, horizontally, and diagonally.

- The fast connect lines are the internal CLB local interconnections from LUT outputs to LUT inputs.

## Dedicated Routing

In addition to the global and local routing resources, dedicated signals are available.

- There are eight global clock nets per quadrant (see **Global Clock Multiplexer Buffers**).
- Horizontal routing resources are provided for on-chip 3-state busses. Four partitionable bus lines are provided per CLB row, permitting multiple busses within a row. (See **3-State Buffers**.)
- Two dedicated carry-chain resources per slice column (two per CLB column) propagate carry-chain MUXCY output signals vertically to the adjacent slice. (See **CLB/Slice Configurations**.)

- One dedicated SOP chain per slice row (two per CLB row) propagate ORCY output logic signals horizontally to the adjacent slice. (See **Sum of Products**.)
- One dedicated shift-chain per CLB connects the output of LUTs in shift-register mode to the input of the next LUT in shift-register mode (vertically) inside the CLB. (See **Shift Registers**, page 16.)

## Creating a Design

Creating Virtex-II designs is easy with Xilinx Integrated Synthesis Environment (ISE) development systems, which support advanced design capabilities, including ProActive Timing Closure, integrated logic analysis, and the fastest place and route runtimes in the industry. ISE solutions enable designers to get the performance they need, quickly and easily.

As a result of the ongoing cooperative development efforts between Xilinx and EDA Alliance partners, designers can take advantage of the benefits provided by EDA technologies in the programmable logic design process. Xilinx development systems are available in a number of easy to use configurations, collectively known as the ISE Series.

### ISE Alliance

The ISE Alliance solution is designed to plug and play within an existing design environment. Built using industry standard data formats and netlists, these stable, flexible products enable Alliance EDA partners to deliver their best design automation capabilities to Xilinx customers, along with the time to market benefits of ProActive Timing Closure.

### ISE Foundation

The ISE Foundation solution delivers the benefits of true HDL-based design in a seamlessly integrated design environment. An intuitive project navigator, as well as powerful HDL design and two HDL synthesis tools, ensure that high-quality results are achieved quickly and easily. The ISE Foundation product includes:

- State Diagram entry using Xilinx StateCAD
- Automatic HDL Testbench generation using Xilinx HDLBench
- HDL Simulation using ModelSim XE

### Design Flow

Virtex-II design flow proceeds as follows:

- Design Entry
- Synthesis
- Implementation
- Verification

Most programmable logic designers iterate through these steps several times in the process of completing a design.

### Design Entry

All Xilinx ISE development systems support the mainstream EDA design entry capabilities, ranging from schematic design to advanced HDL design methodologies. Given the high densities of the Virtex-II family, designs are created most efficiently using HDLs. To further improve their time to market, many Xilinx customers employ incremental, modular, and Intellectual Property (IP) design techniques. When properly used, these techniques further accelerate the logic design process.

To enable designers to leverage existing investments in EDA tools, and to ensure high performance design flows, Xilinx jointly develops tools with leading EDA vendors, including:

- Aldec®
- Cadence®
- Exemplar®
- Mentor Graphics®
- Model Technology®
- Synopsys®
- Synplicity®

Complete information on Alliance Series partners and their associated design flows is available at [www.xilinx.com](http://www.xilinx.com) on the Xilinx Alliance Series web page.

The ISE Foundation product offers schematic entry and HDL design capabilities as part of an integrated design solution - enabling one-stop shopping. These capabilities are powerful, easy to use, and they support the full portfolio of Xilinx programmable logic devices. HDL design capabilities include a color-coded HDL editor with integrated language templates, state diagram entry, and Core generation capabilities.

### Synthesis

The ISE Alliance product is engineered to support advanced design flows with the industry's best synthesis tools. Advanced design methodologies include:

- Physical Synthesis
- Incremental synthesis
- RTL floorplanning
- Direct physical mapping

The ISE Foundation product seamlessly integrates synthesis capabilities purchased directly from Exemplar, Synopsys, and Synplicity. In addition, it includes the capabilities of Xilinx Synthesis Technology.

A benefit of having two seamlessly integrated synthesis engines within an ISE design flow is the ability to apply alternative sets of optimization techniques on designs, helping to ensure that designers can meet even the toughest timing requirements.

## Design Implementation

The ISE Series development systems include Xilinx timing-driven implementation tools, frequently called “place and route” or “fitting” software. This robust suite of tools enables the creation of an intuitive, flexible, tightly integrated design flow that efficiently bridges “logical” and “physical” design domains. This simplifies the task of defining a design, including its behavior, timing requirements, and optional layout (or floorplanning), as well as simplifying the task of analyzing reports generated during the implementation process.

The Virtex-II implementation process is comprised of Synthesis, translation, mapping, place and route, and configuration file generation. While the tools can be run individually, many designers choose to run the entire implementation process with the click of a button. To assist those who prefer to script their design flows, Xilinx provides Xflow, an automated single command line process.

## Design Verification

In addition to conventional design verification using static timing analysis or simulation techniques, Xilinx offers powerful in-circuit debugging techniques using ChipScope ILA (Integrated Logic Analysis). The reconfigurable nature of Xilinx FPGAs means that designs can be verified in real time without the need for extensive sets of software simulation vectors.

For simulation, the system extracts post-layout timing information from the design database, and back-annotates this information into the netlist for use by the simulator. The back annotation features a variety of patented Xilinx techniques, resulting in the industry’s most powerful simulation flows. Alternatively, timing-critical portions of a design can be verified using the Xilinx static timing analyzer or a third party static timing analysis tool like Synopsys Prime Time™, by exporting timing data in the STAMP data format.

For in-circuit debugging, ChipScope ILA enables designers to analyze the real-time behavior of a device while operating at full system speeds. Logic analysis commands and captured data are transferred between the ChipScope software and ILA cores within the Virtex-II FPGA, using industry standard JTAG protocols. These JTAG transactions are driven over an optional download cable (MultiLINX or JTAG), connecting the Virtex device in the target system to a PC or workstation.

ChipScope ILA was designed to look and feel like a logic analyzer, making it easy to begin debugging a design immediately. Modifications to the desired logic analysis can be downloaded directly into the system in a matter of minutes.

## Other Unique Features of Virtex-II Design Flow

Xilinx design flows feature a number of unique capabilities. Among these are efficient incremental HDL design flows; a robust capability that is enabled by Xilinx exclusive hierarchical floorplanning capabilities. Another powerful design

capability only available in the Xilinx design flow is “Modular Design”, part of the Xilinx suite of team design tools, which enables autonomous design, implementation, and verification of design modules.

## Incremental Synthesis

Xilinx unique hierarchical floorplanning capabilities enable designers to create a programmable logic design by isolating design changes within one hierarchical “logic block”, and perform synthesis, verification and implementation processes on that specific logic block. By preserving the logic in unchanged portions of a design, Xilinx incremental design makes the high-density design process more efficient.

Xilinx hierarchical floorplanning capabilities can be specified using the high-level floorplanner or a preferred RTL floorplanner (see the Xilinx web site for a list of supported EDA partners). When used in conjunction with one of the EDA partners’ floorplanners, higher performance results can be achieved, as many synthesis tools use this more predictable detailed physical implementation information to establish more aggressive and accurate timing estimates when performing their logic optimizations.

## Modular Design

Xilinx innovative modular design capabilities take the incremental design process one step further by enabling the designer to delegate responsibility for completing the design, synthesis, verification, and implementation of a hierarchical “logic block” to an arbitrary number of designers - assigning a specific region within the target FPGA for exclusive use by each of the team members.

This team design capability enables an autonomous approach to design modules, changing the hand-off point to the lead designer or integrator from “my module works in simulation” to “my module works in the FPGA”. This unique design methodology also leverages the Xilinx hierarchical floorplanning capabilities and enables the Xilinx (or EDA partner) floorplanner to manage the efficient implementation of very high-density FPGAs.

## Configuration

Virtex-II devices are configured by loading application specific configuration data into the internal configuration memory. Configuration is carried out using a subset of the device pins, some of which are dedicated, while others can be re-used as general purpose inputs and outputs once configuration is complete.

Depending on the system design, several configuration modes are supported, selectable via mode pins. The mode pins M2, M1 and M0 are dedicated pins. An additional pin, HSWAP\_EN is used in conjunction with the mode pins to select whether user I/O pins have pull-ups during configuration. By default, HSWAP\_EN is tied High (internal pull-up) which shuts off the pull-ups on the user I/O pins during configuration. When HSWAP\_EN is tied Low, user I/Os have

pull-ups during configuration. Other dedicated pins are CCLK (the configuration clock pin), DONE, PROG\_B, and the boundary-scan pins: TDI, TDO, TMS, and TCK. Depending on the configuration mode chosen, CCLK can be an output generated by the FPGA, or an input accepting an externally generated clock. The configuration pins and boundary scan pins are independent of the  $V_{CCO}$ . The auxiliary power supply ( $V_{CCAUX}$ ) of 3.3V is used for these pins. All configuration pins are LVTTTL 12 mA. (See **Virtex-II DC Characteristics** in [Data Sheet Module 3](#).)

A persist option is available which can be used to force the configuration pins to retain their configuration function even after device configuration is complete. If the persist option is not selected then the configuration pins with the exception of CCLK, PROG\_B, and DONE can be used as user I/O in normal operation. The persist option does not apply to the boundary-scan related pins. The persist feature is valuable in applications which employ partial reconfiguration or reconfiguration on the fly.

## Configuration Modes

Virtex-II supports the following five configuration modes:

- Slave-serial mode
- Master-serial mode
- Slave SelectMAP mode
- Master SelectMAP mode
- Boundary-Scan mode (IEEE 1532/IEEE 1149)

A detailed description of configuration modes is provided in the *Virtex-II User Guide*.

### Slave-Serial Mode

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

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 internally to the DOUT pin. The data on the DOUT pin changes on the falling edge of CCLK.

Slave-serial mode is selected by applying <111> 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.

### Master-Serial Mode

In master-serial mode, the CCLK pin is an output pin. It is the Virtex-II FPGA device that drives the configuration clock on the CCLK pin to a Xilinx Serial PROM which in turn feeds bit-serial data to the DIN input. The FPGA accepts this data on each rising CCLK edge. After the FPGA has been

loaded, the data for the next device in a daisy-chain is presented on the DOUT pin after the falling CCLK edge.

The interface is identical to slave serial except that an internal oscillator is used to generate the configuration clock (CCLK). A wide range of frequencies can be selected for CCLK which always starts at a slow default frequency. Configuration bits then switch CCLK to a higher frequency for the remainder of the configuration.

### Slave SelectMAP Mode

The SelectMAP mode is the fastest configuration option. Byte-wide data is written into the Virtex-II FPGA device with a BUSY flag controlling the flow of data. An external data source provides a byte stream, CCLK, an active Low Chip Select (CS\_B) signal and a Write signal (RDWR\_B). If BUSY is asserted (High) by the FPGA, the data must be held until BUSY goes Low. Data can also be read using the SelectMAP mode. If RDWR\_B is asserted, configuration data is read out of the FPGA as part of a readback operation.

After configuration, the pins of the SelectMAP port can be used as additional user I/O. Alternatively, the port can be retained to permit high-speed 8-bit readback using the persist option.

Multiple Virtex-II FPGAs can be configured using the SelectMAP mode, and be made to start-up simultaneously. To configure multiple devices in this way, wire the individual CCLK, Data, RDWR\_B, and BUSY pins of all the devices in parallel. The individual devices are loaded separately by deasserting the CS\_B pin of each device in turn and writing the appropriate data.

### Master SelectMAP Mode

This mode is a master version of the SelectMAP mode. The device is configured byte-wide on a CCLK supplied by the Virtex-II FPGA device. Timing is similar to the Slave Serial-MAP mode except that CCLK is supplied by the Virtex-II FPGA.

### Boundary-Scan (JTAG, IEEE 1532) Mode

In boundary-scan mode, dedicated pins are used for configuring the Virtex-II device. The configuration is done entirely through the IEEE 1149.1 Test Access Port (TAP). Virtex-II device configuration using Boundary scan is compliant with IEEE 1149.1-1993 standard and the new IEEE 1532 standard for In-System Configurable (ISC) devices. The IEEE 1532 standard is backward compliant with the IEEE 1149.1-1993 TAP and state machine. The IEEE Standard 1532 for In-System Configurable (ISC) devices is intended to be programmed, reprogrammed, or tested on the board via a physical and logical protocol.

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.

Table 24: Virtex-II Configuration Mode Pin Settings

Configuration Mode <sup>(1)</sup>	M2	M1	M0	CCLK Direction	Data Width	Serial D <sub>OUT</sub> <sup>(2)</sup>
Master Serial	0	0	0	Out	1	Yes
Slave Serial	1	1	1	In	1	Yes
Master SelectMAP	0	1	1	Out	8	No
Slave SelectMAP	1	1	0	In	8	No
Boundary Scan	1	0	1	N/A	1	No

**Notes:**

1. The HSWAP\_EN pin controls the pullups. Setting M2, M1, and M0 selects the configuration mode, while the HSWAP\_EN pin controls whether or not the pullups are used.
2. Daisy chaining is possible only in modes where Serial D<sub>OUT</sub> is used. For example, in SelectMAP modes, the first device does NOT support daisy chaining of downstream devices.

Table 25 lists the total number of bits required to configure each device.

Table 25: Virtex-II Bitstream Lengths

Device	# of Configuration Bits
XC2V40	360,096
XC2V80	635,296
XC2V250	1,697,184
XC2V500	2,761,888
XC2V1000	4,082,592
XC2V1500	5,659,296
XC2V2000	7,492,000
XC2V3000	10,494,368
XC2V4000	15,659,936
XC2V6000	21,849,504
XC2V8000	29,063,072

### Configuration Sequence

The configuration of Virtex-II devices is a three-phase process after Power On Reset or POR. POR occurs when V<sub>CCINT</sub> is greater than 1.2V, V<sub>CCAUX</sub> is greater than 2.5V, and V<sub>CCO</sub> (bank 4) is greater than 1.5V. Once the POR voltages have been reached, the three-phase process begins.

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. The INIT\_B pin can be held Low using an open-drain driver. An open-drain is required since INIT\_B is a bidirectional open-drain pin that is held Low by a Virtex-II FPGA device 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.

The configuration process can also be initiated by asserting the PROG\_B pin. The end of the memory-clearing phase is signaled by the INIT\_B pin going High, and the completion of the entire process is signaled by the DONE pin going High. The Global Set/Reset (GSR) signal is pulsed after the last frame of configuration data is written but before the start-up sequence. The GSR signal resets all flip-flops on the device.

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 Write Enable (GWE) signal is released. This permits the internal storage elements to begin changing state in response to the logic and the user clock.

The relative timing of these events can be changed via configuration options in software. In addition, the GTS and GWE events can be made dependent on the DONE pins of multiple devices all going High, forcing the devices to start synchronously. The sequence can also be paused at any stage, until lock has been achieved on any or all DCMs, as well as the DCI.

### Readback

In this mode, configuration data from the Virtex-II FPGA device can be read back. Readback is supported only in the SelectMAP (master and slave) and Boundary Scan mode.

Along with the configuration data, it is possible to read back the contents of all registers, distributed SelectRAM, and block RAM resources. This capability is used for real-time debugging. For more detailed configuration information, see the Virtex-II Platform FPGA User Guide.

## Bitstream Encryption

Virtex-II devices have an on-chip decryptor using one or two sets of three keys for triple-key Data Encryption Standard (DES) operation. Xilinx software tools offer an optional encryption of the configuration data (bitstream) with a triple-key DES determined by the designer.

The keys are stored in the FPGA by JTAG instruction and retained by a battery connected to the  $V_{BATT}$  pin, when the device is not powered. Virtex-II devices can be configured with the corresponding encrypted bitstream, using any of the configuration modes described previously.

A detailed description of how to use bitstream encryption is provided in the *Virtex-II Platform FPGA User Guide*. For devices that support this feature, please contact your sales representative for specific ordering part number.

## Partial Reconfiguration

Partial reconfiguration of Virtex-II devices can be accomplished in either Slave SelectMAP mode or Boundary-Scan mode. Instead of resetting the chip and doing a full configuration, new data is loaded into a specified area of the chip, while the rest of the chip remains in operation. Data is loaded on a column basis, with the smallest load unit being a configuration “frame” of the bitstream (device size dependent).

Partial reconfiguration is useful for applications that require different designs to be loaded into the same area of a chip, or that require the ability to change portions of a design without having to reset or reconfigure the entire chip.

## Revision History

This section records the change history for this module of the data sheet.

Date	Version	Revision
11/07/00	1.0	Early access draft.
12/06/00	1.1	Initial release.
01/15/01	1.2	Added values to the tables in the <b>Virtex-II Performance Characteristics</b> and <b>Virtex-II Switching Characteristics</b> sections.
01/25/01	1.3	The data sheet was divided into four modules (per the current style standard). A note was added to <b>Table 1</b> .
04/02/01	1.5	<ul style="list-style-type: none"> <li>Under <b>Input/Output Individual Options</b>, the range of values for optional pull-up and pull-down resistors was changed to 10 - 60 K<math>\Omega</math> from 50 - 100 K<math>\Omega</math>.</li> <li>Skipped v1.4 to sync up modules. Reverted to traditional double-column format.</li> </ul>
07/30/01	1.6	<ul style="list-style-type: none"> <li>Added <b>Table 6</b>.</li> <li>Changed definition of multiply and divide integer ranges under <b>Digital Clock Manager (DCM)</b>.</li> <li>Made numerous minor edits throughout this module.</li> </ul>
10/02/01	1.7	<ul style="list-style-type: none"> <li>Updated descriptions under <b>Digitally Controlled Impedance (DCI)</b>, <b>Global Clock Multiplexer Buffers</b>, <b>Digital Clock Manager (DCM)</b>, and <b>Creating a Design</b>.</li> </ul>
10/12/01	1.8	<ul style="list-style-type: none"> <li>Made clarifying edits under <b>Digital Clock Manager (DCM)</b>.</li> </ul>
11/29/01	1.9	<ul style="list-style-type: none"> <li>Changed bitstream lengths for each device in <b>Table 25</b>.</li> </ul>
07/16/02	2.0	<ul style="list-style-type: none"> <li>Updated compatible input standards listed in <b>Table 6</b>.</li> </ul>
09/26/02	2.1	<ul style="list-style-type: none"> <li>Changed number of resources available to the XC2V40 device in <b>Table 12</b>.</li> <li>Clarified Power On Reset information under <b>Configuration Sequence</b>.</li> </ul>
12/06/02	2.1.1	<ul style="list-style-type: none"> <li>Cosmetic edits.</li> </ul>
05/07/03	2.1.2	<ul style="list-style-type: none"> <li>Added qualification note to <b>Figure 13</b>, page 11.</li> <li>Corrected sentence in section <b>Input/Output Individual Options</b>, page 4, to read “The optional weak-keeper circuit is connected to each <i>user I/O pad</i>.”</li> <li>Corrected typographical errors in <b>Table 3</b> for names of HSTL_[x]_DCI_18 standards.</li> </ul>

Date	Version	Revision
06/19/03	2.2	<ul style="list-style-type: none"> <li>Removed Compatible Output Standards and Compatible Input Standards tables.</li> <li>Added new <b>Table 5, Summary of Voltage Supply Requirements for All Input and Output Standards</b>. This table replaces deleted I/O standards tables.</li> <li>Added section <b>Rules for Combining I/O Standards in the Same Bank</b>, page 6.</li> </ul>
08/01/03	3.0	All Virtex-II devices and speed grades now Production. See Table 13, Module 3.
10/14/03	3.1	<ul style="list-style-type: none"> <li>Added section <b>Local Clocking</b>, page 30.</li> <li><b>Table 1, page 1:</b> <ul style="list-style-type: none"> <li>Added SSTL18_I and SSTL18_II.</li> <li>Corrected names of 1.8V HSTL_I-IV standards to "HSTL_I-IV_18".</li> <li>Corrected Input <math>V_{REF}</math> for HSTL_III-IV_18 from 1.08V to 1.1V.</li> <li>Changed "N/A" to "N/R" (no requirement).</li> </ul> </li> <li><b>Table 2, page 2:</b> <ul style="list-style-type: none"> <li>Changed "N/A" to "N/R" (no requirement).</li> </ul> </li> <li><b>Table 3, page 2:</b> <ul style="list-style-type: none"> <li>Added SSTL18_I_DCI, SSTL18_II_DCI, LVDS_33_DCI, LVDSEXT_33_DCI, LVDS_25_DCI, and LVDSEXT_25_DCI.</li> <li>Corrected Input <math>V_{REF}</math> for HSTL_III-IV_18 from 1.08V to 1.1V.</li> </ul> </li> <li>Sections <b>Slave-Serial Mode</b> and <b>Master-Serial Mode</b>, page 37: Changed "rising" to "falling" edge with respect to DOUT.</li> <li>Added verbiage to section <b>Bitstream Encryption</b>, page 39: "For devices that support this feature, please contact your sales representative for specific ordering part number."</li> </ul>

## Virtex-II Data Sheet

The Virtex-II Data Sheet contains the following modules:

- [Virtex™-II Platform FPGAs: Introduction and Overview \(Module 1\)](#)
- [Virtex™-II Platform FPGAs: Detailed Description \(Module 2\)](#)
- [Virtex™-II Platform FPGAs: DC and Switching Characteristics \(Module 3\)](#)
- [Virtex™-II Platform FPGAs: Pinout Information \(Module 4\)](#)