Spartan-3E Libraries Guide for Schematic Designs

ISE 7.1i

XILINX®

About this Guide

The *Spartan-3E™ Libraries Guide for Schematic Designs* is part of the ISE documentation collection. A separate version of this guide is also available for users who prefer to work in a hardware description language (HDL) in their circuit design activities. (See *Spartan-3E™ Libraries Guide for HDL Designs*.)

Guide Contents

This guide contains the following:

- Information about additional resources and conventions used in this guide.
- A general introduction to the Spartan-3E primitives and macros.
- A listing of the Functional Categories into which Spartan-3E design elements are organized.
- Individual sections for each of the applicable design elements.
- Referrals to additional sources of information.

Additional Resources

For additional information, go to http://www.xilinx.com/support. The following table lists some of the resources you can access from this website. You can also directly access these resources using the provided URLs.

Resource	Description/URL
Tutorials	Tutorials covering Xilinx design flows, from design entry to verification and debugging http://www.xilinx.com/support/techsup/tutorials/index.htm
Answer Browser	Database of Xilinx solution records http://www.xilinx.com/xlnx/xil_ans_browser.jsp
Application Notes	Descriptions of device-specific design techniques and approaches http://www.xilinx.com/apps/appsweb.htm
Problem Solvers	Interactive tools that allow you to troubleshoot your design issues http://www.xilinx.com/support/troubleshoot/psolvers.htm
Tech Tips	Latest news, design tips, and patch information for the Xilinx design environment http://www.xilinx.com/xlnx/xil_tt_home.jsp

Conventions

This document uses the following conventions. An example illustrates each convention.

Typographical

The following typographical conventions are used in this document:

Convention	Meaning or Use	Example
`Courier font`	Messages, prompts, and program files that the system displays	`speed grade: - 100`
`Courier bold`	Literal commands that you enter in a syntactical statement	**`ngdbuild`** *design_name*
Helvetica bold	Commands that you select from a menu	**File → Open**
	Keyboard shortcuts	**Ctrl+C**
Italic font	Variables in a syntax statement for which you must supply values	**`ngdbuild`** *design_name*
	References to other manuals	See the *Development System Reference Guide* for more information.
	Emphasis in text	If a wire is drawn so that it overlaps the pin of a symbol, the two nets are *not* connected.
Square brackets [ ]	An optional entry or parameter. However, in bus specifications, such as **`bus[7:0]`**, they are required.	**`ngdbuild`** [*`option_name`*] *`design_name`*
Braces { }	A list of items from which you must choose one or more	**`lowpwr ={on|off}`**
Vertical bar |	Separates items in a list of choices	**`lowpwr ={on|off}`**

Vertical ellipsis
.
.
.
	Repetitive material that has been omitted	`IOB #1: Name = QOUT'`
		`IOB #2: Name = CLKIN'`
		.
		.
		.

Horizontal ellipsis **`...`**	Repetitive material that has been omitted	**`allow block`** *block_name loc1 loc2 ... locn;*

Online Document

The following conventions are used in this document:

Convention	Meaning or Use	Example
Blue text	Cross-reference link to a location in the current document	See the section "Additional Resources" for details.
Red text	Cross-reference link to a location in another document	See Figure 2-5 in the *Virtex-II Handbook.*
Blue, underlined text	Hyperlink to a website (URL)	Go to http://www.xilinx.com for the latest speed files.

Introduction

Xilinx maintains software libraries containing hundreds of functional design elements (primitives and macros) for different device architectures. New functional elements are assembled with each release of development system software. The catalog of design elements is known as the Xilinx Unified Libraries. Elements in these libraries are common to multiple Xilinx device architectures. This “unified” approach means that you can use your circuit design created with “unified” library elements across many current Xilinx device architectures that recognize the element you are using.

Elements that exist in multiple architectures look and function the same, but their implementations might differ to make them more efficient for a particular architecture. A separate library still exists for each architecture and several hundred design elements (primitives and macros) are available across multiple device architectures, providing a common base for programmable logic designs.

If you have active designs that were created with former Xilinx library primitives or macros, you may need to change references to the design elements that you were using to reflect the Unified Libraries elements.

During 2004, libraries guides began to be published only for the latest available architectures (e.g., Virtex-4, Spartan-3E, etc.). This architecture-specific approach is also published in a two-volume set: one for designers who prefer to use hardware description language (HDL), and one for those who prefer to use schematics.

A compendium of the same kind of information for all the most recent architectures continues to be provided as part of the Integrated Software Environment (ISE) program.

The *Spartan-3E Libraries Guide* describes the primitive and macro logic elements supported under the Spartan-3E architecture. Common logic functions can be implemented with these elements and more complex functions can be built by combining macros and primitives.

Functional Categories

The functional categories list the available design elements in each category along with a brief description of each element that is supported under each Xilinx architecture.

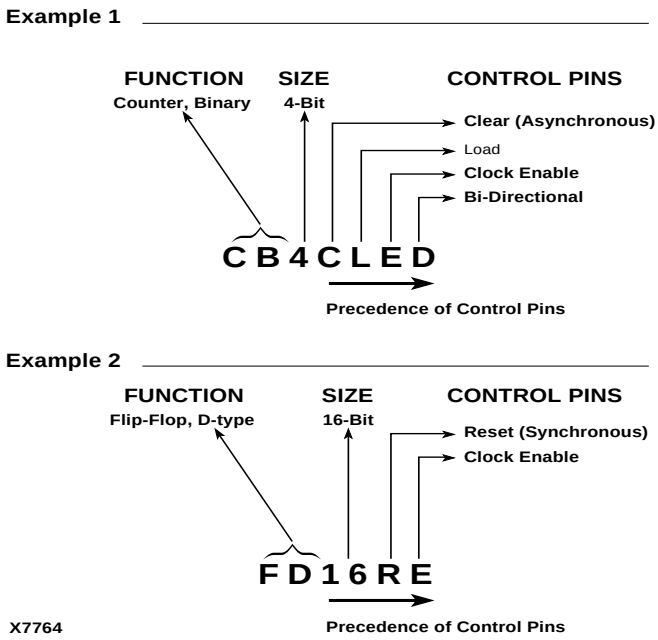
Schematic Symbols

Schematic symbols are included for each device library, and these schematics are illustrated on the first page of the written description of each element in this guide.

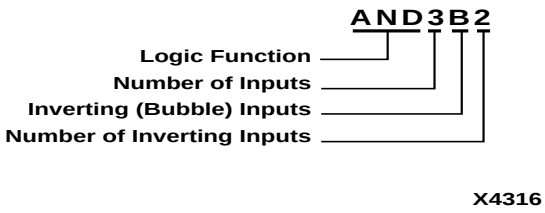
Design elements with bussed or multiple I/O pins (2-, 4-, 8-, 16-bit versions) typically include just one schematic -- generally the 8-bit version. When only one schematic is included, implementation of the smaller and larger elements differs only in the number of sections. In cases where an 8-bit version is very large, an appropriate smaller element serves as the schematic example.

Naming Conventions

Examples of the general naming conventions for the unified library elements are shown in the following figures.



Naming Conventions



Combinatorial Naming Conventions

Attributes and Constraints

Attributes are characteristics that help define the functionality of primitive elements. Where appropriate, an "Available Attributes" table is included in the individual descriptions of design elements. Constraints are instructions placed on components, or nets, to indicate their placement, implementation, optimization, and so forth. The *Constraints Guide* provides additional information about constraints and certain attributes.

Table of Contents

About this Guide

Guide Contents 3
Additional Resources 3
Conventions 4
Typographical 4
Online Document 4
Introduction 5
Functional Categories 5
Schematic Symbols 5
Naming Conventions 5
Attributes and Constraints 6

Functional Categories

Arithmetic Functions 13
Clock Components 13
Comparators 13
Config/BSCAN 14
Counters 14
Decoders 15
General 15
I/O Components 15
Internal Buffers 16
Logic Gates 16
MUXes 17
RAM/ROM 18
Registers & Latches 18
Shift Registers 21
Slice/CLB Primitives 21

About the Spartan-3E Design Elements

ACC4, 8, 16 25
ADD4, 8, 16 29
ADSU4, 8, 16 31
AND2-9 35
AND12, 16 37
BRLSHFT4, 8 39
BSCAN_SPARTAN3 41
BUF 43
BUFG 45
BUFGCE 47
BUFGCE_1 49
BUFGMUX 51

BUFGMUX_1 53
CAPTURE_SPARTAN3 55
CB2CE, CB4CE, CB8CE, CB16CE 57
CB2RE, CB4RE, CB8RE, CB16RE 59
CB2RLE,CB4RLE, CB8RLE, CB16RLE 61
CC8CE, CC16CE 63
CC8CLE, CC16CLE 65
CC8CLED, CC16CLED 67
CC8RE, CC16RE 69
CD4CE 71
CD4CLE 75
CD4RE 77
CD4RLE 79
CJ4CE, CJ5CE, CJ8CE 81
CJ4RE, CJ5RE, CJ8RE 83
COMP2, 4, 8, 16 85
COMPM2, 4, 8, 16 87
COMPMC8, 16 91
CR8CE, CR16CE 93
D2_4E 95
D3_8E 97
D4_16E 99
DCM 101
DEC_CC4, 8, 16 105
DECODE4, 8, 16 107
DECODE32, 64 109
FD 111
FD_1 113
FD4CE, FD8CE, FD16CE 115
FD4RE, FD8RE, FD16RE 119
FDC 121
FDC_1 123
FDCE 125
FDCE_1 127
FDCP 129
FDCP_1 131
FDCPE 133
FDCPE_1 135
FDE 137
FDE_1 139
FDP 141
FDP_1 143

FDPE 145
FDPE_1 147
FDR 149
FDR_1 151
FDRE 153
FDRE_1 155
FDRS 157
FDRS_1 159
FDRSE 161
FDRSE_1 163
FDS 165
FDS_1 167
FDSE 169
FDSE_1 171
FJKC 173
FJKCE 175
FJKP 177
FJKPE 179
FJKRSE 181
FJKSRE 183
FTC 185
FTCE 187
FTCLE 189
FTCLEX 191
FTP 193
FTPE 195
FTPLE 197
FTRSE 199
FTRSLE 201
FTSRE 203
FTSRLE 205
GND 207
IBUF, 4, 8, 16 209
IBUFDS 211
IBUFG 213
IBUFGDS 215
IDDR2 217
IFD, 4, 8, 16 219
IFD_1 223
IFDI 225
IFDI_1 227
IFDX_1 229

IFDX, 4, 8, 16 231
IFDXI_1 235
IFDXI 237
ILD_1 239
ILD, 4, 8, 16 241
ILDI 245
ILDI_1 247
ILDX_1 249
ILDX, 4, 8, 16 251
ILDXI 255
ILDXI_1 257
INV, 4, 8, 16 259
IOBUF 261
IOBUFDS 263
KEEPER 265
LD 267
LD_1 269
LD4, 8, 16 271
LD4CE, LD8CE, LD16CE 273
LDC 275
LDC_1 277
LDCE 279
LDCE_1 281
LDCP 283
LDCP_1 285
LDCPE 287
LDCPE_1 289
LDE 291
LDE_1 293
LDP 295
LDP_1 297
LDPE 299
LDPE_1 301
LUT1, 2, 3, 4 303
LUT1_D, LUT2_D, LUT3_D, LUT4_D 305
LUT1_L, LUT2_L, LUT3_L, LUT4_L 307
M2_1 309
M2_1B1 311
M2_1B2 313
M2_1E 315
M4_1E 317
M8_1E 319

M16_1E 321
MULT_AND 323
MULT18X18SIO 325
MUXCY 327
MUXCY_D 329
MUXCY_L 331
MUXF5 333
MUXF5_D 335
MUXF5_L 337
MUXF6 339
MUXF6_D 341
MUXF6_L 343
MUXF7 345
MUXF7_D 347
MUXF7_L 349
MUXF8 351
MUXF8_D 353
MUXF8_L 355
NAND2-9 357
NAND12, 16 359
NOR2-9 361
NOR12, 16 363
OBUF, 4, 8, 16 365
OBUFDS 367
OBUFT, 4, 8, 16 369
OBUFTDS 371
ODDR2 373
OFD_1 375
OFD, 4, 8, 16 377
OFDE_1 379
OFDE, 4, 8, 16 381
OFDI 383
OFDI_1 385
OFDT_1 387
OFDT, 4, 8, 16 389
OFDX_1 391
OFDX, 4, 8, 16 393
OFDXI 395
OFDXI_1 397
OR2-9 399
OR12, 16 401
PULLDOWN 403

PULLUP 405
RAM16X1D 407
RAM16X1D_1 409
RAM16X1S 411
RAM16X1S_1 413
RAM16X2S 415
RAM16X4S 417
RAM32X1S 419
RAM32X1S_1 421
RAM32X2S 423
RAM64X1S 425
RAM64X1S_1 427
ROM16X1 429
ROM32X1 431
ROM64X1 433
ROM128X1 435
ROM256X1 437
SOP3-4 439
SR4CE, SR8CE, SR16CE 441
SR4CLE, SR8CLE, SR16CLE 443
SR4CLED, SR8CLED, SR16CLED 445
SR4RE, SR8RE, SR16RE 449
SR4RLE, SR8RLE, SR16RLE 451
SR4RLED, SR8RLED, SR16RLED 453
SRL16 457
SRL16_1 459
SRL16E 461
SRL16E_1 463
SRLC16 465
SRLC16_1 467
SRLC16E 469
SRLC16E_1 471
STARTUP_SPARTAN3E 473
VCC 475
XNOR2-9 477
XOR2-9 479
XORCY 481
XORCY_D 483
XORCY_L 485

Functional Categories

This section categorizes, by function, the design elements that are described in detail later in this guide. The elements (primitive and macro implementations) are listed in alphanumeric order under each functional category.

Arithmetic Functions
Clock Components
Comparators
Config / BSCAN
Counters
Decoders
General
I / O Components
Internal Buffers
Logic Gates
MUXes
RAM / ROM
Registers & Latches
Shift Registers
Slice / CLB Primitives

Arithmetic Functions

Design Element	Description
ACC4	Macro : 4-Bit Loadable Cascadable Accumulator with Carry-In, Carry-Out, and Synchronous Reset
ACC8	Macro : 8-Bit Loadable Cascadable Accumulator with Carry-In, Carry-Out, and Synchronous Reset
ACC16	Macro : 16-Bit Loadable Cascadable Accumulator with Carry-In, Carry-Out, and Synchronous Reset
ADD4	Macro : 4-Bit Cascadable Full Adder with Carry-In, Carry-Out, and Overflow
ADD8	Macro : 8-Bit Cascadable Full Adder with Carry-In, Carry-Out, and Overflow
ADD16	Macro : 16-Bit Cascadable Full Adder with Carry-In, Carry-Out, and Overflow
ADSU4	Macro : 4-Bit Cascadable Adder / Subtracter with Carry-In, Carry-Out, and Overflow
ADSU8	Macro : 8-Bit Cascadable Adder / Subtracter with Carry-In, Carry-Out, and Overflow
ADSU16	Macro : 16-Bit Cascadable Adder / Subtracter with Carry-In, Carry-Out, and Overflow
MULT18X18SIO	Primitive: 18x18 Cascadable Signed Multiplier with Optional Input and Output registers, Clock Enable, and Synchronous Reset

Clock Components

Design Element	Description
BUFG	Primitive : Global Clock Buffer
BUFGCE	Primitive : Global Clock MUX with Clock Enable and Output State 0
BUFGCE_1	Primitive : Global Clock MUX Buffer with Clock Enable and Output State 1
BUFGMUX	Primitive : Global Clock MUX Buffer with Output State 0
BUFGMUX_1	Primitive : Global Clock MUX with Output State 1
DCM	Primitive: Digital Clock Manager

Comparators

Design Element	Description
COMP2	Macro : 2-Bit Identity Comparator
COMP4	Macro : 4-Bit Identity Comparator

Design Element	Description
COMP8	Macro : 8-Bit Identity Comparator
COMP16	Macro : 16-Bit Identity Comparator
COMPM2	Macro : 2-Bit Magnitude Comparator
COMPM4	Macro : 4-Bit Magnitude Comparator
COMPM8	Macro : 8-Bit Magnitude Comparator
COMPM16	Macro : 16-Bit Magnitude Comparator
COMPMC8	Macro : 8-Bit Magnitude Comparator
COMPMC16	Macro : 16-Bit Magnitude Comparator

Config/BSCAN

Design Element	Description
BSCAN_SPARTAN3	Primitive : Spartan-3 Boundary Scan Logic Control Circuit
CAPTURE_SPARTAN3	Primitive: Spartan-3 Register State Capture for Bitstream Readback
STARTUP_SPARTAN3E	Primitive: Spartan-3E User Interface to the GSR, GTS, Configuration Startup Sequence and Multi-Boot Trigger Circuitry

Counters

Design Element	Description
CB2CE	Macro : 2-Bit Cascadable Binary Counter with Clock Enable and Asynchronous Clear
CB4CE	Macro : 4-Bit Cascadable Binary Counter with Clock Enable and Asynchronous Clear
CB8CE	Macro : 8-Bit Cascadable Binary Counter with Clock Enable and Asynchronous Clear
CB16CE	Macro : 16-Bit Cascadable Binary Counter with Clock Enable and Asynchronous Clear
CB2RE	Macro : 2-Bit Cascadable Binary Counter with Clock Enable and Synchronous Reset
CB4RE	Macro : 4-Bit Cascadable Binary Counter with Clock Enable and Synchronous Reset
CB8RE	Macro : 8-Bit Cascadable Binary Counter with Clock Enable and Synchronous Reset
CB16RE	Macro : 16-Bit Cascadable Binary Counter with Clock Enable and Synchronous Reset
CB2RLE	Macro : 2-Bit Loadable Cascadable Binary Counter with Clock Enable and Synchronous Reset
CB4RLE	Macro : 4-Bit Loadable Cascadable Binary Counter with Clock Enable and Synchronous Reset
CB8RLE	Macro : 8-Bit Loadable Cascadable Binary Counter with Clock Enable and Synchronous Reset
CB16RLE	Macro : 16-Bit Loadable Cascadable Binary Counter with Clock Enable and Synchronous Reset
CC8CE	Macro : 8-Bit Cascadable Binary Counter with Clock Enable and Asynchronous Clear
CC16CE	Macro : 16-Bit Cascadable Binary Counter with Clock Enable and Asynchronous Clear
CC8CLE	Macro : 8-Bit Loadable Cascadable Binary Counter with Clock Enable and Asynchronous Clear
CC16CLE	Macro : 16-Bit Loadable Cascadable Binary Counter with Clock Enable and Asynchronous Clear
CC8CLED	Macro : 8-Bit Loadable Cascadable Bidirectional Binary Counter with Clock Enable and Asynchronous Clear
CC16CLED	Macro : 16-Bit Loadable Cascadable Bidirectional Binary Counter with Clock Enable and Asynchronous Clear
CC8RE	Macro : 8-Bit Cascadable Binary Counter with Clock Enable and Synchronous Reset
CC16RE	Macro : 16-Bit Cascadable Binary Counter with Clock Enable and Synchronous Reset
CD4CE	Macro : 4-Bit Cascadable BCD Counter with Clock Enable and Asynchronous Clear
CD4CLE	Macro : 4-Bit Loadable Cascadable BCD Counter with Clock Enable and Asynchronous Clear
CD4RE	Macro : 4-Bit Cascadable BCD Counter with Clock Enable and Synchronous Reset
CD4RLE	Macro : 4-Bit Loadable Cascadable BCD Counter with Clock Enable and Synchronous Reset
CJ4CE	Macro : 4-Bit Johnson Counter with Clock Enable and Asynchronous Clear
CJ5CE	Macro : 5-Bit Johnson Counter with Clock Enable and Asynchronous Clear
CJ8CE	Macro : 8-Bit Johnson Counter with Clock Enable and Asynchronous Clear
CJ4RE	Macro : 4-Bit Johnson Counter with Clock Enable and Synchronous Reset

Design Element	Description
CJ5RE	Macro : 5-Bit Johnson Counter with Clock Enable and Synchronous Reset
CJ8RE	Macro : 8-Bit Johnson Counter with Clock Enable and Synchronous Reset
CR8CE	Macro : 8-Bit Negative-Edge Binary Ripple Counter with Clock Enable and Asynchronous Clear
CR16CE	Macro : 16-Bit Negative-Edge Binary Ripple Counter with Clock Enable and Asynchronous Clear

Decoders

Design Element	Description
D2_4E	Macro : 2- to 4-Line Decoder / Demultiplexer with Enable
D3_8E	Macro : 3- to 8-Line Decoder / Demultiplexer with Enable
D4_16E	Macro : 4- to 16-Line Decoder / Demultiplexer with Enable
DEC_CC4	Macro : 4-Bit Active Low Decoder
DEC_CC8	Macro : 8-Bit Active Low Decoder
DEC_CC16	Macro : 16-Bit Active Low Decoder
DECODE4	Macro : 4-Bit Active-Low Decoder
DECODE8	Macro : 8-Bit Active-Low Decoder
DECODE16	Macro : 16-Bit Active-Low Decoder
DECODE32	Macro : 32-Bit Active-Low Decoder
DECODE64	Macro : 64-Bit Active-Low Decoder

General

Design Element	Description
GND	Primitive : Ground-Connection Signal Tag
VCC	Primitive: Vcc-Connection Signal Tag

I/O Components

Design Element	Description
IBUF	Primitive: Single-Ended Input Buffer
IBUF4	Macro: Multiple-Input Buffer
IIBUF8	Macro: Multiple-Input Buffer
IBUF16	Macro: Multiple-Input Buffer
IBUFDS	Primitive : Differential Signaling Input Buffer with Selectable I/O Interface
IBUFG	Primitive : Dedicated Input Buffer with Selectable I/O Interface
IBUFGDS	Primitive : Dedicated Differential Signaling Input Buffer with Selectable I/O Interface
IDDR2	Primitive: Double Data Rate Input D Flip-Flop with Optional Data Alignment, Clock Enable and Programmable Synchronous or Asynchronous Set/Reset
IOBUF	Primitive : Bi-Directional Buffer with Selectable I/0 Interface (multiple primitives)
IOBUFDS	Primitive : 3-State Differential Signaling I/O Buffer with Active Low Output Enable
KEEPER	Primitive : KEEPER Symbol
OBUF	Primitive : Single- and Multiple-Output Buffer
OBUF4	Macro : Multiple-Output Buffer
OBUF8	Macro : Multiple-Output Buffer
OBUF16	Macro : Multiple-Output Buffer
OBUFDS	Primitive : Differential Signaling Output Buffer with Selectable I/O Interface
OBUFT	Primitive : Single and Multiple 3-State Output Buffer with Active Low Output Enable

Design Element	Description
OBUFT4	Macro : Single and Multiple 3-State Output Buffer with Active Low Output Enable
OBUFT8	Macro : Single and Multiple 3-State Output Buffer with Active Low Output Enable
OBUFT16	Macro : Single and Multiple 3-State Output Buffer with Active Low Output Enable
OBUFTDS	Primitive : 3-State Output Buffer with Differential Signaling, Active-Low Output Enable, and Selectable I/O Interface
ODDR2	Primitive: Dual Data Rate Output D Flip-Flop with Optional Data Alignment, Clock Enable and Programmable Synchronous or Asynchronous Set/Reset
PULLDOWN	Primitive : Resistor to GND for Input Pads
PULLUP	Primitive : Resistor to VCC for Input PADs, Open-Drain, and 3-State Outputs

Internal Buffers

Design Element	Description
BUF	Primitive : General Purpose Buffer

Logic Gates

Design Element	Description
AND2	Primitive : 2-Input AND Gate with Inverted and Non-Inverted Inputs
AND3	Primitive : 3-Input AND Gate with Inverted and Non-Inverted Inputs
AND4	Primitive : 4-Input AND Gate with Inverted and Non-Inverted Inputs
AND5	Primitive: 5-Input AND Gate with Inverted and Non-Inverted Inputs
AND6	Macro : 6-Input AND Gate with Inverted and Non-Inverted Inputs
AND7	Macro : 7-Input AND Gate with Inverted and Non-Inverted Inputs
AND8	Macro : 8-Input AND Gate with Inverted and Non-Inverted Inputs
AND9	Macro : 9-Input AND Gate with Inverted and Non-Inverted Inputs
AND12	Macro : 12- Input AND Gate with Non-Inverted Inputs
AND16	Macro : 16- Input AND Gate with Non-Inverted Inputs
INV	Primitive : Single and Multiple Inverters
INV4	Macro : Single and Multiple Inverters
INV8	Macro : Single and Multiple Inverters
INV16	Macro : Single and Multiple Inverters
NAND2	Primitive :2-Input NAND Gate with Inverted and Non-Inverted Inputs
NAND3	Primitive : 3-Input NAND Gate with Inverted and Non-Inverted Inputs
NAND3B1	Primitive :3-Input NAND Gate with Inverted and Non-Inverted Inputs
NAND3B2	Primitive : 3-Input NAND Gate with Inverted and Non-Inverted Inputs
NAND3B3	Primitive : 3-Input NAND Gate with Inverted and Non-Inverted Inputs
NAND4	Primitive : 4-Input NAND Gate with Inverted and Non-Inverted Inputs
NAND5	Macro: 5-Input NAND Gate with Inverted and Non-Inverted Inputs
NAND6	Macro : 6-Input NAND Gate with Inverted and Non-Inverted Inputs.
NAND7	Macro : 7-Input NAND Gate with Inverted and Non-Inverted Inputs.
NAND8	Macro : 8-Input NAND Gate with Inverted and Non-Inverted Inputs.
NAND9	Macro : 9-Input NAND Gate with Inverted and Non-Inverted Inputs.
NAND12	Macro : 12- Input NAND Gate with Non-Inverted Inputs.
NAND16	Macro : 16- Input NAND Gate with Non-Inverted Inputs.
NOR2	Primitive : 2- Input NOR Gate with Inverted and Non-Inverted Inputs.
NOR3	Primitive : 3- Input NOR Gate with Inverted and Non-Inverted Inputs
NOR4	Primitive : 4- Input NOR Gate with Inverted and Non-Inverted Inputs
NOR6	Macro : 6- Input NOR Gate with Inverted and Non-Inverted Inputs

Design Element	Description
NOR7	Macro : 7- Input NOR Gate with Inverted and Non-Inverted Inputs
NOR8	Macro : 8- Input NOR Gate with Inverted and Non-Inverted Inputs
NOR9	Macro : 9- Input NOR Gate with Inverted and Non-Inverted Inputs
NOR12	Macro : 12-Input NOR Gate with Non-Inverted Inputs
NOR16	Macro : 16-Input NOR Gate with Non-Inverted Inputs
OR2	Primitive : 2-Input OR Gate with Inverted and Non-Inverted Inputs
OR3	Primitive : 3-Input OR Gate with Inverted and Non-Inverted Inputs
OR4	Primitive : 4-Input OR Gate with Inverted and Non-Inverted Inputs
OR6	Macro : 6-Input OR Gate with Inverted and Non-Inverted Inputs
OR7	Macro : 6-Input OR Gate with Inverted and Non-Inverted Inputs
OR8	Macro : 8-Input OR Gate with Inverted and Non-Inverted Inputs
OR9	Macro : 9-Input OR Gate with Inverted and Non-Inverted Inputs
OR12	Macro : 12-Input OR Gate with Inverted and Non-Inverted Inputs
OR16	Macro : 16-Input OR Gate with Inverted and Non-Inverted Inputs
SOP3	Macro : Sum of Products
SOP3B1A	Macro : Sum of Products
SOP3B1B	Macro : Sum of Products
SOP3B2A	Macro : Sum of Products
SOP3B2B	Macro : Sum of Products
SOP3B3	Macro : Sum of Products
SOP4	Macro : Sum of Products
SOP4B1	Macro : Sum of Products
SOP4B2A	Macro : Sum of Products
SOP4B2B	Macro : Sum of Products
SOP4B3	Macro : Sum of Products
SOP4B4	Macro : Sum of Products
XNOR2	Primitive : 2-Input XNOR Gate with Non-Inverted Inputs
XNOR3	Primitive : 3-Input XNOR Gate with Non-Inverted Inputs
XNOR4	Primitive : 4-Input XNOR Gate with Non-Inverted Inputs
XNOR9	Macro : 9-Input XNOR Gate with Non-Inverted Inputs
XOR2	Macro : 2-Input XOR Gate with Non-Inverted Inputs
XOR3	Macro : 3-Input XOR Gate with Non-Inverted Inputs
XOR4	Primitive : 4-Input XOR Gate with Non-Inverted Inputs
XOR6	Macro : 6-Input XOR Gate with Non-Inverted Inputs
XOR7	Macro : 7-Input XOR Gate with Non-Inverted Inputs
XOR9	Macro : 9-Input XOR Gate with Non-Inverted Inputs

MUXes

Design Element	Description
M2_1	Macro : 2-to-1 Multiplexer
M2_1B1	Macro : 2-to-1 Multiplexer with D0 Inverted
M2_1B2	Macro : 2-to-1 Multiplexer with D0 and D1 Inverted
M2_1E	Macro : 2-to-1 Multiplexer with Enable
M4_1E	Macro : 4-to-1 Multiplexer with Enable
M8_1E	Macro : 8-to-1 Multiplexer with Enable
M16_1E	Macro : 16-to-1 Multiplexer with Enable

RAM/ROM

Design Element	Description
RAM16X1D	Primitive : 16-Deep by 1-Wide Static Dual Port Synchronous RAM
RAM16X1D_1	Primitive : 16-Deep by 1-Wide Static Dual Port Synchronous RAM with Negative-Edge Clock
RAM16X1S	Primitive : 16-Deep by 1-Wide Static Synchronous RAM
RAM16X1S_1	Primitive : 16-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock
RAM16X2S	Macro : 16-Deep by 2-Wide Static Synchronous RAM
RAM16X4S	Macro : 16-Deep by 4-Wide Static Synchronous RAM
RAM32X1S	Primitive: 32-Deep by 1-Wide Static Synchronous RAM
RAM32X1S_1	Primitive: 32-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock
RAM32X2S	Primitive: 32-Deep by 2-Wide Static Synchronous RAM
RAM64X1S	Primitive: 64-Deep by 1-Wide Static Synchronous RAM
RAM64X1S_1	Primitive: 64-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock
ROM16X1	Primitive: 16-Deep by 1-Wide ROM
ROM32X1	Primitive: 32-Deep by 1-Wide ROM
ROM64X1	Primitive: 64-Deep by 1-Wide ROM
ROM128X1	Primitive: 128-Deep by 1-Wide ROM
ROM256X1	Primitive: 256-Deep by 1-Wide ROM

Registers & Latches

Design Element	Description
FD	Primitive: D Flip-Flop
FD_1	Primitive: D Flip-Flop with Negative-Edge Clock
FD4CE	Macro: 4-Bit Data Register with Clock Enable and Asynchronous Clear
FD8CE	Macro: 8-Bit Data Register with Clock Enable and Asynchronous Clear
FD16CE	Macro: 16-Bit Data Register with Clock Enable and Asynchronous Clear
FD4RE	Macro: 4-Bit Data Register with Clock Enable and Synchronous Reset
FD8RE	Macro: 8-Bit Data Register with Clock Enable and Synchronous Reset
FD16RE	Macro: 16-Bit Data Register with Clock Enable and Synchronous Reset
FDC	Primitive : D Flip-Flop with Asynchronous Clear
FDC_1	Primitive : D Flip-Flop with Negative-Edge Clock and Asynchronous Clear
FDCE	Primitive : D Flip-Flop with Clock Enable and Asynchronous Clear
FDCE_1	Primitive : D Flip-Flop with Negative-Edge Clock, Clock Enable, and Asynchronous Clear
FDCP	Primitive : D Flip-Flop with Asynchronous Preset and Clear
FDCP_1	Primitive : D Flip-Flop with Negative-Edge Clock and Asynchronous Preset and Clear
FDCPE	Primitive : D Flip-Flop with Clock Enable and Asynchronous Preset and Clear
FDCPE_1	Primitive : D Flip-Flop with Negative-Edge Clock, Clock Enable, and Asynchronous Preset and Clear
FDE	Primitive : D Flip-Flop with Clock Enable
FDE_1	Primitive : D Flip-Flop with Negative-Edge Clock and Clock Enable
FDP	Primitive : D Flip-Flop with Asynchronous Preset
FDP_1	Primitive : D Flip-Flop with Negative-Edge Clock and Asynchronous Preset
FDPE	Primitive : D Flip-Flop with Clock Enable and Asynchronous Preset
FDPE_1	Primitive : D Flip-Flop with Negative-Edge Clock, Clock Enable, and Asynchronous Preset
FDR	Primitive : D Flip-Flop with Synchronous Reset
FDR_1	Primitive : D Flip-Flop with Negative-Edge Clock and Synchronous Reset
FDRE	Primitive : D Flip-Flop with Clock Enable and Synchronous Reset
FDRE_1	Primitive : D Flip-Flop with Negative-Clock Edge, Clock Enable, and Synchronous Reset

Design Element	Description
FDRS	Primitive : D Flip-Flop with Synchronous Reset and Set
FDRS_1	Primitive : D Flip-Flop with Negative-Clock Edge and Synchronous Reset and Set
FDRSE	Primitive : D Flip-Flop with Synchronous Reset and Set and Clock Enable
FDRSE_1	Primitive : D Flip-Flop with Negative-Clock Edge, Synchronous Reset and Set, and Clock Enable
FDS	Primitive : D Flip-Flop with Synchronous Set
FDS_1	Primitive : D Flip-Flop with Negative-Edge Clock and Synchronous Set
FDSE	Primitive : D Flip-Flop with Clock Enable and Synchronous Set
FDSE_1	Primitive : D Flip-Flop with Negative-Edge Clock, Clock Enable, and Synchronous Set
FJKC	Macro : J-K Flip-Flop with Asynchronous Clear
FJKCE	Macro : J-K Flip-Flop with Clock Enable and Asynchronous Clear
FJKP	Macro : J-K Flip-Flop with Asynchronous Preset
FJKPE	Macro : J-K Flip-Flop with Clock Enable and Asynchronous Preset
FJKRSE	Macro : J-K Flip-Flop with Clock Enable and Synchronous Reset and Set
FJKSRE	Macro : J-K Flip-Flop with Clock Enable and Synchronous Set and Reset
FTC	Macro : Toggle Flip-Flop with Toggle Enable and Asynchronous Clear
FTCE	Macro : Toggle Flip-Flop with Toggle and Clock Enable and Asynchronous Clear
FTCLE	Macro : Toggle / Loadable Flip-Flop with Toggle and Clock Enable and Asynchronous Clear
FTCLEX	Macro : Toggle / Loadable Flip-Flop with Toggle and Clock Enable and Asynchronous Clear
FTP	Macro : Toggle Flip-Flop with Toggle Enable and Asynchronous Preset
FTPE	Macro : Toggle Flip-Flop with Toggle and Clock Enable and Asynchronous Preset
FTPLE	Macro : Toggle / Loadable Flip-Flop with Toggle and Clock Enable and Asynchronous Preset
FTRSE	Macro : Toggle Flip-Flop with Toggle and Clock Enable and Synchronous Reset and Set
FTRSLE	Macro : Toggle / Loadable Flip-Flop with Toggle and Clock Enable and Synchronous Reset and Set
FTSRE	Macro : Toggle Flip-Flop with Toggle and Clock Enable and Synchronous Set and Reset
FTSRLE	Macro : Toggle / Loadable Flip-Flop with Toggle and Clock Enable and Synchronous Set and Reset
IFD	Macro : Single- and Multiple-Input D Flip-Flop
IFD_1	Macro : Input D Flip-Flop with Inverted Clock
IFD4	Macro : Single- and Multiple-Input D Flip-Flop
IFD8	Macro : Single- and Multiple-Input D Flip-Flop
IFD16	Macro : Single- and Multiple-Input D Flip-Flop
IFDI	Macro : Input D Flip-Flop (Asynchronous Preset)
IFDI_1	Macro : Input D Flip-Flop with Inverted Clock (Asynchronous Preset)
IFDX	Macro : Single- and Multiple-Input D Flip-Flop with Clock Enable
IFDX_1	Macro : Input D Flip-Flop with Inverted Clock and Clock Enable
IFDX4	Macro : Single- and Multiple-Input D Flip-Flop with Clock Enable
IFDX8	Macro : Single- and Multiple-Input D Flip-Flop with Clock Enable
IFDX16	Macro : Single- and Multiple-Input D Flip-Flops with Clock Enable
IFDXI	Macro : Input D Flip-Flop with Clock Enable (Asynchronous Preset)
ILDXI_1	Macro : Input D Flip-Flop with Inverted Clock and Clock Enable (Asynchronous Preset)
ILD	Macro : Transparent Input Data Latch
ILD_1	Macro : Transparent Input Data Latch with Inverted Gate
ILD4	Macro : Transparent Input Data Latch
ILD8	Macro : Transparent Input Data Latch
ILD16	Macro : Transparent Input Data Latch
ILDI	Macro : Transparent Input Data Latch (Asynchronous Preset)
ILDI_1	Macro : Transparent Input Data Latch with Inverted Gate (Asynchronous Preset)
ILDX	Macro : Transparent Input Data Latch

Design Element	Description
ILDX_1	Macro : Transparent Input Data Latch with Inverted Gate
ILDX4	Macro : Transparent Input Data Latch
ILDX8	Macro : Transparent Input Data Latch
ILDX16	Macro : Transparent Input Data Latch
ILDXI	Macro : Transparent Input Data Latch (Asynchronous Preset)
ILDXI_1	Macro : Transparent Input Data Latch with Inverted Gate (Asynchronous Preset)
LD	Primitive : Transparent Data Latch
LD_1	Primitive : Transparent Data Latch with Inverted Gate
LD4	Macro : Multiple Transparent Data Latch
LD8	Macro : Multiple Transparent Data Latch
LD16	Macro : Multiple Transparent Data Latch
LD4CE	Macro : Transparent Data Latch with Asynchronous Clear and Gate Enable
LD8CE	Macro : Transparent Data Latch with Asynchronous Clear and Gate Enable
LD16CE	Macro : Transparent Data Latch with Asynchronous Clear and Gate Enable
LDC	Primitive : Transparent Data Latch with Asynchronous Clear
LDC_1	Primitive : Transparent Data Latch with Asynchronous Clear and Inverted Gate
LDCE	Primitive : Transparent Data Latch with Asynchronous Clear and Gate Enable
LDCE_1	Primitive : Transparent Data Latch with Asynchronous Clear, Gate Enable, and Inverted Gate
LDCP	Primitive : Transparent Data Latch with Asynchronous Clear and Preset
LDCP_1	Primitive : Transparent Data Latch with Asynchronous Clear and Preset and Inverted Gate
LDCPE	Primitive : Transparent Data Latch with Asynchronous Clear and Preset and Gate Enable
LDCPE_1	Primitive : Transparent Data Latch with Asynchronous Clear and Preset, Gate Enable, and Inverted Gate
LDE	Primitive : Transparent Data Latch with Gate Enable
LDE_1	Primitive : Transparent Data Latch with Gate Enable and Inverted Gate
LDP	Primitive : Transparent Data Latch with Asynchronous Preset
LDP_1	Primitive : Transparent Data Latch with Asynchronous Preset and Inverted Gate
LDPE	Primitive : Transparent Data Latch with Asynchronous Preset and Gate Enable
LDPE_1	Primitive : Transparent Data Latch with Asynchronous Preset, Gate Enable, and Inverted Gate
OFD	Macro : Single- and Multiple-Output D Flip-Flops
OFD_1	Macro : Output D Flip-Flop with Inverted Clock
OFD4	Macro : Single- and Multiple-Output D Flip-Flops
OFD8	Macro : Single- and Multiple-Output D Flip-Flops
OFD16	Macro : Single- and Multiple-Output D Flip-Flops
OFDE	Macro : D Flip-Flop with Active-High Enable Output Buffers
OFDE_1	Macro : D Flip-Flop with Active-High Enable Output Buffer and Inverted Clock
OFDE4	Macro : D Flip-Flop with Active-High Enable Output Buffers
OFDE8	Macro : D Flip-Flop with Active-High Enable Output Buffers
OFDE16	Macro : D Flip-Flop with Active-High Enable Output Buffers
OFDI	Macro : Output D Flip-Flop (Asynchronous Preset)
OFDI_1	Macro : Output D Flip-Flop with Inverted Clock (Asynchronous Preset)
OFDT	Macro : Single and Multiple D Flip-Flop with Active-Low 3-State Output Buffers
OFDT_1	Macro : D Flip-Flop with Active-Low 3-State Output Buffer and Inverted Clock
OFDT4	Macro : Single and Multiple D Flip-Flop with Active-Low 3-State Output Buffers
OFDT8	Macro : Single and Multiple D Flip-Flop with Active-Low 3-State Output Buffers
OFDT16	Macro : Single and Multiple D Flip-Flop with Active-Low 3-State Output Buffers
OFDX	Macro : Single- and Multiple-Output D Flip-Flop with Clock Enable
OFDX_1	Macro : Output D Flip-Flop with Inverted Clock and Clock Enable

Design Element	Description
OFDX4	Macro : Single- and Multiple-Output D Flip-Flop with Clock Enable
OFDX8	Macro : Single- and Multiple-Output D Flip-Flop with Clock Enable
OFDX16	Macro : Single- and Multiple-Output D Flip-Flop with Clock Enable
OFDXI	Macro : Output D Flip-Flop with Clock Enable (Asynchronous Preset)
OFDXI_1	Macro : Output D Flip-Flop with Inverted Clock and Clock Enable (Asynchronous Preset)

Shift Registers

Design Element	Description
BRLSHFT4	Macro : 4-Bit Barrel Shifter
BRLSHFT8	Macro : 8-Bit Barrel Shifter
SR4CE	Macro : 4-Bit Serial-In Parallel-Out Shift Register with Clock Enable and Asynchronous Clear
SR8CE	Macro : 8-Bit Serial-In Parallel-Out Shift Register with Clock Enable and Asynchronous Clear
SR16CE	Macro : 16-Bit Serial-In Parallel-Out Shift Register with Clock Enable and Asynchronous Clear
SR4CLE	Macro : 4-Bit Loadable Serial / Parallel-In Parallel-Out Shift Register with Clock Enable and Asynchronous Clear
SR8CLE	Macro : 8-Bit Loadable Serial / Parallel-In Parallel-Out Shift Register with Clock Enable and Asynchronous Clear
SR16CLE	Macro : 16-Bit Loadable Serial / Parallel-In Parallel-Out Shift Register with Clock Enable and Asynchronous Clear
SR4CLED	Macro : 4-Bit Shift Register with Clock Enable and Asynchronous Clear
SR8CLED	Macro : 8-Bit Shift Register with Clock Enable and Asynchronous Clear
SR16CLED	Macro : 16-Bit Shift Register with Clock Enable and Asynchronous Clear
SR4RE	Macro : 4-Bit Serial-In Parallel-Out Shift Register with Clock Enable and Synchronous Reset
SR8RE	Macro : 8-Bit Serial-In Parallel-Out Shift Register with Clock Enable and Synchronous Reset
SR16RE	Macro : 16-Bit Serial-In Parallel-Out Shift Register with Clock Enable and Synchronous Reset
SR4RLE	Macro : 4-Bit Loadable Serial / Parallel-In Parallel-Out Shift Register with Clock Enable and Synchronous Reset
SR8RLE	Macro : 8-Bit Loadable Serial / Parallel-In Parallel-Out Shift Register with Clock Enable and Synchronous Reset
SR16RLE	Macro : 16-Bit Loadable Serial / Parallel-In Parallel-Out Shift Register with Clock Enable and Synchronous Reset
SR4RLED	Macro : 4-Bit Shift Register with Clock Enable and Synchronous Reset
SR8RLED	Macro : 8-Bit Shift Register with Clock Enable and Synchronous Reset
SR16RLED	Macro : 16-Bit Shift Register with Clock Enable and Synchronous Reset
SRL16	Primitive : 16-Bit Shift Register Look-Up-Table (LUT)
SRL16_1	Primitive : 16-Bit Shift Register Look-Up-Table (LUT) with Negative-Edge Clock
SRL16E	Primitive : 16-Bit Shift Register Look-Up-Table (LUT) with Clock Enable
SRL16E_1	Primitive : 16-Bit Shift Register Look-Up-Table (LUT) with Negative-Edge Clock and Clock Enable
SRLC16	Primitive : 16-Bit Shift Register Look-Up-Table (LUT) with Carry
SRLC16_1	Primitive : 16-Bit Shift Register Look-Up-Table (LUT) with Carry and Negative-Edge Clock
SRLC16E	Primitive : 16-Bit Shift Register Look-Up-Table (LUT) with Carry and Clock Enable
SRLC16E_1	Primitive : 16-Bit Shift Register Look-Up-Table (LUT) with Carry, Negative-Edge Clock, and Clock Enable

Slice/CLB Primitives

Design Element	Description
LUT1	Primitive : 1-Bit Look-Up-Table with General Output
LUT2	Primitive : 2-Bit Look-Up-Table with General Output
LUT3	Primitive : 3-Bit Look-Up-Table with General Output
LUT4	Primitive : 4-Bit Look-Up-Table with General Output
LUT1_D	Primitive : 1-Bit Look-Up-Table with Dual Output
LUT2_D	Primitive : 2-Bit Look-Up-Table with Dual Output

Design Element	Description
LUT3_D	Primitive : 3-Bit Look-Up-Table with Dual Output
LUT4_D	Primitive : 4-Bit Look-Up-Table with Dual Output
LUT1_L	Primitive : 1-Bit Look-Up-Table with Local Output
LUT2_L	Primitive : 2-Bit Look-Up-Table with Local Output
LUT3_L	Primitive : 3-Bit Look-Up-Table with Local Output
LUT4_L	Primitive : 4-Bit Look-Up-Table with Local Output
MULT_AND	Primitive : Fast Multiplier AND
MUXCY	Primitive : 2-to-1 Multiplexer for Carry Logic with General Output
MUXCY_D	Primitive : 2-to-1 Multiplexer for Carry Logic with Dual Output
MUXCY_L	Primitive : 2-to-1 Multiplexer for Carry Logic with Local Output
MUXF5	Primitive : 2-to-1 Lookup Table Multiplexer with General Output
MUXF5_D	Primitive : 2-to-1 Lookup Table Multiplexer with Dual Output
MUXF5_L	Primitive : 2-to-1 Lookup Table Multiplexer with Local Output
MUXF6	Primitive : 2-to-1 Lookup Table Multiplexer with General Output
MUXF6_D	Primitive : 2-to-1 Lookup Table Multiplexer with Dual Output
MUXF6_L	Primitive : 2-to-1 Lookup Table Multiplexer with Local Output
MUXF7	Primitive : 2-to-1 Lookup Table Multiplexer with General Output
MUXF7_D	Primitive : 2-to-1 Lookup Table Multiplexer with Dual Output
MUXF7_L	Primitive : 2-to-1 Lookup Table Multiplexer with Local Output
MUXF8	Primitive : 2-to-1 Lookup Table Multiplexer with General Output
MUXF8_D	Primitive : 2-to-1 Lookup Table Multiplexer with Dual Output
MUXF8_L	Primitive : 2-to-1 Lookup Table Multiplexer with Local Output
XORCY	Primitive : XOR for Carry Logic with General Output
XORCY_D	Primitive : XOR for Carry Logic with Dual Output
XORCY_L	Primitive : XOR for Carry Logic with Local Output

About the Spartan-3E Design Elements

The remaining sections in this book describe each primitive design element that can be used under the Spartan-3E architecture.

The design elements are organized in alphanumeric order, with all numeric suffixes in ascending order. For example, FDCPE precedes FDRSE, and IBUF precedes IBUFDS.

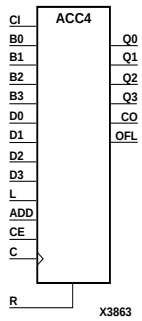
The following information is provided for each library element, where applicable

- Name of each element
- Description of each element, including truth tables, where applicable
- A description of the attributes associated with each design element, where appropriate.
- Referrals to additional sources of information.

Designers who prefer to work with hardware description language (HDL) are encouraged to consult the *Spartan-3E Libraries Guide for HDL Designs*.

ACC4, 8, 16

Macro: 4-, 8-, 16-Bit Loadable Cascadable Accumulators with Carry-In, Carry-Out, and Synchronous Reset



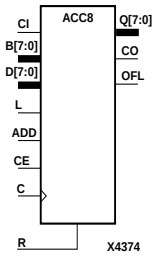
ACC4, ACC8, ACC16 can add or subtract a 4-, 8-, 16-bit unsigned-binary, respectively or twos-complement word to or from the contents of a 4-, 8-, 16-bit data register and store the results in the register. The register can be loaded with the 4-, 8-, 16-bit word.

The synchronous reset (R) has priority over all other inputs, and when High, causes all outputs to go to logic level zero during the Low-to-High clock (C) transition. Clock (C) transitions are ignored when clock enable (CE) is Low.

The accumulator is asynchronously cleared, outputs Low, when power is applied.

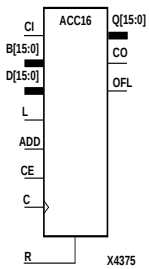
Spartan-3E simulates power-on when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.



Load

When the load input (L) is High, CE is ignored and the data on the D inputs is loaded into the register during the Low-to-High clock (C) transition. ACC4 loads the data on inputs D3 – D0 into the 4-bit register. ACC8 loads the data on D7 – D0 into the 8-bit register. ACC16 loads the data on inputs D15 – D0 into the 16-bit register.



Unsigned Binary Versus Twos Complement

ACC4, ACC8, ACC16 can operate, respectively, on either 4-, 8-, 16-bit unsigned binary numbers or 4-, 8-, 16-bit twos-complement numbers. If the inputs are interpreted as unsigned binary, the result can be interpreted as unsigned binary. If the inputs are interpreted as twos complement, the output can be interpreted as twos complement. The only functional difference between an unsigned binary operation and a twos-complement operation is how they determine when “overflow” occurs. Unsigned binary uses CO, while twos complement uses OFL to determine when “overflow” occurs.

Unsigned Binary Operation

For unsigned binary operation, ACC4 can represent numbers between 0 and 15, inclusive; ACC8 between 0 and 255, inclusive; and ACC16 between 0 and 65535, inclusive. In add mode, CO is active (High) when the sum exceeds the bounds of the adder/subtracter. In subtract mode, CO is an active-Low borrow-out and goes Low when the difference exceeds the bounds. The carry-out (CO) is not registered synchronously with the data outputs. CO always reflects the accumulation of the B inputs (B3 – B0 for ACC4, B7 – B0 for ACC8, B15 – B0 for ACC16) and the contents of the register. This allows cascading of ACC4s, ACC8s, or ACC16s by connecting CO of one stage to CI of the next stage. An unsigned binary “overflow” that is always active-High can be generated by gating the ADD signal and CO as follows.

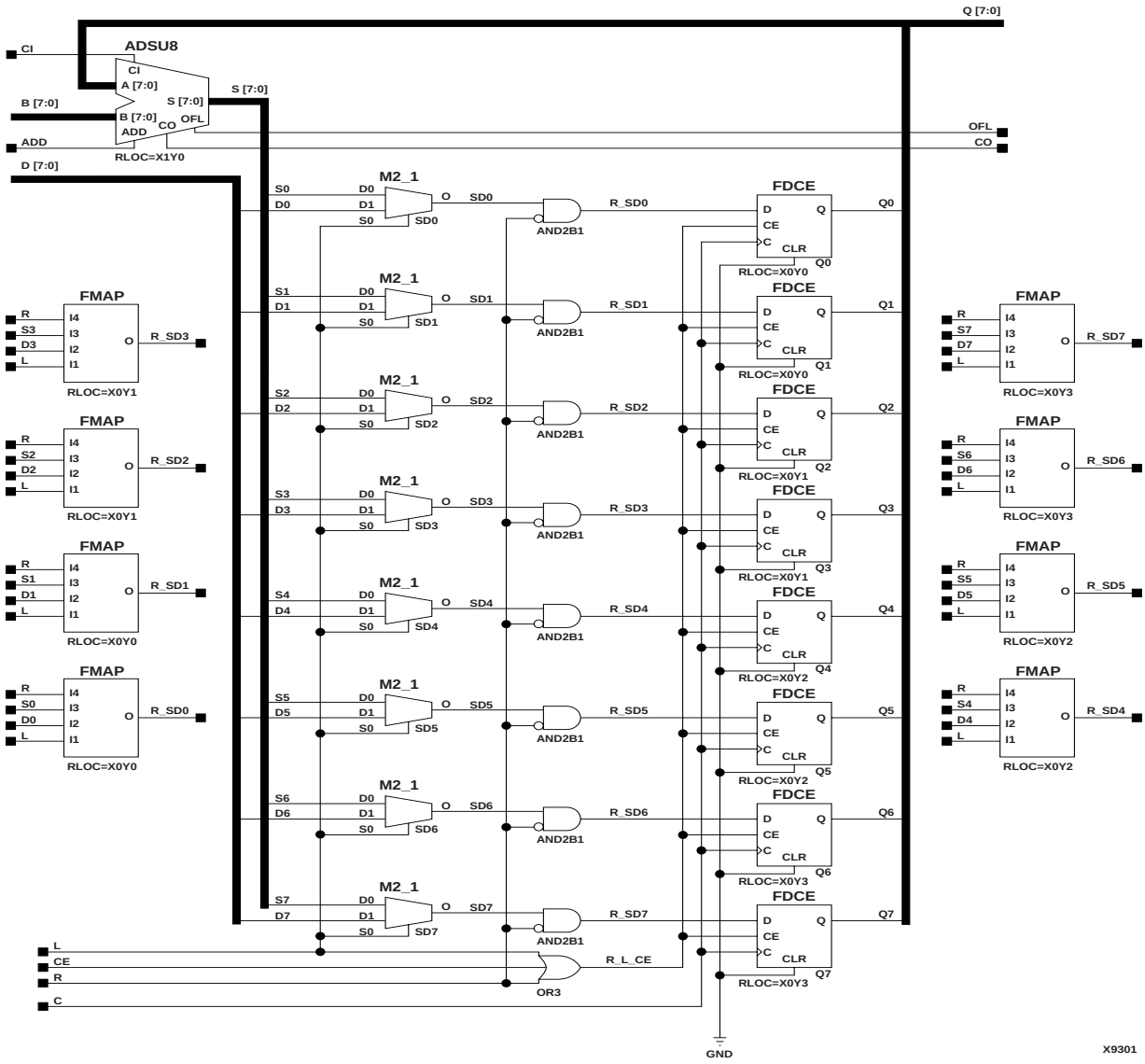
```
unsigned overflow = CO XOR ADD
```

Ignore OFL in unsigned binary operation.

Twos-Complement Operation

For twos-complement operation, ACC4 can represent numbers between -8 and +7, inclusive; ACC8 between -128 and +127, inclusive; ACC16 between -32768 and +32767, inclusive. If an addition or subtraction operation result exceeds this range, the OFL output goes High. The overflow (OFL) is not registered synchronously with the data outputs. OFL always reflects the accumulation of the B inputs (B3 – B0 for ACC4, B7 – B0 for ACC8, B15 – B0 for ACC16) and the contents of the register, which allows cascading of ACC4s, ACC8s, or ACC16s by connecting OFL of one stage to CI of the next stage.

Ignore CO in twos-complement operation.



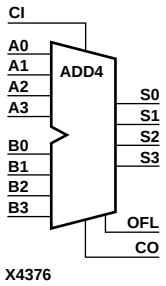
ACC8 Implementation for Spartan-3E

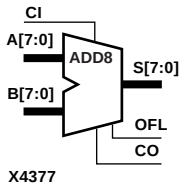
Usage

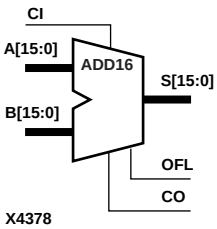
ACC is intended for schematics only.

ADD4, 8, 16

Macro: 4-, 8-, 16-Bit Cascadable Full Adders with Carry-In, Carry-Out, and Overflow







ADD4, ADD8, and ADD16 add two words and a carry-in (CI), producing a sum output and carry-out (CO) or overflow (OFL). ADD4 adds A3 – A0, B3 – B0, and CI producing the sum output S3 – S0 and CO (or OFL). ADD8 adds A7 – A0, B7 – B0, and CI, producing the sum output S7 – S0 and CO (or OFL). ADD16 adds A15 – A0, B15 – B0 and CI, producing the sum output S15 – S0 and CO (or OFL).

Unsigned Binary Versus Twos Complement

ADD4, ADD8, ADD16 can operate on either 4-, 8-, 16-bit unsigned binary numbers or 4-, 8-, 16-bit twos-complement numbers, respectively. If the inputs are interpreted as unsigned binary, the result can be interpreted as unsigned binary. If the inputs are interpreted as twos complement, the output can be interpreted as twos complement. The only functional difference between an unsigned binary operation and a twos-complement operation is how they determine when “overflow” occurs. Unsigned binary uses CO, while twos-complement uses OFL to determine when “overflow” occurs. To interpret the inputs as unsigned binary, follow the CO output. To interpret the inputs as twos complement, follow the OFL output.

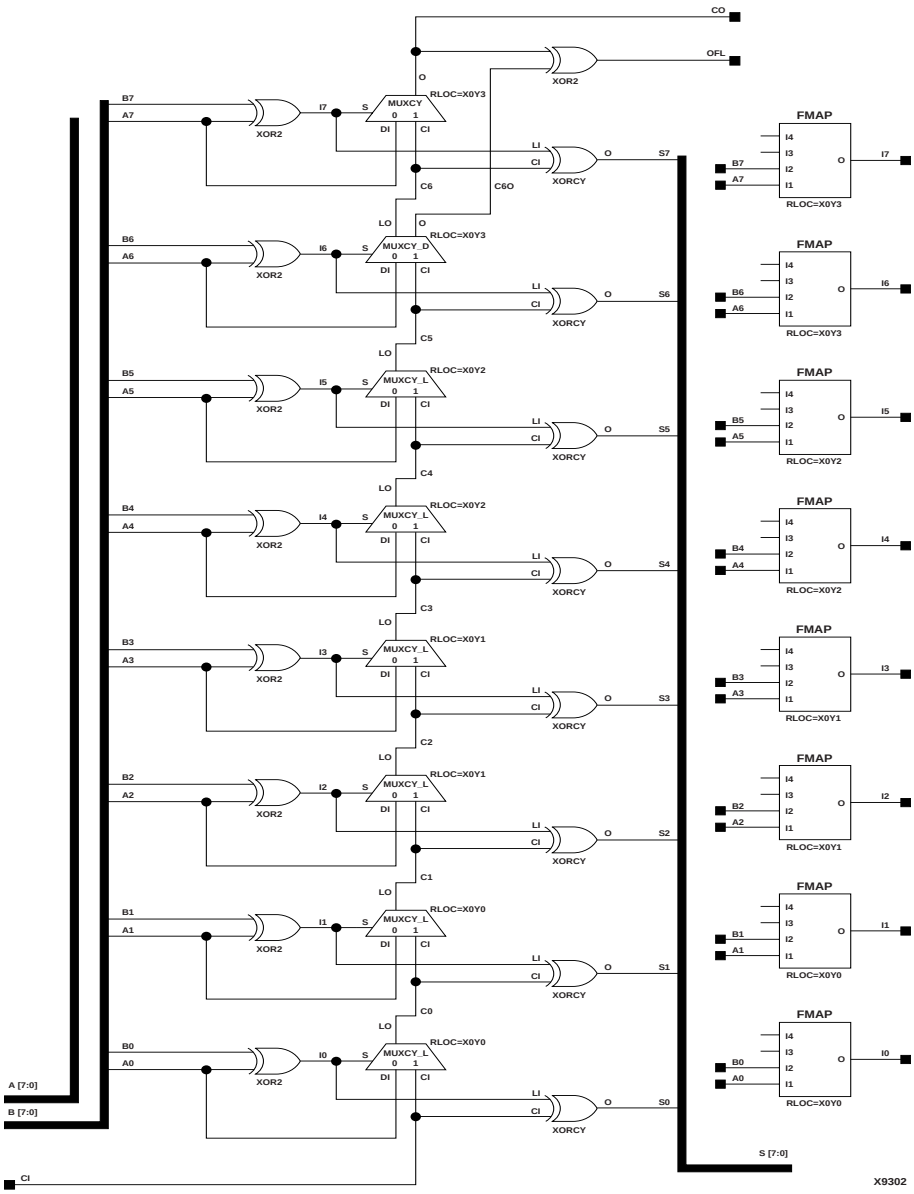
Unsigned Binary Operation

For unsigned binary operation, ADD4 can represent numbers between 0 and 15, inclusive; ADD8 between 0 and 255, inclusive; ADD16 between 0 and 65535, inclusive. CO is active (High) when the sum exceeds the bounds of the adder.

OFL is ignored in unsigned binary operation.

Twos-Complement Operation

For twos-complement operation, ADD4 can represent numbers between -8 and +7, inclusive; ADD8 between -128 and +127, inclusive; ADD16 between -32768 and +32767, inclusive. OFL is active (High) when the sum exceeds the bounds of the adder. CO is ignored in twos-complement operation.



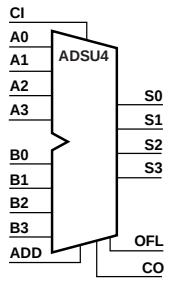
ADD8 Implementation Spartan-3E

Usage

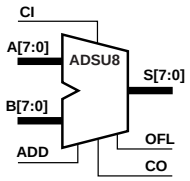
This design element is a schematic only.

ADSU4, 8, 16

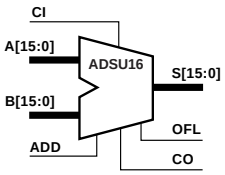
Macro: 4-, 8-, 16-Bit Cascadable Adders/Subtracters with Carry-In, Carry-Out, and Overflow



X4379



X4380



X4381

When the ADD input is High, ADSU4, ADSU8, and ADSU16 add two words and a carry-in (CI), producing a sum output and carry-out (CO) or overflow (OFL). ADSU4 adds two 4-bit words (A3 – A0 and B3 – B0) and a CI, producing a 4-bit sum output (S3 – S0) and CO or OFL. ADSU8 adds two 8-bit words (A7 – A0 and B7 – B0) and a CI producing, an 8-bit sum output (S7 – S0) and CO or OFL. ADSU16 adds two 16-bit words (A15 – A0 and B15 – B0) and a CI, producing a 16-bit sum output (S15 – S0) and CO or OFL.

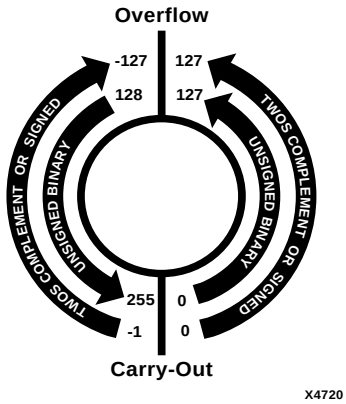
When the ADD input is Low, ADSU4, ADSU8, and ADSU16 subtract Bz – B0 from Az– A0, producing a difference output and CO or OFL. ADSU4 subtracts B3 – B0 from A3 – A0, producing a 4-bit difference (S3 – S0) and CO or OFL. ADSU8 subtracts B7 – B0 from A7 – A0, producing an 8-bit difference (S7 – S0) and CO or OFL. ADSU16 subtracts B15 – B0 from A15 – A0, producing a 16-bit difference (S15 – S0) and CO or OFL.

In add mode, CO and CI are active-High. In subtract mode, CO and CI are active-Low. OFL is active-High in add and subtract modes.

Unsigned Binary Versus Twos Complement

ADSU4, ADSU8, ADSU16 can operate, respectively, on either 4-, 8-, 16-bit unsigned binary numbers or 4-, 8-, 16-bit twos-complement numbers. If the inputs are interpreted as unsigned binary, the result can be interpreted as unsigned binary. If the inputs are interpreted as twos complement, the output can be interpreted as twos complement. The only functional difference between an unsigned binary operation and a twos-complement operation is how they determine when “overflow” occurs. Unsigned binary uses CO, while twos complement uses OFL to determine when “overflow” occurs.

With adder/subtracters, either unsigned binary or twos-complement operations cause an overflow. If the result crosses the overflow boundary, an overflow is generated. Similarly, when the result crosses the carry-out boundary, a carry-out is generated. The following figure shows the ADSU carry-out and overflow boundaries.



ADSU Carry-Out and Overflow Boundaries

Unsigned Binary Operation

For unsigned binary operation, ADSU4 can represent numbers between 0 and 15, inclusive; ADSU8 between 0 and 255, inclusive; ADSU16 between 0 and 65535, inclusive. In add mode, CO is active (High) when the sum exceeds the bounds of the adder/subtracter. In subtract mode, CO is an active-Low borrow-out and goes Low when the difference exceeds the bounds.

An unsigned binary "overflow" that is always active-High can be generated by gating the ADD signal and CO as follows.

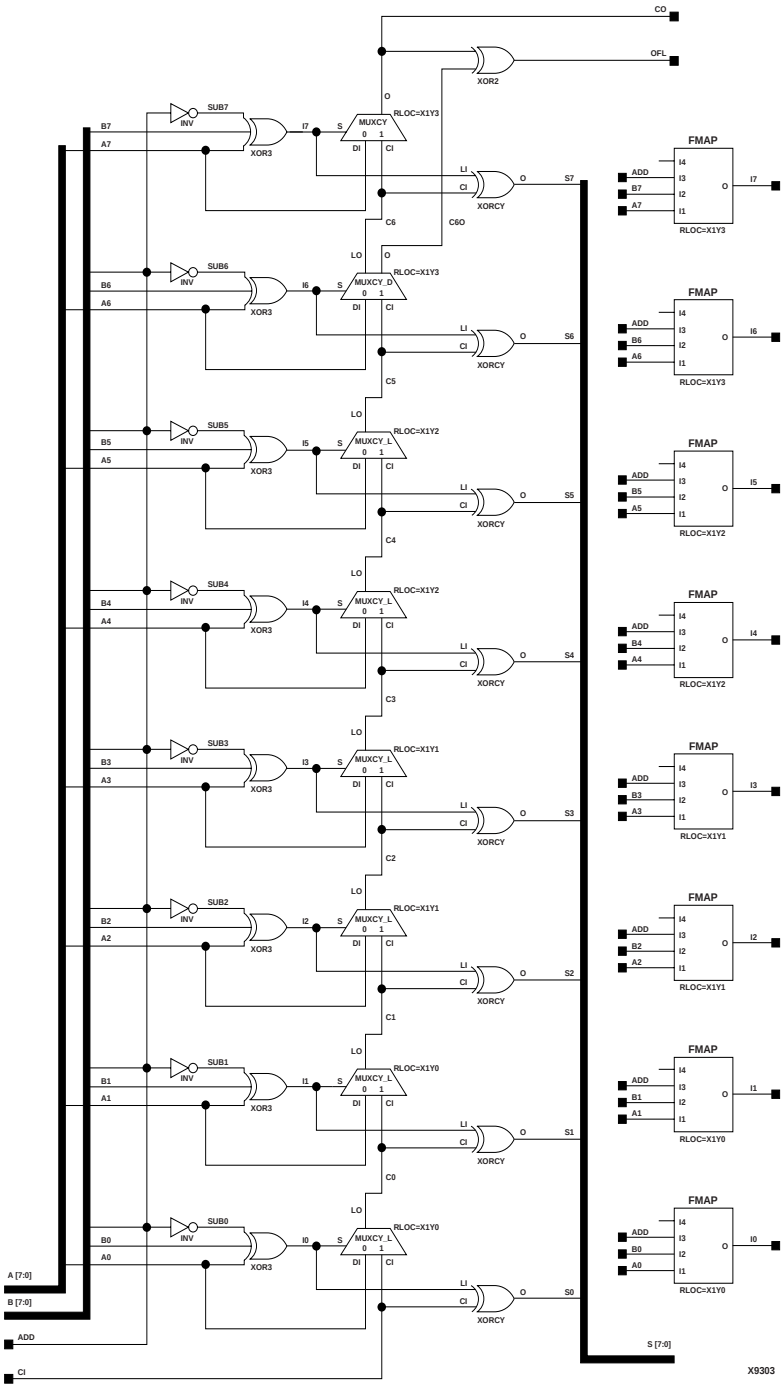
```
unsigned overflow = CO XOR ADD
```

OFL is ignored in unsigned binary operation.

Twos-Complement Operation

For twos-complement operation, ADSU4 can represent numbers between -8 and +7, inclusive; ADSU8 between -128 and +127, inclusive; ADSU16 between -32768 and +32767, inclusive. If an addition or subtraction operation result exceeds this range, the OFL output goes High.

CO is ignored in twos-complement operation.



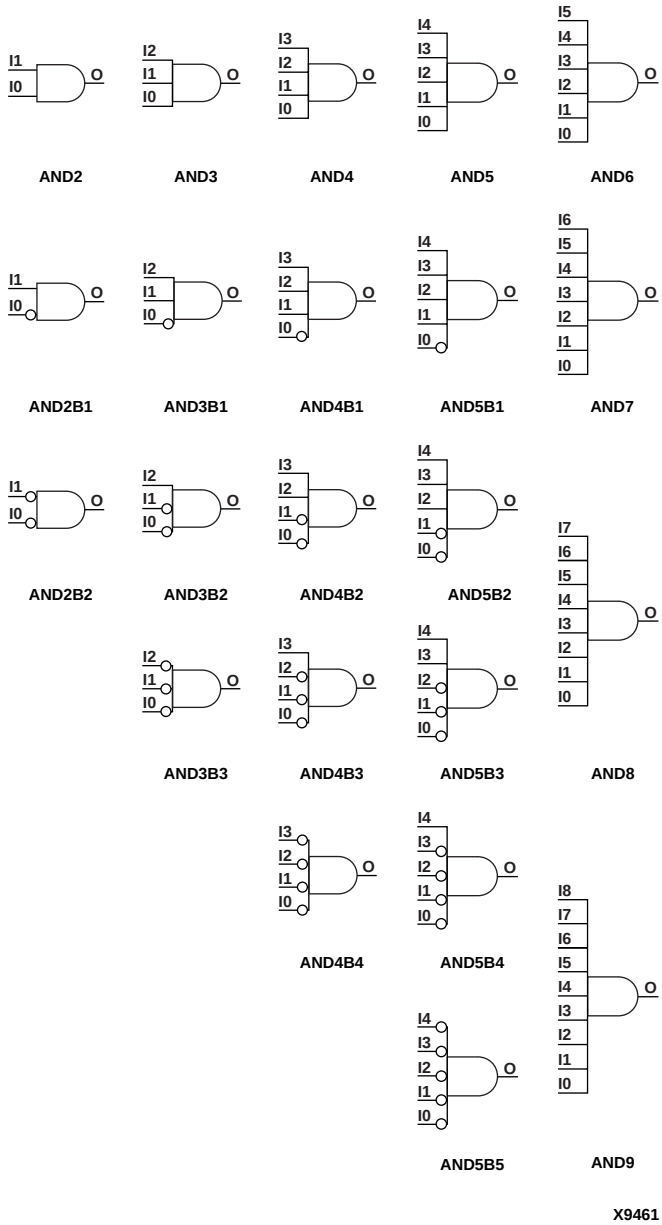
ADSU8 Implementation Spartan-3E

Usage

These design elements are inferred rather than instantiated.

AND2-9

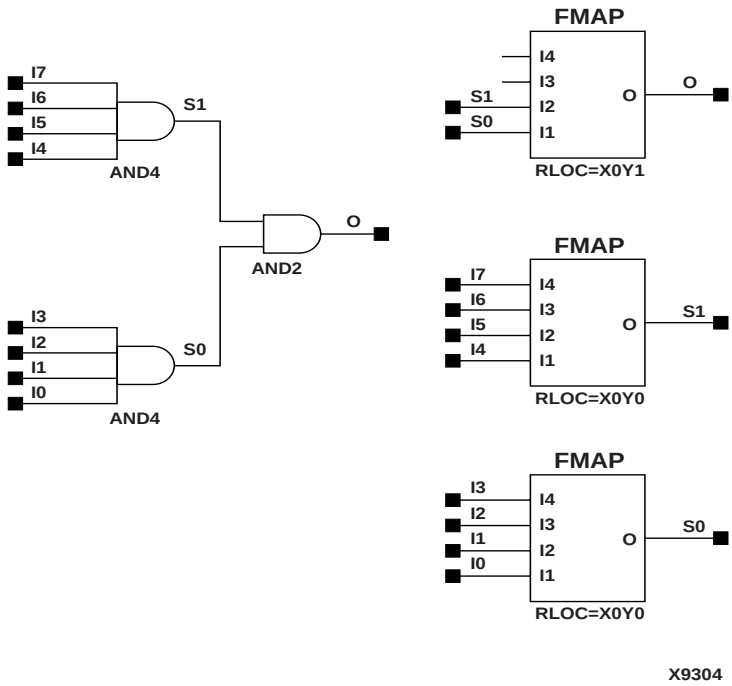
Primitives and Macros: 2- to 9-Input AND Gates with Inverted and Non-Inverted Inputs



AND Gate Representations

AND functions of up to five inputs are available in any combination of inverting and non-inverting inputs. AND functions of six to nine inputs are available with only non-inverting inputs. To make some or all inputs inverting, use external inverters. Because each input uses a CLB resource in Spartan-3E, replacing functions with unused inputs with functions having the appropriate number of inputs.

See “AND12, 16” for information on additional AND functions for Spartan-3E.



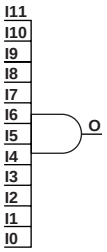
AND8 Implementation Spartan-3E

Usage

If possible, it is recommended that these design elements be inferred rather than instantiated.

AND12, 16

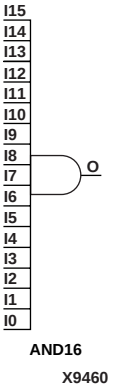
Macros: 12- and 16-Input AND Gates with Non-Inverted Inputs

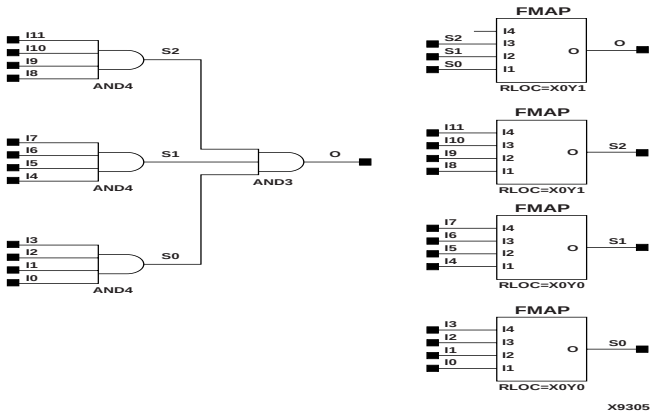


AND12 and AND16 functions are performed in the Configurable Logic Block (CLB) function generator.

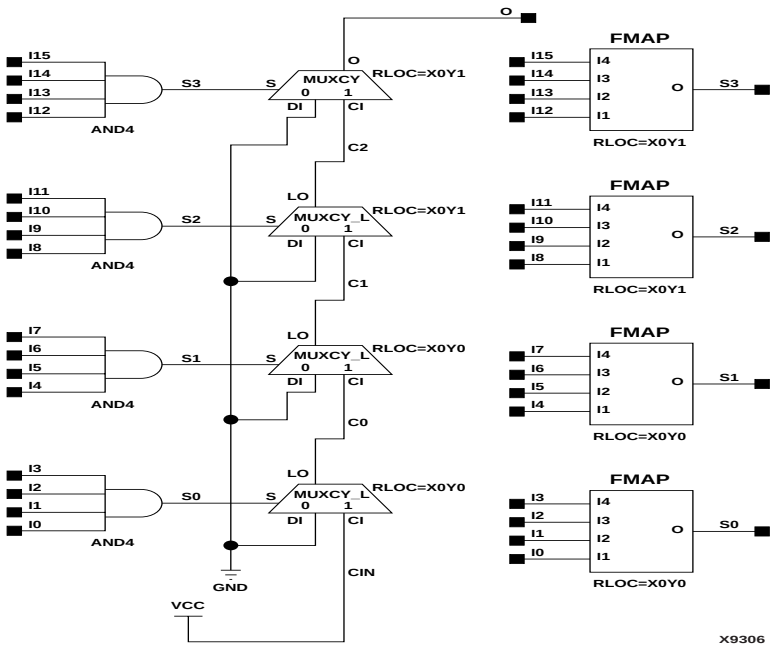
The 12- and 16-input AND functions are available only with non-inverting inputs. To invert all of some inputs, use external inverters.

See "AND2-9" for information on more AND functions.





AND12 Implementation Spartan-3E



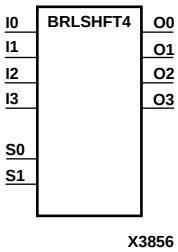
AND16 Implementation Spartan-3E

Usage

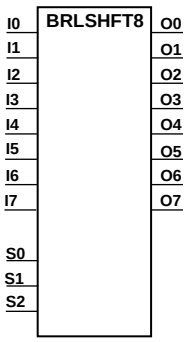
It is recommended that these design elements be inferred rather than instantiated.

BRLSHFT4, 8

Macros: 4-, 8-Bit Barrel Shifters



X3856



X3857

BRLSHFT4, a 4-bit barrel shifter, can rotate four inputs (I3 – I0) up to four places. The control inputs (S1 and S0) determine the number of positions, from one to four, that the data is rotated. The four outputs (O3 – O0) reflect the shifted data inputs.

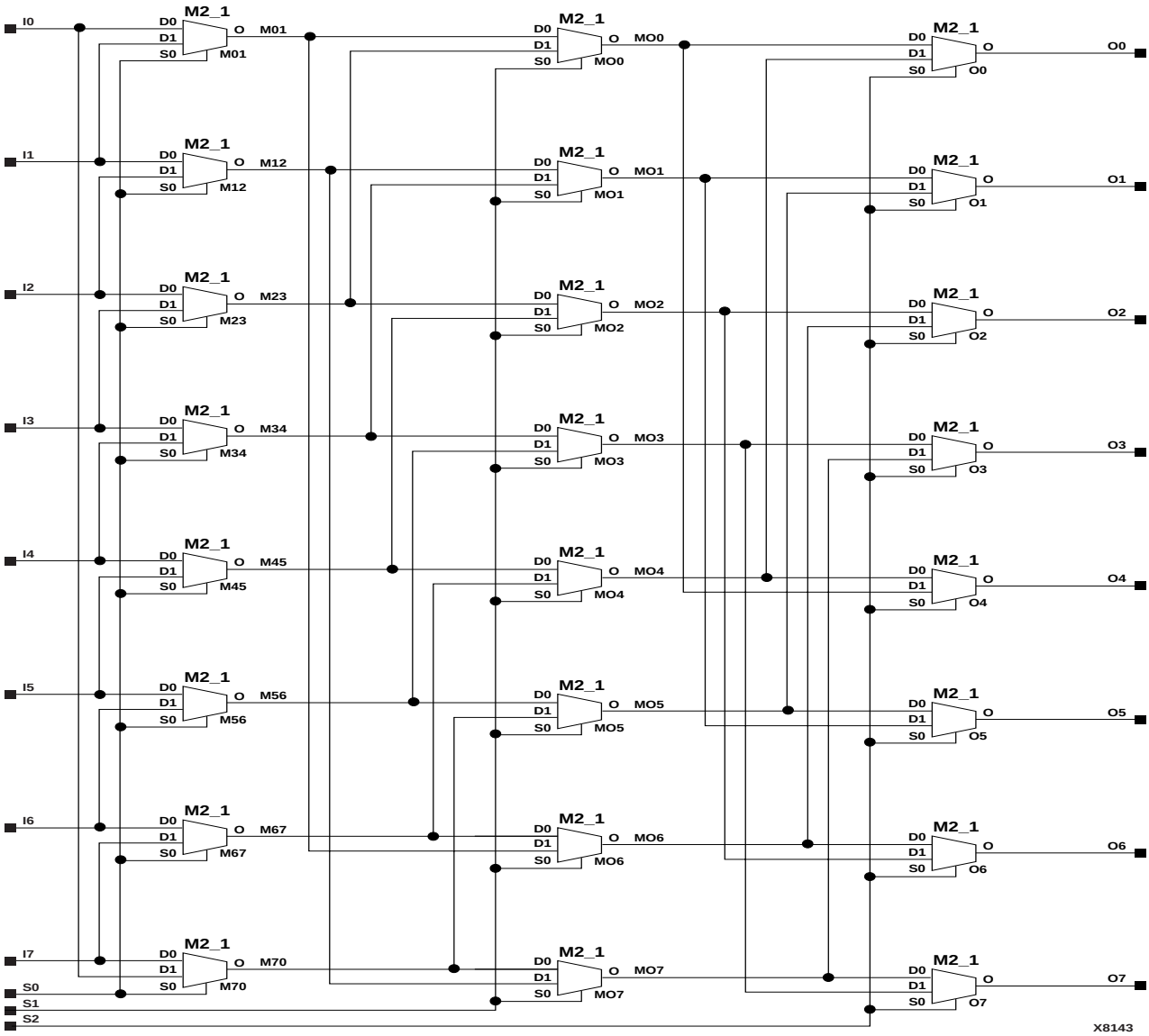
BRLSHFT8, an 8-bit barrel shifter, can rotate the eight inputs (I7 – I0) up to eight places. The control inputs (S2 – S0) determine the number of positions, from one to eight, that the data is rotated. The eight outputs (O7 – O0) reflect the shifted data inputs.

BRLSHFT4 Truth Table

Inputs						Outputs			
S1	S0	I0	I1	I2	I3	O0	O1	O2	O3
0	0	a	b	c	d	a	b	c	d
0	1	a	b	c	d	b	c	d	a
1	0	a	b	c	d	c	d	a	b
1	1	a	b	c	d	d	a	b	c

BRLSHFT8 Truth Table

Inputs											Outputs							
S2	S1	S0	I0	I1	I2	I3	I4	I5	I6	I7	O0	O1	O2	O3	O4	O5	O6	O7
0	0	0	a	b	c	d	e	f	g	h	a	b	c	d	e	f	g	h
0	0	1	a	b	c	d	e	f	g	h	b	c	d	e	f	g	h	a
0	1	0	a	b	c	d	e	f	g	h	c	d	e	f	g	h	a	b
0	1	1	a	b	c	d	e	f	g	h	d	e	f	g	h	a	b	c
1	0	0	a	b	c	d	e	f	g	h	e	f	g	h	a	b	c	d
1	0	1	a	b	c	d	e	f	g	h	f	g	h	a	b	c	d	e
1	1	0	a	b	c	d	e	f	g	h	g	h	a	b	c	d	e	f
1	1	1	a	b	c	d	e	f	g	h	h	a	b	c	d	e	f	g



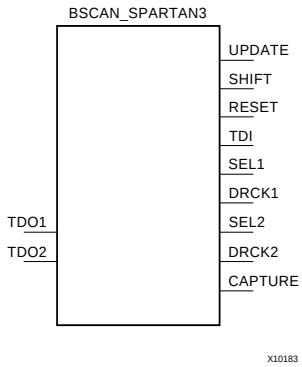
BRLSHFT8 Implementation

Usage

These design elements are inferred rather than instantiated.

BSCAN_SPARTAN3

Primitive: Spartan-3 Boundary Scan Logic Control Circuit



BSCAN_SPARTAN3 currently serves the “BSCAN_SPARTAN-3E” function and provides access to the BSCAN sites on a Spartan-3E device. It is used to create internal boundary scan chains. The 4-pin JTAG interface (TDI, TDO, TCK, and TMS) are dedicated pins in Spartan-3E. To use normal JTAG for boundary scan purposes, just hook up the JTAG pins to the port and go. The pins on the BSCAN_SPARTAN3 symbol do not need to be connected, unless those special functions are needed to drive an internal scan chain.

A signal on the TDO1 input is passed to the external TDO output when the USER1 instruction is executed; the SEL1 output goes High to indicate that the USER1 instruction is active. The DRCK1 output provides USER1 access to the data register clock (generated by the TAP controller). The TDO2 and SEL2 pins perform a similar function for the USER2 instruction and the DRCK2 output provides USER2 access to the data register clock (generated by the TAP controller). The RESET, UPDATE, SHIFT, and CAPTURE pins represent the decoding of the corresponding state of the boundary scan internal state machine. The TDI pin provides access to the TDI signal of the JTAG port in order to shift data into an internal scan chain.

Usage

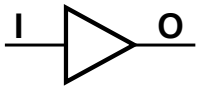
This design element is instantiated rather than inferred.

For More Information

Consult the Spartan-3E Data Sheets.

BUF

Primitive: General-Purpose Buffer



X9444

BUF is a general purpose, non-inverting buffer.

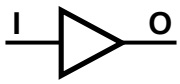
In Spartan-3E, BUF is usually not necessary and is removed by the partitioning software (MAP).

Usage

This design is supported in schematics only.

BUFG

Primitive: Global Clock Buffer



X9428

BUFG, an architecture-independent global buffer, distributes high fan-out clock signals throughout a PLD device. The Xilinx implementation software converts each BUFG to an appropriate type of global buffer for the target PLD device.

Usage

This design element is supported for schematic and instantiation. Synthesis tools usually infer a BUFGP on any clock net. If there are more clock nets than BUFGPs, the synthesis tool usually instantiates BUFGPs for the clocks that are most utilized. The BUFGP contains both a BUFG and an IBUFG.

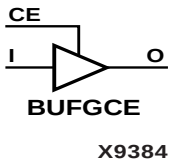
To use a BUFG in a schematic, connect the input of the BUFG symbol to the clock source. The clock source can be an external PAD symbol, an IBUF symbol, or internal logic. For a negative-edge clock input, insert an INV (inverter) symbol between the BUFG output and the clock input. The inversion is implemented at the Configurable Logic Block (CLB) or Input Output Block (IOB) clock pin.

For More Information

Consult the Spartan-3E Data Sheets.

BUFGCE

Primitive: Global Clock Buffer with Clock Enable and Output State 0



X9384

BUFGCE is a clock buffer with one clock input, one clock output, and a clock enable line. Its O output is "0" when clock enable (CE) is Low (inactive). When clock enable (CE) is High, the I input is transferred to the O output.

Inputs		Outputs
I	CE	O
X	0	0
I	1	I

Usage

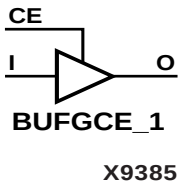
This design element is supported for schematics and instantiations only.

For More Information

Consult the Spartan-3E Data Sheets.

BUFGCE_1

Primitive: Global Clock Buffer with Clock Enable and Output State 1



BUFGCE_1 is a clock buffer with one clock input, one clock output, and a clock enable line. Its O output is High (1) when clock enable (CE) is Low (inactive). When clock enable (CE) is High, the I input is transferred to the O output.

Inputs		Outputs
I	CE	O
X	0	1
I	1	I

Usage

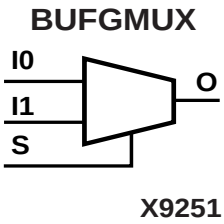
This design element is supported for schematics and instantiations only.

For More Information

Consult the Spartan-3E Data Sheets.

BUFGMUX

Primitive: Global Clock MUX Buffer with Output State 0



BUFGMUX is a multiplexed global clock buffer that can select between two input clocks I0 and I1. When the select input (S) is Low, the signal on I0 is selected for output (O). When the select input (S) is High, the signal on I1 is selected for output.

BUFGMUX and BUFGMUX_1 are distinguished by which state the output assumes when it switches between clocks in response to a change in its select input. BUGFMUX assumes output state 0 and BUFGMUX_1 assumes output state 1.

Note: BUFGMUX guarantees that when S is toggled, the state of the output will remain in the inactive state until the next active clock edge (either I0 or I1) occurs.

Inputs			Outputs
I0	**I1**	**S**	**O**
I0	X	0	I0
X	I1	1	I1
X	X	↑	0
X	X	↓	0

Usage

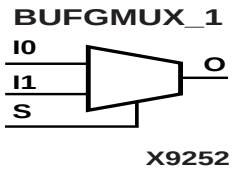
This design element is supported for schematics and instantiations only.

For More Information

Consult the Spartan-3E Data Sheets.

BUFGMUX_1

Primitive: Global Clock MUX Buffer with Output State 1



BUFGMUX_1 is a multiplexed global clock buffer that can select between two input clocks I0 and I1. When the select input (S) is Low, the signal on I0 is selected for output (O). When the select input (S) is High, the signal on I1 is selected for output.

BUFGMUX and BUFGMUX_1 are distinguished by which state the output assumes when it switches between clocks in response to a change in its select input. BUFGMUX assumes output state 0 and BUFGMUX_1 assumes output state 1.

Inputs			Outputs
I0	**I1**	**S**	**O**
I0	X	0	I0
X	I1	1	I1
X	X	↑	1
X	X	↓	1

Usage

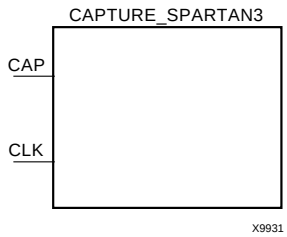
This design element is supported for schematics and instantiations only.

For More Information

Consult the Spartan-3E Data Sheets.

CAPTURE_SPARTAN3

Primitive: Spartan-3 Register State Capture for Bitstream Readback



CAPTURE_SPARTAN3 currently serves the function of “CAPTURE_SPARTAN3E” and provides user control over when to capture register (flip-flop and latch) information for readback. Spartan-3E devices provide the readback function through dedicated configuration port instructions.

The CAPTURE_SPARTAN3 symbol is optional. Without it, readback is still performed, but the asynchronous capture function it provides for register states is not available.

Spartan-3E allows for capturing register (flip-flop and latch) states only. Although LUT RAM, SRL, and block RAM states are read back, they cannot be captured. An asserted high CAP signal indicates that the registers in the device are to be captured at the next Low-to-High clock transition.

By default, data is captured after every trigger (transition on CLK while CAP is asserted). To limit the readback operation to a single data capture, add the ONESHOT attribute to CAPTURE_SPARTAN3. See the *Constraints Guide* for information on the ONESHOT attribute.

Usage

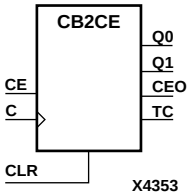
This design element is instantiated rather than inferred.

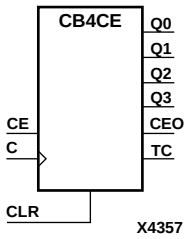
For More Information

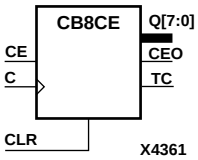
Consult the Spartan-3E Data Sheets.

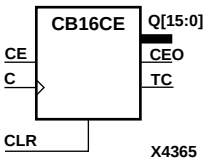
CB2CE, CB4CE, CB8CE, CB16CE

Macro: 2-, 4-, 8-,16-Bit Cascadable Binary Counters with Clock Enable and Asynchronous Clear









CB2CE, CB4CE, CB8CE, and CB16CE are, respectively, 2-, 4-, 8-, and 16-bit (stage), asynchronous, clearable, cascadable binary counters. The asynchronous clear (CLR) is the highest priority input. When CLR is High, all other inputs are ignored; the Q outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero, independent of clock transitions. The Q outputs increment when the clock enable input (CE) is High during the Low-to-High clock (C) transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs are High.

Larger counters are created by connecting the CEO output of the first stage to the CE input of the next stage and connecting the C and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than $n(t_{CE\text{-}TC})$, where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input; use the TC output if it does not.

The counter is asynchronously cleared, outputs Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

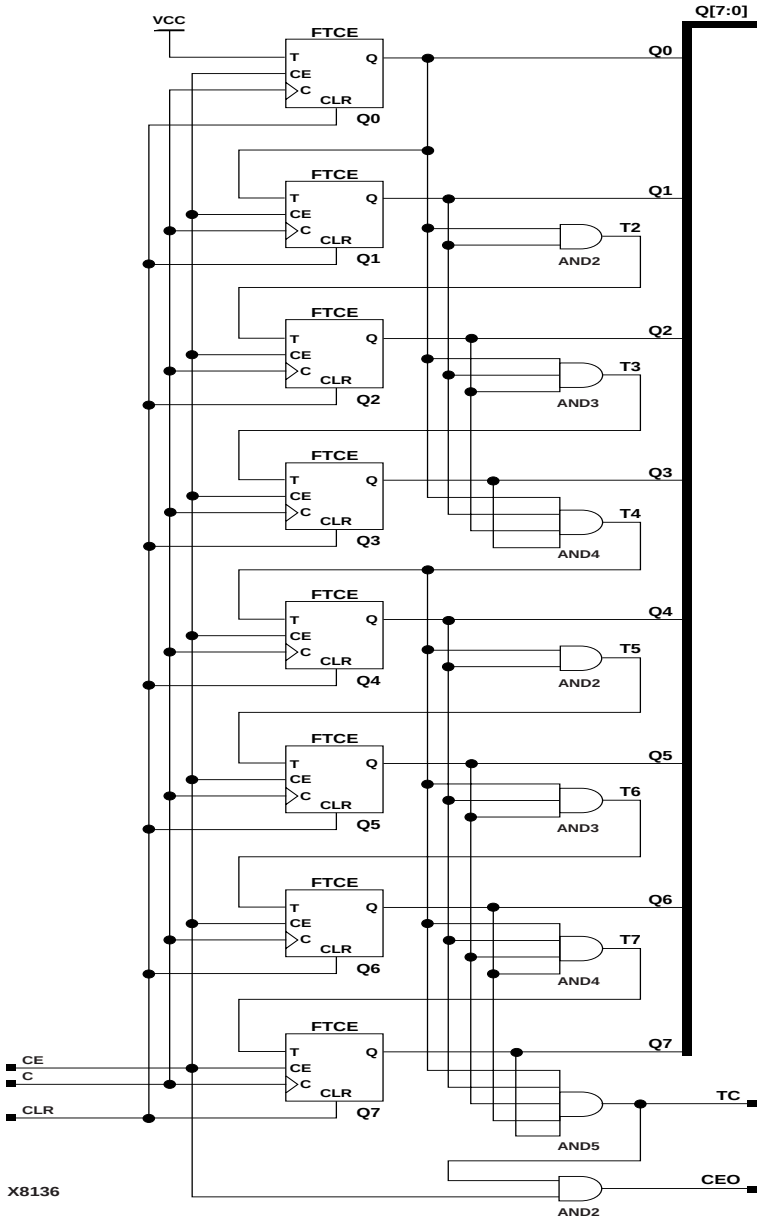
GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs		
CLR	CE	C	Qz-Q0	TC	CEO
1	X	X	0	0	0
0	0	X	No Change	No Change	0
0	1	↑	Inc	TC	CEO

z= 1 for CB2CE; z = 3 for CB4CE; z = 7 for CB8CE; z = 15 for CB16CE

TC = Qz • Q(z-1) • Q(z-2) • ... • Q0

CEO = TC • CE



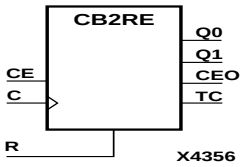
CB8CE Implementation for Spartan-3E

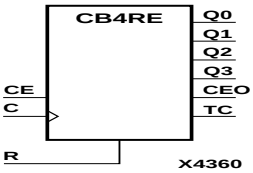
Usage

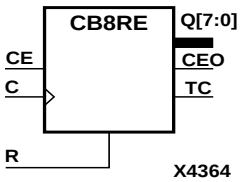
These design elements are inferred rather than instantiated.

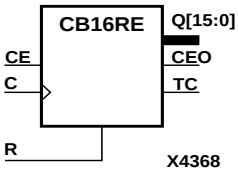
CB2RE, CB4RE, CB8RE, CB16RE

Macro: 2-, 4-, 8-, 16-Bit Cascadable Binary Counters with Clock Enable and Synchronous Reset









CB2RE, CB4RE, CB8RE, and CB16RE are, respectively, 2-, 4-, 8-, and 16-bit (stage), synchronous, resettable, cascadable binary counters. The synchronous reset (R) is the highest priority input. When R is High, all other inputs are ignored; the Q outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero during the Low-to-High clock transition. The Q outputs increment when the clock enable input (CE) is High during the Low-to-High clock (C) transition. The counter ignores clock transitions when CE is Low. The TC output is High when both Q outputs are High.

Larger counters are created by connecting the CEO output of the first stage to the CE input of the next stage and connecting the C and R inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than $n(t_{CE\text{-}TC})$, where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input; use the TC output if it does not.

The counter is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

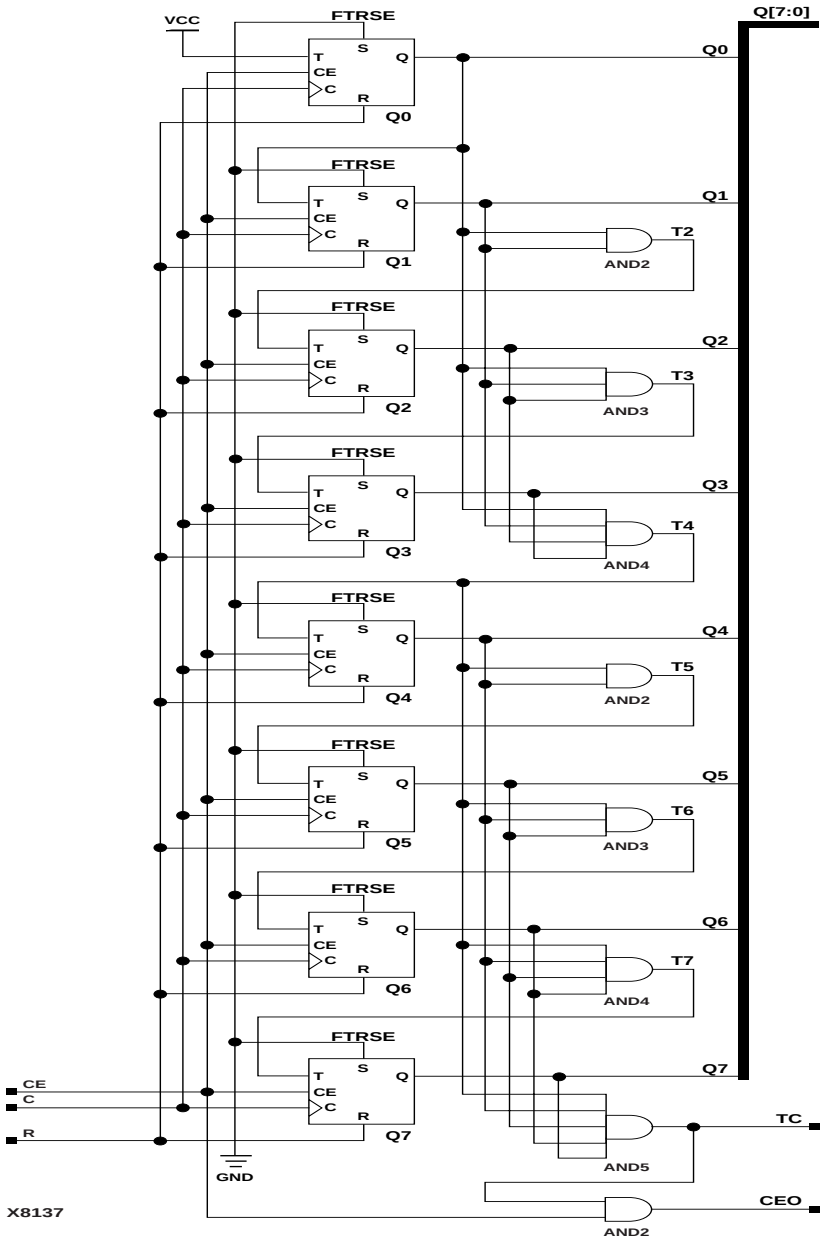
GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs		
R	CE	C	Qz – Q0	TC	CEO
1	X	↑	0	0	0
0	0	X	No Change	No Change	0
0	1	↑	Inc	TC	CEO

z = 1 for CB2RE; z = 3 for CB4RE; z = 7 for CB8RE; z = 15 for CB16RE

TC = Qz • Q(z-1) • Q(z-2) • ... • Q0)

CEO = TC • CE



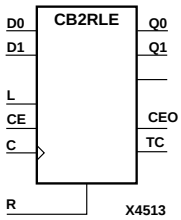
CB8RE Implementation for Spartan-3E

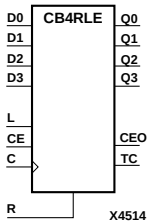
Usage

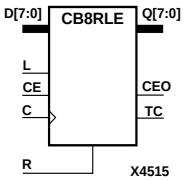
These design elements are inferred rather than instantiated.

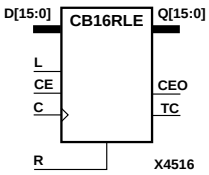
CB2RLE,CB4RLE, CB8RLE, CB16RLE

Macro: 2-, 4-, 8-, 16-Bit Loadable Cascadable Binary Counters with Clock Enable and Synchronous Reset









CB2RLE, CB4RLE, CB8RLE, and CB16RLE are, respectively, 2-, 4-, 8-, and 16-bit (stage), synchronous, loadable, resettable, cascadable binary counters. The synchronous reset (R) is the highest priority input. The synchronous R, when High, overrides all other inputs and resets the Q outputs, terminal count (TC), and clock enable out (CEO) outputs to Low on the Low-to-High clock (C) transition.

The data on the D inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock (C) transition, independent of the state of CE. The Q outputs increment when CE is High during the Low-to-High clock transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs are High. The CEO output is High when all Q outputs and CE are High to allow direct cascading of counters.

Larger counters are created by connecting the CEO output of the first stage to the CE input of the next stage and by connecting the C, L, and R inputs in parallel. The maximum length of the counter is determined by the accumulated CE-to-CEO propagation delays versus the clock period. When cascading counters, use the CEO output if the counter uses the CE input; use the TC output if it does not.

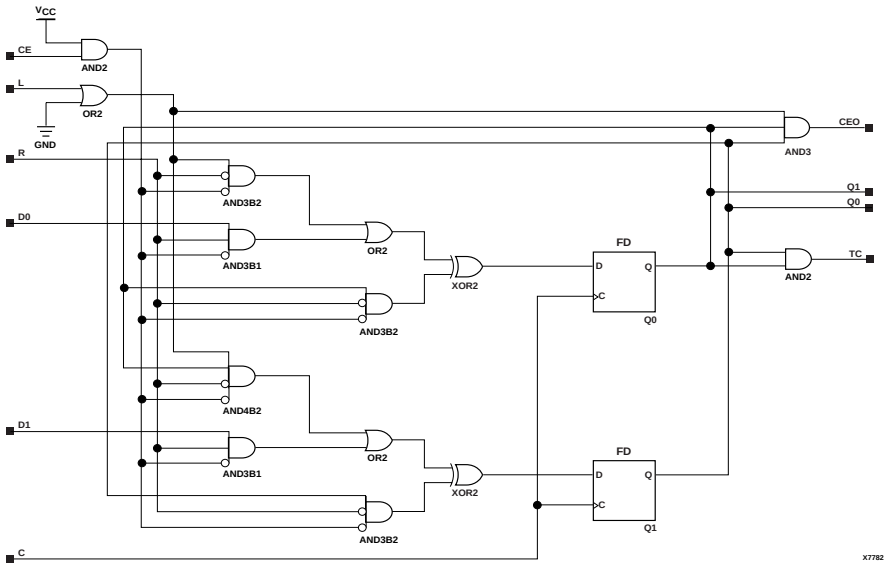
The counter is asynchronously cleared, output Low, when power is applied. For XC9500/XV/XL, CoolRunner XPLA3, and CoolRunner-II, the power-on condition can be simulated by applying a High-level pulse on the PRLD global net.

Inputs					Outputs		
R	L	CE	C	Dz – D0	Qz – Q0	TC	CEO
1	X	X	↑	X	0	0	0
0	1	X	↑	Dn	Dn	TC	CEO
0	0	0	X	X	No Chg	No Chg	0
0	0	1	↑	X	Inc	TC	CEO

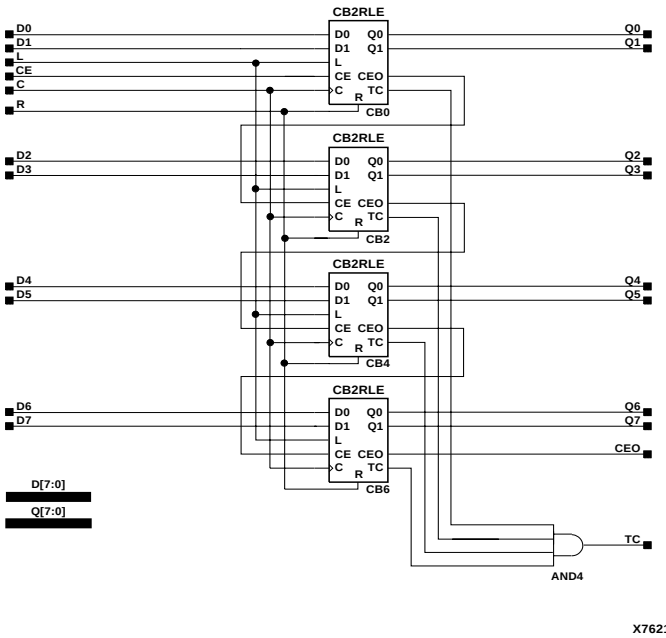
z = 1 for CB2RLE; z = 3 for CB4RLE; z = 7 for CB8RLE; z = 15 for CB16RLE

TC = Qz • Q(z-1) • Q(z-2) • ... • Q0

CEO = TC • CE



CB2RLE Implementation XC9500/XV/XL, CoolRunner XPLA3, CoolRunner-II



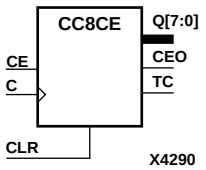
CB8RLE Implementation XC9500/XV/XL, CoolRunner XPLA3, CoolRunner-II

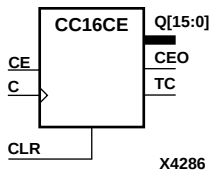
Usage

These design elements are inferred rather than instantiated.

CC8CE, CC16CE

Macro: 8-, 16-Bit Cascadable Binary Counters with Clock Enable and Asynchronous Clear





CC8CE and CC16CE are, respectively, 8- and 16-bit (stage), asynchronous clearable, cascadable binary counters. These counters are implemented using carry logic with relative location constraints to ensure efficient placement of logic. The asynchronous clear (CLR) is the highest priority input. When CLR is High, all other inputs are ignored; the Q outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero, independent of clock transitions. The Q outputs increment when the clock enable input (CE) is High during the Low-to-High clock (C) transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs are High.

Larger counters are created by connecting the count enable out (CEO) output of the first stage to the CE input of the next stage and connecting the C and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than $n(t_{CE\text{-}TC})$, where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input; use the TC output if it does not.

The counter is asynchronously cleared, with Low outputs, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

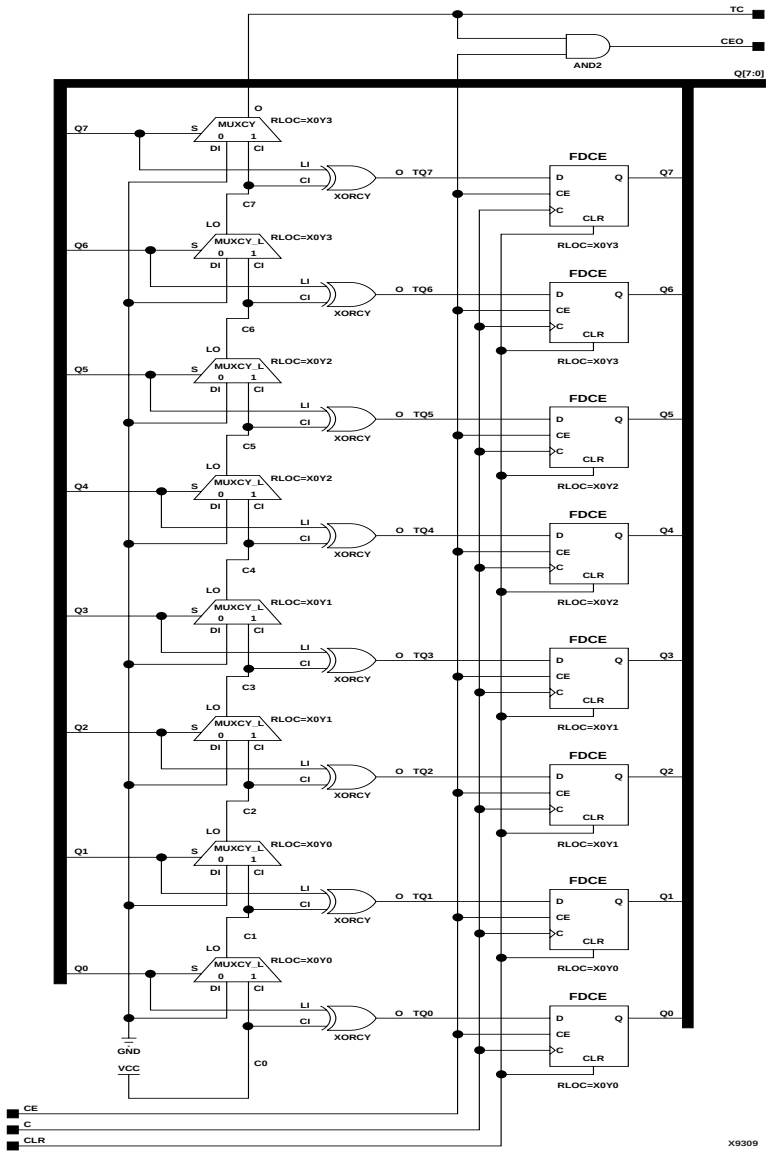
GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs		
CLR	CE	C	Qz – Q0	TC	CEO
1	X	X	0	0	0
0	0	X	No Change	No Change	0
0	1	↑	Inc	TC	CEO

z = 7 for CC8CE; z = 15 for CC16CE

TC = Qz • Q(z-1) • Q(z-2) • ... • Q0

CEO = TC • CE



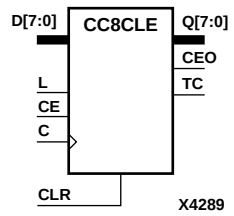
CC8CE Implementation Spartan-3E

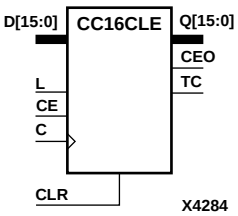
Usage

These design elements are inferred rather than instantiated.

CC8CLE, CC16CLE

Macro: 8-, 16-Bit Loadable Cascadable Binary Counters with Clock Enable and Asynchronous Clear





CC8CLE and CC16CLE are, respectively, 8- and 16-bit (stage), synchronously loadable, asynchronously clearable, cascadable binary counters. These counters are implemented using carry logic with relative location constraints to ensure efficient placement of logic.

The asynchronous clear (CLR) is the highest priority input. When CLR is High, all other inputs are ignored; the Q outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero, independent of clock transitions. The data on the D inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock (C) transition, independent of the state of clock enable (CE). The Q outputs increment when CE is High during the Low-to-High clock transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs are High.

Larger counters are created by connecting the count enable out (CEO) output of the first stage to the CE input of the next stage and connecting the C, L, and CLR inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than $n(t_{CE\text{-}TC})$, where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input; use the TC output if it does not.

The counter is asynchronously cleared, with Low output, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

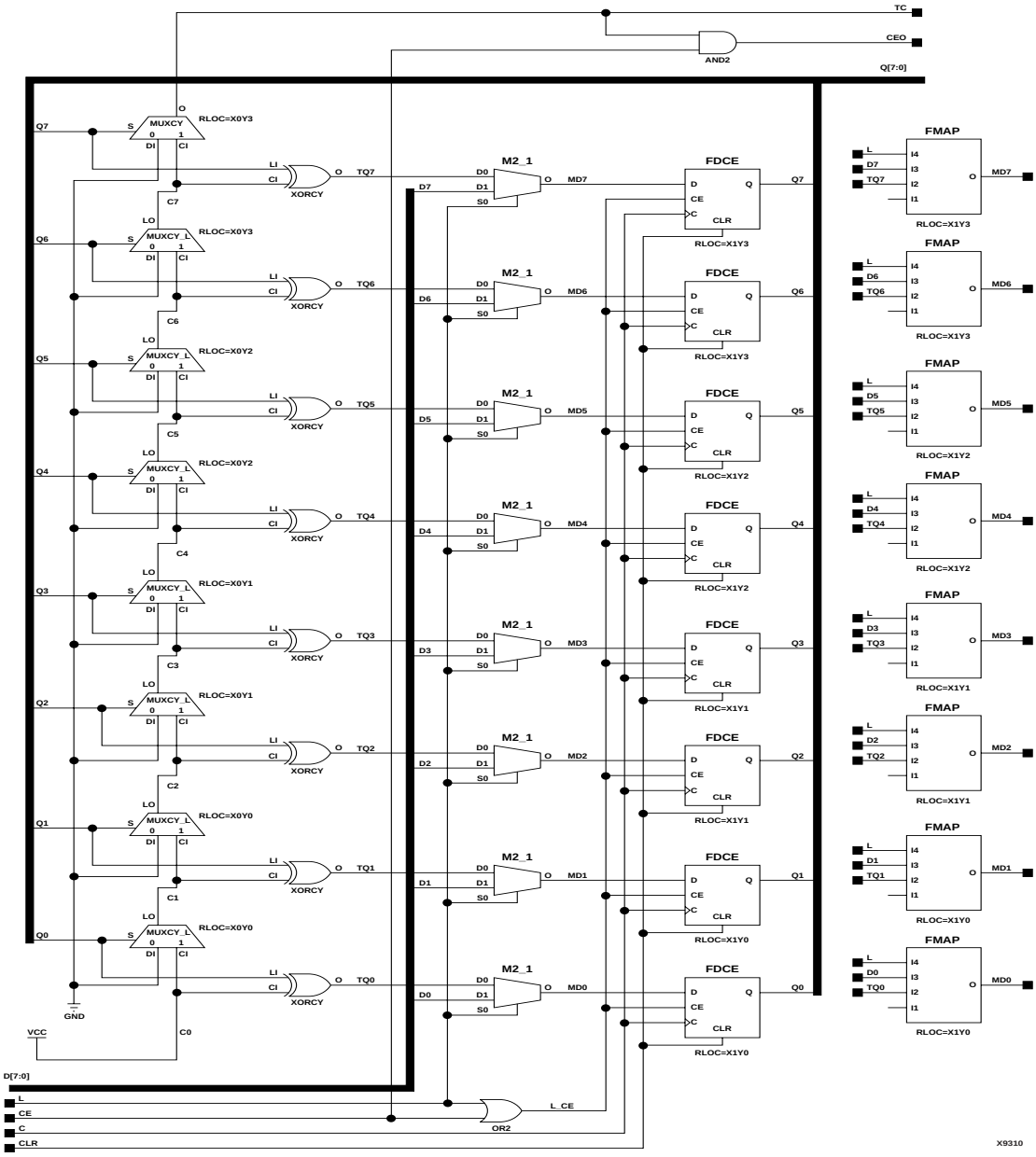
GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs					Outputs		
CLR	L	CE	C	Dz – D0	Qz – Q0	TC	CEO
1	X	X	X	X	0	0	0
0	1	X	↑	Dn	Dn	TC	CEO
0	0	0	X	X	No Change	No Change	0
0	0	1	↑	X	Inc	TC	CEO

z = 7 for CC8CLE; z = 15 for CC16CLE

TC = Qz • Q(z-1) • Q(z-2) • ... • Q0

CEO = TC • CE



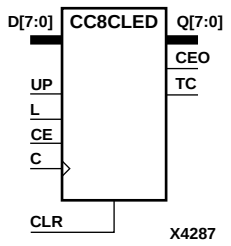
CC8CLE Implementation Spartan-3E

Usage

These design elements are inferred rather than instantiated.

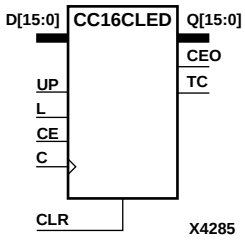
CC8CLED, CC16CLED

Macro: 8-, 16-Bit Loadable Cascadable Bidirectional Binary Counters with Clock Enable and Asynchronous Clear



CC8CLED and CC16CLED are, respectively, 8- and 16-bit (stage), synchronously loadable, asynchronously clearable, cascadable, bidirectional binary counters. These counters are implemented using carry logic with relative location constraints, which assures most efficient logic placement.

The asynchronous clear (CLR) is the highest priority input. When CLR is High, all other inputs are ignored; the Q outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero, independent of clock transitions. The data on the D inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock (C) transition, independent of the state of clock enable (CE). The Q outputs decrement when CE is High and UP is Low during the Low-to-High clock transition. The Q outputs increment when CE and UP are High. The counter ignores clock transitions when CE is Low.



For counting up, the TC output is High when all Q outputs and UP are High. For counting down, the TC output is High when all Q outputs and UP are Low. To cascade counters, the count enable out (CEO) output of each counter is connected to the CE pin of the next stage. The clock, UP, L, and CLR inputs are connected in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than $n(t_{CE\text{-}TC})$, where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input; use the TC output if it does not.

The counter is asynchronously cleared, outputs Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

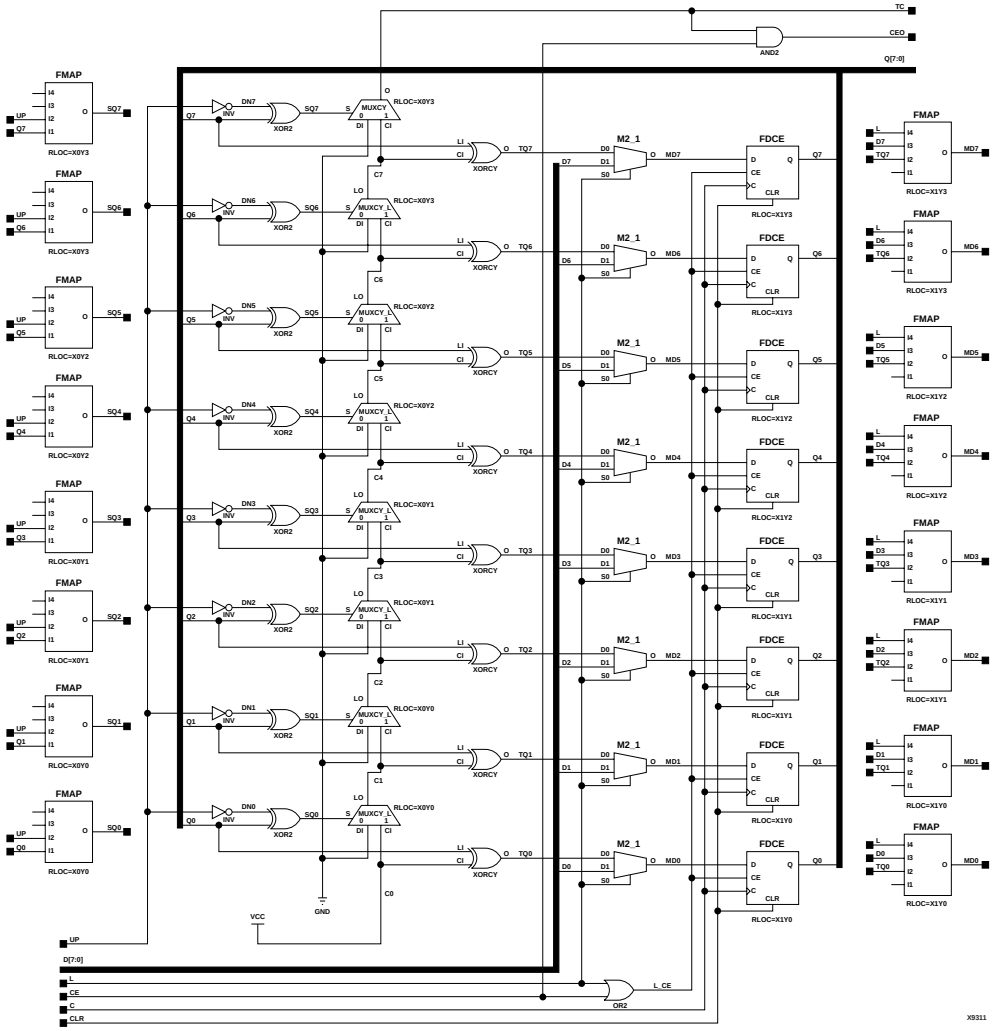
GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs						Outputs		
CLR	L	CE	C	UP	Dz – D0	Qz – Q0	TC	CEO
1	X	X	X	X	X	0	0	0
0	1	X	↑	X	Dn	Dn	TC	CEO
0	0	0	X	X	X	No Change	No Change	0
0	0	1	↑	1	X	Inc	TC	CEO
0	0	1	↑	0	X	Dec	TC	CEO

z = 7 for CC8CLED; z = 15 for CC16CLED

TC = (Qz • Q(z-1) • Q(z-2) • ... • Q0 • UP) + (Qz • Q(z-1) • Q(z-2) • ... • Q0 • UP)

CEO = TC • CE



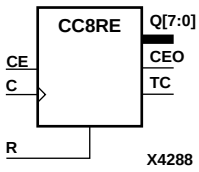
CC8CLED Implementation Spartan-3E

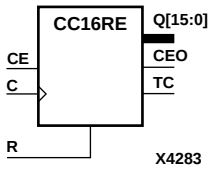
Usage

These design elements are inferred rather than instantiated.

CC8RE, CC16RE

Macro: 8-, 16-Bit Cascadable Binary Counters with Clock Enable and Synchronous Reset





CC8RE and CC16RE are, respectively, 8- and 16-bit (stage), synchronous resettable, cascadable binary counters. These counters are implemented using carry logic with relative location constraints to ensure efficient placement of logic. The synchronous reset (R) is the highest priority input. When R is High, all other inputs are ignored; the Q outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero on the Low-to-High clock (C) transition. The Q outputs increment when the clock enable input (CE) is High during the Low-to-High clock transition. The counter ignores clock transitions when CE is Low. The TC output is High when all Q outputs and CE are High.

Larger counters are created by connecting the CEO output of the first stage to the CE input of the next stage and connecting the C and R inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than $n(t_{CE\text{-}TC})$, where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input; use the TC output if it does not.

The counter is asynchronously cleared, with Low outputs, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

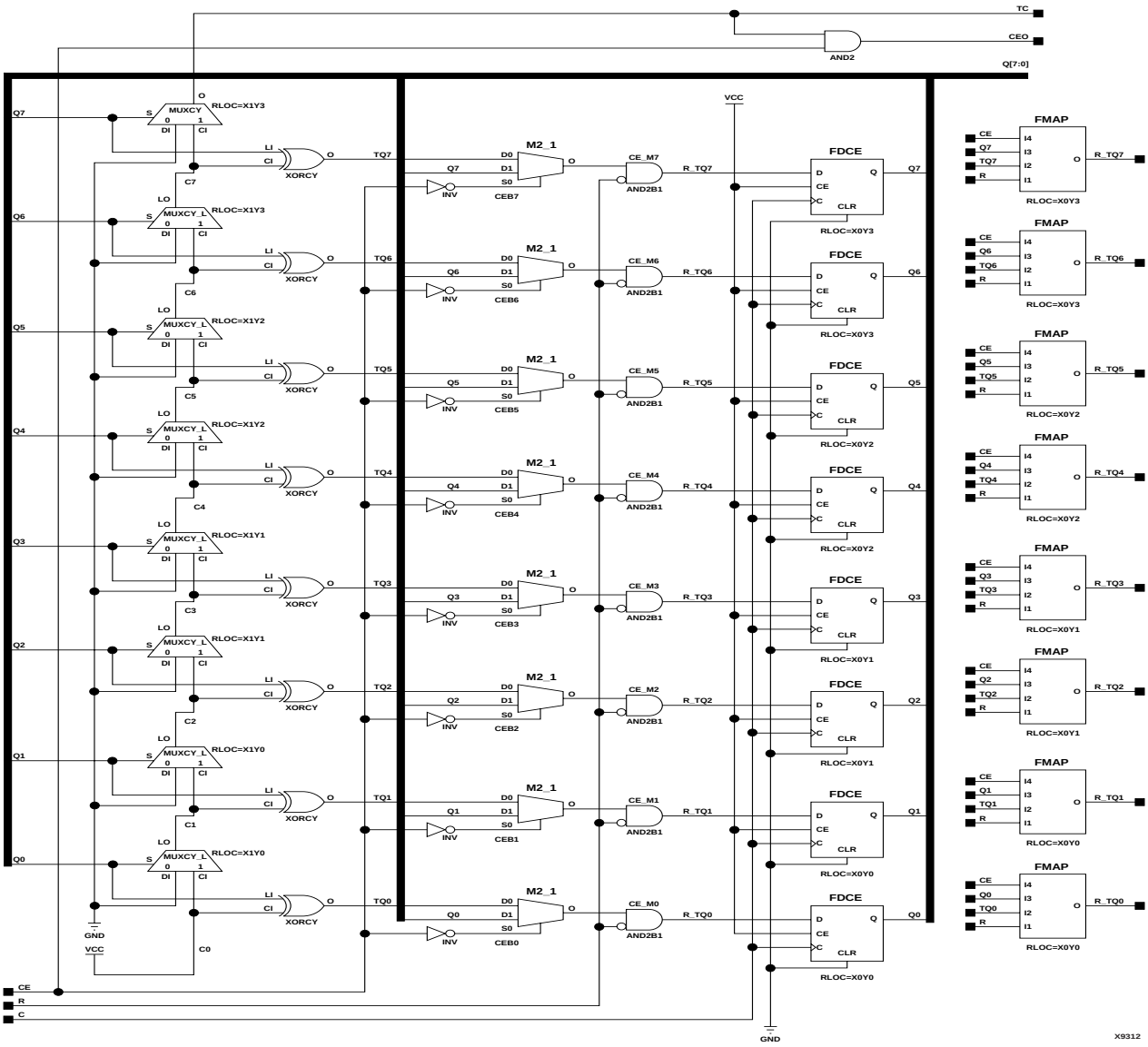
GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs		
R	CE	C	Qz – Q0	TC	CEO
1	X	↑	0	0	0
0	0	X	No Change	No Change	0
0	1	↑	Inc	TC	CEO

z = 7 for CC8RE; z = 15 for CC16RE

TC = Qz • Q(z-1) • Q(z-2) • ... • Q0 • CE

CEO = TC • CE



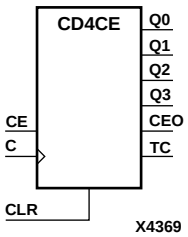
CC8RE Implementation Spartan-3E

Usage

These design elements are inferred rather than instantiated.

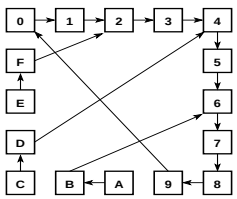
CD4CE

Macro: 4-Bit Cascadable BCD Counter with Clock Enable and Asynchronous Clear



CD4CE is a 4-bit (stage), asynchronous clearable, cascadable binary-coded-decimal (BCD) counter. The asynchronous clear input (CLR) is the highest priority input. When CLR is High, all other inputs are ignored; the Q outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero, independent of clock transitions. The Q outputs increment when clock enable (CE) is High during the Low-to-High clock (C) transition. The counter ignores clock transitions when CE is Low. The TC output is High when Q3 and Q0 are High and Q2 and Q1 are Low.

The counter recovers from any of six possible illegal states and returns to a normal count sequence within two clock cycles for Spartan-3E, as shown in the following state diagram.



Larger counters are created by connecting the count enable out (CEO) output of the first stage to the CE input of the next stage and connecting the CLR and clock inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than $n(t_{CE\text{-}TC})$, where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input; use the TC output if it does not.

The counter is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

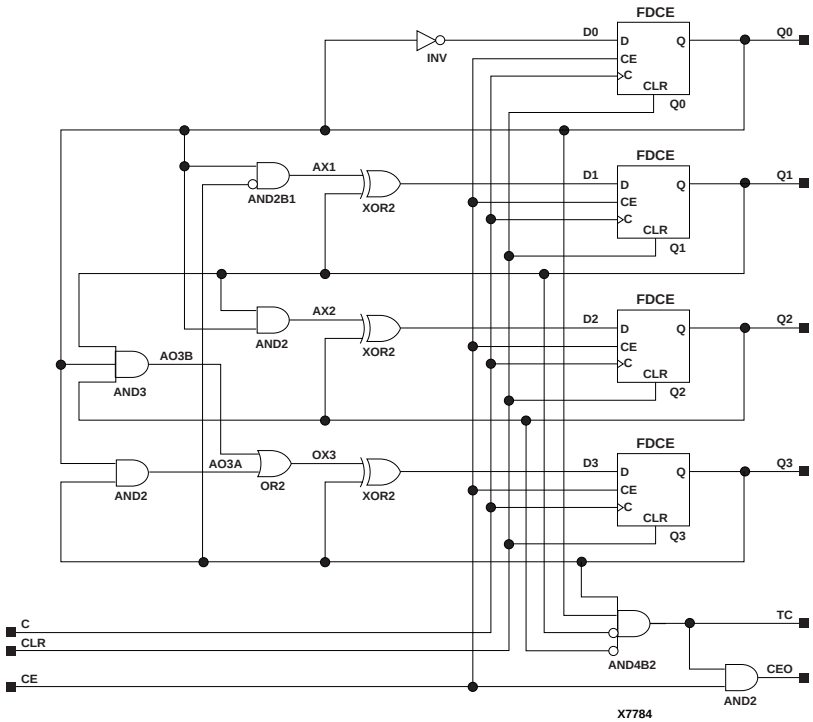
The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs					
CLR	CE	C	Q3	Q2	Q1	Q0	TC	CEO
1	X	X	0	0	0	0	0	0
0	1	↑	Inc	Inc	Inc	Inc	TC	CEO
0	0	X	No Change	No Change	No Change	No Change	TC	0

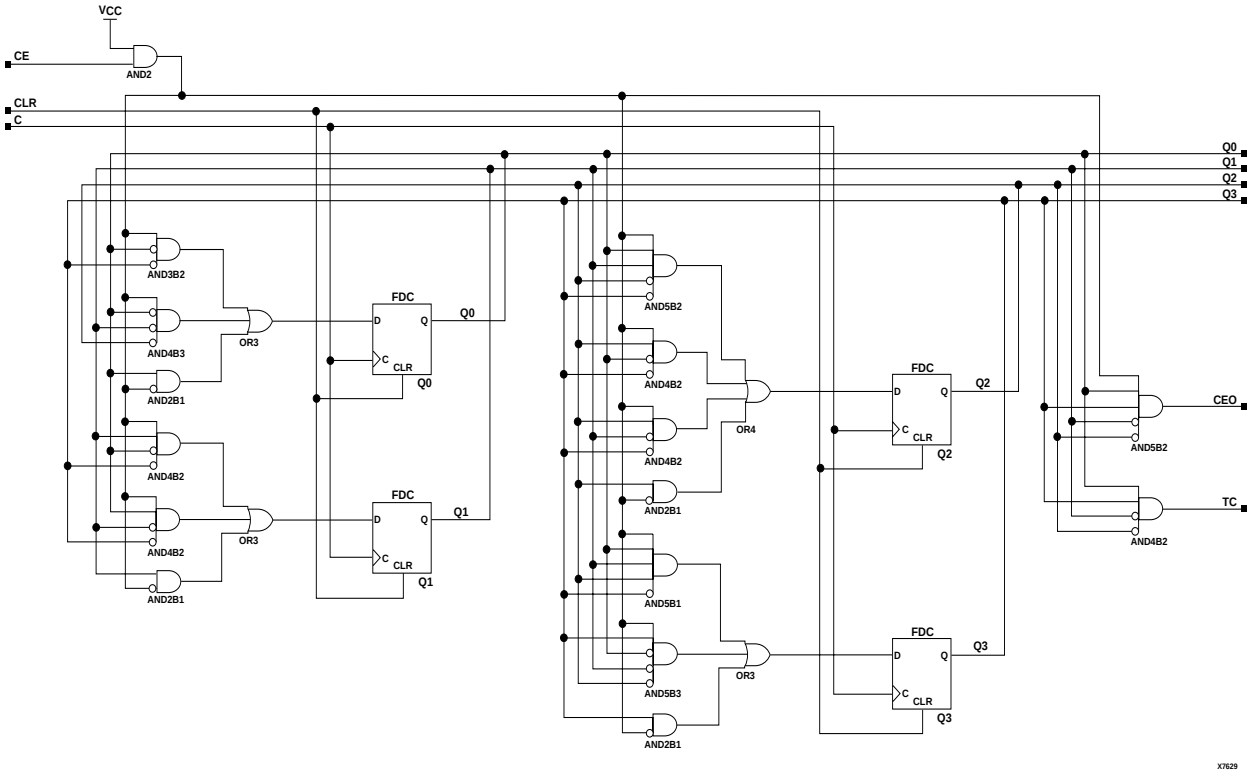
Inputs			Outputs					
CLR	CE	C	Q3	Q2	Q1	Q0	TC	CEO
0	1	X	1	0	0	1	1	1

TC = Q3•!Q2•!Q1•Q0

CEO = TC•CE



CD4CE Implementation for Spartan-3E



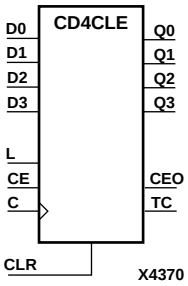
CD4CE Implementation for Spartan-3E

Usage

This design element can be inferred.

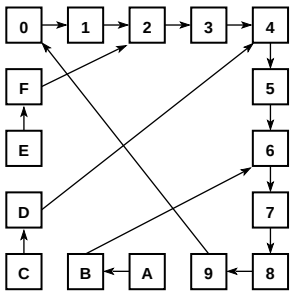
CD4CLE

Macro: 4-Bit Loadable Cascadable BCD Counter with Clock Enable and Asynchronous Clear



CD4CLE is a 4-bit (stage), synchronously loadable, asynchronously clearable, binary-coded-decimal (BCD) counter. The asynchronous clear input (CLR) is the highest priority input. When (CLR) is High, all other inputs are ignored; the (Q) outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero, independent of clock transitions. The data on the (D) inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock (C) transition. The (Q) outputs increment when clock enable input (CE) is High during the Low- to-High clock transition. The counter ignores clock transitions when (CE) is Low. The (TC) output is High when Q3 and Q0 are High and Q2 and Q1 are Low.

The counter recovers from any of six possible illegal states and returns to a normal count sequence within two clock cycles for Xilinx devices, as shown in the following state diagram.



Larger counters are created by connecting the count enable out (CEO) output of the first stage to the (CE) input of the next stage and connecting the CLR, L, and C inputs in parallel. (CEO) is active (High) when (TC) and (CE) are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than $n(t_{CE\text{-}TC})$, where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the (CEO) output if the counter uses the (CE) input; use the (TC) output if it does not.

The counter is asynchronously cleared, output Low, when power is applied.

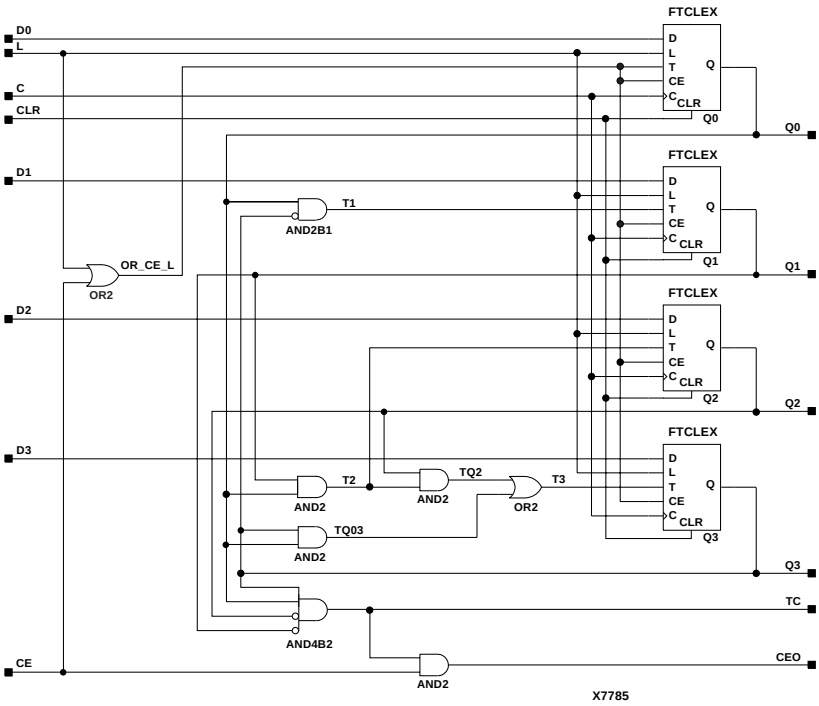
For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs					Outputs					
CLR	L	CE	D3 – D0	C	Q3	Q2	Q1	Q0	TC	CEO
1	X	X	X	X	0	0	0	0	0	0
0	1	X	D3 – D0	↑	D3	D2	D1	D0	TC	CEO
0	0	1	X	↑	Inc	Inc	Inc	Inc	TC	CEO
0	0	0	X	X	No Change	No Change	No Change	No Change	TC	0
0	0	1	X	X	1	0	0	1	1	1

TC = Q3•!Q2•!Q1•Q0

CEO = TC•CE



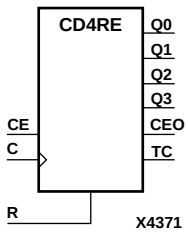
CD4CLE Implementation of Spartan-3E

Usage

These design elements are supported only for schematics *and* instantiation.

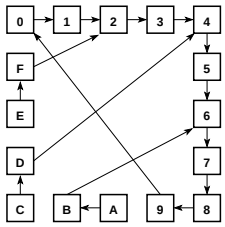
CD4RE

Macro: 4-Bit Cascadable BCD Counter with Clock Enable and Synchronous Reset



CD4RE is a 4-bit (stage), synchronous resettable, cascadable binary-coded-decimal (BCD) counter. The synchronous reset input (R) is the highest priority input. When (R) is High, all other inputs are ignored; the (Q) outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero on the Low-to-High clock (C) transition. The (Q) outputs increment when the clock enable input (CE) is High during the Low-to-High clock transition. The counter ignores clock transitions when (CE) is Low. The (TC) output is High when Q3 and Q0 are High and Q2 and Q1 are Low.

The counter recovers from any of six possible illegal states and returns to a normal count sequence within two clock cycles for Xilinx devices, as shown in the following state diagram.



Larger counters are created by connecting the count enable out (CEO) output of the first stage to the CE input of the next stage and connecting the R and clock inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than $n(t_{CE\text{-}TC})$, where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input; use the TC output if it does not.

The counter is asynchronously cleared, output Low, when power is applied.

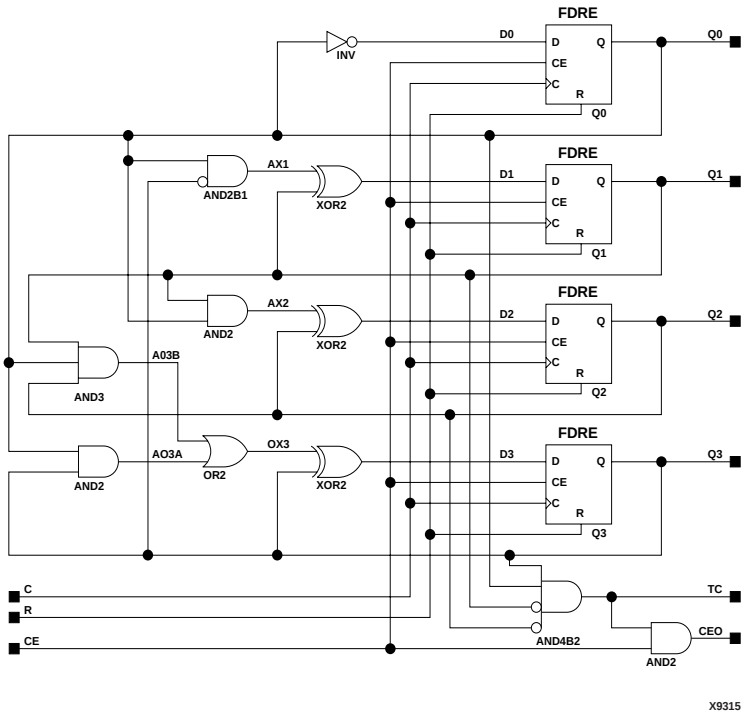
For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs					
R	CE	C	Q3	Q2	Q1	Q0	TC	CEO
1	X	↑	0	0	0	0	0	0
0	1	↑	Inc	Inc	Inc	Inc	TC	CEO
0	0	X	No Change	No Change	No Change	No Change	TC	0
0	1	X	1	0	0	1	1	1

TC = Q3 • !Q2 • !Q1 • Q0

CEO = TC • CE



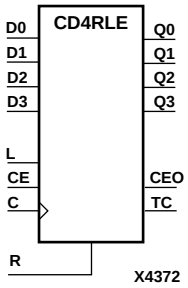
CD4RE Implementation of Spartan-3E

Usage

This design element can be inferred.

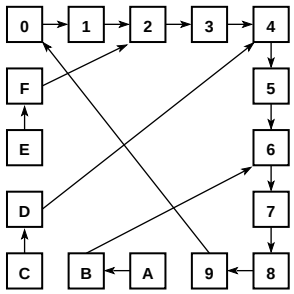
CD4RLE

Macro: 4-Bit Loadable Cascadable BCD Counter with Clock Enable and Synchronous Reset



CD4RLE is a 4-bit (stage), synchronous loadable, resettable, binary-coded-decimal (BCD) counter. The synchronous reset input (R) is the highest priority input. When R is High, all other inputs are ignored; the Q outputs, terminal count (TC), and clock enable out (CEO) go to logic level zero on the Low-to-High clock transitions. The data on the D inputs is loaded into the counter when the load enable input (L) is High during the Low-to-High clock (C) transition. The Q outputs increment when the clock enable input (CE) is High during the Low-to-High clock transition. The counter ignores clock transitions when CE is Low. The TC output is High when Q3 and Q0 are High and Q2 and Q1 are Low.

The counter recovers from any of six possible illegal states and returns to a normal count sequence within two clock cycles, as shown in the following state diagram.



X2355

Larger counters are created by connecting the count enable out (CEO) output of the first stage to the CE input of the next stage and connecting the R, L, and C inputs in parallel. CEO is active (High) when TC and CE are High. The maximum length of the counter is determined by the accumulated CE-to-TC propagation delays versus the clock period. The clock period must be greater than $n(t_{CE\text{-}TC})$, where n is the number of stages and the time $t_{CE\text{-}TC}$ is the CE-to-TC propagation delay of each stage. When cascading counters, use the CEO output if the counter uses the CE input; use the TC output if it does not.

The counter is asynchronously cleared, output Low, when power is applied.

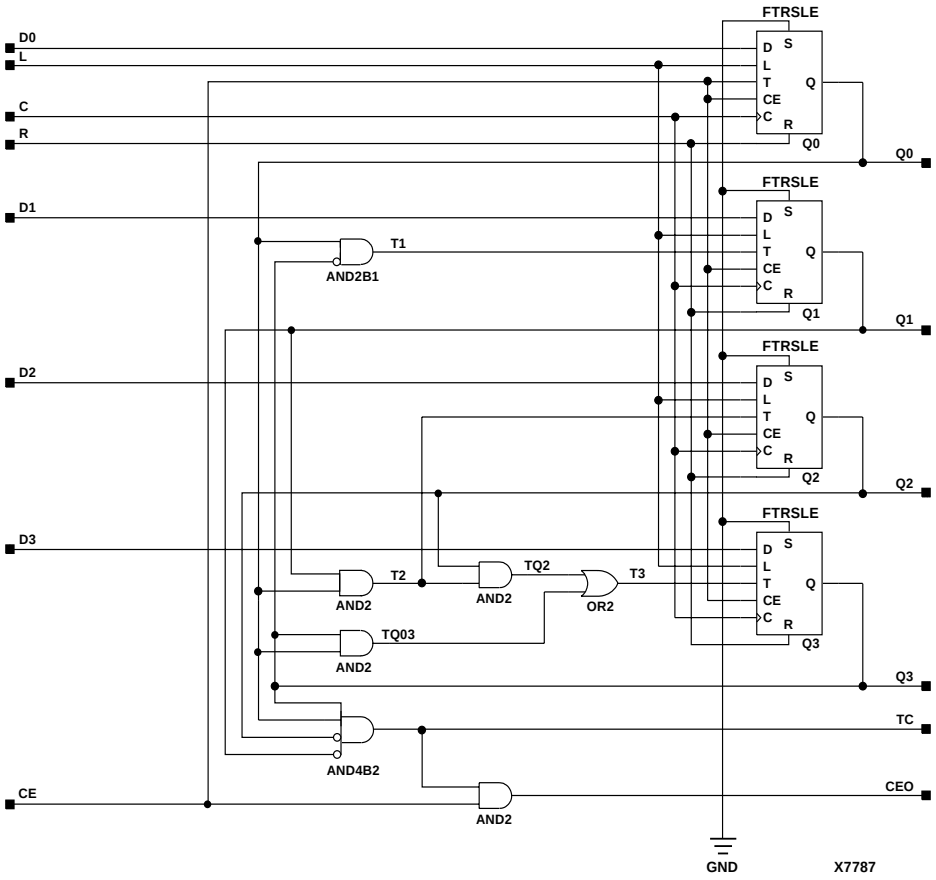
For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs					Outputs					
R	L	CE	D3 – D0	C	Q3	Q2	Q1	Q0	TC	CEO
1	X	X	X	↑	0	0	0	0	0	0
0	1	X	D3 – D0	↑	D3	D	D	D0	TC	CEO
0	0	1	X	↑	Inc	Inc	Inc	Inc	TC	CEO
0	0	0	X	X	No Change	No Change	No Change	No Change	TC	0
0	0	1	X	X	1	0	0	1	1	1

TC = Q3•!Q2•!Q1•Q0

CEO = TC•CE



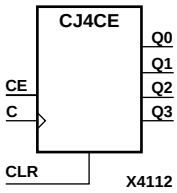
CD4RLE Implementation of Spartan-3E

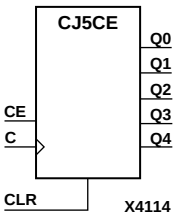
Usage

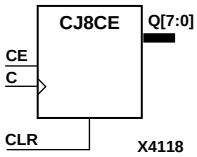
This design element is supported only for schematics *and* instantiation.

CJ4CE, CJ5CE, CJ8CE

Macro: 4-, 5-, 8-Bit Johnson Counters with Clock Enable and Asynchronous Clear







CJ4CE, CJ5CE, and CJ8CE are clearable Johnson / shift counters. The asynchronous clear (CLR) input, when High, overrides all other inputs and causes the data (Q) outputs to go to logic level zero, independent of clock (C) transitions. The counter increments (shifts Q0 to Q1, Q1 to Q2,and so forth) when the clock enable input (CE) is High during the Low-to-High clock transition. Clock transitions are ignored when (CE) is Low.

For CJ4CE, the Q3 output is inverted and fed back to input Q0 to provide continuous counting operation. For CJ5CE, the Q4 output is inverted and fed back to input Q0. For CJ8CE, the Q7 output is inverted and fed back to input Q0.

The counter is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set / reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

CJ4CE Truth Table

Inputs			Outputs			
CLR	CE	C	Q0	Q1	Q2	Q3
1	X	X	0	0	0	0
0	0	X	No Change	No Change	No Change	No Change
0	1	↑	$\overline{q3}$	q0	q1	q2

q = state of referenced output one setup time prior to active clock transition

CJ5CE Truth Table

Inputs			Outputs				
CLR	CE	C	Q0	Q1	Q2	Q3	Q4
1	X	X	0	0	0	0	0
0	0	X	No Change	No Change	No Change	No Change	No Change
0	1	↑	$\overline{q4}$	q0	q1	q2	q3

q = state of referenced output one setup time prior to active clock transition

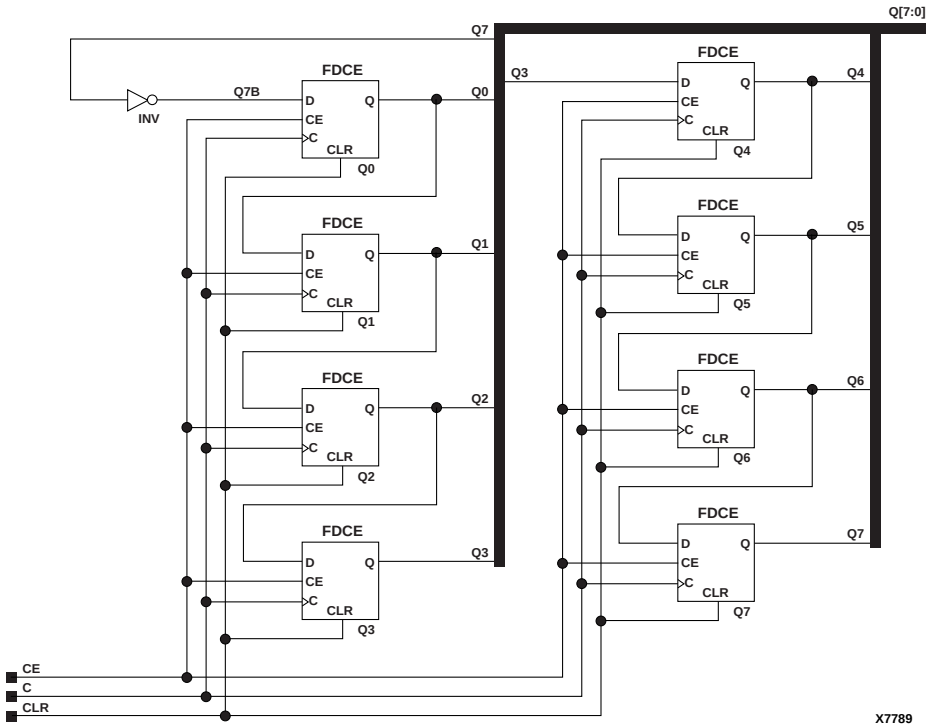
CJ8CE Truth Table

Inputs			Outputs	
CLR	CE	C	Q0	Q1 – Q7
1	X	X	0	0
0	0	X	No Change	No Change

CJ8CE Truth Table

Inputs			Outputs	
CLR	CE	C	Q0	Q1 – Q7
0	1	↑	$\overline{q7}$	q0 – q6

q = state of referenced output one setup time prior to active clock transition



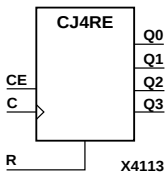
CJ8CE Implementation of Spartan-3E

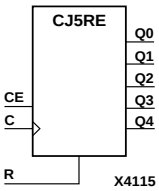
Usage

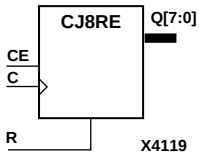
This design element can be inferred but not instantiated.

CJ4RE, CJ5RE, CJ8RE

Macro: 4-, 5-, 8-Bit Johnson Counters with Clock Enable and Synchronous Reset







CJ4RE, CJ5RE, and CJ8RE are resettable Johnson / shift counters. The synchronous reset (R) input, when High, overrides all other inputs and causes the data (Q) outputs to go to logic level zero during the Low-to-High clock (C) transition. The counter increments (shifts Q0 to Q1, Q1 to Q2, and so forth) when the clock enable input (CE) is High during the Low-to-High clock transition. Clock transitions are ignored when CE is Low.

For CJ4RE, the Q3 output is inverted and fed back to input Q0 to provide continuous counting operations. For CJ5RE, the Q4 output is inverted and fed back to input Q0. For CJ8RE, the Q7 output is inverted and fed back to input Q0.

The counter is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set / reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

CJ4RE Truth Table

Inputs			Outputs			
R	CE	C	Q0	Q1	Q2	Q3
1	X	↑	0	0	0	0
0	0	X	No Change	No Change	No Change	No Change
0	1	↑	$\overline{q3}$	q0	q1	q2

q = state of referenced output one setup time prior to active clock transition

CJ5RE Truth Table

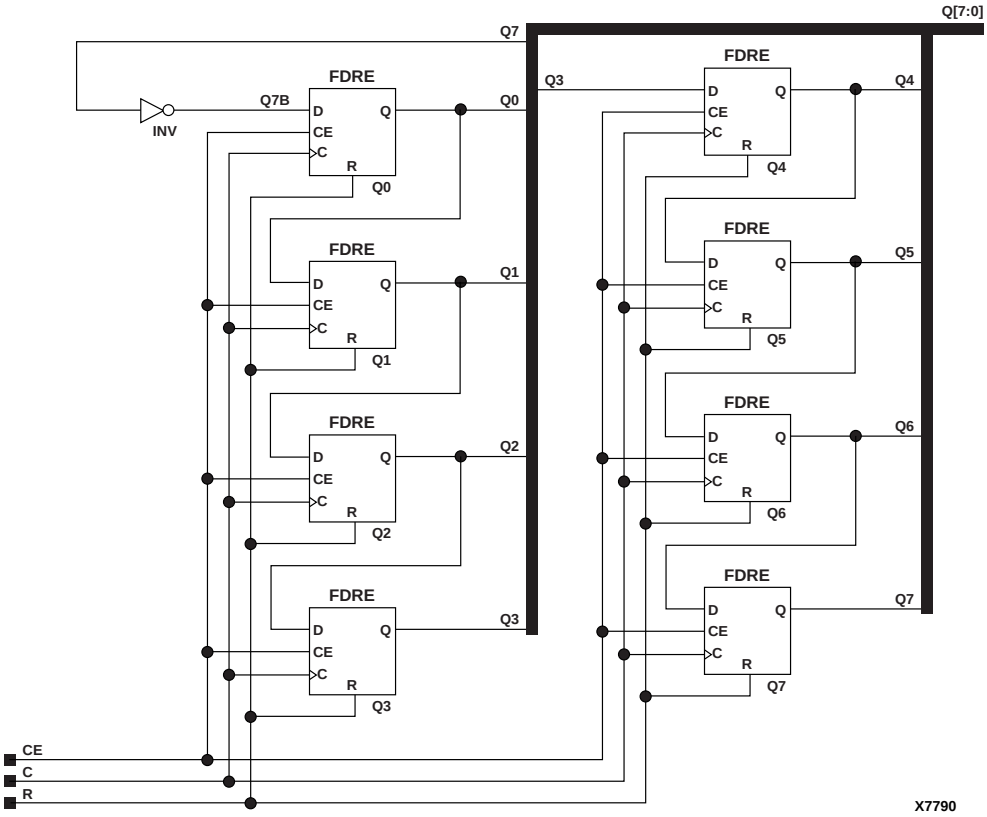
Inputs			Outputs				
R	CE	C	Q0	Q1	Q2	Q3	Q4
1	X	↑	0	0	0	0	0
0	0	X	No Change	No Change	No Change	No Change	No Change
0	1	↑	$\overline{q4}$	q0	q1	q2	q3

q = state of referenced output one setup time prior to active clock transition

CJ8RE Truth Table

Inputs			Outputs	
R	CE	C	Q0	Q1 – Q7
1	X	↑	0	0
0	0	X	No Change	No Change
0	1	↑	$\overline{q7}$	q0 – q6

q = state of referenced output one setup time prior to active clock transition



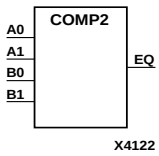
CJ8RE Implementation of Spartan-3E

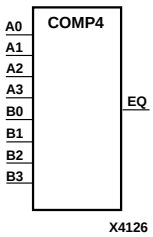
Usage

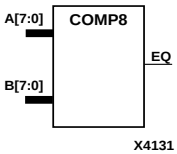
This design element can be inferred but not instantiated.

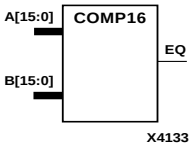
COMP2, 4, 8, 16

Macro: 2-, 4-, 8-, 16-Bit Identity Comparators



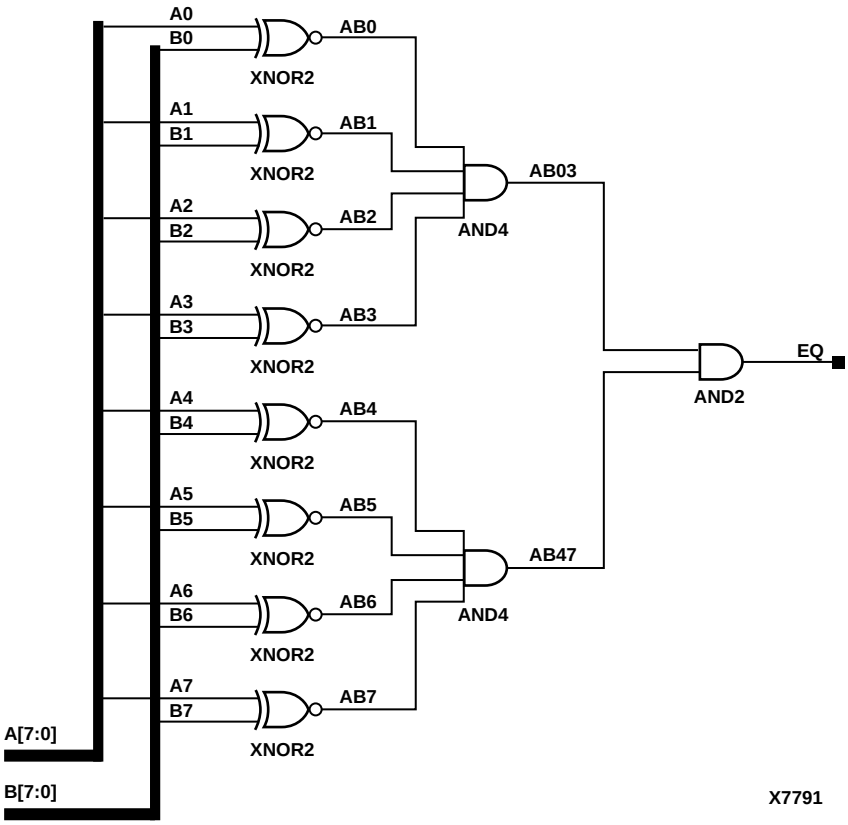






COMP2, COMP4, COMP8, and COMP16 are, respectively, 2-, 4-, 8-, and 16-bit identity comparators. The equal output (EQ) of the COMP2 2-bit, identity comparator is High when the two words A1 – A0 and B1 – B0 are equal. EQ is high for COMP4 when A3 – A0 and B3 – B0 are equal; for COMP8, when A7 – A0 and B7 – B0 are equal; and for COMP16, when A15 – A0 and B15 – B0 are equal.

Equality is determined by a bit comparison of the two words. When any two of the corresponding bits from each word are not the same, the EQ output is Low.



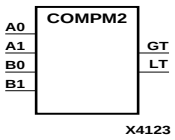
COMP8 Implementation for Spartan-3E

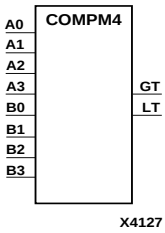
Usage

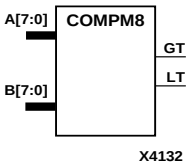
These design elements are inferred rather than instantiated.

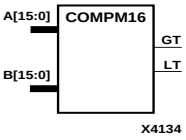
COMPM2, 4, 8, 16

Macro: 2-, 4-, 8-, 16-Bit Magnitude Comparators









COMPM2, COMPM4, COMPM8, and COMPM16 are, respectively, 2-, 4-, 8-, and 16-bit magnitude comparators that compare two positive binary-weighted words.

COMPM2 compares A1 – A0 and B1 – B0, where A1 and B1 are the most significant bits. COMPM4 compares A3 – A0 and B3 – B0, where A3 and B3 are the most significant bits. COMPM8 compares A7 – A0 and B7 – B0, where A7 and B7 are the most significant bits. COMPM16 compares A15 – A0 and B15 – B0, where A15 and B15 are the most significant bits.

The greater-than output (GT) is High when A>B, and the less-than output (LT) is High when A<B. When the two words are equal, both GT and LT are Low. Equality can be measured with this macro by comparing both outputs with a NOR gate.

COMPM2 Truth Table

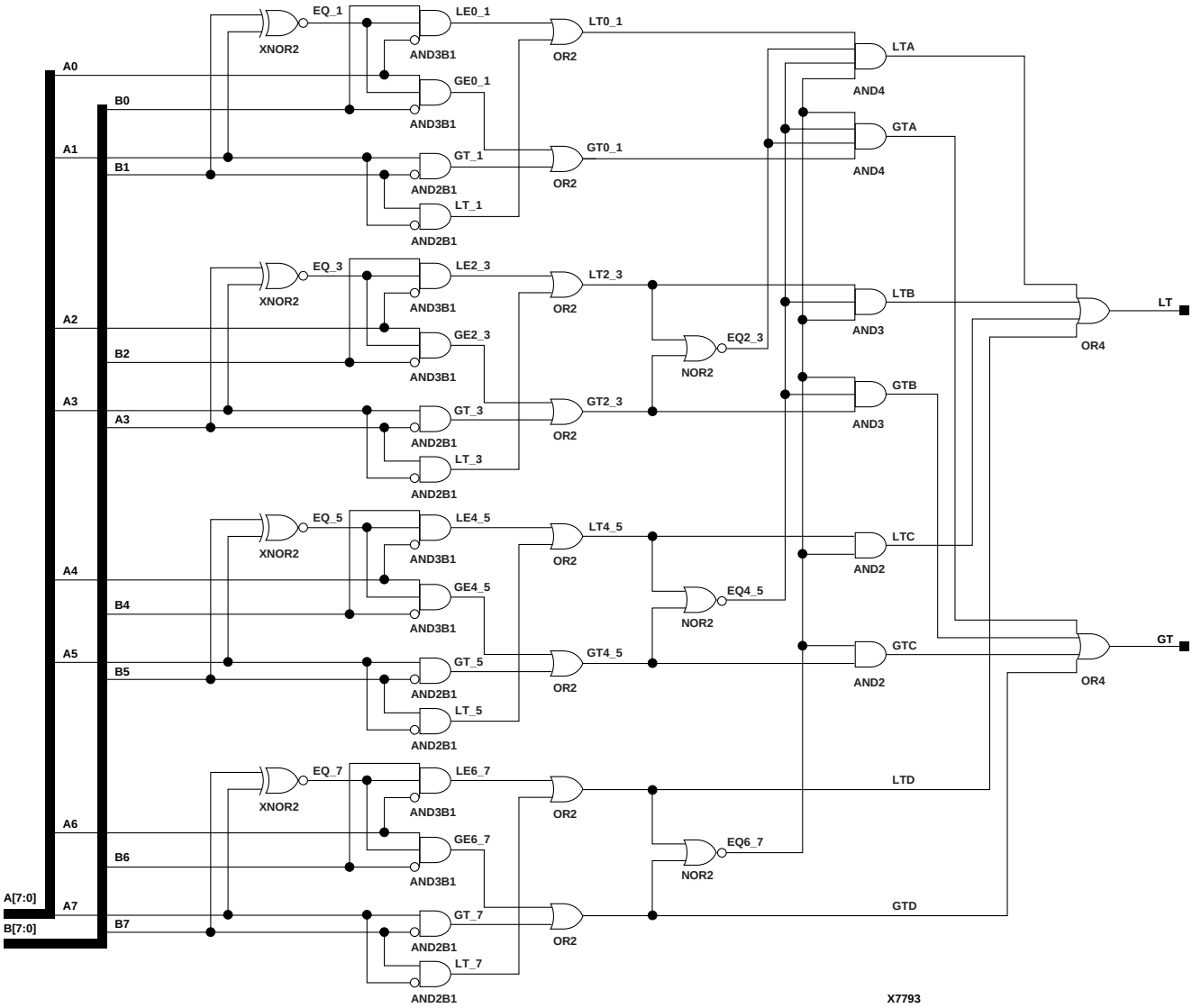
Inputs				Outputs	
A1	B1	A0	B0	GT	LT
0	0	0	0	0	0
0	0	1	0	1	0
0	0	0	1	0	1
0	0	1	1	0	0
1	1	0	0	0	0
1	1	1	0	1	0
1	1	0	1	0	1
1	1	1	1	0	0
1	0	X	X	1	0
0	1	X	X	0	1

COMPM4 Truth Table

Inputs				Outputs	
A3, B3	A2, B2	A1, B1	A0, B0	GT	LT
A3>B3	X	X	X	1	0
A3<B3	X	X	X	0	1
A3=B3	A2>B2	X	X	1	0
A3=B3	A2<B2	X	X	0	1
A3=B3	A2=B2	A1>B1	X	1	0
A3=B3	A2=B2	A1<B1	X	0	1
A3=B3	A2=A2	A1=B1	A0>B0	1	0
A3=B3	A2=B2	A1=B1	A0<B0	0	1
A3=B3	A2=B2	A1=B1	A0=B0	0	0

COMPM8 Truth Table (also representative of COMPM16)

Inputs								Outputs	
A7, B7	A6, B6	A5, B5	A4, B4	A3, B3	A2, B2	A1, B1	A0, B0	GT	LT
A7>B7	X	X	X	X	X	X	X	1	0
A7<B7	X	X	X	X	X	X	X	0	1
A7=B7	A6>B6	X	X	X	X	X	X	1	0
A7=B7	A6<B6	X	X	X	X	X	X	0	1
A7=B7	A6=B6	A5>B5	X	X	X	X	X	1	0
A7=B7	A6=B6	A5<B5	X	X	X	X	X	0	1
A7=B7	A6=B6	A5=B5	A4>B4	X	X	X	X	1	0
A7=B7	A6=B6	A5=B5	A4<B4	X	X	X	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3>B3	X	X	X	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3<B3	X	X	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2>B2	X	X	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2<B2	X	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1>B1	X	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1<B1	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1=B1	A0>B0	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1=B1	A0<B0	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1=B1	A0=B0	0	0



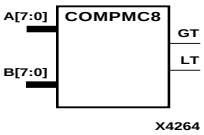
COMPM8 Implementation for Spartan-3E

Usage

These design elements are supported only for schematics.

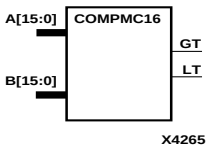
COMPMC8, 16

Macro: 8-, 16-Bit Magnitude Comparators



COMPMC8 is an 8-bit, magnitude comparator that compares two positive binary-weighted words A7 – A0 and B7 – B0, where A7 and B7 are the most significant bits. COMPMC16 is a 16-bit, magnitude comparator that compares two positive binary-weighted words A15 – A0 and B15 – B0, where A15 and B15 are the most significant bits.

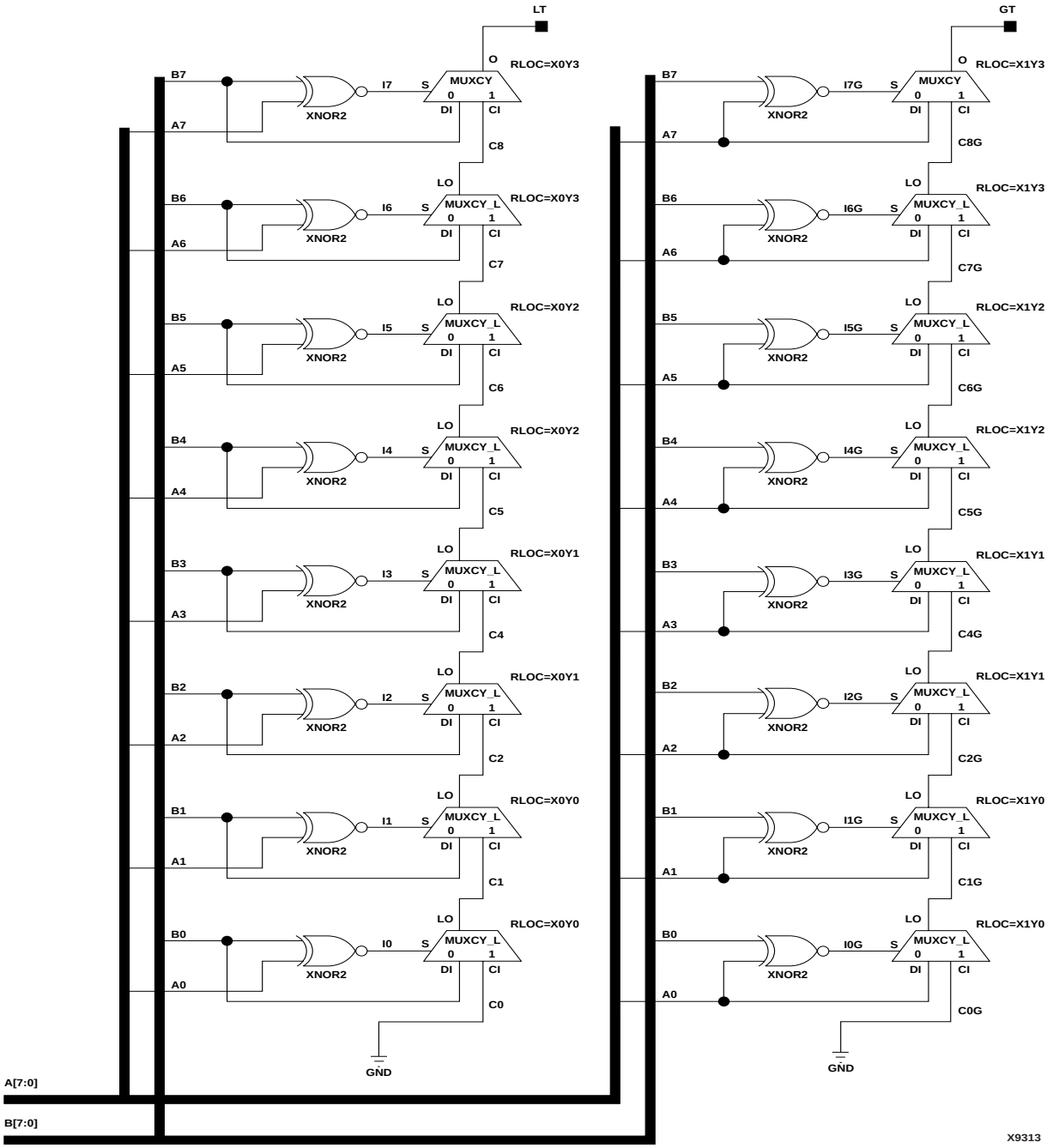
These comparators are implemented using carry logic with relative location constraints to ensure efficient logic placement.



The greater-than output (GT) is High when A>B, and the less-than output (LT) is High when A<B. When the two words are equal, both GT and LT are Low. Equality can be flagged with this macro by connecting both outputs to a NOR gate.

COMPMC8 Truth Table (also representative of COMPMC16)

Inputs								Outputs	
A7, B7	A6, B6	A5, B5	A4, B4	A3, B3	A2, B2	A1, B1	A0, B0	GT	LT
A7>B7	X	X	X	X	X	X	X	1	0
A7<B7	X	X	X	X	X	X	X	0	1
A7=B7	A6>B6	X	X	X	X	X	X	1	0
A7=B7	A6<B6	X	X	X	X	X	X	0	1
A7=B7	A6=B6	A5>B5	X	X	X	X	X	1	0
A7=B7	A6=B6	A5<B5	X	X	X	X	X	0	1
A7=B7	A6=B6	A5=B5	A4>B4	X	X	X	X	1	0
A7=B7	A6=B6	A5=B5	A4<B4	X	X	X	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3>B3	X	X	X	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3<B3	X	X	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2>B2	X	X	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2<B2	X	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1>B1	X	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1<B1	X	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1=B1	A0>B0	1	0
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1=B1	A0<B0	0	1
A7=B7	A6=B6	A5=B5	A4=B4	A3=B3	A2=B2	A1=B1	A0=B0	0	0



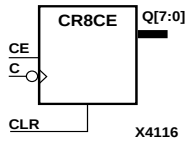
COMPMC8 Implementation for Spartan-3E

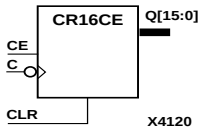
Usage

These design elements are supported only for schematics.

CR8CE, CR16CE

Macro: 8-, 16-Bit Negative-Edge Binary Ripple Counters with Clock Enable and Asynchronous Clear





CR8CE and CR16CE are 8-bit and 16-bit, cascadable, clearable, binary, ripple counters. The asynchronous clear (CLR), when High, overrides all other inputs and causes the Q outputs to go to logic level zero. The counter increments when the clock enable input (CE) is High during the High-to-Low clock (C) transition. The counter ignores clock transitions when CE is Low.

Larger counters can be created by connecting the last Q output (Q7 for CR8CE, Q15 for CR16CE) of the first stage to the clock input of the next stage. CLR and CE inputs are connected in parallel. The clock period is not affected by the overall length of a ripple counter. The overall clock-to-output propagation is $n(t_{C\_Q})$, where n is the number of stages and the time t_{C_Q} is the C-to-Qz propagation delay of each stage.

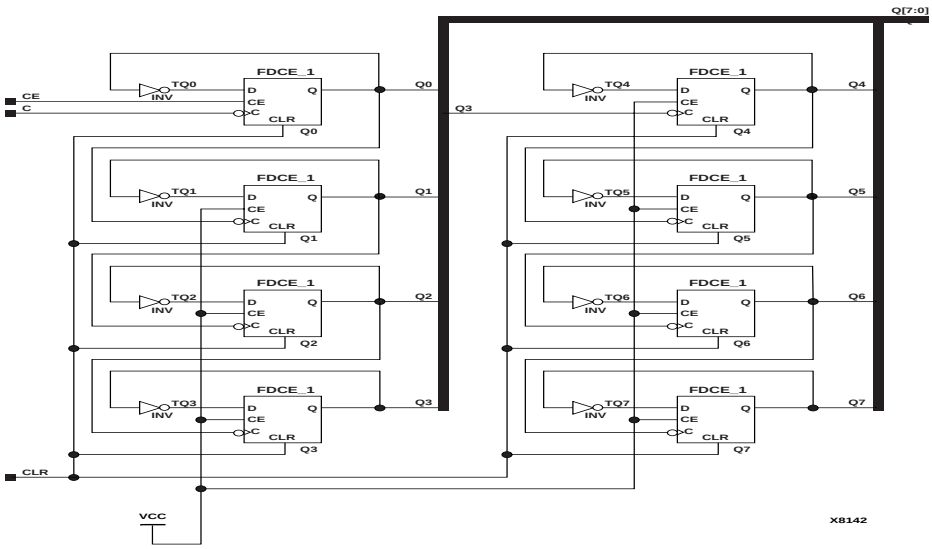
The counter is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
CLR	CE	C	Qz – Q0
1	X	X	0
0	0	X	No Change
0	1	↓	Inc

z = 7 for CR8CE; z = 15 for CR16CE.



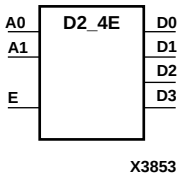
CR8CE Implementation of Spartan-3E

Usage

These design elements are inferred rather than instantiated.

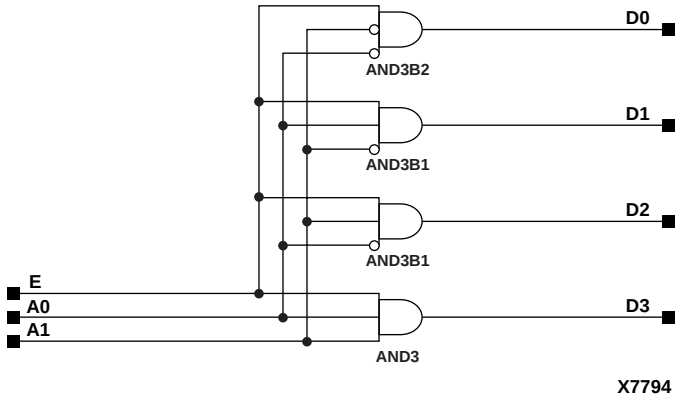
D2_4E

Macro: 2- to 4-Line Decoder/Demultiplexer with Enable



When the enable (E) input of the D2_4E decoder / demultiplexer is High, one of four active-High outputs (D3 – D0) is selected with a 2-bit binary address (A1 – A0) input. The non-selected outputs are Low. Also, when the E input is Low, all outputs are Low. In demultiplexer applications, the E input is the data input.

Inputs			Outputs			
A1	A0	E	D3	D2	D1	D0
X	X	0	0	0	0	0
0	0	1	0	0	0	1
0	1	1	0	0	1	0
1	0	1	0	1	0	0
1	1	1	1	0	0	0



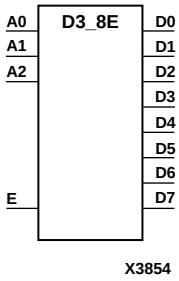
D2_4E Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

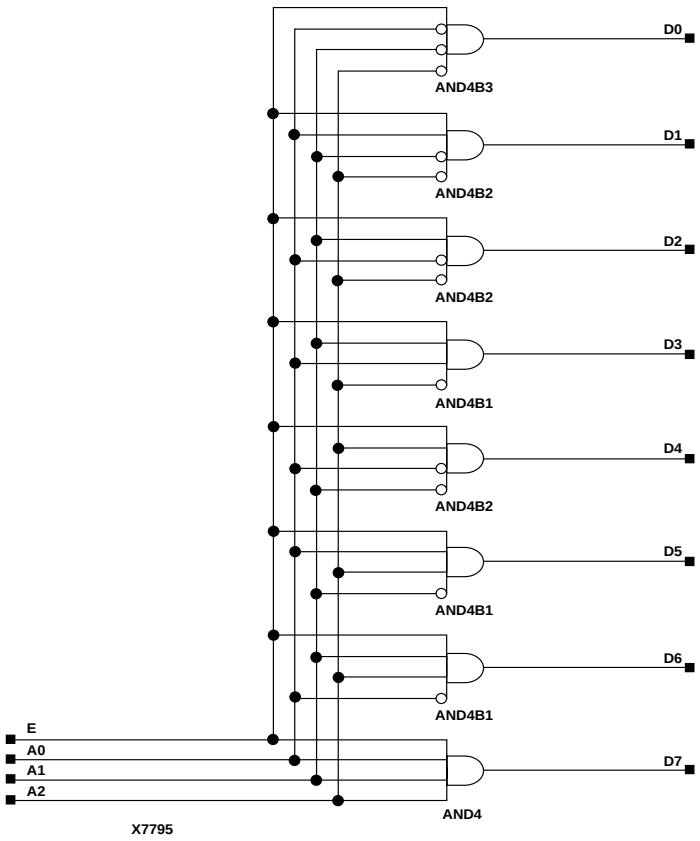
D3_8E

Macro: 3- to 8-Line Decoder/Demultiplexer with Enable



When the enable (E) input of the D3_8E decoder / demultiplexer is High, one of eight active-High outputs (D7 – D0) is selected with a 3-bit binary address (A2 – A0) input. The non-selected outputs are Low. Also, when the E input is Low, all outputs are Low. In demultiplexer applications, the E input is the data input.

Inputs				Outputs							
A2	A1	A0	E	D7	D6	D5	D4	D3	D2	D1	D0
X	X	X	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	1
0	0	1	1	0	0	0	0	0	0	1	0
0	1	0	1	0	0	0	0	0	1	0	0
0	1	1	1	0	0	0	0	1	0	0	0
1	0	0	1	0	0	0	1	0	0	0	0
1	0	1	1	0	0	1	0	0	0	0	0
1	1	0	1	0	1	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0



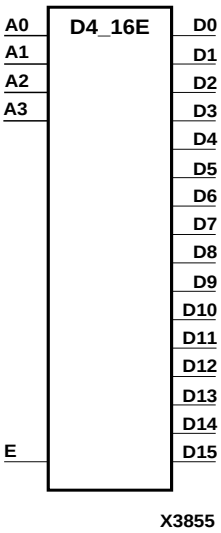
D3_8E Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

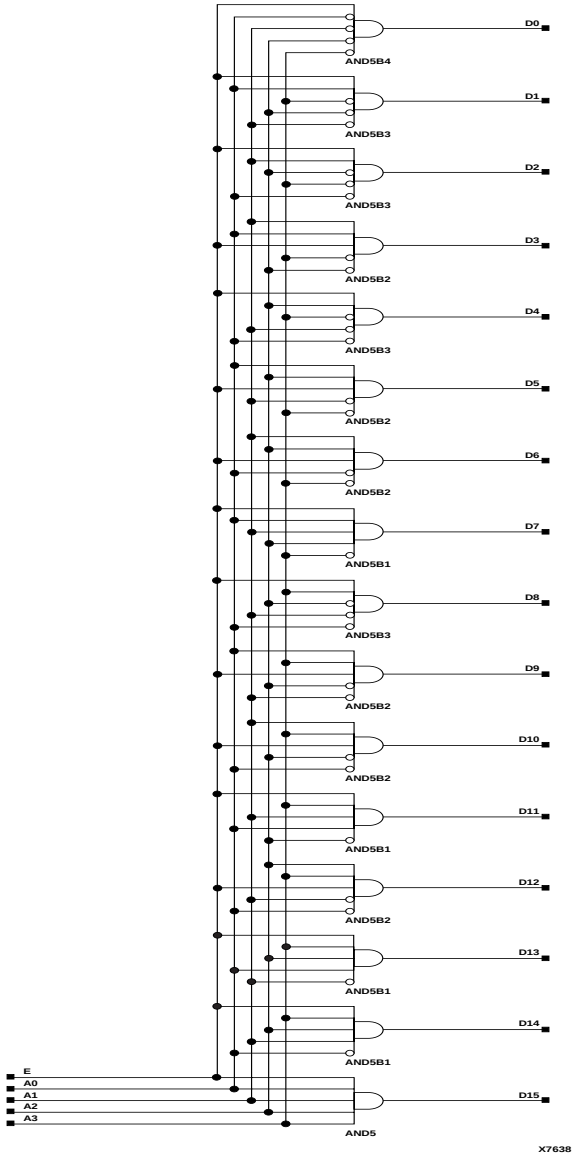
D4_16E

Macro: 4- to 16-Line Decoder/Demultiplexer with Enable



When the enable (E) input of the D4_16E decoder / demultiplexer is High, one of 16 active-High outputs (D15 – D0) is selected with a 4-bit binary address (A3 – A0) input. The non-selected outputs are Low. Also, when the E input is Low, all outputs are Low. In demultiplexer applications, the E input is the data input.

See "D3_8E" for a representative truth table derivation.



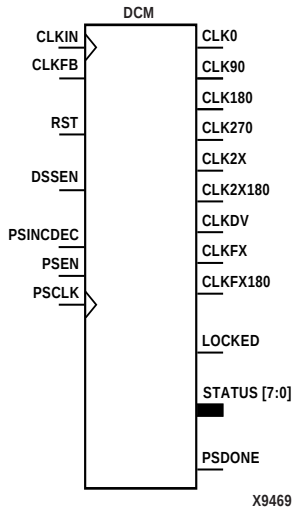
D4_16E Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

DCM

Primitive: Digital Clock Manager



DCM is a digital clock manager that provides multiple functions. It can implement a clock delay locked loop, a digital frequency synthesizer, digital phase shifter, and a digital spread spectrum.

Note: All unused inputs must be driven Low, automatically tying the inputs Low if they are unused.

Clock Delay Locked Loop (DLL)

DCM includes a clock delay locked loop used to minimize clock skew for Spartan-3E devices. DCM synchronizes the clock signal at the feedback clock input (CLKFB) to the clock signal at the input clock (CLKIN). The locked output (LOCKED) is high when the two signals are in phase. The signals are considered to be in phase when their rising edges are within a specified time (ps) of each other.

DCM supports two frequency modes for the DLL. By default, the DLL_FREQUENCY_MODE attribute is set to Low and the frequency of the clock signal at the CLKIN input must be in the Low (DLL_CLKIN_MIN_LF to DLL_CLKIN_MAX_LF) frequency range (MHz). In Low frequency mode, the CLK0, CLK90, CLK180, CLK270, CLK2X, CLKDV, and CLK2X180 outputs are available.

When the DLL_FREQUENCY_MODE attribute is set to High, the frequency of the clock signal at the CLKIN input must be in the High (DLL_CLKIN_MIN_HF to DLL_CLKIN_MAX_HF) frequency range (MHz). In High frequency mode, only the CLK0, CLK180, and CLKDV outputs are available.

On-chip synchronization is achieved by connecting the CLKFB input to a point on the global clock network driven by a BUFG, a global clock buffer. The BUFG connected to the CLKFB input of the DCM must be sourced from either the CLK0 or CLK2X outputs of the same DCM. The CLKIN input should be connected to the output of an IBUFG, with the IBUFG input connected to a pad driven by the system clock.

Off-chip synchronization is achieved by connecting the CLKFB input to the output of an IBUFG, with the IBUFG input connected to a pad. Either the CLK0 or CLK2X output can be used but not both. The CLK0 or CLK2X must be connected to the input of OBUF, an output buffer. The CLK_FEEDBACK attribute controls whether the CLK0 output, the default, or the CLK2X output is the source of the CLKFB input.

The duty cycle of the CLK0 output is 50-50 unless the DUTY_CYCLE_CORRECTION attribute is set to FALSE, in which case the duty cycle is the same as that of the CLKIN input. The duty cycle of the phase shifted outputs (CLK90, CLK180, and CLK270) is the same as that of the CLK0 output. The duty cycle of the CLK2X, CLK2X180, and CLKDV outputs is 50-50 unless CLKDV_DIVIDE is a non-integer and the DLL_FREQUENCY_MODE is High (see “CLKDV_DIVIDE,” in the *Constraints Guide*

for details). The frequency of the CLKDV output is determined by the value assigned to the CLKDV_DIVIDE attribute.

DCM Clock Delay Lock Loop Outputs

Output	Description
CLK0	Clock at 1x CLKIN frequency
CLK180	Clock at 1x CLK0 frequency, shifted 180º with regards to CLK0
CLK270*	Clock at 1x CLK0 frequency, shifted 270º with regards to CLK0
CLK2X*	Clock at 2x CLK0 frequency, in phase with CLK0
CLK2X180*	Clock at 2x CLK0 frequency shifted 180º with regards to CLK2X
CLK90*	Clock at 1x CLK0 frequency, shifted 90º with regards to CLK0
CLKDV	Clock at (1/n) x CLK0 frequency, where n=CLKDV_DIVIDE value. CLKDV is in phase with CLK0.
LOCKED	All enabled DCM features locked.

* The CLK90, CLK270, CLK2X, and CLK2X180 outputs are *not* available if the DLL_FREQUENCY_MODE is set to High.

Digital Frequency Synthesizer (DFS)

The CLKFX and CLKFX180 outputs in conjunction with the CLKFX_MULTIPLY and CLKFX_DIVIDE attributes provide a frequency synthesizer that can be any multiple or division of CLKIN. CLKFX and CLKIN are in phase every CLKFX_MULTIPLY cycles of CLKFX and every CLKFX_DIVIDE cycles of CLKIN when a feedback is provided to the CLKFB input of the DLL. The frequency of CLKFX is defined by the following equation.

$$Frequency_{CLKFX} = (CLKFX\_MULTIPLY\_value / CLKFX\_DIVIDE\_value) * Frequency_{CLKIN}$$

Both the CLKFX or CLKFX180 output can be used simultaneously.

CLKFX180 is 1x the CLKFX frequency, shifted 180º with regards to CLKFX. CLKFX and CLKFX180 always have a 50/50 duty cycle.

The DFS_FREQUENCY_MODE attribute specifies the allowable input clock and output clock frequency ranges.

The CLK_FEEDBACK attribute set to NONE will cause the DCM to be in the Digital Frequency Synthesizer mode. The CLKFX and CLKFX180 will be generated without phase correction with respect to CLKIN.

Digital Phase Shifter (DPS)

The phase shift (skew) between the rising edges of CLKIN and CLKFB may be configured as a fraction of the CLKIN period with the PHASE_SHIFT attribute. This allows the phase shift to remain constant as ambient conditions change. The CLKOUT_PHASE_SHIFT attribute controls the use of the PHASE_SHIFT value. By default, the CLKOUT_PHASE_SHIFT attribute is set to NONE and the PHASE_SHIFT attribute has no effect.

By creating skew between CLKIN and CLKFB, all DCM output clocks are phase shifted by the amount of the skew.

When the CLKOUT_PHASE_SHIFT attribute is set to FIXED, the skew set by the PHASE_SHIFT attribute is used at configuration for the rising edges of CLKIN and CLKFB. The skew remains constant.

When the CLKOUT_PHASE_SHIFT attribute is set to VARIABLE, the skew set at configuration is used as a starting point and the skew value can be changed dynamically during operation using the PS* signals. This digital phase shifter feature is controlled by a synchronous interface. The inputs PSEN (phase shift enable) and PSINCDEC (phase shift increment / decrement) are set up to the rising edge of PSCLK (phase shift clock). The PSDONE (phase shift done) output is clocked with the rising edge of PSCLK (the phase shift clock). PSDONE must be connected to implement the complete synchronous interface. The rising-edge skew between CLKIN and CLKFB may be dynamically adjusted after the LOCKED output goes High.

The PHASE_SHIFT attribute value specifies the initial phase shift amount when the device is configured. Then the PHASE_SHIFT value is changed one unit when PSEN is activated for one period of PSCLK. The PHASE_SHIFT value is incremented when PSINCDEC is High and decremented when PSINCDEC is Low during the period that PSEN is High. When the DCM completes an increment or decrement operation, the PSDONE output goes High for a single PSCLK cycle to indicate the operation is complete. At this point the next change may be made. When RST (reset) is High, the PHASE_SHIFT attribute value is reset to the skew value set at configuration.

If CLKOUT_PHASE_SHIFT is FIXED or NONE, the PSEN, PSINCDEC, and PSCLK inputs must be tied to GND. The program will automatically tie the inputs to GND if they are not connected by the user.

Additional Status Bits

The STATUS output bits return the following information.

DCM Additional Status Bits

Bit	Description
0	Phase Shift Overflow* 1 = |PHASE_SHIFT| > 255
1	DLL CLKIN stopped** 1 = CLKIN stopped toggling
2	DLL CLKFX stopped 1 = CLKFX stopped toggling
3	No
4	No
5	No
6	No
7	No

* Phase Shift Overflow will also go high if the end of the phase shift delay line is reached (see the product data sheet for the most current value of the maximum shifting delay).

** If only the DFS outputs are used (CLKFX & CLKFX180), this status bit will not go high if CLKIN stops.

LOCKED

When LOCKED is high, all enabled signals are locked.

RST

The master reset input (RST) resets DCM to its initial (power-on) state. The signal at the RST input is asynchronous and must be held High for 2ns.

Usage

This component is generally instantiated in the code as it cannot be easily inferred in synthesis tools. Some synthesis tools may allow inference via an attribute. See your synthesis tool documentation.

Available Attributes

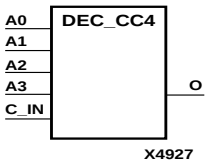
Attribute	Type	Allowed Values	Default
CLKDV_DIVIDE	2-Bit Floating	2.0	1.5,2.0,2.5,3.0,3.5,4.0,4.5,5.0,5.5,6.0,6.5, 7.0,7.5,8.0,9.0,10.0,11.0,12.0,13.0,14.0,15.0 or 16.0
CLKFX_DIVIDE	1-Bit Binary	1.0	1 to 32
CLKFX_MULTI-PLY	4-Bit Binary	4.0	1 to 32
CLKIN_DIVIDE_BY_2	Boolean	FALSE	TRUE
FALSE			
CLKIN_PERIOD		0.0	
CLKOUT_PHASE_SHIFT	Real	NONE	NONE
FIXED			
VARIABLE			
CLK_FEEDBACK	Real	1X	NONE
1X			
2X			
DESKEW_ADJUST	Real	SYSTEM_SYNCHRONOUS	SOURCE_SYNCHRONOUS
SYSTEM_SYNCHRONOUS			
DFS_FREQUENCY_MODE	Real	LOW	HIGH
LOW			
DLL_FREQUENCY_MODE	Real	LOW	HIGH
LOW			
DUTY_CYCLE_CORRECTION	Boolean	TRUE	TRUE
FALSE			
FACTORY_JF	16-Bit Binary	16'hC080	
PHASE_SHIFT	Integer	0	-255 to 255
STARTUP_WAIT	Boolean	FALSE	TRUE
FALSE			

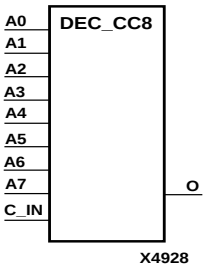
For More Information

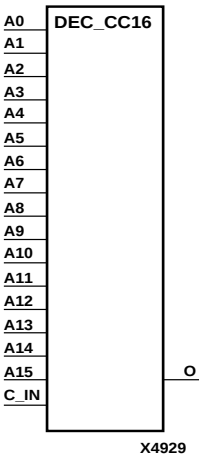
Consult the Spartan-3E Data Sheets.

DEC_CC4, 8, 16

Macro: 4-, 8-, 16-Bit Active Low Decoders



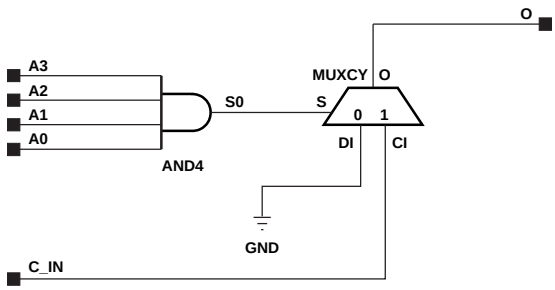




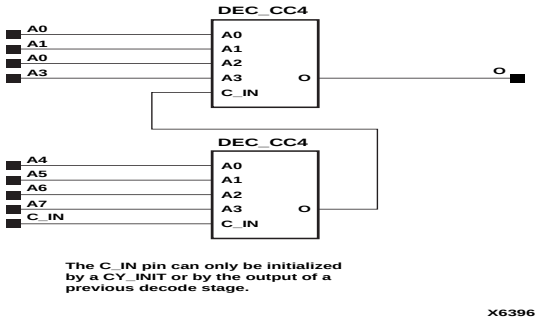
These decoders are used to build wide-decoder functions. They are implemented by cascading CY_MUX elements driven by lookup tables (LUTs). The CIN pin can only be driven by the output (O) of a previous decode stage. When one or more of the inputs (A) are Low, the output is Low. When all the inputs are High and the CIN input is High, the output is High. You can decode patterns by adding inverters to inputs.

Inputs					Outputs
A0	A1	...	Az	C_IN	O
1	1	1	1	1	1
X	X	X	X	0	0
0	X	X	X	X	0
X	0	X	X	X	0
X	X	X	0	X	0

z = 3 for DEC_CC4; z = 7 for DEC_CC8; z = 15 for DEC_CC16



DEC_CC4 Implementation for Spartan-3E



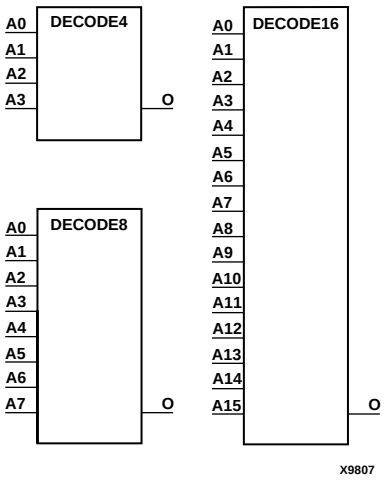
DEC_CC8 Implementation for Spartan-3E

Usage

DEC_CC4 cannot be directly inferred or instantiated. The proper way to use a DEC_CC4 is to infer the primitive components that make up the DEC_CC4.

DECODE4, 8, 16

Macro: 4-, 8-, 16-Bit Active-Low Decoders



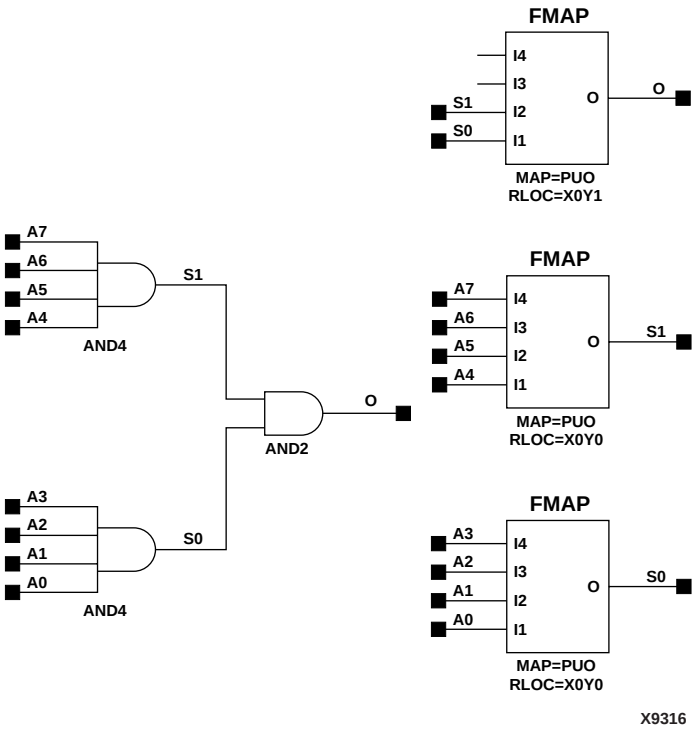
DECODE Representations

In Spartan-3E, decoders are implemented using combinations of LUTs and MUXCYs.

Inputs				Outputs*
A0	A1	...	Az	O
1	1	1	1	1
0	X	X	X	0
X	0	X	X	0
X	X	X	0	0

z = 3 for DECODE4, z = 7 for DECODE8; z = 15 for DECODE16

*A pull-up resistor must be connected to the output to establish High-level drive current.



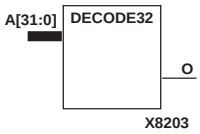
DECODE8 Implementation Spartan-3E

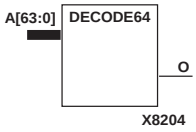
Usage

These design elements are inferrred rather than instantiated.

DECODE32, 64

Macro: 32- and 64-Bit Active-Low Decoders





DECODE32 and DECODE64 are 32- and 64-bit active-low decoders. These decoders are implemented using combinations of LUTs and MUXCYs.

See “DECODE4, 8, 16” for a representative schematic.

Inputs				Outputs
A0	A1	…	Az	O
1	1	1	1	1
0	X	X	X	0
X	0	X	X	0
X	X	X	0	0

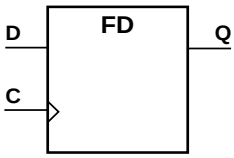
z = 31 for DECODE32, z = 63 for DECODE64

Usage

These design elements are inferred rather than instantiated.

FD

Primitive: D Flip-Flop



X3715

FD is a single D-type flip-flop with data input (D) and data output (Q). The data on the D inputs is loaded into the flip-flop during the Low-to-High clock (C) transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol..

Inputs		Outputs
D	C	Q
0	↑	0
1	↑	1

Usage

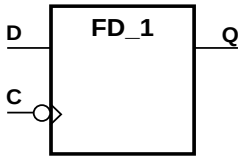
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

FD_1

Primitive: D Flip-Flop with Negative-Edge Clock



X3726

FD_1 is a single D-type flip-flop with data input (D) and data output (Q). The data on the (D) input is loaded into the flip-flop during the High-to-Low clock (C) transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs		Outputs
D	**C**	**Q**
D	↓	D

Usage

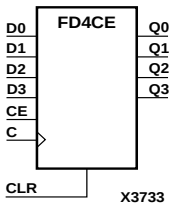
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

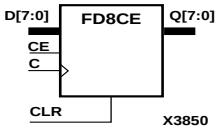
Available Attributes.

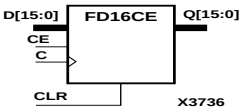
Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0	0 or 1	Sets the initial value of Q output after configuration

FD4CE, FD8CE, FD16CE

Macro: 4-, 8-, 16-Bit Data Registers with Clock Enable and Asynchronous Clear







FD4CE, FD8CE, and FD16CE are, respectively, 4-, 8-, and 16-bit data registers with clock enable and asynchronous clear. When clock enable (CE) is High and asynchronous clear (CLR) is Low, the data on the data inputs (D) is transferred to the corresponding data outputs (Q) during the Low-to-High clock (C) transition. When CLR is High, it overrides all other inputs and resets the data outputs (Q) Low. When CE is Low, clock transitions are ignored.

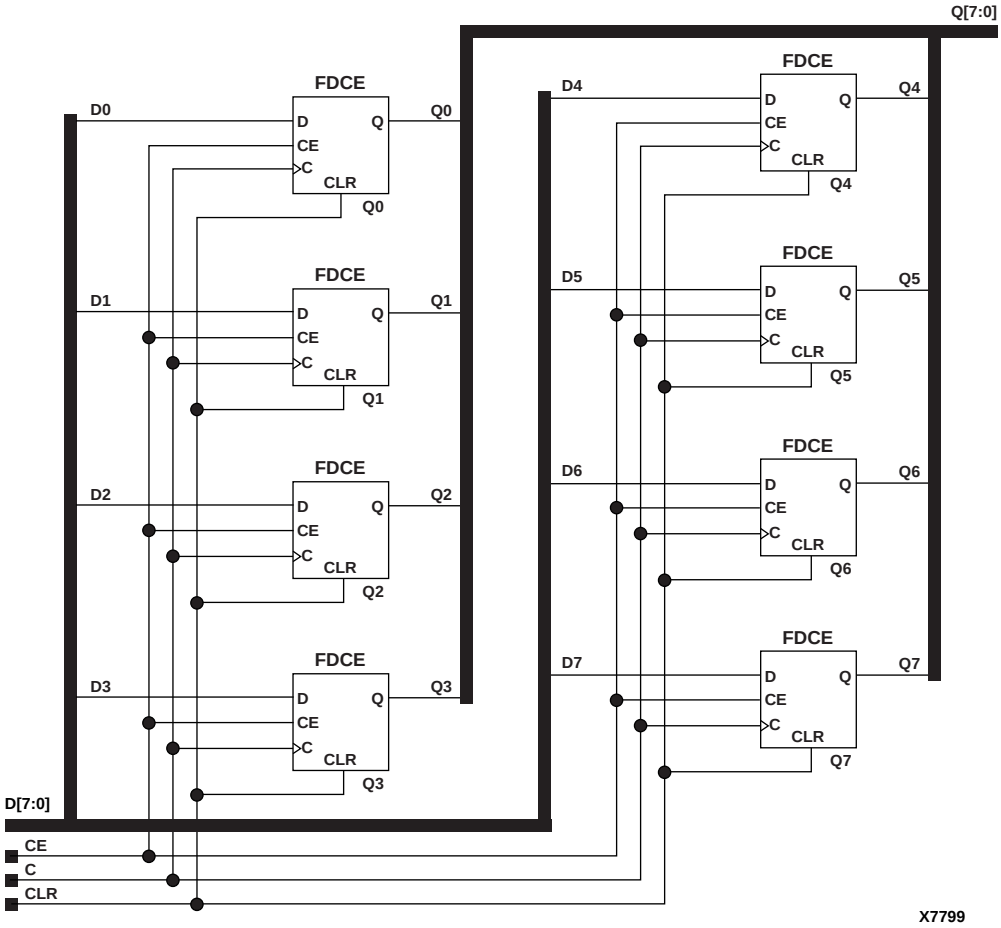
The flip-flops are asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
CLR	CE	Dz – D0	C	Qz – Q0
1	X	X	X	0
0	0	X	X	No Change
0	1	Dn	↑	Dn

z = 3 for FD4CE; z = 7 for FD8CE; z = 15 for FD16CE.



FD8CE Implementation of Spartan-3E

Usage

These design elements are inferred rather than instantiated.

Available Attributes

FD4CE.

Attribute	Type	Allowed Values	Default	Description
INIT	4-Bit Binary	4 bits	4-bit binary	Sets the initial value of Q output after configuration

FD8CE

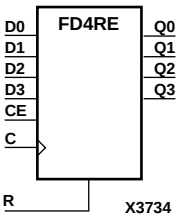
Attribute	Type	Allowed Values	Default	Description
INIT	8-Bit Binary	8 bits	8-bit binary	Sets the initial value of Q output after configuration

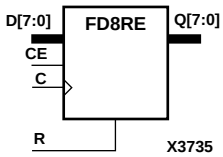
FD16CE

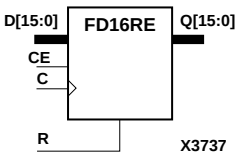
Attribute	Type	Allowed Values	Default	Description
INIT	16-Bit Binary	16 bits	16-bit binary	Sets the initial value of Q output after configuration

FD4RE, FD8RE, FD16RE

Macro: 4-, 8-, 16-Bit Data Registers with Clock Enable and Synchronous Reset







FD4RE, FD8RE, and FD16RE are, respectively, 4-, 8-, and 16-bit data registers. When the clock enable (CE) input is High, and the synchronous reset (R) input is Low, the data on the data inputs (D) is transferred to the corresponding data outputs (Q0) during the Low-to-High clock (C) transition. When R is High, it overrides all other inputs and resets the data outputs (Q) Low on the Low-to-High clock transition. When CE is Low, clock transitions are ignored.

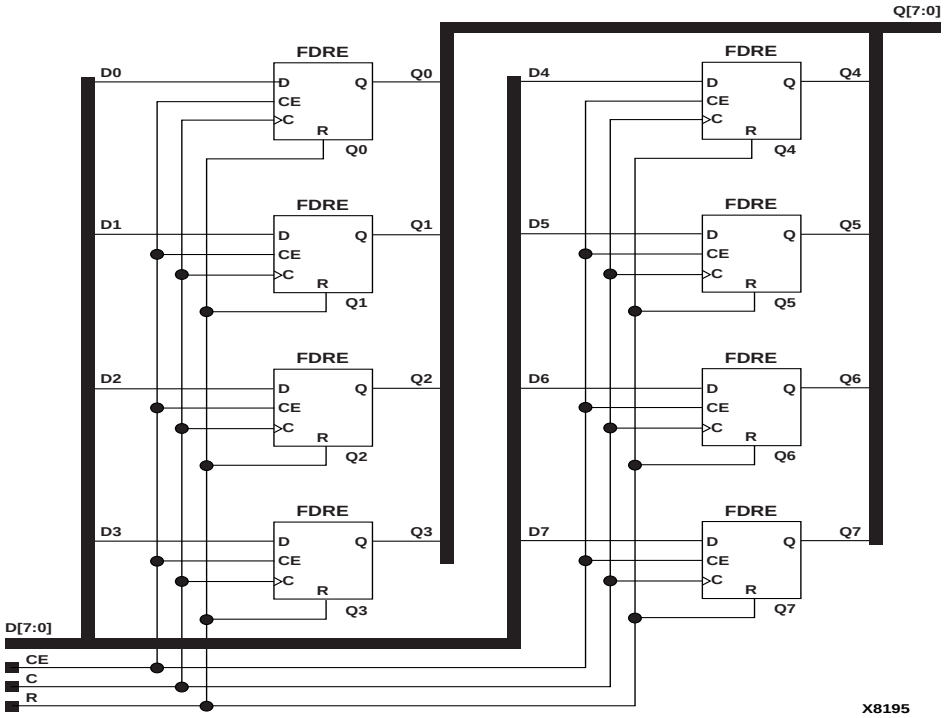
The flip-flops are asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
R	CE	Dz – D0	C	Qz – Q0
1	X	X	↑	0
0	0	X	X	No Change
0	1	Dn	↑	Dn

z = 3 for FD4RE; z = 7 for FD8RE; z = 15 for FD16RE



FD8RE Implementation of Spartan-3E

Usage

These design elements are inferred rather than instantiated.

Available Attributes

FD4RE

Attribute	Type	Allowed Values	Default	Description
INIT	4-Bit Binary	4 bits	4-bit binary	Sets the initial value of Q output after configuration

FD8RE

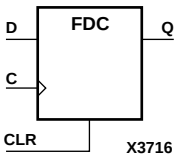
Attribute	Type	Allowed Values	Default	Description
INIT	8-Bit Binary	8 bits	8-bit binary	Sets the initial value of Q output after configuration

FD16RE

Attribute	Type	Allowed Values	Default	Description
INIT	16-Bit Binary	16 bits	16-bit binary	Sets the initial value of Q output after configuration

FDC

Primitive: D Flip-Flop with Asynchronous Clear



FDC is a single D-type flip-flop with data (D) and asynchronous clear (CLR) inputs and data output (Q). The asynchronous CLR, when High, overrides all other inputs and sets the (Q) output Low. The data on the (D) input is loaded into the flip-flop when CLR is Low on the Low-to-High clock transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
CLR	D	C	Q
1	X	X	0
0	D	↑	D

Usage

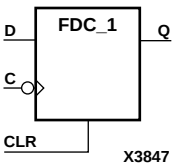
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes.

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0	0 or 1	Sets the initial value of Q output after configuration

FDC_1

Primitive: D Flip-Flop with Negative-Edge Clock and Asynchronous Clear



FDC_1 is a single D-type flip-flop with data input (D), asynchronous clear input (CLR), and data output (Q). The asynchronous CLR, when active, overrides all other inputs and sets the (Q) output Low. The data on the (D) input is loaded into the flip-flop during the High-to-Low clock (C) transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
CLR	D	C	Q
1	X	X	0
0	D	↓	D

Usage

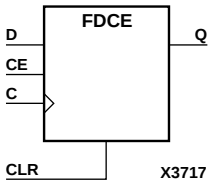
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes.

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0	0 or 1	Sets the initial value of Q output after configuration

FDCE

Primitive: D Flip-Flop with Clock Enable and Asynchronous Clear



FDCE is a single D-type flip-flop with clock enable and asynchronous clear. When clock enable (CE) is High and asynchronous clear (CLR) is Low, the data on the data input (D) of FDCE is transferred to the corresponding data output (Q) during the Low-to-High clock (C) transition. When CLR is High, it overrides all other inputs and resets the data output (Q) Low. When CE is Low, clock transitions are ignored.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
CLR	CE	D	C	Q
1	X	X	X	0
0	0	X	X	No Change
0	1	D	↑	D

Usage

This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes.

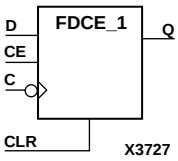
Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

For More Information

Consult the Spartan-3E Data Sheets.

FDCE_1

Primitive: D Flip-Flop with Negative-Edge Clock, Clock Enable, and Asynchronous Clear



FDCE_1 is a single D-type flip-flop with data (D), clock enable (CE), asynchronous clear (CLR) inputs, and data output (Q). The asynchronous CLR input, when High, overrides all other inputs and sets the Q output Low. The data on the (D) input is loaded into the flip-flop when CLR is Low and CE is High on the High-to-Low clock (C) transition. When CE is Low, the clock transitions are ignored.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
CLR	CE	D	C	Q
1	X	X	X	0
0	0	X	↓	No Change
0	1	1	↓	1
0	1	0	↓	0

Usage

This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes.

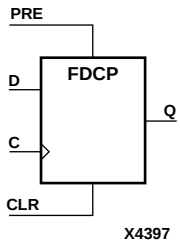
Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

For More Information

Consult the Spartan-3E Data Sheets.

FDCP

Primitive: D Flip-Flop Asynchronous Preset and Clear



FDCP is a single D-type flip-flop with data (D), asynchronous preset (PRE) and clear (CLR) inputs, and data output (Q). The asynchronous PRE, when High, sets the (Q) output High; CLR, when High, resets the output Low. Data on the (D) input is loaded into the flip-flop when PRE and CLR are Low on the Low-to-High clock (C) transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
CLR	PRE	D	C	Q
1	X	X	X	0
0	1	X	X	1
0	0	D	↑	D

Usage

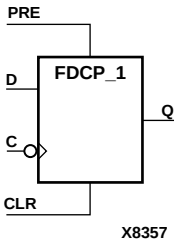
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes.

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

FDCP_1

Primitive: D Flip-Flop with Negative-Edge Clock and Asynchronous Preset and Clear



FDCP_1 is a single D-type flip-flop with data (D), asynchronous preset (PRE) and clear (CLR) inputs, and data output (Q). The asynchronous PRE, when High, sets the (Q) output High; CLR, when High, resets the output Low. Data on the (D) input is loaded into the flip-flop when PRE and CLR are Low on the High-to-Low clock (C) transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the Spartan-3E symbol.

Inputs				Outputs
CLR	PRE	D	C	Q
1	X	X	X	0
0	1	X	X	1
0	0	0	↓	0
0	0	1	↓	1

Usage

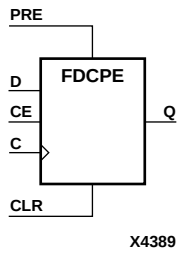
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes.

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

FDCPE

Primitive: D Flip-Flop with Clock Enable and Asynchronous Preset and Clear



FDCPE is a single D-type flip-flop with data (D), clock enable (CE), asynchronous preset (PRE), and asynchronous clear (CLR) inputs and data output (Q). The asynchronous PRE, when High, sets the (Q) output High; CLR, when High, resets the output Low. Data on the (D) input is loaded into the flip-flop when PRE and CLR are Low and CE is High on the Low-to-High clock (C) transition. When CE is Low, the clock transitions are ignored.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, the power on condition can be simulated when global set / reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

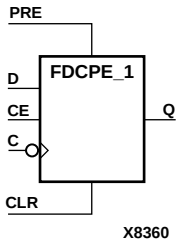
Inputs					Outputs
CLR	PRE	CE	D	C	Q
1	X	X	X	X	0
0	1	X	X	X	1
0	0	0	X	X	No Change
0	0	1	D	↑	D

Usage

This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

FDCPE_1

Primitive: D Flip-Flop with Negative-Edge Clock, Clock Enable, and Asynchronous Preset and Clear



FDCPE_1 is a single D-type flip-flop with data (D), clock enable (CE), asynchronous preset (PRE), and asynchronous clear (CLR) inputs and data output (Q). The asynchronous PRE, when High, sets the (Q) output High; CLR, when High, resets the output Low. Data on the (D) input is loaded into the flip-flop when PRE and CLR are Low and CE is High on the High-to-Low clock (C) transition. When CE is Low, the clock transitions are ignored.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E power on conditions are simulated when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E, symbol.

Inputs					Outputs
CLR	PRE	CE	D	C	Q
1	X	X	X	X	0
0	1	X	X	X	1
0	0	0	X	X	No Change
0	0	1	D	↓	D

Usage

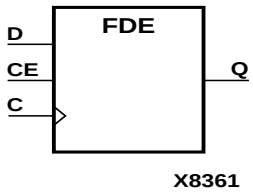
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes.

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

FDE

Primitive: D Flip-Flop with Clock Enable



FDE is a single D-type flip-flop with data input (D), clock enable (CE), and data output (Q). When clock enable is High, the data on the (D) input is loaded into the flip-flop during the Low-to-High clock (C) transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
CE	D	C	Q
0	X	X	No Change
1	0	↑	0
1	1	↑	1

Usage

This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

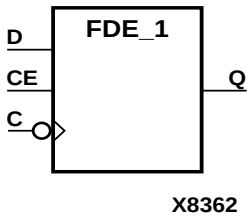
Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

For More Information

Consult the Spartan-3E Data Sheets.

FDE_1

Primitive: D Flip-Flop with Negative-Edge Clock and Clock Enable



FDE_1 is a single D-type flip-flop with data input (D), clock enable (CE), and data output (Q). When clock enable is High, the data on the (D) input is loaded into the flip-flop during the High-to-Low clock (C) transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

The STARTUP_SPARTAN3E component must be instantiated in order to be incorporated into the design. Do not connect any input not needed for the design.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
CE	D	C	Q
0	X	X	No Change
1	0	↓	0
1	1	↓	1

Usage

This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

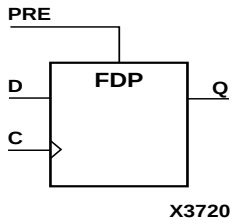
Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

For More Information

Consult the Spartan-3E Data Sheets.

FDP

Primitive: D Flip-Flop with Asynchronous Preset



FDP is a single D-type flip-flop with data (D) and asynchronous preset (PRE) inputs and data output (Q). The asynchronous PRE, when High, overrides all other inputs and presets the (Q) output High. The data on the (D) input is loaded into the flip-flop when PRE is Low on the Low-to-High clock (C) transition.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
PRE	C	D	Q
1	X	X	1
0	↑	D	D
0	↑	0	0

Usage

This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

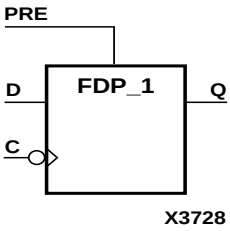
Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

For More Information

Consult the Spartan-3E Data Sheets.

FDP_1

Primitive: D Flip-Flop with Negative-Edge Clock and Asynchronous Preset



FDP_1 is a single D-type flip-flop with data (D) and asynchronous preset (PRE) inputs and data output (Q). The asynchronous PRE, when High, overrides all other inputs and presets the Q output High. The data on the D input is loaded into the flip-flop when PRE is Low on the High-to-Low clock (C) transition.

The flip-flop is asynchronously preset, output High, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
PRE	C	D	Q
1	X	X	1
0	↓	D	D

Usage

This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

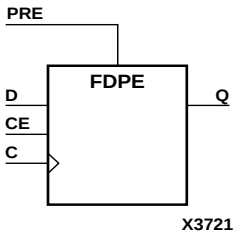
Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

For More Information

Consult the Spartan-3E Data Sheets.

FDPE

Primitive: D Flip-Flop with Clock Enable and Asynchronous Preset



FDPE is a single D-type flip-flop with data (D), clock enable (CE), and asynchronous preset (PRE) inputs and data output (Q). The asynchronous PRE, when High, overrides all other inputs and sets the (Q) output High. Data on the (D) input is loaded into the flip-flop when PRE is Low and CE is High on the Low-to-High clock (C) transition. When CE is Low, the clock transitions are ignored.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol symbol.

Inputs				Outputs
PRE	CE	D	C	Q
1	X	X	X	1
0	0	X	X	No Change
0	1	D	↑	D

Usage

This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

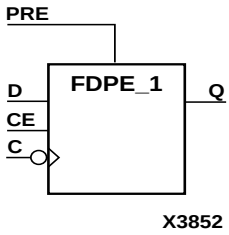
Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

For More Information

Consult the Spartan-3E Data Sheets.

FDPE_1

Primitive: D Flip-Flop with Negative-Edge Clock, Clock Enable, and Asynchronous Preset



FDPE_1 is a single D-type flip-flop with data (D), clock enable (CE), and asynchronous preset (PRE) inputs and data output (Q). The asynchronous PRE, when High, overrides all other inputs and sets the (Q) output High. Data on the (D) input is loaded into the flip-flop when PRE is Low and CE is High on the High-to-Low clock (C) transition. When CE is Low, the clock transitions are ignored.

The flip-flop is asynchronously preset, output High, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
PRE	CE	D	C	Q
1	X	X	X	1
0	0	X	X	No Change
0	1	D	↓	D

Usage

This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

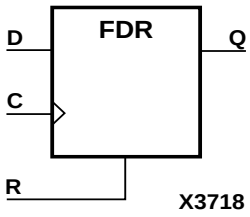
Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

For More Information

Consult the Spartan-3E Data Sheets.

FDR

Primitive: D Flip-Flop with Synchronous Reset



FDR is a single D-type flip-flop with data (D) and synchronous reset (R) inputs and data output (Q). The synchronous reset (R) input, when High, overrides all other inputs and resets the (Q) output Low on the Low-to-High clock (C) transition. The data on the (D) input is loaded into the flip-flop when R is Low during the Low-to-High clock transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
R	D	C	Q
1	X	↑	0
0	D	↑	D

Usage

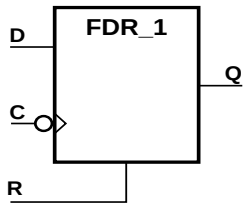
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

FDR_1

Primitive: D Flip-Flop with Negative-Edge Clock and Synchronous Reset



FDR_1 is a single D-type flip-flop with data (D) and synchronous reset (R) inputs and data output (Q). The synchronous reset (R) input, when High, overrides all other inputs and resets the (Q) output Low on the High-to-Low clock (C) transition. The data on the (D) input is loaded into the flip-flop when R is Low during the High-to-Low clock transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol

Inputs			Outputs
R	D	C	Q
1	X	↓	0
0	D	↓	D

Usage

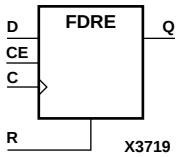
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

FDRE

Primitive: D Flip-Flop with Clock Enable and Synchronous Reset



FDRE is a single D-type flip-flop with data (D), clock enable (CE), and synchronous reset (R) inputs and data output (Q). The synchronous reset (R) input, when High, overrides all other inputs and resets the (Q) output Low on the Low-to-High clock (C) transition. The data on the (D) input is loaded into the flip-flop when R is Low and CE is High during the Low-to-High clock transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
R	CE	D	C	Q
1	X	X	↑	0
0	0	X	X	No Change
0	1	D	↑	D

Usage

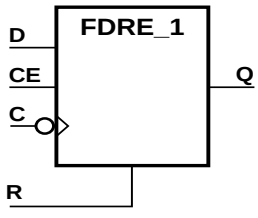
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

FDRE_1

Primitive: D Flip-Flop with Negative-Clock Edge, Clock Enable, and Synchronous Reset



FDRE_1 is a single D-type flip-flop with data (D), clock enable (CE), and synchronous reset (R) inputs and data output (Q). The synchronous reset (R) input, when High, overrides all other inputs and resets the (Q) output Low on the High-to-Low clock (C) transition. The data on the (D) input is loaded into the flip-flop when R is Low and CE is High during the High-to-Low clock transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
R	CE	D	C	Q
1	X	X	↓	0
0	0	X	X	No Change
0	1	D	↓	D

Usage

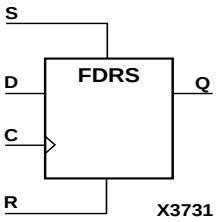
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

FDRS

Primitive: D Flip-Flop with Synchronous Reset and Set



FDRS is a single D-type flip-flop with data (D), synchronous set (S), and synchronous reset (R) inputs and data output (Q). The synchronous reset (R) input, when High, overrides all other inputs and resets the (Q) output Low during the Low-to-High clock (C) transition. (Reset has precedence over Set.) When S is High and R is Low, the flip-flop is set, output High, during the Low-to-High clock transition. When R and S are Low, data on the (D) input is loaded into the flip-flop during the Low-to-High clock transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
R	S	D	C	Q
1	X	X	↑	0
0	1	X	↑	1
0	0	D	↑	D

Usage

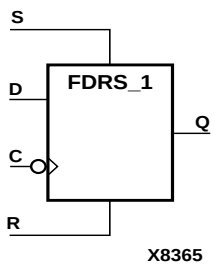
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

FDRS_1

Primitive: D Flip-Flop with Negative-Clock Edge and Synchronous Reset and Set



FDRS_1 is a single D-type flip-flop with data (D), synchronous set (S), and synchronous reset (R) inputs and data output (Q). The synchronous reset (R) input, when High, overrides all other inputs and resets the (Q) output Low during the High-to-Low clock (C) transition. (Reset has precedence over Set.) When S is High and R is Low, the flip-flop is set, output High, during the High-to-Low clock transition. When R and S are Low, data on the (D) input is loaded into the flip-flop during the High-to-Low clock transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
R	S	D	C	Q
1	X	X	↓	0
0	1	X	↓	1
0	0	D	↓	D

Usage

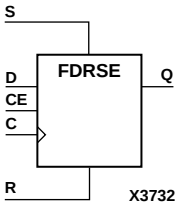
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

FDRSE

Primitive: D Flip-Flop with Synchronous Reset and Set and Clock Enable



FDRSE is a single D-type flip-flop with synchronous reset (R), synchronous set (S), and clock enable (CE) inputs and data output (Q). The reset (R) input, when High, overrides all other inputs and resets the Q output Low during the Low-to-High clock transition. When the set (S) input is High and R is Low, the flip-flop is set, output High, during the Low-to-High clock (C) transition. Data on the D input is loaded into the flip-flop when R and S are Low and CE is High during the Low-to-High clock transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

Spartan-3E simulates power-on when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs					Outputs
R	S	CE	D	C	Q
1	X	X	X	↑	0
0	1	X	X	↑	1
0	0	0	X	X	No Change
0	0	1	1	↑	1
0	0	1	0	↑	0

Usage

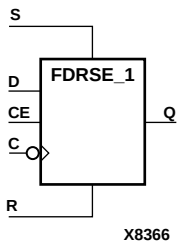
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

FDRSE_1

Primitive: D Flip-Flop with Negative-Clock Edge, Synchronous Reset and Set, and Clock Enable



FDRSE_1 is a single D-type flip-flop with synchronous reset (R), synchronous set (S), and clock enable (CE) inputs and data output (Q). The reset (R) input, when High, overrides all other inputs and resets the (Q) output Low during the High-to-Low clock transition. (Reset has precedence over Set.) When the set (S) input is High and R is Low, the flip-flop is set, output High, during the High-to-Low clock (C) transition. Data on the (D) input is loaded into the flip-flop when (R) and (S) are Low and (CE) is High during the High-to-Low clock transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs					Outputs
R	S	CE	D	C	Q
1	X	X	X	↓	0
0	1	X	X	↓	1
0	0	0	X	X	No Change
0	0	1	D	↓	D

Usage

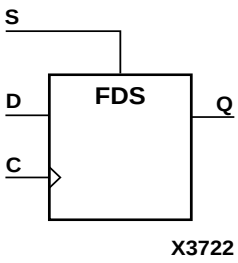
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

FDS

Primitive: D Flip-Flop with Synchronous Set



FDS is a single D-type flip-flop with data (D) and synchronous set (S) inputs and data output (Q). The synchronous set input, when High, sets the Q output High on the Low-to-High clock (C) transition. The data on the D input is loaded into the flip-flop when S is Low during the Low-to-High clock (C) transition.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
S	D	C	Q
1	X	↑	1
0	D	↑	D

Usage

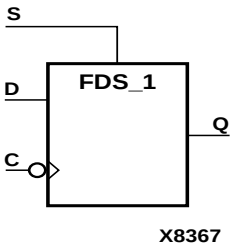
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

FDS_1

Primitive: D Flip-Flop with Negative-Edge Clock and Synchronous Set



FDS_1 is a single D-type flip-flop with data (D) and synchronous set (S) inputs and data output (Q). The synchronous set input, when High, sets the (Q) output High on the High-to-Low clock (C) transition. The data on the (D) input is loaded into the flip-flop when S is Low during the High-to-Low clock (C) transition.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
S	D	C	Q
1	X	↓	1
0	D	↓	D

Usage

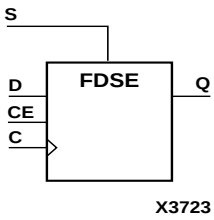
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

FDSE

Primitive: D Flip-Flop with Clock Enable and Synchronous Set



FDSE is a single D-type flip-flop with data (D), clock enable (CE), and synchronous set (S) inputs and data output (Q). The synchronous set (S) input, when High, overrides the clock enable (CE) input and sets the Q output High during the Low-to-High clock (C) transition. The data on the D input is loaded into the flip-flop when S is Low and CE is High during the Low-to-High clock (C) transition.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
S	CE	D	C	Q
1	X	X	↑	1
0	0	X	X	No Change
0	1	D	↑	D

Usage

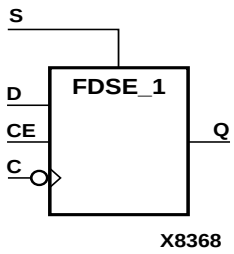
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

FDSE_1

Primitive: D Flip-Flop with Negative-Edge Clock, Clock Enable, and Synchronous Set



FDSE_1 is a single D-type flip-flop with data (D), clock enable (CE), and synchronous set (S) inputs and data output (Q). The synchronous set (S) input, when High, overrides the clock enable (CE) input and sets the Q output High during the High-to-Low clock (C) transition. The data on the D input is loaded into the flip-flop when S is Low and CE is High during the High-to-Low clock (C) transition.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
S	CE	D	C	Q
1	X	X	↓	1
0	0	X	X	No Change
0	1	D	↓	D

Usage

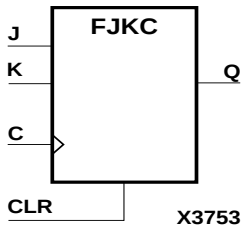
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit Binary	0 or 1	0	Sets the initial value of Q output after configuration

FJKC

Macro: J-K Flip-Flop with Asynchronous Clear



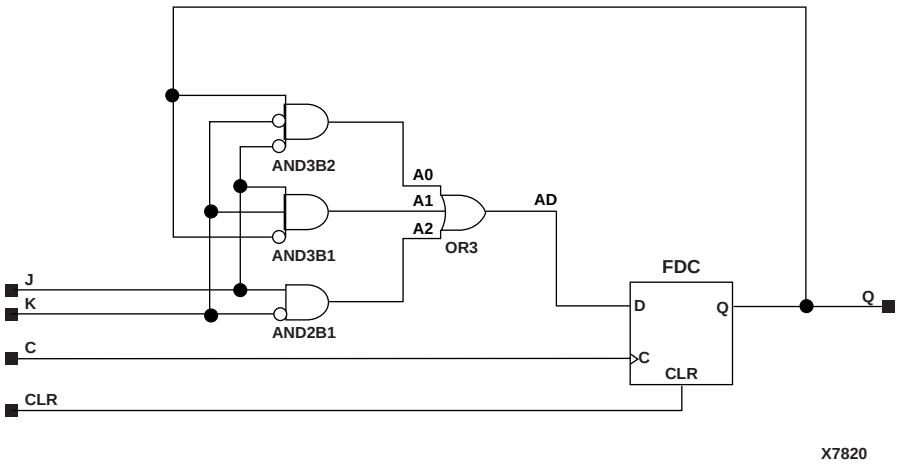
FJKC is a single J-K-type flip-flop with J, K, and asynchronous clear (CLR) inputs and data output (Q). The asynchronous clear (CLR) input, when High, overrides all other inputs and resets the (Q) output Low. When (CLR) is Low, the output responds to the state of the J and K inputs, as shown in the following truth table, during the Low-to-High clock (C) transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
CLR	J	K	C	Q
1	X	X	X	0
0	0	0	↑	No Change
0	0	1	↑	0
0	1	1	↑	1



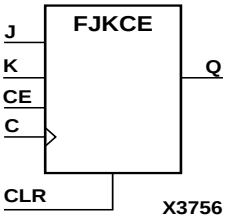
FJKC Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

FJKCE

Macro: J-K Flip-Flop with Clock Enable and Asynchronous Clear



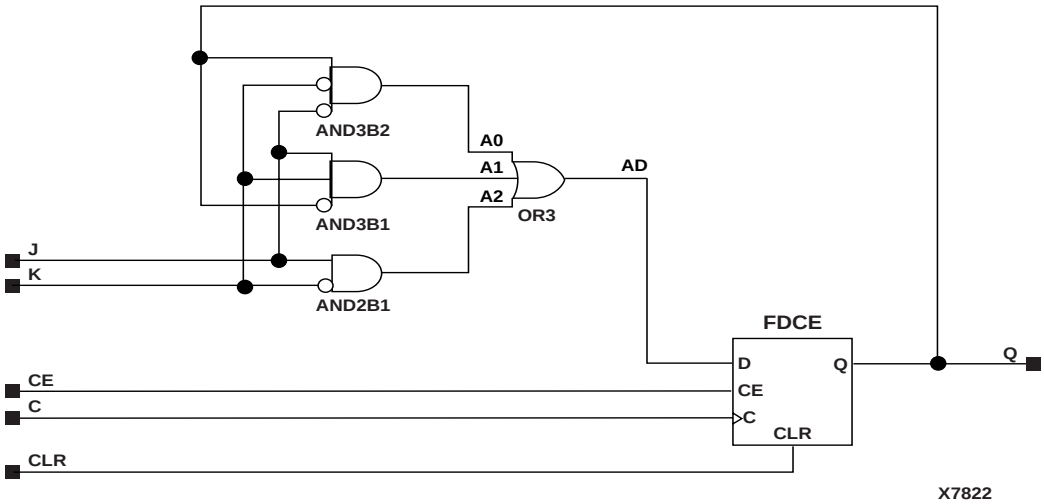
FJKCE is a single J-K-type flip-flop with J, K, clock enable (CE), and asynchronous clear (CLR) inputs and data output (Q). The asynchronous clear (CLR), when High, overrides all other inputs and resets the Q output Low. When CLR is Low and CE is High, Q responds to the state of the J and K inputs, as shown in the following truth table, during the Low-to-High clock transition. When CE is Low, the clock transitions are ignored.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs					Outputs
CLR	CE	J	K	C	Q
1	X	X	X	X	0
0	0	X	X	X	No Change
0	1	0	0	X	No Change
0	1	0	1	↑	0
0	1	1	0	↑	1
0	1	1	1	↑	Toggle



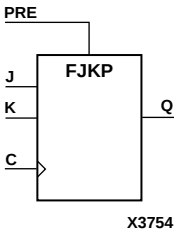
FJKCE Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

FJKP

Macro: J-K Flip-Flop with Asynchronous Preset

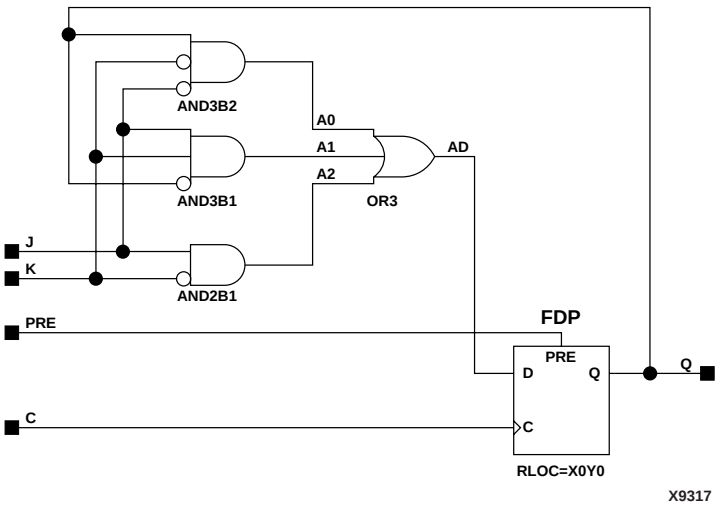


FJKP is a single J-K-type flip-flop with J, K, and asynchronous preset (PRE) inputs and data output (Q). The asynchronous preset (PRE) input, when High, overrides all other inputs and sets the (Q) output High. When (PRE) is Low, the (Q) output responds to the state of the J and K inputs, as shown in the following truth table, during the Low-to-High clock transition.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
PRE	J	K	C	Q
1	X	X	X	1
0	0	0	X	No Change
0	0	1	↑	0
0	1	0	↑	1
0	1	1	↑	Toggle



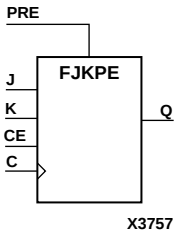
FJKP Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

FJKPE

Macro: J-K Flip-Flop with Clock Enable and Asynchronous Preset

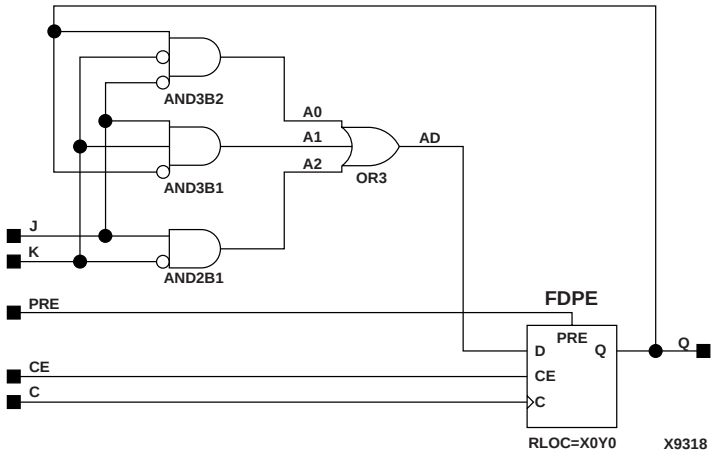


FJKPE is a single J-K-type flip-flop with J, K, clock enable (CE), and asynchronous preset (PRE) inputs and data output (Q). The asynchronous preset (PRE), when High, overrides all other inputs and sets the (Q) output High. When (PRE) is Low and (CE) is High, the (Q) output responds to the state of the J and K inputs, as shown in the truth table, during the Low-to-High clock (C) transition. When (CE) is Low, clock transitions are ignored.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs					Outputs
PRE	CE	J	K	C	Q
1	X	X	X	X	1
0	0	X	X	X	No Change
0	1	0	0	X	No Change
0	1	0	1	↑	0
0	1	1	0	↑	1
0	1	1	1	↑	Toggle



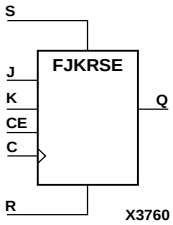
FJKPE Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

FJKRSE

Macro: J-K Flip-Flop with Clock Enable and Synchronous Reset and Set



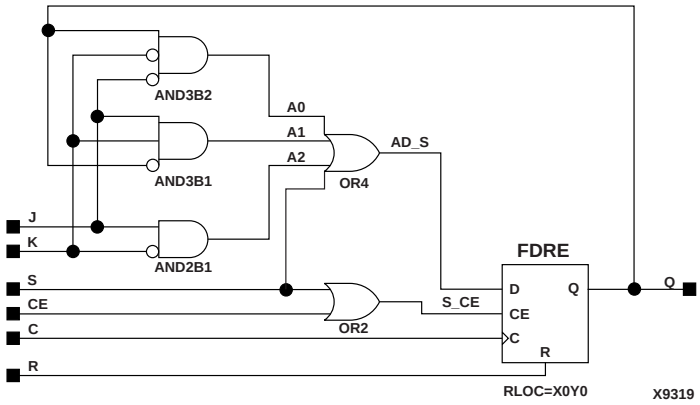
FJKRSE is a single J-K-type flip-flop with J, K, synchronous reset (R), synchronous set (S), and clock enable (CE) inputs and data output (Q). When synchronous reset (R) is High during the Low-to-High clock (C) transition, all other inputs are ignored and output (Q) is reset Low. When synchronous set (S) is High and (R) is Low, output (Q) is set High. When (R) and (S) are Low and (CE) is High, output (Q) responds to the state of the J and K inputs, according to the following truth table, during the Low-to-High clock (C) transition. When (CE) is Low, clock transitions are ignored.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs						Outputs
R	S	CE	J	K	C	Q
1	X	X	X	X	↑	0
0	1	X	X	X	↑	1
0	0	0	X	X	X	No Change
0	0	1	0	0	X	No Change
0	0	1	0	1	↑	0
0	0	1	1	1	↑	Toggle
0	0	1	1	0	↑	1



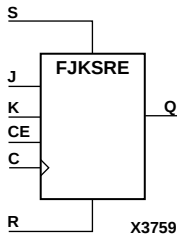
FJKRSE Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

FJKSRE

Macro: J-K Flip-Flop with Clock Enable and Synchronous Set and Reset



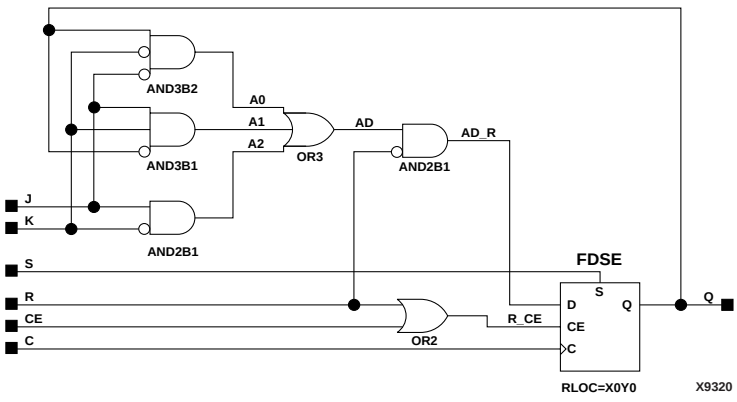
FJKSRE is a single J-K-type flip-flop with J, K, synchronous set (S), synchronous reset (R), and clock enable (CE) inputs and data output (Q). When synchronous set (S) is High during the Low-to-High clock (C) transition, all other inputs are ignored and output (Q) is set High. When synchronous reset (R) is High and (S) is Low, output (Q) is reset Low. When (S) and (R) are Low and (CE) is High, output (Q) responds to the state of the J and K inputs, as shown in the following truth table, during the Low-to-High clock (C) transition. When (CE) is Low, clock transitions are ignored.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs						Outputs
S	R	CE	J	K	C	Q
1	X	X	X	X	↑	1
0	1	X	X	X	↑	0
0	0	0	X	X	X	No Change
0	0	1	0	0	X	No Change
0	0	1	0	1	↑	0
0	0	1	1	0	↑	1
0	0	1	1	1	↑	Toggle



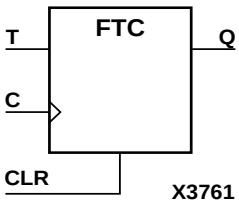
FJKSRE Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

FTC

Macro: Toggle Flip-Flop with Toggle Enable and Asynchronous Clear



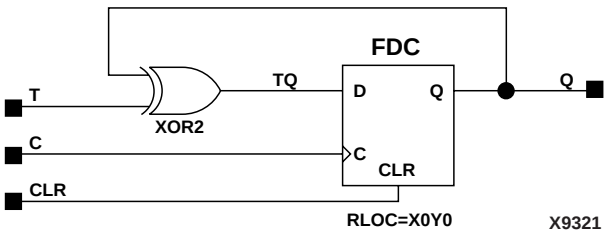
FTC is a synchronous, resettable toggle flip-flop. The asynchronous clear (CLR) input, when High, overrides all other inputs and resets the data output (Q) Low. The (Q) output toggles, or changes state, when the toggle enable (T) input is High and (CLR) is Low during the Low-to-High clock transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
CLR	T	C	Q
1	X	X	0
0	0	X	No Change
0	1	↑	Toggle



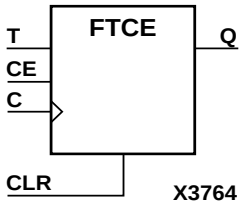
FTC Implementation for Spartan-3E

Usage

This design element can be instantiated or inferred.

FTCE

Macro: Toggle Flip-Flop with Toggle and Clock Enable and Asynchronous Clear



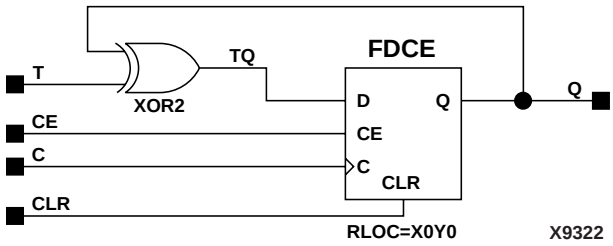
FTCE is a toggle flip-flop with toggle and clock enable and asynchronous clear. When the asynchronous clear (CLR) input is High, all other inputs are ignored and the data output (Q) is reset Low. When (CLR) is Low and toggle enable (T) and clock enable (CE) are High, (Q) output toggles, or changes state, during the Low-to-High clock (C) transition. When (CE) is Low, clock transitions are ignored.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
CLR	CE	T	C	Q
1	X	X	X	0
0	0	X	X	No Change
0	1	0	X	No Change
0	1	1	↑	Toggle



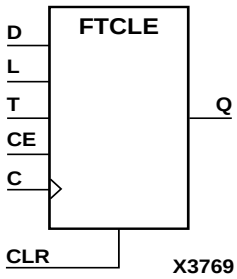
FTCE Implementation for Spartan-3E

Usage

This design element can be inferred or instantiated.

FTCLE

Macro: Toggle/Loadable Flip-Flop with Toggle and Clock Enable and Asynchronous Clear



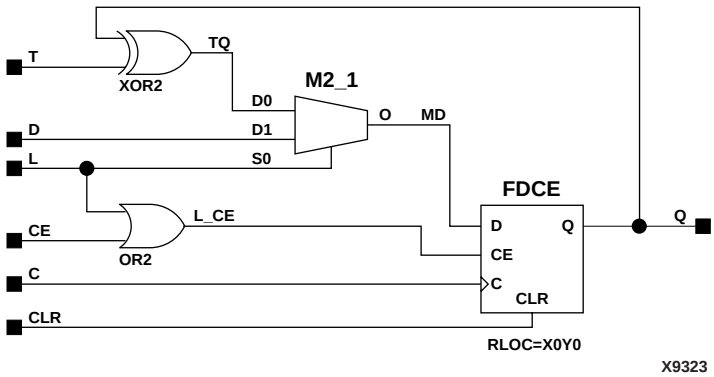
FTCLE is a toggle/loadable flip-flop with toggle and clock enable and asynchronous clear. When the asynchronous clear input (CLR) is High, all other inputs are ignored and output (Q) is reset Low. When load enable input (L) is High and (CLR) is Low, clock enable (CE) is overridden and the data on data input (D) is loaded into the flip-flop during the Low-to-High clock (C) transition. When toggle enable (T) and CE are High and L and (CLR) are Low, output (Q) toggles, or changes state, during the Low-to-High clock transition. When (CE) is Low, clock transitions are ignored.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs						Outputs
CLR	L	CE	T	D	C	Q
1	X	X	X	X	X	0
0	1	X	X	D	↑	D
0	0	0	X	X	X	No Change
0	0	1	0	X	X	No Change
0	0	1	1	X	↑	Toggle



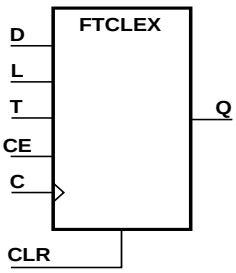
FTCLE Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

FTCLEX

Macro: Toggle/Loadable Flip-Flop with Toggle and Clock Enable and Asynchronous Clear



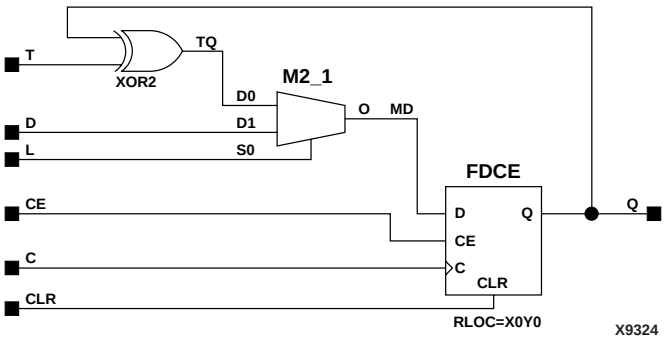
FTCLEX is a toggle/loadable flip-flop with toggle enable and clock enable and asynchronous clear. When the asynchronous clear input (CLR) is High, all other inputs are ignored and output (Q) is reset Low. When load enable input (L) is High, (CLR) is Low, and (CE) is High, the data on data input (D) is loaded into the flip-flop during the Low-to-High clock (C) transition. When toggle enable (T) and (CE) are High and L and (CLR) are Low, output (Q) toggles, or changes state, during the Low-to-High clock transition. When (CE) is Low, clock transitions are ignored.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs						Outputs
CLR	L	CE	T	D	C	Q
1	X	X	X	X	X	0
0	1	1	X	1	↑	1
0	1	1	X	0	↑	0
0	0	0	X	X	X	No Change
0	0	1	0	X	X	No Change
0	0	1	1	X	↑	Toggle



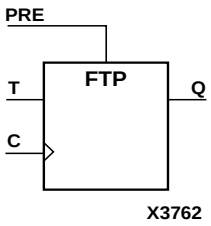
FTCLEX Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

FTP

Macro: Toggle Flip-Flop with Toggle Enable and Asynchronous Preset

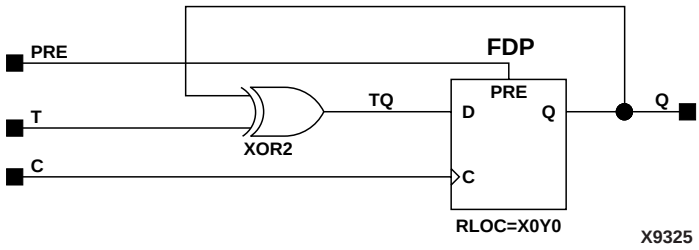


FTP is a toggle flip-flop with toggle enable and asynchronous preset. When the asynchronous preset (PRE) input is High, all other inputs are ignored and output (Q) is set High. When toggle-enable input (T) is High and (PRE) is Low, output (Q) toggles, or changes state, during the Low-to-High clock (C) transition.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
PRE	T	C	Q
1	X	X	1
0	0	X	No Change
0	1	↑	Toggle



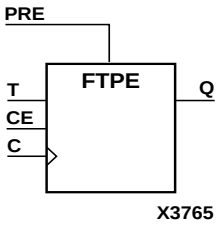
FTP Implementation for Spartan-3E

Usage

This design element can be inferred or instantiated.

FTPE

Macro: Toggle Flip-Flop with Toggle and Clock Enable and Asynchronous Preset

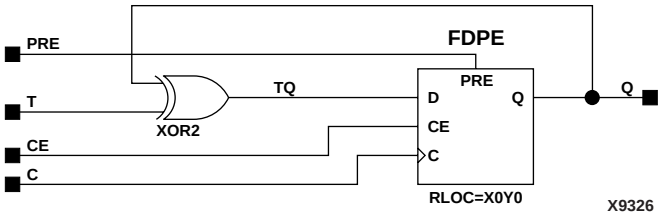


FTPE is a toggle flip-flop with toggle and clock enable and asynchronous preset. When the asynchronous preset (PRE) input is High, all other inputs are ignored and output (Q) is set High. When the toggle enable input (T) is High, clock enable (CE) is High, and (PRE) is Low, output (Q) toggles, or changes state, during the Low-to-High clock transition. When (CE) is Low, clock transitions are ignored.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
PRE	CE	T	C	Q
1	X	X	X	1
0	0	X	X	No Change
0	1	0	X	No Change
0	1	1	↑	Toggle



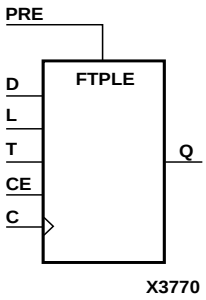
FTPE Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

FTPLE

Macro: Toggle/Loadable Flip-Flop with Toggle and Clock Enable and Asynchronous Preset

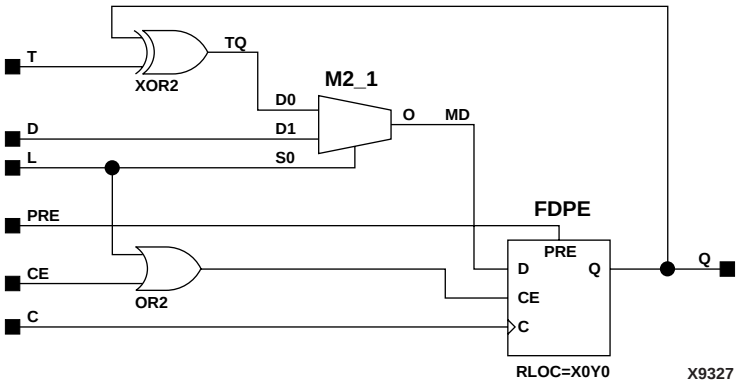


FTPLE is a toggle/loadable flip-flop with toggle and clock enable and asynchronous preset. When the asynchronous preset input (PRE) is High, all other inputs are ignored and output (Q) is set High. When the load enable input (L) is High and (PRE) is Low, the clock enable (CE) is overridden and the data (D) is loaded into the flip-flop during the Low-to-High clock transition. When L and PRE are Low and toggle-enable input (T) and (CE) are High, output (Q) toggles, or changes state, during the Low-to-High clock transition. When (CE) is Low, clock transitions are ignored.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs						Outputs
PRE	L	CE	T	D	C	Q
1	X	X	X	X	X	1
0	1	X	X	D	↑	D
0	0	0	X	X	X	No Change
0	0	1	0	X	X	No Change
0	0	1	1	X	↑	Toggle



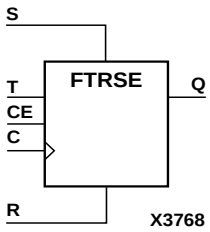
FTPLE Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

FTRSE

Macro: Toggle Flip-Flop with Toggle and Clock Enable and Synchronous Reset and Set



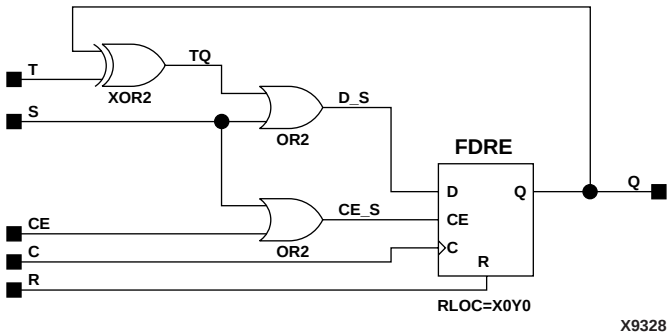
FTRSE is a toggle flip-flop with toggle and clock enable and synchronous reset and set. When the synchronous reset input (R) is High, it overrides all other inputs and the data output (Q) is reset Low. When the synchronous set input (S) is High and (R) is Low, clock enable input (CE) is overridden and output (Q) is set High. (Reset has precedence over Set.) When toggle enable input (T) and (CE) are High and (R) and (S) are Low, output (Q) toggles, or changes state, during the Low-to-High clock transition.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs					Outputs
R	S	CE	T	C	Q
1	X	X	X	↑	0
0	1	X	X	↑	1
0	0	0	X	X	No Change
0	0	1	0	X	No Change
0	0	1	1	↑	Toggle



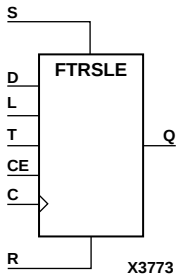
FTRSE Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

FTRSLE

Macro: Toggle/Loadable Flip-Flop with Toggle and Clock Enable and Synchronous Reset and Set



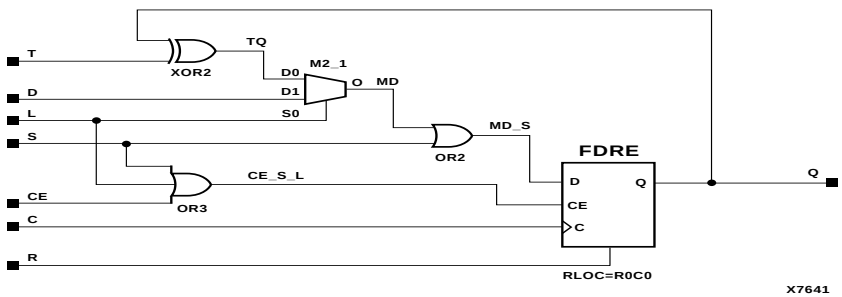
FTRSLE is a toggle/loadable flip-flop with toggle and clock enable and synchronous reset and set. The synchronous reset input (R), when High, overrides all other inputs and resets the data output (Q) Low. (Reset has precedence over Set.) When R is Low and synchronous set input (S) is High, the clock enable input (CE) is overridden and output Q is set High. When R and S are Low and load enable input (L) is High, CE is overridden and data on data input (D) is loaded into the flip-flop during the Low-to-High clock transition. When R, S, and L are Low, CE is High and T is High, output Q toggles, or changes state, during the Low-to-High clock transition. When CE is Low, clock transitions are ignored.

The flip-flop is asynchronously cleared, output Low, when power is applied.

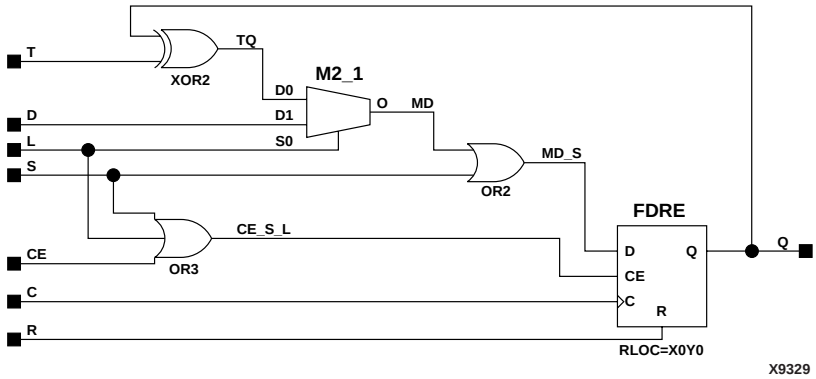
For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

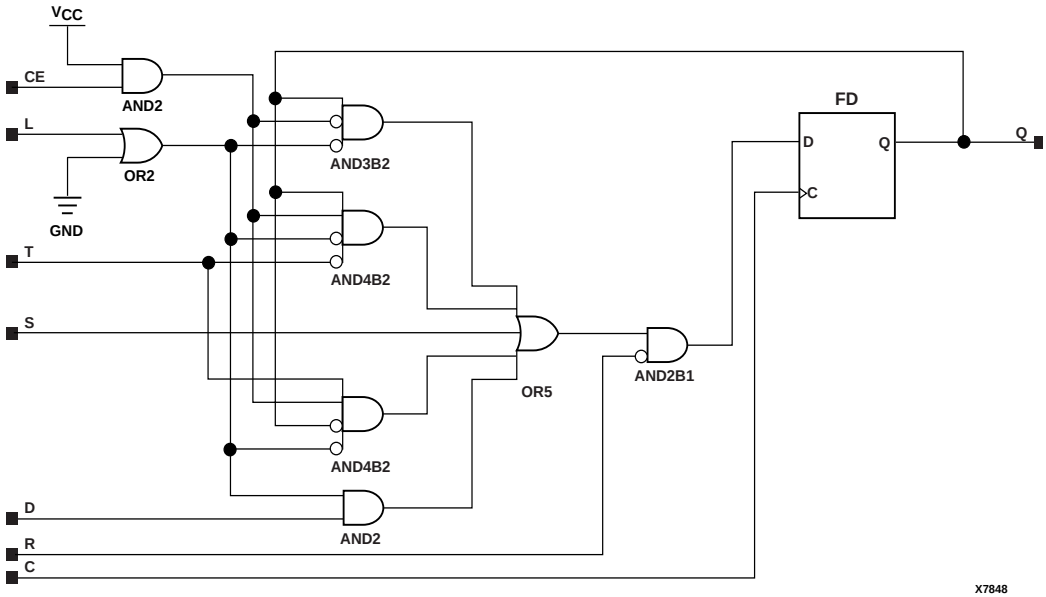
Inputs							Outputs
R	S	L	CE	T	D	C	Q
1	0	X	X	X	X	↑	0
0	1	X	X	X	X	↑	1
0	0	1	X	X	1	↑	1
0	0	1	X	X	0	↑	0
0	0	0	0	X	X	X	No Change
0	0	0	1	0	X	X	No Change
0	0	0	1	1	X	↑	Toggle



FTRSLE Implementation for Spartan-3E



FTRSLE Implementation for Spartan-3E



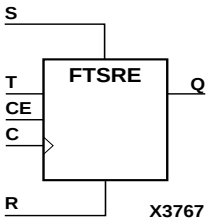
FTRSLE Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

FTSRE

Macro: Toggle Flip-Flop with Toggle and Clock Enable and Synchronous Set and Reset



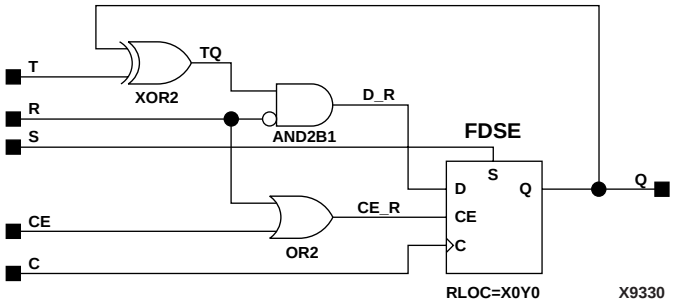
FTSRE is a toggle flip-flop with toggle and clock enable and synchronous set and reset. The synchronous set input, when High, overrides all other inputs and sets data output (Q) High. (Set has precedence over Reset.) When synchronous reset input (R) is High and S is Low, clock enable input (CE) is overridden and output Q is reset Low. When toggle enable input (T) and CE are High and S and R are Low, output Q toggles, or changes state, during the Low-to-High clock transition. When CE is Low, clock transitions are ignored.

The flip-flop is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs					Outputs
S	R	CE	T	C	Q
1	X	X	X	↑	1
0	1	X	X	↑	0
0	0	0	X	X	No Change
0	0	1	0	X	No Change
0	0	1	1	↑	Toggle



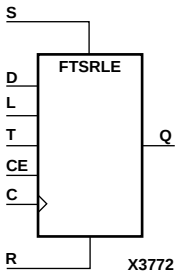
FTSRE Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

FTSRLE

Macro: Toggle/Loadable Flip-Flop with Toggle and Clock Enable and Synchronous Set and Reset

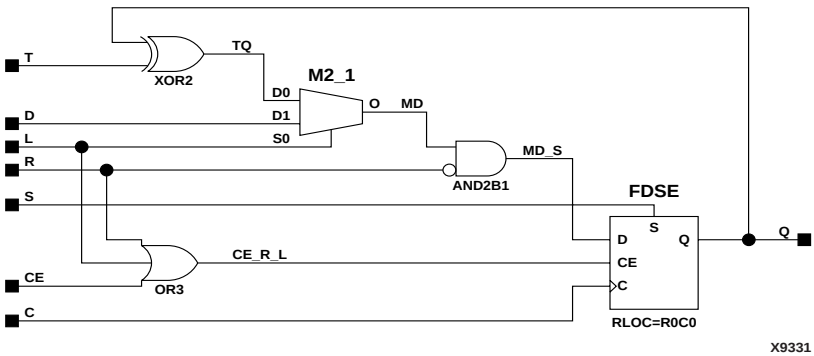


FTSRLE is a toggle/loadable flip-flop with toggle and clock enable and synchronous set and reset. The synchronous set input (S), when High, overrides all other inputs and sets data output (Q) High. (Set has precedence over Reset.) When synchronous reset (R) is High and (S) is Low, clock enable input (CE) is overridden and output (Q) is reset Low. When load enable input (L) is High and S and R are Low, CE is overridden and data on data input (D) is loaded into the flip-flop during the Low-to-High clock transition. When the toggle enable input (T) and (CE) are High and (S), (R), and (L) are Low, output (Q) toggles, or changes state, during the Low-to- High clock transition. When (CE) is Low, clock transitions are ignored.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs							Outputs
S	R	L	CE	T	D	C	Q
1	X	X	X	X	X	↑	1
0	1	X	X	X	X	↑	0
0	0	1	X	X	1	↑	1
0	0	1	X	X	0	↑	0
0	0	0	0	X	X	X	No Change
0	0	0	1	0	X	X	No Change
0	0	0	1	1	X	↑	Toggle



FTSRLE Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

GND

Primitive: Ground-Connection Signal Tag

X3858

The GND signal tag, or parameter, forces a net or input function to a Low logic level. A net tied to GND cannot have any other source.

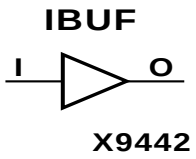
When the logic-trimming software or fitter encounters a net or input function tied to GND, it removes any logic that is disabled by the GND signal. The GND signal is only implemented when the disabled logic cannot be removed.

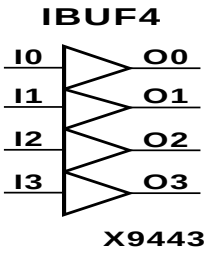
Usage

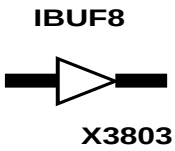
This design element can be instantiated or inferred.

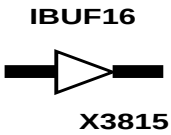
IBUF, 4, 8, 16

Primitive and Macros: Single- and Multiple-Input Buffers

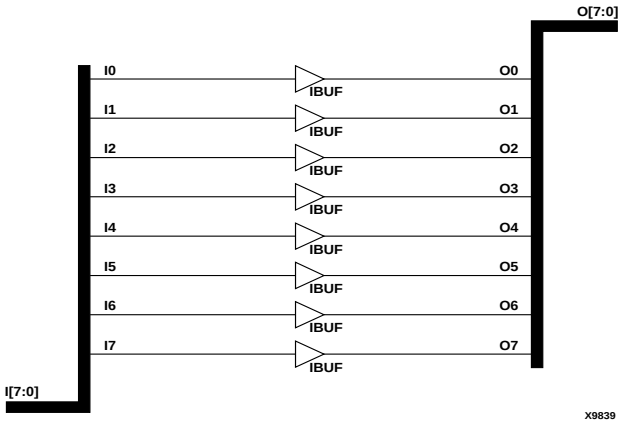








IBUF, IBUF4, IBUF8, and IBUF16 are single- and multiple-input buffers. An IBUF isolates the internal circuit from the signals coming into a chip. IBUFs are contained in input/output blocks (IOBs). IBUF inputs (I) are connected to an IPAD or an IOPAD. IBUF outputs (O) are connected to the internal circuit.



IBUF8 Implementation Spartan-3E

Usage

IBUFs are typically inferred for all top level input ports, but they can also be instantiated if necessary.

Available Attributes

DRIVE

IOSTANDARD

SLEW

IBUF_DELAY_VALUE

IFD_DELAY_VALUE

For More Information

Consult the Spartan-3E Data Sheets for more information about this element.

IBUFDS

Primitive: Differential Signaling Input Buffer with Selectable I/O Interface

X9255

IBUFDS is an input buffer that supports low-voltage, differential signaling. In IBUFDS, a design level interface signal is represented as two distinct ports (I and IB), one deemed the "master" and the other the "slave." The master and the slave are opposite phases of the same logical signal (for example, MYNET and MYNETB).

Inputs		Outputs
I	IB	O
0	0	- *
0	1	0
1	0	1
1	1	- *

* The dash (-) means No Change.

Usage

This design element is supported only for instantiation.

Available Attributes

IOSTANDARD – Specifies the I/O standard to configure this input.

IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the non-registered path out of the IOB. Accepted values are 0 through 16 where 0 (no additional delay added) is the default value. Increasing values increases the magnitude of the delay.

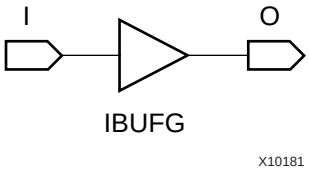
IFD_DELAY_VALUE – Specifies the amount of additional delay to add to the registered path within the IOB. Accepted values are AUTO, 0 through 8 where AUTO (adjust the delay to ensure no positive hold times are present) is the default value. Increasing values increases the magnitude of the delay.

For More Information

Consult the Spartan-3E Data Sheets.

IBUFG

Primitive: Dedicated Input Buffer with Selectable I/O Interface



IBUFG is dedicated to dedicated input buffers for connecting to the clock buffer BUFG or DCM. You can attach an IOSTANDARD attribute to an IBUFG instance.

The Xilinx implementation software converts each BUFG to an appropriate type of global buffer for the target PLD device. The IBUFG input can only be driven by the global clock pins. The IBUFG output can drive CLKIN of a DCM, BUFG, or user logic. IBUFG can be routed to user logic and does not have to be routed to a DCM.

Usage

This design element is supported for schematic and instantiation. Synthesis tools usually infer a BUFGP on any clock net. If there are more clock nets than BUFGPs, the synthesis tool usually instantiates BUFGPs for the clocks that are most utilized. The BUFGP contains both a BUFG and an IBUFG.

Available Attributes

IOSTANDARD – Specifies the I/O standard to configure this input.

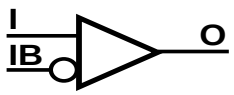
IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the clock path out of the IBUFG. Accepted values are 0 through 16 where 0 (no additional delay added) is the default value. Increasing values increases the magnitude of the delay.

For More Information

Consult the Spartan-3E Data Sheets.

IBUFGDS

Primitive: Dedicated Differential Signaling Input Buffer with Selectable I/O Interface



X9255

IBUFGDS is a dedicated differential signaling input buffer for connection to the clock buffer (BUFG) or DCM. In IBUFGDS, a design level interface signal is represented as two distinct ports (I and IB), one deemed the "master" and the other the "slave." The master and the slave are opposite phases of the same logical signal (for example, MYNET and MYNETB).

Inputs		Outputs
I	IB	O
0	0	- *
0	1	0
1	0	1
1	1	- *

* The dash (-) means No Change.

Usage

This design element is supported only for instantiation.

Available Attributes

IOSTANDARD – Specifies the I/O standard to configure this input.

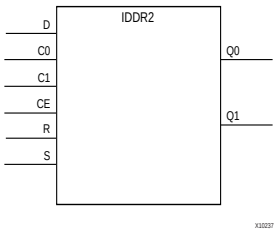
IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the clock path out of the IBUFG. Accepted values are 0 through 16 where 0 (no additional delay added) is the default value. Increasing values increases the magnitude of the delay.

For More Information

Consult the Spartan-3E Data Sheets.

IDDR2

Primitive: Double Data Rate Input D Flip-Flop with Optional Data Alignment, Clock Enable and Programmable Synchronous or Asynchronous Set/Reset



The IDDR2 is an input double data rate (DDR) register useful in capturing double data rate signals entering the FPGA. The IDDR2 requires two clocks to be connected to the component, C0 and C1, so that data is captured at the positive edge of both C0 and C1 clocks. The IDDR2 features an active high clock enable port, CE, which may be used to suspend the operation of the registers, and both set and reset ports that may be configured to be synchronous or asynchronous to the respective clocks. The IDDR2 has an optional alignment feature, which allows both output data ports to the component to be aligned to a single clock.

Usage

The IDDR2 currently must be instantiated in order to be incorporated into the design. In order to change the default behavior of the IDDR2, attributes may be modified via the generic map (VHDL) or named parameter value assignment (Verilog) as a part of the instantiated component. The IDDR2 may be either connected directly to a top-level input port in the design where an appropriate input buffer can be inferred or to an instantiated IBUF, IOBUF, IBUFDS or IOBUFDS. All inputs and outputs of this component should either be connected or properly tied off.

Available Attributes

DDR_ALIGNMENT – Specifies how the data will be presented on the Q0 and Q1 ports. When set to "NONE," the data on the Q0 port will be aligned with the positive edge of the C0 clock and the data on the Q1 port will be aligned with the positive edge of the C1 clock. When set to "C0", the data on both Q0 and Q1 ports are aligned to the positive edge of the C0 clock, and when set to "C1", the data on the Q0 and Q1 ports are aligned to the positive edge of the C1 clock. Data is always captured on the positive edge of both clocks.

INIT_Q0 – Specifies the initial value upon power-up or the assertion of GSR for the Q0 port. This attribute should be set to 0.

INIT_Q1 - Specifies the initial value upon power-up or the assertion of GSR for the Q1 port. This attribute should be set to 0.

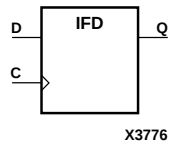
SRTYPE – When set to "SYNC," the reset, R, and set, S, ports are synchronous to the associated clock inputs.

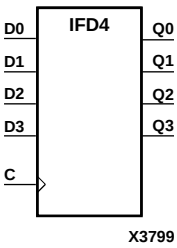
For More Information

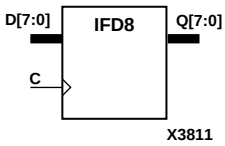
Consult the Spartan-3E Data Sheets.

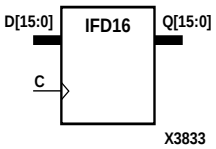
IFD, 4, 8, 16

Macro: Single- and Multiple-Input D Flip-Flops









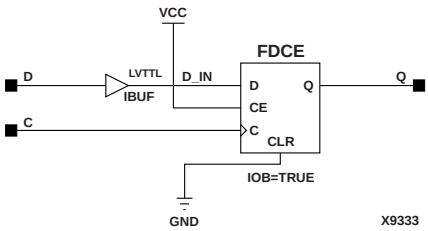
The IFD D-type flip-flop is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD (without using an IBUF). The (D) input provides data input for the flip-flop, which synchronizes data entering the chip. The data on input (D) is loaded into the flip-flop during the Low-to-High clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin.

The flip-flops are asynchronously cleared with Low outputs when power is applied.

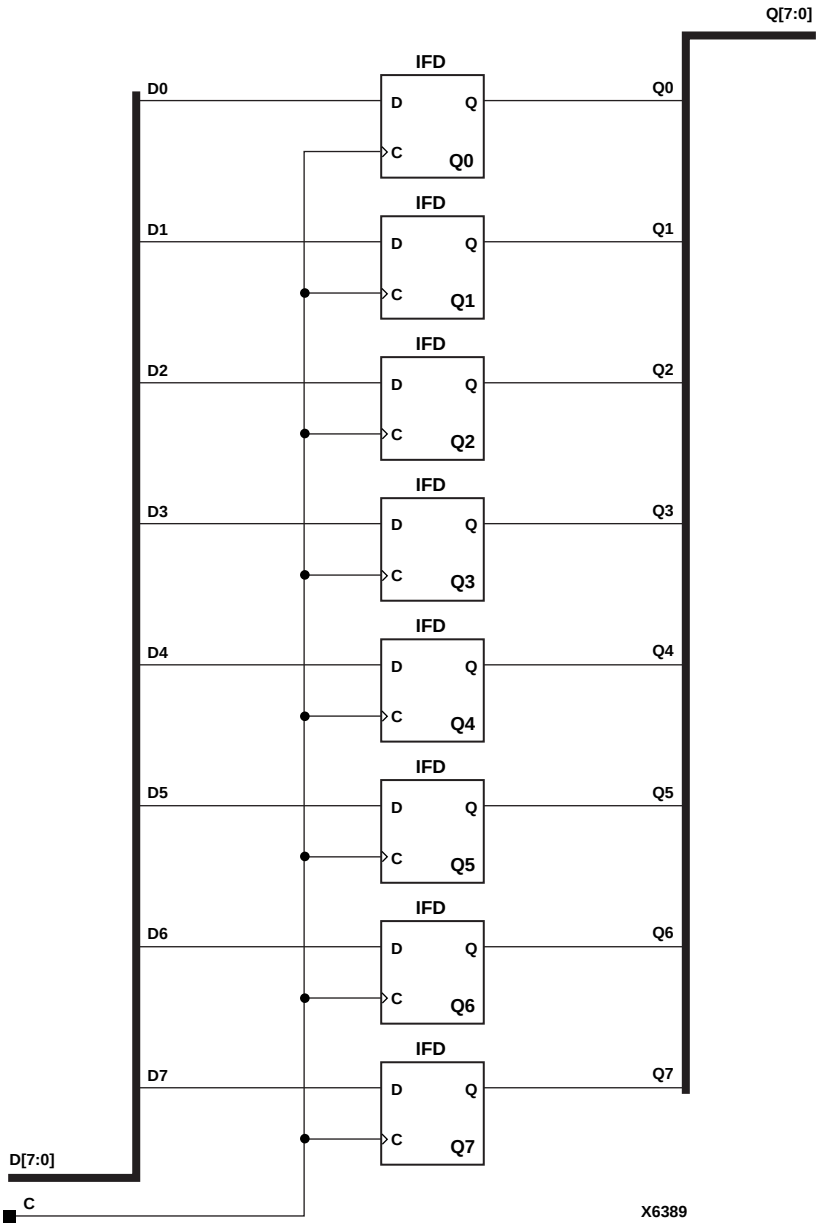
For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs		Outputs
D	C	Q
Dn	↑	Dn



IFD Implementation for Spartan-3E



IFD8 Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an IFD, you would infer an FD and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr i to pack all input registers into the IOBs.

Available Attributes

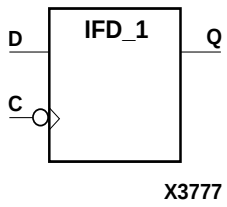
IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the non-registered path out of the IOB. Accepted values are 0 through 16 where 0 (no

additional delay added) is the default value. Increasing values increases the magnitude of the delay.

IFD_DELAY_VALUE – Specifies the amount of additional delay to add to the registered path within the IOB. Accepted values are AUTO, 0 through 8 where AUTO (adjust the delay to ensure no positive hold times are present) is the default value. Increasing values increases the magnitude of the delay.

IFD_1

Macro: Input D Flip-Flop with Inverted Clock



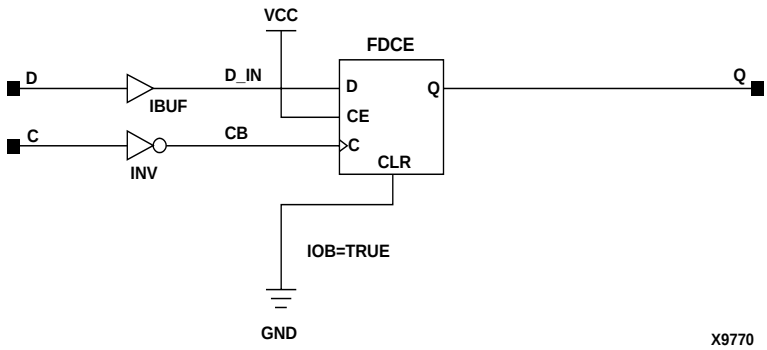
The IFD_1 D-type flip-flop is contained in an input / output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD. The D input also provides data input for the flip-flop, which synchronizes data entering the chip. The D input data is loaded into the flip-flop during the High-to-Low clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin.

The flip-flop is asynchronously cleared with Low output when power is applied.

For Spartan-3E, power on conditions are simulated when global set / reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs		Outputs
D	C	Q
0	↓	0
1	↓	1



IFD_1 Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an IFD_1, you would infer an FD_1 and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr i to pack all input registers into the IOBs.

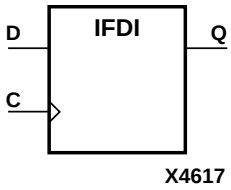
Available Attributes

IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the non-registered path out of the IOB. Accepted values are 0 through 16 where 0 (no additional delay added) is the default value. Increasing values increases the magnitude of the delay.

IFD_DELAY_VALUE – Specifies the amount of additional delay to add to the registered path within the IOB. Accepted values are AUTO, 0 through 8 where AUTO (adjust the delay to ensure no positive hold times are present) is the default value. Increasing values increases the magnitude of the delay.

IFDI

Primitive: Input D Flip-Flop (Asynchronous Preset)



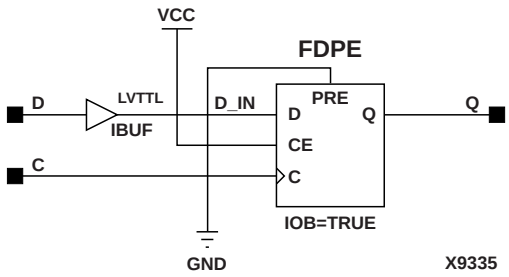
The IFDI D-type flip-flop is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD. The (D) input provides data input for the flip-flop, which synchronizes data entering the chip. The data on input (D) is loaded into the flip-flop during the Low-to-High clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin.

The flip-flop is asynchronously preset, output High, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs		Outputs
D	**C**	**Q**
D	↑	D



IFDI Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an IFDI, you would infer an FDP and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr i to pack all input registers into the IOBs.

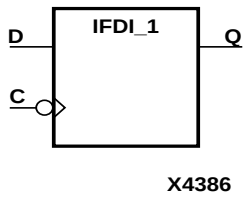
Available Attributes

IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the non-registered path out of the IOB. Accepted values are 0 through 16 where 0 (no additional delay added) is the default value. Increasing values increases the magnitude of the delay.

IFD_DELAY_VALUE – Specifies the amount of additional delay to add to the registered path within the IOB. Accepted values are AUTO, 0 through 8 where AUTO (adjust the delay to ensure no positive hold times are present) is the default value. Increasing values increases the magnitude of the delay.

IFDI_1

Primitive: Input D Flip-Flop with Inverted Clock (Asynchronous Preset)



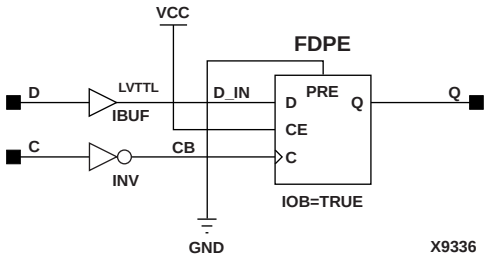
The IFDI_1 D-type flip-flop is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD. The (D) input provides data input for the flip-flop, which synchronizes data entering the chip. The data on input (D) is loaded into the flip-flop during the High-to-Low clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin.

The flip-flop is asynchronously preset, output High, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs		Outputs
D	**C**	**Q**
D	↓	D



IFDI_1 Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an IFDI_1, you would infer an FDP_1 and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr i to pack all input registers into the IOBs.

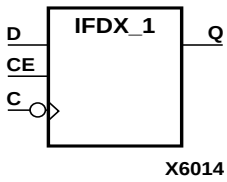
Available Attributes

IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the non-registered path out of the IOB. Accepted values are 0 through 16 where 0 (no additional delay added) is the default value. Increasing values increases the magnitude of the delay.

IFD_DELAY_VALUE – Specifies the amount of additional delay to add to the registered path within the IOB. Accepted values are AUTO, 0 through 8 where AUTO (adjust the delay to ensure no positive hold times are present) is the default value. Increasing values increases the magnitude of the delay.

IFDX_1

Primitive: Input D Flip-Flop with Inverted Clock and Clock Enable



The IFDX_1 D-type flip-flop is contained in an input / output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD. The D input also provides data input for the flip-flop, which synchronizes data entering the chip. When CE is High, the data on input D is loaded into the flip-flop during the High-to-Low clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin. When the CE pin is Low, the output (Q) does not change.

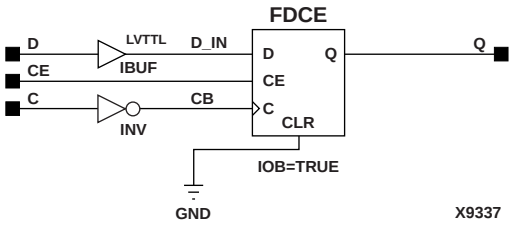
The flip-flop is asynchronously cleared with Low output, when power is applied.

For Spartan-3E, power on conditions are simulated when global set / reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

For more information on IFDX_1, see “ILDX, 4, 8, 16”.

Inputs			Outputs
CE	D	C	Q
1	D	↓	D
0	X	X	No Change



IFDX_1 Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an IFDX_1, you would infer an FDCE_1 and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr i to pack all input registers into the IOBs.

Available Attributes

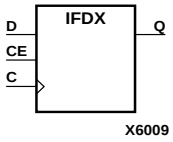
IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the non-registered path out of the IOB. Accepted values are 0 through 16 where 0 (no additional delay added) is the default value. Increasing values increases the magnitude of the delay.

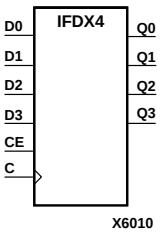
IFD_DELAY_VALUE – Specifies the amount of additional delay to add to the registered path within the IOB. Accepted values are AUTO, 0 through 8 where AUTO

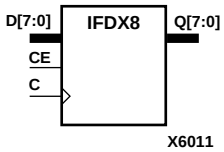
(adjust the delay to ensure no positive hold times are present) is the default value. Increasing values increases the magnitude of the delay.

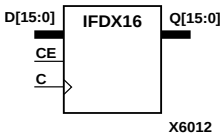
IFDX, 4, 8, 16

Macro: Single- and Multiple-Input D Flip-Flops with Clock Enable









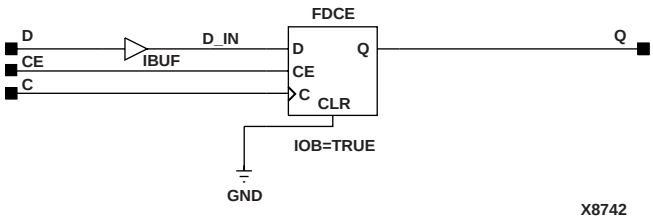
The IFDX D-type flip-flop is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD (without using an IBUF). The D input provides data input for the flip-flop, which synchronizes data entering the chip. When CE is High, the data on input D is loaded into the flip-flop during the Low-to-High clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin. When CE is Low, flip-flop outputs do not change.

The flip-flops are asynchronously cleared with Low outputs when power is applied.

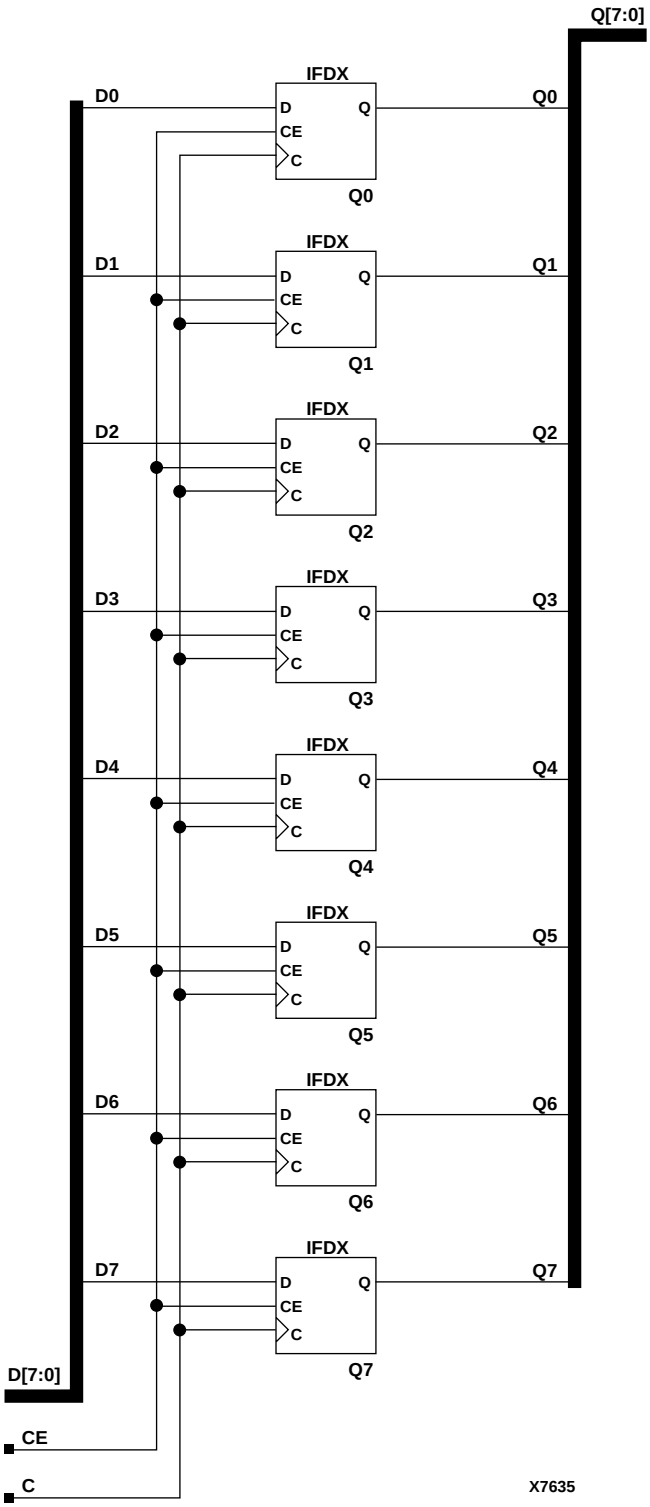
For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
CE	Dn	C	Qn
1	Dn	↑	Dn
0	X	X	No Change



IFDX Implementation for Spartan-3E



IFDX8 Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an IFDX, you would infer an FDCE and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr i to pack all input registers into the IOBs.

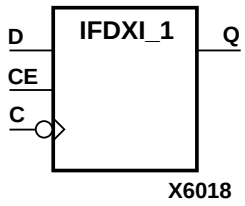
Available Attributes

IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the non-registered path out of the IOB. Accepted values are 0 through 16 where 0 (no additional delay added) is the default value. Increasing values increases the magnitude of the delay.

IFD_DELAY_VALUE – Specifies the amount of additional delay to add to the registered path within the IOB. Accepted values are AUTO, 0 through 8 where AUTO (adjust the delay to ensure no positive hold times are present) is the default value. Increasing values increases the magnitude of the delay.

IFDXI_1

Macro: Input D Flip-Flop with Inverted Clock and Clock Enable (Asynchronous Preset)



The IFDXI_1 D-type flip-flop is contained in an input/output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD. The (D) input provides data input for the flip-flop, which synchronizes data entering the chip. When (CE) is High, the data on input (D) is loaded into the flip-flop during the High-to-Low clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin. When the (CE) pin is Low, the output (Q) does not change.

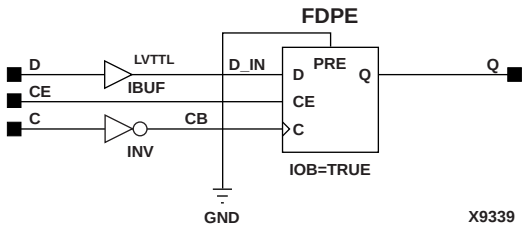
The flip-flop is asynchronously preset with High output when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

For information on legal IFDXI, IFDXI_1, ILDXI, and ILDXI_1 combinations, see "ILDXI".

Inputs			Outputs
CE	D	C	Q
1	D	↓	D
0	X	X	No Change



IFDXI_1 Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an IFDXI_1, you would infer an FDPE_1 and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr i to pack all input registers into the IOBs.

Available Attributes

IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the non-registered path out of the IOB. Accepted values are 0 through 16 where 0 (no

additional delay added) is the default value. Increasing values increases the magnitude of the delay.

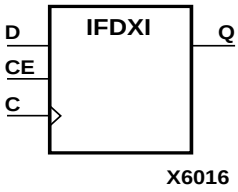
IFD_DELAY_VALUE – Specifies the amount of additional delay to add to the registered path within the IOB. Accepted values are AUTO, 0 through 8 where AUTO (adjust the delay to ensure no positive hold times are present) is the default value. Increasing values increases the magnitude of the delay.

For More Information

Consult the Spartan-3E Data Sheets.

IFDXI

Macro: Input D Flip-Flop with Clock Enable (Asynchronous Preset)



The IFDXI D-type flip-flop is contained in an input / output block (IOB). The input (D) of the flip-flop is connected to an IPAD or an IOPAD. The D input provides data input for the flip-flop, which synchronizes data entering the chip. When CE is High, the data on input D is loaded into the flip-flop during the Low-to-High clock (C) transition and appears at the output (Q). The clock input can be driven by internal logic or through another external pin. When the CE pin is Low, the output (Q) does not change.

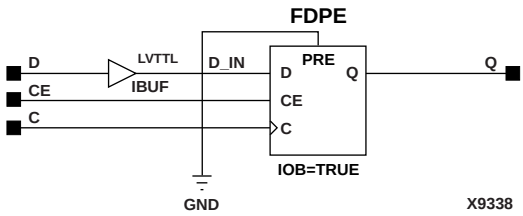
The flip-flop is asynchronously preset with High output, when power is applied.

For Spartan-3E, power on conditions are simulated when global set / reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

For information on legal IFDXI, IFDXI_1, ILDXI, and ILDXI_1 combinations, see "ILDXI".

Inputs			Outputs
CE	D	C	Q
1	D	↑	D
0	X	X	No Change



IFDXI Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an IFDXI, you would infer an FDPE and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr i to pack all input registers into the IOBs.

Available Attributes

IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the non-registered path out of the IOB. Accepted values are 0 through 16 where 0 (no additional delay added) is the default value. Increasing values increases the magnitude of the delay.

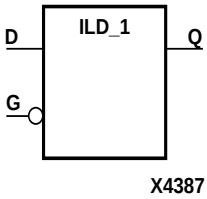
IFD_DELAY_VALUE – Specifies the amount of additional delay to add to the registered path within the IOB. Accepted values are AUTO, 0 through 8 where AUTO (adjust the delay to ensure no positive hold times are present) is the default value. Increasing values increases the magnitude of the delay.

For More Information

Consult the Spartan-3E Data Sheets.

ILD_1

Macro: Transparent Input Data Latch with Inverted Gate



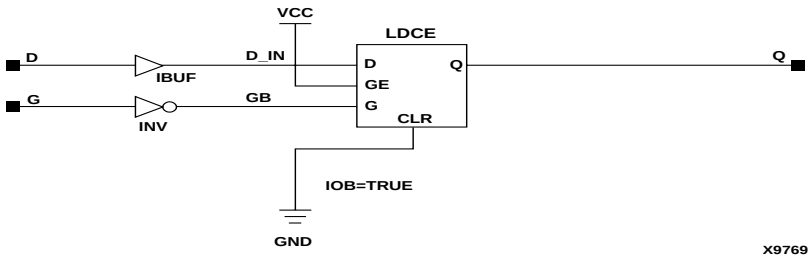
ILD_1 is a transparent data latch, which can be used to hold transient data entering a chip. When the gate input (G) is Low, data on the data input (D) appears on the data output (Q). Data on (D) during the Low-to-High (G) transition is stored in the latch.

The latch is asynchronously cleared with Low output when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs		Outputs
G	D	Q
0	D	D
1	X	D
↑	D	D



ILD_1 Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an ILD_1, you would infer an LD_1 and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr i to pack all input registers into the IOBs.

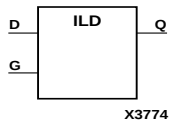
Available Attributes

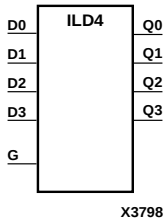
IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the non-registered path out of the IOB. Accepted values are 0 through 16 where 0 (no additional delay added) is the default value. Increasing values increases the magnitude of the delay.

IFD_DELAY_VALUE – Specifies the amount of additional delay to add to the registered path within the IOB. Accepted values are AUTO, 0 through 8 where AUTO (adjust the delay to ensure no positive hold times are present) is the default value. Increasing values increases the magnitude of the delay.

ILD, 4, 8, 16

Macro: Transparent Input Data Latches





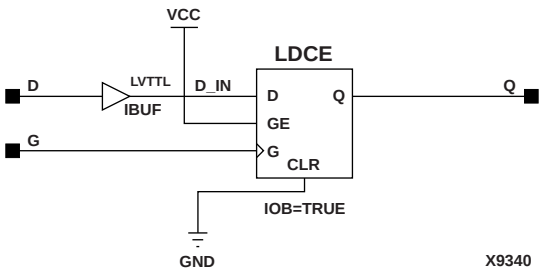
ILD, ILD4, ILD8, and ILD16 are single or multiple transparent data latches, which can be used to hold transient data entering a chip. The ILD latch is contained in an input / output block (IOB). The latch input (D) is connected to an IPAD or an IOPAD (without using an IBUF). When the gate input (G) is High, data on the inputs (D) appears on the outputs (Q). Data on the D inputs during the High-to-Low G transition is stored in the latch.

The latch is asynchronously cleared with Low output when power is applied.

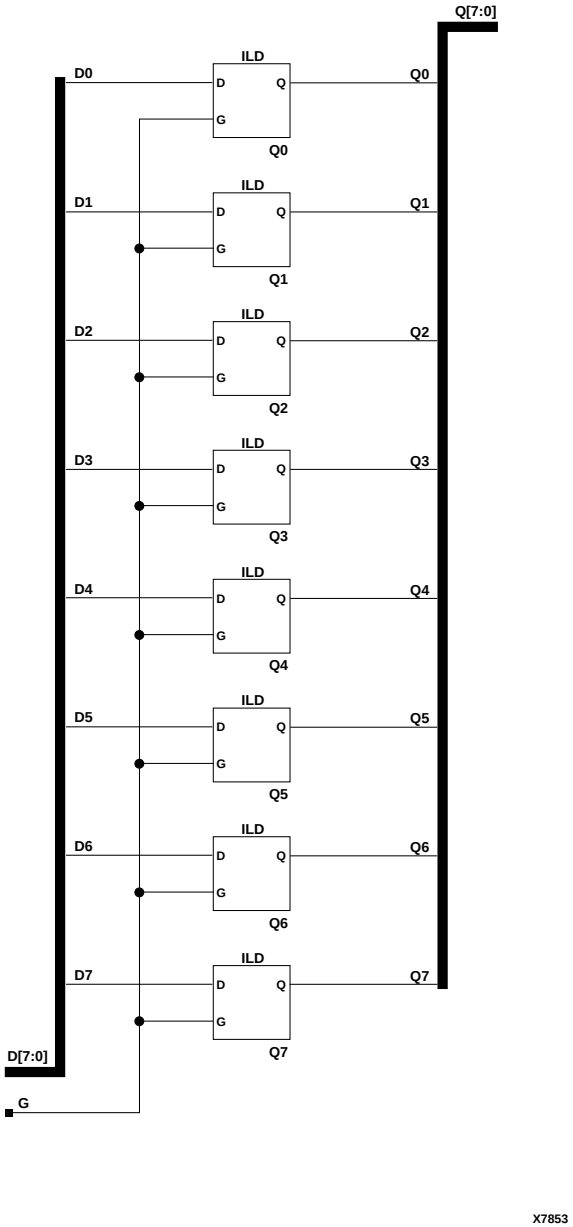
For Spartan-3E, power on conditions are simulated when global set / reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs		Outputs
G	D	Q
1	D	D
0	X	No Change
↓	D	D



ILD Implementation for Spartan-3E



ILD8 Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an ILD, you would infer an LD and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr i to pack all input registers into the IOBs.

Available Attributes

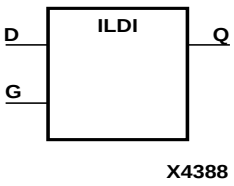
IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the non-registered path out of the IOB. Accepted values are 0 through 16 where 0 (no

additional delay added) is the default value. Increasing values increases the magnitude of the delay.

IFD_DELAY_VALUE – Specifies the amount of additional delay to add to the registered path within the IOB. Accepted values are AUTO, 0 through 8 where AUTO (adjust the delay to ensure no positive hold times are present) is the default value. Increasing values increases the magnitude of the delay.

ILDI

Macro: Transparent Input Data Latch (Asynchronous Preset)



ILDI is a transparent data latch, which can hold transient data entering a chip. When the gate input (G) is High, data on the input (D) appears on the output (Q). Data on the D input during the High-to-Low G transition is stored in the latch.

The latch is asynchronously preset, output High, when power is applied.

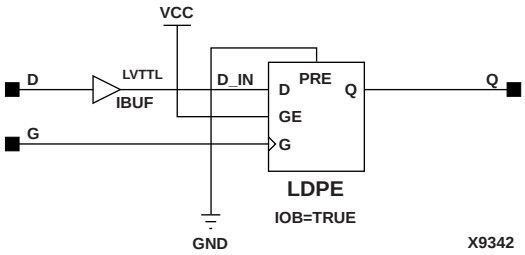
For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

ILDIs and IFDIs

The ILDI is actually the input flip-flop master latch. It is possible to access two different outputs from the input flip-flop: one that responds to the level of the clock signal and another that responds to an edge of the clock signal. When using both outputs from the same input flip-flop, a transparent High latch (ILDI) corresponds to a falling edge-triggered flip-flop (IFDI_1). Similarly, a transparent Low latch (ILDI_1) corresponds to a rising edge-triggered flip-flop (IFDI).

Inputs		Outputs
G	D	Q
1	D	D
0	X	D
↓	D	D



ILDI Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an ILDI, you would infer an LDP and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr i to pack all input registers into the IOBs.

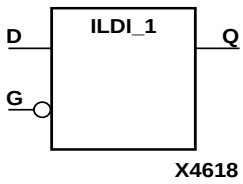
Available Attributes

IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the non-registered path out of the IOB. Accepted values are 0 through 16 where 0 (no additional delay added) is the default value. Increasing values increases the magnitude of the delay.

IFD_DELAY_VALUE – Specifies the amount of additional delay to add to the registered path within the IOB. Accepted values are AUTO, 0 through 8 where AUTO (adjust the delay to ensure no positive hold times are present) is the default value. Increasing values increases the magnitude of the delay.

ILDI_1

Macro: Transparent Input Data Latch with Inverted Gate (Asynchronous Preset)



ILDI_1 is a transparent data latch, which can hold transient data entering a chip. When the gate input (G) is Low, data on the data input (D) appears on the data output (Q). Data on D during the Low-to-High G transition is stored in the latch.

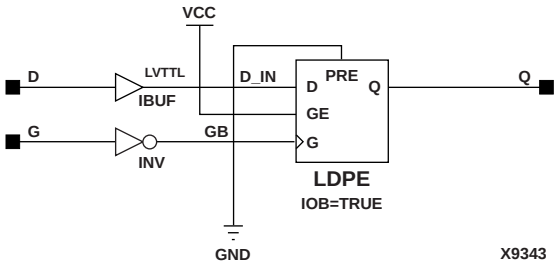
The latch is asynchronously preset, output High, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

For information on ILDI_1, see "ILDI".

Inputs		Outputs
G	D	Q
0	1	1
0	0	0
1	X	D
↑	D	D



ILDI_1 Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an ILDI_1, you would infer an LDP_1 and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr i to pack all input registers into the IOBs.

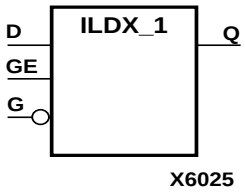
Available Attributes

IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the non-registered path out of the IOB. Accepted values are 0 through 16 where 0 (no additional delay added) is the default value. Increasing values increases the magnitude of the delay.

IFD_DELAY_VALUE – Specifies the amount of additional delay to add to the registered path within the IOB. Accepted values are AUTO, 0 through 8 where AUTO (adjust the delay to ensure no positive hold times are present) is the default value. Increasing values increases the magnitude of the delay.

ILDX_1

Macro: Transparent Input Data Latch with Inverted Gate



ILDX_1 is a transparent data latch, which can be used to hold transient data entering a chip. When the gate input (G) is Low, data on the data input (D) appears on the data output (Q). Data on D during the Low-to-High G transition is stored in the latch.

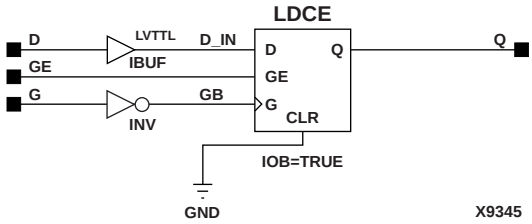
The latch is asynchronously cleared with Low output, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

For more information on ILDX_1, see "ILDX, 4, 8, 16".

Inputs			Outputs
GE	G	D	Q
0	X	X	No Change
1	1	X	No Change
1	0	1	1
1	0	0	0
1	↑	D	D



ILDX_1 Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an ILDX_1, you would infer an LDCE_1 and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr i to pack all input registers into the IOBs.

Available Attributes

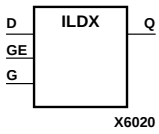
IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the non-registered path out of the IOB. Accepted values are 0 through 16 where 0 (no additional delay added) is the default value. Increasing values increases the magnitude of the delay.

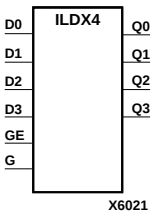
IFD_DELAY_VALUE – Specifies the amount of additional delay to add to the registered path within the IOB. Accepted values are AUTO, 0 through 8 where AUTO

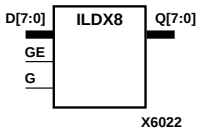
(adjust the delay to ensure no positive hold times are present) is the default value. Increasing values increases the magnitude of the delay.

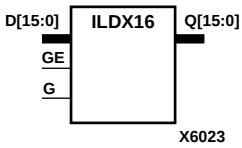
ILDX, 4, 8, 16

Macro: Transparent Input Data Latches









ILDX, ILDX4, ILDX8, and ILDX16 are single or multiple transparent data latches, which can be used to hold transient data entering a chip. The latch input (D) is connected to an IPAD or an IOPAD (without using an IBUF).

The latch is asynchronously cleared, output Low, when power is applied.

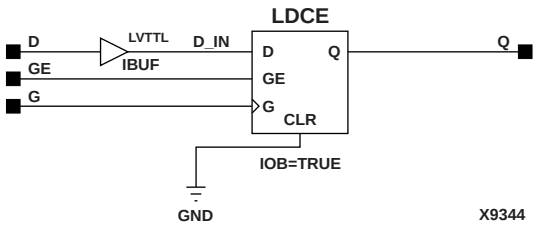
For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

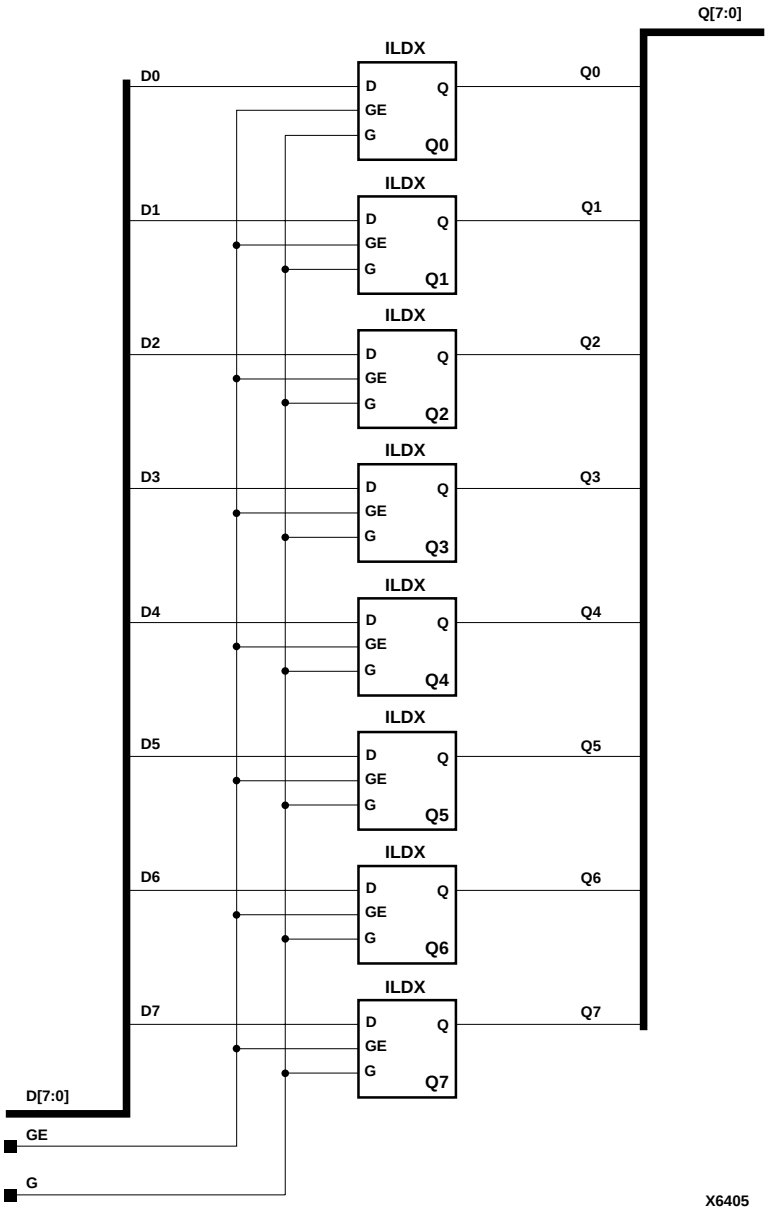
ILDXs and IFDXs

The ILDX is actually the input flip-flop master latch. Two different outputs can be accessed from the input flip-flop: one that responds to the level of the clock signal and another that responds to an edge of the clock signal. When using both outputs from the same input flip-flop, a transparent High latch (ILDX) corresponds to a falling edge-triggered flip-flop (IFDX_1). Similarly, a transparent Low latch (ILDX_1) corresponds to a rising edge-triggered flip-flop (IFDX).

Inputs			Outputs
GE	G	D	Q
0	X	X	No Change
1	0	X	No Change
1	1	D	D



ILDX Implementation for Spartan-3E



ILDX8 Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an ILDX, you would infer an LDCE and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr i to pack all input registers into the IOBs.

Available Attributes

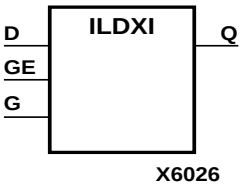
IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the non-registered path out of the IOB. Accepted values are 0 through 16 where 0 (no

additional delay added) is the default value. Increasing values increases the magnitude of the delay.

IFD_DELAY_VALUE – Specifies the amount of additional delay to add to the registered path within the IOB. Accepted values are AUTO, 0 through 8 where AUTO (adjust the delay to ensure no positive hold times are present) is the default value. Increasing values increases the magnitude of the delay.

ILDXI

Macro: Transparent Input Data Latch (Asynchronous Preset)



ILDXI is a transparent data latch, which can hold transient data entering a chip. When the gate input (G) is High, data on the input (D) appears on the output (Q). Data on the (D) input during the High-to-Low (G) transition is stored in the latch.

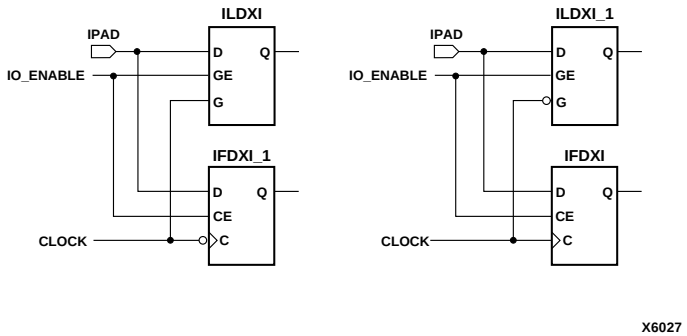
The latch is asynchronously preset, output High, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

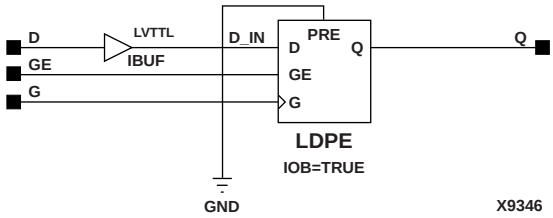
ILDXIs and IFDXIs

The ILDXI is actually the input flip-flop master latch. Two different outputs can be accessed from the input flip-flop: one that responds to the level of the clock signal and another that responds to an edge of the clock signal. When using both outputs from the same input flip-flop, a transparent High latch (ILDXI) corresponds to a falling edge-triggered flip-flop (IFDXI_1). Similarly, a transparent Low latch (ILDXI_1) corresponds to a rising edge-triggered flip-flop (IFDXI). See the following figure for legal IFDXI, IFDXI_1, ILDXI, and ILDXI_1 combinations.



Legal Combinations of IFDXI and ILDXI for a Single IOB

Inputs			Outputs
GE	G	D	Q
0	X	X	No Change
1	0	X	No Change
1	1	D	D
1	↓	D	D



ILDXI Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an ILDXI, you would infer an LDPE and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr i to pack all input registers into the IOBs.

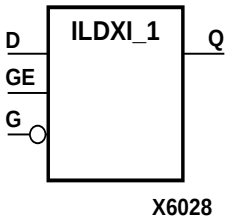
Available Attributes

IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the non-registered path out of the IOB. Accepted values are 0 through 16 where 0 (no additional delay added) is the default value. Increasing values increases the magnitude of the delay.

IFD_DELAY_VALUE – Specifies the amount of additional delay to add to the registered path within the IOB. Accepted values are AUTO, 0 through 8 where AUTO (adjust the delay to ensure no positive hold times are present) is the default value. Increasing values increases the magnitude of the delay.

ILDXI_1

Macro: Transparent Input Data Latch with Inverted Gate (Asynchronous Preset)



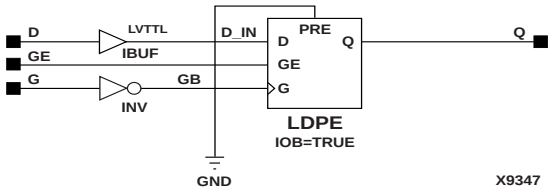
ILDXI_1 is a transparent data latch, which can hold transient data entering a chip. The latch is asynchronously preset, output High, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

For information on legal IFDXI, IFDXI_1, ILDXI, and ILDXI_1 combinations, see "ILDXI".

Inputs			Outputs
GE	G	D	Q
0	X	X	No Change
1	1	X	No Change
1	0	D	D
1	↑	D	D



ILDXI_1 Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an ILDXI_1, you would infer an LDPE_1 and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr i to pack all input registers into the IOBs.

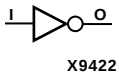
Available Attributes

IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the non-registered path out of the IOB. Accepted values are 0 through 16 where 0 (no additional delay added) is the default value. Increasing values increases the magnitude of the delay.

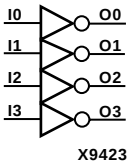
IFD_DELAY_VALUE – Specifies the amount of additional delay to add to the registered path within the IOB. Accepted values are AUTO, 0 through 8 where AUTO (adjust the delay to ensure no positive hold times are present) is the default value. Increasing values increases the magnitude of the delay.

INV, 4, 8, 16

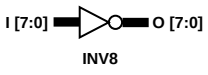
Primitive and Macros: Single and Multiple Inverters



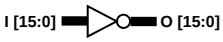
X9422



X9423

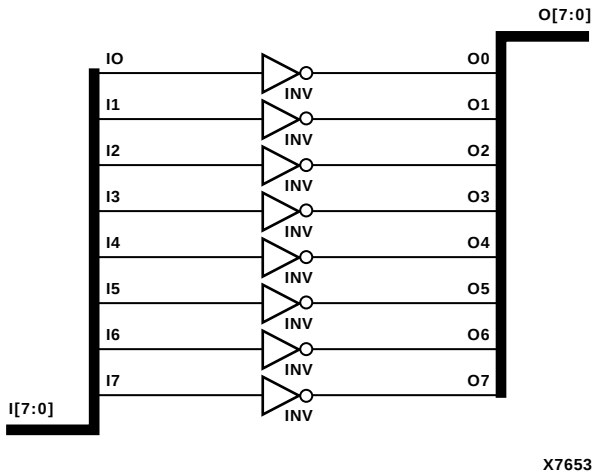


X9853



X9854

INV, INV4, INV8, and INV16 are single and multiple inverters that identify signal inversions in a schematic.



X7653

INV8 Implementation of Spartan-3E

Usage

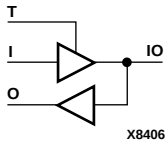
This design element can be instantiated or inferred.

For More Information

Consult the Spartan-3E Data Sheets.

IOBUF

Primitive: Bi-Directional Buffer with Selectable I/O Interface



For Spartan-3E, IOBUF is a bi-directional buffer whose I/O interface corresponds to a specific I/O standard. You can attach an IOSTANDARD attribute to an IOBUF instance.

IOBUF components that use the LVTTL, LVCMOS15, LVCMOS18, LVCMOS25, LVCMOS33 signaling standards have selectable capacitance drive and slew rates using the capacitance DRIVE and SLEW constraints. The defaults are CAPACITANCE = DONT_CARE, DRIVE=12 mA, and SLOW slew.

IOBUFs are composites of IBUF and OBUFT elements. The O output is X (unknown) when IO (input/output) is Z. IOBUFs can be implemented as interconnections of their component elements.

Inputs		Bidirectional	Outputs
T	I	IO	O
1	X	Z	X
0	1	1	1
0	0	0	0

Usage

These design elements are instantiated and inferred.

Available Attributes

IOSTANDARD – Specifies the I/O standard to configure this input.

IBUF_DELAY_VALUE – Specifies the amount of additional delay to add to the non-registered path out of the IOB. Accepted values are 0 through 16 where 0 (no additional delay added) is the default value. Increasing values increases the magnitude of the delay.

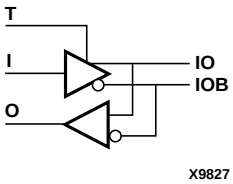
IFD_DELAY_VALUE – Specifies the amount of additional delay to add to the registered path within the IOB. Accepted values are AUTO, 0 through 8 where AUTO (adjust the delay to ensure no positive hold times are present) is the default value. Increasing values increases the magnitude of the delay.

For More Information

Consult the Spartan-3E Data Sheets.

IOBUFDS

Primitive: 3-State Differential Signaling I/O Buffer with Active Low Output Enable



IOBUFDS is a single 3-state, differential signaling input/output buffer with active Low output enable.

Inputs		Bidirectional		Outputs
I	T	IO	IOB	O
X	1	Z	Z	No Change
0	0	0	1	0
1	0	1	0	1

Usage

This design element is instantiated rather than inferred.

Available Attributes

DRIVE

IOSTANDARD

SLEW

IBUF_DELAY_VALUE

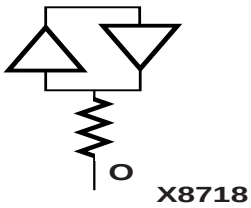
IFD_DELAY_VALUE

For More Information

Consult the Spartan-3E Data Sheets.

KEEPER

Primitive: KEEPER Symbol



KEEPER is a weak keeper element used to retain the value of the net connected to its bidirectional O pin. For example, if a logic 1 is being driven onto the net, KEEPER drives a weak/resistive 1 onto the net. If the net driver is then 3-stated, KEEPER continues to drive a weak/resistive 1 onto the net.

Usage

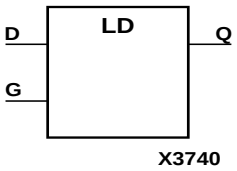
This design element is instantiated rather than inferred.

For More Information

Consult the Spartan-3E Data Sheets.

LD

Primitive: Transparent Data Latch



LD is a transparent data latch. The data output (Q) of the latch reflects the data (D) input while the gate enable (G) input is High. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains Low.

The latch is asynchronously cleared, output Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs		Outputs
G	D	Q
1	D	D
0	X	No Change
↓	D	D

Usage

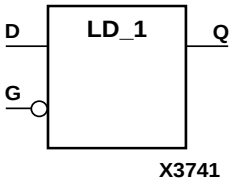
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit	0 or 1	All zeroes	Sets the initial value of Q output after configuration

LD_1

Primitive: Transparent Data Latch with Inverted Gate



LD_1 is a transparent data latch with an inverted gate. The data output (Q) of the latch reflects the data (D) input while the gate enable (G) input is Low. The data on the (D) input during the Low-to-High gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains High.

The latch is asynchronously cleared with Low output when power is applied, or when global reset is active.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs		Outputs
G	D	Q
0	D	D
1	X	No Change
↑	D	D

Usage

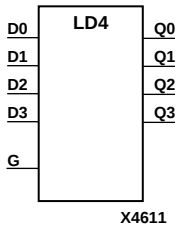
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

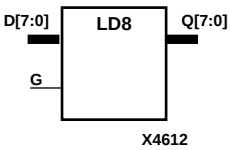
Available Attributes

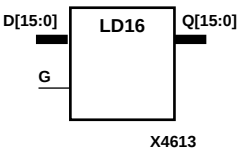
Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit	0 OR 1	0	Sets the initial value of Q output after configuration

LD4, 8, 16

Macros: Multiple Transparent Data Latches







LD4, LD8, and LD16 have, respectively, 4, 8, and 16 transparent data latches with a common gate enable (G). The data output (Q) of the latch reflects the data (D) input while the gate enable (G) input is High. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains Low.

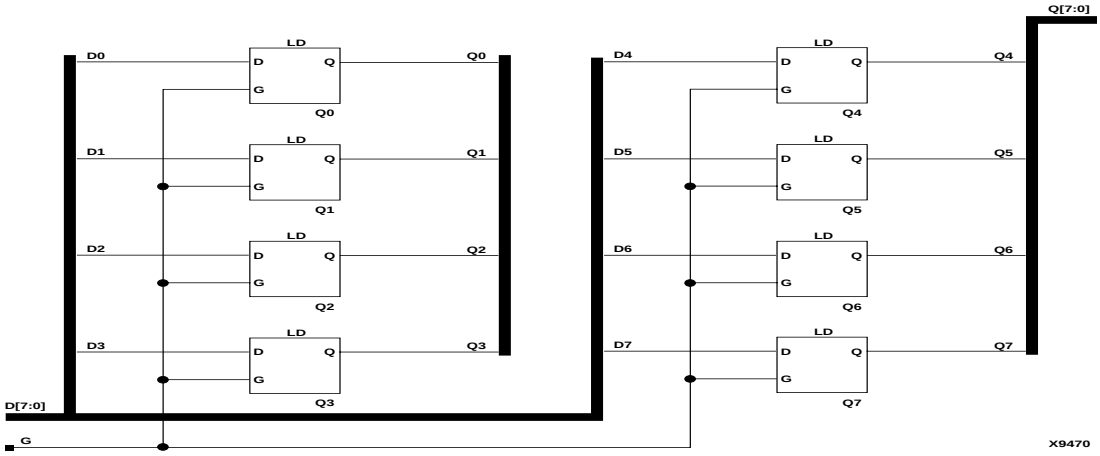
The latch is asynchronously cleared, output Low, when power is applied, or when global reset is active.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

See "LD" for information on single transparent data latches.

Inputs		Outputs
G	**D**	**Q**
1	D	D
0	X	No Change
↓	Dn	Dn



LD8 Implementation of Spartan-3E

Usage

These design elements are inferred rather than instantiated.

Available Attributes

LD4

Attribute	Type	Default	Description
INIT	4-Bit	All zeroes	Sets the initial value of Q output after configuration

LD8

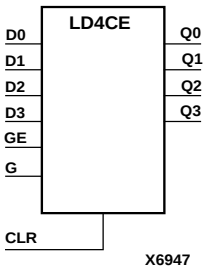
Attribute	Type	Default	Description
INIT	8-Bit	All zeroes	Sets the initial value of Q output after configuration

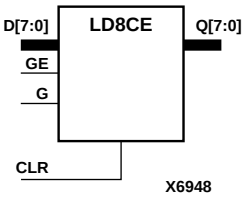
LD16

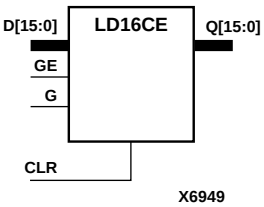
Attribute	Type	Default	Description
INIT	16-Bit	All zeroes	Sets the initial value of Q output after configuration

LD4CE, LD8CE, LD16CE

Macros: Transparent Data Latches with Asynchronous Clear and Gate Enable







LD4CE, LD8CE, and LD16CE have, respectively, 4, 8, and 16 transparent data latches with asynchronous clear and gate enable. When the asynchronous clear input (CLR) is High, it overrides the other inputs and resets the data (Q) outputs Low. (Q) reflects the data (D) inputs while the gate (G) and gate enable (GE) are High, and (CLR) is Low. If (GE) is Low, data on (D) cannot be latched. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) or (GE) remains Low.

The latch is asynchronously cleared with Low output when power is applied, or when global reset is active.

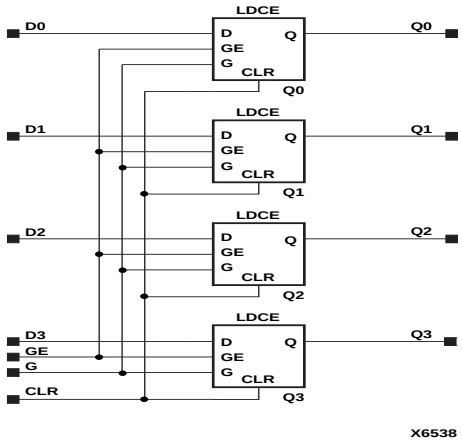
For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

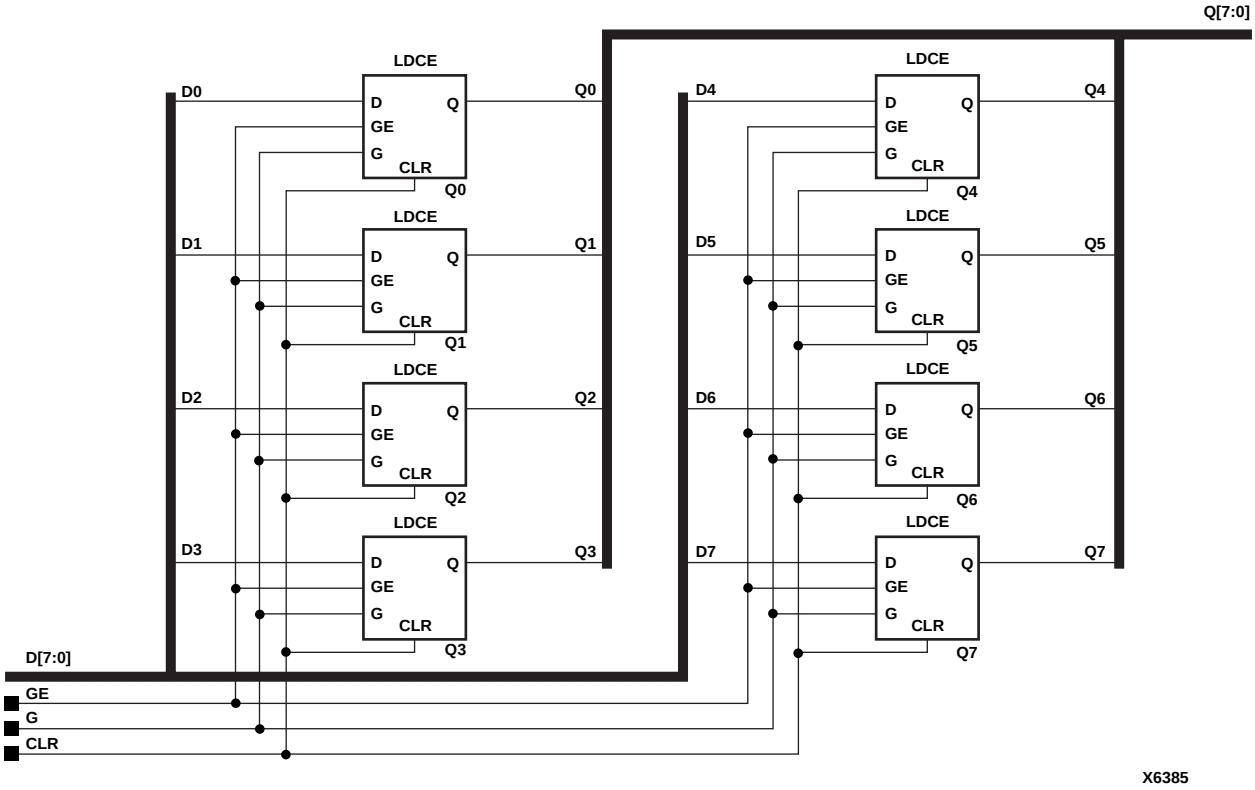
Inputs				Outputs
CLR	GE	G	Dn	Qn
1	X	X	X	0
0	0	X	X	No Change
0	1	1	Dn	Dn
0	1	0	X	No Change
0	1	↓	Dn	Dn

Dn = referenced input, for example, D0, D1, D2

Qn = referenced output, for example, Q0, Q1, Q2



LD4CE Implementation for Spartan-3E



LD8CE Implementation for Spartan-3E

Usage

These design elements are supported for instantiation only.

Available Attributes

LD4CE

Attribute	Type	Default	Description
INIT	4-Bit	All zeroes	Sets the initial value of Q output after configuration

LD8CE

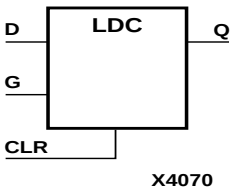
Attribute	Type	Default	Description
INIT	8-Bit	All zeroes	Sets the initial value of Q output after configuration

LD16CE

Attribute	Type	Default	Description
INIT	16-Bit	All zeroes	Sets the initial value of Q output after configuration

LDC

Primitive: Transparent Data Latch with Asynchronous Clear



LDC is a transparent data latch with asynchronous clear. When the asynchronous clear input (CLR) is High, it overrides the other inputs and resets the data (Q) output Low. (Q) reflects the data (D) input while the gate enable (G) input is High and (CLR) is Low. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains low.

The latch is asynchronously cleared with Low output when power is applied, or when global reset is active.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
CLR	G	D	Q
1	X	X	0
0	1	D	D
0	0	X	No Change
0	↓	D	D

Usage

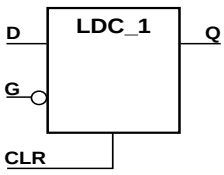
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit	0 or 1	0	Sets the initial value of Q output after configuration

LDC_1

Primitive: Transparent Data Latch with Asynchronous Clear and Inverted Gate



X3752

LDC_1 is a transparent data latch with asynchronous clear and inverted gate. When the asynchronous clear input (CLR) is High, it overrides the other inputs (D and G) and resets the data (Q) output Low. (Q) reflects the data (D) input while the gate enable (G) input and CLR are Low. The data on the (D) input during the Low-to-High gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains High.

The latch is asynchronously cleared with Low output when power is applied, or when global reset is active.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
CLR	G	D	Q
1	X	X	0
0	0	D	D
0	1	X	No Change
0	↑	D	D

Usage

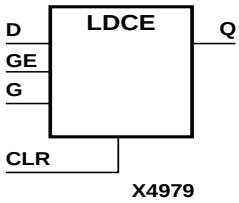
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit	0 or 1	0	Sets the initial value of Q output after configuration

LDCE

Primitive: Transparent Data Latch with Asynchronous Clear and Gate Enable



LDCE is a transparent data latch with asynchronous clear and gate enable. When the asynchronous clear input (CLR) is High, it overrides the other inputs and resets the data (Q) output Low. Q reflects the data (D) input while the gate (G) input and gate enable (GE) are High and CLR is Low. If (GE) is Low, data on (D) cannot be latched. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) or (GE) remains low.

The latch is asynchronously cleared with Low output when power is applied, or when global reset is active.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
CLR	GE	G	D	Q
1	X	X	X	0
0	0	X	X	No Change
0	1	1	D	D
0	1	0	X	No Change
0	1	↓	D	D

Usage

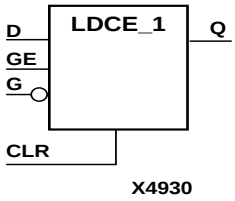
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit	0 or 1	0	Sets the initial value of Q output after configuration

LDCE_1

Primitive: Transparent Data Latch with Asynchronous Clear, Gate Enable, and Inverted Gate



LDCE_1 is a transparent data latch with asynchronous clear, gate enable, and inverted gate. When the asynchronous clear input (CLR) is High, it overrides the other inputs and resets the data (Q) output Low. (Q) reflects the data (D) input while the gate (G) input and (CLR) are Low and gate enable (GE) is High. The data on the (D) input during the Low-to-High gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains High or (GE) remains Low.

The latch is asynchronously cleared with Low output when power is applied, or when global reset is active.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
CLR	GE	G	D	Q
1	X	X	X	0
0	0	X	X	No Change
0	1	0	D	D
0	1	1	X	No Change
0	1	↑	D	D

Usage

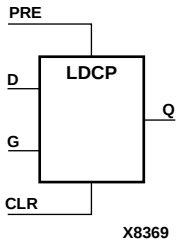
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit	0 or 1	0	Sets the initial value of Q output after configuration

LDCP

Primitive: Transparent Data Latch with Asynchronous Clear and Preset



LDCP is a transparent data latch with data (D), asynchronous clear (CLR) and preset (PRE) inputs. When (CLR) is High, it overrides the other inputs and resets the data (Q) output Low. When PRE is High and (CLR) is low, it presets the data (Q) output High. (Q) reflects the data (D) input while the gate (G) input is High and (CLR) and PRE are Low. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains Low.

The latch is asynchronously cleared, output Low, when power is applied, or when global reset is active.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
CLR	PRE	G	D	Q
1	X	X	X	0
0	1	X	X	1
0	0	1	D	D
0	0	0	X	No Change
0	0	↓	D	D

Usage

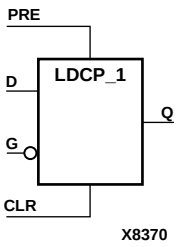
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

INIT - Specifies the initial value upon power-up or the assertion of GSR for the (Q) port. This attribute may be set to 1 or 0.

LDCP_1

Primitive: Transparent Data Latch with Asynchronous Clear and Preset and Inverted Gate



LDCP_1 is a transparent data latch with data (D), asynchronous clear (CLR), preset (PRE) inputs, and inverted gate (G). When (CLR) is High, it overrides the other inputs and resets the data (Q) output Low. When (PRE) is High and (CLR) is Low, it presets the data (Q) output High. (Q) reflects the data (D) input while gate (G) input, (CLR), and (PRE) are Low. The data on the (D) input during the Low-to-High gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains High.

The latch is asynchronously cleared, output Low, when power is applied, or when global reset is active.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
CLR	**PRE**	**G**	**D**	**Q**
1	X	X	X	0
0	1	X	X	1
0	0	0	D	D
0	0	1	X	No Change
0	0	↑	D	D

Usage

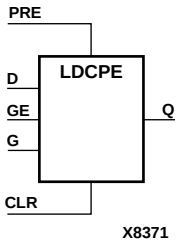
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

INIT - Specifies the initial value upon power-up or the assertion of GSR for the (Q) port. This attribute may be set to 1 or 0.

LDCPE

Primitive: Transparent Data Latch with Asynchronous Clear and Preset and Gate Enable



LDCPE is a transparent data latch with data (D), asynchronous clear (CLR), asynchronous preset (PRE), and gate enable (GE). When (CLR) is High, it overrides the other inputs and resets the data (Q) output Low. When (PRE) is High and (CLR) is Low, it presets the data (Q) output High. Q reflects the data (D) input while the gate (G) input and gate enable (GE) are High and (CLR) and PRE are Low. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the Q output remains unchanged as long as (G) or (GE) remains Low.

The latch is asynchronously cleared, output Low, when power is applied, or when global reset is active.

For Spartan-3E power on conditions are simulated when global set/reset (GSR) is active.

GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs					Outputs
CLR	PRE	GE	G	D	Q
1	X	X	X	X	0
0	1	X	X	X	1
0	0	0	X	X	No Change
0	0	1	1	0	0
0	0	1	1	1	1
0	0	1	0	X	No Change
0	0	1	↓	D	D

Usage

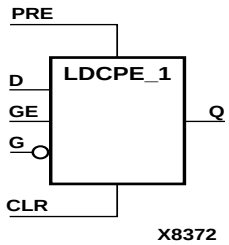
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

Attribute	Type	Allowed Values	Default	Description
INIT	1-Bit	0 or 1	0	Sets the initial value of Q output after configuration

LDCPE_1

Primitive: Transparent Data Latch with Asynchronous Clear and Preset, Gate Enable, and Inverted Gate



LDCPE_1 is a transparent data latch with data (D), asynchronous clear (CLR), asynchronous preset (PRE), gate enable (GE), and inverted gate (G). When (CLR) is High, it overrides the other inputs and resets the data (Q) output Low. When PRE is High and (CLR) is Low, it presets the data (Q) output High. (Q) reflects the data (D) input while gate enable (GE) is High and gate (G), (CLR), and (PRE) are Low. The data on the (D) input during the Low-to-High gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) is High or (GE) is Low.

The latch is asynchronously cleared, output Low, when power is applied, or when global reset is active.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs					Outputs
CLR	PRE	GE	G	D	Q
1	X	X	X	X	0
0	1	X	X	X	1
0	0	0	X	X	No Change
0	0	1	0	D	D
0	0	1	1	X	No Change
0	0	1	↑	D	D

Usage

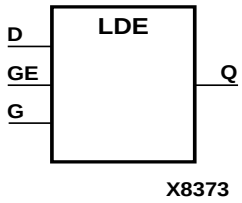
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

INIT - Specifies the initial value upon power-up or the assertion of GSR for the (Q) port. This attribute may be set to 1 or 0.

LDE

Primitive: Transparent Data Latch with Gate Enable



LDE is a transparent data latch with data (D) and gate enable (GE) inputs. Output (Q) reflects the data (D) while the gate (G) input and gate enable (GE) are High. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) or (GE) remains Low.

The latch is asynchronously cleared, output Low, when power is applied, or when global reset is active.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
GE	G	D	Q
0	X	X	No Change
1	1	D	D
1	0	X	No Change
1	↓	D	D

Usage

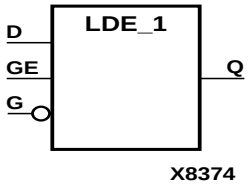
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

INIT - Specifies the initial value upon power-up or the assertion of GSR for the (Q) port. This attribute may be set to 1 or 0.

LDE_1

Primitive: Transparent Data Latch with Gate Enable and Inverted Gate



LDE_1 is a transparent data latch with data (D), gate enable (GE), and inverted gate (G). Output (Q) reflects the data (D) while the gate (G) input is Low and gate enable (GE) is High. The data on the (D) input during the Low-to-High gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) is High or (GE) is Low.

The latch is asynchronously cleared, output Low, when power is applied, or when global reset is active.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
GE	G	D	Q
0	X	X	No Change
1	0	D	D
1	1	X	No Change
1	↑	D	D

Usage

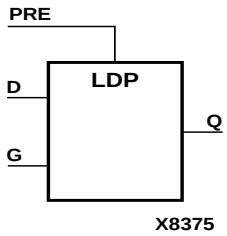
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

INIT - Specifies the initial value upon power-up or the assertion of GSR for the (Q) port. This attribute may be set to 1 or 0.

LDP

Primitive: Transparent Data Latch with Asynchronous Preset



LDP is a transparent data latch with asynchronous preset (PRE). When the (PRE) input is High, it overrides the other inputs and presets the data (Q) output High. (Q) reflects the data (D) input while gate (G) input is High and (PRE) is Low. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains Low.

The latch is asynchronously preset, output High, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
PRE	G	D	Q
1	X	X	1
0	1	0	0
0	1	1	1
0	0	X	No Change
0	↓	D	D

Usage

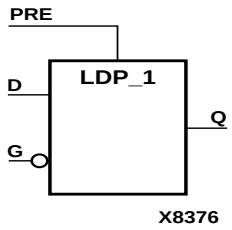
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

INIT - Specifies the initial value upon power-up or the assertion of GSR for the (Q) port. This attribute may be set to 1 or 0.

LDP_1

Primitive: Transparent Data Latch with Asynchronous Preset and Inverted Gate



LDP_1 is a transparent data latch with asynchronous preset (PRE) and inverted gate (G). When the (PRE) input is High, it overrides the other inputs and presets the data (Q) output High. (Q) reflects the data (D) input while gate (G) input and (PRE) are Low. The data on the (D) input during the Low-to-High gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains High.

The latch is asynchronously preset, output High, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
PRE	G	D	Q
1	X	X	1
0	0	D	D
0	1	X	No Change
0	↑	D	D

Usage

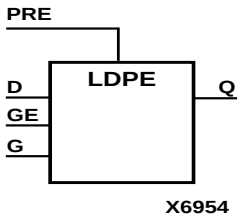
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

INIT - Specifies the initial value upon power-up or the assertion of GSR for the (Q) port. This attribute may be set to 1 or 0.

LDPE

Primitive: Transparent Data Latch with Asynchronous Preset and Gate Enable



LDPE is a transparent data latch with asynchronous preset and gate enable. When the asynchronous preset (PRE) is High, it overrides the other input and presets the data (Q) output High. Q reflects the data (D) input while the gate (G) input and gate enable (GE) are High. The data on the (D) input during the High-to-Low gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) or (GE) remains Low.

The latch is asynchronously preset, output High, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
PRE	GE	G	D	Q
1	X	X	X	1
0	0	X	X	No Change
0	1	1	D	D
0	1	0	X	No Change
0	1	↓	D	D

Usage

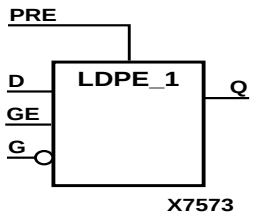
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

Available Attributes

INIT - Specifies the initial value upon power-up or the assertion of GSR for the (Q) port. This attribute may be set to 1 or 0.

LDPE_1

Primitive: Transparent Data Latch with Asynchronous Preset, Gate Enable, and Inverted Gate



LDPE_1 is a transparent data latch with asynchronous preset, gate enable, and inverted gate. When the asynchronous preset (PRE) is High, it overrides the other input and presets the data (Q) output High. (Q) reflects the data (D) input while the gate (G) and (PRE) are Low and gate enable (GE) is High.

The data on the (D) input during the Low-to-High gate transition is stored in the latch. The data on the (Q) output remains unchanged as long as (G) remains High or (GE) remains Low.

The latch is asynchronously preset, output High, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs
PRE	GE	G	D	Q
1	X	X	X	1
0	0	X	X	No Change
0	1	0	D	D
0	1	1	X	No Change
0	1	↑	D	D

Usage

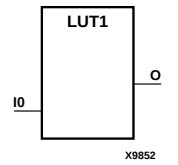
This design element typically should be inferred in the design code; however, the element can be instantiated for cases where strict placement control, relative placement control, or initialization attributes need to be applied.

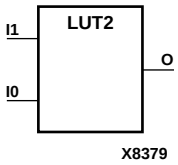
Available Attributes

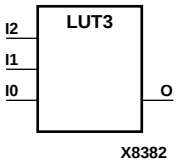
INIT - Specifies the initial value upon power-up or the assertion of GSR for the (Q) port. This attribute may be set to 1 or 0.

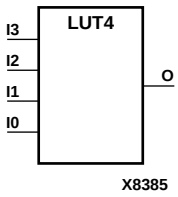
LUT1, 2, 3, 4

Primitive: 1-, 2-, 3-, 4-Bit Look-Up-Table with General Output









LUT1, LUT2, LUT3, and LUT4 are, respectively, 1-, 2-, 3-, and 4-bit look-up-tables (LUTs) with general output (O).

A mandatory INIT attribute, with an appropriate number of hexadecimal digits for the number of inputs, must be attached to the LUT to specify its function.

LUT1 provides a look-up-table version of a buffer or inverter.

LUTs are the basic Spartan-3E building blocks. Two LUTs are available in each CLB slice; four LUTs are available in each CLB. The variants, "LUT1_D, LUT2_D, LUT3_D, LUT4_D" and "LUT1_L, LUT2_L, LUT3_L, LUT4_L" provide additional types of outputs that can be used by different timing models for more accurate pre-layout timing estimation.

LUT3 Function Table

Inputs			Outputs
I2	I1	I0	O
0	0	0	INIT[0]
0	0	1	INIT[1]
0	1	0	INIT[2]
0	1	1	INIT[3]
1	0	0	INIT[4]
1	0	1	INIT[5]
1	1	0	INIT[6]
1	1	1	INIT[7]

INIT = binary equivalent of the hexadecimal number assigned to the INIT attribute

Usage

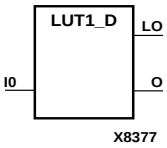
LUTs are generally inferred with the logic portions of the HDL code. Xilinx suggests that you instantiate LUTs only if you have a need to implicitly specify the logic mapping, or if you need to manually place or relationally place the logic.

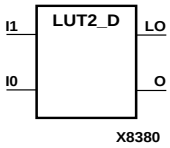
For More Information

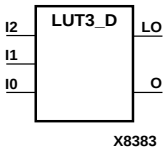
Consult the Spartan-3E Data Sheets.

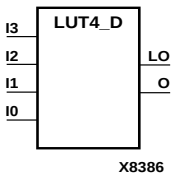
LUT1_D, LUT2_D, LUT3_D, LUT4_D

Primitive: 1-, 2-, 3-, 4-Bit Look-Up-Table with Dual Output









LUT1_D, LUT2_D, LUT3_D, and LUT4_D are, respectively, 1-, 2-, 3-, and 4-bit look-up-tables (LUTs) with two functionally identical outputs, O and LO. The O output is a general interconnect. The LO output is used to connect to another output within the same CLB slice and to the fast connect buffer.

A mandatory INIT attribute, with an appropriate number of hexadecimal digits for the number of inputs, must be attached to the LUT to specify its function.

LUT1_D provides a look-up-table version of a buffer or inverter.

See also "LUT1, 2, 3, 4" and "LUT1_L, LUT2_L, LUT3_L, LUT4_L."

LUT3_D Function Table

Inputs			Outputs	
I2	I1	I0	O	LO
0	0	0	INIT[0]	INIT[0]
0	0	1	INIT[1]	INIT[1]
0	1	0	INIT[2]	INIT[2]
0	1	1	INIT[3]	INIT[3]
1	0	0	INIT[4]	INIT[4]
1	0	1	INIT[5]	INIT[5]
1	1	0	INIT[6]	INIT[6]
1	1	1	INIT[7]	INIT[7]

INIT = binary equivalent of the hexadecimal number assigned to the INIT attribute

Usage

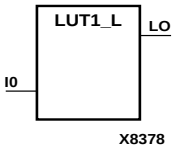
LUTs are generally inferred with the logic portions of the HDL code. Xilinx suggests that you instantiate LUTs only if you have a need to implicitly specify the logic mapping, or if you need to manually place or relationally place the logic.

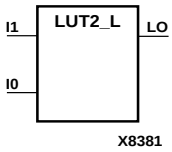
For More Information

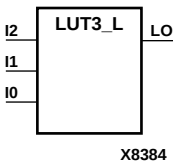
Consult the Spartan-3E Data Sheets.

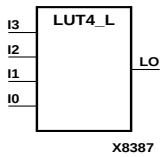
LUT1_L, LUT2_L, LUT3_L, LUT4_L

Primitive: 1-, 2-, 3-, 4-Bit Look-Up-Table with Local Output









LUT1_L, LUT2_L, LUT3_L, and LUT4_L are, respectively, 1-, 2-, 3-, and 4- bit look-up-tables (LUTs) with a local output (LO) that is used to connect to another output within the same CLB slice and to the fast connect buffer.

A mandatory INIT attribute, with an appropriate number of hexadecimal digits for the number of inputs, must be attached to the LUT to specify its function.

LUT1_L provides a look-up-table version of a buffer or inverter.

See also "LUT1, 2, 3, 4" and "LUT1_D, LUT2_D, LUT3_D, LUT4_D."

LUT3_L Function Table

Inputs			Outputs
I2	I1	I0	LO
0	0	0	INIT[0]
0	0	1	INIT[1]
0	1	0	INIT[2]
0	1	1	INIT[3]
1	0	0	INIT[4]
1	0	1	INIT[5]
1	1	0	INIT[6]
1	1	1	INIT[7]

INIT = binary equivalent of the hexadecimal number assigned to the INIT attribute

Usage

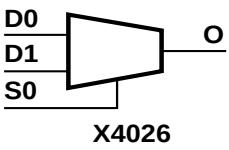
LUTs are generally inferred with the logic portions of the HDL code. Xilinx suggests that you instantiate LUTs only if you have a need to implicitly specify the logic mapping, or if you need to manually place or relationally place the logic.

For More Information

Consult the Spartan-3E Data Sheets.

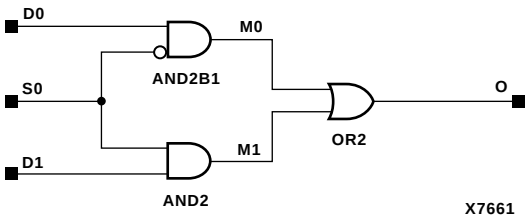
M2_1

Macro: 2-to-1 Multiplexer



The M2_1 multiplexer chooses one data bit from two sources (D1 or D0) under the control of the select input (S0). The output (O) reflects the state of the selected data input. When Low, S0 selects D0 and when High, S0 selects D1.

Inputs			Outputs
S0	**D1**	**D0**	**O**
1	D1	X	D1
0	X	D0	D0



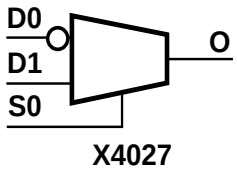
M2_1 Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

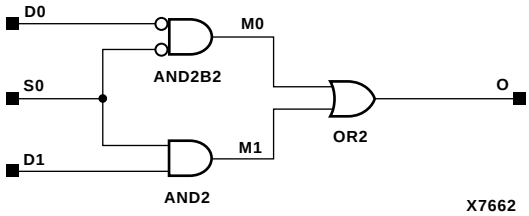
M2_1B1

Macro: 2-to-1 Multiplexer with D0 Inverted



The M2_1B1 multiplexer chooses one data bit from two sources (D1 or D0) under the control of select input (S0). When S0 is Low, the output (O) reflects the inverted value of (D0). When S0 is High, (O) reflects the state of D1.

Inputs			Outputs
S0	D1	D0	O
1	1	X	1
1	0	X	0
0	X	1	0
0	X	0	1



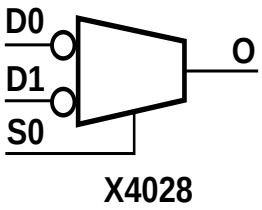
M2_1B1 Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated

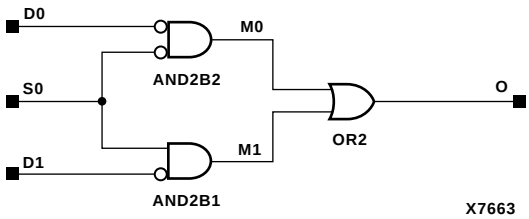
M2_1B2

Macro: 2-to-1 Multiplexer with D0 and D1 Inverted



The M2_1B2 multiplexer chooses one data bit from two sources (D1 or D0) under the control of select input (S0). When S0 is Low, the output (O) reflects the inverted value of D0. When S0 is High, O reflects the inverted value of D1.

Inputs			Outputs
S0	D1	D0	O
1	1	X	0
1	0	X	1
0	X	1	0
0	X	0	1



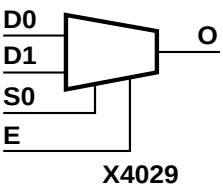
M2_1B2 Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

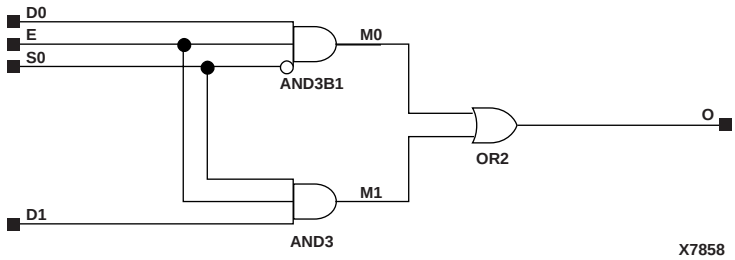
M2_1E

Macro: 2-to-1 Multiplexer with Enable



M2_1E is a 2-to-1 multiplexer with enable. When the enable input (E) is High, the M2_1E chooses one data bit from two sources (D1 or D0) under the control of select input (S0). When Low, S0 selects D0 and when High, S0 selects D1. When (E) is Low, the output is Low.

Inputs				Outputs
E	S0	D1	D0	O
0	X	X	X	0
1	0	X	1	1
1	0	X	0	0
1	1	1	X	1
1	1	0	X	0



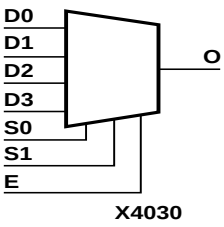
M2_1E Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

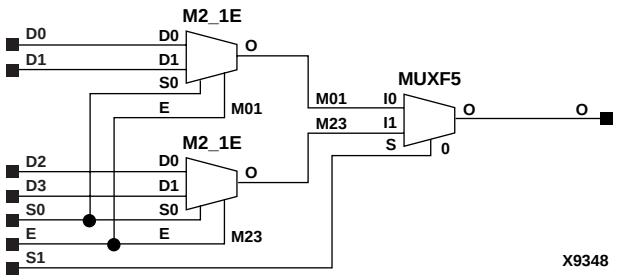
M4_1E

Macro: 4-to-1 Multiplexer with Enable



M4_1E is an 4-to-1 multiplexer with enable. When the enable input (E) is High, the M4_1E multiplexer chooses one data bit from four sources (D3, D2, D1, or D0) under the control of the select inputs (S1 – S0). The output (O) reflects the state of the selected input as shown in the truth table. When(E) is Low, the output is Low.

Inputs							Outputs
E	S1	S0	D0	D1	D2	D3	O
0	X	X	X	X	X	X	0
1	0	0	D0	X	X	X	D0
1	0	1	X	D1	X	X	D1
1	1	0	X	X	D2	X	D2
1	1	1	X	X	X	D3	D3



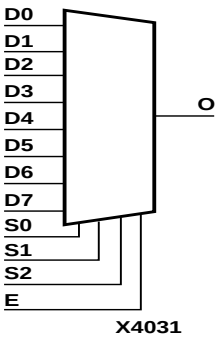
M4_1E Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated.

M8_1E

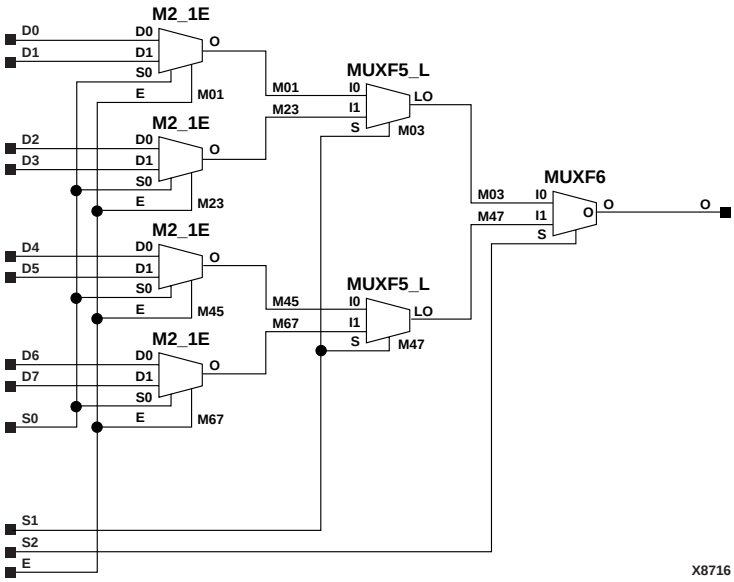
Macro: 8-to-1 Multiplexer with Enable



M8_1E is an 8-to-1 multiplexer with enable. When the enable input (E) is High, the M8_1E multiplexer chooses one data bit from eight sources (D7 – D0) under the control of the select inputs (S2 – S0). The output (O) reflects the state of the selected input as shown in the truth table. When (E) is Low, the output is Low.

Inputs					Outputs
E	S2	S1	S0	D7 – D0	O
0	X	X	X	X	0
1	0	0	0	D0	D0
1	0	0	1	D1	D1
1	0	1	0	D2	D2
1	0	1	1	D3	D3
1	1	0	0	D4	D4
1	1	0	1	D5	D5
1	1	1	0	D6	D6
1	1	1	1	D7	D7

Dn represents signal on the Dn input; all other data inputs are don't-cares (X).



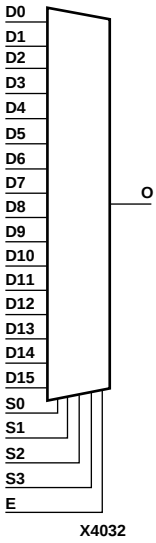
M8_1E Implementation for Spartan-3E

Usage

This design element is inferred rather than instantiated

M16_1E

Macro: 16-to-1 Multiplexer with Enable



M16_1E is a 16-to-1 multiplexer with enable. When the enable input (E) is High, the M16_1E multiplexer chooses one data bit from 16 sources (D15 – D0) under the control of the select inputs (S3 – S0). The output (O) reflects the state of the selected input as shown in the truth table. When (E) is Low, the output is Low.

Inputs						Outputs
E	**S3**	**S2**	**S1**	**S0**	**D15 – D0**	**O**
0	X	X	X	X	X	0
1	0	0	0	0	D0	D0
1	0	0	0	1	D1	D1
1	0	0	1	0	D2	D2
1	0	0	1	1	D3	D3
. . .	. . .	. . .	. . .	. . .	. . .	. . .
1	1	1	0	0	D12	D12
1	1	1	0	1	D13	D13
1	1	1	1	0	D14	D14
1	1	1	1	1	D15	D15

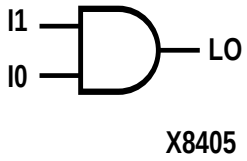
Dn represents signal on the Dn input; all other data inputs are don't-cares (X).

Usage

This design element is inferred rather than instantiated.

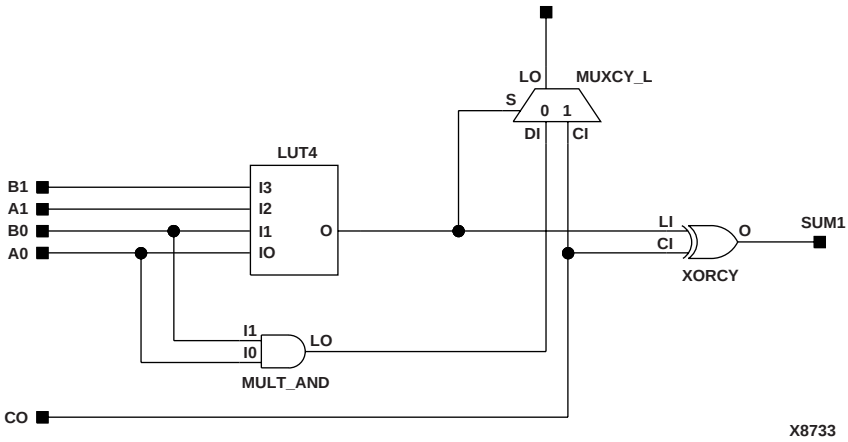
MULT_AND

Primitive: Fast Multiplier AND



MULT_AND is an AND component used exclusively for building fast and smaller multipliers. The I1 and I0 inputs *must* be connected to the I1 and I0 inputs of the associated LUT. The LO output *must* be connected to the DI input of the associated MUXCY, MUXCY_D, or MUXCY_L.

Inputs		Output
I1	I0	LO
0	0	0
0	1	0
1	0	0
1	1	1



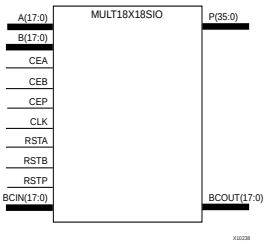
Example Multiplier Using MULT_AND

Usage

This design element can be instantiated and inferred.

MULT18X18SIO

Primitive: 18x18 Cascadable Signed Multiplier with Optional Input and Output registers, Clock Enable, and Synchronous Reset



The MULT18X18SIO is a 36-bit output, 18x18-bit input dedicated signed multiplier. This component can perform asynchronous multiplication operations when the attributes AREG, BREG and PREG are all set to 0. Alternatively, synchronous multiplication operations of different latency and performance characteristics can be performed when any combination of those attributes is set to 1. When using the multiplier in synchronous operation, the MULT18X18SIO features active high clock enables for each set of register banks in the multiplier, CEA, CEB and CEP, as well as synchronous resets, RSTA, RSTB, and RSTP. Multiple MULT18X18SIOs can be cascaded to create larger multiplication functions using the BCIN and BCOUT ports in combination with the B_INPUT attribute.

Usage

The MULT18X18SIO can be inferred by most synthesis tools using standard VHDL or Verilog notation for multiplication. Alternatively, Core Generator™ System and other IP can also create multiplication functions using this component. If preferred, the MULT18X18SIO may be instantiated into the VHDL or Verilog code to give full control over the implementation of the component. In order to change the default behavior of the MULT18X18SIO, attributes may be modified via the generic map (VHDL) or named parameter value assignment (Verilog) as a part of the instantiated component.

Available Attributes

AREG – Enables (1) or disables (0) the register bank capturing data entering the A port of the multiplier.

BREG – Enables (1) or disables (0) the register bank capturing data entering the B port of the multiplier.

B_INPUT – When set to "DIRECT" uses the data presented on the B input port for multiplication operations. When set to "CASCADE", uses data on the BCIN port for the multiplication operation.

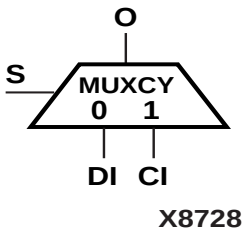
PREG – Enables (1) or disables (0) the register bank capturing data before exiting the P port of the multiplier.

For More Information

Consult the Spartan-3E Data Sheets.

MUXCY

Primitive: 2-to-1 Multiplexer for Carry Logic with General Output



MUXCY is used to implement a 1-bit high-speed carry propagate function. One such function can be implemented per slice, for a total of 4 bits per configurable logic block (CLB) for Spartan-3E.

The direct input (DI) of a slice is connected to the (DI input of the MUXCY. The carry in (CI) input of an LC is connected to the CI input of the MUXCY. The select input (S) of the MUXCY is driven by the output of the lookup table (LUT) and configured as a MUX function. The carry out (O) of the MUXCY reflects the state of the selected input and implements the carry out function of each LC. When Low, S selects DI; when High, S selects CI.

The variants, "MUXCY_D" and "MUXCY_L" provide additional types of outputs that can be used by different timing models for more accurate pre-layout timing estimation.

Inputs			Outputs
S	DI	CI	O
0	1	X	1
0	0	X	0
1	X	1	1
1	X	0	0

Usage

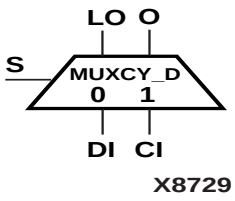
This design element can be instantiated and inferred.

For More Information

Consult the Spartan-3E Data Sheets.

MUXCY_D

Primitive: 2-to-1 Multiplexer for Carry Logic with Dual Output



MUXCY_D is used to implement a 1-bit high-speed carry propagate function. One such function can be implemented per logic cell (LC), for a total of 4-bits per configurable logic block (CLB). The direct input (DI) of an LC is connected to the DI input of the MUXCY_D. The carry in (CI) input of an LC is connected to the CI input of the MUXCY_D. The select input (S) of the MUX is driven by the output of the lookup table (LUT) and configured as an XOR function. The carry out (O and LO) of the MUXCY_D reflects the state of the selected input and implements the carry out function of each LC. When Low, S selects DI; when High, S selects CI.

Outputs O and LO are functionally identical. The O output is a general interconnect.

See also "MUXCY" and "MUXCY_L"

Inputs			Outputs	
S	DI	CI	O	LO
0	1	X	1	1
0	0	X	0	0
1	X	1	1	1
1	X	0	0	0

Usage

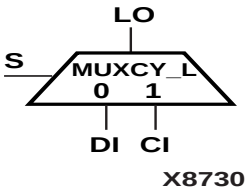
This design element can only be instantiated. Synthesis tools will make use of the MUXCY primitive, then MAP will make use of the MUXCY_D.

For More Information

Consult the Spartan-3E Data Sheets.

MUXCY_L

Primitive: 2-to-1 Multiplexer for Carry Logic with Local Output



MUXCY_L is used to implement a 1-bit high-speed carry propagate function. One such function can be implemented per logic cell (LC), for a total of 4-bits per configurable logic block (CLB). The direct input (DI) of an LC is connected to the DI input of the MUXCY_L. The carry in (CI) input of an LC is connected to the CI input of the MUXCY_L. The select input (S) of the MUXCY_L is driven by the output of the lookup table (LUT) and configured as an XOR function. The carry out (LO) of the MUXCY_L reflects the state of the selected input and implements the carry out function of each (LC). When Low, (S) selects DI; when High, (S) selects (CI).

See also "MUXCY" and "MUXCY_D"

Inputs			Outputs
S	DI	CI	LO
0	1	X	1
0	0	X	0
1	X	1	1
1	X	0	0

Usage

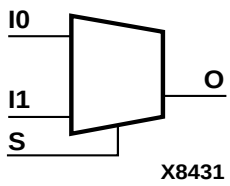
This design element can only be instantiated. Synthesis tools will make sue of the MUXCY primitive, then MAP will make use of the MUXCY_L.

For More Information

Consult the Spartan-3E Data Sheets.

MUXF5

Primitive: 2-to-1 Lookup Table Multiplexer with General Output



MUXF5 provides a multiplexer function in a CLB slice for creating a function-of-5 lookup table or a 4-to-1 multiplexer in combination with the associated lookup tables. The local outputs (LO) from the two lookup tables are connected to the I0 and I1 inputs of the MUXF5. The (S) input is driven from any internal net. When Low, (S) selects I0. When High, (S) selects I1.

The variants, "MUXF5_D" and "MUXF5_L", provide additional types of outputs that can be used by different timing models for more accurate pre-layout timing estimation.

Inputs			Outputs
S	I0	I1	O
0	1	X	1
0	0	X	0
1	X	1	1
1	X	0	0

Usage

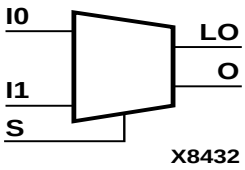
This design element can be instantiated and inferred.

For More Information

Consult the Spartan-3E Data Sheets.

MUXF5_D

Primitive: 2-to-1 Lookup Table Multiplexer with Dual Output



MUXF5_D provides a multiplexer function in a CLB slice for creating a function-of-5 lookup table or a 4-to-1 multiplexer in combination with the associated lookup tables. The local outputs (LO) from the two lookup tables are connected to the I0 and I1 inputs of the MUXF5. The S input is driven from any internal net. When Low, S selects I0. When High, S selects I1.

Outputs O and LO are functionally identical. The O output is a general interconnect. The LO output is used to connect to other inputs within the same CLB slice.

See also "MUXF5" and "MUXF5_L"

Inputs			Outputs	
S	I0	I1	O	LO
0	1	X	1	1
0	0	X	0	0
1	X	1	1	1
1	X	0	0	0

Usage

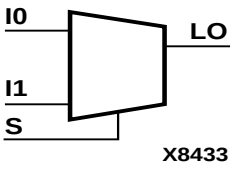
This design element can only be instantiated. Synthesis tools will make use of the MUXF5, then MAP will make use of the MUXF5_D.

For More Information

Consult the Spartan-3E Data Sheets.

MUXF5_L

Primitive: 2-to-1 Lookup Table Multiplexer with Local Output



MUXF5_L provides a multiplexer function in a CLB slice for creating a function-of-5 lookup table or a 4-to-1 multiplexer in combination with the associated lookup tables. The local outputs (LO) from the two lookup tables are connected to the I0 and I1 inputs of the MUXF5. The S input is driven from any internal net. When Low, S selects I0. When High, S selects I1.

The LO output is used to connect to other inputs within the same CLB slice.

See also "MUXF5" and "MUXF5_D".

Inputs			Output
S	I0	I1	LO
0	1	X	1
0	0	X	0
1	X	1	1
1	X	0	0

Usage

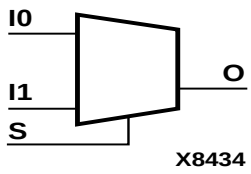
This design element can only be instantiated. Synthesis tools will make use of the MUXF5 primitive, then MAP will make use of the MUXF5_L.

For More Information

Consult the Spartan-3E Data Sheets.

MUXF6

Primitive: 2-to-1 Lookup Table Multiplexer with General Output



MUXF6 provides a multiplexer function in two slices for creating a function-of-6 lookup table or an 8-to-1 multiplexer in combination with the associated four lookup tables and two MUXF5s. The local outputs (LO) from the two MUXF5s in the CLB are connected to the I0 and I1 inputs of the MUXF6. The S input is driven from any internal net. When Low, (S) selects I0. When High, (S) selects I1.

The variants, "MUXF6_D" and "MUXF6_L", provide additional types of outputs that can be used by different timing models for more accurate pre-layout timing estimation.

Inputs			Outputs
S	**I0**	**I1**	**O**
0	1	X	1
0	0	X	0
1	X	1	1
1	X	0	0

Usage

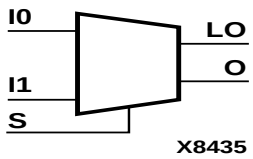
This design element can only be instantiated.

For More Information

Consult the Spartan-3E Data Sheets.

MUXF6_D

Primitive: 2-to-1 Lookup Table Multiplexer with Dual Output



MUXF6_D provides a multiplexer function in a two slices for creating a function-of-6 lookup table or an 8-to-1 multiplexer in combination with the associated four lookup tables and two MUXF5s. The local outputs (LO) from the two MUXF5s in the CLB are connected to the I0 and I1 inputs of the MUXF6. The (S) input is driven from any internal net. When Low, (S) selects I0. When High, (S) selects I1.

Outputs (O) and (LO) are functionally identical. The (O) output is a general interconnect. The (LO) output is used to connect to other inputs within the same CLB slice.

See also "MUXF6" and "MUXF6_L"

Inputs			Outputs	
S	I0	I1	O	LO
0	1	X	1	1
0	0	X	0	0
1	X	1	1	1
1	X	0	0	0

Usage

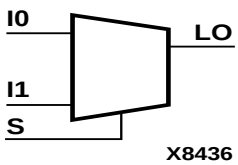
This design element can only be instantiated.

For More Information

Consult the Spartan-3E Data Sheets.

MUXF6_L

Primitive: 2-to-1 Lookup Table Multiplexer with Local Output



MUXF6_L provides a multiplexer function in a full, Spartan-3E CLB (two slices) for creating a function-of-6 lookup table or an 8-to-1 multiplexer in combination with the associated four lookup tables and two MUXF5s. The local outputs (LO) from the two MUXF5s in the (CLB) are connected to the I0 and I1 inputs of the MUXF6. The (S) input is driven from any internal net. When Low, (S) selects I0. When High, (S) selects I1.

The LO output is used to connect to other inputs within the same CLB slice.

See also "MUXF6" and "MUXF6_D".

Inputs			Output
S	I0	I1	LO
0	1	X	1
0	0	X	0
1	X	1	1
1	X	0	0

Usage

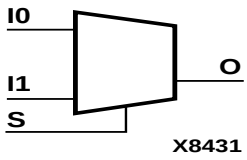
This design element can only be instantiated.

For More Information

Consult the Spartan-3E Data Sheets.

MUXF7

Primitive: 2-to-1 Lookup Table Multiplexer with General Output



MUXF7 provides a multiplexer function in a full Spartan-3E (four slices) for creating a function-of-7 lookup table or a 16-to-1 multiplexer in combination with the associated lookup tables. Local outputs (LO) of MUXF6 are connected to the I0 and I1 inputs of the MUXF7. The (S) input is driven from any internal net. When Low, (S) selects I0. When High,( )S selects I1.

The variants, "MUXF7_D" and "MUXF7_L", provide additional types of outputs that can be used by different timing models for more accurate pre-layout timing estimation.

Inputs			Outputs
S	**I0**	**I1**	**O**
0	I0	X	I0
1	X	I1	I1
X	0	0	0
X	1	1	1

Usage

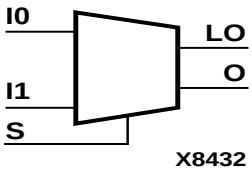
This design element can only be instantiated.

For More Information

Consult the Spartan-3E Data Sheets.

MUXF7_D

Primitive: 2-to-1 Lookup Table Multiplexer with Dual Output



MUXF7_D provides a multiplexer function in a full Spartan-3E CLB (four slices) for creating a function-of-7 lookup table or a 16-to-1 multiplexer in combination with the associated lookup tables. Local outputs (LO) of MUXF6 are connected to the I0 and I1 inputs of the MUXF7. The S input is driven from any internal net. When Low, (S) selects I0. When High, (S) selects I1.

Outputs O and LO are functionally identical. The O output is a general interconnect. The LO output is used to connect to other inputs within the same CLB slice.

See also "MUXF7" and "MUXF7_L".

Inputs			Outputs	
S	I0	I1	O	LO
0	I0	X	I0	I0
1	X	I1	I1	I1
X	0	0	0	0
X	1	1	1	1

Usage

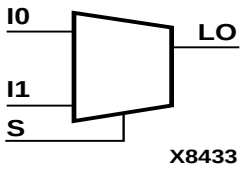
This design element can only be instantiated.

For More Information

Consult the Spartan-3E Data Sheets.

MUXF7_L

Primitive: 2-to-1 Lookup Table Multiplexer with Local Output



MUXF7_L provides a multiplexer function in a full Spartan-3E CLB (four slices) for creating a function-of-7 lookup table or a 16-to-1 multiplexer in combination with the associated lookup tables. Local outputs (LO) of MUXF6 are connected to the I0 and I1 inputs of the MUXF7. The S input is driven from any internal net. When Low, (S) selects I0. When High, (S) selects I1.

The LO output is used to connect to other inputs within the same CLB slice.

See also "MUXF7" and "MUXF7_D".

Inputs			Output
S	**I0**	**I1**	**LO**
0	I0	X	I0
1	X	I1	I1
X	0	0	0
X	1	1	1

Usage

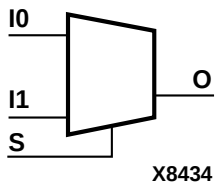
This design element can only be instantiated.

For More Information

Consult the Spartan-3E Data Sheets.

MUXF8

Primitive: 2-to-1 Lookup Table Multiplexer with General Output



MUXF8 provides a multiplexer function in eight slices for creating a function-of-8 lookup table or a 32-to-1 multiplexer in combination with the associated lookup tables, MUXF5s, MUXF6s, and MUXF7s. Local outputs (LO) of MUXF7 are connected to the I0 and I1 inputs of the MUXF8. T(h)e S input is driven from any internal net. When Low, (S) selects I0. When High, S selects I1.

See also "MUXF8_D" and "MUXF8_L".

Inputs			Outputs
S	**I0**	**I1**	**O**
0	I0	X	I0
1	X	I1	I1
X	0	0	0
X	1	1	1

Usage

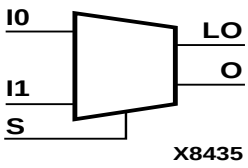
This design element can only be instantiated.

For More Information

Consult the Spartan-3E Data Sheets.

MUXF8_D

Primitive: 2-to-1 Lookup Table Multiplexer with Dual Output



MUXF8_D provides a multiplexer function in eight slices for creating a function-of-8 lookup table or a 32-to-1 multiplexer in combination with the associated four lookup tables and two MUXF8s. Local outputs (LO) of MUXF7 are connected to the I0 (a)nd I1 inputs of the MUXF8. The S input is driven from any internal net. When Low, S selects I0. When High, (S) selects I1.

Outputs O and LO are functionally identical. The O output is a general interconnect. The LO output is used to connect to other inputs within the same CLB slice.

See also "MUXF8" and "MUXF8_L".

Inputs			Outputs	
S	I0	I1	O	LO
0	I0	X	I0	I0
1	X	I1	I1	I1
X	0	0	0	0
X	1	1	1	1

Usage

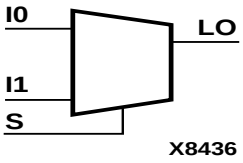
This design element can only be instantiated.

For More Information

Consult the Spartan-3E Data Sheets.

MUXF8_L

Primitive: 2-to-1 Lookup Table Multiplexer with Local Output



MUXF8_L provides a multiplexer function in eight slices for creating a function-of-8 lookup table or a 32-to-1 multiplexer in combination with the associated four lookup tables and two MUXF8s. Local outputs (LO) of MUXF7 are connected to the I0 and I1 inputs of the MUXF8. The S input is driven from any internal net. When Low, (S) selects I0. When High, (S) selects I1.

The LO output is used to connect to other inputs within the same CLB slice.

See also "MUXF8" and "MUXF8_D".

Inputs			Output
S	**I0**	**I1**	**LO**
0	I0	X	I0
1	X	I1	I1
X	0	0	0
X	1	1	1

Usage

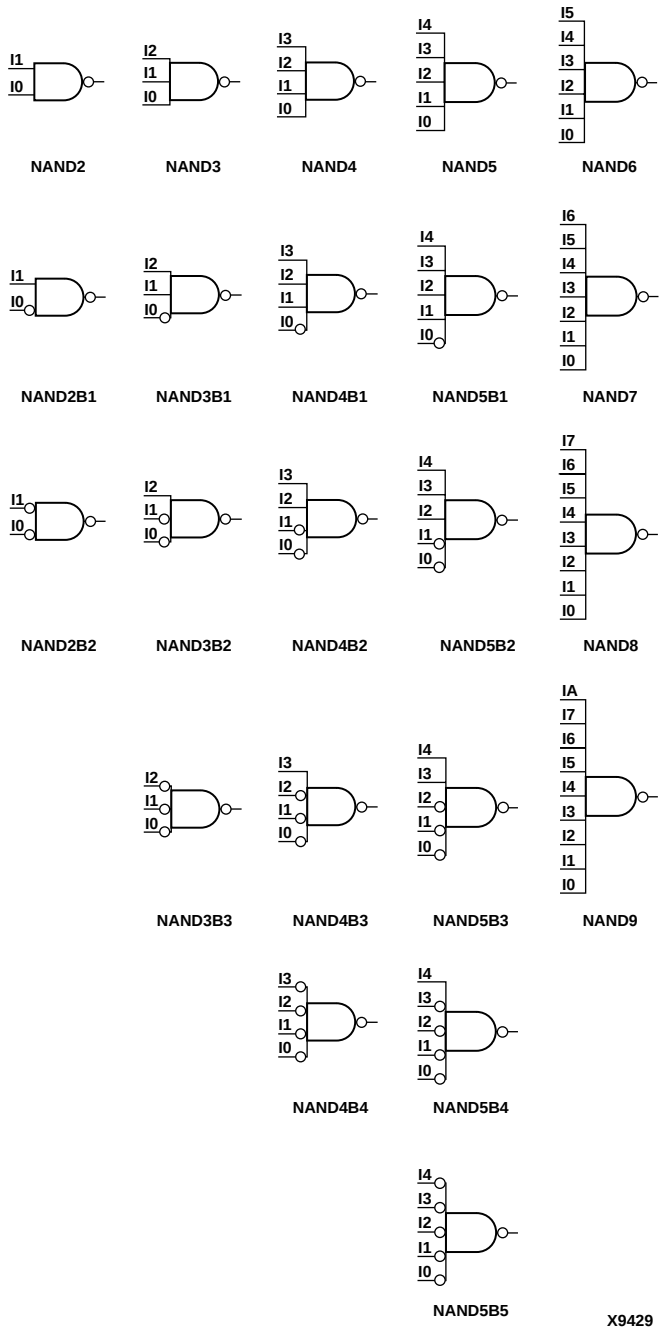
This design element can only be instantiated.

For More Information

Consult the Spartan-3E Data Sheets.

NAND2-9

Primitive: 2- to 9-Input NAND Gates with Inverted and Non-Inverted Inputs

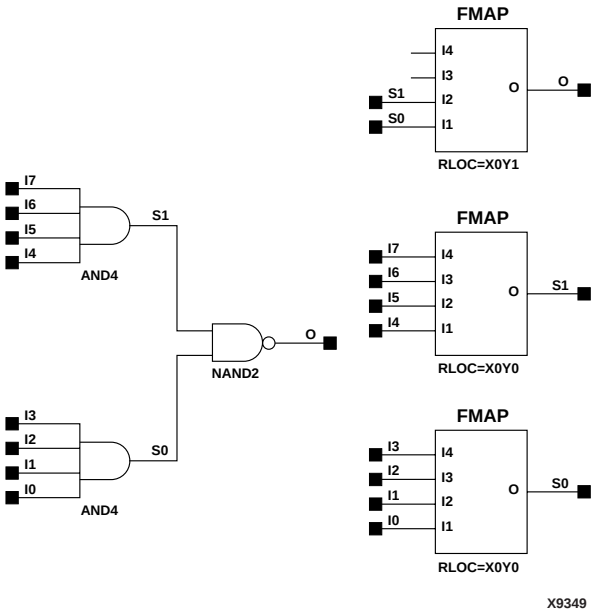


NAND Gate Representations

NAND gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NAND gates of six to nine inputs are available with only non-

inverting inputs. To invert inputs, use external inverters. Since each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

See "NAND12, 16" for information on additional NAND functions.



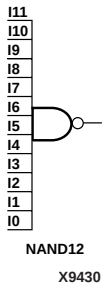
NAND8 Implementation Spartan-3E

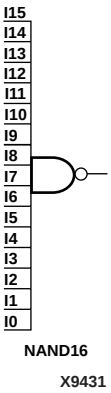
Usage

NAND2 through NAND5 are primitives that can be inferred or instantiated. NAND6 through NAND9 are macros which can be inferred.

NAND12, 16

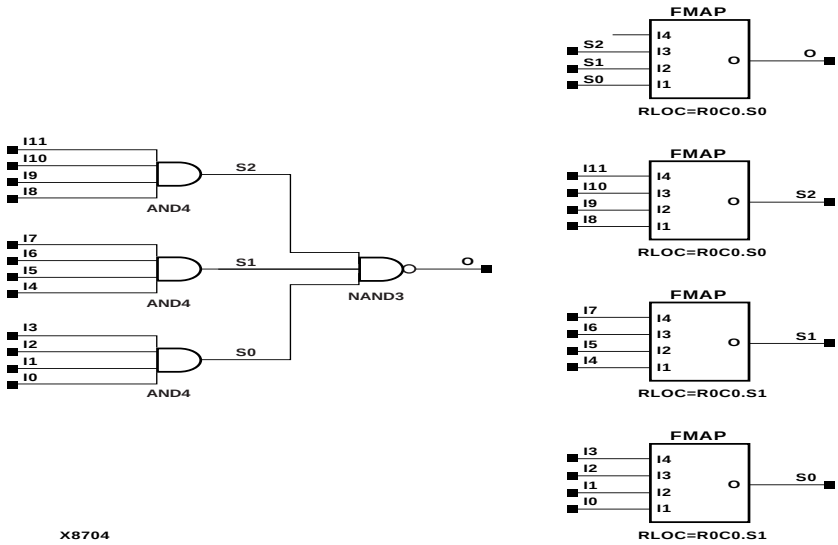
Primitive: 12- and 16-Input NAND Gates with Non-Inverted Inputs



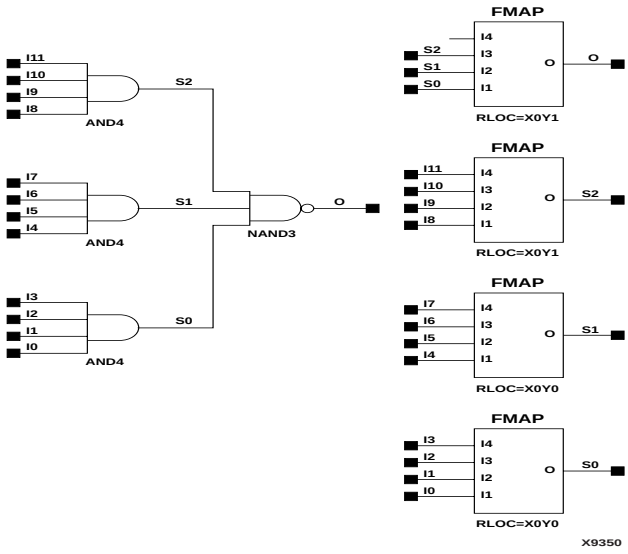


The NAND function is performed in the Configurable Logic Block (CLB) function generators for Spartan-3E. The 12- and 16-input NAND functions are available only with non-inverting inputs. To invert some or all inputs, use external inverters.

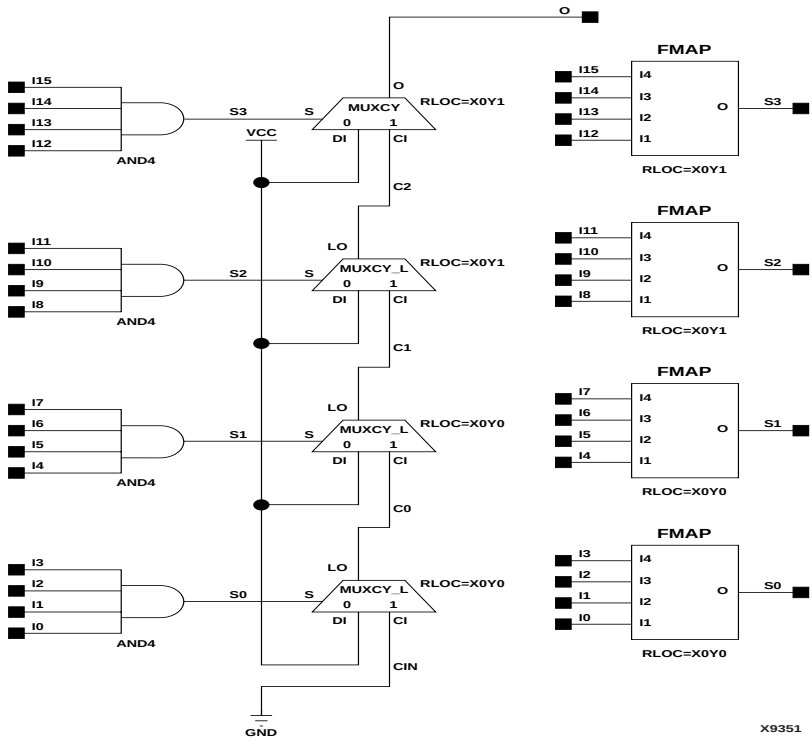
See "NAND2-9" for more information on NAND functions.



NAND12 Implementation of Spartan-3E



NAND12 Implementation Spartan-3E



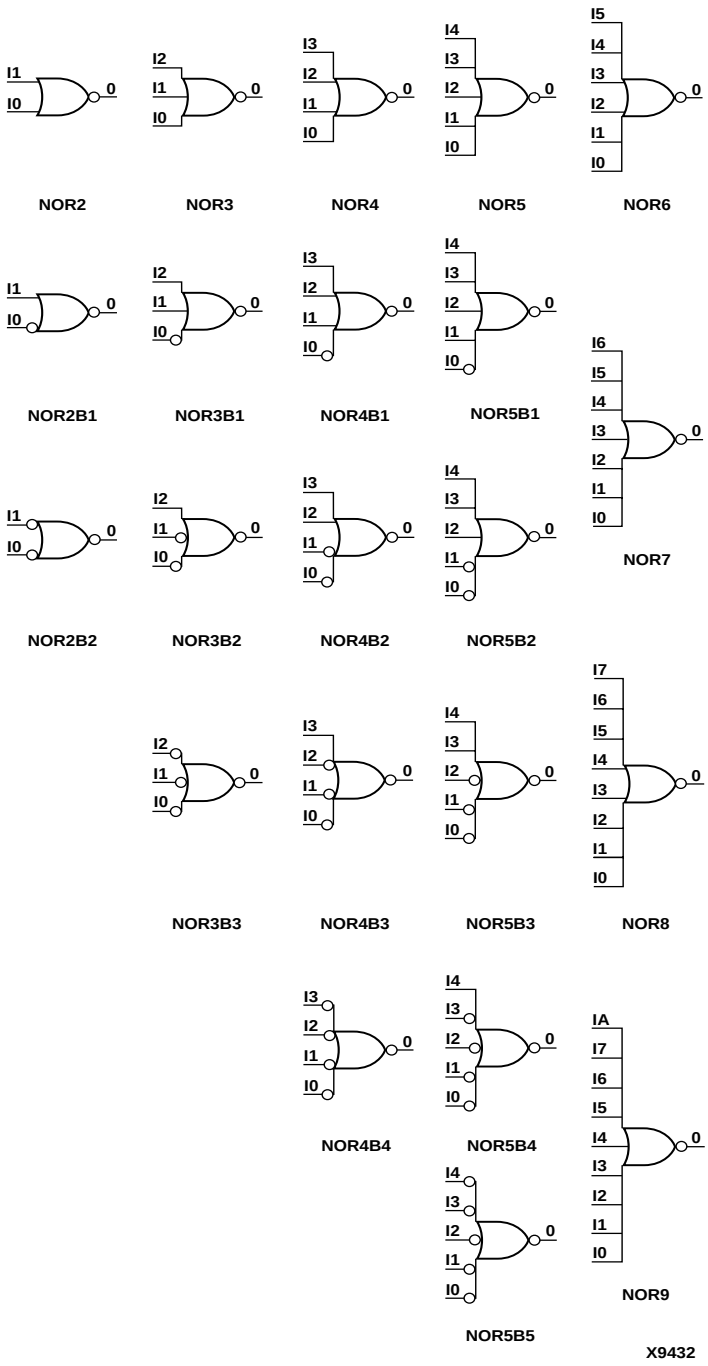
NAND16 Implementation for Spartan-3E

Usage

NAND12 and NAND16 are macros that are inferred. See "NAND2-9" for more information about inferring NAND gates.

NOR2-9

Primitive and Macros: 2- to 9-Input NOR Gates with Inverted and Non-Inverted Inputs

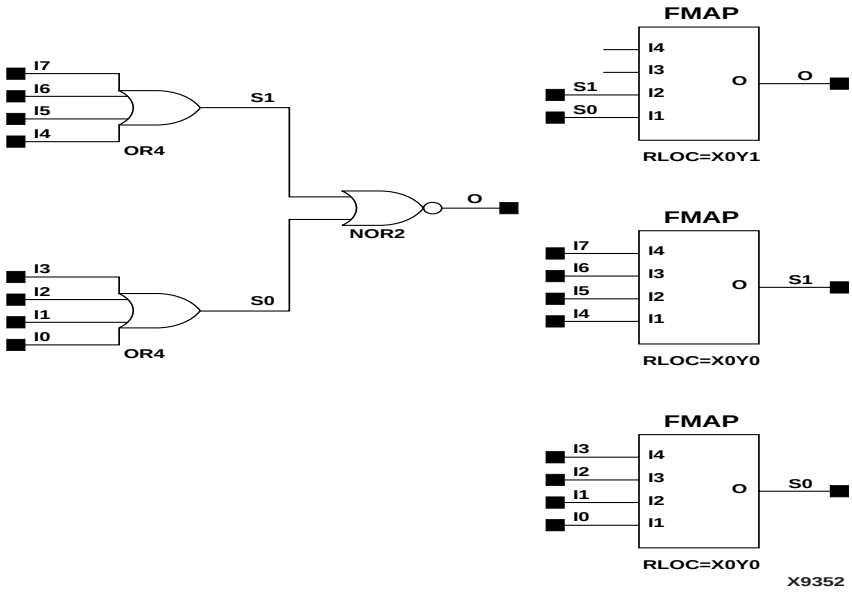


NOR Gate Representations

NOR gates of up to five inputs are available in any combination of inverting and non-inverting inputs. NOR gates of six to nine inputs are available with only non-

inverting inputs. To invert some or all inputs, use external inverters. Since each input uses a CLB resource, replace gates with unused inputs with gates having the necessary number of inputs.

See "NOR12, 16" for information on additional NOR functions.



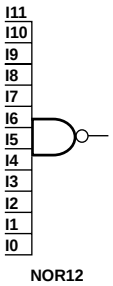
NOR8 Implementation for Spartan-3E

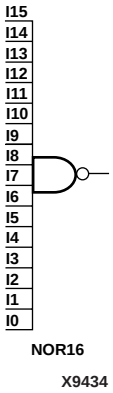
Usage

NOR2 through NOR5 are primitives that can be inferred or instantiated. NOR6 through NOR9 are macros which can be inferred.

NOR12, 16

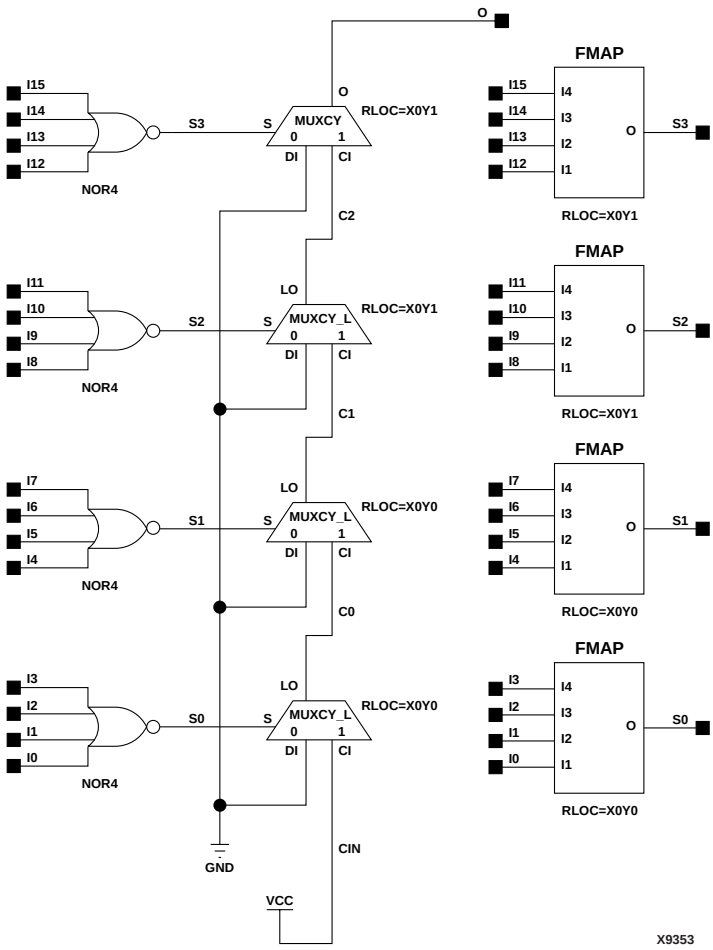
Macros: 12- and 16-Input NOR Gates with Non-Inverted Inputs





The 12- and 16-input NOR functions are available only with non-inverting inputs. To invert some or all inputs, use external inverters.

See "NOR2-9" for more information on NOR functions.



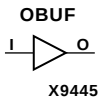
NOR16 Implementation for Spartan-3E

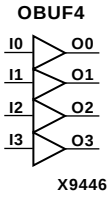
Usage

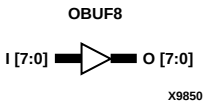
NOR12 and NOR16 are macros that can be inferred. See "NOR2-9" for more information about inferring NOR gates.

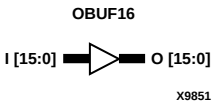
OBUF, 4, 8, 16

Primitive and Macros: Single- and Multiple-Output Buffers



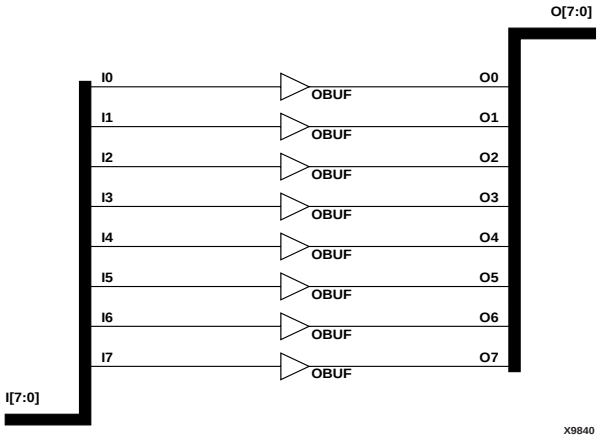






OBUF, OBUF4, OBUF8, and OBUF16 are single and multiple output buffers. An OBUF isolates the internal circuit and provides drive current for signals leaving a chip. OBUFs exist in input/output blocks (IOB). The output (O) of an OBUF is connected to an OPAD or an IOPAD.

The interface standard used by OBUF, 4, 8, and 16 is LVTTL. Also, Spartan-3E OBUF, 4, 8, and 16 have selectable drive and slew rates using the DRIVE and SLOW or FAST constraints. The defaults are DRIVE=12 mA and SLOW slew.



OBUF8 Implementation Spartan-3E

Usage

OBUFs are typically inferred for all top level input ports, but they can also be instantiated if necessary.

Available Attributes

DRIVE

IOSTANDARD

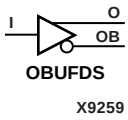
SLEW

For More Information

Consult the Spartan-3E Data Sheets.

OBUFDS

Primitive: Differential Signaling Output Buffer with Selectable I/O Interface



OBUFDS is a single output buffer that supports low-voltage, differential signaling (1.8v CMOS). OBUFDS isolates the internal circuit and provides drive current for signals leaving the chip. Its output is represented as two distinct ports (O and OB), one deemed the "master" and the other the "slave." The master and the slave are opposite phases of the same logical signal (for example, MYNET and MYNETB).

Inputs	Outputs	
I	O	OB
0	0	1
1	1	0

Usage

This design element should be instantiated rather than inferred.

Available Attributes

DRIVE

IOSTANDARD

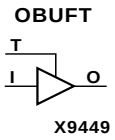
SLEW

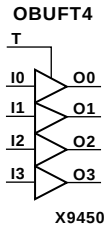
For More Information

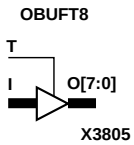
Consult the Spartan-3E Data Sheets.

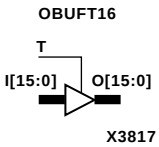
OBUFT, 4, 8, 16

Primitive and Macros: Single and Multiple 3-State Output Buffers with Active-Low Output Enable





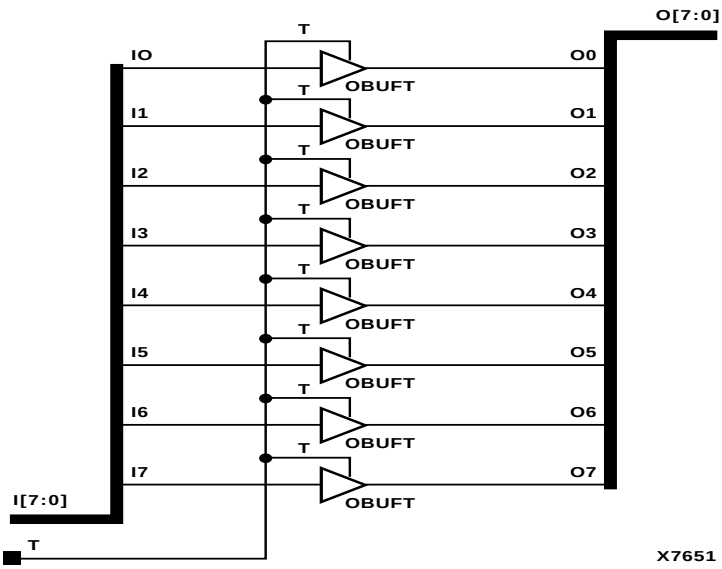




OBUFT, OBUFT4, OBUFT8, and OBUFT16 are single and multiple 3-state output buffers with inputs I, I3 – I0, I7 – I0, I15 – I0, outputs O, O3 – O0, O7 – O0, O15 – O0, and active-Low output enables (T). When (T) is Low, data on the inputs of the buffers is transferred to the corresponding outputs. When (T) is High, the output is high impedance (off or Z state). OBUFTs isolate the internal circuit and provide extra drive current for signals leaving a chip. An OBUFT output is connected to an OPAD or an IOPAD.

OBUFT, 4, 8, and 16 use the LVTTL standard. Also, Spartan-3E OBUFT, 4, 8, and 16 have selectable drive and slew rates using the DRIVE and SLOW or FAST constraints. The defaults are DRIVE=12 mA and SLOW slew.

Inputs		Outputs
T	I	O
1	X	Z
0	I	F



OBUFT8 Implementation for Spartan-3E

Usage

These design elements are instantiated rather than inferred.

Available Attributes

DRIVE

IOSTANDARD

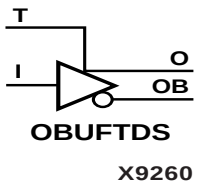
SLEW

For More Information

Consult the Spartan-3E Data Sheets.

OBUFTDS

Primitive: 3-State Differential Signaling Output Buffer with Active Low Output Enable and Selectable I/O Interface



OBUFTDS is a single 3-state, differential signaling output buffer with active Low enable and a Select I/O interface.

When T is Low, data on the input of the buffer is transferred to the output (O) and inverted output (OB). When T is High, both outputs are high impedance (off or Z state).

Inputs		Outputs	
I	T	O	OB
X	1	Z	Z
0	0	0	1
1	0	1	0

Usage

This design element is available for instantiation only.

Available Attributes

DRIVE

IOSTANDARD

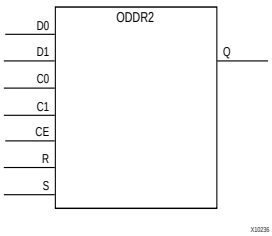
SLEW

For More Information

Consult the Spartan-3E Data Sheets.

ODDR2

Primitive: Double Data Rate Output D Flip-Flop with Optional Data Alignment, Clock Enable and Programmable Synchronous or Asynchronous Set/Reset



The ODDR2 is an output double data rate (DDR) register useful in producing double data rate signals exiting the FPGA. The ODDR2 requires two clocks to be connected to the component, C0 and C1, so that data is provided at the positive edge of both C0 and C1 clocks. The ODDR2 features an active high clock enable port, CE, which may be used to suspend the operation of the registers and both set and reset ports that may be configured to be synchronous or asynchronous to the respective clocks. The ODDR2 has an optional alignment feature, which allows data to be captured by a single clock yet clocked out by two clocks.

Usage

The ODDR2 currently must be instantiated in order to be incorporated into the design. In order to change the default behavior of the ODDR2, attributes may be modified via the generic map (VHDL) or named parameter value assignment (Verilog) as a part of the instantiated component. The ODDR2 may be either connected directly to a top-level output port in the design where an appropriate output buffer can be inferred or to an instantiated OBUF, IOBUF, OBUFDS, OBUFTDS or IOBUFDS. All inputs and outputs of this component should either be connected or properly tied off.

Available Attributes

DDR_ALIGNMENT – Specifies how the data will be captured on the D0 and D1 ports. When set to "NONE", the data on the D0 port will be aligned with the positive edge of the C0 clock and the data on the D1 port will be aligned with the positive edge of the C1 clock. When set to "C0", the data on both D0 and D1 ports are aligned to the positive edge of the C0 clock and when set to "C1", the data on the D0 and D1 ports are aligned to the positive edge of the C1 clock. The output data Q is always presented on the positive edge of both clocks.

INIT – Specifies the initial value upon power-up or the assertion of GSR for the Q port. This attribute may be set to 1 or 0.

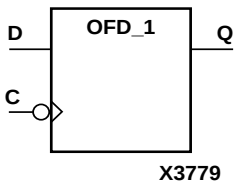
SRTYPE – When set to "SYNC", the reset, R, and set, S, ports are synchronous to the associated clock inputs. When set to "ASYNC", the set and reset ports are asynchronous to the clock.

For More Information

Consult the Spartan-3E Data Sheets.

OFD_1

Macro: Output D Flip-Flop with Inverted Clock



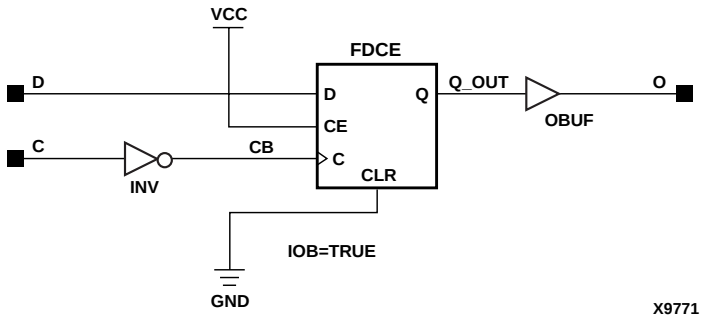
OFD_1 is located in an input/output block (IOB). The output (Q) of the D flip-flop is connected to an OPAD or an IOPAD. The data on the (D) input is loaded into the flip-flop during the High-to-Low clock (C) transition and appears on the (Q) output.

The flip-flop is asynchronously cleared, output Low, when power is applied, or when global reset is active.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs		Outputs
D	C	Q
D	↓	D



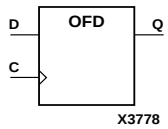
OFD_1 Implementation for Spartan-3E

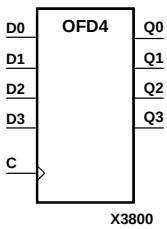
Usage

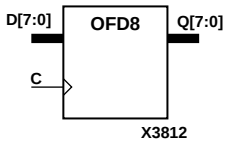
This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an OFD_1, you would infer an FD_1 and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr o to pack all output registers into the IOBs.

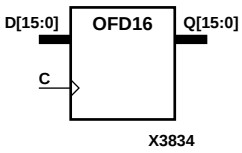
OFD, 4, 8, 16

Macro: Single- and Multiple-Output D Flip-Flops









OFD, OFD4, OFD8, and OFD16 are single and multiple output D flip-flops.

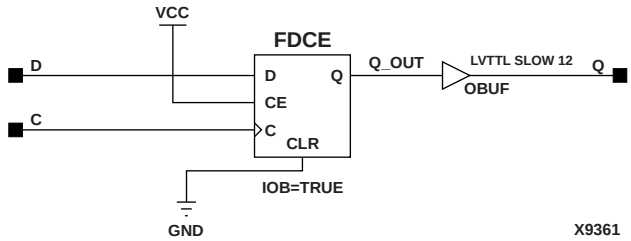
The outputs (for example, Q3 – Q0) are connected to OPADs or IOPADs. The data on the (D) inputs is loaded into the flip-flops during the Low-to-High clock (C) transition and appears on the (Q) outputs.

The flip-flops are asynchronously cleared with Low outputs when power is applied, or when global reset is active.

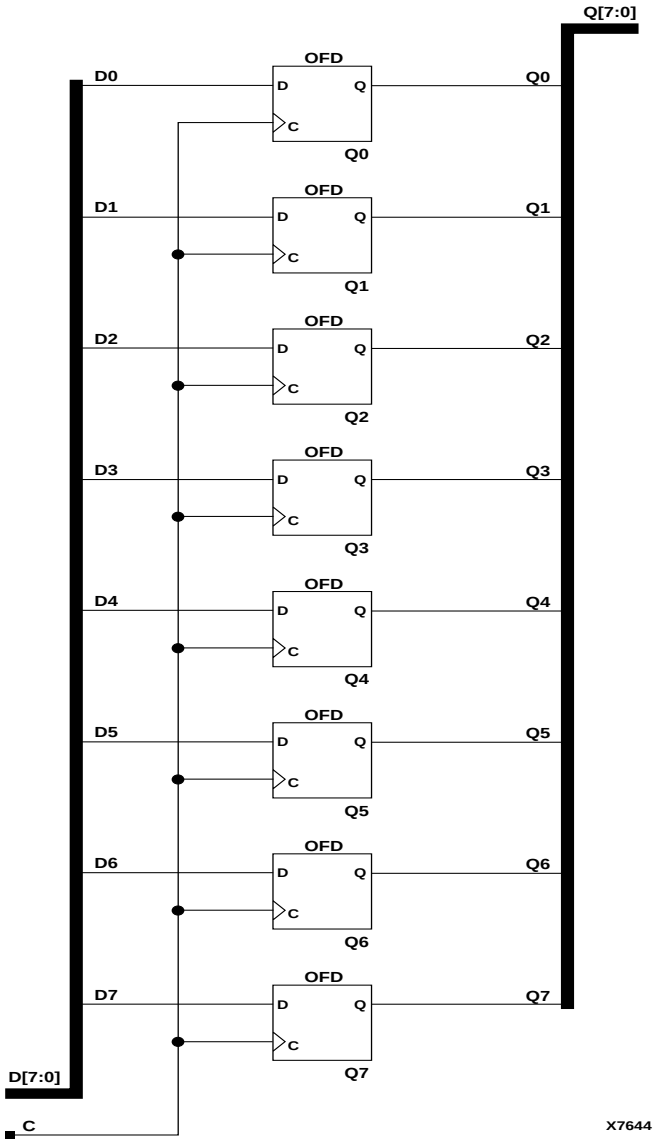
For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs		Outputs
D	C	Q
D	↑	D



OFD Implementation for Spartan-3E



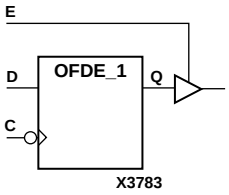
OFD8 Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an OFD, you would infer an FD and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr o to pack all output registers into the IOBs.

OFDE_1

Macro: D Flip-Flop with Active-High Enable Output Buffer and Inverted Clock



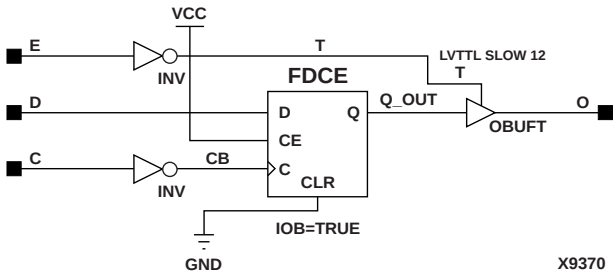
OFDE_1 and its output buffer are located in an input/output block (IOB). The data output of the flip-flop (Q) is connected to the input of an output buffer or OBUFE. The output of the OBUFE is connected to an OPAD or an IOPAD. The data on the data input (D) is loaded into the flip-flop on the High-to-Low clock (C) transition. When the active-High enable input (E) is High, the data on the flip-flop output (Q) appears on the (O) output. When (E) is Low, the output is high impedance (Z state or Off).

The flip-flop is asynchronously cleared with Low output when power is applied, or when global reset is active.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
E	D	C	O
0	X	X	Z
1	D	↓	D



OFDE_1 Implementation for Spartan-3E

Usage

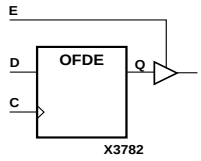
This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an OFDE_1, you would infer an FDE_1 and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr o to pack all output registers into the IOBs.

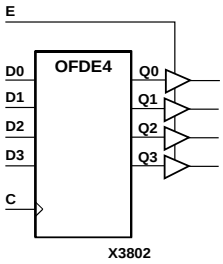
For More Information

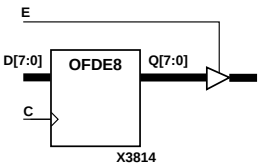
Consult the Spartan-3E Data Sheets.

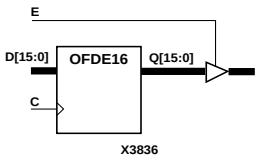
OFDE, 4, 8, 16

Macro: D Flip-Flops with Active-High Enable Output Buffers









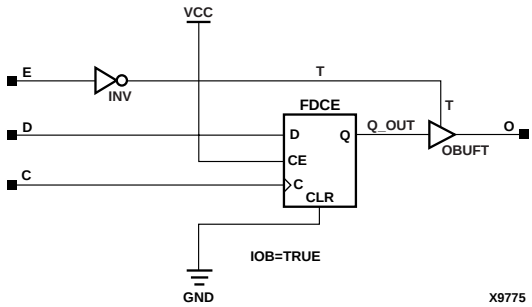
OFDE, OFDE4, OFDE8, and OFDE16 are single or multiple D flip-flops whose outputs are enabled by 3-state buffers. The flip-flop data outputs (Q) are connected to the inputs of output buffers (OBUFE). The OBUFE outputs (O) are connected to OPADs or IOPADs. The data on the data inputs (D) is loaded into the flip-flops during the Low-to-High clock (C) transition. When the active-High enable inputs (E) are High, the data on the flip-flop outputs (Q) appears on the (O) outputs. When (E) is Low, outputs are high impedance (Z state or Off).

The flip-flops are asynchronously cleared with Low outputs when power is applied, or when global reset is active.

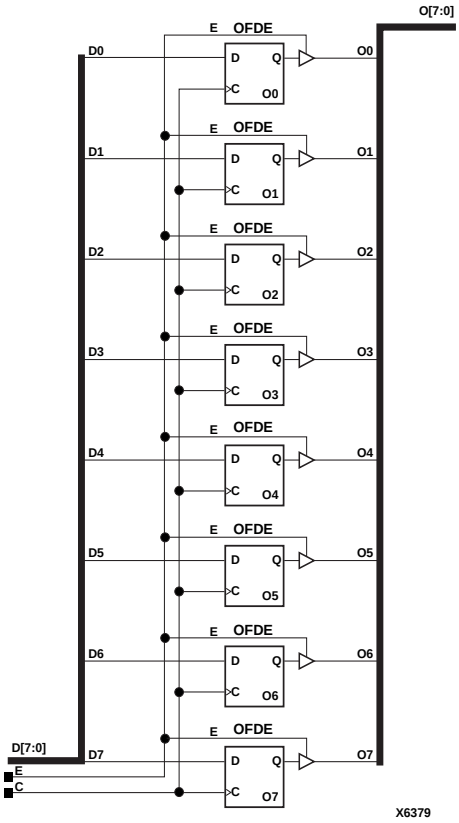
For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
E	D	C	O
0	X	X	Z
1	Dn	↑	Dn



OFDE Implementation for Spartan-3E



OFDE8 Implementation for Spartan-3E

Usage

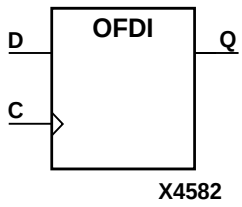
This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an OFDE, you would infer an FDE and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr o to pack all output registers into the IOBs.

For More Information

Consult the Spartan-3E Data Sheets.

OFDI

Macro: Output D Flip-Flop (Asynchronous Preset)



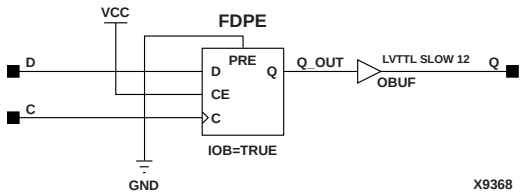
OFDI is contained in an input/output block (IOB). The output (Q) of the (D) flip-flop is connected to an OPAD or an IOPAD. The data on the (D) input is loaded into the flip-flop during the Low-to-High clock (C) transition and appears at the output (Q).

The flip-flop is asynchronously preset, output High, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs		Outputs
D	**C**	**Q**
D	↑	D



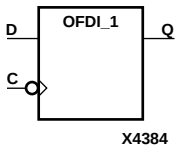
OFDI Implementation Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an OFDI, you would infer an FDP and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr o to pack all output registers into the IOBs.

OFDI_1

Macro: Output D Flip-Flop with Inverted Clock (Asynchronous Preset)



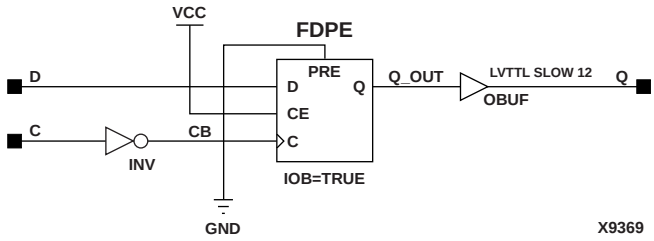
OFDI_1 exists in an input/output block (IOB). The (D) flip-flop output (Q) is connected to an OPAD or an IOPAD. The data on the (D) input is loaded into the flip-flop during the High-to-Low clock (C) transition and appears on the (Q) output.

The flip-flop is asynchronously preset, output High, when power is applied.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs		Outputs
D	**C**	**Q**
D	↓	D



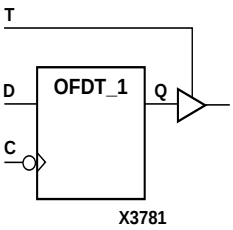
OFDI_1 Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an OFDI_1, you would infer an FDP_1 and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr o to pack all output registers into the IOBs.

OFDT_1

Macro: D Flip-Flop with Active-Low 3-State Output Buffer and Inverted Clock



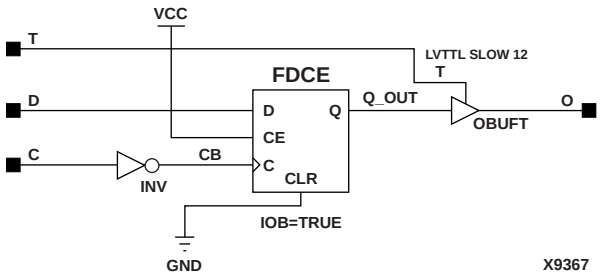
OFDT_1 and its output buffer are located in an input/output block (IOB). The flip-flop data output (Q) is connected to the input of an output buffer (OBUFT). The OBUFT output is connected to an OPAD or an IOPAD. The data on the data input (D) is loaded into the flip-flop on the High-to-Low clock (C) transition. When the active-Low enable input (T) is Low, the data on the flip-flop output (Q) appears on the (O) output. When (T) is High, the output is high impedance (Off).

The flip-flop is asynchronously cleared with Low output when power is applied, or when global reset is active.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol

Inputs			Outputs
T	D	C	O
1	X	X	Z
0	D	↓	D



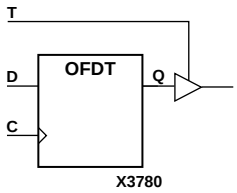
OFDT_1 Implementation for Spartan-3E

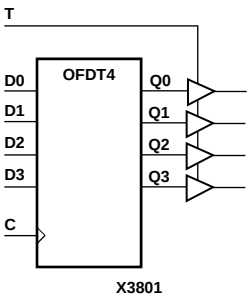
Usage

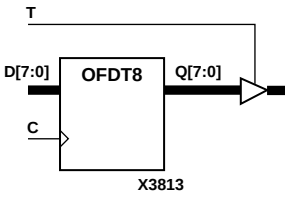
This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an OFDT_1, you would infer an FDCE_1 and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr o to pack all output registers into the IOBs.

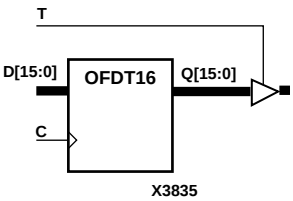
OFDT, 4, 8, 16

Macro: Single and Multiple D Flip-Flops with Active-Low 3-State Output Enable Buffers









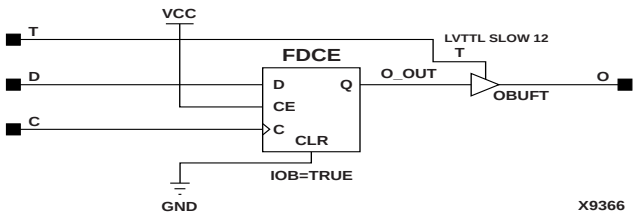
OFDT, OFDT4, OFDT8, and OFDT16 are single or multiple D flip-flops whose outputs are enabled by a 3-state buffers. The data outputs (Q) of the flip-flops are connected to the inputs of output buffers (OBUFT). The outputs of the OBUFTs (O) are connected to OPADs or IOPADs. The data on the data inputs (D) is loaded into the flip-flops during the Low-to-High clock (C) transition. When the active-Low enable inputs (T) are Low, the data on the flip-flop outputs (Q) appears on the (O) outputs. When (T) is High, outputs are high impedance (Off).

The flip-flops are asynchronously cleared with Low outputs, when power is applied, or when global reset is active.

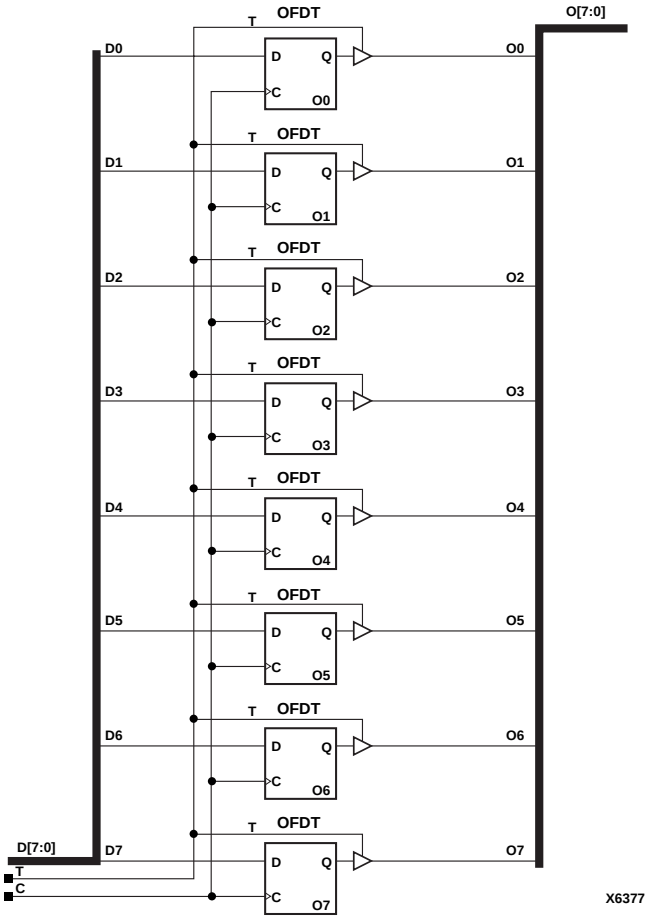
For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
T	D	C	O
1	X	X	Z
0	D	↑	D



OFDT Implementation for Spartan-3E



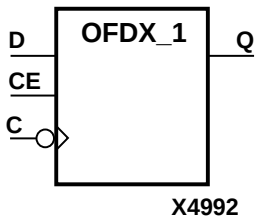
OFDT8 Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an OFDT, you would infer an FDCE and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr o to pack all output registers into the IOBs.

OFDX_1

Macro: Output D Flip-Flop with Inverted Clock and Clock Enable



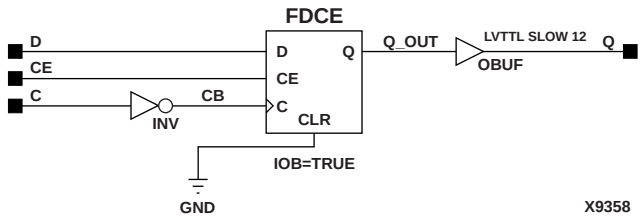
OFDX_1 is located in an input/output block (IOB). The output (Q) of the (D) flip-flop is connected to an OPAD or an IOPAD. The data on the (D) input is loaded into the flip-flop during the High-to-Low clock (C) transition and appears on the (Q) output. When the (CE) pin is Low, the output (Q) does not change.

The flip-flop is asynchronously cleared with Low output when power is applied, or when global reset is active.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
CE	D	C	Q
1	D	↓	D
0	X	X	No Change



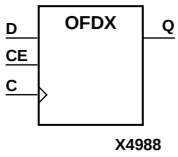
OFDX_1 Implementation for Spartan-3E

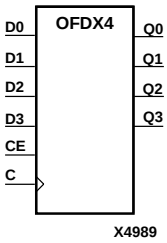
Usage

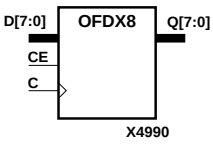
This component is inside of the IOB. It cannot be directly instantiated. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an OFDX_1, you would infer an FDCE_1 and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr o to pack all output registers into the IOBs.

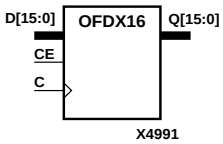
OFDX, 4, 8, 16

Macro: Single- and Multiple-Output D Flip-Flops with Clock Enable









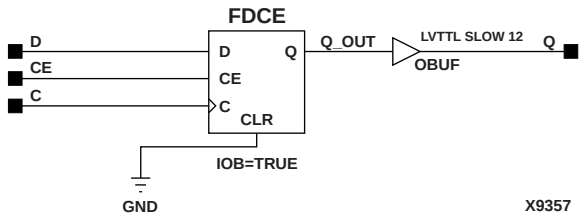
OFDX, OFDX4, OFDX8, and OFDX16 are single and multiple output D flip-flops. The (Q) outputs are connected to OPADs or IOPADs. The data on the (D) inputs is loaded into the flip-flops during the Low-to-High clock (C) transition and appears on the (Q) outputs. When (CE) is Low, flip-flop outputs do not change.

The flip-flops are asynchronously cleared with Low outputs, when power is applied, or when global reset is active.

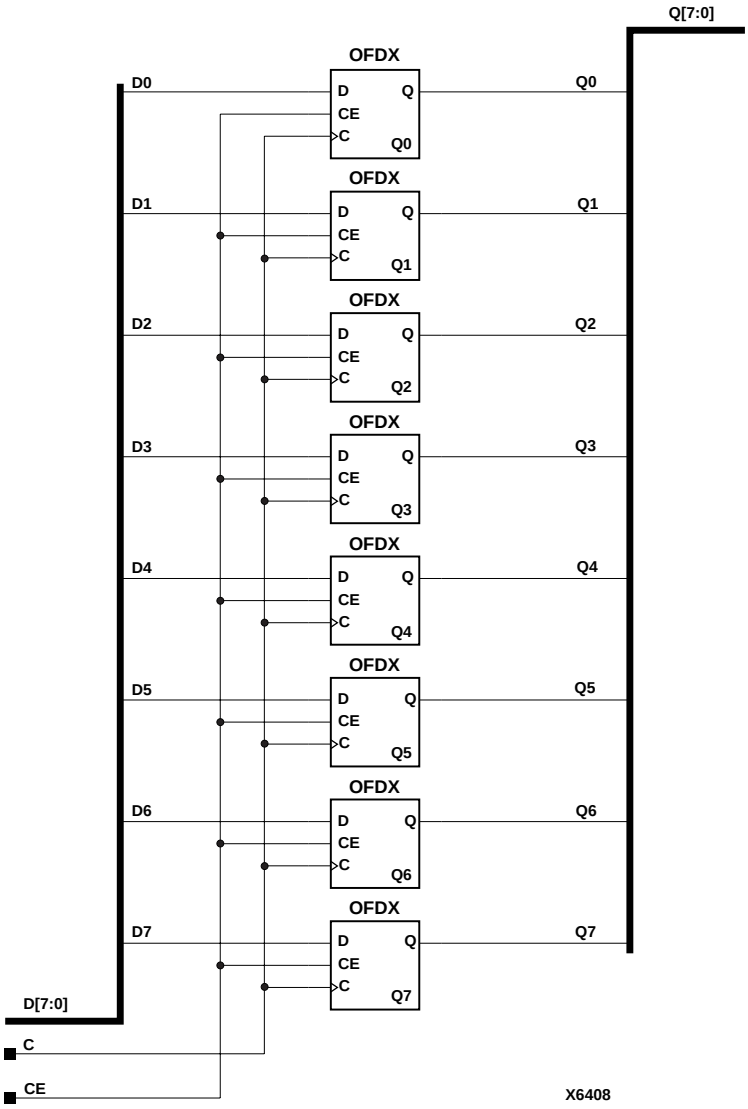
For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
CE	D	C	Q
1	Dn	↑	Dn
0	X	X	No Change



OFDX Implementation for Spartan-3E



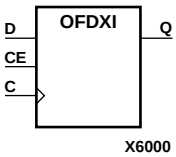
OFDX8 Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred nor instantiated. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an OFDX, you would infer an FDCE and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr o to pack all output registers into the IOBs.

OFDXI

Macro: Output D Flip-Flop with Clock Enable (Asynchronous Preset)



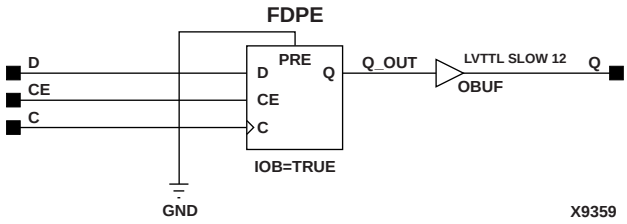
OFDXI is contained in an input/output block (IOB). The output (Q) of the D flip-flop is connected to an OPAD or an IOPAD. The data on the (D) input is loaded into the flip-flop during the Low-to-High clock (C) transition and appears at the output (Q). When(CE) is Low, the output does not change.

The flip-flop is asynchronously preset with High output when power is applied, or when global reset is active.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
CE	D	C	Q
1	D	↑	D
0	X	X	No Change



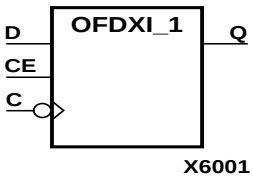
OFDXI Implementation for Spartan-3e

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an OFDXI, you would infer an FDPE and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr o to pack all output registers into the IOBs.

OFDXI_1

Macro: Output D Flip-Flop with Inverted Clock and Clock Enable (Asynchronous Preset)



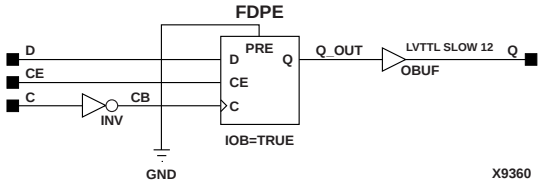
OFDXI_1 is located in an input/output block (IOB). The D flip-flop output (Q) is connected to an OPAD or an IOPAD. The data on the D input is loaded into the flip-flop during the High-to-Low clock (C) transition and appears on the Q output. When CE is Low, the output (Q) does not change.

The flip-flop is asynchronously preset with High output when power is applied, or when global reset is active.

For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs			Outputs
CE	D	C	Q
1	D	↓	D
0	X	X	No Change



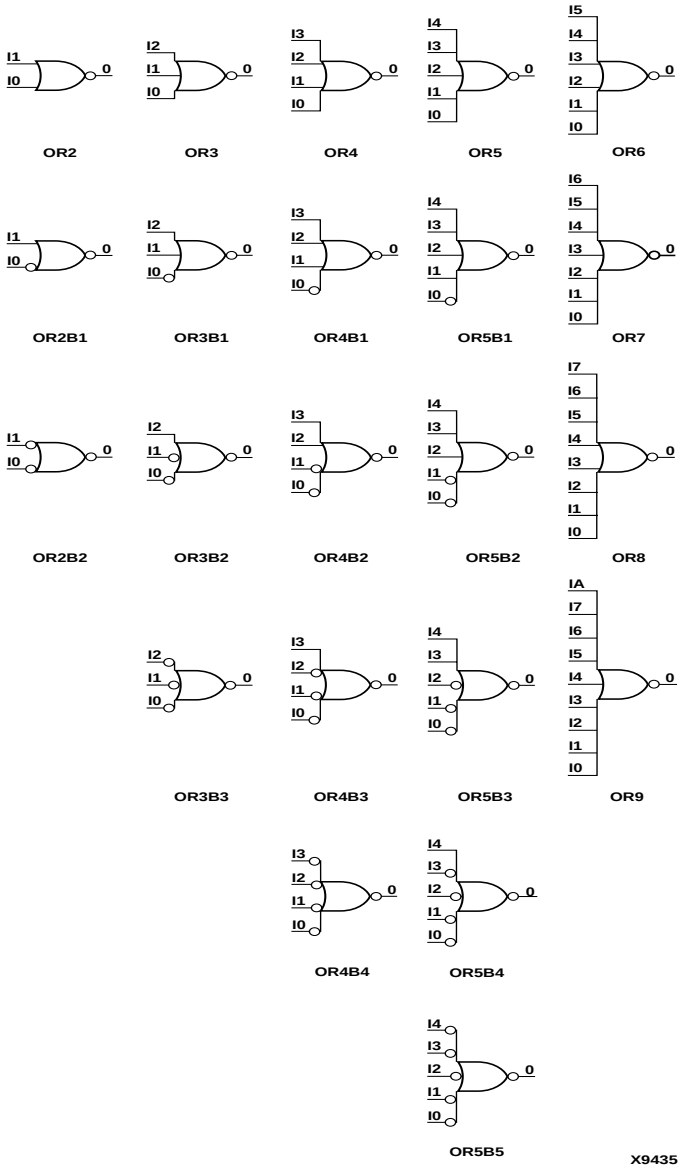
OFDXI_1 Implementation for Spartan-3E

Usage

This component is inside of the IOB. It cannot be directly inferred. The most common design practice is to infer a regular component and put an IOB=TRUE attribute on the component in the UCF file or in the code. For instance, to get an OFDXI_1, you would infer an FDPE_1 and put the IOB = TRUE attribute on the component. Or, you could use the map option –pr o to pack all output registers into the IOBs.

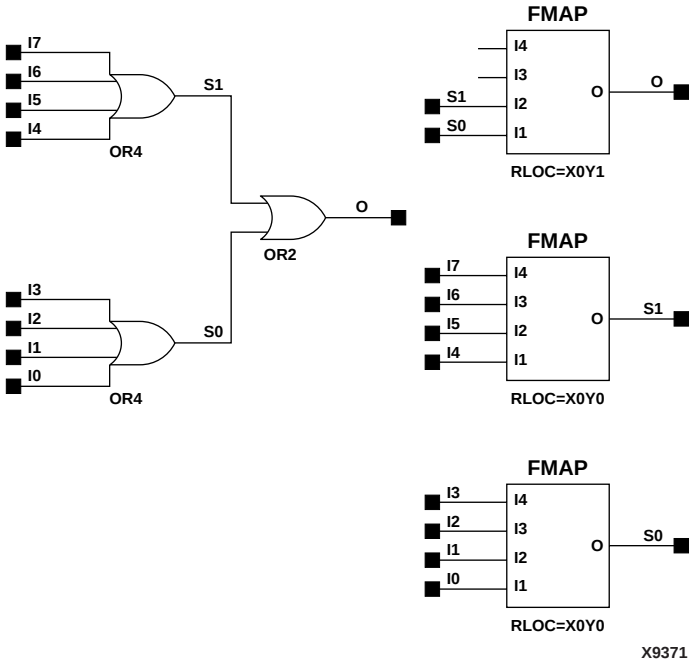
OR2-9

Primitive: 2- to 9-Input OR Gates with Inverted and Non-Inverted Inputs



OR Gate Representations

The OR function is performed in the Configurable Logic Block (CLB) function generators for Spartan-3E. OR functions of up to five inputs are available in any combination of inverting and non-inverting inputs. OR functions of six to nine inputs are available with only non-inverting inputs. To invert some or all inputs, use external inverters. Since each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.



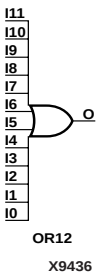
OR8 Implementation for Spartan-3E

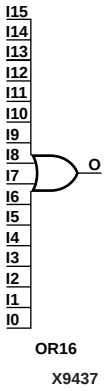
Usage

OR2 through OR5 are primitives that can be inferred or instantiated. OR6 through OR9 are macros which can be inferred.

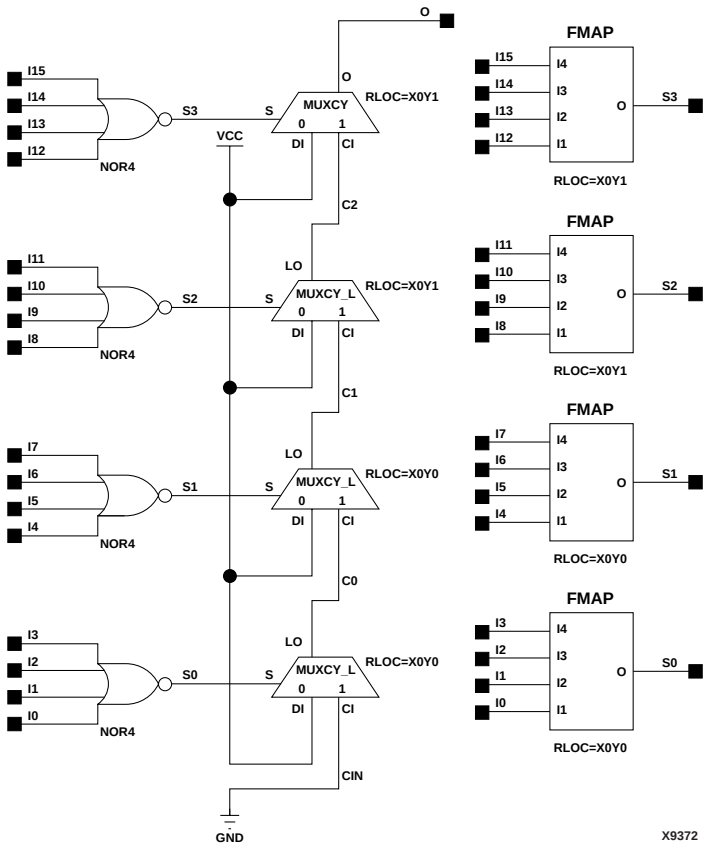
OR12, 16

Primitive: 12- and 16-Input OR Gates with Non-Inverted Inputs





See "OR2-9" for information on OR functions.



OR16 Implementation for Spartan-3E

Usage

OR12 and OR16 are macros that are inferred.

PULLDOWN

Primitive: Resistor to GND for Input Pads

X3860

PULLDOWN resistor elements are connected to input, output, or bidirectional pads to guarantee a logic Low level for nodes that might float.

Usage

The PULLDOWN design element is instantiated rather than inferred.

For More Information

Consult the Spartan-3E Data Sheets.

PULLUP

Primitive: Resistor to VCC for Input PADs, Open-Drain, and 3-State Outputs

X3861

The pull-up elements establish a High logic level for open-drain elements and macros (DECODE, WAND, WORAND) or 3-state nodes (TBUF) when all the drivers are off.

The buffer outputs are connected together as a wired-AND to form the output (O). When all the inputs are High, the output is off. To establish an output High level, a PULLUP resistor(s) is tied to output (O). One PULLUP resistor uses the least power, two pull-up resistors achieve the fastest Low-to-High speed.

To indicate two PULLUP resistors, append a DOUBLE parameter to the pull-up symbol attached to the output (O) node.

Usage

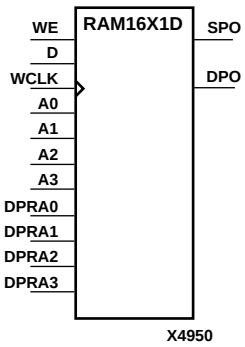
The PULLUP design element is instantiated rather than inferred.

For More Information

Consult the Spartan-3E Data Sheets.

RAM16X1D

Primitive: 16-Deep by 1-Wide Static Dual Port Synchronous RAM



RAM16X1D is a 16-word by 1-bit static dual port random access memory with synchronous write capability. The device has two separate address ports: the read address (DPRA3 – DPRA0) and the write address (A3 – A0). These two address ports are completely asynchronous. The read address controls the location of the data driven out of the output pin (DPO), and the write address controls the destination of a valid write transaction.

When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is High, any positive transition on (WCLK) loads the data on the data input (D) into the word selected by the 4-bit write address. For predictable performance, write address and data inputs must be stable before a Low-to-High (WCLK) transition. This RAM block assumes an active-High (WCLK). (WCLK) can be active-High or active-Low. Any inverter placed on the (WCLK) input net is absorbed into the block.

Mode selection is shown in the following truth table.

Inputs			Outputs	
WE (mode)	**WCLK**	**D**	**SPO**	**DPO**
0 (read)	X	X	data_a	data_d
1 (read)	0	X	data_a	data_d
1 (read)	1	X	data_a	data_d
1 (write)	↑	D	D	data_d
1 (read)	↓	X	data_a	data_d

data_a = word addressed by bits A3-A0

data_d = word addressed by bits DPRA3-DPRA0

The SPO output reflects the data in the memory cell addressed by A3 – A0. The DPO output reflects the data in the memory cell addressed by DPRA3 – DPRA0.

Note: The write process is not affected by the address on the read address port.

Specifying Initial Contents of a RAM

You can use the INIT attribute to specify an initial value directly on the symbol if the RAM is 1 bit wide and 16, 32, 64, or 128 bits deep. The value must be a hexadecimal number, for example, INIT=ABAC. If the INIT attribute is not specified, the RAM is initialized with zero.

Usage

This design element can be inferred or instantiated.

Available Attributes.

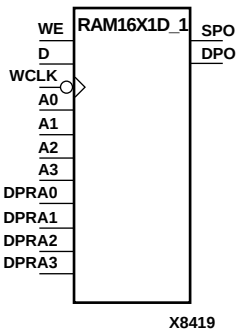
Attribute	Type	Allowed Values	Default
INIT	16-Bit Binary	16 bits of 1's or 0's (Hex values allowed)	All zeroes

For More Information

Consult the Spartan-3E Data Sheets.

RAM16X1D_1

Primitive: 16-Deep by 1-Wide Static Dual Port Synchronous RAM with Negative-Edge Clock



RAM16X1D_1 is a 16-word by 1-bit static dual port random access memory with synchronous write capability and negative-edge clock. The device has two separate address ports: the read address (DPRA3 – DPRA0) and the write address (A3 – A0). These two address ports are completely asynchronous. The read address controls the location of the data driven out of the output pin (DPO), and the write address controls the destination of a valid write transaction.

When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When (WE) is High, any negative transition on (WCLK) loads the data on the data input (D) into the word selected by the 4-bit write address. For predictable performance, write address and data inputs must be stable before a High-to-Low WCLK transition. This RAM block assumes an active-High (WCLK). (WCLK) can be active-High or active-Low. Any inverter placed on the (WCLK) input net is absorbed into the block.

You can initialize RAM16X1D_1 during configuration using the INIT attribute. See "Specifying Initial Contents of a RAM" in the RAM16X1D section.

Mode selection is shown in the following truth table.

Inputs			Outputs	
WE (mode)	WCLK	D	SPO	DPO
0 (read)	X	X	data_a	data_d
1 (read)	0	X	data_a	data_d
1 (read)	1	X	data_a	data_d
1 (write)	↓	D	D	data_d
1 (read)	↑	X	data_a	data_d

data_a = word addressed by bits A3-A0

data_d = word addressed by bits DPRA3-DPRA0

The SPO output reflects the data in the memory cell addressed by A3 – A0. The DPO output reflects the data in the memory cell addressed by DPRA3 – DPRA0.

Note: The write process is not affected by the address on the read address port.

Usage

This design element can be inferred or instantiated.

Available Attributes.

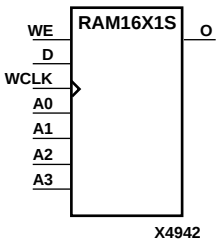
Attribute	Type	Allowed Values	Default
INIT	64 bits	64 zeroes	All zeroes

For More Information

Consult the Spartan-3E Data Sheets.

RAM16X1S

Primitive: 16-Deep by 1-Wide Static Synchronous RAM



RAM16X1S is a 16-word by 1-bit static random access memory with synchronous write capability. When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When (WE) is High, any positive transition on (WCLK) loads the data on the data input (D) into the word selected by the 4-bit address (A3 – A0). For predictable performance, address and data inputs must be stable before a Low-to-High (WCLK) transition. This RAM block assumes an active-High (WCLK). However, (WCLK) can be active-High or active-Low. Any inverter placed on the (WCLK) input net is absorbed into the block.

The signal output on the data output pin (O) is the data that is stored in the RAM at the location defined by the values on the address pins.

You can initialize RAM16X1S during configuration using the INIT attribute.

Mode selection is shown in the following truth table.

Inputs			Outputs
WE(mode)	**WCLK**	**D**	**O**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↑	D	D
1 (read)	↓	X	Data

Data = word addressed by bits A3 – A0

Usage

This design element can be inferred or instantiated.

Available Attributes.

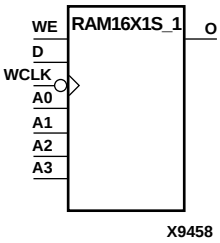
Attribute	Type	Allowed Values	Default
INIT	16-Bit Binary	16 bits of 1's or 0's (Hex values allowed)	All zeroes

For More Information

Consult the Spartan-3E Data Sheets.

RAM16X1S_1

Primitive: 16-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock



RAM16X1S_1 is a 16-word by 1-bit static random access memory with synchronous write capability and negative-edge clock. When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When (WE) is High, any negative transition on (WCLK) loads the data on the data input (D) into the word selected by the 4-bit address (A3 – A0). For predictable performance, address and data inputs must be stable before a High-to-Low WCLK transition. This RAM block assumes an active-Low (WCLK). However, (WCLK) can be active-High or active-Low. Any inverter placed on the (WCLK) input net is absorbed into the block.

The signal output on the data output pin (O) is the data that is stored in the RAM at the location defined by the values on the address pins.

You can initialize RAM16X1S_1 during configuration using the INIT attribute.

Mode selection is shown in the following truth table.

Inputs			Outputs
WE(mode)	**WCLK**	**D**	**O**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↓	D	D
1 (read)	↑	X	Data

Data = word addressed by bits A3 – A0

Usage

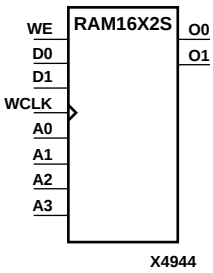
This design element can be inferred or instantiated.

For More Information

Consult the Spartan-3E Data Sheets.

RAM16X2S

Primitive: 16-Deep by 2-Wide Static Synchronous RAM



RAM16X2S is a 16-word by 2-bit static random access memory with synchronous write capability. When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is High, any positive transition on WCLK loads the data on the data input (D1 – D0) into the word selected by the 4-bit address (A3 – A0). For predictable performance, address and data inputs must be stable before a Low-to-High WCLK transition. This RAM block assumes an active-High WCLK. However, WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block.

The signal output on the data output pins (O1 – O0) is the data that is stored in the RAM at the location defined by the values on the address pins.

Except for Spartan-3E, the initial contents of RAM16X2S cannot be specified directly. See "Specifying Initial Contents of a RAM" in the RAM16X1D section.

For Spartan-3E, you can use the INIT_00 and INIT_01 properties to specify the initial contents of RAM16X2S.

Mode selection is shown in the following truth table.

Inputs			Outputs
WE (mode)	**WCLK**	**D1-D0**	**O1-O0**
0 (read)	X	X	Data
1(read)	0	X	Data
1(read)	1	X	Data
1(write)	↑	D1-D0	D1-D0
1 (read)	↓	X	Data

Data = word addressed by bits A3 – A0

Specifying Initial Contents of a Spartan-3E Wide RAM

You can use the INIT_xx properties to specify the initial contents of a Spartan-3E wide RAM. INIT_00 initializes the RAM cells corresponding to the O0 output, INIT_01 initializes the cells corresponding to the O1 output, etc. For example, a RAM16X2S instance is initialized by INIT_00 and INIT_01 containing 4 hex characters each. A RAM16X8S instance is initialized by eight properties INIT_00 through INIT_07 containing 4 hex characters each. A RAM64x2S instance is completely initialized by two properties INIT_00 and INIT_01 containing 16 hex characters each. See the INIT_xx section of the *Constraints Guide* for more information on the INIT_xx attribute.

Usage

This design element can be inferred or instantiated.

Available Attributes.

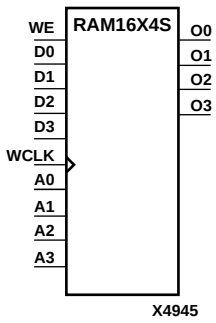
Attribute	Type	Allowed Values	Default
INIT_00 To INIT_01	16-Bit	16 zeries	0 or 1

For More Information

Consult the Spartan-3E Data Sheets.

RAM16X4S

Primitive: 16-Deep by 4-Wide Static Synchronous RAM



RAM16X4S is a 16-word by 4-bit static random access memory with synchronous write capability. When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is High, any positive transition on WCLK loads the data on the data input (D3 – D0) into the word selected by the 4-bit address (A3 – A0). For predictable performance, address and data inputs must be stable before a Low-to-High WCLK transition. This RAM block assumes an active-High WCLK. However, WCLK can be active-High or active-Low. Any inverter placed on the WCLK input net is absorbed into the block.

The signal output on the data output pins (O3 – O0) is the data that is stored in the RAM at the location defined by the values on the address pins.

For Spartan-3E, you can use INIT_00 through INIT_03 to specify the initial contents of RAM16X4S as described in the "Specifying Initial Contents of a Spartan-3E Wide RAM"section in the RAM16X2S section.

Mode selection is shown in the following truth table.

Inputs			Outputs
WE (mode)	**WCLK**	**D3 – D0**	**O3 – O0**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↑	D3-D0	D3-D0
1 (read)	↓	X	Data

Data = word addressed by bits A3 – A0

Usage

This design element can be inferred or instantiated.

Available Attributes.

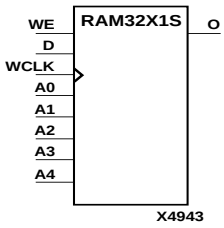
Attribute	Type	Allowed Values	Default
INIT_00 To INIT_03	16-Bit Binary	16 zeroes	0, 1, 2, or 3

For More Information

Consult the Spartan-3E Data Sheets.

RAM32X1S

Primitive: 32-Deep by 1-Wide Static Synchronous RAM



RAM32X1S is a 32-word by 1-bit static random access memory with synchronous write capability. When the write enable is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When WE is High, any positive transition on (WCLK) loads the data on the data input (D) into the word selected by the 5-bit address (A4 – A0). For predictable performance, address and data inputs must be stable before a Low-to-High (WCLK) transition. This RAM block assumes an active-High (WCLK). However, (WCLK) can be active-High or active-Low. Any inverter placed on the (WCLK) input net is absorbed into the block.

The signal output on the data output pin (O) is the data that is stored in the RAM at the location defined by the values on the address pins.

You can initialize RAM32X1S during configuration using the INIT attribute.

Mode selection is shown in the following truth table.

Inputs			Outputs
WE (mode)	**WCLK**	**D**	**O**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↑	D	D
1 (read)	↓	X	Data

Data = word addressed by bits A4 – A0

Usage

This design element can be inferred or instantiated.

Available Attributes

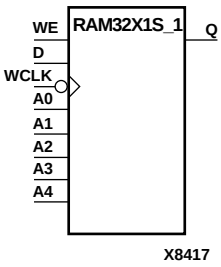
Attribute	Type	Allowed Values	Default
INIT	32-Bit Binary	32 bits of 1's and 0's (Hex values allowed)	All zeroes

For More Information

Consult the Spartan-3E Data Sheets.

RAM32X1S_1

Primitive: 32-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock



RAM32X1S_1 is a 32-word by 1-bit static random access memory with synchronous write capability. When the write enable is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When (WE) is High, any negative transition on (WCLK) loads the data on the data input (D) into the word selected by the 5-bit address (A4 – A0). For predictable performance, address and data inputs must be stable before a High-to-Low (WCLK) transition. This RAM block assumes an active-Low (WCLK). However, (WCLK) can be active-High or active-Low. Any inverter placed on the (WCLK) input net is absorbed into the block.

The signal output on the data output pin (O) is the data that is stored in the RAM at the location defined by the values on the address pins.

You can initialize RAM32X1S_1 during configuration using the INIT attribute.

Mode selection is shown in the following truth table.

Inputs			Outputs
WE (mode)	**WCLK**	**D**	**O**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↓	D	D
1 (read)	↑	X	Data

Data = word addressed by bits A4 – A0

Usage

This design element can be inferred or instantiated.

Available Attributes

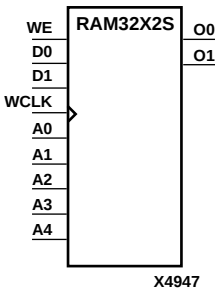
Attribute	Type	Allowed Values	Default
INIT	32-Bit Binary	32 zeroes	All zeroes

For More Information

Consult the Spartan-3E Data Sheets.

RAM32X2S

Primitive: 32-Deep by 2-Wide Static Synchronous RAM



RAM32X2S is a 32-word by 2-bit static random access memory with synchronous write capability. When the write enable (WE) is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When (WE) is High, any positive transition on (WCLK) loads the data on the data input (D1 – D0) into the word selected by the 5-bit address (A4 – A0). For predictable performance, address and data inputs must be stable before a Low-to-High (WCLK) transition. This RAM block assumes an active-High (WCLK) . However, (WCLK) can be active-High or active-Low. Any inverter placed on the (WCLK) input net is absorbed into the block.

The signal output on the data output pins (O1 – O0) is the data that is stored in the RAM at the location defined by the values on the address pins.

For Spartan-3E, you can use the INIT_00 and INIT_01 properties to specify the initial contents of RAM32X2S as described in "Specifying Initial Contents of a Spartan-3E Wide RAM" in the RAM16X2S section.

Mode selection is shown in the following truth table.

Inputs			Outputs
WE (mode)	**WCLK**	**D0-D1**	**O0-O1**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↑	D1-D0	D1-D0
1 (read)	↓	X	Data

Data = word addressed by bits A4 – A0

Usage

This design element can be inferred or instantiated.

Available Attributes

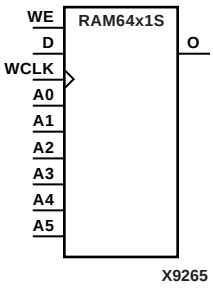
Attribute	Type	Allowed Values	Default
INIT_00 To INIT_01	32-Bit Binary	32 zeroes	0 or 1

For More Information

Consult the Spartan-3E Data Sheets.

RAM64X1S

Primitive: 64-Deep by 1-Wide Static Synchronous RAM



RAM64X1S is a 64-word by 1-bit static random access memory with synchronous write capability. When the write enable is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When (WE) is High, any positive transition on (WCLK) loads the data on the data input (D) into the word selected by the 6-bit address (A5 – A0). For predictable performance, address and data inputs must be stable before a Low-to-High (WCLK) transition. This RAM block assumes an active-High (WCLK) . However, (WCLK) can be active-High or active-Low. Any inverter placed on the (WCLK) input net is absorbed into the block.

The signal output on the data output pin (O) is the data that is stored in the RAM at the location defined by the values on the address pins.

You can initialize RAM64X1S during configuration using the INIT attribute.

Mode selection is shown in the following truth table.

Inputs			Outputs
WE (mode)	**WCLK**	**D**	**O**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↑	D	D
1 (read)	↓	X	Data

Data = word addressed by bits A5 – A0

Usage

This design element can be inferred or instantiated.

Available Attributes.

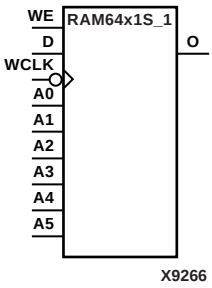
Attribute	Type	Allowed Values	Default
INIT	64-Bit Binary	64 bits of 1's and 0's	All zeroes

For More Information

Consult the Spartan-3E Data Sheets.

RAM64X1S_1

Primitive: 64-Deep by 1-Wide Static Synchronous RAM with Negative-Edge Clock



RAM64X1S_1 is a 64-word by 1-bit static random access memory with synchronous write capability. When the write enable is Low, transitions on the write clock (WCLK) are ignored and data stored in the RAM is not affected. When (WE) is High, any negative transition on (WCLK) loads the data on the data input (D) into the word selected by the 6-bit address (A5 – A0). For predictable performance, address and data inputs must be stable before a High-to-Low (WCLK) transition. This RAM block assumes an active-Low (WCLK) . However, (WCLK) can be active-High or active-Low. Any inverter placed on the (WCLK) input net is absorbed into the block.

The signal output on the data output pin (O) is the data that is stored in the RAM at the location defined by the values on the address pins.

You can initialize RAM32X1S_1 during configuration using the INIT attribute.

Mode selection is shown in the following truth table.

Inputs			Outputs
WE (mode)	**WCLK**	**D**	**O**
0 (read)	X	X	Data
1 (read)	0	X	Data
1 (read)	1	X	Data
1 (write)	↓	D	D
1 (read)	↑	X	Data

Data = word addressed by bits A5 – A0

Usage

This design element can be inferred or instantiated.

Available Attributes.

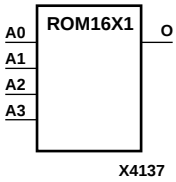
Attribute	Type	Allowed Values	Default
INIT	64-Bit Binary	64bits of 1's and 0's	All zeroes

For More Information

Consult the Spartan-3E Data Sheets.

ROM16X1

Primitive: 16-Deep by 1-Wide ROM



ROM16X1 is a 16-word by 1-bit read-only memory. The data output (O) reflects the word selected by the 4-bit address (A3 – A0). The ROM is initialized to a known value during configuration with the INIT=value parameter. The value consists of four hexadecimal digits that are written into the ROM from the most-significant digit A=FH to the least-significant digit A=0H. For example, the INIT=10A7 parameter produces the data stream:

```
0001 0000 1010 0111
```

An error occurs if the INIT=*value* is not specified.

Usage

The ROM16X1 design element should be instantiated rather than inferred.

Available Attributes

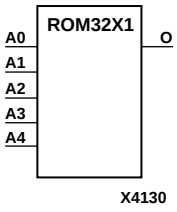
Attribute	Type	Default
INIT	16-Bit Binary	All zeroes

For More Information

Consult the Spartan-3E Data Sheets.

ROM32X1

Primitive: 32-Deep by 1-Wide ROM



ROM32X1 is a 32-word by 1-bit read-only memory. The data output (O) reflects the word selected by the 5-bit address (A4 – A0). The ROM is initialized to a known value during configuration with the INIT=value parameter. The value consists of eight hexadecimal digits that are written into the ROM from the most-significant digit A=1FH to the least-significant digit A=00H. For example, the INIT=10A78F39 parameter produces the data stream:

```
0001 0000 1010 0111 1000 1111 0011 1001
```

An error occurs if the INIT=*value* is not specified.

Usage

The ROM32X1 design element should be instantiated rather than inferred.

Available Attributes.

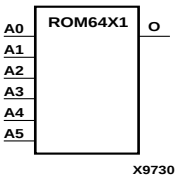
Attribute	Type	Default
INIT	32-Bit Binary	All zeroes

For More Information

Consult the Spartan-3E Data Sheets.

ROM64X1

Primitive: 64-Deep by 1-Wide ROM



ROM64X1 is a 64-word by 1-bit read-only memory. The data output (O) reflects the word selected by the 6-bit address (A5 – A0). The ROM is initialized to a known value during configuration with the INIT=value parameter. The value consists of 16 hexadecimal digits that are written into the ROM from the most-significant digit A=FH to the least-significant digit A=0H.

An error occurs if the INIT=*value* is not specified.

Usage

The ROM64X1 design element should be instantiated rather than inferred.

Available Attributes.

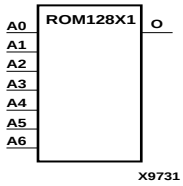
Attribute	Type	Default
INIT	64-Bit Binary	All zeroes

For More Information

Consult the Spartan-3E Data Sheets.

ROM128X1

Primitive: 128-Deep by 1-Wide ROM



ROM128X1 is a 128-word by 1-bit read-only memory. The data output (O) reflects the word selected by the 7-bit address (A6 – A0). The ROM is initialized to a known value during configuration with the INIT=value parameter. The value consists of 32 hexadecimal digits that are written into the ROM from the most-significant digit A=FH to the least-significant digit A=0H.

An error occurs if the INIT=*value* is not specified.

Usage

The ROM128X1 design element should be instantiated rather than inferred.

Available Attributes

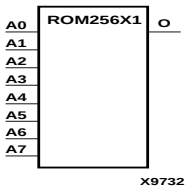
Attribute	Type	Default
INIT	128-Bit Binary	All zeroes

For More Information

Consult the Spartan-3E Data Sheets.

ROM256X1

Primitive: 256-Deep by 1-Wide ROM



ROM256X1 is a 256-word by 1-bit read-only memory. The data output (O) reflects the word selected by the 8-bit address (A7– A0). The ROM is initialized to a known value during configuration with the INIT=value parameter. The value consists of 64 hexadecimal digits that are written into the ROM from the most-significant digit A=FH to the least-significant digit A=0H.

An error occurs if the INIT=*value* is not specified. See the appropriate CAE tool interface user guide for details.

Usage

The ROM256X1 design element should be instantiated rather than inferred.

Available Attributes

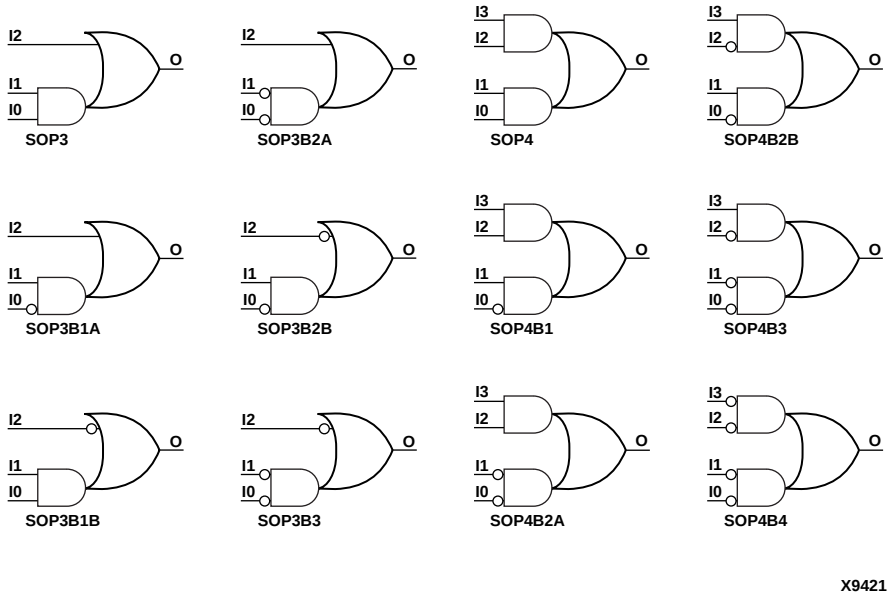
Attribute	Type	Default
INIT	256-Bit Binary	All zeroes

For More Information

Consult the Spartan-3E Data Sheets.

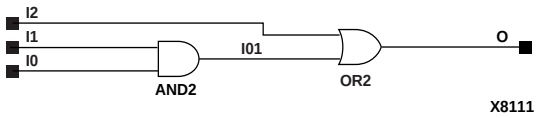
SOP3-4

Macro: Sum of Products

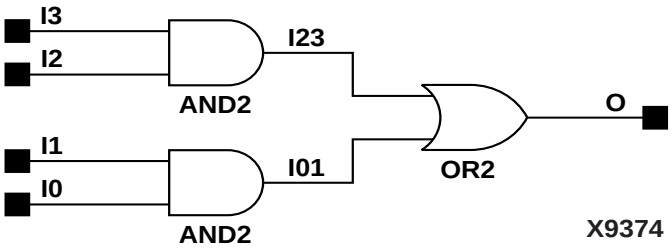


SOP Gate Representations

Sum Of Products (SOP) macros provide common logic functions by OR gating the outputs of two AND functions or the output of one AND function with one direct input. Variations of inverting and non-inverting inputs are available.



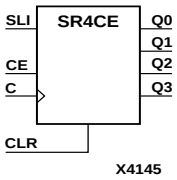
SOP3 Implementation for Spartan-3E



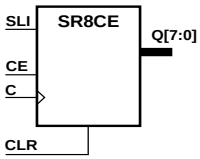
SOP4 Implementation of Spartan-3E

SR4CE, SR8CE, SR16CE

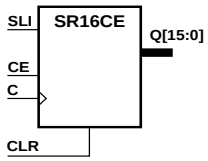
Macro: 4-, 8-, 16-Bit Serial-In Parallel-Out Shift Registers with Clock Enable and Asynchronous Clear



X4145



X4151



X4157

SR4CE, SR8CE, and SR16CE are 4-, 8-, and 16-bit shift registers, respectively, with a shift-left serial input (SLI), parallel outputs (Q), and clock enable (CE) and asynchronous clear (CLR) inputs. The (CLR) input, when High, overrides all other inputs and resets the data outputs (Q) Low. When (CE) is High and (CLR) is Low, the data on the SLI input is loaded into the first bit of the shift register during the Low-to-High clock (C) transition and appears on the (Q0) output. During subsequent Low-to-High clock transitions, when (CE) is High and (CLR) is Low, data is shifted to the next highest bit position as new data is loaded into (Q0) (SLI→Q0, Q0→Q1, Q1→Q2, and so forth). The register ignores clock transitions when (CE) is Low.

Registers can be cascaded by connecting the last (Q) output (Q3 for SR4CE, Q7 for SR8CE, or Q15 for SR16CE) of one stage to the SLI input of the next stage and connecting clock, (CE), and (CLR) in parallel.

The register is asynchronously cleared, outputs Low, when power is applied.

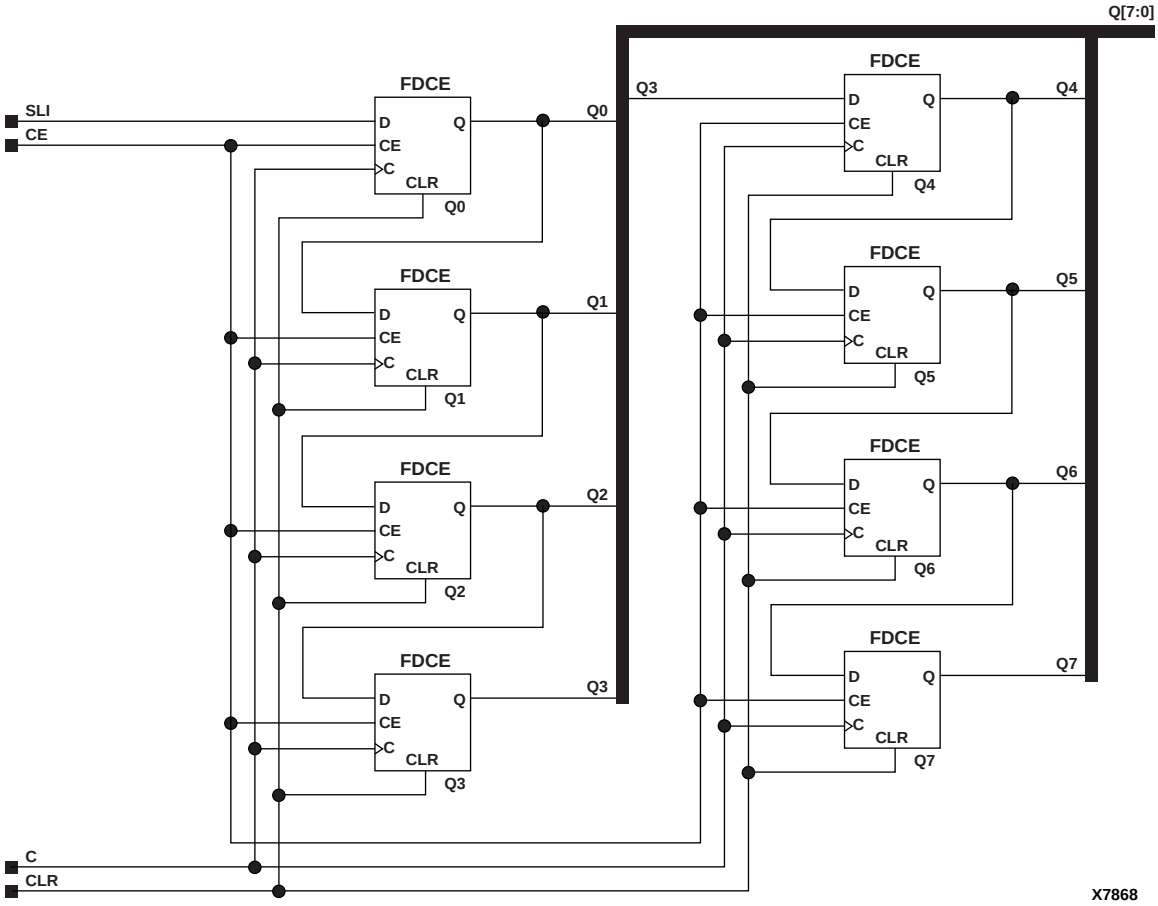
For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs	
CLR	CE	SLI	C	Q0	Qz – Q1
1	X	X	X	0	0
0	0	X	X	No Change	No Change
0	1	SLI	↑	SLI	qn-1

z = 3 for SR4CE; z = 7for SR8CE; z = 15 for SR16CE

qn-1 = state of referenced output one setup time prior to active clock transition



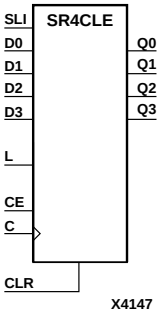
SR8CE Implementation for Spartan-3E

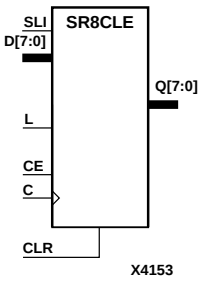
Usage

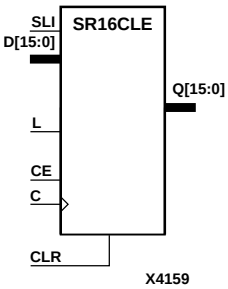
These design elements are inferred rather than instantiated.

SR4CLE, SR8CLE, SR16CLE

Macros: 4-, 8-, 16-Bit Loadable Serial/Parallel-In Parallel-Out Shift Registers with Clock Enable and Asynchronous Clear







SR4CLE, SR8CLE, and SR16CLE are 4-, 8-, and 16-bit shift registers, respectively, with a shift-left serial input (SLI), parallel inputs (D), parallel outputs (Q), and three control inputs: clock enable (CE), load enable (L), and asynchronous clear (CLR).The register ignores clock transitions when (L) and (CE) are Low. The asynchronous (CLR), when High, overrides all other inputs and resets the data outputs (Q) Low. When (L) is High and (CLR) is Low, data on the Dn – D0 inputs is loaded into the corresponding Qn – (Q0) bits of the register.

When (CE) is High and (L) and (CLR) are Low, data on the SLI input is loaded into the first bit of the shift register during the Low-to-High clock (C) transition and appears on the (Q0) output. During subsequent clock transitions, when (CE) is High and (L) and (CLR) are Low, the data is shifted to the next highest bit position as new data is loaded into(Q)0 (SLI→Q0, Q0→Q1, Q1→Q2, and so forth).

Registers can be cascaded by connecting the last (Q) output (Q3 for SR4CLE, Q7 for SR8CLE, or Q15 for SR16CLE) of one stage to the SLI input of the next stage and connecting clock, (CE), (L), and (CLR) inputs in parallel.

The register is asynchronously cleared, outputs Low, when power is applied.

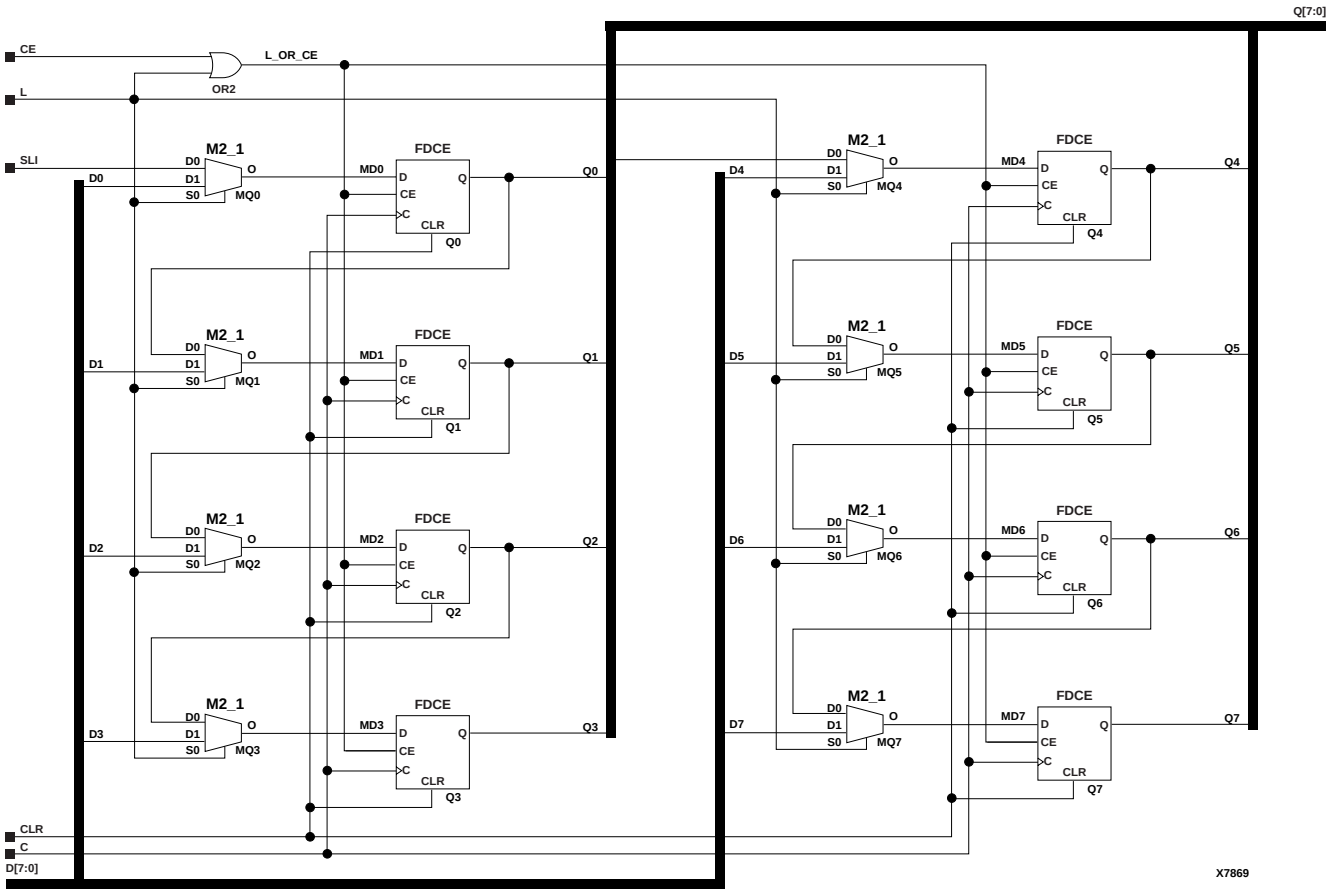
For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs						Outputs	
CLR	L	CE	SLI	Dn – D0	C	Q0	Qz – Q1
1	X	X	X	X	X	0	0
0	1	X	X	Dn – D0	↑	D0	Dn
0	0	1	SLI	X	↑	SLI	qn-1
0	0	0	X	X	X	No Change	No Change

z = 3 for SR4CLE; z = 7 for SR8CLE; z = 15 for SR16CLE

qn-1 = state of referenced output one setup time prior to active clock transition



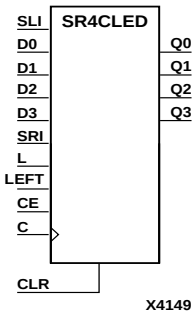
SR8CLE Implementation for Spartan-3E

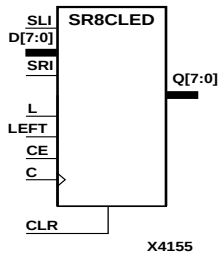
Usage

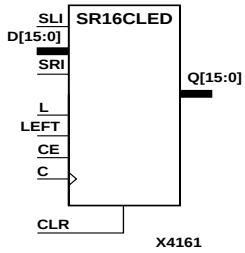
This design element is inferred rather than instantiated.

SR4CLED, SR8CLED, SR16CLED

Macro: 4-, 8-, 16-Bit Shift Registers with Clock Enable and Asynchronous Clear







SR4CLED, SR8CLED, and SR16CLED are 4-, 8-, and 16-bit shift registers, respectively, with shift-left (SLI) and shift-right (SRI) serial inputs, parallel inputs (D), parallel outputs (Q), and four control inputs: clock enable (CE), load enable (L), shift left / right (LEFT), and asynchronous clear (CLR). The register ignores clock transitions when(CE) and (L) are Low. The asynchronous clear, when High, overrides all other inputs and resets the data outputs (Qn) Low.

When (L) is High and (CLR) is Low, the data on the (D) inputs is loaded into the corresponding (Q) bits of the register. When (CE) is High and (L) and (CLR) are Low, data is shifted right or left, depending on the state of the LEFT input. If LEFT is High, data on the SLI is loaded into (Q0) during the Low-to-High clock transition and shifted left (to Q1, Q2, and so forth) during subsequent clock transitions. If LEFT is Low, data on the SRI is loaded into the last (Q) output (Q3 for SR4CLED, Q7 for SR8CLED, or Q15 for SR16CLED) during the Low-to-High clock transition and shifted right (to Q2, Q1,... for SR4CLED; to Q6, Q5,... for SR8CLED; and to Q14, Q13,... for SR16CLED) during subsequent clock transitions. The truth tables for SR4CLED, SR8CLED, and SR16CLED indicate the state of the (Q) outputs under all input conditions for SR4CLED, SR8CLED, and SR16CLED.

The register is asynchronously cleared, outputs Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set / reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

SR4CLED Truth Table

Inputs								Outputs		
CLR	L	CE	LEFT	SLI	SRI	D3 – D0	C	Q0	Q3	Q2 – Q1
1	X	X	X	X	X	X	X	0	0	0
0	1	X	X	X	X	D3– D0	↑	D0	D3	Dn
0	0	0	X	X	X	X	X	No Change	No Change	No Change
0	0	1	1	SLI	X	X	↑	SLI	q2	qn-1
0	0	1	0	X	SRI	X	↑	q1	SRI	qn+1

qn-1 and qn+1 = state of referenced output one setup time prior to active clock transition

SR8CLED Truth Table

Inputs								Outputs		
CLR	L	CE	LEFT	SLI	SRI	D7 – D0	C	Q0	Q7	Q6 – Q1
1	X	X	X	X	X	X	X	0	0	0
0	1	X	X	X	X	D7 – D0	↑	D0	D7	Dn

SR8CLED Truth Table

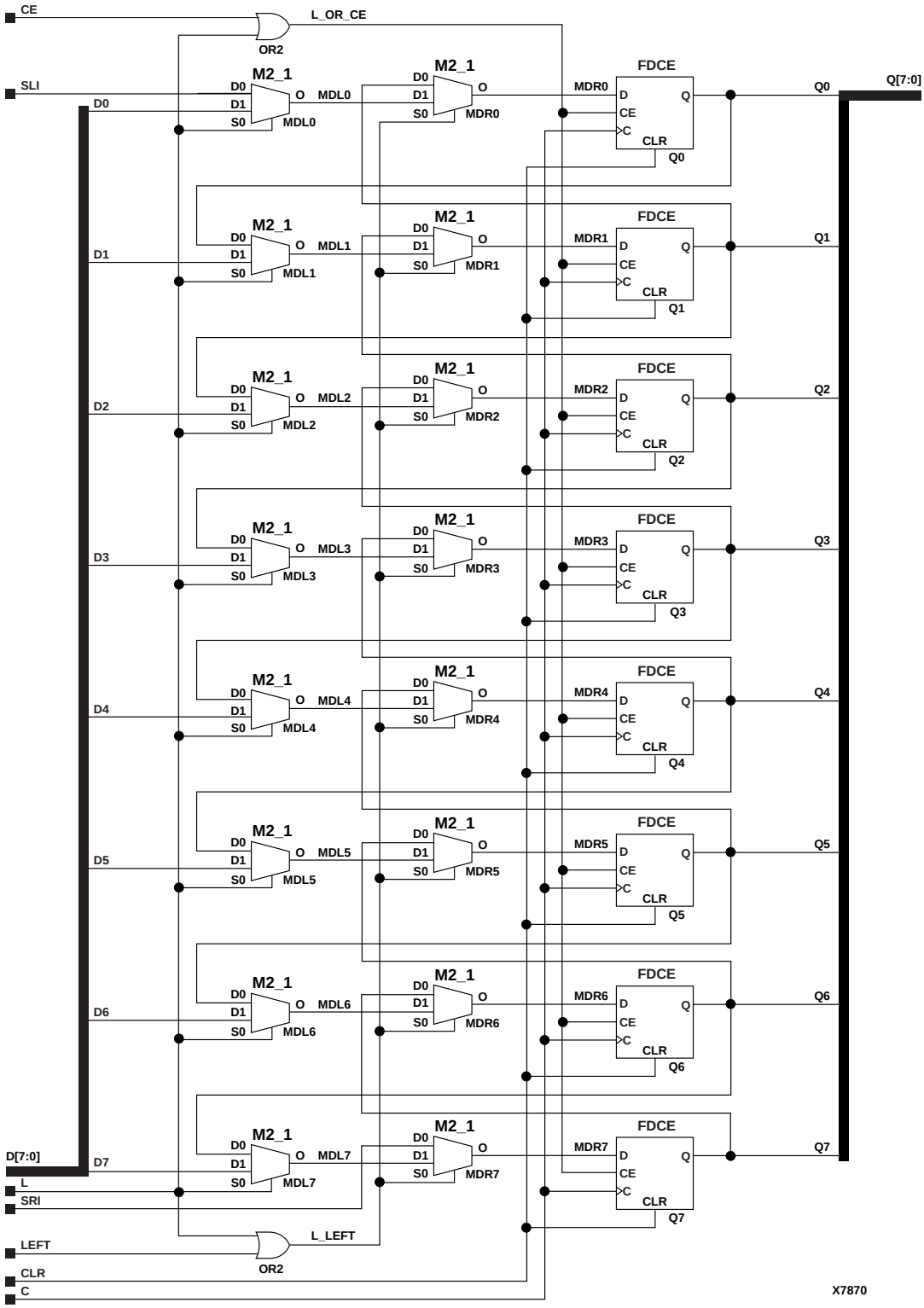
Inputs								Outputs		
CLR	L	CE	LEFT	SLI	SRI	D7 – D0	C	Q0	Q7	Q6 – Q1
0	0	0	X	X	X	X	X	No Change	No Change	No Change
0	0	1	1	SLI	X	X	↑	SLI	q6	qn-1
0	0	1	0	X	SRI	X	↑	q1	SRI	qn+1

qn-1 or qn+1 = state of referenced output one setup time prior to active clock transition

SR16CLED Truth Table

Inputs								Outputs		
CLR	L	CE	LEFT	SLI	SRI	D15 – D0	C	Q0	Q15	Q14 – Q1
1	X	X	X	X	X	X	X	0	0	0
0	1	X	X	X	X	D15 – D0	↑	D0	D15	Dn
0	0	0	X	X	X	X	X	No Change	No Change	No Change
0	0	1	1	SLI	X	X	↑	SLI	q14	qn-1
0	0	1	0	X	SRI	X	↑	q1	SRI	qn+1

qn-1 or qn+1 = state of referenced output one setup time prior to active clock transition



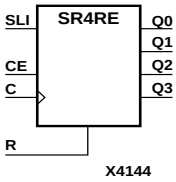
SR8CLED Implementation for Spartan-3E

Usage

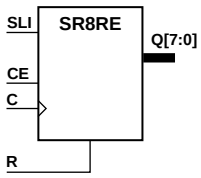
This design element is inferred rather than instantiated.

SR4RE, SR8RE, SR16RE

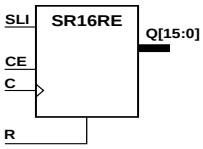
Macros: 4-, 8-, 16-Bit Serial-In Parallel-Out Shift Registers with Clock Enable and Synchronous Reset



X4144



X4150



X4156

SR4RE, SR8RE, and SR16RE are 4-, 8-, and 16-bit shift registers, respectively, with shift-left serial input (SLI), parallel outputs (Qn), clock enable (CE), and synchronous reset (R) inputs. The R input, when High, overrides all other inputs during the Low-to-High clock (C) transition and resets the data outputs (Q) Low.

When (CE) is High and (R) is Low, the data on the (SLI) is loaded into the first bit of the shift register during the Low-to-High clock (C) transition and appears on the (Q0) output. During subsequent Low-to-High clock transitions, when (CE) is High and R is Low, data is shifted to the next highest bit position as new data is loaded into (Q0) (SLI→Q0, Q0→Q1, Q1→Q2, and so forth). The register ignores clock transitions when (CE) is Low.

Registers can be cascaded by connecting the last (Q) output (Q3 for SR4RE, Q7 for SR8RE, or Q15 for SR16RE) of one stage to the SLI input of the next stage and connecting clock, (CE), and (R) in parallel.

The register is asynchronously cleared, outputs Low, when power is applied.

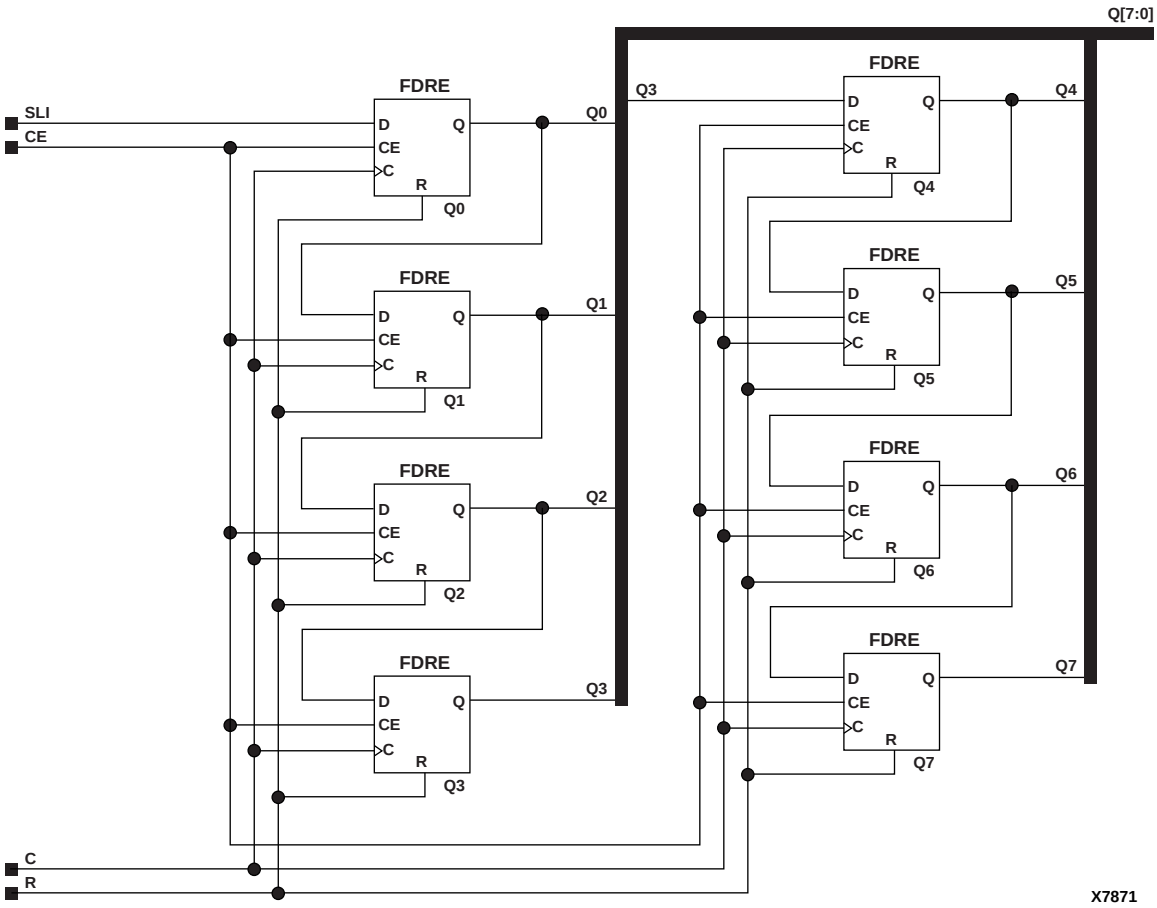
For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs				Outputs	
R	CE	SLI	C	Q0	Qz – Q1
1	X	X	↑	0	0
0	0	X	X	No Change	No Change
0	1	SLI	↑	SLI	qn-1

z = 3 for SR4RE; z = 7 for SR8RE; z = 15 for SR16RE

qn-1 = state of referenced output one setup time prior to active clock transition



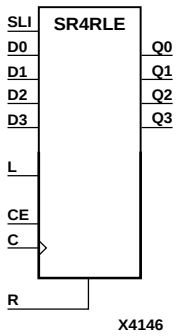
SR8RE Implementation for Spartan-3E

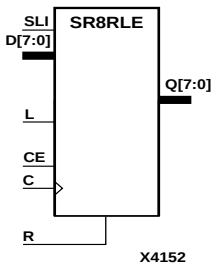
Usage

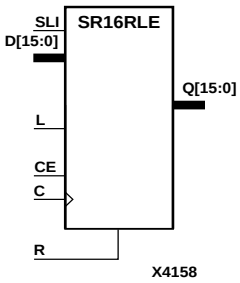
This design element is inferred rather than instantiated.

SR4RLE, SR8RLE, SR16RLE

Macros: 4-, 8-, 16-Bit Loadable Serial/Parallel-In Parallel-Out Shift Registers with Clock Enable and Synchronous Reset







SR4RLE, SR8RLE, and SR16RLE are 4-, 8-, and 16-bit shift registers, respectively, with shift-left serial input (SLI), parallel inputs (D), parallel outputs (Q), and three control inputs: clock enable (CE), load enable (L), and synchronous reset (R). The register ignores clock transitions when (L) and (CE) are Low. The synchronous (R), when High, overrides all other inputs during the Low-to-High clock (C) transition and resets the data outputs (Q) Low. When (L) is High and (R) is Low during the Low-to-High clock transition, data on the (D) inputs is loaded into the corresponding Q bits of the register. When (CE) is High and (L) and (R) are Low, data on the (SLI) input is loaded into the first bit of the shift register during the Low-to-High clock (C) transition and appears on the Q0 output. During subsequent clock transitions, when (CE) is High and (L) and (R) are Low, the data is shifted to the next highest bit position as new data is loaded into Q0 (SLI→Q0, Q0→Q1, Q1→Q2, and so forth).

Registers can be cascaded by connecting the last Q output (Q3 for SR4RLE, Q7 for SR8RLE, or 15 for SR16RLE) of one stage to the SLI input of the next stage and connecting clock, (CE), (L), and (R) inputs in parallel.

The register is asynchronously cleared, outputs Low, when power is applied.

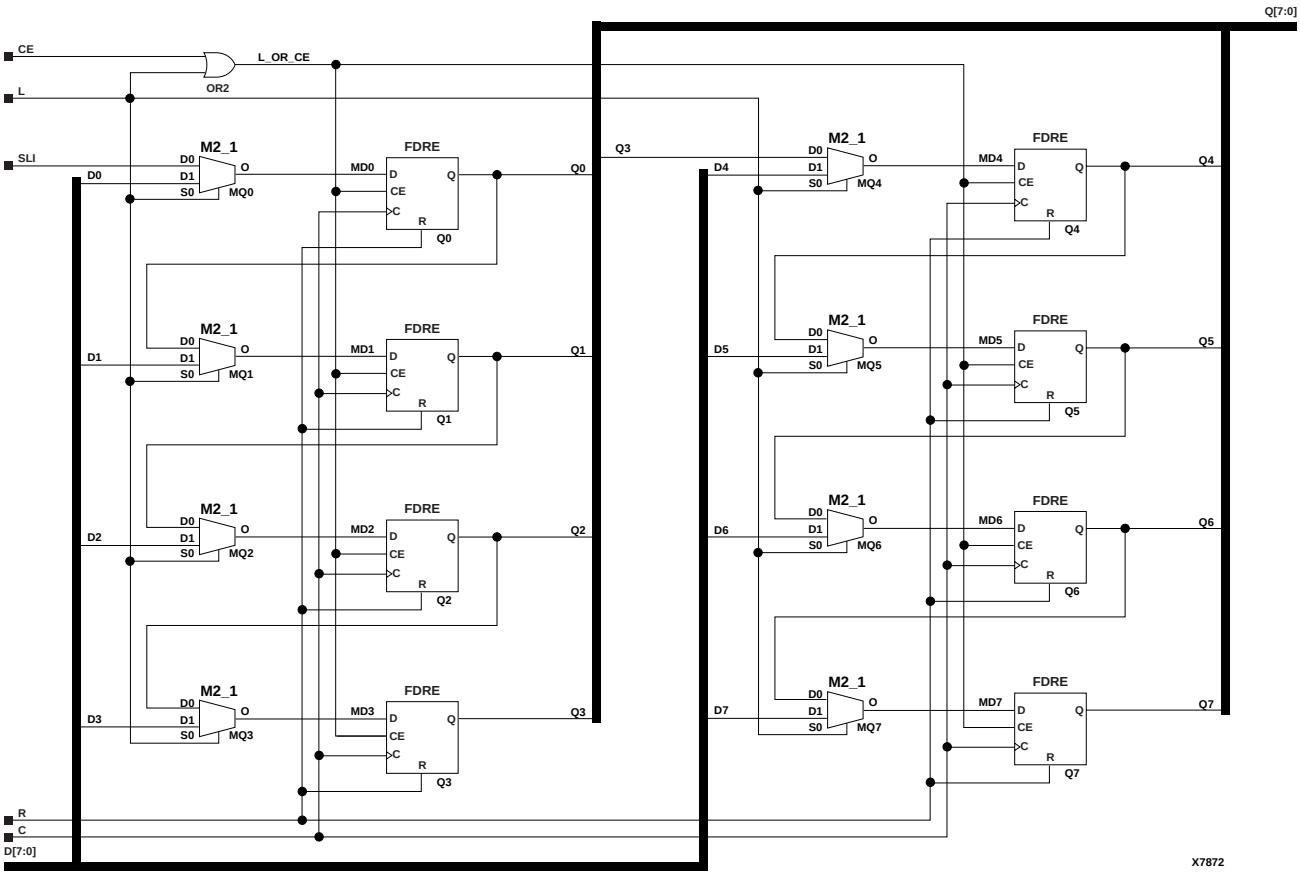
For Spartan-3E, power on conditions are simulated when global set/reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

Inputs						Outputs	
R	L	CE	SLI	Dz – D0	C	Q0	Qz – Q1
1	X	X	X	X	↑	0	0
0	1	X	X	Dz – D0	↑	D0	Dn
0	0	1	SLI	X	↑	SLI	qn-1
0	0	0	X	X	X	No Change	No Change

z = 3 for SR4RLE; z = 7 for SR8RLE; z = 15 for SR16RLE

qn-1 = state of referenced output one setup time prior to active clock transition



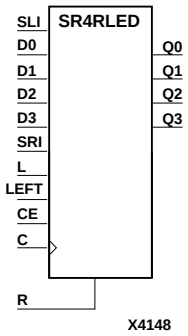
SR8RLE Implementation for Spartan-3E

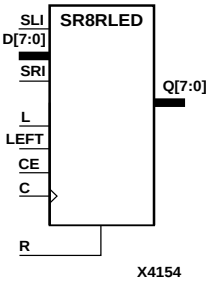
Usage

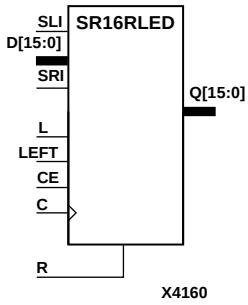
These design elements are inferred rather than instantiated.

SR4RLED, SR8RLED, SR16RLED

Macro: 4-, 8-, 16-Bit Shift Registers with Clock Enable and Synchronous Reset







SR4RLED, SR8RLED, and SR16RLED are 4-, 8-, and 16-bit shift registers, respectively, with shift-left (SLI) and shift-right (SRI) serial inputs, parallel inputs (D), parallel outputs (Q) and four control inputs — clock enable (CE), load enable (L), shift left / right (LEFT), and synchronous reset (R). The register ignores clock transitions when (CE) and (L) are Low. The synchronous (R), when High, overrides all other inputs during the Low-to-High clock (C) transition and resets the data outputs (Q) Low. When (L) is High and (R) is Low during the Low-to-High clock transition, the data on the (D) inputs is loaded into the corresponding (Q) bits of the register.

When (CE) is High and (L) and (R) are Low, data is shifted right or left, depending on the state of the LEFT input. If LEFT is High, data on (SLI) is loaded into (Q0) during the Low-to-High clock transition and shifted left (to Q1, Q2, and so forth) during subsequent clock transitions. If LEFT is Low, data on the (SRI) is loaded into the last (Q) output (Q3 for SR4RLED, Q7 for SR8RLED, or Q15 for SR16RLED) during the Low-to-High clock transition and shifted right (to Q2, Q1,... for SR4RLED; to Q6, Q5,... for SR8RLED; or to Q14, Q13,... for SR16RLED) during subsequent clock transitions. The truth table indicates the state of the (Q) outputs under all input conditions.

The register is asynchronously cleared, outputs Low, when power is applied.

For Spartan-3E, power on conditions are simulated when global set / reset (GSR) is active.

The active level of the GSR defaults to active-High but can be inverted by adding an inverter in front of the GSR input of the STARTUP_SPARTAN3E symbol.

SR4RLED Truth Table

Inputs								Outputs		
R	L	CE	LEFT	SLI	SRI	D3 – D0	C	Q0	Q3	Q2 – Q1
1	X	X	X	X	X	X	↑	0	0	0
0	1	X	X	X	X	D3 – D0	↑	D0	D3	Dn
0	0	0	X	X	X	X	X	No Change	No Change	No Change
0	0	1	1	SLI	X	X	↑	SLI	q2	qn-1
0	0	1	0	X	SRI	X	↑	q1	SRI	qn+1

qn-1 or qn+1 = state of referenced output one setup time prior to active clock transition

SR8RLED Truth Table

Inputs								Outputs		
R	L	CE	LEFT	SLI	SRI	D7– D0	C	Q0	Q7	Q6 – Q1
1	X	X	X	X	X	X	↑	0	0	0
0	1	X	X	X	X	D7 – D0	↑	D0	D7	Dn

SR8RLED Truth Table

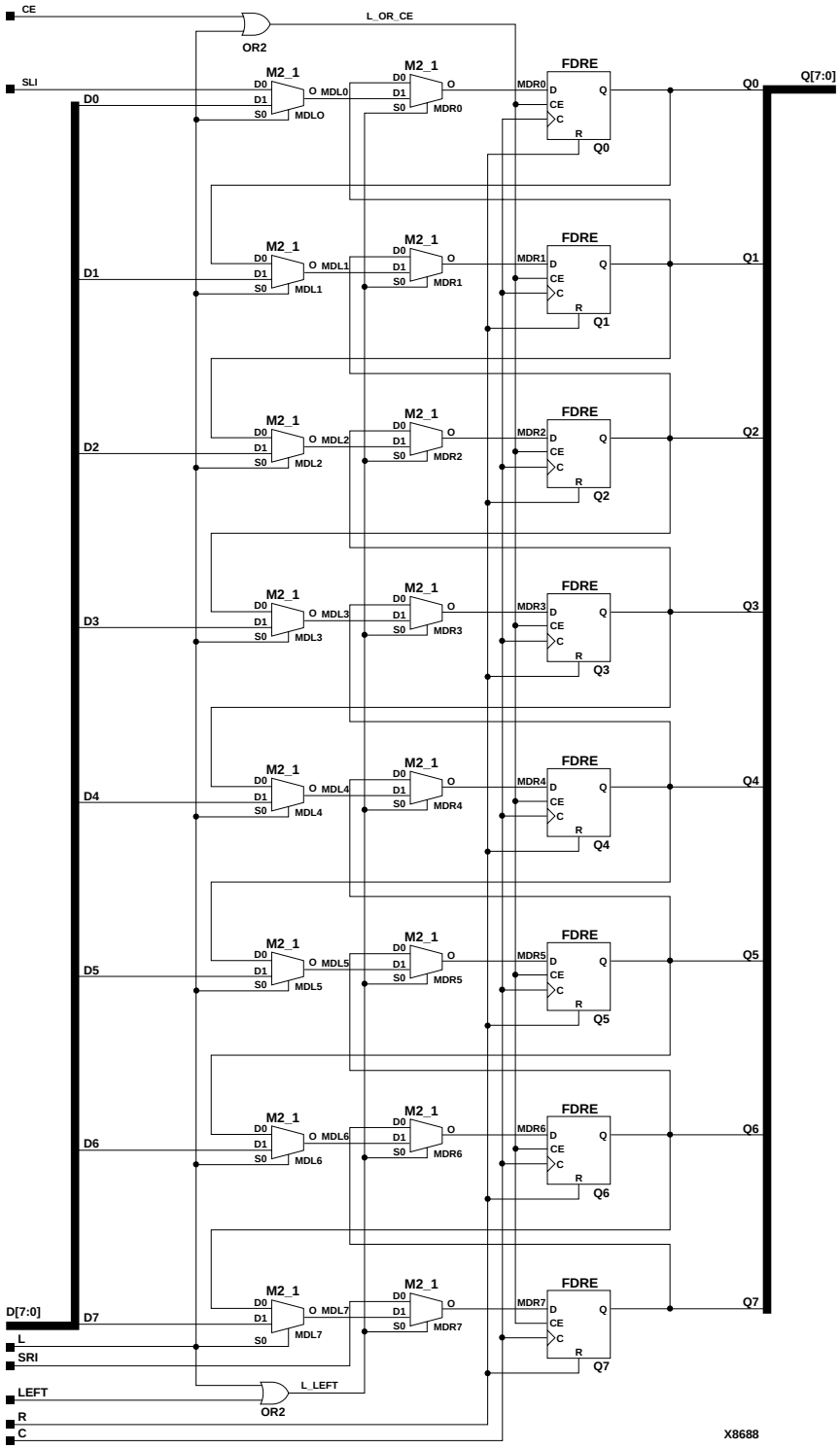
Inputs								Outputs		
R	L	CE	LEFT	SLI	SRI	D7– D0	C	Q0	Q7	Q6 – Q1
0	0	0	X	X	X	X	X	No Change	No Change	No Change
0	0	1	1	SLI	X	X	↑	SLI	q6	qn-1
0	0	1	0	X	SRI	X	↑	q1	SRI	qn+1

qn-1 or qn+1 = state of referenced output one setup time prior to active clock transition

SR16RLED Truth Table

Inputs								Outputs		
R	L	CE	LEFT	SLI	SRI	D15 – D0	C	Q0	Q15	Q14 – Q1
1	X	X	X	X	X	X	↑	0	0	0
0	1	X	X	X	X	D15 – D0	↑	D0	D15	Dn
0	0	0	X	X	X	X	X	No Change	No Change	No Change
0	0	1	1	SLI	X	X	↑	SLI	q14	qn-1
0	0	1	0	X	SRI	X	↑	q1	SRI	qn+1

qn-1 or qn+1 = state of referenced output one setup time prior to active clock transition



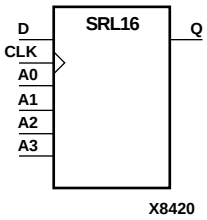
SR8RLED Implementation for Spartan-3E

Usage

These design elements are inferred rather than instantiated.

SRL16

Primitive: 16-Bit Shift Register Look-Up-Table (LUT)



SRL16 is a shift register look up table (LUT). The inputs A3, A2, A1, and A0 select the output length of the shift register. The shift register may be of a fixed, static length or it may be dynamically adjusted.

The shift register LUT contents are initialized by assigning a four-digit hexadecimal number to an INIT attribute. The first, or the left-most, hexadecimal digit is the most significant bit. If an INIT value is not specified, it defaults to a value of four zeros (0000) so that the shift register LUT is cleared during configuration.

The data (D) is loaded into the first bit of the shift register during the Low-to-High clock (CLK) transition. During subsequent Low-to-High clock transitions data is shifted to the next highest bit position as new data is loaded. The data appears on the Q output when the shift register length determined by the address inputs is reached.

Static Length Mode

To get a fixed length shift register, drive the A3 through A0 inputs with static values. The length of the shift register can vary from 1 bit to 16 bits as determined from the following formula:

Length = (8*A3) +(4*A2) + (2*A1) + A0 +1

If A3, A2, A1, and A0 are all zeros (0000), the shift register is one bit long. If they are all ones (1111), it is 16 bits long.

Dynamic Length Mode

The length of the shift register can be changed dynamically by changing the values driving the A3 through A0 inputs. For example, if A2, A1, and A0 are all ones (111) and A3 toggles between a one (1) and a zero (0), the length of the shift register changes from 16 bits to 8 bits.

Internally, the length of the shift register is always 16 bits and the input lines A3 through A0 select which of the 16 bits reach the output.

Inputs			Output
Am	**CLK**	**D**	**Q**
Am	X	X	Q(Am)
Am	↑	D	Q(Am-1)

m= 0, 1, 2, 3

Usage

This design element can be inferred or instantiated.

Available Attributes

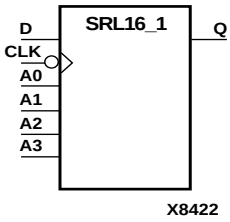
Attribute	Type	Default	Description
INIT	16-Bit Binary	16 zeroes	Sets the initial value of Q output after configuration

For More Information

Consult the Spartan-3E Data Sheets.

SRL16_1

Primitive: 16-Bit Shift Register Look-Up-Table (LUT) with Negative-Edge Clock



SRL16_1 is a shift register look up table (LUT). The inputs A3, A2, A1, and A0 select the output length of the shift register. The shift register may be of a fixed, static length or it may be dynamically adjusted. See "Static Length Mode" and "Dynamic Length Mode" in "SRL16".

The shift register LUT contents are initialized by assigning a four-digit hexadecimal number to an INIT attribute. The first, or the left-most, hexadecimal digit is the most significant bit. If an INIT value is not specified, it defaults to a value of four zeros (0000) so that the shift register LUT is cleared during configuration.

The data (D) is loaded into the first bit of the shift register during the High-to-Low clock (CLK) transition. During subsequent High-to-Low clock transitions data is shifted to the next highest bit position as new data is loaded. The data appears on the Q output when the shift register length determined by the address inputs is reached.

Inputs			Output
Am	**CLK**	**D**	**Q**
Am	X	X	Q(Am)
Am	↓	D	Q(Am-1)

m= 0, 1, 2, 3

Usage

This design element can be inferred or instantiated.

Available Attributes

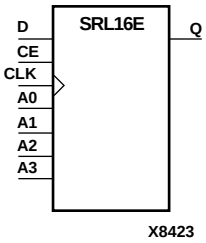
Attribute	Type	Default	Description
INIT	16-Bit Binary	16 0's	Sets the initial value of Q output after configuration

For More Information

Consult the Spartan-3E Data Sheets.

SRL16E

Primitive: 16-Bit Shift Register Look-Up-Table (LUT) with Clock Enable



SRL16E is a shift register look up table (LUT). The inputs A3, A2, A1, and A0 select the output length of the shift register. The shift register may be of a fixed, static length or dynamically adjusted. See "Static Length Mode" and "Dynamic Length Mode" in "SRL16".

The shift register LUT contents are initialized by assigning a four-digit hexadecimal number to an INIT attribute. The first, or the left-most, hexadecimal digit is the most significant bit. If an INIT value is not specified, it defaults to a value of four zeros (0000) so that the shift register LUT is cleared during configuration.

When CE is High, the data (D) is loaded into the first bit of the shift register during the Low-to-High clock (CLK) transition. During subsequent Low-to-High clock transitions, when CE is High, data is shifted to the next highest bit position as new data is loaded. The data appears on the Q output when the shift register length determined by the address inputs is reached.

When CE is Low, the register ignores clock transitions.

Inputs				Output
Am	**CE**	**CLK**	**D**	**Q**
Am	0	X	X	Q(Am)
Am	1	↑	D	Q(Am-1)

m= 0, 1, 2, 3

Usage

This design element can be inferred or instantiated.

Available Attributes

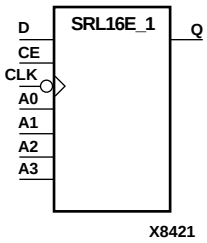
Attribute	Type	Default	Description
INIT	16-Bit Binary	16 0's	Sets the initial value of content and output of shift register after configuration

For More Information

Consult the Spartan-3E Data Sheets.

SRL16E_1

Primitive: 16-Bit Shift Register Look-Up-Table (LUT) with Negative-Edge Clock and Clock Enable



SRL16E_1 is a shift register look up table (LUT) with clock enable (CE). The inputs A3, A2, A1, and A0 select the output length of the shift register. The shift register may be of a fixed, static length or dynamically adjusted. See "Static Length Mode" and "Dynamic Length Mode" in the "SRL16".

The shift register LUT contents are initialized by assigning a four-digit hexadecimal number to an INIT attribute. The first, or the left-most, hexadecimal digit is the most significant bit. If an INIT value is not specified, it defaults to a value of four zeros (0000) so that the shift register LUT is cleared during configuration.

When CE is High, the data (D) is loaded into the first bit of the shift register during the High-to-Low clock (CLK) transition. During subsequent High-to-Low clock transitions, when CE is High, data is shifted to the next highest bit position as new data is loaded. The data appears on the Q output when the shift register length determined by the address inputs is reached.

When CE is Low, the register ignores clock transitions.

Inputs				Output
Am	**CE**	**CLK**	**D**	**Q**
Am	0	X	X	Q(Am)
Am	1	↓	D	Q(Am-1)

m= 0, 1, 2, 3

Usage

This design element can be inferred or instantiated.

Available Attributes

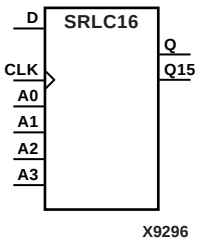
Attribute	Type	Default	Description
INIT	16-Bit Binary	16 0's	Sets the initial value of content and output of shift register after configuration

For More Information

Consult the Spartan-3E Data Sheets.

SRLC16

Primitive: 16-Bit Shift Register Look-Up-Table (LUT) with Carry



SRLC16 is a shift register look up table (LUT) with Carry. The inputs A3, A2, A1, and A0 select the output length of the shift register. The shift register may be of a fixed, static length, or it may be dynamically adjusted.

The shift register LUT contents are initialized by assigning a four-digit hexadecimal number to an INIT attribute. The first, or the left-most, hexadecimal digit is the most significant bit. If an INIT value is not specified, it defaults to a value of four zeros (0000) so that the shift register LUT is cleared during configuration.

The data (D) is loaded into the first bit of the shift register during the Low-to-High clock (CLK) transition. During subsequent Low-to-High clock transitions data is shifted to the next highest bit position as new data is loaded. The data appears on the Q output when the shift register length determined by the address inputs is reached.

The Q15 output is available for the user to cascade multiple shift register LUTs to create larger shift registers.

For information about the static length mode, see “Static Length Mode” in “SRL16”.

For information about the dynamic length mode, see “Dynamic Length Mode” in “SRL16”.

Inputs			Output
Am	**CLK**	**D**	**Q**
Am	X	X	Q(Am)
Am	↑	D	Q(Am-1)

m= 0, 1, 2, 3

Usage

This design element can be inferred or instantiated.

Available Attributes

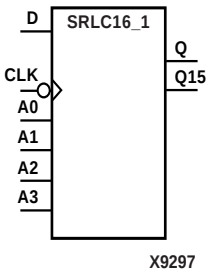
Attribute	Type	Default	Description
INIT	16-Bit Binary	16 0’s	Sets the initial value of content and output of shift register after configuration

For More Information

Consult the Spartan-3E Data Sheets.

SRLC16_1

Primitive: 16-Bit Shift Register Look-Up-Table (LUT) with Carry and Negative-Edge Clock



SRLC16_1 is a shift register look up table (LUT) with carry and a negative-edge clock. The inputs A3, A2, A1, and A0 select the output length of the shift register. The shift register may be of a fixed, static length or it may be dynamically adjusted. See "Static Length Mode" and "Dynamic Length Mode" in "SRL16".

The shift register LUT contents are initialized by assigning a four-digit hexadecimal number to an INIT attribute. The first, or the left-most, hexadecimal digit is the most significant bit. If an INIT value is not specified, it defaults to a value of four zeros (0000) so that the shift register LUT is cleared during configuration.

The data (D) is loaded into the first bit of the shift register during the High-to-Low clock (CLK) transition. During subsequent High-to-Low clock transitions data is shifted to the next highest bit position as new data is loaded. The data appears on the Q output when the shift register length determined by the address inputs is reached.

The Q15 output is available for the user to cascade multiple shift register LUTs to create larger shift registers.

Inputs			Output	
Am	**CLK**	**D**	**Q**	**Q15**
Am	X	X	Q(Am)	No Change
Am	↓	D	Q(Am-1)	Q14

m= 0, 1, 2, 3

Usage

This design element can inferred.

Available Attributes

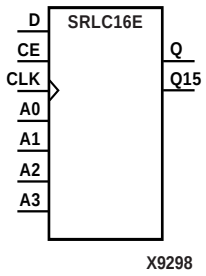
Attribute	Type	Default	Description
INIT	16-Bit Binary	16 0's	Sets the initial value of content and output of shift register after configuration

For More Information

Consult the Spartan-3E Data Sheets.

SRLC16E

Primitive: 16-Bit Shift Register Look-Up-Table (LUT) with Carry and Clock Enable



SRLC16E is a shift register look up table (LUT) with carry and clock enable. The inputs A3, A2, A1, and A0 select the output length of the shift register. The shift register may be of a fixed, static length or it may be dynamically adjusted.

The shift register LUT contents are initialized by assigning a four-digit hexadecimal number to an INIT attribute. The first, or the left-most, hexadecimal digit is the most significant bit. If an INIT value is not specified, it defaults to a value of four zeros (0000) so that the shift register LUT is cleared during configuration.

The data (D) is loaded into the first bit of the shift register during the Low-to-High clock (CLK) transition. When CE is High, during subsequent Low-to-High clock transitions, data is shifted to the next highest bit position as new data is loaded. The data appears on the Q output when the shift register length determined by the address inputs is reached.

The Q15 output is available for the user to cascade multiple shift register LUTs to create larger shift registers.

For information about the static length mode, see "Static Length Mode" in "SRL16".

For information about the dynamic length mode, see "Dynamic Length Mode" in "SRL16".

Inputs				Output	
Am	CLK	CE	D	Q	Q15
Am	X	0	X	Q(Am)	Q(15)
Am	X	1	X	Q(Am)	Q(15)
Am	↑	1	D	Q(Am-1)	Q15
m= 0, 1, 2, 3					

Usage

This design element can be inferred or instantiated.

Available Attributes

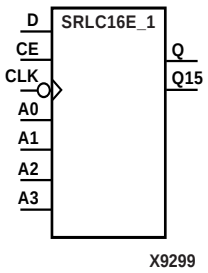
Attribute	Type	Default	Description
INIT	16-Bit Binary	16 0's	Sets the initial value of content and output of shift register after configuration

For More Information

Consult the Spartan-3E Data Sheets.

SRLC16E_1

Primitive: 16-Bit Shift Register Look-Up-Table (LUT) with Carry, Negative-Edge Clock, and Clock Enable



SRLC16E_1 is a shift register look up table (LUT) with carry, clock enable, and negative-edge clock. The inputs A3, A2, A1, and A0 select the output length of the shift register. The shift register may be of a fixed, static length or it may be dynamically adjusted. See "SRLC16" and "Dynamic Length Mode" in "SRL16".

The shift register LUT contents are initialized by assigning a four-digit hexadecimal number to an INIT attribute. The first, or the left-most, hexadecimal digit is the most significant bit. If an INIT value is not specified, it defaults to a value of four zeros (0000) so that the shift register LUT is cleared during configuration.

When (CE) is High, the data (D) is loaded into the first bit of the shift register during the High-to-Low clock (CLK) transition. During subsequent High-to-Low clock transitions data is shifted to the next highest bit position as new data is loaded when (CE) is High. The data appears on the (Q) output when the shift register length determined by the address inputs is reached.

The Q15 output is available for the user to cascade multiple shift register LUTs to create larger shift registers.

Inputs				Output	
Am	**CE**	**CLK**	**D**	**Q**	**Q15**
Am	0	X	X	Q(Am)	No Change
Am	1	X	X	Q(Am)	No Change
Am	1	↓	D	Q(Am-1)	Q14

m= 0, 1, 2, 3

Usage

This design element can be inferred or instantiated.

Available Attributes

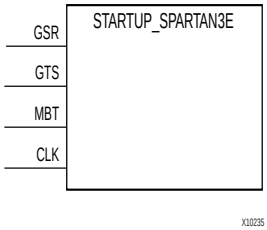
Attribute	Type	Allowed Values	Default	Description
INIT	16-Bit Binary	16 bits of 1's or 0's	X 0000	Sets the initial value of content and output of shift register after configuration

For More Information

Consult the Spartan-3E Data Sheets.

STARTUP_SPARTAN3E

Primitive: Spartan-3E User Interface to the GSR, GTS, Configuration Startup Sequence and Multi-Boot Trigger Circuitry



The STARTUP_SPARTAN3E component is used to allow the connection of ports or user circuitry to control certain dedicated circuitry and routes within the FPGA. Signals connected the GSR port of this component can control the global set/reset (referred to as GSR) of the device. The GSR net connects to all registers in the device and will place the registers into their initial value state. Connecting a signal to the GTS port will connect to the dedicated route controlling the three-state outputs of every pin in the device. Connecting a clock signal to the CLK input will allow the startup sequence after configuration to be synchronized to a user defined clock. The MBT (Multi-Boot Trigger) pin allows the triggering of a new configuration when the device is properly setup for this mode.

Usage

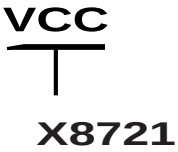
The STARTUP_SPARTAN3E component must be instantiated in order to be incorporated into the design. Do not connect any input not needed for the design.

For More Information

Consult the Spartan-3E Data Sheets.

VCC

Primitive: VCC-Connection Signal Tag



The VCC signal tag or parameter forces a net or input function to a logic High level. A net tied to VCC cannot have any other source.

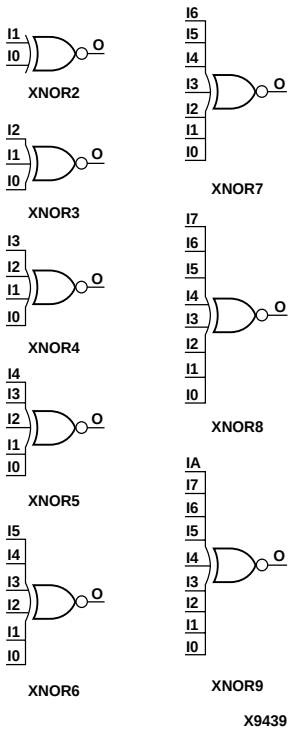
When the placement and routing software encounters a net or input function tied to VCC, it removes any logic that is disabled by the VCC signal. The VCC signal is only implemented when the disabled logic cannot be removed.

Usage

This design element can be instantiated or inferred.

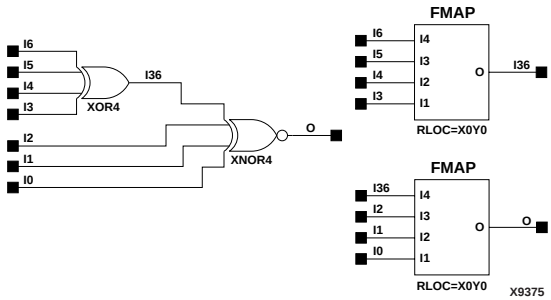
XNOR2-9

Macro: 2- to 9-Input XNOR Gates with Non-Inverted Inputs

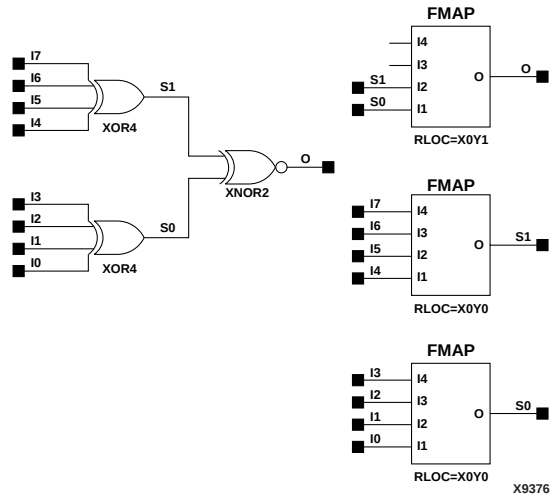


XNOR Gate Representations

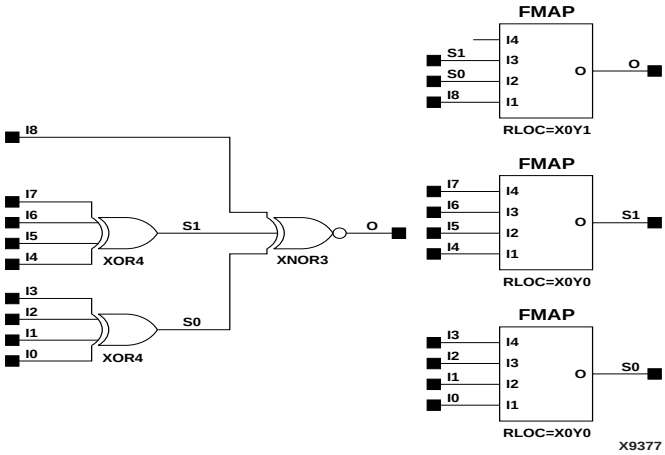
XNOR functions of up to nine inputs are available. All inputs are non-inverting. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.



XNOR7 Implementation for Spartan-3E



XNOR8 Implementation Spartan-3E



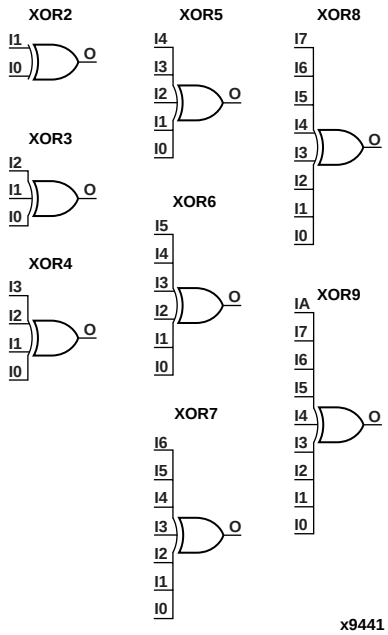
XNOR9 Implementation Spartan-3E

Usage

These design elements can be inferred or instantiated.

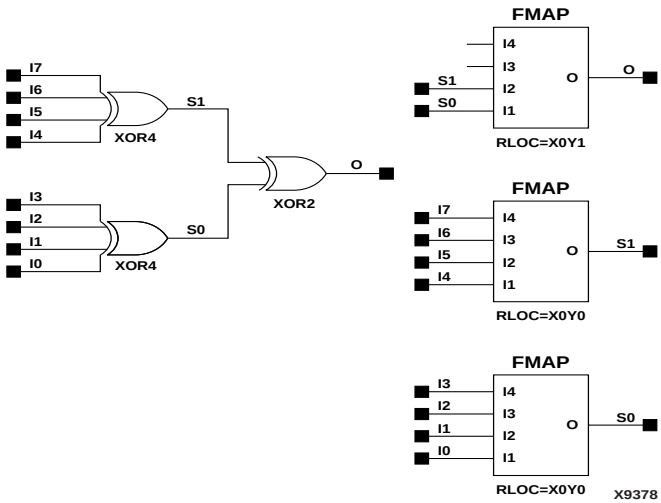
XOR2-9

Primitive and Macros: 2- to 9-Input XOR Gates with Non-Inverted Inputs



XOR Gate Representations

XOR functions of up to nine inputs are available. All inputs are non-inverting. Because each input uses a CLB resource, replace functions with unused inputs with functions having the necessary number of inputs.



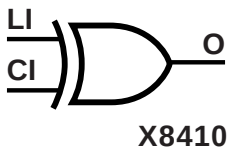
XOR8 Implementation for Spartan-3E

Usage

These design elements can be inferred or instantiated

XORCY

Primitive: XOR for Carry Logic with General Output



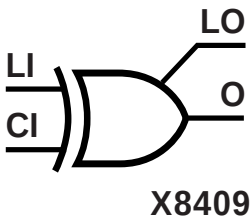
XORCY is a special XOR with general O output used for generating faster and smaller arithmetic functions.

Usage

Its O output is a general interconnect. See also "XORCY_D" and "XORCY_L".

XORCY_D

Primitive: XOR for Carry Logic with Dual Output



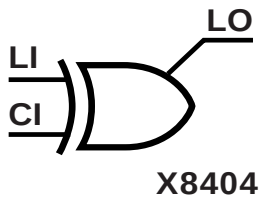
XORCY_D is a special XOR used for generating faster and smaller arithmetic functions.

Usage

XORCY_D has two functionally identical outputs, O and LO. The O output is a general interconnect. The LO output is used to connect to another output within the same CLB slice.

XORCY_L

Primitive: XOR for Carry Logic with Local Output



XORCY_L is a special XOR with local LO output used for generating faster and smaller arithmetic functions.

Usage

The LO output is used to connect to another output within the same CLB slice.