

---

# ***ProASIC3/E Macro Library Guide***

*for Software v6.1*



**Actel Corporation, Mountain View, CA 94043**

© 2004 Actel Corporation. All rights reserved.

Part Number: 50200029-0

Release: July 2004

No part of this document may be copied or reproduced in any form or by any means without prior written consent of Actel.

Actel makes no warranties with respect to this documentation and disclaims any implied warranties of merchantability or fitness for a particular purpose. Information in this document is subject to change without notice. Actel assumes no responsibility for any errors that may appear in this document.

This document contains confidential proprietary information that is not to be disclosed to any unauthorized person without prior written consent of Actel Corporation.

**Trademarks**

Actel and the Actel logo are registered trademarks of Actel Corporation.

Adobe and Acrobat Reader are registered trademarks of Adobe Systems, Inc.

All other products or brand names mentioned are trademarks or registered trademarks of their respective holders.

---

# Table of Contents

## Introduction

## Lists

<b>List of Combinational Macros</b> .....	<b>7</b>
<b>List of Sequential Macros</b> .....	<b>9</b>
<b>List of RAM Macros</b> .....	<b>11</b>
<b>List of Input/Output Macros</b> .....	<b>13</b>
<b>Alphabetical List of Macros</b> .....	<b>15</b>
<b>Combinational/Sequential Macros</b> .....	<b>19</b>
Combinational, AND .....	20
Combinational, AND-OR .....	25
Combinational, AND-OR-INVERT .....	32
Combinational, AND-OR-INVERT .....	33
Combinational, AND-XOR .....	35
Combinational, AND-XOR-INVERT .....	35
Combinational, 3-Input Gate .....	37
Buffers .....	44
Clock Interface .....	45
Sequential, D-Type Flip-Flop .....	47
Sequential D-Type Flip-Flop with Enable .....	53
Sequential, D-Type Flip-Flop .....	64
Sequential, Data Latch .....	80
Sequential, Data Latch with Clear .....	81
Sequential, Data Latch with Preset .....	86
Sequential .....	88
Combinational, Inverters .....	93
Combinational, AND-OR .....	94
Combinational, AND-OR-INVERT .....	97
Combinational, Multiplexer .....	98
Combinational, NAND .....	100
Combinational, NOR .....	104
Combinational, OR-AND .....	107
Combinational, OR-AND-INVERT .....	109
Combinational, OR .....	111
Combinational, XOR-AND .....	115
Combinational, XNOR-AND .....	116
Combinational, XNOR-NAND .....	117

Combinational, XNOR .....	.118
Combinational, XOR-OR .....	.119
Combinational, XOR .....	.120
Combinational, Gate .....	.121
<b>RAM and FIFO Macros .....</b>	<b>123</b>
RAM Macros .....	.124
FIFO Macros .....	.127
<b>I/O Macros .....</b>	<b>131</b>
Input/Output, General Use .....	.132
ProASIC3 and ProASIC3E Input IO Macros .....	.135
Bi-Directional IO Macros .....	.137
Clock Buffers .....	.139
Output Buffers .....	.140
Tri-State Buffer Macros .....	.142
Differential IO Macros .....	.145
DDR Macros .....	.147
<b>PLL Macros .....</b>	<b>149</b>
PLL Macros .....	.150

---

# Introduction

This macro library guide supports only the ProASIC3 (low cost) and ProASIC3E (enhanced) families. The ProASIC3/E devices offer a wide range of unique and special features and require a dedicated macro guide. For information on macros available in other families, see the *Antifuse Macro Library Guide* (for the MX, eX, SX, SX-A, and Axcelerator devices) or the *ProASIC/ProASIC<sup>PLUS</sup> MLG*, as appropriate.

The naming convention of previous antifuse families has been a source of confusion. With ProASIC3/E we are introducing a new naming convention for sequential macros that is unambiguous and extensible, making it possible to understand the function of the macros by their name alone.

The first two mandatory characters of the macro name will indicate the basic macro function:

- DF - D-type flip-flop
- TF - Toggle flip-flop
- JF - JK flip-flop
- DL - D-type latch

The next mandatory character indicates the output polarity:

- I - output inverted (QN with bubble)
- N - output non-inverted (Q without bubble)

The next mandatory number indicates the polarity of the clock or gate:

- 1 - rising edge triggered flip-flop or transparent high latch (non-bubbled)
- 0 - falling edge triggered flip-flop or transparent low latch (bubbled)

The next two optional characters indicate the polarity of the Enable pin, if present:

- E0 - active low enable (bubbled)
- E1 - active high enable (non-bubbled)

The next two optional characters indicate the polarity of the asynchronous Preset pin, if present:

- P0 - active low preset (bubbled)
- P1 - active high preset (non-bubbled)

The next two optional characters indicate the polarity of the asynchronous Clear pin, if present:

- C0 - active low preset (bubbled)

- C1 - active high preset (non-bubbled)

Combinatorial macros all use one tile in the ProASIC3/E families.



---

# List of Combinational Macros

AND2	20	AXO11	41
AND2A	20	AXO12	41
AND2B	21	AXO13	42
AND3	21	AXO14	42
AND3A	22	AXO15	43
AND3B	22	AXO17	43
AND3C	23	BUFF	44
AO1	23	BUFD	44
AO12	25	CLKINT	45
AO13	25	GND	93
AO14	26	INV	93
AO15	26	INVD	94
AO16	27	MAJ3	94
AO17	27	MAJ3X	95
AO18	28	MAJ3XI	95
AO1A	28	MIN3	96
AO1B	29	MIN3X	96
AO1C	29	MIN3XI	97
AO1D	30	MX2	98
AO1E	30	MX2A	98
AOI1	31	MX2B	99
AOI1A	32	MX2C	99
AOI1B	32	NAND2	100
AOI1C	33	NAND2A	101
AOI1D	33	NAND2B	101
AOI5	34	NAND3	102
AX1	35	NAND3A	102
AX1A	35	NAND3B	103
AX1B	36	NAND3C	103
AX1C	36	NOR2	104
AX1D	37	NOR2A	104
AX1E	37	NOR2B	105
AXO1	38	NOR3	105
AXO2	38	NOR3A	106
AXO3	39	NOR3B	106
AXO5	39	NOR3C	107
AXO6	40	OA1	107
AXO7	40	OA1A	108

OA1B .....	108
OA1C .....	109
OAI1 .....	109
OR2 .....	111
OR2A .....	111
OR2B .....	112
OR3 .....	112
OR3A .....	113
OR3B .....	113
OR3C .....	114
VCC .....	114
XA1 .....	115
XA1A .....	115
XA1B .....	116
XA1C .....	116
XAI1 .....	117
XAI1A .....	117
XNOR2 .....	118
XNOR3 .....	118
XO1 .....	119
XO1A .....	119
XOR2 .....	120
XOR3 .....	120
ZOR3 .....	121
ZOR3I .....	121



---

## List of Sequential Macros

DFN1 .....	47	DFIOE1C1 .....	67
DFI1 .....	47	DFI1E0C1 .....	67
DFN0 .....	48	DFIOE0C1 .....	68
DFIO .....	48	DFI1E1C0 .....	68
DFN1C1 .....	49	DFIOE1C0 .....	69
DFN0C1 .....	49	DFI1E0C0 .....	69
DFC1B .....	50	DFIOE0C0 .....	70
DFI1C1 .....	50	DFI1E1P1 .....	70
DFN0C0 .....	51	DFIOE1P1 .....	71
DFI1C0 .....	52	DFI1E0P1 .....	71
DFIOC1 .....	52	DFIOE0P1 .....	72
DFIOC0 .....	53	DFI1E1P0 .....	72
DFN1E1 .....	53	DFIOE1P0 .....	73
DFN1E0 .....	54	DFI1E0P0 .....	73
DFN0E0 .....	54	DFIOE0P0 .....	74
DFN1E1C0 .....	55	DFN0P1 .....	74
DFN0E1C0 .....	55	DFN1P0 .....	75
DFN1E0C0 .....	56	DFI1P1 .....	75
DFN0E0C0 .....	56	DFN0P0 .....	76
DFN1E1C1 .....	57	DFI1P0 .....	76
DFN0E1C1 .....	57	DFI0P1 .....	77
DFN1E0C1 .....	58	DFI0P0 .....	77
DFN0E0C1 .....	59	DFN1P1C1 .....	78
DFN1E1P1 .....	59	DFI1P1C1 .....	78
DFN0E1P1 .....	60	DFN0P1C1 .....	79
DFN1E0P1 .....	60	DFI0P1C1 .....	79
DFN0E0P1 .....	61	DLN1 .....	80
DFN1E1P0 .....	61	DLI1 .....	80
DFN0E1P0 .....	62	DLN0 .....	81
DFN1E0P0 .....	62	DLIO .....	81
DFN0E0P0 .....	63	DLN1C0 .....	82
DFN0E1 .....	63	DLN1C1 .....	82
DFN1P1 .....	64	DLN0C1 .....	83
DFI1E1 .....	64	DLI1C1 .....	83
DFIOE1 .....	65	DLIOC1 .....	84
DFI1E0 .....	65	DLN0C0 .....	84
DFIOE0 .....	66	DLI1C0 .....	85
DFI1E1C1 .....	66	DLIOC0 .....	85

DLN1P1 .....	86
DLN0P1 .....	86
DLN1P0 .....	87
DLN0P0 .....	87
DLI1P0 .....	88
DLI0P0 .....	88
DLI1P1 .....	89
DLI0P1 .....	89
DLN1P1C1 .....	90
DLI1P1C1 .....	90
DLN0P1C1 .....	91
DLI0P1C1 .....	91

---

## List of RAM Macros

RAM4K9 and RAM512X18 . . . . .	124
FIFO4K18 . . . . .	127



---

## List of Input/Output Macros

BIBUF .....	132
CLKBIBUF .....	132
CLKBUF .....	133
INBUF .....	133
OUTBUF .....	134
TRIBUFF .....	134
INBUF_X .....	136
BIBUF_X .....	137
CLKBUF_X .....	139
OUTBUF_X .....	141
TRIBUFF_X .....	143
INBUF_LVDS; INBUF_LVPECL .....	145
CLKBUF_LVDS; CLKBUF_LVPECL .....	145
OUTBUF_LVDS; OUTBUF_LVPECL .....	146
DDR_REG .....	147
DDR_OUT .....	147
PLL .....	150
PLLINT .....	151
UJTAG .....	152
UFROM .....	152



---

## Alphabetical List of Macros

AND2	20	AXOI1	41
AND2A	20	AXOI2	41
AND2B	21	AXOI3	42
AND3	21	AXOI4	42
AND3A	22	AXOI5	43
AND3B	22	AXOI7	43
AND3C	23	BIBUF	132
AO1	23	BIBUF_X	137
AO12	25	BUFD	44
AO13	25	BUFF	44
AO14	26	CLKBIBUF	132
AO15	26	CLKBUF	133
AO16	27	CLKBUF_LVDS; CLKBUF_LVPECL	145
AO17	27	CLKBUF_X	139
AO18	28	CLKINT	45
AO1A	28	DDR_OUT	147
AO1B	29	DDR_REG	147
AO1C	29	DFC1B	50
AO1D	30	DFI0	48
AO1E	30	DFI0C0	53
AOI1	31	DFI0C1	52
AOI1A	32	DFI0E0	66
AOI1B	32	DFI0E0C0	70
AOI1C	33	DFI0E0C1	68
AOI1D	33	DFI0E0P0	74
AOI5	34	DFI0E0P1	72
AX1	35	DFI0E1	65
AX1A	35	DFI0E1C0	69
AX1B	36	DFI0E1C1	67
AX1C	36	DFI0E1P0	73
AX1D	37	DFI0E1P1	71
AX1E	37	DFI0P0	77
AXO1	38	DFI0P1	77
AXO2	38	DFI0P1C1	79
AXO3	39	DFI1	47
AXO5	39	DFI1C0	52
AXO6	40	DFI1C1	50
AXO7	40	DFI1E0	65

DFI1E0C0	69	DLI0P0	88
DFI1E0C1	67	DLI0P1	89
DFI1E0P0	73	DLI0P1C1	91
DFI1E0P1	71	DLI1	80
DFI1E1	64	DLI1C0	85
DFI1E1C0	68	DLI1C1	83
DFI1E1C1	66	DLI1P0	88
DFI1E1P0	72	DLI1P1	89
DFI1E1P1	70	DLI1P1C1	90
DFI1P0	76	DLN0	81
DFI1P1	75	DLN0C0	84
DFI1P1C1	78	DLN0C1	83
DFN0	48	DLN0P0	87
DFN0C0	51	DLN0P1	86
DFN0C1	49	DLN0P1C1	91
DFN0E0	54	DLN1	80
DFN0E0C0	56	DLN1C0	82
DFN0E0C1	59	DLN1C1	82
DFN0E0P0	63	DLN1P0	87
DFN0E0P1	61	DLN1P1	86
DFN0E1	63	DLN1P1C1	90
DFN0E1C0	55	FIFO4K18	127
DFN0E1C1	57	GND	93
DFN0E1P0	62	INBUF	133
DFN0E1P1	60	INBUF_LVDS; INBUF_LVPECL	145
DFN0P0	76	INBUF_X	136
DFN0P1	74	INV	93
DFN0P1C1	79	INVD	94
DFN1	47	MAJ3	94
DFN1C1	49	MAJ3X	95
DFN1E0	54	MAJ3XI	95
DFN1E0C0	56	MIN3	96
DFN1E0C1	58	MIN3X	96
DFN1E0P0	62	MIN3XI	97
DFN1E0P1	60	MX2	98
DFN1E1	53	MX2A	98
DFN1E1C0	55	MX2B	99
DFN1E1C1	57	MX2C	99
DFN1E1P0	61	NAND2	100
DFN1E1P1	59	NAND2A	101
DFN1P0	75	NAND2B	101
DFN1P1	64	NAND3	102
DFN1P1C1	78	NAND3A	102
DLIO	81	NAND3B	103
DLIOC0	85	NAND3C	103
DLIOC1	84	NOR2	104



---

NOR2A	104
NOR2B	105
NOR3	105
NOR3A	106
NOR3B	106
NOR3C	107
OA1	107
OA1A	108
OA1B	108
OA1C	109
OAI1	109
OR2	111
OR2A	111
OR2B	112
OR3	112
OR3A	113
OR3B	113
OR3C	114
OUTBUF	134
OUTBUF_LVDS; OUTBUF_LVPECL	146
OUTBUF_X	141
PLL	150
PLLINT	151
RAM4K9 and RAM512X18	124
TRIBUFF	134
TRIBUFF_X	143
UFROM	152
UJTAG	152
VCC	114
XA1	115
XA1A	115
XA1B	116
XA1C	116
XAI1	117
XAI1A	117
XNOR2	118
XNOR3	118
XO1	119
XO1A	119
XOR2	120
XOR3	120
ZOR3	121
ZOR3I	121

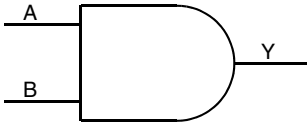


---

# Combinational/Sequential Macros

## AND2

ProASIC3, ProASIC3E



### Function

2-Input AND

### Truth Table

A	B	Y
X	0	0
0	X	0
1	1	1

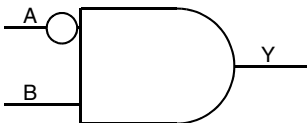
Input  
A, B

Output  
Y

Family	Tiles
All	1

## AND2A

ProASIC3, ProASIC3E



### Function

2-Input AND with active low A Input

### Truth Table

A	B	Y
X	0	0
0	1	1
1	X	0

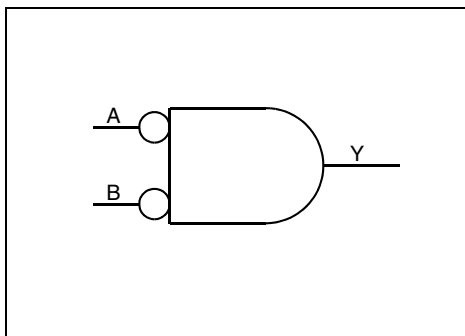
Input  
A, B

Output  
Y

Family	Tiles
All	1

**AND2B**

ProASIC3, ProASIC3E

**Function**

2-Input AND with active low Inputs

**Truth Table**

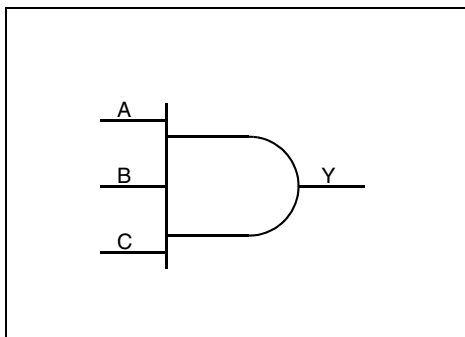
A	B	Y
0	0	1
X	1	0
1	X	0

**Input**  
A, B**Output**  
Y

Family	Tiles
All	1

**AND3**

ProASIC3, ProASIC3E

**Function**

3-Input AND

**Truth Table**

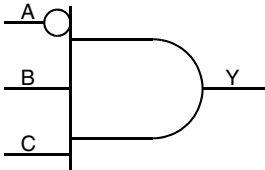
A	B	C	Y
X	X	0	0
X	0	X	0
0	X	X	0
1	1	1	1

**Input**  
A, B, C**Output**  
Y

Family	Tiles
All	1

## AND3A

ProASIC3, ProASIC3E

**Function**

3-Input AND with active low A-Input

**Truth Table**

A	B	C	Y
X	X	0	0
X	0	X	0
0	1	1	1
1	X	X	0

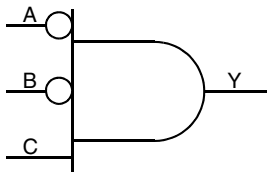
**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

## AND3B

ProASIC3, ProASIC3E

**Function**

3-Input AND with active low A- and B-Inputs

**Truth Table**

A	B	C	Y
X	X	0	0
0	0	1	1
X	1	X	0
1	X	X	0

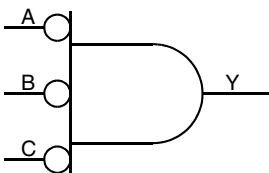
**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

## AND3C

ProASIC3, ProASIC3E



## Function

3-Input AND with active low Inputs

## Truth Table

A	B	C	Y
0	0	0	1
X	X	1	0
X	1	X	0
1	X	X	0

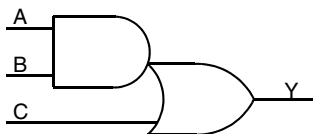
Input  
A, B, C

Output  
Y

Family	Tiles
All	1

## AO1

ProASIC3, ProASIC3E



## Function

3-Input AND-OR

## Truth Table

A	B	C	Y
X	0	0	0
X	X	1	1
0	X	0	0
1	1	X	1

Input  
A, B, C

Output  
Y

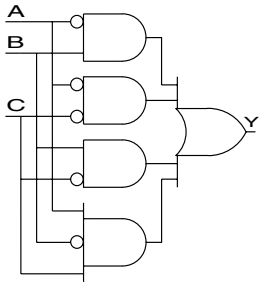
Family	Tiles
All	1





## AO12

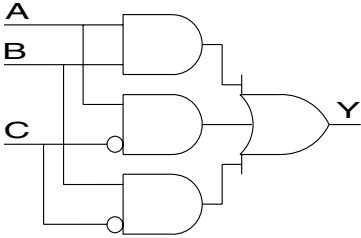
ProASIC3, ProASIC3E

		<b>Function</b>																																					
		3-Input AND-OR																																					
<b>Input</b> A, B, C		<b>Output</b> Y		<b>Truth Table</b>																																			
				<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>C</th> <th>Y</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>0</td></tr> </tbody> </table>	A	B	C	Y	0	0	0	1	1	0	0	0	0	1	0	1	1	1	0	1	0	0	1	0	1	0	1	1	0	1	1	1	1	1	1
A	B	C	Y																																				
0	0	0	1																																				
1	0	0	0																																				
0	1	0	1																																				
1	1	0	1																																				
0	0	1	0																																				
1	0	1	1																																				
0	1	1	1																																				
1	1	1	0																																				

Family	Tiles
All	1

## AO13

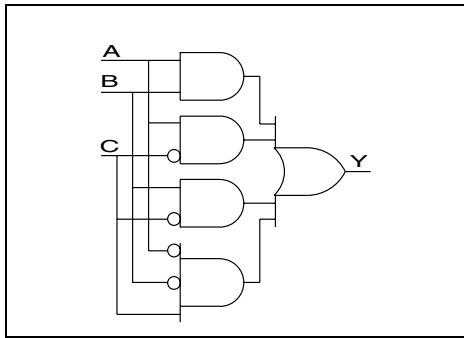
ProASIC3, ProASIC3E

		<b>Function</b>																																					
		3-Input AND-OR																																					
<b>Input</b> A, B, C		<b>Output</b> Y		<b>Truth Table</b>																																			
				<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>C</th> <th>Y</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td></tr> </tbody> </table>	A	B	C	Y	0	0	0	0	1	0	0	1	0	1	0	1	1	1	0	1	0	0	1	0	1	0	1	0	0	1	1	0	1	1	1
A	B	C	Y																																				
0	0	0	0																																				
1	0	0	1																																				
0	1	0	1																																				
1	1	0	1																																				
0	0	1	0																																				
1	0	1	0																																				
0	1	1	0																																				
1	1	1	1																																				

Family	Tiles
All	1

## AO14

ProASIC3, ProASIC3E



Input  
A, B, C

Output  
Y

## Function

3-Input AND-OR

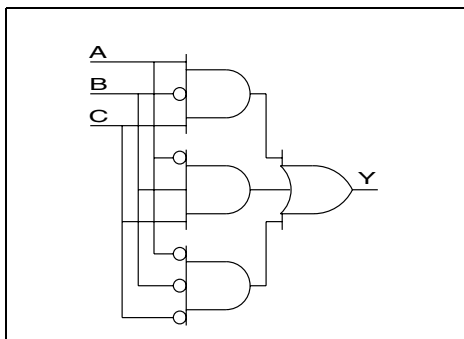
## Truth Table

A	B	C	Y
0	0	0	0
1	0	0	1
0	1	0	1
1	1	0	1
0	0	1	1
1	0	1	0
0	1	1	0
1	1	1	1

Family	Tiles
All	1

## AO15

ProASIC3, ProASIC3E



Input  
A, B, C

Output  
Y

## Function

3-Input AND-OR

## Truth Table

A	B	C	Y
0	0	0	1
1	0	0	0
0	1	0	0
1	1	0	0
0	0	1	0
1	0	1	1
0	1	1	1
1	1	1	0

Family	Tiles
All	1

## AO16

ProASIC3, ProASIC3E

		<b>Function</b> 3-Input AND-OR																																			
		<b>Truth Table</b> <table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>C</th> <th>Y</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>0</td></tr> </tbody> </table>		A	B	C	Y	0	0	0	0	1	0	0	0	0	1	0	0	1	1	0	1	0	0	1	1	1	0	1	0	0	1	1	0	1	1
A	B	C	Y																																		
0	0	0	0																																		
1	0	0	0																																		
0	1	0	0																																		
1	1	0	1																																		
0	0	1	1																																		
1	0	1	0																																		
0	1	1	0																																		
1	1	1	0																																		
<b>Input</b> A, B, C	<b>Output</b> Y																																				

Family	Tiles
All	1

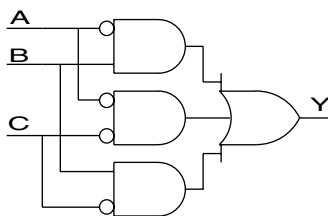
## AO17

ProASIC3, ProASIC3E

		<b>Function</b> 3-Input AND-OR																																			
		<b>Truth Table</b> <table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>C</th> <th>Y</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td></tr> </tbody> </table>		A	B	C	Y	0	0	0	0	1	0	0	0	0	1	0	1	1	1	0	0	0	0	1	1	1	0	1	0	0	1	1	0	1	1
A	B	C	Y																																		
0	0	0	0																																		
1	0	0	0																																		
0	1	0	1																																		
1	1	0	0																																		
0	0	1	1																																		
1	0	1	0																																		
0	1	1	0																																		
1	1	1	1																																		
<b>Input</b> A, B, C	<b>Output</b> Y																																				

Family	Tiles
All	1

## AO18



## Function

3-Input AND-OR

## Truth Table

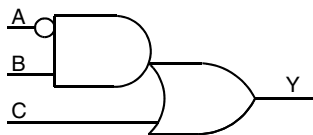
A	B	C	Y
0	0	0	1
1	0	0	0
0	1	0	1
1	1	0	1
0	0	1	0
1	0	1	0
0	1	1	1
1	1	1	0

Input  
A, B, C

Output  
Y

Family	Tiles
All	1

## AO1A



## Function

3-Input AND-OR with active low A-Input

## Truth Table

A	B	C	Y
X	0	0	0
X	X	1	1
0	1	X	1
1	X	0	0

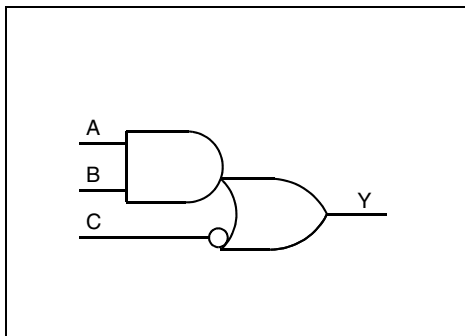
Input  
A, B, C

Output  
Y

Family	Tiles
All	1

## AO1B

ProASIC3, ProASIC3E

**Function**

3-Input AND-OR with active low C-Input

**Truth Table**

A	B	C	Y
X	X	0	1
X	0	1	0
0	X	1	0
1	1	X	1

**Input**

A, B, C

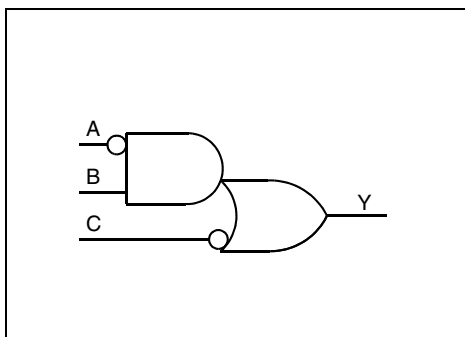
**Output**

Y

Family	Tiles
All	1

## AO1C

ProASIC3, ProASIC3E

**Function**

3-Input AND-OR with active low A- and C-Inputs

**Truth Table**

A	B	C	Y
X	X	0	1
X	0	1	0
0	1	X	1
1	X	1	0

**Input**

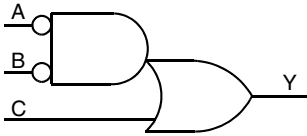
A, B, C

**Output**

Y

Family	Tiles
All	1

## AO1D

**Function**

3-Input AND-OR with active low A- and B-Inputs

**Truth Table**

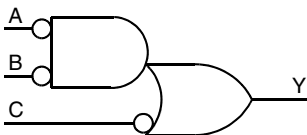
A	B	C	Y
0	0	X	1
X	1	0	0
X	X	1	1
1	X	0	0

**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

## AO1E

**Function**

3-Input AND-OR with active low Inputs

**Truth Table**

A	B	C	Y
X	X	0	1
0	0	X	1
X	1	1	0
1	X	1	0

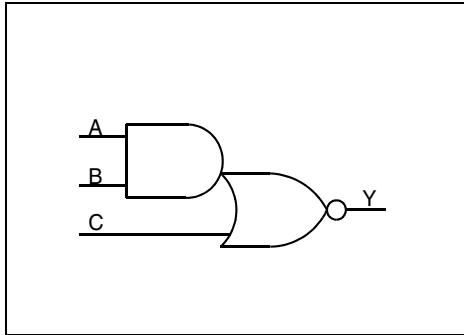
**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

## AOI1

ProASIC3, ProASIC3E



**Function**  
3-Input AND-OR-INVERT

**Truth Table**

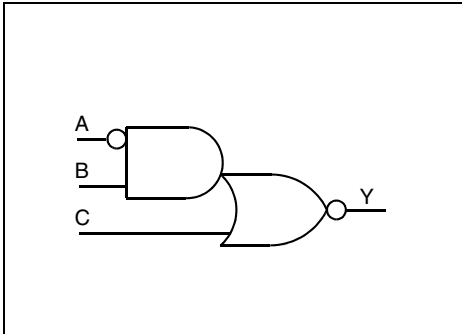
A	B	C	Y
X	0	0	1
X	X	1	0
0	X	0	1
1	1	X	0

**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

## AOI1A

**Function**

3-Input AND-OR-INVERT with active low A-Input

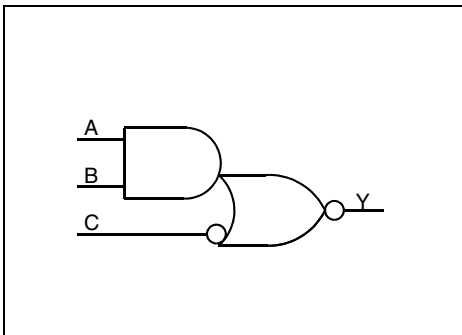
**Truth Table**

A	B	C	Y
X	0	0	1
X	X	1	0
0	1	X	0
1	X	0	1

Input	Output
A, B, C	Y

Family	Tiles
All	1

## AOI1B

**Function**

3-Input AND-OR-INVERT with active low C-Input

**Truth Table**

A	B	C	Y
X	X	0	0
X	0	1	1
0	X	1	1
1	1	X	0

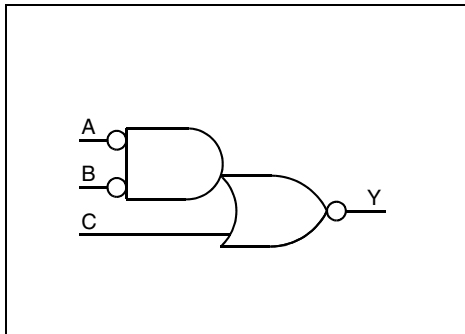
Input	Output
A, B, C	Y

Family	Tiles
All	1



## AOI1C

ProASIC3, ProASIC3E

**Function**

3 Input AND-OR-INVERT with active low A- and B-Inputs

**Truth Table**

A	B	C	Y
0	0	X	0
X	1	0	1
X	X	1	0
1	X	0	1

**Input**

A, B, C

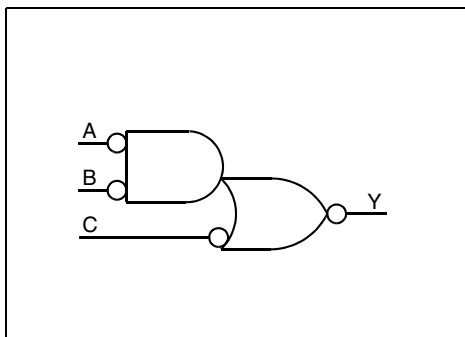
**Output**

Y

Family	Tiles
All	1

## AOI1D

ProASIC3, ProASIC3E

**Function**

3-Input AND-OR-INVERT with active low Inputs

**Truth Table**

A	B	C	Y
X	X	0	0
0	0	X	0
X	1	1	1
1	X	1	1

**Input**

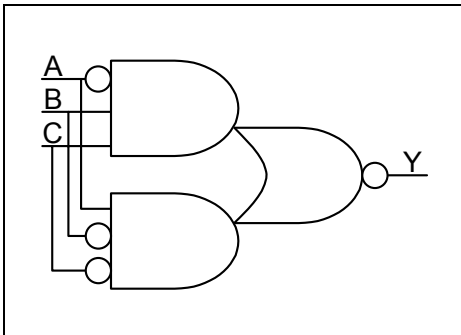
A, B, C

**Output**

Y

Family	Tiles
All	1

## AOI5

**Function**

3-Input AND-OR-INVERT

**Truth Table**

A	B	C	Y
0	0	0	1
1	0	0	0
0	1	0	1
1	1	0	1
0	0	1	1
1	0	1	1
0	1	1	0

**Input**

A, B, C

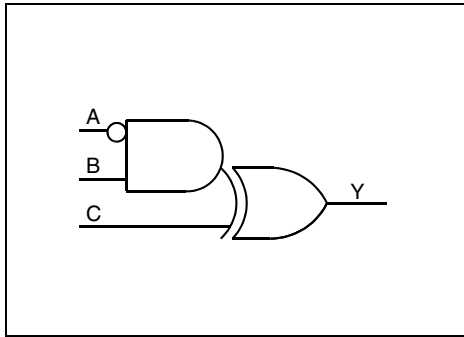
**Output**

Y

Family	Tiles
All	1

## AX1

ProASIC3, ProASIC3E



<b>Input</b> A, B, C	<b>Output</b> Y
-------------------------	--------------------

**Function**

3-Input AND-XOR with active low A-Input

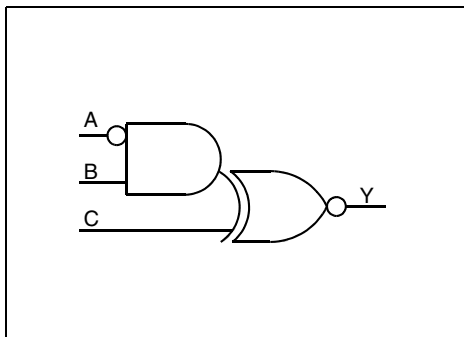
**Truth Table**

A	B	C	Y
X	0	0	0
X	0	1	1
0	1	0	1
0	1	1	0
1	X	0	0
1	X	1	1

Family	Tiles
All	1

## AX1A

ProASIC3, ProASIC3E



<b>Input</b> A, B, C	<b>Output</b> Y
-------------------------	--------------------

**Function**

3-Input AND-XOR-INVERT with active low A-Input

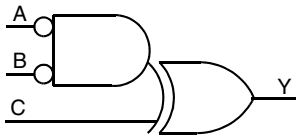
**Truth Table**

A	B	C	Y
X	0	0	1
X	0	1	0
0	1	0	0
0	1	1	1
1	X	0	1
1	X	1	0

Family	Tiles
All	1

## AX1B

ProASIC3, ProASIC3E

**Function**

3-Input AND-XOR with active low A- and B-Inputs

**Truth Table**

A	B	C	Y
0	0	0	1
0	0	1	0
X	1	0	0
X	1	1	1
1	X	0	0
1	X	1	1

**Input**

A, B, C

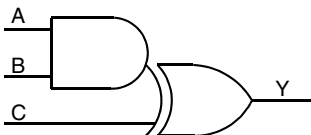
**Output**

Y

Family	Tiles
All	1

## AX1C

ProASIC3, ProASIC3E

**Function**

3-Input AND-XOR

**Truth Table**

A	B	C	Y
X	0	0	0
X	0	1	1
0	X	0	0
0	X	1	1
1	1	0	1
1	1	1	0

**Input**

A, B, C

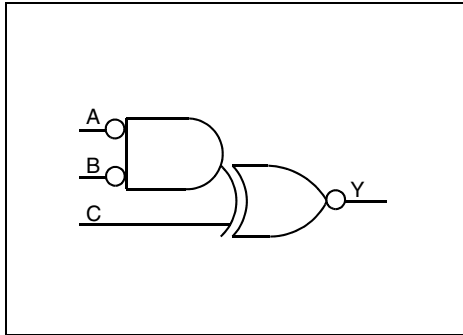
**Output**

Y

Family	Tiles
All	1

## AX1D

ProASIC3, ProASIC3E



**Input**  
A, B, C

**Output**  
Y

**Function**  
3-Input AND-XNOR

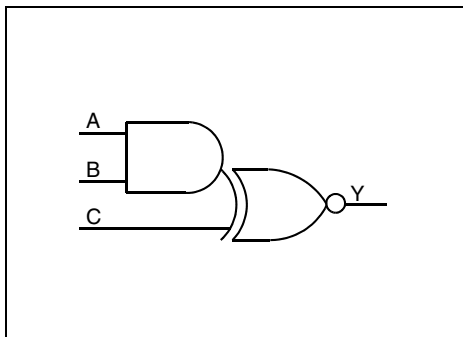
**Truth Table**

A	B	C	Y
0	0	0	0
1	0	0	1
0	1	0	1
1	1	0	1
0	0	1	1
1	0	1	0
0	1	1	0
1	1	1	0

Family	Tiles
All	1

## AX1E

ProASIC3, ProASIC3E



**Input**  
A, B, C

**Output**  
Y

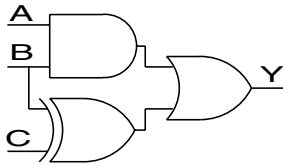
**Function**  
3-Input AND-XNOR

**Truth Table**

A	B	C	Y
0	0	0	1
1	0	0	1
0	1	0	1
1	1	0	0
0	0	1	0
1	0	1	0
0	1	1	0
1	1	1	1

Family	Tiles
All	1

## AXO1

**Function**

3-Input Combinatorial Gate

**Truth Table**

A	B	C	Y
0	0	0	0
1	0	0	0
0	1	0	1
1	1	0	1
0	0	1	1
1	0	1	1
0	1	1	0
1	1	1	1

**Input**

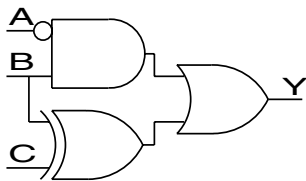
A, B, C

**Output**

Y

Family	Tiles
All	1

## AXO2

**Function**

3-Input Combinatorial Gate

**Truth Table**

A	B	C	Y
0	0	0	0
1	0	0	0
0	1	0	1
1	1	0	1
0	0	1	1
1	0	1	1
0	1	1	1
1	1	1	0

**Input**

A, B, C

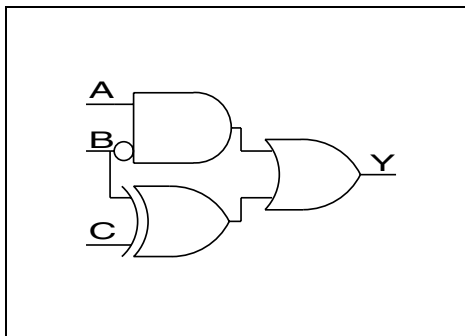
**Output**

Y

Family	Tiles
All	1

## AXO3

ProASIC3, ProASIC3E



**Input**  
A, B, C

**Output**  
Y

**Function**

3-Input Combinatorial Gate

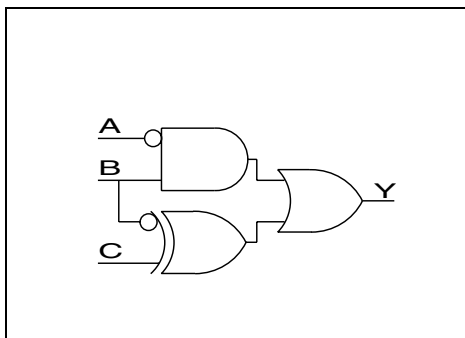
**Truth Table**

A	B	C	Y
0	0	0	0
1	0	0	1
0	1	0	1
1	1	0	1
0	0	1	1
1	0	1	1
0	1	1	0
1	1	1	0

Family	Tiles
All	1

## AXO5

ProASIC3, ProASIC3E



**Input**  
A, B, C

**Output**  
Y

**Function**

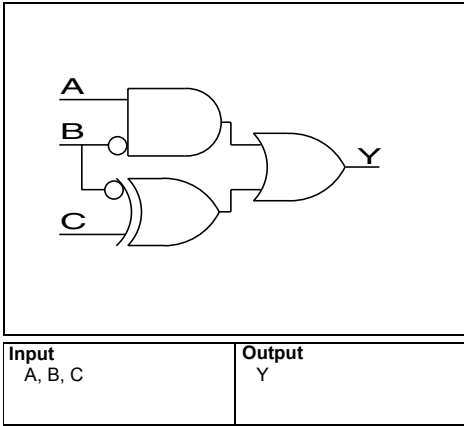
3-Input Combinatorial Gate

**Truth Table**

A	B	C	Y
0	0	0	1
1	0	0	1
0	1	0	1
1	1	0	0
0	0	1	0
1	0	1	0
0	1	1	1
1	1	1	1

Family	Tiles
All	1

**AXO6**



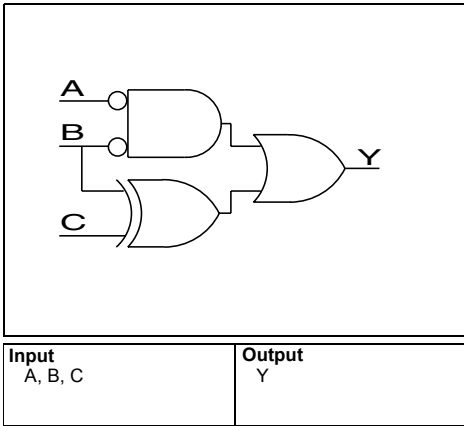
**Function**  
3-Input Combinatorial Gate

**Truth Table**

A	B	C	Y
0	0	0	1
1	0	0	1
0	1	0	0
1	1	0	0
0	0	1	0
1	0	1	1
0	1	1	1
1	1	1	1

Family	Tiles
All	1

**AXO7**



**Function**  
3-Input Combinatorial Gate

**Truth Table**

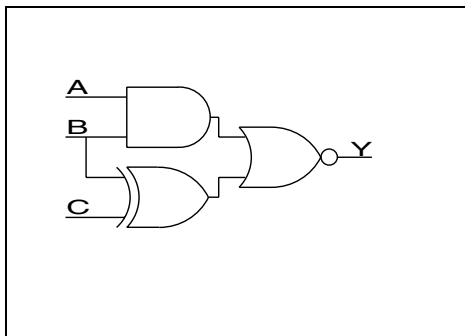
A	B	C	Y
0	0	0	1
1	0	0	0
0	1	0	1
1	1	0	1
0	0	1	1
1	0	1	1
0	1	1	0
1	1	1	0

Family	Tiles
All	1



## AXO11

ProASIC3, ProASIC3E



**Input**  
A, B, C

**Output**  
Y

**Function**

3-Input Combinatorial Gate

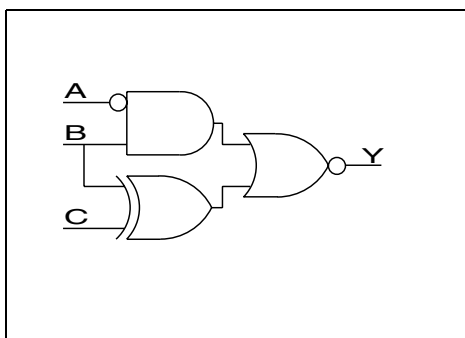
**Truth Table**

A	B	C	Y
0	0	0	1
1	0	0	1
0	1	0	0
1	1	0	0
0	0	1	0
1	0	1	0
0	1	1	1
1	1	1	0

Family	Tiles
All	1

## AXO12

ProASIC3, ProASIC3E



**Input**  
A, B, C

**Output**  
Y

**Function**

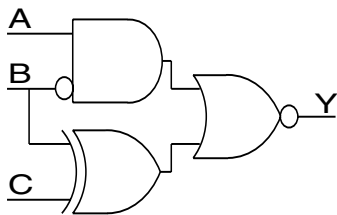
3-Input Combinatorial Gate

**Truth Table**

A	B	C	Y
0	0	0	1
1	0	0	1
0	1	0	0
1	1	0	0
0	0	1	0
1	0	1	0
0	1	1	0
1	1	1	1

Family	Tiles
All	1

## AXOI3



**Input**  
A, B, C

**Output**  
Y

**Function**

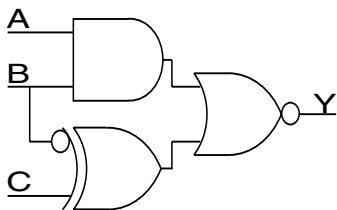
3-Input Combinatorial Gate

**Truth Table**

A	B	C	Y
0	0	0	1
1	0	0	0
0	1	0	0
1	1	0	0
0	0	1	0
1	0	1	0
0	1	1	1
1	1	1	1

Family	Tiles
All	1

## AXOI4



**Input**  
A, B, C

**Output**  
Y

**Function**

3-Input Combinatorial Gate

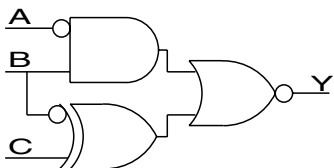
**Truth Table**

A	B	C	Y
0	0	0	0
1	0	0	0
0	1	0	1
1	1	0	0
0	0	1	1
1	0	1	1
0	1	1	0
1	1	1	0

Family	Tiles
All	1

## AXOI5

ProASIC3, ProASIC3E



Input

A, B, C

Output

Y

## Function

3-Input Combinatorial Gate

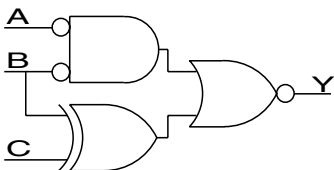
## Truth Table

A	B	C	Y
0	0	0	0
1	0	0	0
0	1	0	0
1	1	0	1
0	0	1	1
1	0	1	1
0	1	1	0
1	1	1	0

Family	Tiles
All	1

## AXOI7

ProASIC3, ProASIC3E



Input

A, B, C

Output

Y

## Function

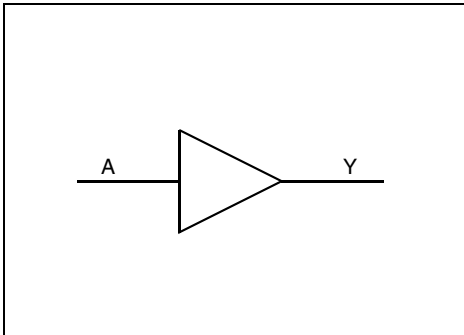
3-Input Combinatorial Gate

## Truth Table

A	B	C	Y
0	0	0	0
1	0	0	1
0	1	0	0
1	1	0	0
0	0	1	0
1	0	1	0
0	1	1	1
1	1	1	1

Family	Tiles
All	1

# BUFF



**Function**  
Buffer

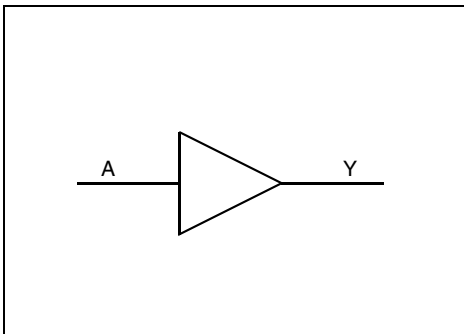
**Truth Table**

A	Y
0	0
1	1

<b>Input</b> A	<b>Output</b> Y
-------------------	--------------------

Family	Tiles
All	1

# BUFD



**Function**  
Buffer  
NOTE: The Combiner will not remove this macro

**Truth Table**

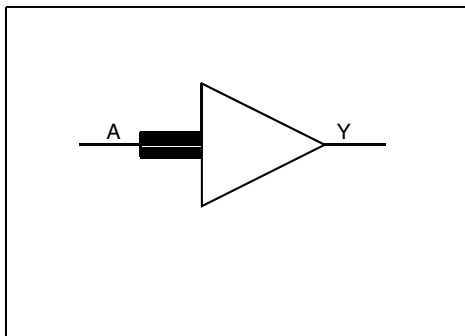
A	Y
0	0
1	1

<b>Input</b> A	<b>Output</b> Y
-------------------	--------------------

Family	Tiles
All	1

## CLKINT

ProASIC3, ProASIC3E

**Function**

Internal Clock Interface

**Truth Table**

A	Y
0	0
1	1

**Input**

A

**Output**

Y

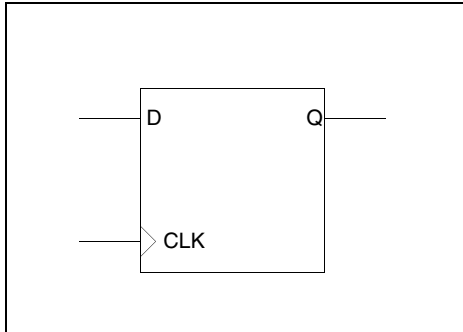
NOTE: CLKINT does not use any tiles. .

For more information on the Global Clock Network, refer to the latest Actel datasheet.



## DFN1

ProASIC3, ProASIC3E



## Function

D-Type Flip-Flop

## Truth Table

CLK	$Q_{n+1}$
↑	D

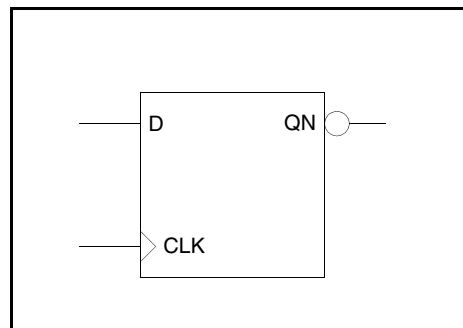
**Input**  
D, CLK

**Output**  
Q

Family	Tiles
All	1

## DFI1

ProASIC3, ProASIC3E



## Function

D-Type Flip-Flop with inverted Output

## Truth Table

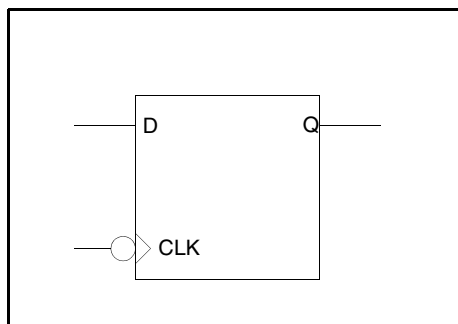
CLK	$QN_{n+1}$
↑	ID

**Input**  
D, CLK

**Output**  
QN

Family	Tiles
All	1

## DFN0



### Function

D-Type Flip-Flop with active low Clock

### Truth Table

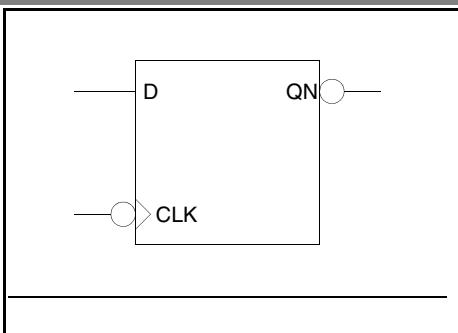
CLK	$Q_{n+1}$
↓	D

**Input**  
D, CLK

**Output**  
Q

Family	Tiles
All	1

## DFI0



### Function

D-Type Flip-Flop with active low Clock and inverted Output

### Truth Table

CLK	$QN_{n+1}$
↓	!D

**Input**  
D, CLK

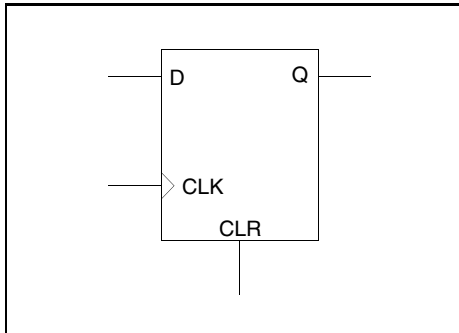
**Output**  
QN

Family	Tiles
All	1



## DFN1C1

ProASIC3, ProASIC3E

**Function**

D-Type Flip-Flop with active high Clear

**Truth Table**

CLR	CLK	$Q_{n+1}$
1	X	0
0	↑	D

**Input**

CLR, D, CLK

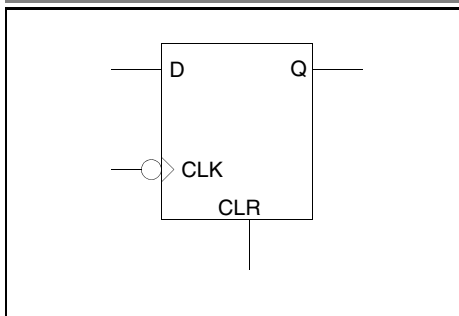
**Output**

Q

Family	Tiles
All	1

## DFN0C1

ProASIC3, ProASIC3E

**Function**

D-Type Flip-Flop with active high Clear and active low Clock

**Truth Table**

CLR	CLK	$Q_{n+1}$
1	X	0
0	↓	D

**Input**

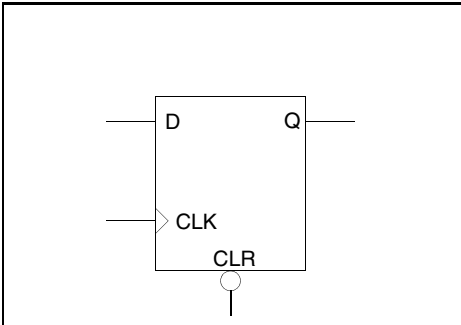
CLR, D, CLK

**Output**

Q

Family	Tiles
All	1

## DFN1C0



### Function

D-Type Flip-Flop with active low Clear

### Truth Table

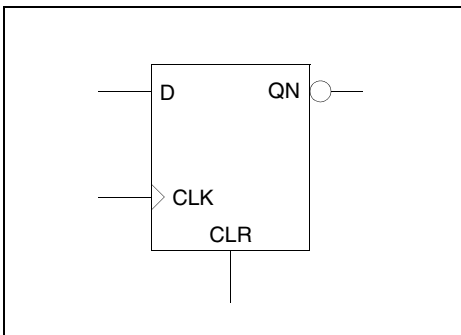
CLR	CLK	$Q_{n+1}$
0	X	0
1	↑	D

**Input**  
CLR, D, CLK

**Output**  
Q

Family	Tiles
All	1

## DFI1C1



### Function

D-Type Flip-Flop with active high Clear and Clock

### Truth Table

CLR	CLK	$QN_{n+1}$
1	X	1
0	↑	!D

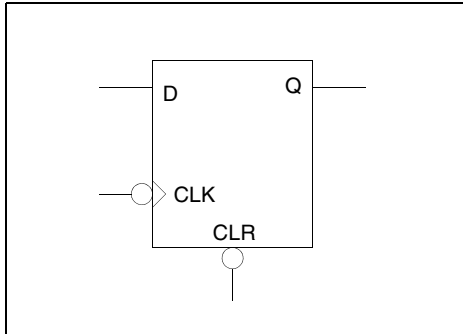
**Input**  
CLR, D, CLK

**Output**  
QN

Family	Tiles
All	1

## DFN0C0

ProASIC3, ProASIC3E

**Function**

D-Type Flip-Flop with active low Clear and Clock

**Truth Table**

CLR	CLK	$Q_{n+1}$
0	X	0
1	↓	D

**Input**

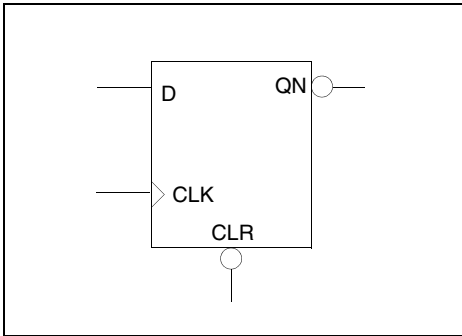
CLR, D, CLK

**Output**

Q

Family	Tiles
All	1

## DFI1C0



**Function**  
D-Type Flip-Flop with active low Clear and inverted Output

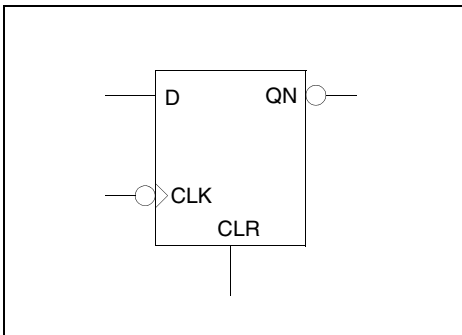
**Truth Table**

CLR	CLK	QN <sub>n+1</sub>
0	X	1
1	↑	!D

<b>Input</b> CLR, D, CLK	<b>Output</b> QN
-----------------------------	---------------------

Family	Tiles
All	1

## DFI0C1



**Function**  
D-Type Flip-Flop with active high Clear, active low Clock and inverted Output

**Truth Table**

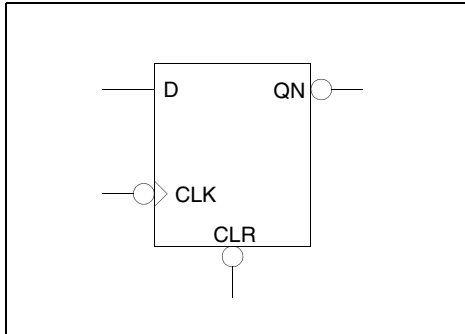
CLR	CLK	QN <sub>n+1</sub>
1	X	1
0	↓	!D

<b>Input</b> CLR, D, CLK	<b>Output</b> QN
-----------------------------	---------------------

Family	Tiles
All	1

## DFI0C0

ProASIC3, ProASIC3E



**Function**

D-Type Flip-Flop with active low Clear, Clock and inverted Output

**Truth Table**

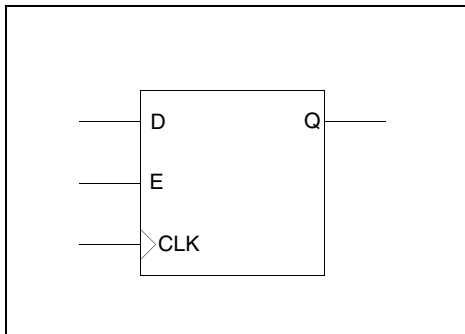
CLR	CLK	QN
0	X	1
1	↓	!D

<b>Input</b> CLR, D, CLK	<b>Output</b> QN
-----------------------------	---------------------

Family	Tiles
All	1

## DFN1E1

ProASIC3, ProASIC3E



**Function**

D-Type Flip-Flop with active high Enable

**Truth Table**

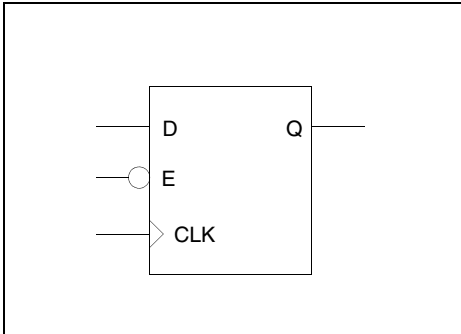
E	CLK	Q <sub>n+1</sub>
0	X	Q
1	↑	D

<b>Input</b> D, E, CLK	<b>Output</b> Q
---------------------------	--------------------

Family	Tiles
All	1

**DFN1E0**

ProASIC3, ProASIC3E


**Function**

D-Type Flip-Flop with active low Enable

**Truth Table**

E	CLK	$Q_{n+1}$
1	X	Q
0	↑	D

**Input**

D, E, CLK

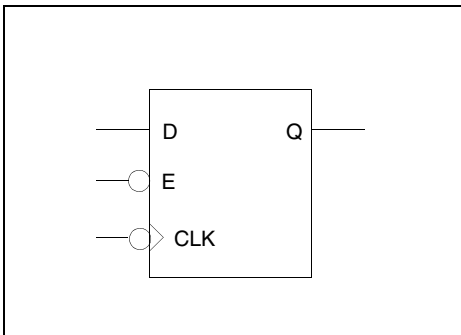
**Output**

Q

Family	Tiles
All	1

**DFN0E0**

ProASIC3, ProASIC3E


**Function**

D-Type Flip-Flop with active low Enable and Clock

**Truth Table**

E	CLK	$Q_{n+1}$
1	X	Q
0	↓	D

**Input**

D, E, CLK

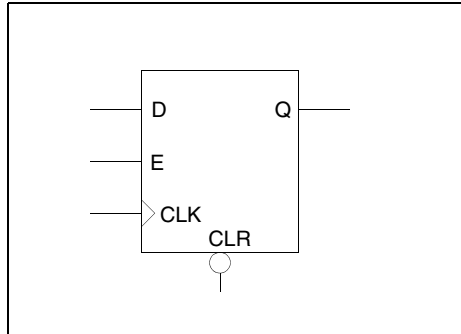
**Output**

Q

Family	Tiles
All	1

# DFN1E1C0

ProASIC3, ProASIC3E



**Function**  
D-Type Flip-Flop, with Enable and active low Clear

Truth Table

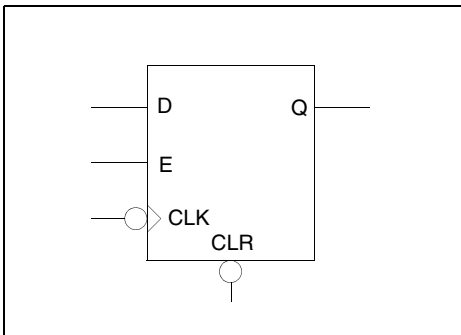
CLR	E	CLK	Q <sub>n+1</sub>
0	X	X	0
1	0	X	Q
1	1	↑	D

<b>Input</b> CLR, D, E, CLK	<b>Output</b> Q
--------------------------------	--------------------

Family	Tiles
All	1

# DFN0E1C0

ProASIC3, ProASIC3E



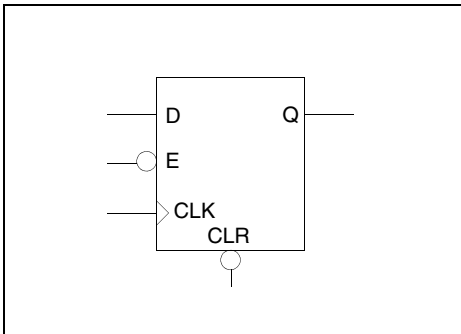
**Function**  
D-Type Flip-Flop with Enable and active low Clear and Clock

Truth Table

CLR	E	CLK	Q <sub>n+1</sub>
0	X	X	0
1	0	X	Q
1	1	↓	D

<b>Input</b> CLR, D, E, CLK	<b>Output</b> Q
--------------------------------	--------------------

Family	Tiles
All	1

**DFN1E0C0**

**Function**

D-Type Flip-Flop with Active Low Enable and Clear

**Truth Table**

CLR	E	CLK	$Q_{n+1}$
0	X	X	0
1	1	X	Q
1	0	↑	D

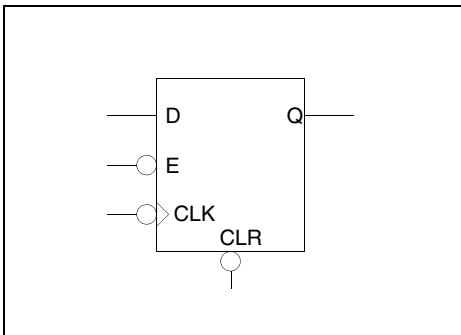
**Input**

CLR, D, E, CLK

**Output**

Q

Family	Tiles
All	1

**DFN0E0C0**

**Function**

D-Type Flip-Flop with active low Enable, Clear and Clock

**Truth Table**

CLR	E	CLK	$Q_{n+1}$
0	X	X	0
1	1	X	Q
1	0	↓	D

**Input**

CLR, D, E, CLK

**Output**

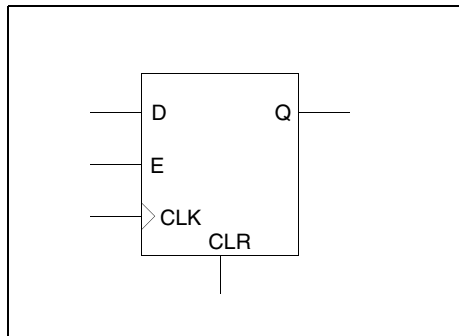
Q

Family	Tiles
All	1



## DFN1E1C1

ProASIC3, ProASIC3E

**Function**

D-Type Flip-Flop, with Enable and active high Clear

**Truth Table**

CLR	E	CLK	$Q_{n+1}$
1	X	X	0
0	0	X	Q
0	1	↑	D

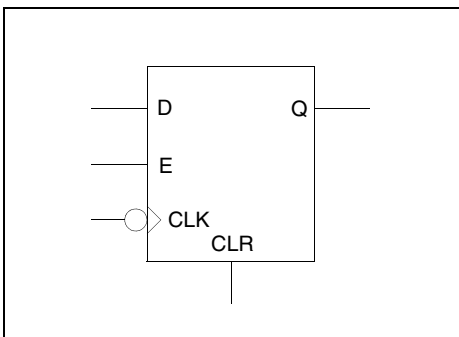
**Input**  
CLR, D, E, CLK

**Output**  
Q

Family	Tiles
All	1

## DFN0E1C1

ProASIC3, ProASIC3E

**Function**

D-Type Flip-Flop with Enable and active high Clear and active low Clock

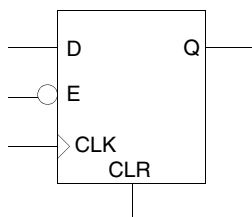
**Truth Table**

CLR	E	CLK	$Q_{n+1}$
1	X	X	0
0	0	X	Q
0	1	↓	D

**Input**  
CLR, D, E, CLK

**Output**  
Q

Family	Tiles
All	1

**DFN1E0C1**

**Function**

D-Type Flip-Flop with Active Low Enable and active high Clear

**Truth Table**

CLR	E	CLK	$Q_{n+1}$
1	X	X	0
0	1	X	Q
0	0	↑	D

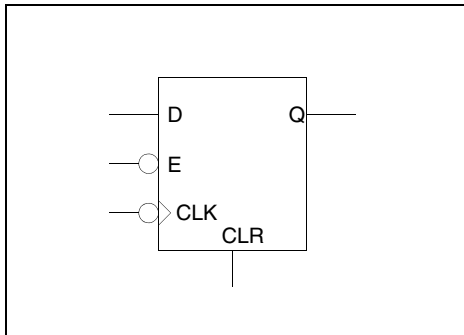
**Input**  
 CLR, D, E, CLK

**Output**  
 Q

Family	Tiles
All	1

## DFN0E0C1

ProASIC3, ProASIC3E

**Function**

D-Type Flip-Flop with active low Enable, Clock and active high Clear

**Truth Table**

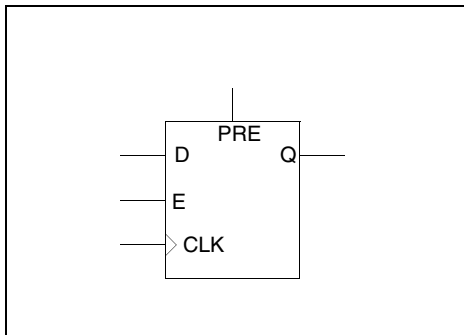
CLR	E	CLK	$Q_{n+1}$
1	X	X	0
0	1	X	Q
0	0	↓	D

**Input**  
CLR, D, E, CLK**Output**  
Q

Family	Tiles
All	1

## DFN1E1P1

ProASIC3, ProASIC3E

**Function**

D-Type Flip-Flop with active high Enable and Preset

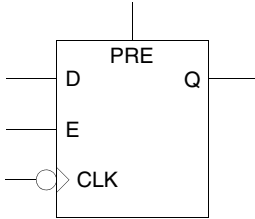
**Truth Table**

PRE	E	CLK	$Q_{n+1}$
1	X	X	1
0	0	X	Q
0	1	↑	D

**Input**  
D, E, PRE, CLK**Output**  
Q

Family	Tiles
All	1

## DFN0E1P1



### Function

D-Type Flip-Flop with active high Enable and Preset, and active low Clock

### Truth Table

PRE	E	CLK	$Q_{n+1}$
1	X	X	1
0	0	X	Q
0	1	↓	D

### Input

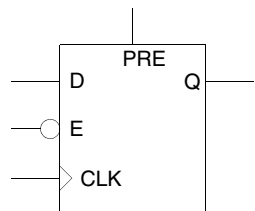
D, E, PRE, CLK

### Output

Q

Family	Tiles
All	1

## DFN1E0P1



### Function

D-Type Flip-Flop with active low Enable, and active high Preset

### Truth Table

PRE	E	CLK	$Q_{n+1}$
1	X	X	1
0	1	X	Q
0	0	↑	D

### Input

D, E, PRE, CLK

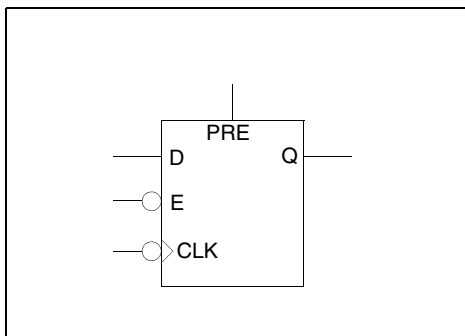
### Output

Q

Family	Tiles
All	1

## DFN0E0P1

ProASIC3, ProASIC3E

**Function**

D-Type Flip-Flop with active low Enable and Clock, and active high Preset

**Truth Table**

PRE	E	CLK	$Q_{n+1}$
1	X	X	1
0	1	X	Q
0	0	↓	D

**Input**

D, E, PRE, CLK

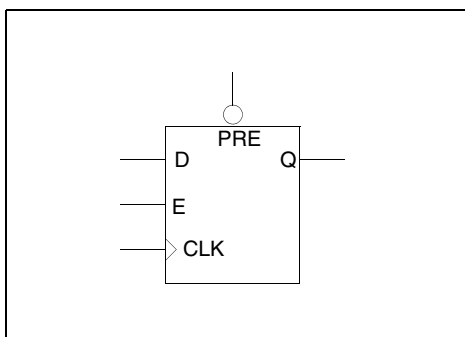
**Output**

Q

Family	Tiles
All	1

## DFN1E1P0

ProASIC3, ProASIC3E

**Function**

D-Type Flip-Flop with active high Enable and active low Preset

**Truth Table**

PRE	E	CLK	$Q_{n+1}$
0	X	X	1
1	0	X	Q
1	1	↑	D

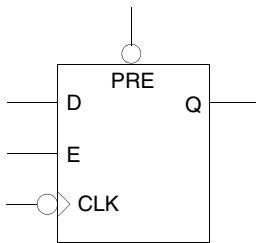
**Input**

D, E, PRE, CLK

**Output**

Q

Family	Tiles
All	1

**DFN0E1P0**

**Function**

D-Type Flip-Flop with active high Enable and active low Preset and Clock

**Truth Table**

PRE	E	CLK	$Q_{n+1}$
0	X	X	1
1	0	X	Q
1	1	↓	D

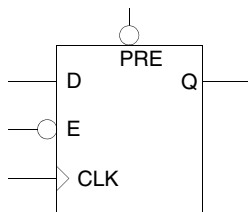
**Input**

D, E, PRE, CLK

**Output**

Q

Family	Tiles
All	1

**DFN1E0P0**

**Function**

D-Type Flip-Flop with active low Enable and Preset

**Truth Table**

PRE	E	CLK	$Q_{n+1}$
0	X	X	1
1	1	X	Q
1	0	↑	D

**Input**

D, E, PRE, CLK

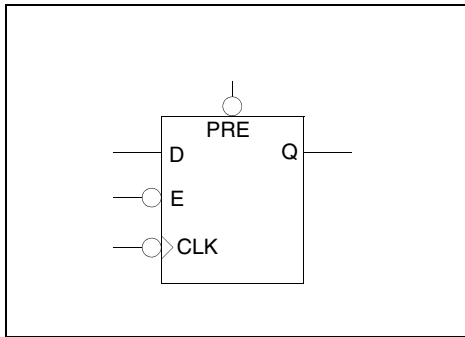
**Output**

Q

Family	Tiles
All	1

## DFN0E0P0

ProASIC3, ProASIC3E



### Function

D-Type Flip-Flop with active low Enable, Clock, and Preset

### Truth Table

PRE	E	CLK	$Q_{n+1}$
0	X	X	1
1	1	X	Q
1	0	↓	D

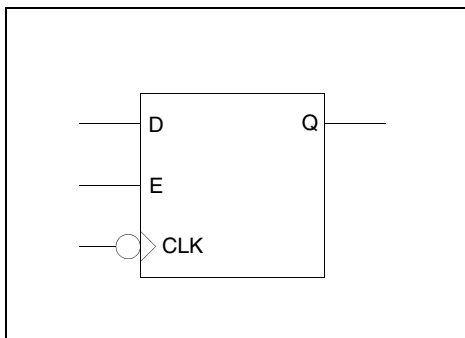
**Input**  
D, E, PRE, CLK

**Output**  
Q

Family	Tiles
All	1

## DFN0E1

ProASIC3, ProASIC3E



### Function

D-Type Flip-Flop, with Enable, and active low Clock

### Truth Table

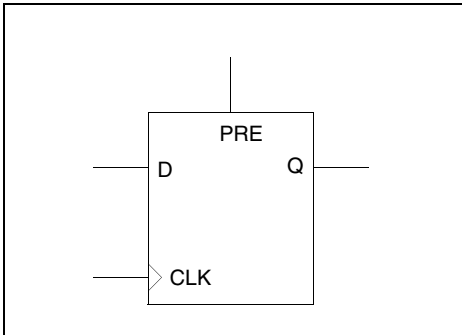
E	CLK	$Q_{n+1}$
0	X	Q
1	↓	D

**Input**  
D, E, CLK

**Output**  
Q

Family	Tiles
All	1

## DFN1P1



**Function**  
D-Type Flip-Flop with active high Preset

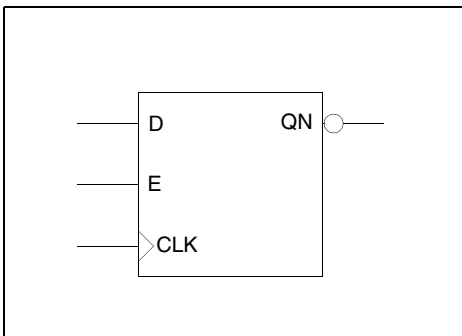
**Truth Table**

PRE	CLK	$Q_{n+1}$
1	X	1
0	↑	D

<b>Input</b> D, PRE, CLK	<b>Output</b> Q
-----------------------------	--------------------

Family	Tiles
All	1

## DFI1E1



**Function**  
D-Type Flip-Flop with active high Enable and inverted output

**Truth Table**

E	CLK	$QN_{n+1}$
0	X	QN
1	↑	!D

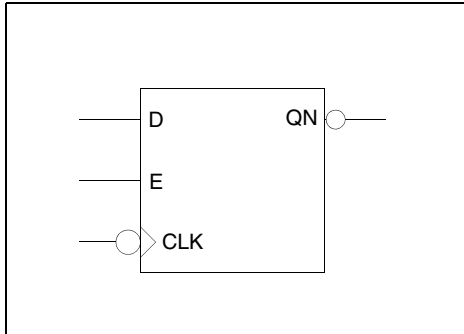
<b>Input</b> D, E, CLK	<b>Output</b> QN
---------------------------	---------------------

Family	Tiles
All	1



## DFI0E1

ProASIC3, ProASIC3E



### Function

D-Type Flip-Flop, with Enable, and active low Clock and inverted output

### Truth Table

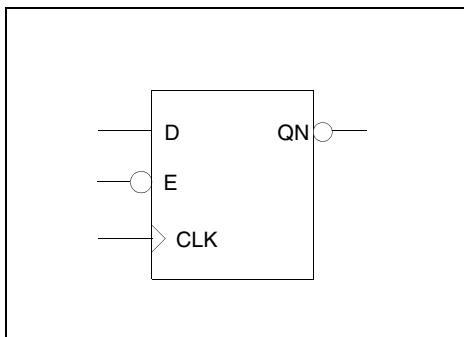
E	CLK	QN <sub>n+1</sub>
0	X	QN
1	↓	!D

<b>Input</b> D, E, CLK	<b>Output</b> QN
---------------------------	---------------------

Family	Tiles
All	1

## DFI1E0

ProASIC3, ProASIC3E



### Function

D-Type Flip-Flop with active low Enable and inverted output

### Truth Table

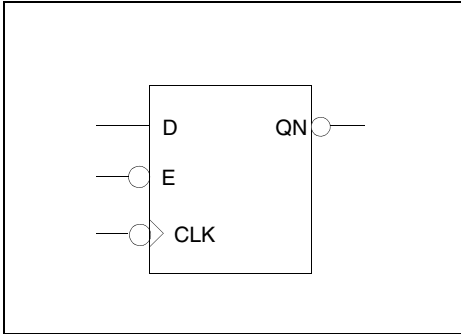
E	CLK	QN <sub>n+1</sub>
1	X	QN
0	↑	!D

<b>Input</b> D, E, CLK	<b>Output</b> QN
---------------------------	---------------------

Family	Tiles
All	1

**DF10E0**

ProASIC3, ProASIC3E


**Function**

D-Type Flip-Flop with active low Enable and Clock and inverted output

**Truth Table**

E	CLK	Q <sub>n+1</sub>
1	X	QN
0	↓	!D

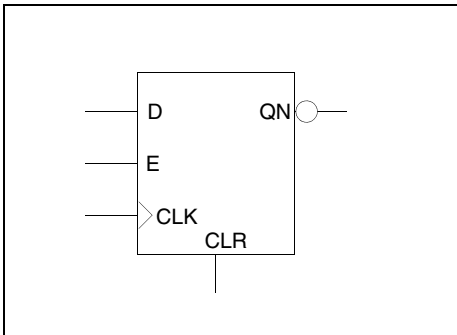
**Input**  
D, E, CLK

**Output**  
QN

Family	Tiles
All	1

**DF11E1C1**

ProASIC3, ProASIC3E


**Function**

D-Type Flip-Flop, with Enable and active high Clear and inverted output

**Truth Table**

CLR	E	CLK	Q <sub>n+1</sub>
1	X	X	1
0	0	X	QN
0	1	↑	!D

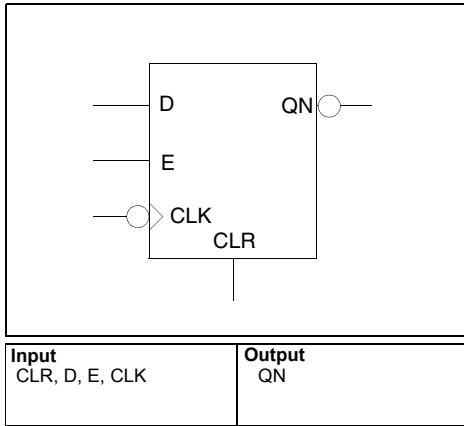
**Input**  
CLR, D, E, CLK

**Output**  
QN

Family	Tiles
All	1

## DFI0E1C1

ProASIC3, ProASIC3E



### Function

D-Type Flip-Flop with Enable and active high Clear and active low Clock and inverted output

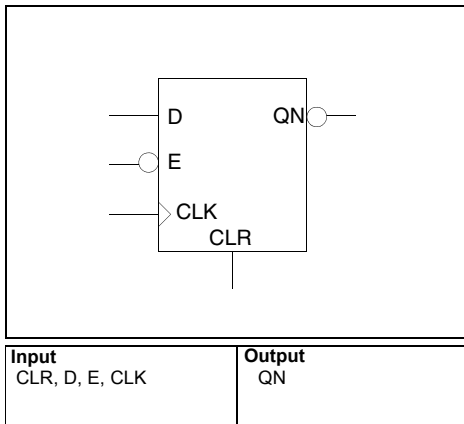
### Truth Table

CLR	E	CLK	QN <sub>n+1</sub>
1	X	X	1
0	0	X	QN
0	1	↓	!D

Family	Tiles
All	1

## DFI1E0C1

ProASIC3, ProASIC3E



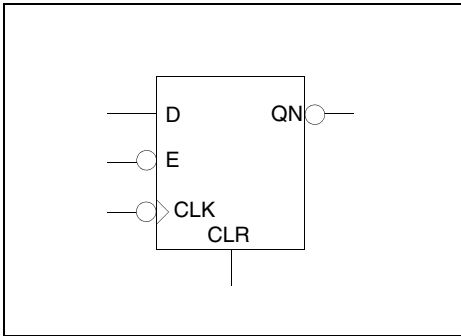
### Function

D-Type Flip-Flop with Active Low Enable and active high Clear and inverted output

### Truth Table

CLR	E	CLK	QN <sub>n+1</sub>
1	X	X	1
0	1	X	QN
0	0	↑	!D

Family	Tiles
All	1

**DFI0E0C1**


**Input**  
CLR, D, E, CLK

**Output**  
QN

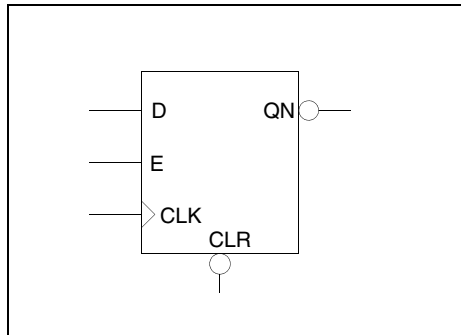
**Function**

D-Type Flip-Flop with active low Enable, Clock, active high Clear, and inverted output

**Truth Table**

CLR	E	CLK	QN <sub>n+1</sub>
1	X	X	1
0	1	X	QN
0	0	↓	!D

Family	Tiles
All	1

**DFI1E1C0**


**Input**  
CLR, D, E, CLK

**Output**  
QN

**Function**

D-Type Flip-Flop, with Enable and active low Clear and inverted output

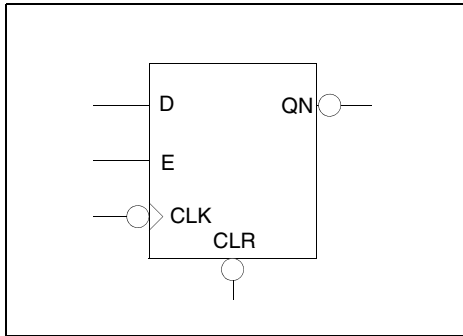
**Truth Table**

CLR	E	CLK	QN <sub>n+1</sub>
0	X	X	1
1	0	X	QN
1	1	↑	!D

Family	Tiles
All	1

**DFI0E1C0**

ProASIC3, ProASIC3E


**Function**

D-Type Flip-Flop with Enable and active low Clear and Clock and inverted output

**Truth Table**

CLR	E	CLK	Q <sub>n+1</sub>
0	X	X	1
1	0	X	QN
1	1	↓	!D

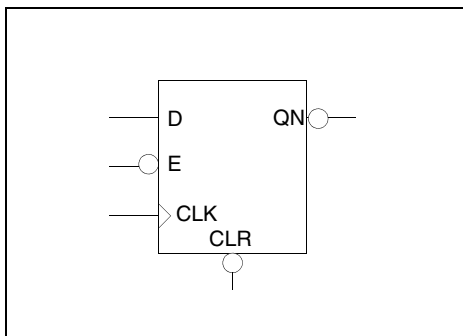
**Input**  
CLR, D, E, CLK

**Output**  
QN

Family	Tiles
All	1

**DFI1E0C0**

ProASIC3, ProASIC3E


**Function**

D-Type Flip-Flop with Active Low Enable and Clear and inverted output

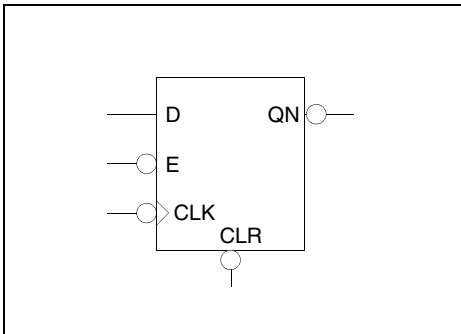
**Truth Table**

CLR	E	CLK	Q <sub>n+1</sub>
0	X	X	1
1	1	X	QN
1	0	↑	!D

**Input**  
CLR, D, E, CLK

**Output**  
QN

Family	Tiles
All	1

**DFI0E0C0**

**Function**

D-Type Flip-Flop with active low Enable, Clear, Clock and inverted output

**Truth Table**

CLR	E	CLK	QN <sub>n+1</sub>
0	X	X	1
1	1	X	QN
1	0	↓	!D

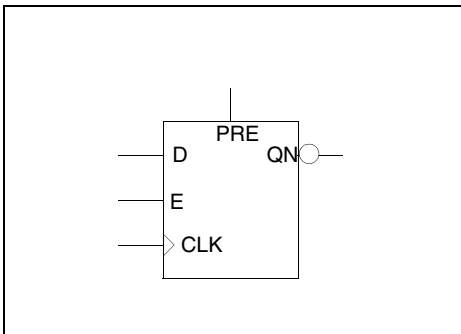
**Input**

CLR, D, E, CLK

**Output**

QN

Family	Tiles
All	1

**DFI1E1P1**

**Function**

D-Type Flip-Flop with active high Enable and Preset and inverted output

**Truth Table**

PRE	E	CLK	QN <sub>n+1</sub>
1	X	X	0
0	0	X	QN
0	1	↑	!D

**Input**

D, E, PRE, CLK

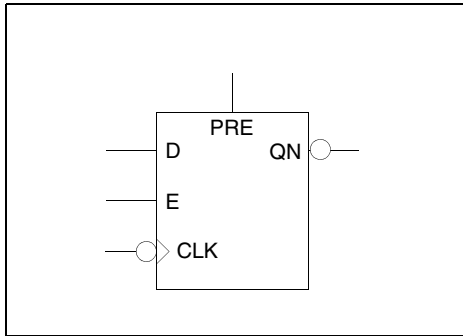
**Output**

QN

Family	Tiles
All	1

**DFI0E1P1**

ProASIC3, ProASIC3E


**Function**

D-Type Flip-Flop with active high Enable and Preset, active low Clock and inverted output

**Truth Table**

PRE	E	CLK	QN <sub>n+1</sub>
1	X	X	0
0	0	X	QN
0	1	↓	!D

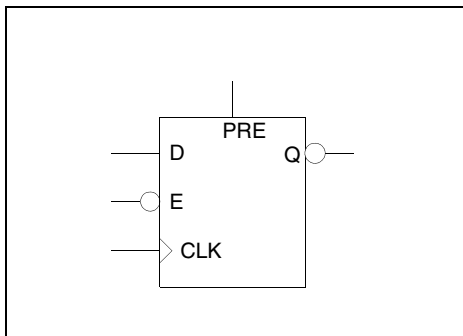
**Input**  
 D, E, PRE, CLK

**Output**  
 QN

Family	Tiles
All	1

**DFI1E0P1**

ProASIC3, ProASIC3E


**Function**

D-Type Flip-Flop with active low Enable, active high Preset and inverted output

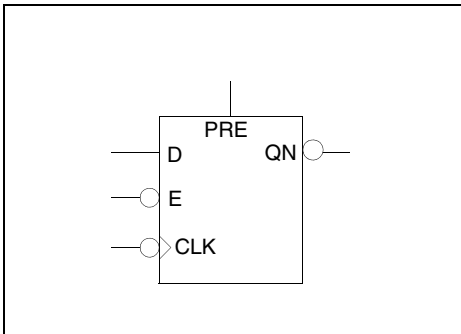
**Truth Table**

PRE	E	CLK	QN <sub>n+1</sub>
1	X	X	0
0	1	X	QN
0	0	↑	!D

**Input**  
 D, E, PRE, CLK

**Output**  
 QN

Family	Tiles
All	1

**DFI0E0P1**


**Input**  
D, E, PRE, CLK

**Output**  
QN

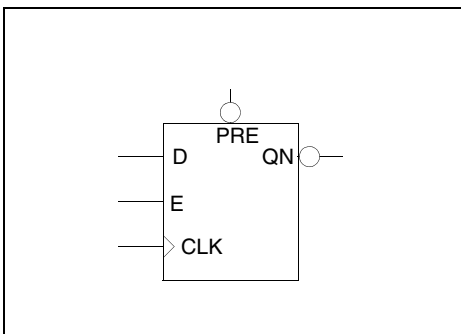
**Function**

D-Type Flip-Flop with active low Enable and Clock, active high Preset and inverted output

**Truth Table**

PRE	E	CLK	QN <sub>n+1</sub>
1	X	X	0
0	1	X	QN
0	0	↓	!D

Family	Tiles
All	1

**DFI1E1P0**


**Input**  
D, E, PRE, CLK

**Output**  
QN

**Function**

D-Type Flip-Flop with active high Enable, active low Preset, and inverted output

**Truth Table**

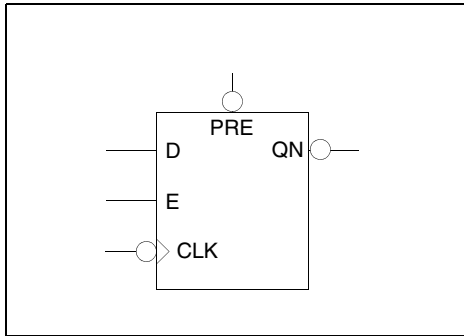
PRE	E	CLK	QN <sub>n+1</sub>
0	X	X	0
1	0	X	QN
1	1	↑	!D

Family	Tiles
All	1



**DFI0E1P0**

ProASIC3, ProASIC3E


**Function**

D-Type Flip-Flop with active high Enable, active low Preset and Clock, and inverted output

**Truth Table**

PRE	E	CLK	QN <sub>n+1</sub>
0	X	X	0
1	0	X	QN
1	1	↓	!D

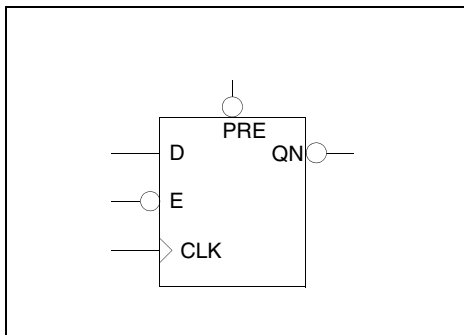
**Input**  
 D, E, PRE, CLK

**Output**  
 QN

Family	Tiles
All	1

**DFI1E0P0**

ProASIC3, ProASIC3E


**Function**

D-Type Flip-Flop with active low Enable and Preset, and inverted output

**Truth Table**

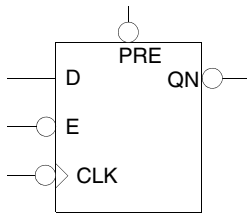
PRE	E	CLK	QN <sub>n+1</sub>
0	X	X	0
1	1	X	QN
1	0	↑	!D

**Input**  
 D, E, PRE, CLK

**Output**  
 QN

Family	Tiles
All	1

## DFI0E0P0



### Function

D-Type Flip-Flop with active low Enable, Clock, and Preset, and inverted output

### Truth Table

PRE	E	CLK	QN <sub>n+1</sub>
0	X	X	0
1	1	X	QN
1	0	↓	!D

### Input

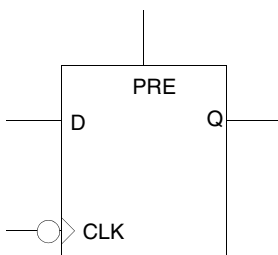
D, E, PRE, CLK

### Output

QN

Family	Tiles
All	1

## DFN0P1



### Function

D-Type Flip-Flop with active high Preset, and active low Clock

### Truth Table

PRE	CLK	Q <sub>n+1</sub>
1	X	1
0	↓	D

### Input

D, PRE, CLK

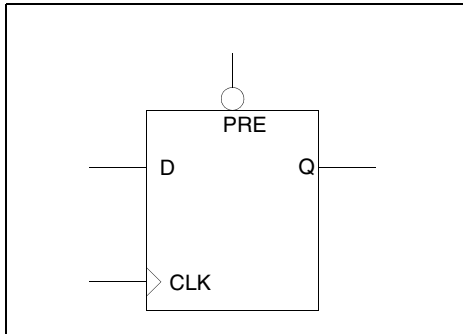
### Output

Q

Family	Tiles
All	1

## DFN1P0

ProASIC3, ProASIC3E

**Function**

D-Type Flip-Flop with active low Preset

**Truth Table**

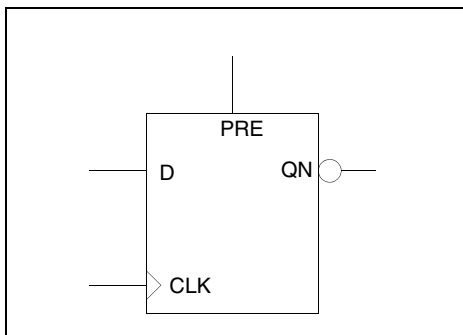
PRE	CLK	$Q_{n+1}$
0	X	1
1	↑	D

**Input**  
D, PRE, CLK**Output**  
Q

Family	Tiles
All	1

## DFI1P1

ProASIC3, ProASIC3E

**Function**

D-Type Flip-Flop with active high Preset, and inverted Output

**Truth Table**

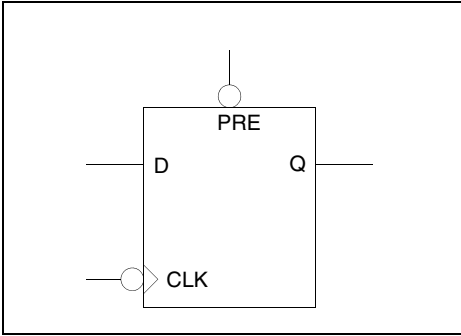
PRE	CLK	$QN_{n+1}$
1	X	0
0	↑	!D

**Input**  
D, PRE, CLK**Output**  
QN

Family	Tiles
All	1

## DFN0P0

ProASIC3, ProASIC3E



**Function**

D-Type Flip-Flop with active low Preset and Clock

**Truth Table**

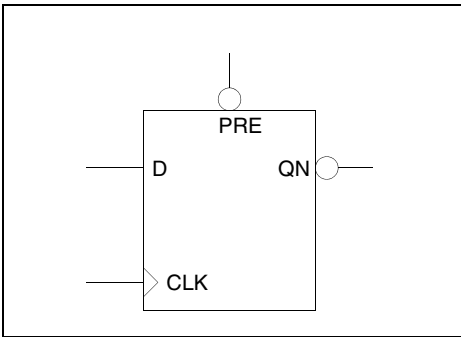
PRE	CLK	$Q_{n+1}$
0	X	1
1	↓	D

<b>Input</b> D, PRE, CLK	<b>Output</b> Q
-----------------------------	--------------------

Family	Tiles
All	1

## DFI1P0

ProASIC3, ProASIC3E



**Function**

D-Type Flip-Flop with active low Preset and inverted Output

**Truth Table**

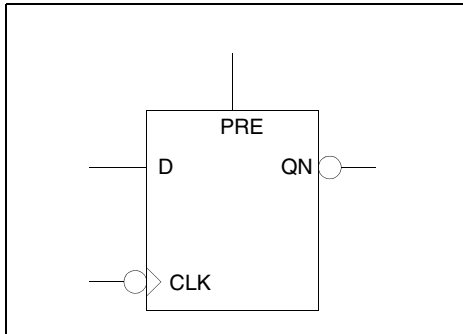
PRE	CLK	$QN_{n+1}$
0	X	0
1	↑	!D

<b>Input</b> D, PRE, CLK	<b>Output</b> QN
-----------------------------	---------------------

Family	Tiles
All	1

## DFI0P1

ProASIC3, ProASIC3E

**Function**

D-Type Flip-Flop with active high Preset, and active low Clock and inverted Output

**Truth Table**

PRE	CLK	QN <sub>n+1</sub>
1	X	0
0	↓	!D

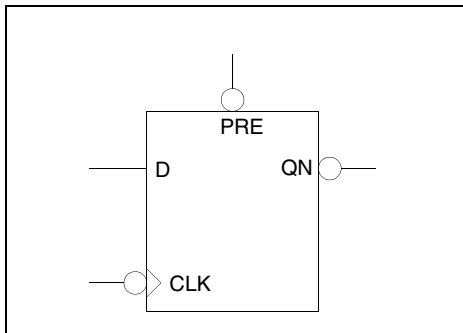
**Input**  
D, PRE, CLK

**Output**  
QN

Family	Tiles
All	1

## DFI0P0

ProASIC3, ProASIC3E

**Function**

D-Type Flip-Flop with active low Preset, Clock and inverted Output

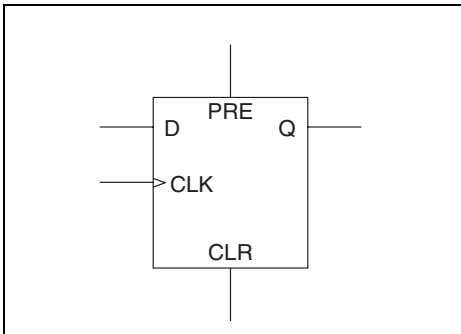
**Truth Table**

PRE	CLK	QN <sub>n+1</sub>
0	X	0
1	↓	!D

**Input**  
D, PRE, CLK

**Output**  
QN

Family	Tiles
All	1

**DFN1P1C1**

**Function**

Rising Edge Triggered D-Type Flip-Flop with Active High Preset and Clear

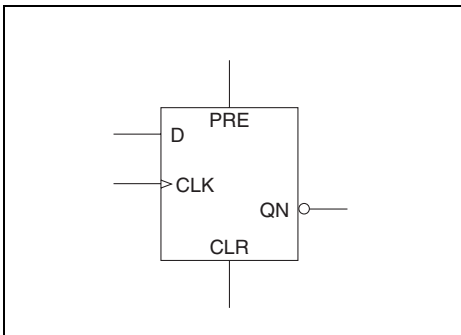
**Truth Table**

CLK	PRE	CLR	$Q_{n+1}$
X	1	0	1
X	X	1	0
↑	0	0	D

**Input**  
CLR, PRE, CLK, D

**Output**  
Q

Family	Tiles
All	1

**DFI1P1C1**

**Function**

Rising Edge Triggered D-Type Flip-Flop with Active High Preset and Clear and inverted Output

**Truth Table**

CLK	PRE	CLR	$QN_{n+1}$
X	1	0	0
X	X	1	1
↑	0	0	!D

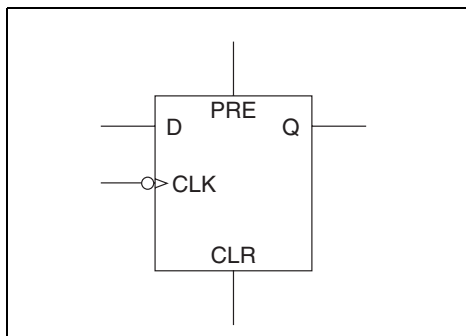
**Input**  
CLR, PRE, CLK, D

**Output**  
QN

Family	Tiles
All	1

**DFN0P1C1**

ProASIC3, ProASIC3E


**Function**

Falling Edge Triggered D-Type Flip-Flop with Active High Preset and Clear

**Truth Table**

CLK	PRE	CLR	$Q_{n+1}$
X	1	0	1
X	X	1	0
↓	0	0	D

**Input**

CLR, PRE, CLK, D

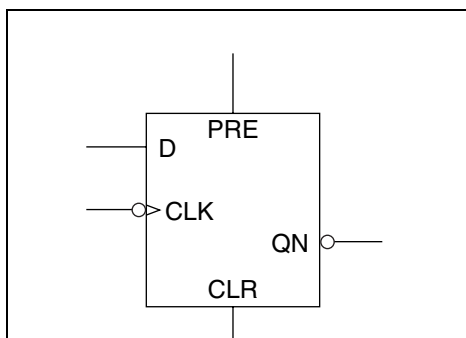
**Output**

Q

Family	Tiles
All	1

**DFI0P1C1**

ProASIC3, ProASIC3E


**Function**

Falling Edge Triggered D-Type Flip-Flop with Active High Preset and Clear and inverted Output

**Truth Table**

CLK	PRE	CLR	$QN_{n+1}$
X	1	0	0
X	X	1	1
↓	0	0	!D

**Input**

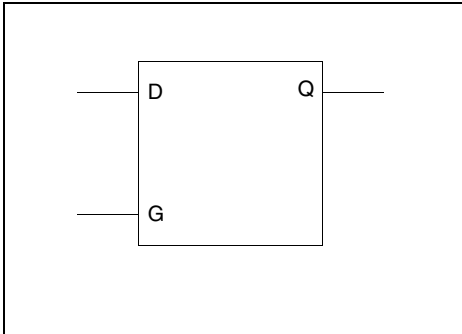
CLR, PRE, CLK, D

**Output**

QN

Family	Tiles
All	1

# DLN1



**Function**  
Data Latch

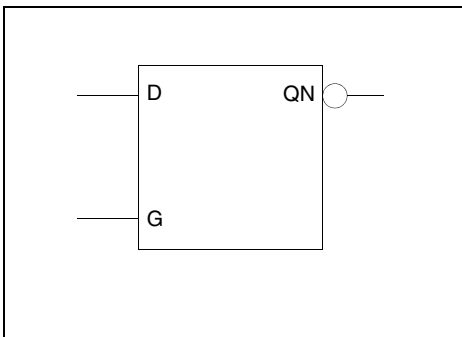
**Truth Table**

G	Q <sub>n+1</sub>
0	Q
1	D

<b>Input</b> D, G	<b>Output</b> Q
----------------------	--------------------

Family	Tiles
All	1

# DLI1



**Function**  
Data Latch with inverted Output

**Truth Table**

G	QN <sub>n+1</sub>
0	QN
1	!D

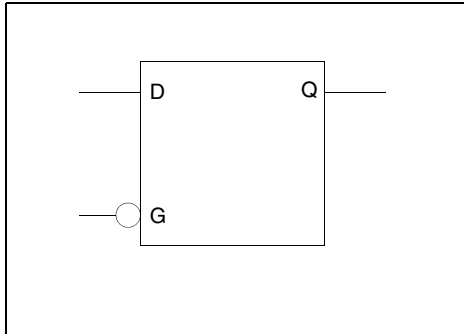
<b>Input</b> D, G	<b>Output</b> QN
----------------------	---------------------

Family	Tiles
All	1



## DLN0

ProASIC3, ProASIC3E

**Function**

Data Latch with active low Clock

**Truth Table**

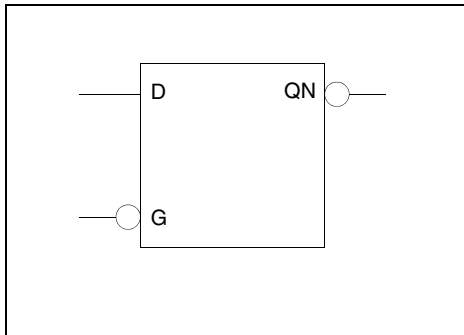
G	$Q_{n+1}$
1	Q
0	D

**Input**  
D, G**Output**  
Q

Family	Tiles
All	1

## DLI0

ProASIC3, ProASIC3E

**Function**

Data Latch, with active low Clock and inverted Output

**Truth Table**

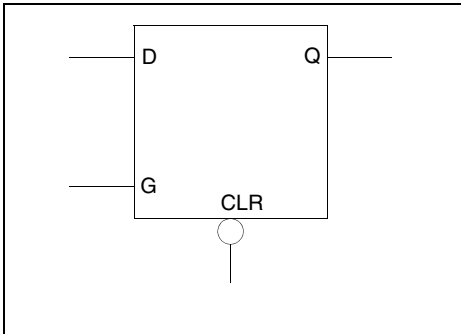
G	$QN_{n+1}$
1	QN
0	ID

**Input**  
D, G**Output**  
QN

Family	Tiles
All	1

**DLN1C0**

ProASIC3, ProASIC3E


**Function**

Data Latch with active low Clear

**Truth Table**

CLR	G	$Q_{n+1}$
0	X	0
1	0	Q
1	1	D

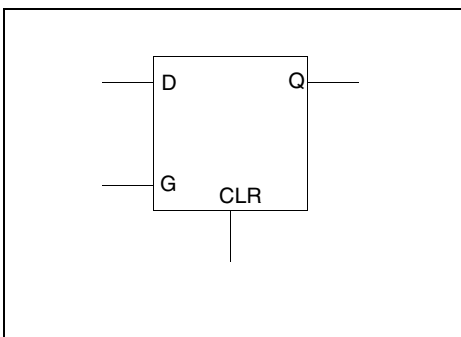
**Input**  
CLR, D, G

**Output**  
Q

Family	Tiles
All	1

**DLN1C1**

ProASIC3, ProASIC3E


**Function**

Data Latch with active high Clear

**Truth Table**

CLR	G	$Q_{n+1}$
1	X	0
0	0	Q
0	1	D

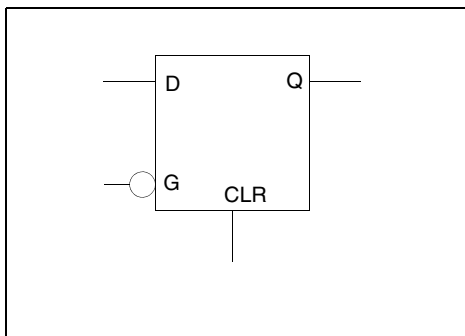
**Input**  
CLR, D, G

**Output**  
Q

Family	Tiles
All	1

## DLN0C1

ProASIC3, ProASIC3E

**Function**

Data Latch with active high Clear and active low Clock

**Truth Table**

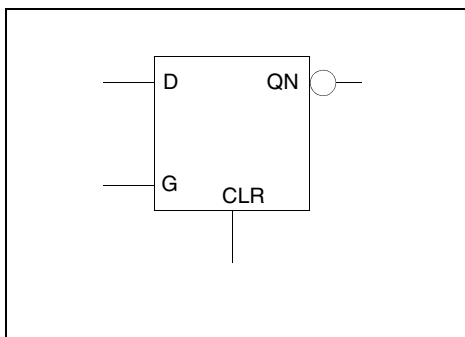
CLR	G	$Q_{n+1}$
1	X	0
0	1	Q
0	0	D

**Input**  
CLR, D, G**Output**  
Q

Family	Tiles
All	1

## DL1C1

ProASIC3, ProASIC3E

**Function**

Data Latch with active high Clear and inverted Output

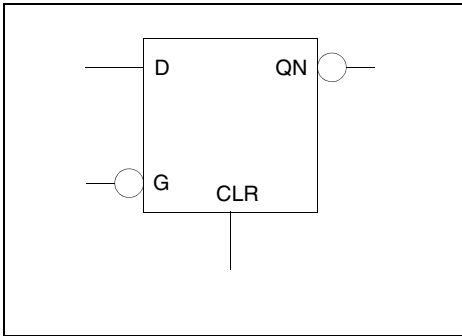
**Truth Table**

CLR	G	$QN_{n+1}$
1	X	1
0	0	QN
0	1	!D

**Input**  
CLR, D, G**Output**  
QN

Family	Tiles
All	1

# DLI0C1



**Function**  
Data Latch with active high Clear and active low Clock and inverted Output

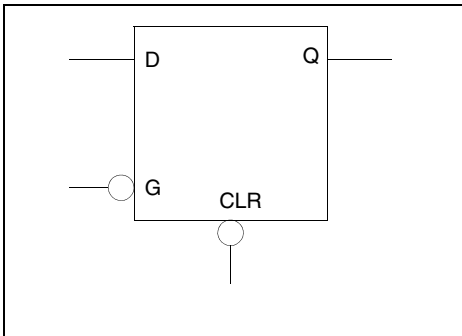
**Truth Table**

CLR	G	QN <sub>n+1</sub>
1	X	1
0	1	QN
0	0	!D

<b>Input</b> CLR, D, G	<b>Output</b> QN
---------------------------	---------------------

Family	Tiles
All	1

# DLN0C0



**Function**  
Data Latch with active low Clear and Clock

**Truth Table**

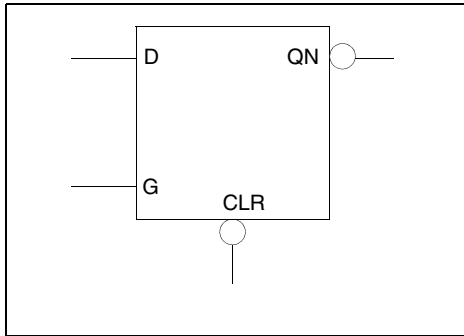
CLR	G	Q <sub>n+1</sub>
0	X	0
1	1	Q
1	0	D

<b>Input</b> CLR, D, G	<b>Output</b> Q
---------------------------	--------------------

Family	Tiles
All	1

## DL1C0

ProASIC3, ProASIC3E

**Function**

Data Latch with active low Clear and inverted output

**Truth Table**

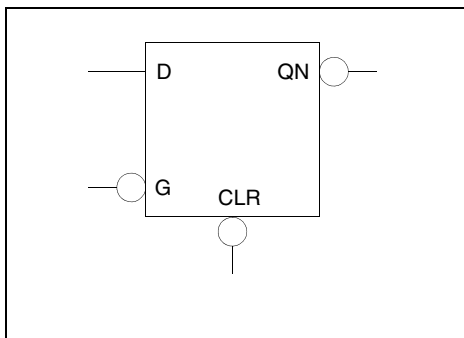
CLR	G	QN <sub>n+1</sub>
0	X	1
1	0	QN
1	1	!D

**Input**  
CLR, D, G**Output**  
QN

Family	Tiles
All	1

## DL10C0

ProASIC3, ProASIC3E

**Function**

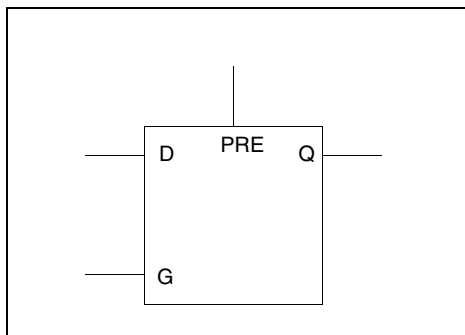
Data Latch with active low Clear, Clock, and inverted Output

**Truth Table**

CLR	G	QN <sub>n+1</sub>
0	X	1
1	1	QN
1	0	!D

**Input**  
CLR, D, G**Output**  
QN

Family	Tiles
All	1

**DLN1P1**

**Function**

Data Latch with active high Preset and Clock

**Truth Table**

PRE	G	$Q_{n+1}$
1	X	1
0	0	Q
0	1	D

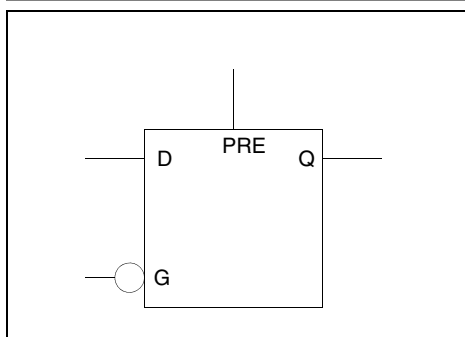
**Input**

D, G, PRE

**Output**

Q

Family	Tiles
All	1

**DLN0P1**

**Function**

Data Latch with active high Preset and active low Clock

**Truth Table**

PRE	G	$Q_{n+1}$
1	X	1
0	1	Q
0	0	D

**Input**

D, G, PRE

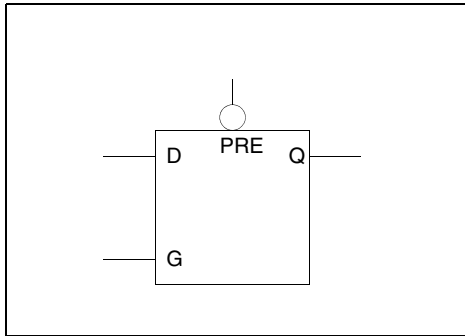
**Output**

Q

Family	Tiles
All	1

## DLN1P0

ProASIC3, ProASIC3E

**Function**

Data Latch with active low Preset and active high Clock

**Truth Table**

PRE	G	$Q_{n+1}$
0	X	1
1	0	Q
1	1	D

**Input**

D, G, PRE

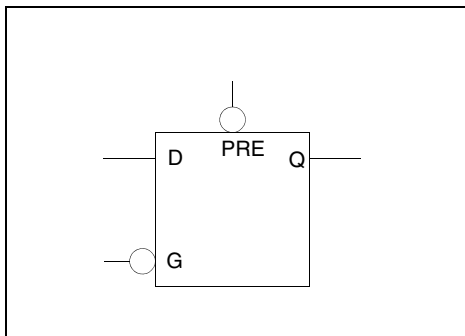
**Output**

Q

Family	Tiles
All	1

## DLN0P0

ProASIC3, ProASIC3E

**Function**

Data Latch with active low Preset and Clock

**Truth Table**

PRE	G	$Q_{n+1}$
0	X	1
1	1	Q
1	0	D

**Input**

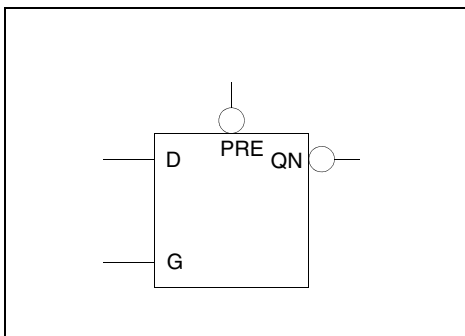
D, G, PRE

**Output**

Q

Family	Tiles
All	1

## DLI1P0



**Function**  
Data Latch with active low Preset and Output, and active high Clock

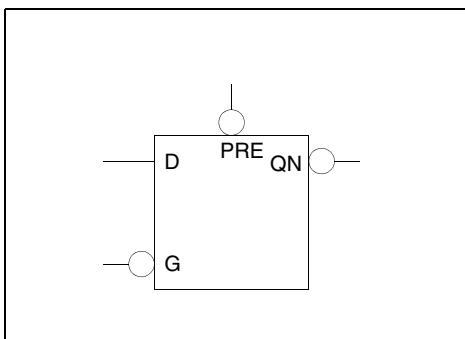
**Truth Table**

PRE	G	QN <sub>n+1</sub>
0	X	0
1	0	QN
1	1	!D

<b>Input</b> D, G, PRE	<b>Output</b> QN
---------------------------	---------------------

Family	Tiles
All	1

## DLI0P0



**Function**  
Data Latch with active low Preset, Clock, and inverted Output

**Truth Table**

PRE	G	QN <sub>n+1</sub>
0	X	0
1	0	!D
1	1	QN

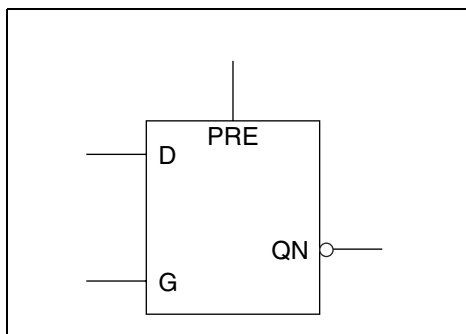
<b>Input</b> D, G, PRE	<b>Output</b> QN
---------------------------	---------------------

Family	Tiles
All	1



## DLI1P1

ProASIC3, ProASIC3E

**Function**

Active High Latch with Active High Preset and inverted Output

**Truth Table**

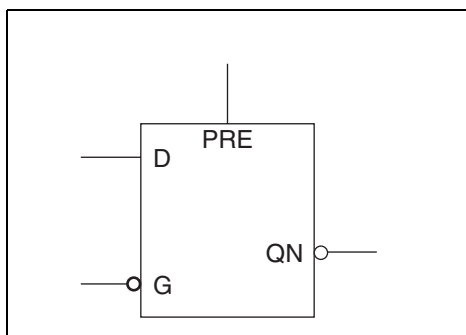
<b>G</b>	<b>PRE</b>	<b>QN<sub>n+1</sub></b>
X	1	0
0	0	QN
1	0	!D

**Input**  
PRE, G, D**Output**  
QN

<b>Family</b>	<b>Tiles</b>
All	1

## DLI0P1

ProASIC3, ProASIC3E

**Function**

Active Low Latch with Active High Preset and inverted Output

**Truth Table**

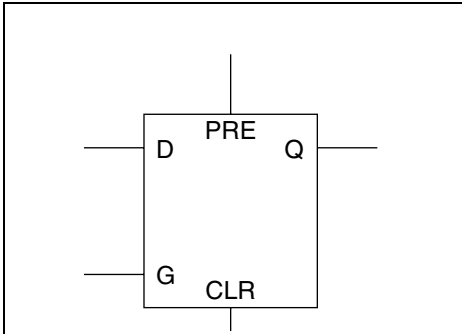
<b>G</b>	<b>PRE</b>	<b>QN<sub>n+1</sub></b>
X	1	0
0	0	!D
1	0	QN

**Input**  
PRE, G, D**Output**  
QN

<b>Family</b>	<b>Tiles</b>
All	1

# DLN1P1C1

ProASIC3, ProASIC3E



**Function**  
Active High Latch with Active High Preset and Clear

**Truth Table**

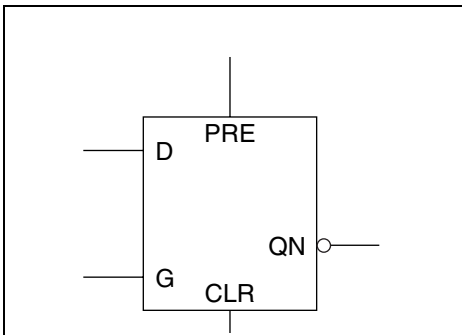
G	PRE	CLR	$Q_{n+1}$
X	1	0	1
X	X	1	0
1	0	0	D
0	0	0	Q

<b>Input</b> CLR, PRE, G, D	<b>Output</b> Q
--------------------------------	--------------------

Family	Tiles
All	1

# DLI1P1C1

ProASIC3, ProASIC3E



**Function**  
Active High Latch with Active High Preset and Clear and inverted Output

**Truth Table**

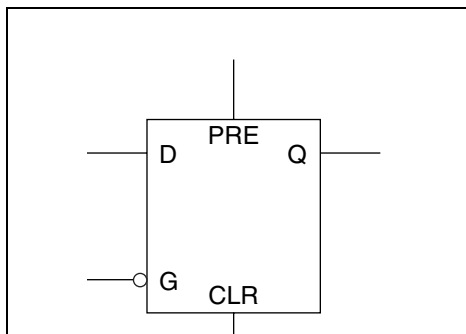
G	PRE	CLR	$QN_{n+1}$
X	1	0	0
X	X	1	1
1	0	0	!D
0	0	0	QN

<b>Input</b> CLR, PRE, G, D	<b>Output</b> QN
--------------------------------	---------------------

Family	Tiles
All	1

## DLN0P1C1

ProASIC3, ProASIC3E



Input	Output
CLR, PRE, G, D	Q

**Function**

Active Low Latch with Active High Preset and Clear

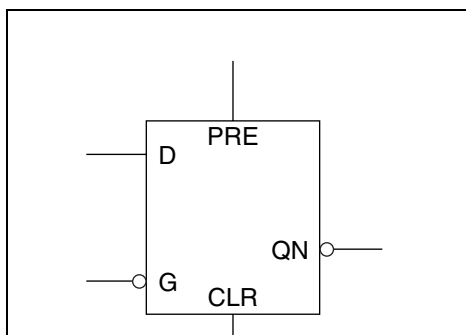
**Truth Table**

G	PRE	CLR	$Q_{n+1}$
X	1	0	1
X	X	1	0
0	0	0	D
1	0	0	Q

Family	Tiles
All	1

## DLI0P1C1

ProASIC3, ProASIC3E



Input	Output
CLR, PRE, G, D	QN

**Function**

Active Low Latch with Active High Preset and Clear and inverted Output

**Truth Table**

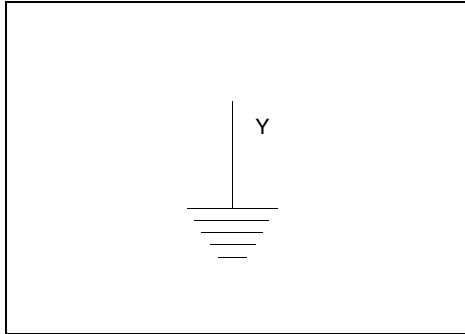
G	PRE	CLR	$QN_{n+1}$
X	1	0	0
X	X	1	1
0	0	0	!D
1	0	0	QN

Family	Tiles
All	1



**GND**

ProASIC3, ProASIC3E

**Function**

Ground

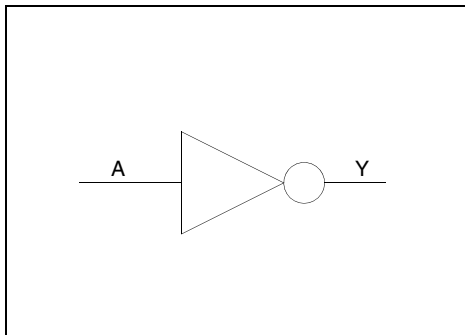
**Input****Output**

Y

NOTE: Ground does not use any tiles.

**INV**

ProASIC3, ProASIC3E

**Function**

Inverter with active low Output

**Truth Table**

A	Y
0	1
1	0

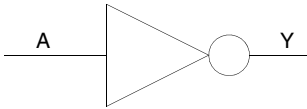
**Input**

A

**Output**

Y

Family	Tiles
All	1

**INVD****Function**

Inverter with active low Output  
NOTE: The Combiner will not remove this macro

**Truth Table**

A	Y
0	1
1	0

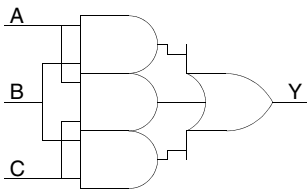
**Input**

A

**Output**

Y

Family	Tiles
All	1

**MAJ3****Function**

3-Input majority function

**Truth Table**

A	B	C	Y
X	0	0	0
0	0	X	0
0	X	0	0
X	1	1	1
1	X	1	1
1	1	X	1

**Input**

A, B, C

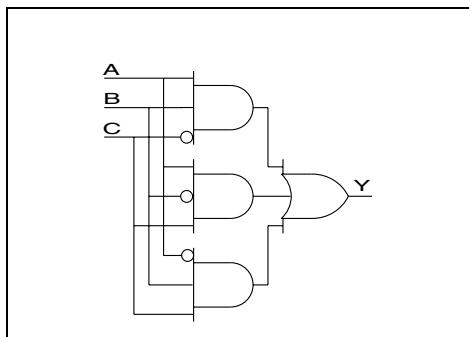
**Output**

Y

Family	Tiles
All	1

## MAJ3X

ProASIC3, ProASIC3E



**Input**  
A, B, C

**Output**  
Y

**Function**

2 of 3 function

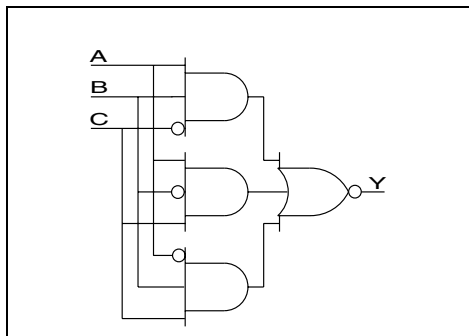
**Truth Table**

A	B	C	Y
0	0	0	0
1	0	0	0
0	1	0	0
1	1	0	1
0	0	1	0
1	0	1	1
0	1	1	1
1	1	1	0

Family	Tiles
All	1

## MAJ3XI

ProASIC3, ProASIC3E



**Input**  
A, B, C

**Output**  
Y

**Function**

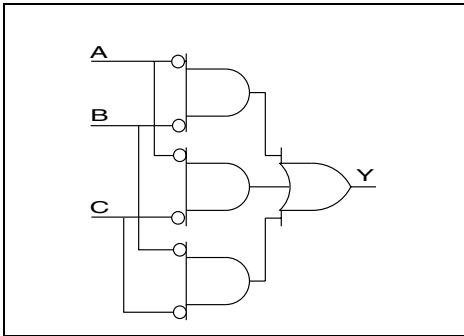
2 of 3 function with active low output

**Truth Table**

A	B	C	Y
0	0	0	1
1	0	0	1
0	1	0	1
1	1	0	0
0	0	1	1
1	0	1	0
0	1	1	0
1	1	1	1

Family	Tiles
All	1

# MIN3



**Function**  
3-Input minority function

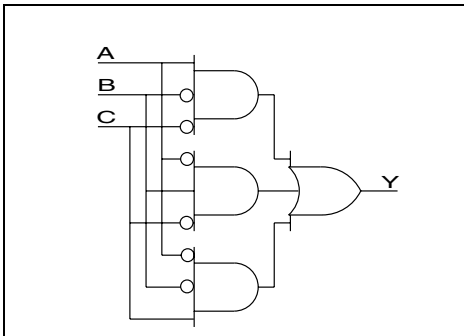
**Truth Table**

A	B	C	Y
X	0	0	1
0	0	X	1
0	X	0	1
X	1	1	0
1	X	1	0
1	1	X	0

<b>Input</b> A, B, C	<b>Output</b> Y
-------------------------	--------------------

Family	Tiles
All	1

# MIN3X



**Function**  
1 of 3 function

**Truth Table**

A	B	C	Y
0	0	0	0
1	0	0	1
0	1	0	1
1	1	0	0
0	0	1	1
1	0	1	0
0	1	1	0
1	1	1	0

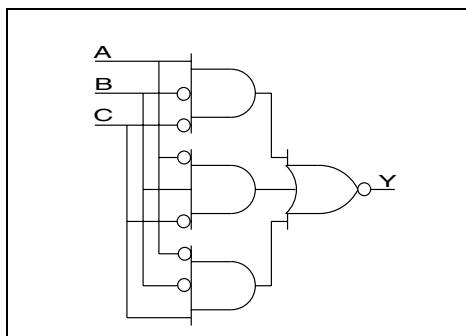
<b>Input</b> A, B, C	<b>Output</b> Y
-------------------------	--------------------

Family	Modules	
	Seq	Comb
54SX, 54SX-A, 54SX-S, eX		1



## MIN3XI

ProASIC3, ProASIC3E



**Input**  
A, B, C

**Output**  
Y

**Function**

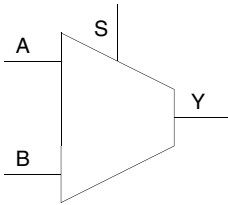
1 of 3 function with active low output

**Truth Table**

A	B	C	Y
0	0	0	1
1	0	0	0
0	1	0	0
1	1	0	1
0	0	1	0
1	0	1	1
0	1	1	1
1	1	1	1

Family	Tiles
All	1

## MX2

**Function**

2 to 1 Multiplexer

**Truth Table**

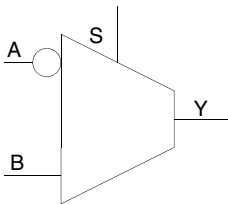
S	Y
0	A
1	B

**Input**  
A, B, S

**Output**  
Y

Family	Tiles
All	1

## MX2A

**Function**

2 to 1 Multiplexer with active low A-Input

**Truth Table**

S	Y
0	$\overline{A}$
1	B

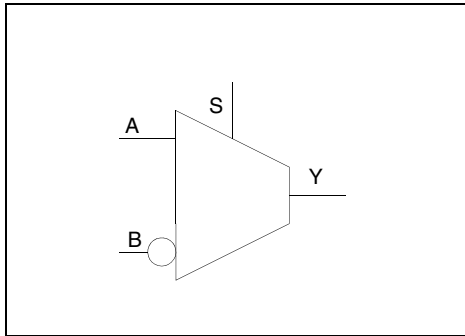
**Input**  
A, B, S

**Output**  
Y

Family	Tiles
All	1

## MX2B

ProASIC3, ProASIC3E

**Function**

2 to 1 Multiplexer with active low B-Input

**Truth Table**

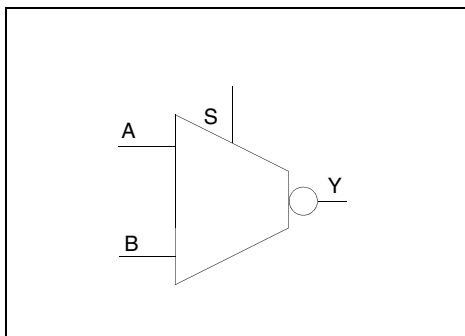
S	Y
0	A
1	$\overline{B}$

**Input**  
A, B, S**Output**  
Y

Family	Tiles
All	1

## MX2C

ProASIC3, ProASIC3E

**Function**

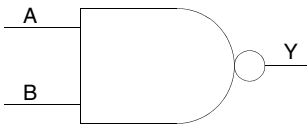
2 to 1 Multiplexer with active low Output

**Truth Table**

S	Y
0	$\overline{A}$
1	$\overline{B}$

**Input**  
A, B, S**Output**  
Y

Family	Tiles
All	1

**NAND2****Function**

2-Input NAND

**Truth Table**

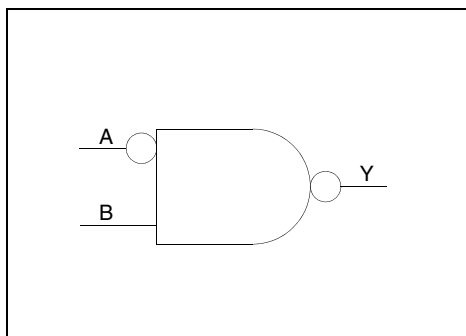
A	B	Y
X	0	1
0	X	1
1	1	0

**Input**  
A, B**Output**  
Y

Family	Tiles
All	1

**NAND2A**

ProASIC3, ProASIC3E

**Function**

2-Input NAND with active low A-Input

**Truth Table**

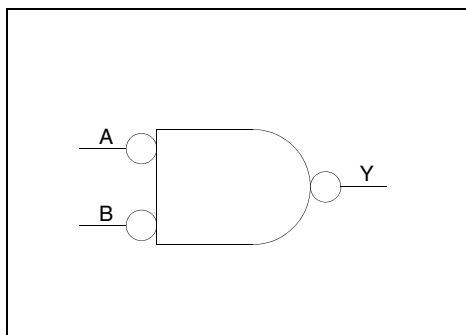
A	B	Y
X	0	1
0	1	0
1	X	1

Input  
A, BOutput  
Y

Family	Tiles
All	1

**NAND2B**

ProASIC3, ProASIC3E

**Function**

2-Input NAND with active low Inputs

**Truth Table**

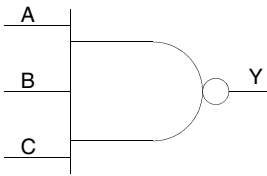
A	B	Y
0	0	0
X	1	1
1	X	1

Input  
A, BOutput  
Y

Family	Tiles
All	1

## NAND3

ProASIC3, ProASIC3E

**Function**

3-Input NAND

**Truth Table**

A	B	C	Y
X	X	0	1
X	0	X	1
0	X	X	1
1	1	1	0

**Input**

A, B, C

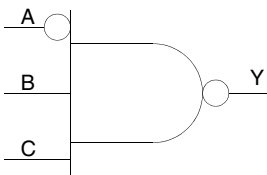
**Output**

Y

Family	Tiles
All	1

## NAND3A

ProASIC3, ProASIC3E

**Function**

3-Input NAND with active low A-Input

**Truth Table**

A	B	C	Y
X	X	0	1
X	0	X	1
0	1	1	0
1	X	X	1

**Input**

A, B, C

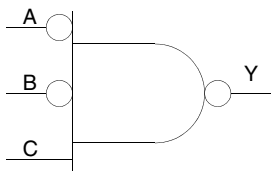
**Output**

Y

Family	Tiles
All	1

## NAND3B

ProASIC3, ProASIC3E

**Function**

3-Input NAND with active low A- and B-Inputs

**Truth Table**

A	B	C	Y
X	X	0	1
0	0	1	0
X	1	X	1
1	X	X	1

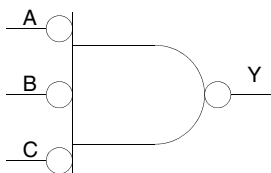
**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

## NAND3C

ProASIC3, ProASIC3E

**Function**

3-Input NAND with active low Inputs

**Truth Table**

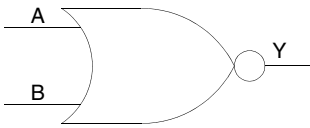
A	B	C	Y
0	0	0	0
X	X	1	1
X	1	X	1
1	X	X	1

**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

## NOR2

**Function**

2-Input NOR

**Truth Table**

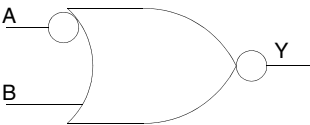
A	B	Y
0	0	1
X	1	0
1	X	0

**Input**  
A, B

**Output**  
Y

Family	Tiles
All	1

## NOR2A

**Function**

2-Input NOR with active low A-Input

**Truth Table**

A	B	Y
0	X	0
1	0	1
X	1	0

**Input**  
A, B

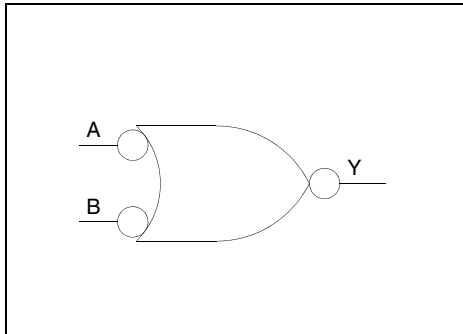
**Output**  
Y

Family	Tiles
All	1



## NOR2B

ProASIC3, ProASIC3E

**Function**

2-Input NOR with active low Inputs

**Truth Table**

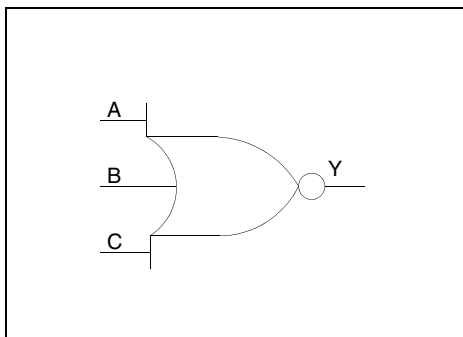
A	B	Y
X	0	0
0	X	0
1	1	1

Input  
A, BOutput  
Y

Family	Tiles
All	1

## NOR3

ProASIC3, ProASIC3E

**Function**

3-Input NOR

**Truth Table**

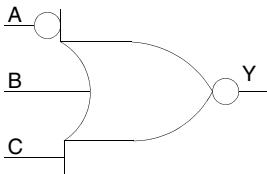
A	B	C	Y
0	0	0	1
X	X	1	0
X	1	X	0
1	X	X	0

Input  
A, B, COutput  
Y

Family	Tiles
All	1

## NOR3A

ProASIC3, ProASIC3E

**Function**

3-Input NOR with active low A-Input

**Truth Table**

A	B	C	Y
0	X	X	0
1	0	0	1
X	X	1	0
X	1	X	0

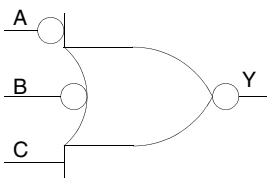
**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

## NOR3B

ProASIC3, ProASIC3E

**Function**

3-Input NOR with active low A- and B-Inputs

**Truth Table**

A	B	C	Y
X	0	X	0
0	X	X	0
1	1	0	1
X	X	1	0

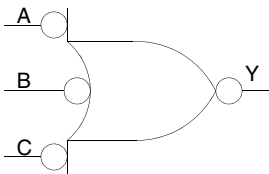
**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

## NOR3C

ProASIC3, ProASIC3E

**Function**

3-Input NOR with active low Inputs

**Truth Table**

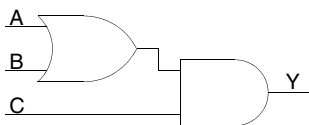
A	B	C	Y
X	X	0	0
X	0	X	0
0	X	X	0
1	1	1	1

**Input**  
A, B, C**Output**  
Y

Family	Tiles
All	1

## OA1

ProASIC3, ProASIC3E

**Function**

3 Input OR-AND

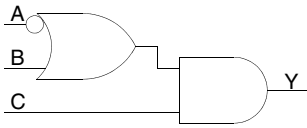
**Truth Table**

A	B	C	Y
X	X	0	0
0	0	X	0
X	1	1	1
1	X	1	1

**Input**  
A, B, C**Output**  
Y

Family	Tiles
All	1

## OA1A

**Function**

3 Input OR-AND with active low A-Input

**Truth Table**

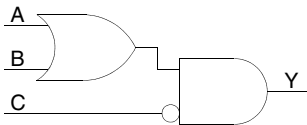
A	B	C	Y
X	X	0	0
0	X	1	1
1	0	X	0
X	1	1	1

**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

## OA1B

**Function**

3 Input OR-AND with active low C-Input

**Truth Table**

A	B	C	Y
0	0	X	0
X	1	0	1
X	X	1	0
1	X	0	1

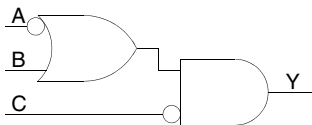
**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

## OA1C

ProASIC3, ProASIC3E

**Function**

3 Input OR-AND with active low A- and C-Inputs

**Truth Table**

A	B	C	Y
0	X	0	1
X	X	1	0
1	0	X	0
X	1	0	1

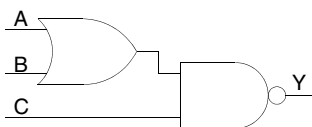
**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

## OAI1

ProASIC3, ProASIC3E

**Function**

3-Input OR-AND-INVERT

**Truth Table**

A	B	C	Y
X	X	0	1
0	0	X	1
X	1	1	0
1	X	1	0

**Input**  
A, B, C

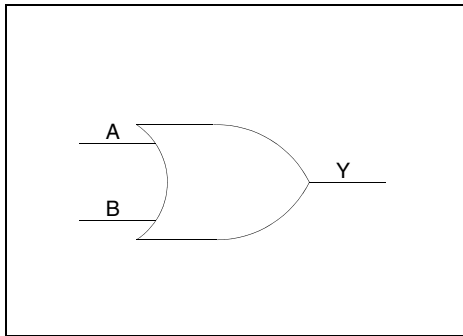
**Output**  
Y

Family	Tiles
All	1



## OR2

ProASIC3, ProASIC3E

**Function**

2-Input OR

**Truth Table**

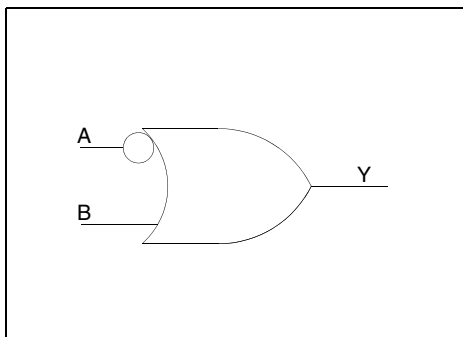
A	B	Y
0	0	0
X	1	1
1	X	1

**Input**  
A, B**Output**  
Y

Family	Tiles
All	1

## OR2A

ProASIC3, ProASIC3E

**Function**

2-Input OR with active low A-Input

**Truth Table**

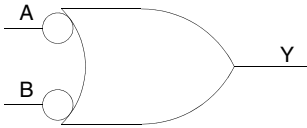
A	B	Y
0	X	1
1	0	0
X	1	1

**Input**  
A, B**Output**  
Y

Family	Tiles
All	1

## OR2B

ProASIC3, ProASIC3E

**Function**

2-Input OR with active low Inputs

**Truth Table**

A	B	Y
X	0	1
0	X	1
1	1	0

**Input**

A, B

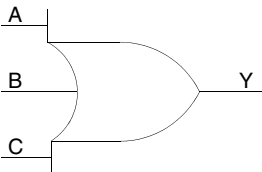
**Output**

Y

Family	Tiles
All	1

## OR3

ProASIC3, ProASIC3E

**Function**

3-Input OR

**Truth Table**

A	B	C	Y
0	0	0	0
X	X	1	1
X	1	X	1
1	X	X	1

**Input**

A, B, C

**Output**

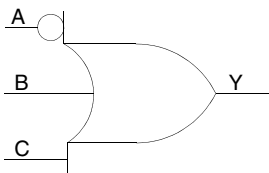
Y

Family	Tiles
All	1



## OR3A

ProASIC3, ProASIC3E

**Function**

3-Input OR with active low A-Input

**Truth Table**

A	B	C	Y
0	X	X	1
1	0	0	0
X	X	1	1
X	1	X	1

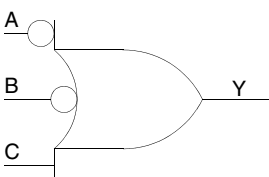
**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

## OR3B

ProASIC3, ProASIC3E

**Function**

3-Input OR with active low A- and B-Inputs

**Truth Table**

A	B	C	Y
X	0	X	1
0	X	X	1
1	1	0	0
X	X	1	1

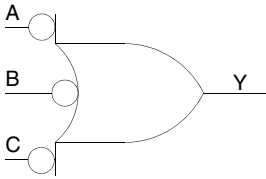
**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

## OR3C

ProASIC3, ProASIC3E

**Function**

3-Input OR with active low Inputs

**Truth Table**

A	B	C	Y
X	X	0	1
X	0	X	1
0	X	X	1
1	1	1	0

**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

## VCC

ProASIC3, ProASIC3E

**Function**

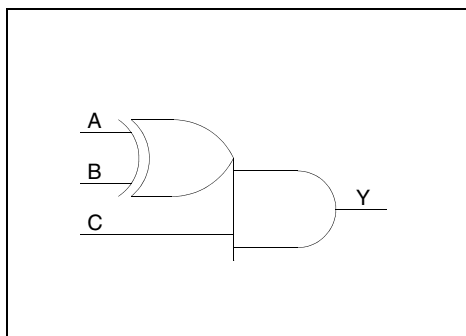
Power

**Input**
**Output**  
Y

NOTE: VCC does not use any modules.

**XA1**

ProASIC3, ProASIC3E



**Function**  
3-Input XOR-AND

**Truth Table**

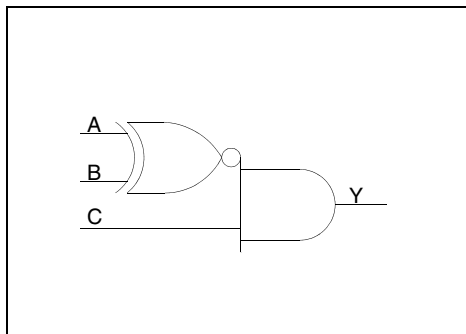
A	B	C	Y
X	X	0	0
0	0	X	0
0	1	1	1
1	0	1	1
1	1	X	0

<b>Input</b> A, B, C	<b>Output</b> Y
-------------------------	--------------------

Family	Tiles
All	1

**XA1A**

ProASIC3, ProASIC3E



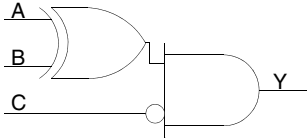
**Function**  
3-Input XNOR-AND

**Truth Table**

A	B	C	Y
X	X	0	0
0	0	1	1
0	1	X	0
1	0	X	0
1	1	1	1

<b>Input</b> A, B, C	<b>Output</b> Y
-------------------------	--------------------

Family	Tiles
All	1

**XA1B****Function**

3-Input XNOR-AND with active low C-input

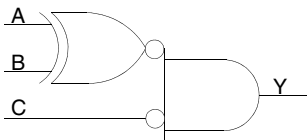
**Truth Table**

A	B	C	Y
X	X	1	0
0	0	X	0
1	0	0	1
0	1	0	1
1	1	X	0

**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

**XA1C****Function**

3-Input XNOR-AND with active low C-input

**Truth Table**

A	B	C	Y
X	X	1	0
0	0	0	1
1	0	X	0
0	1	X	0
1	1	0	1

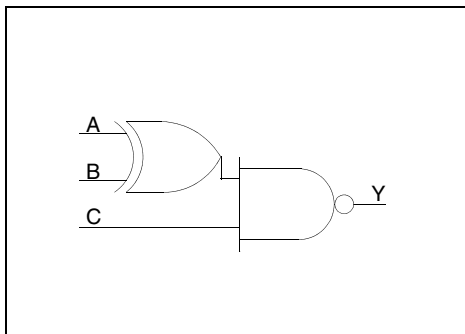
**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

**XAI1**

ProASIC3, ProASIC3E



**Function**  
3-Input XNOR-NAND

**Truth Table**

A	B	C	Y
X	X	0	1
0	0	X	1
1	0	1	0
0	1	1	0
1	1	X	1

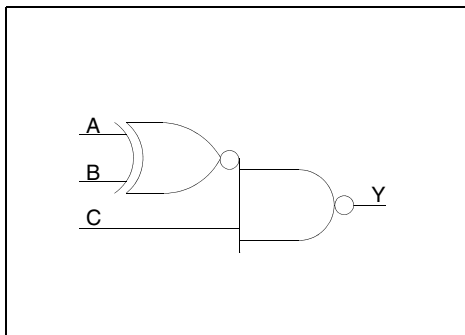
**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

**XAI1A**

ProASIC3, ProASIC3E



**Function**  
3-Input XNOR-NAND

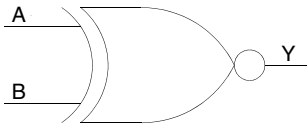
**Truth Table**

A	B	C	Y
X	X	0	1
0	0	1	0
1	0	X	1
0	1	X	1
1	1	1	0

**Input**  
A, B, C

**Output**  
Y

Family	Tiles
All	1

**XNOR2****Function**

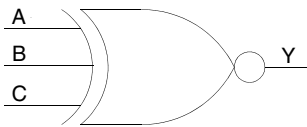
2- Input XNOR

**Truth Table**

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1

**Input**  
A, B**Output**  
Y

Family	Tiles
All	1

**XNOR3****Function**

3-Input XNOR

**Truth Table**

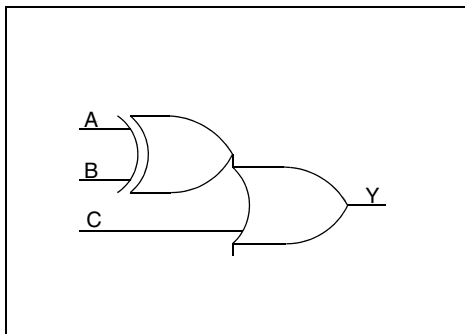
A	B	C	Y
0	0	0	1
1	0	0	0
0	1	0	0
1	1	0	1
0	0	1	0
1	0	1	1
0	1	1	1
1	1	1	0

**Input**  
A, B, C**Output**  
Y

Family	Tiles
All	1

**XO1**

ProASIC3, ProASIC3E

**Function**

3-Input XOR-OR

**Truth Table**

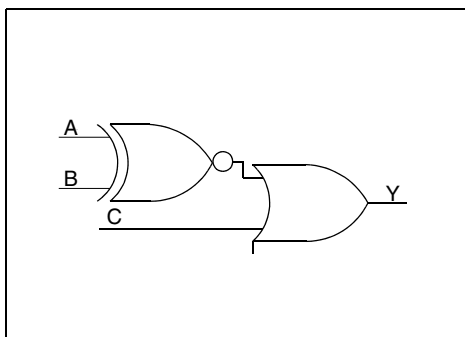
A	B	C	Y
0	0	0	0
X	X	1	1
0	1	X	1
1	0	X	1
1	1	0	0

**Input**  
A, B, C**Output**  
Y

Family	Tiles
All	1

**XO1A**

ProASIC3, ProASIC3E

**Function**

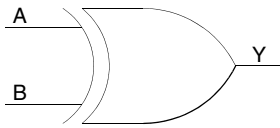
3-Input XNOR-OR

**Truth Table**

A	B	C	Y
0	0	0	1
X	X	1	1
0	1	0	0
1	0	0	0
1	1	0	1

**Input**  
A, B, C**Output**  
Y

Family	Tiles
All	1

**XOR2****Function**

2-Input XOR

**Truth Table**

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

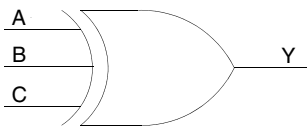
**Input**

A, B

**Output**

Y

Family	Tiles
All	1

**XOR3****Function**

3-Input XOR

**Truth Table**

A	B	C	Y
0	0	0	0
1	0	0	1
0	1	0	1
1	1	0	0
0	0	1	1
1	0	1	0
0	1	1	0
1	1	1	1

**Input**

A, B, C

**Output**

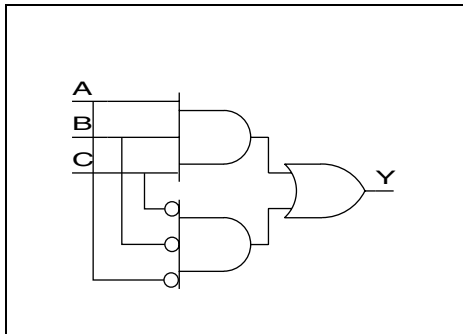
Y

Family	Tiles
All	1



## ZOR3

ProASIC3, ProASIC3E



<b>Input</b> A, B, C	<b>Output</b> Y
-------------------------	--------------------

**Function**

3-Input function

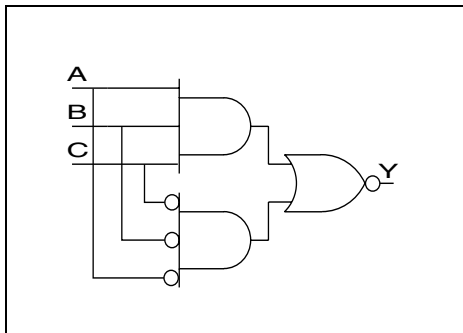
**Truth Table**

A	B	C	Y
0	0	0	1
1	0	0	0
0	1	0	0
1	1	0	0
0	0	1	0
1	0	1	0
0	1	1	0
1	1	1	1

Family	Tiles
All	1

## ZOR3I

ProASIC3, ProASIC3E



<b>Input</b> A, B, C	<b>Output</b> Y
-------------------------	--------------------

**Function**

3-Input function

**Truth Table**

A	B	C	Y
0	0	0	0
1	0	0	1
0	1	0	1
1	1	0	1
0	0	1	1
1	0	1	1
0	1	1	1
1	1	1	0

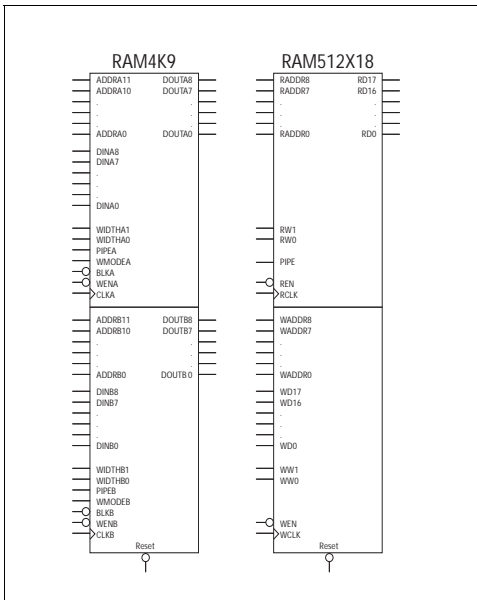
Family	Tiles
All	1



---

## *RAM and FIFO Macros*

# RAM4K9 and RAM512X18



## Function

RAM4K9 is a fully synchronous, true dual-port RAM with an optional pipeline stage; RAM512X18 is a fully synchronous, two-port RAM with an optional pipeline stage

Truth tables are listed below.

### Input

Inputs are shown on the left of the diagrams. For example, ADDR11, ADDR10, ..., ADDR0.

### Output

Outputs are shown at the right on the diagrams. For example, DOUT8, DOUT7, ..., DOUT0.

There are two RAM macros in the ProASIC3/ProASIC3E library: RAM4K9 and RAM512X18. The RAM4K9 is a fully synchronous, true dual-port RAM with an optional pipeline stage. It can be used for word widths up to 9 bits. Both ports are capable of reading and writing, making it possible to write with both ports or read with both ports simultaneously. You can also read from one port while writing to the other. Each port also has an optional pipeline stage that can be controlled separately via the PIPE pins. The RAM512X18 is a fully synchronous, two-port RAM with an optional pipeline stage. You can use it for word widths of 9 or 18 bits. It has one dedicated read port and one dedicated write port (you can read from one port while writing to the other). The read port also has an optional pipeline stage that you can control separately via the PIPE pin.

During the write operation of the RAM4K9, the WMODE pins control the data that appears on the read pins of the same port. When WMODE is high, the same data appears on the read and write ports at the rising CLK edge. When WMODE is low, the old data stored in the current memory location being addressed appears on the read port. There are no WMODE pins on the RAM512X18.

The aspect ratio of each port can be specified independently via the WIDTHA and WIDTHB pins. For the RAM512X18, the allowable values are 18 x 256 and 9 x 512. For the RAM4K9, the allowable values are 9 x 512, 4 x 1K, 2 x 2K, and 1 x 4K. Although it is possible to dynamically reconfigure the aspect ratios, the RAM was designed with only static configuration in mind, so the timing is unknown and you are discouraged from performing such operations. The same is true for the WMODE and PIPE configuration pins. The RAM simulation models reset the memory whenever the configuration pins change to reflect this.

The RAM4K9 only needs 2 bits to configure the WIDTH. The allowable RAM4K9 WIDTHA and WIDTHB values are shown in the table below.

*RAM4K9 WIDTHA and WIDTHB Values*

WIDTHA1, WIDTHA0	WIDTHB1, WIDTHB0	W x D
00	00	1 x 4K
01	01	2 x 2K
10	10	4 x 1K
11	11	9 x 512

The RAM512X18 also needs 2 bits to configure the read and write widths. The allowable RAM512X18 WW and RW values are shown in the table below.

*RAM512x18 WW and RW Values*

WW1, WW0	RW1, RW0	W x D
01	01	9 x 512
10	10	18 x 256
00, 11	00, 11	Illegal

When specifying a width that is less than the maximum (e.g. 1), the upper unused data input pins (e.g. DIN<sub>A8</sub> - DIN<sub>A1</sub>) must be connected to GND. When specifying a depth that is less than the maximum (e.g. 512), the upper unused address pins (e.g. ADD<sub>R</sub>A<sub>11</sub> - ADD<sub>R</sub>A<sub>9</sub>) must also be connected to GND.

When widths of 1, 2, and 4 are used, the ninth bit is skipped. This can cause counter-intuitive effects when these widths are used for read operations and larger widths are used for write operations (or vice versa). For example, if a width of 9 is used for writing and a width of 1 for reading, every 9th bit will be dropped. This effect may be desirable for removing parity bits. If a write width of 4 and read width of 9 is used, the 9th bit may either contain garbage or remnants of previous write operations when a write width of 9 or higher was being used. For this reason, ACTgen only supports the following aspect ratio combinations when one of the ports is configured with a 1-, 2-, or 4-bit width using the RAM4K9.

*ACTgen Supported Aspect Ratio Combinations for the RAM4K9*

READ	WRITE
1 x 4K	1 x 4K
1 x 4K	2 x 2K
1 x 4K	4 x 1K
2 x 2K	1 x 4K
2 x 2K	2 x 2K
2 x 2K	4 x 1K
4 x 1K	1 x 4K
4 x 1K	2 x 2K
4 x 1K	4 x 1K

The RAM4K9 can still be used for 9-bit width applications, but no other bit-width can be used with it other than 9-bits.

*ACTgen Supported Aspect Ratios for 9-bit Width Applications*

READ	WRITE
9 x 512	9 x 512

There are several restrictions that apply when you use an 18 x 256 aspect ratio. For this reason, ACTgen uses the RAM512X18 whenever 18-bit widths are specified. The only allowable combinations of read and write configurations for the RAM512X18 are as follows:

*RAM512X18 Read and Write Combinations*

READ	WRITE
18 x 256	18 x 256
18 x 256	9 x 512
9 x 512	18 x 256

The RADDR pins are always used for the read address in the above configurations and the WADDR pins are used for the write address. The RW pin is used to specify the read width and the WW pin for the write width. The WD pins are used for writing data and the RD pins for reading data.

*RAM4K9 Truth Table*

Operation	Address	CLK	BLK	WMODE	WEN	RESET	DI	DO
Deselect	X	X	H	X	X	H	X	Data-Last
Reset	X	X	X	X	X	L	X	L
Read	RADDR	Rising Edge	L	L	H	H	X	Data
Write (0)	RADDR	Rising Edge	L	L	L	H	WData	Data-Last
Write (1)	RADDR	Rising Edge	L	H	L	H	WData	WData

When deasserted, the BLK pins will cause the DO outputs to hold their last value. When asserted, the WEN pins can be used to switch each port between write and read mode. The RESET pin sets all outputs low but does not reset the memory. The WMODE pins are used to either allow the write data to appear immediately on the output pins or to hold the last value.

*RAM512x18 Truth Table*

Operation	Address	WCLK	REN	WEN	RESET	WD	RD
Reset	X	X	X	X	L	X	L
Read	RADDR	Rising Edge	L	X	H	X	Stored Data
Write	WADDR	Rising Edge	X	L	H	WData	Data-Last

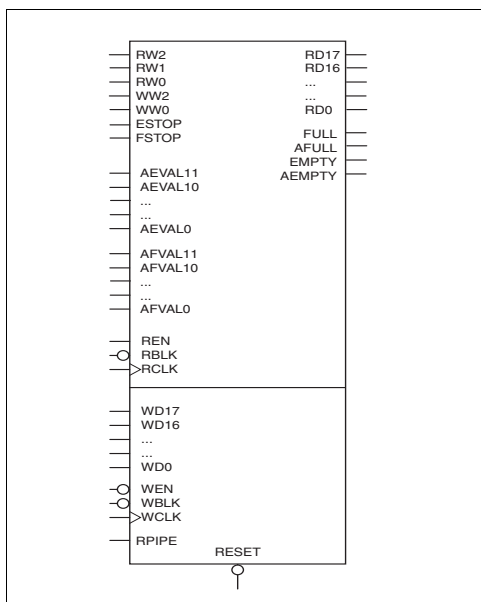
Use ACTgen to configure the RAM for typical use. ACTgen will not support dynamic reconfiguration or cascading width-wise. Customers wishing to avail themselves of such features must instantiate and configure the RAM macro manually.

## Warnings

- Simultaneous write to the same address from both ports is possible, but the results are undefined. Avoid writing to the same address simultaneously from both ports.
- Dynamic reconfiguration of any pins possible but not supported by ACTgen. Simulation models will reset memory.
- Cascading is possible and limited only by the number of available RAM blocks in a row, which is device dependent. ACTgen prompts you for device type information in order to correctly calculate the maximum.
- RESET has priority over BLKA and BLKB.
- In read mode (i.e. when WEN high) WMODE is ignored.
- Dual-port operation not possible unless both ports have the same aspect ratio.

## FIFO4K18

ProASIC3, ProASIC3E

**Function**

FIFO4K18 is fully synchronous and has its own built-in controller, capable of variable aspect ratios.

Truth tables are listed below.

**Input**

Inputs are shown on the left of the diagram. For example, AEVAL11, AEVAL10, ..., AEVAL0.

**Output**

Outputs are shown at the right of the diagram. For example, RD17, RD16, ..., RD0.

FIFO4K18 is fully synchronous and has its own built-in controller. Like the RAM, the FIFO can have different write and read aspect ratios that can be configured dynamically. The WW and RW pins are used to specify one of five allowable aspect ratios, as shown below.

*FIFO4K18 Aspect Ratios*

WW2, WW1, WW0 and RW2, RW1, RW0	W x H
000	1 x 4K
001	2 x 2K
010	4 x 1K
011	9 x 512
100	18 x 256
101, 110, 111	Illegal

The AEVAL and AFVAL pins are used to specify the almost empty and almost full threshold values, respectively. In order to handle different read and write aspect ratios, the values specified by the AEVAL and AFVAL pins are to be interpreted as the address of the last word stored in the FIFO. The FIFO actually contains separate write address (WADDR) and read address (RADDR) counters. These counters calculate the 12-bit memory address that is a function of WW and RW, respectively. WADDR is incremented every time a write operation is performed and RADDR is incremented every time a read operation is performed. Whenever the difference between WADDR and RADDR is greater than or equal to AFVAL, the AFULL output is raised. Likewise, whenever the difference

between WADDR and RADDR is less than or equal to AEVAL, the AEMPTY output is raised. Therefore AEVAL and AFVAL must be left-justified for widths greater than one (i.e. unused lsb bits must be grounded).

### *Aspect Ratio and Related Bits to Ground*

Aspect ratio	AEVAL/AFVAL bits to ground
1 x 4K	none
2 x 2K	0
4 x 1K	1:0
9 x 512	2:0
18 x 256	3:0

When the number of words stored in the FIFO reaches the amount specified by AEVAL while reading, the AEMPTY output will go high. Likewise when the number of words stored in the FIFO reaches the amount specified by AFVAL while writing, the AFULL output will go high. The FULL and EMPTY outputs will go high when the FIFO is completely full or empty, respectively.

It should be noted that the internal memory size is 512 X 9. When widths of 1, 2, and 4 are specified, the 9th bit is skipped.

The ESTOP pin is used to stop the read counter from counting any further once the FIFO is empty (i.e. the EMPTY flag goes high). Likewise, the FSTOP pin is used to stop the write counter from counting any further once the FIFO is full (i.e. the FULL flag goes high). These are configuration pins that should not be dynamically reconfigured. ACTgen treats them as static configuration pins and always ties them high.

Independent read and write operations are allowed, however only the read port can be pipelined. Data on the appropriate WD pins are written to the FIFO every rising WCLK edge as long as WEN and WBLK are low. Data is read from the FIFO and output on the appropriate RD pins every rising RCLK edge as long as REN is high and RBLK is low.

The active low RESET pin is used to asynchronously clear the outputs of the FIFO and reset the internal read and write address counters. It sets all the RD pins low, the FULL and AFULL pins low, and the EMPTY and AEMPTY pins high, however the contents of the memory remain unchanged. RESET has priority over RBLK and WBLK.

When instantiating the FIFO4K18, all unused input pins must be connected to GND.

## Warnings

- The WW, RW, AEVAL, and AFVAL pins can be dynamically configured, but only static configuration will be supported by ACTgen.
- The RPIPE signal can be dynamically configured, but only static configuration will be supported by ACTgen.
- No pipeline on the write port.
- Cascading allowed and supported in the width direction only by ACTgen. Cascading in the depth direction requires the use of a soft controller (i.e. implemented with core logic).
- WMODE pins should always be tied low for FIFO operations.
- ESTOP and FSTOP applications not clear. The effect of activating ESTOP is to allow the read pointer to wrap around, allowing the memory contents to be read over and over again with rewriting after EMPTY. The effect of activating FSTOP is not clear, however, since the write pointer could wrap around allowing overwriting of data which is never read. Therefore ACTgen will always tie these pins off high.



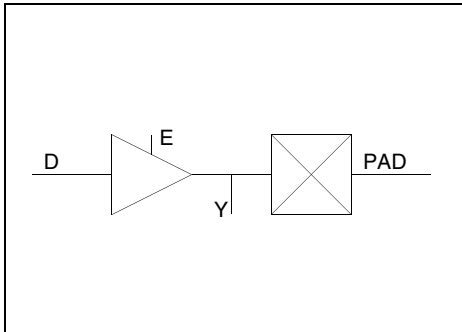




---

## *I/O Macros*

## BIBUF



<b>Input</b> D, E, PAD	<b>Output</b> PAD, Y
---------------------------	-------------------------

Family	I/O Tiles
All	1

**Function**

Bidirectional Buffer, High Slew (with Hidden Buffer at Y pin)

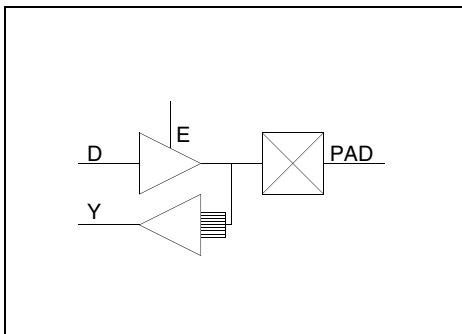
**Truth Table**

MODE	E	D	PAD	Y
OUTPUT	1	X	D	D
INPUT	0	X	X	PAD

**Attribute Default Values**

Attribute	Default Value	
	ProASIC3	ProASIC3E
IO_THRESH	LVTTL	LVTTL
OUT_DRIVE	12	12
SLEW	HIGH	HIGH
SKEW	OFF	OFF
IN_DELAY	N/A	OFF
SCHMITT_TRIGGER	N/A	OFF
RES_PULL	NONE	NONE

## CLKBIBUF



<b>Input</b> D, E, PAD	<b>Output</b> PAD, Y
---------------------------	-------------------------

Family	I/O Tiles
All	1

**Function**

Bidirectional with Input Dedicated to routed Clock Network

**Truth Table**

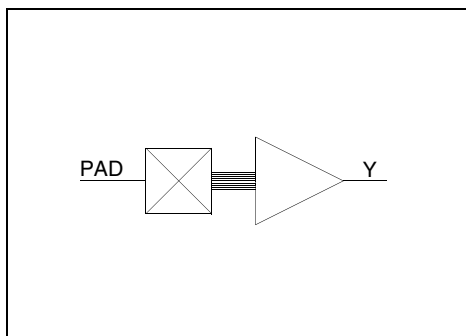
D	E	PAD	Y
X	0	Z	X
X	0	0	0
X	0	1	1
0	1	0	0
1	1	1	1

**Attribute Default Values**

Attribute	Default Value	
	ProASIC3	ProASIC3E
IO_THRESH	LVTTL	LVTTL
OUT_DRIVE	12	12
SLEW	HIGH	HIGH
SKEW	OFF	OFF
IN_DELAY	N/A	OFF
SCHMITT_TRIGGER	N/A	OFF
RES_PULL	NONE	NONE

## CLKBUF

ProASIC3, ProASIC3E



Input PAD	Output Y
--------------	-------------

Family	I/O Tiles
All	1

**Function**

Input for Dedicated Routed Clock Network

**Truth Table**

PAD	Y
0	0
1	1

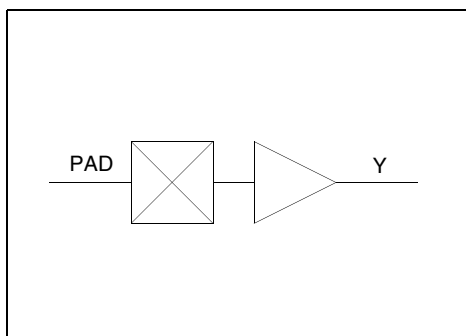
**Attribute Default Values**

Attribute	Default Value	
	ProASIC3	ProASIC3E
IO_THRESH	LVTTTL	LVTTTL
IN_DELAY	N/A	OFF
SCHMITT_TRIGGER	N/A	OFF
RES_PULL	NONE	NONE

NOTE 1: For an internal Clock net, refer to the CLKINT macro.

## INBUF

ProASIC3, ProASIC3E



Input PAD	Output Y
--------------	-------------

Family	I/O Tiles
All	1

**Function**

Input Buffer

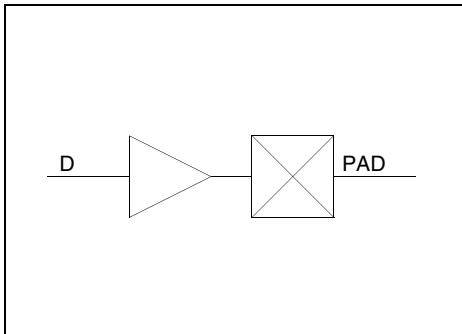
**Truth Table**

PAD	Y
0	0
1	1

**Attribute Default Values**

Attribute	Default Value	
	ProASIC3	ProASIC3E
IO_THRESH	LVTTTL	LVTTTL
IN_DELAY	N/A	OFF
SCHMITT_TRIGGER	N/A	OFF
RES_PULL	NONE	NONE

## OUTBUF



<b>Input</b>	<b>Output</b>
D	PAD

Family	I/O Tiles
All	1

**Function**  
Output Buffer, High Slew

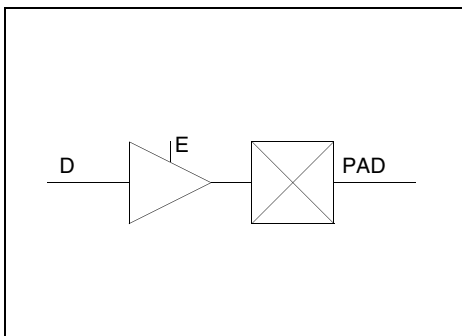
**Truth Table**

D	PAD
0	0
1	1

**Attribute Default Values**

Attribute	Default Value	
	ProASIC3	ProASIC3E
IO_THRESH	LVTTTL	LVTTTL
OUT_DRIVE	12	12
SLEW	HIGH	HIGH
RES_PULL	NONE	NONE

## TRIBUFF



<b>Input</b>	<b>Output</b>
D, E	PAD

Family	I/O Tiles
All	1

**Function**  
Tristate Output, High Slew

**Truth Table**

E	PAD
0	Z
1	D

**Attribute Default Values**

Attribute	Default Value	
	ProASIC3	ProASIC3E
IO_THRESH	LVTTTL	LVTTTL
OUT_DRIVE	12	12
SLEW	HIGH	HIGH
SKEW	OFF	OFF
RES_PULL	NONE	NONE

## ProASIC3 and ProASIC3E Input IO Macros

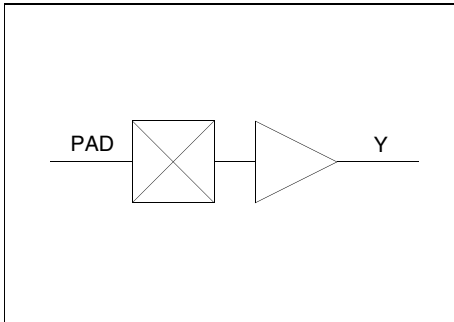
Names for the input buffers are composed of up to 4 parts:

- A base name indicating the type of buffer: INBUF
- IO Technology like LVCMOS
- An optional number code 33, 25, 18 or 15 indicating a 3.3, 2.5, 1.8 OR 1.5 voltage level.
- An optional one character code (U/D) designating a pull-up/down resistor. When the buffer has no resistor, this code is omitted.

For example:

- INBUF\_LVCMOS25U - An input LVCMOS buffer with 2.5 CMOS voltage levels, pull-up resistor.
- INBUF\_PCIX - An input PCIX buffer

## INBUF\_X

**Function**

Global Input Buffer

**Truth Table**

PAD	Y
0	0
1	1

**Input**  
PAD

**Output**  
Y

Family	I/O Tiles
All	1

**Available INBUF\_X Macro Types**

Name	Description
INBUF_LVCMOS5	LVCMOS Input buffer with 5.0 CMOS voltage level
INBUF_LVCMOS5D	LVCMOS Input buffer with 5.0 CMOS voltage level, pull-down resistor
INBUF_LVCMOS5U	LVCMOS Input buffer with 5.0 CMOS voltage level, pull-up resistor
INBUF_LVCMOS33	LVCMOS Input buffer with 3.3 CMOS voltage level
INBUF_LVCMOS33U	LVCMOS Input buffer with 3.3 CMOS voltage level, pull-up resistor
INBUF_LVCMOS33D	LVCMOS Input buffer with 3.3 CMOS voltage level, pull-down resistor
INBUF_LVCMOS25	LVCMOS Input buffer with 2.5 CMOS voltage level*
INBUF_LVCMOS25U	LVCMOS Input buffer with 2.5 CMOS voltage level, pull-up resistor*
INBUF_LVCMOS25D	LVCMOS Input buffer with 2.5 CMOS voltage level, pull-down resistor*
INBUF_LVCMOS18	LVCMOS Input buffer with 1.8 CMOS voltage level
INBUF_LVCMOS18U	LVCMOS Input buffer with 1.8 CMOS voltage level, pull-up resistor
INBUF_LVCMOS18D	LVCMOS Input buffer with 1.8 CMOS voltage level, pull-down resistor
INBUF_LVCMOS15	LVCMOS Input buffer with 1.5 CMOS voltage level
INBUF_LVCMOS15U	LVCMOS Input buffer with 1.5 CMOS voltage level, pull-up resistor
INBUF_LVCMOS15D	LVCMOS Input buffer with 1.5 CMOS voltage level, pull-down resistor
INBUF_PCI	PCI Input buffer
INBUF_PCIX	PCIX Input buffer
INBUF_GTL25	GTL Input buffer with 2.5 CMOS voltage level*
INBUF_GTL33	GTL Input buffer with 3.3 CMOS voltage level*
INBUF_GTLP25	GTLP Input buffer with 2.5 CMOS voltage level*
INBUF_GTLP33	GTLP Input buffer with 3.3 CMOS voltage level*
INBUF_HSTL_I	HSTL Class I Input buffer*
INBUF_HSTL_II	HSTL Class II Input buffer*
INBUF_SSTL2_I	SSTL2 Class I Input buffer*
INBUF_SSTL2_II	SSTL2 Class II Input buffer*
INBUF_SSTL3_I	SSTL3 Class I Input buffer*
INBUF_SSTL3_II	SSTL3 Class II Input buffer*

\* = not supported in ProASIC3



## Bi-Directional IO Macros

Names for the bi-directional buffers are composed of up to 4 parts:

- A base name indicating the type of buffer: BIBUF
- Optional IO Technology like LVCMOS
- An optional number code indicating drive strength in milli-amps.
- An optional one character code (S/F) indicating high(F) slew or low(S) slew
- An optional one character code (U/D) designating a pull-up/down resistor. When the buffer has no resistor, this code is omitted.

For example:

- BIBUF\_LVCMOS25 - A bi-directional LVCMOS buffer with 2.5 CMOS voltage levels, pull-up resistor
- BIBUF\_S\_8- A bi-directional buffer with low slew and 8 mA drive strength

BIBUF_X		ProASIC3, ProASIC3E														
		<b>Function</b> Bidirectional Buffer (with Hidden Buffer at Y pin)														
		<b>Truth Table</b> <table border="1"> <thead> <tr> <th>MODE</th> <th>E</th> <th>D</th> <th>PAD</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>OUTPUT</td> <td>1</td> <td>X</td> <td>D</td> <td>D</td> </tr> <tr> <td>INPUT</td> <td>0</td> <td>X</td> <td>X</td> <td>PAD</td> </tr> </tbody> </table>		MODE	E	D	PAD	Y	OUTPUT	1	X	D	D	INPUT	0	X
MODE	E	D	PAD	Y												
OUTPUT	1	X	D	D												
INPUT	0	X	X	PAD												
<b>Input</b> D, E, PAD		<b>Output</b> Y, PAD														
		<b>Family</b> <table border="1"> <thead> <tr> <th>Family</th> <th>I/O Tiles</th> </tr> </thead> <tbody> <tr> <td>All</td> <td>1</td> </tr> </tbody> </table>		Family	I/O Tiles	All	1									
Family	I/O Tiles															
All	1															

### BIBUF\_X Macro Types

Name	Description
BIBUF_LVCMOS5	LVCMOS Bi-directional buffer with 5.0 CMOS voltage level
BIBUF_LVCMOS5D	LVCMOS Bi-directional buffer with 5.0 CMOS voltage level, pull-down resistor
BIBUF_LVCMOS5U	LVCMOS Bi-directional buffer with 5.0 CMOS voltage level, pull-up resistor
BIBUF_LVCMOS33	LVCMOS Bi-directional buffer with 3.3 CMOS voltage level
BIBUF_LVCMOS33U	LVCMOS Bi-directional buffer with 3.3 CMOS voltage level, pull-up resistor
BIBUF_LVCMOS33D	LVCMOS Bi-directional buffer with 3.3 CMOS voltage level, pull-down resistor
BIBUF_LVCMOS25	LVCMOS Bi-directional buffer with 2.5 CMOS voltage level*
BIBUF_LVCMOS25U	LVCMOS Bi-directional buffer with 2.5 CMOS voltage level, pull-up resistor*
BIBUF_LVCMOS25D	LVCMOS Bi-directional buffer with 2.5 CMOS voltage level, pull-down resistor*
BIBUF_LVCMOS18	LVCMOS Bi-directional buffer with 1.8 CMOS voltage level
BIBUF_LVCMOS18U	LVCMOS Bi-directional buffer with 1.8 CMOS voltage level, pull-up resistor
BIBUF_LVCMOS18D	LVCMOS Bi-directional buffer with 1.8 CMOS voltage level, pull-down resistor
BIBUF_LVCMOS15	LVCMOS Bi-directional buffer with 1.5 CMOS voltage level
BIBUF_LVCMOS15U	LVCMOS Bi-directional buffer with 1.5 CMOS voltage level, pull-up resistor
BIBUF_LVCMOS15D	LVCMOS Bi-directional buffer with 1.5 CMOS voltage level, pull-down resistor
BIBUF_PCI	PCI Bi-directional buffer
BIBUF_PCIX	PCIX Bi-directional buffer
BIBUF_SSTL2_I	SSTL2 class I bi-directional buffer*
BIBUF_SSTL2_II	SSTL2 class II bi-directional buffer*
BIBUF_SSTL3_I	SSTL3 class I bi-directional buffer
BIBUF_SSTL3_II	SSTL3 class II bi-directional buffer*
BIBUF_HSTL_I	HSTL class I bi-directional buffer*
BIBUF_HSTL_II	HSTL class II bi-directional buffer*
BIBUF_GTL25	GTL bi-directional buffer*
BIBUF_GTL33	GTL bi-directional buffer*

**BIBUF\_X Macro Types (Continued)**

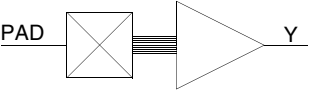
Name	Description
BIBUF_GTLP25	GTLP Bi-directional buffer with 2.5 CMOS voltage level*
BIBUF_GTLP33	GTLP Bi-directional buffer with 3.3 CMOS voltage level*
BIBUF_F_2	Bi-directional buffer with high slew
BIBUF_F_2U	Bi-directional buffer with high slew and pull-up resistor
BIBUF_F_2D	Bi-directional buffer with high slew and pull-down resistor
BIBUF_F_4	Bi-directional buffer with high slew
BIBUF_F_4U	Bi-directional buffer with high slew and pull-up resistor
BIBUF_F_4D	Bi-directional buffer with high slew and pull-down resistor
BIBUF_F_6	Bi-directional buffer with high slew
BIBUF_F_6U	Bi-directional buffer with high slew and pull-up resistor
BIBUF_F_6D	Bi-directional buffer with high slew and pull-down resistor
BIBUF_F_8	Bi-directional buffer with high slew
BIBUF_F_8U	Bi-directional buffer with high slew and pull-up resistor
BIBUF_F_8D	Bi-directional buffer with high slew and pull-down resistor
BIBUF_F_12	Bi-directional buffer with high slew
BIBUF_F_12U	Bi-directional buffer with high slew and pull-up resistor
BIBUF_F_12D	Bi-directional buffer with high slew and pull-down resistor
BIBUF_F_16	Bi-directional buffer with high slew
BIBUF_F_16U	Bi-directional buffer with high slew and pull-up resistor
BIBUF_F_16D	Bi-directional buffer with high slew and pull-down resistor
BIBUF_F_24	Bi-directional buffer with high slew*
BIBUF_F_24U	Bi-directional buffer with high slew and pull-up resistor*
BIBUF_F_24D	Bi-directional buffer with high slew and pull-down resistor*
BIBUF_S_2	Bi-directional buffer with low slew
BIBUF_S_2U	Bi-directional buffer with low slew and pull-up resistor
BIBUF_S_2D	Bi-directional buffer with low slew and pull-down resistor
BIBUF_S_4	Bi-directional buffer with low slew
BIBUF_S_4U	Bi-directional buffer with low slew and pull-up resistor
BIBUF_S_4D	Bi-directional buffer with low slew and pull-down resistor
BIBUF_S_6	Bi-directional buffer with low slew
BIBUF_S_6U	Bi-directional buffer with low slew and pull-up resistor
BIBUF_S_6D	Bi-directional buffer with low slew and pull-down resistor
BIBUF_S_8	Bi-directional buffer with low slew
BIBUF_S_8U	Bi-directional buffer with low slew and pull-up resistor
BIBUF_S_8D	Bi-directional buffer with low slew and pull-down resistor
BIBUF_S_12	Bi-directional buffer with low slew
BIBUF_S_12U	Bi-directional buffer with low slew and pull-up resistor
BIBUF_S_12D	Bi-directional buffer with low slew and pull-down resistor
BIBUF_S_16	Bi-directional buffer with low slew
BIBUF_S_16U	Bi-directional buffer with low slew and pull-up resistor
BIBUF_S_16D	Bi-directional buffer with low slew and pull-down resistor
BIBUF_S_24	Bi-directional buffer with low slew *
BIBUF_S_24U	Bi-directional buffer with low slew and pull-up resistor *
BIBUF_S_24D	Bi-directional buffer with low slew and pull-down resistor *

\* = Not supported in ProASIC3

## Clock Buffers

Names for the input buffers are composed of up to 3 parts:

- A base name indicating the type of buffer: CLKBUF
- IO Technology like LVCMOS
- An optional number code 33, 25, 18 or 15 indicating a 3.3, 2.5, 1.8 OR 1.5 voltage level

<b>CLKBUF_X</b>		ProASIC3, ProASIC3E										
<div style="display: flex; justify-content: space-between;"> <div style="border: 1px solid black; padding: 10px; width: 40%;">  </div> <div style="border: 1px solid black; padding: 10px; width: 55%;"> <p><b>Function</b> Input for Dedicated Routed Clock Network</p> <p><b>Truth Table</b></p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th>PAD</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> </tr> </tbody> </table> <p><b>Family</b></p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th>Family</th> <th>I/O Tiles</th> </tr> </thead> <tbody> <tr> <td>All</td> <td>1</td> </tr> </tbody> </table> </div> </div>			PAD	Y	0	0	1	1	Family	I/O Tiles	All	1
PAD	Y											
0	0											
1	1											
Family	I/O Tiles											
All	1											
<p><b>Input</b> PAD</p>	<p><b>Output</b> Y</p>											

NOTE 1: For an internal Clock net, refer to the CLKINT macro.

### Available CLKBUF\_X Macro Types

Name	Description
CLKBUF_LVCMOS5	LVCMOS Clock buffer with 5.0 CMOS voltage level
CLKBUF_LVCMOS33	LVCMOS Clock buffer with 3.3 CMOS voltage level
CLKBUF_LVCMOS25	LVCMOS Clock buffer with 2.5 CMOS voltage level *
CLKBUF_LVCMOS18	LVCMOS Clock buffer with 1.8 CMOS voltage level
CLKBUF_LVCMOS15	LVCMOS Clock buffer with 1.5 CMOS voltage level
CLKBUF_PCI	PCI Clock buffer
CLKBUF_PCIX	PCIX Clock buffer
CLKBUF_GTL25	GTL Clock buffer with 2.5 CMOS voltage level *
CLKBUF_GTL33	GTL Clock buffer with 3.3 CMOS voltage level*
CLKBUF_GTLP25	GTLP Clock buffer with 2.5 CMOS voltage level *
CLKBUF_GTLP33	GTLP Clock buffer with 3.3 CMOS voltage level *
CLKBUF_HSTL_I	HSTL Class I Clock buffer *
CLKBUF_HSTL_II	HSTL Class II Clock buffer *
CLKBUF_SSTL2_I	SSTL2 Class I Clock buffer *
CLKBUF_SSTL2_II	SSTL2 Class II Clock buffer *
CLKBUF_SSTL3_I	SSTL3 Class I Clock buffer *
CLKBUF_SSTL3_II	SSTL3 Class II Clock buffer *

\* = Not supported in ProASIC3

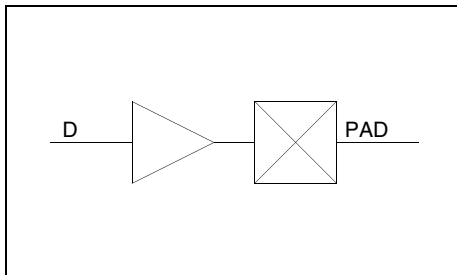
### Output Buffers

Names for the bi-directional buffers are composed of up to 4 parts:

- A base name indicating the type of buffer: OUTBUF
- Optional IO Technology like LVC MOS
- An optional number code indicating drive strength in milli-amps.
- An optional one character code (S/F) indicating high (F) slew or low (S) slew

## OUTBUF\_X

ProASIC3, ProASIC3E

**Function**

Output Buffer,

**Truth Table**

D	PAD
0	0
1	1

**Family**

Family	I/O Tiles
All	1

**Input**

D

**Output**

PAD

**Available OUTBUF\_X Macro Types**

Name	Description
OUTBUF_LVCMOS5	LVC MOS Output buffer with 5.0 CMOS voltage level
OUTBUF_LVCMOS33	LVC MOS Output buffer with 3.3 CMOS voltage level
OUTBUF_LVCMOS25	LVC MOS Output buffer with 2.5 CMOS voltage level *
OUTBUF_LVCMOS18	LVC MOS Output buffer with 1.8 CMOS voltage level
OUTBUF_LVCMOS15	LVC MOS Output buffer with 1.5 CMOS voltage level
OUTBUF_PCI	PCI Output buffer
OUTBUF_PCIX	PCIX Output buffer
OUTBUF_HSTL_I	HSTL Class I Output buffer *
OUTBUF_HSTL_II	HSTL Class II Output buffer *
OUTBUF_SSTL2_I	SSTL2 Class I Output buffer *
OUTBUF_SSTL2_II	SSTL2 Class II Output buffer *
OUTBUF_SSTL3_I	SSTL3 Class I Output buffer *
OUTBUF_SSTL3_II	SSTL3 Class II Output buffer *
OUTBUF_GTL25	GTL Output buffer with 2.5 CMOS voltage level *
OUTBUF_GTL33	GTL Output buffer with 3.3 CMOS voltage level *
OUTBUF_GTLP25	GTLP Output buffer with 2.5 CMOS voltage level *
OUTBUF_GTLP33	GTLP Output buffer with 3.3 CMOS voltage level *
OUTBUF_F_2	Output buffer with high slew
OUTBUF_F_4	Output buffer with high slew
OUTBUF_F_6	Output buffer with high slew
OUTBUF_F_8	Output buffer with high slew
OUTBUF_F_12	Output buffer with high slew
OUTBUF_F_16	Output buffer with high slew
OUTBUF_F_24	Output buffer with high slew *
OUTBUF_S_2	Output buffer with low slew
OUTBUF_S_4	Output buffer with low slew
OUTBUF_S_6	Output buffer with low slew
OUTBUF_S_8	Output buffer with low slew
OUTBUF_S_12	Output buffer with low slew
OUTBUF_S_16	Output buffer with low slew
OUTBUF_S_24	Output buffer with low slew *

\* = Not supported in ProASIC3

## Tri-State Buffer Macros

Names for the tri-state outputs are composed of up to 4 parts:

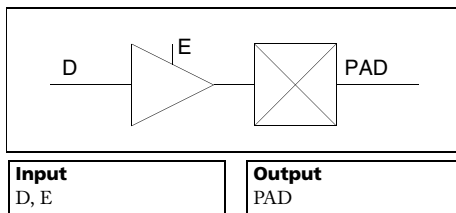
- A base name indicating the type of buffer: TRIBUFF
- Optional IO Technology like LVCMOS
- An optional number code indicating drive strength in milli-amps.
- An optional one character code (S/F) indicating high(F) slew or low(S) slew
- An optional one character code (U/D) designating a pull-up/down resistor. When the buffer has no resistor, this code is omitted.

For example:

- TRIBUFF\_LVCMOS25 - A tri-state LVCMOS output with 2.5 CMOS voltage levels, pull-up resistor
- TRIBUFF\_S\_8- A tri-state output with low slew and 8 mA drive strength

## TRIBUFF\_X

ProASIC3, ProASIC3E

**Function**

Tristate Output

**Truth Table**

E	PAD
0	Z
1	D

**Family**

Family	I/O Tiles
All	1

**TRIBUFF\_X Macro Types**

Name	Description
TRIBUFF_LVCMOS5	LVCMOS tri-state output with 5.0 CMOS voltage level
TRIBUFF_LVCMOS5U	LVCMOS tri-state output with 5.0 CMOS voltage level, pull-up resistor
TRIBUFF_LVCMOS5D	LVCMOS tri-state output with 5.0 CMOS voltage level, pull-down resistor
TRIBUFF_LVCMOS33	LVCMOS tri-state output with 3.3 CMOS voltage level
TRIBUFF_LVCMOS33U	LVCMOS tri-state output with 3.3 CMOS voltage level, pull-up resistor
TRIBUFF_LVCMOS33D	LVCMOS tri-state output with 3.3 CMOS voltage level, pull-down resistor
TRIBUFF_LVCMOS25	LVCMOS tri-state output with 2.5 CMOS voltage level *
TRIBUFF_LVCMOS25U	LVCMOS tri-state output with 2.5 CMOS voltage level, pull-up resistor *
TRIBUFF_LVCMOS25D	LVCMOS tri-state output with 2.5 CMOS voltage level, pull-down resistor *
TRIBUFF_LVCMOS18	LVCMOS tri-state output with 1.8 CMOS voltage level
TRIBUFF_LVCMOS18U	LVCMOS tri-state output with 1.8 CMOS voltage level, pull-up resistor
TRIBUFF_LVCMOS18D	LVCMOS tri-state output with 1.8 CMOS voltage level, pull-down resistor
TRIBUFF_LVCMOS15	LVCMOS tri-state output with 1.5 CMOS voltage level
TRIBUFF_LVCMOS15U	LVCMOS tri-state output with 1.5 CMOS voltage level, pull-up resistor
TRIBUFF_LVCMOS15D	LVCMOS tri-state output with 1.5 CMOS voltage level, pull-down resistor
TRIBUFF_PCI	PCI tri-state output
TRIBUFF_PCIX	PCIX tri-state output
TRIBUFF_GTL25	GTL tri-state output with 2.5 CMOS voltage level *
TRIBUFF_GTL33	GTL tri-state output with 3.3 CMOS voltage level *
TRIBUFF_GTLP25	GTLP tri-state output with 2.5 CMOS voltage level *
TRIBUFF_GTLP33	GTLP tri-state output with 3.3 CMOS voltage level *
TRIBUFF_HSTL_I	HSTL Class I tri-state output buffer *
TRIBUFF_HSTL_II	HSTL Class II tri-state output buffer *
TRIBUFF_SSTL2_I	SSTL2 Class I tri-state output buffer *
TRIBUFF_SSTL2_II	SSTL2 Class II tri-state output buffer *
TRIBUFF_SSTL3_I	SSTL3 Class I tri-state output buffer *
TRIBUFF_SSTL3_II	SSTL3 Class II tri-state output buffer *
TRIBUFF_F_2	Tri-state output with high slew
TRIBUFF_F_2U	Tri-state output with high slew and pull-up resistor
TRIBUFF_F_2D	Tri-state output with high slew and pull-down resistor
TRIBUFF_F_4	Tri-state output with high slew
TRIBUFF_F_4U	Tri-state output with high slew and pull-up resistor
TRIBUFF_F_4D	Tri-state output with high slew and pull-down resistor
TRIBUFF_F_6	Tri-state output with high slew
TRIBUFF_F_6U	Tri-state output with high slew and pull-up resistor
TRIBUFF_F_6D	Tri-state output with high slew and pull-down resistor
TRIBUFF_F_8	Tri-state output with high slew
TRIBUFF_F_8U	Tri-state output with high slew and pull-up resistor
TRIBUFF_F_8D	Tri-state output with high slew and pull-down resistor
TRIBUFF_F_12	Tri-state output with high slew
TRIBUFF_F_12U	Tri-state output with high slew and pull-up resistor
TRIBUFF_F_12D	Tri-state output with high slew and pull-down resistor
TRIBUFF_F_16	Tri-state output with high slew

**TRIBUFF\_X Macro Types (Continued)**

<b>Name</b>	<b>Description</b>
TRIBUFF_F_16U	Tri-state output with high slew and pull-up resistor
TRIBUFF_F_16D	Tri-state output with high slew and pull-down resistor
TRIBUFF_F_24	Tri-state output with high slew*
TRIBUFF_F_24U	Tri-state output with high slew and pull-up resistor *
TRIBUFF_F_24D	Tri-state output with high slew and pull-down resistor *
TRIBUFF_S_2	Tri-state output with low slew
TRIBUFF_S_2U	Tri-state output with low slew and pull-up resistor
TRIBUFF_S_2D	Tri-state output with low slew and pull-down resistor
TRIBUFF_S_4	Tri-state output with low slew
TRIBUFF_S_4U	Tri-state output with low slew and pull-up resistor
TRIBUFF_S_4D	Tri-state output with low slew and pull-down resistor
TRIBUFF_S_6	Tri-state output with low slew
TRIBUFF_S_6U	Tri-state output with low slew and pull-up resistor
TRIBUFF_S_6D	Tri-state output with low slew and pull-down resistor
TRIBUFF_S_8	Tri-state output with low slew
TRIBUFF_S_8U	Tri-state output with low slew and pull-up resistor
TRIBUFF_S_8D	Tri-state output with low slew and pull-down resistor
TRIBUFF_S_12	Tri-state output with low slew
TRIBUFF_S_12U	Tri-state output with low slew and pull-up resistor
TRIBUFF_S_12D	Tri-state output with low slew and pull-down resistor
TRIBUFF_S_16	Tri-state output with low slew
TRIBUFF_S_16U	Tri-state output with low slew and pull-up resistor
TRIBUFF_S_16D	Tri-state output with low slew and 3x pull-down resistor
TRIBUFF_S_24	Tri-state output with low slew *
TRIBUFF_S_24U	Tri-state output with low slew and pull-up resistor *
TRIBUFF_S_24D	Tri-state output with low slew and pull-down resistor *

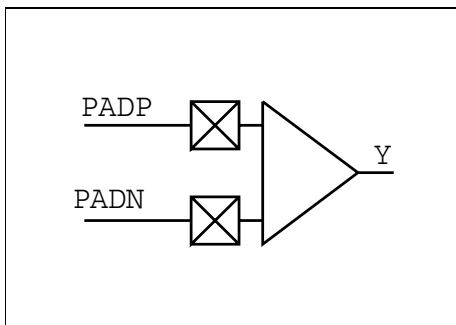
\* = Not supported in ProASIC3



# Differential IO Macros

## INBUF\_LVDS; INBUF\_LVPECL

ProASIC3, ProASIC3E


**Function**

INBUF\_LVDS and INBUF\_LVPECL

**Input**

PADP; PADN

**Output**

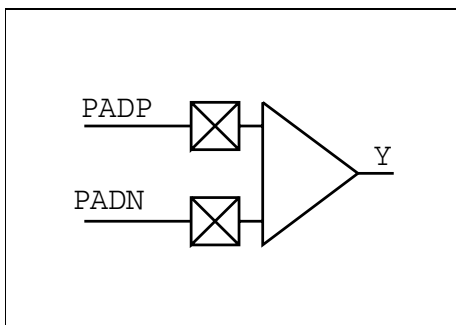
Y

**Available Differential Macro Types**

Name	Description
INBUF_LVDS	
INBUF_LVPECL	

## CLKBUF\_LVDS; CLKBUF\_LVPECL

ProASIC3, ProASIC3E


**Function**

CLKBUF\_LVDS and CLKBUF\_LVPECL

**Input**

PADP; PADN

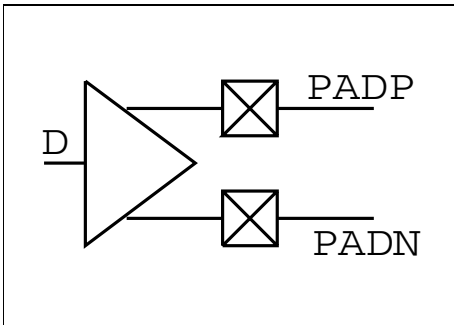
**Output**

Y

**Available Differential Macro Types**

Name	Description
CLKBUF_LVDS	
CLKBUF_LVPECL	

# OUTBUF\_LVDS; OUTBUF\_LVPECL



**Function**  
 OUTBUF\_LVDS and OUTBUF\_LVPECL

**Input**  
 D

**Output**  
 PADP, PADN

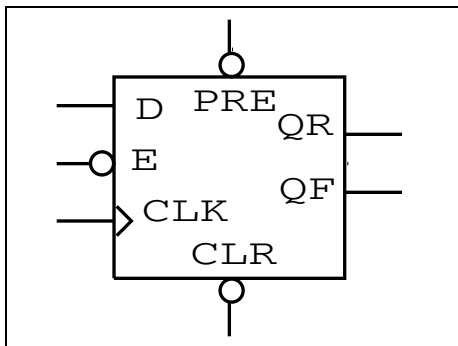
**Available Differential Macro Types**

Name	Description
OUTBUF_LVDS	
OUTBUF_LVPECL	

# DDR Macros

## DDR\_REG

ProASIC3, ProASIC3E



**Function**  
 DDR (DDR) Register with active low write and read enables; please refer to the ProASIC3/E datasheet for more information on the DDR\_REG

Truth Table

CLR	PRE	E	CLK	QR(n+1)	QF(n+1)
0	X	X	X	0	0
1	0	X	X	1	1
1	1	1	X	QR(n)	QF(n)
1	1	0	↑	D(↑)	X
1	1	0	↓	X	D(↓)

**Input**  
 D, CLK, E, PRE, CLR

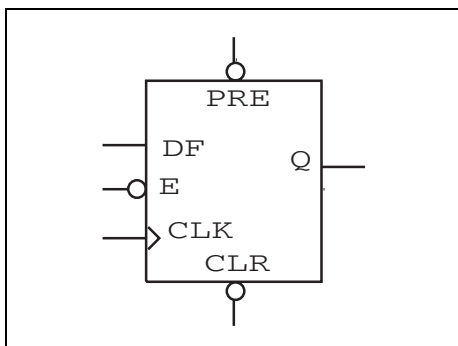
**Output**  
 QR, QF

Family

Family	I/O Tiles
All	1

## DDR\_OUT

ProASIC3, ProASIC3E



**Function**  
 DDR (DDR) output with active low write and read enables; please refer to the ProASIC3/E datasheet for more information on the DDR\_OUT

Truth Table

CLR	PRE	E	CLK	Q
0	X	X	X	0
1	0	X	X	1
1	1	1	X	QR(n)
1	1	0	↑	D(↑)
1	1	0	↓	X

**Input**  
 DF, CLK, E, PRE, CLR

**Output**  
 Q

Family

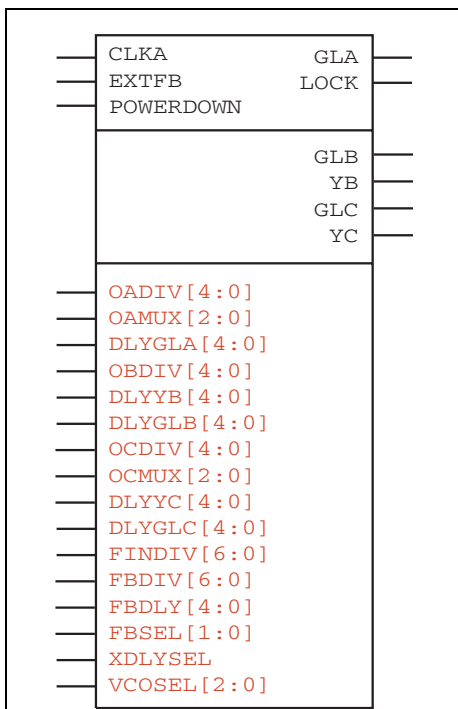
Family	I/O Tiles
All	1



---

## ***PLL Macros***

## PLL

**Function**

Static PLL

Actel recommends that you use ACTgen to generate your PLLs; ACTgen calculates the settings for all the pins in the PLL for the required input-output frequency combinations.

Refer to the latest Actel datasheets on PLLs for ProASIC3 / ProASIC3E for more information. They are available at <http://www.actel.com>.

**Inputs / Outputs**

See the description below for an explanation of the inputs and outputs available on the Dynamic PLL; all inputs are shown on the left, and outputs are to the right.

The static PLL supports only a single input. The Combiner is able to combine the PLL with the regular CLKBUF macros and any of the CCC macros to utilize available unused globals.

In the symbol shown above, all the required user-accessible inputs and outputs are above the top horizontal line. The ones below the top line are optional inputs and outputs. The static configuration inputs are below the third line. These pins can only be connected to GND or VCC.

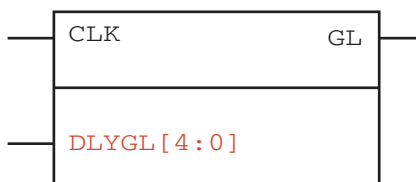
The table below summarizes the configuration control bits.

## Configuration Control Bits Summary

NAME	FUNCTION
FINDIV<6:0>	7-BIT INPUT DIVIDER (/N)
FBDIV<6:0>	7-BIT FEEDBACK DIVIDER (/M)
OADIV<4:0>	5-BIT OUTPUT DIVIDER (/U)
OBDIV<4:0>	5-BIT OUTPUT DIVIDER (/V)
OCDIV<4:0>	5-BIT OUTPUT DIVIDER (/W)
OAMUX<2:0>	3-BIT POST-PLL MUXA (BEFORE DIVIDER /U)
OBMUX<2:0>	3-BIT POST-PLL MUXB (BEFORE DIVIDER /V)
OCMUX<2:0>	3-BIT POST-PLL MUXC (BEFORE DIVIDER /W)
FBSEL<1:0>	2-BIT PLL FEEDBACK MUX
FBDLY<4:0>	FEEDBACK DELAY
XDLYSEL	1-BIT PLL FEEDBACK MUX
DLYGLA<4:0>	DELAY ON GLOBAL A
DLYGLB<4:0>	DELAY ON GLOBAL B
DLYGLC<4:0>	DELAY ON GLOBAL C
DLYB<4:0>	DELAY ON YB
DLYC<4:0>	DELAY ON YC
VCOSSEL<2:0>	3-BIT VCO GEAR CONTROL (4 FREQUENCY RANGES)

## Static Clock with Divider and/or Delay

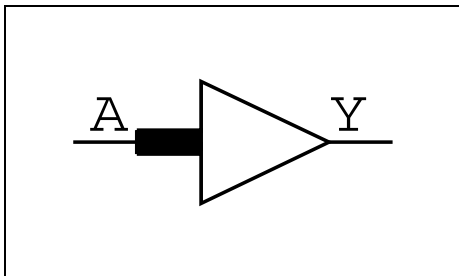
The Combiner is able to combine the clock conditioning circuit macro with the regular CLKBUF macros and the PLL to utilize available unused globals.



The CLKDLY is essentially a CLKBUF with a delay. The PLLINT macro is included to unambiguously show Designer which routing resources are required to connect the REFCLK input: The PLLINT is used when REFCLK is driven by a pad in a different I/O tile.

# PLLINT

ProASIC3, ProASIC3E



**Function**

PLL Int

**Truth Table**

A	Y
0	0
1	1

Connect only to the REFCLK input of PLL when the PLL is driven by a pad other than the one in the same super cluster.

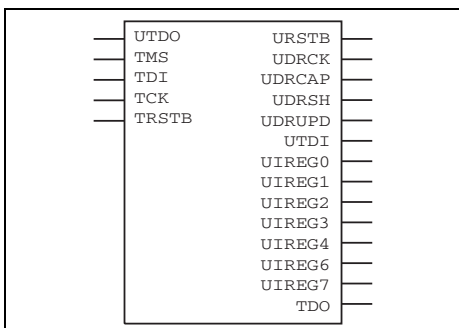
**Input**  
A

**Output**  
Y

Refer to the latest Actel datasheets on PLLs for ProASIC / ProASIC 3E for more information. They are available at <http://www.actel.com>.

# UJTAG

ProASIC3, ProASIC3E



**Function**

The UJTAG macro is a special purpose macro. It is provided to allow users access to the user JTAG circuitry on board the chip. You must instantiate a UJTAG macro in their design if they plan to make use of the user JTAG feature. It is identical to the APA and A500K UJTAG macro.

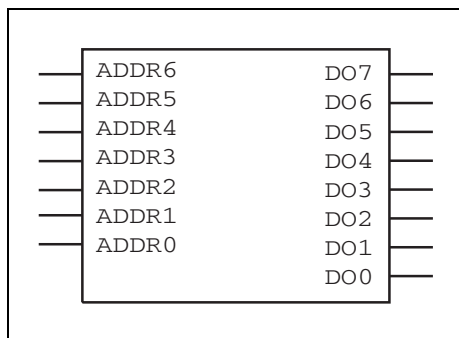
**Input**

**Output**



## UFROM

ProASIC3, ProASIC3E



Input

Output

**Function**

The UFROM is the USER FlashROM macro. It is a simple 128 X 8 asynchronous read-only memory. There is only one UFROM per chip. New data appears on the DO pins whenever the address changes on the ADDR pins. The UFROM can only be programmed by the user via the JTAG pins. There is currently no support for programming the UFROM in any of the CAE tools or libraries, however the simulation models will utilize a memory initialization file so users can specify the contents of the memory for simulation purposes. The memory initialization file will be an ASCII format text file containing exactly 128 lines of 8-character binary strings.



---

# Product Support

Actel backs its products with various support services including Customer Service, a Customer Technical Support Center, a web site, an FTP site, electronic mail, and worldwide sales offices. This appendix contains information about contacting Actel and using these support services.

## Customer Service

Contact Customer Service for non-technical product support, such as product pricing, product upgrades, update information, order status, and authorization.

From Northeast and North Central U.S.A., call **650.318.4480**

From Southeast and Southwest U.S.A., call **650.318.4480**

From South Central U.S.A., call **650.318.4434**

From Northwest U.S.A., call **650.318.4434**

From Canada, call **650.318.4480**

From Europe, call **650.318.4252** or **+44 (0)1276.401500**

From Japan, call **650.318.4743**

From the rest of the world, call **650.318.4743**

Fax, from anywhere in the world **650.318.8044**

## Actel Customer Technical Support Center

Actel staffs its Customer Technical Support Center with highly skilled engineers who can help answer your hardware, software, and design questions. The Customer Technical Support Center spends a great deal of time creating application notes and answers to FAQs. So, before you contact us, please visit our online resources. It is very likely we have already answered your questions.

## Actel Technical Support

Visit the [Actel Customer Support website \(www.actelcom/.custsup/search.html\)](http://www.actelcom/.custsup/search.html) for more information and support. Many answers available on the searchable web resource include diagrams, illustrations, and links to other resources on the Actel web site.

## Website

You can browse a variety of technical and non-technical information on Actel's [home page](http://www.actel.com), at [www.actel.com](http://www.actel.com).

# Contacting the Customer Technical Support Center

Highly skilled engineers staff the Technical Support Center from 7:00 A.M. to 6:00 P.M., Pacific Time, Monday through Friday. Several ways of contacting the Center follow:

## Email

You can communicate your technical questions to our email address and receive answers back by email, fax, or phone. Also, if you have design problems, you can email your design files to receive assistance. We constantly monitor the email account throughout the day. When sending your request to us, please be sure to include your full name, company name, and your contact information for efficient processing of your request.

The technical support email address is [tech@actel.com](mailto:tech@actel.com).

## Phone

Our Technical Support Center answers all calls. The center retrieves information, such as your name, company name, phone number and your question, and then issues a case number. The Center then forwards the information to a queue where the first available application engineer receives the data and returns your call. The phone hours are from 7:00 A.M. to 6:00 P.M., Pacific Time, Monday through Friday. The Technical Support numbers are:

**650.318.4460**

**800.262.1060**

Customers needing assistance outside the US time zones can either contact technical support via email ([tech@actel.com](mailto:tech@actel.com)) or contact a local sales office. [Sales office listings](#) can be found at [www.actel.com/contact/offices/index.html](http://www.actel.com/contact/offices/index.html).

**For more information about Actel's products, visit our website at <http://www.actel.com>**

**Actel Corporation** • 2061 Stierlin Court • Mountain View, CA 94043 USA  
Customer Service: 650.318.1010 • Customer Applications Center: 800.262.1060

**Actel Europe Ltd.** • Dunlop House, Riverside Way • Camberley, Surrey GU15 3YL • United Kingdom  
Phone +44 (0)1276 401 463 • Fax +44 (0)1276 401 490

**Actel Japan** • EXOS Ebisu Bldg. 4F • 1-24-14 Ebisu Shibuya-ku • Tokyo 150 • Japan  
Phone +81.03.3445.7671 Fax +81.03.3445.7668

**Actel Hong Kong** • 39<sup>th</sup> Floor, One Pacific Place • 88 Queensway, Admiralty Hong Kong  
Phone +852.227.35712 Fax +852.227.35999

50200029-0

