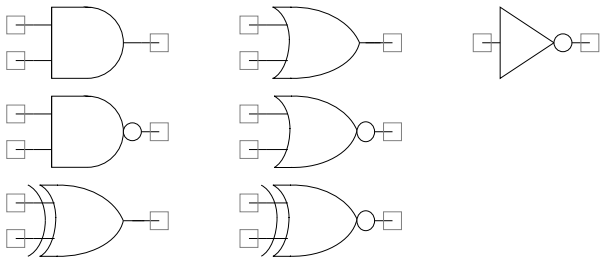
Digital Logic Gates

1.0

Features

- Industry-standard logic gates
- Configurable number of inputs up to 8
- Optional array of gates



General Description

Logic gates provide basic boolean operations. The output of a logic gate is a boolean combinatorial function of the inputs. There are seven basic logic gates: AND, OR, Inverter (NOT), NAND, NOR, XOR, and XNOR.

When to Use a Logic Gate

Use logic gates when you want to perform basic logical operations. You can perform more complex operations with various combinations of the basic logic gates.

Input/Output Connections

This section describes the various input and output connections for the Logic Gates. An asterisk (*) in the list of I/Os indicates that the I/O may be hidden on the symbol under the conditions listed in the description of that I/O.

Input 1

Every logic gate has at least one digital input

Input 2

All logic gates, except the Inverter (NOT), have a second digital input.

Inputs 3 - 8 *

All logic gates, except the Inverter (NOT), can be configured to have up to eight input terminals.

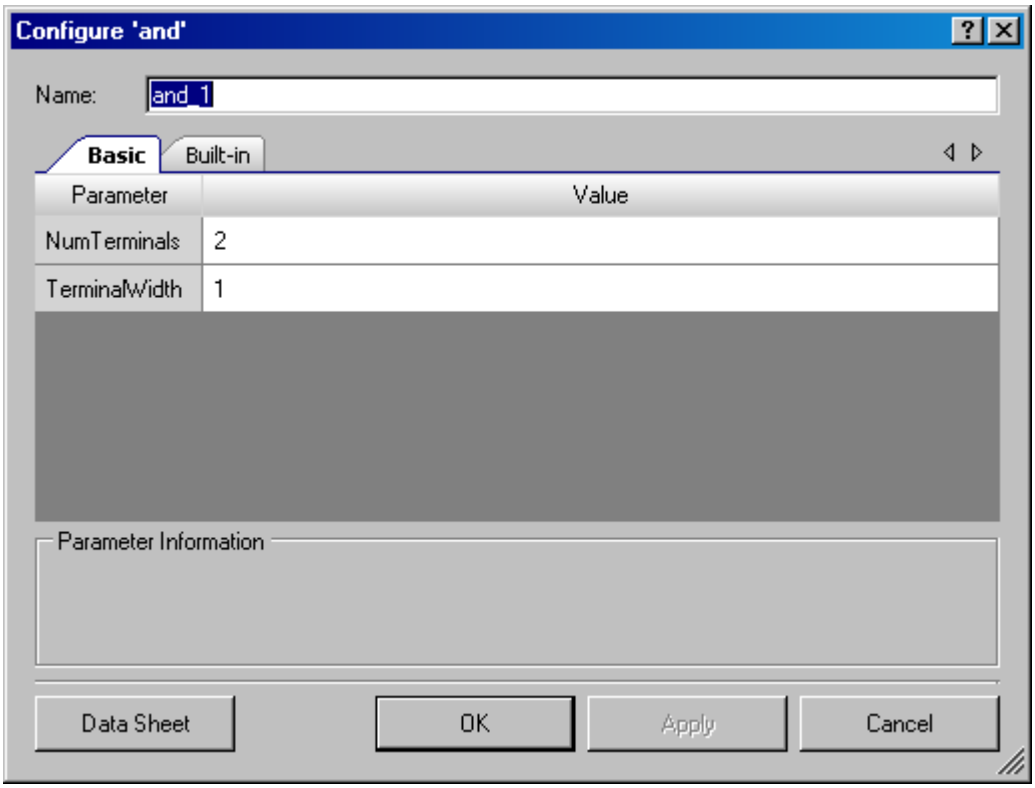
Output

All logic gates have one output.

Component Parameters

Drag a logic gate onto your design and double-click it to open the **Configure** dialog.

Figure 1. Configure AND Dialog



Logic gates provide the following parameters:

NumTerminals

For all logic gates, except Inverter (NOT), this parameter determines the number of input terminals. The minimum is **2** (default) and the maximum is **8**. The Inverter (NOT) gate does not have this parameter.

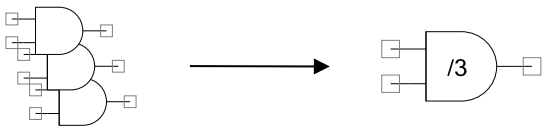
TerminalWidth

This parameter determines the array width of the input and output terminals (default is **1**).

You can create an array of the logic gate, which may be useful when a logic function is performed across buses of inputs and a bus of outputs. It defines the number of wires in the buses that can be attached to the same number of discrete logic gates in parallel.

Figure 2 is a conceptual example of an AND gate with 2 inputs and an array width of 3. In practice, only a single AND gate would be shown with 2 inputs and 1 output, each of which would connect to a bus with a width of 3 wires.

Figure 2. Array Conceptual Example



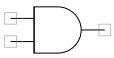
Resources

All digital logic gates are converted to a sum of products and placed into a Universal Digital Block (UDB) programmable logic. This process results in digital logic gates being automatically optimized and placed into the PSoC device. Resource use depends on the specific logic created and cannot be determined before project compilation in PSoC Creator.

Functional Description

This section describes each of the logic gates separately. The gate logic descriptions use the following convention to describe logic levels:

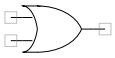
- True = 1 = high logic level
- False = 0 = low logic level

AND Gate 

The AND gate performs logical multiplication in the same way as the logical AND operator. It has two or more inputs and one output. As shown in Table 1, the output is true when all inputs are true. Otherwise, the output is false.

Table 1. AND Truth Table

Input 1	Input 2	Output
0	0	0
0	1	0
1	0	0
1	1	1

OR Gate 

The OR gate performs logical addition in the same way as the logical inclusive OR operator. It has two or more inputs and one output. As shown in Table 2 the output is true if the inputs are true. If all inputs are false then the output is false.

Table 2. OR Truth Table

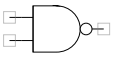
Input 1	Input 2	Output
0	0	0
0	1	1
1	0	1
1	1	1

Inverter (NOT) Gate 

The Inverter, also called a NOT gate, performs the basic logic function called inversion. In other words, this gate changes one logic level (true/false) to the opposite logic level. The NOT gate has only one input and one output. As shown in Table 3, the output is false when the input is true and vice-versa.

Table 3. Inverter (NOT) Truth Table

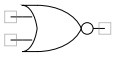
Input	Output
1	0
0	1

NAND Gate 

The NAND gate operates as an AND gate followed by a NOT gate. It acts in the manner of the logical operation AND followed by negation. It has two or more inputs and one output. As shown in Table 4, the output is false if all inputs are true. Otherwise, the output is true.

Table 4. NAND Truth Table

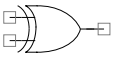
Input 1	Input 2	Output
0	0	1
0	1	1
1	0	1
1	1	0

NOR Gate 

The NOR gate operates as an OR gate followed by a NOT gate. It has two or more inputs and one output. As shown in Table 5, the output is true if all inputs are false. Otherwise, the output is false.

Table 5. NOR Truth Table

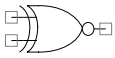
Input 1	Input 2	Output
0	0	1
0	1	0
1	0	0
1	1	0

XOR Gate 

The XOR (exclusive-OR) gate is useful as a parity generator. It has two or more inputs and one output. As shown in Table 6, the output is true when there are an odd number of true inputs. Otherwise, the output is false.

Table 6. XOR Truth Table

Input 1	Input 2	Input 3	Output
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

XNOR gate 

The XNOR (exclusive-NOR) gate operates as an XOR gate followed by a NOT gate. It has two or more inputs and one output. As shown in Table 7, the output is true when there is an even number of true inputs. Otherwise, the output is false.

Table 7. XNOR Truth Table

Input 1	Input 2	Input 3	Output
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

Component Changes

This section lists the major changes in the component from the previous version.

Version	Description of Changes	Reason for Changes / Impact
1.0.f	Minor datasheet edits.	
1.0.e	Minor datasheet edits.	
1.0.d	Minor datasheet update.	
1.0.c	Minor datasheet edits and updates.	
1.0.b	Minor datasheet edits and updates.	
1.0.a	Updated datasheet.	XNOR truth table was incorrect.

© Cypress Semiconductor Corporation, 2009-2017. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC (“Cypress”). This document, including any software or firmware included or referenced in this document (“Software”), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress’s patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage (“Unintended Uses”). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, WICED, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.