SYNOPSYS®

DesignWare® GTECH Library

Databook

K-2015.06
June 2015

Copyright Notice and Proprietary Information Notice

Destination Control Statement

All technical data contained in this publication is subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the reader's responsibility to determine the applicable regulations and to comply with them.

Disclaimer

SYNOPSYS, INC., AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Trademarks

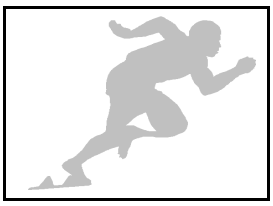
Synopsys and certain Synopsys product names are trademarks of Synopsys, as set forth at http://www.synopsys.com/Company/Pages/Trademarks.aspx.

All other product or company names may be trademarks of their respective owners.

Synopsys, Inc.
690 E. Middlefield Road
Mountain View, CA 94043

www.synopsys.com

Contents



DesignWare Building Block IP

For information on the following topics, refer to the *DesignWare Building Block Quick Reference* guide:

- Overview of DesignWare
- Updating DesignWare Building Block IP
- Setting Up DesignWare Building Block IP
- Accessing DesignWare Building Block IP
- Synthesizing DesignWare Building Block IP
- Simulating DesignWare Building Block IP
- Technical Support

Obsoleted IP

A list of Obsoleted DesignWare Building Block IP is maintained at the end of the *DesignWare Building Block IP Release Notes.*

Using DesignWare Building Block IP

For information on the following topics, refer to the Using DesignWare Building Block IP topics:

- DesignWare Building Block IP Directory Structure
- Setting Up Your DesignWare Building Block IP Environment
- Integrating DesignWare Building Block IP into Your Design
  - Inferring DesignWare Building Block IP
  - Parameter and Port Mapping
  - Accessing DesignWare Building Block IP Usage Examples
  - Extracting the Appropriate Code Example
- Evaluating IP
- Simulating withDesignWare Building Block IP
  - Simulating with Gate-Level GTECH Models
  - Simulating with RTL Source Code Simulation Models
  - Post-Synthesis Gate-Level Simulation
  - Layout and Back-Annotation
  - Issues Specific to VHDL Simulation
  - Analyze entities before respective architectures.
  - Issues Specific to Verilog Simulation

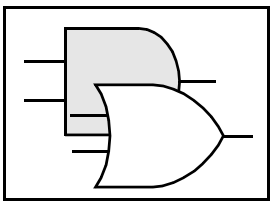
- SimulatingDesignWare Building Block IP
- Compiling the Design
- Hierarchy
- Writing Out the Design
- Reading Designs Back in After Layout

- Test Considerations

Revision History

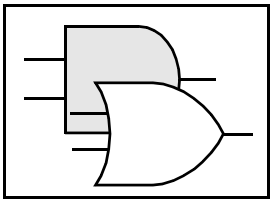
Table 1-1 Revision History

Date	Description
October 2011	Fixed last row of truth table for GTECH_AO22 on page 43.



GTECH Library

Synopsys provides the GTECH technology-independent library to aid users in developing technology-independent parts. Also, DesignWare IP often use these cells for their implementation. This generic technology library, called gtech.db, contains common logic elements. gtech.db can be found under the Synopsys root directory in libraries/syn. Simulation models are located under the Synopsys root directory in packages/gtech/src (VHDL) and packages/gtech/src_ver (Verilog).This chapter contains datasheets that describe the generic parts. Each datasheet contains pin descriptions and a truth table, in addition to examples of how to instantiate the part in VHDL and in Verilog.



GTECH_NOT

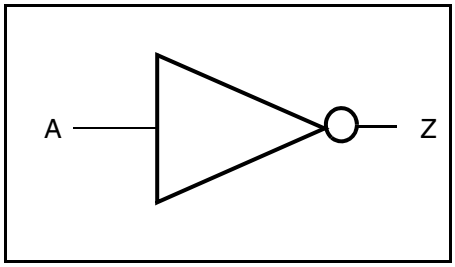
Inverter

Truth Table

A	Z
0	1
1	0

Pin Description

Pin	Width	Direction
A	1	Input
Z	1	Output



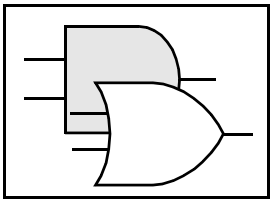
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_not_inst is
  port(in1  : in std_logic;
       out1 : out std_logic);

end GT_not_inst;

architecture sim of GT_not_inst is
begin
  U1 : GTECH_NOT
    port map( A => in1,
   Z => out1);
end sim;
```

Verilog

```
module GT_not_inst ( in1, out1 );
  input  in1;
  output out1;

  GTECH_NOT
    U1 (.A(in1), .Z(out1) );
endmodule
```



GTECH_BUF

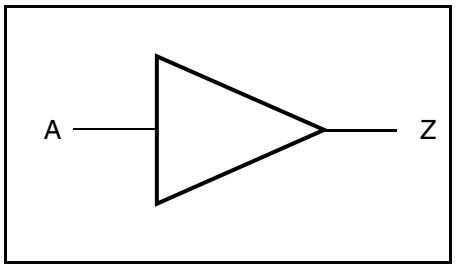
Non-Inverting Buffer

Truth Table

A	Z
0	0
1	1

Pin Description

Pin	Width	Direction
A	1	Input
Z	1	Output



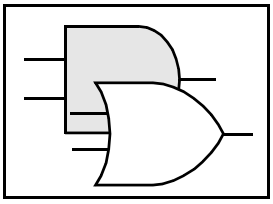
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_buf_inst is
  port(in1  : in std_logic;
       out1 : out std_logic);
end GT_buf_inst;

architecture sim of GT_buf_inst is
begin
  U1 : GTECH_BUF
    port map(A => in1,
             Z => out1);
end sim;
```

Verilog

```
module GT_buf_inst ( in1, out1 );
  input  in1;
  output out1;

  GTECH_BUF
    U1 (.A(in1), .Z(out1) );
endmodule
```



GTECH_AND2

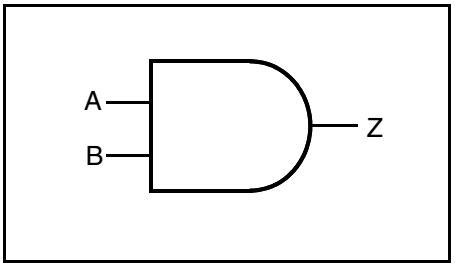
2-Input AND Gate

Truth Table

A	B	Z
0	X	0
X	0	0
1	1	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
Z	1	Output



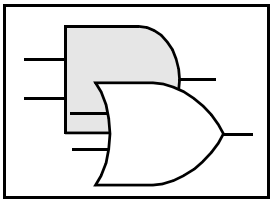
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_and2_inst is
  port(in1, in2: in std_logic;
       out1    : out std_logic);
end GT_and2_inst;

architecture sim of GT_and2_inst is
begin
  U1 : GTECH_AND2
    port map(A => in1, B => in2,
             Z => out1);
end sim;
```

Verilog

```
module GT_and2_inst (in1, in2,
                     out1);
  input  in1, in2;
  output out1;

  GTECH_AND2
    U1 (.A(in1), .B(in2),
        .Z(out1) );

endmodule
```



GTECH_AND3

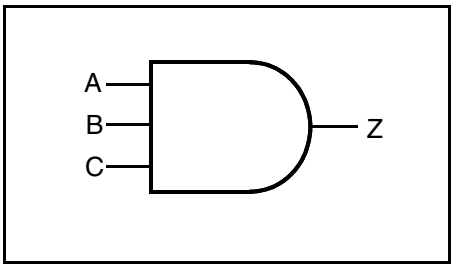
3-Input AND Gate

Truth Table

A	B	C	Z
0	X	X	0
X	0	X	0
X	X	0	0
1	1	1	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
Z	1	Output



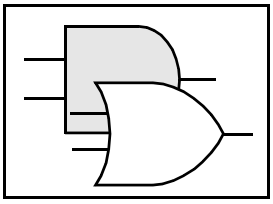
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_and3_inst is
  port(in1, in2, in3 : in std_logic;
       out1 : out std_logic);
end GT_and3_inst;

architecture sim of GT_and3_inst is
begin
  U1 : GTECH_AND3
    port map(A => in1, B => in2,
             C => in3, Z => out1);
end sim;
```

Verilog

```
module GT_and3_inst (in1, in2, in3,
                     out1 );
  input  in1, in2, in3;
  output out1;

  GTECH_AND3
    U1 (.A(in1), .B(in2),
        .C(in3), .Z(out1) );

endmodule
```



GTECH_AND4

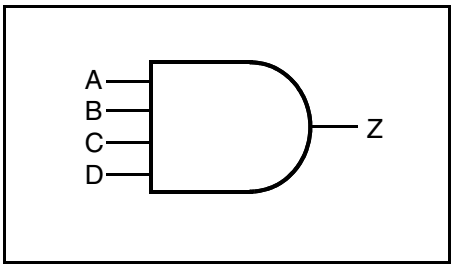
4-Input AND Gate

Truth Table

A	B	C	D	Z
0	X	X	X	0
X	0	X	X	0
X	X	0	X	0
X	X	X	0	0
1	1	1	1	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
Z	1	Output



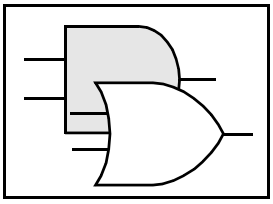
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_and4_inst is
  port(in1, in2, in3,
       in4  : in std_logic;
       out1 : out std_logic);
end GT_and4_inst;

architecture sim of GT_and4_inst is
begin
  U1 : GTECH_AND4
    port map(A => in1, B => in2,
             C => in3, D => in4,
             Z => out1);
end sim;
```

Verilog

```
module GT_and4_inst (in1, in2, in3,
                     in4, out1 );
  input in1, in2, in3, in4;
  output out1;

GTECH_AND4
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .Z(out1) );
endmodule
```



GTECH_AND5

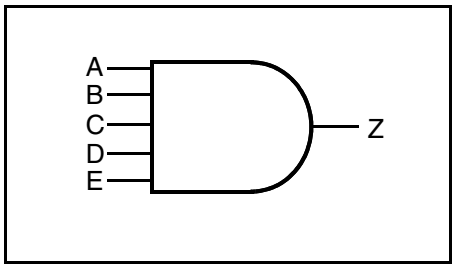
5-Input AND Gate

Truth Table

A	B	C	D	E	Z
0	X	X	X	X	0
X	0	X	X	X	0
X	X	0	X	X	0
X	X	X	0	X	0
X	X	X	X	0	0
1	1	1	1	1	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
E	1	Input
Z	1	Output



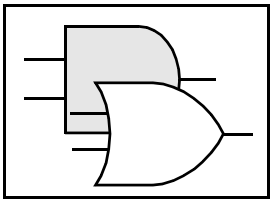
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_and5_inst is
  port(in1, in2, in3, in4
       in5  : in std_logic;
       out1 : out std_logic);
end GT_and5_inst;

architecture sim of GT_and5_inst is
begin
  U1 : GTECH_AND5
    port map(A => in1, B => in2,
             C => in3, D => in4,
             E => in5, Z => out1);
end sim;
```

Verilog

```
module GT_and5_inst (in1, in2, in3,
                     in4, in5, out1);
  input in1, in2, in3, in4, in5;
  output out1;

GTECH_AND5
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .E(in5), .Z(out1) );
endmodule
```



GTECH_AND8

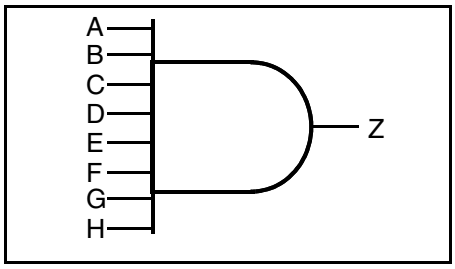
8-Input AND Gate

Truth Table

A	B	C	D	E	F	G	H	Z
0	X	X	X	X	X	X	X	0
X	0	X	X	X	X	X	X	0
X	X	0	X	X	X	X	X	0
X	X	X	0	X	X	X	X	0
X	X	X	X	0	X	X	X	0
X	X	X	X	X	0	X	X	0
X	X	X	X	X	X	0	X	0
X	X	X	X	X	X	X	0	0
1	1	1	1	1	1	1	1	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
E	1	Input
F	1	Input
G	1	Input
H	1	Input
Z	1	Output



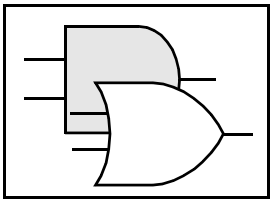
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_and8_inst is
  port(in1, in2, in3, in4, in5, in6,
       in7, in8 : in std_logic;
       out1 : out std_logic);
end GT_and8_inst;

architecture sim of GT_and8_inst is
begin
  U1 : GTECH_AND8
    port map(A => in1, B => in2,
             C => in3, D => in4,
             E => in5, F => in6,
             G => in7, H => in8,
             Z => out1);
end sim;
```

Verilog

```
module GT_and8_inst (in1, in2, in3,
                     in4, in5, in6,
                     in7, in8, out1);
  input in1, in2, in3, in4, in5,
        in6, in7, in8;
  output out1;

GTECH_AND8
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .E(in5), .F(in6),
      .G(in7), .H(in8), .Z(out1) ;
endmodule
```



GTECH_NAND2

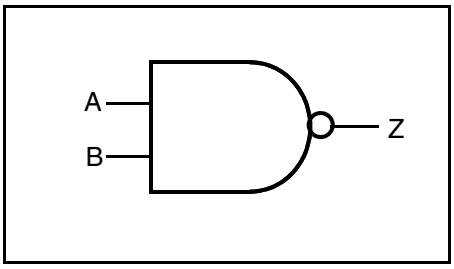
2-Input NAND Gate

Truth Table

A	B	Z
0	X	1
X	0	1
1	1	0

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
Z	1	Output



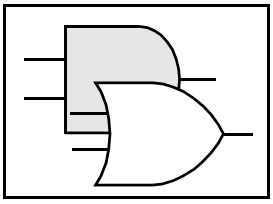
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_nand2_inst is
  port(in1, in2: in std_logic;
       out1    : out std_logic);
end GT_nand2_inst;

architecture sim of GT_nand2_inst is
begin
  U1 : GTECH_NAND2
    port map(A => in1, B => in2,
             Z => out1);
end sim;
```

Verilog

```
module GT_nand2_inst (in1, in2,
                      out1);
  input  in1, in2;
  output out1;

  GTECH_NAND2
    U1 (.A(in1), .B(in2),
        .Z(out1) );

endmodule
```



GTECH_NAND3

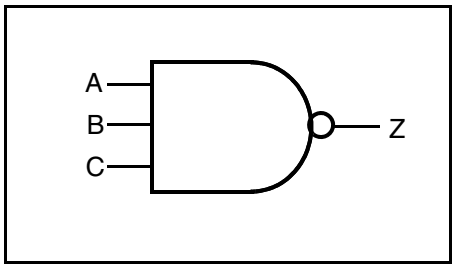
3-Input NAND Gate

Truth Table

A	B	C	Z
0	X	X	1
X	0	X	1
X	X	0	1
1	1	1	0

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
Z	1	Output



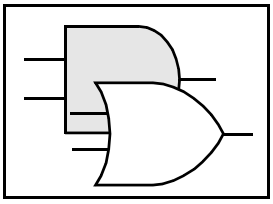
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_nand3_inst is
  port(in1, in2, in3 : in std_logic;
       out1 : out std_logic);
end GT_nand3_inst;

architecture sim of GT_nand3_inst is
begin
  U1 : GTECH_NAND3
    port map(A => in1, B => in2,
             C => in3, Z => out1);
end sim;
```

Verilog

```
module GT_nand3_inst (in1, in2, in3,
                      out1 );
  input  in1, in2, in3;
  output out1;

  GTECH_NAND3
    U1 (.A(in1), .B(in2),
        .C(in3), .Z(out1) );

endmodule
```



GTECH_NAND4

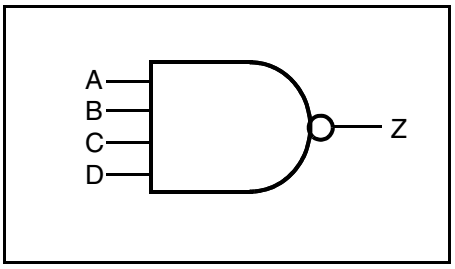
4-Input NAND Gate

Truth Table

A	B	C	D	Z
0	X	X	X	1
X	0	X	X	1
X	X	0	X	1
X	X	X	0	1
1	1	1	1	0

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
Z	1	Output



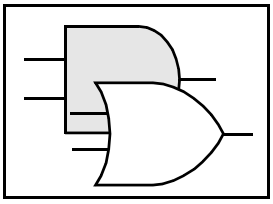
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_nand4_inst is
  port(in1, in2, in3,
       in4  : in std_logic;
       out1 : out std_logic);
end GT_nand4_inst;

architecture sim of GT_nand4_inst is
begin
  U1 : GTECH_NAND4
    port map(A => in1, B => in2,
             C => in3, D => in4,
             Z => out1);
end sim;
```

Verilog

```
module GT_nand4_inst (in1, in2, in3,
                      in4, out1 );
  input in1, in2, in3, in4;
  output out1;

GTECH_NAND4
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .Z(out1) );
endmodule
```



GTECH_NAND5

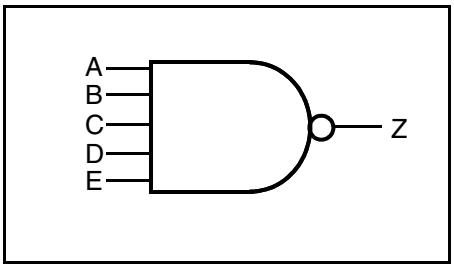
5-Input NAND Gate

Truth Table

A	B	C	D	E	Z
0	X	X	X	X	1
X	0	X	X	X	1
X	X	0	X	X	1
X	X	X	0	X	1
X	X	X	X	0	1
1	1	1	1	1	0

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
E	1	Input
Z	1	Output



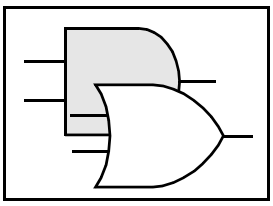
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_nand5_inst is
  port(in1, in2, in3, in4
       in5  : in std_logic;
       out1 : out std_logic);
end GT_nand5_inst;

architecture sim of GT_nand5_inst is
begin
  U1 : GTECH_NAND5
    port map(A => in1, B => in2,
             C => in3, D => in4,
             E => in5, Z => out1);
end sim;
```

Verilog

```
module GT_nand5_inst (in1, in2, in3,
                      in4, in5, out1);
  input in1, in2, in3, in4, in5;
  output out1;

GTECH_NAND5
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .E(in5), .Z(out1) );
endmodule
```



GTECH_NAND8

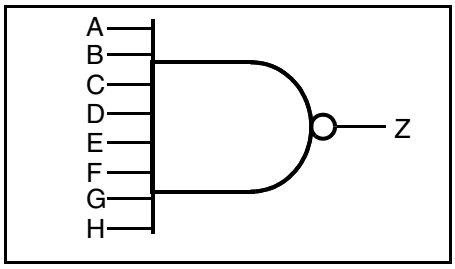
8-Input NAND Gate

Truth Table

A	B	C	D	E	F	G	H	Z
0	X	X	X	X	X	X	X	1
X	0	X	X	X	X	X	X	1
X	X	0	X	X	X	X	X	1
X	X	X	0	X	X	X	X	1
X	X	X	X	0	X	X	X	1
X	X	X	X	X	0	X	X	1
X	X	X	X	X	X	0	X	1
X	X	X	X	X	X	X	0	1
1	1	1	1	1	1	1	1	0

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
E	1	Input
F	1	Input
G	1	Input
H	1	Input
Z	1	Output



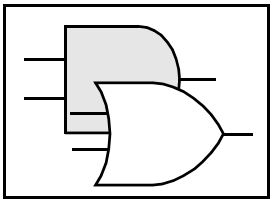
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_nand8_inst is
  port(in1, in2, in3, in4, in5, in6,
       in7, in8 : in std_logic;
       out1 : out std_logic);
end GT_nand8_inst;

architecture sim of GT_nand8_inst is
begin
  U1 : GTECH_NAND8
    port map(A => in1, B => in2,
             C => in3, D => in4,
             E => in5, F => in6,
             G => in7, H => in8,
             Z => out1);
end sim;
```

Verilog

```
module GT_nand8_inst (in1, in2, in3,
                      in4, in5, in6,
                      in7, in8, out1);
  input in1, in2, in3, in4, in5,
        in6, in7, in8;
  output out1;

GTECH_NAND8
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .E(in5), .F(in6),
      .G(in7), .H(in8), .Z(out1) ;
endmodule
```



GTECH_OR2

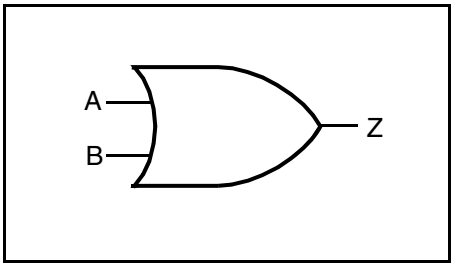
2-Input OR Gate

Truth Table

A	B	Z
0	0	0
1	X	1
X	1	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
Z	1	Output



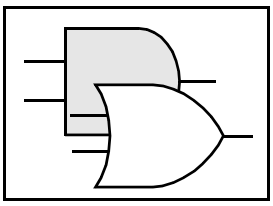
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_or2_inst is
  port(in1, in2: in std_logic;
       out1    : out std_logic);
end GT_or2_inst;

architecture sim of GT_or2_inst is
begin
  U1 : GTECH_OR2
    port map(A => in1, B => in2,
             Z => out1);
end sim;
```

Verilog

```
module GT_or2_inst (in1, in2,
                    out1);
  input  in1, in2;
  output out1;

  GTECH_OR2
    U1 (.A(in1), .B(in2),
        .Z(out1) );

endmodule
```



GTECH_OR3

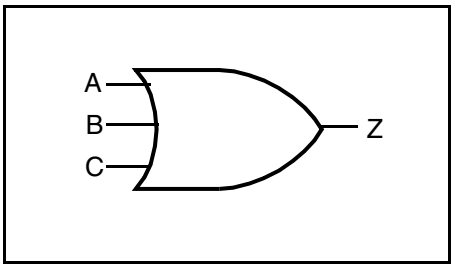
3-Input OR Gate

Truth Table

A	B	C	Z
0	0	0	0
1	X	X	1
X	1	X	1
X	X	1	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
Z	1	Output



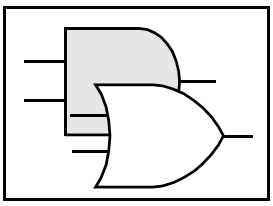
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_or3_inst is
  port(in1, in2, in3 : in std_logic;
       out1 : out std_logic);
end GT_or3_inst;

architecture sim of GT_or3_inst is
begin
  U1 : GTECH_OR3
    port map(A => in1, B => in2,
             C => in3, Z => out1);
end sim;
```

Verilog

```
module GT_or3_inst (in1, in2, in3,
                    out1 );
  input  in1, in2, in3;
  output out1;

  GTECH_OR3
    U1 (.A(in1), .B(in2),
        .C(in3), .Z(out1) );

endmodule
```



GTECH_OR4

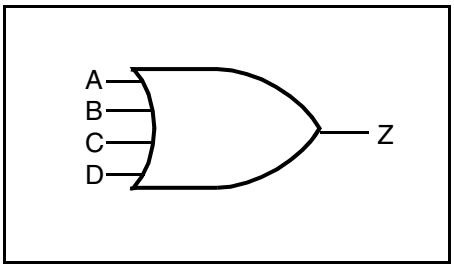
4-Input OR Gate

Truth Table

A	B	C	D	Z
0	0	0	0	0
1	X	X	X	1
X	1	X	X	1
X	X	1	X	1
X	X	X	1	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
Z	1	Output



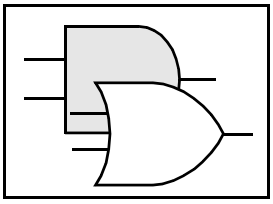
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_or4_inst is
  port(in1, in2, in3,
       in4  : in std_logic;
       out1 : out std_logic);
end GT_or4_inst;

architecture sim of GT_or4_inst is
begin
  U1 : GTECH_OR4
    port map(A => in1, B => in2,
             C => in3, D => in4,
             Z => out1);
end sim;
```

Verilog

```
module GT_or4_inst (in1, in2, in3,
                    in4, out1 );
  input in1, in2, in3, in4;
  output out1;

GTECH_OR4
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .Z(out1) );
endmodule
```



GTECH_OR5

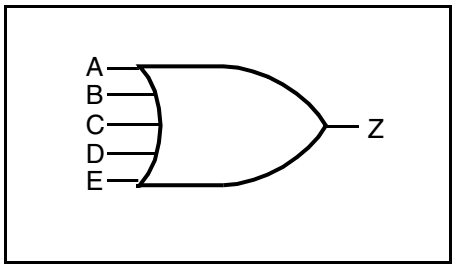
5-Input OR Gate

Truth Table

A	B	C	D	E	Z
0	0	0	0	0	0
1	X	X	X	X	1
X	1	X	X	X	1
X	X	1	X	X	1
X	X	X	1	X	1
X	X	X	X	1	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
E	1	Input
Z	1	Output



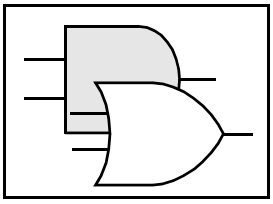
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_or5_inst is
  port(in1, in2, in3, in4
       in5  : in std_logic;
       out1 : out std_logic);
end GT_or5_inst;

architecture sim of GT_or5_inst is
begin
  U1 : GTECH_OR5
    port map(A => in1, B => in2,
             C => in3, D => in4,
             E => in5, Z => out1);
end sim;
```

Verilog

```
module GT_or5_inst (in1, in2, in3,
                    in4, in5, out1);
  input in1, in2, in3, in4, in5;
  output out1;

GTECH_OR5
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .E(in5), .Z(out1) );
endmodule
```



GTECH_OR8

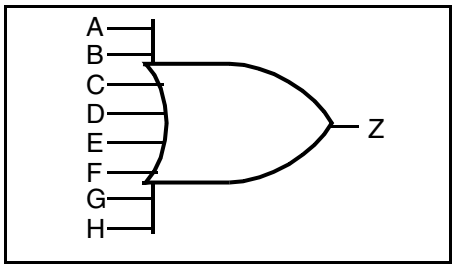
8-Input OR Gate

Truth Table

A	B	C	D	E	F	G	H	Z
0	0	0	0	0	0	0	0	0
1	X	X	X	X	X	X	X	1
X	1	X	X	X	X	X	X	1
X	X	1	X	X	X	X	X	1
X	X	X	1	X	X	X	X	1
X	X	X	X	1	X	X	X	1
X	X	X	X	X	1	X	X	1
X	X	X	X	X	X	1	X	1
X	X	X	X	X	X	X	1	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
E	1	Input
F	1	Input
G	1	Input
H	1	Input
Z	1	Output



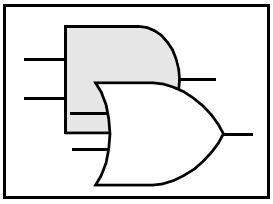
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_or8_inst is
  port(in1, in2, in3, in4, in5, in6,
       in7, in8 : in std_logic;
       out1 : out std_logic);
end GT_or8_inst;

architecture sim of GT_or8_inst is
begin
  U1 : GTECH_OR8
    port map(A => in1, B => in2,
             C => in3, D => in4,
             E => in5, F => in6,
             G => in7, H => in8,
             Z => out1);
```

Verilog

```
module GT_or8_inst (in1, in2, in3,
                    in4, in5, in6,
                    in7, in8, out1);
  input in1, in2, in3, in4, in5,
        in6, in7, in8;
  output out1;

GTECH_OR8
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .E(in5), .F(in6),
      .G(in7), .H(in8), .Z(out1) ;
endmodule
```



GTECH_NOR2

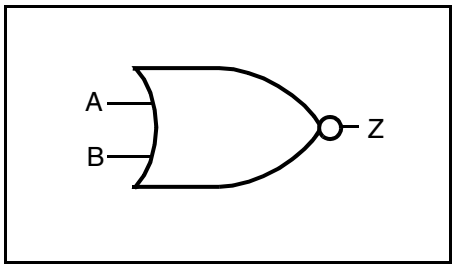
2-Input NOR Gate

Truth Table

A	B	Z
0	0	1
1	X	0
X	1	0

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
Z	1	Output



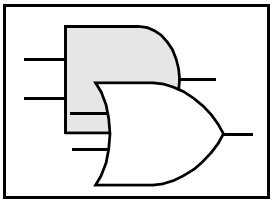
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_nor2_inst is
  port(in1, in2: in std_logic;
       out1    : out std_logic);
end GT_nor2_inst;

architecture sim of GT_nor2_inst is
begin
  U1 : GTECH_NOR2
    port map(A => in1, B => in2,
             Z => out1);
end sim;
```

Verilog

```
module GT_nor2_inst (in1, in2,
                     out1);
  input  in1, in2;
  output out1;

  GTECH_NOR2
    U1 (.A(in1), .B(in2),
        .Z(out1) );

endmodule
```



GTECH_NOR3

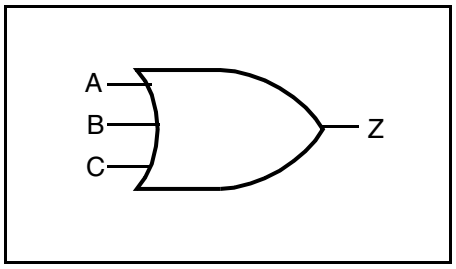
3-Input NOR Gate

Truth Table

A	B	C	Z
0	0	0	1
1	X	X	0
X	1	X	0
X	X	1	0

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
Z	1	Output



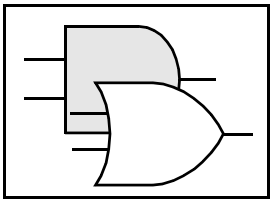
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_nor3_inst is
  port(in1, in2, in3 : in std_logic;
       out1 : out std_logic);
end GT_nor3_inst;

architecture sim of GT_nor3_inst is
begin
  U1 : GTECH_NOR3
    port map(A => in1, B => in2,
             C => in3, Z => out1);
end sim;
```

Verilog

```
module GT_nor3_inst (in1, in2, in3,
                     out1 );
  input  in1, in2, in3;
  output out1;

  GTECH_NOR3
    U1 (.A(in1), .B(in2),
        .C(in3), .Z(out1) );

endmodule
```



GTECH_NOR4

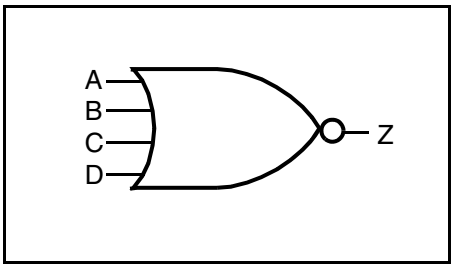
4-Input NOR Gate

Truth Table

A	B	C	D	Z
0	0	0	0	1
1	X	X	X	0
X	1	X	X	0
X	X	1	X	0
X	X	X	1	0

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
Z	1	Output



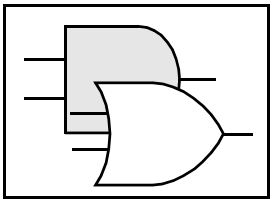
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_nor4_inst is
  port(in1, in2, in3,
       in4  : in std_logic;
       out1 : out std_logic);
end GT_nor4_inst;

architecture sim of GT_nor4_inst is
begin
  U1 : GTECH_NOR4
    port map(A => in1, B => in2,
             C => in3, D => in4,
             Z => out1);
end sim;
```

Verilog

```
module GT_nor4_inst (in1, in2, in3,
                     in4, out1 );
  input in1, in2, in3, in4;
  output out1;

GTECH_NOR4
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .Z(out1) );
endmodule
```



GTECH_NOR5

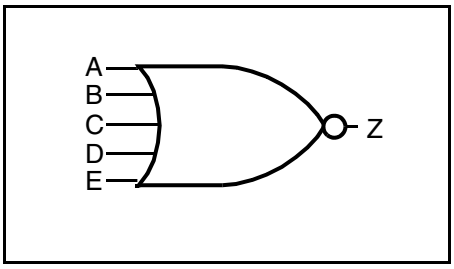
5-Input NOR Gate

Truth Table

A	B	C	D	E	Z
0	0	0	0	0	1
1	X	X	X	X	0
X	1	X	X	X	0
X	X	1	X	X	0
X	X	X	1	X	0
X	X	X	X	1	0

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
E	1	Input
Z	1	Output



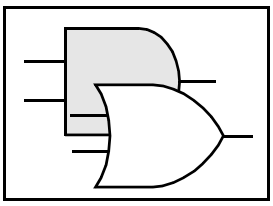
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_nor5_inst is
  port(in1, in2, in3, in4
       in5  : in std_logic;
       out1 : out std_logic);
end GT_nor5_inst;

architecture sim of GT_nor5_inst is
begin
  U1 : GTECH_NOR5
    port map(A => in1, B => in2,
             C => in3, D => in4,
             E => in5, Z => out1);
end sim;
```

Verilog

```
module GT_nor5_inst (in1, in2, in3,
                     in4, in5, out1);
  input in1, in2, in3, in4, in5;
  output out1;

GTECH_NOR5
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .E(in5), .Z(out1) );
endmodule
```



GTECH_NOR8

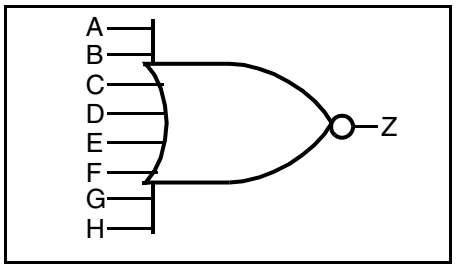
8-Input NOR Gate

Truth Table

A	B	C	D	E	F	G	H	Z
0	0	0	0	0	0	0	0	1
1	X	X	X	X	X	X	X	0
X	1	X	X	X	X	X	X	0
X	X	1	X	X	X	X	X	0
X	X	X	1	X	X	X	X	0
X	X	X	X	1	X	X	X	0
X	X	X	X	X	1	X	X	0
X	X	X	X	X	X	1	X	0
X	X	X	X	X	X	X	1	0

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
E	1	Input
F	1	Input
G	1	Input
H	1	Input
Z	1	Output



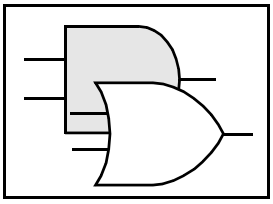
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_nor8_inst is
  port(in1, in2, in3, in4, in5, in6,
       in7, in8 : in std_logic;
       out1 : out std_logic);
end GT_nor8_inst;

architecture sim of GT_nor8_inst is
begin
  U1 : GTECH_NOR8
    port map(A => in1, B => in2,
             C => in3, D => in4,
             E => in5, F => in6,
             G => in7, H => in8,
             Z => out1);
```

Verilog

```
module GT_nor8_inst (in1, in2, in3,
                     in4, in5, in6,
                     in7, in8, out1);
  input in1, in2, in3, in4, in5,
        in6, in7, in8;
  output out1;

GTECH_NOR8
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .E(in5), .F(in6),
      .G(in7), .H(in8), .Z(out1) ;
endmodule
```



GTECH_XOR2

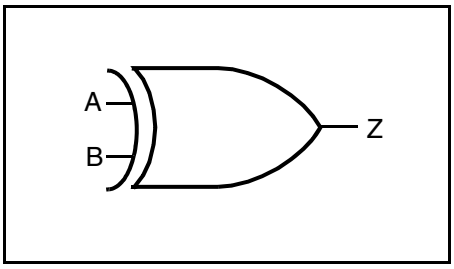
2-Input XOR Gate

Truth Table

A	B	Z
0	0	0
0	1	1
1	0	1
1	1	0

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
Z	1	Output



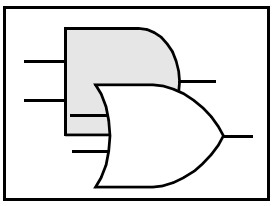
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_xor2_inst is
  port(in1, in2: in std_logic;
       out1    : out std_logic);
end GT_xor2_inst;

architecture sim of GT_xor2_inst is
begin
  U1 : GTECH_XOR2
    port map(A => in1, B => in2,
             Z => out1);
end sim;
```

Verilog

```
module GT_xor2_inst (in1, in2,
                     out1);
  input  in1, in2;
  output out1;

  GTECH_XOR2
    U1 (.A(in1), .B(in2),
        .Z(out1) );

endmodule
```



GTECH_XOR3

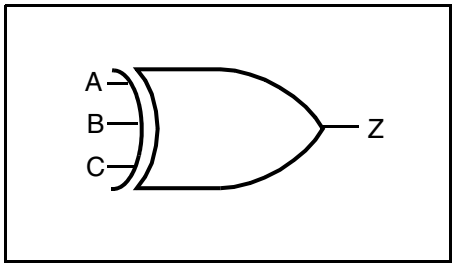
3-Input XOR Gate

Truth Table

A	B	C	Z
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

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
Z	1	Output



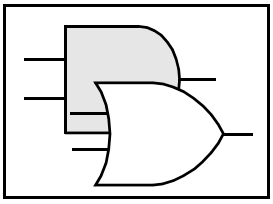
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_xor3_inst is
  port(in1, in2, in3 : in std_logic;
       out1 : out std_logic);
end GT_xor3_inst;

architecture sim of GT_xor3_inst is
begin
  U1 : GTECH_XOR3
    port map(A => in1, B => in2,
             C => in3, Z => out1);
end sim;
```

Verilog

```
module GT_xor3_inst (in1, in2, in3,
                     out1 );
  input  in1, in2, in3;
  output out1;

  GTECH_XOR3
    U1 (.A(in1), .B(in2),
        .C(in3), .Z(out1) );

endmodule
```

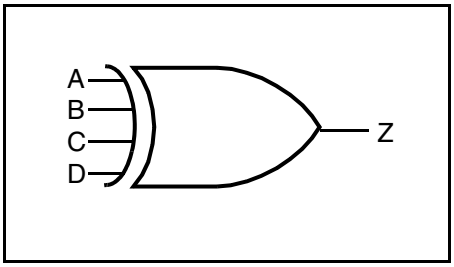


GTECH_XOR4

4-Input XOR Gate

Truth Table

A	B	C	D	Z	A	B	C	D	Z
0	0	0	0	0	1	0	0	0	1
0	0	0	1	1	1	0	0	1	0
0	0	1	0	1	1	0	1	0	0
0	0	1	1	0	1	0	1	1	1
0	1	0	0	1	1	1	0	0	0
0	1	0	1	0	1	1	0	1	1
0	1	1	0	0	1	1	1	0	1
0	1	1	1	1	1	1	1	1	0



Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
Z	1	Output

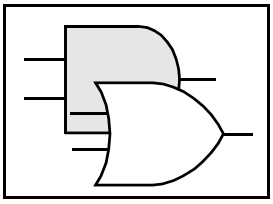
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_xor4_inst is
  port(in1, in2, in3,
       in4  : in std_logic;
       out1 : out std_logic);
end GT_xor4_inst;

architecture sim of GT_xor4_inst is
begin
  U1 : GTECH_XOR4
    port map(A => in1, B => in2,
             C => in3, D => in4,
             Z => out1);
```

Verilog

```
module GT_xor4_inst (in1, in2, in3,
                     in4, out1 );
  input in1, in2, in3, in4;
  output out1;

GTECH_XOR4
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .Z(out1) );
endmodule
```



GTECH_XNOR2

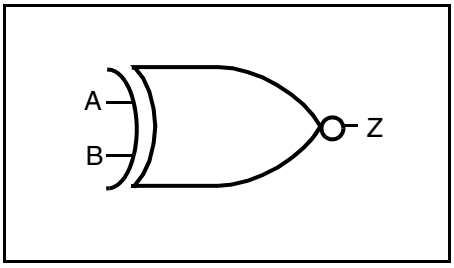
2-Input Exclusive-NOR Gate

Truth Table

A	B	Z
0	0	1
0	1	0
1	0	0
1	1	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
Z	1	Output



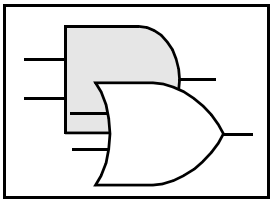
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_xnor2_inst is
  port(in1, in2: in std_logic;
       out1    : out std_logic);
end GT_xnor2_inst;

architecture sim of GT_xnor2_inst is
begin
  U1 : GTECH_XNOR2
    port map(A => in1, B => in2,
             Z => out1);
end sim;
```

Verilog

```
module GT_xnor2_inst (in1, in2,
                      out1);
  input  in1, in2;
  output out1;

  GTECH_XNOR2
    U1 (.A(in1), .B(in2),
        .Z(out1) );

endmodule
```



GTECH_XNOR3

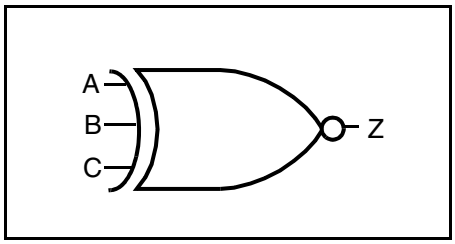
3-Input Exclusive-NOR Gate

Truth Table

A	B	C	Z
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

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
Z	1	Output



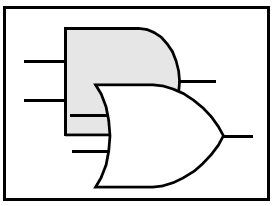
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_xnor3_inst is
  port(in1, in2, in3 : in std_logic;
       out1 : out std_logic);
end GT_xnor3_inst;

architecture sim of GT_xnor3_inst is
begin
  U1 : GTECH_XNOR3
    port map(A => in1, B => in2,
             C => in3, Z => out1);
end sim;
```

Verilog

```
module GT_xnor3_inst (in1, in2, in3,
                      out1 );
  input  in1, in2, in3;
  output out1;

  GTECH_XNOR3
    U1 (.A(in1), .B(in2),
        .C(in3), .Z(out1) );

endmodule
```

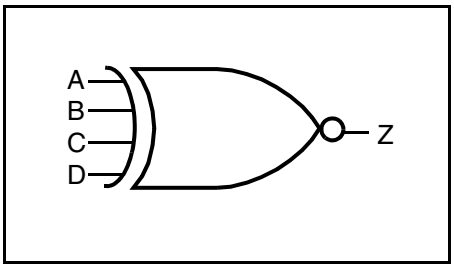


GTECH_XNOR4

4-Input Exclusive-OR Gate

Truth Table

A	B	C	D	Z	A	B	C	D	Z
0	0	0	0	1	1	0	0	0	0
0	0	0	1	0	1	0	0	1	1
0	0	1	0	0	1	0	1	0	1
0	0	1	1	1	1	0	1	1	0
0	1	0	0	0	1	1	0	0	1
0	1	0	1	1	1	1	0	1	0
0	1	1	0	1	1	1	1	0	0
0	1	1	1	0	1	1	1	1	1



Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
Z	1	Output

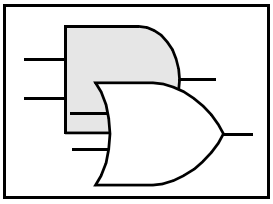
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_xnor4_inst is
  port(in1, in2, in3,
       in4  : in std_logic;
       out1 : out std_logic);
end GT_xnor4_inst;

architecture sim of GT_xnor4_inst is
begin
  U1 : GTECH_XNOR4
    port map(A => in1, B => in2,
             C => in3, D => in4,
             Z => out1);
```

Verilog

```
module GT_xnor4_inst (in1, in2, in3,
                      in4, out1 );
  input in1, in2, in3, in4;
  output out1;

GTECH_XNOR4
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .Z(out1) );
endmodule
```



GTECH_AND_NOT

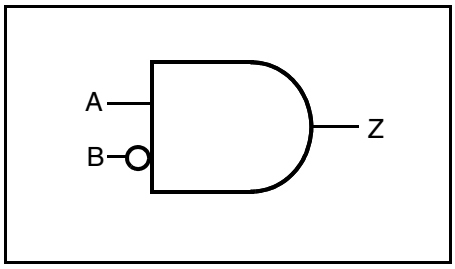
2-Input AND Gate, Inverted Input

Truth Table

A	B	Z
0	X	0
X	1	0
1	0	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
Z	1	Output



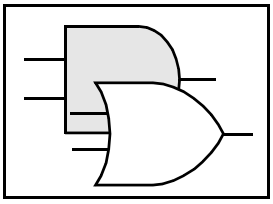
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_and_not_inst is
  port(in1, in2: in std_logic;
       out1    : out std_logic);
end GT_and_not_inst;

architecture sim of GT_and_not_inst
is
begin
  U1 : GTECH_AND2
    port map(A => in1, B => in2,
             Z => out1);
end sim;
```

Verilog

```
module GT_and_not_inst (in1, in2,
                        out1);
  input  in1, in2;
  output out1;

  GTECH_AND_NOT
    U1 (.A(in1), .B(in2),
        .Z(out1) );

endmodule
```



GTECH_OR_NOT

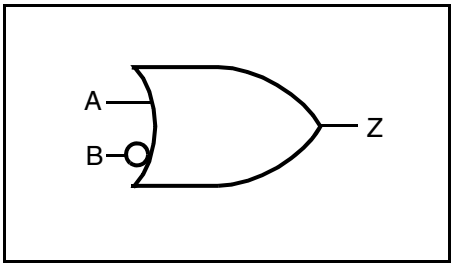
2-Input OR Gate, Inverted Input

Truth Table

A	B	Z
X	0	1
1	X	1
0	1	0

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
Z	1	Output



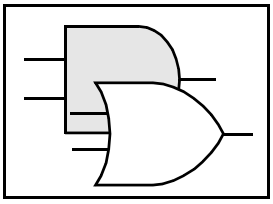
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_or_not_inst is
  port(in1, in2: in std_logic;
       out1    : out std_logic);
end GT_or_not_inst;

architecture sim of GT_or_not_inst is
begin
  U1 : GTECH_OR_NOT
    port map(A => in1, B => in2,
             Z => out1);
end sim;
```

Verilog

```
module GT_or_not_inst (in1, in2,
                       out1);
  input  in1, in2;
  output out1;

  GTECH_OR_NOT
    U1 (.A(in1), .B(in2),
        .Z(out1) );

endmodule
```



GTECH_AO21

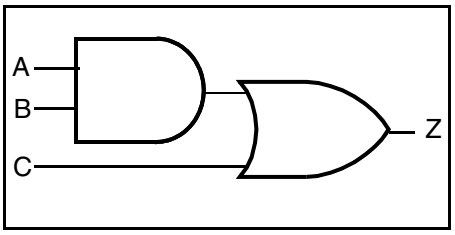
2-Input AND into 2-input OR

Truth Table

A	B	C	Z
0	X	0	0
X	0	0	0
X	X	1	1
1	1	X	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
Z	1	Output



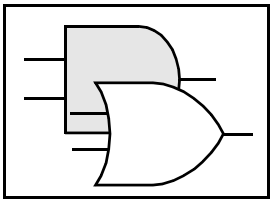
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_ao21_inst is
  port(in1, in2, in3 : in std_logic;
       out1 : out std_logic);
end GT_ao21_inst;

architecture sim of GT_ao21_inst is
begin
  U1 : GTECH_AO21
    port map(A => in1, B => in2,
             C => in3, Z => out1);
end sim;
```

Verilog

```
module GT_ao21_inst (in1, in2, in3,
                     out1 );
  input  in1, in2, in3;
  output out1;

  GTECH_AO21
    U1 (.A(in1), .B(in2),
        .C(in3), .Z(out1) );

endmodule
```



GTECH_OA21

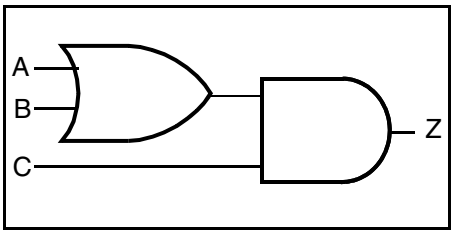
2-Input OR into 2-input AND

Truth Table

A	B	C	Z
0	0	X	0
X	X	0	0
X	1	1	1
1	X	1	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
Z	1	Output



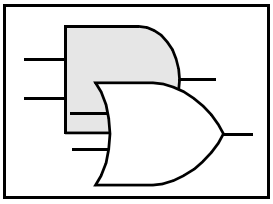
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_oa21_inst is
  port(in1, in2, in3 : in std_logic;
       out1 : out std_logic);
end GT_oa21_inst;

architecture sim of GT_oa21_inst is
begin
  U1 : GTECH_OA21
    port map(A => in1, B => in2,
             C => in3, Z => out1);
end sim;
```

Verilog

```
module GT_oa21_inst (in1, in2, in3,
                     out1 );
  input  in1, in2, in3;
  output out1;

  GTECH_OA21
    U1 (.A(in1), .B(in2),
        .C(in3), .Z(out1) );

endmodule
```



GTECH_AO22

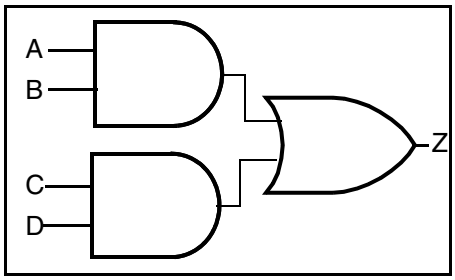
2-Input ANDs into One 2-input OR

Truth Table

A	B	C	D	Z
0	X	X	0	0
0	X	0	X	0
X	0	0	X	0
X	0	X	0	0
1	1	X	X	1
X	X	1	1	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
Z	1	Output



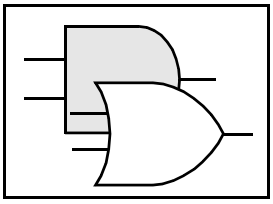
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_ao22_inst is
  port(in1, in2, in3,
       in4  : in std_logic;
       out1 : out std_logic);
end GT_ao22_inst;

architecture sim of GT_ao22_inst is
begin
  U1 : GTECH_AO22
    port map(A => in1, B => in2,
             C => in3, D => in4,
             Z => out1);
end sim;
```

Verilog

```
module GT_ao22_inst (in1, in2, in3,
                     in4, out1 );
  input in1, in2, in3, in4;
  output out1;

GTECH_AO22
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .Z(out1) );
endmodule
```



GTECH_OA22

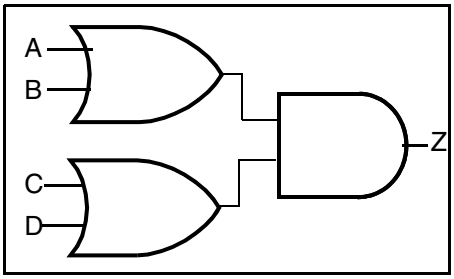
2-Input ORs into One 2-input AND

Truth Table

A	B	C	D	Z
0	0	X	X	0
X	X	0	0	0
X	1	1	X	1
X	1	X	1	1
1	X	X	1	1
1	X	1	X	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
Z	1	Output



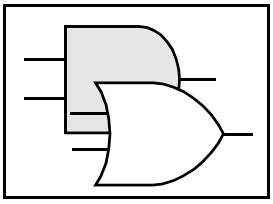
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_oa22_inst is
  port(in1, in2, in3,
       in4  : in std_logic;
       out1 : out std_logic);
end GT_oa22_inst;

architecture sim of GT_oa22_inst is
begin
  U1 : GTECH_OA22
    port map(A => in1, B => in2,
             C => in3, D => in4,
             Z => out1);
end sim;
```

Verilog

```
module GT_oa22_inst (in1, in2, in3,
                     in4, out1 );
  input in1, in2, in3, in4;
  output out1;

GTECH_OA22
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .Z(out1) );
endmodule
```



GTECH_AO21

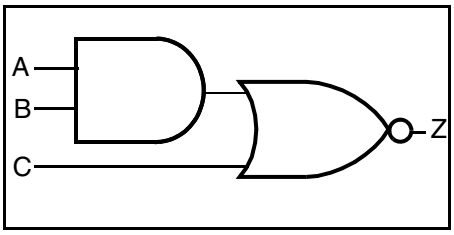
2-Input AND into 2-input NOR

Truth Table

A	B	C	Z
0	X	0	1
X	0	0	1
X	X	1	0
1	1	X	0

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
Z	1	Output



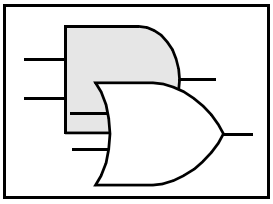
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_aoi21_inst is
  port(in1, in2, in3 : in std_logic;
       out1 : out std_logic);
end GT_aoi21_inst;

architecture sim of GT_aoi21_inst is
begin
  U1 : GTECH_AOI21
    port map(A => in1, B => in2,
             C => in3, Z => out1);
end sim;
```

Verilog

```
module GT_aoi21_inst (in1, in2, in3,
                      out1 );
  input  in1, in2, in3;
  output out1;

  GTECH_AOI21
    U1 (.A(in1), .B(in2),
        .C(in3), .Z(out1) );

endmodule
```



GTECH_OAI21

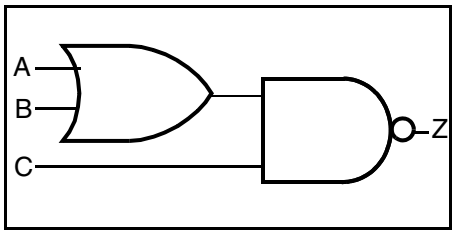
2-Input OR into 2-input NAND

Truth Table

A	B	C	Z
0	0	X	1
X	X	0	1
X	1	1	0
1	X	1	0

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
Z	1	Output



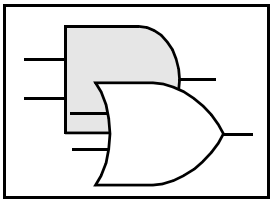
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_oai21_inst is
  port(in1, in2, in3 : in std_logic;
       out1 : out std_logic);
end GT_oai21_inst;

architecture sim of GT_oai21_inst is
begin
  U1 : GTECH_OAI21
    port map(A => in1, B => in2,
             C => in3, Z => out1);
end sim;
```

Verilog

```
module GT_oai21_inst (in1, in2, in3,
                      out1 );
  input  in1, in2, in3;
  output out1;

  GTECH_OAI21
    U1 (.A(in1), .B(in2),
        .C(in3), .Z(out1) );

endmodule
```



GTECH_AOI22

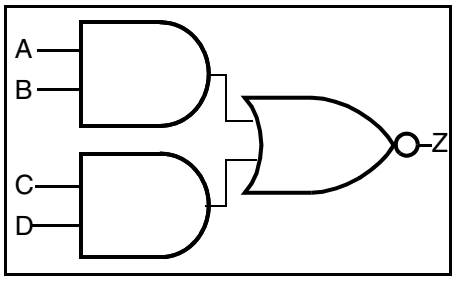
2-Input ANDs into One 2-input NOR

Truth Table

A	B	C	D	Z
0	X	X	0	1
0	X	0	X	1
X	0	0	X	1
X	0	X	0	1
1	1	X	X	0
1	1	X	X	0

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
Z	1	Output



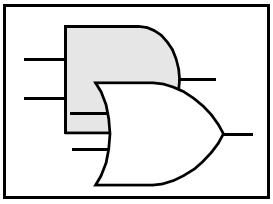
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_aoi22_inst is
  port(in1, in2, in3,
       in4  : in std_logic;
       out1 : out std_logic);
end GT_aoi22_inst;

architecture sim of GT_aoi22_inst is
begin
  U1 : GTECH_AOI22
    port map(A => in1, B => in2,
             C => in3, D => in4,
             Z => out1);
end sim;
```

Verilog

```
module GT_aoi22_inst (in1, in2, in3,
                      in4, out1 );
  input in1, in2, in3, in4;
  output out1;

GTECH_AOI22
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .Z(out1) );
endmodule
```



GTECH_OAI22

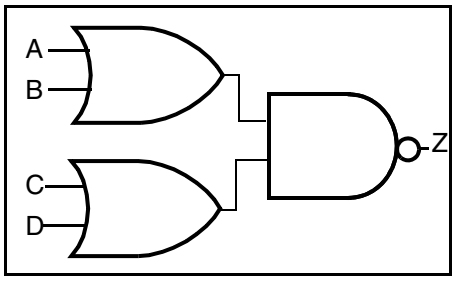
2-Input ORs into One 2-input NAND

Truth Table

A	B	C	D	Z
0	0	X	X	1
X	X	0	0	1
X	1	1	X	0
X	1	X	1	0
1	X	X	1	0
1	X	1	X	0

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
Z	1	Output



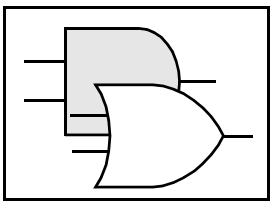
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_oai22_inst is
  port(in1, in2, in3,
       in4  : in std_logic;
       out1 : out std_logic);
end GT_oai22_inst;

architecture sim of GT_oai22_inst is
begin
  U1 : GTECH_OAI22
    port map(A => in1, B => in2,
             C => in3, D => in4,
             Z => out1);
end sim;
```

Verilog

```
module GT_oai22_inst (in1, in2, in3,
                      in4, out1 );
  input in1, in2, in3, in4;
  output out1;

GTECH_OAI22
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .Z(out1) );
endmodule
```



GTECH_AOI222

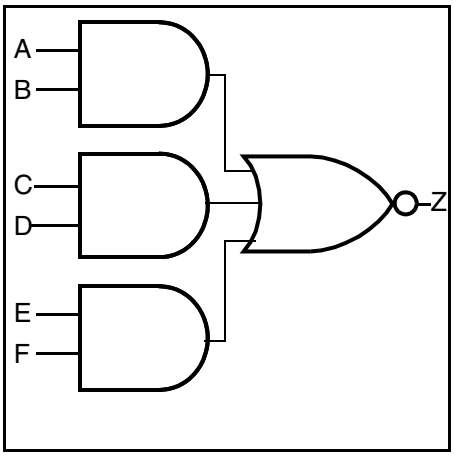
Three 2-Input ANDs into One 3-input NOR

Truth Table

A	B	C	D	E	F	Z
X	0	X	0	X	0	1
X	0	X	0	0	X	1
X	0	0	X	X	0	1
X	0	0	X	0	X	1
0	X	X	0	X	0	1
0	X	X	0	0	X	1
0	X	0	X	X	0	1
0	X	0	X	0	X	1
X	X	X	X	1	1	0
X	X	1	1	X	X	0
1	1	X	X	X	X	0

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
Z	1	Output



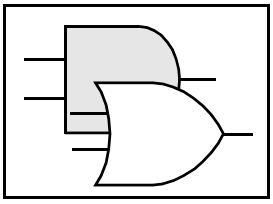
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_aoi222_inst is
  port(in1, in2, in3, in4,
       in5, in6 : in std_logic;
       out1 : out std_logic);
end GT_aoi222_inst;

architecture sim of GT_aoi222_inst
is
begin
  U1 : GTECH_AOI222
    port map(A => in1, B => in2,
             C => in3, D => in4,
             E => in5, F => in6,
             Z => out1);
end sim;
```

Verilog

```
module GT_aoi222_inst (in1, in2,
in3,
                in4, in5, in6, out1
);
  input in1, in2, in3, in4, in5,
in6;
  output out1;

GTECH_AOI222
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .E(in5), .F(in6),
```



GTECH_AOI2N2

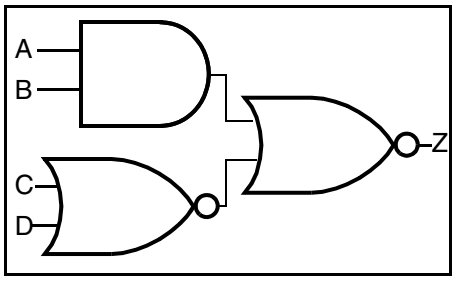
2-Input AND and 2-Input NOR into One 2-input NOR

Truth Table

A	B	C	D	Z
1	1	X	X	0
X	X	0	0	0
0	X	X	1	1
X	0	X	1	1
0	X	1	X	1
X	0	1	X	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
Z	1	Output



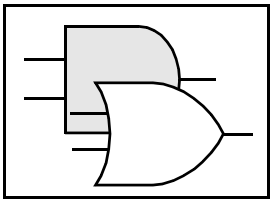
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_aoi2n2_inst is
  port(in1, in2, in3,
       in4  : in std_logic;
       out1 : out std_logic);
end GT_aoi2n2_inst;

architecture sim of GT_aoi2n2_inst is
begin
  U1 : GTECH_AOI2N2
    port map(A => in1, B => in2,
             C => in3, D => in4,
             Z => out1);
end sim;
```

Verilog

```
module GT_aoi2n2_inst (in1, in2,
                       in3, in4, out1
);
  input in1, in2, in3, in4;
  output out1;

GTECH_AOI2N2
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .Z(out1) );
endmodule
```



GTECH_OAI2N2

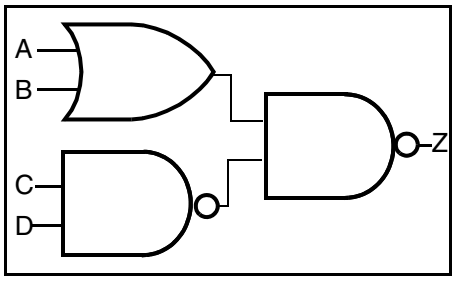
2-Input OR and 2-Input NAND into One 2-input NAND

Truth Table

A	B	C	D	Z
1	X	0	X	0
X	1	0	X	0
1	X	X	0	0
X	1	X	0	0
0	0	X	X	1
X	X	1	1	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
D	1	Input
Z	1	Output



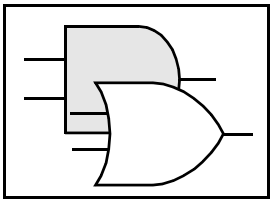
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_oai2n2_inst is
  port(in1, in2, in3,
       in4  : in std_logic;
       out1 : out std_logic);
end GT_oai2n2_inst;

architecture sim of GT_oai2n2_inst is
begin
  U1 : GTECH_OAI2N2
    port map(A => in1, B => in2,
             C => in3, D => in4,
             Z => out1);
end sim;
```

Verilog

```
module GT_oai2n2_inst (in1, in2,
                       in3, in4, out1
);
  input in1, in2, in3, in4;
  output out1;

GTECH_OAI2N2
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .Z(out1) );
endmodule
```



GTECH_MAJ23

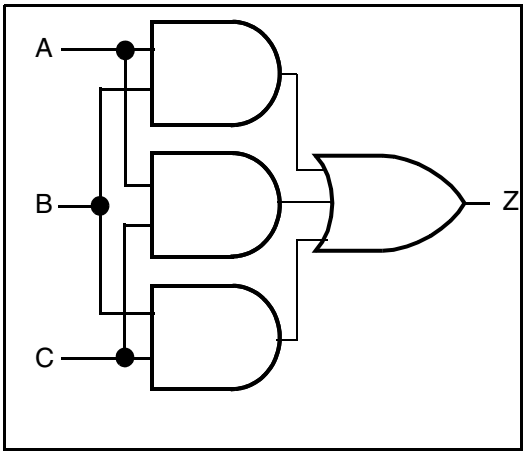
Two-of-Three Majority Function

Truth Table

A	B	C	Z
0	0	X	0
0	X	0	0
X	0	0	0
X	1	1	1
1	X	1	1
1	1	X	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
Z	1	Output

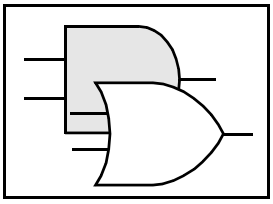


HDL Usage Through Instantiation

```
use GTECH.GTECH_components.all;

entity GT_maj23_inst is
  port(in1, in2, in3, in4,
       in5, in6 : in std_logic;
       out1 : out std_logic);
end GT_maj23_inst;

architecture sim of GT_maj23_inst
is
begin
  U1 : GTECH_MAJ23
    port map(A => in1, B => in2,
             C => in3, Z => out1);
end sim;
```

Verilog

```
module GT_maj23_inst (in1, in2, in3,
                      out1 );
  input in1, in2, in3;
  output out1;

GTECH_MAJ23
  U1 (.A(in1), .B(in2), .C(in3),
      .D(in4), .Z(out1) );
endmodule
```



GTECH_MUX2

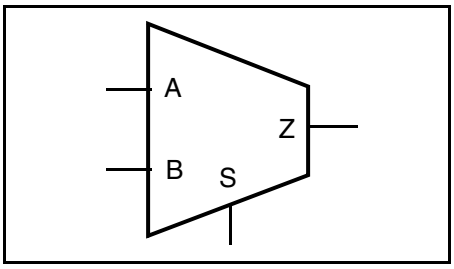
2-Bit Multiplexer

Truth Table

S	Z
0	A
1	B

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
S	1	Input
Z	1	Output



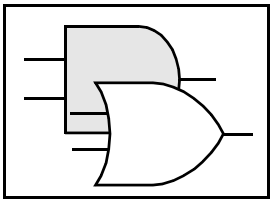
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_mux2_inst is
  port(in1, in2, sel: in std_logic;
       out1    : out std_logic);
end GT_mux2_inst;

architecture sim of GT_mux2_inst is
begin
  U1 : GTECH_MUX2
    port map(A => in1, B => in2,
             S => sel, Z => out1);
end sim;
```

Verilog

```
module GT_mux2_inst (in1, in2, sel,
                     out1);
  input  in1, in2, sel;
  output out1;

  GTECH_MUX2
    U1 (.A(in1), .B(in2),
        .S(sel), .Z(out1) );

endmodule
```



GTECH_MUXI2

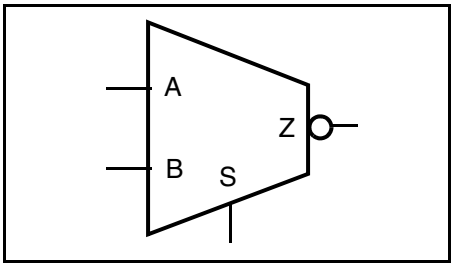
2-Bit Multiplexer

Truth Table

A	B	S	Z
0	X	0	1
1	X	0	0
X	0	1	1
X	1	1	0

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
S	1	Input
Z	1	Output



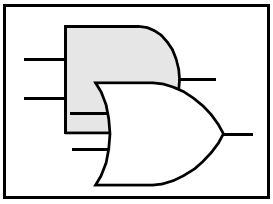
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_muxi2_inst is
  port(in1, in2, sel: in std_logic;
       out1    : out std_logic);
end GT_muxi2_inst;

architecture sim of GT_muxi2_inst is
begin
  U1 : GTECH_MUXI2
    port map(A => in1, B => in2,
             S => sel, Z => out1);
end sim;
```

Verilog

```
module GT_muxi2_inst (in1, in2, sel,
                      out1);
  input  in1, in2, sel;
  output out1;

  GTECH_MUXI2
    U1 (.A(in1), .B(in2),
        .S(sel), .Z(out1) );

endmodule
```



GTECH_MUX4

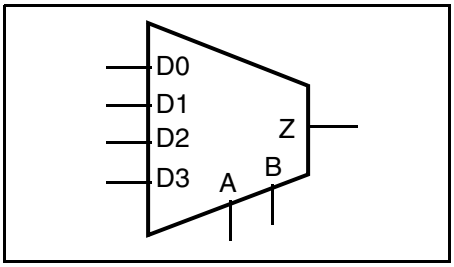
4-Bit Multiplexer

Truth Table

B	A	Z
0	0	D0
0	1	D1
1	0	D2
1	1	D3

Pin Description

Pin	Width	Direction
D0	1	Input
D1	1	Input
D2	1	Input
D3	1	Input
A	1	Input
B	1	Input
Z	1	Output



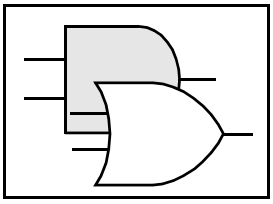
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_mux4_inst is
  port(in1, in2, in3, in4,
       sel1, sel2: in std_logic;
       out1      : out std_logic);
end GT_mux4_inst;

architecture sim of GT_mux4_inst is
begin
  U1 : GTECH_MUX4
    port map(D0 => in1, D1 => in2,
             D2 => in3, D3 => in4,
              A => sel1, B => sel2,
              Z => out1);
end sim;
```

Verilog

```
module GT_mux4_inst (in1, in2, in3,
             in4, sel1, sel2, out1);
  input in1, in2, in3, in4,
        sel1, sel2;
  output out1;

  GTECH_MUX4
    U1 (.D0(in1), .D1(in2),
        .D2(in3), .D3(in4),
        .A(sel1), .B(sel2),
        .Z(out1) );

endmodule
```



GTECH_MUX8

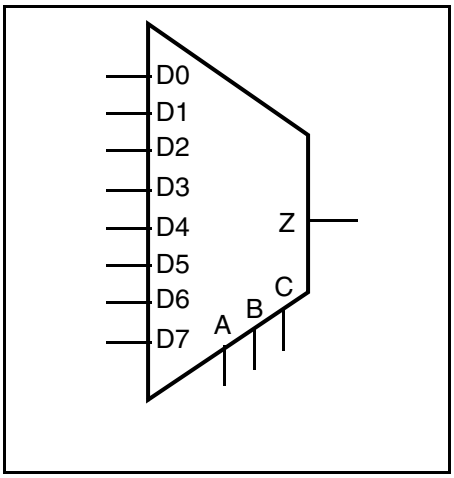
8-Bit Multiplexer

Truth Table

C	B	A	Z
0	0	0	D0
0	0	1	D1
0	1	0	D2
0	1	1	D3
1	0	0	D4
1	0	1	D5
1	1	0	D6
1	1	1	D7

Pin Description

Pin	Width	Direction
D0	1	Input
D1	1	Input
D2	1	Input
D3	1	Input
D4	1	Input
D5	1	Input
D6	1	Input
D7	1	Input
A	1	Input
B	1	Input
Z	1	Output



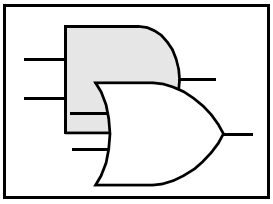
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_mux8_inst is
  port(in1, in2, in3, in4,
       in5, in6, in7, in8, sel1
       sel2, sel3: in std_logic;
       out1      : out std_logic);
end GT_mux8_inst;

architecture sim of GT_mux8_inst is
begin
  U1 : GTECH_MUX8
    port map(D0 => in1, D1 => in2,
             D2 => in3, D3 => in4,
             D4 => in5, D5 => in6,
             D6 => in7, D7 => in8,
              A => sel1, B => sel2,
```

Verilog

```
module GT_mux8_inst (in1, in2, in3,
            in4, in5, in6, in7, in8,
            sel1, sel2, sel3, out1);
  input in1, in2, in3, in4,
        in5, in6, in7, in8,
        sel1, sel2, sel3;
  output out1;

  GTECH_MUX4
    U1 (.D0(in1), .D1(in2),
        .D2(in3), .D3(in4),
        .D4(in5), .D5(in6),
        .D6(in7), .D7(in8),
        .A(sel1), .B(sel2),
        .C(sel3), .Z(out1) );

endmodule
```



GTECH_ADD_AB

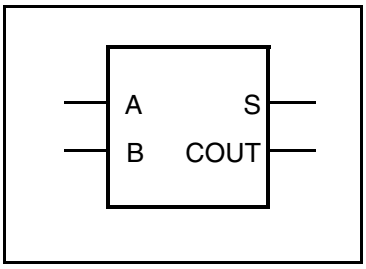
Half Adder

Truth Table

A	B	S	COUT
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
S	1	Output
COUT	1	Output



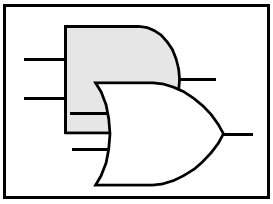
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_add_ab_inst is
  port(in1, in2 : in std_logic;
       sum, cout: out std_logic);
end GT_add_ab_inst;

architecture sim of GT_add_ab_inst
is
begin
  U1 : GTECH_ADD_AB
    port map(A => in1, B => in2,
             S => sum, COUT => cout);
end sim;
```

Verilog

```
module GT_add_ab_inst (in1, in2,
                       sum, cout);
  input  in1, in2;
  output sum, cout;

  GTECH_ADD_AB
    U1 (.A(in1), .B(in2),
        .S(sum), .COUT(cout) );

endmodule
```



GTECH_ADD_ABC

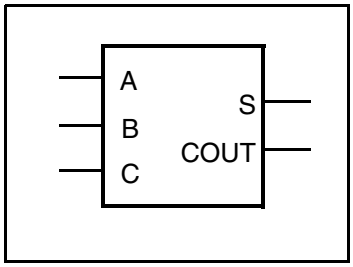
Full Adder

Truth Table

A	B	C	S	COUT
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Pin Description

Pin	Width	Direction
A	1	Input
B	1	Input
C	1	Input
S	1	Output
COUT	1	Output



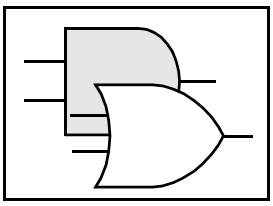
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_add_abc_inst is
  port(in1, in2, cin : in std_logic;
       sum, cout     : out std_logic);
end GT_add_abc_inst;

architecture sim of GT_add_abc_inst is
begin
  U1 : GTECH_ADD_ABC
    port map(A => in1, B => in2,
             C => cin,
             S => sum, COUT => cout);
end sim;
```

Verilog

```
module GT_add_abc_inst (in1, in2,
                        cin, sum,
cout);
  input  in1, in2, cin;
  output sum, cout;

  GTECH_ADD_ABC
    U1 (.A(in1), .B(in2), .C(cin),
        .S(sum), .COUT(cout) );

endmodule
```



GTECH_TBUF

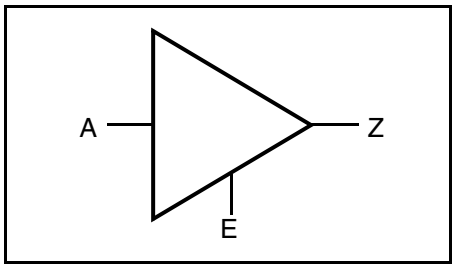
Non-Inverting 3-State Buffer

Truth Table

E	Z
1	A
0	high Z

Pin Description

Pin	Width	Direction
A	1	Input
E	1	Input
Z	1	Output



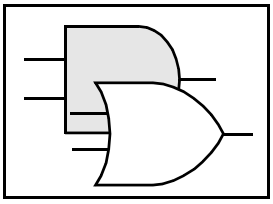
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_tbuf_inst is
  port(in1, enable: in std_logic;
       out1       : out std_logic);
end GT_tbuf_inst;

architecture sim of GT_tbuf_inst is
begin
  U1 : GTECH_TBUF
    port map(A => in1, E => enable,
             Z => out1);
end sim;
```

Verilog

```
module GT_tbuf_inst (in1, enable,
                     out1);
  input  in1, enable;
  output out1;

  GTECH_TBUF
    U1 (.A(in1), .E(enable),
        .Z(out1) );

endmodule
```



GTECH_INBUF

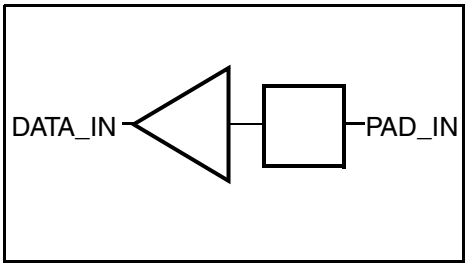
Input Buffer

Truth Table

PAD_IN	DATA_IN
0	0
1	1
Z	X

Pin Description

Pin	Width	Direction
PAD_IN	1	Input
DATA_IN	1	Output



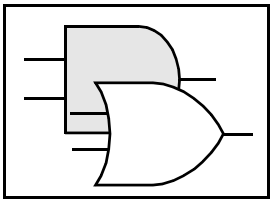
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_inbuf_inst is
  port(in1  : in std_logic;
       out1 : out std_logic);
end GT_inbuf_inst;

architecture sim of GT_inbuf_inst is
begin
  U1 : GTECH_INBUF
    port map(PAD_IN => in1,
             DATA_IN => out1);
end sim;
```

Verilog

```
module GT_inbuf_inst (in1, out1);
  input  in1;
  output out1;

  GTECH_INBUF
    U1 (.PAD_IN(in1),
        .DATA_IN(out1) );

endmodule
```

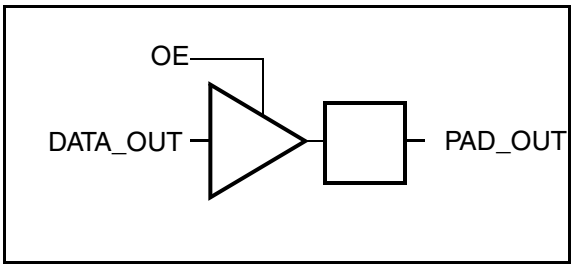


GTECH_OUTBUF

Output Buffer

Truth Table

DATA_OUT	OE	PAD_OUT
X	0	Z
1	1	1
0	1	0
Z	1	X



Pin Description

Pin	Width	Direction
DATA_OUT	1	Input
OE	1	Input
PAD_OUT	1	Output

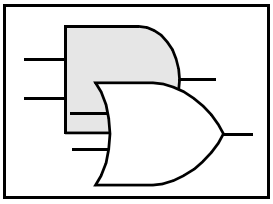
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_outbuf_inst is
  port(in1, enable: in std_logic;
       out1       : out std_logic);
end GT_outbuf_inst;

architecture sim of GT_outbuf_inst
is
begin
  U1 : GTECH_OUTBUF
    port map(DATA_OUT => in1,
     OE => enable, DATA_IN => out1);
end sim;
```

Verilog

```
module GT_outbuf_inst (in1, enable,
                       out1);
  input  in1, enable;
  output out1;

  GTECH_OUTBUF
    U1 (.DATA_OUT(in1), .OE(enable),
        .PAD_OUT(out1) );

endmodule
```

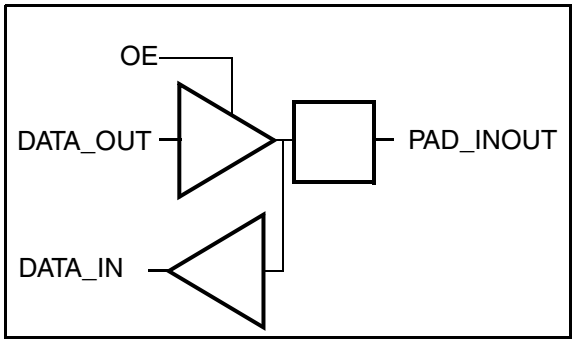


GTECH_INOUTBUF

Input-Output Buffer

Truth Table

DATA_OUT	OE	PAD_INOUT	PAD_INOUT
X	0	Z	X
X	0	1	1
X	0	0	0
1	1	1	1
0	1	0	0
Z	1	X	X



Pin Description

Pin	Width	Direction
DATA_OUT	1	Input
OE	1	Input
DATA_IN	1	Output
PAD_INOUT	1	Input/Output

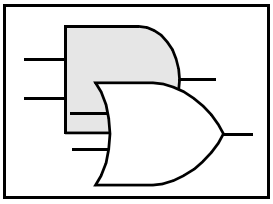
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_inoutbuf_inst is
  port(in1, enable: in std_logic;
       inout1     : inout std_logic;
       out1       : out std_logic);
end GT_inoutbuf_inst;

architecture sim of GT_inoutbuf_inst
is
begin
  U1 : GTECH_INOUTBUF
    port map(DATA_OUT => in1,
     OE => enable, PAD_INOUT =>
```

Verilog

```
module GT_inoutbuf_inst (in1,
            enable, inout1, out1);
  input  in1, enable;
  output out1;
  inout inout1;

  GTECH_INOUTBUF
    U1 (.DATA_OUT(in1),
        .OE(enable),
        .PAD_INOUT(inout1),
        .DATA_IN(out1 );

endmodule
```



GTECH_FD1

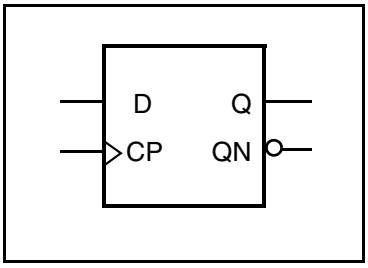
D Flip-Flop

Truth Table

D	CP	Q	QN
0	↑	0	1
1	↑	1	0
X	0	Q	QN
X	1	Q	QN

Pin Description

Pin	Width	Direction
D	1	Input
CP	1	Input
Q	1	Output
QN	1	Output



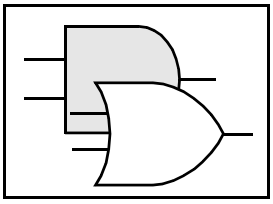
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fd1_inst is
  port(in1, cp: in std_logic;
       q, qb  : out std_logic);
end GT_fd1_inst;

architecture sim of GT_fd1_inst is
begin
  U1 : GTECH_FD1
    port map(D => in1, CP => cp,
             Q => q, QN => qb);
end sim;
```

Verilog

```
module GT_fd1_inst (in1, cp, q, qb);
  input  in1, cp;
  output q, qb;

  GTECH_FD1
    U1 (.D(in1), .CP(cp),
        .Q(q), .QN(qb) );

endmodule
```



GTECH_FD14

D Flip-Flop - 4 Bit

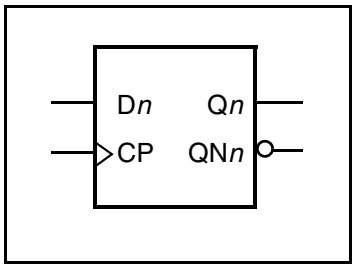
Truth Table

D*n*[a]	CP	Q*n*[a]	QN*n*[a]
0	↑	0	1
1	↑	1	0
X	0	Q	QN
X	1	Q	QN

a. The *n* denotes an individual signal of the D, Q, or QN bus.

Pin Description

Pin	Width	Direction
D0	1	Input
D1	1	Input
D2	1	Input
D3	1	Input
CP	1	Input
Q0	1	Output
Q1	1	Output
Q2	1	Output
Q3	1	Output
QN0	1	Output
QN1	1	Output
QN2	1	Output
QN3	1	Output



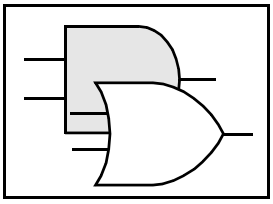
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fd14_inst is
  port(in0, in1, in2, in3,
       cp       : in std_logic;
       q0, q1, q2, q3, qb0, qb1,
       qb2, qb3 : out std_logic);
end GT_fd14_inst;

architecture sim of GT_fd14_inst
is
begin
  U1 : GTECH_FD14
    port map(D0 => in0, D1 => in1,
             D2 => in2, D3 => in3,
             CP => cp,
             Q0 => q0, Q1 => q1,
             Q2 => q2, Q3 => q3,
             QN0 => qb0, QN1 =>
qb1
             QN2 => qb2, QN3 =>
qb3
             );
end sim;
```

Verilog

```
module GT_fd14_inst (in0, in1, in2,
          in3, cp,
          q0, q1, q2, q3,
          qb0, qb1, qb2, qb3 );
  input  in0, in1, in2, in3, cp ;
  output q0, q2, q3, q4,
         qb0, qb1, qb2, qb3 ;

  GTECH_FD14
    U1 (.D0(in0), .D1(in1), .D2(in2),
        .D3(in3), .CP(cp),  .Q0(q0),
        .Q1(q1),  .Q2(q2),  .Q3(q3),
        .QN0(qb0), .QN1(qb1),
        .QN2(qb2), .QN3(qb3) );

endmodule
```



GTECH_FD18

D Flip-Flop - 8 Bit

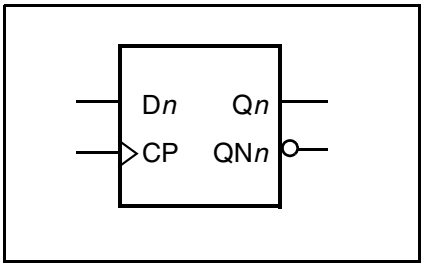
Truth Table

D*n*[a]	CP	Q*n*[a]	QN*n*[a]
0	↑	0	1
1	↑	1	0
X	0	Q	QN
X	1	Q	QN

a. The *n* denotes an individual signal of the D, Q, or QN bus.

Pin Description

Pin	Width	Direction
D0	1	Input
D1	1	Input
D2 - D6	1	Input
D7	1	Input
CP	1	Input
Q0	1	Output
Q1	1	Output
Q2 - Q6	1	Output
Q7	1	Output
QN0	1	Output
QN1	1	Output
QN2 - QN6	1	Output
QN7	1	Output



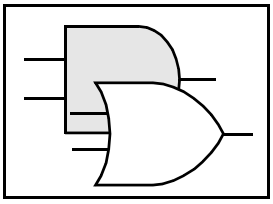
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fd18_inst is
  port(in0, in1, in2, in3,
       in4, in5, in6, in7,
       cp       : in std_logic;
       q0, q1, q2, q3, q4,
       q5, q6, q7, qb0, qb1,
       qb2, qb3, qb4, qb5, qb6,
       qb7 : out std_logic);
end GT_fd18_inst;

architecture sim of GT_fd18_inst is
begin
  U1 : GTECH_FD18
    port map(D0 => in0, D1 => in1,
             D2 => in2, D3 => in3,
             D4 => in4, D5 => in5,
             D6 => in6, D7 => in7,
             CP => cp,
             Q0 => q0, Q1 => q1,
             Q2 => q2, Q3 => q3,
             Q4 => q4, Q5 => q5,
             Q6 => q6, Q7 => q7,
             QN0 => qb0, QN1 => qb1
             QN2 => qb2, QN3 => qb3
             QN4 => qb4, QN5 => qb5
             QN6 => qb6, QN7 => qb7
             );
end sim;
```

Verilog

```
module GT_fd18_inst (in0, in1, in2,
          in3, in4, in5, in6, in7, cp,
          q0, q1, q2, q3,
          q4, q5, q6, q7,
          qb0, qb1, qb2, qb3,
          qb4, qb5, qb6, qb7 );
  input in0, in1, in2, in3, in4,
        in5, in6, in7, cp ;
  output q0, q1, q2, q3, q4, q5,
         q6, q7,
         qb0, qb1, qb2, qb3 qb4, qb5,
         qb6, qb7 ;

  GTECH_FD18
    U1 (.D0(in0), .D1(in1), .D2(in2),
        .D3(in3), .D4(in4), .D5(in5),
        .D6(in6), .D7(in7), .CP(cp),
        .Q0(q0), .Q1(q1), .Q2(q2),
        .Q3(q3), .Q4(q4), .Q5(q5),
        .Q6(q6), .Q7(q7),
        .QN0(qb0), .QN1(qb1),
        .QN2(qb2), .QN3(qb3) );

endmodule
```



GTECH_FD1S

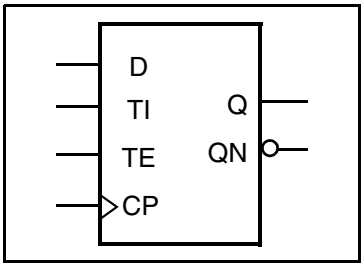
D Flip-Flop with Scan Test Pins

Truth Table

D	TI	TE	CP	Q	QN
0	X	0	↑	0	1
1	X	0	↑	1	0
X	0	1	↑	0	1
X	1	1	↑	1	0
X	X	X	0	Q	QN
X	X	X	1	Q	QN

Pin Description

Pin	Width	Direction
D	1	Input
TI	1	Input
TE	1	Input
CP	1	Input
Q	1	Output
QN	1	Output



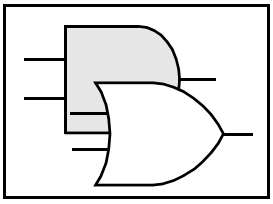
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fd1s_inst is
  port(in1, ti, te,
       cp: in std_logic;
       q, qb  : out std_logic);
end GT_fd1s_inst;

architecture sim of GT_fd1s_inst
is
begin
  U1 : GTECH_FD1S
    port map(D => in1, CP => ti,
             TI => te, TE => cp,
             Q => q, QN => qb);
end sim;
```

Verilog

```
module GT_fd1s_inst (in1, ti, te, cp,
                     q, qb);
  input  in1, ti, te, cp;
  output q, qb;

  GTECH_FD1S
    U1 (.D(in1), .CP(ti), .TI(te),
        .TE(cp), .Q(q), .QN(qb) );

endmodule
```



GTECH_FD2

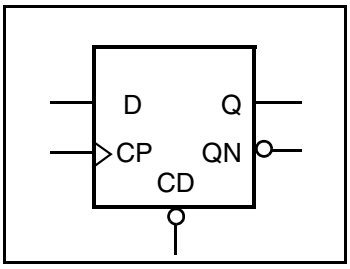
D Flip-Flop with Clear

Truth Table

D	CP	CD	Q	QN
0	↑	1	0	1
1	↑	1	1	0
X	0	1	Q	QN
X	1	1	Q	QN
X	X	0	0	1

Pin Description

Pin	Width	Direction
D	1	Input
CP	1	Input
CD	1	Input
Q	1	Output
QN	1	Output



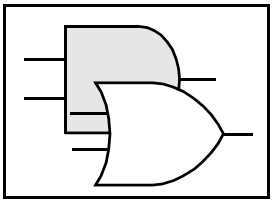
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fd2_inst is
  port(in1, cp, clr : in std_logic;
       q, qb : out std_logic);
end GT_fd2_inst;

architecture sim of GT_fd2_inst is
begin
  U1 : GTECH_FD2
    port map(D => in1, CP => cp,
             CD => clr,
             Q => q, QN => qb);
end sim;
```

Verilog

```
module GT_fd2_inst (in1, cp, clr,
                    q, qb);
  input  in1, cp, clr;
  output q, qb;

  GTECH_FD2
    U1 (.D(in1), .CP(cp), .CD(clr),
        .Q(q), .QN(qb) );

endmodule
```



GTECH_FD24

D Flip-Flop with Clear - 4 Bit

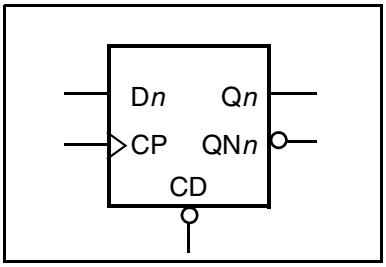
Truth Table

D*n*[a]	CP	CD	Q*n*[a]	QN*n*[a]
0	↑	1	0	1
1	↑	1	1	0
X	0	1	Q	QN
X	1	1	Q	QN
X	X	0	0	1

a. The *n* denotes an individual signal of the D, Q, or QN bus.

Pin Description

Pin	Width	Direction
D0	1	Input
D1 - D3	1	Input
CP	1	Input
CD	1	Input
Q0	1	Output
Q1 - Q3	1	Output
QN0	1	Output
QN1 - QN 3	1	Output



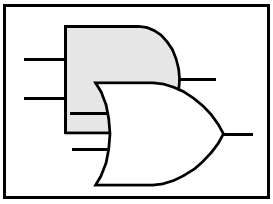
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fd24_inst is
  port(in0, in1, in2, in3,
       cp, clr  : in std_logic;
       q0, q1, q2, q3, qb0, qb1,
       qb2, qb3 : out std_logic);
end GT_fd24_inst;

architecture sim of GT_fd24_inst
is
begin
  U1 : GTECH_FD24
    port map(D0 => in0, D1 => in1,
             D2 => in2, D3 => in3,
             CP => cp,  CD => clr,
             Q0 => q0,  Q1 => q1,
             Q2 => q2,  Q3 => q3,
            QN0 => qb0, QN1 =>
```

Verilog

```
module GT_fd24_inst (in0, in1, in2,
          in3, cp, clr,
          q0, q1, q2, q3,
          qb0, qb1, qb2, qb3 );
  input  in0, in1, in2, in3, cp ;
  output q0, q2, q3, q4,
         qb0, qb1, qb2, qb3 ;

  GTECH_FD24
    U1 (.D0(in0), .D1(in1), .D2(in2),
        .D3(in3), .CP(cp),  .CD(clr),
        .Q0(q0),  .Q1(q1),  .Q2(q2),
        .Q3(q3),
        .QN0(qb0), .QN1(qb1),
        .QN2(qb2), .QN3(qb3) );

endmodule
```



GTECH_FD28

D Flip-Flop with Clear - 8 Bit

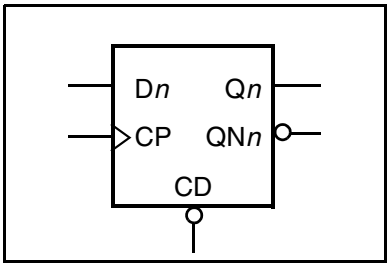
Truth Table

D*n*[a]	CP	CD	Q*n*[a]	QN*n*[a]
0	↑	1	0	1
1	↑	1	1	0
X	0	1	Q	QN
X	1	1	Q	QN
X	X	0	0	1

a. The *n* denotes an individual signal of the D, Q, or QN bus.

Pin Description

Pin	Width	Direction
D0	1	Input
D1	1	Input
D2 - D7	1	Input
CP	1	Input
CD	1	Input
Q0	1	Output
Q1	1	Output
Q2 - Q7	1	Output
QN0	1	Output
QN1	1	Output
QN2 - QN7	1	Output



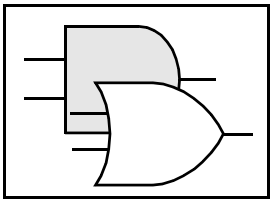
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fd28_inst is
  port(in0, in1, in2, in3,
       in4, in5, in6, in7,
       cp, clr : in std_logic;
       q0, q1, q2, q3, q4,
       q5, q6, q7, qb0, qb1,
       qb2, qb3, qb4, qb5, qb6,
       qb7 : out std_logic);
end GT_fd28_inst;

architecture sim of GT_fd28_inst
is
begin
  U1 : GTECH_FD28
    port map(D0 => in0, D1 => in1,
             D2 => in2, D3 => in3,
             D4 => in4, D5 => in5,
             D6 => in6, D7 => in7,
             CP => cp, CD => clr,
             Q0 => q0, Q1 => q1,
             Q2 => q2, Q3 => q3,
             Q4 => q4, Q5 => q5,
             Q6 => q6, Q7 => q7,
            QN0 => qb0, QN1 =>
qb1,
            QN2 => qb2, QN3 =>
qb3,
            QN4 => qb4, QN5 =>
qb5,
            QN6 => qb6, QN7 => qb7
             );
end sim;
```

Verilog

```
module GT_fd28_inst (in0, in1, in2,
          in3, in4, in5, in6, in7, cp,
          clr, q0, q1, q2, q3,
          q4, q5, q6, q7,
          qb0, qb1, qb2, qb3,
          qb4, qb5, qb6, qb7 );
  input in0, in1, in2, in3, in4,
        in5, in6, in7, cp, clr;
  output q0, q1, q2, q3, q4, q5,
         q6, q7,
         qb0, qb1, qb2, qb3 qb4, qb5,
         qb6, qb7 ;

  GTECH_FD28
    U1 (.D0(in0), .D1(in1), .D2(in2),
        .D3(in3), .D4(in4), .D5(in5),
        .D6(in6), .D7(in7),
        .CP(cp), .CD(clr),
        .Q0(q0), .Q1(q1), .Q2(q2),
        .Q3(q3), .Q4(q4), .Q5(q5),
        .Q6(q6), .Q7(q7),
        .QN0(qb0), .QN1(qb1),
        .QN2(qb2), .QN3(qb3) );

endmodule
```



GTECH_FD2S

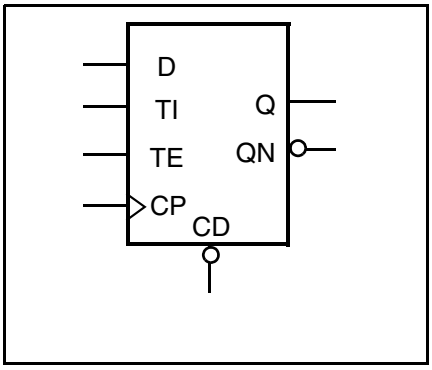
D Flip-Flop with Clear, Scan

Truth Table

D	TI	TE	CP	CD	Q	QN
0	X	0	↑	1	0	1
1	X	0	↑	1	1	0
X	0	1	↑	1	0	1
X	1	1	↑	1	1	0
X	X	X	0	1	Q	QN
X	X	X	1	1	Q	QN
X	X	X	X	0	0	1

Pin Description

Pin	Width	Direction
D	1	Input
TI	1	Input
TE	1	Input
CP	1	Input
CD	1	Input
Q	1	Output
QN	1	Output



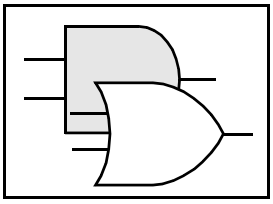
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fd2s_inst is
  port(in1, ti, te,
       cp, clr : in std_logic;
       q, qb   : out std_logic);
end GT_fd2s_inst;

architecture sim of GT_fd2s_inst
is
begin
  U1 : GTECH_FD2S
    port map(D => in1, CP => ti,
             CD => clr,
             TI => te, TE => cp,
             Q => q, QN => qb);
end sim;
```

Verilog

```
module GT_fd2s_inst (in1, ti, te, cp,
                     clr, q, qb);
  input  in1, ti, te, cp, clr;
  output q, qb;

  GTECH_FD2S
    U1 (.D(in1), .CP(ti), .CD(clr),
        .TI(te), .TE(cp),
        .Q(q), .QN(qb) );

endmodule
```



GTECH_FD3

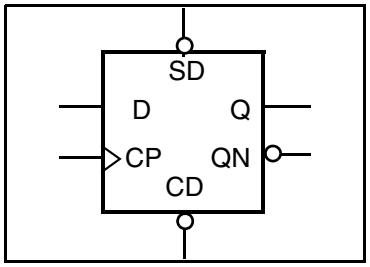
D Flip-Flop with Clear, Set

Truth Table

D	CP	CD	SD	Q	QN
0	↑	1	1	0	1
1	↑	1	1	1	0
X	0	1	1	Q	QN
X	1	1	1	Q	QN
X	X	0	1	0	1
X	X	1	0	1	0
X	X	0	0	0	0

Pin Description

Pin	Width	Direction
D	1	Input
CP	1	Input
CD	1	Input
SD	1	Input
Q	1	Output
QN	1	Output



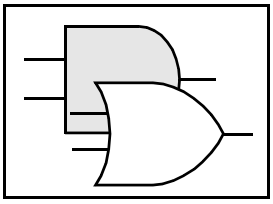
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fd3_inst is
  port(in1, cp, clr,
       set: in std_logic;
       q, qb : out std_logic);
end GT_fd3_inst;

architecture sim of GT_fd3_inst is
begin
  U1 : GTECH_FD3
    port map(D => in1, CP => cp,
             CD => clr, SD => set,
             Q => q, QN => qb);
end sim;
```

Verilog

```
module GT_fd3_inst (in1, cp, clr,
                    set, q, qb);
  input  in1, cp, clr, set;
  output q, qb;

  GTECH_FD3
    U1 (.D(in1), .CP(cp), .CD(clr),
        .SD(set), .Q(q), .QN(qb) );

endmodule
```



GTECH_FD34

D Flip-Flop with Clear, Set - 4 Bit

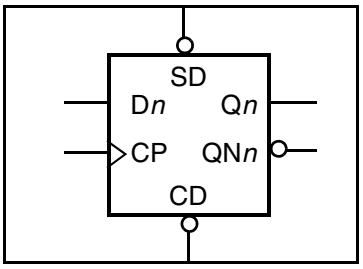
Truth Table

D*n*[a]	CP	CD	SD	Q*n*[a]	QN*n*[a]
0	↑	1	1	0	1
1	↑	1	1	1	0
X	0	1	1	Q	QN
X	1	1	1	Q	QN
X	X	0	1	0	1
X	X	1	0	1	0
X	X	0	0	0	0

a. The *n* denotes an individual signal of the D, Q, or QN bus.

Pin Description

Pin	Width	Direction
D0	1	Input
D1 - D3	1	Input
CP	1	Input
CD	1	Input
SD	1	Input
Q0	1	Output
Q1 - Q3	1	Output
QN0	1	Output
QN1 - QN 3	1	Output



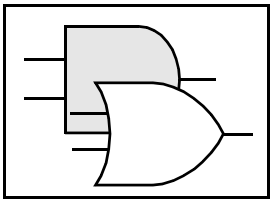
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fd34_inst is
  port(in0, in1, in2, in3,
       cp, clr, set: in std_logic;
       q0, q1, q2, q3, qb0, qb1,
       qb2, qb3 : out std_logic);
end GT_fd34_inst;

architecture sim of GT_fd34_inst is
begin
  U1 : GTECH_FD34
    port map(D0 => in0, D1 => in1,
             D2 => in2, D3 => in3,
             CP => cp,  CD => clr,
             SD => set,              Q0
=> q0,  Q1 => q1,
             Q2 => q2,  Q3 => q3,
            QN0 => qb0, QN1 => qb1,
             QN2 => qb2, QN3 => qb3
             );
end sim;
```

Verilog

```
module GT_fd34_inst (in0, in1, in2,
          in3, cp, clr, set,
          q0, q1, q2, q3,
          qb0, qb1, qb2, qb3 );
  input  in0, in1, in2, in3, cp,
         clr, set;
  output q0, q2, q3, q4,
         qb0, qb1, qb2, qb3 ;

  GTECH_FD34
    U1 (.D0(in0), .D1(in1), .D2(in2),
        .D3(in3), .CP(cp),  .CD(clr),
        .SD(set), .Q0(q0),
        .Q1(q1),  .Q2(q2), .Q3(q3),
        .QN0(qb0), .QN1(qb1),
        .QN2(qb2), .QN3(qb3) );

endmodule
```



GTECH_FD38

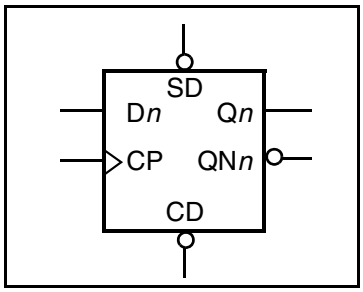
D Flip-Flop with Clear, Set - 8 Bit

Truth Table

D*n*[a]	CP	CD	SD	Q*n*[a]	QN*n*[a]
0	↑	1	1	0	1
1	↑	1	1	1	0
X	0	1	1	Q	QN
X	1	1	1	Q	QN
X	X	0	1	0	1
X	X	1	0	1	0
X	X	0	0	0	0

Pin Description

Pin	Width	Direction
D0	1	Input
D1	1	Input
D2 - D7	1	Input
CP	1	Input
CD	1	Input
SD	1	Input
Q0	1	Output
Q1	1	Output
Q2 - Q7	1	Output
QN0	1	Output
QN1	1	Output
QN2 - QN7	1	Output



a. The *n* denotes an individual signal of the D, Q, or QN bus.

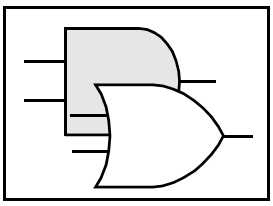
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fd38_inst is
  port(in0, in1, in2, in3,
       in4, in5, in6, in7,
       cp, clr, set: in std_logic;
       q0, q1, q2, q3, q4,
       q5, q6, q7, qb0, qb1,
       qb2, qb3, qb4, qb5, qb6,
       qb7 : out std_logic);
end GT_fd38_inst;

architecture sim of GT_fd38_inst is
begin
  U1 : GTECH_FD38
    port map(D0 => in0, D1 => in1,
             D2 => in2, D3 => in3,
             D4 => in4, D5 => in5,
             D6 => in6, D7 => in7,
             CP => cp, CD => clr,
             SD => set,
             Q0 => q0, Q1 => q1,
             Q2 => q2, Q3 => q3,
             Q4 => q4, Q5 => q5,
             Q6 => q6, Q7 => q7,
            QN0 => qb0, QN1 => qb1,
            QN2 => qb2, QN3 => qb3,
            QN4 => qb4, QN5 => qb5,
            QN6 => qb6, QN7 => qb7
             );
end sim;
```

Verilog

```
module GT_fd38_inst (in0, in1, in2,
          in3, in4, in5, in6, in7, cp,
          clr, set, q0, q1, q2, q3,
          q4, q5, q6, q7,
          qb0, qb1, qb2, qb3,
          qb4, qb5, qb6, qb7 );
  input in0, in1, in2, in3, in4,
        in5, in6, in7, cp, clr, set;
  output q0, q1, q2, q3, q4, q5,
         q6, q7,
         qb0, qb1, qb2, qb3 qb4, qb5,
         qb6, qb7 ;

  GTECH_FD38
    U1 (.D0(in0), .D1(in1), .D2(in2),
        .D3(in3), .D4(in4), .D5(in5),
        .D6(in6), .D7(in7),
        .CP(cp), .CD(clr), .SD(set),
        .Q0(q0), .Q1(q1), .Q2(q2),
        .Q3(q3), .Q4(q4), .Q5(q5),
        .Q6(q6), .Q7(q7),
        .QN0(qb0), .QN1(qb1),
        .QN2(qb2), .QN3(qb3) );

endmodule
```



GTECH_FD3S

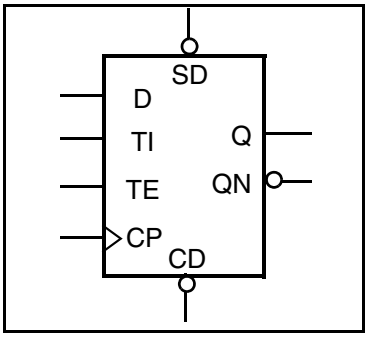
D Flip-Flop with Clear, Set, and Scan

Truth Table

D	TI	TE	CP	CD	SD	Q	QN
0	X	0	↑	1	1	0	1
1	X	0	↑	1	1	1	0
X	0	1	↑	1	1	0	1
X	1	1	↑	1	1	1	0
X	X	X	0	1	1	Q	QN
X	X	X	1	1	1	Q	QN
X	X	X	X	0	1	0	1
X	X	X	X	1	0	1	0
X	X	X	X	0	0	0	0

Pin Description

Pin	Width	Direction
D	1	Input
TI	1	Input
TE	1	Input
CP	1	Input
CD	1	Input
SD	1	Input
Q	1	Output
QN	1	Output



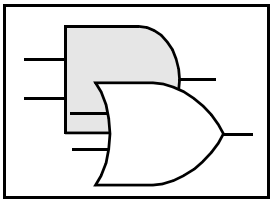
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fd3s_inst is
  port(in1, ti, te,
       cp, clr, set : in
std_logic;
       q, qb   : out std_logic);
end GT_fd3s_inst;

architecture sim of GT_fd3s_inst
is
begin
  U1 : GTECH_FD3S
    port map(D => in1, CP => ti,
             CD => clr, SD => set,
             TI => te, TE => cp,
             Q => q, QN => qb);
end sim;
```

Verilog

```
module GT_fd3s_inst (in1, ti, te, cp,
                     clr, set, q, qb);
  input  in1, ti, te, cp, clr, set ;
  output q, qb;

  GTECH_FD3S
    U1 (.D(in1), .CP(ti), .CD(clr),
        .SD(set), .TI(te), .TE(cp),
        .Q(q), .QN(qb) );

endmodule
```



GTECH_FD4

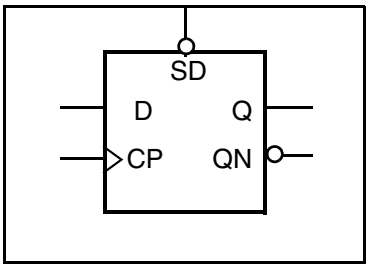
D Flip-Flop with Set

Truth Table

D	CP	SD	Q	QN
0	↑	1	0	1
1	↑	1	1	0
X	0	1	Q	QN
X	1	1	Q	QN
X	X	0	1	0

Pin Description

Pin	Width	Direction
D	1	Input
CP	1	Input
SD	1	Input
Q	1	Output
QN	1	Output



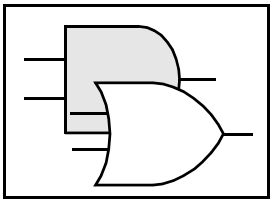
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fd4_inst is
  port(in1, cp,
       set: in std_logic;
       q, qb : out std_logic);
end GT_fd4_inst;

architecture sim of GT_fd4_inst is
begin
  U1 : GTECH_FD4
    port map(D => in1, CP => cp,
             SD => set,
             Q => q, QN => qb);
end sim;
```

Verilog

```
module GT_fd4_inst (in1, cp,
                    set, q, qb);
  input  in1, cp, set;
  output q, qb;

  GTECH_FD4
    U1 (.D(in1), .CP(cp),
        .SD(set), .Q(q), .QN(qb) );

endmodule
```



GTECH_FD44

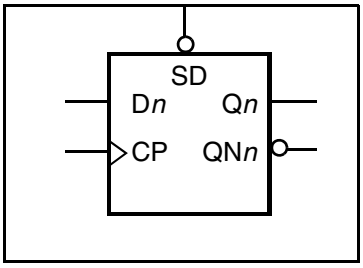
D Flip-Flop with Set - 4 Bit

Truth Table

D*n*[a]	CP	SD	Q*n*[a]	QN*n*[a]
0	↑	1	0	1
1	↑	1	1	0
X	0	1	Q	QN
X	1	1	Q	QN
X	X	0	1	0

Pin Description

Pin	Width	Direction
D0	1	Input
D1 - D3	1	Input
CP	1	Input
SD	1	Input
Q0	1	Output
Q1 - Q3	1	Output
QN0	1	Output
QN1 - QN 3	1	Output



a. The *n* denotes an individual signal of the D, Q, or QN bus.

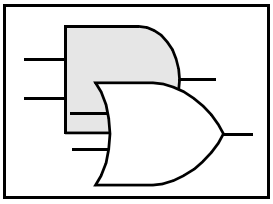
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fd44_inst is
  port(in0, in1, in2, in3,
       cp, set: in std_logic;
       q0, q1, q2, q3, qb0, qb1,
       qb2, qb3 : out std_logic);
end GT_fd44_inst;

architecture sim of GT_fd44_inst is
begin
  U1 : GTECH_FD44
    port map(D0 => in0, D1 => in1,
             D2 => in2, D3 => in3,
             CP => cp,
             SD => set,
             Q0 => q0,  Q1 => q1,
             Q2 => q2,  Q3 => q3,
            QN0 => qb0, QN1 => qb1,
             QN2 => qb2, QN3 => qb3
```

Verilog

```
module GT_fd44_inst (in0, in1, in2,
          in3, cp, set,
          q0, q1, q2, q3,
          qb0, qb1, qb2, qb3 );
  input  in0, in1, in2, in3, cp, set;
  output q0, q2, q3, q4,
         qb0, qb1, qb2, qb3 ;

  GTECH_FD44
    U1 (.D0(in0), .D1(in1), .D2(in2),
        .D3(in3), .CP(cp),
        .SD(set), .Q0(q0),
        .Q1(q1),  .Q2(q2), .Q3(q3),
        .QN0(qb0), .QN1(qb1),
        .QN2(qb2), .QN3(qb3) );

endmodule
```



GTECH_FD48

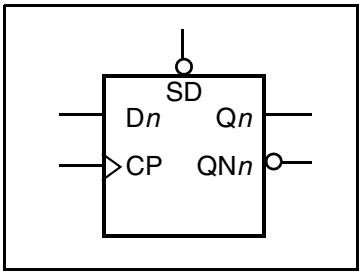
D Flip-Flop with Set - 8 Bit

Truth Table

D*n*[a]	CP	SD	Q*n*[a]	QN*n*[a]
0	↑	1	0	1
1	↑	1	1	0
X	0	1	Q	QN
X	1	1	Q	QN
X	X	0	1	0

Pin Description

Pin	Width	Direction
D0	1	Input
D1	1	Input
D2 - D7	1	Input
CP	1	Input
SD	1	Input
Q0	1	Output
Q1	1	Output
Q2 - Q7	1	Output
QN0	1	Output
QN1	1	Output
QN2 - QN7	1	Output



a. The *n* denotes an individual signal of the D, Q, or QN bus.

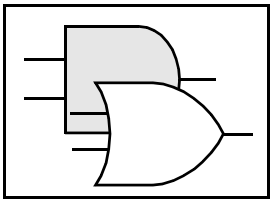
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fd48_inst is
  port(in0, in1, in2, in3,
       in4, in5, in6, in7,
       cp, set: in std_logic;
       q0, q1, q2, q3, q4,
       q5, q6, q7, qb0, qb1,
       qb2, qb3, qb4, qb5, qb6,
       qb7 : out std_logic);
end GT_fd48_inst;

architecture sim of GT_fd48_inst is
begin
  U1 : GTECH_FD48
    port map(D0 => in0, D1 => in1,
             D2 => in2, D3 => in3,
             D4 => in4, D5 => in5,
             D6 => in6, D7 => in7,
             CP => cp, SD => set,
             Q0 => q0, Q1 => q1,
             Q2 => q2, Q3 => q3,
             Q4 => q4, Q5 => q5,
             Q6 => q6, Q7 => q7,
            QN0 => qb0, QN1 => qb1,
            QN2 => qb2, QN3 => qb3,
            QN4 => qb4, QN5 => qb5,
            QN6 => qb6, QN7 => qb7
             );
end sim;
```

Verilog

```
module GT_fd48_inst (in0, in1, in2,
          in3, in4, in5, in6, in7, cp,
          set, q0, q1, q2, q3,
          q4, q5, q6, q7,
          qb0, qb1, qb2, qb3,
          qb4, qb5, qb6, qb7 );
  input in0, in1, in2, in3, in4,
        in5, in6, in7, cp, set;
  output q0, q1, q2, q3, q4, q5,
         q6, q7,
         qb0, qb1, qb2, qb3 qb4, qb5,
         qb6, qb7 ;

  GTECH_FD48
    U1 (.D0(in0), .D1(in1), .D2(in2),
        .D3(in3), .D4(in4), .D5(in5),
        .D6(in6), .D7(in7),
        .CP(cp),  .SD(set),
        .Q0(q0), .Q1(q1), .Q2(q2),
        .Q3(q3), .Q4(q4), .Q5(q5),
        .Q6(q6), .Q7(q7),
        .QN0(qb0), .QN1(qb1),
        .QN2(qb2), .QN3(qb3) );

endmodule
```



GTECH_FD4S

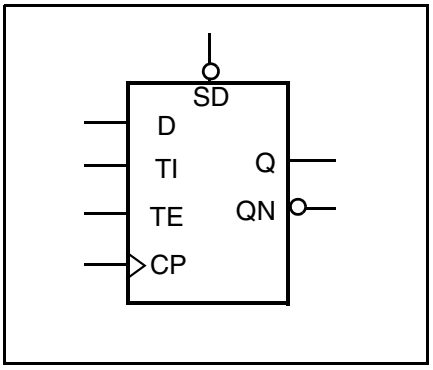
D Flip-Flop with Set, Scan

Truth Table

D	TI	TE	CP	SD	Q	QN
0	X	0	↑	1	0	1
1	X	0	↑	1	1	0
X	0	1	↑	1	0	1
X	1	1	↑	1	1	0
X	X	X	0	1	Q	QN
X	X	X	1	1	Q	QN
X	X	X	X	0	1	1

Pin Description

Pin	Width	Direction
D	1	Input
TI	1	Input
TE	1	Input
CP	1	Input
SD	1	Input
Q	1	Output
QN	1	Output



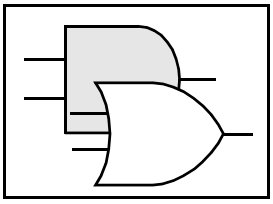
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fd4s_inst is
  port(in1, ti, te,
       cp, set : in std_logic;
       q, qb   : out std_logic);
end GT_fd4s_inst;

architecture sim of GT_fd4s_inst
is
begin
  U1 : GTECH_FD4S
    port map(D => in1, CP => ti,
             SD => te,
             TI => cp, TE => set,
             Q => q, QN => qb);
end sim;
```

Verilog

```
module GT_fd4s_inst (in1, ti, te, cp,
                     set, q, qb);
  input  in1, ti, te, cp, set;
  output q, qb;

  GTECH_FD4S
    U1 (.D(in1), .CP(ti), .SD(te),
        .TI(cp), .TE(set),
        .Q(q), .QN(qb) );

endmodule
```



GTECH_FJK1

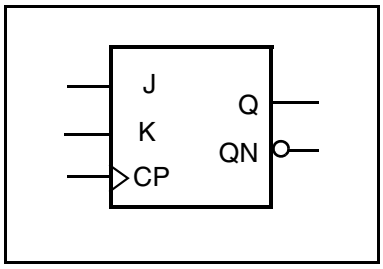
JK Flip-Flop

Truth Table

J	K	CP	Q	QN
0	0	↑	Q	QN
0	1	↑	0	1
1	0	↑	1	0
1	1	↑	QN	Q
X	X	0	Q	QN
X	X	1	Q	QN

Pin Description

Pin	Width	Direction
J	1	Input
K	1	Input
CP	1	Input
Q	1	Output
QN	1	Output



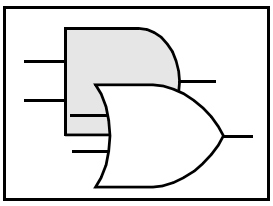
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fjk1_inst is
  port(jin, kin, cp: in std_logic;
       q, qb  : out std_logic);
end GT_fjk1_inst;

architecture sim of GT_fjk1_inst
is
begin
  U1 : GTECH_FJK1
    port map(J => jin, K => kin,
             CP => cp,
             Q => q, QN => qb);
end sim;
```

Verilog

```
module GT_fjk1_inst (jin, kin, cp,
                     q, qb);
  input  jin, kin, cp;
  output q, qb;

  GTECH_FJK1
    U1 (.J(jin), .K(kin), .CP(cp),
        .Q(q), .QN(qb) );

endmodule
```



GTECH_FJK1S

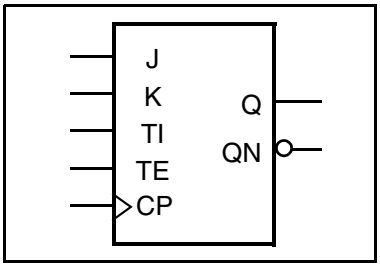
JK Flip-Flop with Scan Test Pins

Truth Table

J	K	TI	TE	CP	Q	QN
0	0	X	0	↑	Q	QN
0	1	X	0	↑	0	1
1	0	X	0	↑	1	0
1	1	X	0	↑	QN	Q
X	X	0	1	↑	0	1
X	X	1	1	↑	1	0
X	X	X	X	0	Q	QN
X	X	X	X	1	Q	QN

Pin Description

Pin	Width	Direction
J	1	Input
K	1	Input
TI	1	Input
TE	1	Input
CP	1	Input
Q	1	Output
QN	1	Output



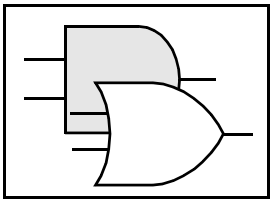
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fjk1s_inst is
  port(jin, kin, ti, te,
       cp: in std_logic;
       q, qb  : out std_logic);
end GT_fjk1s_inst;

architecture sim of GT_fjk1s_inst
is
begin
  U1 : GTECH_FJK1S
    port map(J => jin, K => kin,
             CP => ti,
             TI => te, TE => cp,
             Q => q, QN => qb);
end sim;
```

Verilog

```
module GT_fjk1s_inst (jin, kin, ti,
                      te, cp, q, qb);
  input  jin, kin, ti, te, cp;
  output q, qb;

  GTECH_FJK1S
    U1 (.J(jin), .K(kin), .CP(ti),
        .TI(te), .TE(cp),
        .Q(q), .QN(qb) );

endmodule
```



GTECH_FJK2

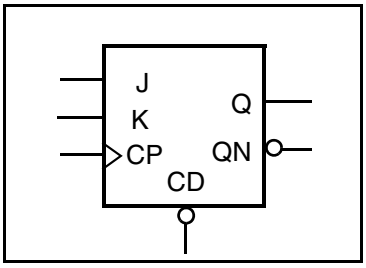
JK Flip-Flop with Clear

Truth Table

J	K	CP	CD	Q	QN
0	0	↑	1	Q	QN
0	1	↑	1	0	1
1	0	↑	1	1	0
1	1	↑	1	QN	Q
X	X	0	1	Q	QN
X	X	1	1	Q	QN
X	X	X	0	0	1

Pin Description

Pin	Width	Direction
J	1	Input
K	1	Input
CP	1	Input
CD	1	Input
Q	1	Output
QN	1	Output



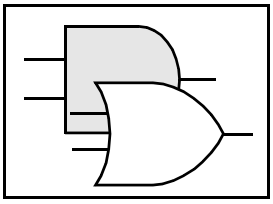
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fjk2_inst is
  port(jin, kin,
       cp, clr : in std_logic;
       q, qb   : out std_logic);
end GT_fjk2_inst;

architecture sim of GT_fjk2_inst is
begin
  U1 : GTECH_FJK2
    port map(J => jin, K => kin,
             CP => cp, CD => clr,
             Q => q, QN => qb);
end sim;
```

Verilog

```
module GT_fjk2_inst (jin, kin, cp,
                     clr, q, qb);
  input  jin, kin, cp, clr;
  output q, qb;

  GTECH_FJK2
    U1 (.J(in1), .K(kin), .CP(cp),
        .CD(clr), .Q(q), .QN(qb) );

endmodule
```



GTECH_FJK2S

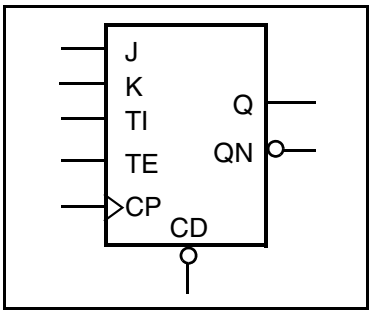
JK Flip-Flop with Clear, Scan

Truth Table

J	K	TI	TE	CP	CD	Q	QN
0	0	X	0	↑	1	Q	QN
0	1	X	0	↑	1	0	1
1	0	X	0	↑	1	1	0
1	1	X	0	↑	1	QN	Q
X	X	0	1	↑	1	0	1
X	X	1	1	↑	1	1	0
X	X	X	X	0	1	Q	QN
X	X	X	X	1	1	Q	QN
X	X	X	X	X	0	0	1

Pin Description

Pin	Width	Direction
J	1	Input
K	1	Input
TI	1	Input
TE	1	Input
CP	1	Input
CD	1	Input
Q	1	Output
QN	1	Output



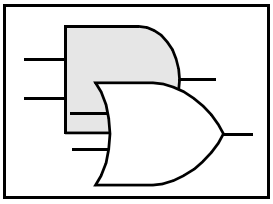
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fjk2s_inst is
  port(jin, kin, ti, te,
       cp, clr : in std_logic;
       q, qb   : out std_logic);
end GT_fjk2s_inst;

architecture sim of GT_fjk2s_inst
is
begin
  U1 : GTECH_FJK2S
    port map(J => jin, K => kin,
             CP => ti, CD => te,
             TI => cp, TE => clr,
             Q => q, QN => qb);
end sim;
```

Verilog

```
module GT_fjk2s_inst (jin, kin, ti,
                      te, cp,
                      clr, q, qb);
  input  jin, kin, ti, te, cp, clr;
  output q, qb;

  GTECH_FJK2S
    U1 (.J(jin), .K(kin), .CP(ti),
        .CD(te),
        .TI(cp), .TE(clr),
        .Q(q), .QN(qb) );

endmodule
```



GTECH_FJK3

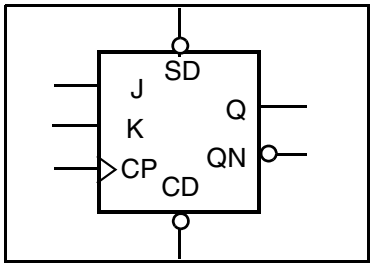
JK Flip-Flop with Clear, Set

Truth Table

J	K	CP	CD	SD	Q	QN
0	0	↑	1	1	Q	QN
0	1	↑	1	1	0	1
1	0	↑	1	1	1	0
1	1	↑	1	1	QN	Q
X	X	0	1	1	Q	QN
X	X	1	1	1	Q	QN
X	X	X	0	1	0	1
X	X	X	1	0	1	0
X	X	X	0	0	0	0

Pin Description

Pin	Width	Direction
J	1	Input
K	1	Input
CP	1	Input
CD	1	Input
SD	1	Input
Q	1	Output
QN	1	Output



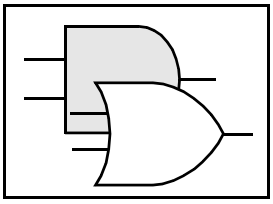
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fjk3_inst is
  port(jin, kin, cp, clr,
       set: in std_logic;
       q, qb : out std_logic);
end GT_fjk3_inst;

architecture sim of GT_fjk3_inst is
begin
  U1 : GTECH_FJK3
    port map(J => jin, K => kin,
             CP => cp,
             CD => clr, SD => set,
             Q => q, QN => qb);
end sim;
```

Verilog

```
module GT_fjk3_inst (jin, kin, cp,
                     clr, set, q,
qb);
  input  jin, kin, cp, clr, set;
  output q, qb;

  GTECH_JK3
    U1 (.J(in1), .K(kin), .CP(cp),
        .CD(clr), .SD(set),
        .Q(q), .QN(qb) );

endmodule
```



GTECH_FJK3S

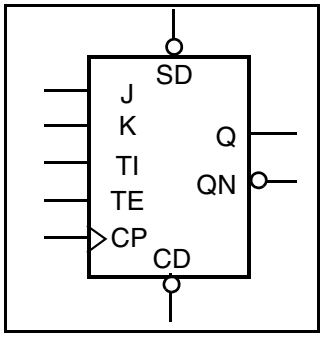
JK Flip-Flop with Clear, Set, and Scan

Truth Table

J	K	TI	TE	CP	CD	SD	Q	QN
0	0	X	0	↑	1	1	Q	QN
0	1	X	0	↑	1	1	0	1
1	0	X	0	↑	1	1	1	0
1	1	X	0	↑	1	1	QN	Q
X	X	0	1	↑	1	1	0	1
X	X	1	1	↑	1	1	1	0
X	X	X	X	0	1	1	Q	QN
X	X	X	X	1	1	1	Q	QN
X	X	X	X	X	0	1	0	1
X	X	X	X	X	1	0	1	0
X	X	X	X	X	0	0	0	0

Pin Description

Pin	Width	Direction
J	1	Input
K	1	Input
TI	1	Input
TE	1	Input
CP	1	Input
CD	1	Input
SD	1	Input
Q	1	Output
QN	1	Output



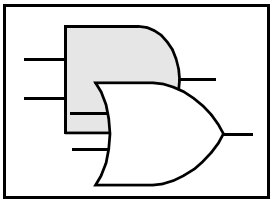
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_fjk3s_inst is
  port(jin, kin, ti, te,
       cp, clr, set : in std_logic;
       q, qb   : out std_logic);
end GT_fjk3s_inst;

architecture sim of GT_fjk3s_inst
is
begin
  U1 : GTECH_FJK3S
    port map(J => jin, K => kin,
             CP => ti,
             CD => te, SD => cp,
```

Verilog

```
module GT_fjk3s_inst (jin, kin,
                      ti, te, cp,
                     clr, set, q,
qb);
  input  in, kin, ti, te, cp,
         clr, set ;
  output q, qb;

  GTECH_FJK3S
    U1 (.J(jin), .K(kin),
        .CP(ti), .CD(te),
        .SD(cp), .TI(clr), .TE(set),
        .Q(q), .QN(qb) );

endmodule
```



GTECH_LD1

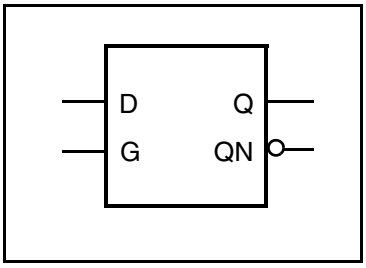
D Latch

Truth Table

D	G	Q	QN
0	1	0	1
1	1	1	0
X	0	Q	QN

Pin Description

Pin	Width	Direction
D	1	Input
G	1	Input
Q	1	Output
QN	1	Output



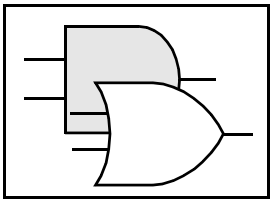
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_ld1_inst is
  port(in1, enable: in std_logic;
       q, qb  : out std_logic);
end GT_fd1_inst;

architecture sim of GT_ld1_inst is
begin
  U1 : GTECH_LD1
    port map(D => in1, G =>
enable,
             Q => q, QN => qb);
end sim;
```

Verilog

```
module GT_ld1_inst (in1, enable,
                    q, qb);
  input  in1, enable;
  output q, qb;

  GTECH_LD1
    U1 (.D(in1), .G(enable),
        .Q(q), .QN(qb) );

endmodule
```



GTECH_LD2

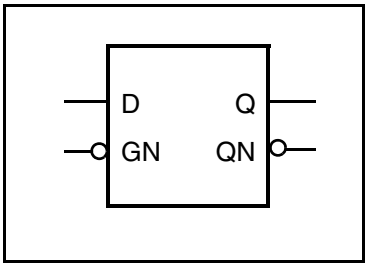
D Latch, Active Low

Truth Table

D	GN	Q	QN
0	0	0	1
1	0	1	0
X	1	Q	QN

Pin Description

Pin	Width	Direction
D	1	Input
GN	1	Input
Q	1	Output
QN	1	Output



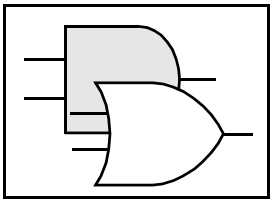
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_ld2_inst is
  port(in1, enable: in std_logic;
       q, qb  : out std_logic);
end GT_ld2_inst;

architecture sim of GT_ld2_inst is
begin
  U1 : GTECH_LD2
    port map(D => in1, GN => enable,
             Q => q, QN => qb);
end sim;
```

Verilog

```
module GT_ld2_inst (in1, enable,
                    q, qb);
  input  in1, enable;
  output q, qb;

  GTECH_LD2
    U1 (.D(in1), .GN(enable),
        .Q(q), .QN(qb) );

endmodule
```



GTECH_LD2_1

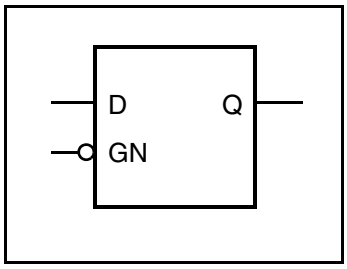
D Latch, Active Low, Single Output

Truth Table

D	GN	Q
0	0	0
1	0	1
X	1	Q

Pin Description

Pin	Width	Direction
D	1	Input
GN	1	Input
Q	1	Output



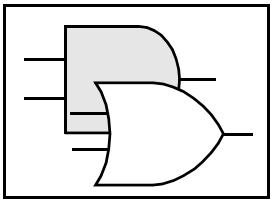
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_ld2_1_inst is
  port(in1, enable: in std_logic;
       q, : out std_logic);
end GT_ld2_1_inst;

architecture sim of GT_ld2_1_inst is
begin
  U1 : GTECH_LD2_1
    port map(D => in1, GN => enable,
             Q => q);
end sim;
```

Verilog

```
module GT_ld2_1_inst (in1, enable,
                      q);
  input  in1, enable;
  output q;

  GTECH_LD2_1
    U1 (.D(in1), .GN(enable),
        .Q(q) );

endmodule
```



GTECH_LD3

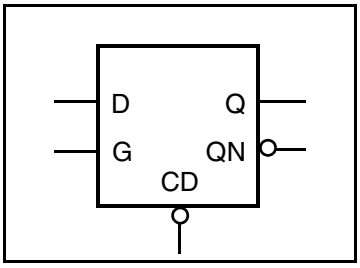
D Latch with Clear

Truth Table

D	G	CD	Q	QN
0	1	1	0	1
1	1	1	1	0
X	0	1	Q	QN
X	X	0	0	1

Pin Description

Pin	Width	Direction
D	1	Input
G	1	Input
CD	1	Input
Q	1	Output
QN	1	Output



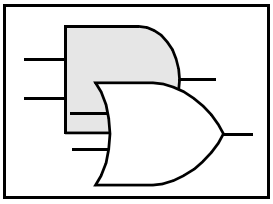
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_ld3_inst is
  port(in1, enable,
       clr : in std_logic;
       q, qb : out std_logic);
end GT_ld3_inst;

architecture sim of GT_ld3_inst is
begin
  U1 : GTECH_LD3
    port map(D => in1, G => enable,
             CD => clr,
             Q => q, QN => qb);
end sim;
```

Verilog

```
module GT_ld3_inst (in1, enable,
clr,
                    q, qb);
  input  in1, enable, clr;
  output q, qb;

  GTECH_LD3
    U1 (.D(in1), .G(enable),
        .CD(clr),
        .Q(q), .QN(qb) );

endmodule
```



GTECH_LD4

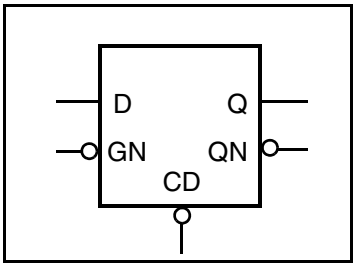
D Latch, Active Low with Clear

Truth Table

D	GN	CD	Q	QN
0	0	1	0	1
1	0	1	1	0
X	1	1	Q	QN
X	X	0	0	1

Pin Description

Pin	Width	Direction
D	1	Input
GN	1	Input
CD	1	Input
Q	1	Output
QN	1	Output



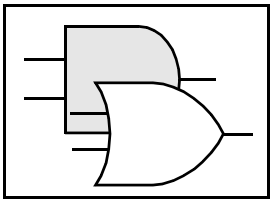
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_ld4_inst is
  port(in1, enable,
       clr : in std_logic;
       q, qb : out std_logic);
end GT_ld4_inst;

architecture sim of GT_ld4_inst is
begin
  U1 : GTECH_LD4
    port map(D => in1, GN => enable,
             CD => clr,
             Q => q, QN => qb);
end sim;
```

Verilog

```
module GT_ld4_inst (in1, enable,
clr,
                    q, qb);
  input  in1, enable, clr;
  output q, qb;

  GTECH_LD4
    U1 (.D(in1), .GN(enable),
        .CD(clr),
        .Q(q), .QN(qb) );

endmodule
```



GTECH_LD4_1

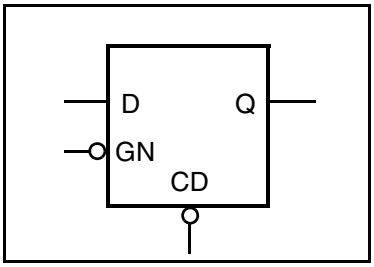
D Latch, Active Low, Single Output with Clear

Truth Table

D	GN	CD	Q
0	0	1	0
1	0	1	1
X	1	1	Q
X	X	0	0

Pin Description

Pin	Width	Direction
D	1	Input
GN	1	Input
CD	1	Input
Q	1	Output



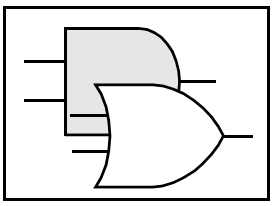
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_ld4_1_inst is
  port(in1, enable,
       clr : in std_logic;
       q : out std_logic);
end GT_ld4_1_inst;

architecture sim of GT_ld4_1_inst is
begin
  U1 : GTECH_LD4_1
    port map(D => in1, GN => enable,
             CD => clr,
             Q => q );
end sim;
```

Verilog

```
module GT_ld4_1_inst (in1, enable,
                      clr, q );
  input  in1, enable, clr;
  output q;

  GTECH_LD4_1
    U1 (.D(in1), .GN(enable),
        .CD(clr),
        .Q(q) );

endmodule
```



GTECH_LSR0

SR Latch

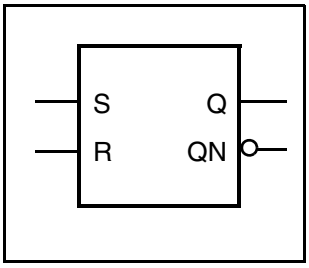
Truth Table

		Previous State		Next State	
S	**R**	**Q**	**QN**	**Q(*t*+1)**	**QN(*t*+1)**
0	0	X	X	0	0
0	1	X	X	1	0
1	0	X	X	0	1
1	1	0	0	?	?
1	1	0	1	0	1
1	1	1	0	1	0
1	1	1	1	?	?

X = Don't Care
? = Indeterminate

Pin Description

Pin	Width	Direction
S	1	Input
R	1	Input
Q	1	Output
QN	1	Output



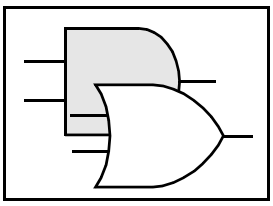
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_lsr0_inst is
  port(set, clr : in std_logic;
       q, qb    : out std_logic);
end GT_lsr0_inst;

architecture sim of GT_lsr0_inst is
begin
  U1 : GTECH_LSR0
    port map(S => set, R => clr,
             Q => q, QN => qb);
end sim;
```

Verilog

```
module GT_lsr0_inst (set, clr,
                     q, qb);
  input  set, clr;
  output q, qb;

  GTECH_LSR0
    U1 (.S(set), .R(clr),
        .Q(q), .QN(qb) );

endmodule
```

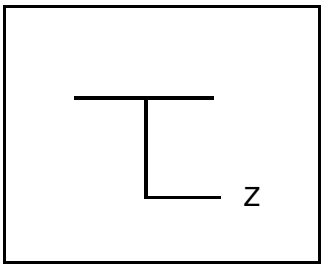


GTECH_ONE

Logic High

Pin Description and Value

Pin	Width	Direction	Value
Z	1	Output	1



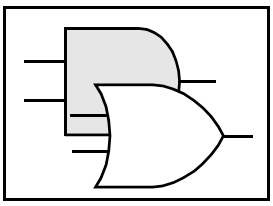
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_one_inst is
  port( out1 : out std_logic);
end GT_one_inst;

architecture sim of GT_one_inst is
begin
  U1 : GTECH_ONE
    port map( Z => out1 );
end sim;
```

Verilog

```
module GT_one_inst ( out1 );
  output out1;

  GTECH_ONE
    U1 (.Z(out1) );
endmodule
```

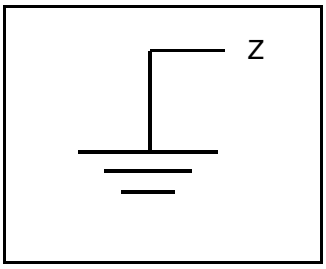


GTECH_ZERO

Logic Low

Pin Description and Value

Pin	Width	Direction	Value
Z	1	Output	0



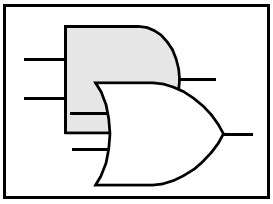
HDL Usage Through Instantiation

VHDL

```
library IEEE, GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GT_zero_inst is
  port( out1 : out std_logic);
end GT_zero_inst;

architecture sim of GT_zero_inst is
begin
  U1 : GTECH_ZERO
    port map( Z => out1 );
end sim;
```

Verilog

```
module GT_zero_inst ( out1 );
  output out1;

  GTECH_ZERO
    U1 (.Z(out1) );
endmodule
```

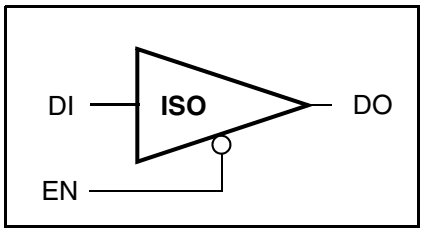


GTECH_ISO0_EN0

Isolation Buffer- Forced to 0

Truth Table

EN	DI	DO	Description
0	0	0	DI drives DO
0	1	1	DI drives DO
1	0	0	DI isolated. DO forced 0
1	1	0	DI isolated. DO forced 0



Pin Description

Pin	Width	Direction
DI	1	Input
DO	1	Output
EN	1	Input

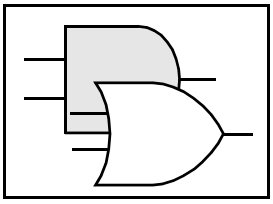
HDL Usage Through Instantiation

VHDL

```
library IEEE,GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GTECH_ISO0_EN0_inst is
      port (
       inst_EN : in std_logic;
       inst_DI : in std_logic;
       DO_inst : out std_logic
       );
    end GTECH_ISO0_EN0_inst;

architecture inst of GTECH_ISO0_EN0_inst
is

begin

    -- Instance of GTECH_ISO0_EN0
    U1 : GTECH_ISO0_EN0
   port map ( EN => inst_EN, DI =>
inst_DI, DO => DO_inst );

end inst;
```

Verilog

```
module GTECH_ISO0_EN0_inst( inst_EN,
inst_DI, DO_inst );

input inst_EN;
input inst_DI;
output DO_inst;

    // Instance of GTECH_ISO0_EN0
   GTECH_ISO0_EN0 U1 ( .EN(inst_EN),
.DI(inst_DI), .DO(DO_inst) );

endmodule
```

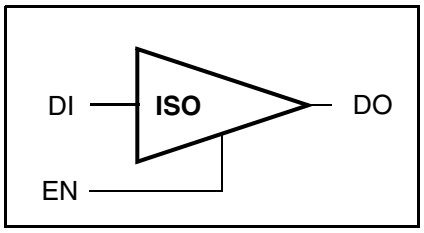


GTECH_ISO0_EN1

Isolation Buffer- Forced to 0

Truth Table

EN	DI	DO	Description
0	0	0	DI isolated. DO forced 0
0	1	0	DI isolated. DO forced 0
1	0	0	DI drives DO
1	1	1	DI drives DO



Pin Description

Pin	Width	Direction
DI	1	Input
DO	1	Output
EN	1	Input

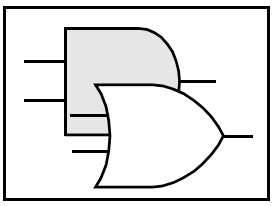
HDL Usage Through Instantiation

VHDL

```
library IEEE,GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GTECH_ISO0_EN1_inst is
      port (
       inst_EN : in std_logic;
       inst_DI : in std_logic;
       DO_inst : out std_logic
       );
    end GTECH_ISO0_EN1_inst;

architecture inst of GTECH_ISO0_EN1_inst
is

begin

    -- Instance of GTECH_ISO0_EN1
    U1 : GTECH_ISO0_EN1
   port map ( EN => inst_EN, DI =>
inst_DI, DO => DO_inst );

end inst;
```

Verilog

```
module GTECH_ISO0_EN1_inst( inst_EN,
inst_DI, DO_inst );

input inst_EN;
input inst_DI;
output DO_inst;

    // Instance of GTECH_ISO0_EN1
   GTECH_ISO0_EN1 U1 ( .EN(inst_EN),
.DI(inst_DI), .DO(DO_inst) );

endmodule
```

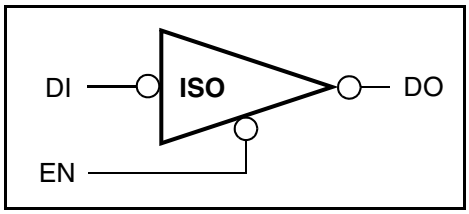


GTECH_ISO1_EN0

Isolation Buffer- Forced to 1

Truth Table

EN	DI	DO	Description
0	0	0	DI drives DO
0	1	1	DI drives DO
1	0	1	DI isolated. DO forced 1
1	1	1	DI isolated. DO forced 1



Pin Description

Pin	Width	Direction
DI	1	Input
DO	1	Output
EN	1	Input

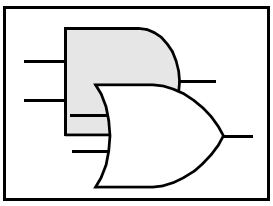
HDL Usage Through Instantiation

VHDL

```
library IEEE,GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GTECH_ISO1_EN0_inst is
      port (
       inst_EN : in std_logic;
       inst_DI : in std_logic;
       DO_inst : out std_logic
       );
    end GTECH_ISO1_EN0_inst;

architecture inst of GTECH_ISO1_EN0_inst
is

begin

    -- Instance of GTECH_ISO1_EN0
    U1 : GTECH_ISO1_EN0
   port map ( EN => inst_EN, DI =>
inst_DI, DO => DO_inst );

end inst;
```

Verilog

```
module GTECH_ISO1_EN0_inst( inst_EN,
inst_DI, DO_inst );

input inst_EN;
input inst_DI;
output DO_inst;

    // Instance of GTECH_ISO1_EN0
   GTECH_ISO1_EN0 U1 ( .EN(inst_EN),
.DI(inst_DI), .DO(DO_inst) );

endmodule
```

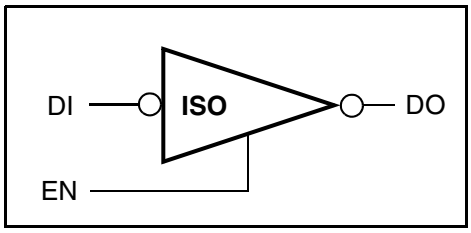


GTECH_ISO1_EN1

Isolation Buffer- Forced to 1

Truth Table

EN	DI	DO	Description
0	0	1	DI isolated. DO forced 1
0	1	1	DI isolated. DO forced 1
1	0	0	DI drives DO
1	1	1	DI drives DO



Pin Description

Pin	Width	Direction
DI	1	Input
DO	1	Output
EN	1	Input

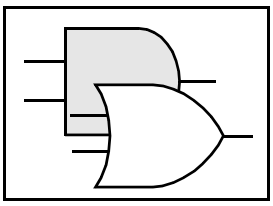
HDL Usage Through Instantiation

VHDL

```
library IEEE,GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GTECH_ISO1_EN1_inst is
      port (
       inst_EN : in std_logic;
       inst_DI : in std_logic;
       DO_inst : out std_logic
       );
    end GTECH_ISO1_EN1_inst;

architecture inst of GTECH_ISO1_EN1_inst
is

begin

    -- Instance of GTECH_ISO1_EN1
    U1 : GTECH_ISO1_EN1
   port map ( EN => inst_EN, DI =>
inst_DI, DO => DO_inst );

end inst;
```

Verilog

```
module GTECH_ISO1_EN1_inst( inst_EN,
inst_DI, DO_inst );

input inst_EN;
input inst_DI;
output DO_inst;

    // Instance of GTECH_ISO1_EN1
   GTECH_ISO1_EN1 U1 ( .EN(inst_EN),
.DI(inst_DI), .DO(DO_inst) );

endmodule
```

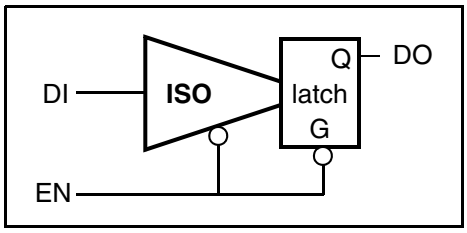


GTECH_ISOLATCH_EN0

Isolation Latch - Zero Enable

Truth Table

EN	DI	DO	Description
0	0	0	DI drives DO
0	1	1	DI drives DO
1	0	DO	DI isolated. DO latched
1	1	DO	DI isolated. DO latched



Pin Description

Pin	Width	Direction
DI	1	Input
DO	1	Output
EN	1	Input

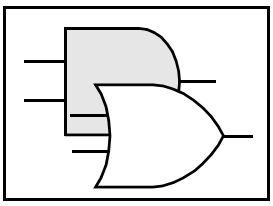
HDL Usage Through Instantiation

VHDL

```
library IEEE,GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GTECH_ISOLATCH_EN0_inst is
      port (
       inst_EN : in std_logic;
       inst_DI : in std_logic;
       DO_inst : out std_logic
       );
    end GTECH_ISOLATCH_EN0_inst;

architecture inst of
GTECH_ISOLATCH_EN0_inst is

begin
    -- Instance of GTECH_ISOLATCH_EN0
    U1 : GTECH_ISOLATCH_EN0
   port map ( EN => inst_EN, DI =>
inst_DI, DO => DO_inst );
end inst;
```

Verilog

```
module GTECH_ISOLATCH_EN0_inst(
inst_EN, inst_DI, DO_inst );

input inst_EN;
input inst_DI;
output DO_inst;

    // Instance of GTECH_ISOLATCH_EN0
    GTECH_ISOLATCH_EN0 U1 (
.EN(inst_EN), .DI(inst_DI),
.DO(DO_inst) );

endmodule
```

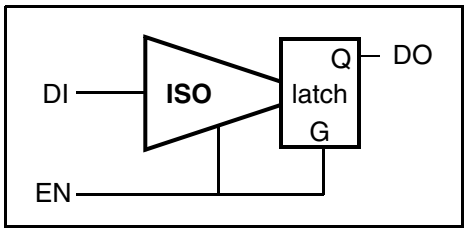


GTECH_ISOLATCH_EN1

Isolation Latch - One Enable

Truth Table

EN	DI	DO	Description
0	0	DO	DI isolated. DO latched
0	1	DO	DI isolated. DO latched
1	0	0	DI drives DO
1	1	1	DI drives DO



Pin Description

Pin	Width	Direction
DI	1	Input
DO	1	Output
EN	1	Input

HDL Usage Through Instantiation

VHDL

```
library IEEE,GTECH;
use IEEE.std_logic_1164.all;
use GTECH.GTECH_components.all;

entity GTECH_ISOLATCH_EN1_inst is
      port (
       inst_EN : in std_logic;
       inst_DI : in std_logic;
       DO_inst : out std_logic
       );
    end GTECH_ISOLATCH_EN1_inst;

architecture inst of
GTECH_ISOLATCH_EN1_inst is

begin

    -- Instance of GTECH_ISOLATCH_EN1
    U1 : GTECH_ISOLATCH_EN1
   port map ( EN => inst_EN, DI =>
inst_DI, DO => DO_inst );

end inst;
```

Verilog

```
module GTECH_ISOLATCH_EN1_inst(
inst_EN, inst_DI, DO_inst );

input inst_EN;
input inst_DI;
output DO_inst;

    // Instance of GTECH_ISOLATCH_EN1
    GTECH_ISOLATCH_EN1 U1 (
.EN(inst_EN), .DI(inst_DI),
.DO(DO_inst) );

endmodule
```