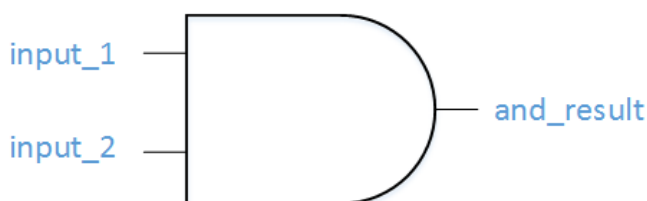


[THE BOOK](#) [THE GO BOARD](#) [FPGA-101](#) [LEARN VERILOG](#) [LEARN VHDL](#) [Q](#)

Tutorial – Introduction to VHDL

VHDL is a horrible acronym. It stands for **V**HSIC **H**ardware **D**escription Language. An acronym inside an acronym, awesome! VHSIC stands for **V**ery **H**igh **S**peed **I**ntegrated **C**ircuit. Therefore, VHDL expanded is **V**ery **H**igh **S**peed **I**ntegrated **C**ircuit **H**ardware **D**escription Language. PHEW that's a mouthful. VHDL is one of the two languages used by education and business to design FPGAs and ASICs. You might first benefit from an [introduction to FPGAs and ASICs](#) if you are unfamiliar with these fascinating pieces of circuitry. VHDL and Verilog are the two languages digital designers use to describe their circuits, and they are different by design than your traditional software languages such as C and Java.

For the example below, we will be creating a VHDL file that *describes* an And Gate. As a refresher, a simple And Gate has two inputs and one output. The output is equal to 1 only when both of the inputs are equal to 1. Below is a picture of the And Gate that we will be describing with VHDL.



An And Gate

Let's get to it! The fundamental unit of VHDL is called a **signal**. For now let's assume that a signal can be either a 0 or a 1 (there are actually other possibilities, but we will get to that). Here is some basic VHDL logic:

```
1 | signal and_gate : std_logic;  
2 | and_gate <= input_1 and input_2;
```

The first line of code defines a signal of type `std_logic` and it is called `and_gate`. `Std_logic` is the type that is most commonly used to define signals, but there are others that you will learn about. This code will generate an AND gate with a single output (`and_gate`) and 2 inputs (`input_1` and `input_2`). The keyword "and" is reserved in VHDL. The `<=` operator is known as the assignment operator. When you verbally parse the code above, you can say out loud, "The signal `and_gate` GETS `input_1` and-ed with `input_2`."

Now you may be asking yourself where `input_1` and `input_2` come from. Well as their name implies they are inputs to this file, so you need to tell the tools

Learn Verilog

[Verilog Tutorials](#) [Verilog Reserved Words \(Keywords\)](#)

Modules

[Verilog & VHDL Modules](#)

Learn VHDL

[VHDL Tutorials](#) [Introduction To VHDL for beginners with code examples](#)[Introduction to Modelsim for beginners](#)[A Very Special Keyword: Process](#)[Your First VHDL Program: An LED Blinker](#)[Recommended Coding Style for VHDL](#)[Dealing with unused signals in VHDL](#)[List of tick attributes](#)[View variables in Modelsim Waveform](#)

about them. Inputs and outputs to a file are defined in an **entity**. An entity contains a port that defines all inputs and outputs to a file. Let's create a simple entity:

```
1 | entity example_and is
2 |     port (
3 |         input_1    : in  std_logic;
4 |         input_2    : in  std_logic;
5 |         and_result : out std_logic
6 |     );
7 | end example_and;
```

This is your basic entity. It defines an entity called `example_and` and 3 signals, 2 inputs and 1 output, all of which are of type `std_logic`. One other VHDL keyword is needed to make this complete and that is **architecture**. An architecture is used to describe the functionality of a particular entity. Think of it a thesis paper: the entity is the table of contents and the architecture is the content. Let's create an architecture for this entity:

```
1 | architecture rtl of example_and is
2 |     signal and_gate : std_logic;
3 | begin
4 |     and_gate <= input_1 and input_2;
5 |     and_result <= and_gate;
6 | end rtl;
```

The above code defines an architecture called `rtl` of entity `example_and`. All signals that are used by the architecture must be defined between the "is" and the "begin" keywords. The actual architecture logic comes between the "begin" and the "end" keywords. You're almost done with this file. One last thing you need to tell the tools is which library to use. A library defines how certain keywords behave in your file. For now, just take it for granted that you need to have these 2 lines at the top of your file:

```
1 | library ieee;
2 | use ieee.std_logic_1164.all;
```

Congratulations! You have created your first VHDL file. You can see the completed file here:

```
1 | library ieee;
2 | use ieee.std_logic_1164.all;
3 |
4 | entity example_and is
5 |     port (
6 |         input_1    : in  std_logic;
7 |         input_2    : in  std_logic;
8 |         and_result : out std_logic
9 |     );
10 | end example_and;
11 |
12 | architecture rtl of example_and is
13 |     signal and_gate : std_logic;
14 | begin
15 |     and_gate <= input_1 and input_2;
16 |     and_result <= and_gate;
17 | end rtl;
```

Does it seem like you had to write a lot of code just to create a stupid and gate? First of all, and gates aren't stupid. Secondly, you are correct; VHDL is a very verbose language. Get used to the fact that doing something that was very easy in software will take you significantly longer in an HDL such as Verilog or VHDL. But just ask some software guy to try to generate an image to a VGA monitor that displays **Conway's Game of Life** and watch their head spin in amazement! By the way, that video is created with VHDL and an FPGA. You will be able to do that soon enough!

Next we will discuss another fundamental VHDL keyword: process.

Variables vs. Signals

Common VHDL Conversions

VHDL Reserved Words
(Keywords)



3 Comments



Dale Parson October 26, 2022 at 8:49 pm - Reply

Thanks. This was a very helpful example for someone in another continent trying to figure out how to get a professor's whiteboard drawings of circuits accessible to a blind student. They asked me because of this.
<https://research.library.kutztown.edu/cisfaculty/5/>



Tope January 30, 2023 at 3:08 pm - Reply

Hi Russell,

I've been trying to use your website for some weeks now, but there's a google ad banner that covers the content. Could you please do something about this? It's practically impossible to do anything and to be honest, I'd have used another website if I didn't have a go board.
Thank you.



Russell January 30, 2023 at 4:50 pm - Reply

I've turned off auto-ads on the site. Please let me know if this problem still exists!

Leave A Comment

Comment...

Name (required)

Email (required)



I'm not a robot

reCAPTCHA
[Privacy](#) - [Terms](#)

Post Comment

NAND•LAND•



ORDER NOW »