

# Vector3 ○

← gates4 (wiki/gates4)

vectorr ○ (wiki/vectorr) →

Part selection (/wiki/Vector1) was used to select portions of a vector. The concatenation operator {a, b, c} is used to create larger vectors by concatenating smaller portions of a vector together.

```
{3'b111, 3'b000} => 6'b111000
{1'b1, 1'b0, 3'b101} => 5'b10101
{4'ha, 4'd10} => 8'b10101010 // 4'ha and 4'd10 are both 4'b1010 in binary
```

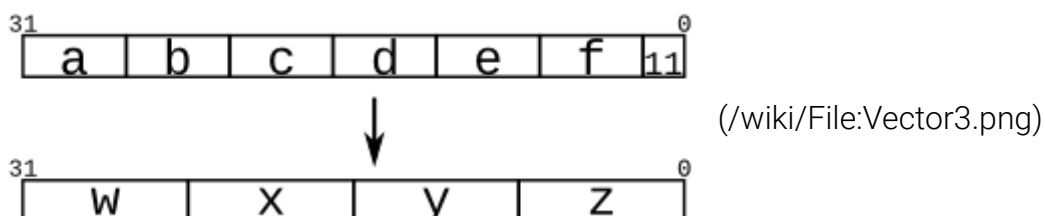
Concatenation needs to know the width of every component (or how would you know the length of the result?). Thus, {1, 2, 3} is illegal and results in the error message: `unsized constants are not allowed in concatenations`.

The concatenation operator can be used on both the left and right sides of assignments.

```
input [15:0] in;
output [23:0] out;
assign {out[7:0], out[15:8]} = in; // Swap two bytes. Right side and
left side are both 16-bit vectors.
assign out[15:0] = {in[7:0], in[15:8]}; // This is the same thing.
assign out = {in[7:0], in[15:8]}; // This is different. The 16-bit
vector on the right is extended to // match the 24-bit vector on the
left, so out[23:16] are zero. // In the first two examples,
out[23:16] are not assigned.
```

## A Bit of Practice

Given several input vectors, concatenate them together then split them up into several output vectors. There are six 5-bit input vectors: a, b, c, d, e, and f, for a total of 30 bits of input. There are four 8-bit output vectors: w, x, y, and z, for 32 bits of output. The output should be a concatenation of the input vectors followed by two 1 bits:



## Module Declaration

```
module top_module (  
    input [4:0] a, b, c, d, e, f,  
    output [7:0] w, x, y, z );
```


## Write your solution here

```
1 module top_module (  
2     input [4:0] a, b, c, d, e, f,  
3     output [7:0] w, x, y, z );//  
4  
5     // assign { ... } = { ... };  
6  
7 endmodule  
8
```

Submit

Submit (new window)

Upload a source file... 

[← gates4](/wiki/gates4)  (/wiki/gates4)

vectorr  (/wiki/vectorr) [→](#)

Retrieved from "http://hdlbits.01xz.net/mw/index.php?title=Vector3&oldid=1036"  
(http://hdlbits.01xz.net/mw/index.php?title=Vector3&oldid=1036)"










## Problem Set Contents

► Getting Started

▼ Verilog Language

► Basics

▼ Vectors

-  [Vectors \(/wiki/vector0\)](/wiki/vector0)
-  [Vectors in more detail \(/wiki/vector1\)](/wiki/vector1)
-  [Vector part select \(/wiki/vector2\)](/wiki/vector2)
-  [Bitwise operators \(/wiki/vectorgates\)](/wiki/vectorgates)
-  [Four-input gates \(/wiki/gates4\)](/wiki/gates4)
-  **[Vector concatenation operator \(/wiki/vector3\)](/wiki/vector3)**
-  [Vector reversal 1 \(/wiki/vectorr\)](/wiki/vectorr)
-  [Replication operator \(/wiki/vector4\)](/wiki/vector4)
-  [More replication \(/wiki/vector5\)](/wiki/vector5)

- ▶ Modules: Hierarchy
- ▶ Procedures
- ▶ More Verilog Features
- ▶ Circuits
- ▶ Verification: Reading Simulations
- ▶ Verification: Writing Testbenches