

1. Simple Router

Prompt

Build a router circuit which forwards data from the input (`din`) to one of four outputs (`dout0`, `dout1`, `dout2`, or `dout3`), specified by the address input (`addr`). The address is a two bit value whose decimal representation determines which output value to use. Append to `dout` the decimal representation of `addr` to get the output signal name `dout{address decimal value}`. For example, if `addr=b11` then the decimal representation of `addr` is 3, so the output signal name is `dout3`. The input has an enable signal (`din_en`), which allows the input to be forwarded to an output when enabled. If an output is not currently being driven to, then it should be set to 0.

Input and Output Signals

- `din` - Input data.
- `din_en` - Enable signal for `din`. Forwards data from input to an output if 1, does not forward data otherwise.
- `addr` - Two bit destination address. For example `addr = b11 = 3` indicates `din` should be forwarded to output value 3 (`dout3`).
- `dout0` - Output 0. Corresponds to `addr = b00`.
- `dout1` - Output 1. Corresponds to `addr = b01`.
- `dout2` - Output 2. Corresponds to `addr = b10`.
- `dout3` - Output 3. Corresponds to `addr = b11`.

```
module model #(parameter
```

DATA_WIDTH = 32

 $) ($

```
input [DATA_WIDTH-1:0] din,
```

input din_en,

input [1:0] addr,

```
output logic [DATA_WIDTH-1:0] dout0,
```

```
output logic [DATA_WIDTH-1:0] dout1,
```

```
output logic [DATA_WIDTH-1:0] dout2,
```

```
output logic [DATA_WIDTH-1:0] dout3
```

);

always $@(*)$ begin

```
dout0 = 32'h0;
```

```
dout1 = 32'h0;
```

```
dout2 = 32'h0;
```

```
dout3 = 32'h0;
```

```
if(din_en )
```

```
case(addr)
```

```
2'b00: dout0 = din;
```

```
2'b01: dout1 = din;
```

```
2'b10: dout2 = din;
```

```
2'b11: dout3 = din;
```

endcase

end

endmodule

output :-

1. Simple Router

1. Simple Router

Success: 1 of 1 passed.

Signal	Value
model.addr[1:0]	h1
model.din[31:0]	hbee
model.dout0[31:0]	h0
model.dout1[31:0]	hbee
model.dout2[31:0]	h0
model.dout3[31:0]	h0
dout0 (expected)	h0
dout1 (expected)	hbee
dout2 (expected)	h0
dout3 (expected)	h0



Status	Title	Difficulty	Companies
✓	1. Simple Router	Easy	Apple, Nvidia
-	2. Second Largest	Easy	Nvidia, Intel
-	3. Rounding Division	Medium	Apple
-	4. Gray Code Counter	Hard	Google, Qualcomm
-	5. Reversing Bits	Easy	AMD, Broadcom
-	6. Edge Detector	Easy	Nvidia, Tesla
-	7. Parallel-in, Serial-out Shift Register	Easy	Tesla, Apple
-	8. Serial-in, Parallel-out Shift Register	Easy	Nvidia, AMD
-	9. Fibonacci Generator	Easy	Cadence, Synposys
-	10. Counting Ones	Easy	Microsoft, ARM

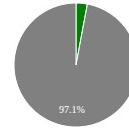
10 / Page



Page 1 of 4



Copyright © 2025 ChipDev. All rights reserved.



● Easy ● New

Easy Question Progress



Medium Question Progress



Hard Question Progress

