Modul 3

Rangkaian Kombinasional

1. Tujuan

a. Merancang rangkaian combinational menggunakan Verilog.

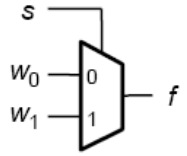
2. Materi

Rangkaian kombinasional adalah rangkaian yang outputnya hanya tergantung pada keadaan input pada saat itu. Hal ini dikarenakan rangkaian kombinasional tidak memiliki elemen penyimpan seperti pada rangkaian sekuensial. Pada bagian ini kita akan mempelajari multiplexer, decoder, encoder dan code converter.

a. Multiplexer

Multiplexer adalah rangkaian yang memiliki sejumlah input, satu atau lebih input selector, dan satu input. Multiplexer dapat berupa 2 ke 1, 4 ke 1, 8 ke 1, atau seterusnya. Banyaknya input dan lebar input selector saling berhubungan. Hal ini dikarenakan selector-lah yang akan memilih input mana yang akan dilewatkan.

Misalnya multiplexer 2 ke 1, memiliki 2 input dan 1 bit selector seperti pada gambar 1.



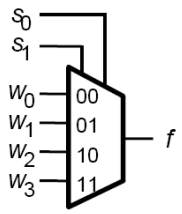
(a) Graphical symbol

s	f
0	w_0
1	w_1

(b) Truth table

Gambar 1 Multiplexer 2 ke 1

Atau multiplexer 4 ke 1 yang memiliki 4 input dan 2 bit selector



(a) Graphic symbol

s_1	s_0	f
0	0	w_0
0	1	w_1
1	0	w_2
1	1	w_3

(b) Truth table

Gambar 2 Multiplexer 4 ke 1

Multiplexer dapat diimplementasikan dalam beberapa bentuk sintak verilog, antara lain

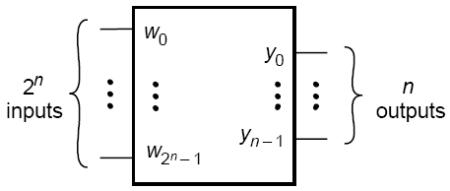
```
f = (conditional_statement_1) ? statement_1 :
    (conditional_statement_2) ? statement_2 :
                                Statement_3;

always@(sensitivity_list)
begin
        if(conditional_statement_1)
            f = statement_1;
```

```
        else if(conditional_statement_2)
            f = statement_2;
        else
            f = statement_3;
end
```

Namun demikian jika menggunakan sintaks always maka statement_1, statement_2, dan seterusnya harus dituliskan semua pada bagian sensitivity list. Jika tidak maka compiler dapat mensintesisnya sebagai latch. Oleh karena, untuk rangkaian kombinasional disarankan menggunakan sintaks assign.
Pada dasarnya, semua rangkaian kombinasional dapat dimplementasikan dengan multiplexer. Oleh karena itu, di dalam FPGA terdapat blok-blok multiplexer yang digunakan untuk mensintesis rangkaian kombinasional.

b. Encoder
Encoder memiliki fungsi untuk meng-*encode* informasi menjadi lebih *compact*. Encoder digunakan untuk mengurangi jumlah data yang diperlukan untuk merepresentasikan informasi. Misalnya dalam pengiriman informasi pada sistem digital, dengan cara ini pula kabel yang diperlukan menjadi lebih sedikit. Encoder juga berguna dalam penyimpanan informasi sehingga dengan kapasitas yang sama data yang di-*encode* dapat disimpan dalam jumlah yang lebih banyak.

1. Binary encoder
   Binary encoder meng-encode 2^n input menjadi n-bit kode, seperti pada gambar 5.



Gambar 3 Simbol binary emcoder

   Tabel kebenaran dari binary encoder 4-to-2 adalah sebagai berikut

w_3	w_2	w_1	w_0	y_1	y_0
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

Gambar 4 Tabel kebenaran binary decoder

   Pada data input hanya ada satu bit yang bernilai 1 sedangkan output akan menunjukan pada posisi bit ke berapa bit 1 pada input tersebut.

2. Priority encoder
   Pada priority encoder masing-masing input memiliki prioritas. Output dari priority encoder akan menunjukan input aktif mana yang memiliki prioritas yang paling tinggi. Ketika input dengan prioritas yang paling tinggi diberikan, input lainnya dengan prioritas yang lebih rendah akan diabaikan. Table kebenaran dari 4-to-2 priority encoder dapat kita lihat pada gambar 7.

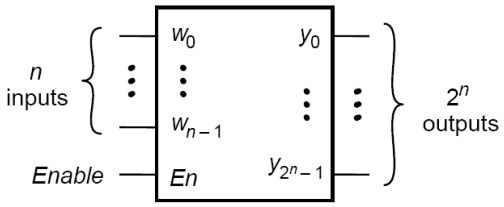
w_3	w_2	w_1	w_0	y_1	y_0	z
0	0	0	0	d	d	0
0	0	0	1	0	0	1
0	0	1	x	0	1	1
0	1	x	x	1	0	1
1	x	x	x	1	1	1

Gambar 5 Tabel kebenaran Priority encoder

Pada table ini w3 memiliki prioritas paling tinggi sedangkan w0 memiliki prioritas yang paling rendah. Sehingga ketika w3 bernilai 1, w2, w1,dan w0 diabaikan, maka output yang muncul adalah 111.
Output z diperlukan untuk menunjukan bahwa kondisi input. Nilai z akan 0 jika semua input bernilai 0

c. Decoder

Rangkaian decoder berfungsi untuk men-*decode* informasi yang diterima. Decoder berfungsi untuk mengembalikan data ke nilai semula setelah di-*encode*. Pada gambar 3 ditunjukan sebuah decoder dengan n input dan 2^n output. Meskipun memiliki banyak output decoder hanya menghasilkan satu output pada suatu saat.
Decoder juga memiliki sinyal enable yang jika En = 0 maka tidak ada input yang dilewatkan.



Gambar 6 Simbol Decoder

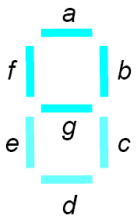
Salah satu jenis metode decoder adalah **one-hot encoding**, dimana pada satu waktu, bit pada output hanya satu yang bernilai logic 1.
Misalnya 2-to-4 decoder one-hot encoding memiliki tabel kebenaran pada gambar 4.

En	w_1	w_0	y_0	y_1	y_2	y_3
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1
0	x	x	0	0	0	0

Gambar 7 Tabel kebenaran 2-to-4 one hot decoder

d. Code converter

Code converter berguna untuk mengubah satu jenis kode menjadi kode lainnya yang lebih bermakna. Salah satu kode converter yang banyak ditemukan adalah BCD-to-7 segment decoder. BCD-to-7 segment decoder akan mengubah digit binary-coded decimal (BCD) menjadi informasi yang dapat men-*drive* input untuk menyalakan 7-segment.
Rangkaian 7-segment terdiri dari light-emitting diode (LED) yang dilabeli dari huruf a sampai f seperti pada gambar 8.



Gambar 8 7-segment dispaly

Masing-masing LED akan menyala jika diberi input tertentu. Table kebenaran BCD-to-7 segment decoder diberikan pada gambar 9.

w_3	w_2	w_1	w_0	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1

Gambar 9 Tabel Kebenaran BCD-to-7 segment decoder

Untuk masing-masing input w3,…w0, 7-segment akan menampilkan bilangan decimal yang sesuai. 6 baris terakhir dari kemungkinan 16 input tidak ditunjukan karena itu merupakan input yang tidak valid sehingga akan diabaikan oleh BCD-to-7 segment decoder.

3. Latihan

1. Simulasikan multiplexer dalam Vivado.
2. Simulasikan decoder dalam Vivado.
3. Simulasikan binary encoder dalam Vivado.
4. Simulasikan priority encoder dalam Vivado.
5. Simulasikan BCD-to-7 segment decoder dalam Vivado.