**Bab 1**

**PENDAHULUAN**

* 1. **Tujuan**

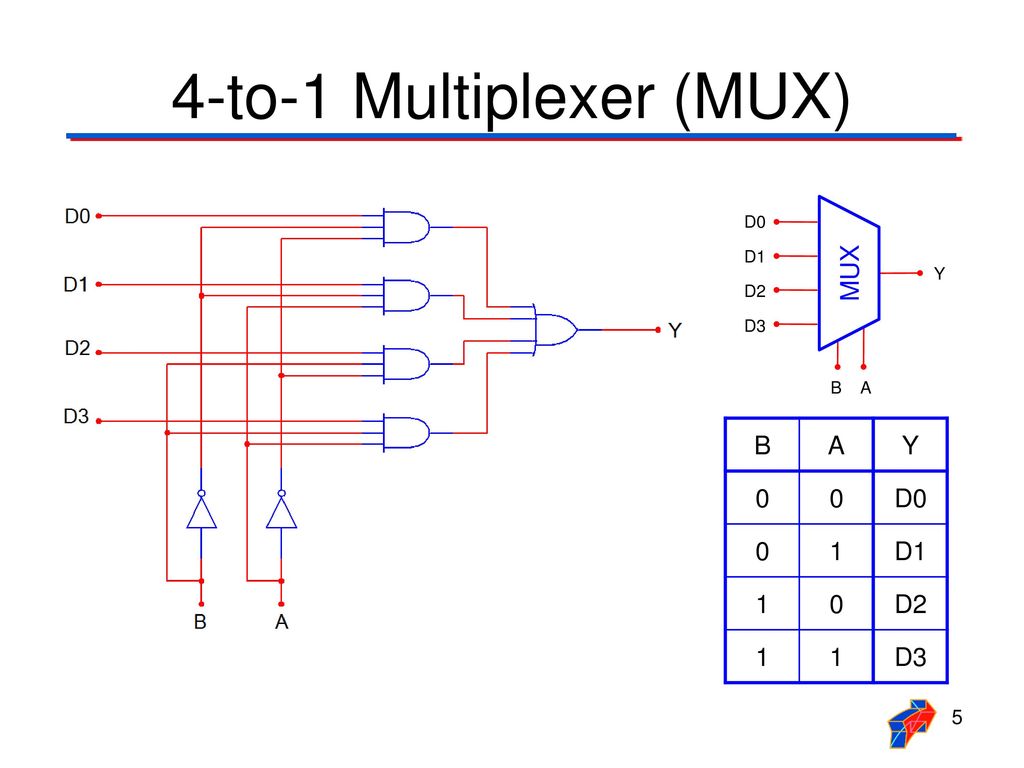
1. Mahasiswa dapat merancang rangkaian full adder dengan decoder 3 ke 8 yang menggunakan 2 buah gerbang OR
2. Mahasiswa dapat menganalisis sebuah rangkaian yang terdiri dari multiplexer, demultiplexer, encoder, decoder
   1. **Permasalahan**
3. Bagaimana cara merancang rangkaian full adder dengan decoder 3 ke 8 yang menggunakan 2 buah gerbang OR?
4. Bagaimana analisa dari sebuah rangkaian yang terdiri dari multiplexer, demultiplexer, encoder, decoder

**Bab II**

**Dasar Teori**

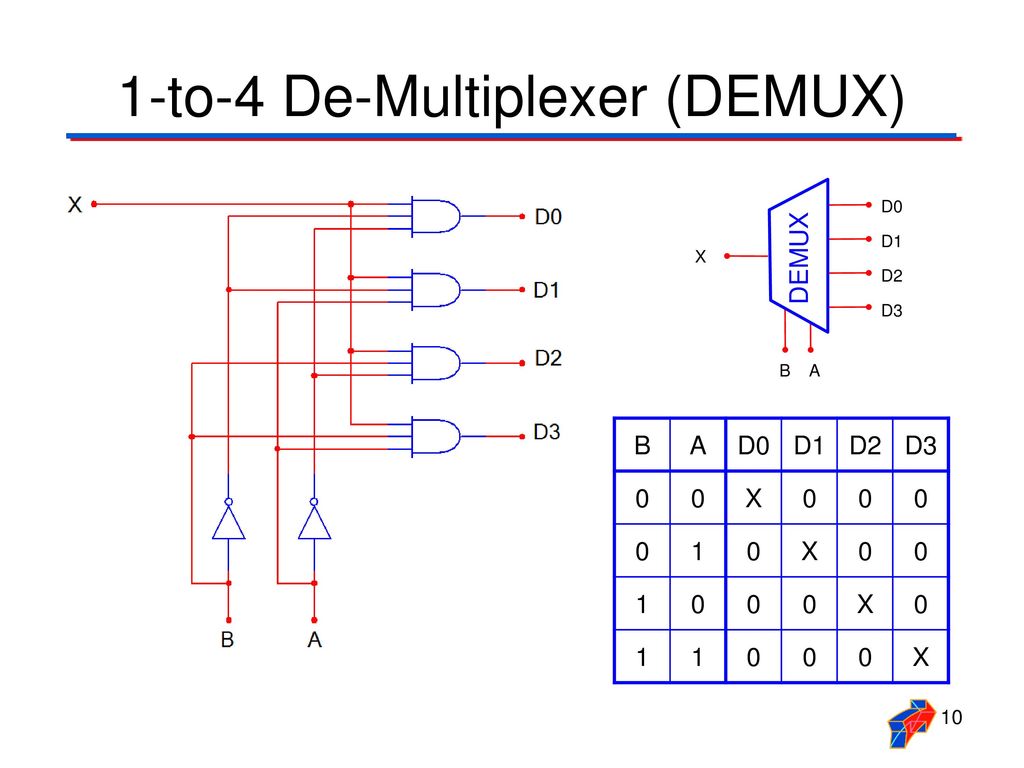
1. **Multiplexer**

Multiplexer merupakan rangkaian logika yang berfungsi memilih data yang ada pada inputnya untuk disalurkan ke outputnya dengan bantuan sinyal pemilih atau sinyal control. Kata multiplexer sering dikemukakan dalam bentuk singkatnya yakni MUX. Multiplexer disebut juga pemilih data. Jumlah input multiplexer adalah 2n (n=1,2,3,…) dengan n merupakan jumlah bit sinyal pemilih, sehingga terdapat MUX 2 ke 1 dengan 1 bit sinyal pemilih, MUX 4 ke 1 dengan 2 bit sinyal pemilih, MUX 8 ke 1 dengan 3 bit sinyal pemilih, dan seterusnya



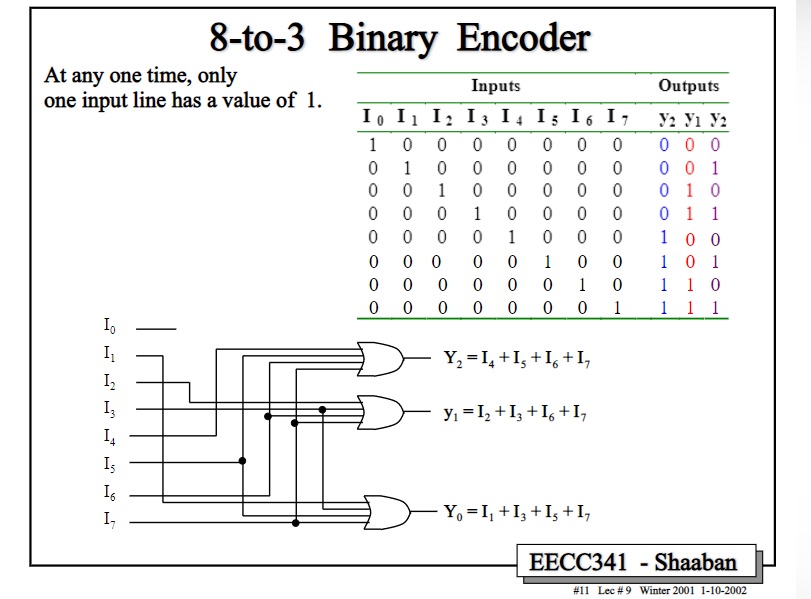
1. **Demultiplexer**

Demultiplexer merupakan rangkaian logika yang berfungsi menyalurkan data yang ada pada inputnya ke salah satu dari beberapa outputnya dengan bantuan sinyal pemilih atau sinyal control. Demultiplexer sering disebut DEMUX. Demultiplexer disebut juga sebagai penyalur data(data distributor), dan fungsinya merupakan kebalikan dari fungsi multiplexer. Jumlah output demultiplexer adalah 2n (n=1,2,3,…), dalam hal ini n merupakan jumlah sinyal bit pemilih, sehingga terdapat DEMUX 1 ke 2 dengan 1 bit sinyal pemilih, DEMUX 1 ke 4 dengan 2 bit sinyal pemilih, DEMUX 1 ke 8 dengan 3 bit sinyal pemilih, dan seterusnya



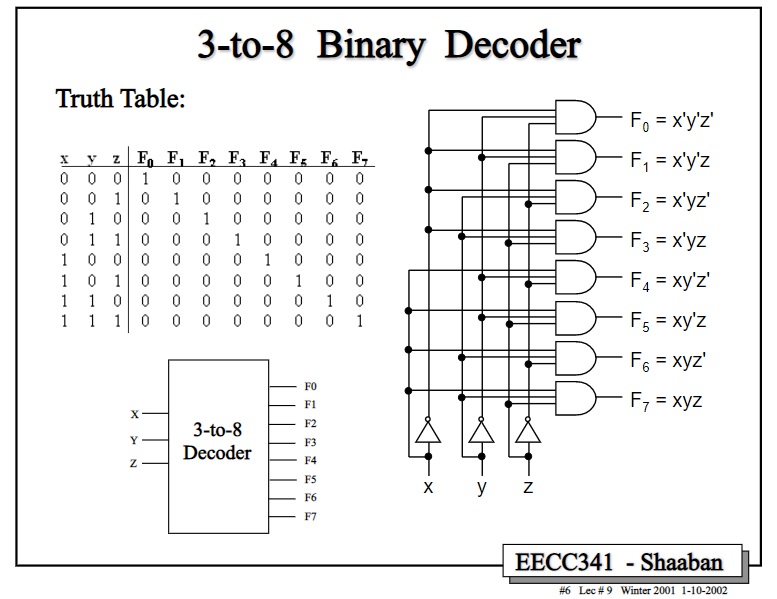
1. **Encoder**

Encoder merupakan rangkaian logika yang berfungsi mengubah data yang ada pada inputnya menjadi kode-kode biner pada outputnya. Contoh encoder octal ke biner atau disebut juga encoder 8 ke 3, berfungsi mengubah data bilangan octal pada inputnya menjadi kode biner 3 bit pada outputnya.



1. **Decoder**

Decoder merupakan rangkaian logika yang berfungsi mengkode ulang atau menafsirkan kode-kode biner yang ada ada pada inputnya menjadi data asli pada outputnya, dan fungsinya merupakan kebalikan dari fungsi encoder. Contoh: decoder 2 ke 4 berfungsi menafsirkan kode-kode biner 2 bit menjadi data asli bilangan decimal 0 sampai dengan 3. Decoder biner ke oktal atau decoder 3 ke 8 berfungsi menafsirkan kode-kode biner 3 bit menjadi data asli sistem oktal. Decoder BCD ke decimal atau decoder 4 ke 10 berfungsi menafsirkan kode-kode BCD menjadi bilangan decimal. Decoder BCD ke peraga 7 segmen berfungsi mengubah kode-kode BCD menjadi kode-kode penggerak peraga 7 segmen



**Bab III**

**PEMBAHASAN**

* 1. **Membuat Rangkaian**

Rangkaian full adder dengan decoder 3 ke 8 yang menggunakan 2 buah gerbang OR dapat dibuat dengan menggabungkan rangkaian decoder 3 ke 8 dengan implementasi dari full adder. Untuk membuat sebuah decoder 3 ke 8, kita perlu membuat table kebenarannya terlebih dahulu dan menentukan keluaran

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **A** | **B** | **C** | **y7** | **y6** | **y5** | **y4** | **y3** | **y2** | **y1** | **y0** |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

y0 = A’B’C’

y1 = A’B’C

y2 = A’BC’

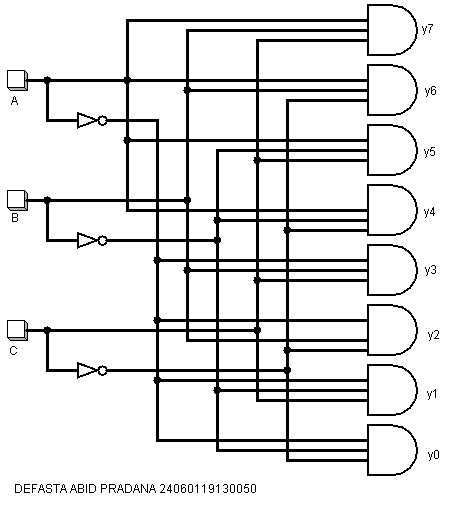
y3 = A’BC

y4 = AB’C’

y5 = AB’C

y6 = ABC’

y7 = ABC

Lalu kita buat rangkaian decoder yang diperoleh dari output diatas

Setelah itu kita membuat table kebenaran dan minterm untuk full adder

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A | B | Cin | Sum | 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 |

m0

m1

m2

m3

m4

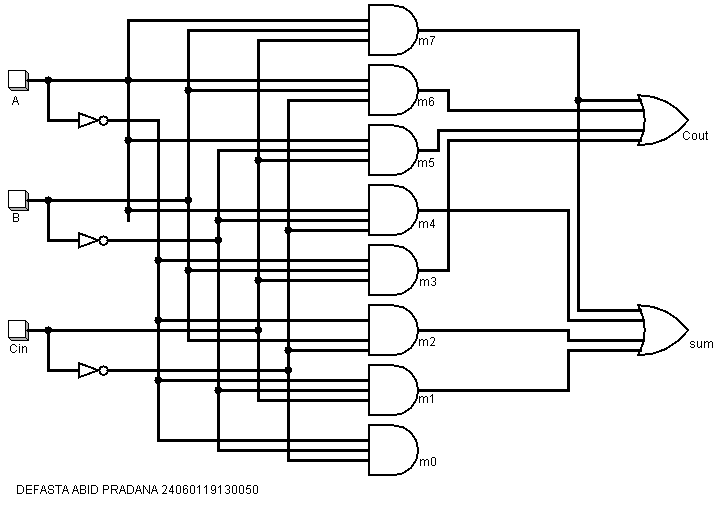
m5

m6

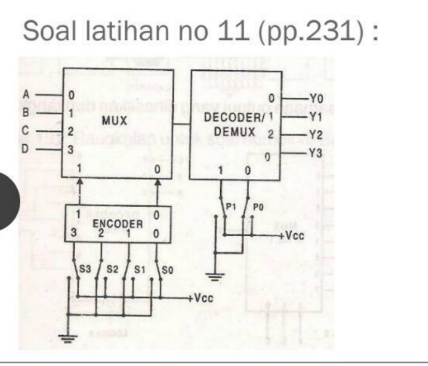
m7

Fs = ∑(m1,m2,m4,m7)

Fcout = ∑(m3,m5,m6,m7)

Kemudian kita buat rangkaian full adder menggunakan decoder 3 ke 8 yang telah dibuat

* 1. **Analisis Rangkaian**



Dari rangkaian diatas, kita dapat membuat sebuah tabel. KA menunjukan saklar yang mengarah kekanan atau dalam posisi aktif. Sedangkan KI menunjukan saklar yang mengarah ke kiri atau tidak aktif.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| SAKLAR ENCODER | | | | SAKLAR DECODER | | OUTPUT DECODER/DEMUX | | | |
| S3 | S2 | S1 | S0 | P1 | P0 | YO | Y1 | Y2 | Y3 |
| KI | KI | KI | KA | KA | KI | 0 | 0 | A | 0 |
| KI | KA | KA | KI | KA | KA | 0 | 0 | 0 | C |
| KI | KI | KA | KI | KI | KI | B | 0 | 0 | 0 |
| KA | KA | KI | KI | KI | KA | 0 | D | 0 | 0 |
| KI | KI | KA | KI | KA | KI | 0 | A | 0 | 0 |
| KI | KA | KI | KI | KI | KI | C | 0 | 0 | 0 |

Pada rangkaian diatas digunakan prinsip encoder prioritas, artinya untuk membangkitkan kode biner pada outputnya, saklar pada input tertinggi saja yang diperhatikan atau diprioritaskan. Keadaan saklar-saklar selain saklar pada input tertinggi diabaikan. Kita ambil contoh pada baris kedua tabel tersebut. Terlihat saklar encoder yang diaktifkan adalah saklar 2 dan saklar 1. Berdasarkan prinsip prioritas, maka yang perlu diperhatikan hanyalah saklar 2. Sehingga encoder tersebut menghasilkan kode biner dari 2 yaitu 10. Kemudian nilai tersebut masuk ke dalam multiplexer sebagai selector input. Hasil dari selector dengan input 10 menyebabkan data pada input 2 terpilih yaitu huruf C. Lalu huruf C tersebut masuk ke decoder/demux. Pada decoder/demux ini kembali terdapat selector untuk output. Fungsi dari selector ini adalah untuk memilih output berdasarkan input pada selector. Dalam hal ini P1 dan P2 bernilai KA atau 1, sehingga menghasilkan biner 11. Biner 11 sama dengan 3 desimal yang kemudian menentukan output yaitu pada Y3. Sehingga input huruf C tadi akan keluar pada output Y3.

**Bab IV**

**PENUTUP**

* 1. **Kesimpulan**

Dari pembahasan diatas dapat disimpulkan bahwa rangkaian full adder dengan decoder 3 ke 8 yang menggunakan 2 buah gerbang OR dapat dibuat dengan membuat decoder 3 ke 8 terlebih dahulu dengan menggunakan tabel kebenaran yang kemudian dilihat nilai dari tiap keluarannya. Lalu membuat tabel kebenaran yang kemudian ditentukan minterm dari setiap output yang dalam hal ini adalah sum dan cout. Kemudian menambahkan 2 gerbang OR yang menampung minterm dari tiap output untuk membuat rangkaian full adder.

Analisa dari rangkaian yang ada menunjukkan penggunaan dari encoder prioritas untuk menentukan output dari encoder tersebut. Kemudian output tersebut dijadikan selector pada multiplexer untuk menentukan input berapa yang dipilih. Lalu output dari multiplexer masuk ke decoder/demux sebagai input. Terdapat juga selector untuk menentukan pada output berapa yang akan dipilih untuk mengeluarkan nilai.