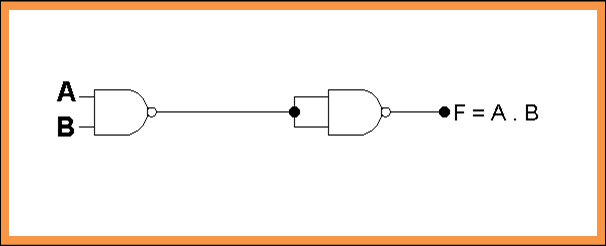
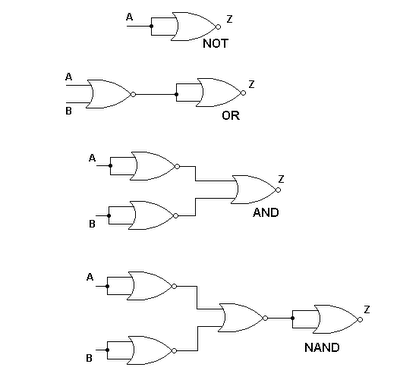
**BAB VII**

**KONVERSI GERBANG LOGIKA**

Salah satu pertimbangan utama dalam perancangan untai logika (selain jumlah elemen logika adalah tipe gerbang logika yang akan di gunakan. Gerng logika dibuat dalah sebuah IC yang berisi seumlah gerabng serupa misalnya kuad NAND dua masukan (berisi empat buah gerbang NAND dan dua masukan), dual AND empat masukan (berisi dua gerbang AND dan 4 masukan) dan kuad OR dua masukan (berisi empat gerbang OR dua masukan). Tingkat integrasi yang lebih tinggi akan menghasilkan IC yang mempunyai gerbang yang lebih banyak.

Dalam perancangan logika, gerbang logika diskrit tidak selalu di gunakan, tetapi biasanya berisi banyak gerbang. Karena itu, biasanya di sukai untuk memanfaatkan satu jenis gerbang dan bukan campuran dari beberapa gerbang untuk alasan ini konversi gerbang digunakan untuk menyatakan suatu fungsi gerbang tertentu dengan cara mengkombinasikan beberapa gerbang yang bertipe sama.

1. (b)

Fungsi not misalnya dapat di peroleh dengan sebuah gerbang NAND yang di hubungkan singkat seperti pada gambar (a) di atas. Dengan cara yang sama pada gerbang NOR yang masukan di hubungkan singkat juga akan menghasilkan gerbang NOT seperti gambar (b).

Contoh :

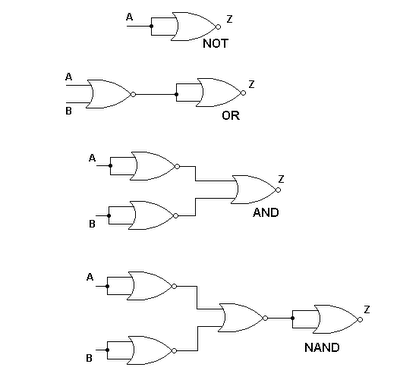
Tunjukan bahwa gerbang AND dapat di ubah hanya dengan menggunakan gerbang NAND

Jawab :

Fungsi NAND dapat diperoleh dengan membalik fungsi gerbang NAND seperti pada gambar berikut :



Dengan cara yang sama, fungsi OR dapat diperoleh dengan dua buah gerbang NOR seperti berikut :



Contoh 2 :

Tunjukan bahwa gerbang OR dapat diperoleh dengan menggunakan gerbang NAND

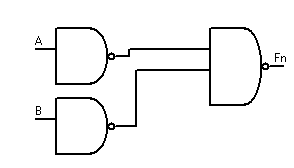
Jawab :

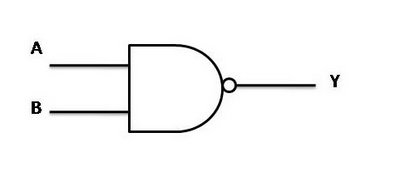
Fungsi gerbang OR dapat I nyatakan sebagai F = A+B

Dengan menggunakan teorema demorgan pertam kita peroleh :

A+B =

Persamaan di atas dapat dinyatakan dengan untai logika gambar berikut :



images.jpg A +B =  A+B

Persamaan di atas merupakan konversi yang sangat penting. Persamaan tersebut menyebutkan bahwa gerbang OR dapat di ganti dengan gerbang NAND jika setiap masukannya di komplemenkan seperti pada (b)

Contoh 3

Tunjukan bahwa fungsi AND dapat diperoleh dengan hanya menggunakan NOR

Jawab :

Fungsi AND dinyatakan sebagai F= A.B

Dengan menggunakan teorema De Morgan ke dua diperoleh :

= = ++

Pernyataan di atas dapat di nyatakan dengan gerbang logika seperti di bawah ini :



Contoh 4 :

Buatlah sebuah untai logika untuk melakukan fungsi seperti yang dinyatakan dalam tabel kebenaran pada tabel di bawah ini :

|  |  |  |  |
| --- | --- | --- | --- |
| **A** | **B** | **C** | **F** |
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |

Jawab :

Langkah pertama adalah menyusun fungsi boolean yang di sederhanakan berdasarkan tabel kebenaran tersebut (lihat contoh sebelumnya), yaitu :

Fungsi yang di sederhanakan di atas dapat di hasilkan denganmengumpulkan masing-masing komponen

, , C dan . C ke gerbang OR tiga masukan G6 seperti pada gambar di bawah ini. Masukan gerbang OR dapat diperoleh dengan cara sebagai berikut :

* Masukan A pada gerbang OR diperoleh dengan membalikan masukan dengan menggunakan gerbang pembalik G1
* B. diperoleh dengan menggunakan gerbang AND G4 dan pembalik G3
* .C diperoleh dengan menggunakan gerbang AND G5 dan pembalik G2



* Untai logika yang tersaji pada gambar di atas memerlukan tiga jenis gerbang logika yang berbeda, sehingga memerlukan tiga buah IC yang berbeda pula, karena itu tidak praktis dan tidak ekonomis.
* Untuk menghindari hal ini, untaian tersebut sebaiknya di rangkai ulang dengan hanya menggunakan sebuah tipe gerbang. Sebagai contoh : dengan hanya mengguanakan gerbang NAND, gerbang OR tiga masukan dalam gerbang di atas dapat di ganti dengan gerbang NAND U5 dengan masing-masing komponen di komplemen seperti di bawah ini :

menjadi A

B. menjadi

.C mejnadi

* Masing-masing komponen dapat diperoleh dari masukan A, B,C sebagai berikut :

Masukan A di umpankan langsung ke U5 diperoleh dengan menggunakan gerbang NAND U3 dan gerbangdiperoleh dengan gerbang NAND U4 dan gerbang pembalik U2



1. **Gerbang Tiga Keadaan**

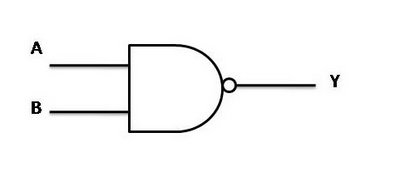
Pada penjumlahan aplikasi seperti pengiriman data di dalam sistem mikroprocessor diperlukan keadaaan tiga keadaan yang di sebut keadaan yang berimpedensi tinggi atau untai terbuka. Gerbang ini disebut sebagai gerbang tiga keadaan yang diperoleh dengan menambahkan masukan kontrol.

not.jpg

output

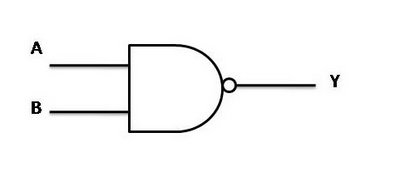
F

INPUT



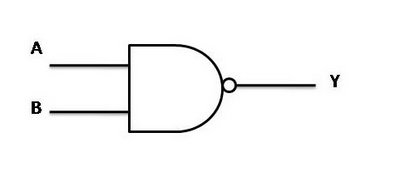
Control

(ENABLE

* Keluaran dari gerbang tiga keadaan akan valid hanya jika masukan kontrol adalah Enable
* Masukan kontrol biasanya katif rendah dalam arti bahwa gerbang akan aktif jika kontrol menuju ke logika 0.
* Jika masukan kontrol NOT ENABLE, keluaran gerbang akan dipaksa ke keadaan impedensi tinggi, atau untai terbuka, (open circuit, 0 /c)
* Gambar di atas menunjukan sebuah pembalik tiga keadaan dan jalur kontrol pada aktif rendah enable. Tabel kebenaran untuk gerbang tersebut tersaji sebagai berikut :

INPUT

OUTPUT F



CONTROL (EN)

* Berikut adalah gerbang NAND tiga keadaan dengan kontrol aktif rendah Enable

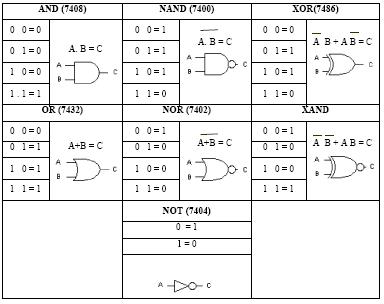
Tabel Kebenaran

|  |  |  |  |
| --- | --- | --- | --- |
| **INPUT** | | **CONTROL** | **OUTPUT** |
| **A** | **B** | **EN** | **F** |
| 0 | 0 | 1 | 0/C |
| 0 | 1 | 1 | 0/C |
| 1 | 0 | 1 | 0/C |
| 1 | 1 | 1 | 0/C |
| 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 |

1. **Penjumlahan Paruh Gerbang Logika**

Gerbang logika atau sering juga disebut gerbang logika Boolean merupakan sebuah sistem pemrosesan dasar yang dapat memproses input-input yang berupa bilangan biner menjadi sebuah *output* yang berkondisi yang akhirnya digunakan untuk proses selanjutnya. Gerbang logika dapat mengkondisikan *input - input* yang masuk kemudian menjadikannya sebuah output yang sesuai dengan apa yang ditentukan olehnya. Terdapat tiga gerbang logika dasar, yaitu : gerbang AND, gerbang OR, gerbang NOT. Ketiga gerbang ini menghasilkan empat gerbang berikutnya, yaitu : gerbang NAND, gerbang NOR, gerbang XOR, gerbang XAND.

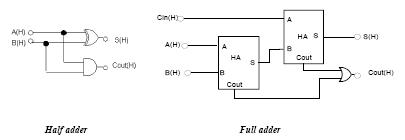
Berikut tabel kebenaran gerbang logika:



Rangkaian aritmatika dasar termasuk kedalam rangkaian kombinasional yaitu suatu rangkaian yang outputnya tidak tergantung pada kondisi *output* sebelumnya, hanya tergantung pada *present state* dari *input*.

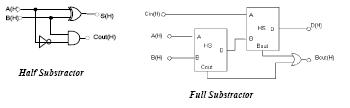
1. **Half Adder dan Full Adder**

Sebuah rangkaian kombinasional yang melaksanakan penjumlahan 2 digit biner disebut dengan *half adder*, sedangkan rangkaian yang melaksanakan penjumlahan 3 bit disebut *full adder*. Rangkaian *full adder* dapat tersusun dari dua buah *half adder*. Di pasaran rangkaian *full adder* sudah ada yang berbentuk IC, seperti 74LS83 (4-bit *full adder*).



1. **Half Substractor dan Full Substractor**

Rangkaian *half substractor* hampir sama dengan rangkaian *half adder*. D (*Difference*) ekivalen dengan S (*sum*), dan B (*borrow*) ekivalen dengan C (*carry*) pada *half adder*. Kedua rangkaian ini melakukan operasi pengurangan biner. *Half substractor* untuk pengurangan satu *bit* biner, sedangkan *full substractor* untuk pengurangan lebih dari satu *bit* biner.



1. **Decoder**

*Decoder* adalah rangkaian kombinasional logika dengan n-masukan dan 2n keluaran yang berfungsi mengaktifkan 2n keluaran untuk setiap pola masukan yang berbeda-beda. Hanya satu *output decoder* yang aktif pada saat diberi suatu *input* n-bit. Sebuah *decoder* biasanya dilengkapi dengan sebuah *input enable low* sehingga rangkaian ini bisa di on-off-kan untuk tujuan tertentu. Fungsi *enable* untuk meng-aktif-kan atau men-tidak-aktif-kan keluarannya.

1. **Priority Encoder**

Sebuah *Priority encoder* adalah rangkaian *encoder* yang mempunyai fungsi prioritas. Operasi dari rangkaian *priority encoder* adalah sebagai berikut :

jika ada dua atau lebih *input* bernilai 1 pada saat yang sama, maka *input* yang mempunyai prioritas tertinggi yang akan diambil. Kondisi x adalah kondisi *don`t care*, yang menyatakan nilai *input* bisa 1 atau 0.

1. **Multiplexer**

*Multiplexer* merupakan rangkaian logika yang berfungsi memilih data yang ada pada *input*-nya untuk disalurkan ke *output*-nya dengan bantuan sinyal pemilih atau selektor. *Multiplexer* disebut juga sebagai pemilih data (data selector). *Multiplexer* adalah rangkaian yang memiliki fungsi untuk memilih dari 2n *bit data input* ke satu tujuan *output*.