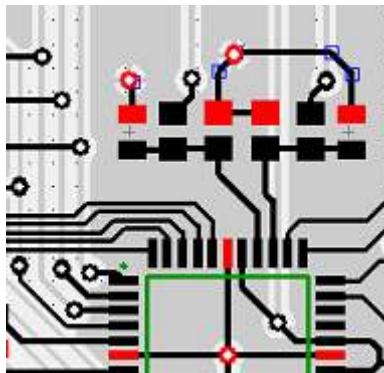


Prasetyo Priadi
prasetyopriadi@yahoo.co.id
www.PrasetyoLabs.Co.Cc

Versi Dokumen : 4.00
ISBN 0-000-000000-0

Lisensi Dokumen:

Copyright © 2008 IlmuKomputer.Com
Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.



Sistem digit merupakan bidang yang agak baru dalam dunia elektronika kurang lebih sejak awal tahun 50-an, yang pada akhirnya nanti membantu atau menambah perbendaharaan instruksi microprocessor dalam tugasnya mengolah data baik secara perangkat keras (dalam sirkuit internal microprocessor) maupun software (instruksi microprocessor yang dijalankan melalui program atau software).

Contohnya secara software, sifat khas XOR, salah satu gerbang logika sistem digit yang diadaptasikan dalam perintah microprocessor tersebut sering dimanfaatkan oleh para programmer untuk tujuan mengosongkan register tertentu (register, variable dalam microprocessor sebagai input atau output dari processor sebelum atau sesudah instruksi dijalankan).

Walaupun pengosongan register dapat dilakukan dengan perintah yang tampaknya lebih sederhana, seperti misalnya 'MOV CX,0' (untuk mengosongkan register CX), namun pada kenyataannya programmer lebih cenderung untuk menggunakan perintah 'XOR, CX,CX', karena lebih singkat hanya dua byte, sementara perintah MOV menggunakan tiga byte. Efisiensi!

Pembahasan di atas menggunakan bahasa Assembler untuk processor 16bit sedangkan untuk processor 32bit.

```
'MOV ECX,0'  
'XOR ECX,ECX'
```

Penambahan huruf E di depan nama register menjelaskan bahwa itu adalah perintah assembler untuk processor 32bit.

Saat ini sistem digit terutama dipakai dalam alat-alat digital seperti Komputer, PDA, PMP, Televisi, Telepon, sistem kendali industri dan masih banyak lagi contoh lainnya. Boleh dikatakan hampir semua peralatan modern menggunakan sistem digital untuk mengendalikan olah kerjanya.

GERBANG LOGIKA DIGITAL

Gerbang digit dikenal pula sebagai perangkat digit atau sebagai perangkat logika (logic device). Perangkat ini memiliki satu atau lebih masukan dan satu keluaran. Masing-masing masukan (input) atau keluaran (output) hanya mengenal dua keadaan logika, yaitu logika '0' (nol, rendah) atau logika '1' (satu, tinggi) yang oleh perangkat logika, '0' direpresentasikan dengan tegangan 0 sampai 0,7 Volt DC (Direct Current, arus searah), sedangkan logika '1' diwakili oleh tegangan DC setinggi 3,5 sampai 5 Volt untuk jenis perangkat logika IC TTL (Integrated Circuit Transistor-Transistor Logic) dan 3,5 sampai 15 Volt untuk jenis perangkat IC CMOS (Integrated Circuit Complementary Metal Oxyde Semiconductor).

GERBANG AND

Gerbang AND dapat memiliki dua masukan atau lebih. Gerbang ini akan menghasilkan keluaran 1 hanya apabila semua masukannya sebesar 1. Dengan kata lain apabila salah satu masukannya 0 maka keluarannya pasti 0.

Sebagai contoh, perhatikanlah kasus berikut:

Kasus

Sebuah tim ganda dari regu bulutangkis Indonesia, adalah absah apabila kedua anggotanya lengkap hadir, yaitu Amir dan Badu. Apabila *salah satu* dari Amir atau Badu ada yang absen atau tidak hadir, maka regu tersebut tidak absah untuk Mewakili Indonesia dalam turnamen bulu tangkis tersebut.

Dalam dunia logika digital, semua aspek positif dari suatu kasus diinterpretasikan sebagai true (baca: tru) suatu kata bahasa Inggris yang berarti 'benar'. Pada komputer (sebagai perangkat), 'true' diwujudkan sebagai logika '1' atau 'high' (baca: hay) = tinggi. Pada tingkat perangkat keras, 'true' mempunyai acuan tegangan listrik mendekati 5 Volt DC (dalam TTL Level).}

Pada kasus di atas, yang termasuk aspek positif adalah 'absah' dan 'hadir'.

Sebaliknya, logika digital menentukan bahwa semua aspek negatif dalam suatu kasus harus dianggap sebagai false (baca: fals) yang berarti 'salah'. Ini dimanifestasikan sebagai logika '0' atau low = rendah oleh komputer (sebagai perangkat). Perangkat keras melaksanakan hal ini dengan memberikan tegangan DC mendekati atau sama dengan nol Volt, TTL level.

Yang termasuk aspek negatif dalam hal ini adalah 'tidak absah' dan 'absen'.

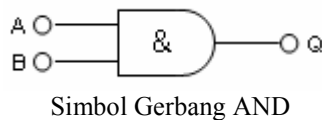
Dengan demikian, kita sudah dapat menjabarkan kasus tersebut secara logika seperti ini:

- a. Penyelesaian (output) kasus disandikan dengan 'Q'.
- b. Peserta (input), dalam hal ini Amir dan Badu, disandikan sebagai A dan B.
- c. Sinopsis yang dihasilkan menyatakan bahwa:

- Q akan true apabila A dan B true
- Q akan false bila salah satu di antara A dan B ada yang false

Bentuk logika kasus diatas disebut logika 'AND', yang dalam bahasa Indonesia berarti 'DAN'. Tampaknya, nama logika ini diperoleh dengan mengambil patokan pada sinopsis bagian pertama, yang menyatakan bahwa output akan true bila A dan B true.

Penjabaran dapat lebih disederhanakan lagi dengan mempergunakan tabel yang bernama 'Tabel Kebenaran' (truth table). Bentuk tabel kebenaran dalam kasus ini adalah sebagai berikut:



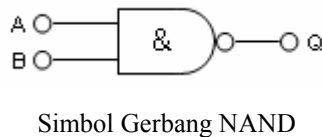
Simbol Gerbang AND

Masukan		Keluaran
A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

Tabel kebenaran logika AND

GERBANG NAND (NOT AND)

Berlawanan dengan gerbang AND, pada gerbang NAND keluaran akan selalu 1 apabila salah satu masukannya 0. Dan keluaran akan sebesar 0 hanya apabila semua masukannya 1. Gerbang NAND ekuivalen dengan NOT AND. Tabel kebenaran gerbang NAND adalah sebagai berikut.



Simbol Gerbang NAND

Masukan		Keluaran
A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0

Tabel kebenaran logika NAND

GERBANG OR

Keluaran gerbang OR akan sebesar 0 hanya apabila semua masukannya 0. Dan keluarannya akan sebesar 1 apabila saling tidak ada salah satu masukannya yang bernilai 1. Sebagai contoh, perhatikanlah kasus berikut:

Kasus

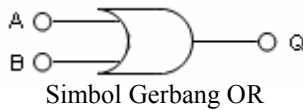
Dalam suatu rapat Universitas, Amir dan Badu bertindak sebagai wakil resmi Fakultas Teknik jurusan elektro. Sidang rapat menyatakan apabila *salah satu* dari Amir atau Badu hadir, maka hal itu sudah absah untuk mewakili fakultas tersebut.

Untuk kasus ini, penjabaran masalah tidak banyak berbeda dengan yang sebelumnya yaitu:

- Penyelesaian (output) kasus disandikan dengan 'Q'.
- Peserta (input), dalam hal ini Amir dan Badu, disandikan sebagai A dan B.
- Sinopsis yang dihasilkan menyatakan bahwa:

- Q akan true apabila salah satu dari A dan B ada dalam kondisi true.
- Q akan false, apabila A dan B (semuanya) ada dalam keadaan false.

Kasus ini memakai bentuk logika 'OR' dan tabel kebenarannya menjadi tersusun sebagai berikut:



Masukan		Keluaran
A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

Tabel kebenaran logika OR

GERBANG NOR (NOT OR)

Gerbang NOR ekuivalen dengan NOT OR. Berlawanan dengan gerbang OR, keluaran sebesar 1 hanya akan terjadi apabila semua masukannya sebesar 0. Dan keluaran 0 akan terjadi apabila terdapat masukan yang bernilai 1. Tabel kebenaran gerbang NOR.

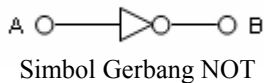


Masukan		Keluaran
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0

Tabel kebenaran logika NOR

GERBANG NOT

Pada gerbang ini nilai keluarannya selalu berlawanan dengan nilai masukannya. Apabila masukannya sebesar 0 maka keluarannya akan sebesar 1 dan sebaliknya apabila masukannya sebesar 1 maka keluarannya akan sebesar 0. Pada tabel kebenaran gerbang NOT berikut, yaitu tabel yang menggambarkan hubungan antara masukan (A) dan keluaran (B) perangkat digit gerbang NOT.



Masukan	Keluaran
A	B
0	1
1	0

Tabel kebenaran logika NOT

GERBANG XOR (Exclusive OR)

Apabila input A dan B ada dalam keadaan logika yang sama, maka output Q akan menghasilkan logika 0, sedangkan bila input A dan B ada dalam keadaan logika yang berbeda, maka output akan menjadi logika 1. XOR sebetulnya merupakan variasi dari cara kerja logika OR. Untuk lebih jelas, coba perhatikan tabel kebenarannya:



Masukan		Keluaran
A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

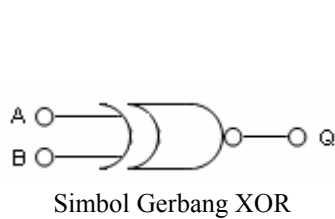
Tabel kebenaran logika XOR

Sangat mirip, bukan? Ya, karena tabel kebenaran XOR ini hanya berbeda satu langkah saja dengan tabel kebenaran OR, yaitu pada langkah terakhir saat input A dan B keduanya pada logika 1, outputnya menghasilkan 0, bukan 1 seperti pada logika OR.

Aplikasi dari proses logika XOR ini dapat dimanfaatkan untuk membandingkan dua buah data, yaitu apabila data-data tersebut mengandung informasi yang persis sama, maka XOR akan memberikan output logika 0.

GERBANG XNOR (Exclusive NOR)

Apabila input A dan B ada dalam keadaan logika yang sama, maka output Q akan menghasilkan logika 1, sedangkan bila input A dan B ada dalam keadaan logika yang berbeda, maka output akan menjadi logika 0. XNOR bisa juga dikatakan memiliki sifat dari kebalikan XOR. XNOR dan NOR hanyalah berbeda pada langkah ke-empat yaitu apabila A dan B pada logika 1 maka output Q juga 1, bukan 0 seperti pada logika NOR.



Masukan		Keluaran
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	1

Tabel kebenaran logika XNOR

PENUTUP

Semoga para pembaca sekalian mendapat tambahan wawasan dengan membaca E-Book ini. Sebuah peribahasa menyatakan 'Tiada gading yang tak retak' maka sudilah kiranya pembaca mengirimkan kritik dan saran yang membangun demi perbaikan ataupun *up dating* E-Book ini agar supaya menjadi lebih baik untuk masa kedepannya dan untuk perkembangan ilmu pengetahuan.

Kritik, saran ataupun *up dating* dapat anda kirimkan ke alamat E-mail.

prasetyopriadi@yahoo.co.id

Sekian.

DAFTAR PUSTAKA

- Hakim, Rusman, *Menjelajah Sistem Komputer dengan Debug*, Jakarta: PT. Elex Media Komputindo, 1994.
- Ibrahim, KF, *Prinsip Dasar Elektronika*, Jakarta: PT. Elex Media Komputindo, 1986.

Prasetyo Priadi



Seorang pribadi yang sangat suka, tertarik dan antusias dengan rumus empiris, algoritma sebagai penyelesaian yang logis, desain rangkaian elektronika, pemrograman, robotika dan ilmu pengetahuan yang lain.

Pengagum berat teori John von Neuman, tentang 'game of life' atau permainan hidup. Permainan bayangan (shadow play) atau pagelaran wayang kulit. Teori konspirasi dan lain sebagainya.

Senang mengkaji ilmu filsafat dan energi gelombang makna. Bertanya dengan apa, bagaimana dan mengapa sesuatu itu terjadi.

Memiliki cita-cita ingin menjadi Ilmuwan Robotika masa depan. Sekarang sedang studi di SMK Komputama Majenang. Lahir Cilacap, 17 April 1991. Berasal dari ras Jawa-Sunda. Bertempat tinggal di Majenang, Cilacap, Jawa Tengah.



Prasetyo Priadi

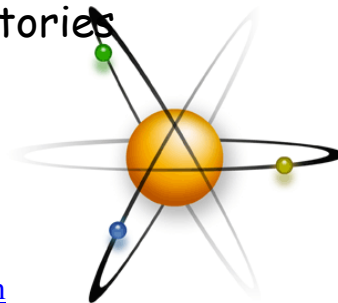
admin @ InterNETwork

Prasetyo Laboratories

- © Coco Island : www.PrasetyoLabs.Co.Cc
- © E-Book : www.LightBook.Co.Cc
- © E-Mail : [prasetyopriadi\[at\]yahoo.co.id](mailto:prasetyopriadi@yahoo.co.id)

Alternative.

- © PrasetyoLabs : <http://PrasetyoLabs.blogspot.com>
- © LightBook : <http://www.4shared.com/dir/6372876/49411bc6/sharing.html>



Pasal 14 UU No. 19 Tahun 2002 C.



"Tidak dianggap sebagai pelanggaran Hak Cipta apabila pengambilan berita aktual baik seluruhnya maupun sebagian dari kantor berita, Lembaga Penyiaran, atau surat kabar atau sumber sejenis lain, dengan ketentuan sumbernya harus disebutkan secara lengkap."