

# Rangkaian Digital untuk Teknik Elektro Vol. 1

Materi dalam buku ini dikelompokkan ke dalam lima bab yang telah disusun secara urut dan sistematis sehingga pembaca dapat memperoleh pengetahuan yang utuh terhadap rangkaian digital. Pada bab ini membahas tentang sistem bilangan dan kode, dasar gerbang logika, teknik aljabra Boolean, sinyal dan saklar elektronika, kode konverter, multiplexer, dan demultiplexer. Selain diperuntukkan bagi mahasiswa program studi Teknik Elektro, buku ajar ini juga dapat digunakan oleh para mahasiswa program studi lain yang serumpun seperti Teknik Informatika, Teknik Komputer, Sistem Informasi, Ilmu Komputer dan bahkan oleh para Ilmu Fisika dan Pendidikan Fisika yang mengambil mata kuliah sejenis dengan Rangkaian Digital.

Rangkaian Digital untuk Teknik Elektro  
Dody Susilo, RD Laksono



# Rangkaian Digital untuk Teknik Elektro Vol. 1

Dody Susilo, RD Laksono



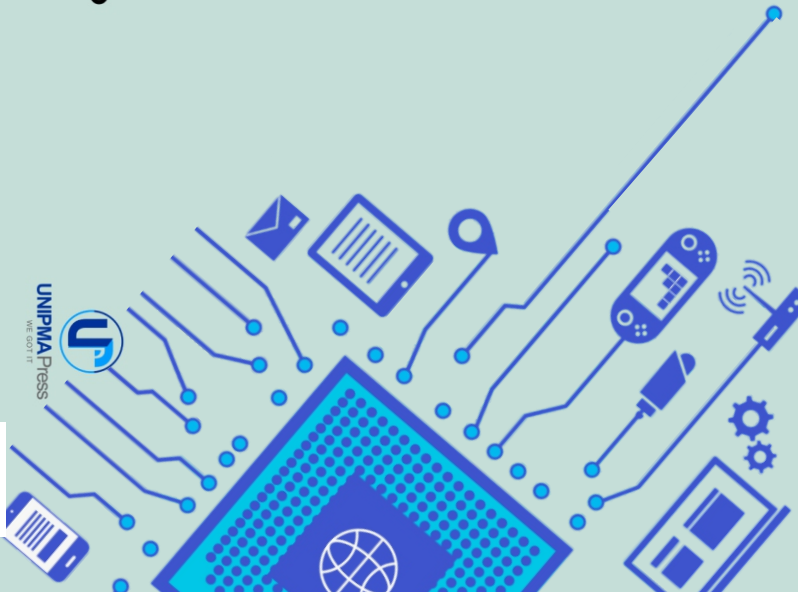
UNIPMA Press  
WE GOT IT

**Penerbit UNIPMA Press**  
Universitas PGRI Madiun  
Jl. Setia Budi No. 85, Jawa Timur 63118  
E-mail: [upress@unipma.ac.id](mailto:upress@unipma.ac.id)  
Website: [kww.unipma.ac.id](http://kww.unipma.ac.id)

ISBN 978-623-8095-38-4 (jil. 1)



9 786238 095384



# **Rangkaian Digital untuk Teknik Elektro**

## **Vol. 1**

**Dody Susilo**  
**Ridam Dwi Laksono**



**UNIPMA**Press  
WE GOT IT

# **RANGKAIAN DIGITAL UNTUK TEKNIK ELEKTRO**

## **Vol. 1**

### **Penulis:**

Dody Susilo

Ridam Dwi Laksono

### **Editor:**

Agus

### **Perancang Sampul:**

Tim Kreatif Unipma Press

### **Penata Letak:**

Dody Susilo

Cetakan Pertama, November 2023

### **Diterbitkan Oleh:**

UNIPMA Press Universitas PGRI Madiun

Jl. Setiabudi No. 85 Madiun Jawa Timur 63118

E-Mail: [upress@unipma.ac.id](mailto:upress@unipma.ac.id)

Website: [kwu.unipma.ac.id](http://kwu.unipma.ac.id)

Anggota IKAPI: No. 207/Anggota Luar Biasa/JTI/2018

**ISBN: 978-623-8095-38-4**

Hak Cipta dilindungi oleh Undang-Undang

*All right reserved*

## **PRAKARTA**

Puji syukur kami panjatkan kepada Tuhan Yang Maha Esa atas segala limpahan rahmat, hidayah, serta karunia-Nya, sehingga kami dapat menyelesaikan buku ajar ini dengan judul "Rangkaian Digital untuk Teknik Elektro".

Buku ajar ini merupakan upaya kami untuk memberikan kontribusi dalam bidang pendidikan teknik elektro, khususnya dalam memahami dan mengaplikasikan konsep-konsep dasar rangkaian digital. Perkembangan teknologi informasi dan elektronika semakin pesat, sehingga pemahaman yang mendalam mengenai rangkaian digital menjadi sangat penting bagi para mahasiswa dan praktisi teknik elektro.

Dalam buku ini, kami berusaha menjelaskan dengan jelas dan sistematis mengenai konsep dasar rangkaian digital, mulai dari logika dasar, gerbang logika, aljabar Boolean, hingga desain rangkaian kombinasional dan sekuen. Kami juga merinci penerapan praktis melalui contoh-contoh kasus yang relevan, sehingga pembaca dapat menghubungkan konsep teori dengan aplikasi nyata. Tidak lupa, apresiasi setinggi-tingginya kami sampaikan kepada berbagai pihak yang telah memberikan dukungan dalam penulisan buku ini. Terima kasih kepada institusi pendidikan, dosen, teman-teman, dan keluarga yang senantiasa memberikan motivasi dan semangat kepada penulis.

Kami menyadari bahwa buku ini masih jauh dari kesempurnaan. Oleh karena itu, saran dan kritik membangun dari para pembaca sangat kami harapkan guna perbaikan di masa yang akan datang. Semoga buku ajar ini dapat menjadi panduan yang bermanfaat bagi para pembaca dalam memahami dan menguasai konsep rangkaian digital.

Akhir kata, kami berharap buku ajar ini dapat memberikan kontribusi yang berarti dalam pengembangan ilmu pengetahuan dan teknologi di bidang teknik elektro.

Salam,

Dody Susilo

## DAFTAR ISI

PRAKARTA	ii
DAFTAR ISI	iii
DAFTAR GAMBAR	v
DAFTAR TABEL	xii
BAB I SISTEM BILANGAN DAN KODE	1
1.1 Digital versus Analog	1
1.2 Representasi Digital Kuantitas Analog	1
1.3 Sistem Penomoran Desimal (Basis 10)	5
1.4 Sistem Penomoran Biner (Basis 2)	6
1.5 Konversi Desimal ke Biner	8
1.6 Sistem Penomoran Oktal (Basis 8)	10
1.7 Konversi Oktal	11
1.8 Sistem Penomoran Heksadesimal (Basis 16)	12
1.9 Konversi Heksadesimal	13
1.10 Sistem Binary Coded Decimal	15
1.11 Perbandingan Sistem Penomoran	16
1.12 Kode ASCII	16
1.13 Aplikasi Sistem Penomoran	18
BAB II GERBANG LOGIKA DASAR	20
2.1 Gerbang AND	20
2.2 Gerbang OR	23
2.3 Analisis Timing	25
2.4 Fungsi Enable dan Disable	28
2.5 Teknik Troubleshooting	29
2.6 Inverter	33
2.7 Gerbang NAND	34
2.8 Gerbang NOR	37
2.9 Logic Gate Waveform	40
2.10 Gerbang Logika IC	42
BAB III ALJABAR BOOLEAN DAN TEKNIK REDUKSI	47
3.1 Logika Kombinasi	47
3.2 Hukum dan Aturan Aljabar Boolean	49
3.3 Penyederhanaan Rangkaian Logika Kombinasional Menggunakan Aljabar Boolean	54
3.4 Teorema De Morgan	59
3.5 Kemampuan Universal Gerbang NAND dan NOR	70

3.6	Gerbang AND–OR– INVERT untuk Mengimplementasikan Ekspresi Sum of-Products	75
3.7	Pemetaan Karnaugh	79
3.8	Aplikasi Desain Sistem	87
<b>BAB IV SINYAL DAN SAKELAR ELEKTRONIK DIGITAL</b>		<b>90</b>
4.1	Sinyal Digital	90
4.2	Clock Waveform Timing	90
4.3	Representasi Serial	92
4.4	Representasi Paralel	93
4.5	Sakelar di Sirkuit Elektronik	97
4.6	Relai sebagai Sakelar	98
4.7	Dioda Sebagai Sakelar	101
4.8	Transistor Sebagai Sakelar	103
4.9	TTL Integrated Circuit	105
4.10	Simulasi MultiSIM dari Sirkuit Sakelar	108
4.11	Sirkuit Terpadu CMOS	110
4.12	Perangkat Pemasangan Permukaan	111
<b>BAB V KODE KONVERTER, MULTIPLEXER DAN DEMULTIPLEXER</b>		<b>112</b>
5.1	Komparator	112
5.2	Komparator VHDL Menggunakan IF-THEN-ELSE	115
5.3	Decoding	117
5.4	Decoder Diimplementasikan dalam Bahasa VHDL	124
5.5	Encoding	127
5.6	Kode Konverter	130
5.7	Multiplexer	135
5.8	Demultiplexer	139
5.9	Aplikasi Desain Sistem	142
5.10	Aplikasi Desain FPGA Menggunakan LPM	148
<b>DAFTAR PUSTAKA</b>		<b>153</b>
<b>BIOGRAFI PENULIS</b>		<b>154</b>

## DAFTAR GAMBAR

Gambar 1. 1 Analog Versus Digital: (a) Bentuk Gelombang Analog; (b) Bentuk Gelombang Digital; (c) Jam Tangan Analog; (d) Jam Tangan Digital .....	1
Gambar 1. 2 (a) Representasi Digital Dari Tiga Titik Data Pada Bentuk Gelombang Analog; (b) Mengubah Tegangan Analog 2V Menjadi Rangkaian Output Digital .....	2
Gambar 1. 3 Proses Konversi Suara Analog Ke Digital Dan Kemudian Kembali Ke Analog .....	3
Gambar 1. 4 Menambahkan Derau Elektrostatis Yang Tidak Diinginkan Ke (a) Bentuk Gelombang Analog Dan (b) Bentuk Gelombang Digital .....	4
Gambar 1. 5 Sistem Pencatat Data Radiasi Matahari: (a) Diagram Blok Sistem; (b) Subsistem Pencatat Data .....	4
Gambar 1. 6 Pembagian Berurutan Dengan 2 Untuk Mengembangkan Faktor Pembobotan Biner Fraksional Dan Menunjukkan Bahwa 20 Sama Dengan 1 .....	7
Gambar 1. 7 (a) Sambungan Sirkuit Untuk Monitor Suhu Dan Tekanan Di Fasilitas Pembangkit Listrik Panas Bumi; (b) Tata Letak Data Biner Yang Dibaca Oleh Sistem Pemantauan Komputer .....	18
Gambar 2. 1 Simbol Gerbang AND Dua Masukan .....	20
Gambar 2. 2 Gerbang AND: (a) Digunakan untuk mengaktifkan alarm pencuri; (b) semua kombinasi kunci ON (K) dan pintu OPEN (D).....	21
Gambar 2. 3 Analogi Kelistrikan Untuk Gerbang AND: (a) Menggunakan Sakelar Manual; (b) Menggunakan Sakelar Transistor .....	21
Gambar 2. 4 Simbol Gerbang AND Multi-Input: (a) 4-Input; (b) 3-Input Dibentuk Dengan Dua Gerbang 2-Input; (c) 8-Input.....	22
Gambar 2. 5 Gerbang OR Dua Masukan: (a) Simbol; (b) Semua Kombinasi Masukan .....	23
Gambar 2. 6 Analogi Listrik Untuk Gerbang OR: (a) Menggunakan Sakelar Manual; (b) Menggunakan Sakelar Transistor .....	24
Gambar 2. 7 Gerbang OR Tiga Masukan: (a) Simbol; (b) Tiga Input Dibentuk Dengan Dua Gerbang 2-Input .....	24
Gambar 2. 8 Simbol Gerbang OR Delapan Masukan.....	24
Gambar 2. 9 Operasi Gerbang Dasar AND dan OR .....	25
Gambar 2. 10 Analisis Waktu Gerbang AND: (a) Sketsa Bentuk Gelombang; (b) Tampilan Penganalisa Logika Aktual .....	26
Gambar 2. 11 Gerbang Untuk Contoh 2.2 .....	26
Gambar 2. 12 Solusi untuk Contoh 2.2.....	27
Gambar 2. 13 Gerbang untuk Contoh 2.3 .....	27
Gambar 2. 14 Solusi Untuk Contoh 2.3.....	27
Gambar 2. 15 Gerbang Untuk Contoh 2.4.....	28
Gambar 2. 16 Solusi Untuk Contoh 2.4.....	28
Gambar 2. 17 Menggunakan Gerbang AND Untuk Mengaktifkan/Menonaktifkan Clock Oscillator .....	29
Gambar 2. 18 Menggunakan Gerbang OR Untuk Mengaktifkan/Menonaktifkan Clock Oscillator .....	29
Gambar 2. 19 Pulser Logika Dan Probe Logika .....	30
Gambar 2. 20 Empat Kesalahan Sirkuit Tercetak Yang Umum: (a) Misalignment Pin 14; (b) Papan Retak; (c) Jembatan Solder; (d) Transistor Yang Terbakar .....	31

Gambar 2. 21 Koneksi Untuk Troubleshooting Satu Gerbang Quad AND IC.....	32
Gambar 2. 22 Koneksi untuk troubleshooting satu gerbang OR pada IC 7432.....	33
Gambar 2. 23 Simbol Inverter dan Tabel Kebenaran .....	34
Gambar 2. 24 Analisis Waktu Gerbang Inverter: (a) Sketsa Bentuk Gelombang dan (b) Tampilan Osiloskop .....	34
Gambar 2. 25 Gerbang NAND: (a) Simbol; (b) AND–INVERT Setara Dengan Gerbang NAND Dengan $A = 1, B = 1$ .....	35
Gambar 2. 26 Input Dan Output IC NAND 7400 Quad .....	35
Gambar 2. 27 Simbol Untuk Gerbang NAND Tiga Dan Delapan Masukan.....	36
Gambar 2. 28 Gerbang untuk Contoh 2.7 .....	36
Gambar 2. 29 Analisis Waktu Gerbang NAND.....	37
Gambar 2. 30 Analisis Waktu Gerbang NAND Dengan Input Kontrol Logika Simbol .....	37
Gambar 2. 31 Bentuk Gelombang untuk Contoh 2.8.....	37
Gambar 2. 32 Simbol gerbang NOR dan persamaan OR–INVERT dengan $A = 0, B = 0$ .....	38
Gambar 2. 33 Input dan Output dari 7402 Quad NOR IC .....	38
Gambar 2. 34 Gerbang untuk Contoh 2.9 .....	39
Gambar 2. 35 Analisis Waktu Gerbang NOR.....	39
Gambar 2. 36 Gerbang Untuk Contoh 2.10.....	39
Gambar 2. 37 Analisis Waktu Gerbang NOR Tiga Masukan.....	39
Gambar 2. 38 Gerbang Untuk Contoh 2.11 .....	40
Gambar 2. 39 Persyaratan Bentuk Gelombang Input Untuk Menghasilkan Output Tertentu .....	40
Gambar 2. 40 Johnson Shift Counter Waveform Generation: (a) Sketsa Waveform; (b) Tampilan Penganalisa Logika .....	41
Gambar 2. 41 Menghasilkan Pulsa HIGH 3-Ms Menggunakan Gerbang AND dan Penghitung Shift Johnson .....	42
Gambar 2. 42 Konfigurasi Pin IC Gerbang AND Dua Masukan 7408 Quad.....	43
Gambar 2. 43 IC TTL 7408 Pada Rangkaian Aktif Clock .....	43
Gambar 2. 44 Konfigurasi Pin Untuk IC Gerbang TTL dan CMOS AND dan OR Populer Lainnya: (A) 7411 (74HC11); (B) 7421 (74HC21); (C) 7432 (74HC32) .....	44
Gambar 2. 45 Konfigurasi Pin Inverter 7404 TTL dan 4049 CMOS .....	45
Gambar 2. 46 (a) Konfigurasi Pin 7402 TTL NOR dan 4001 CMOS NOR; (b) Konfigurasi Pin 7400 TTL NAND dan 4011 CMOS NAND .....	45
Gambar 2. 47 Gerbang untuk Contoh 2.12 .....	46
Gambar 2. 48 Jawaban Rangkaian untuk Contoh 2.12.....	46
Gambar 3. 1 Persyaratan Logika Kombinasional Untuk Buzzer Peringatan Mobil .....	47
Gambar 3. 2 Sirkuit Logika Yang Diperkecil Untuk Buzzer Mobil .....	48
Gambar 3. 3 Solusi Untuk Contoh 3.1 .....	48
Gambar 3. 4 Solusi Untuk Contoh 3.2.....	49
Gambar 3. 5 Sirkuit Logika Kombinasional Untuk Contoh 3.3 .....	49
Gambar 3. 6 Hukum Komutatif Penambahan Untuk Mengatur Ulang Gerbang OR .....	50
Gambar 3. 7 Hukum Perkalian Komutatif Untuk Mengatur Ulang Gerbang AND .....	50
Gambar 3. 8 Hukum Penjumlahan Asosiatif Untuk Mengatur Ulang Pengelompokan Gerbang OR.....	50
Gambar 3. 9 Hukum Asosiatif Perkalian Untuk Mengatur Ulang Pengelompokan Gerbang AND .....	50
Gambar 3. 10 Hukum Distributif Untuk Membentuk Rangkaian Ekuivalen .....	50



Gambar 3. 11 Hukum Distributif Untuk Membentuk Rangkaian Ekuivalen (Metode FOIL).....	51
Gambar 3. 12 (a) Sirkuit Logika Untuk Alarm Pencuri Sederhana (b) Menonaktifkan Alarm Pencuri Dengan Membuat Alarm .....	51
Gambar 3. 13 Rangkaian Logika dan Tabel Kebenaran Yang Mengilustrasikan Aturan 2 .....	52
Gambar 3. 14 Rangkaian Logika dan Tabel Kebenaran Yang Mengilustrasikan Aturan 3 .....	52
Gambar 3. 15 Rangkaian Logika dan Tabel Kebenaran Yang Mengilustrasikan Aturan 4 .....	52
Gambar 3. 16 Rangkaian Logika dan Tabel Kebenaran Yang Mengilustrasikan Aturan 5 .....	52
Gambar 3. 17 Rangkaian Logika dan Tabel Kebenaran Yang Mengilustrasikan Aturan 6 .....	53
Gambar 3. 18 Rangkaian Logika dan Tabel Kebenaran Yang Mengilustrasikan Aturan 7 .....	53
Gambar 3. 19 Rangkaian Logika dan Tabel Kebenaran Yang Mengilustrasikan Aturan 8 .....	53
Gambar 3. 20 Rangkaian Logika dan Tabel Kebenaran Yang Mengilustrasikan Aturan 9 .....	53
Gambar 3. 21 Rangkaian Logika Untuk Contoh 3.5 .....	55
Gambar 3. 22 Rangkaian Logika Yang Disederhanakan Untuk Contoh 3.5 .....	56
Gambar 3. 23 Rangkaian Logika Untuk Contoh 3.6 .....	56
Gambar 3. 24 Rangkaian Logika Yang Disederhanakan Untuk Contoh 3.6 .....	56
Gambar 3. 25 Rangkaian Logika Untuk Contoh 3.7 .....	57
Gambar 3. 26 Rangkaian Yang Disederhanakan .....	58
Gambar 3. 27 Rangkaian Logika Untuk Contoh 3.8 .....	58
Gambar 3. 28 Rangkaian Yang Disederhanakan .....	59
Gambar 3. 29 Teorema De Morgan Yang Diterapkan Pada Gerbang NAND Menghasilkan Dua Tabel Kebenaran Yang Identik .....	60
Gambar 3. 30 (a) Teorema De Morgan Yang Diterapkan Pada Gerbang NOR Menghasilkan Dua Tabel Kebenaran Yang Identik; (b) Menggunakan Simbol NOR Alternatif Memudahkan Penyederhanaan Sirkuit; (c) Ringkasan Simbol Gerbang Alternatif .....	61
Gambar 3. 31 Rangkaian Logika Untuk Contoh 3.9 .....	62
Gambar 3. 32 Rangkaian Logika Yang Disederhanakan Untuk Contoh 3.9 .....	62
Gambar 3. 33 Rangkaian Logika Untuk Contoh 3.10 .....	63
Gambar 3. 34 Rangkaian Logika Yang Disederhanakan Untuk Contoh 3.10 .....	63
Gambar 3. 35 Solusi Yang Setara Dengan Contoh 3.10 .....	63
Gambar 3. 36 Rangkaian Logika Untuk Contoh 3.11 .....	64
Gambar 3. 37 Rangkaian Logika Yang Disederhanakan Untuk Contoh 3.11 .....	64
Gambar 3. 38 Rangkaian Logika Untuk Contoh 3.12 .....	65
Gambar 3. 39 Rangkaian Logika Yang Disederhanakan Untuk Contoh 3.12 .....	65
Gambar 3. 40 Rangkaian Logika Untuk Contoh 3.13 .....	66
Gambar 3. 41 Solusi Parsial Untuk Contoh 3.14 .....	66
Gambar 3. 42 Rangkaian Logika Persamaan Untuk Contoh 3.14 .....	67
Gambar 3. 43 Rangkaian Logika Untuk Persamaan Contoh 3.15 .....	68
Gambar 3. 44 (a) Sirkuit Logika Asli; (b) Rangkaian Logika Ekuivalen .....	69
Gambar 3. 45 Sirkuit Gerbang Umum Yang Digunakan Untuk Akses Memori Mikroprosesor .	70
Gambar 3. 46 Membentuk Inverter Dari NAND .....	71
Gambar 3. 47 Membentuk AND Dari Dua NAND .....	71
Gambar 3. 48 Rangkaian Logika Yang Akan Diimplementasikan Hanya Menggunakan NAND	71
Gambar 3. 49 Rangkaian Logika Ekuivalen Hanya Menggunakan NAND .....	71
Gambar 3. 50 Sambungan Eksternal Ke IC TTL 7400 Untuk Membentuk Rangkaian Gambar 3.49 .....	72
Gambar 3. 51 Membentuk OR Dari Tiga NAND .....	72

Gambar 3. 52 Membentuk NOR Dari Empat NAND.....	72
Gambar 3. 53 Membentuk Inverter Dari Gerbang NOR .....	73
Gambar 3. 54 Mengimplementasikan Fungsi $X = A + B$ Hanya Menggunakan Gerbang NOR. 73	
Gambar 3. 55 Sambungan Eksternal Ke IC CMOS 4001 Untuk Mengimplementasikan Rangkaian Gambar 3.54.....	74
Gambar 3. 56 Implementasi Sinyal.....	74
Gambar 3. 57 (a) Rangkaian Logika Yang Menghasilkan Bentuk Gelombang Di X; (b) Rangkaian Bagian (b) Digambar Ulang Hanya Menggunakan NAND.....	75
Gambar 3. 58 Rangkaian Logika Untuk Ekspresi POS .....	76
Gambar 3. 59 Rangkaian Logika Untuk Ekspresi SOP .....	77
Gambar 3. 60 Konfigurasi Pin Dan Simbol Logika Untuk Gerbang 74LS54 AOI .....	78
Gambar 3. 61 Menggunakan IC AOI Untuk Mengimplementasikan Persamaan SOP .....	78
Gambar 3. 62 Rangkaian Asli Untuk Contoh 3.18 .....	79
Gambar 3. 63 Menggunakan IC AOI Untuk Mengimplementasikan Persamaan SOP Yang Disederhanakan Untuk Contoh 3.18.....	79
Gambar 3. 64 Peta Karnaugh Dengan Dua, Tiga, Dan Empat Variabel.....	80
Gambar 3. 65 Tabel Kebenaran Dan Peta Karnaugh $X = ABC + ABC + ABC$ .....	81
Gambar 3. 66 Mengelilingi Sel Yang Berdekatan Di Peta Karnaugh .....	82
Gambar 3. 67 Peta Karnaugh Dan Persamaan Akhir Untuk Contoh 3.19 .....	83
Gambar 3. 68 Solusi untuk Contoh 3.20.....	83
Gambar 3. 69 Solusi Untuk Contoh 3.21 .....	84
Gambar 3. 70 Solusi Untuk Contoh 3.22 Mengilustrasikan Fitur Sampul .....	85
Gambar 3. 71 Solusi Untuk Contoh 3.33.....	85
Gambar 3. 72 Solusi Untuk Contoh 3.34.....	86
Gambar 3. 73 Solusi Untuk Contoh 3.35.....	87
Gambar 3. 74 (a) Persamaan Sederhana Diturunkan Dari Peta Karnaugh; (b) Implementasi Dekoder Bilangan Ganjil Menggunakan AOI .....	88
Gambar 3. 75 (a) Persamaan Sederhana Diturunkan Dari Peta Karnaugh; (b) Penerapan Alarm Tangki Bahan Kimia Menggunakan AOI.....	89
Gambar 4. 1 (a) Sinyal Digital Tipikal; (b) Osiloskop Menampilkan Bentuk Gelombang Digital Dari Instrumen Generator Clock.....	90
Gambar 4. 2 Bentuk Gelombang Clock Periodik Seperti Yang Terlihat Pada Osiloskop Yang Menampilkan Tegangan Terhadap Waktu.....	91
Gambar 4. 3 Solusi Untuk Contoh 4.3.....	92
Gambar 4. 4 Bentuk Gelombang Contoh 4.4.....	92
Gambar 4. 5 Komunikasi Serial Antar Komputer .....	93
Gambar 4. 6 Representasi serial dari bilangan biner 0110110 .....	93
Gambar 4. 7 Komunikasi Paralel Asli Antara Komputer Dan Printer.....	94
Gambar 4. 8 Representasi Paralel Dari Bilangan Biner 01101100.....	95
Gambar 4. 9 Solusi Untuk Contoh 4.5.....	95
Gambar 4. 10 Solusi Untuk Contoh 4.6.....	96
Gambar 4. 11 Sakelar Manual: (a) Sakelar Terbuka, $R = \infty Ohms$ ; (b) Sakelar Tertutup, $R = 0 Ohms$ .....	97
Gambar 4. 12 Level Output = 1 .....	97
Gambar 4. 13 Level Output = 0 .....	97

Gambar 4. 14 Representasi Fisik Dari Relai Elektromekanis: (a) Relai Tertutup Normal (NC); (b) Relai Normal Terbuka (NO); (C) Foto Relay Sebenarnya .....	98
Gambar 4. 15 Representasi Simbolik Dari Relai Elektromekanis: (a) Relai NC Digunakan Dalam Rangkaian dan (b) Relai NO Digunakan Dalam Rangkaian .....	99
Gambar 4. 16 Relai Digunakan Dalam Rangkaian Digital.....	100
Gambar 4. 17 Diagram Waktu .....	100
Gambar 4. 18 Rangkaian Untuk Contoh 4.7.....	100
Gambar 4. 19 Solusi Untuk Contoh 4.7.....	101
Gambar 4. 20 Dioda Dalam Rangkaian Seri: (a) Forward Biased dan (b) Reverse Biased .....	102
Gambar 4. 21 Katup Periksa Sistem Air.....	102
Gambar 4. 22 Kurva Karakteristik Tegangan versus Arus Dioda .....	102
Gambar 4. 23 Dioda Forward-Biased Dalam Rangkaian Listrik: (a) Rangkaian Asli dan (b) Rangkaian Ekuivalen Yang Menunjukkan Penurunan Tegangan Dioda dan $V_{out} = 5 - 0,7 = 4,3V$ .....	103
Gambar 4. 24 Transistor Bipolar NPN: (a) Tata Letak Fisik; (b) Simbol; (c) Foto .....	104
Gambar 4. 25 Saklar Transistor NPN: (a) Transistor ON dan (b) Transistor OFF.....	104
Gambar 4. 26 Rangkaian Transistor Emitor Umum Beroperasi Sebagai Inverter .....	105
Gambar 4. 27 Kalkulasi Common-Emitter .....	106
Gambar 4. 28 Skema Rangkaian Inverter TTL.....	107
Gambar 4. 29 Chip IC TTL 7404.....	107
Gambar 4. 30 Konfigurasi Pin Inverter Hex 7404.....	108
Gambar 4. 31 Foto Tiga IC Yang Umum Digunakan: 74HC00, 74ACT244, dan 74150 .....	108
Gambar 4. 32 Simulasi MultiSIM Dari Rangkaian Switching .....	109
Gambar 4. 33 Konfigurasi Pin Inverter Hex CMOS 4049.....	110
Gambar 4. 34 Perangkat Surface Mount Device (SMD) dan Tapaknya: (a) Small Outline (SO); (b) Plastic Leaded Chip Carrier (PLCC); (c) Ball Grid Array (BGA); (d) Foto SMD Yang Sebenarnya; (e) Foto SMD Yang Dipasang Pada Papan Sirkuit Tercetak .....	111
Gambar 5. 1 Komparator Biner Untuk Membandingkan Dua String Biner 4-Bit.....	113
Gambar 5. 2 Komparator Magnitudo 4-Bit 7485: (a) Konfigurasi Pin dan (b) Simbol Logika .	114
Gambar 5. 3 Perbandingan Besaran Dua String Biner 8-Bit (Atau Kata Biner) .....	114
Gambar 5. 4 Pembanding 8-Bit Mirip Dengan Gambar 5.3: (a) Daftar VHDL; (b) File Simbol Blok .....	115
Gambar 5. 5 Diagram Alur Yang Menunjukkan Eksekusi Berurutan Dalam PROSES.....	116
Gambar 5. 6 Simulasi Pembanding 8-Bit .....	117
Gambar 5. 7 Dekoder ABCD Memilih Lampu Penunjuk Desimal Yang Benar Berdasarkan Masukan BCD.....	117
Gambar 5. 8 Persyaratan Logika Untuk Menghasilkan LOW pada Output 3 Untuk Input 011.	119
Gambar 5. 9 Rangkaian Lengkap Untuk Dekoder Oktal Keluaran LOW-Aktif (1-Dari-8).....	119
Gambar 5. 10 Decoder Oktal 74138: (a) Konfigurasi Pin; (b) Simbol Logika; (c) Diagram Logika; (d) Tabel Fungsi.....	121
Gambar 5. 11 Dekoder 7442 BCD-Ke-DEC: (a) Konfigurasi Pin; (b) Simbol Logika; (c) Diagram Logika; (d) Tabel Fungsi .....	123
Gambar 5. 12 Dekoder 74154 1 Dari 16: (a) Konfigurasi Pin; (b) Simbol Logika; (c) Diagram Logika; (d) Tabel Fungsi .....	123
Gambar 5. 13 Diagram Blok Dekoder: (a) Dekoder Oktal; (b) Dekoder Oktal Dengan Input Kontrol Aktif-TINGGI.....	124

Gambar 5. 14 Decoder Oktal: (a) Program VHDL Menggunakan Persamaan Boolean; (b) File Simbol Blok; (c) Simulasi Bentuk Gelombang Yang Didekodekan .....	125
Gambar 5. 15 Dekoder Oktal Diimplementasikan Dengan Vektor dan Penetapan Sinyal Yang Dipilih: (a) Daftar VHDL; (b) File Simbol Blok .....	126
Gambar 5. 16 Dekoder Oktal Dengan Input Aktif: (a) Daftar VHDL; (b) File Simbol Blok ....	127
Gambar 5. 17 Diagram Blok Umum Untuk Pembuat Enkode: (A)a) Pembuat Enkode Desimal Ke BCD Dan (b) Pembuat Enkode Oktal Ke Biner .....	127
Gambar 5. 18 Dasar Pembuat Desimal Ke BCD Enkoder.....	128
Gambar 5. 19 Encoder 74148 Oktal-Ke-Biner (8-Baris-Ke-3-Baris): (a) Simbol Logika; (b) Tabel Fungsional.....	129
Gambar 5. 20 Faktor Pembobotan Untuk Posisi Bit BCD.....	130
Gambar 5. 21 Simbol Logika Untuk Konverter 74184 BCD-Ke-Biner .....	131
Gambar 5. 22 Konverter BCD Ke Biner Enam Bit .....	131
Gambar 5. 23 Konversi BCD ke biner menggunakan 74184 dan konversi biner ke BCD menggunakan 74185: (a) Konverter BCD ke biner selama dua dekade BCD; (b) Konverter BCD-ke-biner selama tiga dekade BCD; (c) Konverter biner-ke-BCD 6-bit; (d) Konverter biner-ke-BCD 8-bit.....	132
Gambar 5. 24 Gray Code Wheel.....	134
Gambar 5. 25 Konverter Kode Biner Ke Gray .....	134
Gambar 5. 26 Konverter Kode Gray Ke Biner .....	134
Gambar 5. 27 Diagram Fungsional Multiplexer Empat Baris.....	135
Gambar 5. 28 Diagram Logika Untuk Multiplexer Empat Baris.....	136
Gambar 5. 29 Multiplexer Delapan Baris 74151: (a) Simbol Logika; (b) Diagram Logika .....	137
Gambar 5. 30 Multiplexer 4 Baris Pada Contoh 8–19: (a) Daftar VHDL; (b) Block Simbol File .....	138
Gambar 5. 31 Bentuk Gelombang Yang Disimulasikan Untuk Multiplexer 4 Baris Pada Contoh 8–19.....	138
Gambar 5. 32 Demultiplexer 4 Baris: (a) Diagram Fungsional; (b) Simulasi Bentuk Gelombang .....	139
Gambar 5. 33 Demultiplexer 4 Baris Ganda 74139: (a) Simbol Logika; (b) Diagram Logika ..	140
Gambar 5. 34 Koneksi Untuk Merutekan Sinyal Data Masukan Ke Keluaran 2a Demultiplexer 74139.....	140
Gambar 5. 35 Koneksi Demultiplexer 74154 Untuk Merutekan Sinyal Input Ke Output 5 .....	141
Gambar 5. 36 Multiplexer/Demultiplexer Analog CMOS 4051 .....	142
Gambar 5. 37 74HCT138 Untuk Decoder Alamat Memori Dalam Sistem Mikroprosesor 8085A .....	143
Gambar 5. 38 74148 Untuk Mengkodekan Alarm Aktif Untuk Dipantau Oleh Mikrokontroler	144
Gambar 5. 39 Multiplexer, Demultiplexer, dan Mikrokontroler Untuk Menyediakan Kemampuan Komunikasi Ke Beberapa Terminal Data Serial.....	145
Gambar 5. 40 Multiplexer Analog 74HCT4051 Digunakan Sebagai Generator Tangga.....	147
Gambar 5. 41 Diagram Blok Tampilan Empat Digit Multipleks.....	148
Gambar 5. 42 Modul LPM Untuk Komparator 8-Bit .....	149
Gambar 5. 43 Bentuk Gelombang Simulasi Untuk Pembandingan LPM .....	149
Gambar 5. 44 Modul LPM Untuk Dekoder Oktal .....	150
Gambar 5. 45 Bentuk Gelombang Simulasi Untuk Dekoder LPM .....	151
Gambar 5. 46 Modul LPM Untuk Multiplexer 8 Jalur .....	151

Gambar 5. 47 Bentuk Gelombang Simulasi Untuk Multiplekser LPM..... 152

## DAFTAR TABEL

Tabel 1. 1 Kekuatan Dari 2 Faktor Pembobotan Biner.....	6
Tabel 1. 2 Sistem Penomoran Oktal .....	10
Tabel 1. 3 Sistem Penomoran Heksadesimal .....	13
Tabel 1. 4 Perbandingan Sistem Penomoran .....	16
Tabel 1. 5 Kode Standar Amerika untuk Pertukaran Informasi.....	17
Tabel 2. 1 Tabel Kebenaran Gerbang AND Dua Masukan .....	20
Tabel 2. 2 Tabel Kebenaran Gerbang AND Empat Masukan .....	22
Tabel 2. 3 Tabel Kebenaran Gerbang OR Dua Masukan .....	23
Tabel 2. 4 Tabel Kebenaran Gerbang OR Tiga Masukan.....	25
Tabel 2. 5 Status Probe Logika .....	30
Tabel 2. 6 Tabel Kebenaran Gerbang NAND Dua Masukan .....	36
Tabel 2. 7 Tabel Kebenaran Untuk Gerbang NAND Tiga Masukan.....	36
Tabel 2. 8 Tabel Kebenaran Gerbang NOR.....	38
Tabel 3. 1 Menggunakan Tabel Kebenaran Untuk Membuktikan Persamaan pada Aturan 10....	54
Tabel 3. 2 Hukum dan Aturan Boolean Untuk Pengurangan Rangkaian Logika Kombinasional	54
Tabel 3. 3 Tabel Kebenaran Untuk Contoh 3.14 .....	67
Tabel 3. 4 Tabel Kebenaran Untuk Contoh 3.15 .....	68
Tabel 3. 5 Tabel Kebenaran Diselesaikan Menggunakan Ekspresi SOP.....	76
Tabel 3. 6 Tabel Kebenaran Hex Digunakan Untuk Menentukan Persamaan Untuk Angka Ganjil Dari 0 Hingga 9.....	88
Tabel 4. 1 Prefiks Rekayasa Umum.....	91
Tabel 5. 1 Tabel Kebenaran Untuk Decoder Oktal.....	118
Tabel 5. 2 IC Decoder .....	120
Tabel 5. 3 Tabel Kebenaran Desimal ke BCD Encoder .....	128
Tabel 5. 4 Four-Bit Gray Code .....	133
Tabel 5. 5 Kode Input Pemilihan Data untuk Gambar 5.27.....	135
Tabel 5. 6 Multiplexer TTL dan CMOS .....	136

# BAB I

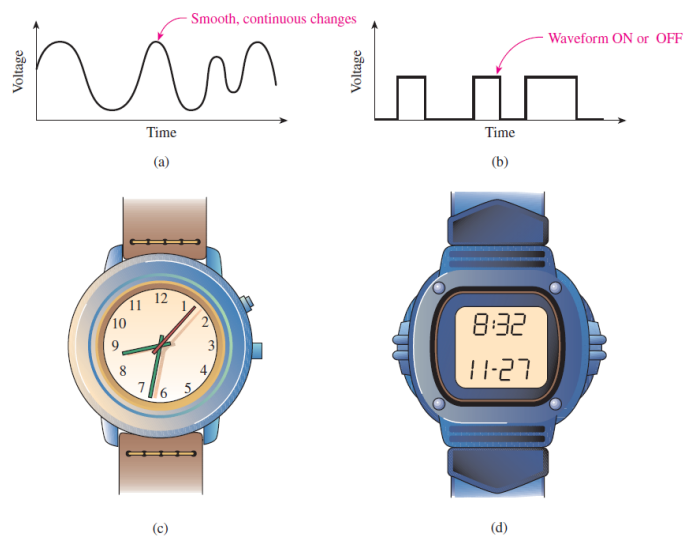
## SISTEM BILANGAN DAN KODE

### 1.1 Digital versus Analog

Sistem digital beroperasi pada digit diskrit yang mewakili angka, huruf, atau simbol. Mereka berurusan secara ketat dengan status *ON* dan *OFF*, yang dapat wakili dengan 0 dan 1. Sistem analog mengukur dan merespons besaran listrik atau fisik yang bervariasi secara terus menerus. Perangkat analog terintegrasi secara elektronik ke dalam sistem untuk terus memantau dan mengontrol besaran seperti suhu, tekanan, kecepatan, dan posisi dan untuk menyediakan kontrol otomatis berdasarkan level besaran ini. Gambar 1.1 menunjukkan beberapa contoh besaran digital dan analog.

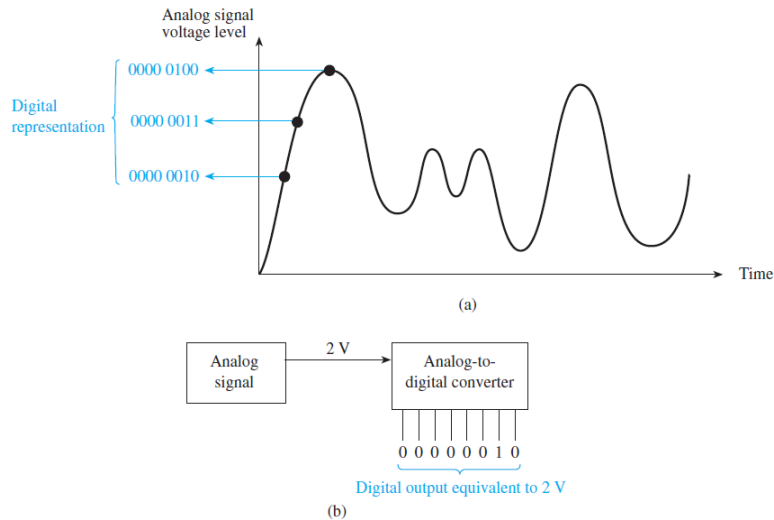
### 1.2 Representasi Digital Kuantitas Analog

Sebagian besar kuantitas fisik yang terjadi secara alami di dunia bersifat analog. Sinyal analog adalah kuantitas listrik atau fisik yang terus berubah. Pikirkan tentang termometer tabung berisi merkuri; saat suhu naik, merkuri mengembang dengan cara analog dan membuat gerakan yang mulus dan terus menerus relatif terhadap skala yang diukur dalam derajat. Seorang pemain bisbol mengayunkan pemukul dengan gerakan analog. Kecepatan dan kekuatan yang digunakan musisi untuk memukul tuts piano bersifat analog. Bahkan getaran yang dihasilkan dari senar piano adalah getaran analog sinusoidal.



Gambar 1. 1 Analog Versus Digital: (a) Bentuk Gelombang Analog; (b) Bentuk Gelombang Digital; (c) Jam Tangan Analog; (d) Jam Tangan Digital

Jadi mengapa perlu menggunakan representasi digital di dunia analog alami? Jawabannya adalah jika menginginkan mesin elektronik untuk menafsirkan, mengkomunikasikan, memproses, dan menyimpan informasi analog, akan lebih mudah bagi mesin untuk menanganinya jika terlebih dahulu mengubah informasi tersebut menjadi format digital. Nilai digital diwakili oleh kombinasi level tegangan *ON* dan *OFF* yang ditulis sebagai string 1 dan 0.



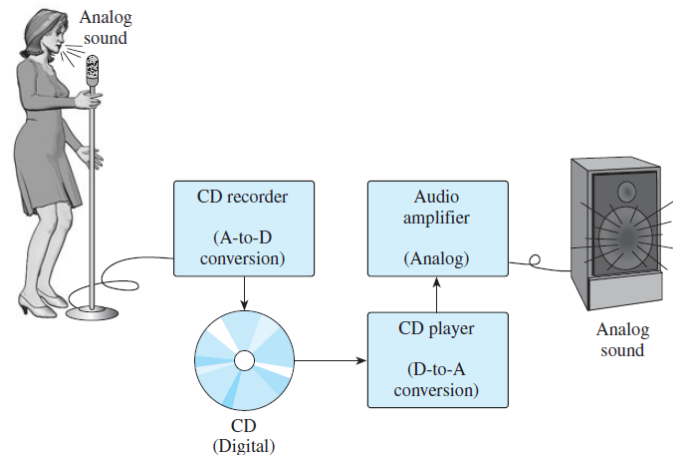
Gambar 1. 2 (a) Representasi Digital Dari Tiga Titik Data Pada Bentuk Gelombang Analog; (b) Mengubah Tegangan Analog 2V Menjadi Rangkaian Output Digital

Misalnya, termometer analog yang mencatat  $72^{\circ}\text{F}$  dapat direpresentasikan dalam rangkaian digital sebagai rangkaian level tegangan *ON* dan *OFF*. Fitur yang mudah digunakan dalam menggunakan level tegangan *ON/OFF* adalah sirkuit yang digunakan untuk menghasilkan, memanipulasi, dan menyimpannya sangat sederhana. Alih-alih berurusan dengan rentang dan interval tak terbatas dari level tegangan analog, yang perlu digunakan hanyalah tegangan *ON* atau *OFF* (biasanya  $5\text{V} = \text{ON}$  dan  $0\text{V} = \text{OFF}$ ).

Contoh yang baik dari penggunaan representasi digital dari kuantitas analog adalah rekaman audio musik. *Compact Disk (CD)* dan *Digital Versatile Disk (DVD)* adalah hal yang biasa dan terbukti menjadi alat yang unggul untuk merekam dan memutar musik. Alat musik dan suara manusia menghasilkan sinyal analog, dan telinga manusia secara alami merespons sinyal analog. Meskipun prosesnya membutuhkan kerja ekstra, industri rekaman mengubah sinyal analog menjadi format digital dan kemudian menyimpan informasinya dalam *CD* atau *DVD*. Pemutar *CD* atau *DVD* kemudian mengubah level digital kembali ke sinyal analog yang sesuai sebelum memutarnya kembali untuk telinga manusia.

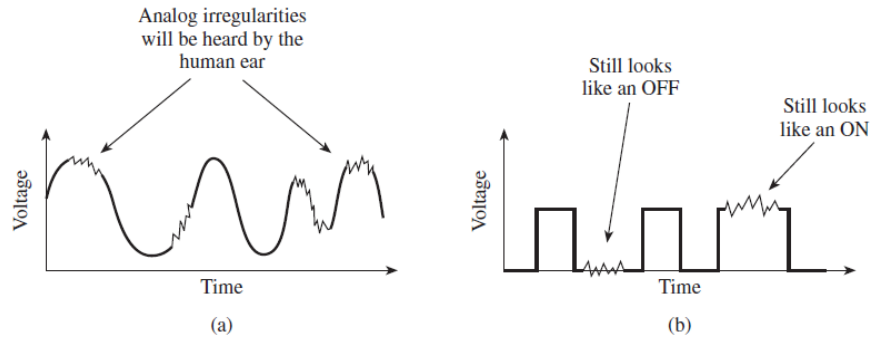


Untuk secara akurat mewakili sinyal musik yang kompleks sebagai string digital (rangkaiannya 1 dan 0), beberapa sampel sinyal analog harus diambil, seperti yang ditunjukkan pada Gambar 1–2(a). Konversi pertama yang diilustrasikan adalah pada titik di bagian naik dari sinyal analog. Pada titik tersebut, tegangan analog adalah 2V. Dua volt diubah menjadi rangkaian digital 0000 0010, seperti yang ditunjukkan pada Gambar 1.2(b). Konversi berikutnya diambil saat sinyal analog pada Gambar 1.2(a) masih naik, dan konversi ketiga diambil pada level tertinggi. Proses ini berlanjut sepanjang seluruh karya musik yang akan direkam. Untuk memutar ulang musik, prosesnya dibalik. Konversi digital ke analog dibuat untuk membuat ulang sinyal analog asli. Jika jumlah sampel yang cukup tinggi diambil dari sinyal analog asli, reproduksi musik asli yang hampir persis dapat dibuat.



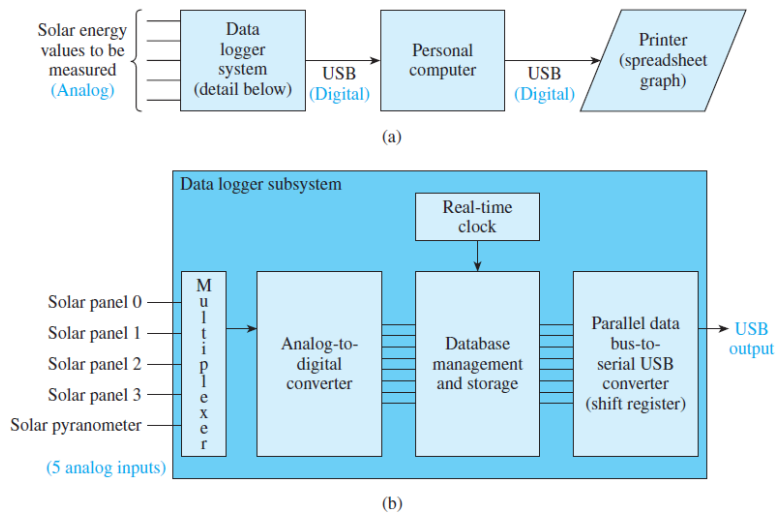
Gambar 1. 3 Proses Konversi Suara Analog Ke Digital Dan Kemudian Kembali Ke Analog

Ini tentu saja pekerjaan ekstra, tetapi rekaman digital hampir menghilangkan masalah seperti kebisingan elektrostatis dan desisan pita magnetik yang terkait dengan metode perekaman audio sebelumnya. Masalah-masalah ini telah diatasi karena, ketika ketidaksempurnaan diperkenalkan ke sinyal digital, sedikit variasi pada level digital tidak mengubah level *ON* ke level *OFF*, sedangkan sedikit perubahan pada level analog mudah ditangkap oleh telinga manusia, seperti yang ditunjukkan pada Gambar 1.4.



Gambar 1. 4 Menambahkan Derau Elektrostatis Yang Tidak Diinginkan Ke (a) Bentuk Gelombang Analog Dan (b) Bentuk Gelombang Digital

Aplikasi lain dari representasi digital besaran analog adalah data logging dari sumber energi alternatif. Sangat penting bagi teknisi energi untuk melacak efisiensi sistem pengumpulan energi mereka. Dalam kasus sistem solarcollection yang ditunjukkan pada Gambar 1.5(a) dan (b), efisiensi sistem dapat ditentukan dengan membagi jumlah watt yang dihasilkan oleh panel surya fotovoltaik (PV) dengan total energi matahari (radiasi) yang mencolok. panel. Namun, karena semua kuantitas yang terjadi secara alami seperti matahari, angin, suhu, dan tekanan adalah nilai analog, perlu mengubahnya menjadi representasi digital sebelum dapat dipahami oleh sistem komputer.



Gambar 1. 5 Sistem Pencatat Data Radiasi Matahari: (a) Diagram Blok Sistem; (b) Subsystem Pencatat Data

Pada Gambar 1.5(a) ada lima input besaran matahari analog ke sistem pencatatan data. Pencatat data mendigitalkan nilai-nilai ini dan mengeluarkannya sebagai aliran data dalam format

*USB (Universal Serial Bus)* ke komputer pribadi, yang kemudian dapat digunakan untuk menganalisis data melalui spreadsheet untuk menentukan efisiensi.

Detail dari sistem pencatatan data ditunjukkan pada Gambar 1.5(b). Ini menunjukkan input ke sistem sebagai empat panel surya *PV* dan satu piranometer surya. Piranometer digunakan untuk mengukur energi matahari yang mengenai bumi di lokasi tersebut dalam watt-permeter<sup>2</sup>. Karena panel surya *PV* mengubah sinar matahari menjadi daya (Watt), setiap panel juga menyediakan tegangan analog yang sebanding dengan watt yang dihasilkan. Keempat nilai analog ini terhubung ke multiplexer, yang secara bergantian merutekan masing-masing besaran analog, satu per satu, ke *Analog Digital Converter (ADC)*. Saat setiap nilai diterima, *ADC* mengeluarkan output yang setara dengan angka digital 8-bit (tersedia konverter *ADC* 8, 10, 12 dan bit yang lebih tinggi). Data ini perlu diberi stempel waktu untuk membantu teknisi melacak efisiensi pada waktu yang berbeda dalam satu hari dan modifikasi lain yang mungkin dilakukannya pada panel di siang hari. Sirkuit jam digital real-time menyediakan stempel waktu ini.

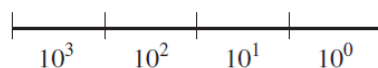
Terakhir, sebelum data logger dapat berkomunikasi dengan *PC*, data digital yang sekarang dalam format “paralel” harus diubah menjadi format “serial” untuk memenuhi standar *USB* yang digunakan oleh *PC*. Konversi paralel ke serial ini dibuat oleh register geser. Bagian berikut mengajarkan cara mengembangkan dan menginterpretasikan kode biner yang digunakan dalam sistem digital.

### 1.3 Sistem Penomoran Desimal (Basis 10)

Dalam sistem penomoran desimal, setiap posisi berisi 10 kemungkinan digit yang berbeda. Angka-angka ini adalah 0, 1, 2, 3, 4, 5, 6, 7, 8, dan 9. Setiap posisi dalam angka multidigit akan memiliki faktor pembobotan berdasarkan pangkat 10.

#### Contoh 1.1

Dalam angka desimal empat digit, posisi paling signifikan (paling kanan) memiliki faktor bobot  $10^0$ ; posisi paling signifikan (paling kiri) memiliki faktor bobot  $10^3$ :



Dimana:

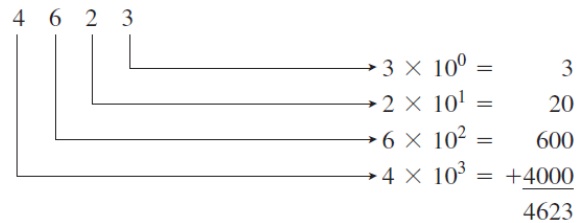
$$10^3 = 1000$$

$$10^2 = 100$$

$$10^1 = 10$$

$$10^0 = 1$$

Untuk mengevaluasi angka desimal 4623, angka di setiap posisi dikalikan dengan faktor bobot yang sesuai:



Contoh 1.1 mengilustrasikan prosedur yang digunakan untuk mengonversi dari beberapa sistem bilangan ke persamaan desimalnya (basis 10). Sekarang lihat basis 2 (biner), basis 8 (oktal), dan basis 16 (heksadesimal).

#### 1.4 Sistem Penomoran Biner (Basis 2)

Elektronik digital menggunakan sistem penomoran biner karena hanya menggunakan angka 0 dan 1, yang dapat direpresentasikan secara sederhana dalam sistem digital dengan dua level tegangan yang berbeda, seperti 5V = 1 dan 0V = 0. Faktor pembobot untuk posisi biner adalah pangkat 2 yang ditunjukkan pada Tabel 1.1.

Tabel 1. 1 Kekuatan Dari 2 Faktor Pembobotan Biner

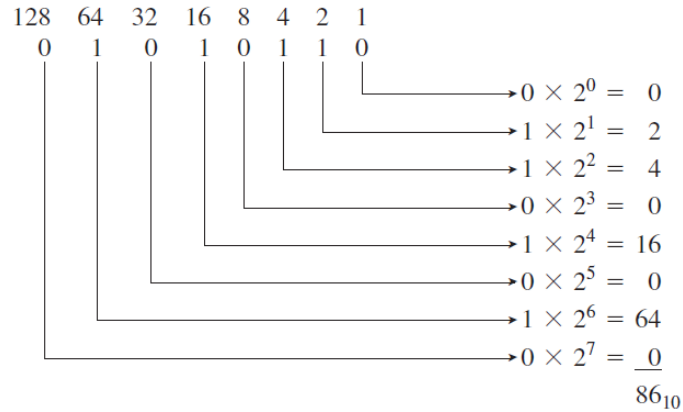
128	64	32	16	8	4	2	1	$2^0 =$	1
$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$	$2^1 =$	2
								$2^2 =$	4
								$2^3 =$	8
								$2^4 =$	16
								$2^5 =$	32
								$2^6 =$	64
								$2^7 =$	128

Contoh 1.2

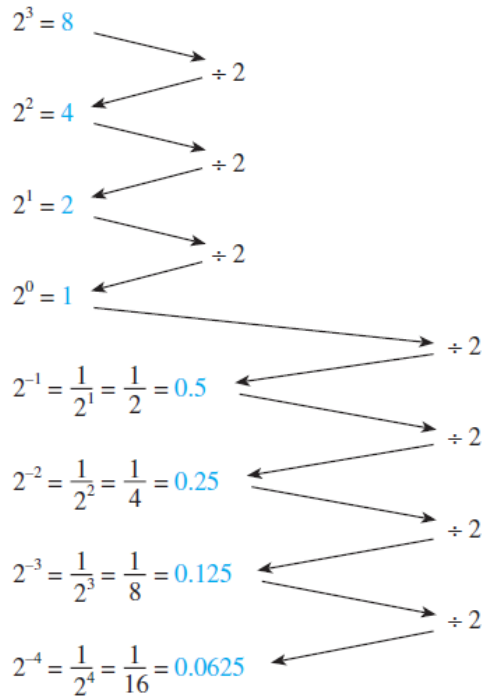
Ubah bilangan biner  $01010110_2$  menjadi desimal.

Solusi:

Kalikan setiap digit biner dengan faktor bobot yang sesuai dan jumlahkan hasilnya.



Meskipun jarang digunakan dalam sistem digital, pembobotan biner untuk nilai kurang dari 1 dimungkinkan (bilangan biner pecahan). Faktor-faktor ini dikembangkan dengan membagi faktor pembobot secara berturut-turut dengan 2 untuk setiap penurunan pangkat 2. Hal ini juga berguna untuk mengilustrasikan mengapa  $2^0$  sama dengan 1, bukan nol (lihat Gambar 1.6).



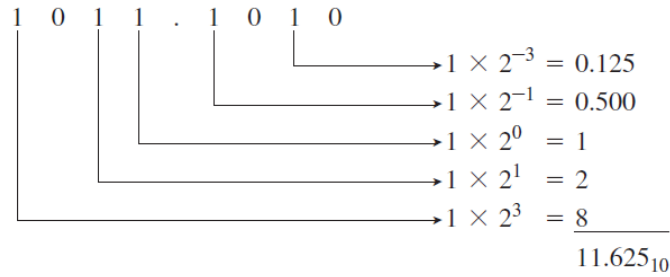
Gambar 1. 6 Pembagian Berurutan Dengan 2 Untuk Mengembangkan Faktor Pembobotan Biner Fraksional Dan Menunjukkan Bahwa  $2^0$  Sama Dengan 1

### Contoh 1.3

Ubah bilangan biner pecahan  $1011 \cdot 1010_2$  menjadi desimal.

Solusi:

Kalikan setiap digit biner dengan faktor pembobotan yang sesuai yang diberikan pada Gambar 1–6, dan jumlahkan hasilnya.



### 1.5 Konversi Desimal ke Biner

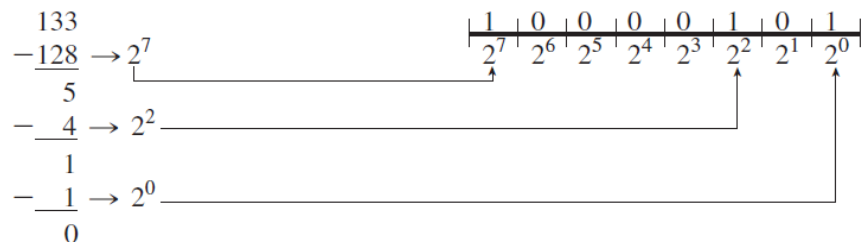
Konversi dari biner ke desimal biasanya dilakukan oleh komputer digital untuk kemudahan interpretasi oleh orang yang membaca angka tersebut. Sebaliknya, ketika seseorang memasukkan angka desimal ke dalam komputer digital, angka tersebut harus diubah menjadi biner sebelum dapat dioperasikan. Mari lihat konversi desimal ke biner.

Contoh 1.4

Ubah  $133_{10}$  menjadi biner.

Solusi:

Mengacu pada Tabel 1-1, dapat melihat bahwa pangkat terbesar dari 2 yang sesuai dengan 133 adalah  $2^7$  ( $2^7 = 128$ ), tetapi masih menyisakan nilai 5 ( $133 - 128 = 5$ ) untuk diperhitungkan. Lima dapat diselesaikan dengan  $2^2$  dan  $2^0$  ( $2^2 = 4$ ,  $2^0 = 1$ ). Jadi prosesnya terlihat seperti ini:



Jawaban:  $110000101_2$

Catatan: Pangkat 2 yang diperlukan untuk memberikan angka 133 ditentukan terlebih dahulu. Kemudian semua posisi lainnya diisi dengan nol.



### Contoh 1.6

Ubah 15210 menjadi biner menggunakan pembagian berurutan

Solusi:

$$152 \div 2 = 76 \text{ sisa } 0 \text{ (LSB)}$$

$$76 \div 2 = 38 \text{ sisa } 0$$

$$38 \div 2 = 19 \text{ sisa } 0$$

$$19 \div 2 = 9 \text{ sisa } 1$$

$$9 \div 2 = 4 \text{ sisa } 1$$

$$4 \div 2 = 2 \text{ sisa } 0$$

$$2 \div 2 = 1 \text{ sisa } 0$$

$$1 \div 2 = 0 \text{ sisa } 1 \text{ (MSB)}$$

Jawaban:  $10011000_2$

### 1.6 Sistem Penomoran Oktal (Basis 8)

Sistem penomoran oktal adalah metode pengelompokan bilangan biner ke dalam kelompok tiga. Delapan digit yang diperbolehkan adalah 0, 1, 2, 3, 4, 5, 6, dan 7. Sistem penomoran oktal digunakan oleh produsen komputer yang menggunakan kode 3-bit untuk menunjukkan instruksi atau operasi yang akan dilakukan. Dengan menggunakan representasi oktal, bukan biner, pengguna dapat menyederhanakan tugas memasukkan atau membaca instruksi komputer sehingga menghemat waktu. Pada Tabel 1.2, melihat bahwa ketika bilangan oktal melebihi 7, posisi oktal yang paling signifikan akan disetel ulang ke nol dan posisi paling signifikan berikutnya bertambah 1.

Tabel 1. 2 Sistem Penomoran Oktal

Decimal	Binary	Octal
0	000	0
1	001	1
2	010	2
3	011	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	10
9	1001	11
10	1010	12



## 1.7 Konversi Oktal

Mengubah dari biner ke oktal hanyalah masalah mengelompokkan posisi biner dalam kelompok tiga (mulai dari posisi paling signifikan) dan menuliskan ekuivalen oktal.

Contoh 1.7

Ubah  $011101_2$  menjadi oktal.

Solusi:

$$\underbrace{011}_3 \quad \underbrace{101}_5 = 35_8$$

Contoh 1.8

Konversikan  $10111001_2$  ke oktal.

Solusi:

$$\begin{array}{ccccccc} & & & & 1 & 0 & & 1 & 1 & 1 & & 0 & 0 & 1_2 \\ & & & & & \underbrace{\phantom{010}} & & \underbrace{\phantom{111}} & & \underbrace{\phantom{001}} & & & & & \\ \text{add a leading zero} \rightarrow & & & & 0 & 1 & 0 & & & & & & & & \\ & & & & \underbrace{\phantom{010}} & & & & & & & & & & \\ & & & & 2 & & & & & 7 & & & & & 1 & & = 271_8 \end{array}$$

Untuk mengonversi oktal ke biner, maka membalik prosesnya.

Contoh 1.9

Konversikan  $624_8$  ke biner

Solusi:

$$\underbrace{6}_{110} \quad \underbrace{2}_{010} \quad \underbrace{4}_{100} = 110010100_2$$

Contoh 1.10

Ubah  $326_8$  menjadi desimal

Solusi:

$$\begin{array}{l} \begin{array}{ccc} 3 & 2 & 6 \\ | & | & | \\ \hline & & \rightarrow 6 \times 8^0 = 6 \times 1 = 6 \\ & & \rightarrow 2 \times 8^1 = 2 \times 8 = 16 \\ & & \rightarrow 3 \times 8^2 = 3 \times 64 = 192 \\ & & \hline & & 214_{10} \end{array} \end{array}$$

Untuk mengkonversi dari desimal ke oktal, prosedur pembagian berurutan dapat digunakan.

Contoh 1.11

Ubah  $486_{10}$  menjadi oktal.

Solusi:

$$\left. \begin{array}{r} 486 \div 8 = 60 \text{ remainder } 6 \\ 60 \div 8 = 7 \text{ remainder } 4 \\ 7 \div 8 = 0 \text{ remainder } 7 \end{array} \right\} 746_8$$
$$486_{10} = 746_8 \text{ Answer}$$

Memeriksa:

$$\begin{array}{r} 7 \quad 4 \quad 6 \\ \left. \begin{array}{l} \longrightarrow 6 \times 8^0 = 6 \\ \longrightarrow 4 \times 8^1 = 32 \\ \longrightarrow 7 \times 8^2 = 448 \end{array} \right\} \\ \hline 486 \end{array}$$

## 1.8 Sistem Penomoran Heksadesimal (Basis 16)

Sistem penomoran heksadesimal, seperti sistem oktal, adalah metode pengelompokan bit untuk mempermudah memasukkan dan membaca instruksi atau data yang ada dalam sistem komputer digital. Heksadesimal menggunakan pengelompokan 4-bit; oleh karena itu, instruksi atau data yang digunakan dalam sistem komputer 8, 16, atau 32-bit dapat direpresentasikan sebagai kode heksadesimal dua, empat, atau delapan digit alih-alih menggunakan string panjang digit biner (lihat Tabel 1.3).

Hexadecimal (hex) menggunakan 16 digit berbeda dan merupakan metode pengelompokan bilangan biner dalam kelompok empat. Karena digit hex harus diwakili oleh satu karakter, huruf dipilih untuk mewakili nilai yang lebih besar dari 9. 16 digit hex yang diperbolehkan adalah 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, dan F.

Untuk menandakan nomor hex, digunakan subskrip 16 atau huruf H (yaitu,  $A7_{16}$  atau  $A7H$ ). Dua digit hex digunakan untuk mewakili 8 bit (juga dikenal sebagai byte). Empat bit (satu digit hex) terkadang disebut nibble.

Tabel 1. 3 Sistem Penomoran Heksadesimal

Decimal	Binary	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F
16	0001 0000	1 0
17	0001 0001	1 1
18	0001 0010	1 2
19	0001 0011	1 3
20	0001 0100	1 4

### 1.9 Konversi Heksadesimal

Untuk mengonversi dari biner ke heksadesimal, kelompokkan bilangan biner dalam kelompok empat dan tuliskan digit heksadesimal yang setara.

Contoh 1.12

Ubah  $01101101_2$  menjadi hex.

Solusi:

$$\underbrace{0110}_6 \quad \underbrace{1101}_D = 6D_{16}$$

Untuk mengonversi heksadesimal ke biner, gunakan proses kebalikannya

Contoh 1.13

Mengkonversi  $A9_{16}$  ke biner

Solusi:

$$\underbrace{A}_{1010} \quad \underbrace{9}_{1001} = 10101001_2$$

Contoh 1.14

Ubah  $2A6_{16}$  menjadi desimal.

Solusi:

$$\begin{array}{r} 2 \quad A \quad 6 \\ \left. \begin{array}{l} \longrightarrow \\ \longrightarrow \\ \longrightarrow \end{array} \right\} \begin{array}{l} 6 \times 16^0 = 6 \times 1 = 6 \\ A \times 16^1 = 10 \times 16 = 160 \\ 2 \times 16^2 = 2 \times 256 = \underline{512} \\ \hline 678_{10} \end{array} \end{array}$$

Contoh 1.15

Ulangi Contoh 1.14 dengan mengonversi terlebih dahulu ke biner, lalu ke desimal

Solusi:

$$\overbrace{0010}^2 \quad \overbrace{1010}^A \quad \overbrace{0110}^6 = 2 + 4 + 32 + 128 + 512 = 678_{10}$$

Untuk mengonversi dari desimal ke heksadesimal, gunakan pembagian berurutan.

Contoh 1.16

Ubah  $151_{10}$  menjadi hex

Solusi:

$$151 \div 16 = 9 \text{ sisa } 7 \text{ (LSD)}$$

$$9 \div 16 = 0 \text{ sisa } 9 \text{ (MSD)}$$

$$151_{10} = 97_{16}$$

Memeriksa:

$$\begin{array}{r} 97_{16} \\ \left. \begin{array}{l} \longrightarrow \\ \longrightarrow \end{array} \right\} \begin{array}{l} 7 \times 16^0 = 7 \\ 9 \times 16^1 = \underline{144} \\ \hline 151 \end{array} \end{array}$$

Contoh 1.17

Ubah 49810 menjadi hex.

Solusi:

$$498 \div 16 = 31 \text{ sisa } 2 \text{ (LSD)}$$

$$31 \div 16 = 1 \text{ sisa } 15 \text{ (= F)}$$

$$1 \div 16 = 0 \text{ sisa } 1 \text{ (MSD)}$$

$$498_{10} = 1F2_{16}$$

Memeriksa:

$$\begin{array}{r}
 1 \text{ F } 2_{16} \quad 2 \times 16^0 = 2 \times 1 = 2 \\
 \quad \quad \quad \text{F} \times 16^1 = 15 \times 16 = 240 \\
 \quad \quad \quad 1 \times 16^2 = 1 \times 256 = \underline{256} \\
 \quad \quad \quad \quad \quad \quad \quad 498
 \end{array}$$

### 1.10 Sistem Binary Coded Decimal

Sistem *Binary Coded Decimal (BCD)* digunakan untuk mewakili masing-masing dari 10 angka desimal sebagai kode biner 4-bit. Kode ini berguna untuk mengeluarkan ke tampilan yang selalu numerik (0 hingga 9), seperti yang ditemukan di jam digital atau voltmeter digital. Untuk membentuk angka *BCD*, cukup ubah setiap angka desimal menjadi kode biner 4-bit.

Contoh 1.18

Ubah 4 9 610 menjadi *BCD*

Solusi:

$$\begin{array}{ccc}
 \overbrace{0100}^4 & \overbrace{1001}^9 & \overbrace{0110}^6 \\
 0100 & 1001 & 0110_{\text{BCD}}
 \end{array} = 0100 \ 1001 \ 0110_{\text{BCD}}$$

Untuk mengubah *BCD* menjadi desimal, cukup balik prosesnya

Contoh 1.19

Ubah 0111 0101 1000 *BCD* menjadi desimal

Solusi:

$$\begin{array}{ccc}
 \overbrace{0111}^7 & \overbrace{0101}^5 & \overbrace{1000}^8 \\
 7 & 5 & 8
 \end{array} = 758_{10}$$

Contoh 1.20

Ubah 0110 0100 1011 *BCD* menjadi desimal

Solusi:

$$\begin{array}{ccc}
 0110 & 0100 & 1011 \\
 6 & 4 & *
 \end{array}$$

Konversi ini tidak mungkin karena 1011 bukan desimal berkode biner yang valid. Itu tidak dalam kisaran 0 hingga 9.

### 1.11 Perbandingan Sistem Penomoran

Tabel 1.4 membandingkan bilangan yang ditulis dalam lima sistem bilangan yang biasa digunakan dalam elektronik digital dan sistem komputer.

Tabel 1. 4 Perbandingan Sistem Penomoran

Decimal	Binary	Octal	Hexadecimal	BCD
0	0000	0	0	0000
1	0001	1	1	0001
2	0010	2	2	0010
3	0011	3	3	0011
4	0100	4	4	0100
5	0101	5	5	0101
6	0110	6	6	0110
7	0111	7	7	0111
8	1000	1 0	8	1000
9	1001	1 1	9	1001
10	1010	1 2	A	0001 0000
11	1011	1 3	B	0001 0001
12	1100	1 4	C	0001 0010
13	1101	1 5	D	0001 0011
14	1110	1 6	E	0001 0100
15	1111	1 7	F	0001 0101
16	0001 0000	2 0	1 0	0001 0110
17	0001 0001	2 1	1 1	0001 0111
18	0001 0010	2 2	1 2	0001 1000
19	0001 0011	2 3	1 3	0001 1001
20	0001 0100	2 4	1 4	0010 0000

### 1.12 Kode ASCII

Untuk memasukkan dan mengeluarkan informasi dari komputer, memerlukan lebih dari sekadar representasi numerik; kami juga harus menjaga semua huruf dan simbol yang digunakan dalam pemrosesan sehari-hari. Informasi seperti nama, alamat, dan deskripsi barang harus dimasukkan dan dikeluarkan dalam format yang dapat dibaca. Tapi ingat bahwa sistem digital hanya dapat menangani 1 dan 0. Oleh karena itu, diperlukan kode khusus untuk merepresentasikan semua data alfanumerik (huruf, simbol, dan angka). Sebagian besar industri telah menetapkan kode *input/output (I/O)* yang disebut *American Standard Code for Information Interchange (ASCII)*. Kode *ASCII* menggunakan 7 bit untuk mewakili semua data alfanumerik yang digunakan dalam *I/O* komputer. Tujuh bit akan menghasilkan 128 kombinasi kode yang berbeda, seperti tercantum dalam Tabel 1.5.

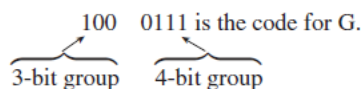
Tabel 1. 5 Kode Standar Amerika untuk Pertukaran Informasi

MSB LSB								
	000	001	010	011	100	101	110	111
0000	NUL	DLE	SP	0	@	P	'	p
0001	SOH	DC <sub>1</sub>	!	1	A	Q	a	q
0010	STX	DC <sub>2</sub>	"	2	B	R	b	r
0011	ETX	DC <sub>3</sub>	#	3	C	S	c	s
0100	EOT	DC <sub>4</sub>	\$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	'	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	\	l	
1101	CR	GS	-	=	M	]	m	}
1110	SO	RS	.	>	N	↑	n	~
1111	SI	US	/	?	O	—	o	DEL

Definitions of control abbreviations:		FS	Form separator
ACK	Acknowledge	GS	Group separator
BEL	Bell	HT	Horizontal tab
BS	Backspace	LF	Line feed
CAN	Cancel	NAK	Negative acknowledge
CR	Carriage return	NUL	Null
DC <sub>1</sub> –DC <sub>4</sub>	Direct control	RS	Record separator
DEL	Delete idle	SI	Shift in
DLE	Data link escape	SO	Shift out
EM	End of medium	SOH	Start of heading
ENQ	Enquiry	SP	Space
EOT	End of transmission	STX	Start text
ESC	Escape	SUB	Substitute
ETB	End of transmission block	SYN	Synchronous idle
ETX	End text	US	Unit separator
FF	Form feed	VT	Vertical tab

Setiap kali tombol ditekan pada keyboard *ASCII*, tombol tersebut diubah menjadi kode *ASCII* dan diproses oleh komputer. Kemudian, sebelum mengeluarkan konten komputer ke terminal tampilan atau printer, semua informasi diubah dari *ASCII* menjadi bahasa Inggris standar. Untuk menggunakan tabel, tempatkan grup 4-bit di posisi paling signifikan dan grup 3-bit di posisi paling signifikan.

Contoh 1.21



Contoh 1.22

Menggunakan Tabel 1–5, tentukan kode ASCII untuk huruf kecil *p*.

Solusi:

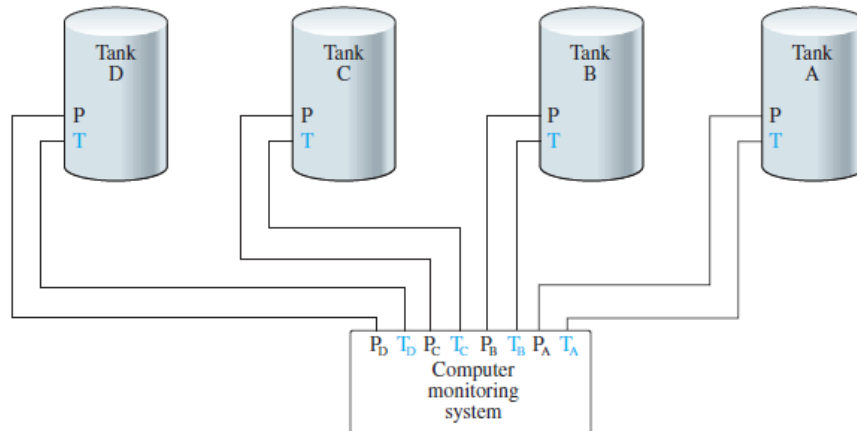
1110000 (Catatan: Seringkali, nol di depan ditambahkan untuk membentuk hasil 8-bit, sehingga  $p = 01110000$ )

### 1.13 Aplikasi Sistem Penomoran

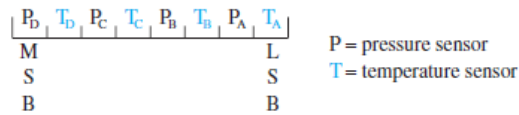
Karena sistem digital bekerja terutama dengan 1 dan 0, kami telah menghabiskan banyak waktu bekerja dengan berbagai sistem bilangan. Sistem mana yang digunakan tergantung pada bagaimana data dikembangkan dan bagaimana mereka akan digunakan. Di bagian ini, kami bekerja dengan beberapa aplikasi yang bergantung pada terjemahan dan interpretasi representasi digital ini.

#### Aplikasi 1.1

Fasilitas pembangkit listrik panas bumi menggunakan komputer untuk memantau suhu dan tekanan empat tangki penyimpanan cairan, seperti yang ditunjukkan pada Gambar 1–7(a). Setiap kali suhu atau tekanan melebihi batas bahaya, sensor tangki internal menerapkan 1 ke output yang sesuai ke komputer. Jika semua kondisi OK, maka semua output adalah 0.



(a)



(b)

Gambar 1. 7 (a) Sambungan Sirkuit Untuk Monitor Suhu Dan Tekanan Di Fasilitas Pembangkit Listrik Panas Bumi; (b) Tata Letak Data Biner Yang Dibaca Oleh Sistem Pemantauan Komputer



(a) Jika komputer membaca string biner 0010 1000, masalah apa yang muncul?

Solusi:

Memasukkan string biner tersebut ke dalam bagan Gambar 1–7(b) menunjukkan bahwa tekanan dalam tangki C dan B sangat tinggi.

(b) Masalah apa yang muncul jika komputer membaca 55H (55 hex)?

Solusi:

55H 0101 0101, artinya semua temperatur terlalu tinggi

(c) Berapa angka heksadesimal yang dibaca komputer jika suhu dan tekanan di kedua tangki D dan B tinggi?

Solusi:

CCH (1100 1100 CCH)

(d) Tangki A dan B tidak digunakan lagi, dan keluaran sensornya dihubungkan ke 1s. Seorang pemrogram komputer harus menulis sebuah program untuk mengabaikan kondisi sirkuit baru ini. Program komputer harus memeriksa bahwa nilai yang dibaca selalu kurang dari berapa desimal yang setara ketika tidak ada masalah?

Solusi:

$< 31_{10}$ , karena, dengan 4 bit orde rendah *HIGH*, jika  $T_C$  berjalan *HIGH*, maka string binernya adalah 00011111, yang sama dengan  $31_{10}$ .

(e) Di area lain dari instalasi, hanya tiga tangki (A, B, dan C) yang harus dipantau. Berapa angka oktal yang terbaca jika tangki B memiliki suhu dan tekanan yang tinggi?

Solusi:

$$14_8(001100_2 = 14_8)$$

## BAB II

### GERBANG LOGIKA DASAR

#### 2.1 Gerbang AND

Mari mulai dengan melihat gerbang AND dua masukan yang simbol skematiknya ditunjukkan pada Gambar 2.1. Pengoperasian gerbang AND sederhana dan didefinisikan sebagai berikut: Output, X, *HIGH* jika input A DAN input B keduanya *HIGH*. Dengan kata lain, jika AND  $B = 1$ , maka  $X = 1$ . Jika A atau B atau keduanya *LOW*, keluarannya akan *LOW*.



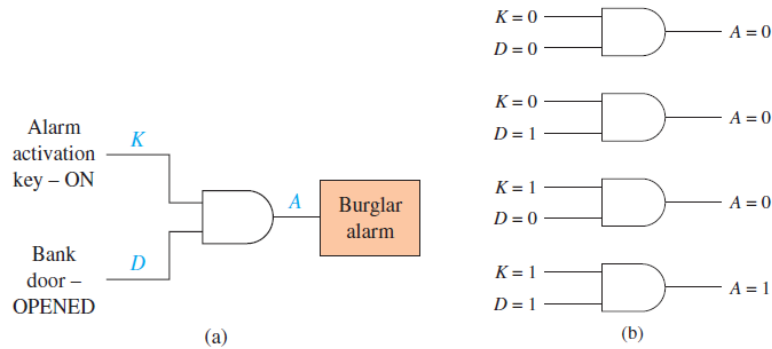
Gambar 2. 1 Simbol Gerbang AND Dua Masukan

Cara terbaik untuk mengilustrasikan bagaimana level output dari sebuah gerbang merespon semua kemungkinan kombinasi level input adalah dengan tabel kebenaran. Tabel 2.1 adalah tabel kebenaran untuk gerbang AND dua masukan. Di sisi kiri tabel kebenaran, semua kemungkinan kombinasi tingkat input dicantumkan, dan di sisi kanan, output yang dihasilkan dicantumkan.

Tabel 2. 1 Tabel Kebenaran Gerbang AND Dua Masukan

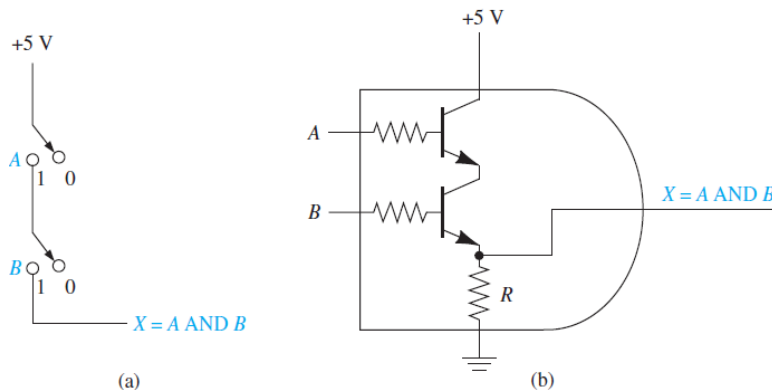
Inputs		Output
A	B	$X = AB$
0	0	0
0	1	0
1	0	0
1	1	1

Dari tabel kebenaran, dapat melihat bahwa keluaran pada X adalah *HIGH* hanya jika A AND B keduanya *HIGH*. Jika gerbang AND ini adalah sirkuit terpadu TTL, *HIGH* berarti dan *LOW* berarti 0 V (yaitu, 1 didefinisikan sebagai 5V dan 0 didefinisikan sebagai 0V). Salah satu contoh bagaimana gerbang AND dapat digunakan adalah dalam sistem alarm pencuri bank. Output gerbang AND akan *HIGH* untuk menyalakan alarm jika kunci aktivasi alarm dalam posisi *ON* AND pintu depan dibuka. Pengaturan ini diilustrasikan pada Gambar 2.2(a). Gambar 2.2(b) menunjukkan hasil untuk setiap kombinasi Kunci (K) dan Pintu (D).



Gambar 2. 2 Gerbang AND: (a) Digunakan untuk mengaktifkan alarm pencuri; (b) semua kombinasi kunci ON (K) dan pintu OPEN (D)

Cara lain untuk mengilustrasikan operasi gerbang AND adalah dengan menggunakan rangkaian listrik seri. Pada Gambar 2.3, dengan menggunakan sakelar manual dan transistor, output pada X adalah *HIGH* jika kedua sakelar A AND B *HIGH* (1). Gambar 2.3 juga menunjukkan apa yang dikenal sebagai persamaan Boolean untuk fungsi AND,  $X = A \text{ dan } B$ , yang dapat dianggap sebagai X sama dengan 1 jika A AND B keduanya sama dengan 1.

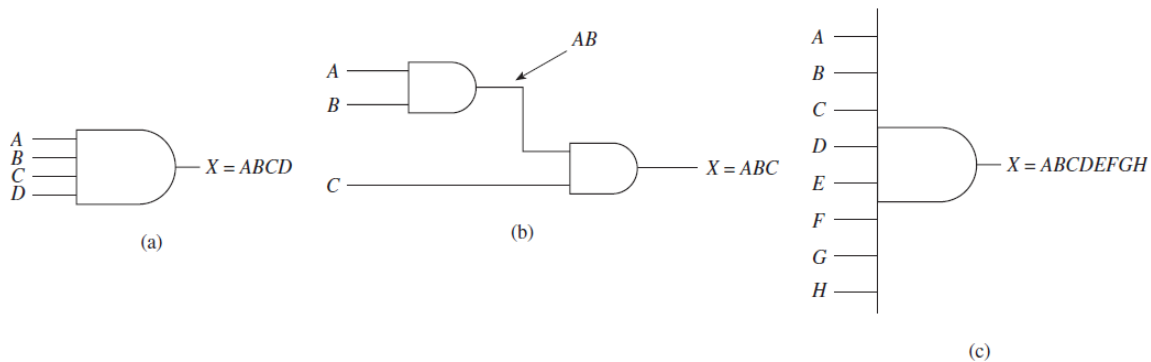


Gambar 2. 3 Analogi Kelistrikan Untuk Gerbang AND: (a) Menggunakan Sakelar Manual; (b) Menggunakan Sakelar Transistor

Persamaan Boolean untuk fungsi AND dapat lebih sederhana ditulis sebagai  $X = A \cdot B$  atau hanya  $X = AB$  (yang dibaca sebagai “X sama dengan A AND B”). Persamaan Boolean akan digunakan sepanjang sisa buku ini untuk menggambarkan operasi gerbang logika atau kombinasi gerbang logika secara aljabar.

Gerbang AND dapat memiliki lebih dari dua input. Gambar 2.4 menunjukkan gerbang AND empat masukan, tiga masukan, dan delapan masukan. Tabel kebenaran gerbang AND dengan empat masukan ditunjukkan pada Tabel 2.2. Untuk menentukan jumlah total kombinasi yang berbeda untuk dicantumkan dalam tabel kebenaran, gunakan persamaan 2.1.

$$\text{Jumlah Kombinasi} = 2^N, \text{ dimana } N = \text{Jumlah Input} \quad (2.1)$$



Gambar 2. 4 Simbol Gerbang AND Multi-Input: (a) 4-Input; (b) 3-Input Dibentuk Dengan Dua Gerbang 2-Input; (c) 8-Input

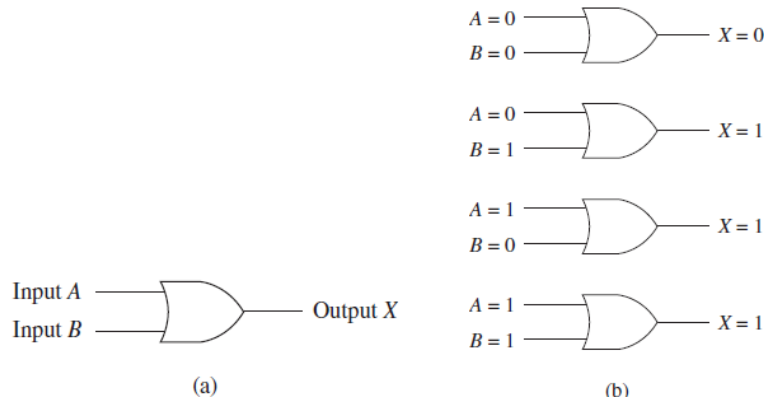
Oleh karena itu, dalam kasus gerbang AND empat masukan, banyaknya kemungkinan kombinasi masukan adalah  $2^4 = 16$ . Saat membuat tabel kebenaran, pastikan untuk mencantumkan semua 16 kombinasi level masukan yang berbeda. Salah satu cara mudah untuk memastikan bahwa tidak mengabaikan kombinasi variabel ini atau menggandakan kombinasi secara tidak sengaja adalah dengan mencantumkan input dalam urutan penghitung biner (0000, 0001, 0010, . . . , 1111). Perhatikan juga pada Tabel 2.2 bahwa kolom A mencantumkan delapan 0, lalu delapan 1; kolom B mencantumkan empat 0, empat 1, empat 0, empat 1; kolom C mencantumkan dua 0, dua 1, dua 0, dua 1, dan seterusnya; dan kolom D mencantumkan satu 0, satu 1, satu 0, satu 1, dan seterusnya.

Tabel 2. 2 Tabel Kebenaran Gerbang AND Empat Masukan

A	B	C	D	X
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

## 2.2 Gerbang OR

Gerbang OR juga memiliki dua atau lebih input dan satu output. Simbol untuk gerbang OR dua masukan ditunjukkan pada Gambar 2.5. Pengoperasian gerbang OR dua-input didefinisikan sebagai berikut: Keluaran pada X akan HIGH setiap kali input A OR input B *HIGH* atau keduanya *HIGH*. Sebagai persamaan Boolean, ini dapat ditulis  $X = A + B$  (yang dibaca sebagai “X sama dengan A OR B”). Perhatikan penggunaan simbol + untuk mewakili fungsi OR.



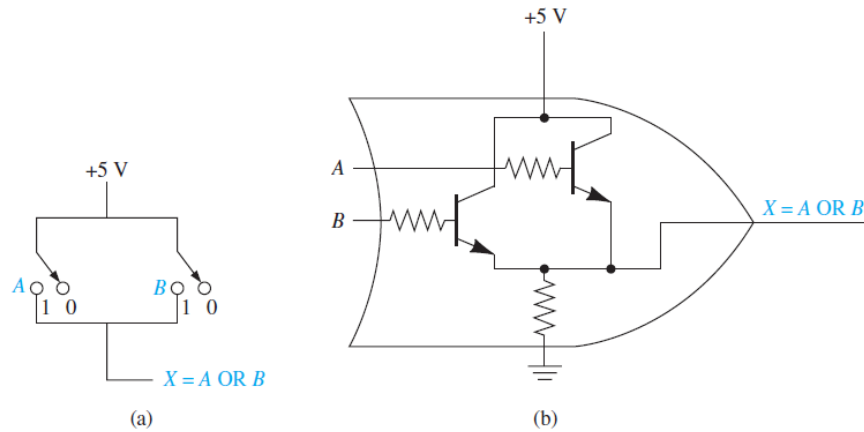
Gambar 2. 5 Gerbang OR Dua Masukan: (a) Simbol; (b) Semua Kombinasi Masukan

Tabel kebenaran untuk gerbang OR dua masukan ditunjukkan pada Tabel 2.3.

Tabel 2. 3 Tabel Kebenaran Gerbang OR Dua Masukan

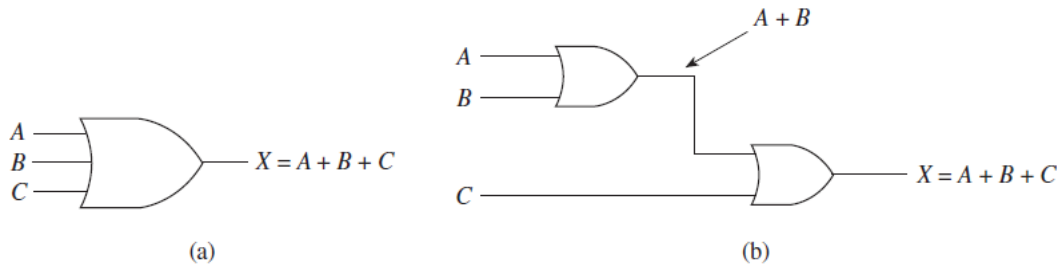
Inputs		Output
A	B	$X = A + B$
0	0	0
0	1	1
1	0	1
1	1	1

Dari tabel kebenaran, dapat melihat bahwa X adalah 1 jika A OR B adalah 1 atau jika A dan B sama-sama 1. Dengan menggunakan sakelar manual atau transistor dalam rangkaian listrik, seperti yang ditunjukkan pada Gambar 2.6, dapat mengamati analogi kelistrikan ke gerbang OR. Dari gambar tersebut, melihat bahwa keluaran di X akan menjadi 1 jika A atau B, atau keduanya, *HIGH* (1).

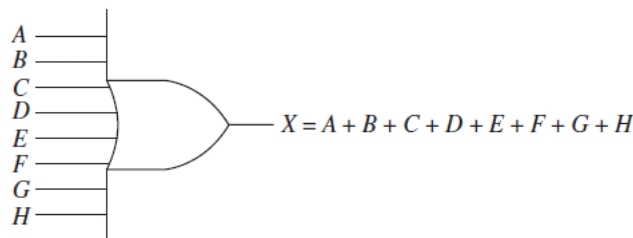


Gambar 2. 6 Analogi Listrik Untuk Gerbang OR: (a) Menggunakan Sakelar Manual; (b) Menggunakan Sakelar Transistor

Gerbang OR juga dapat memiliki lebih dari dua input. Gambar 2.7 menunjukkan gerbang OR tiga masukan dan Gambar 2.8 menunjukkan gerbang OR delapan masukan. Tabel kebenaran untuk gerbang OR tiga masukan akan memiliki delapan masukan ( $2^3 = 8$ ), dan gerbang OR delapan masukan akan memiliki 256 masukan ( $2^8 = 256$ ).



Gambar 2. 7 Gerbang OR Tiga Masukan: (a) Simbol; (b) Tiga Input Dibentuk Dengan Dua Gerbang 2-Input



Gambar 2. 8 Simbol Gerbang OR Delapan Masukan

Maka buat tabel kebenaran untuk gerbang OR tiga masukan. Tabel kebenaran dari Tabel 2.4 dibangun dengan terlebih dahulu menggunakan Persamaan 2.1 untuk menentukan bahwa akan ada delapan entri, kemudian mendaftarkan delapan kombinasi input dalam urutan pencacah biner (000 hingga 111), dan kemudian mengisi kolom output (X) dengan menyadari bahwa X akan selalu

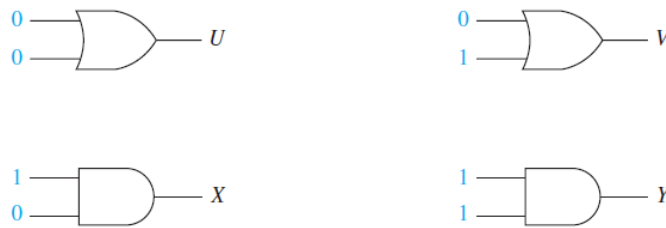
*HIGH* selama setidaknya salah satu inputnya *HIGH*. Ketika melihat tabel kebenaran yang lengkap, dapat melihat bahwa satu-satunya waktu keluaran *LOW* adalah ketika semua masukan *LOW*.

Tabel 2. 4 Tabel Kebenaran Gerbang OR Tiga Masukan

<i>A</i>	<i>B</i>	<i>C</i>	<i>X</i>
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Contoh 2.1

Tentukan keluaran pada U, V, W, X, Y, dan Z pada Gambar 2.9



Gambar 2. 9 Operasi Gerbang Dasar AND dan OR

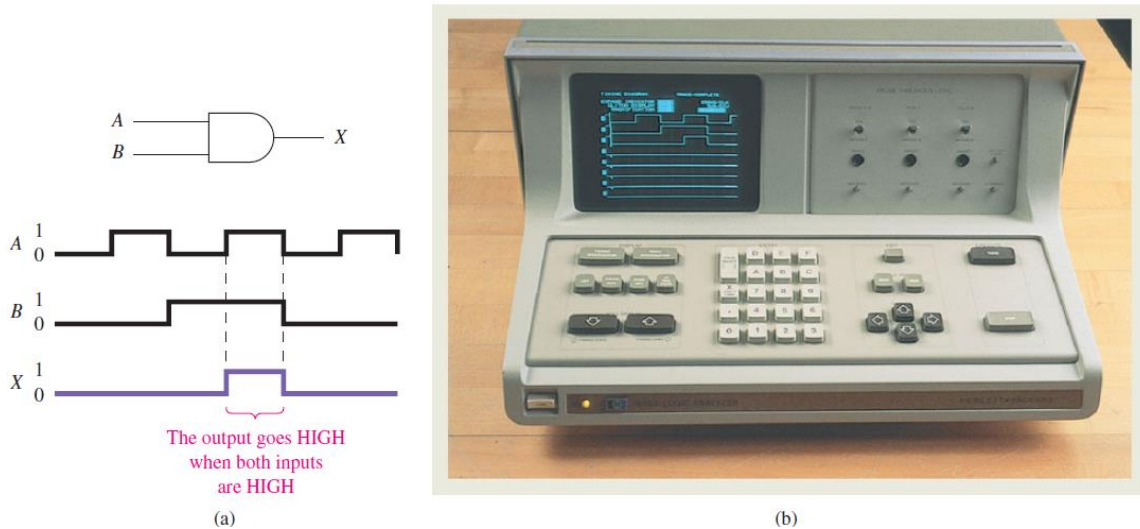
Solusi:

$$\begin{aligned}
 U &= 0 && (0 \text{ OR } 0 = 0) \\
 V &= 1 && (0 \text{ OR } 1 = 1) \\
 W &= 1 && (0 \text{ OR } 1 \text{ OR } 0 = 1) \\
 X &= 0 && (1 \text{ AND } 0 = 0) \\
 Y &= 1 && (1 \text{ AND } 1 = 1) \\
 Z &= 0 && (0 \text{ AND } 1 \text{ AND } 0 = 0)
 \end{aligned}$$

**2.3 Analisis Timing**

Cara lain yang berguna untuk menganalisis respons keluaran gerbang terhadap berbagai perubahan tingkat masukan adalah dengan menggunakan diagram waktu. Sebuah diagram waktu digunakan untuk mengilustrasikan secara grafis bagaimana tingkat output berubah sebagai respons terhadap perubahan tingkat input. Diagram waktu pada Gambar 2.10 menunjukkan dua bentuk gelombang input (A dan B) yang diterapkan pada gerbang AND dua input dan output X yang dihasilkan dari operasi AND. (Untuk *TTL* dan sebagian besar gerbang logika *CMOS*, 1 = 5V dan

0 = 0V. Seperti yang dilihat, analisis waktu sangat berguna untuk mengilustrasikan secara visual level pada output untuk berbagai perubahan level input.

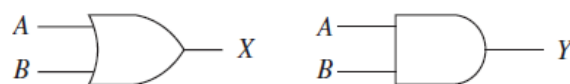


Gambar 2. 10 Analisis Waktu Gerbang AND: (a) Sketsa Bentuk Gelombang; (b) Tampilan Penganalisa Logika Aktual

Bentuk gelombang pengaturan waktu dapat diamati pada osiloskop atau penganalisa logika. Osiloskop *Dual Trace* dapat menampilkan dua bentuk gelombang tegangan versus waktu pada sumbu x yang sama. Itu ideal untuk membandingkan hubungan satu bentuk gelombang relatif dengan yang lain. Alat analisis waktu lainnya adalah penganalisa logika. Antara lain, ia dapat menampilkan hingga 16 bentuk gelombang tegangan versus waktu pada sumbu x yang sama (lihat Gambar 2.10[b]). Itu juga dapat menampilkan level dari beberapa sinyal digital dalam tabel keadaan, yang mencantumkan level biner dari semua bentuk gelombang, pada interval yang telah ditentukan, dalam biner, heksadesimal, atau oktal. Analisis waktu dari 8 atau 16 saluran secara bersamaan sangat penting saat menganalisis sistem digital dan mikroprosesor canggih di mana keterkaitan beberapa sinyal digital sangat penting untuk operasi sirkuit yang tepat.

Contoh 2.2

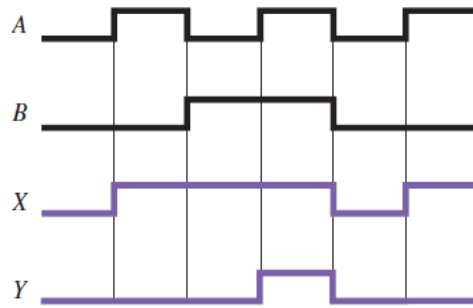
Buat sketsa bentuk gelombang keluaran pada X dan Y untuk gerbang OR dan gerbang AND dua masukan yang ditunjukkan pada Gambar 2.11, dengan bentuk gelombang masukan A dan B yang diberikan pada Gambar 2.12.



Gambar 2. 11 Gerbang Untuk Contoh 2.2



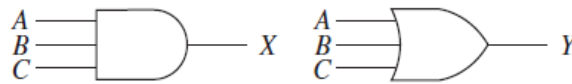
Solusi:



Gambar 2. 12 Solusi untuk Contoh 2.2

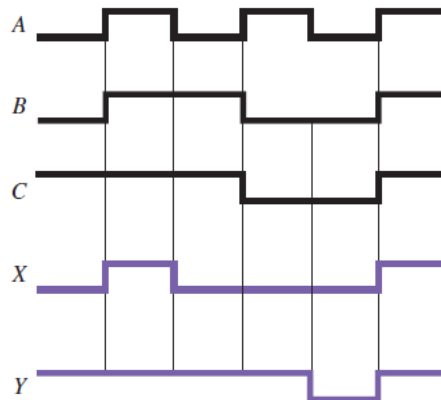
Contoh 2.3

Buat sketsa bentuk gelombang keluaran pada X untuk gerbang AND dan gerbang OR tiga masukan yang ditunjukkan pada Gambar 2.13, dengan bentuk gelombang masukan A, B, dan C yang diberikan pada Gambar 2.14.



Gambar 2. 13 Gerbang untuk Contoh 2.3

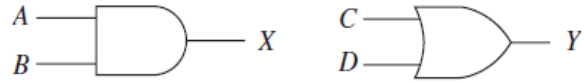
Solusi:



Gambar 2. 14 Solusi Untuk Contoh 2.3

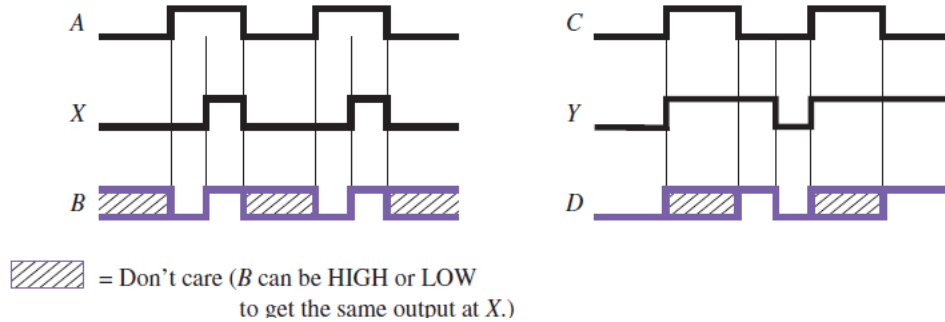
Contoh 2.4

Bentuk gelombang masukan pada A dan bentuk gelombang keluaran pada X diberikan untuk gerbang AND pada Gambar 2.15. Buat sketsa bentuk gelombang input yang diperlukan di B untuk menghasilkan output di X pada Gambar 2.16. Ulangi untuk gerbang OR.



Gambar 2. 15 Gerbang Untuk Contoh 2.4

Solusi:

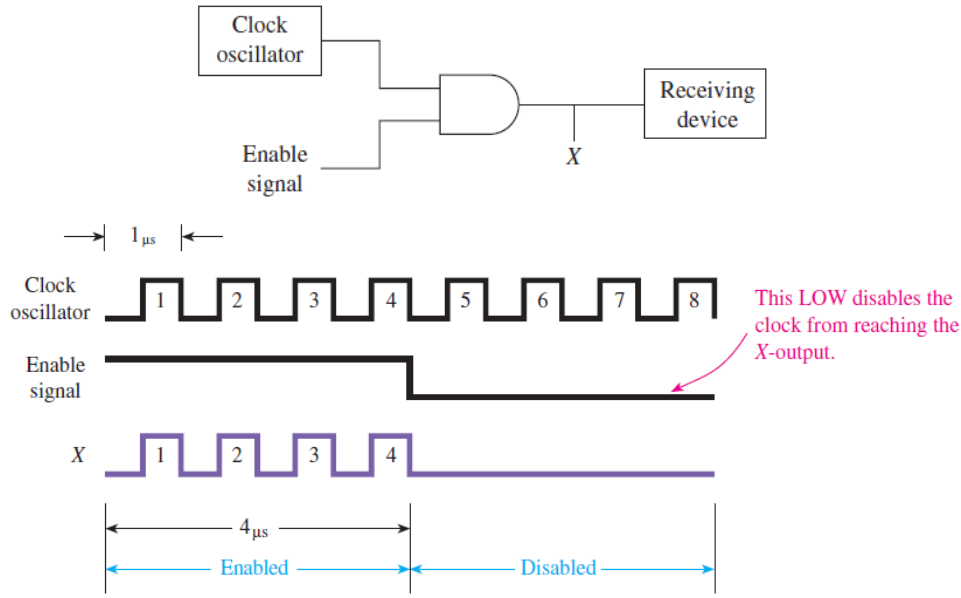


Gambar 2. 16 Solusi Untuk Contoh 2.4

## 2.4 Fungsi Enable dan Disable

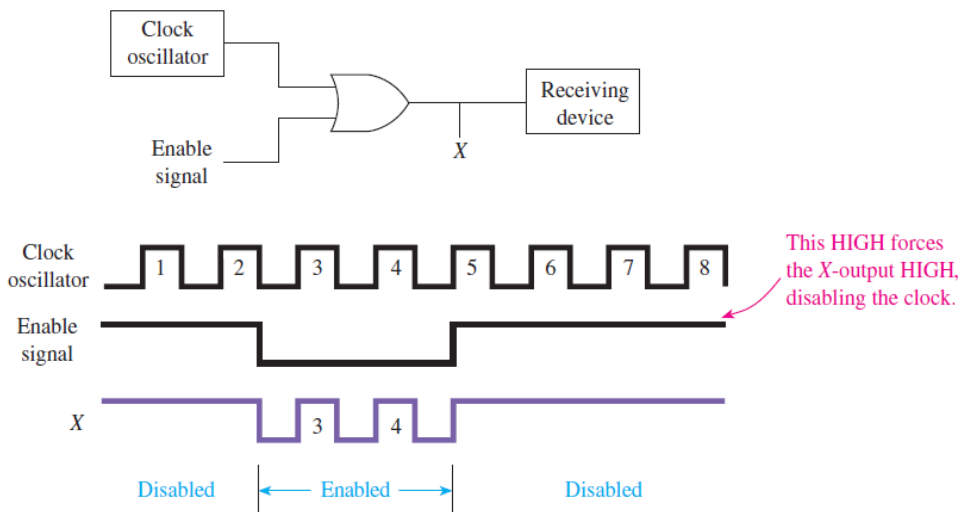
Gerbang AND dan OR dapat digunakan untuk mengaktifkan atau menonaktifkan bentuk gelombang yang ditransmisikan dari satu titik ke titik lainnya. Misalnya, ingin osilator jam 1 MHz mengirimkan hanya empat pulsa ke beberapa perangkat penerima. Saat ingin mengaktifkan empat pulsa jam untuk dikirim dan kemudian menonaktifkan transmisi sejak saat itu.

Frekuensi clock 1 MHz dikonversi menjadi 1 ms (1/1 MHz) untuk setiap periode clock. Oleh karena itu, untuk mentransmisikan empat pulsa clock, harus menyediakan sinyal aktif selama 4 ms. Gambar 2.17 menunjukkan rangkaian dan bentuk gelombang untuk mengaktifkan empat pulsa clock. Agar clock pulsa *HIGH* melewati gerbang AND ke titik X, input kedua ke gerbang AND (aktifkan input sinyal) harus *HIGH*; jika tidak, keluaran gerbang AND akan menjadi LOW. Oleh karena itu, ketika sinyal aktifkan *HIGH* selama 4 ms, empat pulsa clock melewati gerbang AND. Ketika sinyal aktif menjadi *LOW*, gerbang AND menonaktifkan pulsa *clock* lebih lanjut untuk mencapai perangkat penerima.



Gambar 2. 17 Menggunakan Gerbang AND Untuk Mengaktifkan/Menonaktifkan Clock Oscillator

Gerbang OR juga dapat digunakan untuk menonaktifkan suatu fungsi. Perbedaannya adalah input sinyal pengaktifan dibuat *HIGH* untuk dinonaktifkan, dan output gerbang OR menjadi *HIGH* saat dinonaktifkan, seperti yang ditunjukkan pada Gambar 2.18.

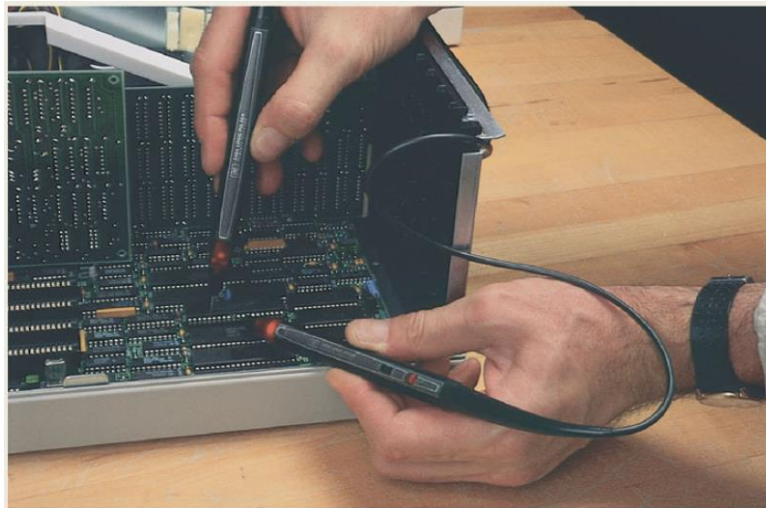


Gambar 2. 18 Menggunakan Gerbang OR Untuk Mengaktifkan/Menonaktifkan Clock Oscillator

## 2.5 Teknik Troubleshooting

Seperti perangkat elektronik lainnya, IC dan sirkuit elektronik digital bisa rusak. Pemecahan masalah adalah istilah yang diberikan untuk prosedur yang digunakan untuk menemukan kesalahan, atau masalah, di sirkuit. Untuk menjadi pemecah masalah yang baik, harus

terlebih dahulu memahami teori dan pengoperasian rangkaian, perangkat, dan *IC* yang diduga buruk. Jika memahami bagaimana *IC* tertentu seharusnya beroperasi, itu adalah tugas sederhana untuk menguji *IC* atau menjalankan fungsinya untuk melihat apakah itu beroperasi seperti yang diharapkan.



Gambar 2. 19 Pulser Logika Dan Probe Logika

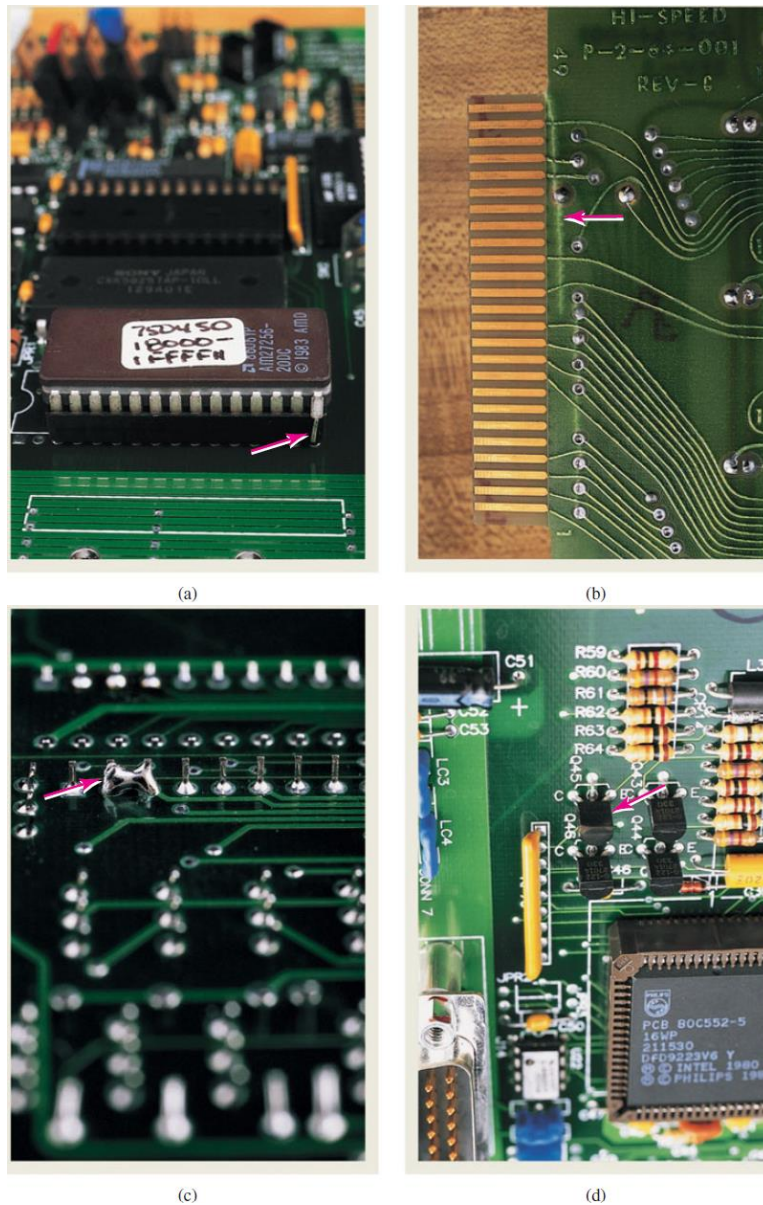
Ada dua alat sederhana yang akan memulai untuk menguji *IC* dan rangkaian digital. Mereka adalah logika pulser dan probe logika (lihat Gambar 2.19). Probe logika memiliki ujung logam yang ditempatkan pada pin *IC*, jejak papan sirkuit tercetak, atau ujung perangkat yang ingin diuji. Ini juga memiliki lampu indikator yang menyala, memberi tahu level digital pada saat itu. Jika level *HIGH* (1), lampu menyala terang. Jika levelnya *LOW* (0), lampu padam. Jika level mengambang (rangkaiannya terbuka, bukan *HIGH* atau *LOW*), lampu akan menyala redup. Tabel 2.5 merangkum status probe logika.

Tabel 2. 5 Status Probe Logika

Logic Level	Indicator Lamp
HIGH (1)	On
LOW (0)	Off
Float	Dim

Pulser logika digunakan untuk memberikan pulsa digital ke sirkuit yang sedang diuji. Dengan menerapkan pulsa ke sirkuit dan secara bersamaan mengamati probe logika, dapat mengetahui apakah sinyal pulsa melewati *IC* atau perangkat seperti yang diharapkan. Ketika menjadi semakin berpengalaman dalam pemecahan masalah, akan menemukan bahwa sebagian

besar kesalahan IC dan perangkat disebabkan oleh terminal input atau output yang terbuka atau pendek.



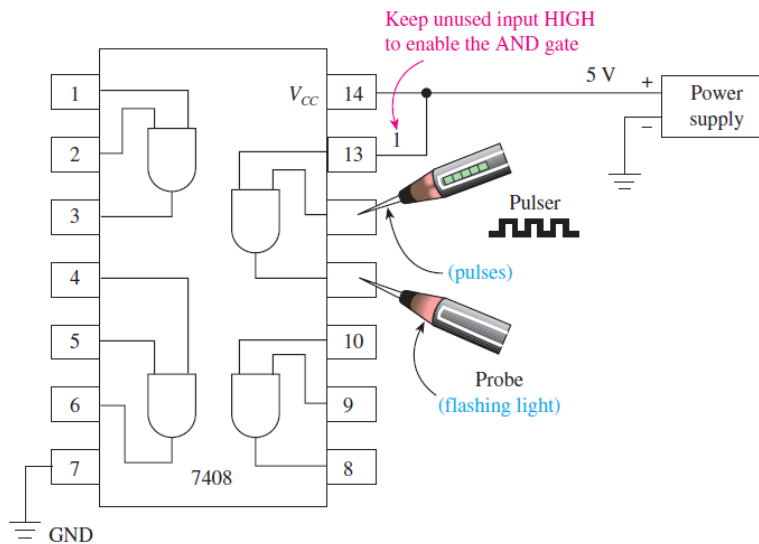
Gambar 2. 20 Empat Kesalahan Sirkuit Tercetak Yang Umum: (a) Misalignment Pin 14; (b) Papan Retak; (c) Jembatan Solder; (d) Transistor Yang Terbakar

Gambar 2.20 menunjukkan empat masalah umum yang akan temukan pada papan sirkuit tercetak yang akan menyebabkan terbuka atau korsleting. Gambar 2.20(a) menunjukkan IC yang dimasukkan ke soketnya secara sembarangan, menyebabkan pin 14 kehilangan lubangnya dan bertindak seperti terbuka. Pada Gambar 2.20(b), papan sirkuit tercetak jelas retak, yang menyebabkan sirkuit terbuka di setiap jejak tembaga yang digunakan untuk melewati retakan.

Penyolderan yang buruk menghasilkan jembatan solder yang terlihat pada Gambar 2.20(c). Di tengah foto ini, dapat melihat di mana terlalu banyak solder digunakan, menyebabkan jembatan listrik antara dua pin IC yang berdekatan dan membuatnya pendek. Pemecah masalah berpengalaman juga akan secara visual memeriksa papan sirkuit tercetak untuk komponen yang mungkin tampak gelap karena panas yang berlebihan. Perhatikan empat transistor di tengah Gambar 2.20(d). Yang di kiri bawah terlihat hangus dan mungkin terbakar habis, sehingga berfungsi seperti terbuka. Contoh pemecahan masalah berikut akan mengilustrasikan beberapa teknik pemecahan masalah dasar menggunakan probe logika dan pulser.

### Contoh 2.5

Gerbang IC AND pada Gambar 2.21 diduga mengalami kesalahan dan ingin mengujinya. Prosedur apa yang harus diikuti?



Gambar 2. 21 Koneksi Untuk Troubleshooting Satu Gerbang Quad AND IC

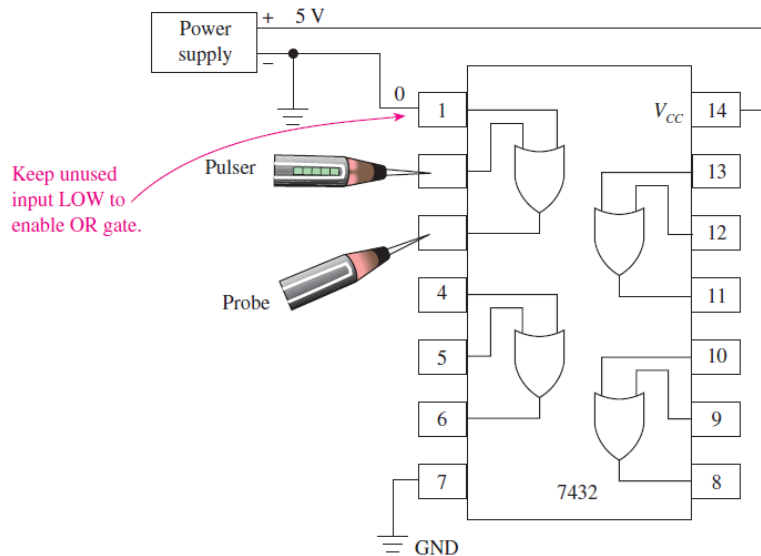
### Solusi:

Pertama menerapkan daya ke V<sub>CC</sub> (pin 14) dan GND (pin 7). Selanjutnya ingin memeriksa setiap gerbang AND dengan pulser/probe. Karena dibutuhkan *HIGH* (1) pada kedua input ke gerbang AND untuk membuat output menjadi *HIGH*, jika menempatkan *HIGH* (5V) pada satu input dan pulsa yang lain, akan mengharapkan untuk mendapatkan pulsa pada output dari gerbang. Gambar 2.21 menunjukkan koneksi untuk menguji salah satu gerbang quad AND IC. Ketika pulser diletakkan pada pin 12, lampu di ujung probe berkedip dengan kecepatan yang sama dengan pulser, menunjukkan bahwa gerbang AND melewati pulsa melalui gerbang (mirip dalam operasinya dengan rangkaian pengaktifan *clock* pada Gambar 2.17). Pemeriksaan selanjutnya

adalah membalikkan koneksi ke pin 12 dan 13 dan memeriksa probe. Jika probe masih menyala, gerbang itu baik-baik saja. Lanjutkan ke tiga gerbang lainnya dan ikuti prosedur yang sama. Ketika salah satu keluaran gerbang tidak berkedip, maka telah menemukan kesalahannya. Seperti disebutkan sebelumnya, kunci untuk mengatasi masalah IC adalah memahami cara kerja IC.

### Contoh 2.6

Buat sketsa koneksi untuk memecahkan masalah gerbang pertama dari gerbang 7432 quad OR.



Gambar 2. 22 Koneksi untuk troubleshooting satu gerbang OR pada IC 7432

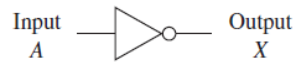
Solusi:

Koneksi ditunjukkan pada Gambar 2.22 Probe seharusnya berkedip jika gerbangnya bagus. Perhatikan bahwa masukan kedua ke gerbang OR yang diperiksa terhubung ke *LOW* (0) bukan *HIGH*. Alasannya adalah bahwa output akan selalu *HIGH* jika salah satu input terhubung *HIGH*. Karena satu input terhubung *LOW*, output akan berkedip bersamaan dengan pulsa dari logic pulser jika gerbangnya bagus.

## 2.6 Inverter

Inverter digunakan untuk melengkapi, atau membalikkan, sinyal digital. Ini memiliki input tunggal dan output tunggal. Jika level *HIGH* (1) masuk, itu menghasilkan output level *LOW* (0). Jika level *LOW* (0) masuk, itu menghasilkan output level *HIGH* (1). Simbol dan tabel kebenaran untuk gerbang inverter ditunjukkan pada Gambar 2.23. (Catatan: Lingkaran adalah bagian dari

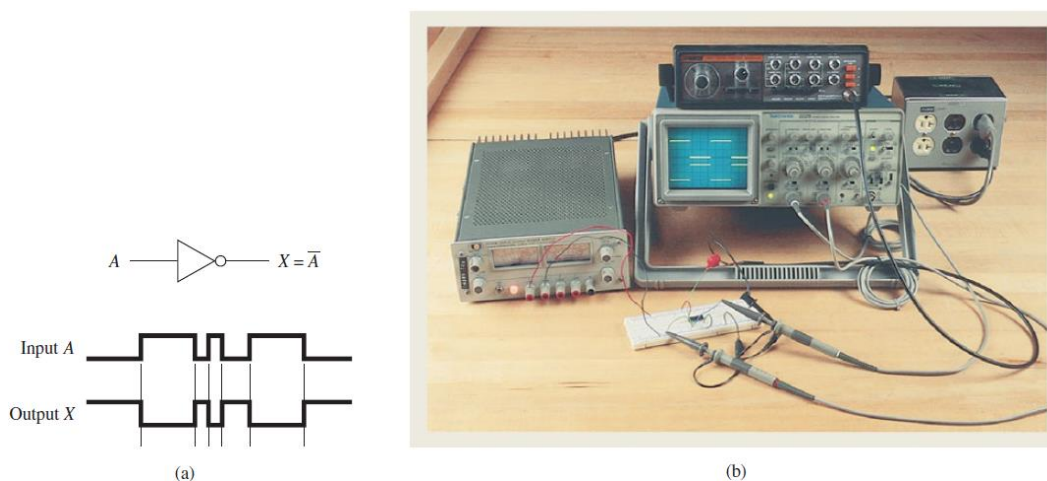
simbol yang menunjukkan pembalikan. Lingkaran pembalikan akan digunakan di gerbang lain di bagian yang akan datang).



Input A	Output X
0	1
1	0

Gambar 2. 23 Simbol Inverter dan Tabel Kebenaran

Pengoperasian inverter sangat sederhana dan dapat diilustrasikan lebih lanjut dengan mempelajari diagram waktu pada Gambar 2.24. Diagram waktu secara grafis menunjukkan kepada pengoperasian inverter. Jika inputnya *HIGH*, outputnya *LOW*, dan jika inputnya *LOW*, outputnya *HIGH*. Oleh karena itu, bentuk gelombang keluaran adalah pelengkap yang tepat dari masukan. Persamaan Boolean untuk inverter ditulis  $X = \bar{A}$  ( $X = \text{NOT } A$ ). Batang di atas A adalah batang inversi, digunakan untuk menandakan komplemen. Inverter kadang-kadang disebut sebagai gerbang NOT.

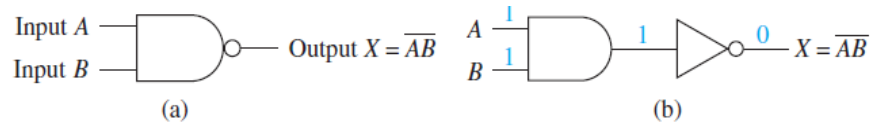


Gambar 2. 24 Analisis Waktu Gerbang Inverter: (a) Sketsa Bentuk Gelombang dan (b) Tampilan Osiloskop

## 2.7 Gerbang NAND

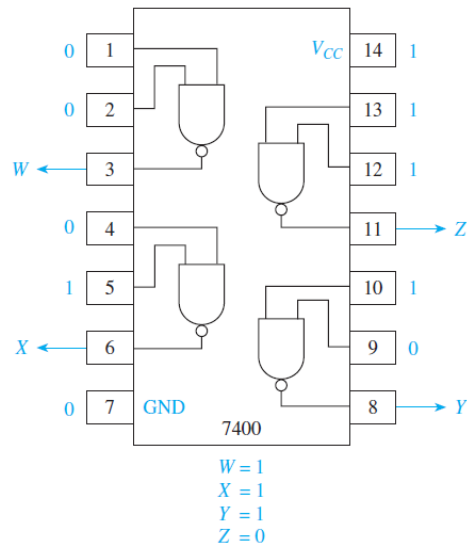
Pengoperasian gerbang NAND sama dengan gerbang AND kecuali outputnya terbalik. Maka dapat menganggap gerbang NAND sebagai gerbang AND dengan inverter pada keluarannya. Simbol gerbang NAND dibuat dari gerbang AND dengan lingkaran inversi (gelembung) pada keluarannya, seperti yang ditunjukkan pada Gambar 2.25(a).





Gambar 2. 25 Gerbang NAND: (a) Simbol; (b) AND–INVERT Setara Dengan Gerbang NAND Dengan  $A = 1, B = 1$

Dalam diagram sirkuit digital, akan menemukan lingkaran kecil yang digunakan setiap kali tindakan komplementer (inversi) ditunjukkan. Lingkaran pada output bertindak seperti sebuah inverter, sehingga gerbang NAND dapat digambar secara simbolis sebagai gerbang AND dengan inverter yang terhubung ke outputnya, seperti yang ditunjukkan pada Gambar 2.25(b). Bentuk *TTL* dari NAND adalah IC 7400 (atau 74LS00 atau 74HC00, dan lain-lain). Gambar 2.26 menunjukkan hasil keluaran untuk semua kemungkinan kombinasi masukan yang diterapkan pada 7400 quad NAND.



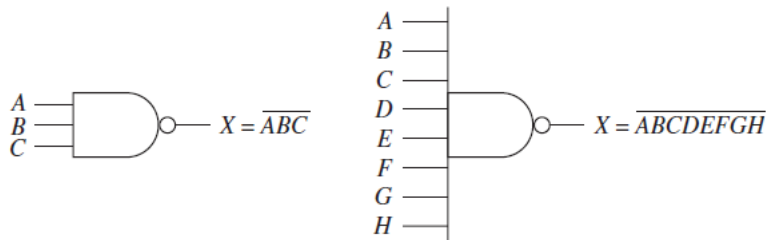
Gambar 2. 26 Input Dan Output IC NAND 7400 Quad

Persamaan Boolean untuk gerbang NAND ditulis  $X = \overline{AB}$ . Bilah inversi digambar di atas (A dan B), artinya keluaran NAND adalah komplemen dari (A dan B) [NOT (A dan B)]. Karena membalikkan keluaran, keluaran tabel kebenaran pada Tabel 2.6 akan menjadi komplemen dari keluaran tabel kebenaran gerbang AND. Cara mudah untuk membuat tabel kebenaran adalah dengan memikirkan bagaimana gerbang AND akan merespons input dan kemudian membalikkan jawaban. Dari Tabel 2.6, dapat melihat bahwa keluarannya *LOW* ketika kedua masukan A dan B *HIGH* (kebalikan dari gerbang AND). Juga, outputnya *HIGH* setiap kali salah satu inputnya *LOW*.

Tabel 2. 6 Tabel Kebenaran Gerbang NAND Dua Masukan

<i>A</i>	<i>B</i>	$X = \overline{AB}$
0	0	1
0	1	1
1	0	1
1	1	0

Gerbang NAND juga dapat memiliki lebih dari dua input. Gambar 2.27 menunjukkan simbol gerbang NAND tiga dan delapan masukan. Tabel kebenaran untuk gerbang NAND tiga masukan (lihat Tabel 2.7) menunjukkan bahwa keluaran selalu *HIGH* kecuali semua masukan menjadi *HIGH*.



Gambar 2. 27 Simbol Untuk Gerbang NAND Tiga Dan Delapan Masukan

Tabel 2. 7 Tabel Kebenaran Untuk Gerbang NAND Tiga Masukan

<i>A</i>	<i>B</i>	<i>C</i>	<i>X</i>
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Analisis waktu juga dapat digunakan untuk mengilustrasikan pengoperasian gerbang NAND.

Contoh 2.7

Buat sketsa bentuk gelombang keluaran pada X untuk gerbang NAND yang ditunjukkan pada Gambar 2.28, dengan bentuk gelombang masukan yang diberikan pada Gambar 2.29.

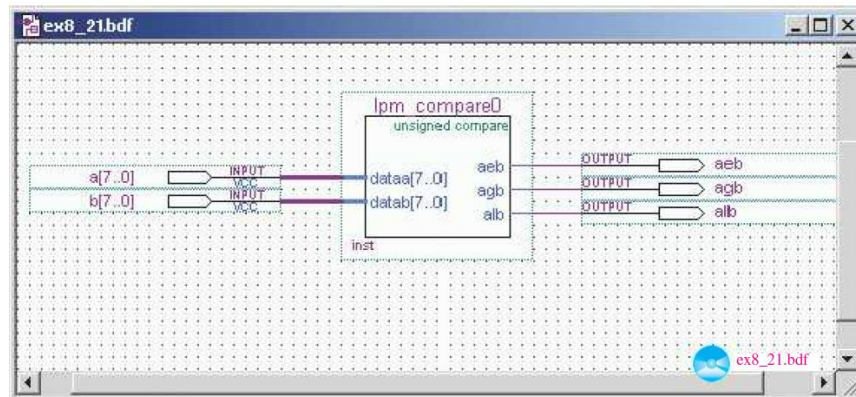


Gambar 2. 28 Gerbang untuk Contoh 2.7

Buat file desain grafis yang menggunakan modul *LPM* untuk membandingkan dua string 8-bit.

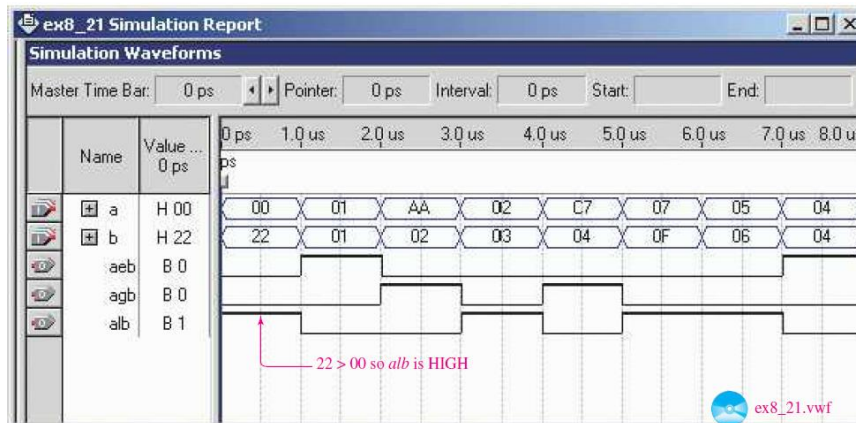
Solusi:

Modul bernama *lpm\_compare* ditemukan di subdirektori */megafunctions/arithmetic*. Untuk sampai ke sana, pilih: *File>New>Block Diagram*. Kemudian klik kanan di area kerja dan sisipkan simbol *lpm\_compare*. Gunakan manajer *MegaWizard* seperti yang dijelaskan pada Contoh 7–28 untuk menentukan *LPM*. Tentukan input 8-bit dan tiga output berikut:  $a = b$  (*aeb*),  $a > b$  (*agb*) dan  $a < b$  (*alb*). Juga di halaman 2 centang *No* dan *unsigned* dan di halaman 3 centang *No*. Diagram blok yang dihasilkan oleh *MegaWizard* akan terlihat seperti Gambar 5.42.



Gambar 5. 42 Modul *LPM* Untuk Komparator 8-Bit

File simulasi bentuk gelombang ditunjukkan pada Gambar 5.43. Periksa dengan cermat ketiga keluaran untuk setiap kombinasi masukan untuk memverifikasi operasi pembandingan yang benar.



Gambar 5. 43 Bentuk Gelombang Simulasi Untuk Pembandingan *LPM*

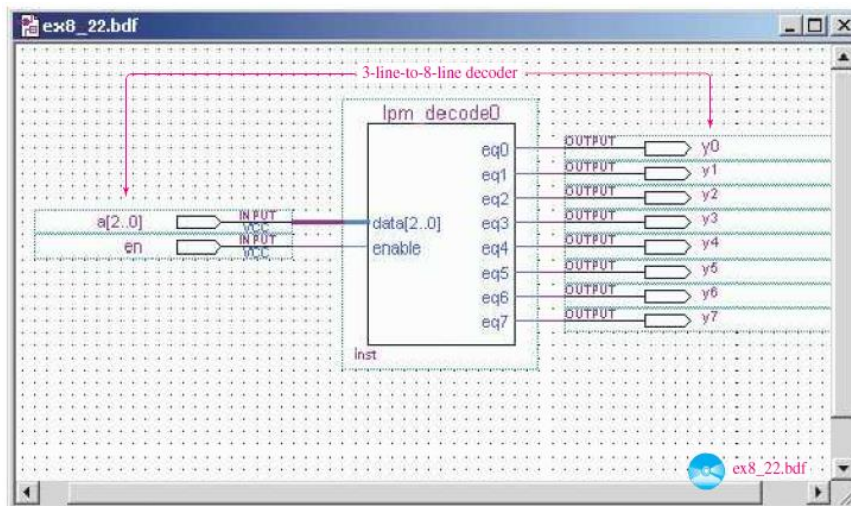
### Contoh 5.1

#### LPM Decoder

Buat file desain blok yang menggunakan modul *LPM* untuk membentuk dekoder 3 baris ke 8 baris (oktal).

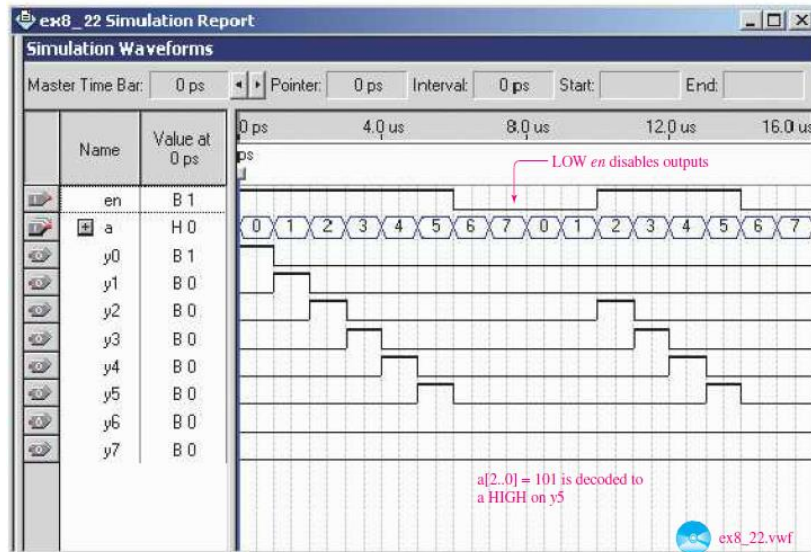
Solusi:

Modul bernama *lpm\_decode* ditemukan di subdirektori *megafunctions/arithmetic*. Untuk sampai ke sana, pilih: *File>New>Block Diagram*. Kemudian klik kanan di area kerja dan sisipkan simbol *lpm\_decode*. Gunakan manajer *MegaWizard* untuk mendefinisikan *LPM*. Tentukan bus masukan 3-bit dengan masukan Aktifkan. Tekan Tambahkan semua untuk memilih kedelapan output 'eq' untuk didekode. Diagram blok yang dihasilkan oleh *MegaWizard* akan tampak seperti Gambar 5.44.



Gambar 5. 44 Modul *LPM* Untuk Dekoder Oktal

File simulasi bentuk gelombang ditunjukkan pada Gambar 5.45. Perhatikan bahwa selama *en* bernilai *HIGH*, angka 3-bit pada masukan *a* akan menentukan keluaran *y* mana yang bernilai *HIGH*. Juga, ketika *en* adalah *LOW*, semua keluaran tetap *LOW*. Periksa dengan cermat ketujuh keluaran untuk setiap kombinasi masukan untuk memverifikasi pengoperasian yang benar.



Gambar 5. 45 Bentuk Gelombang Simulasi Untuk Dekoder LPM

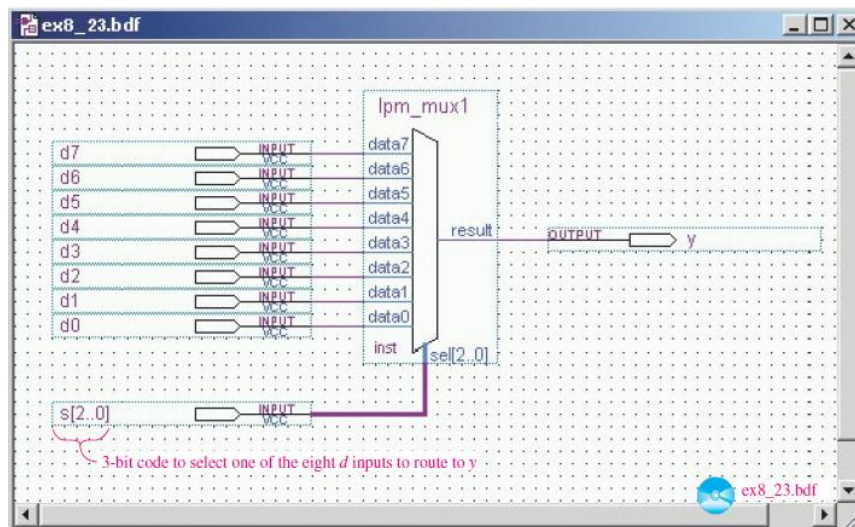
### Contoh 5.2

#### LPM Multiplexer

Buat file desain blok yang menggunakan modul LPM untuk membentuk multiplexer 8 baris.

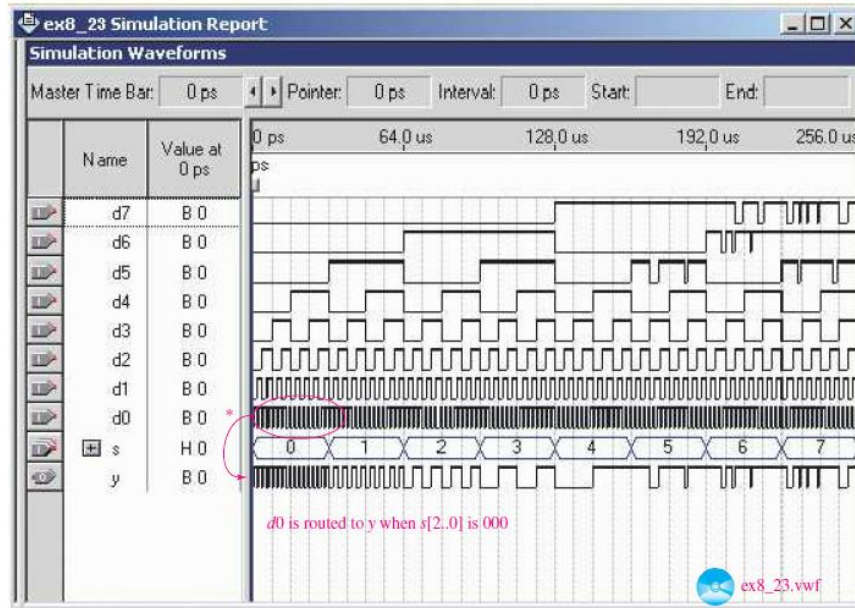
Solusi:

Modul bernama lpm\_mux terdapat di subdirektori/megafunctions/arithmetic. Untuk sampai ke sana, pilih: File>New>Block Diagram. Kemudian klik kanan pada area kerja dan masukkan simbol lpm\_mux. Gunakan manajer MegaWizard untuk mendefinisikan LPM. Tentukan 8 input data, bus input dan output 1-bit, dan Tanpa pipeline. Diagram blok yang dihasilkan oleh MegaWizard akan terlihat seperti Gambar 5.46.



Gambar 5. 46 Modul LPM Untuk Multiplexer 8 Jalur

File simulasi bentuk gelombang ditunjukkan pada Gambar 5.47. Beberapa bentuk gelombang tidak beraturan ditarik secara sewenang-wenang untuk menguji operasi. Untuk setiap nilai baru dari bit pemilihan (s), d yang sesuai dirutekan ke output y. Periksa dengan hati-hati y untuk memverifikasi bahwa itu menduplikasi bagian yang sesuai dari bentuk gelombang d untuk kedelapan nilai s.



Gambar 5. 47 Bentuk Gelombang Simulasi Untuk Multiplexser *LPM*

## DAFTAR PUSTAKA

- Akhiriyanto, P. M., Rahadjeng, I. R., & Purnamasari, I. (2019). *Rangkaian Digital*. Yogyakarta: Graha Ilmu.
- Ali, M., & Nugraha, A. C. (2018). *Teknik Digital Teori dan Aplikasi*. Yogyakarta: UNY Press.
- Arjit, S., & Nilotpal, M. (2007). *Digital Principles and Logic Design*. New Delhi: INFINITY SCIENCE PRESS.
- Brown, S., & Vranesic, Z. (2009). *Fundamentals of Digital Logic with VHDL Design* (3rd ed.). Canada: Higher Education.
- Budiharto, W. (2018). *Elektronika Digital dan Sistem Embedded*. Yogyakarta: Andi.
- Crowe, J., & Gill, B. H. (1998). *Introduction to Digital Electronic*. US: Elsevier.
- Harris, D. M., & Harris, S. L. (2013). *Digital Design and Computer Architecture* (Second ed.). USA: Elsevier.
- Kleitz, W. (2012). *Digital Electronics: A Practical Approach with VHDL* (9th ed.). New York: Pearson.
- Roth, J. C., & Kinney, L. L. (2014). *Fundamentals of Logic Design* (7rd ed.). USA: Cengage Learning.

## BIOGRAFI PENULIS



**Dody Susilo**, lahir di Pekanbaru, 06 Maret 1991 merupakan anak pertama dari dua bersaudara pasangan Bapak Budi Susilo dan Ibu Siti Wahyunnah Kusrintarsih. Penulis memulai pendidikan di SD Negeri 02 Mandau, lalu melanjutkan di SMP Negeri 04 Mandau. Penulis menempuh jenjang pendidikan di SMA Negeri 2 Mandau. Pada tahun 2013 penulis menempuh pendidikan sarjana di Fakultas Teknik di Program Studi Teknik Elektro Universitas Brawijaya Malang. Setelah menyelesaikan studi di tahap sarjana, pada tahun 2017 penulis meneruskan pendidikan di Program Magister Fakultas Teknologi Elektro di Departement Teknik Elektro Institut Teknologi Sepuluh Nopember Surabaya.

E-mail: [susilodody@unipma.ac.id](mailto:susilodody@unipma.ac.id)

Hp: 082234031206



**Ridam Dwi Laksono**, Lahir di Balikpapan 26 Agustus 1983. Lulus Pascasarjana Pendidikan Sains di Universitas Sebelas Maret Surakarta pada 28 April 2012. Pernah mengajar di prodi pendidikan matematika STKIP PGRI Ngawi Semenjak Mei 2012 hingga 26 September 2019. Kemudian awal Februari 2020 mengajar di Teknik Elektro Fakultas Teknik Universitas PGRI Madiun. Di prodi Teknik Elektro penulis mengampu matakuliah Kalkulus dan Matematika Teknik, Kewirausahaan dan Technopreneur. Di bidang teknologi penulis memiliki sertifikasi MTCNA, MTCRE dan IC3-Gmatrik. Di bidang Kepemudaan penulis juga telah mencapai MPG-Ungu dari Kementrian Pemuda Dan Olah Raga tahun 2015 dan alumni PUP - TAPLAI dari LEMHANNAS-RI tahun 2023.

E-mail: [ridam.dl@unipma.ac.id](mailto:ridam.dl@unipma.ac.id), [ridam83@gmail.com](mailto:ridam83@gmail.com)

Hp: 081234105388