

PERWAKILAN NOMBOR

Sistem Nombor

- Kebanyakan sistem komputer (sistem Digital) melakukan operasi pengiraan nombor dalam kuantiti yang banyak.
- Maka, Sistem Pernomboran yang digunakan oleh Sistem Digital perlu diketahui dari segi:-

Jenis-jenis Sistem Nombor

- Decimal (asas 10)
- Binary (asas 2)
- Octal (asas 8)
- Hexadecimal (asas 16)

Nombor Decimal

- Terdiri daripada 10 angka iaitu 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. Ia merupakan nombor 'Asas 10'.
- Salah satu contoh dalam sistem nombor Decimal adalah **1428.79** atau **1428.79₁₀**. Kedudukan setiap digit menunjukkan magnitud bagi setiap digit tersebut iaitu:-

Pemberat	10^3	10^2	10^1	10^0		10^{-1}	10^{-2}
Nilai	1	4	2	8	.	7	9

- Secara pernyataan matematik:-

$$1428_{10} = 1 \times 10^3 + 4 \times 10^2 + 2 \times 10^1 + 8 \times 10^0$$

Nombor Binari

- Terdiri daripada 2 angka iaitu 0,1. Ia merupakan nombor 'Asas 2'.
- Salah satu contoh dalam sistem nombor Binary adalah **1001.01** atau **1001.01₂**. Kedudukan setiap digit menunjukkan magnitud bagi setiap digit tersebut iaitu:-

Pemberat	2^3	2^2	2^1	2^0		2^{-1}	2^{-2}
Nilai	1	0	0	1	.	0	1

- Secara pernyataan matematik:-

$$1001_2 = 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$

- Dalam no. Binari, bilangan digit dipanggil *bit*.

Nombor Octal

- Terdiri daripada 8 angka iaitu 0, 1, 2, 3, 4, 5, 6, 7. Ia merupakan nombor 'Asas 8'.
- Salah satu contoh dalam sistem nombor Octal adalah **5641.27** atau **5641.27₈**. Kedudukan setiap digit menunjukkan magnitud bagi setiap digit tersebut iaitu:-

Pemberat	8^3	8^2	8^1	8^0		8^{-1}	8^{-2}
Nilai	5	6	4	1	.	2	7

- Secara pernyataan matematik:-

$$5641_8 = 5 \times 8^3 + 6 \times 8^2 + 4 \times 8^1 + 1 \times 8^0$$

Nombor Heksadesimal

- Terdiri daripada 16 angka iaitu 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. Ia merupakan nombor 'Asas 16'.
- Salah satu contoh dalam sistem nombor Hexadesimal adalah **5B8F.27** atau **5B8F.27₁₆**. Kedudukan setiap digit menunjukkan magnitud bagi setiap digit tersebut iaitu:-

Pemberat	16^3	16^2	16^1	16^0		16^{-1}	16^{-2}
Nilai	5	B	8	F	.	2	7

- Secara pernyataan matematik:-

$$5B8F_{16} = 5 \times 16^3 + B \times 16^2 + 8 \times 16^1 + F \times 16^0$$

Decimal	Binary	Octal	Hexadecimal
0	0000	00	0
1	0001	01	1
2	0010	02	2
3	0011	03	3
4	0100	04	4
5	0101	05	5
6	0110	06	6
7	0111	07	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Penukaran Nombor

- Penukaran Binari ke Desimal

$$1001_2 = 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$

$$= 8 + 0 + 0 + 1$$

- Penukaran Desimal ke Binari

$$18_{10} = ?_2$$

Berhenti apabila $N = 0$

$$18_{10} = 10010_2$$

<i>N</i>	<i>X</i>	<i>Baki</i> (<i>N-X</i>)
18	=18	0
9	=8	1
4	=4	0
2	=2	0
1	=0	1
0		

- Penukaran Octal ke Desimal

$$1271_8 = 1 \times 8^3 + 2 \times 8^2 + 7 \times 8^1 + 1 \times 8^0$$

$$= 512 + 128 + 56 + 1$$

- Penukaran Octal ke Desimal

$$697_{10} = ?_8$$

<i>N</i>	<i>X</i>	<i>Baki</i> (<i>N-X</i>)
697	=696	1
87	=80	7
10	=8	2
1	=0	1
0		

Latihan

- Tukar no. Decimal kepada no. Binary, (a) 33_{10} (b) 78_{10} (c) 101_{10}
(a) 33_{10} (b) 78_{10} (c) 101_{10}
- Tukar no. Binary kepada no. Decimal, (a) 111101_2 (b) 011010110_2 (c) 0101_2
(a) 111101_2 (b) 011010110_2 (c) 0101_2
- Tukar no. pecahan kepada no. Binary, (a) 23.25_{10} (b) 68.75_{10}
(a) 23.25_{10} (b) 68.75_{10}


GET LOGIK

Pengenalan Get Logik


- Logik dalam sistem Binari digunakan untuk menyatakan proses dan operasi sesuatu maklumat Binary dari segi penyataan matematik!
- Logik Binary terdiri pembolehubah Binary dan juga **operasi logik**'.

Get Logik Asas


Get TAK

Simbol			
Fungsi		A	C
Jadual benar (truth table)		0	1
		1	0
$C = \overline{A}$ @ $C = \text{NOT } A$			


Get ATAU

Simbol				
Fungsi		A	B	C
Jadual benar (truth table)		0	0	0
		0	1	1
		1	0	1
		1	1	1
	$C = A+B$ @ $C = A \text{ OR } B$			

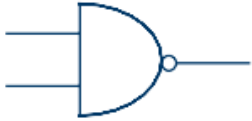
Get DAN

Simbol				
Fungsi		A	B	C
Jadual benar (truth table)		0	0	0
		0	1	0
		1	0	0
		1	1	1
	$C = A \cdot B$ @ $C = A \text{ AND } B$			


Get TAKATAU

Simbol				
Fungsi		A	B	C
Jadual benar (truth table)		0	0	1
		0	1	0
		1	0	0
		1	1	0
$C = \overline{A+B}$ @ $C = A \text{ NOR } B$				


Get TAKDAN

Simbol				
Fungsi		A	B	C
Jadual benar (truth table)		0	0	1
		0	1	1
		1	0	1
		1	1	0
$C = \overline{A \cdot B}$ @ $C = A \text{ NAND } B$				

Get EKSLUSIF-ATAU

Simbol				
Fungsi		A	B	C
Jadual benar (truth table)		0	0	0
		0	1	1
		1	0	1
		1	1	0
$C = A \cdot B + \overline{A} \cdot \overline{B} = A \oplus B$ @ $C = A \text{ EX-OR } B$				

Get EKSLUSIF-TAKATAU

Simbol				
Fungsi		A	B	C
Jadual benar (truth table)		0	0	1
		0	1	0
		1	0	0
		1	1	1
$C = \bar{A} \cdot \bar{B} + A \cdot B = A \oplus B$ <p>@ C = A EX-NOR B</p>				

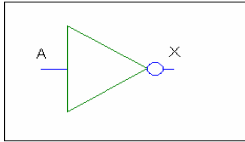
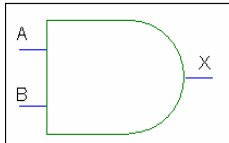
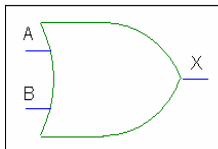
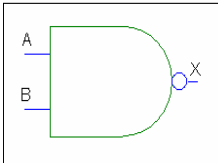
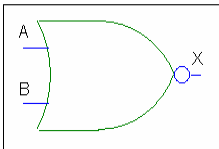
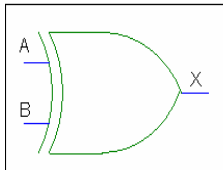
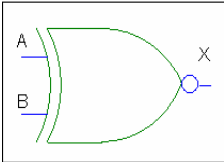
Get Logik dan Jadual Kebenaran

- Sistem penduaan menjadi sistem nombor pilihan dalam litar elektronik dan sistem perkomputeran. Ini adalah kerana hanya dua simbol atau digit digunakan, iaitu “1” dan “0”.
- Dua bit ini dapat mewakili dua paras voltan yang terhasil dalam litar elektronik. Paras 5 V diwakili oleh bit “1” manakala paras 0 V diwakili oleh bit “0”.
- Bit “1” dan “0” juga boleh mewakili mana- mana dua keadaan (state) yang dihasilkan oleh sesuatu sistem seperti yang ditunjukkan dalam jadual

Sistem	Perwakilan “0”	Perwakilan “0”
Pernyataan berlogik	Salah / Palsu	Betul / Benar
Pernyataan kepastian	Tidak	Ya
Pensuisan	Tutup	Buka
Aras Voltan	Tinggi / 5 V	Rendah / 0 V
Lampu	Tidak menyala	Menyala

Jadual 3.1 Perwakilan Nombor Penduaan

- Konsep perwakilan dalam jadual di atas disebut sebagai logik positif. Sekiranya perwakilan “1” dan “0” diterbalikkan, iaitu “1” mewakili 0 V dan “0” mewakili 5 V, maka perwakilan ini disebut logik negatif.
- Bagaimanakah litar elektronik beroperasi hanya dengan menggunakan logik “1” dan “0”?
- Beberapa teorem dan peraturan turut diperkenalkan yang dikenali sebagai Teorem Boole (Boolean Theorem) atau Peraturan Boole (Boolean Rules).
- Dalam litar elektronik, elemen yang dapat melaksanakan operasi-operasi logik ini dikenali sebagai get. Get memerlukan satu atau lebih input tetapi hanya menghasilkan satu output sahaja.
- Terdapat tiga get asas: get TAK (**NOT gate** / Inverter), get ATAU (**OR gate**) dan get DAN (**AND gate**). Daripada tiga get asas ini, beberapa get lain yang mempunyai operasi tertentu dapat dihasilkan.
- Antaranya get TAK-ATAU (**NOR gate**), get TAK-DAN (**NAND gate**), get eksklusif ATAU (**XOR gate**) dan get eksklusif TAK-ATAU (**XNOR gate**).
- Setiap get mempunyai jadual kebenaran (truth table). Jadual kebenaran ialah satu senarai semua input yang mungkin dan output yang sepadan. Rajah berikut menunjukkan simbol, jadual kebenaran, pernyataan Boolean dan nombor cip untuk setiap jenis get.

Get	Simbol logik	Pernyataan Boole	Jadual Kebenaran															
TAK (Inverter) No. cip: 7404		$X = A$ —	<table><tr><th>A</th><th>X</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	A	X	0	1	1	0									
A	X																	
0	1																	
1	0																	
DAN (AND) No. cip: 7408		$X = A.B$	<table><tr><th>A</th><th>B</th><th>X</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	X	0	0	0	0	1	0	1	0	0	1	1	1
A	B	X																
0	0	0																
0	1	0																
1	0	0																
1	1	1																
ATAU (OR) No. cip: 7432		$X = A + B$	<table><tr><th>A</th><th>B</th><th>X</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	X	0	0	0	0	1	1	1	0	1	1	1	1
A	B	X																
0	0	0																
0	1	1																
1	0	1																
1	1	1																
TAK-DAN (NAND) No. cip: 7400		$X = A.B$ —	<table><tr><th>A</th><th>B</th><th>X</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	A	B	X	0	0	1	0	1	1	1	0	1	1	1	0
A	B	X																
0	0	1																
0	1	1																
1	0	1																
1	1	0																
TAK-ATAU (NOR) No. cip: 7402		$X = \overline{A + B}$	<table><tr><th>A</th><th>B</th><th>X</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	A	B	X	0	0	1	0	1	0	1	0	0	1	1	0
A	B	X																
0	0	1																
0	1	0																
1	0	0																
1	1	0																
X-ATAU (XOR) No. cip: 7486		$X = AB + A\overline{B}$ $= A \oplus B$ —	<table><tr><th>A</th><th>B</th><th>X</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	A	B	X	0	0	0	0	1	1	1	0	1	1	1	0
A	B	X																
0	0	0																
0	1	1																
1	0	1																
1	1	0																
X- TAK-ATAU (XNOR)		$X = A\overline{B} + \overline{A}B$ $= A \oplus B$ —	<table><tr><th>A</th><th>B</th><th>X</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	X	0	0	1	0	1	0	1	0	0	1	1	1
A	B	X																
0	0	1																
0	1	0																
1	0	0																
1	1	1																

Jadual 3.1 Simbol, Pernyataan Boolean dan Jadual Keben

