

1 SISTEM BILANGAN

Desimal , Biner, Oktal dan Heksadesimal

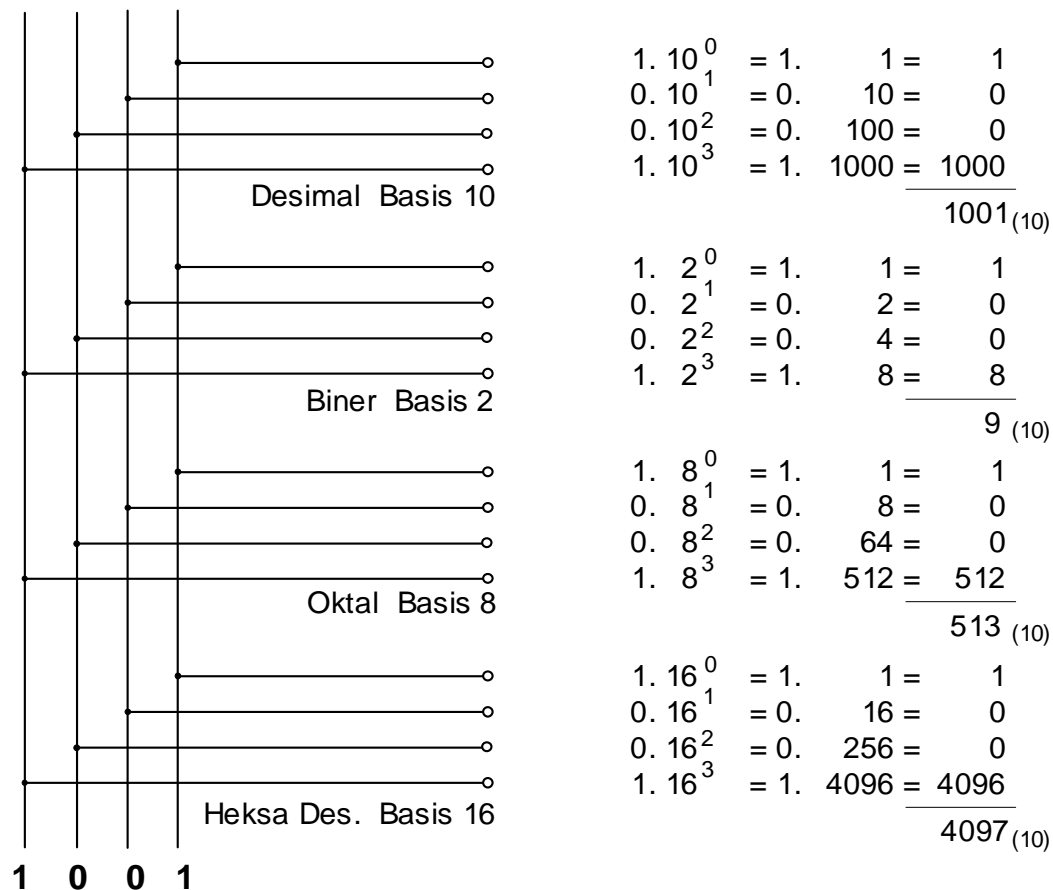
Tujuan : Setelah mempelajari Sistem Bilangan diharapkan dapat,

1. Memahami jenis-jenis sistem bilangan yang digunakan pada teknik mikroprosesor
2. Memahami konversi sistem bilangan desimal ke sistem bilangan biner
3. Memahami konversi sistem bilangan desimal ke sistem bilangan oktal
4. Memahami konversi sistem bilangan desimal ke sistem bilangan heksadesimal
5. Memahami konversi sistem bilangan biner ke sistem bilangan oktal atau sebaliknya
6. Memahami konversi sistem bilangan biner ke sistem bilangan heksadesimal atau sebaliknya
7. Memahami konversi sistem bilangan desimal dan sistem bilangan biner antara 0 dan 1
8. Mampu merubah bilangan desimal ke bentuk BCD atau sebaliknya
9. Mampu merubah bilangan desimal ke bentuk BCH atau sebaliknya
10. Memahami ASCII Code untuk pembentukan karakter

1.1. Sistem Bilangan

1.1.1. Umum

Dalam kehidupan sehari-hari, bilangan yang kita gunakan untuk menghitung adalah bilangan yang berbasis 10 atau disebut Sistem Desimal. Setiap tempat penulisan dapat terdiri dari simbol-simbol 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. Susunan penulisan bilangan menunjukkan harga / nilai tempat dari bilangan tersebut misalnya, satuan, puluhan, ratusan dst. Tempat penulisan semakin ke kiri menunjukkan nilai tempat bilangan yang semakin tinggi. Dalam teknik Digital maupun teknik mikroprosesor pada umumnya bilangan yang dipakai adalah bilangan yang berbasis 2 atau Sistem Biner. Dalam sistem biner di setiap tempat penulisan hanya mungkin menggunakan simbol 0, atau simbol 1, sedangkan nilai tempat bilangan tersusun seperti pada sistem desimal. Di bawah ini adalah bilangan 1001 dalam beberapa bentuk sistem bilangan.



Beberapa Sistem Bilangan

Disamping sistem Desimal dan sistem Biner dalam gambar terlihat pula bilangan yang berbasis 8 atau sistem Oktal dan bilangan yang berbasis 16 atau sistem Heksadesimal.

1.1.2. Sistem Desimal (Dinari)

Pada sistem desimal (lat. decum =10), seperti telah kita ketahui bersama bahwa sistem ini berbasis 10 dan mempunyai 10 simbol yaitu dari angka 0 hingga 9. Setiap tempat mempunyai nilai kelipatan dari 10^0 , 10^1 , 10^2 , dst . Penulisan bilangan terbagi dalam beberapa tempat dan banyaknya tempat tergantung dari besarnya bilangan. Setiap tempat mempunyai besaran tertentu yang harga masing-masing tempat secara urut dimulai dari kanan disebut

	ribuan	ratusan	puluhan	satuan
	10^3	10^2	10^1	10^0

Contoh

Angka Desimal 10932 ($10932_{(10)}$)

1 0 9 3 2

	Pertama	$2 \cdot 10^0 = 2 \cdot 1 = 2$
	Kedua	$3 \cdot 10^1 = 3 \cdot 10 = 30$
	Ketiga	$9 \cdot 10^2 = 9 \cdot 100 = 900$
	Keempat	$0 \cdot 10^3 = 0 \cdot 1000 = 0$
	Kelima	$1 \cdot 10^4 = 1 \cdot 10000 = 10000$
		<hr/> 10932

Kebiasaan sehari-hari harga suatu bilangan desimal dituliskan dalam bentuk yang mudah sbb :

$$\begin{aligned} 10932 &= 1 \cdot 10000 + 0 \cdot 1000 + 9 \cdot 100 + 3 \cdot 10 + 2 \cdot 1 \\ &= 1 \cdot 10^4 + 0 \cdot 10^3 + 9 \cdot 10^2 + 3 \cdot 10^1 + 2 \cdot 10^0 \end{aligned}$$

1.1.3. Sistem Biner

Sistem Biner (lat. Dual) atau “duo” yang berarti 2, banyak dipakai untuk sinyal elektronik dan pemrosesan data. Kekhususan sistem biner untuk elektronik yaitu bahwa sistem biner hanya mempunyai 2 simbol yang berbeda, sehingga pada sistem ini hanya dikenal angka “0” dan angka “1”.

Contoh

1 0 1 0 1

	Pertama	$1 \cdot 2^0 = 1 \cdot 1 = 1$
	Kedua	$0 \cdot 2^1 = 0 \cdot 2 = 0$
	Ketiga	$1 \cdot 2^2 = 1 \cdot 4 = 4$
	Keempat	$0 \cdot 2^3 = 0 \cdot 8 = 0$
	Kelima	$1 \cdot 2^4 = 1 \cdot 16 = 16$
		<hr/> 21

Dari gambaran di atas seperti halnya pada sistem desimal, cara penulisannya dapat dinyatakan secara langsung sbb :

$$\begin{aligned} 10101 &= 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 \\ \text{Dual} &= 1 \cdot 16 + 0 \cdot 8 + 1 \cdot 4 + 0 \cdot 2 + 1 \cdot 1 \\ &= 21 \text{ (desimal) } \end{aligned}$$

Setiap tempat pada bilangan biner mempunyai kelipatan 2^0 , 2^1 , 2^2 , 2^3 dst. yang dihitung dari kanan kekiri. Selanjutnya kita juga dapat merubah bilangan desimal ke bilangan biner atau sebaliknya dari bilangan biner ke bilangan desimal.

1.1.6. Konversi Basis Bilangan

1.1.6.1. Konversi Bilangan Desimal Ke Sistem Bilangan Lain

Sistem bilangan desimal secara mudah dapat dirubah dalam bentuk sistem bilangan yang lain. Ada banyak cara untuk melakukan konversi bilangan, proses yang paling mudah dan sering digunakan untuk memindah bentuk bilangan adalah “ Proses Sisa “. Tabel di bawah memperlihatkan bilangan 0 sampai 22 basis 10 (desimal) dalam bentuk bilangan berbasis 2 (biner), berbasis 8 (Oktal) dan berbasis 16 (Heksadesimal).

Basis 10	Basis 2	Basis 8	Basis 16
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	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
16	10000	20	10
17	10001	21	11
18	10010	22	12
19	10011	23	13
20	10100	24	14
21	10101	25	15
22	10110	26	16

Untuk merubah bilangan desimal ke bilangan yang berbasis lain cukup membagi bilangan desimal dengan basis bilangan yang baru hingga habis.

Contoh 1

Konversi Bilangan Desimal $Z_{(10)} = 83$ ke bilangan Biner $Z_{(2)}$ 83 dibagi dengan basis bilangan baru yaitu 2

$$83 : 2 = 41 \quad \text{sisa } 1.$$

Sisa 1 ini merupakan digit pertama dari bilangan biner ...x x x x 1. Untuk mendapatkan harga pada digit berikutnya adalah :

$$41 : 2 = 20 \quad \text{sisa } 1$$

Sisa 1 ini menempati digit selanjutnya sehingga bentuk binernya ...x x x 1 1 dan seterusnya seperti di bawah ini.

83	:	2	=	41	sisa	1	
41	:	2	=	20	sisa	1	
20	:	2	=	10	sisa	0	
10	:	2	=	5	sisa	0	
5	:	2	=	2	sisa	1	
2	:	2	=	1	sisa	0	
1	:	2	=	0	sisa	1	

$83_{(10)} = 1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 1_{(2)}$

Jadi $Z_{(10)} = 83$ adalah $Z_{(2)} = 1010011$. Untuk meyakinkan bahwa hasil konversi di atas benar maka kita lakukan test sbb :

$$\begin{aligned}
 \text{Test} &\rightarrow 1 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 \\
 &= 1 \cdot 64 + 0 \cdot 32 + 1 \cdot 16 + 0 \cdot 8 + 0 \cdot 4 + 1 \cdot 2 + 1 \cdot 1 \\
 Z_{(10)} &= 83
 \end{aligned}$$

Contoh 2

Konversi Bilangan Desimal $Z_{(10)} = 1059$ ke bilangan Oktal $Z_{(8)}$

1059	:	8	=	132	sisa	3	
132	:	8	=	16	sisa	4	
16	:	8	=	2	sisa	0	
2	:	8	=	0	sisa	2	

$1059_{(10)} = 2 \quad 0 \quad 4 \quad 3_{(8)}$

Jadi $Z_{(10)} = 1059$ adalah $Z_{(8)} = 2043$

$$\begin{aligned}
 \text{Test} &\rightarrow 2 \cdot 8^3 + 0 \cdot 8^2 + 4 \cdot 8^1 + 3 \cdot 8^0 \\
 &= 2 \cdot 512 + 0 \cdot 64 + 4 \cdot 8 + 3 \cdot 1 \\
 &= 1024 + 0 + 32 + 3 \\
 Z_{(10)} &= 1059
 \end{aligned}$$

Contoh 3

Konversi Bilangan Desimal $Z_{(10)} = 10846$ ke bilangan Heksadesimal $Z_{(16)}$

$$\begin{array}{rcll}
 10846 : 16 & = & 677 & \text{sisa } 14 \\
 677 : 16 & = & 42 & \text{sisa } 5 \\
 42 : 16 & = & 2 & \text{sisa } 10 \\
 2 : 16 & = & 0 & \text{sisa } 2
 \end{array}$$

$10846_{(10)} = 2 \quad A \quad 5 \quad E_{(16)}$

Jadi $Z_{(10)} = 10846$ adalah $Z_{(16)} = 2A5E$

$$\begin{aligned}
 \text{Test} &\rightarrow 2 \cdot 16^3 + 10 \cdot 16^2 + 5 \cdot 16^1 + 14 \cdot 16^0 \\
 &= 2 \cdot 4096 + 10 \cdot 256 + 5 \cdot 16 + 14 \cdot 1 \\
 &= 8192 + 2560 + 80 + 14 \\
 Z_{(10)} &= 10846
 \end{aligned}$$

1.1.6.2. Konversi Basis Bilangan Lain Ke Bilangan Desimal

Untuk merubah satu sistem bilangan ke bilangan desimal, cukup dengan mengalikan masing-masing angka dengan basis yang pangkatnya sesuai dengan tempat masing-masing. Hasil penjumlahan merupakan bilangan desimal yang dicari.

Contoh 1

Konversi Bilangan Biner $Z_{(2)} = 10101010$ ke bilangan Desimal $Z_{(10)}$

$$\begin{array}{rcll}
 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\
 \hline
 & & & & & & 0 \cdot 2^0 & = 0 \cdot 1 = 0 \\
 & & & & & & 1 \cdot 2^1 & = 1 \cdot 2 = 2 \\
 & & & & & & 0 \cdot 2^2 & = 0 \cdot 4 = 0 \\
 & & & & & & 1 \cdot 2^3 & = 1 \cdot 8 = 8 \\
 & & & & & & 0 \cdot 2^4 & = 0 \cdot 16 = 0 \\
 & & & & & & 1 \cdot 2^5 & = 1 \cdot 32 = 32 \\
 & & & & & & 0 \cdot 2^6 & = 0 \cdot 64 = 0 \\
 & & & & & & 1 \cdot 2^7 & = 1 \cdot 128 = 128 \\
 \hline
 & & & & & & & 10101010_{(2)} = 170_{(10)}
 \end{array}$$

Jadi $Z_{(2)} = 10101010$ adalah $Z_{(10)} = 170$

Contoh 2

Konversi Bilangan Oktal $Z_{(8)} = 4327$ ke bilangan Desimal $Z_{(10)}$

4 3 2 7

7	$\cdot 8^0$	=	7	$\cdot 1$	=	7
2	$\cdot 8^1$	=	2	$\cdot 8$	=	16
3	$\cdot 8^2$	=	3	$\cdot 64$	=	192
4	$\cdot 8^3$	=	4	$\cdot 512$	=	2048
						<hr/>
$4327_{(8)}$					=	$2263_{(10)}$

Jadi $Z_{(8)} = 4327$ adalah $Z_{(10)} = 2263$

Contoh 3

Konversi Bilangan Heksadesimal $Z_{(16)} = B3C9$ ke bilangan Desimal $Z_{(10)}$

B 3 C 9

9	$\cdot 16^0$	=	9	$\cdot 1$	=	9
12	$\cdot 16^1$	=	12	$\cdot 16$	=	192
3	$\cdot 16^2$	=	3	$\cdot 256$	=	768
11	$\cdot 16^3$	=	11	$\cdot 4096$	=	45056
						<hr/>
$B3C9_{(16)}$					=	$46025_{(10)}$

Jadi $Z_{(16)} = B3C9$ adalah $Z_{(10)} = 46025$

1.1.6.3. Konversi Basis Bilangan Ke Basis Bilangan Lain

Untuk merubah dari satu sistem bilangan ke sistem bilangan yang lain memerlukan dua langkah. Pertama kita rubah sistem bilangan yang lama ke bilangan desimal kemudian dari bilangan desimal dirubah ke sistem bilangan yang diinginkan.

Contoh 1

Konversi Bilangan Biner $Z_{(2)} = 101101$ ke bilangan Heksadesimal $Z_{(16)}$

Langkah Pertama

1 0 1 1 0 1

1	$\cdot 2^0$	=	1	$\cdot 1$	=	1
0	$\cdot 2^1$	=	0	$\cdot 2$	=	0
1	$\cdot 2^2$	=	1	$\cdot 4$	=	4
1	$\cdot 2^3$	=	1	$\cdot 8$	=	8
0	$\cdot 2^4$	=	0	$\cdot 16$	=	0
1	$\cdot 2^5$	=	1	$\cdot 32$	=	32
						<hr/>
$101101_{(2)}$					=	$45_{(10)}$

Langkah Kedua

$$\begin{array}{rclcl}
 45 & : & 16 & = & 2 & \text{ sisa } 13 \\
 2 & : & 16 & = & 0 & \text{ sisa } 2
 \end{array}$$

$$45_{(10)} = 2D_{(16)}$$

Jadi $Z_{(2)} = 101101$ adalah $Z_{(12)} = 2D$

Contoh 2

Konversi Bilangan Heksadesimal $Z_{(16)} = 2FC$ ke bilangan Biner $Z_{(2)}$

Langkah Pertama

2 F C

$$\begin{array}{rclcl}
 12 & \cdot & 16^0 & = & 12 \cdot 1 & = & 12 \\
 15 & \cdot & 16^1 & = & 15 \cdot 16 & = & 240 \\
 2 & \cdot & 16^2 & = & 2 \cdot 256 & = & 512 \\
 \hline
 & & & & 2FC_{(16)} & = & 764_{(10)}
 \end{array}$$

Langkah Kedua

$$\begin{array}{rclcl}
 764 & : & 2 & = & 382 & \text{ sisa } 0 \\
 382 & : & 2 & = & 191 & \text{ sisa } 0 \\
 191 & : & 2 & = & 95 & \text{ sisa } 1 \\
 95 & : & 2 & = & 47 & \text{ sisa } 1 \\
 47 & : & 2 & = & 23 & \text{ sisa } 1 \\
 23 & : & 2 & = & 11 & \text{ sisa } 1 \\
 11 & : & 2 & = & 5 & \text{ sisa } 1 \\
 5 & : & 2 & = & 2 & \text{ sisa } 1 \\
 2 & : & 2 & = & 1 & \text{ sisa } 0 \\
 1 & : & 2 & = & 0 & \text{ sisa } 1
 \end{array}$$

$$764_{(10)} = 1011111100_{(2)}$$

Jadi $Z_{(16)} = 2FC$ adalah $Z_{(2)} = 1011111100$

1.1.7. Bentuk Bilangan Desimal dan Bilangan Biner antara 0 dan 1

Pada pembahasan sebelumnya kita telah membicarakan tentang sistem bilangan, dan konversi bilangan dalam bentuk bilangan bulat positif. Kali ini kita akan membahas tentang bilangan antara 0 dan 1 yang kita kenal dengan sebutan bilangan pecahan positif. Untuk menuliskan bentuk bilangan pecahan desimal, kita cukup menuliskan koma (,) dibelakang bilangan bulatnya. Setiap tempat dibelakang koma mempunyai kelipatan $1/10$.

Di bawah ini adalah contoh penulisan bilangan pecahan desimal yang sering kita jumpai.

Contoh

0, 5 3 7 1 ₍₁₀₎			
1	tempat ke-4 setelah koma	$1 \cdot 1/10^4 = 1 \cdot 10^{-4} = 1 \cdot 0,0001 = 0,0001$	
7	tempat ke-3 setelah koma	$7 \cdot 1/10^3 = 7 \cdot 10^{-3} = 7 \cdot 0,001 = 0,007$	
3	tempat ke-2 setelah koma	$3 \cdot 1/10^2 = 3 \cdot 10^{-2} = 3 \cdot 0,01 = 0,03$	
5	tempat ke-1 setelah koma	$5 \cdot 1/10^1 = 5 \cdot 10^{-1} = 5 \cdot 0,1 = 0,5$	
0	tempat ke-1 sebelum koma	$0 \cdot 10^0 = 0 \cdot 1 = 0$	

$$0,5371 = 0 + 0,5 + 0,03 + 0,007 + 0,0001$$

Di bawah ini adalah bentuk bilangan biner antara 0₍₂₎ dan 1₍₂₎

Contoh

0, 1 0 1 ₍₂₎			
1	tempat ke-3 setelah koma	$1 \cdot 1/2^3 = 1 \cdot 2^{-3} = 1 \cdot 0,001 = 0,001$	
0	tempat ke-2 setelah koma	$0 \cdot 1/2^2 = 0 \cdot 2^{-2} = 0 \cdot 0,01 = 0,00$	
1	tempat ke-1 setelah koma	$1 \cdot 1/2^1 = 1 \cdot 2^{-1} = 1 \cdot 0,1 = 0,1$	
0	tempat ke-1 sebelum koma	$0 \cdot 2^0 = 0 \cdot 1 = 0$	

$$0,101_{(2)} = 0_{(2)} + 0,1_{(2)} + 0,00_{(2)} + 0,001_{(2)}$$

Untuk merubah bilangan desimal yang besarnya lebih kecil dari 1 (satu) ke bentuk bilangan biner kita lakukan proses perkalian seperti di bawah ini.

Contoh

$$\begin{aligned} 0,4375 \cdot 2 &= 0 \text{ sisa } 0,8750 \\ 0,8750 \cdot 2 &= 1 \text{ sisa } 0,7500 \\ 0,7500 \cdot 2 &= 1 \text{ sisa } 0,5000 \\ 0,5000 \cdot 2 &= 1 \text{ sisa } 0 \end{aligned}$$

$$\text{jadi } 0,4375_{(10)} = 0,0111_{(2)}$$

Sebagai koreksi untuk mengetahui kebenaran konversi, dapat kita lakukan proses balik seperti di bawah ini,

$$\begin{aligned} 0, \quad 0 \quad 1 \quad 1 \quad 1_{(2)} &= \\ 0 + 0 \cdot 2^{-1} + 1 \cdot 2^{-2} + 1 \cdot 2^{-3} + 1 \cdot 2^{-4} &= \\ 0 + 0,0,5 + 1,0,25 + 1,0,125 + 1,0,0625 &= 0,4375 \end{aligned}$$

Tidak semua konversi dari bilangan desimal ke bilangan biner menghasilkan sisa 0 seperti pada contoh di atas . Untuk mengatasi hal tsb. maka dalam konversi kita batasi sampai beberapa angka dibelakang koma. Semakin banyak angka dibelakang koma maka kesalahannya semakin kecil.

Contoh

$$\begin{array}{ll}
 0,5371 \cdot 2 = 1 \text{ sisa } 0,0742 & \\
 0,0742 \cdot 2 = 0 \text{ sisa } 0,1484 & \\
 0,1484 \cdot 2 = 0 \text{ sisa } 0,2968 & \\
 0,2968 \cdot 2 = 0 \text{ sisa } 0,5936 & \\
 0,5936 \cdot 2 = 1 \text{ sisa } 0,1872 & 0,5371_{(10)} = 0,10001_{(2)} \\
 0,1872 \cdot 2 = 0 \text{ sisa } 0,3744 & \\
 0,3744 \cdot 2 = 0 \text{ sisa } 0,7488 & \\
 0,7488 \cdot 2 = 1 \text{ sisa } 0,4976 & 0,5371_{(10)} = 0,10001001_{(2)}
 \end{array}$$

Jika proses diakhiri sampai perkalian kelima,

$$\begin{aligned}
 0,10001_{(2)} &= 0,5 + 0,03125 = 0,53125 \\
 \text{kesalahan} &= 0,5371 - 0,53125 = 0,00585
 \end{aligned}$$

Jika proses diakhiri sampai perkalian kedelapan,

$$\begin{aligned}
 0,10001001_{(2)} &= 0,5 + 0,03125 + 0,00390625 = 0,53515625 \\
 \text{kesalahan} &= 0,5371 - 0,53515625 = 0,00194375
 \end{aligned}$$

Melalui kombinasi dari bilangan positif di atas 1 dan bilangan positif di bawah 1 dapat dinyatakan bentuk bilangan positif seperti di bawah ini,

Contoh

$$323, 4375_{(10)} = ?_{(2)}$$

Konversi bilangan desimal $325_{(10)}$

$$\begin{array}{ll}
 325 : 2 &= 162 \text{ sisa } 1 \\
 162 : 2 &= 81 \text{ sisa } 0 \\
 81 : 2 &= 40 \text{ sisa } 1 \\
 40 : 2 &= 20 \text{ sisa } 0 \\
 20 : 2 &= 10 \text{ sisa } 0 \\
 10 : 2 &= 5 \text{ sisa } 0 \\
 5 : 2 &= 2 \text{ sisa } 1 \\
 2 : 2 &= 1 \text{ sisa } 0 \\
 1 : 2 &= 0 \text{ sisa } 1
 \end{array}$$

$$325_{(10)} = 101000101_{(2)}$$

Konversi bilangan desimal $0,4375_{(10)}$

$$\begin{array}{rcl} 0,4375 \cdot 2 & = & 0 \text{ sisa } 0,8750 \\ 0,8750 \cdot 2 & = & 1 \text{ sisa } 0,7500 \\ 0,7500 \cdot 2 & = & 1 \text{ sisa } 0,5000 \\ 0,5000 \cdot 2 & = & 1 \text{ sisa } 0 \end{array}$$

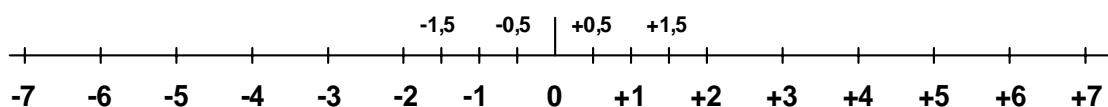
$$0,4375_{(10)} = 0,0111_{(2)}$$

$$\text{Jadi bilangan } 325,4375_{(10)} = 101000101,0111_{(2)}$$

$$\begin{aligned} \text{Test : } 101000101,0111_{(2)} &= 1.2^8 + 1.2^6 + 1.2^2 + 1.2^0 + 1.2^{-2} + 1.2^{-3} + 1.2^{-4} \\ &= 256 + 64 + 4 + 1 + 0,25 + 0,125 + 0,0625 \\ &= 325,4375_{(10)} \end{aligned}$$

1.1.8. Bentuk Bilangan Negatif

Dengan berpatokan pada titik 0 (nol), bilangan dapat dibedakan menjadi bilangan positif dan bilangan negatif. Disebut bilangan positif jika harga bilangan tsb. lebih besar dari nol (disebelah kanan titik nol) dan disebut bilangan negatif jika harga bilangan tsb. lebih kecil dari nol (disebelah kiri titik nol).



Bilangan +3 terletak pada 3 skala sebelah kanan setelah nol, sedangkan bilangan -3 terletak pada 3 skala sebelah kiri setelah nol. Jadi + dan - adalah suatu tanda dari bilangan. Secara prinsip tanda positif (+) dan tanda negatif (-) berlaku juga untuk bilangan biner. Pada mikroprosesor jumlah bit data sudah tertentu yaitu 8 bit, 16 bit atau 32 bit. Kita ambil contoh mikroprosesor famili intel 8080/8085, famili Zilog Z80 dan famili motorola 6809 mempunyai 8 bit data dan dalam bentuk biner dapat dituliskan sbb : $00000000_{(2)} = 0_{(10)}$ sampai $11111111_{(2)} = 255_{(10)}$, tanpa menghiraukan tanda positif dan negatif. Jika dalam 8 bit data kita menghiraukan tanda positif dan tanda negatif, maka daerah bilangan di atas dibagi menjadi dua bagian sehingga bilangan tersebut menjadi +127 dan -128. Untuk daerah positif bilangan dimulai dari $00000000_{(2)}$ dan $00000001_{(2)}$ sampai bilangan maksimum positif adalah $01111111_{(2)}$ sedangkan daerah negatif dimulai dari $11111111_{(2)}$ untuk $-1_{(10)}$ sampai $10000000_{(2)}$ untuk $-128_{(10)}$, tetapi range 8 bit data masih sama yaitu 255_{10} (dari +127 hingga -128).

Di bawah ini menunjukkan susunan 8 bit data dengan menghiraukan tanda (+) dan (-).

Desimal	Biner	
+127	01111111	
+126	01111110	
+125	01111101	
+124	01111100	
+123	01111011	
.....	
+ 7	00000111	
+ 6	00000110	
+ 5	00000101	
+ 4	00000100	
+ 3	00000011	
+ 2	00000010	
+ 1	00000001	
0	00000000	
- 1	11111111	
- 2	11111110	
- 3	11111101	
- 4	11111100	
- 5	11111011	
- 6	11111010	
- 7	11111001	
- 8	11111000	
.....	
- 124	10000100	
- 125	10000011	
- 126	10000010	
- 127	10000001	
- 128	10000000	

Daerah Positif Bilangan : 0 sampai ($2^{n-1}-1$)

Daerah Negatif Bilangan : -1 sampai -2^{n-1}

n = jumlah bit, dalam contoh di atas adalah 8

Pada susunan ini tempat tertinggi atau disebut Most Significant Bit (2^7), hanya digunakan sebagai Bit tanda. Untuk harga 0 pada bit 2^7 adalah tanda bilangan positif sedangkan harga 1 pada bit 2^7 merupakan tanda bilangan negatif.

1.1.9. Bentuk Bilangan Dalam Code Form

Mengkonversi bilangan yang berharga besar, memerlukan hitungan yang cukup melelahkan. Melalui bilangan dalam Code Form maka pekerjaan konversi bilangan dapat dipermudah dan dipercepat. Di bawah ini adalah Code Form dalam bilangan Desimal, Bilangan Oktal dan bilangan Heksadesimal yang sering dipergunakan.

1.1.9.1. Bentuk BCD - Biner Code Desimal

Bilangan desimal pada setiap tempat dapat terdiri dari 10 bilangan yang berbeda-beda. Untuk bilangan biner bentuk dari 10 elemen yang berbeda beda memerlukan 4 bit. Sebuah BCD mempunyai 4 bit biner untuk setiap tempat bilangan desimal.

Contoh

$$\begin{array}{ccc} Z_{(10)} = 317 \\ \begin{array}{ccc} \underline{3} & \underline{1} & \underline{7} \\ 0011 & 0001 & 0111 \end{array} \end{array} \quad \begin{array}{l} \text{Desimal} \\ \text{Biner Code Desimal} \end{array}$$

Dalam contoh ini BCD terdiri dari 3 kelompok bilangan masing-masing terdiri dari 4 bit , dan jika bilangan desimal tersebut di atas dikonversi ke dalam bilangan biner secara langsung adalah $317_{(10)} = 100111101_{(2)}$ dan hanya memerlukan 9 bit. Untuk contoh proses sebaliknya dapat dilihat di bawah ini.

Contoh

$$\begin{array}{cccc} \text{Biner Code Desimal} & \underline{0101} & \underline{0001} & \underline{0111} & \underline{0000} \\ \text{Desimal} & 5 & 1 & 7 & 0 \end{array}$$

Jadi bentuk BCD di atas adalah bilangan $Z_{(10)} = 5170$.

1.1.9.2. Bentuk BCO - Biner Code Oktal

Bilangan oktal pada setiap tempat terdiri dari 8 bilangan yang berbeda-beda. Untuk 8 elemen yang berbeda-beda diperlukan 3 bit. Sebuah BCO mempunyai 3 bit biner untuk setiap tempat bilangan oktal.

Contoh

$$\begin{array}{ccc} Z_{(8)} = 634 \\ \begin{array}{ccc} \underline{6} & \underline{3} & \underline{4} \\ 110 & 011 & 100 \end{array} \end{array} \quad \begin{array}{l} \text{Bilangan Oktal} \\ \text{Biner Code Oktal} \end{array}$$

Untuk proses sebaliknya adalah setiap 3 bit dikonversi ke dalam bilangan oktal.

Contoh

$$\begin{array}{cccc} \text{Biner Code Oktal} & \underline{101} & \underline{100} & \underline{000} & \underline{001} \\ \text{Bilangan Oktal} & 5 & 4 & 0 & 1 \end{array}$$

Jadi bentuk BCO diatas adalah bilangan $Z_{(8)} = 5401$.

1.1.9.3. Bentuk BCH - Biner Code Heksadesimal

Bilangan heksadesimal dalam setiap tempat dapat terdiri dari 16 bilangan yang berbeda-beda (angka dan huruf). Bentuk biner untuk 16 elemen memerlukan 4 bit. Sebuah BCH mempunyai 4 bit biner untuk setiap tempat bilangan heksadesimal.

Contoh

$$Z_{(16)} = 31AF$$

<i>Bilangan Heksadesimal</i>	3	1	A	F
<i>Biner Code Heksadesimal</i>	<u>0011</u>	<u>0001</u>	<u>1010</u>	<u>1111</u>

Untuk proses sebaliknya, setiap 4 bit dikonversi ke dalam bilangan heksadesimal.

Contoh

<i>Biner Code Heksadesimal</i>	<u>1010</u>	<u>0110</u>	<u>0001</u>	<u>1000</u>
<i>Bilangan Heksadesimal</i>	A	6	1	8

Jadi bentuk BCH diatas adalah bilangan $Z_{(16)} = A618$.

1.1.10. Metoda Balikan

Metoda yang kita gunakan bisa dibalik yaitu dimulai dari bilangan Heksadesimal dirubah kedalam bentuk BCH (group digit biner empat-empat). Buat group ulang ke bentuk BCO (group digit biner tiga-tiga) dari titik desimal untuk mengkonversikan ke dalam bilangan Oktal. Akhirnya bilangan Oktal dapat dikonversikan ke dalam bentuk bilangan desimal dengan metoda biasa dan dengan cara ini konversi basis bilangan dapat dipermudah.

Contoh 1

Tunjukkan bilangan Heksadesimal **4B2,1A6₁₆** ke bentuk bilangan Biner, Oktal dan Bilangan Desimal yang ekuivalen.

- Lakukanlah :
- Tulis ulang 4B2,1A6₁₆ dalam bentuk BCH
 - Groupkan ulang kedalam bentuk BCO dari titik Desimal
 - Tunjukkan ekuivalen Oktalnya setiap BCO
 - Akhirnya konversikan bilangan Oktal ke ekuivalen Desimal

Jika ke-4 langkah di atas dilakukan dengan benar akan menghasilkan,

- 0100 1011 0010 , 0001 1010 0110₂

b. 010 010 110 010 , 000 110 100 110₂

c. 2 2 6 2 , 0 6 4 6₈

d. 1202,103₁₀

Contoh 2

Selesaikan bilangan Heksadesimal **2E3,4D₁₆** ke bentuk bilangan Biner, Oktal dan

$$\begin{aligned} \mathbf{2E3,4D_{16}} &= 0010 \ 1110 \ 0011 , 0100 \ 1101_2 \\ &= 001 \ 011 \ 100 \ 011 , 010 \ 011 \ 010_2 \\ &= 1 \ 3 \ 4 \ 3 , 2 \ 3 \ 2_8 \\ &= 739,301_{10} \end{aligned}$$

1.1.11. ASCII Code - American Standard Code For Information Interchange

Dalam bidang mikrokomputer ASCII-Code mempunyai arti yang sangat khusus, yaitu untuk mengkodekan karakter (Huruf, Angka dan tanda baca yang lainnya). Code-code ini merupakan code standard yang dipakai oleh sebagian besar sistem mikrokomputer. Selain huruf, angka dan tanda baca yang lain ada 32 (mis ACK, NAK dsb.) merupakan kontrol untuk keperluan transportasi data. Di bawah ini adalah tabel 7 bit ASCII Code beserta beberapa penjelasan yang diperlukan.

Singkatan	Arti	Ket. dlm. Bhs Inggris
STX	Awal dari text	Start of Text
ETX	Akhir dari text	End of text
ACK	Laporan balik positif	Acknowledge
NAK	Laporan balik negatif	Negative Acknowledge
CAN	Tidak berlaku	Cancel
CR	Carriage Return	Carriage Return
FF	Form Feed	Form Feed
LF	Line Feed	Line Feed
SP	Jarak	Space
DEL	Hapus	Delete

Tabel ASCII Code

Bit	b7	b6	b5	b4	b3	b2	b1	H	0	0	0	0	1	1	1	1
								E	0	0	1	1	0	0	1	1
								X	0	1	0	1	0	1	0	1
									0	1	2	3	4	5	6	7
				0	0	0	0	0	NUL	DLE		0	@	P	`	p
				0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
				0	0	1	0	2	STX	DC2	"	2	B	R	b	r
				0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
				0	1	0	0	4	EOT	DC4	\$	4	D	T	d	t
				0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
				0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
				0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
				1	0	0	0	8	BS	CAN	(8	H	X	h	x
				1	0	0	1	9	HT	EM)	9	I	Y	I	y
				1	0	1	0	A	LF	SUB	*	:	J	Z	j	z
				1	0	1	1	B	VT	ESC	+	;	K	[k	{
				1	1	0	0	C	FF	FS	,	<	L	\	l	l
				1	1	0	1	D	CR	GS	-	=	M]	m	}
				1	1	1	0	E	SO	RS	.	>	N	^	n	~
				1	1	1	1	F	SI	US	/	?	O	_	o	DEL

Contoh

Untuk mendapatkan ASCII Code bagi karakter N adalah 100 1110 (4E₁₆) dengan penjelasan bahwa 100 adalah b7, b6 dan b5 yang lurus keatas terhadap huruf N dan dan berharga 4 sedangkan 1110 adalah b4, b3, b2 dan b1 yang lurus kesamping kiri terhadap huruf N dan berharga E.

LATIHAN

- 1** a. Bilangan biner adalah bilangan yang berbasis
- b. Bilangan heksadesimal adalah bilangan yang berbasis

a. dua b. enam belas

- 2** Konversikan bilangan desimal di bawah ini ke dalam bilangan biner
- a. 1234_{10} b. 5670_{10} c. 2321_{10}

a. 10011010010 b. 1011000100110 c. 100100010001

- 3** Konversikan bilangan biner di bawah ini ke dalam bilangan desimal
- a. 10101010 b. 01010101 c. 11001100 d. 10011111

a. 170 b. 85 c. 204 d. 159

- 4** Konversikan bilangan biner di bawah ini ke dalam bilangan oktal
- a. $1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 1_2$ b. $1\ 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 1_2$

a. 5371_8 b. 6267_8

- 5** Konversikan bilangan oktal di bawah ini ke dalam bilangan biner
- a. 2170_8 b. 3571_8

a. 0 1 0 0 0 1 1 1 1 0 0 0 b. 0 1 1 1 0 1 1 1 1 0 0 1

- 6** Konversikan bilangan desimal di bawah ini ke dalam bilangan heksadesimal
- a. 1780_{10} b. 3666_{10} c. 5230_{10} d. 6744_{10}

a. 06F4 b. 0E52 c. 146E d. 1A58

7 Konversikan bilangan heksadesimal di bawah ini ke dalam bilangan desimal

- a. $ABCD_{16}$ b. 2170_{16} c. $B75F_{16}$ d. $EBED_{16}$

- a. 43981 b. 8560 c. 46943 d. 60397

8 Konversikan bilangan pecahan desimal di bawah ini ke dalam bilangan biner

- a. $0,3125_{10}$ b. $0,65625_{10}$ c. $0,34375_{10}$ d. $0,140625_{10}$

- a. 0,0101 b. 0,10101 c. 0,01011 d. 0,001001

9 Konversikan bilangan desimal di bawah ini ke dalam bilangan biner

- a. $11,625_{10}$ b. $0,6875_{10}$ c. $0,75_{10}$ d. $25,75_{10}$

- a. 1011,101 b. 0,1011 d. 11001, 11

10 Konversikan bilangan desimal di bawah ini ke dalam bilangan heksadesimal

- a. $348,654_{10}$ b. $1784,240_{10}$

- a. 15C,A78 b. 6F8,3D5

11 Konversikan bilangan di bawah ini ke dalam bilangan desimal

- a. $010100011,001111101_2$ b. $654,276_8$ c. $4C5,2B8_{16}$

- a. 163,245 b. 428,371 c. 1221,1699

12 Rubahlah bilangan biner di bawah ini ke dalam bentuk BCD

- a. 10100110000111_2 b. 1010101100011_2

- a. 2987 b. 1563

13 Rubahlah bentuk BCD di bawah ini ke dalam bilangan biner

- a. 1987 b. 2346 c. 501

a. 1 1001 1000 0111 b. 10 0011 0100 0110 c. 101 0000 0001

14 Rubahlah bilangan biner di bawah ini ke dalam BCO

- a. 11111101001_2 b. $101110\ 010100_2$ c. 1100000010_2

a. 3751 b. 5624 c. 1402

15 Rubahlah bilangan biner di bawah ini ke dalam BCH

- a. 1101111100101110_2 b. 110100110000001_2

a. CF2E b. 6981

16 Rubahlah Bentuk BCH di bawah ini ke dalam bilangan heksadesimal

- a. F0DE b. 1CAB c. 834

a. 1111 0000 1101 1110 b. 1 1100 1010 1011 c. 1000 0011 0100

17 Nyatakan positif atau negatif bilangan biner di bawah ini

- a. 01111111 b. 10000000 c. 01111011

a. Positif 127 b. Negatif 128 c. Positif 123

18 Nyatakan bilangan biner negatif di bawah ini ke dalam bilangan desimal

- a. 10001000 b. 11110111 c. 10000101 d. 10011100

a. -120 b. -9 c. -123 d. -100

19 Nyatakan ASCII Code di bawah ini dalam bentuk karakter

- a. 41_{16} b. $5A_{16}$ c. 24_{16} d. 77_{16}

a. A b. Z c. \$ d. W

20 Nyatakan Karakter di bawah ini dalam ASCII Code

- a. a b. x c. m d. H

a. 61_{16} b. 78_{16} c. $6D_{16}$ d. 57_{16}

21 Dengan Keyboard standard ASCII, pada layar monitor nampak tulisan sebagai berikut

PRINT X

Nyatakan Keluaran pada Keyboard tersebut.

P (101 0000); R (101 0010); I (100 1001); N (100 1110)
T (101 0100); space (010 0000); X (101 1000)