

7

PENCACAH

Rangkaian pencacah (counter) merupakan rangkaian yang sederhana dan sangat umum pemakaiannya dalam sistem-sistem digital, baik dalam sistem yang kecil-kecil maupun dalam sistem besar seperti prosesor untuk komputer.

Setiap flip-flop dalam rangkaian ini berubah keadaan menurut aturan/urutan yang ditentukan. Perubahan keadaan itu terjadi serentak (sebenarnya dengan sedikit tundaan waktu) dengan munculnya pulsa penabuh (clock pulse). Jadi rangkaian ini adalah rangkaian berurut serempak.

Harga maksimum yang dapat dicapai oleh satu pencacah ditentukan oleh cacah flip-flop yang membentuk rangkaiannya. Untuk suatu pencacah dengan 3 flip-flop (disebut juga pencacah 3 bit), harga maksimum yang dapat dicapai adalah 7 yang terjadi saat keseluruhan flip-flop itu berkeadaan 1; untuk pencacah dengan 4 flip-flop harga maksimum yang dapat dicapai adalah 15 (1111 biner), dan seterusnya. Dalam pencacah biasa, setelah mencapai harga cacahan maksimum, keadaan pencacah akan kembali kepada keadaan mula, misalnya 0. Tetapi untuk keperluan-keperluan khusus sering dibutuhkan pencacah yang mencacah tidak secara berurutan dari 0, 1, 2, ... dan seterusnya atau turun dari suatu harga maksimum sampai ... 3, 2, 1, 0, melainkan melompat-lompat menurut urutan yang dibutuhkan. Dalam bab ini diuraikan perencanaan rangkaian pencacah, baik yang berurutan maupun yang acak.

7.1 Pencacah Berurutan dan tak berurutan

Urutan pencacahan pada pencacah biner berurutan mulai dari 0 sampai harga maksimum untuk pencacah naik atau mulai dari harga maksimum turun sampai 0 pada pencacah turun. Harga desimal pencacahan untuk pencacah naik biner yang terdiri atas 4 flip-flop adalah 0 s/d 15, yaitu: 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15, kembali ke 0,1,2,.. dst sedangkan untuk pencacah turun adalah 15 - 0, yaitu: 15,14,13,12,11,10,9,8,7,6,5,4,3,2,1,0, kembali ke 15,14,.. dst. Kalau masing-masing flip-flop dalam pencacah tersebut dinamakan A, B, C, dan D, maka keadaan-sekarang dan keadaan-berikutnya (setelah adanya pulsa penabuh) pencacah tersebut dapat diurutkan seperti ditunjukkan pada Tabel 7.1(a) dan (b).

Dalam tabel keadaan-berikut untuk pencacah turun dalam Tabel 7.1, keadaan-sekarang (ABCD) diurutkan naik sebagaimana lazimnya tabel kebe-

nan. Perhatikan bahwa harga desimal daripada keadaan-berikut ($A^+B^+C^+D^+$) pencacah turun ini tetap lebih kecil 1 dari harga desimal keadaan-sekarang, kecuali untuk keadaan-sekarang 0000 yang keadaan-berikutnya 1111. Urutan pencacahan menurun mungkin akan lebih jelas dilihat bila penulisan keadaan-sekarang diurutkan menurun mulai dari 1111. Tetapi pengurutan seperti pada Tabel 7.1 lebih enak dipakai dalam pembuatan peta Karnaugh.

Tabel 7.1. Tabel keadaan pencacah biner berurutan.

A	B	C	D	A ⁺	B ⁺	C ⁺	D ⁺	A	B	C	D	A ⁺	B ⁺	C ⁺	D ⁺
0	0	0	0	0	0	0	1	0	0	0	0	1	1	1	1
0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0
0	0	1	0	0	0	1	1	0	0	1	0	0	0	0	1
0	0	1	1	0	1	0	0	0	0	1	1	0	0	1	0
0	1	0	0	0	1	0	1	0	1	0	0	0	0	1	1
0	1	0	1	0	1	1	0	0	1	0	1	0	1	0	0
0	1	1	0	0	1	1	1	0	1	1	0	0	1	0	1
0	1	1	1	1	0	0	0	0	1	1	1	0	1	1	0
1	0	0	0	1	0	0	1	1	0	0	0	0	1	1	1
1	0	0	1	1	0	1	0	1	0	0	1	1	0	0	0
1	0	1	0	1	0	1	1	1	0	1	0	1	0	0	1
1	0	1	1	1	1	0	0	1	0	1	1	1	0	1	0
1	1	0	0	1	1	0	1	1	1	0	0	1	0	1	1
1	1	0	1	1	1	1	0	1	1	0	1	1	1	0	0
1	1	1	0	1	1	1	1	1	1	1	0	1	1	0	1
1	1	1	1	0	0	0	0	1	1	1	1	1	1	1	0

(a)
Pencacah naik

(b)
Pencacah turun

Di samping pencacah berurutan biner di atas, jenis pencacah berurutan lainnya yang juga sering dipakai adalah pencacah desimal. Perbedaan antara pencacah biner dengan pencacah desimal hanyalah pada harga maksimum pencacahan yang dapat dicapai. Kalau dalam pencacah biner pencacahan kembali ke 0000 setelah mencapai 1111, maka pada pencacah desimal pencacahan kembali ke keadaan 0000 setelah mencapai 1001 (= 9 dalam desimal) sebab di dalam sistem bilangan desimal tidak ada angka yang lebih besar daripada 9. Pencacah seperti ini disebut juga *Pencacah Modus 9*. Jadi untuk pencacah desimal, 6 keadaan-sekarang yang paling bawah dalam Tabel 7.1 tidak ditentukan sehingga dalam implementasinya dapat dianggap di"abaikan". Hal serupa berlaku untuk pencacah-pencacah dengan modus lainnya, misalnya

modus 7, 5 dan lain-lainnya yang kembali ke keadaan 0000 setelah mencapai 7, 5 dan sebagainya.

Seperti telah disebutkan di depan, sering dibutuhkan pencacah yang mencah secara tak-berurutan (acak) tetapi mempunyai pola pencacahan tertentu. Ini misalnya diperlukan dalam penentuan waktu penabuhan komponen-komponen dalam suatu komputer (timing control). Keluaran rangkaian pencacah ini misalnya dipakai untuk menentukan urutan pengaktifan register-register dalam unit pengolah pusat (Central Processing Unit, CPU) dalam komputer. Sebagai contoh, urutan cacahan dapat dibuat 000, 010, 111, 100, 000,... dan seterusnya. Baik pencacah berurutan maupun tak berurutan dapat direalisasikan dengan semua jenis flip-flop yang telah diuraikan pada bab sebelumnya.

Berikut ini akan diuraikan perencanaan rangkaian pencacah berturut-turut dengan memakai flip-flop T, RS, JK, dan D.

7.2 Pencacah Biner memakai Flip-flop T

Dari karakteristik flip-flop T yang telah diuraikan dalam bab sebelumnya, dapat disusun tabel masukannya sebagai berikut ini:

Tabel Keadaan:			Tabel masukan:			T Pers. Masukan: $T = Q\bar{Q} \oplus Q^+$
T	Q	Q^+	Q	Q^+		
0	0	0	0	0		
0	1	1	0	1		
1	0	1	1	0		
1	1	0	1	0		

Dari tabel keadaan yang juga disertakan dalam tabel di atas dapat dilihat bahwa keluaran flip-flop T akan berubah ke komplementennya hanya bila masukan $T = 1$ dan keluaran flip-flop itu tetap bila $T = 0$ (lihat juga pers. masukan). Dengan kenyataan ini maka rangkaian pencacah dengan flip-flop T dapat ditentukan dengan mudah.

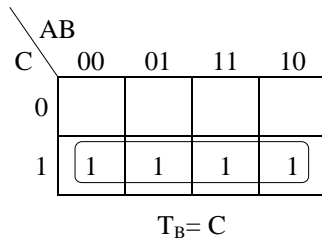
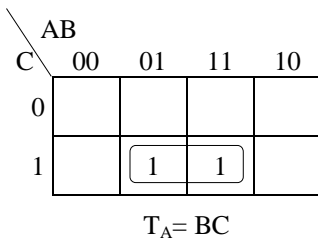
Misalkan kita hendak menyusun suatu pencacah biner 3 bit dengan memakai flip-flop T. Berdasarkan tabel masukan di atas dan dengan memperhatikan hanya 3 bit paling kanan tabel keadaan pencacah biner yang ditunjukkan pada Tabel 7.1 di depan, maka tabel keadaan dan masukan rangkaian pencacah yang diinginkan dapat disusun seperti ditunjukkan pada Tabel 7.2(a), dengan A, B, C = Keadaan-sekarang; A^+, B^+, C^+ = keadaan-berikut, dan T_A, T_B, T_C = masukan.

Sebagai contoh, untuk keadaan-sekarang 101 dan keadaan-berikut 110, keadaan flip-flop A tetap (tidak berubah), keadaan flip-flop B berubah dari 0 ke 1, dan C berubah dari 1 ke 0 sehingga masukan yang diperlukan adalah $T_A=0$, $T_B=T_C=1$.

Tabel 7.2. Tabel keadaan dan masukan pencacah biner dengan flip-flop T.

(a) Pencacah Naik				(b) Pencacah Turun									
A	B	C	A ⁺ B ⁺ C ⁺	T _A	T _B	T _C	A	B	C	A ⁺ B ⁺ C ⁺	T _A	T _B	T _C
0	0	0		0	0	1	0	0	1	0	0	1	
1			1	1			1						
0	0	1		0	1	0	0	1	1	0	0	0	
0			0	1									
0	1	0		0	1	1	0	0	1	0	0	0	
1			0	1			1						
0	1	1		1	0	0	1	1	1	0	1	0	
0			0	1									
<hr/>				<hr/>				<hr/>					
1	0	0		1	0	1	0	0	1	1	0	0	
1			1	1			1						
1	0	1		1	1	0	0	1	1	0	1	0	
1	1	0		1	1	1	0	0	1	1	0	1	
1			0				1	1					
1	1	1		0	0	0	1	1	1	1	1	0	
0			0	1									

Untuk memperoleh persamaan masukan masing-masing flip-flop dibuat peta Karnaugh masing-masing flip-flop, seperti ditunjukkan dalam Gambar 7.1. Dalam Gambar 7.1 tidak digambarkan peta masukan flip-flop C sebab seperti dapat dilihat dari tabel keadaan dan masukan pada Tabel 7.2(a) di atas, T_C tetap 1.



Gambar 7.1. Peta keadaan-berikut pencacah biner naik dengan flip-flop T

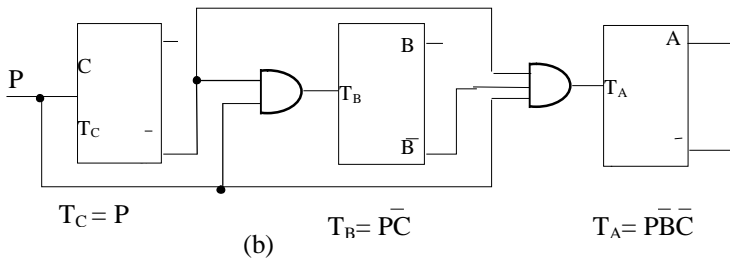
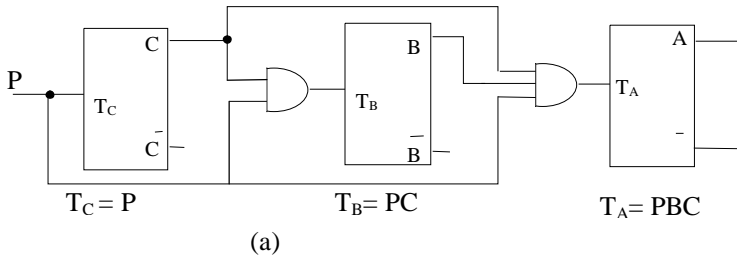
Untuk menyerempakkan perubahan keadaan semua flip-flop maka masukan yang ditunjukkan pada peta Karnaugh di atas di-AND-kan dengan sinyal penambah P sehingga diperoleh persamaan masukan :

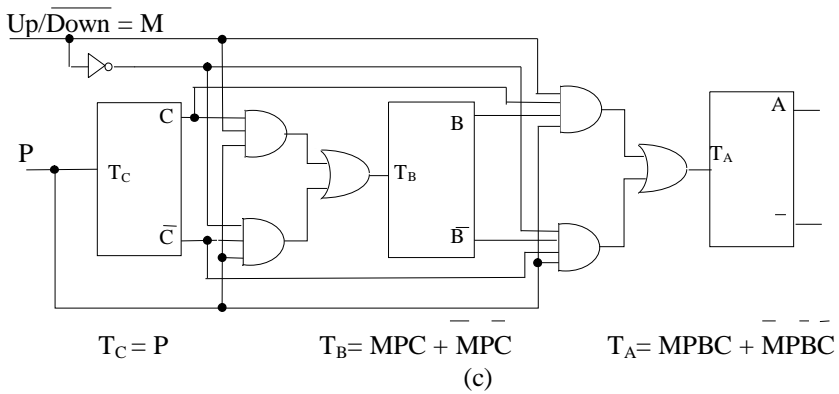
$$T_A = PBC \qquad T_B = PC \qquad T_C = P$$

Untuk pencacah turun, dengan cara yang sama, dari Tabel 7.2 (b) dapat diperoleh persamaan masukan sebagai berikut:

$$T_A = \bar{P}\bar{B}\bar{C} \qquad T_B = \bar{P}\bar{C} \qquad T_C = P$$

Rangkaian pencacah biner naik dan turun yang diperoleh di atas ditunjukkan pada Gambar 7.2. Perhatikan bahwa untuk mengubah pencacah naik menjadi pencacah turun, atau sebaliknya, dibutuhkan hanya sedikit perubahan. Bandingkan persamaan masukan kedua jenis pencacah di atas. Jadi yang dibutuhkan hanyalah sejenis Multiplexer yang memilih masukan bagi flip-flop A dan B. Dalam Gambar 7.2 (c) multipelxer ini direalisasikan dengan 2 AND dan 1 OR pada masukan kedua flip-flop A dan B. Sinyal M= Up/Down yang berfungsi sebagai pemilih modus, naik ($\bar{U}p$) bila M= 1 dan turun (Down) bila M= 0.





Gambar 7.2. Rangkaian pencacah biner dengan flip-flop T

(a) Pencacah naik, (b) Pencacah turun, (c) Pencacah naik/turun.

Perhatikan bahwa bila $M=1$ hanya keluaran AND sebelah atas yang berpengaruh dan bila $M=0$ hanya keluaran AND sebelah bawah yang berpengaruh.

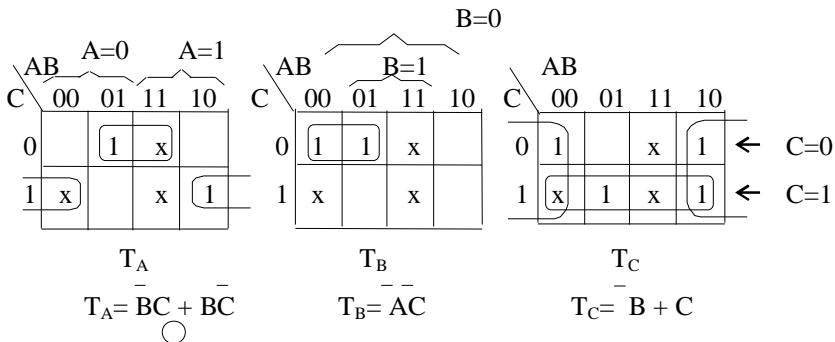
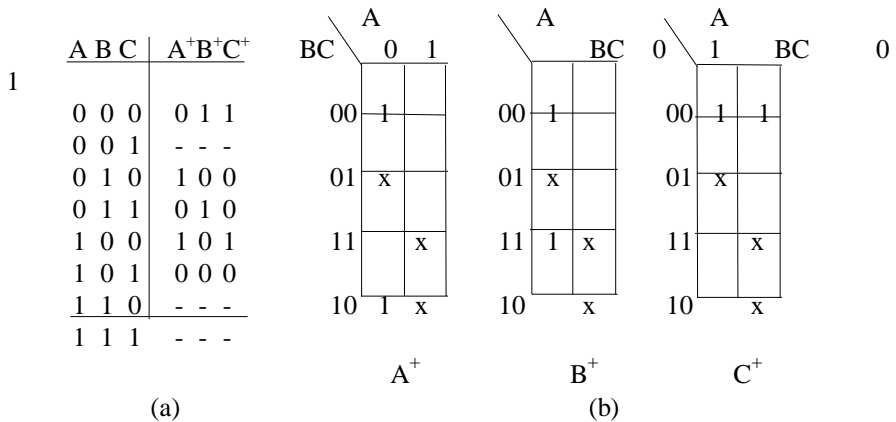
7.3 Pencacah Tak Berurutan Dengan Flip-flop T

Pada dasarnya, desain pencacah dengan urutan acak sama saja dengan desain pencacah berurutan seperti yang telah diuraikan di depan. Langkah pertama adalah penyusunan tabel keadaan dan masukan, disusul dengan pemetaan masukan dan minimisasi fungsi masukan tersebut. Untuk melihat pelaksanaannya, marilah kita rancang suatu pencacah dengan urutan pencacahan 000, 011, 010, 100, 101 kembali lagi ke 000, dan seterusnya. Pada Gambar 7.3 (a) ditunjukkan tabel keadaan berikut untuk pencacah ini. Sebagaimana dilakukan pada perencanaan pencacah sebelumnya, tabel dan peta masukan masing-masing flip-flop T dapat diturunkan dari tabel keadaan-berikut ini. Tetapi pada Gambar 7.3 ditunjukkan cara lain dengan pertama-tama menggambarkan peta keadaan-berikut, Gambar 7.3 (b), dan dari peta ini diturunkan peta masukan untuk masing-masing flip-flop. Dengan cara seperti ini, penurunan persamaan masukan untuk sembarang flip-flop menjadi lebih sistematis.

Pada Gambar 7.3, baik peta keadaan-berikut maupun peta masukan digambarkan sebagai fungsi keadaan-sekarang. Peta keadaan-berikut semata-mata pemetaan tabel keadaan-berikut. Sebagai contoh, untuk keadaan-sekarang 010 ($A=0, B=1, C=0$), keadaan-berikut adalah 100 dan pada peta A^+, B^+ , dan C^+

untuk $ABC = 010$ berturut-turut diisikan 1, 0 dan 0. Peta masukan diisi berdasarkan karakteristik flip-flop T, yaitu $T=1$ bila $Q^+ = \bar{Q}$ dan $T=0$ bila $Q^+ = Q$. Untuk keadaan-sekarang 011 , kotak untuk keadaan ini pada peta A^+ berisi 0. Ini berarti bahwa $A=0$ dan $A^+=0$ sehingga pada peta T_A untuk 011 diisikan 0. Pada kotak yang sama pada peta C^+ berisi 0. Ini berarti bahwa $C=1$ berubah menjadi $C^+ = 0$ sehingga pada peta T_C diisikan 1. Dengan cara serupa semua kotak dapat diisi. Untuk keadaan-berikut yang tak dirinci pada tabel keadaan (diberi tanda "-") seperti untuk keadaan-sekarang 001 , semua kotak bersangkutan diisi tanda x (abaikan).

Pengisian peta masukan diatas dapat dipermudah dengan memakai sifat flip-flop T: $T = Q^+Q + Q\bar{Q}^+ = Q + \bar{Q}^+$ yang berarti bahwa $T=Q^+$ untuk $Q=0$ dan $T=\bar{Q}^+$ untuk $Q=1$. Jadi, kotak-kotak dengan $Q=0$ pada peta T dapat diisi dengan harga yang ada pada kotak yang sama pada peta Q^+ dan untuk kotak-kotak dengan $Q=1$, kotak pada peta T diisi dengan komplemen dari isi peta Q^+ (Q berarti A, B, atau C).



$$= B + C$$

(c)

Gambar 7.3. Perencanaan pencacah tak berurutan. (a) Tabel keadaan-berikut (b) Peta keadaan-berikut (c) Peta masukan

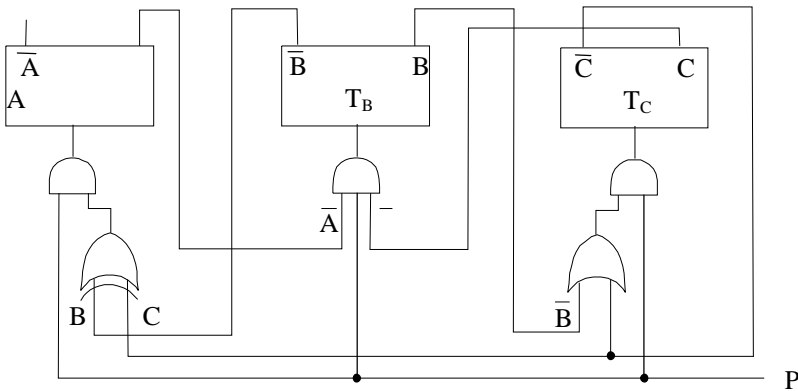
Setelah peta masukan untuk semua flip-flop lengkap diisi, maka segera dapat disederhanakan untuk memperoleh persamaan masukan yang minimum. Seperti yang ditunjukkan juga pada Gambar 7.3 (c), persamaan masukan untuk masing-masing flip-flop adalah :

$$T_A = \bar{B}C + B\bar{C} = B \oplus C$$

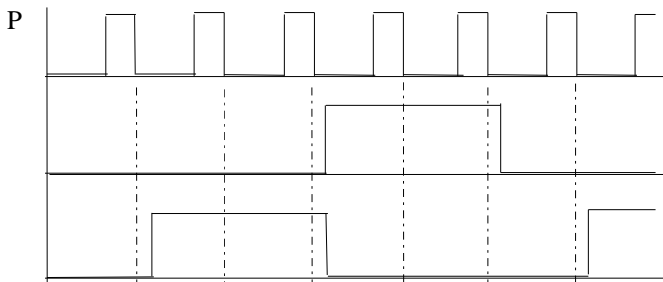
$$T_B = \bar{A}\bar{C}$$

$$T_C = \bar{B} + C$$

Dengan meng-AND-kan masing-masing masukan ini dengan sinyal penabuh P akan diperoleh diagram rangkaian Gambar 7.4.



Gambar 7.4. Rangkaian pencacah dengan flip-flop T dengan urutan seperti pada Gambar 7.3.



A	0	0	0	1	1	0
B	0		1	1	0	0
C	0	1	0	0	1	0
T_A						
T_B						
T_C						

Gambar 7.5. Diagram waktu pencacah Gambar 7.4.

Untuk meyakinkan bahwa rangkaian akan bekerja sebagaimana yang diinginkan, rancangan itu perlu dianalisis dengan melacak perubahan keadaannya dengan memberikan sederetan pulsa penabuh sebanyak yang dibutuhkan untuk membuat rangkaian menjalani 1 siklus cacahannya. Pelacakan perubahan keadaan itu dapat dilakukan dengan menggambarkan diagram waktu rangkaian seperti yang ditunjukkan pada Gambar 7.5.

Dalam gambar ini juga disertakan keadaan sinyal masukan T masing-masing flip-flop. Dari sini dapat dilihat bahwa rangkaian melakukan urutan pencacahan seperti yang diinginkan.

7.4 Pencacah dengan flip-flop RS

Perbedaan perancangan pencacah memakai flip-flop T dengan memakai flip-flop jenis lain terletak pada kombinasi masukan flip-flop itu untuk menghasilkan keadaan yang diinginkan, jadi pada persamaan masukan flip-flop. Untuk merancang pencacah pada sub-bab 7.3 dengan menggunakan flip-flop RS, tabel kebenaran masukan setiap flip-flop dapat disusun seperti ditunjukkan pada Gambar 7.6 (a). Keadaan R dan S dalam tabel ini diperoleh dengan memakai tabel kebenaran masukan pada Gambar 7.6 (b) yang diperoleh dari tabel pada Gambar 6-4 di depan. Perlu dicatat bahwa untuk keadaan yang tetap 0, haruslah

S=0 tetapi R boleh 1 atau 0; untuk $Q = Q^+ = 1$, haruslah $R = 0$ dan S boleh 0 atau 1. Peta Karnaugh untuk masing-masing masukan pada Gambar 7.6(a) ditunjukkan pada Gambar 7.6(c). Dari peta ini dapat diperoleh persamaan masukan masing-masing flip-flop, yang juga ditunjukkan di bawah peta masing-masing, yaitu:

$$\begin{aligned}
 S_A &= \bar{B}\bar{C} & R_A &= C \\
 S_B &= \bar{A}\bar{B} & R_B &= \bar{B}\bar{C} \\
 S_C &= \bar{B}\bar{C} & R_C &= C
 \end{aligned}$$

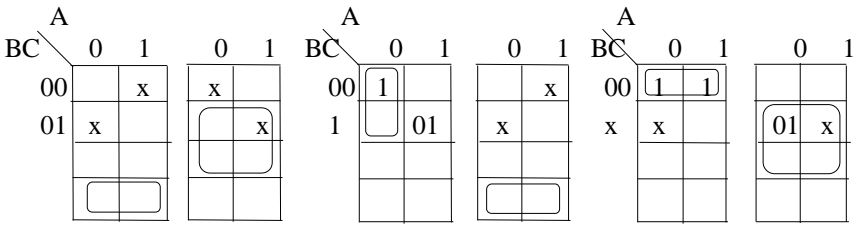
Dengan persamaan masukan ini, diagram rangkaian pencacah dapat dibuat seperti pada Gambar 7.7.

Biasanya peta masukan tersebut dapat diperoleh lebih mudah dengan menggunakan peta keadaan-berikut seperti pada Gambar 7.6(d), tanpa membuat tabel masukan semua flip-flop. Bila pada peta keadaan-berikut kotak-kotak untuk $Q = 0$ berisi 1 maka pada peta masukan diisikan $S_Q = 1, R_Q = 0$, dengan Q di sini mewakili A, B, dan C [lihat tabel pada Gambar 7.6 (b)]. Bilamana kotak-kotak untuk $Q = 0$ pada peta Q^+ berisi 0, maka pada peta masukan diisikan $S_Q = 0, R_Q = x$. Bila kotak-kotak untuk $Q = 1$ pada peta Q^+ berisi 0, maka pada peta masukan diisikan $S_Q = 1, R_Q = 0$, sedangkan bila kotak itu berisi 1 maka pada peta masukan diisikan $S_Q = x$ dan $R_Q = 0$.

A	B	C	A ⁺	B ⁺	C ⁺	S _A	R _A	S _B	R _B	S _C	R _C	Q	Q ⁺	S	R
0	0	0	0	1	1	0	x	1	0	1	0	0	0	0	x
0	0	1	-	-	-	x	x	x	x	x	x	0	1	1	0
0	1	0	1	0	0	1	0	0	1	0	x	1	0	0	1
0	1	1	0	1	0	0	x	x	0	0	1	1	1	x	0
1	0	0	1	0	1	x	0	0	x	1	0				
1	0	1	0	0	0	0	1	0	x	0	1				
1	1	0	-	-	-	x	x	x	x	x	x				
1	1	1	-	-	-	x	x	x	x	x	x				

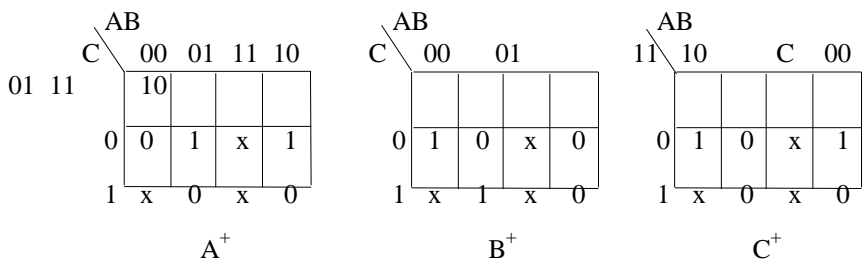
(b)

(a)



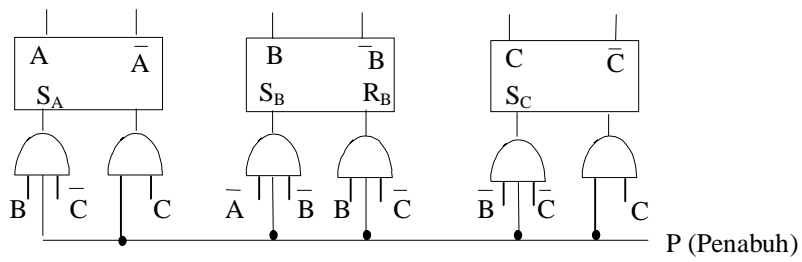
11	x	x	x	11	x	x	x	11	x	1	x
10	1	x	x	10	x	1	x	10	x	x	x
S_A		R_A		S_B		R_B		S_C		R_C	
$S_A = BC$		$R_A = C$		$S_B = AB$		$R_B = BC$		$S_C = BC$		$R_C = C$	

(c)



(d)

Gambar 7.6. Tabel dan Peta Karnaugh rancangan pencacah dengan flip-flop RS



Gambar 7.7. Rangkaian pencacah dengan flip-flop RS.

7.5 Pencacah Dengan Flip-flop JK

Seperti disebutkan pada bab sebelumnya, flip-flop JK sangat mirip dengan flip-flop RS, bedanya hanyalah pada kombinasi 11 untuk RS dan JK. Kalau pada flip-flop RS, $RS = 11$ tidak diperkenankan, maka pada flip-flop JK kombinasi

ini tidak dilarang, tetapi mengubah keluaran dari 0 menjadi 1 atau dari 1 menjadi 0.

Dari tabel keadaan-berikut untuk flip-flop JK yang diberikan pada Gambar 6.7 (c) di depan dapat diperoleh tabel kebenaran masukan untuk flip-flop JK sebagai fungsi keadaan-sekarang (Q) dan keadaan-berikut (Q^+) yang ditunjukkan pada Gambar 7.8 (a). Berdasarkan tabel ini, maka tabel kebenaran untuk semua flip-flop untuk pencacah yang diuraikan pada bagian 7.4, dapat ditentukan seperti yang ditunjukkan pada Gambar 7.8(b).

Seperti yang dilakukan pada perencanaan dengan flip-flop RS, maka peta masukan dapat diisikan baik dari tabel masukan maupun langsung dari peta keadaan yang ditunjukkan pada Gambar 7.8(c). Peta masukan tersebut ditunjukkan pada Gambar 7.8(d). Dengan persamaan masukan yang diberikan dibawah peta masukan, maka rangkaian pencacah yang dicari dapat digambarkan seperti pada Gambar 7.8 (e).

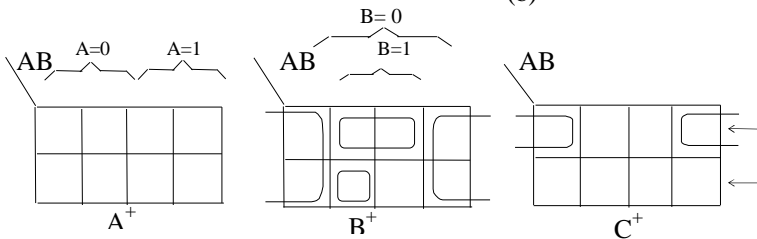
Tetapi untuk flip-flop JK, persamaan masukan dapat diperoleh langsung dari peta keadaan-berikut tanpa melalui peta masukan. Perhatikan tabel kebenaran pada Gambar 7.8(a). Kelihatan bahwa untuk $Q= 1$ harga J adalah abaikan (x) dan untuk $Q= 0$, harga K adalah abaikan. Jadi setiap 1 pada peta untuk $Q= 0$ dapat bergabung dengan x pada bagian $Q=1$ dan setiap 1 pada bagian peta K untuk $Q= 1$ dapat bergabung dengan x pada bagian $Q=0$. Ini berarti bahwa baik J maupun K bukanlah fungsi Q, artinya J_A dan K_A bukan fungsi A, J_B dan K_B bukan fungsi B, dan seterusnya. Juga dari tabel Gambar 7.8(a) tersebut dapat dilihat bahwa dengan mengambil keuntungan dari sifat suku abaikan x, untuk $Q=0$ maka $J= Q^+$, dan untuk $Q=1$ maka $K= Q^+$ atau $K= \bar{Q}^+$. Karena itu, persamaan J dapat dilihat langsung dari peta keadaan-berikut bagian $Q=0$ dan K dapat diperoleh langsung dari bagian $Q= 1$ peta tersebut. J diambil dari semua 1 pada bagian $Q= 0$ sedang-

Q	Q^+	J	K
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

(a)

A	B	C	A^+	B^+	C^+	J_A	K_A	J_B	K_B	J_C	K_C
0	0	0	0	1	1	0	x	1	x	1	x
0	0	1	-	-	-	x	x	x	x	x	x
0	1	0	1	0	0	1	x	x	1	0	x
0	1	1	0	1	0	0	x	x	0	x	1
1	0	0	1	0	1	x	0	0	x	1	x
1	0	1	0	0	0	x	1	0	x	x	1
1	1	0	-	-	-	x	x	x	x	x	x
1	1	1	-	-	-	x	x	x	x	x	x

(b)

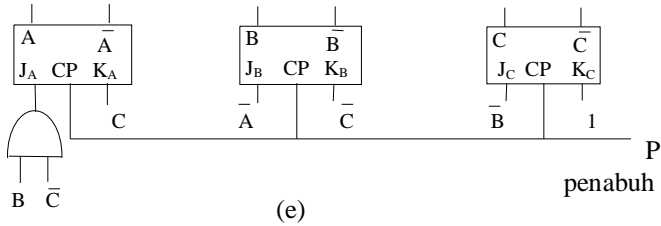


(c)

C	00	01	11	10	C	00	01	11	10	C	00	01	11	10	
0	0	1	x	1	0	1	0	x	0	0	0	0	x	1	C=0
1	x	0	x	0	1	x		1	0	0	1	x	0	x	0

C=1

A				A				A							
BC	0	1	0	1	BC		0	1		0	1	BC		0	1
00	0	x	x	0	00	1	0	x	x	00	1	1	x	x	
01	x	x	x	x	01	x	0	x	x	01	x	x	x	1	
11	0	x	x	1	11	x	x	0	x	11	0	x	1	x	
10	1	x	x	x	10	x	x	1	x	10	0	x	x	x	



Gambar 7.8. Pencacah dengan flip-flop JK.

kan K diambil dari semua 0 pada bagian Q= 1, dan menggabungkannya dengan suku abaikan x, kalau mungkin. Yang mudahnya untuk K, pandanglah semua 0 pada bagian Q=1 sebagai 1 dan tentukan gabungan sukuminnya.

Dalam Gambar 7.8 (c) ditunjukkan penggabungan untuk memperoleh J dan K untuk masing-masing flip-flop yang menghasilkan persamaan yang tepat sama dengan yang diperoleh sebelumnya. Perhatikan bahwa variabel A tidak muncul dalam persamaan J_A dan K_A , B tidak muncul dalam persamaan J_B dan K_B , C tidak muncul dalam persamaan J_C dan K_C . Karena itu, walaupun J_A yang dihasilkan oleh keadaan 1 pada kotak **010** pada peta A^+ mestinya ABC, hasil akhir menjadi hanya BC. Penggabungan 0 pada sukumin 101 dan x pada 111

untuk mencari K_A , yang seharusnya menghasilkan $K_A = AC$ menjadi tinggal $K_A = C$.

Cara penentuan persamaan J dan K yang diterangkan di atas berlaku secara umum, bukan hanya untuk perencanaan pencacah di atas saja.

7.6 Pencacah dengan Flip-flop D

Karena untuk flip-flop D, $Q^+ = D$, maka peta masukan untuk flip-flop D adalah peta keadaan-berikut itu sendiri dan karena itu persamaan masukan dapat diperoleh langsung dari peta keadaan-berikut flip-flop itu. Untuk pencacah yang direncanakan di bagian depan, peta keadaan-berikut pada Gambar 7.3(b) digambarkan kembali pada Gambar 7.9(a).

Penyederhanaan tabel keadaan-berikut ini menghasilkan persamaan masukan untuk realisasi dengan flip-flop D sebagai berikut:

$$D_A = A^+ = \bar{A}C + \bar{B}C = (A + B)\bar{C}$$

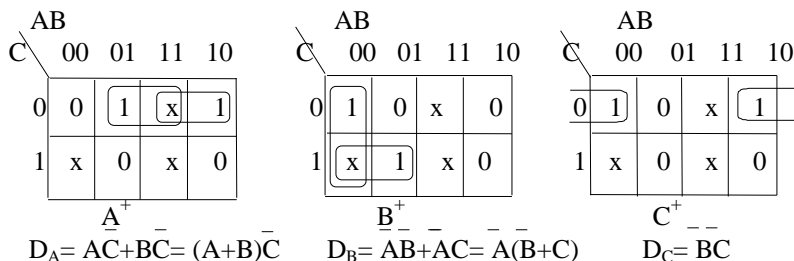
$$D_B = B^+ = \bar{A}B + \bar{A}C = \bar{A}(B + C)$$

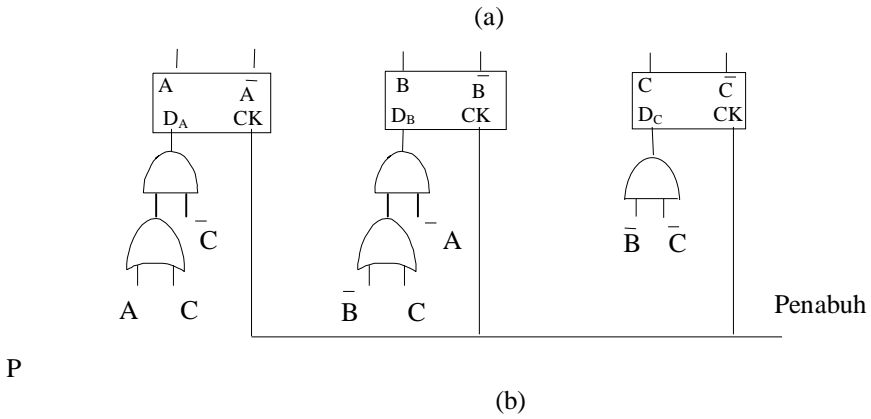
$$D_C = C^+ = \bar{B}C$$

Persamaan masukan ini memberikan rangkaian pencacah yang ditunjukkan pada Gambar 7.9(b)

7.7 Pencacah dalam Rangkaian Terpadu

Pencacah merupakan rangkaian berurut yang sangat banyak penggunaannya, baik dalam sistem yang kecil-kecil maupun dalam sistem besar seperti prosesor untuk komputer. Karena banyaknya kebutuhan pencacah ini, maka pencacah juga dibuat dalam bentuk Rangkaian Terpadu (Integrated Circuit, IC), misalnya tipe : '90, '92, '93, '196, '197 dan sebagainya [misalnya 74L90, 74393, atau 74LS90].

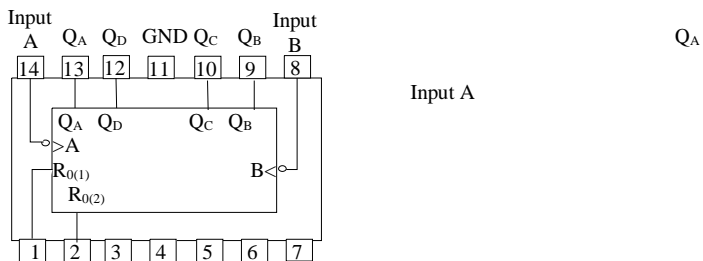




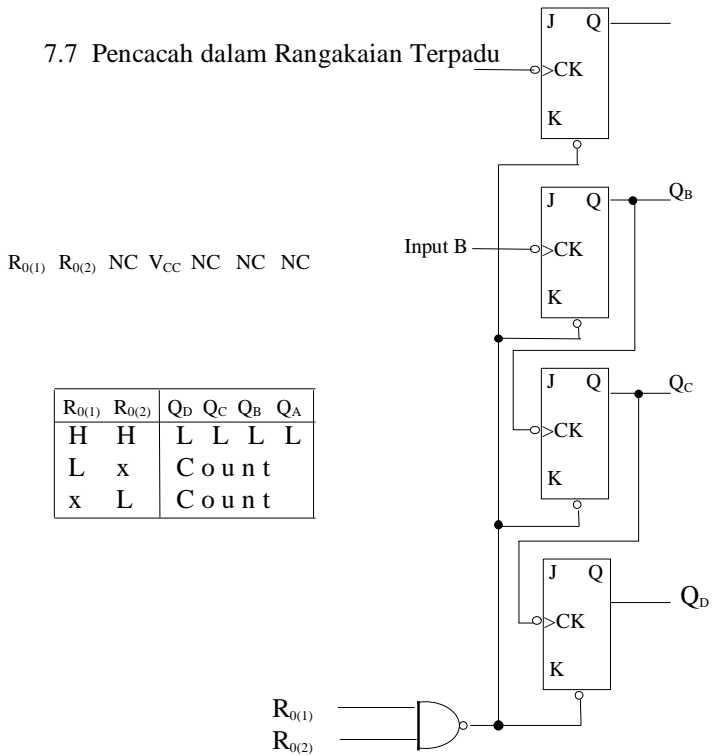
Gambar 7.9. Rangkaian pencacah dengan flip-flop D

Kita perhatikan suatu pencacah biner 4-bit tipe 7493A yang diproduksi oleh Texas Instruments (sering juga disingkat dengan TI). Dalam “Data Book” yang dikeluarkan oleh produsen, serpih ini digambarkan seperti yang ditunjukkan dalam Gambar 7.10 yang menunjukkan bahwa pencacah 4-bit ini tersusun dari 4 flip-flop JK dengan 2 masukan penabuh (clock) dan 2 masukan reset $R_{0(1)}$ dan $R_{0(2)}$. Kedua kendali yang di-NAND-kan secara internal ini digunakan untuk mereset (membuat semua keluaran pencacah berlogika 0) dengan membuat keduanya bertegangan tinggi (berlogika 1).

Perhatikan bahwa masukan J dan K semua flip-flop tak dapat dijangkau (diakses) dari luar. Ini menyiratkan bahwa semua flip-flop JK tersebut dioperasikan dengan $J=K=1$ sehingga berfungsi sebagai flip-flop T dengan penabuh sebagai masukan T. Perhatikan juga masukan penabuh yang aktif rendah (active low) yang menunjukkan bahwa pencacahan terjadi pada perubahan penabuh dari tegangan tinggi ke tegangan rendah. Bila input B dihubungkan ke sumber penabuh luar, maka flip-flop A tidak ada hubungan dengan 3 flip-flop lainnya; flip-flop A akan berdiri sendiri yang berubah keadaan setiap kemunculan penabuh



7.7 Pencacah dalam Rangkaian Terpadu



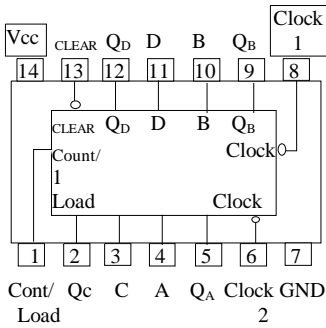
Gambar 7.10. Rangkaian Terpadu Pencacah Biner 4-bit tipe 7493A

dan flip-flop B, C, dan D akan membentuk pencacah 3-bit. Supaya membentuk pencacah 4-bit, maka penabuh flip-flop B (input B) harus dihubungkan ke keluaran flip-flop A (keluaran Q_A), sehingga setiap Q_A berubah dari logika 1 ke 0, yang terjadi setiap kemunculan dua penabuh di input A, terjadi satu penabuhan pada flip-flop B. Bila diinginkan membentuk pencacah lebih besar, maka beberapa serpih pencacah ini dapat dihubungkan secara seri dengan memberikan keluaran Q_D sebagai penabuh flip-flop A (input A) jenjang berikutnya. Perhatikan juga bahwa pencacah ini tak dapat diberi harga awal (inisial) selain reset.

Untuk memenuhi kebutuhan pemberian harga awal tersedia pencacah tipe '160, '163, '176, '177, '196, '197, dan sebagainya yang menyediakan saluran data masukan 4-bit.

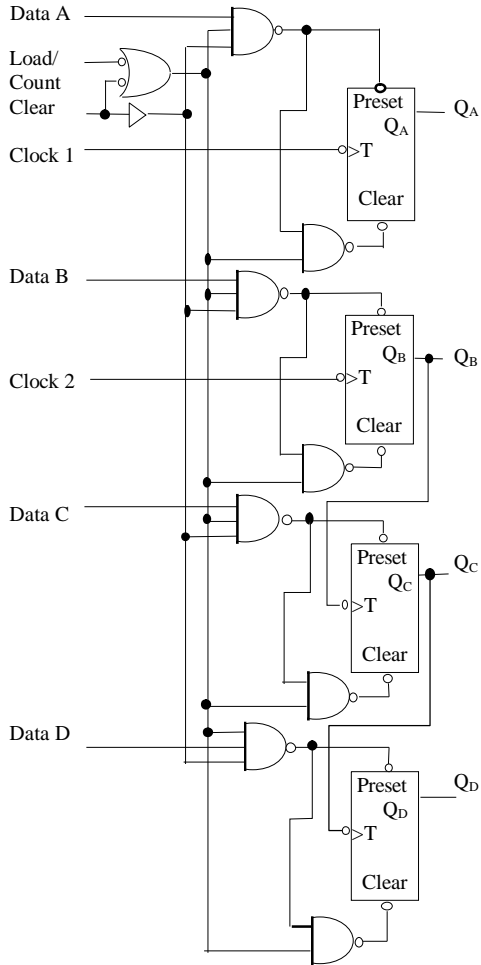
Pada Gambar 7.11 ditunjukkan tatasusunan pencacah tipe '197 yang dapat mencacah baik desimal maupun biner. Bila masukan COUNT/LOAD dibuat berlogika 0, maka keluaran Q_A Q_B Q_C Q_D akan berharga sama dengan data yang

diletakkan pada masukan data ABCD. Tetapi bila masukan COUNT/LOAD dibuat berlogika 1, rangkaian ini akan beroperasi sebagai pencacah. Bila CLOCK 2 dihubungkan ke keluaran Q_A , maka pencacah ini akan beroperasi sebagai pencacah biner 4-bit.



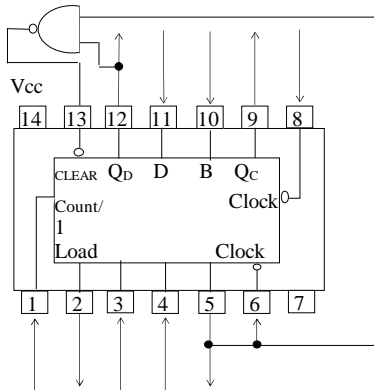
Cacahan	Keluaran*			
	Q_D	Q_C	Q_B	Q_A
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

* Keluaran bila



Pencacah ini dapat dibuat beroperasi sebagai pencacah biner desimal (0-9) dengan menghubungkan keluaran Q_C ke pin CLEAR. Untuk beroperasi sebagai pencacah biner 4-bit, dapat dilakukan dengan menghubungkan keluaran Q_A dan Q_D dan memberikannya kepada masukan CLEAR seperti ditunjukkan pada

Gambar 7.12. Dengan rangkaian seperti ini keluaran pencacah akan menjadi 0000 setelah munculnya pulsa penabuh dalam keadaan keluaran 1000.



Gambar 7.12. Rangkaian Pencacah desimal dengan menggunakan 74197.

7.8 Soal Latihan

- Rencanakanlah suatu pencacah desimal yang melakukan pencacahan dengan urutan: 0,1,2,3,4,5,6,7,8,9,0,1,.. dengan menggunakan:
 - Flip-flop RS
 - Flip-flop JK
 - Flip-flop T

Sediakanlah sarana yang memungkinkan semua flip-flopnya direset (berkeadaan 0).
- Rencanakanlah suatu pencacah 4 bit yang melakukan cacahan dengan urutan: 0,1,3,2,6,14,10,11,9,8,0,1,.. dengan menggunakan:
 - Flip-flop RS
 - Flip-flop JK
 - Flip-flop T
 - Flip-flop D
- Rencanakanlah register 4 bit dengan menggunakan flip-flop RS yang dapat menaikkan dan menurunkan 1 harga yang disimpannya. Register tersebut dapat diberi harga awal melalui masukan paralelnya.

4. Dengan menggunakan rangkaian terpadu 74197 Gambar 7.12, rancanglah suatu pencacah yang akan melakukan pencacahan: 2, 3, 4, 5, 6, 7, 8, 2, 3,

Isyarat: setel (set) data masukan pada harga yang dibutuhkan dan berikan sinyal yang berlogika 0 kepada masukan COUNT/LOAD pada saat pemuatan (loading) dibutuhkan.