



DASAR SISTEM

PERTEMUAN 5
Komparator, ADDer

AGENDA

1. Komparator
2. Adder
3. Multiplexer - Demultiplexer
4. Encoder - Decoder
5. Implementasi Logika Kombinasi dengan Multiplexer dan Decoder

A. Komparator

- Fungsi : Membandingkan keadaan logika input-inputnya
 - **Non equality** → output high bila input beda
 - **Equality** → output high bila input sama
- Tabel kebenaran?
- Karnaugh map?
- NAND? NOR?
- XOR? XNOR

Non-Equality		
input		output
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

Equality		
input		output
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1

Karnaugh map		
B\A	0	1
0	0	1
1	1	0

Karnaugh map		
B\A	0	1
0	1	0
1	0	1

Non Equality Comparator

- Bentuk NAND

- $Y(A,B) = \Sigma(1,2)$

- $Y = A'B + AB'$
$$= \overline{\overline{A'B + AB'}}$$

$$= \overline{\overline{A'B} \cdot \overline{AB'}}$$

- Bentuk NOR

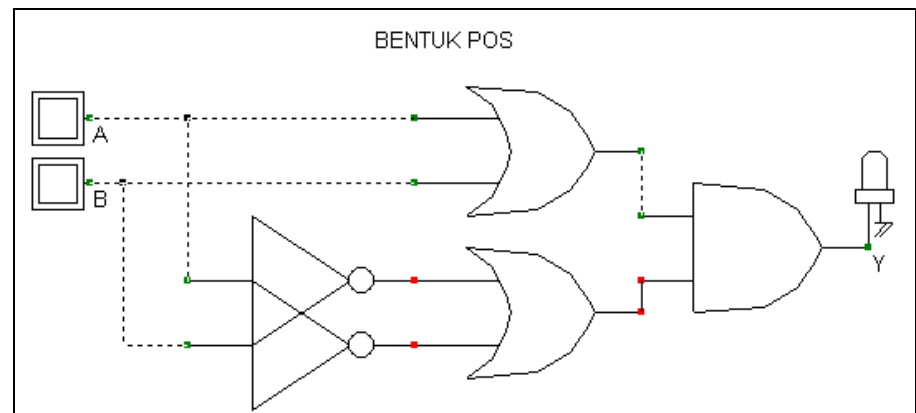
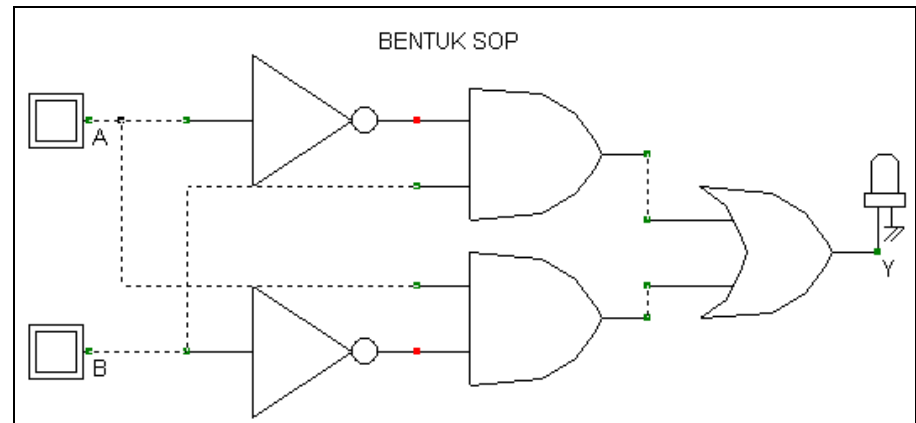
- $Y(A,B) = \Pi(0,3)$

- $Y = (A+B)(A'+B')$
$$= \overline{\overline{(A+B)(A'+B')}} = \overline{(\overline{A+B}) + (\overline{A'+B'})}$$

$$= \overline{(\overline{A+B}) + (A+B')}$$

Non-Equality

input		output
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0



Equality Comparator

- Bentuk NAND

- $Y(A,B) = \Sigma(0,3)$

- $Y = A'B' + AB$

$$= \overline{\overline{A'B' + AB}}$$

$$= \overline{A'B' \cdot AB}$$

- Bentuk NOR

- $Y(A,B) = \Pi(1,2)$

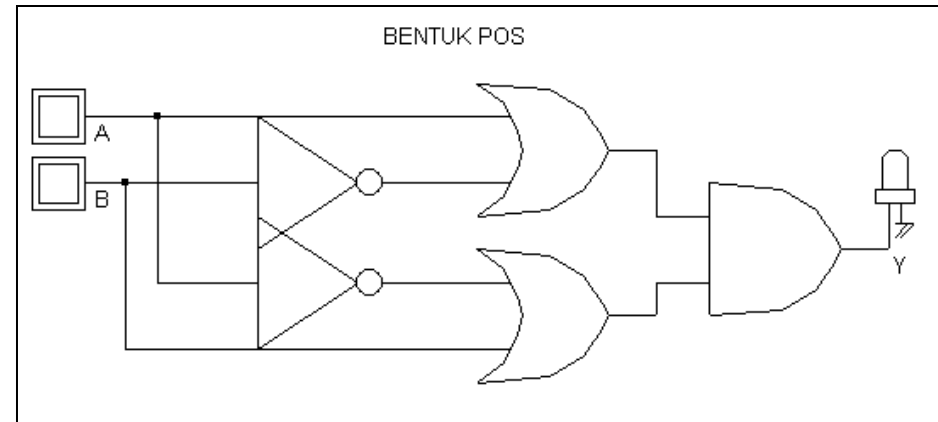
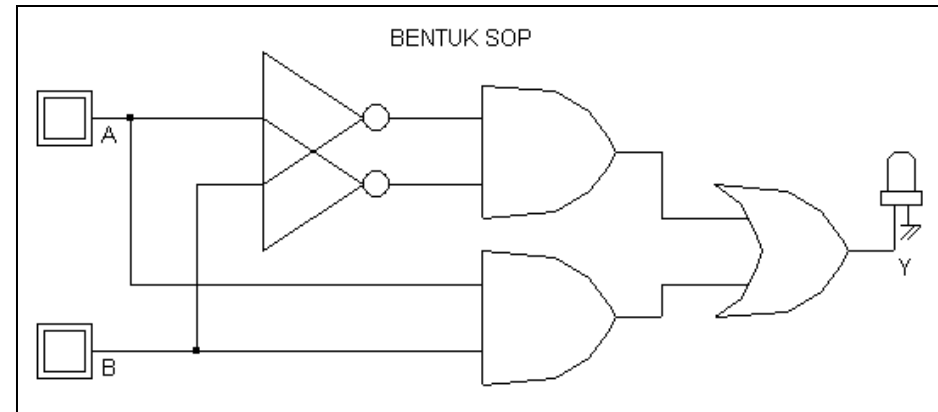
- $Y = (A+B')(A'+B)$

$$= \overline{\overline{(A+B')(A'+B)}}$$

$$= \overline{(\overline{A+B'}) + (\overline{A'+B})}$$

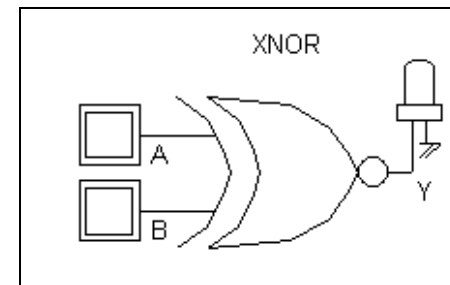
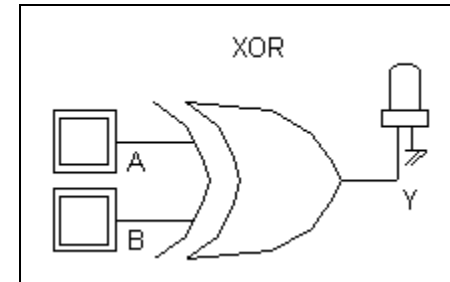
Equality

input		output
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1



XOR dan XNOR

- Kecuali dalam bentuk SOP, POS, NAND, dan NOR, comparator dapat diperoleh dengan menggunakan gerbang :
 - **XOR untuk Non Equality Comp**
 - **XNOR untuk Equality Comp**
- Buktikan :
 - Hasil NOT dari $Y = A'B + AB'$ (non equality comp, XOR) adalah equality comparator $Y = (A+B')(A'+B)$

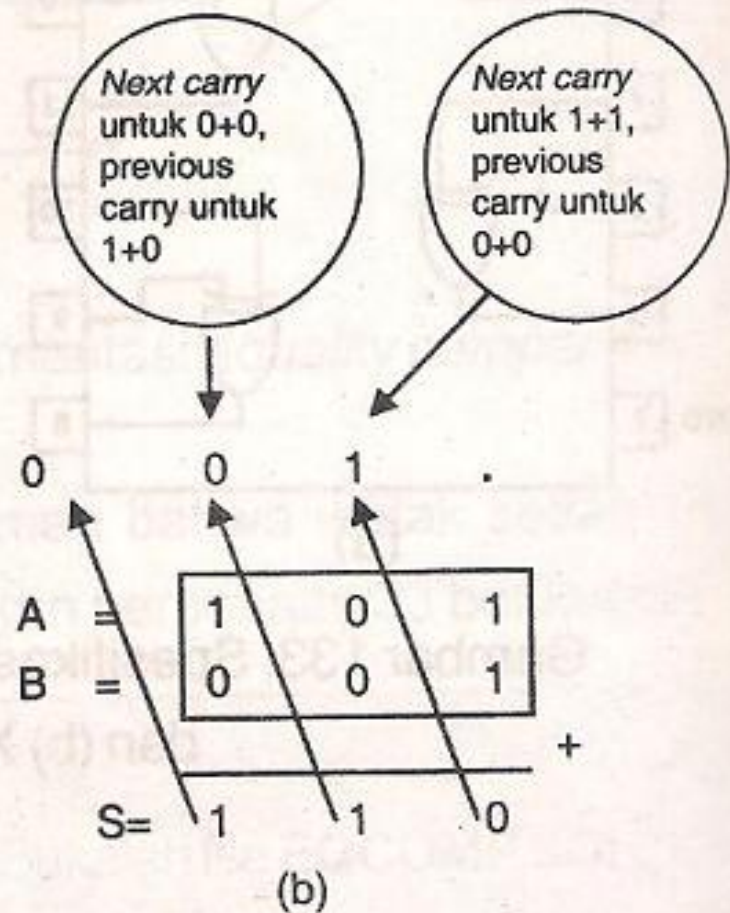
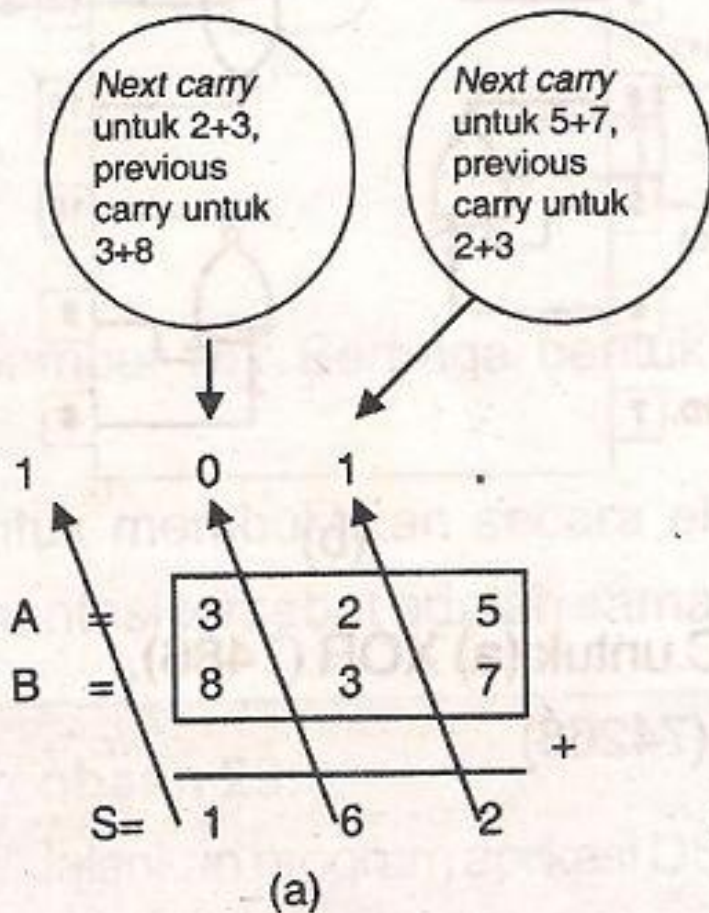


B. Penjumlahan (Adder)

Ilustrasi penjumlahan

a) bilangan desimal

b) bilangan biner



Penjumlah (Adder)

- Fungsi : penjumlahan bilangan biner
 - Half adder → tidak menyertakan bawaan sblmnya (pp. 169)
 - Full adder → menyertakan bawaan sebelumnya (pp. 173)
 - Full adder **paralel** !
- Tabel kebenaran?

Half Adder

Input		Output	
A	B	S	Cn
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

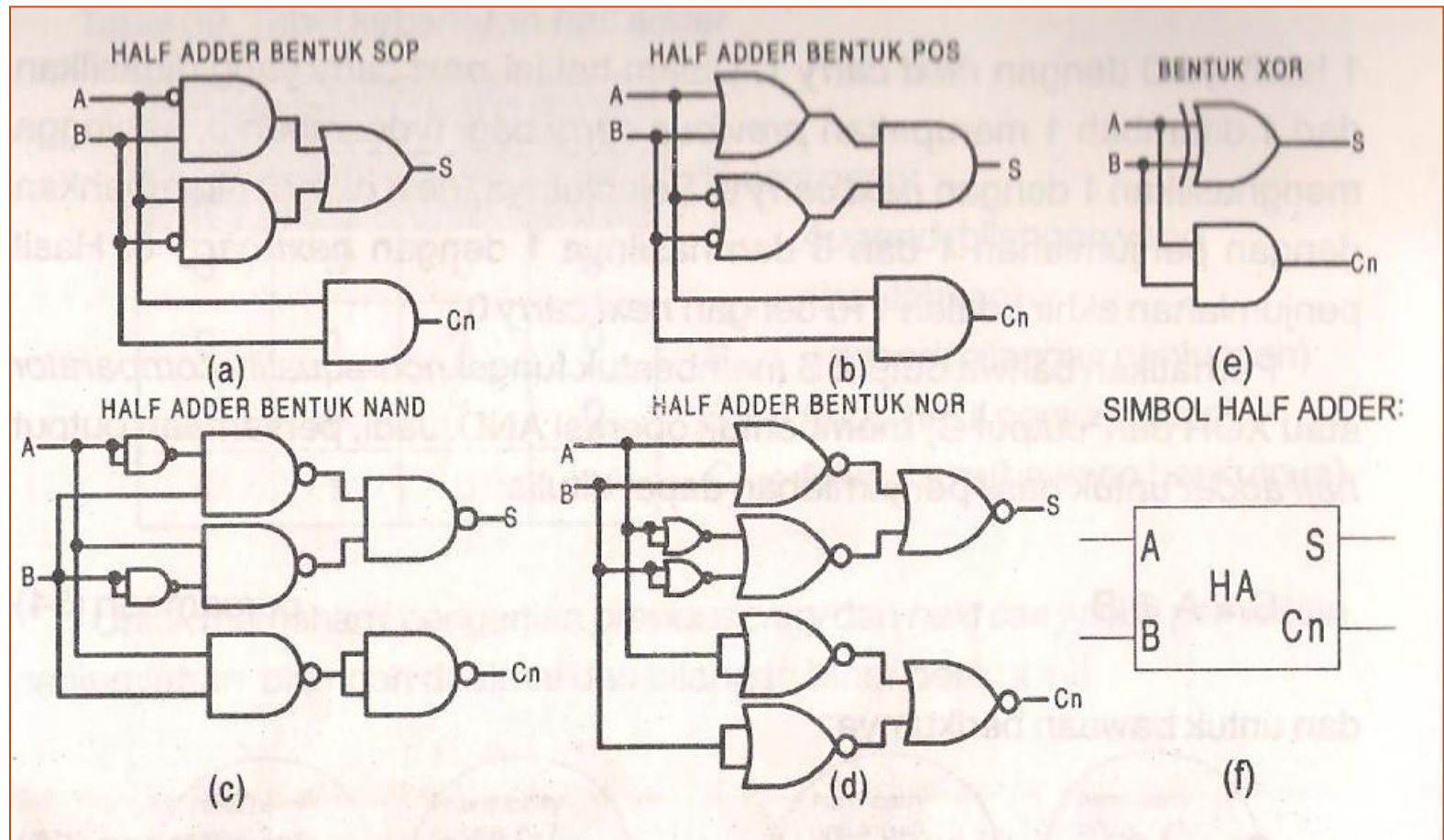
Full Adder

Input			Output	
A	B	Cp	S	Cn
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Half Adder

$$S = A \oplus B$$

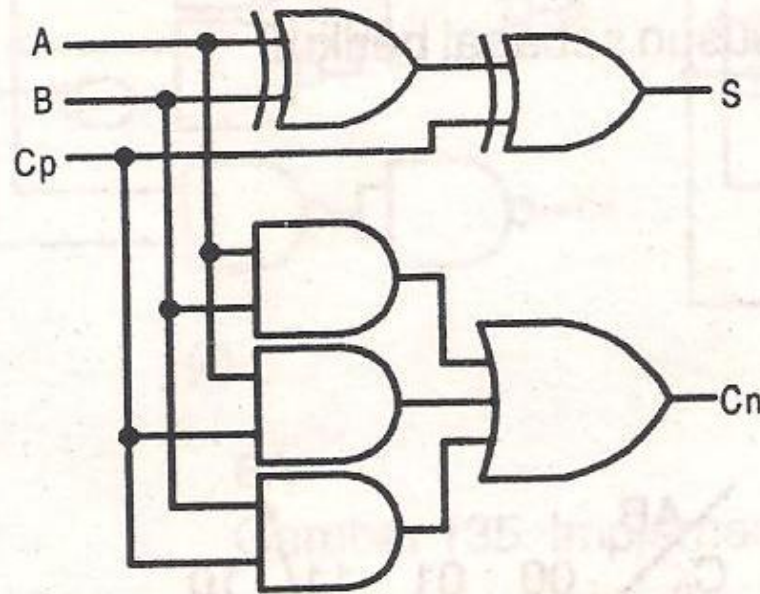
$$C_n = AB$$



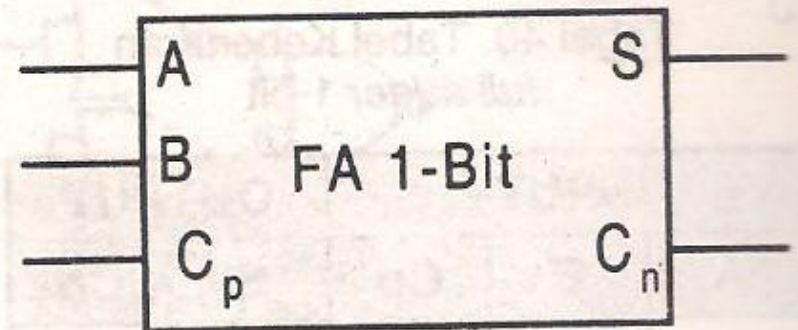
Full Adder

$$S = A \oplus B \oplus C_p$$

$$C_n = AB + AC_p + BC_p$$



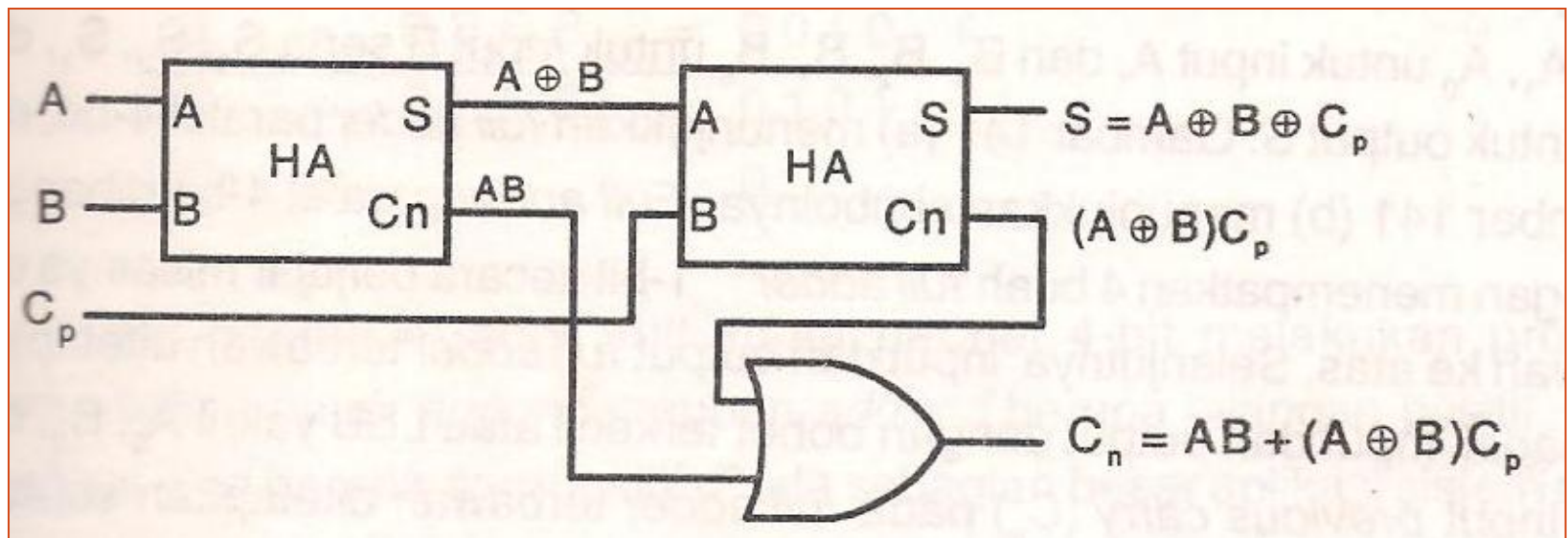
(a)



(b)

Full Adder dari 2 Half Adder

Lihat persamaan (58) → pp. 174

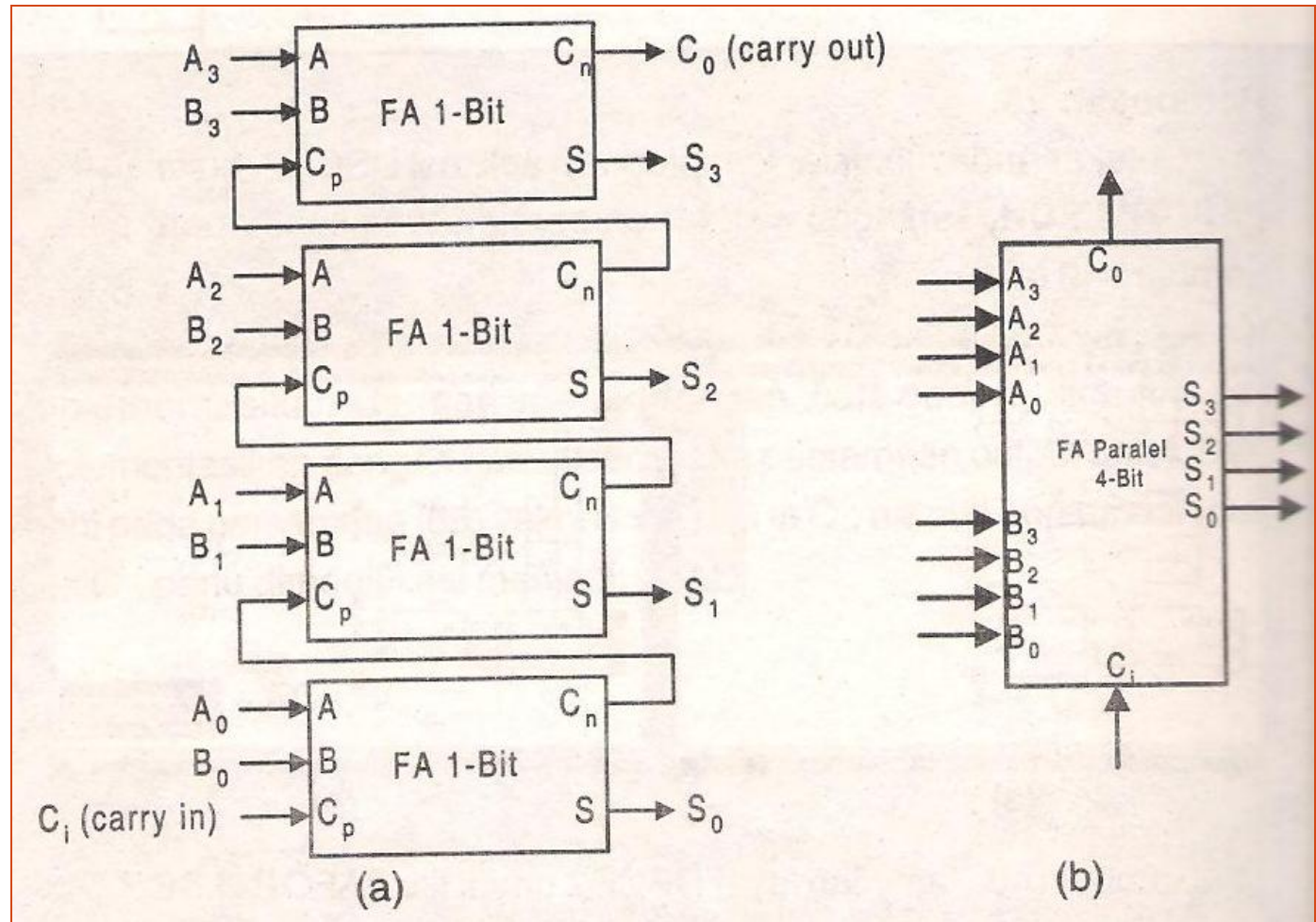


Full Adder Parallel 4 bit

$a_3 a_2 a_1 a_0$

$b_3 b_2 b_1 b_0$

$S_3 S_2 S_1 S_0$



Bagaimana untuk pengurangan??

- Kita ulas dulu ilustrasi representasi bilangan yang dimengerti oleh **ALU**.....

Arithmetic & Logic Unit (ALU)

- Melakukan kalkulasi
- Semua perangkat komputer yang lain bekerja untuk mendukung bagian ini
- Menangani bilangan integer
- ~~• Juga mampu menangani bilangan floating point (real)~~

Representasi integer

- Hanya punya 0 dan 1 untuk mewakili segalanya
- Angka positif dalam bentuk biner
e.g. $41 = 00101001$
- Tidak punya tanda negatif
- ~~• Tidak punya koma~~
- Teknik Sign-Magnitude
- **Two's compliment**

Representasi negatif pada biner

Normal		Bit Bertanda		Komplemen 1		Komplemen 2	
000	+0	000	+0	000	+0	000	+0
001	+1	001	+1	001	+1	001	+1
010	+2	010	+2	010	+2	010	+2
011	+3	011	+3	011	+3	011	+3
100	+4	100	-0	100	-3	100	-4
101	+5	101	-1	101	-2	101	-3
110	+6	110	-2	110	-1	110	-2
111	+7	111	-3	111	-0	111	-1

Jenis Representasi	+5	-5
SM	0101	1101
S1C	-	1010
S2C	-	1011

Lihat Muchlas, Rangkaian Digital, pp. 178

Bit Bertanda (Sign-Magnitude)

- Bit paling kiri adalah bit penanda (sign bit)
- 0 berarti positif, 1 berarti negatif
- $+18 = 00010010$
- $-18 = 10010010$
- Problems
 - Need to consider both sign and magnitude in arithmetic
 - Two representations of zero (+0 and -0)
 - Solusinya → menggunakan **komplemen 2**

Keuntungan Komplemen 2

- Satu representasi nol
- Operasi aritmatika lebih mudah
- Mudah untuk me-negasi-kan
 - $3 = 00000011$
 - Boolean complement gives 11111100
 - Add 1 to LSB 11111101

OK, sekarang kita lihat hubungan antara penjumlahan dengan pengurangan

- Ubah substahan (bagian pengurang) ke bentuk komplemen 2 nya, dan tambahkan ke minuend (bagian yang dikurangi)

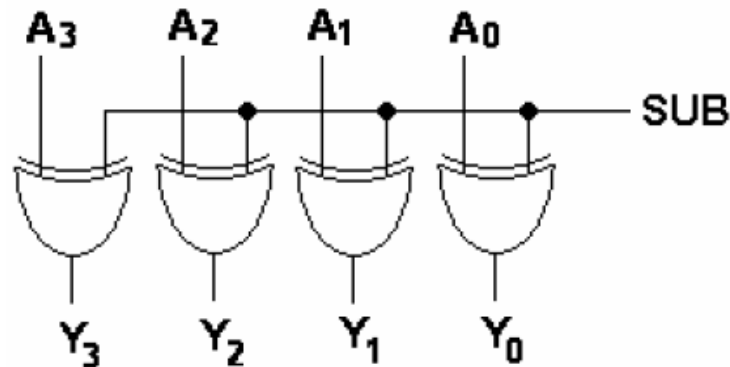
contoh:

$$a - b = a + (-b)$$

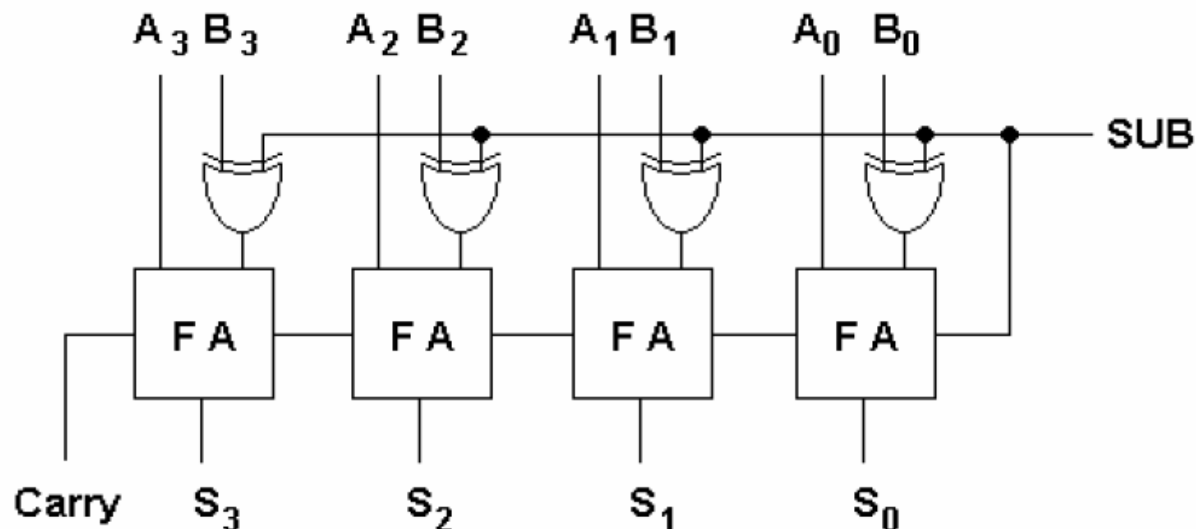
- Tampak bahwa kita hanya butuh rangkaian adder dan komplemen, untuk membentuk subtractor

Membuat rangkaian Adder-Subtractor

Rangkaian diagram logika 1'S complement untuk 4 bit:



Rangkaian diagram Adder-Subtractor (aplikasi paralel full adder) adalah:



$$\begin{array}{r} a_3 a_2 a_1 a_0 \\ b_3 b_2 b_1 b_0 \\ \hline S_3 S_2 S_1 S_0 \end{array}$$

Sekarang, kita latihan dulu 😊

- Silahkan siapkan alat tulis untuk ikut mencoba dan mengikuti contoh-contoh yang akan diberikan.....

Contoh: tunjukkan operasi *full adder* paralel dalam melakukan operasi aritmetika +2+3, +2-3, -2+3, dan -2-3! Anggap $C_i=0$ dan bilangan biner negatif direpresentasikan dalam S2C!

Jawab:

	Desimal	Biner	
C_i	= 0	= 0	
A	= +2	= 0010	
B	= +3	= 0011	+
<hr/>			
S	= +5	= 0101	
C_o	= 0	= 0	
	1		

	Desimal	Biner	
C_i	= 0	= 0	
A	= -2	= 1110	
B	= +3	= 0011	+
<hr/>			
S	= +1	= 0001	
C_o	= 1	= 1	

	Desimal	Biner	
C_i	= 0	= 0	
A	= +2	= 0010	
B	= -3	= 1101	+
<hr/>			
S	= -1	= 1111	
C_o	= 0	= 0	

	Desimal	Biner	
C_i	= 0	= 0	
A	= -2	= 1110	
B	= -3	= 1101	+
<hr/>			
S	= -5	= 1011	
C_o	= 1	= 1	

Aritmatika Bilangan Biner 4 bit

Dalam representasi komplement 2

a) $(-7) + (+5)$

1001

0101

0010 = -2

b) $(-4) + (+4)$

1100

0100

1 0010 = 0

c) $(+3) + (+4)$

0011

0100

0111 = 7

d) $(-4) + (-1)$

1100

1111

1011 = -5

e) $(+5) + (+4)$

0101

0100

1001 = overflow

f) $(-7) + (-6)$

1001

1010

1 0011 = overflow

- Solusi Overflow:

Pada ALU, harus memberikan sinyal tentang keadaan ini sehingga tidak terdapat usaha untuk menggunakan hasil operasi tersebut.

- Indikator Overflow:

Bila dua buah bilangan ditambahkan, keduanya positif atau keduanya negatif, maka overflow akan dan hanya terjadi bila hasilnya memiliki tanda yang berlawanan

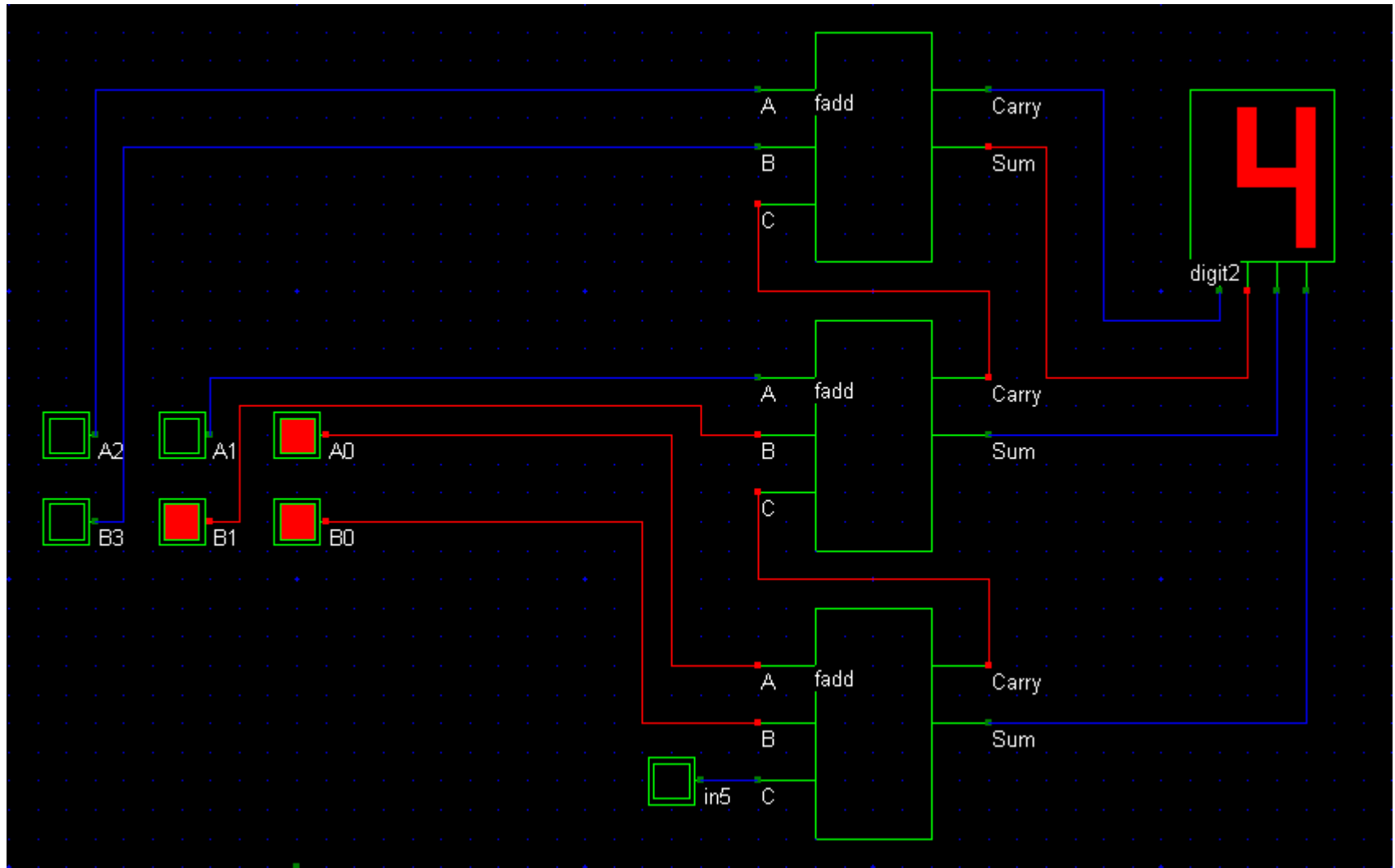
Latihan mandiri

- lihat tugas 27 (pp. 180)

Tunjukkan operasi **full adder paralel 4-bit** dalam melakukan operasi aritmetika $+4+2$, $+4-2$, $-4+2$, dan $-4-2$

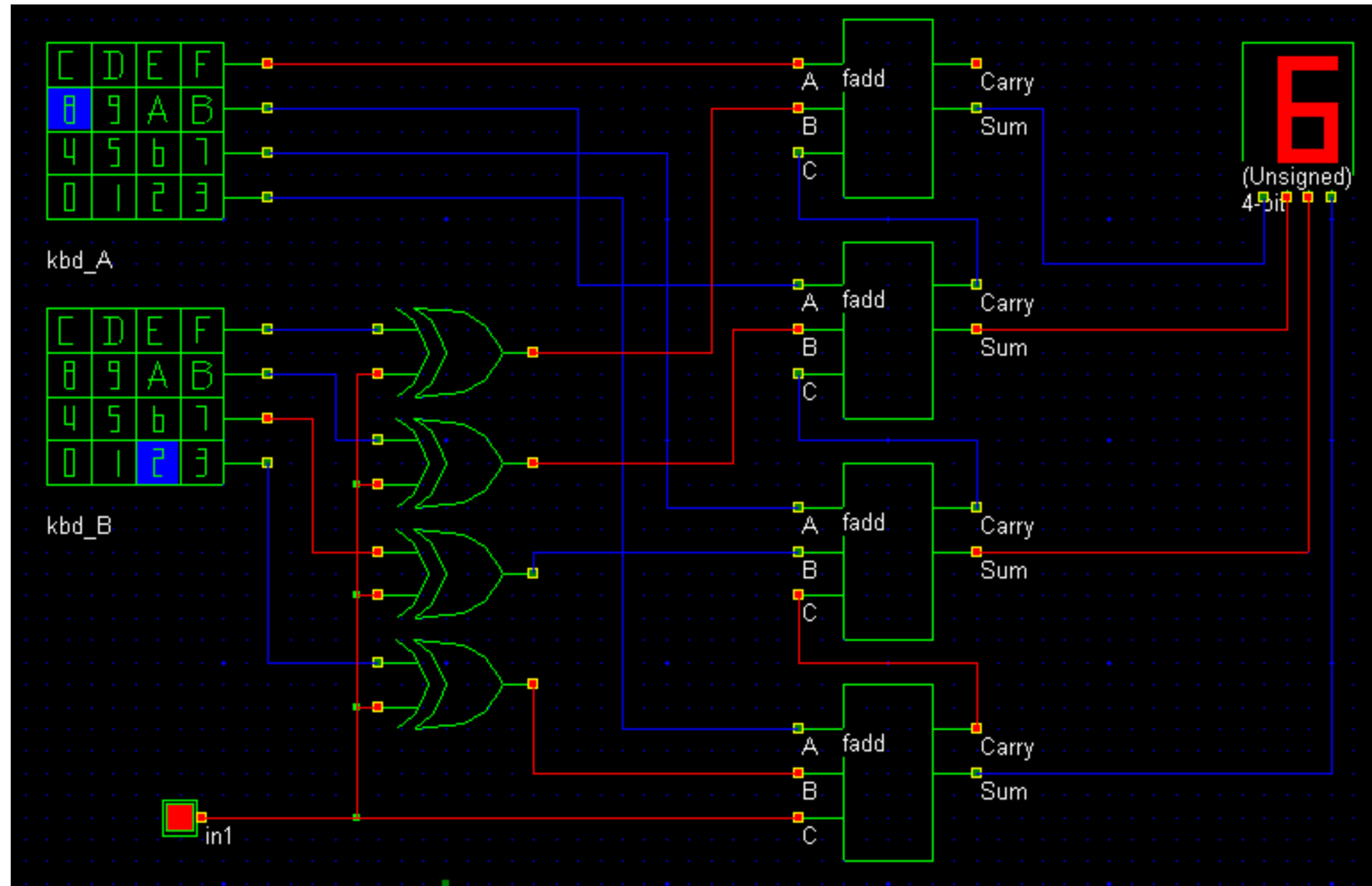
- Soal nomor 5, pp. 230 akan dikerjakan di praktikum

Contoh 3 bit full adder paralel



Contoh 4 bit full adder-subtractor

$A + B$ atau $A + (-B)$



DSCH → ADD4SIGNED.SCH