

LAB #1

DASAR RANGKAIAN DIGITAL

TUJUAN

1. Untuk mempelajari operasi dari gerbang logika dasar.
2. Untuk membangun rangkaian logika dari persamaan Boolean.
3. Untuk memperkenalkan beberapa konsep dasar dan teknik laboratorium dalam bekerja menggunakan gerbang logika digital.

PENDAHULUAN

Sebuah gerbang logika adalah sekumpulan rangkaian dasar dari rangkaian digital. Kebanyakan gerbang logika memiliki dua *input* dan satu *output*. Pada setiap saat, setiap terminal dalam satu dari dua kondisi biner *LOW* (0) atau *HIGH* (1), diwakili oleh level tegangan yang berbeda. Keadaan logika terminal bisa, dan umumnya tidak, sering berubah-ubah, sebagai data rangkaian proses. Dalam kebanyakan gerbang logika, kondisi *LOW* adalah sekitar nol volt (0 V), sedangkan kondisi *HIGH* adalah sekitar lima volt positif (+5 V).

Gerbang logika merupakan komponen yang paling sederhana dari setiap sirkuit logika. Jadi, untuk memahami logika komputer, Anda harus memahami dan menguasai operator logika (gerbang). Gerbang adalah rangkaian elektronik digital yang hanya memiliki satu *output* tetapi satu atau lebih *input*. *Output* atau sinyal akan muncul pada *output* gerbang hanya pada kombinasi sinyal *input* tertentu.

Ada banyak jenis gerbang logika; seperti AND, OR dan NOT, yang biasanya disebut tiga gerbang dasar. Gerbang populer lainnya adalah NAND dan gerbang NOR; yang hanya kombinasi AND atau gerbang OR dengan gerbang NOT dimasukkan sebelum sinyal keluaran. Gerbang XOR lainnya termasuk "Exclusive-OR" dan XNOR gerbang "Exclusive NOR".

Menggunakan kombinasi gerbang logika, operasi yang kompleks dapat dilakukan. Secara teori, tidak ada batasan untuk jumlah gerbang yang dapat tersusun bersama dalam satu perangkat. Tapi dalam prakteknya, ada batasan untuk jumlah gerbang yang dapat dikemas ke dalam ruang fisik yang diberikan. *Array* gerbang logika ditemukan dalam sirkuit digital terintegrasi (IC). Sebagai kemajuan teknologi IC, volume fisik yang diperlukan untuk setiap penurunan gerbang logika individu dan perangkat digital dengan ukuran yang sama atau lebih kecil menjadi mampu melakukan operasi yang terus lebih-rumit pada kecepatan yang terus meningkat.

Dalam percobaan ini, kita akan menyelidiki semua gerbang logika dikenal dan mempelajari operasi mereka sesuai dengan tabel kebenaran.

KEBUTUHAN

1. Koper lengkap HBE-LogicCircuit-Digital
2. Kabel tembaga

TUGAS PENDAHULUAN

1. Baca dan pahami *Technical Guide* terlebih dahulu!
2. Pelajari semua *datasheet* dari semua IC yang digunakan pada percobaan ini!
3. Sebutkan tiga keluarga yang digunakan dalam Digital IC dan keuntungannya!
4. Apa yang dimaksud dengan *Truth Table*?
5. Gambarkan rangkaian 2-input NAND dan NOT dengan menggunakan transistor BJT, jelaskan cara kerjanya!
6. Tuliskan persamaan Hukum De Morgan!
7. Apa pengoperasian POS dan SOP? Jelaskan!
8. Apa yang anda ketahui tentang *Binary Half Adder* dan *Binary Full Adder*!
9. $4-7 = -3$, jelaskan pengurangan persamaan ini dalam biner!

EKSPERIMEN 1 : GERBANG LOGIKA

[Percobaan 1] Operasi AND

[Persiapan]

Alat I/O	Slide Switch (SW1, SW2), LED (D1)
Modul	Modul AND
Lain-lain	Kabel (untuk menghubungkan alat I/O dengan modul)

[Prosedur]

1. Pada Fig. I-1, gunakan modul AND dan hubungkan rangkaian menggunakan kabel.

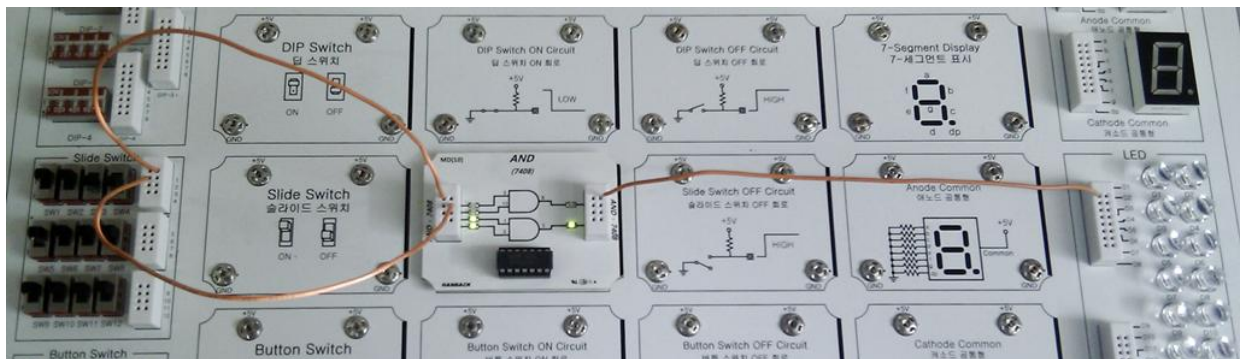
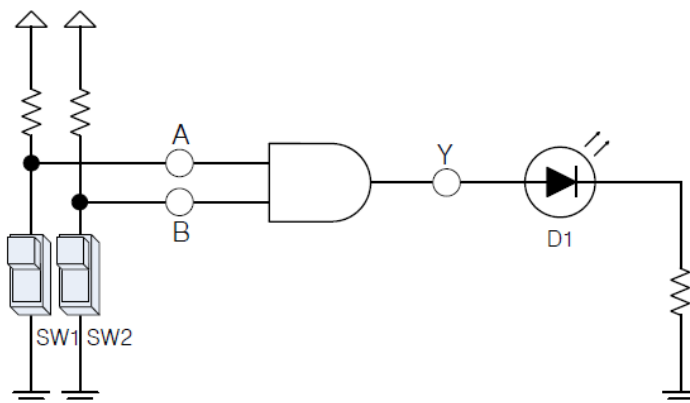


Fig. I-1. Diagram Operasi AND

2. Catat hasil percobaan pada Tabel I-1.

Tabel I-1. Hasil Operasi AND		
Input		Output
A	B	Y
0	0	
0	1	
1	0	
1	1	

[Percobaan 2] Operasi OR

[Persiapan]

Alat I/O	Slide Switch (SW1, SW2), LED (D1)
Modul	Modul OR
Lain-lain	Kabel (untuk menghubungkan alat I/O dengan modul)

[Prosedur]

1. Pada Fig. I-2, gunakan modul OR dan hubungkan rangkaian menggunakan kabel.

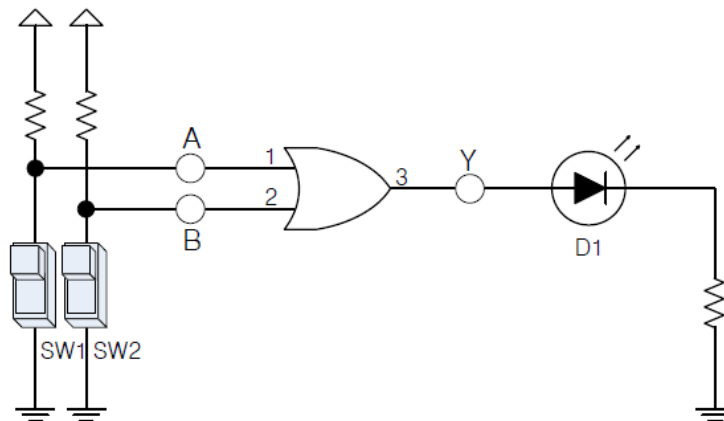


Fig. I-2. Diagram Operasi OR

2. Catat hasil percobaan pada Tabel I-2.

Tabel I-2. Hasil Operasi OR		
Input		Output
A	B	Y
0	0	
0	1	
1	0	
1	1	

[Percobaan 3] Operasi NOT

[Persiapan]

Alat I/O	Slide Switch (SW1), LED (D1)
Modul	Modul NOT
Lain-lain	Kabel (untuk menghubungkan alat I/O dengan modul)

[Prosedur]

1. Pada Fig. I-3, gunakan modul NOT dan hubungkan rangkaian menggunakan kabel.

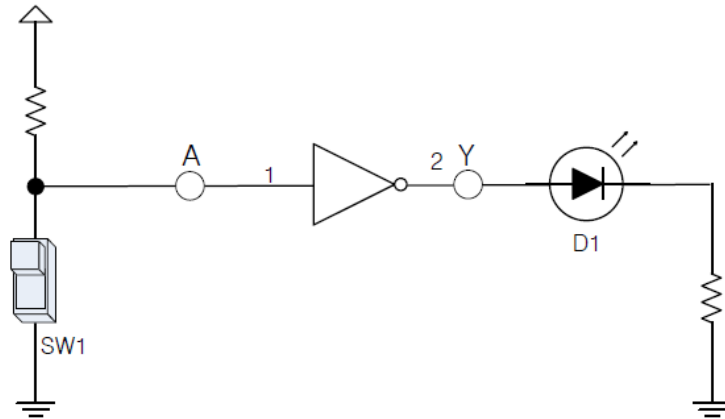


Fig. I-3. Diagram Operasi NOT

2. Catat hasil percobaan pada Tabel I-3.

Tabel I-3. Hasil Operasi NOT	
Input A	Output Y
0	
1	

[Percobaan 4] Operasi NAND

[Persiapan]

Alat I/O	Slide Switch (SW1, SW2), LED (D1)
Modul	Modul NAND
Lain-lain	Kabel (untuk menghubungkan alat I/O dengan modul)

[Prosedur]

1. Pada Fig. I-4, gunakan modul NAND dan hubungkan rangkaian menggunakan kabel.

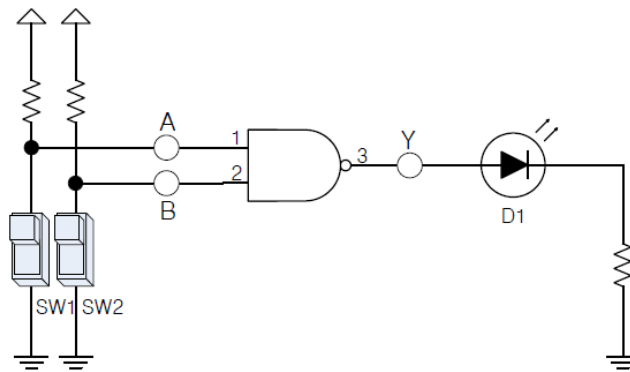


Fig. I-4. Diagram Operasi NAND

2. Catat hasil percobaan pada Tabel I-4.

Tabel I-4. Hasil Operasi NAND		
Input		Output
A	B	Y
0	0	
0	1	
1	0	
1	1	

[Percobaan 5] Operasi NOR

[Persiapan]

Alat I/O	Slide Switch (SW1, SW2), LED (D1)
Modul	Modul NOR
Lain-lain	Kabel (untuk menghubungkan alat I/O dengan modul)

[Prosedur]

1. Pada Fig. I-5, gunakan modul NOR dan hubungkan rangkaian menggunakan kabel.

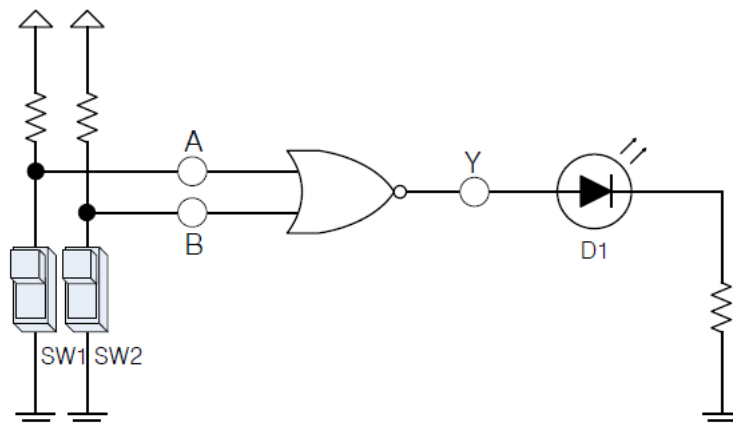


Fig. I-5. Diagram Operasi NOR

2. Catat hasil percobaan pada Tabel I-5.

Tabel I-5. Hasil Operasi NOR		
Input		Output
A	B	Y
0	0	
0	1	
1	0	
1	1	

[Percobaan 6] Operasi XOR dan XNOR

[Persiapan]

Alat I/O	Slide Switch (SW1, SW2), LED (D1)
Modul	Modul XOR
Lain-lain	Kabel (untuk menghubungkan alat I/O dengan modul)

[Prosedur]

1. Pada Fig. I-6, gunakan modul XOR dan hubungkan rangkaian menggunakan kabel.

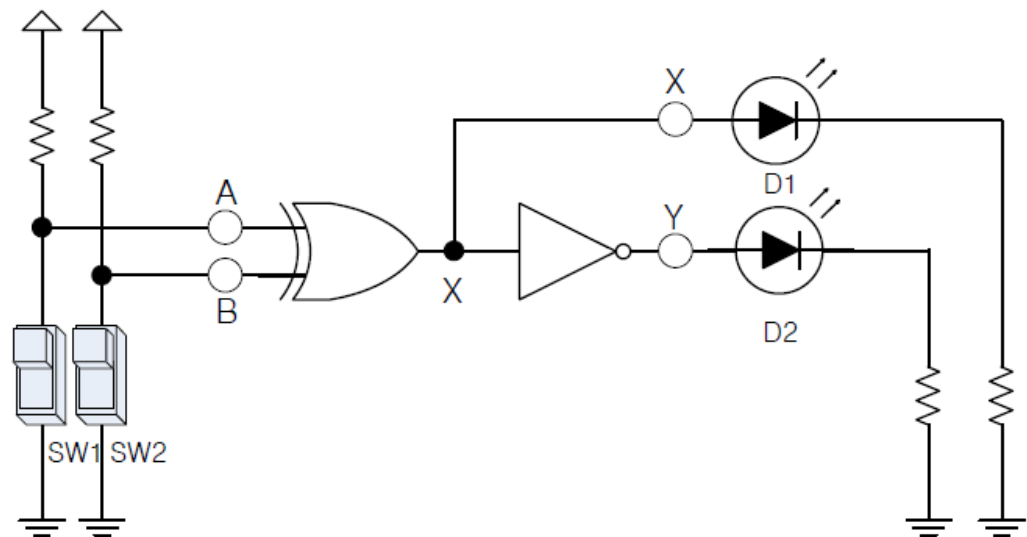


Fig. I-6. Diagram Operasi XOR

2. Catat hasil percobaan pada Tabel I-6.

Tabel I-6. Hasil Operasi XOR		
Input		Output
A	B	Y
0	0	
0	1	
1	0	
1	1	

Hasil Operasi XNOR		
Input		Output

A	B	Y
0	0	
0	1	
1	0	
1	1	

EKSPERIMEN 2 : ALJABAR BOOLEAN

[Percobaan 7] Hukum Distributif : $A + (B \cdot C) = (A + B) \cdot (A + C)$

[Persiapan]

Alat I/O	Slide Switch (SW1, SW2, SW3), LED (D1)
Modul	Modul AND, Modul OR
Lain-lain	Kabel (untuk menghubungkan alat I/O dengan modul)

[Prosedur]

- Gunakan modul AND dan modul OR untuk membangun rangkaian (a) dan (b) seperti pada Fig. I-7.

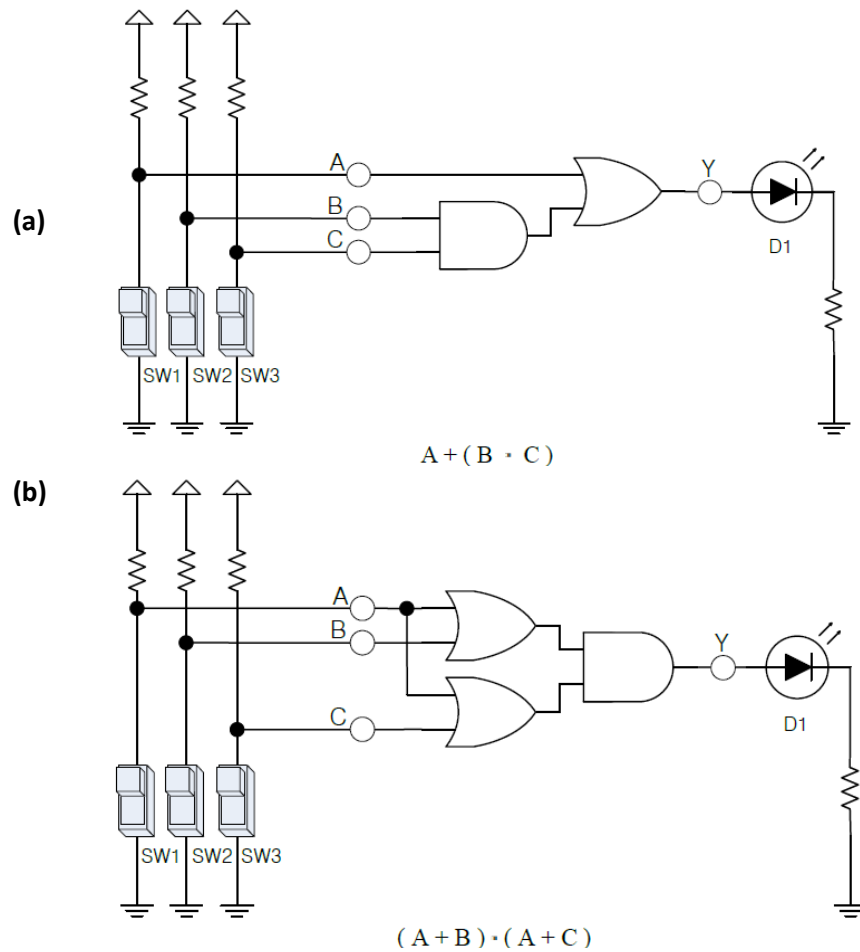


Fig. I-7. Diagram Operasi Hukum Distributif

2. Catat hasil percobaan pada Tabel I-7.

Tabel I-7. Hasil Operasi Hukum Distributif				
Input			Output	
A	B	C	Y(a)	Y(b)
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

EKSPERIMEN 3 : RANGKAIAN LOGIKA KOMBINASIONAL

[Percobaan 8] Half Adder

[Persiapan]

Alat I/O	Slide Switch (SW1, SW2), LED (D1, D2)
Modul	Modul XOR, Modul AND
Lain-lain	Kabel (untuk menghubungkan alat I/O dengan modul)

[Prosedur]

1. Gunakan semua modul yang dibutuhkan untuk membangun rangkaian *Half Adder* seperti pada Fig. I-8.

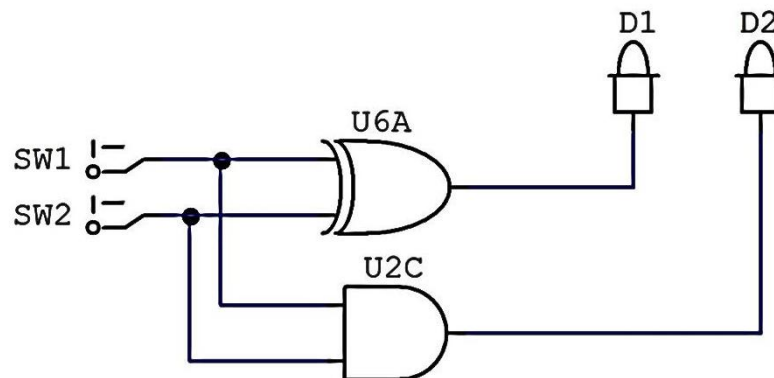


Fig. I-8. Diagram Operasi Half Adder

2. Catat hasil percobaan pada Tabel I-8.

Tabel I-8. Hasil Operasi Half Adder			
Input		Output	
A (SW1)	B (SW2)	S (D1)	C (D2)
0	0		
0	1		

1	0		
1	1		

3. Simulasikan menggunakan DAQ. Lalu simpan *waveform*-nya dan lampirkan dalam laporan.

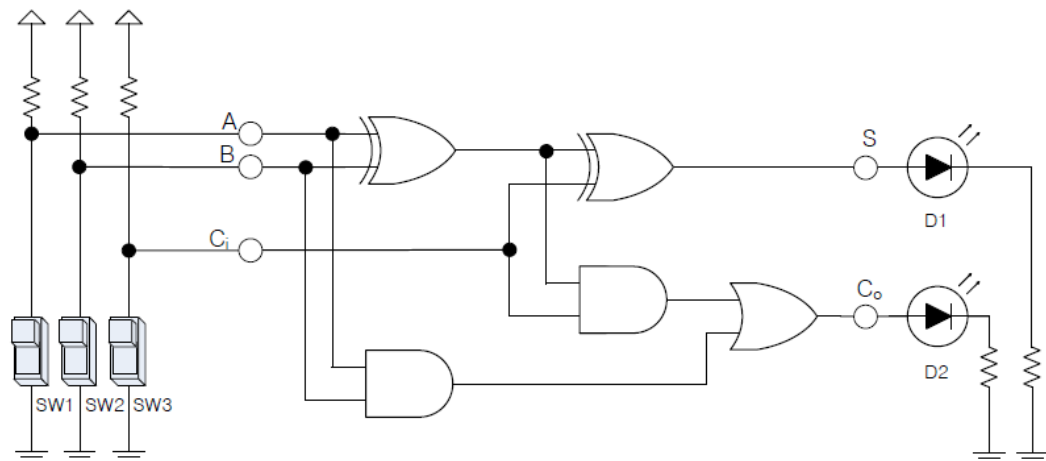
[Percobaan 9] Full Adder

[Persiapan]

Alat I/O	Slide Switch (SW1, SW2, SW3), LED (D1, D2)
Modul	Modul XOR, Modul AND, Modul OR
Lain-lain	Kabel (untuk menghubungkan alat I/O dengan modul)

[Prosedur]

1. Gunakan semua modul yang dibutuhkan untuk membangun rangkaian *Full Adder* seperti pada Fig. I-9 lalu hubungkan rangkainnya dengan menggunakan kabel.



Full Adder

Fig. I-9. Diagram Operasi Full Adder

2. Catat hasil percobaan pada Tabel I-9.

Tabel I-9. Hasil Operasi Full Adder				
Input			Output	
A (SW1)	B (SW2)	C _i (SW3)	S (D1)	C _o (D2)
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

3. Simulasikan menggunakan DAQ. Lalu simpan *waveform*-nya dan lampirkan dalam laporan.

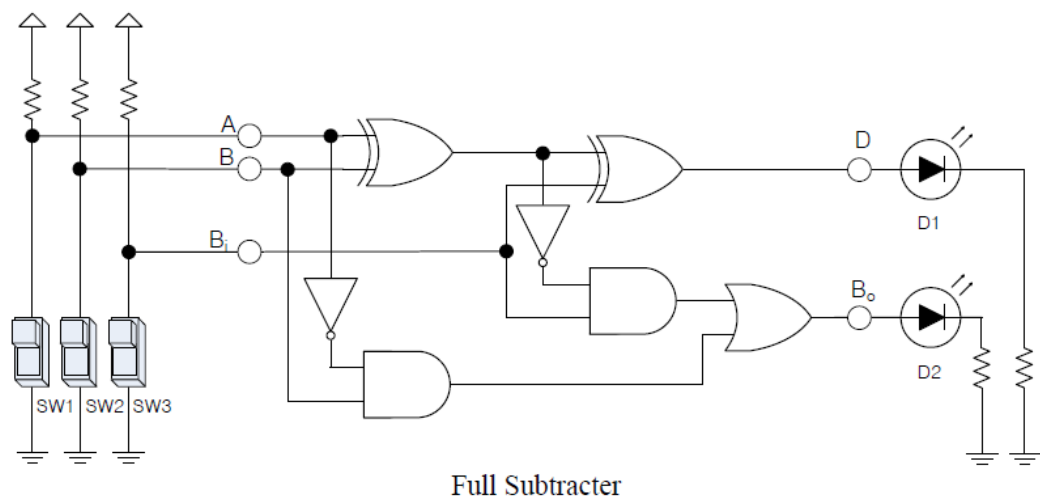
[Percobaan 10] Full Subtractor

[Persiapan]

Alat I/O	Slide Switch (SW1, SW2, SW3), LED (D1, D2)
Modul	Modul XOR, Modul AND, Modul OR, Modul NOT
Lain-lain	Kabel (untuk menghubungkan alat I/O dengan modul)

[Prosedur]

1. Gunakan semua modul yang dibutuhkan untuk membangun rangkaian *Full Subtractor* seperti pada Fig. I-10.



Full Subtractor
Fig. I-10. Diagram Operasi Full Subtractor

2. Catat hasil percobaan pada Tabel I-10.

Tabel I-10. Hasil Operasi Full Subtractor				
Input			Output	
A (SW1)	B (SW2)	Bi (SW3)	S (D1)	Bo (D2)
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

3. Simulasikan menggunakan DAQ. Lalu simpan *waveform*-nya dan lampirkan dalam laporan.

TUGAS MODUL

1. Simulasikan semua percobaan pada Circuit Maker!

2. Isi tabel berikut :

A	B	\overline{A}	B	$\overline{A} \cdot \overline{B}$	$\overline{\overline{A} \cdot \overline{B}}$	A + B
0	0					
0	1					
1	0					
1	1					

3. Buat gerbang OR dan XOR dari gerbang NAND dan NOT!
4. Buatlah gerbang XOR dan XNOR tiga input dengan gate dasar dua input. (simulasikan hasil jawaban anda menggunakan circuit maker)
5. Dalam Hukum Distributif, mana yang lebih mudah dan murah untuk membuat:
 - a. Hanya menggunakan modul NAND saja untuk membuat rangkaian
 - b. Menggunakan modul AND dan OR gerbang seperti pada percobaanJelaskan jawabanmu!
6. Jelaskan bagaimana cara kerja *Half Adder*, *Full Adder*, and *Full Subtractor*!