

LAB SHEET 4

OPERASI LOGIKA

A. TUJUAN

1. Dapat menjalankan program operasi logika serta mengecek isi setiap register yang digunakan.

B. TUJUAN

1. Dapat menjalankan program operasi logika serta mengecek isi setiap register yang digunakan.

C. BAHAN DAN ALAT

1. Lembar tugas
2. Z80 Simulator IDE Oshonsoft

D. TEORI DASAR

Operasi logika itu terdiri dari AND, OR, NOT, XOR.

Tabel kebenaran dari operasi logika:

AND			OR			XOR		
P	Q	(P×Q)	P	Q	(P+Q)	P	Q	(P+Q)
0	0	0	0	0	0	0	0	0
0	1	0	0	1	1	0	1	1
1	0	0	1	0	1	1	0	1
1	1	1	1	1	1	1	1	0

Contoh:

P = 10110001B = B2H

Q = 01101010B = 6AH

6AH

————— AND

00100010B = 22H

P = 10110010B = B2H

Q = 01101010B = 6AH

————— OR

11111010B = FAH

P = 10110010B = B2H

Q = 01101010B =

————— XOR

11011000B = D8H

Register flag merupakan register pembantu terhadap operasi aritmatik dan operasi logika. Bantuk bantuannya berupa menyimpan tanda keadaan bilamana terjadi carry, non carry, borrow, zero, parity/overflow. Letak definisi simpan keadaan flag pada register F dapat dilukiskan sebagai berikut:

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
S	Z	-	H	-	P/V	N	C

- S = sign
- Z = zero
- H = half carry
- P/V = parity/overflow (=P/O)
- N = non carry
- C = carry

Dengan adanya operasi aritmatik atau logika, maka kejadian dari register F dapat dibuat tabel sebagai berikut:

Bit	Nama Flag	Keterangan
0	C	0 : operasi aritmatik/logika tidak terjadi carry 1 : operasi aritmatik/logik terjadi carry
1	N	0 : operasi yang telah terjadi bukan substract 1 : operasi yang telah terjadi adalah substract
2	P/V	0 : parity even/genap tidak terjadi overflow 1 : parity odd/ganjil terjadi overflow
3	-	TIDAK DIGUNAKAN
4	H	0 : adisi/substract non carry di bit 4 Reg. A 1 : adisi/substract ada carry di bit 4 Reg. A
5	-	TIDAK DIGUNAKAN
6	Z	0 : hasil operasi aritmatik/logik tidak nol 1 : hasil operasi aritmatik/logik sama dengan nol
7	S	0 : hasil operasi aritmatik/logik adalah positif 1 : hasil operasi aritmatik/logik adalah negatif

Untuk latihan cobalah lakukan pengujian menggunakan MPF-1 Z-80 program berikut di mana di dalamnya terdapat operasi aritmatik dan operasi logik.

- LD B, 1FH Isikan 1FH ke Reg. B
- LD C, B Kutip isi Reg B ke RegC
- LD A, 01H Isikan 01H ke Reg. A
- AND 0FH Isi reg A di-AND-kan

INC B

dengan 0FH dan hasilnya disimpan pada reg A = 00H
Isi reg B ditambah satu dan hasilnya disimpan pada reg B = 20H

Langkah percobaan :

1. Jalankan Z80 Simulator IDE
2. Klik Tools / Assembler dan ketik program berikut (tanpa komentar, hanya perintah menmonik) :

```
LD B,1FH
LD C,B
LD A,01H
AND 0FH
INC B
HALT
.END3.
```

3. Pada jendela Assembler klik Tools / Assemble untuk cek kesalahan, simpan file ASM ke folder anda masing-masing.
4. Muat program ke Simulator dengan mengklik Tools / Assemble & Load.
5. Jalankan simulator dengan mengklik Simulation / Start
6. Cek hasil tampilan jendela Simulator Z80, lihat nilai-nilai register apakah sudah sesuai dengan yang diisikan dan cek pula pada tampilan file log.txt

The screenshot displays the Z80 Simulator IDE interface. The top-left pane shows the main registers (A, B, C, D, E, H, L) and alternate registers (A', B', C', D', E', H', L'). The top-right pane shows the assembly code being assembled. The bottom pane shows the simulation log, which includes the instruction stream and the state of the registers and special registers.

Assembler - labsheet4.asm

```
0001 LD B,1FH
0002 LD C,B
0003 LD A,01H
0004 AND 0FH
0005 INC B
0006 HALT
0007 .END
```

Simulation Log

```
Simulation started at 20/09/2010 5:38:22.
No. PC Instruction A SZYHXPNC B C D E H L IX IY SP A' SZYHXPNC' B' C' D' E' H' L'
1 0000 LD B,1FH FF 11111111 1F FF FF FF FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
2 0002 LD C,B FF 11111111 1F 1F FF FF FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
3 0003 LD A,01H 01 11111111 1F 1F FF FF FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
4 0005 AND 0FH 01 00111000 1F 1F FF FF FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
5 0007 INC B 01 00111000 20 1F FF FF FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
6 0008 HALT 01 00111000 20 1F FF FF FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
Simulation stopped at 20/09/2010 5:38:23.
0.43 seconds elapsed.
Real-time duration of the simulation is 33 clock cycles or 8.25 µs at 4 MHz.
```

Dari program di atas jika dilihat isi registernya masing-masing akan terlihat seperti pada keterangan program. Selain itu dapat dilihat pengaruhnya terhadap isi register flagnya setelah terjadi operasi aritmatik atau operasi logik yaitu dengan melihat harga-harga yang ada pada masing-masing bit dari register F (flag). Setelah terjadi operasi aritmatik/logik pada prgram di atas maka register flag yang terpengaruh adalah sebagai berikut:

	S	Z	-	H	-	P/V	N	C
- operasi ADD A, B	0	0	-	1	-	0	0	0
- operasi AND 0FH	0	1	-	1	-	1	0	0
- operasi INC B	0	0	-	1	-	0	0	0

Keterangan:

- Setelah terjadi operasi logik AND maka reg. F yang terpengaruh adalah S, Z, P, maka setelah terjadi operasi logik AND 0FH dapat dijabarkan sebagai berikut:
 - S = 0 : tidak terjadi perubahan tanda
 - Z = 0 : hasilnya tidak sama dengan nol
 - H = 1 : terjadi half carry
 - V = 0 : pada operasi tersebut tidak terjadi overflow

D. LANGKAH PERCOBAAN

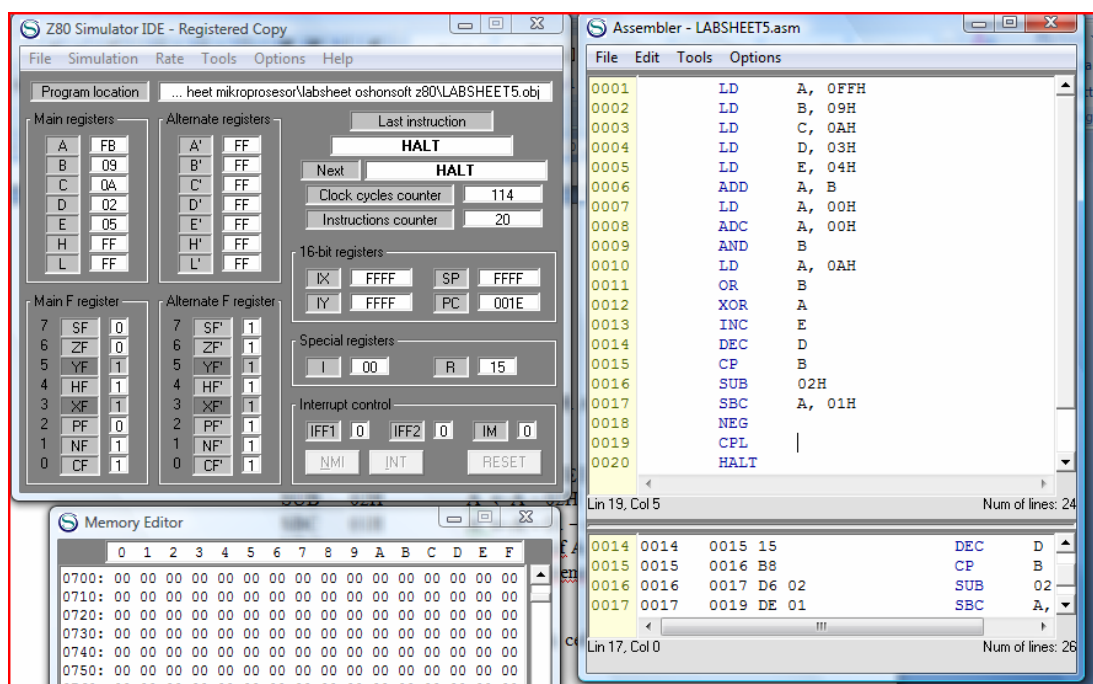
PROGRAM I

1. Jalankan Z80 Simulator IDE
2. Klik Tools / Assembler dan ketik program berikut (tanpa komentar, hanya perintah menmonik) :

LD	A, 0FFH	A ← FFH
LD	B, 09H	B ← 09H
LD	C, 0AH	C ← 0AH
LD	D, 03H	D ← 03H
LD	E, 04H	E ← 04H
ADD	B	A ← A + B
LD	A, 00H	A ← 00H
ADC	A, 00H	A ← A + 00H + cy
AND	B	A ← A AND B
LD	A, 0AH	A ← 0AH
OR	B	A ← A OR B

XOR	A	A ← A XOR A
INC	E	E ← E + 1
DEC	D	D ← D - 1
CP	B	A ← B, A TETAP
SUB	02H	A ← A - 02H
SBC	01H	A ← A - 01 - cy
NEG		A ← negatif A
CPL		A ← komplemen A
HALT		
.END3.		

3. Pada jendela Assembler klik Tools / Assemble untuk cek kesalahan, simpan file ASM ke folder anda masing-masing.
4. Muat program ke Simulator dengan mengklik Tools / Assemble & Load.
5. Jalankan simulator dengan mengklik Simulation / Start
6. Cek hasil tampilan jendela Simulator Z80, lihat nilai-nilai register apakah sudah sesuai dengan yang diisikan dan cek pula pada tampilan file log.txt



Assembler - LABSHEET5.asm

File Edit Tools Options

```

0001      LD      A, 0FFH
0002      LD      B, 09H
0003      LD      C, 0AH
0004      LD      D, 03H
0005      LD      E, 04H
0006      ADD     A, B
0007      LD      A, 00H
0008      ADC     A, 00H
0009      AND     B
0010      LD      A, 0AH
0011      OR      B
0012      XOR     A
0013      INC     E
0014      DEC     D
0015      CP      B
0016      SUB     02H
0017      SBC     A, 01H
0018      NEG
0019      CPL
0020      HALT

```

Lin 19, Col 5 Num of lines: 24

```

0001 0001 0000 3E FF      LD      A, 0FFH
0002 0002 0002 06 09      LD      B, 09H
0003 0003 0004 0E 0A      LD      C, 0AH
0004 0004 0006 16 03      LD      D, 03H
0005 0005 0008 1E 04      LD      E, 04H
0006 0006 000A 80          ADD     A, B
0007 0007 000B 3E 00      LD      A, 00H
0008 0008 000D CE 00      ADC     A, 00H
0009 0009 000F A0          AND     B
0010 0010 0010 3E 0A      LD      A, 0AH
0011 0011 0012 B0          OR      B
0012 0012 0013 AF          XOR     A
0013 0013 0014 1C          INC     E
0014 0014 0015 15          DEC     D
0015 0015 0016 B8          CP      B
0016 0016 0017 D6 02      SUB     02H
0017 0017 0019 DE 01      SBC     A, 01H
0018 0018 001B ED 44      NEG
0019 0019 001D 2F          CPL

```

```

d:\program files (x86)\z80 simulator ide\log.txt
Print Copy
Simulation started at 20/09/2010 5:51:00.
No. PC Instruction A SZYHXPNC' B C D E H L IX IY SP A' SZYHXPNC' B' C' D' E' H' L'
1 0000 LD A,0FFH FF 11111111 FF FF FF FF FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
2 0002 LD B,09H FF 11111111 09 FF FF FF FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
3 0004 LD C,0AH FF 11111111 09 0A FF FF FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
4 0006 LD D,03H FF 11111111 09 0A 03 FF FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
5 0008 LD E,04H FF 11111111 09 0A 03 04 FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
6 000A ADD A,B 08 00111001 09 0A 03 04 FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
7 000B LD A,00H 00 00111001 09 0A 03 04 FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
8 000D ADC A,00H 01 00101000 09 0A 03 04 FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
9 000F AND B 01 00111000 09 0A 03 04 FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
10 0010 LD A,0AH 0A 00111000 09 0A 03 04 FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
11 0012 OR B 0B 00101000 09 0A 03 04 FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
12 0013 XOR A 00 01101100 09 0A 03 04 FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
13 0014 INC E 00 00101000 09 0A 03 05 FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
14 0015 DEC D 00 00101010 09 0A 02 05 FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
15 0016 CP B 00 10111011 09 0A 02 05 FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
16 0017 SUB 02H FE 10111011 09 0A 02 05 FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
17 0019 SBC A,01H FC 10101010 09 0A 02 05 FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
18 001B NEG 04 00111011 09 0A 02 05 FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
19 001D CPL FB 00111011 09 0A 02 05 FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
20 001E HALT FB 00111011 09 0A 02 05 FF FF FFFF FFFF FFFF FF 11111111 FF FF FF FF FF FF
Simulation stopped at 20/09/2010 5:51:02.
1.53 seconds elapsed.
Real-time duration of the simulation is 114 clock cycles or 28.50 µs at 4 MHz.

```

7. Cek isi dari register Flag dan bandingkan dengan teori dasar di atas.
8. Ulangi langkah 1 sampai dengan 6 dengan nilai register A = 0FH, register B = 0AH .