# LAB SHEET 1

# PENGENALAN Z80 SIMULATOR IDE OSHONSOFT

#### A. TUJUAN

- 1. Dapat mengoperasikan program simulator Z80 IDE Oshonsoft dengan baik.
- 2. Dapat melihat/mengecek isi memori pada lintas data untuk setiap program.

#### **B. BAHAN DAN ALAT**

- 1. Lembar tugas
- 2. Sofware Z80 Simulator IDE Oshonsoft

### C. TEORI DASAR

Z80 Simulator IDE adalah applikasi simulator mikroprosesor Z80 yang memudahkan mahasiswa mempelajari mikroprosesor Z80. Simulator ini berbasis grafis yang terintegrasi dengan kompiler BASIC dan Assembler serta dilengkapi fasilitas debugger dan disassembler untuk mikroprosesor Z80. Aplikasi simulator ini mampu menunjukkan isi dari register internal termasuk status flagnya, mnemonic yang telah dieksekusi dan yang akan dieksekusi, nilai clock, instruction counter dan interupt interface.

Secara umum tampilan Z80 Simulator IDE dapat dilihat dalam gambar berikut



Sedangkan tampilan lengkap dengan memory editor dan assembler sebagai berikut :



Hasil eksekusi juga bisa dilihat dalam resume dalam file berekstensi .txt yang akan menjelaskan urutan proses dan perubahan nilai register yang terlibat dalam suatu program aplikasi Z80 yang telah dieksekusi.

S di	\program file	es (x86)\z80 simulator ic	le\log.txt		- 7	-		10000												Σ	٢
Print	Сору																				
∣Simu	ulation s	tarted at 17/09/	/2010 0:58:52.																		
NO.	PC	Instruction	A SZYHXPNC	В	C	D	E H	L	IX	IY	SP	Α'	SZYHXPNC'	в'	с'	D'	Ε'	н'	L' -		
1	0000		CC 11111111	04					FFFF	FFFF	FFFF		11111111								
12	0002	LD TX 1900H	FF 11111111	04	FF	FF	FF F	F FF	1900	FFFF	FFFF	FF	111111111	FF	FF	FF	FF	FF	FF		
3	0006	XOR A	00 01101100	04	FF	FF	FF F	F FF	1900	FFFF	FFFF	FF	111111111	FF	FF	FF	FF	FF	FF		
4	0007	LD A, (IX+00H)	00 01101100	04	FF	FF	FF F	F FF	1900	FFFF	FFFF	FF	11111111	FF	FF	FF	FF	FF	FF		
5	000A	ADC A, (IX+04H)	00 01101000	04	FF	FF	FF F	F FF	1900	FFFF	FFFF	FF	11111111	FF	FF	FF	FF	FF	FF		
6	000D	LD (IX+08H),A	00 01101000	04	FF	FF	FF F	F FF	1900	FFFF	FFFF	FF	11111111	FF	FF	FF	FF	FF	FF		
6	0010	INC IX	00 01101000	04	FF	FF	FF F	F FF	1901	FFFF	FFFF	FF	111111111	FF	FF	FF	FF	FF	FF		
ŏ	0012	JP NZ 0007H	00 00101010	03	EF.	FF.			1001	FFFF	FFFF	쁥			EE.	쁥	EF.	EF.	EF.		
10	0007	D = (TX+00H)	00 00101010	03	FF	FF	FF F		1901	FFFF	FFFF	FF	111111111	FF	FF	FF	FF	FF	FF		
11	000A	ADC A. (IX+04H)	00 01101000	03	FF	FF	FF F	F FF	1901	FFFF	FFFF	FF	111111111	FF	FF	FF	FF	FF	FF		
12	000D	LD (IX+08H),A	00 01101000	03	FF	FF	FF F	F FF	1901	FFFF	FFFF	FF	111111111	FF	FF	FF	FF	FF	FF		
13	0010	INC IX	00 01101000	03	FF	FF	FF F	F FF	1902	FFFF	FFFF	FF	11111111	FF	FF	FF	FF	FF	FF		
14	0012	DEC B	00 00101010	02	FF	FF	FF F	F FF	1902	FFFF	FFFF	FF	111111111	FF	FF	FF	FF	FF	FF		Ξ
15	0013	JP NZ,0007H	00 00101010	02	FF	FF	FF F	F FF	1902	FFFF	FFFF	FF	111111111	FF	FF	FF	FF	FF	FF		
17	0007	LD A, (IX+00H)	00 00101010	02	FF	FF		F FF	1902	FFFF	FFFF	EF.		FF	EF.	EF.	FF	FF.	FF.		
18	0000	ADC A, $(1X+04H)$	00 01101000	02	FF			F FF C CC	1902	FFFF	FFFF	FF	111111111	EE			EE		EE		
19	0010	TNC TX	00 01101000	02	FF	FF	FF F	F FF	1903	FFFF	FFFF	FF	11111111	FF	FF	FF	FF	FF	FF		
20	0012	DEC B	00 00101010	01	FF	FF	FF F	F FF	1903	FFFF	FFFF	FF	111111111	FF	FF	FF	FF	FF	FF		
21	0013	JP NZ,0007H	00 00101010	01	FF	FF	FF F	F FF	1903	FFFF	FFFF	FF	11111111	FF	FF	FF	FF	FF	FF		
22	0007	LD A,(IX+00H)	00 00101010	01	FF	FF	FF F	F FF	1903	FFFF	FFFF	FF	111111111	FF	FF	FF	FF	FF	FF		
23	000A	ADC A, (IX+04H)	00 01101000	01	FF	FF	FF F	F FF	1903	FFFF	FFFF	FF	111111111	FF	FF	FF	FF	FF	FF		
24	0000	LD (IX+08H),A	00 01101000	01	FF	FF	FF F	F FF	1903	FFFF	FFFF	FF	111111111	FF	FF	FF	FF	FF	FF		
25	0010	DEC B	00 01101000	00	FF			F FF C CC	1904	FFFF	FFFF	FF	111111111	FF			EE		FF		
27	0013	JP NZ.0007H	00 01101010	00	FF	FF	FFF	F FF	1904	FFFF	FFFF	FF	11111111	FF	FF	FF	FF	FF	FF		
28	0016	HALT	00 01101010	00	FF	FF	FF F	F FF	1904	FFFF	FFFF	FF	111111111	FF	FF	FF	FF	FF	FF		
Simu	lation s	topped at 17/09/	/2010 0:58:53.																		
0.77	? seconds	elapsed.																			
Rea	I-time du	ration of the si	imulation is 3	53	c100	k c	ycle	s or	88.2	5µs	at 4 I	MHZ.									-
																				Þ	

Penjelasan masing-masing fungsi tombol menu Z80 Simulator IDE dapat dijelaskan sebagai berikut :

- Menu File
  - Clear Memory ; memerintahkan simulator untuk mengembalikan isi memori 64K dari alamat 0000H-FFFFH ke nilai awal (00H).
  - Load Program ; memuat program ke memori CPU. Program file harus dalam format HEX atau binary image (OBJ extension) yang termuat dalam memori dimulai dari alamat 0000H sampai alamat maksimal 64K.
  - Save Memory ; perintah ini untuk menyimpan isi dari memori ke sebuah file.
- Menu Simulation
  - Start ; membuat Z80 Simulator IDE dalam mode simulasi dan memulai mengeksekusi perintah yang dimulai dari alamat 0000H atau alamat awal program lain yang bisa disesuaikan dengan cara mengubah isi perintah Change Starting Address dari menu Option.
  - Step ; Perintah ini dapat digunakan jika rate simulasi dipilih dengan mode Step By Step. Instruksi berikutnya akan dieksekusi tiap penekanan tombol F2 pada keyboard.
  - Stop ; mengakhiri mode simulasi dan menyajikan informasi tentang jumlah total instruksi yang telah dieksekusi dan lama simulasi.
- Menu Rate
  - Memungkinkan pengguna untuk mengubah laju simulasi. Ada beberapa pilihan yaitu : Step By Step, Slow, Normal, Fast, Extremely Fast, dan Ultimate.
- Menu Tools
  - Memory Editor ; untuk mengakses interface grafis ke 64K memori. Untuk mengganti isi memori dilakukan dengan mengklik lokasi memori yang diinginkan kemudian memasukkan nilai bilangan hexadesimal dan diakhiri dengan menekan tombol ENTER pada keyboard, kursor akan otomatis pindah ke lokasi memori satu alamat berikutnya.

- Disassembler ; digunakan untuk memunculkan perintah mnemonic Z80 dari suatu file berektensi HEX atau OBJ. Hasil list perintah berupa file berekstensi LST.
- Peripheral Devices ; digunakan untuk memonitor perintah IN dan OUT. Jumlah I/O yang bisa diatur sampai 4 buah dan satu terminal output untuk melihat karakter ASCII yang dikirimkan ke salah satu port.
- I/O Port Editor ; untuk mengatur isi dari port I/O. Nilai dari port I/O dapat diganti dengan mengklik port yang diinginkan dan setelah nilai dimasukkan diakhiri dengan menekan tombol ENTER keyboard.
- External Modules ; digunakan untuk memantapkan interface otomasi sampai lima modul eksternal client/server,
- Assembler ; file program assembler Z80 dapat langsung diketik, dikompile dan dimuat ke memori simulator dalam satu lingkungan tampilan grafis. Ekstensi file berupa .ASM, ketika proses assembly berhasil dilakukan akan dibangkitkan dua file yaitu file OBJ yang dapat dimuat ke memori CPU dan file LST hasil proses debugger, Sebuah file HEX juga bisa dibangkitkan jika kita memilih Generate HEX File Also dalam tag Option.
- Menu Option
  - Enable Logging ; memungkinkan simulator untuk menampilkan isi dari file LOG.TXT yang menunjukkan perubahan isi register dan status flag Z80 setelah program dieksekusi.
  - HALT Stops Simulation ; memungkinkan menghentikan simulasi secara otomatis jika menemui perintah HALT.
  - FF Power On Defaults ; mengganti nilai awal register Z80 dari nilai 00H ke FFH.
  - Refresh Memory Editor ; memperbaharui isi memori setiap kali perintah simulasi diaktifkan untuk semua nilai laju simulasi.
  - Refresh Breakpoint Manager ; jika pilihan ini diaktifkan dan Breakpoint Manager dimulai, akan memperbaharui breakpoint tiap instruksi simulasi dimulai.

- Save Position ; menetapkan posisi masing-masing jendela tampilan simulator.
- Auto Start Option ; mengatur tampilan dan bagian menu apa saja yang akan ditampilkan saat kita menjalankan pertama kali Z80 Simulator IDE ini juga file yang akan dihasilkan.
- Change Clock Frequency ; mengganti nilai parameter frekuensi untuk menghitung durasi real-time simulasi, nilainya dalam MHz.
- Change Starting Address ; mengganti alamat awal program yang akan disimulasikan, nilai default adalah 0000H.

### D. LANGKAH KERJA

Sebagai latihan lakikan beberapa langkah kegiatan sebagai berikut :

1. Jalankan Z80 Simulator IDE



2. Klik Tools / Assembler dan ketik program coba sebagai berikut, dengan ketentuan setiap mengetik perintah harus didahului dengan menekan tombol Tab pada keyboard untuk menghindari kesalahan tempat penulisan Label :

S Ass	embler - coba1.asm		X
File	Edit Tools Options		
0001	LD A,04H		_ <b>_</b>
0002	LD B,05H		
0003	LD C,06H		
0004	LD (1910H),A		
0005	LD A, (1920H)		
0006	HALT ;halt cpu		
0007	.END		
8000			-
	*		- F
Lin 1, Co	10	Num of	lines: 8

3. Pada jendela Assembler, klik Tools / Assemble untuk mengkompile program dan mengetahui apakah ada kesalahan yang terjadi.

	S Ass	embler - o	oba1.asm	1	
	File	Edit Too	ls Optio	ons	
	0001		LD A,	04H	<b>_</b>
l	0002		LD B,	05H	
l	0003		LD C,	06H	
J	0004		LD (1	L910H),A	
l	0005		LD A,	(1920H)	
l	0006		HALT	;halt cp	u
	0007		.END		
l	8000				<u> </u>
l		*			
1					
	Lin 1, Co	0 10			Num of lines: 8
	Lin 1, Co	0003	0004 (	)F 06	Num of lines: 8
	Lin 1, Co 0003 0004	0003	0004 0	DE 06	Num of lines: 8
	Lin 1, Co 0003 0004 0005	0003 0004 0005	0004 0 0006 3 0009 3	DE 06 32 10 19 3A 20 19	Num of lines: 8
	Lin 1, Co 0003 0004 0005 0006	0003 0004 0005 0006	0004 0 0006 3 0009 3 000C 7	DE 06 32 10 19 3A 20 19 76	Num of lines: 8
	Lin 1, Co 0003 0004 0005 0006 0007	0003 0004 0005 0006 0007	0004 0 0006 3 0009 3 000C 7 000D	DE 06 32 10 19 3A 20 19 76	Num of lines: 8
	Lin 1, Co 0003 0004 0005 0006 0007 0008	0003 0004 0005 0006 0007 Number	0004 0 0006 3 0009 3 000C 7 000D of err	DE 06 32 10 19 3A 20 19 76 cors = 0	Num of lines: 8
	Lin 1, Co 0003 0004 0005 0006 0007 0008 0009	0003 0004 0005 0006 0007 Number	0004 0 0006 3 0009 3 000C 7 000D of err	DE 06 32 10 19 3A 20 19 76 cors = 0	Num of lines: 8
	Lin 1, Co 0003 0004 0005 0006 0007 0008 0009	0003 0004 0005 0006 0007 Number	0004 0 0006 3 0009 3 000C 7 000D of err	DE 06 32 10 19 3A 20 19 76 cors = 0	Num of lines: 8
	Lin 1, Co 0003 0004 0005 0006 0007 0008 0009 Lin 9, Co	0003 0004 0005 0006 0007 Number	0004 0 0006 3 0009 3 000C 7 000D of err	DE 06 32 10 19 3A 20 19 76 rors = 0	Num of lines: 8

Dapat dilihat dalam tampilan jendela assembler, jendela atas menampilkan mnemonik Z80 yang kita ketik dan jendela bawah menampilkan hasil kompile yang menunjukkan opcode dari perintah yang ada di atas ( bisa dibandingkan dengan tabel mnemonik Z80) dan jumlah kesalahan yang ditemukan.

4. Pada jendela Assembler klik Tools / Assemble & Load untuk memuat program ke simulator, hasilnya dilihat pada jendela Z80 Simulator IDE nampak ada perubahan pada bagian program location

S Z80 Simulator ID	DE - Registered Copy	
File Simulation	Rate Tools Opti	ons Help
Program location	D:\Program Fi	iles (x86)\Z80 Simulator IDE\coba1.obj
Main registers	Alternate registers	Last instruction
A FF B FF D FF E FF H FF L FF	A' FF B' FF D' FF E' FF L' FF L' FF	Next Clock cycles counter 0 Instructions counter 0 16-bit registers X FFFF SP FFFF
Main F register       7     SF     1       6     ZF     1       5     YF     1       4     HF     1       3     XF     1       2     PF     1       1     NF     1       0     CF     1	Alternate F register       7     SF'     1       6     ZF'     1       5     YF'     1       4     HF'     1       3     XF'     1       2     PF'     1       1     NF'     1       0     CF'     1	IY FFFF PC 00000   Special registers I 00 R 00   Interrupt control IFF1 0 IFF2 0 IM 0   NMI INT RESET

5. Pada jendela Z80 Simulator IDE klik Tools / Memory Editor, scroll ke bawah sampai tampil lokasi memori 1920H. Klik lokasi tersebut, masukkan data AA dan diakhiri menekan ENTER.

🚫 Mei	moŋ	y Ed	itor						
	0	1	2	3	4	5	6	7	8
18A0:	00	00	00	00	00	00	00	00	00
18B0:	00	00	00	00	00	00	00	00	00
18C0:	00	00	00	00	00	00	00	00	00
18D0:	00	00	00	00	00	00	00	00	00
18E0:	00	00	00	00	00	00	00	00	00
18F0:	00	00	00	00	00	00	00	00	00
1900:	00	00	00	00	00	00	00	00	00
1910:	00	00	00	00	00	00	00	00	00
1920:	AA		00	00	00	00	00	00	00
1930:	00	00	00	00	00	00	00	00	00

6. Cek pilihan Rate adalah Fast dan Enable Logging pada menu Option telah dipilih.

7. Jalankan simulasi dengan mengklik Simulation / Start atau menekan F1

8. Amati perubahan di alamat 1910H dari bernilai 00H menjadi 04H dan akan tampil jendela baru log.txt yang menampilkan resume semua langkah dan hasil simulasi berupa perubahan nilai register dan status flag.

18E0:	00	00	00
18F0:	00	00	00
1900:	00	00	00
1910:	04	00	00
1920:	AA	00	00
1930:	00	00	00

e	) d:\p	orogram file	es (x86)\z80 si	mulator ide	\log.	.txt																		×
F	Print	Сору																						
	5imu]	lation s	tarted at	17/09/2	2010	3:26:38.																		
Ň	10.	PC	Instruct	ion	A	SZYHXPNC	в	C	D	Е	н	L	IX	IY	SP	Α'	SZYHXPNC'	в'	c'	D'	Ε'	н'	L'	
1	L	0000	LD A,04H		04	111111111	FF	FF	FF	FF	FF	FF	FFFF	FFFF	FFFF	FF	111111111	FF	FF	FF	FF	FF	FF	
2	2	0002	LD B,05H		04	11111111	05	FF	FF	FF	FF	FF	FFFF	FFFF	FFFF	FF	11111111	FF	FF	FF	FF	FF	FF	
3	3	0004	LD C,06H		04	11111111	05	06	FF	FF	FF	FF	FFFF	FFFF	FFFF	FF	11111111	FF	FF	FF	FF	FF	FF	
4	Ļ	0006	LD (1910	н),А	04	11111111	05	06	FF	FF	FF	FF	FFFF	FFFF	FFFF	FF	11111111	FF	FF	FF	FF	FF	FF	
5	5	0009	LD A,(19	20H)	AA	11111111	05	06	FF	FF	FF	FF	FFFF	FFFF	FFFF	FF	11111111	FF	FF	FF	FF	FF	FF	
6	5	000C	HALT		AA	11111111	05	06	FF	FF	FF	FF	FFFF	FFFF	FFFF	FF	11111111	FF	FF	FF	FF	FF	FF	
S	5imu]	lation s	topped at	17/09/2	2010	0 3:26:38.																		
0	).15	seconds	elapsed.																					
R	teal-	-time du	ration of	the sim	nula	ation is 5	1 c	loc	k cy	/c10	es (	or 1	L2.75	µs at	t4MH	١z.								