

METODA PENYUSUNAN TABEL PERUBAHAN REGISTER DAN PENGGUNAAN IDE SEBAGAI ALTERNATIP PENGGANTI DEVELOPMENT BOARD DALAM USAHA PEMAHAMAN MIKROPROSESOR

Zaenal Husin^{1*}

¹Jurusan Teknik Elektro Universitas Sriwijaya,
*E-mail : zaenalhusin@gmail.com

Abstrak—Dengan perkembangan yang sangat pesat pada fabrikasi Cip prosesor dan memori, baik secara fisik, kapasitas, kinerja dan harga. Fungsi Mikropeoseor telah digantikan oleh MikroKontroler yang merupakan suatu produk integrasi Mikroprosesor, memori dan rangkaian input/output dalam satu Cip saja. MikroKontroler telah dipergunakan secara luas dalam rangkaian-rangkaian sistem tertanam (embedded system), jauh telah meninggalkan pemakaian Cip Mikroprosesor. Karena Cip Mikrokontroler telah dapat diproduksi dengan harga yang sangat ekonomis dibandingkan biaya produksi Cip Mikroprosesor. Mikrokontroler mempergunakan bahasa pemrograman yang dikategorikan sebagai tingkat menengah dan tinggi. Berbeda dengan Mikroprosesor yang masih mempergunakan bahasa tingkat rendah (Assembly), yang berinteraksi langsung dengan komponen-komponen yang berada dalam Cip tersebut. Interaksi secara langsung ini sangat diperlukan untuk menguasai pengetahuan Arsitektur Komputer Digital. Dengan semakin langkanya Cip mikroprosesor, maka keberadaan Development Board menjadi ikut langka juga. Sehingga diperlukan alternatif penggantinya sebagai sarana pemahaman perangkat digital pada Mikroprosesor. Implementasi Metoda penyusunan tabel perubahan register dan penggunaan IDE (Integerated Development Environment) telah dapat berfungsi sebagai alternatif penggantian Z80 Development Board dengan hasil yang baik.

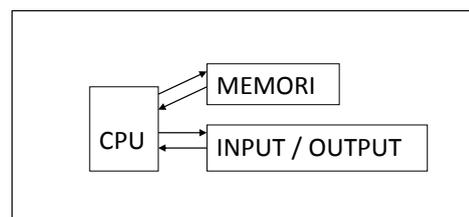
Kata kunci: Mikroprosesor, Mikrokontroler, Assembly, bahasa pemrograman tingkat menengah dan tinggi .

Abstract—We have seen astronomical development on fabrication of processor and memory chips, in the aspect of dimension, capacity, performace and price since last few years. Functions of Microperoccesor have been replaced by MicroController which is a product of integration of Microprocessor, memory and input/output circuits in one chip only. MicroControllers have been used widely at embedded systems, and we may say the use of Microprocesor Chips has been taken over, due to the economical production cost of MicroController Chips. One drawback in studying Computer Architecture, Microcontrollers are programmed in middle and high level of programming languages. And Microprocesors are programmed in low level languages (Assembly), which interacts directly to components inside the chips. These direct interaction is essential to get the grip the knowledge of Digital Computer Architecture. With the diminishing of Microprocessor chips, makes Microprocessor Development Board imminently extinct too. Therefore replacement of such Board's Function is needed as a mean to understand the Microprocessor internal circuits. The Implementation of Table of Changings of Registers' Contents Methode and the use of IDE (Integerated Development Environment) have been proven as alternative replacement of diminishing Z80 Development Boards with good results.

Keywords. Mikroprocessor, Microcontroller, Assembly, Medium and High level pemogramming language

I. PENDAHULUAN

Suatu Komputer digital terdiri atas 3 bagian utama, yang dapat dijelaskan dengan gambar 1.



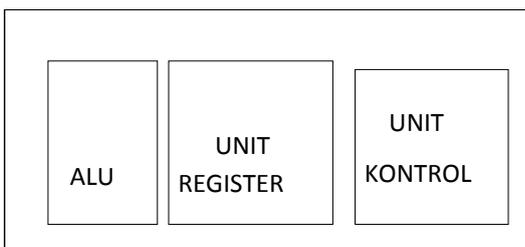
Gambar 1. Blok Diagram Suatu Komputer

Komputer dirancang untuk suatu pemakaian umum, yang berarti aplikasi-nya dapat diubah-ubah hanya dengan merubah program yang disimpan dalam Memori.

A. CPU atau yang juga dinamakan MikroProsesor

CPU (Central Processing Unit) dapat menghubungkan Memori dan sebaliknya. Demikian juga antara CPU dan INPUT/OUTPUT adalah hubungan dua arah. Sedangkan antara Memory dan Input/Output apabila terhubung secara langsung maka dinamakan DMA (Direct Memory Access).

Dalam setiap dekade, telah terjadi perubahan yang begitu cepat pada ketiga bagian tersebut, CPU bertambah cepat, memori telah berlipat kapasitasnya demikian juga kecepatan akses-nya dan bagian input telah berubah dengan bertambah mudahnya memasukkan dan mengeluarkan data dan aplikasi melalui USB misalnya. Sedangkan CPU sendiri dapat dibagi ke tiga bagian:



Gambar 2. Blok Diagram suatu CPU

ALU (Arithmetic Logical Unit) adalah bagian yang melaksanakan operasi Arithmetik seperti penambahan, pengurangan dan sebagainya beserta operasi logika seperti OR, AND dan sebagainya. Unit Register merupakan pencatatan data dan instruksi, penghitung program dan sebagainya. Sedangkan Unit Kontrol mengatur aliran data dan instruksi.

Dalam perkembangannya, beberapa jenis CPU telah di-integrasikan dengan memori dan input/output membentuk suatu mikro-kontroler yang diaplikasikan ke pengaturan peralatan perumahan dan industri.

Akhir-akhir ini, selain integrasi dengan input/output, masih juga diintegrasikan dengan memori, sehingga terjadilah sebuah Prosesor Aplikasi (Application Processor) yang banyak diterapkan pada telepon pintar.

B. Memori

Bagian yang berfungsi sebagai pencatat (register) atau penyimpan. Dalam blok CPU terdapat memori yang dinamakan Register dan Cache. Sedangkan diluar blok CPU terdapat memori yang dinamakan RAM (Memori Utama) yang volatile dan Memori sekunder yang non Volatile

C. INPUT/OUTPUT

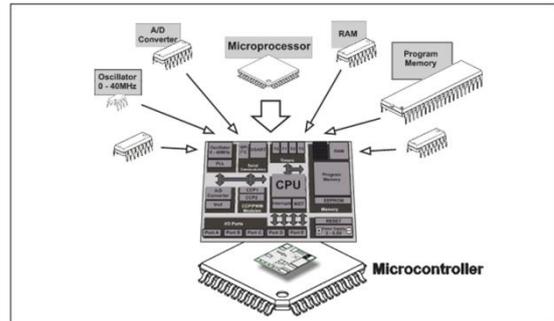
Bagian ini merupakan penghubung antara komputer dengan dunia nyata, yang sering kita jumpai adalah keyboard sebagai alat penginput ke komputer sedangkan output atau hasil olahan komputer ditunjukkan pada layar monitor atau pencetak.

Masih banyak alat lain yang sifatnya khusus, seperti sensor dan camera sebagai input dan relay pada output misalnya. Yang terhubung kekomputer melalui suatu antar

muka (interface) yang diprogram dengan metoda tertentu dan diberi alamat spesifik pula.

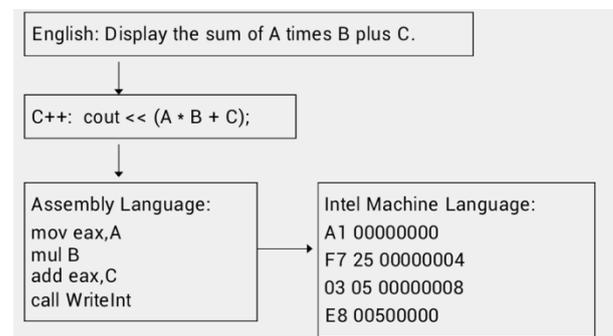
D. Perbandingan MikroProsesor dan MikroKontroler

Mikrokontroler telah dilengkapi dengan Memori danInterface Input Output, yang dapat ditampilkan pada gambar 3.



Gambar 3. Perbandingan antara Mikroprosesor dan Mikrokontroler

Sedangkan perbandingan bahasa pemrogramannya adalah:

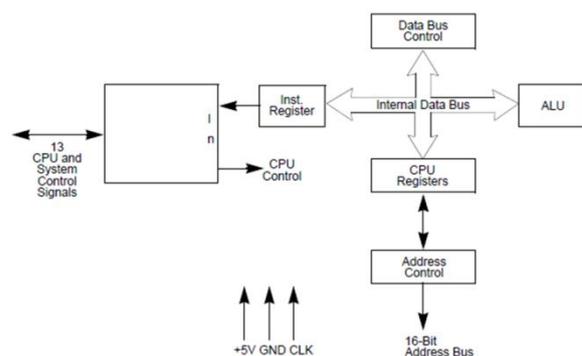


II. TINJAUAN PUSTAKA

A. Arsitektur and operasi dasar mikroprosesor Z80

Gambar 4 berikut menunjukkan arsitektur internal sebuah mikroprosesor Z80 (untuk selanjutnya disebut Z80). Z80 secara garis besar terdiri atas 3 blok yaitu:

- REGISTER UNIT,
- ARITHMATIC LOGIC UNIT (ALU) dan
- CONTROL UNIT

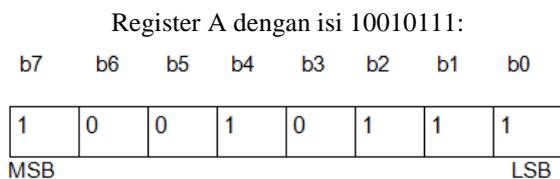


Gambar 4. Bagian-bagian CPU Z80 [1]

Suatu Register adalah rangkaian yang mengingat informasi digital secara sementara, register dipergunakan untuk menyimpan data input yang ditujukan ke ALU dan menyimpan hasil olahan. Register dapat mengambil dan menyampaikan data dari/ke IC MEMORY yang terhubung ke prosesor dan juga dari/ke Port Input/Output. Z80 dilengkapi dengan 7 Register Pemakaian Umum (General Purpose) 8 bit: A, B, C, D, E, H dan L. Register register ini juga dapat dipergunakan sebagai kombinasi pasangan seperti: BC, DE, dan HL untuk menyimpan data 16 bit.

Ada 2 jenis Register, yaitu A, B, C dan sebagainya disebut Register Utama, terdapat juga Register Pengganti (Alternate) A', B' dan C' dstnya. Pemakaian sekaligus kedua jenis register ini tidak diperkenankan, tetapi hanya dapat dipindahkan melalui suatu instruksi.

Gambar 5. menunjukkan keadaan dimana data disimpan dalam Register A. Order paling tinggi disebut MSB (Most Significant Bit) atau b7 sedangkan yang terendah disebut LSB (Least Significant Bit) atau b0



Gambar 5. Register A

TABEL 1
ISI REGISTER F

b7	b6	b5	b4	b3	b2	b1	b0
S	Z		H		P/V		CY

FLAG	Bit flag berikut akan di-set ke "1" sebagai hasil operasi
S	Kalau MSB (BIT 7) = 1
Z	Kalau semua bit adalah nol
H (Half Carry)	Kalau terjadi carry keluar dari b3
P/V (Parity/OverFlow)	Pada operasi logika, jumlah 1 adalah genap
	Pada operasi arimatik, hasil terlalu besar dari yang diperkenankan
CY (Carry)	Pada operasi Carry keluar dari MSB (b7)

ALU merupakan suatu Unit yang melaksanakan operasi Arithmetic dan Logic. Seperti terlihat pada gambar 3, transfer data diantara register register dan ALU dilaksanakan melalui 1 internal bus (one bus method architecture).

Dalam pelaksanaan operasi Arithmetic atau Logic, data dari salah satu dari 6 General Purpose (B, C, D, E, H dan L) di transfer ke ALU melalui internal bus, sedangkan data lainnya ditransfer ke ALU setelah disimpan di Register A sebelum dilaksanakan. Untuk itu Register A disebut juga AKUMULATOR.

Z80 hanya dapat memproses data 8 bit pada waktu yang sama, dinamakan prosesor 8 bit. Terdapat juga prosesor-prosesor 1, 4, 8,16, 32 dan 64 bit. Pada setiap pelaksanaan proses dalam suatu prosesor, suatu instruksi

dalam kode binary (machine instruction, machine code) didefinisikan. Suatu koleksi machine instruction yang berhubungan disebut PROGRAM.

Control Unit menjemput suatu instruksi yang sudah berbentuk kode mesin dari memori, and mengeluarkan sinyal-sinyal yang sesuai untuk melaksanakan instruksi itu, Sebagian dari sinyal itu di transfer ke ALU dan instruksi itu berisi operasi apa yang akan dilaksanakan.

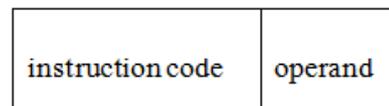
Bus (jalur) data pada mikroprosesor Z80 selebar 8 bit, menghubungkan ketiga bagian utama dan eksternal melalui suatu buffer. Hal ini dapat terlihat jelas pada 8 buah pin pada fisik cip ini.

Sedangkan bus alamat (address bus) yang menghubungkan bagian register dan bagian control unit terdiri atas 16 bit. sebagai contoh lokasi beralamat 8000H atau 1000000000000000 B, bus alamat ini juga dihubungkan ke eksternal dari cip melalui suatu buffer.

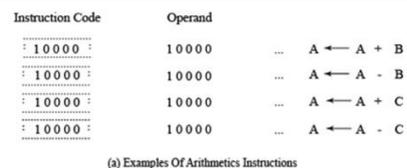
B. Format Instruksi dari Z80

Suatu OPCODE terdiri atas :

1. Bagian Kode Instruksi (Instruction Code)
2. Bagian Objek (Operand)



Operand dapat saja berupa : angka, character, Isi Register, alamat yang ditunjuk oleh isi register. Beberapa contoh OPCODE mengenai penambahan dan pengurangan arimatik:



Gambar 6. Menunjukkan contoh Instruksi arimatik dari suatu Z80 [2]

Pada bagian kiri yaitu kode instruksi (a):
 10000 ----- penjumlahan ke register A
 10010 ----- pengurangan dari register A
 3 bit kekanan menunjukkan objek dari operasi tersebut:
 10000000 ----- pejumlahan isi register A dan isi register B
 10010001 ----- pengurangan isi register A – isi register C

Sedangkan (b) menunjukkan contoh

- data transfer dari memory ke register A, kode instruksinya 00111010

- data transfer dari register A ke memory, kode instruksinya 00110010 Singkatan dan Akronim

C. Bahasa ASSEMBLY

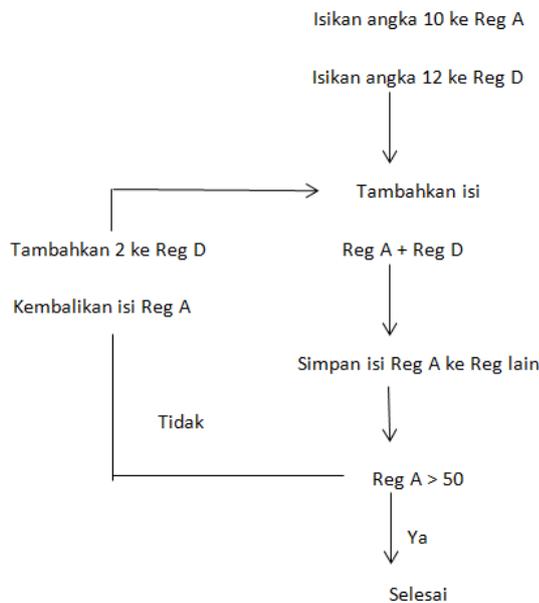
Operasi Arimatik:

Melakukan operasi aritmatik penambahan dan pengurangan Register A disebut juga akumulator harus dipakai sebagai yang ditambah, sedangkan penambah dapat berasal dari isi Register lain. Instruksi dapat berupa: ADD A, D (penambahan isi register A dengan isi register D), ADD A, 100 (penambahan angka secara langsung ke register A) atau ADD A,(HL), suatu penambahan isi memory yang alamatnya tersimpan dalam suatu pasangan Register HL ke register A

D. Metoda Penyusunan Tabel Perubahan Register

Contoh : Buatlah suatu penambahan angka 10 dengan angka yang berturut-turut tersimpan sebagai suatu deret 12, 14, 16,...sampai dengan jumlah penambahan > angka 50.

FLOWCHART



Jawab :	LD A, 10	; Angka yang ditambah ke Reg A
	LD D, 12	; Angka penambah 1 ke Reg D
LOOP	ADD A, D	; Operasi penambahan
	LD E, A	; Bila Reg A disalin ke Reg E
	SUB A, 50	; Untuk mengetahui isi Reg A
	JP P, SELESAI	; Kalau lebih besar dari 50 selesai
	LD A, E	; Kalau <50, kembalikan isi Reg A
	INC D	; Naikkan 1 isi Reg D
	INC D	; Naikkan 1 isi Reg D
	JP LOOP	
SELESAI	HALT	

Pembuktian program tersebut berjalan sesuai dengan permintaan dapat dilakukan dengan menyusun Tabel perubahan isi Register-Register yang dipergunakan yaitu Reg A, D, E dan Status Flag.

TABEL 2
PERUBAHAN REGISTER-REGISTER SETELAH PROGRAM DIAJALANKAN (RUN)

	Reg A	Reg D	Reg E	Status Flag P
LD A, 10	10	-	-	-
LD D, 12	10	12	-	-
ADD A, D	22	12	-	-
LD E, A	22	12	22	-
SUB A, 50	-28	12	22	0
LD A, E	22	12	22	-
INC D	22	13	22	-
INC D	22	14	22	-
ADD A, D	36	14	22	-
LD E, A	36	14	36	-
SUB A, 50	-14	14	36	0
LD A, E	36	14	36	-
INC D	36	15	36	-
INC D	36	16	36	-
ADD A, E	52	16	36	-
LD E, A	52	16	52	-
SUB A, 50	2	16	52	1
HALT				

E. Pemakaian Z80 Simulator IDE versi 9.8.1

Simulator ini mempunyai fasilitas antara lain:

- Z80 assembler, beserta editornya dan disassembler
- Breakpoint manager, sebagai code debugging
- Memori editor
- I/O ports editor dan peripheral devices interface untuk simulasi instruksi I/O

Dengan simulator z80, dapat dilakukan simulasi program-program yang lebih panjang daripada kedua contoh diatas. Fasilitas 'break point' dapat dipergunakan untuk 'debugging' apabila terjadi Error. Adanya Assembler juga dapat melatih mahasiswa untuk tidak membuat kesalahan-kesalahan 'syntax'.

III. METODE PENELITIAN

Penelitian ini dilakukan di Ruang kelas dan Laboratorium Kontrol Digital Jurusan Teknik Elektro Fakultas Teknik, Universitas Sriwijaya dengan langkah yang tersusun sebagai berikut:

A. Studi literatur dan referensi

Pengumpulan bahan-bahan berupa referensi tentang teori dan manual/datasheet dari perangkat keras yang akan dipergunakan dari sumber-sumber seperti internet, dan

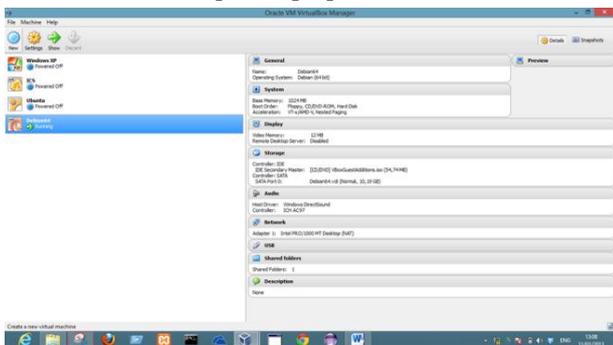
pemesanan buku yang akan dipergunakan sebagai penunjang penelitian ini.

B. Perancangan

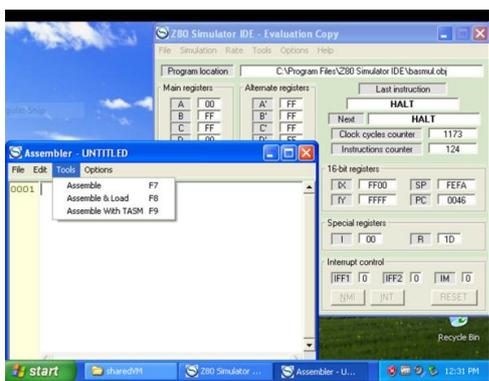
Perancangan penelitian dilakukan setelah mengumpulkan referensi mengenai Mikroprosesor dan IDE.

C. Persiapan

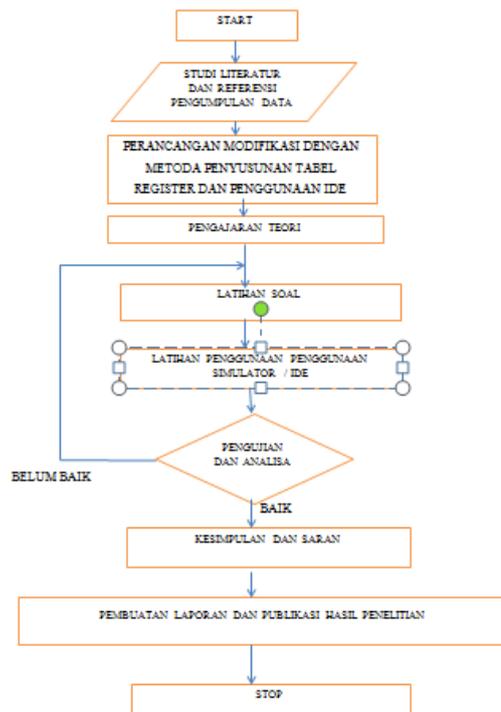
Persiapan sarana penelitian penggunaan IDE yang meliputi Instalasi Sistem Operasi yang sesuai pada suatu laptop. Ada beberapa cara instalasi, tetapi pilihan jatuh pada pemakaian Oracle Virtualbox dengan beberapa alasan antara lain: a. Pemanfaatan Mesin Virtual menghindarkan partisi dengan format ext secara fisik pada hard disk (hdd) laptop, b. adanya dukungan perangkat keras yang memadai, yaitu laptop dengan prosesor Intel i7 serta RAM 8 GB, sehingga program mesin virtual dan program tumpangannya dapat berjalan dengan lancar (tidak tersendat-sendat) dengan spesifikasi tersebut, c. Tersedianya dukungan perangkat lunak dari Oracle berupa program mesin virtual VirtualBox beserta Extension Pack sehingga dapat memanfaatkan pemakaian folder bersama dengan host (Windows 8). Sistem Operasi XP yang ditumpangkan ke Mesin Virtual disebut guest. Pemakaian Virtual Machine ini sekaligus menyesuaikan sistem 32 bit IDE pada laptop 64 bit



Gambar 7. Suatu mesin virtual dapat menumpangkan beberapa Sistem Operasi guest



Gambar 8. Menunjukkan Simulator z80 dari OshonSoft yang dapat diunduh dari OshonSoft.com. [3]



Gambar 9. Diagram alir penelitian

IV. HASIL DAN PEMBAHASAN

Pengujian yang dilakukan meliputi:

- Keberhasilan pengerjaan soal latihan oleh mahasiswa baik secara bersama maupun secara individu di depan kelas,
- Keberhasilan memanfaatkan IDE dalam melakukan penulisan program, assembly dengan IDE.

TABEL 3

PERBANDINGAN ANTARA HASIL YANG DICAPAI MAHASISWA DALAM PENYERAPAN KEDUA METODA

Mahasiswa yang	Hasil pengerjaan soal / Tugas	Hasil Ujian Akhir Semester
Hanya mengikuti Metoda Perubahan Register -register	Baik	Cukup dan Baik
Mengikuti Metoda Perubahan Register – register DAN z80 Simulator	Baik	Sangat Baik

Mahasiswa – mahasiswa yang aktif mempergunakan IDE / simulator akan mendapatkan nilai ujian akhir yang melebihi daripada yang tidak mempergunakan. Hal ini disebabkan IDE dapat menunjukkan adanya kesalahan pada penulisan program dan hasil akhirnya yang ‘online’.

V. KESIMPULAN DAN SARAN

1. Kedua metoda, yaitu Penyusunan Tabel perubahan Register –register dan pemakaian simulator /IDE

apabila digabung, telah dapat memenuhi kebutuhan dalam penguasaan dasar Mikroprosesor.

2. Masih terdapat kendala, yaitu tidak semua mahasiswa dapat melakukan simulasi dengan simulator / IDE karena keterbatasan pemilikan laptop dan software
3. Sebagai saran, pemenuhan sarana pembelajaran tersebut dapat memberikan dampak yang sangat positif

DAFTAR PUSTAKA

- [1] Zilog Publication – Z80 CPU User's Manual,
- [2] Sendai University -Fundamental Programming and Interrupt
- [3] <http://www.oshonsoft.com/z80.html> -Z80 Simulator IDE version 9.8.1