MICROPROCESSOR AND MICRO-CONTROLLER LABORATORY

CODE : - AEIE 3115

LABORATORY MANUAL & ASSIGNMENTS

HERITAGE INSTITUTE OF TECHNOLOGY
KOLKATA

PROCEDURE FOR ENTERING USER PROGRAM

Execution of program through keyboard

1. CONNECT THE 5-VOLT POWER SUPPLY FROM THE SMPS TO THE MICROPROCESSOR KIT.

2. THE WORD '**SDA 85'** SHOULD BE DISPLAYED IN THE INTEGRAL DISPLAY OF THE KIT AFTER POWER IS SWITCHED ON. IF NOT , PRESS 'RESET' KEY.

3. TO WRITE THE PROGRAM IN THE KIT FOLLOW THE INSTRUCTIONS GIVEN BELOW: -

>PRESS 'SUBST MEM' KEY
>ENTER THE INITIAL ADDRESS
>PRESS 'NEXT' KEY
>ENTER THE HEX CODE
> PRESS 'NEXT' KEY
>ENTER THE NEXT HEX CODE
> PRESS 'NEXT' KEY
>PRESS 'RESET' KEY

CONTINUE IN THIS WAY TILL ALL THE HEX CODES HAVE BEEN ENTERED. DO NOT FORGET TO PRESS THE 'NEXT' KEY AFTER THE LAST HEX CODE HAS BEEN ENTERED.

4. TO EXECUTE THE PROGRAM FOLLOW THE INSTRUCTIONS GIVEN BELOW: -

>PRESS 'GO' KEY
>ENTER INITIAL ADDRESS
>PRESS 'EXEC' KEY

5. TO VIEW THE CONTENTS OF ACCUMULATOR (OR ANY OTHER REGISTER B, C, D, E, H, L): -

>PRESS 'EXAM REG' after that 'Reg A' KEY (OR 'Reg B', 'Reg C', 'Reg D', 'Reg E', 'Reg H, 'Reg L' KEY)

6. TO VIEW THE CONTENTS OF ANY MEMORY LOCATION: -

>PRESS 'SUBST MEM' KEY
>ENTER THE ADDRESS OF THE MEMORY LOCATION
>PRESS 'NEXT' KEY

Execution of program through serial communication (PC COM port)

INITIAL CHECKING:

1) TALK □ OPTION □Target Board □ 8085 KIT
2) TALK □ Properties □ DRT enable

START TO MATCH BAUD RATE AND ACTIVATE COMMAND PROMPT (>):

1) On 8085 KIT; sub.mem. FFA6H= 0AH & FFA7 = 00H

2) CRT 0

START USING OF TALK AND PROGAMMING:

1) To show the list of commands Type **H**

2) To assemble a program type **A**, it ask for program starting address put address then write program as follows and press <**Esc**> at the end of the program.

Eg. starting address: 9100 <press enter>

Line	**Source Code**	**Loc**	**OBJ**
0001	MVI A, 02	9100	3E, 02
0002	RST 1	9102	CF

3) To disassemble a program type **Z**, it ask for program starting address put address then it shows whole program as follows

Eg. starting address: 9100 <press enter>
ending address: 9106 <press enter>

Line	Source Code	Loc	OBJ
0001	MVI A, 02	9100	3E, 02
0002	RST 1	9102	CF

4) To execute a program type **G**, then put
starting address: 9100 <press enter>

5) To see the register content type X, it ask for register name, put required register name
Eg. Register: A <press spacebar> B <press spacebar>…….<Esc>

6) To load data in memory address, type **M** to modify, it ask for starting address put address then press <space> followed by data.
Eg. program starting address: 9101 < press spacebar> 02 – 04< press spacebar>……. <Esc>

7) To insert data in memory address, type **I**, it ask for address followed by data.
starting address of user program:
ending address of user program:
Data to be inserted at address:
No. of byte: 2<press,> or <press spacebar> 02 – 04<press space>……. <Esc>

8) To delete data in memory address, type **K**, it ask for address followed by data.
starting address of user program:
ending address of user program:
starting address:
ending address:
02 –<Esc>

CONFIGURATION OF SDA-85-ME TRAINER

SYSTEM SPECIFICATION:

❖ **CPU**	:	8085 operating at 3.072MHZ
❖ **MEMORY**	:	Total 256KB of memory is provided in the kit
EPROM	:	Two JEDEC compatible 28 pin sockets provide up to 16/32K bytes using 2x27128/27256
RAM	:	1 JEDEC compatible sockets are provided for 8/32KRAM. 8KB of RAM will be will be supplied
❖ **PARALLEL I/O**	:	48 I/O lines using two 8255s
❖ **SERIAL I/O**	:	One RS 232C compatible interface using USART 8251A, with programmable baud rate using one channel of 8253 timer with MAX 232C IC.
❖ **TIMER**	:	Three 16 bit counter/timers using 8253 timer. 1 counter is used for serial I/O baud rate generation.
❖ **PIC**	:	Programming Interrupt controller using 8259A provides interrupt vectors for 8 jumper selectable Internal/external sources
❖ **KEYBOARD**	:	Consists of 28 numbers of computer grade keys or equivalent
❖ **DISPLAY**	:	Six numbers of bright seven segment displays for address and data display
❖ **INTERRUPTS** **NMI**	: :	Provision for connecting NMI to a key switch.
INTR	:	Programmable Interrupt controller using 8259A provides interrupt vectors for 8 jumper selectable internal/external sources. Onboard sources include 8251A,TXRDY and RXRDY,8255 and 8087
❖ **INTERFACE BUS SIGNALS** **CPU BUS**	:	All address, data and control lines are TTL compatible and are terminated in 50 pin berg strip header.
PARALLEL I/O	:	All signals are TTL compatible and terminated in 26 pin berg strip headers for PPI expansion. It's compatible with all of our experimental interface modules
SERIAL I/O	:	Serial port signals are terminated in standard 9-pin 'D' type connector.
❖ **MONITOR SOFTWARE**	:	16k bytes monitor(27128A) that allows program entry verify and debug user programs including onboard Assembler and Disassembler commands
❖ **POWER REQUIREMENTS**	:	+5V DC with 2.5 Amps current rating (max)

INSTALLATION PROCEDURE OF SDA-85-ME TRAINER

Power Supply Requirements:

The trainer requires a DC regulated power supply with +5V at 1.5Amps outputs for its basic operation. The power connections are made through 9-pin D type connector.

Pin9	**-**	**+5V**	**Orange/Blue/White**
Pin 4,5	-	GND	Black/yellow

Keyboard Operation

As soon as +5V and GND are connected and power supply is switched on, the sign on message **" SDA 85"** should appear on the display**.** The labeling of the keys clearly indicates their function.

KEY LABEL	DESCRIPTION
RESET	Transfer control to the monitor at location 0H key is connected to the RESET input pin
VECT	A user interrupt key connected to the RST 7.5 input of the CPU. transfer control to location FFB1 in RAM if the interrupt is unmasked and interrupts are enabled
NEXT	The monitor interrupts this key as a delimiter
EXEC	The command terminator
BLOCK MOVE	Select the block move command
EXAM REG	Select the Examine/Modify CPU register
GO	Select the GO command(prog
INS	Select the Insert data bytes function
PREV	A delimiter key
SINGLE STEP	Select the single step function STEP
SUBSET MEMORY	Select memory Examine/Modify function
DEL	Select the delete data bytes function
C COMP	Hex key C Select the memory complement command
8 H	Hex key 8 The content s of the H register to be displayed
4 SPH	Hex key 4 The content of high byte stack pointer to be displayed
0	Hex key 0
D	Hex key D The content s of the D register to be displayed
9 L	Hex key 9 The content s of the L register to be displayed
5 SPL	Hex key 5 The content s of the low stack pointer register to be displayed
E CRT	HEX E, Select the CRT mode when pressed after Reset
A	Hex key A
6 PCH	Hex key 6 The high byte of the program counter to be displayed
2	Hex key 2
F FILL	Hex key F. Select the memory fill Command
B	Hex key B

MEMORY & I/O MAPPING CONFIGURATION OF SDA- 85-ME TRAINER

Memory Configuration

JUMPER PIN	**27128(U29)**	**27128(U27)**	**6264(U22)**
JP14 OPEN	0-3FFFH	4000-7FFFH	E000-FFFFH
JP15 CLOSE(2-3)			
JP12 CLOSE			
JUMPER PIN	27256(U29)		62256(U22)
JP14 CLOSE	0-7FFFH		8000 -FFFFH
JP15 CLOSE(1-2)			
JP12 OPEN			

RAM configuration

I/O Device	**I/O Address**
TIMER 8253(U11)	C8 Timer 0 C9 Timer 1 CA Timer 2 CB CONTROL
KEYBOARD DISPLAY CONTROLLER 8279 (U5)	D0 DATA D1 CONTROL
PPI, 8255 (U4)	D8 PORT A D9 PORT B DA PORT C DB CONTROL
PPI, 8255 (U3)	F0 PORT A F1 PORT B F2 PORT C F3 CONTROL
8251(U6)	C0 DATA C1 CONTROL

Programmable Timer 8253

Timer 0 of the 8253 has been used on card for the single step function. To enable this capability ensure that the jumper JP9(1-2) should be close.

Timer1 of the 8253 has been used on card for generation of the TXD and RXD baud clock required by USART (8251A). To enable this short JP9(3-4) should be close.

All timer gate, clock and out lines are terminated at 26 pin connector P5

Serial I/O 8251A

We can operate the serial I/O port for RS232C using DB9 connector P1

CONNECTOR PIN DETAILS

BUS EXPANSION CONNECTOR CN1 (50 pin berg, male)

PIN	Signal Description	PIN	Signal Description
1	Gnd	2	Gnd
3	Address line A0	4	Address line A1
5	Address line A2	6	Address line A3
7	Address line A4	8	Address line A5
9	Address line A6	10	Address line A7
11	Address line A8	12	Address line A9
13	Address line A10	14	Address line A11
15	Address line A12	16	Address line A13
17	Address line A14	18	Address line A15
19	CPU status line,S1	20	CPU SOD line,S0
21	INTA line I?P pin2	22	CPU status line,S0
23	Data line D0	24	Data line D1
25	Data line D2	26	Data line D3
27	Data line D4	28	Data line D5
29	Data line D6	30	Data line D7
31	CPU SID line	32	Bus clock to inverter
33	INTR	34	RST 6.5
35	RST 5.5	36	RST7.5
37	HOLD	38	HLDA
39	VCC	40	VCC
41	RD line	42	WR line
43	NC	44	IO?M line
45	ALE line	46	READY line
47	GND	48	RESET signal
49	GND	50	CLK signal

Auxiliary Connector (26 pin) P5

PIN	**Signal Description**	**PIN**	**Signal Description**
1	Gate0 I/P of timer 8253	2	Gate1 I/P of timer 8253
3	Gate2 I/P of timer 8253	4	OUT2 o/p of timer 8253
5	CLK2 o/p of timer 8253	6	OUT1 o/p of timer 8253
7	OUT0 o/p of timer 8253	8	I/P to 8259 through JP3(IRQ0)
9	CLK1 o/p of timer 8253	10	I/P to 8259 through JP3(IRQ1)
11	CLK0 o/p of timer 8253	12,15,25	GND
13,16,17,18,26	NC	14	Through JP4 to inverter U8 pin 10
19	I/P of 8259A(IRQ2)	20	I/P of 8259A(IRQ3)
21	I/P of 8259A(IRQ4)	22	I/P of 8259A(IRQ5)
23	I/P of 8259A(IRQ6)	24	I/P of 8259A(IRQ7)

PPI CONNECTOR P2 & P3 (26 pin male)

The port lines of 855I and II (U14, U15) are terminated in this connector, as shown below:

Pin No.	**Signal**	**Pin No.**	**Signal description**

	description		
1	PC4, IC pin 13	2	PC5, IC PIN 12
3	PC2, IC pin 16	4	PC3, IC pin 17
5	PC0, IC pin 14	6	PC1, IC pin 15
7	PB6, IC pin 24	8	PB7, IC pin 25
9	PB4, IC pin 22	10	PB5, IC pin 23
11	PB2, IC pin 20	12	PB3, IC pin 21
13	PB0, IC pin 18	14	PB1, IC pin 19
15	PA6, IC pin 38	16	PA7, IC pin 37
17	PA4, IC pin 40	18	PA5, IC pin 39
19	PA2, IC pin 2	20	PA3, IC pin 1
21	PA0, IC pin 4	22	PA1, IC pin 3
23	PC6, IC pin 11	24	PC7, IC pin 10
25	+ 5V JP1	26	GND

Serial I/O connector –P1

Pin No	Signal description
3	RS-232C O/P-TXD
2	RS-232C I/P-RXD
4	RS-232C O/P-DTR
6	RS-232C O/P-DSR
5	GND

Power Connector P4

Pin No	Signal description	Colour Code
4,5	GND	White /Orange/Blue
9	+ 5V I/P	Black/Yellow

To branch to the serial monitor commands the following initialization of memory is required to select the baud rate

FFA6	FFA7	BAUD RATE
05	00	19200
0A	00	9600
14	00	4800
28	00	2400
50	00	1200
A0	00	600
40	01	300

CRT command

Syntax: E/CRT

This command allows the mode to be change to serial for communication with a crt or PC.RAM locations FFA6 and FFA7 should be initialized to the values corresponding to the required baud rate. Pressing the E?CRT key followed by ‘0’ invokes serial monitor. Pressing the E/CRT key followed by ‘1’ selects CRT/KBD mode of operation. Mode identifier is initialized to 01H. Control is transferred to the

CRT/keyboard, the system display is cleared and the keyboard is locked out, pressing the RESET key restores the mode to system keyboard/display.

Serial Monitor Commands:

The RESET key is then pressed followed by E CRT & 0 key to transfer control to a CRT terminal or a PC COM port connector to the connector DB9-P1. The above values are based on the 1.536 MHz clock/ip to the 8253.

Help Menu

Syntax H<cr>

On pressing the key 'H' the following menu will be displayed

EX:>H<

Memory Commands		**Utility Commands**	
<D>	Display	<X>	Examine Register
<D>	Display	<D>	Display
<M>	Modify	<A>	Assemble
<B>	Block Move	< Z>	Disassemble
<I>	Insert	<E>	EPROM Programmer
<K>	Delete	<R>	cassette Save
<F>	Block Fill	<P>	Cassette Load
<C>	Block Complement	<H>	Help

Line Assembler Command

Syntax : A
Starting Address :[9100]<cr>

Line	Label	Source Code	Loc	Obj
0001		MVI A,23<CR>	9100	3E23
0002		MVI C,11<CR>	9102	0E11
0003		MOV B,C<CR>	9104	41
0004		RST 1	9105	CF

Press ESC to terminate the assembler

Disassemble Memory Command
Syntax : Z
starting address :9100<cr>
Ending address :9109<cr>

Line	Label	Source Code	Loc	Obj
0001		MVI A,23<CR>	9100	3E23
0002		MVI C,11<CR>	9102	0E11
0003		MOV B,C<CR>	9104	41
0004		RST 1	9105	CF

Go Command

Syntax : G
starting address :9100<cr>
The above executes the program at location 9100

Examine Register Command

Syntax : X
Register : A=09-66<space bar>…<cr>
The content of register are 09 and the – informs the user can changes if required.

Modified Memory Command

Syntax :M<cr>
starting address :8100<space bar>
8100 3E-32 11-54…..<cr>
The above command displays the previous content of memory location. The user can changes if required.

Insert Command

Syntax : I
starting address :9100<cr>
ending address :9109<cr>
address at which data to be inserted :9105<cr>
no of bytes : 3<space>32-11-91…..<cr>

The above command indicate to insert 3 bytes starting from location 9105.

ASSIGNMENT-1

STUDY OF PREWRITTEN PROGRAMS USING BASIC INSTRUCTION SET(DATA TRANSFER, LOAD/STORE,ARITHMETIC,LOGICAL) ON THE SIMULATOR

1. LOAD 23H TO D REGISTER AND 5643H TO BC REGISTER PAIR. COPY THE CONTENT OF D REGISTER TO A REGISTER.

Label	Address	Mnemonics	Hex Code			Comments
	C200H	MVI D,23H	16H	23H		Load Reg. D with 23h
	C202H	LXI B,5643H	01	43	56	Load Reg. pair B-C with 5643h
	C205H	MOV A,D	7A			Copy content of D reg. to ACC
	C206H	RST 1	CFH			To stop execution

INPUT: B ← 56 H
C ← 43H
D ← 23H

OUTPUT: A ← 23H
B← 56H
C← 43H
D← 23H

2. LOAD 44H TO A REGISTER AND 08H TO B REGISTER. 'ADD' THE CONTENTS OF REGISTER A AND B.

Label	Address	Mnemonics	Hex Code			Comments
	C100H	MVI A,44H	3EH	44H		Load Reg. A with 44h
	C102H	MVI B,08H	06H	08H		Load Reg. B with 08h
	C104H	ADD B	80			ADD the content of ACC with content of reg.B
	C105H	RST 1	CFH			To stop execution

INPUT: A← 44 H
B ← 08H

OUTPUT: A← 4CH

3. LOAD 23H TO A REGISTER AND 08H TO B REGISTER. ' AND' THE CONTENTS OF REGISTER A AND B.

Label	Address	Mnemonics	Hex Code			Comments
	C000H	MVI A,23H	3EH	23H		Load Reg. A with 23h
	C002H	MVI B,08H	06H	08H		Load Reg. B with 08h
	C004H	ANA B	A0H			AND the content of ACC with content of reg.B
	C005H	RST 1	CFH			To stop execution

INPUT: A ← 23H
B ← 08H

OUTPUT: A ← 0H
F ← content of **flag reg.**

ASSIGNMENT-2

1. TWO 8-BIT NUMBERS ARE STORED INTO THE MEMORY LOCATION C100H & C101H. ADD TWO NUMBERS AND STORED THE RESULT INTO MEMORY LOCATIONS C102H & C103H.
    a. DATA: - C100H → 23H
        i. C101H → FAH

2. SUBTRACT THE CONTENT OF ONE REGISTER FROM THE CONTENT OF ACCUMULATOR.

    DATA: REGISTER CONTENT : FAH,

    ACCUMULATOR CONTENT : 79H

3. TWO 16-BIT NUMBERS ARE STORED INTO THE MEMORY LOCATION STARTING FROM C200H TO C203H. ADD THESE DATA & RESULTS WILL STORED INTO NEXT MEMORY LOCATION.
    DATA: - C200H (LOWER BYTE) → 23H
    C201H (UPPER BYTE) → EBH
    C202H (LOWER BYTE) → 79H
    C203H (UPPER BYTE) → 72H

.

4. .SHIFT A BLOCK OF DATA STARTING FROM MEMORY LOCATION C050H TO ANOTHER BLOCK STARTING FROM C000H

5. FOUR 8-BIT NUMBERS ARE STORED IN MEMORY LOCATION STARTING FROM C300H. PERFORM THE BCD ADDITION ON THESE NUMBERS AND STORED THE RESULTS INTO NEXT MEMORY LOCATION.
    DATA: - C200H → 23H
    C201H → 44H
    C202H → 79H
    C203H → 72H

6. WRITE A PROGRAM FOR PACKING AND UNPACKING OF BCD NUMBERS.

7 . LOAD 9EH INTO MEMORY LOCATION C100H. CONVERT THE NUMBER INTO ASCII CODE AND STORE THE RESULT IN NEXT MEMORY LOCATION.

ADDITIONAL ASSIGNMENT

1. FIVE NUMBERS ARE STORED IN MEMORY STARTING FROM C200H LOCATION. ADD ALL POSITIVE NUMBERS AND STORE THE RESULT INTO MEMORY LOCATION C206H. FIND THE TOTAL NUMBER OF NEGETIVE NUMBERS AND STORE THE SAME INTO MEMORY LOCATION C207H.

2. WRITE A PROGRAM TO STORE ALL THE ODD NUMBERS STARTING FROM 01H TO 0FH INTO THE MEMORY LOCATION STARTING FROM C200H. WRITE THE SAME PROGRAM FOR EVEN NUMBER ALSO.

3. STORE A BLOCK OF DATA STARTING FROM C050H. ADD THEM ONE BY ONE TILL YOU GET A ZERO. STORE THE RESULT IN A REGISTER AND CHECK WHETHER THERE IS A CARRY.

4. WRITE A PROGRAM TO ARRANGE 5 NUMBERS STORED INTO MEMORY STARTING FROM C200H LOCATION IN ASCENDING ORDER. REPEAT THE ABOVE FOR DESCENDING ORDER.

5. FEW NUMBERS ARE STORED INTO MEMORY STARTING FROM C300H LOCATION. WRITE A PROGRAM TO FIND OUT HOW MANY TIMES THE NUMBER 22H IS PRESENT THERE.

6. WRITE A PROGRAM TO GENERATE A FIBBONACI SERIES.

7. A STRING OF READINGS IS STORED IN MEMORY LOCATIONS STARTING AT C070H, AND THE END OF THE STRING IS INDICATED BY THE BYTE 0DH. WRITE A PROGRAM TO CHECK EACH BYTE IN THE STRING, AND SAVE THE BYTES IN THE RANGE OF 30H TO 39H (BOTH INCLUSIVE) IN MEMORY LOCATIONS STARING FROM C090H.
   DATA: 35H, 2FH, 30H, 39H, 3AH, 37H, 7FH, 31H, 0DH, 32H

ASSIGNMENT-3

Familiarization of 8086A microprocessor kit/simulator and assembly language programming using 8086A microprocessor/simulator for :

a) Addition of two 32-bit Hex numbers
b) String matching
c) Shifting a block of data from one memory location to another
d) Finding the largest/ smallest number from an array

ASSIGNMENT-4

Interfacing with switches and LEDs and glowing LEDs according to read switch status and scrolling-blinking using delay subroutines through

a) PPI 8255A with 8085A trainer kit
b) 8051 microcontroller

ASSIGNMENT-5

Interfacing with seven segment displays through 8-bit latch (e.g., 74LS373) using-

a) 8085A trainer kit,

b)8086A trainer kit and 8255A PPI employing absolute and partial decoding concept as a peripheral mapped output port with absolute address decoding

ASSIGNMENT-6

ADC, DAC and Stepper motor interfacing with 8086A microprocessor/8051 microcontroller and their programming.

USER AVAILABLE RAM 0:1000 $_{H}$TO 0: FFFF$_{H}$
START PROMPT (i.e DYNA – 86>)

HOW TO WRITE A PROGRAM STEPS: -

1. Switch **ON** the power supply. It gives a sign-on-message **“DYNA -86>”.**
2. Type **A Segment address:Offset (starting) address> and then press <enter> key from keyboard**
   e.g. A 0:1000<enter> It returns 0000:1000
3. Now **WRITE** source statement**,** press **<enter>** key after every source statement.
4. Terminate your program by entering **INT 3** instruction(jump to sign on prompt**)**. Again press**<enter>** key
5. To return start prompt **press <esc>** key from keyboard.

HOW TO CHECK A PROGRAM STEPS: -

1. To disassemble **U <Segment address:Offset(starting) address>,<segment address:Offset(end) address> press <enter> at start prompt**

e.g. DYNA— 86> U 0:1000,0:1009<enter>

2. Press **<enter>** until end address is encountered.

HOW TO CHANGE ANY INSTRUCTION IN PROGRAM STEPS: -

1. Go to Assembler mode by typing **A <Segment address: Offset(starting) address>** followed by **<enter>**. At start prompt

e.g. DYNA—86> A 0:1009 <enter>

HOW TO EXECUTE A PROGRAM: -

From **start** prompt (i.e **DYNA-86>**) Type **G<segment address: Offset(starting) address>** press **<enter> key from keyboard**

e.g. DYNA— 86> G0:1000<enter>

HOW TO EXECUTE A PROGRAM STEP BY STEP: -

1. Use command **T <Segment address:offset(starting) address>** to trace step by step execution and check the output register content.

e.g. DYNA—86> T 0:2000 <enter>

2. To execute next instruction type T and press <enter>.

e.g. **DYNA— 86>T 0:2000 <enter>**

HOW TO CHECK THE RESULT OF YOUR PROGRAM

Examine Register –. Press **R** followed by <enter> at command prompt**.**

e.g. **DYNA— 86> R <enter>**

e.g. **DYNA— 86> RAX <enter>**

Examine Memory – Press D0:2000 <enter> at command prompt.

Press SPACE BAR key to display next set of 8 bytes

e.g. 0000:2000 FF . FF FF FF FF FF FF FF FF

Familiarization with DYNA – 86Trainer Kit Components:

SYSTEM SPECIFICATION:

❖ **CPU**	:	DYNA-86L is based on Intel 8086 high performance CPU operating at 8 MHz
❖ **MEMORY**	:	
EPROM	:	Monitor Firmware in two 27256 is placed in the highest 64KB bank (F0000H to FFFFFH)
RAM	:	64KB Static RAM with powerful battery backup is provided in the address range 00000 to 0FFFFH keyboard/display interface
❖ **PARALLEL I/O**	:	Two 8255's are present onboard,out of which 1 is used for DYNA-PIO cards and 1 for printer interface
❖ **SERIAL I/O**	:	One RS 232C compatible interface using USART 8251, along with 1488,1489 driver chips provides necessary signals for this interface. The signals are brought out on the 9 pin D type male connector (J5).Baud rate is 2400 for this system which can be selected through software.
❖ **TIMER**	:	Three 16 bit counter/timers using 8254.CH0 is used for baud rate generation. CH1 and CH2 signals are brought out on a 7 pin Relimate connector (J^) and can be used by the User.
❖ **PIC**	:	Programming Interrupt controller using 8259 provides 8 priorities interrupt levels.IRQ5 to IRQ7 are brought out on 50 pin FRC connector (J7) and can be used by the user except IRQ3 all other interrupts are masked.
❖ **KEYBOARD**	:	Computer keyboard can be hocked on to the trainer.
❖ **DISPLAY**	:	40 X 2 LCD display module.

INSTALLATION PROCEDURE OF DYNA-86L TRAINER

1. First connect power supply cord to the power socket of the kit
2. Switch on the Power Supply, Display(LCD display) will show **DYNA-86**
3. Now the system is ready for use.

MEMORY & I/O DETAILS OF DYNA – 86L TRAINER

Chip No.	Description	Add. In Hex	Function
62256	SRAM	00000-0FFFFH	Program Memory
	EPROM	F0000-FFFFF	Firmware
8279	Key/Disp.Cont.	50H and 52H	Data reg. and Status reg.
8259	Int.Cnt.	00h and 02h	ICWA1, and ICW2
8255 #1	PPI U6	60H	PORT A, PORT B,……..
8255 #2	PPI U7	61H	PORT A, PORT B,……..
8251	USART	10H and 12H	Data reg. and Status reg.

REGISTER INITIALIZATION

When DYNA-86L is initialized during power on or when [RES] key is pressed. The sign on message is displayed. When initialized the 8086 registers are set to the values shown in the table given below.

CS	Code Segment	0H
DS	Data Segment	0H
ES	Extra Segment	0H
SS	Stack Segment	0H
IP	Instruction Pointer	0H
FL	Flag	0H
SP	Stack Pointer	06FFH

LOCAL MODE

The LOCAL mode of DYNA-86L contains almost all the commands available in the HEX KEYPAD mode of DYNA-86 assembler and disassembler. All commands are typed on the 1010 keys keyboard and results are displayed on 40 X 2 LCD display

On Power-On or Reset, a jump to the Local Mode takes place and control is passed to its monitor. Press a Enter key, display shows DYNA-86.

Memory reserved for 0:0 to 0:6FFH. The user program should start from 0000:0700H

	Mnemonic	Hex	Mnemonic	Hex	Mnemonic	Hex	Mnemonic
CE	ACI 8-bit	2B	DCX H	52	MOV D,D	E5	PUSH H
8F	ADC A	3B	DCX SP	53	MOV D,E	F5	PUSH PSW
88	ADC B	F3	DI	54	MOV D,H	17	RAL
89	ADC C	FB	EI	55	MOV D,L	1F	RAR
8A	ADC D	76	HLT	56	MOV D,M	D8	RC
8B	ADC E	DB	IN 8-bit	5F	MOV E, A	C9	RET
8C	ADC H	3C	INR A	58	MOV E,B	20	RIM
8D	ADC L	04	INR B	59	MOV E,C	07	RLC
8E	ADC M	0C	INR C	5A	MOV E,D	F8	RM
87	ADD A	14	INR D	5B	MOV E,E	D0	RNC
80	ADD B	1C	INR E	5C	MOV E,H	C0	RNZ
81	ADD C	24	INR H	5D	MOV E,L	F0	RP
82	ADD D	2C	INR L	5E	MOV E,M	E8	RPE
83	ADD E	34	INR M	67	MOV H,A	E0	RPO
84	ADD H	03	INX B	60	MOV H,B	0F	RRC
85	ADD L	13	INX D	61	MOV H,C	C7	RST 0
86	ADD M	23	INX H	62	MOV H,D	CF	RST 1
C6	ADI 8-bit	33	INX SP	63	MOV H,E	D7	RST 2
A7	ANA A	DA	JC 16-bit	64	MOV H,H	DF	RST 3
A0	ANA B	FA	JM 16-bit	65	MOV H,L	E7	RST 4
A1	ANA C	C3	JMP 16-bit	66	MOV H,M	EF	RST 5
A2	ANA D	D2	JNC 16-bit	6F	MOV L,A	F7	RST 6
A3	ANA E	C2	JNZ 16-bit	68	MOV L,B	FF	RST 7
A4	ANA H	F2	JP 16-bit	69	MOV L,C	C8	RZ
A5	ANA L	EA	JPE 16-bit	6A	MOV L,D	9F	SBB A
A6	ANA M	E2	JPO 16-bit	6B	MOV L,E	98	SBB B
E6	ANI 8-bit	CA	JZ 16-bit	6C	MOV L,H	99	SBB C
CD	CALL 16-bit	3A	LDA 16-bit	6D	MOV L,L	9A	SBB D
DC	CC 16-bit	0A	LDAX B	6E	MOV L,M	9B	SBB E
FC	CM 16-bit	1A	LDAX D	77	MOV M,A	9C	SBB H
2F	CMA	2A	LHLD 16-bit	70	MOV M,B	9D	SBB L
3F	CMC	01	LXI B, 16-bit	71	MOV M,C	9E	SBB M
BF	CMP A	11	LXI D, 16-bit	72	MOV M,D	DE	SBI 8-bit
B8	CMP B	21	LXI H, 16-bit	73	MOV M,E	22	SHLD 16-bit
B9	CMP C	31	LXI SP, 16-bit	74	MOV M,H	30	SIM
BA	CMP D	7F	MOV A,A	75	MOV M,L	F9	SPHL
BB	CMP E	78	MOV A,B	3E	MVI A, 8-bit	32	STA 16-bit
BC	CMP H	79	MOV A,C	06	MVI B, 8-bit	02	STAX B
BD	CMP L	7A	MOV A,D	0E	MVI C, 8-bit	12	STAX D
BE	CMP M	7B	MOV A,E	16	MVI D, 8-bit	37	STC
D4	CNC 16-bit	7C	MOV A,H	1E	MVI E, 8-bit	97	SUB A
C4	CNZ 16-bit	7D	MOV A,L	26	MVI H, 8-bit	90	SUB B
F4	CP 16-bit	7E	MOV A,M	2E	MVI L, 8-bit	91	SUB C
EC	CPE 16-bit	47	MOV B,A	36	MVI M, 8-bit	92	SUB D
FE	CPI 8-bit	40	MOV B,B	00	NOP	93	SUB E
E4	CPO 16-bit	41	MOV B,C	B7	ORA A	94	SUB H
CC	CZ 16-bit	42	MOV B,D	B0	ORA B	95	SUB L
27	DAA	43	MOV B,E	B1	ORA C	96	SUB M
09	DAD B	44	MOV B,H	B2	ORA D	D6	SUI 8-bit
19	DAD D	45	MOV B,L	B3	ORA E	EB	XCHG
29	DAD H	46	MOV B,M	B4	ORA H	AF	XRA A
39	DAD SP	4F	MOV C,A	B5	ORA L	A8	XRA B
3D	DCR A	48	MOV C,B	B6	ORA M	A9	XRA C
05	DCR B	49	MOV C,C	F6	ORI 8-bit	AA	XRA D
0D	DCR C	4A	MOV C,D	D3	OUT 8-bit	AB	XRA E
15	DCR D	4B	MOV C,E	E9	PCHL	AC	XRA H
1D	DCR E	4C	MOV C,H	C1	POP B	AD	XRA L
25	DCR H	4D	MOV C,L	D1	POP D	AE	XRA M
2D	DCR L	4E	MOV C,M	E1	POP H	EE	XRI 8-bit
35	DCR M	57	MOV D,A	F1	POP PSW	E3	XTHL
0B	DCX B	50	MOV D,B	C5	PUSH B		
1B	DCX D	51	MOV D,C	D5	PUSH D		