

รายการอ้างอิง

1. Vincent Tseng. Microprocessor Development and Development Systems. London: Granada Publishing, 1982.
2. Ratiquzzaman M. Microprocessors and Microcomputer Development Systems. U.S.A.: Harper & Row Publishers, Inc. 1984.
3. Paul L. Borrill. Backplane Bus Standards, Why we need them, What we have got, Who makes them. Microprocessors and Microsystems. Vol.6(No.9), pp. 450-454. 1982.
4. David B. Gustavson. Computer Buses - A Tutorial. IEEE Micro. pp. 7-22. August 1984.
5. Paul L. Borrill. Microprocessor Bus Structures and Standards. IEEE Micro. pp.84-95. February 1981.
6. Harvey J. Hindin. Board-Level Computers, Computer Design. U.S.A.: October 1986.
7. Microtek International. MICE-II User's Guide for 8-bit Series Microprocessor. TAIWAN: Microtek International, 1985.
8. ZAX Corporation. ICD - 278 for Z80 User's Manual. ZAX Corporation, 1985.
9. Pro-Log. System Data Book. California: Pro-Log, 1987.
10. Octagon Systems. Octagon STD Bus Handbook. U.S.A.: Octagon Systems, 1990.
11. Arthur L. Dexter. Microcomputer Bus Structures and Bus Interface Design. New York: Marcel Dekker, 1987.
12. STD Manufacturer's Group. STD Bus Specification and Practice. California: Pro-Log, 1984.

13. IEEE Computer Society. IEEE Standard for an 8 - bit Microcomputer Bus System : STD Bus. New York : The Institute of Electrical and Electronics Engineering , 1988.
14. Lewis C. Eggebercht, Howard W. Sams. Interfacing to the IBM Personal Computer. U.S.A.: 1983.
15. ประเสริฐ เวียงสุโขทัย. STD Bus ตั้่นตำรับระบบบัสมาตรฐานอุตสาหกรรม. เซมิคอนดัคเตอร์ อิเล็กทรอนิกส์. ฉบับที่ 105(มี.ค.-เม.ย. 34). หน้า 79-89.
16. R. Zaks , A. Lesea. Microprocessor Interfacing Techniques. U.S.A. : Sybex, 1979.
17. Multitech Industrial Corp. MPF-I Manual. TAIWAN: Multitech Industrial Corp. 1981.
18. Advanced Electronic Systems. SDA - 85 Manual. Bangalore: Advanced Electronic Systems.
19. ET.Board 3.0. เซมิคอนดัคเตอร์ อิเล็กทรอนิกส์. ฉบับที่ 93. หน้า 193-195. 2532.
20. John Birkner, Vincent Coil. Programmable Array Logic Handbook. Sunnyvale: Monolithic Memories. 1981.
21. คู่มือชิพซัพพอร์ตและหน่วยความจำ. ครั้งที่ 1. กรุงเทพฯ: ซีเอ็ดดูเคชั่น จำกัด. 2529.
22. 80/DS Expert-PL/M Microprocessor Program Development Tools. U.S.A.: Caine, Fiber and Gordon. 1986.
23. Daniel D. McCracken. A Guide to PL/M Programming for Microcomputer Applications. California: Addison-Wesley Publishing. 1982.
24. Microtec Research. Paragon MCC85/MCCZ80/MCC180 Version 1.5 C Cross Compiler. California: Microtec Research. 1987.
25. Microtec Research ,ZAX. Paragon ASMZ80 and LODZ80 Manual. California: ZAX Corporation. 1985.

26. สุภานันท์ หิรัญยพิสูทธิกุล, ดร.สมบูรณ์ จงชัยกิจ และ รศ.กฤษดา วิศวธีรานนท์.
เครื่องช่วยพัฒนาระบบไมโครโปรเซสเซอร์ ระดับบอร์ด. การประชุม
 วิชาการทางวิศวกรรมไฟฟ้าครั้งที่ 12. มหาวิทยาลัยเกษตรศาสตร์.2532.
27. ทวี โชคดีพัฒนเทวี และ รศ.กฤษดา วิศวธีรานนท์. อินเทอร์กิต อีมีเลเตอร์ Z-80.
 วิทยานิพนธ์ปริญญาวิศวกรรมศาสตรบัณฑิต จุฬาลงกรณ์มหาวิทยาลัย. 2531.
28. Semware. QEdit Advanced. U.S.A.: Semware. 1990.
29. Borland. SideKick Version 1.56A. U.S.A.: Borland. 1985.
30. บุญนาค สายสนิทเสวีกุล. การใช้โปรแกรม CuWriter Version 1.52.
 กรุงเทพมหานคร: ศูนย์วิจัยและพัฒนาแพทยศาสตรศึกษา คณะแพทยศาสตร์
 จุฬาลงกรณ์มหาวิทยาลัย. 2534.
31. Bop Campbell. Mastering PROCOMM Plus V.1.1B. Sybex . 1988.
32. XelTex. Universal Programming Manual Book. Korea : 1990.
33. สนิท ศิริสวัสดิ์วัฒนา, สุวัฒน์ ภัทรมาลัย และ รศ.กฤษดา วิศวธีรานนท์.
Software Debugger for 68000 Board. กรุงเทพมหานคร:
 ภาควิชาวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย. 2533.

ศูนย์วิทยทรัพยากร
 จุฬาลงกรณ์มหาวิทยาลัย



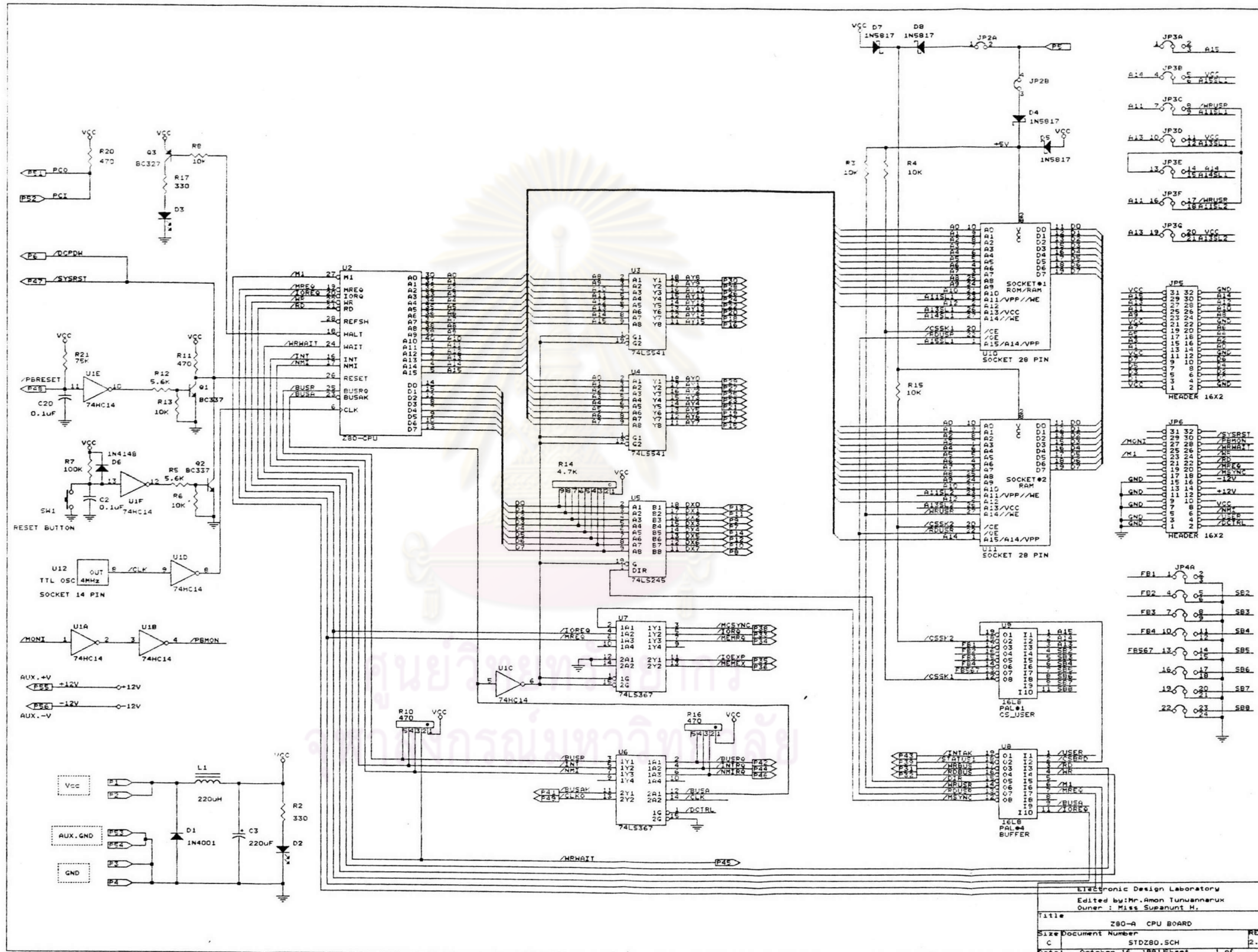
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ก

วงจรของบอร์ดต่าง ๆ ในระบบพัฒนาไมโครโปรเซสเซอร์ระดับบอร์ด

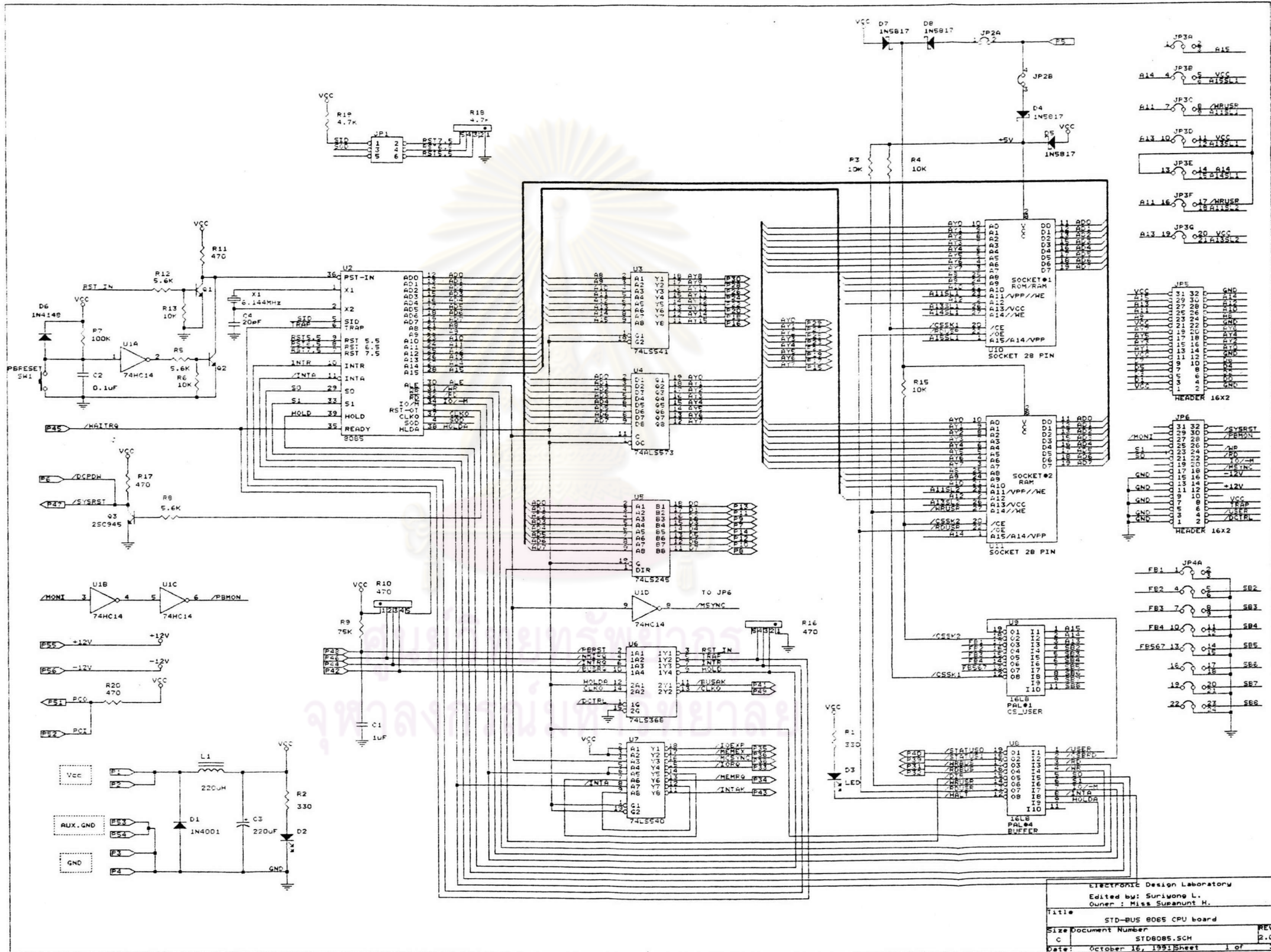


ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

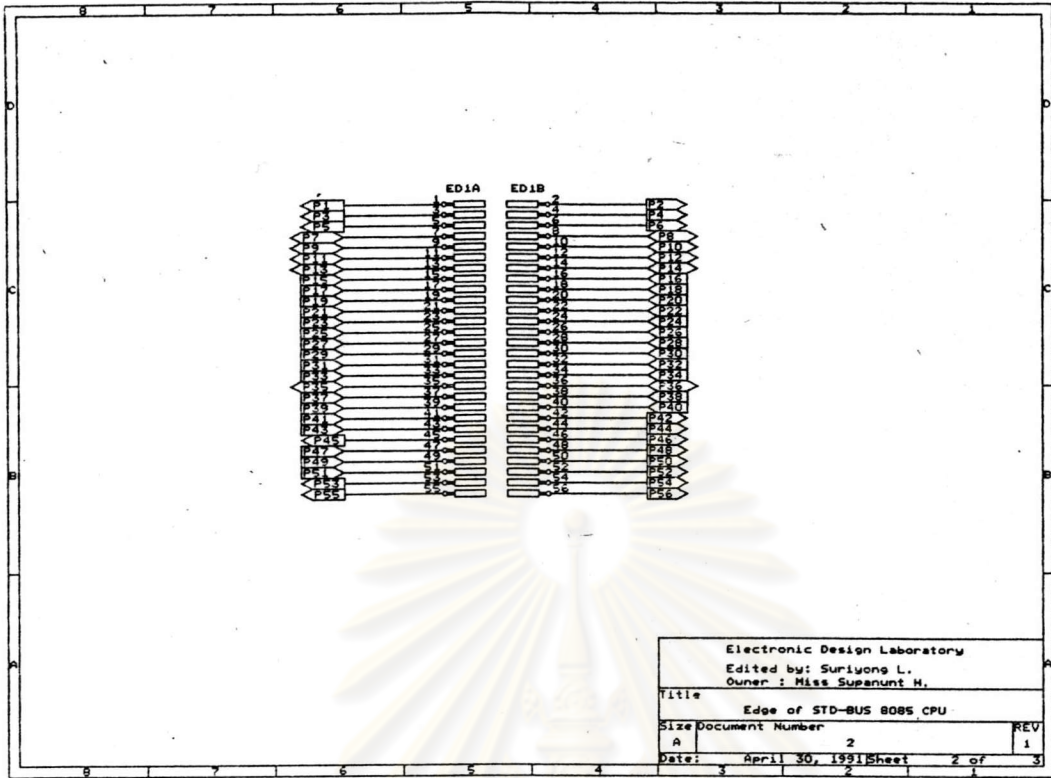


Electronic Design Laboratory
 Edited by: Mr. Amon Tunwanerux
 Owner: Miss Supanunt H.

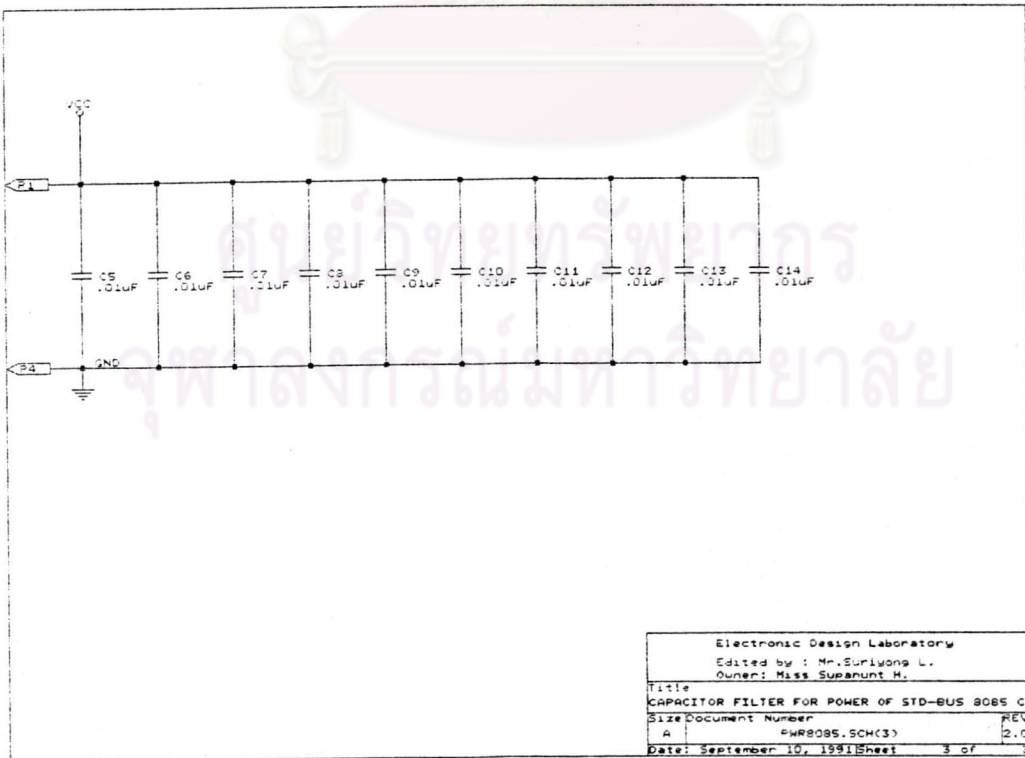
Title	Z80-A CPU BOARD
Size Document Number	STDZ80.SCH
C	2.0
Date:	October 16, 1991
Sheet	1 of 3



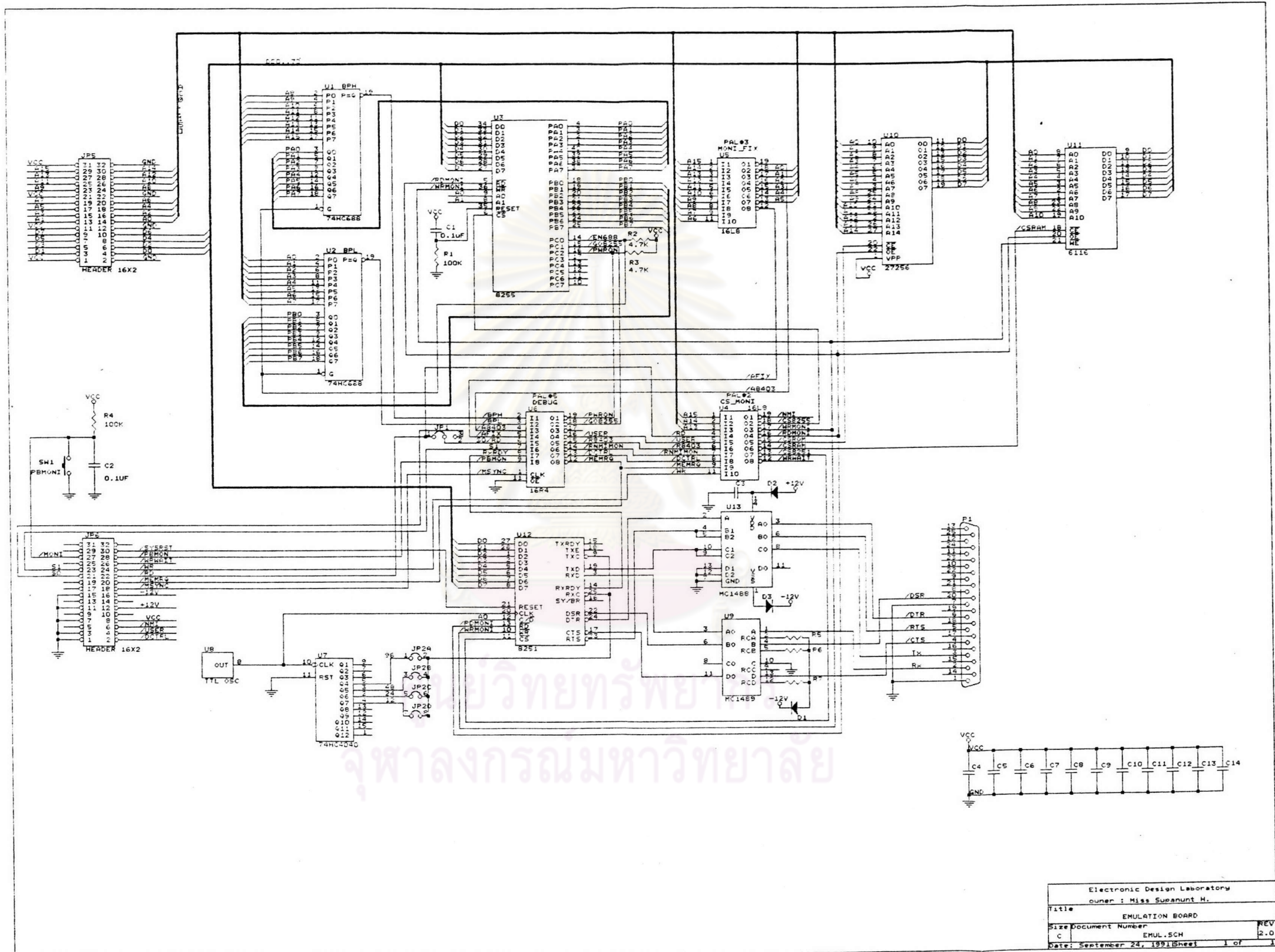
ELECTRONIC Design Laboratory
 Edited by: Suriyong L.
 Owner: Miss Subanunt H.
 Title: STD-BUS 8085 CPU board
 Size Document Number: STD8085.SCH REV: 2.0
 Date: October 16, 1991 Sheet 1 of 3

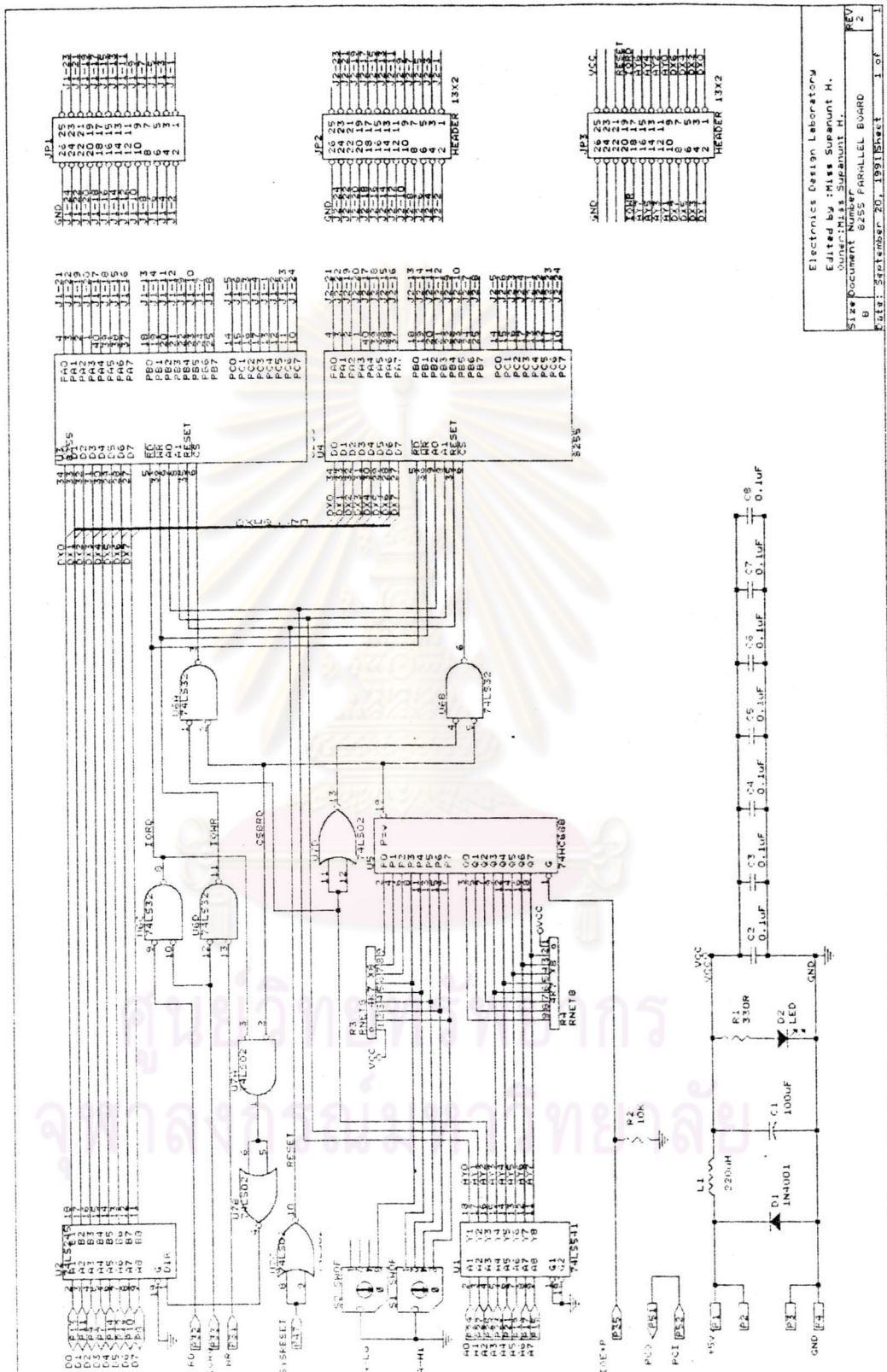


Electronic Design Laboratory		
Edited by: Suriyong L.		
Owner: Miss Supanunt H.		
Title	Edge of STD-BUS 8085 CPU	
Size	Document Number	REV
A	2	1
Date:	April 30, 1991	Sheet 2 of 3



Electronic Design Laboratory		
Edited by: Mr. Suriyong L.		
Owner: Miss Supanunt H.		
Title	CAPACITOR FILTER FOR POWER OF STD-BUS 8085 C	
Size	Document Number	REV
A	4WR8085.SCH(3)	2.0
Date:	September 10, 1991	Sheet 3 of 3





Electronics Design Laboratory
Edited by: Miss Supanunt H.
Owner: Miss Supanunt H.
Size: Document Number: 8555 PARALLEL BOARD
B
Date: September 20, 1991 Sheet 1 of 1

ภาคผนวก ข

โปรแกรมของ PAL บนบอร์ดต่าง ๆ ในระบบพัฒนา



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

โปรแกรม PAL บนบอร์ดซีพียู Z-80



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

TITLE Chip Select of User Memory Decoder on CPU Board
 PATTERN NCS_USER.PDS
 REVISION 3
 AUTHOR Supanunt Hirunyaphisutthikul
 COMPANY EDL
 DATE 9-24-91

CHIP PAL1 PAL16L8

:1	2	3	4	5	6	7	8	9	10
A15	A14	A13	SB2	SB3	SB4	SB5	SB6	SB7	GND
SB8	CSSOCKET1	FB567	FB4	FB3	FB2	FB1	CSSOCKET2	CSBOARD	VCC
:11	12	13	14	15	16	17	18	19	20

EQUATIONS

$$\begin{aligned}
 /CSSOCKET1 &= /A15 * /A14 * /A13 * /FB1 \\
 &+ /A15 * /A14 * A13 * /FB2 \\
 &+ /A15 * A14 * /A13 * /FB3 \\
 &+ /A15 * A14 * A13 * /FB4 \\
 &+ A15 * /A14 * /A13 * /FB567 \\
 &+ A15 * /A14 * A13 * /FB567 \\
 &+ A15 * A14 * /A13 * /FB567
 \end{aligned}$$

$$\begin{aligned}
 /CSSOCKET2 &= /A15 * /A14 * A13 * /SB2 * FB2 \\
 &+ /A15 * A14 * /A13 * /SB3 * FB3 \\
 &+ /A15 * A14 * A13 * /SB4 * FB4 \\
 &+ A15 * /A14 * /A13 * /SB5 * FB567 \\
 &+ A15 * /A14 * A13 * /SB6 * FB567 \\
 &+ A15 * A14 * /A13 * /SB7 * FB567 \\
 &+ A15 * A14 * A13 * /SB8
 \end{aligned}$$

$$\begin{aligned}
 /CSBOARD &= /A15 * /A14 * /A13 * /FB1 \\
 &+ /A15 * /A14 * A13 * /FB2 \\
 &+ /A15 * A14 * /A13 * /FB3 \\
 &+ /A15 * A14 * A13 * /FB4 \\
 &+ A15 * /A14 * /A13 * /FB567 \\
 &+ A15 * /A14 * A13 * /FB567 \\
 &+ A15 * A14 * /A13 * /FB567 \\
 &+ /CSSOCKET2
 \end{aligned}$$

จุฬาลงกรณ์มหาวิทยาลัย

TITLE Chip Select and READ/WRITE of Z80 Monitor Memory Decoder
 PATTERN NCS_MONZ.PDS
 REVISION 5
 AUTHOR Supanunt Hirunyaphisutthikul
 COMPANY EDL
 DATE 9-24-91

CHIP PAL2 PAL16L8

:1	2	3	4	5	6	7	8	9	10
A15	A14	A13	RD	USER	R8403	RNMIMON	DISCTRL	MEMRQ	GND
WR	WRWAIT	CS8251	CSRAMMON	CSROMMON	RDMON	WRMON	CS8255	NMI	VCC
:11	12	13	14	15	16	17	18	19	20

EQUATIONS

;Monitor force /NMI to CPU for any break or single step
 ;when already disable INTRQ,BUSRQ,NMIRQ from bus

/NMI = /RNMIMON
 NMI.TRST = DISCTRL ;when DISCTRL high,NMI can active

/CSROMMON = /A15 ;0000-7FFF
 ;ROM
 /CSRAMMON = A15 * /A14 * /A13 ;8000-9FFF
 ;RAM
 /CS8251 = A15 * A14 * /A13 ;C000-DFFF
 ;DATA = C000
 ;CONTROL = C001
 /CS8255 = A15 * A14 * A13 ;E000-EFFF
 ;PORT A = E000
 ;PORT B = E001
 ;PORT C = E002
 ;CONTROL = E003

/WRWAIT = A15 * A14 * USER * /MEMRQ * RD * WR
 WRWAIT.TRST= A15 * A14 * USER * /MEMRQ * RD * WR

/RDMON = /RD * /MEMRQ * USER ;read monitor memory

/WRMON = /WR * /MEMRQ * USER * R8403 ;write monitor memory
 ;when R8403 not active

จุฬาลงกรณ์มหาวิทยาลัย

TITLE Z80 Monitor Fixed Address for Debugging (0066H ->NMI,8403H ->G0)
 PATTERN MNONFIXZ.PDS
 REVISION 3
 AUTHOR Supanunt Hirunyaphisutthikul
 COMPANY EDL
 DATE 9-24-91

CHIP PAL3 PAL16L8

:1	2	3	4	5	6	7	8	9	10
A15	A14	A13	A12	A11	A10	A9	A8	A7	GND
A6	A0066	A5	A4	A3	A2	A1	A0	A8403	VCC
:11	12	13	14	15	16	17	18	19	20

EQUATIONS

:address A0-A15 of PAL3 contact from A0-A15 of CPU Z80 directly

$$/A0066 = /A15*/A14*/A13*/A12*/A11*/A10*/A9*/A8*/A7* A6* A5*/A4*/A3* A2* A1*/A0$$

$$/A8403 = A15*/A14*/A13*/A12*/A11* A10*/A9*/A8*/A7*/A6*/A5*/A4*/A3*/A2* A1* A0$$

ศูนย์วิทยทรัพยากร
 จุฬาลงกรณ์มหาวิทยาลัย

TITLE Control bus Buffer of CPU Z80 Board
 PATTERN NBUFFERZ.PDS
 REVISION 4
 AUTHOR Supanunt Hirunyaphisutthikul
 COMPANY EDL
 DATE 9-24-91

CHIP PAL4 PAL16L8

;1	2	3	4	5	6	7	8	9	10
USER	CSBOARD	RD	WR	NC	M1	MEMRQ	NC	BUSAK	GND
IORQ	MCSYNC	RDUSER	WRUSER	DIR	RDBUS	WRBUS	STATUS1	INTAK	VCC
;11	12	13	14	15	16	17	18	19	20

EQUATIONS

```
;RD signal at bus
/RDBUS = /RD * MEMRQ ;read not memory (may be I/O)
        + /RD * /MEMRQ * /USER ;read user memory on either this board or other
RDBUS.TRST= BUSAK ;enable when no BUSAK

;WR signal at bus
/WRBUS = /WR * MEMRQ ;write not memory (may be I/O)
        + /WR * /MEMRQ * /USER ;write user memory on either this board or other
WRBUS.TRST= BUSAK ;enable when no BUSAK

;read user memory on CPU board
/RDUSER = /RD * /MEMRQ * /USER

;write user memory on CPU board
/WRUSER = /WR * /MEMRQ * /USER

/INTAK = /M1 * /IORQ

;direction of data buffer active to be input
/DIR = /M1 * /IORQ ;interrupt acknowledge
      + /RD * /MEMRQ * /USER * CSBOARD ;read user memory not in CPU board
      + /RD * /IORQ ;read I/O

;machine cycle sync. active low when not read,write or interrupt acknowledge
/MCSYNC = RD * WR * /( /M1 * /IORQ)

;while Z80 I/O (e.g. Z80 CTC) make /INTRQ ,it does not wait for /INTAK
;but looks for /M1 ,2 clock after that CPU will give /IORQ for acknowledge
;that interrupt and RD signal must be high
;so M1 must be masked to prevent interrupt acknowledge when use monitor
;STATUS1 is M1 signal of user
/STATUS1 = /M1 * /USER
STATUS1.TRST = BUSAK ;enable when no BUSAK
```


TITLE Z80 Change Page Monitor/User for Debugging
 PATTERN NDEBUGMZ.PDS
 REVISION 6
 AUTHOR Supanunt Hirunyaphisutthikul
 COMPANY EDL
 DATE 9-24-91

CHIP PAL5 PAL16R4

;1	2	3	4	5	6	7	8	9	10
MCSYNC	BPH	BPL	A8403	A0066	RD	M1	RxRDY	PBMON	GND
OE	MEMRQ	DISCTRL	RNMIMON	R8403	USER	NC	G08255	PWRON	VCC
;11	12	13	14	15	16	17	18	19	20

EQUATIONS

```

;DISCTRL active to disable INTRQ,BUSRQ,NMIRQ from bus
DISCTRL = /BPH * /BPL * /M1 * /MEMRQ * /G08255 ;execute breakpoint
        + RxRDY * /G08255 ;hit any key from IBM
        + /PBMON * /G08255 ;push PBMON
        + /( /USER * /G08255) ;not execute user program
        + /RNMIMON ;force NMI to CPU

;RNMIMON active to force NMI to CPU :to break user program
/RNMIMON := /BPH * /BPL * /M1 * /MEMRQ * /G08255 ;execute breakpoint
        + RxRDY * /G08255 ;hit any key from IBM
        + /PBMON * /G08255 ;push PBMON
        + /RNMIMON * /USER * /G08255 ;latch while run user prog.

;if CPU read or fetch at 8403h ,it will latch A8403 for goto user
/R8403 := /RD * /MEMRQ * USER * /A8403 ;G-read and D,E-fetch

/USER := /MEMRQ*USER*/R8403 */PWRON*/(/A0066*/M1*/MEMRQ*/RNMIMON)
        + /( /M1*/MEMRQ)*/USER */PWRON*/(/A0066*/M1*/MEMRQ*/RNMIMON)
        + /USER*/G08255 */PWRON*/(/A0066*/M1*/MEMRQ*/RNMIMON)

```

ศูนย์วิทยทรัพยากร
 จุฬาลงกรณ์มหาวิทยาลัย

โปรแกรม PAL บนบอร์ดซีพียู 8085



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

TITLE Chip Select of User Memory Decoder on CPU Board
 PATTERN NCS_USER.PDS
 REVISION 3
 AUTHOR Supanunt Hirunyaphisutthikul
 COMPANY EDL
 DATE 9-24-91

CHIP PAL1 PAL16L8

;1	2	3	4	5	6	7	8	9	10
A15	A14	A13	SB2	SB3	SB4	SB5	SB6	SB7	GND
SB9	CSSOCKET1	FB567	FB4	FB3	FB2	FB1	CSSOCKET2	CSBOARD	VCC
:11	12	13	14	15	16	17	18	19	20

EQUATIONS

$$\begin{aligned} /CSSOCKET1 = & /A15 * /A14 * /A13 * /FB1 \\ & + /A15 * /A14 * A13 * /FB2 \\ & + /A15 * A14 * /A13 * /FB3 \\ & + /A15 * A14 * A13 * /FB4 \\ & + A15 * /A14 * /A13 * /FB567 \\ & + A15 * /A14 * A13 * /FB567 \\ & + A15 * A14 * /A13 * /FB567 \end{aligned}$$

$$\begin{aligned} /CSSOCKET2 = & /A15 * /A14 * A13 * /SB2 * FB2 \\ & + /A15 * A14 * /A13 * /SB3 * FB3 \\ & + /A15 * A14 * A13 * /SB4 * FB4 \\ & + A15 * /A14 * /A13 * /SB5 * FB567 \\ & + A15 * /A14 * A13 * /SB6 * FB567 \\ & + A15 * A14 * /A13 * /SB7 * FB567 \\ & + A15 * A14 * A13 * /SB8 \end{aligned}$$

$$\begin{aligned} /CSBOARD = & /A15 * /A14 * /A13 * /FB1 \\ & + /A15 * /A14 * A13 * /FB2 \\ & + /A15 * A14 * /A13 * /FB3 \\ & + /A15 * A14 * A13 * /FB4 \\ & + A15 * /A14 * /A13 * /FB567 \\ & + A15 * /A14 * A13 * /FB567 \\ & + A15 * A14 * /A13 * /FB567 \\ & + /CSSOCKET2 \end{aligned}$$

จุฬาลงกรณ์มหาวิทยาลัย

TITLE Chip Select and READ/WRITE of 8085 Monitor Memory Decoder
 PATTERN NCSMON85.PDS
 REVISION 3
 AUTHOR Supanunt Hirunyaphisutthikul
 COMPANY EDL
 DATE 9-25-91

CHIP PAL2 PAL16L3

:1	2	3	4	5	6	7	8	9	10
A15	A14	A13	RD	USER	R8403	RNMIMON	DISCTRL	MEMRQ	GND
WR	NC	CS8251	CSRAMMON	CSROMMON	RDMON	WRMON	CS8255	TRAP	VCC
:11	12	13	14	15	16	17	18	19	20

EQUATIONS

;Monitor force TRAP to CPU for any break or single step
 ;when already disable INTRQ,BUSRQ,NMIRQ from bus

TRAP = /RNMIMON
 TRAP.TRST = DISCTRL ;when DISCTRL high,TRAP can active

/CSROMMON = /A15 ;0000-7FFF
 ;ROM
 /CSRAMMON = A15 * /A14 * /A13 ;8000-9FFF
 ;RAM
 /CS8251 = A15 * A14 * /A13 ;C000-DFFF
 ;DATA = C000
 ;CONTROL = C001
 /CS8255 = A15 * A14 * A13 ;E000-FFFF
 ;PORT A = E000
 ;PORT B = E001
 ;PORT C = E002
 ;CONTROL = E003

/RDMON = /RD * /MEMRQ * USER ;read monitor memory
 /WRMON = /WR * /MEMRQ * USER * R8403 ;write monitor memory
 ;when R8403 not active

จุฬาลงกรณ์มหาวิทยาลัย

TITLE 8085 Monitor Fixed Address for Debugging (0024H->TRAP,8403H->GO)
 PATTERN NMONFIX8.PDS
 REVISION 2
 AUTHOR Supanunt Hirunyaphisutthikul
 COMPANY EDL
 DATE 9-25-91

CHIP PAL3 PAL16L8

:1	2	3	4	5	6	7	8	9	10
A15	A14	A13	A12	A11	A10	A9	A8	A7	GND
A6	A0024	A5	A4	A3	A2	A1	A0	A8403	VCC
;11	12	13	14	15	16	17	18	19	20

EQUATIONS

:address A0-A7 of PAL3 contact from A0-A7 of CPU 8085 after pass 74LS573
 ;but address A8-A15 of PAL3 get from A8-A15 of CPU 8085 directly

$$/A0024 = /A15*/A14*/A13*/A12*/A11*/A10*/A9*/A8*/A7*/A6* A5*/A4*/A3* A2*/A1*/A0$$

$$/A8403 = A15*/A14*/A13*/A12*/A11* A10*/A9*/A8*/A7*/A6*/A5*/A4*/A3*/A2* A1* A0$$

ศูนย์วิทยทรัพยากร
 จุฬาลงกรณ์มหาวิทยาลัย

TITLE Control Bus Buffer of CPU 8085 Board
 PATTERN NBUFFER8.PDS
 REVISION 2
 AUTHOR Supanunt Hirunyaphisutthikul
 COMPANY EDL
 DATE 9-25-91

CHIP PAL4 PAL16L8

:1	2	3	4	5	6	7	8	9	10
USER	CSBOARD	RD	WR	S0	S1	MEMRQ	INTA	HOLDA	GND
NC	HALT	RUSER	WRUSER	DIR	RDBUS	WRBUS	STATUS1	STATUS0	VCC
:11	12	13	14	15	16	17	18	19	20

EQUATIONS

;RD signal at bus
 /RDBUS = /RD * MEMRQ ;read not memory (may be I/O)
 + /RD * /MEMRQ * /USER ;read user memory on either this board or other
 RDBUS.TRST= /HOLDA ;enable when no HOLDA

;WR signal at bus
 /WRBUS = /WR * MEMRQ ;write not memory (may be I/O)
 + /WR * /MEMRQ * /USER ;write user memory on either this board or other
 WRBUS.TRST= /HOLDA ;enable when no HOLDA

;read user memory on CPU board
 /RUSER = /RD * /MEMRQ * /USER

;write user memory on CPU board
 /WRUSER = /WR * /MEMRQ * /USER

;direction of data buffer active to be input
 /DIR = /INTA ;interrupt acknowledge
 + /RD * /MEMRQ * /USER * CSBOARD ;read user memory not in CPU board
 + /RD * MEMRQ ;read I/O

;S0 and S1 must be masked to prevent usage when use monitor

;STATUS1 is S1 signal of user
 /STATUS1 = /S1 * /USER
 STATUS1.TRST = /HOLDA ;enable when no HOLDA

;STATUS0 is S0 signal of user
 /STATUS0 = /S0 * /USER
 STATUS0.TRST = /HOLDA ;enable when no HOLDA

;HALT signal active low and show LED when S1 and S0 are low
 /HALT = /S0 * /S1

TITLE 8085 Change Page Monitor/User for Debugging
 PATTERN NDEBUG8.PDS
 REVISION 3
 AUTHOR Supanunt Hirunyaphisutthikul
 COMPANY EDL
 DATE 9-25-91

CHIP PAL5 PAL16V8

;1	2	3	4	5	6	7	8	9	10
MCSYNC	BPH	BPL	A8403	A0024	S0	S1	RxRDY	PBMON	GND
OE	MEMRQ	DISCTRL	RNMIMON	R8403	USER	NC	G08255	PWRON	VCC
;11	12	13	14	15	16	17	18	19	20

EQUATIONS

;For 8085 ,we use MCSYNC = /ALE and it can't latch all signal performed by /RD
 ; because ALE fall down before /RD active
 ; so we must use S0 ,S1 and IO/M to tell its status

;DISCTRL active to disable INTRQ,BUSEQ,NMIHQ from bus
 DISCTRL = /BPH * /BPL * S1 * S0 * /MEMRQ * /G08255 ;execute breakpoint
 + RxRDY * /G08255 ;hit any key from IBM
 + /PBMON * /G08255 ;push PBMON
 + /(/USER * /G08255) ;not execute user program
 + /RNMIMON ;force TRAP(/NMI) to CPU

;RNMIMON active to force TRAP(/NMI) to CPU :to break user program
 /RNMIMON := /BPH * /BPL * S1 * S0 * /MEMRQ * /G08255 ;execute breakpoint
 + RxRDY * /G08255 ;hit any key from IBM
 + /PBMON * /G08255 ;push PBMON
 + /RNMIMON * /USER * /G08255 ;latch while run user prog

/R8403 := S1 * /MEMRQ * USER * /A8403

/USER := /MEMRQ*USER*/R8403 */PWRON*/(/A0024* S1* S0*/MEMRQ*/RNMIMON)
 + /(S1* S0*/MEMRQ)*/USER */PWRON*/(/A0024* S1* S0*/MEMRQ*/RNMIMON)
 + /USER*/G08255 */PWRON*/(/A0024* S1* S0*/MEMRQ*/RNMIMON)

จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ค

Software System



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

C:\>DIR

Volume in drive C is EMULATOR
Directory of C:\

SK	<DIR>		1-01-80	12:01a
CW	<DIR>		8-02-90	4:40p
MCC85	<DIR>		1-04-91	6:58p
80DS	LIB <DIR>		1-01-80	12:01a
TMP	<DIR>		1-01-80	1:41a
STD-MDS	<DIR>		10-10-91	9:12p
COMMAND	COM	25307	3-17-87	12:00p
CONFIG	SYS	55	9-20-90	8:07p
AUTOEXEC	BAT	246	1-01-80	12:10a
9 File(s)			1087488 bytes	free



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
C:\CW>DIR
Volume in drive C is MAXTOR70401
Directory of C:\CW
```

```

.           <DIR>           7-16-91    7:03p
..          <DIR>           7-16-91    7:03p
README     BAT             14      7-11-91   12:29p
CUPRINT    CFG            402     10-09-91  12:48p
CW         CFG             24      8-19-91   4:12a
THAI      COM            30085   6-13-91   8:04a
CW152     DOC            21900   7-11-91   1:44p
TEST      DOC            2136    5-24-82   11:57p
22X16     EXE            15858   1-16-89   8:30a
8X20      EXE            14950   1-16-89   8:30a
CU2RW     EXE            19495   1-01-80   7:46p
CUFONT    EXE            48881   6-03-91   7:57p
CUPRINT    EXE            45886   6-12-91   6:03p
CW        EXE            70643   6-10-91   9:24p
ITALIC    FON             5120    11-07-89  1:38a
NORMAL    FON             5120    10-05-90  6:41p
ITALIC    P24            12096   5-24-82   11:53p
ITALICS   P24            8064    6-06-89   3:35p
NORMAL    P24            12096   5-24-82   11:51p
NORMALS   P24            8064    8-28-89   1:18a
ITALIC    PRN            11264   8-28-89   12:56a
ITALICS   PRN            11264   1-01-80   5:11a
NORMAL    PRN            11264   8-28-89   12:39a
NORMALS   PRN            11264   5-17-89   8:10p
24 File(s) 8781824 bytes free
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

C:\MCC85>DIR

Volume in drive C is EMULATOR
Directory of C:\MCC85

.	<DIR>		1-04-91	6:58p
..	<DIR>		1-04-91	6:58p
TMP		0	10-10-91	10:08p
INCL1	SRC	384	4-03-85	6:27p
SIEVE	C	1110	8-08-85	10:46a
MMER	EXE	39760	7-31-86	4:09p
CTYPE	H	1937	11-05-86	5:49p
MATH	H	262	11-05-86	5:49p
SETJMP	H	323	11-05-86	5:49p
STDIO	H	541	11-05-86	5:49p
STDLIB	H	776	11-05-86	5:49p
STRING	H	3400	11-05-86	5:49p
MXRFC	BAT	256	11-06-86	4:36p
MXRFC	CER	8510	11-06-86	4:38p
MCC85	LIB	84195	11-07-86	5:48p
MACH	SRC	648	11-07-86	5:57p
CSYS85	SRC	21132	11-07-86	5:58p
MCC85	OVL	166464	11-10-86	12:46p
CCG85	OVL	139888	11-10-86	12:47p
SIEVE	CRC	3135	11-10-86	1:12p
SIEVE	CMD	128	11-10-86	1:20p
INSTALL1	BAT	128	11-11-86	6:47p
INSTALL	BAT	128	11-11-86	6:49p
MCC85	DOC	19405	11-12-86	4:52p
XRFC	EXE	82816	11-18-86	1:21p
TST85	CMD	391	12-16-86	9:51a
TST85	SRC	6144	8-03-87	3:53p
ASM85	DOC	2176	8-04-87	1:57p
CMP85	ABS	1094	8-05-87	11:46a
MCC85	EXE	77072	7-07-88	2:46p
LOD85	EXE	111019	7-07-88	2:52p

31 File(s) 1130496 bytes free

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

C:\80DS.LIB>DIR

Volume in drive C is EMULATOR
Directory of C:\80DS.LIB

.			<DIR>	1-01-80	12:01a
..			<DIR>	1-01-80	12:01a
80DS	INI	119		1-17-90	2:53p
80PC	EXE	15690		6-01-85	4:19p
80AS	EXE	52596		6-01-85	4:21p
80CROBJ	EXE	14920		6-01-85	4:24p
80LINK	EXE	27000		6-01-85	4:24p
80DSOBJ	EXE	17592		6-01-85	4:25p
80LOC	EXE	25306		6-01-85	4:27p
80HEX	EXE	13292		6-01-85	4:28p
80MAP	EXE	16724		6-01-85	4:29p
80LIBCR	EXE	15790		6-01-85	4:31p
80STRIP	EXE	13808		6-01-85	4:31p
80LIBLS	EXE	13906		6-01-85	4:32p
XSORT	EXE	13578		10-30-85	4:51p
80PL	LIB	11043		11-11-85	2:08p
80P2	EXE	41448		11-11-85	2:15p
80PCG	EXE	57254		11-11-85	2:23p
80PJO	EXE	29832		11-11-85	2:41p
80PP	EXE	18304		11-11-85	2:42p
80PSYM	EXE	21374		4-17-86	12:45p
80PXRF	EXE	14352		4-17-86	1:01p
80P1	EXE	43560		4-17-86	3:42p
80PFO	EXE	30756		4-18-86	10:44a
PLM	BAT	889		1-02-91	9:17p
J	BAT	39		4-15-90	10:38a
README		1056		1-02-91	9:18p
		27 File(s)		1136640 bytes free	

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

C:\STD-MDS>DIR

Volume in drive C is EMULATOR
Directory of C:\STD-MDS

.	<DIR>		10-10-91	9:12p
..	<DIR>		10-10-91	9:12p
Q	EXE	47420	6-26-91	7:13p
QHELP	TXT	963	6-26-91	7:13p
QCONFIG	DAT	7497	1-01-80	11:13p
ASMZ80	<DIR>		8-20-90	3:27p
ASM85	<DIR>		1-01-80	12:00a
PROCOMM	<DIR>		10-10-91	9:33p
UNIPRO	<DIR>		10-10-91	9:35p
9 File(s)			1112064 bytes free	



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

C:\STD-MDS\ASMZ80>DIR

Volume in drive C is EMULATOR
Directory of C:\STD-MDS\ASMZ80

.		<DIR>	8-20-90	3:27p
..		<DIR>	8-20-90	3:27p
ASMZ80	EXE	132304	1-01-80	2:47a
LODZ80	EXE	128535	1-01-80	2:47a
LIFTO	ASM	1027	1-01-80	1:59a
LIFTO	OBJ	347	1-01-80	2:01a
LIFTO	LST	3553	1-01-80	2:01a
LIFTO	HEX	305	1-01-80	2:01a
LIFTO	MAP	465	1-01-80	2:01a
		9 File(s)	1112064 bytes	free



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

C:\STD-MDS\ASM85>DIR

Volume in drive C is EMULATOR
Directory of C:\STD-MDS\ASM85

.	<DIR>		1-01-80	12:00a
..	<DIR>		1-01-80	12:00a
ASM85	EXE	115612	7-07-88	2:51p
LOD85	EXE	111019	7-07-88	2:52p
4 File(s)			1159168 bytes free	



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

C:\STD-MDS\PROCOMM>DIR

Volume in drive C is EMULATOR
Directory of C:\STD-MDS\PROCOMM

.			<DIR>	10-10-91	9:33p
..			<DIR>	10-10-91	9:33p
PCPLUS	EXE	185392		8-24-88	2:42p
PCSETUP	EXE	33182		8-24-88	3:29p
PCPLUS	PRM	1190		10-10-91	9:42p
PCPLUS	XLT	256		10-10-91	9:40p
PCPLUS	KEY	500		10-10-91	9:40p
PCPLUS	DIR	15050		10-10-91	9:44p
			8 File(s)	1136640 bytes	free



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

C:\STD-MDS\UNIPRO>DIR

Volume in drive C is EMULATOR
Directory of C:\STD-MDS\UNIPRO

.		<DIR>	10-10-91	9:35p
..		<DIR>	10-10-91	9:35p
ROM212	EXE	175919	11-27-89	7:27a
HEXOBJ	EXE	14288	4-27-91	1:20a
ROM	DAT	4	10-07-91	8:03p
5 File(s)		1116160 bytes	free	



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ง

Listing Program of ROM monitor



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

Listing Program of ROM monitor Z-80

COMPILEZ.BAT

G80.ASM

Z80.PLM

T80.PLM

OPC-INS.T80

INSMNE.T80

INSOPR.T80

INSBYTE.T80

INSOPC.T80

MNEMONIC.T80

OPERAND.T80

MNE-INS.T80

CB-INS.T80

DDCB-INS.T80

DD-INS.T80

ED-INS.T80

FD-INS.T80

FDCB-INS.T80

ศูนย์วิทยุทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
80as g80  
80pc -J z80 t80  
80link g80.obj z80.obj t80.obj  
80loc -c -m r80.map -l0 -scode -l08700h -sdata -l087e0h -sstack=20h  
80map -sa d.out > r80.smb  
80hex d.out > r80.hex
```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

pagesw EQU 8403h
MONISP EQU 8800H
DATA8251 EQU 0c000H
PORT8255C EQU 0e002H
POP_H EQU 0E1H
JMP_ EQU 0C3H
PUSH_AF EQU 0F5H
LDAX_ EQU 00Ah
RET_ EQU 0C9h
STAX_ EQU 002h
EI_ EQU 0fbh
DI_ EQU 0f3h

```

```

EXTRN init,prompt2,allregister
EXTRN reg

```

```

; DECLARE reg(13) ADDRESS
;
; reg: DS 13*2
;PC: reg(0) = reg+0
;AF: reg(1) = reg+2
; F: LOW(reg(1)) = reg+2
; A: HIGH(reg(1)) = reg+3
;BC: reg(2) = reg+4
;DE: reg(3) = reg+6
;HL: reg(4) = reg+8
; I: reg(5) = reg+10
;IFF: LOW(reg(5)) = reg+10
; I: HIGH(reg(5)) = reg+11
;SP: reg(6) = reg+12
;IX: reg(7) = reg+14
;IY: reg(8) = reg+16
;AF': reg(9) = reg+18
;BC': reg(10) = reg+20
;DE': reg(11) = reg+22
;HL': reg(12) = reg+24

```

```

;public procedure
PUBLIC gopc,peek,poke

```

```

CSEG
NOP
NOP
NOP
NOP
JMP init ;goto start main and set communication

ORG 0038H ;FFh may be found and make program confuse
JMP init ;so goto main and set communication again

ORG 0066H ;monitor force NMI for break
JMP brk ;so goto save register value (BRK = 0124H)

```

```

;subroutine for GO command
gopc: MVI A,JMP_ ;restore PC
STA pagesw - 2
LHLD reg+0
SHLD pagesw - 1

```

```

;DB OXXH FOR Z80 CODE
LHLD reg+18 ;restore AF'
PUSH H
POP PSW

LHLD reg+20 ;restore BC'
PUSH H
POP B

LHLD reg+22 ;restore DE'
PUSH H
POP D

LHLD reg+24 ;restore HL

DB 0D9H ;EXX
;HL'=HL,DE'=DE,BC'=BC
DB 008H ;EX AF,AF'
;AF'=AF
LHLD reg+16 ;RESTORE IY
PUSH H
DB 0FDH,0E1H ;POP IY

LHLD reg+14 ;RESTORE IX
PUSH H
DB 0DDH,0E1H ;POP IX

LHLD reg+10 ;RESTORE IFF
MOV A,L
RAR ;IF IFF=1 THEN ENABLE
JNC DISABLE ;ELSE DISABLE

```

```

MVI A,EI_
JMP restore
DISABLE: MVI A,DI_
restore: STA pagesw - 3
MOV A,H ;RESTORE I
DB OEDH,047H ;LD 1,A

LHLD reg+2 ;restore AF
PUSH H
POP PSW

LHLD reg+12 ;restore SP
SPHL

LHLD reg+4 ;restore BC
MOV C,L
MOV B,H

LHLD reg+6 ;restore DE
MOV E,L
MOV D,H

LHLD reg+8 ;restore HL

JMP pagesw - 3
;pagesw-3 EI/DI
;pagesw-2 JMP
;pagesw-1 low(PC)
;pagesw high(PC)

;subroutine of NMI
; save register,display register
; and jump to main (at prompt2)
brk: SHLD reg+8 ;SAVE HL

XCHG ;SAVE DE
SHLD reg+6

MOV H,B ;SAVE BC
MOV L,C
SHLD reg+4

STA reg+3 ;SAVE A
;If hit any IBM key ,when run user
;it will make a break then RxDY active
;until CPU read 8251 dataport
LDA DATA8251 ;DISABLE RxDY
MVI A,0000011B ;DISABLE /G08255,/EN688
STA PORT8255C ;PORT C OF 8255

LXI H,pagesw ;SAVE PC
MVI A,POP_H ;BY POP H :HL <- (SP) :HL = PC
MOV M,A
INX H
MVI A,JMP_
MOV M,A
INX H
MVI A,LOW(LABEL1)
MOV M,A
INX H
MVI A,HIGH(LABEL1)
MOV M,A
JMP pagesw
;FETCH OPCODE FROM RAM_MONI
;8403 POP H ;READ PC FROM STACK_USER
; JMP LABEL1 ;CONTINUE FETCH OPCODE FROM ROM_MONI
LABEL1: SHLD reg+0

LXI H,pagesw ;SAVE F
MVI A,PUSH_AF ;BY PUSH PSW :(SP) <- AF
MOV M,A ; (SP) = AF
INX H
MVI A,JMP_
MOV M,A
INX H
MVI A,LOW(LABEL2)
MOV M,A
INX H
MVI A,HIGH(LABEL2)
MOV M,A
JMP pagesw
;FETCH OPCODE FROM RAM_MONI
;8403 PUSH PSW ;WRITE AF TO STACK_USER
; JMP LABEL2 ;CONTINUE FETCH OPCODE FROM ROM_MONI
LABEL2: LXI H,pagesw ;THEN POP H :HL <- (SP)
MVI A,POP_H ; HL = AF
MOV M,A
INX H
MVI A,JMP_
MOV M,A

```

```

    INX H
    MVI A,LOW(LABEL3)
    MOV M,A
    INX H
    MVI A,HIGH(LABEL3)
    MOV M,A
    JMP pagesw
;FETCH OPCODE FROM RAM_MONI
;8403 POP H ;READ AF FROM STACK USER
; JMP LABEL2 ;CONTINUE FETCH OPCODE FROM ROM_MONI

LABEL3: MOV A,L ;SAVE F
        STA reg+2

        LXI H,0000H ;SAVE SP
        DAD SP ;HL = SP
        SHLD reg+12

        LXI SP,MONISP ;INIT MONITOR STACK

;For Z80 only
DB 0EDH,057H ;LD A,I;BIT(F,2)=IFF
PUSH PSW
POP H ;H = I ,L bit 2 = IFF
MOV A,L ;mask IFF before save
RAR
RAR
ANI 01h
MOV L,A ;high = I,low = IFF
SHLD reg+10

DB 0DDH,0E5H ;PUSH IX
POP H
SHLD reg+14

DB 0FDH,0E5H ;PUSH IY
POP H
SHLD reg+16

DB 0D9H ;EXX
;HL=HL',DE=DE',BC=BC'
DB 008H ;EX AF,AF'
;AF=AF'
PUSH H ;SAVE HL'
POP H
SHLD reg+24

PUSH D ;SAVE DE'
POP H
SHLD reg+22

PUSH B ;SAVE BC'
POP H
SHLD reg+20

PUSH PSW ;SAVE AF'
POP H
SHLD reg+18

CALL allregister

JMP prompt2 ;jump to main program

;FUNCTION peek
;PL/M call this routine as
;PROCEDURE(ADDR$) BYTE EXTERNAL
;first parameter from PL/M sent by BC or C
;second parameter from PL/M sent by DE or E
;third or more parameter sent by stack
;e.g. CONTENT = peek(OPERAND(0));
; BC = OPERAND(0)
;it return parameter by HL or A
;other registers may be changed

peek: LXI H,pagesw
      MVI M,LDAX_ ;8403 0A LD A,(BC)
      INX H
      MVI M,RET_ ;2404 C9 RET
      JMP pagesw
;FETCH OPCODE FROM RAM_MONI
;8403 LDAX B ;READ CONTENT FROM MEMORY_USER (addr.XYYY)
; RET ;CONTINUE FETCH OPCODE FROM ROM_MONI

;FUNCTION poke
;PL/M call this routine as
;PROCEDURE(ADDR$,CONTENT) EXTERNAL
;e.g. CALL poke(OPERAND(0),OPERAND(1));
; BC = OPERAND(0)

```

```
;          E = OPERAND(1)
;other registers may be changed

poke:     LXI H,pagesw
          MVI M,STAX_   ;8403 02  LD (BC),A
          INX H
          MVI M,RET_   ;2404 C9  RET
          MOV A,E
          JMP pagesw
;FETCH OPCODE FROM RAM_MONI
;8403  STAX B           ;WRITE CONTENT TO MEMORY_USER (addr.XXYY)
;      RET             ;CONTINUE FETCH OPCODE FROM ROM_MONI

          END
```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย


```

z80:DO;
DECLARE /* symbol */
NUL      LITERALLY '00h', /* ^a */
ETX      LITERALLY '03h', /* ^C */
ACK      LITERALLY '06h', /* ^F */
BEL      LITERALLY '07h', /* ^G */
BS       LITERALLY '08h',
HT       LITERALLY '09h', /* ^I */
LF       LITERALLY '0ah', /* ^J */
CR       LITERALLY '0dh',
XON      LITERALLY '11h', /* ^q */
XOFF     LITERALLY '13h', /* ^S */
NAK      LITERALLY '15h', /* ^U */
ESC      LITERALLY '1bh',
DEL      LITERALLY '7fh', /* like NUL */
newline LITERALLY 'CR,LF',
err      LITERALLY '40h',
true     LITERALLY '0ffh',
false    LITERALLY '0',
REPEAT   LITERALLY 'DO WHILE 1',
UNTIL    LITERALLY 'IF',
ENDREPEAT LITERALLY 'THEN UNDO;END';
/* REPEAT;
      OUTPUT (51h) = 40h;
      UNTIL INPUT (50h) = 0000$0010b ENDREPEAT; */

DECLARE /* label */
(init,prompt,prompt2) LABEL PUBLIC;

DECLARE /* port */
DATA8251 BYTE AT(0c000H),
CONTROL8251 BYTE AT(0c001H),
PORT8255A BYTE AT(0e000H),
PORT8255B BYTE AT(0e001H),
PORT8255C BYTE AT(0e002H),
CONTROL8255 BYTE AT(0e003H);

DECLARE /* table */
mnemonic(69) STRUCTURE(char(4) BYTE) EXTERNAL,
operand(55) STRUCTURE(char(8) BYTE) EXTERNAL,
insopc(698) STRUCTURE(opc(3) BYTE) EXTERNAL,
insmne(698) BYTE EXTERNAL,
insopr(698) STRUCTURE(opr(2) BYTE) EXTERNAL,
insbyte(698) BYTE EXTERNAL,
mneinsptr(69) ADDRESS EXTERNAL,
opcinsptr(256) ADDRESS EXTERNAL,
cbinsptr(256) ADDRESS EXTERNAL,
edins(56) STRUCTURE(ptr ADDRESS,opc2 BYTE) EXTERNAL,
ddcbins(31) STRUCTURE(ptr ADDRESS,opc4 BYTE) EXTERNAL,
fdcbins(31) STRUCTURE(ptr ADDRESS,opc4 BYTE) EXTERNAL,
ddins(40) STRUCTURE(ptr ADDRESS,opc2 BYTE) EXTERNAL,
fdins(40) STRUCTURE(ptr ADDRESS,opc2 BYTE) EXTERNAL;

DECLARE /* constant */
help(*) BYTE PUBLIC DATA(
Assemble  A [address]',newline,'
Breakpoint B [address]',newline,'
Clearbreak C',newline,'
Dump      D [address|range]',newline,'
Enter     E address [list]',newline,'
Fill      F range list',newline,'
Go        G [address]',newline,'
Help      H',newline,'
Input     I address',newline,'
Load      L',newline,'
Output    O address byte',newline,'
Register  R [registername]',newline,'
Trace     T',newline,'
Unassemble U [S] [address|range]',newline,'
Write     W range',newline,'
reset     X',ETX
),
error(*)BYTE PUBLIC DATA(' Error',newline,ETX),
write$guide(*) BYTE PUBLIC DATA(
Use write COM to file utility of your terminal.',newline,'
Strike a key when ready..',newline,ETX),
intelhexend(*) BYTE PUBLIC DATA(
:00000001FF',newline,
1ah,newline,'
Write ready.',newline,ETX),
loadguide(*) BYTE PUBLIC DATA(
Use load file to COM utility of your terminal.',newline,ETX),
loadready(*) BYTE PUBLIC DATA(
newline,'Load ready.',newline,ETX);

DECLARE /* constant */
icename(*) BYTE PUBLIC DATA(
STD Debugger for Z80',newline,ETX),
pinname(*) BYTE PUBLIC DATA
(HT,'RESET',HT,'BUSRQ',HT,'NMI',HT,'INT',HT,'(Enable/Disable)',newline,ETX),

```

```

blank(*) BYTE PUBLIC DATA(' ',ETX),
allregname(*) BYTE DATA('
PC A F(SZHPNC) B C D E H L I IFF SP',
' IX IY AF',27h,' BC',27h,' DE',27h,' HL',27h,
endline,ETX);

```

```

DECLARE regname(28) STRUCTURE(char(3) BYTE) DATA(
'PC ',
'AF ',
'BC ',
'DE ',
'HL ',
'SP ',
'IX ',
'IY ',
'AF',27h,
'BC',27h,
'DE',27h,
'HL',27h,
'A ',
'F ',
'B ',
'C ',
'D ',
'E ',
'H ',
'L ',
'I ',
'IFF',
'SF ',
'ZF ',
'HF ',
'PF ',
'NF ',
'CF ');

```

```

DECLARE regspect(28) STRUCTURE(spec BYTE,number BYTE) DATA(
/*PC ',*/ 2,0,
/*AF ',*/ 2,1,
/*BC ',*/ 2,2,
/*DE ',*/ 2,3,
/*HL ',*/ 2,4,
/*SP ',*/ 2,6,
/*IX ',*/ 2,7,
/*IY ',*/ 2,8,
/*AF',27h,*/ 2,9,
/*BC',27h,*/ 2,10,
/*DE',27h,*/ 2,11,
/*HL',27h,*/ 2,12,
/*A ',*/ 1,1,
/*F ',*/ 0,1,
/*B ',*/ 1,2,
/*C ',*/ 0,2,
/*D ',*/ 1,3,
/*E ',*/ 0,3,
/*H ',*/ 1,4,
/*L ',*/ 0,4,
/*I ',*/ 1,5,
/*IFF',*/ 10h,5,
/*SF ',*/ 17h,1,
/*ZF ',*/ 16h,1,
/*HF ',*/ 14h,1,
/*PF ',*/ 12h,1,
/*NF ',*/ 11h,1,
/*CF ');*/ 10h,1);

```

```

DECLARE /* variable that user can read write */
reg(13) ADDRESS PUBLIC,
(bp) BYTE PUBLIC,
(breakaddress) ADDRESS PUBLIC;

```

```

DECLARE /* variable */
string(64) BYTE PUBLIC,
paraword(4) ADDRESS PUBLIC,
(locationcounter,lastdump,nextunasm,addr,ptr) ADDRESS PUBLIC,
(character,function,digit,n,numword,numbyte,numdigit,numline,
pass,start,stop,tmp,sum,end$of$file) BYTE PUBLIC;

```

```

DECLARE
(unasmtosrc,operandnumber) BYTE;

```

```

DECLARE
ramcode(5) BYTE,
beginramcode ADDRESS DATA(.ramcode),
newregister BYTE PUBLIC;

```

```

/* external procedure from g80.asm */
peek:PROCEDURE(location) BYTE EXTERNAL;
DECLARE
location ADDRESS;

```

```

END peek;

poke:PROCEDURE(location,content) EXTERNAL;
  DECLARE
    location ADDRESS,
    content BYTE;
  END poke;

go$pc:PROCEDURE EXTERNAL;
  END;

/***** ASSEMBLE UTILITY *****/
/* find end of input-string */
findend:PROCEDURE;
  CALL skipblank;
  IF string(n) <> 0
  THEN function = err;
  END findend;

/* find mnemonic */
findmnemonic:PROCEDURE;
  numbyte = 0;
  REPEAT;
    LOW(paraword(numbyte)) = upcase(string(n));
    numbyte = numbyte + 1;n = n + 1;
  UNTIL(string(n) = ' ')OR(string(n) = 0)OR(numbyte = 4)ENDREPEAT;
  IF (string(n) = ' ')OR(string(n) = 0)
  THEN DO WHILE numbyte < 4;
    LOW(paraword(numbyte)) = ' ';
    numbyte = numbyte + 1;
  END;
  ELSE function = err;
  IF function <> err
  THEN DO;
    numword,numbyte = 0;
    REPEAT;
      IF LOW(paraword(numbyte)) = mnemonic(numword).char(numbyte)
      THEN numbyte = numbyte + 1;
      ELSE DO;
        numbyte = 0;
        numword = numword + 1;
      END;
    UNTIL(numbyte = 4)OR(numword > LAST(mnemonic))ENDREPEAT;
    IF numword > LAST(mnemonic)
    THEN function = err;
  END;
  END findmnemonic;

/* find hex operand */
findhexoperand:PROCEDURE(count);
  DECLARE count BYTE;
  numdigit,pass,addr = 0;
  CALL skipblank;
  IF string(n)<>0
  THEN REPEAT;
    IF (string(n)>='0')AND(string(n)<='9')
    OR (string(n)>='A')AND(string(n)<='F')
    OR (string(n)>='a')AND(string(n)<='f')
    THEN DO;
      numdigit = numdigit + 1; pass = 1;
      IF (string(n)>='0')AND(string(n)<='9')
      THEN digit = string(n) - 30h;
      IF (string(n)>='A')AND(string(n)<='F')
      THEN digit = string(n) - 37h;
      IF (string(n)>='a')AND(string(n)<='f')
      THEN digit = string(n) - 57h;
      IF numdigit <= count
      THEN addr = addr * 16 + digit;
      ELSE function = err;
      n = n + 1;
    END;
  UNTIL
  NOT((string(n)>='0')AND(string(n)<='9')
  OR (string(n)>='A')AND(string(n)<='F')
  OR (string(n)>='a')AND(string(n)<='f'))ENDREPEAT;
  END findhexoperand;

/* compare operand */
compareoperand:PROCEDURE;
  numbyte = 0;
  compare:REPEAT;
    IF operand(numword).char(numbyte) = ' '
    THEN DO;
      CALL skipblank;
    END;
  ELSE IF(operand(numword).char(numbyte) = 'b')
  THEN DO;
    CALL findhexoperand(2);
    IF (pass = 1)AND(function <> err)
    THEN DO;

```

```

    paraword(stop) = addr;
    stop = stop + 1; /* number of operand byte */
    pass = 0;
    END;
ELSE DO;
    pass,function = 0;
    UNDO compare;
    END;
END;
ELSE IF(operand(numword).char(numbyte) = 'd')
THEN DO;
    IF string(n) = '+'
    THEN DO;
        n = n + 1;
        CALL findhexoperand(2);
        IF(pass = 1)AND(function <> err)AND(addr <= 7fh)
        THEN DO;
            paraword(stop) = addr;
            stop = stop + 1; /* number of operand byte */
            pass = 0;
            END;
        ELSE DO;
            pass,function = 0;
            UNDO compare;
            END;
        END;
    ELSE IF string(n) = '-'
    THEN DO;
        n = n + 1;
        CALL findhexoperand(2);
        IF(pass = 1)AND(function <> err)AND(addr <= 80h)
        THEN DO;
            paraword(stop) = - addr;
            stop = stop + 1; /* number of operand byte */
            pass = 0;
            END;
        ELSE DO;
            pass,function = 0;
            UNDO compare;
            END;
        END;
    ELSE UNDO compare;
    END;
ELSE IF(operand(numword).char(numbyte) = 'a')
OR
(operand(numword).char(numbyte) = 'r')
THEN DO;
    CALL findhexoperand(4);
    IF (pass = 1)AND(function <> err)
    THEN DO;
        paraword(stop) = addr;
        stop = stop + 1; /* number of operand byte */
        pass = 0;
        END;
    ELSE DO;
        pass,function = 0;
        UNDO compare;
        END;
    END;
ELSE IF operand(numword).char(numbyte) = upcase(string(n))
THEN DO;
    n = n + 1;
    END;
ELSE UNDO compare;
numbyte = numbyte + 1;
UNTIL(numbyte = 8)ENDREPEAT compare;
END compareoperand;
/* fill byte-operand and address-operand */
filloperand:PROCEDURE;
    IF stop = 2
    THEN DO; /* 2 byte-operands */
        CALL poke(locationcounter + start,LOW(paraword(0)));
        CALL displaybyte(LOW(paraword(0)));
        CALL poke(locationcounter + start + 1,LOW(paraword(1)));
        CALL displaybyte(LOW(paraword(1)));
        END;
    ELSE IF stop = 1
    THEN DO;
        IF(insopr(ptr).opr(0) = 29)
        OR(insopr(ptr).opr(1) = 29)
        OR(insopr(ptr).opr(0) = 44)
        OR(insopr(ptr).opr(1) = 44)
        THEN DO; /* 1 address-operand */
            CALL poke(locationcounter + start,LOW(paraword(0)));
            CALL displaybyte(LOW(paraword(0)));
            CALL poke(locationcounter + start + 1,HIGH(paraword(0)));
            CALL displaybyte(HIGH(paraword(0)));
            END;
        ELSE DO; /* 1 byte-operand */
            CALL poke(locationcounter + start,LOW(paraword(0)));

```



```

ELSE DO; /* 1-byte opcode */
  start = 1; /* operands start at byte 1 */
  IF(insopr(ptr).opr(0) = 30)
  OR(insopr(ptr).opr(1) = 30)
  THEN DO; /* jmp relative from next address to paraword(0) */
    addr = locationcounter + 2; /* next address */
    IF((paraword(0) - addr) <= 07fh)
    OR((addr - paraword(0)) <= 080h)
    THEN DO;
      CALL poke(locationcounter, insopc(ptr).opc(0));
      CALL displaybyte(insopc(ptr).opc(0));
      CALL poke(locationcounter + 1, LOW(paraword(0) - addr));
      CALL displaybyte(LOW(paraword(0) - addr));
      locationcounter = addr;
    END;
  ELSE DO;
    CALL disparray(.endline, '    Far Target', ETX);
  END;
  END;
  ELSE DO; /* not jump relative */
    CALL poke(locationcounter, insopc(ptr).opc(0));
    CALL displaybyte(insopc(ptr).opc(0));
    CALL filloperand;
    locationcounter = locationcounter + insbyte(ptr);
  END;
  END;
  END;
  ELSE function = err;
  END findoperand;

/* find pointer to instruction table */
findpointer:PROCEDURE;
  ptr, numbyte, numdigit = 0;
  DO n = 0 TO 3;
    string(n) = peek(nextunasm + n);
  END;
  start = 2; /* if 2-byte opcodes operands start at string(2) */
  IF string(0) = 0cbh
  /* look up CB-INS table */
  THEN ptr = cbinsptr(string(1));
  ELSE IF string(0) = 0edh
  /* search ED-INS table */
  THEN REPEAT;
    IF string(1) = edins(numbyte).opc2
    THEN ptr = edins(numbyte).ptr;
    ELSE numbyte = numbyte + 1;
    UNTIL(ptr <> 0) OR (numbyte = LENGTH(edins)) ENDREPEAT;
  ELSE IF string(0) = 0ddh
  THEN DO;
    IF string(1) = 0cbh
    /* search DDCB-INS table */
    THEN REPEAT;
      IF string(3) = ddcbins(numbyte).opc4
      THEN ptr = ddcbins(numbyte).ptr;
      ELSE numbyte = numbyte + 1;
      UNTIL(ptr <> 0) OR (numbyte = LENGTH(ddcbins)) ENDREPEAT;
    /* search DD-INS table */
    ELSE REPEAT;
      IF string(1) = ddins(numbyte).opc2
      THEN ptr = ddins(numbyte).ptr;
      ELSE numbyte = numbyte + 1;
      UNTIL(ptr <> 0) OR (numbyte = LENGTH(ddins)) ENDREPEAT;
    END;
  ELSE IF string(0) = 0fdh
  THEN DO;
    IF string(1) = 0cbh
    /* search FDCB-INS table */
    THEN REPEAT;
      IF string(3) = fdcbins(numbyte).opc4
      THEN ptr = fdcbins(numbyte).ptr;
      ELSE numbyte = numbyte + 1;
      UNTIL(ptr <> 0) OR (numbyte = LENGTH(fdcbins)) ENDREPEAT;
    /* search FD-INS table */
    ELSE REPEAT;
      IF string(1) = fdins(numbyte).opc2
      THEN ptr = fdins(numbyte).ptr;
      ELSE numbyte = numbyte + 1;
      UNTIL(ptr <> 0) OR (numbyte = LENGTH(fdins)) ENDREPEAT;
    END;
  /* look up OPC-INS table */
  ELSE DO;
    start = 1; /* 1-byte opcode operands start at string(1) */
    ptr = opcinsptr(string(0));
  END;
  END findpointer;

/* display instruction */
displayinstruction:PROCEDURE;
  CALL display$word(nextunasm); CALL displayspace;
  DO n = 0 TO 3;

```

```

    IF n < stop
    THEN CALL displaybyte(string(n));
    ELSE DO;
        CALL displayspace;CALL displayspace;
    END;
    END;
CALL displayspace;
DO n = 0 TO 3;
    CALL display(mnemonic(insmne(ptr) - 1).char(n));
    END;
CALL displayspace;
IF insopr(ptr).opr(0) <> 0
THEN DO n = 0 TO 7;
    operandnumber = insopr(ptr).opr(0);
    character = operand(operandnumber - 1).char(n);
    CALL displayoperand;
    END;
IF insopr(ptr).opr(1) <> 0
THEN DO;
    CALL display(',');
    numdigit = numdigit + 1;
    DO n = 0 TO 7;
        operandnumber = insopr(ptr).opr(1);
        character = operand(operandnumber - 1).char(n);
        CALL displayoperand;
    END;
    END;
END displayinstruction;

/* display source instruction */
displaysourceinstruction:PROCEDURE;
    CALL display(HT);
    DO n = 0 TO 3;
        CALL display(mnemonic(insmne(ptr) - 1).char(n));
    END;
    CALL displayspace;
    IF insopr(ptr).opr(0) <> 0
    THEN DO n = 0 TO 7;
        operandnumber = insopr(ptr).opr(0);
        character = operand(operandnumber - 1).char(n);
        CALL displaysourceoperand;
    END;
    IF insopr(ptr).opr(1) <> 0
    THEN DO;
        CALL display(',');
        numdigit = numdigit + 1;
        DO n = 0 TO 7;
            operandnumber = insopr(ptr).opr(1);
            character = operand(operandnumber - 1).char(n);
            CALL displaysourceoperand;
        END;
    END;
    END displaysourceinstruction;

/* display operand */
displayoperand:PROCEDURE PUBLIC;
    IF character <> ' '
    THEN DO;
        IF character = 'a'
        THEN DO;
            CALL displaybyte(string(start + 1));
            CALL displaybyte(string(start));
            numdigit = numdigit + 4;
        END;
        ELSE IF character = 'b'
        THEN DO;
            CALL displaybyte(string(start));
            start = start + 1;
            numdigit = numdigit + 2;
        END;
        ELSE IF character = 'd'
        THEN DO; /* displacement */
            IF string(start) < 80h
            THEN DO; /* +00 t0 +7F */
                CALL display('+');
                CALL displaybyte(string(start));
            END;
            ELSE DO; /* -01 t0 -80 */
                CALL display('-');
                CALL displaybyte( - string(start));
            END;
            start = start + 1;
            numdigit = numdigit + 3;
        END;
        ELSE IF character = 'r'
        THEN DO;
            IF string(start) < 80h
            THEN CALL displayword(nextunasm + stop + string(start));
            ELSE CALL displayword(nextunasm + stop - ( - string(start)));
            numdigit = numdigit + 4;
        END;
    END;

```

```

    END;
ELSE DO;
    CALL display(character);
    numdigit = numdigit + 1;
END;
END;
END displayoperand;

/* display operand for source */
displaysourceoperand:PROCEDURE;
    IF (operandnumber >= 10)AND(operandnumber <= 17)
    AND(n = 2)
    THEN DO; /* RST 00H to RST 38H */
        CALL display('H');
        numdigit = numdigit + 1;
    END;
    IF character <> ' '
    THEN DO;
        IF character = 'a' /* address */
        THEN DO;
            CALL display('0');
            CALL displaybyte(string(start + 1));
            CALL displaybyte(string(start));
            CALL display('H');
            numdigit = numdigit + 6;
        END;
        ELSE IF character = 'b' /* byte */
        THEN DO;
            CALL display('0');
            CALL displaybyte(string(start));
            CALL display('H');
            start = start + 1;
            numdigit = numdigit + 4;
        END;
        ELSE IF character = 'd' /* displacement */
        THEN DO;
            IF string(start) < 80h
            THEN DO; /* +00 to +7F */
                CALL display('+');
                CALL displaybyte(string(start));
            END;
            ELSE DO; /* -01 to -80 */
                CALL display('-');
                CALL displaybyte(- string(start));
            END;
            CALL display('H');
            start = start + 1;
            numdigit = numdigit + 4;
        END;
        ELSE IF character = 'r' /* relative */
        THEN DO;
            CALL display('0');
            IF string(start) < 80h
            THEN CALL displayword(nextunasm + stop + string(start));
            ELSE CALL displayword(nextunasm + stop - (- string(start)));
            CALL display('H');
            numdigit = numdigit + 6;
        END;
        ELSE DO;
            CALL display(character);
            numdigit = numdigit + 1;
        END;
    END;
END displaysourceoperand;

/* display defind byte for error code */
displaydefindbyte:PROCEDURE;
    CALL display$word(nextunasm);CALL displayspace;
    DO n = 0 TO 3;
        IF n < stop
        THEN CALL displaybyte(string(n));
        ELSE DO;
            CALL displayspace;CALL displayspace;
        END;
    END;
    CALL displayspace;
    CALL disparray(.'DB ',ETX));
    DO n = 0 TO 3;
        IF n < stop
        THEN DO;
            CALL displaybyte(string(n));
            numdigit = numdigit + 2;
        END;
        IF n < stop - 1
        THEN DO;
            CALL display(',');
            numdigit = numdigit + 1;
        END;
    END;
END displaydefindbyte;

```



```

/* display defined byte for error code */
displaysourcedefindbyte:PROCEDURE;
  CALL display(HT);CALL disparray(.('DB ',ETX));
  DO n = 0 TO 3;
    IF n < stop
      THEN DO;
        CALL display('0');
        CALL displaybyte(string(n));
        CALL display('H');
        numdigit = numdigit + 4;
        END;
    IF n < stop - 1
      THEN DO;
        CALL display(',');
        numdigit = numdigit + 1;
        END;
    END;
  END displaysourcedefindbyte;

/* display comment */
displaycomment:PROCEDURE;
  CALL display(HT);CALL display(';');CALL displayspace;
  CALL display$word(nextunasm);CALL displayspace;
  DO n = 0 TO 3;
    IF n < stop
      THEN CALL displaybyte(string(n));
    ELSE DO;
      CALL displayspace;CALL displayspace;
      END;
    END;
  END displaycomment;

/***** ASSEMBLE COMMAND *****/
/* Assemble */
assemble:PROCEDURE PUBLIC;
  CALL findaddress;CALL findnul;
  CALL saveregister;
  IF numword = 1
    THEN locationcounter = paraword(0);
  readasm:DO WHILE 1;
    CALL display$word(locationcounter);
    CALL displayspace;CALL display(ACK);CALL displayspace;
    CALL readstring;
    DO WHILE n < 64;
      CALL displayspace;
      n = n + 1;
      END;
    n = 0;CALL skipblank;
    IF string(n) <> 0
      THEN DO;
        CALL findmnemonic;
        IF function <> err
          THEN DO;
            IF numword < 67 /* mnemonic number */
              THEN DO;
                CALL findoperand;
                END;
            ELSE IF numword = 67 /* DB */
              THEN DO;
                numbyte = 0;CALL findlist;CALL findend;
                IF(numbyte > 0)AND(function <> err)
                  THEN DO;
                    DO n = 0 TO numbyte - 1;
                      CALL poke(locationcounter + n,string(n));
                      END;
                    locationcounter = locationcounter + numbyte;
                    END;
                END;
            ELSE IF numword = 68 /* DW */
              THEN DO;
                CALL findhex(4);CALL findend;
                IF(pass = 1)AND(function <> err)
                  THEN DO;
                    CALL poke(locationcounter,LOW(addr));
                    CALL poke(locationcounter +1,HIGH(addr));
                    locationcounter = locationcounter + 2;
                    END;
                END;
            END;
            IF function = err
              THEN CALL disparray(.('endline,' Error Code',ETX));
            CALL newline;
            END;
          ELSE DO;
            CALL newline;
            UNDO readasm;
            END;
          END;
        END assemble;

```

```

/* Unassemble */
unassemble:PROCEDURE PUBLIC;
  IF upcase(beginnextword) = 'S'
  THEN DO;
    unasmtosrc = true;
    n = n + 1;
  END;
  ELSE unasmtosrc = false;
  CALL findrange;CALL findnul;
  CALL saveregister;
  IF numword > 0
  THEN nextunasm = paraword(0);
  ELSE paraword(0) = 0;
  IF numword < 2
  THEN paraword(1) = 0ffffh;
  numline = 0; /* set number of lines count */
  IF paraword(0) <= paraword(1)
  THEN REPEAT;
    CALL findpointer;
    IF ptr <> 0
    THEN DO;
      ptr = ptr - 1;
      stop = insbyte(ptr);
      IF unasmtosrc
      THEN CALL displaysourceinstruction;
      ELSE CALL displayinstruction;
    END;
    ELSE DO; /* error code -> DB xx[,xx[,xx[,xx]]] */
      IF (string(0) = 0cbh)OR(string(0) = 0edh)
      THEN stop = 2;
      ELSE IF (string(0) = 0ddh)OR(string(0) = 0fdh)
      THEN DO;
        IF string(1) = 0cbh
        THEN stop = 4;
        ELSE stop = 1;
      END;
      IF unasmtosrc
      THEN CALL displaysourcedefindbyte;
      ELSE CALL displaydefindbyte;
    END;
    DO WHILE numdigit < 11;
      CALL displayspace;
      numdigit = numdigit + 1;
    END;
    IF unasmtosrc
    THEN CALL displaycomment;
    CALL newline;
    IF numword < 2
    THEN numline = numline + 1;
    nextunasm = nextunasm + stop;
    UNTIL (CARRY) OR (numline = 11)
    OR (nextunasm > paraword(1)) ENDREPEAT;
  ELSE function = err;
  END unassemble;

/***** BREAKPOINT UTILITY *****/
/* display breakpoint */
displaybreakpoint:PROCEDURE PUBLIC;
  IF NOT(bp)
  THEN CALL disparray(.'Break Point Clear',ETX);
  ELSE DO;
    CALL disparray(.'Break Point Address ',ETX);
    CALL displayword(breakaddress);
  END;
  CALL newline;
  END displaybreakpoint;

/* set breakpoint */
setbreakpoint:PROCEDURE PUBLIC;
  bp = true;
  breakaddress = paraword(0);
  END setbreakpoint;

/* clear breakpoint */
clearbreakpoint:PROCEDURE PUBLIC;
  bp = false;
  END clearbreakpoint;

/* Breakpoint */
breakpoint:PROCEDURE PUBLIC;
  numword,pass = 0;
  CALL findaddress;CALL findnul;
  IF numword = 0
  THEN CALL display$breakpoint;
  ELSE CALL set$breakpoint;
  END breakpoint;

saveregister:PROCEDURE PUBLIC;
  END saveregister;

```

```

peekmoni:PROCEDURE(location) BYTE PUBLIC;
  DECLARE
    location ADDRESS,
    mem BASED location BYTE;
  RETURN mem;
  END peekmoni;

pokemoni:PROCEDURE(location,content) PUBLIC;
  DECLARE
    location ADDRESS,
    content BYTE,
    mem BASED location BYTE;
  mem = content;
  END pokemoni;

peekport:PROCEDURE(location) BYTE PUBLIC;
  DECLARE location ADDRESS;
  ramcode(0) = 0dbh;          /* IN location */
  ramcode(1) = LOW(location);
  ramcode(2) = 0c9h;         /* RET          */
  CALL beginramcode;
  END peekport;

pokeport:PROCEDURE(location,content) PUBLIC;
  DECLARE location ADDRESS,content BYTE;
  ramcode(0) = 03eh;         /* MVI A,content */
  ramcode(1) = content;
  ramcode(2) = 0d3h;         /* OUT location */
  ramcode(3) = LOW(location);
  ramcode(4) = 0c9h;         /* RET          */
  CALL beginramcode;
  END pokeport;

pokereg:PROCEDURE(spec,number,content) PUBLIC;
  DECLARE
    (spec,number) BYTE,
    content ADDRESS;
  DO case spec;
    /* 0 = register low */
    LOW(reg(number)) = LOW(content);
    /* 1 = register high */
    HIGH(reg(number)) = LOW(content);
    /* 2 = register pair */
    reg(number) = content;
  END;
  END pokereg;

resetprocessor:PROCEDURE PUBLIC;
  nextunasm = 0; /* after reset Z80 */
  reg(0) = 0; /* PC = 0 */
  reg(5) = 0; /* I = 0 */
  DISABLE; /* IFF = 0 */
  END resetprocessor;

/***** PROCESSOR UTILITY *****/
displayallreg:PROCEDURE;
  CALL displayspace;
  CALL displayword(reg(0)); /* PC */
  CALL displayspace;
  CALL displaybyte(HIGH(reg(1))); /* A */
  tmp = LOW(reg(1));
  CALL displayspace;
  CALL displaybyte(tmp); /* F */
  CALL displayspace;
  CALL display(bit(tmp,7)+30h); /* S */
  CALL display(bit(tmp,6)+30h); /* Z */
  CALL display(bit(tmp,4)+30h); /* H */
  CALL display(bit(tmp,2)+30h); /* P */
  CALL display(bit(tmp,1)+30h); /* N */
  CALL display(bit(tmp,0)+30h); /* C */
  DO tmp = 2 TO 4;
    CALL displayspace;
    CALL displaybyte(HIGH(reg(tmp))); /* B D H */
    CALL displayspace;
    CALL displaybyte(LOW(reg(tmp))); /* C E L */
  END;
  CALL displayspace;
  CALL displaybyte(HIGH(reg(5))); /* I */
  CALL displayspace;
  CALL display(bit(reg(5),0)+30h); /* IFF */
  DO tmp = 6 TO 12;
    CALL displayspace;
    CALL displayword(reg(tmp));
  END;
  CALL newline;
  END displayallreg;

/* read data to change register pair */

```

```

readreg:PROCEDURE(spec,number);
  DECLARE (spec,number) BYTE;
  /* 0,1,2 register byte or word */
  IF spec <= 2
  THEN DO;
    CALL readnibble((spec AND 10b) + 2);
    CALL checkdata((spec AND 10b) + 2);
    IF (pass = 1)OR(pass = -1)
    THEN CALL pokereg(spec,number,addr);
    ELSE IF pass = 0
    THEN GOTO prompt;
    END;
  ELSE IF spec >= 10h
  THEN DO;
    CALL readbit(1);
    IF string(0) = 0
    THEN GOTO prompt;
    IF string(0) = '0'
    THEN CALL pokereg(0,number,clearbit(reg(number),spec - 10h));
    IF string(0) = '1'
    THEN CALL pokereg(0,number,setbit(reg(number),spec - 10h));
    END;
  END readreg;

/* find register name */
findregname:PROCEDURE;
  CALL skipblank;
  numbyte,pass = 0;
  IF string(n)<>0
  THEN DO;
    REPEAT;
      LOW(paraword(numbyte)) = upcase(string(n));
      numbyte = numbyte + 1;n = n + 1;
    UNTIL(string(n) = ' 'OR(string(n) = 0)OR(numbyte = 3)ENDREPEAT;
    DO WHILE numbyte < 3;
      LOW(paraword(numbyte)) = ' ';
      numbyte = numbyte + 1;
    END;
    numword,numbyte = 0;
    REPEAT;
      IF LOW(paraword(numbyte)) = regname(numword).char(numbyte)
      THEN numbyte = numbyte + 1;
      ELSE DO;
        numbyte = 0;
        numword = numword + 1;
      END;
    UNTIL(numbyte = 3)OR(numword > LAST(regname))ENDREPEAT;
    IF numword > LAST(regname)
    THEN function = err;
    ELSE pass = 1;
    END;
  END findregname;

/* display and read one register */
oneregister:PROCEDURE;
  DO n = 0 TO 2;
    CALL display(regname(numword).char(n));
    END;
  CALL displayspace;
  IF regspect(numword).spec = 0
  THEN CALL displaybyte(LOW(reg(regspect(numword).number)));
  ELSE IF regspect(numword).spec = 1
  THEN CALL displaybyte(HIGH(reg(regspect(numword).number)));
  ELSE IF regspect(numword).spec = 2
  THEN CALL displayword(reg(regspect(numword).number));
  ELSE IF regspect(numword).spec >= 10h
  THEN CALL display(
    bit(
      LOW(reg(regspect(numword).number)),regspect(numword).spec - 10h
    ) + 30h );
  CALL newline;CALL display(ACK);
  CALL readreg(regspect(numword).spec,regspect(numword).number);
  CALL newline;
  END oneregister;

/* display and read all registers */
allregister:PROCEDURE PUBLIC;
  CALL disparray.allregname);
  CALL displayallreg;
  END allregister;

/***** PROCESSOR COMMAND *****/
/* Input */
inputcommand:PROCEDURE PUBLIC;
  CALL findaddress;CALL findnul;
  IF numword = 1 AND paraword(0) < 100h
  THEN DO;
    CALL saveregister;
    CALL display$byte(peekport(paraword(0)));
    CALL newline;
  
```

```

    END;
    ELSE function = err;
    END inputcommand;

/* Outputcommand */
outputcommand:PROCEDURE PUBLIC;
CALL findaddress;CALL findbyte;CALL findnul;
IF numword = 1 AND paraword(0) < 100h
AND numbyte = 1
THEN DO;
    CALL saveregister;
    CALL pokeport(paraword(0),string(0));
    END;
ELSE function = err;
END outputcommand;

/* Register */
register:PROCEDURE PUBLIC;
CALL findregname;CALL findnul;
CALL saveregister;
IF pass = 1
THEN CALL oneregister;
ELSE CALL allregister;
END register;

/* Trace Command */
tracecommand:PROCEDURE PUBLIC;
CALL findnul;
/* execute with breakpoint at PC */
PORT8255A = HIGH(reg(0));/* out A15 - A8 */
PORT8255B = LOW(reg(0));/* out A7 - A0 */
PORT8255C = 0000$0000b; /* enable G08255 and EN688 */
CALL gopc;
END tracecommand;

/* Go */
gocommand:PROCEDURE PUBLIC;
CALL findaddress;CALL findnul;
IF numword = 1 /* set new PC */
THEN reg(0) = paraword(0);
IF NOT bp
THEN DO; /* execute without break */
    PORT8255C = 0000$0001b; /* enable G08255 ,diabile EN688 */
    END;
ELSE DO; /* execute with breakpoint */
    PORT8255A = HIGH(breakaddress);/* out A15 - A8 */
    PORT8255B = LOW(breakaddress);/* out A7 - A0 */
    PORT8255C = 0000$0000b; /* enable G08255 and EN688 */
    END;
CALL gopc;
END gocommand;

/***** SOFTWARE UTILITY *****/
/* set error while find Intel hex record */
seterror:PROCEDURE;
function = err;
GOTO prompt2;
END seterror;

/* skip blank in input-string */
skipblank:PROCEDURE PUBLIC;
DO WHILE string(n)=' ';
    n = n + 1;
END;
END skipblank;

/* return upper case */
upcase:PROCEDURE(char) BYTE PUBLIC;
DECLARE char BYTE;
IF(char >= 'a')AND(char <= 'z')
THEN char = char - 20h;
RETURN char;
END upcase;

/* return bit value */
bit:PROCEDURE(onebyte,count) BYTE PUBLIC;
DECLARE (onebyte,count) BYTE;
RETURN (SHR(onebyte,count) AND 01b);
END bit;

/* set bit */
setbit:PROCEDURE(onebyte,count) BYTE PUBLIC;
DECLARE (onebyte,count) BYTE;
RETURN (onebyte OR ROL(0000$0001b,count));
END setbit;

/* clear bit */
clearbit:PROCEDURE(onebyte,count) BYTE PUBLIC;
DECLARE (onebyte,count) BYTE;
RETURN (onebyte AND ROL(1111$1110b,count));

```

```

END clearbit;

/* initialize value */
initvalue:PROCEDURE PUBLIC;
DECLARE checkstart BYTE;
/* warmstart */
CALL setrs232;
locationcounter,function,bp = 0;
lastdump = 0ffffh;
IF checkstart <> (NOT(LOW(.checkstart)))
THEN DO; /* coldstart */
DO n = 0 TO 12;
    reg(n) = 0;
    END;
CALL newline;
CALL disparray(.icename);
CALL disparray(.help);
checkstart = (NOT(LOW(.checkstart)));
END;
CALL resetprocessor;
END initvalue;

/* clear string for input */
clearstring:PROCEDURE PUBLIC;
DO n = 0 TO 63;
    string(n) = 0;
    END;
END clearstring;

/* read input from keyboard until press RETURN */
readstring:PROCEDURE PUBLIC;
CALL clearstring;n = 0;function = 0;
REPEAT;
    character = readkbd;
    IF (character>=' ')AND(character<7fh)
    THEN DO;
        IF n < 63 /* max = 63 character */
        THEN DO;
            CALL display(character);
            string(n) = character;
            n = n + 1;
            END;
        ELSE CALL display(BEL);
        END;
    IF (character=BS)AND(n>0)
    THEN DO;
        CALL display(BS);
        CALL displayspace;
        CALL display(BS);
        string(n-1) = 0;
        n = n - 1;
        END;
    IF character=ESC
    THEN GOTO prompt;
    UNTIL character=CR ENDREPEAT;
END readstring;

readline:PROCEDURE PUBLIC;
CALL readstring;
CALL newline;
END readline;

/* find function */
findfunction:PROCEDURE PUBLIC;
n,numword,numbyte = 0;
CALL skipblank;
IF (string(n)<>0)AND(function=0)
THEN DO;
    character = upcase(string(n));
    IF(character >= err)AND(character <= 'Z')
    THEN function = character;
    ELSE function =err;
    n = n + 1;
    END;
END findfunction;

/* find hex value */
findhex:PROCEDURE(count) PUBLIC;
DECLARE count BYTE;
numdigit,pass,addr = 0;
DO WHILE (string(n)=' ')OR(string(n)=',');
    n = n + 1;
    END;
IF string(n)<>0
THEN REPEAT;
    IF (string(n)>='0')AND(string(n)<='9')
    OR (string(n)>='A')AND(string(n)<='F')
    OR (string(n)>='a')AND(string(n)<='f')
    THEN DO;
        numdigit = numdigit + 1; pass = 1;

```

```

    IF (string(n)>='0')AND(string(n)<='9')
    THEN digit = string(n) - 30h;
    IF (string(n)>='A')AND(string(n)<='F')
    THEN digit = string(n) - 37h;
    IF (string(n)>='a')AND(string(n)<='f')
    THEN digit = string(n) - 57h;
    IF numdigit <= count
    THEN addr = addr * 16 + digit;
    ELSE function = err;
    n = n + 1;
    END;
  UNTIL
  NOT((string(n)>='0')AND(string(n)<='9')
  OR (string(n)>='A')AND(string(n)<='F')
  OR (string(n)>='a')AND(string(n)<='f'))ENDREPEAT;
END findhex;

/* return next character */
nextcharacter:PROCEDURE BYTE PUBLIC;
  RETURN string(n);
  END nextcharacter;

/* return begin character of next word */
beginnextword:PROCEDURE BYTE PUBLIC;
  DO WHILE (string(n)=' ')OR(string(n)=','');
  n = n + 1;
  END;
  RETURN string(n);
  END beginnextword;

/* find byte */
findbyte:PROCEDURE PUBLIC;
  CALL findhex(2);
  string(numbyte) = addr;
  numbyte = numbyte + pass;
  END findbyte;

/* find address */
findaddress:PROCEDURE PUBLIC;
  CALL findhex(4);
  paraword(numword) = addr;
  numword = numword + pass;
  END findaddress;

/* find address or range */
findrange:PROCEDURE PUBLIC;
  CALL findaddress;
  character = beginnextword;
  IF pass = 1
  THEN IF (character='L')OR(character='l')
  THEN DO;
    n = n + 1;
    CALL findhex(4);
    IF (pass = 0)OR(pass = 1)AND(addr = 0)
    THEN function = err;
    paraword(numword)=paraword(numword-1) + addr - 1;
    numword = numword + pass;
    END;
  ELSE CALL findaddress;
  END findrange;

/* find string in quotation marks */
findstringinquote:PROCEDURE(mark) PUBLIC;
  DECLARE mark BYTE;
  REPEAT;
  n = n + 1;/* skip begin mark and get next character */
  IF (string(n)>=' ')AND(string(n)<7fh)AND(string(n)<>mark)
  THEN DO;
    string(numbyte) = string(n);
    numbyte = numbyte + 1;
    END;
  IF string(n)=0
  /* end of line without end mark -> error */
  THEN function = err;
  UNTIL (string(n)=mark)OR(function = err) ENDREPEAT;
  n = n + 1;/* skip end mark */
  END findstringinquote;

/* find list */
findlist:PROCEDURE PUBLIC;
  REPEAT;
  character = beginnextword;
  IF character = 27h
  /* string in single quotation marks */
  THEN CALL findstringinquote(27h);
  ELSE IF character = 22h
  /* string in double quotation marks */
  THEN CALL findstringinquote(22h);
  ELSE IF character <> 0
  /* a byte in hex digit form */

```

```

    THEN DO;
      CALL findbyte;
      IF pass = 0
        THEN DO;
          function = err;
          n = n + 1;
        END;
      END;
    UNTIL (string(n)=0)OR(function = err) ENDREPEAT;
  END findlist;

/* find nul */
findnul:PROCEDURE PUBLIC;
  IF beginnextword <> 0
    THEN function = err;
  IF function = err
    THEN DO;
      GOTO prompt2;
    END;
  END findnul;

readbytehyphen:PROCEDURE PUBLIC;
  CALL readnibble(2);
  DO n = 0 TO 2;
    IF string(n) = '-'
      THEN CALL display('-');
    END;
  CALL newline;
  END readbytehyphen;

/* read nibble */
readnibble:PROCEDURE(count) PUBLIC;
  DECLARE count BYTE;
  CALL clearstring;
  n = 0;
  REPEAT;
    character = upcase(readkbd);
    IF(n < count) AND
      ( (character>='0')AND(character<='9')
        OR(character>='A')AND(character<='F'))
      THEN DO;
        CALL display(character);
        string(n) = character;
        n = n + 1;
      END;
    IF (character=BS)AND(n>0)
      THEN DO;
        CALL display(BS);CALL displayspace;CALL display(BS);
        string(n-1) = 0;
        n = n - 1;
      END;
    IF (character=' ')
      OR (character='-')
      THEN string(n) = character;
    IF (character=ESC)
      THEN GOTO prompt;
    UNTIL (character=CR)
      OR (character=' ')
      OR (character='-')ENDREPEAT;
  DO WHILE n <= count;
    CALL displayspace;
    n = n + 1;
  END;
  END readnibble;

/* read bit */
readbit:PROCEDURE(count) PUBLIC;
  DECLARE count BYTE;
  DO n = 0 TO count;
    string(n) = 0;
  END;
  n = 0;
  REPEAT;
    character = readkbd;
    IF (n < count) AND
      ( (character='0')
        OR(character='1')
        OR(character=' '))
      THEN DO;
        CALL display(character);
        string(n) = character;
        n = n + 1;
      END;
    IF (character=BS)AND(n>0)
      THEN DO;
        CALL display(BS);
        CALL displayspace;
        CALL display(BS);
        string(n-1) = 0;
        n = n - 1;
      END;
  END;

```



```

    END;
    IF (character=ESC)
    THEN GOTO prompt;
    UNTIL(n=count)AND(character=' ')
    OR (character=CR) ENDREPEAT;
DO WHILE n <= count;
    CALL displayspace;
    n = n + 1;
    END;
END readbit;

/* check data */
checkdata:PROCEDURE(count) PUBLIC;
    DECLARE count BYTE;
    n = 0;
    CALL findhex(count);
    IF (pass=0)AND(string(0)=' ')
    THEN pass='+';
    IF (pass=0)AND(string(0)='-')
    THEN pass='-';
    IF (pass=1)AND(string(n)='-')
    THEN pass=-1;
    END checkdata;

/* display 16 byte from memory */
display$line:PROCEDURE PUBLIC;
    CALL display$word(addr);CALL displayspace;
    DO n = 0 TO 0fh ;
        ptr = addr + n;
        IF (n < start) OR (n > stop)
        THEN DO;
            CALL displayspace;CALL displayspace;CALL displayspace;
            END;
        ELSE IF (n = 8) AND (n <> start)
        THEN DO;
            string(n) = peek(ptr);
            CALL display('-');CALL display$byte(string(n));
            END;
        ELSE DO;
            string(n) = peek(ptr);
            CALL displayspace;CALL display$byte(string(n));
            END;
        END;
    CALL displayspace;CALL displayspace;CALL displayspace;
    DO n = 0 TO stop;
        ptr = addr + n;
        IF n < start
        THEN CALL displayspace;
        ELSE IF (string(n)>=' ')AND(string(n)<7fh)
        THEN CALL display(string(n));
        ELSE CALL display('.');
        END;
    lastdump = ptr;
    CALL newline;
    END display$line;

/* display 16 byte from memory in Intel HEX-ASCII format */
displayintelhexline:PROCEDURE PUBLIC;
    sum = stop - start + 1;
    CALL display(':');CALL display$byte(sum);
    ptr = addr + start;
    sum = sum + HIGH(ptr) + LOW(ptr);
    CALL display$word(ptr);CALL display('0');CALL display('0');
    DO n = start TO stop;
        tmp = peek(addr + n);
        sum = sum + tmp;
        CALL display$byte(tmp);
        END;
    CALL display$byte( - sum);
    CALL newline;
    END displayintelhexline;

/* read Intel hex line */
readIntelhex:PROCEDURE PUBLIC;
/* Intel HEX-ASCII format
: B C A A A T T H H ... H H C C
|_| |_| |_| |_| |_| |_| |_|
|_| |_| |_| |_| |_| |_| |_|
|_| |_| |_| |_| |_| |_| |_|
|_| |_| |_| |_| |_| |_| |_|
checksum : checksum + sum = 0
data bytes : 2 characters per byte
record type: 00 = data,01 = end of file
address : first address
byte count : number of data bytes

```

data used ASCII 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F
 NUL (00) and DEL (7F) are ignored.

example
 :100000031650221650236003A6502FE01D21B000D
 :100010003E04D39021650234C3080021650236FFF7
 :100020003A6502FE01D233003E05D39021650234C9

```

:10003000C3200021650236003A6502FE02D24B0061
:100040003E06D39021650234C33800216502360094
:100050003A6502FE01D263003E04D390216502346A
:05006000C35000FB7617
:00000001FF
*** byte count in end of file record must be 00
*/

```

```

/* wait for start of record ':' */
DO WHILE read$with$timeout <> ':';
  END;
sum = 0;
/* read byte count */
/* number of data buffer is 64 bytes */
numbyte = readdatabyte;
IF numbyte > 64
  THEN CALL loaderror;
/* read address */
HIGH(addr) = readdatabyte;
LOW(addr) = readdatabyte;
/* read record type */
tmp = readdatabyte;
IF tmp = 00
  THEN end$of$file = false;
ELSE IF (tmp = 01)AND(numbyte = 0)
  THEN end$of$file = true;
ELSE CALL loaderror;
DO n = 1 TO numbyte;
  /* fill data byte into buffer */
  string(n - 1) = readdatabyte;
  END;
/* checksum */
tmp = readdatabyte;
IF sum <> 0
  THEN CALL loaderror;
END readintelhex;

```

```

/* write memory */
writememory:PROCEDURE PUBLIC;
  CALL saveregister;
  DO n = 1 TO numbyte;
    CALL poke(addr + n - 1,string(n - 1));
  END;
END writememory;

```

```

/* write monitormemory */
writemoni:PROCEDURE PUBLIC;
  CALL saveregister;
  DO n = 1 TO numbyte;
    CALL pokemoni(addr + n - 1,string(n - 1));
  END;
END writemoni;

```

```

/* read data byte */
readdatabyte:PROCEDURE BYTE PUBLIC;
  repeat1:
  character = read$with$timeout;
  IF (character>='0')AND(character<='9')
    THEN digit = character - 30h;
  ELSE IF (character>='A')AND(character<='F')
    THEN digit = character - 37h;
  ELSE IF (character<>NUL)AND(character<>DEL)
    THEN CALL loaderror;
  ELSE GOTO repeat1;
  repeat2:
  character = read$with$timeout;
  IF (character>='0')AND(character<='9')
    THEN digit = SHL(digit,4) + character - 30h;
  ELSE IF (character>='A')AND(character<='F')
    THEN digit = SHL(digit,4) + character - 37h;
  ELSE IF (character<>NUL)AND(character<>DEL)
    THEN CALL loaderror;
  ELSE GOTO repeat2;
  sum = sum + digit;
  RETURN digit;
END readdatabyte;

```

```

/* find data byte */
finddatabyte:PROCEDURE BYTE PUBLIC;
  character = string(n);n = n + 1;
  IF (character>='0')AND(character<='9')
    THEN digit = character - 30h;
  ELSE IF (character>='A')AND(character<='F')
    THEN digit = character - 37h;
  ELSE CALL seterror;
  character = string(n);n = n + 1;
  IF (character>='0')AND(character<='9')
    THEN digit = SHL(digit,4) + character - 30h;
  ELSE IF (character>='A')AND(character<='F')
    THEN digit = SHL(digit,4) + character - 37h;
  ELSE CALL seterror;

```



```

END readkbd;

/* display character from ASCII code */
display:PROCEDURE(chr) PUBLIC;
  DECLARE chr BYTE;
  REPEAT;
    UNTIL TxRdy ENDREPEAT;
  data8251 = chr;
  END display;

/* display space */
displayspace:PROCEDURE PUBLIC;
  CALL display(' ');
  END displayspace;

/* display new line */
newline:PROCEDURE PUBLIC;
  CALL display(CR);
  CALL display(LF);
  END newline;

/* display array of characters */
disparray:PROCEDURE(startaddress) PUBLIC;
  DECLARE
    (startaddress,count) ADDRESS,
    char BYTE BASED count;
  count = startaddress;
  REPEAT;
    CALL display(char);
    count = count + 1;
  UNTIL char = ETX ENDREPEAT;
  END disparray;

/* display byte in hexadecimal format */
display$byte:PROCEDURE(onebyte) PUBLIC;
  DECLARE onebyte BYTE;
  /* display high nibble */
  IF SHR(onebyte,4) < 10
  THEN CALL display(SHR(onebyte,4) + 30h);
  ELSE CALL display(SHR(onebyte,4) + 37h);
  /* display low nibble */
  IF (onebyte AND 0fh) < 10
  THEN CALL display((onebyte AND 0fh) + 30h);
  ELSE CALL display((onebyte AND 0fh) + 37h);
  END display$byte;

/* display word in hexadecimal format */
display$word:PROCEDURE(oneword) PUBLIC;
  DECLARE oneword ADDRESS;
  /* display high byte */
  CALL display$byte(HIGH(oneword));
  /* display low byte */
  CALL display$byte(LOW(oneword));
  END display$word;

/***** COMMAND PROCEDURE *****/
/*
j:01xxx00ncc
01 = number of data = 1 (nn)
xxx = start address
00 = record type data
nn = data = number of data to be shown
cc = checksum
*/
jmenu:PROCEDURE;
  CALL findIntelhex;CALL findnul;
  IF numbyte = 1
  THEN DO;
    CALL display(':');CALL display$byte(string(0));
    sum = string(0) + HIGH(addr) + LOW(addr);
    CALL display$word(addr);CALL display('0');CALL display('0');
    DO n = 1 TO string(0);
      tmp = peek(addr + n - 1);
      sum = sum + tmp;
      CALL display$byte(tmp);
    END;
    CALL display$byte( - sum);
    CALL newline;
  END;
  ELSE function = err;
  END jmenu;

kmenu:PROCEDURE;
  CALL findIntelhex;CALL findnul;
  CALL writememory;
  END kmenu;

ymenu:PROCEDURE;
  CALL findIntelhex;CALL findnul;
  IF numbyte = 1

```

```

THEN DO;
  CALL display(':');CALL display$byte(string(0));
  sum = string(0) + HIGH(addr) + LOW(addr);
  CALL display$word(addr);CALL display('0');CALL display('0');
  DO n = 1 TO string(0);
    tmp = peekmoni(addr + n - 1);
    sum = sum + tmp;
    CALL display$byte(tmp);
  END;
  CALL display$byte(- sum);
  CALL newline;
  END;
ELSE function = err;
END ymenu;

zmenu:PROCEDURE;
CALL findIntelhex;CALL findnul;
CALL writemoni;
END zmenu;

/* Dump */
dump:PROCEDURE;
CALL findrange;CALL findnul;
DO CASE numword;
  /* no paraword -> display 40h bytes after last dump */
  DO;
    paraword(0)=lastdump+1;
    IF paraword(0)<0ff80h
      THEN paraword(1)=paraword(0)+7fh;
      ELSE paraword(1)=0ffffh;
    END;
  /* 1 paraword -> display 40h bytes */
  DO;
    IF paraword(0)<0ff80h
      THEN paraword(1)=paraword(0)+7fh;
      ELSE paraword(1)=0ffffh;
    END;
  /* 2 parawords -> display memory */
  ;
  /* 3 parawords -> error */
  function = err;
  END;
  /* display memory */
  IF function <> err
  THEN DO;
    IF paraword(1) >= paraword(0)
    THEN DO;
      CALL saveregister;
      addr = paraword(0) AND 0fff0h;
      REPEAT;
        IF paraword(0)-addr <= 0fh
          THEN start = paraword(0) - addr;
          ELSE start = 00h;
        IF paraword(1)-addr <= 0fh
          THEN stop = paraword(1) - addr;
          ELSE stop = 0fh;
        CALL display$line;
        addr = addr + 10h;
        UNTIL(CARRY)
          OR(addr > (paraword(1) AND 0fff0h))ENDREPEAT;
      END;
    ELSE function = err;
    END;
  END dump;

/* Enter */
enter:PROCEDURE;
CALL findaddress;CALL findlist;CALL findnul;
IF (numword = 1)AND(numbyte = 0) /* display/enter byte by byte */
THEN DO;
  CALL saveregister;
  REPEAT;
    CALL display$word(paraword(0));
    CALL displayspace;CALL displayspace;
    CALL display$byte(peek(paraword(0)));
    CALL displayspace;CALL display(ACK);CALL displayspace;
    /* read new data from keyboard */
    CALL readbytehyphen;
    CALL checkdata(2);
    IF pass=1
    THEN DO;
      CALL poke(paraword(0),LOW(addr));
      paraword(0) = paraword(0) + 1;
      END;
    ELSE IF pass=-1
    THEN DO;
      CALL poke(paraword(0),LOW(addr));
      paraword(0) = paraword(0) - 1;
      END;
    ELSE IF pass='+'

```

```

        THEN paraword(0) = paraword(0) + 1;
        ELSE IF pass='- '
        THEN paraword(0) = paraword(0) - 1;
        UNTIL pass=0 ENDREPEAT;
    END;
ELSE IF numword = 1 /* enter list */
THEN DO;
    CALL saveregister;
    DO n = 0 TO numbyte - 1;
        CALL poke(paraword(0) + n ,string(n));
    END;
END;
ELSE function = err;
END enter;

/* Fill memory block with list */
fill:PROCEDURE;
CALL findrange;CALL findlist;CALL findnul;
IF (numword=2)AND(paraword(1)>=paraword(0))AND(numbyte>0)
THEN DO;
    CALL saveregister;
    ptr = paraword(0);n = 0;
    REPEAT;
        CALL poke(ptr,string(n));
        n = n + 1;
        IF n = numbyte
        THEN n = 0;
        ptr = ptr + 1;
    UNTIL(ptr - 1) = paraword(1) ENDREPEAT;
END;
ELSE function = err;
END fill;

/* Load command */
load:PROCEDURE;
CALL findnul;
/* send load guide */
CALL disparray(.loadguide);
/* wait for first character */
CALL wait$first;
REPEAT;
    /* read Intel hex line */
    CALL read$intel$hex;
    /* write memory */
    CALL writememory;
    /* send ACK */
    CALL display(ACK);
    UNTIL end$of$file ENDREPEAT;
nextunasm,reg(0) = addr;
/* send load ready */
CALL disparray(.loadready);
END load;

/* Write */
write:PROCEDURE;
CALL findrange;CALL findnul;
IF (numword=2)AND(paraword(1)>=paraword(0))
THEN DO;
    CALL saveregister;
    CALL disparray(.write$guide);
    character = readkbd;
    addr = paraword(0) AND 0ff0h;
    REPEAT;
        IF paraword(0)-addr <= 0fh
        THEN start = paraword(0) - addr;
        ELSE start = 00h;
        IF paraword(1)-addr <= 0fh
        THEN stop = paraword(1) - addr;
        ELSE stop = 0fh;
        CALL displayintelhexline;
        addr = addr + 10h;
    UNTIL(CARRY)
    OR(addr > (paraword(1) AND 0ff0h))ENDREPEAT;
    CALL disparray(.intelhexend);
END;
ELSE function = err;
END write;

/***** MAIN PROGRAM *****/
init:CALL initvalue;
prompt:CALL newline;
prompt2:
DO WHILE 1;
    IF function = err
    THEN DO;
        CALL disparray(.error);
        CALL display(NAK);
        END;
    ELSE CALL display(ACK);
    CALL display('>');

```

```

CALL readline;
CALL findfunction;
IF function <> 0
THEN DO CASE function - 'a';
; /* a error */
CALL assemble; /* A assemble */
CALL breakpoint; /* B breakpoint */
CALL clearbreakpoint; /* C clearbreakpoint */
CALL dump; /* D dump */
CALL enter; /* E enter */
CALL fill; /* F fill */
CALL goccommand; /* G go */
DO;
CALL dispparray(.help);
CALL newline;
END; /* H help menu */
CALL inputcommand; /* I input */
CALL j$menu; /* J menu */
CALL k$menu; /* K menu */
CALL load; /* L load */
function = err; /* M error */
function = err; /* N error */
CALL outputcommand; /* O output */
function = err; /* P error */
function = err; /* Q error */
CALL register; /* R register */
function = err; /* S error */
CALL tracecommand; /* T trace */
CALL unassemble; /* U unassemble */
function = err; /* V error */
CALL write; /* W write */
CALL resetprocessor; /* X reset */
CALL y$menu; /* Y menu */
CALL z$menu; /* Z menu */
END;
END;
END z80;

```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

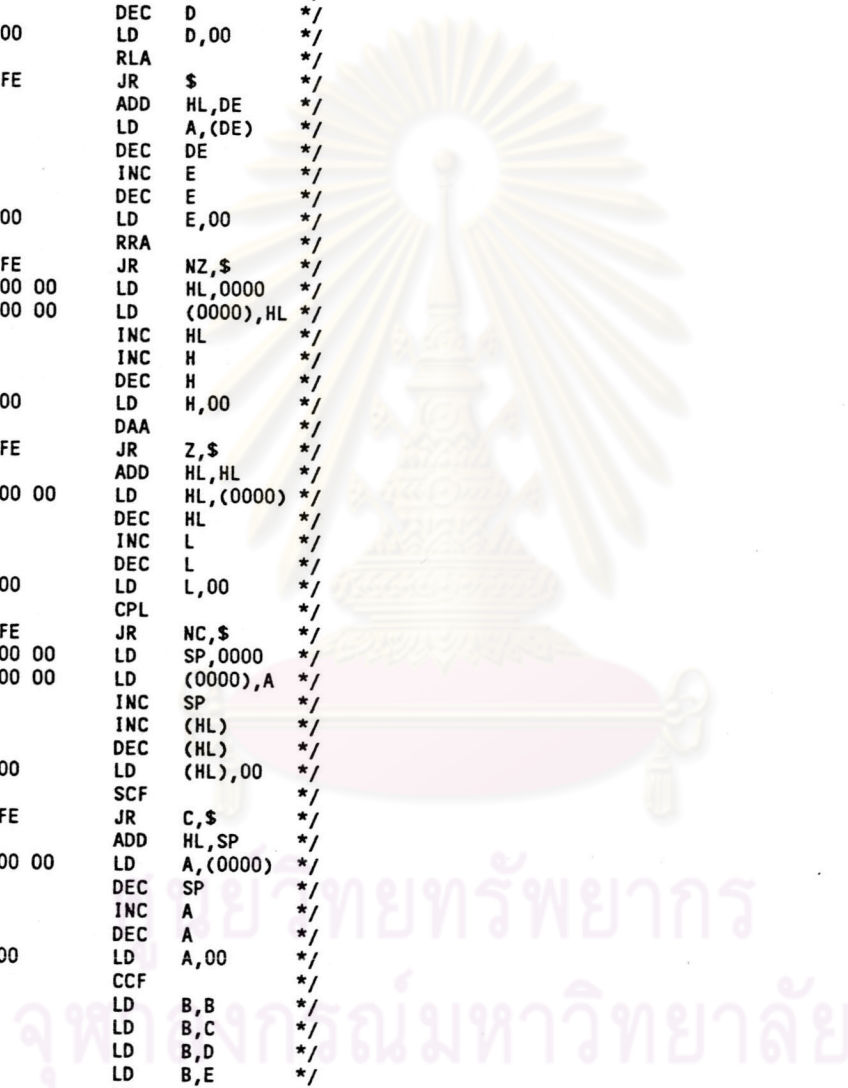
```

table:D0;
$NOLIST
  DECLARE
    mnemonic(69) STRUCTURE(char(4) BYTE) PUBLIC DATA(
$INCLUDE(mnemonic.t80)
    ),
    operand(55) STRUCTURE(char(8) BYTE) PUBLIC DATA(
$INCLUDE(operand.t80)
    ),
    insopc(698) STRUCTURE(opc(3) BYTE) PUBLIC DATA(
$INCLUDE(insopc.t80)
    ),
    insmne(698) BYTE PUBLIC DATA(
$INCLUDE(insmne.t80)
    ),
    insopr(698) STRUCTURE(opr(2) BYTE) PUBLIC DATA(
$INCLUDE(insopr.t80)
    ),
    insbyte(698) BYTE PUBLIC DATA(
$INCLUDE(insbyte.t80)
    ),
    mneinsptr(69) ADDRESS PUBLIC DATA(
$INCLUDE(mne-ins.t80)
    ),
    opcinsptr(256) ADDRESS PUBLIC DATA(
$INCLUDE(opc-ins.t80)
    ),
    cbinsptr(256) ADDRESS PUBLIC DATA(
$INCLUDE(cb-ins.t80)
    ),
    edins(56) STRUCTURE(ptr ADDRESS,opc2 BYTE) PUBLIC DATA(
$INCLUDE(ed-ins.t80)
    ),
    ddcbins(31) STRUCTURE(ptr ADDRESS,opc4 BYTE) PUBLIC DATA(
$INCLUDE(ddcb-ins.t80)
    ),
    fdcbins(31) STRUCTURE(ptr ADDRESS,opc4 BYTE) PUBLIC DATA(
$INCLUDE(fdcb-ins.t80)
    ),
    ddins(40) STRUCTURE(ptr ADDRESS,opc2 BYTE) PUBLIC DATA(
$INCLUDE(dd-ins.t80)
    ),
    fdins(40) STRUCTURE(ptr ADDRESS,opc2 BYTE) PUBLIC DATA(
$INCLUDE(fd-ins.t80)
    );
  END table;

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

368,/*	00	NOP	*/
292,/*	01 00 00	LD BC,0000	*/
231,/*	02	LD (BC),A	*/
199,/*	03	INC BC	*/
198,/*	04	INC B	*/
161,/*	05	DEC B	*/
290,/*	06 00	LD B,00	*/
516,/*	07	RLCA	*/
179,/*	08	EX AF,AF'	*/
27,/*	09	ADD HL,BC	*/
264,/*	0A	LD A,(BC)	*/
162,/*	0B	DEC BC	*/
200,/*	0C	INC C	*/
163,/*	0D	DEC C	*/
303,/*	0E 00	LD C,00	*/
539,/*	0F	RRCA	*/
174,/*	10 FE	DJNZ \$	*/
316,/*	11 00 00	LD DE,0000	*/
232,/*	12	LD (DE),A	*/
202,/*	13	INC DE	*/
201,/*	14	INC D	*/
164,/*	15	DEC D	*/
314,/*	16 00	LD D,00	*/
505,/*	17	RLA	*/
230,/*	18 FE	JR \$	*/
28,/*	19	ADD HL,DE	*/
265,/*	1A	LD A,(DE)	*/
165,/*	1B	DEC DE	*/
203,/*	1C	INC E	*/
166,/*	1D	DEC E	*/
327,/*	1E 00	LD E,00	*/
528,/*	1F	RRA	*/
228,/*	20 FE	JR NZ,\$	*/
340,/*	21 00 00	LD HL,0000	*/
260,/*	22 00 00	LD (0000),HL	*/
205,/*	23	INC HL	*/
204,/*	24	INC H	*/
167,/*	25	DEC H	*/
338,/*	26 00	LD H,00	*/
156,/*	27	DAA	*/
229,/*	28 FE	JR Z,\$	*/
29,/*	29	ADD HL,HL	*/
339,/*	2A 00 00	LD HL,(0000)	*/
168,/*	2B	DEC HL	*/
208,/*	2C	INC L	*/
171,/*	2D	DEC L	*/
356,/*	2E 00	LD L,00	*/
155,/*	2F	CPL	*/
227,/*	30 FE	JR NC,\$	*/
362,/*	31 00 00	LD SP,0000	*/
257,/*	32 00 00	LD (0000),A	*/
209,/*	33	INC SP	*/
194,/*	34	INC (HL)	*/
157,/*	35	DEC (HL)	*/
240,/*	36 00	LD (HL),00	*/
566,/*	37	SCF	*/
226,/*	38 FE	JR C,\$	*/
30,/*	39	ADD HL,SP	*/
269,/*	3A 00 00	LD A,(0000)	*/
172,/*	3B	DEC SP	*/
197,/*	3C	INC A	*/
160,/*	3D	DEC A	*/
279,/*	3E 00	LD A,00	*/
139,/*	3F	CCF	*/
284,/*	40	LD B,B	*/
285,/*	41	LD B,C	*/
286,/*	42	LD B,D	*/
287,/*	43	LD B,E	*/
288,/*	44	LD B,H	*/
289,/*	45	LD B,L	*/
280,/*	46	LD B,(HL)	*/
283,/*	47	LD B,A	*/
297,/*	48	LD C,B	*/
298,/*	49	LD C,C	*/
299,/*	4A	LD C,D	*/
300,/*	4B	LD C,E	*/
301,/*	4C	LD C,H	*/
302,/*	4D	LD C,L	*/
293,/*	4E	LD C,(HL)	*/
296,/*	4F	LD C,A	*/
308,/*	50	LD D,B	*/
309,/*	51	LD D,C	*/
310,/*	52	LD D,D	*/
311,/*	53	LD D,E	*/
312,/*	54	LD D,H	*/
313,/*	55	LD D,L	*/
304,/*	56	LD D,(HL)	*/
307,/*	57	LD D,A	*/
321,/*	58	LD E,B	*/
322,/*	59	LD E,C	*/



323,/*	5A	LD	E,D	*/
324,/*	5B	LD	E,E	*/
325,/*	5C	LD	E,H	*/
326,/*	5D	LD	E,L	*/
317,/*	5E	LD	E,(HL)	*/
320,/*	5F	LD	E,A	*/
332,/*	60	LD	H,B	*/
333,/*	61	LD	H,C	*/
334,/*	62	LD	H,D	*/
335,/*	63	LD	H,E	*/
336,/*	64	LD	H,H	*/
337,/*	65	LD	H,L	*/
328,/*	66	LD	H,(HL)	*/
331,/*	67	LD	H,A	*/
350,/*	68	LD	L,B	*/
351,/*	69	LD	L,C	*/
352,/*	6A	LD	L,D	*/
353,/*	6B	LD	L,E	*/
354,/*	6C	LD	L,H	*/
355,/*	6D	LD	L,L	*/
346,/*	6E	LD	L,(HL)	*/
349,/*	6F	LD	L,A	*/
234,/*	70	LD	(HL),B	*/
235,/*	71	LD	(HL),C	*/
236,/*	72	LD	(HL),D	*/
237,/*	73	LD	(HL),E	*/
238,/*	74	LD	(HL),H	*/
239,/*	75	LD	(HL),L	*/
182,/*	76	HALT		*/
233,/*	77	LD	(HL),A	*/
271,/*	78	LD	A,B	*/
272,/*	79	LD	A,C	*/
273,/*	7A	LD	A,D	*/
274,/*	7B	LD	A,E	*/
275,/*	7C	LD	A,H	*/
277,/*	7D	LD	A,L	*/
266,/*	7E	LD	A,(HL)	*/
270,/*	7F	LD	A,A	*/
20,/*	80	ADD	A,B	*/
21,/*	81	ADD	A,C	*/
22,/*	82	ADD	A,D	*/
23,/*	83	ADD	A,E	*/
24,/*	84	ADD	A,H	*/
25,/*	85	ADD	A,L	*/
16,/*	86	ADD	A,(HL)	*/
19,/*	87	ADD	A,A	*/
5,/*	88	ADC	A,B	*/
6,/*	89	ADC	A,C	*/
7,/*	8A	ADC	A,D	*/
8,/*	8B	ADC	A,E	*/
9,/*	8C	ADC	A,H	*/
10,/*	8D	ADC	A,L	*/
1,/*	8E	ADC	A,(HL)	*/
4,/*	8F	ADC	A,A	*/
681,/*	90	SUB	B	*/
682,/*	91	SUB	C	*/
683,/*	92	SUB	D	*/
684,/*	93	SUB	E	*/
685,/*	94	SUB	H	*/
686,/*	95	SUB	L	*/
677,/*	96	SUB	(HL)	*/
680,/*	97	SUB	A	*/
555,/*	98	SBC	A,B	*/
556,/*	99	SBC	A,C	*/
557,/*	9A	SBC	A,D	*/
558,/*	9B	SBC	A,E	*/
559,/*	9C	SBC	A,H	*/
560,/*	9D	SBC	A,L	*/
551,/*	9E	SBC	A,(HL)	*/
554,/*	9F	SBC	A,A	*/
43,/*	A0	AND	B	*/
44,/*	A1	AND	C	*/
45,/*	A2	AND	D	*/
46,/*	A3	AND	E	*/
47,/*	A4	AND	H	*/
48,/*	A5	AND	L	*/
39,/*	A6	AND	(HL)	*/
42,/*	A7	AND	A	*/
692,/*	A8	XOR	B	*/
693,/*	A9	XOR	C	*/
694,/*	AA	XOR	D	*/
695,/*	AB	XOR	E	*/
696,/*	AC	XOR	H	*/
697,/*	AD	XOR	L	*/
688,/*	AE	XOR	(HL)	*/
691,/*	AF	XOR	A	*/
373,/*	B0	OR	B	*/
374,/*	B1	OR	C	*/
375,/*	B2	OR	D	*/
376,/*	B3	OR	E	*/

377,/*	B4	OR	H	*/
378,/*	B5	OR	L	*/
369,/*	B6	OR	(HL)	*/
372,/*	B7	OR	A	*/
144,/*	B8	CP	B	*/
145,/*	B9	CP	C	*/
146,/*	BA	CP	D	*/
147,/*	BB	CP	E	*/
148,/*	BC	CP	H	*/
149,/*	BD	CP	L	*/
140,/*	BE	CP	(HL)	*/
143,/*	BF	CP	A	*/
488,/*	C0	RET	NZ	*/
393,/*	C1	POP	BC	*/
220,/*	C2 00 00	JP	NZ,0000	*/
225,/*	C3 00 00	JP	0000	*/
133,/*	C4 00 00	CALL	NZ,0000	*/
399,/*	C5	PUSH	BC	*/
26,/*	C6 00	ADD	A,00	*/
541,/*	C7	RST	00	*/
492,/*	C8	RET	Z	*/
484,/*	C9	RET		*/
224,/*	CA 00 00	JP	Z,0000	*/
0,/*	CB			*/
137,/*	CC 00 00	CALL	Z,0000	*/
138,/*	CD 00 00	CALL	0000	*/
11,/*	CE 00	ADC	A,00	*/
543,/*	CF	RST	08	*/
487,/*	DO	RET	NC	*/
394,/*	D1	POP	DE	*/
219,/*	D2 00 00	JP	NC,0000	*/
389,/*	D3 00	OUT	(00),A	*/
132,/*	D4 00 00	CALL	NC,0000	*/
400,/*	D5	PUSH	DE	*/
687,/*	D6 00	SUB	00	*/
545,/*	D7	RST	10	*/
485,/*	D8	RET	C	*/
181,/*	D9	EXX		*/
217,/*	DA 00 00	JP	C,0000	*/
187,/*	DB 00	IN	A,(00)	*/
130,/*	DC 00 00	CALL	C,0000	*/
0,/*	DD			*/
561,/*	DE 00	SBC	A,00	*/
546,/*	DF	RST	18	*/
491,/*	E0	RET	PO	*/
395,/*	E1	POP	HL	*/
223,/*	E2 00 00	JP	PO,0000	*/
176,/*	E3	EX	(SP),HL	*/
136,/*	E4 00 00	CALL	PO,0000	*/
401,/*	E5	PUSH	HL	*/
49,/*	E6 00	AND	00	*/
547,/*	E7	RST	20	*/
490,/*	E8	RET	PE	*/
214,/*	E9	JP	(HL)	*/
222,/*	EA 00 00	JP	PE,0000	*/
180,/*	EB	EX	DE,HL	*/
135,/*	EC 00 00	CALL	PE,0000	*/
0,/*	ED			*/
698,/*	EE 00	XOR	00	*/
548,/*	EF	RST	28	*/
489,/*	F0	RET	P	*/
392,/*	F1	POP	AF	*/
221,/*	F2 00 00	JP	P,0000	*/
173,/*	F3	DI		*/
134,/*	F4 00 00	CALL	P,0000	*/
398,/*	F5	PUSH	AF	*/
379,/*	F6 00	OR	00	*/
549,/*	F7	RST	30	*/
486,/*	F8	RET	M	*/
359,/*	F9	LD	SP,HL	*/
218,/*	FA 00 00	JP	M,0000	*/
175,/*	FB	EI		*/
131,/*	FC 00 00	CALL	M,0000	*/
0,/*	FD			*/
150,/*	FE 00	CP	00	*/
550,/*	FF	RST	38	*/

```

/* ADC */ 1, /* A,(HL) */
/* ADC */ 1, /* A,(IX+b) */
/* ADC */ 1, /* A,(IY+b) */
/* ADC */ 1, /* A,A */
/* ADC */ 1, /* A,B */
/* ADC */ 1, /* A,C */
/* ADC */ 1, /* A,D */
/* ADC */ 1, /* A,E */
/* ADC */ 1, /* A,H */
/* ADC */ 1, /* A,L */
/* ADC */ 1, /* A,b */
/* ADC */ 1, /* HL,BC */
/* ADC */ 1, /* HL,DE */
/* ADC */ 1, /* HL,HL */
/* ADC */ 1, /* HL,SP */
/* ADD */ 2, /* A,(HL) */
/* ADD */ 2, /* A,(IX+b) */
/* ADD */ 2, /* A,(IY+b) */
/* ADD */ 2, /* A,A */
/* ADD */ 2, /* A,B */
/* ADD */ 2, /* A,C */
/* ADD */ 2, /* A,D */
/* ADD */ 2, /* A,E */
/* ADD */ 2, /* A,H */
/* ADD */ 2, /* A,L */
/* ADD */ 2, /* A,b */
/* ADD */ 2, /* HL,BC */
/* ADD */ 2, /* HL,DE */
/* ADD */ 2, /* HL,HL */
/* ADD */ 2, /* HL,SP */
/* ADD */ 2, /* IX,BC */
/* ADD */ 2, /* IX,DE */
/* ADD */ 2, /* IX,IX */
/* ADD */ 2, /* IX,SP */
/* ADD */ 2, /* IY,BC */
/* ADD */ 2, /* IY,DE */
/* ADD */ 2, /* IY,IY */
/* ADD */ 2, /* IY,SP */
/* AND */ 3, /* (HL) */
/* AND */ 3, /* (IX+b) */
/* AND */ 3, /* (IY+b) */
/* AND */ 3, /* A */
/* AND */ 3, /* B */
/* AND */ 3, /* C */
/* AND */ 3, /* D */
/* AND */ 3, /* E */
/* AND */ 3, /* H */
/* AND */ 3, /* L */
/* AND */ 3, /* b */
/* BIT */ 4, /* 0,(HL) */
/* BIT */ 4, /* 0,(IX+b) */
/* BIT */ 4, /* 0,(IY+b) */
/* BIT */ 4, /* 0,A */
/* BIT */ 4, /* 0,B */
/* BIT */ 4, /* 0,C */
/* BIT */ 4, /* 0,D */
/* BIT */ 4, /* 0,E */
/* BIT */ 4, /* 0,H */
/* BIT */ 4, /* 0,L */
/* BIT */ 4, /* 1,(HL) */
/* BIT */ 4, /* 1,(IX+b) */
/* BIT */ 4, /* 1,(IY+b) */
/* BIT */ 4, /* 1,A */
/* BIT */ 4, /* 1,B */
/* BIT */ 4, /* 1,C */
/* BIT */ 4, /* 1,D */
/* BIT */ 4, /* 1,E */
/* BIT */ 4, /* 1,H */
/* BIT */ 4, /* 1,L */
/* BIT */ 4, /* 2,(HL) */
/* BIT */ 4, /* 2,(IX+b) */
/* BIT */ 4, /* 2,(IY+b) */
/* BIT */ 4, /* 2,A */
/* BIT */ 4, /* 2,B */
/* BIT */ 4, /* 2,C */
/* BIT */ 4, /* 2,D */
/* BIT */ 4, /* 2,E */
/* BIT */ 4, /* 2,H */
/* BIT */ 4, /* 2,L */
/* BIT */ 4, /* 3,(HL) */
/* BIT */ 4, /* 3,(IX+b) */
/* BIT */ 4, /* 3,(IY+b) */
/* BIT */ 4, /* 3,A */
/* BIT */ 4, /* 3,B */
/* BIT */ 4, /* 3,C */
/* BIT */ 4, /* 3,D */
/* BIT */ 4, /* 3,E */
/* BIT */ 4, /* 3,H */
/* BIT */ 4, /* 3,L */
/* BIT */ 4, /* 4,(HL) */

```



มหาวิทยาลัยราชภัฏวชิรเวศน์

วิทยาลัยพยาบาล

```

/* BIT */ 4, /* 4,(IX+b) */
/* BIT */ 4, /* 4,(IY+b) */
/* BIT */ 4, /* 4,A */
/* BIT */ 4, /* 4,B */
/* BIT */ 4, /* 4,C */
/* BIT */ 4, /* 4,D */
/* BIT */ 4, /* 4,E */
/* BIT */ 4, /* 4,H */
/* BIT */ 4, /* 4,L */
/* BIT */ 4, /* 5,(HL) */
/* BIT */ 4, /* 5,(IX+b) */
/* BIT */ 4, /* 5,(IY+b) */
/* BIT */ 4, /* 5,A */
/* BIT */ 4, /* 5,B */
/* BIT */ 4, /* 5,C */
/* BIT */ 4, /* 5,D */
/* BIT */ 4, /* 5,E */
/* BIT */ 4, /* 5,H */
/* BIT */ 4, /* 5,L */
/* BIT */ 4, /* 6,(HL) */
/* BIT */ 4, /* 6,(IX+b) */
/* BIT */ 4, /* 6,(IY+b) */
/* BIT */ 4, /* 6,A */
/* BIT */ 4, /* 6,B */
/* BIT */ 4, /* 6,C */
/* BIT */ 4, /* 6,D */
/* BIT */ 4, /* 6,E */
/* BIT */ 4, /* 6,H */
/* BIT */ 4, /* 6,L */
/* BIT */ 4, /* 7,(HL) */
/* BIT */ 4, /* 7,(IX+b) */
/* BIT */ 4, /* 7,(IY+b) */
/* BIT */ 4, /* 7,A */
/* BIT */ 4, /* 7,B */
/* BIT */ 4, /* 7,C */
/* BIT */ 4, /* 7,D */
/* BIT */ 4, /* 7,E */
/* BIT */ 4, /* 7,H */
/* BIT */ 4, /* 7,L */
/* CALL */ 5, /* C,a */
/* CALL */ 5, /* M,a */
/* CALL */ 5, /* NC,a */
/* CALL */ 5, /* NZ,a */
/* CALL */ 5, /* P,a */
/* CALL */ 5, /* PE,a */
/* CALL */ 5, /* PO,a */
/* CALL */ 5, /* Z,a */
/* CALL */ 5, /* a */
/* CCF */ 6, /* */
/* CP */ 7, /* (HL) */
/* CP */ 7, /* (IX+b) */
/* CP */ 7, /* (IY+b) */
/* CP */ 7, /* A */
/* CP */ 7, /* B */
/* CP */ 7, /* C */
/* CP */ 7, /* D */
/* CP */ 7, /* E */
/* CP */ 7, /* H */
/* CP */ 7, /* L */
/* CP */ 7, /* b */
/* CPD */ 8, /* */
/* CPDR */ 9, /* */
/* CPI */ 10, /* */
/* CPIR */ 11, /* */
/* CPL */ 12, /* */
/* DAA */ 13, /* */
/* DEC */ 14, /* (HL) */
/* DEC */ 14, /* (IX+b) */
/* DEC */ 14, /* (IY+b) */
/* DEC */ 14, /* A */
/* DEC */ 14, /* B */
/* DEC */ 14, /* BC */
/* DEC */ 14, /* C */
/* DEC */ 14, /* D */
/* DEC */ 14, /* DE */
/* DEC */ 14, /* E */
/* DEC */ 14, /* H */
/* DEC */ 14, /* HL */
/* DEC */ 14, /* IX */
/* DEC */ 14, /* IY */
/* DEC */ 14, /* L */
/* DEC */ 14, /* SP */
/* DI */ 15, /* */
/* DJNZ */ 16, /* r */
/* EI */ 17, /* */
/* EX */ 18, /* (SP),HL */
/* EX */ 18, /* (SP),IX */
/* EX */ 18, /* (SP),IY */
/* EX */ 18, /* AF,AF' */
/* EX */ 18, /* DE,HL */

```



ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

/* EXX */19, /* */
/* HALT */20, /* */
/* IM */21, /* 0 */
/* IM */21, /* 1 */
/* IM */21, /* 2 */
/* IN */22, /* A,(C) */
/* IN */22, /* A,(b) */
/* IN */22, /* B,(C) */
/* IN */22, /* C,(C) */
/* IN */22, /* D,(C) */
/* IN */22, /* E,(C) */
/* IN */22, /* H,(C) */
/* IN */22, /* L,(C) */
/* INC */23, /* (HL) */
/* INC */23, /* (IX+b) */
/* INC */23, /* (IY+b) */
/* INC */23, /* A */
/* INC */23, /* B */
/* INC */23, /* BC */
/* INC */23, /* C */
/* INC */23, /* D */
/* INC */23, /* DE */
/* INC */23, /* E */
/* INC */23, /* H */
/* INC */23, /* HL */
/* INC */23, /* IX */
/* INC */23, /* IY */
/* INC */23, /* L */
/* INC */23, /* SP */
/* IND */24, /* */
/* INDR */25, /* */
/* INI */26, /* */
/* INIR */27, /* */
/* JP */28, /* (HL) */
/* JP */28, /* (IX) */
/* JP */28, /* (IY) */
/* JP */28, /* C,a */
/* JP */28, /* M,a */
/* JP */28, /* NC,a */
/* JP */28, /* NZ,a */
/* JP */28, /* P,a */
/* JP */28, /* PE,a */
/* JP */28, /* PO,a */
/* JP */28, /* Z,a */
/* JP */28, /* a */
/* JR */29, /* C,r */
/* JR */29, /* NC,r */
/* JR */29, /* NZ,r */
/* JR */29, /* Z,r */
/* JR */29, /* r */
/* LD */30, /* (BC),A */
/* LD */30, /* (DE),A */
/* LD */30, /* (HL),A */
/* LD */30, /* (HL),B */
/* LD */30, /* (HL),C */
/* LD */30, /* (HL),D */
/* LD */30, /* (HL),E */
/* LD */30, /* (HL),H */
/* LD */30, /* (HL),L */
/* LD */30, /* (HL),b */
/* LD */30, /* (IX+b),A */
/* LD */30, /* (IX+b),B */
/* LD */30, /* (IX+b),C */
/* LD */30, /* (IX+b),D */
/* LD */30, /* (IX+b),E */
/* LD */30, /* (IX+b),H */
/* LD */30, /* (IX+b),L */
/* LD */30, /* (IX+b),b */
/* LD */30, /* (IY+b),A */
/* LD */30, /* (IY+b),B */
/* LD */30, /* (IY+b),C */
/* LD */30, /* (IY+b),D */
/* LD */30, /* (IY+b),E */
/* LD */30, /* (IY+b),H */
/* LD */30, /* (IY+b),L */
/* LD */30, /* (IY+b),b */
/* LD */30, /* (a),A */
/* LD */30, /* (a),BC */
/* LD */30, /* (a),DE */
/* LD */30, /* (a),HL */
/* LD */30, /* (a),IX */
/* LD */30, /* (a),IY */
/* LD */30, /* (a),SP */
/* LD */30, /* A,(BC) */
/* LD */30, /* A,(DE) */
/* LD */30, /* A,(HL) */
/* LD */30, /* A,(IX+b) */
/* LD */30, /* A,(IY+b) */
/* LD */30, /* A,(a) */
/* LD */30, /* A,A */

```



```

/* LD */30, /* A,B */
/* LD */30, /* A,C */
/* LD */30, /* A,D */
/* LD */30, /* A,E */
/* LD */30, /* A,H */
/* LD */30, /* A,I */
/* LD */30, /* A,L */
/* LD */30, /* A,R */
/* LD */30, /* A,b */
/* LD */30, /* B,(HL) */
/* LD */30, /* B,(IX+b) */
/* LD */30, /* B,(IY+b) */
/* LD */30, /* B,A */
/* LD */30, /* B,B */
/* LD */30, /* B,C */
/* LD */30, /* B,D */
/* LD */30, /* B,E */
/* LD */30, /* B,H */
/* LD */30, /* B,L */
/* LD */30, /* B,b */
/* LD */30, /* BC,(a) */
/* LD */30, /* BC,a */
/* LD */30, /* C,(HL) */
/* LD */30, /* C,(IX+b) */
/* LD */30, /* C,(IY+b) */
/* LD */30, /* C,A */
/* LD */30, /* C,B */
/* LD */30, /* C,C */
/* LD */30, /* C,D */
/* LD */30, /* C,E */
/* LD */30, /* C,H */
/* LD */30, /* C,L */
/* LD */30, /* C,b */
/* LD */30, /* D,(HL) */
/* LD */30, /* D,(IX+b) */
/* LD */30, /* D,(IY+b) */
/* LD */30, /* D,A */
/* LD */30, /* D,B */
/* LD */30, /* D,C */
/* LD */30, /* D,D */
/* LD */30, /* D,E */
/* LD */30, /* D,H */
/* LD */30, /* D,L */
/* LD */30, /* D,b */
/* LD */30, /* DE,(a) */
/* LD */30, /* DE,a */
/* LD */30, /* E,(HL) */
/* LD */30, /* E,(IX+b) */
/* LD */30, /* E,(IY+b) */
/* LD */30, /* E,A */
/* LD */30, /* E,B */
/* LD */30, /* E,C */
/* LD */30, /* E,D */
/* LD */30, /* E,E */
/* LD */30, /* E,H */
/* LD */30, /* E,L */
/* LD */30, /* E,b */
/* LD */30, /* H,(HL) */
/* LD */30, /* H,(IX+b) */
/* LD */30, /* H,(IY+b) */
/* LD */30, /* H,A */
/* LD */30, /* H,B */
/* LD */30, /* H,C */
/* LD */30, /* H,D */
/* LD */30, /* H,E */
/* LD */30, /* H,H */
/* LD */30, /* H,L */
/* LD */30, /* H,b */
/* LD */30, /* HL,(a) */
/* LD */30, /* HL,a */
/* LD */30, /* I,A */
/* LD */30, /* IX,(a) */
/* LD */30, /* IX,a */
/* LD */30, /* IY,(a) */
/* LD */30, /* IY,a */
/* LD */30, /* L,(HL) */
/* LD */30, /* L,(IX+b) */
/* LD */30, /* L,(IY+b) */
/* LD */30, /* L,A */
/* LD */30, /* L,B */
/* LD */30, /* L,C */
/* LD */30, /* L,D */
/* LD */30, /* L,E */
/* LD */30, /* L,H */
/* LD */30, /* L,L */
/* LD */30, /* L,b */
/* LD */30, /* R,A */
/* LD */30, /* SP,(a) */
/* LD */30, /* SP,HL */
/* LD */30, /* SP,IX */

```



มหาวิทยาลัยราชภัฏวชิรวิทยาดงใต้
 วิทยาลัยพยาบาล
 วิทยาลัยพยาบาล
 วิทยาลัยพยาบาล

```

/* LD */30, /* SP,IY */
/* LD */30, /* SP,a */
/* LDD */31, /* */
/* LDDR */32, /* */
/* LDI */33, /* */
/* LDIR */34, /* */
/* NEG */35, /* */
/* NOP */36, /* */
/* OR */37, /* (HL) */
/* OR */37, /* (IX+b) */
/* OR */37, /* (IY+b) */
/* OR */37, /* A */
/* OR */37, /* B */
/* OR */37, /* C */
/* OR */37, /* D */
/* OR */37, /* E */
/* OR */37, /* H */
/* OR */37, /* L */
/* OR */37, /* b */
/* OTDR */38, /* */
/* OTIR */39, /* */
/* OUT */40, /* (C),A */
/* OUT */40, /* (C),B */
/* OUT */40, /* (C),C */
/* OUT */40, /* (C),D */
/* OUT */40, /* (C),E */
/* OUT */40, /* (C),H */
/* OUT */40, /* (C),L */
/* OUT */40, /* (b),A */
/* OUTD */41, /* */
/* OUTI */42, /* */
/* POP */43, /* AF */
/* POP */43, /* BC */
/* POP */43, /* DE */
/* POP */43, /* HL */
/* POP */43, /* IX */
/* POP */43, /* IY */
/* PUSH */44, /* AF */
/* PUSH */44, /* BC */
/* PUSH */44, /* DE */
/* PUSH */44, /* HL */
/* PUSH */44, /* IX */
/* PUSH */44, /* IY */
/* RES */45, /* 0,(HL) */
/* RES */45, /* 0,(IX+b) */
/* RES */45, /* 0,(IY+b) */
/* RES */45, /* 0,A */
/* RES */45, /* 0,B */
/* RES */45, /* 0,C */
/* RES */45, /* 0,D */
/* RES */45, /* 0,E */
/* RES */45, /* 0,H */
/* RES */45, /* 0,L */
/* RES */45, /* 1,(HL) */
/* RES */45, /* 1,(IX+b) */
/* RES */45, /* 1,(IY+b) */
/* RES */45, /* 1,A */
/* RES */45, /* 1,B */
/* RES */45, /* 1,C */
/* RES */45, /* 1,D */
/* RES */45, /* 1,E */
/* RES */45, /* 1,H */
/* RES */45, /* 1,L */
/* RES */45, /* 2,(HL) */
/* RES */45, /* 2,(IX+b) */
/* RES */45, /* 2,(IY+b) */
/* RES */45, /* 2,A */
/* RES */45, /* 2,B */
/* RES */45, /* 2,C */
/* RES */45, /* 2,D */
/* RES */45, /* 2,E */
/* RES */45, /* 2,H */
/* RES */45, /* 2,L */
/* RES */45, /* 3,(HL) */
/* RES */45, /* 3,(IX+b) */
/* RES */45, /* 3,(IY+b) */
/* RES */45, /* 3,A */
/* RES */45, /* 3,B */
/* RES */45, /* 3,C */
/* RES */45, /* 3,D */
/* RES */45, /* 3,E */
/* RES */45, /* 3,H */
/* RES */45, /* 3,L */
/* RES */45, /* 4,(HL) */
/* RES */45, /* 4,(IX+b) */
/* RES */45, /* 4,(IY+b) */
/* RES */45, /* 4,A */
/* RES */45, /* 4,B */
/* RES */45, /* 4,C */
/* RES */45, /* 4,D */

```



มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี
 วิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี


```

/* RES */45, /* 4,E */
/* RES */45, /* 4,H */
/* RES */45, /* 4,L */
/* RES */45, /* 5,(HL) */
/* RES */45, /* 5,(IX+b) */
/* RES */45, /* 5,(IY+b) */
/* RES */45, /* 5,A */
/* RES */45, /* 5,B */
/* RES */45, /* 5,C */
/* RES */45, /* 5,D */
/* RES */45, /* 5,E */
/* RES */45, /* 5,H */
/* RES */45, /* 5,L */
/* RES */45, /* 6,(HL) */
/* RES */45, /* 6,(IX+b) */
/* RES */45, /* 6,(IY+b) */
/* RES */45, /* 6,A */
/* RES */45, /* 6,B */
/* RES */45, /* 6,C */
/* RES */45, /* 6,D */
/* RES */45, /* 6,E */
/* RES */45, /* 6,H */
/* RES */45, /* 6,L */
/* RES */45, /* 7,(HL) */
/* RES */45, /* 7,(IX+b) */
/* RES */45, /* 7,(IY+b) */
/* RES */45, /* 7,A */
/* RES */45, /* 7,B */
/* RES */45, /* 7,C */
/* RES */45, /* 7,D */
/* RES */45, /* 7,E */
/* RES */45, /* 7,H */
/* RES */45, /* 7,L */
/* RET */46, /* */
/* RET */46, /* C */
/* RET */46, /* M */
/* RET */46, /* NC */
/* RET */46, /* NZ */
/* RET */46, /* P */
/* RET */46, /* PE */
/* RET */46, /* PO */
/* RET */46, /* Z */
/* RETI */47, /* */
/* RETN */48, /* */
/* RL */49, /* (HL) */
/* RL */49, /* (IX+b) */
/* RL */49, /* (IY+b) */
/* RL */49, /* A */
/* RL */49, /* B */
/* RL */49, /* C */
/* RL */49, /* D */
/* RL */49, /* E */
/* RL */49, /* H */
/* RL */49, /* L */
/* RLA */50, /* */
/* RLC */51, /* (HL) */
/* RLC */51, /* (IX+b) */
/* RLC */51, /* (IY+b) */
/* RLC */51, /* A */
/* RLC */51, /* B */
/* RLC */51, /* C */
/* RLC */51, /* D */
/* RLC */51, /* E */
/* RLC */51, /* H */
/* RLC */51, /* L */
/* RLCA */52, /* */
/* RLD */53, /* */
/* RR */54, /* (HL) */
/* RR */54, /* (IX+b) */
/* RR */54, /* (IY+b) */
/* RR */54, /* A */
/* RR */54, /* B */
/* RR */54, /* C */
/* RR */54, /* D */
/* RR */54, /* E */
/* RR */54, /* H */
/* RR */54, /* L */
/* RRA */55, /* */
/* RRC */56, /* (HL) */
/* RRC */56, /* (IX+b) */
/* RRC */56, /* (IY+b) */
/* RRC */56, /* A */
/* RRC */56, /* B */
/* RRC */56, /* C */
/* RRC */56, /* D */
/* RRC */56, /* E */
/* RRC */56, /* H */
/* RRC */56, /* L */
/* RRCA */57, /* */
/* RRD */58, /* */

```



ศูนย์วิทยทรัพยากร
 ภาควิชาคณิศร มหาวิททยาลัย

```

/* RST */59, /* 00 */
/* RST */59, /* 0 */
/* RST */59, /* 08 */
/* RST */59, /* 8 */
/* RST */59, /* 10 */
/* RST */59, /* 18 */
/* RST */59, /* 20 */
/* RST */59, /* 28 */
/* RST */59, /* 30 */
/* RST */59, /* 38 */
/* SBC */60, /* A,(HL) */
/* SBC */60, /* A,(IX+b) */
/* SBC */60, /* A,(IY+b) */
/* SBC */60, /* A,A */
/* SBC */60, /* A,B */
/* SBC */60, /* A,C */
/* SBC */60, /* A,D */
/* SBC */60, /* A,E */
/* SBC */60, /* A,H */
/* SBC */60, /* A,L */
/* SBC */60, /* A,b */
/* SBC */60, /* HL,BC */
/* SBC */60, /* HL,DE */
/* SBC */60, /* HL,HL */
/* SBC */60, /* HL,SP */
/* SCF */61, /* */
/* SET */62, /* 0,(HL) */
/* SET */62, /* 0,(IX+b) */
/* SET */62, /* 0,(IY+b) */
/* SET */62, /* 0,A */
/* SET */62, /* 0,B */
/* SET */62, /* 0,C */
/* SET */62, /* 0,D */
/* SET */62, /* 0,E */
/* SET */62, /* 0,H */
/* SET */62, /* 0,L */
/* SET */62, /* 1,(HL) */
/* SET */62, /* 1,(IX+b) */
/* SET */62, /* 1,(IY+b) */
/* SET */62, /* 1,A */
/* SET */62, /* 1,B */
/* SET */62, /* 1,C */
/* SET */62, /* 1,D */
/* SET */62, /* 1,E */
/* SET */62, /* 1,H */
/* SET */62, /* 1,L */
/* SET */62, /* 2,(HL) */
/* SET */62, /* 2,(IX+b) */
/* SET */62, /* 2,(IY+b) */
/* SET */62, /* 2,A */
/* SET */62, /* 2,B */
/* SET */62, /* 2,C */
/* SET */62, /* 2,D */
/* SET */62, /* 2,E */
/* SET */62, /* 2,H */
/* SET */62, /* 2,L */
/* SET */62, /* 3,(HL) */
/* SET */62, /* 3,(IX+b) */
/* SET */62, /* 3,(IY+b) */
/* SET */62, /* 3,A */
/* SET */62, /* 3,B */
/* SET */62, /* 3,C */
/* SET */62, /* 3,D */
/* SET */62, /* 3,E */
/* SET */62, /* 3,H */
/* SET */62, /* 3,L */
/* SET */62, /* 4,(HL) */
/* SET */62, /* 4,(IX+b) */
/* SET */62, /* 4,(IY+b) */
/* SET */62, /* 4,A */
/* SET */62, /* 4,B */
/* SET */62, /* 4,C */
/* SET */62, /* 4,D */
/* SET */62, /* 4,E */
/* SET */62, /* 4,H */
/* SET */62, /* 4,L */
/* SET */62, /* 5,(HL) */
/* SET */62, /* 5,(IX+b) */
/* SET */62, /* 5,(IY+b) */
/* SET */62, /* 5,A */
/* SET */62, /* 5,B */
/* SET */62, /* 5,C */
/* SET */62, /* 5,D */
/* SET */62, /* 5,E */
/* SET */62, /* 5,H */
/* SET */62, /* 5,L */
/* SET */62, /* 6,(HL) */
/* SET */62, /* 6,(IX+b) */
/* SET */62, /* 6,(IY+b) */
/* SET */62, /* 6,A */

```



มหาวิทยาลัยราชภัฏวชิรวิทยาดงรัก
 วิทยาลัยพยาบาลราชภัฏวชิรวิทยาดงรัก

```

/* SET */62, /* 6,B */
/* SET */62, /* 6,C */
/* SET */62, /* 6,D */
/* SET */62, /* 6,E */
/* SET */62, /* 6,H */
/* SET */62, /* 6,L */
/* SET */62, /* 7,(HL) */
/* SET */62, /* 7,(IX+b) */
/* SET */62, /* 7,(IY+b) */
/* SET */62, /* 7,A */
/* SET */62, /* 7,B */
/* SET */62, /* 7,C */
/* SET */62, /* 7,D */
/* SET */62, /* 7,E */
/* SET */62, /* 7,H */
/* SET */62, /* 7,L */
/* SLA */63, /* (HL) */
/* SLA */63, /* (IX+b) */
/* SLA */63, /* (IY+b) */
/* SLA */63, /* A */
/* SLA */63, /* B */
/* SLA */63, /* C */
/* SLA */63, /* D */
/* SLA */63, /* E */
/* SLA */63, /* H */
/* SLA */63, /* L */
/* SRA */64, /* (HL) */
/* SRA */64, /* (IX+b) */
/* SRA */64, /* (IY+b) */
/* SRA */64, /* A */
/* SRA */64, /* B */
/* SRA */64, /* C */
/* SRA */64, /* D */
/* SRA */64, /* E */
/* SRA */64, /* H */
/* SRA */64, /* L */
/* SRL */65, /* (HL) */
/* SRL */65, /* (IX+b) */
/* SRL */65, /* (IY+b) */
/* SRL */65, /* A */
/* SRL */65, /* B */
/* SRL */65, /* C */
/* SRL */65, /* D */
/* SRL */65, /* E */
/* SRL */65, /* H */
/* SRL */65, /* L */
/* SUB */66, /* (HL) */
/* SUB */66, /* (IX+b) */
/* SUB */66, /* (IY+b) */
/* SUB */66, /* A */
/* SUB */66, /* B */
/* SUB */66, /* C */
/* SUB */66, /* D */
/* SUB */66, /* E */
/* SUB */66, /* H */
/* SUB */66, /* L */
/* SUB */66, /* b */
/* XOR */67, /* (HL) */
/* XOR */67, /* (IX+b) */
/* XOR */67, /* (IY+b) */
/* XOR */67, /* A */
/* XOR */67, /* B */
/* XOR */67, /* C */
/* XOR */67, /* D */
/* XOR */67, /* E */
/* XOR */67, /* H */
/* XOR */67, /* L */
/* XOR */67, /* b */

```



ศูนย์วิทยทรัพยากร
 วิทยาลัย

```

/* ADC A, */33, /* (HL) */18,
/* ADC A, */33, /* (IX+b) */31,
/* ADC A, */33, /* (IY+b) */32,
/* ADC A, */33, /* A */33,
/* ADC A, */33, /* B */34,
/* ADC A, */33, /* C */21,
/* ADC A, */33, /* D */35,
/* ADC A, */33, /* E */36,
/* ADC A, */33, /* H */37,
/* ADC A, */33, /* L */38,
/* ADC A, */33, /* b */41,
/* ADC HL, */52, /* BC */50,
/* ADC HL, */52, /* DE */51,
/* ADC HL, */52, /* HL */52,
/* ADC HL, */52, /* SP */53,
/* ADD A, */33, /* (HL) */18,
/* ADD A, */33, /* (IX+b) */31,
/* ADD A, */33, /* (IY+b) */32,
/* ADD A, */33, /* A */33,
/* ADD A, */33, /* B */34,
/* ADD A, */33, /* C */21,
/* ADD A, */33, /* D */35,
/* ADD A, */33, /* E */36,
/* ADD A, */33, /* H */37,
/* ADD A, */33, /* L */38,
/* ADD A, */33, /* b */41,
/* ADD HL, */52, /* BC */50,
/* ADD HL, */52, /* DE */51,
/* ADD HL, */52, /* HL */52,
/* ADD HL, */52, /* SP */53,
/* ADD IX, */54, /* BC */50,
/* ADD IX, */54, /* DE */51,
/* ADD IX, */54, /* IX */54,
/* ADD IX, */54, /* SP */53,
/* ADD IY, */55, /* BC */50,
/* ADD IY, */55, /* DE */51,
/* ADD IY, */55, /* IY */55,
/* ADD IY, */55, /* SP */53,
/* AND (HL) */18,0,
/* AND (IX+b) */31,0,
/* AND (IY+b) */32,0,
/* AND A */33,0,
/* AND B */34,0,
/* AND C */21,0,
/* AND D */35,0,
/* AND E */36,0,
/* AND H */37,0,
/* AND L */38,0,
/* AND b */41,0,
/* BIT 0, */1, /* (HL) */18,
/* BIT 0, */1, /* (IX+b) */31,
/* BIT 0, */1, /* (IY+b) */32,
/* BIT 0, */1, /* A */33,
/* BIT 0, */1, /* B */34,
/* BIT 0, */1, /* C */21,
/* BIT 0, */1, /* D */35,
/* BIT 0, */1, /* E */36,
/* BIT 0, */1, /* H */37,
/* BIT 0, */1, /* L */38,
/* BIT 1, */2, /* (HL) */18,
/* BIT 1, */2, /* (IX+b) */31,
/* BIT 1, */2, /* (IY+b) */32,
/* BIT 1, */2, /* A */33,
/* BIT 1, */2, /* B */34,
/* BIT 1, */2, /* C */21,
/* BIT 1, */2, /* D */35,
/* BIT 1, */2, /* E */36,
/* BIT 1, */2, /* H */37,
/* BIT 1, */2, /* L */38,
/* BIT 2, */3, /* (HL) */18,
/* BIT 2, */3, /* (IX+b) */31,
/* BIT 2, */3, /* (IY+b) */32,
/* BIT 2, */3, /* A */33,
/* BIT 2, */3, /* B */34,
/* BIT 2, */3, /* C */21,
/* BIT 2, */3, /* D */35,
/* BIT 2, */3, /* E */36,
/* BIT 2, */3, /* H */37,
/* BIT 2, */3, /* L */38,
/* BIT 3, */4, /* (HL) */18,
/* BIT 3, */4, /* (IX+b) */31,
/* BIT 3, */4, /* (IY+b) */32,
/* BIT 3, */4, /* A */33,
/* BIT 3, */4, /* B */34,
/* BIT 3, */4, /* C */21,
/* BIT 3, */4, /* D */35,
/* BIT 3, */4, /* E */36,
/* BIT 3, */4, /* H */37,
/* BIT 3, */4, /* L */38,
/* BIT 4, */5, /* (HL) */18,

```



มหาวิทยาลัย
วิศวกรรมมหาวิทาลัย

```

/* BIT 4, *5, /* (IX+b) */31,
/* BIT 4, *5, /* (IY+b) */32,
/* BIT 4, *5, /* A */33,
/* BIT 4, *5, /* B */34,
/* BIT 4, *5, /* C */21,
/* BIT 4, *5, /* D */35,
/* BIT 4, *5, /* E */36,
/* BIT 4, *5, /* H */37,
/* BIT 4, *5, /* L */38,
/* BIT 5, *6, /* (HL) */18,
/* BIT 5, *6, /* (IX+b) */31,
/* BIT 5, *6, /* (IY+b) */32,
/* BIT 5, *6, /* A */33,
/* BIT 5, *6, /* B */34,
/* BIT 5, *6, /* C */21,
/* BIT 5, *6, /* D */35,
/* BIT 5, *6, /* E */36,
/* BIT 5, *6, /* H */37,
/* BIT 5, *6, /* L */38,
/* BIT 6, *7, /* (HL) */18,
/* BIT 6, *7, /* (IX+b) */31,
/* BIT 6, *7, /* (IY+b) */32,
/* BIT 6, *7, /* A */33,
/* BIT 6, *7, /* B */34,
/* BIT 6, *7, /* C */21,
/* BIT 6, *7, /* D */35,
/* BIT 6, *7, /* E */36,
/* BIT 6, *7, /* H */37,
/* BIT 6, *7, /* L */38,
/* BIT 7, *8, /* (HL) */18,
/* BIT 7, *8, /* (IX+b) */31,
/* BIT 7, *8, /* (IY+b) */32,
/* BIT 7, *8, /* A */33,
/* BIT 7, *8, /* B */34,
/* BIT 7, *8, /* C */21,
/* BIT 7, *8, /* D */35,
/* BIT 7, *8, /* E */36,
/* BIT 7, *8, /* H */37,
/* BIT 7, *8, /* L */38,
/* CALL C, */21, /* a */29,
/* CALL M, */22, /* a */29,
/* CALL NC, */23, /* a */29,
/* CALL NZ, */24, /* a */29,
/* CALL P, */25, /* a */29,
/* CALL PE, */26, /* a */29,
/* CALL PO, */27, /* a */29,
/* CALL Z, */28, /* a */29,
/* CALL a */29,0,
/* CCF */0,0,
/* CP (HL) */18,0,
/* CP (IX+b) */31,0,
/* CP (IY+b) */32,0,
/* CP A */33,0,
/* CP B */34,0,
/* CP C */21,0,
/* CP D */35,0,
/* CP E */36,0,
/* CP H */37,0,
/* CP L */38,0,
/* CP b */41,0,
/* CPD */0,0,
/* CPDR */0,0,
/* CPI */0,0,
/* CPIR */0,0,
/* CPL */0,0,
/* DAA */0,0,
/* DEC (HL) */18,0,
/* DEC (IX+b) */31,0,
/* DEC (IY+b) */32,0,
/* DEC A */33,0,
/* DEC B */34,0,
/* DEC BC */50,0,
/* DEC C */21,0,
/* DEC D */35,0,
/* DEC DE */51,0,
/* DEC E */36,0,
/* DEC H */37,0,
/* DEC HL */52,0,
/* DEC IX */54,0,
/* DEC IY */55,0,
/* DEC L */38,0,
/* DEC SP */53,0,
/* DI */0,0,
/* DJNZ r */30,0,
/* EI */0,0,
/* EX (SP), */47, /* HL */52,
/* EX (SP), */47, /* IX */54,
/* EX (SP), */47, /* IY */55,
/* EX AF, */49, /* AF' */48,
/* EX DE, */51, /* HL */52,

```

ศูนย์วิทยทรัพยากร
 วิทยาลัย

```

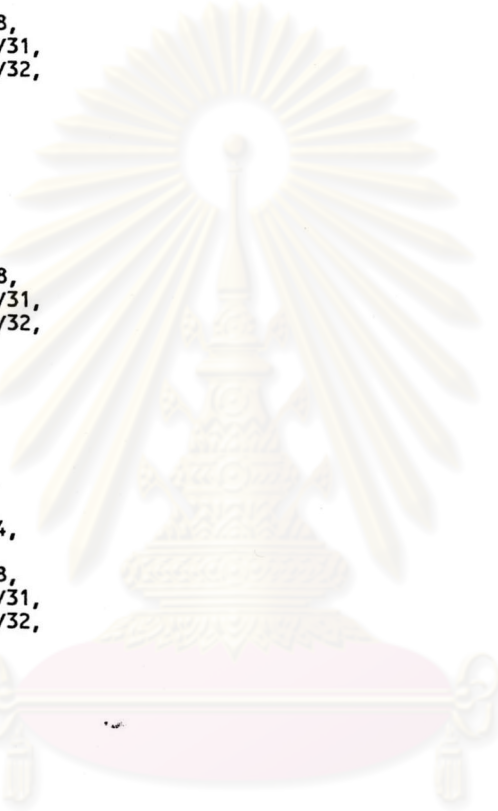
/* EXX */0,0,
/* HALT */0,0,
/* IM 0 */1,0,
/* IM 1 */2,0,
/* IM 2 */3,0,
/* IN A, */33, /* (C) */42,
/* IN A, */33, /* (b) */43,
/* IN B, */34, /* (C) */42,
/* IN C, */21, /* (C) */42,
/* IN D, */35, /* (C) */42,
/* IN E, */36, /* (C) */42,
/* IN H, */37, /* (C) */42,
/* IN L, */38, /* (C) */42,
/* INC (HL) */18,0,
/* INC (IX+b) */31,0,
/* INC (IY+b) */32,0,
/* INC A */33,0,
/* INC B */34,0,
/* INC BC */50,0,
/* INC C */21,0,
/* INC D */35,0,
/* INC DE */51,0,
/* INC E */36,0,
/* INC H */37,0,
/* INC HL */52,0,
/* INC IX */54,0,
/* INC IY */55,0,
/* INC L */38,0,
/* INC SP */53,0,
/* IND */0,0,
/* INDR */0,0,
/* INI */0,0,
/* INIR */0,0,
/* JP (HL) */18,0,
/* JP (IX) */19,0,
/* JP (IY) */20,0,
/* JP C, */21, /* a */29,
/* JP M, */22, /* a */29,
/* JP NC, */23, /* a */29,
/* JP NZ, */24, /* a */29,
/* JP P, */25, /* a */29,
/* JP PE, */26, /* a */29,
/* JP PO, */27, /* a */29,
/* JP Z, */28, /* a */29,
/* JP a */29,0,
/* JR C, */21, /* r */30,
/* JR NC, */23, /* r */30,
/* JR NZ, */24, /* r */30,
/* JR Z, */28, /* r */30,
/* JR r */30,0,
/* LD (BC), */45, /* A */33,
/* LD (DE), */46, /* A */33,
/* LD (HL), */18, /* A */33,
/* LD (HL), */18, /* B */34,
/* LD (HL), */18, /* C */21,
/* LD (HL), */18, /* D */35,
/* LD (HL), */18, /* E */36,
/* LD (HL), */18, /* H */37,
/* LD (HL), */18, /* L */38,
/* LD (HL), */18, /* b */41,
/* LD (IX+b), */31, /* A */33,
/* LD (IX+b), */31, /* B */34,
/* LD (IX+b), */31, /* C */21,
/* LD (IX+b), */31, /* D */35,
/* LD (IX+b), */31, /* E */36,
/* LD (IX+b), */31, /* H */37,
/* LD (IX+b), */31, /* L */38,
/* LD (IX+b), */31, /* b */41,
/* LD (IY+b), */32, /* A */33,
/* LD (IY+b), */32, /* B */34,
/* LD (IY+b), */32, /* C */21,
/* LD (IY+b), */32, /* D */35,
/* LD (IY+b), */32, /* E */36,
/* LD (IY+b), */32, /* H */37,
/* LD (IY+b), */32, /* L */38,
/* LD (IY+b), */32, /* b */41,
/* LD (a), */44, /* A */33,
/* LD (a), */44, /* BC */50,
/* LD (a), */44, /* DE */51,
/* LD (a), */44, /* HL */52,
/* LD (a), */44, /* IX */54,
/* LD (a), */44, /* IY */55,
/* LD (a), */44, /* SP */53,
/* LD A, */33, /* (BC) */45,
/* LD A, */33, /* (DE) */46,
/* LD A, */33, /* (HL) */18,
/* LD A, */33, /* (IX+b) */31,
/* LD A, */33, /* (IY+b) */32,
/* LD A, */33, /* (a) */44,
/* LD A, */33, /* A */33,

```



วิทยาลัยพยาบาล
กรมมหาวิทยาลัย

/* LD A, */33, /* B */34,
/* LD A, */33, /* C */21,
/* LD A, */33, /* D */35,
/* LD A, */33, /* E */36,
/* LD A, */33, /* H */37,
/* LD A, */33, /* I */39,
/* LD A, */33, /* L */38,
/* LD A, */33, /* R */40,
/* LD A, */33, /* b */41,
/* LD B, */34, /* (HL) */18,
/* LD B, */34, /* (IX+b) */31,
/* LD B, */34, /* (IY+b) */32,
/* LD B, */34, /* A */33,
/* LD B, */34, /* B */34,
/* LD B, */34, /* C */21,
/* LD B, */34, /* D */35,
/* LD B, */34, /* E */36,
/* LD B, */34, /* H */37,
/* LD B, */34, /* L */38,
/* LD B, */34, /* b */41,
/* LD BC, */50, /* (a) */44,
/* LD BC, */50, /* a */29,
/* LD C, */21, /* (HL) */18,
/* LD C, */21, /* (IX+b) */31,
/* LD C, */21, /* (IY+b) */32,
/* LD C, */21, /* A */33,
/* LD C, */21, /* B */34,
/* LD C, */21, /* C */21,
/* LD C, */21, /* D */35,
/* LD C, */21, /* E */36,
/* LD C, */21, /* H */37,
/* LD C, */21, /* L */38,
/* LD C, */21, /* b */41,
/* LD D, */35, /* (HL) */18,
/* LD D, */35, /* (IX+b) */31,
/* LD D, */35, /* (IY+b) */32,
/* LD D, */35, /* A */33,
/* LD D, */35, /* B */34,
/* LD D, */35, /* C */21,
/* LD D, */35, /* D */35,
/* LD D, */35, /* E */36,
/* LD D, */35, /* H */37,
/* LD D, */35, /* L */38,
/* LD D, */35, /* b */41,
/* LD DE, */51, /* (a) */44,
/* LD DE, */51, /* a */29,
/* LD E, */36, /* (HL) */18,
/* LD E, */36, /* (IX+b) */31,
/* LD E, */36, /* (IY+b) */32,
/* LD E, */36, /* A */33,
/* LD E, */36, /* B */34,
/* LD E, */36, /* C */21,
/* LD E, */36, /* D */35,
/* LD E, */36, /* E */36,
/* LD E, */36, /* H */37,
/* LD E, */36, /* L */38,
/* LD E, */36, /* b */41,
/* LD H, */37, /* (HL) */18,
/* LD H, */37, /* (IX+b) */31,
/* LD H, */37, /* (IY+b) */32,
/* LD H, */37, /* A */33,
/* LD H, */37, /* B */34,
/* LD H, */37, /* C */21,
/* LD H, */37, /* D */35,
/* LD H, */37, /* E */36,
/* LD H, */37, /* H */37,
/* LD H, */37, /* L */38,
/* LD H, */37, /* b */41,
/* LD HL, */52, /* (a) */44,
/* LD HL, */52, /* a */29,
/* LD I, */39, /* A */33,
/* LD IX, */54, /* (a) */44,
/* LD IX, */54, /* a */29,
/* LD IY, */55, /* (a) */44,
/* LD IY, */55, /* a */29,
/* LD L, */38, /* (HL) */18,
/* LD L, */38, /* (IX+b) */31,
/* LD L, */38, /* (IY+b) */32,
/* LD L, */38, /* A */33,
/* LD L, */38, /* B */34,
/* LD L, */38, /* C */21,
/* LD L, */38, /* D */35,
/* LD L, */38, /* E */36,
/* LD L, */38, /* H */37,
/* LD L, */38, /* L */38,
/* LD L, */38, /* b */41,
/* LD R, */40, /* A */33,
/* LD SP, */53, /* (a) */44,
/* LD SP, */53, /* HL */52,
/* LD SP, */53, /* IX */54,



วิทยาลัยพยาบาล
กรุงเทพมหานคร
มหาวิทยาลัย

```

/* LD SP, */53, /* IY */55,
/* LD SP, */53, /* a */29,
/* LDD */0,0,
/* LDDR */0,0,
/* LDI */0,0,
/* LDIR */0,0,
/* NEG */0,0,
/* NOP */0,0,
/* OR (HL) */18,0,
/* OR (IX+b) */31,0,
/* OR (IY+b) */32,0,
/* OR A */33,0,
/* OR B */34,0,
/* OR C */21,0,
/* OR D */35,0,
/* OR E */36,0,
/* OR H */37,0,
/* OR L */38,0,
/* OR b */41,0,
/* OTDR */0,0,
/* OTIR */0,0,
/* OUT (C), */42, /* A */33,
/* OUT (C), */42, /* B */34,
/* OUT (C), */42, /* C */21,
/* OUT (C), */42, /* D */35,
/* OUT (C), */42, /* E */36,
/* OUT (C), */42, /* H */37,
/* OUT (C), */42, /* L */38,
/* OUT (b), */43, /* A */33,
/* OUTD */0,0,
/* OUTI */0,0,
/* POP AF */49,0,
/* POP BC */50,0,
/* POP DE */51,0,
/* POP HL */52,0,
/* POP IX */54,0,
/* POP IY */55,0,
/* PUSH AF */49,0,
/* PUSH BC */50,0,
/* PUSH DE */51,0,
/* PUSH HL */52,0,
/* PUSH IX */54,0,
/* PUSH IY */55,0,
/* RES 0, */1, /* (HL) */18,
/* RES 0, */1, /* (IX+b) */31,
/* RES 0, */1, /* (IY+b) */32,
/* RES 0, */1, /* A */33,
/* RES 0, */1, /* B */34,
/* RES 0, */1, /* C */21,
/* RES 0, */1, /* D */35,
/* RES 0, */1, /* E */36,
/* RES 0, */1, /* H */37,
/* RES 0, */1, /* L */38,
/* RES 1, */2, /* (HL) */18,
/* RES 1, */2, /* (IX+b) */31,
/* RES 1, */2, /* (IY+b) */32,
/* RES 1, */2, /* A */33,
/* RES 1, */2, /* B */34,
/* RES 1, */2, /* C */21,
/* RES 1, */2, /* D */35,
/* RES 1, */2, /* E */36,
/* RES 1, */2, /* H */37,
/* RES 1, */2, /* L */38,
/* RES 2, */3, /* (HL) */18,
/* RES 2, */3, /* (IX+b) */31,
/* RES 2, */3, /* (IY+b) */32,
/* RES 2, */3, /* A */33,
/* RES 2, */3, /* B */34,
/* RES 2, */3, /* C */21,
/* RES 2, */3, /* D */35,
/* RES 2, */3, /* E */36,
/* RES 2, */3, /* H */37,
/* RES 2, */3, /* L */38,
/* RES 3, */4, /* (HL) */18,
/* RES 3, */4, /* (IX+b) */31,
/* RES 3, */4, /* (IY+b) */32,
/* RES 3, */4, /* A */33,
/* RES 3, */4, /* B */34,
/* RES 3, */4, /* C */21,
/* RES 3, */4, /* D */35,
/* RES 3, */4, /* E */36,
/* RES 3, */4, /* H */37,
/* RES 3, */4, /* L */38,
/* RES 4, */5, /* (HL) */18,
/* RES 4, */5, /* (IX+b) */31,
/* RES 4, */5, /* (IY+b) */32,
/* RES 4, */5, /* A */33,
/* RES 4, */5, /* B */34,
/* RES 4, */5, /* C */21,
/* RES 4, */5, /* D */35,

```



มหาวิทยาลัย
 วิทยาลัยพยาบาล
 วิทยาลัยพยาบาล


```

/* RES 4, */5, /* E */36,
/* RES 4, */5, /* H */37,
/* RES 4, */5, /* L */38,
/* RES 5, */6, /* (HL) */18,
/* RES 5, */6, /* (IX+b) */31,
/* RES 5, */6, /* (IY+b) */32,
/* RES 5, */6, /* A */33,
/* RES 5, */6, /* B */34,
/* RES 5, */6, /* C */21,
/* RES 5, */6, /* D */35,
/* RES 5, */6, /* E */36,
/* RES 5, */6, /* H */37,
/* RES 5, */6, /* L */38,
/* RES 6, */7, /* (HL) */18,
/* RES 6, */7, /* (IX+b) */31,
/* RES 6, */7, /* (IY+b) */32,
/* RES 6, */7, /* A */33,
/* RES 6, */7, /* B */34,
/* RES 6, */7, /* C */21,
/* RES 6, */7, /* D */35,
/* RES 6, */7, /* E */36,
/* RES 6, */7, /* H */37,
/* RES 6, */7, /* L */38,
/* RES 7, */8, /* (HL) */18,
/* RES 7, */8, /* (IX+b) */31,
/* RES 7, */8, /* (IY+b) */32,
/* RES 7, */8, /* A */33,
/* RES 7, */8, /* B */34,
/* RES 7, */8, /* C */21,
/* RES 7, */8, /* D */35,
/* RES 7, */8, /* E */36,
/* RES 7, */8, /* H */37,
/* RES 7, */8, /* L */38,
/* RET */0,0,
/* RET C */21,0,
/* RET M */22,0,
/* RET NC */23,0,
/* RET NZ */24,0,
/* RET P */25,0,
/* RET PE */26,0,
/* RET PO */27,0,
/* RET Z */28,0,
/* RETI */0,0,
/* RETN */0,0,
/* RL (HL) */18,0,
/* RL (IX+b) */31,0,
/* RL (IY+b) */32,0,
/* RL A */33,0,
/* RL B */34,0,
/* RL C */21,0,
/* RL D */35,0,
/* RL E */36,0,
/* RL H */37,0,
/* RL L */38,0,
/* RLA */0,0,
/* RLC (HL) */18,0,
/* RLC (IX+b) */31,0,
/* RLC (IY+b) */32,0,
/* RLC A */33,0,
/* RLC B */34,0,
/* RLC C */21,0,
/* RLC D */35,0,
/* RLC E */36,0,
/* RLC H */37,0,
/* RLC L */38,0,
/* RLCA */0,0,
/* RLD */0,0,
/* RR (HL) */18,0,
/* RR (IX+b) */31,0,
/* RR (IY+b) */32,0,
/* RR A */33,0,
/* RR B */34,0,
/* RR C */21,0,
/* RR D */35,0,
/* RR E */36,0,
/* RR H */37,0,
/* RR L */38,0,
/* RRA */0,0,
/* RRC (HL) */18,0,
/* RRC (IX+b) */31,0,
/* RRC (IY+b) */32,0,
/* RRC A */33,0,
/* RRC B */34,0,
/* RRC C */21,0,
/* RRC D */35,0,
/* RRC E */36,0,
/* RRC H */37,0,
/* RRC L */38,0,
/* RRCA */0,0,
/* RRD */0,0,

```



ศูนย์วิทยทรัพยากร

ฟาลงกรณ์มหาวิทยาลัย

```

/* RST 00 */10,0,
/* RST 0 */1,0,
/* RST 08 */11,0,
/* RST 8 */9,0,
/* RST 10 */12,0,
/* RST 18 */13,0,
/* RST 20 */14,0,
/* RST 28 */15,0,
/* RST 30 */16,0,
/* RST 38 */17,0,
/* SBC A, */33, /* (HL) */18,
/* SBC A, */33, /* (IX+b) */31,
/* SBC A, */33, /* (IY+b) */32,
/* SBC A, */33, /* A */33,
/* SBC A, */33, /* B */34,
/* SBC A, */33, /* C */21,
/* SBC A, */33, /* D */35,
/* SBC A, */33, /* E */36,
/* SBC A, */33, /* H */37,
/* SBC A, */33, /* L */38,
/* SBC A, */33, /* b */41,
/* SBC HL, */52, /* BC */50,
/* SBC HL, */52, /* DE */51,
/* SBC HL, */52, /* HL */52,
/* SBC HL, */52, /* SP */53,
/* SCF */0,0,
/* SET 0, */1, /* (HL) */18,
/* SET 0, */1, /* (IX+b) */31,
/* SET 0, */1, /* (IY+b) */32,
/* SET 0, */1, /* A */33,
/* SET 0, */1, /* B */34,
/* SET 0, */1, /* C */21,
/* SET 0, */1, /* D */35,
/* SET 0, */1, /* E */36,
/* SET 0, */1, /* H */37,
/* SET 0, */1, /* L */38,
/* SET 1, */2, /* (HL) */18,
/* SET 1, */2, /* (IX+b) */31,
/* SET 1, */2, /* (IY+b) */32,
/* SET 1, */2, /* A */33,
/* SET 1, */2, /* B */34,
/* SET 1, */2, /* C */21,
/* SET 1, */2, /* D */35,
/* SET 1, */2, /* E */36,
/* SET 1, */2, /* H */37,
/* SET 1, */2, /* L */38,
/* SET 2, */3, /* (HL) */18,
/* SET 2, */3, /* (IX+b) */31,
/* SET 2, */3, /* (IY+b) */32,
/* SET 2, */3, /* A */33,
/* SET 2, */3, /* B */34,
/* SET 2, */3, /* C */21,
/* SET 2, */3, /* D */35,
/* SET 2, */3, /* E */36,
/* SET 2, */3, /* H */37,
/* SET 2, */3, /* L */38,
/* SET 3, */4, /* (HL) */18,
/* SET 3, */4, /* (IX+b) */31,
/* SET 3, */4, /* (IY+b) */32,
/* SET 3, */4, /* A */33,
/* SET 3, */4, /* B */34,
/* SET 3, */4, /* C */21,
/* SET 3, */4, /* D */35,
/* SET 3, */4, /* E */36,
/* SET 3, */4, /* H */37,
/* SET 3, */4, /* L */38,
/* SET 4, */5, /* (HL) */18,
/* SET 4, */5, /* (IX+b) */31,
/* SET 4, */5, /* (IY+b) */32,
/* SET 4, */5, /* A */33,
/* SET 4, */5, /* B */34,
/* SET 4, */5, /* C */21,
/* SET 4, */5, /* D */35,
/* SET 4, */5, /* E */36,
/* SET 4, */5, /* H */37,
/* SET 4, */5, /* L */38,
/* SET 5, */6, /* (HL) */18,
/* SET 5, */6, /* (IX+b) */31,
/* SET 5, */6, /* (IY+b) */32,
/* SET 5, */6, /* A */33,
/* SET 5, */6, /* B */34,
/* SET 5, */6, /* C */21,
/* SET 5, */6, /* D */35,
/* SET 5, */6, /* E */36,
/* SET 5, */6, /* H */37,
/* SET 5, */6, /* L */38,
/* SET 6, */7, /* (HL) */18,
/* SET 6, */7, /* (IX+b) */31,
/* SET 6, */7, /* (IY+b) */32,
/* SET 6, */7, /* A */33,

```



```

/* SET 6, */7, /* B */34,
/* SET 6, */7, /* C */21,
/* SET 6, */7, /* D */35,
/* SET 6, */7, /* E */36,
/* SET 6, */7, /* H */37,
/* SET 6, */7, /* L */38,
/* SET 7, */8, /* (HL) */18,
/* SET 7, */8, /* (IX+b) */31,
/* SET 7, */8, /* (IY+b) */32,
/* SET 7, */8, /* A */33,
/* SET 7, */8, /* B */34,
/* SET 7, */8, /* C */21,
/* SET 7, */8, /* D */35,
/* SET 7, */8, /* E */36,
/* SET 7, */8, /* H */37,
/* SET 7, */8, /* L */38,
/* SLA (HL) */18,0,
/* SLA (IX+b) */31,0,
/* SLA (IY+b) */32,0,
/* SLA A */33,0,
/* SLA B */34,0,
/* SLA C */21,0,
/* SLA D */35,0,
/* SLA E */36,0,
/* SLA H */37,0,
/* SLA L */38,0,
/* SRA (HL) */18,0,
/* SRA (IX+b) */31,0,
/* SRA (IY+b) */32,0,
/* SRA A */33,0,
/* SRA B */34,0,
/* SRA C */21,0,
/* SRA D */35,0,
/* SRA E */36,0,
/* SRA H */37,0,
/* SRA L */38,0,
/* SRL (HL) */18,0,
/* SRL (IX+b) */31,0,
/* SRL (IY+b) */32,0,
/* SRL A */33,0,
/* SRL B */34,0,
/* SRL C */21,0,
/* SRL D */35,0,
/* SRL E */36,0,
/* SRL H */37,0,
/* SRL L */38,0,
/* SUB (HL) */18,0,
/* SUB (IX+b) */31,0,
/* SUB (IY+b) */32,0,
/* SUB A */33,0,
/* SUB B */34,0,
/* SUB C */21,0,
/* SUB D */35,0,
/* SUB E */36,0,
/* SUB H */37,0,
/* SUB L */38,0,
/* SUB b */41,0,
/* XOR (HL) */18,0,
/* XOR (IX+b) */31,0,
/* XOR (IY+b) */32,0,
/* XOR A */33,0,
/* XOR B */34,0,
/* XOR C */21,0,
/* XOR D */35,0,
/* XOR E */36,0,
/* XOR H */37,0,
/* XOR L */38,0,
/* XOR b */41,0

```



ศูนย์วิทยทรัพยากร
 ภาลงกรณ์มหาวิทยาลัย

```

/*08EH,*1,/* ADC A,(HL) */
/*0DDH,*3,/* ADC A,(IX+00) */
/*0FDH,*3,/* ADC A,(IY+00) */
/*08FH,*1,/* ADC A,A */
/*088H,*1,/* ADC A,B */
/*089H,*1,/* ADC A,C */
/*08AH,*1,/* ADC A,D */
/*08BH,*1,/* ADC A,E */
/*08CH,*1,/* ADC A,H */
/*08DH,*1,/* ADC A,L */
/*0CEH,*2,/* ADC A,00 */
/*0EDH,*2,/* ADC HL,BC */
/*0EDH,*2,/* ADC HL,DE */
/*0EDH,*2,/* ADC HL,HL */
/*0EDH,*2,/* ADC HL,SP */
/*086H,*1,/* ADD A,(HL) */
/*0DDH,*3,/* ADD A,(IX+00) */
/*0FDH,*3,/* ADD A,(IY+00) */
/*087H,*1,/* ADD A,A */
/*080H,*1,/* ADD A,B */
/*081H,*1,/* ADD A,C */
/*082H,*1,/* ADD A,D */
/*083H,*1,/* ADD A,E */
/*084H,*1,/* ADD A,H */
/*085H,*1,/* ADD A,L */
/*0C6H,*2,/* ADD A,00 */
/*009H,*1,/* ADD HL,BC */
/*019H,*1,/* ADD HL,DE */
/*029H,*1,/* ADD HL,HL */
/*039H,*1,/* ADD HL,SP */
/*0DDH,*2,/* ADD IX,BC */
/*0DDH,*2,/* ADD IX,DE */
/*0DDH,*2,/* ADD IX,IX */
/*0DDH,*2,/* ADD IX,SP */
/*0FDH,*2,/* ADD IY,BC */
/*0FDH,*2,/* ADD IY,DE */
/*0FDH,*2,/* ADD IY,IY */
/*0FDH,*2,/* ADD IY,SP */
/*0A6H,*1,/* AND (HL) */
/*0DDH,*3,/* AND (IX+00) */
/*0FDH,*3,/* AND (IY+00) */
/*0A7H,*1,/* AND A */
/*0A0H,*1,/* AND B */
/*0A1H,*1,/* AND C */
/*0A2H,*1,/* AND D */
/*0A3H,*1,/* AND E */
/*0A4H,*1,/* AND H */
/*0A5H,*1,/* AND L */
/*0E6H,*2,/* AND 00 */
/*0CBH,*2,/* BIT 0,(HL) */
/*0DDH,*4,/* BIT 0,(IX+00) */
/*0FDH,*4,/* BIT 0,(IY+00) */
/*0CBH,*2,/* BIT 0,A */
/*0CBH,*2,/* BIT 0,B */
/*0CBH,*2,/* BIT 0,C */
/*0CBH,*2,/* BIT 0,D */
/*0CBH,*2,/* BIT 0,E */
/*0CBH,*2,/* BIT 0,H */
/*0CBH,*2,/* BIT 0,L */
/*0CBH,*2,/* BIT 1,(HL) */
/*0DDH,*4,/* BIT 1,(IX+00) */
/*0FDH,*4,/* BIT 1,(IY+00) */
/*0CBH,*2,/* BIT 1,A */
/*0CBH,*2,/* BIT 1,B */
/*0CBH,*2,/* BIT 1,C */
/*0CBH,*2,/* BIT 1,D */
/*0CBH,*2,/* BIT 1,E */
/*0CBH,*2,/* BIT 1,H */
/*0CBH,*2,/* BIT 1,L */
/*0CBH,*2,/* BIT 2,(HL) */
/*0DDH,*4,/* BIT 2,(IX+00) */
/*0FDH,*4,/* BIT 2,(IY+00) */
/*0CBH,*2,/* BIT 2,A */
/*0CBH,*2,/* BIT 2,B */
/*0CBH,*2,/* BIT 2,C */
/*0CBH,*2,/* BIT 2,D */
/*0CBH,*2,/* BIT 2,E */
/*0CBH,*2,/* BIT 2,H */
/*0CBH,*2,/* BIT 2,L */
/*0CBH,*2,/* BIT 3,(HL) */
/*0DDH,*4,/* BIT 3,(IX+00) */
/*0FDH,*4,/* BIT 3,(IY+00) */
/*0CBH,*2,/* BIT 3,A */
/*0CBH,*2,/* BIT 3,B */
/*0CBH,*2,/* BIT 3,C */
/*0CBH,*2,/* BIT 3,D */
/*0CBH,*2,/* BIT 3,E */
/*0CBH,*2,/* BIT 3,H */
/*0CBH,*2,/* BIT 3,L */
/*0CBH,*2,/* BIT 4,(HL) */

```

```

/*ODDH,*1/4,* BIT 4,(IX+00) */
/*OFDH,*1/4,* BIT 4,(IY+00) */
/*OCBH,*1/2,* BIT 4,A */
/*OCBH,*1/2,* BIT 4,B */
/*OCBH,*1/2,* BIT 4,C */
/*OCBH,*1/2,* BIT 4,D */
/*OCBH,*1/2,* BIT 4,E */
/*OCBH,*1/2,* BIT 4,H */
/*OCBH,*1/2,* BIT 4,L */
/*OCBH,*1/2,* BIT 5,(HL) */
/*ODDH,*1/4,* BIT 5,(IX+00) */
/*OFDH,*1/4,* BIT 5,(IY+00) */
/*OCBH,*1/2,* BIT 5,A */
/*OCBH,*1/2,* BIT 5,B */
/*OCBH,*1/2,* BIT 5,C */
/*OCBH,*1/2,* BIT 5,D */
/*OCBH,*1/2,* BIT 5,E */
/*OCBH,*1/2,* BIT 5,H */
/*OCBH,*1/2,* BIT 5,L */
/*OCBH,*1/2,* BIT 6,(HL) */
/*ODDH,*1/4,* BIT 6,(IX+00) */
/*OFDH,*1/4,* BIT 6,(IY+00) */
/*OCBH,*1/2,* BIT 6,A */
/*OCBH,*1/2,* BIT 6,B */
/*OCBH,*1/2,* BIT 6,C */
/*OCBH,*1/2,* BIT 6,D */
/*OCBH,*1/2,* BIT 6,E */
/*OCBH,*1/2,* BIT 6,H */
/*OCBH,*1/2,* BIT 6,L */
/*OCBH,*1/2,* BIT 7,(HL) */
/*ODDH,*1/4,* BIT 7,(IX+00) */
/*OFDH,*1/4,* BIT 7,(IY+00) */
/*OCBH,*1/2,* BIT 7,A */
/*OCBH,*1/2,* BIT 7,B */
/*OCBH,*1/2,* BIT 7,C */
/*OCBH,*1/2,* BIT 7,D */
/*OCBH,*1/2,* BIT 7,E */
/*OCBH,*1/2,* BIT 7,H */
/*OCBH,*1/2,* BIT 7,L */
/*ODCH,*1/3,* CALL C,0000 */
/*OFCH,*1/3,* CALL M,0000 */
/*OD4H,*1/3,* CALL NC,0000 */
/*OC4H,*1/3,* CALL NZ,0000 */
/*OF4H,*1/3,* CALL P,0000 */
/*OECH,*1/3,* CALL PE,0000 */
/*OE4H,*1/3,* CALL PO,0000 */
/*OCCH,*1/3,* CALL Z,0000 */
/*OCDH,*1/3,* CALL 0000 */
/*O3FH,*1/1,* CCF */
/*OBEH,*1/1,* CP (HL) */
/*ODDH,*1/3,* CP (IX+00) */
/*OFDH,*1/3,* CP (IY+00) */
/*OBFH,*1/1,* CP A */
/*OB8H,*1/1,* CP B */
/*OB9H,*1/1,* CP C */
/*OBAH,*1/1,* CP D */
/*OBBH,*1/1,* CP E */
/*OBCH,*1/1,* CP H */
/*OBDH,*1/1,* CP L */
/*OFEH,*1/2,* CP 00 */
/*OEDH,*1/2,* CPD */
/*OEDH,*1/2,* CPDR */
/*OEDH,*1/2,* CPI */
/*OEDH,*1/2,* CPIR */
/*O2FH,*1/1,* CPL */
/*O27H,*1/1,* DAA */
/*O35H,*1/1,* DEC (HL) */
/*ODDH,*1/3,* DEC (IX+00) */
/*OFDH,*1/3,* DEC (IY+00) */
/*O3DH,*1/1,* DEC A */
/*O05H,*1/1,* DEC B */
/*O0BH,*1/1,* DEC BC */
/*O0DH,*1/1,* DEC C */
/*O15H,*1/1,* DEC D */
/*O1BH,*1/1,* DEC DE */
/*O1DH,*1/1,* DEC E */
/*O25H,*1/1,* DEC H */
/*O2BH,*1/1,* DEC HL */
/*ODDH,*1/2,* DEC IX */
/*OFDH,*1/2,* DEC IY */
/*O2DH,*1/1,* DEC L */
/*O3BH,*1/1,* DEC SP */
/*OF3H,*1/1,* DI */
/*O10H,*1/2,* DJNZ $ */
/*OFBH,*1/1,* EI */
/*OE3H,*1/1,* EX (SP),HL */
/*ODDH,*1/2,* EX (SP),IX */
/*OFDH,*1/2,* EX (SP),IY */
/*O08H,*1/1,* EX AF,AF' */
/*OEBH,*1/1,* EX DE,HL

```

```

/*0D9H,*/1,/* EXX */
/*076H,*/1,/* HALT */
/*OEDH,*/2,/* IM 0 */
/*OEDH,*/2,/* IM 1 */
/*OEDH,*/2,/* IM 2 */
/*OEDH,*/2,/* IN A,(C) */
/*ODBH,*/2,/* IN A,(00) */
/*OEDH,*/2,/* IN B,(C) */
/*OEDH,*/2,/* IN C,(C) */
/*OEDH,*/2,/* IN D,(C) */
/*OEDH,*/2,/* IN E,(C) */
/*OEDH,*/2,/* IN H,(C) */
/*OEDH,*/2,/* IN L,(C) */
/*034H,*/1,/* INC (HL) */
/*ODDH,*/3,/* INC (IX+00) */
/*OFDH,*/3,/* INC (IY+00) */
/*03CH,*/1,/* INC A */
/*004H,*/1,/* INC B */
/*003H,*/1,/* INC BC */
/*00CH,*/1,/* INC C */
/*014H,*/1,/* INC D */
/*013H,*/1,/* INC DE */
/*01CH,*/1,/* INC E */
/*024H,*/1,/* INC H */
/*023H,*/1,/* INC HL */
/*ODDH,*/2,/* INC IX */
/*OFDH,*/2,/* INC IY */
/*02CH,*/1,/* INC L */
/*033H,*/1,/* INC SP */
/*OEDH,*/2,/* IND */
/*OEDH,*/2,/* INDR */
/*OEDH,*/2,/* INI */
/*OEDH,*/2,/* INIR
(OE9H,*/1,/* JP (HL)
(ODDH,*/2,/* JP (IX)
(OFDH,*/2,/* JP (IY)
(ODAH,*/3,/* JP C,0000
(OFAH,*/3,/* JP M,0000
(OD2H,*/3,/* JP NC,0000
(OC2H,*/3,/* JP NZ,0000
(OF2H,*/3,/* JP P,0000
(OEAH,*/3,/* JP PE,0000
(OE2H,*/3,/* JP PO,0000
(OCAH,*/3,/* JP Z,0000
(OC3H,*/3,/* JP 0000
(O38H,*/2,/* JR C,$
(O30H,*/2,/* JR NC,$
(O20H,*/2,/* JR NZ,$
(O28H,*/2,/* JR Z,$
(O18H,*/2,/* JR $
(O02H,*/1,/* LD (BC),A
(O12H,*/1,/* LD (DE),A
(O77H,*/1,/* LD (HL),A
(O70H,*/1,/* LD (HL),B
(O71H,*/1,/* LD (HL),C
(O72H,*/1,/* LD (HL),D
(O73H,*/1,/* LD (HL),E
(O74H,*/1,/* LD (HL),H
(O75H,*/1,/* LD (HL),L
(O36H,*/2,/* LD (HL),00
(ODDH,*/3,/* LD (IX+00),A
(ODDH,*/3,/* LD (IX+00),B
(ODDH,*/3,/* LD (IX+00),C
(ODDH,*/3,/* LD (IX+00),D
(ODDH,*/3,/* LD (IX+00),E
(ODDH,*/3,/* LD (IX+00),H
(ODDH,*/3,/* LD (IX+00),L
(ODDH,*/4,/* LD (IX+00),00
(OFDH,*/3,/* LD (IY+00),A
(OFDH,*/3,/* LD (IY+00),B
(OFDH,*/3,/* LD (IY+00),C
(OFDH,*/3,/* LD (IY+00),D
(OFDH,*/3,/* LD (IY+00),E
(OFDH,*/3,/* LD (IY+00),H
(OFDH,*/3,/* LD (IY+00),L
(OFDH,*/4,/* LD (IY+00),00
(O32H,*/3,/* LD (0000),A
(OEDH,*/4,/* LD (0000),BC
(OEDH,*/4,/* LD (0000),DE
(O22H,*/3,/* LD (0000),HL
(ODDH,*/4,/* LD (0000),IX
(OFDH,*/4,/* LD (0000),IY
(OEDH,*/4,/* LD (0000),SP
(O0AH,*/1,/* LD A,(BC)
(O1AH,*/1,/* LD A,(DE)
(O7EH,*/1,/* LD A,(HL)
(ODDH,*/3,/* LD A,(IX+00)
(OFDH,*/3,/* LD A,(IY+00)
(O3AH,*/3,/* LD A,(0000)
(O7FH,*/1,/* LD A,A

```



วิทยาลัยพยาบาล
กรมมหาวิทยาลัย

```

/*078H,*1,/* LD A,B */
/*079H,*1,/* LD A,C */
/*07AH,*1,/* LD A,D */
/*07BH,*1,/* LD A,E */
/*07CH,*1,/* LD A,H */
/*0EDH,*2,/* LD A,I */
/*07DH,*1,/* LD A,L */
/*0EDH,*2,/* LD A,R */
/*03EH,*2,/* LD A,00 */
/*046H,*1,/* LD B,(HL) */
/*0DDH,*3,/* LD B,(IX+00) */
/*0FDH,*3,/* LD B,(IY+00) */
/*047H,*1,/* LD B,A */
/*040H,*1,/* LD B,B */
/*041H,*1,/* LD B,C */
/*042H,*1,/* LD B,D */
/*043H,*1,/* LD B,E */
/*044H,*1,/* LD B,H */
/*045H,*1,/* LD B,L */
/*006H,*2,/* LD B,00 */
/*0EDH,*4,/* LD BC,(0000) */
/*001H,*3,/* LD BC,0000 */
/*04EH,*1,/* LD C,(HL) */
/*0DDH,*3,/* LD C,(IX+00) */
/*0FDH,*3,/* LD C,(IY+00) */
/*04FH,*1,/* LD C,A */
/*048H,*1,/* LD C,B */
/*049H,*1,/* LD C,C */
/*04AH,*1,/* LD C,D */
/*04BH,*1,/* LD C,E */
/*04CH,*1,/* LD C,H */
/*04DH,*1,/* LD C,L */
/*00EH,*2,/* LD C,00 */
/*056H,*1,/* LD D,(HL) */
/*0DDH,*3,/* LD D,(IX+00) */
/*0FDH,*3,/* LD D,(IY+00) */
/*057H,*1,/* LD D,A */
/*050H,*1,/* LD D,B */
/*051H,*1,/* LD D,C */
/*052H,*1,/* LD D,D */
/*053H,*1,/* LD D,E */
/*054H,*1,/* LD D,H */
/*055H,*1,/* LD D,L */
/*016H,*2,/* LD D,00 */
/*0EDH,*4,/* LD DE,(0000) */
/*011H,*3,/* LD DE,0000 */
/*05EH,*1,/* LD E,(HL) */
/*0DDH,*3,/* LD E,(IX+00) */
/*0FDH,*3,/* LD E,(IY+00) */
/*05FH,*1,/* LD E,A */
/*058H,*1,/* LD E,B */
/*059H,*1,/* LD E,C */
/*05AH,*1,/* LD E,D */
/*05BH,*1,/* LD E,E */
/*05CH,*1,/* LD E,H */
/*05DH,*1,/* LD E,L */
/*01EH,*2,/* LD E,00 */
/*066H,*1,/* LD H,(HL) */
/*0DDH,*3,/* LD H,(IX+00) */
/*0FDH,*3,/* LD H,(IY+00) */
/*067H,*1,/* LD H,A */
/*060H,*1,/* LD H,B */
/*061H,*1,/* LD H,C */
/*062H,*1,/* LD H,D */
/*063H,*1,/* LD H,E */
/*064H,*1,/* LD H,H */
/*065H,*1,/* LD H,L */
/*026H,*2,/* LD H,00 */
/*02AH,*3,/* LD HL,(0000) */
/*021H,*3,/* LD HL,0000 */
/*0EDH,*2,/* LD I,A */
/*0DDH,*4,/* LD IX,(0000) */
/*0DDH,*4,/* LD IX,0000 */
/*0FDH,*4,/* LD IY,(0000) */
/*0FDH,*4,/* LD IY,0000 */
/*06EH,*1,/* LD L,(HL) */
/*0DDH,*3,/* LD L,(IX+00) */
/*0FDH,*3,/* LD L,(IY+00) */
/*06FH,*1,/* LD L,A */
/*068H,*1,/* LD L,B */
/*069H,*1,/* LD L,C */
/*06AH,*1,/* LD L,D */
/*06BH,*1,/* LD L,E */
/*06CH,*1,/* LD L,H */
/*06DH,*1,/* LD L,L */
/*02EH,*2,/* LD L,00 */
/*0EDH,*2,/* LD R,A */
/*0EDH,*4,/* LD SP,(0000) */
/*0F9H,*1,/* LD SP,HL */
/*0DDH,*2,/* LD SP,IX */

```

```

/*OFDH,*2,/* LD SP,IY */
/*O31H,*3,/* LD SP,0000 */
/*OEDH,*2,/* LDD */
/*OEDH,*2,/* LDDR */
/*OEDH,*2,/* LDI */
/*OEDH,*2,/* LDIR */
/*OEDH,*2,/* NEG */
/*O00H,*1,/* NOP */
/*OB6H,*1,/* OR (HL) */
/*ODDH,*3,/* OR (IX+00) */
/*OFDH,*3,/* OR (IY+00) */
/*OB7H,*1,/* OR A */
/*OB0H,*1,/* OR B */
/*OB1H,*1,/* OR C */
/*OB2H,*1,/* OR D */
/*OB3H,*1,/* OR E */
/*OB4H,*1,/* OR H */
/*OB5H,*1,/* OR L */
/*OF6H,*2,/* OR 00 */
/*OEDH,*2,/* OTDR */
/*OEDH,*2,/* OTIR */
/*OEDH,*2,/* OUT (C),A */
/*OEDH,*2,/* OUT (C),B */
/*OEDH,*2,/* OUT (C),C */
/*OEDH,*2,/* OUT (C),D */
/*OEDH,*2,/* OUT (C),E */
/*OEDH,*2,/* OUT (C),H */
/*OEDH,*2,/* OUT (C),L */
/*OD3H,*2,/* OUT (00),A */
/*OEDH,*2,/* OUTD */
/*OEDH,*2,/* OUTI */
/*OF1H,*1,/* POP AF */
/*OC1H,*1,/* POP BC */
/*OD1H,*1,/* POP DE */
/*OE1H,*1,/* POP HL */
/*ODDH,*2,/* POP IX */
/*OFDH,*2,/* POP IY */
/*OF5H,*1,/* PUSH AF */
/*OC5H,*1,/* PUSH BC */
/*OD5H,*1,/* PUSH DE */
/*OE5H,*1,/* PUSH HL */
/*ODDH,*2,/* PUSH IX */
/*OFDH,*2,/* PUSH IY */
/*OCBH,*2,/* RES 0,(HL) */
/*ODDH,*4,/* RES 0,(IX+00) */
/*OFDH,*4,/* RES 0,(IY+00) */
/*OCBH,*2,/* RES 0,A */
/*OCBH,*2,/* RES 0,B */
/*OCBH,*2,/* RES 0,C */
/*OCBH,*2,/* RES 0,D */
/*OCBH,*2,/* RES 0,E */
/*OCBH,*2,/* RES 0,H */
/*OCBH,*2,/* RES 0,L */
/*OCBH,*2,/* RES 1,(HL) */
/*ODDH,*4,/* RES 1,(IX+00) */
/*OFDH,*4,/* RES 1,(IY+00) */
/*OCBH,*2,/* RES 1,A */
/*OCBH,*2,/* RES 1,B */
/*OCBH,*2,/* RES 1,C */
/*OCBH,*2,/* RES 1,D */
/*OCBH,*2,/* RES 1,E */
/*OCBH,*2,/* RES 1,H */
/*OCBH,*2,/* RES 1,L */
/*OCBH,*2,/* RES 2,(HL) */
/*ODDH,*4,/* RES 2,(IX+00) */
/*OFDH,*4,/* RES 2,(IY+00) */
/*OCBH,*2,/* RES 2,A */
/*OCBH,*2,/* RES 2,B */
/*OCBH,*2,/* RES 2,C */
/*OCBH,*2,/* RES 2,D */
/*OCBH,*2,/* RES 2,E */
/*OCBH,*2,/* RES 2,H */
/*OCBH,*2,/* RES 2,L */
/*OCBH,*2,/* RES 3,(HL) */
/*ODDH,*4,/* RES 3,(IX+00) */
/*OFDH,*4,/* RES 3,(IY+00) */
/*OCBH,*2,/* RES 3,A */
/*OCBH,*2,/* RES 3,B */
/*OCBH,*2,/* RES 3,C */
/*OCBH,*2,/* RES 3,D */
/*OCBH,*2,/* RES 3,E */
/*OCBH,*2,/* RES 3,H */
/*OCBH,*2,/* RES 3,L */
/*OCBH,*2,/* RES 4,(HL) */
/*ODDH,*4,/* RES 4,(IX+00) */
/*OFDH,*4,/* RES 4,(IY+00) */
/*OCBH,*2,/* RES 4,A */
/*OCBH,*2,/* RES 4,B */
/*OCBH,*2,/* RES 4,C */
/*OCBH,*2,/* RES 4,D */

```




```

/*OCBH,*/2,/* RES 4,E */
/*OCBH,*/2,/* RES 4,H */
/*OCBH,*/2,/* RES 4,L */
/*OCBH,*/2,/* RES 5,(HL) */
/*ODDH,*/4,/* RES 5,(IX+00) */
/*OFDH,*/4,/* RES 5,(IY+00) */
/*OCBH,*/2,/* RES 5,A */
/*OCBH,*/2,/* RES 5,B */
/*OCBH,*/2,/* RES 5,C */
/*OCBH,*/2,/* RES 5,D */
/*OCBH,*/2,/* RES 5,E */
/*OCBH,*/2,/* RES 5,H */
/*OCBH,*/2,/* RES 5,L */
/*OCBH,*/2,/* RES 6,(HL) */
/*ODDH,*/4,/* RES 6,(IX+00) */
/*OFDH,*/4,/* RES 6,(IY+00) */
/*OCBH,*/2,/* RES 6,A */
/*OCBH,*/2,/* RES 6,B */
/*OCBH,*/2,/* RES 6,C */
/*OCBH,*/2,/* RES 6,D */
/*OCBH,*/2,/* RES 6,E */
/*OCBH,*/2,/* RES 6,H */
/*OCBH,*/2,/* RES 6,L */
/*OCBH,*/2,/* RES 7,(HL) */
/*ODDH,*/4,/* RES 7,(IX+00) */
/*OFDH,*/4,/* RES 7,(IY+00) */
/*OCBH,*/2,/* RES 7,A */
/*OCBH,*/2,/* RES 7,B */
/*OCBH,*/2,/* RES 7,C */
/*OCBH,*/2,/* RES 7,D */
/*OCBH,*/2,/* RES 7,E */
/*OCBH,*/2,/* RES 7,H */
/*OCBH,*/2,/* RES 7,L */
/*OC9H,*/1,/* RET */
/*OD8H,*/1,/* RET C */
/*OF8H,*/1,/* RET M */
/*ODOH,*/1,/* RET NC */
/*OCO8,*/1,/* RET NZ */
/*OFOH,*/1,/* RET P */
/*OE8H,*/1,/* RET PE */
/*OEOH,*/1,/* RET PO */
/*OC8H,*/1,/* RET Z */
/*OEDH,*/2,/* RETI */
/*OEDH,*/2,/* RETN */
/*OCBH,*/2,/* RL (HL) */
/*ODDH,*/4,/* RL (IX+00) */
/*OFDH,*/4,/* RL (IY+00) */
/*OCBH,*/2,/* RL A */
/*OCBH,*/2,/* RL B */
/*OCBH,*/2,/* RL C */
/*OCBH,*/2,/* RL D */
/*OCBH,*/2,/* RL E */
/*OCBH,*/2,/* RL H */
/*OCBH,*/2,/* RL L */
/*O17H,*/1,/* RLA */
/*OCBH,*/2,/* RLC (HL) */
/*ODDH,*/4,/* RLC (IX+00) */
/*OFDH,*/4,/* RLC (IY+00) */
/*OCBH,*/2,/* RLC A */
/*OCBH,*/2,/* RLC B */
/*OCBH,*/2,/* RLC C */
/*OCBH,*/2,/* RLC D */
/*OCBH,*/2,/* RLC E */
/*OCBH,*/2,/* RLC H */
/*OCBH,*/2,/* RLC L */
/*O07H,*/1,/* RLCA */
/*OEDH,*/2,/* RLD */
/*OCBH,*/2,/* RR (HL) */
/*ODDH,*/4,/* RR (IX+00) */
/*OFDH,*/4,/* RR (IY+00) */
/*OCBH,*/2,/* RR A */
/*OCBH,*/2,/* RR B */
/*OCBH,*/2,/* RR C */
/*OCBH,*/2,/* RR D */
/*OCBH,*/2,/* RR E */
/*OCBH,*/2,/* RR H */
/*OCBH,*/2,/* RR L */
/*O1FH,*/1,/* RRA */
/*OCBH,*/2,/* RRC (HL) */
/*ODDH,*/4,/* RRC (IX+00) */
/*OFDH,*/4,/* RRC (IY+00) */
/*OCBH,*/2,/* RRC A */
/*OCBH,*/2,/* RRC B */
/*OCBH,*/2,/* RRC C */
/*OCBH,*/2,/* RRC D */
/*OCBH,*/2,/* RRC E */
/*OCBH,*/2,/* RRC H */
/*OCBH,*/2,/* RRC L */
/*OOFH,*/1,/* RRCA */
/*OEDH,*/2,/* RRD */

```



ศูนย์วิทยทรัพยากร
 วิศวกรรมมหาวิทาลัย

```

/*0C7H,*1,/* RST 00 */
/*0C7H,*1,/* RST 0 */
/*0CFH,*1,/* RST 08 */
/*0CFH,*1,/* RST 8 */
/*0D7H,*1,/* RST 10 */
/*0DFH,*1,/* RST 18 */
/*0E7H,*1,/* RST 20 */
/*0EFH,*1,/* RST 28 */
/*0F7H,*1,/* RST 30 */
/*0FFH,*1,/* RST 38 */
/*09EH,*1,/* SBC A,(HL) */
/*0DDH,*3,/* SBC A,(IX+00) */
/*0FDH,*3,/* SBC A,(IY+00) */
/*09FH,*1,/* SBC A,A */
/*098H,*1,/* SBC A,B */
/*099H,*1,/* SBC A,C */
/*09AH,*1,/* SBC A,D */
/*09BH,*1,/* SBC A,E */
/*09CH,*1,/* SBC A,H */
/*09DH,*1,/* SBC A,L */
/*0DEH,*2,/* SBC A,00 */
/*0EDH,*2,/* SBC HL,BC */
/*0EDH,*2,/* SBC HL,DE */
/*0EDH,*2,/* SBC HL,HL */
/*0EDH,*2,/* SBC HL,SP */
/*037H,*1,/* SCF */
/*0CBH,*2,/* SET 0,(HL) */
/*0DDH,*4,/* SET 0,(IX+00) */
/*0FDH,*4,/* SET 0,(IY+00) */
/*0CBH,*2,/* SET 0,A */
/*0CBH,*2,/* SET 0,B */
/*0CBH,*2,/* SET 0,C */
/*0CBH,*2,/* SET 0,D */
/*0CBH,*2,/* SET 0,E */
/*0CBH,*2,/* SET 0,H */
/*0CBH,*2,/* SET 0,L */
/*0CBH,*2,/* SET 1,(HL) */
/*0DDH,*4,/* SET 1,(IX+00) */
/*0FDH,*4,/* SET 1,(IY+00) */
/*0CBH,*2,/* SET 1,A */
/*0CBH,*2,/* SET 1,B */
/*0CBH,*2,/* SET 1,C */
/*0CBH,*2,/* SET 1,D */
/*0CBH,*2,/* SET 1,E */
/*0CBH,*2,/* SET 1,H */
/*0CBH,*2,/* SET 1,L */
/*0CBH,*2,/* SET 2,(HL) */
/*0DDH,*4,/* SET 2,(IX+00) */
/*0FDH,*4,/* SET 2,(IY+00) */
/*0CBH,*2,/* SET 2,A */
/*0CBH,*2,/* SET 2,B */
/*0CBH,*2,/* SET 2,C */
/*0CBH,*2,/* SET 2,D */
/*0CBH,*2,/* SET 2,E */
/*0CBH,*2,/* SET 2,H */
/*0CBH,*2,/* SET 2,L */
/*0CBH,*2,/* SET 3,(HL) */
/*0DDH,*4,/* SET 3,(IX+00) */
/*0FDH,*4,/* SET 3,(IY+00) */
/*0CBH,*2,/* SET 3,A */
/*0CBH,*2,/* SET 3,B */
/*0CBH,*2,/* SET 3,C */
/*0CBH,*2,/* SET 3,D */
/*0CBH,*2,/* SET 3,E */
/*0CBH,*2,/* SET 3,H */
/*0CBH,*2,/* SET 3,L */
/*0CBH,*2,/* SET 4,(HL) */
/*0DDH,*4,/* SET 4,(IX+00) */
/*0FDH,*4,/* SET 4,(IY+00) */
/*0CBH,*2,/* SET 4,A */
/*0CBH,*2,/* SET 4,B */
/*0CBH,*2,/* SET 4,C */
/*0CBH,*2,/* SET 4,D */
/*0CBH,*2,/* SET 4,E */
/*0CBH,*2,/* SET 4,H */
/*0CBH,*2,/* SET 4,L */
/*0CBH,*2,/* SET 5,(HL) */
/*0DDH,*4,/* SET 5,(IX+00) */
/*0FDH,*4,/* SET 5,(IY+00) */
/*0CBH,*2,/* SET 5,A */
/*0CBH,*2,/* SET 5,B */
/*0CBH,*2,/* SET 5,C */
/*0CBH,*2,/* SET 5,D */
/*0CBH,*2,/* SET 5,E */
/*0CBH,*2,/* SET 5,H */
/*0CBH,*2,/* SET 5,L */
/*0CBH,*2,/* SET 6,(HL) */
/*0DDH,*4,/* SET 6,(IX+00) */
/*0FDH,*4,/* SET 6,(IY+00) */
/*0CBH,*2,/* SET 6,A */

```



มหาวิทยาลัย
 วิทยาลัยพยาบาล
 วิทยาลัยพยาบาล

```

/*OCBH,*/2,/* SET 6,B */
/*OCBH,*/2,/* SET 6,C */
/*OCBH,*/2,/* SET 6,D */
/*OCBH,*/2,/* SET 6,E */
/*OCBH,*/2,/* SET 6,H */
/*OCBH,*/2,/* SET 6,L */
/*OCBH,*/2,/* SET 7,(HL) */
/*ODDH,*/4,/* SET 7,(IX+00) */
/*OFDH,*/4,/* SET 7,(IY+00) */
/*OCBH,*/2,/* SET 7,A */
/*OCBH,*/2,/* SET 7,B */
/*OCBH,*/2,/* SET 7,C */
/*OCBH,*/2,/* SET 7,D */
/*OCBH,*/2,/* SET 7,E */
/*OCBH,*/2,/* SET 7,H */
/*OCBH,*/2,/* SET 7,L */
/*OCBH,*/2,/* SLA (HL) */
/*ODDH,*/4,/* SLA (IX+00) */
/*OFDH,*/4,/* SLA (IY+00) */
/*OCBH,*/2,/* SLA A */
/*OCBH,*/2,/* SLA B */
/*OCBH,*/2,/* SLA C */
/*OCBH,*/2,/* SLA D */
/*OCBH,*/2,/* SLA E */
/*OCBH,*/2,/* SLA H */
/*OCBH,*/2,/* SLA L */
/*OCBH,*/2,/* SRA (HL) */
/*ODDH,*/4,/* SRA (IX+00) */
/*OFDH,*/4,/* SRA (IY+00) */
/*OCBH,*/2,/* SRA A */
/*OCBH,*/2,/* SRA B */
/*OCBH,*/2,/* SRA C */
/*OCBH,*/2,/* SRA D */
/*OCBH,*/2,/* SRA E */
/*OCBH,*/2,/* SRA H */
/*OCBH,*/2,/* SRA L */
/*OCBH,*/2,/* SRL (HL) */
/*ODDH,*/4,/* SRL (IX+00) */
/*OFDH,*/4,/* SRL (IY+00) */
/*OCBH,*/2,/* SRL A */
/*OCBH,*/2,/* SRL B */
/*OCBH,*/2,/* SRL C */
/*OCBH,*/2,/* SRL D */
/*OCBH,*/2,/* SRL E */
/*OCBH,*/2,/* SRL H */
/*OCBH,*/2,/* SRL L */
/*096H,*/1,/* SUB (HL) */
/*ODDH,*/3,/* SUB (IX+00) */
/*OFDH,*/3,/* SUB (IY+00) */
/*097H,*/1,/* SUB A */
/*090H,*/1,/* SUB B */
/*091H,*/1,/* SUB C */
/*092H,*/1,/* SUB D */
/*093H,*/1,/* SUB E */
/*094H,*/1,/* SUB H */
/*095H,*/1,/* SUB L */
/*0D6H,*/2,/* SUB 00 */
/*0AEH,*/1,/* XOR (HL) */
/*ODDH,*/3,/* XOR (IX+00) */
/*OFDH,*/3,/* XOR (IY+00) */
/*0AFH,*/1,/* XOR A */
/*0A8H,*/1,/* XOR B */
/*0A9H,*/1,/* XOR C */
/*0AAH,*/1,/* XOR D */
/*0ABH,*/1,/* XOR E */
/*0ACH,*/1,/* XOR H */
/*0ADH,*/1,/* XOR L */
/*0EEH,*/2,/* XOR 00 */

```



มหาวิทยาลัยเทคโนโลยีพระยากร
 วิทยาลัย

'ADC'
'ADD'
'AND'
'BIT'
'CALL'
'CCF'
'CP'
'CPD'
'CPDR'
'CPI'
'CPIR'
'CPL'
'DAA'
'DEC'
'DI'
'DJNZ'
'EI'
'EX'
'EXX'
'HALT'
'IM'
'IN'
'INC'
'IND'
'INDR'
'INI'
'INIR'
'JP'
'JR'
'LD'
'LDD'
'LDDR'
'LDI'
'LDIR'
'NEG'
'NOP'
'OR'
'OTDR'
'OTIR'
'OUT'
'OUTD'
'OUTI'
'POP'
'PUSH'
'RES'
'RET'
'RETI'
'RETN'
'RL'
'RLA'
'RLC'
'RLCA'
'RLD'
'RR'
'RRA'
'RRC'
'RRCA'
'RRD'
'RST'
'SBC'
'SCF'
'SET'
'SLA'
'SRA'
'SRL'
'SUB'
'XOR'
'DB'
'DW'



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* 1*/ '0
/* 2*/ '1
/* 3*/ '2
/* 4*/ '3
/* 5*/ '4
/* 6*/ '5
/* 7*/ '6
/* 8*/ '7
/* 9*/ '8
/*10*/ '00
/*11*/ '08
/*12*/ '10
/*13*/ '18
/*14*/ '20
/*15*/ '28
/*16*/ '30
/*17*/ '38
/*18*/ '( HL )
/*19*/ '( IX )
/*20*/ '( IY )
/*21*/ 'C
/*22*/ 'M
/*23*/ 'NC
/*24*/ 'NZ
/*25*/ 'P
/*26*/ 'PE
/*27*/ 'PO
/*28*/ 'Z
/*29*/ 'a
/*30*/ 'r
/*31*/ '( IX d )
/*32*/ '( IY d )
/*33*/ 'A
/*34*/ 'B
/*35*/ 'D
/*36*/ 'E
/*37*/ 'H
/*38*/ 'L
/*39*/ 'I
/*40*/ 'R
/*41*/ 'b
/*42*/ '( C )
/*43*/ '( b )
/*44*/ '( a )
/*45*/ '( BC )
/*46*/ '( DE )
/*47*/ '( SP )
/*48*/ 'AF',27h
/*49*/ 'AF
/*50*/ 'BC
/*51*/ 'DE
/*52*/ 'HL
/*53*/ 'SP
/*54*/ 'IX
/*55*/ 'IY

```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* 1  ADC */,1
/* 2  ADD */,16
/* 3  AND */,39
/* 4  BIT */,50
/* 5  CALL */,130
/* 6  CCF */,139
/* 7  CP */,140
/* 8  CPD */,151
/* 9  CPDR */,152
/* 10 CPI */,153
/* 11 CPIR */,154
/* 12 CPL */,155
/* 13 DAA */,156
/* 14 DEC */,157
/* 15 DI */,173
/* 16 DJNZ */,174
/* 17 EI */,175
/* 18 EX */,176
/* 19 EXX */,181
/* 20 HALT */,182
/* 21 IM */,183
/* 22 IN */,186
/* 23 INC */,194
/* 24 IND */,210
/* 25 INDR */,211
/* 26 INI */,212
/* 27 INIR */,213
/* 28 JP */,214
/* 29 JR */,226
/* 30 LD */,231
/* 31 LDD */,363
/* 32 LDDR */,364
/* 33 LDI */,365
/* 34 LDIR */,366
/* 35 NEG */,367
/* 36 NOP */,368
/* 37 OR */,369
/* 38 OTDR */,380
/* 39 OTIR */,381
/* 40 OUT */,382
/* 41 OUTD */,390
/* 42 OUTI */,391
/* 43 POP */,392
/* 44 PUSH */,398
/* 45 RES */,404
/* 46 RET */,484
/* 47 RETI */,493
/* 48 RETN */,494
/* 49 RL */,495
/* 50 RLA */,505
/* 51 RLC */,506
/* 52 RLCA */,516
/* 53 RLD */,517
/* 54 RR */,518
/* 55 RRA */,528
/* 56 RRC */,529
/* 57 RRCA */,539
/* 58 RRD */,540
/* 59 RST */,541
/* 60 SBC */,551
/* 61 SCF */,566
/* 62 SET */,567
/* 63 SLA */,647
/* 64 SRA */,657
/* 65 SRL */,667
/* 66 SUB */,677
/* 67 XOR */,688
/* 68 DB */,0
/* 69 DW */,0

```



ศูนย์วิทยทรัพยากร

ศาลงกรณ์มหาวิทยาลัย

08EH, 000H, 000H, /*	ADC	A, (HL) */
0DDH, 08EH, 000H, /*	ADC	A, (IX+b) */
0FDH, 08EH, 000H, /*	ADC	A, (IY+b) */
08FH, 000H, 000H, /*	ADC	A, A */
088H, 000H, 000H, /*	ADC	A, B */
089H, 000H, 000H, /*	ADC	A, C */
08AH, 000H, 000H, /*	ADC	A, D */
08BH, 000H, 000H, /*	ADC	A, E */
08CH, 000H, 000H, /*	ADC	A, H */
08DH, 000H, 000H, /*	ADC	A, L */
0CEH, 000H, 000H, /*	ADC	A, b */
0EDH, 04AH, 000H, /*	ADC	HL, BC */
0EDH, 05AH, 000H, /*	ADC	HL, DE */
0EDH, 06AH, 000H, /*	ADC	HL, HL */
0EDH, 07AH, 000H, /*	ADC	HL, SP */
086H, 000H, 000H, /*	ADD	A, (HL) */
0DDH, 086H, 000H, /*	ADD	A, (IX+b) */
0FDH, 086H, 000H, /*	ADD	A, (IY+b) */
087H, 000H, 000H, /*	ADD	A, A */
080H, 000H, 000H, /*	ADD	A, B */
081H, 000H, 000H, /*	ADD	A, C */
082H, 000H, 000H, /*	ADD	A, D */
083H, 000H, 000H, /*	ADD	A, E */
084H, 000H, 000H, /*	ADD	A, H */
085H, 000H, 000H, /*	ADD	A, L */
0C6H, 000H, 000H, /*	ADD	A, b */
009H, 000H, 000H, /*	ADD	HL, BC */
019H, 000H, 000H, /*	ADD	HL, DE */
029H, 000H, 000H, /*	ADD	HL, HL */
039H, 000H, 000H, /*	ADD	HL, SP */
0DDH, 009H, 000H, /*	ADD	IX, BC */
0DDH, 019H, 000H, /*	ADD	IX, DE */
0DDH, 029H, 000H, /*	ADD	IX, IX */
0DDH, 039H, 000H, /*	ADD	IX, SP */
0FDH, 009H, 000H, /*	ADD	IY, BC */
0FDH, 019H, 000H, /*	ADD	IY, DE */
0FDH, 029H, 000H, /*	ADD	IY, IY */
0FDH, 039H, 000H, /*	ADD	IY, SP */
0A6H, 000H, 000H, /*	AND	(HL) */
0DDH, 0A6H, 000H, /*	AND	(IX+b) */
0FDH, 0A6H, 000H, /*	AND	(IY+b) */
0A7H, 000H, 000H, /*	AND	A */
0A0H, 000H, 000H, /*	AND	B */
0A1H, 000H, 000H, /*	AND	C */
0A2H, 000H, 000H, /*	AND	D */
0A3H, 000H, 000H, /*	AND	E */
0A4H, 000H, 000H, /*	AND	H */
0A5H, 000H, 000H, /*	AND	L */
0E6H, 000H, 000H, /*	AND	b */
0CBH, 046H, 000H, /*	BIT	0, (HL) */
0DDH, 0CBH, 046H, /*	BIT	0, (IX+b) */
0FDH, 0CBH, 046H, /*	BIT	0, (IY+b) */
0CBH, 047H, 000H, /*	BIT	0, A */
0CBH, 040H, 000H, /*	BIT	0, B */
0CBH, 041H, 000H, /*	BIT	0, C */
0CBH, 042H, 000H, /*	BIT	0, D */
0CBH, 043H, 000H, /*	BIT	0, E */
0CBH, 044H, 000H, /*	BIT	0, H */
0CBH, 045H, 000H, /*	BIT	0, L */
0CBH, 04EH, 000H, /*	BIT	1, (HL) */
0DDH, 0CBH, 04EH, /*	BIT	1, (IX+b) */
0FDH, 0CBH, 04EH, /*	BIT	1, (IY+b) */
0CBH, 04FH, 000H, /*	BIT	1, A */
0CBH, 048H, 000H, /*	BIT	1, B */
0CBH, 049H, 000H, /*	BIT	1, C */
0CBH, 04AH, 000H, /*	BIT	1, D */
0CBH, 04BH, 000H, /*	BIT	1, E */
0CBH, 04CH, 000H, /*	BIT	1, H */
0CBH, 04DH, 000H, /*	BIT	1, L */
0CBH, 056H, 000H, /*	BIT	2, (HL) */
0DDH, 0CBH, 056H, /*	BIT	2, (IX+b) */
0FDH, 0CBH, 056H, /*	BIT	2, (IY+b) */
0CBH, 057H, 000H, /*	BIT	2, A */
0CBH, 050H, 000H, /*	BIT	2, B */
0CBH, 051H, 000H, /*	BIT	2, C */
0CBH, 052H, 000H, /*	BIT	2, D */
0CBH, 053H, 000H, /*	BIT	2, E */
0CBH, 054H, 000H, /*	BIT	2, H */
0CBH, 055H, 000H, /*	BIT	2, L */
0CBH, 05EH, 000H, /*	BIT	3, (HL) */
0DDH, 0CBH, 05EH, /*	BIT	3, (IX+b) */
0FDH, 0CBH, 05EH, /*	BIT	3, (IY+b) */
0CBH, 05FH, 000H, /*	BIT	3, A */
0CBH, 058H, 000H, /*	BIT	3, B */
0CBH, 059H, 000H, /*	BIT	3, C */
0CBH, 05AH, 000H, /*	BIT	3, D */
0CBH, 05BH, 000H, /*	BIT	3, E */
0CBH, 05CH, 000H, /*	BIT	3, H */
0CBH, 05DH, 000H, /*	BIT	3, L */
0CBH, 066H, 000H, /*	BIT	4, (HL) */

ODDH, 0CBH, 066H, /* BIT 4,(IX+b) */
 OFDH, 0CBH, 066H, /* BIT 4,(IY+b) */
 OCBH, 067H, 000H, /* BIT 4,A */
 OCBH, 060H, 000H, /* BIT 4,B */
 OCBH, 061H, 000H, /* BIT 4,C */
 OCBH, 062H, 000H, /* BIT 4,D */
 OCBH, 063H, 000H, /* BIT 4,E */
 OCBH, 064H, 000H, /* BIT 4,H */
 OCBH, 065H, 000H, /* BIT 4,L */
 OCBH, 06EH, 000H, /* BIT 5,(HL) */
 ODDH, 0CBH, 06EH, /* BIT 5,(IX+b) */
 OFDH, 0CBH, 06EH, /* BIT 5,(IY+b) */
 OCBH, 06FH, 000H, /* BIT 5,A */
 OCBH, 068H, 000H, /* BIT 5,B */
 OCBH, 069H, 000H, /* BIT 5,C */
 OCBH, 06AH, 000H, /* BIT 5,D */
 OCBH, 06BH, 000H, /* BIT 5,E */
 OCBH, 06CH, 000H, /* BIT 5,H */
 OCBH, 06DH, 000H, /* BIT 5,L */
 OCBH, 076H, 000H, /* BIT 6,(HL) */
 ODDH, 0CBH, 076H, /* BIT 6,(IX+b) */
 OFDH, 0CBH, 076H, /* BIT 6,(IY+b) */
 OCBH, 077H, 000H, /* BIT 6,A */
 OCBH, 070H, 000H, /* BIT 6,B */
 OCBH, 071H, 000H, /* BIT 6,C */
 OCBH, 072H, 000H, /* BIT 6,D */
 OCBH, 073H, 000H, /* BIT 6,E */
 OCBH, 074H, 000H, /* BIT 6,H */
 OCBH, 075H, 000H, /* BIT 6,L */
 OCBH, 07EH, 000H, /* BIT 7,(HL) */
 ODDH, 0CBH, 07EH, /* BIT 7,(IX+b) */
 OFDH, 0CBH, 07EH, /* BIT 7,(IY+b) */
 OCBH, 07FH, 000H, /* BIT 7,A */
 OCBH, 078H, 000H, /* BIT 7,B */
 OCBH, 079H, 000H, /* BIT 7,C */
 OCBH, 07AH, 000H, /* BIT 7,D */
 OCBH, 07BH, 000H, /* BIT 7,E */
 OCBH, 07CH, 000H, /* BIT 7,H */
 OCBH, 07DH, 000H, /* BIT 7,L */
 ODCH, 000H, 000H, /* CALL C,a */
 OFCH, 000H, 000H, /* CALL M,a */
 OD4H, 000H, 000H, /* CALL NC,a */
 OC4H, 000H, 000H, /* CALL NZ,a */
 OF4H, 000H, 000H, /* CALL P,a */
 OECH, 000H, 000H, /* CALL PE,a */
 OE4H, 000H, 000H, /* CALL PO,a */
 OCCH, 000H, 000H, /* CALL Z,a */
 OCDH, 000H, 000H, /* CALL a */
 O3FH, 000H, 000H, /* CCF */
 OBEH, 000H, 000H, /* CP (HL) */
 ODDH, OBEH, 000H, /* CP (IX+b) */
 OFDH, OBEH, 000H, /* CP (IY+b) */
 OBFH, 000H, 000H, /* CP A */
 O88H, 000H, 000H, /* CP B */
 OB9H, 000H, 000H, /* CP C */
 OBAH, 000H, 000H, /* CP D */
 OBBH, 000H, 000H, /* CP E */
 OBCH, 000H, 000H, /* CP H */
 OBDH, 000H, 000H, /* CP L */
 OFEH, 000H, 000H, /* CP b */
 OEDH, 0A9H, 000H, /* CPD */
 OEDH, 0B9H, 000H, /* CPDR */
 OEDH, 0A1H, 000H, /* CPI */
 OEDH, 0B1H, 000H, /* CPIR */
 O2FH, 000H, 000H, /* CPL */
 O27H, 000H, 000H, /* DAA */
 O35H, 000H, 000H, /* DEC (HL) */
 ODDH, 035H, 000H, /* DEC (IX+b) */
 OFDH, 035H, 000H, /* DEC (IY+b) */
 O3DH, 000H, 000H, /* DEC A */
 O05H, 000H, 000H, /* DEC B */
 O0BH, 000H, 000H, /* DEC BC */
 O0DH, 000H, 000H, /* DEC C */
 O15H, 000H, 000H, /* DEC D */
 O1BH, 000H, 000H, /* DEC DE */
 O1DH, 000H, 000H, /* DEC E */
 O25H, 000H, 000H, /* DEC H */
 O2BH, 000H, 000H, /* DEC HL */
 ODDH, 02BH, 000H, /* DEC IX */
 OFDH, 02BH, 000H, /* DEC IY */
 O2DH, 000H, 000H, /* DEC L */
 O3BH, 000H, 000H, /* DEC SP */
 OF3H, 000H, 000H, /* DI */
 O10H, 000H, 000H, /* DJNZ r */
 OFBH, 000H, 000H, /* EI */
 OE3H, 000H, 000H, /* EX (SP),HL */
 ODDH, OE3H, 000H, /* EX (SP),IX */
 OFDH, OE3H, 000H, /* EX (SP),IY */
 O08H, 000H, 000H, /* EX AF,AF' */
 OE3H, 000H, 000H, /* EX DE,HL */

0D9H, 000H, 000H, /*	EXX	*/
076H, 000H, 000H, /*	HALT	*/
0EDH, 046H, 000H, /*	IM	0 */
0EDH, 056H, 000H, /*	IM	1 */
0EDH, 05EH, 000H, /*	IM	2 */
0EDH, 078H, 000H, /*	IN	A, (C) */
0DBH, 000H, 000H, /*	IN	A, (b) */
0EDH, 040H, 000H, /*	IN	B, (C) */
0EDH, 048H, 000H, /*	IN	C, (C) */
0EDH, 050H, 000H, /*	IN	D, (C) */
0EDH, 058H, 000H, /*	IN	E, (C) */
0EDH, 060H, 000H, /*	IN	H, (C) */
0EDH, 068H, 000H, /*	IN	L, (C) */
034H, 000H, 000H, /*	INC	(HL) */
0DDH, 034H, 000H, /*	INC	(IX+b) */
0FDH, 034H, 000H, /*	INC	(IY+b) */
03CH, 000H, 000H, /*	INC	A */
004H, 000H, 000H, /*	INC	B */
003H, 000H, 000H, /*	INC	BC */
00CH, 000H, 000H, /*	INC	C */
014H, 000H, 000H, /*	INC	D */
013H, 000H, 000H, /*	INC	DE */
01CH, 000H, 000H, /*	INC	E */
024H, 000H, 000H, /*	INC	H */
023H, 000H, 000H, /*	INC	HL */
0DDH, 023H, 000H, /*	INC	IX */
0FDH, 023H, 000H, /*	INC	IY */
02CH, 000H, 000H, /*	INC	L */
033H, 000H, 000H, /*	INC	SP */
0EDH, 0AAH, 000H, /*	IND	*/
0EDH, 0BAH, 000H, /*	INDR	*/
0EDH, 0A2H, 000H, /*	INI	*/
0EDH, 0B2H, 000H, /*	INIR	*/
0E9H, 000H, 000H, /*	JP	(HL) */
0DDH, 0E9H, 000H, /*	JP	(IX) */
0FDH, 0E9H, 000H, /*	JP	(IY) */
0DAH, 000H, 000H, /*	JP	C, a */
0FAH, 000H, 000H, /*	JP	M, a */
0D2H, 000H, 000H, /*	JP	NC, a */
0C2H, 000H, 000H, /*	JP	NZ, a */
0F2H, 000H, 000H, /*	JP	P, a */
0EAH, 000H, 000H, /*	JP	PE, a */
0E2H, 000H, 000H, /*	JP	PO, a */
0CAH, 000H, 000H, /*	JP	Z, a */
0C3H, 000H, 000H, /*	JP	a */
038H, 000H, 000H, /*	JR	C, r */
030H, 000H, 000H, /*	JR	NC, r */
020H, 000H, 000H, /*	JR	NZ, r */
028H, 000H, 000H, /*	JR	Z, r */
018H, 000H, 000H, /*	JR	r */
002H, 000H, 000H, /*	LD	(BC), A */
012H, 000H, 000H, /*	LD	(DE), A */
077H, 000H, 000H, /*	LD	(HL), A */
070H, 000H, 000H, /*	LD	(HL), B */
071H, 000H, 000H, /*	LD	(HL), C */
072H, 000H, 000H, /*	LD	(HL), D */
073H, 000H, 000H, /*	LD	(HL), E */
074H, 000H, 000H, /*	LD	(HL), H */
075H, 000H, 000H, /*	LD	(HL), L */
036H, 000H, 000H, /*	LD	(HL), b */
0DDH, 077H, 000H, /*	LD	(IX+b), A */
0DDH, 070H, 000H, /*	LD	(IX+b), B */
0DDH, 071H, 000H, /*	LD	(IX+b), C */
0DDH, 072H, 000H, /*	LD	(IX+b), D */
0DDH, 073H, 000H, /*	LD	(IX+b), E */
0DDH, 074H, 000H, /*	LD	(IX+b), H */
0DDH, 075H, 000H, /*	LD	(IX+b), L */
0DDH, 036H, 000H, /*	LD	(IX+b), b */
0FDH, 077H, 000H, /*	LD	(IY+b), A */
0FDH, 070H, 000H, /*	LD	(IY+b), B */
0FDH, 071H, 000H, /*	LD	(IY+b), C */
0FDH, 072H, 000H, /*	LD	(IY+b), D */
0FDH, 073H, 000H, /*	LD	(IY+b), E */
0FDH, 074H, 000H, /*	LD	(IY+b), H */
0FDH, 075H, 000H, /*	LD	(IY+b), L */
0FDH, 036H, 000H, /*	LD	(IY+b), b */
032H, 000H, 000H, /*	LD	(a), A */
0EDH, 043H, 000H, /*	LD	(a), BC */
0EDH, 053H, 000H, /*	LD	(a), DE */
022H, 000H, 000H, /*	LD	(a), HL */
0DDH, 022H, 000H, /*	LD	(a), IX */
0FDH, 022H, 000H, /*	LD	(a), IY */
0EDH, 073H, 000H, /*	LD	(a), SP */
00AH, 000H, 000H, /*	LD	A, (BC) */
01AH, 000H, 000H, /*	LD	A, (DE) */
07EH, 000H, 000H, /*	LD	A, (HL) */
0DDH, 07EH, 000H, /*	LD	A, (IX+b) */
0FDH, 07EH, 000H, /*	LD	A, (IY+b) */
03AH, 000H, 000H, /*	LD	A, (a) */
07FH, 000H, 000H, /*	LD	A, A */



078H, 000H, 000H, /* LD	A, B */
079H, 000H, 000H, /* LD	A, C */
07AH, 000H, 000H, /* LD	A, D */
07BH, 000H, 000H, /* LD	A, E */
07CH, 000H, 000H, /* LD	A, H */
0EDH, 057H, 000H, /* LD	A, I */
07DH, 000H, 000H, /* LD	A, L */
0EDH, 05FH, 000H, /* LD	A, R */
03EH, 000H, 000H, /* LD	A, b */
046H, 000H, 000H, /* LD	B, (HL) */
0DDH, 046H, 000H, /* LD	B, (IX+b) */
0FDH, 046H, 000H, /* LD	B, (IY+b) */
047H, 000H, 000H, /* LD	B, A */
040H, 000H, 000H, /* LD	B, B */
041H, 000H, 000H, /* LD	B, C */
042H, 000H, 000H, /* LD	B, D */
043H, 000H, 000H, /* LD	B, E */
044H, 000H, 000H, /* LD	B, H */
045H, 000H, 000H, /* LD	B, L */
006H, 000H, 000H, /* LD	B, b */
0EDH, 04BH, 000H, /* LD	BC, (a) */
001H, 000H, 000H, /* LD	BC, a */
04EH, 000H, 000H, /* LD	C, (HL) */
0DDH, 04EH, 000H, /* LD	C, (IX+b) */
0FDH, 04EH, 000H, /* LD	C, (IY+b) */
04FH, 000H, 000H, /* LD	C, A */
048H, 000H, 000H, /* LD	C, B */
049H, 000H, 000H, /* LD	C, C */
04AH, 000H, 000H, /* LD	C, D */
04BH, 000H, 000H, /* LD	C, E */
04CH, 000H, 000H, /* LD	C, H */
04DH, 000H, 000H, /* LD	C, L */
00EH, 000H, 000H, /* LD	C, b */
056H, 000H, 000H, /* LD	D, (HL) */
0DDH, 056H, 000H, /* LD	D, (IX+b) */
0FDH, 056H, 000H, /* LD	D, (IY+b) */
057H, 000H, 000H, /* LD	D, A */
050H, 000H, 000H, /* LD	D, B */
051H, 000H, 000H, /* LD	D, C */
052H, 000H, 000H, /* LD	D, D */
053H, 000H, 000H, /* LD	D, E */
054H, 000H, 000H, /* LD	D, H */
055H, 000H, 000H, /* LD	D, L */
016H, 000H, 000H, /* LD	D, b */
0EDH, 05BH, 000H, /* LD	DE, (a) */
011H, 000H, 000H, /* LD	DE, a */
05EH, 000H, 000H, /* LD	E, (HL) */
0DDH, 05EH, 000H, /* LD	E, (IX+b) */
0FDH, 05EH, 000H, /* LD	E, (IY+b) */
05FH, 000H, 000H, /* LD	E, A */
058H, 000H, 000H, /* LD	E, B */
059H, 000H, 000H, /* LD	E, C */
05AH, 000H, 000H, /* LD	E, D */
05BH, 000H, 000H, /* LD	E, E */
05CH, 000H, 000H, /* LD	E, H */
05DH, 000H, 000H, /* LD	E, L */
01EH, 000H, 000H, /* LD	E, b */
066H, 000H, 000H, /* LD	H, (HL) */
0DDH, 066H, 000H, /* LD	H, (IX+b) */
0FDH, 066H, 000H, /* LD	H, (IY+b) */
067H, 000H, 000H, /* LD	H, A */
060H, 000H, 000H, /* LD	H, B */
061H, 000H, 000H, /* LD	H, C */
062H, 000H, 000H, /* LD	H, D */
063H, 000H, 000H, /* LD	H, E */
064H, 000H, 000H, /* LD	H, H */
065H, 000H, 000H, /* LD	H, L */
026H, 000H, 000H, /* LD	H, b */
02AH, 000H, 000H, /* LD	HL, (a) */
021H, 000H, 000H, /* LD	HL, a */
0EDH, 047H, 000H, /* LD	I, A */
0DDH, 02AH, 000H, /* LD	IX, (a) */
0DDH, 021H, 000H, /* LD	IX, a */
0FDH, 02AH, 000H, /* LD	IY, (a) */
0FDH, 021H, 000H, /* LD	IY, a */
06EH, 000H, 000H, /* LD	L, (HL) */
0DDH, 06EH, 000H, /* LD	L, (IX+b) */
0FDH, 06EH, 000H, /* LD	L, (IY+b) */
06FH, 000H, 000H, /* LD	L, A */
068H, 000H, 000H, /* LD	L, B */
069H, 000H, 000H, /* LD	L, C */
06AH, 000H, 000H, /* LD	L, D */
06BH, 000H, 000H, /* LD	L, E */
06CH, 000H, 000H, /* LD	L, H */
06DH, 000H, 000H, /* LD	L, L */
02EH, 000H, 000H, /* LD	L, b */
0EDH, 04FH, 000H, /* LD	R, A */
0EDH, 07BH, 000H, /* LD	SP, (a) */
0F9H, 000H, 000H, /* LD	SP, HL */
0DDH, 0F9H, 000H, /* LD	SP, IX */

```

OFDH, 0F9H, 000H, /* LD SP,IY */
O31H, 000H, 000H, /* LD SP,a */
OEDH, 0A8H, 000H, /* LDD */
OEDH, 0B8H, 000H, /* LDDR */
OEDH, 0A0H, 000H, /* LDI */
OEDH, 0B0H, 000H, /* LDIR */
OEDH, 044H, 000H, /* NEG */
000H, 000H, 000H, /* NOP */
OB6H, 000H, 000H, /* OR (HL) */
ODDH, 0B6H, 000H, /* OR (IX+b) */
OFDH, 0B6H, 000H, /* OR (IY+b) */
OB7H, 000H, 000H, /* OR A */
OB0H, 000H, 000H, /* OR B */
OB1H, 000H, 000H, /* OR C */
OB2H, 000H, 000H, /* OR D */
OB3H, 000H, 000H, /* OR E */
OB4H, 000H, 000H, /* OR H */
OB5H, 000H, 000H, /* OR L */
OF6H, 000H, 000H, /* OR b */
OEDH, 0BBH, 000H, /* OTDR */
OEDH, 0B3H, 000H, /* OTIR */
OEDH, 079H, 000H, /* OUT (C),A */
OEDH, 041H, 000H, /* OUT (C),B */
OEDH, 049H, 000H, /* OUT (C),C */
OEDH, 051H, 000H, /* OUT (C),D */
OEDH, 059H, 000H, /* OUT (C),E */
OEDH, 061H, 000H, /* OUT (C),H */
OEDH, 069H, 000H, /* OUT (C),L */
OD3H, 000H, 000H, /* OUT (b),A */
OEDH, 0ABH, 000H, /* OUTD */
OEDH, 0A3H, 000H, /* OUTI */
OF1H, 000H, 000H, /* POP AF */
OC1H, 000H, 000H, /* POP BC */
OD1H, 000H, 000H, /* POP DE */
OE1H, 000H, 000H, /* POP HL */
ODDH, 0E1H, 000H, /* POP IX */
OFDH, 0E1H, 000H, /* POP IY */
OF5H, 000H, 000H, /* PUSH AF */
OC5H, 000H, 000H, /* PUSH BC */
OD5H, 000H, 000H, /* PUSH DE */
OE5H, 000H, 000H, /* PUSH HL */
ODDH, 0E5H, 000H, /* PUSH IX */
OFDH, 0E5H, 000H, /* PUSH IY */
OCBH, 086H, 000H, /* RES 0,(HL) */
ODDH, 0CBH, 086H, /* RES 0,(IX+b) */
OFDH, 0CBH, 086H, /* RES 0,(IY+b) */
OCBH, 087H, 000H, /* RES 0,A */
OCBH, 080H, 000H, /* RES 0,B */
OCBH, 081H, 000H, /* RES 0,C */
OCBH, 082H, 000H, /* RES 0,D */
OCBH, 083H, 000H, /* RES 0,E */
OCBH, 084H, 000H, /* RES 0,H */
OCBH, 085H, 000H, /* RES 0,L */
OCBH, 08EH, 000H, /* RES 1,(HL) */
ODDH, 0CBH, 08EH, /* RES 1,(IX+b) */
OFDH, 0CBH, 08EH, /* RES 1,(IY+b) */
OCBH, 08FH, 000H, /* RES 1,A */
OCBH, 088H, 000H, /* RES 1,B */
OCBH, 089H, 000H, /* RES 1,C */
OCBH, 08AH, 000H, /* RES 1,D */
OCBH, 08BH, 000H, /* RES 1,E */
OCBH, 08CH, 000H, /* RES 1,H */
OCBH, 08DH, 000H, /* RES 1,L */
OCBH, 096H, 000H, /* RES 2,(HL) */
ODDH, 0CBH, 096H, /* RES 2,(IX+b) */
OFDH, 0CBH, 096H, /* RES 2,(IY+b) */
OCBH, 097H, 000H, /* RES 2,A */
OCBH, 090H, 000H, /* RES 2,B */
OCBH, 091H, 000H, /* RES 2,C */
OCBH, 092H, 000H, /* RES 2,D */
OCBH, 093H, 000H, /* RES 2,E */
OCBH, 094H, 000H, /* RES 2,H */
OCBH, 095H, 000H, /* RES 2,L */
OCBH, 09EH, 000H, /* RES 3,(HL) */
ODDH, 0CBH, 09EH, /* RES 3,(IX+b) */
OFDH, 0CBH, 09EH, /* RES 3,(IY+b) */
OCBH, 09FH, 000H, /* RES 3,A */
OCBH, 098H, 000H, /* RES 3,B */
OCBH, 099H, 000H, /* RES 3,C */
OCBH, 09AH, 000H, /* RES 3,D */
OCBH, 09BH, 000H, /* RES 3,E */
OCBH, 09CH, 000H, /* RES 3,H */
OCBH, 09DH, 000H, /* RES 3,L */
OCBH, 0A6H, 000H, /* RES 4,(HL) */
ODDH, 0CBH, 0A6H, /* RES 4,(IX+b) */
OFDH, 0CBH, 0A6H, /* RES 4,(IY+b) */
OCBH, 0A7H, 000H, /* RES 4,A */
OCBH, 0A0H, 000H, /* RES 4,B */
OCBH, 0A1H, 000H, /* RES 4,C */
OCBH, 0A2H, 000H, /* RES 4,D */

```



วิทยาลัยพยาบาล
รัตนมทาววิทยาลัย

OCBH, 0A3H, 000H, /*	RES	4, E	*/
OCBH, 0A4H, 000H, /*	RES	4, H	*/
OCBH, 0A5H, 000H, /*	RES	4, L	*/
OCBH, 0AEH, 000H, /*	RES	5, (HL)	*/
ODDH, 0CBH, 0AEH, /*	RES	5, (IX+b)	*/
OFDH, 0CBH, 0AEH, /*	RES	5, (IY+b)	*/
OCBH, 0AFH, 000H, /*	RES	5, A	*/
OCBH, 0A8H, 000H, /*	RES	5, B	*/
OCBH, 0A9H, 000H, /*	RES	5, C	*/
OCBH, 0AAH, 000H, /*	RES	5, D	*/
OCBH, 0ABH, 000H, /*	RES	5, E	*/
OCBH, 0ACH, 000H, /*	RES	5, H	*/
OCBH, 0ADH, 000H, /*	RES	5, L	*/
OCBH, 0B6H, 000H, /*	RES	6, (HL)	*/
ODDH, 0CBH, 0B6H, /*	RES	6, (IX+b)	*/
OFDH, 0CBH, 0B6H, /*	RES	6, (IY+b)	*/
OCBH, 0B7H, 000H, /*	RES	6, A	*/
OCBH, 0B0H, 000H, /*	RES	6, B	*/
OCBH, 0B1H, 000H, /*	RES	6, C	*/
OCBH, 0B2H, 000H, /*	RES	6, D	*/
OCBH, 0B3H, 000H, /*	RES	6, E	*/
OCBH, 0B4H, 000H, /*	RES	6, H	*/
OCBH, 0B5H, 000H, /*	RES	6, L	*/
OCBH, 0BEH, 000H, /*	RES	7, (HL)	*/
ODDH, 0CBH, 0BEH, /*	RES	7, (IX+b)	*/
OFDH, 0CBH, 0BEH, /*	RES	7, (IY+b)	*/
OCBH, 0BFH, 000H, /*	RES	7, A	*/
OCBH, 0B8H, 000H, /*	RES	7, B	*/
OCBH, 0B9H, 000H, /*	RES	7, C	*/
OCBH, 0BAH, 000H, /*	RES	7, D	*/
OCBH, 0BBH, 000H, /*	RES	7, E	*/
OCBH, 0BCH, 000H, /*	RES	7, H	*/
OCBH, 0BDH, 000H, /*	RES	7, L	*/
OC9H, 000H, 000H, /*	RET		*/
OD8H, 000H, 000H, /*	RET	C	*/
OF8H, 000H, 000H, /*	RET	M	*/
OD0H, 000H, 000H, /*	RET	NC	*/
OC0H, 000H, 000H, /*	RET	NZ	*/
OF0H, 000H, 000H, /*	RET	P	*/
OE8H, 000H, 000H, /*	RET	PE	*/
OE0H, 000H, 000H, /*	RET	PO	*/
OC8H, 000H, 000H, /*	RET	Z	*/
OEDH, 04DH, 000H, /*	RETI		*/
OEDH, 045H, 000H, /*	RETN		*/
OCBH, 016H, 000H, /*	RL	(HL)	*/
ODDH, 0CBH, 016H, /*	RL	(IX+b)	*/
OFDH, 0CBH, 016H, /*	RL	(IY+b)	*/
OCBH, 017H, 000H, /*	RL	A	*/
OCBH, 010H, 000H, /*	RL	B	*/
OCBH, 011H, 000H, /*	RL	C	*/
OCBH, 012H, 000H, /*	RL	D	*/
OCBH, 013H, 000H, /*	RL	E	*/
OCBH, 014H, 000H, /*	RL	H	*/
OCBH, 015H, 000H, /*	RL	L	*/
017H, 000H, 000H, /*	RLA		*/
OCBH, 006H, 000H, /*	RLC	(HL)	*/
ODDH, 0CBH, 006H, /*	RLC	(IX+b)	*/
OFDH, 0CBH, 006H, /*	RLC	(IY+b)	*/
OCBH, 007H, 000H, /*	RLC	A	*/
OCBH, 000H, 000H, /*	RLC	B	*/
OCBH, 001H, 000H, /*	RLC	C	*/
OCBH, 002H, 000H, /*	RLC	D	*/
OCBH, 003H, 000H, /*	RLC	E	*/
OCBH, 004H, 000H, /*	RLC	H	*/
OCBH, 005H, 000H, /*	RLC	L	*/
007H, 000H, 000H, /*	RLCA		*/
OEDH, 06FH, 000H, /*	RLD		*/
OCBH, 01EH, 000H, /*	RR	(HL)	*/
ODDH, 0CBH, 01EH, /*	RR	(IX+b)	*/
OFDH, 0CBH, 01EH, /*	RR	(IY+b)	*/
OCBH, 01FH, 000H, /*	RR	A	*/
OCBH, 018H, 000H, /*	RR	B	*/
OCBH, 019H, 000H, /*	RR	C	*/
OCBH, 01AH, 000H, /*	RR	D	*/
OCBH, 01BH, 000H, /*	RR	E	*/
OCBH, 01CH, 000H, /*	RR	H	*/
OCBH, 01DH, 000H, /*	RR	L	*/
01FH, 000H, 000H, /*	RRA		*/
OCBH, 00EH, 000H, /*	RRC	(HL)	*/
ODDH, 0CBH, 00EH, /*	RRC	(IX+b)	*/
OFDH, 0CBH, 00EH, /*	RRC	(IY+b)	*/
OCBH, 00FH, 000H, /*	RRC	A	*/
OCBH, 008H, 000H, /*	RRC	B	*/
OCBH, 009H, 000H, /*	RRC	C	*/
OCBH, 00AH, 000H, /*	RRC	D	*/
OCBH, 00BH, 000H, /*	RRC	E	*/
OCBH, 00CH, 000H, /*	RRC	H	*/
OCBH, 00DH, 000H, /*	RRC	L	*/
00FH, 000H, 000H, /*	RRCA		*/
OEDH, 067H, 000H, /*	RRD		*/



มหาวิทยาลัย
 วิทยาลัย
 วิทยาลัย

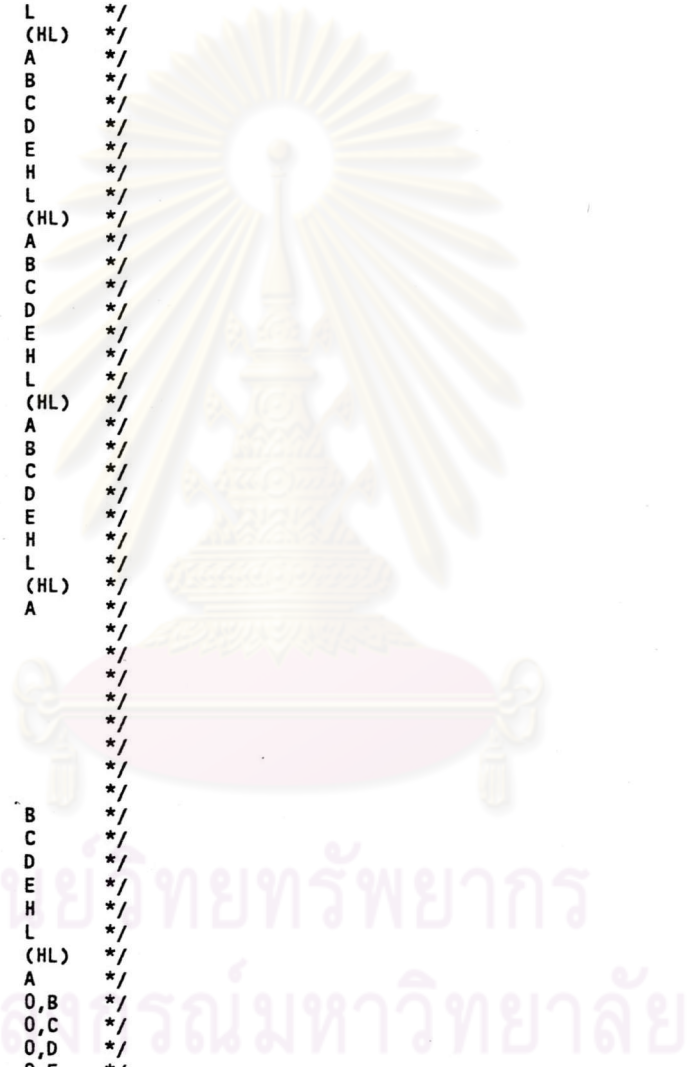
0C7H, 000H, 000H, /*	RST	00	*/
0C7H, 000H, 000H, /*	RST	0	*/
0CFH, 000H, 000H, /*	RST	08	*/
0CFH, 000H, 000H, /*	RST	8	*/
0D7H, 000H, 000H, /*	RST	10	*/
0DFH, 000H, 000H, /*	RST	18	*/
0E7H, 000H, 000H, /*	RST	20	*/
0EFH, 000H, 000H, /*	RST	28	*/
0F7H, 000H, 000H, /*	RST	30	*/
0FFH, 000H, 000H, /*	RST	38	*/
09EH, 000H, 000H, /*	SBC	A, (HL)	*/
0DDH, 09EH, 000H, /*	SBC	A, (IX+b)	*/
0FDH, 09EH, 000H, /*	SBC	A, (IY+b)	*/
09FH, 000H, 000H, /*	SBC	A, A	*/
098H, 000H, 000H, /*	SBC	A, B	*/
099H, 000H, 000H, /*	SBC	A, C	*/
09AH, 000H, 000H, /*	SBC	A, D	*/
09BH, 000H, 000H, /*	SBC	A, E	*/
09CH, 000H, 000H, /*	SBC	A, H	*/
09DH, 000H, 000H, /*	SBC	A, L	*/
0DEH, 000H, 000H, /*	SBC	A, b	*/
0EDH, 042H, 000H, /*	SBC	HL, BC	*/
0EDH, 052H, 000H, /*	SBC	HL, DE	*/
0EDH, 062H, 000H, /*	SBC	HL, HL	*/
0EDH, 072H, 000H, /*	SBC	HL, SP	*/
037H, 000H, 000H, /*	SCF	*/	*/
0CBH, 0C6H, 000H, /*	SET	0, (HL)	*/
0DDH, 0CBH, 0C6H, /*	SET	0, (IX+b)	*/
0FDH, 0CBH, 0C6H, /*	SET	0, (IY+b)	*/
0CBH, 0C7H, 000H, /*	SET	0, A	*/
0CBH, 0C0H, 000H, /*	SET	0, B	*/
0CBH, 0C1H, 000H, /*	SET	0, C	*/
0CBH, 0C2H, 000H, /*	SET	0, D	*/
0CBH, 0C3H, 000H, /*	SET	0, E	*/
0CBH, 0C4H, 000H, /*	SET	0, H	*/
0CBH, 0C5H, 000H, /*	SET	0, L	*/
0CBH, 0CEH, 000H, /*	SET	1, (HL)	*/
0DDH, 0CBH, 0CEH, /*	SET	1, (IX+b)	*/
0FDH, 0CBH, 0CEH, /*	SET	1, (IY+b)	*/
0CBH, 0CFH, 000H, /*	SET	1, A	*/
0CBH, 0C8H, 000H, /*	SET	1, B	*/
0CBH, 0C9H, 000H, /*	SET	1, C	*/
0CBH, 0CAH, 000H, /*	SET	1, D	*/
0CBH, 0CBH, 000H, /*	SET	1, E	*/
0CBH, 0CCH, 000H, /*	SET	1, H	*/
0CBH, 0CDH, 000H, /*	SET	1, L	*/
0CBH, 0D6H, 000H, /*	SET	2, (HL)	*/
0DDH, 0CBH, 0D6H, /*	SET	2, (IX+b)	*/
0FDH, 0CBH, 0D6H, /*	SET	2, (IY+b)	*/
0CBH, 0D7H, 000H, /*	SET	2, A	*/
0CBH, 0D0H, 000H, /*	SET	2, B	*/
0CBH, 0D1H, 000H, /*	SET	2, C	*/
0CBH, 0D2H, 000H, /*	SET	2, D	*/
0CBH, 0D3H, 000H, /*	SET	2, E	*/
0CBH, 0D4H, 000H, /*	SET	2, H	*/
0CBH, 0D5H, 000H, /*	SET	2, L	*/
0CBH, 0DEH, 000H, /*	SET	3, (HL)	*/
0DDH, 0CBH, 0DEH, /*	SET	3, (IX+b)	*/
0FDH, 0CBH, 0DEH, /*	SET	3, (IY+b)	*/
0CBH, 0DFH, 000H, /*	SET	3, A	*/
0CBH, 0D8H, 000H, /*	SET	3, B	*/
0CBH, 0D9H, 000H, /*	SET	3, C	*/
0CBH, 0DAH, 000H, /*	SET	3, D	*/
0CBH, 0DBH, 000H, /*	SET	3, E	*/
0CBH, 0DCH, 000H, /*	SET	3, H	*/
0CBH, 0DDH, 000H, /*	SET	3, L	*/
0CBH, 0E6H, 000H, /*	SET	4, (HL)	*/
0DDH, 0CBH, 0E6H, /*	SET	4, (IX+b)	*/
0FDH, 0CBH, 0E6H, /*	SET	4, (IY+b)	*/
0CBH, 0E7H, 000H, /*	SET	4, A	*/
0CBH, 0E0H, 000H, /*	SET	4, B	*/
0CBH, 0E1H, 000H, /*	SET	4, C	*/
0CBH, 0E2H, 000H, /*	SET	4, D	*/
0CBH, 0E3H, 000H, /*	SET	4, E	*/
0CBH, 0E4H, 000H, /*	SET	4, H	*/
0CBH, 0E5H, 000H, /*	SET	4, L	*/
0CBH, 0EEH, 000H, /*	SET	5, (HL)	*/
0DDH, 0CBH, 0EEH, /*	SET	5, (IX+b)	*/
0FDH, 0CBH, 0EEH, /*	SET	5, (IY+b)	*/
0CBH, 0EFH, 000H, /*	SET	5, A	*/
0CBH, 0E8H, 000H, /*	SET	5, B	*/
0CBH, 0E9H, 000H, /*	SET	5, C	*/
0CBH, 0EAH, 000H, /*	SET	5, D	*/
0CBH, 0EBH, 000H, /*	SET	5, E	*/
0CBH, 0ECH, 000H, /*	SET	5, H	*/
0CBH, 0EDH, 000H, /*	SET	5, L	*/
0CBH, 0F6H, 000H, /*	SET	6, (HL)	*/
0DDH, 0CBH, 0F6H, /*	SET	6, (IX+b)	*/
0FDH, 0CBH, 0F6H, /*	SET	6, (IY+b)	*/
0CBH, 0F7H, 000H, /*	SET	6, A	*/

วิทยาลัยพยาบาล
ศรีนครินทรวิถียาลัย

OCBH, 0F0H, 000H, /*	SET	6,B	*/
OCBH, 0F1H, 000H, /*	SET	6,C	*/
OCBH, 0F2H, 000H, /*	SET	6,D	*/
OCBH, 0F3H, 000H, /*	SET	6,E	*/
OCBH, 0F4H, 000H, /*	SET	6,H	*/
OCBH, 0F5H, 000H, /*	SET	6,L	*/
OCBH, 0FEH, 000H, /*	SET	7,(HL)	*/
ODDH, OCBH, 0FEH, /*	SET	7,(IX+b)	*/
OFDH, OCBH, 0FEH, /*	SET	7,(IY+b)	*/
OCBH, 0FFH, 000H, /*	SET	7,A	*/
OCBH, 0F8H, 000H, /*	SET	7,B	*/
OCBH, 0F9H, 000H, /*	SET	7,C	*/
OCBH, 0FAH, 000H, /*	SET	7,D	*/
OCBH, 0FBH, 000H, /*	SET	7,E	*/
OCBH, 0FCH, 000H, /*	SET	7,H	*/
OCBH, 0FDH, 000H, /*	SET	7,L	*/
OCBH, 026H, 000H, /*	SLA	(HL)	*/
ODDH, OCBH, 026H, /*	SLA	(IX+b)	*/
OFDH, OCBH, 026H, /*	SLA	(IY+b)	*/
OCBH, 027H, 000H, /*	SLA	A	*/
OCBH, 020H, 000H, /*	SLA	B	*/
OCBH, 021H, 000H, /*	SLA	C	*/
OCBH, 022H, 000H, /*	SLA	D	*/
OCBH, 023H, 000H, /*	SLA	E	*/
OCBH, 024H, 000H, /*	SLA	H	*/
OCBH, 025H, 000H, /*	SLA	L	*/
OCBH, 02EH, 000H, /*	SRA	(HL)	*/
ODDH, OCBH, 02EH, /*	SRA	(IX+b)	*/
OFDH, OCBH, 02EH, /*	SRA	(IY+b)	*/
OCBH, 02FH, 000H, /*	SRA	A	*/
OCBH, 028H, 000H, /*	SRA	B	*/
OCBH, 029H, 000H, /*	SRA	C	*/
OCBH, 02AH, 000H, /*	SRA	D	*/
OCBH, 02BH, 000H, /*	SRA	E	*/
OCBH, 02CH, 000H, /*	SRA	H	*/
OCBH, 02DH, 000H, /*	SRA	L	*/
OCBH, 03EH, 000H, /*	SRL	(HL)	*/
ODDH, OCBH, 03EH, /*	SRL	(IX+b)	*/
OFDH, OCBH, 03EH, /*	SRL	(IY+b)	*/
OCBH, 03FH, 000H, /*	SRL	A	*/
OCBH, 038H, 000H, /*	SRL	B	*/
OCBH, 039H, 000H, /*	SRL	C	*/
OCBH, 03AH, 000H, /*	SRL	D	*/
OCBH, 03BH, 000H, /*	SRL	E	*/
OCBH, 03CH, 000H, /*	SRL	H	*/
OCBH, 03DH, 000H, /*	SRL	L	*/
096H, 000H, 000H, /*	SUB	(HL)	*/
ODDH, 096H, 000H, /*	SUB	(IX+b)	*/
OFDH, 096H, 000H, /*	SUB	(IY+b)	*/
097H, 000H, 000H, /*	SUB	A	*/
090H, 000H, 000H, /*	SUB	B	*/
091H, 000H, 000H, /*	SUB	C	*/
092H, 000H, 000H, /*	SUB	D	*/
093H, 000H, 000H, /*	SUB	E	*/
094H, 000H, 000H, /*	SUB	H	*/
095H, 000H, 000H, /*	SUB	L	*/
0D6H, 000H, 000H, /*	SUB	b	*/
0AEH, 000H, 000H, /*	XOR	(HL)	*/
ODDH, 0AEH, 000H, /*	XOR	(IX+b)	*/
OFDH, 0AEH, 000H, /*	XOR	(IY+b)	*/
0AFH, 000H, 000H, /*	XOR	A	*/
0ABH, 000H, 000H, /*	XOR	B	*/
0A9H, 000H, 000H, /*	XOR	C	*/
0AAH, 000H, 000H, /*	XOR	D	*/
0ABH, 000H, 000H, /*	XOR	E	*/
0ACH, 000H, 000H, /*	XOR	H	*/
0ADH, 000H, 000H, /*	XOR	L	*/
0EEH, 000H, 000H, /*	XOR	b	*/

วิทยาลัยพยาบาล
 วิทยาลัยพยาบาล
 วิทยาลัยพยาบาล

510,/*	CB 00	RLC B	*/
511,/*	CB 01	RLC C	*/
512,/*	CB 02	RLC D	*/
513,/*	CB 03	RLC E	*/
514,/*	CB 04	RLC H	*/
515,/*	CB 05	RLC L	*/
506,/*	CB 06	RLC (HL)	*/
509,/*	CB 07	RLC A	*/
533,/*	CB 08	RRC B	*/
534,/*	CB 09	RRC C	*/
535,/*	CB 0A	RRC D	*/
536,/*	CB 0B	RRC E	*/
537,/*	CB 0C	RRC H	*/
538,/*	CB 0D	RRC L	*/
529,/*	CB 0E	RRC (HL)	*/
532,/*	CB 0F	RRC A	*/
499,/*	CB 10	RL B	*/
500,/*	CB 11	RL C	*/
501,/*	CB 12	RL D	*/
502,/*	CB 13	RL E	*/
503,/*	CB 14	RL H	*/
504,/*	CB 15	RL L	*/
495,/*	CB 16	RL (HL)	*/
498,/*	CB 17	RL A	*/
522,/*	CB 18	RR B	*/
523,/*	CB 19	RR C	*/
524,/*	CB 1A	RR D	*/
525,/*	CB 1B	RR E	*/
526,/*	CB 1C	RR H	*/
527,/*	CB 1D	RR L	*/
518,/*	CB 1E	RR (HL)	*/
521,/*	CB 1F	RR A	*/
651,/*	CB 20	SLA B	*/
652,/*	CB 21	SLA C	*/
653,/*	CB 22	SLA D	*/
654,/*	CB 23	SLA E	*/
655,/*	CB 24	SLA H	*/
656,/*	CB 25	SLA L	*/
647,/*	CB 26	SLA (HL)	*/
650,/*	CB 27	SLA A	*/
661,/*	CB 28	SRA B	*/
662,/*	CB 29	SRA C	*/
663,/*	CB 2A	SRA D	*/
664,/*	CB 2B	SRA E	*/
665,/*	CB 2C	SRA H	*/
666,/*	CB 2D	SRA L	*/
657,/*	CB 2E	SRA (HL)	*/
660,/*	CB 2F	SRA A	*/
0,/*	CB 30		*/
0,/*	CB 31		*/
0,/*	CB 32		*/
0,/*	CB 33		*/
0,/*	CB 34		*/
0,/*	CB 35		*/
0,/*	CB 36		*/
0,/*	CB 37		*/
671,/*	CB 38	SRL B	*/
672,/*	CB 39	SRL C	*/
673,/*	CB 3A	SRL D	*/
674,/*	CB 3B	SRL E	*/
675,/*	CB 3C	SRL H	*/
676,/*	CB 3D	SRL L	*/
667,/*	CB 3E	SRL (HL)	*/
670,/*	CB 3F	SRL A	*/
54,/*	CB 40	BIT 0,B	*/
55,/*	CB 41	BIT 0,C	*/
56,/*	CB 42	BIT 0,D	*/
57,/*	CB 43	BIT 0,E	*/
58,/*	CB 44	BIT 0,H	*/
59,/*	CB 45	BIT 0,L	*/
50,/*	CB 46	BIT 0,(HL)	*/
53,/*	CB 47	BIT 0,A	*/
64,/*	CB 48	BIT 1,B	*/
65,/*	CB 49	BIT 1,C	*/
66,/*	CB 4A	BIT 1,D	*/
67,/*	CB 4B	BIT 1,E	*/
68,/*	CB 4C	BIT 1,H	*/
69,/*	CB 4D	BIT 1,L	*/
60,/*	CB 4E	BIT 1,(HL)	*/
63,/*	CB 4F	BIT 1,A	*/
74,/*	CB 50	BIT 2,B	*/
75,/*	CB 51	BIT 2,C	*/
76,/*	CB 52	BIT 2,D	*/
77,/*	CB 53	BIT 2,E	*/
78,/*	CB 54	BIT 2,H	*/
79,/*	CB 55	BIT 2,L	*/
70,/*	CB 56	BIT 2,(HL)	*/
73,/*	CB 57	BIT 2,A	*/
84,/*	CB 58	BIT 3,B	*/
85,/*	CB 59	BIT 3,C	*/



86,/*	CB 5A	BIT	3,D	*/
87,/*	CB 5B	BIT	3,E	*/
88,/*	CB 5C	BIT	3,H	*/
89,/*	CB 5D	BIT	3,L	*/
80,/*	CB 5E	BIT	3,(HL)	*/
83,/*	CB 5F	BIT	3,A	*/
94,/*	CB 60	BIT	4,B	*/
95,/*	CB 61	BIT	4,C	*/
96,/*	CB 62	BIT	4,D	*/
97,/*	CB 63	BIT	4,E	*/
98,/*	CB 64	BIT	4,H	*/
99,/*	CB 65	BIT	4,L	*/
90,/*	CB 66	BIT	4,(HL)	*/
93,/*	CB 67	BIT	4,A	*/
104,/*	CB 68	BIT	5,B	*/
105,/*	CB 69	BIT	5,C	*/
106,/*	CB 6A	BIT	5,D	*/
107,/*	CB 6B	BIT	5,E	*/
108,/*	CB 6C	BIT	5,H	*/
109,/*	CB 6D	BIT	5,L	*/
100,/*	CB 6E	BIT	5,(HL)	*/
103,/*	CB 6F	BIT	5,A	*/
114,/*	CB 70	BIT	6,B	*/
115,/*	CB 71	BIT	6,C	*/
116,/*	CB 72	BIT	6,D	*/
117,/*	CB 73	BIT	6,E	*/
118,/*	CB 74	BIT	6,H	*/
119,/*	CB 75	BIT	6,L	*/
110,/*	CB 76	BIT	6,(HL)	*/
113,/*	CB 77	BIT	6,A	*/
124,/*	CB 78	BIT	7,B	*/
125,/*	CB 79	BIT	7,C	*/
126,/*	CB 7A	BIT	7,D	*/
127,/*	CB 7B	BIT	7,E	*/
128,/*	CB 7C	BIT	7,H	*/
129,/*	CB 7D	BIT	7,L	*/
120,/*	CB 7E	BIT	7,(HL)	*/
123,/*	CB 7F	BIT	7,A	*/
408,/*	CB 80	RES	0,B	*/
409,/*	CB 81	RES	0,C	*/
410,/*	CB 82	RES	0,D	*/
411,/*	CB 83	RES	0,E	*/
412,/*	CB 84	RES	0,H	*/
413,/*	CB 85	RES	0,L	*/
404,/*	CB 86	RES	0,(HL)	*/
407,/*	CB 87	RES	0,A	*/
418,/*	CB 88	RES	1,B	*/
419,/*	CB 89	RES	1,C	*/
420,/*	CB 8A	RES	1,D	*/
421,/*	CB 8B	RES	1,E	*/
422,/*	CB 8C	RES	1,H	*/
423,/*	CB 8D	RES	1,L	*/
414,/*	CB 8E	RES	1,(HL)	*/
417,/*	CB 8F	RES	1,A	*/
428,/*	CB 90	RES	2,B	*/
429,/*	CB 91	RES	2,C	*/
430,/*	CB 92	RES	2,D	*/
431,/*	CB 93	RES	2,E	*/
432,/*	CB 94	RES	2,H	*/
433,/*	CB 95	RES	2,L	*/
424,/*	CB 96	RES	2,(HL)	*/
427,/*	CB 97	RES	2,A	*/
438,/*	CB 98	RES	3,B	*/
439,/*	CB 99	RES	3,C	*/
440,/*	CB 9A	RES	3,D	*/
441,/*	CB 9B	RES	3,E	*/
442,/*	CB 9C	RES	3,H	*/
443,/*	CB 9D	RES	3,L	*/
434,/*	CB 9E	RES	3,(HL)	*/
437,/*	CB 9F	RES	3,A	*/
448,/*	CB A0	RES	4,B	*/
449,/*	CB A1	RES	4,C	*/
450,/*	CB A2	RES	4,D	*/
451,/*	CB A3	RES	4,E	*/
452,/*	CB A4	RES	4,H	*/
453,/*	CB A5	RES	4,L	*/
444,/*	CB A6	RES	4,(HL)	*/
447,/*	CB A7	RES	4,A	*/
458,/*	CB A8	RES	5,B	*/
459,/*	CB A9	RES	5,C	*/
460,/*	CB AA	RES	5,D	*/
461,/*	CB AB	RES	5,E	*/
462,/*	CB AC	RES	5,H	*/
463,/*	CB AD	RES	5,L	*/
454,/*	CB AE	RES	5,(HL)	*/
457,/*	CB AF	RES	5,A	*/
468,/*	CB B0	RES	6,B	*/
469,/*	CB B1	RES	6,C	*/
470,/*	CB B2	RES	6,D	*/
471,/*	CB B3	RES	6,E	*/

472,/*	CB B4	RES	6,H	*/
473,/*	CB B5	RES	6,L	*/
464,/*	CB B6	RES	6,(HL)	*/
467,/*	CB B7	RES	6,A	*/
478,/*	CB B8	RES	7,B	*/
479,/*	CB B9	RES	7,C	*/
480,/*	CB BA	RES	7,D	*/
481,/*	CB BB	RES	7,E	*/
482,/*	CB BC	RES	7,H	*/
483,/*	CB BD	RES	7,L	*/
474,/*	CB BE	RES	7,(HL)	*/
477,/*	CB BF	RES	7,A	*/
571,/*	CB C0	SET	0,B	*/
572,/*	CB C1	SET	0,C	*/
573,/*	CB C2	SET	0,D	*/
574,/*	CB C3	SET	0,E	*/
575,/*	CB C4	SET	0,H	*/
576,/*	CB C5	SET	0,L	*/
567,/*	CB C6	SET	0,(HL)	*/
570,/*	CB C7	SET	0,A	*/
581,/*	CB C8	SET	1,B	*/
582,/*	CB C9	SET	1,C	*/
583,/*	CB CA	SET	1,D	*/
584,/*	CB CB	SET	1,E	*/
585,/*	CB CC	SET	1,H	*/
586,/*	CB CD	SET	1,L	*/
577,/*	CB CE	SET	1,(HL)	*/
580,/*	CB CF	SET	1,A	*/
591,/*	CB D0	SET	2,B	*/
592,/*	CB D1	SET	2,C	*/
593,/*	CB D2	SET	2,D	*/
594,/*	CB D3	SET	2,E	*/
595,/*	CB D4	SET	2,H	*/
596,/*	CB D5	SET	2,L	*/
587,/*	CB D6	SET	2,(HL)	*/
590,/*	CB D7	SET	2,A	*/
601,/*	CB D8	SET	3,B	*/
602,/*	CB D9	SET	3,C	*/
603,/*	CB DA	SET	3,D	*/
604,/*	CB DB	SET	3,E	*/
605,/*	CB DC	SET	3,H	*/
606,/*	CB DD	SET	3,L	*/
597,/*	CB DE	SET	3,(HL)	*/
600,/*	CB DF	SET	3,A	*/
611,/*	CB E0	SET	4,B	*/
612,/*	CB E1	SET	4,C	*/
613,/*	CB E2	SET	4,D	*/
614,/*	CB E3	SET	4,E	*/
615,/*	CB E4	SET	4,H	*/
616,/*	CB E5	SET	4,L	*/
607,/*	CB E6	SET	4,(HL)	*/
610,/*	CB E7	SET	4,A	*/
621,/*	CB E8	SET	5,B	*/
622,/*	CB E9	SET	5,C	*/
623,/*	CB EA	SET	5,D	*/
624,/*	CB EB	SET	5,E	*/
625,/*	CB EC	SET	5,H	*/
626,/*	CB ED	SET	5,L	*/
617,/*	CB EE	SET	5,(HL)	*/
620,/*	CB EF	SET	5,A	*/
631,/*	CB F0	SET	6,B	*/
632,/*	CB F1	SET	6,C	*/
633,/*	CB F2	SET	6,D	*/
634,/*	CB F3	SET	6,E	*/
635,/*	CB F4	SET	6,H	*/
636,/*	CB F5	SET	6,L	*/
627,/*	CB F6	SET	6,(HL)	*/
630,/*	CB F7	SET	6,A	*/
641,/*	CB F8	SET	7,B	*/
642,/*	CB F9	SET	7,C	*/
643,/*	CB FA	SET	7,D	*/
644,/*	CB FB	SET	7,E	*/
645,/*	CB FC	SET	7,H	*/
646,/*	CB FD	SET	7,L	*/
637,/*	CB FE	SET	7,(HL)	*/
640,/*	CB FF	SET	7,A	*/

วิทยาลัยพยาบาล
รณมหาวิทยาลัย

507,/*	DD	CB	00	*/	006H,/*	RLC	(IX+00)	*/
530,/*	DD	CB	00	*/	00EH,/*	RRC	(IX+00)	*/
496,/*	DD	CB	00	*/	016H,/*	RL	(IX+00)	*/
519,/*	DD	CB	00	*/	01EH,/*	RR	(IX+00)	*/
648,/*	DD	CB	00	*/	026H,/*	SLA	(IX+00)	*/
658,/*	DD	CB	00	*/	02EH,/*	SRA	(IX+00)	*/
668,/*	DD	CB	00	*/	03EH,/*	SRL	(IX+00)	*/
51,/*	DD	CB	00	*/	046H,/*	BIT	0,(IX+00)	*/
61,/*	DD	CB	00	*/	04EH,/*	BIT	1,(IX+00)	*/
71,/*	DD	CB	00	*/	056H,/*	BIT	2,(IX+00)	*/
81,/*	DD	CB	00	*/	05EH,/*	BIT	3,(IX+00)	*/
91,/*	DD	CB	00	*/	066H,/*	BIT	4,(IX+00)	*/
101,/*	DD	CB	00	*/	06EH,/*	BIT	5,(IX+00)	*/
111,/*	DD	CB	00	*/	076H,/*	BIT	6,(IX+00)	*/
121,/*	DD	CB	00	*/	07EH,/*	BIT	7,(IX+00)	*/
405,/*	DD	CB	00	*/	086H,/*	RES	0,(IX+00)	*/
415,/*	DD	CB	00	*/	08EH,/*	RES	1,(IX+00)	*/
425,/*	DD	CB	00	*/	096H,/*	RES	2,(IX+00)	*/
435,/*	DD	CB	00	*/	09EH,/*	RES	3,(IX+00)	*/
445,/*	DD	CB	00	*/	0A6H,/*	RES	4,(IX+00)	*/
455,/*	DD	CB	00	*/	0AEH,/*	RES	5,(IX+00)	*/
465,/*	DD	CB	00	*/	0B6H,/*	RES	6,(IX+00)	*/
475,/*	DD	CB	00	*/	0BEH,/*	RES	7,(IX+00)	*/
568,/*	DD	CB	00	*/	0C6H,/*	SET	0,(IX+00)	*/
578,/*	DD	CB	00	*/	0CEH,/*	SET	1,(IX+00)	*/
588,/*	DD	CB	00	*/	0D6H,/*	SET	2,(IX+00)	*/
598,/*	DD	CB	00	*/	0DEH,/*	SET	3,(IX+00)	*/
608,/*	DD	CB	00	*/	0E6H,/*	SET	4,(IX+00)	*/
618,/*	DD	CB	00	*/	0EEH,/*	SET	5,(IX+00)	*/
628,/*	DD	CB	00	*/	0F6H,/*	SET	6,(IX+00)	*/
638,/*	DD	CB	00	*/	0FEH,/*	SET	7,(IX+00)	*/



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

31,/*	DD */	009H,/*	ADD	IX,BC	*/
32,/*	DD */	019H,/*	ADD	IX,DE	*/
343,/*	DD */	021H,/*	LD	IX,0000	*/
261,/*	DD */	022H,/*	LD	(0000),IX	*/
206,/*	DD */	023H,/*	INC	IX	*/
33,/*	DD */	029H,/*	ADD	IX,IX	*/
342,/*	DD */	02AH,/*	LD	IX,(0000)	*/
169,/*	DD */	02BH,/*	DEC	IX	*/
195,/*	DD */	034H,/*	INC	(IX+00)	*/
158,/*	DD */	035H,/*	DEC	(IX+00)	*/
248,/*	DD */	036H,/*	LD	(IX+00),00	*/
34,/*	DD */	039H,/*	ADD	IX,SP	*/
281,/*	DD */	046H,/*	LD	B,(IX+00)	*/
294,/*	DD */	04EH,/*	LD	C,(IX+00)	*/
305,/*	DD */	056H,/*	LD	D,(IX+00)	*/
318,/*	DD */	05EH,/*	LD	E,(IX+00)	*/
329,/*	DD */	066H,/*	LD	H,(IX+00)	*/
347,/*	DD */	06EH,/*	LD	L,(IX+00)	*/
242,/*	DD */	070H,/*	LD	(IX+00),B	*/
243,/*	DD */	071H,/*	LD	(IX+00),C	*/
244,/*	DD */	072H,/*	LD	(IX+00),D	*/
245,/*	DD */	073H,/*	LD	(IX+00),E	*/
246,/*	DD */	074H,/*	LD	(IX+00),H	*/
247,/*	DD */	075H,/*	LD	(IX+00),L	*/
241,/*	DD */	077H,/*	LD	(IX+00),A	*/
267,/*	DD */	07EH,/*	LD	A,(IX+00)	*/
17,/*	DD */	086H,/*	ADD	A,(IX+00)	*/
2,/*	DD */	08EH,/*	ADC	A,(IX+00)	*/
678,/*	DD */	096H,/*	SUB	(IX+00)	*/
552,/*	DD */	09EH,/*	SBC	A,(IX+00)	*/
40,/*	DD */	0A6H,/*	AND	(IX+00)	*/
689,/*	DD */	0AEH,/*	XOR	(IX+00)	*/
370,/*	DD */	0B6H,/*	OR	(IX+00)	*/
141,/*	DD */	0BEH,/*	CP	(IX+00)	*/
0,/*	DD */	0CBH,/*			*/
396,/*	DD */	0E1H,/*	POP	IX	*/
177,/*	DD */	0E3H,/*	EX	(SP),IX	*/
402,/*	DD */	0E5H,/*	PUSH	IX	*/
215,/*	DD */	0E9H,/*	JP	(IX)	*/
360,/*	DD */	0F9H,/*	LD	SP,IX	*/

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

188,/*	ED /*	040H,/*	IN	B,(C)	*/
383,/*	ED /*	041H,/*	OUT	(C),B	*/
562,/*	ED /*	042H,/*	SBC	HL,BC	*/
258,/*	ED /*	043H,/*	LD	(0000),BC	*/
367,/*	ED /*	044H,/*	NEG		*/
494,/*	ED /*	045H,/*	RETN		*/
183,/*	ED /*	046H,/*	IM	0	*/
341,/*	ED /*	047H,/*	LD	I,A	*/
189,/*	ED /*	048H,/*	IN	C,(C)	*/
384,/*	ED /*	049H,/*	OUT	(C),C	*/
12,/*	ED /*	04AH,/*	ADC	HL,BC	*/
291,/*	ED /*	04BH,/*	LD	BC,(0000)	*/
493,/*	ED /*	04DH,/*	RETI		*/
357,/*	ED /*	04FH,/*	LD	R,A	*/
190,/*	ED /*	050H,/*	IN	D,(C)	*/
385,/*	ED /*	051H,/*	OUT	(C),D	*/
563,/*	ED /*	052H,/*	SBC	HL,DE	*/
259,/*	ED /*	053H,/*	LD	(0000),DE	*/
184,/*	ED /*	056H,/*	IM	1	*/
276,/*	ED /*	057H,/*	LD	A,I	*/
191,/*	ED /*	058H,/*	IN	E,(C)	*/
386,/*	ED /*	059H,/*	OUT	(C),E	*/
13,/*	ED /*	05AH,/*	ADC	HL,DE	*/
315,/*	ED /*	05BH,/*	LD	DE,(0000)	*/
185,/*	ED /*	05EH,/*	IM	2	*/
278,/*	ED /*	05FH,/*	LD	A,R	*/
192,/*	ED /*	060H,/*	IN	H,(C)	*/
387,/*	ED /*	061H,/*	OUT	(C),H	*/
564,/*	ED /*	062H,/*	SBC	HL,HL	*/
540,/*	ED /*	067H,/*	RRD		*/
193,/*	ED /*	068H,/*	IN	L,(C)	*/
388,/*	ED /*	069H,/*	OUT	(C),L	*/
14,/*	ED /*	06AH,/*	ADC	HL,HL	*/
517,/*	ED /*	06FH,/*	RLD		*/
565,/*	ED /*	072H,/*	SBC	HL,SP	*/
263,/*	ED /*	073H,/*	LD	(0000),SP	*/
186,/*	ED /*	078H,/*	IN	A,(C)	*/
382,/*	ED /*	079H,/*	OUT	(C),A	*/
15,/*	ED /*	07AH,/*	ADC	HL,SP	*/
358,/*	ED /*	07BH,/*	LD	SP,(0000)	*/
365,/*	ED /*	0A0H,/*	LDI		*/
153,/*	ED /*	0A1H,/*	CPI		*/
212,/*	ED /*	0A2H,/*	INI		*/
391,/*	ED /*	0A3H,/*	OUTI		*/
363,/*	ED /*	0A8H,/*	LDD		*/
151,/*	ED /*	0A9H,/*	CPD		*/
210,/*	ED /*	0AAH,/*	IND		*/
390,/*	ED /*	0ABH,/*	OUTD		*/
366,/*	ED /*	0B0H,/*	LDIR		*/
154,/*	ED /*	0B1H,/*	CPIR		*/
213,/*	ED /*	0B2H,/*	INIR		*/
381,/*	ED /*	0B3H,/*	OTIR		*/
364,/*	ED /*	0B8H,/*	LDDR		*/
152,/*	ED /*	0B9H,/*	CPDR		*/
211,/*	ED /*	0BAH,/*	INDR		*/
380,/*	ED /*	0BBH,/*	OTDR		*/

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

35,/*	FD */	009H,/*	ADD	IY,BC	*/
36,/*	FD */	019H,/*	ADD	IY,DE	*/
345,/*	FD */	021H,/*	LD	IY,0000	*/
262,/*	FD */	022H,/*	LD	(0000),IY	*/
207,/*	FD */	023H,/*	INC	IY	*/
37,/*	FD */	029H,/*	ADD	IY,IY	*/
344,/*	FD */	02AH,/*	LD	IY,(0000)	*/
170,/*	FD */	02BH,/*	DEC	IY	*/
196,/*	FD */	034H,/*	INC	(IY+00)	*/
159,/*	FD */	035H,/*	DEC	(IY+00)	*/
256,/*	FD */	036H,/*	LD	(IY+00),00	*/
38,/*	FD */	039H,/*	ADD	IY,SP	*/
282,/*	FD */	046H,/*	LD	B,(IY+00)	*/
295,/*	FD */	04EH,/*	LD	C,(IY+00)	*/
306,/*	FD */	056H,/*	LD	D,(IY+00)	*/
319,/*	FD */	05EH,/*	LD	E,(IY+00)	*/
330,/*	FD */	066H,/*	LD	H,(IY+00)	*/
348,/*	FD */	06EH,/*	LD	L,(IY+00)	*/
250,/*	FD */	070H,/*	LD	(IY+00),B	*/
251,/*	FD */	071H,/*	LD	(IY+00),C	*/
252,/*	FD */	072H,/*	LD	(IY+00),D	*/
253,/*	FD */	073H,/*	LD	(IY+00),E	*/
254,/*	FD */	074H,/*	LD	(IY+00),H	*/
255,/*	FD */	075H,/*	LD	(IY+00),L	*/
249,/*	FD */	077H,/*	LD	(IY+00),A	*/
268,/*	FD */	07EH,/*	LD	A,(IY+00)	*/
18,/*	FD */	086H,/*	ADD	A,(IY+00)	*/
3,/*	FD */	08EH,/*	ADC	A,(IY+00)	*/
679,/*	FD */	096H,/*	SUB	(IY+00)	*/
553,/*	FD */	09EH,/*	SBC	A,(IY+00)	*/
41,/*	FD */	0A6H,/*	AND	(IY+00)	*/
690,/*	FD */	0AEH,/*	XOR	(IY+00)	*/
371,/*	FD */	0B6H,/*	OR	(IY+00)	*/
142,/*	FD */	0BEH,/*	CP	(IY+00)	*/
0,/*	FD */	0CBH,/*			*/
397,/*	FD */	0E1H,/*	POP	IY	*/
178,/*	FD */	0E3H,/*	EX	(SP),IY	*/
403,/*	FD */	0E5H,/*	PUSH	IY	*/
216,/*	FD */	0E9H,/*	JP	(IY)	*/
361,/*	FD */	0F9H,/*	LD	SP,IY	*/

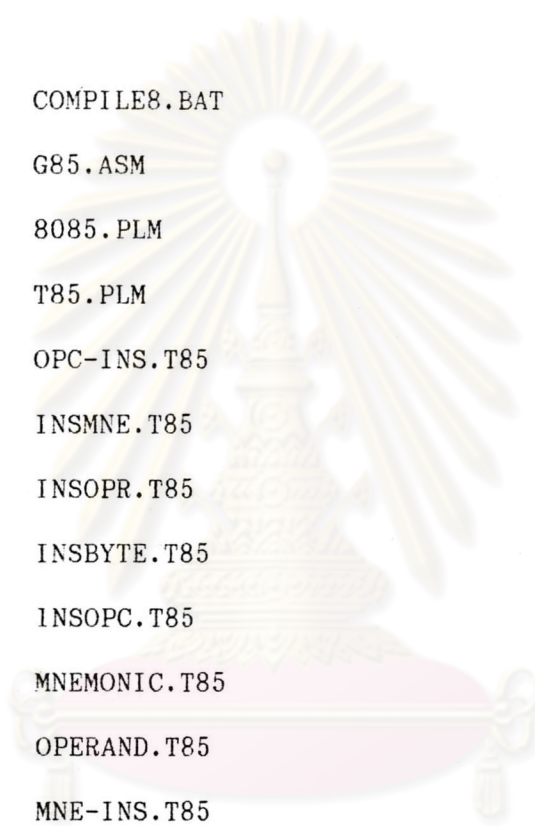
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

508,/*	FD CB 00 */	006H,/*	RLC	(IY+00)	*/
531,/*	FD CB 00 */	00EH,/*	RRC	(IY+00)	*/
497,/*	FD CB 00 */	016H,/*	RL	(IY+00)	*/
520,/*	FD CB 00 */	01EH,/*	RR	(IY+00)	*/
649,/*	FD CB 00 */	026H,/*	SLA	(IY+00)	*/
659,/*	FD CB 00 */	02EH,/*	SRA	(IY+00)	*/
669,/*	FD CB 00 */	03EH,/*	SRL	(IY+00)	*/
52,/*	FD CB 00 */	046H,/*	BIT	0,(IY+00)	*/
62,/*	FD CB 00 */	04EH,/*	BIT	1,(IY+00)	*/
72,/*	FD CB 00 */	056H,/*	BIT	2,(IY+00)	*/
82,/*	FD CB 00 */	05EH,/*	BIT	3,(IY+00)	*/
92,/*	FD CB 00 */	066H,/*	BIT	4,(IY+00)	*/
102,/*	FD CB 00 */	06EH,/*	BIT	5,(IY+00)	*/
112,/*	FD CB 00 */	076H,/*	BIT	6,(IY+00)	*/
122,/*	FD CB 00 */	07EH,/*	BIT	7,(IY+00)	*/
406,/*	FD CB 00 */	086H,/*	RES	0,(IY+00)	*/
416,/*	FD CB 00 */	08EH,/*	RES	1,(IY+00)	*/
426,/*	FD CB 00 */	096H,/*	RES	2,(IY+00)	*/
436,/*	FD CB 00 */	09EH,/*	RES	3,(IY+00)	*/
446,/*	FD CB 00 */	0A6H,/*	RES	4,(IY+00)	*/
456,/*	FD CB 00 */	0AEH,/*	RES	5,(IY+00)	*/
466,/*	FD CB 00 */	0B6H,/*	RES	6,(IY+00)	*/
476,/*	FD CB 00 */	0BEH,/*	RES	7,(IY+00)	*/
569,/*	FD CB 00 */	0C6H,/*	SET	0,(IY+00)	*/
579,/*	FD CB 00 */	0CEH,/*	SET	1,(IY+00)	*/
589,/*	FD CB 00 */	0D6H,/*	SET	2,(IY+00)	*/
599,/*	FD CB 00 */	0DEH,/*	SET	3,(IY+00)	*/
609,/*	FD CB 00 */	0E6H,/*	SET	4,(IY+00)	*/
619,/*	FD CB 00 */	0EEH,/*	SET	5,(IY+00)	*/
629,/*	FD CB 00 */	0F6H,/*	SET	6,(IY+00)	*/
639,/*	FD CB 00 */	0FEH,/*	SET	7,(IY+00)	*/



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

Listing Program of ROM monitor 8085



COMPILE8.BAT

G85.ASM

8085.PLM

T85.PLM

OPC-INS.T85

INSMNE.T85

INSOPR.T85

INSBYTE.T85

INSOPC.T85

MNEMONIC.T85

OPERAND.T85

MNE-INS.T85

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
80as g85  
80pc -J 8085 t85  
80link g85.obj 8085.obj t85.obj  
80loc -c -m r85.map -l0 -scode -l08700h -sdata -l087e0h -sstack=20h  
80map -sa d.out > r85.smb  
80hex d.out > r85.hex
```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย


```

PAGESW EQU 8403h
MONISP EQU 8800H
DATA8251 EQU 0c000H
PORT8255C EQU 0e002H
POP_H EQU 0E1H
JMP_ EQU 0C3H
PUSH_AF EQU 0F5H
LDAX_ EQU 00Ah
RET_ EQU 0C9h
STAX_ EQU 002h
EI_ EQU 0fbh
DI_ EQU 0f3h

```

```

EXTRN init,prompt2,allregister
EXTRN reg
; DECLARE reg(7) ADDRESS
; reg: DS 7*2
;PC: reg(0) = reg+0
;AF: reg(1) = reg+2
; F: LOW(reg(1)) = reg+2
; A: HIGH(reg(1)) = reg+3
;BC: reg(2) = reg+4
;DE: reg(3) = reg+6
;HL: reg(4) = reg+8
; I: reg(5) = reg+10
;IE: LOW(reg(5)) = reg+10
;IM: HIGH(reg(5)) = reg+11
;SP: reg(6) = reg+12

;public procedure
PUBLIC gopc,peek,poke

CSEG
NOP
NOP
NOP
NOP
JMP init ;goto start main and set communication

ORG 0024H ;monitor force TRAP for break
JMP brk ;so goto save register value

ORG 002ch ;RST may be found and make program confuse
JMP init ;so goto main and set communication again
ORG 0034h
JMP init
ORG 0038H
JMP init
ORG 003ch
JMP init

;subroutine for GO command
gopc: MVI A,JMP_ ;restore PC
STA pagesw - 2
LHLD reg+0
SHLD pagesw - 1

;restore all register values except IM
;because IM set after user change register

LDA reg+10 ;restore IE
RAR ;IF IE=1 THEN ENABLE
JNC DISABLE ;ELSE DISABLE
MVI A,EI_
JMP restore
DISABLE: MVI A,DI_
restore: STA pagesw - 3

LHLD reg+2 ;restore AF
PUSH H
POP PSW

LHLD reg+12 ;restore SP
SPHL

LHLD reg+4 ;restore BC
MOV C,L
MOV B,H

LHLD reg+6 ;restore DE
MOV E,L
MOV D,H

LHLD reg+8 ;restore HL

JMP pagesw - 3
;pagesw-3 EI/DI
;pagesw-2 JMP
;pagesw-1 low(PC)

```

;pagesw high(PC)

257

;subroutine of TRAP:save register and jump to main (at prompt2)

```
brk: SHLD reg+8 ;SAVE HL

      XCHG ;SAVE DE
      SHLD reg+6

      MOV H,B ;SAVE BC
      MOV L,C
      SHLD reg+4

      STA reg+3 ;SAVE A
                ;If hit any IBM key ,when run user
                ;it will make a break then RxRDY active
                ;until CPU read 8251 dataport
      LDA DATA8251 ;DISABLE RxRDY
      MVI A,00000011B ;DISABLE /G08255,/EN688
      STA PORT8255C ;PORT C OF 8255

      LXI H,pagesw ;SAVE PC
      MVI A,POP_H ;BY POP H :HL <- (SP) :HL = PC
      MOV M,A
      INX H
      MVI A,JMP_
      MOV M,A
      INX H
      MVI A,LOW(LABEL1)
      MOV M,A
      INX H
      MVI A,HIGH(LABEL1)
      MOV M,A
      JMP pagesw
;FETCH OP CODE FROM RAM_MONI
;2403 POP H ;READ PC FROM STACK_USER
; JMP LABEL1 ;CONTINUE FETCH OP CODE FROM ROM_MONI
LABEL1: SHLD reg+0

      LXI H,pagesw ;SAVE F
      MVI A,PUSH_AF ;BY PUSH PSW :(SP) <- AF
      MOV M,A ; (SP) = AF
      INX H
      MVI A,JMP_
      MOV M,A
      INX H
      MVI A,LOW(LABEL2)
      MOV M,A
      INX H
      MVI A,HIGH(LABEL2)
      MOV M,A
      JMP pagesw
;FETCH OP CODE FROM RAM_MONI
;2403 PUSH PSW ;WRITE AF TO STACK_USER
; JMP LABEL2 ;CONTINUE FETCH OP CODE FROM ROM_MONI
LABEL2: LXI H,pagesw ;THEN POP H :HL <- (SP)
      MVI A,POP_H ; HL = AF
      MOV M,A
      INX H
      MVI A,JMP_
      MOV M,A
      INX H
      MVI A,LOW(LABEL3)
      MOV M,A
      INX H
      MVI A,HIGH(LABEL3)
      MOV M,A
      JMP pagesw
;FETCH OP CODE FROM RAM_MONI
;2403 POP H ;READ AF FROM STACK_USER
; JMP LABEL2 ;CONTINUE FETCH OP CODE FROM ROM_MONI
LABEL3: MOV A,L ;SAVE F
      STA reg+2

      LXI H,0000H ;SAVE SP
      DAD SP ;HL = SP
      SHLD reg+12

      LXI SP,MONISP ;INIT MONITOR STACK

;save user IM after break
      RIM ;read interrupt mask
      STA reg+11 ;SAVE IM
      ANI 00001000b ;bit(IM,4) = IE
      JZ saveDI
      MVI A,1
      JMP saveIE
saveDI: XRA A
saveIE: STA reg+10

      CALL allregister
```

```

;FUNCTION peek
;PL/M call this routine as
;PROCEDURE(ADDR$,) BYTE EXTERNAL
;first parameter from PL/M sent by BC or C
;second parameter from PL/M sent by DE or E
;third or more parameter sent by stack
;e.g. CONTENT = peek(OPERAND(0));
;      BC = OPERAND(0)
;it return parameter by HL or A
;other registers may be changed

peek:   LXI H,pagesw
        MVI M,LDAX_ ;2403 0A LD A,(BC)
        INX H
        MVI M,RET_ ;2404 C9 RET
        JMP pagesw
;FETCH OPCODE FROM RAM_MONI
;2403 LDAX B ;READ CONTENT FROM MEMORY_USER (addr.XXYY)
;      RET ;CONTINUE FETCH OPCODE FROM ROM_MONI
    
```

```

;FUNCTION poke
;PL/M call this routine as
;PROCEDURE(ADDR$,CONTENT) EXTERNAL
;e.g. CALL poke(OPERAND(0),OPERAND(1));
;      BC = OPERAND(0)
;      E = OPERAND(1)
;other registers may be changed

poke:   LXI H,pagesw
        MVI M,STAX_ ;2403 02 LD (BC),A
        INX H
        MVI M,RET_ ;2404 C9 RET
        MOV A,E
        JMP pagesw
;FETCH OPCODE FROM RAM_MONI
;2403 STAX B ;WRITE CONTENT TO MEMORY_USER (addr.XXYY)
;      RET ;CONTINUE FETCH OPCODE FROM ROM_MONI

        END
    
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

main8085:DO;
DECLARE /* symbol */
NUL      LITERALLY '00h',/* `a */
ETX      LITERALLY '03h',/* `C */
ACK      LITERALLY '06h',/* `F */
BEL      LITERALLY '07h',/* `G */
BS       LITERALLY '08h',
HT       LITERALLY '09h',/* `I */
LF       LITERALLY '0ah',/* `J */
CR       LITERALLY '0dh',
XON      LITERALLY '11h',/* `Q */
XOFF     LITERALLY '13h',/* `S */
NAK      LITERALLY '15h',/* `U */
ESC      LITERALLY '1bh',
DEL      LITERALLY '7fh',/* like NUL */
newline  LITERALLY 'CR,LF',
err       LITERALLY '40h',
true     LITERALLY '0ffh',
false    LITERALLY '0',
REPEAT   LITERALLY 'DO WHILE 1',
UNTIL    LITERALLY 'IF',
ENDREPEAT LITERALLY 'THEN UNDO;END';
/* REPEAT;
      OUTPUT (51h) = 40h;
      UNTIL INPUT (50h) = 0000$0010b ENDREPEAT; */

DECLARE /* port */
DATA8251 BYTE AT(0c000H),
CONTROL8251 BYTE AT(0c001H),
PORT8255A BYTE AT(0e000H),
PORT8255B BYTE AT(0e001H),
PORT8255C BYTE AT(0e002H),
CONTROL8255 BYTE AT(0e003H);

DECLARE
RxRdy    LITERALLY '(control8251 AND 0000$0010b)<>0',
TxRdy    LITERALLY '(control8251 AND 1000$0001b)=1000$0001b';

DECLARE /* constant */
icename(*) BYTE PUBLIC DATA('
STD Debugger for 8085',newline,ETX),
blank(*) BYTE PUBLIC DATA(' ',ETX),
allregname(*) BYTE DATA('
PC A F(SZAPC) B C D E H L IM IE SP',
newline,ETX);

DECLARE /* constant */
help(*) BYTE PUBLIC DATA('
Assemble A [address]',newline,'
Breakpoint B [address]',newline,'
Clearbreak C',newline,'
Dump D [address|range]',newline,'
Enter E address [list]',newline,'
Fill F range list',newline,'
Go G [address]',newline,'
Help H',newline,'
Input I address',newline,'
Load L',newline,'
Output O address byte',newline,'
Register R [registername]',newline,'
Trace T',newline,'
Unassemble U [S][address|range]',newline,'
Write W range',newline,'
reset X',ETX
),
error(*)BYTE PUBLIC DATA(' Error',newline,ETX),
write$guide(*) BYTE PUBLIC DATA('
Use write COM to file utility of your terminal.',newline,'
Strike a key when ready..',newline,ETX),
intelhexend(*) BYTE PUBLIC DATA('
:00000001FF',newline,
1ah,newline,'
Write ready.',newline,ETX),
loadguide(*) BYTE PUBLIC DATA('
Use load file to COM utility of your terminal.',newline,ETX),
loadready(*) BYTE PUBLIC DATA(
newline,'Load ready.',newline,ETX);

DECLARE regname(21) STRUCTURE(char(3) BYTE) DATA(
'PC ',
'AF ',
'BC ',
'DE ',
'HL ',
'SP ',
'A ',
'F ',
'B ',
'C ',
'D ',

```

```
'E ',
'H ',
'L ',
'IM ',
'IE ',
'SF ',
'ZF ',
'AF ',
'PF ',
'CF ');
```

```
DECLARE regspec(21) STRUCTURE(spec BYTE,number BYTE) DATA(
/*'PC ',*/ 2,0,
/*'AF ',*/ 2,1,
/*'BC ',*/ 2,2,
/*'DE ',*/ 2,3,
/*'HL ',*/ 2,4,
/*'SP ',*/ 2,6,
/*'A ',*/ 1,1,
/*'F ',*/ 0,1,
/*'B ',*/ 1,2,
/*'C ',*/ 0,2,
/*'D ',*/ 1,3,
/*'E ',*/ 0,3,
/*'H ',*/ 1,4,
/*'L ',*/ 0,4,
/*'IM ',*/ 1,5,
/*'IE ',*/ 10h,5,
/*'SF ',*/ 17h,1,
/*'ZF ',*/ 16h,1,
/*'AF ',*/ 14h,1,
/*'PF ',*/ 12h,1,
/*'CF ');*/ 10h,1);
```

```
DECLARE
/* table */
mnemonic(82) STRUCTURE(char(4) BYTE) EXTERNAL,
operand(20) STRUCTURE(char(3) BYTE) EXTERNAL,
insopc(246) BYTE EXTERNAL,
insmne(246) BYTE EXTERNAL,
insopr(246) STRUCTURE(opr(2) BYTE) EXTERNAL,
insbyte(246) BYTE EXTERNAL,
mneinsptr(82) BYTE EXTERNAL,
opcinsptr(256) BYTE EXTERNAL;
```

```
DECLARE /* label */
(init,prompt,prompt2) LABEL PUBLIC;
```

```
DECLARE /* variable that user can read write */
reg(13) ADDRESS PUBLIC,
(bp) BYTE PUBLIC,
(breakaddress) ADDRESS PUBLIC;
```

```
DECLARE /* variable */
string(64) BYTE PUBLIC,
paraword(4) ADDRESS PUBLIC,
(locationcounter,lastdump,nextunasm,addr,ptr) ADDRESS PUBLIC,
(character,function,digit,n,numword,numbyte,numdigit,numline,
pass,start,stop,tmp,sum,endsof$file) BYTE PUBLIC;
```

```
DECLARE /* variable */
ramcode(5) BYTE,
beginramcode ADDRESS DATA(.ramcode),
newregister BYTE PUBLIC;
```

```
/****** EXTERNAL UTILITY *****/
```

```
/* Read Interrupt Mask procedure form 80pl.lib */
RMASK:PROCEDURE BYTE EXTERNAL;
END RMASK;
```

```
/* Set Interrupt Mask procedure form 80pl.lib */
SMASK:PROCEDURE(bytevalue) EXTERNAL;
DECLARE bytevalue BYTE;
END SMASK;
```

```
/* external procedure from g80.asm */
peek:PROCEDURE(location) BYTE EXTERNAL;
DECLARE
location ADDRESS;
END peek;
```

```
poke:PROCEDURE(location,content) EXTERNAL;
DECLARE
location ADDRESS,
content BYTE;
END poke;
```

```
go$pc:PROCEDURE EXTERNAL;
END;
```

```

/***** ASSEMBLE UTILITY *****/
/* find end of input-string */
findend:PROCEDURE;
  CALL skipblank;
  IF string(n) <> 0
  THEN function = err;
  END findend;

/* find mnemonic */
findmnemonic:PROCEDURE;
  numbyte = 0;
  REPEAT;
    LOW(paraword(numbyte)) = upcase(string(n));
    numbyte = numbyte + 1; n = n + 1;
  UNTIL(string(n) = ' ')OR(string(n) = 0)OR(numbyte = 4)ENDREPEAT;
  IF (string(n) = ' ')OR(string(n) = 0)
  THEN DO WHILE numbyte < 4;
    LOW(paraword(numbyte)) = ' ';
    numbyte = numbyte + 1;
  END;
  ELSE function = err;
  IF function <> err
  THEN DO;
    numword,numbyte = 0;
    REPEAT;
      IF LOW(paraword(numbyte)) = mnemonic(numword).char(numbyte)
      THEN numbyte = numbyte + 1;
      ELSE DO;
        numbyte = 0;
        numword = numword + 1;
      END;
    UNTIL(numbyte = 4)OR(numword > LAST(mnemonic))ENDREPEAT;
    IF numword > LAST(mnemonic)
    THEN function = err;
  END;
  END findmnemonic;

/* find hex operand */
findhexoperand:PROCEDURE(count);
  DECLARE count BYTE;
  numdigit,pass,addr = 0;
  CALL skipblank;
  IF string(n)<>0
  THEN REPEAT;
    IF (string(n)>='0')AND(string(n)<='9')
    OR (string(n)>='A')AND(string(n)<='F')
    OR (string(n)>='a')AND(string(n)<='f')
    THEN DO;
      numdigit = numdigit + 1; pass = 1;
      IF (string(n)>='0')AND(string(n)<='9')
      THEN digit = string(n) - 30h;
      IF (string(n)>='A')AND(string(n)<='F')
      THEN digit = string(n) - 37h;
      IF (string(n)>='a')AND(string(n)<='f')
      THEN digit = string(n) - 57h;
      IF numdigit <= count
      THEN addr = addr * 16 + digit;
      ELSE function = err;
      n = n + 1;
    END;
  UNTIL
  NOT((string(n)>='0')AND(string(n)<='9')
  OR (string(n)>='A')AND(string(n)<='F')
  OR (string(n)>='a')AND(string(n)<='f'))ENDREPEAT;
  END findhexoperand;

/* compare operand */
compareoperand:PROCEDURE;
  numbyte = 0;
  compare:REPEAT;
    IF operand(numword).char(numbyte) = ' '
    THEN DO;
      CALL skipblank;
    END;
    ELSE IF(operand(numword).char(numbyte) = 'b')
    THEN DO;
      CALL findhexoperand(2);
      IF (pass = 1)AND(function <> err)
      THEN DO;
        paraword(stop) = addr;
        stop = stop + 1; /* number of operand byte */
        pass = 0;
      END;
    ELSE DO;
      pass,function = 0;
      UNDO compare;
    END;
  END;
  ELSE IF(operand(numword).char(numbyte) = 'd')
  THEN DO;

```

```

IF string(n) = '+'
THEN DO;
  n = n + 1;
  CALL findhexoperand(2);
  IF(pass = 1)AND(function <> err)AND(addr <= 7fh)
  THEN DO;
    paraword(stop) = addr;
    stop = stop + 1;/* number of operand byte */
    pass = 0;
    END;
  ELSE DO;
    pass,function = 0;
    UNDO compare;
    END;
  END;
ELSE IF string(n) = '-'
THEN DO;
  n = n + 1;
  CALL findhexoperand(2);
  IF(pass = 1)AND(function <> err)AND(addr <= 80h)
  THEN DO;
    paraword(stop) = - addr;
    stop = stop + 1;/* number of operand byte */
    pass = 0;
    END;
  ELSE DO;
    pass,function = 0;
    UNDO compare;
    END;
  END;
ELSE UNDO compare;
END;
ELSE IF(operand(numword).char(numbyte) = 'a')
OR (operand(numword).char(numbyte) = 'r')
THEN DO;
  CALL findhexoperand(4);
  IF (pass = 1)AND(function <> err)
  THEN DO;
    paraword(stop) = addr;
    stop = stop + 1;/* number of operand byte */
    pass = 0;
    END;
  ELSE DO;
    pass,function = 0;
    UNDO compare;
    END;
  END;
ELSE IF operand(numword).char(numbyte) = upcase(string(n))
THEN DO;
  n = n + 1;
  END;
ELSE UNDO compare;
  numbyte = numbyte + 1;
  UNTIL(numbyte = 3)ENDREPEAT compare;
END compareoperand;

/* fill byte-operand and address-operand */
filloperand:PROCEDURE;
  IF stop = 2
  THEN DO;/* 2 byte-operands */
    CALL poke(locationcounter + start,LOW(paraword(0)));
    CALL displaybyte(LOW(paraword(0)));
    CALL poke(locationcounter + start + 1,LOW(paraword(1)));
    CALL displaybyte(LOW(paraword(1)));
    END;
  ELSE IF stop = 1
  THEN DO;
    IF(insopr(ptr).opr(0) = 19)
    OR(insopr(ptr).opr(1) = 19)
    OR(insopr(ptr).opr(0) = 44)
    OR(insopr(ptr).opr(1) = 44)
    THEN DO;/* 1 address-operand */
      CALL poke(locationcounter + start,LOW(paraword(0)));
      CALL displaybyte(LOW(paraword(0)));
      CALL poke(locationcounter + start + 1,HIGH(paraword(0)));
      CALL displaybyte(HIGH(paraword(0)));
      END;
    ELSE DO;/* 1 byte-operand */
      CALL poke(locationcounter + start,LOW(paraword(0)));
      CALL displaybyte(LOW(paraword(0)));
      END;
    END;
  END filloperand;

/* fill opcode and operand of 2-byte opcode instruction */
fill2opcode:PROCEDURE;
  CALL poke(locationcounter,insopc(ptr));
  CALL poke(locationcounter + 1,insopc(ptr));
  CALL filloperand;
  locationcounter = locationcounter + insbyte(ptr);

```

```

/* find operand */
findoperand:PROCEDURE;
CALL skipblank;tmp = n; /* location of start character of operand */
ptr = mneinsptr(numword) - 1; /* instruction pointer */
pass = 0;
REPEAT;
  stop = 0; /* number of operand byte */
  IF insopr(ptr).opr(0) = 0
  THEN DO;
    IF string(n) = 0
    THEN pass = 1;
    END;
  ELSE IF insopr(ptr).opr(1) = 0
  THEN DO;
    numword = insopr(ptr).opr(0) - 1;
    CALL compareoperand;
    IF numbyte = 3
    THEN DO;
      CALL skipblank;
      IF string(n) = 0
      THEN pass = 1;
      END;
    END;
  ELSE DO;
    numword = insopr(ptr).opr(0) - 1;
    CALL compareoperand;
    IF numbyte = 3
    THEN DO;
      CALL skipblank;
      IF string(n) = ','
      THEN DO;
        n = n + 1;
        numword = insopr(ptr).opr(1) - 1;
        CALL skipblank;
        CALL compareoperand;
        IF numbyte = 3
        THEN DO;
          CALL skipblank;
          IF string(n) = 0
          THEN pass = 1;
          END;
        END;
      END;
    END;
  END;
  IF pass = 0
  THEN DO;
    ptr = ptr + 1;
    n = tmp;
    END;
  UNTIL(pass = 1)
  OR(insmne(ptr) <> insmne(ptr - 1))
  OR(ptr > LAST(insopr))ENDREPEAT;
IF pass = 1 /* ptr point at correct instruction */
THEN DO;
  start = 2; /* 2-byte opcode operands start at byte 2 */
  IF(insopc(ptr) = 0cbh)
  OR(insopc(ptr) = 0edh)
  THEN DO; /* 2-byte opcode */
    CALL fill2opcode;
    END;
  ELSE IF(insopc(ptr) = 0ddh)
  OR(insopc(ptr) = 0fdh)
  THEN DO;
    IF(insopc(ptr) = 0cbh)
    THEN DO;
      CALL poke(locationcounter + 3,insopc(ptr));
      END;
    CALL fill2opcode;
    END;
  ELSE DO; /* 1-byte opcode */
    start = 1; /* operands start at byte 1 */
    IF(insopr(ptr).opr(0) = 30)
    OR(insopr(ptr).opr(1) = 30)
    THEN DO; /* jmp relative from next address to paraword(0) */
      addr = locationcounter + 2; /* next address */
      IF((paraword(0) - addr) <= 07fh)
      OR((addr - paraword(0)) <= 080h)
      THEN DO;
        CALL poke(locationcounter,insopc(ptr));
        CALL displaybyte(insopc(ptr));
        CALL poke(locationcounter + 1,LOW(paraword(0) - addr));
        CALL displaybyte(LOW(paraword(0) - addr));
        locationcounter = addr;
        END;
      ELSE DO;
        CALL disparray(.('      Jump too far',endline,ETX));
        END;
      END;
    END;
  END;

```



```

        ELSE DO; /* not jump relative */
            CALL poke(locationcounter, insopc(ptr));
            CALL displaybyte(insopc(ptr));
            CALL filloperand;
            locationcounter = locationcounter + insbyte(ptr);
        END;
    END;
END;
ELSE function = err;
END findoperand;

/* display operand */
displayoperand:PROCEDURE PUBLIC;
IF character <> ' '
THEN DO;
    IF character = 'a'
    THEN DO;
        CALL displaybyte(string(start + 1));
        CALL displaybyte(string(start));
        numdigit = numdigit + 4;
    END;
    ELSE IF character = 'b'
    THEN DO;
        CALL displaybyte(string(start));
        start = start + 1;
        numdigit = numdigit + 2;
    END;
    ELSE IF character = 'd'
    THEN DO; /* displacement */
        IF string(start) < 80h
        THEN DO; /* +00 t0 +7F */
            CALL display('+');
            CALL displaybyte(string(start));
        END;
        ELSE DO; /* -01 t0 -80 */
            CALL display('-');
            CALL displaybyte(- string(start));
        END;
        start = start + 1;
        numdigit = numdigit + 3;
    END;
    ELSE IF character = 'r'
    THEN DO;
        IF string(start) < 80h
        THEN CALL displayword(nextunasm + stop + string(start));
        ELSE CALL displayword(nextunasm + stop - (- string(start)));
        numdigit = numdigit + 4;
    END;
    ELSE DO;
        CALL display(character);
        numdigit = numdigit + 1;
    END;
END;
END displayoperand;

/***** ASSEMBLE COMMAND *****/
/* Assemble */
assemble:PROCEDURE PUBLIC;
CALL findaddress;CALL findnul;
CALL saveregister;
IF numword = 1
THEN locationcounter = paraword(0);
readasm:DO WHILE 1;
    CALL display$word(locationcounter);
    CALL displayspace;CALL display(ACK);CALL displayspace;
    CALL readstring;
    DO WHILE n < 64;
        CALL displayspace;
        n = n + 1;
    END;
    n = 0;CALL skipblank;
    IF string(n) <> 0
    THEN DO;
        CALL findmnemonic;
        IF function <> err
        THEN DO;
            IF numword < 80 /* mnemonic number */
            THEN DO;
                CALL findoperand;
            END;
            ELSE IF numword = 80 /* DB */
            THEN DO;
                numbyte = 0;CALL findlist;CALL findend;
                IF(numbyte > 0)AND(function <> err)
                THEN DO;
                    DO n = 0 TO numbyte - 1;
                        CALL poke(locationcounter + n,string(n));
                    END;
                    locationcounter = locationcounter + numbyte;
                END;
            END;
        END;
    END;
END;

```

```

        END;
    ELSE IF numword = 81 /* DW */
    THEN DO;
        CALL findhex(4);CALL findend;
        IF(pass = 1)AND(function <> err)
        THEN DO;
            CALL poke(locationcounter,LOW(addr));
            CALL poke(locationcounter +1,HIGH(addr));
            locationcounter = locationcounter + 2;
        END;
    END;
    END;
    IF function = err
    THEN CALL disparray(.endline,' Error Code',ETX));
    CALL newline;
    END;
ELSE DO; /* no character */
    CALL newline;
    UNDO readasm;
    END;
END readasm;
END assemble;

/* Unassemble */
unassemble:PROCEDURE PUBLIC;
CALL findrange;CALL findnul;
IF numword > 0
THEN nextunasm = paraword(0);
ELSE paraword(0) = 0;
IF numword < 2
THEN paraword(1) = 0ffffh;
numline = 0; /* set number of lines count */
IF paraword(0) <= paraword(1)
THEN REPEAT;
    ptr,numbyte,numdigit = 0;
    DO n = 0 TO 3;
        string(n) = peek(nextunasm + n);
    END;
    start = 2; /* if 2-byte opcodes operands start at string(2) */
    /* look up OPC-INS table */
    DO;
        start = 1; /* 1-byte opcode operands start at string(1) */
        ptr = opcinsptr(string(0));
    END;
    IF ptr <> 0
    THEN DO;
        ptr = ptr - 1;
        stop = insbyte(ptr);
        CALL display$word(nextunasm);CALL display(' ');
        DO n = 0 TO 3;
            IF n < stop
            THEN CALL displaybyte(string(n));
            ELSE DO;
                CALL display(' ');CALL display(' ');
            END;
        END;
        CALL display(' ');
        DO n = 0 TO 3;
            CALL display(mnemonic(insmne(ptr) - 1).char(n));
        END;
        CALL display(' ');
        IF insopr(ptr).opr(0) <> 0
        THEN DO n = 0 TO 2;
            character = operand(insopr(ptr).opr(0) - 1).char(n);
            CALL displayoperand;
        END;
        IF insopr(ptr).opr(1) <> 0
        THEN DO;
            CALL display(',');
            numdigit = numdigit + 1;
            DO n = 0 TO 2;
                character = operand(insopr(ptr).opr(1) - 1).char(n);
                CALL displayoperand;
            END;
        END;
    END;
    END;
ELSE DO; /* error code -> DB xx[,xx[,xx[,xx]]] */
    IF(string(0) = 028h)OR(string(0) = 038h)
    THEN stop = 2;
    ELSE IF(string(0) = 0ddh)OR(string(0) = 0fdh)
    THEN stop = 3;
    ELSE stop = 1;
    CALL display$word(nextunasm);CALL display(' ');
    DO n = 0 TO 3;
        IF n < stop
        THEN CALL displaybyte(string(n));
        ELSE DO;
            CALL display(' ');CALL display(' ');
        END;
    END;
END;

```

```

CALL display(' ');
CALL disparray(.('DB ',ETX));
DO n = 0 TO 3;
  IF n < stop
  THEN DO;
    CALL displaybyte(string(n));
    numdigit = numdigit + 2;
  END;
  IF n < stop - 1
  THEN DO;
    CALL display(',');
    numdigit = numdigit + 1;
  END;
END;
END;
DO WHILE numdigit < 10;
CALL display(' ');
numdigit = numdigit + 1;
END;
CALL newline;
IF numword < 2
THEN numline = numline + 1;
nextunasm = nextunasm + stop;
UNTIL (CARRY) OR (numline = 11)
OR (nextunasm > paraword(1)) ENDREPEAT;
ELSE function = err;
END unassemble;

/***** BREAKPOINT UTILITY *****/
/* display breakpoint */
displaybreakpoint:PROCEDURE PUBLIC;
  IF NOT(bp)
  THEN CALL disparray(.('Break Point Clear',ETX));
  ELSE DO;
    CALL disparray(.('Break Point Address ',ETX));
    CALL displayword(breakaddress);
  END;
  CALL newline;
END displaybreakpoint;

/* set breakpoint */
setbreakpoint:PROCEDURE PUBLIC;
  bp = true;
  breakaddress = paraword(0);
END setbreakpoint;

/* clear breakpoint */
clearbreakpoint:PROCEDURE PUBLIC;
  bp = false;
END clearbreakpoint;

/* Breakpoint */
breakpoint:PROCEDURE PUBLIC;
  numword,pass = 0;
  CALL findaddress;CALL findnul;
  IF numword = 0
  THEN CALL display$breakpoint;
  ELSE CALL set$breakpoint;
  END breakpoint;

/***** EMULATION PROCEDURE *****/
saverregister:PROCEDURE PUBLIC;
  IF LOW(reg(5))
  THEN DO; /* IE = 1 */
    HIGH(reg(5)) = RMASK OR 0001$0000b;
    LOW(reg(5)) = 1;
  END;
  ELSE DO; /* IE = 0 */
    HIGH(reg(5)) = RMASK AND 1110$1111b;
    LOW(reg(5)) = 0;
  END;
END saverregister;

peekmoni:PROCEDURE(location) BYTE PUBLIC;
  DECLARE
  location ADDRESS,
  mem BASED location BYTE;
  RETURN mem;
END peekmoni;

pokemoni:PROCEDURE(location,content) PUBLIC;
  DECLARE
  location ADDRESS,
  content BYTE,
  mem BASED location BYTE;
  mem = content;
  END pokemoni;

peekport:PROCEDURE(location) BYTE PUBLIC;
  DECLARE location ADDRESS;

```

```

ramcode(0) = 0dbh;          /* IN location */
ramcode(1) = LOW(location);
ramcode(2) = 0c9h;        /* RET */
CALL beginramcode;
END peekport;

pokeport:PROCEDURE(location,content) PUBLIC;
DECLARE location ADDRESS,content BYTE;
ramcode(0) = 03eh;        /* MVI A,content */
ramcode(1) = content;
ramcode(2) = 0d3h;        /* OUT location */
ramcode(3) = LOW(location);
ramcode(4) = 0c9h;        /* RET */
CALL beginramcode;
END pokeport;

pokereg:PROCEDURE(spec,number,content) PUBLIC;
DECLARE
(spec,number) BYTE,
content ADDRESS;
DO case spec;
/* 0 = register low */
LOW(reg(number)) = LOW(content);
/* 1 = register high */
HIGH(reg(number)) = LOW(content);
/* 2 = register pair */
reg(number) = content;
END;
END pokereg;

resetprocessor:PROCEDURE PUBLIC;
nextunasm = 0; /* after reset 8085 */
reg(0) = 0; /* PC = 0 */
/* SOD = 0,all interrupts masked */
CALL SMASK(0101$1111b);
DISABLE;
HIGH(reg(5)) = RMASK;
LOW(reg(5)) = 0;
END resetprocessor;

/* update hardware after set register or use Zmenu */
updatehardware:PROCEDURE PUBLIC;
IF bit(LOW(reg(5)),7) /* new IM */
THEN DO;
CALL SMASK(HIGH(reg(5)));
HIGH(reg(5)) = RMASK;
END;
END updatehardware;

displayallreg:PROCEDURE;
CALL displayspace;
CALL displayword(reg(0)); /* PC */
CALL displayspace;
CALL displaybyte(HIGH(reg(1))); /* A */
tmp = LOW(reg(1));
CALL displayspace;
CALL displaybyte(tmp); /* F */
CALL displayspace;
CALL display(bit(tmp,7)+30h); /* S */
CALL display(bit(tmp,6)+30h); /* Z */
CALL display(bit(tmp,4)+30h); /* A */
CALL display(bit(tmp,2)+30h); /* P */
CALL display(bit(tmp,0)+30h); /* C */
DO tmp = 2 TO 4;
CALL displayspace;
CALL displaybyte(HIGH(reg(tmp))); /* B D H */
CALL displayspace;
CALL displaybyte(LOW(reg(tmp))); /* C E L */
END;
CALL displayspace;
CALL displaybyte(HIGH(reg(5))); /* IM */
CALL displayspace;
CALL display(bit(reg(5),0)+30h); /* IE */
CALL displayspace;
CALL displayword(reg(6)); /* SP */
CALL newline;
END displayallreg;

/* read data to change register pair */
readreg:PROCEDURE(spec,number);
DECLARE (spec,number) BYTE;
/* 0,1,2 register byte or word */
IF spec <= 2
THEN DO;
CALL readnibble((spec AND 10b) + 2);
CALL checkdata((spec AND 10b) + 2);
IF (pass = 1)OR(pass = -1)
THEN DO;
CALL pokereg(spec,number,addr);
IF number = 5 /* set flag to set new IM */

```

```

        THEN LOW(reg(5)) = LOW(reg(5))OR 1000$0000b;
        END;
        ELSE IF pass = 0
        THEN GOTO prompt;
        END;
        ELSE IF spec >= 10h
        THEN DO;
            CALL readbit(1);
            IF string(0) = 0
            THEN GOTO prompt;
            IF string(0) = '0'
            THEN CALL pokereg(0,number,clearbit(reg(number),spec - 10h));
            IF string(0) = '1'
            THEN CALL pokereg(0,number,setbit(reg(number),spec - 10h));
            END;
        END readreg;

/* find register name */
findregname:PROCEDURE;
    CALL skipblank;
    numbyte,pass = 0;
    IF string(n)<>0
    THEN DO;
        REPEAT;
            LOW(paraword(numbyte)) = upcase(string(n));
            numbyte = numbyte + 1;n = n + 1;
            UNTIL(string(n) = ' 'OR(string(n) = 0)OR(numbyte = 3))ENDREPEAT;
        DO WHILE numbyte < 3;
            LOW(paraword(numbyte)) = ' ';
            numbyte = numbyte + 1;
            END;
        numword,numbyte = 0;
        REPEAT;
            IF LOW(paraword(numbyte)) = regname(numword).char(numbyte)
            THEN numbyte = numbyte + 1;
            ELSE DO;
                numbyte = 0;
                numword = numword + 1;
                END;
            UNTIL(numbyte = 3)OR(numword > LAST(regname))ENDREPEAT;
        IF numword > LAST(regname)
        THEN function = err;
        ELSE pass = 1;
        END;
    END findregname;

/* display and read one register */
oneregister:PROCEDURE;
    DO n = 0 TO 2;
        CALL display(regname(numword).char(n));
        END;
    CALL displayspace;
    IF regspect(numword).spec = 0
    THEN CALL displaybyte(LOW(reg(regspect(numword).number)));
    ELSE IF regspect(numword).spec = 1
    THEN CALL displaybyte(HIGH(reg(regspect(numword).number)));
    ELSE IF regspect(numword).spec = 2
    THEN CALL displayword(reg(regspect(numword).number));
    ELSE IF regspect(numword).spec >= 10h
    THEN CALL display(
        bit(
        LOW(reg(regspect(numword).number)),regspect(numword).spec - 10h
        ) + 30h
        );
    CALL newline;CALL display(ACK);
    CALL readreg(regspect(numword).spec,regspect(numword).number);
    CALL newline;
    END oneregister;

/* display and read all registers */
allregister:PROCEDURE PUBLIC;
    CALL disparray(.allregname);
    CALL displayallreg;
    END allregister;

/***** PROCESSOR COMMAND *****/
/* Input */
inputcommand:PROCEDURE PUBLIC;
    CALL findaddress;CALL findnul;
    IF numword = 1 AND paraword(0) < 100h
    THEN DO;
        CALL saveregister;
        CALL display$byte(peekport(paraword(0)));
        CALL newline;
        END;
    ELSE function = err;
    END inputcommand;

/* Outputcommand */
outputcommand:PROCEDURE PUBLIC;
    CALL findaddress;CALL findbyte;CALL findnul;

```

```

IF numword = 1 AND paraword(0) < 100h
AND numbyte = 1
THEN DO;
    CALL saveregister;
    CALL pokeport(paraword(0),string(0));
END;
ELSE function = err;
END outputcommand;

/* Register */
register:PROCEDURE PUBLIC;
CALL findregname;CALL findnul;
CALL saveregister;
IF pass = 1
THEN CALL oneregister;
ELSE CALL allregister;
CALL updatehardware;
END register;

/* Trace Command */
tracecommand:PROCEDURE PUBLIC;
CALL findnul;
/* execute with breakpoint at PC */
PORT8255A = HIGH(reg(0));/* out A15 - A8 */
PORT8255B = LOW(reg(0)); /* out A7 - A0 */
PORT8255C = 0000$0000b; /* enable G08255 and EN688 */
CALL gopc;
END tracecommand;

/* Go */
gocommand:PROCEDURE PUBLIC;
CALL findaddress;CALL findnul;
IF numword = 1 /* set new PC */
THEN reg(0) = paraword(0);
IF NOT bp
THEN DO;/* execute without break */
    PORT8255C = 0000$0001b; /* enable G08255 ,diable EN688 */
END;
ELSE DO;/* execute with breakpoint */
    PORT8255A = HIGH(breakaddress);/* out A15 - A8 */
    PORT8255B = LOW(breakaddress); /* out A7 - A0 */
    PORT8255C = 0000$0000b; /* enable G08255 and EN688 */
END;
CALL gopc;
END gocommand;

/***** SOFTWARE UTILITY *****/
/* set error while find Intel hex record */
seterror:PROCEDURE;
function = err;
GOTO prompt2;
END seterror;

/* skip blank in input-string */
skipblank:PROCEDURE PUBLIC;
DO WHILE string(n)=' ';
    n = n + 1;
END;
END skipblank;

/* return upper case */
upcase:PROCEDURE(char) BYTE PUBLIC;
DECLARE char BYTE;
IF(char >= 'a')AND(char <= 'z')
THEN char = char - 20h;
RETURN char;
END upcase;

/* return bit value */
bit:PROCEDURE(onebyte,count) BYTE PUBLIC;
DECLARE (onebyte,count) BYTE;
RETURN (SHR(onebyte,count) AND 01b);
END bit;

/* set bit */
setbit:PROCEDURE(onebyte,count) BYTE PUBLIC;
DECLARE (onebyte,count) BYTE;
RETURN (onebyte OR ROL(0000$0001b,count));
END setbit;

/* clear bit */
clearbit:PROCEDURE(onebyte,count) BYTE PUBLIC;
DECLARE (onebyte,count) BYTE;
RETURN (onebyte AND ROL(1111$1110b,count));
END clearbit;

/* initialize value */
initvalue:PROCEDURE PUBLIC;
DECLARE checkstart BYTE;
/* warmstart */

```

```

CALL setrs232;
locationcounter,function,bp = 0;
lastdump = 0ffffh;
IF checkstart <> (NOT(LOW(.checkstart)))
THEN DO; /* coldstart */
  DO n = 0 TO 12;
    reg(n) = 0;
  END;
  CALL newline;
  CALL disparray(.icename);
  CALL disparray(.help);
  checkstart = (NOT(LOW(.checkstart)));
  END;
CALL resetprocessor;
END initvalue;

/* clear string for input */
clearstring:PROCEDURE PUBLIC;
  DO n = 0 TO 63;
    string(n) = 0;
  END;
END clearstring;

/* read input from keyboard until press RETURN */
readstring:PROCEDURE PUBLIC;
  CALL clearstring;n = 0;function = 0;
  REPEAT;
    character = readkbd;
    IF (character>=' ')AND(character<7fh)
    THEN DO;
      IF n < 63 /* max = 63 character */
      THEN DO;
        CALL display(character);
        string(n) = character;
        n = n + 1;
      END;
      ELSE CALL display(BEL);
      END;
    IF (character=BS)AND(n>0)
    THEN DO;
      CALL display(BS);
      CALL displayspace;
      CALL display(BS);
      string(n-1) = 0;
      n = n - 1;
      END;
    IF character=ESC
    THEN GOTO prompt;
    UNTIL character=CR ENDREPEAT;
  END readstring;

readline:PROCEDURE PUBLIC;
  CALL readstring;
  CALL newline;
  END readline;

/* find function */
findfunction:PROCEDURE PUBLIC;
  n,numword,numbyte = 0;
  CALL skipblank;
  IF (string(n)<>0)AND(function=0)
  THEN DO;
    character = upcase(string(n));
    IF(character >= err)AND(character <= 'Z')
    THEN function = character;
    ELSE function =err;
    n = n + 1;
  END;
  END findfunction;

/* find hex value */
findhex:PROCEDURE(count) PUBLIC;
  DECLARE count BYTE;
  numdigit,pass,addr = 0;
  DO WHILE (string(n)=' ')OR(string(n)=' ');
    n = n + 1;
  END;
  IF string(n)<>0
  THEN REPEAT;
    IF (string(n)>='0')AND(string(n)<='9')
    OR (string(n)>='A')AND(string(n)<='F')
    OR (string(n)>='a')AND(string(n)<='f')
    THEN DO;
      numdigit = numdigit + 1; pass = 1;
      IF (string(n)>='0')AND(string(n)<='9')
      THEN digit = string(n) - 30h;
      IF (string(n)>='A')AND(string(n)<='F')
      THEN digit = string(n) - 37h;
      IF (string(n)>='a')AND(string(n)<='f')
      THEN digit = string(n) - 57h;
    END;
  END REPEAT;

```

```

        IF numdigit <= count
        THEN addr = addr * 16 + digit;
        ELSE function = err;
        n = n + 1;
        END;
    UNTIL
    NOT((string(n)>='0')AND(string(n)<='9')
    OR (string(n)>='A')AND(string(n)<='F')
    OR (string(n)>='a')AND(string(n)<='f'))ENDREPEAT;
    END findhex;

/* return next character */
nextcharacter:PROCEDURE BYTE PUBLIC;
    RETURN string(n);
    END nextcharacter;

/* return begin character of next word */
beginnextword:PROCEDURE BYTE PUBLIC;
    DO WHILE (string(n)=' ')OR(string(n)=',' );
        n = n + 1;
    END;
    RETURN string(n);
    END beginnextword;

/* find byte */
findbyte:PROCEDURE PUBLIC;
    CALL findhex(2);
    string(numbyte) = addr;
    numbyte = numbyte + pass;
    END findbyte;

/* find address */
findaddress:PROCEDURE PUBLIC;
    CALL findhex(4);
    paraword(numword) = addr;
    numword = numword + pass;
    END findaddress;

/* find address or range */
findrange:PROCEDURE PUBLIC;
    CALL findaddress;
    character = beginnextword;
    IF pass = 1
    THEN IF (character='L')OR(character='I')
    THEN DO;
        n = n + 1;
        CALL findhex(4);
        IF (pass = 0)OR(pass = 1)AND(addr = 0)
        THEN function = err;
        paraword(numword)=paraword(numword-1) + addr - 1;
        numword = numword + pass;
        END;
    ELSE CALL findaddress;
    END findrange;

/* find string in quotation marks */
findstringinquote:PROCEDURE(mark) PUBLIC;
    DECLARE mark BYTE;
    REPEAT;
        n = n + 1; /* skip begin mark and get next character */
        IF (string(n)>=' ')AND(string(n)<7fh)AND(string(n)<>mark)
        THEN DO;
            string(numbyte) = string(n);
            numbyte = numbyte + 1;
            END;
        IF string(n)=0
        /* end of line without end mark -> error */
        THEN function = err;
        UNTIL (string(n)=mark)OR(function = err) ENDREPEAT;
        n = n + 1; /* skip end mark */
    END findstringinquote;

/* find list */
findlist:PROCEDURE PUBLIC;
    REPEAT;
        character = beginnextword;
        IF character = 27h
        /* string in single quotation marks */
        THEN CALL findstringinquote(27h);
        ELSE IF character = 22h
        /* string in double quotation marks */
        THEN CALL findstringinquote(22h);
        ELSE IF character <> 0
        /* a byte in hex digit form */
        THEN DO;
            CALL findbyte;
            IF pass = 0
            THEN DO;
                function = err;
                n = n + 1;
            END;
        END;
    END;

```



```

        END;
        END;
        UNTIL (string(n)=0)OR(function = err) ENDREPEAT;
    END findlist;

/* find nul */
findnul:PROCEDURE PUBLIC;
    IF beginnextword <> 0
    THEN function = err;
    IF function = err
    THEN DO;
        GOTO prompt2;
    END;
    END findnul;

readbytehyphen:PROCEDURE PUBLIC;
    CALL readnibble(2);
    DO n = 0 TO 2;
        IF string(n) = '-'
        THEN CALL display('-');
    END;
    CALL newline;
    END readbytehyphen;

/* read nibble */
readnibble:PROCEDURE(count) PUBLIC;
    DECLARE count BYTE;
    CALL clearstring;
    n = 0;
    REPEAT;
        character = upcase(readkbd);
        IF(n < count) AND
        ( (character='0')AND(character<='9')
        OR(character='A')AND(character<='F'))
        THEN DO;
            CALL display(character);
            string(n) = character;
            n = n + 1;
        END;
        IF (character=BS)AND(n>0)
        THEN DO;
            CALL display(BS);CALL displayspace;CALL display(BS);
            string(n-1) = 0;
            n = n - 1;
        END;
        IF (character=' ')
        OR (character='-')
        THEN string(n) = character;
        IF (character=ESC)
        THEN GOTO prompt;
        UNTIL (character=CR)
        OR (character=' ')
        OR (character='-')ENDREPEAT;
    DO WHILE n <= count;
        CALL displayspace;
        n = n + 1;
    END;
    END readnibble;

/* read bit */
readbit:PROCEDURE(count) PUBLIC;
    DECLARE count BYTE;
    DO n = 0 TO count;
        string(n) = 0;
    END;
    n = 0;
    REPEAT;
        character = readkbd;
        IF (n < count) AND
        ( (character='0')
        OR(character='1')
        OR(character=' ')
        THEN DO;
            CALL display(character);
            string(n) = character;
            n = n + 1;
        END;
        IF (character=BS)AND(n>0)
        THEN DO;
            CALL display(BS);
            CALL displayspace;
            CALL display(BS);
            string(n-1) = 0;
            n = n - 1;
        END;
        IF (character=ESC)
        THEN GOTO prompt;
        UNTIL(n=count)AND(character=' ')
        OR (character=CR) ENDREPEAT;
    DO WHILE n <= count;

```

```

        CALL displayspace;
        n = n + 1;
        END;
    END readbit;

/* check data */
checkdata:PROCEDURE(count) PUBLIC;
    DECLARE count BYTE;
    n = 0;
    CALL findhex(count);
    IF (pass=0)AND(string(0)=' ')
    THEN pass='+';
    IF (pass=0)AND(string(0)='-')
    THEN pass='-';
    IF (pass=1)AND(string(n)='-')
    THEN pass=-1;
    END checkdata;

/* display 16 byte from memory */
display$line:PROCEDURE PUBLIC;
    CALL display$word(addr);CALL displayspace;
    DO n = 0 TO 0fh ;
        ptr = addr + n;
        IF (n < start) OR (n > stop)
        THEN DO;
            CALL displayspace;CALL displayspace;CALL displayspace;
            END;
        ELSE IF (n = 8) AND (n <> start)
        THEN DO;
            string(n) = peek(ptr);
            CALL display('-');CALL display$byte(string(n));
            END;
        ELSE DO;
            string(n) = peek(ptr);
            CALL displayspace;CALL display$byte(string(n));
            END;
        END;
    CALL displayspace;CALL displayspace;CALL displayspace;
    DO n = 0 TO stop;
        ptr = addr + n;
        IF n < start
        THEN CALL displayspace;
        ELSE IF (string(n)>=' ')AND(string(n)<7fh)
        THEN CALL display(string(n));
        ELSE CALL display('.');
        END;
    lastdump = ptr;
    CALL newline;
    END display$line;

/* display 16 byte from memory in Intel HEX-ASCII format */
displayintelhexline:PROCEDURE PUBLIC;
    sum = stop - start + 1;
    CALL display(':');CALL display$byte(sum);
    ptr = addr + start;
    sum = sum + HIGH(ptr) + LOW(ptr);
    CALL display$word(ptr);CALL display('0');CALL display('0');
    DO n = start TO stop;
        tmp = peek(addr + n);
        sum = sum + tmp;
        CALL display$byte(tmp);
        END;
    CALL display$byte( - sum);
    CALL newline;
    END displayintelhexline;

/* read Intel hex line */
readIntelhex:PROCEDURE PUBLIC;
/* Intel HEX-ASCII format
: B C A A A T T H H ... H H C C
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | |
checksum : checksum + sum = 0
data bytes : 2 characters per byte
record type: 00 = data,01 = end of file
address : first address
byte count : number of data bytes

```

data used ASCII 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F
 NUL (00) and DEL (7F) are ignored.

example

```

:100000031650221650236003A6502FE01D21B000D
:100010003E04D39021650234C3080021650236FFF7
:100020003A6502FE01D233003E05D39021650234C9
:10003000C3200021650236003A6502FE02D24B0061
:100040003E06D39021650234C33800216502360094
:100050003A6502FE01D263003E04D390216502346A
:05006000C35000FB7617
:00000001FF

```

*** byte count in end of file record must be 00

```

*/
/* wait for start of record ':' */
DO WHILE read$with$timeout <> ':';
  END;
sum = 0;
/* read byte count */
/* number of data buffer is 64 bytes */
numbyte = readdatabyte;
IF numbyte > 64
THEN CALL loaderror;
/* read address */
HIGH(addr) = readdatabyte;
LOW(addr) = readdatabyte;
/* read record type */
tmp = readdatabyte;
IF tmp = 00
THEN end$of$file = false;
ELSE IF (tmp = 01)AND(numbyte = 0)
THEN end$of$file = true;
ELSE CALL loaderror;
DO n = 1 TO numbyte;
  /* fill data byte into buffer */
  string(n - 1) = readdatabyte;
  END;
/* checksum */
tmp = readdatabyte;
IF sum <> 0
THEN CALL loaderror;
END readintelhex;

/* write memory */
writememory:PROCEDURE PUBLIC;
CALL saveregister;
DO n = 1 TO numbyte;
  CALL poke(addr + n - 1,string(n - 1));
  END;
END writememory;

/* write monitormemory */
writemoni:PROCEDURE PUBLIC;
CALL saveregister;
DO n = 1 TO numbyte;
  CALL pokemoni(addr + n - 1,string(n - 1));
  END;
END writemoni;

/* read data byte */
readdatabyte:PROCEDURE BYTE PUBLIC;
repeat1:
character = read$with$timeout;
IF (character>='0')AND(character<='9')
THEN digit = character - 30h;
ELSE IF (character>='A')AND(character<='F')
THEN digit = character - 37h;
ELSE IF (character<>NUL)AND(character<>DEL)
THEN CALL loaderror;
ELSE GOTO repeat1;
repeat2:
character = read$with$timeout;
IF (character>='0')AND(character<='9')
THEN digit = SHL(digit,4) + character - 30h;
ELSE IF (character>='A')AND(character<='F')
THEN digit = SHL(digit,4) + character - 37h;
ELSE IF (character<>NUL)AND(character<>DEL)
THEN CALL loaderror;
ELSE GOTO repeat2;
sum = sum + digit;
RETURN digit;
END readdatabyte;

/* find data byte */
finddatabyte:PROCEDURE BYTE PUBLIC;
character = string(n);n = n + 1;
IF (character>='0')AND(character<='9')
THEN digit = character - 30h;
ELSE IF (character>='A')AND(character<='F')
THEN digit = character - 37h;
ELSE CALL seterror;
character = string(n);n = n + 1;
IF (character>='0')AND(character<='9')
THEN digit = SHL(digit,4) + character - 30h;
ELSE IF (character>='A')AND(character<='F')
THEN digit = SHL(digit,4) + character - 37h;
ELSE CALL seterror;
sum = sum + digit;
RETURN digit;
END finddatabyte;

findIntelhex:PROCEDURE PUBLIC;
IF string(n) <> ':'

```

```

THEN CALL seterror;
n = n + 1;
sum = 0;
/* read byte count */
numbyte = finddatabyte;
/* read address */
HIGH(addr) = finddatabyte;
LOW(addr) = finddatabyte;
/* read record type */
tmp = finddatabyte;
IF NOT(tmp = 00h
OR tmp = 01h AND numbyte = 0)
THEN CALL seterror;
DO tmp = 1 TO numbyte;
/* fill data byte into buffer */
string(tmp - 1) = finddatabyte;
END;
/* checksum */
tmp = finddatabyte;
IF sum <> 0
THEN CALL seterror;
END findintelhex;

/* set RS232 */
setRS232:PROCEDURE PUBLIC;
CONTROL8251 = 00H; /* reset 8251 to set mode word */
CONTROL8251 = 00H;
CONTROL8251 = 00H;
CONTROL8251 = 40H;
CONTROL8251 = 0100$1110B; /* mode word */
/* 1 stop bit no parity 8 bit baud rate 16x */
CONTROL8251 = 0011$0111B; /* command word */
/* 0 0 1 1 0 1 1 1 */
/* transmit enable */
/* DTR =0,data terminal ready */
/* receive enable */
/* reset all error flag */
/* RST =0,request to send */
CONTROL8255 = 80H; /* control word */
/* mode 0 : port a,b,c is output port */
PORT8255C = 0000$0011B; /* disable G08255 and EN688 */
END setRS232;

/* wait for first character */
wait$first:PROCEDURE PUBLIC;
/*check RxRdy */
DO WHILE NOT(RxRdy);
END;
END waitfirst;

/* load error */
loaderror:PROCEDURE PUBLIC;
function = err;
CALL disparray(.,(endline,'Load',ETX));
GOTO prompt2;
END loaderror;

/* read with time out */
read$with$timeout:PROCEDURE BYTE PUBLIC;
DECLARE
count ADDRESS,
round BYTE;
/*check RxRdy */
count = 0;round = 4;
DO WHILE NOT(RxRdy);
count = count - 1;
IF count = 0
THEN DO;
round = round - 1;
IF round = 0
THEN CALL loaderror;
END;
END;
RETURN data8251;
END read$with$timeout;

/* read ASCII from keyboard */
readkbd:PROCEDURE BYTE PUBLIC;
REPEAT;
UNTIL RxRdy ENDREPEAT;
RETURN data8251;
END readkbd;

/* display character from ASCII code */
display:PROCEDURE(chr) PUBLIC;
DECLARE chr BYTE;
REPEAT;
UNTIL TxRdy ENDREPEAT;
data8251 = chr;
END display;

```

```

/* display space */
display$space:PROCEDURE PUBLIC;
  CALL display(' ');
  END display$space;

/* display new line */
newline:PROCEDURE PUBLIC;
  CALL display(CR);
  CALL display(LF);
  END newline;

/* display array of characters */
disparray:PROCEDURE(startaddress) PUBLIC;
  DECLARE
    (startaddress,count) ADDRESS,
    char BYTE BASED count;
    count = startaddress;
  REPEAT;
    CALL display(char);
    count = count + 1;
  UNTIL char = ETX ENDREPEAT;
  END disparray;

/* display byte in hexadecimal format */
display$byte:PROCEDURE(onebyte) PUBLIC;
  DECLARE onebyte BYTE;
  /* display high nibble */
  IF SHR(onebyte,4) < 10
  THEN CALL display(SHR(onebyte,4) + 30h);
  ELSE CALL display(SHR(onebyte,4) + 37h);
  /* display low nibble */
  IF (onebyte AND 0fh) < 10
  THEN CALL display((onebyte AND 0fh) + 30h);
  ELSE CALL display((onebyte AND 0fh) + 37h);
  END display$byte;

/* display word in hexadecimal format */
display$word:PROCEDURE(oneword) PUBLIC;
  DECLARE oneword ADDRESS;
  /* display high byte */
  CALL display$byte(HIGH(oneword));
  /* display low byte */
  CALL display$byte(LOW(oneword));
  END display$word;

/***** COMMAND PROCEDURE *****/
/*
j:01xxxx00nncc
01 = number of data = 1 (nn)
xxxx = start address
00 = record type data
nn = data = number of data to be shown
cc = checksum
*/
/* read user memory */
jmenu:PROCEDURE;
  CALL findIntelhex;CALL findnul;
  IF numbyte = 1
  THEN DO;
    CALL saveregister;
    CALL display(':');CALL display$byte(string(0));
    sum = string(0) + HIGH(addr) + LOW(addr);
    CALL display$word(addr);CALL display('0');CALL display('0');
    DO n = 1 TO string(0);
      tmp = peek(addr + n - 1);
      sum = sum + tmp;
      CALL display$byte(tmp);
    END;
    CALL display$byte(- sum);
    CALL newline;
  END;
  ELSE function = err;
  END jmenu;

/* write user memory */
kmenu:PROCEDURE;
  CALL findIntelhex;CALL findnul;
  CALL saveregister;
  CALL writememory;
  END kmenu;

/* read monitor memory */
ymenu:PROCEDURE;
  CALL findIntelhex;CALL findnul;
  IF numbyte = 1
  THEN DO;
    CALL saveregister;
    CALL display(':');CALL display$byte(string(0));
    sum = string(0) + HIGH(addr) + LOW(addr);

```

```

CALL display$word(addr);CALL display('0');CALL display('0');
DO n = 1 TO string(0);
  tmp = peekmon( addr + n - 1);
  sum = sum + tmp;
  CALL display$byte(tmp);
  END;
CALL display$byte( - sum);
CALL newline;
END;
ELSE function = err;
END ymenu;

/* write monitor memory */
zmenu:PROCEDURE;
CALL findIntelhex;CALL findnul;
CALL saveregister;
CALL writemoni;
CALL updatehardware;
END zmenu;

/* Dump */
dump:PROCEDURE;
CALL findrange;CALL findnul;
DO CASE numword;
  /* no paraword -> display 40h bytes after last dump */
  DO;
    paraword(0)=lastdump+1;
    IF paraword(0)<0ff80h
      THEN paraword(1)=paraword(0)+7fh;
      ELSE paraword(1)=0ffffh;
    END;
  /* 1 paraword -> display 40h bytes */
  DO;
    IF paraword(0)<0ff80h
      THEN paraword(1)=paraword(0)+7fh;
      ELSE paraword(1)=0ffffh;
    END;
  /* 2 parawords -> display memory */
  ;
  /* 3 parawords -> error */
  function = err;
  END;
/* display memory */
IF function <> err
THEN DO;
  IF paraword(1) >= paraword(0)
  THEN DO;
    CALL saveregister;
    addr = paraword(0) AND 0fff0h;
    REPEAT;
      IF paraword(0)-addr <= 0fh
      THEN start = paraword(0) - addr;
      ELSE start = 00h;
      IF paraword(1)-addr <= 0fh
      THEN stop = paraword(1) - addr;
      ELSE stop = 0fh;
      CALL display$line;
      addr = addr + 10h;
      UNTIL(CARRY)
      OR(addr > (paraword(1) AND 0fff0h))ENDREPEAT;
    END;
  ELSE function = err;
  END;
END dump;

/* Enter */
enter:PROCEDURE;
CALL findaddress;CALL findlist;CALL findnul;
IF (numword = 1)AND(numbyte = 0) /* display/enter byte by byte */
THEN DO;
  CALL saveregister;
  REPEAT;
    CALL display$word(paraword(0));
    CALL displayspace;CALL displayspace;
    CALL display$byte(peek(paraword(0)));
    CALL displayspace;CALL display(ACK);CALL displayspace;
    /* read new data from keyboard */
    CALL readbytehyphen;
    CALL checkdata(2);
    IF pass=1
    THEN DO;
      CALL poke(paraword(0),LOW(addr));
      paraword(0) = paraword(0) + 1;
      END;
    ELSE IF pass=-1
    THEN DO;
      CALL poke(paraword(0),LOW(addr));
      paraword(0) = paraword(0) - 1;
      END;
    ELSE IF pass='+'

```

```

        *
        THEN paraword(0) = paraword(0) + 1;
        ELSE IF pass='- '
        THEN paraword(0) = paraword(0) - 1;
        UNTIL pass=0 ENDREPEAT;
    END;
ELSE IF numword = 1 /* enter list */
THEN DO;
    CALL saveregister;
    DO n = 0 TO numbyte - 1;
        CALL poke(paraword(0) + n ,string(n));
    END;
    END;
ELSE function = err;
END enter;

/* Fill memory block with list */
fill:PROCEDURE;
CALL findrange;CALL findlist;CALL findnul;
IF (numword=2)AND(paraword(1)>=paraword(0))AND(numbyte>0)
THEN DO;
    CALL saveregister;
    ptr = paraword(0);n = 0;
    REPEAT;
        CALL poke(ptr,string(n));
        n = n + 1;
        IF n = numbyte
        THEN n = 0;
        ptr = ptr + 1;
    UNTIL(ptr - 1) = paraword(1) ENDREPEAT;
    END;
ELSE function = err;
END fill;

/* Load command */
load:PROCEDURE;
CALL findnul;
/* send load guide */
CALL disparray(.loadguide);
/* wait for first character */
CALL wait$first;
REPEAT;
    /* read Intel hex line */
    CALL read$intel$h$hex;
    /* write memory */
    CALL writememory;
    /* send ACK */
    CALL display(ACK);
    UNTIL end$of$file ENDREPEAT;
nextunasm,reg(0) = addr;
/* send load ready */
CALL disparray(.loadready);
END load;

/* Write */
write:PROCEDURE;
CALL findrange;CALL findnul;
IF (numword=2)AND(paraword(1)>=paraword(0))
THEN DO;
    CALL saveregister;
    CALL disparray(.write$guide);
    character = readkbd;
    addr = paraword(0) AND 0fff0h;
    REPEAT;
        IF paraword(0)-addr <= 0fh
        THEN start = paraword(0) - addr;
        ELSE start = 00h;
        IF paraword(1)-addr <= 0fh
        THEN stop = paraword(1) - addr;
        ELSE stop = 0fh;
        CALL displayintelhexline;
        addr = addr + 10h;
    UNTIL(CARRY)
    OR(addr > (paraword(1) AND 0fff0h))ENDREPEAT;
    CALL disparray(.intelhexend);
    END;
ELSE function = err;
END write;

/***** MAIN PROGRAM *****/
init:CALL initvalue;
prompt:CALL newline;
prompt2:
DO WHILE 1;
    IF function = err
    THEN DO;
        CALL disparray(.error);
        CALL display(NAK);
    END;
    ELSE CALL display(ACK);
    CALL display('>');

```

```

CALL readline;
CALL findfunction;
IF function <> 0
THEN DO CASE function - 'a';
; /* @ error */
CALL assemble; /* A assemble */
CALL breakpoint; /* B breakpoint */
CALL clearbreakpoint; /* C clearbreakpoint */
CALL dump; /* D dump */
CALL enter; /* E enter */
CALL fill; /* F fill */
CALL goccommand; /* G go */
DO;
CALL disparray(.help);
CALL newline;
END; /* H help menu */
CALL inputcommand; /* I input */
CALL j$menu; /* J menu */
CALL k$menu; /* K menu */
CALL load; /* L load */
function = err; /* M error */
function = err; /* N error */
CALL outputcommand; /* O output */
function = err; /* P error */
function = err; /* Q error */
CALL register; /* R register */
function = err; /* S error */
CALL tracecommand; /* T trace */
CALL unassemble; /* U unassemble */
function = err; /* V error */
CALL write; /* W write */
CALL resetprocessor; /* X reset */
CALL y$menu; /* Y menu */
CALL z$menu; /* Z menu */
END;
END main8085;

```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย


```
table:D0;
$NOLIST
  DECLARE
    mnemonic(82) STRUCTURE(char(4) BYTE) PUBLIC DATA(
$INCLUDE(mnemonic.t85)
),
    operand(20) STRUCTURE(char(3) BYTE) PUBLIC DATA(
$INCLUDE(operand.t85)
),
    insopc(246) BYTE PUBLIC DATA(
$INCLUDE(insopc.t85)
),
    insmne(246) BYTE PUBLIC DATA(
$INCLUDE(insmne.t85)
),
    insopr(246) STRUCTURE(opr(2) BYTE) PUBLIC DATA(
$INCLUDE(insopr.t85)
),
    insbyte(246) BYTE PUBLIC DATA(
$INCLUDE(insbyte.t85)
),
    mneinsptr(82) BYTE PUBLIC DATA(
$INCLUDE(mne-ins.t85)
),
    opcinsptr(256) BYTE PUBLIC DATA(
$INCLUDE(opc-ins.t85)
);
END table;
```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/*0CEH,*2,/* ACI 00 */
/*08FH,*1,/* ADC A */
/*088H,*1,/* ADC B */
/*089H,*1,/* ADC C */
/*08AH,*1,/* ADC D */
/*08BH,*1,/* ADC E */
/*08CH,*1,/* ADC H */
/*08DH,*1,/* ADC L */
/*08EH,*1,/* ADC M */
/*087H,*1,/* ADD A */
/*080H,*1,/* ADD B */
/*081H,*1,/* ADD C */
/*082H,*1,/* ADD D */
/*083H,*1,/* ADD E */
/*084H,*1,/* ADD H */
/*085H,*1,/* ADD L */
/*086H,*1,/* ADD M */
/*0C6H,*2,/* ADI 00 */
/*0A7H,*1,/* ANA A */
/*0A0H,*1,/* ANA B */
/*0A1H,*1,/* ANA C */
/*0A2H,*1,/* ANA D */
/*0A3H,*1,/* ANA E */
/*0A4H,*1,/* ANA H */
/*0A5H,*1,/* ANA L */
/*0A6H,*1,/* ANA M */
/*0E6H,*2,/* ANI 00 */
/*0CDH,*3,/* CALL 0000 */
/*0DCH,*3,/* CC 0000 */
/*0FCH,*3,/* CM 0000 */
/*02FH,*1,/* CMA */
/*03FH,*1,/* CMC */
/*0BFH,*1,/* CMP A */
/*088H,*1,/* CMP B */
/*089H,*1,/* CMP C */
/*0BAH,*1,/* CMP D */
/*0BBH,*1,/* CMP E */
/*0BCH,*1,/* CMP H */
/*0BDH,*1,/* CMP L */
/*0BEH,*1,/* CMP M */
/*0D4H,*3,/* CNC 0000 */
/*0C4H,*3,/* CNZ 0000 */
/*0F4H,*3,/* CP 0000 */
/*0ECH,*3,/* CPE 0000 */
/*0FEH,*2,/* CPI 00 */
/*0E4H,*3,/* CPO 0000 */
/*0CCH,*3,/* CZ 0000 */
/*027H,*1,/* DAA */
/*009H,*1,/* DAD B */
/*019H,*1,/* DAD D */
/*029H,*1,/* DAD H */
/*039H,*1,/* DAD SP */
/*03DH,*1,/* DCR A */
/*005H,*1,/* DCR B */
/*00DH,*1,/* DCR C */
/*015H,*1,/* DCR D */
/*01DH,*1,/* DCR E */
/*025H,*1,/* DCR H */
/*02DH,*1,/* DCR L */
/*035H,*1,/* DCR M */
/*00BH,*1,/* DCX B */
/*01BH,*1,/* DCX D */
/*02BH,*1,/* DCX H */
/*03BH,*1,/* DCX SP */
/*0F3H,*1,/* DI */
/*0FBH,*1,/* EI */
/*076H,*1,/* HLT */
/*0DBH,*2,/* IN 00 */
/*03CH,*1,/* INR A */
/*004H,*1,/* INR B */
/*00CH,*1,/* INR C */
/*014H,*1,/* INR D */
/*01CH,*1,/* INR E */
/*024H,*1,/* INR H */
/*02CH,*1,/* INR L */
/*034H,*1,/* INR M */
/*003H,*1,/* INX B */
/*013H,*1,/* INX D */
/*023H,*1,/* INX H */
/*033H,*1,/* INX SP */
/*0DAH,*3,/* JC 0000 */
/*0FAH,*3,/* JM 0000 */
/*0C3H,*3,/* JMP 0000 */
/*0D2H,*3,/* JNC 0000 */
/*0C2H,*3,/* JNZ 0000 */
/*0F2H,*3,/* JP 0000 */
/*0EAH,*3,/* JPE 0000 */
/*0E2H,*3,/* JPO 0000 */
/*0CAH,*3,/* JZ 0000 */
/*03AH,*3,/* LDA 0000 */

```



มหาวิทยาลัยราชภัฏวชิรเวศน์
 วิทยาลัยพยาบาล
 วิทยาลัยการแพทย์
 วิทยาลัยการสาธารณสุข

```

/*00AH,*1,/* LDAX B */
/*01AH,*1,/* LDAX D */
/*02AH,*3,/* LHLD 0000 */
/*001H,*3,/* LXI B,0000 */
/*011H,*3,/* LXI D,0000 */
/*021H,*3,/* LXI H,0000 */
/*031H,*3,/* LXI SP,0000 */
/*07FH,*1,/* MOV A,A */
/*078H,*1,/* MOV A,B */
/*079H,*1,/* MOV A,C */
/*07AH,*1,/* MOV A,D */
/*07BH,*1,/* MOV A,E */
/*07CH,*1,/* MOV A,H */
/*07DH,*1,/* MOV A,L */
/*07EH,*1,/* MOV A,M */
/*047H,*1,/* MOV B,A */
/*040H,*1,/* MOV B,B */
/*041H,*1,/* MOV B,C */
/*042H,*1,/* MOV B,D */
/*043H,*1,/* MOV B,E */
/*044H,*1,/* MOV B,H */
/*045H,*1,/* MOV B,L */
/*046H,*1,/* MOV B,M */
/*04FH,*1,/* MOV C,A */
/*048H,*1,/* MOV C,B */
/*049H,*1,/* MOV C,C */
/*04AH,*1,/* MOV C,D */
/*04BH,*1,/* MOV C,E */
/*04CH,*1,/* MOV C,H */
/*04DH,*1,/* MOV C,L */
/*04EH,*1,/* MOV C,M */
/*057H,*1,/* MOV D,A */
/*050H,*1,/* MOV D,B */
/*051H,*1,/* MOV D,C */
/*052H,*1,/* MOV D,D */
/*053H,*1,/* MOV D,E */
/*054H,*1,/* MOV D,H */
/*055H,*1,/* MOV D,L */
/*056H,*1,/* MOV D,M */
/*05FH,*1,/* MOV E,A */
/*058H,*1,/* MOV E,B */
/*059H,*1,/* MOV E,C */
/*05AH,*1,/* MOV E,D */
/*05BH,*1,/* MOV E,E */
/*05CH,*1,/* MOV E,H */
/*05DH,*1,/* MOV E,L */
/*05EH,*1,/* MOV E,M */
/*067H,*1,/* MOV H,A */
/*060H,*1,/* MOV H,B */
/*061H,*1,/* MOV H,C */
/*062H,*1,/* MOV H,D */
/*063H,*1,/* MOV H,E */
/*064H,*1,/* MOV H,H */
/*065H,*1,/* MOV H,L */
/*066H,*1,/* MOV H,M */
/*06FH,*1,/* MOV L,A */
/*068H,*1,/* MOV L,B */
/*069H,*1,/* MOV L,C */
/*06AH,*1,/* MOV L,D */
/*06BH,*1,/* MOV L,E */
/*06CH,*1,/* MOV L,H */
/*06DH,*1,/* MOV L,L */
/*06EH,*1,/* MOV L,M */
/*077H,*1,/* MOV M,A */
/*070H,*1,/* MOV M,B */
/*071H,*1,/* MOV M,C */
/*072H,*1,/* MOV M,D */
/*073H,*1,/* MOV M,E */
/*074H,*1,/* MOV M,H */
/*075H,*1,/* MOV M,L */
/*03EH,*2,/* MVI A,00 */
/*006H,*2,/* MVI B,00 */
/*00EH,*2,/* MVI C,00 */
/*016H,*2,/* MVI D,00 */
/*01EH,*2,/* MVI E,00 */
/*026H,*2,/* MVI H,00 */
/*02EH,*2,/* MVI L,00 */
/*036H,*2,/* MVI M,00 */
/*000H,*1,/* NOP */
/*0B7H,*1,/* ORA A */
/*0B0H,*1,/* ORA B */
/*0B1H,*1,/* ORA C */
/*0B2H,*1,/* ORA D */
/*0B3H,*1,/* ORA E */
/*0B4H,*1,/* ORA H */
/*0B5H,*1,/* ORA L */
/*0B6H,*1,/* ORA M */
/*0F6H,*2,/* ORI 00 */
/*0D3H,*2,/* OUT 00 */
/*0E9H,*1,/* PCHL */

```



วิทยาลัยพยาบาล

กรุงเทพมหานครมหาวิทยาลัย

```

/*0C1H,*/1,/* POP B */
/*0D1H,*/1,/* POP D */
/*0E1H,*/1,/* POP H */
/*0F1H,*/1,/* POP PSW */
/*0C5H,*/1,/* PUSH B */
/*0D5H,*/1,/* PUSH D */
/*0E5H,*/1,/* PUSH H */
/*0F5H,*/1,/* PUSH PSW */
/*017H,*/1,/* RAL */
/*01FH,*/1,/* RAR */
/*0D8H,*/1,/* RC */
/*0C9H,*/1,/* RET */
/*020H,*/1,/* RIM */
/*007H,*/1,/* RLC */
/*0F8H,*/1,/* RM */
/*0D0H,*/1,/* RNC */
/*0C0H,*/1,/* RNZ */
/*0F0H,*/1,/* RP */
/*0E8H,*/1,/* RPE */
/*0E0H,*/1,/* RPO */
/*00FH,*/1,/* RRC */
/*0C7H,*/1,/* RST 0 */
/*0CFH,*/1,/* RST 1 */
/*0D7H,*/1,/* RST 2 */
/*0DFH,*/1,/* RST 3 */
/*0E7H,*/1,/* RST 4 */
/*0EFH,*/1,/* RST 5 */
/*0F7H,*/1,/* RST 6 */
/*0FFH,*/1,/* RST 7 */
/*0C8H,*/1,/* RZ */
/*09FH,*/1,/* SBB A */
/*098H,*/1,/* SBB B */
/*099H,*/1,/* SBB C */
/*09AH,*/1,/* SBB D */
/*09BH,*/1,/* SBB E */
/*09CH,*/1,/* SBB H */
/*09DH,*/1,/* SBB L */
/*09EH,*/1,/* SBB M */
/*0DEH,*/2,/* SBI 00 */
/*022H,*/3,/* SHLD 0000 */
/*030H,*/1,/* SIM */
/*0F9H,*/1,/* SPHL */
/*032H,*/3,/* STA 0000 */
/*002H,*/1,/* STAX B */
/*012H,*/1,/* STAX D */
/*037H,*/1,/* STC */
/*097H,*/1,/* SUB A */
/*090H,*/1,/* SUB B */
/*091H,*/1,/* SUB C */
/*092H,*/1,/* SUB D */
/*093H,*/1,/* SUB E */
/*094H,*/1,/* SUB H */
/*095H,*/1,/* SUB L */
/*096H,*/1,/* SUB M */
/*0D6H,*/2,/* SUI 00 */
/*0EBH,*/1,/* XCHG */
/*0AFH,*/1,/* XRA A */
/*0A8H,*/1,/* XRA B */
/*0A9H,*/1,/* XRA C */
/*0AAH,*/1,/* XRA D */
/*0ABH,*/1,/* XRA E */
/*0ACH,*/1,/* XRA H */
/*0ADH,*/1,/* XRA L */
/*0AEH,*/1,/* XRA M */
/*0EEH,*/2,/* XRI 00 */
/*0E3H,*/1,/* XTHL */

```



ศูนย์วิทยทรัพยากร
 ภาควิชาวิศวกรรมมหาวิทาลัย

```

/* ACI */ 1, /* b */
/* ADC */ 2, /* A */
/* ADC */ 2, /* B */
/* ADC */ 2, /* C */
/* ADC */ 2, /* D */
/* ADC */ 2, /* E */
/* ADC */ 2, /* H */
/* ADC */ 2, /* L */
/* ADC */ 2, /* M */
/* ADD */ 3, /* A */
/* ADD */ 3, /* B */
/* ADD */ 3, /* C */
/* ADD */ 3, /* D */
/* ADD */ 3, /* E */
/* ADD */ 3, /* H */
/* ADD */ 3, /* L */
/* ADD */ 3, /* M */
/* ADI */ 4, /* b */
/* ANA */ 5, /* A */
/* ANA */ 5, /* B */
/* ANA */ 5, /* C */
/* ANA */ 5, /* D */
/* ANA */ 5, /* E */
/* ANA */ 5, /* H */
/* ANA */ 5, /* L */
/* ANA */ 5, /* M */
/* ANI */ 6, /* b */
/* CALL */ 7, /* a */
/* CC */ 8, /* a */
/* CM */ 9, /* a */
/* CMA */ 10, /* */
/* CMC */ 11, /* */
/* CMP */ 12, /* A */
/* CMP */ 12, /* B */
/* CMP */ 12, /* C */
/* CMP */ 12, /* D */
/* CMP */ 12, /* E */
/* CMP */ 12, /* H */
/* CMP */ 12, /* L */
/* CMP */ 12, /* M */
/* CNC */ 13, /* a */
/* CNZ */ 14, /* a */
/* CP */ 15, /* a */
/* CPE */ 16, /* a */
/* CPI */ 17, /* b */
/* CPO */ 18, /* a */
/* CZ */ 19, /* a */
/* DAA */ 20, /* */
/* DAD */ 21, /* B */
/* DAD */ 21, /* D */
/* DAD */ 21, /* H */
/* DAD */ 21, /* SP */
/* DCR */ 22, /* A */
/* DCR */ 22, /* B */
/* DCR */ 22, /* C */
/* DCR */ 22, /* D */
/* DCR */ 22, /* E */
/* DCR */ 22, /* H */
/* DCR */ 22, /* L */
/* DCR */ 22, /* M */
/* DCX */ 23, /* B */
/* DCX */ 23, /* D */
/* DCX */ 23, /* H */
/* DCX */ 23, /* SP */
/* DI */ 24, /* */
/* EI */ 25, /* */
/* HLT */ 26, /* */
/* IN */ 27, /* b */
/* INR */ 28, /* A */
/* INR */ 28, /* B */
/* INR */ 28, /* C */
/* INR */ 28, /* D */
/* INR */ 28, /* E */
/* INR */ 28, /* H */
/* INR */ 28, /* L */
/* INR */ 28, /* M */
/* INX */ 29, /* B */
/* INX */ 29, /* D */
/* INX */ 29, /* H */
/* INX */ 29, /* SP */
/* JC */ 30, /* a */
/* JM */ 31, /* a */
/* JMP */ 32, /* a */
/* JNC */ 33, /* a */
/* JNZ */ 34, /* a */
/* JP */ 35, /* a */
/* JPE */ 36, /* a */
/* JPO */ 37, /* a */
/* JZ */ 38, /* a */
/* LDA */ 39, /* a */

```



ศูนย์วิทยทรัพยากร
 ภาลงกรณ์มหาวิทยาลัย

```

/* LDAX */40, /* B */
/* LDAX */40, /* D */
/* LHLD */41, /* a */
/* LXI */42, /* B,a */
/* LXI */42, /* D,a */
/* LXI */42, /* H,a */
/* LXI */42, /* SP,a */
/* MOV */43, /* A,A */
/* MOV */43, /* A,B */
/* MOV */43, /* A,C */
/* MOV */43, /* A,D */
/* MOV */43, /* A,E */
/* MOV */43, /* A,H */
/* MOV */43, /* A,L */
/* MOV */43, /* A,M */
/* MOV */43, /* B,A */
/* MOV */43, /* B,B */
/* MOV */43, /* B,C */
/* MOV */43, /* B,D */
/* MOV */43, /* B,E */
/* MOV */43, /* B,H */
/* MOV */43, /* B,L */
/* MOV */43, /* B,M */
/* MOV */43, /* C,A */
/* MOV */43, /* C,B */
/* MOV */43, /* C,C */
/* MOV */43, /* C,D */
/* MOV */43, /* C,E */
/* MOV */43, /* C,H */
/* MOV */43, /* C,L */
/* MOV */43, /* C,M */
/* MOV */43, /* D,A */
/* MOV */43, /* D,B */
/* MOV */43, /* D,C */
/* MOV */43, /* D,D */
/* MOV */43, /* D,E */
/* MOV */43, /* D,H */
/* MOV */43, /* D,L */
/* MOV */43, /* D,M */
/* MOV */43, /* E,A */
/* MOV */43, /* E,B */
/* MOV */43, /* E,C */
/* MOV */43, /* E,D */
/* MOV */43, /* E,E */
/* MOV */43, /* E,H */
/* MOV */43, /* E,L */
/* MOV */43, /* E,M */
/* MOV */43, /* H,A */
/* MOV */43, /* H,B */
/* MOV */43, /* H,C */
/* MOV */43, /* H,D */
/* MOV */43, /* H,E */
/* MOV */43, /* H,H */
/* MOV */43, /* H,L */
/* MOV */43, /* H,M */
/* MOV */43, /* L,A */
/* MOV */43, /* L,B */
/* MOV */43, /* L,C */
/* MOV */43, /* L,D */
/* MOV */43, /* L,E */
/* MOV */43, /* L,H */
/* MOV */43, /* L,L */
/* MOV */43, /* L,M */
/* MOV */43, /* M,A */
/* MOV */43, /* M,B */
/* MOV */43, /* M,C */
/* MOV */43, /* M,D */
/* MOV */43, /* M,E */
/* MOV */43, /* M,H */
/* MOV */43, /* M,L */
/* MVI */44, /* A,b */
/* MVI */44, /* B,b */
/* MVI */44, /* C,b */
/* MVI */44, /* D,b */
/* MVI */44, /* E,b */
/* MVI */44, /* H,b */
/* MVI */44, /* L,b */
/* MVI */44, /* M,b */
/* NOP */45, /* */
/* ORA */46, /* A */
/* ORA */46, /* B */
/* ORA */46, /* C */
/* ORA */46, /* D */
/* ORA */46, /* E */
/* ORA */46, /* H */
/* ORA */46, /* L */
/* ORA */46, /* M */
/* ORI */47, /* b */
/* OUT */48, /* b */
/* PCHL */49, /* */

```



ศูนย์วิทยทรัพยากร
 วิทยาลัย

```

/* POP */50, /* B */
/* POP */50, /* D */
/* POP */50, /* H */
/* POP */50, /* PSW */
/* PUSH */51, /* B */
/* PUSH */51, /* D */
/* PUSH */51, /* H */
/* PUSH */51, /* PSW */
/* RAL */52, /* */
/* RAR */53, /* */
/* RC */54, /* */
/* RET */55, /* */
/* RIM */56, /* */
/* RLC */57, /* */
/* RM */58, /* */
/* RNC */59, /* */
/* RNZ */60, /* */
/* RP */61, /* */
/* RPE */62, /* */
/* RPO */63, /* */
/* RRC */64, /* */
/* RST */65, /* 0 */
/* RST */65, /* 1 */
/* RST */65, /* 2 */
/* RST */65, /* 3 */
/* RST */65, /* 4 */
/* RST */65, /* 5 */
/* RST */65, /* 6 */
/* RST */65, /* 7 */
/* RZ */66, /* */
/* SBB */67, /* A */
/* SBB */67, /* B */
/* SBB */67, /* C */
/* SBB */67, /* D */
/* SBB */67, /* E */
/* SBB */67, /* H */
/* SBB */67, /* L */
/* SBB */67, /* M */
/* SBI */68, /* b */
/* SHLD */69, /* a */
/* SIM */70, /* */
/* SPHL */71, /* */
/* STA */72, /* a */
/* STAX */73, /* B */
/* STAX */73, /* D */
/* STC */74, /* */
/* SUB */75, /* A */
/* SUB */75, /* B */
/* SUB */75, /* C */
/* SUB */75, /* D */
/* SUB */75, /* E */
/* SUB */75, /* H */
/* SUB */75, /* L */
/* SUB */75, /* M */
/* SUI */76, /* b */
/* XCHG */77, /* */
/* XRA */78, /* A */
/* XRA */78, /* B */
/* XRA */78, /* C */
/* XRA */78, /* D */
/* XRA */78, /* E */
/* XRA */78, /* H */
/* XRA */78, /* L */
/* XRA */78, /* M */
/* XRI */79, /* b */
/* XTHL */80, /* */

```



ศูนย์วิทยทรัพยากร

คลังกรรมมหาวิทาลัย

0CEH, /*	ACI	b	*/
08FH, /*	ADC	A	*/
088H, /*	ADC	B	*/
089H, /*	ADC	C	*/
08AH, /*	ADC	D	*/
08BH, /*	ADC	E	*/
08CH, /*	ADC	H	*/
08DH, /*	ADC	L	*/
08EH, /*	ADC	M	*/
087H, /*	ADD	A	*/
080H, /*	ADD	B	*/
081H, /*	ADD	C	*/
082H, /*	ADD	D	*/
083H, /*	ADD	E	*/
084H, /*	ADD	H	*/
085H, /*	ADD	L	*/
086H, /*	ADD	M	*/
0C6H, /*	ADI	b	*/
0A7H, /*	ANA	A	*/
0A0H, /*	ANA	B	*/
0A1H, /*	ANA	C	*/
0A2H, /*	ANA	D	*/
0A3H, /*	ANA	E	*/
0A4H, /*	ANA	H	*/
0A5H, /*	ANA	L	*/
0A6H, /*	ANA	M	*/
0E6H, /*	ANI	b	*/
0CDH, /*	CALL	a	*/
0DCH, /*	CC	a	*/
0FCH, /*	CM	a	*/
02FH, /*	CMA	*/	*/
03FH, /*	CMC	*/	*/
0BFH, /*	CMP	A	*/
0B8H, /*	CMP	B	*/
0B9H, /*	CMP	C	*/
0BAH, /*	CMP	D	*/
0BBH, /*	CMP	E	*/
0BCH, /*	CMP	H	*/
0BDH, /*	CMP	L	*/
0BEH, /*	CMP	M	*/
0D4H, /*	CNC	a	*/
0C4H, /*	CNZ	a	*/
0F4H, /*	CP	a	*/
0ECH, /*	CPE	a	*/
0FEH, /*	CPI	b	*/
0E4H, /*	CPO	a	*/
0CCH, /*	CZ	a	*/
027H, /*	DAA	*/	*/
009H, /*	DAD	B	*/
019H, /*	DAD	D	*/
029H, /*	DAD	H	*/
039H, /*	DAD	SP	*/
03DH, /*	DCR	A	*/
005H, /*	DCR	B	*/
00DH, /*	DCR	C	*/
015H, /*	DCR	D	*/
01DH, /*	DCR	E	*/
025H, /*	DCR	H	*/
02DH, /*	DCR	L	*/
035H, /*	DCR	M	*/
00BH, /*	DCX	B	*/
01BH, /*	DCX	D	*/
02BH, /*	DCX	H	*/
03BH, /*	DCX	SP	*/
0F3H, /*	DI	*/	*/
0FBH, /*	EI	*/	*/
076H, /*	HLT	*/	*/
0DBH, /*	IN	b	*/
03CH, /*	INR	A	*/
004H, /*	INR	B	*/
00CH, /*	INR	C	*/
014H, /*	INR	D	*/
01CH, /*	INR	E	*/
024H, /*	INR	H	*/
02CH, /*	INR	L	*/
034H, /*	INR	M	*/
003H, /*	INX	B	*/
013H, /*	INX	D	*/
023H, /*	INX	H	*/
033H, /*	INX	SP	*/
0DAH, /*	JC	a	*/
0FAH, /*	JM	a	*/
0C3H, /*	JMP	a	*/
0D2H, /*	JNC	a	*/
0C2H, /*	JNZ	a	*/
0F2H, /*	JP	a	*/
0EAH, /*	JPE	a	*/
0E2H, /*	JPO	a	*/
0CAH, /*	JZ	a	*/
03AH, /*	LDA	a	*/



ศูนย์วิทยทรัพยากร
 ภาลงกรณ์มหาวิทยาลัย

00AH, /*	LDAX	B */
01AH, /*	LDAX	D */
02AH, /*	LHLD	a */
001H, /*	LXI	B,a */
011H, /*	LXI	D,a */
021H, /*	LXI	H,a */
031H, /*	LXI	SP,a */
07FH, /*	MOV	A,A */
078H, /*	MOV	A,B */
079H, /*	MOV	A,C */
07AH, /*	MOV	A,D */
07BH, /*	MOV	A,E */
07CH, /*	MOV	A,H */
07DH, /*	MOV	A,L */
07EH, /*	MOV	A,M */
047H, /*	MOV	B,A */
040H, /*	MOV	B,B */
041H, /*	MOV	B,C */
042H, /*	MOV	B,D */
043H, /*	MOV	B,E */
044H, /*	MOV	B,H */
045H, /*	MOV	B,L */
046H, /*	MOV	B,M */
04FH, /*	MOV	C,A */
048H, /*	MOV	C,B */
049H, /*	MOV	C,C */
04AH, /*	MOV	C,D */
04BH, /*	MOV	C,E */
04CH, /*	MOV	C,H */
04DH, /*	MOV	C,L */
04EH, /*	MOV	C,M */
057H, /*	MOV	D,A */
050H, /*	MOV	D,B */
051H, /*	MOV	D,C */
052H, /*	MOV	D,D */
053H, /*	MOV	D,E */
054H, /*	MOV	D,H */
055H, /*	MOV	D,L */
056H, /*	MOV	D,M */
05FH, /*	MOV	E,A */
058H, /*	MOV	E,B */
059H, /*	MOV	E,C */
05AH, /*	MOV	E,D */
05BH, /*	MOV	E,E */
05CH, /*	MOV	E,H */
05DH, /*	MOV	E,L */
05EH, /*	MOV	E,M */
067H, /*	MOV	H,A */
060H, /*	MOV	H,B */
061H, /*	MOV	H,C */
062H, /*	MOV	H,D */
063H, /*	MOV	H,E */
064H, /*	MOV	H,H */
065H, /*	MOV	H,L */
066H, /*	MOV	H,M */
06FH, /*	MOV	L,A */
068H, /*	MOV	L,B */
069H, /*	MOV	L,C */
06AH, /*	MOV	L,D */
06BH, /*	MOV	L,E */
06CH, /*	MOV	L,H */
06DH, /*	MOV	L,L */
06EH, /*	MOV	L,M */
077H, /*	MOV	M,A */
070H, /*	MOV	M,B */
071H, /*	MOV	M,C */
072H, /*	MOV	M,D */
073H, /*	MOV	M,E */
074H, /*	MOV	M,H */
075H, /*	MOV	M,L */
03EH, /*	MVI	A,b */
006H, /*	MVI	B,b */
00EH, /*	MVI	C,b */
016H, /*	MVI	D,b */
01EH, /*	MVI	E,b */
026H, /*	MVI	H,b */
02EH, /*	MVI	L,b */
036H, /*	MVI	M,b */
000H, /*	NOP	*/
0B7H, /*	ORA	A */
0B0H, /*	ORA	B */
0B1H, /*	ORA	C */
0B2H, /*	ORA	D */
0B3H, /*	ORA	E */
0B4H, /*	ORA	H */
0B5H, /*	ORA	L */
0B6H, /*	ORA	M */
0F6H, /*	ORI	b */
0D3H, /*	OUT	b */
0E9H, /*	PCHL	*/



ศูนย์วิทยทรัพยากร

คลังกรรมมหาวิทาลัย

0C1H,	/*	POP	B	*/
0D1H,	/*	POP	D	*/
0E1H,	/*	POP	H	*/
0F1H,	/*	POP	PSW	*/
0C5H,	/*	PUSH	B	*/
0D5H,	/*	PUSH	D	*/
0E5H,	/*	PUSH	H	*/
0F5H,	/*	PUSH	PSW	*/
017H,	/*	RAL		*/
01FH,	/*	RAR		*/
0D8H,	/*	RC		*/
0C9H,	/*	RET		*/
020H,	/*	RIM		*/
007H,	/*	RLC		*/
0F8H,	/*	RM		*/
0D0H,	/*	RNC		*/
0C0H,	/*	RNZ		*/
0F0H,	/*	RP		*/
0E8H,	/*	RPE		*/
0E0H,	/*	RPO		*/
00FH,	/*	RRC		*/
0C7H,	/*	RST	0	*/
0CFH,	/*	RST	1	*/
0D7H,	/*	RST	2	*/
0DFH,	/*	RST	3	*/
0E7H,	/*	RST	4	*/
0EFH,	/*	RST	5	*/
0F7H,	/*	RST	6	*/
0FFH,	/*	RST	7	*/
0C8H,	/*	RZ		*/
09FH,	/*	SBB	A	*/
098H,	/*	SBB	B	*/
099H,	/*	SBB	C	*/
09AH,	/*	SBB	D	*/
09BH,	/*	SBB	E	*/
09CH,	/*	SBB	H	*/
09DH,	/*	SBB	L	*/
09EH,	/*	SBB	M	*/
0DEH,	/*	SBI	b	*/
022H,	/*	SHLD	a	*/
030H,	/*	SIM		*/
0F9H,	/*	SPHL		*/
032H,	/*	STA	a	*/
002H,	/*	STAX	B	*/
012H,	/*	STAX	D	*/
037H,	/*	STC		*/
097H,	/*	SUB	A	*/
090H,	/*	SUB	B	*/
091H,	/*	SUB	C	*/
092H,	/*	SUB	D	*/
093H,	/*	SUB	E	*/
094H,	/*	SUB	H	*/
095H,	/*	SUB	L	*/
096H,	/*	SUB	M	*/
0D6H,	/*	SUI	b	*/
0EBH,	/*	XCHG		*/
0AFH,	/*	XRA	A	*/
0A8H,	/*	XRA	B	*/
0A9H,	/*	XRA	C	*/
0AAH,	/*	XRA	D	*/
0ABH,	/*	XRA	E	*/
0ACH,	/*	XRA	H	*/
0ADH,	/*	XRA	L	*/
0AEH,	/*	XRA	M	*/
0EEH,	/*	XRI	b	*/
0E3H,	/*	XTHL		*/



ศูนย์วิทยทรัพยากร
 ภาลงกรณ์มหาวิทยาลัย

```

/* ACI b */20,0,
/* ADC A */9,0,
/* ADC B */10,0,
/* ADC C */11,0,
/* ADC D */12,0,
/* ADC E */13,0,
/* ADC H */14,0,
/* ADC L */15,0,
/* ADC M */16,0,
/* ADD A */9,0,
/* ADD B */10,0,
/* ADD C */11,0,
/* ADD D */12,0,
/* ADD E */13,0,
/* ADD H */14,0,
/* ADD L */15,0,
/* ADD M */16,0,
/* ADI b */20,0,
/* ANA A */9,0,
/* ANA B */10,0,
/* ANA C */11,0,
/* ANA D */12,0,
/* ANA E */13,0,
/* ANA H */14,0,
/* ANA L */15,0,
/* ANA M */16,0,
/* ANI b */20,0,
/* CALL a */19,0,
/* CC a */19,0,
/* CM a */19,0,
/* CMA */0,0,
/* CMC */0,0,
/* CMP A */9,0,
/* CMP B */10,0,
/* CMP C */11,0,
/* CMP D */12,0,
/* CMP E */13,0,
/* CMP H */14,0,
/* CMP L */15,0,
/* CMP M */16,0,
/* CNC a */19,0,
/* CNZ a */19,0,
/* CP a */19,0,
/* CPE a */19,0,
/* CPI b */20,0,
/* CPO a */19,0,
/* CZ a */19,0,
/* DAA */0,0,
/* DAD B */10,0,
/* DAD D */12,0,
/* DAD H */14,0,
/* DAD SP */18,0,
/* DCR A */9,0,
/* DCR B */10,0,
/* DCR C */11,0,
/* DCR D */12,0,
/* DCR E */13,0,
/* DCR H */14,0,
/* DCR L */15,0,
/* DCR M */16,0,
/* DCX B */10,0,
/* DCX D */12,0,
/* DCX H */14,0,
/* DCX SP */18,0,
/* DI */0,0,
/* EI */0,0,
/* HLT */0,0,
/* IN b */20,0,
/* INR A */9,0,
/* INR B */10,0,
/* INR C */11,0,
/* INR D */12,0,
/* INR E */13,0,
/* INR H */14,0,
/* INR L */15,0,
/* INR M */16,0,
/* INX B */10,0,
/* INX D */12,0,
/* INX H */14,0,
/* INX SP */18,0,
/* JC a */19,0,
/* JM a */19,0,
/* JMP a */19,0,
/* JNC a */19,0,
/* JNZ a */19,0,
/* JP a */19,0,
/* JPE a */19,0,
/* JPO a */19,0,
/* JZ a */19,0,
/* LDA a */19,0,

```



ศูนย์วิทยทรัพยากร

ฟาลงกรณ์มหาวิทยาลัย

```

/* LDAX B */10,0,
/* LDAX D */12,0,
/* LHL a */19,0,
/* LXI B, */10, /* a */19,
/* LXI D, */12, /* a */19,
/* LXI H, */14, /* a */19,
/* LXI SP, */18, /* a */19,
/* MOV A, */9, /* A */9,
/* MOV A, */9, /* B */10,
/* MOV A, */9, /* C */11,
/* MOV A, */9, /* D */12,
/* MOV A, */9, /* E */13,
/* MOV A, */9, /* H */14,
/* MOV A, */9, /* L */15,
/* MOV A, */9, /* M */16,
/* MOV B, */10, /* A */9,
/* MOV B, */10, /* B */10,
/* MOV B, */10, /* C */11,
/* MOV B, */10, /* D */12,
/* MOV B, */10, /* E */13,
/* MOV B, */10, /* H */14,
/* MOV B, */10, /* L */15,
/* MOV B, */10, /* M */16,
/* MOV C, */11, /* A */9,
/* MOV C, */11, /* B */10,
/* MOV C, */11, /* C */11,
/* MOV C, */11, /* D */12,
/* MOV C, */11, /* E */13,
/* MOV C, */11, /* H */14,
/* MOV C, */11, /* L */15,
/* MOV C, */11, /* M */16,
/* MOV D, */12, /* A */9,
/* MOV D, */12, /* B */10,
/* MOV D, */12, /* C */11,
/* MOV D, */12, /* D */12,
/* MOV D, */12, /* E */13,
/* MOV D, */12, /* H */14,
/* MOV D, */12, /* L */15,
/* MOV D, */12, /* M */16,
/* MOV E, */13, /* A */9,
/* MOV E, */13, /* B */10,
/* MOV E, */13, /* C */11,
/* MOV E, */13, /* D */12,
/* MOV E, */13, /* E */13,
/* MOV E, */13, /* H */14,
/* MOV E, */13, /* L */15,
/* MOV E, */13, /* M */16,
/* MOV H, */14, /* A */9,
/* MOV H, */14, /* B */10,
/* MOV H, */14, /* C */11,
/* MOV H, */14, /* D */12,
/* MOV H, */14, /* E */13,
/* MOV H, */14, /* H */14,
/* MOV H, */14, /* L */15,
/* MOV H, */14, /* M */16,
/* MOV L, */15, /* A */9,
/* MOV L, */15, /* B */10,
/* MOV L, */15, /* C */11,
/* MOV L, */15, /* D */12,
/* MOV L, */15, /* E */13,
/* MOV L, */15, /* H */14,
/* MOV L, */15, /* L */15,
/* MOV L, */15, /* M */16,
/* MOV M, */16, /* A */9,
/* MOV M, */16, /* B */10,
/* MOV M, */16, /* C */11,
/* MOV M, */16, /* D */12,
/* MOV M, */16, /* E */13,
/* MOV M, */16, /* H */14,
/* MOV M, */16, /* L */15,
/* MVI A, */9, /* b */20,
/* MVI B, */10, /* b */20,
/* MVI C, */11, /* b */20,
/* MVI D, */12, /* b */20,
/* MVI E, */13, /* b */20,
/* MVI H, */14, /* b */20,
/* MVI L, */15, /* b */20,
/* MVI M, */16, /* b */20,
/* NOP */0,0,
/* ORA A */9,0,
/* ORA B */10,0,
/* ORA C */11,0,
/* ORA D */12,0,
/* ORA E */13,0,
/* ORA H */14,0,
/* ORA L */15,0,
/* ORA M */16,0,
/* ORI b */20,0,
/* OUT b */20,0,
/* PCHL */0,0,

```



วิทยาลัยพยาบาล

นครณ์มหาวิทยาลัย

```

/* POP      B */10,0,
/* POP      D */12,0,
/* POP      H */14,0,
/* POP      PSW */17,0,
/* PUSH     B */10,0,
/* PUSH     D */12,0,
/* PUSH     H */14,0,
/* PUSH     PSW */17,0,
/* RAL      */0,0,
/* RAR      */0,0,
/* RC       */0,0,
/* RET      */0,0,
/* RIM      */0,0,
/* RLC      */0,0,
/* RM       */0,0,
/* RNC      */0,0,
/* RNZ      */0,0,
/* RP       */0,0,
/* RPE      */0,0,
/* RPO      */0,0,
/* RRC      */0,0,
/* RST      0 */1,0,
/* RST      1 */2,0,
/* RST      2 */3,0,
/* RST      3 */4,0,
/* RST      4 */5,0,
/* RST      5 */6,0,
/* RST      6 */7,0,
/* RST      7 */8,0,
/* RZ       */0,0,
/* SBB      A */9,0,
/* SBB      B */10,0,
/* SBB      C */11,0,
/* SBB      D */12,0,
/* SBB      E */13,0,
/* SBB      H */14,0,
/* SBB      L */15,0,
/* SBB      M */16,0,
/* SBI      b */20,0,
/* SHLD     a */19,0,
/* SIM      */0,0,
/* SPHL     */0,0,
/* STA      a */19,0,
/* STAX     B */10,0,
/* STAX     D */12,0,
/* STC      */0,0,
/* SUB      A */9,0,
/* SUB      B */10,0,
/* SUB      C */11,0,
/* SUB      D */12,0,
/* SUB      E */13,0,
/* SUB      H */14,0,
/* SUB      L */15,0,
/* SUB      M */16,0,
/* SUI      b */20,0,
/* XCHG     */0,0,
/* XRA      A */9,0,
/* XRA      B */10,0,
/* XRA      C */11,0,
/* XRA      D */12,0,
/* XRA      E */13,0,
/* XRA      H */14,0,
/* XRA      L */15,0,
/* XRA      M */16,0,
/* XRI      b */20,0,
/* XTHL     */0,0,

```



ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

/* 1 ACI */ 1
/* 2 ADC */ 2
/* 3 ADD */ 10
/* 4 ADI */ 18
/* 5 ANA */ 19
/* 6 ANI */ 27
/* 7 CALL */ 28
/* 8 CC */ 29
/* 9 CM */ 30
/* 10 CMA */ 31
/* 11 CMC */ 32
/* 12 CMP */ 33
/* 13 CNC */ 41
/* 14 CNZ */ 42
/* 15 CP */ 43
/* 16 CPE */ 44
/* 17 CPI */ 45
/* 18 CPO */ 46
/* 19 CZ */ 47
/* 20 DAA */ 48
/* 21 DAD */ 49
/* 22 DCR */ 53
/* 23 DCX */ 61
/* 24 DI */ 65
/* 25 EI */ 66
/* 26 HLT */ 67
/* 27 IN */ 68
/* 28 INR */ 69
/* 29 INX */ 77
/* 30 JC */ 81
/* 31 JM */ 82
/* 32 JMP */ 83
/* 33 JNC */ 84
/* 34 JNZ */ 85
/* 35 JP */ 86
/* 36 JPE */ 87
/* 37 JPO */ 88
/* 38 JZ */ 89
/* 39 LDA */ 90
/* 40 LDAX */ 91
/* 41 LHLD */ 93
/* 42 LXI */ 94
/* 43 MOV */ 98
/* 44 MVI */ 161
/* 45 NOP */ 169
/* 46 ORA */ 170
/* 47 ORI */ 178
/* 48 OUT */ 179
/* 49 PCHL */ 180
/* 50 POP */ 181
/* 51 PUSH */ 185
/* 52 RAL */ 189
/* 53 RAR */ 190
/* 54 RC */ 191
/* 55 RET */ 192
/* 56 RIM */ 193
/* 57 RLC */ 194
/* 58 RM */ 195
/* 59 RNC */ 196
/* 60 RNZ */ 197
/* 61 RP */ 198
/* 62 RPE */ 199
/* 63 RPO */ 200
/* 64 RRC */ 201
/* 65 RST */ 202
/* 66 RZ */ 210
/* 67 SBB */ 211
/* 68 SBI */ 219
/* 69 SHLD */ 220
/* 70 SIM */ 221
/* 71 SPHL */ 222
/* 72 STA */ 223
/* 73 STAX */ 224
/* 74 STC */ 226
/* 75 SUB */ 227
/* 76 SUI */ 235
/* 77 XCHG */ 236
/* 78 XRA */ 237
/* 79 XRI */ 245
/* 80 XTHL */ 246
/* 81 DB */ 0
/* 82 DW */ 0

```



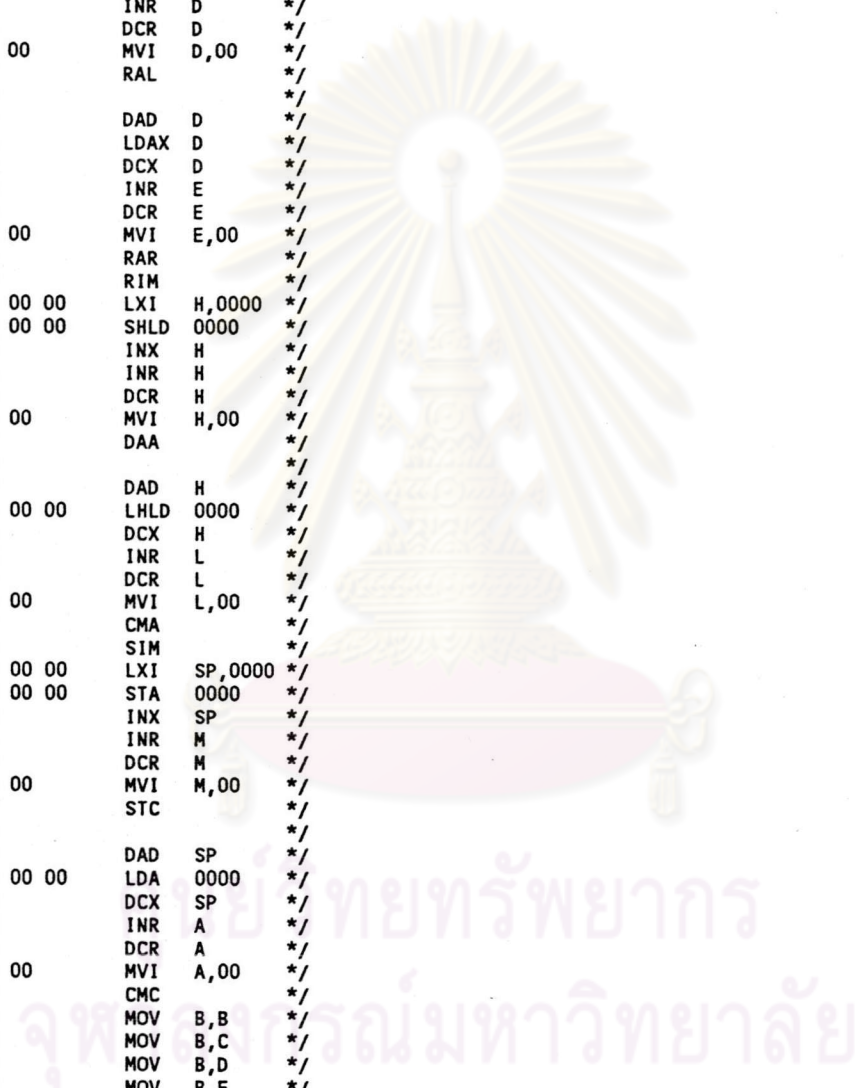
ศูนย์วิทยทรัพยากร
 ภาลงกรณ์มหาวิทยาลัย

'ACI'
'ADC'
'ADD'
'ADI'
'ANA'
'ANI'
'CALL'
'CC'
'CM'
'CMA'
'CMC'
'CMP'
'CNC'
'CNZ'
'CP'
'CPE'
'CPI'
'CPO'
'CZ'
'DAA'
'DAD'
'DCR'
'DCX'
'DI'
'EI'
'HLT'
'IN'
'INR'
'INX'
'JC'
'JM'
'JMP'
'JNC'
'JNZ'
'JP'
'JPE'
'JPO'
'JZ'
'LDA'
'LDAX'
'LHLD'
'LXI'
'MOV'
'MVI'
'NOP'
'ORA'
'ORI'
'OUT'
'PCHL'
'POP'
'PUSH'
'RAL'
'RAR'
'RC'
'RET'
'RIM'
'RLC'
'RM'
'RNC'
'RNZ'
'RP'
'RPE'
'RPO'
'RRC'
'RST'
'RZ'
'SBB'
'SBI'
'SHLD'
'SIM'
'SPHL'
'STA'
'STAX'
'STC'
'SUB'
'SUI'
'XCHG'
'XRA'
'XRI'
'XTHL'
'DB'
'DW'



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

169,/*	00	NOP		*/
94,/*	01 00 00	LXI	B,0000	*/
224,/*	02	STAX	B	*/
77,/*	03	INX	B	*/
70,/*	04	INR	B	*/
54,/*	05	DCR	B	*/
162,/*	06 00	MVI	B,00	*/
194,/*	07	RLC		*/
0,/*	08			*/
49,/*	09	DAD	B	*/
91,/*	0A	LDAX	B	*/
61,/*	0B	DCX	B	*/
71,/*	0C	INR	C	*/
55,/*	0D	DCR	C	*/
163,/*	0E 00	MVI	C,00	*/
201,/*	0F	RRC		*/
0,/*	10			*/
95,/*	11 00 00	LXI	D,0000	*/
225,/*	12	STAX	D	*/
78,/*	13	INX	D	*/
72,/*	14	INR	D	*/
56,/*	15	DCR	D	*/
164,/*	16 00	MVI	D,00	*/
189,/*	17	RAL		*/
0,/*	18			*/
50,/*	19	DAD	D	*/
92,/*	1A	LDAX	D	*/
62,/*	1B	DCX	D	*/
73,/*	1C	INR	E	*/
57,/*	1D	DCR	E	*/
165,/*	1E 00	MVI	E,00	*/
190,/*	1F	RAR		*/
193,/*	20	RIM		*/
96,/*	21 00 00	LXI	H,0000	*/
220,/*	22 00 00	SHLD	0000	*/
79,/*	23	INX	H	*/
74,/*	24	INR	H	*/
58,/*	25	DCR	H	*/
166,/*	26 00	MVI	H,00	*/
48,/*	27	DAA		*/
0,/*	28			*/
51,/*	29	DAD	H	*/
93,/*	2A 00 00	LHLD	0000	*/
63,/*	2B	DCX	H	*/
75,/*	2C	INR	L	*/
59,/*	2D	DCR	L	*/
167,/*	2E 00	MVI	L,00	*/
31,/*	2F	CMA		*/
221,/*	30	SIM		*/
97,/*	31 00 00	LXI	SP,0000	*/
223,/*	32 00 00	STA	0000	*/
80,/*	33	INX	SP	*/
76,/*	34	INR	M	*/
60,/*	35	DCR	M	*/
168,/*	36 00	MVI	M,00	*/
226,/*	37	STC		*/
0,/*	38			*/
52,/*	39	DAD	SP	*/
90,/*	3A 00 00	LDA	0000	*/
64,/*	3B	DCX	SP	*/
69,/*	3C	INR	A	*/
53,/*	3D	DCR	A	*/
161,/*	3E 00	MVI	A,00	*/
32,/*	3F	CMC		*/
107,/*	40	MOV	B,B	*/
108,/*	41	MOV	B,C	*/
109,/*	42	MOV	B,D	*/
110,/*	43	MOV	B,E	*/
111,/*	44	MOV	B,H	*/
112,/*	45	MOV	B,L	*/
113,/*	46	MOV	B,M	*/
106,/*	47	MOV	B,A	*/
115,/*	48	MOV	C,B	*/
116,/*	49	MOV	C,C	*/
117,/*	4A	MOV	C,D	*/
118,/*	4B	MOV	C,E	*/
119,/*	4C	MOV	C,H	*/
120,/*	4D	MOV	C,L	*/
121,/*	4E	MOV	C,M	*/
114,/*	4F	MOV	C,A	*/
123,/*	50	MOV	D,B	*/
124,/*	51	MOV	D,C	*/
125,/*	52	MOV	D,D	*/
126,/*	53	MOV	D,E	*/
127,/*	54	MOV	D,H	*/
128,/*	55	MOV	D,L	*/
129,/*	56	MOV	D,M	*/
122,/*	57	MOV	D,A	*/
131,/*	58	MOV	E,B	*/
132,/*	59	MOV	E,C	*/



จุฬาลงกรณ์มหาวิทยาลัย
 วิทยาลัยพยาบาล

133,/*	5A	MOV	E,D	*/
134,/*	5B	MOV	E,E	*/
135,/*	5C	MOV	E,H	*/
136,/*	5D	MOV	E,L	*/
137,/*	5E	MOV	E,M	*/
130,/*	5F	MOV	E,A	*/
139,/*	60	MOV	H,B	*/
140,/*	61	MOV	H,C	*/
141,/*	62	MOV	H,D	*/
142,/*	63	MOV	H,E	*/
143,/*	64	MOV	H,H	*/
144,/*	65	MOV	H,L	*/
145,/*	66	MOV	H,M	*/
138,/*	67	MOV	H,A	*/
147,/*	68	MOV	L,B	*/
148,/*	69	MOV	L,C	*/
149,/*	6A	MOV	L,D	*/
150,/*	6B	MOV	L,E	*/
151,/*	6C	MOV	L,H	*/
152,/*	6D	MOV	L,L	*/
153,/*	6E	MOV	L,M	*/
146,/*	6F	MOV	L,A	*/
155,/*	70	MOV	M,B	*/
156,/*	71	MOV	M,C	*/
157,/*	72	MOV	M,D	*/
158,/*	73	MOV	M,E	*/
159,/*	74	MOV	M,H	*/
160,/*	75	MOV	M,L	*/
67,/*	76	HLT		*/
154,/*	77	MOV	M,A	*/
99,/*	78	MOV	A,B	*/
100,/*	79	MOV	A,C	*/
101,/*	7A	MOV	A,D	*/
102,/*	7B	MOV	A,E	*/
103,/*	7C	MOV	A,H	*/
104,/*	7D	MOV	A,L	*/
105,/*	7E	MOV	A,M	*/
98,/*	7F	MOV	A,A	*/
11,/*	80	ADD	B	*/
12,/*	81	ADD	C	*/
13,/*	82	ADD	D	*/
14,/*	83	ADD	E	*/
15,/*	84	ADD	H	*/
16,/*	85	ADD	L	*/
17,/*	86	ADD	M	*/
10,/*	87	ADD	A	*/
3,/*	88	ADC	B	*/
4,/*	89	ADC	C	*/
5,/*	8A	ADC	D	*/
6,/*	8B	ADC	E	*/
7,/*	8C	ADC	H	*/
8,/*	8D	ADC	L	*/
9,/*	8E	ADC	M	*/
2,/*	8F	ADC	A	*/
228,/*	90	SUB	B	*/
229,/*	91	SUB	C	*/
230,/*	92	SUB	D	*/
231,/*	93	SUB	E	*/
232,/*	94	SUB	H	*/
233,/*	95	SUB	L	*/
234,/*	96	SUB	M	*/
227,/*	97	SUB	A	*/
212,/*	98	SBB	B	*/
213,/*	99	SBB	C	*/
214,/*	9A	SBB	D	*/
215,/*	9B	SBB	E	*/
216,/*	9C	SBB	H	*/
217,/*	9D	SBB	L	*/
218,/*	9E	SBB	M	*/
211,/*	9F	SBB	A	*/
20,/*	A0	ANA	B	*/
21,/*	A1	ANA	C	*/
22,/*	A2	ANA	D	*/
23,/*	A3	ANA	E	*/
24,/*	A4	ANA	H	*/
25,/*	A5	ANA	L	*/
26,/*	A6	ANA	M	*/
19,/*	A7	ANA	A	*/
238,/*	A8	XRA	B	*/
239,/*	A9	XRA	C	*/
240,/*	AA	XRA	D	*/
241,/*	AB	XRA	E	*/
242,/*	AC	XRA	H	*/
243,/*	AD	XRA	L	*/
244,/*	AE	XRA	M	*/
237,/*	AF	XRA	A	*/
171,/*	B0	ORA	B	*/
172,/*	B1	ORA	C	*/
173,/*	B2	ORA	D	*/
174,/*	B3	ORA	E	*/

175,/*	B4	ORA	H	*/
176,/*	B5	ORA	L	*/
177,/*	B6	ORA	M	*/
170,/*	B7	ORA	A	*/
34,/*	B8	CMP	B	*/
35,/*	B9	CMP	C	*/
36,/*	BA	CMP	D	*/
37,/*	BB	CMP	E	*/
38,/*	BC	CMP	H	*/
39,/*	BD	CMP	L	*/
40,/*	BE	CMP	M	*/
33,/*	BF	CMP	A	*/
197,/*	C0	RNZ		*/
181,/*	C1	POP	B	*/
85,/*	C2 00 00	JNZ	0000	*/
83,/*	C3 00 00	JMP	0000	*/
42,/*	C4 00 00	CNZ	0000	*/
185,/*	C5	PUSH	B	*/
18,/*	C6 00	ADI	00	*/
202,/*	C7	RST	0	*/
210,/*	C8	RZ		*/
192,/*	C9	RET		*/
89,/*	CA 00 00	JZ	0000	*/
0,/*	CB			*/
47,/*	CC 00 00	CZ	0000	*/
28,/*	CD 00 00	CALL	0000	*/
1,/*	CE 00	ACI	00	*/
203,/*	CF	RST	1	*/
196,/*	D0	RNC		*/
182,/*	D1	POP	D	*/
84,/*	D2 00 00	JNC	0000	*/
179,/*	D3 00	OUT	00	*/
41,/*	D4 00 00	CNC	0000	*/
186,/*	D5	PUSH	D	*/
235,/*	D6 00	SUI	00	*/
204,/*	D7	RST	2	*/
191,/*	D8	RC		*/
0,/*	D9			*/
81,/*	DA 00 00	JC	0000	*/
68,/*	DB 00	IN	00	*/
29,/*	DC 00 00	CC	0000	*/
0,/*	DD			*/
219,/*	DE 00	SBI	00	*/
205,/*	DF	RST	3	*/
200,/*	E0	RPO		*/
183,/*	E1	POP	H	*/
88,/*	E2 00 00	JPO	0000	*/
246,/*	E3	XTHL		*/
46,/*	E4 00 00	CPO	0000	*/
187,/*	E5	PUSH	H	*/
27,/*	E6 00	ANI	00	*/
206,/*	E7	RST	4	*/
199,/*	E8	RPE		*/
180,/*	E9	PCHL		*/
87,/*	EA 00 00	JPE	0000	*/
236,/*	EB	XCHG		*/
44,/*	EC 00 00	CPE	0000	*/
0,/*	ED			*/
245,/*	EE 00	XRI	00	*/
207,/*	EF	RST	5	*/
198,/*	F0	RP		*/
184,/*	F1	POP	PSW	*/
86,/*	F2 00 00	JP	0000	*/
65,/*	F3	DI		*/
43,/*	F4 00 00	CP	0000	*/
188,/*	F5	PUSH	PSW	*/
178,/*	F6 00	ORI	00	*/
208,/*	F7	RST	6	*/
195,/*	F8	RM		*/
222,/*	F9	SPHL		*/
82,/*	FA 00 00	JM	0000	*/
66,/*	FB	EI		*/
30,/*	FC 00 00	CM	0000	*/
0,/*	FD			*/
45,/*	FE 00	CPI	00	*/
209,/*	FF	RST	7	*/

- /* 1*/ '0
- /* 2*/ '1
- /* 3*/ '2
- /* 4*/ '3
- /* 5*/ '4
- /* 6*/ '5
- /* 7*/ '6
- /* 8*/ '7
- /* 9*/ 'A
- /*10*/ 'B
- /*11*/ 'C
- /*12*/ 'D
- /*13*/ 'E
- /*14*/ 'H
- /*15*/ 'L
- /*16*/ 'M
- /*17*/ 'PSW
- /*18*/ 'SP
- /*19*/ 'a
- /*20*/ 'b



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ประวัติผู้เขียน

นางสาว สุภันท์ หิรัญยพิสูทธิกุล เกิดวันที่ 7 มกราคม พ.ศ. 2510 ที่จังหวัด กรุงเทพมหานคร สำเร็จการศึกษาระดับปริญญาตรีวิศวกรรมศาสตรบัณฑิต สาขาอิเล็กทรอนิกส์ ภาควิชาวิศวกรรมไฟฟ้า จากคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าฯ เจ้าคุณทหารลาดกระบัง ในปีการศึกษา 2530 และเข้าศึกษาต่อในหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต ณ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย