



مدرسین تهران

۱- تدریس خصوصی تمامی دروس دانشگاهی در تمامی مقاطع

- مهندسی مکانیک
- مهندسی برق
- مهندسی عمران
- مهندسی کامپیوتر، صنایع و ...

۲- مشاوره علمی جهت انجام پروژه‌های صنعتی و بخش نرم‌افزاری پروژه‌های دانشجویی

۳- آموزش تمامی نرم‌افزارهای تخصصی

شماره تماس

۰۲۱-۷۷۴۹۹۹۲۵

۰۹۲۱-۲۰۲۸۲۹۵

آدرس سایت : www.ModaresineTehran.com

پست الکترونیک : ModaresineTehran@gmail.com

بسمه تعالی



سوالات امتحان آخر ترم درس ریزپردازنده کد: ۱۹۲۰ گروه کامپیوتر استاد فاتی
 دانشگاه آزاد اسلامی واحد تهران جنوب - ترم تابستان سال تحصیلی ۸۷-۸۶
 مدت امتحان: ۲ ساعت تاریخ امتحان: ۱۳۸۶/۱۱/۱۵ جزوه بسته و ماشین حساب لازم نیست

نام و نام خانوادگی:

تذکر ۱: برگه سوال، ضمیمه پاسخنامه خواهد بود. می‌توانید به برخی از سوالات در برگه سوال جواب دهید.

تذکر ۲: برای کاهش قلم خوردگی توصیه می‌شود با مداد پاسخ دهید.

تذکر ۳: نمره نهایی از ۱۱۰ نمره این امتحان و ۱۰ نمره کلاس مشخص خواهد شد.

۱- پایه‌های وقفه و عکس‌العمل میکرو را در ریزپردازنده‌های Z80 و 8086 توضیح دهید. (۱۰ نمره)

۲- روش‌های آدرس‌دهی دستگاه‌های جانبی Isolated و Memory Mapped را در مقایسه باهم توضیح دهید. (۱۰ نمره)

۳- دستورالعمل‌های زیر دارای چه ماشین‌سایکل‌هایی هستند آنها را به ترتیب اجرا ذکر کرده و بنویسید در هر کدام از سیکل ماشین‌ها دقیقاً چه کاری انجام می‌شود. (۱۵ نمره) RET و IN 80h

۴- با توجه به مدار ضمیمه به سوالات زیر با ذکر دلیل کافی پاسخ دهید. (هر کدام ۵ نمره = ۱۵ نمره)

a. محدوده و اندازه آدرس‌های هر یک از خروجی‌های دیکودرها را به طور مشخص بنویسید.

b. آدرس پورت‌های ۸۲۵۵ را به صورت دقیق مشخص کنید. (آدرس‌های تکراری مشخص شود)

c. کلمه برنامه ریزی ۸۲۵۵ را بدست آورده و دستورات برنامه ریزی متناسب را بنویسید.

۵- با توجه به نرم افزار پیوست (کنترل چهار راه ۴ زمانه) به سوالات زیر پاسخ دهید. (۳۰ نمره)

a. به سه پایه وقفه سه کلید وصل شده است عملکرد آنها را نسبت به هم توضیح دهید.

b. زمان چراغ‌های سبز، زرد و قرمز هر ورودی به چه ترتیبی مشخص و کنترل می‌شود.

c. روتین تاخیر B ثانیه خالی می‌باشد آنرا بطور کامل نوشته و زمان تاخیر آنرا برای B ثانیه تنظیم کنید. (f=10MHz)

d. ترجمه زبان ماشین ۴ خط ستاره‌دار را بطور کامل و دقیق بنویسید.

۶- قطعه برنامه‌ای (زیربرنامه‌ای) بنویسید که مقدار پارامتر ورودی در رجیستر B را که از ۰۰ تا ۹۹ می‌باشد را در منبای ۱۰ در دو عدد نمایشگر 7-Seg در آدرس 80h (یکان) و 81h (دهگان) بنویسد. (۲۰ نمره)

۷- شکل بلوکی یک میکروکنترلر نمونه را ترسیم کرده اجزا و قابلیت‌های آنرا توضیح دهید. (۱۰ نمره)

بسمه تعالی

سوالات امتحان آخر ترم درس ریزپردازنده کد: ۱۹۲۰ گروه کامپیوتر استاد فاتی

دانشگاه آزاد اسلامی واحد تهران جنوب - ترم تابستان سال تحصیلی ۸۷-۸۶

مدت امتحان: ۲ ساعت تاریخ امتحان: ۱۳۸۶/۱۱/۱۵ جزوه بسته و ماشین حساب لازم نیست



8085 Instruction Set

Mnemonic	Op	SZAPC	-s	Description	Notes	Mnemonic	Op	SZAPC	-s	Description	Notes
ACI n	ICE	*****	7	Add with Carry Immediate	A=A+n+CY	PUSH B	C5	-----	12	Push BC	-[SP]=BC
ADC r	8F	*****	4	Add with Carry	A=A+r+CY (21X)	PUSH D	D5	-----	12	Push DE	-[SP]=DE
ADC M	8E	*****	7	Add with Carry to Memory	A=A+[HL]+CY	PUSH H	E5	-----	12	Push HL	-[SP]=HL
ADD r	87	*****	4	Add	A=A+r (20X)	PUSH PSW	F5	-----	12	Push Processor Status Word	-[SP]=(PSW,A)
ADD M	86	*****	7	Add to Memory	A=A+[HL]	RAL	17	-----*	4	Rotate Accumulator Left	A=[CY,A]<-
ADI n	C6	*****	7	Add Immediate	A=A+n	RAR	1F	-----*	4	Rotate Accumulator Right	A=>[CY,A]
ANA r	A7	*****	4	AND Accumulator	A=A&r (24X)	RET	C9	-----	10	Return	PC=[SP]+
ANA M	A6	*****	7	AND Accumulator and Memory	A=A&[HL]	RC	D8	-----	6	Return on Carry	If CY=1 (12-s)
ANI n	E6	*****	7	AND Immediate	A=A&n	RIM	20	-----	4	Read Interrupt Mask	A=mask
CALL a	CD	-----	18	Call unconditional	[-(SP)]=PC, PC=a	RM	F8	-----	6	Return on Minus	If S=1 (12-s)
CC a	DC	-----	9	Call on Carry	If CY=1 (18-s)	RNC	D0	-----	6	Return on No Carry	If CY=0 (12-s)
CM a	FC	-----	9	Call on Minus	If S=1 (18-s)	RNZ	C0	-----	6	Return on No Zero	If Z=0 (12-s)
CMA	2F	-----	4	Complement Accumulator	A=~A	RF	F0	-----	6	Return on Plus	If S=0 (12-s)
CMC	3F	-----	4	Complement Carry	CY=~CY	RPE	E8	-----	6	Return on Parity Even	If P=1 (12-s)
CMP r	BE	*****	4	Compare	A-r (27X)	RPO	E0	-----	6	Return on Parity Odd	If P=0 (12-s)
CMP M	BD	*****	7	Compare with Memory	A-[HL]	RE	C8	-----	6	Return on Zero	If Z=1 (12-s)
CMC a	DA	-----	9	Call on No Carry	If CY=0 (18-s)	RLC	07	-----	4	Rotate Left Circular	A<-A
CNZ a	CA	-----	9	Call on No Zero	If Z=0 (18-s)	RRC	0F	-----*	4	Rotate Right Circular	A->A
CP a	FA	-----	9	Call on Plus	If S=0 (18-s)	RST z	C7	-----	12	Restart (3X7)	[-(SP)]=PC, PC=z
CPE a	EC	-----	9	Call on Parity Even	If P=1 (18-s)	SBB r	9F	*****	4	Subtract with Borrow	A=A-r-CY
CPI n	FE	*****	7	Compare Immediate	A-n	SBB M	9E	*****	7	Subtract with Borrow	A=A-[HL]-CY
CPO a	EA	-----	9	Call on Parity Odd	If P=0 (18-s)	SBI n	DE	*****	7	Subtract with Borrow Immed	A=A-n-CY
CZ a	CC	-----	9	Call on Zero	If Z=1 (18-s)	SHLD a	22	-----	16	Store HL Direct	[a]=HL
DAA	27	*****	4	Decimal Adjust Accumulator	A=BCD format	SIM	30	-----	4	Set Interrupt Mask	mask=A
DAD B	09	-----	10	Double Add BC to HL	HL=HL+BC	SPHL	F9	-----	6	Move HL to SP	SP=HL
DAD D	19	-----	10	Double Add DE to HL	HL=HL+DE	STA a	32	-----	13	Store Accumulator	[a]=A
DAD H	29	-----	10	Double Add HL to HL	HL=HL+HL	STAX B	02	-----	7	Store Accumulator indirect	[BC]=A
DAD SP	39	-----	10	Double Add SP to HL	HL=HL+SP	STAX D	12	-----	7	Store Accumulator indirect	[DE]=A
DCR r	3D	*****	4	Decrement	r=r-1 (0X5)	STC	37	-----	4	Set Carry	CY=1
DCR M	35	*****	10	Decrement Memory	[HL]=[HL]-1	SUB r	97	*****	4	Subtract	A=A-r (22X)
DCX B	08	-----	6	Decrement BC	BC=BC-1	SUB M	96	*****	7	Subtract Memory	A=A-[HL]
DCX D	18	-----	6	Decrement DE	DE=DE-1	SUI n	D6	*****	7	Subtract Immediate	A=A-n
DCX H	28	-----	6	Decrement HL	HL=HL-1	XCHG	E3	-----	4	Exchange HL with DE	HL<->DE
DCX SP	38	-----	6	Decrement Stack Pointer	SP=SP-1	XRA r	AF	*****	4	Exclusive OR Accumulator	A=A&r (25X)
DI	F3	-----	4	Disable Interrupts		XRA M	AE	*****	7	Exclusive OR Accumulator	A=A&[HL]
EI	FB	-----	4	Enable Interrupts		XRI n	EE	*****	7	Exclusive OR Immediate	A=A&n
HLT	76	-----	5	Halt		XTHL	E3	-----	16	Exchange stack Top with HL	[SP]<->HL
IN p	DB	-----	10	Input	A=[p]						
INR r	3C	*****	4	Increment	r=r+1 (0X4)	PSW	-*01			Flag unaffected/affected/reset/set	
INR M	3C	*****	10	Increment Memory	[HL]=[HL]+1	S	S			Sign (Bit 7)	
INX B	03	-----	6	Increment BC	BC=BC+1	Z	Z			Zero (Bit 6)	
INX D	13	-----	6	Increment DE	DE=DE+1	A	A			Auxiliary Carry (Bit 4)	
INX H	23	-----	6	Increment HL	HL=HL+1	P	P			Parity (Bit 2)	
INX SP	33	-----	6	Increment Stack Pointer	SP=SP+1	CY	C			Carry (Bit 0)	
JMP a	C3	-----	7	Jump unconditional	PC=a						
JC a	DA	-----	7	Jump on Carry	If CY=1 (10-s)	a p				Direct addressing	
JM a	FA	-----	7	Jump on Minus	If S=1 (10-s)	M z				Register indirect addressing	
JNC a	D2	-----	7	Jump on No Carry	If CY=0 (10-s)	n nn				Immediate addressing	
JNE a	C2	-----	7	Jump on No Zero	If Z=0 (10-s)	r				Register addressing	
JP a	F2	-----	7	Jump on Plus	If S=0 (10-s)						
JPE a	EA	-----	7	Jump on Parity Even	If P=1 (10-s)	DB n,(n)				Define Byte(s)	
JPO a	E2	-----	7	Jump on Parity Odd	If P=0 (10-s)	DB 'string'				Define Byte ASCII character string	
JZ a	CA	-----	7	Jump on Zero	If Z=1 (10-s)	DS nn				Define Storage Block	
LDA a	3A	-----	13	Load Accumulator direct	A=[a]	DW nn,(nn)				Define Word(s)	
LDAX B	0A	-----	7	Load Accumulator indirect	A=[BC]						
LDAX D	1A	-----	7	Load Accumulator indirect	A=[DE]	A B C D E H L				Registers (8-bit)	
LHLD a	2A	-----	16	Load HL Direct	HL=[a]	BC DE HL				Register pairs (16-bit)	
LXI B,nn	01	-----	10	Load Immediate BC	BC=nn	PC				Program Counter register (16-bit)	
LXI D,nn	11	-----	10	Load Immediate DE	DE=nn	PSW				Processor Status Word (8-bit)	
LXI H,nn	21	-----	10	Load Immediate HL	HL=nn	SP				Stack Pointer register (16-bit)	
LXI SP,nn	31	-----	10	Load Immediate Stack Ptr	SP=nn						
MOV r1,r2	7F	-----	4	Move register to register	r1=r2 (1XX)	a nn				16-bit address/data (0 to 65535)	
MOV M,r	77	-----	7	Move register to Memory	[HL]=r (16X)	n p				8-bit data/port (0 to 255)	
MOV r,M	7E	-----	7	Move Memory to register	r=[HL] (1X6)	r				Register (X=B,C,D,E,H,L,M,A)	
MVI r,n	3E	-----	7	Move Immediate	r=n (0X6)	x				Vector (X=0H,8H,10H,18H,20H,28H,30H,38H)	
MVI M,n	36	-----	10	Move Immediate to Memory	[HL]=n	+ -				Arithmetic addition/subtraction	
NOP	00	-----	4	No Operation		& -				Logical AND/NOT	
ORA r	B7	*****	4	Inclusive OR Accumulator	A=A r (26X)	v x				Logical inclusive/exclusive OR	
ORA M	B6	*****	7	Inclusive OR Accumulator	A=A [HL]	<- ->				Rotate left/right	
ORI n	F6	*****	7	Inclusive OR Immediate	A=A n	<->				Exchange	
OUT p	D3	-----	10	Output	[p]=A	[]				Indirect addressing	
PCHL	E9	-----	6	Jump HL indirect	PC=[HL]	[] + - []				Indirect address auto-inc/decrement	
POP B	CD	-----	10	Pop BC	BC=[SP]+	{ }				Combination operands	
POP D	DD	-----	10	Pop DE	DE=[SP]+	{ X }				Octal op code where X is a 3-bit code	
POP H	ED	-----	10	Pop HL	HL=[SP]+	If (~s)				Number of cycles if condition true	
POP PSW	F1	-----	10	Pop Processor Status Word	{PSW,A}=[SP]+						

بسمه تعالی



سوالات امتحان آخر ترم درس ریزپردازنده کد: ۱۹۲۰ گروه کامپیوتر استاد فاتحی

دانشگاه آزاد اسلامی واحد تهران جنوب - ترم تابستان سال تحصیلی ۸۷-۸۶

مدت امتحان: ۲ ساعت تاریخ امتحان: ۱۳۸۶/۱۱/۱۵ جزوه بسته و ماشین حساب لازم نیست

```

PortA      EQU 8000h
PortB      EQU 8001h
PortCLower EQU 8002h
PortCUpper EQU 8002h
PortCMD    EQU 8003h
RAM        EQU E000h
Stack      EQU FFFFh

        JMP Start
        ORG 36          ;24h
        JMP GrTime
        ORG 44          ;2Ch
        JMP YelTime
        ORG 52          ;34h
        JMP GrYelKey

Start:   LXI SP, Stack ;Initilize
        MVI A, 91h
        STA PortCMD
        MVI E, 1
        MVI D, 1
        EI

MainLoop: MVI A, 92h    ;Entry#1
        STA PortB
        MVI A, 10h
        STA PortCUpper
        MOV B, E       ;GrTime
        CALL dlynS

        MVI A, 92h
        STA PortB
        MVI A, 20h
        STA PortCUpper
        MOV B, D       ;YelTime
        CALL dlynS

        MVI A, 90h    ;Entry#2
        STA PortB
        MVI A, C0h
        STA PortCUpper
        MOV B, E       ;GrTime
        CALL dly1S

        MVI A, 91h
        STA PortB
        MVI A, 40h
        STA PortCUpper
        MOV B, D       ;YelTime
        CALL dlynS

        MVI A, 86h    ;Entry#3
        STA PortB
        MVI A, 40h
        STA PortCUpper
        MOV B, E       ;GrTime
        CALL dlynS
        MVI A, 8Ah
        STA PortB
        MVI A, 40h
        STA PortCUpper

```

صفحه ۴

بسمه تعالی

سوالات امتحان آخر ترم درس ریزپردازنده کد: ۱۹۲۰ گروه کامپیوتر استاد فاتحی

دانشگاه آزاد اسلامی واحد تهران جنوب - ترم تابستان سال تحصیلی ۸۷-۸۶

مدت امتحان: ۲ ساعت تاریخ امتحان: ۱۳۸۶/۱۱/۱۵ جزوه بسته و ماشین حساب لازم نیست



```

MOV B,D ;YelTime
CALL dlynS

MVI A,32h ;Entry#4
STA PortB
MVI A,40h
STA PortCUpper
MOV B,E ;GrTime
CALL dlynS

MVI A,52h
STA PortB
MVI A,40h
STA PortCUpper
MOV B,D ;YelTime
CALL dlynS
JMP MainLoop

GrTime:
LDA IncDcrFlg
ADI 0
JZ GrInc
DCR E
RET

GrInc:
INC E
RET

YelTime:
LDA IncDcrFlg
ADI 0
JZ YelInc
DCR D
RET

YelInc:
INC D
RET

GrYelKey:
LDA IncDcrFlg *
XRA FFh *
STA IncDcrFlg *
RET *

dlynS:
... ;F=10MHz

RET

IncDcrFlg:
ORG RAM
DB 0

END

```

بسمه تعالی

سوال‌های امتحان آخر ترم درس ریزپردازنده گروه کامپیوتر

دانشگاه آزاد اسلامی واحد جنوب تهران - ترم اول سال تحصیلی ۸۴-۸۳

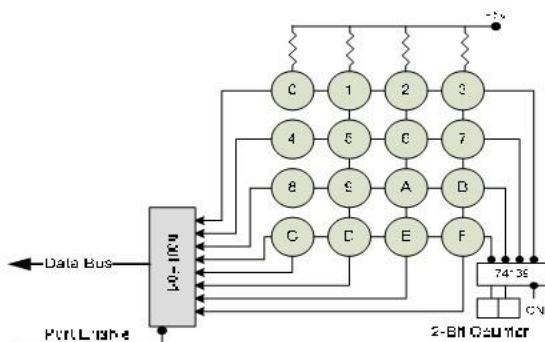
کد درس: ۱۹۲۰ مدت امتحان: دو ساعت تاریخ امتحان: ۱۳۸۳/۱۰/۳۰ جزوه بسته



تذکر یک: اگر در برگه سوال جواب دادید، در پاسخنامه به آن اشاره کنید.

تذکر دو: می‌توانید با مداد پاسخ بدهید.

- ۱- قابلیت‌های ریزپردازنده ۸۰۸۵ و Z80 را باهم مقایسه کنید. (۱۵ نمره)
- ۲- یک میکروکنترلر در مقایسه با ریزپردازنده و سیستم حداقل چه توانایی دارد. بلوک دیاگرام یک میکروکنترلر نمونه را توضیح دهید. (۱۰ نمره)
- ۳- مدار ایجاد دو کلاک تاخیر را طرح کرده و با رسم TIMING مربوط به OF عملکرد درست آنرا نشان دهید. (۱۰ نمره)
- ۴- با توجه به مدار ضمیمه به سوالات زیر با ذکر دلیل کافی پاسخ دهید. (هر کدام ۵ نمره = ۲۵ نمره)
 (آ) محدوده و اندازه آدرسهای هر یک از خروجی‌های ۷۴۱۳۸ را به طور مشخص بنویسید.
 (ب) آدرس پورت‌های ۸۲۵۵ را به صورت دقیق مشخص کنید.
 (ت) کلمه برنامه ریزی ۸۲۵۵ را بدست آورده و دستورات برنامه ریزی متناسب را بنویسید.
 (پ) اگر اندازه RAM و ROM هر کدام یک کیلو بایت باشد، آدرس‌دهی آنها چگونه خواهد بود.
 (ث) یک قطعه RAM یک کیلو بیتی دیگر با کمترین تغییر سخت افزاری به سیستم اضافه کنید.
 ۵- با توجه به نرم افزار پیوست (پاسخ میان ترم) به سوالات زیر پاسخ دهید. (۳۰ نمره)
 (آ) عملکرد هر کدام از زیر برنامه‌های ذکر شده را توضیح دهید. (از توضیح دستورات خوداری فرماید) DirReq, NxtStep, CurReq
 (ب) عملکرد برنامه اصلی (Mainloop) را با رسم فلوچارت بدون توضیح جزئیات زیربرنامه‌ها بنویسید.
 (پ) روتین تاخیر (dlyNs) کامل نمی‌باشد آنرا با توجه به توضیحات جانبی برنامه کامل کنید.



- ۶- با توجه به مدار صفحه کلید در شکل رویرو نحوه عملکرد آنرا با رسم فلوچارت توضیح دهید. اگر بخواهیم خبر فشرده شدن کلید به صورت وقفه به CPU داده شود چه پیشنهادی دارید. (۱۰ نمره)

موفق باشید- فاتحی

صفحه ۱

بسمه تعالی

سوالات امتحان آخر ترم درس ریزیردازنده گروه کامپیوتر

دانشگاه آزاد اسلامی واحد جنوب تهران - ترم اول سال تحصیلی ۸۴-۸۳

کد درس : ۱۹۲۰ مدت امتحان : دو ساعت تاریخ امتحان : ۱۳۸۳/۱۰/۳۰ جزوه بسته



8085 Instruction Set

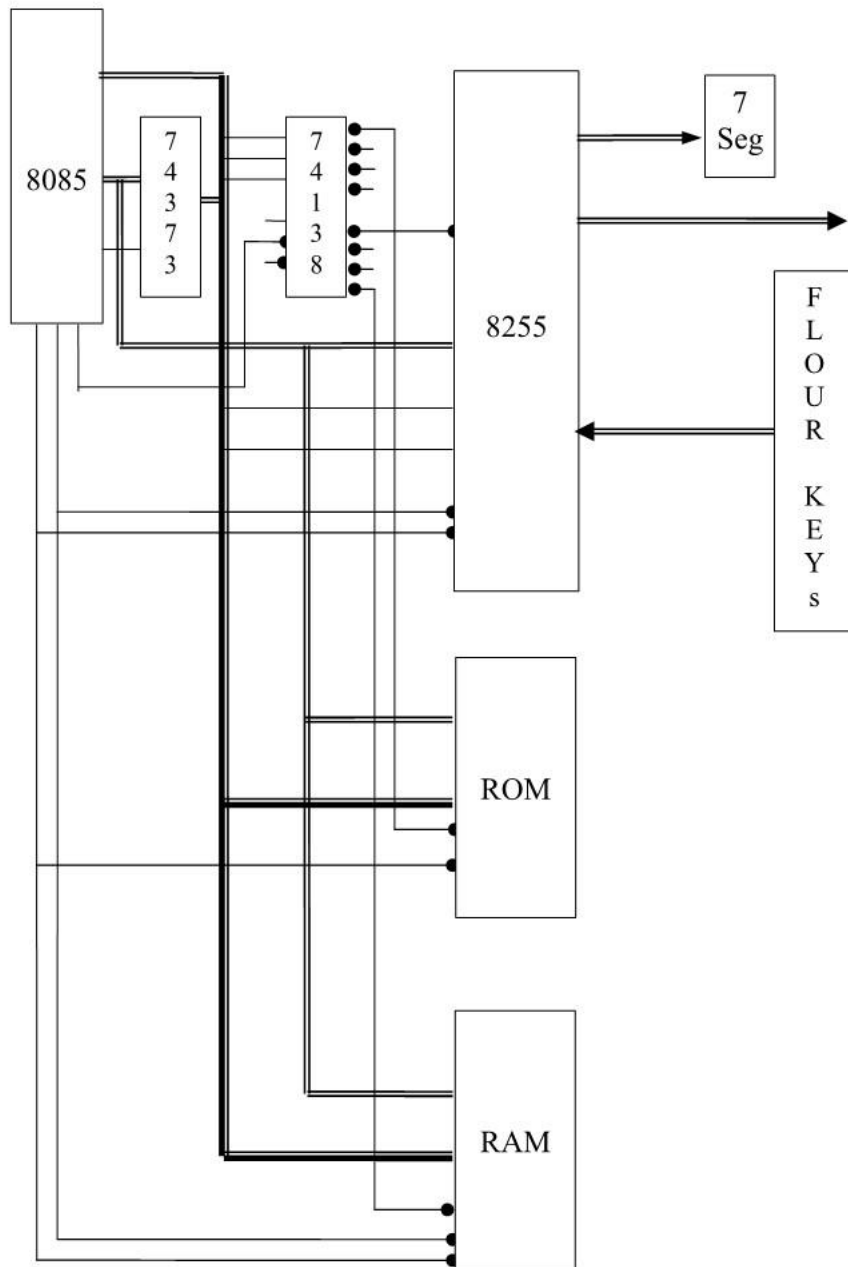
Mnemonic	Op	SZAPC	-s	Description	Notes	Mnemonic	Op	SZAPC	-s	Description	Notes
ACI n	CE	*****	7	Add with Carry Immediate	A←A+n+CY	(PUSH B	CS	----	12	Push BC	← SPJ -BC
ADC r	8F	*****	4	Add with Carry	A←A+r+CY(21X)	(PUSH D	DS	----	12	Push DE	← SPJ -DE
ADC M	8E	*****	7	Add with Carry to Memory	A←A+[HL]+CY	(PUSH H	ES	----	12	Push HL	← SPJ =HL
ADD r	B7	*****	4	Add	A←A+r (20X)	(PUSH PSW	F5	----	12	Push Processor Status Word	← SPJ -[PSW,A]
ADD M	86	*****	7	Add to Memory	A←A+[HL]	RAL	17	----	4	Rotate Accumulator Left	A←(CY,A)←
ADI n	CE	*****	7	Add Immediate	A←A+n	RAR	1F	----	4	Rotate Accumulator Right	A←>(CY,A)
ANA r	A7	*****	4	AND Accumulator	A←A&r (24X)	RET	C9	----	10	Return	PC←SPJ+
ANA M	A6	*****	7	AND Accumulator and Memory	A←A&[HL]	RC	D8	----	6	Return on Carry	If CY=1(12-s)
ANI n	E6	*****	4	AND Immediate	A←A&n	RIM	20	----	4	Read Interrupt Mask	A←mask
CALL a	CD	----	18	Call unconditional	←[SP]-PC,PC←a	RM	F8	----	6	Return on Minus	If S=1(12-s)
CC a	DC	----	9	Call on Carry	If CY=1(18-s)	RNC	D0	----	6	Return on No Carry	If CY=0(12-s)
CM a	FC	----	9	Call on Minus	If S=1(18-s)	RNZ	CO	----	6	Return on No Zero	If Z=0(12-s)
CMA	2F	----	4	Complement Accumulator	A←~A	RP	F0	----	6	Return on Plus	If S=0(12-s)
CMC	3F	----	4	Complement Carry	CF←~CF	RPE	E8	----	6	Return on Parity Even	If P=1(12-s)
CMF r	BF	*****	4	Compare	A←r (27X)	RPO	E0	----	6	Return on Parity Odd	If P=0(12-s)
CMP M	BF	*****	7	Compare with Memory	A←[HL]	RZ	C8	----	6	Return on Zero	If Z=1(12-s)
CMC a	D4	----	9	Call on No Carry	If CY=0(18-s)	RLC	07	----	4	Rotate Left Circular	A←A<
CNZ a	C4	----	9	Call on No Zero	If Z=0(18-s)	RRC	0F	----	4	Rotate Right Circular	A←A>
CP a	FA	----	9	Call on Plus	If S=0(18-s)	RST z	C7	----	12	Restart	(3X7)←SPJ-PC,PC←z
CPE a	EC	----	9	Call on Parity Even	If P=1(18-s)	SBB r	9F	*****	4	Subtract with Borrow	A←A-r-CY
CFI n	FE	*****	7	Compare Immediate	A←n	SBB M	9E	*****	7	Subtract with Borrow	A←A-[HL]-CY
CPD a	EA	----	9	Call on Parity Odd	If P=0(18-s)	SBI n	DE	*****	7	Subtract with Borrow Immediate	A←A-n-CY
CZ a	CC	----	9	Call on Zero	If Z=1(18-s)	SHLD a	22	----	16	Store HL Direct	[a]←HL
DAA	27	*****	4	Decimal Adjust Accumulator	A←BCD format	SIM	30	----	4	Set Interrupt Mask	mask←A
DAD B	09	----	10	Double Add BC to HL	HL←HL+BC	SPHL	F9	----	6	Move HL to SP	SP←HL
DAD D	19	----	10	Double Add DE to HL	HL←HL+DE	STA a	32	----	13	Store Accumulator	[a]←A
DAD H	29	----	10	Double Add HL to HL	HL←HL+HL	STAX B	02	----	7	Store Accumulator indirect	[BC]←A
DAD SP	39	----	10	Double Add SP to HL	HL←HL+SP	STAX D	12	----	7	Store Accumulator indirect	[DE]←A
DCR r	3D	*****	4	Decrement	r←r-1 (0X5)	STC	37	----	4	Set Carry	CY=1
DCR M	35	*****	10	Decrement Memory	[HL]←[HL]-1	SUB r	97	*****	4	Subtract	A←A-r (22X)
DCX B	0B	----	6	Decrement BC	BC←BC-1	SUB M	96	*****	7	Subtract Memory	A←A-[HL]
DCX D	1B	----	6	Decrement DE	DE←DE-1	SUB n	D6	*****	7	Subtract Immediate	A←A-n
DCX H	2B	----	6	Decrement HL	HL←HL-1	XCHG	EB	----	4	Exchange HL with DE	HL↔DE
DCX SP	3B	----	6	Decrement Stack Pointer	SP←SP-1	XRA r	AF	**0*0	4	Exclusive OR Accumulator	A←A&r (25X)
DI	F3	----	4	Disable Interrupts		XRA M	AE	**0*0	7	Exclusive OR Accumulator	A←A&[HL]
EI	FB	----	4	Enable Interrupts		XRI n	EE	**0*0	7	Exclusive OR Immediate	A←A&n
HLT	76	----	5	Halt		XTHL	1F3	----	16	Exchange stack Top with HL	[SP]↔HL
IN p	DB	----	10	Input	A←[p]						
INR r	3C	*****	4	Increment	r←r+1 (0X4)	PSW	-*01			Flag unaffected/affected/reset/set	
INR M	3C	*****	10	Increment Memory	[HL]←[HL]+1	S	S			Sign (Bit 7)	
INX B	03	----	6	Increment BC	BC←BC+1	Z	Z			Zero (Bit 6)	
INX D	13	----	6	Increment DE	DE←DE+1	AC	A			Auxiliary Carry (Bit 4)	
INX H	23	----	6	Increment HL	HL←HL+1	P	P			Parity (Bit 2)	
INX SP	33	----	6	Increment Stack Pointer	SP←SP+1	CY	C			Carry (Bit 0)	
JMP a	C3	----	7	Jump unconditional	PC←a						
JC a	DA	----	7	Jump on Carry	If CY=1(10-s)	a p				Direct addressing	
JM a	FA	----	7	Jump on Minus	If S=1(10-s)	m z				Register indirect addressing	
JNC a	D2	----	7	Jump on No Carry	If CY=0(10-s)	n nn				Immediate addressing	
JNE a	C2	----	7	Jump on No Zero	If Z=0(10-s)	r				Register addressing	
JP a	FA	----	7	Jump on Plus	If S=0(10-s)						
JPE a	EA	----	7	Jump on Parity Even	If P=1(10-s)						
JPO a	E2	----	7	Jump on Parity Odd	If P=0(10-s)						
JZ a	CA	----	7	Jump on Zero	If Z=1(10-s)						
LDA a	3A	----	13	Load Accumulator direct	A←[a]						
LDAX B	0A	----	7	Load Accumulator indirect	A←[BC]	A B C D E H L				Registers (8-bit)	
LDAX D	1A	----	7	Load Accumulator indirect	A←[DE]	BC DE HL				Register pairs (16-bit)	
LHLD a	2A	----	16	Load HL Direct	HL←[a]	PC				Program Counter register (16-bit)	
LXI B,nn	01	----	10	Load Immediate BC	BC←nn	PSW				Processor Status Word (8-bit)	
LXI D,nn	11	----	10	Load Immediate DE	DE←nn	SP				Stack Pointer register (16-bit)	
LXI H,nn	21	----	10	Load Immediate HL	HL←nn						
LXI SP,nn	31	----	10	Load Immediate Stack Ptr	SP←nn						
MOV r1,r2	7F	----	4	Move register to register	r1←r2 (1XX)	a nn				16-bit address/data (0 to 65535)	
MOV M,r	77	----	7	Move register to Memory	[HL]←r (16X)	n p				8-bit data/port (0 to 255)	
MOV r,M	7E	----	7	Move Memory to register	r←[HL] (1X6)	r				Register (X=B,C,D,E,H,L,M,A)	
MVI r,n	3E	----	7	Move Immediate	r←n (0X6)	z				Vector (X=0H, 8H, 10H, 18H, 20H, 28H, 30H, 38H)	
MVI M,n	36	----	10	Move Immediate to Memory	[HL]←n						
NOP	00	----	4	No Operation							
ORA r	B7	**0*0	4	Inclusive OR Accumulator	A←A r (26X)	+ -				Arithmetic addition/subtraction	
ORA M	B6	**0*0	7	Inclusive OR Accumulator	A←A [HL]	& ~				Logical AND/NOT	
ORI n	F6	**0*0	7	Inclusive OR Immediate	A←A n	v x				Logical inclusive/exclusive OR	
OUT p	D3	----	10	Output	[p]←A	<- >				Rotate left/right	
FCHL	E9	----	6	Jump HL indirect	PC←[HL]	<->				Exchange	
POP B	C1	----	10	Pop BC	BC←[SP]+	[]				Indirect addressing	
POP D	D1	----	10	Pop DE	DE←[SP]+	[] + []				Indirect address auto-inc/decrement	
POP H	E1	----	10	Pop HL	HL←[SP]+	{ }				Combination operands	
POP PSW	F1	----	10	Pop Processor Status Word	[PSW,A]←[SP]+	{ X }				Octal op code where X is a 3-bit code	
						If (-s)				Number of cycles if condition true	

بسمه تعالی

سوالات امتحان آخر ترم درس ریزپردازنده گروه کامپیوتر

دانشگاه آزاد اسلامی واحد جنوب تهران - ترم اول سال تحصیلی ۸۳-۸۴

کد درس : ۱۹۲۰ مدت امتحان : دو ساعت تاریخ امتحان : ۱۳۸۳/۱۰/۳۰ جزوه بسته



صفحه ۳

بسمه تعالی

سوال‌های امتحان آخر ترم درس ریزپردازنده گروه کامپیوتر

دانشگاه آزاد اسلامی واحد جنوب تهران - ترم اول سال تحصیلی ۸۴-۸۳

کد درس: ۱۹۲۰ مدت امتحان: دو ساعت تاریخ امتحان: ۱۳۸۳/۱۰/۳۰ جزوه بسته



```

SegPort EQU ???? ;Port A
CtrlPort EQU ???? ;PortB
KeyPort EQU ???? ;PortC
RAM EQU ????
Stack EQU ????
UpMove EQU ??
DnMove EQU ??
Stop EQU ??

ORG 0000
JMP Start

Start: ;Initial
MVI A,?? ;Command
STA CMD
LXI SP,Stack
MVI E,01h ;Cur. flour
MVI D,FFh ;Cur. Dir.
MVI C,0 ;Cur. Req.

Mainloop:
Call TestKey
Call DirReq
jnc ChgDir
Call NxtStep
Call Display
MVI B,20 ;2second
Call dlyNs
Call CurReq
Jz Mainloop
MVI A,Stop
STA CtrlPort
MVI B,100 ;10second
Call dlyNs
MOV A,C
XRA E
MOV C,A
jmp Mainloop
ChgDir: MOV A,D
XRA FFh
MOV D,A
jmp Mainloop

TESTKEY: LDA KeyPort
ORA C
MOV C,A
Ret

dlyNs: ;f=10MHz
;B*100ms
;cnt=?
Loop1:
Call TestKey
DCR B
jnz Loop1

ret

```

بسمه تعالی

سوالات امتحان آخر ترم درس ریزپردازنده گروه کامپیوتر

دانشگاه آزاد اسلامی واحد جنوب تهران - ترم اول سال تحصیلی ۸۴-۸۳

کد درس : ۱۹۲۰ مدت امتحان : دو ساعت تاریخ امتحان : ۱۳۸۳/۱۰/۳۰ جزوه بسته



```

Display:      LXI      H,SegTab
              MOV      A,E
              ANI      0Fh
              MOV      L,A
              MOV      A,M
              STA      SegPort
              ret

NxtStep:      XRA      A          ;Clear A
              ADD      D
              MOV      A,E
              Jz       DnStep
UpStep:      RLC
              MOV      E,A
              MVI      A,UpMove
              Jmp      NxtEnd
DnStep:      RRC
              MOV      E,A
              MVI      A,DnMove
NxtEnd:      STA      CtrlPort
              ret

CurReq:      MOV      A,E
              ANA      C
              ret

DirReq:      XRA      A          ;Clear A
              ADD      D
              Jz       Down
UP:          MVI      D,0
Loop3:      MOV      A,E
              CPI      08h
              Jz       Exit1
              RLC
              MOV      B,A
              ANA      C
              MOV      A,B
              Jz       Loop3
              MVI      D,FFh
              Jmp      Exit0
Down:      MVI      D,FFh
              MOV      A,E
Loop4:      CPI      01h
              jz       Exit1
              RRC
              MOV      B,A
              ANA      C
              MOV      A,B
              Jz       Loop4
Exit0:      STC
              ret
Exit1:      STC
              CMC
              ret

SegTab:      ORG      0100h
              DB      ??          ;1
              DB      ??          ;2
              DB      ??          ;3
              DB      ??          ;4

```

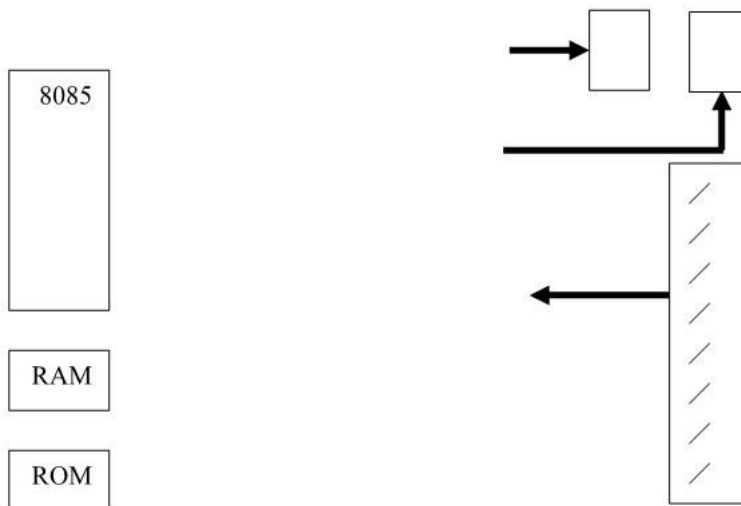
بسمه تعالی

سوالات میان‌ترم درس ریزپردازنده - دانشگاه آزاد اسلامی واحد جنوب تهران - جزوه باز - ترم دوم سال ۸۴-۸۳

سیستمی طرح کنید که دارای:

- ۱- هشت عدد کلید فشاری باشد.
- ۲- یک نمایشگر دو رقمی با دو عدد نمایشگر ۷ قطعه‌ای (7-Seg) باشد. (یکان و دهگان)
- ۳- یک حافظه RAM با گنجایش 2KB و یک حافظه ROM با گنجایش 2KB باشد.
بتواند عملکرد زیر را داشته باشد: (تا حد امکان برنامه هر قسمت مشخص باشد)
- ۴- کلیدها را بررسی کرده و در صورت فشرده شدن کلید، شماره آنرا بدست آورده و به صورت یک رقم بین ۰ تا ۷ در رقم یکان نمایشگر بنویسد. (فرض می شود در هر زمان فقط یک کلید فشرده می شود و فشرده شدن طولانی مدت یک کلید یک بار محسوب شود).
- ۵- شماره کلید قبلی از موقعیت یکان به سمت چپ جابجا شود. (به جای دهگان) بطوریکه همیشه آخرین دو رقم (آخرین دو کلید فشرده شده) نمایش داده می شود.
- ۶- توجه داشته باشید که در هنگام نمایش، شماره کلید را به کد 7-Seg تبدیل کند.
- ۷- توجه داشته باشید که سیستم، باید فشرده شدن و رها شدن کلید را تشخیص بدهد تا بتواند کلیدهای فشرده شده را از هم تفکیک کند.

سخت افزار مورد نیاز این سیستم را با جزئیات طرح و رسم کرده و نرم افزار لازم آنرا در قالب یک برنامه بنویسید. (استفاده از فلوجارت برای بیان الگوریتم خود در ارزیابی پاسخ شما مفید خواهد بود)



موفق باشید - فاتحی

بسمه تعالی

سوال‌ات امتحان آخر ترم درس ریزپردازنده گروه کامپیوتر

دانشگاه آزاد اسلامی واحد جنوب تهران - ترم اول سال تحصیلی ۸۶-۸۵

کد درس: ۱۹۲۰ مدت امتحان: ۲ ساعت تاریخ امتحان: ۱۳۸۵/۱۰/۲۷

امتحان جزوه بسته است و ماشین حساب لازم نیست



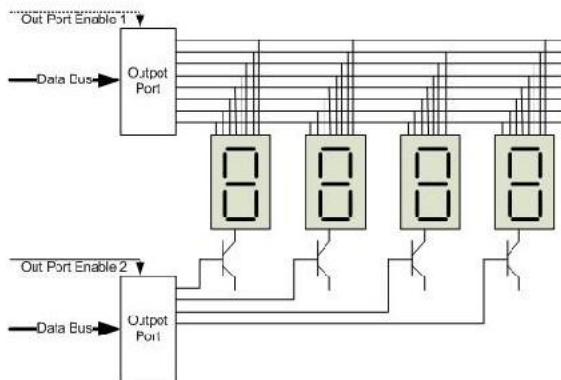
نام و نام خانوادگی:

تذکر ۱: برگه سوال، ضمیمه پاسخنامه خواهد بود. می‌توانید به برخی از سوالات در برگه سوال جواب دهید.

تذکر ۲: برای کاهش قلم خوردگی توصیه می‌شود با مداد پاسخ دهید.

تذکر ۳: نمره نهایی از ۱۰۰ نمره این امتحان و ۱۰ نمره کلاس مشخص خواهد شد.

- ۱- قابلیت‌های وقفه را در ریزپردازنده‌های 8085 و Z80 و 8086 در مقایسه با هم بنویسید. (۱۵ نمره)
- ۲- با رسم بلوک دیاگرام یک میکروکنترلر نمونه قابلیت‌های آنرا توضیح دهید. (۱۰ نمره)
- ۳- دستورالعمل‌های زیر دارای چه ماشین‌های ساینکل‌هایی هستند آنها را به ترتیب اجرا ذکر کرده و بنویسید در هر کدام دقیقاً چه کاری انجام می‌شود. (۱۰ نمره) RET و IN 80h
- ۴- با توجه به مدار ضمیمه به سوالات زیر با ذکر دلیل کافی پاسخ دهید. (هر کدام ۵ نمره = ۲۵ نمره)
 - a. محدوده و اندازه آدرسهای هر یک از خروجی‌های دیکودرها را به طور مشخص بنویسید.
 - b. آدرس پورت‌های ۸۲۵۵ را به صورت دقیق مشخص کنید. (آدرس‌های تکراری مشخص شود)
 - c. کلمه برنامه ریزی ۸۲۵۵ را بدست آورده و دستورات برنامه ریزی متناسب را بنویسید.
 - d. اگر اندازه RAM و ROM هر کدام یک کیلو بایت باشد، آدرس‌دهی آنها چگونه خواهد بود.
 - e. یک قطعه RAM یک کیلو بایتی دیگر با کمترین تغییر سخت افزاری به سیستم اضافه کرده و آدرس آنرا مشخص کنید.
- ۵- با توجه به نرم افزار پیوست (کنترل چهار راه ۴ زمانه) به سوالات زیر پاسخ دهید. (۲۰ نمره)
 - a. به سه پایه وقفه سه کلید وصل شده است عملکرد آنها را نسبت به هم توضیح دهید.
 - b. زمان چراغهای سبز، زرد و قرمز هر ورودی به چه ترتیبی مشخص و کنترل می‌شود.
 - c. روتین تاخیر B ثانیه خالی می‌باشد آنرا بطور کامل نوشته و زمان تاخیر آنرا برای B ثانیه تنظیم کنید. (f=10MHz)
 - d. ترجمه زبان ماشین ۴ خط ستاره‌دار را بطور کامل و دقیق بنویسید.



- ۶- قطعه برنامه‌ای بنویسید که بتواند مقدار یک شمارنده ۱۶ بیتی مثل زوج رجیستر B و C را در منبای ۱۶ در ۴ عدد نمایشگر شکل روبرو بنویسد. آدرس پورت ۱ 80h و آدرس پورت ۲ 81h فرض شود. (۲۰ نمره)

صفحه ۱

بسمه تعالی

سوالات امتحان آخر ترم درس ریزپردازنده گروه کامپیوتر

دانشگاه آزاد اسلامی واحد جنوب تهران - ترم اول سال تحصیلی ۸۶-۸۵

کد درس : ۱۹۲۰ مدت امتحان: ۲ ساعت تاریخ امتحان: ۱۳۸۵/۱۰/۲۷

امتحان جزوه بسته است و ماشین حساب لازم نیست



8085 Instruction Set

Mnemonic	Op	sz	APC	-s	Description	Notes
ADD r	8E	*****	7		Add with Carry Immediate	A=A+r+CY
ADC r	8F	*****	4		Add with Carry	A=A+r+CY (21X)
ADC M	8E	*****	7		Add with Carry to Memory	A=A+[HL]+CY
ADD r	87	*****	4		Add	A=A+r (20X)
ADD M	86	*****	7		Add to Memory	A=A+[HL]
ADI n	C6	*****	7		Add Immediate	A=A+n
ANA r	A7	*****	4		AND Accumulator	A=A&r (24X)
ANA M	A6	*****	7		AND Accumulator and Memory	A=A&[HL]
ANI n	E6	****0*	7		AND Immediate	A=A&n
CALL a	CD	-----	18		Call unconditional	-[SP]=PC,PC=a
CC a	DC	-----	9		Call on Carry	If CY=1 (18-s)
CM a	FC	-----	9		Call on Minus	If S=1 (18-s)
CMA	2F	-----	4		Complement Accumulator	A=~A
CMC	3F	-----	4		Complement Carry	CY=~CY
CMP r	BF	*****	4		Compare	A-r (27X)
CMP M	BF	*****	7		Compare with Memory	A-[HL]
CMC a	D4	-----	9		Call on No Carry	If CY=0 (18-s)
CNZ a	C4	-----	9		Call on No Zero	If Z=0 (18-s)
CP a	FA	-----	9		Call on Plus	If S=0 (18-s)
CPE a	EC	-----	9		Call on Parity Even	If P=1 (18-s)
CPI n	FE	*****	7		Compare Immediate	A-n
CPO a	EA	-----	9		Call on Parity Odd	If P=0 (18-s)
CZ a	CC	-----	9		Call on Zero	If Z=1 (18-s)
DAA	27	*****	4		Decimal Adjust Accumulator	A=BCD format
DAD B	09	-----	10		Double Add BC to HL	HL=HL+BC
DAD D	19	-----	10		Double Add DE to HL	HL=HL+DE
DAD H	29	-----	10		Double Add HL to HL	HL=HL+HL
DAD SP	39	-----	10		Double Add SP to HL	HL=HL+SP
DCR r	3D	*****	4		Decrement	r=r-1 (0X5)
DCR M	35	*****	10		Decrement Memory	[HL]=[HL]-1
DCX B	0B	-----	6		Decrement BC	BC=BC-1
DCX D	1B	-----	6		Decrement DE	DE=DE-1
DCX H	2B	-----	6		Decrement HL	HL=HL-1
DCX SP	3B	-----	6		Decrement Stack Pointer	SP=SP-1
DI	FB	-----	4		Disable Interrupts	
EI	FB	-----	4		Enable Interrupts	
HLT	76	-----	5		Halt	
IN p	DB	-----	10		Input	A=[p]
INR r	3C	*****	4		Increment	r=r+1 (0X4)
INR M	3C	*****	10		Increment Memory	[HL]=[HL]+1
INX B	03	-----	6		Increment BC	BC=BC+1
INX D	13	-----	6		Increment DE	DE=DE+1
INX H	23	-----	6		Increment HL	HL=HL+1
INX SP	33	-----	6		Increment Stack Pointer	SP=SP+1
JMP a	C3	-----	7		Jump unconditional	PC=a
JC a	DA	-----	7		Jump on Carry	If CY=1 (10-s)
JM a	FA	-----	7		Jump on Minus	If S=1 (10-s)
JNC a	DA	-----	7		Jump on No Carry	If CY=0 (10-s)
JNZ a	CA	-----	7		Jump on No Zero	If Z=0 (10-s)
JP a	F2	-----	7		Jump on Plus	If S=0 (10-s)
JPE a	EA	-----	7		Jump on Parity Even	If P=1 (10-s)
JPO a	E2	-----	7		Jump on Parity Odd	If P=0 (10-s)
JZ a	CA	-----	7		Jump on Zero	If Z=1 (10-s)
LDA a	3A	-----	13		Load Accumulator direct	A=[a]
LDAX B	0A	-----	7		Load Accumulator indirect	A=[BC]
LDAX D	1A	-----	7		Load Accumulator indirect	A=[DE]
LHLD a	2A	-----	16		Load HL Direct	HL=[a]
LXI B,nn	01	-----	10		Load Immediate BC	BC=nn
LXI D,nn	11	-----	10		Load Immediate DE	DE=nn
LXI H,nn	21	-----	10		Load Immediate HL	HL=nn
LXI SP,nn	31	-----	10		Load Immediate Stack Ptr	SP=nn
MOV r1,r2	7F	-----	4		Move register to register	r1=r2 (1XX)
MOV M,r	77	-----	7		Move register to Memory	[HL]=r (16X)
MOV r,M	7E	-----	7		Move Memory to register	r=[HL] (16X)
MVI r,n	3E	-----	7		Move Immediate	r=n (0X6)
MVI M,n	36	-----	10		Move Immediate to Memory	[HL]=n
INOP	00	-----	4		No Operation	
ORA r	B7	****0*	4		Inclusive OR Accumulator	A=A r (26X)
ORA M	B6	****0*	7		Inclusive OR Accumulator	A=A [HL]
ORI n	F6	****0*	7		Inclusive OR Immediate	A=A n
OUT p	D3	-----	10		Output	[p]=A
PCHL	E9	-----	6		Jump HL indirect	PC=[HL]
POP B	CD	-----	10		Pop BC	BC=(SP)+
POP D	DD	-----	10		Pop DE	DE=(SP)+
POP H	ED	-----	10		Pop HL	HL=(SP)+
POP PSW	FD	-----	10		Pop Processor Status Word	[PSW,A]=[SP]+
PUSH B	C5	-----	12		Push BC	-[SP]=BC
PUSH D	D5	-----	12		Push DE	-[SP]=DE
PUSH H	E5	-----	12		Push HL	-[SP]=HL
PUSH PSW	F5	-----	12		Push Processor Status Word	-[SP]=[PSW,A]
RAL	17	-----*	4		Rotate Accumulator Left	A=[CY,A]<-
RAR	1F	-----*	4		Rotate Accumulator Right	A=>[CY,A]
RET	C9	-----	10		Return	PC=[SP]+
RC	D8	-----	6		Return on Carry	If CY=1 (12-s)
RIM	20	-----	4		Read Interrupt Mask	A=mask
RM	F8	-----	6		Return on Minus	If S=1 (12-s)
RNC	D0	-----	6		Return on No Carry	If Z=0 (12-s)
RNZ	C0	-----	6		Return on No Zero	If Z=0 (12-s)
RP	F0	-----	6		Return on Plus	If S=0 (12-s)
RPE	E8	-----	6		Return on Parity Even	If P=1 (12-s)
RPO	E0	-----	6		Return on Parity Odd	If P=0 (12-s)
RZ	C8	-----	6		Return on Zero	If Z=1 (12-s)
RLC	07	-----*	4		Rotate Left Circular	A=A<-
RRC	0F	-----*	4		Rotate Right Circular	A=A>
RST z	C7	-----	12		Restart	-[SP]=PC,PC=z (3X7)
SBB r	9F	*****	4		Subtract with Borrow	A=A-r-CY
SBB M	9E	*****	7		Subtract with Borrow	A=A-[HL]-CY
SBI n	DE	*****	7		Subtract with Borrow Immed	A=A-n-CY
SHLD a	22	-----	16		Store HL Direct	[a]=HL
SIM	30	-----	4		Set Interrupt Mask	mask=A
SPHL	F9	-----	6		Move HL to SP	SP=HL
STA a	32	-----	13		Store Accumulator	[a]=A
STAX B	02	-----	7		Store Accumulator indirect	[BC]=A
STAX D	12	-----	7		Store Accumulator indirect	[DE]=A
STC	37	-----	4		Set Carry	CY=1
SUB r	97	*****	4		Subtract	A=A-r (22X)
SUB M	96	*****	7		Subtract Memory	A=A-[HL]
SUI n	D6	*****	7		Subtract Immediate	A=A-n
XCHG	EB	-----	4		Exchange HL with DE	HL<->DE
XRA r	AF	****0*	4		Exclusive OR Accumulator	A=A^r (25X)
XRA M	AE	****0*	7		Exclusive OR Accumulator	A=A^[HL]
XRI n	EF	****0*	7		Exclusive OR Immediate	A=A^n
XTHL	E3	-----	16		Exchange stack Top with HL	[SP]<->HL
PSW	-*01				Flag unaffacted/affected/reset/set	
S	S				Sign (Bit 7)	
Z	Z				Zero (Bit 6)	
A	A				Auxiliary Carry (Bit 4)	
P	P				Parity (Bit 2)	
C	C				Carry (Bit 0)	
a p					Direct addressing	
M z					Register indirect addressing	
n nn					Immediate addressing	
r					Register addressing	
DB n,(n)					Define Byte(s)	
DB 'string'					Define Byte ASCII character string	
DS nn					Define Storage Block	
DW nn,(nn)					Define Word(s)	
A B C D E H L					Registers (8-bit)	
BC DE HL					Register pairs (16-bit)	
PC					Program Counter register (16-bit)	
PSW					Processor Status Word (8-bit)	
SP					Stack Pointer register (16-bit)	
a nn					16-bit address/data (0 to 65535)	
n p					8-bit data/port (0 to 255)	
r					Register (X=B,C,D,E,H,L,M,A)	
z					Vector (X=0H,8H,10H,18H,20H,28H,30H,38H)	
+ -					Arithmetic addition/subtraction	
& ~					Logical AND/NOT	
v x					Logical inclusive/exclusive OR	
< ->					Rotate left/right	
<->					Exchange	
[]					Indirect addressing	
[]+ -[]					Indirect address auto-inc/decrement	
{ }					Combination operands	
{ X }					Octal op code where X is a 3-bit code	
If (~s)					Number of cycles if condition true	

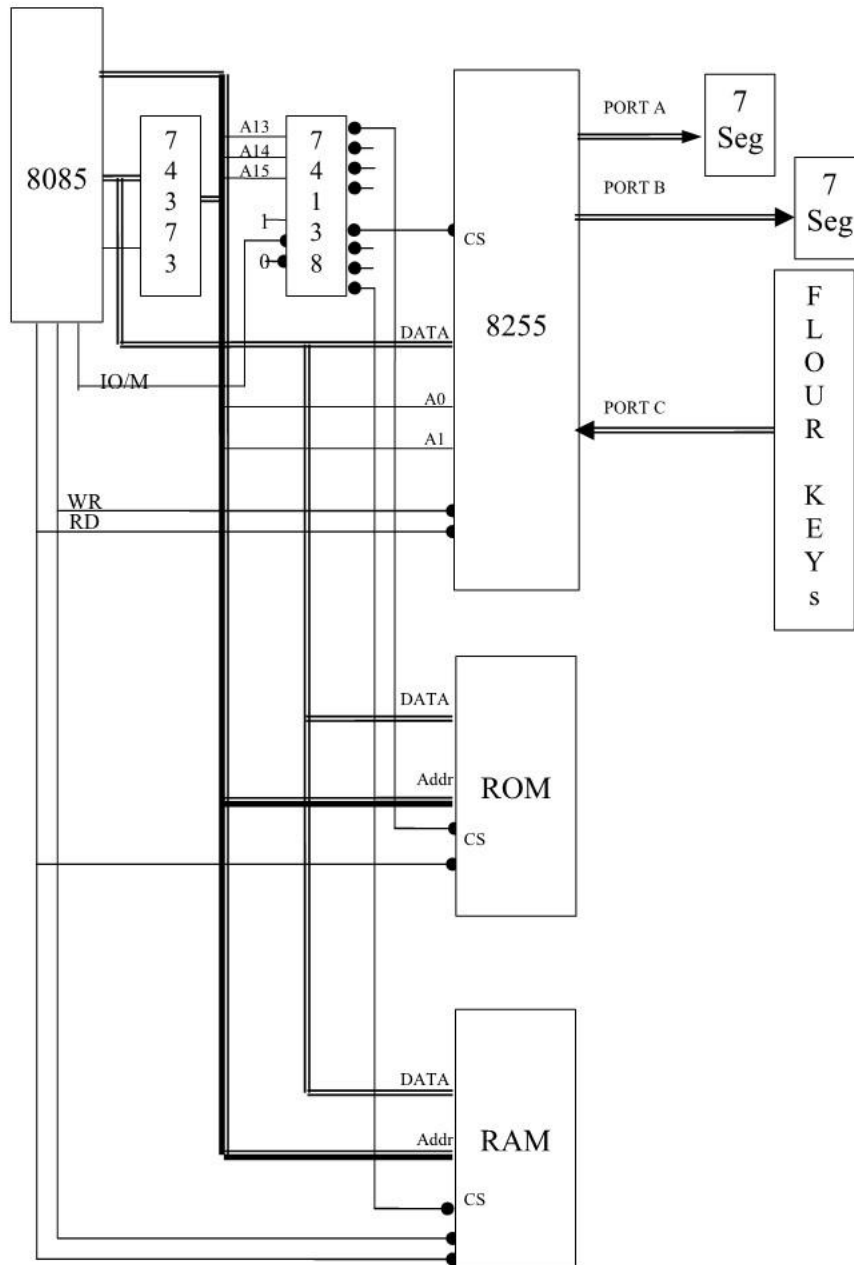
بسمه تعالی

سوالات امتحان آخر ترم درس ریزپردازنده گروه کامپیوتر

دانشگاه آزاد اسلامی واحد جنوب تهران - ترم اول سال تحصیلی ۸۶-۸۵

کد درس : ۱۹۲۰ مدت امتحان: ۲ ساعت تاریخ امتحان: ۱۳۸۵/۱۰/۲۷

امتحان جزوه بسته است و ماشین حساب لازم نیست



صفحه ۳

بسمه تعالی

سوال‌های امتحان آخر ترم درس ریزپردازنده گروه کامپیوتر

دانشگاه آزاد اسلامی واحد جنوب تهران - ترم اول سال تحصیلی ۸۶-۸۵

کد درس: ۱۹۲۰ مدت امتحان: ۲ ساعت تاریخ امتحان: ۱۳۸۵/۱۰/۲۷

امتحان جزوه بسته است و ماشین حساب لازم نیست



```

PortA EQU 8000h
PortB EQU 8001h
PortCLower EQU 8002h
PortCUpper EQU 8002h
PortCMD EQU 8003h
RAM EQU E000h
Stack EQU FFFFh

JMP Start
ORG 36 ;24h
JMP GrYelTime1
ORG 44 ;2Ch
JMP GrYelTime2
ORG 52 ;34h
JMP GrYelKey

Start: LXI SP, Stack ;Initilize
MVI A, 91h
STA PortCMD
MVI E, 1
MVI D, 1

MainLoop: MVI A, 92h ;Entry#1
STA PortB
MVI A, 10h
STA PortCUpper
MOV B, E ;GrTime
CALL dly1S

MVI A, 92h
STA PortB
MVI A, 20h
STA PortCUpper
MOV B, D ;YelTime
CALL dly1S

MVI A, 90h ;Entry#2
STA PortB
MVI A, C0h
STA PortCUpper
MOV B, E ;GrTime
CALL dly1S

MVI A, 91h
STA PortB
MVI A, 40h
STA PortCUpper
MOV B, D ;YelTime
CALL dly1S

MVI A, 86h ;Entry#3
STA PortB
MVI A, 40h
STA PortCUpper
MOV B, E ;GrTime
CALL dly1S
MVI A, 8Ah
STA PortB

```

صفحه ۴

بسمه تعالی

سوال‌های امتحان آخر ترم درس ریزپردازنده گروه کامپیوتر

دانشگاه آزاد اسلامی واحد جنوب تهران - ترم اول سال تحصیلی ۸۶-۸۵

کد درس: ۱۹۲۰ مدت امتحان: ۲ ساعت تاریخ امتحان: ۱۳۸۵/۱۰/۲۷

امتحان جزوه بسته است و ماشین حساب لازم نیست



```

MVI A,40h
STA PortCUpper
MOV B,D ;YelTime
CALL dly1S

MVI A,32h ;Entry#4
STA PortB
MVI A,40h
STA PortCUpper
MOV B,E ;GrTime
CALL dly1S

MVI A,52h
STA PortB
MVI A,40h
STA PortCUpper
MOV B,D ;YelTime
CALL dly1S
JMP MainLoop

GrYelTime1:
LDA GrYelFlg
ADI 0
JZ Green
Yellow: INC D
RET
Green: INC E
RET

GrYelTime2:
LDA GrYelFlg
ADI 0
JZ Green
Yellow: DCR D
RET
Green: DCR E
RET

GrYelKey: LDA GrYelFlg *
XOR FFh *
STA GrYelFlg *
RET *

dly1S: ... ;F=10MHz

RET

GrYelFlg: ORG RAM
DB 0

END

```

بسمه تعالی



سوالات امتحان آخر ترم درس ریزپردازنده کد: ۱۹۲۰ گروه کامپیوتر استاد فاتحی
 دانشگاه آزاد اسلامی واحد تهران جنوب - ترم تابستان سال تحصیلی ۸۷-۸۶
 مدت امتحان: ۲ ساعت تاریخ امتحان: ۱۳۸۶/۱۱/۱۵ جزوه بسته و ماشین حساب لازم نیست

نام و نام خانوادگی:

تذکر ۱: برگه سوال، ضمیمه پاسخنامه خواهد بود. می‌توانید به برخی از سوالات در برگه سوال جواب دهید.

تذکر ۲: برای کاهش قلم خوردگی توصیه می‌شود با مداد پاسخ دهید.

تذکر ۳: نمره نهایی از ۱۱۰ نمره این امتحان و ۱۰ نمره کلاس مشخص خواهد شد.

۱- پایه‌های وقفه و عکس‌العمل میکرو را در ریزپردازنده‌های Z80 و 8086 توضیح دهید. (۱۰ نمره)

۲- روش‌های آدرس‌دهی دستگاه‌های جانبی Isolated و Memory Mapped را در مقایسه باهم توضیح دهید. (۱۰ نمره)

۳- دستورالعمل‌های زیر دارای چه ماشین‌سایکل‌هایی هستند آنها را به ترتیب اجرا ذکر کرده و بنویسید در هر کدام از سیکل ماشین‌ها دقیقاً چه کاری انجام می‌شود. (۱۵ نمره) RET و IN 80h

۴- با توجه به مدار ضمیمه به سوالات زیر با ذکر دلیل کافی پاسخ دهید. (هر کدام ۵ نمره = ۱۵ نمره)
 a. محدوده و اندازه آدرس‌های هر یک از خروجی‌های دیکودرها را به طور مشخص بنویسید.
 b. آدرس پورت‌های ۸۲۵۵ را به صورت دقیق مشخص کنید. (آدرس‌های تکراری مشخص شود)
 c. کلمه برنامه ریزی ۸۲۵۵ را بدست آورده و دستورات برنامه ریزی متناسب را بنویسید.

۵- با توجه به نرم افزار پیوست (کنترل چهار راه ۴ زمانه) به سوالات زیر پاسخ دهید. (۳۰ نمره)
 a. به سه پایه وقفه سه کلید وصل شده است عملکرد آنها را نسبت به هم توضیح دهید.
 b. زمان چراغهای سبز، زرد و قرمز هر ورودی به چه ترتیبی مشخص و کنترل می‌شود.
 c. روتین تاخیر B ثانیه خالی می‌باشد آنرا بطور کامل نوشته و زمان تاخیر آنرا برای B ثانیه تنظیم کنید. (f=10MHz)
 d. ترجمه زبان ماشین ۴ خط ستاره‌دار را بطور کامل و دقیق بنویسید.

۶- قطعه برنامه‌ای (زیربرنامه‌ای) بنویسید که مقدار پارامتر ورودی در رجیستر B را که از ۰۰ تا ۹۹ می‌باشد را در مبنای ۱۰ در دو عدد نمایشگر 7-Seg در آدرس 80h (یکان) و 81h (دهگان) بنویسد. (۲۰ نمره)

۷- شکل بلوکی یک میکروکنترلر نمونه را ترسیم کرده اجزا و قابلیت‌های آنرا توضیح دهید. (۱۰ نمره)

بسمه تعالی

سوالات امتحان آخر ترم درس ریزپردازنده کد: ۱۹۲۰ گروه کامپیوتر استاد فاتی

دانشگاه آزاد اسلامی واحد تهران جنوب - ترم تابستان سال تحصیلی ۸۷-۸۶

مدت امتحان: ۲ ساعت تاریخ امتحان: ۱۳۸۶/۱۱/۱۵ جزوه بسته و ماشین حساب لازم نیست



8085 Instruction Set

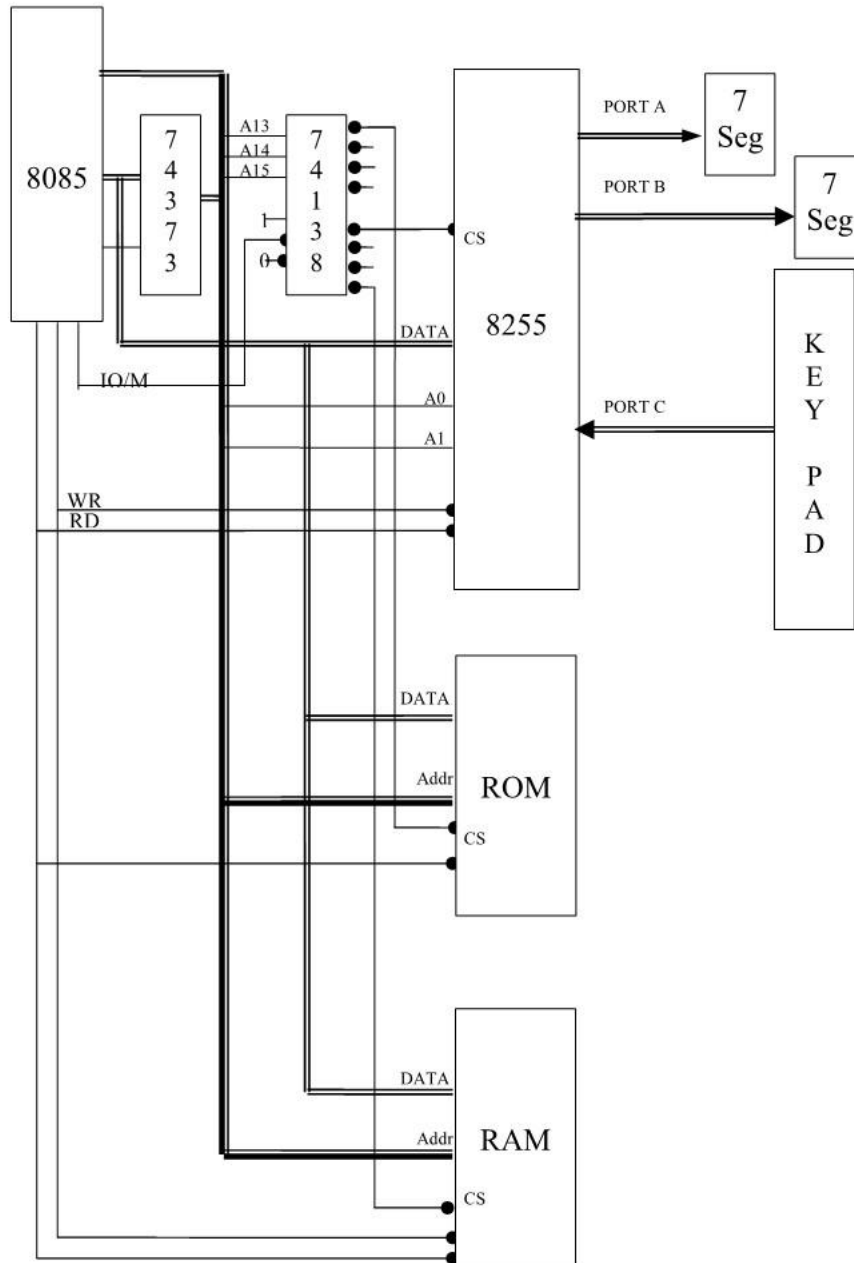
Mnemonic	Op	SZAPC	-s	Description	Notes	Mnemonic	Op	SZAPC	-s	Description	Notes
ACI n	ICE	*****	7	Add with Carry Immediate	A=A+n+CY	PUSH B	C5	-----	12	Push BC	-[SP]=BC
ADC r	8F	*****	4	Add with Carry	A=A+r+CY (21X)	PUSH D	D5	-----	12	Push DE	-[SP]=DE
ADC M	8E	*****	7	Add with Carry to Memory	A=A+[HL]+CY	PUSH H	E5	-----	12	Push HL	-[SP]=HL
ADD r	87	*****	4	Add	A=A+r (20X)	PUSH PSW	F5	-----	12	Push Processor Status Word	-[SP]=(PSW,A)
ADD M	86	*****	7	Add to Memory	A=A+[HL]	RAL	17	-----*	4	Rotate Accumulator Left	A=[CY,A]<-
ADI n	C6	*****	7	Add Immediate	A=A+n	RAR	1F	-----*	4	Rotate Accumulator Right	A=>[CY,A]
ANA r	A7	*****	4	AND Accumulator	A=A&r (24X)	RET	C9	-----	10	Return	PC=[SP]+
ANA M	A6	*****	7	AND Accumulator and Memory	A=A&[HL]	RC	D8	-----	6	Return on Carry	If CY=1 (12-s)
ANI n	E6	*****	7	AND Immediate	A=A&n	RIM	20	-----	4	Read Interrupt Mask	A=mask
CALL a	CD	-----	18	Call unconditional	[-(SP)]=PC, PC=a	RM	F8	-----	6	Return on Minus	If S=1 (12-s)
CC a	DC	-----	9	Call on Carry	If CY=1 (18-s)	RNC	D0	-----	6	Return on No Carry	If CY=0 (12-s)
CM a	FC	-----	9	Call on Minus	If S=1 (18-s)	RNZ	C0	-----	6	Return on No Zero	If Z=0 (12-s)
CMA	2F	-----	4	Complement Accumulator	A=~A	RF	F0	-----	6	Return on Plus	If S=0 (12-s)
CMC	3F	-----	4	Complement Carry	CY=~CY	RPE	E8	-----	6	Return on Parity Even	If P=1 (12-s)
CMP r	BE	*****	4	Compare	A-r (27X)	RPO	E0	-----	6	Return on Parity Odd	If P=0 (12-s)
CMP M	BD	*****	7	Compare with Memory	A-[HL]	RE	C8	-----	6	Return on Zero	If Z=1 (12-s)
CMC a	DA	-----	9	Call on No Carry	If CY=0 (18-s)	RLC	07	-----	4	Rotate Left Circular	A<-A
CNZ a	CA	-----	9	Call on No Zero	If Z=0 (18-s)	RRC	0F	-----*	4	Rotate Right Circular	A->A
CP a	FA	-----	9	Call on Plus	If S=0 (18-s)	RST z	C7	-----	12	Restart (3X7)	[-(SP)]=PC, PC=z
CPE a	EC	-----	9	Call on Parity Even	If P=1 (18-s)	SBB r	9F	*****	4	Subtract with Borrow	A=A-r-CY
CPI n	FE	*****	7	Compare Immediate	A-n	SBB M	9E	*****	7	Subtract with Borrow	A=A-[HL]-CY
CPO a	EA	-----	9	Call on Parity Odd	If P=0 (18-s)	SBI n	DE	*****	7	Subtract with Borrow Immed	A=A-n-CY
CZ a	CC	-----	9	Call on Zero	If Z=1 (18-s)	SHLD a	22	-----	16	Store HL Direct	[a]=HL
DAA	27	*****	4	Decimal Adjust Accumulator	A=BCD format	SIM	30	-----	4	Set Interrupt Mask	mask=A
DAD B	09	-----	10	Double Add BC to HL	HL=HL+BC	SPHL	F9	-----	6	Move HL to SP	SP=HL
DAD D	19	-----	10	Double Add DE to HL	HL=HL+DE	STA a	32	-----	13	Store Accumulator	[a]=A
DAD H	29	-----	10	Double Add HL to HL	HL=HL+HL	STAX B	02	-----	7	Store Accumulator indirect	[BC]=A
DAD SP	39	-----	10	Double Add SP to HL	HL=HL+SP	STAX D	12	-----	7	Store Accumulator indirect	[DE]=A
DCR r	3D	*****	4	Decrement	r=r-1 (0X5)	STC	37	-----	4	Set Carry	CY=1
DCR M	35	*****	10	Decrement Memory	[HL]=[HL]-1	SUB r	97	*****	4	Subtract	A=A-r (22X)
DCX B	08	-----	6	Decrement BC	BC=BC-1	SUB M	96	*****	7	Subtract Memory	A=A-[HL]
DCX D	18	-----	6	Decrement DE	DE=DE-1	SUI n	D6	*****	7	Subtract Immediate	A=A-n
DCX H	28	-----	6	Decrement HL	HL=HL-1	XCHG	E3	-----	4	Exchange HL with DE	HL<->DE
DCX SP	38	-----	6	Decrement Stack Pointer	SP=SP-1	XRA r	AF	*****	4	Exclusive OR Accumulator	A=A&r (25X)
DI	F3	-----	4	Disable Interrupts		XRA M	AE	*****	7	Exclusive OR Accumulator	A=A&[HL]
EI	FB	-----	4	Enable Interrupts		XRI n	EE	*****	7	Exclusive OR Immediate	A=A&n
HLT	76	-----	5	Halt		XTHL	E3	-----	16	Exchange stack Top with HL	[SP]<->HL
IN p	DB	-----	10	Input	A=[p]						
INR r	3C	*****	4	Increment	r=r+1 (0X4)	PSW			*01	Flag unaffected/affected/reset/set	
INR M	3C	*****	10	Increment Memory	[HL]=[HL]+1	S			S	Sign (Bit 7)	
INX B	03	-----	6	Increment BC	BC=BC+1	Z			Z	Zero (Bit 6)	
INX D	13	-----	6	Increment DE	DE=DE+1	AC			A	Auxiliary Carry (Bit 4)	
INX H	23	-----	6	Increment HL	HL=HL+1	P			P	Parity (Bit 2)	
INX SP	33	-----	6	Increment Stack Pointer	SP=SP+1	CY			C	Carry (Bit 0)	
JMP a	C3	-----	7	Jump unconditional	PC=a						
JC a	DA	-----	7	Jump on Carry	If CY=1 (10-s)	a p				Direct addressing	
JM a	FA	-----	7	Jump on Minus	If S=1 (10-s)	M z				Register indirect addressing	
JNC a	D2	-----	7	Jump on No Carry	If CY=0 (10-s)	n nn				Immediate addressing	
JNE a	C2	-----	7	Jump on No Zero	If Z=0 (10-s)	r				Register addressing	
JP a	F2	-----	7	Jump on Plus	If S=0 (10-s)						
JPE a	EA	-----	7	Jump on Parity Even	If P=1 (10-s)	DB n,(n)				Define Byte(s)	
JPO a	E2	-----	7	Jump on Parity Odd	If P=0 (10-s)	DB 'string'				Define Byte ASCII character string	
JZ a	CA	-----	7	Jump on Zero	If Z=1 (10-s)	DS nn				Define Storage Block	
LDA a	3A	-----	13	Load Accumulator direct	A=[a]	DW nn,(nn)				Define Word(s)	
LDAX B	0A	-----	7	Load Accumulator indirect	A=[BC]						
LDAX D	1A	-----	7	Load Accumulator indirect	A=[DE]	A B C D E H L				Registers (8-bit)	
LHLD a	2A	-----	16	Load HL Direct	HL=[a]	BC DE HL				Register pairs (16-bit)	
LXI B,nn	01	-----	10	Load Immediate BC	BC=nn	PC				Program Counter register (16-bit)	
LXI D,nn	11	-----	10	Load Immediate DE	DE=nn	PSW				Processor Status Word (8-bit)	
LXI H,nn	21	-----	10	Load Immediate HL	HL=nn	SP				Stack Pointer register (16-bit)	
LXI SP,nn	31	-----	10	Load Immediate Stack Ptr	SP=nn						
MOV r1,r2	7F	-----	4	Move register to register	r1=r2 (1XX)	a nn				16-bit address/data (0 to 65535)	
MOV M,r	77	-----	7	Move register to Memory	[HL]=r (16X)	n p				8-bit data/port (0 to 255)	
MOV r,M	7E	-----	7	Move Memory to register	r=[HL] (1X6)	r				Register (X=B,C,D,E,H,L,M,A)	
MVI r,n	3E	-----	7	Move Immediate	r=n (0X6)	x				Vector (X=0H,8H,10H,18H,20H,28H,30H,38H)	
MVI M,n	36	-----	10	Move Immediate to Memory	[HL]=n	+ -				Arithmetic addition/subtraction	
NOP	00	-----	4	No Operation		& -				Logical AND/NOT	
ORA r	B7	*****	4	Inclusive OR Accumulator	A=A r (26X)	v x				Logical inclusive/exclusive OR	
ORA M	B6	*****	7	Inclusive OR Accumulator	A=A [HL]	<- ->				Rotate left/right	
ORI n	F6	*****	7	Inclusive OR Immediate	A=A n	<->				Exchange	
OUT p	D3	-----	10	Output	[p]=A	[]				Indirect addressing	
PCHL	E9	-----	6	Jump HL indirect	PC=[HL]	[] + - []				Indirect address auto-inc/decrement	
POP B	CD	-----	10	Pop BC	BC=[SP]+	{ }				Combination operands	
POP D	DD	-----	10	Pop DE	DE=[SP]+	{ (X)				Octal op code where X is a 3-bit code	
POP H	ED	-----	10	Pop HL	HL=[SP]+	If (~s)				Number of cycles if condition true	
POP PSW	F1	-----	10	Pop Processor Status Word	{PSW,A}=[SP]+						

بسمه تعالی

سوالات امتحان آخر ترم درس ریزپردازنده کد: ۱۹۲۰ گروه کامپیوتر استاد فاتحی

دانشگاه آزاد اسلامی واحد تهران جنوب - ترم تابستان سال تحصیلی ۸۶-۸۷

مدت امتحان: ۲ ساعت تاریخ امتحان: ۱۳۸۶/۱۱/۱۵ جزوه بسته و ماشین حساب لازم نیست



بسمه تعالی



سوالات امتحان آخر ترم درس ریزپردازنده کد: ۱۹۲۰ گروه کامپیوتر استاد فاتحی

دانشگاه آزاد اسلامی واحد تهران جنوب - ترم تابستان سال تحصیلی ۸۷-۸۶

مدت امتحان: ۲ ساعت تاریخ امتحان: ۱۳۸۶/۱۱/۱۵ جزوه بسته و ماشین حساب لازم نیست

```

PortA      EQU 8000h
PortB      EQU 8001h
PortCLower EQU 8002h
PortCUpper EQU 8002h
PortCMD    EQU 8003h
RAM        EQU E000h
Stack      EQU FFFFh

        JMP Start
        ORG 36          ;24h
        JMP GrTime
        ORG 44          ;2Ch
        JMP YelTime
        ORG 52          ;34h
        JMP GrYelKey

Start:   LXI SP, Stack ;Initilize
        MVI A, 91h
        STA PortCMD
        MVI E, 1
        MVI D, 1
        EI

MainLoop: MVI A, 92h   ;Entry#1
        STA PortB
        MVI A, 10h
        STA PortCUpper
        MOV B, E      ;GrTime
        CALL dlynS

        MVI A, 92h
        STA PortB
        MVI A, 20h
        STA PortCUpper
        MOV B, D      ;YelTime
        CALL dlynS

        MVI A, 90h   ;Entry#2
        STA PortB
        MVI A, C0h
        STA PortCUpper
        MOV B, E      ;GrTime
        CALL dly1S

        MVI A, 91h
        STA PortB
        MVI A, 40h
        STA PortCUpper
        MOV B, D      ;YelTime
        CALL dlynS

        MVI A, 86h   ;Entry#3
        STA PortB
        MVI A, 40h
        STA PortCUpper
        MOV B, E      ;GrTime
        CALL dlynS
        MVI A, 8Ah
        STA PortB
        MVI A, 40h
        STA PortCUpper

```

صفحه ۴

بسمه تعالی

سوالات امتحان آخر ترم درس ریزپردازنده کد: ۱۹۲۰ گروه کامپیوتر استاد فاتحی

دانشگاه آزاد اسلامی واحد تهران جنوب - ترم تابستان سال تحصیلی ۸۷-۸۶

مدت امتحان: ۲ ساعت تاریخ امتحان: ۱۳۸۶/۱۱/۱۵ جزوه بسته و ماشین حساب لازم نیست



```

MOV B,D ;YelTime
CALL dlynS

MVI A,32h ;Entry#4
STA PortB
MVI A,40h
STA PortCUpper
MOV B,E ;GrTime
CALL dlynS

MVI A,52h
STA PortB
MVI A,40h
STA PortCUpper
MOV B,D ;YelTime
CALL dlynS
JMP MainLoop

GrTime:
LDA IncDcrFlg
ADI 0
JZ GrInc
DCR E
RET

GrInc:
INC E
RET

YelTime:
LDA IncDcrFlg
ADI 0
JZ YelInc
DCR D
RET

YelInc:
INC D
RET

GrYelKey:
LDA IncDcrFlg *
XRA FFh *
STA IncDcrFlg *
RET *

dlynS:
... ;F=10MHz

RET

IncDcrFlg:
ORG RAM
DB 0

END

```