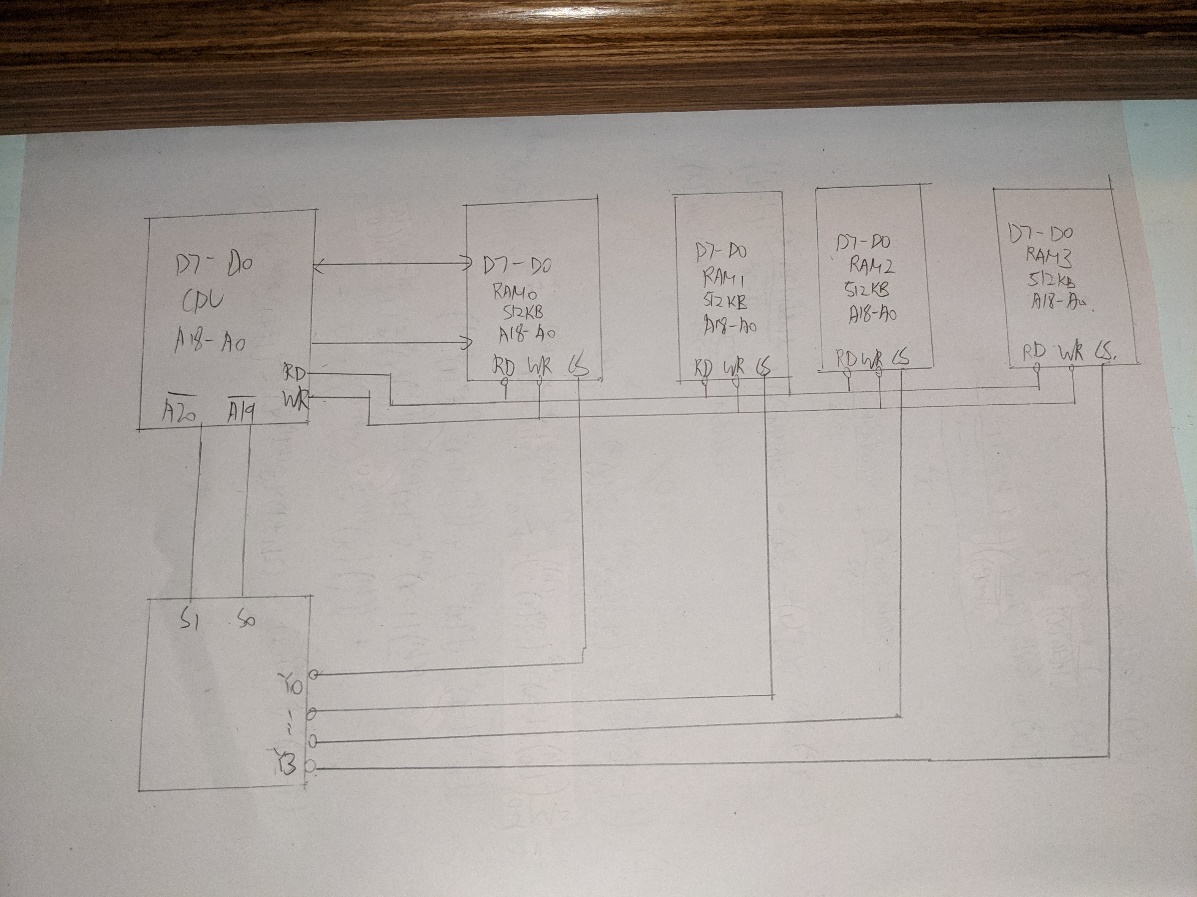
EE2004 Exam answer sheet

SID: 56046680

1. 2. 1. active high memory read, memory write and chip select signals







The circuit will set the address of the memory location which is 80000h for A and enable the chip enable. The circuit will set read signal.

The memory chip then puts the content of the selected location on D0 D7, now external circuit can access the data from D0 D7 and the CPU can get number A.





The circuit will set the address of the memory location which is 80002h for C and enable the chip enable. The circuit will set write signal.

External circuit puts the data on D0 D7 and save the incoming data to the required location.



DELAY:

movlw 0x3e ;62

movwf R1

AGAIN:

movlw 0x9a ;154

movwf R2

HERE:

nop

nop

decf R2, f

bnz HERE

decf R1, f

bnz AGAIN

nop

nop

nop

nop

nop

nop

nop

nop

nop

nop

return



bcf TRISB, 2

movlw 0x02

movwf T0CON

HERE:

movlw 0x8a

movwf TMR0H

movlw 0xd0

movwf TMR0L

bcf INTCON, TMR0IF

call DELAY

btg PORTB, 2

bra HERE

DELAY:

bsf T0CON, TMR0ON

AGAIN:

btfss INTCON, TMR0IF

bra AGAIN

bcf T0CON, TMR0ON

return



From 00000 0000 to 0000 0011



LIST P=18F4520 ;directive to define processor

#include <P18F4520.INC> ;CPU specific variable definitions

ORG 0x0000

cblock 0x10

char\_A1, char\_A2 ,char\_A3 ,char\_A4 ,char\_A5

endc

cblock 0x90

char\_B1 ,char\_B2 ,char\_B3 ,char\_B4 ,char\_B5

endc

cblock 0x70

char\_C1 ,char\_C2 ,char\_C3 ,char\_C4 ,char\_C5

endc

movf char\_A1, w

subwf char\_B1, w

movwf char\_C1

movf char\_A2, w

subwf char\_B2, w

movwf char\_C2

movf char\_A3, w

subwf char\_B3, w

movwf char\_C3

movf char\_A4, w

subwf char\_B4, w

movwf char\_C4

movf char\_A5, w

subwf char\_B5, w

movwf char\_C5

END



LIST P=18F4520 ;directive to define processor

#include <P18F4520.INC> ;CPU specific variable definitions

ORG 0x0000

count EQU 0x05

cblock 0x10

char\_A1, char\_A2 ,char\_A3 ,char\_A4 ,char\_A5

endc

cblock 0x90

char\_B1 ,char\_B2 ,char\_B3 ,char\_B4 ,char\_B5

endc

cblock 0x70

char\_C1 ,char\_C2 ,char\_C3 ,char\_C4 ,char\_C5

endc

lfsr 0, char\_A1

lfsr 1, char\_B1

lfsr 2, char\_C1

movlw d'5'

movwf count

Loop:

movf INDF0, w

addwf INDF1, w

movwf INDF2

incf FSR0L, f

incf FSR1L, f

incf FSR2L, f

decf count, f

bnz Loop

END



LIST P=18F4520 ;directive to define processor

#include <P18F4520.INC> ;CPU specific variable definitions

ORG 0x0000

count EQU 0x05

cblock 0x10

char\_A1, char\_A2 ,char\_A3 ,char\_A4 ,char\_A5

endc

cblock 0x90

char\_B1 ,char\_B2 ,char\_B3 ,char\_B4 ,char\_B5

endc

cblock 0x70

char\_C1 ,char\_C2 ,char\_C3 ,char\_C4 ,char\_C5

endc

lfsr 0, char\_A1

lfsr 1, char\_B1

lfsr 2, char\_C1

setf TRISB

Main:

btfss PORTB, 0

call ADD

btfss PORTB, 1

call SUB

goto Main

ADD:

movlw d'5'

movwf count

Loopadd:

movf INDF0, w

addwf INDF1, w

movwf INDF2

incf FSR0L, f

incf FSR1L, f

incf FSR2L, f

decf count, f

bnz Loopadd

return

SUB:

movlw d'5'

movwf count

Loopsub:

movf INDF0, w

subwf INDF1, w

movwf INDF2

incf FSR0L, f

incf FSR1L, f

incf FSR2L, f

decf count, f

bnz Loopsub

return

END