92

Name: Ameer Nasrallah

Student ID #: 1090338

Birzeit University - Faculty of Information Technology
Computer Systems Engineering Department – ENCS531
2nd semester - 2012/13 - second hour exam – duration (90 minutes)- 30 /4/2013

Real-Time Applications and Embedded Systems

Instructor: Dr. Ahmad Afaneh

Q1. Select the most correct answer. Write down your final answers in the following table. (30 points)

30

1	2	3	4	5	6
c	a	a	d	a	b

1) Marshalling in RPC is performed by

a. Client code
b. server code
c. stubs
d. all of the above

2) The Look up table cab be used to store data in

a. program memory
b. data memory
c. both program and data memories
d. none of the above

3) The PIC memory architecture is based on the

a. Harvard Architecture
b. Von-Neuman Architecture
c. both
d. none of the above

4) In order to enable timer1 interrupt you should enable

a. global interrupts
b. peripheral interrupt
c. Timer1 interrupt
d. All of the above

5) If the external oscillator frequency is 4MHz most instructions on average will need

a. 1 us b. 4 us c. 0.25 us d. 2 s

6) The 16f877 uses a

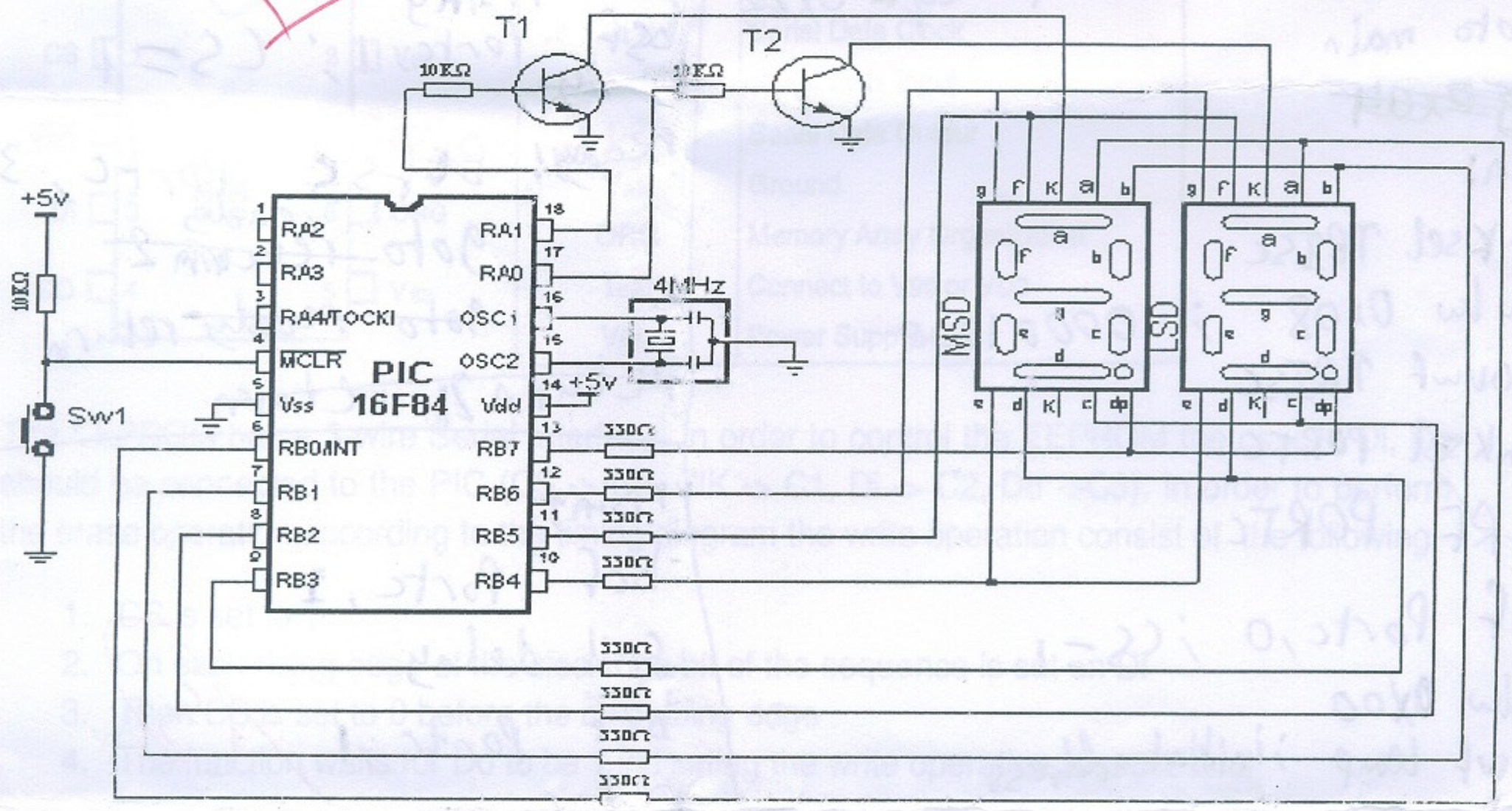
a. 12 bit instructions
b. 14 bit instructions
c. 16 bit instructions
d. all of the above

Name: Ameer Nasrallah

Student ID #: 1090338

Q2. Using the following schematics write a function display number to display a two digit number on the 7 segment displays assume the number is written as BCD in W(e.g. 45 w=0100 0101 . use timer0 interrupt to implement the delay(35 points)

27



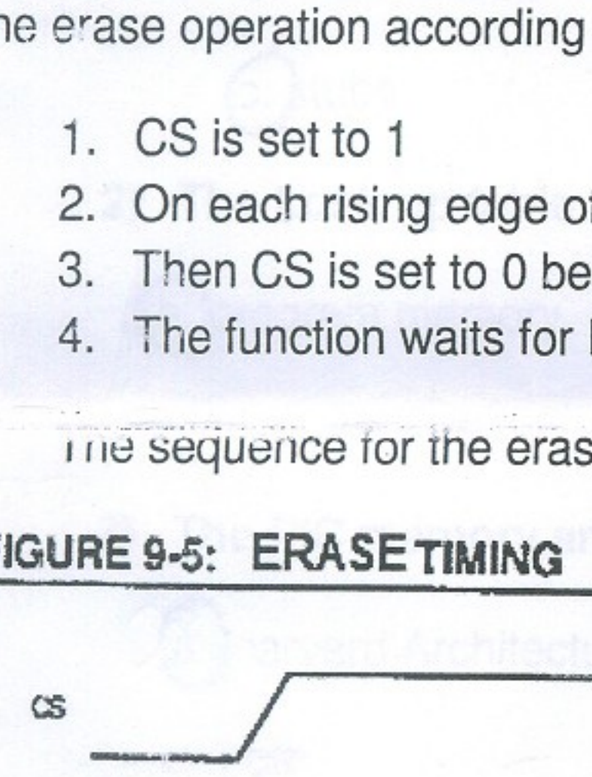

```
PIC 16F877
_CONFIG  0x3731
include "PIC16F877A.INC"
org 0x00
goto main
org 0x04

BCF INTCON,2
RETIE
temp equ 0x20
num1 equ 0x21
num2 equ 0x22      key equ 0x23
main:
movwf temp          ;save initial value
movwf num1
movlw 0x0F
ANDWF num1, 1       ; contains first digit
Banksel TRISA
movlw 0x00
movwf TRISA
Banksel TRISB
movlw 0x00
movwf TRISB
Banksel PortA
CLRF PortA
BSF PortA,0 ;Enable first 7-seg
movf num1, w
movwf key
call seven_seg
Banksel PortB
movwf PortB
Swapf temp,0
movwf num2
movlw 0x0F
andwf num1, 1 ;contains second digit
call delay
Banksel PortA
Bcf PortA,0
Bsf PortA,1
movf num2,w
movwf key
call seven-seg
Banksel PortB
movwf PortB

delay:
movlw 0x07
movwf OPTREG
movlw 0xA0
movwf INTCON
movlw D'61'
Banksel TMR0
movwf TMR0

wait: BTFSS
BTFSS INTCON,2
goto wait
return

seven_seg:
movf key,w
addwf PCL ;0
nop
retlw '00001100' ;9
retlw
```

Name: Ameer Nasrallah

Student ID #: 1090338

Q2. Write a function to perform the write operation on the 93C56 - Serial EEPROM (40 points)

35



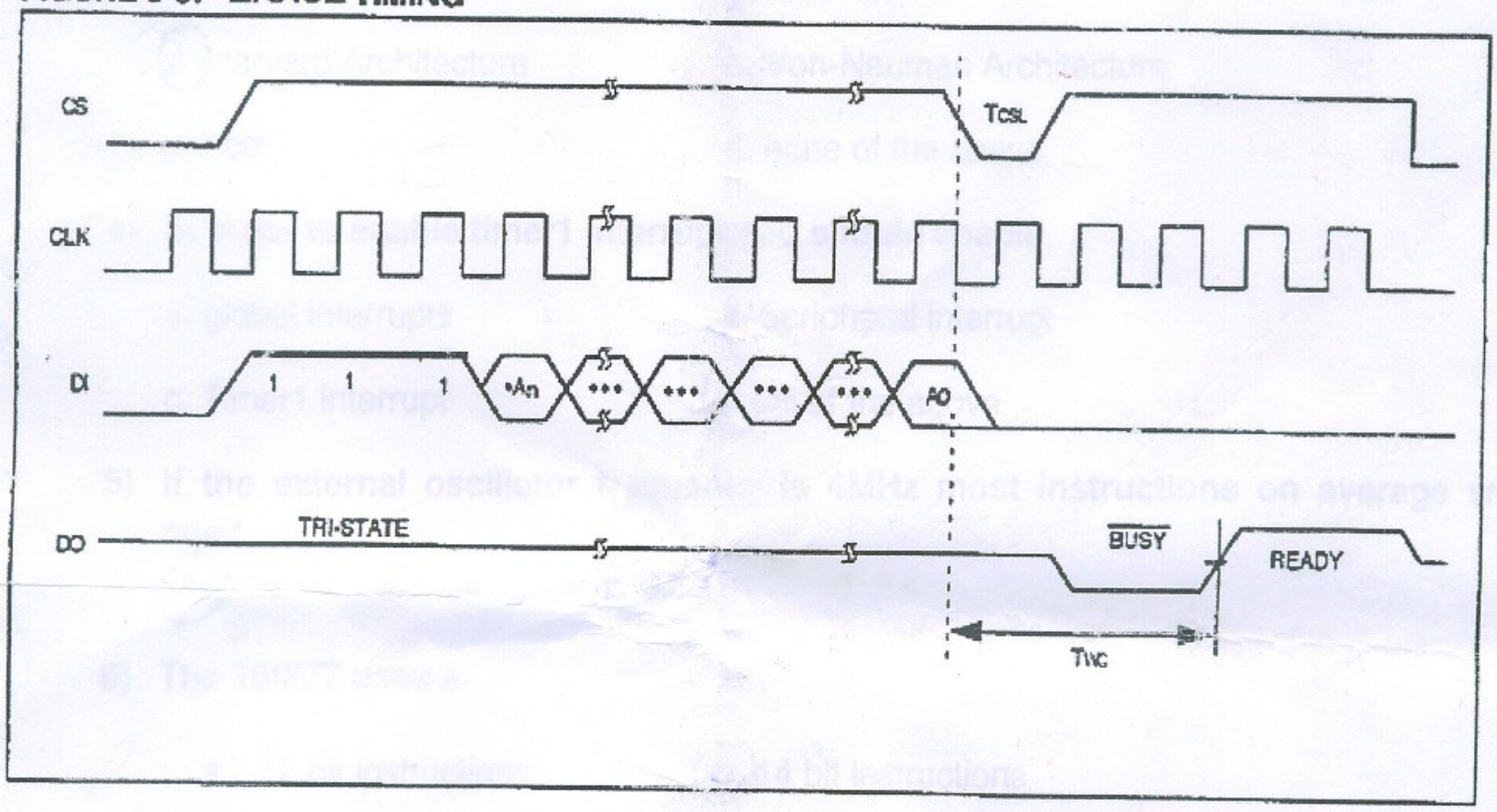
Name	Function
CS	Chip Select
CLK	Serial Data Clock
DI	Serial Data Input
DO	Serial Data Output
Vss	Ground
ORG	Memory Array Organization
Test	Connect to Vss or Vcc
Vcc	Power Supply +5V

The EEPROM has a 3-wire Serial Interface. In order to control the EEPROM the cs, clk, Di, Do should be connected to the PIC (CS -> C0, CIK -> C1, Di -> C2, Do ->C3). In order to perform the erase operation according to the timing diagram the write operation consist of the following

1. CS is set to 1
2. On each rising edge of the clock one bit of the sequence is set on DI
3. Then CS is set to 0 before the next rising edge
4. The function waits for Do to be 1 indicating the write operation is successful

The sequence for the erase operation **111<8 bit address A7to A0>**

FIGURE 9-5: ERASE TIMING



Write a function to write the erase the whole EEPROM (0x00 to 0xFF)

Good Luck

PIC 16F877

-Config 0x3731

include "PIC16F877A.INC"

temp equ 0x20

count equ 0x21

Count2 equ 0x22

temp2 equ 0x23

Org 0x00

goto main

~~Org 0x04~~

main:

Banksel TRISC

movlw 0x08 ; 00001000

BMOVWF TRISC

Banksel PORTC

CLRF PORTC

Bsf Portc,0 ;CS=1

movlw 0x00

movwf temp ;initial address

MOVLW 0x0FF

movwf count ;number of addresses

write; call write-one address

incf temp,1

decfsz count,1

goto write

goto end1

~~End~~

Write-one address:

~~call rising~~

movlw 0x07

movwf count2

~~movlw 0x07~~

~~Call rising~~

Command: call ~~write-bit~~

call ~~rising~~ write-bit ;write one bit to D;

decfsz count2,1 ; number of bits to be transferred

goto command

movlw 0x08

movwf count2

~~movf temp,0~~ ~~movwf t~~ movf temp,0 ;address to be sent

~~address: call rising~~

movwf temp2

address: call rising

~~movf temp2~~

Call write_bit

decfsz count2,1

goto address

bcf Portc,0 ;CS=0

call rising

bsf Portc,1 ; CS=1

~~ready:~~

ready: Btfss Portc,3 ;POLL D0

goto ready

~~goto return2~~

~~goto ready~~ return

~~return2:~~ return

rising:

bcf Portc,1

call delay

bsf Portc,1

return

write_bit:

RRF temp2,1 ; rotate write

BTFSC STATUS,C ;if(C=0)

goto write-one ; send 0

~~goto~~ bcf PortC,2 ;if(C=1)

return ; send 1

write-one: bsf PortC,2

return.

end1: end