EMK 310

Microprocessors

Practical 2

Serial Communication Between

PIC, PC & EEPROM

Student Name	Student Number	Contribution
MT Mokgotho	[redacted]	25%
F Botha	29042951	25%
S Dinath	[redacted]	25%
S Taghizadeh	[redacted]	25%

May 2012

Table of Contents

1. **Introduction**.....4
2. **Objectives**.....4
3. **Problem Statement**.....5
4. **Apparatus**.....5
5. **Flowchart of the Software**.....6
6. **System diagrams**.....8
7. **System description**.....9
8. **Circuit Design**.....10
9. **Code**.....12
10. **References**.....46

1. Introduction

This practical is about familiarizing the students with different ways of communication. In this practical we designed a system that makes a communication between PIC18F13K50 that execute a USB to serial bridge to make a communication between a PC and PIC18F45K20. Hyper terminal (Windows based software) is used to communicate between the PIC18F45K20 and the serial port of PC through the USB-RS232 Bridge that will be implemented on the PIC18F13K50. There must be another communication between the PIC18F45K20 and an external EEPROM that uses the I^2C protocol. The I^2C link must be hardcoded, the onboard I^2C functionality of the PIC cannot be used.

2. Objectives

The objectives of this practical for students are to get familiar with the communication between

- The PIC and a PC.
- An external EEPROM and the PIC.

3. Problem Statement

Design a system that uses a PIC18F13K50 to implement a USB to serial bridge to facilitate communication between a PC and the EUSART of the PIC18F45K20. There must be a communication between the PIC18F 45K20 and an external EEPROM using I^2C.

4. Apparatus

- PIC18F45K20
- PIC18F13K50
- External serial memory device, e.g Atmel AT24C01A
- External EEPROM
- PC
- Various resistors and capacitors

5. Flowchart of the Software

i. Flowchart description

The flowchart is shown in figure 1 below. The user enters a command; the program waits until the command gets in through the Hyper Terminal and then checks the type of the command. If the command is the word “Names” then the names of the group members will be displayed. If the command is “New” then the program allows one to enter a new names and contact. Then these new data will be stored in EEPROM for retrieving later. If the command is [any letter] then the names starting with that specific letter will be displayed. This is the state that the data is retrieved back from the EEPROM. The last state is the “Delete” state, which the information is erased from the EEPROM.

ii. Flowchart

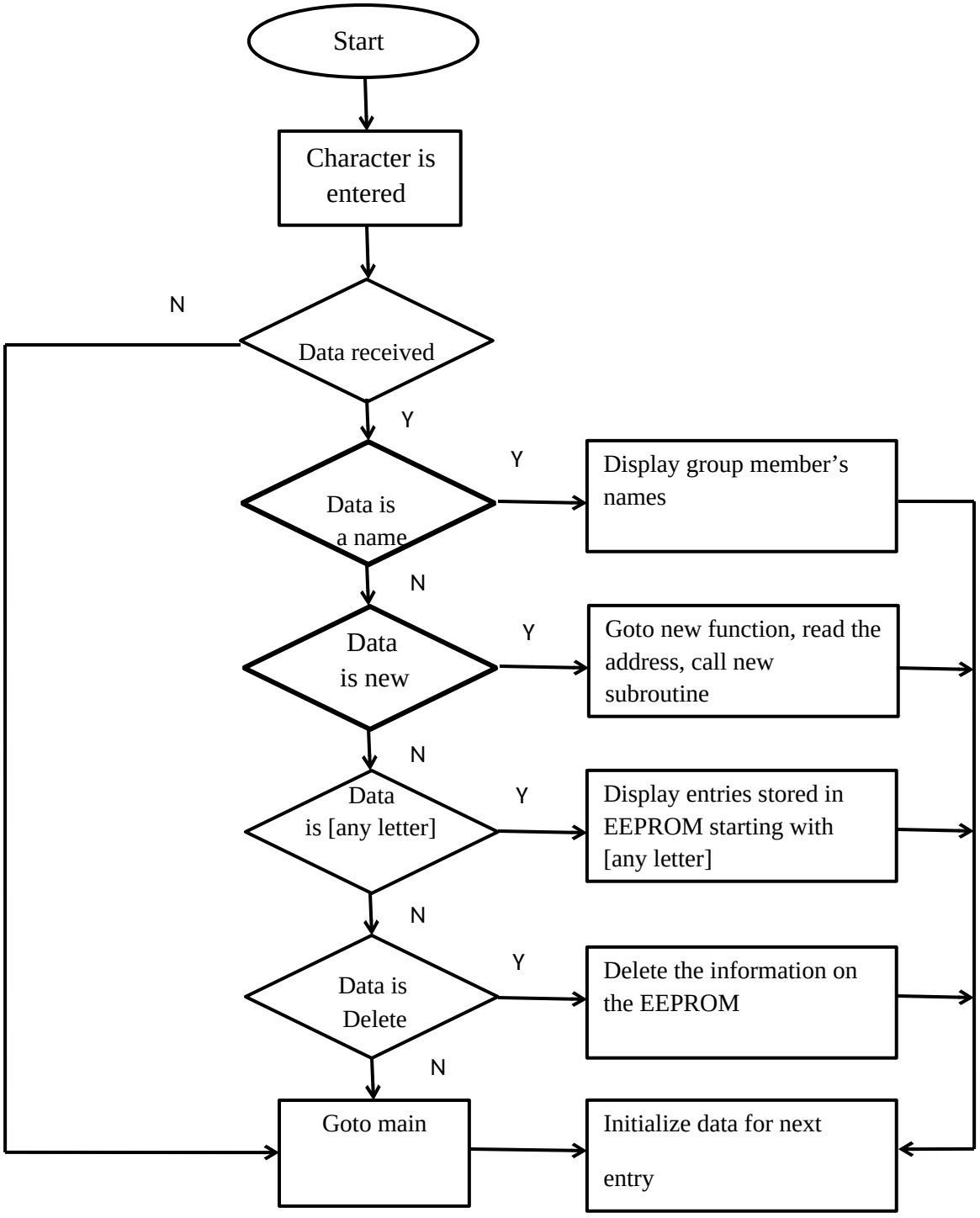


Figure 2 – Flowchart

6. System diagrams

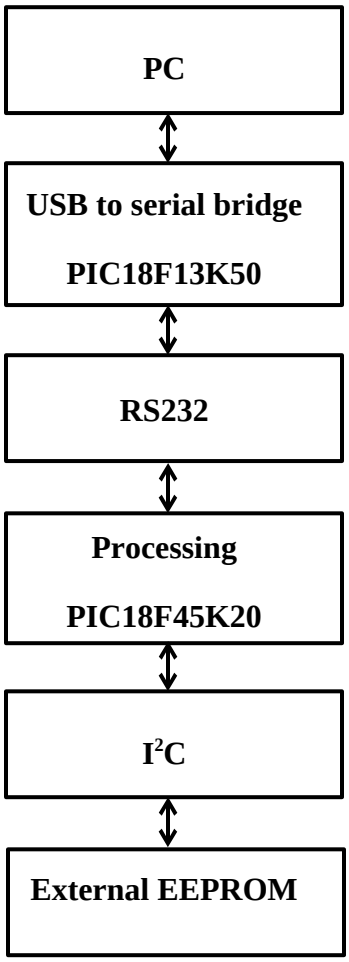


Figure 3 – System diagram indicating the communications

7. System description

The PC access the EEPROM through the two PICs. The system must be able to display the data that is stored on the EEPROM on the PC and also store data in the EEPROM by entering data using serial communication application such as Hyperterminal. The following commands will be used by the system:

- "**Names**": when the command is sent to PIC, it should respond with the names of group members and their student number, each start on the new line from the left of the screen.

- "**New**": when the command is sent, the PIC respond with the massage "Enter name of new contact" that is stored on the PIC and retrieved via Hyper Terminal. The next entry must be written and stored in EEPROM. Next, the PIC must display the massage "ENTER CONTACT NUMBER" that is stored in it and retrieved via Hyper Terminal. The number must be stored on EEPROM in an address following the name entry. Upon entry of the number, the PIC must display the massage "NEW ENTRY SAVED" that has to be stored on the PIC itself. The entry following the massage "ENTER NAME OF NEW CONTACT" must have the maximum size of 15 ASCII characters and after the massage "ENTER CONTACT NUMBER" just numerals maybe accepted and they must be the maximum size of 10 ASCII characters.

- "[**any letter**]": when any single letter is sent to PIC, the entries stored in EEPROM starting with [any letter] must be retrieved via the PIC and displayed in Hyper Terminal

- "**Delete**": when "Delete" command is sent to the PIC, it should empty the EEPROM and send "EEPROM DELETED" over the RS232 interface.

8. Circuit Design

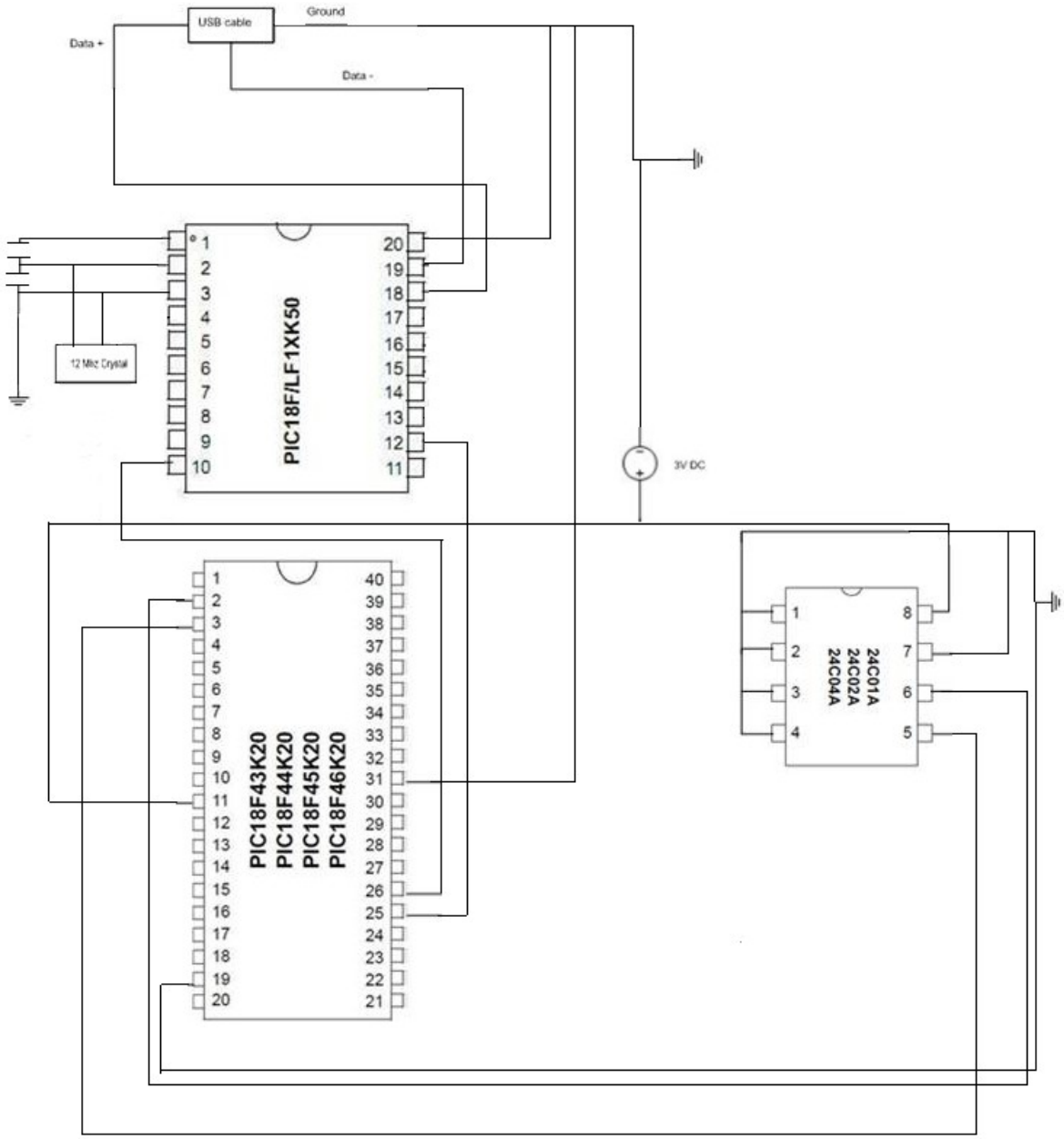


Figure 4-circuit diagram

Circuit description

The use of external memory was to get more I/O ports.

The RS232 is a standard for serial binary data interchange that used as a USB for local communication. A minimal “3-wire” RS232 asynchronous connection used in this practical that consist of transmit data (T_x), receive data (R_x), and ground. In a RS232 asynchronous serial communication a start signal is sent before each byte or code word that prepare the receiving mechanism for the registration and a stop signal is sent after each code which serves to make the receiving mechanism to rest and prepare for the next code word.

The external memory EEPROM uses two bi-directional wires which are for data and clock. It allowes the chips to interconnect to each other over that two wires, and also allows the chips to be added or removed safetly without turning the power off.

The device operation is tested by using Hyper Terminal that is a standard Windows peorgram.

9. Code

```
;MT Mokgotho [redacted]
;F Botha          29042951
;S Dinath         [redacted]
;S Taghizadeh     [redacted]
; 14/05/2012
; this program is to form a brigde to communicate to the usb using the
;  rs232 communication protocol with access to external memory.

        title "Prac_2"
                LIST P=PIC18F45K20
                #include "C:\Program Files (x86)\Microchip\MPASM Suite\p18f45k20.inc"

;_____config bits_____
                config  FOSC=INTIO67
                config  WDTEN=OFF
                config  LVP=OFF
                config  MCLRE=ON


;_____variables_____
                cblock 0x20
                        i2c_count
                        i2c_Data
                        i2c_delaycount
                        i2c_address                     ;count data bit biegn
transmited
                        i2c_config                      ;config i2c
        (0=Stop,1=Change,2=,3=,4=,5=,6=,7=)
                        USART_recieve           ;stores last recieved value
                        USART_flag                      ;bit 0 indicates recieve
                        lenth
                        any_store
                        any_store2
                        offset
                        count
                endc
```

```
;_____________reset vector_____________
                org     00h
                goto    start
;_____________interupt vector_____________
                org     08h
                goto    ISR;


;_____________start_____________________
start
;_____________Setup_____________________
;_____________ocilator__________________
                bsf             OSCCON,IRCF2
                bsf             OSCCON,IRCF1
                bsf             OSCCON,IRCF0
;_____________timer 2 for usart delay__
                bcf             T2CON,TMR2ON
                bsf             T2CON,T2OUTPS0
                bcf             T2CON,T2OUTPS1
                bcf             T2CON,T2OUTPS2
                bcf             T2CON,T2OUTPS3
                bcf             T2CON,T2CKPS0
                bcf             T2CON,T2CKPS1
                bcf             PIR1,TMR2IF
                movlw   0x00
                movwf   TMR2
                movlw   d'100'
                movwf   PR2
;_____________setup usart_______________
                clrf    PORTC
                clrf    ANSEL
                clrf    ANSELH
                clrf    PORTC
                clrf    LATC
                movlw   b'11000000'                     ; Setup port C for serial port.
                                        ; TRISC<7>=1 and TRISC<6>=1.
                movwf   TRISC
                movlw   d'12'                                   ; 19200 BAUD @ 16 MHz
                movwf   SPBRG                                   ; load baudrate register
                bcf     TXSTA,SYNC                      ; enable asynchronous transmission
                bcf     TXSTA,BRGH                      ; Low BAUDrate
```

```
                bcf             BAUDCON,BRG16
                bsf     RCSTA, SPEN                     ; Enable serial port.
                bsf             TXSTA,TXEN
                bsf             RCSTA,CREN
;_____________Setup i2c timer_____________
                bcf             T0CON,TMR0ON                    ;setup tmr0
                bsf             T0CON,T08BIT
                bcf             T0CON,T0CS
                bcf             T0CON,PSA
                bcf             T0CON,T0PS2
                bcf             T0CON,T0PS1
                bcf             T0CON,T0PS0
                movlw  d'236'
                movwf  TMR0L
;_____________Port D i2c_________________
                clrf    TRISD                   ;Clock on D0 data on D1
                clrf    LATD
                setf    PORTD
;_____________set up interrupts_________

                bsf     PIE1,RCIE               ; enable receive interrupt
                bcf             INTCON,TMR0IF                   ; disable timer 1
interupt
                bsf             INTCON,PEIE                     ; Enable peripheral
interrupts
                bsf             INTCON,GIE                      ; Global interrupt
enable

;_____________examples_____________________
                ;movlw b'11110000'               ;value to be written
                ;movwf i2c_Data
                ;movlw 0x00                      ;adress
                ;movwf i2c_address
                ;call    write_i2c               ;call write
                ;call    i2c_delay

                ;clrf    i2c_Data                ;clear buffer file
                ;movlw 0x00                      ;set addres
                ;movwf i2c_address
                ;call    read_i2c                ;call read
```

```
;main
                ;btfss  USART_flag,0            ;poll USART
                ;goto   main
                ;bcf    USART_flag,0

                ;movf   USART_recieve,0                 ;reflect
                ;btfsc  PIR1,TXIF
                ;movwf TXREG
                ;call   USART_delay

                ;goto   main

;_____________MAIN_____________________
main
                movlw  d'7'
                movwf  lenth
                lfsr    0,0x100
                call    display_in
poll

                bsf    PIE1,RCIE
poll1
                btfss   USART_flag,0                    ;poll USART
                goto    poll1
                bcf    PIE1,RCIE
                bcf             USART_flag,0
                movf    USART_recieve,0                         ;reflect
                btfsc   PIR1,TXIF
                movwf  TXREG
                call    USART_delay

                movf    USART_recieve,0
                movwf  POSTINC0

                movlw  0x0A
                cpfseq  USART_recieve
                bra             compare0
                bra             compare1
compare0
                decfsz  lenth
                bra             poll
```

```
                bra             incorrect_

compare1
                movlw   0x0A
                lfsr    0,0x100
                cpfseq  INDF0
                bra             compare2
                bra     incorrect_
compare2
                movlw   'N'
                cpfseq  POSTINC0
                bra             compare5
                bra     compare3
compare3
                movlw   'e'
                cpfseq  INDF0
                bra             compare4
                bra             new_
compare4
                movlw   'a'
                cpfseq  INDF0
                bra             incorrect_
                bra             names_
compare5
                lfsr    FSR0,0x100
                movlw   '['
                cpfseq  INDF0
                bra             compare6
                bra             any_letter_
compare6
                movlw   'D'
                cpfseq  INDF0
                bra             incorrect_
                bra             delete_
delete_
                call    delete
                bra             main
any_letter_
                call    any_letter
                bra             main
names_
```

```
                call    names
                bra             main
new_
                call    new
                bra             main
incorrect_
                bra             main


;_____________ISR_____________________
ISR
                bcf     INTCON,GIE
                btfsc   INTCON,TMR0IF
                bra             tmr0
                btfss   PIR1,RCIF
                bra             exit

                bcf             PIR1,RCIF
                movf    RCREG,0
                movwf   USART_recieve
                bsf             USART_flag,0
                bra             exit
tmr0
                bcf             INTCON,TMR0IF
                movlw   d'236'
                movwf   TMR0L
                incf    i2c_count
one
                movlw   d'1'
                cpfseq  i2c_count
                bra             two
                btg             PORTD,0
                bcf             i2c_config,0

                bra             exit
two
                movlw   d'2'
                cpfseq  i2c_count
                bra             three
                bsf             i2c_config,1
                bra             exit
```

```
three
                movlw  d'3'
                cpfseq  i2c_count
                bra             four
                btg             PORTD,0
                bcf             i2c_config,1
                bra             exit
four
                bsf             i2c_config,0
                movlw  d'0'
                movwf  i2c_count
exit
                bsf     INTCON,GIE
                retfie
;_____________instructions_________
;_____________Names_________________
names
                movlw  0x0A
                btfsc   PIR1,TXIF
                movwf  TXREG
                call    USART_delay

                clrf    i2c_Data                        ;clear buffer file
                                                                ;set addres
                clrf    i2c_address
read_n
                call    read_i2c                        ;call read
                ;delay
                movlw  0x00
                cpfseq  i2c_Data
                bra     next_n

                movlw  0x0A
                btfsc   PIR1,TXIF
                movwf  TXREG
                call    USART_delay
                return
next_n
                movf    i2c_Data,0
                btfsc   PIR1,TXIF
                movwf  TXREG
```

```
                call    USART_delay
                incf    i2c_address
                bra             read_n
;_____________New_____________________
new
                movlw   d'17'
                movwf   lenth
                lfsr    0,0x100
                call    display_new_name
polln
                bsf     PIE1,RCIE
poll1n
                btfss   USART_flag,0            ;poll USART
                goto    poll1n
                bcf     PIE1,RCIE
                bcf             USART_flag,0
                movf    USART_recieve,0                 ;reflect
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    USART_delay

                movf    USART_recieve,0
                movwf   POSTINC0

                movlw   0x0A
                cpfseq  USART_recieve
                bra             comparen
                bra             store
comparen
                decfsz  lenth
                bra             polln
                bra             endn
store
                bcf     PIE1,RCIE
                clrf    i2c_Data                        ;clear buffer file
                                                                ;set addres
                clrf    i2c_address
read_ne
                call    read_i2c                        ;call read
                movlw   0x00
                cpfseq  i2c_Data
```

```
                bra     next_ne
                bra             store00
next_ne
                incf    i2c_address
                bra             read_ne
store00
                lfsr    0,0x100
store0
                movlw   0x0A
                cpfseq  INDF0
                bra             store1
                bra             store2
store1
                movf    POSTINC0,0                      ;value to be written
                movwf   i2c_Data
                call    write_i2c
                call    i2c_delay
                incf    i2c_address
                bra             store0
store2
                movlw   0x0A                    ;value to be written
                movwf   i2c_Data
                call    write_i2c
                call    i2c_delay
                incf    i2c_address
new2
                movlw   d'17'
                movwf   lenth
                lfsr    0,0x100
                call    display_new_no
pollnn
                bsf     PIE1,RCIE
poll1nn
                btfss   USART_flag,0            ;poll USART
                goto    poll1nn
                bcf     PIE1,RCIE
                bcf             USART_flag,0
                movf    USART_recieve,0                 ;reflect
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    USART_delay
```

```
                movf    USART_recieve,0
                movwf   POSTINC0

                movlw   0x0A
                cpfseq  USART_recieve
                bra             comparennn
                bra             storenn
comparennn
                decfsz  lenth
                bra             pollnn
                bra             endn
storenn
                clrf    i2c_Data                        ;clear buffer file
store00nn
                lfsr    0,0x100
store0nn
                movlw   0x0A
                cpfseq  INDF0
                bra             store1nn
                bra             store2nn
store1nn
                movf    POSTINC0,0                      ;value to be written
                movwf   i2c_Data
                call    write_i2c
                call    i2c_delay
                incf    i2c_address
                bra             store0nn
store2nn
                movlw   0x0A                    ;value to be written
                movwf   i2c_Data
                call    write_i2c
                call    i2c_delay
                incf    i2c_address
endn
                movlw   0x00                    ;value to be written
                movwf   i2c_Data
                call    write_i2c
                call    i2c_delay
                incf    i2c_address
                return
```

```
;_____[?]_____
any_letter
        clrf    any_store
        clrf    i2c_address
        lfsr    0,0x101
anytest
        clrf    i2c_Data                ;clear buffer file
        call    read_i2c                ;call read
        movlw   0x00
        cpfseq  i2c_Data
        bra             anytest1
        bra             exitany
anytest1
        movlw   0x0A
        cpfseq  i2c_Data
        bra             anytest2
        movf    i2c_address,0
        movwf   any_store
        incf    i2c_address
        bra             anytest
anytest2
        movf    i2c_Data,0
        cpfseq  INDF0
        bra             anytest3
        movf    i2c_address,0
        movwf   any_store2
        call    display_any
        movf    any_store2,0
        movwf   i2c_address
        incf    i2c_address
        bra             anytest
anytest3
        incf    i2c_address
        bra             anytest
exitany
        return
;_____
display_any
        movf    any_store,0
        movwf   i2c_address
```

```
                incf    i2c_address
read_any
                clrf    i2c_Data
                call    read_i2c
                movlw   0x00
                cpfseq  i2c_Data
                bra             cont_any0
                bra             exit_dis_any
cont_any0
                movf    i2c_Data,0                      ;reflect
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    USART_delay
                movlw   0x0A
                cpfseq  i2c_Data
                bra             cont_any1
                bra     next_any
cont_any1
                incf    i2c_address
                bra             read_any
next_any
                movlw   0x00
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    USART_delay
                incf    i2c_address
read_any2
                clrf    i2c_Data
                call    read_i2c
                movlw   0x0A
                cpfseq  i2c_Data
                bra             cont_any2
                bra             exit_dis_any
cont_any2
                movf    i2c_Data,0                      ;reflect
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    USART_delay
                movlw   0x0A
                cpfseq  i2c_Data
                bra             cont_any3
```

```
                bra     exit_dis_any
cont_any3
                incf    i2c_address
                bra             read_any2
exit_dis_any
                movlw   0x0A
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    USART_delay
                return
;_____________________________________________
delete
                movlw   0x00                    ;value to be written
                movwf   i2c_Data
                movlw   0x00                            ;adress
                movwf   i2c_address
                call    write_i2c                       ;call write
                call    i2c_delay
                call    display_deleted
                return
;_____________display_instructions_____
display_in
                movlw   'E'
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    i2c_delay

                movlw    'N'
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    i2c_delay

                movlw   'T'
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    i2c_delay

                movlw    'E'
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    i2c_delay
```

```
movlw  'R'
btfsc     PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw  ' '
btfsc     PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw  'I'
btfsc     PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw   'N'
btfsc     PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw  'S'
btfsc     PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw   'T'
btfsc     PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw  'R'
btfsc     PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw   'U'
btfsc     PIR1,TXIF
movwf  TXREG
call       i2c_delay
```

```
                movlw  'C'
                btfsc     PIR1,TXIF
                movwf  TXREG
                call       i2c_delay

                movlw   'T'
                btfsc     PIR1,TXIF
                movwf  TXREG
                call       i2c_delay

                movlw  'I'
                btfsc     PIR1,TXIF
                movwf  TXREG
                call       i2c_delay

                movlw   'O'
                btfsc     PIR1,TXIF
                movwf  TXREG
                call       i2c_delay

                movlw   'N'
                btfsc     PIR1,TXIF
                movwf  TXREG
                call       i2c_delay

                movlw   0x0D
                btfsc     PIR1,TXIF
                movwf  TXREG
                call       USART_delay

                movlw   0x0A
                btfsc     PIR1,TXIF
                movwf  TXREG
                call       USART_delay

                return
;_____________display_new_name_____
display_new_name
                movlw  'E'
                btfsc     PIR1,TXIF
                movwf  TXREG
```

```
call    i2c_delay

movlw   'N'
btfsc   PIR1,TXIF
movwf   TXREG
call    i2c_delay

movlw   'T'
btfsc   PIR1,TXIF
movwf   TXREG
call    i2c_delay

movlw   'E'
btfsc   PIR1,TXIF
movwf   TXREG
call    i2c_delay

movlw   'R'
btfsc   PIR1,TXIF
movwf   TXREG
call    i2c_delay

movlw   ' '
btfsc   PIR1,TXIF
movwf   TXREG
call    i2c_delay

movlw   'N'
btfsc   PIR1,TXIF
movwf   TXREG
call    i2c_delay

movlw   'A'
btfsc   PIR1,TXIF
movwf   TXREG
call    i2c_delay

movlw   'M'
btfsc   PIR1,TXIF
movwf   TXREG
call    i2c_delay
```

```
movlw   'E'
btfsc     PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw  ' '
btfsc     PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw   'O'
btfsc     PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw  'F'
btfsc     PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw   ' '
btfsc     PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw  'C'
btfsc     PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw   'O'
btfsc     PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw   'N'
btfsc     PIR1,TXIF
movwf  TXREG
call       i2c_delay
```

```
                movlw   'T'
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    i2c_delay

                movlw   'A'
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    i2c_delay

                movlw   'C'
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    i2c_delay

                movlw   'T'
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    i2c_delay

                movlw   0x0D
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    USART_delay

                movlw   0x0A
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    USART_delay

                return

;_____display_new_no_____
display_new_no
                movlw   'E'
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    i2c_delay

                movlw   'N'
                btfsc   PIR1,TXIF
```

```
movwf  TXREG
call       i2c_delay

movlw  'T'
btfsc      PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw   'E'
btfsc      PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw  'R'
btfsc      PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw  ' '
btfsc      PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw  'C'
btfsc      PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw   'O'
btfsc      PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw   'N'
btfsc      PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw   'T'
btfsc      PIR1,TXIF
movwf  TXREG
```

```
call    i2c_delay

movlw  'A'
btfsc   PIR1,TXIF
movwf  TXREG
call    i2c_delay

movlw   'C'
btfsc   PIR1,TXIF
movwf  TXREG
call    i2c_delay

movlw   'T'
btfsc   PIR1,TXIF
movwf  TXREG
call    i2c_delay

movlw  ' '
btfsc   PIR1,TXIF
movwf  TXREG
call    i2c_delay

movlw   'N'
btfsc   PIR1,TXIF
movwf  TXREG
call    i2c_delay

movlw   'U'
btfsc   PIR1,TXIF
movwf  TXREG
call    i2c_delay

movlw   'M'
btfsc   PIR1,TXIF
movwf  TXREG
call    i2c_delay

movlw  'B'
btfsc   PIR1,TXIF
movwf  TXREG
call    i2c_delay
```

```
                movlw   'E'
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    i2c_delay

                movlw   'R'
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    i2c_delay

                movlw   0x0D
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    USART_delay

                movlw   0x0A
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    USART_delay

                return
;_____display_deleted_____
display_deleted
                movlw   'E'
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    i2c_delay

                movlw   'E'
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    i2c_delay

                movlw   'P'
                btfsc   PIR1,TXIF
                movwf   TXREG
                call    i2c_delay

                movlw   'R'
                btfsc   PIR1,TXIF
```

```
movwf  TXREG
call       i2c_delay

movlw  'O'
btfsc      PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw   'M'
btfsc      PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw  ' '
btfsc      PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw   'D'
btfsc      PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw  'E'
btfsc      PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw   'L'
btfsc      PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw  'E'
btfsc      PIR1,TXIF
movwf  TXREG
call       i2c_delay

movlw   'T'
btfsc      PIR1,TXIF
movwf  TXREG
```

```
                call    i2c_delay

                movlw  'E'
                btfsc   PIR1,TXIF
                movwf  TXREG
                call    i2c_delay

                movlw   'D'
                btfsc   PIR1,TXIF
                movwf  TXREG
                call    i2c_delay

                movlw   0x0D
                btfsc   PIR1,TXIF
                movwf  TXREG
                call    USART_delay

                movlw   0x0A
                btfsc   PIR1,TXIF
                movwf  TXREG
                call    USART_delay

                return

;_____________________________________
incorrect
                return

;____________usart_delay_____________________
USART_delay
                bsf             T2CON,TMR2ON
                movlw  0x00
                movwf  TMR2
back
                btfss   PIR1,TMR2IF
                bra             back
                bcf             PIR1,TMR2IF
                bcf             T2CON,TMR2ON
                return
;_____________i2c delay_____________________
i2c_delay
```

```
                movlw  d'70'
                movwf  i2c_delaycount
backddd
                call       USART_delay
                decfsz  i2c_delaycount
                bra             backddd
                return
;_____________write i2c_____________________
write_i2c
                clrf       i2c_count                        ;set initial conditions

                clrf       i2c_config
                bcf             PORTD,1                                 ;start condition

                bsf             T0CON,TMR0ON                    ;start timer
back0                                                                   ;delay two clocks
                btfss   INTCON,TMR0IF
                bra             back0
                bcf             INTCON,TMR0IF
                movlw  d'236'
                movwf  TMR0L
back1
                btfss   INTCON,TMR0IF
                bra             back1
                bcf             INTCON,TMR0IF
                movlw  d'236'
                movwf  TMR0L
                bsf       INTCON,TMR0IE
cont0
                btfss   i2c_config,1
                bra             cont0
                bcf             i2c_config,1
                bsf             PORTD,1
cont1
                btfss   i2c_config,1
                bra             cont1
                bcf             i2c_config,1
                bcf             PORTD,1
cont2
                btfss   i2c_config,1
                bra             cont2
```

```
                bcf             i2c_config,1
                bsf             PORTD,1
cont3
                btfss   i2c_config,1
                bra             cont3
                bcf             i2c_config,1
                bcf             PORTD,1
cont4
                btfss   i2c_config,1
                bra             cont4
                bcf             i2c_config,1
                bcf             PORTD,1
cont5
                btfss   i2c_config,1
                bra             cont5
                bcf             i2c_config,1
                bcf             PORTD,1
cont6
                btfss   i2c_config,1
                bra             cont6
                bcf             i2c_config,1
                bcf             PORTD,1
cont7
                btfss   i2c_config,1
                bra             cont7
                bcf             i2c_config,1
                bcf             PORTD,1
acnol0
                btfss   i2c_config,1
                bra             acnol0
                bcf             i2c_config,1
                bsf             PORTD,1
add0
                btfss   i2c_config,1
                bra             add0
                bcf             i2c_config,1
                bsf             PORTD,1
                btfss   i2c_address,0
                bcf             PORTD,1
add1
                btfss   i2c_config,1
```

```
                bra             add1
                bcf             i2c_config,1
                bsf             PORTD,1
                btfss   i2c_address,1
                bcf             PORTD,1
add2
                btfss   i2c_config,1
                bra             add2
                bcf             i2c_config,1
                bsf             PORTD,1
                btfss   i2c_address,2
                bcf             PORTD,1
add3
                btfss   i2c_config,1
                bra             add3
                bcf             i2c_config,1
                bsf             PORTD,1
                btfss   i2c_address,3
                bcf             PORTD,1
add4
                btfss   i2c_config,1
                bra             add4
                bcf             i2c_config,1
                bsf             PORTD,1
                btfss   i2c_address,4
                bcf             PORTD,1
add5
                btfss   i2c_config,1
                bra             add5
                bcf             i2c_config,1
                bsf             PORTD,1
                btfss   i2c_address,5
                bcf             PORTD,1
add6
                btfss   i2c_config,1
                bra             add6
                bcf             i2c_config,1
                bsf             PORTD,1
                btfss   i2c_address,6
                bcf             PORTD,1
add7
```

```
                btfss   i2c_config,1
                bra             add7
                bcf             i2c_config,1
                bsf             PORTD,1
                btfss   i2c_address,7
                bcf             PORTD,1
acnol1
                btfss   i2c_config,1
                bra             acnol1
                bcf             i2c_config,1
                bsf             PORTD,1
bit0
                btfss   i2c_config,1
                bra             bit0
                bcf             i2c_config,1
                bsf             PORTD,1
                btfss   i2c_Data,0
                bcf             PORTD,1
bit1
                btfss   i2c_config,1
                bra             bit1
                bcf             i2c_config,1
                bsf             PORTD,1
                btfss   i2c_Data,1
                bcf             PORTD,1
bit2
                btfss   i2c_config,1
                bra             bit2
                bcf             i2c_config,1
                bsf             PORTD,1
                btfss   i2c_Data,2
                bcf             PORTD,1
bit3
                btfss   i2c_config,1
                bra             bit3
                bcf             i2c_config,1
                bsf             PORTD,1
                btfss   i2c_Data,3
                bcf             PORTD,1
bit4
                btfss   i2c_config,1
```

```
                bra             bit4
                bcf             i2c_config,1
                bsf             PORTD,1
                btfss   i2c_Data,4
                bcf             PORTD,1
bit5
                btfss   i2c_config,1
                bra             bit5
                bcf             i2c_config,1
                bsf             PORTD,1
                btfss   i2c_Data,5
                bcf             PORTD,1
bit6
                btfss   i2c_config,1
                bra             bit6
                bcf             i2c_config,1
                bsf             PORTD,1
                btfss   i2c_Data,6
                bcf             PORTD,1
bit7
                btfss   i2c_config,1
                bra             bit7
                bcf             i2c_config,1
                bsf             PORTD,1
                btfss   i2c_Data,7
                bcf             PORTD,1
acnol2
                btfss   i2c_config,1
                bra             acnol2
                bcf             i2c_config,1
                bsf             PORTD,1
beforestop
                btfss   i2c_config,1
                bra             beforestop
                bcf             i2c_config,1
                bcf             PORTD,1
stop0
                btfss   i2c_config,0
                bra             stop0
                bcf             i2c_config,0
                bcf     INTCON,TMR0IE
```

```
                bcf             T0CON,TMR0ON
                bsf             PORTD,1

                return
;_____________________________________________
;_____________read i2c_____________________
read_i2c
                clrf    i2c_count                       ;set initial conditions

                clrf    i2c_config
                bcf             PORTD,1                                 ;start condition

                bsf             T0CON,TMR0ON                    ;start timer
back0r                                                          ;delay two clocks
                btfss   INTCON,TMR0IF
                bra             back0r
                bcf             INTCON,TMR0IF
                movlw   d'236'
                movwf   TMR0L
back1r
                btfss   INTCON,TMR0IF
                bra             back1r
                bcf             INTCON,TMR0IF
                movlw   d'236'
                movwf   TMR0L
                bsf     INTCON,TMR0IE
cont0r
                btfss   i2c_config,1
                bra             cont0r
                bcf             i2c_config,1
                bsf             PORTD,1
cont1r
                btfss   i2c_config,1
                bra             cont1r
                bcf             i2c_config,1
                bcf             PORTD,1
cont2r
                btfss   i2c_config,1
                bra             cont2r
                bcf             i2c_config,1
                bsf             PORTD,1
```

```
cont3r
        btfss   i2c_config,1
        bra     cont3r
        bcf     i2c_config,1
        bcf     PORTD,1
cont4r
        btfss   i2c_config,1
        bra     cont4r
        bcf     i2c_config,1
        bcf     PORTD,1
cont5r
        btfss   i2c_config,1
        bra     cont5r
        bcf     i2c_config,1
        bcf     PORTD,1
cont6r
        btfss   i2c_config,1
        bra     cont6r
        bcf     i2c_config,1
        bcf     PORTD,1
cont7r
        btfss   i2c_config,1
        bra     cont7r
        bcf     i2c_config,1
        bcf     PORTD,1
acnol0r
        btfss   i2c_config,1
        bra     acnol0r
        bcf     i2c_config,1
        bsf     PORTD,1
add0r
        btfss   i2c_config,1
        bra     add0r
        bcf     i2c_config,1
        bsf     PORTD,1
        btfss   i2c_address,0
        bcf     PORTD,1
add1r
        btfss   i2c_config,1
        bra     add1r
        bcf     i2c_config,1
```

```
                bsf     PORTD,1
                btfss   i2c_address,1
                bcf     PORTD,1
add2r
                btfss   i2c_config,1
                bra     add2r
                bcf     i2c_config,1
                bsf     PORTD,1
                btfss   i2c_address,2
                bcf     PORTD,1
add3r
                btfss   i2c_config,1
                bra     add3r
                bcf     i2c_config,1
                bsf     PORTD,1
                btfss   i2c_address,3
                bcf     PORTD,1
add4r
                btfss   i2c_config,1
                bra     add4r
                bcf     i2c_config,1
                bsf     PORTD,1
                btfss   i2c_address,4
                bcf     PORTD,1
add5r
                btfss   i2c_config,1
                bra     add5r
                bcf     i2c_config,1
                bsf     PORTD,1
                btfss   i2c_address,5
                bcf     PORTD,1
add6r
                btfss   i2c_config,1
                bra     add6r
                bcf     i2c_config,1
                bsf     PORTD,1
                btfss   i2c_address,6
                bcf     PORTD,1
add7r
                btfss   i2c_config,1
                bra     add7r
```

```
                bcf     i2c_config,1
                bsf     PORTD,1
                btfss   i2c_address,7
                bcf     PORTD,1
acnol1r
                btfss   i2c_config,1
                bra     acnol1r
                bcf     i2c_config,1
                bsf     PORTD,1
beforestart
                btfss   i2c_config,1
                bra     beforestart
                bcf     i2c_config,1
                bsf     PORTD,1
start0
                btfss   i2c_config,0
                bra     start0
                bcf     i2c_config,0
                bcf     PORTD,1
cont0rr
                btfss   i2c_config,1
                bra     cont0rr
                bcf     i2c_config,1
                bsf     PORTD,1
cont1rr
                btfss   i2c_config,1
                bra     cont1rr
                bcf     i2c_config,1
                bcf     PORTD,1
cont2rr
                btfss   i2c_config,1
                bra     cont2rr
                bcf     i2c_config,1
                bsf     PORTD,1
cont3rr
                btfss   i2c_config,1
                bra     cont3rr
                bcf     i2c_config,1
                bcf     PORTD,1
cont4rr
                btfss   i2c_config,1
```

```
                bra             cont4rr
                bcf             i2c_config,1
                bcf             PORTD,1
cont5rr
                btfss   i2c_config,1
                bra             cont5rr
                bcf             i2c_config,1
                bcf             PORTD,1
cont6rr
                btfss   i2c_config,1
                bra             cont6rr
                bcf             i2c_config,1
                bcf             PORTD,1
cont7rr
                btfss   i2c_config,1
                bra             cont7rr
                bcf             i2c_config,1
                bsf             PORTD,1
acnol2r
                btfss   i2c_config,1
                bra             acnol2r
                bsf             i2c_config,1
                bsf             PORTD,1
                bsf             TRISD,1
readacno
                btfss   i2c_config,0
                bra             readacno
                bcf             i2c_config,0
bit0r
                btfss   i2c_config,0
                bra             bit0r
                bcf             i2c_config,0
                bsf             i2c_Data,0
                btfss   PORTD,1
                bcf             i2c_Data,0
bit1r
                btfss   i2c_config,0
                bra             bit1r
                bcf             i2c_config,0
                bsf             i2c_Data,1
                btfss   PORTD,1
```

```
                bcf             i2c_Data,1
bit2r
                btfss   i2c_config,0
                bra             bit2r
                bcf             i2c_config,0
                bsf             i2c_Data,2
                btfss   PORTD,1
                bcf             i2c_Data,2
bit3r
                btfss   i2c_config,0
                bra             bit3r
                bcf             i2c_config,0
                bsf             i2c_Data,3
                btfss   PORTD,1
                bcf             i2c_Data,3
bit4r
                btfss   i2c_config,0
                bra             bit4r
                bcf             i2c_config,0
                bsf             i2c_Data,4
                btfss   PORTD,1
                bcf             i2c_Data,4
bit5r
                btfss   i2c_config,0
                bra             bit5r
                bcf             i2c_config,0
                bsf             i2c_Data,5
                btfss   PORTD,1
                bcf             i2c_Data,5
bit6r
                btfss   i2c_config,0
                bra             bit6r
                bcf             i2c_config,0
                bsf             i2c_Data,6
                btfss   PORTD,1
                bcf             i2c_Data,6
bit7r
                btfss   i2c_config,0
                bra             bit7r
                bcf             i2c_config,0
                bsf             i2c_Data,7
```

```
                btfss   PORTD,1
                bcf             i2c_Data,7
acnol4r
                clrf    TRISD
                btfss   i2c_config,1
                bra             acnol4r
                bsf             i2c_config,1
                bsf             PORTD,1
beforestopr
                btfss   i2c_config,1
                bra             beforestopr
                bcf             i2c_config,1
                bcf             PORTD,1
stop0r
                btfss   i2c_config,0
                bra             stop0r
                bcf             i2c_config,0
                bsf             PORTD,1
                bcf     INTCON,TMR0IE
                bcf             T0CON,TMR0ON
                return
end;
```

10. References

- EMK 310 Study guide, Prof T Hanekom, 30 January 2012
- PIC18F13K50 data sheet
- Microchip application note AN956
- EEPROM datasheet
- http://www.ee.up.ac.za/main/en/undergrad/subjects/emk310/index , Prof T Hanekom, 29 February 2012
- PICKIT45K20 data sheet