

Implementació de la Funció de DataLogger amb la utilització de la EEPROM interna

Jaume Bestard Martí, Damià Roca Mari

*Segon curs d'Enginyeria Tècnica de Telecomunicacions, Especialitat en Telemàtica
Pràctica de l'assignatura de Microordinadors*

jaume.bestard.marti@gmail.com

damianroca@gmail.com

Resum— Aquest article explica la implementació d'un sistema DataLogger. Aquest sistema realitzarà la captura de mostres analògiques amb el sensor de llum incorporat a l'entrenador amb una resolució de 10 bits. Es realitzarà la mitjana cada vint mostres seguides que es desarà a la memòria EEPROM interna a mode de memòria cíclica de 4 posicions. Segons els valors dels interruptors RA4 i RA5 es podrà recollir mostres cada 50ms o cada 500ms i també esborrar les posicions de la memòria EEPROM o bé visualitzar les quatre posicions de la memòria, que es corresponen amb les darreres mitges realitzades. Tota aquesta informació es mostrarà per la pantalla LCD.

I. INTRODUCCIÓ

Per la realització d'aquesta pràctica es necessita una pantalla LCD(per anar mostrant els diferents missatges), els interruptors RA4 i RA5(per poder detectar les quatre diferents opcions de treball), un sensor de llum o potenciòmetre (per anar agafant les diferents mostres de lluminositat amb deu bits, i per això necessitam un byte alt i un byte baix per a una dada) i el convertidor analògic/digital(per convertir les mostres agafades amb el sensor). Amb tots elements anteriors es pot realitzar de manera correcta la funció de DataLogger que es vol implementar.

Per això, s'empra el PIC16F876 que ofereix tots aquests serveis. Aquest microcontrolador és un entrenador que utilitza el programa RealPic per ser programat amb el codi que es vol introduir. Es connecta a l'ordinador a través del port RS232.

A. Objectius

Per realitzar aquest treball s'ha programat el PIC16F876 perquè comenci comprovant els valors dels dos interruptors per saber la funció que s'ha de realitzar. Un cop completat, ja es pot passar a la subrutina corresponent.

En el cas de les captures (ja sigui a 50ms o bé a 500ms) es van agafant mostres fins arribar a les 20. Mentre, es va visualitzant un missatge per veure a quina velocitat s'agafen les mostres, juntament amb el valor digital amb hexadecimal i el número de mostra al que correspon. Quan es tenen totes les mostres, s'indica amb un missatge i a continuació es mostra la mitja. Després es guarda a la EEPROM(esborrant la mitja més antiga de les quatre que hi pot haver).

Si es vol esborrar la EEPROM es mostra un missatge i es posen a zero totes les posicions. En el cas de voler mostrar els quatre valors que hi ha a la EEPROM, es visualitzaran als quatre cantons de la pantalla LCD.[1]

Un cop realitzada la subrutina corresponent es torna a l'inici, comprovant de nou els valors dels interruptors per poder continuar amb la funció de DataLogger.

II. IMPLEMENTACIÓ HARDWARE

Les connexions que s'han de fer són: alimentar l'entrenador, a una potència de 12w per poder tenir un correcte funcionament a nivell elèctric de la placa i també s'ha de connectar a l'ordinador mitjançant el port RS232 per poder programar l'entrenador amb el codi que s'ha realitzat prèviament i que ja no té errors de compilació. [2]

A més, tal com com s'indica a la Figura 1, s'han de posar els jumpers de la placa a les posicions correctes per poder emprar el LCD i el sensor o el potenciòmetre(amb els valors analògics),.

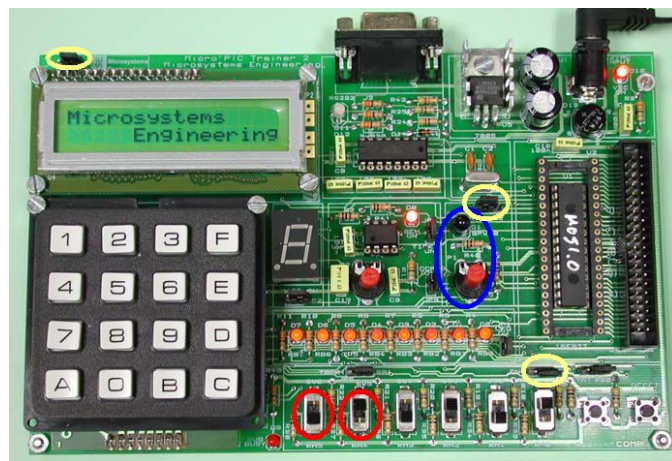


Fig.1 L'entrenador

En vermell s'han marcat els dos interruptors que s'han d'emprar (RA4 que es el de la dreta i RA5 que és el de l'esquerra)

En groc s'indiquen els jumpers que s'han de posicionar per poder emprar el sensor, per poder treballar amb valors analògics, emprar el convertidor A/D i per poder mostrar els diferents missatges i nombres per el LCD

En blau hi ha el sensor, o en el seu defecte el potenciòmetre, que s'ha d'anar girant perquè varii el seu valor. Segons la posició del jumper que tenen damunt es pot

emprar el sensor (si el jumper està cap a l'esquerra) o el potenciómetre (si el jumper està cap a la dreta).

També es pot observar el port RS232 per on es connecta a l'ordinador i el cable de l'alimentació. [2]

III. IMPLEMENTACIÓ SOFTWARE

Inicialment s'ha de sincronitzar la placa amb l'ordinador mitjançant el programa RealPic per poder programar l'entrenador correctament.

A la Figura 2 es mostra el diagrama de flux del codi:

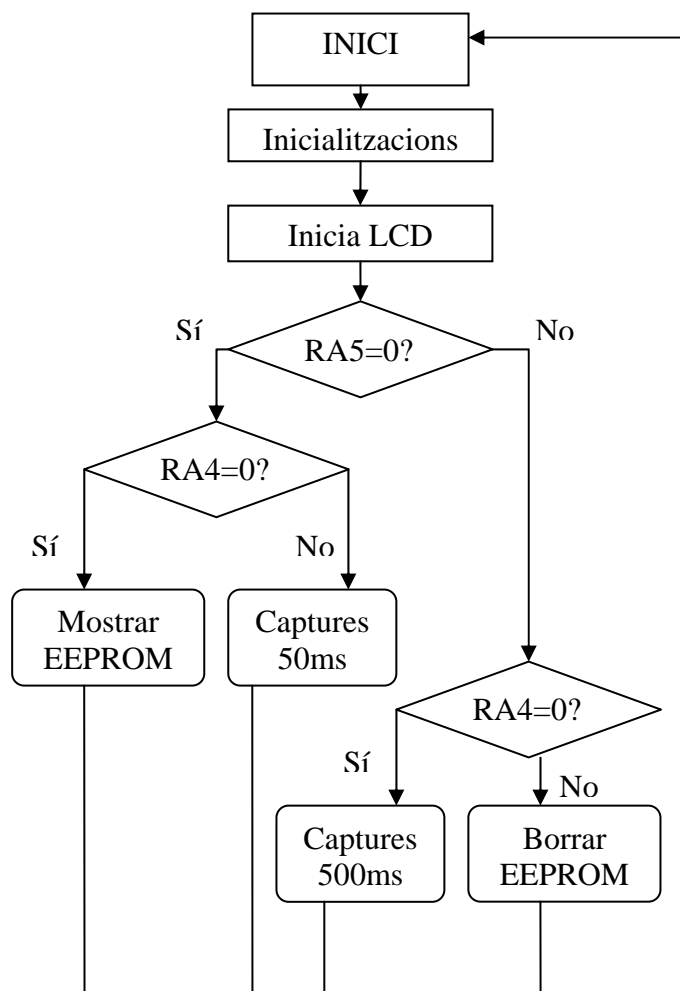


Fig. 2 Diagrama de flux del codi

Primer es realitzen les inicialitzacions per configurar correctament els ports i declarar i inicialitzar totes les variables que posteriorment utilitzarem.

Tot seguit, es comprova el valor de RA5 i després el de RA4. Així es pot saber quina és la funció que es vol realitzar. Segons aquests dos valors es crida a una subrutina o una altra, que és la que s'encarrega de realitzar el codi corresponent a cada funció. A continuació es mostra un tros del codi:

```

    btfsc ValorPA,5
    goto RA5_es1
    btfss ValorPA,5
    goto RA5_es0
  
```

```

RA5_es1:    btfsc ValorPA,4
            call borrar_EE
            btfss ValorPA,4
            call dades_500ms
            goto Inicio

RA5_es0:    btfsc ValorPA,4
            call dades_50ms
            btfss ValorPA,4
            call mostra_EE
            goto Inicio
  
```

En aquest fragment es realitza el control dels botons i segons els seus valors es crida a les subrutines.

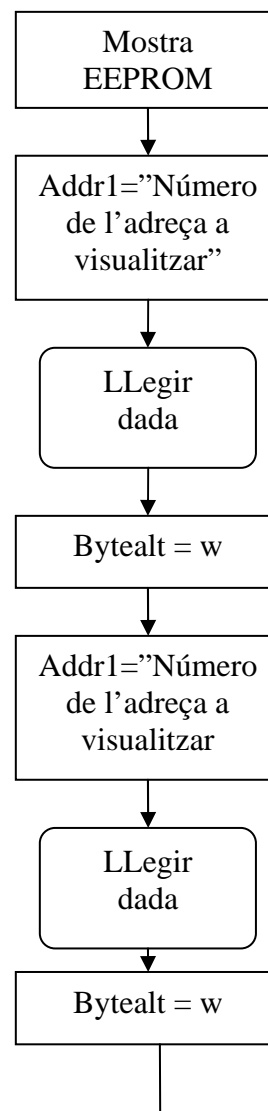


Fig. 3 Diagrama de flux de la subrutina "Mostrar EEPROM"

Aquest diagrama (Fig. 3) no es correspon a la subrutina completa perquè el que es fa es repetir això quatre vegades. A continuació es mostra un tros del codi corresponent:

```

movlw 0x30
movwf addr1
call Llegir_dada
movf valor_llegit,w
movwf bytealt
movlw 0x31
movwf addr1
call Llegir_dada
movf valor_llegit,w
movwf bytebaix
movlw 0x80
movwf poshex_2
movlw 0x81
movwf poshex_1
movlw 0x82
movwf poshex_0
call Visualiza_hex

```

Aquest tros de codi serveix per mostrar dos bytes enregistrats a la memòria EEPROM, concretament a les posicions 0x30 i 0x31. Es mostren en format hexadecimal. L'únic que s'ha de fer és repetir-la tres vegades més canviant les posicions que es mostren i al lloc del LCD on es pinta. [1]

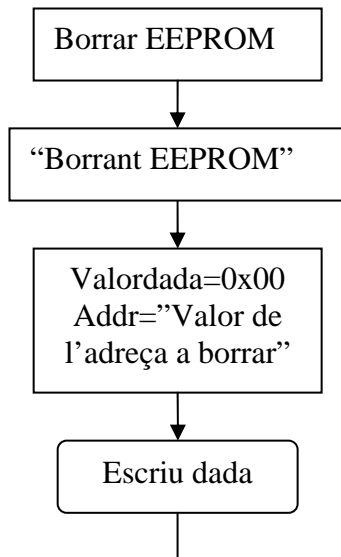


Fig. 4 Diagrama de flux de la subrutina "Esborrar EEPROM"

Aquest diagrama (Fig. 4) no es correspon a la subrutina completa perquè el que es fa es repetir això quatre vegades. A continuació es mostra un tros del codi corresponent:

```

movlw 0x00
movwf valordada
movlw 0x30
movwf addr
call Escriu_dada
movlw 0x00
movwf valordada
movlw 0x31
movwf addr
call Escriu_dada

```

Aquest tros de codi serveix per esborrar dos bytes enregistrats a la memòria EEPROM, concretament a les posicions 0x30 i 0x31. L'únic que s'ha de fer es repetir-la tres vegades més canviant les posicions que s'esborren. [1]

L'única diferència entre "Captures a 50ms" i "Captures a 500ms" és el retard que s'aplica per esperar el temps que toca.

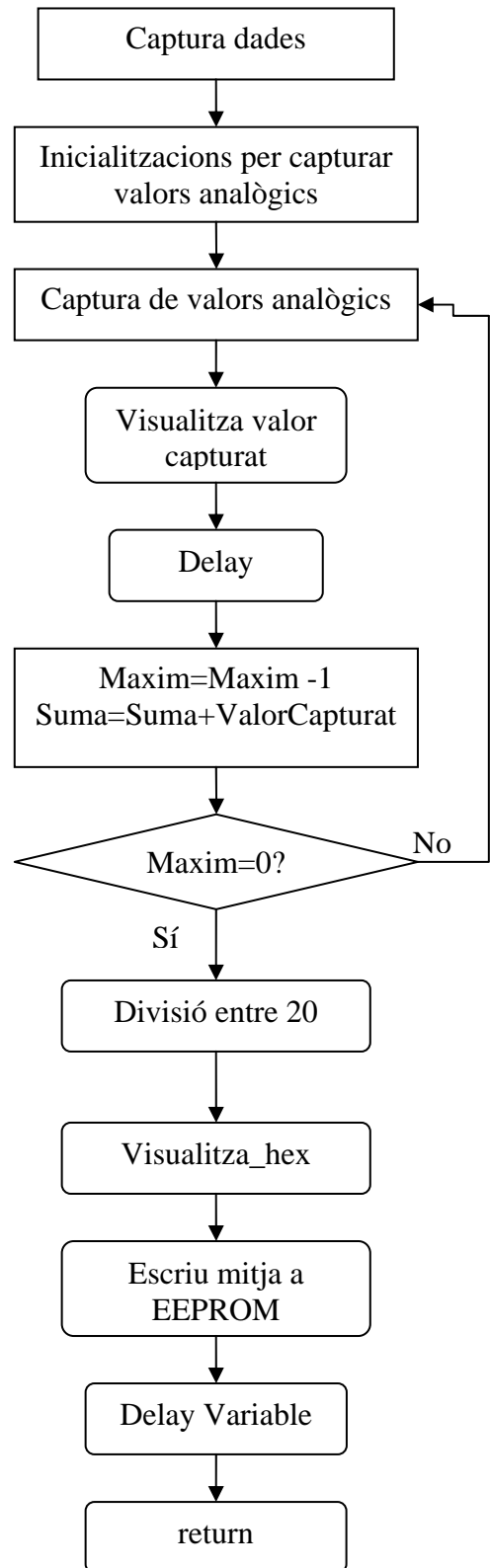


Fig.7 Diagrama de flux del subprograma "Captura dades"

El diagrama que es mostra a la Figura 7 correspon a “Captura dades” i a continuació es mostren uns trossos del codi:

	bsf	STATUS,RP0
	movlw	b'00111111'
	movwf	TRISA
	clrf	ADCON1
	bsf	ADCON1, 7
	bcf	STATUS,RP0
	bcf	PIR1,ADIF
	nop	
	bsf	ADCON0,GO
ADC_Wait:	btfss	PIR1,ADIF
	goto	ADC_Wait

Aquest tros de codi realitza la conversió d’analogic a digital dels valors que li va donant el sensor i guarda el resultat dins ADRESL i ADRESH. [1]

	bsf	STATUS,RP0
	movlw	b'10000110'
	movwf	ADCON1
	bcf	STATUS,RP0
	call	UP_LCD
	call	Visualiza
	movf	ADRESH_1,w
	addwf	suma_h,f
	movf	ADRESL_1,w
	addwf	suma_l,f
	btfsc	STATUS, C
	incf	suma_h,f
increment:	decf	maxim,f
	btfss	STATUS, Z
	goto	Loop

Amb aquest codi es va incrementant la suma de tots els valors que es van agafant, es controla el nombre de mostra a la que es troba i també es mira que no es passi de les vint mostres.

	decf	posicio
	movf	posicio,w
	btfsc	STATUS, Z
	movlw	d'4'
	movwf	posicio
Es_4:	movlw	d'4'
	subwf	posicio, w

	btfsc	STATUS,Z
	goto	escriure_pos0
Es_3:	movlw	d'3'
	subwf	posicio, w
	btfsc	STATUS,Z
	goto	escriure_pos1
Es_2:	movlw	d'2'
	subwf	posicio, w
	btfsc	STATUS,Z
	goto	escriure_pos2
Es_1:	movlw	d'1'
	subwf	posicio, w
	btfsc	STATUS,Z
	goto	escriure_pos3

Aquí es reflecteix com es realitza el control sobre la posició de la EEPROM (s’accedeix a dues posicions ja que a cada una s’hi pot guardar un byte i la mitja està formada per dos bytes) a la qual s’hi escriu la nova mitja, esborrant la mitja més vella de les que hi ha guardades.

IV. CONCLUSIONS

En aquest article s’ha desenvolupat la funció DataLogger sobre l’entrenador PIC16F876. S’ha vist com s’ha d’emprar el programa RealPic, com s’han de realitzar les connexions, com s’empra la memòria EEPROM, com agafar valors analògics (amb el sensor o bé amb el potenciómetre) i també s’ha treballat amb el conversor analògic/digital. S’han tractat dades de dos bytes per separat (byte_alt i byte_baix) ja que les memòries i els registres són d’un byte.

A més, s’ha realitzat un cas pràctic que podria servir per controlar la intensitat de llum mitjana (i amb diferents mostres) d’un lloc.

REFERÈNCIES

- [1] Tomeu Alorda, Apunts de l’assignatura de Microordinadors, de 2on de Telemàtica.
- [2] Microchip PIC16F87X Data Sheet, 28/40-Pin 8-bit CMOS FLASHMicrocontrollers

Assignatura impartida per Bartomeu Alorda i Pere Pons



Jaume Bestard Martí

Batxillerat al Col·legi Sant Josep Obrer
Estudiant de 2on d’Enginyeria Tècnica de Telecomunicacions especialitat Telemàtica



Damià Roca Marí

Batxillerat al Col·legi Sant Pere
Estudiant de 2on d’Enginyeria Tècnica de Telecomunicacions especialitat Telemàtica