



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Facultat d'Informàtica de Barcelona

Enunciat de la pràctica de laboratori

Ports d'entrada/sortida

Ports d'entrada/sortida

1. Introducció

En aquesta pràctica es prendrà contacte amb la plataforma de desenvolupament EasyPIC6 (Fig.1) que utilitzarem al llarg de tot el curs. A partir de la documentació proporcionada, identificareu components en esquemes electrònics, coneixereu l'entorn de programació, veureu com descarregar els programes a la placa de desenvolupament i fareu unes proves senzilles per agafar confiança amb el conjunt.

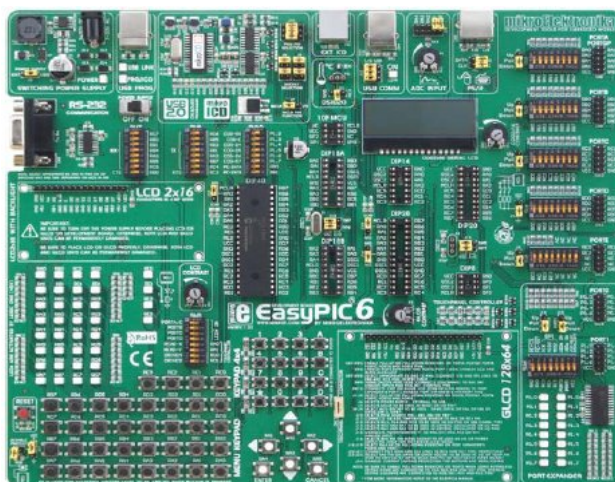


Fig1. Plataforma de desenvolupament EasyPIC6

El Integrated Development Environment (IDE) utilitzat a les pràctiques serà el Proteus. Proteus integra un editor d'esquemàtics, editor de codi, compilador, assemblador i simulador en temps real. Per aquesta pràctica i fins a acabar el quadrimestre, es farà servir el **compilador XC8** (llenguatge C). A l'hora de crear un nou projecte amb Proteus i seleccionar el firmware amb el que treballareu, també haureu d'indicar el tipus de compilador (per defecte, MPASM).

Un cop compilat el vostre codi, Proteus generarà un arxiu amb extensió “.cof” que permet simular el conjunt hardware/codi descrit a l'entorn Proteus, o un arxiu amb extensió “.hex” que és executable sobre el PIC. Per executar el programa a la placa de desenvolupament, haureu de posar-vos en mode "Release" (a la finestra "Source Code" del Proteus) i compilar el vostre codi. Es generarà un arxiu “.hex” a la vostra carpeta de treball que haureu de descarregar a la placa a través del bus USB. La descàrrega del firmware a la placa la fareu a través del programa anomenat **MIKROPROG**. Un cop carregat el firmware, el microcomputador provocarà un reset i executarà la vostra aplicació.

2. Objectius

L'objectiu d'aquesta pràctica és encendre i apagar uns leds connectats al port de sortida a partir dels senyals dels pulsadors connectats al port d'entrada.

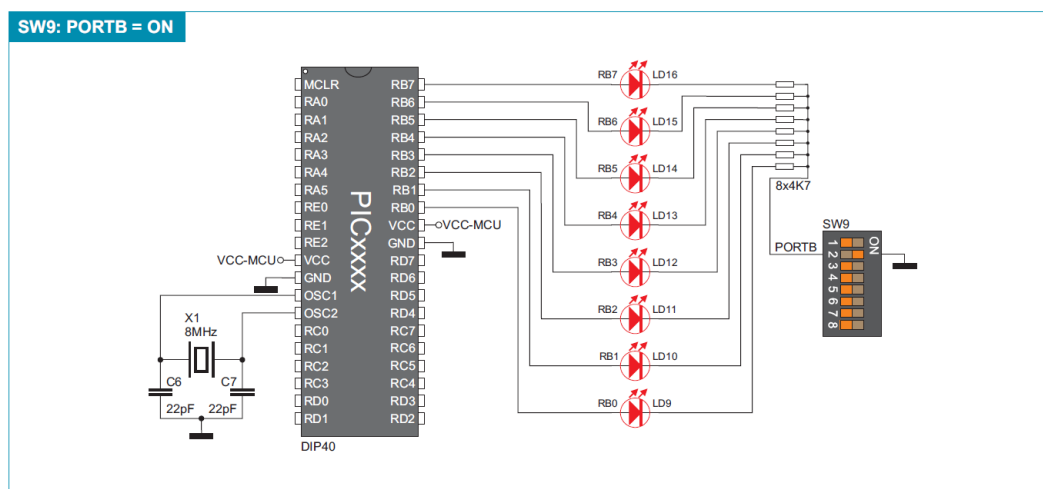
En acabar la pràctica l'alumne serà capaç de:

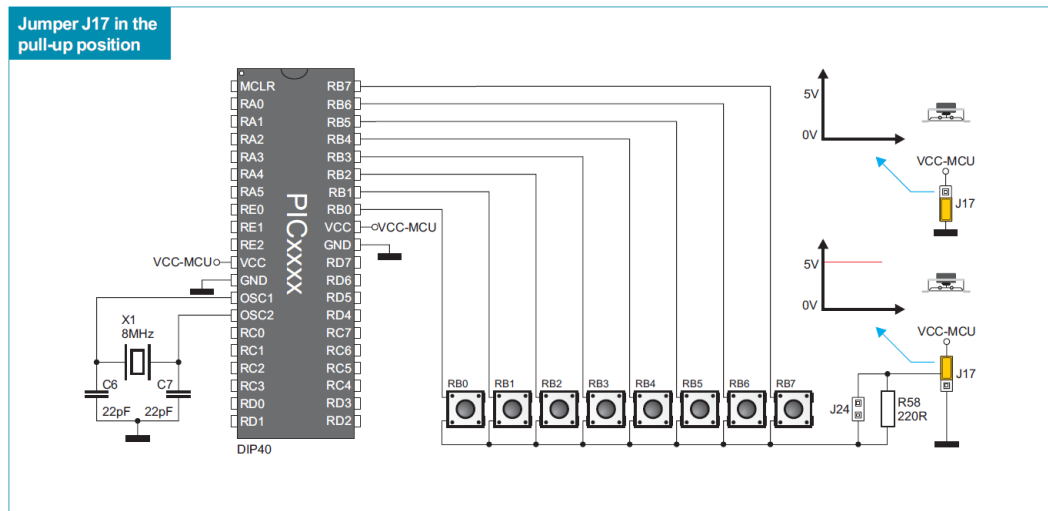
- manegar informació tècnica donada pel fabricant de la placa de desenvolupament que utilitzem a les pràctiques.
- cercar la informació necessària dins dels manuals de referència.
- identificar els elements 'pulsador' i 'led' i com es connecten al microcomputador.
- fer un primer experiment respecte a les diverses funcionalitats dels pins del microcomputador.
- crear i configurar un projecte amb la plataforma de desenvolupament.
- debugar el programa, utilitzar breakpoints i tracejar els valors de les diferents variables i registres.
- descarregar els fitxers executables ja generats a la placa de prototipus.
- entendre el funcionament del conjunt, veient la relació entre el codi i el comportament del microcomputador.
- gestionar adequadament senyals d'entrada per nivell i per flanc, i a utilitzar variables d'estat.

3. Treball previ

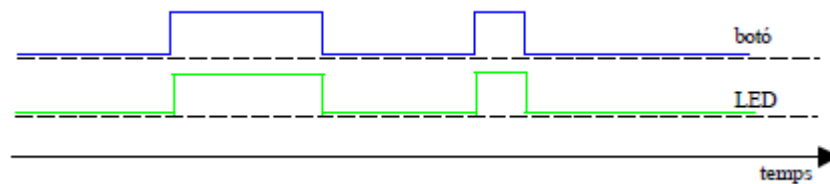
Temps estimat: 4 hores.

1. Cal entendre la configuració interna dels I/O PORTS A i B del microcomputador (pàgines 113-118 del *PIC18F4550 Data Sheet*). En particular la configuració del PORTA i PORTB com a PORT digital mitjançant el registre ADON1, així com la configuració dels ports com a IN o OUT mitjançant els registres TRIS.
2. Disseny de l'esquema elèctric sobre Proteus. Cal dissenyar el circuit basat en l'esquemàtic que trobareu a la documentació lliurada sobre la placa EASYPIC6 (i a la figura 2). Tingueu sempre present que tot component de la placa ha de tenir el seu equivalent a l'esquema de Proteus i a l'inrevés (copieu només aquells components que necessiteu per a fer la pràctica, **NO intenteu dissenyar la placa sencera**). Vigileu que l'entrada MCLR (Master Reset) estigui a 1, per evitar que hi hagi resets aleatoris si queda a l'aire.

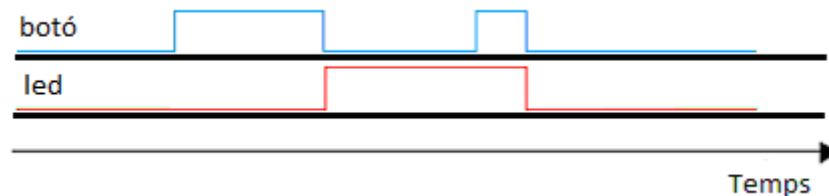




3. Fer un programa en llenguatge C en el que cada cop que es premi el pulsador associat al bit 0 del PORTB (RB0) s'encengui el LED associat al bit 0 del PORTA (RA0) i quan es deixi anar el pulsador s'apagui el LED (veure figura 3). S'espera que aquesta versió del codi es realitzi **per enquesta**: el vostre programa consultarà contínuament l'estat de l'entrada associada al pulsador. Al final de l'enunciat trobareu un esquelet de codi que us servirà com a guia per aquest exercici i el següent.



4. Fer un altre programa en que cada cop que es premi el polsador RB0, canviï l'estat del LED, és a dir: si el LED està apagat i es prem el polsador, aquest s'ha d'encendre i quedar encès fins que no es torni a prémer. Inversament, si el LED està encès i es prem el polsador, s'haurà d'apagar i deixar apagat. En aquest programa la sortida no queda determinada pel nivell d'entrada, sinó per un canvi en el nivell d'entrada, concretament quan aquesta passa d' 1 a 0 (**flanc de baixada**). S'emularà el funcionament d'un biestable tipus 'T' (veure cronograma).



5. Execució, test i debugat dels vostres programes sobre Proteus. **No us oblideu d'incloure el fitxer 'config.h' abans del vostre codi.**
En el segon programa, poseu un *software breakpoint* després de la lectura del PORTB quan aquest canvia d'estat. Observeu el valor del PORTA, TRISA, LATA, PORTB, TRISB i LATB un cop s'atura l'execució.
Poseu un *hardware breakpoint* que s'activi quan s'encengui el LED. Observeu el valor del PORTA, TRISA, LATA, PORTB, TRISB i LATB un cop s'atura l'execució. Observeu com varien aquests valors a mesura que seguim executant instruccions del programa.
6. A l'inici de classe haureu d'entregar al professor el full adjunt amb dos diagrames de flux corresponents als 2 programes implementats, i les respostes al que se us demana.

Entregueu el projecte PROTEUS (inclou diagrama esquemàtic, codi font implementat i el fitxer executable ".hex") via web, ABANS de la vostra sessió de pràctiques. Per a garantir compatibilitats de versions, lliureu sempre el vostre treball salvat en una versió compatible de PROTEUS (versió actual als laboratoris, v8.4 SP0).

3. Pràctica al laboratori

A l'inici de la sessió, l'alumne ha de mostrar els 2 programes corrent sobre Proteus. Un cop comprovat el funcionament correcte, la pràctica consistirà en els següents apartats:

1. Configuració de la placa de forma que:
 - Estiguin actius només els leds de sortida del PORTA (SW9).
 - Els pulsadors d'entrada del PORTB generin un 0 al estar lliures i un 1 a l'estar premut (J17-SW2-J2). Veure figura 5.

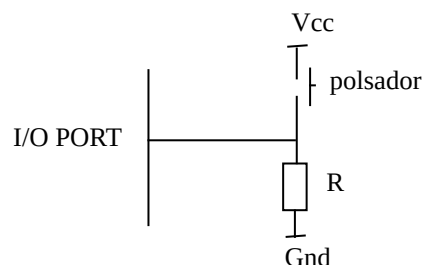
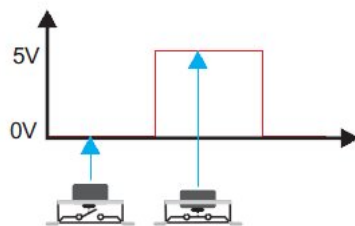


Fig 5. Configuració dels pulsadors

2. Per poder fer/comprovar la correcta configuració de la placa de prototipus, cal entendre bé el conexionat dels LEDS i dels pulsadors als ports d'entrada i

sortida del microcomputador (pàgines 17 i 18 del *EasyPIC User Manual*). En particular cal vigilar la configuració del Switch 9 (SW9) i del jumper 17 (J17).

3. També cal conèixer la configuració de *switches* i *jumpers* de la placa en relació als pulls up i pulls down de les entrades. En el cas del PORT B s'encarreguen el SW2 i el J2 (pàgines 24 i 25 del *EasyPIC User Manual*).
4. Baixar el vostre primer programa a la placa fent servir MIKROPROG. Provar que funciona. Si no és així corregir la posició dels *switches* i *jumpers*. Quan funcioni ensenyeu l'execució del codi al professor.
5. Baixar el vostre segon programa a la placa. Provar que funciona. Si no és així no cal que corregiu la posició dels *switches* i *jumpers*, que serà correcta si funciona el vostre primer programa. El problema caldrà buscar-lo en el vostre codi. Seguiu les indicacions del professor. Quan funcioni ensenyeu l'execució del codi al professor.
6. En aquest punt el professor us donarà un nou enunciat "in-situ". Implementeu les modificacions de codi i/o configuració de la placa que el professor us demani.

// A proposal for your code Just to inspire you

```
#include <xc.h>
```

```
#include "config.h"
```

```
#define PINA0 PORTAbits.RA0
```

```
#define PINB0 PORTBbits.RB0
```

```
void configPIC(){
```

```
    //config PIC
```

```
    ADCON1=...; // All PORTs are digital
```

```
    TRISA=...;
```

```
    TRISB=...;
```

```
}
```

```
void main (void) {
```

```
    configPIC();
```

```
    while(1) {
```

```
        // your code
```

```
    }
```

```
}
```

Qüestionari Pràctica ports E/S

1) Dibuixeu els diagrames de flux dels 2 programes implementats

2) Que conté el fitxer 'config.h'?

3) indica els valors dels registres després del software breakpoint

PORTA=	TRISA=	LATA=
PORTB=	TRISB=	LATB=

4) indica els valors dels registres després del hardware breakpoint

PORTA=	TRISA=	LATA=
PORTB=	TRISB=	LATB=