

```

#include <xc.h>

#include <stdio.h>

#include <string.h>

#include "config.h"

#include "GLCD.h"

#define _XTAL_FREQ 8000000


char buff[11]; //màxim tindrem 11 caràcters

const char * s1 = "L4B GLCD\n";

const char * s2 = "-----\n";

const char * yan = "Yanick Palacios\n";

const char * arn = "Arnau Cullèll\n";

const char * buit = "          ";


int hores = 1; //int perquè sinó fa un comportament estrany, i dona igual si són 8 bits o més
perquè no aniran a cap port

int minuts = 0;

int segons = 0;

int decimes = 0;


char prevRC0 = 0;

char estat = 0;

char prev_estat = -1;


char* red = "Ready";

char* run = "Running...";

char* sto = "Stopped!";


void tic() {

    decimes -= 1;

    if (decimes == -1) {

```

```
    decimes = 9;
    --segons;
}
```

```
if (segons == -1) {
    segons = 59;
    --minuts;
}
```

```
if (minuts == -1) {
    minuts = 59;
    --hores;
}
}
```

```
void interrupt service_routine_HighP (){
```

```
    if (TMR0IE && TMR0IF) {
        INTCONbits.TMR0IF = 0; //baixem el flag
        TMR0H = 0x3C; //reiniciem registres per tornar a fer 0.1 seg
        TMR0L = 0xAF;
        tic();
    }
}
```

```
void configPIC() {
    ANSELB=0x00;
    ANSELD=0x00;
    ANSELC=0x00; //digital RC0

    TRISD=0x00;
```

```

TRISB=0x00;
TRISC=0x01; //RC0 input

PORTD=0x00;
PORTB=0x00;
PORTC=0x00;

//configuració interrupcions
RCONbits.IPEN = 1;
INTCONbits.GIEL = 1;
INTCONbits.GIEH = 1; //activem interrupcions alta prioritat
INTCONbits.TMR0IF = 0; //flag a 0
INTCON2bits.TMR0IP = 1; //alta prioritat

//configurar TMR0
T08BIT = 0; //16 bits pel TMR0
TOCS = 0; //Fosc/4
PSA = 0; //amb preescalar
T0CONbits.TOPS = 0b001; // PRE = 4
TMR0H = 0x3C; //posem TMR0 amb valor adequat
TMR0L = 0xAF;
}

void writeTxt(byte page, byte y, char * s) {
    int i=0;
    while (*s!='\n' && *s!='\0')
    {
        putchGLCD(page, y+i, *(s++));
        i++;
    };
}

```

```

void mostrar_noms(char* s1, char* s2, char* yan, char* arn) {
    writeTxt(2, 9, s1); //y, x, missatge
    writeTxt(3, 9, s2); //y, x, missatge
    writeTxt(4, 5, yan); //y, x, missatge
    writeTxt(5, 6, arn); //y, x, missatge

    __delay_ms(1000);
    clearGLCD(0,7,0,127); //Esborrem pantalla després de un 1seg
}

```

```

void config_flancs() {
    prevRC0 = PORTCbits.RC0; //actualitzem prevRC0
}

```

```

void updateGLCD(){
    if (estat == 0) {
        writeTxt(0, 0, buit); //esborrem el cronometre anterior
        sprintf(buff, "01:00:00.0"); //resetejem
        writeTxt(0, 0, buff);
        writeTxt(0, 13, buit); //esborrem estat anterior
        writeTxt(0, 20, red); //posem el ready
    }
}

```

```

else if (estat == 1) {
    TMR0ON = 1; // toca encendre el cronometre
    TMR0IE = 1;
    sprintf(buff, "0%d:%d:%d.%d", hores, minuts, segons, decimes); //actualitzem cronòmetre
    writeTxt(0, 0, buff);
    if (estat != prev_estat) { //si just canvia d'estat es posa running, sino no cal
        writeTxt(0, 13, buit);
    }
}

```

```

        writeTxt(0, 15, run);
    }
}

else {
    TMR0ON = 0; // toca apagar el cronometre
    writeTxt(0, 16, buit);
    writeTxt(0, 17, sto);
    hores = 1; //resetejem tot
    minuts = 0;
    segons = 0;
    decimes = 0;
}
}

void main(void) {
    configPIC();
    GLCDinit();
    clearGLCD(0, 7, 0, 127);
    setStartLine(0);
    mostrar_noms(s1, s2, yan, arn); //mostrem els noms durant 1seg
    updateGLCD();

    while (1) {
        if (prevRC0 == 1 && PORTCbits.RC0 == 0) {
            ++estat;
            if (estat > 2) {
                estat = 0;
                prev_estat = 2;
            }
            else prev_estat = estat-1;
        }
    }
}

```

```
}
```

```
else prev_estat = estat;
```

```
if (prev_estat != estat || estat == 1) updateGLCD();
```

```
config_flancs();
```

```
__delay_ms(75);
```

```
}
```

```
}
```