

CODI L4a PREVI

```
#include <xc.h>

#include <stdio.h>

#include <string.h>

#include "config.h"

#include "GLCD.h"

#define _XTAL_FREQ 8000000

const char * s1 = "L4A GLCD\n";

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

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

const char * arn = "Arnau Cullell\n";

char buff[11]; //com a màxim hi haurà 11 caràcters al (x, y)

const char* buit = "    ";

void configPIC() {

    ANSELA=0x03; //analogic AN0 i AN1

    ANSELB=0x00;

    ANSELC=0x00;

    ANSELD=0x00;


    TRISD=0x00;

    TRISB=0x00;

    TRISA=0x03; //inputs són els dos


    PORTD=0x00;

    PORTB=0x00;

    PORTA=0x00;

    //configuració del AD

    ADCON0 = 0x01; //seleccionem el canal AN0 (CHS3:CHS0 = 0000), AD encés (ADON = 1)

    ADCON1 = 0x00; //referencia VDD y VSS

    ADCON2 = 0xA9; //ADFM=1 (justificació a la dreta), ACQT = 101 (12 Tad), ADCS = 001 (Fosc/8)

}

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
}

int funcio_AD(char canal) { //no és un uint8_t perquè sinó llavors el màxim valor possible és 255, i volem
valors fins a 1023

    if (canal == 0) ADCON0 = 0x01; //canal 0 seleccionat

    else ADCON0 = 0x05; //canal 1 seleccionat

    ADCON0 |= 0x02; //fem que comenci la conversio

    while(ADCON0bits.GO_nDONE); //esperem que acabi la conversio

    return (ADRESH << 8) + ADRESL;
}

void main(void) {
    configPIC();
    GLCDinit();
    clearGLCD(0, 7, 0, 127);
```

```
setStartLine(0);

mostrar_noms(s1, s2, yan, arn); //mostrem els noms durant 1seg


int num0 = 0; //no són uint8_t perquè poden arribar valor de fins a 1023, i amb uint8_t només podriem
guardar fins a 255

int num1 = 0;

int x = 0, y = 0;

int prev_x = 63, prev_y = 63;


while (1) {

    num0 = funcio_AD(0);

    num1 = funcio_AD(1);


    x = (num0 / 8); //escalat horitzontal

    y = 63 - (num1 / 16); //escalat vertical i invertit


    if (x >= 127) x = 127; //comprovació per que no surti dels límits

    if (x <= 0) x = 0;

    if (y >= 63) y = 63;

    if (y <= 0) y = 0;


    if (y != prev_y || x != prev_x) { //si x o y canvien actualitzem

        writeTxt(0, 10, buit); //buidem la pantalla

        ClearDot(prev_y, prev_x); //treiem el punt antic


        sprintf(buff, "(%d,%d)", num0, num1);

        writeTxt(0, 10, buff); //y, x, missatge

        SetDot(y, x); //posem el punt nou

    }

    prev_x = x; //actualitzem la x

    prev_y = y; //actualitzem la y

}
```