

UNIVERZITET U SARAJEVU



ELEKTROTEHNIČKI FAKULTET
ODSJEK ZA AUTOMATIKU I ELEKTRONIKU

Laboratorijska vježba 3

Analogni ulazi i displeji

Bazdar Faris

Predmet: Ugradbeni sistemi

Akadska godina: 2019/2020

Sarajevo, mart 2020.

1 Zadatak 1 alternativa

```
#include "mbed.h"
#include "C12832.h"

C12832 lcd(SPI_MOSI, SPI_SCK, SPI_MISO, p8, p11);
AnalogIn ainOTPOR(p15);
AnalogIn ainNAPON(p16);
DigitalIn tasterNAPON(p8);
DigitalIn tasterOTPOR(p9);

int main() {
    float otpor = 0;
    float napon = 0;
    int kontrola = 0;
    float struja = 0.00033;
    while (1) {
        otpor = ainOTPOR.read() * 3.3 / struja;
        napon = ainNAPON.read() * 3.3;
        lcd.locate(0, 3);
        if (kontrola == 0) {
            lcd.cls();
            lcd.printf("Napon na ulazu je %f V", napon);
        } else if (kontrola == 1) {
            lcd.cls();
            lcd.printf("Otpor na ulazu je %f Ohma", otpor);
        }
        if (tasterNAPON) kontrola = 0;
        else if (tasterOTPOR) kontrola = 1;
        wait_ms(0.1);
    }
}
```

Opis koda: Inicijaliziran je LCD, 2 potencimetra i 2 tastera. Iz razloga jer je struja na analognom ulazu približna nuli (zbog vrlo velikog otpora analognog ulaza). Uz usvojenih $10k\Omega$, struja je izračunata kao $0.00033A$. Napon i otpor se mogu izračunati čitanjem samog analognog ulaza koji pokazuje od (0 - 1) što je bilo potrebno skalirati sa naponom napajanja $3.3V$. Vršeći kontrolu koji je taster pritisnut uz pomoćnu varijablu zadržava se prikaz odredjenog tipa (napon ili otpor). Da bi se ažurirao ekran stalno se vrši `lcd.cls()`, što jednostavno ukloni sav sadržaj sa ekrana. Potrebno je detektovati da li je taster pritisnut pa je zbog tog korišteno `wait ms(0.1)` pri većim wait-ovima bio je slučaj da je odziv na pritisak sporiji jer u biti ne bude detektovan.

2 Zadatak 2 alternativa

```
#include "mbed.h"
#include "stm32f413h_discovery_ts.h"
#include "stm32f413h_discovery_lcd.h"
#include "string"
#include "sstream"

AnalogIn LM35(p15);

int main() {
    int NAPON;
    int tempC;
    int vrijeme = 0;
    int xnovo = 0;
    int ynovo = 0;
    int xstaro = 10;
    int ystaro = 220;
    while(1) {
        BSP_LCD_SetFont(&Font12);
        BSP_LCD_DisplayStringAt(40, 5, (uint8_t *) "Temperatura(C)", LEFT_MODE);
        BSP_LCD_DisplayStringAt(170, 225, (uint8_t *) "Sekunde(s)", LEFT_MODE);
        BSP_LCD_DisplayChar(10, 10, 51);
        BSP_LCD_DisplayChar(18, 10, 51);
        BSP_LCD_DisplayChar(26, 10, 48);
        BSP_LCD_DisplayChar(10, 120, 49);
        BSP_LCD_DisplayChar(18, 120, 53);
        BSP_LCD_DisplayChar(26, 120, 48);
        std::string napSTR = "Napon: ";
        std::string tempSTR = "Temp: ";
        std::string vrijemeSTR = "Vrijeme: ";
        BSP_LCD_Init();
        tempC = LM35.read() * 100 * 3.3;
        NAPON = LM35.read() * 1000 * 3.3;
        //ispisivanje napona
        napSTR += std::to_string(NAPON);
        napSTR += " mV";
        unsigned char *val = new unsigned char[napSTR.length() + 1];
        strcpy((char *)val, napSTR.c_str());
        BSP_LCD_SetFont(&Font12);
        BSP_LCD_DisplayStringAt(70, 5, val, CENTER_MODE);
        //ispisivanje temperature
        tempSTR += std::to_string(tempC);
        tempSTR += " C";
        unsigned char *val1 = new unsigned char[tempSTR.length() + 1];
```

```
strcpy((char *)val1,tempSTR.c_str());
BSP_LCD_DisplayStringAt(75, 15, val1, CENTER_MODE);
//timer
vrijemeSTR+=std::to_string(vrijeme);
vrijemeSTR+=" s ";
unsigned char *val2=new unsigned char[vrijemeSTR.length
    ()+1];
strcpy((char *)val2,vrijemeSTR.c_str());
BSP_LCD_DisplayStringAt(75, 25, val2, CENTER_MODE);
//grafik
BSP_LCD_DrawHLine (0, 220, 220);
BSP_LCD_DrawVLine (10, 10, 220);
xnovo = 10 + vrijeme * 5;
ynovo = 220 - tempC * 2/(3.3);
BSP_LCD_DrawLine (xstaro, ystaro, xnovo, ynovo);
if(ynovo!=ystaro) BSP_LCD_FillCircle(xstaro, ystaro, 2);
xstaro = xnovo;
ystaro = ynovo;
//wait-ovi
wait_ms(1000);
vrijeme++;
    }
}
```

Opis koda: Korištene su komande Setfont da bi se postavio font na displayu, za tekstualni prikaz korišteno je DisplayStringAt, ukoliko je bilo potrebno prikazati samo jedan karakter onda je korištena komanda DisplayChar. Sve ove komande su zahtjevale poznavanje širine i dužine display-a (u pixelima). Uzeto je da je display veličine 240x240p. Prikaz samih vrijednosti koje je bilo lagano očitati koristeći se logikom prethodnog koda se vršio tako što bi se često ažurirao sami string koji bi se prikazivao na ekranu. Bilo je potrebno taj isti string pretvoriti u pokazivač na niz karaktera (tzv. "C-ovski string") da bi se moglo prikazati koristeći komandu DisplayStringAt. S obzirom da timer nije funkcionalan unutar simulatora korišten je brojač koji se povećavao kad god bi prošla jedna sekunda (sigurno prođe malo više od 1 sekunde zbog vremena izvršenja nekih od komandi, ali uzeto je da je vrijeme izvršenja tih istih zanemarljivo). Da bi se iscrtao grafik korištene su četiri komande, dvije za apscisu i ordinatu (DrawHLine, DrawVLine), jednu za crtanje prave između dvije tačke DrawLine i jednu za crtanje kruga pri promijeni y-koordinate, da bi bilo što sličnije postavci FillCircle. Same tačke odabira su bile ishodište našeg koordinatnog sistema te 2 tačke koje bi figurisalo vrijeme i vrijednost temperature xnovo i ynovo, nakon prve iteracije xstaro i ystaro više nisu koordinate ishodišta već se ažuriraju.