

# Podstawy techniki mikroprocesorowej

## Sprawozdanie z laboratorium

Data	Tytuł zajęć	Uczestnicy
07.04.2017 14:15	Programowanie obsługi LCD	Iwo Bujkiewicz (226203)

## Listingi programów

### Programowanie LCD

```
LJMP START

; Zdefiniowanie adresów szyn danych LCD pod odpowiednimi nazwami
LCDstatus      EQU    0xFF2E
LCDcontrol     EQU    0xFF2C
LCDdataWR      EQU    0xFF2D
LCDdataRD      EQU    0xFF2F

; Zdefiniowanie wartości kodów sterujących LCD
#define HOME    0x80    // place caret in first line
#define INIT    0x38    // LCD 8-bit init
#define HOME2   0xC0    // place caret in second line
#define LCDON   0x0E    // init caret, switch cursor off, switch blinking off
#define CLEAR   0x01    // clear LCD lines

ORG 0x0100

LCDcntrlWR MACRO x      ; Makro służące do wysłania kodu sterującego x do LCD
    LOCAL loop          ; Informacja dla assemblera, że etykieta 'loop' jest symbolem
    lokalnym dla makra
loop:
    MOV DPTR, #LCDstatus ; Ustawienie wskaźnika danych na adres szyny statusu
    MOVX A, @DPTR        ; Odczytanie kodu statusu wyświetlacza
    JB ACC.7, loop       ; Sprawdzenie, czy LCD jest gotowy, jeśli nie -
    kontynuacja pętli

    MOV DPTR, #LCDcontrol ; Ustawienie wskaźnika danych na adres szyny
    sterującej wyświetlacza
    MOV A, x              ; Wypełnienie akumulatora wartością parametru x
    MOVX @DPTR, A        ; Wysłanie kodu sterującego x na szynę sterującą LCD
    ENDM                 ; Koniec definicji makra

LCDcharWR MACRO
    LOCAL loop1, loop2

    PUSH ACC
loop1: MOV DPTR, #LCDstatus
    MOVX A, @DPTR
    JB ACC.7, loop1      ; check if LCD busy

loop2: MOV DPTR, #LCDdataWR ; write data to LCD
    POP ACC
    MOVX @DPTR, A
    ENDM

init_LCD MACRO
    LCDcntrlWR #INIT
    LCDcntrlWR #CLEAR
    LCDcntrlWR #LCDON
```

```

        ENDM

charStor MACRO x
        MOV A, x
        MOVX @DPTR, A
        INC DPTR
        INC R1
        ENDM

charSeqWr MACRO
        LOCAL loop
loop:    MOVX A, @DPTR
        LCDcharWR
        INC DPTR
        DJNZ R1, loop
        ENDM

START:  init_LCD

        LCDcntrlWR #HOME
        MOV R1, #0x00
        MOV DPTR, #0x8080
        charStor #'$'
        charStor #' '
        charStor #'l'
        charStor #'s'

        MOV DPTR, #0x8080
        charSeqWr

        LCDcntrlWR #HOME2
        MOV R1, #0x00
        MOV DPTR, #0x8080
        charStor #'$'
        charStor #' '
        charStor #'s'
        charStor #'u'
        charStor #'d'
        charStor #'o'
        charStor #' '
        charStor #'r'
        charStor #'m'
        charStor #' '
        charStor #'-'
        charStor #'r'
        charStor #'f'
        charStor #' '
        charStor #'/'

        MOV DPTR, #0x8080
        charSeqWr

        ;LCDcntrlWR #HOME2
        ;MOV DPTR, #0x8100
        ;charStor #'*'
        ;charStor #' '

        NOP
        NOP
        NOP
        JMP $
END START

```