

Podstawy Techniki Mikroprocesorowej - Sprawozdanie

Czwartek 11.15 TP Grupa B

Piotr Łach 256761

Jakub Szpak 252782

Temat: Timery

Zadanie 1

Zadanie polegało na napisaniu programu umożliwiającego start/stop zegara wyświetlanego na LCD. Wciśnięcie przycisku **D** z klawiatury zatrzymuje zegar, **A** - wznowia jego działanie.

Taktyka: Rejestr 1 zawiera informacje o stanie zegara:

- wartość 1 w rejestrze - zegar chodzi (start programu* oraz wciśnięcie przycisku A)
- wartość 0 - zegar stop (wciśnięcie przycisku D)

*w łatwy sposób można zmienić działanie programu na początku, tzn. jeśli zmodyfikujemy wartość zadeklarowanej w pamięci programu wartości START_STATE na 0, program zacznie działanie od zatrzymanego zegara, 1 - zegar chodzi już po załadowaniu programu

```
//kody skaningowe znakow specjalnych
#define LETTER_A 0x7E
#define LETTER_D 0xEE

#define START_STATE 0x01 ; decydujaca czy program rozpoczyna sie z wlaczonym
czasomierzem

...

START:  init_LCD
        MOV R1, #START_STATE    ; zmienna stala zdefiniowana na gorze, decydujaca
czy program rozpoczyna sie z wlaczonym czasomierzem
        MOV TMOD, #01H          ; konfiguracja timera
        MOV TH0, #3CH            ; ladowanie
        MOV TL0, #0B0H          ; stalej timera na 50ms
        SETB TR0                 ; timer start
        MOV IE, #82H            ; przerwania wlacz
        MOV R5, #00H            ; inicjacja zegara
        MOV R6, #00H
        MOV R7, #0FFH
        ACALL ZEGAR              ; wyswietlenie zainicjowanego zegara
        MOV A, #0FH
        MOV P1, A                ; zapalenie diod
        MOV R0, #20              ; licznik odmierzen 20 x 50ms
CZEKAM: MOV A, R0                ; czekam, a timer
        ;JNZ CZEKAM              ; mierzy laczny czas 1s
```

```

        DJNZ R0, CZEKAM          ; na potrzebe testowania zmienilismy te linijke,
aby oprócz skoku warunkowego, dekrementowała R0
        MOV R0, #20              ; po zgłoszeniu przerwania - ustawiam na nowo
licznik odmierzen 20 x 50ms
        ACALL ZEGAR              ; uruchomienie procedury obsługi i wyświetlenia
zegara
        MOV A, P1                ; zmiana
        CPL A                    ; świecenia
        MOV P1, A                ; diód

        ;-----

; zmienna r1 świadczy o tym czy program aktualnie działa w stanie START czy
STOP
; init: r1 -> 1 zegar chodzi
; D: r1 -> 0 zegar stop
; A: r1 -> 1 zegar start

key_1:
        mov r0, #LINE_1 ;0111 1111 sterowanie na p5
        mov a, r0
        mov P5, a ;linia 1 na p5
        mov a, P7 ;wartosc p7 do a
        anl a, r0 ;and p7 i p5

        ;sprawdzenie czy kliknieto "A", kod skaningowy = 0x7E
        clr c
        subb a, #LETTER_A
        jz key_4

        ; jesli nie wcisnieto przycisku A sprawdz czy program nie jest w
stanie zegara STOP (0)
        ; jesli r1 rowne zero - zegar stop - petla powraca do key_1
        mov a, r1
        jz key_1
        ;acall delay

key_4:  mov r1, #01 ; jesli program kiedykolwiek dochodzi do tej linijki
oznacza to, ze przycisk STOP nie byl klikniety ani razu
        ; lub ze petla w key_1 zostala przerwana wcisnieciem A ->
wtedy ustawiamy r1 zeby wskazywalo na stan zegara - START (1)
        mov r0, #LINE_4
        mov a, r0
        mov P5, a
        mov a, P7
        anl a, r0

        ;sprawdzenie czy kliknieto "D", kod skaningowy = 0xEE
        clr c
        subb a, #LETTER_D
        jnz CZEKAM ; jesli nie wcisnieto przycisku to skocz do aktualizacji
zegara

```

```
        ; jesli wcisnieto przycisk D to ustaw stan zegara r1 na STOP (0) i
wykonuj petle key_1 do momentu wcisniecia przycisku A
        MOV r1, #00
        jmp key_1
        ;-----

        NOP
        NOP
        NOP
        JMP $
END START
```

Warto zaznaczyć, że bazą napisanego przez nas programu jest program napisany przez prowadzącego na laboratoriach.