



Instytut Teleinformatyki



Wydział Fizyki, Matematyki i Informatyki
Politechnika Krakowska

Mikroprocesory i mikrokontrolery

„Obsługa portów wejścia i wyjścia procesora AVR – joystick i diody”

laboratorium: 07
autor: mgr inż. Katarzyna Smelcerz

Kraków, 2015

Spis treści

Spis treści	2
1. Wiadomości wstępne	3
1.1. Niezbędne wiadomości	3
1.2. Konfiguracja portów	3
2. Przebieg laboratorium	4
2.1. Zadanie 1. Na ocenę 3.0 (dst)	4
2.2. Zadanie 2. Na ocenę 4.0 (db)	5
2.3. Zadanie 3. Na ocenę 5.0 (bdb)	5

1. Wiadomości wstępne

Pierwsza część niniejszej instrukcji zawiera podstawowe wiadomości teoretyczne dotyczące omawianego tematu. Poznanie tych wiadomości umożliwi prawidłowe zrealizowanie praktycznej części laboratorium.

1.1. Niezbędne wiadomości

Pamięć programu, wewnętrzna pamięć RAM, mapa pamięci, rejestry, tryby adresowania oraz lista instrukcji, dostępna w dokumentacji:
<http://www.atmel.com/images/doc2503.pdf>

Schemat płytki ewaluacyjnej AVRZL15
<http://www.cyfronika.com.pl/uruchomieniowe/zl15avr.pdf>

Bieżące laboratorium jest przeprowadzane w środowisku AVR Studio4. Do fizycznego zaprogramowania mikrokontrolera używany jest programator (np. KamPROG)

1.2. Konfiguracja portów

W poprzednim laboratorium została omówiona konfiguracja portów dla stanu wyjścia. W bieżącym zestawie zadań, celem jest zarówno konfiguracja portów jako wyjść (diody), jak i wejść (joystick). Poniżej na Listingu 1 znajduje się przykładowa konfiguracja portu A jako wyjścia.

Listing 1 Przykładowy fragment programu ustawiający port A jako wejście

LDI	R16, 0x00	; załadowanie rejestru roboczego wartością 00
OUT	DDRA,r16	; ustawienie portu A jako wejścia
LDI	R16, 0xFF	; załadowanie rejestru roboczego wartością FF
OUT	PORTA, R16	; ustawienie wejścia na stan wysoki

2. Przebieg laboratorium

Druga część instrukcji zawiera zadania do praktycznej realizacji, które demonstrują zastosowanie technik z omawianego zagadnienia.

2.1. Zadanie 1. Na ocenę 3.0 (dst)

Proszę przeanalizować program przedstawiony na Listingu. Następnie proszę go nagrać na zestaw uruchomieniowy i przeanalizować kod raz jeszcze w trybie pracy krokowej. Proszę na podstawie schematu do płytki, połączyć ze sobą złącza, na które wyprowadzony jest joystick oraz diody, z odpowiednimi portami. Proszę zwrócić uwagę na działanie instrukcji CPI oraz ANDI oraz flagi, które mogą one ustawić.

Listing 2

```
.nolist
.include "m32def.inc"
.list
.listmac
.cseg
.org 0
rjmp ResetProcessor ;

ResetProcessor      :
    CLI
    LDI R16, LOW(RAMEND)
    OUT SPL, R16

    LDI R16, 0x00
    OUT DDRA,r16      ; port A - wejście
    LDI R16, 0xFF
    OUT PORTA, R16    ; wejścia na stanie wysokim

    LDI R16, 0xFF
    OUT PORTB, R16    ; B - wyjście
    OUT DDRB, R16     ; wyjścia na stanie wysokim
Main                :
    IN R16, PINA      ; wczytanie PINA do R16
```

```
ANDI R16, 0x03 ;sprawdzenie stanu pinu PA0
CPI R16, 0x02
BREQ Wcisniete ; tak: skok do etykiety (jeśli Z=1)

Wcisniete RJMP Main ; nie: powrót do pętli Main
: ;
CBI PORTB, 0 ; Zmiana stanu diody
RJMP Main ;
.exit
```

2.2. Zadanie 2. Na ocenę 4.0 (db)

Program sygnalizujący zwarte lub rozwarte styki. Proszę na podstawie schematu do płytki, połączyć ze sobą złącza, na które wyprowadzony jest joystick oraz diody, z odpowiednimi portami. Proszę wprowadzić modyfikacje do programu prezentowanego na Listingu 2, tak aby dioda reagowała na joystick, tylko przez czas, w którym jest on wciśnięty. W momencie, w którym użytkownik puszcza guzik, dioda ma powrócić do stanu pierwotnego.

2.3. Zadanie 3. Na ocenę 5.0 (bdb)

Proszę na podstawie schematu do płytki, połączyć ze sobą złącza, na które wyprowadzony jest joystick oraz diody, z odpowiednimi portami. Proszę wprowadzić modyfikacje do programu prezentowanego na Listingu 2, tak aby w momencie wciśnięcia kierunku W na joysticku, reagowała zmianą stanu dioda D1, w chwili wciśnięcia kierunku N powracała do stanu poprzedniego.