



# Instytut Teleinformatyki



Wydział Fizyki, Matematyki i Informatyki  
Politechnika Krakowska

---

## Mikroprocesory i mikrokontrolery

---

### „Obsługa portów wejścia/wyjścia mikrokontrolera 8051”

laboratorium: 02  
autor: mgr inż. Michał Lankosz  
dr hab. Zbysław Tabor, prof. PK

Kraków, 2014

## Spis treści

Spis treści .....	2
1. Wiadomości wstępne .....	3
1.1. Niezbędne wiadomości .....	3
1.2. Zestaw ZL2MCS51 .....	4
2. Przebieg laboratorium .....	5
2.1. Zadanie 1. Na ocenę 3.0 (dst) .....	5
2.2. Zadanie 2. Na ocenę 4.0 (db) .....	6
2.3. Zadanie 3. Na ocenę 5.0 (bdb) .....	7
2.4. Zadanie 4. Na ocenę 5.0 (bdb) .....	8

# 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: <http://www.8052.com/tut8051>.

Lista instrukcji asemblera 8051:

[http://www.keil.com/support/man/docs/is51/is51\\_instructions.htm](http://www.keil.com/support/man/docs/is51/is51_instructions.htm)

Laboratorium asemblera 8051 jest realizowane w środowisku developerskim [Keil microVision](#) (środowisko cross-kompilacji C51 i symulacji pracy 8051), do fizycznego oprogramowania mikrokontrolera używamy programator Flip 3.4.7.

Tematem ćwiczenia jest demonstracja sterowania liniami wejścia/wyjścia mikrokontrolera 8051 i obsługi prostych urządzeń podłączonych do wyprowadzeń mikrokontrolera, takich jak LED i przełączniki. Ponieważ porty są mapowane na wewnętrzną pamięć RAM (obszar rejestrów specjalnych), odczyt i zapis do portów odbywa się tak samo jak odczyt i zapis innych komórek pamięci.

Przykład wpisania danej 1F do portu P2:

```
MOV P2,#1Fh
```

Przykład wyzerowania bitu 5 portu P2:

```
CLR P2.5
```

Przykład odczytu portu P3:

```
MOV A,P3
```

Przykład skoku warunkowego, gdy występuje stan niski na wyprowadzeniu P3.0:

```
JNB P3.0,skok1
```

## 1.2. Zestaw ZL2MCS51

Programowanie zestawu ZL2MCS51 powinno przebiegać w następującej kolejności. Na początek w ustawieniach projektu w programie  $\mu$ Vision należy ustawić opcje generowania pliku HEX, który jest jednym z najpopularniejszych formatów przechowywania skompilowanego programu. W tym celu z górnego menu należy wybrać **Project -> Options for Target** i w zakładce **Output** zaznaczyć opcję **Create HEX File**. Podczas kompilacji projektu, w folderze z projektem zostanie utworzony plik o nazwie identycznej z nazwą projektu i rozszerzeniem hex. Następnie, jeśli płytką zestawu jest podłączona do komputera PC złączem RS232 i jest do niej podłączone zasilanie, należy użyć programu **Flip**, który umieści skompilowany program w pamięci flash mikrokontrolera. Jeśli program Flip został wcześniej prawidłowo skonfigurowany, wystarczy z menu File wybrać **Load HEX File** i wskazać właściwy plik. Następnie wcisnąć przycisk **Run**, a po zaprogramowaniu przycisk **Start Application** lub przycisk **RESET** zestawu uruchomieniowego. Do czasowej analizy pracy programu w sesji symulatora trzeba określić częstotliwość zegarową. W tym celu z górnego menu należy wybrać **Project -> Options for Target** i w zakładce **Target** wpisać częstotliwość rezonatora kwarcowego równą 11.0592MHz.

Zastosowane w zestawie diody LED (D0, D1,..., D7) są podłączone do odpowiednich linii portu P2 poprzez przełącznik DIP. Diody świecą po ustawieniu przełącznika w pozycji ON i podaniu na odpowiednią linię portu wartości 0. Zestaw wyposażony jest również w 8-przyciskową klawiaturę (przyciski S0, S1,..., S7) podłączoną do linii portu S3. Wciśnięcie przycisku powoduje, że na porcie pojawia się stan 0.

### Literatura:

- [1] <http://www.8052.com/tut8051>
- [2] [http://www.keil.com/support/man/docs/is51/is51\\_instructions.htm](http://www.keil.com/support/man/docs/is51/is51_instructions.htm)

## 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)

Należy przeanalizować pracę mikrokontrolera poprzez wykonanie symulacji krokowej, a następnie zaprogramować fizyczny mikrokontroler i sprawdzić działanie programu przedstawionego na Listingu 1.

#### Listing 1

```
PROG SEGMENT CODE

CSEG AT 0
JMP start

RSEG PROG
start:
    CLR P2.5
    ;MOV P2,#0AAh
    SJMP $
END
```

Za pomocą symulatora/debuggera proszę prześledzić działanie powyższego programu, zwracając uwagę na stan portu P2 (**Peripherals -> I/O Ports -> P2**). Proszę odkomentować drugą linię programu głównego i sprawdzić działanie programu.

Proszę skompilować program z opcją generowania pliku HEX i zaprogramować mikrokontroler zestawu uruchomieniowego. Proszę sprawdzić działanie obu programów.

Na Listingu 2 przedstawiono kod programu sterującego pracą diod i wykorzystującego proste pętle opóźniające, przeniesione do osobnych procedur.

### Listing 2

```
PROG SEGMENT CODE

CSEG AT 0
JMP start

RSEG PROG
start:
    MOV P2,#0AAh
    ACALL CZEKAJ
    MOV P2,#055h
    ACALL CZEKAJ
    SJMP start

CZEKAJ:
    MOV R0,#255
ODLICZAJ2:
    MOV R1,#255
ODLICZAJ1:
    NOP
    NOP
    DJNZ R1,ODLICZAJ1
    DJNZ R0,ODLICZAJ2

RET

END
```

Proszę za pomocą symulatora/debuggera prześledzić działanie powyższego programu, zwracając uwagę na stan portu P2 oraz upływający czas. Proszę zaprogramować mikrokontroler i zaobserwować jego prace. Proszę zmodyfikować program tak, aby uzyskać inne niż wynikający z powyższego kodu efekty świetlne.

## 2.2. Zadanie 2. Na ocenę 4.0 (db)

Na Listingu 3 przedstawiono kod programu sterującego pracą diod i odczytującego stan klawiszy. Po wciśnięciu klawiszy zestawu testowego na odpowiednim porcie pojawia się stan niski (0). Klawiszowi wyciśniętemu odpowiada stan wysoki (1)

### Listing 3

```
PROG SEGMENT CODE

LED BIT P2.0
SWITCH BIT P3.7

CSEG AT 0
JMP start

RSEG PROG
start:
    MOV C,SWITCH
    MOV LED,C
    ;MOV A,P3
    ;ORL A,#0Fh
    ;MOV P2,A
    SJMP start
END
```

Proszę zaprogramować mikrokontroler powyższym programem i sprawdzić reakcję na wciśnięcie przycisku S7.

Zamiast dwóch pierwszych rozkazów po etykiecie start proszę odkomentować trzy następne. Proszę sprawdzić program w rzeczywistym układzie.

Proszę napisać program, który po każdym wciśnięciu klawisza S0 zmieni stan diody D1 na przeciwny. W tym celu można użyć rozkazów:

### **JB switch, skok CPL LED**

Jeśli program nie funkcjonuje prawidłowo należy użyć symulatora i zasymulować wciśnięcie klawisza poprzez odpowiednie wypełnienie pól Pins podglądu stanu portu P3.

## **2.3. Zadanie 3. Na ocenę 5.0 (bdb)**

Proszę napisać program, który zapala/włącza diodę o numerze n po naciśnięciu klawisza o numerze n. Program ma umożliwić sterowanie zapalaniem/włączaniem WSZYSTKICH diod.

## 2.4. Zadanie 4. Na ocenę 5.0 (bdb)

Proszę napisać program testujący refleks użytkownika zestawu testowego ZL2MCS51. Diody D0-D3 zapalają się w losowej kolejności na krótki odstęp czasu (proszę wykorzystać generator liczb pseudolosowych z poprzedniego zestawu). Czas wyświetlania i przerwę można zmniejszać/zwiększać klawiszami S7/S6. Jeśli w czasie wyświetlania na danej diodzie użytkownik wciśnie odpowiedni klawisz, wyświetlany jest na diodach sygnał zakończenia (dowolna sekwencja kolejno zapalanych diod) i program kończy działanie.

**Bardziej ambitna wersja:** Program startuje od małego okresu wyświetlania na diodach. Każde nieudane N prób powoduje powiększenie czasu wyświetlania.