

## Technika Mikroprocesorowa

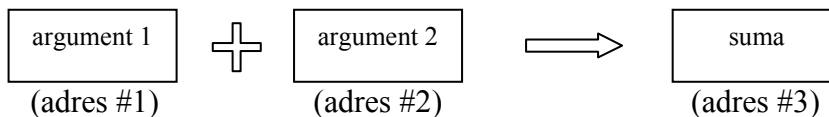
# Laboratorium 1: Wprowadzenie do środowiska programowego oraz podstawowe operacje na rejestrach i komórkach pamięci

### Cel ćwiczenia:

Zapoznanie się ze środowiskiem programowym: poznanie funkcji asemblera, poznanie funkcji symulatora. Operacje na plikach, edycja programu, ustawianie parametrów programów, uruchamianie programów, uruchamianie krokowe, zastawianie pułapek, obserwacja rejestrów.

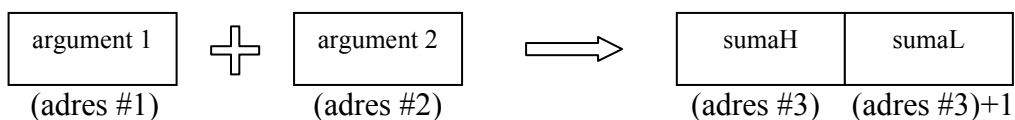
### Zadania do wykonania:

1. Napisać program w asemblerze sumujący zawartość dwóch komórek pamięci i zapisujący wynik dodawania do trzeciej komórki:



2. Uruchomić program w symulatorze i wykonując go krok po kroku sprawdzić poprawność jego działania dla kilku zestawów argumentów, zwracając uwagę na zmiany zawartości rejestrów procesora (zwłaszcza akumulatora i rejestru stanu).

3. Wyjaśnić przyczynę „błędny” wyniku w komórce ‘adres #3’ dla argumentów, których suma jest większa niż 255 (0FFh). Zmodyfikować program tak, aby wynik dodawania był zapisywany jako liczba dwubajtowa:



### Elementy wymagane przy sprawozdaniu:

Charakterystyka mikroprocesora, opis użytych rozkazów, kod oraz opis programu.

## Technika Mikroprocesorowa

### Laboratorium 2: Budowa pętli programowych

#### Cel ćwiczenia:

Zapoznanie się z realizacją operacji iteracyjnych w języku assembler.

#### Zadania do wykonania:

1. Zbudować pętlę typu **for** – napisać program w assemblerze wykonujący zadaną ilość razy określoną operację np. inkrementację zawartości wybranego rejestru. Wykorzystać do budowy pętli instrukcję **DJNZ** i licznik pętli w arg1.

START:  
    } instrukcje programu  
DJNZ arg1, START

2. Zbudować pętlę typu **repeat..until** – napisać program, który w pętli modyfikuje (np. inkrementuje) zawartość arg1 tak długo aż nie osiągnie on wartości takiej, jaką ma arg2.

START:  
    } instrukcje programu  
CJNE arg1, arg2, START

#### Elementy wymagane przy sprawozdaniu:

Opis użytych rozkazów (w tym instrukcji warunkowych), kod oraz opis programu.

## Technika Mikroprocesorowa

### Laboratorium 3: Zastosowanie instrukcji warunkowych i adresowania pośredniego

#### Cel ćwiczenia:

Zapoznanie się z technikami adresowania pośredniego i ich realizacją iteracyjną.

#### Zadania do wykonania:

1. Napisać program w języku asembler realizujący sumowanie zawartości określonej ilości komórek pamięci. Przy założeniu, że liczba komórek nie będzie większa niż 255 wynik sumowania ma być zapisany jako liczba dwubajtowa.

Zakres komórek ma być podany na dwa sposoby:

- adres pierwszej komórki i ilość komórek
- adres pierwszej i adres ostatniej komórki

W programie uwzględnić możliwość wystąpienia przepełnienia po każdym dodawaniu w pętli.

2. Rozwiązać zadanie na dwa sposoby:

- zbudować pętlę **for**
- zbudować pętlę typu **repeat..until**

Do adresowania sumowanych komórek użyć adresowania pośredniego z wykorzystaniem rejestru R0.

#### Elementy wymagane przy sprawozdaniu:

Charakterystyka metod adresowania mikroprocesora 8051, kod oraz opis programu.

## Technika Mikroprocesorowa

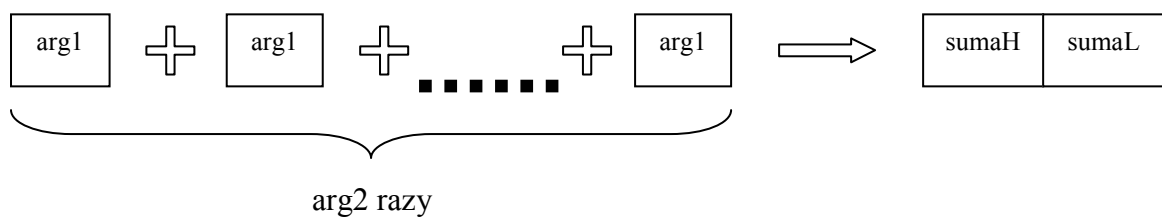
### Laboratorium 4: Zastosowanie poznanych technik w bardziej złożonych algorytmach

#### Cel ćwiczenia:

Opanowanie konstrukcji programowych języka assembler.

#### Zadania do wykonania:

1. Napisać program realizujący mnożenie dwóch liczb na zasadzie wielokrotnego dodawania, wraz sprawdzeniem czy któryś z argumentów nie jest równy zero. Rozbudować program tak, aby przynajmniej jeden z czynników mógł być dwubajtowy.



2. Zmodyfikować program w celu zoptymalizowania szybkości jego działania tzn. tak by w razie potrzeby mógł przestawić argumenty tak, aby  $\text{arg1} > \text{arg2}$ .

Program ma być napisany tak żeby pętla sumująca występowała w nim tylko jeden raz.

#### Elementy wymagane przy sprawozdaniu:

Kod oraz opis programu.

## **Technika Mikroprocesorowa**

### **Laboratorium 5: Podprogramy, przekazywanie parametrów, obsługa klawiatury**

#### **Cel ćwiczenia:**

Zapoznanie się z konstrukcją podprogramów w języku assembler, obsługa symulatora klawiatury.

#### **Zadania do wykonania:**

1. Napisać program prostego kalkulatora, w którym wcześniej napisane programy będą wykorzystane jako procedury realizujące określone funkcje np. dodawanie, mnożenie, wypisywanie tekstów. Do sterowania pracą kalkulatora użyć symulatora klawiatury, której stan ma być sprawdzany przez program główny.
2. Zwrócić uwagę na sposób wymiany informacji między programem głównym a procedurami.

#### **Elementy wymagane przy sprawozdaniu:**

Kod oraz opis programów, konstrukcja podprogramów, obsługa klawiatury.

## **Technika Mikroprocesorowa**

# **Laboratorium 6: Programowanie układów czasowo/licznikowych, obsługa systemu przerwań**

### **Cel ćwiczenia:**

Zapoznanie się z budową i obsługą układów czasowo-licznikowych oraz systemu przerwań mikrokontrolera 8051.

### **Zadania do wykonania:**

1. Napisać program stopera wykorzystującego jako wzorzec czasu wbudowany w mikrokontroler licznik generujący przerwania ze ściśle określoną częstotliwością.
2. Napisać program, który korzystając z wbudowanych liczników będzie realizował funkcję alarmu. Do ustawiania alarmu użyć symulatora klawiatury, której stan ma być sprawdzany przez program główny. Zwrócić uwagę na odpowiednie ustawienie priorytetów przerwań.

### **Elementy wymagane przy sprawozdaniu:**

Charakterystyka układów czasowo-licznikowych i systemu przerwań, kod oraz opis programów.

## Technika Mikroprocesorowa

### **Laboratorium 7: Programowanie układów wejścia/wyjścia mikrokontrolera**

#### **Cel ćwiczenia:**

1. Programowania portów wejścia/wyjścia.
2. Sterowanie urządzeniami włączanymi przy pomocy linii portu.
3. Obsługa wyświetlacza alfanumerycznego LCD.
4. Obsługa wyjścia analogowego.

#### **Zadania do wykonania:**

1. Napisać program realizujący na wyjściach jednego z portów tzw. „krążącą jedynekę”.
2. Korzystając z systemu DSM-51 i gotowych procedur napisać program sterujący diodą świecącą i brzęczykiem np. załączający je na zadany okres czasu z zadaną częstotliwością. Do obsługi programu wykorzystać klawiaturę systemu DSM-51.
3. Napisać program wypisujący na wyświetlaczu tekstowo numer naciśniętego klawisza.
4. Napisać program generujący na wyjściu analogowym przebieg trójkątny i piłokształtny.