**MERSUL LUCRĂRII**

**Exerciţiul 1  
1.Indicarea continutului memoriei si schimbarea lui**

**1.1**

|  |  |  |
| --- | --- | --- |
| **Adresa** | **Cod** | **Verificare** |
| 080A | E5 | + |
| 0801 | 10 | + |
| 0802 | 20 | + |
| 0803 | 30 | + |
| 0804 | 40 | + |
| 0805 | 50 | + |
| 0806 | 60 | + |
| 0807 | 70 | + |
| 0808 | 80 | + |
| 0809 | 90 | + |
| 080A | A0 | + |
| 080B | B0 | + |
| 080C | C0 | + |
| 080D | D0 | + |
| 080E | E0 | + |
| 080F | F0 | + |
| 0810 | 0F | + |

|  |  |  |
| --- | --- | --- |
| **Adresa** | **Cod** | **Verificare** |
| 0A00 | C3 | + |
| 0C01 | 05 | - |
| 0100 | 03 | - |

**1.2**

In urma verificarii memoriei completate prin presarea tastei “П” apoi a lacunei “-“ in mod succesiv,am descoperit ca datele se inscriu in toate adresele de memorie,in afara de adresele 0C01 si 0100.Acest lucru se intimpla deaorece dispozitivul de memorie operativa DMO este limitat intre adresele 0800 si 0BFF,iar adresele noastre 0C01 si 0100 depasesc aceasta limita din dreapta intervalului,si respective din stinga.

**1.3**

|  |  |  |
| --- | --- | --- |
| **Adresa** | **Cod** | **Verificare** |
| 0800 | 00 | + |
| 0801 | 01 | + |
| 0802 | 02 | + |
| 0803 | 03 | + |
| 0804 | 04 | + |
| 0805 | 05 | + |
| 0806 | 06 | + |
| 0807 | 07 | + |
| 0808 | 08 | + |
| 0809 | 09 | + |
| 080A | 0A | + |
| 080B | 0B | + |
| 080C | 0C | + |
| 080D | 0D | + |
| 080E | 0E | + |
| 080F | 0F | + |
| 0810 | 10 | + |

Pentru a verifica informatia introdusa efectuam urmatoarele operatii:  
1.Tastam butonul “СБ” pentru a reseta SPMS in stare initiala;  
2.Apasam tasta “П” pentru a citi informatia inscrisa in memorie;  
3.Prin apasarea succesiva a tastei “-“ citim continutul tuturor adreselor de memorie.

**Concluzie:** La introducerea valorilor noi in celulele incarcate anterior,valorile vechi se pierd!!!

**2.Indicarea si schimbarea continutului registrelor  
2.1**

|  |  |
| --- | --- |
| **Registru** | **Cod** |
| PCH | 00 |
| PCL | 40 |
| SPH | 0B |
| SPL | CA |
| H | 12 |
| L | 34 |
| A | AA |
| B | BB |
| C | CC |
| D | DD |
| E | EE |
| F | FF |

Registrele PCH,SPH,H-retin bitii superiori,iar PCL,SPL si L-bitii inferiori.

**2.2**

|  |  |  |
| --- | --- | --- |
| **Registru** | **Cod** | **Verificare** |
| A | 00 | + |
| B | 01 | + |
| C | 02 | + |
| D | 03 | + |
| E | 04 | + |
| F | 05 | + |
| PH | 06 | + |
| PL | 07 | + |
| SH | 08 | + |
| SL | 09 | + |

**2.3** In urma verificarii continutului registrilor am vazut ca datele au fost introduse corect.

**Concluzie :** In orice registru putem introduce informatie(atat in registrul inferior,cit si in cel superior) !!!

**3.Completarea masivului memoriei cu o constanta**

**3.1**

Dispozitivul de memorie operativa(DMO) se utilizeaza pentru stocarea programelor si datelor ce se schimba. DMO are capacitatea de 1 kilobyte cu adresele 0800-0BFF,din care ultimele 54 celule de memorie se utilizeaza de catre programul “Monitor” pentru “stiva”. Deci,pentru a calcula analitic adresa inceputului stivei,este necesar de a scadea 0BFF-54 celule. Insa,0BFF trebuie transformat in prealabil in sistemul zecimal pentru a efectua aceasta operatie:  
***(0BFF)16=(3071)10***

Apoi efectuam operatia propriu-zisa de scadere: ***(3071)10 - (54)10 = (3017)10***

Pentru a afla adresa inceputului stivei reconvertim numarul in sistemul hexazecimal:  
***(3017)10=(BC9)16***

Comparind aceasta valoare cu ce-a citita din SPMS observam ca valoarea analitica indica la o pozitie cu o unitate mai joasa decit cea citita anterior.

**3.2**  Pentru a completa masivul cu o constanta ne conducem de procedura urmatoare:  
Tastam butonul **“ЗК” ADRESA1 <<->>ADRESA2 <<->> D <<ВП>>**  
 unde ADRESA1,ADRESA2 sunt corespunzator adresa initiala si cea finala ale masivului memoriei,ce se completeaza;iar D-o constanta,un byte de date.

|  |  |  |
| --- | --- | --- |
| **Adresa** | **Constanta** | **Verificare** |
| 0A00 | AA | + |
| 0A01 | AA | + |
| 0A02 | AA | + |
| 0A03 | AA | + |
| 0A04 | AA | + |
| 0A05 | AA | + |
| 0A06 | AA | + |
| 0A07 | AA | + |
| 0A08 | AA | + |
| 0A09 | AA | + |
| 0A0A | AA | + |
| 0A0B | AA | + |
| 0A0C | AA | + |
| 0A0D | AA | + |
| 0A0E | AA | + |
| 0A0F | AA | + |
| 0A10 | AA | + |

In urma verificarii am confirmat ca constanta AA a fost inscrisa in toti registrii.

**4.Determinarea sumei de control a masivului memoriei**  Suma de control se determina efectuand urmatorii pasi:  
<<КС>> ADRESA1 <<->> ADRESA2 <<ВП>>

**4.1**

|  |  |  |
| --- | --- | --- |
| **Adresa** | **Date** | **Suma de Control** |
| 0800 | 01 |  |
| 0801 | 03 |  |
| 0802 | 05 |  |
| 0803 | 07 | ***SC=3E*** |
| 0804 | 09 |  |
| 0805 | 0A |  |
| 0806 | 0C |  |
| 0807 | 0F |  |

**4.3**

SC calculata analitic: ***01+03+05+07+09+0A+0C+0F= 3E*  
Concluzie:** Observam ca SC calculata analitic este egala cu cea calculata automat!!!

**4.4**

|  |  |  |
| --- | --- | --- |
| **Adresa** | **Date** | **Suma de Control** |
| 0800 | 01 |  |
| 0801 | 03 |  |
| 0802 | 05 |  |
| 0803 | 07 |  |
| 0804 | 09 | ***SC=2E*** |
| 0805 | 0A |  |
| 0806 | 0C |  |
| 0807 | 0F |  |
| 0808 | F0 |  |

SC calculata analitic: ***3E+F0=12E***

**Concluzie:** In acest caz observam ca rezultatul analitic difera de cel obtinut cu ajutorul SMPS.Asta se intimpla deoarece o celula de memorie are doar un octet,iar rezultatul obtinut SC depaseste 1 octet.Rezulta ca ne este afisat doar bitii inferiori-2E, bitii superiori fiind omisi.

**5.Deplasarea masivului memoriei in spatiul adreselor**

Deplasarea masivului se efectueaza presind succesiv urmatoarele taste:  
<<ПМ>> ADRESA1 <<->> ADRESA2 <<->> ADRESA3 <<ВП>>  
unde ADRESA3 este adresa initiala a masivului deplasarii.

**5.1**

|  |  |
| --- | --- |
| **Adresa** | **Date** |
| 0800 | 01 |
| 0801 | 03 |
| 0802 | 05 |
| 0803 | 07 |
| 0804 | 09 |
| 0805 | 0A |
| 0806 | 0C |
| 0807 | 0F |
| 0808 | F0 |

**5.2**

Pentru a deplasa masivul nostru efectuam urmatoarea succesiune de operatii:  
<<ПМ>> 0800 <<->> 0808 <<->> 0A10 <<ВП>>

**5.3**

In urma verificarii continutului masivelor plasarii si deplasarii am descoperit ca masivul nostru s-a deplasat(copiat),pozitia initiala a acestuia fiind 0A10,iar cea finala 0A18.

|  |  |
| --- | --- |
| **Adresa** | **Date** |
| 0A10 | 01 |
| 0A11 | 03 |
| 0A12 | 05 |
| 0A13 | 07 |
| 0A14 | 09 |
| 0A15 | 0A |
| 0A16 | 0C |
| 0A17 | 0F |
| 0A18 | F0 |

**Concluzie:** La deplasare se pastreaza ambele masive!!!

**Ministerul Educației al Republicii Modova**

**Universitatea de Stat din Moldova**

**Facultatea de Fizică si Inginerie**

**Catedra:Fizica si Informatica aplicata**

**Cursul: Logica si Sisteme Digitale**

**Lucrare de laborator Nr. 1**

**Tema: “Studierea complexului microprocesor de studiu”**

|  |
| --- |
| **Efectuat de studentul grupei 2.2 anul II:**  **Ciobanasu Ion**  **Controlat de: conf.univ.**  **Chetrus Petru** |

**Chișinău 2015**