Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики і обчислювальної техніки

Кафедра обчислювальної техніки

Лабораторна робота №2

З архітектури комп’ютерів

*Виконав:*

Студент групи ІО-12

**Нестерук Ю.О.**

**ЗК №1220**

*Перевірив:*

Сергієнко А.М.

м. Київ

2013 р.

**1. Варіант завдання:**

№ ЗК: 1220

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| a6 | a5 | a4 | a3 | a2 | a1 |
| 0 | 0 | 0 | 1 | 0 | 0 |

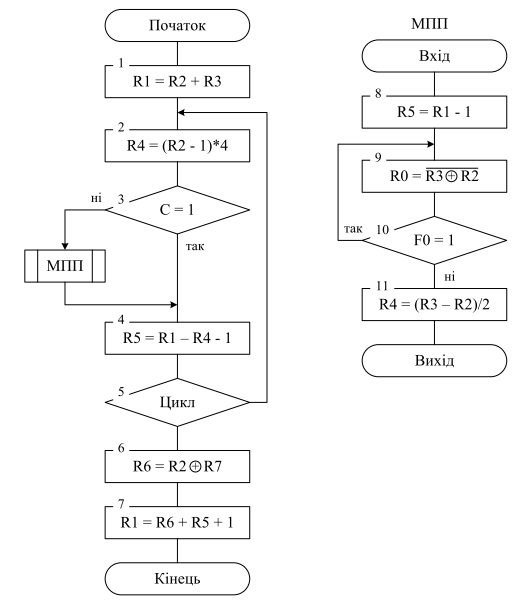
Значення операндів:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| h2 | h1 | X0 | X1 | X2 | X3 | X4 | X5 | X6 | Кількість ітерацій циклу |
| R0 | R1 | R2 | R3 | R4 | R5 | R6 | R7 |
| 0 | 0 | 4A | D3 | E5 | 24 | 06 | 20 | 05 | 3 |

|  |  |  |
| --- | --- | --- |
| h5 | h2 | Порти |
| 0 | 0 | Р1 |

|  |  |  |
| --- | --- | --- |
| h4 | h3 | Алгоритм |
| 0 | 1 | Рис. 6.1, б |

**Алгоритм:**



**2. Лістинг програми:**

; Занесення даних (X0-X6) у регістри(R0-R6)

In A, P1

Mov R0, A

In A, P1

Mov R1, A

In A, P1

Mov R2, A

In A, P1

Mov R3, A

In A, P1

Mov R4, A

In A, P1

Mov R5, A

In A, P1

Mov R6, A

Mov R7, #3

;=====================================================

; R1 = R2 + R3

; Переводимо R2 у доповняльний код:

Mov A, R2

Jb7 Dk1

; Регістр R2 додатній, нічого не робимо:

Jmp Ndk1

Dk1:

Sel Rb1

Mov R2, #FFH

Sel Rb0

Ndk1:

; Переводимо R3 у доповняльний код:

Mov A, R3

Jb7 Dk2

; Регістр R3 додатній, нічого не робимо:

Jmp Ndk2

Dk2:

Sel Rb1

Mov R3, #FFH

Sel Rb0

Ndk2:

; Після приведення у ДК, можна виконувати додавання:

Add A, R2

Mov R1, A

Sel Rb1

Mov A, R3

Addc A, R2

Mov R1, A

Sel Rb0

;=====================================================

; Цикл:

Cycle:

; Тут починається ітерація циклу:

;=====================================================

; R4 = (R2 - 1) \* 4:

Mov A, R2

Jnz Nc1

; Потрібен перенос із старших розрядів:

Sel Rb1

Dec R2

Sel Rb0

Nc1:

; Перенос зі старших розрядів не потрібен:

Dec A

; Заносимо у R4 <- (R2 - 1):

Mov R4, A

Sel Rb1

Mov A, R2

Mov R4, A

Sel Rb0

; Виконуємо множення на 4 (звувами вліво):

Mov A, R4

Clr C

Rlc A

Mov R4, A

Jnc Nc2

Sel Rb1

Mov A, R4

Rlc A

Mov R4, A

Sel Rb0

Nc2:

Mov A, R4

Clr C

Rlc A

Mov R4, A

Jnc Nc4

Sel Rb1

Mov A, R4

Rlc A

Mov R4, A

Sel Rb0

Nc4:

;=====================================================

; Якщо С <> 1, то виконаємо перехід до МПП:

Jnc Mpp

Nmpp:

;=====================================================

; R5 = R1 - R4 - 1:

Mov A, R4

Cpl A

Add A, R1

Mov R5, A

Sel Rb1

Mov A, R4

Cpl A

Addc A, R1

Mov R5, A

Sel Rb0

;=====================================================

; Якщо R7 <> 0, то виконуємо ще одну ітерацію циклу:

Djnz R7, Cycle

;=====================================================

; Після завершення циклу

;=====================================================

; R6 = R2 xor R7

; Оскільки по завершенню циклу R7 = 0, a

; (R2 xor 0) = R2, то просто заносимо в R6 <- R2:

Mov A, R2

Mov R6, A

Sel Rb1

Mov A, R2

Mov R6, A

Sel Rb0

;=====================================================

; R1 = R6 + R5 + 1:

Mov A, R5

Clr C

Cpl C

Addc A, R6

Mov R1, A

Sel Rb1

Mov A, R5

Addc A, R6

Mov R1, A

Jmp Endp

; Перехід до кінця програми:

;=====================================================

; МПП:

;=====================================================

Mpp:

; R5 = R1 - 1:

Mov A, R1

Jnz Nc3

; Потрібен перенос із старших розрядів:

Sel Rb1

Dec R1

Sel Rb0

Nc3:

; Перенос зі старших розрядів не потрібен:

Dec A

Mov R5, A

Sel Rb1

Mov A, R1

Mov R5, A

Sel Rb0

;=====================================================

; R0 = not(R3 xor R2)

; Врахуємо, що not(R3 xor R2) = [not(R3 or R2) or (R3 and R2)]

F0ret:

Mov A, R3

Orl A, R2

Cpl A

Mov R0, A

; Тепер R0 = not(R3 or R2)

Mov A, R3

Anl A, R2

Orl A, R0

Mov R0, A

;=====================================================

; Якщо F0 = 1, то виконуємо перехід:

Jf0 F0ret

;=====================================================

; R4 = (R3 - R2)/2:

Mov A, R2

Cpl A

Clr C

Cpl C

Addc A, R3

Mov R4, A

Sel Rb1

Mov A, R2

Cpl A

Addc A, R3

Mov R4, A

; Маємо R4 = (R3 - R2)

; Виконаємо ділення на 2 (зсувами праворуч):

Clr C

Rlc A

Mov R4, A

Sel Rb0

Mov A, R4

Rlc A

Mov R4, A

;=====================================================

; Вихід у основну програму:

Jmp Nmpp

;=====================================================

; Кінець програми:

Endp:

Nop

**3. Стан регістрів після завершення програми:**

|  |  |  |
| --- | --- | --- |
|  | RB0 | RB1 |
| R0 | 4A | 00 |
| R1 | 5E | 00 |
| R2 | E5 | FF |
| R3 | 24 | 00 |
| R4 | 90 | FF |
| R5 | 78 | 00 |
| R6 | E5 | FF |
| R7 | 00 | 00 |

Двобайтові результати зберігаються у одному (за номером) регістрі різних банків, регістр банку RB1 – старший, RB0 – молодший.