# *Національний технічний університет України*

# *«Київський політехнічний інститут»*

#### ***Факультет інформатики та обчислювальної техніки***

## Лабораторна робота №6

*з курсу "****Комп’ютерна архітектура****"*

***Виконав:***

*Бедь А.М.*

***Група*** *ІО-12,*

***Номер варіанту*** *1202*

***Київ - 2013р.***

**ВИКОНАННЯ КОМАНД В ЕОМ**

Ціль роботи – вивчити етапи виконання команд передачі керування. Навчитися розробляти мікроалгоритми і мікропрограми виконання зазначених команд.

**Завдання:**

Розробити програму в кодах команд (командний рівень), що містить чотири команди:

* умовний перехід,
* множення,
* двохадресна команда, зазначена в роботі 5,
* безумовний перехід.

Варіант 120210 = 10010110010

|  |  |  |
| --- | --- | --- |
| а5 | а4 | Операція |
| 1 | 0 | V |

|  |  |  |
| --- | --- | --- |
| а5 | а4 | Тип адресації |
| 1 | 0 | АІ |

|  |  |
| --- | --- |
| а3 | Тип адресації |
| 0 | Пряма |

**Лістинг коду:**

\Налагодження зв'язків

link ewh: 16

link l1: ct

link l2: rdm

link rb: z, 2, 1, 0

accept rdm\_delay: 2

equ COD:0800h

equ TYPE\_ADDR:0210h

equ COD1:5800h \Умовний перехід

equ COD2:6000h \Безумовний перехід

\Завдання значень комірок пам'яті

dw 0002h:1010101010101010%

dw 0004h:1111000011110000%

dw 0050h:6040h

dw 0062h:5842h

dw 0056h:8AD5h

dw 0057h:5800h

dw 0040h:0062h

dw 0042h:0056h

\Адреса мікрокоманди

accept r6:0002h

accept r5:0004h

accept r7:0050h

accept r15:0070h

start

\Область програми

{xor nil, r11, r11; oey; ewh;}

{or nil, r7, z; oey; ewl;}

\Завантеження мікрокоманди з ОП у регістр R8

load\_k

{cjp rdm, load\_k; r; or r8, bus\_d, z; load rm, flags;}

{cjp rm\_z, stop;}

\Перевірка формату команди

{cjp rm\_n, double\_a;}

\Перевірка операції

{and r12, r8, 7800h;}

{xor nil, r12, COD; load rm, flags;}

{cjp not rm\_z, nMainMul;}

\Перевірка адресації

{and nil, r8, 0400h; load rm, flags;}

{cjp not rm\_z, indirect;}

\Завантеження адреси операнда на шину даних

{xor nil, r11, r11; oey; ewh;}

\{and r11, r11, 03ffh;}

{or r11, 03ffh, z; }

{and nil, r8, r11; oey; ewl;}

\Завантаження операнда з ОП в R10

load\_o

{cjp rdm, load\_o; r; or r10, bus\_d, z;load rm, flags;}

\Множення 2-им способом

\Якщо операнд в r10 рівний нулю

\обнуляємо r15 і пропускаємо множення

{cjp not rm\_z, main\_mull;}

{xor r15, r15, r15;}

{cjp nz, end\_cmd;}

main\_mull

\Підготовка даних в регістрах

\r11,r10 - M

\r12 - X

{xor r11, r11, r11;}

{xor r12, r12, r12;}

{or r12, r12, r15;load rm, flags;}

{cjp rm\_z, end\_cmd;}

{xor r15, r15, r15;}

\Перевірка знаку для X

{or r9, r9, 8000h;}

{and r9, r9, r12; load rm, flags;}

{and r12, r12, 7fffh;}

\Перевірка знаку для M

{or r13, r13, 8000h;}

{and r13, r13, r10; load rm, flags;}

{cjp rm\_z, lable1;}

{sub r10, z, r10, nz;}

\Цикл множення

lable1

{or r5, r5, 0001h;}

{and r5, r5, r12;load rm, flags;}

{cjp rm\_z, lable2;}

{add r15, r15, r10, z; load rm, flags;}

{add r14, r14, r11, rm\_c;}

lable2

{or srl, r12, r12, z;}

{or sll, r10, r10, z;}

{or sl.25, r11, r11, z;}

{or r12, r12, z; load rm, flags;}

{cjp not rm\_z, lable1;}

\Встановлення знаку результату

{xor r9, r9, r13;}

{or r14, r14, r9;}

{cjp nz, end\_cmd;}

\Непряма адресація

indirect

{cjp nz, end\_cmd;}

\Не моя операція множення

nMainMul

\Перевірка на умовний перехід

{xor r12, r12, r12;}

{and r12, r8, 7800h;}

{xor nil, r12, COD1; load rm, flags;}

{cjp rm\_z, cond;}

\Перевірка на безумовний перехід

{xor r12, r12, r12;}

{and r12, r8, 7800h;}

{xor nil, r12, COD2; load rm, flags;}

{cjp rm\_z, uncond;}

\Якщо не один з попередніх варіантів то до іншої операції

{cjp nz, noJump;}

\Реалізація умовного переходу

cond

\Перевірка типу адресації

{xor r12, r12, r12;}

{and r12, r8, 0400h; load rm, flags;}

{cjp not rm\_z, idcon;}

\Пряма адресація

{or r12, r15, z;}

{and r12, r12, 0080h; load rm, flags;}

{cjp not rm\_z, end\_cmd;}

{or r12, 03ffh, z; }

{and nil, r8, r12; oey; ewl;}

laddrc

{cjp rdm, laddrc; r; or r14, bus\_d, z;}

{xor r7, r7, r7;}

{or r7, r14, z;}

{cjp nz, naddr;}

\Непряма адресація

idcon

{cjp nz, end\_cmd;}

{cjp nz, naddr;}

\Реалізація безумовного переходу

uncond

\Перевірка типу адресації

{xor r12, r12, r12;}

{and r12, r8, 0400h; load rm, flags;}

{cjp not rm\_z, iducon;}

\Пряма адресація

{or r12, 03ffh, z; }

{and nil, r8, r12; oey; ewl;}

laddrunc

{cjp rdm, laddrunc; r; or r14, bus\_d, z;}

{xor r7, r7, r7;}

{or r7, r14, z;}

{cjp nz, naddr;}

\Непряма адресація

iducon

{cjp nz, end\_cmd;}

{cjp nz, naddr;}

noJump

{cjp nz, end\_cmd;}

\Двоадресна команда

double\_a

\Перевірка операції

{and r12, r8, 7c00h;}

{xor nil, r12, COD; load rm, flags;}

{cjp not rm\_z, nDisyunk;}

\Перевірка типу адресації

{xor r12, r12, r12;}

{and r12, r8, 0318h;}

{xor nil, r12, TYPE\_ADDR; load rm, flags;}

{cjp not rm\_z, nAutoInc;}

\Отримуємо РЗП1 і РЗП2 в rb

{or nil, r8, z; oey; load rb;}

\Зчитуємо перший операнд з ОП

{or nil, rb, z; oey; ewl;}

{or r3, r3, 0002h;}

{add rb, rb, r3;}

load\_op1

{cjp rdm, load\_op1; r; or r0, bus\_d, z;}

\Зчитуємо другий операнд з ОП зразу робимо оп. АБО і поміщаємо рез. в R15

{or srl, r10, r8, z;}

{or srl, r10, r10, z;}

{or srl, r10, r10, z;}

{or srl, r10, r10, z;}

{or srl, r10, r10, z;}

{or nil, r10, z; oey; load rb;}

{or nil, rb, z; oey; ewl;}

{add rb, rb, r3;}

load\_op2

{cjp rdm, load\_op2; r; or r15, bus\_d, r0;}

{or nil, rb, z; oey; ewl;}

write\_rez

{cjp rdm, write\_rez; w; or nil, r15, z; oey;}

{}

{cjp nz, end\_cmd;}

nAutoInc

{}

{cjp nz, end\_cmd;}

nDisyunk

{}

{cjp nz, end\_cmd;}

end\_cmd

\Формування адреси наступної команди

{add r7, r7, z, nz;}

naddr

{cjp nz, start;}

stop

{}