

Національний технічний університет України  
«Київський політехнічний інститут»  
Кафедра обчислювальної техніки

## **Лабораторна робота №4**

з дисципліни «Архітектура комп'ютерів»

Виконав: студент 2 курсу  
ФІОТ гр. ІО-92  
Петрук В.О.

Київ 2011 р.

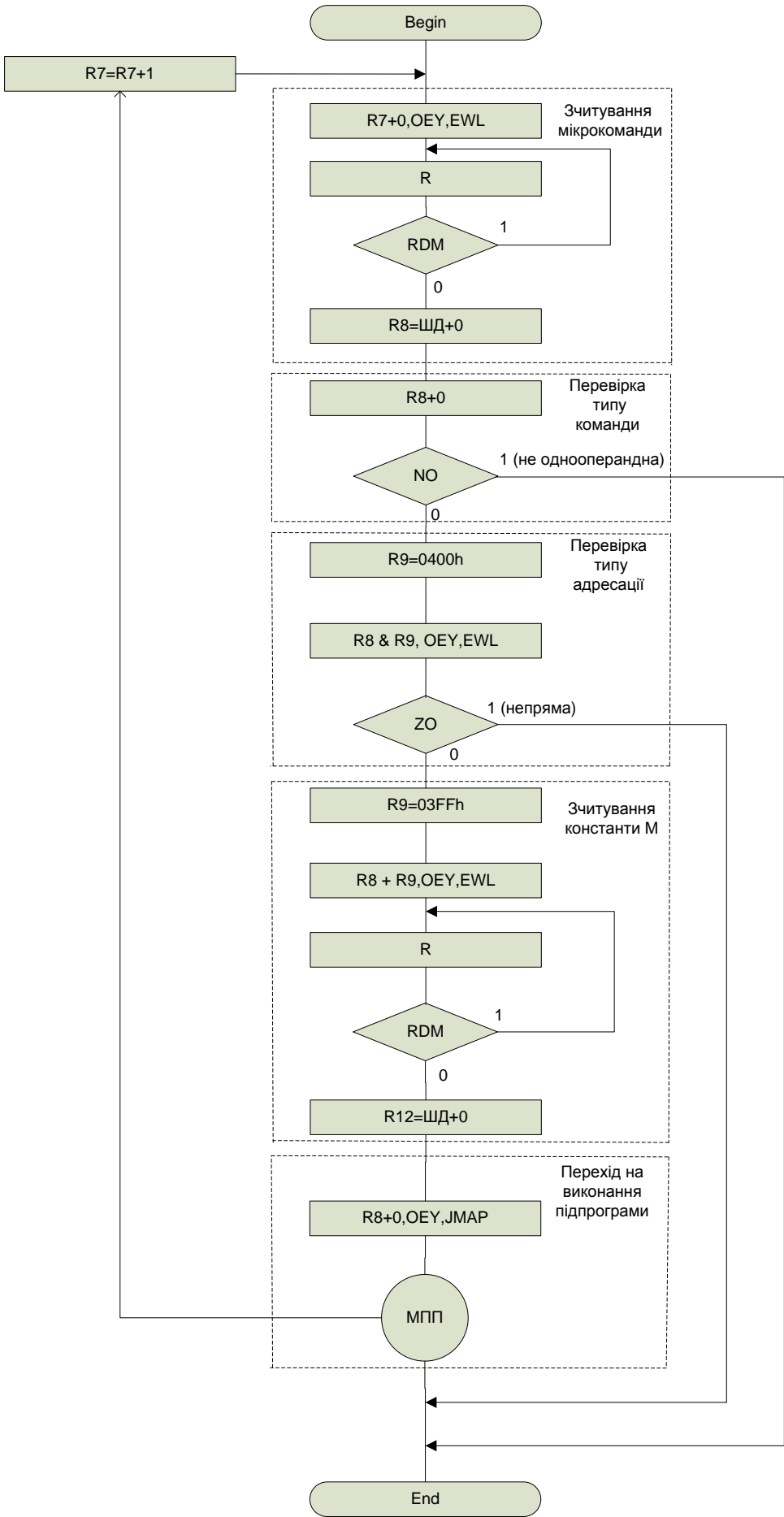
**Мета:** вивчити етапи виконання команд в ЕОМ. Навчитися розробляти мікроалгоритми вибірки, розпакування команд, виконання операцій і формування адреси наступної команди. Одержати навички розробки мікропрограм для ЕОМ з використанням мнемонічного мікроасемблера.

**Варіант №10010000000011**

Тип адресації – пряма

Код операції множення – 0011

**ФС-мікроалгоритм**



## Код програми на асемблері:

```
link l1: ct
link l2: rdm
link ewh:16
link m:z,z,z,z,z,z,z,z,z,14,13,12,11
\-----commands-----\
dw 0:1C50h
dw 1:1C51h
dw 2:1C52h
dw 3:8000h
\-----const M-----\
dw 50h:1
dw 51h:1
dw 52h:500

accept r14:0          \Yh,Rh
accept r15:500        \Yl,Rl
accept r7:0           \counter
\-----program begin-----\
org 11111%
NEXT {ADD NIL, R7, Z; OEY; EWL;}
{R; CJP RDM, CP; ADD R8, BUS_D, Z;} \reading of command
{ADD NIL, R8, Z; CJP NO, end;}      \N

{XOR R9, R9, R9;}                \ TA
{ADD R9, R9, 0400h;}
{AND NIL, R9, R8; CJP ZO, end;}

{XOR R9, R9, R9;}                \ r12 <- M
{ADD R9, R9, 03FFh;}
{AND NIL, R9, R8; OEY; EWL;}
{R; CJP RDM, CP; ADD R12, BUS_D, Z;}

{ADD nil,r8,z;oey;JMAP;}          \ jump to MPP
RETURN{ADD R7,R7,z,nz; cjp nz, NEXT;} \ R7 := R7+1

\-----begin mul-----\
org 0011%
    {xor r9,r9,r9;}
    {xor r10,r10,r10;}
    {or r14,z,r15;}
    {xor r15,r15,r15;}
    {load rm,z;}
    {or r11,z,r12;}
    {xor r11,r11,r14;load rn,flags;}
    {or r11,z,8000h;}
    {and nil,r12,r11;load rm,flags;cjp rm_z,plus_x;}
    {sub r12,z,r12,nz;}
plus_x {and nil,r14,r11;load rm,flags;cjp rm_z,plus_y;}
    {SUB R14,z,R14,nz;}
plus_y {or srl,r14,r14,z;}          \shift
    {or sr.9,r15,r15,z;}
begin  {or sll,r12,r12,z;load rm,flags;CJP NOT RM_N, notadd;}
    {add r10,r10,r15,z;LOAD RM,FLAGS;}
    {add r9,r9,r14,RM_C;}
notadd {or srl,r14,r14,z;}
    {or sr.9,r15,r15,z;}
    {or nil,r12,r12;load rm, flags;}
    {cjp not RM Z, begin;}
    {cjp not RN N, endMul;}
    {ADD r9,r9,r11,z;}
endMul {or r14,z,r9;}
    {or r15,z,r10;cjp nz,RETURN;}

end {}
```