

Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут»

**Лабораторна робота №4**

***з дисципліни «Комп'ютерна схемотехніка»***

**«Емуляція системи команд в обчислювальній системі з мікропрограм ним керуванням»**

Виконав студент III курсу

групи: КВ-11

ПІБ: Терентьєв Іван Дмитрович

Перевірив: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Київ 2024**

*Завдання для лабораторної роботи*

Реалізувати методом емуляції системи команд задану операцію за варіантом. Кожна операція потребує для своєї реалізації дві або три команди Асемблера, а кожній команді Асемблера має відповідати мікропрограма в ПМК. Перед безпосередній виконанням операції операнд(операнди), який має формат INT, необхідно перетворити в формат REAL. Кінцевий результат операції має бути подано в форматі REAL. Операнди A, B слід розмістити в регістрах віртуального процесора.

*Порядок виконання роботи*

За допомогою редактора набрати мікропрограму, виконати її та перевірити правильність роботи на контрольних прикладах.

***Варіант за номером залікової книжки 54 (***

*Таблиця 1.* Варіант

|  |  |  |  |
| --- | --- | --- | --- |
|  | Операція | Формат операндів | |
| A | B |
| 1 0 0 |  | INT | REAL |

***Розміщення послідовності команд Асемблера, які реалізують виконання заданої мікрооперації, в оперативній пам’яті***

dw 100h:  0c41h  \ CONV

dw 101h:   641h  \ SQRT

dw 102h: 0a242h  \ MULF

dw 103h:  2242h  \ ADDF

dw 104h:   0ffh  \ FINISH

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Адреса в ОП** | **КОП** | **REGx** | **REGy** | **Команда асемблера** |
| 100h | 0C41h | 1 | - | CONV REGa |
| 101h | 0641h | 2 | - | SQRT REGa |
| 102h | A242h | - | 2 | MULF REGb, REGb |
| 103h | 2242h | 1 | 2 | ADDF |
| 104h | 00FFh | - | - | FINISH |

***Структура мікропрограми емуляції заданої операції***

ROZM

S1 SUBROUT1

{cjp rn\_z, singeop;}

singleop{…}

start { jmap; or nil, r2, z, oey;}

S2 SUBROUT2

{cjp nz, S1;}

ORG 0C20h

\ CONV

MCCONV

ORG 0220h

\ ADDF

MCADDF

ORG 0620h

\ SQRT

MCSQRT

ORG 0A20h

\ MULF

MCMULF

ORG 0FF0h

MCFINISH

END{}

***Код***

macro mov reg1,reg2:{or reg1,z,reg2;}

\Rozm

accept r0: 0bh

link l1: ct

link l2: rdm

link ewh: 16

link m: 7,6,5,4,3,2,1,0,z,z,z,z

dw 0bh: 0000h \PSW

dw 0h:100h \ PC:=100h

dw 100h:  0c21h  \ CONV

dw 101h:   621h  \ SQRT

dw 102h: 0a222h  \ MULF

dw 103h:  2212h  \ ADDF

dw 104h:   0ffh  \ FINISH

dw 01h: 0051h  \ Rg1 - A

dw 02h: 0C180h \ Rg2 - B

accept rdm\_delay: 2

\SUBROUT1

S1 {}

{xor r1,r1; oey; ewh;}

{or nil,r1,z; oey; ewl;}

{xor r6,r6;}

P1 {cjp rdm, P1; r; or r6,bus\_d, z;}

{or nil, r6, z; oey; ewl;}

{xor r4,r4;}

P2 {cjp rdm, P2; r; or r4,bus\_d,z;}

{xor r2,r2;}

{or r2,r4,z;}

{and nil,r2,0f000h; load rn, flags;}

{cjp rn\_z,singleop;}

{push nz, 7;}

{rfct; or sr.0, r2,r2,z;}

{xor r5,r5;}

{or r5,r4;}

{and r4,000fh;}

{and r5,00f0h;}

{push nz,3;}

{rfct; or sr.0,r5,z;}

{or nil,r4,z; oey; ewl;}

{xor r10,r10;}

P3 {cjp rdm,P3; r; or r10,bus\_d,z;}

{or nil,r5,z; oey; ewl;}

{xor r11,r11; }

P4 {cjp rdm,P4; r; or r11,bus\_d,z;}

{cjp nz,start; }

singleop {}

{and nil,r2,00f00h; load rn, flags;}

{cjp rn\_z, start;}

{push nz,3; }

{rfct; or sr.0, r2,r2,z;}

{xor r5,r5;}

{or r5,r4,z;}

{and r5,000fh;}

{xor r11,r11;}

{or nil,r5,z; oey; ewl;}

P44 {cjp rdm,P44; r; or r11,bus\_d,z;}

start { jmap; or nil,r2,z; oey; }

\SUBROUT2

S2 {or nil,r0,z; oey; ewl;}

{xor r12,r12;}

P5 {cjp rdm,P5; r; or r12,bus\_d,z;}

{cjp rn\_v,D1;}

{and r12,0fbffh;}

{cjp nz,J1;}

D1 {or r12,0400h;}

J1 {cjp rn\_c,D2;}

{and r12,0fffeh;}

{cjp nz,J2;}

D2 {or r12,0001h;}

J2 {cjp rn\_n,D3;}

{and r12,0feffh;}

{cjp nz,J3;}

D3 {or r12, 0100h;}

J3 {load rm,flags; and nil,r15,00ffh;}

{cjp rm\_z,D4;}

{and r12,0ffbfh;}

{cjp nz,J4;}

D4 {or r12,0040h;}

J4 {or nil,r0,z; oey; ewl;}

RR1{cjp rdm,RR1; w; or nil,r12,z; oey;}

{or nil,r5,z; oey; ewl;}

RR2{cjp rdm,RR2; w; or nil,r15,z; oey;}

{add r6,1;}

{xor r1,r1;}

{or nil,r1,z; oey; ewl;}

RR3{cjp rdm,RR3; w; or nil,r6,z; oey;}

{cjp nz,S1;}

\-----------------------------

ORG 0c20h

\ CONV

{mov r14,r0;}

{mov r13,r5;}

{mov r9,r6;}

{mov r0,r11;}

{xor r1,r1;}

{xor r2,r2;}

{xor r3,r3;}

{xor r4,r4;}

{mov r1,r0;}

{and nil,r1,8000h; load rn,flags;}

{cjp rn\_z,c1;}

{mov r2,8000h;}

{xor r1,0ffffh;}

{add r1,1;}

c1 {}

{mov r3,0f00h;}

{mov r4,4000h;}

c2 {}

{and nil,r1,r4; load rn,flags;}

{cjp not rn\_z,c3;}

{or srl,r4,r4;}

{sub r3,r3,0100h,nz; load rn,flags;}

{cjp rn\_z,c3;}

{cjp nz,c2;}

c3 {}

{sub r4,0800h,r3,nz; load rn,flags;}

{cjp rn\_n,c4;}

c5 {}

{and nil,r4,0ffffh; load rn,flags;}

{cjp rn\_z,c6;}

{or sll,r1,r1;}

{sub r4,r4,0100h,nz;}

{cjp nz,c5;}

c4 {sub r4,r3,0800h,nz;}

cf {}

{and nil,r4,0ffffh; load rn,flags;}

{cjp rn\_z,c6;}

{or srl,r1,r1;}

{sub r4,r4,0100h,nz;}

{cjp nz,cf;}

c6 {}

{add r3,4000h;}

{or r2,r3;}

{or r2,r1;}

{mov r15,r2;}

{mov r0,r14;}

{mov r5,r13;}

{mov r6,r9;}

{cjp nz,S2;}

ORG 0220h

\ ADDF

{mov r14,r0;}

{mov r13,r5;}

{mov r9,r6;}

{mov r0,r11;}

{mov r1,r10;}

{xor r2,r2;}

{xor r4,r4;}

{xor r5,r5;}

{xor r6,r6;}

{xor r7,r7;}

{xor r8,r8;}

a1 {}

{and r4,7f00h,r0;}

{and r5,00ffh,r0;}

{and r6,7f00h,r1;}

{and r7,00ffh,r1;}

al1 {}

{sub nil,r4,r6,nz; load rn,flags;}

{cjp rn\_z,a2;}

{cjp rn\_n,av;}

{add r6,r6,100h;}

{or sra,r7,z,r7;}

{cjp nz,al1;}

av {}

{add r4,r4,100h;}

{or sra,r5,z,r5;}

{cjp nz,al1;}

a2 {}

{and nil,r0,8000h; load rn,flags;}

{cjp rn\_z,al2;}

{xor r5,0ffffh;}

{add r5,0,nz;}

al2 {}

{and nil,r1,8000h; load rn,flags;}

{cjp rn\_z,al3;}

{xor r7,0ffffh;}

{add r7,0,nz;}

al3 {}

{and r2,0000h;}

{mov r8,r5;}

{add r8,r7;}

{and nil,r8,0200h; load rn,flags;}

{cjp rn\_z,a3;}

{xor r8,0ffffh;}

{add r8,0,nz;}

{or  r2,8000h;}

a3 {}

{and nil,r8,0100h; load rn,flags;}

{cjp rn\_z,al4;}

{add r4,100h;}

{or sra,r8,r8;}

{cjp nz,al5;}

al4 {}

{and nil,r8,0080h; load rn,flags;}

{cjp rn\_z,al6;}

{cjp nz,al5;}

al6 {}

{or sla,r8,r8;}

{sub r4,r4,100h,nz;}

{cjp nz,al4;}

al5 {}

{or r2,r4;}

{or r2,r8;}

{and nil,r4,8000h; load rn,flags;}

{cjp rn\_z,afin;}

{add r4,0ffffh; load rn,flags;}

afin {}

{mov r15,r2;}

{mov r0,r14;}

{mov r5,r13;}

{mov r6,r9;}

{cjp nz,S2;}

ORG 0620h

\ SQRT

{mov r14,r0;}

{mov r13,r5;}

{mov r10,r6;}

{mov r0,r11;}

{and nil,r0,8000h; load rn,flags;}

{cjp not rn\_z,sneg;}

{and r1,0ff00h,r0;}

{and r2,00ffh,r0;}

{and nil,r1,0100h; load rn,flags;}

{cjp rn\_z,sparity;}

{push nz,4;}

{or sll,r2,r2;}

{rfct;}

{add r1,0100h;}

{cjp nz,sbegin;}

sparity {}

{push nz,5;}

{or sll,r2,r2;}

{rfct;}

sbegin  {}

{mov r3,r2;}

{mov r4,0h;}

{mov r5,0h;}

{mov r6,0h;}

{mov r7,0c000h;}

smain {}

{or sll,r3,r3;}

{or sra,r7,r7;}

{xor r4,r4;}

{add r4,r3;}

{sub r4,r4,r5,nz;}

{sub r4,r4,r5,nz;}

{add r4,r7;}

{and nil,r4,8000h; load rn,flags;}

{cjp rn\_z,snol;}

{mov r8,0dfffh;}

{xor r9,r9;}

ss1  {}

{sub nil,r6,r9,nz; load rn,flags;}

{cjp rn\_z,ss2;}

{add r9,1;}

{or sra,r8,r8;}

{cjp nz,ss1;}

ss2 {}

{and r5,r8;} \PZ(Lch):=0

{cjp nz,sich;}

snol  {}

{mov r3,r4;}

{mov r8,2000h;}

{xor r9,r9;}

ss3 {}

{sub nil,r6,r9,nz; load rn,flags;}

{cjp rn\_z,ss4;}

{add r9,1;}

{or sra,r8,r8;}

{cjp nz,ss3;}

ss4  {}

{or r5,r8;}

{cjp nz,sich;}

sich {}

{add r6,1;}

{sub nil,r6,8h,nz; load rn,flags;}

{cjp rn\_z,sfin;}

{cjp nz,smain;}

sfin  {}

{push nz,5;}

{or srl,r5,r5;}

{rfct;}

{sub r3,r1,4000h,nz; load rn,flags;}

{cjp rn\_n, shneg;}

{or srl,r3,r3;}

{and r3,0ff00h;}

{add r1,r3,4000h;}

{cjp nz,sres;}

shneg {}

{or sll,r3,r3;}

{and r3,0ff00h;}

{sub r1,4000h,r3,nz;}

sres {}

{or r5,r1;}

{cjp nz,sanw;}

sneg {}

{mov r5,4000h;}

{or r1,0ffffh;}

{add r1,0ffffh; load rn,flags;}

sanw {}

{mov r15,r5;}

{mov r0,r14;}

{mov r5,r13;}

{mov r6,r10;}

{cjp nz,S2; }

ORG 0a20h

\ MULF

{mov r14,r0;}

{mov r13,r5;}

{mov r9,r6;}

{mov r0,r11;}

{mov r1,r10;}

{and r3,r0,8000h;}

{and r4,r1,8000h;}

{xor r3,r4;}

{mov r2,r3;}

{and r3,r0,7f00h;}

{and r4,r1,7f00h;}

{sub r3,r3,4000h,nz;}

{sub r4,r4,4000h,nz;}

{add r3,r3,r4;}

{add r3,r3,4000h;}

{and nil,r3,8000h; load rn,flags;}

{cjp not rn\_z,overfl;}

{or r2,r3;}

{xor r3,r3;}

{xor r4,r4;}

{and r0,00ffh;}

{push nz,7;}

{or sll,r0,r0;}

{rfct;}

{and r1,00ffh;}

{push nz,7;}

{or sll,r1,r1;}

{rfct;}

{and nil,r0,0ffffh; load rn,flags;}

{cjp rn\_z,mnol;}

{and nil,r1,0ffffh; load rn,flags;}

{cjp rn\_z,mnol;}

mmain {}

{and nil,r0,0001h; load rn,flags;}

{cjp rn\_z,m1;}

{add r3,r1;} \PZ:=PZ+PY;

m1 {}

{or srl,r3,r3;}

{or srl,r0,r0;}

{add r4,1;}

{sub nil,r4,0010h,nz; load rn,flags;}

{cjp rn\_z,mfin;}

{cjp nz,mmain;}

mnol {}

{mov r2,4000h;}

{cjp nz,ENDING;}

overfl {}

{mov r2,4000h;}

{or r3,0ffffh;}

{add r3,0ffffh; load rn,flags;}

{cjp nz,ENDING;}

mfin {}

{push nz,7;}

{or srl,r3,r3;}

{rfct;}

{and nil,r3,0080h; load rn,flags;}

{cjp not rn\_z,mok;}

m2 {}

{and nil,r3,0080h; load rn,flags;}

{cjp not rn\_z,mok;}

{or sll,r3,r3;}

{sub r2,r2,0100h,nz;}

{cjp nz,m2;}

mok {or r2,r3;}

ENDING  {}

{mov r15,r2;}

{mov r0,r14;}

{mov r5,r13;}

{mov r6,r9;}

{cjp nz,S2;}

ORG 0ff0h

\ FINISH

{cjp nz,END;}

END {}



Рис.1 – Виконання при (44A0)