

Основи рачунарске технике 2

Испит – Л4

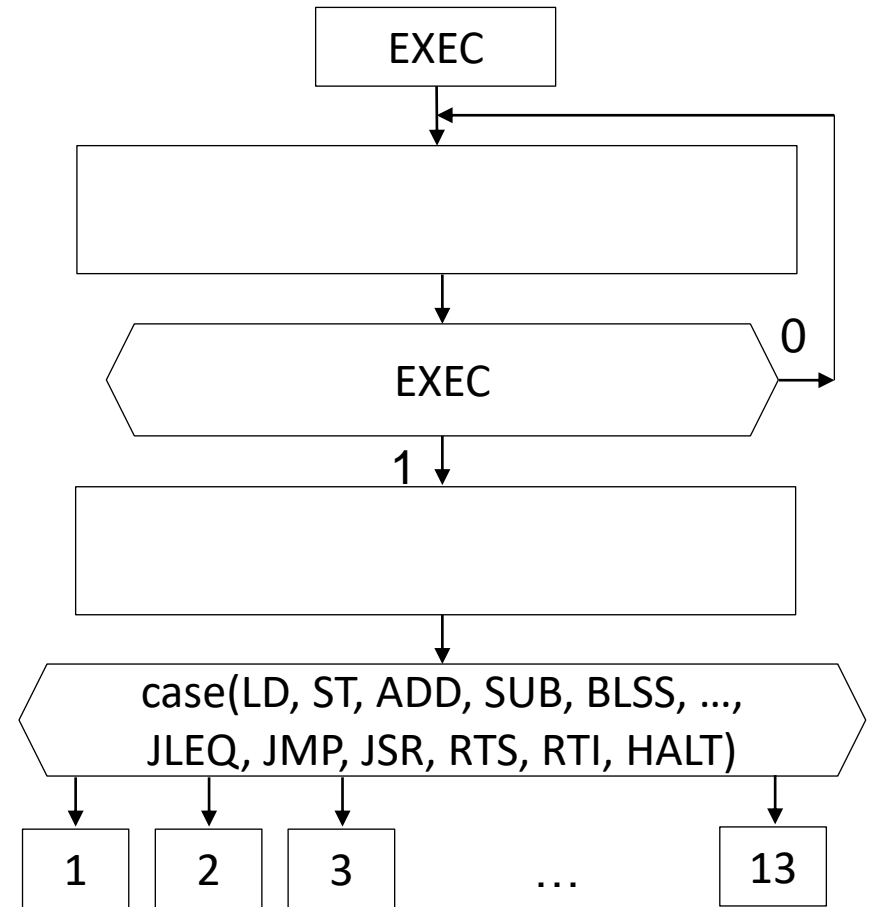
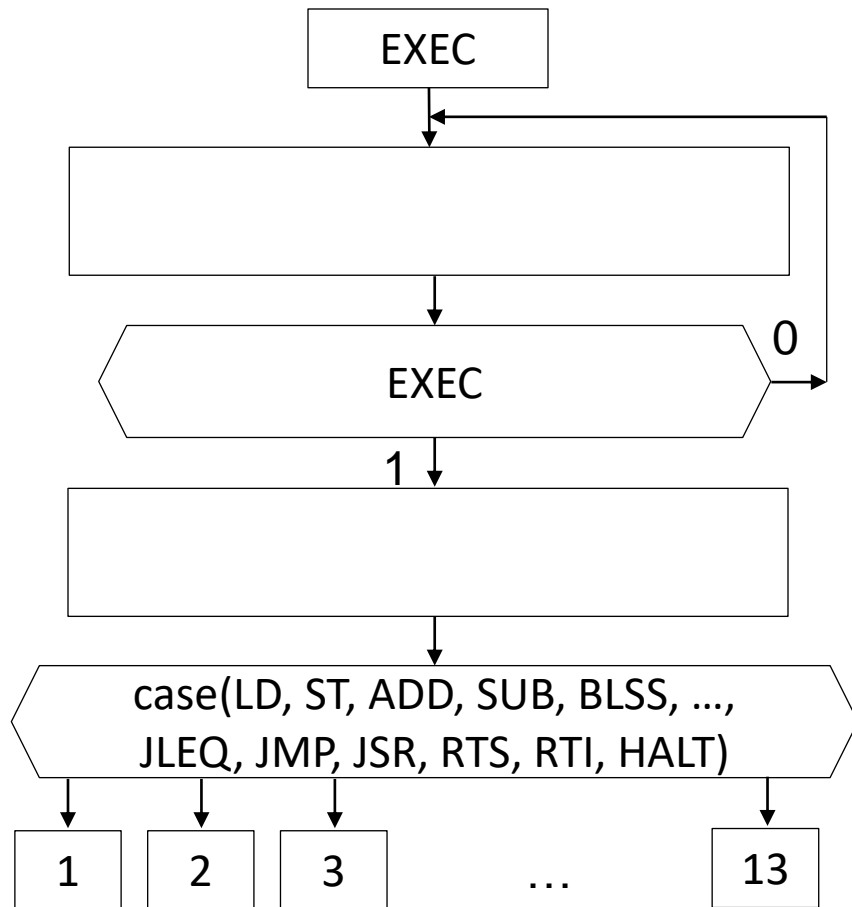
Основи рачунарске технике 2

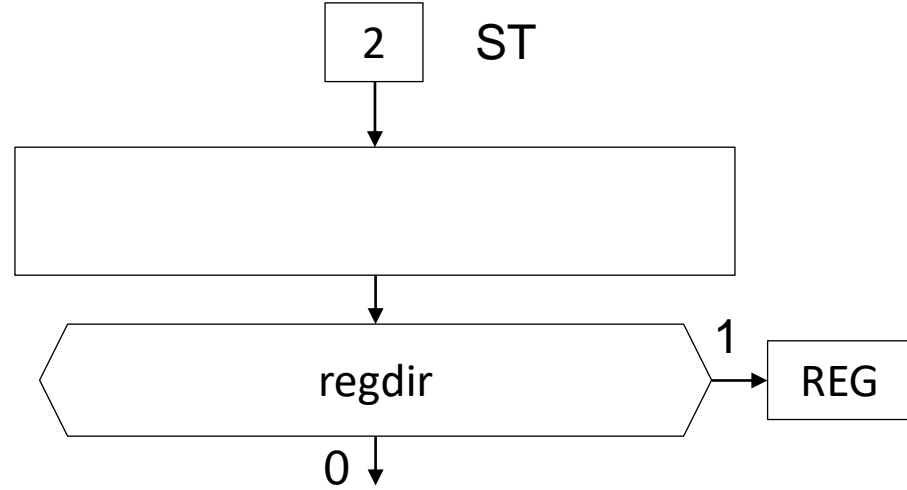
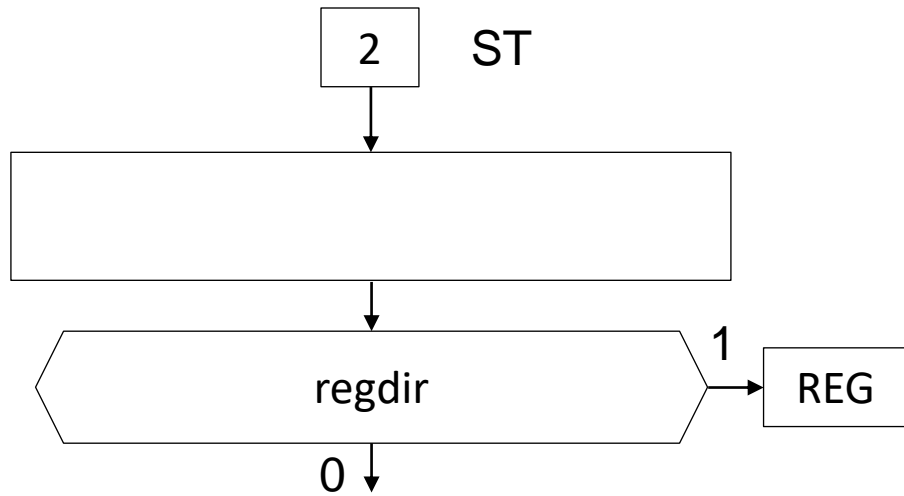
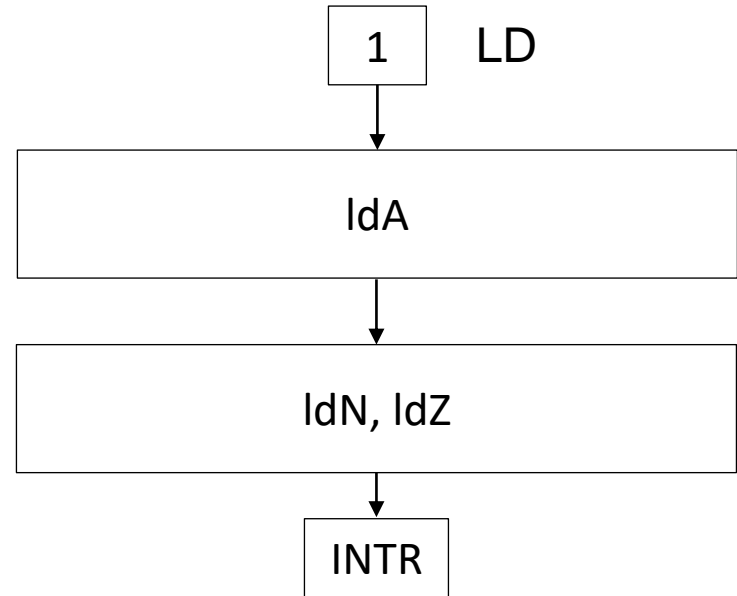
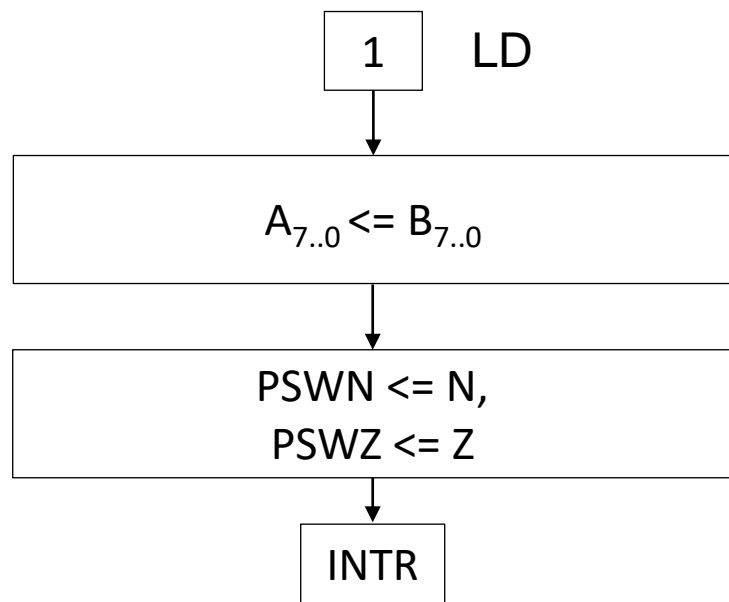
EXEC блок

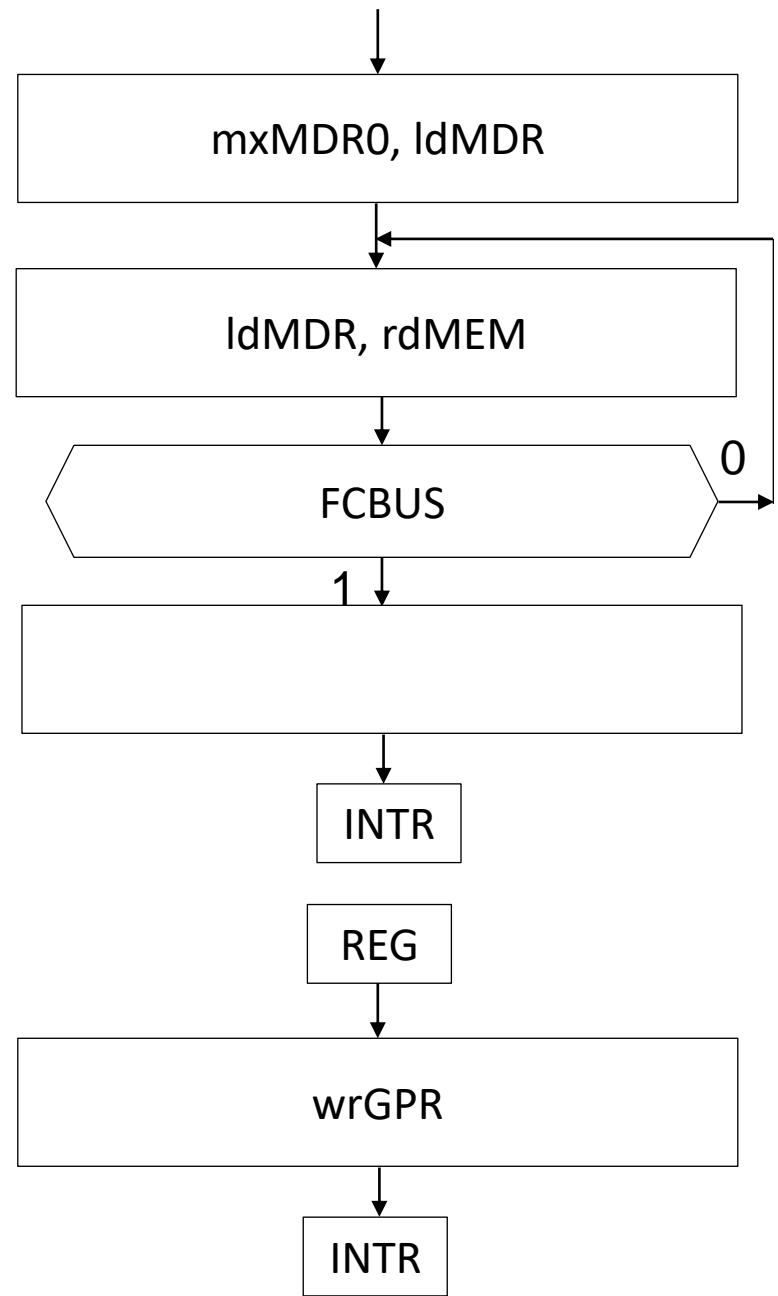
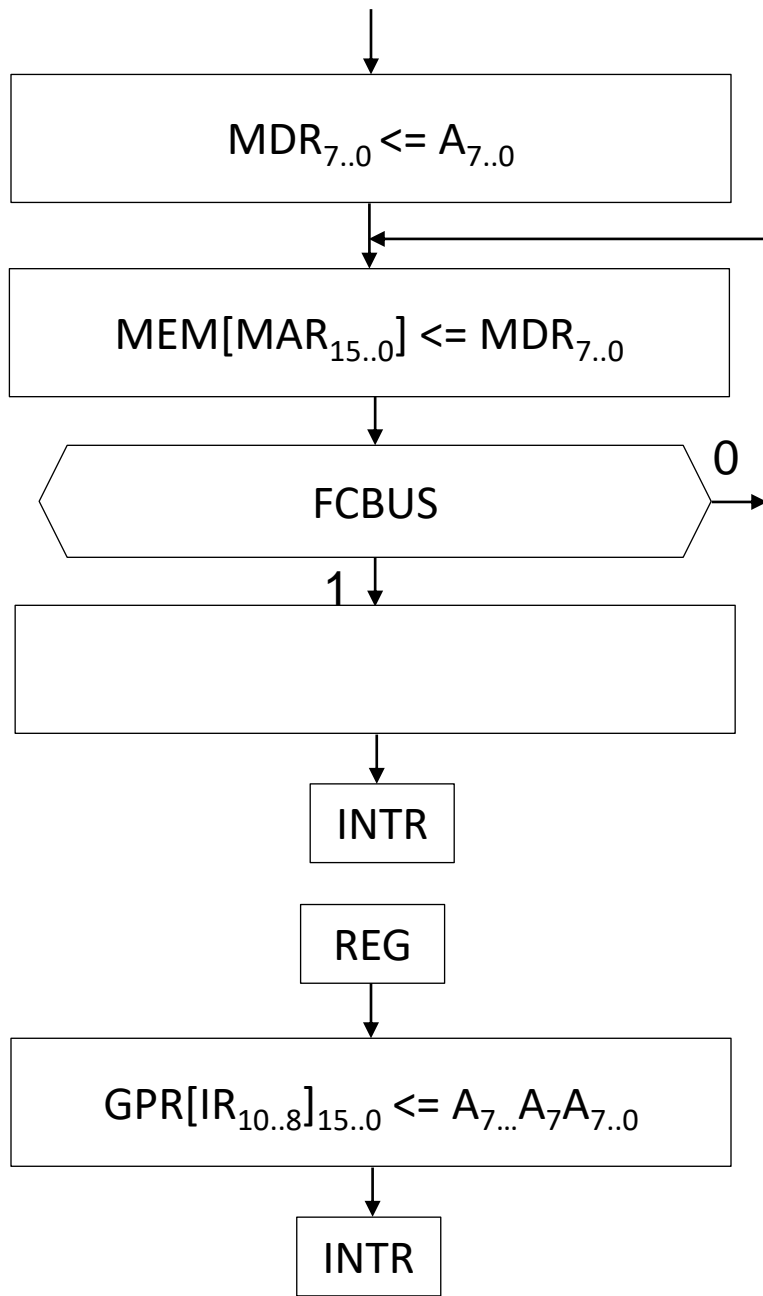
Инструкција	IR _{23..19}	Инструкција	IR _{23..16}	IR _{15..8}	IR _{7..0}	Дужина
LD	0011 0b	JSR	0010 0001b	млађи бајт	старији бајт	3B
ST	0011 1b	JMP	0010 0000b	млађи бајт	старији бајт	3B
ADD	0100 0b	JLEQ	0001 0010b	млађи бајт	старији бајт	3B
SUB	0100 1b	BGREU	0001 0001b	померај	/	2B
		BLSS	0001 0000b	померај	/	2B
		POPGPR	0000 0100b	/	/	1B
		PUSHGPR	0000 0011b	/	/	1B
		RTI	0000 0010b	/	/	1B
		RTS	0000 0001b	/	/	1B
		HALT	0000 0000b	/	/	1B

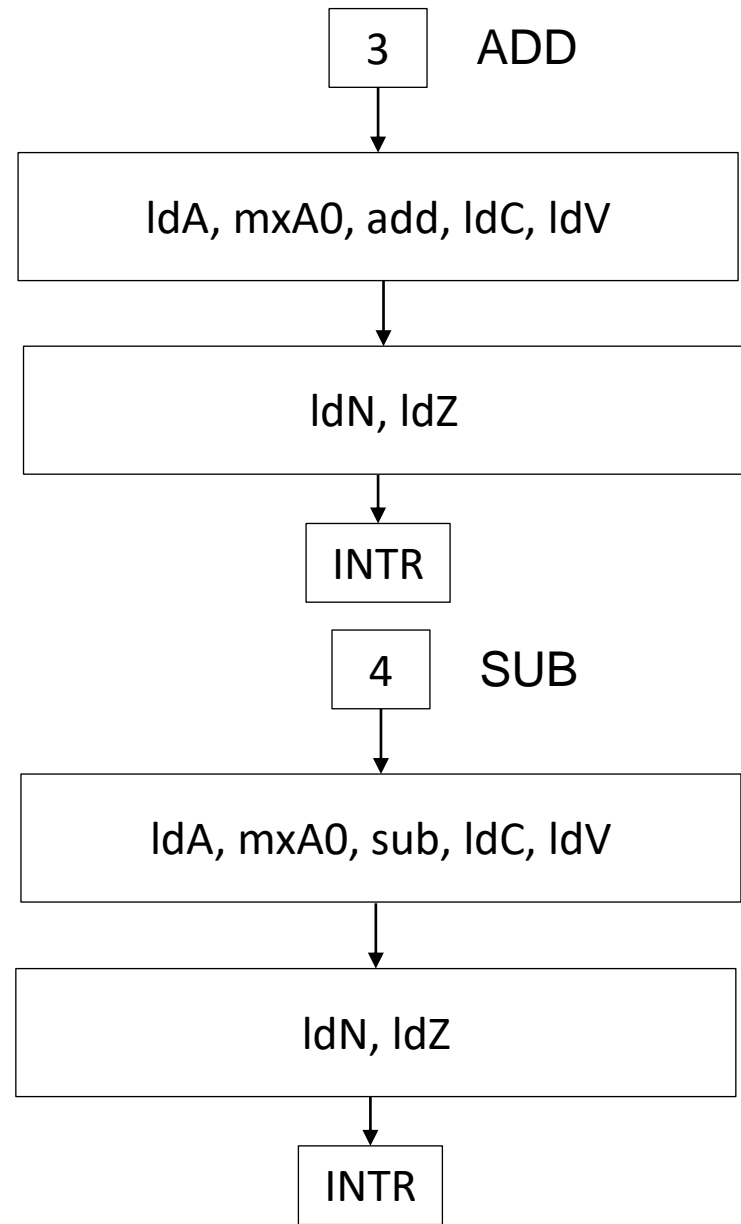
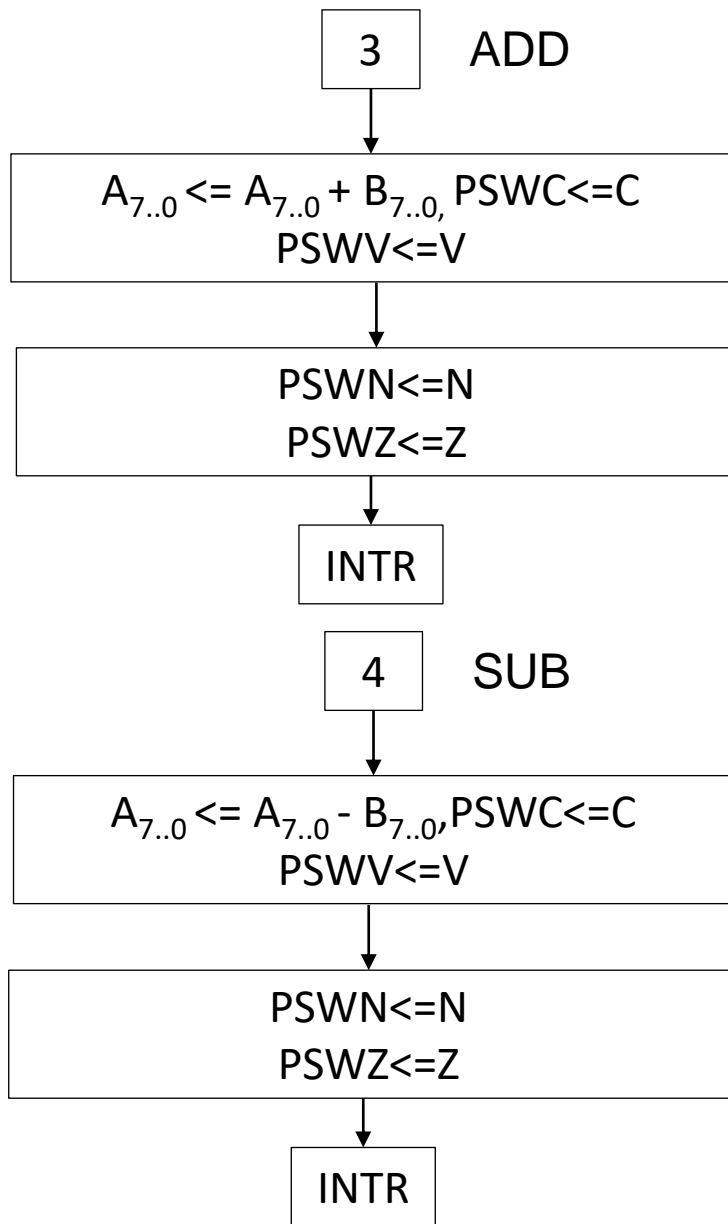
Адресирања	IR _{18..16}	IR _{15..8}	IR _{7..0}	Дужина
regindpom	011b	PPPP PRRRb	/	2B
regdir	010b	XXXX XRRRb	/	2B
memdir	001b	млађи бајт	старији бајт	3B
immed	000b	податак	/	2B

	Нижа адреса	Виша адреса
Адреса	млађи	старији



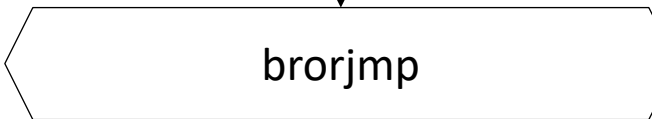
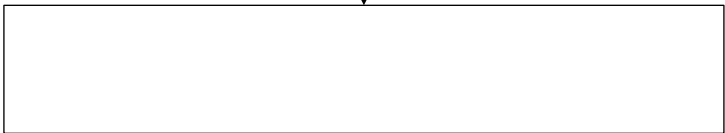






5

BLSS, BGREU



0

INTR

1

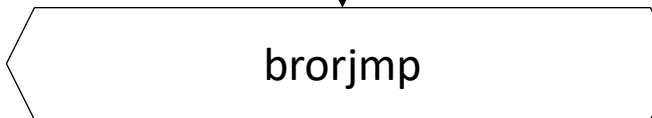
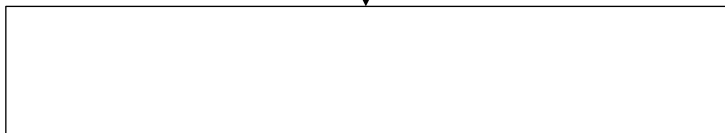


$PC_{15..0} \leq PC_{15..0} + IR_{15..IR_{15}} IR_{15..8}$

INTR

6

JLEQ



0

INTR

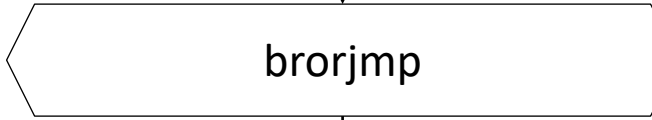
1



7

5

BLSS, BGREU



0

INTR

1

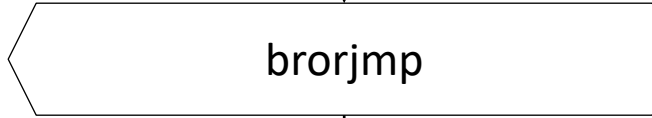


mxPC0, IdPC

INTR

6

JLEQ



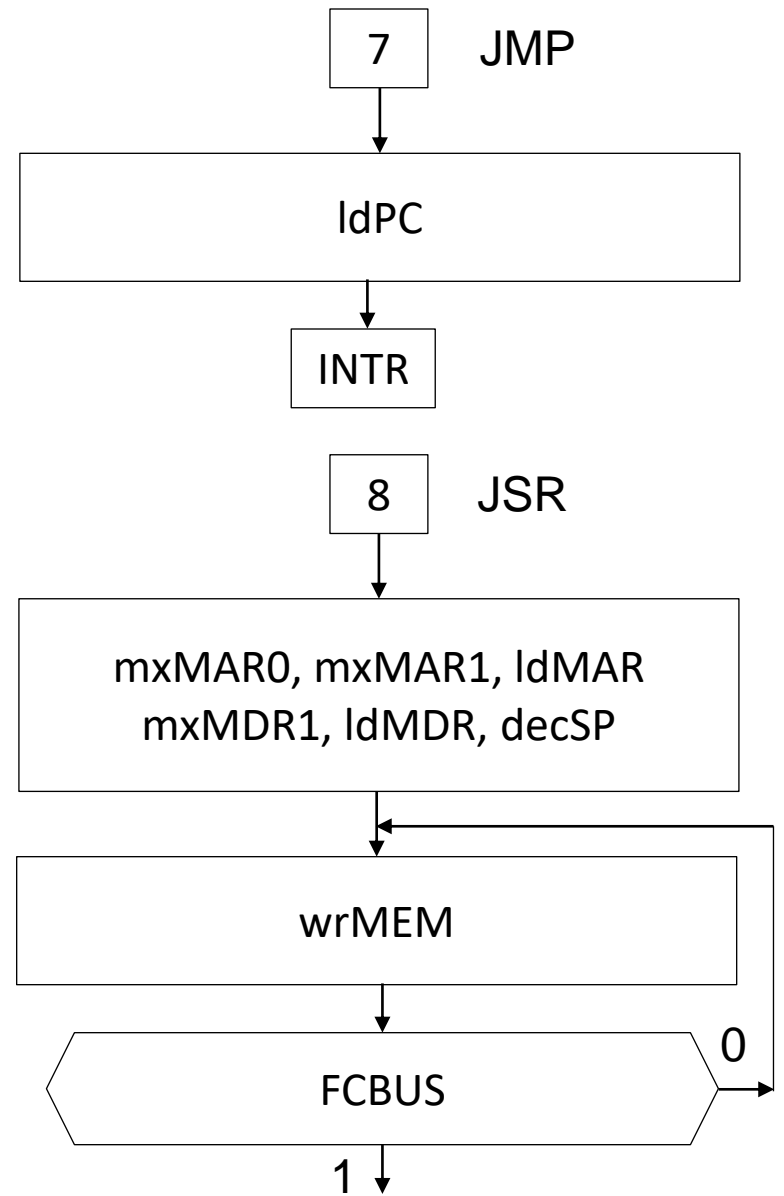
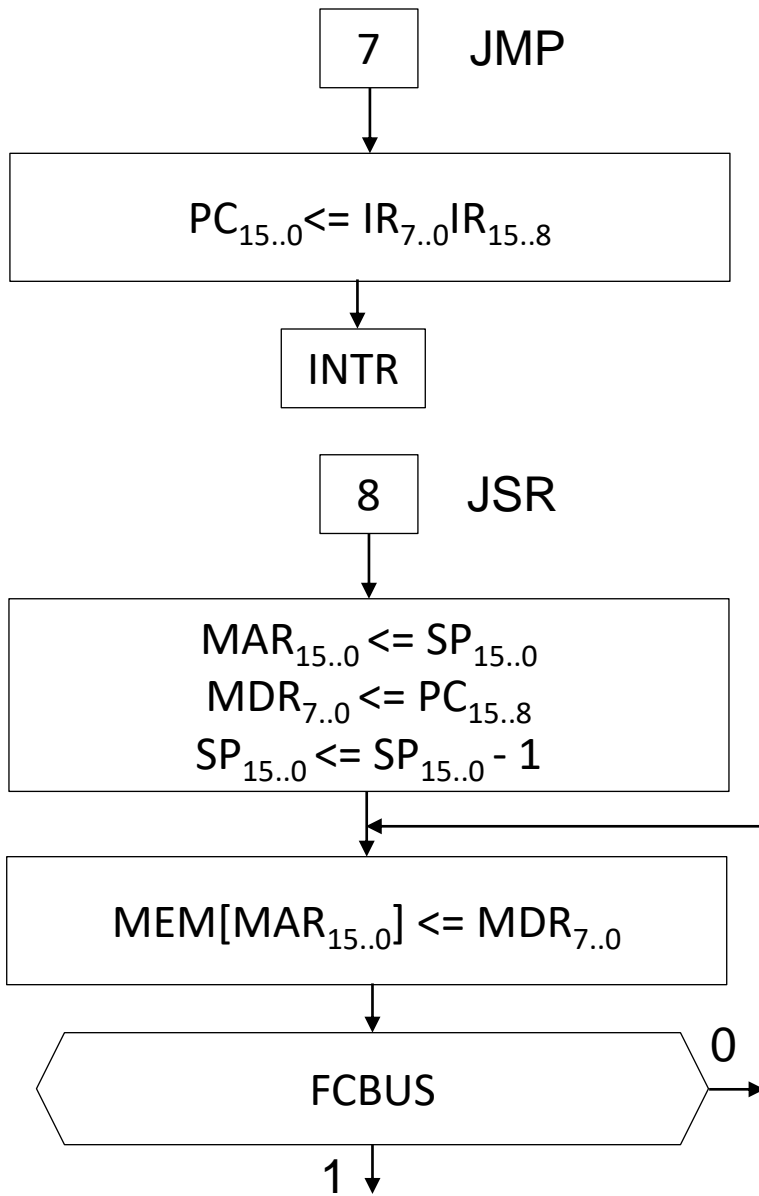
0

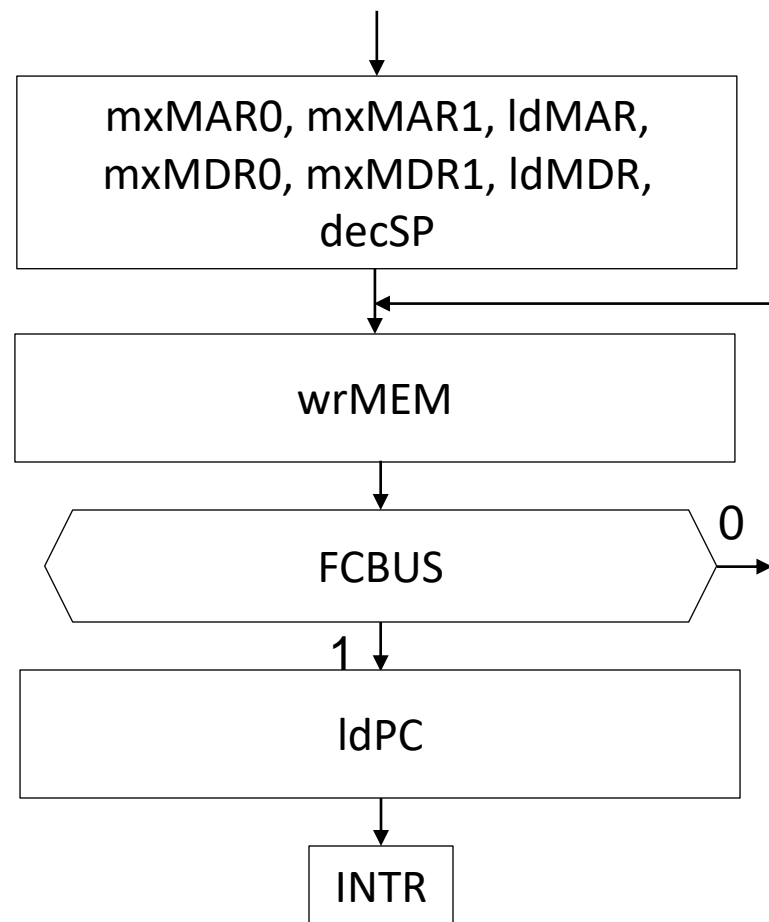
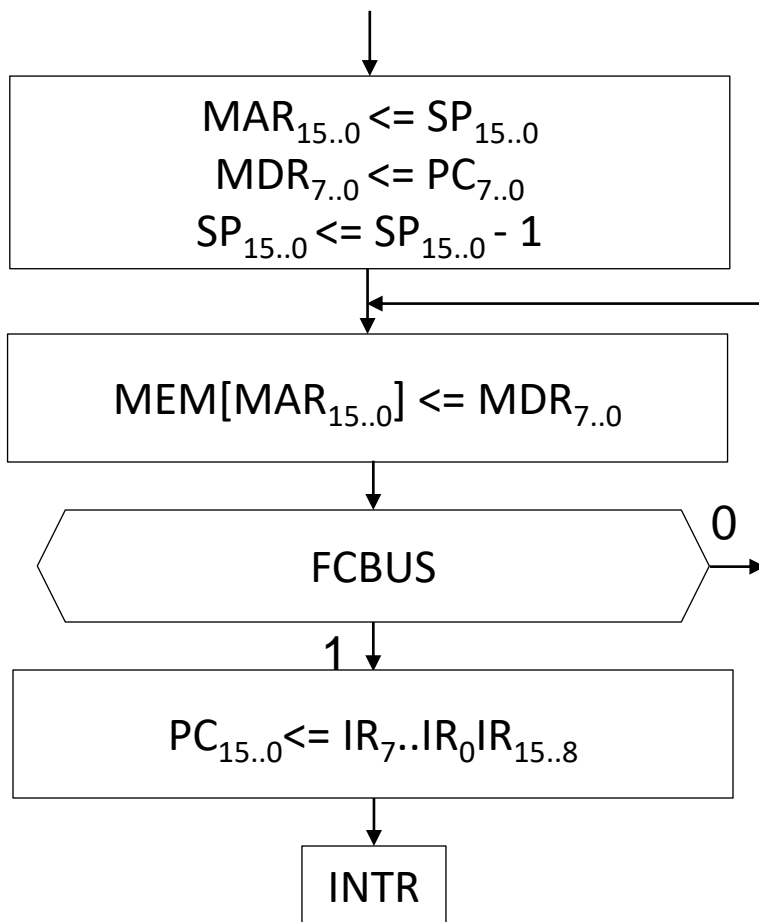
INTR

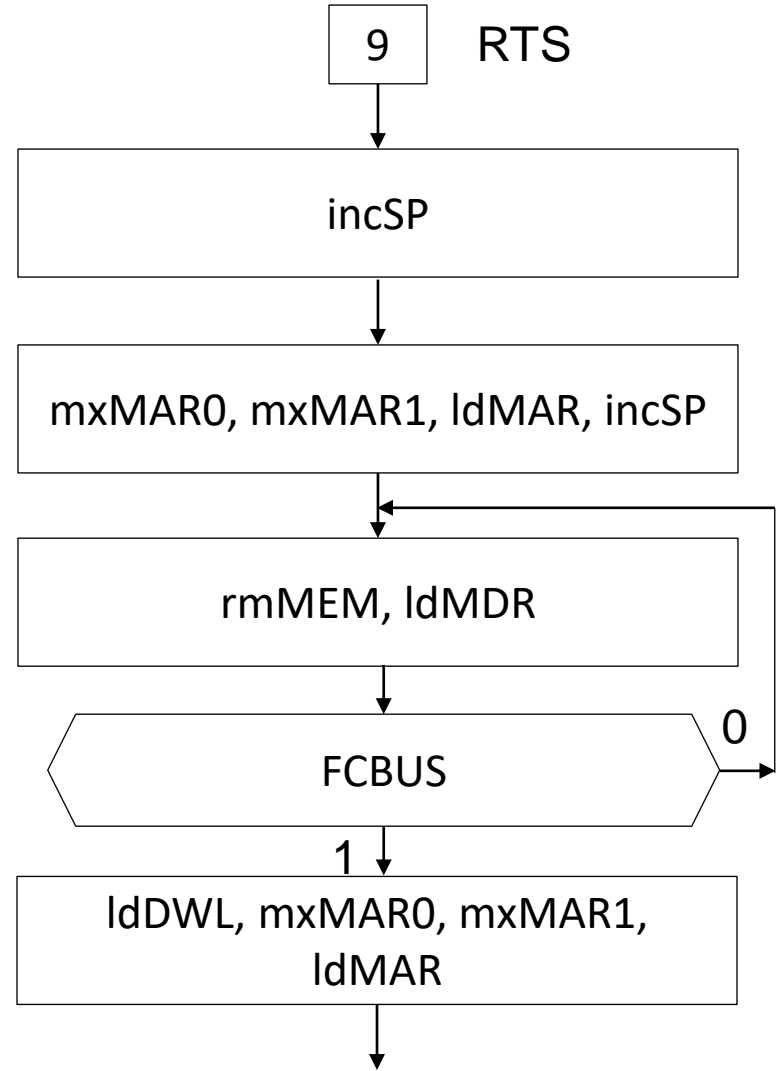
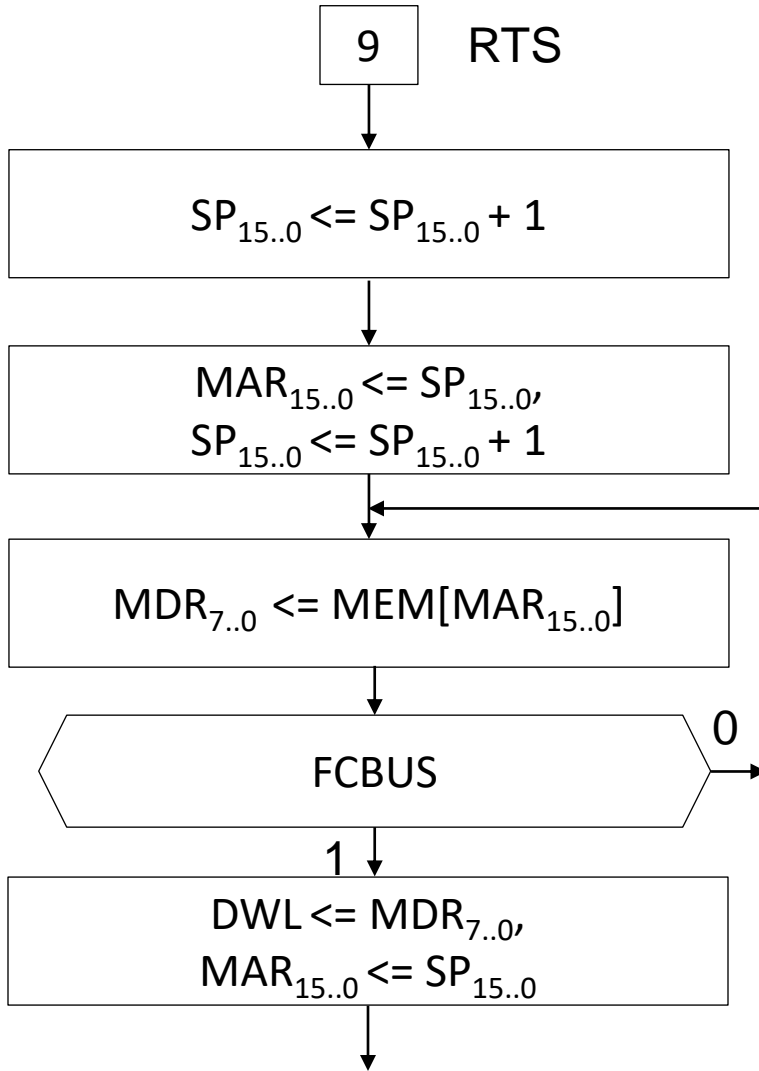
1

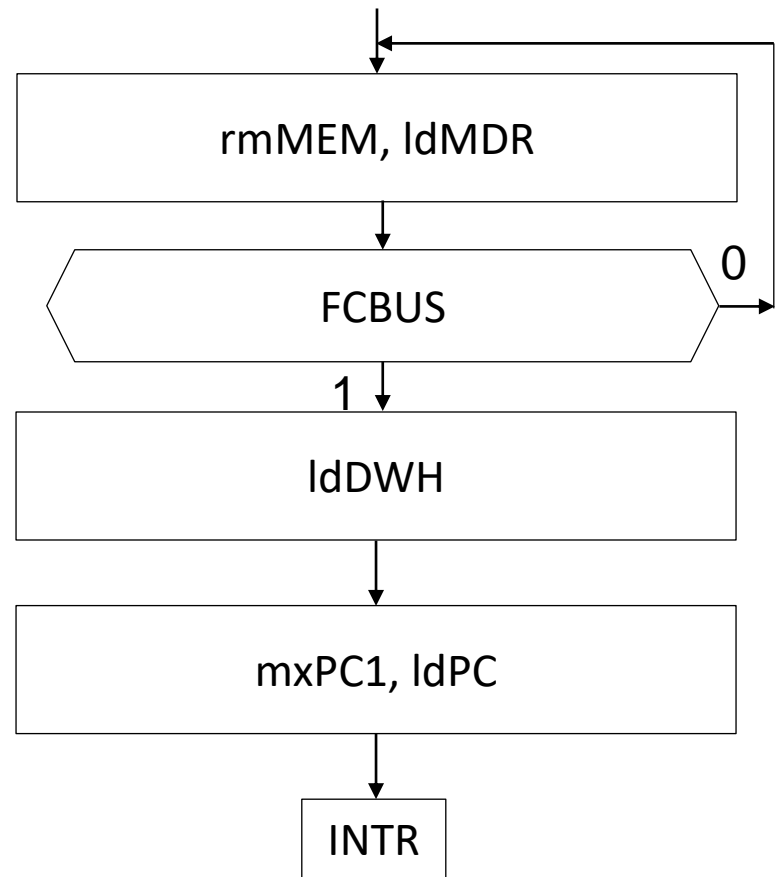
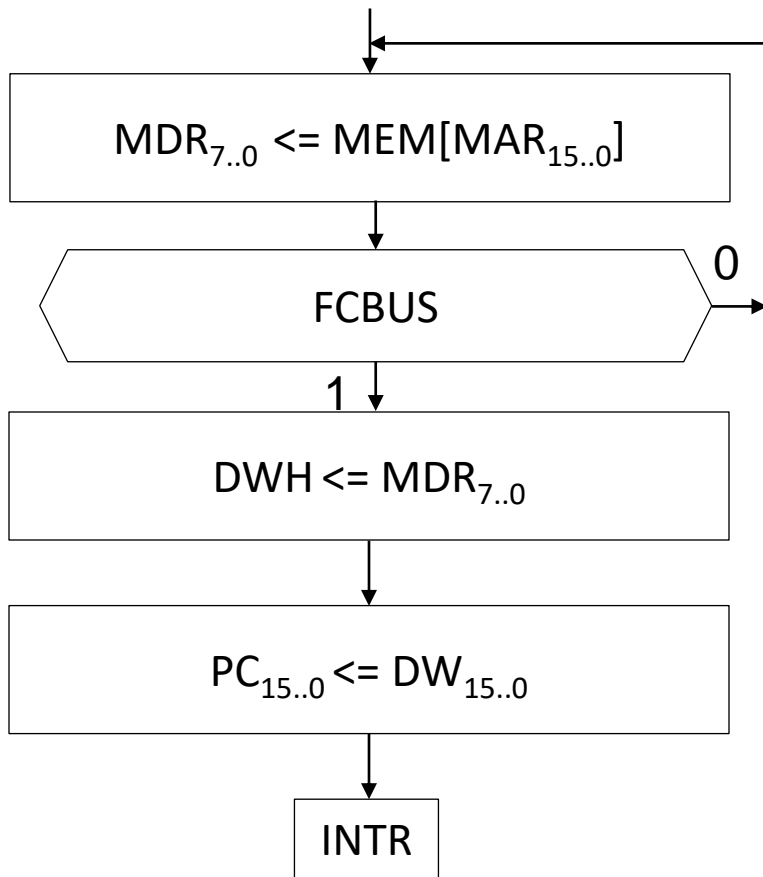


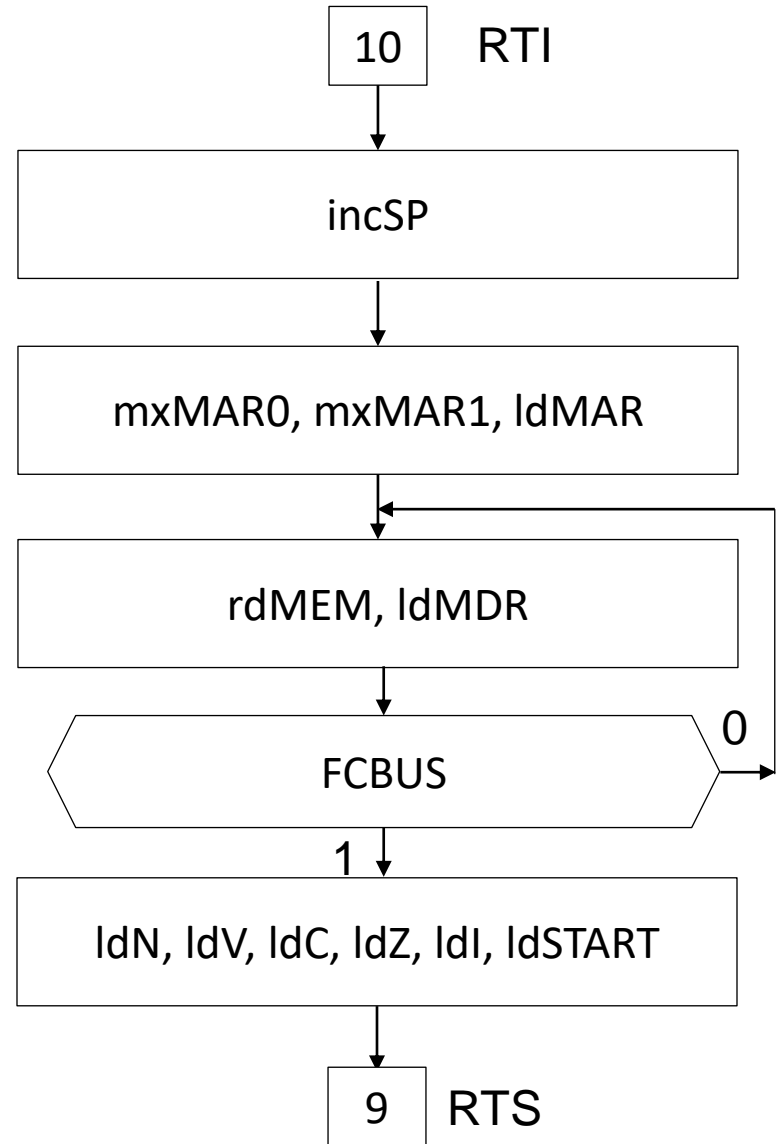
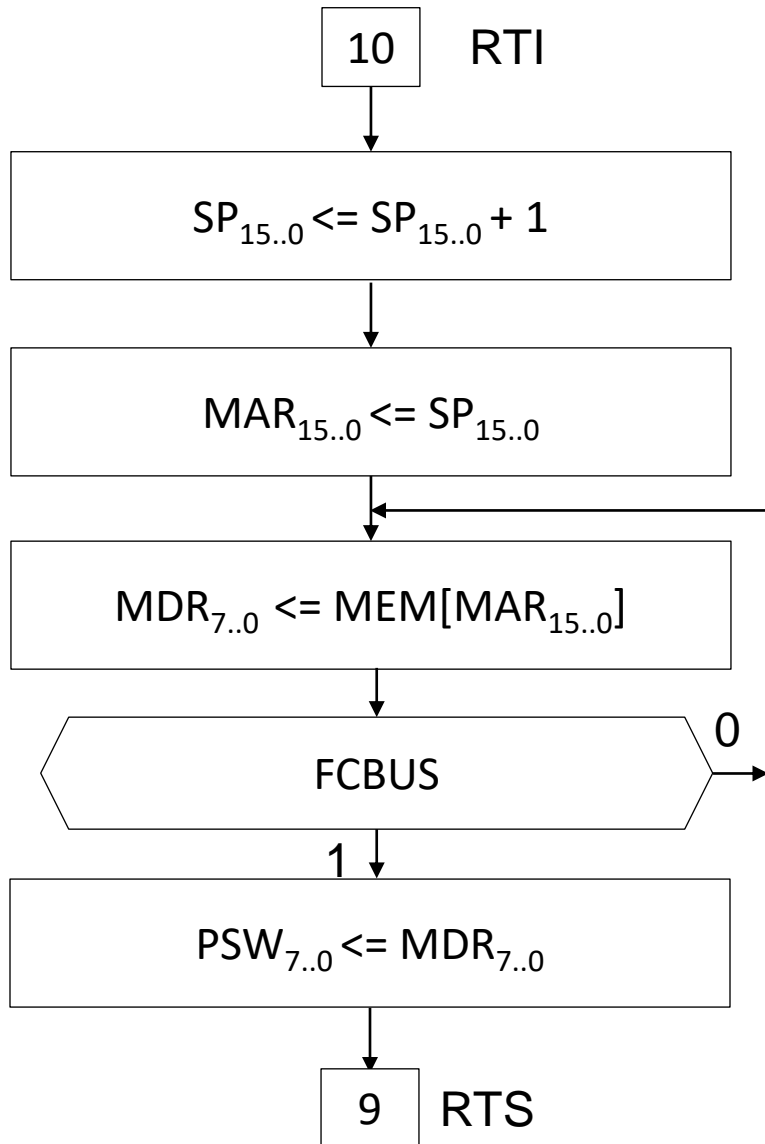
7

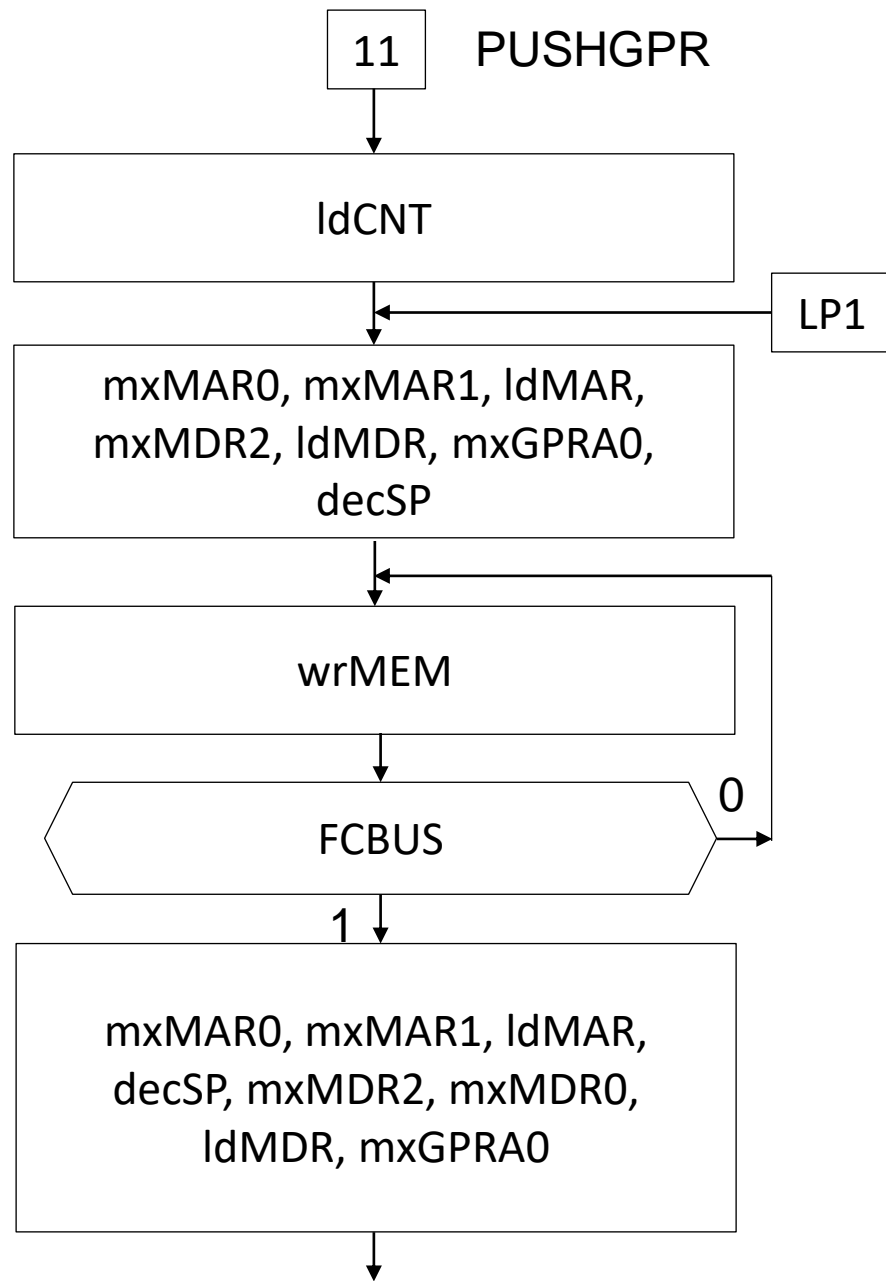
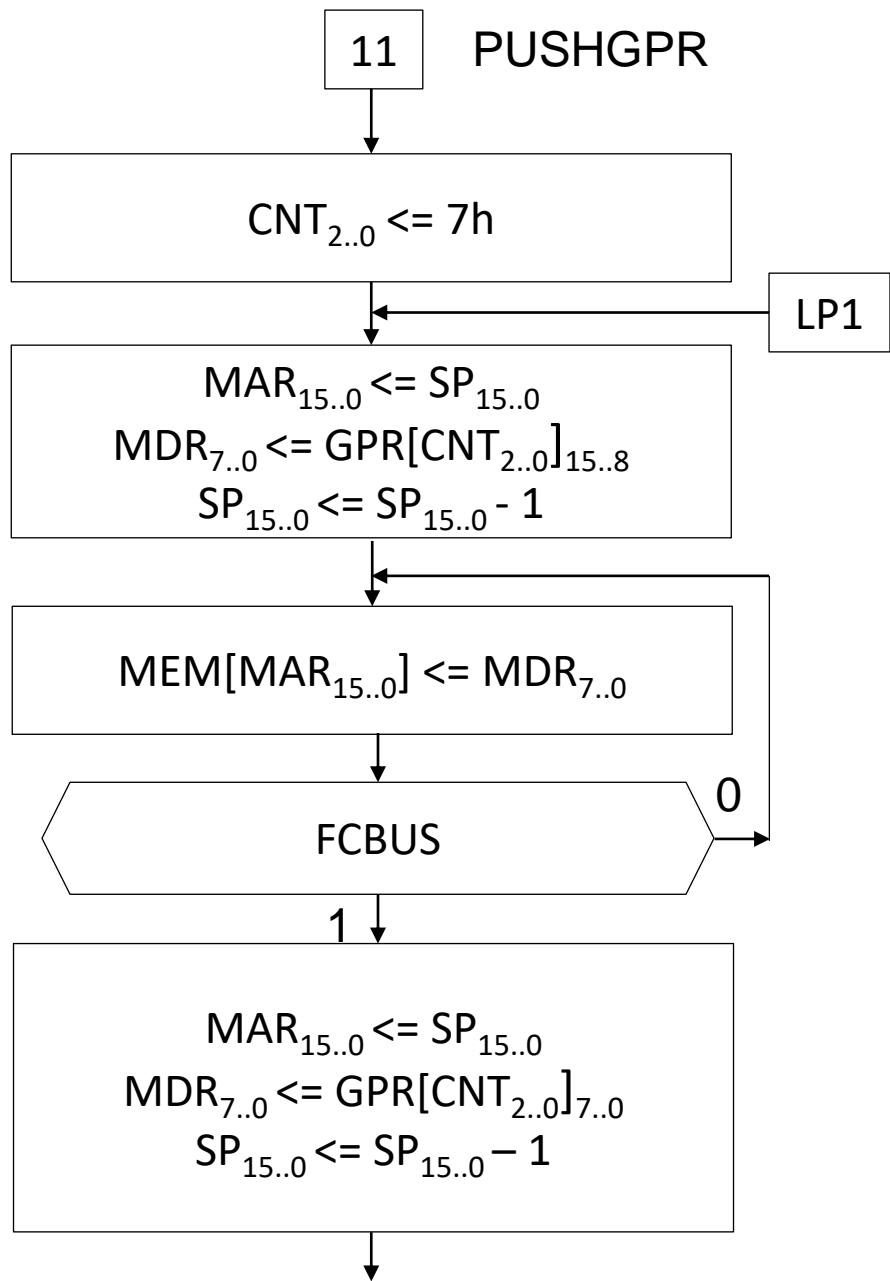


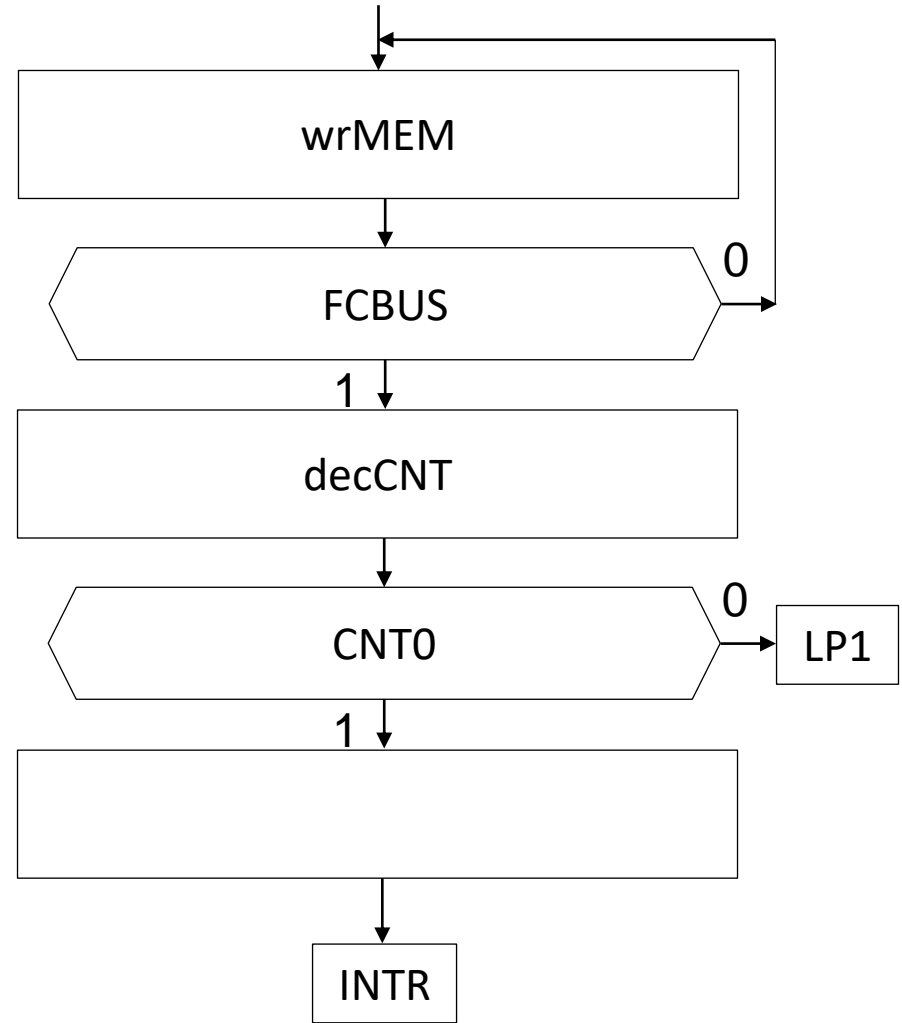
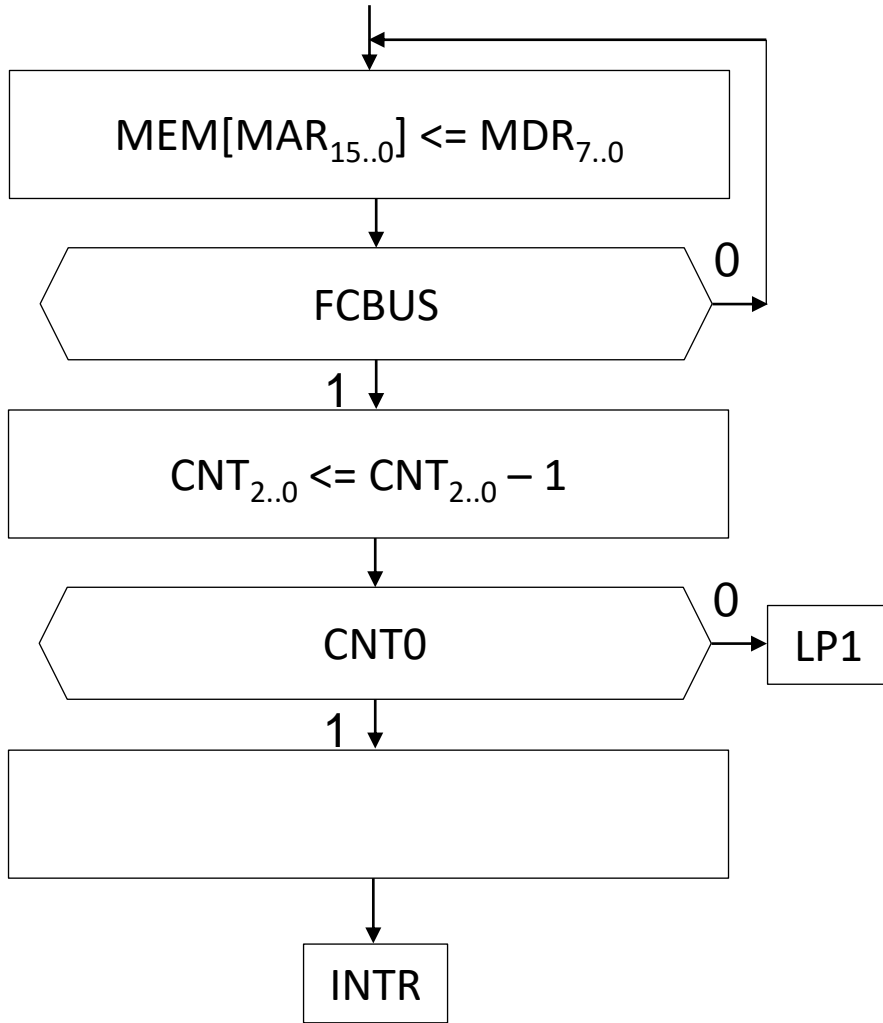


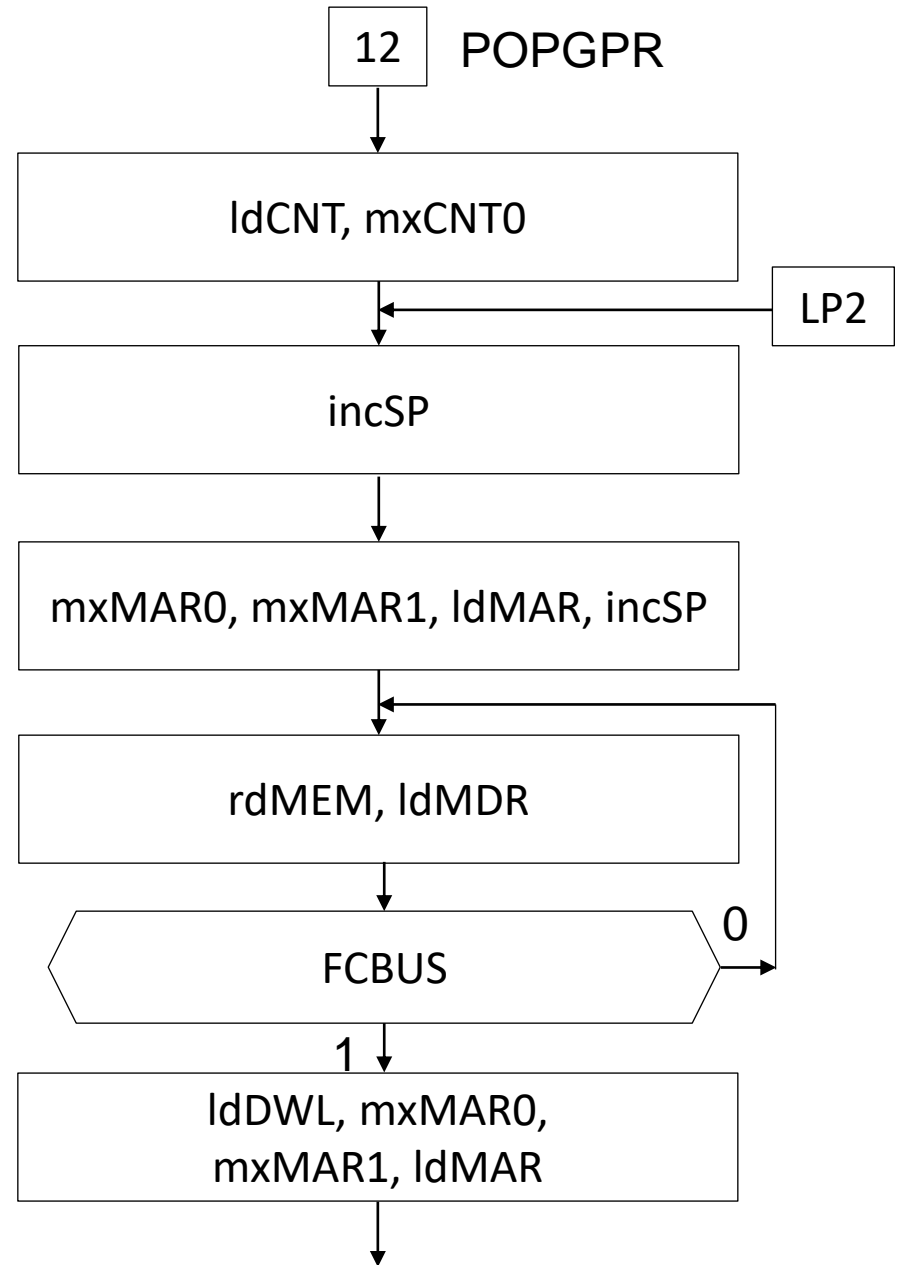
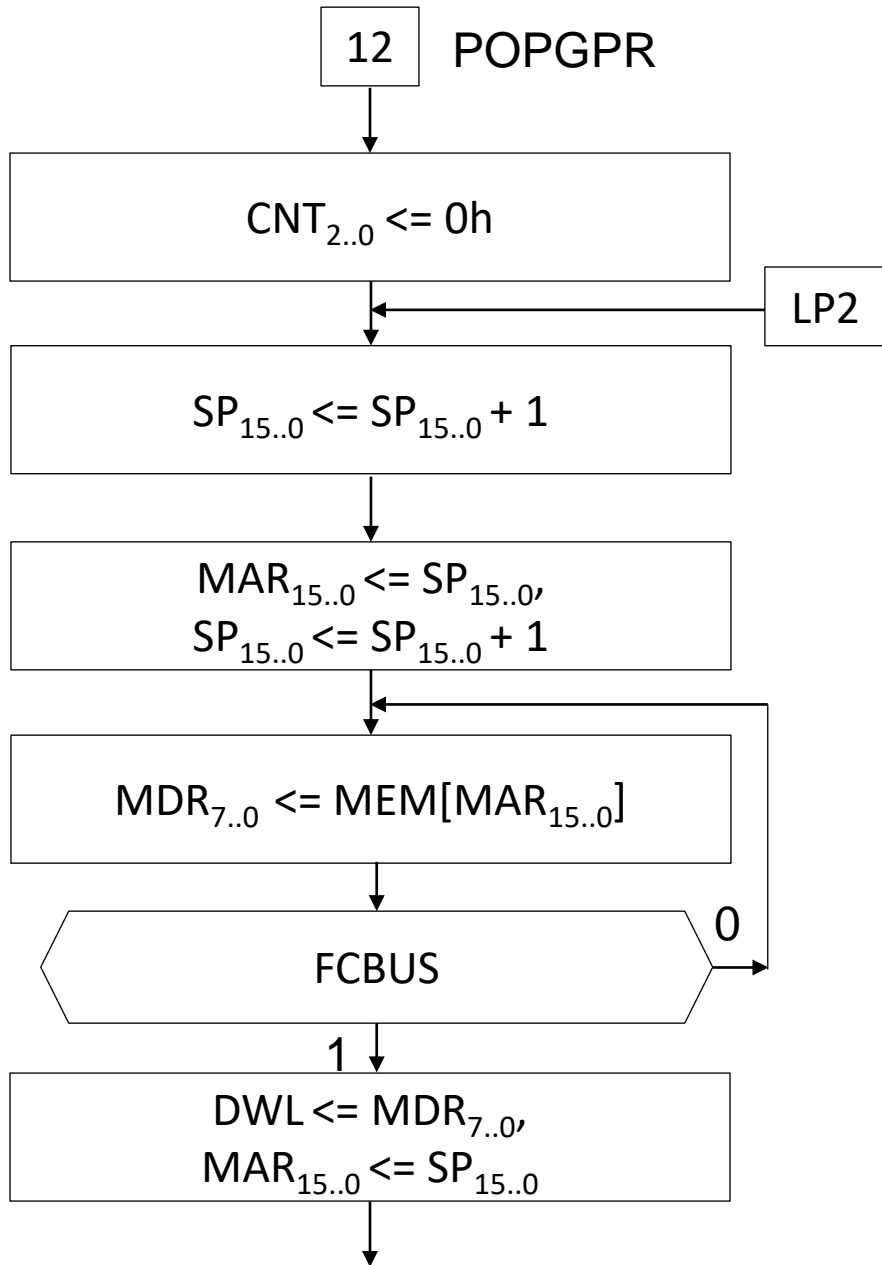


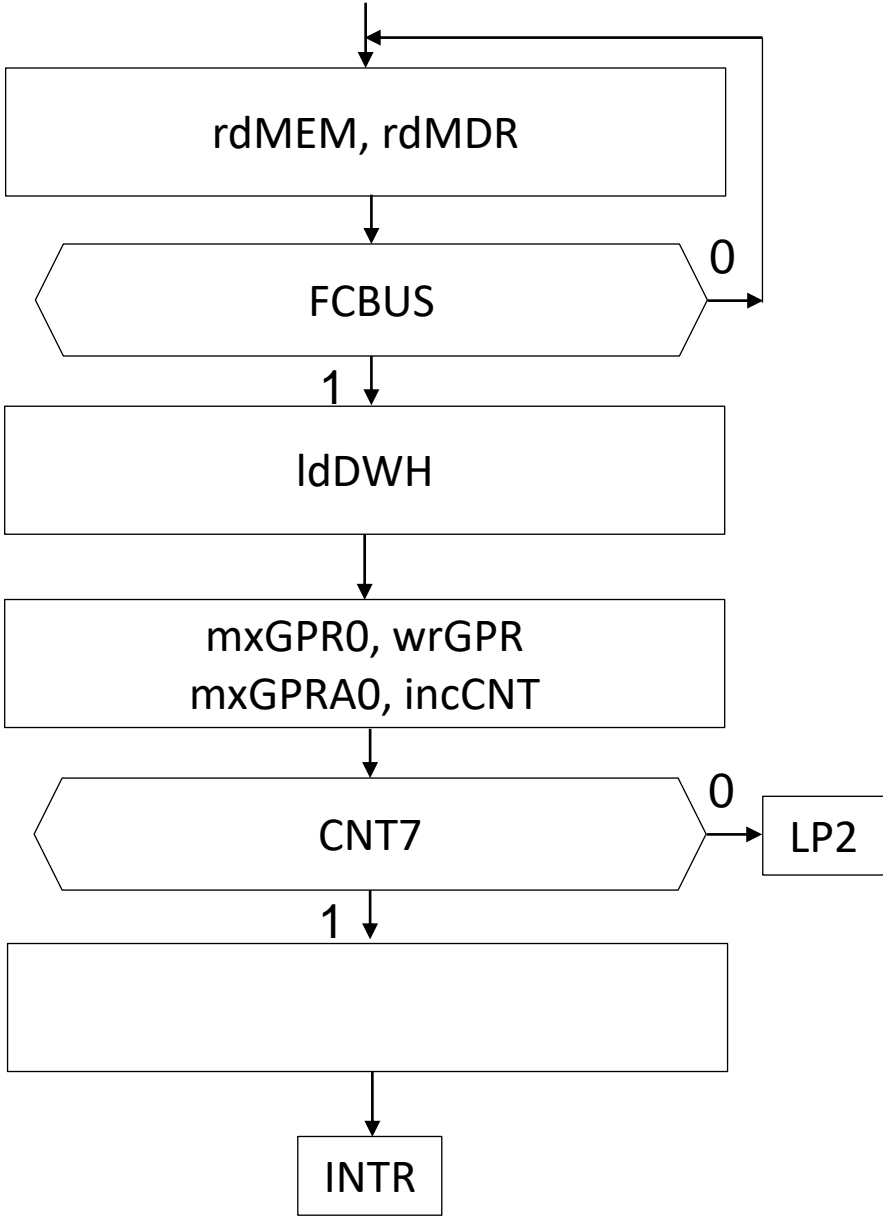
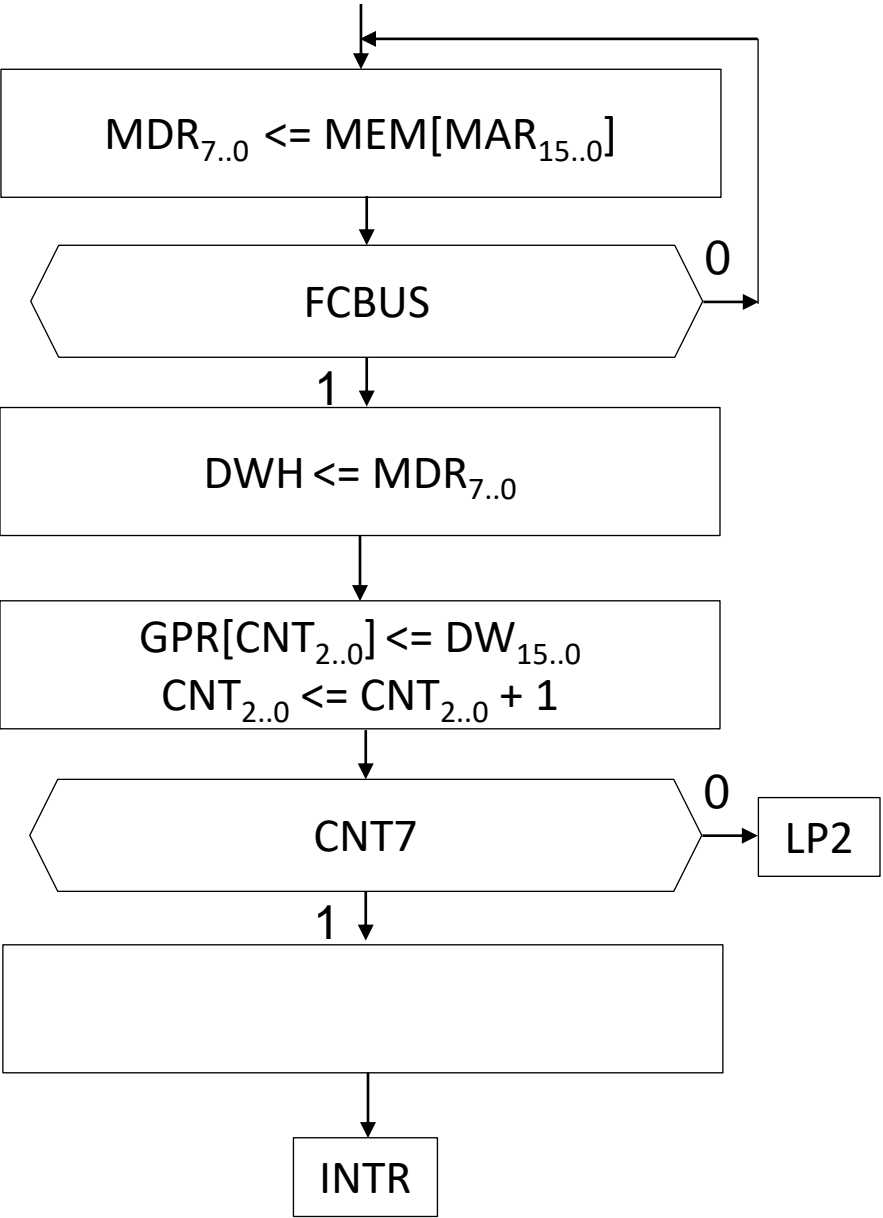


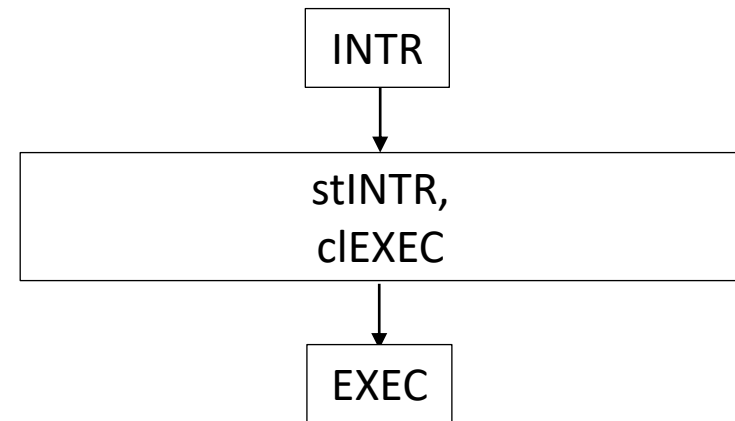
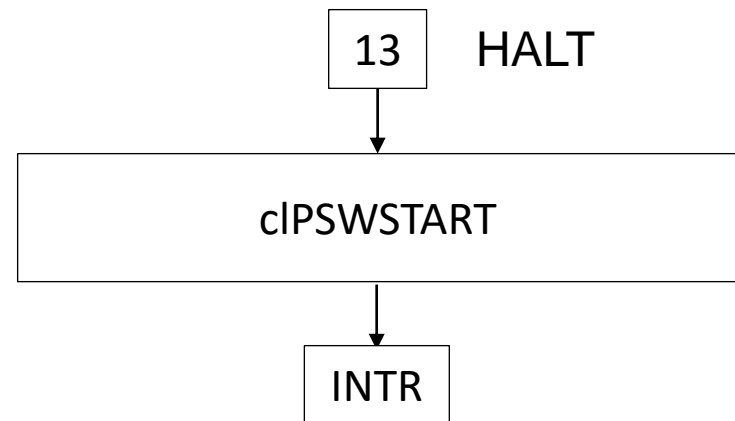
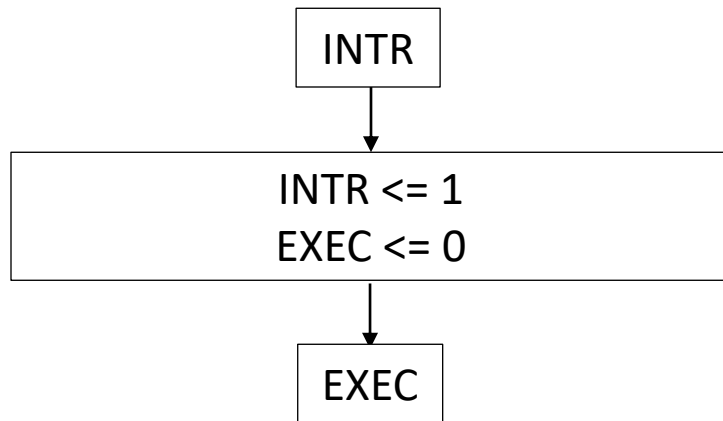
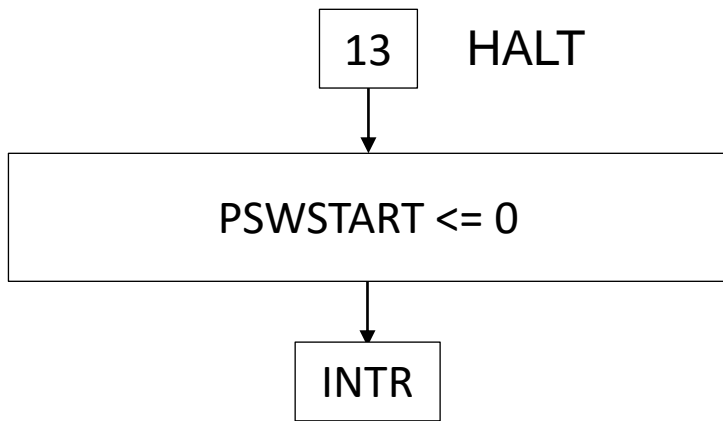




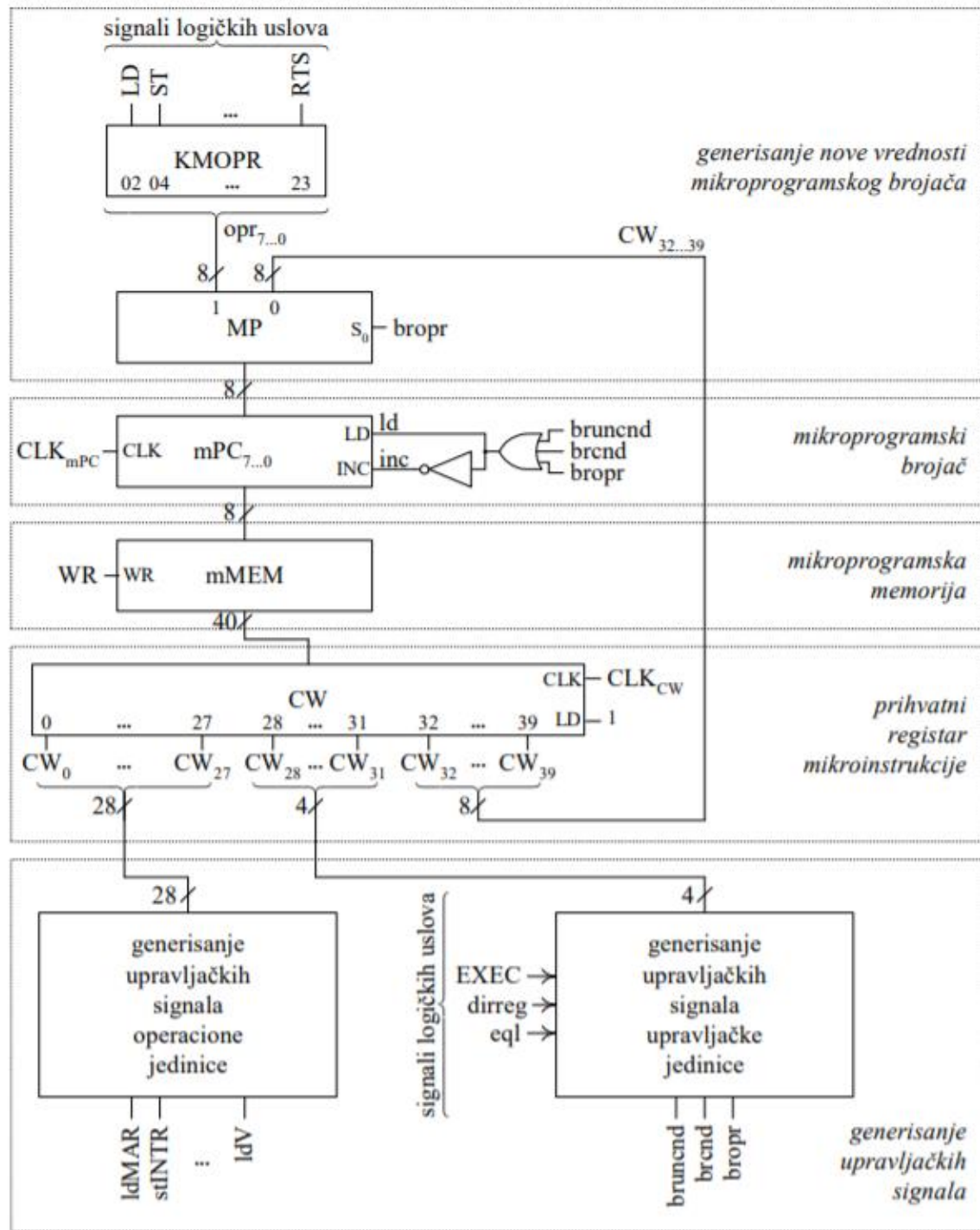






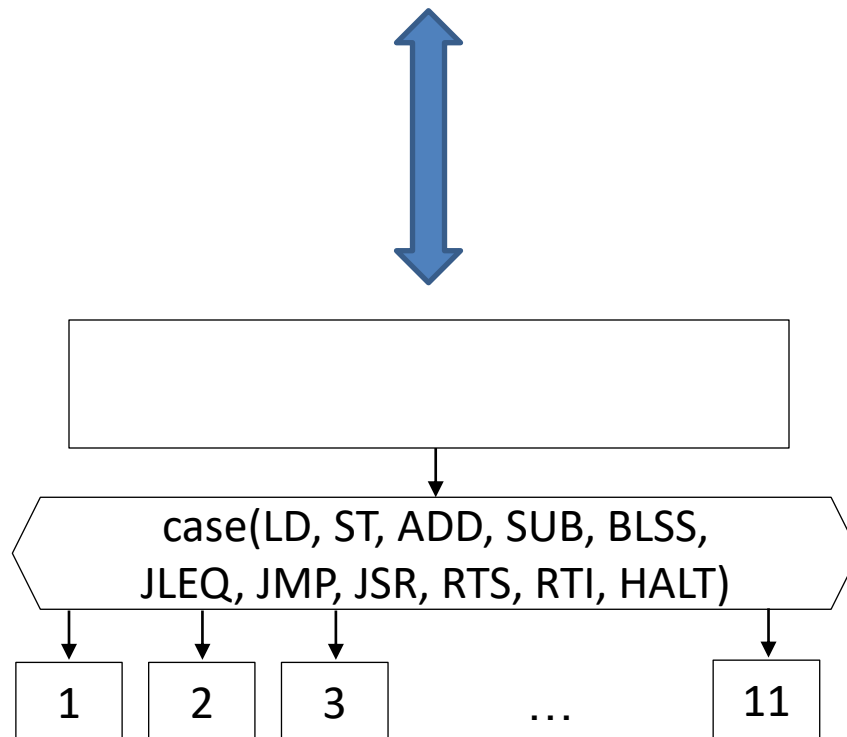


Управљачка јединица



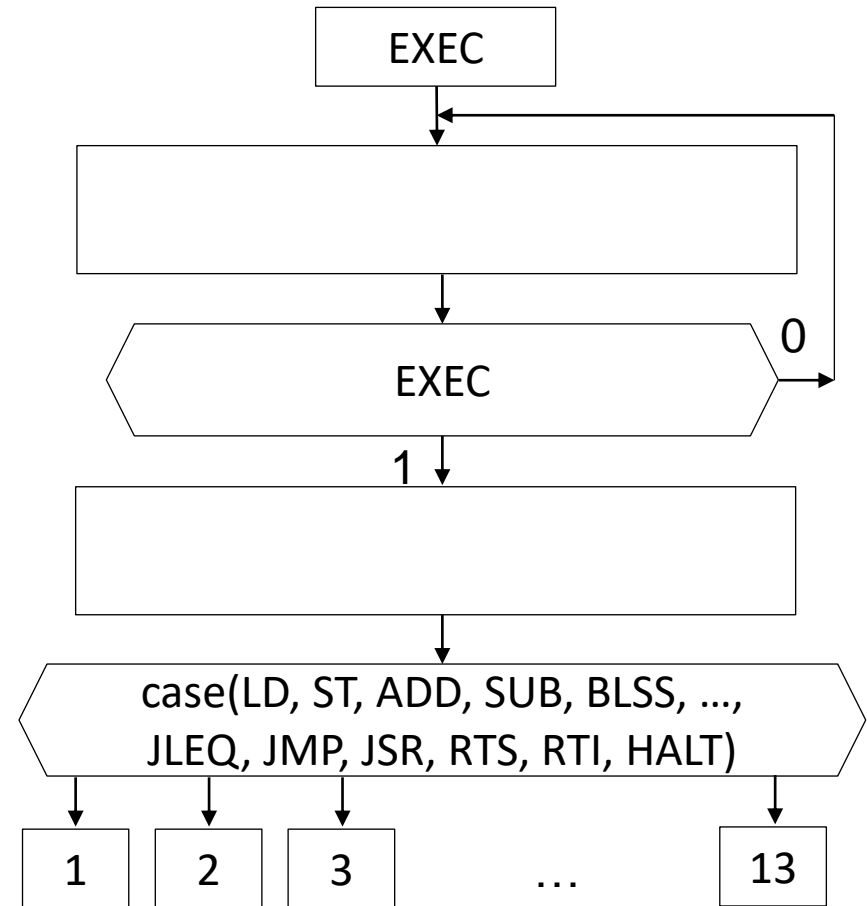
Секвенца управљачких сигнала

$\text{step}_{xx} \text{ br}(\text{case}(\text{LD}, \text{ST}, \dots, \text{RTI}, \text{HALT}) \text{ then}(\text{LD}, \mathbf{\text{step}}), (\text{ST}, \mathbf{\text{step}}) \dots, (\text{RTI}, \mathbf{\text{step}}), (\text{HALT}, \mathbf{\text{step}}))$



step00 br(if notEXEC then step00)

step01 br(case(LD, ST,..., POPGPR, HALT)
then(LD, **step02**), (ST, **step**),... ,(POPGPR,
step),(HALT, **step**))



step00 br(if notEXEC then step00)
step01 br(case(LD, ST,..., POPGPR, HALT)
then(LD, **step02**), (ST, **step04**),...
,(POPGPR, **step**),(HALT, **step**))

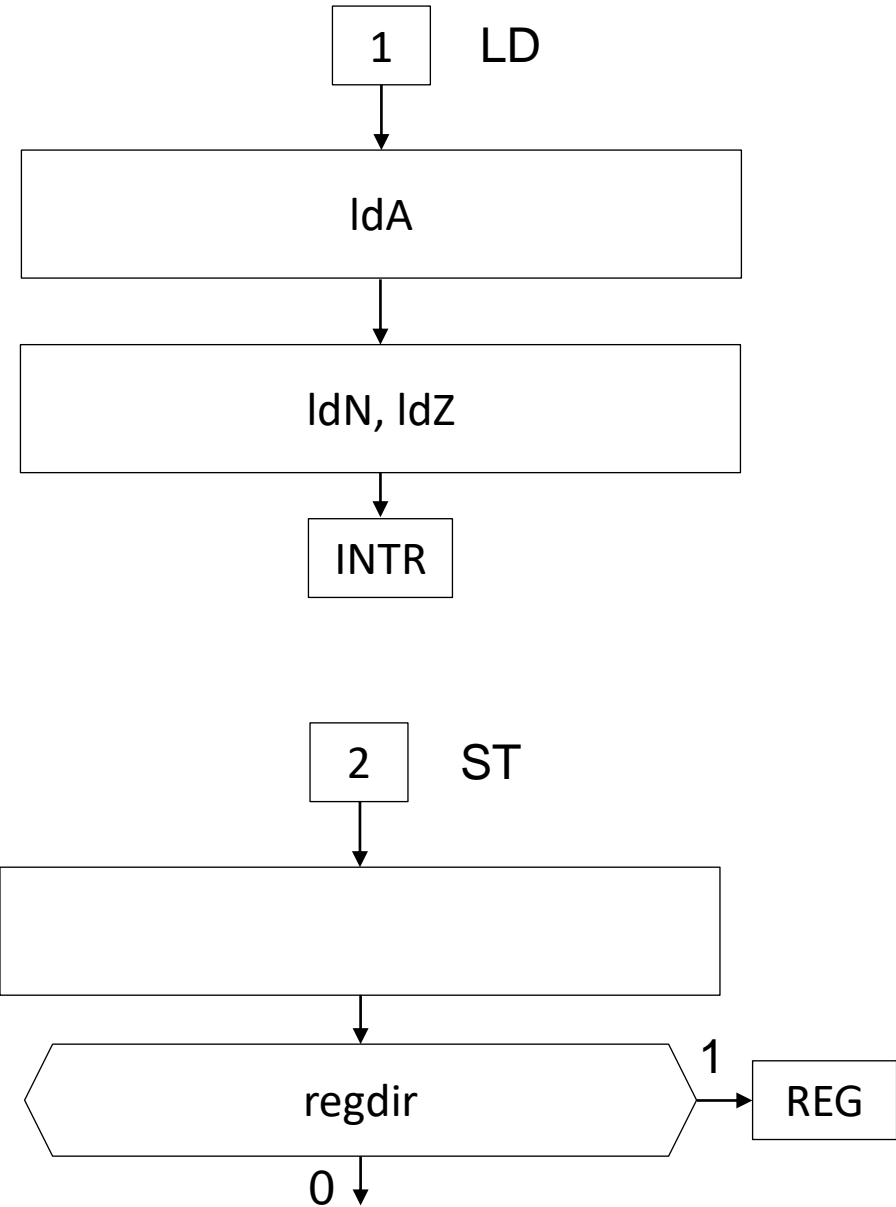
//-----LD-----

step02 ldA

step03 ldN, ldZ, br **step**

//-----ST-----

step04 br(if regdir then **step**)



step00 br(if notEXEC then step00)

step01 br(case(LD, ST,..., POPGPR, HALT)
then(LD, **step02**), (ST, **step04**),...
,(POPGPR, **step**),(HALT, **step**))

//-----LD-----

step02 ldA

step03 ldN, ldZ, br **step**

//-----ST-----

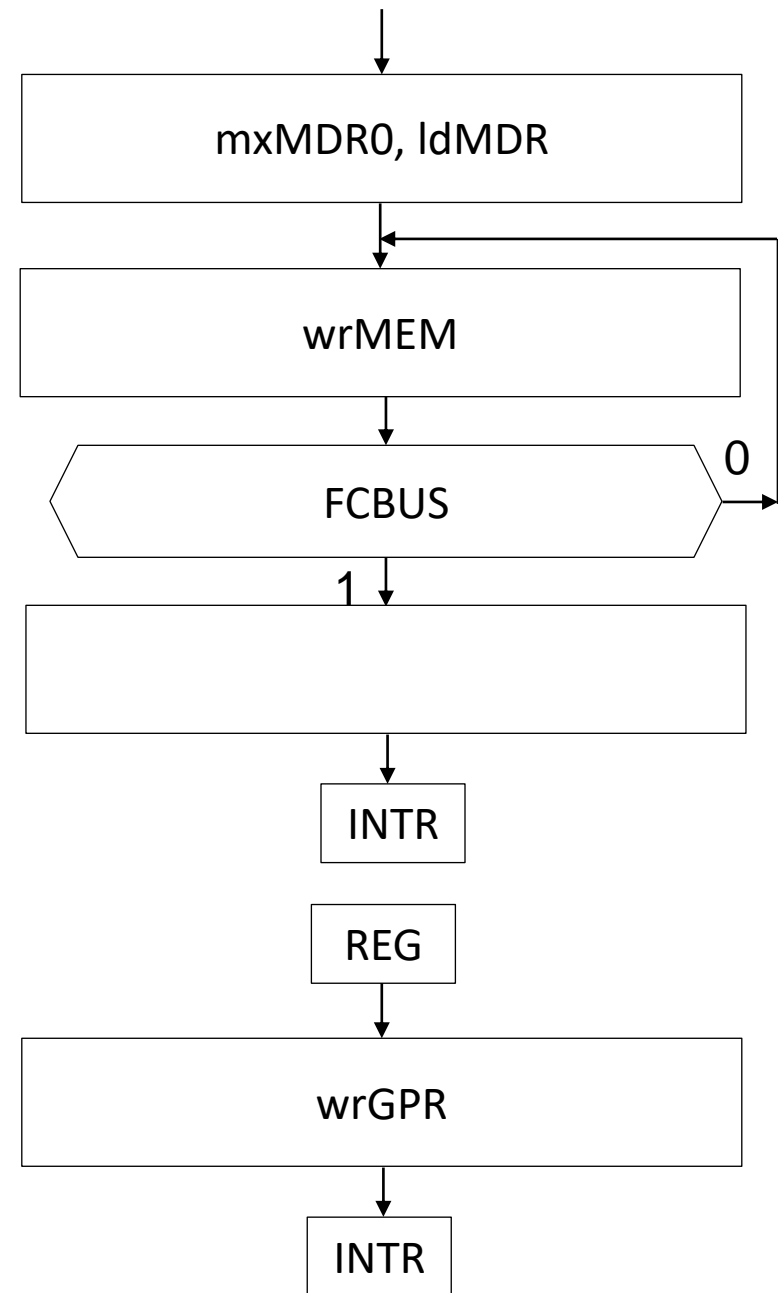
step04 br(if regdir then **step08**)

step05 mxMDR0, ldMDR

step06 wrMEM,
br(if notFCBUS then step06)

step07 br **step**

step08 wrGPR, br **step**



//-----ADD-----

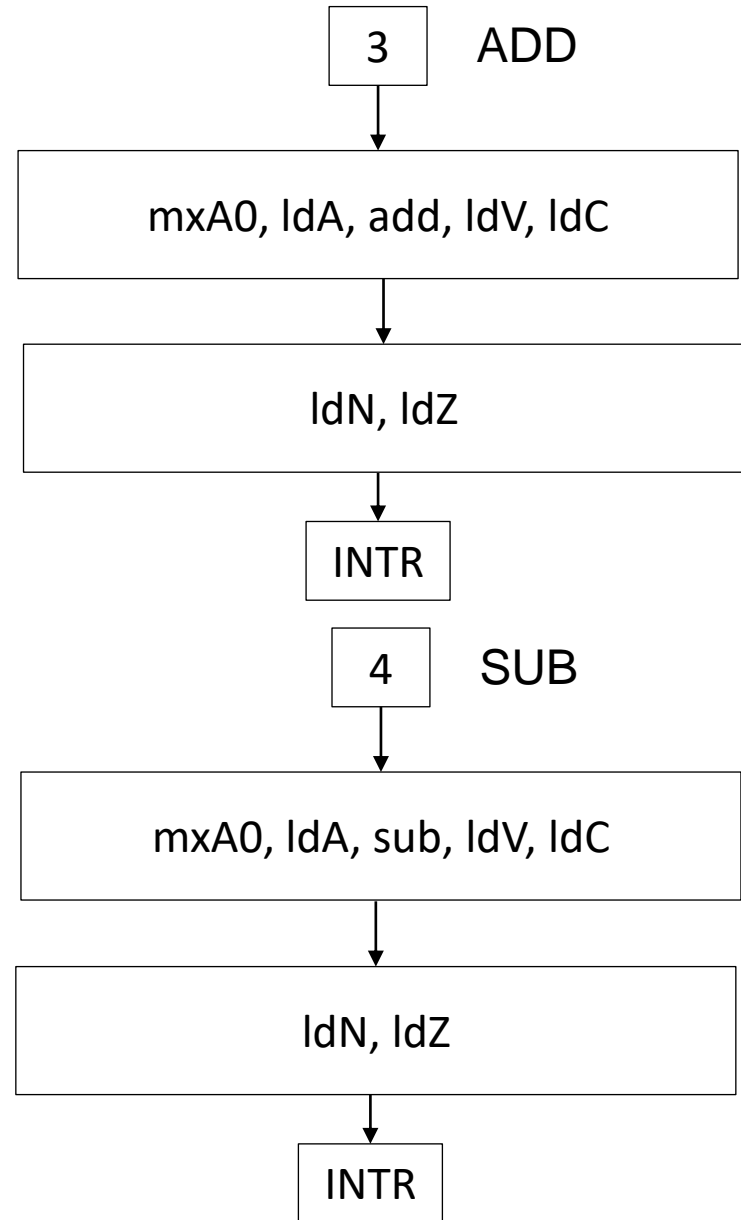
step09 mxA0, ldA, add, ldV, ldC

step0A ldN, ldZ, br **step**

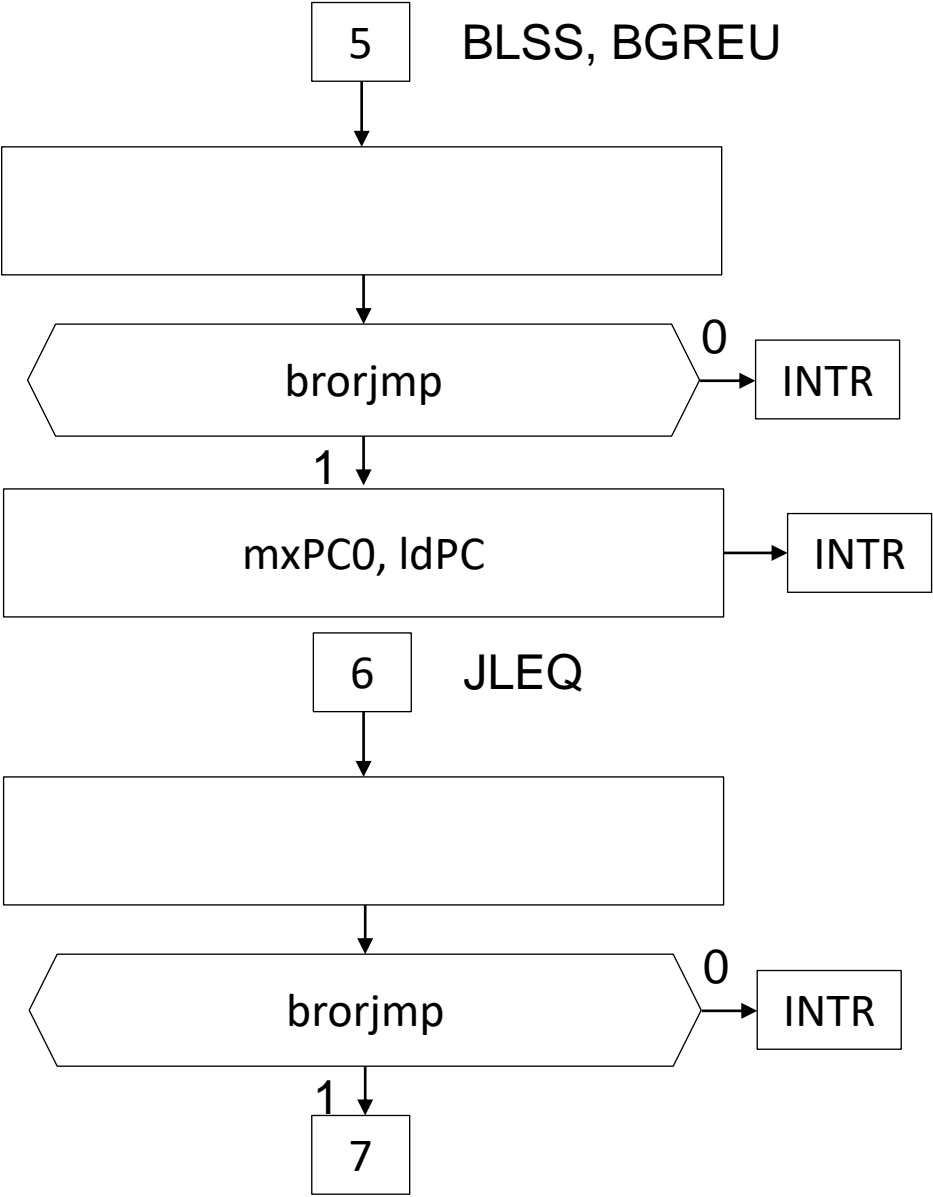
//-----SUB-----

step0B mxA0, ldA, sub, ldV, ldC

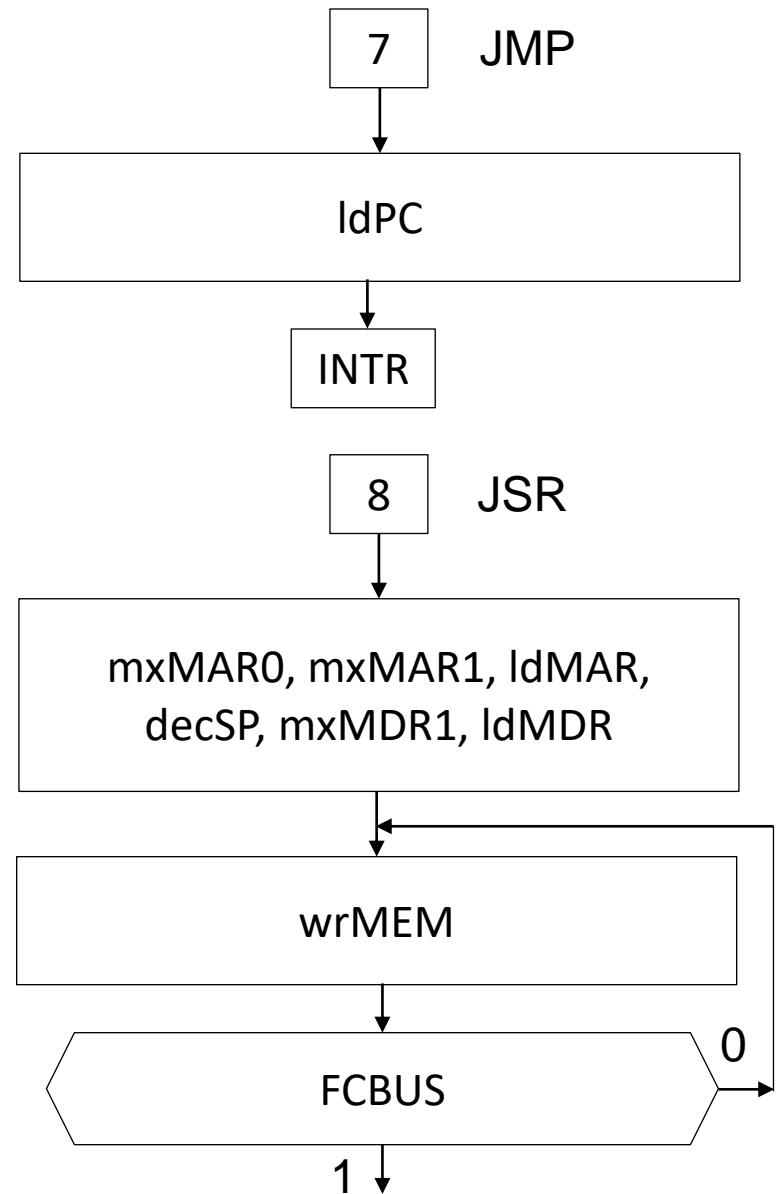
step0C ldN, ldZ, br **step**



//-----ADD-----
step09 mxA0, ldA, add, ldV, ldC
step0A ldN, ldZ, br **step**
//-----SUB-----
step0B mxA0, ldA, sub, ldV, ldC
step0C ldN, ldZ, br **step**
//-----BLSS,BGREU-----
step0D br(if **not**brorjmp then **step**)
step0E mxPC0, ldPC, br **step**
//-----JLEQ-----
step0F br(if **not**brorjmp then **step**)
//-----JMP-----



```
//-----ADD-----
step09 mxA0, ldA, add, ldV, ldC
step0A ldN, ldZ, br step
//-----SUB-----
step0B mxA0, ldA, sub, ldV, ldC
step0C ldN, ldZ, br step
//-----BLSS,BGREU-----
step0D br(if notbrorjmp then step)
step0E mxPC0, ldPC, br step
//-----JLEQ-----
step0F br(if notbrorjmp then step)
//-----JMP-----
step10 ldPC, br step
//-----JSR-----
step11 mxMAR0, mxMAR1, ldMAR,
        decSP, mxMDR1, ldMDR
step12 wrMEM,
        br(if notFCBUS then step12)
```



//-----JSR-----

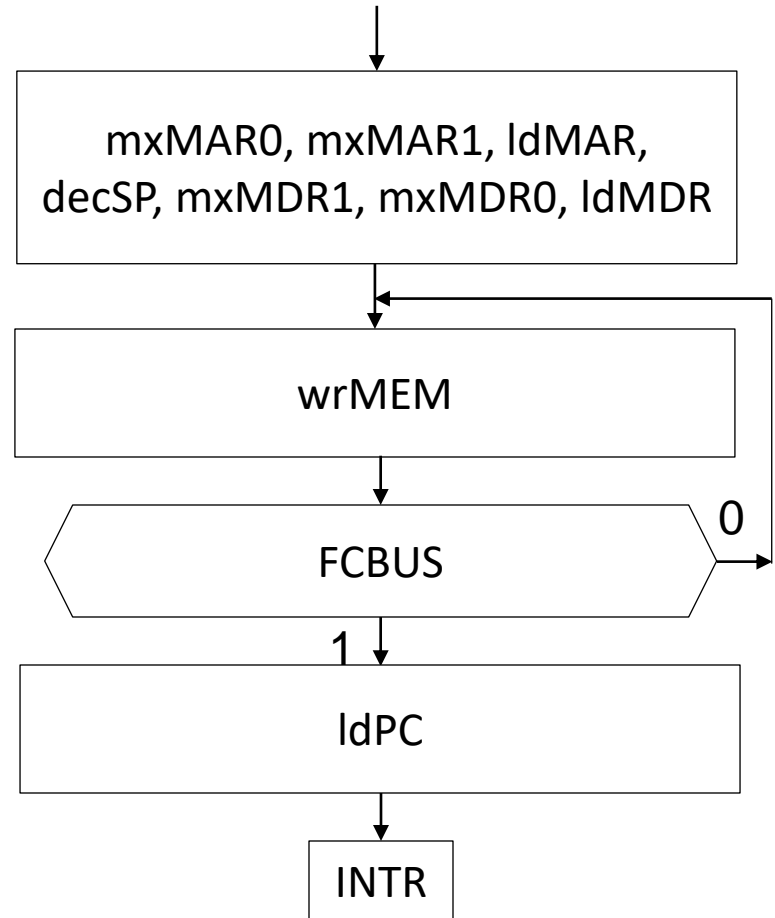
step11 mxMAR0, mxMAR1, ldMAR,
decSP, mxMDR1, ldMDR

step12 wrMEM,
br(if notFCBUS then **step12**)

step13 mxMAR0, mxMAR1, ldMAR,
decSP, mxMDR1, mxMDR0, ldMDR

step14 wrMEM,
br(if notFCBUS then **step14**)

step15 ldPC, br **step**



//-----JSR-----

step11 mxMAR0, mxMAR1, ldMAR,
decSP, mxMDR1, ldMDR

step12 wrMEM,
br(if notFCBUS then **step12**)

step13 mxMAR0, mxMAR1, ldMAR,
decSP, mxMDR1, mxMDR0, ldMDR

step14 wrMEM,
br(if notFCBUS then **step14**)

step15 ldPC, br **step**

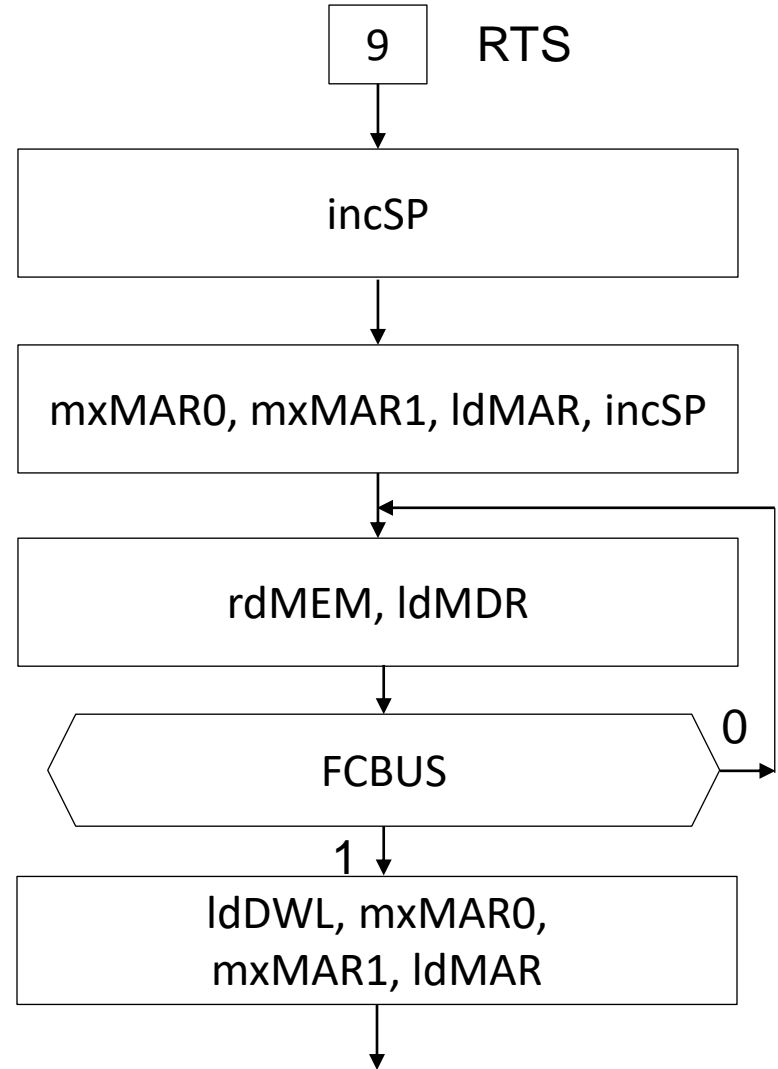
//-----RTS-----

step16 incSP

step17 mxMAR0, mxMAR1, ldMAR, incSP

step18 rdMEM, ldMDR,
br(if notFCBUS then **step18**)

step19 ldDWL, mxMAR0, mxMAR1,
ldMAR



//-----RTS-----

step16 incSP

step17 mxMAR0,mxMAR1, ldMAR, incSP

step18 rdMEM, ldMDR,

br(if notFCBUS then **step18**)

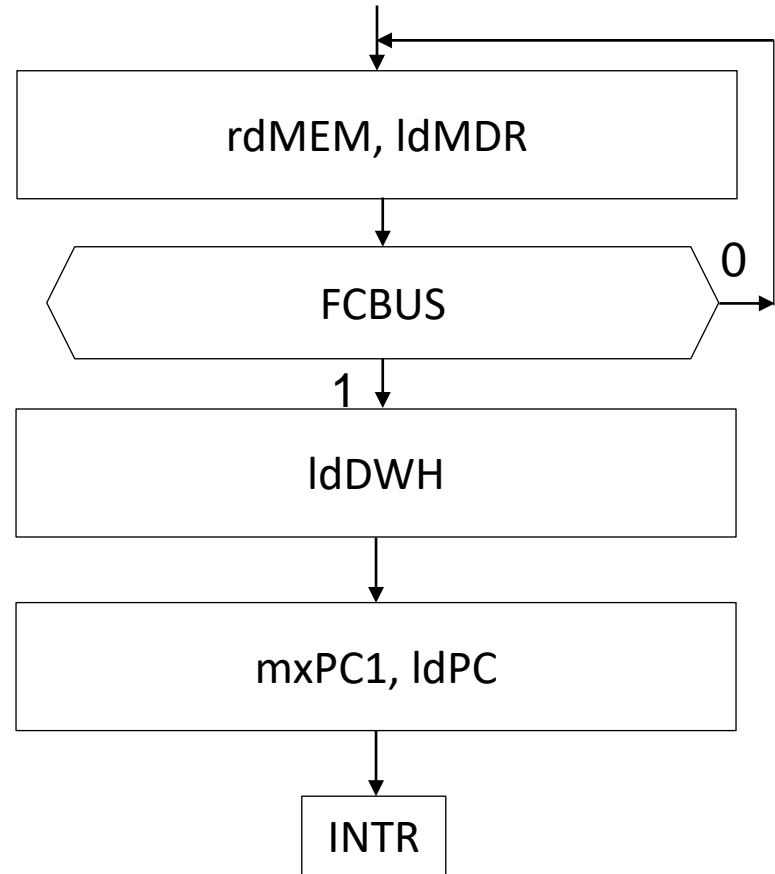
step19 ldDWL, mxMAR0, mxMAR1,
ldMAR

step1A rdMEM, ldMDR,

br(if notFCBUS then **step1A**)

step1B ldDWH

step1C mxPC1, ldPC, br **step**



//-----RTS-----

step16 incSP

step17 mxMAR0,mxMAR1, ldMAR, incSP

step18 rdMEM, ldMDR,
br(if notFCBUS then **step18**)

step19 ldDWL, mxMAR0, mxMAR,ldMAR

step1A rdMEM, ldMDR,
br(if notFCBUS then **step1A**)

step1B ldDWH

step1C mxPC1, ldPC, br **step**

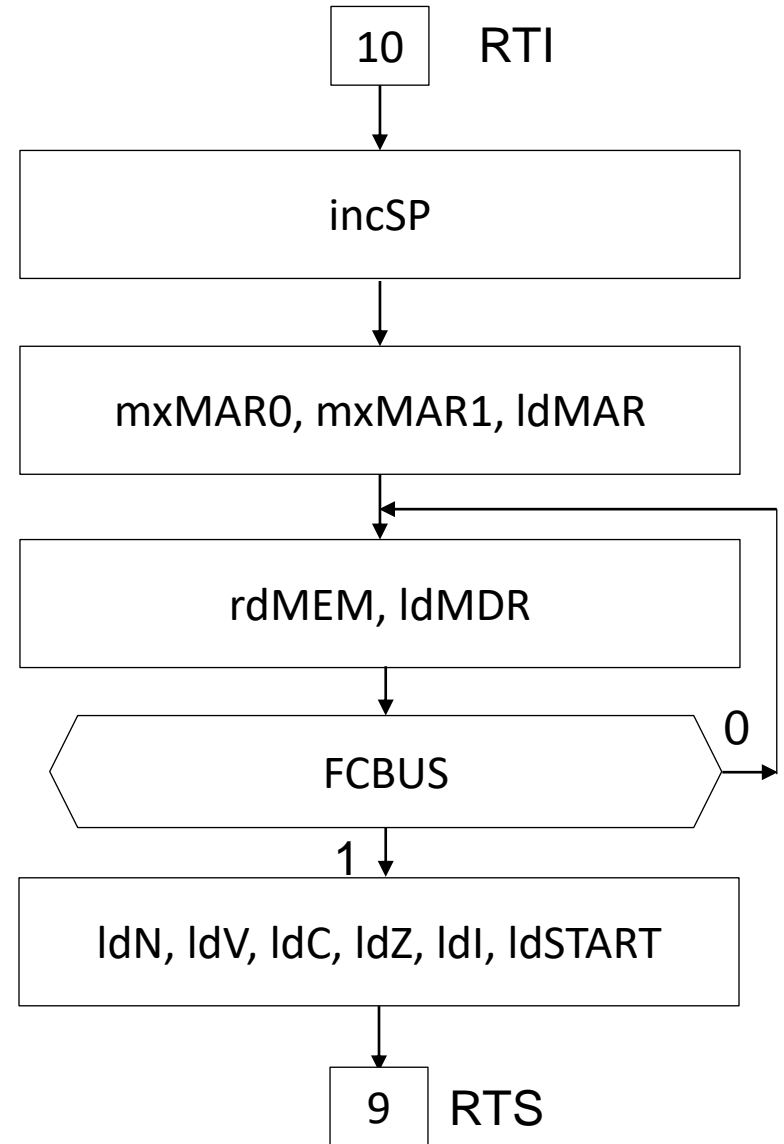
//-----RTI-----

step1D incSP

step1E mxMAR0, mxMAR1, ldMAR

step1F rdMEM, ldMDR,
br(if notFCBUS then **step1F**)

step20 ldN, ldV, ldC, ldZ, ldI, ldSTART,
br **step16**



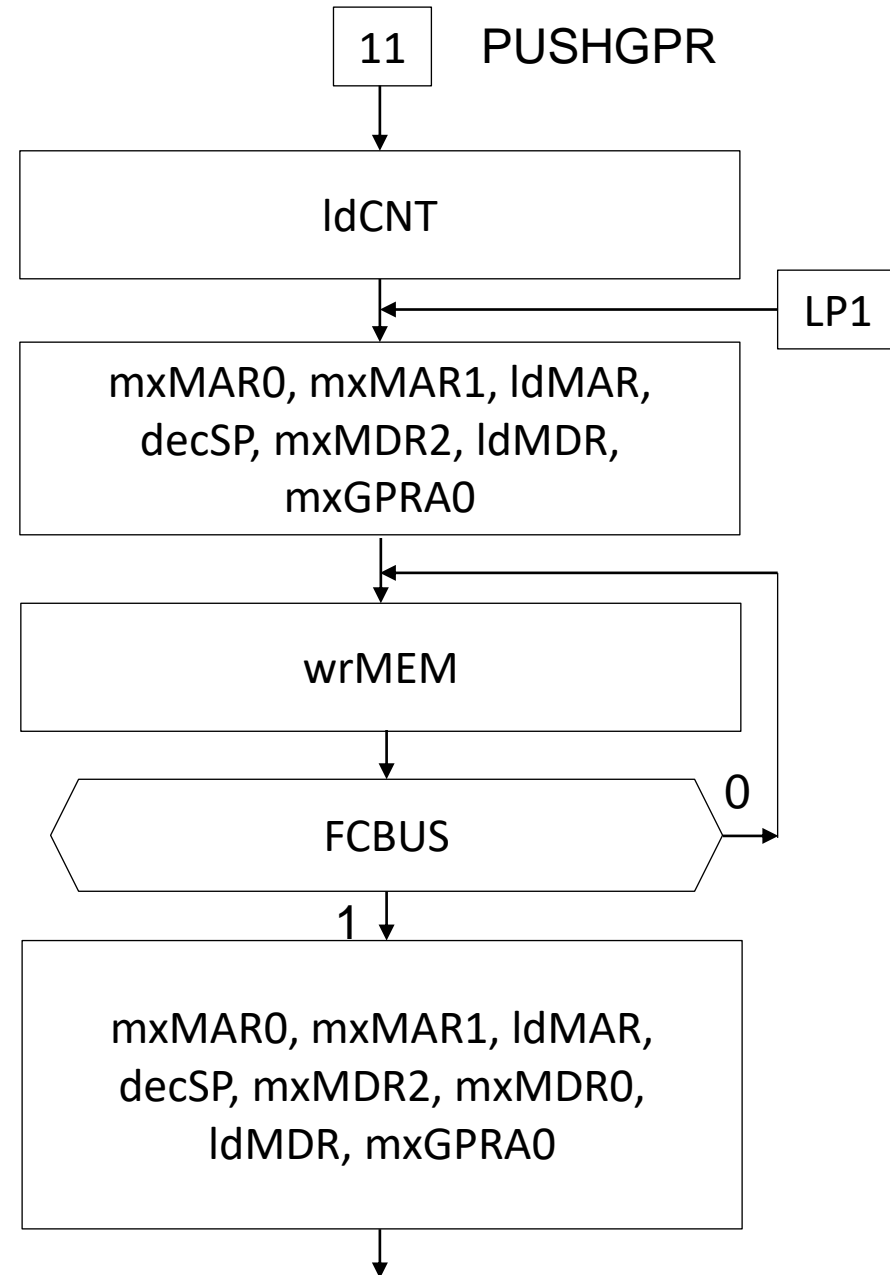
//-----PUSHGPR-----

step21 ldCNT

step22 mxMAR0, mxMAR1, ldMAR,
decSP, mxMDR2, ldMDR,
mxGPRA0

step23 wrMEM,
br(if notFCBUS then **step23**)

step24 mxMAR0, mxMAR1, ldMAR,
decSP, mxMDR2, mxMDR0,
ldMDR, mxGPRA0



//-----PUSHGPR-----

step21 ldCNT

step22 mxMAR0, mxMAR1, ldMAR,
decSP, mxMDR2, ldMDR,
mxGPRA0

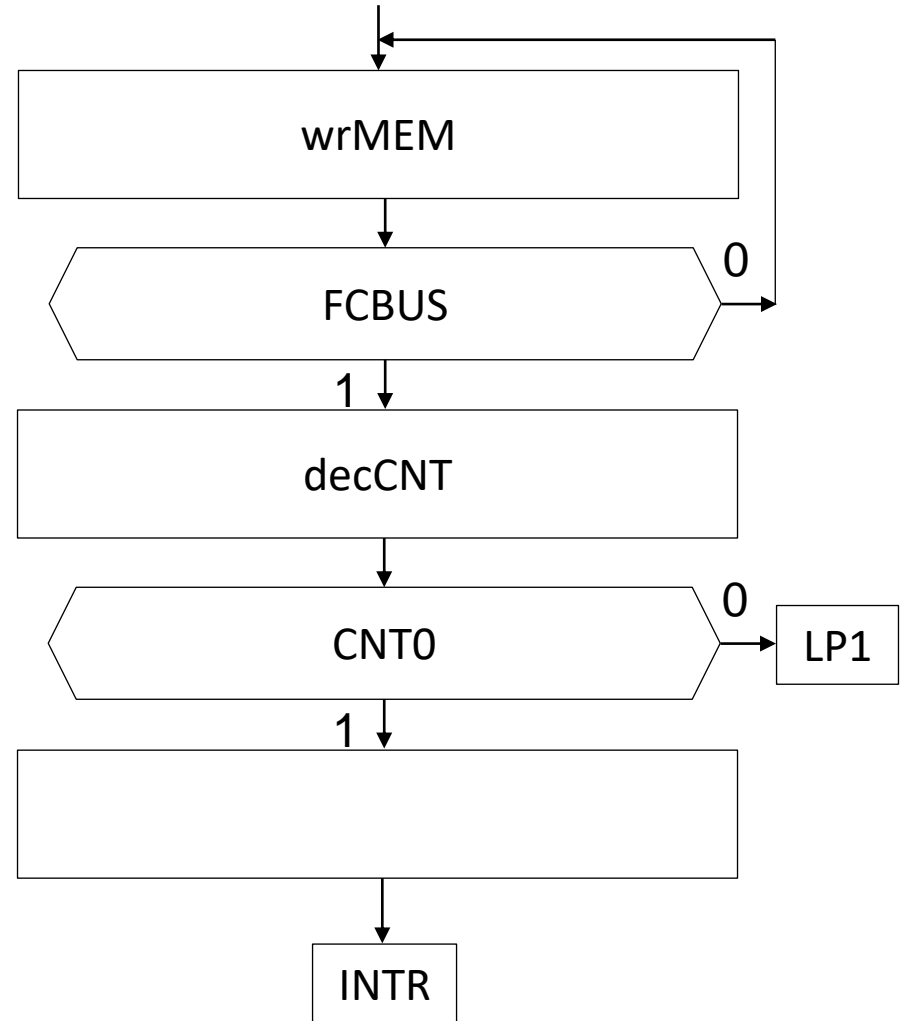
step23 wrMEM,
br(if notFCBUS then **step23**)

step24 mxMAR0, mxMAR1, ldMAR,
decSP, mxMDR2, mxMDR0,
ldMDR, mxGPRA0, decCNT

step25 wrMEM,
br(if notFCBUS then **step25**)

step26 decCNT
br(if notCNT0 then **step22**)

step27 br step



//-----POPGPR-----

step28 ldCNT, mxCNT0

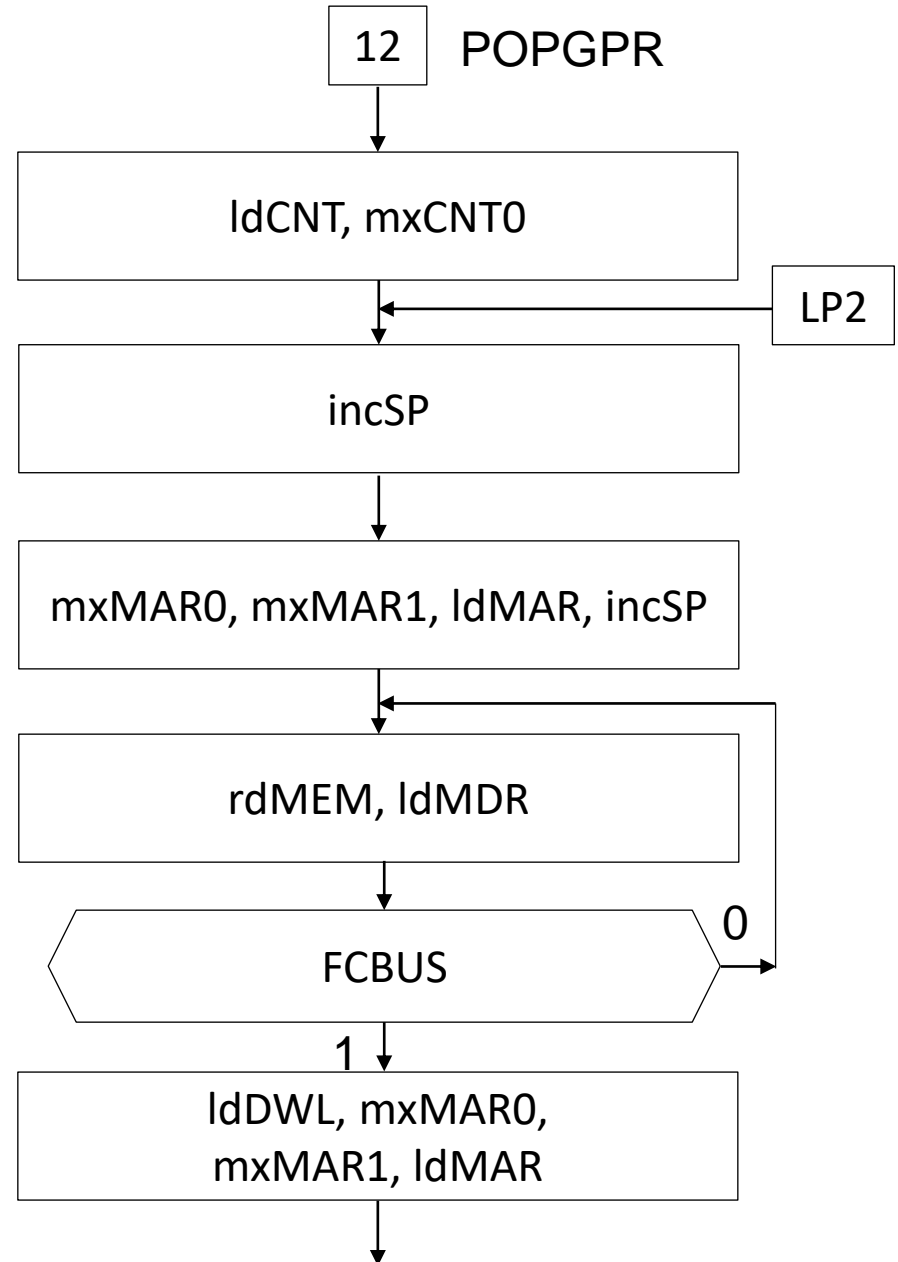
step29 incSP

step2A mxMAR0, mxMAR1, ldMAR,
incSP

step2B rdMEM, ldMDR

br(if notFCBUS then **step2B**)

step2C ldDWL, mxMAR0, mxMAR1,
ldMAR



//-----POPGPR-----

step28 ldCNT, mxCNT0

step29 incSP

step2A mxMAR0, mxMAR1, ldMAR,
incSP

step2B rdMEM, ldMDR

br(if notFCBUS then **step2B**)

step2C ldDWL, mxMAR0, mxMAR1,
ldMAR

step2D rdMEM, ldMDR

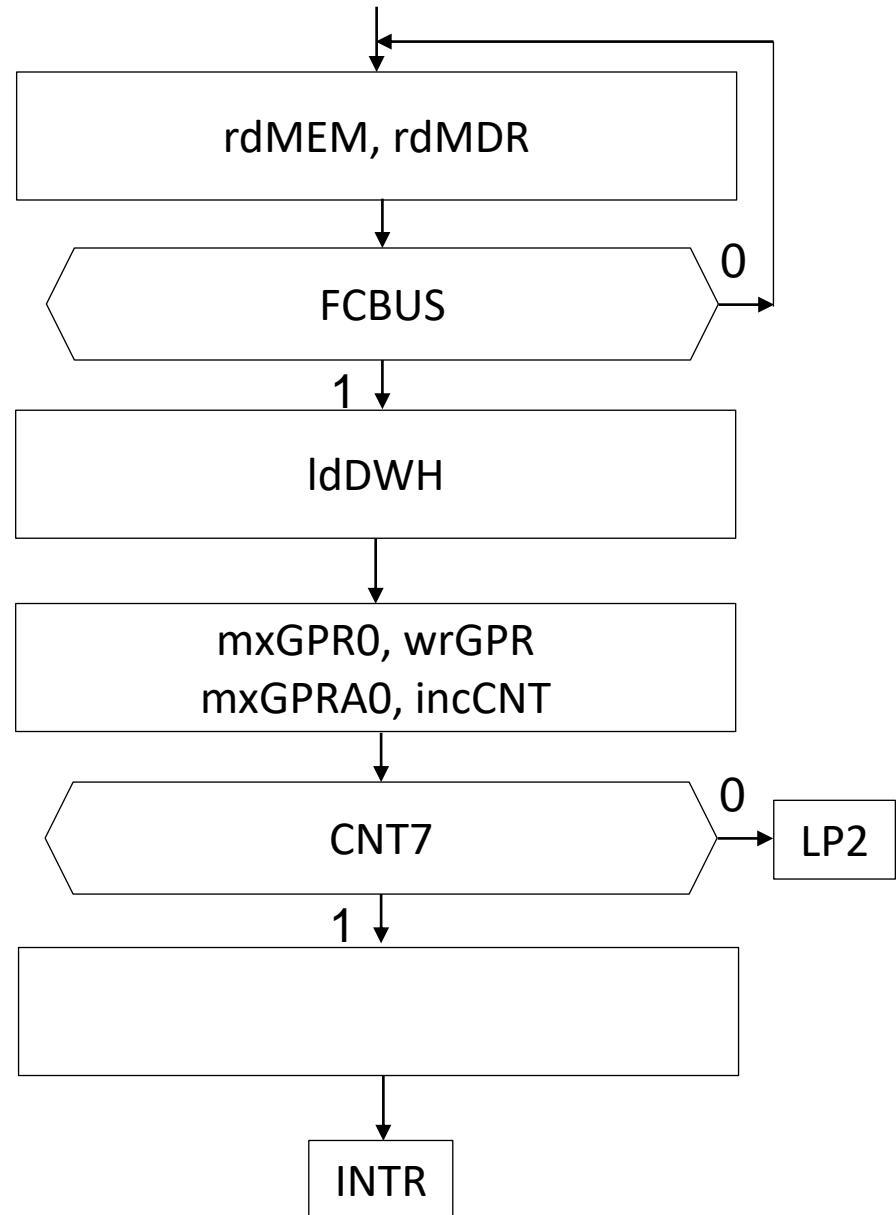
br(if notFCBUS then **step2D**)

step2E ldDWH

step2F mxGPR0, wrGPR, mxGPRA0,
incCNT

br(if notCNT7 then **step29**)

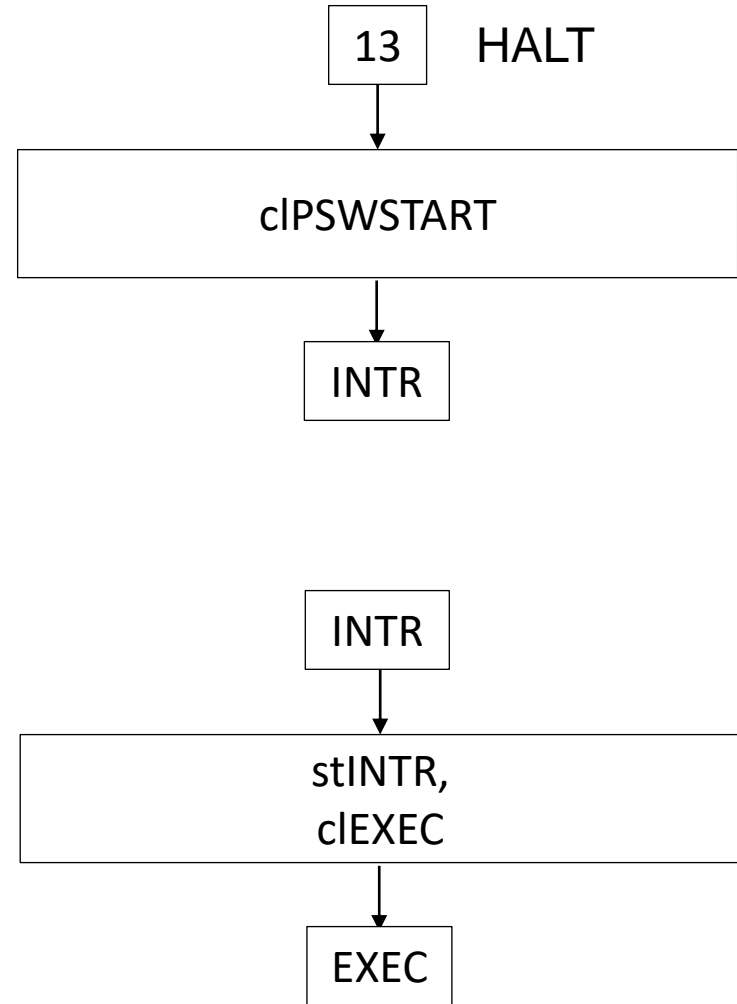
step30 br step



//-----HALT-----

step31 cIPSWSTART

step32 stINTR, cIEXEC, br **step00**



step00 br(if notEXEC then step00)
step01 br(case(LD, ST,..., POPGPR, HALT)
then(LD, **step02**), (ST, **step04**),...
,(POPGPR, **step**),(HALT, **step**))

//-----LD-----

step02 ldA

step03 ldN, ldZ, br **step32**

//-----ST-----

step04 br(if regdir then **step08**)

step05 mxMDR0, ldMDR

step06 wrMEM,
br(if notFCBUS then **step06**)

step07 br **step32**

step08 wrGPR, br **step32**

//-----ADD-----

step09 mxA0, ldA, add, ldV, ldC

step0A ldN, ldZ, br **step32**

0	1	2	3
ldA	ldN	ldZ	mxMDR0

4	5	6	7
ldMDR	wrMEM	wrGPR	mxA0

8	9	10	11
add	ldV	ldC	

12	13	14	15

16	17	18	19

20	21	22	23

//-----SUB-----

step0B mxA0, ldA, sub, ldV, ldC

step0C ldN, ldZ, br **step32**

//-----BLSS,BGREU-----

step0D br(if **not**brorjmp then **step32**)

step0E mxPC0, ldPC, br **step32**

//-----JLEQ-----

step0F br(if **not**brorjmp then **step32**)

//-----JMP-----

step10 ldPC, br **step32**

//-----JSR-----

step11 mxMAR0, mxMAR1, ldMAR,
decSP, mxMDR1, ldMDR

step12 wrMEM,
br(if notFCBUS then **step12**)

0	1	2	3
ldA	ldN	ldZ	mxMDR0

4	5	6	7
ldMDR	wrMEM	wrGPR	mxA0

8	9	10	11
add	ldV	ldC	sub

12	13	14	15
mxPC0	ldPC	mxMAR0	mxMAR1

16	17	18	19
ldMAR	decSP	mxMDR1	

20	21	22	23

step13 mxMAR0, mxMAR1, ldMAR,
decSP, mxMDR1, mxMDR0, ldMDR

step14 wrMEM,
br(if notFCBUS then **step14**)

step15 ldPC, br **step32**

//-----RTS-----

step16 incSP

step17mxMAR0, mxMAR1, ldMAR, incSP

step18 rdMEM, ldMDR,
br(if notFCBUS then **step18**)

step19 ldDWL, mxMAR0, mxMAR1,
ldMAR

step1A rdMEM, ldMDR,
br(if notFCBUS then **step1A**)

step1B ldDWH

step1C mxPC1, ldPC, br **step32**

0	1	2	3
ldA	ldN	ldZ	mxMDR0

4	5	6	7
ldMDR	wrMEM	wrGPR	mxA0

8	9	10	11
add	ldV	ldC	sub

12	13	14	15
mxPC0	ldPC	mxMAR0	mxMAR1

16	17	18	19
ldMAR	decSP	mxMDR1	incSP

20	21	22	23
rdMEM	ldDWL	ldDWH	mxPC1

//-----RTI-----

step1D incSP

step1E mxMAR0, mxMAR1, ldMAR

step1F rdMEM, ldMDR,
br(if notFCBUS then **step1F**)

step20 ldN, ldV, ldC, ldZ, ldI, ldSTART,
br **step16**

//-----PUSHGPR-----

step21 ldCNT

step22 mxMAR0, mxMAR1, ldMAR,
decSP, mxMDR2, ldMDR,
mxGPRA0

24	25	26	27
ldSTART	ldCNT	mxMDR2	mxGPRA0

28	29	30	31

32	33	34	35

36	37	38	39

40	41	42	43

44	45	46	47

step23 wrMEM,
br(if notFCBUS then **step23**)

step24 mxMAR0, mxMAR1, ldMAR,
decSP, mxMDR2, mxMDR0,
ldMDR, mxGPRA0

step25 wrMEM,
br(if notFCBUS then **step25**)

step26 decCNT,
br(if notCNT0 then **step22**)

step27 br **step32**

//-----POPGPR-----

step28 ldCNT, mxCNT0

step29 incSP

step2A mxMAR0, mxMAR1, ldMAR,
incSP

step2B rdMEM, ldMDR
br(if notFCBUS then **step2B**)

step2C ldDWL, mxMAR0, mxMAR1,
ldMAR

24	25	26	27
ldSTART	ldCNT	mxMDR2	mxGPRA0

28	29	30	31
decCNT	mxCNT0		

32	33	34	35

36	37	38	39

40	41	42	43

44	45	46	47

step2D rdMEM, ldMDR
br(if notFCBUS then **step2D**)

step2E ldDWH

step2F mxGPR0, wrGPR, mxGPRA0,
incCNT
br(if notCNT7 then **step29**)

24	25	26	27
ldSTART	ldCNT	mxMDR2	mxGPRA0

28	29	30	31
decCNT	mxCNT0	incCNT	mxGPR0

step30 br **step32**
//-----HALT-----

32	33	34	35
clPSWSTART	ldl	stINTR	clEXEC

step31 clPSWSTART

step32 stINTR, clEXEC, br **step00**

36	37	38	39
/	/	/	/

40	41	42	43
/	/	/	/

44	45	46	47
/	/	/	/

Сигнал безусловног скока	CC
bruncnd	1

Сигнал условног скока	CC	Сигнал услова
brnotEXEX	2	$\overline{\text{EXEC}}$
brregdir	3	regdir
brbrorjmp	4	brorjmp
brnotCNT0	5	$\overline{\text{CNT0}}$
brnotCNT7	6	$\overline{\text{CNT7}}$
brnotFCBUS	7	$\overline{\text{FCBUS}}$

Сигнал вишеструког условног скока	CC
bropr	8

Услов	Корак
LD	2
ST	4
...	...
HALT	31

0	1	2	3
ldA	ldN	ldZ	mxMDR0

4	5	6	7
ldMDR	wrMEM	wrGPR	mxA0

8	9	10	...
add	ldV	ldC	...

45	46	47	48
/	/	/	/

52	53	54	55
cc			

56	57...	...62	63
ba			