

from any state

(J=37) state (36): ^(new latch) OLD-PSR \leftarrow PSR
LD.OLD-PSR=1
GatePSR=1

save user mode
vals (PSR, USP)

(J=38) (37): ^(new latch) USP \leftarrow R6
LD.USP=1
ALUK=11 (PASSA)
SRIMUX2=1

if SRIMUX2=1 and SRIMUX=0 \rightarrow use R6

GateALU=1

SUPERV.EN=1 (38): PSR[15] \leftarrow 0 (supervisor mode)
LD.REG=1 R6 \leftarrow SSP

DRMUX2=1 \rightarrow DRMUX=0

GateSSP=1

(J=39)

DEC=1 (sr2 mux (39): R6 \leftarrow R6-2 (decrement SSP)
ALUK=00 will select -2 as sr2

(add) GateALU=1 LD.REG=1 (J=40)

DRMUX2=1 SRIMUX2=1

SRIMUX2=1

ALUK=11

GateALU=1 LD.MAR=1

GateOLD-PSR=1 (40): MAR \leftarrow R6 (J=41)
LD.MDR=1; DATA.SIZE=1 (41): MDR \leftarrow OLD-PSR (J=44)

COND=01 R (44): M[MAR] \leftarrow MDR (J=44)

R.W=1

DATA.SIZE=1

DEC=1 ALUK=00 (46): R6 \leftarrow R6-2 (push PC onto stack) (J=42) SRIMUX2=1
GateALU=1, DRMUX2=1, LD.REG=1

push old PSR
onto system stack

SRIMUXZ=1

ALUK=11

GateALU=1

LD.MAR=1

GatePC=1

LD.MDR=1

DATA.SIZE=1

COND=01R

R.W.=1

DATA.SIZE=1

ADDR2MUXZ=1

(select INTV
For 1shf1)

ADDR1MUXZ=1
(select 0x200)

COND=01B

LD.MDR=1

DATA.SIZE=1

LD.PC=1

GateMDR=1

PCMUX=01, DATA.SIZE=1

MAR ← RL

MDR ← PC

M[MAR] ← MDR

MAR ← 0x200 + LSHF(INTV, 1)

MAR ← 0x200 (select address)

MDR ← M[MAR]

PC ← MDR

(J=43)

(J=48)

(J=48)

(J=52)

(J=52)

(J=18)

if EXC=1
use EXCV
nsh EXC

18

What restores condition codes
on context switch?

RTT → also RTI saves.
see restores USP?

RTI



PRIV_CLK=1 (8:) IF (PSR[15] == 1) priv mode violation (J=56)
SRMUX2=1, ALUK=11 MAR ← R6
GateALU=1, LD.MAR=1



COND=01 R (56:) MDR ← M[MAR] (J=56)
LD.MDR=1, MIO.EN=1
DATA.SIZE=1



LD.PC=1 (58:) PC ← MDR (J=34)
GateMDR=1
PCMUX=01, DATA.SIZE=1



INC=1 (sr2mux) (34:) R6 ← R6 + 2 (J=45)
ALUK=00 will select +2 as sr2
GateALU=1, LD.REG=1
DRMUX2=1

SRIMUX2=1



SRIMUX2=1 (45:) MAR ← R6 (J=60)
ALUK=11, GateALU=1
LD.MAR=1



COND=01 R (60:) MDR ← M[MAR] (J=60)
LD.MDR=1, DATA.SIZE=1
MIO.EN=1



LD.OLD_PSR=1 (62:) OLD_PSR ← MDR (J=47)
GateMDR=1, DATA.SIZE=1



INC=1, ALUK=00 (47:) R6 ← R6 + 2 (J=49)
GateALU=1, LD.REG=1, DRMUX2=1

SRIMUX2=1

SRIMUXZ=1 (49) SSP ← R6 (save SSP, restore USP)
ALUK=11, GateALL=1 (J=51)
LD.SSP=1

LD.REG=1 (51) R6 ← USP (J=53)
DRMUXZ=1, GateUSP=1

LD.PSR=1 (53) PSR ← OLD_PSR (J=18)
Gate(LD-PSR)=1 restore CCs
Rest-CCs=1 (restore CCs based on PSR)

★ (Add writethrough for CC to PSR) ★
(make push/pop manual for int.asm) 18