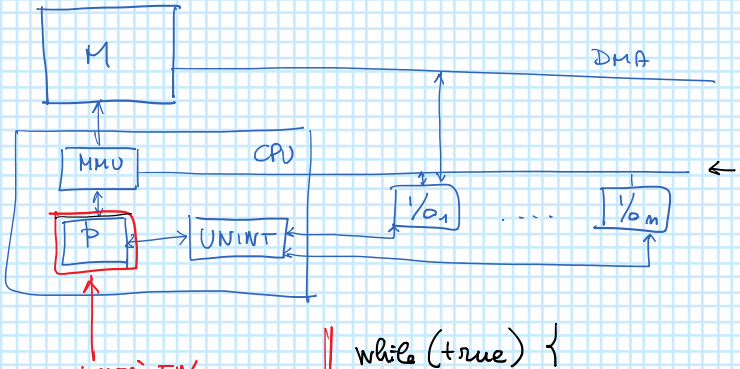
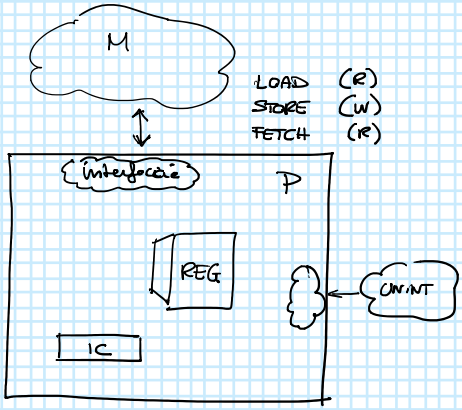
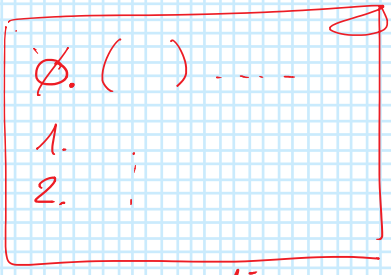
INTRODUZIONE

venerdì 28 ottobre 2016 11:52



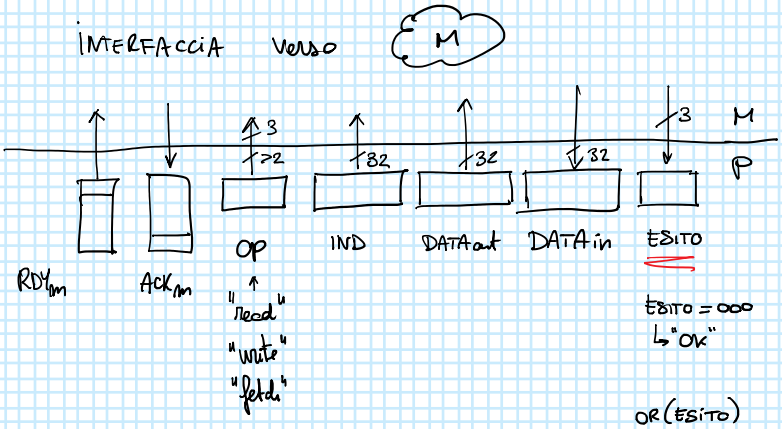


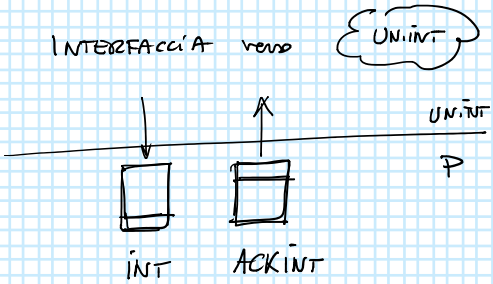

```
while (true) {
    fetch ISTR (IC/PC)
    decode ISTR
    exec ISTR
    trattamento interruzioni
}
```



μ-codice

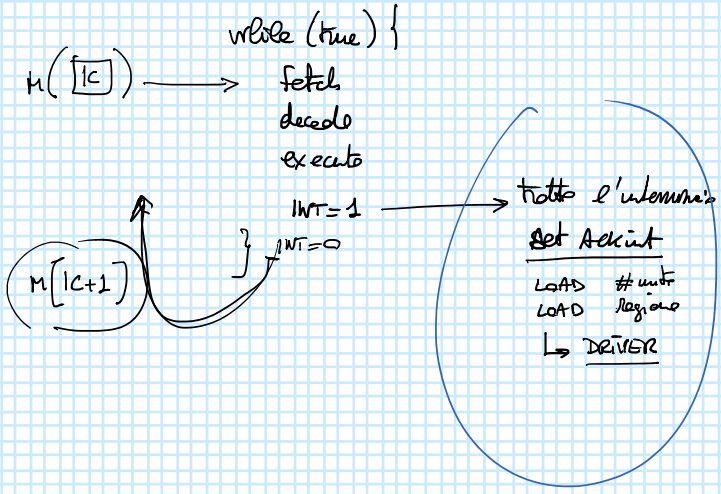
INTERPRETE ASM a livello FW





Ø. richiedere a M la lettura M[IC]

1. { attende la lettura
   switch (OPOP) {
   case "ADD" : ——
   case "SUB" : ——
   case "LOAD" : ——
   } { 2. ...
   3. ...
   4. (INTEQ) ... , Ø
   (INT=1) ... }



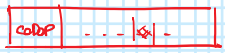
INTERPRETE FIRMWARE (1ª VERSIONE)

martedì 8 novembre 2016 11:48

IR instruction register

IR.COP
IR.REGA IR.REGB IR.REGC
IR.OFFSET

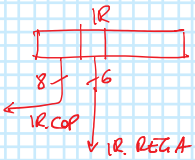
8 32

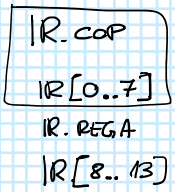


∅. "fetch" → OP, IC → IND, set RDY_m, 1.

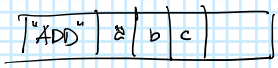
1. $(ACK_m, OR(ESITO) = 0 -)$ nop, 1
   $(=10)$ DATAIN → IR, reset ACK_m, 2 ← in DATAIN
   $(=11)$ ...., tratteccezioni

2. $(IR.COP, INT = "ADD", 0)$
   $REG[IR.REGA] + REG[IR.REGB] \to REG[IR.REGC]$,
   $IC+1 \to IC$, ∅





$(= "ADD", 1)$
REG[IR.REGA] + REG[IR.REGB] → REG[IR.REGC],
IC+1 → IC, tratt int
↑ inizio di una parte di micro-codice che realizza il trattamento dell'int

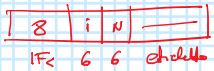


$(= "LOAD", \bar{\emptyset})$ "read" → OP, REG[IR.REGA] + REG[IR.REGB] → IND,
set RDY_m, reset ACK_m,

LOAD $R_{base}, R_{indice}, R_{dest}$
$M[R[base] + R[indice]] \to R[dest]$

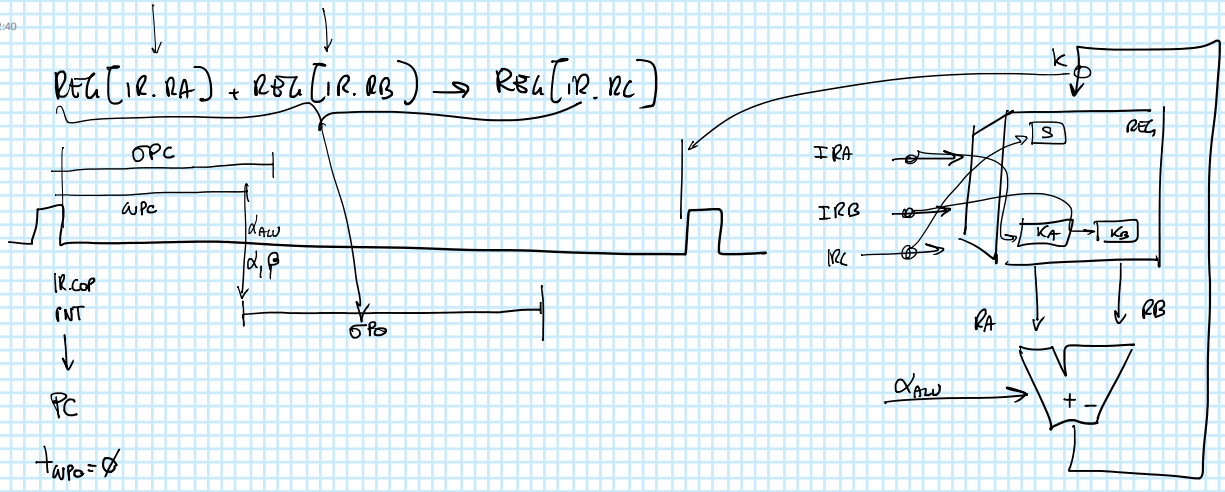
$(= "IF<", \bar{\emptyset})$ SEGNO (REG[IR.REGA] − REG[IR.REGB]) → S, 4

IF< R_i, R_N, loop

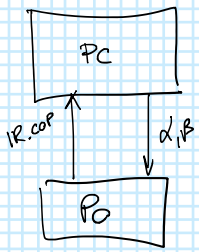


3. $(ACK_m, OR(ESITO) = 0 -)$ nop, 3 (INT)
   $(=100)$ DATA in → REG[IR.REGC], reset ACK_m, IC+1 → IC, ∅
   $(=101)$ " " " " , tratt int
   $(=11-)$ ..... , tratteccezioni

4. $(S, INT = 1\emptyset)$ IC + IR.etichetta → IC, ∅
   $(=11)$ IC + IR.etichetta → IC, tratt interruzioni
   $(=00)$ IC+1 → IC, ∅
   $(=01)$ IC+1 → IC, tratt interruzioni

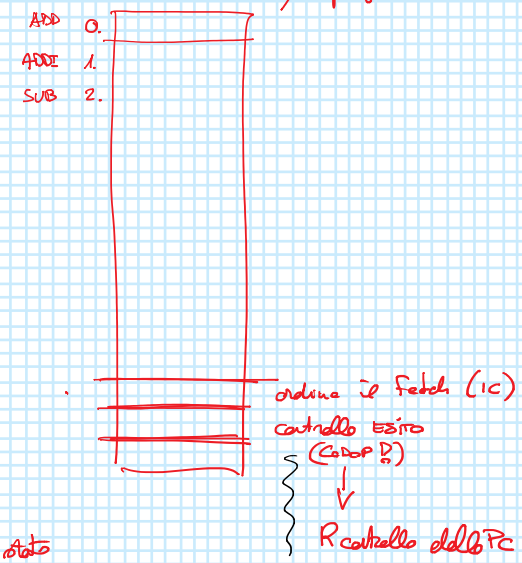


$t_{WPo} = \emptyset$



ADD	0
ADDI	1
SUB	2
⋮	
LOAD	15
⋮	

μ-programma



ch$\emptyset$. "fetch" → OP, IC → IND, set RDY_m, ch1

ch1. (ACK_m, OR(ESITO) = $\emptyset$ −) nop, ch$\emptyset$

(= 10) DATAIN → IR, ~~IR.COP~~ → RC

DATAIN.COP

registro di stato dello PC (# istruzioni che eseguiranno immediatamente dopo)

add$\emptyset$. (INT = 0) REG[IR.RA] + REG[IR.RB] → REG[IR.RC], IC+1 → ic, ch$\emptyset$.

(= 1) " " , halt int

sub$\emptyset$. (INT = 0) REG[IR.RA] − REG[IR.RB] → REG[IR.RC], IC+1 → ic, ch$\emptyset$

(= 1) " " , halt int

load$\emptyset$. "read" → OP, REG[IR.RA] + REG[IR.RB] → IND, set RDY_m, load1

load1. (ACK_m, OR(ESITO), INT = 0 − −) nop, load1

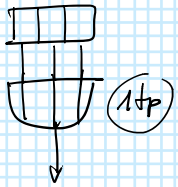
(= 100) DATAIN → REG[IR.RC], IC+1 → ic, ch$\emptyset$

(= 101) " " , halt int

(= 11 −) halt eccezioni

$ESITO_0$, $ESITO_1$, $ESITO_2$ = 000

OR(ESITO)



IF$\emptyset$. segue (REG[IR.RA] − REG[IR.RB]) → S, IF1

IF1. (S, INT = 10) IC + IR.etichetta → IC, ch$\emptyset$

(= 11) " , halt int

(= 00) IC+1 → IC, ch$\emptyset$

(= 01)

(=11)
(=00) IC+1 → IC ,
(=01) " ,