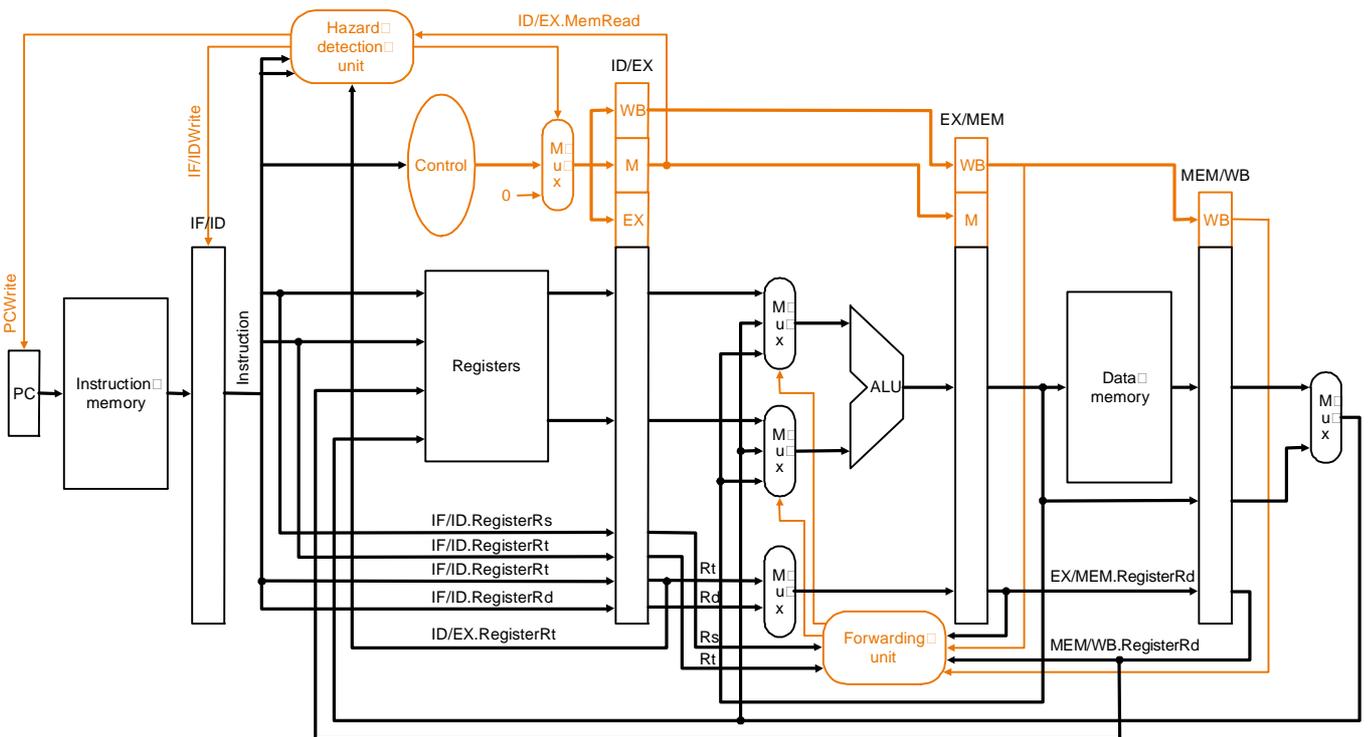


2. Si faccia riferimento al seguente diagramma, che illustra la nota implementazione del MIPS mediante pipeline riportando in particolare l'unità di propagazione verso lo stadio E e l'unità di gestione delle criticità. Si consideri il caso in cui un'istruzione lw sia seguita immediatamente da un'istruzione che ne utilizza il risultato, ad esempio:

lw \$t1, 20(\$t0)

add \$s2, \$t1, \$t2

Quando l'istruzione lw è nello stadio M, se l'istruzione add è nello stadio E quale dato viene propagato verso la ALU? Quale operando sostituisce? Come si può giustificare questo comportamento? [3]



3. Si consideri il seguente frammento di codice MIPS:

```
add    $t2, $t0, $s1
sub    $t2, $t0, $t0
lw     $t0, 20($t2)
sub    $t0, $t0, $t2
add    $s2, $t2, $t2
```

Si consideri l'implementazione con pipeline a 5 stadi (F: Fetch, D: Decode, E: Execute, M: Mem, W: Write-Back). Si chiede di:

- a) individuare in modo preciso tutte le dipendenze tra i dati
- b) tracciare il diagramma temporale delle istruzioni (indicando esplicitamente le eventuali propagazioni e, per ognuna di esse, quale dato è propagato) in ognuna delle seguenti ipotesi:
 - è disponibile un'unità di propagazione verso lo stadio E
 - è disponibile un'unità di propagazione verso lo stadio E ed una verso lo stadio M.

Nei diagrammi, si chiede di indicare il numero di cicli di penalità.

[4]

4. Si consideri un processore MIPS, dotato di pipeline a 5 stadi, che disponga di un'unità di propagazione solamente verso lo stadio E.

Il processore utilizza una cache primaria (distinta per i dati e le istruzioni), mentre non dispone di cache secondaria. La cache, che in caso di successo consente di accedere all'istruzione o al dato in un ciclo di clock, presenta le seguenti caratteristiche:

- percentuale di successo (hit rate): 90% per le istruzioni, 80% per i dati
- penalità di fallimento (sia in scrittura che in lettura): 10 cicli di clock

I salti incondizionati comportano in ogni caso una penalità di 1 ciclo di clock, mentre per i salti condizionati si utilizza la tecnica di predizione statica di "salto non effettuato", in cui l'esecuzione del salto (calcolo della condizione ed eventuale aggiornamento del program counter) avviene al terzo stadio.

Si assuma un carico di lavoro che prevede la seguente distribuzione delle istruzioni MIPS:

lw:	20 %
sw:	20 %
Tipo-R:	30 %
salti incondizionati:	10 %
salti condizionati:	20 %

tale che:

- il 50% delle istruzioni che seguono lw non ne utilizzano il risultato, il 20% sono istruzioni di TIPO-R che ne utilizzano il risultato, il 5% sono sw che utilizzano il risultato della lw per il calcolo dell'indirizzo, il 5% sono sw che utilizzano il risultato sia per il calcolo dell'indirizzo sia per immagazzinarlo in memoria, il rimanente 20% sono sw che utilizzano il risultato della lw solo per immagazzinarlo in memoria.
- il 10% delle istruzioni che seguono istruzioni di Tipo-R ne utilizzano il risultato nello stadio E, il 10% nello stadio M ed il rimanente 80% non ne utilizzano il risultato.
- il 60% dei salti condizionati non vengono effettuati.

Si calcoli il CPI (numero medio di cicli di clock per istruzione) ottenuto. Si ripeta il calcolo supponendo di disporre anche di una unità di propagazione verso M. [6]

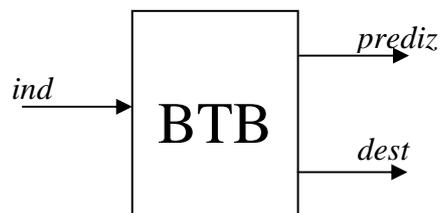
5. Nell'ambito dei processori che utilizzano la pipeline, si illustri brevemente il concetto di interruzione precisa e interruzione imprecisa. Con una pipeline dinamica, si illustri sinteticamente un'organizzazione dell'hardware del processore in grado di garantire che tutte le eccezioni siano precise. [3]

6. Con riferimento alla memoria virtuale, si illustri sinteticamente (4-5 righe al massimo) la funzione del TLB. Si consideri il caso in cui il numero di pagina virtuale non sia contenuto nel TLB: è possibile dire che ciò provoca in ogni caso un'eccezione per mancanza di pagina (page-fault)? Perché?

Nel caso in cui la pagina fisica associata alla pagina virtuale sia presente in memoria, si illustri come viene ottenuto l'indirizzo fisico a partire da quello virtuale, evidenziando i campi comuni e distinti. [4]

7. Descrivere sinteticamente il protocollo di handshaking. Enunciare due motivi fondamentali che favoriscono, per i bus di I/O (input/output), l'adozione della temporizzazione asincrona al posto di quella sincrona. [3]

8. Nel diagramma successivo viene riportato il datapath della pipeline a 5 stadi del MIPS, in cui per il salto condizionato *beq* il calcolo della destinazione (la quale, si ricorda, dipende da PC e offset) ed il calcolo della condizione di salto vengono entrambi effettuati nel quarto stadio. Si chiede di modificare il datapath introducendo un BTB per la predizione dinamica di salto. Il BTB da inserire può essere rappresentato nel modo seguente:



Il BTB restituisce nell'uscita *prediz*:

- 0 se l'indirizzo in ingresso *ind* non si trova nel BTB oppure, trovandosi nel BTB, la predizione di salto è negativa;
- 1 se l'indirizzo *ind* si trova nel BTB e la predizione di salto è positiva.

Nel caso in cui *prediz* sia 1, l'uscita *dest* restituisce l'indirizzo di destinazione predetto.

Si assuma che, nel BTB, l'etichetta per identificare i salti condizionati sia il loro indirizzo completo. [3]

