



## PIPELINED CPU

Με υποσύνολο της αρχιτεκτονικής συνόλου  
εντολών: CHARIS (CHAnia Risc Instruction Set)

Ονοματεπώνυμο: ΠΕΤΑΚΑΚΗΣ ΓΕΩΡΓΟΣ

ΑΜ: 2021030127

Ονοματεπώνυμο: ΚΥΡΙΑΚΑΚΗΣ ΙΩΑΝΝΗΣ ΓΕΩΡΓΙΟΣ

ΑΜ: 2020030103

Στο 3ο μέρος του Project μάς ζητήθηκε να τροποποιήσουμε την λειτουργικότητα του επεξεργαστή χρησιμοποιώντας Pipeline αντί για επεξεργαστή πολλών κύκλων.

Για την επίτευξη του Pipeline διατηρήσαμε τους Registers, οι οποίου βρίσκονται ενδιάμεσα σε κάθε ένα από τα 5 Stages του επεξεργαστή και φροντίσαμε οι έξοδοι του κάθε stage να περνάνε ως είσοδοι στους 3 ενδιάμεσους Pipeline Registers (ID/EXEC, EXEC/MEM, MEM/WB) έτσι ώστε να γίνεται μετάβαση των signals των καταχωρητών και των control signals του Control Unit κατά την διάσχιση από βαθμίδα σε βαθμίδα σε βαθμίδα κατά την εκτέλεση της εκάστοτε εντολής.

Με αυτόν τον τρόπο είναι εφικτό να γίνονται fetch νέες εντολές σε κάθε νέο κύκλο ρολογιού και η εκτέλεσή τους αντίστοιχα να γίνεται με διαφορά ενός κύκλο ρολογιού στο τέλος του WB stage.

Παρ'όλα αυτά, για να εξασφαλίσουμε την ομαλή λειτουργία του Pipeline, έπρεπε να λάβουμε υπόψιν μας τα διάφορα Hazards που μπορεί να προξενήσουν προβλήματα κατά την ανάγνωση και το Write Back των καταχωρητών.

Οι περιπτώσεις των Hazards που αντιμετωπίσαμε ήταν 2:

Data Hazards:

Συμβαίνουν όταν ο Rd της προηγούμενης εντολής χρησιμοποιείται ως Rs ή Rt operand της επόμενης ή της μεθεπόμενης εντολής

πχ.

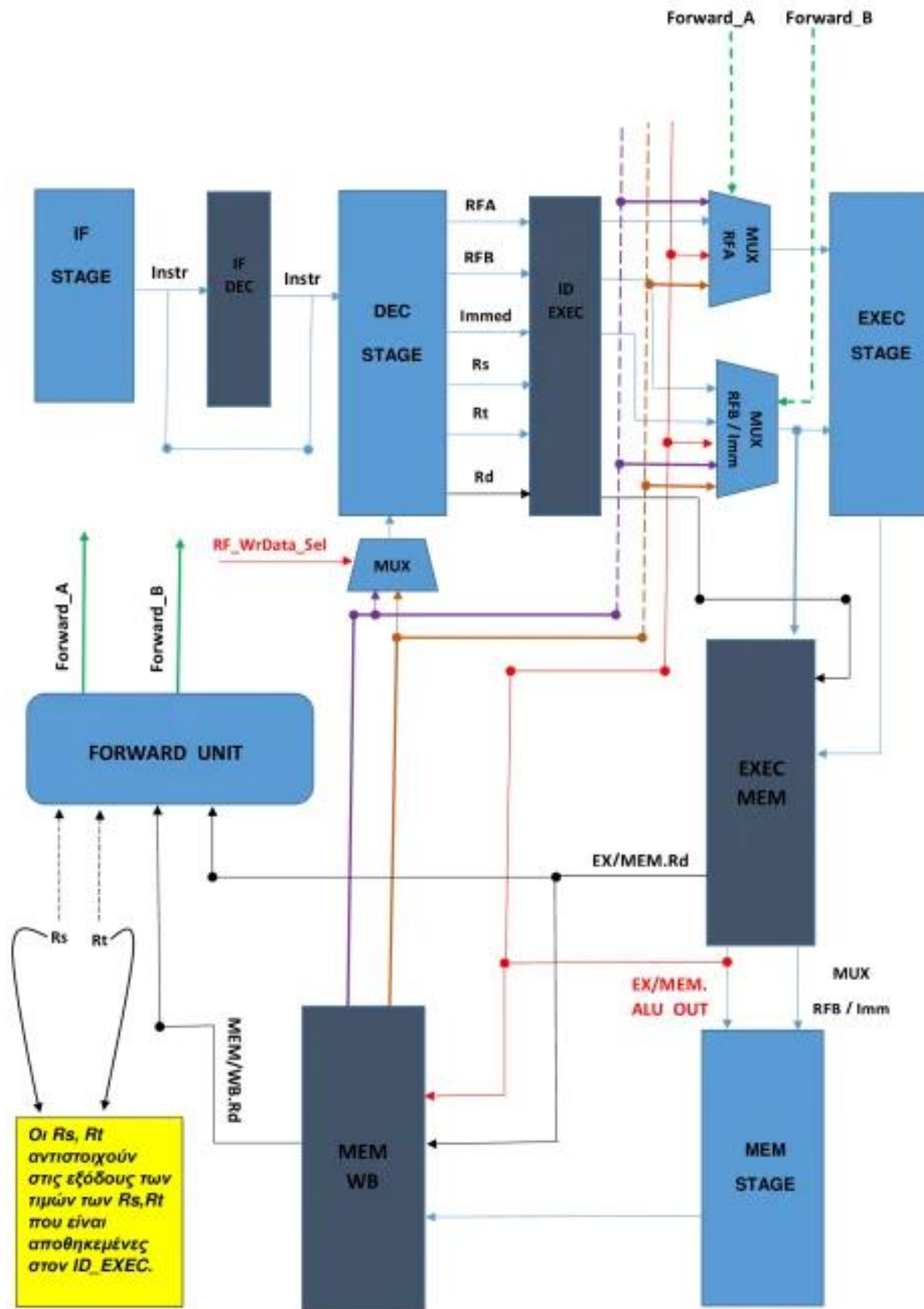
add R1,R2,R3

add R4,R1,R5

add R6,R7,R1 (R1 with a different colour)

Πρέπει να φροντίσουμε ότι η τιμή του του R1 που γίνεται στην πρώτη εντολή add στο παράδειγμα μας, θα είναι διαθέσιμη κατά το EXEC stage(όπου γίνονται οι πράξεις στην ALU) της επόμενης και της μεθεπόμενης εντολής, διαφορετικά θα έχουμε λάθος αποτελέσματα.

Αυτό το επιτυγχάνουμε μέσω του Forwarding Unit το οποίο φαίνεται στην παρακάτω εικόνα.



Στην αμέσως επόμενη εντολή συγκρίνουμε εάν η τιμή του Rs ή του Rt είναι ίση με την τιμή του EX/MEM.Rd. Εάν  $R_s = EX/MEM.rd$  τότε το Forwarding Unit θα στείλει κατάλληλη έξοδο μέσω του control signal ForwardA έτσι ώστε από τον MUX\_RFA να περάσει η τιμή της ALU που είχε υπολογιστεί στον προηγούμενο κύκλο, η οποία αντιστοιχεί στην τιμή του Rf<sub>d</sub>.

Επιπλέον, από τον MUX\_RFB θα πρέπει να περάσει η τιμή του Rt από το Register File εφόσον η συνθήκη είναι αληθής μόνο για τον Rs. Αυτό το επιτυγχάνουμε δίνοντας κατάλληλη τιμή στο control signal ForwardB. Έτσι, με αυτόν τον τρόπο η τιμή του Rf<sub>d</sub> της προηγούμενης εντολής είναι διαθέσιμη προτού γίνει το WB της.

Στην περίπτωση όπου  $R_t = EX/MEM.rd$  οι μόνες διαφορές είναι ότι το control signal είναι το ForwardB και ο multiplexer που μας ενδιαφέρει είναι ο MUX\_RFB και το ForwardA θα φροντίσει έτσι ώστε από τον MUX\_RFA να περάσει η κανονική τιμή του Rt από το Register File.

Στην μεθεπόμενη εντολή συγκρίνουμε εάν η τιμή του Rs ή του Rt είναι ίση με την τιμή του MEM/WB.Rd. Με την ίδια ακριβώς λογική, εάν  $R_s = MEM/WB.Rd$  τότε το ForwardA control signal θα φροντίσει έτσι ώστε να περάσει από τον MUX\_RFA η τιμή της ALU η οποία είχε υπολογιστεί δύο κύκλους πριν και η τιμή της παρέμεινε αποθηκευμένη στον MEM/WB Pipeline Register. Το ForwardB θα φροντίσει έτσι ώστε ο MUX\_RFB να περάσει την τιμή από του Rs από το Register File.

Αντίστοιχα, εάν  $R_t = MEM/WB.Rd$ , τότε το ForwardB θα φροντίσει έτσι ώστε η αποθηκευμένη τιμή της ALU να περάσει από τον MUX\_RFB και το ForwardA θα φροντίσει έτσι ώστε από τον MUX\_RFA να περάσει η κανονική τιμή του Rt από το Register File.

Stall unit :

Το stall unit χρησιμοποιείται κυρίως για ακραίες καταστάσεις όπου δεν μπορούμε παρά να περιμένουμε για κάτι να γίνει, έτσι ώστε να προχωρήσουμε το πρόγραμμα φυσιολογικά. Μία τέτοια κατάσταση δημιουργείται με την εντολή load word (lw).

```
lw $r2,0($r10)
```

```
li $r2,$r5,$r9
```

Για παράδειγμα στον παραπάνω κώδικα assembly, δεν μπορούμε παρά να περιμένουμε την πληροφορία στην μνήμη με την τιμή του καταχωρητή \$r10 να περάσει στον καταχωρητή \$r2, για την επόμενη εντολή (li). Δεν είναι μία απλή μαθηματική πράξη για να εφαρμόσουμε bypass σε κάποιον ξεχωριστό pipeline register, αλλά πρέπει να περιμένουμε, να βγει η τιμή που θέλουμε από την RAM.