



ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ  
ΗΡΥ 302: ΟΡΓΑΝΩΣΗ ΥΠΟΛΟΓΙΣΤΩΝ  
ΕΑΡΙΝΟ ΕΞΑΜΗΝΟ 2021-2022  
Παρεμβάσεις: Κυπριανός Παπαδημητρίου

## Εργασία #2: Σχεδίαση επεξεργαστή πολλαπλών κύκλων και μετατροπή του σε pipeline

Χωρίζεται σε 2 φάσεις

**4<sup>η</sup> φάση: χρόνος ολοκλήρωσης 10 ημέρες**

**«Σχεδίαση επεξεργαστή πολλαπλών κύκλων»**

- Μελετήστε πρώτα καλά όλη την εκφώνηση -

### Σκοπός της 4<sup>ης</sup> φάσης

Μετατροπή του επεξεργαστή ενός κύκλου που σχεδιάσατε στην Εργασία#1 σε επεξεργαστή πολλαπλών κύκλων.

### Προαπαιτούμενα

Λειτουργική σχεδίαση επεξεργαστή ενός κύκλου

### Διεξαγωγή

#### **A) Μετατροπή του datapath του επεξεργαστή ενός κύκλου σε datapath πολλαπλών κύκλων**

Μελετήστε το datapath που κατασκευάσατε στην Εργασία#1 και προσθέστε καταχωρητές μεταξύ των βαθμίδων όπου χρειάζεται, ώστε να κρατάτε τις τιμές για κάποια σήματα που παράγονται από μια βαθμίδα, τα οποία χρειάζεται να χρησιμοποιηθούν σε επόμενη βαθμίδα. Μπορείτε να κάνετε όσες επιπλέον αλλαγές κρίνετε απαραίτητες, είτε εσωτερικά σε επίπεδο σχεδίασης του datapath είτε σε επίπεδο διεπαφής. Ονομάστε το αρχείο σας **DATAPATH\_MC.vhd**.

#### **B) Σχεδιασμός και υλοποίηση της μονάδας ελέγχου (control)**

Σχεδιάστε τη μονάδα ελέγχου, η οποία θα είναι μια Μηχανή Πεπερασμένων Καταστάσεων (FSM) που θα ελέγχει τη ροή εκτέλεσης της κάθε εντολής γεννώντας τα απαιτούμενα σήματα ελέγχου σε κάθε κύκλο ρολογιού. Η FSM έχει σαν είσοδο την εντολή (το `opcode`), πιθανώς flags όπως το `ALUzero`, κ.λ.π., και σαν εξόδους όλα τα σήματα ελέγχου του Datapath. Διερευνήστε τι τύπου θα είναι η FSM που θα φτιάξετε, Moore ή Mealy; Η FSM θα αντικαταστήσει τη μονάδα ελέγχου που είχατε φτιάξει για τον επεξεργαστή ενός κύκλου (το control της Εργασίας #1), την οποία τώρα πρέπει να αφαιρέσετε από την σχεδίασή σας. Ονομάστε την FSM σας **CONTROL\_MC.vhd**.

### Γ) Ολοκλήρωση και έλεγχος ορθότητας

Συνδέστε το datapath με το control για να υλοποιήσετε την πλήρη λειτουργία ενός επεξεργαστή πολλαπλών κύκλων. Ονομάστε το αρχείο σας **PROCESSOR\_MC.vhd**. Η κύρια μνήμη θα πρέπει να βρίσκεται εκτός αυτού του module.

Δείξτε την ορθότητα της σχεδίασης σας με τα προγράμματα αναφοράς της Εργασίας#1 και επίσης δημιουργήστε τουλάχιστον ένα ακόμη δικό σας πρόγραμμα αναφοράς το οποίο θα περιλαμβάνει όλες τις εντολές του ISA τουλάχιστον από μία φορά.

## 5<sup>η</sup> φάση: χρόνος ολοκλήρωσης 12 ημέρες

### «Σχεδίαση επεξεργαστή pipeline»

- Μελετήστε πρώτα καλά όλη την εκφώνηση -

#### Σκοπός της 5<sup>ης</sup> φάσης

Μετατροπή του επεξεργαστή πολλαπλών κύκλων σε επεξεργαστή pipeline.

#### Προαπαιτούμενα

Λειτουργική σχεδίαση επεξεργαστή ενός κύκλου/πολλαπλών κύκλων

#### Διεξαγωγή

##### A) Αλλαγές στο datapath του επεξεργαστή πολλαπλών κύκλων

Προσθέστε τους κατάλληλους καταχωρητές pipeline όπου κρίνετε ότι χρειάζεται ώστε για όλες τις βαθμίδες του datapath που υλοποιήσατε στην 4<sup>η</sup> φάση η έξοδος της κάθε βαθμίδας να γίνεται είσοδος στην επόμενη. Λάβετε επιπρόσθετα υπόψη ότι χρειάζεται προώθηση (forwarding) ή/και stalls για την αντιμετώπιση των κινδύνων δεδομένων (data hazards). Βάσει αυτού υλοποιήστε τις απαιτούμενες επεκτάσεις στη σχεδίαση σας. Ονομάστε το αρχείο σας **DATAPATH\_PIPELINE.vhd**.

##### B) Σχεδιασμός και υλοποίηση της μονάδας ελέγχου (control)

Σχεδιάστε τη μονάδα ελέγχου ώστε να παράγει τα απαιτούμενα σήματα ελέγχου για κάθε εντολή. Μπορείτε να χρησιμοποιήσετε είτε τη μονάδα ελέγχου του επεξεργαστή ενός κύκλου (Εργασία#1) ως σημείο εκκίνησης, είτε τη μονάδα ελέγχου του επεξεργαστή πολλαπλών κύκλων (4<sup>η</sup> φάση). Ανάλογα με τη σχεδίαση της μονάδας ελέγχου που θα χρησιμοποιήσετε ως σημείο εκκίνησης θα χρειαστεί να κάνετε και τις απαιτούμενες αλλαγές. Ακολουθήστε τη θεωρία για να σχεδιάσετε το control για τον pipeline επεξεργαστή σύμφωνα και με τα παραδείγματα που παρουσιάστηκαν στις διαλέξεις. Λάβετε υπόψη ότι υπάρχουν και data hazards μεταξύ εντολών που μπορούν να αντιμετωπιστούν με forwarding ή/και με stalls, και πιθανό να χρειαστούν επιπλέον επεκτάσεις στο control (ανάλογα με τη συνολική σχεδίαση του επεξεργαστή σας). Αγνοήστε τα control hazards. Ονομάστε το αρχείο σας **CONTROL\_PIPELINE.vhd**.

Προτείνεται να βασιστείτε στην μονάδα ελέγχου του επεξεργαστή ενός κύκλου ως σημείο εκκίνησης.

##### Γ) Ολοκλήρωση και έλεγχος ορθότητας

Συνδέστε το datapath με το control ώστε να υλοποιήσετε την πλήρη λειτουργία ενός pipeline επεξεργαστή. Ονομάστε το αρχείο σας **PROCESSOR\_PIPELINE.vhd**.

Δείξτε την ορθότητα της σχεδίασης σας με ένα δικό σας πρόγραμμα αναφοράς τα οποίο θα περιλαμβάνει μόνο τις εντολές **li**, **lw**, **sw**, **add** πολλές φορές ώστε να φανεί η λειτουργία του pipeline και η αντιμετώπιση των κινδύνων δεδομένων.

## Αναλυτική περιγραφή παραδοτέων Εργασίας #2

### - Ακολουθήστε τις οδηγίες όπως είναι ακριβώς -

Ο τελικός φάκελος που θα υποβάλλετε θα φέρει το “b\_”, το επίθετο, και τον αριθμό μητρώου σας π.χ. [b\\_christodoulou\\_2019123456](#). **Κάνετε zip τον φάκελο** (συγκεκριμένα zip, και όχι rar ή 7z ή οτιδήποτε άλλο). Προσοχή: το επίθετο σας να είναι σε greeklish και επίσης παρεμβάλλετε το σύμβολο “\_” μεταξύ του επιθέτου και του ΑΜ. Εσωτερικά δημιουργείτε 3 υποφακέλους: REPORT, SOURCES, WAVEFORMS, και οι 3 με κεφαλαία γράμματα. Προσέξτε την ονομασία του φακέλου να είναι σωστή και πλήρης. Αν υπάρχει λάθος ο κωδικός σας δεν θα βαθμολογηθεί. Με την αποσυμπίεση του .zip να προκύπτει ένας μόνο φάκελος π.χ. [b\\_christodoulou\\_2019123456](#).

#### 1. Αναφορά (μέχρι 6 σελίδες, PDF)

Βάλτε την στον υποφάκελο REPORT. Η αναφορά δεν θα περιλαμβάνει κώδικα, εκτός αν είναι code snippet μέχρι 10 γραμμές και ακολουθείται από τον αντίστοιχο σχολιασμό. Η αναφορά πρέπει να περιλαμβάνει block diagrams που να αποτυπώνουν καλά και περιεκτικά τη δουλειά σας, π.χ. κάποια σήματα και συνδέσεις που θεωρείτε σημαντικά να δείξετε, ή, νέα modules που φτιάξατε που δεν αναφέρονται ρητά στην εκφώνηση.

#### 2. Κώδικας VHDL

Εντός του φακέλου SOURCES δημιουργήστε 2 φακέλους με τα παρακάτω ονόματα (όλοι με κεφαλαία γράμματα και ακριβώς 2 φάκελοι - μην αλλάξετε τα ονόματα). Όλα τα αρχεία .vhd που έχετε φτιάξει, μαζί με τα testbenches και τα libraries, θα τα βάλετε μέσα σε αυτούς τους υποφακέλους. Η υποβολή των testbenches είναι υποχρεωτική και χωρίς αυτά η υποβολή θεωρείται ελλιπής.

#### Περιεχόμενα φακέλου SOURCES

1. Φάκελος **MC**: όλα τα αρχεία .vhd για τον επεξεργαστή πολλαπλών κύκλων
  2. Φάκελος **PIPELINE**: όλα τα αρχεία .vhd για τον pipeline επεξεργαστή
- Σε κάθε φάκελο βάζετε μόνο αρχεία με προέκταση .vhd (το .vhd με μικρά γράμματα). Αν δημιουργήσατε και κάποια νέα δική σας library να την συμπεριλάβετε και αυτή στον αντίστοιχο φάκελο.
  - Από τους 2 παραπάνω φακέλους, MC και PIPELINE, κανένας να μην περιέχει υποφακέλους.
  - Κάθε φάκελος από τους παραπάνω θα περιέχει μέσα ένα (1) μόνο testbench : το τελικό με το οποίο ελέγξατε την αντίστοιχη σχεδίαση, MC ή PIPELINE.
  - Όλα τα καινούργια αρχεία που φτιάξατε για μια συγκεκριμένη υλοποίηση θα βρίσκονται στον αντίστοιχο φάκελο, MC ή PIPELINE.
  - **Δεν χρειάζεται να υποβάλλετε τα .vhd αρχεία που είχατε υποβάλλει στην Εργασία#1, εκτός αν τα τροποποιήσατε.** Σε αυτή την περίπτωση δημιουργήστε ένα επιπλέον φάκελο **μέσα στον φάκελο SOURCES με όνομα SUBMODULES** και βάλτε τα εκεί μαζί με τα testbenches τους. Κάθε .vhd αρχείο στο φάκελο SUBMODULES θα πρέπει να περιέχει σε σχόλια μια σύντομη περιγραφή και αιτιολόγηση των αλλαγών που κάνατε, και επίσης σε ένα αρχείο .txt.
  - Για τα τροποποιημένα από την 1<sup>η</sup> Εργασία .vhd modules: θα περιέχονται 2 φορές, μια φορά μέσα στον MC/PIPELINE, και μια φορά μέσα στο SUBMODULES.
  - Αν έχετε πολλά rom.data, βάλτε τα στον φάκελο SUBMODULES. Και στον WAVEFORMS

- μπορείτε να τα βάλετε. Π.χ. rom1.data, rom2.data, rom3.data κλπ.
- Ένα (1) μόνο αρχείο rom.data μπορείτε να το βάλετε μέσα στους φακέλους MC και PIPELINE, αυτό που θα δείξετε στην εξέταση.
  - .vhdl αρχεία που δημιουργήσατε στην 1<sup>η</sup> εργασία και τα χρησιμοποιήσατε όπως ακριβώς είναι χωρίς αλλαγές και στην 2<sup>η</sup> εργασία π.χ. ALU, RF, χρειάζεται να τα συμπεριλάβετε και πάλι στους φακέλους MC, PIPELINE. **Μόνο** σε αυτούς τους φακέλους, **όχι** στον SUBMODULES. Αν όμως αλλάξατε κάποιο αρχείο, π.χ. αλλαγή στην ALU έστω και μικρή, τότε να το συμπεριλάβετε **και στο SUBMODULES**.

### **3. Κυματομορφές Προσομοίωσης και Προγράμματα Αναφοράς**

Στον φάκελο WAVEFORMS βάλετε τις κυματομορφές προσομοίωσης. Θα σας βοηθήσει η χρήση του Waveform Configuration File (.wcfg) που προσφέρει η Xilinx, για ν' αποθηκεύετε τα σήματα σε αρχείο. Στον ίδιο φάκελο θα βρίσκονται και τα προγράμματα αναφοράς που δημιουργήσατε, συνοδευόμενο το καθένα από ένα text αρχείο που θα περιλαμβάνει τις assembly εντολές του προγράμματος αναφοράς. Στον φάκελο WAVEFORMS δεν θα υπάρχουν υποφάκελοι. Δεν χρειάζονται προσομοιώσεις της προηγούμενης Εργασίας#1. Σημείωση: το αρχείο .wcfg το κάνετε "Load" μετά από compile, για να εμφανίζονται αυτόματα τα σήματα που θέλετε να παρακολουθείτε στην προσομοίωση.