

# ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

## Αρχιτεκτονική Υπολογιστών

### Εργαστηριακή Άσκηση 1 (25% εργαστηριακού βαθμού)

Ομαδική εργασία σε ομάδες τριών (3) μελών

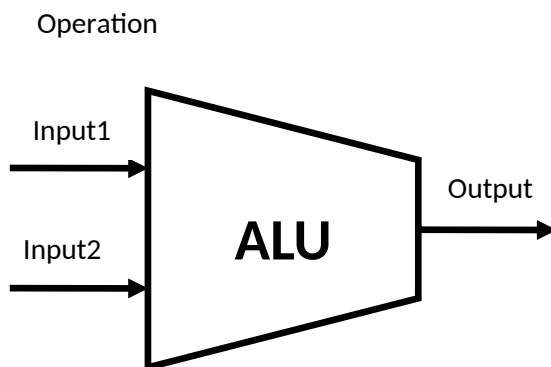
Ημερ/νία παράδοσης: 13 Απριλίου, 2022, 21:00 (μέσω e-class)

Ημερ/νία εξέτασης: 14 Απριλίου, 2022, ώρες εργαστηρίου

- Για όλα τα προβλήματα να πραγματοποιήσετε functional και timing simulation για το πρόβλημα ώστε να επαληθεύσετε την σωστή λειτουργία του κυκλώματος.
- Για κάθε κύκλωμα θα πρέπει να παραδίδετε το αρχείο VHDL, το RTL διάγραμμα του κυκλώματος (σε μορφή εικόνας) καθώς και το functional και timing simulation (σε μορφή εικόνας).
- Χρησιμοποιείτε τυχαία σήματα για τα input των κυκλωμάτων σας με την προϋπόθεση να απεικονίζονται όλες οι καταστάσεις του εκάστοτε πίνακα αληθείας.
- Να έχετε (λίγα αλλά περιεκτικά) σχόλια στην VHDL όταν χρειάζεται να εξηγήσετε «δύσκολα» κομμάτια κώδικα, ρουτίνες, κλπ.
- Γράψτε (σε σχόλια στην αρχή του προγράμματος) **τα ονόματά σας και ΑΜ.**

α) Θα κατασκευάσετε με VHDL σε **structural κώδικα** (όχι σε behavioral, πρέπει να σχεδιάσετε πύλη-προς-πύλη το εργαστήριο αυτό χωρίς να χρησιμοποιήσετε δομικά στοιχεία του Quartus αλλά μόνο αυτά που θα ορίσετε εσείς) μια ALU για κάποιον επεξεργαστή.

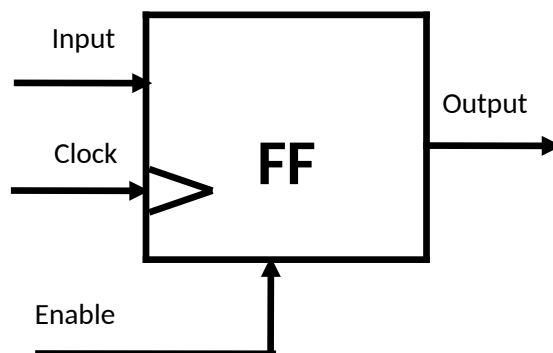
Το κύκλωμα θα παίρνει σαν είσοδο δύο 16 bitους αριθμούς και ένα 3bitο σήμα το οποίο δηλώνει την πράξη που πρέπει να εκτελεστεί, και θα βγάζει το 16 bito αποτέλεσμα.



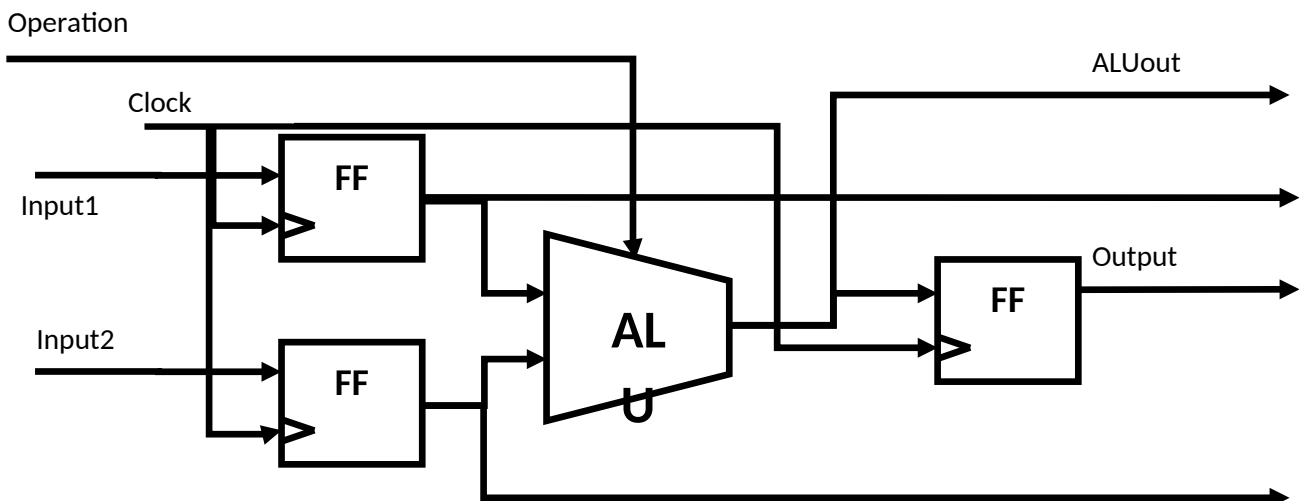
Εντολή	Περιγραφή	Πράξη	Operation
ADD	Πρόσθεση	$O = I1 + I2$	000
SUB	Αφαίρεση	$O = I1 - I2$	001
AND	Λογικό AND (bit προς bit)	$O = I1 \text{ AND } I2$	010
OR	Λογικό OR (bit προς bit)	$O = I1 \text{ OR } I2$	011
GEQ	Μεγαλύτερο- Ίσο (Greater Equal)	$O = (I1 \geq 0)$	100
NOT	Not	$O = (I1 == 0)$	101

- Οι αριθμοί θα είναι σε μορφή συμπλήρωμα ως προς 2, συνεπώς οι πράξεις θα πρέπει να γίνονται όπως ορίζονται στο πακέτο *ieee.std\_logic\_signed.all*
- Η εντολή GEQ βάζει στο Output τον αριθμό 1 εάν το Input1 είναι μη αρνητικός ( $\geq 0$ ) ή τον αριθμό 0 εάν είναι αρνητικός ( $< 0$ ). Γι' αυτό μας βοηθάει το πιο σημαντικό ψηφίο του.
- Η εντολή NOT βάζει στο Output τον αριθμό 1 (15 bit 0 και το LSB 1) εάν το Input1 είναι 0 ή τον αριθμό 0 εάν διάφορος του 0. **ΔΕΝ** αλλάζει τον αριθμό bit προς bit.

β) Κατασκευάστε ένα 16 bit Flip Flop σε structural κώδικα (πύλη προς πύλη) με όνομα reg σε VHDL. Το κύκλωμα θα παίρνει είσοδο το ρολόι, και ένα Enable σήμα. Η εγγραφή θα γίνεται μόνο όταν το Enable είναι 1. Για τις μονές ομάδες το κύκλωμα θα είναι αρνητικά ακμοπυροδότητο, ενώ για τις ζυγές το κύκλωμα θα είναι θετικά ακμοπυροδότητο.



γ) Χρησιμοποιώντας τα κομμάτια που φτιάξατε παραπάνω, να κατασκευάσετε το παρακάτω κύκλωμα σε VHDL.



➔ Τι παρατηρείτε στο simulation για το Operation και το χρόνο που αυτό δίνεται; Γράψτε με comment την απάντησή σας στο main VHDL αρχείο

## Αποστολή και Παρατηρήσεις:

- Στην αρχή των αρχείων παράδοσης, γράψτε σε comments τα ονόματα της ομάδας σας και τα ΑΜ σας.
- Καλούνται οι φοιτητές κάθε ομάδας να αποθηκεύσουν το έργο τους σε αρχείο που φέρει το όνομα της ομάδας τους, π.χ.: OM-01.zip
- Η παράδοση της άσκησης θα πρέπει να γίνει μέσω του e-class από την επιλογή Εργασίες Φοιτητών σύμφωνα με το παρακάτω υπόδειγμα.
- Η παρουσία σας στην εξέταση είναι υποχρεωτική για την βαθμολόγησή σας.
- Κατά την εξέταση της άσκησης είναι απαραίτητη η επίδειξη της φοιτητικής σας ταυτότητας για την ταυτοποίηση σας.
- Ανταλλαγή πληροφορίας μεταξύ των ομάδων επιτρέπεται όσο προάγει την ουσιαστική γνώση. Οποιαδήποτε παράβαση (αντιγραφή μεταξύ των ομάδων, παράδοση άσκησης από άτομο που δεν κατέχει το περιεχόμενο) οδηγεί σε μηδενισμό του εργαστηριακού βαθμού όλων των εμπλεκομένων (++).

Εργασίες μαθήματος ⓘ

Τίτλος:	Στοιχεία εργασίας
Περιγραφή:	Εργαστηριακή Άσκηση 1
Προβλεπόμενη έναρξη:	Δείτε την έναρξη που έχει αναρτηθεί στο Έγγραφο
Προβλεπόμενη υποβολή:	01-03-2010
Τέλος εργασίας:	24-03-2010 (απομένουν 23 ημέρες)
	Ομάδική

Αρχείο:	Υποβολή
Σχόλιο:	<div>Αποστολή</div>

Αν στείλετε κάποιο άλλο αρχείο, το αρχείο που υπάρχει αυτή τη στιγμή θα διαγραφεί και θα αντικατασταθεί με το νέο.

Επιστροφή

Α) Επισύναψη

Υποβολή