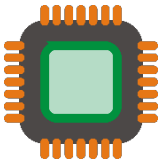
Τρίτη 26 Μαρ 2019

ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

ΕΡΓΑΣΤΗΡΙΟ ΜΙΚΡΟΕΠΕΞΕΡΓΑΣΤΩΝ

& ΥΛΙΚΟΥ

«ΗΡΥ 591 ΑΝΑΔΙΑΤΑΣΣΟΜΕΝΑ ΨΗΦΙΑΚΑ ΣΥΣΤΗΜΑΤΑ»



Αναφορά εργασίας εξαμήνου milestone 1

Ομάδα Εργασίας : LAB59140120
Μποκαλίδης Αναστάσιος 2014030069
Χατζηπέτρος Αλέξανδρος 2013030151

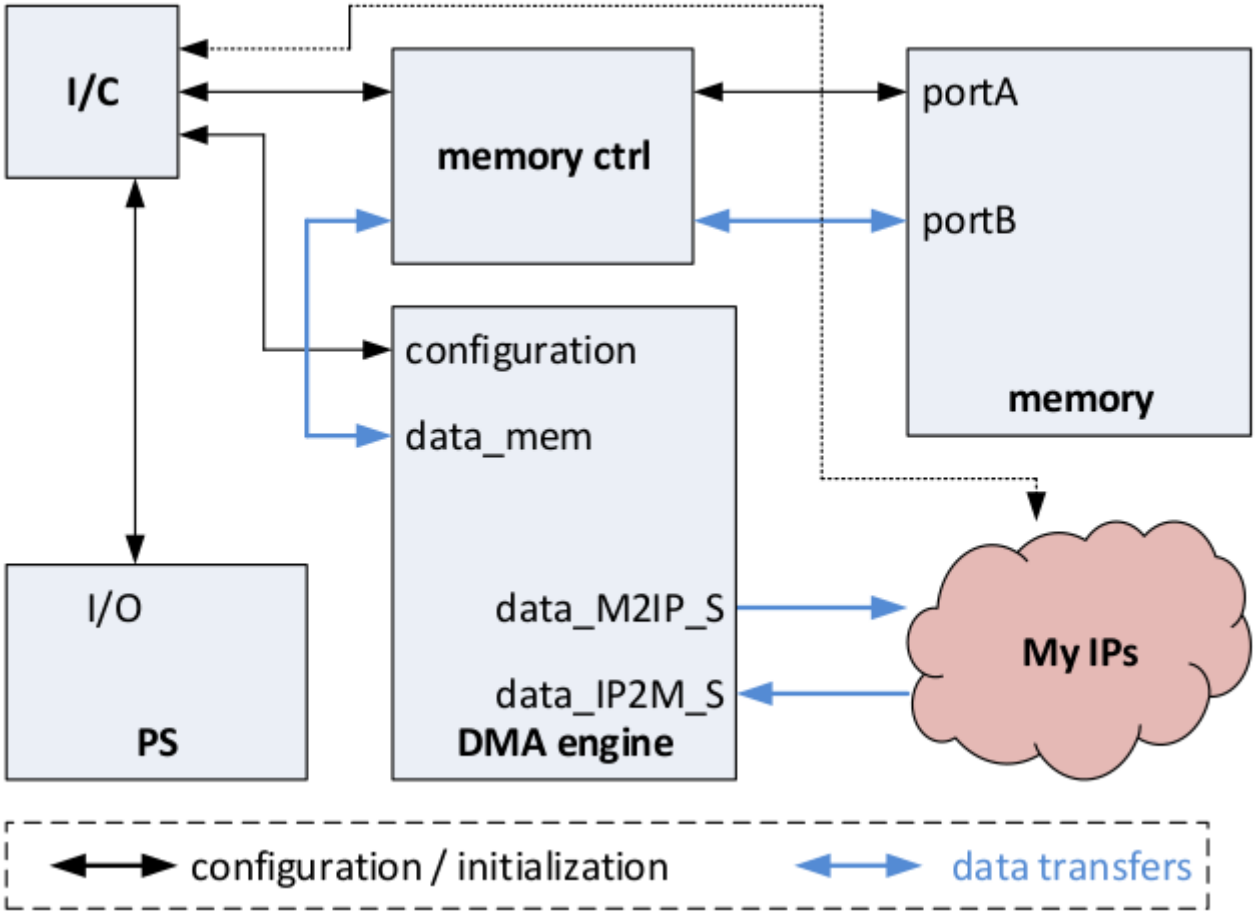
Σκοπός εργαστηριακής άσκησης

Σκοπός της εν λόγω εργασίας είναι η σχεδίαση και υλοποίηση σε hardware ενός αλγορίθμου, χρησιμοποιώντας την γλώσσα περιγραφής υλικού VHDL και την γλώσσα υψηλού επιπέδου για την περιγραφή υλικού (High-Level Synthesis HDLs). Την σχεδίαση της παραπάνω λογικής θα την συνδέσουμε σε έναν ενσωματωμένο επεξεργαστή (embedded processor) και κατόπιν θα μελετήσουμε την απόδοση των κυκλωμάτων του συστήματος με προσομοίωση.

Κατά την διάρκεια εκτέλεσης μιας εργασίας σε programmable logic (PL) μια γνωστή προσέγγιση είναι η χρήση Direct Memory Engines (DMAe). Βασική ιδέα είναι ο επεξεργαστής να προγραμματίσει το DMAe με συγκεκριμένες δομές (descriptors) στα οποία θα αναφέρεται πόσα δεδομένα και από ποιά διεύθυνση να τα διαβάσει ή πόσα δεδομένα περιμένει και σε ποιά διεύθυνση να τα γράψει.

Περιγραφή άσκησης

Για την υλοποίηση της παραπάνω εργασίας μας δόθηκε η παρακάτω δομή ενός System-on-Chip (reconfigurable SoC-rSoC) που χρησιμοποιεί ένα DMAe για τη μεταφορά δεδομένων από και προς τη μνήμη, στην συγκεκριμένη δομή φανερώνονται σχηματικά και τα modules που την απαρτίζουν ως εξής :



Επιπλέον μας δόθηκε υλοποίηση-κώδικας σε VHDL ως reference την οποία επεκτείναμε (My IPs) ώστε η λογική μας να μπορεί να διαβάσει δεδομένα από τη μνήμη, να τα επεξεργαστεί και στη συνέχεια να τα γράψει πίσω στη μνήμη σύμφωνα με την σειρά των βημάτων που μας δόθηκε στην εκφώνηση.

My IPs

Συνεχίζοντας για να υλοποιήσουμε τα παραπάνω αλλάξαμε τον κώδικα VHDL και δημιουργήσαμε μια FIFO ουρά 64 θέσεων (32bits μέγεθος ανά λέξη) η οποία λειτουργεί με τα σήματα που μας δόθηκαν στην εκφώνηση. Συγκεκριμένα ορίσαμε ένα array τύπου BYTE_INFO_TYPE με 64 θέσεις και κάθε θέση έχει ένα vector μεγέθους 32bit. Η FIFO φαίνεται με το σήμα stream_data_fifo στον κώδικα στο module slave.

Διαδιακασία εγγραφής στη FIFO

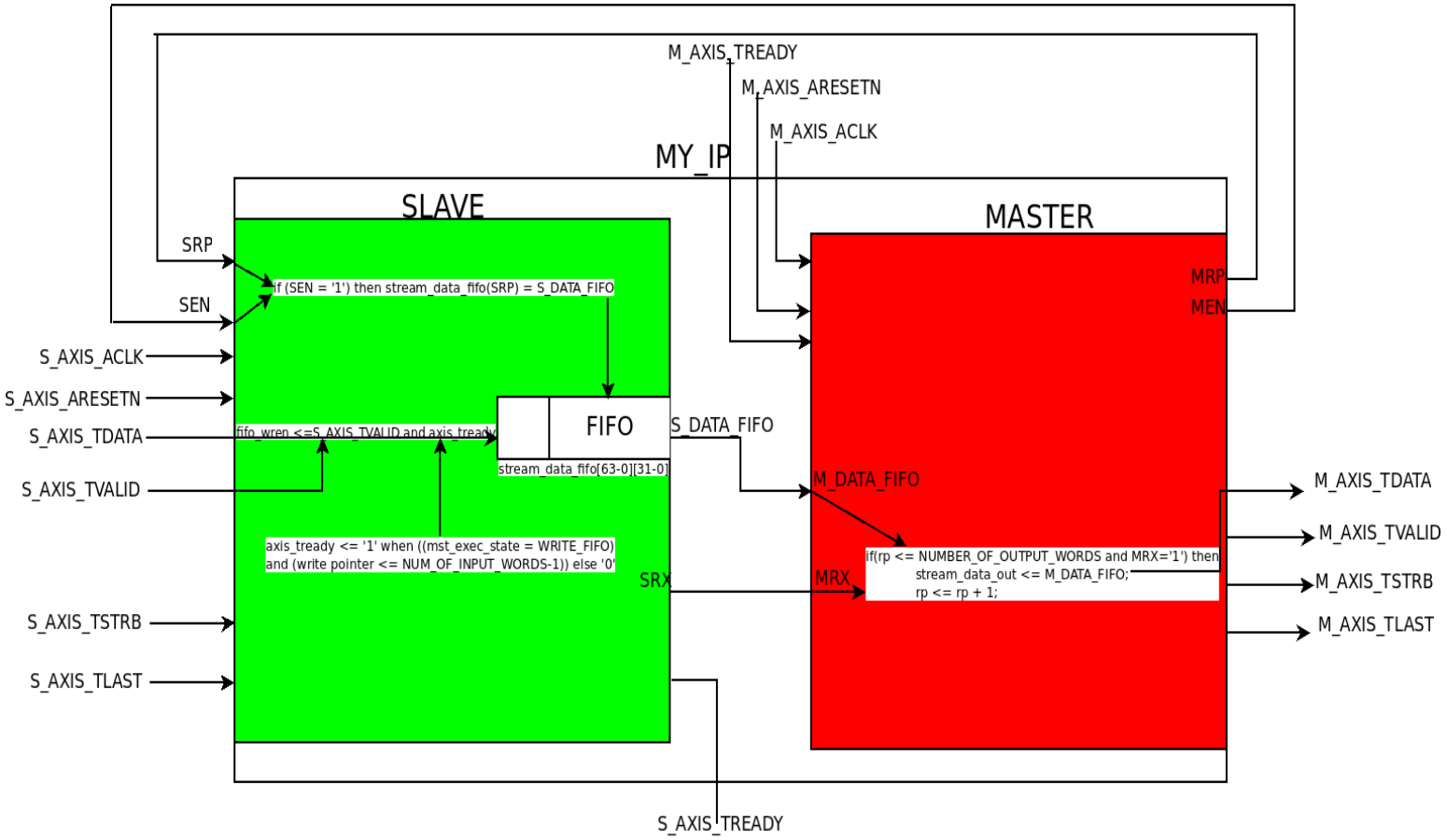
Κατά τη διαδικασία εγγραφής στη FIFO πρέπει το fifo_wren να είναι '1', αυτό συμβαίνει όταν το σήμα S_AXIS_TVALID = '1' και το axis_tready = '1'. Με αυτό τον τρόπο και μέσα από ένα process καταφέρνουμε να στείλουμε ένα σήμα ώστε να γράψουμε νέα δεδομένα στη FIFO στην επιθυμητή θέση μνήμης μέσω του write_pointer <stream_data_fifo(write_pointer)>.

Διαδικασία ανάγνωσης στη FIFO

Κατά τη διαδικασία ανάγνωσης στη FIFO παρατηρούμε, μέσω της κυματομορφής, ότι το σήμα M_AXIS_TREADY δεν λαμβάνεται στα επιθυμητά χρονικά σημεία, έτσι λοιπόν σχεδιάσαμε και υλοποιήσαμε μια νέα λογική δημιουργώντας ένα νέο read_pointer που τρέχει παράλληλα με τον δοθέν. Στη σχεδίασή μας όταν σταματήσει ο slave να γράφει στη FIFO και ο read_pointer δεν έχει διαβάσει αυτά τα δεδομένα(ή είναι σε χαμηλότερη θέση από αυτή του write_pointer), αποστέλλεται σήμα στον master. Έτσι αφού διαβάσει τα δεδομένα από τη θέση μνήμης της FIFO που έχει ορίσει ο νέος read_pointer, έπειτα τον αυξάνει και συνεχίζει το διάβασμα αν υπάρχουν περισσότερα δεδομένα προς ανάγνωση ή σταματά αν φτάσει σε ίδιο σημείο με τον write_pointer.

Σχηματική αναπαράσταση παράλληλων modules

Παρακάτω παρατίθεται η σχηματική αναπαράσταση των παράλληλων modules κατόπιν παρέμβασής μας καθώς και οι μεταξύ τους συνδέσεις.



(Το block diagram υπάρχει και σε ξεχωριστό αρχείο στο .zip)

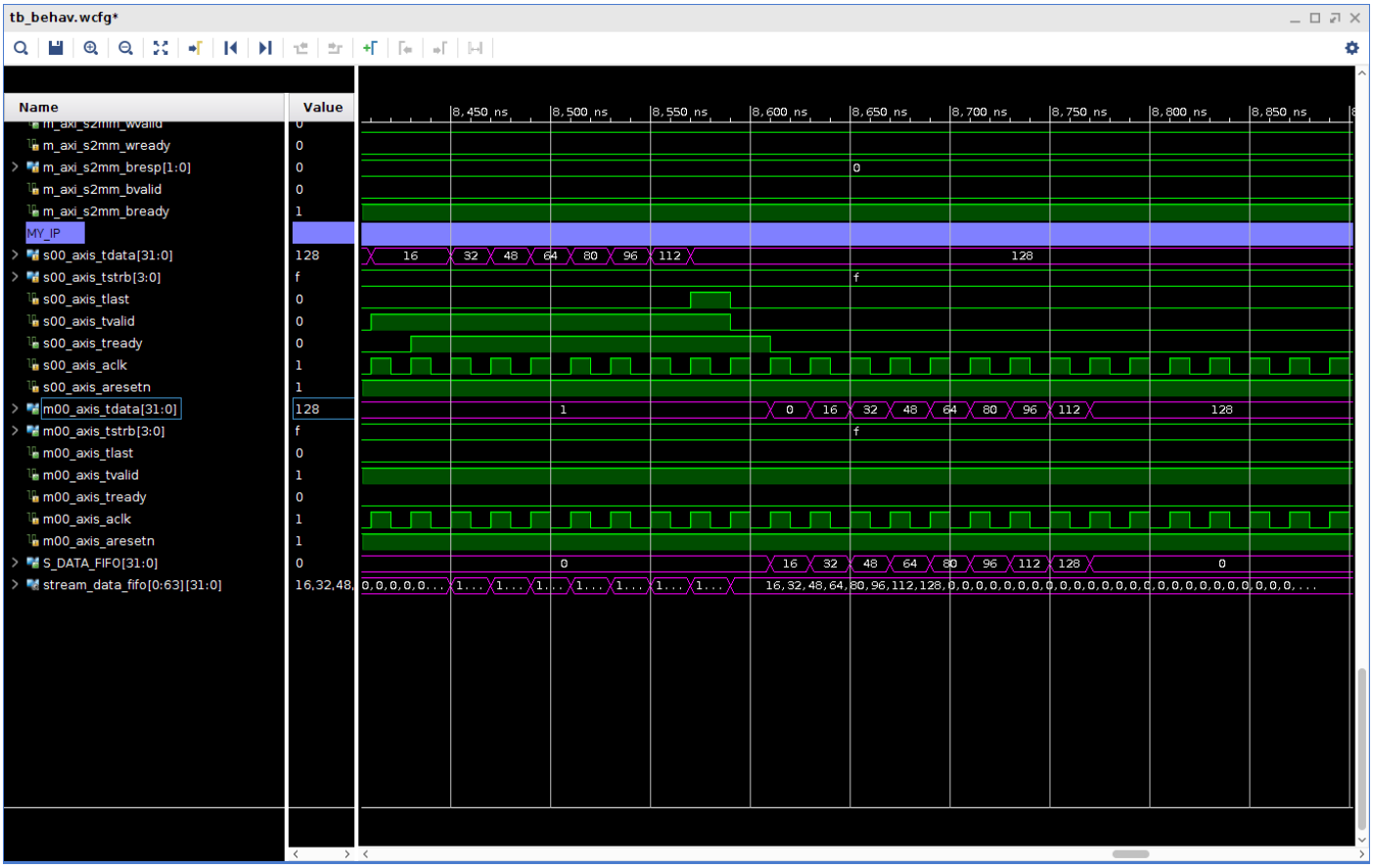
Προσθέσαμε στα module του **MASTER(M)** και του **SLAVE(S)** επιπλέον 4 σήματα στο καθένα :

- **M_DATA_FIFO - S_DATA_FIFO** : είναι τα δεδομένα που παίρνει από τη FIFO που βρίσκεται στο module του SLAVE.
- **MRX - SRX** : είναι σήμα που παίρνουμε απο τον SLAVE όταν αυτός τελειώνει το γράψιμο νέων δεδομένων στην FIFO.

- **MRP - SRP** : είναι ένας ακέραιος ο οποίος δείχνει από ποια θέση της FIFO θέλει να διαβάσει δεδομένα ο MASTER (στην ουσία είναι ο read_pointer που φτιάξαμε και MRP = rp σύμφωνα με κώδικα και block diagram).

- **MEN - SEN** : είναι ένα σήμα που στέλνει ο MASTER στην FIFO του SLAVE ότι είναι έτοιμος να διαβάσει νέα δεδομένα.

Κυματομορφές



(Η κυματομορφή υπάρχει και σε ξεχωριστό αρχείο στο .zip)

Στην παραπάνω κυματομορφη βλέπουμε κάποια σήματα με μώβ χρώμα. Το 1ο σήμα ***(s00_axis_tdata[31:0])*** είναι τα δεδομένα που έρχονται απο τον DMA προς τον slave. Το 2ο σήμα ***(m00_axis_tdata[31:0])*** είναι τα δεδομένα που εξέρχονται απο τον master και κατευθύνονται προς τον DMA. Το 3ο σήμα ***(S_DATA_FIFO[31:0])*** είναι ποιά δεδομένα παίρνει ο master απο την FIFO. Το 4ο σήμα ***(stream_data_fifo [0:63][31:0])*** είναι τα δεδομένα που έχει κάθε στιγμή η FIFO .