

HY220: Εργαστήριο Ψηφιακών Κυκλωμάτων

Τμήμα Επιστήμης Υπολογιστών, Πανεπιστήμιο Κρήτης
Χειμερινό Εξάμηνο 2015

Εργαστήριο 5: Υλοποίηση Ελεγκτή Μνήμης

9 Νοεμβρίου έως 29 Νοεμβρίου 2015 (3 εβδομάδες)

1.1 Σκοπός της Εργαστηριακής Άσκησης

Σε αυτήν την Άσκηση θα επιχειρήσουμε να δημιουργήσουμε κύκλωμα που να υλοποιεί ένα ελεγκτή μνήμης (256 θέσεις, με πλάτος 8-bit). Ο ελεγκτής μνήμης στην συνέχεια πρέπει να συνδεθεί με ένα κύκλωμα σειριακής επικοινωνίας (UART) κύκλωμα (βλ. παρακάτω σχήμα), το οποίο προωθεί εντολές για ανάγνωση και εγγραφή δεδομένων μέσω σειριακής σύνδεσης.

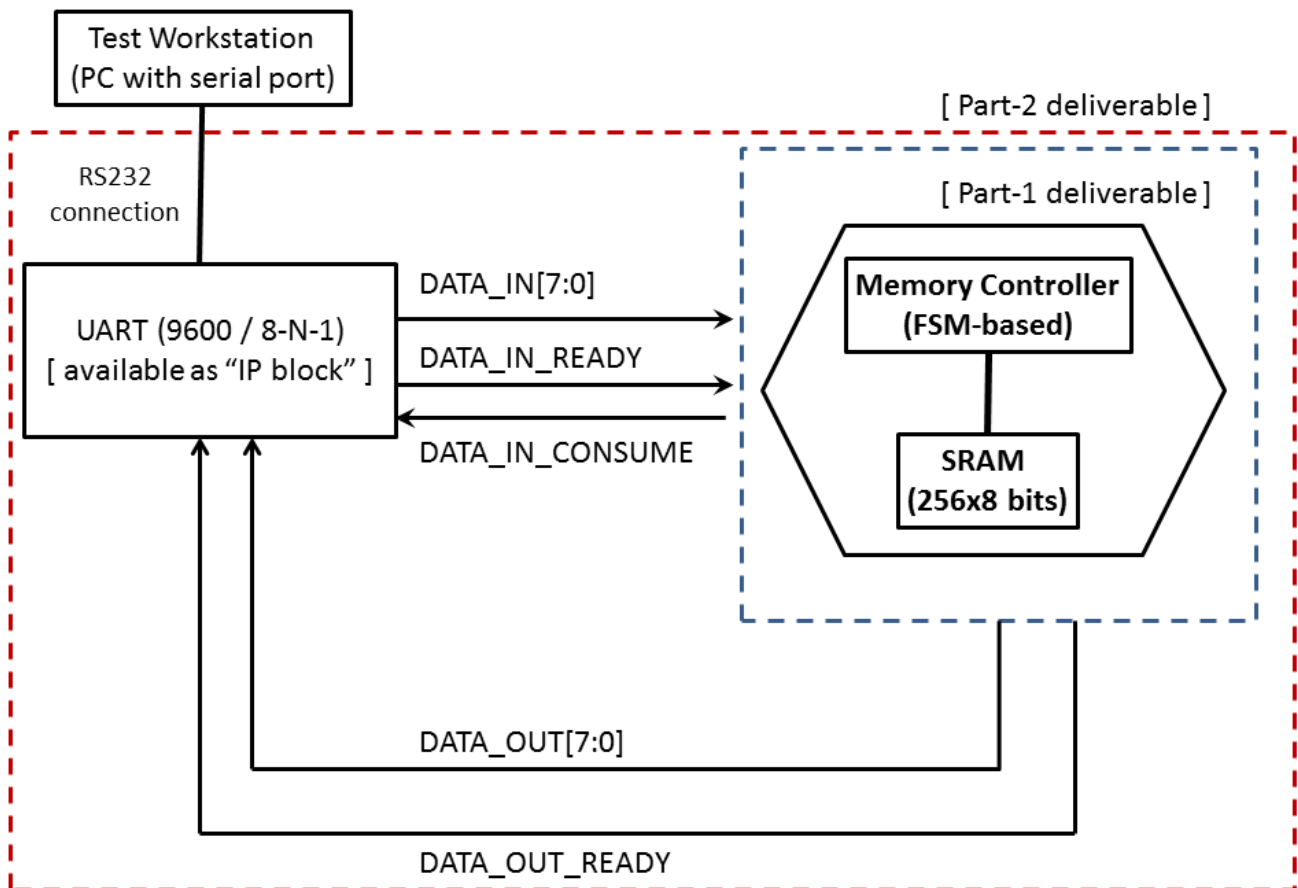


Figure 1: Σύστημα ελέγχου μνήμης (256 θέσεις, 8-bit δεδομένα), συνδεδεμένο με σειριακή διεπαφή (interface).

Τα δεδομένα από το σειριακό τερματικό δίδονται ως ακολουθίες χαρακτήρων, και αντίστοιχα στέλνονται τα αποτελέσματα στο σειριακό τερματικό. Με χρήση του σειριακού τερματικού θα πρέπει να επιδείξετε την λειτουργία του συστήματος αυτού.

1.2 Σχεδιασμός Συστήματος και Αναφορά

Πρώτο βήμα για την υλοποίηση της εργασίας είναι ο σχεδιασμός, στο χαρτί ή και σε ηλεκτρονική μορφή, το κύκλωμα που υλοποιεί την λογική ελέγχου μνήμης, κάνοντας χρήση του BRAM block που παράγεται από το εργαλείο σχεδίασης Xilinx ISE. Για την λογική ελέγχου της μνήμης θα χρειαστεί να σχεδιάσετε κατάλληλη μηχανή πεπερασμένων καταστάσεων (Finite State Machine - FSM). Αφού σχεδιάσετε το κύκλωμα, πρέπει να γράψετε στην αναφορά σας την λειτουργία του σε υψηλό επίπεδο, μετά να περιγράψετε την υλοποίησή σας και τέλος τα τεστ προσομοίωσης που θα κάνετε για να επαληθεύσετε την ορθότητα των κυκλωμάτων σας. Στην προσομοίωση δεν χρειάζεται να περιλάβετε το UART, θα προσομοιώσετε τα σήματα εισόδου και εξόδου του UART με "test vectors".

Στο επόμενο βήμα, το κύκλωμα ελέγχου μνήμης θα πρέπει να συνδυαστεί με πραγματικό κύκλωμα UART. Το UART προωθεί εντολές εγγραφής και ανάγνωσης από ένα σειριακό τερματικό. Οι εντολές δίδονται από το σειριακό τερματικό ως ακολουθίες χαρακτήρων ASCII, με το εξής συντακτικό:

- Εγγραφή: **W**<space><Address><space><Data><CR>
- Ανάγνωση: **R**<space><Address><CR>

Οι παράμετροι <Address> και <Data> δίδονται ως 2 δεκαεξαδικά ψηφία. Η κάθε εντολή τερματίζεται με τον ειδικό χαρακτήρα carriage return (<CR>).

1.3 Συγγραφή του Κώδικα Verilog και Προσομοίωση

Αφού ετοιμάσατε την αναφορά σας θα πρέπει να την μετατρέψετε σε κώδικα Verilog. Αμέσως μετά πρέπει να δημιουργήσετε τα testbench modules με το οποίο θα δοκιμάσετε να προσομοιώσετε το σύστημα.

1.4 Δημιουργία UCF και Τοποθέτηση

Αφού ολοκληρώσετε τα προηγούμενα βήματα, πρέπει να δημιουργήσετε το Αρχείο Περιορισμών Χρήστη (User Constraints File – UCF) δηλώνοντας τις εισόδους και τις εξόδους που θα χρησιμοποιήσετε στην πλακέτα και την τάση που θα τους ασκήσετε, όπως δείχνει το ucf αρχείο που λάβατε στο εργαστήριο 0 το οποίο είναι κομμάτι του master ucf που έχει δοθεί από τους κατασκευαστές.

1.5 Παράδοση

Την 2η εβδομάδα (16/11-20/11) θα παραδώσετε την αναφορά σας και την verilog, και την τρίτη εβδομάδα (23/11-29/11) θα τρέξετε το σχέδιο σας στην πλακέτα και θα παραδώσετε όλα σας τα αρχεία. Σας συστήνουμε να έχετε έτοιμη την αναφορά σας την δεύτερη εβδομάδα, ώστε αν έχετε προβλήματα ή απορίες να τα δούμε την δεύτερη εβδομάδα και να τα επανεξετάσουμε την τρίτη εβδομάδα. Για τις ώρες της εξέτασης κλείνετε timeslot μέσω του Rendezvous.