

# HY220: Εργαστήριο Ψηφιακών Κυκλωμάτων

Τμήμα Επιστήμης Υπολογιστών, Πανεπιστήμιο Κρήτης  
Χειμερινό Εξάμηνο 2015

## Εργαστήριο 3: Υλοποίηση Ελεγκτή Ουράς Δεδομένων

30 Νοεμβρίου έως 20 Δεκεμβρίου 2015 (3 εβδομάδες)

### 1.1 Σκοπός της Εργαστηριακής Άσκησης

Σε αυτήν την Άσκηση θα επιχειρήσουμε να δημιουργήσουμε κύκλωμα που να υλοποιεί ένα ελεγκτή ουράς δεδομένων (FIFO - 256 θέσεις, με πλάτος 8-bit). Ο ελεγκτής 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.

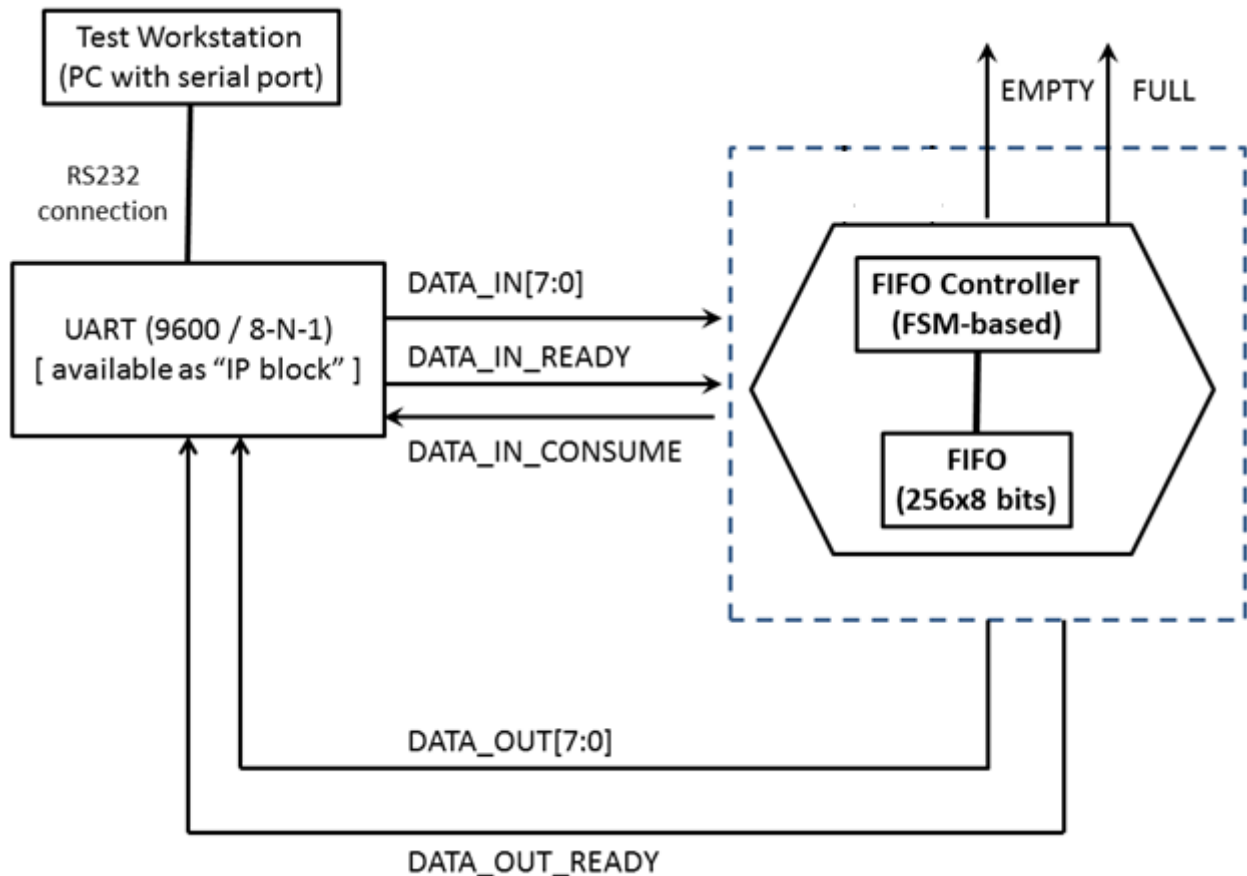


Figure 1: Σύστημα ελέγχου ουράς δεδομένων (256 θέσεις, 8-bit δεδομένα), συνδεδεμένο με σειριακή διεπαφή.

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 ουράς δεδομένων, κάνοντας χρήση του FIFO block που παράγεται από το εργαλείο σχεδίασης Xilinx ISE. Για την λογική ελέγχου της ουράς δεδομένων θα χρειαστεί να σχεδιάσετε κατάλληλες μηχανές πεπερασμένων καταστάσεων (Finite State Machines - FSMs) 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 ουράς δεδομένων θα πρέπει να συνδυαστεί με πραγματικό κύκλωμα UART (το οποίο θα δοθεί έτοιμο για χρήση). Το UART προωθεί δεδομένα από ένα σειριακό τερματικό προς την ουρά δεδομένων - λειτουργία ENQueue, και αντίστοιχα προωθεί δεδομένα από την ουρά δεδομένων προς το σειριακό τερματικό όταν ο χρήστης ενεργοποιήσει την λειτουργία DEQueue. Οι εντολές δίδονται από το σειριακό τερματικό ως ακολουθίες χαρακτήρων ASCII, με το εξής συντακτικό:

- ENQ: **W**<space>Data<CR>
- DEC: **R**<CR>

Η ουρά δεδομένων παρέχει ένδειξη για το εάν είναι άδεια (EMPTY) ή πλήρης (FULL). Υπο αυτές τις συνθήκες, οι λειτουργίες DEQueue και ENQueue δεν μπορούν να ολοκληρωθούν. Οι ενδείξεις EMPTY, FULL πρέπει να γίνονται ορατές με την χρήση των LEDs.

Τα δεδομένα εκφράζονται ως 2 δεκαεξαδικά ψηφία (συνολικά 8 bits πληροφορίας), και τερματίζονται με τον ειδικό χαρακτήρα carriage return (<CR>).

### **1.3 Συγγραφή του Κώδικα Verilog και Προσομοίωση**

Αφού ετοιμάσατε το σχέδιο του κυκλώματος, θα πρέπει να το μετατρέψετε σε κώδικα Verilog. Αμέσως μετά πρέπει να δημιουργήσετε ένα testbench module 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 (7/12-13/12) θα παραδώσετε την αναφορά σας και την verilog, και την τρίτη εβδομάδα (14/12-20/12) 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.