

Axelou Olympia, 2161

oaxelou@uth.gr

ce430: Spartan 3 FPGA

04/12/2018

Lab3: VGA Controller Implementation

Στόχος αυτού του εργαστηρίου είναι η υλοποίηση ενός Ελεγκτή/Οδηγού θύρας οθόνης VGA (Video Graphics Array), με σκοπό την οδήγηση μιας συμβατικής οθόνης και την εμφάνιση εικόνας σε αυτή. Μέρος της εσωτερικής μνήμης της FPGA χρησιμοποιείται ως Video RAM.

Προβάλλω την προτεινόμενη εικόνα και συγκεκριμένα στο πρώτο $\frac{1}{4}$ της εικόνας την οριζόντια αλληλουχία <άσπρο, κόκκινο, άσπρο>, αντίστοιχα στο 2^ο και 3^ο τέταρτο της εικόνας πράσινο και μπλε αντί για κόκκινο. Στο τελευταίο, είναι η κάθετη αλληλουχία <άσπρο, 2 x κόκκινο, 2 x πράσινο, 2 x μπλε, άσπρο>.

Part A: VRAM Implementation

Υλοποίηση

Χρησιμοποίησα τρεις 16x1 Single-Port Block Ram memories της Spartan 3 FPGA, μία για κάθε χρώμα RGB. Χρησιμοποιώ τις περιοχές μνήμης από [12287, 0], δηλαδή η ανάλυση της εικόνας που αποθηκεύω είναι 128x96. Τα υπόλοιπα τα αρχικοποιώ σε 1 (κυρίως για λόγους debugging, ώστε αν βγει εκτός των ορίων και διαβάσει από αυτό το μέρος της μνήμης που δε χρησιμοποιείται να γίνει αντιληπτό). Το module όπου είναι τα instantiations είναι στο bram.v

Τα μέρη στα οποία έχω χωρίσει τη μνήμη είναι τα εξής:

1. Κόκκινο – Οριζόντιες γραμμές [12,287 – 9,216]
2. Πράσινο – Οριζόντιες γραμμές [9,215 – 6,144]
3. Μπλε – Οριζόντιες γραμμές [6,143 – 3,072]
4. Multicolor – Κάθετες γραμμές [3,071 – 0]

Επαλήθευση

Ελέγχω και τις 4 περιοχές της μνήμης θέτοντας το address στην κατάλληλη τιμή για το τμήμα που ελέγχω και τσεκάροντας τι τιμές επιστρέφει για τα red, green & blue.

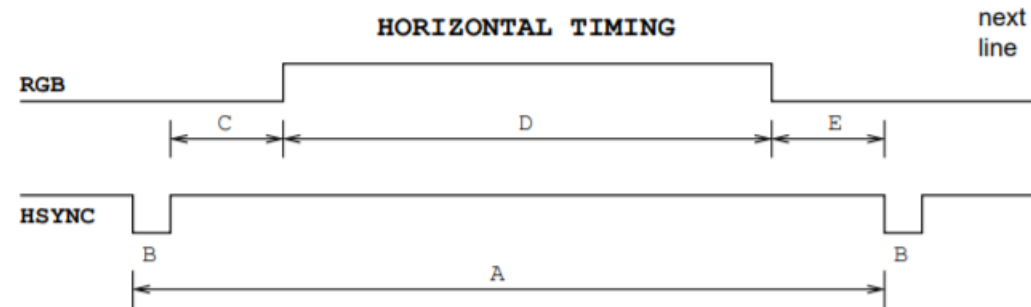
Η διαδικασία είναι αυτοματοποιημένη και απλά εκτυπώνεται στο monitor η διεύθυνση και οι τιμές σε κάθε μνήμη και αν η τιμή που διαβάζει είναι η αναμενόμενη τιμή, τότε εκτυπώνεται στην αρχή της γραμμής το “[OK]”.

Πείραμα/Τελική Υλοποίηση

Αυτό το μέρος δεν έχει ελεγχθεί στην FPGA.

Part B: HSYNC Implementation & Οριζόντιου Μετρητή Pixel

Υλοποίηση



Διάστημα	Περιγραφή	Τιμή
A	Χρόνος Σάρωσης Γραμμής - Scanline Time	32 μ sec
B	Πλάτος Παλμού HSYNC - HSYNC Pulse Width	3.84 μ sec
C	Πίσω Όψη - Back Porch	1.92 μ sec
D	Χρόνος Απεικόνισης - Display Time	25.6 μ sec
E	Μπροστινή Όψη - Front Porch	0.640 μ sec

Με βάση τους παραπάνω χρόνους η μέγιστη τιμή του counter κύκλων 20ns θα είναι:

- Μετά το τέλος του B: $192 - 1$
- Μετά το τέλος του (B + C): $288 - 1$
- Μετά το τέλος του (B + C + D): $1568 - 1$
- Μετά το τέλος του A (ολοκλήρωση ενός κύκλου): $1600 - 1$

Τα υποκυκλώματα είναι:

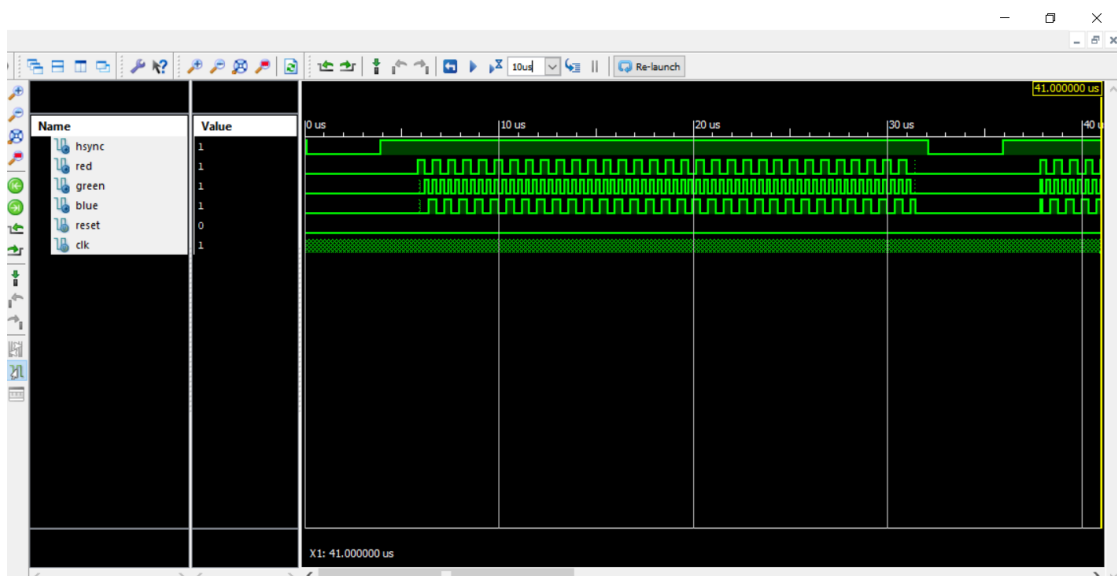
1. Vgacontroller (Top Level Module): Περιέχει τα instantiations των pixelcontroller και hsynchronizer.
2. BRAM (from partA)
3. Pixelcontroller: Υπολογίζει ποιο pixel είναι για προβολή, συντάσσει την τελική διεύθυνση για τις μνήμες των χρωμάτων και οδηγεί κατάλληλα τα χρώματα. Περιέχει 1 always block όπου αυξάνονται οι counters για την οριζόντια προσπέλαση της μνήμης.
4. Hsynchronizer: Υπολογίζει πότε να ενεργοποιήσει το hsync και πότε είναι display_time. Περιέχει 2 always blocks:
 - i. Στο πρώτο αυξάνεται ο μετρητής για τον κύκλο του hsync.
 - ii. Στο δεύτερο αλλάζουν τιμές τα hsync και display_time.
5. Testbench του κυκλώματος: Αρχικοποιεί το κύκλωμα με το reset και ορίζει το ρολόι.

Σχόλια για το PixelController: Με βάση το display_time (25.6 usec) κάθε οριζόντιο pixel πρέπει να προβληθεί για 10 κύκλους για να είμαστε στο χρονικό περιθώριο (αντί για 5 που είναι η αναλογία μεταξύ του 640x480 και 128x96 και το οποίο ταιριάζει για τον κάθετο μετρητή του vsync). Ως max τιμή του counter που ελέγχει πόσους κύκλους έχω ορίσει το 5 και όχι το 10 για να ταιριάζει με το vsync. Συνεπώς κάθε γραμμή της εικόνας εκτυπώνεται δύο φορές στην οθόνη.

Επαλήθευση

Έχω ορίσει να διατρέχει οριζοντίως τη γραμμή 22 (multicolor section). Συνεπώς στις κυματομορφές το αποτέλεσμα για μία ενεργή ακμή του display_time είναι <1 άσπρο, 2 κόκκινο, 2 πράσινο, 2 μπλε, 1 άσπρο>.

Ο έλεγχος γίνεται παρατηρώντας τις κυματομορφές.

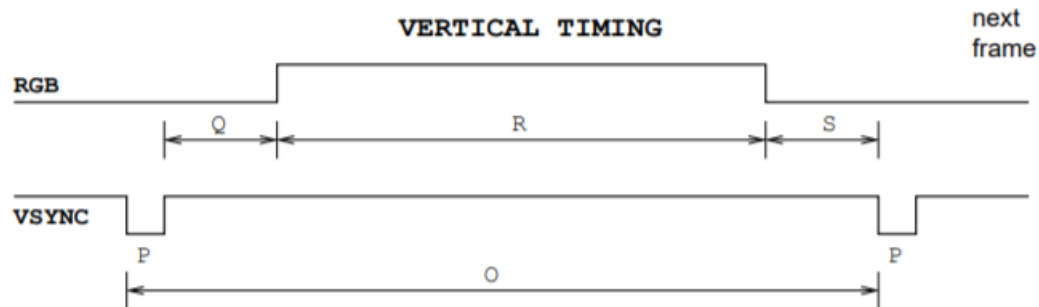


Πείραμα/Τελική Υλοποίηση

Αυτό το μέρος δεν έχει ελεγχθεί στην FPGA.

Part C: VSYNC Implementation & Κάθετου Μετρητή Pixel

Υλοποίηση



Διάστημα	Περιγραφή	Τιμή
O	Συνολικός Χρόνος Εικόνας - Total Frame Time	16.67msec
P	Πλάτος Παλμού VSYNC - VSYNC Pulse Width	64 μ sec
Q	Πίσω Όψη - Back Porch	928 μ sec
R	Χρόνος Ενεργής Απεικόνισης - Active Video Time	15.36 msec
S	Μπροστινή Όψη - Front Porch	320 μ sec

Με βάση τους παραπάνω χρόνους η μέγιστη τιμή του counter κύκλων 20ns θα είναι:

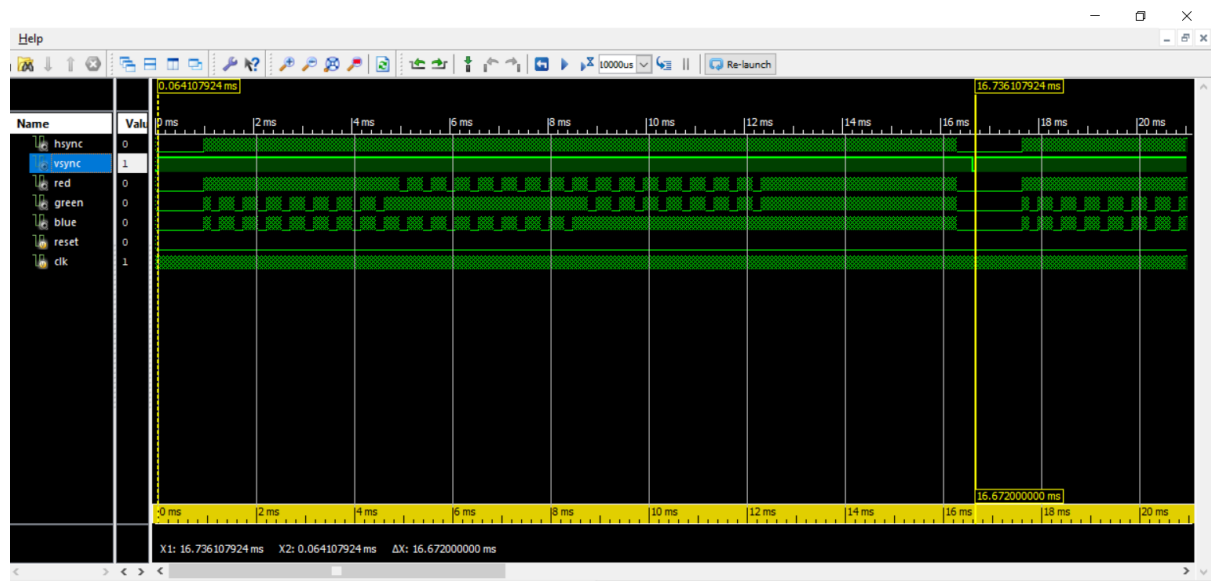
- Μετά το τέλος του P: $3200 - 1$
- Μετά το τέλος του (P + Q): $49600 - 1$
- Μετά το τέλος του (P + Q + R): $812000 - 1$
- Μετά το τέλος του O (ολοκλήρωση ενός κύκλου): $833600 - 1$

Τα υποκυκλώματα είναι:

1. Vgacontroller (Top Level Module): Περιέχει τα instantiations των pixelcontroller και hsynchronizer.
2. BRAM (from partA)
3. Pixelcontroller: (from partB με τη μόνη διαφορά ότι η γραμμή της μνήμης είναι input στο κύκλωμα από το vsynchronizer κύκλωμα και δεν είναι μία fixed τιμή)
4. Hsynchronizer: (from partB με τη μόνη διαφορά πως όταν ολοκληρωθεί ένας κύκλος του hsync αυξάνονται δύο counter: ένας που δείχνει σε ποια γραμμή είμαστε και ένας multiplier counter που υπολογίζει για πόσους κύκλους του hsync θα παραμείνουμε σε μία γραμμή της εικόνας)
5. Vsynchronizer: Παρόμοιος τρόπος προσέγγισης με το hsynchronizer του partB (χωρίς τους 2 counter για την κάθετη προσπέλαση της εικόνας), με μόνη διαφορά τις τιμές του counter στις οποίες ενεργοποιείται το vsync και το hsync_time (κατ' αντιστοιχία με τα hsync & display_time)
6. Testbench του κυκλώματος: Αρχικοποιεί το κύκλωμα με το reset και ορίζει το ρολόι.

Επαλήθευση

Ο έλεγχος γίνεται παρατηρώντας τις κυματομορφές.



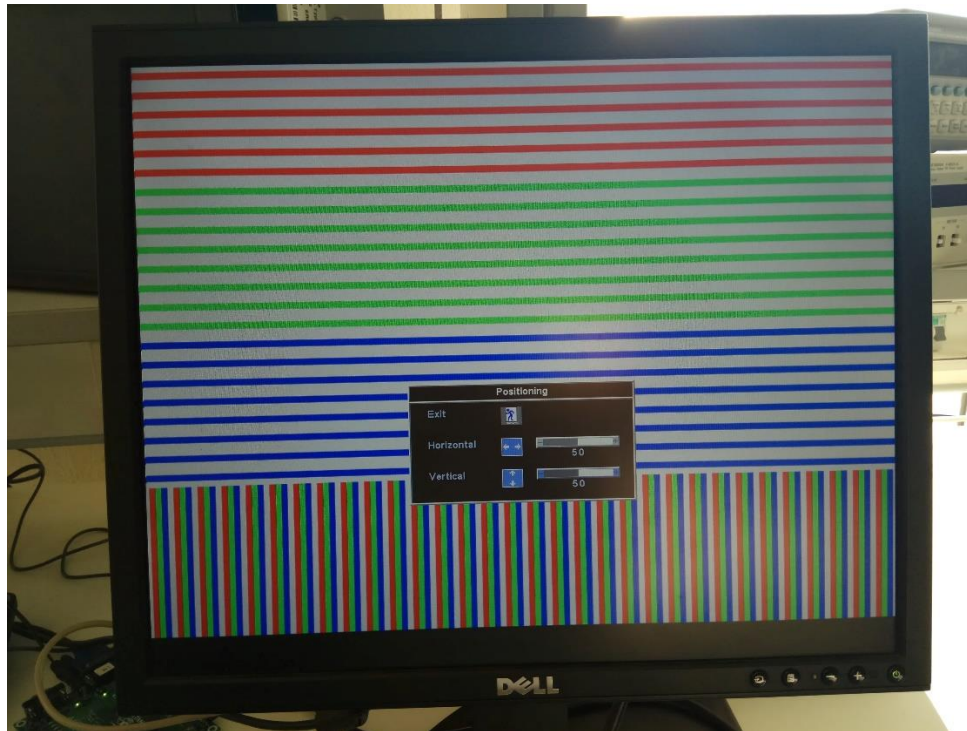
Πείραμα/Τελική Υλοποίηση

Την πρώτη φορά που φόρτωσα την πλακέτα, υπήρχαν 2 bugs:

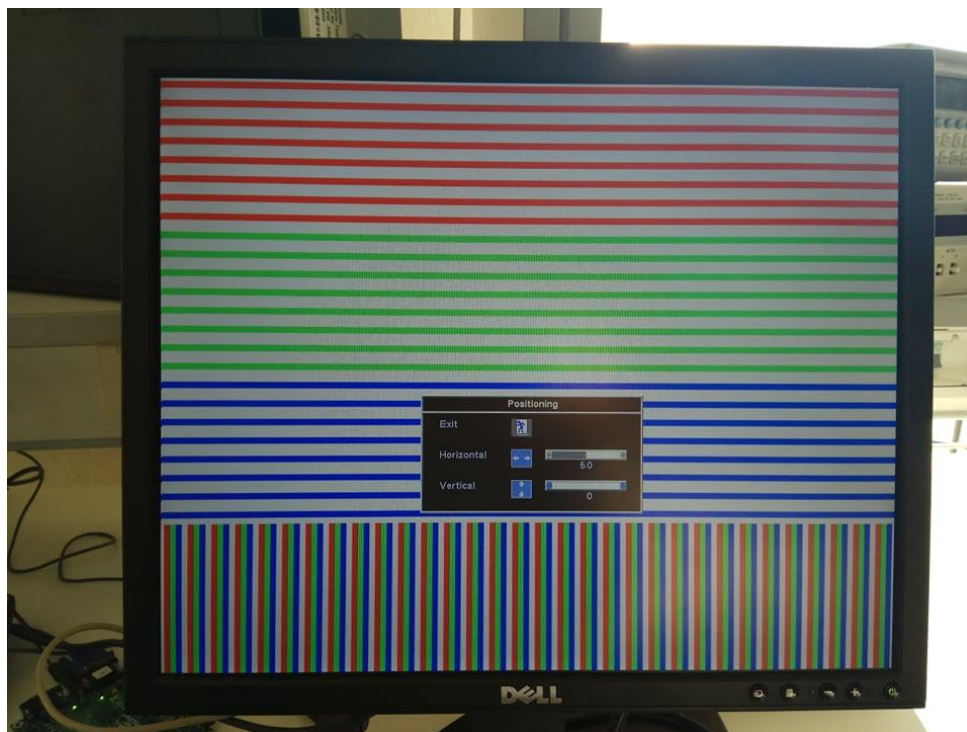
- Το πρώτο ήταν πως η εικόνα που προβαλλόταν είχε πολλές αποχρώσεις των χρωμάτων μπλε, κόκκινο, ροζ, κυανό κλπ... Παρόλο που σκοπός ήταν τα 3 βασικά χρώματα. Επίσης φαινόταν διαβαθμίσεις: για παράδειγμα το μπλε (που έπρεπε να δείχνει) σταδιακά γινόταν μαύρο. Μετά από παρατήρηση της οθόνης λύθηκε το πρόβλημα: όταν το display_time είναι μηδέν αδιαφορούσα για τις τιμές των σημάτων των χρωμάτων ενώ δεν έπρεπε. Έτσι, έβαλα έναν πολυπλέκτη που οδηγούσε το χρώμα από τη μνήμη στην έξοδο μόνο αν ήταν ενεργό το display_time, διαφορετικά η έξοδος είναι μηδέν.
- Το άλλο πρόβλημα που παρουσιάστηκε ήταν πως μία λωρίδα στο κάτω μέρος της οθόνης δεν γραφόταν, δηλαδή ήταν μαύρη και έλλειπε ίδιου πάχους τμήμα από την αρχή. Παρόλες τις προσπάθειες για debugging, το πρόβλημα αυτό δεν επιλύθηκε. Αναλύω το θέμα αυτό παρακάτω.

Όπως φαίνεται και παραπάνω στο screenshot, ακόμα και σε Post-Route Simulation δεν παρατηρήθηκε κανένα πρόβλημα, ούτε σχετικά με την προσπέλαση της μνήμης, ούτε με τον συγχρονισμό των hsync & vsync σημάτων.

Παρόλα αυτά, το τελικό αποτέλεσμα κατά το τρέξιμο στην πλακέτα είναι το παρακάτω (σε auto-adjust):



Δηλαδή, η εικόνα ήταν μετατοπισμένη προς τα πάνω: τμήμα του κόκκινου μέρους έλλειπε και στην κάτω πλευρά υπήρχε μία μαύρη λωρίδα που δεν γραφόταν. Βέβαια, «πειράζοντας» το positioning της οθόνης έβγαине το επιθυμητό αποτέλεσμα:



Οι κυματομορφές στο simulation είναι φαινομενικά σωστές και μετά από ώρες debugging δεν κατάφερα να διορθώσω αυτό το πρόβλημα.

Συνολικές φορές που προγραμματίσα την πλακέτα: 2.

Συμπεράσματα

Η διαδικασία της ανάπτυξης του κώδικα έγινε γρήγορα. Δεν παρουσιάστηκε κανένα ιδιαίτερο πρόβλημα στα part A, B και C, το simulation φαινόταν να δουλεύει κανονικά ακόμα και σε Post-Route Simulation, όμως κάποια bugs εμφανίστηκαν κατά το τρέξιμο του κώδικα στην πλακέτα και η διαδικασία του debugging ήταν που πήρε την περισσότερη ώρα.