



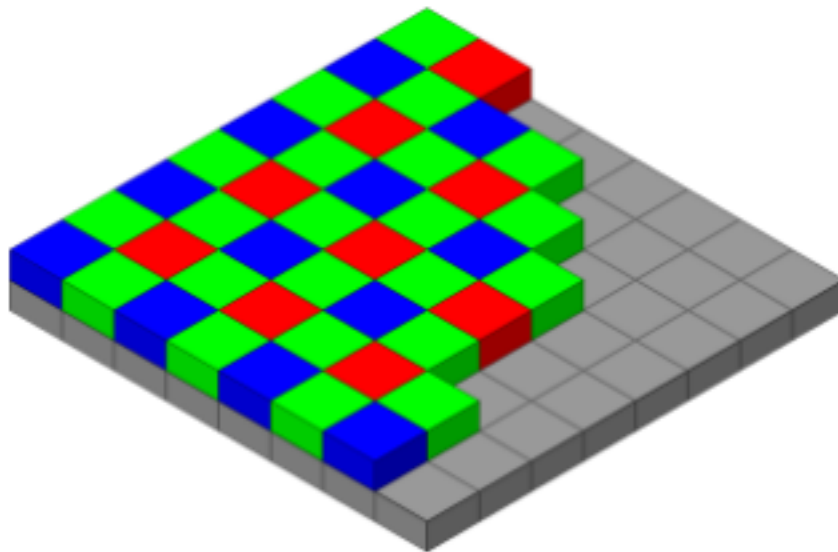
## Ψηφιακά Συστήματα VLSI

8ο εξάμηνο, Ακαδημαϊκή περίοδος 2024-2025

6η Εργαστηριακή Άσκηση: Υλοποίηση Debayering Φίλτρου με AXI διεπαφή σε Zynq SoC  
FPGA

Δημήτρης Καμπανάκης: 03121012

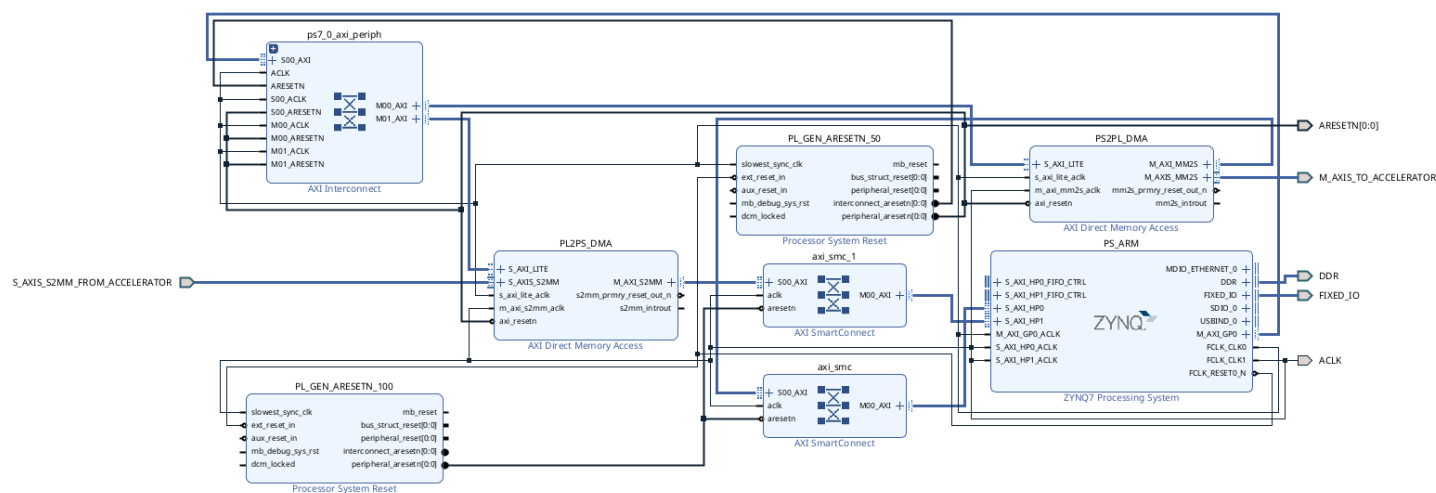
Αγγελική Ζέρβα: 03121101



# 1 Εισαγωγή

Στην εργαστηριακή άσκηση αυτή έχουμε υλοποιήσει ένα πλήρες debayering filter, το οποίο λαμβάνει έναν πίνακα από έναν αισθητήρα Bayer και επιστρέφει την RGB εικόνα που αντιστοιχεί. Παρακάτω αναλύουμε την μεθοδολογία που ακολουθήσαμε.

## 2 Μπλοκ Διάγραμμα Κυκλώματος



### 3 Βασικές Δομικές Μονάδες του Κυκλώματος

### 3.1 Calculator

Η οντότητα αυτή του κυκλώνατος είναι υπεύθυνη να υπολογίζει τις RGB τιμές για ένα pixel. Έχει σαν generic μεταβλητές τον αριθμό bits για κάθε pixel ου λαμβάνει και την διάσταση της εικόνας N. Δέχεται ως είσοδο την τιμή μιας 3x3 γειτονιάς pixel και τον αριθμό της γραμμής και στήλης του pixel, για να καθοριστεί τι χρώματος είναι το κεντρικό pixel της γειτονιάς. Υπολογίζει το χρώμα του κεντρικού pixel και τις τιμές για τα άλλα 2 χρώματα του pixel από τις τιμές αυτών της γειτονιάς και δίνει στην έξοδο της R,G,B τιμές για αυτό το pixel.

### 3.2 Serial to Parallel

Τα δεδομένα που λαμβάνει το φίλτρο έρχονται σειριακά, όμως για την επεξεργασία ενός δισδιάστατου σήματος όπως μια εικόνα πρέπει να τα διατάξουμε σωστά. Την σωστή κατανομή των pixel που λαμβάνονται σειριακά σε παράλληλες δομές που σχηματίζουν στην συνέχεια τις 3x3 γειτονιές αναλαμβάνει αυτή η οντότητα. Έχουμε χρησιμοποιήσει 3 fifo ουρές, οι οποίες έχουν μέγιστη χωρητικότητα 1024 τιμές pixel των 8 bits και είναι τύπου "First Word Fall Through". Οι ουρές είναι συνδεδεμένες σε σειρά μεταξύ τους για να μπορούν να περάσουν τιμές από την μία στην άλλη και με μια δομή 9 D-flip flops που είναι επίσης ανά 3 σε σειρά και θα αποτελούν τον τρόπο να αποθηκευτεί κάθε φορά η 3x3 γειτονιά για τους υπολογισμούς. Η οντότητα αυτή αναλαμβάνει για κάθε pixel που δέχεται να το τοποθετήσει στην 1η ουρά και να προωθήσει κατάλληλα τα pixel στις επόμενες ουρές και στα flip flops. Στην συνέχεια, με δεδομένο τον αριθμό στήλης και γραμμής που βρίσκεται το pixel που επεξεργάζεται κάθε φορά για την έξοδο, αντιστοιχεί τα δεδομένα των flip flops στα pixel της 3x3 γειτονιάς. Σημειώνεται για την τελευταία διαδικασία ότι έχουμε αντιστροφή 180 μοιρών ανάμεσα στην γειτονιά και στα dff και για τις τιμές κάποιων pixel πρέπει να παρέμβουμε στις τιμές για το κατάλληλο zero padding.

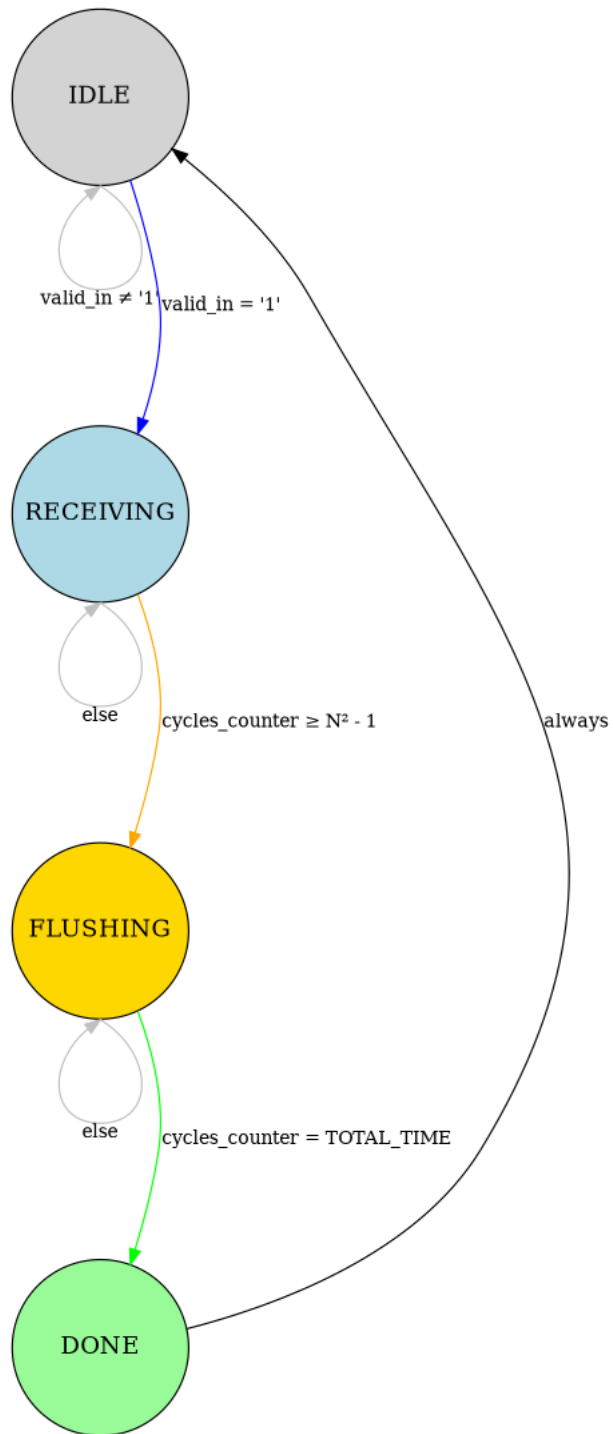
### 3.3 Debayering

Η οντότητα αυτή αποτελεί το κυρίως σώμα του φίλτρου μας. Διαχειρίζεται την οντότητα serial to parallel (η οποία δίνει έξοδο στον calculator) και χρησιμοποιεί ένα fsm 4 καταστάσεων, IDLE όταν δεν πραγματοποιεί κάποιο υπολογισμό, RECEIVING όταν βρίσκεται σε κατάσταση εισόδου δεδομένων και επεξεργασίας, FLUSHING όταν η είσοδος έχει σταματήσει και πραγματοποιεί τους εναπομείναντες υπολογισμούς και DONE όταν τελειώσει με τον υπολογισμό της τελικής εικόνας. Οι καταστάσεις του FSM επηρεάζουν τον τρόπο και τον ρυθμό με τον οποίο προωθούνται τα δεδομένα για υπολογισμό, ώστε να πετύχουμε τον συγχρονισμό και τα σήματα που δίνονται τελικά στην έξοδο.

### 3.4 Συνολική Λειτουργία Κυκλώματος

Στο συνολικό μας κύκλωμα χρησιμοποιούμε τις παραπάνω οντότητες και έχω σύστημα χειρισμού του debayering. Το σύστημα αυτό λαμβάνει τα δεδομένα εισόδου και φροντίζει να τα προωθή σωστά, δηλαδή μετατρέποντας το valid in σήμα σε παλμό, καθυστερώντας έναν κύκλο το πέρασμα κάθε pixel για σωστό συγχρονισμό και τέλος δίνοντας σωστά το συνοδευτικό σήμα valid out της εξόδου.

## 4 Διάγραμμα fsm



## 5 Ανάλυση χρήσης πόρων για τον FPGA

- $N = 64$

Resource	Utilization	Available	Utilization %
LUT	3572	17600	20.30
LUTRAM	439	6000	7.32
FF	5265	35200	14.96
BRAM	12.50	60	20.83

- $N = 128$

Resource	Utilization	Available	Utilization %
LUT	3580	17600	20.34
LUTRAM	439	6000	7.32
FF	5271	35200	14.97
BRAM	12.50	60	20.83

Παρατηρούμε όπως είναι λογικό, ότι για μεγαλύτερο μέγεθος εικόνας έχουμε μια μικρή αύξηση στην χρήση πόρων. Επειδή, όμως, υπάρχουν τεχνικές βελτιστοποίησης του design μας κατά το implementation δεν παρατηρούμε μεγάλη αύξηση, αλλά σχετικά σταθερή χρήση πόρων, ακόμα και για αύξηση του μεγέθους της εικόνας.

## 6 Χρονική Ανάλυση Κυκλώματος

- $N = 64$

Για εικόνα 64x64 pixel έχουμε περίπου  $2.5 \cdot 10^4$  κύκλους latency. Έχουμε σύνολο 4096 pixels και άρα το throughput θα είναι 0.164 εξόδους ανά κύκλο ρολογιού.

- $N = 128$

Για εικόνα μεγέθους 128x128 pixel έχουμε περίπου  $9.8 \cdot 10^4$  κύκλους latency. Εδώ έχουμε 16384 pixel και αυτό σημαίνει ότι το throughput θα είναι 0.167 εξόδοι ανά κύκλο ρολογιού.

Για το throughput για  $N = 64$  μπορούμε να συμπεράνουμε ότι αυτό που πραγματικά πληρώνουμε είναι όχι η διαδικασία του debayering, αλλά η αρχικοποίηση του DMA και άλλων στοιχείων και το overhead του DMA. Για μεγαλύτερο  $N$  η βελτίωση έρχεται ως αποτέλεσμα των ίδιων καθυστερήσεων σε ένα βαθμό, αλλά για σημαντικά περισσότερες εξόδους. Πραγματικό κέρδος με αυτόν τον τρόπο υπολογισμού, λοιπόν, έχουμε για πραγματικά μεγάλο  $N$ .