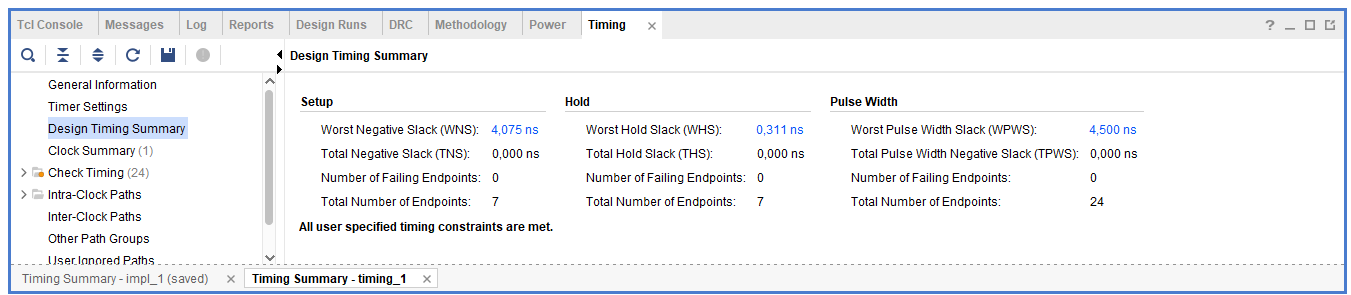
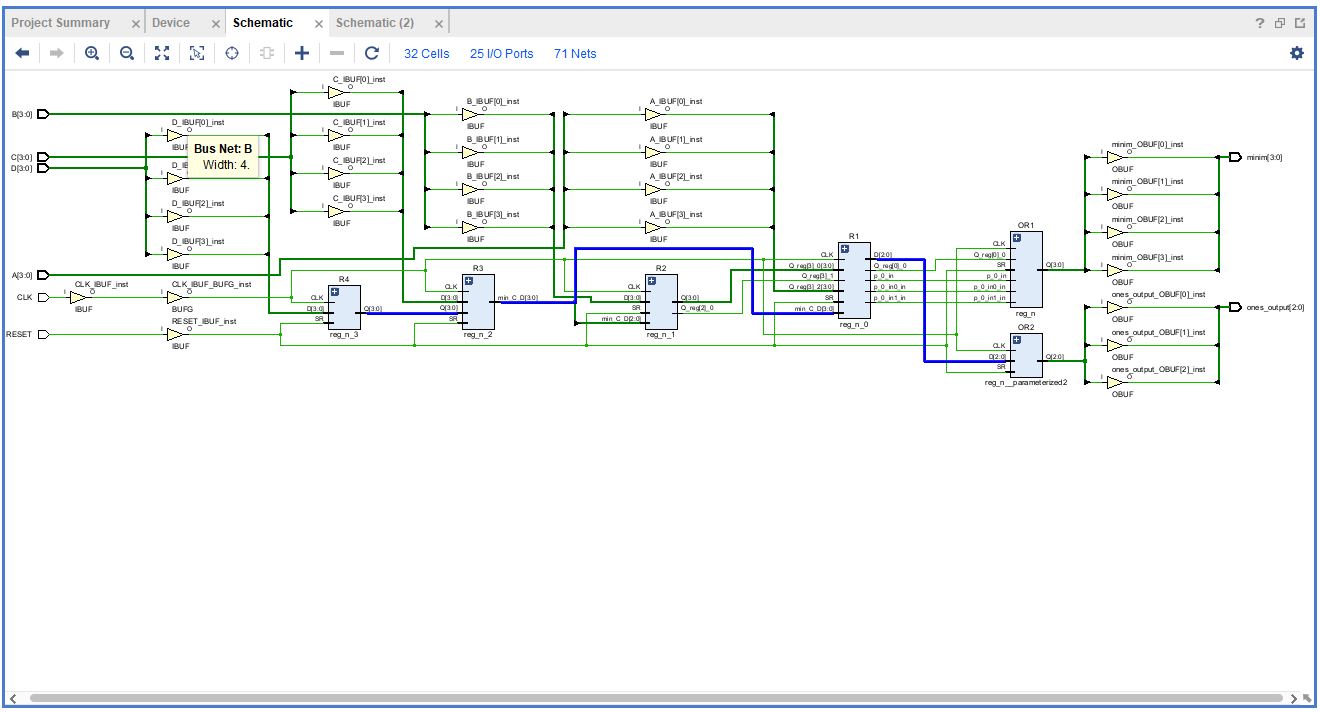
**Απαντήσεις ερωτήματος 3**

**α)Υλοποίηση χωρίς pipeline**

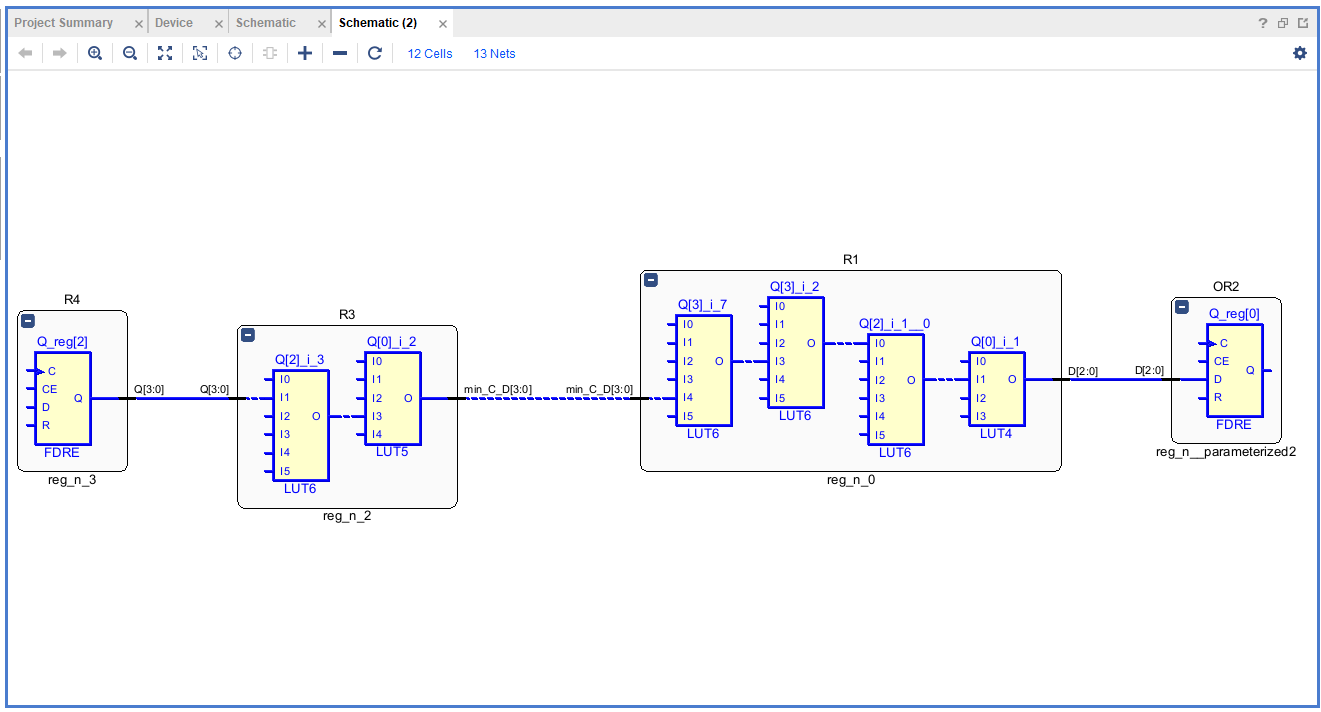
Αρχικά το clock έχει περίοδο 10 ns άρα η συχνότητα λειτουργίας είναι 108 Hz.

****

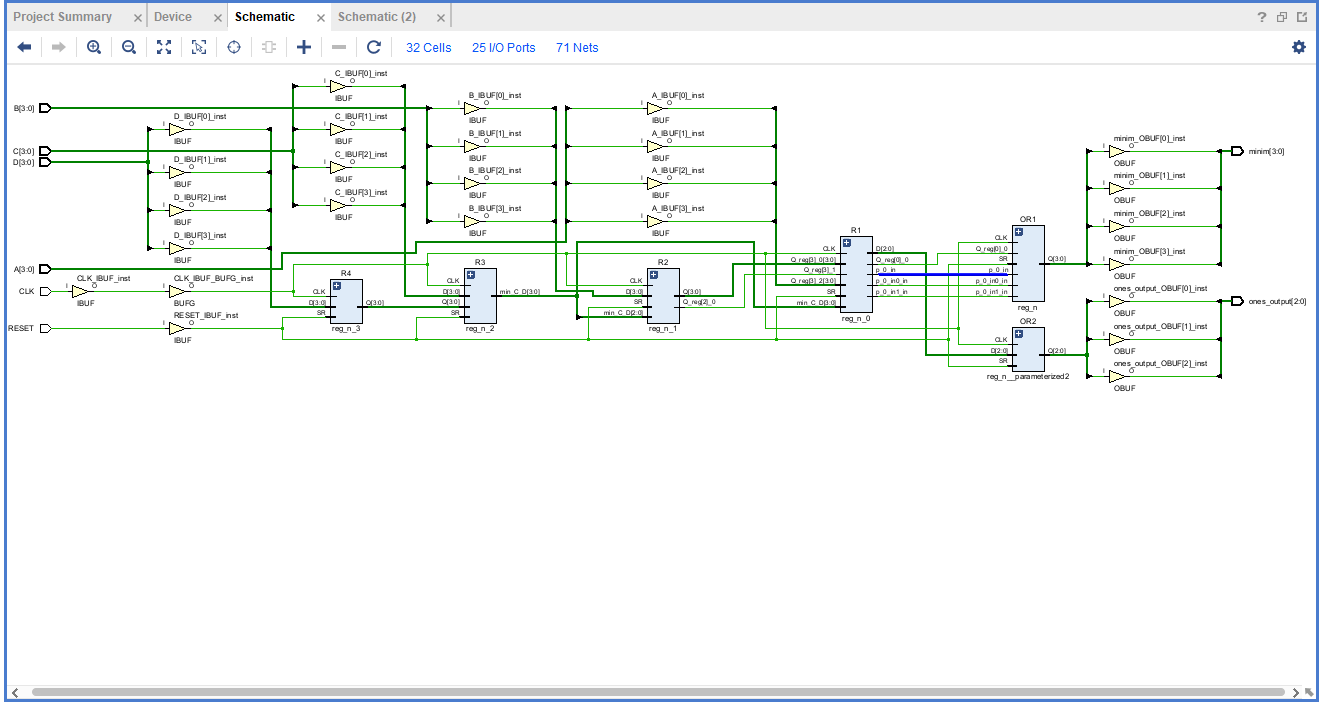
Εικόνα :Παρατηρούμε πως το περιθώριο (Slack) για τον χρόνο setup είναι 4,075 ns ενώ για τον χρόνο hold 0,311 ns.



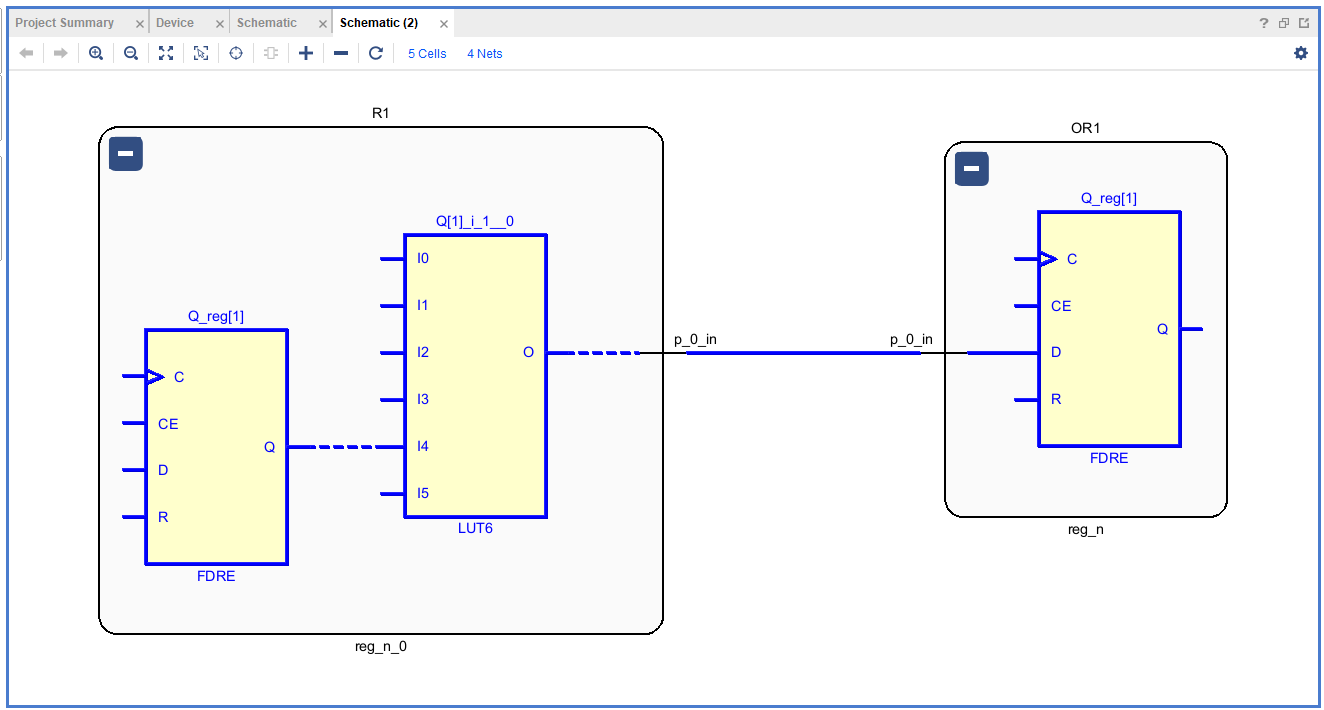
Εικόνα .1:Το χειρότερο μονοπάτι με βάση το περιθώριο (Slack) για τον χρόνο setup (path 1)



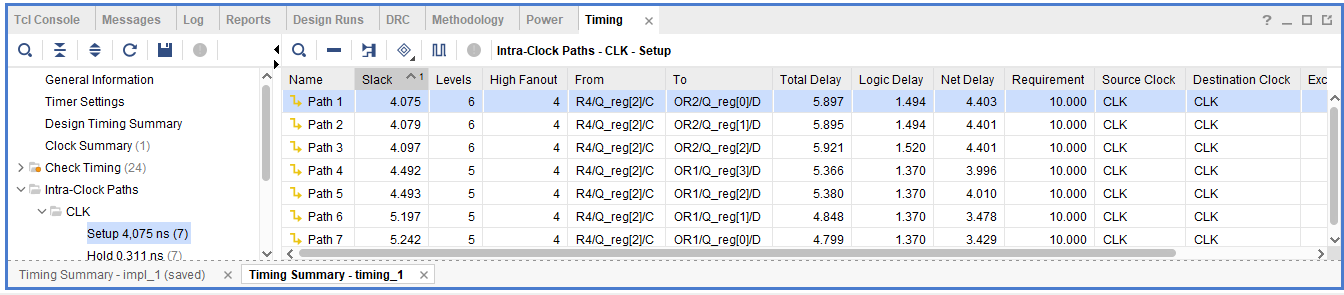
Εικόνα 2.2:Το χειρότερο μονοπάτι με βάση το περιθώριο (Slack) για τον χρόνο setup (path 1)



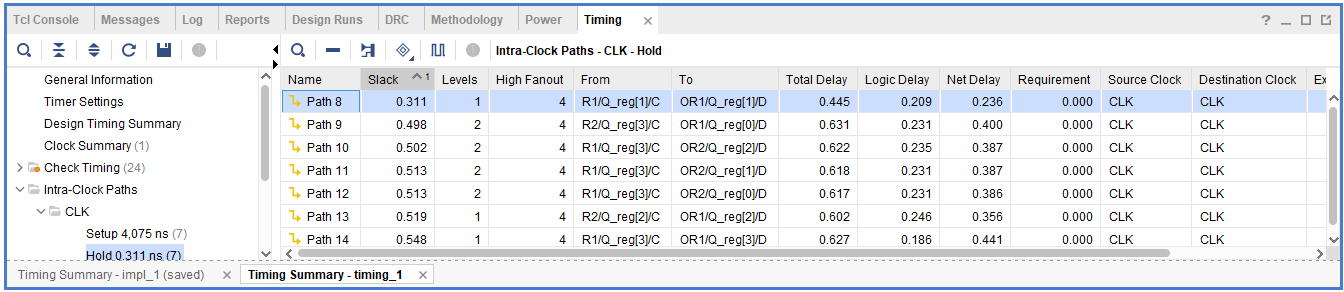
Εικόνα .1:Το χειρότερο μονοπάτι με βάση το περιθώριο (Slack) για τον χρόνο hold (path 8)



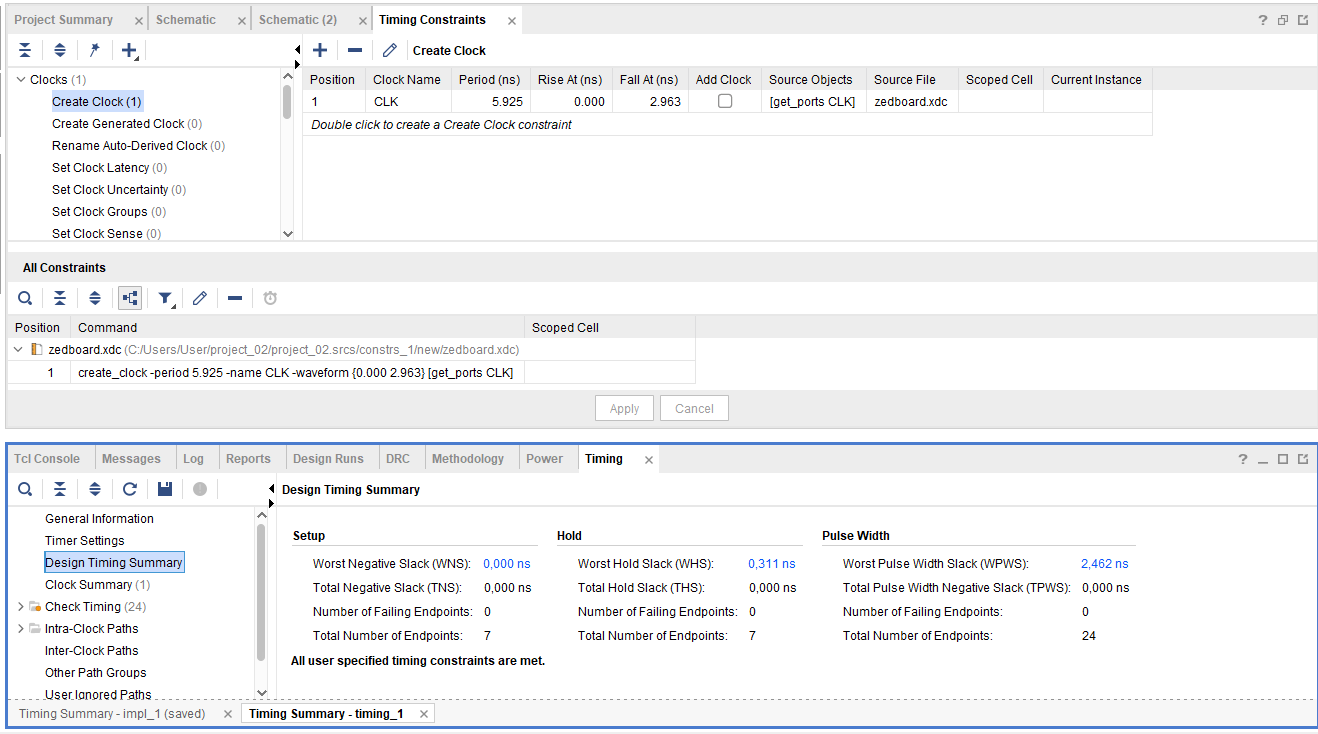
Εικόνα 3.2:Το χειρότερο μονοπάτι με βάση το περιθώριο (Slack) για τον χρόνο hold (path 8)



Εικόνα :Η καθυστέρηση διάδοσης προσδιορίζεται από το χειρότερο μονοπάτι (path 1) με βάση το περιθώριο (Slack) για τον χρόνο setup και συγκεκριμένα από το πεδίο Total Delay όπου παρατηρούμε πως αναγράφει 5,897 ns



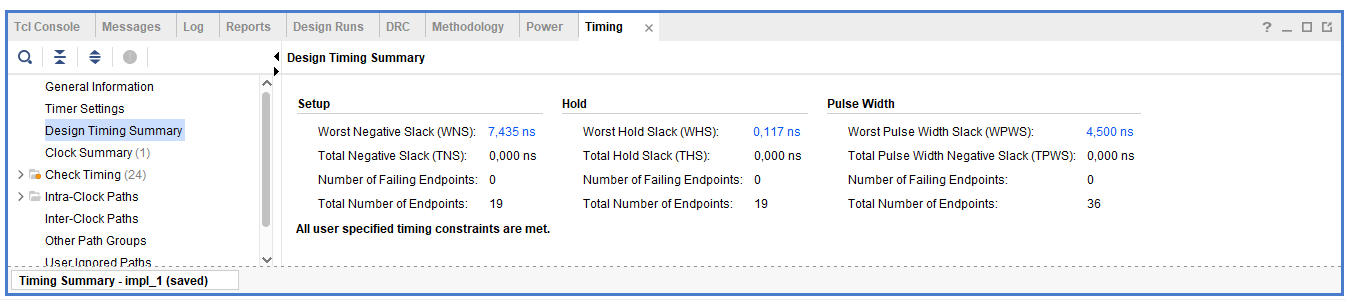
Εικόνα 5:Η καθυστέρηση μόλυνσης προσδιορίζεται από το χειρότερο μονοπάτι (path 8) με βάση το περιθώριο (Slack) για τον χρόνο hold και συγκεκριμένα από το πεδίο Total Delay όπου παρατηρούμε πως αναγράφει 0,445 ns



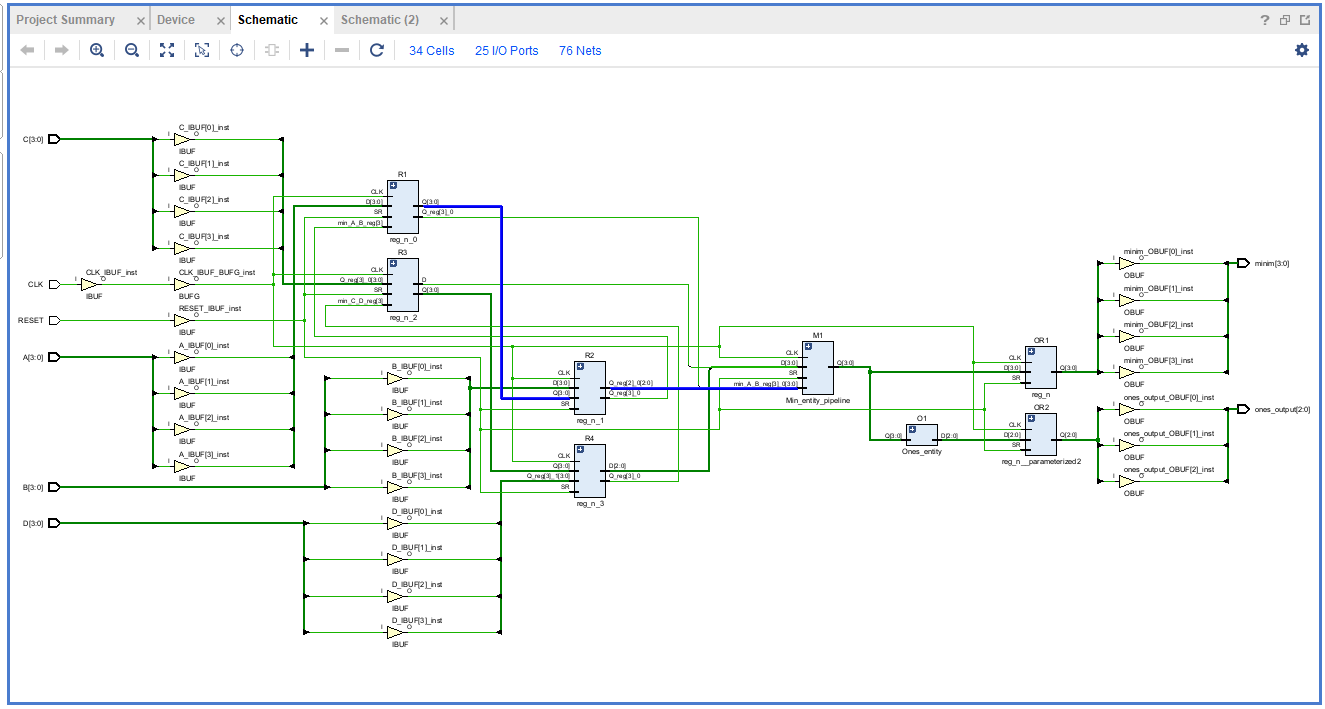
Εικόνα 6:Η μέγιστη συχνότητα λειτουργίας του κυκλώματος επιτυγχάνεται όταν το περιθώριο (Slack) του χειρότερου μονοπατιού με βάση τον χρόνο setup (path 1) προσεγγίζει το μηδέν ή και ακόμα γίνει μηδέν χωρίς να έχουμε κανένα αριθμό από failing endpoints. Άρα η μέγιστη περίοδος λειτουργίας παρατηρούμε πως είναι 5.925 ns και η μέγιστη συχνότητα λειτουργίας 0,168×109 Hz.

**β)Υλοποίηση με pipeline**

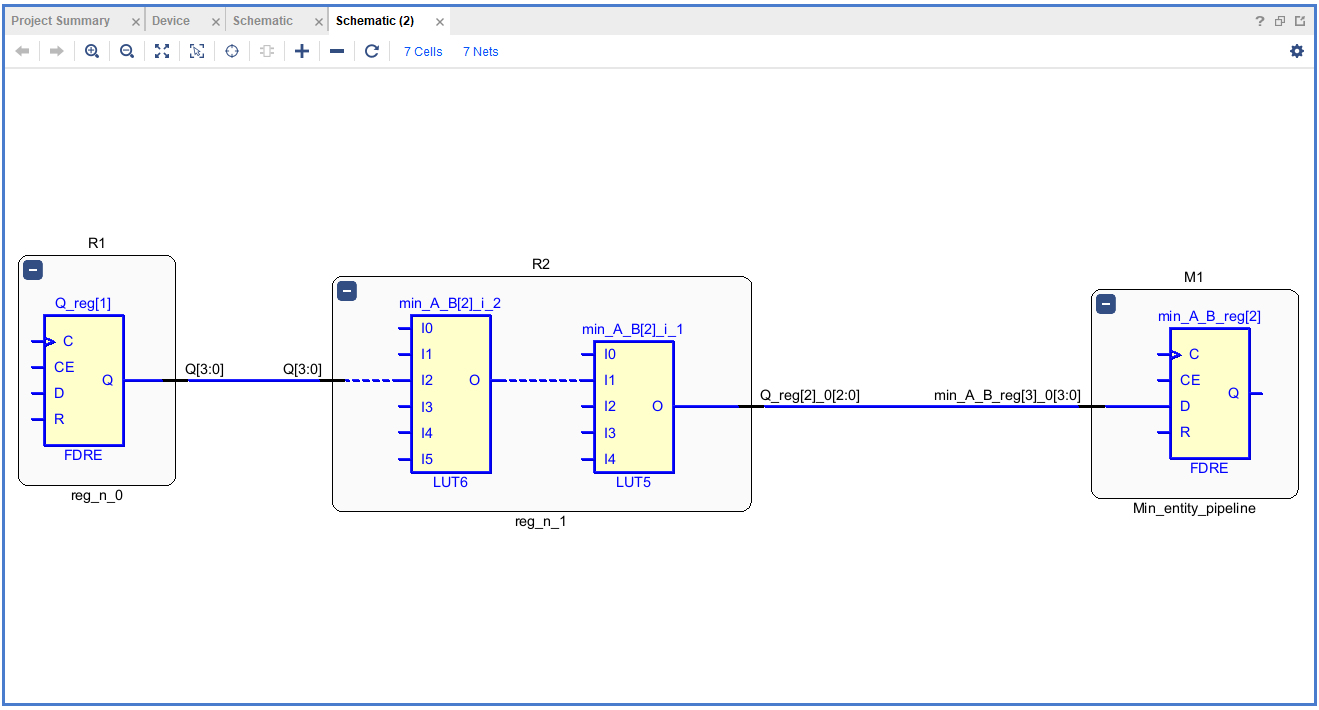
Αρχικά το clock έχει περίοδο 10 ns άρα η συχνότητα λειτουργίας είναι 108 Hz.



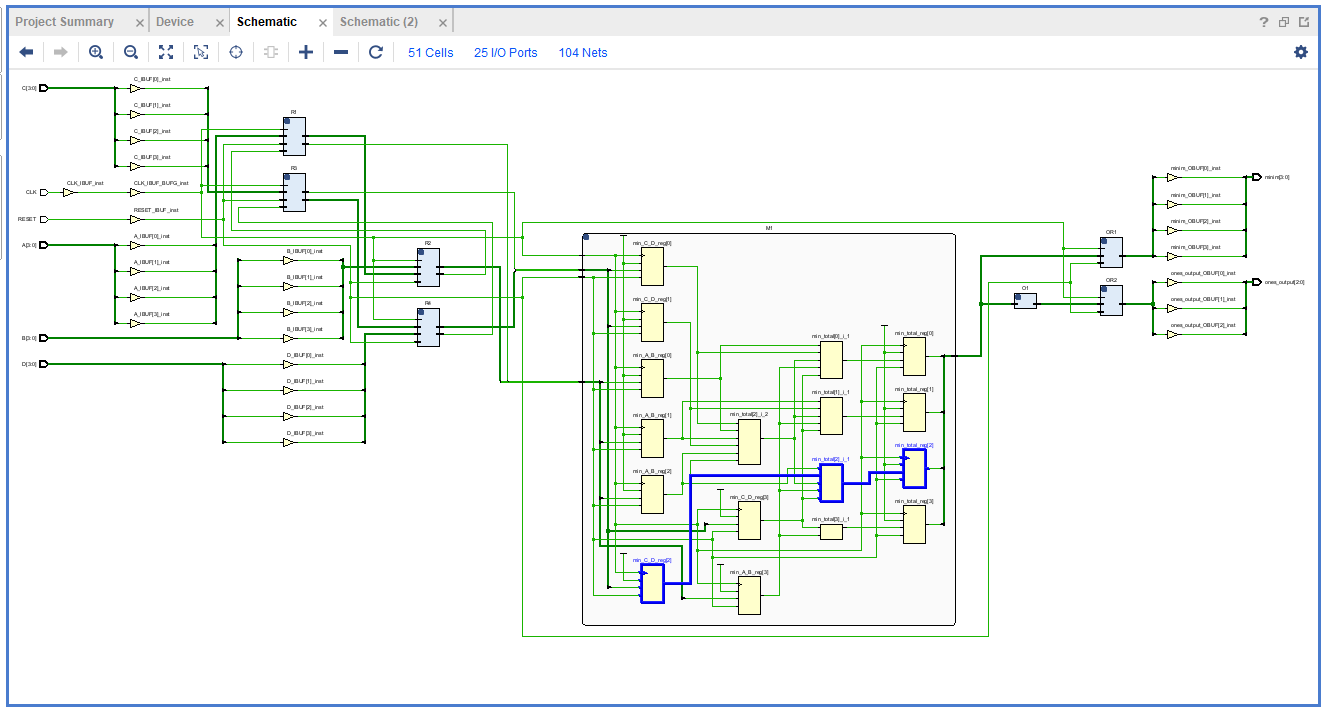
Εικόνα 7:Παρατηρούμε πως το περιθώριο (Slack) για τον χρόνο setup είναι 7,435 ns ενώ για τον χρόνο hold 0,117 ns.



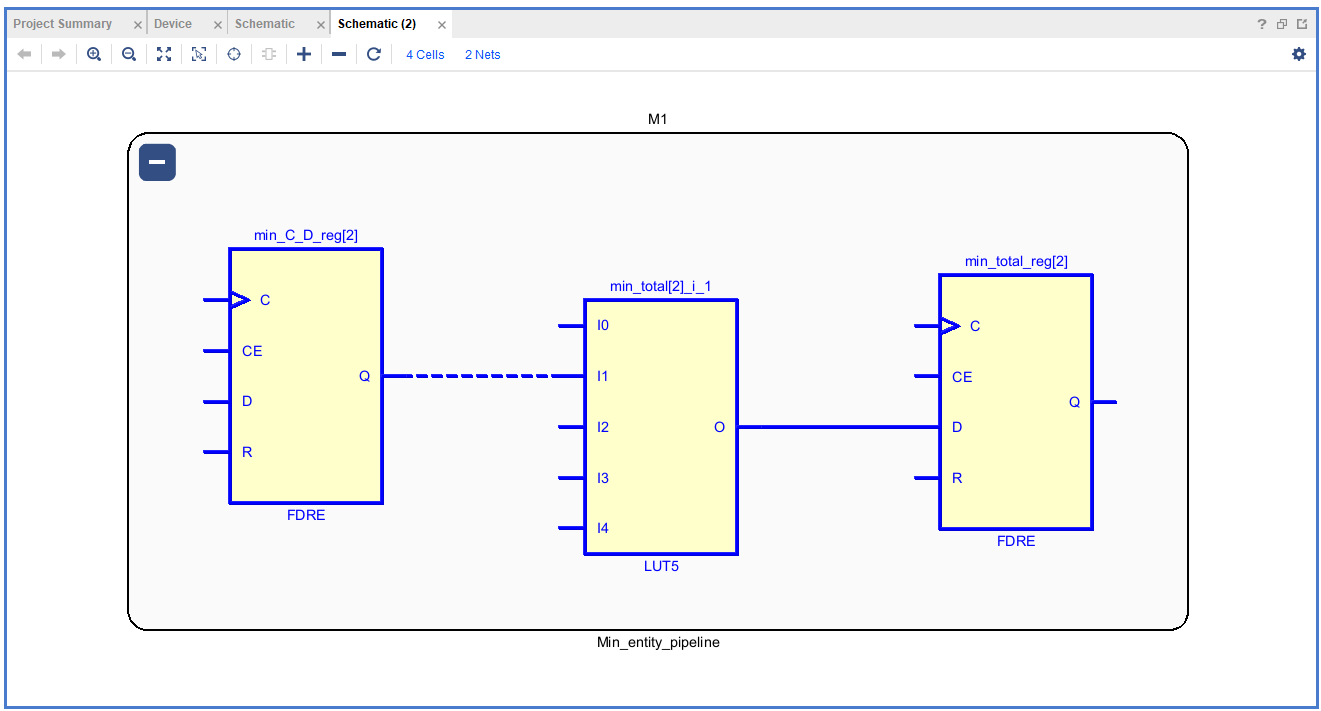
Εικόνα 8.1:Το χειρότερο μονοπάτι με βάση το περιθώριο (Slack) για τον χρόνο setup (path 1)



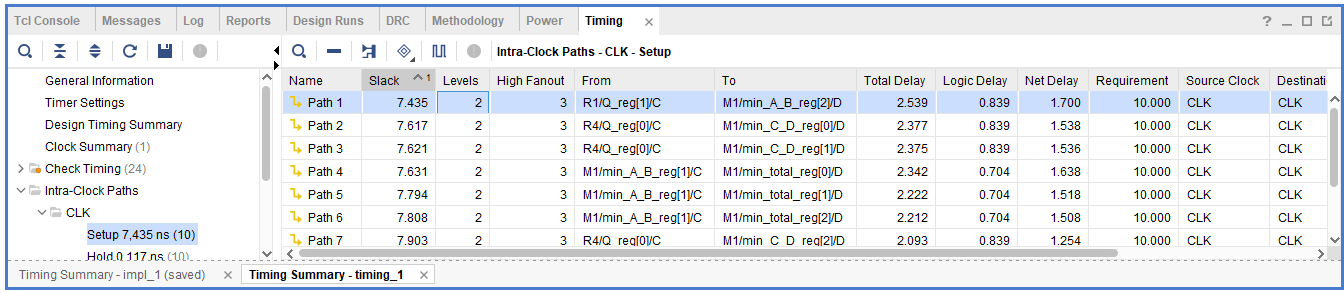
Εικόνα 8.2:Το χειρότερο μονοπάτι με βάση το περιθώριο (Slack) για τον χρόνο setup (path 1)



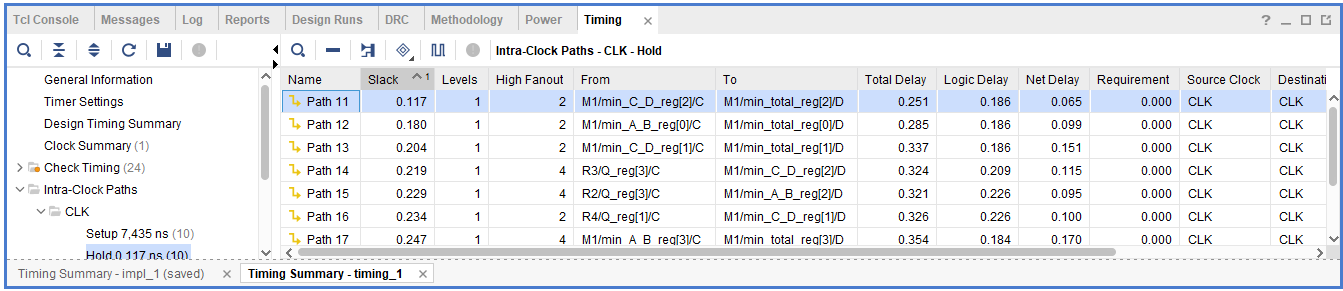
Εικόνα 9.1:Το χειρότερο μονοπάτι με βάση το περιθώριο (Slack) για τον χρόνο hold (path 11)



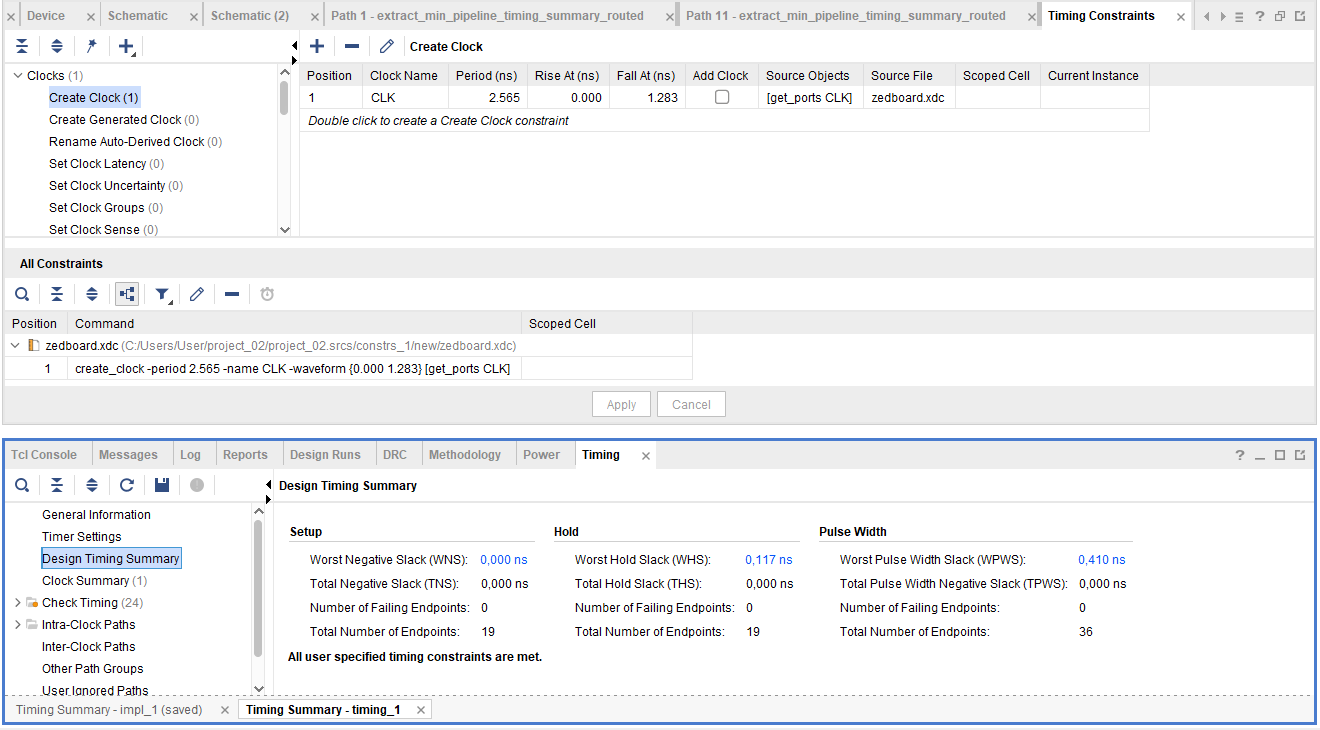
Εικόνα 9.2:Το χειρότερο μονοπάτι με βάση το περιθώριο (Slack) για τον χρόνο hold (path 11)



Εικόνα 10:Η καθυστέρηση διάδοσης προσδιορίζεται από το χειρότερο μονοπάτι (path 1) με βάση το περιθώριο (Slack) για τον χρόνο setup και συγκεκριμένα από το πεδίο Total Delay όπου παρατηρούμε πως αναγράφει 2.539 ns



Εικόνα 11:Η καθυστέρηση μόλυνσης προσδιορίζεται από το χειρότερο μονοπάτι (path 11) με βάση το περιθώριο (Slack) για τον χρόνο hold και συγκεκριμένα από το πεδίο Total Delay όπου παρατηρούμε πως αναγράφει 0,251 ns



Εικόνα 12:Η μέγιστη συχνότητα λειτουργίας του κυκλώματος επιτυγχάνεται όταν το περιθώριο (Slack) του χειρότερου μονοπατιού με βάση τον χρόνο setup (path 1) προσεγγίζει το μηδέν ή και ακόμα γίνει μηδέν χωρίς να έχουμε κανένα αριθμό από failing endpoints. Άρα η μέγιστη περίοδος λειτουργίας παρατηρούμε πως είναι 2,565 ns και η μέγιστη συχνότητα λειτουργίας 0,389×109 Hz.

Παρατηρούμε λοιπόν πως η υλοποίηση με pipeline έχει υψηλότερη μέγιστη συχνότητα λειτουργίας (τουλάχιστον διπλάσια και το clock του έχει σχεδόν την μισή περίοδο) άρα το σύστημα είναι πιο γρήγορο σε σχέση με την υλοποίηση χωρίς pipeline γεγονός που είναι απολύτως λογικό αφού στο data path προσθέσαμε κάποιους καταχωρητές ώστε να το διαιρέσουμε σε μικρότερα κομμάτια έτσι ώστε να μικρύνει η καθυστέρηση διάδοσης μεταξύ δύο διαδοχικών καταχωρητών με αποτέλεσμα να μικρύνει κατάλληλα και η απαιτούμενη περίοδος του clock (προσέχοντας να μην έχουμε επικαλύψεις δεδομένων) που σημαίνει πως θα διοχετεύονται πιο γρήγορα καινούργια δεδομένα στο σύστημα ενώ σε κάθε τμήμα του συστήματος (διαδρομή μεταξύ δύο διαδοχικών καταχωρητών) θα βρίσκεται ένα διαφορετικό σετ δεδομένων που θα επεξεργάζεται, κατορθώνοντας έτσι στην έξοδο του συστήματος να έχουμε μεγαλύτερη παραγωγή δεδομένων εξόδου ανά μονάδα χρόνου. Προφανώς η υλοποίηση με pipeline χρησιμοποιεί περισσότερους πόρους από την υλοποίηση χωρίς pipeline καθώς έχουν τοποθετηθεί επιπλέον καταχωρητές.

Η υλοποίηση με pipeline πραγματοποιήθηκε στην entity όπου υπολογίζεται το ελάχιστο από το a, b, c, d σήματα εισόδου και πιο συγκεκριμένα για την σύγκριση των σημάτων, ακολουθήθηκε υλοποίηση συγκριτών και πολυπλεκτών 2 σε 1 σε δομή δέντρου (το a συγκρίνονταν με το b και παράλληλα το c συγκρίνονταν με το d και τα αποτελέσματα αυτών των δύο αυτών συγκρίσεων στην συνέχεια συγκρίνονταν παράγοντας έτσι το τελικό αποτέλεσμα) όπου μετά από κάθε πολυπλέκτη υπήρχε ένας καταχωρητής να αποθηκεύει το αποτέλεσμα της σύγκρισης (συνολικά χρειάστηκαν τρείς καταχωρητές). Η υλοποίηση αυτή καθυστερεί δύο clock το αποτέλεσμα να βγει στην έξοδο του συστήματος. Μια εναλλακτική υλοποίηση είναι αυτή των συγκριτών και πολυπλεκτών σε αλυσίδα (το a συγκρίνονταν με το b, στην συνέχεια το αποτέλεσμα της σύγκρισης αυτής συγκρίνονταν με το c και τέλος το αποτελέσματα της δεύτερης σύγκρισης συγκρίνονταν με το d παράγοντας έτσι το τελικό αποτέλεσμα) όπου μετά από κάθε πολυπλέκτη θα υπήρχε ένας καταχωρητής να αποθηκεύει το αποτέλεσμα της σύγκρισης (επίσης συνολικά χρειάστηκαν τρείς καταχωρητές). Η υλοποίηση αυτή καθυστερεί τρία clock το αποτέλεσμα να βγει στην έξοδο του συστήματος ωστόσο πιθανώς να βελτιώνει την μέγιστη συχνότητα λειτουργίας του συστήματος μειώνοντας κι άλλο την περίοδο του clock, κατορθώνοντας έτσι στην έξοδο του συστήματος να έχουμε ίσως ακόμη μεγαλύτερη παραγωγή δεδομένων εξόδου ανά μονάδα χρόνου. Για να διαπιστώσουμε ποια από τις δύο υλοποιήσεις οδηγεί το σύστημα σε μεγαλύτερη συχνότητα λειτουργίας θα πρέπει να υλοποιήσουμε και να συγκρίνουμε τις μέγιστες συχνότητες λειτουργίας τους.