



ΕΛΛΗΝΙΚΟ ΑΝΟΙΚΤΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ

ΣΥΝΤΟΜΟ ΠΡΟΓΡΑΜΜΑ ΣΠΟΥΔΩΝ «ΣΧΕΔΙΑΣΗ ΕΝΣΩΜΑΤΩΜΕΝΩΝ
ΣΥΣΤΗΜΑΤΩΝ ΚΑΙ ΕΦΑΡΜΟΓΕΣ ΜΙΚΡΟΕΛΕΓΚΤΩΝ ΣΤΟ ΔΙΑΔΙΚΤΥΟ ΤΩΝ
ΠΡΑΓΜΑΤΩΝ (IoT)»
ΑΚΑΔΗΜΑΪΚΟ ΕΤΟΣ: 2019-20

ΣΤΟΙΧΕΙΑ ΠΟΥ ΣΥΜΠΛΗΡΩΝΕΙ Ο ΦΟΙΤΗΤΗΣ

ΟΝΟΜΑΤΕΠΩΝΥΜΟ	<ονοματεπώνυμο>
Α/Α ΓΡΑΠΤΗΣ ΕΡΓΑΣΙΑΣ	2η ΕΡΓΑΣΙΑ
ΗΜΕΡΟΜΗΝΙΑ ΑΠΟΣΤΟΛΗΣ	Click here to enter a date.
ΣΧΟΛΙΑ ΠΡΟΣ ΚΑΘΗΓΗΤΗ	

Ημερομηνία ανακοίνωσης εργασίας: 13/12/2019

Ημερομηνία παράδοσης εργασίας: 8/1/2019



**ΣΥΝΤΟΜΟ ΠΡΟΓΡΑΜΜΑ ΣΠΟΥΔΩΝ «ΣΧΕΔΙΑΣΗ ΕΝΣΩΜΑΤΩΜΕΝΩΝ
ΣΥΣΤΗΜΑΤΩΝ ΚΑΙ ΕΦΑΡΜΟΓΕΣ ΜΙΚΡΟΕΛΕΓΚΤΩΝ ΣΤΟ ΔΙΑΔΙΚΤΥΟ ΤΩΝ
ΠΡΑΓΜΑΤΩΝ (IoT)»
ΑΚΑΔΗΜΑΪΚΟ ΕΤΟΣ: 2019-20**

ΣΤΟΙΧΕΙΑ ΠΟΥ ΣΥΜΠΛΗΡΩΝΕΙ Ο ΚΑΘΗΓΗΤΗΣ

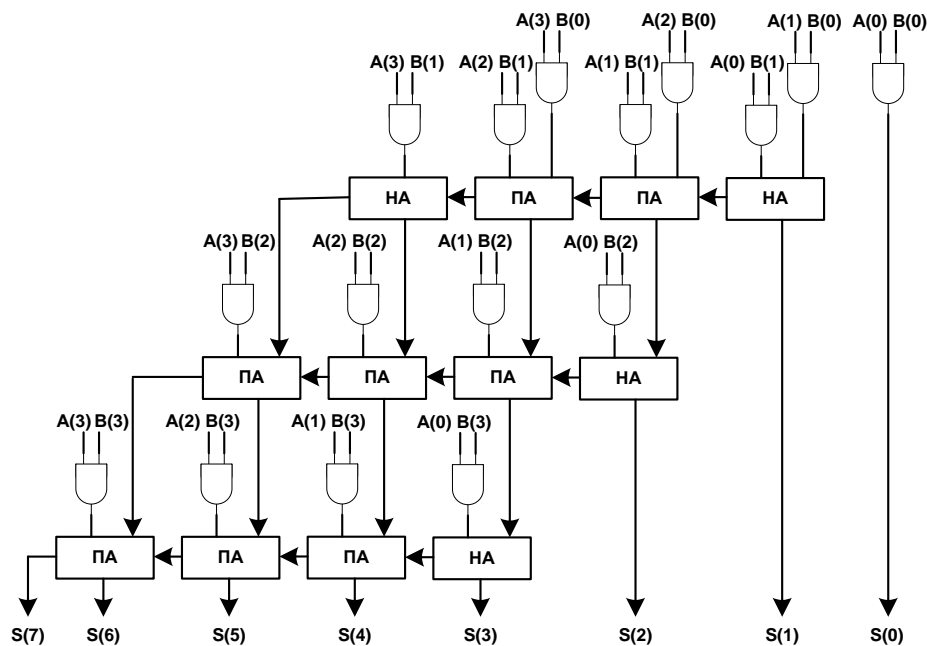
ΟΝΟΜΑΤΕΠΩΝΥΜΟ	
ΗΜΕΡΟΜΗΝΙΑ ΑΞΙΟΛΟΓΗΣΗΣ	Click here to enter a date.
ΒΑΘΜΟΣ	<αριθμητικώς> (<ολογράφως>)

ΣΧΟΛΙΑ ΠΡΟΣ ΦΟΙΤΗΤΗ / ΦΟΙΤΗΤΡΙΑ:



ΑΣΚΗΣΗ 1 (20 ΜΟΝΑΔΕΣ)

Να γραφεί ο κώδικας VHDL του πολλαπλασιαστή των 4-bit, του παρακάτω σχήματος, χρησιμοποιώντας ημιαθροιστές (HA) και πλήρεις αθροιστές (ΠΑ) του 1-bit. Τα δεδομένα εισόδου και εξόδου να είναι τύπου `std_logic` και `std_logic_vector`.



Στη συνέχεια να υλοποιήσετε τον πολλαπλασιαστή στην εργαστηριακή πλακέτα, DSD-I1 και να επαληθεύσετε τη λειτουργία του με διάφορες τιμές εισόδων απεικονίζοντας το αποτέλεσμα σε οκτώ διαδοχικά LEDs της πλακέτας. Τις τιμές των εισόδων να τις δώσετε μέσω των ακροδεκτών (FPGA pins του HD2) της πλακέτας.

Να συμπεριλάβετε στα παραδοτέα τον συμπιεσμένο φάκελο του project που δημιουργήσατε στο Quartus και να συμπεριλάβετε δύο - τρεις φωτογραφίες από το κύκλωμα σε λειτουργία για τουλάχιστον δύο συνδυασμούς της επιλογής σας σημάτων εισόδου σας.

ΑΣΚΗΣΗ 2 (5 + 10 + 5 + 5 = 25 ΜΟΝΑΔΕΣ)

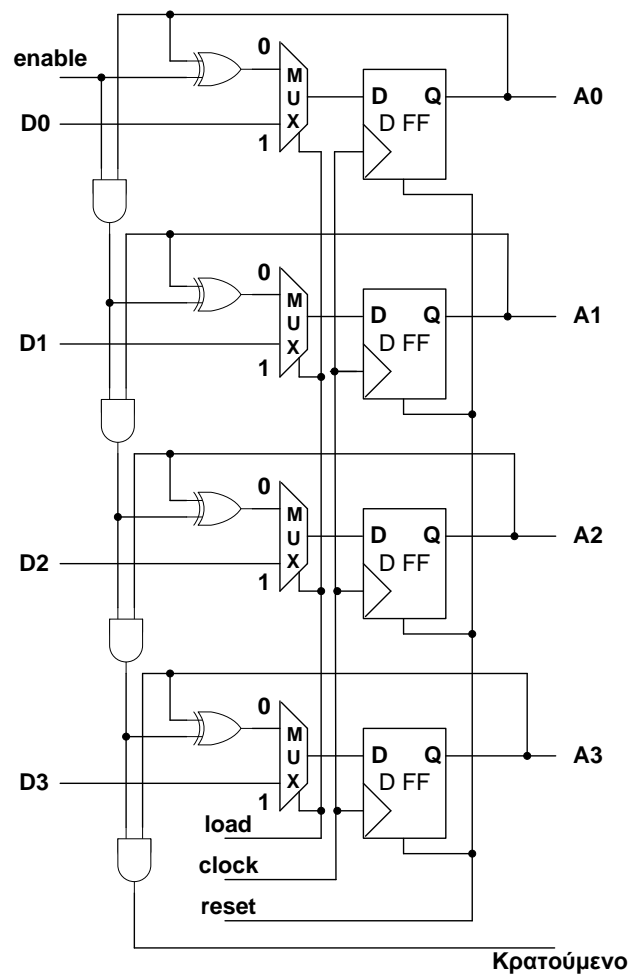
α) Η εργαστηριακή πλακέτα, DSD-I1 έχει συχνότητα ρολογιού ίση με 50 MHz. Να δημιουργήσετε τον κώδικα VHDL ενός διαιρέτη συχνότητας που παράγει ένα νέο ρολόι με συχνότητα 1 Hz.

β) Να γραφεί ο κώδικας VHDL (σε structural μορφή) του δυαδικού μετρητή με παράλληλη φόρτωση των 4-bit, του παρακάτω σχήματος, χρησιμοποιώντας τέσσερα D flip flops, 4 πολυπλέκτες και τις κατάλληλες λογικές πύλες. Η ιδιαίτερη λειτουργία του μετρητή έχει να κάνει με την αρχική τιμή που μπορεί να φορτωθεί για να μετρήσει από εκεί και πάνω. Αυτό

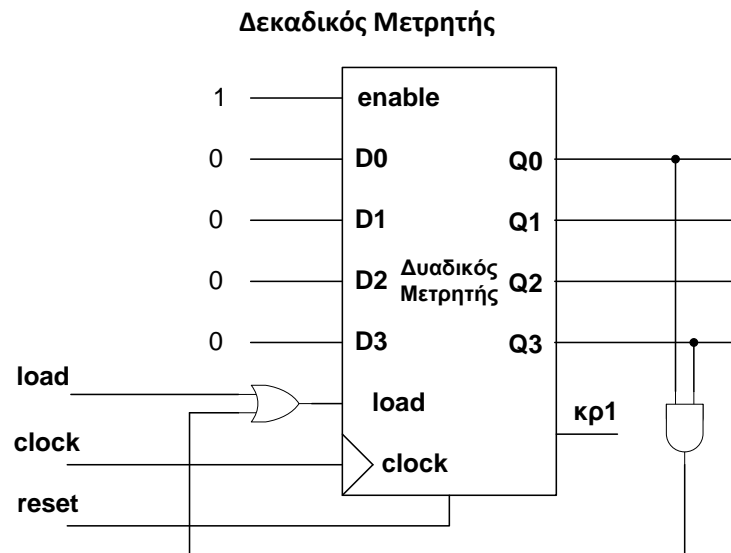


γίνεται όταν το σήμα load γίνει 1 για ένα παλμό ρολογιού και ταυτόχρονα δίνεται την αρχική τιμή μέτρησης στην παράλληλη είσοδο D(0:3). Αν θέλετε ο μετρητής να μετρά από τη τιμή 0000 θα πρέπει το σήμα load να είναι συνεχώς στο 0.

Δυαδικός Μετρητής



γ) Να γραφεί ο κώδικας VHDL (σε structural μορφή) του δεκαδικού μετρητή (BCD) του ενός ψηφίου, του παρακάτω σχήματος, χρησιμοποιώντας τον δυαδικό μετρητή του ερωτήματος β και τις κατάλληλες λογικές πύλες. Το σήμα ρολογιού του δεκαδικού μετρητή να είναι το ρολόι του 1 Hz του ερωτήματος α.

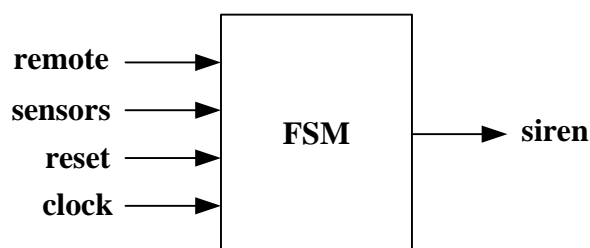


δ) Την έξοδο του μετρητή BCD να την αναπαραστήσετε στο 7-segment display, συνδεδεμένο σε διαθέσιμους ακροδέκτες της εργαστηριακής πλακέτας DSD-I1 της επιλογής σας. Ο κώδικας για την αναπαράσταση στο 7-segment display να είναι behavioral.

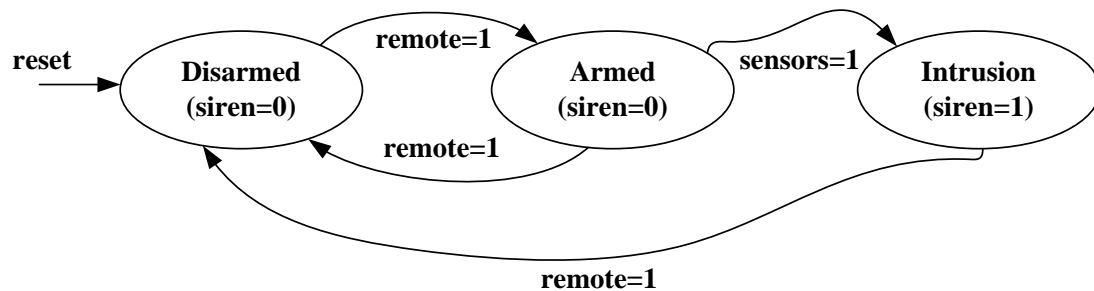
Να συμπεριλάβετε στα παραδοτέα τον συμπιεσμένο φάκελο του project που δημιουργήσατε στο Quartus και να συμπεριλάβετε από μία φωτογραφία από το κύκλωμα σε λειτουργία.

ΑΣΚΗΣΗ 3 (10 + 5 = 15 ΜΟΝΑΔΕΣ)

Παρακάτω δίνεται το μπλοκ διάγραμμα ενός συναγερμού ενός αυτοκινήτου.



Το σήμα remote ενεργοποιεί ή απενεργοποιεί τον συναγερμό. Το σήμα sensors είναι το σήμα που δέχεται η FSM αν ενεργοποιηθεί κάποιος από τους αισθητήρες του αυτοκινήτου. Το σήμα siren είναι η έξοδος του συναγερμού και δείχνει ότι έχει ενεργοποιηθεί ο συναγερμός. Το διάγραμμα καταστάσεων αυτού του συστήματος συναγερμού φαίνεται παρακάτω.



α) Να γράψετε τον κώδικα VHDL αυτής της FSM που περιγράφει το σύστημα συναγερμού του αυτοκινήτου.

β) Να δημιουργήσετε ένα testbench και να εξομοιώσετε τη λειτουργία του συναγερμού. Το σήμα remote να το έχετε ενεργοποιημένο μέχρι δύο παλμούς ρολογιού.

Να δώσετε μερικά στιγμιότυπα από τις κυματομορφές.

ΑΣΚΗΣΗ 4 (5 + 5 + 5 + 5 = 20 ΜΟΝΑΔΕΣ)

Δίδεται το ακόλουθο πρόγραμμα σε συμβολική γλώσσα του Nios II, όπου στην θέση μνήμης constx βρίσκεται αποθηκευμένος ένας φυσικός αριθμός x στο διάστημα [0,7].

- Αντικαταστήστε τα κενά στο πρόγραμμα με τιμές της επιλογής σας ώστε να αρχικοποιήσετε την θέση μνήμης constx με ένα φυσικό αριθμό, όπως ορίστηκε παραπάνω.
- Εκτελέστε τον κώδικα σε προσομοιωτή της επιλογής σας και παραθέστε αποτελέσματα οθόνης, όπου να φαίνονται τα ενδιάμεσα και τελικά αποτελέσματα (τιμές καταχωρητών και μνήμης).
- Για κάθε γραμμή του προγράμματος (εκτός των assembler directives) δώστε με την μορφή σχολίου μια σύντομη επεξήγηση της λειτουργίας που επιτελούν.
- Δώστε τη μαθηματική σχέση που συνδέει το αποτέλεσμα που αποθηκεύεται στη θέση μνήμης consty (δηλαδή εκφράστε την συνάρτηση $y=f(x)$).

```
.data
constx: .word 0x_____
consty: .word 0x00000000

.text
.global _start
_start:

movia r4, constx
ldw r5, (r4)
ldw r6, (r4)
mov r7, r0
addi r8, r0, 0x1
Loop:
add r7, r7, r5
```



```
sub r6,r6,r8
bne r6,r0,loop
mov r6,r7
rol r7,r7,0x1
add r7,r7,r6
mov r6,r7
ldw r7,(r4)
rol r7,r7,0x1
rol r7,r7,0x1
ldw r10,(r4)
add r7,r7,r10
addi r10,r0,0x13
sub r7,r7,r10
add r7,r7,r6
addi r4,r0,0x4
stw r7,0(r4)
```



ΑΣΚΗΣΗ 5 (10 + 10 = 20 ΜΟΝΑΔΕΣ)

Σχεδιάστε παραμετροποιώντας κατάλληλα ένα σύστημα επεξεργαστή NIOS II με τα βασικά περιφερειακά, όπως αυτά έχουν παρουσιαστεί στον οδηγό σχεδιασμού ενσωματωμένων συστημάτων βασισμένων στον πυρήνα NIOS-II.

- α) Υλοποιήστε στο παραπάνω σύστημα πρόγραμμα, το οποίο θα ελέγχει ένα από τα διαθέσιμα LED της πλακέτας DSD-i1 μέσω μιας εισόδου σε ένα από τα διαθέσιμα pins. Η λειτουργία του θα πρέπει να είναι τύπου “toggle” που σημαίνει ότι με κάθε μετάβαση του σήματος στην είσοδο από χαμηλή σε υψηλή στάθμη θα πρέπει να αλλάζει η κατάσταση του LED (αναμμένο σε σβησμένο και το αντίστροφο).
- β) Τροποποιήστε το σύστημα του ερωτήματος Α κατάλληλα ώστε να προστεθεί μια καθυστέρηση ενεργοποίησης του LED κατά 1 δευτερόλεπτο. Η απενεργοποίηση του LED πρέπει να γίνεται χωρίς επιπλέον καθυστέρηση.