# Logo Description automatically generated

**Κωνσταντίνος Νικολός**

**2019030096**

# Full Adder N-bits

**Σκοπός της άσκησης**: Η τριβή με τις βασικές αρχές σύνθεσης και δημιουργίας βιβλιοθηκών με σκοπό την μελλοντική χρήση τους σε μεγαλύτερα project

**Περιγραφή της άσκησης**: Προς το παρόν, χρησιμοποιείται σαν δομικό στοιχείο ένας FULL\_ADDER τριών εισόδων και δύο εξόδων με σκοπό τη δημιουργία ενός αθροιστή διάδοσης κρατουμένου n-bits. Αντλώντας τα ζητούμενα στοιχεία από κάθε FULL\_ADDER δημιουργείται η δομή nelist η οποία τελικά τυπώνεται σε ένα νέο αρχείο txt για να δοθεί στον χρήστη.

**Δεδομένα εισόδου:**

Για την απλότητα της συγκεκριμένης άσκησης οι είσοδοι του προγράμματος δεν αντλούνται από κάποιο αρχείο αλλά δημιουργούνται από την αρχή και τοποθετούνται στην δομή netlist\_nbits.

**Περιγραφή λειτουργίας του προγράμματος:**

* Αρχικά ζητείται από τον χρήστη ο αριθμός bits του αθροιστή. Δημιουργείται η δομή netlist\_nbits και οι είσοδοι του προγράμματος με την μέθοδο**:**

**netlist\_nbits\* createNelist()**

* Δημιουργείται ο πρώτος FULL\_ADDER με εισόδους Α0,Β0,Cin και στη συνέχεια οι υπόλοιποι με εισόδους Αi,Bi,ui\_Cout χρησιμοποιώντας την μέθοδο:

**Full\_adder\* createfull\_ader()**

* Κάθε FULL\_ADDER τοποθετείται στην μεταβλητή Full\_adder\*\* addersm της δομής netlist\_nbits
* Δημιουργείται ένας πίνακας από strings όπου τοποθετείται το ζητούμενο netlist με σκοπό την εκτύπωση του σε νέο αρχείο με τη μέθοδο

**Char\*\* createTable()**

* Δημιουργείται το αρχείο που περιέχει το netlist με όνομα netlist\_final.txt
* Απελευθερώνεται ο χώρος που δεσμεύτηκε για την δημιουργία του netlist χρησιμοποιώντας τις μεθόδους:

**void freeAdder() , void freeNetlist()**

* Δομές που χρησιμοποιήθηκαν**: netlist\_nbits , Full\_adder**