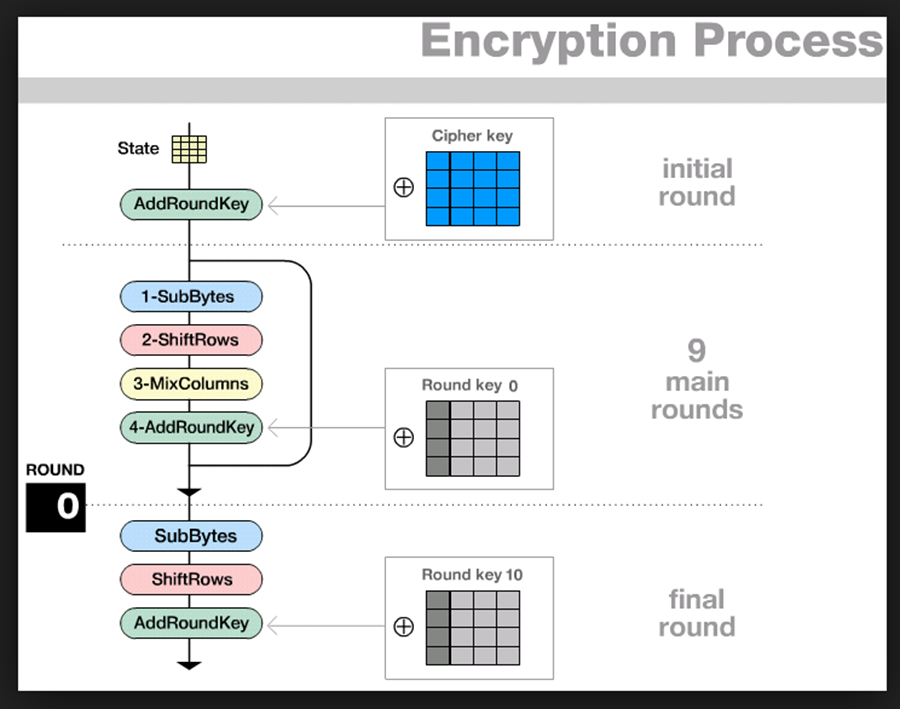
Στο αρχείο κείμενο ανάλυση αλγοριθμού AES αναλύουμε τα στάδια του αλγορίθμου.

Σε αυτό το κείμενο θα δούμε πως εκτελούνται αυτά τα στάδια καθώς και την απόδοση τους, στα κυκλώματα που κατέχουμε.

Να υπενθυμίσουμε ότι τα στάδια είναι τα παρακάτω σύμφωνα με την εικόνα:



Δηλαδή οι συναρτήσεις – διαδικασίες θα εκτελεστούν με την παρακάτω σειρά:

1. AddRoundkey
2. SubBytes
3. ShiftRows
4. MixColumns
5. AddRoundKey

…

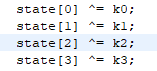
21. SubBytes

22. ShiftRows

23. AddRoundKey

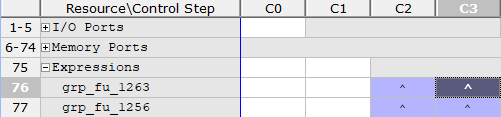
**1ο κύκλωμα:**

Η addRoundKey γίνεται στις παρακάτω εντολές:



Όπου κ είναι τα κλειδιά.

Βλέπουμε από την ανάλυση ότι όλα αυτά κρατάνε 2 κύκλους:

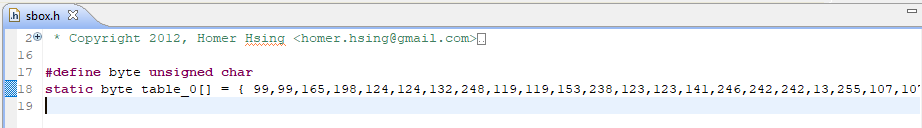


Ενώ βλέπουμε να γίνονται εδώ πράξεις τύπου xor.

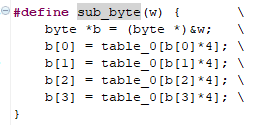
Η SubBytes γίνεται στις παρακάτω εντολές:

Αυτή η συνάρτηση χρησιμοποιεί σύμφωνα με τον αλγόριθμο τον πίνακα sbox.

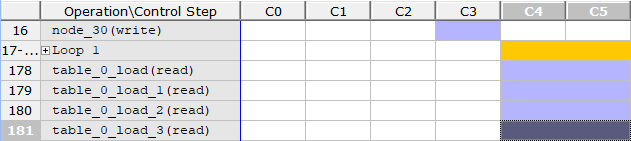
Βρίσκουμε αυτόν τον πίνακα σε αρχείο header που ονομάζεται sbox.h και έχει οριστεί ως table\_0[] όπως βλέπουμε παρακάτω:



Βλέπουμε ότι εκτελείται με την συνάρτηση sub\_byte



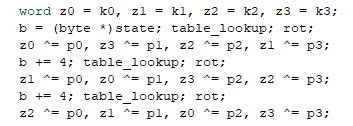
Όπως βλέπουμε από την παρακάτω εικόνα κάνει 2 κύκλους.



Συναντάμε την συνάρτηση byte καθώς γινόμενα και εντολές xor.

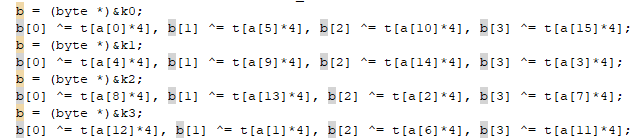
H shiftRows γίνεται στις παρακάτω εντολές:

Εδώ γίνονται οι αντιμεταθέσεις:



Συναντάμε την συνάρτηση byte καθώς και εντολές xor.

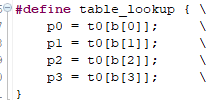
Η MixColumns γίνεται στις παρακάτω εντολές:



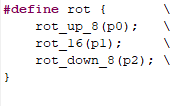
Συναντάμε την συνάρτηση byte καθώς γινόμενα και εντολές xor.

Για τον υπολογισμού του RoundKey χρησιμοποιείτε η συνάρτηση rot και table\_lookup που είναι οι εξής:

Table\_lookup:



Rot:



Με τις εξής υποσυναρτήσεις:



Εδώ βλέπουμε τις πράξεις shift left logical καθώς και shift right logical.

Δυστυχώς το εργαλείο δεν μπόρεσε να μας υπολογίζει του κύκλους που κάνουν οι συναρτήσεις shiftRows και MixColumns.

Αλλά μπορούμε να συμπεράνουμε από το latency που κάνει το loop, το οποίο σύμφωνα με την εικόνα:



Είναι 45. Δηλαδή τα άλλα 6 είναι από τις εντολές που είναι εκτός loop , άρα συμπεραίνουμε αθροιστικά ότι η ΑddRoundKey,subBytes και shiftRows καθυστερούν κατά 6 κύκλους ρολογιού.

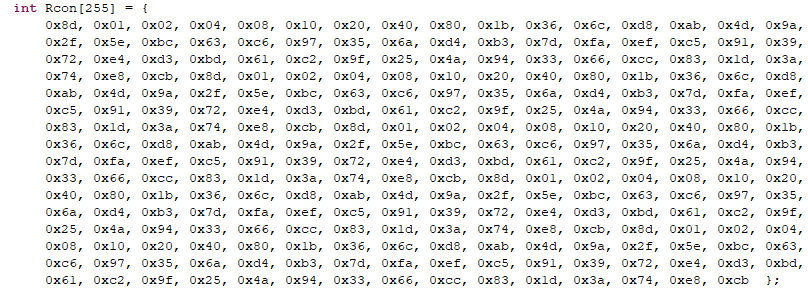
Και εφόσον η AddRounKey και η subBytes τελειώνουν σε 2 κύκλους η κάθε μία άρα και η shiftRows τελειώνει σε 2 κύκλους.

**2ο κύκλωμα:**

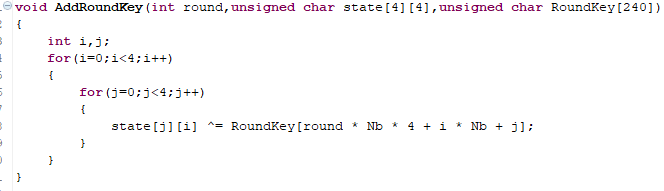
Βλέπουμε ότι το Sbox το έχει καταχωρήσει σε μια ακέραια μεταβλητή καθώς την καλεί από μία συνάρτηση:



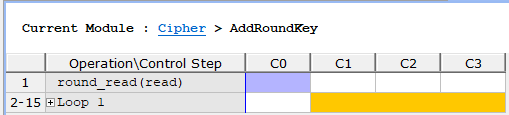
Ενώ το Rcon το έχει καταχωρήσει σε μια ακέραια μεταβλητή που ορίζεται όταν καλείται η main συνάρτηση cipher χωρίς να καλείται από συνάρτηση.



Η addRoundKey γίνεται στις παρακάτω εντολές:



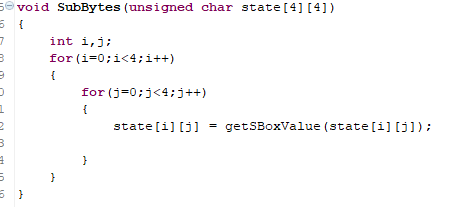
Και κάνει τους παρακάτω κύκλους:



Δηλαδή 4 κύκλους.

Βλέπουμε εκτός από τις πράξεις που κάνει ένα for loop , βλέπουμε xor καθώς και γινόμενα.

Η SubBytes γίνεται στις παρακάτω εντολές:



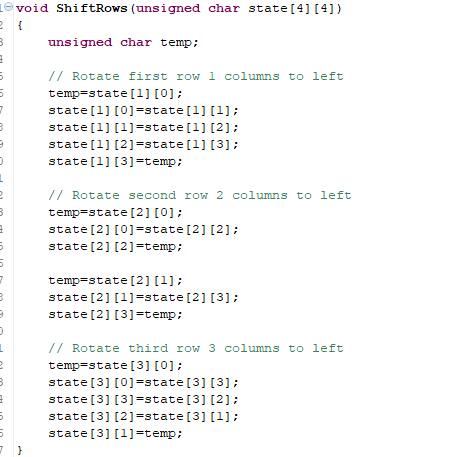
Και κάνει τους παρακάτω κύκλους:



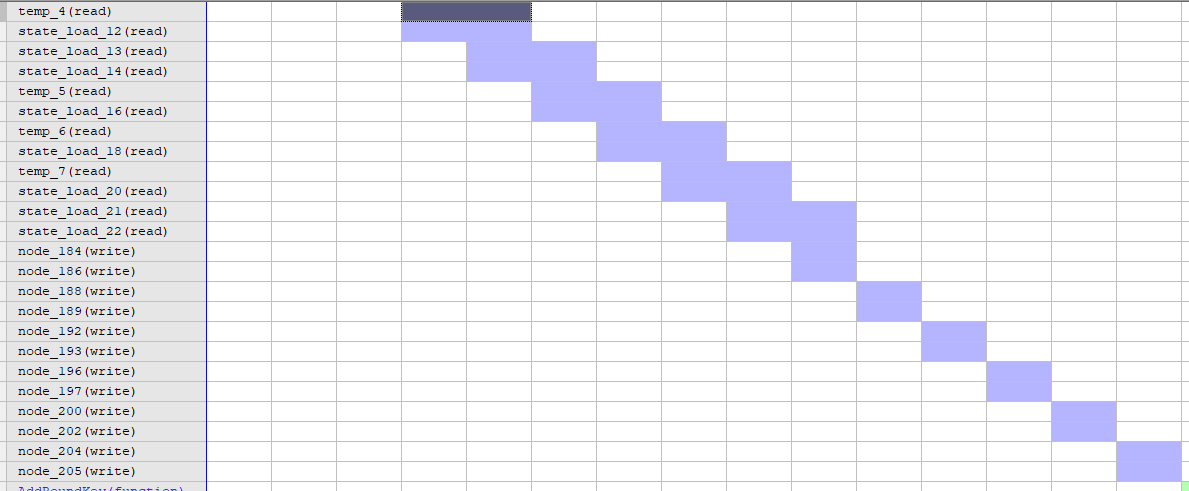
Δηλαδή 5 κύκλους.

Εδώ βλέπουμε εκτός από τις πράξεις που κάνει το ίδιο το loop, βλέπουμε ότι καλείται 16 φορές η συνάρτηση getSBoxValue.

H shiftRows γίνεται στις παρακάτω εντολές:



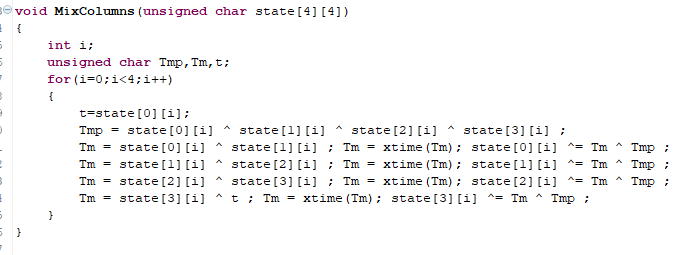
Η συνάρτηση shiftRows αρχίζει από το 8ο κύκλο και τελειώνει στο 19ο κύκλο όπως βλέπουμε στην εικόνα:



Άρα ολοκληρώνεται σε 11 κύκλους ρολογιού.

Εδώ βλέπουμε μόνο κάποιες εκχωρήσεις ανάμεσα σε κάποιες μεταβλητές.

Η MixColumns γίνεται στις παρακάτω εντολές:



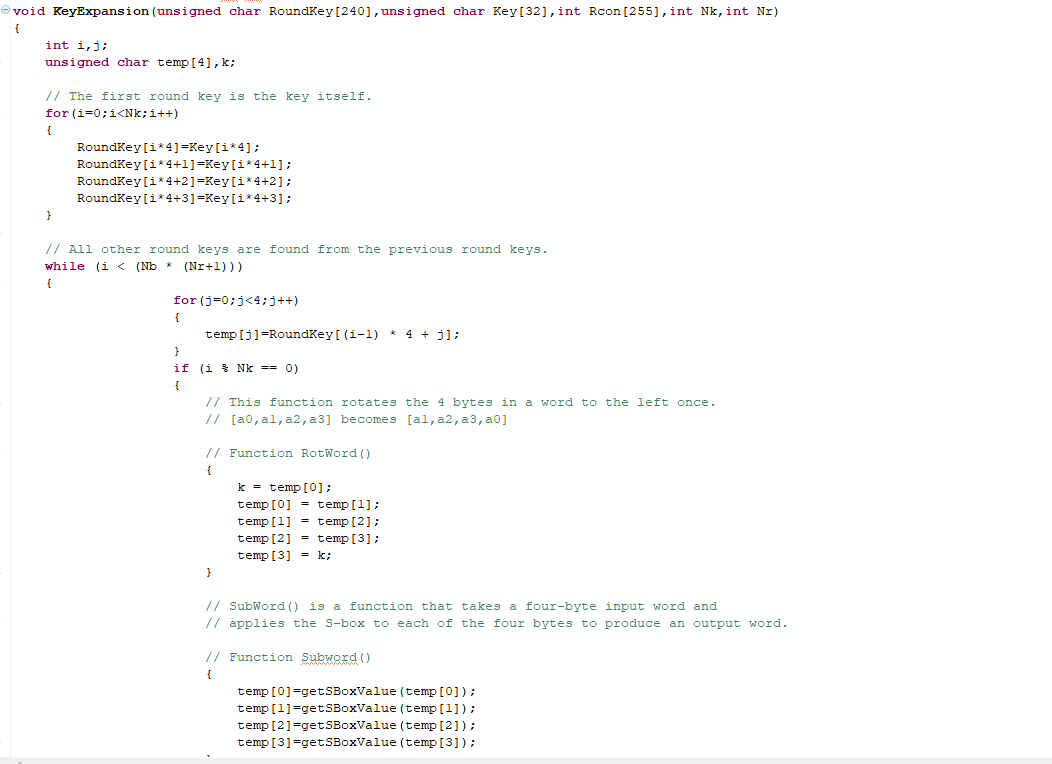
Αυτό το loop3.1 αποτελεί την συνάρτηση MixColumns:

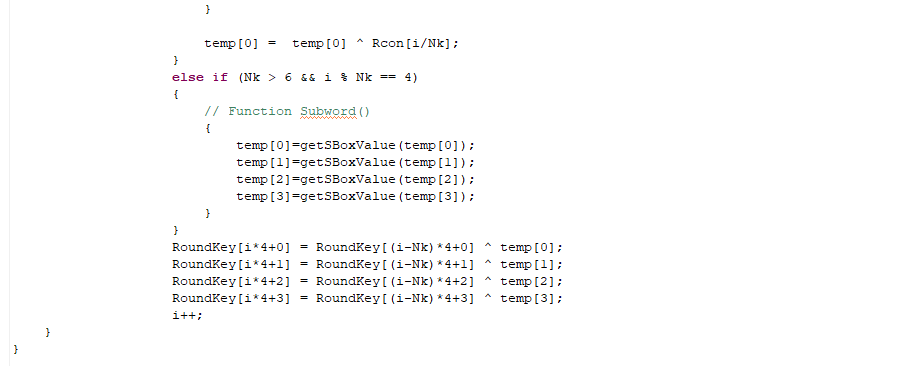


Και από τι βλέπουμε ότι ολοκληρώνεται σε 4 κύκλους ρολογιού.

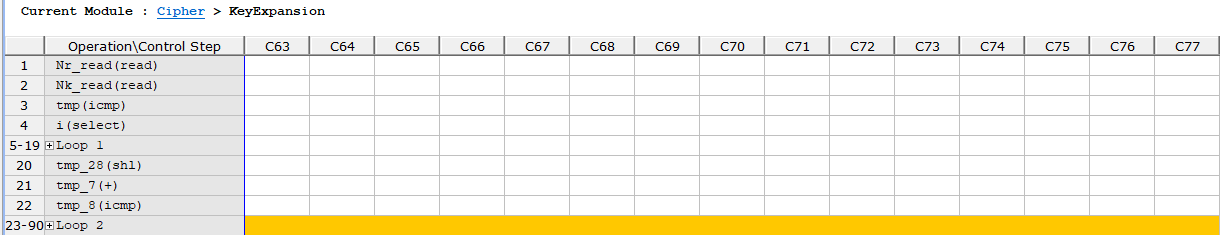
Εδώ συναντάμε εκτός από τις πράξεις που κάνει ένα loop , βλέπουμε και εντολές xor.

Για τον υπολογισμού του RoundKey χρησιμοποιείτε η συνάρτηση keyExpansion που είναι οι εξής:





Και κάνει τους παρακάτω κύκλους:



Δηλαδή 88 κύκλους.

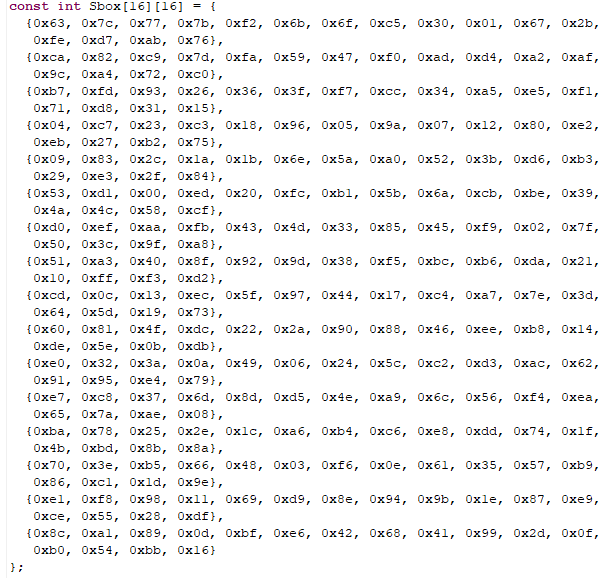
Στον παραπάνω κώδικα συναντάμε 1 for loop καθώς και 1 while loop το οποίο εμπεριέχει 1 for loop και 2 συνθήκες.

Από πράξεις εδώ , εκτός από τις πράξεις που κάνει το ίδιο loop που είναι σύγκριση και πρόσθεση, συναντάμε γινόμενα, αφαιρέσεις ακόμη και εντολές xor.

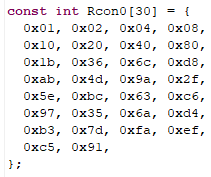
Eπιπλέον βλέπουμε ότι καλεί την συνάρτηση getSboxValue αρκετές φορές.

**3ο κύκλωμα:**

Εδώ βλέπουμε ότι το Sbox το έχουμε βάλει σε μια μεταβλητή τύπου ακεραίου:

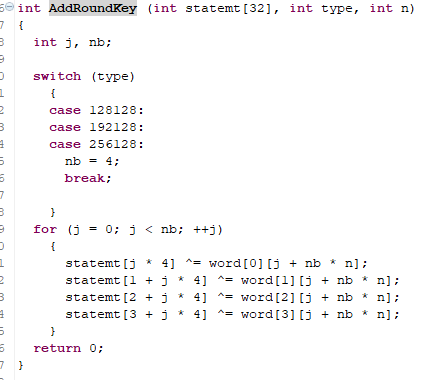


Ομοίως το Rcon:

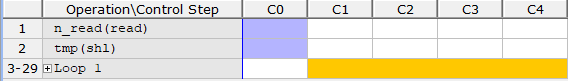


Στο συγκεκριμένο κύκλωμα έχουμε μίξη διαφόρων συναρτήσεων

Η addRoundKey γίνεται στις παρακάτω εντολές:



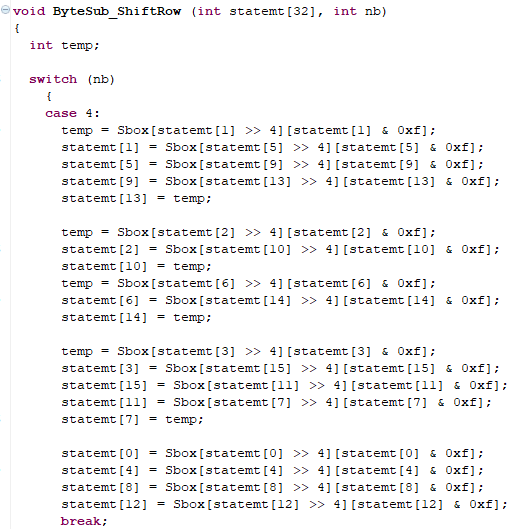
Με τους παρακάτω κύκλους:



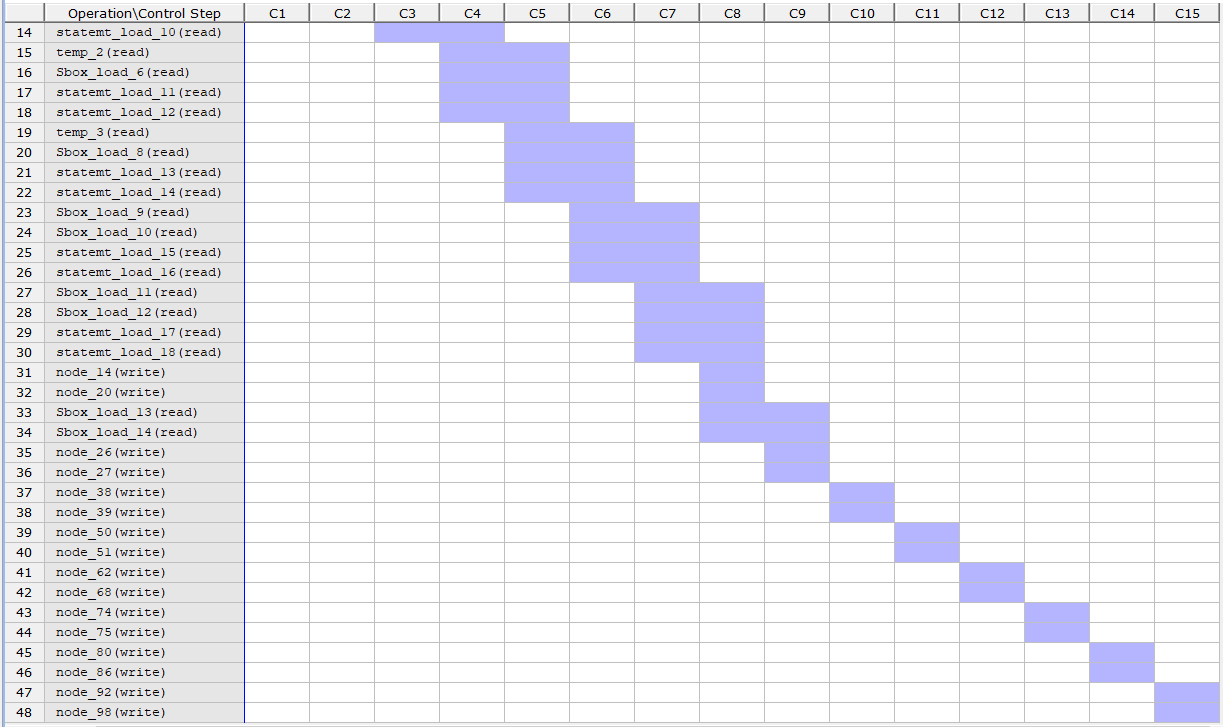
Εδώ συναντάμε εκτός από το switch και τις πράξεις που κάνει το loop, βλέπουμε και γινόμενο.

Δηλαδή 5 κύκλους για να τελειώσει.

Η SubBytes\_ShiftRowγίνεται στις παρακάτω εντολές:



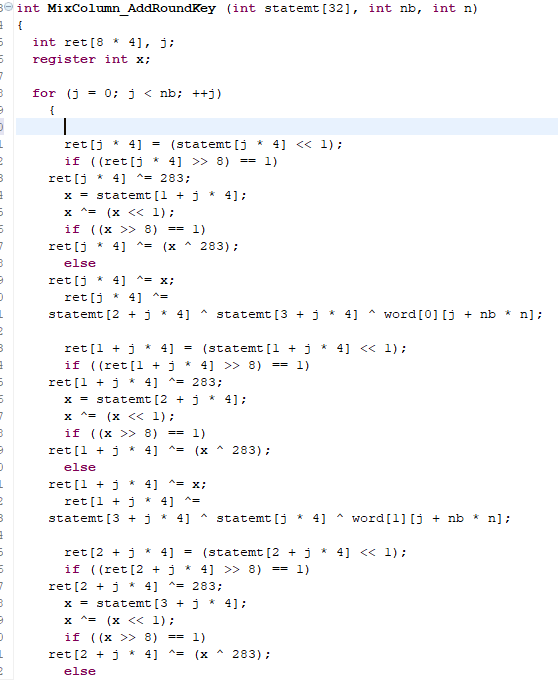
Με τους παρακάτω κύκλους:

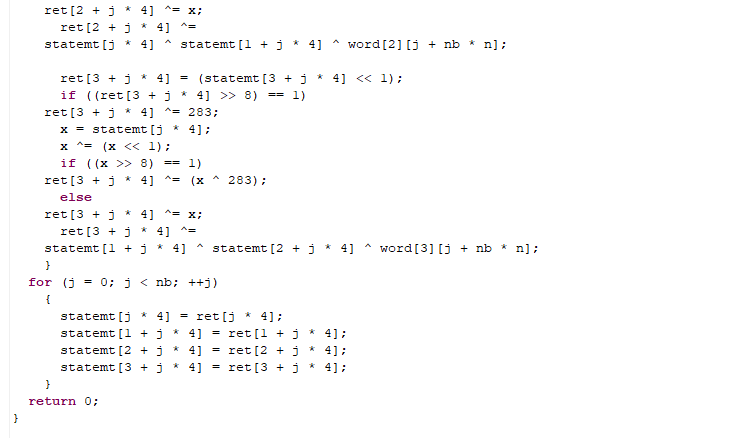


Δηλαδή με 16 κύκλους για να τελειώσει.

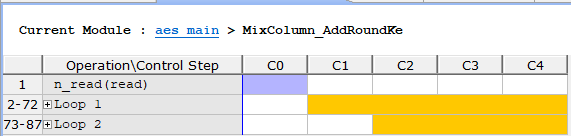
Εδώ βλέπουμε εκτός από το switch που εκτελείται μόνο στην περίπτωση στην οποία έχουμε στιγμιότυπο βλέπουμε και εντολές shift left logical καθώς και το λογικό AND.

Η MixColumn\_AddRoundKey γίνεται στις παρακάτω εντολές:





Όπου το τελευταίο loop κάνει την πρόσθεση του RoundKey.



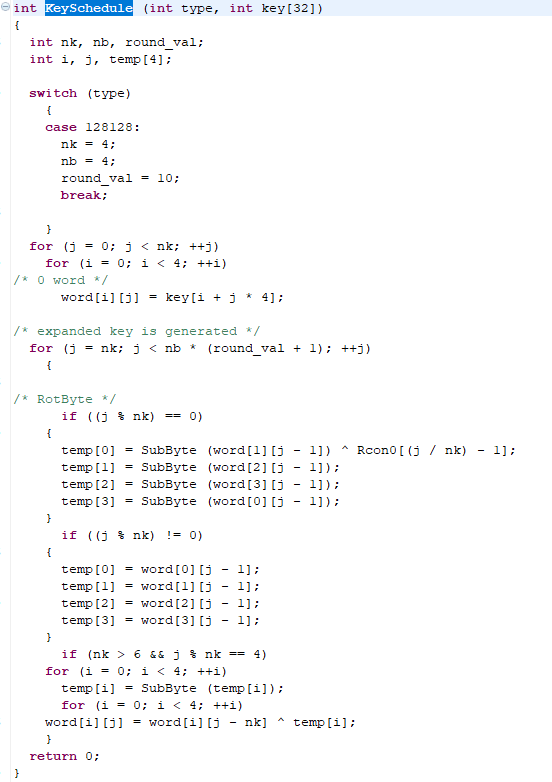
Και βλέπουμε ότι και αυτό κάνει 4 κύκλους για να τελειώσει. Να υπενθυμίσουμε ότι το 1ο loop αποτελεί το MixColumn.

Εδώ βλέπουμε όσον αφορά την συνάρτηση MixColumn ότι υπάρχει ένα loop, που εκτός από προσθέσεις, συγκρίσεις βλέπουμε και shift left logical που είναι μέσα σε μια συνθήκη, εντολές xor ,shift right logical καθώς και διάφορους πολλαπλασιασμούς.

Ενώ στην AddRoundKey

Εκτός από τις πράξεις που κάνει το loop , συναντάμε και γίνομενο.

Για τον υπολογισμού του RoundKey χρησιμοποιείτε η συνάρτηση keySchedule που είναι οι εξής:



Και κάνει συνολικά 8 κύκλους σύμφωνα με την εικόνα:



Από τι βλέπουμε χρησιμοποιεί μια case συνάρτηση που δεν χρειάζεται και στην συνέχεια 2 loop.

Σε κάθε loop κάνει προσθέσεις και συγκρίσεις.

Στο 1ο loop βλέπουμε ότι κάνει επιπλέον πρόσθεση και πολλαπλασιασμό.

Ενώ στο 2ο loop βλέπουμε να κάνει (nb \*(round\_val)+1) πολλαπλασιασμούς και επιπλέον έχουμε και 3 συνθήκες που θα καθορίσουν αν θα καλέσουμε την συνάρτηση subByte ή αν θα κάνουμε μια απλή καταχώρηση ή αν θα κάνουμε και τα δύο μαζί και xor όπως βλέπουμε στο τελευταίο loop.