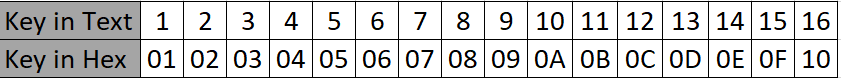
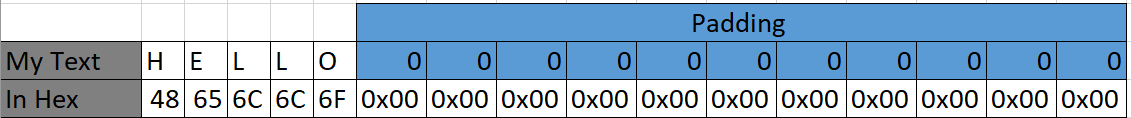
The Warden File System

# AES Example:

My Key:



My Text:



* The Padding is used for any text that is less then 16 bytes. We add 0x00 until 16 bytes are available.

## Prep for Rounds:

**Text Prep:**

Each of the rounds only work on 16 bytes of text at once. Therefor we must set our first 16 bytes as the state. Each round will change the state and after 10 rounds we add that state and the coming states to one ciphertext.

**Key Prep:**

For the 10 round we need 10 separate keys, the pre round will use the original key we supply the algorithm. For all the round we need 176 bytes, 16 bytes per key. The first 16 will be the original, follow the below steps for generating the rest of the 160 bytes.

* Store the last four bytes generated in a temporary field



The bytes above are for the first go through the expansion only.

* Every 16 bytes that are done (including the original) we need to preform the **core** of the expansion on the Temporary Bytes. This involves:
  + Left shift the Bytes by one.



* + For each of these bytes change them out for their sBox value.

Link <https://en.wikipedia.org/wiki/Rijndael_S-box>

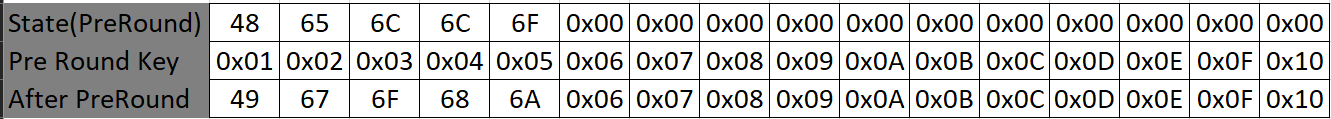


* + For the first byte of the temp field, we need to XOR its value with the corresponding value from the RCON look up table <https://cryptography.fandom.com/wiki/Rijndael_key_schedule>



* + Use these four bytes in the next step of the expansion.
* For each of the bytes we XOR them with the first 4 bytes of the previous 16 done
  + **AA** XOR with **01** = **AB**, and so on…
  + We save the results to the end of the previously done bytes. (first go through at this point we have 20 bytes done, 4 new and 16 original key)
  + I can show the Results from the code.

## Pre-Round:



The Pre Round is only doing the AddRoundKey algorithm, for this we take each byte and XOR that value with our Pre-Round key (Original Key)

## Rounds 1-9:

For each of the 9 rounds, we need to follow four steps:

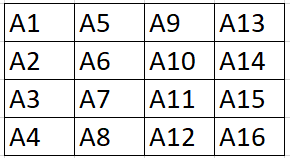
1. Sub Bytes
2. Shift Rows
3. Mix Columns.
4. AddRoundKey

### Sub Bytes

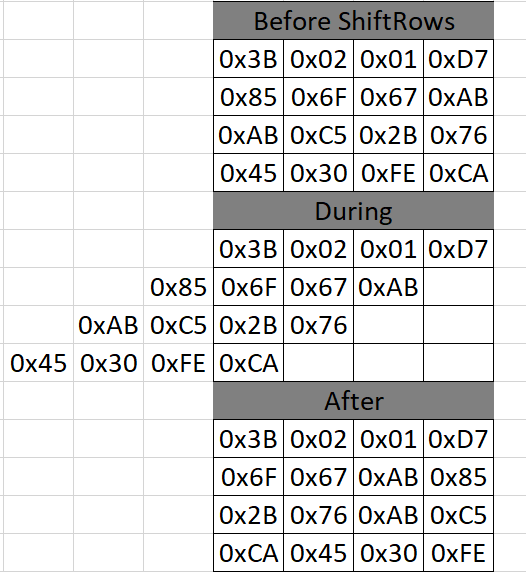
For the Sub Bytes step we exchange the value with its corresponding value in the sBox look up table for example, 0x49 will become 0x3B this will continue for each of the 16 bytes of the state.

### Shift Rows

For Shift rows it is easier to picture the 16 bytes of the state in a column matrix, in a column matrix the values are saved as such.



Here is an example using the bytes from the previous algorithm.

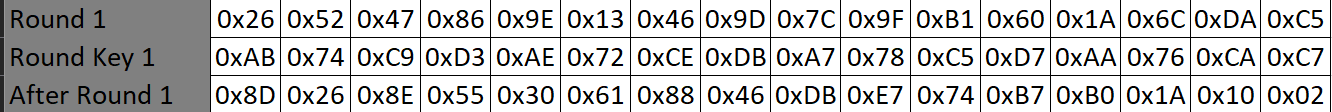


### Mix Columns

Please refer to the images below. It was easier to explain with writing.

The Images explain the steps involve but in the code, it is using 3 look up tables, mul1, mul2 and mul3 that are predetermined <https://en.wikipedia.org/wiki/Rijndael_MixColumns> and XORing the values from the results.

### AddRoundKey



This was explained before, but the one difference is the key, since this is round 1 we use bytes 16-30 from our Extended key. This changes each round.

## Final Round:

In the Final Round we do the following steps:

* Sub Bytes
* Shift Rows
* Add Round Key

For the Final round we use the last 16 bytes of the generated key. Each of these have been explained above.

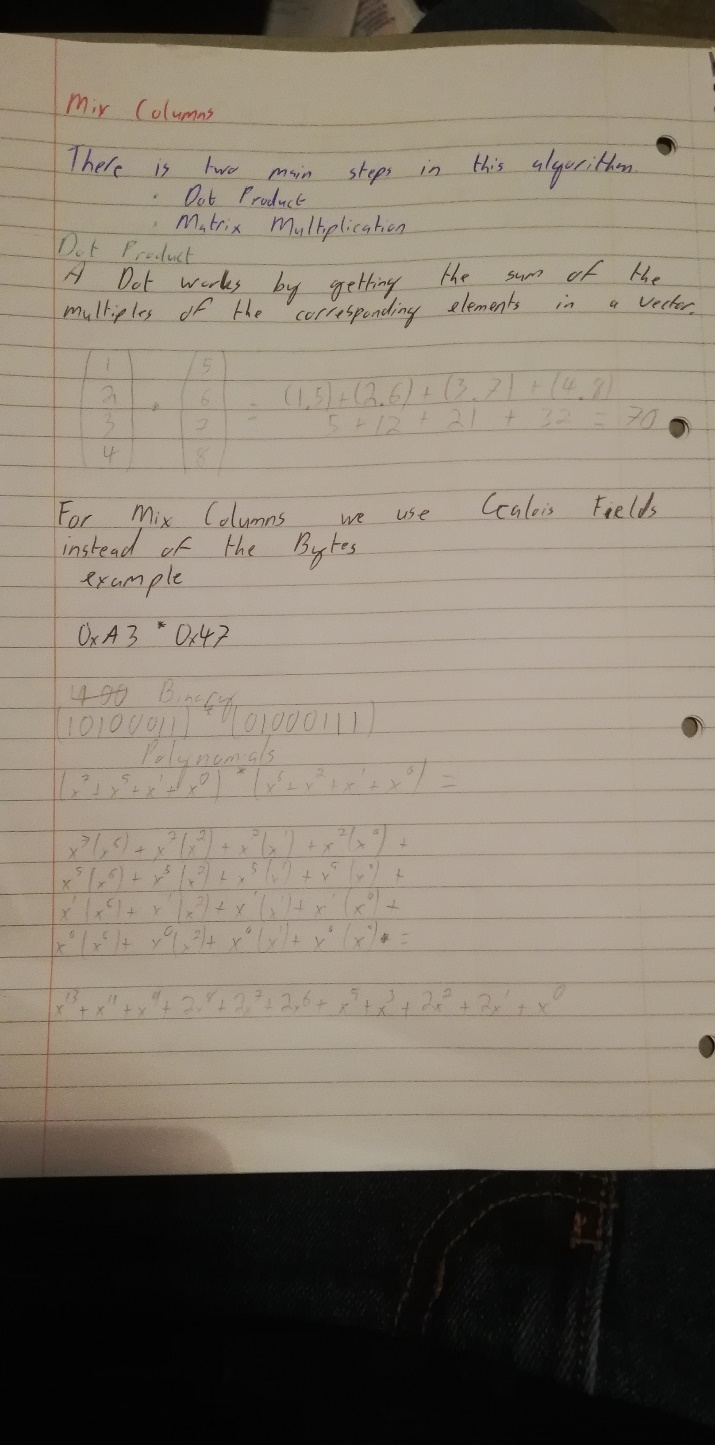


Figure 1: page 1 of Mix Columns.

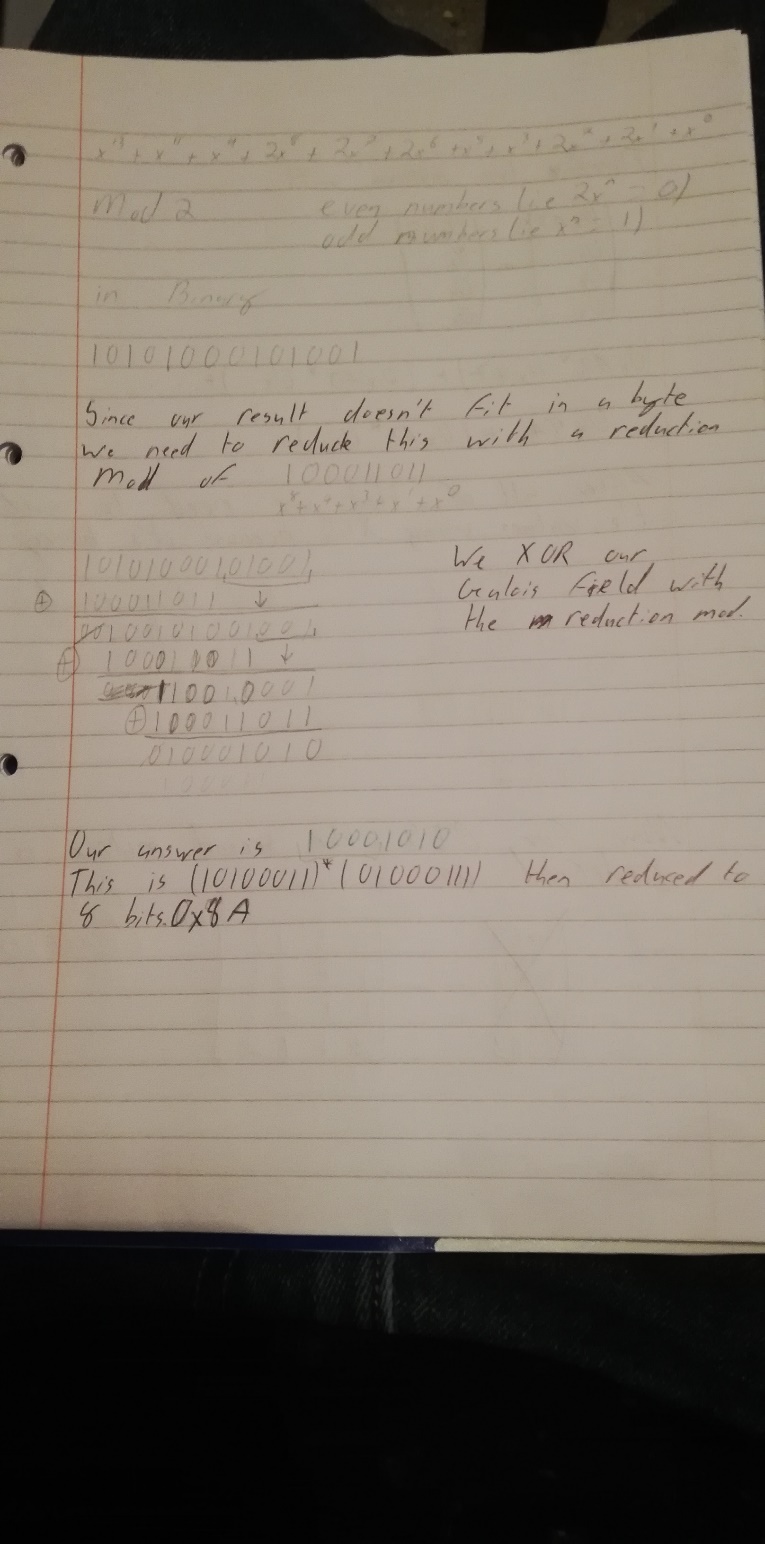


Figure 2: page 2 of Mix Columns.

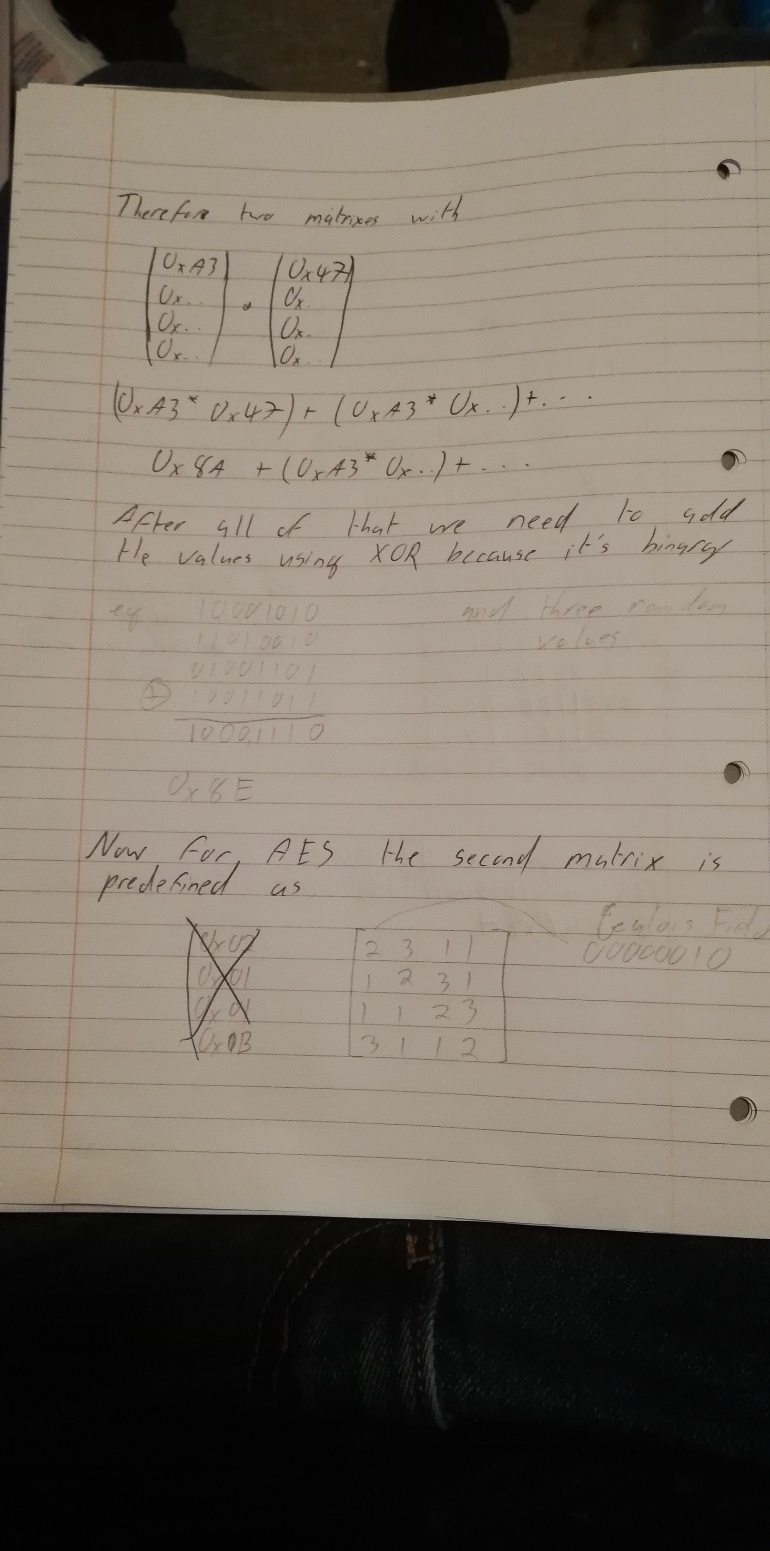


Figure 3: page 2 of Mix Columns.