Data Encryption using XOR

Date: 06/10/17

Submitted by:

MEHNAZ YUNUS-16C0124

RAMYA B - 16C0239

XOR Cipher, an encryption algorithm, is a cryptographic method for information protection. It works on the following principle:

$$A \wedge B = C$$

$$C \wedge B = A$$

(where A is the input message, B is the key and C is the encrypted data. XORing the encrypted data with the key gives back our message A).

Operations performed:

- Read input message
- Display current key
- Display encrypted message
- Display new key

Components Required:

- Multiplexers
- Shift registers using flip flops
- XOR gates
- Clock pulse

An 8-bit array taken as input (initial state) and the 8-bit array generated as output (initial state) are stored in a parallel-load-parallel-out registers. The final state (encrypted data) is obtained by XORing the initial state with a key which is not a constant value but generated using Rule 30 of cellular automaton (so the key changes for every encryption).

Rule 30 of Cellular Automaton:

This is used to generate a key for the next encryption. Consider a 1D array of cells $C_{i,}$ i=0,1,2, 3…7(since an 8-bit key is to be generated) where each cell's state is 0 or 1(i.e. C_i=0 or C_i=1). The initial key considered is C_i =0 for i=0,1,2,3,5,6,7 and C_i=1(for i=4). Hence the initial key is **00001000**.

At the end of encryption, we generate a new key dependent on the present key. Ci state (for the new key) is $\mathbf{C_{i-1} \wedge (C_i \text{ or } C_{i+1})}$ (where $C_{i-1,}$ Ci and C_{i+1} are states of the present key). This new key is fed back to the initial state register for the next encryption. A clock is used to control this feedback so that old key values are not corrupted by the new key before encryption.

References:

1. https://electronicsmail.wordpress.com/2012/10/14/data-encryption-and-decryption-system-using-74xx-logic-gates/
2. https://en.wikipedia.org/wiki/XOR_cipher
3. https://www.cs.rit.edu/~ark/winter2012/440/case01/casestudy.shtml