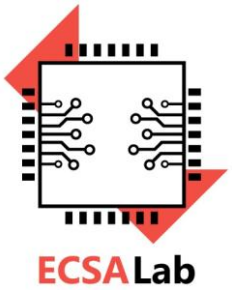
Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Πελοποννήσου

Εργαστήριο Ηλεκτρονικών Κυκλωμάτων, Συστημάτων και Εφαρμογών (ECSA Lab.)



Ασφάλεια Υπολογιστικών Συστημάτων

Διδάσκων: Δρ. Παρασκευάς Κίτσος

Άσκηση στη θεματική ενότητα του Σχεδιασμού Ασφαλούς Υλικού

1) Έστω ο αλγόριθμο RSA που εκτελεί εκθετοποίηση με τον αλγόριθμο square-and-multiply. Με βάση το παρακάτω ίχνος κατανάλωσης που λαμβάνεται κατά την αποκρυπτογράφηση: S M S S M S
Όπου S = square step (σταθερό pattern κατανάλωσης)
M = multiply step (χαρακτηριστική αιχμή)

(α) Ποιο είναι το bit-pattern του εκθέτη d;
(β) Υποθέστε ότι ο αλγόριθμος εκτελείται MSB→LSB. Ποιο είναι το μυστικό κλειδί πραγματικό d;

<u>Λύση</u>

Στον αλγόριθμο square-and-multiply (MSB→LSB) ξεκινάμε πάντα με MSB = 1 (init s = m). Για κάθε επόμενο bit i: εκτελείται πάντα S, και αν bit==1 τότε ακολουθεί M. Άρα κάθε **bit μετά το MSB** αντιστοιχεί σε μία ομάδα που ξεκινά με S και έχει προαιρετικό M.
Διαχωρίζοντας την ακολουθία σε ομάδες (κάθε ομάδα αρχίζει με S):

Ομάδα1: S M → bit_1 = **1**
Ομάδα2: S → bit_2 = **0**
Ομάδα3: S M → bit_3 = **1**
Ομάδα4: S → bit_4 = **0**
Άρα τα bits *μετά* το MSB είναι: 1 0 1 0. Αν ο MSB = 1 (προϋπόθεση του αλγορίθμου), τότε ολόκληρος ο εκθέτης d είναι:
MSB ακολουθούμενο από τη σειρά [1 0 1 0] → 1 1 0 1 0_2 = **11010_2**.
(Σημείωση: η παραδοχή MSB=1 είναι τυπική για την αρχικοποίηση του αλγορίθμου "square-and-multiply")

2) Έστω ο αλγόριθμο RSA που εκτελεί εκθετοποίηση με τον αλγόριθμο square-and-multiply. Για το κλειδί d = 0xF0 = $(11110000)_2$ να βρείτε τα παρακάτω.
α) Ποιες πράξεις (S=Square, M=Multiply) θα εκτελεστούν;

β) Πόσες πράξεις Square και πόσες Multiply θα παρατηρηθούν;

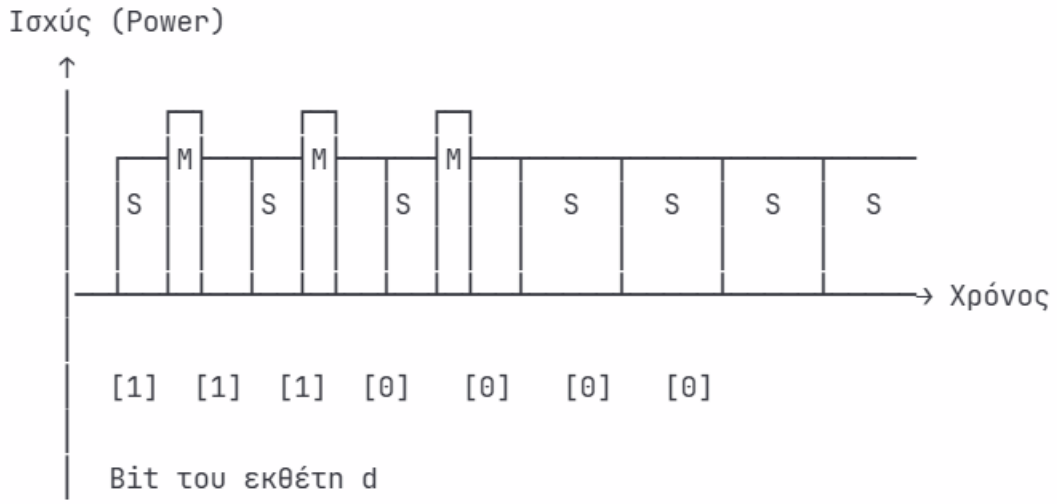
Και επίσης να σχεδιάστε το αναμενόμενο μοτίβο κατανάλωσης

<u>Λύση</u>
Αρχικοποίηση: s = m (από το πρώτο bit που είναι 1)
Bits του κλειδιού: 1-1-1-1-0-0-0-0
α)

Βήμα	Bit	Πράξη	Περιγραφή	Αποτέλεσμα
Init	1 (MSB)	-	Αρχικοποίηση s = m	$s = m^1$
1	1	S + M	$s = s^2 \times m$	$s = m^3$
2	1	S + M	$s = s^2 \times m$	$s = m^7$
3	1	S + M	$s = s^2 \times m$	$s = m^{15}$
4	0	S	$s = s^2$	$s = m^{30}$
5	0	S	$s = s^2$	$s = m^{60}$
6	0	S	$s = s^2$	$s = m^{120}$
7	0	S	$s = s^2$	$s = m^{240}$

β) Πράξεις: 7 SQUARE + 3 MULTIPLY
γ) Χαρακτηριστικά Κατανάλωσης:

1. SQUARE (S): Μεσαία κατανάλωση ισχύος, μικρότερη διάρκεια
2. MULTIPLY (M): Υψηλότερη κατανάλωση ισχύος, μεγαλύτερη διάρκεια



3) Να σχεδιάσετε ένα συνδυαστικό (Combinational) mini-trojan (hardware) με τρεις εισόδους που να:

- Ενεργοποιείται όταν A=1 και B=0 και C=1
- Αντικαθιστά το bit Y με το notY

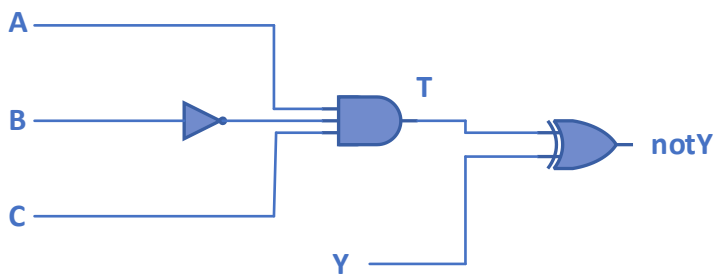
Ζητείται:
(α) Σχεδιάγραμμα λογικών πυλών
(β) Trigger & payload
(γ) Είναι εύκολα εντοπίσιμο;

<u>Λύση</u>

Ενεργοποίηση (trigger) όταν A=1, B=0, C=1 και payload = Y ← notY (αντικαθιστά το bit Y με το αντίθετό του).

Σχεδίαση λογικής

- Payload (εφαρμογή στο Y): Y_out = Y ⊕ T (XOR μεταξύ Y και trigger).
- Trigger logic: T = A & (notB) & C.
    - Χρειάζεται μία NOT για την είσοδο B, και μια AND τριών εισόδων (ή σειρά από 2-input ANDs).



Ανιχνευσιμότητα / σχολιασμός
Μικρό αποτύπωμα: μόνο λίγες πύλες — stealthy.
Εύκολη ανίχνευση με functional test αν ο εισαγωγικός συνδυασμός A=1,B=0,C=1 δοκιμαστεί και ελέγξεις Y.
Πιθανή αποφυγή αν το μοτίβο trigger είναι σπάνιο (σπάνιο συνδυαστικό vector)