CS771A: Assignment 1

Abstract

This pdf file contains submissions for the Question 1, Question 2 and Question 4

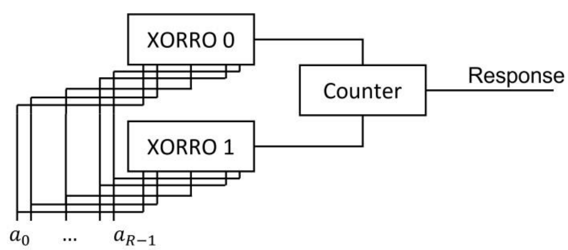
parts of the assignment.

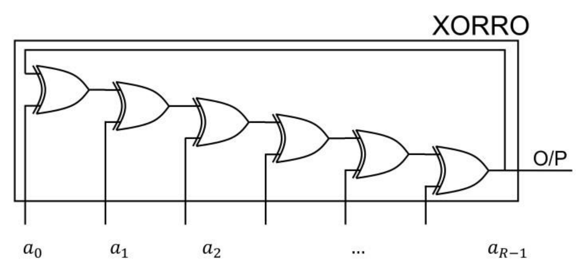
1 Question 1

Given: Simple XORRO PUF has two XORROs and has no select bits and no multiplexers. A single XORRO PUF has R XORs. The challenge is an R-bit string that is fed into both XORROs as their configuration bits:

$$a \overset{\text{def}}{=} [a_0, a_1, .. a_{R-1}]$$

Now, we will find the time delay for i^{th} XOR.





Here a_i denotes the config bit for the i^{th} XOR. Let $\delta^i_{00}, \delta^i_{01}, \delta^i_{10}, \delta^i_{11}$ be the time that i^{th} XOR gate takes before giving its output. When the input to that gate is 00,01,10 and 11, respectively. Let D_i be the time delay for the i^{th} XOR.

When $\mu_i = 0$,

$\Delta_i^{\mu_i=0} = a_i\delta^i_{01} + (1 - a_i)\delta^i_{00}$ —–(1)

When $\mu_i = 1$,

$\Delta_i^{\mu_i=1} = a_i\delta^i_{11} + (1 - a_i)\delta^i_{10}$ —–(2)

From (1) and (2), we will get the time delay of i^{th} XOR.

$\Delta_i = \mu_i\Delta_i^{\mu_i=1} + (1 - \mu_i)\Delta_i^{\mu_i=0}$

Preprint. Under review.

$\Delta_i = \mu_i(a_i\delta^i_{11} + (1 - a_i)\delta^i_{10}) + (1 - \mu_i)(a_i\delta^i_{01} + (1 - a_i)\delta^i_{00})$

Time Delays:

For 0^{th} XOR,

When $\mu_i = 0$,
$\Delta_0^{\mu_0=0} = a_0\delta_{01}^0 + (1-a_0)\delta_{00}^0$ —–(3)

When $\mu_0 = 1$,
$\Delta_0^{\mu_0=1} = a_0\delta_{11}^0 + (1-a_0)\delta_{10}^0$ —–(4)

Adding (3) and (4), we get,
$\Delta_0^{\mu_0=0} + \Delta_0^{\mu_0=1} = a_0(\delta_{01}^0 + \delta_{11}^0 - \delta_{00}^0 - \delta_{10}^0) + (\delta_{00}^0 + \delta_{10}^0)$

For 1^{st} XOR, When $\mu_0^r = 0$, We know that when one of the two inputs to XOR gate is 0, it acts as an identity i.e, it will give the 2^{nd} input as output.

$\therefore \mu_1 = a_0$
$\Delta_1^{\mu_0=0} = \mu_1(a_1\delta_{11}^1 + (1-a_1)\delta_{10}^1) + (1-\mu_1)(a_1\delta_{01}^1 + (1-a_1)\delta_{00}^1)$ —–(5)

When $\mu_0 = 1$, We know that when one of the inputs to XOR gate is 1, it acts as an inverter i.e, it will give the negation of the 2^{nd} input as output.
$\therefore \mu_1 = 1 - a_0$
$\Delta_1^{\mu_0=1} = (1-a_0)(a_1\delta_{11}^1 + (1-a_1)\delta_{10}^1) + a_0(a_1\delta_{01}^1 + (1-a_1)\delta_{00}^1)$ —–(6)

Adding (5) and (6), we get,
$\Delta_1^{\mu_0=0} + \Delta_1^{\mu_0=1} = a_1(\delta_{01}^1 + \delta_{11}^1 - \delta_{00}^1 - \delta_{10}^1) + \delta_{00}^1 + \delta_{10}^1$

Similarly, For the 3^{rd} XOR,

When $\mu_0 = 0$, $\mu_2 = (1 - a_0)a_1 +$

$a_0(1 - a_1)$

$\Delta_2^{\mu_0=0} = \mu_2(a_2\delta_{11}^2 + (1-a_2)\delta_{10}^2) + (1-\mu_2)(a_2\delta_{01}^2 + (1-a_2)\delta_{00}^2)$ —–(7)

When $\mu_0 = 1$, $\mu_2 = [1 - [(1 - a_0)a_1 +$

$a_0(1 - a_1)]]$

$\Delta^{\mu}{}_{2}{}^{0=1} = [1-[(1-a_0)a_1 + a_0(1-a_1)]](a_2\delta_{11}{}^2 + (1-a_2)\delta_{10}{}^2) + [(1-a_0)a_1 + a_0(1-a_1)](a_2\delta_{01}{}^2 +$
$(1-a_2)\delta_{00}^2)$ —–(8)

Adding (7) and (8), we get,
$\Delta_2^{\mu_0=0} + \Delta_2^{\mu_0=1} = a_2(\delta_{01}^2 + \delta_{11}^2 - \delta_{00}^2 - \delta_{10}^2) + \delta_{00}^2 + \delta_{10}^2$

Now, From the pattern, we can write,
$\Delta_i^{\mu_0=0} + \Delta_i^{\mu_0=1} = a_i(\delta_{01}^i + \delta_{11}^i - \delta_{00}^i - \delta_{10}^i) + \delta_{00}^i + \delta_{10}^i$
Now, $\Sigma_{i=0}^{1=R-1}(\Delta_2^{\mu_0=0} + \Delta_2^{\mu_0=1}) = \Sigma a_i(\delta_{01}^i + \delta_{11}^i - \delta_{00}^i - \delta_{10}^i) + \Sigma(\delta_{00}^i + \delta_{10}^i)$ We know that,

$\Sigma_{i=0}^{1=R-1}\Delta_i^{\mu_0=0} = t_0$ and $\Sigma_{i=0}^{1=R-1}\Delta_i^{\mu_0=1} = t_1$
$\therefore t_0 + t_1 = \Sigma a_i(\delta_{01}^i + \delta_{11}^i - \delta_{00}^i - \delta_{10}^i) + \Sigma(\delta_{00}^i + \delta_{10}^i)$

For Upper XORRO, or XORRO2,
$(t_0 + t_1)^{upper} = [\Sigma a_i(\delta^i_{01} + \delta^i_{11} - \delta^i_{00} - \delta^i_{10}) + \Sigma(\delta^i_{00} + \delta^i_{10})]^{upper}$ ––(9)

For Lower XORRO, or XORRO1,
$(t_0 + t_1)^{lower} = [\Sigma a_i(\delta^i_{01} + \delta^i_{11} - \delta^i_{00} - \delta^i_{10}) + \Sigma(\delta^i_{00} + \delta^i_{10})]^{lower}$ ––(10)

Subtracting (10) from (9), we get,

$(t_0 + t_1)_{upper} - (t_0 + t_1)_{lower} = [\Sigma a_i(\delta_{01i} + \delta_{11i} - \delta_{00i} - \delta_{10i}) + \Sigma(\delta_{00i} + \delta_{10i})]_{upper} - [\Sigma a_i(\delta_{01i} + \delta_{11i} - \delta_{00i} - \delta_{10i}) + \Sigma(\delta_{00i} + \delta_{10i})]_{lower}$

$\Longrightarrow (t_0+t_1)_{upper} - (t_0+t_1)_{lower} = [\Sigma a_i[(\delta_{01i} + \delta_{11i} - \delta_{00i} - \delta_{10i})_{upper} - (\delta_{01i} + \delta_{11i} - \delta_{00i} - \delta_{10i})_{lower}] + \Sigma[(\delta_{00i} + \delta_{10i})_{upper} - (\delta_{00i} + \delta_{10i})_{lower}]$

Given: If upper XORRO has higher frequency, then the output in 1.

$\Longrightarrow (t_0 + t_1)_{upper} - (t_0 + t_1)_{lower} < 0$ Response = 1 and,
$\Longrightarrow (t_0 + t_1)_{upper} - (t_0 + t_1)_{lower} > 0$ Response = 0

$\Longrightarrow$ output = $\frac{1+sign((t_0+t_1)^{lower}-(t_0+t_1)^{upper})}{2}$

$\Longrightarrow$ output = $\frac{1+sign([\Sigma[(\delta^i_{01}+\delta^i_{11}-\delta^i_{00}-\delta^i_{10})^{lower}-(\delta^i_{01}+\delta^i_{11}-\delta^i_{00}-\delta^i_{10})^{upper}]a_i+\Sigma[(\delta^i_{00}+\delta^i_{10})^{lower}-(\delta^i_{00}+\delta^i_{10})^{upper}])}{2}$

Comparing the above equation with, $\Longrightarrow$ output = $\frac{1+sign(w^T\phi(c)+b}{2}$

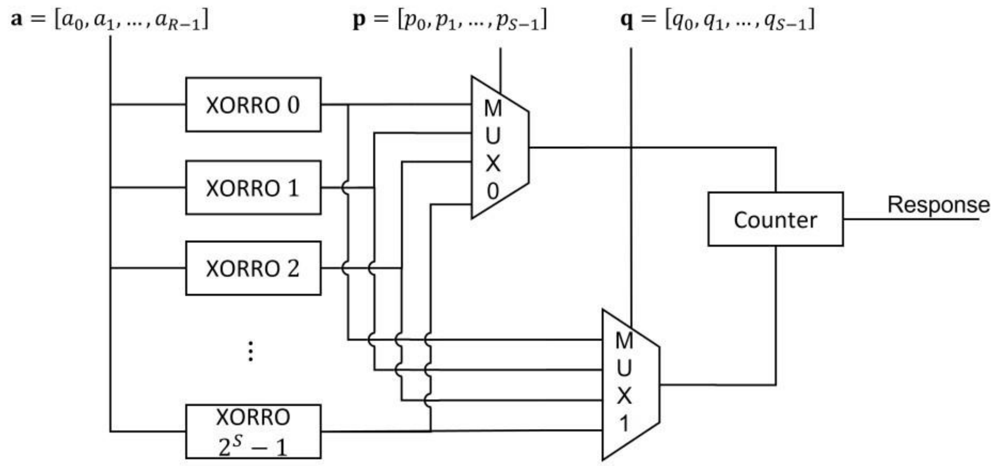
We see that, $w_i = [(\delta_{01i} + \delta_{11i} - \delta_{00i} - \delta_{10i})_{lower} - (\delta_{01i} + \delta_{11i} - \delta_{00i} - \delta_{10i})_{upper}]$ $b = \Sigma[(\delta_{00i} + \delta_{10i})_{lower} - (\delta_{00i} + \delta_{10i})_{upper}]$

$X = a$

Given the map $\phi : [0,1]^R \to R^D \Longrightarrow D = R$ and $\phi(c) = c$.

2 Question 2

Given: Advanced XORRO contains 2^s XORROs and 2 Multiplexers.
Input for MUX0: P = $[p_0,p_1,...,p_{s-1}]$
Input for MUX1: Q = $[q_0,q_1,...,q_{s-1}]$



To find the XORRO chosen by MUX0 and MUX1, we convert the given bits p and q into their corresponding decimal number, say P and Q.

The upper XORRO: $P = \Sigma_{i=0}^{s-1} 2^i . p_{s-(i+1)}$

The lower XORRO: $Q = \Sigma_{i=0}^{s-1} 2^i . q_{s-(i+1)}$

The total number of unique pairs (P, Q) possible is N = $2^s(2^s - 1)$

For a given (P_0,Q_0), the response for (Q_0,P_0) will be the negation of the response for (P_0,Q_0). Hence, we can reduce the number of linear models to M = $2^{s-1}(2^s - 1)$ by switching the response of (Q_0,P_0) while training the model of (P_0,Q_0).

To solve the problem of the advanced XORRO PUF, we will train M models now. Each model will be trained by dividing the data set into M parts in which all the CRPs having (P_0,Q_0) and (Q_0,P_0) as their Pand Q will be grouped together. We can reduce this problem to M simple XORRO PUF models as follows:

For a given P, Q: $w_i = [(\delta_{01i} + \delta_{11i} - \delta_{00i} - \delta_{10i})_{Q_{th}XORRO} - (\delta_{01i} + \delta_{11i} - \delta_{00i} - \delta_{10i}$

$)_{P_{th}XORRO}]$ $b = \Sigma[(\delta_{00i} + \delta_{10i})_{Q_{th}XORRO} - (\delta_{00i} + \delta_{10i})_{P_{th}XORRO}]$

$X = a$

where $P,Q \in [0,2^{s-1}]$ and $P /= Q$

3 Question 3

The python file for question 3 is included in the folder as submit.py.

4 Question 4

4.1 Question 4a

LinearSVC:

loss	Accuracy (%)	Time (secs)
Square hinge	94.7375	4.735

Hinge	93.9725	4.222

4.2 Question 4b

LinearSVC:

C	Accuracy (%)	Time (secs)
0.01	90.3975	3.3247
1	94.74	4.7953
100	93.935	4.3577

Logistic regression:

C	Accuracy (%)	Time (secs)
0.01	82.5175	4.089
1	93.9175	4.7354
100	94.9425	6.85

4.3 Question 4c

LinearSVC:

tol	Accuracy (%)	Time (secs)
$1e-8$	93.9175	4.5361
$1e-4$	93.9175	4.728
$1e^0$	93.78	3.9921

Logistic regression:

tol	Accuracy (%)	Time (secs)
$1e-8$	94.725	4.855
$1e-4$	94.74	4.7947
$1e^0$	94.5425	3.391

4.4 Question 4d

LinearSVC:

penalty	Accuracy (%)	Time (secs)
l1	94.7325	4.7435
l2	94.7425	4.788

Logistic regression:

penalty	Accuracy (%)	Time (secs)
l1	93.9175	4.565
l2	93.9175	4.719