EE577A Fall 2019 – VLSI System Design

Instructor: Prof. Massoud Pedram TA: Hongxiang Gao

Final Project Part II:

RSA Cryptography System

Notes:

- There will be a demo session of this project, you need to show your work and simulation result in the discussion session on 12/13.
- The due date of final report of this project is 12/13 11:59 a.m., late submission won't be graded.
- This project is group work, 3 students at most in a team.
- The cells you designed in this lab can be used in your final project, but not required.
- Submit your final report on the Blackboard website.
- Ask questions only on blackboard discussion forum or in the office hours. DO NOT send email asking technical questions.
- Start as soon as possible.

Introduction

Design an RSA Cryptography System. Figure 1 shows the general block diagram of the design. A plain text ***m*** is encrypted into a cipher text ***c*** with public keys ***n, e***. A cipher text ***c*** is decrypted into a plain text ***m*** with private keys ***n, d***. In this project, you need to implement the decryption part, that is to calculate the plain text ***m*** based on cipher text ***c***, private keys ***n*** and ***d***.

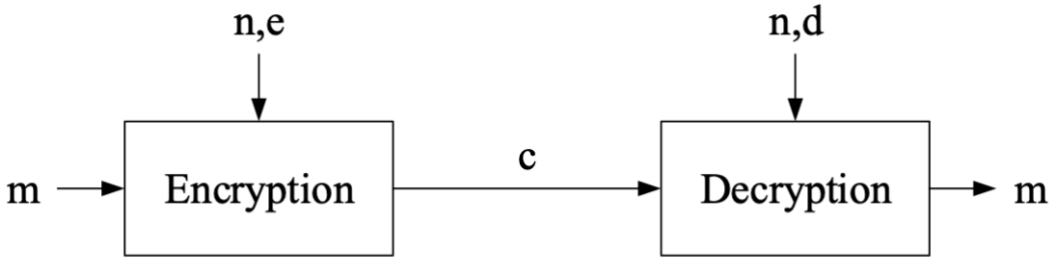

Figure 1: General block diagram of an RSA Cryptography System

The private key ***n*** is defined as the product of two large primes ***p*** and ***q***. The way to calculate ***m*** is $m = c^d \pmod{n} = c^d \pmod{p*q}$. Based on Chinese Remainder Theorem (CRT), two results M_p and M_q are introduced. The intermediate result $M_p = c_p^{d_p} \pmod{p}$ and similar for M_q. Here $c_p = c \pmod{p}$, $d_p = (d \bmod p)$. The value of M_p can be calculated by combination of a series of Montgomery unit designed in part I, which is called Montgomery Modular Exponentiation. Its

schematic is shown in Figure 2. The value of $R^2 \pmod p = r_p$ is generated by the Python code.

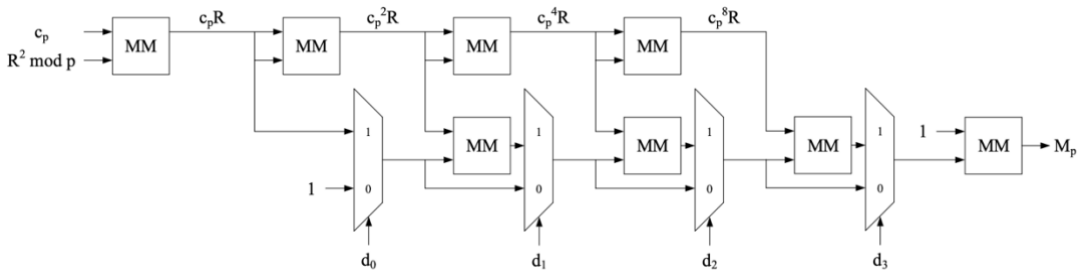

Figure 2: Montgomery Modular Exponentiation Schematic

The complete schematic is shown in Figure 3. The output needs to be loaded to a flip-flop. Here the intermediate value q^{-1} is generated by Python code.

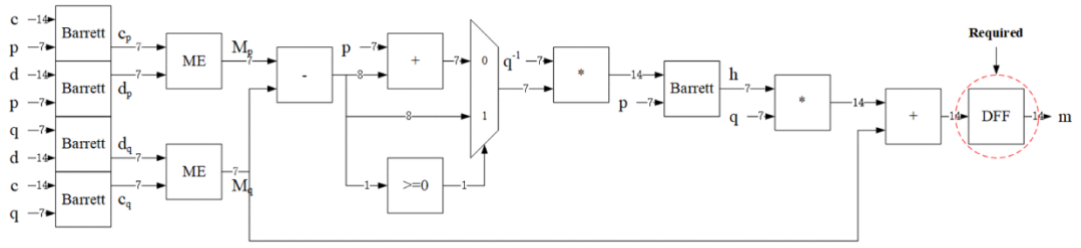

Figure 3: Complete Schematic

General Instructions

a) Use $\beta = \mu_n/\mu_p = 4$ for all gates in your design.
b) You may use all the unit you designed.
c) Use $V_{DD} = 1.8V$.
d) All inputs and outputs of the modulo unit need to be registered with positive-edge triggered master-slave flip-flops.
e) Use rise/fall time = 10 ps.
f) You are not required to measure the worst cases delay, but the clock you set of FFs should be large enough to avoid the setup-time violation.
g) You can insert DFFs between stages to build a pipelined circuit, therefore, a better clock can be achieved this way. However, it will lead to a larger area in layout. You need to make a trade-off in this issue.

Layout Guidelines

a) The height of all your designs of one level should be 50-60 lambdas which is 5-6 µm in NCSU_TechLib_tsmc02 technology. The height of one level is measure from the middle of power rail to the middle of ground rail.
b) VDD and GND should be routed in metal 1 at top and bottom of the cell. The metal width of

VDD and GND should be 10 lambdas (1 µm).

c) For the internal routing you can use poly (usually for short routing) or metal layer. (you may use any metal layer up to **metal 6**.) We recommend you use metal 1 for horizontal and metal 2 for vertical connections or vice versa. Try to use as few metal layers as possible therefore routing would be more convenient for your future assignments which may use this part.

d) You will have to try building your layout looks like a square shape. All the input signals should be given from the sides and all the output signal should be connected at the right-hand side.

Functionality Test

You will have to perform the functionality test using the input patterns generated by Perl or Python:

a) Design the Perl/Python script which can randomly generates input pattern and its corresponding result in both decimal and binary format.

b) For the intermediate value of p^{-1}, q^{-1}, r_p and r_q, you can directly calculate it in your Python/Perl program and convert it as a part of the input pattern.

c) Fill in the table given below after running your script. All the input patterns are given. Here the time is defined as the time when all outputs are generated as in Figure 4.

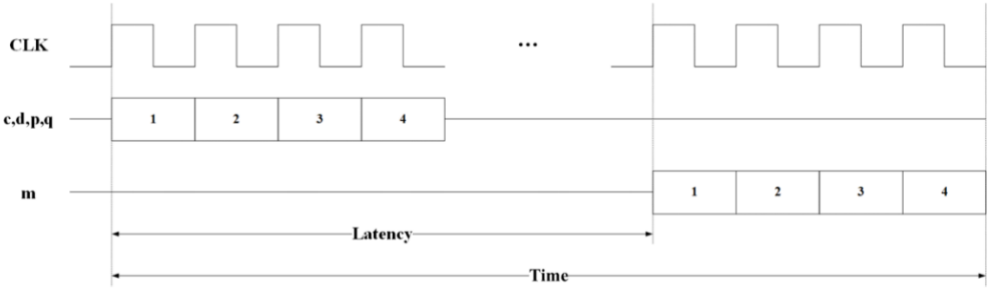

Figure 4: Time Latency Graph

Function								
Inputs								Outputs
Given				Python				Hardware
c	d	p	q	r_p	p^{-1}	r_q	q^{-1}	m
831	2971	127	113					
4624	9833	107	97					
4058	2831	113	109					
6757	6593	103	89					
Metric								
Time (ns)			Area (mm^2)			Area*Time		

d) Apply the patterns in Cadence and compare the waveforms with the result generated above. You can either use vector file or directly set the input patterns in Cadence while vector file is preferred since in future lab assignments the input patterns will be much more complicated.

Report Checklist

Your final report will contain two parts, the Montgomery Unit and the RSA Encryption System. The Montgomery Unit will take 20% of the final report.

For Montgomery Unit (30 points):

a) Schematics of the circuit (6 points)
b) Explanation of the working principle of the circuit (6 points)
c) Functionality test of schematic including Python code (9 points)
d) Layout and LVS match report (3 points)
e) Functionality test of layout (3 points)
f) A summary of key parameters of your design, including the minimum clock, height, width and area of your layout (3 points)

For RSA Encryption System (70 points):

a) Schematics of the Montgomery Exponential Unit (5 points)
b) Schematic of complete circuit (20 points)
c) Explanation of the working principle of the circuit (10 points)
d) Functionality test of schematic including Python code (20 points)
e) Layout and LVS match report (5 points)
f) Functionality test of layout (5 points)
g) A summary of key parameters of your design, including the minimum clock, height, width and area of your layout (5 points)
h) You need to come to the demo session to demonstrate your design, otherwise you will lose some points of your final project.

Up to three teams could get a bonus based on their performance (Area * Time) of circuit.

Submission Guidelines

a) Briefly explain your work.
b) Name your report file in following format:
"firstname_lastname_studentID_Final_EE577A_Fall19.pdf", for example:
"Hongxiang_Gao_1111111111_Final_EE577A_Fall19.pdf"