בס"ד

הפקולטה להנדסה ומדעי המחשב
החוג להנדסת חשמל ואלקטרוניקה

הצעת פרויקט גמר באלקטרוניקה

מימוש בחומרה של אלגוריתם ההצפנה Simon

Hardware Implementation of the Simon Encryption Algorithm

מגיש: ליאור ברנר
מנחה: מר אורי שטרו

תשרי תשפ״ד
ירושלים

בס"ד

Table of Contents

1. Background and Motivation

The Simon encryption algorithm was developed in 2013 by the US National Security Agency (NSA) with the aim of developing a lightweight, flexible and easy-to-analyze cipher that offers excellent throughput, performance and low energy costs. The algorithm was designed specifically for use within the Internet of Things (IoT). Simon was designed to optimize performance in hardware implementations. Along with Simon, The NSA developed the Speck algorithm, which is designed for optimal performance in software implementations. Both ciphers were accepted by the International Organization for Standardization (ISO) in 2018 as part of the RFID air interface standard.

Data security is one of the most important aspects of our time. Efficient and secure data transfer is critical in many cases. In order to transfer data securely, many different cryptographic algorithms are used. Included among these is the Advanced Encryption Standard (AES) which is currently the standard encryption algorithm for many organizations including the US government. The drawback to these algorithms is the fact that they require a lot of time, energy and memory to operate effectively. When dealing with more powerful machines such as computers this is fine but when we want to encrypt data in machines with lesser technological capabilities these algorithms become inefficient.

The IoT is a system of devices with communication technology which allows them to connect and communicate with each other. These devices can range from smartphones to laptops to lighting systems and more. Many of these devices do not have the ability to encrypt data with AES or similar algorithms as the technology that they use to communicate has limited abilities. Therefore, it is important to develop an encryption algorithm that requires very little power and memory in order to operate efficiently. For this the NSA created the Simon and Speck algorithms.

An FPGA is an integrated circuit that can be reprogrammed to have its hardware changed after it is created. Its hardware is designed using a Hardware Description Language (HDL), most commonly VHDL or Verilog. One of the great advantages of FPGA is that many operations can be done in parallel thus making it more time efficient. Since the hardware of the FPGA can be changed, it is good at performing various and unrelated tasks efficiently as per the user's need.

2. Project Objective

Within the framework of the project, the Simon encryption algorithm will be implemented in hardware. With the final product it will be possible to encrypt and decrypt data using the Simon encryption algorithm.

The algorithm will be implemented on the Basys 3 board which is manufactured by Digilent Inc. and uses FPGA technology developed by Xilinx-AMD. The Basys 3 will be connected to a Raspberry Pi. The purpose of the Raspberry Pi is to format the data that the user inputs into the system - including the plaintext or ciphertext that will be encrypted or decrypted, the key, the number of bits that will be encrypted each time and whether the data is being encrypted or decrypted - in a way that it can be transferred to the Basys 3. In addition, it will serve as a controller for the Basys 3 and make sure that everything works as intended and that all the data enters at the correct time. The Basys 3 will encrypt the data and transfer it back to the Raspberry Pi which will display the data on the screen.

3. Block Diagram of the System

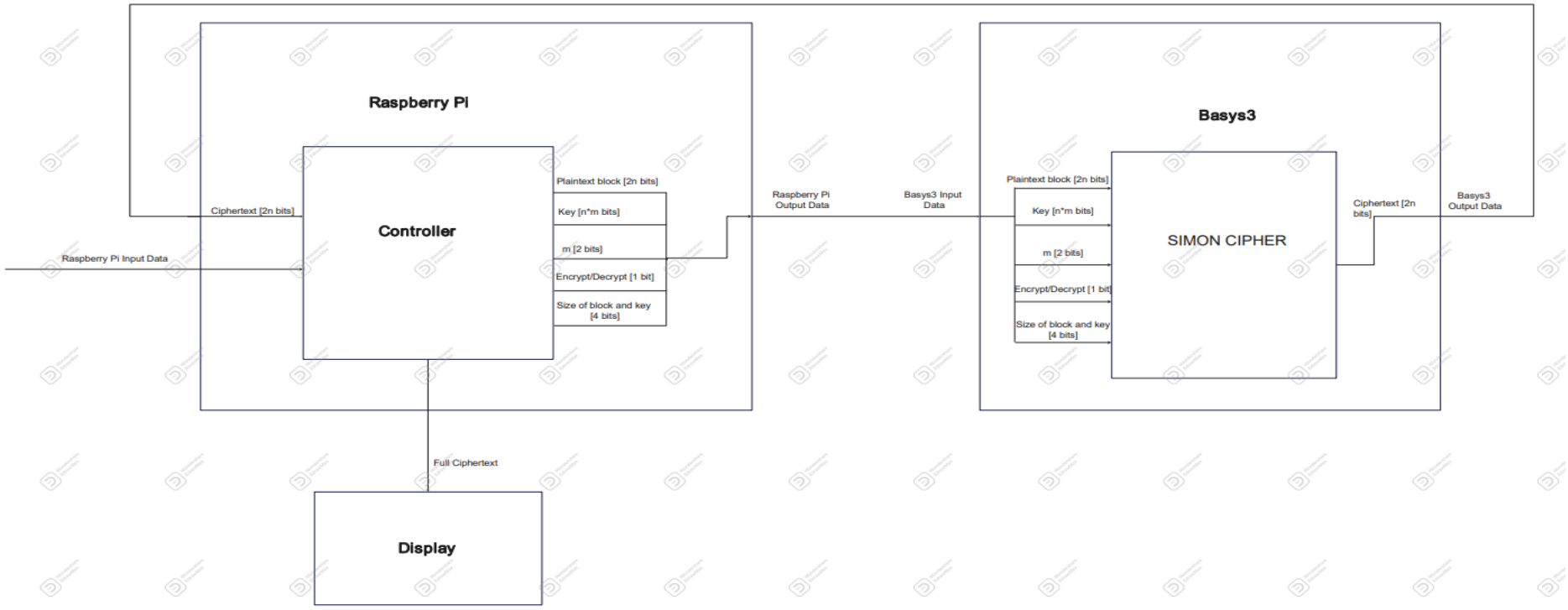

Figure 1 - Block Diagram

The Raspberry Pi receives the initial data and sends it to the the Basys 3. This data the plaintext, the key, their ratio (defined as m), encryption/decryption and the number of bits of the touch and the key (there are 10 possible combinations). The final encrypted data is sent from the Basys 3 back to the Raspberry Pi and is outputted on screen. Note: The diagram uses “Plaintext block” to refer to both plaintext or ciphertext depending on if the system is encrypting or decrypting the data.

4. Work Schedule for the Year

Notes:

Primary tasks:

1) Learning about the algorithm, the Basys 3 and the Raspberry Pi

2) Designing the inputs and outputs of all the components in the system and the connection between them

3) Writing the algorithms in Verilog and implementing them on the Basys 3

4) Writing the final report and preparing all the required presentations

5) Studying the course in cryptography taught by Mr. Stroh

6) Participation in the Xilinx Open Hardware Design Competition in the field of FPGA technology implementation

The diagram includes breaks for holidays throughout the year, the Duration column indicates actual working days (for example 7 working days is 9 days in total because working days are Sunday through Thursday inclusive), the arrows mark tasks that depend on each other.

ID	Task name	Start	Finish	Duration	Complete
1	**Learning**	2023-10-08	2023-10-26	15.0 d.	0.0%
2	Learning the algorithm	2023-10-08	2023-10-26	15.0 d.	0.0%
3	Learning the Basys3 board	2023-10-08	2023-10-26	15.0 d.	0.0%
4	Learning the Raspberry Pi	2023-10-08	2023-10-26	15.0 d.	0.0%
5	**Wrapper Design**	2023-10-29	2024-01-11	55.0 d.	0.0%
6	Designing the RPi front end	2023-10-29	2023-11-13	12.0 d.	0.0%
7	**Designing the FPGA wrapper**	2023-11-13	2024-01-11	43.0 d.	0.0%
8	Design FPGA I/O	2023-11-13	2023-11-29	12.0 d.	0.0%
9	Simulation of FPGA	2023-11-29	2023-12-15	12.0 d.	0.0%
10	Synthesis of FPGA	2023-12-17	2024-01-02	12.0 d.	0.0%
11	P&R	2024-01-02	2024-01-11	7.0 d.	0.0%
12	**Writing algorithm in Verilog**	2024-01-14	2024-03-12	43.0 d.	0.0%
13	Writing algorithm in Verilog	2024-01-14	2024-01-30	12.0 d.	0.0%
14	Simulation of Verilog code	2024-01-30	2024-02-15	12.0 d.	0.0%
15	Synthesis of Verilog code	2024-02-16	2024-03-04	12.0 d.	0.0%
16	P&R	2024-03-05	2024-03-13	7.0 d.	0.0%
17	**Final Project Submisions**	2024-03-13	2024-06-23	72.0 d.	0.0%
18	**Writing the paper**	2024-03-13	2024-05-31	57.0 d.	0.0%
19	Studying more about cryptography	2024-03-14	2024-03-22	7.0 d.	0.0%
20	Writing the paper	2024-03-25	2024-05-31	40.0 d.	0.0%
21	Creating PowerPoint	2024-06-02	2024-06-10	7.0 d.	0.0%
22	Creating Poster	2024-06-13	2024-06-23	7.0 d.	0.0%
23	**Preparation for Final Test**	2024-06-24	2024-07-11	14.0 d.	0.0%
24	Review	2024-06-24	2024-07-11	14.0 d.	0.0%
25	**Verilog Course**	2024-03-03	2024-07-16	97.0 d.	0.0%
26	Course	2024-03-03	2024-06-26	83.0 d.	0.0%
27	Study for final test	2024-06-27	2024-07-16	14.0 d.	0.0%
28	**Competitions**	2024-02-19	2024-07-02	96.0 d.	0.0%
29	Entering Competion	2024-02-19	2024-02-20	0.0 d.	0.0%
30	Creating Special Features	2024-03-13	2024-07-02	69.0 d.	0.0%

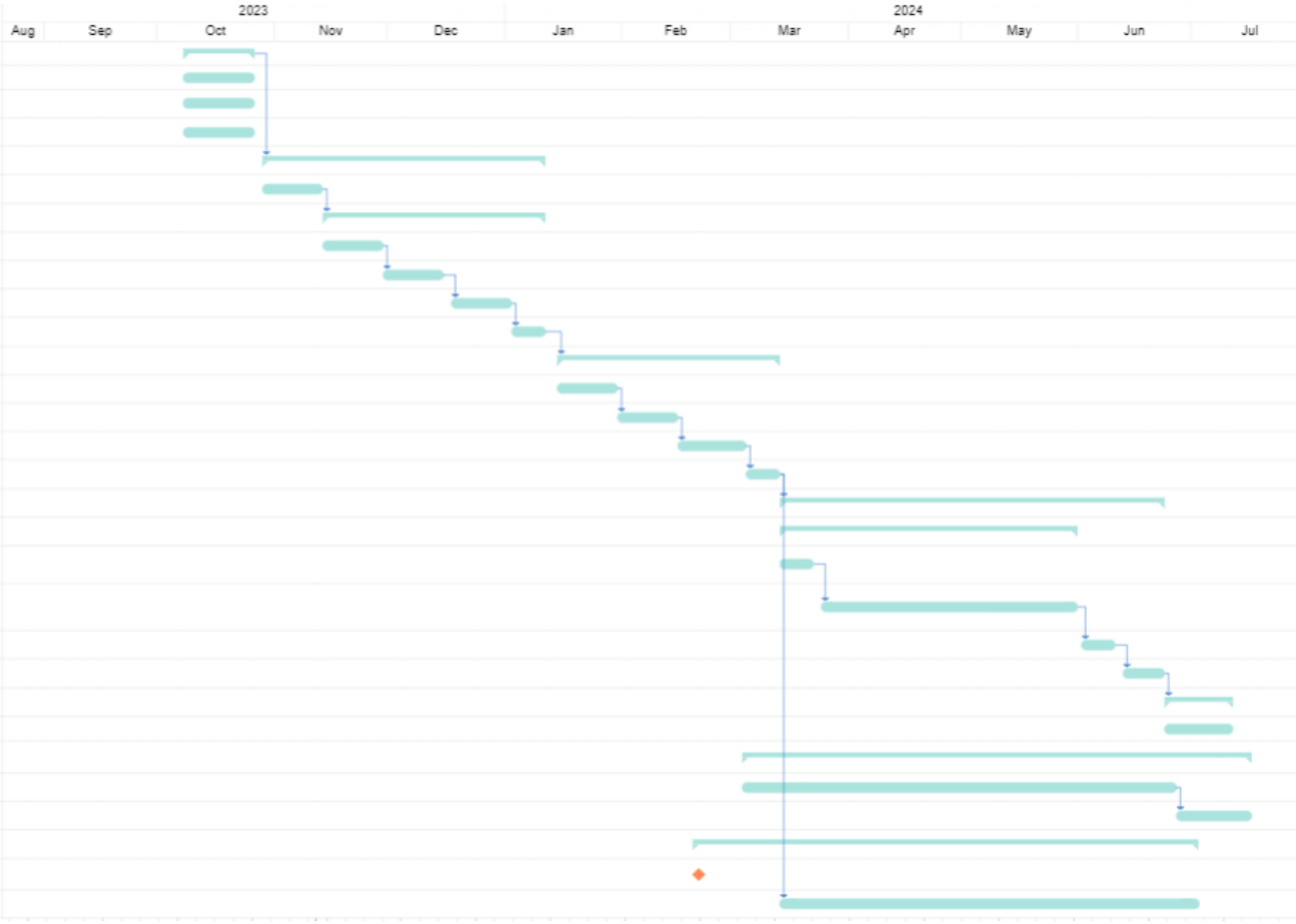

Figure 2 - Gantt Chart

5. List of Academic Topics Covered in the Project

a. Cryptography
b. HDL Languages
c. FPGA
d. Raspberry Pi
e. Hardware Implementation of Encryption Algorithms
f. Digital Hardware Design
g. Verification