# Project Proposal: FPGA-Based Digital Lock System with Bluetooth Authentication

## 1. Project Title

**Secure FPGA-Based Digital Lock System Using AES-128 Encryption and Bluetooth Communication**

## 2. Team Members

* **Phil Nevins**
* **Nick Allmeyer**

## 3. Introduction

Security and access control systems are crucial in various domains, including **home automation, industrial control, and data security**. Traditional locks rely on physical keys, which are susceptible to loss or duplication. This project proposes a **modern, secure, and convenient digital lock system** implemented on an **FPGA** (Nexys A7) with **AES-encrypted authentication over Bluetooth**.

The system allows users to **unlock a secured mechanism** by sending an **AES-encrypted PIN from a mobile application** to the FPGA via an **HC-05 Bluetooth module**. The FPGA decrypts the PIN, compares it with a stored valid PIN, and controls a **servo motor** to unlock or deny access.

## 4. Objectives

The primary objectives of this project are:

1. **Develop an FPGA-based secure lock system** using AES decryption.
2. **Implement a Bluetooth-based authentication mechanism** to receive encrypted PINs from a mobile app.
3. **Design an AES decryption module in Verilog** to verify and decrypt PINs.
4. **Integrate a servo motor** that physically unlocks or locks based on authentication.
5. **Develop a mobile app (Android) that encrypts and sends PINs securely.**
6. **Implement a UART-based data exchange** between FPGA and the mobile app via HC-05.
7. **Test and validate** the security, response time, and functionality of the system.

## 5. System Overview

The digital lock system consists of three major components:

1. **Mobile App (Android)**
   * User enters a PIN.
   * The app encrypts the PIN using AES-128.
   * The encrypted PIN is sent to the FPGA via Bluetooth (HC-05).
   * Stretch Goal: User login with username and password
   * Stretch Goal: Storing new pins of new users in BRAM
2. **FPGA-Based Digital Lock**
   * Receives encrypted PIN via Bluetooth UART.
   * Decrypts the PIN using an AES decryption module implemented in Verilog.
   * Compares the decrypted PIN with a stored value in BRAM (Block RAM).
   * If valid, a servo motor unlocks the lock, and an LED indicates access granted.
   * If invalid, the system denies access and logs the failed attempt.
3. **Lock Mechanism**
   * The servo motor rotates to unlock or lock the system.
   * An LED display shows the lock status (OPEN/CLOSED).
   * A time-based auto-lock feature ensures security after a predefined timeout.

## 6. System Architecture & Components

### 6.1 Hardware Components

* **Nexys A7-100T FPGA Development Board**
* **HC-05 Bluetooth Module**
* **Servo Motor** (for physical lock mechanism)
* **LED Indicators** (status feedback)
* **UART Interface** (for Bluetooth communication)

### 6.2 Software Components

* **Vivado & Vitis** (for FPGA programming and embedded C development)
* **Verilog (HDL)** (for AES decryption and UART communication)
* **Embedded C** (MicroBlaze soft processor for testing/debugging)
* **Android Studio (Java/Kotlin)** (for mobile app development)
* **Python** (for test encryption)

## 7. Technical Approach

### 7.1 Mobile App (Encryption & Bluetooth Communication)

* Users input a 4- to 6-digit PIN.
* The PIN is AES-128 encrypted using a pre-shared key.
* The encrypted PIN is transmitted via Bluetooth (HC-05).
* The mobile app receives lock status updates from the FPGA.

### 7.2 FPGA-Based AES Decryption

* FPGA receives the encrypted PIN via UART.
* The AES decryption module decrypts the received PIN.
* The decrypted PIN is compared to a stored value in BRAM.
* If correct, the servo unlocks, else access is denied.

### 7.3 Servo Motor Control

* If authentication is successful:
  + The servo rotates to unlock the system.
  + An LED changes state to indicate access granted.
  + The system auto-locks after 10 seconds.

## 8. Expected Challenges & Solutions

| **Challenge** | **Solution** |
| --- | --- |
| Securely storing and retrieving AES keys on FPGA | Implement BRAM-based key storage |
| Bluetooth pairing issues | Use predefined MAC addresses for secure pairing |
| Mobile app security | Use SSL for app-to-FPGA communication |

## 9. Project Milestones, Timeline & Work Split

One person will develop the FPGA hardware to decode the AES-128 key and unlock the servo  
One person will develop the Android App to encode the AES-128 key and receive updates from the FPGA  
Both persons will work together to enable BT communication via HC-05 on UART

## 10. Expected Outcomes

By the end of the project, we expect to achieve:

* A fully **functional FPGA-based digital lock**.
* A **secure AES-128 decryption system** on FPGA.
* A **working mobile app** that encrypts and transmits PINs.
* **Successful hardware integration** with a servo-based locking mechanism.

## 11. Conclusion

This project presents a modern digital lock system that leverages FPGA computing power, AES encryption, and Bluetooth communication for enhanced security and real-time access control. The project will provide insights into hardware security, cryptography, and embedded system design, making it an excellent use case for secure authentication in IoT and embedded applications.

## 12. References

1. **AES-128 Specification**: NIST FIPS PUB 197
2. **HC-05 Bluetooth Module Datasheet**
3. **Xilinx Vivado & Vitis Documentation**
4. **MicroBlaze Soft Processor User Guide**