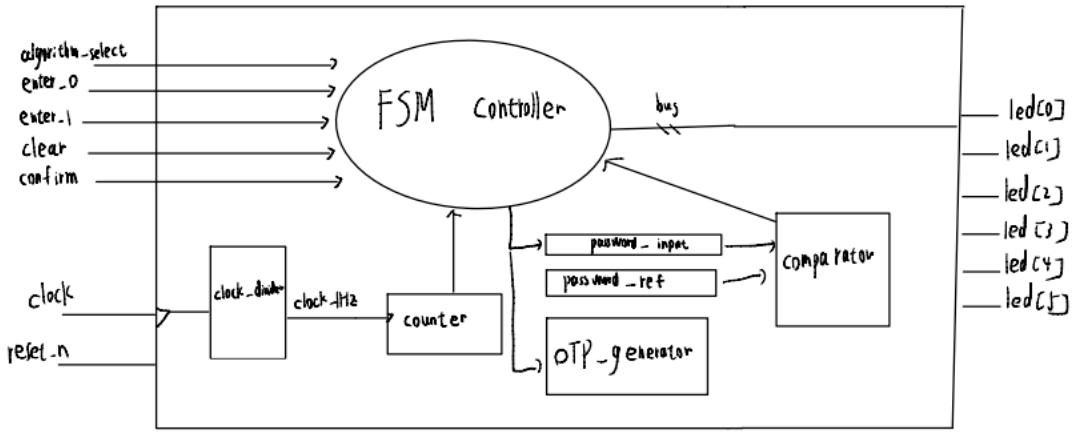
Project Milestone

1. Schematic and data path



System Overview:

This is a password verification system controlled by a Finite State Machine (FSM). It consists of several key components working together to generate a One-Time Password (OTP), accept user input, compare passwords, and display results using LEDs.

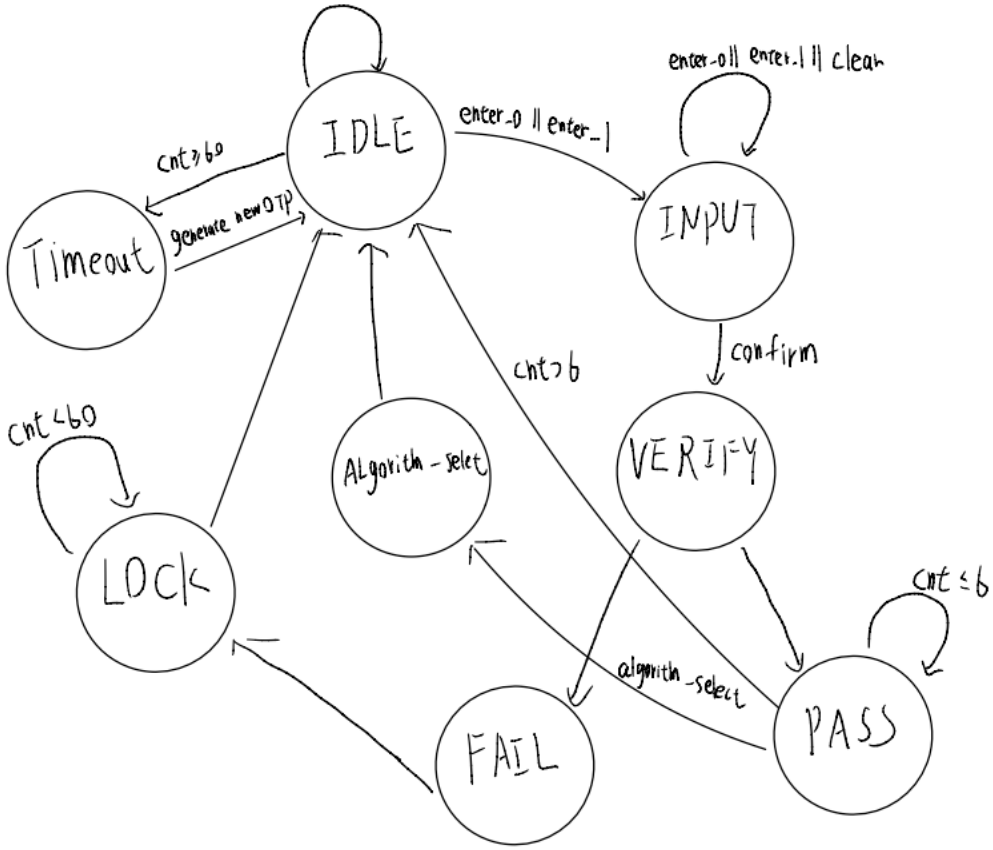
Module Descriptions:

1. **FSM Controller**
   - Acts as the central control unit.
   - Takes in inputs like `algorithm_select`, `enter_0`, `enter_1`, `clear`, and `confirm`.
   - Controls the overall flow of the system including password input, OTP generation, and result display.
2. **Clock Divider (`clock_div`)**
   - Converts a high-frequency clock signal into a 1 Hz clock signal.
   - Ensures the system operates at a manageable pace (one operation per second).
3. **Counter**
   - Driven by the 1 Hz clock.
   - Used for timeout counting, lockout timing.
4. **OTP Generator**
   - Generates a One-Time Password (OTP) using a Linear Feedback Shift Register (LFSR).
   - The `algorithm_select` input allows selection of different OTP generation algorithms.
5. **Password Register (`passwd_reg`)**

- Stores the password entered by the user.
- Inputs are given using `enter_0` and `enter_1` to enter binary digits of the password.

6. **Password Comparator**
   - Compares the user-entered password with the reference.
   - Sends the result back to the FSM for further decision-making.
7. **LED Display**
   - Five LEDs (`led[0]` to `led[5]`) are used to indicate system status, such as:
     - Password correct
     - Password incorrect
     - Show OTP

2. FSM



One important thing to note is that, only after entering correct password, the algorithm_selct State can be entered.

3.TEST

Software Simulation

Goal: To verify the correctness of OTP/password handling logic, timing behaviors

Scenarios to Cover:

(1) Timeout

Test Case 1.1: Wait for more than 60 seconds without input, system should generate a new OTP.

Validation: Assert that a new OTP is generated and old OTP is invalidated.

(2) Correct Password Entry

Test Case 2.1: Input correct password and press Confirm, then system unlocks.

Validation: Check LED signal

(3) Incorrect Password

Test Case 3.1: Input incorrect password once and press Confirm, system is locked.

Test Case 3.2: Input incorrect password three times, system is locked for longer time after each try.

Validation: Check LED signal and new OTP is generated after returning to IDLE

(4) Clear Input

Test Case 4.1: Input some digits and press Clear, password buffer should be empty.

Validation: Check password buffer

(5) Algorithm selection

Test Case 5.1: Enter correct password to enter algorithm selection mode. Enter the binary code to choose different algorithm. And repeat Tests listed before

FPGA-based Hardware Testing

Goal:Validate the design in real hardware, including button inputs, debounce (joggle), and long presses.

Scenarios to Cover:

(1) Button Debounce

Simulate noisy input: rapidly toggle a button pin within short time (simulate bounce).

Validation: Ensure system only registers one press.

(2) Long Button Press

Hold down a button for more than 1 second.

Validation: Should either be interpreted as a single input, or trigger alternate behavior if specified.

(3) All 5 Functional Scenarios from Software

(4) Stress Test

Repeated random input and clear presses

Rapid confirm and cancel toggles