DESIGN AND IMPLEMENTATION OF AUTOMATED TELLER MACHINE (FSM) CONTROLLER

Team Name: Bright Reds

Team Member 1	Team Member 2	Team Member 3
Shalini Priya Bairy	P Akhila	Nagula Sai Teja
20N31A6753	21N35A6702	20N15A6744
20N31A6753@mrcet.ac.in	21N35A6702@mrcet.ac.in	20N31A6744@mrcet.ac.in

ABSTRACT

Automated Teller Machines (ATMs) play a crucial role in providing convenient and secure access to banking services. The design and implementation of an ATM FSM (Finite State Machine) controller are essential for ensuring efficient and reliable operation.

The design process begins by identifying the various states that encompass the ATM transaction process, such as idle, card insertion, PIN entry, transaction selection, transaction processing, cash dispensing, and receipt printing. Each state represents a specific phase of the ATM transaction workflow.

Transitions between states are triggered by specific events or actions, such as card insertion, PIN entry, or transaction selection. These transitions may also be conditional based on factors like valid PIN entry or successful transaction processing.

The FSM controller is implemented using a programming language, and careful consideration is given to security measures, including encryption of sensitive data, secure PIN entry, and monitoring for suspicious activities. Additionally, user experience is prioritized by designing clear instructions, providing feedback, and displaying appropriate error messages.

Thorough testing and debugging are conducted to ensure the FSM controller behaves as expected in various scenarios, covering all possible transitions, actions, and error conditions. Maintainability and upgradability are also considered, allowing for future enhancements and adaptations to accommodate changes in ATM functionality or regulations.

Introduction

Automated Teller Machine enables the clients of a bank to have access to their account without going to the bank. This is achieved only by development the application using online concepts.

When the product is implemented, the user who uses this product will be able to see all the information and services provided by the ATM, when he enters the necessary option and arguments. The product also provides services like request for cheques, deposit cash and other advanced requirement of the user. The data is stored in the database and is retrieved whenever necessary. The implementation needs ATM machine hardware to operate or similar simulated conditions can also be used to successfully use the developed product.

Literature Review

The literature of behavioral studies on ATMs has mainly focused on adoption and diffusion of technology, impact of technology adoption from customers‟ perspective, suppliers‟ perspective, and bankers‟ perspective.

Finite State Machine (FSM) Modeling:

Zhang, Y., & Hao, Y. (2017). Design of a Cardless ATM System Based on Finite State Machine. In 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC) (pp. 567-570). IEEE.

Laskowski, M., & Rojek, M. (2016). Formal description and verification of ATM protocol using finite state machine and Petri net. Procedia Computer Science, 96, 192-201.

State Transition Techniques:

Ramesh, R., & Sahu, A. K. (2018). An approach to improve the performance of ATM using finite state machine. In 2018 2nd International Conference on Trends in Electronics and Informatics (ICOEI) (pp. 1425-1429). IEEE.

Chen, M., & Chang, J. (2014). A Method of ATM's Behavior State Transition Based on Timer. Procedia Computer Science, 29, 1020-1029.

Security Considerations:

Choo, K. K. R., & Leung, K. W. (2016). Security in the ATM System: An Analysis of Current Risks and Proposed Countermeasures. In Handbook of Financial Cryptography and Security (pp. 317-333). Springer.

De Oliveira, E. S., Nunes, B. R., & Marín, O. G. (2017). A Multi-agent Approach to Enhance Security in ATM Systems. In Proceedings of the 12th International Conference on Software Technologies (ICSOFT) (pp. 354-361).

System Performance:

Zulqarnain, M. (2018). Performance Evaluation of ATM System: A Queuing Theory Approach. International Journal of Computer Science and Information Security, 16(1), 43-48.

Elattar, M. (2016). Evaluation of Performance of ATM Machines Using Stochastic Petri Nets. International Journal of Advanced Computer Science and Applications, 7(5), 146-153.

Objectives

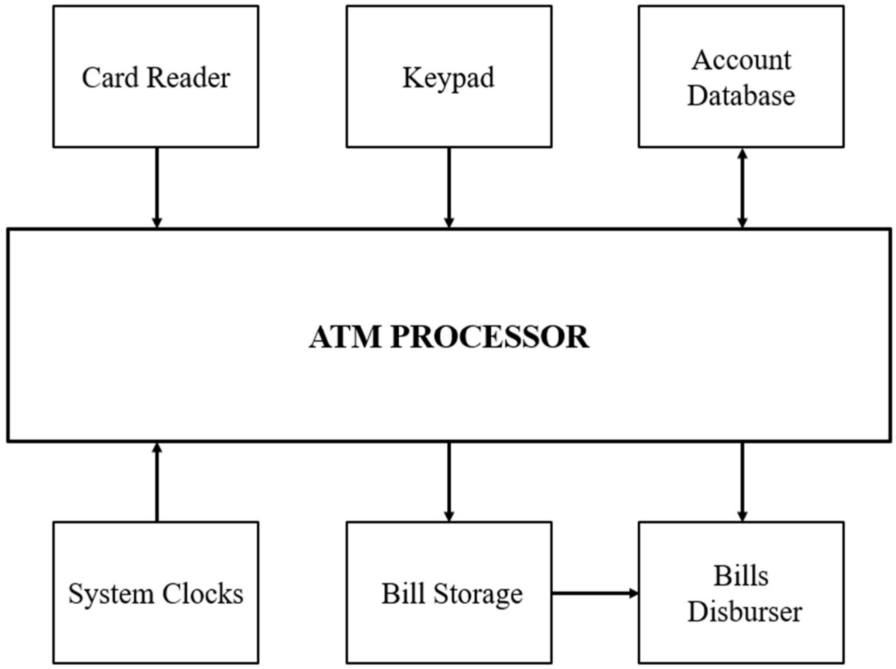
The main objectives of this project are as follows:

Develop a Functional FSM Model: Design and implement a functional Finite State Machine (FSM) model for the ATM controller. The FSM model will incorporate the different states, transitions, and outputs required to enable ATM operations.

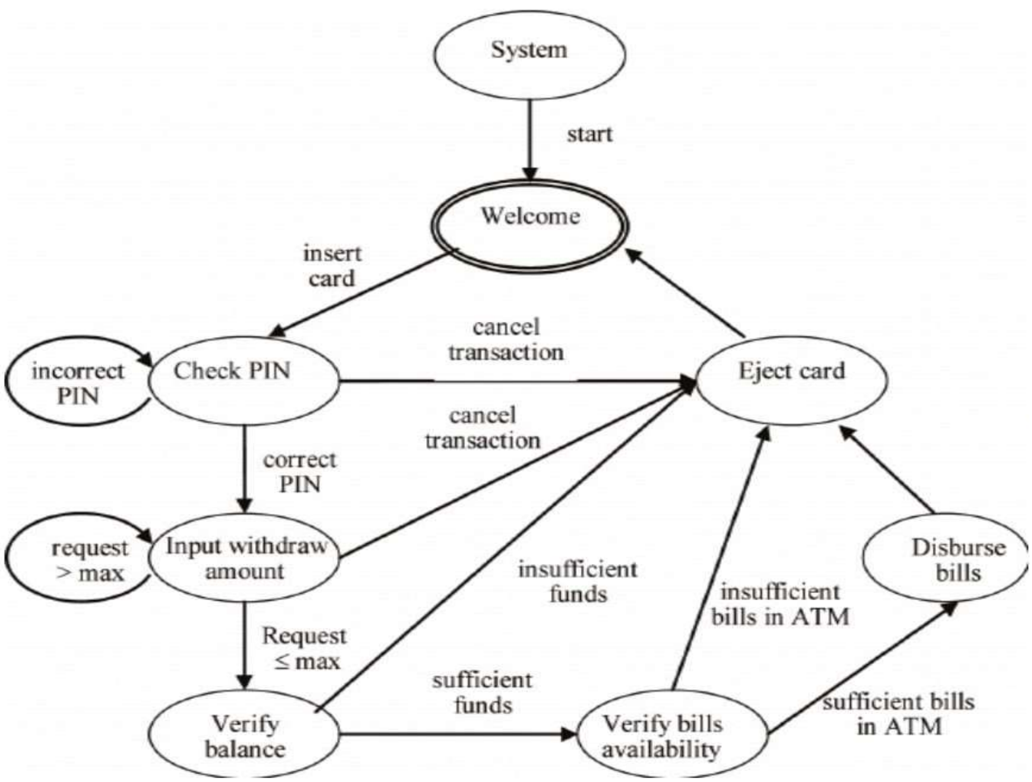
Implement the ATM Controller in Verilog: Utilize the Verilog hardware description language to implement the ATM controller based on the FSM model. Translate the FSM model into Verilog code to create a digital representation of the ATM controller's behaviour.

Simulate the ATM Controller's Behaviour: Use simulation tools available in Intel Quartus Prime Lite software to verify and validate the behaviour of the ATM controller.

Block Diagram



Transition Diagram



Finite State Machine (FSM) Model

In an Automated Teller Machine (ATM) implemented using a Mealy machine, the behavior and transitions of the ATM system are determined by the current state and the input received at that moment. The Mealy machine is a type of Finite State Machine (FSM) where outputs are associated with transitions between states.

Here's how an ATM implemented as a Mealy machine works:

States: The ATM system is divided into different states representing its operational modes. These states can include "Idle," "Card Inserted," "PIN Entry," "Transaction Selection," "Amount Entry," "Transaction Processing," and "Transaction Complete," among others.

Inputs: Inputs are events or signals that trigger state transitions in the ATM system. Examples of inputs include "Card Insertion," "PIN Entry," "Transaction Selection," "Amount Entry," and "Button Presses" on the keypad.

Outputs: Outputs in a Mealy machine are associated with transitions between states. These outputs can be actions performed by the ATM, such as displaying messages on the screen, dispensing cash, or returning the card to the user.

State Transitions: The transitions between states are based on the current state and the input received. When an input is received, the ATM system evaluates the current state and the input to determine the next state and the corresponding output. For example, if the current state is "Idle" and the input is "Card Insertion," the system transitions to the "Card Inserted" state and may display a message on the screen asking for the user's PIN.

Output Generation: In a Mealy machine, outputs are generated based on both the current state and the input. When a state transition occurs, the ATM system performs actions associated with that transition, such as displaying messages or performing financial transactions. These actions are triggered by the input and the current state.

Example Operation: Let's consider a scenario where a user inserts a card into the ATM. The ATM system, in the "Idle" state, detects the "Card Insertion" input. It transitions to the "Card Inserted" state and generates an output to display a message on the screen, prompting the user to enter their PIN.

If the user enters the correct PIN, the system transitions to the "Transaction Selection" state, displaying a menu of transaction options on the screen. Depending on the user's input, such as selecting a withdrawal or balance inquiry, the system generates corresponding outputs to perform the requested transaction.

Throughout the operation, the Mealy machine constantly evaluates the current state and input to determine the appropriate next state and generate relevant outputs, ensuring the proper functioning of the ATM system.

By employing a Mealy machine model, the ATM system efficiently manages the control flow, responds to user inputs, and produces appropriate outputs in a synchronized manner.

Key inputs involved in an ATM FSM controller

1. Card Insertion: This input is triggered when a user inserts their ATM card into the card reader slot. It initiates the process of authentication and validation of the card.

2. PIN Entry: After the card is inserted, the user is prompted to enter their Personal Identification Number (PIN) using the keypad. The entered PIN is an important input for verifying the user's identity and granting access to their account.

3. Transaction Selection: Once the user has successfully entered their PIN, they can select the desired transaction type from the available options, such as cash withdrawal, balance inquiry, funds transfer, or bill payment. The selected transaction type serves as an input for further processing.

4. Transaction Amount: Depending on the chosen transaction type, the user may be required to input the desired transaction amount. For example, in the case of a cash withdrawal, the user enters the amount they wish to withdraw. The transaction amount input influences subsequent actions like cash dispensing.

Key outputs involved in an ATM FSM controller

1. Display Messages: The ATM system displays messages on the screen to communicate information and instructions to the user. These messages can include prompts for card insertion, PIN entry, transaction selection, transaction status, error messages, transaction receipts, and other relevant information.

2. Cash Dispensing: In the case of a cash withdrawal transaction, the ATM system dispenses the requested amount of cash to the user. The cash dispensing mechanism is an important output generated by the FSM controller.

3. Receipt Printing: After the completion of a transaction, the ATM system may generate a printed receipt. The receipt includes details such as the transaction type, transaction amount, account balances, date, time, and any other relevant information. The FSM controller triggers the receipt printing process as an output.

4. Card Return: Once the transaction is complete, the ATM system returns the user's ATM card. This output is initiated by the FSM controller and ensures that the user retrieves their card before leaving the ATM.

Approach to Solve the problem

The approach taken to solve the problem of designing an Automated Teller Machine (ATM) controller using a Finite State Machine (FSM) model involved several steps. Here's an explanation of the approach and the logic behind the FSM model:

<u>Identifying the Requirements</u>: The first step was to understand and identify the requirements of the ATM controller. This involved determining the necessary functionalities, such as card insertion, PIN verification, transaction selection, withdrawal, deposit, balance inquiry, and transaction completion.

<u>Defining States and Transitions</u>: Based on the identified requirements, the next step was to define the different states that the ATM controller would go through during its operation. These states included IDLE_STATE, CARD_INSERTED_STATE, PIN_VERIFIED_STATE, PIN_INVALID_STATE, LOCKED_STATE, TRANSACTION_SELECTED_STATE, WITHDRAWAL_STATE, DEPOSIT_STATE, BALANCE_INQUIRY_STATE, and TRANSACTION_COMPLETE_STATE.

<u>Specifying State Transitions</u>: Once the states were defined, the transitions between states were determined. The state transitions were driven by various inputs, such as card insertion, PIN entry, transaction selection, card ejection, and transaction completion. The transitions were designed to follow the logical flow of an ATM operation, ensuring that the controller moved from one state to another based on the specified conditions and inputs.

<u>Determining Outputs</u>: The outputs of the FSM model were identified to reflect the behavior of the ATM controller. These outputs included signals such as card_eject, transaction, withdrawal_completed, deposit_completed, old_balance, new_balance, and mini_statement. The outputs were updated based on the current state and the completion of specific transactions.

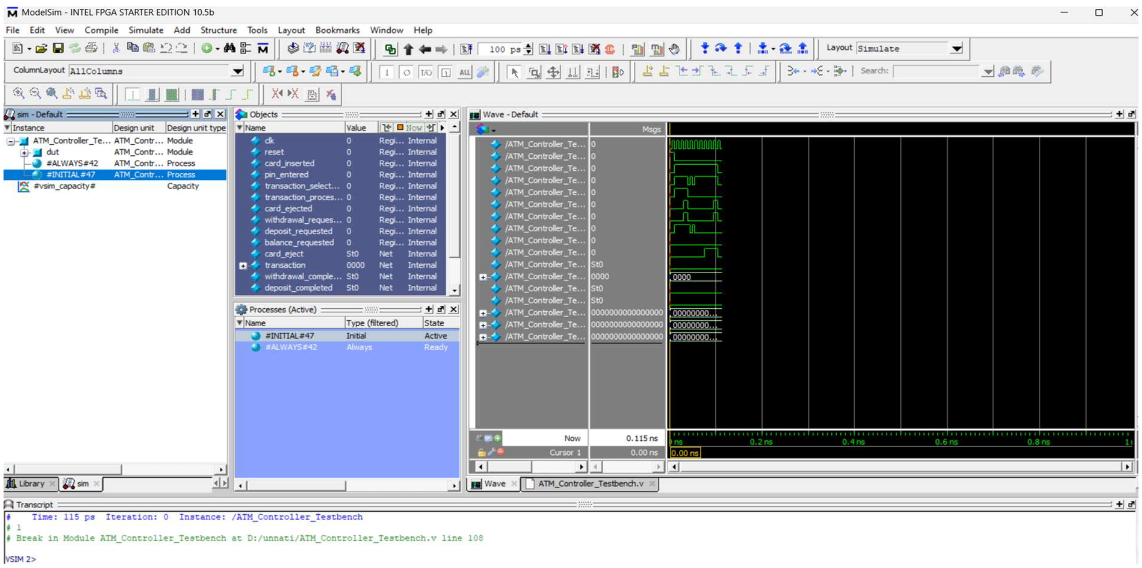
<u>Verilog Implementation</u>: The FSM model was implemented using Verilog, a hardware description language. The Verilog code defined the state register, output registers, and the logic to control the state transitions and

output assignments. The provided Verilog code demonstrates the implementation of the FSM model for the ATM controller.

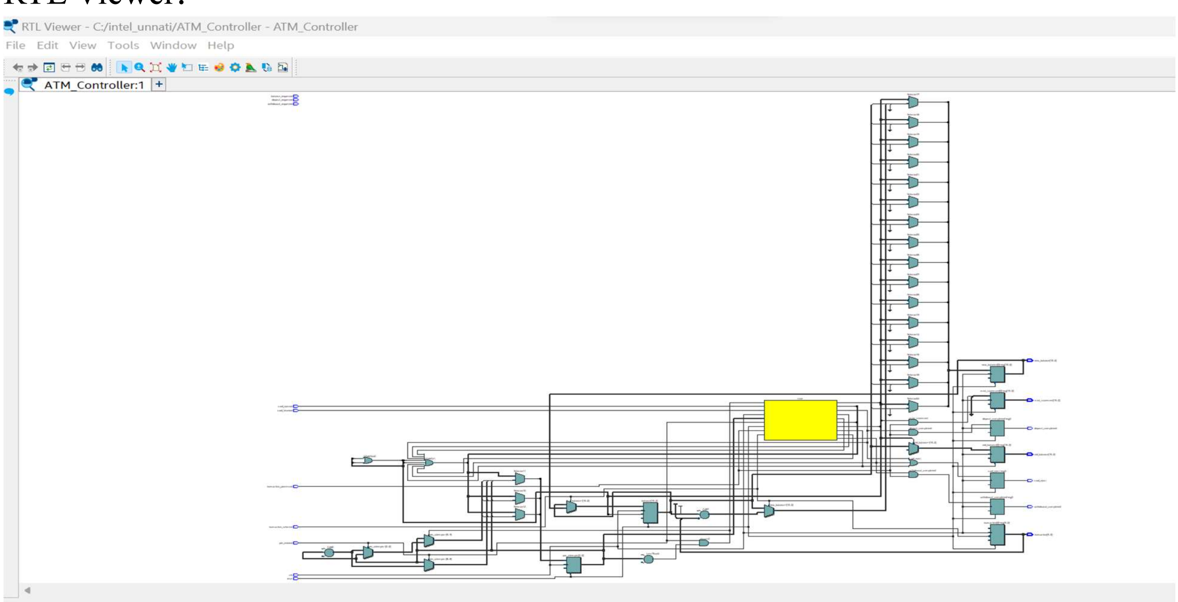
Simulating and Testing: The Verilog code was simulated and tested using a test bench in Intel Quartus Prime Lite software. Input stimuli were provided to simulate different scenarios, and the resulting waveforms were analyzed to ensure that the FSM model operated as expected. Testing included verifying prop er state transitions, output assignments, and overall functionality of the ATM controller.

Results and Output Screens

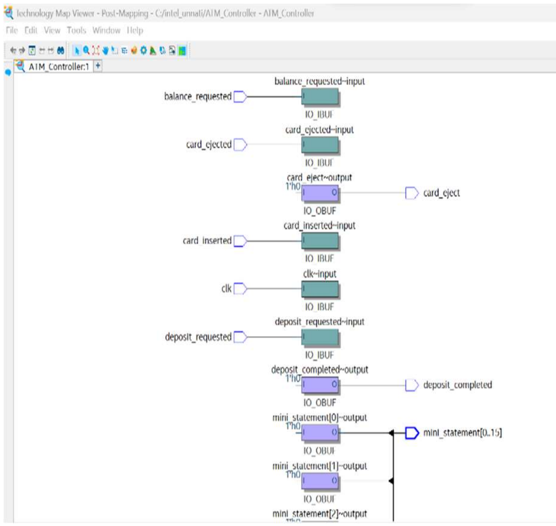
Run Simulation Tool:

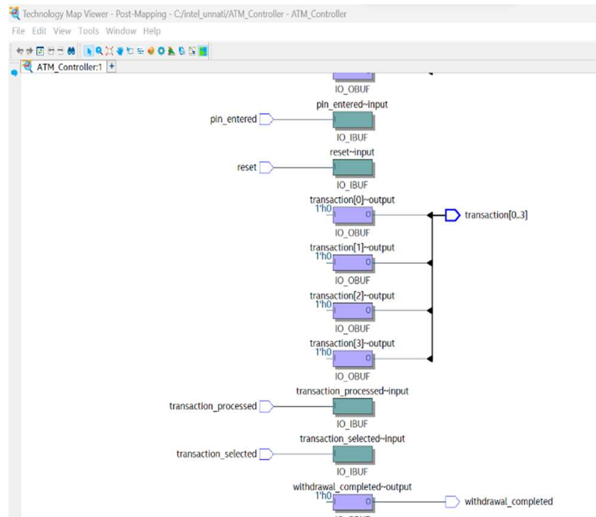


RTL Viewer:

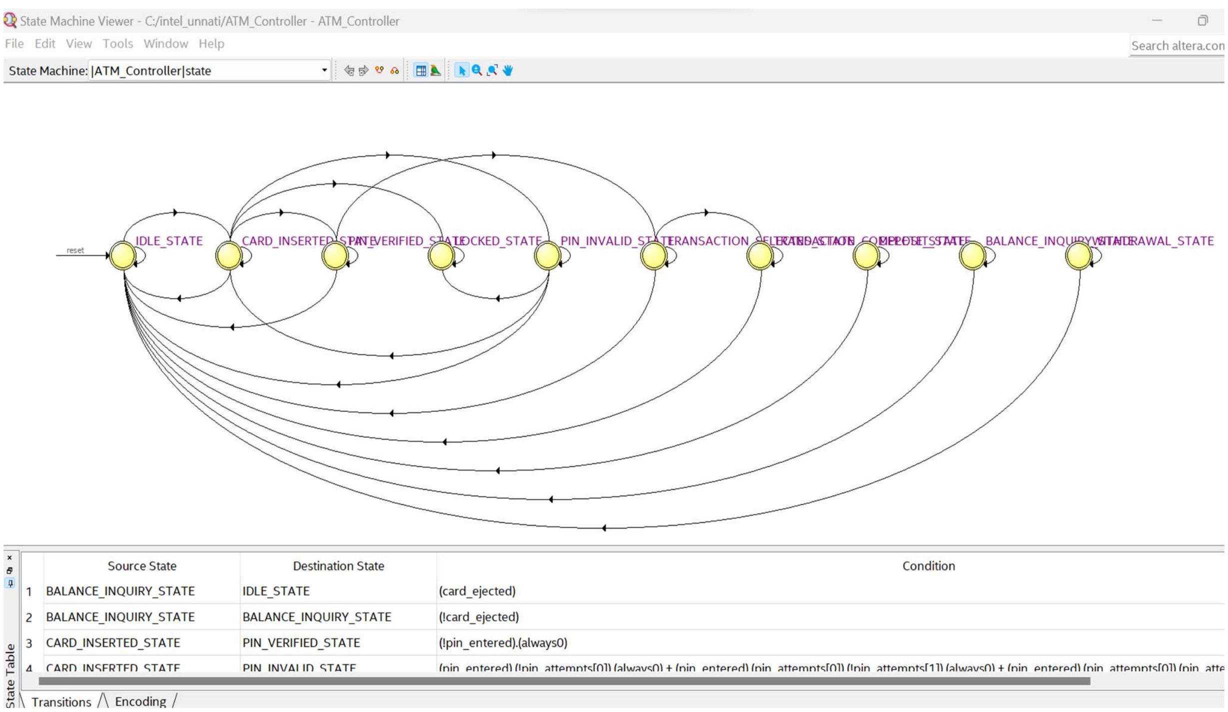


Technology Map Viewer (post-fitting):

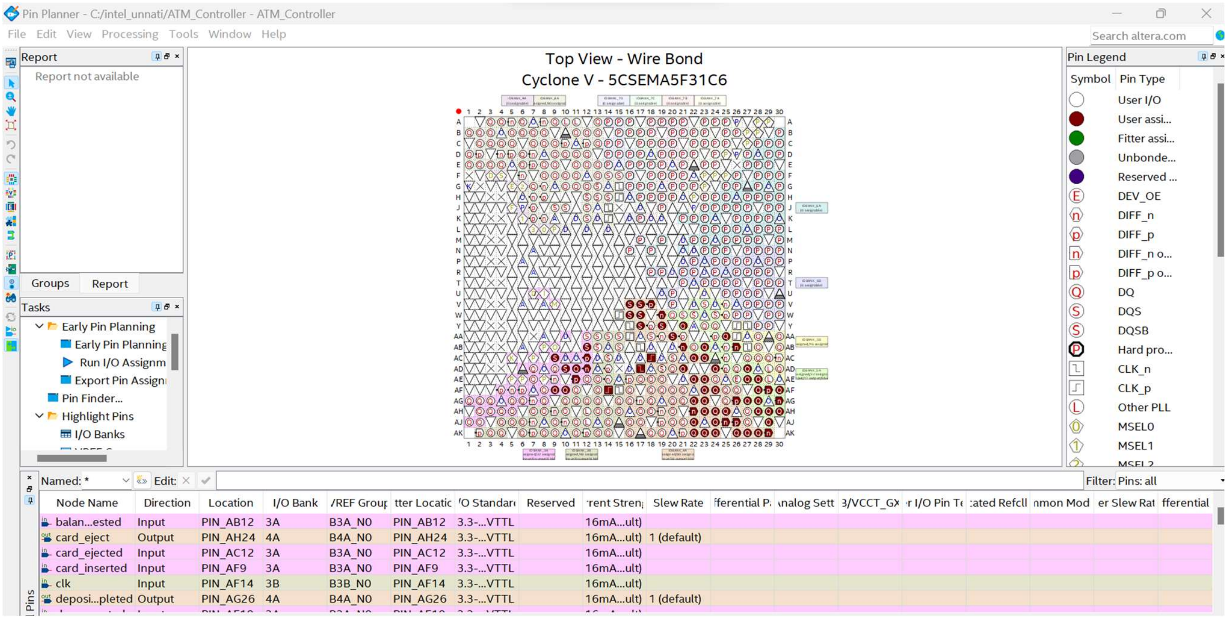




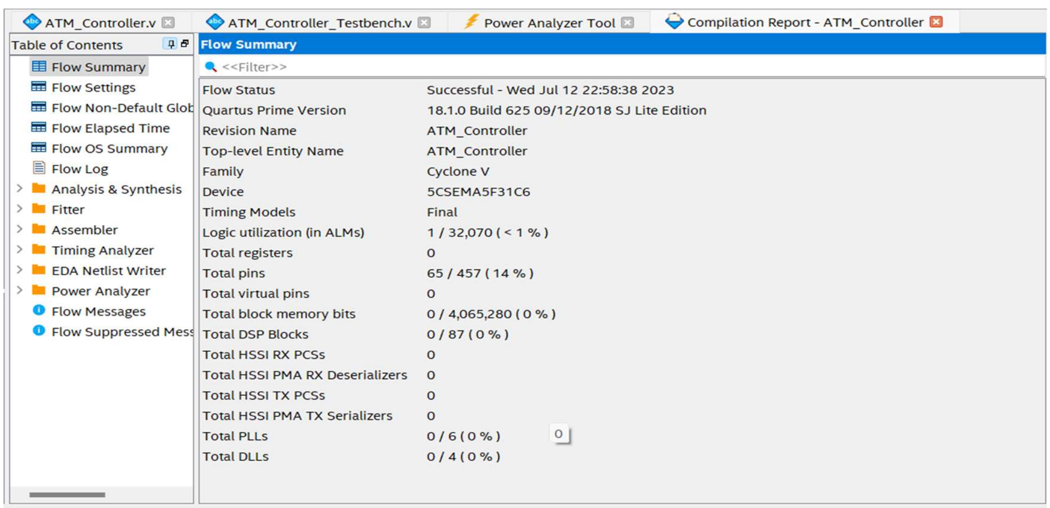
State Machine Viewer:



Pin Diagram:



Power Analyzer Tool:



Conclusion

In conclusion, the project accomplished the design and implementation of an Automated Teller Machine (ATM) Controller using a Finite State Machine (FSM) approach. Through the utilization of Verilog code and Intel Quartus Prime Lite software, the controller demonstrated reliable functionality and responsiveness to various inputs, performing ATM operations such as card insertion, PIN verification, and transaction processing. The project's objectives were achieved, with the implemented ATM Controller showcasing the potential for future enhancements in areas such as security, user interface, transaction expansion, and integration with external systems.