Vending Machine Project Documentation

Project Team

- Olha Melnyk (Aluno27384)
- Sami Erdogmus (Aluno27375)
- İrem Akpınar (Aluno27364)

Project Overview

This project involves the design and implementation of a vending machine system that allows users to purchase chocolate by inserting coins. The machine operates based on a finite state machine (FSM) modeled as a Moore Machine. The aim of the project is to simulate the functionality of a real-world vending machine using digital logic design and FPGA programming.

Key Features

1. **Finite State Machine (FSM):**
   - The vending machine operates using a Moore Machine FSM.
   - The FSM transitions through various states based on the number of coins inserted and user actions (e.g., pressing a button to dispense chocolate).
2. **States and Outputs:**
   - The machine transitions between states such as Idle, Coin 1, Coin 2, and Dispense states.
   - Each state generates specific outputs to control the machine's components (e.g., activating dispensers).
3. **Digital Logic Design:**
   - The FSM is implemented using J-K flip-flops and combinational logic gates.
   - Boolean equations define the transitions and outputs for each state.
4. **FPGA Integration:**
   - The vending machine logic is deployed on an FPGA (MAX1000).
   - The design includes inputs (e.g., coin detector, selection buttons) and outputs (e.g., dispenser activators).
5. **User Interaction:**
   - Users insert coins and make a selection to receive chocolate.
   - The system resets to the idle state after dispensing the product.

Project Components

1. **State Diagram:**
    - Visual representation of the FSM, showing state transitions and corresponding outputs.
2. **Truth Table:**
    - A table defining the relationship between current states, inputs, next states, and outputs.
3. **Boolean Equations:**
    - Logical expressions derived from the FSM to control state transitions and outputs.
4. **Circuit Diagram:**
    - A schematic showing the implementation of the FSM using logic gates and flip-flops.
5. **Simulation Results:**
    - Waveform analysis to verify the correct operation of the FSM under various input conditions.

Challenges and Solutions

- **Challenge:** Implementing accurate state transitions.
    - **Solution:** Derived Boolean equations for each state transition and validated them using simulation tools.
- **Challenge:** Integrating hardware components on the FPGA.
    - **Solution:** Mapped inputs and outputs to specific pins on the FPGA and tested functionality.

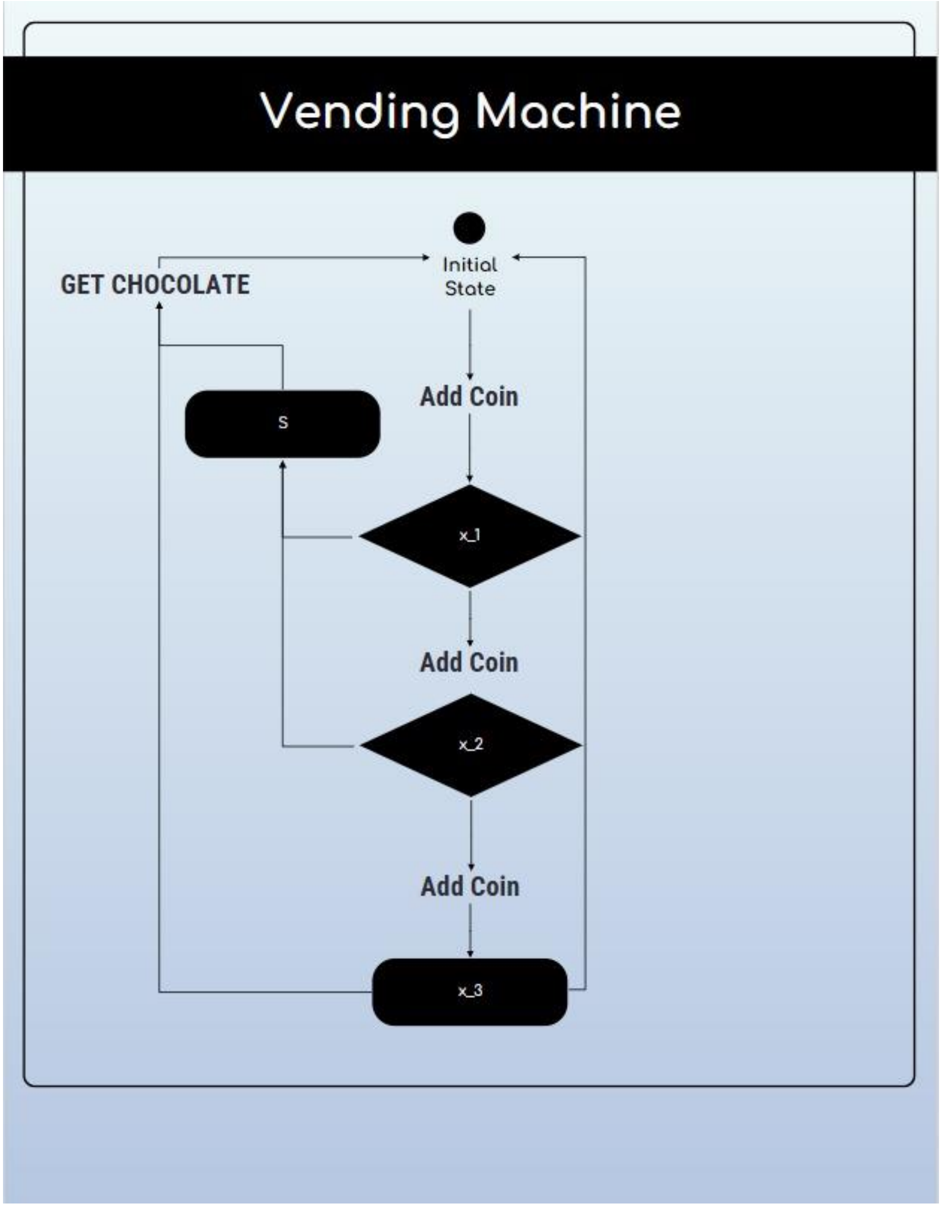
Future Improvements

- Expand the machine's functionality to support multiple product types.
- Add features such as a display screen to show the remaining balance or product selection.
- Implement a real-time clock to manage power-saving modes.

Conclusion

The vending machine project demonstrates the application of finite state machines and digital logic in real-world scenarios. By integrating the design into an FPGA, the project highlights the practical implementation of theoretical concepts in digital systems.

Flowchart for the Vending Machine Process



This flowchart represents the vending machine's operational logic, detailing the sequential steps from coin insertion to product dispensing. Here is a detailed explanation of each component:

Initial State:

- The vending machine begins in the **Initial State**, awaiting user input. No coins have been inserted at this stage.

Adding Coins:

1. **First Coin (x1):**

    a. When the first coin is inserted, the system transitions to the next state and registers the coin.
    b. At this point, the vending machine is ready to accept additional coins or a product selection.
2. **Second Coin (x2):**
    a. Upon inserting a second coin, the machine transitions to the next state, acknowledging the second coin.
    b. The system checks if the required amount has been reached for dispensing a product.
3. **Third Coin (x3):**
    a. Inserting a third coin allows the vending machine to prepare for product dispensing.

Product Dispensing (Get Chocolate):

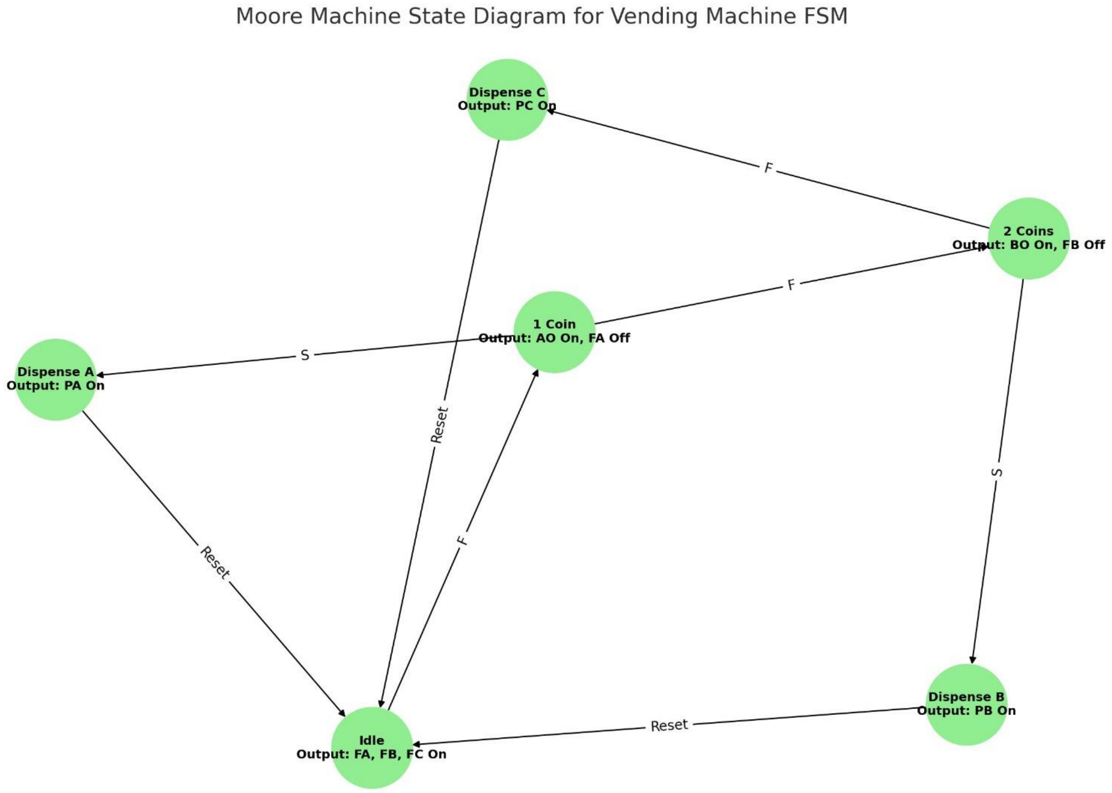
- When the user presses the **S** button, the system dispenses the selected product (chocolate).
- After dispensing, the machine resets itself to the **Initial State**, ready for the next transaction.

Key Elements of the Flowchart:

- **State Transitions:** Each decision point (x1 ,x2 ,x3) corresponds to a new state in the vending machine's FSM, ensuring a smooth progression through the system.

Reset Mechanism: The reset feature ensures that after dispensing a product, the machine returns to its idle state for the next user.

Moore Machine State Diagram Explanation



This state diagram represents the finite state machine (FSM) of a vending machine implemented as a **Moore Machine**, where outputs depend only on the current state. Below is a detailed explanation of the states, transitions, and outputs:

States

1. **Idle (Initial State):**
    a. Outputs: `FA = On`, `FB = On`, `FC = On`.
    b. Represents the starting point of the vending machine, waiting for user input (coin insertion).
    c. Transition:
        i. **F (Coin inserted):** Moves to the `1 Coin` state.
2. **1 Coin:**
    a. Outputs: `AO = On`, `FA = Off`.
    b. Indicates that one coin has been inserted. The machine is waiting for additional coins or a selection.
    c. Transitions:
        i. **F (Another coin inserted):** Moves to the `2 Coins` state.
        ii. **S (Selection made):** Resets to `Idle` state without dispensing a product.

3. **2 Coins:**
    a. Outputs: `BO = On`, `FB = Off`.
    b. Two coins have been inserted. At this point, the user can choose a product or continue adding coins.
    c. Transitions:

i. **F (Another coin inserted):** Moves to the `Dispense C` state.
ii. **S (Selection made):** Resets to `Idle` state without dispensing.

4. **Dispense A:**
    a. Outputs: `PA = On`.
    b. Dispenses the product associated with the first selection.
    c. Transition:
        i. **Reset:** Returns to the `Idle` state.
5. **Dispense B:**
    a. Outputs: `PB = On`.
    b. Dispenses the product associated with the second selection.
    c. Transition:
        i. **Reset:** Returns to the `Idle` state.
6. **Dispense C:**
    a. Outputs: `PC = On`.
    b. Dispenses the product associated with the third selection.
    c. Transition:
        i. **Reset:** Returns to the `Idle` state.

Transitions

- **F (Coin inserted):**
    - Causes the state to progress based on the number of coins inserted.
    - Triggers movement from `Idle` → `1 Coin` → `2 Coins` → `Dispense C`.
- **S (Selection made):**
    - Depending on the current state, moves to the respective dispense state (`Dispense A`, `Dispense B`, `Dispense C`).
- **Reset:**
    - Used after dispensing a product to bring the machine back to the `Idle` state.

Outputs

- **Idle State:** All lights (`FA`, `FB`, `FC`) are on, signaling readiness for operation.
- **Coin States:**
    - Outputs like `AO` or `BO` indicate the progression of coins inserted and deactivate previous indicators (e.g., `FA` turns off in the `1 Coin` state).
- **Dispense States:**
    - Outputs `PA`, `PB`, or `PC` activate to dispense the corresponding product based on the selection.

Boolean Equations Overview

1. $J_0 = \overline{Q_0} \cdot F, K_0 = Q_0 \cdot \overline{F}$
2. $J_1 = \overline{Q_1} \cdot Q_0 \cdot F, K_1 = Q_1 \cdot (\overline{Q_0} + \overline{F})$
3. $J_2 = \overline{Q_2} \cdot Q_1 \cdot Q_0, K_2 = Q_2 \cdot (\overline{Q_1} + \overline{Q_0})$
4. $PA = Q_0 \cdot \overline{Q_1} \cdot S, PB = Q_0 \cdot Q_1 \cdot \overline{Q_2} \cdot S, PC = Q_0 \cdot Q_1 \cdot Q_2 \cdot S$
5. $AO = Q_0, BO = Q_0 \cdot Q_1, CO = Q_0 \cdot Q_1 \cdot Q_2$

This image outlines the Boolean equations that govern the state transitions and outputs of the vending machine's finite state machine (FSM). These equations ensure the correct behavior of the system by defining the inputs for the flip-flops (J and K inputs) and the output signals for the vending machine.

1. Flip-Flop Input Equations:

- $J_0 = \overline{Q_0} \cdot F$:
  - The first flip-flop (Q0) is set when it is currently inactive (!*Q_0*) and a coin input (F) is detected.
- $K_0 = Q_0 \cdot \overline{F}$:
  - The first flip-flop (Q0) resets when it is currently active (Q_0) and no coin input (!F) is detected.
- $J_1 = \overline{Q_1} \cdot Q_0 \cdot F$:
  - The second flip-flop (!Q_1) is set when the first flip-flop (Q_0) is active, and a coin (F) is inserted.
- $K_1 = Q_1 \cdot (\overline{Q_0} + \overline{F})$:
  - The second flip-flop (Q1) resets if either Q0 is inactive (!Q_0) or no coin (!F) is detected.
- $J_2 = \overline{Q_2} \cdot Q_1 \cdot Q_0$:
  - The third flip-flop (!Q2) is set when both Q1 and Q0 are active, and Q2 is currently inactive (!Q2).
- $K_2 = Q_2 \cdot (\overline{Q_1} + \overline{Q_0})$:
  - The third flip-flop (Q2) resets if either !Q1 or !Q0 becomes inactive.

2. Output Equations for Product Dispensing:

- $PA = Q_0 \cdot \overline{Q_1} \cdot S$:
  - The dispenser for product A (PA) is activated if:
    - Q0 is active (indicating one coin is inserted),
    - Q1 and Q2 are inactive (indicating only one coin is inserted),
    - The user has pressed the selection button (S).
- $PB = Q_0 \cdot Q_1 \cdot \overline{Q_2} \cdot S$:
  - The dispenser for product B (PB) is activated if:
    - Q0 and Q1 are active (indicating two coins are inserted),
    - Q2 is inactive,
    - The user has pressed the selection button (S).
- $PC = Q_0 \cdot Q_1 \cdot Q_2 \cdot S$:

- The dispenser for product C (PC) is activated if:
    - Q0, Q1, and Q2 are all active (indicating three coins are inserted),
    - The user has pressed the selection button (SSS).

3. Auxiliary Output Equations:

- $AO = Q_0$:
    - This output indicates that one coin has been inserted.
- $BO = Q_0 \cdot Q_1$:
    - This output indicates that two coins have been inserted.
- $CO = Q_0 \cdot Q_1 \cdot Q_2$:
    - This output indicates that three coins have been inserted.

Truth Table for the Vending Machine FSM

Current State (Q1, Q2)	Input (Coin, S)	Next State (Q1', Q2')	Output (FA, FB, FC, AO, BO, C)	J1	K1	J2	K2
00 (Idle)	Coin	01 (Coin1)	FA=0, FB=1, FC=1, AO=1	1	0	0	X
01 (Coin1)	S	00 (Idle)	FA=1, FB=1, FC=1	0	1	X	1
01 (Coin1)	Coin	10 (Coin2)	FA=0, FB=0, FC=1, BO=1	0	X	1	0
10 (Coin2)	S	00 (Idle)	FA=1, FB=1, FC=1	X	1	0	1
10 (Coin2)	Coin	11 (Coin3)	FA=0, FB=0, FC=0, C=1	1	0	1	0
11 (Coin3)	-	00 (Idle)	FA=1, FB=1, FC=1	X	1	X	1

This truth table illustrates the behavior of the vending machine's finite state machine (FSM) by describing the current state, inputs, next state, outputs, and flip-flop inputs. Each row represents a unique scenario in the vending machine's operation, guiding its transitions and functionality.

1. Current State (Q1, Q2):

- Represents the current state of the FSM, encoded using two flip-flops:
    - `00 (Idle)`: The initial state where no coins have been inserted.
    - `01 (Coin1)`: Indicates one coin has been inserted.
    - `10 (Coin2)`: Indicates two coins have been inserted.
    - `11 (Coin3)`: Indicates three coins have been inserted.

2. Inputs (Coin, S):

- `Coin`: Represents the insertion of a coin into the vending machine.
- `S`: Represents the selection signal when the user chooses to dispense a product.

3. Next State (Q1', Q2'):

- Describes the state the FSM transitions to based on the current state and inputs.

4. Outputs (FA, FB, FC, AO, BO, C):

- Control signals that indicate the vending machine's status:
  - `FA`, `FB`, `FC`: Status indicators for readiness or reset.
  - `AO`, `BO`: Auxiliary outputs signaling coin detection (e.g., one coin or two coins).
  - `C`: Dispensing signal indicating the machine is ready to release a product.

5. Flip-Flop Inputs (J1, K1, J2, K2):

- Specify the set (`J`) and reset (`K`) inputs for the flip-flops controlling state transitions.

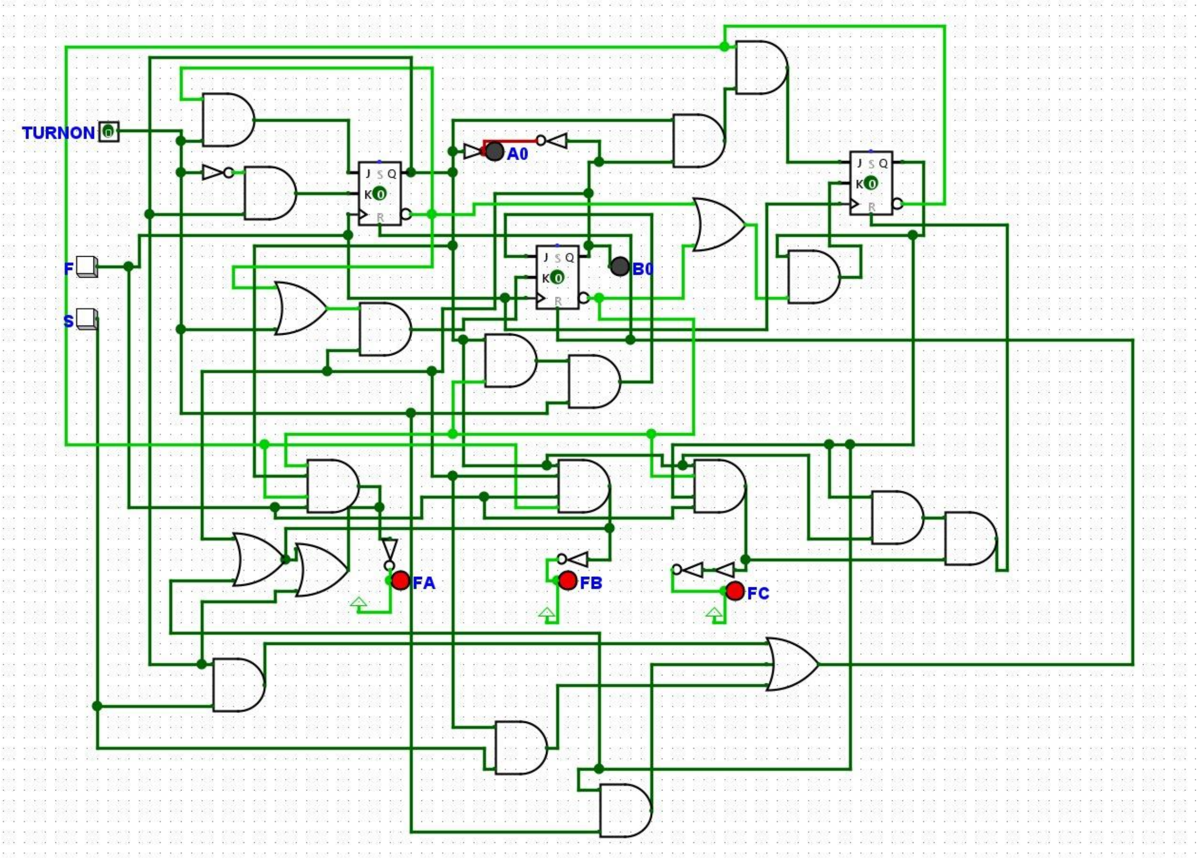
Row-by-Row Details:

1. **Idle State to Coin1 State:**
   a. **Current State:** `00 (Idle)`
   b. **Input:** Coin inserted.
   c. **Next State:** `01 (Coin1)` - Transitions to the first coin state.
   d. **Output:** `FA=0, FB=1, FC=1, AO=1` - Indicates readiness and coin detection.
   e. **Flip-Flop Inputs:** `J1=1, K1=0, J2=X, K2=X`
2. **Coin1 State to Idle State:**
   a. **Current State:** `01 (Coin1)`
   b. **Input:** Selection signal (`S`).
   c. **Next State:** `00 (Idle)` - Resets back to the idle state.
   d. **Output:** `FA=1, FB=1, FC=1` - Signals reset with no product dispensed.
   e. **Flip-Flop Inputs:** `J1=0, K1=1, J2=1, K2=1`
3. **Coin1 State to Coin2 State:**
   a. **Current State:** `01 (Coin1)`
   b. **Input:** Coin inserted.
   c. **Next State:** `10 (Coin2)` - Transitions to the two-coin state.
   d. **Output:** `FA=0, FB=0, FC=1, BO=1` - Indicates two coins detected.
   e. **Flip-Flop Inputs:** `J1=X, K1=1, J2=1, K2=0`
4. **Coin2 State to Idle State:**
   a. **Current State:** `10 (Coin2)`
   b. **Input:** Selection signal (`S`).
   c. **Next State:** `00 (Idle)` - Resets back to the idle state.
   d. **Output:** `FA=1, FB=1, FC=1` - Signals reset with no product dispensed.
   e. **Flip-Flop Inputs:** `J1=X, K1=0, J2=1, K2=1`
5. **Coin2 State to Coin3 State:**
   a. **Current State:** `10 (Coin2)`

b. **Input:** Coin inserted.
c. **Next State:** `11 (Coin3)` - Transitions to the three-coin state.
d. **Output:** `FA=0, FB=0, FC=0, C=1` - Indicates readiness to dispense a product.
e. **Flip-Flop Inputs:** `J1=1, K1=0, J2=0, K2=1`

6. **Coin3 State to Idle State:**
   a. **Current State:** `11 (Coin3)`
   b. **Input:** None (end of transaction).
   c. **Next State:** `00 (Idle)` - Resets back to the idle state.
   d. **Output:** `FA=1, FB=1, FC=1` - Signals readiness for a new transaction.
   e. **Flip-Flop Inputs:** `J1=X, K1=1, J2=1, K2=1`

Circuit Diagram for the Vending Machine FSM



This circuit diagram illustrates the implementation of the vending machine's finite state machine (FSM) using digital logic components. The design is based on the Moore Machine concept, with the circuit controlling state transitions and outputs. Below is a detailed explanation of the circuit's components and functionality.

1. Inputs:

- **TURNON:** Power input to activate the vending machine circuit.
- **F (Coin Input):** Indicates that a coin has been inserted into the machine.
- **S (Selection Input):** Represents the user's selection to dispense a product.

2. Flip-Flops:

- The circuit uses **J-K Flip-Flops** to store the current state of the FSM:
  - **Q0, Q1, Q2:** Represent the binary states of the FSM.
  - **J** and **K Inputs:** Control the set and reset behavior of each flip-flop, derived from the Boolean equations.

3. Logic Gates:

- **AND Gates:** Combine inputs and state variables to produce specific conditions for state transitions and outputs.
- **OR Gates:** Merge multiple conditions to simplify the logic.
- **NOT Gates:** Invert signals where necessary, such as in the Boolean expressions for state transitions.

4. Outputs:

- **FA, FB, FC (Indicators):** Show the status of the vending machine (e.g., readiness or reset).
- **AO, BO (Auxiliary Outputs):** Indicate the number of coins inserted into the machine.
- **C (Dispense Signal):** Activates when the machine is ready to dispense a product.

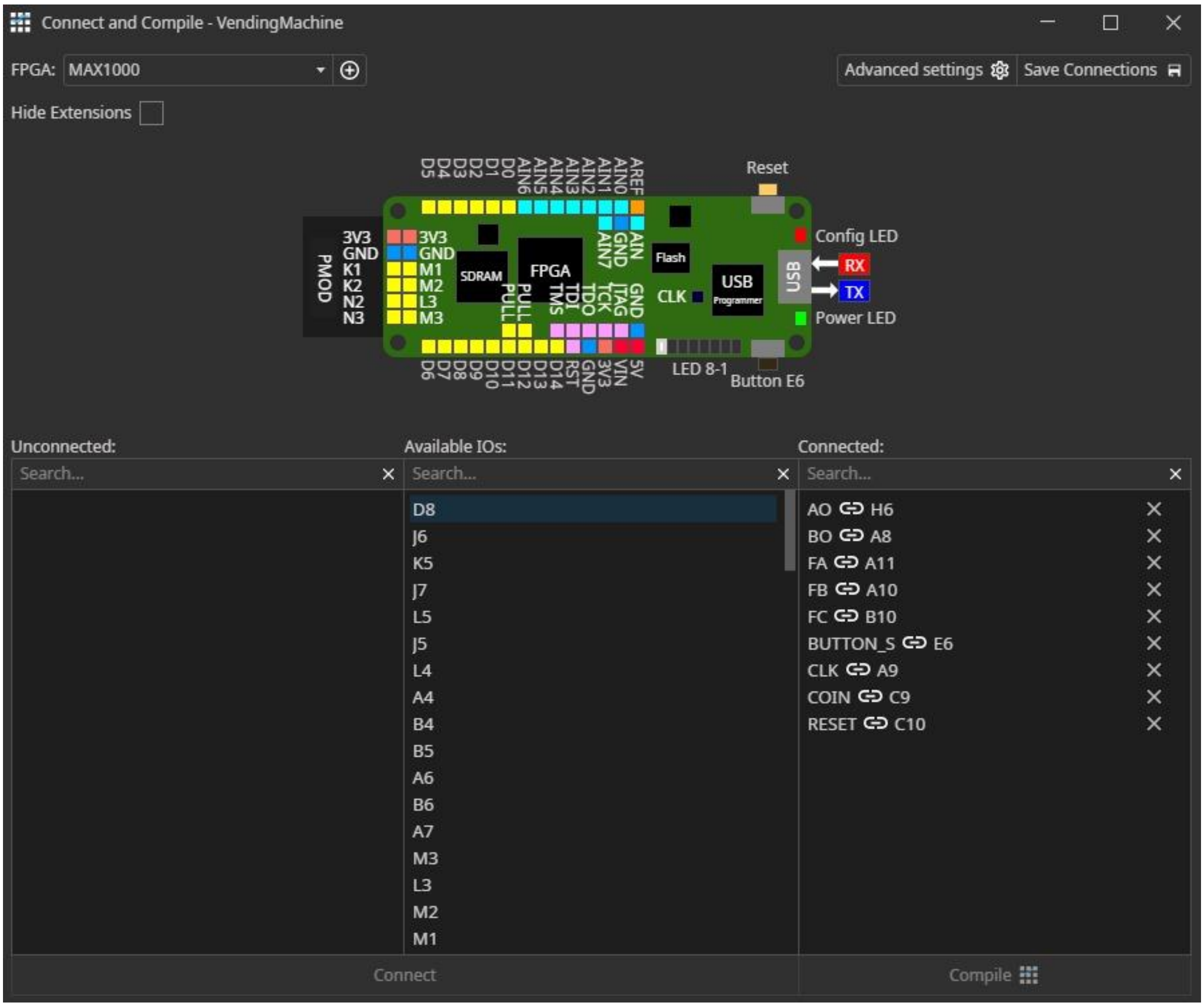
5. Working Mechanism:

1. **State Transitions:**
   a. Flip-flops transition between states based on the logic derived from the Boolean equations.
   b. Example:
      i. If Q0 is active and a coin is inserted (F), the circuit transitions to the next state, activating Q1.
2. **Output Control:**
   a. Outputs are activated based on the current state of the FSM.
   b. For instance:

i. FA is active in the idle state (Q0 active, Q1 and Q2 inactive).
ii. C is activated in the dispensing state when Q0, Q1, and Q2 are all active.

3. **Reset Mechanism:**
   a. After dispensing a product, the circuit resets to the idle state (Q0 active, Q1 and Q2 inactive).

FPGA Pin Configuration for the Vending Machine FSM



This image displays the pin configuration and mapping for implementing the vending machine FSM on the FPGA (Field-Programmable Gate Array). The FPGA used in this project is the **MAX1000** development board, and the connections are as follows:

1. FPGA Overview:

- The **MAX1000** FPGA board is utilized to implement the vending machine's finite state machine. This board provides configurable input and output pins (I/Os) that interface with external components such as buttons, LEDs, and sensors.

2. Inputs and Outputs Mapping:

The table below summarizes the I/O assignments:

Signal	FPGA Pin	Description
AO	H6	Auxiliary output indicating one coin inserted.
BO	A8	Auxiliary output indicating two coins inserted.
FA	A11	Status indicator for readiness or reset.
FB	A10	Status indicator for readiness or reset.
FC	B10	Status indicator for readiness or reset.
BUTTON_S	E6	Selection button input.
CLK	A9	Clock input for synchronous operation.
COIN	C9	Coin detection input signal.
RESET	C10	Reset input to bring the system back to the idle state.

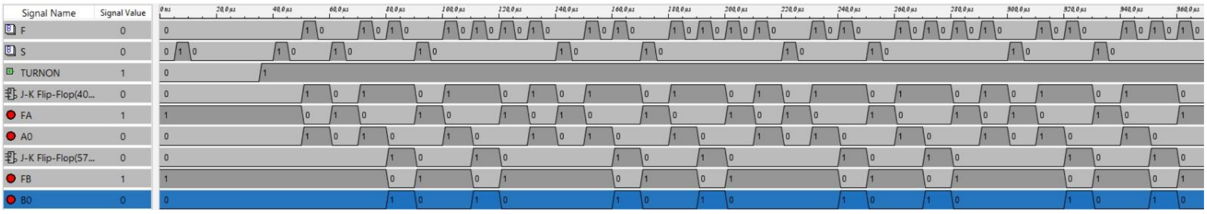
3. Key Features of the Configuration:

1. **Inputs:**
   a. **BUTTON_S, COIN, RESET:** These inputs are connected to buttons or sensors to handle user actions such as coin insertion, product selection, and resetting the system.
   b. **CLK:** Ensures the FSM operates synchronously.
2. **Outputs:**
   a. **AO, BO:** Display the coin count status (e.g., one coin or two coins inserted).
   b. **FA, FB, FC:** Indicators for the machine's readiness or reset status.
3. **Pin Mapping:**
   a. Each signal is mapped to specific FPGA pins, as listed in the table. These mappings were selected to ensure clear and efficient connections between the FPGA and external components.

4. Compilation and Simulation:

- The "Connect and Compile" interface on the FPGA tool ensures that all pins are properly assigned and the design is ready for synthesis.
- After connecting all signals, the design can be compiled and uploaded to the FPGA for testing.

Simulation Waveform for the Vending Machine FSM



This simulation waveform demonstrates the operation of the vending machine's finite state machine (FSM). It visually represents the relationship between the inputs, outputs, and state transitions over time. Below is a detailed explanation of the key signals and their interactions:

1. Input Signals:

- **F (Coin Input):**
  - Indicates the insertion of a coin. When the value toggles to `1`, the FSM progresses to the next state.
- **S (Selection Input):**
  - Represents the selection made by the user. When toggled to `1`, it triggers the dispensing process.
- **TURNON:**
  - Activates the vending machine system.

2. State Control Signals:

- **J-K Flip-Flop (Q0 and Q1):**
  - These signals represent the FSM's state storage:
    - **Q0:** Indicates the lower-order bit of the current state.
    - **Q1:** Indicates the higher-order bit of the current state.
  - Their transitions are controlled by the input signals (`F`, `S`) and determine the next state.

3. Output Signals:

- **FA, FB, FC:**
  - Indicate the status of the vending machine:
    - **FA:** Signals readiness or reset state.
    - **FB:** Indicates intermediate states (e.g., one coin detected).
    - **FC:** Indicates completion states (e.g., all coins inserted).
- **AO, BO:**
  - Auxiliary outputs used to indicate the number of coins inserted:
    - **AO:** Active after one coin is detected.
    - **BO:** Active after two coins are detected.

4. Time-Based Analysis:

- **0 to 80 µs:**
  - **TURNON** is activated.
  - **F toggles to 1**, indicating a coin insertion. **AO** becomes active, and the system transitions to the `Coin1` state.
- **80 to 160 µs:**
  - Another **F toggle to 1** occurs, indicating a second coin insertion. **BO** becomes active, and the system transitions to the `Coin2` state.
- **160 to 240 µs:**
  - The user toggles **S to 1**, selecting the product. The system transitions to the dispensing state, and the corresponding output signal is activated.
- **240 µs and Beyond:**
  - The system resets, returning to the `Idle` state (**FA=1, FB=1, FC=1**).

5. Key Observations:

- State transitions align precisely with the toggling of **F** and **S** signals.
- Outputs **AO, BO** accurately reflect the number of coins inserted at each stage.
- The FSM resets correctly after the dispensing process.