Innovation & Entrepreneurship Hub for Educated Rural Youth (SURE Trust – IERY)

Vending Machine Controller Using Verilog HDL

The domain of the Project: RTL Design

COURSE NAME: Integrated VLSI

Team Mentor :

Mr. Satish Devarapalli (Emulation Verification Engineer , Apple)

Team Members:

1. Ms.Marthala Haripriya --- B.Tech, 4 th year pursuing -------------- Team Leader
2. Ms.Pera Mamatha ---------- B.Tech, 4 th year pursuing ------- Team Member
3. Ms.Buti Usha ---------- B.Tech, 4 th year pursuing ------- Team Member

Period of the project

October 2025 to November 2025

Declaration

The project titled "**Vending Machine Controller Using Verilog HDL**" has been mentored by **Satish Sir** , organised by SURE Trust, from October 2025 to November 2025, for the benefit of the educated unemployed rural youth for gaining hands-on experience in working on industry relevant projects that would take them closer to the prospective employer. I declare that to the best of my knowledge the members of the team mentioned below, have worked on it successfully and enhanced their practical knowledge in the domain.

Team Members:	Signatures:
1. Ms.M.HariPriya	M.Hari priya
2. Ms.P.Mamatha	P.Mamatha
3. Ms.B.Usha	B.usha

Mentor's Name : Mr. Satish Devarapalli
Designation— Emulation Verification Engineer , Apple

Prof. Radhakumari
Executive Director & Founder
SURE Trust

Table of Contents

Executive Summary

This project involves the design and implementation of a **Vending Machine Controller** using **Verilog HDL**. The objective of the project is to model the core functionality of a vending machine using digital hardware logic, focusing on item selection, currency handling, configuration, and dispensing control.

The design follows a **modular RTL approach**, where individual Verilog modules are developed for item selection, currency accumulation, control logic using a Finite State Machine (FSM), configuration through an APB interface, pulse synchronization, and output processing. A top-level module integrates all these submodules to form a complete vending machine controller.

The system operates in two main modes. In **configuration mode**, item price and availability are programmed using an APB-based interface. In **normal operation mode**, the controller processes user inputs such as item selection and currency insertion and performs dispensing when valid conditions are met.

The design is fully hardware-based and does not rely on any software processor. Functional verification is carried out using a SystemVerilog testbench with directed test cases. Simulation waveforms and synthesis results confirm correct functionality of the design. This project provides practical exposure to RTL design, FSM implementation, clock domain synchronization, and verification techniques.

Introduction

- **Background and context of the project**

Vending machines are automated systems used to dispense products such as snacks, beverages, or tickets based on user selection and payment. These machines are commonly deployed in places such as offices, railway stations, airports, and educational institutions. Most traditional vending machines use microcontroller-based firmware, which limits flexibility and makes scaling and modification difficult.

With increasing demand for configurable and reliable digital systems, hardware-based implementations using **Hardware Description Languages (HDLs)** are becoming more relevant. Verilog HDL allows designers to implement control logic directly in hardware, providing predictable timing, parallel execution, and better control over system behavior.

This project focuses on developing a **hardware-based Vending Machine Controller** using Verilog HDL. The design supports multiple items, configurable prices and quantities, and controlled dispensing. The controller is built using modular RTL blocks and FSM-based control logic, making it easier to understand, modify, and extend.

The project demonstrates how real-world control systems can be modeled using digital design principles without relying on embedded software or external processors.

- **Problem statement or goals of the project**

Modern vending machines often face challenges related to real-time responsiveness, configurability, and efficient hardware utilization. Many existing solutions rely on microcontroller-based software, which can introduce execution delays, reduce timing predictability, and limit scalability when deployed in hardware-oriented platforms such as FPGAs.

In applications where deterministic performance and precise control are required, software-based systems may not be sufficient. Additionally, frequent changes in item pricing and availability demand a flexible configuration mechanism that does not interrupt normal operation.

Therefore, there is a need for a **hardware-based vending machine controller** that provides modular design, fast and deterministic operation, and configurable item management without relying on external processors or software control. A dual-mode system supporting both configuration and normal operation is essential to simplify deployment and maintenance in real-world environments.

Key Functional Requirements

1. **Clock Domains**
   System clock used for normal vending operation.
   Configuration clock used for APB-based configuration.

2. **Modes of Operation**
   Reset Mode: Initializes all internal registers and control signals.
   Configuration Mode: Loads item price and available item count using an APB interface.
   Operation Mode: Accepts item selection and currency input, processes transactions, and controls dispensing and change return.

3. **Currency Input Handling**
   Supports fixed currency denominations: 5, 10, 15, 20, 50, and 100.
   Currency input is asynchronous to the system clock and synchronized internally.

4. **Interfaces**
   APB-based configuration interface for programming item parameters.

Scope of the Project

The scope of this project is to design and implement a **modular vending machine controller using Verilog HDL**, suitable for FPGA or ASIC-based digital systems. The focus is on real-time hardware operation, configurability, and efficient RTL design without dependence on software processors.

1. Technical Scope

- Implemented entirely using Verilog HDL.
- Supports up to 1024 items through configurable addressing.
- Handles item selection, currency accumulation, and dispensing using hardware logic.
- Includes an FSM-based controller to manage system flow.
- Uses an APB-based configuration interface to program item price and availability.
- Uses pulse synchronization logic to safely handle asynchronous inputs.
- Verified through simulation using a SystemVerilog testbench

2. Functional Scope

- **User Operation Mode:**
  Users can select an item, insert currency, and receive the selected item along with calculated change.
- **Configuration Mode:**
  Item price and availability can be configured through the APB interface.
- **Validation Logic:**
  Ensures sufficient currency is inserted and item availability exists before dispensing.
- **Change Calculation:**
  Calculates and returns remaining balance after dispensing.
- **Inventory Tracking:**
  Updates available item count after each successful dispense.

3. Design Scope

- Fully synchronous RTL design
- FSM-based control for sequencing operations.
- One controlled clock domain crossing using pulse synchronizers.
- No mechanical components such as motors or sensors.
- Focused purely on digital control logic.

4. Development Scope

The following Verilog modules were developed as part of this project:

- pulse_sync – for synchronizing asynchronous pulses.
- item_selection – for capturing item ID and quantity.
- currency_input – for accumulating inserted currency.
- main_controller – FSM-based control logic.
- config_module – APB-based configuration logic.
- output_info – dispensing and change calculation.
- vending_machine_top – top-level integration module.

A complete SystemVerilog testbench was developed to verify functionality under different scenarios.

- Single item dispense per transaction.

Limitations

1. **Hardware Implementation Limitation**
 The design has been verified only through simulation and synthesis and has not been implemented on physical FPGA or ASIC hardware.

2. **Currency Handling Constraints**
 The system supports only predefined currency denominations and does not include physical note or coin validation.

3. **Fixed Denominations**
 Only specific denominations (5, 10, 15, 20, 50, 100) are supported. Any change requires RTL modification.

4. **Single Item per Transaction**
 The controller supports dispensing only one item per transaction.

5. **Limited User Interaction Features**
 Features such as transaction cancellation, timeout handling, or digital payment methods are not implemented.

6. **No Security or Fault Detection**
 The design does not include tamper detection, fault handling, or error recovery mechanisms.

7. **Asynchronous Input Sensitivity**
 Asynchronous inputs require careful synchronization and may be sensitive to metastability if not handled correctly.

8. **No External Monitoring or Logging**
   The system does not support external transaction logging or real-time monitoring interfaces.

Innovation

1. **FSM-Based Hardware Control**
   The project uses an FSM to control the complete vending sequence, ensuring deterministic and predictable behavior without software intervention.

2. **Pulse-Synchronized Input Handling**
   Asynchronous user inputs such as item selection and currency insertion are safely synchronized using dedicated pulse synchronization logic.

3. **Configurable Hardware Design**
   Item price and availability can be updated dynamically through an APB interface without resetting the system.

4. **Automatic Change Calculation**
   The system calculates and returns excess currency automatically when the inserted amount exceeds the item price.

5. **Modular RTL Architecture**
   The design is divided into well-defined, reusable modules, making it easy to understand, test, and extend.

Project Objectives

1. Develop a Configurable Vending Machine Controller Using Verilog HDL

Objective:
To design and implement a modular vending machine controller using Verilog HDL that manages core vending operations such as item selection, currency input, configuration, and dispensing control.

Expected Outcome:
A synthesizable and reusable Verilog-based controller that supports multiple items, configurable prices and availability, and can be integrated into FPGA-based digital systems.

2. Implement Dual-Mode Operation (Configuration Mode and Operation Mode)

Objective:
To enable the controller to operate in two distinct modes: configuration mode for programming item data and operation mode for executing vending transactions.

Expected Outcome:
A stable and controlled mode-switching mechanism that allows item price and availability to be configured through an APB interface without disturbing normal vending operation.

3. Design Currency Accumulation and Validation Logic

Objective:
To design a currency handling module that accepts user-inserted currency inputs, accumulates the total amount, and validates whether sufficient funds are available for the selected item.

Expected Outcome:
A reliable currency input system that supports incremental currency insertion, verifies sufficient payment, and enables correct dispensing and change calculation.

4. Implement FSM-Based Control for Vending Operations

Objective:
To design a finite state machine (FSM) that controls the vending process, including item selection, currency validation, dispensing, and system reset.

Expected Outcome:
A predictable and deterministic FSM-driven controller that ensures correct sequencing of vending operations and smooth transaction flow.

5. Support Configurable Item Data Through APB Interface

Objective:
To provide an APB-based configuration interface for programming item price and available quantity from an external configuration source.

Expected Outcome:
A configuration mechanism that allows administrators to update item data dynamically using standard APB signals, verified through simulation.

6. Handle Asynchronous Inputs Using Synchronization Logic

Objective:
To safely handle asynchronous inputs such as item selection and currency insertion that are not aligned with the system clock.

Expected Outcome:
A reliable synchronization mechanism using pulse synchronizer logic that prevents metastability and ensures safe operation across clock boundaries.

7. Develop a Comprehensive Testbench for Verification

Objective:
To verify the functional correctness of the vending machine controller using a SystemVerilog testbench with directed test cases.

Expected Outcome:
A working testbench that validates item selection, currency handling, FSM transitions, dispensing behavior, and change calculation through simulation waveforms and console outputs.

8. Deliver a Synthesizable and Modular RTL Design

Objective:
To ensure that the complete design is fully synthesizable, modular, and suitable for hardware implementation.

Expected Outcome:
A clean RTL design that passes synthesis checks, follows good digital design practices, and is suitable for future FPGA implementation or academic extensions.

Methodology and Results

- **Methods/Technology used**

1. RTL Design Using Verilog HDL

The vending machine controller was designed using **Verilog HDL**, targeting synthesizable digital hardware. The RTL design follows a modular and parameterized approach, allowing scalability and clarity. Parameters such as item address width, currency width, and total amount width enable support for multiple items and flexible configuration.

2. Modular Architecture

The design is divided into independent functional modules, each responsible for a specific task. These include item selection, currency handling, FSM-based control, configuration handling, pulse synchronization, and output processing. This modular structure simplifies debugging, verification, and future enhancement.

3. FSM-Based Control Logic

A **Finite State Machine (FSM)** implemented in the `main_controller` module governs the overall operation of the vending machine. The FSM controls transitions between idle, selection, currency accumulation, and dispensing states. This ensures deterministic sequencing and predictable behavior for every transaction.

4. APB-Based Configuration Interface

An **AMBA APB interface** is implemented in the `config_module` to configure item price and available quantity. Configuration is performed when `cfg_mode` is active. This interface allows external configuration logic to program item data without interfering with normal vending operation.

5. Clock Domain Crossing Handling

The design operates with two clocks: the system clock (`clk`) and the configuration clock (`pclk`). To safely handle asynchronous inputs such as item selection and currency insertion, **pulse synchronizer modules (`pulse_sync`)** are used. These synchronizers convert asynchronous pulses into clean, single-cycle pulses in the system clock domain, preventing metastability.

6. Currency Input and Accumulation

The `currency_input` module accumulates currency values inserted by the user. Each valid currency pulse is synchronized and added to the running total. The accumulated amount is continuously compared against the selected item cost to determine whether sufficient funds are available for dispensing.

7. Parameterization for Scalability

Key parameters such as item address width, currency width, and total amount width are defined to allow scalability. This enables the controller to support a large number of items and varying currency ranges without modifying core logic.

8. Simulation-Based Verification

A comprehensive **SystemVerilog testbench** was developed to verify the design. The testbench simulates item selection, currency insertion, APB configuration, valid and invalid transactions, and reset conditions. Functional correctness was verified using waveform analysis and console outputs.

- **Tools/Software used**
  Verilog HDL – RTL design and implementation
  SystemVerilog – Testbench development
  Simulation Tool (TCL Console) – Functional verification
  Synthesis Tool – Logic synthesis and design validation

- **Project Architecture**

Project Architecture

The vending machine controller is implemented as a digital RTL IP core that manages item selection, currency handling, configuration, and dispensing operations. Control and data flow are managed entirely in hardware using FSM logic and synchronized interfaces.

1. System Clock and Reset Initialization

The design operates using a system clock for vending operation and a separate clock for APB configuration. An active-low reset initializes all registers, FSM states, and control signals to known values.

2.High-Level Architecture

S.No	Mode	Mode determination	Description
1	Reset Mode	rstn == 0	The rstn pin is low, and other pins are don't care
2	Configure Mode	rstn == 1’b1 && Cfg_mode == 1’b1	Design is out of reset, and cfg_mode is set
3	Operation Mode	rstn == 1’b1 && Cfg_mode == 1’b0	Design is out of reset, and cfg_mode is not set

3. Configuration Mode via APB Interface

In configuration mode, item price and availability are configured through the APB interface using signals such as `psel`, `pwrite`, `paddr`, `pwdata`, and `prdata`. Configuration data is stored in internal registers and used during normal vending operation.

4. Item Selection Handling

The `item_selection` module captures the selected item ID and quantity when a valid selection pulse is detected. The selection is synchronized to the system clock and validated before being processed further.

5. Currency Input and Accumulation

Currency inputs are provided asynchronously using `currency_value` and `currency_valid`. These signals are synchronized using pulse synchronizers and accumulated in the `currency_input` module until the required amount is reached.

6. FSM-Controlled Vending Logic

During operation mode, the FSM:

- Waits for valid item selection
- Accumulates currency
- Checks item availability and sufficient funds
- Enables dispensing when conditions are met

7. Output Signal Generation

The `output_info` module generates vending outputs:

- `dispense_valid` indicates successful dispensing
- `item_dispensed` provides the dispensed item ID
- `no_items_dispensed` shows quantity
- `change_dispensed` returns remaining balance

8. Inventory Tracking

After successful dispensing, the available item count is updated through configuration logic. The APB interface allows reading updated values for monitoring purposes.

Module Responsibilities

Module Name	**Functionality**
pulse_sync	Synchronizes asynchronous input pulses such as currency insertion and item selection to the system clock, ensuring safe clock domain crossing and preventing metastability.
item_selection	Captures the user's selected item ID and quantity when a valid selection pulse is detected and generates a `selection_done` signal.
currency_input	Accumulates synchronized currency inputs, updates the total inserted amount, and generates a `currency_done` signal when valid currency is detected.
main_controller	Acts as the central FSM that controls system flow, manages transitions between idle, selection, and currency states, and enables dispensing through `dispense_enable`.
config_module	Interfaces with the host system using the APB protocol to configure item price and available quantity during configuration mode.
output_info	Validates sufficient funds and item availability, generates `dispense_valid`, determines the dispensed item and quantity, and calculates the change to be returned.
vending_machine_top	Integrates all submodules into a single top-level RTL design, handling signal routing, synchronization, and complete vending machine operation.

Registers Used

I. item_selected

This register stores the item ID selected by the user. It is captured in the `item_selection` module when a valid selection pulse is detected and is used during the vending operation.

II. no_items_selected

This register stores the number of items selected by the user. It is used to calculate the total cost and control dispensing.

III. selection_done

This register flag indicates that a valid item selection has been completed. It becomes high for one clock cycle when item selection is successful.

IV. total_amount

This register stores the cumulative value of all currency inserted by the user so far. It is continuously updated in the `currency_input` module and compared against the item cost.

V. currency_done

This register indicates that a valid currency pulse has been received and processed. It is asserted for one clock cycle when valid currency is detected.

VI. item_price

This register holds the configured price of the selected item. The value is provided by the `config_module` during operation mode.

VII. avail_count

This register stores the available stock count of the selected item. It is used to validate whether dispensing is possible.

VIII. dispense_enable

This control register is generated by the `main_controller` FSM to enable the dispensing operation once all conditions are satisfied.

IX. dispense_valid

This register indicates that the item dispensing operation is valid and successful. It is asserted for one clock cycle during a successful transaction.

X. item_dispensed

This register stores the item ID that is dispensed. It is used to drive the output interface.

XI. no_items_dispensed

This register stores the number of items dispensed during a transaction**.**

XII. change_dispensed

This register stores the remaining currency value after deducting the item cost from the total inserted amount. This value represents the change returned to the user.

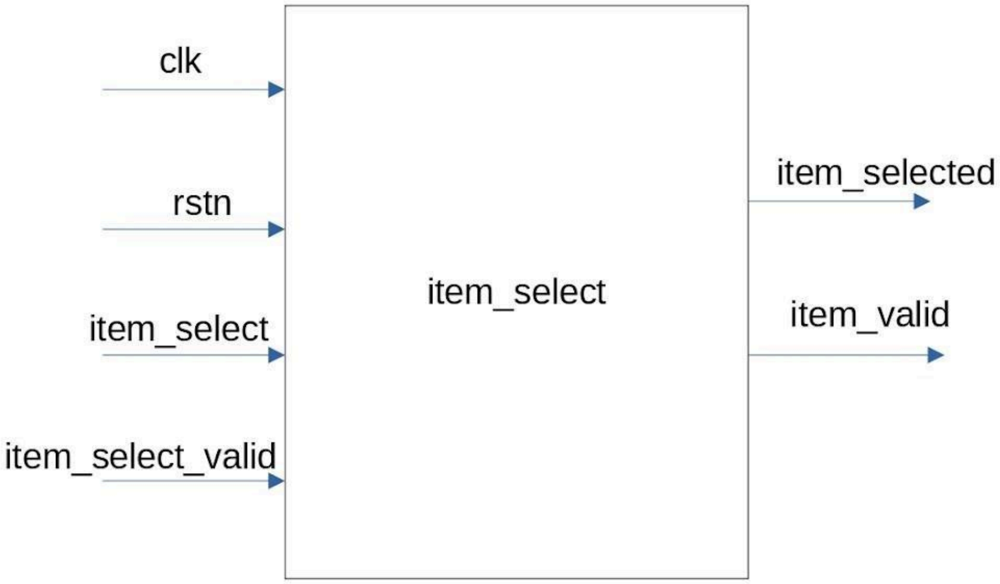
XIII. select_valid_pulse

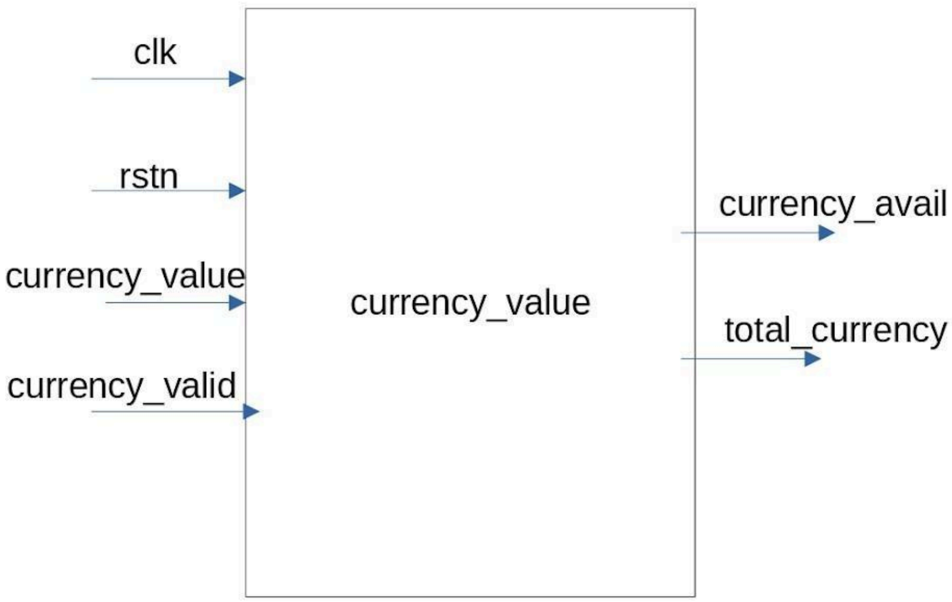
This internal register holds the synchronized item selection pulse generated by the `pulse_sync` module

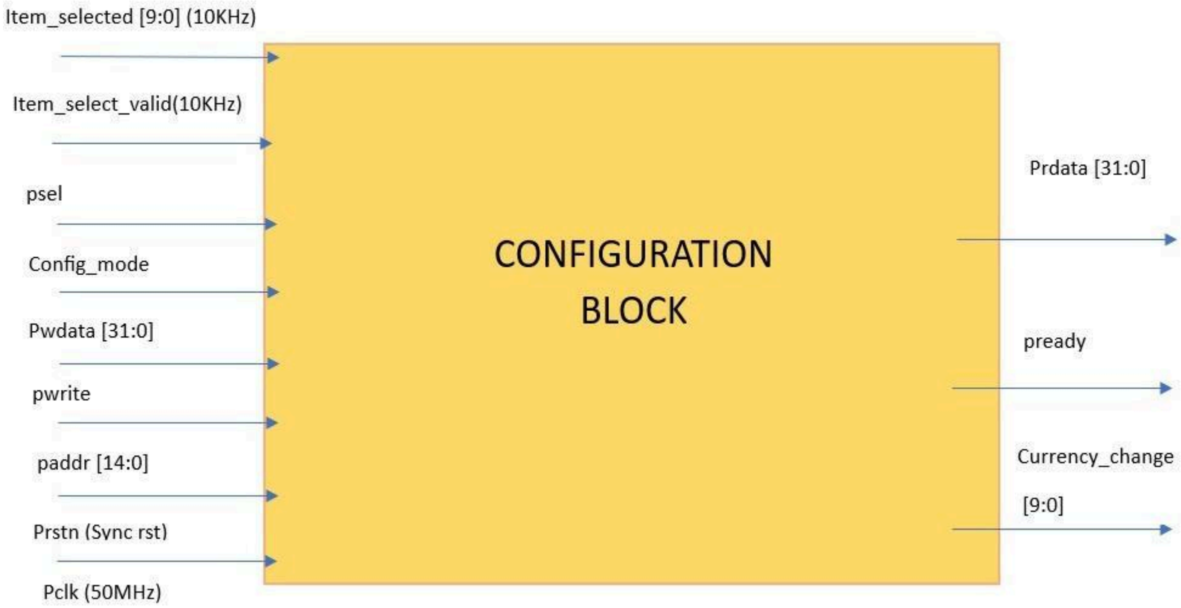
XIV. currency_valid_pulse

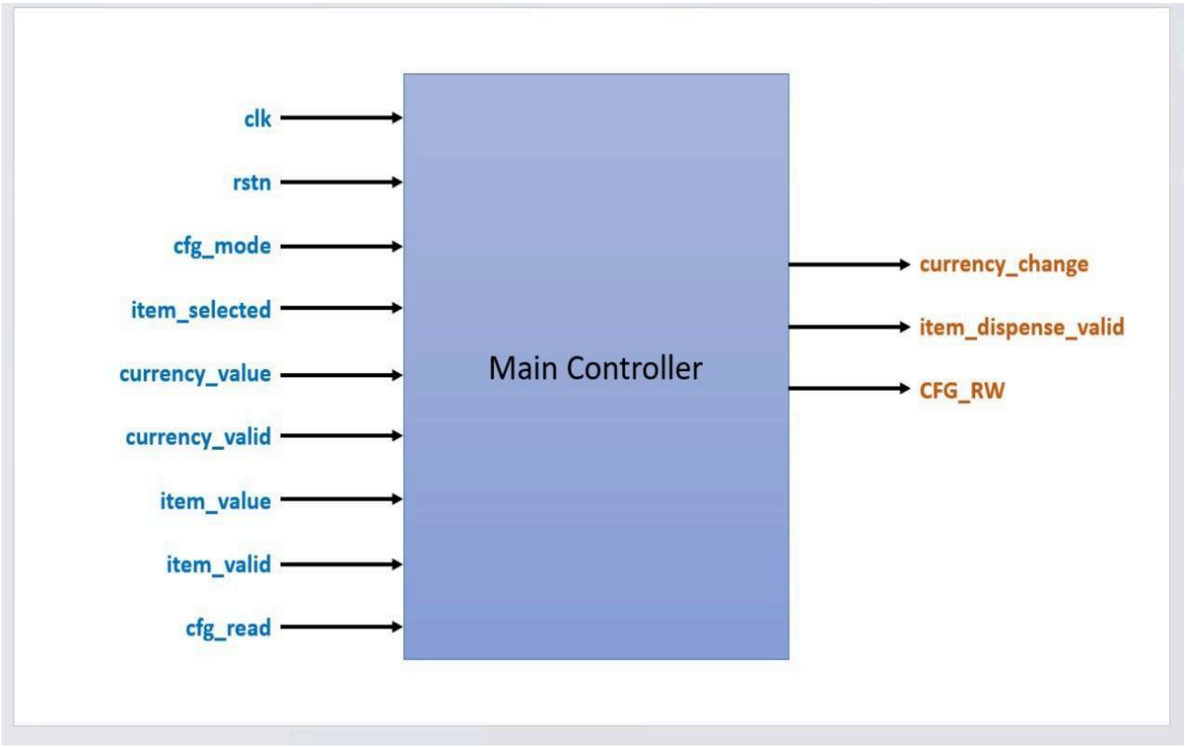
This internal register holds the synchronized currency input pulse generated by the `pulse_sync` module.

Input_block

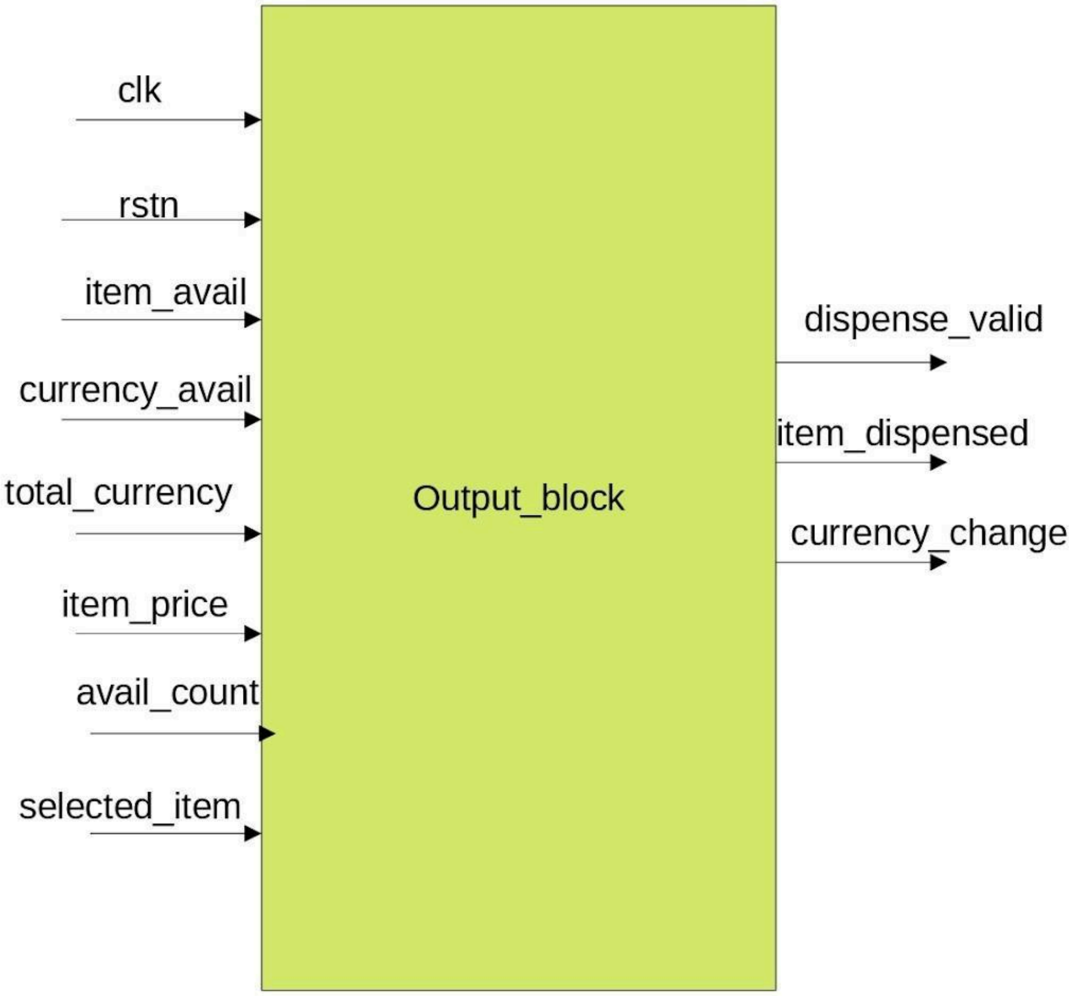

clk
rstn
item_select
item_select_valid
item_select
item_selected
item_valid


clk
rstn
currency_value
currency_valid
currency_value
currency_avail
total_currency


Item_selected [9:0] (10KHz)
Item_select_valid(10KHz)
psel
Config_mode
Pwdata [31:0]
pwrite
paddr [14:0]
Prstn (Sync rst)
Pclk (50MHz)
CONFIGURATION BLOCK
Prdata [31:0]
pready
Currency_change [9:0]


clk
rstn
cfg_mode
item_selected
currency_value
currency_valid
item_value
item_valid
cfg_read
Main Controller
currency_change
item_dispense_valid
CFG_RW

Output_block

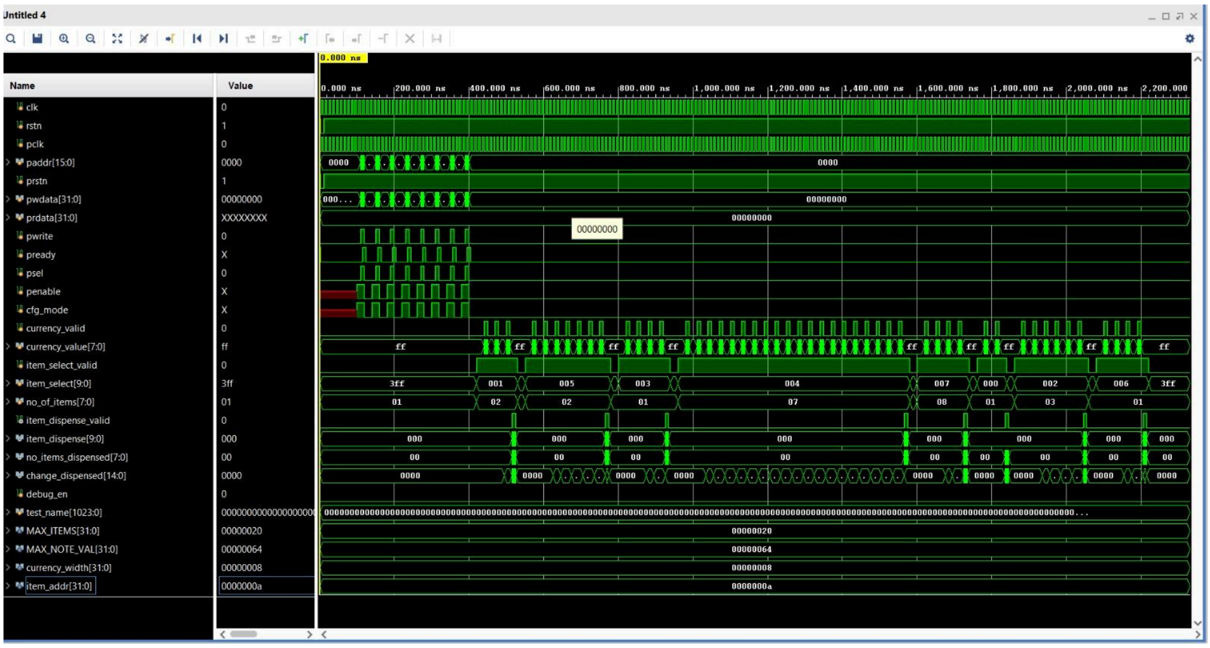


- **Results**

  The design was verified using a SystemVerilog testbench. Simulation waveforms and TCL console outputs confirm correct operation of item selection, currency accumulation, FSM transitions, item dispensing, and change calculation.

  The synthesis report validates the logical correctness and efficient implementation of the design.

- **Final project working screenshots along with supporting explanation**



The above waveform represents the functional verification of the Vending Machine Controller using a SystemVerilog testbench. It demonstrates correct interaction between item selection, currency input, FSM control, configuration, and output logic.

Clock and Reset Operation

- clk and pclk show continuous clock operation for the main logic and APB interface.
- rstn and prstn are asserted low initially, ensuring all modules start from a known reset state.
- After reset de-assertion, normal operation begins.

Configuration Phase (APB Interface)

- During the initial part of the simulation, the configuration interface signals become active.
- This phase is used to configure item price and available item count through the configuration interface.
- While configuration is active, normal vending operation is disabled to prevent unintended dispensing.

Item Selection Operation

- Item selection pulses indicate the user selecting an item.
- Along with the selection pulse, the selected item number and the required quantity are provided.

- These values are captured and stored by the item selection module.

Currency Input Operation

- Currency input pulses represent the insertion of currency by the user.
- Multiple pulses are observed, showing that the total required amount is inserted in steps.
- The currency input module accumulates the inserted value internally.

FSM Controlled Dispensing

- Once item selection is completed and sufficient currency is available, the controller allows dispensing.
- The dispense valid signal is asserted, indicating a successful transaction.
- The dispensed item number and quantity are shown, and the correct change amount is calculated.

Transaction Completion

- After dispensing, the internal signals are cleared and the system returns to the idle state.
- This shows that the design is ready for the next transaction.

Observation

- The waveform confirms correct FSM transitions, accurate currency accumulation, valid item dispensing, and correct change calculation.
- The design operates as expected for multiple transactions.

Project GitHub Link

https://github.com/sure-trust/HARI-PRIYA-MARTHALA-g2-25-integrated-vlsi-design/tree/main/Final%20capstone%20project

Learning and Reflection

Completing our **Vending Machine Controller project using Verilog HDL** was a valuable learning experience for our entire team. Through the phases of **designing, coding, simulation, debugging, and integration**, we gained practical exposure to real-world digital system development.

This project strengthened our understanding of **RTL design, Finite State Machines (FSMs), and clock domain synchronization**. This project helped us understand how theoretical concepts are applied practically in hardware-based designs and improved our confidence in working with complex digital systems.Working with asynchronous inputs and verifying the design through simulation helped us improve our debugging skills and confidence in writing **synthesizable Verilog code**.Beyond technical knowledge, the project enhanced our **teamwork, communication, time management, and documentation skills**. Overall, this experience bridged the gap between theory and practice and prepared us for future challenges in **VLSI.**

Learning :

1.Hari priya (TEAM LEADER):

a) Worked on the **development of the Item Selection Module**, implementing logic to select items based on user input and available stock.

b) Also contributed to the **Pulse synchronizer Module** using Verilog HDL, which included to **safely transfer a short pulse signal from one clock domain to another** without causing errors like metastability.

c) Played a major role in **leading and coordinating the team**, ensuring smooth communication among members, resolving technical issues, and guiding the team toward timely completion of the project. These efforts significantly contributed to the successful execution of the project.

d) Gained practical knowledge in **integrating multiple submodules into a single top-level module**, improving understanding of overall system architecture.

2.Mamatha(Team member)

a) Worked on the **top-level integration** of different modules such as item selection, currency handling, item dispense logic, and control signals to form a complete vending machine controller.

b) Contributed to the implementation of the **configuration module** by supporting item price and availability loading through the APB interface. Assisted in handling read/write operations and ensuring safe data transfer across clock domains.

c) Contributed to the development of **output logic** by generating correct item dispense and currency change signals based on controller decisions. Supported timing control to ensure single-cycle valid output pulses.

d) Gained experience in **testing and verification**, ensuring that all modules worked together correctly and the system operated as expected.

3.Usha (Team Member)

a) Contributed to the implementation of the **currency input logic** by validating **asynchronous currency pulses** and accumulating the **total inserted amount**. Supported proper **synchronization and coordination** with the controller to ensure accurate transaction processing.

b)Contributed to the design of the **main controller FSM**, which manages **state transitions** for item selection, currency validation, and dispensing. Assisted in ensuring **correct control flow and timing** for reliable vending machine operation.

c)Gained practical experience in **connecting different modules into one top-level module**, helping to understand the overall system structure.

LEARNING EXPERIENCE :

1.Hari priya (Team leader)

As a team member and Team Lead, I worked on both the **Item Selection Module** and the **Pulse Synchronizer Module** of the Vending Machine Controller project. In the Item Selection module, I implemented logic to select items based on user input and available stock, which strengthened my understanding of RTL design and control flow. I also contributed to the Pulse Synchronizer module using Verilog HDL to safely transfer short pulse signals across different clock domains, helping me understand metastability and clock domain crossing challenges.

Along with technical responsibilities, I played a key role in **leading and coordinating the team**, ensuring smooth communication, resolving technical issues, and guiding the team to complete the project on time. I also gained practical experience in **integrating multiple submodules into a single top-level module**, which improved my understanding of overall system architecture and module interaction. This project enhanced my technical skills, leadership abilities, and confidence in working on industry-oriented VLSI designs.

2. Mamatha

As a team member, I worked on the **top-level integration** of various modules including item selection, currency handling, item dispense logic, and control signals to build a complete vending machine controller. This responsibility helped me understand how individual RTL modules interact and function together as a unified system.

I also contributed to the **configuration module**, supporting item price and availability loading through the **APB interface**. This involved assisting with read and write operations and ensuring **safe data transfer across clock domains**. In addition, I contributed to the **output logic**, where I helped generate correct item dispense and currency change signals based on controller decisions, with proper timing control to ensure single-cycle valid pulses.Through extensive **testing and verification**, I ensured that all integrated modules worked correctly and that the overall system operated as expected. This experience improved my skills in RTL integration, CDC handling, simulation, debugging, and system-level verification, providing strong practical exposure to real-world digital design workflows.

3. Usha

As a team member, I contributed to the **currency input logic** by validating asynchronous currency pulses and accumulating the total inserted amount. This work involved ensuring proper **signal synchronization** and coordination with the controller to achieve accurate and reliable transaction processing.

I also contributed to the **design of the main controller FSM**, which manages state transitions for item selection, currency validation, and item dispensing. Through this task, I gained a strong understanding of **control flow, timing management, and FSM-based decision making** in digital systems.

In addition, I gained practical experience in **integrating multiple modules into a single top-level design**, which helped me understand the overall system structure and interaction between different modules. This experience strengthened my skills in RTL integration and system-level design.

Conclusion and Future Scope

Objectives

The primary objective of the **Vending Machine Controller project** was to design and implement a **digital IP core** that defines the architecture, interfaces, and operational behavior of a smart vending machine controller. The key objectives of the project are as follows:

1. To develop a **reusable and configurable vending machine controller IP** that can be easily integrated into different vending machine systems with flexible item configurations.

2. To support a wide range of items, **up to 1024 products**, and accept multiple currency denominations including **₹5, ₹10, ₹15, ₹20, ₹50, and ₹100**.

3. To ensure **fast and efficient operation**, with a response time of **less than ten system clock cycles** from the receipt of valid currency to the item dispense decision.

4. To implement **three distinct operating modes**:

    - **Reset Mode** to initialize the system and reset all registers and memories

    - **Configuration Mode** to load item data, prices, and availability using an APB interface

    - **Operation Mode** to handle real-time user actions such as item selection, currency insertion, dispensing, and change return

5. To standardize system interfaces for smooth communication between modules, including:

    - System clock and reset signals

    - Currency and item selection inputs

    - Item dispense and change return outputs

    - APB interface for configuration and monitoring

6. To maintain accurate **inventory tracking**, including the current available count and total dispensed count for each item.

7. To provide a **user-friendly experience** by instantly returning currency for unavailable items and clearly indicating successful or failed transactions.

8. To allow **easy customization through parameterization**, such as defining the total number of supported items and maximum accepted currency value.

Achievements

The **Vending Machine Controller project** was successfully completed and achieved all the defined objectives. The major achievements of the project are listed below:

1. **Reusable and Scalable Design**
   A flexible and reusable digital controller IP was developed, capable of supporting up to **1024 items** and handling currency values up to **₹100**. The parameterized design allows easy adaptation to different vending machine requirements.

2. **Support for Multiple Operating Modes**
   The controller operates reliably in **Reset, Configuration, and Operation modes**, enabling smooth system initialization, vendor configuration through APB, and real-time vending operations.

3. **Reliable Handling of Asynchronous Currency Inputs**
   The system successfully handles currency inputs arriving at slower, human-operated speeds while operating on a high-speed system clock, ensuring correct synchronization and reliable operation.

4. **Accurate Inventory and Transaction Management**
   The design maintains precise tracking of item prices, available stock, and dispensed item counts, ensuring correct transaction processing and inventory updates.

5. **Real-Time Error Detection and Response**
   The controller provides immediate feedback by returning inserted currency when selected items are unavailable and clearly signaling transaction status, improving overall user experience.

6. **Easy Configuration Using APB Interface**
   The implementation of an APB-based configuration interface enables simple loading of item data, resetting of counters, and monitoring of system status.

Conclusion:

The **Vending Machine Controller** is a flexible and well-structured **digital IP core** developed to support modern vending machine applications. The controller efficiently handles real-time transactions by delivering fast and accurate responses, made possible through support for **asynchronous currency inputs** and **customizable item counts and pricing**. This design ensures reliable operation even when user inputs occur at varying speeds, closely reflecting real-world vending scenarios.

The system operates through clearly defined modes—**Reset Mode, Configuration Mode, and Operation Mode**—which help streamline system initialization, vendor configuration, and normal vending operations. These operating modes simplify maintenance and enable seamless integration into larger digital systems. The modular and parameterized architecture further enhances flexibility, allowing the controller to be adapted easily to different vending machine requirements.

An **APB-based configuration interface** is included to allow vendors to load item prices, manage stock availability, and reset counters during configuration mode. This interface improves usability and reduces complexity during system setup and monitoring. Overall, the vending machine controller improves system performance, ensures accurate inventory and transaction management, and provides a strong, scalable platform for future enhancements in **vending machine automation and smart embedded systems**.

Future Scope:

Vending machines are no longer limited to basic product dispensing and are steadily evolving into smart and automated systems. Modern vending solutions aim to improve user convenience while also supporting sustainability and the adoption of emerging technologies. The future of vending machines focuses on intelligent functionality, energy-efficient operation, and enhanced user interaction, enabling more reliable, eco-friendly, and user-centric vending experiences.

1. Secure Digital Transactions

In future implementations, the vending machine controller can be enhanced to support **secure digital transaction mechanisms**. Technologies such as blockchain can be integrated to maintain transparent and tamper-proof transaction records. Each transaction can be securely logged, reducing the risk of fraud and enabling users to verify their purchase history. This improvement increases trust and reliability in automated vending systems.

2. Sustainable and Energy-Efficient Operation

To promote environmental sustainability, vending machines can be designed to operate using **renewable energy sources such as solar power**. Solar-powered vending machines are especially useful in outdoor locations and remote areas with limited access to electricity. This approach helps reduce energy consumption, operational costs, and environmental impact, making vending solutions more eco-friendly.

3. Intelligent Product Recommendation System

The integration of **Artificial Intelligence (AI) and Machine Learning (ML)** can enable vending machines to provide personalized product recommendations. By analyzing customer behavior, purchase history, time of day, and environmental conditions, the system can suggest suitable products to users. This enhancement improves user experience and increases overall efficiency and sales potential.

4. Touchless and User-Friendly Interfaces

Future vending machines can adopt **touchless interaction methods** such as voice commands and gesture recognition. These interfaces improve accessibility for all users and enhance hygiene by minimizing physical contact. Touchless systems are particularly beneficial in public places, hospitals, and post-pandemic environments where safety and cleanliness are critical.

5. Recycling and Reward-Based Sustainability Models

Vending machines can also be extended to support **recycling-based reward systems**. By accepting used packaging such as bottles or cans, the machine can encourage users to participate in recycling initiatives. In return, users can be rewarded with discounts or product credits, promoting sustainable practices while increasing user engagement.