Test Plan

1) Sanity , Reset , Configuration Tests

TC ID	Testcase Name	Description	Input Stimulus	Expected Output
TC_01	Sanity_PowerOn_Flow	Full basic flow: Reset → Config → Load Items → Operation → Buy Item	1) rstn=0 →12) cfg_mode=1, program 2 items:Item0: cost=₹10, stock=5Item1: cost=₹20, stock=33) cfg_mode=04) Select Item0, insert ₹10	Item0 dispensed, no change; stock reduces to 4; counters updated
TC_02	Reset_Default_State	Check reset brings controller to initial state	rstn=0 applied for 5 cycles, cfg_mode don't care	All registers cleared, stock=0, disp_count=0, no outputs active
TC_03	Reset_During_Transaction	Reset during a currency insertion operation	Start buying item (cost ₹20), insert ₹10 then assert rstn=0	Transaction aborted, no dispense, system resets to default

2) Configuration Mode Tests

TC ID	Testcase Name	Description	Input Stimulus	Expected Output
TC_04	Config_Write_Read	Validate write/read of APB registers	Write item0 cost=₹10, stock=10; Read back via APB	Read data matches written values
TC_05	Config_Max_Items	Configure maximum items (e.g., 32)	Write no_of_items=32	System accepts max limit without error
TC_06	Config_Item_Value_Load	Load different cost per item	Program: Item0=₹10, Item1=₹15, Item2=₹50	Each register stores correct value
TC_07	Config_Clear_Dispense_Count	Dispense some items then reset disp_count using APB	Dispense item, then write APB to clear disp_count	disp_count becomes 0

3) Mode Transition Tests

TC ID	Testcase Name	Description	Input Stimulus	Expected Output
TC_08	Mode_Reset_to_Config	Ensure proper transition	rstn=1, cfg_mode=1	Enters configuration mode
TC_09	Mode_Config_to_Operation	Verify switching to operation	cfg_mode goes from 1→0	System enters operation mode and starts accepting currency

4) Item Selection Tests

TC ID	Testcase Name	Description	Input Stimulus	Expected Output
TC_10	Item_Valid_Selection	Validate correct item selection	Item0 configured (₹10, stock=5). Apply `item_select_valid=1, item_select=0`	Item0 is latched, ready for transaction
TC_11	Item_Invalid_Selection	Validate rejecting invalid item	Only 2 items configured. Apply `item_select_valid=1, item_select=5`	No selection made, system ignores input
TC_12	Item_Multiple_Quantity _Select	Select multiple quantities of same item	Item0: ₹10. Apply `item_select=0, no_of_items=3`	Total cost = ₹30 stored internally

5) Currency Handling Tests

TC ID	Testcase Name	Description	Input Stimulus	Expected Output
TC_13	Currency_Valid_Inputs	Verify accumulation of valid currencies	Insert ₹10 then ₹20	Total currency = ₹30 accumulated
TC_14	Currency_Invalid_Denomination	Handle unsupported value	Insert ₹30 (not in supported list)	Ignored; no effect on accumulated amount
TC_15	Currency_Multi_Step_Insert	Insert currency in steps	Item cost=₹20; Insert ₹5 + ₹5 + ₹10	Accumulated to ₹20 and triggers dispense

6) Dispense & Change Logic Tests

TC ID	Testcase Name	Description	Input Stimulus	Expected Output
TC_16	Dispense_Exact_ Amount	Verify dispense with exact currency	Item cost=₹20; Insert ₹20	One item dispensed, change=₹0
TC_17	Dispense_With_Change	Extra money returns change	Item cost=₹15; Insert ₹20	One item dispensed, change=₹5
TC_18	Dispense_Insufficient_ Currency	No dispense until enough currency	Item cost=₹50; Insert only ₹20	No dispense, waiting state
TC_19	Dispense_Multiple_Ite ms	Multi-item exact purchase	Item cost=₹10; Qty=3; Insert ₹30	3 items dispensed, change=₹0

7) Multi-Item and Stock Management Tests

TC ID	Testcase Name	Description	Input Stimulus	Expected Output
TC_20	Multi_Item_Excess_Change	Multi-item with extra money	Item cost=₹10; Qty=2; Insert ₹30	2 items dispensed, change=₹10
TC_21	Multi_Item_Insufficient	Currency insufficient for multi-item	Item cost=₹10; Qty=4; Insert ₹20	No dispense; wait state
TC_22	Multi_Item_Limited_Stock	Stock less than requested quantity	Item cost=₹10; Qty=5; Stock=3; Insert ₹50	Only 3 items dispensed, change for 2 returned, empty_item indication asserted

8) Empty Item Tests

TC ID	Testcase Name	Description	Input Stimulus	Expected Output
TC_23	Empty_Item_Immediate_Refund	Select item with 0 stock	Item0 stock=0; Select Item0; Insert ₹20	No dispense; entire ₹20 returned immediately
TC_24	Empty_Item_Indicator_Check	Special empty code output	Item0 stock=0; select and insert money	`item_dispense` indicates empty item code

9) Counter & Performance Tests

TC ID	Testcase Name	Description	Input Stimulus	Expected Output
TC_25	Counter_Update_Verify	Check counters after transaction	Dispense 2 items	`disp_items +=2, avail_items -=2`
TC_26	Latency_Requirement_Check	Measure latency from last currency input	Insert final currency and count cycles	Dispense must generate output within 10 cycles