ELEC 402

Project 1 Report

Finite State Machine of an ATM;
https://github.com/mchuahua/ELEC402/tree/master/Proj1

Martin Chua - 35713411

9-29-2021

General Description

The Finite State Machine (FSM) is of a generic bank ATM for withdrawing and depositing money. The FSM is intended to be instantiated with parameter settings for CORRECT_PIN, SAVINGS_FUNDS_AMOUNT, and CHEQUING_FUNDS_AMOUNT, which are then stored locally in the instantiation following a cycle with reset high. If left default, the pin will be 1234, savings will contain 1000, and chequing will contain 25. States consist of 4 major sections with a total of 13 states:

- Initial phase (Initial startup, pin validation, and selecting deposit/withdrawals)
- Deposit phase (Account selection, depositing cash vs check, open atm deposit slot)
- Withdrawal phase (Account selection, withdraw amount, checking for insufficient funds, open atm withdrawal slot)
- End phase (Withdrawal of card)

The states are controlled based on inputs to the module and outputs allow for ATM to signal ready, open atm out (deposit), or open atm in (withdrawal). For more information on the states, see below sections.

Disclaimer: as this is a basic bank ATM FSM to demonstrate FSM rather than bank ATM functionality, there may be some differences in the bank ATM FSM function compared to ones in real life. I.e. this bank ATM FSM is not realistic…

IO + FSM Modules + Testbench

Input/outputs definitions and the purpose/description of each state are below. Unless otherwise specified, inputs/outputs are one bit width in size.

Testbench is commented within the code as specified within the project documentation.

FSM Inputs

- clk
  - Basic clock to drive entire FSM module
- rst
  - Basic reset to initialize entire FSM module, and to reset it if anything happens, which resets local chequing and savings values to the instantiated parameters of the module.
- bank_card_insert
  - Signal to indicate that a bank card has been inserted, thus starting the various FSM states
- deposit_withdrawal_selection
  - Signal to select a deposit or withdrawal, indicated by 0 – Withdrawal, 1 – Deposit
- account_selection
  - Signal to select account, either 0 – Chequing, 1 – Savings
- amount[13:0]
  - Bus to indicate amount for withdrawing or depositing.
- pin[13:0]
  - Bus to indicate input pin, there is validation for correct pin vs input pin.

FSM outputs

- open_atm_out
  - Signal to open the ATM deposit out slot for dispensing cash
- open_atm_in
  - Signal to open the ATM withdrawal in slot for receiving cash or check
- ready
  - Signal to indicate that the ATM is ready to be used (and not used by another)

States

The 13 states are initialized using "enum", which allows for ease of reading during test-benching (as seen in the waveforms in the last part). The states are as follows:

- idle
  - (1) Idle state for after reset.
- pin_check
  - (2) Pin check state for validating input pin is correct to local pin.
- select_deposit_withdrawal
  - (3) Deposit/Withdrawal check state for selecting either deposit or withdrawal
- deposit_account_selection

    - (4) If deposit was chosen in (3), this state starts the deposit phase and selects the deposit account
- deposit_cash_or_check
    - (5) Confirmation state that moves to open_atm_in state
- open_atm_in
    - (6) State for ATM deposit slot to be opened.
- withdrawal_account_selection
    - (7) If withdrawal was chosen in (3), this state starts the deposit phase and selects the withdrawal account
- withdrawal_amount_selection
    - (8) Records amount to be withdrawn from the amount input
- withdraw_chequing
    - (9) Confirmation state for chequing (automatically moves to insufficient funds check)
- withdraw_savings
    - (10) Confirmation state for savings (automatically moves to insufficient funds check)
- insufficient_funds_check
    - (11) Checks for insufficient funds in chosen chequing/savings to withdraw from
- open_atm_out
    - (12) State for ATM withdrawal slot to be opened.
- withdraw_card
    - (13) Loop to make sure card is withdrawn (bank_card_insert set to low)

Testbench input outputs

Testbenches do not have input / outputs as testbenches are not supposed to have them. However, it does instantiate the DUT, which in this case is the FSM. Please see code for the explicit instantiation.

Testbench tests

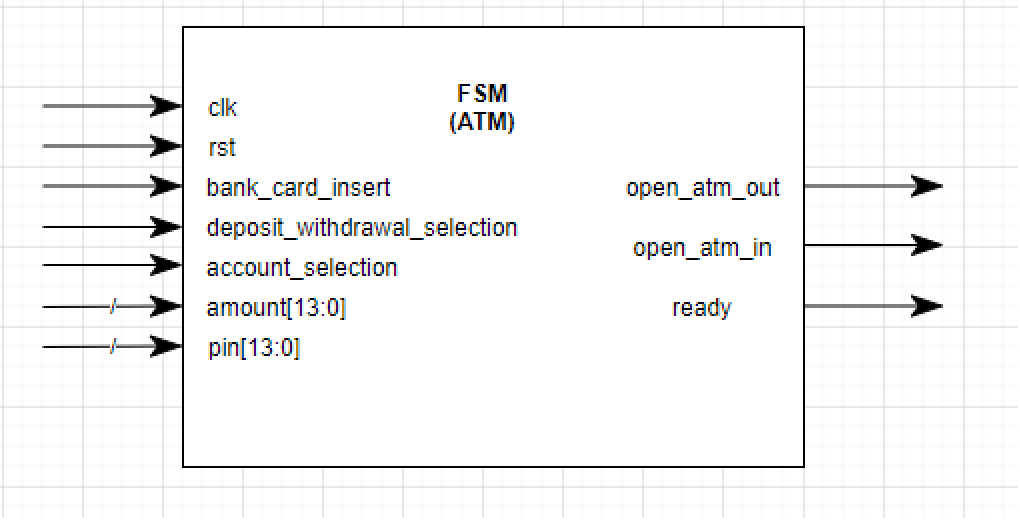
There are various tests implemented in the testbench. The majority of these tests verify state transition and use asserts + error counts to log how many failed. These include:

- Assert ready for when bank card isn't inserted
- Incorrect pin test using a pin that does not match the correct pin
- Correct pin test using the correct pin for state transition
- Withdrawal selection test for selecting withdrawal or deposit
- Deposit selection (and reset) tests for correct resetting and deposit account selection
- Deposit funds amount test for the correct record of funds deposited
- Withdrawal amount test for correct record of funds withdrawn
- Insufficient funds test using an amount greater than what is present in the specified account
- Withdraw card test to make sure card is withdrawn
- Test that state loops back to idle after card is withdrawn

These tests are further elaborated in the code / in the waveforms in the last section.

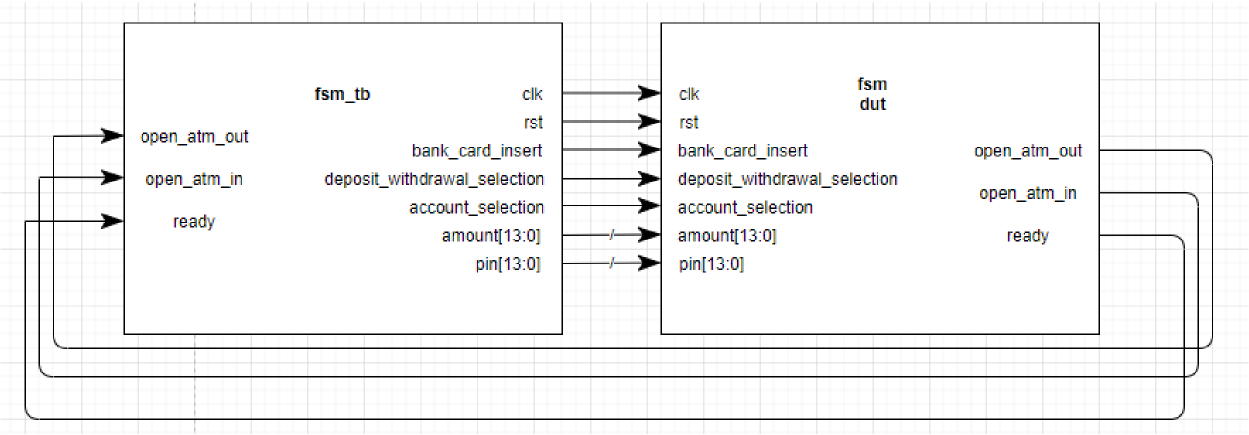
FSM Block Diagram

See .\fsm.drawio



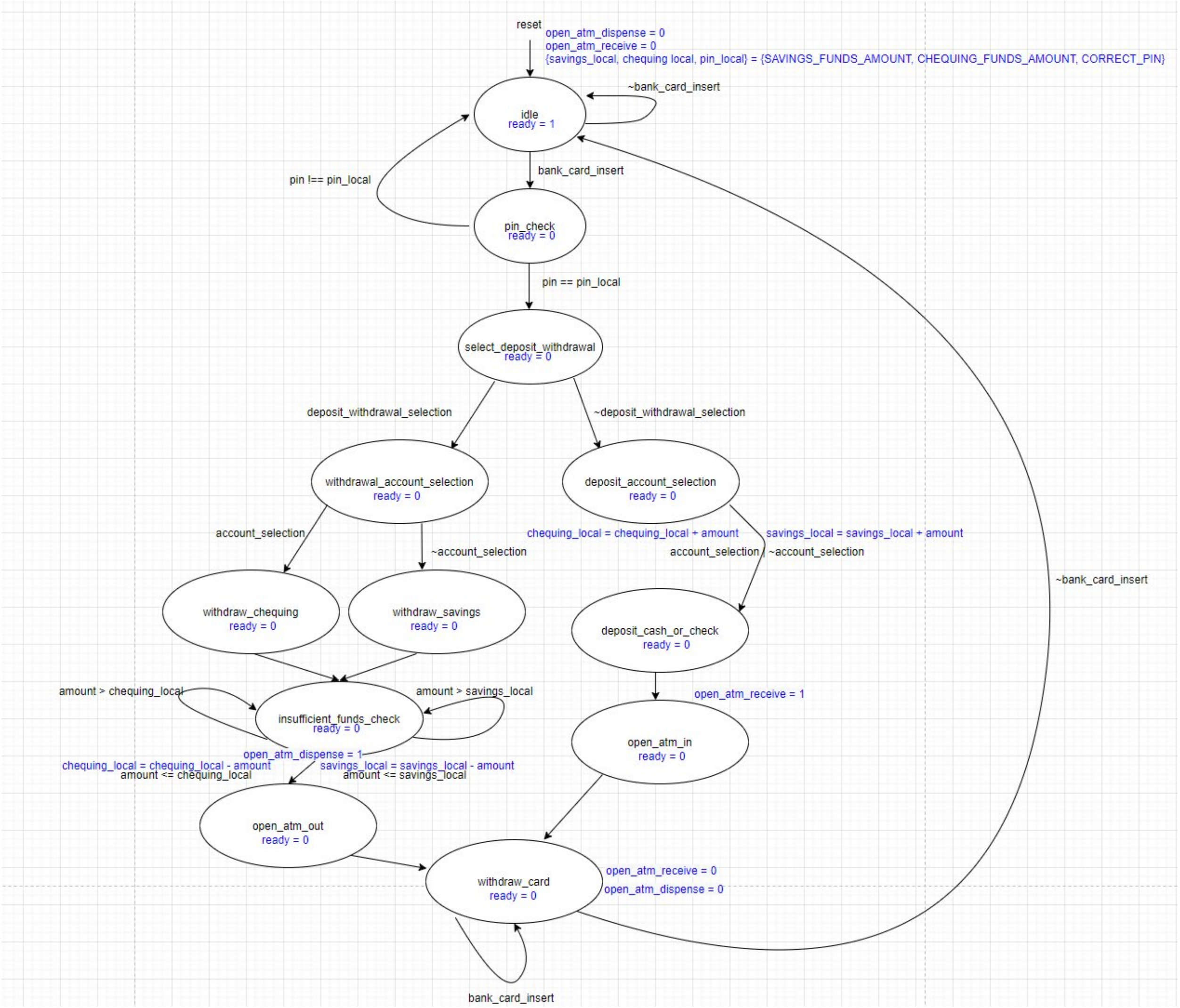
Module + Testbench Block Diagram

See .\fsm_tb_connection.drawio



FSM State Diagram

See .\fsm_state_diagram.drawio for file. Blue text indicates data being set, the black text indicates what causes the states to change.



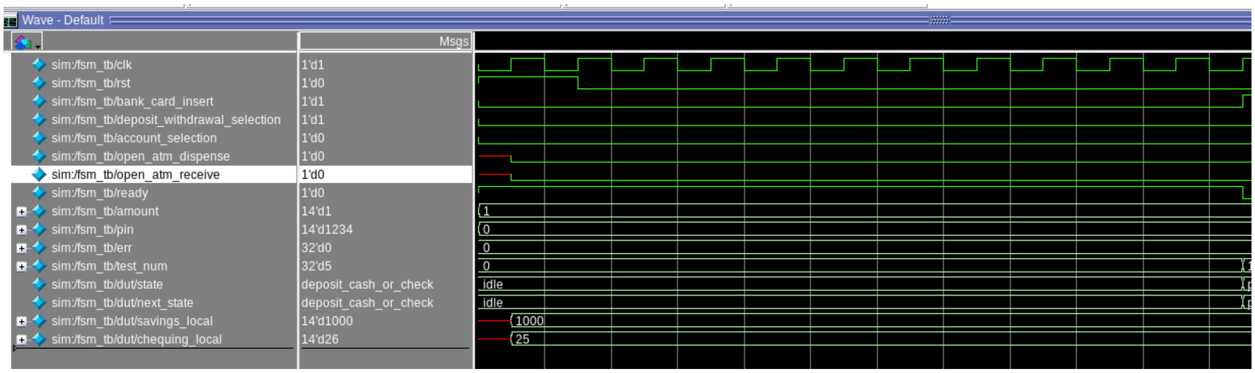
Copy of code

See .\fsm.sv and .\fsm_tb.sv for ATM FSM and FSM Testbench code respectively. (as mentioned in piazza this is OK).

Simulation waveform results

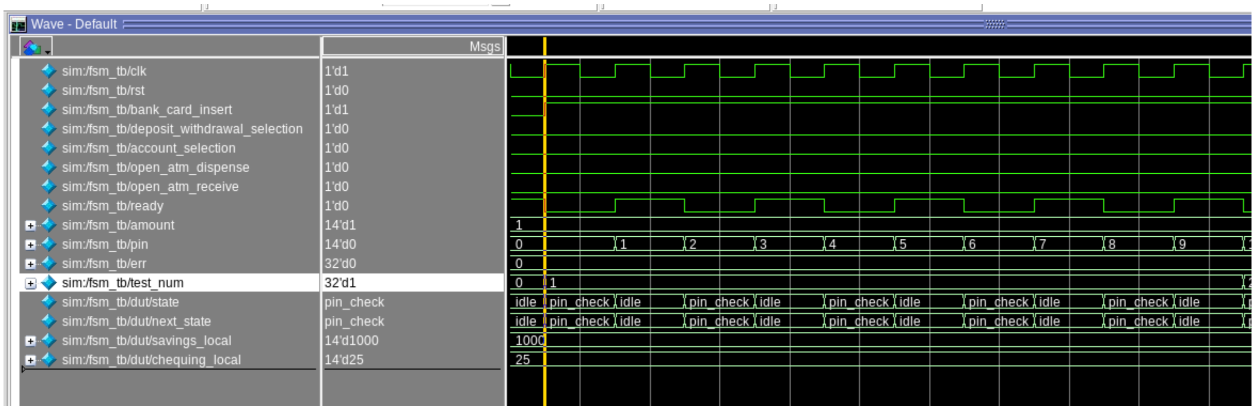
The testbench tests all possible ways of traversing the FSM. The list of tests can be seen above, or listed below to match the simulation results:

Test 0: Check to see if ready is asserted (idle state), since bank card isn't asserted high



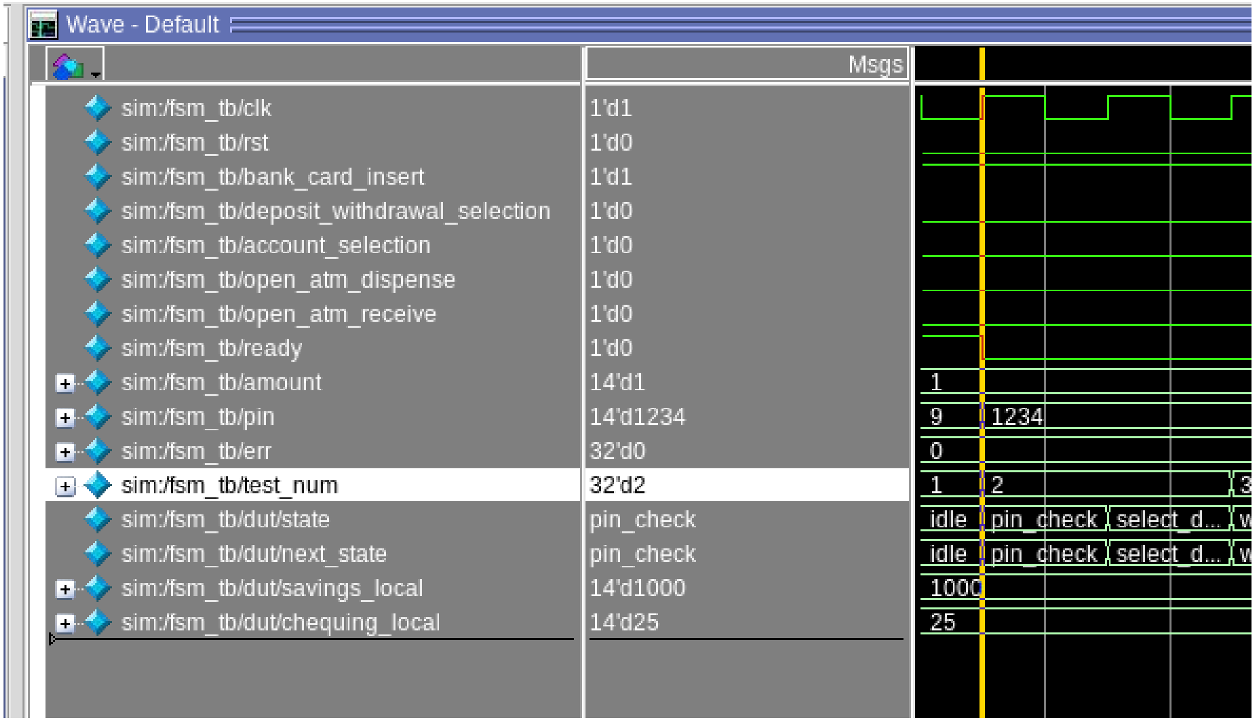
	Msgs
sim:/fsm_tb/clk	1'd1
sim:/fsm_tb/rst	1'd0
sim:/fsm_tb/bank_card_insert	1'd1
sim:/fsm_tb/deposit_withdrawal_selection	1'd1
sim:/fsm_tb/account_selection	1'd0
sim:/fsm_tb/open_atm_dispense	1'd0
sim:/fsm_tb/open_atm_receive	1'd0
sim:/fsm_tb/ready	1'd0
sim:/fsm_tb/amount	14'd1
sim:/fsm_tb/pin	14'd1234
sim:/fsm_tb/err	32'd0
sim:/fsm_tb/test_num	32'd5
sim:/fsm_tb/dut/state	deposit_cash_or_check
sim:/fsm_tb/dut/next_state	deposit_cash_or_check
sim:/fsm_tb/dut/savings_local	14'd1000
sim:/fsm_tb/dut/chequing_local	14'd26

Test 1: Checks to see if incorrect pin gives expected states of idle and pin check. Iterates from 0 to 9 for pin.


Wave - Default
1
0 1 2 3 4 5 6 7 8 9
0
0 1
idle pin_check idle pin_check idle pin_check idle pin_check idle pin_check idle
idle pin_check idle pin_check idle pin_check idle pin_check idle pin_check idle
1000
25

	Msgs
sim:/fsm_tb/clk	1'd1
sim:/fsm_tb/rst	1'd0
sim:/fsm_tb/bank_card_insert	1'd1
sim:/fsm_tb/deposit_withdrawal_selection	1'd0
sim:/fsm_tb/account_selection	1'd0
sim:/fsm_tb/open_atm_dispense	1'd0
sim:/fsm_tb/open_atm_receive	1'd0
sim:/fsm_tb/ready	1'd0
sim:/fsm_tb/amount	14'd1
sim:/fsm_tb/pin	14'd0
sim:/fsm_tb/err	32'd0
sim:/fsm_tb/test_num	32'd1
sim:/fsm_tb/dut/state	pin_check
sim:/fsm_tb/dut/next_state	pin_check
sim:/fsm_tb/dut/savings_local	14'd1000
sim:/fsm_tb/dut/chequing_local	14'd25

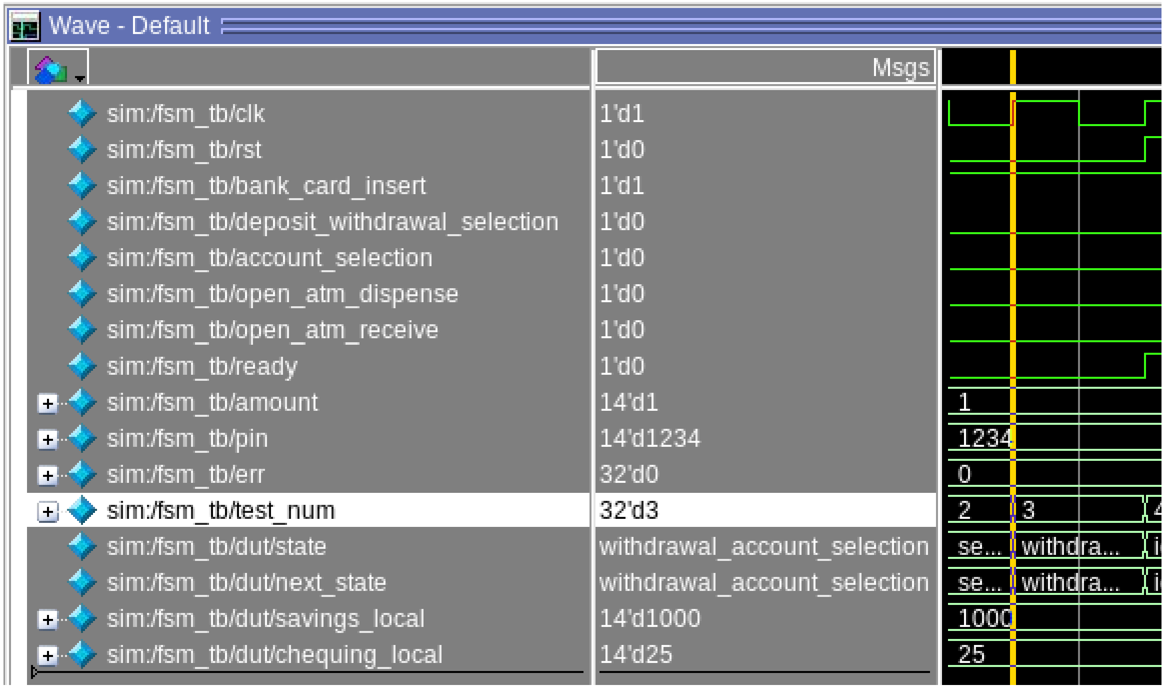
Test 2: Checks to see if the correct pin gives an expected state, which shouldn't be idle or pin_check (after one cycle).



Wave - Default

	Msgs
sim:/fsm_tb/clk	1'd1
sim:/fsm_tb/rst	1'd0
sim:/fsm_tb/bank_card_insert	1'd1
sim:/fsm_tb/deposit_withdrawal_selection	1'd0
sim:/fsm_tb/account_selection	1'd0
sim:/fsm_tb/open_atm_dispense	1'd0
sim:/fsm_tb/open_atm_receive	1'd0
sim:/fsm_tb/ready	1'd0
sim:/fsm_tb/amount	14'd1
sim:/fsm_tb/pin	14'd1234
sim:/fsm_tb/err	32'd0
sim:/fsm_tb/test_num	32'd2
sim:/fsm_tb/dut/state	pin_check
sim:/fsm_tb/dut/next_state	pin_check
sim:/fsm_tb/dut/savings_local	14'd1000
sim:/fsm_tb/dut/chequing_local	14'd25

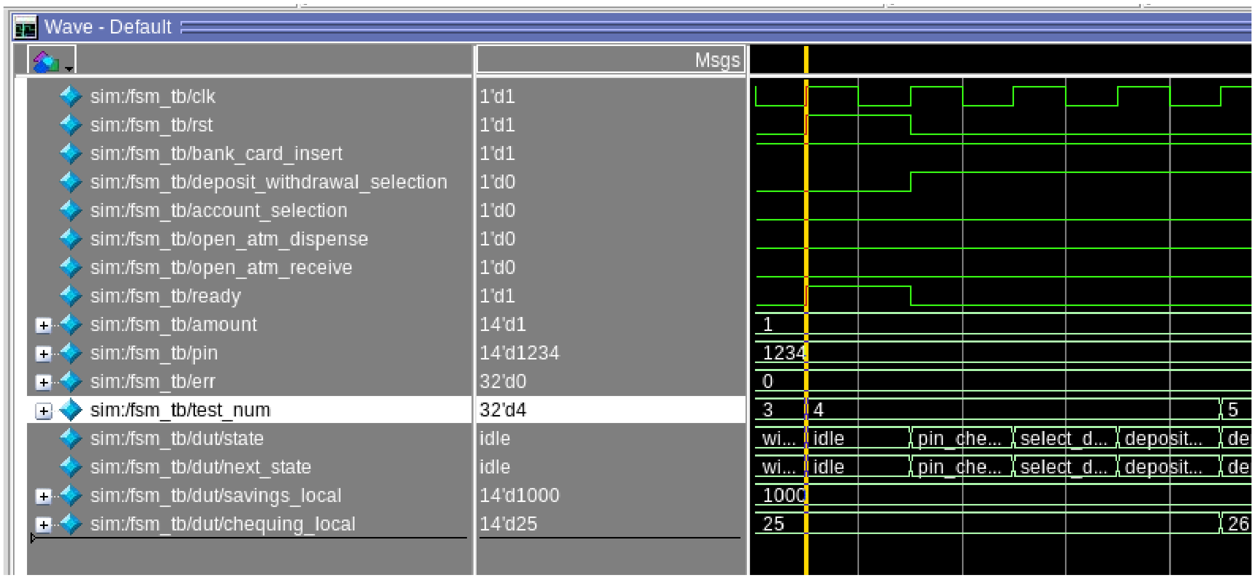
Test 3: Checks to see if withdrawal (0) makes FSM go to the correct state



Wave - Default

	Msgs
sim:/fsm_tb/clk	1'd1
sim:/fsm_tb/rst	1'd0
sim:/fsm_tb/bank_card_insert	1'd1
sim:/fsm_tb/deposit_withdrawal_selection	1'd0
sim:/fsm_tb/account_selection	1'd0
sim:/fsm_tb/open_atm_dispense	1'd0
sim:/fsm_tb/open_atm_receive	1'd0
sim:/fsm_tb/ready	1'd0
sim:/fsm_tb/amount	14'd1
sim:/fsm_tb/pin	14'd1234
sim:/fsm_tb/err	32'd0
sim:/fsm_tb/test_num	32'd3
sim:/fsm_tb/dut/state	withdrawal_account_selection
sim:/fsm_tb/dut/next_state	withdrawal_account_selection
sim:/fsm_tb/dut/savings_local	14'd1000
sim:/fsm_tb/dut/chequing_local	14'd25

Test 4: Checks to see reset works and if deposit (1) goes to correct state

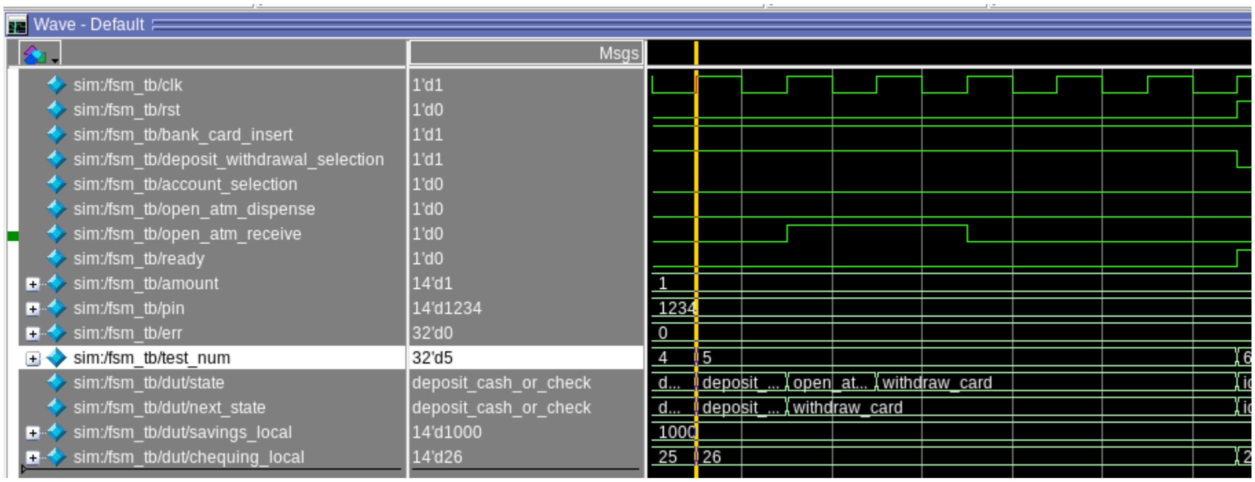


Wave - Default

	Msgs
sim:/fsm_tb/clk	1'd1
sim:/fsm_tb/rst	1'd1
sim:/fsm_tb/bank_card_insert	1'd1
sim:/fsm_tb/deposit_withdrawal_selection	1'd0
sim:/fsm_tb/account_selection	1'd0
sim:/fsm_tb/open_atm_dispense	1'd0
sim:/fsm_tb/open_atm_receive	1'd0
sim:/fsm_tb/ready	1'd1
sim:/fsm_tb/amount	14'd1
sim:/fsm_tb/pin	14'd1234
sim:/fsm_tb/err	32'd0
sim:/fsm_tb/test_num	32'd4
sim:/fsm_tb/dut/state	idle
sim:/fsm_tb/dut/next_state	idle
sim:/fsm_tb/dut/savings_local	14'd1000
sim:/fsm_tb/dut/chequing_local	14'd25

1
1234
0
3 4 5
wi... idle pin_che... select_d... deposit... de
wi... idle pin_che... select_d... deposit... de
1000
25 26

Test 5: Checks to see if deposited correct. Notice chequing local changes.

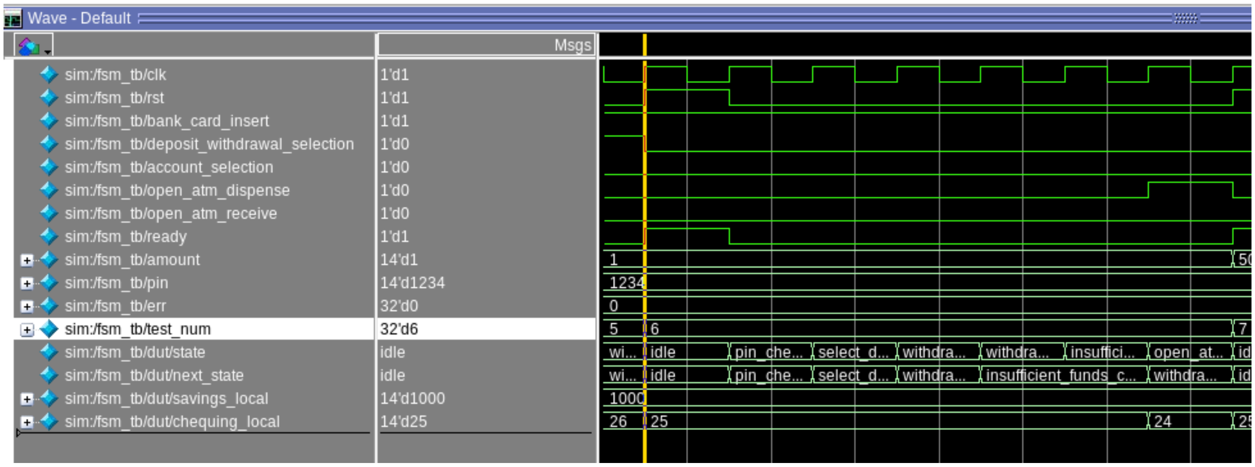


Wave - Default

	Msgs
sim:/fsm_tb/clk	1'd1
sim:/fsm_tb/rst	1'd0
sim:/fsm_tb/bank_card_insert	1'd1
sim:/fsm_tb/deposit_withdrawal_selection	1'd1
sim:/fsm_tb/account_selection	1'd0
sim:/fsm_tb/open_atm_dispense	1'd0
sim:/fsm_tb/open_atm_receive	1'd0
sim:/fsm_tb/ready	1'd0
sim:/fsm_tb/amount	14'd1
sim:/fsm_tb/pin	14'd1234
sim:/fsm_tb/err	32'd0
sim:/fsm_tb/test_num	32'd5
sim:/fsm_tb/dut/state	deposit_cash_or_check
sim:/fsm_tb/dut/next_state	deposit_cash_or_check
sim:/fsm_tb/dut/savings_local	14'd1000
sim:/fsm_tb/dut/chequing_local	14'd26

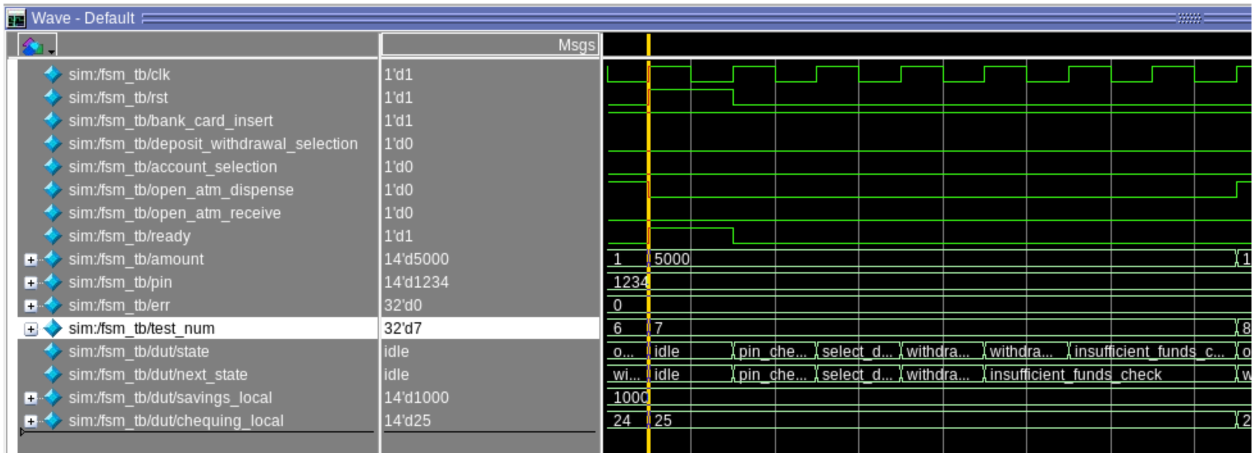
1
1234
0
4 5 6
d... deposit ... open_at... withdraw_card id
d... deposit ... withdraw_card id
1000
25 26 2

Test 6: Checks to see if reset + withdrawal was correct. Notice chequing local was withdrawn.


Wave - Default

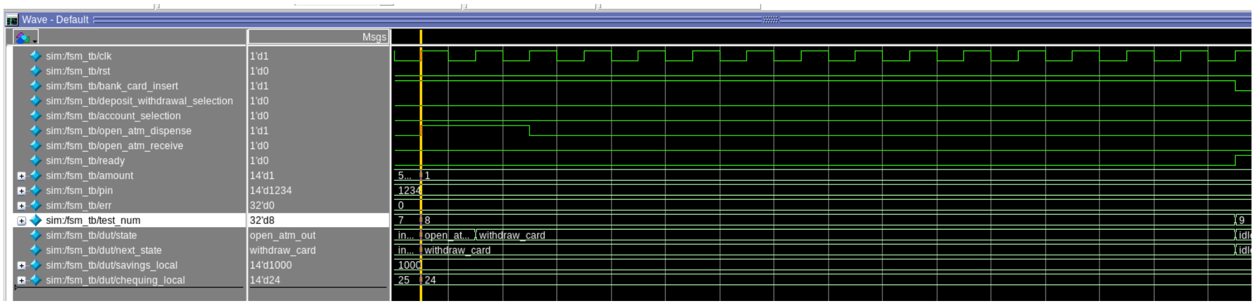
	Msgs
sim:/fsm_tb/clk	1'd1
sim:/fsm_tb/rst	1'd1
sim:/fsm_tb/bank_card_insert	1'd1
sim:/fsm_tb/deposit_withdrawal_selection	1'd0
sim:/fsm_tb/account_selection	1'd0
sim:/fsm_tb/open_atm_dispense	1'd0
sim:/fsm_tb/open_atm_receive	1'd0
sim:/fsm_tb/ready	1'd1
sim:/fsm_tb/amount	14'd1
sim:/fsm_tb/pin	14'd1234
sim:/fsm_tb/err	32'd0
sim:/fsm_tb/test_num	32'd6
sim:/fsm_tb/dut/state	idle
sim:/fsm_tb/dut/next_state	idle
sim:/fsm_tb/dut/savings_local	14'd1000
sim:/fsm_tb/dut/chequing_local	14'd25

Test 7: Checks to see if insufficient funds works. Should loop insufficient funds.


Wave - Default

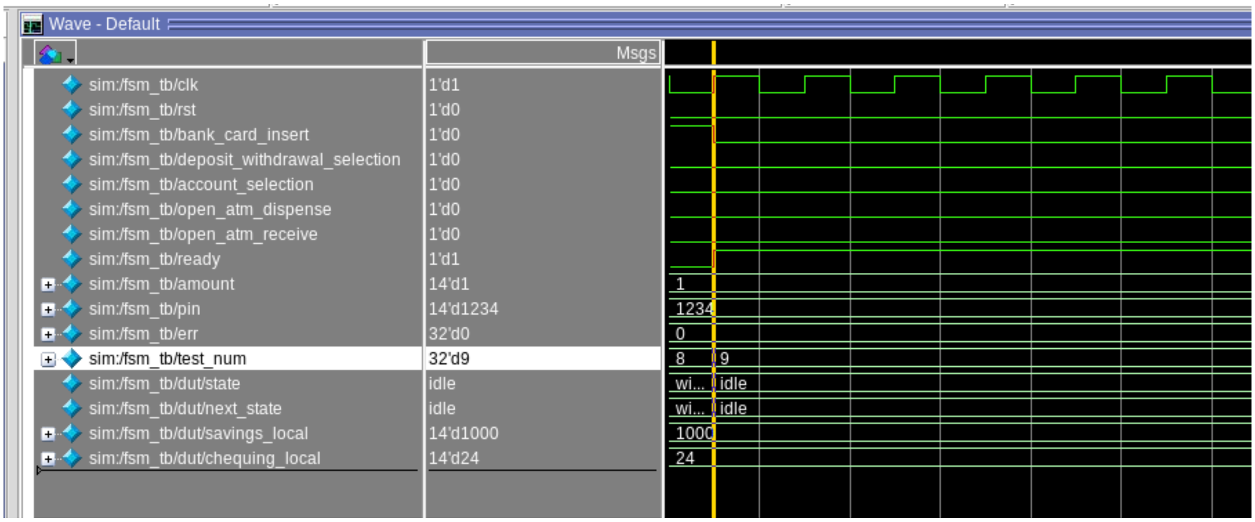
	Msgs
sim:/fsm_tb/clk	1'd1
sim:/fsm_tb/rst	1'd1
sim:/fsm_tb/bank_card_insert	1'd1
sim:/fsm_tb/deposit_withdrawal_selection	1'd0
sim:/fsm_tb/account_selection	1'd0
sim:/fsm_tb/open_atm_dispense	1'd0
sim:/fsm_tb/open_atm_receive	1'd0
sim:/fsm_tb/ready	1'd1
sim:/fsm_tb/amount	14'd5000
sim:/fsm_tb/pin	14'd1234
sim:/fsm_tb/err	32'd0
sim:/fsm_tb/test_num	32'd7
sim:/fsm_tb/dut/state	idle
sim:/fsm_tb/dut/next_state	idle
sim:/fsm_tb/dut/savings_local	14'd1000
sim:/fsm_tb/dut/chequing_local	14'd25

Test 8: Checks to see if withdraw card not being withdrawn (1) makes it wait for withdrawing the bank card.


Wave - Default

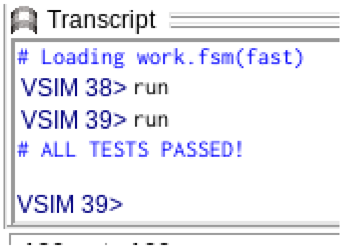
	Msgs
sim:/fsm_tb/clk	1'd1
sim:/fsm_tb/rst	1'd0
sim:/fsm_tb/bank_card_insert	1'd1
sim:/fsm_tb/deposit_withdrawal_selection	1'd0
sim:/fsm_tb/account_selection	1'd0
sim:/fsm_tb/open_atm_dispense	1'd1
sim:/fsm_tb/open_atm_receive	1'd0
sim:/fsm_tb/ready	1'd0
sim:/fsm_tb/amount	14'd1
sim:/fsm_tb/pin	14'd1234
sim:/fsm_tb/err	32'd0
sim:/fsm_tb/test_num	32'd8
sim:/fsm_tb/dut/state	open_atm_out
sim:/fsm_tb/dut/next_state	withdraw_card
sim:/fsm_tb/dut/savings_local	14'd1000
sim:/fsm_tb/dut/chequing_local	14'd24

Test 9: Checks to see if withdrawing the card (0) makes it go back to idle, thus ready is brought high.



```
sim:/fsm_tb/clk                            1'd1
sim:/fsm_tb/rst                            1'd0
sim:/fsm_tb/bank_card_insert               1'd0
sim:/fsm_tb/deposit_withdrawal_selection   1'd0
sim:/fsm_tb/account_selection              1'd0
sim:/fsm_tb/open_atm_dispense              1'd0
sim:/fsm_tb/open_atm_receive               1'd0
sim:/fsm_tb/ready                          1'd1
sim:/fsm_tb/amount                         14'd1
sim:/fsm_tb/pin                            14'd1234
sim:/fsm_tb/err                            32'd0
sim:/fsm_tb/test_num                       32'd9
sim:/fsm_tb/dut/state                      idle
sim:/fsm_tb/dut/next_state                 idle
sim:/fsm_tb/dut/savings_local              14'd1000
sim:/fsm_tb/dut/chequing_local             14'd24
```

All tests passed:



```
# Loading work.fsm(fast)
VSIM 38> run
VSIM 39> run
# ALL TESTS PASSED!

VSIM 39>
```