Interrupt Intro

Snack food machine

Theft possibility

Theft Possibility (Accel)

Timer

button pushed

Alarm & lock-up

Theft Possibility

Display Price

Timer

Regular State

Idle State

return

Coin inserted

Return $

Return

Coin inserted

Money state (& money not too much)

button press

Verificat

Validate $ & item

If $$ >= item

Vend & return change
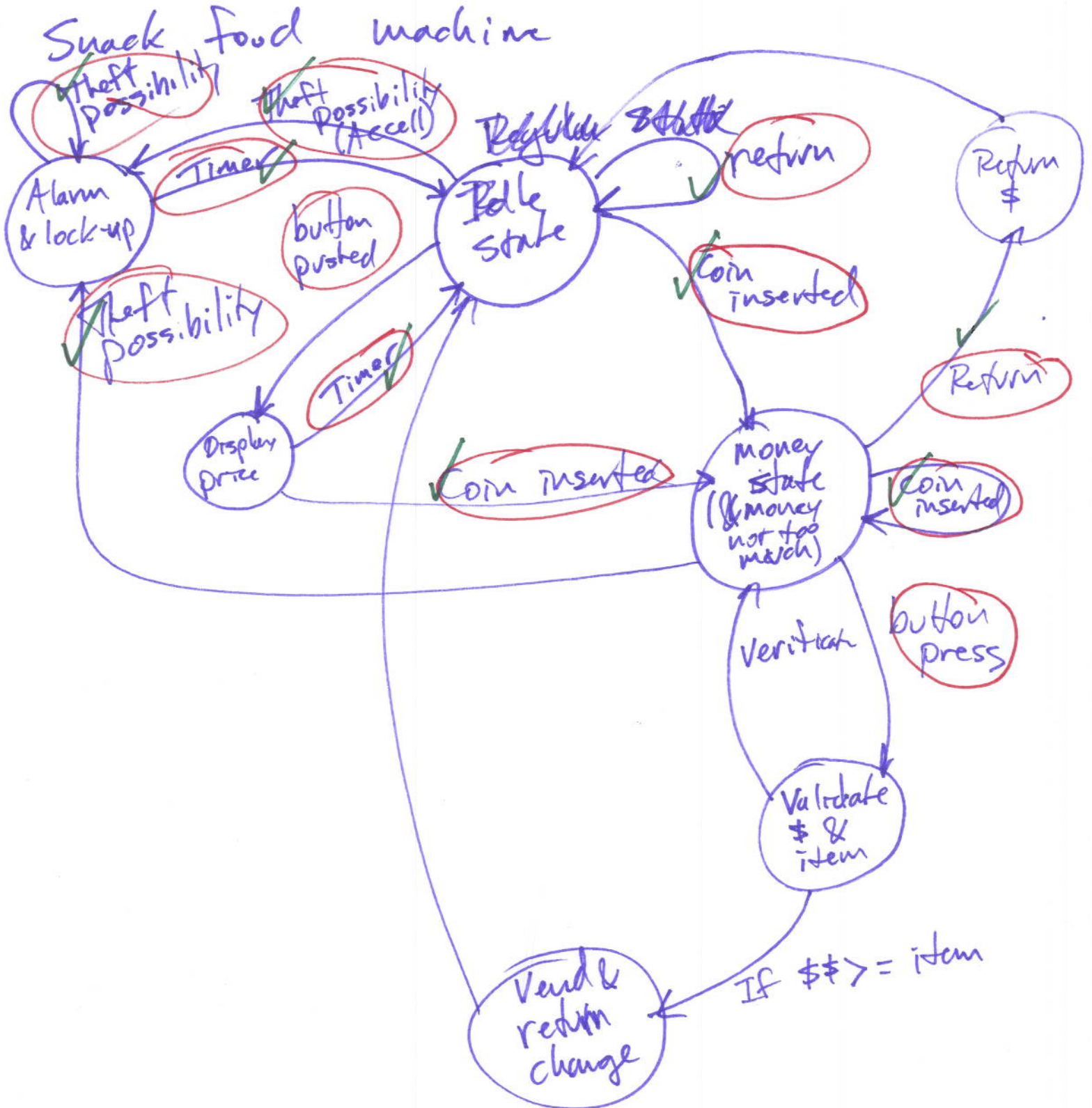
theft_timer_ISR (void)
    state = idle_state

Display_timer_ISR (void)
    state = idle_state

return_button_ISR (void)
    if state == money_state
      then state = return_money

Coin_inserted_ISR (void)
    if not a valid coin, return coin, return
    state = money_state
    save coin value in coin_inserted (variable)

theft_possibility_ISR (void)
    state = alarm_lock_up
    start_theft_timer

button_press_ISR (void)
    if state == idle state
      state = display_state
    else if state == money_state
      state = verification_state
    save button press in button_value (variable)

```
while (1) {

    switch (state)

        idle_state:      idle_state_function
        display_state:   display_state_function
        alarm_state:     alarm_lockup_function
        return_money:    return_money_function
        money_state:
        validate_item:
        vend_and_return:
    end switch

}
```

```
return_money_function (void) {
    return amount of money in    money_count
                                    (variable)
    money_count = 0
    coin_inserted = 0
    button_value = 0
    state = idle_state
```

alarm _ lockup _ ~~function~~ (void)
    lock drawer, dispenser, coins
    sound alarm
    activate taser

display_ price _ function
    display price of item (button-value)