SPEC: Vending Machine

I.   I/O Ports:
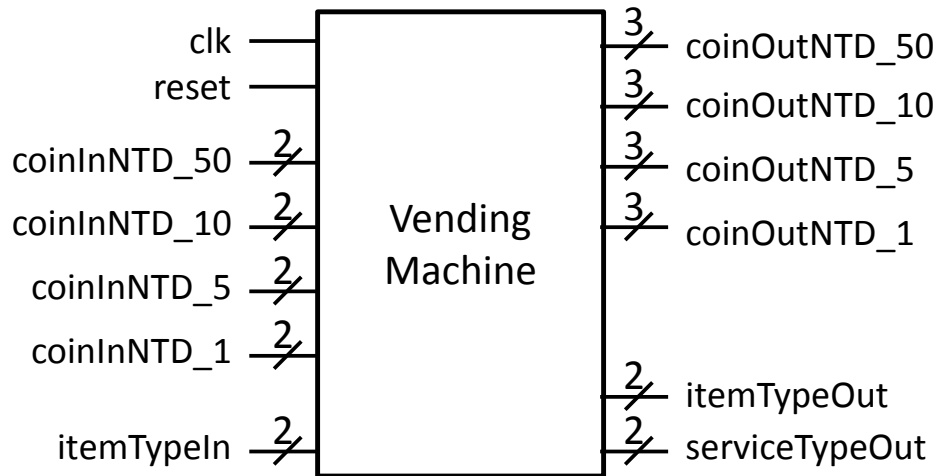

Figure 1.   The I/O ports of the vending machine

Table 1.   Table 1. I/O port descriptions

signal	I/O	bit-width	description
clk	input	1	General clock signal.
reset	input	1	General reset signal.
coinInNTD_XX	input	2	The signal(s) specifying the input number(s) of the corresponding coin(s). (**NTD_50**, **NTD_10**, **NTD_5**, and **NTD_1**).
itemTypeIn	input	2	The signal specifying the request item. It could be **ITEM_A**: cost 8 NT dollars. **ITEM_B**: cost 15 NT dollars. **ITEM_C**: cost 22 NT dollars. **ITEM_NONE**: no request.
coinOutNTD_XX	output	3	The signal(s) specifying the number(s) of the corresponding coin(s) in the change. (**NTD_50**, **NTD_10**, **NTD_5**, and **NTD_1**).

itemTypeOut	output	2	The signal specifying the item given by the vending machine. It could be
			ITEM_A, **ITEM_B**, and **ITEM_C**:
			   successful request.
			ITEM_NONE:
			   no request or change not enough.

serviceTypeOut	output	2	The signal specifying the vending machine status. It could be **SERVICE_ON**: waiting for request. **SERVIE_BUSY**: dealing with request. **SERVICE_OFF**: output result.

II. System Description:

The vending machine accepts requests from customers, and then it calculates the change to the customer depending on the input coin value and the request item. If the input coin value is not enough for the item or the coins in the machine is not enough for the change, the machine gives nothing and returns the input coins to the customer. Otherwise, the machine gives the item that the customer requested and the change to the customer.
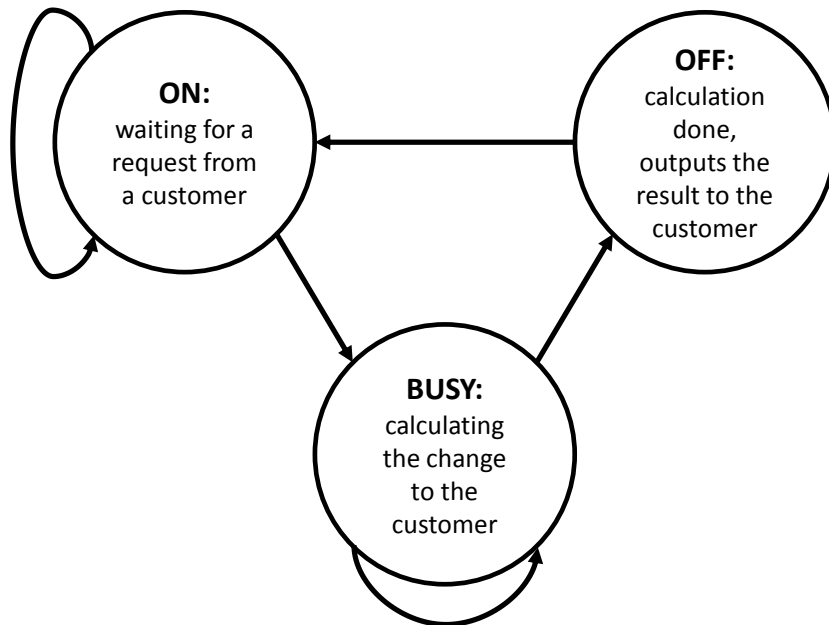
III. State Diagram:

Figure 2. **The state diagram of the vending machine**

IV. Assumptions:

(a) The input coin numbers is limited to 3 (2-bits) and the output coin numbers is limited to 7 (3-bits).

(b) The stored coin numbers is limited to 7 (3-bits), so the machine may lose coins when the capacity is reached. (e.g. Assume there are 5 coins of NTD 10 in the machine. After inserting 3 coins of NTD 10, the machine still stores 7 coins of NTD 10 instead of 8 coins. That is, the machine loses 1 coin definitely.)

(c) The machine is not initialized and out of service until a *reset* signal is arrived. The machine initially stores 2 coins for each type of NTD.

(d) The request is only accepted when the machine is in **SERVICE_ON** and the input *itemTypeIn* is not **ITEM_NONE**. The input coins are taken also only on such condition.

(e) The machine will never "eat money".