Minor-1

Vending Machine :

→ Accepted Denominations — ₹5, ₹10, ₹20, ₹50

Items — 40 (given)

Types — 5 → Packets : 30 rupees
Cans : 40 rupees
P_bottles : 25 rupees
Wrappers : 15 rupees
P_tubs : 60 rupees.

① I/p's, O/p's :

I/p's in → input amount

clk, rst

item → to select an item from given list of 40 items and give as input.

Outputs :

reg out → to give the output product. (indicates '1' if amount is sufficient) else '0'.

reg add_still → used to describe if the given input amount is not sufficient to buy the required product.

It indicates '1' if still amount needed to be added. and '0' if not.

reg change → to give back the extra amount after buying the required product.

product out → It gives the required product. i-e mentions which product we asked for.

Assumptions :

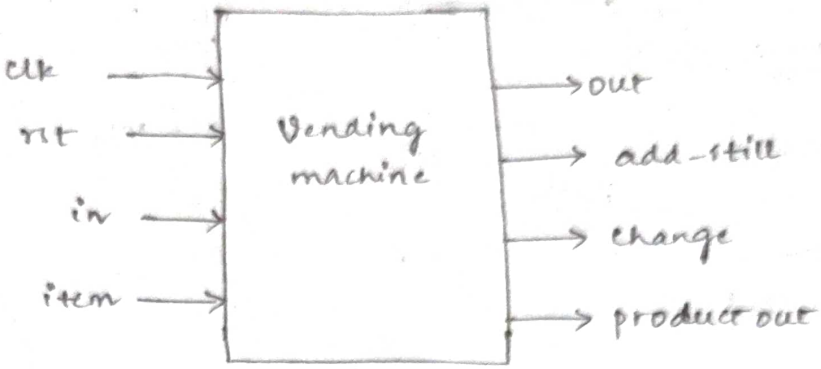
→ So, from the given 40 items list, we take and group the items according to their existance.

i-e some items comes under the packets and some comes under the category of Cans and same for p-bottles, wrappers, p-tubs as well.

→ My machine will give the change back after calculating the amount of required product.

So, my machine is honest.

Block diagram:



- internal variables:

main_item → to declare type of item.

ns → indicates the state.

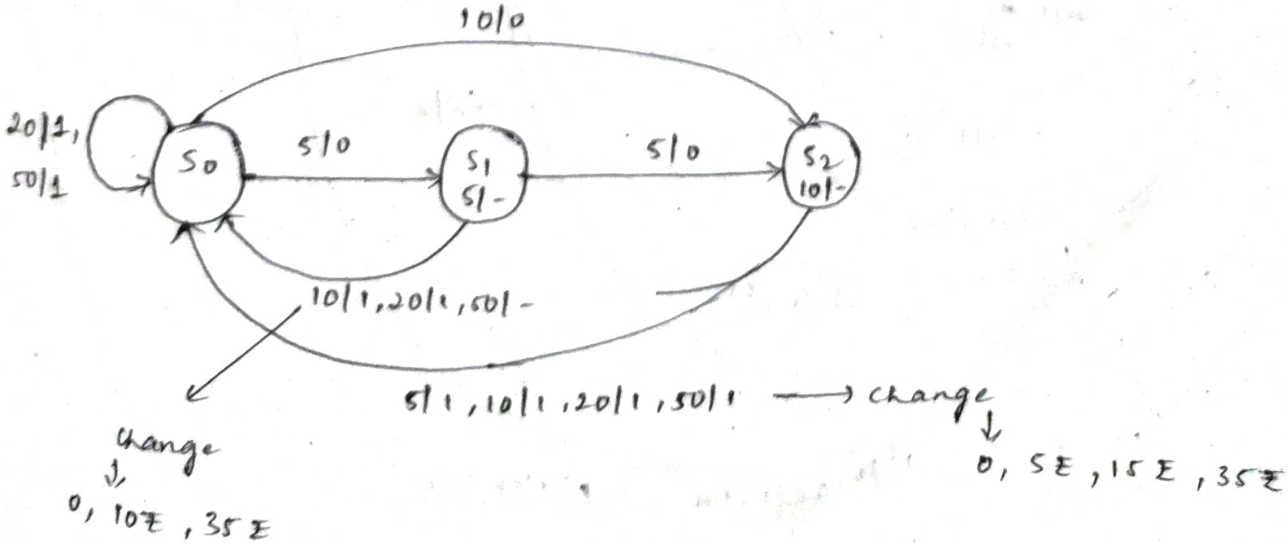
→ Overall there will be 11 states for amount.

highest amount product is plastic tubs i-e '60Z'.

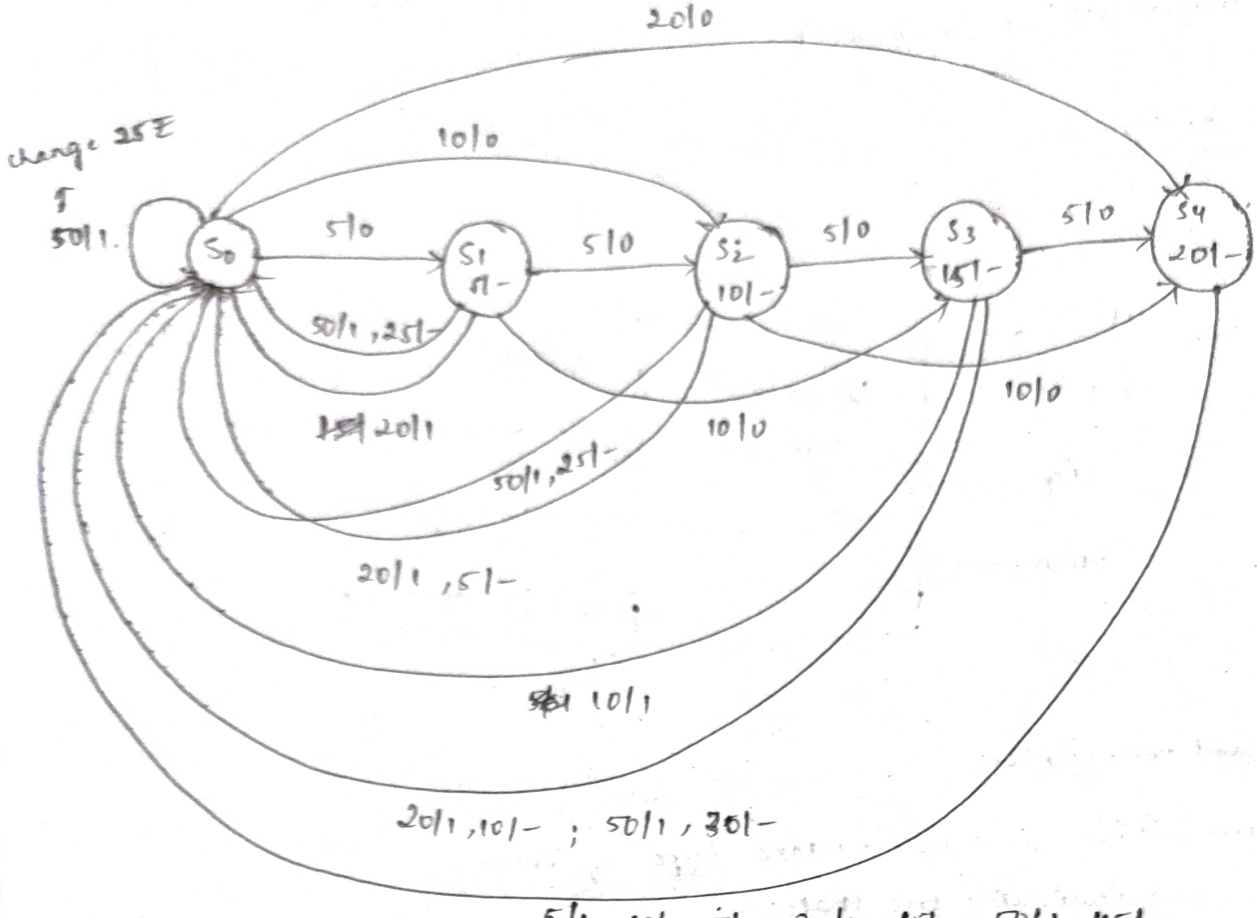
→ For 40 items there will be 40 states.

② State Diagram: (idle state will depend on rst signal).

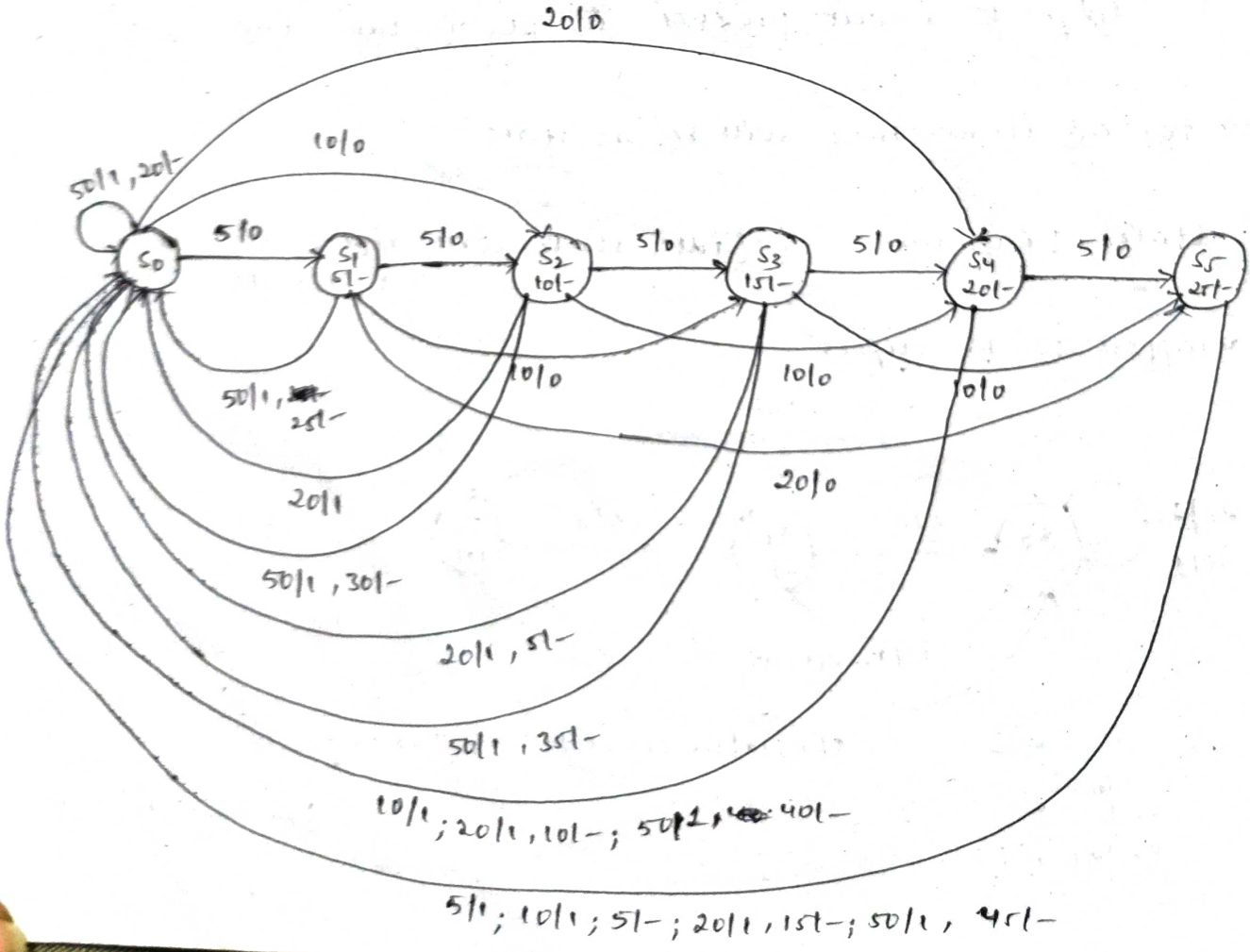
wrappers : 15 rupees.



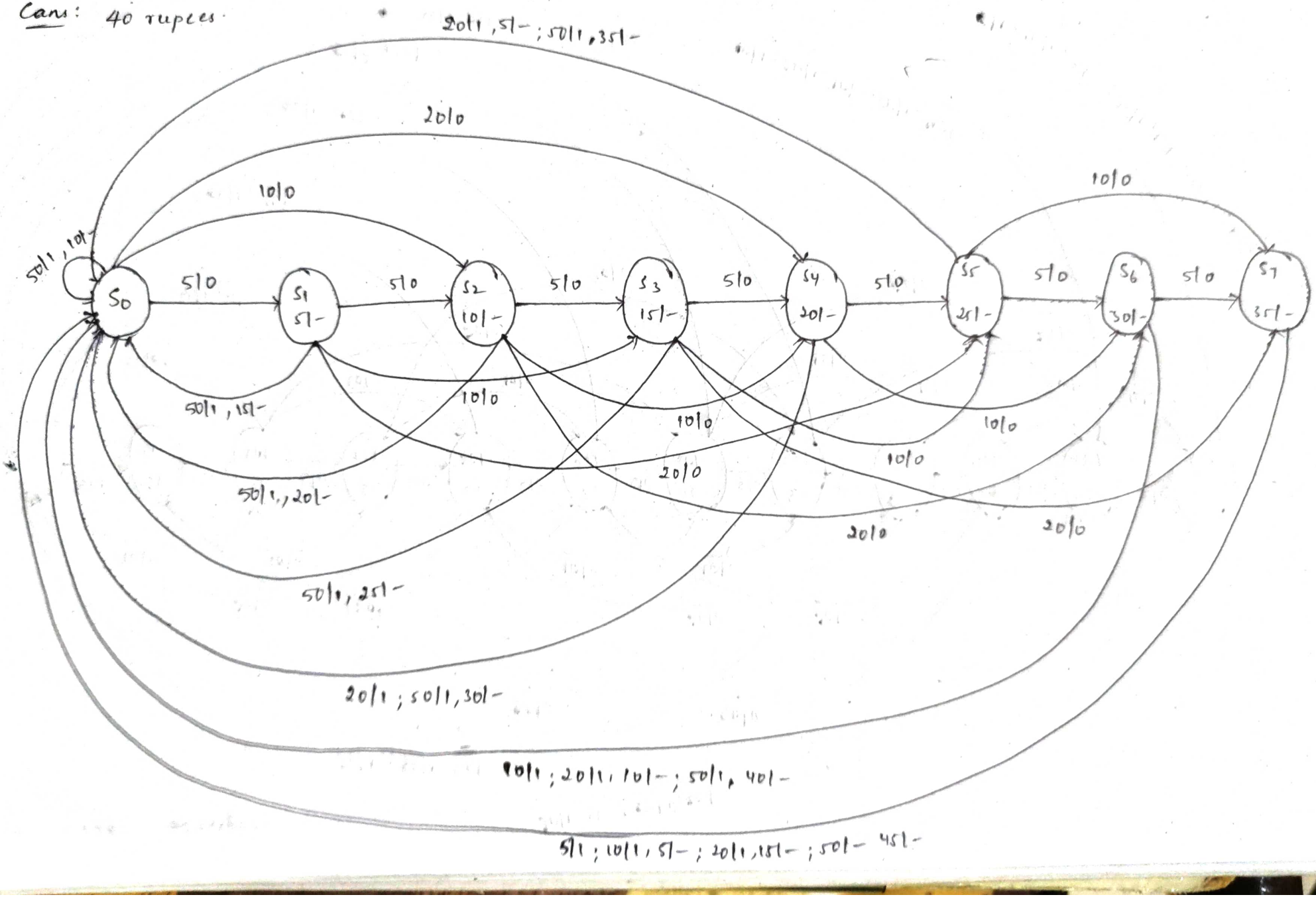
Plastic_bottles: 25 rupees



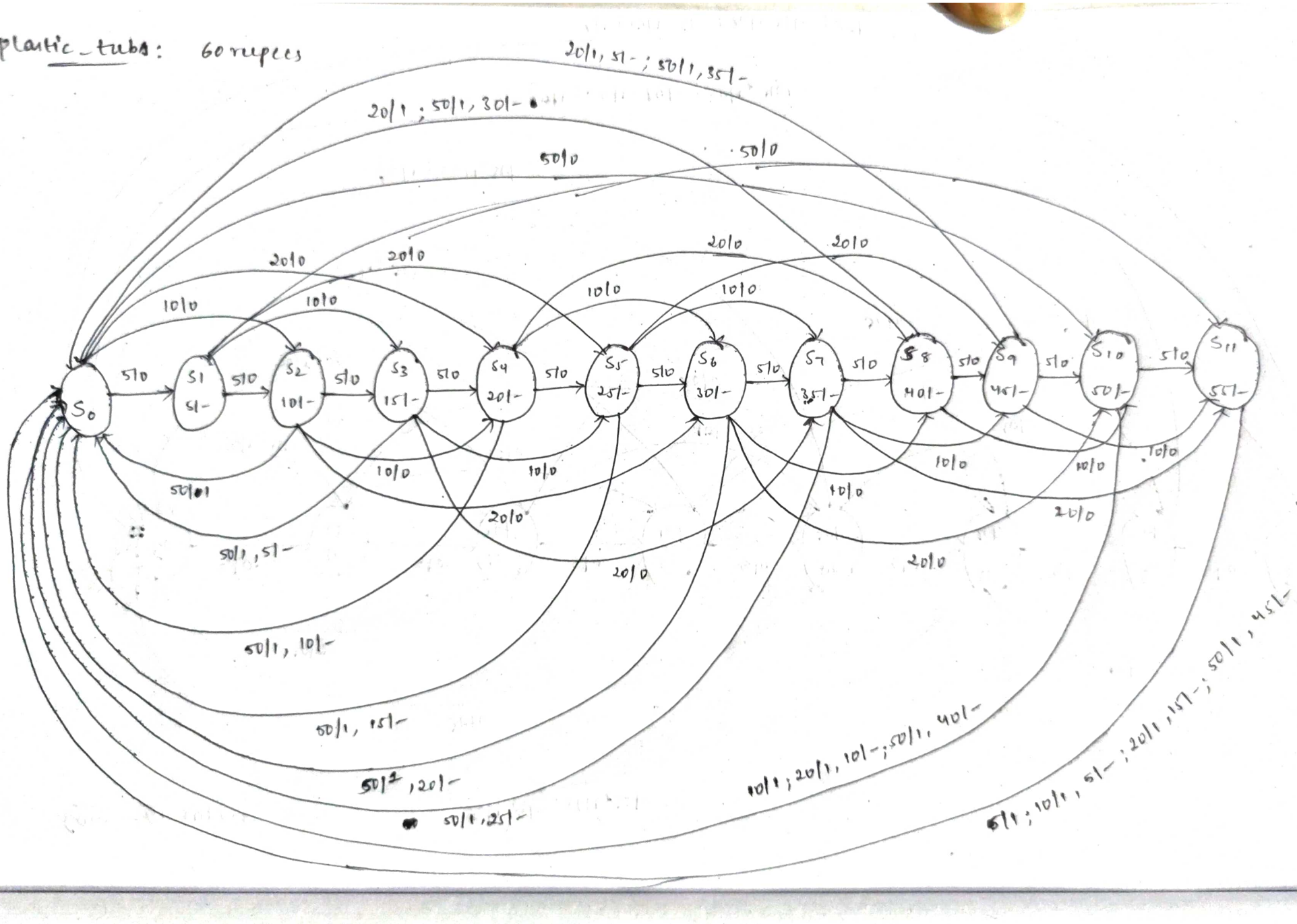
Packets: 30 rupees.



Cans: 40 rupees



plastic_tubs: 60 rupees



③ Functionality :

→ Selects an item from the list of 40 items.

→ If the rst is high, then stays in idle state and starts working when reset is low.

→ At the clk of posedge it works.

→ Based on the input item, it decides what type of item we have asked for.

→ Then based on the input amount, it decides for what state it need to change.

→ Then after reaching the state, it there is any requirement of giving back the change for the given input amount, it returns.

→ It gives the output product and change.

→ It works on principle of Mealy.