

## LAB 1: Thiết kế máy bán nước ngọt tự động

### 1. Mạch Điều Khiển Máy Bán Nước Soda

#### a) Phân Tích Vấn Đề

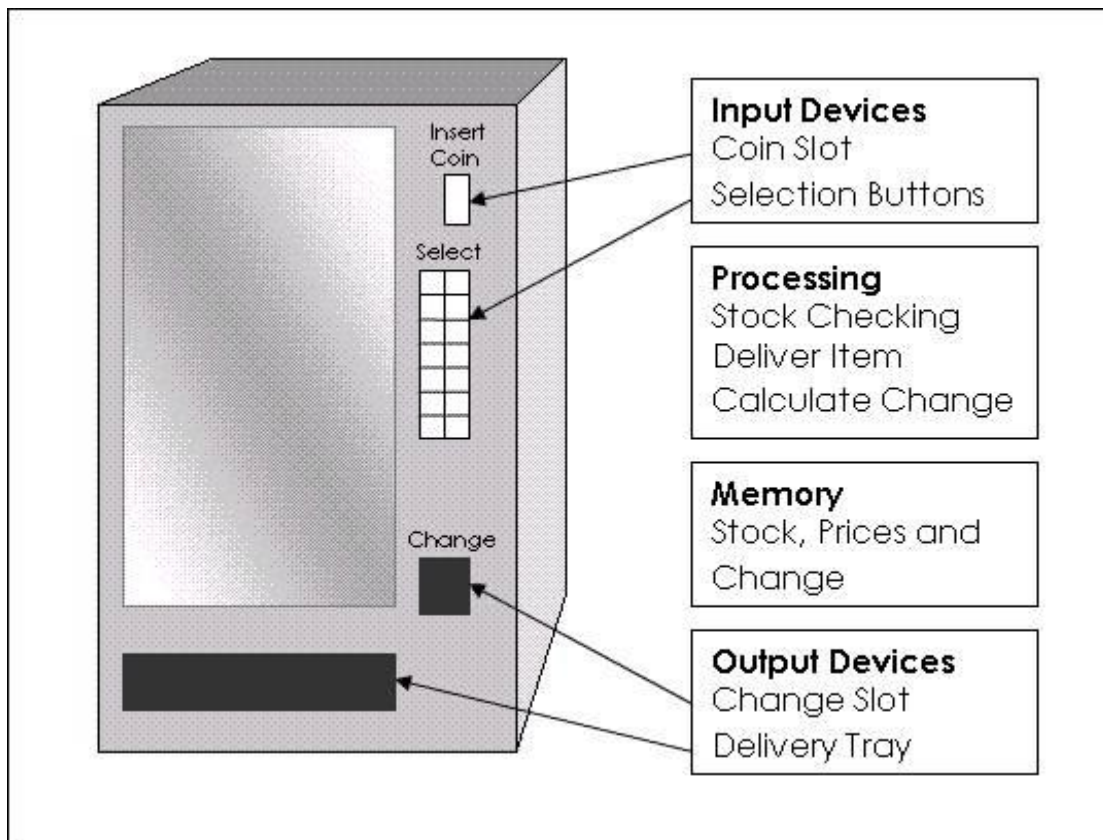
Thiết kế một bộ điều khiển máy bán nước Soda có giá  $\phi 20$ . Người mua sẽ bỏ đồng xu các loại \$5 (Nickel), \$10 (Dime), \$25 (Quarter) vào máy bán nước ngọt. Khi máy nhận đủ hoặc quá số tiền \$20 thì máy sẽ lấy nước ra và trả lại tiền dư nếu có. Vending Machine có:

- 3 inputs: nickel, dime, và quarter
- 2 outputs: soda, change

Nickle, Dime, và Quarter trong mỗi cycle chỉ có 1 trong 3 là tích cực mức cao.

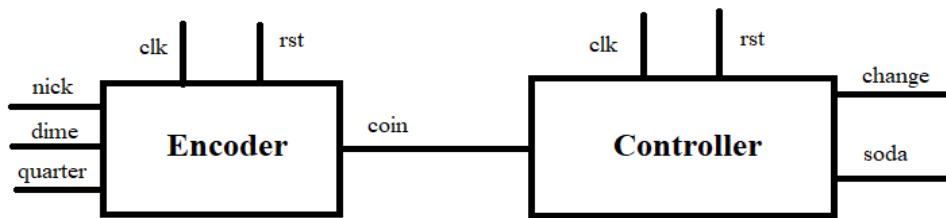
Change là tiền thối thì không cần đổi thành Nickle, Dime, và Quarter, chỉ cần cho biết số tiền thối:

- 000: \$0
- 001: \$5
- 010: \$10
- 011: \$15
- 100: \$20



#### b) Thiết kế

- Sơ đồ khối của Vending machine:



- Thiết kế sơ đồ khối:

+ Khối Encoder:

Mã hóa ngõ vào về dạng nhị phân để đưa vào khối control xử lý.

- 001: \$5

- 010: \$10

- 101: \$25

Input : clk,rst,nick (\$5),dime(\$10),quarter(\$25)

Output: coin

Đoạn code:

```
module encoder(clk,rst,nick,dime,quarter,coin);
```

```
input clk,rst;
```

```
input nick,dime,quarter;
```

```
output reg [2:0] coin;
```

```
always @(posedge clk)
```

```
begin
```

```
    if (rst) coin<=3'b0;
```

```
    else
```

```
    begin
```

```
        if (nick==1'b1) coin<=3'h1;
```

```
        else if(dime==1'b1) coin<=3'h2;
```

```
        else if(quarter==1'b1) coin<=3'h5;
```

```
        else coin<=3'b0;
```

```
    end
```

```
end
```

```
endmodule
```

+ Khối controller

Xử lý tín hiệu từ khối encoder, sau đó cho ra tín hiệu để quyết định hành động tiếp theo. Nếu số tiền đã bỏ vào lớn hơn \$20 , máy sẽ bán soda và thối lại tiền thừa nếu có. Nếu số tiền nhỏ hơn \$20, máy sẽ không làm gì cả

Input :clk, rst, coin

Output : Soda, Change

Đoạn code:

```

module control(clk,rst,coin,soda,change);
input clk,rst;
input [2:0] coin;
output reg soda;
output [2:0]change;
reg[3:0] out;

always @(posedge clk)
begin
    if (rst)
        begin
            total = 4'b0;
            out = 4'b0;
        end
    else
        begin
            total = total + coin;
            if (total >= 4'h4)
                begin
                    soda = 1'b1;
                    out = total - 4'h4;
                    total = 4'b0;
                end
            else
                begin
                    soda = 1'b0;
                    out = 4'b0;
                end
        end
    end
end
assign change = out[2:0];
endmodule

```

+ Kết nối các khối

Đoạn code:

```

module TEST;
reg clk,rst,nick_i,dime_i,quarter_i;
wire soda_i;
wire [2:0] change_i,coin_i;

encoder encoder0 (.clk(clk),
                  .rst(rst),
                  .nick(nick_i),

```

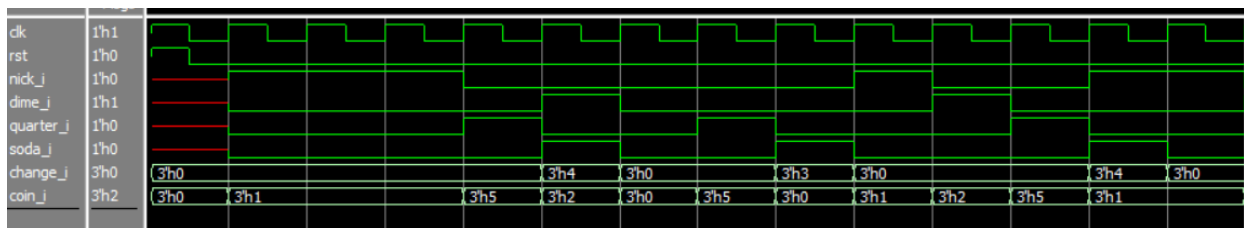
```

        .dime(dime_i),
        .quarter(quarter_i),
        .coin(coin_i));

control control0 (.clk(clk),
                 .rst(rst),
                 .coin(coin_i),
                 .soda(soda_i),
                 .change(change_i));
Endmodule

```

## 2 . Kết quả mô phỏng



## 3 . Kết luận và định hướng phát triển

- Phân thiết kế, kiểm tra khá chính xác.
- Có thể mở rộng sản phẩm theo nhiều hướng: thêm nhiều loại nước, nhận dạng nhiều loại tiền có mệnh giá khác nhau , thêm nút hoàn tiền để trả tiền lại nếu số tiền bỏ vào không đủ mua nước
- Sản phẩm còn cần phụ thuộc vào phần cơ khí.