


```

        next = STT_1100;
    STT_1100:
        if (in)
            next = STT_11001;
        else
            next = STT_0;
    STT_11001:
        if (in)
            next = STT_11;
        else
            next = STT_0;
    endcase
end
end

```

همچنین بر سر ضرب بالارونده کلاک، مقدار بعدی را در state قرار می‌دهیم.

```

always @(posedge clk, posedge reset) begin
    if (reset)
        state <= STT_START;
    else
        state <= next;
end

```

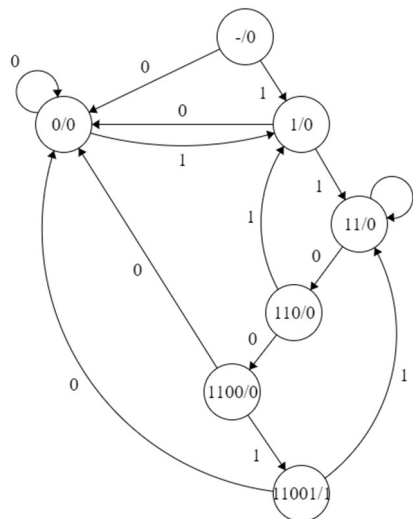
همچنین یک مدار ترکیبی برای مشخص کردن خروجی از روی state و ورودی فعلی می‌سازیم. توجه کنید که خروجی تنها در صورتی برابر 1 است که در حالت 1100 باشیم و ورودی 1 باشد.

```

always @(*) begin
    if (in && state == STT_1100)
        out = 1;
    else
        out = 0;
end

```

در ماشین Moore، خروجی به state وابسته است. بنابراین در FSM این ماشین، خروجی را در هر state مشخص می‌کنیم. FSM مورد نظر، به شکل زیر خواهد شد.

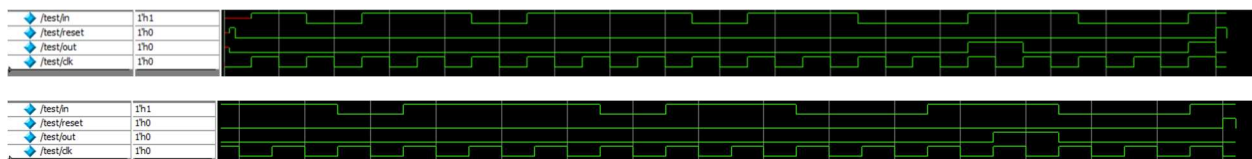


مشابها می‌توانیم مدار مورد نظر را پیاده‌سازی کنیم. توجه کنید که در این پیاده‌سازی، state تنها در لبه بالارونده کلاک تغییر می‌کند. از آنجایی که خروجی نیز تابع state است، خروجی نیز تنها در لبه بالارونده کلاک تغییر می‌کند. در ادامه بخشی از کد را مشاهده می‌کنید که مربوط به موارد مذکور در پیاده‌سازی Moore می‌شود.

```
always @(posedge clk, posedge reset) begin
    if (reset)
        state <= STT_START;
    else
        state <= next;
end

assign out = (state == STT_11001);
```

توجه بفرمایید که در فایل ارسالی، پیاده‌سازی Moore به صورت comment نوشته شده است.



در تصاویر بالا می‌توانید شکل موج به ترتیب ماشین Mealy و Moore پیاده‌سازی شده را مشاهده فرمایید. توجه کنید که این تفاوت در Output کاملاً طبیعی است چرا که ماشین Mealy در لحظه خروجی خود را تغییر می‌دهد در صورتی که ماشین Moore، تابع state است.