

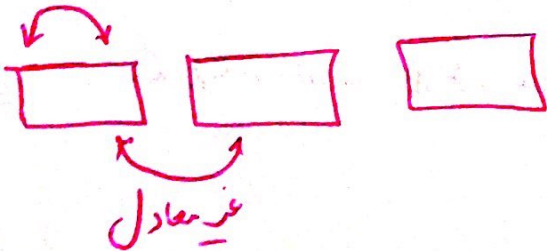
(II) در یک FSM، دو حالت  $s_i$  و  $s_j$  را دو حالت معادل (equivalent) گویند اگر و تنها اگر به ازای تمام وضعیت های ورودی (دنباله های ورودی)، دنباله های خروجی آن ها مستقل از آنکه  $s_i$  و  $s_j$  شروع کنیم، یکسان باشند.

- مثلاً در مثال حل شده قبل با فرض ورودی  $x$  و خروجی  $z$ ،  $s_i$  و  $s_j$  معادل باشند،  
 - **نمود:** اگر دو حالت  $s_i$  و  $s_j$  معادل باشند،  
 $w = 01101001$   
 $z = 01101101$   
 نگاه کنید K-SUC های آن ها (به ازای تمام مقادیر K) نیز یکسان هستند.

- معمولاً پیدا کردن state هایی که معادل هستند مشکل است. اما با بررسی می توان نشان داد دو state معادل نیستند و آن ها را از هم جدا کرد. ← **ایده روش partitioning**

**تعریف:** یک partition از چند بلوک تشکیل شده که در هر بلوک تعدادی state وجود دارد. state های موجود در یک بلوک امکان دارد معادل هم باشند ولی قطعاً معادل state هایی که در بلوک دیگر هستند نمی باشند.

ممکن است معادل باشند



## روش partitioning :

(۱) ابتدا فرض می‌کنیم  $P_1$  state ها معادلند. شکل  $P_1$

(۲) سپس state ها را براساس یکان بودن خودی به (ازای تمام حالت‌های ورودی دلتا بندی می‌کنیم.

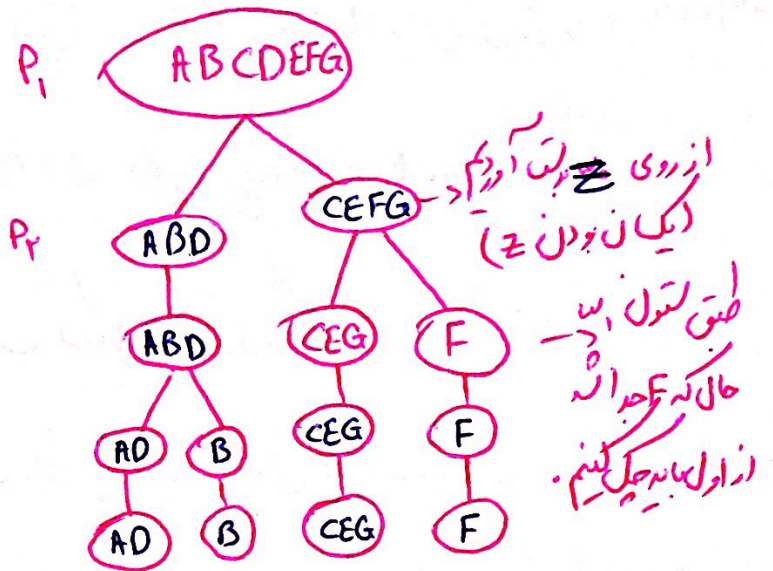
شکل  $P_2$

(۳) در هر بلوک state هایی را که  $k$ -suc آن‌ها در بلوک متفاوت از  $k$ -suc state، دیگر آن بلوک می‌باشد را از آن بلوک جدا می‌کنیم و بلوک جدیدی تشکیل می‌دهیم.

(۴) مرحله ۳ را تا زمانی که بلوک جدیدی ایجاد نشود (دانشی دهیم در پایان state هایی که داخل یک بلوک هستند معادلند.

مثال :

CS	NS		Z
	w=0	w=1	
A	B	C	1
B	D	F	1
C	F	E	0
D	B	G	1
E	F	C	0
F	E	D	0
G	F	G	0



$G$  و  $E$  معادلند و  $D$  و  $A$  نیز معادلند!



جدول معادل :

A	B	C	1
B	A	F	1
C	F	C	0
F	C	A	0

**تمرین:** یک vending machine را در نظر بگیرید که تنها اسکناسهای ۵۰۰ و ۱۰۰۰ تومانی می‌گیرد. برای هر نوشتنی ۱۵۰۰ تومان باید پرداخت گردد و اگر کسی ۲۰۰۰ تومان پرداخت کند ماشین بقیه پول را برای نفرات بعدی پس انداز کند.

**(الف)** حاکم داخل ماشین، پس از گرفتن اسکناسهای ۵۰۰ و ۱۰۰۰ تومانی سیگنال F و T را برای حسگر ۱ می‌کند. مداری طراحی کنید که سیگنال F و T را به ترتیب یک CLK به‌دراز "۱" شدن Sense T و Sense F یک کرده و بعد از آن "۰" کند.

**(ب)** FSM مربوط به این مدار را رسم کنید.

**(ج)** state table را کپی و تعداد state ها را کاهش دهید.

**(د)** state diagram معادل را رسم کنید.

**(ه)** مدار معادل mealy را با state های کمتر رسم کنید.

سؤال:

CS	NS		Z	
	w=0	w=1	w=0	w=1
A	B	C	0	0
B	D	-	0	-
C	F	E	0	1
D	B	G	0	0
E	F	C	0	1
F	E	D	0	1
G	F	-	0	-

- همه عادل don't care می باشند!

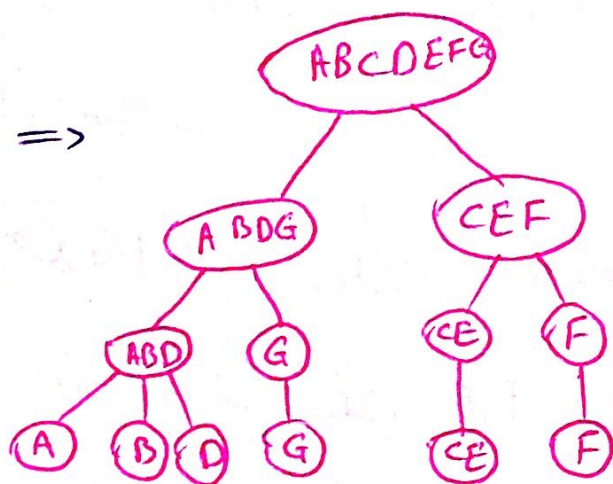
باید حالت های مختلف d های دارای

خودگی بررسی کنیم. (تکلیف d بودن خروجی)

بسیار بهتر از d بودن NS ها است.

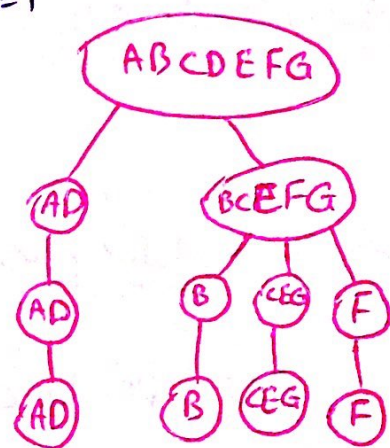
مثلاً فرض کنید

d خروجی = 0  
خروجی



6 state

d خروجی = 1



4 state

\* برای هر d در خروجی باید دو حالت در نظر بگیریم و حالت بسازد و استخراج کنیم.

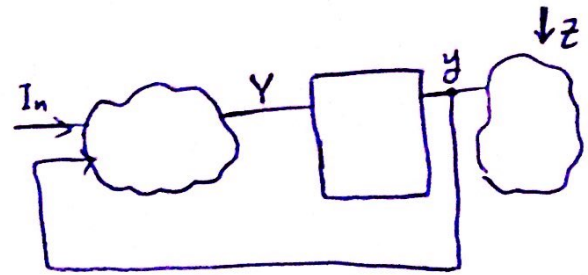
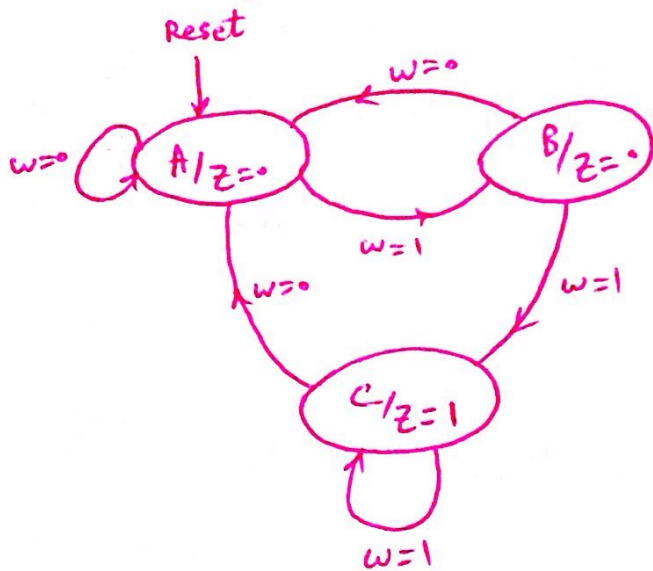
در موارد فوق 4 حالت داریم!

\* اگر از این قسمت قرار باشد که توانایی را امتحان باید مثلاً 2 تا d در خروجی می آید که 4 حالت داشته باشیم.

در ادامه به بررسی چند که Verilog می پردازیم.



Moore type:



Module Moore (clk, w, rst, z);

input clk, w, rst;

output z;

reg [1:0] y, Y;

Parameter A = 2'b00, B = 2'b01, C = 2'b10;

always @ (w, y)

begin

case (y)

A: if (w == 0) Y = A;

else Y = B;

B: if (w == 0) Y = A;

else Y = C;

C: if (w == 0) Y = A;

else Y = C;

default: Y = 2'bxx;

end case

end

always @ (posedge clk)

begin

if (rst == 0)

y <= 2'b00;

else

y <= Y;

end

assign z = (y == C);

endmodule