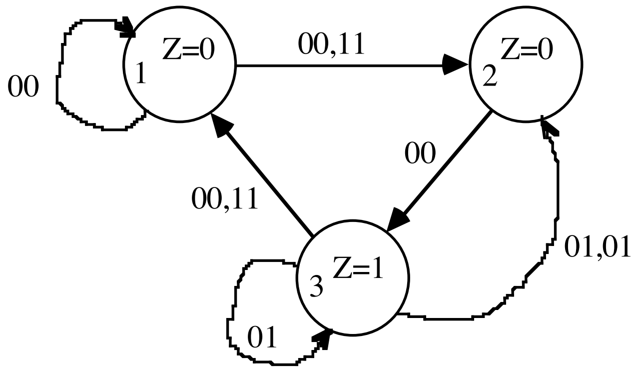
SEQUENTIAL CIRCUIT DESIGN EXAMPLE



Generate flow table from state diagram. Then assign a unique y-state to each row. Note that, since we need at least 2 state variables for 3 rows, and since 2 variables have 4 states, there is an extra state shown below, which is a don't care state, i.e., we never expect the system to be in this state and so we don't care what the output is for it or what the next-state values are for that state. Also, since the input state 10 never occurs, all next-state entries of the corresponding column of the flow table are don't cares. (We usually don't bother showing such don't care rows and columns.)

	AB						
	00	01	11	10	Z	y1	y2
1	1	2	2	-	0	0	0
2	3	2	2	-	0	0	1
3	1	3	1	-	1	1	0
-	-	-	-	-	-	1	1

We can immediately find a logic expression for Z, since Z is a function only of the y's. In effect we already have a truth table for Z on the right side of the above table. It is easy to see that, exploiting the don't care, we can set Z=y1.

Next, we construct a Y-matrix (see below) that specifies Y1 and Y2 values for each position in the table. The Yi signal is the next value of yi. So, in the above table, wherever the next-state is specified at 1, the values of Y1Y2 should be set to the values of y1y2 in state 1, which is 00. Where the next state is 2, since y1y2=01 for state 2, we must set Y1Y2=01. Wherever the next state is 3, we set Y1Y2=11.

	AB						
	00	01	11	10	Z	y1	y2
1	00	01	01	--	0	0	0
2	10	01	01	--	0	0	1
3	00	10	00	--	1	1	0
-	-	-	-	-	-	1	1

Y1Y2

We now have the equivalent of a truth table for Z, T1, and Y2 as functions of A, B, y1, and y2. That is, for every state of A, B, y1, and y2, we have specified values for Y1 and Y2, or indicated that we don't care what those values are. Altho there is usually no reason to recast this table in conventional truth table form, it is shown below just to make the above point clear.

A	B	y1	y2	Y1	Y2
0	0	0	0	0	0
0	0	0	1	1	0
0	0	1	0	0	0
0	0	1	1	-	-
0	1	0	0	0	1
0	1	0	1	0	1
0	1	1	0	1	0
0	1	1	1	-	-
1	0	0	0	-	-
1	0	0	1	-	-
1	0	1	0	-	-
1	0	1	1	-	-
1	1	0	0	0	1
1	1	0	1	0	1
1	1	1	0	0	0
1	1	1	1	-	-

From the Y-matrix, we can see that Y1 is required to be 1 for only two states, ABy1y2=0001 and 0110. So a valid expression for Y1, which would work, is $Y1=\bar{A}\bar{B}\bar{y}1y2+\bar{A}By1\bar{y}2$. But this does not exploit the don't cares in column-10 and row-11. These allow us to add p-terms $A\bar{B}\bar{y}1y2$, $\bar{A}\bar{B}y1y2$, $\bar{A}By1y2$, $A\bar{B}y1y2$. This converts the expression to

$Y1=\bar{A}\bar{B}\bar{y}1y2+\bar{A}By1\bar{y}2+A\bar{B}\bar{y}1y2+\bar{A}\bar{B}y1y2+\bar{A}By1y2+A\bar{B}y1y2$

Now we can unite the first and fourth terms to get $\bar{A}\bar{B}y2$, the third and sixth terms to get $A\bar{B}y2$. Uniting these terms then gives us $\bar{B}y2$. The second and sixth terms can be united to obtain $\bar{A}By1$. Thus Y1 reduces to $\bar{B}y2+\bar{A}By1$.

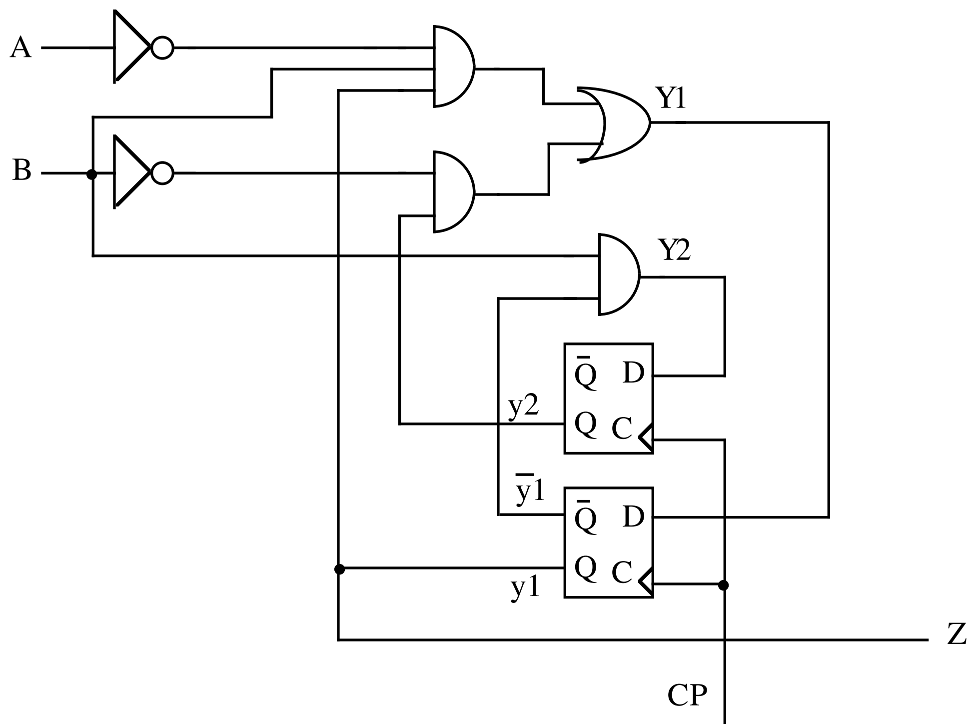
The minterm expression for Y2 is

$Y2=\bar{A}B\bar{y}1\bar{y}2+\bar{A}B\bar{y}1\bar{y}2+AB\bar{y}1y2+AB\bar{y}1y2$.

Repeated use of the uniting theorem reduces this first to

$Y2=\bar{A}B\bar{y}1+AB\bar{y}1$ and then to $Y2=B\bar{y}1$.

The circuit corresponding to these expressions is shown below.



A
B
Y1
Y2
Q̄
D
y2
Q
C
ȳ1
Q̄
D
Q
C
y1
Z
CP