"World Acclaimed" Wisconsin Bowling Ball Waxer Machine

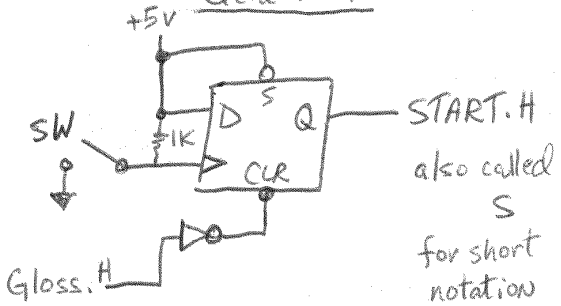
Controller:

Inputs: Start.H (S)
Gloss.H (G)

Outputs: Wax.H (W)
Spin Ball.H (SB)
HEAT.L (H)
Buff.L (B)

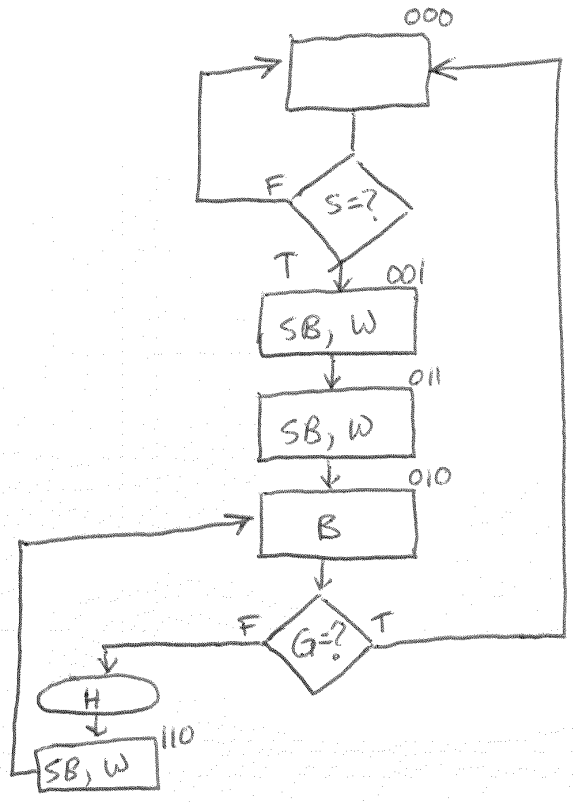
Assumptions:
- when the person puts in their quarters, start goes true
- start stays true for the entire process until the state machine returns to state 000.

Possible Start Signal Generator



- SW is normally closed.
- put in your quarters and SW opens, sets START=H
- Gloss resets the START signal

ASM Diagram



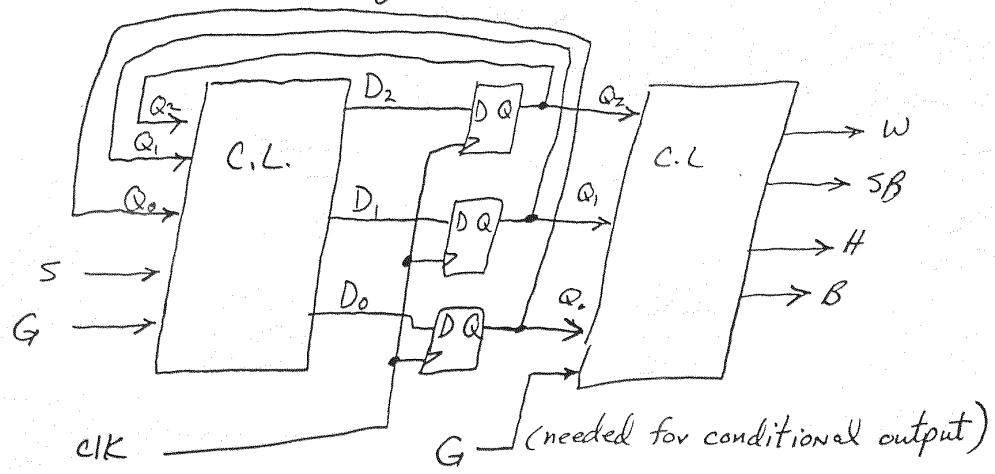
Procedure:

① Determine ASM flow chart (given in this problem)

② Count states ⇒ 5
& compute # of F/Fs needed ⇒ 3

③ Label states on ASM flow chart
- try to label such that only one bit will change for each state transition... this will simplify the state variable equations

④ Functional Block Diagram & select F/F Type



⑤ Next State Table

(See next Page)

5)

inputs		Present State			Outputs				Next state			F/F inputs		
S	G	Q_2	Q_1	Q_0	W	SB	H	B	Q_2^+	Q_1^+	Q_0^+	D_2	D_1	D_0
0	X	X	X	X	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	1	0	0	1
1	0	0	0	1	1	1	0	0	0	1	1	0	1	1
1	0	0	1	0	0	0	1	1	1	1	0	1	1	0
1	0	0	1	1	1	1	0	0	0	1	0	0	1	0
1	0	1	0	0	X	X	X	X	X	X	X	X	X	X
1	0	1	0	1	X	X	X	X	X	X	X	X	X	X
1	0	1	1	0	1	1	0	0	0	1	0	0	1	0
1	0	1	1	1	X	X	X	X	X	X	X	X	X	X
1	1	0	0	0	0	0	0	0	0	0	1	0	0	1
1	1	0	0	1	1	1	0	0	0	0	1	0	0	1
1	1	0	1	0	0	0	0	1	0	0	0	0	0	0
1	1	0	1	1	1	1	0	0	0	1	0	0	1	0
1	1	1	0	0	X	X	X	X	X	X	X	X	X	X
1	1	1	0	1	X	X	X	X	X	X	X	X	X	X
1	1	1	1	0	1	1	0	0	0	1	0	0	1	0
1	1	1	1	1	X	X	X	X	X	X	X	X	X	X

6) Find State Variable Equations

- since S=T we count & when S=F we go to ϕ & stay
- we can remove S from K-Maps and place in equations via inspection

i.e. $D_2 = (\text{equation from K-Map})\,S$

$D_1 = (\text{from K-Map})\,S$

$D_0 = (\text{from K-Map})\,S$

* Also, if our F/Fs had a clear input signal (low true) we could connect S to this and remove from the equations.

"D_2"

Q_1Q_0 \ GQ_2	00	01	11	10
00	0	X	X	0
01	0	X	X	0
11	0	X	X	0
10	1	0	0	0

$$D_2 = (\bar{G}\bar{Q_2}Q_1\bar{Q_0})S$$

"D_1"

Q_1Q_0 \ GQ_2	00	01	11	10
00	0	X	X	0
01	1	X	X	0
11	1	X	X	1
10	1	1	1	0

$$D_1 = (Q_2 + Q_1Q_0 + Q_0\bar{G} + Q_1\bar{G})S$$

check

$Q_2 = 0$, $Q_1 = 1$, $Q_0 = 0$, $G = 1$ } should go to 010

else go 110 when $G = 0$

Q_1Q_0 \ GQ_2	00	01	11	10
00	1	X	X	1
01	1	X	X	1
11	0	X	X	0
10	0	0	0	0

$$D_0 = (\bar{Q_1})S$$

7) Find Equations for Output Variables

- H is only true in state 010 when G = False (Conditional Output)

  thus by inspection $H = \bar{Q}_2 Q_1 \bar{Q}_0 \bar{G}$

W, SB & B are all unconditional outputs

and ∴ only dependent on Q_2, Q_1 & Q_0

"W"

Q_1Q_0 \ Q_2	0	1
00	0	X
01	1	X
11	1	X
10	0	1

$W = Q_2 + Q_0$

↑ check in Asm flow chart

"SB"

Q_1Q_0 \ Q_2	0	1
00	0	X
01	1	X
11	1	X
10	0	1

$SB = Q_2 + Q_0$

B

Q_1Q_0 \ Q_2	0	1
00	0	X
01	0	X
11	0	X
10	1	0

$B = \bar{Q}_2 Q_1 \bar{Q}_0$

↑ also you can get this by inspection

(8A) Physical Implementation w/ Gates

State Eq. $D_2 = \bar{Q}_2 Q_1 \bar{Q}_0 \bar{G} S$, $D_1 = SQ_2 + SQ_1Q_0 + SQ_0\bar{G} + SQ_1\bar{G}$

$D_0 = \bar{Q}_1 S$

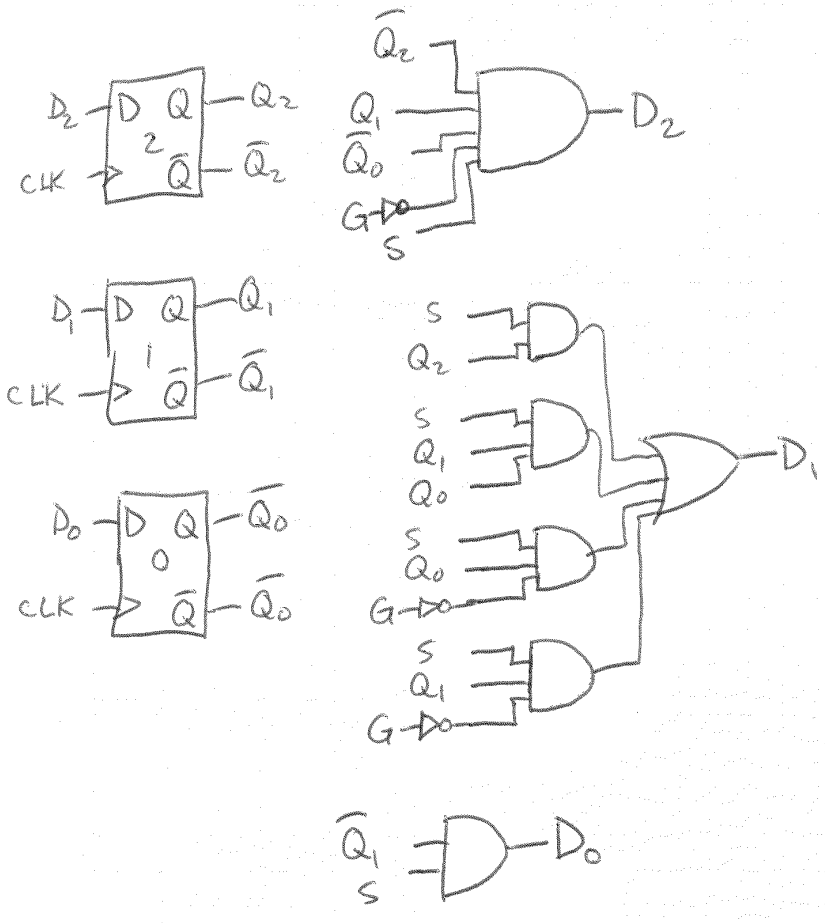
These ARe Logic Equations!

Outputs

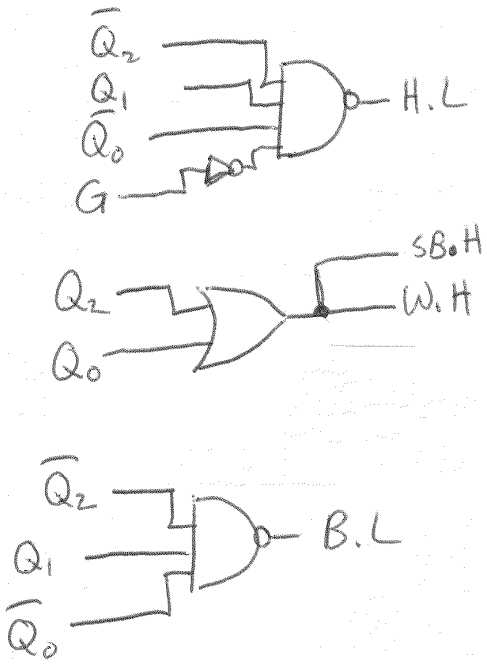
$H = \bar{Q}_2 Q_1 \bar{Q}_0 \bar{G}$, $W = Q_2 + Q_0$, $SB = Q_2 + Q_0$, $B = \bar{Q}_2 Q_1 \bar{Q}_0$

Circuitry

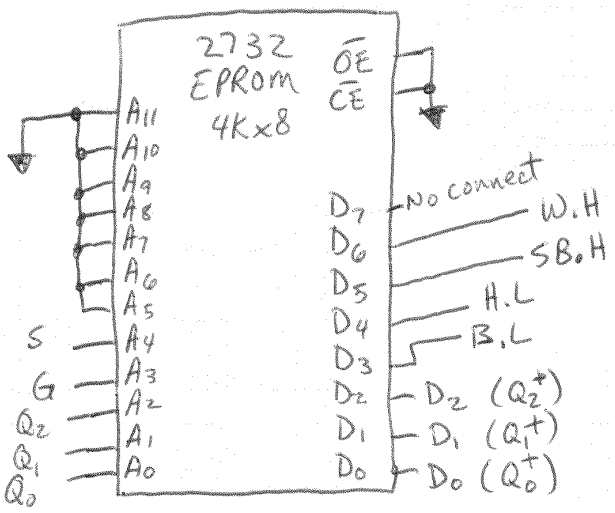
State Generator



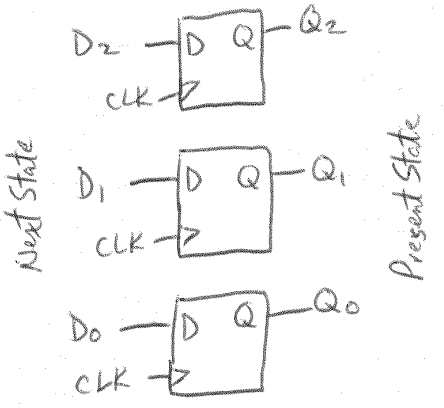
Output Generation



(8B) Physical Implementation w/ EPROM



Flip Flops synchronize the events to a clock (they are still needed!)



Note: we don't need $\bar{Q}$ in F/F now

Memory Contents

read straight out of Next state table

Address Hex Value	Address BINARY $S\ G\ Q_2\ Q_1\ Q_0$	DATA Logic D_6 W	D_5 SB	D_4 H	D_3 B	D_2 D2	D_1 D_1	D_0 D_0	Actual Voltage to be Programmed D_6 W	D_5 SB	D_4 H	D_3 B	D_2 D_2	D_1 D_1	D_0 D_0	Hex Data Programmed
0	0 0 0 0 0	0	0	0	0	0	0	0	L	L	H	H	L	L	L	18
1	0 0 0 0 1	0	0	0	0	0	0	0	L	L	H	H	L	L	L	18
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
0F	0 1 1 1 1	0	0	0	0	0	0	0	L	L	H	H	L	L	L	18
10	1 0 0 0 0	0	0	0	0	0	0	1	L	L	H	H	L	L	H	19
11	1 0 0 0 1	1	1	0	0	0	1	1	H	H	H	H	L	H	H	7B
12	1 0 0 1 0	0	0	1	1	1	1	0	L	L	H	L	H	H	L	06
13	1 0 0 1 1	1	1	0	0	0	1	0	H	H	H	H	L	H	L	7A
14	1 0 1 0 0	x	x	x	x	x	x	x	L	L	L	L	L	L	L	00
15	1 0 1 0 1	x	x	x	x	x	x	x	L	L	L	L	L	L	L	00
16	1 0 1 1 0	1	1	0	0	0	1	0	H	H	H	H	L	H	L	7A
17	1 0 1 1 1	x	x	x	x	x	x	x	L	L	L	L	L	L	L	00
18	1 1 0 0 0	0	0	0	0	0	0	1	L	L	H	H	L	L	H	19
19	1 1 0 0 1	1	1	0	0	0	0	1	H	H	H	H	L	L	H	79
1A	1 1 0 1 0	0	0	0	1	0	0	0	L	L	H	L	L	L	L	10
1B	1 1 0 1 1	1	1	0	0	0	1	0	H	H	H	H	L	H	L	7A
1C	1 1 1 0 0	x	x	x	x	x	x	x	L	L	L	L	L	L	L	00
1D	1 1 1 0 1	x	x	x	x	x	x	x	L	L	L	L	L	L	L	00
1E	1 1 1 1 0	1	1	0	0	0	1	0	H	H	H	H	H	H	L	7E
1F	1 1 1 1 1	x	x	x	x	x	x	x	L	L	L	L	L	L	L	00

Note: X will be programmed "L"

The end!