Each iteration of the sequence involves 6 steps; at each step, the ASM checks to see whether the input is as specified, and if not, it goes into an alternative, error-handling sequence. If a complete correct sequence occurs, the counter is decremented if it is non-zero. If not, it is incremented. State 0 acts as a “wait for input” phase.
Notice that every combination of values for \textit{clap} and \textit{stomp} (00, 01, 10, and 11) occurs in the “conditions” column. Accordingly, I have treated (clap, stomp) as a two-bit number and used a decoder (with active-high outputs) to generate a TRUE output corresponding to each of the possible combinations of values of clap and stomp.

This simplifies the circuit somewhat. You are \textit{not} required to take this approach in your solution.
TF Count = 0
clock
dec
reset
inc
dec
2-bit register
D3
D1
D2
D0
Q3
Q1
Q2
Q0
BF
C
Dn
An
Bn
Cn
Dn
AP
DP
T
clap stomp
3 2 1 0
decoder