TM to recognize strings in (0 + 1)* with twice as many 1's as 0's

Concept:
Find a 0 and X it out.
Find two 1's and X them out.
If you can't find two 1's, there are too few 1's
When all the 0's are gone, verify all the 1's are gone also.

State 0: Find a 0
Always starts at left end of tape. Move head right looking for a 0.
f hit a 0, replace it with X, go to state 1
If hit right end (see a blank) go to verify state 5.

State 1: Go back to start of tape
Go left till get to left end of tape. Go to state 2.

State 2: Find a 1
Always starts at left end of tape. Move head right looking for a 1.
If hit a 1, replace it with X, go to state 3

State 3: Find another 1
Keep moving head right looking for a 1.
If hit a 1, replace it with X, go to state 4

State 4: Go back to start of tape
Go left till get to left end of tape. Go to state 0.

State 5: Verify no leftover 1's
Go left as long as you see X.
If you see blank go to state 6.

State 6: Accept

Note: States 2 and 3 will have no transition specified on blank. If you hit a blank this means you have run off the right end of the tape without finding two 1's to match the 0 found in state 0. This is how strings with not enough 1's are rejected.

Strings with too many 1's are rejected if state 5 hits a 1 while scanning back to the left end. At state 5, all 0's have been X'd out, so the tape should be all X's unless there were too many 1's.

This machine will accept empty string. The original assignment said there should be at least one 0. It will be ok to have no 0's.