

COMP SCI 7411 Event Driven Computing Practice 1 Plan

Tinson Lai
a1812422

August 15, 2020

1 Locking

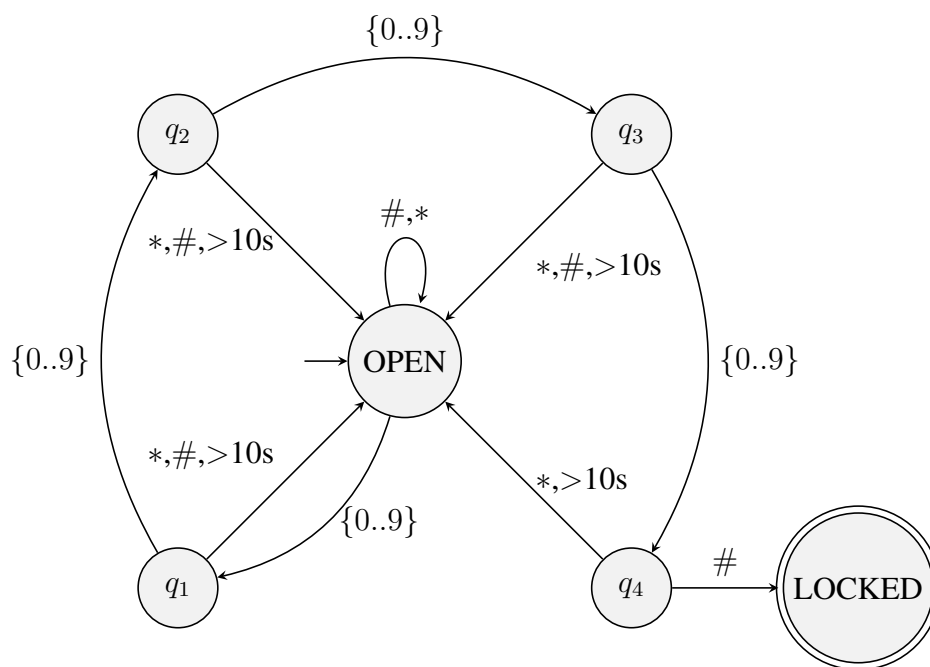


Figure 1: Locking Process

2 Unlocking

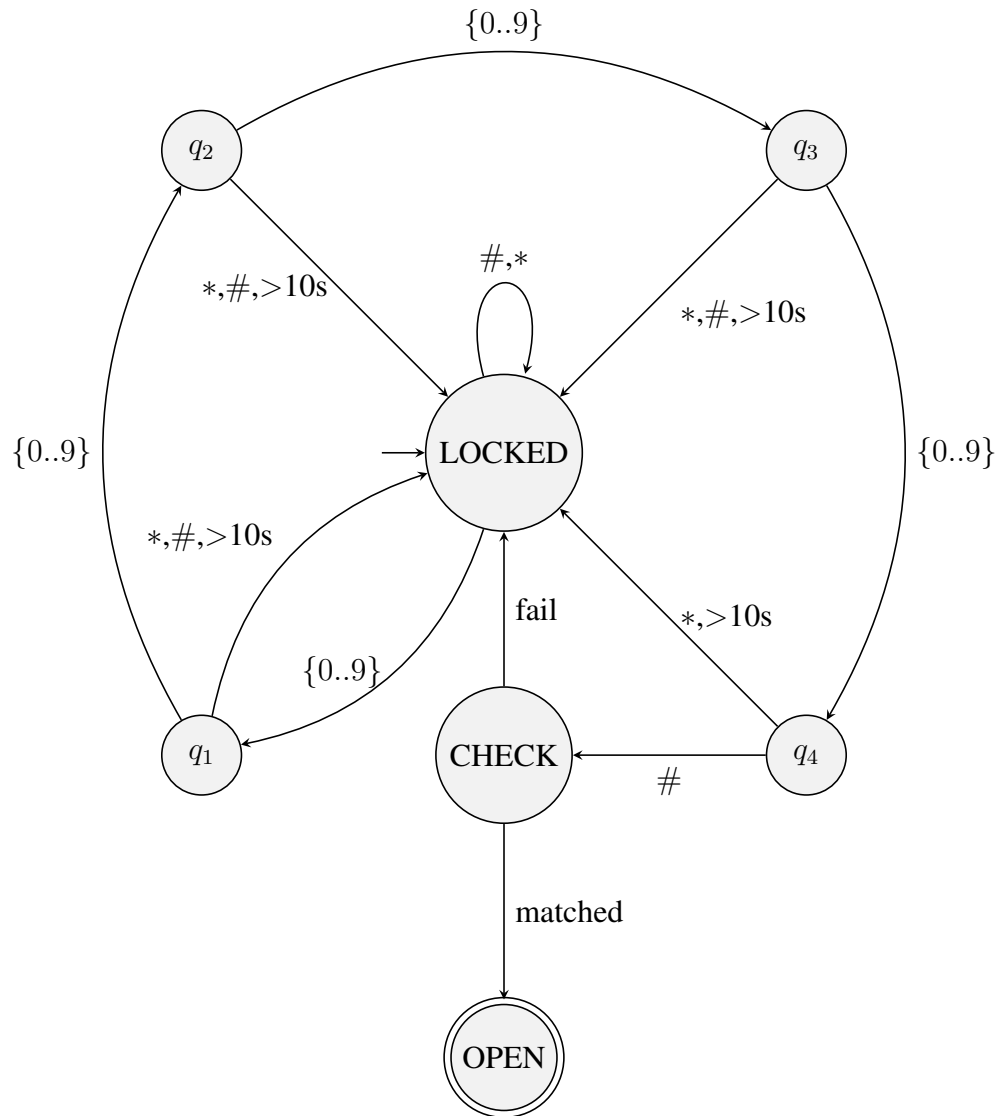


Figure 2: Unlocking Process

3 Notations

The designs above have several notations:

- Intermediate states within the digits entering process are named q_i where i represents the number of digits shown in the panel display.
- $\#$ means pressing the key $\#$ once, similar for $*$. $\{0..9\}$ means pressing one of the numeric keys on the panel once.
- q_1 will also start a timer.
- q_4 will lock all numeric keys so there are only two possible inputs ($*$ and $\#$).

- The CHECK state in Figure 2 is a special intermediate state which are not visually presented to the user. It just depicts a state where the system has to check for password correctness.
- ">10s" means the timer has recorded a duration longer than 10 seconds. An alert will be delivered to the user and the input will be cleared as required.

4 Timer

The timer started in state q_1 in Figure 1 and 2 can be interrupted by $*$ or $\#$ as described previously. Any subsequent inputs of digits following the first digit will not affect the status of the timer. Thus the diagram will be as simple as Figure 3.

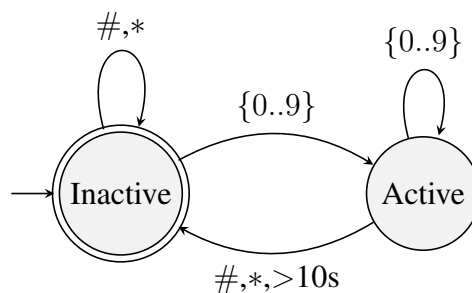


Figure 3: Timer

5 Disabling Numeric Keys

Another design choice is disabling numeric keys in the panel when q_4 is reached.

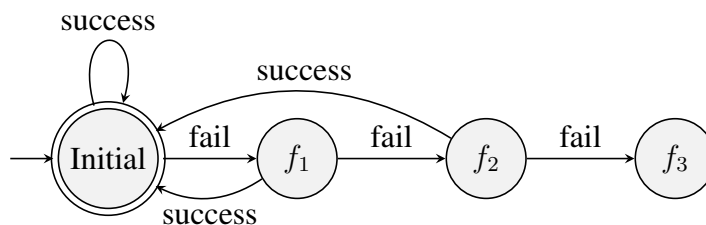


Figure 4: Numeric Keys Disabling

A permanent lock will also be applied if the user fails to enter the correct password three times and it's not possible to input digits anymore, and this is show as the state f_3 in Figure 4.

6 Notes

Although $\#$ and $*$ seem to have identical behaviour in the state diagram in the state diagram, in the actual implementation, I tried to display different alerts to the user if they press $\#$ without an appropriate input. For example, an alert warning the user about incorrect password etc.