- ❖ **<u>MPU interface Section</u>:**
    - This section includes eight bidirectional data lines (DB0-DB7), one Interrupt Request line (IRQ), and six lines for interfacing, including the buffer address line (A_0).
    - When A_0 is high, signals are interpreted control words or status; when A_0 is low, signals are interpreted as data.
    - The IRQ line goes high whenever data entries are stored in the FIFO.
    - This signal is used to interrupt the MPU to indicate the availability of data.
- ❖ **<u>Key Debounce:</u>**
    - When a mechanical pushbutton key, shown in figure (a), is pressed or released, the metal contacts of the key momentarily bounce before giving a steady-state reading, as shown in figure (b).

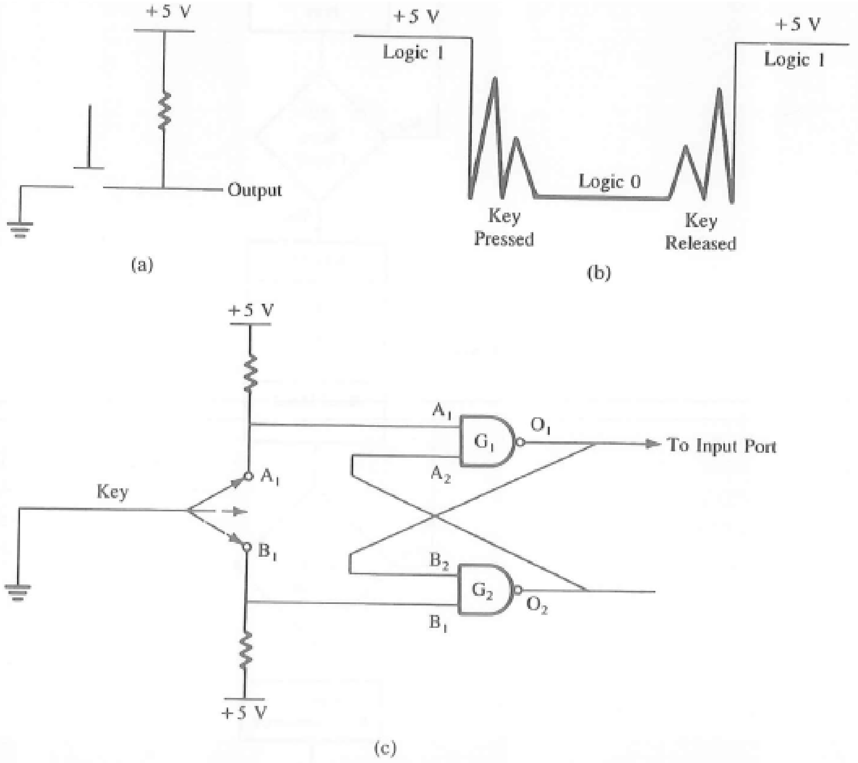

Figure 3.11 (a) Pushbutton Key, (b) Key Debounce and (c) Key Debounce Circuit Using NAND Gates

- Therefore, it is necessary that the bouncing of the key should not be read as in input.
- The key bounce can be eliminated from input data by the key-debounce technique, using either hardware or software.
- Figure (c), shows a key debounce circuit.
- In this circuit, the outputs of the NAND gates do not change even if the key is released from position A_1.

- The outputs change when the key makes a contact with position B1. When the key is connected to A_1, A_1 goes low.
- If one of the inputs to gate G_1 is low, the output O_1 becomes 1, which makes B_2 high.
- Because line B_1 is already high, the output of O_2 goes low, which makes A_2 low.
- When the key connection is released from A_1, it goes high, but because A_2 is low the output doesn't change.
- When the key makes contact (+5V) to another contact (ground), the output does not change during the transition period, thus eliminating multiple readings.
- In the software technique, when a key closure is found, the microprocessor waits for 10 to 20 ms before it accepts the key as an input.
- The delay routine is as follows:

```
                PUSH B
                PUSH PSW
                LXI B, COUNT
LOOP:           DCX B
                MOV A,C
                ORA B
                JNZ LOOP
                POP PSW
                POP BC
                RET
```

Interrupt controller interface [8259] :

- ❖ The 8259A is a programmable interrupt controller designed to work with microprocessors 8085, 8086 and 8088.
- ❖ The 8259A interrupt controller can
  1. Manage 8 interrupts according to the instructions written into its control registers. This is equivalent to providing eight interrupt pins on the processor in place of one INTR (8085) pin.
  2. Vector an interrupt request anywhere in the memory map. However all eight interrupts are spaced at the interval of either four or eight locations. This eliminates the major drawback of the 8085 interrupts in which all interrupts are vectored to memory locations on page 00H.
  3. Resolve eight levels of interrupt priorities in a variety of modes, such as
     - Fully nested mode
     - Automatic rotation mode
     - Specific rotation mode
  4. Mask each interrupt request individually.