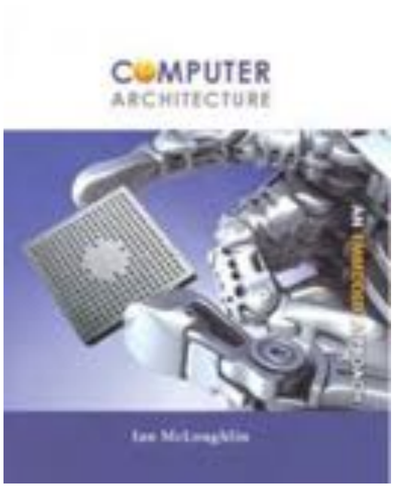
Computer Peripherals

School of Computer Engineering
Nanyang Technological University
Singapore

These notes are part of a 3rd year undergraduate course called "Computer Peripherals", taught at Nanyang Technological University School of Computer Engineering in Singapore, and developed by Associate Professor Kwoh Chee Keong. The course covered various topics relevant to modern computers (at that time), such as displays, buses, printers, keyboards, storage devices etc... The course is no longer running, but these notes have been provided courtesy of him although the material has been compiled from various sources and various people. I do not claim any copyright or ownership of this work; third parties downloading the material agree to not assert any copyright on the material. If you use this for any commercial purpose, I hope you would remember where you found it.

Further reading is suggested at the end of each chapter, however you are recommended to consider a much more modern alternative reference text as follows:



Computer Architecture: ***an embedded approach***
Ian McLoughlin
McGraw-Hill 2011

Chapter 4. Keyboards

Input devices can be classified into 2 groups:

1. User interface devices for human users to control and provide data to the system. Dealt with in this course.
2. Non user devices like temperature sensors, limit switches, A/D converters and network interfaces. Normally the scope of courses on instrumentation and control, or data communications.

4.1 USER INPUT DEVICES

- Keyboards
- Mice
- Scanners
- Digitizers
- Touch Screens / Pen Input

COMPARISON OF DATA ENTRY TECHNIQUES

Method	TIME for 20 chars.	Error rate	Label size	Label cost	Reader cost	Advantages	Disadvantages
Manual	10 Sec.	High	0.4 x 2.2	Low	Low	Low cost	Human reader, Slow, Inflexible
OCR	4 Sec	Medium	0.5 x 2.5	Low	High	Human readable	Poor error rate, Inflexible
MICR	Machine Scanned	Low	0.5 x 2.5	Medium	High	Human readable	Expensive, Inflexible
Magnetic Strip	4 Sec	Low	0.4 x 1.0	Medium	Medium	Updateable	Inflexible, contact reading
Voice Recognition	20 Sec	High	0.4 x 2.2	Low	High	No hands	Human user, "Training"
Machine Vision	Machine Scanned	Depends	Variable	Variable	Very High	Integrated System	Expensive, special use only

Bar Code	4 Sec	Low	0.6 x 2.5	Low	Low	Flexible printers and readers	

4.2 Introduction to Keyboards

Of the various devices available for transmitting information from the human to the computer, the switch is the simplest form of input device. However, the keyboard is certainly the most generally applied. It is clearly the device of choice for applications involving significant amounts of textual input. It is also the computer input technology most affected by historical precedent. The data entry keyboard achieved some degree of design standardization long before its linkage to the computer. While this linkage has had a substantial effect on the design of current keyboards, the functionality, layout, mechanics, and dimensions of these devices are still in many ways a reflection of nineteenth-century technology.

4.3 CLASSIFICATION

1. FIXED-FUNCTION
2. VARIABLE FUNCTION (PROGRAMMABLE)
3. COMBINATION

4.3.1 FIXED-FUNCTION

Each key shall be dedicated to a single function. Examples such as handheld calculators and touch tone telephones all employ fixed-function keyboards. Fixed-function keyboards have several advantages relative to variable-function keyboards:

- Simplicity of operation—typically only one key is pressed at a time and the same function is always performed by the same key.
- All available functions can be determined by scanning the keys.
- Relatively little software support is necessary.
- Keys can be arranged in logical groups.

Disadvantages of fixed-function keyboards include:

- A large number of required functions requires a large number of keys.
- Frequent visual search and arm and hand movement may be required over a large area.
- Changes require hardware modification.
- It may be difficult to group keys logically for all operating procedures.

The selection of a fixed-function keyboard is appropriate when one set of functions is frequently employed, when functions must be executed quickly, and when correct initial selection of functions is critical to satisfactory operation of the system.

4.3.2 VARIABLE FUNCTION (PROGRAMMABLE)

The functions of variable-function keys are generally varied in one of two ways: mode change keys may be used, permitting the user to vary a key's function among a small number of alternatives, or the functions of the keys may be placed under software control with the user informed of the key function relationships via a labeled keyboard overlay or an associated video display unit. Common applications of variable-function keyboards include uppercase/lowercase typewriters, ATM keyboards, touch screen keyboards and advanced hand-held calculators with mode change keys.

General advantages of variable-function keyboards include:

- Fewer keys are needed relative to a fixed-function keyboard of equivalent power.
- Less visual search and arm and hand movement are required.

Software controlled key-function relationships have several additional advantages:

- The operating procedure and sequence can be guided by programmed instructions displayed on the associated video display.
- Changes require software rather than hardware modification.
- Labels can be logically grouped for each operating procedure.
- Keys for functions inappropriate to the current situation may be made inactive.

There are also certain disadvantages associated with variable-function keyboards:

- With mode change keys, the user must press more than one key to execute some of the functions. This additional complexity increases entry time and number of errors. It also becomes difficult to label the keys clearly with their multiple functions and to group labels logically for all procedures.
- With software controlled function assignment, the user must select the currently appropriate function-to-key assignment. It is, therefore, likely that he or she will require more training than would the user of a fixed-function keyboard. The training problem is complicated further when the user employs a variety of software packages. There can easily be a lack of equivalence in function-to-name and function-to-key assignments among the different software packages. Thus, a given function may go by different names, a given name may represent different functions, and a given function may be associated with different keys across the software packages.

The selection of variable-function keyboards appears to be appropriate when there are frequently used subsets of functions.

4.3.3 COMBINATION

Typical computer keyboard with programmable function keys and some automatic teller machines employ the combination keyboards.

Fixed function keys for frequently used keys as in the alphabet and number keys and variable function keys to provide flexibility in for frequently used sequences, special functions, macros.

4.4 KEYBOARD LAYOUT

The general considerations apply to the layout of keys on a provide a detailed set of guidelines for keyboard layout:

1. Determine the characters and numbers of keys required.
2. Arrange the keys according to their frequency of use and user characteristics. The most frequently used keys should be assigned to the stronger fingers. To enhance keying speed, the keys should also be arranged to maximize alternation of key presses between hands.
3. Follow historical precedent.
4. Follow established standards.
5. Group frequently used keys under the resting position of the hand where the user can determine their locations by touch.
6. Group related functions together.
7. Group logically and according to sequence of use.
8. Locate according to importance.
9. Code the keys so that the user can easily locate important or frequently used keys and key groups. In addition to key labels, keys can be coded by variations of shape, color, surface texture, and spacing.
10. Consider all factors, including the intended applications, costs, and manufacturing requirements.

4.4.1 THE QWERTY LAYOUT

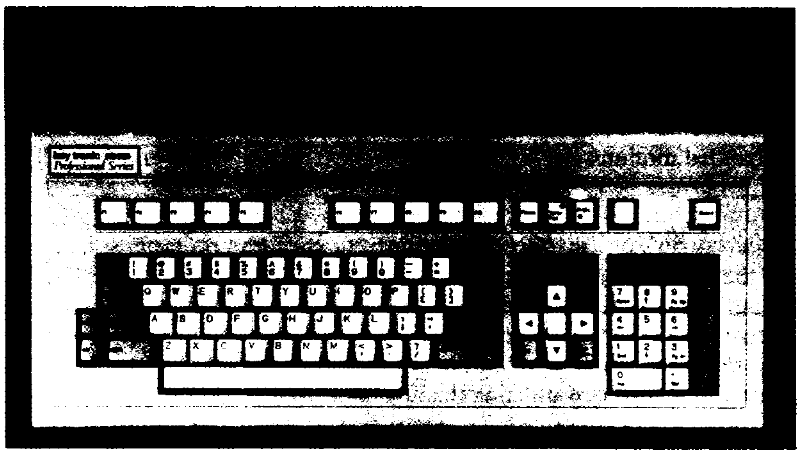


Figure 0-1 The QWERTY keyboard layout

The QWERTY keyboard layout has been adopted as an ANSI standard for alphanumeric keyboard arrangement. Some of the criticisms directed at the QWERTY layout over the past 50 years include:

- It overloads the left hand—57% of typing is carried out by the non-preferred hand for the majority of the population.
- It overloads certain fingers. (The differential strength of fingers, however, is perhaps less an issue with today's electronic keyboards than it was with earlier manual typewriters.)
- Too little typing is carried out on the home row of keys (32%). Too much typing (52%) is carried out on the top row. (Most critics of the QWERTY layout have assumed that home-row keying is the fastest. However, that while this may have been so on manual typewriters, top-row keying appears to be fastest for skilled typists on electric typewriters.)
- Excessive row hopping is required in frequently used sequences, often from the bottom row to the top row and down to the bottom again.
- Many common words are typed by the left hand alone.
- Forty-eight percent of all motions to reposition the fingers laterally between consecutive strokes are one-handed rather than easier two handed motions.

4.4.2 DVORAK LAYOUT

Of the many efforts to improve upon the QWERTY layout, the Dvorak layout has been proven to be the most enduring. A variant of this layout has, in fact, been accepted by the American National Standards Institute as an alternative standard in 1983.

Figure 0-2 Dvorak Layout

The Dvorak Simplified Keyboard was arranged on the basis of the frequencies with which letters and letter sequences occur in English text. It was designed to the following criteria:

- The right hand was given more work (56%) than the left hand (44%).
- The amount of typing assigned to different fingers was proportional to their skill and strength.

- Seventy percent of typing was carried out on the home row—the most frequently used letters were arranged on this row. Only 22% and 8% of typing was carried out on the top and bottom rows, respectively.
- Letters often occurring together were assigned positions so that alternate hands could strike them.
- Finger motions from row to row and difficult, awkward reaches from the home row were minimized.
- Thirty-five percent of the words typically used were typed exclusively on the home row.

4.4.3 COMPARISONS

- It has been claimed that, relative to the QWERTY layout, the Dvorak layout is easier to learn, is less fatiguing to use, and permits faster data entry with fewer errors.
- Because the Dvorak layout was designed to minimize the awkward finger movements that commonly occur in QWERTY typing, it is quite possible that it would achieve reduced key-to-key times relative to those achieved with the QWERTY layout. A daily production advantage up to 10% has been estimated.
- Alphabetical layout is helpful only to unskilled users, and offer no advantage once the user has some practice.
- Future development might include DWIM ("Do what I mean") error correction facility, as in some Chinese word processors.

4.5 Keyboard Requirement and Feedback

4.5.1 Keyboard Life Requirements

An important specification required in selecting or developing a keyboard is keyboard life, usually specified as the number of key actuations made on each key without failure. This specification is important because if it is set too low, the keyboard is likely to fail prematurely in the working environment. If set too high, keyboards may be unnecessarily costly when certain components are designed to accommodate the extended life requirement. An important objective, then, is to match the keyboard life requirement to the needs of the application in question.

Keyboard life requirements for keyboard intensive applications such as word processing and electronic typewriters are typically in the range of 20 million actuations per key. "Worst case" estimates for design life are based on assumptions. For example, estimated keyboard usage is approximately 120 hours per month at an input rate of 12,000 keystrokes per hour (approximately equivalent to 40 words per minute). Assuming that the "E" key (the most frequent letter in English text, which constitutes about 13.8% of all typed characters) will be the limiting factor at 20 million keystrokes, it is estimated that the alphabetic keyboard array will receive approximately 145 million keystrokes. At the assumed usage rate of 12,000 characters per hour for 120 hours per month, the calculated keyboard life would be approximately 8.4 years.

Most graphic workstation users will seldom maintain equivalent rates for sustained periods, it is likely that more modest keyboard life requirements will be acceptable. Keyboards on most popular personal/ professional microcomputers have life cycle requirements of 10 million cycles per key. Non-contacting keyswitches have higher ratings of 100 million cycles.

4.5.2 Feedback

Gathering feedback is very important.

4.5.2.1 TACTILE FEEDBACK

- Light / Heavy
- Linear resistance
- Increasing resistance
- Non-linear (hysteresis) resistance
- Full / Limited Travel Keyboard

The return mechanism of a keyswitch is responsible both for restoring the key upon release to its uppermost position and for providing resistance to the downward motion of the key as it is being pressed. This resistance varies as a function of key displacement and is a major determinant of the tactile feel of the keyboard. Key activation forces from 0.25 to 1.47 N and total key displacements between 1 to 6 mm appear to be preferred by operators. Typical production keyboards for computer input have key activation forces ranging from 0.4 to 1.2 N with key displacements between 3 and 5 mm.

The studies that deal with the design of workplace and equipment and human feeling have becoming a field of research known as human factor or human ergonomics.

4.5.2.2 AUDIO & VISUAL FEEDBACK

- Mechanical noise
- Generated "beep"
- Video display of key entered.

The major source of feedback for the highly skilled user of a full travel keyboard appears to be the kinesthetic-proprioceptive-tactual feedback that the user receives by actually making the movement and striking the key. Visual feedback is important during training and for the correction of errors. Greater amounts of visual feedback permitted the users to review their performance and correct their errors. Generated “beep” is commonly used as warning for illegal character of keyboard buffer full.

4.6 KEYSWITCH TECHNOLOGY

4.6.1 Mechanical Contact

The simplest switch mechanism employs mechanical contacts that are brought together as the key plunger is depressed (Figure 0-3 An example of a mechanical contact keyswitch.). Although low in cost, one problem with mechanical contacts is that they suffer from contact bounce, which can generate false signals. This problem usually necessitates the use of appropriate debounce circuitry. Another disadvantage of mechanical contact switches is that contacts can be contaminated by particles or oxidation. Such contamination results in increased contact resistance, causing unreliable operation or even total switch failure. Key life is in the range of 5 to 10 million actuations, but survival rate often drops rapidly above 5 million operations.

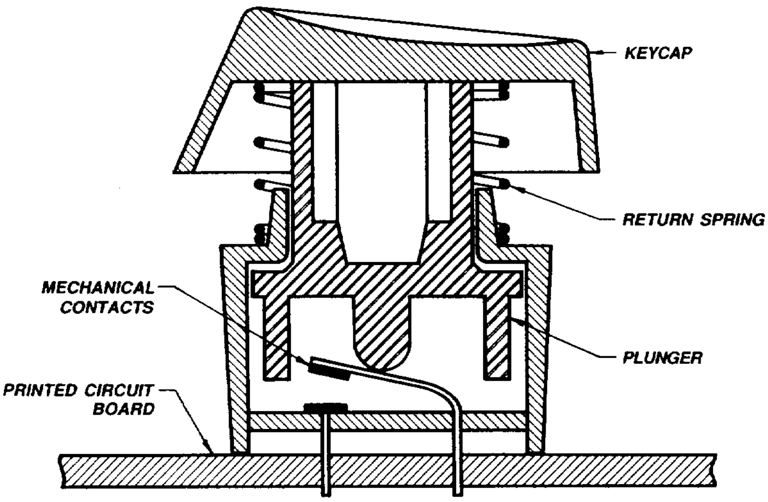


Figure 0-3 An example of a mechanical contact keyswitch.

4.6.2 Full Travel Membrane

Although essentially a contact switch technology, full travel membrane keyboards present a low cost, reliable keyboard alternative. A matrix of full travel (approximately 4 mm) key button assembly is mounted above a three-layer sandwich of circuit materials that comprise the switching mechanisms for all keys on the keyboard (Figure 0-4). The top and bottom layers of the sandwich are thin sheets of flexible film (typically, clear polymer) with a conductive silver composition deposited onto facing sides of the flexible film. The silver composition is "printed" in a pattern that forms the electrical contacts for each key station and all electrical interconnections. Between the conductive layers is a third layer that provides a gap between contacts through holes in the layer. The entire three-layer switch assembly is bonded to a sheet metal base plate. When a key is pressed, a switch closure is accomplished by deflecting the top film layer through the hole in the separation layer so that the conductive material on the top film layer comes in contact with the conductive material on the bottom layer. Typical key life characteristic for full travel membrane keyboards now exceed 10 million keystrokes. Unfortunately, this type of keyboards also suffers from switch bounce.

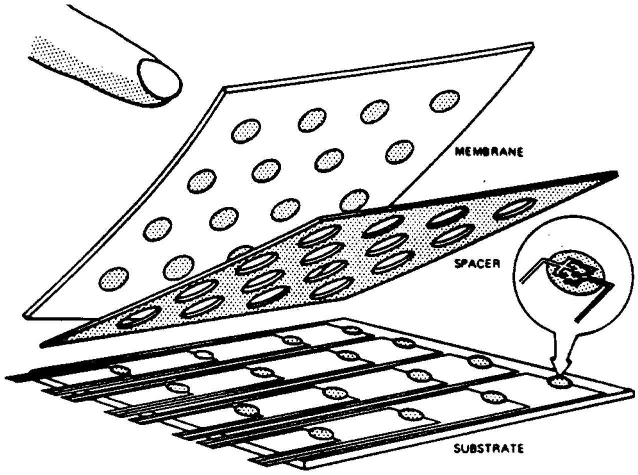


Figure 0-4 Exploded view of a full travel membrane keyboard showing unitized frame, space layer, adhesive layers, and base plate.

4.6.3 Capacitance Switches

One of the most popular keyboard technologies uses capacitance changes to signal key activation. In its most common form, the capacitance switching mechanism (Figure 0-5 An example of a capacitance keyswitch.) employs a foam pad that is bonded to the bottom of the key plunger. A sheet of conductive foil is attached to the bottom of the foam pad. As a key is pressed, the foil, which acts as a capacitive plate, is brought into contact with a printed circuit board mounted under the array of key plungers. The increase in capacitance as the foil contacts the circuit board passes a low-level signal onto detector circuits, which signals key activation. Because capacitance switching does not depend on the closing of a circuit through low resistance contacts, capacitance keyboards exhibit high reliability. Typical key life characteristics range from 10 to 50 million keystrokes. Additional circuits and AC signal needed to sense switch closure.

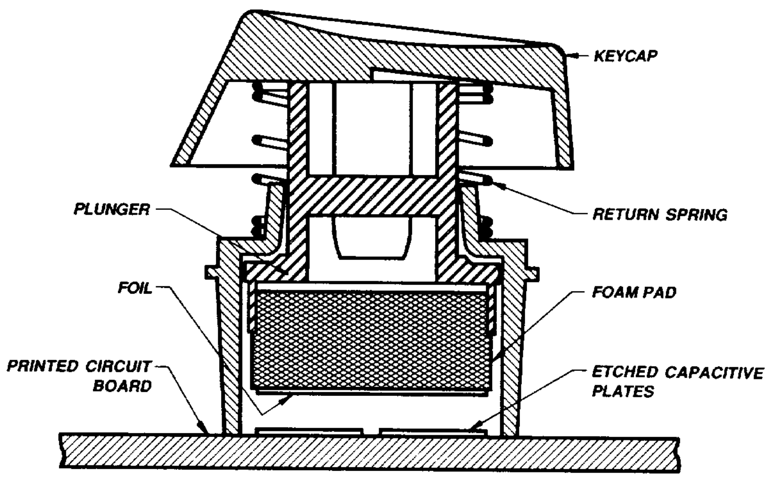


Figure 0-5 An example of a capacitance keyswitch.

4.6.4 Hall effect

In Hall effect keyboards, a magnetic transducer (typically an integrated circuit) is mounted near a magnet, which is affixed to the key plunger. Depression of a key moves the magnet closer to the magnetic transducer, causing the transducer to output a voltage change signalling key activation. Because the switching mechanism employs solid state electronics, Hall effect keyboards exhibit extremely high reliability. Key life of more than 100 million actuations per key is typical. In addition, because individual keyswitches can be sealed, Hall effect keyboards are often used in military, marine, industrial, and aerospace applications where reliability is a critical factor. The disadvantages of Hall effect switches are relatively high cost and large size of individual keyswitch modules.

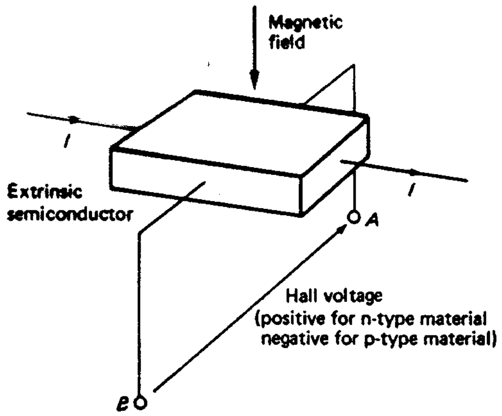


Figure 0-6 Hall Effect

4.6.5 Reed

Reed switches are essentially mechanical contacts, problems caused by contamination are eliminated by enclosing the switch contacts in a sealed capsule, which is usually made of glass. Depression of a key moves a magnet close to the encapsulated reeds. The attraction of the magnet on the reeds causes them to be brought together until the contacts touch. Because the sealed capsule prevents contamination of the switch contacts, the long-term reliability is generally good. Key life of 50 million operations is possible. The biggest disadvantage of reed switches is their high cost relative to other technologies.

4.6.6 Dome Switch

One popular switch technology is the dome switch. As shown in the cross section of Figure 0-7 Metal Dome switch, the metal dome provides a spring action, and when pressed it snaps to the closed position. This provides some travel and good tactile response. Dome switches are available as a single push button and in a keyboard matrix. A combination of membrane and metal dome technologies is widely used. It combines the low cost and environmental immunity of the membrane technology with the tactile response of the dome switches.

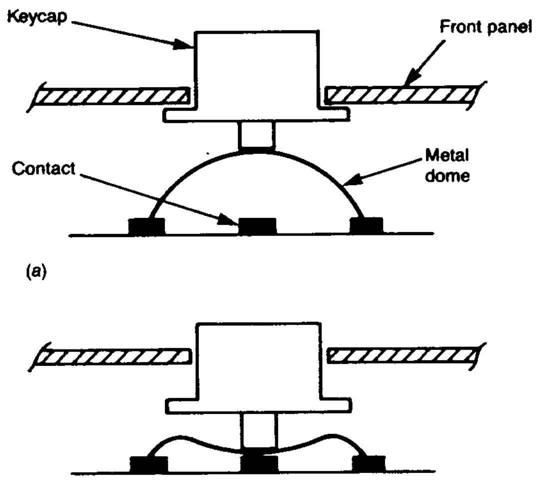


Figure 0-7 Metal Dome switch

4.7 Debouncing Methods

There is a problem which exists in mechanical switches, due to their construction, is 'contact bounce'. Rather than obtaining a *single* clean pulse output, a series of pulses results due to the switch contacts not coming to rest immediately. As shown in Figure 0-8 Contact bounce., a single physical push of the button results in multiple electrical signals being generated and sent to the RESET input of the CPU. The response time of the switch is several orders of magnitude slower than that of the computer (tens of milliseconds, rather than microseconds), so that the computer could read the single switch closure many times over during the time the switch is operated, interpreting each LO signal as a new input whereas in fact it is the same one all the time.

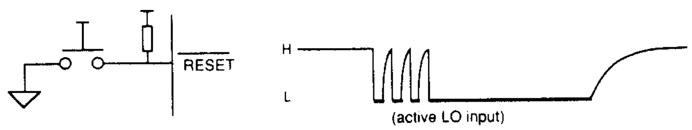


Figure 0-8 Contact bounce.

Contact bounce can be overcome in practice either by one of the techniques illustrated in Figure 0-9 Remedies to overcoming contact bounce., by synchronizing the switch to the system clock, or by using software debouncing.

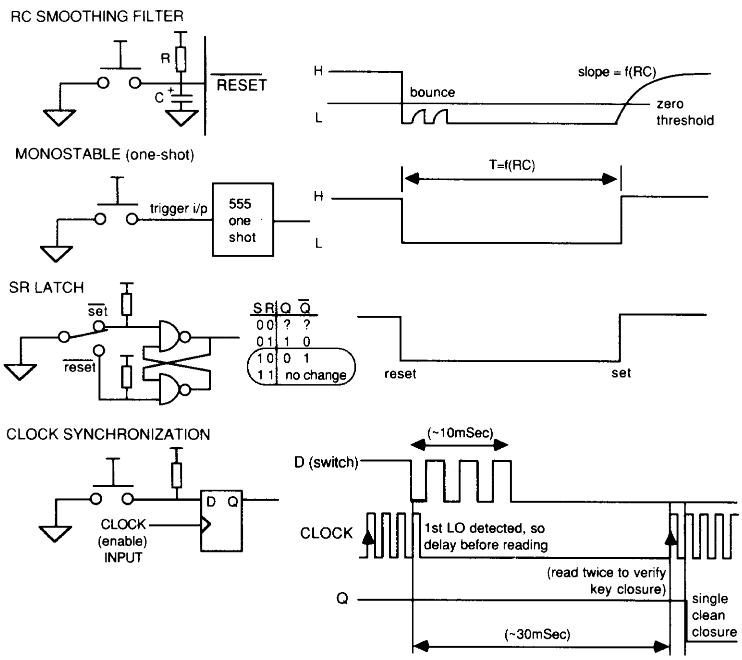


Figure 0-9 Remedies to overcoming contact bounce.

4.7.1 The smoothing filter:

The RC time constant of the integrator (smoothing filter) determines the rate at which the capacitor charges up towards the supply voltage once the earth connection via the switch has been removed. So long as the capacitor voltage does not exceed the zero threshold value, then the RESET input will continue to recognize a logic 0.

4.7.2 A monostable

A monostable or one-shot is a dedicated hardware timer chip which produces a fixed pulse width output in response to the first LO signal applied to its input; it ignores all subsequent LO inputs from the switch until it has finished producing its own pulse output.

4.7.3 SR latch

Like the one-shot, the SR latch only responds to the first LO signal applied to its Reset input, and ignores all subsequent (spurious) LOs applied to it.

4.7.4 Clock Synchronization

Synchronizing the switch input requires the use of an external D-type latch, and results in additional software overhead, in order to sample or poll the switch after it has had time to come to rest.

4.7.5 Software Debouncing

The basic software debounce algorithm is to keep a count of the length of time each key has been pressed or released. Only when the count exceeds a predetermined value (such as 10 ms) is the switch considered to have been pressed or released. This is easily implemented with an interrupt-driven multiplexed interface. When a key is detected, software stores the key code in a temporary location. If this key is still pressed the next time that row is read, then the switch is considered to be pressed, and the key code is passed to the appropriate service routine. If the bounce time is longer than the scan time, then the number of scans that the key has been pressed (or released) must be counted, and only when the count reaches a predetermined value is the key code passed to the processing routine.

4.8 KEYBOARD INTERFACING

Keyboards and other switches can be interfaced to the microprocessor in several ways. Some methods are preferable for small numbers of switches, and others for larger numbers. The software burden imposed on the processor for monitoring the keyboard also varies with the interfacing approach. This section describes a variety of approaches and their advantages and disadvantages.

4.8.1 NON-MULTIPLEXED INTERFACE

The simplest technique for interfacing a small number of switches to a microprocessor system is to connect each to an input port bit. Figure 0-10 shows eight switches connected in this way. Pull-up resistors provide a high level when a switch is open, allowing single-throw switches to be used. When a switch is closed, the input is held low, and the corresponding bit is 0 when the microprocessor reads the input port. This simple interface requires very little software. It is often used to read system configuration (WP) switches. It can a so be used for small numbers of keyboard switches.

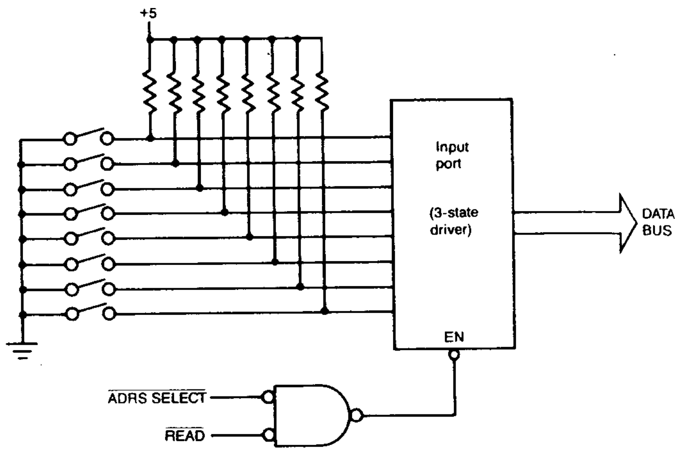


Figure 0-10 Basic non-multiplexed interface

4.8.2 MULTIPLEXED SWITCH INTERFACE

For large numbers of switch positions, an array of push-button switches provides a much more efficient solution than either dedicated or multi-position switches. A keyboard is nothing more than an array of switches, in the form of an m-row x n-column matrix. Only m + n physical wires need connect to the computer's data bus, however m x n discrete switch positions are catered for in the matrix itself. Additional software overhead is required in order to resolve individual switch closures.

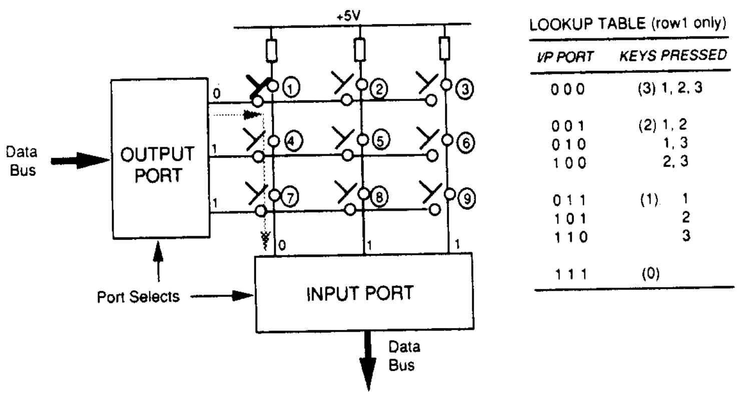


Figure 0-11 3 x 3 Keyboard matrix (array).

In the 3-row x 3-column keyboard matrix shown in Figure 0-11, the computer's data bus is divided in two, with three lines permanently connected to the rows, and another three to the columns. In the row-scan technique of keyboard encoding, each row has a zero written to it in turn, with the columns being read immediately after each write. Whenever a zero is detected in a column, this indicates that a key has been pressed; a lookup table then needs to be consulted to determine which key was in fact pressed. There are eight possibilities for each row; 000 indicates that all three keys are pressed, 001 keys 1 and 2, and so on down to 111 (indicating no key pressed). However, this software decoding technique is memory intensive, assuming all possibilities need to be catered for (8 x 3 = 24 in this simple example). This may or may not be a significant penalty, with memory capacities increasing (and cost per bit decreasing) as rapidly as they have been in recent years. Often only single key presses would be of interest; multiple key presses would be treated as miskeying on the part of the operator.

A problem can exist with such matrix keyboards, known as “ghosting” or “ghost key”. Consider the 3 x 3 matrix shown in Figure 0-12: with the three switches closed as indicated and a zero written to row #2, a zero appears on the middle column output as expected, but an unexpected zero also appears on the right-hand column output (due to the alternate current path shown). **The simple remedy for ghosting is to place diodes in the columns as indicated**, to ensure that currents cannot flow back through one of the top two switches, forcing another row LO.

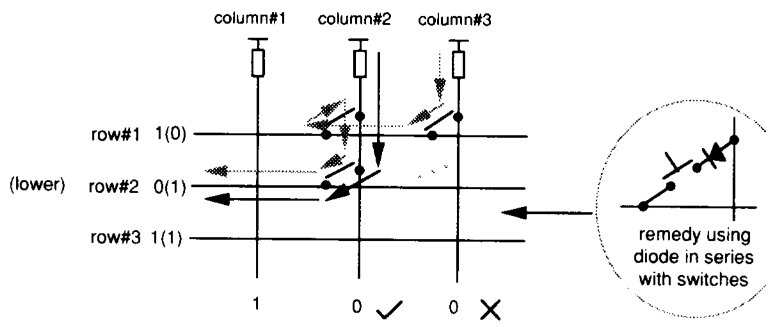


Figure 0-12 Ghost key and its remedy

In a matrix keyboard design, the size of the keyboard can be easily increased. Eight-bit input and output ports provide a 64-key interface. Adding a second output port provides another eight rows and doubles the number of keys that can be supported. (Alternatively, another input port can be added for additional columns.) With 16-bit input and output ports, 256 keys can be supported. Usually the mode keys, *SHIFT, CTRL,* and *ALT* are connected direct and not scanned. Auto repeat function, debouncing and rollover features usually incorporated with the scanning software.

4.9 KEYBOARD LOCKOUT

Minimum acceptance time for keyboards is typically 20 to 30 ms to take care of switch bounce. Inter keystroke interval for the average typist can be less than this for certain key sequences, with resultant errors due to character omissions. Also there is overlap of finger motions such that more than one key will be depressed at any given time. The software must determine how this is handled. The simplest technique is called *two-key lockout*. When any key is pressed, all others are ignored until the first key is released. This is simple to implement and is acceptable for special-function key-boards.

The following example shows how this method responds:

Key action	Program action
A pressed	A pressed
B pressed	none
B released	
A released	A released

Thus, the B key is entirely missed.

4.10 KEYBOARD ROLLOVER

In applications where rapid keyboard entry is likely, the overlap of finger motions means that more than one key will be depressed at any given time. A keyboard feature that allows a keystroke to be accepted while another key or keys (depending on the specific design) is depressed is known as rollover. A keyboard with no rollover disallows other keystrokes as long as one key is depressed. Several versions of rollover exist. Figure 0-13 shows the timing characteristics of several rollover schemes.

Two-key rollover (also known as shadow rollover) allows a second key to be depressed when one key is already down. The second key, however, will output a signal only as the first key is released. If the second key is released before the first key, the second keystroke will be ignored. If two keys are pressed simultaneously, all output is blocked.

Three-key rollover provides valid data for two sequentially depressed keys. Depression of a third key does not produce a character output until one of the first two keys is released.

N-key rollover allows a keyboard to accept all keystrokes and generate all valid characters in stroked sequence.

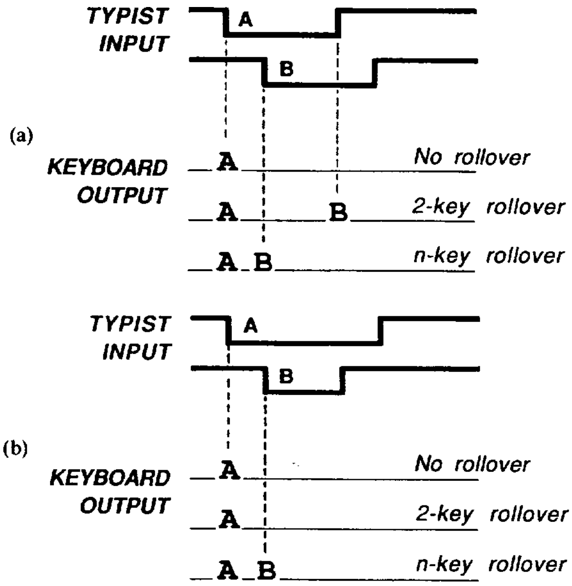


Figure 0-13 no rollover,(a) two key rollover, and (b) nkey rollover

Figure 0-13 Shows the action of keyboards with no rollover, two key rollover, and nkey rollover: (a) the effect on keyboard output when two keys are pressed simultaneously and the first key is released before the second, (b) when two keys are pressed simultaneously and the second key is released before the first.

Although n-key rollover reduces errors by allowing unlimited overlap of key depressions, it is unlikely that typists will depress more than three or four keys at once. Therefore, the capability to accept an unlimited number of overlapping keystrokes may be unnecessary.