# Portable Product Miniaturization and the Ergonomic Threshold

by

David H. Levy

S.B. Mechanical Engineering, MIT 1987
S.M. Mechanical Engineering, MIT 1987

Submitted to the department of Mechanical Engineering in partial fulfillment of the requirements for the degree of
Ph.D. In Mechanical Engineering
at the

**Massachusetts Institute of Technology**

June 1997

[September 1997]



Signature of author:________________________
Department of Mechanical Engineering
May 27, 1997

Certified by: ________________________
Alexander Slocum
Associate Professor of Mechanical Engineering
Thesis Supervisor

Accepted by:________________________
Prof. Ain Sonin
Graduate Officer

# Portable Product Miniaturization and the Ergonomic Threshold

by
David H. Levy

Submitted to the department of Mechanical Engineering in partial fulfillment of the requirements for the degree of Ph.D. in Mechanical Engineering

ABSTRACT

Portable products have exhibited two notable and conflicting trends over the last thirty years: They have simultaneously grown smaller and more complex. Superimposed on these trends is the unchanging size of the human hand. Together, these three curves define the limitations on miniaturization of portable products, a market segment inextricably entwined with our concept of the future.

This thesis:

1) identifies the "ergonomic threshold" as the transition between electronic-limited miniaturization and interface-limited miniaturization, stating it to be an important juncture, affecting portable products directly, as well as creating an ongoing dynamic between the interface and electronic industries.

2) investigates the ergonomic threshold with respect to miniaturization technologies of the present and future, and identifies a notable gap in the stream of technologic advance along the miniaturization curve in terms of price and performance. As proof of this gap, several key family groups are identified whose technologic development has been stalled due to the non-existence of suitable input devices.

3) introduces a series of three input technologies that address the technology gap. The first technology increases input density by an order of magnitude without compromising on ergonomic quality or increasing cost. The second solves an ergonomic problem that prevents non-chorded keyboards from use in the highly miniaturized world of wearable computing. The third offers the advantages of traditional keyboard input in augmented reality applications while providing a second order-of-magnitude size decrease.

Thesis Supervisor: Alexander Henry Slocum, Ph.D.
Title: Alex and Brit d'Arbeloff Associate Professor of Mechanical Engineering

# 1. Overview

Product miniaturization has been central to our concept of the future for generations. There is an unstated assumption that products will continue to miniaturize as they become more complex, providing ever-increasing portability and power. This thesis challenges that assumption, demonstrating that the path to the future is not so clear. As is often the case, the stumbling block to technologic advance is not the technology itself, but the human concerns that encompass it. Namely, while technology may drive electronics ever smaller - and paradoxically, more complex - the human hand will not adjust to accommodate.

Throughout the age of electronics, the net result of electronic miniaturization and increased product complexity has been product miniaturization. However, this trend must stop when product size reaches the limitations created by size of the human hand. Once this boundary is reached, the limits of product miniaturization shift from being dictated by the electronics within to being dictated by the ergonomics without, a boundary identified by this thesis, and called the Ergonomic Threshold. Crossing the ET impacts the development of over $124 billion worth of goods within the portable product and/or electronic industries, as well as the ability for our society to enter a next generation of technologic advance in which computation, communication and information become as accessible as the time of day.

This thesis is comprised primarily of four sections.

• The first defines and explores the ergonomic threshold as a purely theoretical construct.

• The second investigates the impact of the ergonomic threshold in practical terms as it affects the portable product industries, input device technologies, and the electronic industry. Case studies are used to demonstrate the impact of the ET.

• The third evaluates input technologies of the present and future, and determines that a significant gap exists in the availability of interface technologies on a performance per dollar basis. The analysis is supported by case study, data collection, and library work.

• The fourth section proposes three new input device technologies that help to address the technology gap by moving the ET farther out in time. The first establishes a fundamental new no-compromise keyboard paradigm that reduces keyboard size by an order of magnitude. The second renders the first applicable to an augmented reality environment. The third miniaturizes the second by an additional order of magnitude, again without compromising on ergonomics.

## 1.1 Scope

In the broadest sense, this thesis encompasses portable products available in the next thirty years. However, the author is highly aware that any work that deals with technologic prognostication risks inaccuracy and irrelevance. The intent is to try to balance aggressive theoretical exploration with unadorned applicability. *The intent is to focus not on technology for technology's sake, but on the analysis of clear trends to the end of achieving maximal economic and social impact.* Notably, this thesis avoids market-based analyses, largely because they are believed (by the author) to be demonstrably less accurate than weather prediction, which is itself a rather unsuccessful practice[1]. Therefore, the scope of this thesis is to analyze trends that control the direction of portable product miniaturization based exclusively on well-established, widely-accepted parameters, and to stop probing when these criteria are not met. These areas will be identified throughout the thesis.

## 1.2 Sociological Significance Of Miniaturization

Obviating the need for an office (even at home) would represent a dramatic, all-encompassing change to the organization of society. Accomplishing this goal is dependent on the successful miniaturization of computation and communication products, which is in turn dictated by the consequences of the Ergonomic Threshold.

## 1.3 Economic Significance Of Miniaturization

Consumer product industries closely associated with miniaturization include laptop computers, Personal Digital Assistants (PDAs), cellular telephones, pagers, and cellular fax communication, together representing approximately $124 billion dollars annually in 1996[2]. While any specific estimates would be purely

---

[1] Weather reporting is a science that depends on billions of dollars worth of equipment, collecting a large volume of continuous, instantaneously applicable data to evaluate a large, yet ultimately deterministic system in which characteristic geographies helps establish patterns over time. The goal is to predict a few days into the future. The result is poor. Compare this with market research data, in which a relatively tiny amount of data is used once, in a hypothetical situation using "systems" as repeatable, and deterministic as human beings, themselves making predictive decisions based on a subset of information made available by researchers trying also to be predictive. The goal is to predict human behavior months, sometimes years in advance. Clearly, weather prediction should have a higher chance of accuracy than a marketing study.

[2] Compiled from *Dataquest*, April 1997

speculative, increased portability would have additional economic impact on the use of communications. New business enterprises will cater to an increasingly portable society. Additional industries that would benefit from product miniaturization include avionics and aerospace, dental, medical, security, and interactive television. Specific examples are provided.

## 1.4 Objectives

The objectives of this work are:

1) To establish the validity and importance of the concept of an Ergonomic Threshold through investigation of the following:

- There has been a long-standing assumption that products will continue to become smaller. This thesis questions that assumption, and replaces it with the assertion that product miniaturization progresses based on successive ergonomic thresholds, each established by a local input device paradigm.

- Even more contrary to the long-standing assumption that products will continue to miniaturize, is the assertion is that once the ET is crossed products will begin to increase in size. And, this trend of product increase will reverse once a new threshold is established by the creation of a viable new paradigm for interface technology.

- Demand for miniaturized electronics (in the consumer product market segment) will decrease once the ET is crossed.

- Demand for miniaturized input devices increases significantly after an ET is crossed.

- Lastly, because of the two preceding points, the first crossing of the ET will establish a dynamic between future advancements in interface and electronic technologies. The economic value of the products at the first crossing is on the order of $100 billion.

2) To move the ET further out in time by:

- Establishing new interface paradigm, specifically, by challenging the long-standing notion of maximal information density being determined by the human hand. The objective is to invent, develop and introduce a practical, low-cost device that allows comfortable, no-compromise ergonomic operation of a keyed device in which the keys are significantly smaller than suggested by the human hand and offer a performance enhancement discontinuous with the existing paradigm.

- Building upon the paradigm by enabling traditional keyboards to be applicable within the key area of wearable computing.

- Introducing a third technology that maintains a no-compromise ergonomic standard while providing an order of magnitude size reduction in the critical wearable computing industry.

## 1.5 Contributions

The contributions of the thesis are:

• Observing the ET exists and will have long-term impact on: miniaturized products, the process of product design, input device technology and the electronic industry. The historic trend of continuous product miniaturization is not sustainable.

• Providing a conceptual analysis of the ET. Beyond noting the existence of an ergonomic threshold this thesis provides an improved understanding of the relationship between electronic miniaturization and input devices. This provides the electronic and consumer product industries tools to enable researchers and developers to better direct their efforts.

• Inventing and developing a fundamental new keyboard design that re-establishes the interface asymptote at a level beneath that established by conventional keyboards, thereby shifting the miniaturization trajectory from the interface curve to the electronic miniaturization curve. The design is basic enough to offer a lasting contribution that extends beyond the miniaturized product industries.

• Inventing two additional input technologies that address the ergonomic voids in the advance of wearable computing.

## 1.6 What is Ergonomics?

Ergonomics is a poorly understood and vastly under-rated field. We often think of ergonomics as comfortable chairs or easy-to-grip handles, but ergonomics is the study of making objects "user friendly" to mind and body, a significance far exceeding chairs and kitchen utensils.

The study of "ergonomics" has held a variety of meanings since its inception in 1857[3]. In the middle 19th century the tasks of animals and humans were often similar and the science of ergonomics was conceived to guide the task of making work fit for humans. The goal was to humanize working conditions and the "ergonomics" of the time was embroiled with social, political and economic import, as it often pitted the interests of industry against those of the workers. However, in the heyday of the industrial age there was little need for the subtleties of a "science," as poorly designed machinery routinely cost people's limbs or lives. Work humanization continued, but the "science of ergonomics" faded from academic literature for nearly a century.

The subject reappeared during the Second World War when the serviceability and usability of relatively complex devices in combat situations became of paramount importance. War and the stress of combat was the influence that brought psychology into the realm of ergonomics for the first time under the name of "Human Engineering"," Technical Psychology" or more specifically, Knob and Dials Ergonomics. This was the introduction of the concept of user friendliness. This concept originally fell within a sub class of ergonomics known as Praxiology, or the study of practices. While it is impossible to determine the degree of importance to the war effort provided by the increased ability to quickly and

---

3 *Connective Networks in Ergonomics* Edward Franus Elsevier Press 1991

correctly operate war machinery in battlefield conditions, it is clear that ergonomics again played an important social role.

The focus of the field has changed yet again. The American Heritage dictionary defines ergonomics as "the applied science of equipment design, as for the work-place, intended to maximize productivity by reducing operator fatigue and discomfort." The field then bifurcates into mental and physical sides. In common parlance the word "ergonomics" is associated with supportive, comfortable chairs or handles that conform to the palm. This portion of ergonomics is well understood, with many handbooks providing dimensions and guidelines for the optimal ergonomic design of common objects. The mental side of ergonomics has strongly entered mainstream society and is commonly referred to as people discuss the "user friendliness" of a product.

The issue of "user friendliness" (Praxeology) remains one of the most used and less understood phrases in product design. One of the largest economic battles in history was recently fought over a praxeologic issue. The Microsoft Corp. provided a well known, widely used interface called DOS. A small company called Apple Computer introduced a different interface based on graphics. After only a few years, the graphical interface was promptly copied by Microsoft, due to its overwhelming success in the marketplace. Today it is nearly impossible to find a computer that does not have a graphical user interface. The importance here is that the secret to winning was to adopt the principles of sound ergonomic design.

The automobile industry is one of the largest in the world. Again ergonomics is the primary tool used by the industry to differentiate its products. Comfy chairs. More headroom. Cup holders. Smooth ride.

But isn't a smooth ride a performance issue? Then again, isn't "performance," as an entity, an ergonomic issue?

For that matter, if ergonomics encompasses the mental and physical aspects of optimizing to meet human expectations and needs, then the question arises: what ISN'T ergonomics? If ergonomics is about minimizing our mental and physical burdens, then the invention of the automobile was itself an ergonomic advance, as was the telephone and the microwave oven. And the conventional oven. And the wheel.

Once this idea is internalized it is hard to find any technologic advance that is not ergonomic in nature. Indeed any device which has an interface and provides utility, from a door, to the control panel for a nuclear plant has ergonomic concerns deeply embroiled in its conception and implementation. Without considering it, most evaluations of the devices we use are ergonomic evaluations. Every time we are confused, injured, frustrated, fatigued, or stressed by a device - the blame and eventually, the answer, lie in ergonomics.

It raises the question: is this definition of ergonomics so broad as to be meaningless? I will argue the opposite: I believe that seeing the world through ergonomic eyes focuses the designer the way the designer should be focused. Advances should NOT be made in the vacuum of a need to make advances, an approach that often results in technically workable, yet functionally poor products. There is a benefit to maintaining the true context in mind. Advances should be made with the direct goal of satisfying human need. Ergonomics is the appropriate lens to guide the vision of any designer.

# 2. The Ergonomic Threshold - Theory

As one gets farther from basic physics, the elegant simplicity of mathematics drops away, and is replaced by the dirty complexities of the real world. This thesis seeks to develop meaningful theory within a real world context by basing it on the analysis of well-established trends.

## 2.1 Overview

There are several trends within the field of product development that are commonly known, and well-established. Products get smaller. Products get more complex. These two trends alone are enough to determine that at some point in time, a conflict will occur. There is another, less-considered "trend," of the human hand not changing. The theory of the ergonomic threshold is derived from evaluation of these three trends and determining the levels of their respective asymptotes.

## 2.2 Electronic Miniaturization Curve

In 1965, Intel founder Gordon Moore stated his belief that transistor count available per dollar would increase by a factor of two every 18 to 24 months. As shown in Figure 1, this prediction, now known as Moore's law, has proved quite accurate over three decades. Likewise, products that contain electronics have decreased in size at an impressive rate, although no curve analogous to Moore's law has been suggested. It is however well known that the size of consumer products such as radios, calculators, television sets, computers, and cellular communications have all dramatically reduced in size over the same period.

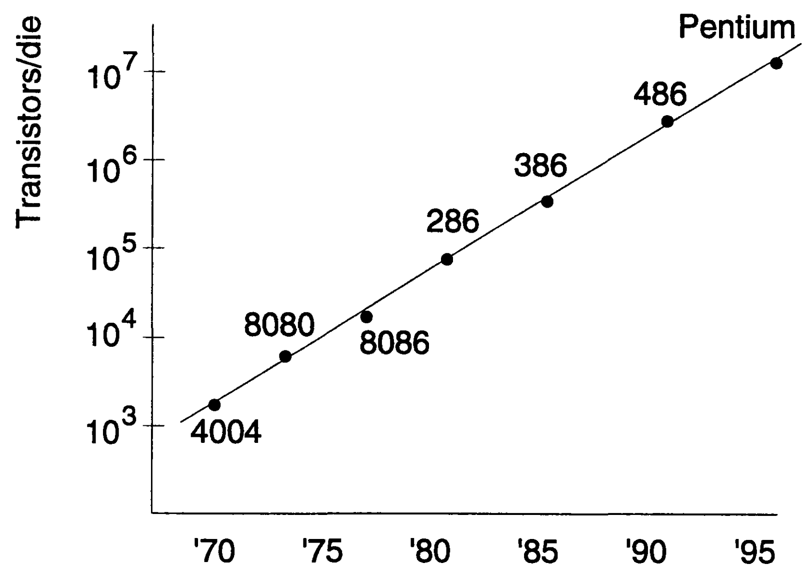



**Figure 1 - Moore's Law**

How long can this trend continue? While electronics will ultimately face a miniaturization asymptote dictated by quantum effects, this boundary is difficult to identify. Pessimistic industry predictions estimate this boundary will be approached by the year 2010[4]. If so, the pressure will mount for a new electronic paradigm. Optimists suggest that the quantum effects themselves will be harnessed as the mechanism for a next generation of electronic hardware, making

---

[4]*1993 Interrnational Electronic Device meeting Technical Digest*. Piscataway, NJ IEEE, December 5-8, 1993.

the quantum physical barrier merely a local asymptote, thereby amplifying the significance of this work. (However, the specifics of such theorizing would be pure conjectural and will not be pursued here.) Even under the most constrained estimates, transistor density is expected to be greater than two orders of magnitude smaller than it is today[5].

## 2.3 Product Miniaturization Asymptote

If quantum effects dictate the ultimate miniaturization asymptote for electronics, what dictates the ultimate miniaturization asymptote for overall product size? The answer is clearly that the product must be large enough to be useable by the consumer. For products that use a physical input device (as opposed to voice or neural control), the miniaturization asymptote for products is dictated by ergonomic concerns, namely the dimensions of the human hand. While voice shall inevitably become a primary input device, even for low-cost consumer products, physical interfaces, such as keyboard and handwriting recognition will still be necessary. While discussed in more detail in Section 4.4.8, the voice interface is simply inappropriate in most portable device contexts: in coffeehouses, during meetings, in classrooms, or in transit. Therefore, until a viable neural interface is developed, a miniaturization asymptote exists for the vast majority of portable products, and it is defined by the human hand.

## 2.4 Definition of ET

Observation of the electronic and product miniaturization curves are enough to introduce the concept of an Ergonomic Threshold. As shown in Figure

---

[5]*The National Technology Roadmap for SemiConductors, 1994* , Semiconductor Industry Association

2 the large discrepancies between the asymptotes of these curves suggest that at some point in time products will cease being miniaturization-limited by the electronics and become miniaturization-limited by the interface, defining an Ergonomic Threshold to miniaturization.

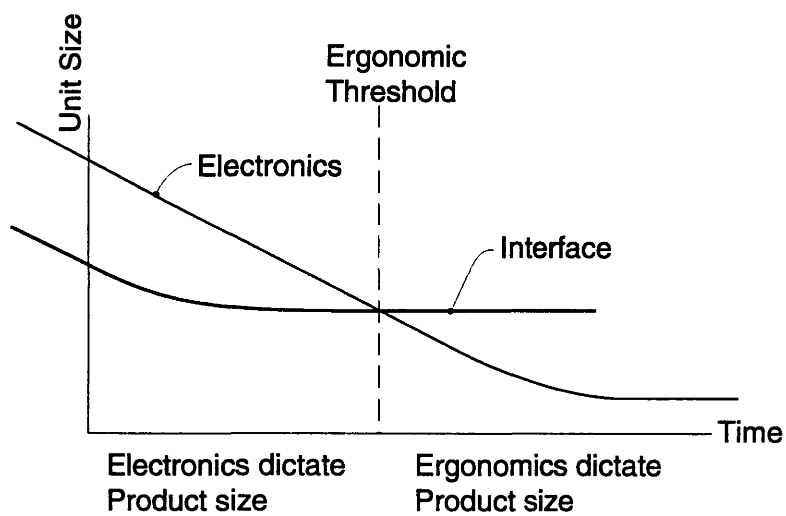



**Figure 2 - The Ergonomic Threshold**

Common sense dictates that continued decrease in electronic component size, on its exponential slope, and a continued constancy in the size of the human hand will eventually result in either: products too small for the human hand to operate, or the potential to ship empty space inside a "miniaturized" product. The first case suggests the need for a new ergonomic paradigm. The second case suggests that the need for advanced electronic technologies will decrease over time. Rather than requiring ever-smaller electronics, products may be manufactured with relatively old electronic technologies.

An examination of the characteristics of the design process on either side of the threshold:

### 2.4.1 Electronic-Limited Product Regime

Product designers have worked in on the electronics-limited side of the threshold since the dawn of electronics. In this regime product size is determined by the volume, or increasingly, the area of electronics needed to enable the product. Electronic engineers provide an envelope dimension to the mechanical engineers who then work to optimize the product within the electronic constraints provided. This is a relatively open-loop design process, with minimal feedback between electronics and mechanics, in which the electronics dictate mechanical design.

### 2.4.2 Interface-limited Product Regime

Once the ET is crossed, product size is determined by the interface, and two new issues arise in the design process.

1) The product designer must decide upon the balance between product miniaturization and ergonomic quality. The mechanical designer has gained control of a highly-significant variable, that of overall product size. Product size is an absolutely critical parameter in the success of portable products, but once the ergonomic threshold is crossed, smaller is not necessarily better. Excess miniaturization results in deteriorated ergonomic quality, and negatively impacts sales. Beyond the threshold, designers must become increasingly adept at the tradeoffs between portability and usability.

2) Once the ergonomic threshold is crossed, an additional feedback loop is created between the mechanical and electrical aspects of the design process. Namely, once product becomes interface-limited there will be an increasing amount of potential space available as electronic technologies continue to advance. However, there will be no reason for the electrical designers to pay for these new, more sophisticated electronic technologies as they become available, if

the same functionality can be provided by older, and larger, technologies. Therefore, once products cross the ergonomic threshold there is a decreasing need for state-of-the art electronics. Certainly, products will continue to become more sophisticated and require more powerful and faster circuitry. However, during the last thirty years of battle between complexity and miniaturization, miniaturization has always won, as products have continued to shrink. Based on the assumption that this trend will continue, the real estate inside interface-limited products will become increasingly available. The value of this space will therefore decrease and increasingly "low-rent" electronics will be available to move in. Section 3.10 discusses this trend in detail.

## 2.5 Increased Functionality

The other major trend in the technologic advance of product is a steady increase in functionality. Features are constantly being added, another benefit of Moore's law, as it provides a steady increase in the number of transistors available per dollar. Along with this increase in product functionality is a corresponding increase in functionality of the user interface, suggesting that once products cross the ET, they will increase in size to maintain constant ergonomic quality. Or, they must sacrifice ergonomics to maintain the a desired size.

## 2.6 Functionality asymptote

While there is strong stereotype that products have an ever-increasing number of buttons as functionality increases, interface complexity also faces an asymptote. Because the desktop keyboard is observed to provide adequate interface to thousands of different software packages and seems to perform well, this suggests that the existing computer keyboard provides a complete set of input possibilities with approximately 106 keys representing approximately 132 labeled functions. It is increasingly clear that this level of functionality is an asymptote toward which many products aspire. This will be examined more in 4.2. For now, let us accept that the size of the interface also has an asymptote.

## 2.7 Product Miniaturization Curve

History suggests that despite an increase in product complexity, the overall size of consumer products decreases over time. However, this increase is still governed by ergonomic limitations. What is the net result of increased product complexity and decreased electronic size? Figure 3 shows product size within a product family over time within a given ergonomic paradigm at a constant ergonomic level, suggesting that beyond the threshold product will begin to grow in size until a level of "full functionality" is achieved, at which time product growth will stop.

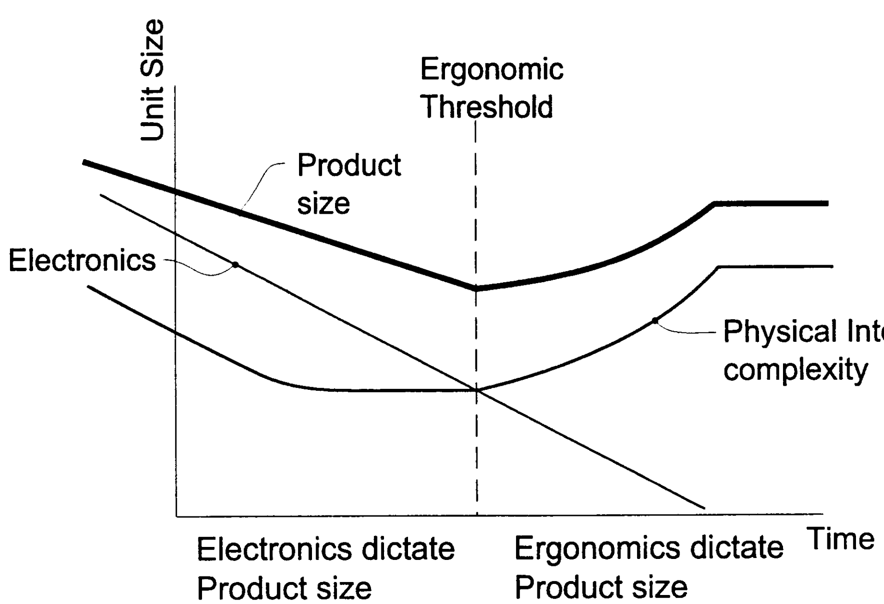



**Figure 3 - Product Size Over Time**

## 2.8 Electronic/Interface Dynamic

As suggested in Section 2.4.2, once the threshold is crossed for the first time a dynamic is created between the electronics and the interface. Referencing Figure 4, once products are interface-limited, products enter a regime with three possible outcomes.

- Products will increase in size, potentially to the point at which it will not be developed because the size is considered unacceptably large (See 3.3.3)
- Products will continue to be developed using electronics that are increasingly out-dated, as relatively old technologies provide adequate miniaturization. This scenario has direct impact to the electronics industry as it reduces demand for high tech electronics in the largest sectors demanding advanced componentry: portable products.
- Ergonomics degrade, affecting the usability, and hence desirability, of the products.

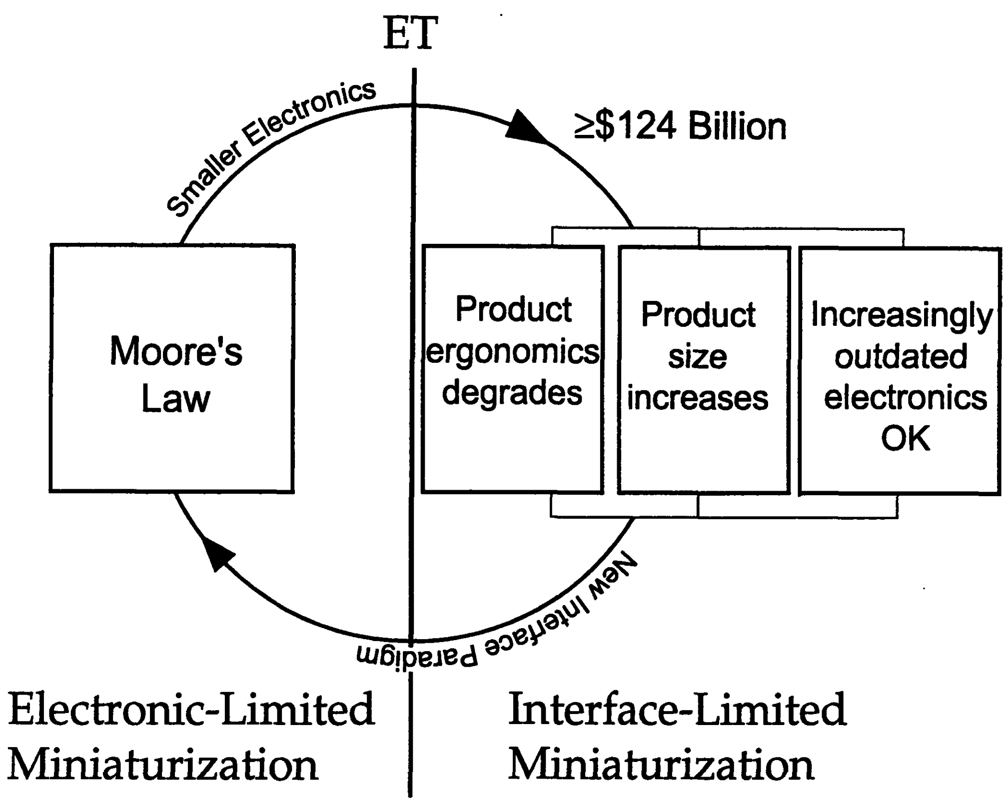



**Figure 4 - Electronic/Interface Dynamic**

All three cases stimulate the need for a new interface paradigm. When this new paradigm is both found and cost-effective, products will transition back across the Ergonomic Threshold into an electronic-limited regime. When products are electronic-limited, the current design practices and economic structure will again be in effect until Moore's law drives the system across the ET again.

## 2.9 Summary of the Ergonomic Threshold Theory

While it is common knowledge that electronic miniaturization has enabled products to miniaturize for the last 30+ years, this trend must cease as product size approaches the limitations of the human hand, a transition predictable by the large discrepancies between the electronic and ergonomic asymptotes. **The ergonomic threshold is defined as the point at which the minimum product size is determined by the interface rather than the electronics.** Once this threshold is crossed, a counter-veiling relationship will be established between the portable product industry and the segment of the electronic industry that supports portable products. There will be a gradually increasing disincentive to use state-of-the-art electronics, shifting the mix toward older technologies. Furthermore, because product size has decreased during the last 30 years, despite a general increase in product (and hence user interface) complexity, products will begin to increase in size once the countervailing force of electronic miniaturization may not exert an affect. That is to say that beyond the ergonomic threshold, there is a tendency for products to grow larger over time. This growth in product size exacerbates the size differences between the interface and the electronics, and will therefore amplify the effect upon the electronic industry.

The ergonomic threshold also presents a significant change to the process of product design, as products beyond the threshold must take diverging paths between an optimized interface, or an optimized size, but not both. This places additional responsibilities on the product designer that overlap strongly with marketing concerns.

As shown in Figure 5, these problems will continue to increase until a new ergonomic paradigm is established. To be effective, the new technology must be of sufficiently high ergonomic standards to be widely acceptable, and also cost-effective for application into portable products.

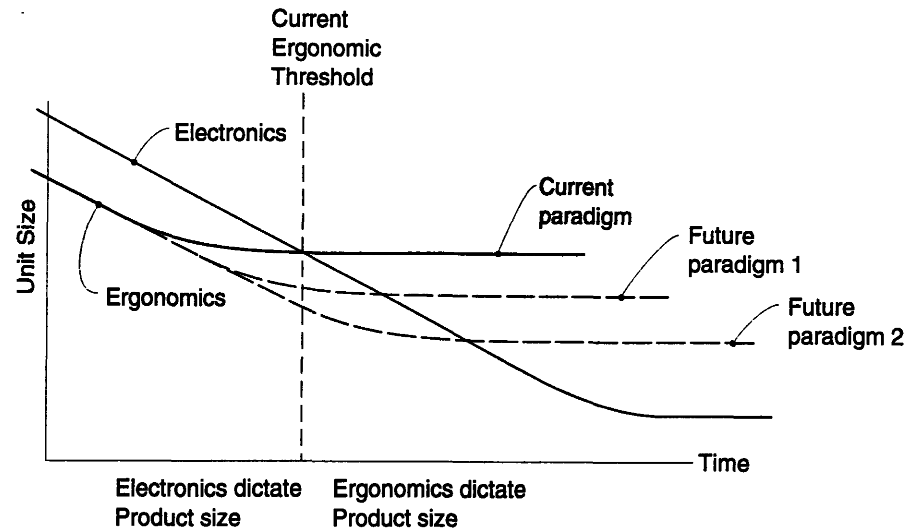



**Figure 5 - The Ergonomic Threshold**

The issue of an ergonomic threshold is new because most consumer products have been operating in a region far from it. While a few products ahead of their time, such as the calculator watch, introduced the possibility of conflict beyond the ergonomic threshold, these were individual drops, long in advance of the storm. Large scale conflict between interface and electronics is a new topic whose importance will only increase.

## 2.10 Predictions Made by ET Theory

The theory of an Ergonomic Threshold predicts the following:

- Products will stop miniaturizing at the ET.
- Products will start *increasing* in size once the ergonomic threshold is crossed.

- The electronic industry will be affected by decreased demand for future advances in miniaturized electronics.

- Once the ergonomic threshold is crossed, there be a dynamic established between the need for a new interface paradigm and the need for advanced electronics.

- The dynamic between the interface and electronic industries will be superseded once a new input device is established.

While a few of these predictions may only be answered with time, most are addressed in the balance of this thesis. As will be shown, the evidence supports the assertion that the theory of the ET that has tangible impact to the electronic, portable product and input device industries.

# 3. Ergonomic Threshold - Practice

To demonstrate the ergonomic threshold theories as they exert an influence on product development, this section presents examples of existing individual products, as well as families of products of the past and present. Examples are also given of products beyond the ET that have forced the product to grow dramatically in size. Finally, examples are given of real-world "virtual" products. These "virtual" products do not exist - specifically because they lie so far beyond the ergonomic threshold that these industries cannot find an input device small enough to make the product viable.

## 3.1 Case History of the Mobile Data Terminal

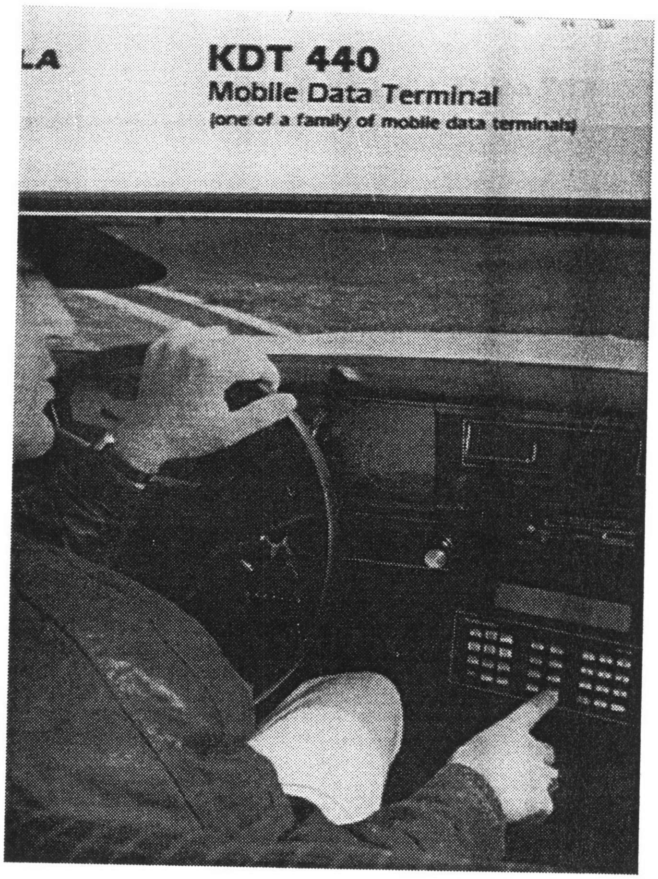



**Figure 6 - Mobile Data Terminal KDT 440**

In 1980 Motorola introduced the Mobile Data Terminal KDT 440, shown in Figure 6. The product operated in the FM band and allowed users access to data and written communication from a central source. A 12-key numeric keypad and 20 additional customizable function keys provided the interface in a 3.75″ x 12″ area for a key density of 1.4 keys per square inch.

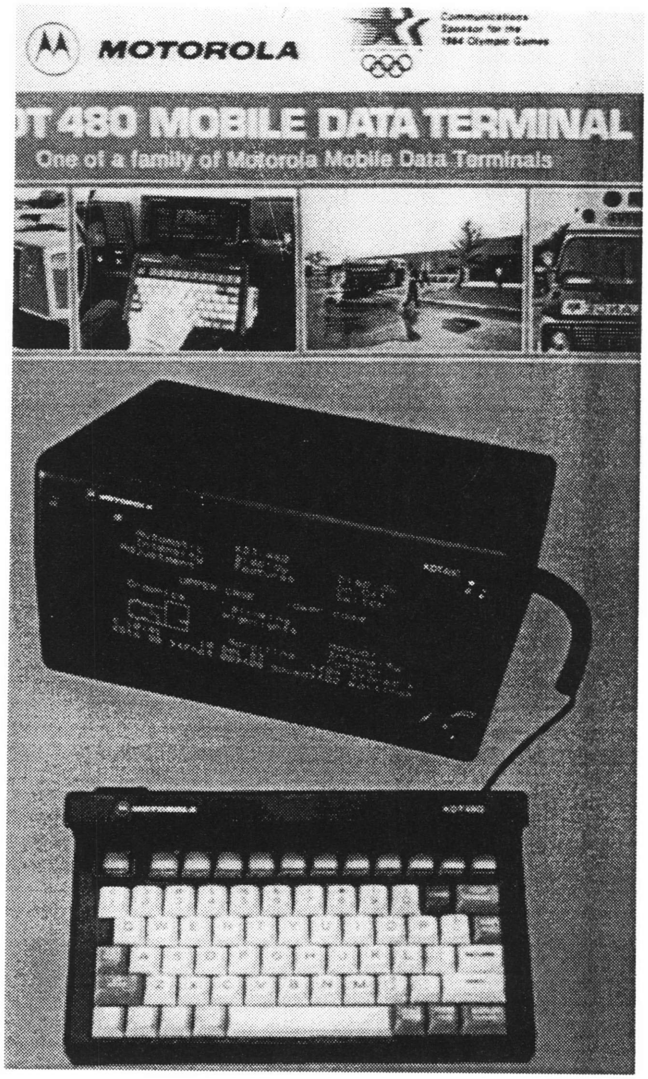



**Figure 7 - Mobile Data Terminal KDT 480**

In 1987 Motorola introduced the Mobile Data Terminal KDT 480, shown in Figure 7. The product introduces a full QWERTY alphanumeric key board and 13 additional functions in a 5″ x 10.5″ interface area for a key density of 1.16 keys per square inch.

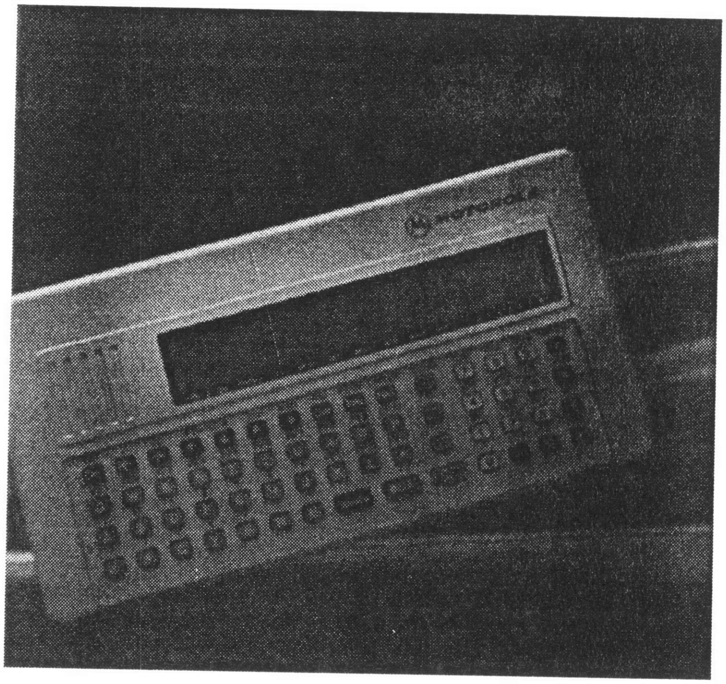


**Figure 8 -Mobile Data Terminal 800**

A year later the company introduced the 800 series of Mobile Data Terminals, shown in Figure 8. The product line again experienced significant growth in interface complexity and interface density. The product grew to 57 keys, each with secondary functions in an area of 2.3" x 8.2" for a key density of 3 keys per square inch.

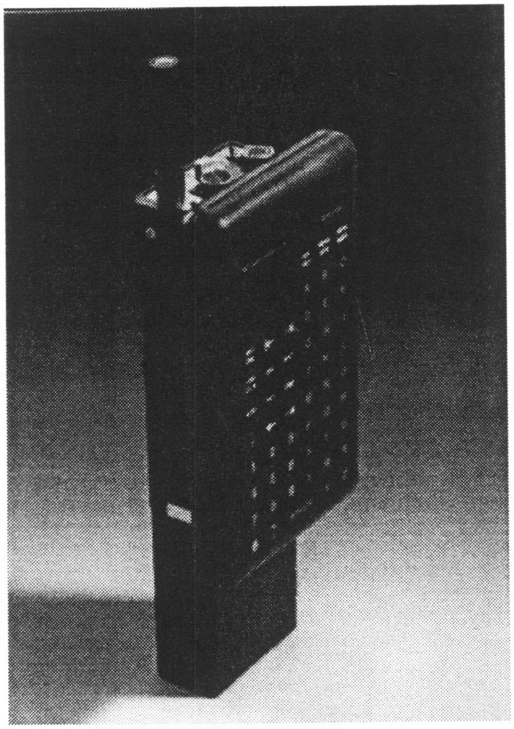


**Figure 9 - Mobile Data Phone 4800**

Ultimately the company introduced the MDC 4800 Mobile Data Phone, shown in Figure 9. With 54 keys in a 2.75" x 4.5" interface area, the product has a key density of 4.4 keys per square inch. As will be later shown, this is the ergonomic maximum key density. And, in order to achieve this density, the product was modified to accommodate an interface that has clearly grown larger than its associated technology. This is a product that has passed the Ergonomic Threshold.

## 3.2 Case History of the Pager[6]

Pager technology has slowly advanced for over 40 years from an extremely simple device of basic utility to a powerful communication tool. It also maps well onto the ET theory.

### 3.2.1 Beeps only

The pager was introduced in 1951 as an alternative to the public address systems used in hospitals. The device was designed to operate within loops of wire which circumnavigated the periphery of a building. All units operated at the same frequency, non-selectively, and therefore all pagers within the building were activated each time the system was used. This problem was addressed the following year when selective radio oscillators allowed pagers to be beep selectively, to indicate that a particular doctor was needed. The device had a volume over 20 cubic inches. The interface was a volume control and on/off switch. Technologic advances concentrated on increasing range and the number of pagers that could be fit onto a network. By 1958, the first fully-transistorized unit had decreased size to less than half the original volume of the product. FM transmission eliminated the need for customized antenna systems within each building. By 1971, pagers had decreased to 4.8 cubic inches. Part count had decreased from 210 to 80. Range had increased to cover entire cities and a network could support up to 7,000 units.

[6] Data compiled from various sources at the *Motorola Museum for Electronics*, Schaumurg, Il.

### 3.2.2 Voice

Voice pagers were introduced in 1960, but never claimed more than a small fraction of the paging market. Voice was more expensive, and the marketplace was uninterested in paying for the extra functionality. By 1971 voice pagers had added message-storing feature and the interface grew a few additional keys, but is not considered a trend in itself due to the small significance of voice paging.

### 3.2.3 Receive Numeric and Alphanumeric

The first modern pager was introduced in 1974. The Motorola *Metropage* introduced an LCD screen for displaying text pages. Still, the basic functionality of receiving text demanded only minimal interface growth. The functionality to store, retrieve, erase, protect and scroll, added up to six buttons, yet was sufficient to handle even sophisticated numeric and alphanumeric messages. Volume decreased to as small as .6 cubic inches in the RSVP pager, integrated into a cellular phone battery.

### 3.2.4 Send Alphanumeric

With the advent of the capability to send messages, the need for interface outgrew the size of the device. In 1994 Skytel introduced a limited two-way AN pager in which the message options were limited to 16 pre-programmed messages. an example of a product beyond the ergonomic threshold and already beginning to increase in size. The first version of two-way pager included an on-screen "keyboard." A year later Motorola introduced *Tango*, a product that allowed messages to be written in the pager using the interface shown in Figure 10. All characters appeared on-screen, in a line, and the cursor could be scrolled left or right by pressing buttons to select each character. This was tedious.

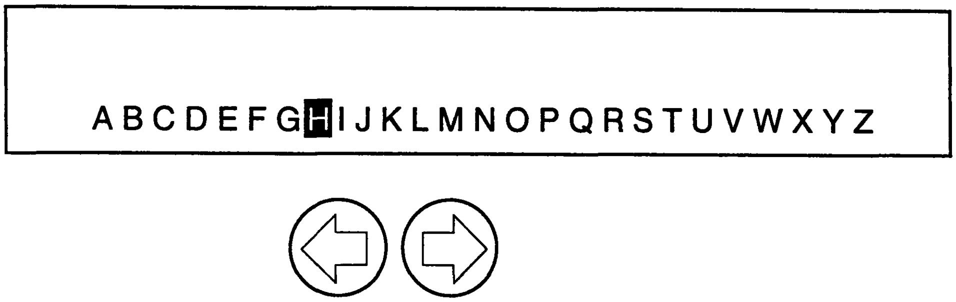



**Figure 10 - Tango Interface**

Sales were quite poor. Less than 1/3% of pagers sold are two-way pagers. Motorola has identified the reason as the awkwardness of the interface. Motorola marketing research predicts AN two-way paging to replace 50% of the existing numeric pagers within five years if a useful interface can be developed.[7]

In 1997, the company introduced the *Pagewriter 250,* shown in Figure 11, and the *Pagewriter 2000,* shown in Figure 12. The *Pagewriter 250* has 36 keys with an interface density 10 keys per square inch and increases product size over the *Tango* by 20%.

[7] Jim Page, Director of Marketing. Motorola pagers

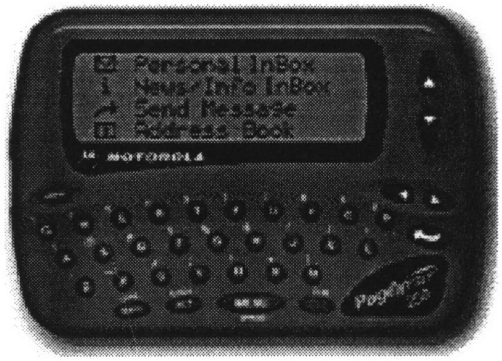



**Figure 11 - Pagewriter 250**

The *Pagewriter 2000* has 49 keys with an interface density 8 keys per square inch and increases product size by 26%.


**Figure 12 - Pagewriter 2000**

The case history of the pager, as it pertains to the ET, is summarized in Figure 13. From its introduction in 1952 until 1990, the volume of the pager decreased dramatically as the complexity of its interface grew slightly. As soon as pager technology advanced to include the ability to *send* data, interface growth increased by an order of magnitude, while product size increased by a factor of six It is clear that pager technology crossed the ergonomic threshold in 1990.

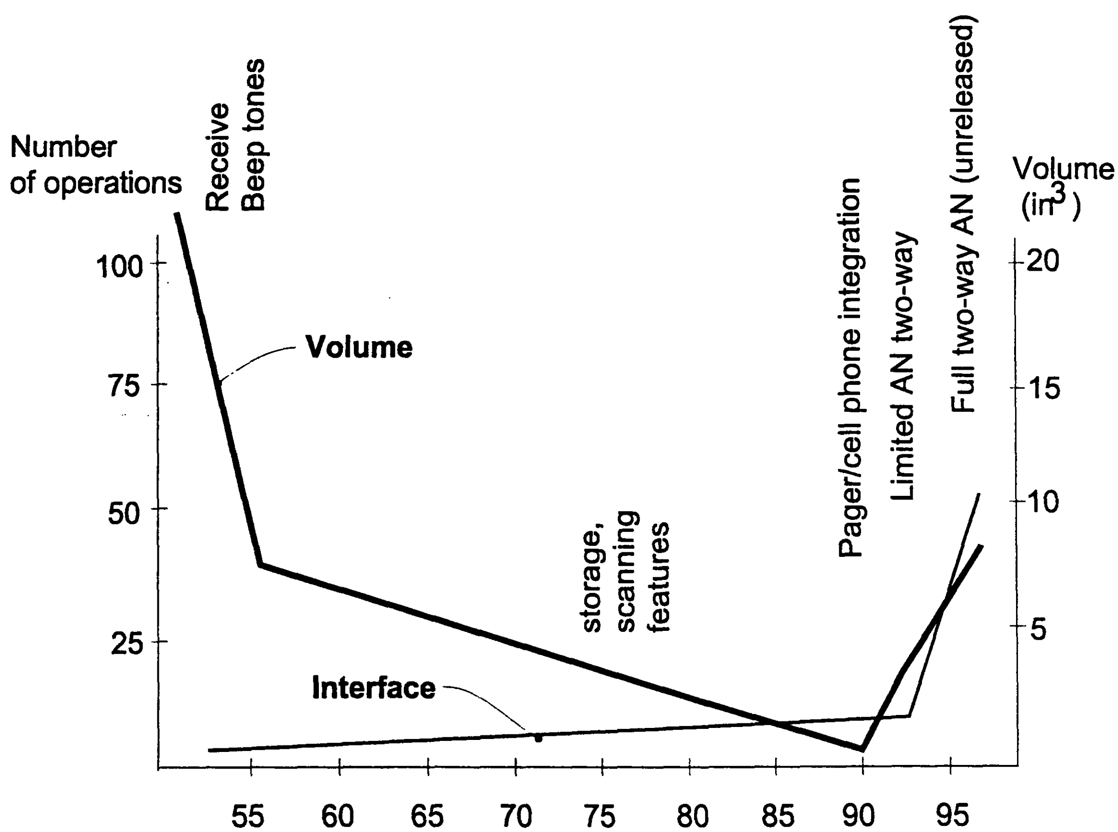



**Figure 13 - Pager Interface vs. Volume**

## 3.3 Products Beyond the ET

Below are four examples of products within fundamental industries that are at or beyond the ET. In each case the products are not wild new creations, but logical progressions in the flow of technologic advance. The suggestion is that the ET is also a phenomena inherent in the advance of technology.

### 3.3.1 Pagers

As shown above, the technologic development of pagers exhibits follows the theory and appears to have crossed the threshold in 1990.

### 3.3.2 Computers

All notebook computers are width-limited by the keyboard. The result is a host of new keyboard technologies that allow operation in a full size mode and compact when not in use. One such example is U.S. Patent #5,163,765 to the author (assigned to Apple Computer) for a Collapsible Keyboard. (See Appendix 1) This device takes advantage of the rather large inherent spaces between key cap contact areas. Standard size rigid key caps are insert molded onto an elastomeric frame disposed over a split printed circuit board. In the collapsed state the key caps nearly touch, allowing a normal full-size keyboard to fit in a width as small as 7.5." When the elastomer is stretched to full width, the key caps are spaced correctly for operation as a full sized keyboard. The most commercially successful product which has sought to address this problem is the IBM *Thinkpad*. IBM is apparently well aware of the issue. In 1995 the company released the IBM *Palmtop 1000*, a full-function computer of dimensions 6.25" x 3.5" x 1.25". The New York Times wrote glowingly of the device and reported its only weakness as a "chicklet-sized keyboard sized for use by a child[8]." Clearly, the discrepancy between the volume of computer electronics and the area for interface created by that volume is already a serious issue, one that will be increasing in the foreseeable future.

---

[8] *New York Times*, Technology Trends June 14, 1995

### 3.3.3 Telephone

There is currently much interest in cellular telephones that can provide access to Internet data. The problem is the lack of a usable interface that can provide the necessary functionality in form factor of a telephone. There is no such phone at this time, although the industry is actively seeking a solution[9]. It is expected that as soon as this functionality is available in a cellular phone it will also be desirable in desktop telephones. Cordless phones outsell corded phones by a factor of 2:1, a ratio that is increasing[10]. Cordless phones have the same interface problem as cellular phones, suggesting a $3 billion ET problem[11].

### 3.3.4 Interactive Television

Television and the computer are becoming integrated. There is currently no way to provide computer functionality into a TV remote control. The interface is just too big. Should a remote control for interactive television lack portions of the user interface? Or should it be a desktop keyboard that rests on your lap? The industry is actively seeking a solution[12].

---

[9] Hide Satoh, Sony Engineer in Interview. August, 1996

[10] *Dataquest* April, 1997

[11] Ibid.

[12] Mr. Nakano, Sony Engineer in Interview, March 11, 1996 San Jose, CA.

## 3.4 Magnitude of Economic Impact

The total value of products approaching the ET currently is approximately $124 billion, as determined by April 1997 Dataquest information:

- Pagers..................................$1.02 B/yr. (30% AGR)
- Cellular phones..................$1.2 B/Yr. (30% AGR)
- Cordless phones.................$1.7B/Yr. (36% AGR)
- Laptop computers..............$117B/Yr. (17% AGR)
- Handheld computers.........$3.3B/Yr. (26% AGR)

Aggregate growth rates are also provided, confirming that these products are in quickly advancing market segments.

# 4. Technology Gap

The time that transpires between input device paradigms (See Figure 5) can be seen as a technologic gap. As the gap progresses, the affects of the ET are increasingly pronounced until a new technology establishes a new threshold. This section discusses the nature of input device advance, input devices themselves and predicts size of the technology gap we are now entering.

## 4.1 Existing Ergonomic Keyboard Paradigm

The existing keyboard paradigm should be defined. The cornerstone of physical ergonomics is the dimensions of the human body, as determined by measured distributions. Traditionally, products are designed for the 90-95th percentile of the distribution. For input devices in general terms, the notion of a finger being associated with a finger-sized region has been central to the human/machine interface since the days of the first hand operated devices. The trigger of a rifle may well be the beginning of this paradigm. Regulations limiting the implementation of this paradigm have been established by the International Standards Organization (ISO). The ISO sets standards for a wide range of ergonomic issues with the express purpose of protecting individuals from fatigue or injury from poorly designed devices. ISO guidelines are used by ergonomic regulatory agencies, such as the German agency TUV to establish their approval or disapproval of a product. Many companies will not purchase products that do not meet ISO standards. However, standards for keyboards apply almost exclusively to desktop keyboards. This is assumedly due in part to the relative infancy of portable devices, and in part due to the lack of clarity as to the direction portable devices will take.

The following standards were investigated for applicability: MIL-STD-1280, ISO/IEC 10646, ISO/IEC DIS 14755, ISO/IEC 9995. A standard for portable (non-desktop) devices was not found. However, a summary of pertinent data from TABLE 5.3 BS 7179:Part 4:1990 of ISO/IEC 9995 Specification for desktop keyboards is provided as a reference.

- Housing and keytop surfaces matte finish, diffuse reflection factor between 0.15 and 0.75, using diffuse reflection chart.
- Specular reflection 45 gloss units or less.
- Keytop legends robust, durable, minimum height 2.6mm and minimum contrast ratio 3:1.
- Key spacing between 18mm and 20mm horizontally and vertically between center lines of adjacent keys.
- Keytop size minimum 12mm x 12mm if square (maximum 15mm) or 113mm$^2$ if not square.
- Keytop shape molded concave.
- Key travel between 1.5mm and 6m: between 2mm and 4mm preferred
- Key force between 0.25N and 1.5N between 0.5N and 0.6N preferred
- Keying feedback tactile feedback preferred audible shall be able to be switched off and should be adjustable in volume

There are as yet no standards for handwriting, voice or neural technologies.

## 4.2 The Advance Of Interface Complexity

While the ET exists independently from the advance of interface complexity, we live in an era of extreme interface growth. Advances occur in discrete steps in which there are few significant increments, shown in Figure 14.

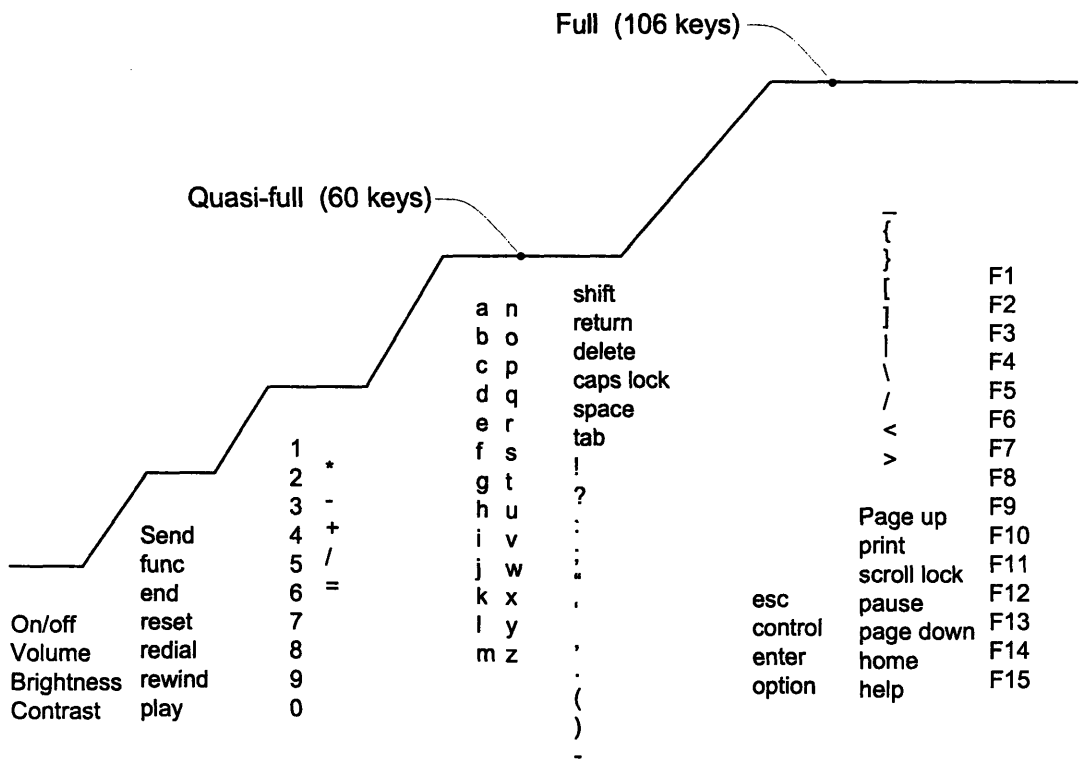



**Figure 14 - Advance of Function Complexity**

These significant increments may be categorized roughly as follows:

(The number in parenthesis indicates the number of characters typically associated with that advance. The television remote control will be used as an example.)

### 4.2.1 Introductory Product-Specific Controls.

Technologies are introduced with the minimal controls necessary for its implementation. Example: The first TV remotes included volume, on/off, and channel increment/ decrement.

### 4.2.2 Advanced Product-Specific Controls (1-4)

As a product is enhanced with additional functionality, the interface must grow to accommodate. Example: Mute and color controls were added.

### 4.2.3 Numerics (10)

The addition of a numeric pad offers significantly enhanced functionality Example: A numeric keypad was added.

### 4.2.4 Numeric operators (5-30)

Once numbers are available, many possibilities become available. Example: programmable features. Time set.

### 4.2.5 Alpha and associated characters (44)

Crossing the boundary at which alpha keys become necessary requires more than the alphabetic symbols. Alphabetic capability typically mandates the addition of a space bar, return, delete, and basic punctuation, suggesting the need for an additional 18 functions. This level of functionality is enough to provide a basic computer functionality, and therefore considered to be a quasi-full interface. This interface is typically found on handheld computers. Future example: Interactive television will require alpha characters and a cursor control unit.

#### 4.2.6 Full functionality (38)

Beyond the quasi-full character set there is the full desktop interface, consisting of approximately 106 keys.

## 4.3 Input Device Evaluation

Input devices are difficult to evaluate for two reasons.

Firstly, the quantity, quality and contextual aspects cannot be independently considered. The size, cost, degree of intuitiveness, speed of input, error rate, quantity of use, location of use, and ergonomic "quality" must all be considered to evaluate the level of applicability a device offers.

Secondly, input devices are largely subjective. Taking cursor control devices as an example, it is easy to find individuals who "hate" mice, trackballs, cursor control keys, joysticks and trackpads. Likewise, the distribution of input device aficionados is evenly distributed enough to sustain thriving product lines in each market segment.

For the purpose of this work, the task is simplified considerably by limiting the evaluation to determining whether or not a technology qualifies as a viable option: establishing a threshold of acceptability. Let us first define the evaluation parameters of acceptability for a portable input device:

- Small enough for the selected product
- Its cost must be appropriate for the overall cost of the specific product. Portable products vary in cost from $100 to several thousand.
- Contextually suitable for the task. Ex: Voice input is not appropriate for devices that will be used in public.
- Accommodates user mentally and physically, as opposed to requiring the user to conform. This is a broad category that encompasses learning rate, error rate, fatigue and input rate. With respect to meeting a threshold of acceptability, the vagaries of "performance" are subsumed by the more fundamental ergonomic issue of either accommodating the expectations of the user, or not.

## 4.4 Interface Technologies

In each of the examples in Chapter 3, the size of the keyboard defines the size, (or potential size) of the device. However, there are many other technologies that provide the same functionality, some more advanced. Therefore, with respect to the broader issue of the ergonomic threshold, the "key" question is: Does the existence of alternate input technologies obviate the importance of the traditional keypad interface being used to establish the validity of an ergonomic threshold? Handwriting recognition exists. Voice recognition exists. Both offer a smaller interface than a keyboard. There are a host of lesser-known alternate keyboard technologies. What effect do these other technologies have on the ET?

This section examines each alternative and summarizes why keypad size continues to be the most significant factor in determining product size and hence to establishing the validity of the Ergonomic Threshold theory.

## Scale the Keypad Smaller

The most direct way to increase the number of functions in a keypad is to scale the keys smaller, reducing key cap size and/or decreasing the distance between key cap centers. The results are obvious: There is an increased likelihood of accidental input. The keyboard is less comfortable use. The user feels constrained. Products with sub-finger-sized keypads suffer from the impression that they are toy-like, largely because they are, in fact, scaled for use by a child.

### *Theoretical Maximum for Ergonomic Information Density*

We must consider the existing keyboard paradigm and establish a theoretical maximum for key density.

The existing keyboard paradigm is given by the basics:

- each key results in one non-ambiguous operation.
- dimensions must accommodate operation by a human finger

Figure 15 shows the critical variables of a keyed interface.

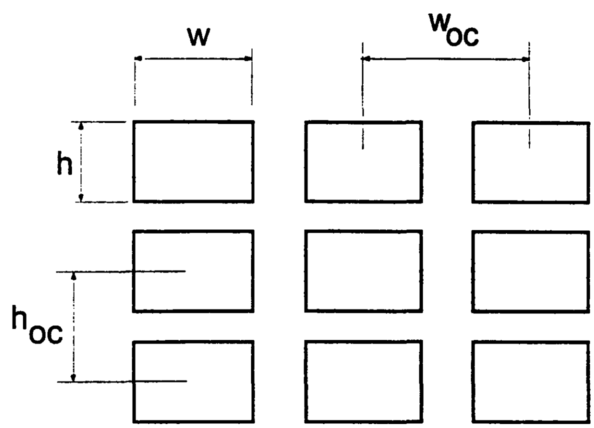



**Figure 15 - Generic Key Layout**

To establish a theoretical minimum, the dimensions of the keys themselves (w & h) can be reduced to approximately zero. Next, the distance between keys can be reduced to half the width of the human finger. Because the ultimate goal is functionality, it is important to keep theory within the realm of applicability. It is therefore worth noting that this implausible design may be implemented with a grid of pinholes with light emanating therefrom. When a finger covers the holes, light would reflect back into the hole and actuate the key.

Both modifications are shown in Figure 16. This is the smallest possible size that follows the existing paradigm. It allows each key to accommodate a finger and to result in a non-ambiguous operation.

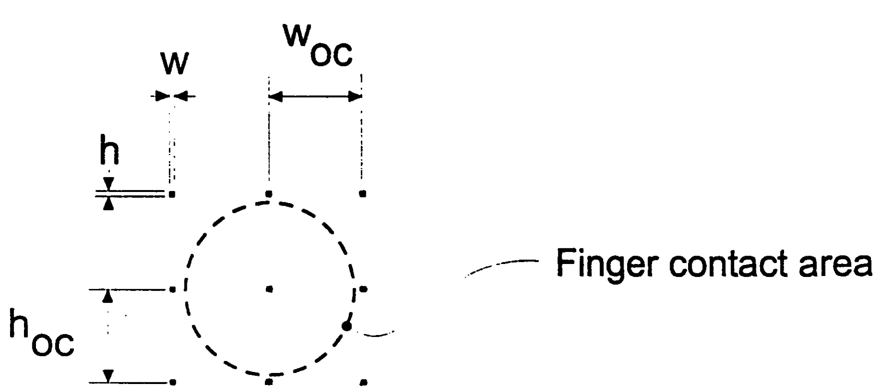



**Figure 16 - Theoretical Minimum Keyboard Layout**

Note: One can imagine spacing the pinholes at twice, or more, the density, resulting in a plurality of switch actuations per intended operation, using software to identify intention, with a commensurate increase in required finger placement accuracy. However, simple observation (made in the spirit of real-world engineering[13]) indicates that the threshold for accurate finger placement without concerted effort is about one-fourth of a finger width. Therefore, this proposed maximum theoretical density is liberal, as it fails to take this in to account.

Before calculating a theoretical maximum density, we must determine a finger contact area that represents an acceptable percentage of the user population.

### *4.4.1.2 Determining Ergonomic Finger Size*

The wide variations within human physiology makes ergonomic product design a statistical process. It is impossible to design for everyone. Companies typically design to meet either the needs of 90% to 95% of the populace. Because the author could not find a source for these values, they were determined experimentally.

The approach was to collect a random sampling of finger and thumb impressions using an inked stamp pad. A non-toxic, washable, children's light

---

[13] An important note to make at this time is that the range of natural human capabilities coupled with the learning curve, coupled with variations in the use patterns of daily life (such as using a device in a car, subway, or even variation due to using a supported thumb versus a non-supported finger) are significant. A rigorous analysis of these variables could determine the error rates associated with specific key resolutions and a thereby a statistically determined key error width, but the deviation would be so large that the exercise would be pointless. The simple rule of +/- 1/4 finger width offers a meaningful and realistic rule of "thumb."

yellow stamp pad was used to reduce the number of objections people might have to the experiment. It was actually quite surprising how readily people would ink their fingers and give impressions. Only one person declined. Under the assumption that only males would be in the top 10% of thumb size, thumb and index finger prints of 50 adult males were taken purely at random by starting at one end of a subway platform and asking each adult male on the platform. The raw data is shown in Appendix 2. The results are shown in Figure 17 and Figure 18, including the results of the subsequent calculation of maximum ergonomic density.

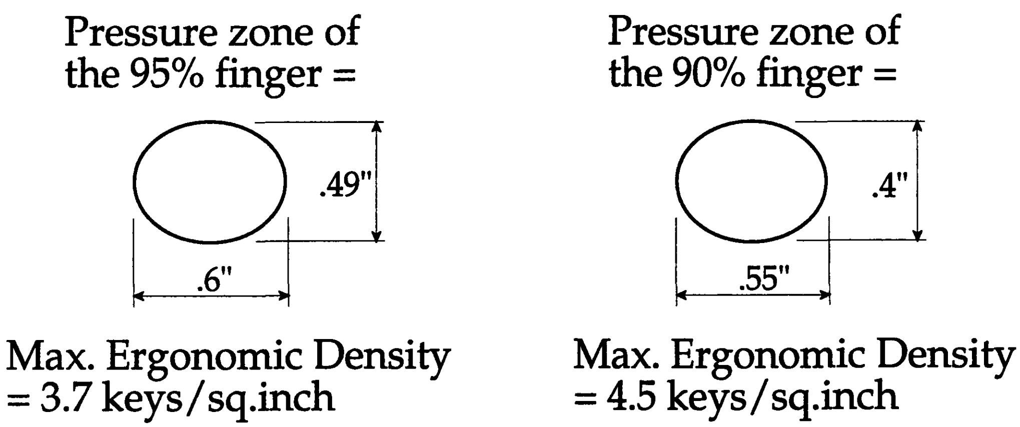



**Figure 17 - Results of Finger Study**

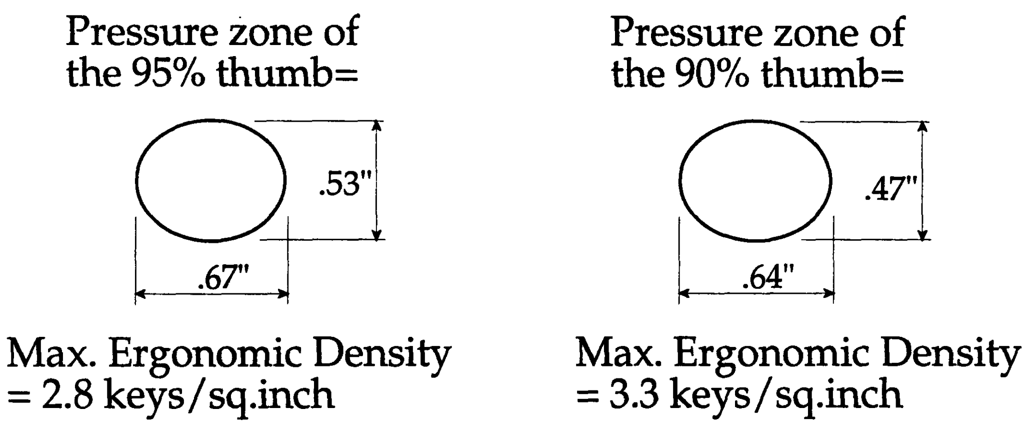



**Figure 18 - Results of Thumb Study**

Using the 90% finger, we can now calculate a theoretical maximum key cap density as outlined in Section 4.4.1.1. This theoretical maximum is approximately 7.5 keys/square, roughly twice the ergonomic maximum established here.

### *4.4.1.3 Determining Product Distribution*

An additional study was conducted to compare the results of the finger study to products that are currently sold. Twenty body styles representing over 50 product numbers were measured to determine their key density. (See Appendix 3) As shown in Figure 19, products clustered around 4.5 keys per square inch, indicating that most companies used the 90% finger size as design criteria.

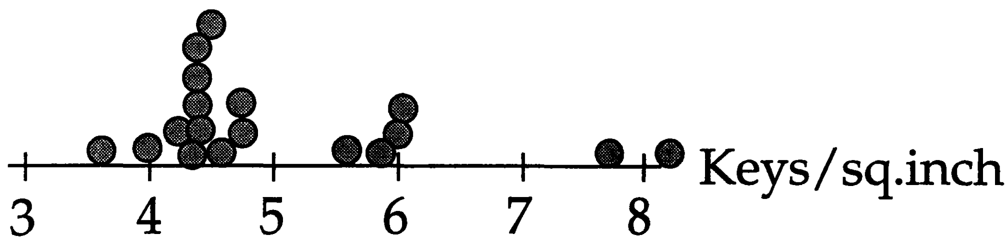



**Figure 19 - Hand held Product Distribution**

### *4.4.1.4 Summary regarding Traditional Keypad Miniaturization*

Although the traditional keyboard concept can theoretically be reduced to 7.5 keys per square inch, practical limitations restrict keyboard miniaturization to about 4.5 keys per square inch to conform to the physical needs of 90% of the population. Most products are designed to meet this 4.5 keys per square inch criteria.

### 4.4.2 Chorded keyboards

Chorded keyboards consist of a relatively small number of keys, often linearly disposed to conform with the resting position of the fingers on the human hand. The fingers operate in combination to comprise each input, similar to playing a musical instrument, and offering $2^N$ number of different inputs where N is the number of keys. Examples are the Twiddler[14] and the Data Egg[15].

---

[14] HandyKey Corp. 141 Mt. Sinai Ave., Mt. Sinai, NY 11766

[15] DataEgg InHand Development Group. Pasadena, CA 91105 USA

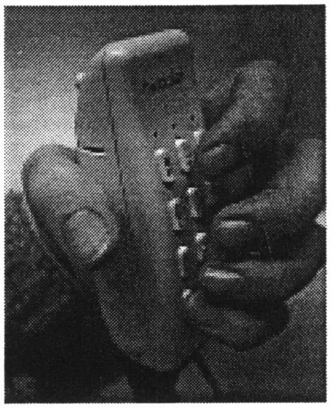


**Figure 20 - The Twiddler**

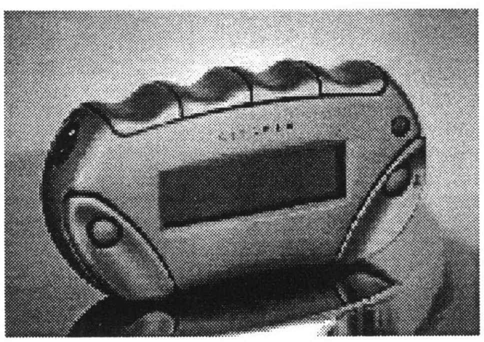


**Figure 21 - The Data Egg**

The primary problem with chorded solutions is that it requires the user to memorize many input combinations and to develop the necessary motor skills. While the chorded keyboard provides a workable - and in some instances a preferred - solution for extremely dedicated users, it is impractical and requires an unrealistically large amount of practice for most.

Extensive research has been conducted, generally concluding that the devices are useful after significant training. (Gopher and Raij found skilled typists at approximately 20 wpm after 10 hours use and 30 wpm after 26 hours of use[16].) Using the Dvorak keyboard as precedent, it is clear that users are extremely reluctant to learn new device skills[17]. (Note the author cites the Dvorak precedent with great hesitance because research uniformly indicates it offers only slight advantage over the QWERTY standard[18]. Nonetheless, the Dvorak reference seems valid in this example because the public *perception* is that it offers significant advantages, yet almost no one extends the minimal effort required to try to learn it.

Chorded keyboards force the user to conform both mentally and physically, entailing significant compromise on the user's behalf.

### 4.4.3 Modal Solutions

There are a variety of modalities that have been used for keypad input as a means to increase functionality with a given number of keys.

• Time variance modality varies the time between key operations to define the output function. This is non-intuitive and restricts the pace at which an operator may work.

• In shifted key modality the output of a given key varies as a function of key or keys pressed before it (i.e. a shift key). While a shift key doubles the possible outputs of a keypad, it also doubles the number of keystrokes required.

---

[16] Gopher, D., and Raij, D. (1988) Typing With A Two-Handed Chord Keyboard: Will The Qwerty Become Obsolete? IEEE Transactions on systems, and and Cybernetics, 18, 601-609.

[17] Paul A David, Clio and the Economics of QWERTY, 75 Am Econ. Rev. 332 1985

[18] Donald A Norman and David E. Rumelhart Studies of Typing from the LNR Research Group. in Cognitive Aspects of Skilled typewriting 45, (William E. Cooper ed. 1983); and A. Miller & J. C. Thomas, Behavioral Issues in the Use of interactive Systems, 9 Int. J. of Man-Machine Stud. 509 (1977).

• Force sensitive modality incorporates a plurality of force-levels to a single button, and thereby a plurality of functions. Tests show that there is a wide variation in the forces naturally applied by users and wide variations in the levels of force sensitivity between users[19]. Even once a force parameters are established for a given user, the solution is highly non-intuitive. For these reasons, force sensitive keys are not widely applicable, nor desirable.

Time variance and shift key modalities are exemplified in the Half-QWERTY keyboard of section 4.4.5.

### 4.4.4 On-screen Keyboards

There are two types of on-screen keyboards. Indirect on-screen keyboards use a mouse, mouse-equivalent, or arrow keys to navigate a cursor. Direct technologies allow the user to touch the screen directly.

#### *4.4.4.1 Indirect*

This area has been extensively studied with consistent results[20],[21],[22]: In general, the input device is far more relevant than the layout of the keys.

A simple method to provide many functions is to display them on-screen and provide means to scroll between them sequentially, as shown in Figure 10.

---

[19] Force sensitivity and response, International Ergonomic Compendium 1992

[20] *On-Screen Keyboards: Which Arrangements Should Be Used?* Laurie Quill, David Biers, *Presented That The Human Factors And Ergonomics Society 37Th Annual Meeting,* 1993

[21] Card, S.K, English, W.K., and Burr, B.J. (1978) Evaluation of mouse, rate-controlled isometric Joystick, step keys and text keys for text selection on CRT. *Ergonomics* 21 (8), 601-613.

[22] Norman, D.A., & Fisher, D. (1982). Why alphabetic keyboards are not easy to use: keyboard layout doesn't much matter. *Human Factors,* 24, (5). 509-519

While inexpensive and requiring little-to-no dedicated area, the method is extremely tedious.

#### *4.4.4.2 Direct*

Tapping directly onto a touch screen, either with finger or stylus has also been the subject of extensive research, such as that done at the University of Geoulf[23]. This work is particularly interesting because it is generally applicable to all keyboards used with one finger. The work focuses on a piece of software that computes theoretical typing speeds of novice and expert based on Fitts law, Hick-Hyman law and a 27 x 27 matrix of digraphs indicating relative frequencies for each letter pairing in common English, plus space bar. By inputting the location and size of each key with respect to each other, typing speed may be estimated. The predicted accuracy of the software is moderate. The only serious discrepancy is with QWERTY, where true novice users cannot be found. The results are given in Figure 22.

[23] *Text Entry Using Soft Keyboards*, (1997) I. Scott MacKenzie, R. William Soukoreff, and Shawn Zhang Department of Computing and Information Science University of Guelph

| | Novice (predicted) | "Novice" (measured) |
|---|---|---|
| QWERTY | 8.9 | 21.1 |
| Dvorak | 8.7 | 8.5 |
| ABC | 9.6 | 10.7 |
| Fitaly | 9.7 | 8.0 |
| Telephone | 9.1 | 8.0 |
| JustType | 9.8 | 7.0 |

**Figure 22 - Results of 1997 Geoulf Study**

Because it touches on various peripheral topics, the conclusion to the work is offered in full:

> Text entry on small mobile systems remains a challenge for computing systems of the future. Stylus tapping on a soft keyboard offers easy entry; however, rates are moderate at best and a keyboard must be presented on the system's display, thus occupying screen real estate. Although expert entry rates may reach 30 wpm for the QWERTY layout, or in excess of 40 wpm for optimized layouts, such rates are probably not sustainable. Because eye fixation is a requirement of interaction with soft keyboards, fatigue may prove a factor with prolonged use. Novice entry rates are in the 7 to 10 word per minute rate for most layout permutations. However, experienced users of desktop computers may enter text with an immediate rate of about 21 words per minute on a soft keyboard with a QWERTY layout. This suggests that the venerable QWERTY layout is here to stay, both for physical keyboards on desktop computers and for soft keyboards that support stylus tapping.

Other research has found significant differences in performance between flat (on-screen or flat-panel) keyboards and physical keyboards[24]. Both novice and

[24] Julia Barret and Helmut Krueger "Performance effects of reduced proprioceptive feedback on touch typists and casual users in a typing task", BIT; Behaviour and Information Technology. 13(6):373-381, 1994.

skilled typists had significantly lowered performance on keyboards that lacked kinesthetic and tactile feedback. Performance did not increase with practice.

Experiments evaluating input performance with the size of on-screen keyboards determined that performance deteriorates from approximately 20 wpm to 10 wpm as keyboard width varies from 24.6 cm to 6.8 cm wide[25].

### 4.4.5 Half-QWERTY

**Figure 23 - Half QWERTY Keyboard**

Shown in Figure 24, the half QWERTY keyboard allows users to transfer two-handed typing skills to single-handed use[26].

[25] Andrew Sears, Doreen Revis, Janet Swatski, Rob Crittenden, and Ben Shneiderman. "Investigating Touchscreen Typing: The Effect of Keyboard Size on Typing Speed", BIT; Behaviour and Information Technology. 12(1):17-22, 1993.

[26] Available from The Matias Corporation

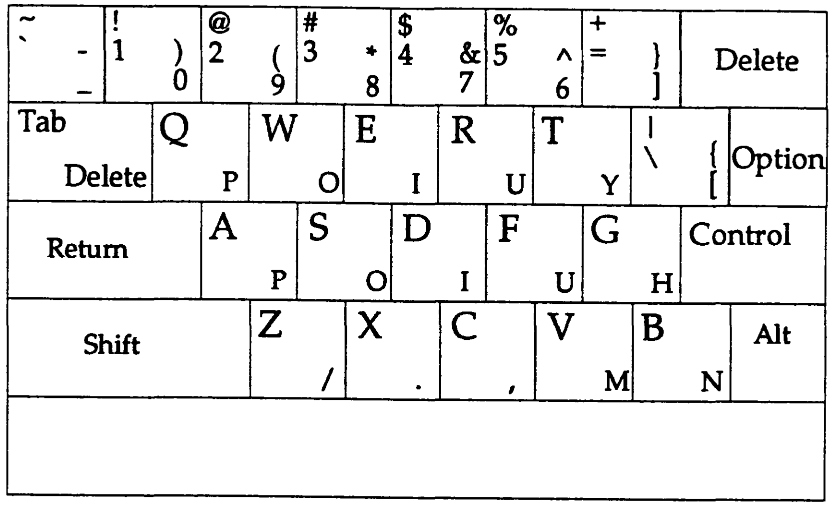



**Figure 24 - Half QWERTY keyboard**

The design reduces size by a factor of two. One study show subjects achieving 41-73% their two-handed speed after 10 hours of use[27]. The following text is included partially in reference to Section 4.4.3, as an example of a modal application, and of how non-intuitive and complex they can be. According to the study:

> Pressing and releasing the spacebar within a time-out generates a space character. The time-out reduces the number of erroneous spaces generated as a side-effect of using the space bar as a modifier key. It is often the case that a typist will depress the space bar with the intention of mirroring the state of another key but then change their mind and release. Without the time-out, such actions would result in an unwanted space character. For the study, the time-out was 267 ms. Modifier keys (such as shift and control) are supported via "latch" mechanism, commonly known as "sticky keys." Depressing and releasing a modifier key once activates it for the next key pressed. Depressing it twice locks that key until it is unlocked by depressing it again. Sticky keys allow one key to do the work of several.

[27] Half-Qwerty: A One- Handed Keyboard Facilitating Skill Transfer From Qwerty, Edgar Matias, Scott MacKenzie, Wiliam Buxton, Presented at The Human Factors And Ergonomics Society 37Th Annual Meeting, 1993

The device provides insufficient miniaturization to be of interest to the applications of interest, although it provides excellent utility to disabled users.

### 4.4.6 Disambiguation Keyboards

There are a variety of solutions in which the character input by any given key is modified as a consequence of the previous letters entered, based on likely probabilities within a given language, an approach called disambiguation. JustType is one such technology, provided by Aiki Ltd, a Seattle company. With JustType, the probability of the device guessing correctly increases with the length of word. However, the first letter of each word in always a problem, such as requiring three key presses to access the letter "R". Obviously any key switch technology, including the object of this invention, may be combined with a Smart key solution. While an intriguing solution for alpha-only applications, implementing a quasi-full character set is problematic.

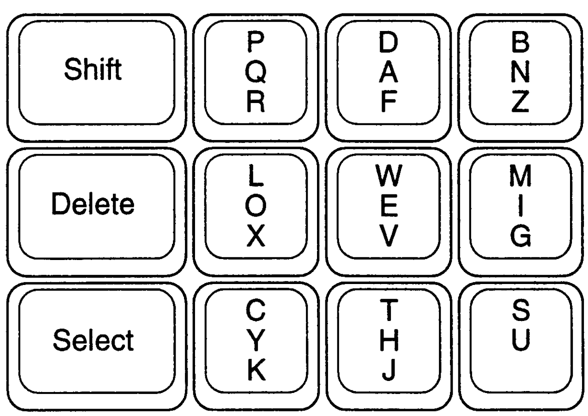



**Figure 25 - Aiki Keyboard**

Aiki Ltd. has conducted limited performance analysis. They paid a temporary employee to use the device for two days and report 30 wpm performance. The company also report 70 wpm after repeatedly typing one sentence. The 1997 Geoulf study (Section 4.4.4.2) ranked JustType as the worst-performing technology of the research at 7 words per minute.

### 4.4.7 Handwriting Recognition

There have been many arguments made against handwriting recognition, irrespective of the technology. The most significant are that writing by hand is becoming an obsolete skill, people are bad at it, and it is inherently slow. Indeed, a relatively high percentage of people's handwriting is illegible by human readers who are allowed ample time to figure out what was written. It is unreasonable to expect a machine to accomplish the same task in real-time. However, by watching the stroke sequence, word context sensitivity and even marginal handwriting can be recognized. The *Newton MessagePad 2000* from Apple Computer and CIC's *Handwriter Manta* are two examples of the recent maturation of this technology[28]. These products both require a significant amount of computational ability, the details of which are discussed below.

Far less computation is necessary to recognize custom, or reduced information character sets, such as *Graffiti* . Rather than requiring the machine to recognize the wide range of strokes used to describe alphabetic characters, *Graffiti* requires the user to write each character with a carefully defined simple, relatively quick motion that approximates the graphic of the character. The start points, stroke pattern and end points are carefully defined to not coincide with alternate possibilities, resulting in a compromise between the demands placed on the

---

[28] PC World, *Better Dictation , Less Typing* June 1997 v15 n6 p61(1)

machine and those placed on the operator. The cost/performance offering between these two devices provide insight into the nature of the tradeoffs.

At issue are the restriction on the applicability for use of this excellent technology, either by price, or by use context, both discussed below.

### 4.4.8 Voice Recognition

During the development of this technology, there have been tradeoffs made between the vocabulary size, continuous vs. discontinuous speech, degree of training necessary, user dependence vs user independence, and ultimately the amount of computing horsepower necessary for implementation. Until very recently the tradeoffs rendered VR as extremely limited[29]. A major advance in voice Recognition technology occurred within a month of the completion of this document. Dragon Systems demonstrated its 30,000 word , continuous speech recognizer, *NaturallySpeaking*.[30]

Independent of the technology itself, is the issue of VR's applicability by cost and by context. Voice is simply inappropriate input device for all situations. For example, portable products are commonly used in airplanes, cafes, meetings, libraries, and lecture halls. None of these locations are appropriate for a vocal human/machine interface. Therefore, regardless of any technologic advances that

---

[29] This thesis was written entirely with a voice recognition system. The system requires 24 megabytes of RAM, 50 megabyte of non-volatile memory, and a 64 bit wide processor at 80 MHz. Using this hardware, the speed of the recognizer is severely limited, requiring the user to pause unnaturally after each word. Its accuracy is poor. (There were 33 corrections made in this paragraph.) The microphone must be positioned and angled *precisely*. The room must be quiet. The user cannot have a cold, or even a scratchy throat. Other problems are: false starts on the part of the speaker, mispronunciations of any part of the utterance, dialect effects, wind, breathing, and random noises.

[30] PC World, *Cheese of Staff* June 10 1997 v15 n6 p61(1).

may be achieved, an alternate input device will be needed for many portable products

At issue are the restrictions regarding the applicability of this excellent technology, either by price, or by use context. The question appropriate to this thesis is: when will VR be able to provide performance/cost parity with handwriting recognition or a keyboard technology? This is discussed in Section 4.7.

### 4.4.9 Neural Human-System Interface

Much work is being done on neural human-system interface, primarily for application with handicapped[31]. Adequate signal processing power is now available at reasonable cost to implement in near-real time a wide range of spectral, neural network, and dynamic systems algorithms for extracting information about psychological state or intent from multidimensional EEG signals, video images of the eyes and face, neural implants, and other psychophysiological and/or behavioral data[32]. While fascinating, it is too early to make an evaluation of the commercial applicabilities of this technology.

### 4.4.10 Summary of Input technologies

The goal of technology is to augment the natural abilities of human kind. In this way, input devices are unique: Input devices technologies don't augment human ability. Instead, they work very hard merely to allow humans to express themselves at their normal rate. There is no technology known that enables

---

[31] Handicapped Neural Database

[32] NHSI Work at Yale

human expression to occur more quickly or of higher quality than it occurs naturally.

The goal of input technologies, therefore, is to minimize the barriers to natural human expression. The range of technologies that enable this to occur is surprisingly small. Many technologies demand relatively high levels of physical and/or mental effort in addition to the expression itself. Not only do these obstacles slow down the rate of expression, but they cause it to deteriorate in quality. The goal on any input device should be to not get in the way.

## 4.5 Compromise/No-compromise Devices

This thesis asserts that a requisite characteristic of a mainstream product is that it must accommodate the user mentally and physically, as opposed to requiring the user to conform to it. Devices which meet this standard are considered "no-compromise" devices. No-compromise devices have the capability of quickly becoming transparent to the user. These include:

- Standard keyboard with key density lower than ~4.5 keys / square inch
- Hand writing recognition
- Voice Recognition

On the other hand, compromise devices, while they may be very clever, will be relegated to niche markets and the greater the compromise, the smaller the niche. These include:

- Standard keyboard with key density higher than ~4.5 keys / square inch
- Disambiguation Methods
- Modal Methods
- Chorded Methods

This is to say that the world of mainstream, high quality, no-compromise input technologies is quite small.

## 4.6 No-compromise Technologies by Application

Figure 26 shows the no-compromise interface options for various products after they require a quasi-full interface.

| | Standard | Hand-writing | Voice |
|---|---|---|---|
| Internet Cellular | × | × | ○ |
| Internet Cordless phone | × | × | ○ |
| Wearable | × | × | ○ |
| Pager | × | × | ○ |
| TV remote | × | × | ○ |
| Smart card reader | × | — | ○ |
| Portable fax | × | × | ○ |
| Standard telephone | × | × | ○ |
| Hand held | — | ○ | ○ |
| Laptop | ○ | ○ | ○ |
| Dental | — | × | ○ |
| Security | — | × | ○ |
| Airline | — | × | ○ |
| Aerospace | — | × | ○ |
| Oven/Microwave | — | × | ○ |

○ Desireable option

— Functional option

× Not an option

**Figure 26 - Interface Options for Various Products**

Applying the standards discussed in this chapter, one of three weights are assigned, desirable, functional, or not an option. In general:

### 4.6.1 Keyboards

The existing keyboard paradigm is physically too large to provide a sufficient number of keys inside of the area available for many full or quasi-full interface products without compromising on the usability of the product.

Each of the cases cited here are supported by specific corporate interest in a new-paradigm keyboard, discussed in Section 5.1.

### 4.6.2 Hand Writing Recognition

HR is a viable, strong technology that has limited applicability, because of context. Writing is a dedicated two handed operation. Writing into a telephone or as means to communicate with a worn computer, pager or TV remote would be awkward, if not implausible much of the time.

### 4.6.3 Voice Recognition

Despite the significant limitations of VR as discussed above, verbal communication is natural and easy. While alternative inputs may always be necessary ( because portable products are commonly used in public places) voice is the smallest, ultimately less expensive, most natural technology under consideration, with extremely wide applicability.

## 4.7 Predicting Cost Parity

In addition to the context issue, there is cost, the time at which these technologies (VR and HR) will be available at approximate cost parity with the keyboard. When will these technologies be available for use in mainstream products?

## Hand writing recognition

The availability of a high-quality hand-writing product (*Newton*) allows us to identify how much computing power is necessary to accomplish the task. It is then relatively simple to estimate the cost of similar computing ability Hand writing recognition is already a commercially viable interface. However, determining the "cost" of the interface is difficult to determine because the screen, and processor are needed regardless of the interface. One way to establish processor cost is to look at the difference between the computational ability required by the operations normally performed on a hand-held device and that required to perform the recognition task.

To achieve its high standard of HR, the Newton 2.0 uses a powerful Strong Arm SA-110 RISC processor running at 162 MHz, with a quantity price of approximately $50. Based on the well-established Moore's curve and an increase clock speed (along the cost/performance curve) of 10% annually[1], handwriting recognition should approach the cost of keyboards in the year 2005.

## Voice Recognition

The same approach may be adopted to calculate the crossover of VR, with a keyboard, using the *NaturallySpeaking* product as reference. We will compensate

---

[1] [1]*The National Technology Roadmap for SemiConductors, 1994*, Semiconductor Industry Association

for the fact that the system still falters, and has several additional real-world issues, noted above, to accommodate, by assuming a doubling of computational ability. The system currently requires a Pentium 133 with a sound card, suggesting that voice recognition will be available for the price of a keyboard about the year 2010. Note: This is NOT a graph of when these technologies are expected to enter these products. It is a graph of when these technologies will be available at similar cost. The former is a marketing issue, the latter technologic. As a final note, there are non-linearities expected as each of these curves approach their asymptotes. However, this level of detail becomes too speculative for this work.

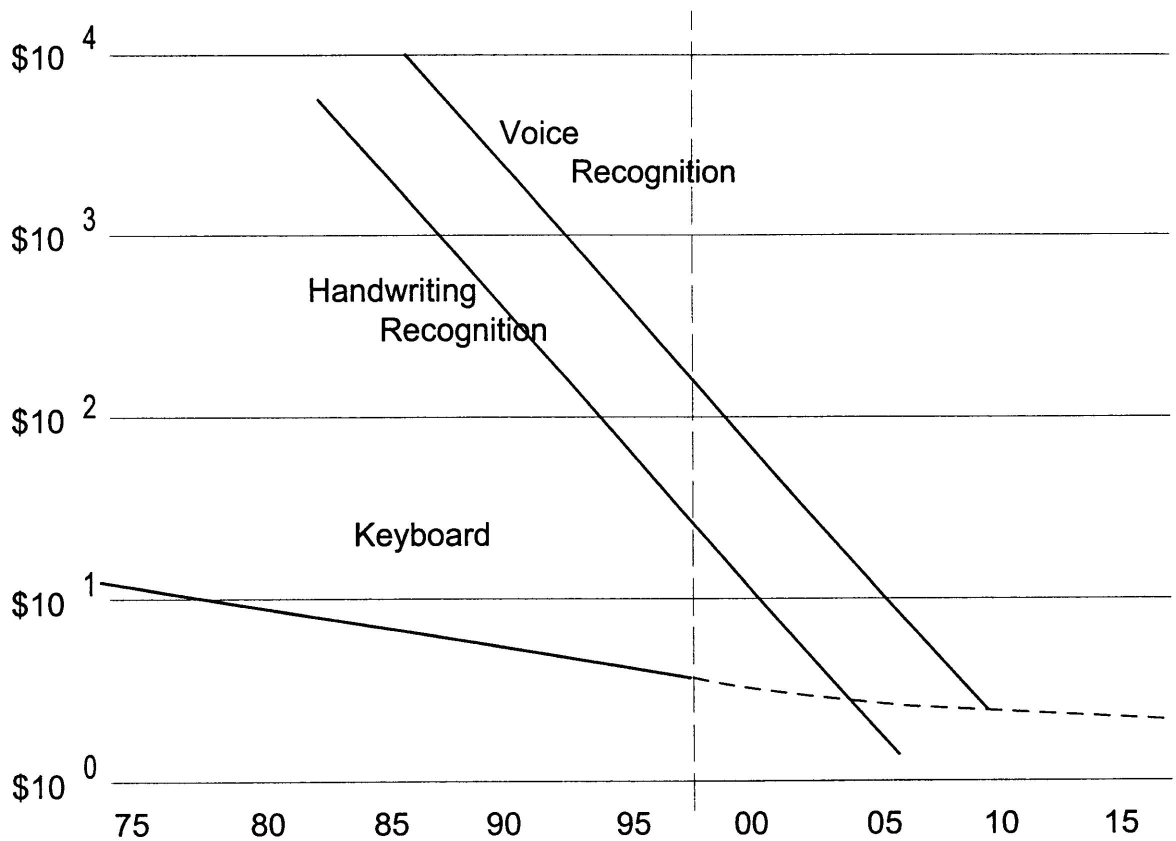



**Figure 27 - No-Compromise Technologies, Cost vs Time**

# 5. Chapter Four - Gap fillers

Three generations of keyboard advancement are presented:

- Sub Miniature Ergonomic Keyboard - increases key density by one order of magnitude without reducing ergonomic quality, or increasing cost.

- Augmented Reality Keyboard for wearable computing - solves problem that renders traditional (non-chorded) keyboards unsuitable for use in wearable computing applications.

- Miniaturized Augmented Reality Keyboard- reduces size of the tactile keyboard above by another order of magnitude.

## 5.1 Sub Miniature Ergonomic Keyboard

The Sub Miniature Ergonomic Keyboard (SMEK) here described has been given US patent #5,612,690. (Appendix 4) The subject matter of the patent, and the significance to this thesis, is that this is the first keyboard to offer full-sized keys in a non full-sized device. The design enables a next-generation of product miniaturization without causing ergonomic deterioration. Moreover, the SMEK is generic and fundamental, providing a lasting advance to the art of keyed input. As will be shown, the SMEK qualifies as a new input device paradigm because it 1) violates the existing paradigm 2) does not entail tradeoffs between parameters, but improvements in each. Accordingly, it provides performance incongruous with that which precedes it.

### 5.1.1 SMEK Conception

The Sub Miniature Ergonomic Keyboard was conceived in response to the fundamental observation of this thesis: products become smaller and more sophisticated while the human hand remains unchanged. The original functional requirements, in order of importance were:

- Must be smaller than currently conceived.
- Must be intuitive: no memorization, or training to use, as with chording keyboards.
- Must accommodate a large adult human finger comfortably.
- No modalities, such as key shifting, or time-based modifications
- No complex motions, such as side-to-side or rocking.
- Low cost.
- High reliability

In summary, the objective was to develop a small keyboard that was as intuitive and comfortable to use as a large keyboard.

### 5.1.2 Passive Chording Concept

The SMEK achieves the theoretical maximum key density, nearly twice the practical limit, as described in Section 4.4.1.1 by introducing the concept of passive chording. Traditional chording forces the user to actively remember different combinations to achieve each desired output. With passive chording these combinations are hard-wired into the device. As shown in Figure 28, each key is one forth the size of a traditional key. To access the "7" key the user would press the "7" graphic, thereby actuating four keys associated with the "7" and contacting a key area larger than a 95% percentile thumb.

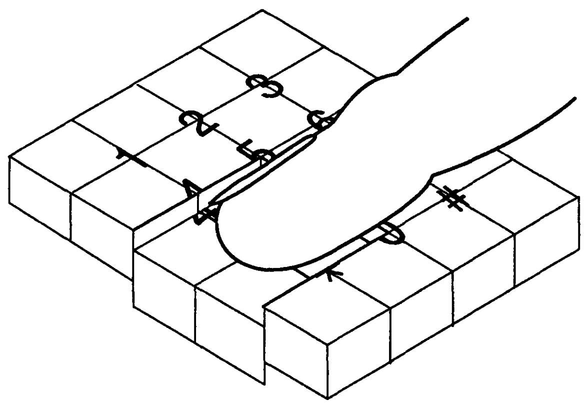


**Figure 28 - Passive Chording Concept**

Technically, the device does not require actuation of all four adjacent keys to correctly identify an intended operation. Any two diagonally opposed keys can be used to uniquely identify the associated combination key. The electronics are designed to take advantage of this fact, allowing the device to function correctly, despite the user not pressing with sufficient force and/or accuracy to actuate all four associated keys, significantly increasing the robustness of operation.

Equally important: in the theoretical example of Section 4.4.1.1, inaccurate finger placement would result in the actuation of an adjacent (zero-area) sensors, resulting in ambiguous intention. Because of the curved nature of the finger, finger placement inaccuracies of 1/4 finger width are insufficient to cause erroneous input.

This scheme provides the theoretical maximum density of 2F-1 key caps in a linear width that would normally support F keys, where F is the width of the human finger. This is equivalent to $4F^2 - 4F+1$ keys in an area that would normally support $F^2$ finger-sized keys, an improvement of approximately a factor of four.

## Auxiliary Use Of Individual Keys

In order to use each quarter key individually, the centers of each key are elevated by a distance approximately equal with the stroke length, as shown in Figure 1. The Pythagorean distance between contours provide space for a finger to rest comfortably between four adjacent key caps. The contours provide a gentle guide for the finger to the interstitial key.

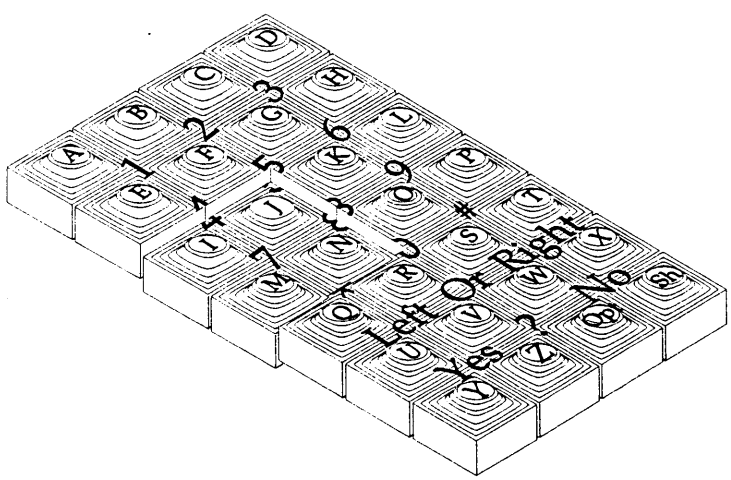



**Figure 1 - Key Centers Elevated**

Simultaneously, the elevated key cap contour offer physical isolation from the adjacent keys in the Z axis, as detailed in Figure 30, an exaggerated view of a finger pressing on an elevated key center. The elevated center maintains physical isolation from adjacent key caps, despite the natural compression of the finger.

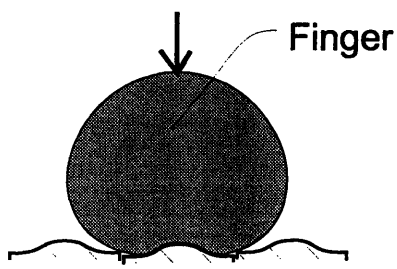



**Figure 30 - Functionality of Elevated Key Centers**

5.1.4 SMEK Exceeds Theoretical Maximum Density

Because SMEK violates the existing paradigm (by using more than one key to provide one operation) it also violates the theoretical maximum ergonomic keypad density. The improved functionality is substantial. Using the techniques here described the SMEK achieves a density of 17.9 keys per square inch, while still offering comfortable access to each function with a single touch of a digit the size of a 95th percentile thumb. This is sufficient to provide 67 independently actuatable full-sized keys in an area 40% smaller than a standard credit card, as shown in Figure 31.

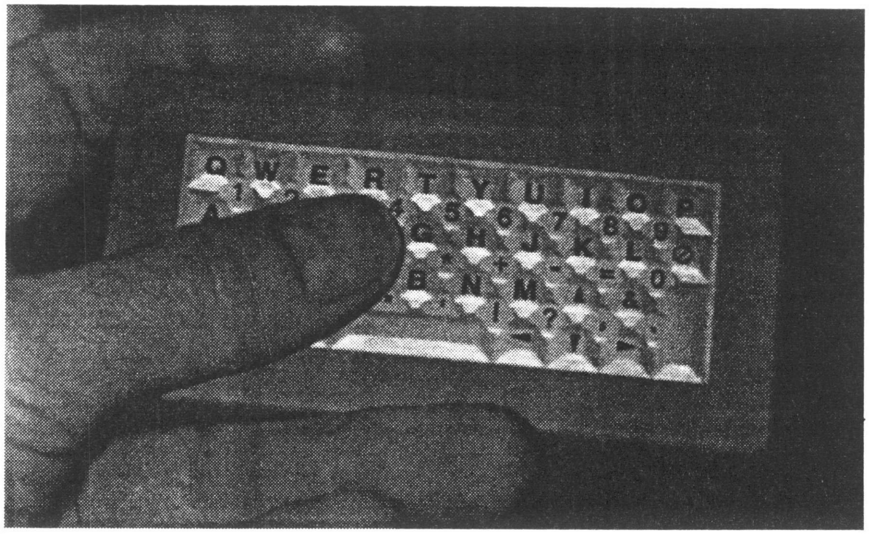


**Figure 31 - Prototype SMEK for PC**

### 5.1.5 Encoding

When first conceived, the intent was for each key associated with an interstice to be used to identify that interstice. It was quickly realized that all four keys are not necessary. The keypad was designed to operate using the minimal information content necessary to uniquely identify any combination key. As shown in Figure 32, any opposing diagonals are sufficient, providing redundancy in the coding scheme.

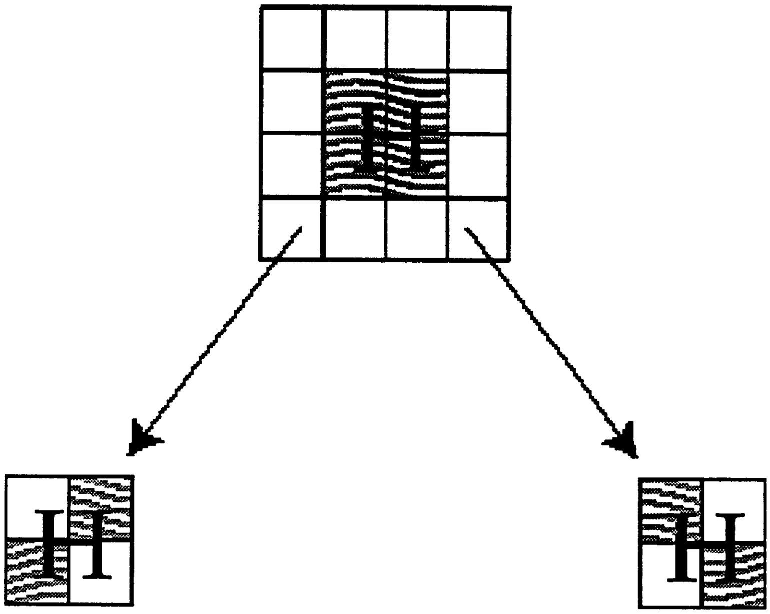



**Figure 32 - Diagonal Identifiers**

### *5.1.5.1 Analogy to Nature*

Mother nature is well-known for efficient engineering. For this reason, mimicking nature is often an effective design technique. While the author cannot claim to have intentionally considered this goal in the design of the SMEK, there are similarities that can be drawn between passive chording and DNA coding. In order to increase the robustness of the DNA code, amino acids are comprised of a variety of codon triplets. The redundancies are used to reduce the error rate as information is transferred. Figure 33 shows the table of codons used to the amino acids that are the basis of all life.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| UUU | Phe | UCU | Ser | UAU | Tyr | UGU | Cys |
| UUC | | UCC | | UAC | | UGC | |
| UUA | Leu | UCA | | UAA | Leu | UGA | Tryp |
| UUG | | UCG | | UAG | | UGG | |
| CUU | Leu | CCU | Pro | CAU | His | CGU | Arg |
| CUC | | CCC | | CAC | | CGC | |
| CUA | | CCA | | CAA | GluN | CGA | |
| CUG | | CCG | | CAG | | CGG | |
| AUU | Ileu | ACU | Thr | AAU | AspN | AGU | Ser |
| AUC | | ACC | | AAC | | AGC | |
| AUA | | ACA | | AAA | | AGA | Arg |
| AUG | Met | ACG | | AAG | Lys | AGG | |
| GUU | Val | GCU | Ala | GAU | Asp | GGU | Gly |
| GUC | | GCC | | GAC | | GGC | |
| GUA | | GCA | | GAA | Glu | GGA | |
| GUG | | GCG | | GAG | | GGG | |

The SMEK is likewise formed of a set of redundant codes that are used to reduce the error rate as information is transferred. Figure 34 shown a table for output from the SMEK as a function of the keys pressed.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| QS WA | 1 | )X (C | % | AX ZS | @ | o, =. | & |
| WD SE | 2 | Z( ShX | $ | SC XD | # | PL DeO | 9 |
| Sp1 Sp2 Sp3 | Space | GN BH | * | &Re RiRe | : | LRe &De | 0 |
| | | XD SC | # | VSp BSp | " | BH GN | * |
| Re Re Re | Return | GY TH | 5 | NRi MSp | ! | NJ HM | + |
| De | Delete | YJ HU | 6 | MDo RiUp | ? | MK JUp | - |
| EF DR | 3 | JI UK | 7 | 5+ *6 | H | DV CF | \ |
| RG FT | 4 | IL OK | 8 | 2\ #3 | D | VG FB | / |

**Figure 34 - Coding for SMEK**

While unintentional, it is still interesting that two highly compact data transmission means share this feature. There is a certain "rightness" to advanced engineering systems mimicking biologic systems that have already been refined for millions of years.

### 5.1.6 SMEK Ergonomics

It has been shown that key densities greater than 4.5 keys per square inch are compromised interfaces. Keys smaller than the contact area of the finger are less comfortable to use than objects that are similarly sized to the finger. Figure 35 shows the advantages of the SMEK in terms of key density and ergonomic quality. Eight keyboards are graphed in order of descending size and plotted against key density and key-to-finger contact area ratio. While overall size maps consistently with the ratio between key area and finger, there is a dramatic discontinuity in ergonomic quality. The sub miniature ergonomic keyboard, while offering key density second only to the infamously difficult-to-use calculator watch, offers a contact area equal to that of a desktop keyboard.

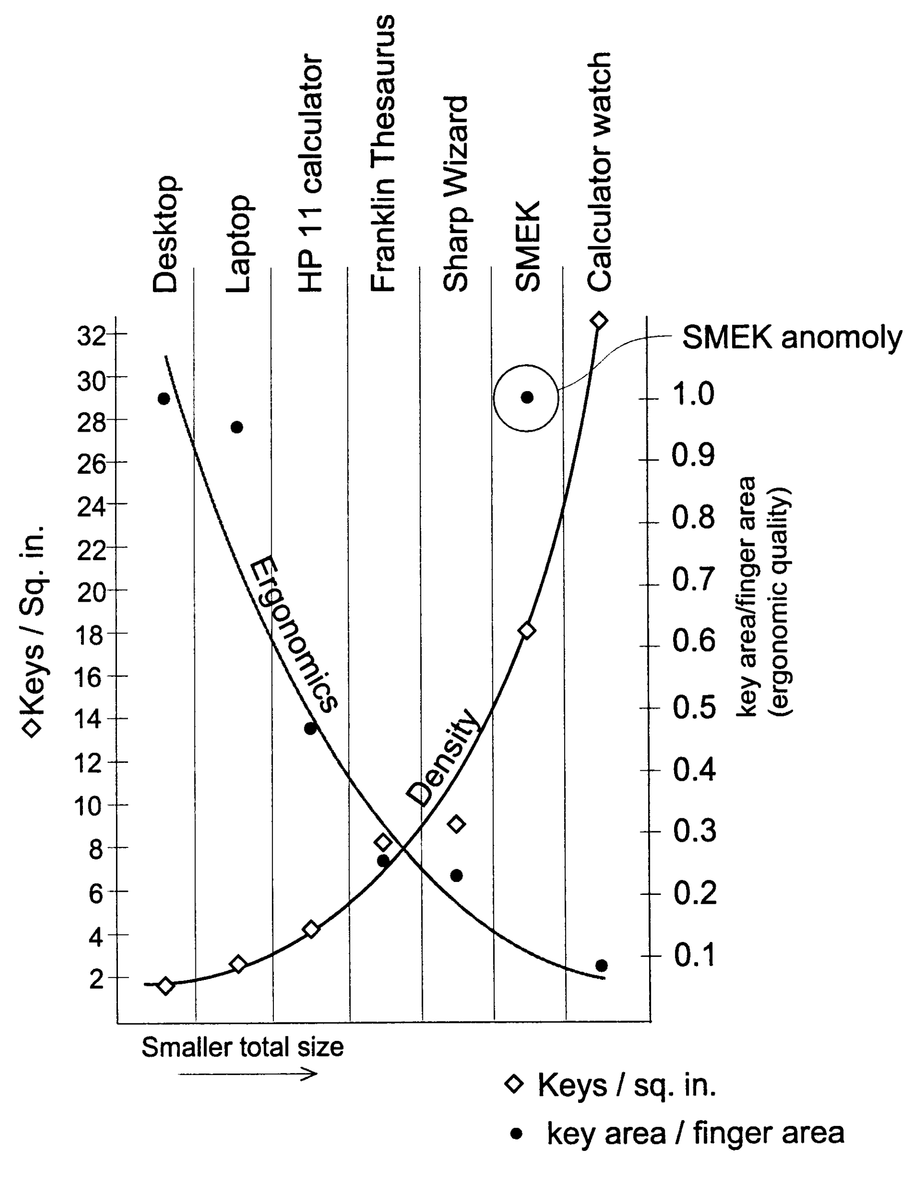



**Figure 35 - Density and Ergonomic Quality vs Overall Size**

Figure 36 provides a detailed comparison with the keypad of the *Pagewriter 2000*. Note: between the *250* and *2000* interface, the *2000* is the better of the two.

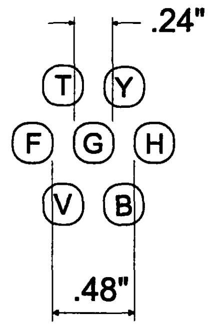
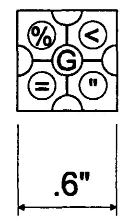
| | Pagewriter | SMEK | Improvement factor |
|---|---|---|---|
| Key Density | 8 keys/sq. inch | 18 keys/sq. inch | **2.25x** |
| **Ergonomics** |  |  | |
| Space between keys | .48" | .6" | **1.25x** |
| Key width | .24" | .6" | **2.5x** |
| Key contact area | .057 in$^2$ | .36 in$^2$ | **6.3x** |





**Figure 36 - Pagewriter vs. SMEK**

Rather than making engineering tradeoffs between parameters, the SMEK offers more than twice the density while simultaneously increasing the contact area, key width and space between adjacent keys.

5.1.7 SMEK Applications

Applications for the SMEK include:

#### 5.1.7.1 *Wearable Computers*

Wearable computing is currently an obscure technology, used primarily by researchers. Data is typically entered into the computer with chorded keyboards, such as the Data Egg, or the Twiddler. However, it is assumed chorded keyboards will meet resistance as wearable computing enters the mainstream because they are relatively difficult to use and take significant time to learn. An obvious solution is to use a small traditional keyboard, perhaps worn on one wrist. The industry leaders in wearable computing (Rockwell, Xybernaut and ViA) are all interested in the SMEK for this purpose.

Note: There is a problem with using any keyboard in any augmented reality application. See Section 5.2.

#### 5.1.7.2 *Two-Way Paging*

With key densities at 8 keys/square inch or higher, existing two-way pagers provide a badly compromised interface. This is why the companies in the industry (Motorola, Wireless Access/Skytel, RIM) are interested in the SMEK.

#### 5.1.7.3 *Internet Telephone*

Several companies are working to provide Internet access through telephone, both desktop and cellular. Providing Internet capability on a cordless phone or cellular device (while maintaining an acceptable level of ergonomic functionality) is currently impossible. The value of the cellular market was $1.275 billion dollars in 1994[34], although any estimates for the value of the Internet

---

[34] 1995 Electronics Market Data Book, Electronic Industries Association

cellular market would be purely speculative. The industry leaders in Internet telephone (Motorola, Qualcomm, Nokia, Ericsson) are all interested in the SMEK for this purpose.

Internet telephone on the desktop is less of a problem because desk space constraints are relatively accommodating. However, cordless phones provide a similar use pattern as cellular phones. Cordless phones outsell corded phones in the home market by nearly 2: 1, selling 16.7 million units worth $1.1 billion in 1994[35]. The depth of market penetration

#### 5.1.7.4 *Personal Digital Assistants/ Handheld Computers*

Until VR is available, the options for no-compromise input is limited to handwriting recognition and keyboard input. Although over 90% of PDAs and handheld computers sold are keyboard-based[36], there are already highly successful HR products at the high-end of the industry. However, for the next decade, HR will remain significantly more expensive than keyboard devices. A combination of cost differentiation and personal preference creates a demand for miniaturized ergonomic keyboards.

#### 5.1.7.5 *Standard Telephone*

The number of area codes is increasing quickly as the system expands to accommodate the cellular phones, fax machines and data lines that belong to a growing populace of increasingly technologically sophisticated individuals. Current industry prediction is that the existing 10 digit telephone numbering

---

[35] Ibid.

[36] Dataquest April 1997

scheme will reach capacity in 20-25 years[37]. The plan is to add three additional digits at that time. However, other sources indicate that (for ergonomic reasons) a 10 digit LOCAL telephone number is simply too many digits[38]. An alternative being considered for a long-term solution is the implementation of a ten digit alphanumeric phone number, offering 1,400 times the capacity of the existing system. Again, the size of the interface is a critical issue.

### *5.1.7.6 Interactive Television*

As the distinctions between television and computer blur, so will the distinction between their respective input devices. Television input is typically provided by a hand held device, operated remotely with an array of buttons. Computer input requires an extensive keyboard and cursor control. Ultimately these functionalities will need to merge into a comprehensive device that suits both tasks. There is currently no option to provide desktop functionality into a hand held device. Sony and Philips are known to be actively pursuing a solution to this problem.

### *5.1.7.7 Miscellaneous Applications*

There are many miscellaneous applications for a reduced size alphanumeric keyboard. Examples derived from active market interest are:

---

[37] Jim Deak, Manager, North American Numbering Plan Administration. In interview 4/28/97

[38] Greg Blonder, Director, Customer Expectations Research, AT&T Labs. In interview 2/26/96.

- Dentistry - Modern dental practice currently includes a computer located at the dental chair as a means to access dental records quickly and efficiently. However, space is at a premium and hygienics are critical. The Seltzer Institute of Dental Technology is seeking a small, portable AN keyboard, that can be easily washed.

- Avionics and aerospace - There are several applications for an AN input device in aircraft, yet space is a premium. Rockwell has expressed interest in the SMEK for this purpose.

- Smart card readers - The advent of readable and writable smart cards creates the need for a small, inexpensive means to interface to them. Motorola is interested in the SMEK for this purpose.

- Security - The addition of alphabetic characters on a keypad combination lock increases the number of combinations while allowing the user greater flexibility in selecting a code he or she will find easy to remember.

- Obviously, electronic dictionaries, language translators and the like require extremely low cost, miniaturized input technology.

### 5.1.8 SMEK Development

New input devices are notoriously difficult to implement, largely because of the ambiguity of getting the "feel" correct. Additionally, it is well known that providing "user friendliness" significantly increases the burden on the product designer. The SMEK is a chorded keyboard that offers the user-friendliness of chording automatically, a feature called Passive Chording. Providing passive chording in a well-functioning unit entails several obstacles, especially electronic, that are not readily apparent.

#### *5.1.8.1 Legend printing*

One of the first issues to address in SMEK development was that the combination keys require a legend to identify each intersection of four keys. Various possibilities were considered:

- Locating these interstice legends close to, yet not on, the interstices seemed an unacceptable solution, primarily for aesthetic reasons.

- Printing legend characters directly on the interstices of independently movable keys presented a tolerance problem. The tolerance stack up for printing, molding, and of the normal variabilities required to allow smooth operation indicated that the accumulated error would be greater than the width the legend traces.

- Molding the entire surface from a single elastomeric sheet, as shown in Figure 37 was ultimately chosen.

This solution eliminates the physical nature of the interstice, but retains its functionality. The solution offers additional advantages, namely low cost, extreme thinness and the opportunity for a sealed unit. An interstitial pad slightly larger than the interstitial legend provides a flat surface and locally eliminates the textural gaps between adjacent key caps, thereby easing printing requirements.

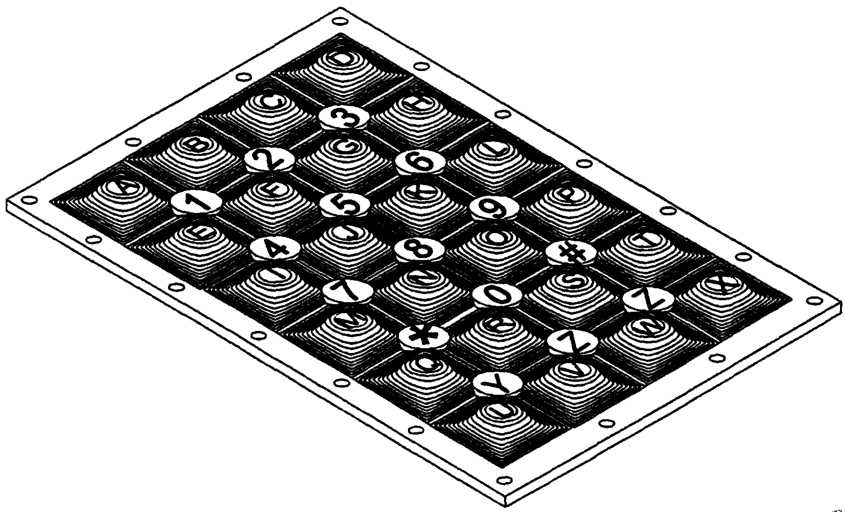


**Figure 37 - Elastomer SMEK**

Figure 38 shows a sectional view of keypad implemented with a continuous elastomer. On the right side of the drawing, the elastomer material continues beyond the contact surface, under the constraint flange to provide mechanical constraint. This addresses a problem found in the first design, which experienced a slight billowing in the center when elastomer edges were held by friction.

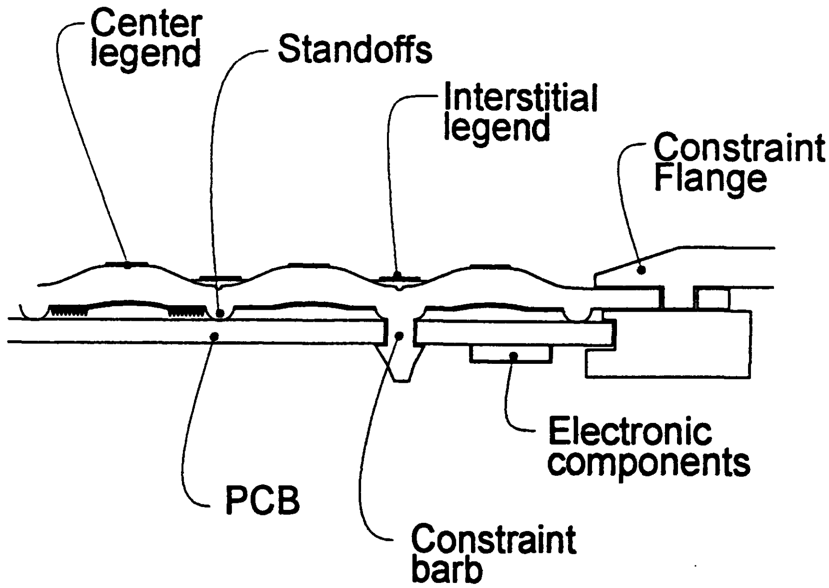



**Figure 38 - Sectional View of Keypad**

The difference in height between the interstitial legend and the center legend is approximately 1 mm. Though small, experiments indicate that this difference in height is enough to enable operation of key switches that would otherwise be difficult to operate independently. The surface undulation provides tactile and visible identifiers to distinguish the elevated keys from the non-elevated keys. Note that the overall height from the center legend to the textural gaps may exceed the difference in height between the interstitial legend and the center legend, thereby creating greater visual differentiation than the tactile difference between the height of the interstitial legend and the center legend. Unused printed circuit board area maybe used for unrelated product components, for complete product integration.

#### 5.1.8.2 *Feedback*

While some type of feedback option[39] is necessary, it was determined early in development that providing feedback with each key was problematic because slight variations in feedback timing would result in inconsistent feedback. Sometimes a single "click" would be heard, sometimes a succession of clicks, as each feedback mechanism was actuated at slightly different points in the stroke. The solution was to use a speaker to provide an audio and tactile "click" for each key actuation. Because the area is so small, it is impossible for the user to distinguish the feedback is not localized to the finger.

#### 5.1.8.3 *Switch Options*

While the patent covers any switch technology, the decision to use a continuous elastomeric sheet as the contact surface suggests mimicking a standard membrane keyboard[40]. The separation layer is replaced by small standoffs molded into the elastomeric sheet, reducing the assembly by one component, thereby saving material and a manufacturing step. The conductive pads may be implemented with one of two methods. Figure 39 shows carbon-doped rubber pucks that are insert-molded during the elastomer manufacture.

---

[39] There is a trend to eliminate audio feedback from many portable devices because they are often used in situations in which any noise is undesireable.

[40] Note: In membrane keyboards, the legends are printed on a deformable memory with discrete conductive pads located under each "key." The scan matrix is printed on either a flexible or rigid substrate. The legend membrane is held from contact with the scan matrix by a separation layer, sandwiched between. The separation layer is perforated by holes located under each key, allowing the conductive pads to contact the scan matrix when the user presses on the switch.

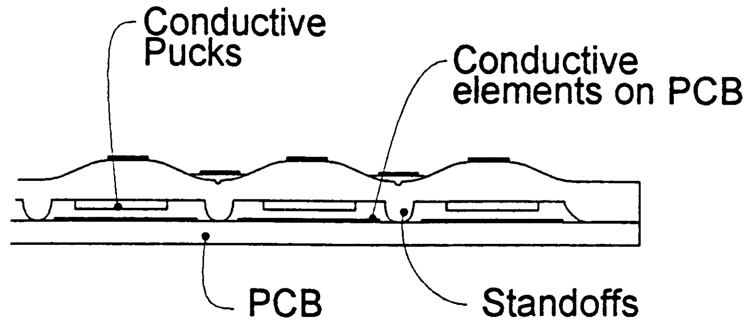



**Figure 39 - Conductive Puck Switch Technology**

Figure 40 shows the conductive pads printed with conductive inks in a second manufacturing step. The figure also demonstrates that a variety of shapes are available to improve reliability.

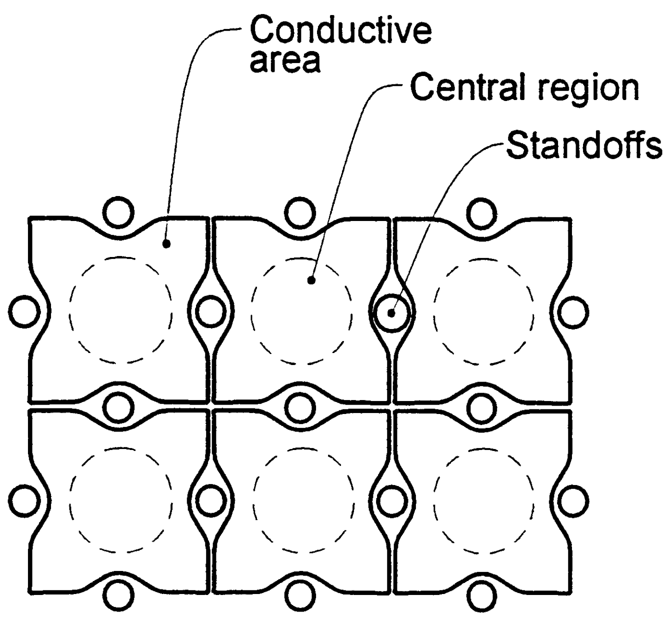



**Figure 40 - Printed Conductive Regions**

The floral-shape demonstrates the conductive region at its maximum extent. This shape increases reliability by maximizing contact at the interstice. While both pucks and printing may offer variable puck shape, stamped pucks introduce the issue of orienting each puck, increasing manufacturing setup costs and reducing manufacturing reliability. Non-round conductive pucks would either stripping a pill carrier, or orientation of individual pills. Therefore, high-volume manufacturing with non-round conductive regions suggests the use of conductive inks. However, conductive inks have reduced adhesion to lower durometer materials. A final decision between ink or puck is a function of elastomer durometer and the shape of the conductive region[41].

Figure 40 also shows the central region recessed slightly to increase the robustness of electrical contact at the corners of each individual key. Recessing the central region increases reliability by reducing the net force required to activate at least two diagonally opposed associated individual key caps. The reason is as follows: If the user is slightly off-center of the interstice, the key(s) toward the off-center direction to strike the printed circuit board earlier during the stroke. Recessing the region in the centers of individual key caps, causes the material located in the direction of the offset to present less resistance to the finger, allowing the elastomer material underneath the finger to deform more readily for a given force, thereby reducing the overall force required to actuate all keys of the interstice.

[41] Shinetus, a leading keyboard manufacturere, has provided a high-volume production quote for the carbon-doped puck implementation of $0.60 each.

### *5.1.8.4 Prototype Switches*

The prototypes use conductive .21" x .21" conductive square pills contacting a similarly sized "E" type switch grid, as shown in Figure 41.

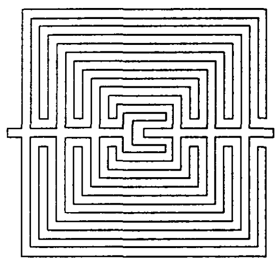


**Figure 41 - Switch Grid**

These pucks were molded into continuous silicone sheets, durometer 40 on the shore A scale. Traces are .010" wide with gold flash with .010" spaces. Gold flash is crucial to eliminate oxide generation in an non-wiped switch contact. Overall height of traces, trace width and gap width are critical because they determine the force/resistance curve for each switch. These are the mechanical parameters that must be optimized for optimal switch functionality.

*5.1.8.5 Switch Matrix*

The prototype switch matrix consists of four rows and ten columns etched on a printed circuit board, as shown in Figure 42. Each square represents one switch grid

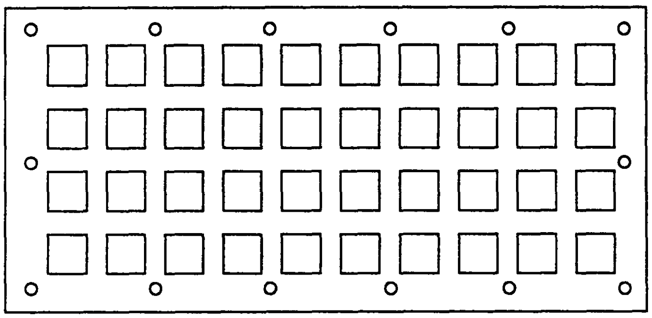


**Figure 42 - Switch Matrix**

*5.1.8.6 Debounce*

Tests indicated a worst-case settling time of 12 milliseconds. Figure 43 shows a typical trace of the rise time as a conductive puck contacts the traces on the printed circuit board. The time base is 1ms/division.

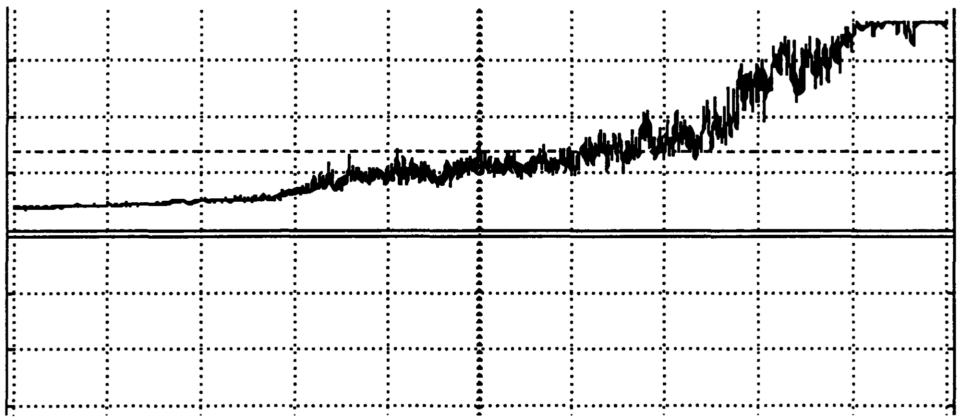


**Figure 43 - Typical "Make" Trace**

Figure 44 shows a typical settling time as the switch contact is broken.

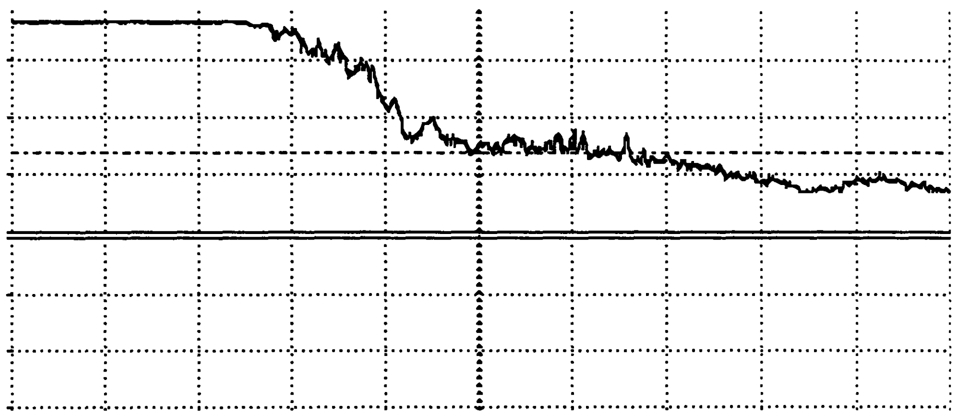


**Figure 44 - Typical "Break" Trace**

5.1.9 Scanning Algorithm

The scanning algorithm of a standard keyboard matrix is shown in Figure 45. The matrix is scanned, an intersection determined. The intersection is looked up in a table and the character is sent.

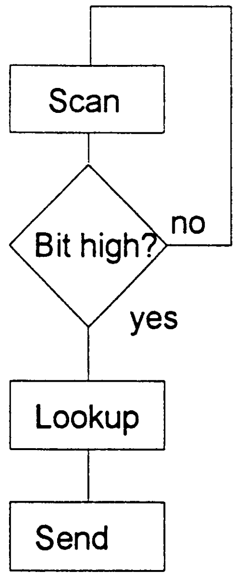



**Figure 45 - Standard Scanning Algorithm**

Referring to Figure 46, the SMEK starts by performing the standard algorithm and then stores this information. The software then scans local rows, searching for additional key switch contacts adjacent to the initially found key. If it locates a second key before a time-out occurs (See Figure 47) then stores the second key location and performs a lookup with both pieces of data.

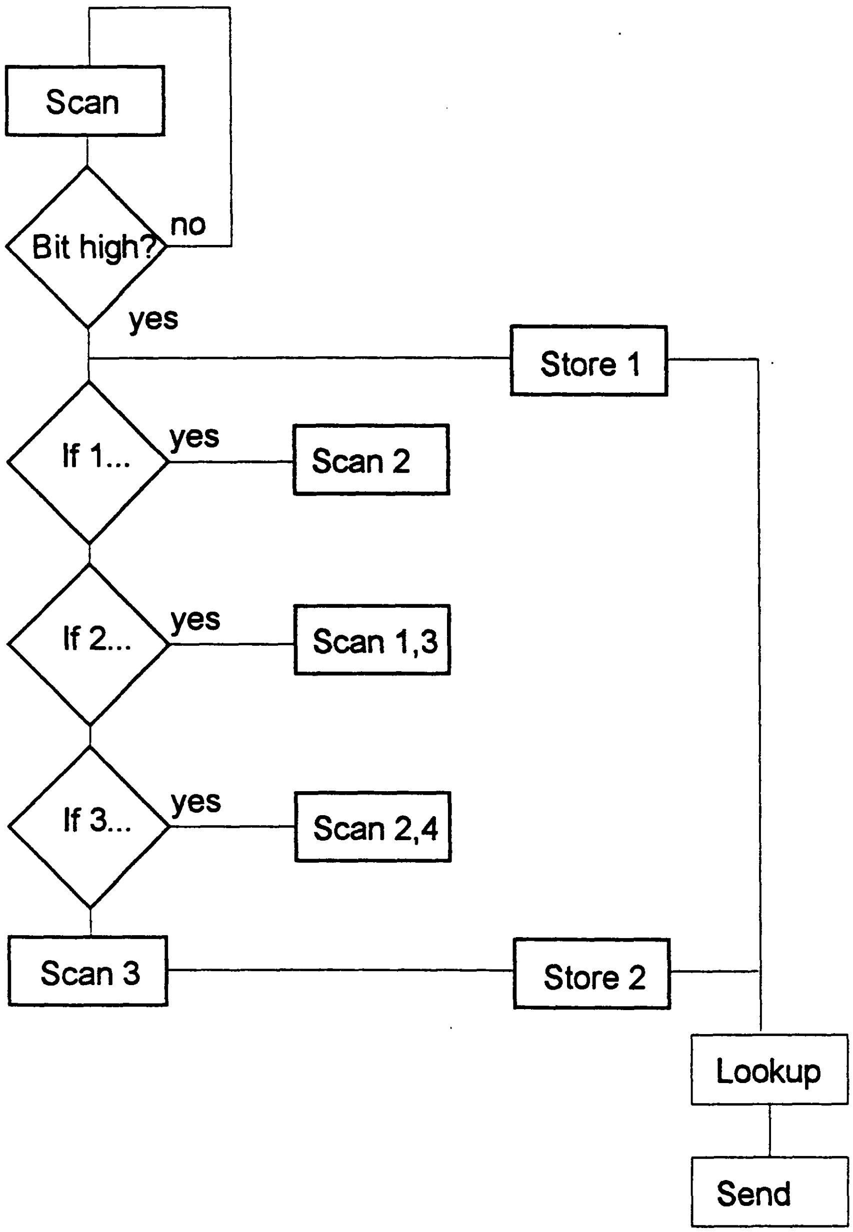



**Figure 46 - SMEK Scanning Algorithm**

5.1.10 Timing

The timing issues are delicate to ensure reliable AND fast actuation of both individual and combination keys. Figure 47 shows the timing strategy. The details of implementation are rather extensive. The code is provided in the Appendix.

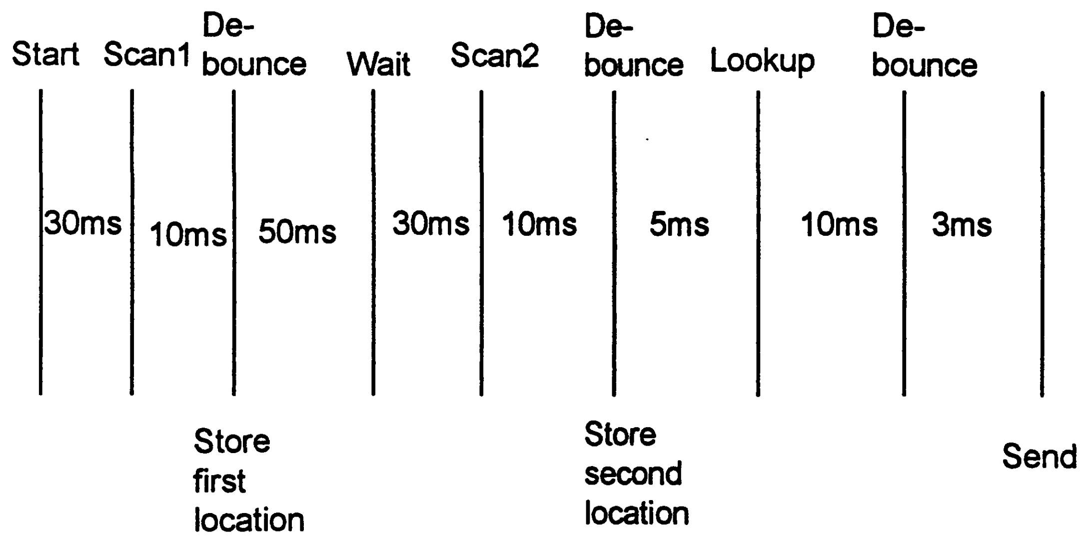



**Figure 47 - Timing Strategy**

5.1.11 Electronics

There are several electrical implementations that allow a plurality of switches to be sensed simultaneously.

### *5.1.11.1 Diode Matrix*

Figure 48 shows a diode matrix of a 5 x 5 matrix offering full decoding to the keypad. The drivers inject a signal which presents a unique output composite to the receivers as shown in

Figure 49. In this configuration, the relative physical position of the switches are independent of the actual matrix connection (i.e. a four switch combination can be actually implemented using any four of the 25 switches shown.) This statement is not true for Figure 50 where the keys must be arranged

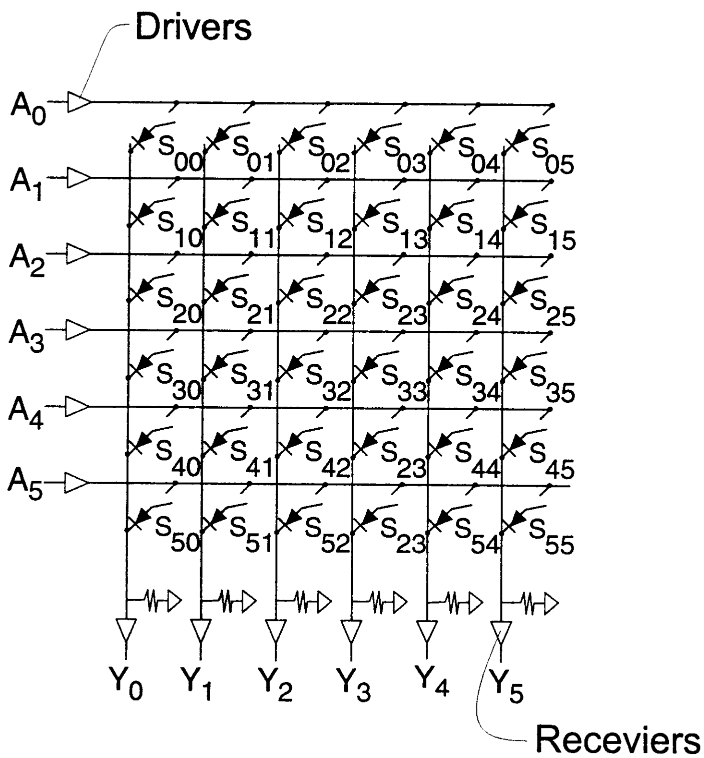



**Figure 48 - Diode Matrix**

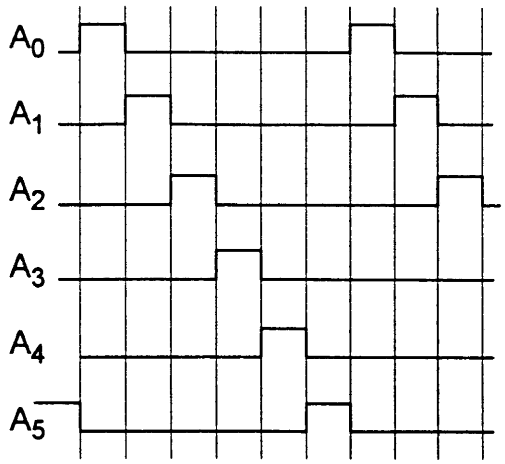



**Figure 49 - Timing Of Diode matrix**

### *5.1.11.2 Tristate Driver*

Figure 50 shows an open-drain or tristate driver and a receiver solution, which does not offer full decoding, but is nonetheless adequate to the application. A "1" on Ai enables the tristate driver which drives a "0" on the horizontal line. If switch Sij is closed then a low level is sensed on output Yj when input Aj is active.

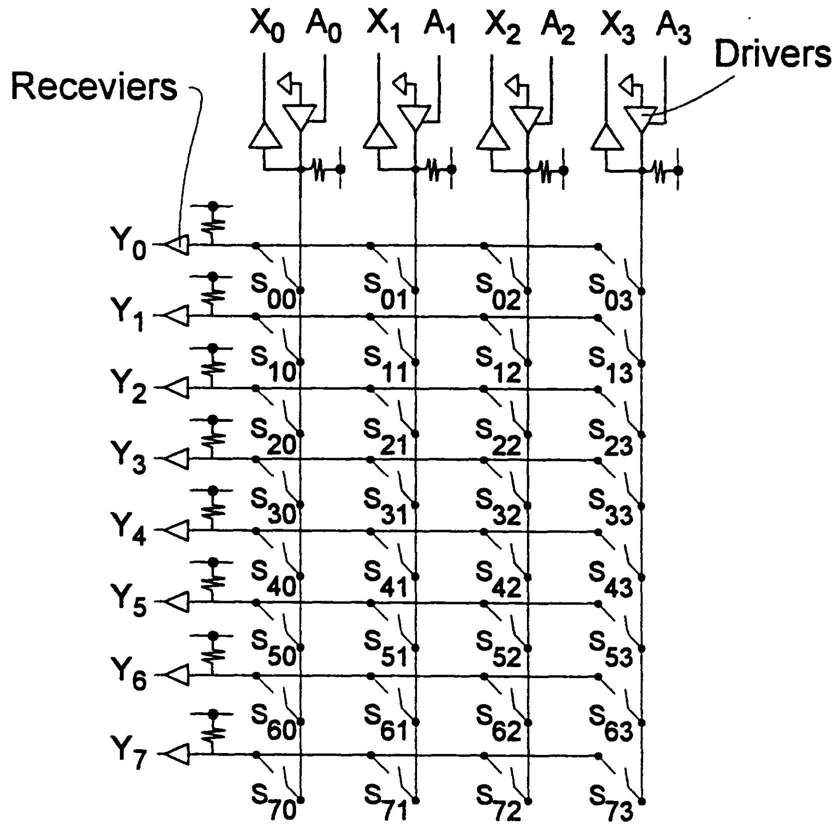



**Figure 50 - Tristate Driver Solution**

Sensing algorithms are as follows:

#### 5.1.11.2.1 Single Keys:

If switch Si,j is closed a low signal level is sensed on output Yz when input Ai is active.

#### 5.1.11.2.2 Combination keys

- If switches: Si,j; Si,j+1; and (Si+1,j OR Si+1,j+1) are closed a low signal level is sensed on the outputs: Yi; Yi+1; and Xj+1 when input Aj is active and on the outputs Yi; Yi+1 and Xj when input Aj+1 is active

- If switches: Si,j; Si+1,j+1; and (Si,j+1 OR Si+1,j) are closed a low signal level is sensed on the outputs: Yi, Yi+1 and Xj+1 when input Aj is active.

- If switches: Si,j; Si+1,j+1; Si,j+1 and Si+1,j) are closed a low signal level is sensed on the outputs: Yj; Yj+1; and Xi+1 when input Aj is active, and on the outputs Yi; Yi+1 and Xj when input Aj+1 is active.

- If switches Sij and Si,j+1 are closed, a low signal level is sensed on Yi and Xj+1 when Aj is active and low level is sensed on Yi and Xj when Aj+1 is active.

- If switches Sij and Si+1 are closed, a low signal level is sensed on Yi and Yi+1 when Aj is active.

  If switches Sij and Si+1,j+1 are closed, a low signal level is sensed on Yi when Aj is active and low level on Yi+1 when Aj+1 is active.

#### *5.1.11.3 Tristate Driver Timing*

Figure 51 shows the timing diagram for the tristate driver design. Inputs can be erroneously read if the matrix is scanned too slowly and without a validation constant. Scanning must be faster than human response time, but without wasting power. A scan rate of approximately .030 sec per matrix is adequate. Validation time constant is the time (in scanning cycles) that a key or key combination must be maintained to be recognized as a valid input. This value can be user selected. Variously, the validation time constant may be adaptable such that a large number of repeated correction strokes will result in the validation constant being increased automatically. Likewise, an adaptable validation constant may be lowered automatically to a threshold level if the user does not make a mistake for an extended period. It is also possible to monitor forbidden keystroke combinations (potentially two-stroke combinations) and automatically adjust the validation constant to a value longer than the three-sigma distribution of this duration.

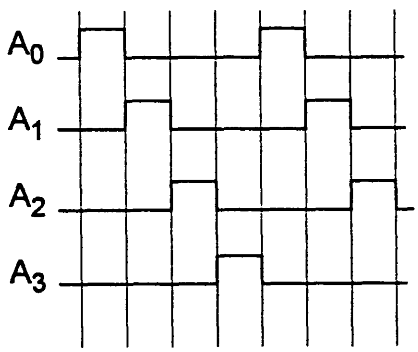



**Figure 51 - Timing Diagram**

### *5.1.11.4 Direct Solution*

Figure 52 shows another circuit that may be used to scan the matrix. The microprocessor outputs a row "number", 0-3 which goes to the 2 to 4 line decoder. The 2 to 4 line decoder drives the corresponding output low, which turns on the transistor, which drives the entire row low. The microprocessor then reads the columns to see if any of the columns are low, which would indicate a pressed key cap. (Pull-up resistors on the columns dictate columns will read high if no keys are pressed.) The microprocessor is constantly driving the rows and reading the columns. If a key is down for three consecutive scans, it is considered valid. After this is detected, the microprocessor then looks to see if any of the adjacent keys are activated to decide whether it is single key or not. If the appropriate adjacent keys are activated, the "chorded" (combination) key is sent, otherwise the single key is sent. There is an additional 2 to 4 line decoder not shown.

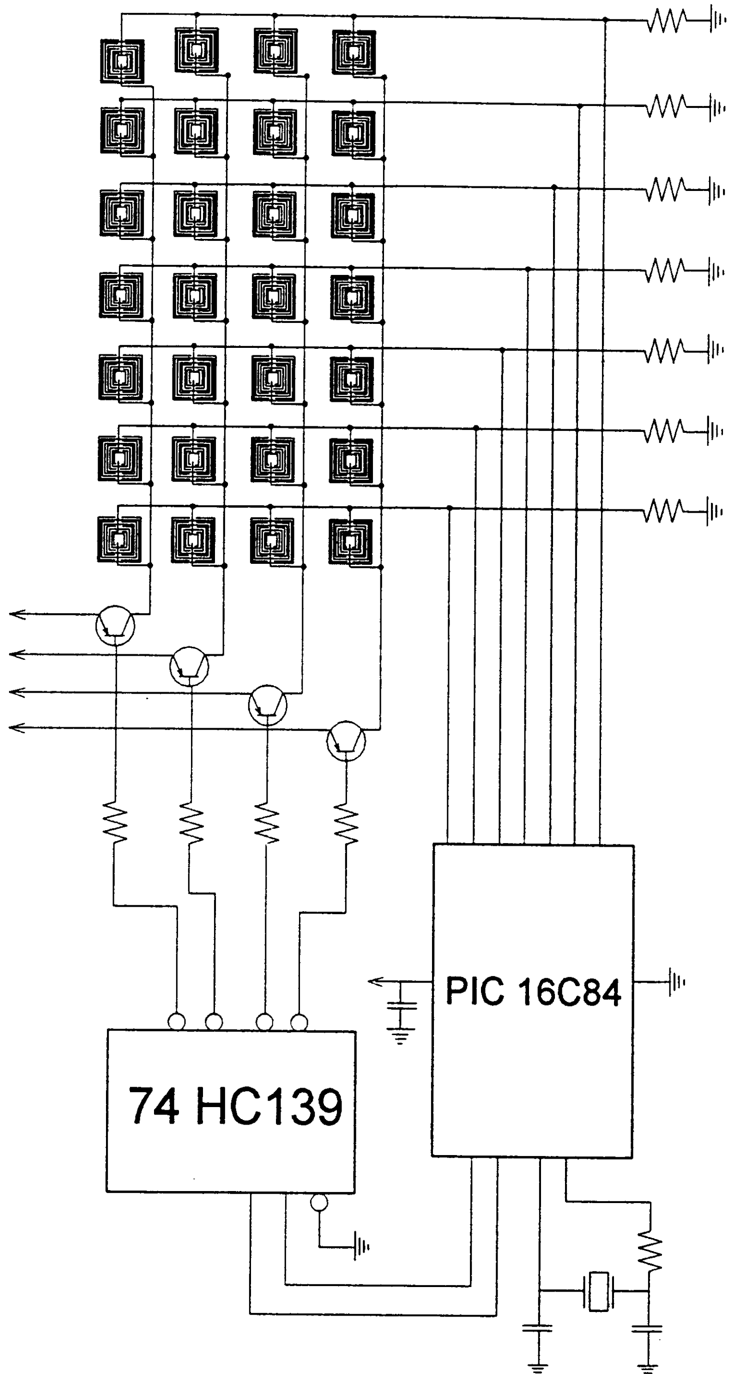



Figure 52 - Circuitry for Direct Scanning

### 5.1.12 Prototype Development

#### *5.1.12.1 Integral Pointing Device*

Figure 54 shows the preferred embodiment of an isometric view of a SMEK with an integral force-sensitive pointing device located at elevated key between the R and S keys. The pointing device is integrally molded within the elastomer to disallow liquids or particulates from lodging between it and the keypad components. While a gasket may be used, complete integration saves material and assembly cost while presenting a cleaner appearance and provides a more robust liquid barrier.

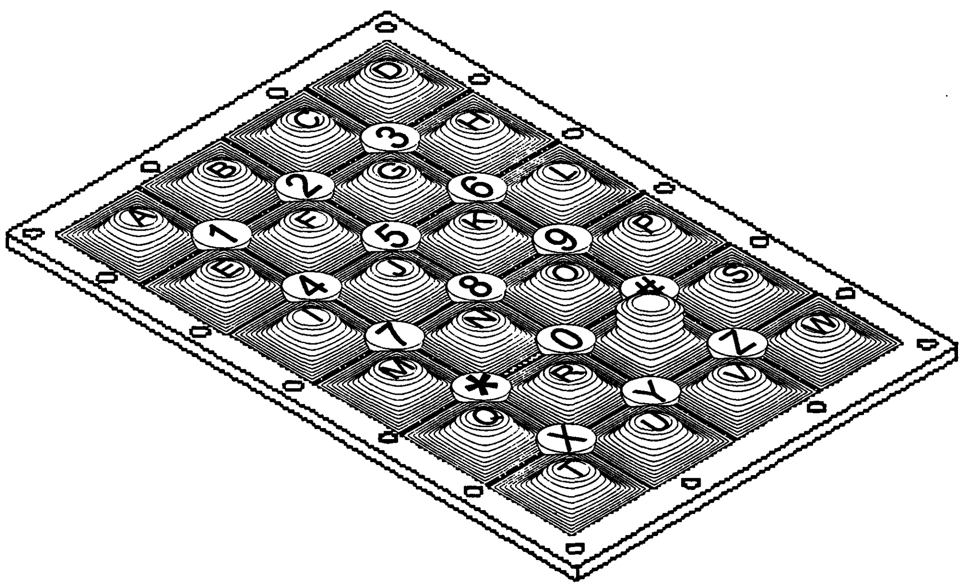


**Figure 53 - SMEK with Pointing Device**

Figure 54 shows a cross sectional view of the elastomeric sheet with a pointing device. A force transmitting member, transmits force from the cursor control point to the force sensor, shown mounted below the printed circuit board, allowing the force sensor to be internal to the product, minimizing surface irregularities.

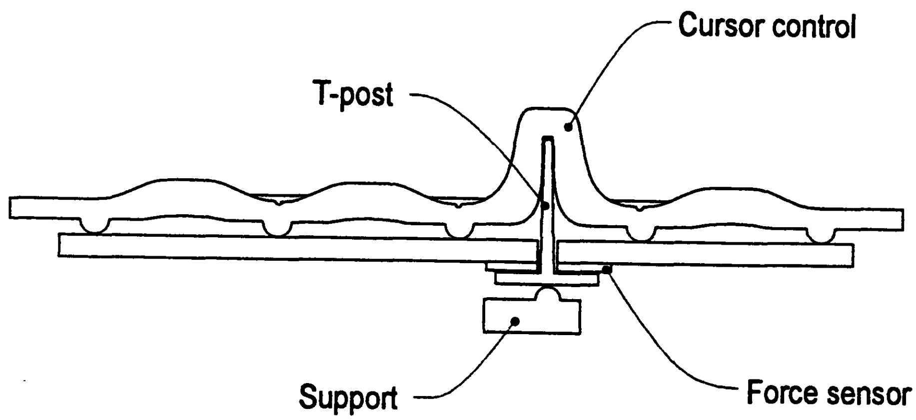



**Figure 54 - SMEK with Integral Pointing Device**

5.1.13 Extensions of SMEK concept

Additional geometries were included in the patent such as a triangular based unit as shown in Figure 55.

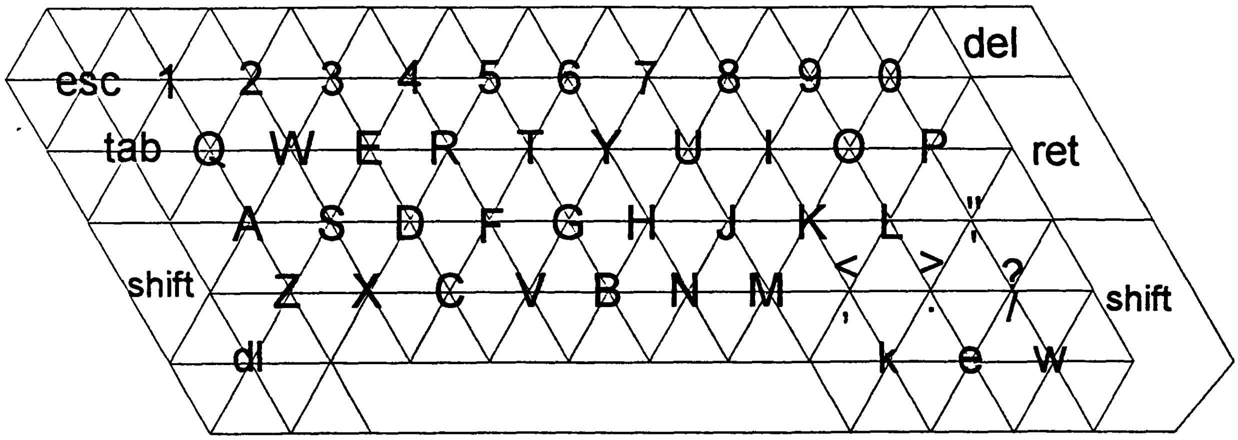



**Figure 55 - QWERTY based on Hexagonal Key**

## 5.2 Augmented Reality Keyboard

Allows traditional keyboards to be used in a wearable computing (augmented reality) environment.

### 5.2.1 Background

Wearable computing is currently an obscure technology used primarily by researchers. In this rarefied environment, the input device of choice are chorded keyboards. It is assumed that as wearable computing enters the mainstream there will be resistance to chorded keyboards as they take significant time to learn and require constant use to remain proficient. An obvious alternative is to use a small traditional keyboard (or a SMEK), perhaps worn on one wrist. However, traditional keyboards present a serious problem with augmented reality applications because the user must shift gaze from the display located on one's eye, to the keyboard at one's hand on a near-constant basis. This is an annoying requirement that results in frustration and optical fatigue.

5.2.2 Overview

The AR keyboard solves this problem by adding an at-a-distance sensor capable of imaging a human finger beneath the keyboard. The major dimensions of the sensor must be approximately equal to those of the overall keyboard. The sensor detects, minimally, the X and Y motion of a finger and reports this XY data to the head-mounted display where it is used to drive a cursor that mimics finger motion. Figure 56 shows a finger hovering above the "X" key of a keyboard located on a wrist or in a pocket.

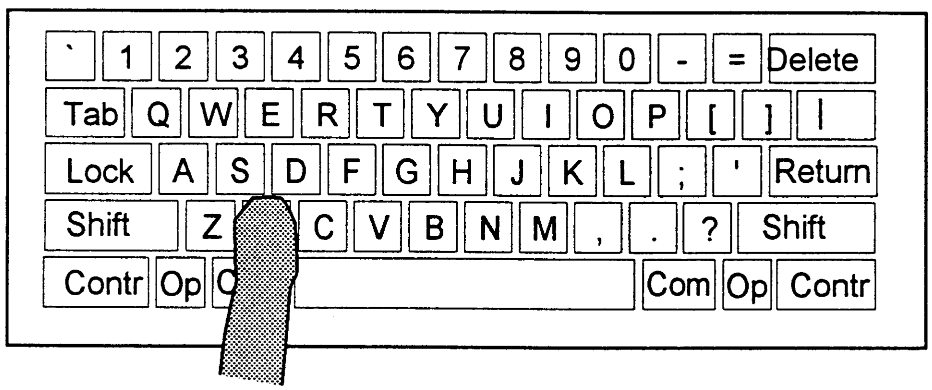



**Figure 56 - Finger hovering over keyboard**

Figure 57 shows the view within the head-mounted display. In this way the user receives feedback as to the location of her finger without being forced to look away from the screen image and to change focus.

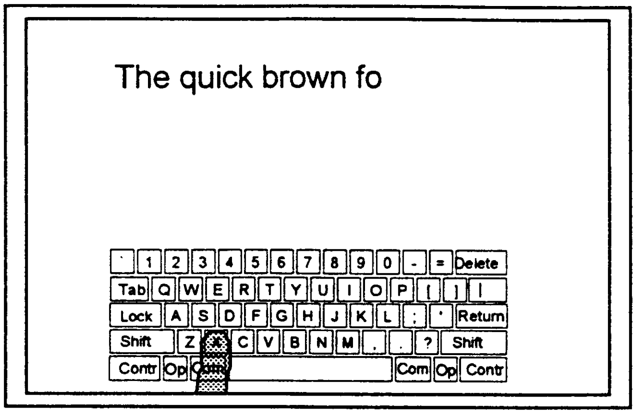



**Figure 57 - Finger Position Reported In Augmented Reality**

### 5.2.3 Determining Input

Key selection is determined in software, based on the image provided to the user at time of finger contact. Therefore, the keyboard itself does not need to differentiate between "keys."

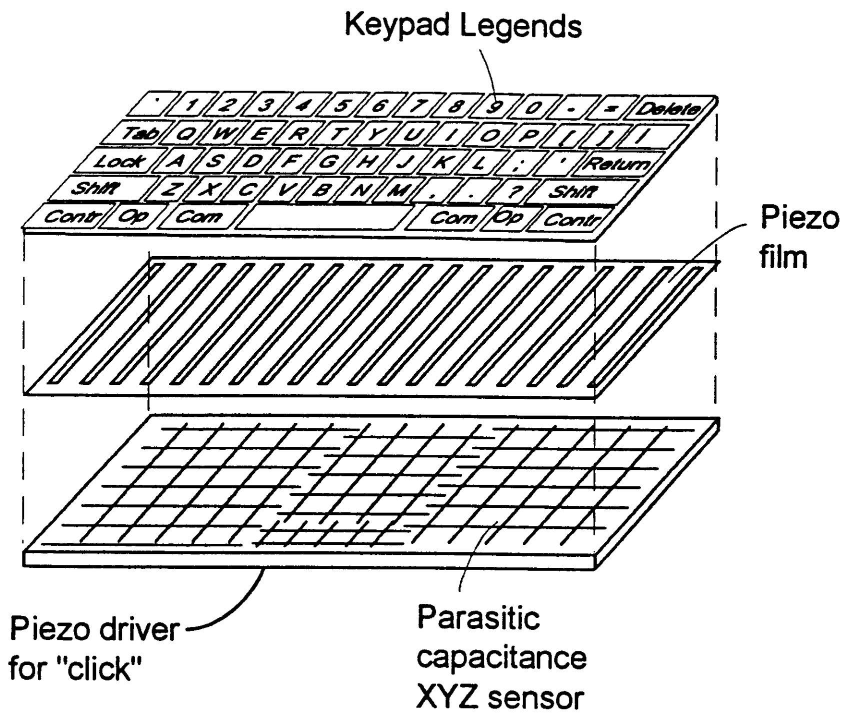



**Figure 58 - Exploded View of AR keyboard.**

In fact, to minimize cost, the entire keyboard is comprised of a single keyswitch switch over the entire keyboard area, as shown Figure 58. Because the resolution of at-a-distance sensors is not sufficient to mimic key switch operation, the wide-area key switch indicates whether or not the user is intending to actuate a key with a high degree of accuracy. Determination of *which* key the user intends is provided primarily by XY data and secondarily by Z data as provided by the at-a-distance sensor. A reliable, thin, low-cost area sensor area switch may be implemented with a piezo film, such as sold by Amp. XY data may be provided by a number of commercially available means as well as EF tomography. The sensor plane extends approximately to the full extent of keyboard graphic. One problem with capacitive sensing techniques is the limitation of sensing only one digit at a time. This is unacceptable for a keypad application. Therefore a plurality of sense regions are used. Figure 58 shows four regions of capacitive sensor grids to allow tracking more than one finger at a time. Tests indicate four independent grids (one for just the space bar) are sufficient to track two fingers in the keyboards under 5" in width, although it may be simpler to implement a regular grid of overlapping EF tomographic sense grids. This embodiment would provides for the independent tracking of any number of fingers, providing a single "contact" output whenever the substrate is pressed at any location.

It is worth noting that because the user will probably never look at the keyboard, it needn't have a graphic. It can be located in pocket, or manufactured of conductive fabrics and pinned to a shirt with any graphic whatsoever. As the user moves the finger over the keyboard surface, the active letter (at the cursor position) changes accordingly. When the correct letter appears, the user depresses the key. Through repetition, the user learns to touch type, with the added benefit of constant feedback.

### 5.2.4 Textural Feedback

In the preferred design, the keyboard has a textured surface, such as provided by the SMEK surface. This surface provides the user feedback to help keep her oriented and to provide a path for kinesthetic learning.

### 5.2.5 Visual Feedback

Also in the spirit of duplicating reality, each "cursor" is finger shaped, and displayed in an angular rotation that corresponds with the angular rotation of the associated finger.

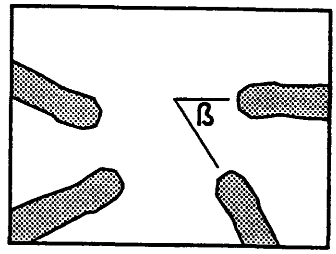



**Figure 59 - Variable Finger Orientation**

The active point of the finger-shaped cursor is located near the tip, mimicking a real finger. The area of the finger-shaped cursor is semi transparent, allowing the user to read graphic legend beneath. These features offer a greater degree of physical presence in a virtual world and provide a higher quality interface. The cursor is instantly locatable and provides the user a clear mental mapping between the real and virtual worlds, although precise identification of the finger location is not necessary because "true" finger location is determined within the virtual world.

### 5.2.6 Modal solution

The device may also be used as a cursor control device outside of the keyboard context. The preferred means to toggle back and forth between the two contexts is for the user to use fingers individually in the keyboard context and to place the index and middle fingers together as indication that a "mouse" context is desired, as shown in Figure 60. The image perceived by the sensor is shown in gray over the hand illustrations.

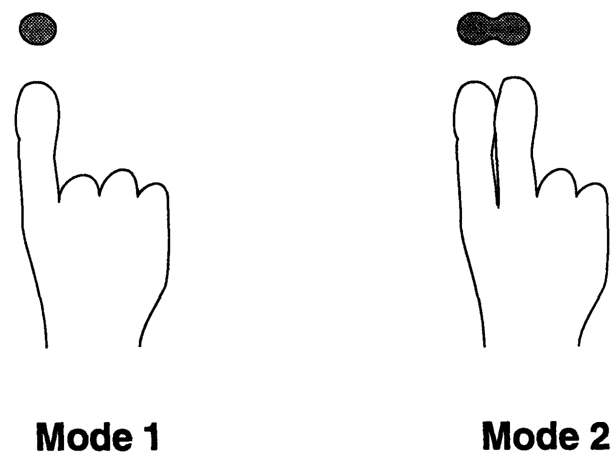



**Figure 60 - Toggle Method**

Whenever the sensor detected Mode 1 operation, the system would assume the user was intending keyboard information. Whenever the sensor detected Mode 2 operation, the system would assume the user was intending mouse operation. The device is sensitive to the change in the aspect ratio of the input digit and alters system performance accordingly.

5.2.7 Body LAN Keyboard

There are several means to implement wireless communication between a wearable keyboard and central processor. Modulating the electric field of the body (body LAN) is preferred, as it reduces cost through the elimination of a dedicated keyboard receiver. Within this category are several techniques. Transmitted data may consist of absolute or relative XY location data, or specific key location as determined by known sensing technologies.

A novel sensing technology is to use the body LAN as a finger location identification means. This is accomplished by providing a number of discrete transmitter points (electrodes) at or below the contact surface of an intended virtual keypad. Each transmitter point to provide a distinct signal. The body LAN can detect finger position by measuring the strength of signal with respect to each transmitter point.

## 5.3 Miniaturized Augmented Reality Keyboard - MARK

Reduces size of a SMEK AR keyboard by another order of magnitude.

### 5.3.1 Concept

Because user perceives the AR keyboard only in virtual space, because all motions may be relative to any given key within that virtual space and because accuracy and repeatedly is NOT required (only linearity) it is possible to implement a truly virtual keyboard by using the concept of the AR keyboard with a 3D device. A purely virtual means to provide keyboard input to a head-mounted display without forcing the user to shift gaze is as follows: Track finger motion in space, having displacement move "cursor" on head-mounted display. Use a real switch at tip of finger to indicate intended "key" actuation and provide a transparent means for changing between keyboard and mouse functionality.

### 5.3.2 Implementation

The basic technique is to build and collapse a driven magnetic field close to a tuned RC circuit. The field charges the circuit. The driven field is turned off and the passive RC circuit drains, building its own magnetic field. Measurement of this decay provides the information to calculate the distance between source and circuit. The technique is known in the art, although primarily used for data transfer, predominantly within anti-theft technologies. However, over extremely limited ranges the technology may provide distance information. The operation is repeated in all three axes. By modulating amplitude and frequency, there is sufficient data to calculate all three angles of rotation.

For this application the transmitter is built into a watch, or other wrist-worn device, and the transponder a thimble, as shown in Figure 61. Because the thimble electronics are so simple, they can be tiny. Because they are completely passive, no battery is required.

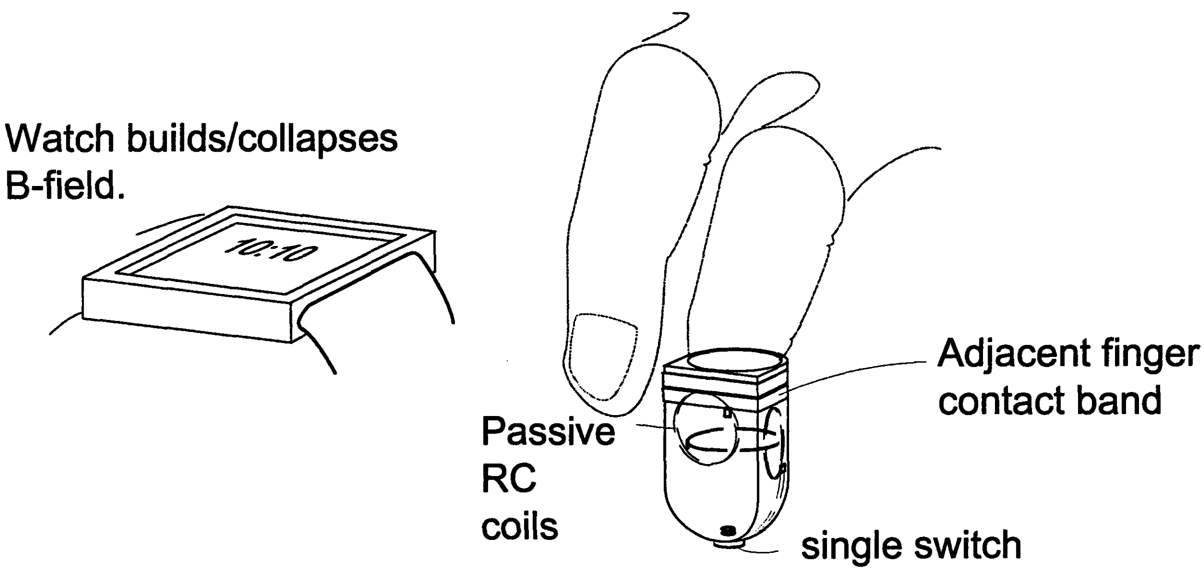



**Figure 61 - MARK Concept**

5.3.3 Switch Actuation

A single switch at the base of the thimble indicates when the user intends to select a key. The user can therefore "type" on any surface. The switch breaks the RC circuit, providing a discontinuity the circuitry interprets as a key press. (this renders the circuitry inoperative until the switch is released, but, by definition, the circuit is not needed whenever the user is key is actuating a key. Each key press also resets the "zero" position of the device, so all motion is measured relative to the last letter entered. Because the only reference is in virtual reality, true displacement is unimportant. The inherent non-linearities of the magnetic fields are remapped by software into approximate linearity, which is more than adequate for the application, because feedback is constant: It never matters where the finger actually is, only where the user perceives it to be.

### 5.3.4 Electronics

Three coils are tuned to distinct frequencies using standard LRC techniques. One axis of the system is outlined in Figure 62. The transmitter builds a magnetic field along one axis, which charges the coils in the thimble proportional to the orientation relative to the charge coil and as a function of the relative frequencies. The transmitter turns off. The receiver then measures the field as it decays from each of the three RC circuits, again proportional to the orientation relative to the charge coil and as a function of the relative frequencies. Because of the high frequency of operation relative to finger motion, no motion occurs during the cycle. This cycle is repeated for each axis, providing sufficient data to calculate linear and angular position of thimble relative to watch.

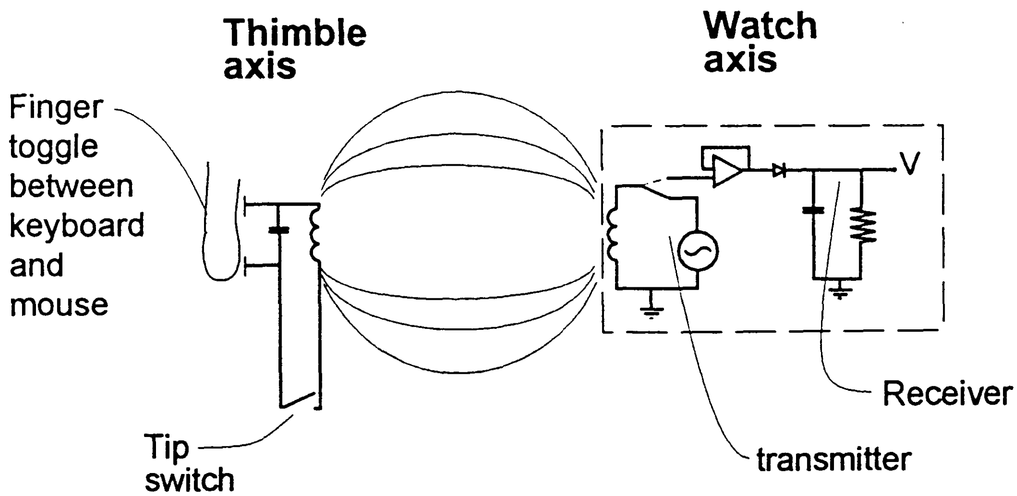



**Figure 62 - MARK Electronics**

5.3.5 Mouse Operation

The device must also provide mouse functionality. The device toggles between keyboard and mouse functionality by the user touching the adjacent finger to the thimble, or not. Figure 61 shows a contact band located on the periphery of the finger mounted device, preferably distal or proximate to a first phalangeal joint when worn on a finger. The band is coupled to one of the RC circuits similar to the coupling of the tip switch, although the coupling is capacitive, as shown in Figure 62.

# 6. Conclusion

The ergonomic threshold for portable product miniaturization defines two distinct regimes of product development. The threshold is defined as the crossover point between which product size is miniaturization-limited by the electronics within, or the external ergonomics. The ergonomic threshold may be characterized as that place in time where product development objectives have the opportunity of diverging between offering optimized ergonomics, optimized size, or a combination there between. Beyond the threshold, products can be designed too small to be comfortably used; or too large to necessitate advanced components.

In the electronic-limited miniaturization regime, product design occurs in a predominantly open-loop fashion, in which mechanical engineers minimize the overall product based on the volume of electronics that must be contained. In the interface-limited regime, a feedback loop is created back to the electrical engineers in which older, less miniaturized components are sufficient; as a function of the degree of ergonomic performance elected by the mechanical engineers.

The economic impact of the ET is to affect the development of at least $124 billion dollars worth of portable products. Case studies support the assertion that product families pass through a size minimum at the ergonomic threshold, and begin to grow thereafter. The current high cost of a next-generation of input device technologies (voice and handwriting recognition) force product development that occurs within mainstream product to use keyboard-based input. Based on the established price/performance curves, these advanced technologies will not be competitive with the keyboard for at least seven years.

Therefore, three input technologies are presented to address this technology gap. The first technology increases input density by an order of magnitude without compromising on ergonomic quality or increasing cost. The second solves an ergonomic problem that prevents non-chorded keyboards from use in the highly miniaturized world of wearable computing. The third offers the advantages of traditional keyboard input in augmented reality applications while providing a second order-of-magnitude size decrease.

# 7. Future Work

Questions and issues to be investigated, as well as additional work to be performed as a consequence of this thesis include:

### 7.1.1 Marketing-related Issues

This thesis identifies the nature and size of the economic dynamic beyond the ET without predicting its character. Identifying the ratio between customer acceptance of products with degraded ergonomics vs willingness to use enlarged devices is a market-related issue.

Identifying the specific degree to which the electronic industry is affected by the decline in need for miniaturized components would be a highly speculative, market-related estimation at this time. This work is better done at a future date.

Also, cost/benefit locations at which consumers embrace these technologies is primarily economic and marketing issues.

### 7.1.2 Reliability Development

SMEK requires significant development to meet 6-sigma quality standards.

### 7.1.3 Augmented Reality Keyboards

AR1 and AR2 requires significant research and development to implement as viable device.

# 8. Bibliography

*Connective Networks in Ergonomics* Edward Franus Elsevier Press 1991

*Investigating Touchscreen Typing: the effect of keyboard size on typing speed*
Sears, A.; Revis, D.; Swatski, J.; Crittenden, R.; Shneiderman, B.
Dept. of Computer Science, Maryland Univ., College Park, MD
*Behavior and Information Technology* vol. 12, no.1 17-22 Jan.-Feb. 1993

*Ergonomics hold key to Keyboard Switch Design*
Ozeki, Kumio, SMK Corp. *Journal of Electrical Engineering* v 30 n318 Jun 1993. p 28-31, 1993

*An Evaluation of Miniaturized Aircraft Keyboards*
Butkus, K.; Hughes, E.; Moroney, W.
Dayton Univ., OH, USA

*Proceedings of the IEEE 1992 National Aerospace and Electronics Conference,*
NAECON 1992 (Cat. No 92CH3158-3) 698-7404 vol. 2

*User Performance on Typing Tasks Involving Reduced-size, Touch Screen Keyboards* Coleman, Marilyn F.; Loring, Beth A.; Wiklund, Michael E. American Institute for Research, Bedford, MA

Vehicle Navigation and Information Systems Conference Proceedings Part 1 (of 2)
*1991 Proceedings - Society of Automobile Engineers* n P-253 pt 1. p 543-549.

*1993 Interrnational Electronic Device meeting Technical Digest.* Piscataway, NJ IEEE, December 5-8, 1993.

*The National Technology Roadmap for SemiConductors, 1994* , Semiconductor Industry Association

*Motorola Museum for Electronics,* Schaumurg,

*New York Times,* Technology Trends June 14, 1995

*Dataquest* April, 1997

Gopher, D., and Raij, D. (1988) *Typing With A Two-Handed Chord Keyboard: Will The Qwerty Become Obsolete?* IEEE Transactions on systems, and and Cybernetics, 18, 601-609.

Paul A David, Clio and the Economics of QWERTY, 75 Am Econ. Rev. 332 1985

Donald A Norman and David E. Rumelhart Studies of *Typing from the LNR Research Group* in Cognitive Aspects of Skilled typewriting 45, (William E. Cooper ed. 1983); and A. Miller & J. C. Thomas, Behavioral Issues in the Use of interactive Systems, 9 Int. J. of Man-Machine Stud. 509 (1977)

*Force sensitivity and Response,* International Ergonomic Compendium 1992

*On-Screen Keyboards: Which Arrangements Should Be Used?* Laurie Quill, David Biers,*Presented At The Human Factors And Ergonomics Society 37Th Annual Meeting,* 1993

Card, S.K, English, W.K., and Burr, B.J. (1978) Evaluation of mouse, rate-controlled isometric Joystick, step keys and text keys for text selection on CRT. *Ergonomics* 21 (8), 601-613.

Norman, D.A., & Fisher, D. (1982). Why alphabetic keyboards are not easy to use: keyboard layout doesn't much matter. *Human Factors,* 24, (5). 509-519

*Text Entry Using Soft Keyboards,* (1997) I. Scott MacKenzie, R. William Soukoreff, and Shawn Zhang Department of Computing and Information Science University of Guelph[1] Julia Barret and Helmut Krueger "Performance effects of reduced proprioceptive feedback on touch typists and casual users in a typing task", BIT; Behaviour and Information Technology. 13(6):373-381, 1994.

Andrew Sears, Doreen Revis, Janet Swatski, Rob Crittenden, and Ben Shneiderman.

*Investigating Touchscreen Typing: The Effect of Keyboard Size on Typing Speed,* BIT; Behaviour and Information Technology. 12(1):17-22, 1993.

Half-Qwerty: A One- Handed Keyboard Facilitating Skill Transfer From Qwerty, Edgar Matias, Scott MacKenzie, Wiliam Buxton, Presented at The Human Factors And Ergonomics Society 37Th Annual Meeting, 1993

PC World, *Better Dictation , Less Typing* June 1997 v15 n6 p61(1)

PC World, *Cheese of Staff* June 10 1997 v15 n6 p61(1).

1995 Electronics Market Data Book, Electronic Industries Association

*Reduced Keyboard Designs Using Disambuation*
Kreifeldt, J.G.; Levine,S.L.; Iyengar, C.
Dept. of Engineering Design, Tufts Univ. Medford, MA
*Proceedings of the Human Factors Society 33rd Annual Meeting,* 1989. Perspectives 441-444 vol. 1

# 9. Appendices

## 9.1 Appendix - Collapsible Keyboard Patent

US005163765A

# United States Patent [19]

Levy

[11] Patent Number: 5,163,765

[45] Date of Patent: Nov. 17, 1992

[54] COLLAPSIBLE KEYBOARD

[75] Inventor: David H. Levy, Santa Clara, Calif.

[73] Assignee: Apple Computer, Inc., Cupertino, Calif.

[21] Appl. No.: 846,188

[22] Filed: Mar. 4, 1992

[51] Int. Cl.[5] ............................ B41J 5/16; B41J 5/12

[52] U.S. Cl. .................................. 400/492; 400/472; 200/5 A; 235/145 R

[58] Field of Search ............... 400/492, 490, 493, 495, 400/472; 200/5 R, 5 A, 515, 514, 513, 512; 235/145 R

[56] References Cited

U.S. PATENT DOCUMENTS

| | | | |
|---|---|---|---|
| 3,699,294 | 10/1972 | Sudduth | 200/5 A |
| 3,940,758 | 2/1976 | Margolin | 200/5 A |
| 4,302,648 | 11/1981 | Sago et al. | 200/515 |
| 4,492,829 | 1/1985 | Rodrique | 200/5 A |
| 4,536,625 | 8/1985 | Bebie | 200/5 A |
| 4,710,597 | 12/1987 | Loheac | 235/145 R |
| 4,800,243 | 1/1989 | Osawa et al. | 200/514 |
| 4,950,874 | 8/1990 | Damitio et al. | 400/472 |
| 4,952,761 | 8/1990 | Viebrantz | 200/513 |
| 5,044,798 | 9/1991 | Roylance et al. | 400/472 |

FOREIGN PATENT DOCUMENTS

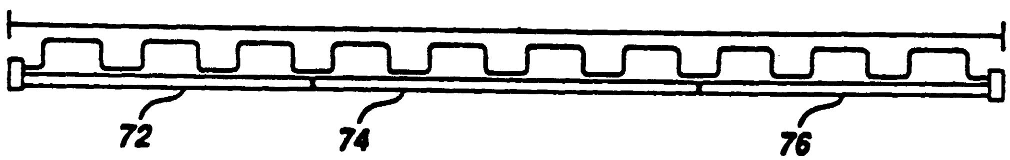
| | | | |
|---|---|---|---|
| 2586117 | 2/1987 | France | 235/145 R |
| 8200064 | 1/1982 | PCT Int'l Appl. | 235/145 R |

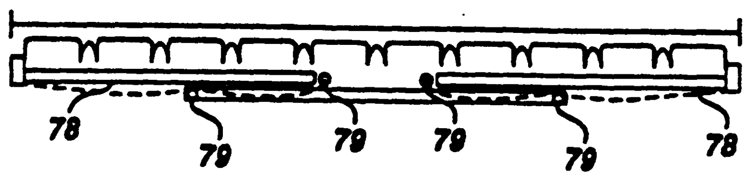
*Primary Examiner*—Edgar S. Burr
*Assistant Examiner*—Anthony Nguyen
*Attorney, Agent, or Firm*—Jonathan B. Penn

[57] ABSTRACT

A collapsible keyboard for use with portable personal computers is disclosed. A first preferred embodiment of the present invention comprises a keyboard molded from a conductive, elastomeric material and a collapsible frame/circuit board substrate, the substrate having a plurality of electrical contacts to indicate to the computer when a key is pressed. As the keyboard is formed from an elastic material, it can be compressed into a first, closed position of minimum size to facilitate carrying the computer and to minimize the computer's size. In the keyboard's second, expanded position, each of the molded keys overlies a pair of contacts. When a key is pressed while the keyboard is in this second position, an electrical circuit is formed by the key and the contacts, indicating to the computer both that a key has been pressed and which key has been pressed. In another embodiment of the present invention, two pairs of contacts underlie each key, one pair being used when the keyboard is in its first position and the other pair being used when the keyboard is in its second position.

5 Claims, 1 Drawing Sheet



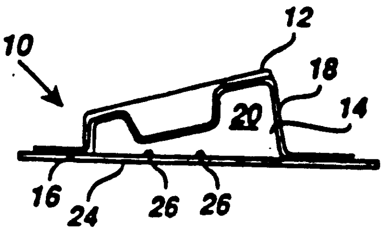



FIG. 1

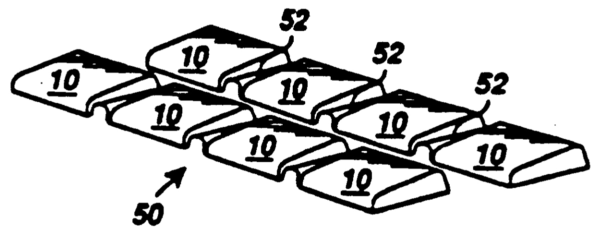



FIG. 3

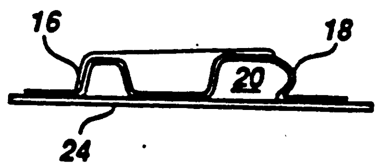



FIG. 2

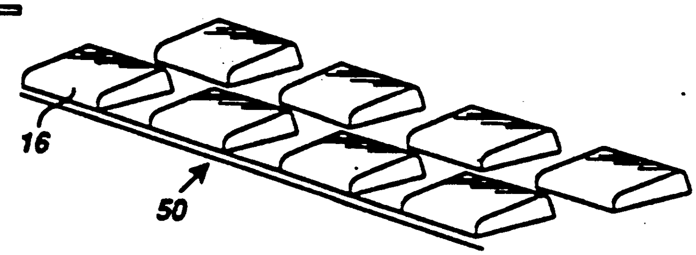



FIG. 4

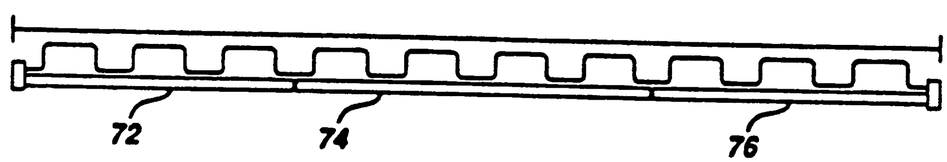



FIG. 5

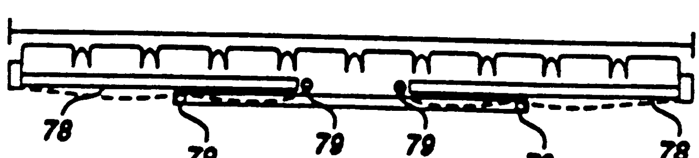



FIG. 6

# COLLAPSIBLE KEYBOARD

## BACKGROUND OF THE INVENTION

This invention is in the field of keyboards. In particular, a first embodiment of the present invention relates to keyboards for portable personal computers.

Alphanumeric keyboards and their use as data input devices to digital computers are known. Although many different types of keyboards are known, including the well-known "QWERTY" keyboard, a standard size keyboard has evolved. The necessity for a standard keyboard size, including a standard size for the individual keys, as well as for the overall size and arrangement of the keys, should be obvious, as it allows an individual to use any keyboard with roughly equal facility.

Although standard size keyboards are adequate for computers used in a single fixed location, they have the obvious drawback of their size when incorporated into the increasingly ubiquitous portable personal computer. In such computers, the electronics which comprise the operative circuitry does not determine the final size of the portable computer. Rather, the input device, such as the keyboard, and the output device, typically a liquid crystal display ("LCD"), dictate the final size.

The keyboard in particular presents a difficult design problem. Obviously, the individual keys used in the typical keyboard for a non-portable computer could be reduced in size. However, the limit beyond which shrinking the individual keys creates increasing user difficulties in operating the keyboard is rapidly reached. Most work on shrinking keyboard size has concentrated on reducing the thickness of the keyboard, not on reducing its length or width.

A keyboard or reduced length and/or width, which nonetheless retains the operating convenience inherent in a standard, full-size keyboard, would permit further reductions in the size of portable computers. To date, no such keyboard is known.

## SUMMARY OF THE INVENTION

The present invention, in its first preferred embodiment, comprises a molded plastic keyboard mounted in a length-wise collapsible frame. The key tops are molded of a hard plastic material. The rest of the keys and the keyboard itself are molded from a flexible conductive elastomeric material. This assembly is mounted on a metal or plastic frame that can be collapsed and expanded along its length-wise axis. As the keyboard is formed from a flexible material, it can be "accordioned" to form a very compact and short keyboard assembly when not in use. In the first embodiment, the keyboard contacts would only be operable when the keyboard was in its expanded condition. In a second embodiment, the keyboard could be operated in its first, compressed condition as well as in its expanded condition.

The present invention will now be described in detail with reference to the figures listed and described below.

## BRIEF DESCRIPTION OF THE ILLUSTRATIONS

FIG. 1 shows one key of the present invention in an inactive position;

FIG. 2 shows the key of FIG. 1 is an active position;

FIG. 3 shows the entire keyboard of the present invention in a compacted position;

FIG. 4 shows the keyboard of FIG. 3 in an expanded position;

FIG. 5 shows the keyboard and frame of the present invention in an expanded position; and

FIG. 6 shows the keyboard and frame of FIG. 5 in a compacted position.

## DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

A single key of the keyboard and frame that comprise the first embodiment of the present invention is shown in FIG. 1. Key 10 is comprised of two different plastics. Top surface 12 of key 10 is formed from a hard plastic. Although any type of hard, impact-resistant plastic would be acceptable, one preferred type of plastic is acrylonitrile butadiene styrene ("ABS"). Body 14 of key 10 is formed from a conductive, flexible elastomeric material. Although any one of several types of elastomeric material is acceptable, key body 14 in this embodiment is made from a silicon elastomer doped with a conductor such as carbon or silver. This type of elastomer is known. It should be noted that the entire keyboard, not just key 10, is fabricated at the same time as part of a two-step injection molding process. In this two-step process, key tops 12 would be injected and solidified first, followed by the flexible elastomeric material.

Key 10 is fashioned so that side 16 of the key is much shorter than the other sides. Side 16 acts as the hinge of the key, so that when key 10 is depressed, it pivots from side 16 downward. When key 10 is pressed into an active position, as shown in FIG. 2, long side 18 of key 10 is compressed, as well as right side 20 and the left side (not shown). This structure also insures that when key 10 is depressed, it will reliably contact the same spot on underlying printed circuit board 24, which has a plurality of key contact points 26, a pair of contacts 26 underlying each key 10. Although there is decoding circuitry which indicates to the computer coupled to the present invention which key has been pressed, that circuitry is known and forms no part of the present invention. Therefore it is not illustrated herein.

FIG. 3 shows how a group of keys 10 would appear when they form part of a complete keyboard assembly, which is shown in its compressed state in FIG. 3. The elastomeric material which forms the bulk of keyboard 50 is compressed between each key 10, forming a row of accordion pleats 52 between each key 10. In the expanded position of keyboard 50 shown in FIG. 4, these ridges of material lie flat against the printed circuit board which forms the substrate of keyboard 50.

One possible frame arrangement for keyboard 50 is shown in FIGS. 5 and 6. In FIG. 5, keyboard 50 is held in its expanded position by frame 70, which is here comprised of three sliding member 72, 74 and 76. In this expanded position, middle sliding member 74 is locked into place between members 72 and 76, holding the keyboard in its open position. In the closed position illustrated in FIG. 6, pins 79 in members 72 and 76 slide over the top of member 74 along rails 78, allowing the keyboard to be compressed into its closed position. As noted on FIGS. 5 and 6, the keyboard in its collapsed position measures 205 mm and 274 mm in its expanded position. Although one particular frame arrangement is illustrated herein, many two position, mechanically collapsible frames are known and the use of any one of these would be within the scope of the present invention.

Although the embodiment illustrated herein uses only a single pair of contacts beneath each key (see FIG. 1), one variant that would have a great deal of utility uses two pairs of contacts beneath each key, one contact pair being activated when the key is pressed when the keyboard is in its open, expanded position and the other contact pair being activated when the key is pressed when the keyboard is in its closed position. Although normal typing would be very difficult, if not impossible while the keyboard is in its closed position, there would be many situations where only a few keys would need to be struck. Such situations could include opening a file or recalling an address. This two contact pair arrangement would allow for limited key use while the keyboard was in its closed position.

Given the range of possible variations in the present invention, including the use of different types of mechanically expanding and collapsing frames, the different types of elastomeric material that could be used, and the different contact arrangements that could be made to allow the keyboard's use in two or more positions, it is intended that the appended claims be interpreted as including the mentioned variations as well as all other changes and modifications.

What is claimed is:

1. A collapsible keyboard comprising:

frame means having a closed first position and an open second position, the frame means additionally comprising a first plurality of contact means capable of generating a signal; and

keyboard means comprising an elastomeric and conductive substrate, the substrate being molded to form a keyboard with a plurality of keys, the top of each of the keys being formed from a hard, impact-resistant plastic, the elastomeric substrate being capable of being collapsed into a first closed position and expanded into a second, open position, the keys overlying the first contact means when in the open position, an electrical circuit being formed when the keys are pressed to contact the contacts, the creation of the electrical circuit indicating to a computer to which the keyboard has been coupled that a key has been depressed.

2. A method for creating a collapsible keyboard comprising the steps of

molding a plurality of key tops from a hard plastic material;

molding a plurality of key bodies and material to interconnect the key bodies from a conductive elastic material, the key bodies being molded onto the key tops; and

mounting the key tops, key bodies and interconnecting material unto a collapsible frame, the frame having at least a first, open position and a second, closed position, the key bodies overlying a first plurality of key contacts when the frame is in the first position.

3. The method of claim 2, wherein the key bodies overlie a second plurality of key contacts when the frame is in the second position.

4. The collapsible keyboard of claim 1 wherein the keys overlie a second plurality of contact means when the frame is in the first, closed position.

5. A collapsible keyboard comprising:

flexible conductive elastomer keyboard body having a plurality of flexible molded key bodies;

a plurality of key tops comprised of a hard plastic material mounted on the plurality of key bodies;

printed circuit means having a plurality of key contact points, the printed circuit means being capable of being collapsed to a first closed position and opened to a second, expanded position;

frame means having a first closed position and a second, expanded position, the printed circuit means being mounted on the frame means and the keyboard body being mounted on the printed circuit means, the key bodies and attached key tops overlying the key contact points of the printed circuit means when the frame means is in the second position.

* * * * *

## 9.2 Appendix - Finger Pressure

| Finger Min | Finger Max | | Thumb Min | Thumb Max |
|---|---|---|---|---|
| 0.37 | 0.6 | | 0.53 | 0.67 |
| 0.45 | 0.45 | | 0.67 | 0.49 |
| 0.4 | 0.5 | | 0.53 | 0.69 |
| 0.34 | 0.4 | | 0.75 | 0.94 |
| 0.46 | 0.48 | | 0.5 | 0.71 |
| 0.55 | 0.56 | | 0.41 | 0.5 |
| 0.42 | 0.46 | | 0.53 | 0.6 |
| 0.41 | 0.485 | | 0.53 | 0.63 |
| 0.42 | 0.46 | | 0.47 | 0.51 |
| 0.37 | 0.45 | | 0.44 | 0.58 |
| 0.37 | 0.38 | | 0.47 | 0.64 |
| 0.48 | 0.5 | | 0.48 | 0.52 |
| 0.32 | 0.45 | | 0.56 | 0.71 |
| 0.45 | 0.46 | | 0.48 | 0.6 |
| 0.42 | 0.47 | | 0.54 | 0.55 |
| 0.43 | 0.5 | | 0.53 | 0.66 |
| 0.37 | 0.43 | | 0.49 | 0.51 |
| 0.31 | 0.41 | | 0.52 | 0.6 |
| 0.59 | 0.59 | | 0.45 | 0.62 |
| 0.49 | 0.64 | | 0.51 | 0.55 |
| 0.29 | 0.36 | | 0.47 | 0.51 |
| 0.34 | 0.38 | | 0.6 | 0.85 |
| 0.49 | 0.6 | | 0.61 | 0.93 |
| 0.38 | 0.42 | | 0.49 | 0.7 |
| 0.44 | 0.46 | | 0.55 | 0.59 |
| 0.44 | 0.48 | | 0.44 | 0.45 |
| 0.4 | 0.5 | | 0.46 | 0.8 |
| 0.37 | 0.41 | | 0.47 | 0.54 |
| 0.4 | 0.55 | | 0.41 | 0.49 |
| 0.46 | 0.54 | | 0.5 | 0.6 |
| 0.39 | 0.5 | | 0.53 | 0.61 |
| 0.42 | 0.49 | | 0.52 | 0.62 |
| 0.5 | 0.56 | | 0.47 | 0.54 |
| 0.4 | 0.43 | | 0.44 | 0.47 |
| 0.33 | 0.4 | | 0.46 | 0.53 |
| 0.41 | 0.41 | | 0.58 | 0.61 |
| 0.41 | 0.49 | | 0.54 | 0.57 |
| 0.41 | 0.42 | | 0.44 | 0.61 |
| 0.4 | 0.45 | | 0.42 | 0.57 |
| 0.43 | 0.45 | | 0.51 | 0.64 |
| 0.35 | 0.41 | | 0.47 | 0.65 |
| 0.38 | 0.51 | | 0.39 | 0.46 |
| 0.34 | 0.45 | | 0.55 | 0.76 |
| 0.39 | 0.51 | | 0.43 | 0.52 |
| 0.5 | 0.61 | | 0.44 | 0.52 |
| 0.4 | 0.44 | | 0.44 | 0.54 |

## 9.3 Appendix - Key Density Distribution

| | # in grid | grid width | Grid Height | # Extra keys | Overall width | Overall height | key density | total keys |
|---|---|---|---|---|---|---|---|---|
| TI data bank | 34 | 95 | 30 | 8 | 105 | 60 | 7.69473684 | 42 |
| Casio DC2000 | 30 | 110 | 30 | 12 | 116 | 70 | 5.86363636 | 42 |
| Sharp EL6065 | 39 | 105 | 40 | 18 | 105 | 65 | 5.98928571 | 57 |
| TI PS 2460i | 32 | 118 | 40 | 14 | 133 | 80 | 4.37288136 | 46 |
| Casio 7800 | 30 | 125 | 34 | 32 | 130 | 85 | 4.55294118 | 62 |
| Sharp 6490 | 46 | 125 | 50 | 15 | 135 | 85 | 4.7472 | 61 |
| TI 3960 | 44 | 115 | 45 | 16 | 145 | 85 | 5.48405797 | 60 |
| Casio 4600C | 46 | 130 | 38 | 13 | 140 | 83 | 6.00607287 | 59 |
| Sharp YO-430 | 39 | 130 | 45 | 38 | 135 | 38 | 4.3 | 77 |
| Casio SF 5580 | 39 | 155 | 45 | 42 | 165 | 90 | 3.60645161 | 81 |
| TI 6960 Si | 48 | 155 | 47 | 20 | 130 | 85 | 4.24982841 | 68 |
| Sharp Zaurus | 42 | 145 | 45 | 23 | 150 | 85 | 4.15172414 | 65 |
| Casio Cassiopia | 61 | 155 | 60 | 3 | 165 | 92 | 4.23064516 | 64 |
| | | | | | | | | |
| | | | | | | | | average |
| TI85 | 35 | 75 | 75 | | | | 4.01333333 | 60.308 |
| HP17B | 42 | 72 | 80 | | | | 4.703125 | |
| sharp 6050B | 39 | 101 | 40 | | | | 6.22648515 | |
| HP11C | 40 | 118 | 50 | | | | 4.37288136 | |
| | | | | | | | | |
| remote fs185b | 27 | 41 | 70 | | | | 6.06794425 | |
| | | | | | | | | |

## 9.4 Appendix - SMEK Patent

US005612690A

# United States Patent [19]

Levy

[11] Patent Number: 5,612,690

[45] Date of Patent: Mar. 18, 1997

[54] **COMPACT KEYPAD SYSTEM AND METHOD**

[76] Inventor: **David Levy**, 723 Raney Ct., Santa Clara, Calif. 95050-5533

[21] Appl. No.: **71,242**

[22] Filed: **Jun. 3, 1993**

[51] **Int. Cl.**$^{6}$ .................................................. **H03K 17/94**

[52] **U.S. Cl.** ........................ **341/22**; 364/709.15; 400/485

[58] **Field of Search** ........................ 341/22, 34; 345/168, 345/173; 200/5 A; 400/485; 364/709.15; 340/543, 426, 430

[56] **References Cited**

U.S. PATENT DOCUMENTS



| | | | |
|---|---|---|---|
| 4,891,777 | 1/1990 | Lapeyre | 364/709.15 X |
| 4,912,462 | 3/1990 | Washizuka et al. | 345/173 |

OTHER PUBLICATIONS

"Compact Keyboard", IBM Tech. Disch. Bulletin (Beausoleil et al.), vol. 13, No. 11, Apr. 1971.


*Primary Examiner*—Thomas Mullen

[57] **ABSTRACT**

In a key pad key cap symbols are placed at the interstices between key caps as well as at their centers. The key pad electronics are designed to register simultaneously actuated key caps at an interstice as an input uniquely associated with the symbol locate at the interstice. The linear dimension of a row of key caps are approximately one-half that of a conventional key cap requiring approximately two key caps to form a linear dimension sizable enough to comfortably accommodate the tip of an adult finger. Additionally, the symbols located at the centers of key caps are elevated slightly, allowing unimpeded and ergonomic access to individual keys as well as the interstices.

**9 Claims, 3 Drawing Sheets**

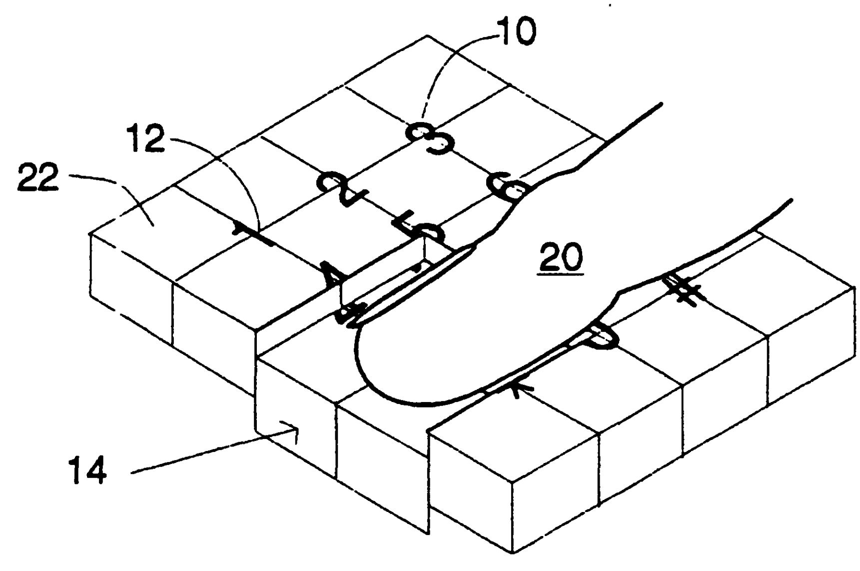


FIGURE 1

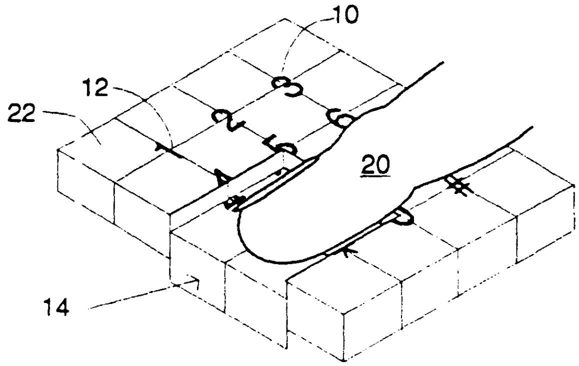



FIGURE 3

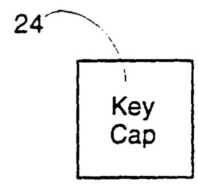



FIGURE 2

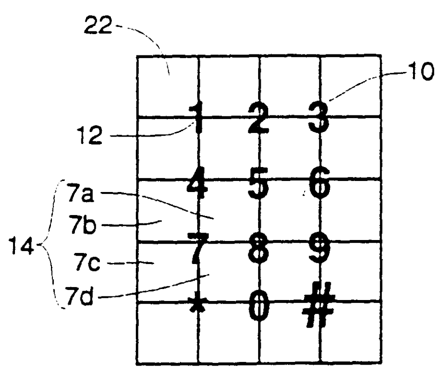



FIGURE 4

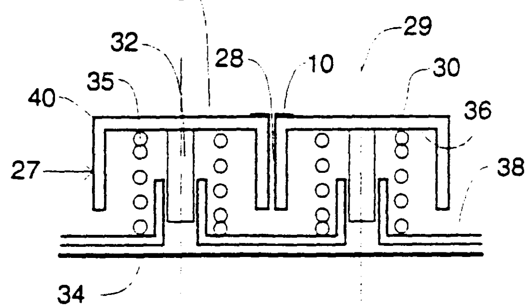



FIGURE 5

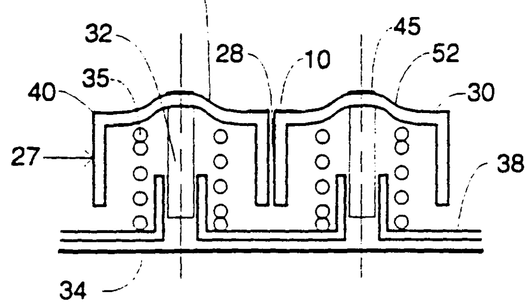


FIGURE 6

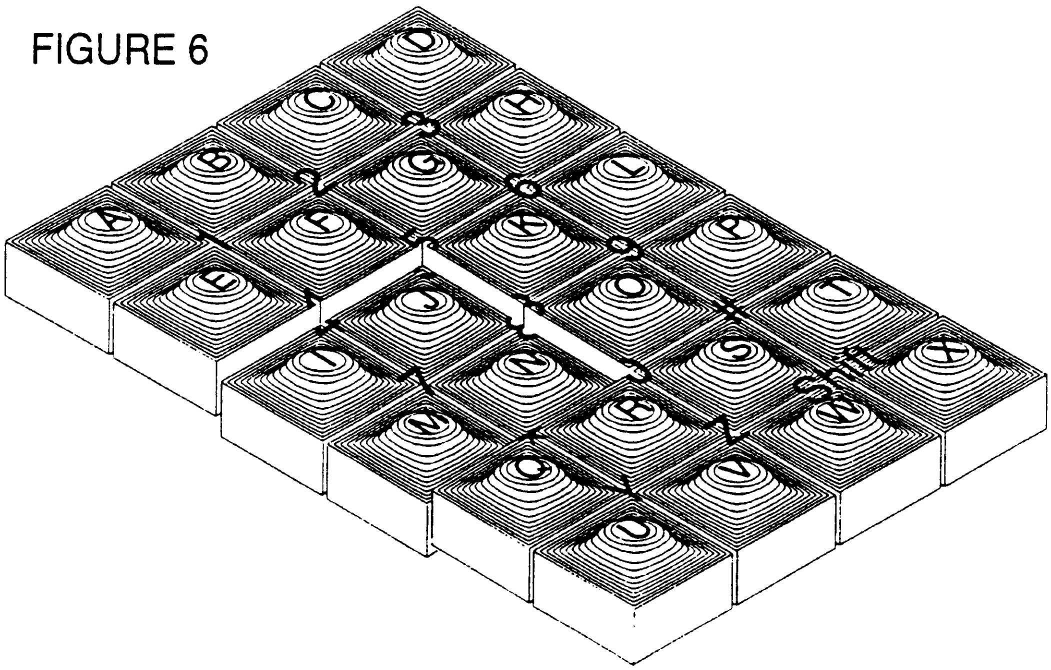


FIGURE 7a

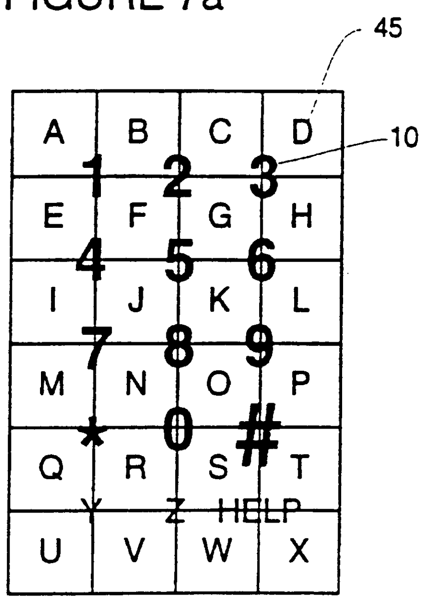



FIGURE 7b

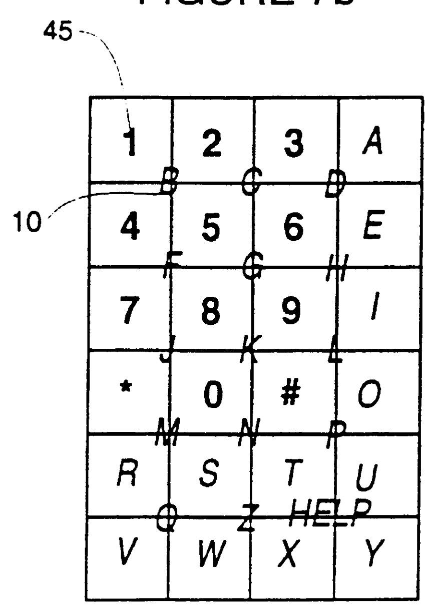


FIGURE 8

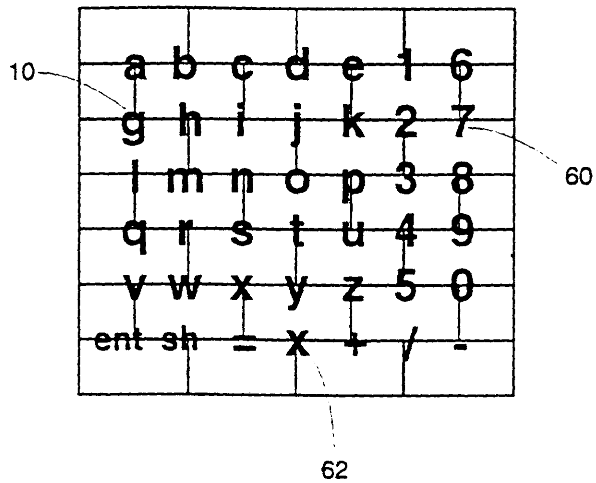
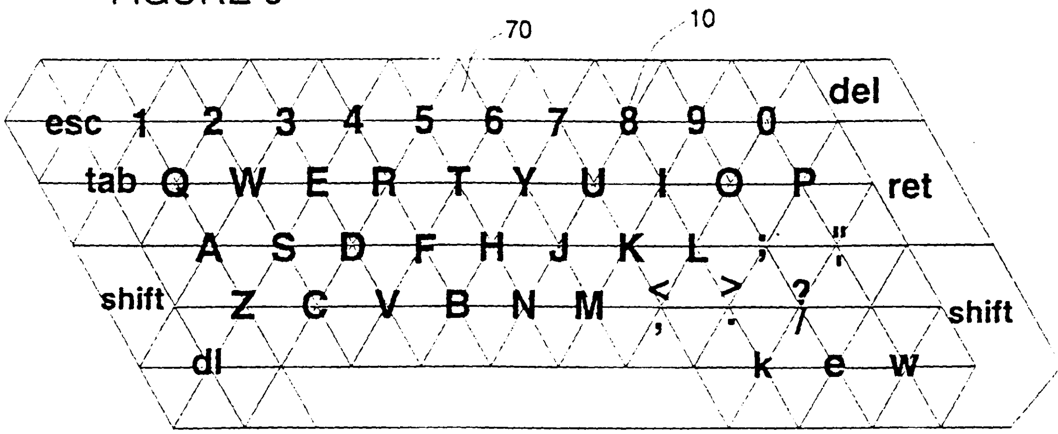



FIGURE 9



# COMPACT KEYPAD SYSTEM AND METHOD

## FIELD OF THE INVENTION

This invention is in the field of keypad input for small electronic devices, particularly for communications and combination locks.

## BACKGROUND OF THE INVENTION

Electronic devices continue to become smaller and more complex. As the complexity increases, there is a tendency for these devices to require more buttons, including entire alphanumeric character sets and as well as additional function-specific keys. Combination keypad locks are more difficult to pick when an increased number combinations are possible. Specific examples include: remote control units for complex tasks such as interactive television; cellular telephones and pagers with written communication capabilities; combination keypad lock interfaces; and wired telephones with peripheral control abilities.

Such devices are problematic because the human hand remains relatively constant in size while the componentry shrinks. The result is that the interface to the hand, the keypad, often dictates the smallest possible size of an electronic device. It is therefore increasingly important to minimize the size of the keypad without reducing the size of the keys smaller than the human hand may use comfortably.

Previous efforts to address this issue consist of the following:

Scale the Keypad Smaller

The most obvious way to increase the number of functions in a given area is to scale the keypad smaller reducing key cap size and decreasing the distance between key cap centers. This technique causes the user to feel constrained. Products which use small keypads suffer from the impression that they are toy-like, largely because, in fact, they are scaled for use by a child. Decreasing key cap size makes the keys less comfortable to a full-sized adult finger. Decreasing the distance between key cap centers increases the likelihood of accidental input. Decreasing the size of the legend reduces legibility and the ease of viewing. In these ways, this solution is workable but far from optimal.

Chorded keyboards

Chorded keyboards have a relatively small number of keys, often linearly disposed to conform with the resting position of the human hand, which operate in combination to form each input, offering 2 raised to the N number of different inputs where N is the number of keys. The primary problem with this solution is that it requires the user to memorize 2^N input combinations and to develop the necessary motor skills. While this is a workable solution for extremely dedicated users, it is impractical and requires an unrealistically large amount of practice for most.

Modal Solutions

There are a variety of solutions in which individual keys are given a plurality of functions: Time variance modality varies the time between key operations to define the output function. This is non-intuitive and severely restricts the pace at which an operator may work; Function key modality alters the functionality of the keystroke(s) which follow in a predetermined manner. While this is an extremely common technique, it has limitations. While a function key doubles the possible outputs of a keypad, it also doubles the number of keystrokes required. Further, it is impractical to reduce the dimension of the standard 10-key keypad or the space required by the 26 letters of the alphabet by implementing a function key solution because the characters of these sets are of near-equal significance. Force sensitive modality incorporates a plurality of force-levels to a single button, and thereby a plurality of functions. Tests show that there is a wide variation in the forces naturally applied by users and wide variations in the levels of force sensitivity between users. The solution is highly non-intuitive. For these reasons, force sensitive keys are not widely applicable, nor desirable.

What is desirable is a keypad which increases the number of functions which can be output from a given number of keys, without compromising the ergonomic advantages of full-size and individually operable key caps. It is further desirable to not require any learning or practice. The keypad would be non-modal to eliminate the confusion and errors commonly associated with modalities. Each output function would permanently correlate with a single key cap location. The identifying legends would be full-size for easy identification. Functions would be accessible by a single finger push. The keypad would be small without being toy-like because it is designed to accommodate a full-size human finger. Ideally a compact keypad would effect full-sized key caps with on center distances smaller than full-sized key cap dimensions allow.

## SUMMARY OF THE INVENTION

In this invention the above limitations are overcome and objects and advantages achieved by placing the key cap legends at the interstices of (rather than at the centers of) the key caps. The electronics are designed to register the simultaneous actuation of the plurality of key caps associated with each interstice. Further, in a preferred embodiment, the size of each key is reduced to approximately half that of a standard full-sized key cap. Therefore, as a finger presses an interstice, it presses on an area approximately equal to a standard key cap, but is actuating a plurality of key switches rather than only one. In this manner 2N−1 separate inputs may be ergonomically accommodated in the linear width which would otherwise accommodate only N inputs.

The international (ISO) standard for the smallest dimension between key switches (herein called the key switch dimension) was established at 18 mm because this is considered to be the smallest dimension which allows the human finger easy access to a specific key without the risk of accidentally actuating an adjacent key. The present invention overcomes this limitation by using each area of the keypad redundantly, thereby increasing the number of functions which may be comfortably and reliably accessed in a given area. The advantages provided by the invention are independent of the key switch technology used to implement it.

A further increased number of inputs are made available by additionally using each key individually. There is an apparent contradiction in reducing the size of said key caps and the stated goal of offering ease of use to the user. However, in the preferred embodiment a contoured key cap elevates the centers of each individual key cap, thereby allowing comfortable and ergonomic access to each individual key as well as to the interstices which comprise the aforementioned inputs. In embodiments with displaceable key switches, the height of the contour is approximately equal to the distance of key switch displacement. The gaps between the gentle contours offer comfortable and ergonomic access to the inputs located at the interstices because of the increased contact area afforded by the Pythagorean distance between key cap centers and the subtlety of the contour.

The improved functionality is substantial. For example, the preferred embodiment allows 52 additional independently actuatable functions (61 total) into the area normally occupied by nine standard keys. This is accomplished without decreasing the level of ergonomic comfort to access any of the functions. Conversely, the invention may be used to decrease the size of an existing keypad while retaining the existing functionality, ergonomics and ease of use.

The surface which comes into contact with the finger is here called the contact surface. In the present invention the contact surface may be implemented with a continuous elastomeric or plastic material and the key switches are not required to physically displace during actuation. However, there is a wide range of capacitive and force sensitive pad technologies which use analog measuring techniques to establish the position of the finger. These technologies are inapplicable to the present invention which requires at least two discrete and independently operable key switches to be used for a single functional input. The use of discrete switches allows the full integration of the position and actuation functional aspects that are inherent to the operation of a keypad, whereas these functional aspects are independently achieved in analog position sense technologies, thereby adding to their complexity and cost.

The contact surface of the present invention, if comprised of discrete key caps, are designed differently than standard discrete key caps. Standard key caps have a skirt, or taper to provide visual and tactile differentiation between key caps. This also serves to prevent accidental inputs from occurring. The present invention is the opposite in design. The distance between key cap top surfaces is minimized with the goal of reducing the visual and tactile void between adjacent key caps.

The electronics of the present invention are able to sense the simultaneous actuation of adjacent key switches. However, not all key switches associated with an interstice must be actuated to uniquely identify an interstice. Therefore, the electronics are designed to interpret the user's intended input based upon the minimum number of key switch actuations to uniquely identify the interstice. A brief timing delay (on the order of 0.2 seconds) may be incorporated to eliminate non-intended actuations. The implementation of these techniques are known to those in the art.

It is therefore a goal of the present invention to provide an compact keypad in which each function may be actuated comfortably by an adult-sized human finger.

It is an additional goal of this invention to provide an increased number of functions within a given area without compromising the ergonomic advantages of full-size key caps.

It is an additional goal of this invention to provide a compact keypad in which each function may be accessed by a single finger motion.

It is a further goal to provide an increased number of inputs without requiring memorization, training to use, or the introduction of a modality.

It is yet a further goal of this invention to provide a means to derive the above benefits without the need to restrict the discrete key switch technology used for its implementation, except for specifically not using analog or continuous technologies which would have a detrimental impact on cost.

## BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 shows a perspective view of a finger actuating the function "number 7" in a telephone keypad embodiment of the present invention.



FIG. 2 shows a plan view of the drawing of FIG. 1 with the finger removed.

FIG. 3 shows a standard key cap as a dimensional reference.

FIG. 4 shows a side view of generic key cap and key switch as a reference.

FIG. 5 shows a side view of the preferred embodiment.

FIG. 6 shows a perspective view of the preferred embodiment.

FIGS. *7a* and *7b* show plan views of two potential layouts of alphanumeric character sets using the preferred embodiment.

FIG. 8 shows an embodiment of an alphanumeric character set and calculator keys in a brick patterned compact keypad.

FIG. 9 shows an embodiment with an alphanumeric character set with triangular key cap sub-units.

## REFERENCE NUMERALS IN DRAWINGS

*7a–7d* Key caps associated with the number "7"
**10** Interstice legend
**12** Interstice
**14** Associated key caps
**20** Finger
**22** Key cap
**24** Standard individual key cap
**27** Key switch mechanism
**28** Side
**29** Central axis
**30** Top surface
**32** Contact element
**34** Input element
**35** Spring element
**36** Inner surface
**38** Housing
**40** Radius
**45** Center legend
**50** Dome
**52** Finger curve
**60** Alphanumeric character set
**62** Calculator keys
**70** Triagonal key caps

## DETAILED DESCRIPTION

FIG. 1 shows the present invention embodied as a compact telephone keypad during actuation by a finger **20**. Each interstice legend **10** is placed at the interstice **12** formed by the intersection of associated key caps **14**. The term "associated key caps" **14** will be used throughout this document to refer to the set of individual key caps **22** which are designed to operate in unison and thereby inform the control electronics that the user intends the function identified by the interstice legend **10** located at the intersection of the associated key caps **14**. In this embodiment each interstice legend **10** has four associated key caps **14** and each individual key cap **22** is sized such that each group of four associated key caps **14** are approximately comparable in size to a standard individual key cap **24**, as shown in FIG. 3. The finger **20** is shown pressing the associated key caps **14** which actuate the number "7". The figure is not to scale.

FIG. 2 shows a plan view of the keypad shown in FIG. 1 with the finger **20** removed for clarification. The four key caps **22** associated with the legend **10** "7" are labeled *7a*, *7b*, *7c*, and *7d*.

This embodiment of the invention allows approximately 2n−1 legends, and hence 2n−1 functions, to fit into the linear distance which normally allows only N legends (and N functions) using key caps the size of standard individual key caps 24. FIG. 2 is submitted as a full-scale drawing, allowing the reader to more easily imagine using the invention.

FIG. 3 shows a standard individual key cap 24 as reference. This figure may be used to establish a real world scale for the other figures independent of the scale to which these drawings may be reproduced.

FIG. 4 shows two generic displacement actuated discrete key switches as a reference. The figure is not to scale. There are three primary differences between the design parameters of key switches for standard descrete key caps and those of the key switches for this invention. In the preferred embodiment, the key caps of the invention are significantly smaller than conventional key caps. The reduced dimension of the top surface places less torque on the key switch mechanism **27** which may therefore be of lower tolerance and/or rigidity without resulting in the key caps binding due to torque applied off the central axis **29**. The side **28** of each key cap **22** abuts the side **28** of the adjacent key cap **22** and the peripheral edge defined by the radius **40** is closer to the adjacent key cap **22** and its radius **40** than adjacent key caps of the standard individual key cap **24** design. The distance between the radii **40** of adjacent top surfaces **30** are approximately 1 mm. A contact element **32** contacts the input element **34** when the key cap **22** is pressed. A spring element **35** exerts force upon the inner surface **36** and the housing **38** and raises the key cap **22** when it is not in use. The force created by the spring element **35** is chosen according to the number of key caps **22** which comprise the associated key caps **14** in the chosen embodiment. (Embodiments with fewer and greater number of associated key caps **14** are discussed below.) The top surface **30** is relatively flat and onto it is disposed the interstice legend **10**. The dimension between the the central axes **29** is known as the key switch dimension **33**.

FIG. **5** shows the preferred embodiment, implemented with the same key switch mechanism as shown in FIG. **4**. A center legend **45** is aligned with the central axis **29** and elevated above the top surface **30** by a gently contoured dome **50**. Other contours may be used. The center legend **45** is elevated by approximately the same distance as the stroke of the key switch mechanism **27**. An interstice legend **10** is disposed onto the top surface **30**. The graphic comprising each center legend **45** is disposed, approximately equally on all four of the associated key caps **14**.

FIG. **6** shows an perspective view of the preferred embodiment with the number "7" actuated. The drawing is larger than life size.

FIG. *7a* shows a plan view of the drawing of FIG. **6**: a standard numeric keypad implemented on the interstice legends **10** and the alphabet on the center legends **45**. The drawing is to scale, allowing the reader to more easily imagine using this keypad.

FIG. *7b* shows an embodiment with a standard numeric keypad implemented on the center legends **45** and the alphabet on the interstice legends **10**. The drawing is to scale, allowing the reader to imagine using this keypad.

FIG. **8** shows an embodiment of an alphanumeric character set **60** and calculator keys **62** in a brick patterned compact keypad. The drawing is to scale.

FIG. **9** shows an embodiment of an alphanumeric keypad using triagonal key caps **70** forming hexagons with interstice legends **10**. The primary advantage to this configuration is to offset each key in a manner similar to a standard typist's keyboard. This offers full-sized keys in a well-known familiar configuration in approximately one-third the area. The drawing is to scale.



While particular embodiments of the particular invention have been shown and described, it will be obvious to those skilled in the art that changes and modifications may be made without departing from this invention in its broader aspects and therefore, the appended claims are to encompass within their scope all such changes and modifications as fall within the true spirit and scope of this invention. Accordingly, the scope of the invention should not be limited to the embodiment illustrated, but by the appended claims and their legal equivalents.

I claim:

1. A compact keypad system comprising:

a housing;

a plurality of key switch means disposed on said housing;

a contact surface coupled to said plurality of key switch means;

a plurality of first symbols disposed upon said contact surface, wherein each of said first symbols (i) graphically identifies an associated location on said contact surface, (ii) corresponds with an associated one of a plurality of unique output functions, and (iii) is approximately equidistant to the key switch means of an associated one of a plurality of subsets, each subset comprising at least two adjacent ones of said plurality of key switch means wherein at least one of the dimensions between adjacent ones of said plurality of key switch means is less than the width of the adult human finger; and

an electronic detection circuit sensing the simultaneous actuation of said plurality of key switch means in a subset and subsequently outputting a first signal corresponding to said unique output function associated with said subset and its associated first symbol.

2. The invention of claim 1 wherein said contact surface includes a plurality of key cap means with a contoured to surface such that a center portion of each key cap means is approximately 2–3 mm higher than an outer portion of that key cap means wherein each of the key cap means has disposed on said center portion an associated one of a plurality of second symbols, thereby identifying an output function uniquely associated with each one of said key cap means.

3. A method for miniaturizing the keys of a keypad such that the contact area of each key is smaller than the associated contact area of the adult human finger without compromising ergonomic efficiency comprising the steps of:

a) designing a top surface with a set of key caps to allocate a plurality of areas on said top surface for the placement of a corresponding plurality of graphic legends thereon such that each of the legends is readily associated with an associated one of a plurality of subsets of said key caps, wherein each subset comprises two or more key caps, wherein at least one of the dimensions of each of said key caps, as measured along said top surface, is less than the width of the adult human finger;

b) disposing upon each of said areas an associated one of said plurality of graphic legends, each of which designating an associated one of a plurality of unique functions associated with said subsets;

c) coupling key switch means beneath each of said key caps; and

d) generating a plurality of signals, each of said plurality of signals to be generated in response to the simultaneous pressing of the key caps of an associated one of said plurality of subsets of said key caps and the resulting actuation of at least an associated subgroup of key switch means, said subgroup consisting of any set of key switch means within a subset which uniquely identifies the associated subset from the remaining plurality of subsets.

4. The invention of claim 3 further comprising the steps of:

e) elevating a center portion of each key cap by approximately 2 to 3 millimeters; and

f) printing a central graphic legend on said center portion of each of said key caps, said central graphic legend indicating an operation associated with said key switch located directly beneath said central graphic legend.

5. A compact keypad system including:

a first plurality of contact surface areas, each one of said first plurality having an associated one of a plurality of legends, adjacent ones of said first plurality located at an approximate unit distance X from each other, said first plurality disposed proximate to a first reference plane;

a second plurality of contact surface areas, each one of said second plurality having an associated one of said plurality of legends, each one of said second plurality located at said approximate unit distance X from adjacent ones of said second plurality, said second plurality disposed proximate to a second reference plane, said second reference plane approximately parallel with said first reference plane,

wherein each one of said second plurality is disposed between ones of said first plurality such that each one of said second plurality is approximately equidistant to adjacent ones of said first plurality;

and said second reference plane is displaced from said first reference plane by approximately 2-3 mm, thereby providing comfortable and ergonomic access to each function of said compact keypad system.

6. The invention of claim 5 wherein said unit distance X is defined as approximately one-half the distance occupied by a standard dimension key cap, or approximately 10 mm, thereby providing comfortable and ergonomic access to approximately 2N−1 contact surface areas within said first plurality and approximately 2N contact surface areas within said second plurality, and hence a total of approximately 4N−1 different legends in a row of length equal to that of a row of N standard dimension contact surface areas of approximately 18 mm in width.

7. The invention of claim 6 wherein said first plurality of contact surface areas and said second plurality of contact surface areas are both disposed in a predominantly undulating surface contour that connects contact surface areas of said first plurality to contact surface areas of said second plurality.

8. A compact keypad for inputting data with a human finger, said keypad comprising:

a first key switch;

a second key switch;

a third key switch;

a fourth key switch;

a fifth key switch;

a sixth key switch, wherein said key switches are disposed in a matrix comprising two horizontal rows and three vertical rows, wherein the set of first, second, and third key switches are above and adjacent to the set of fourth, fifth, and sixth key switches and the set of first and fourth key switches are to the left of and adjacent to the set of second and fifth key switches and the third and sixth key switches are to the right of and adjacent to the set of second and fifth key switches, and wherein at least one of the dimensions between adjacent ones of said plurality of key switch means is less than the width of the adult human finger;

a first associated set consisting of the first, second, fourth, and fifth key switches;

a second associated set consisting of the second, third, fifth, and sixth key switches;

a first electrical signal corresponding to the actuation of any of a plurality of subsets of key switches of said first associated set which uniquely identify said first associated set;

a second electrical signal corresponding to the actuation of any of a plurality of subsets of key switches of said second associated set which uniquely identify said second associated set;

electronics to generate the first and second signals;

a plurality of key cap means wherein each of said key cap means is disposed above an associated one of said key switches;

a first legend, said first legend disposed on said key cap means approximately equidistant from said first, second, fourth and fifth switches; and

a second legend, said second legend disposed on said key cap means approximately equidistant from the second, third, fifth and sixth key switches.

9. The invention of claim 8 wherein each of said key cap means has a contoured shape such that a center portion is approximately 2-3 mm higher than an outer portion; and

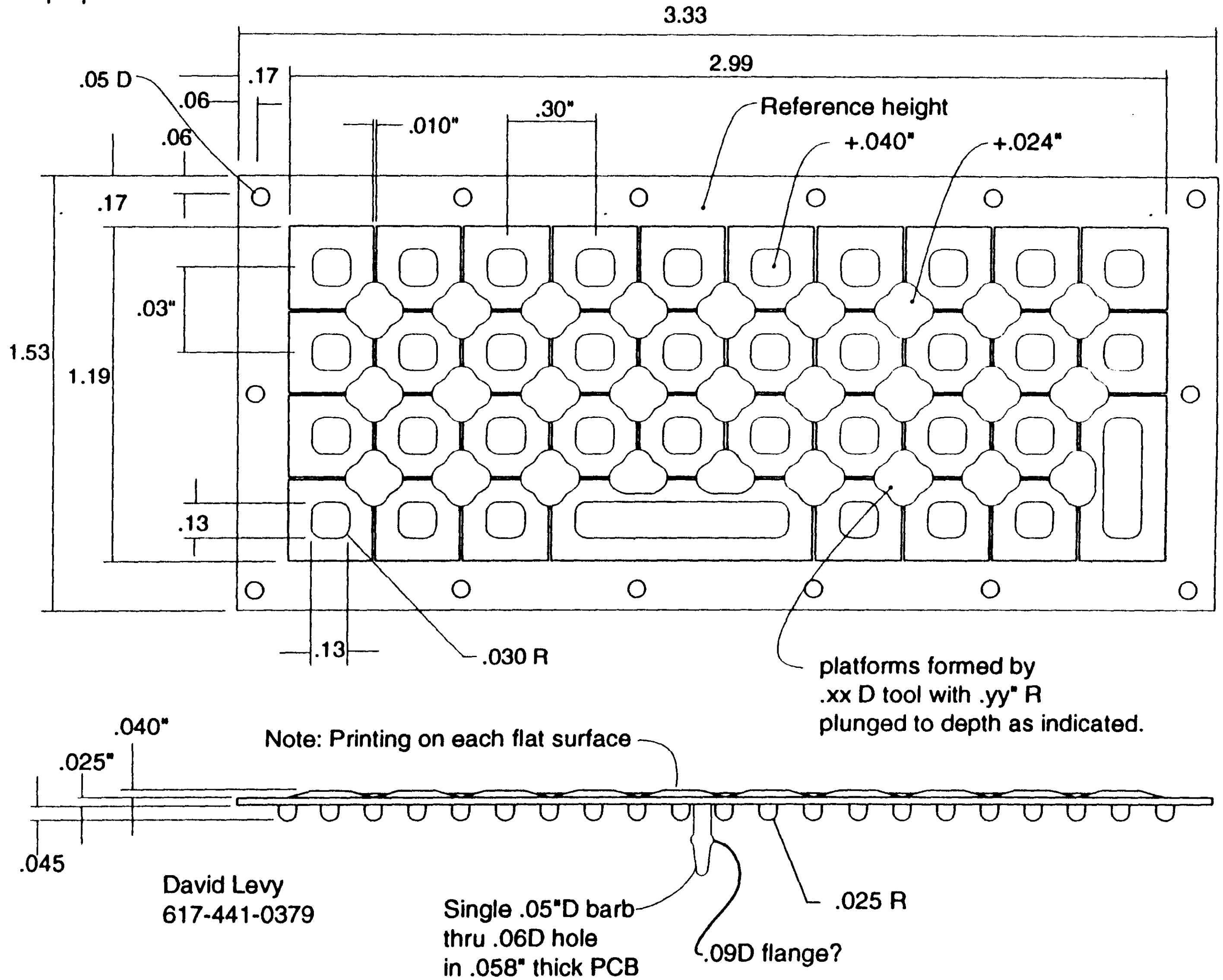
further including a central legend means located on said center portion of each of said key caps.

* * * * *

## 9.5 Appendix - Elastomer Drawings

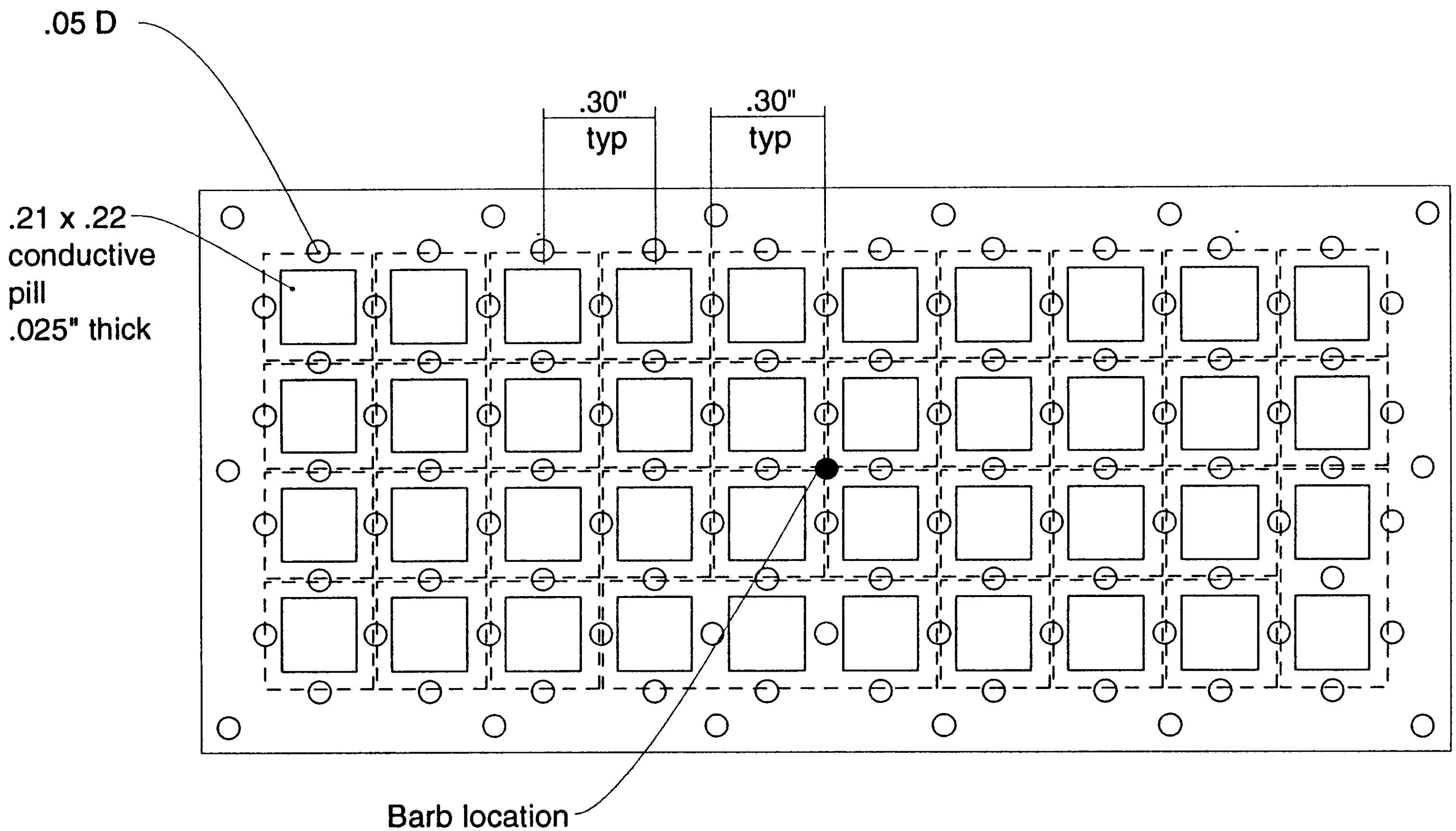
Preliminary
for discussion purposes

# Keypad Elastomer - Version 1/97





David Levy
617-441-0379


.05 D
.30"
typ
.30"
typ
.21 x .22
conductive
pill
.025" thick
Barb location

Back View

## 9.6 Appendix - SMEK Code

The code of the first SMEK prototype, written by consultant.

```
include "picreg84.equ"          ; default EQUates for the PIC registers

        list c=132,E=2,N=60,P=16C84

; keypad scanner.
;
; 16C84 is connected to a 3.6864MHz xtal, yielding a 921.6KHz
; I-clk.  So, each I-clk lasts 1.085us.
;

TMR0        equ     RTCCR
PCLATH              equ     0ah
PCL         equ     02h
INTCON              equ     0bh

KEYVARS             ORG     0ch     ; origin for data (16C5x & 16C84)
k0          RES     01h     ; 0c    \
k1          RES     01h     ; 0d     \ key
k2          RES     01h     ; 0e     / slots
k3          RES     01h     ; 0f    /
            RES     01h     ; 10
curRow              RES     01h     ; 11    scanner state
curCol      RES     01h     ; 12
temp        RES     01h     ; 13
XmtReg              RES     01h     ; 14
XmtCnt              RES     01h     ; 15
lastk0      RES     01h     ; 16
lastk1      RES     01h     ; 17
lastk2      RES     01h     ; 18
lastk3      RES     01h     ; 19
dumpTemp    RES     01h     ; 1a
sideTemp    RES     01h     ; 1b
```

```
waitAllUp   RES   01h   ; 1c
diagTemp    RES   01h   ; 1d
diagColTemp RES   01h   ; 1e
colMatch    RES   01h   ; 1f
                        ;       16C84-only
hexTemp           RES   01h     ; 20
lastRow           RES   01h     ; 21
lastCol     RES   01h   ; 22
            RES   01h   ; 23
            RES   01h   ; 24
            RES   01h   ; 25
            RES   01h   ; 26
            RES   01h   ; 27
            RES   01h   ; 28
            RES   01h   ; 29
            RES   01h   ; 2a
            RES   01h   ; 2b
            RES   01h   ; 2c
            RES   01h   ; 2d
            RES   01h   ; 2e
            RES   01h   ; 2f


;
; Port A:     4 ---------- Serial to LCD controller
;           3 2 ------ Row select
;               1 0 --------------------\ Columns
; Port B:           7 6 5 4 3 2 1 0 --/ 10 - 1
;

SERIAL_BIT  equ   4

ROW_0             equ   3
ROW_1             equ   2

COL_0             EQU   7       ; PORT_B
COL_1             EQU   6
COL_2             EQU   5
COL_3             EQU   4
COL_4             EQU   3
COL_5             EQU   2
```

```
COL_6           EQU  1
COL_7           EQU  0
COL_8           EQU  0      ; PORT_A
COL_9           EQU  1


        ORG   00h

;
==================================================================
=
; Reset Entry!
;

Start
        goto    Init

; Lookup table for singleWay keys.
; W holds kslot entry (xxxxyytt).

singleLookup
        andlw 0fch              ; strip off scancnt
        movwf temp
        bcf     STATUS,C
        rrf     temp
        rrf     temp,W                  ; W = 00xxxxyy

        ; 6 bit offset, so 64 entries...BUT, we know that the
        ; column will never be > 10, so only 44 entries are actually
        ; here.
        ; column numbers start with 1, not 0
        ; row numbers start with 0

        addwf PCL

        retlw   0               ; 00 0000 00   *UNUSED*
        retlw   0               ; 00 0000 01   *UNUSED*
        retlw   0               ; 00 0000 10   *UNUSED*
        retlw   0               ; 00 0000 11   *UNUSED*

        retlw   '1'             ; 00 0001 00   col 1
```

```
retlw   '@'     ; 00 0001 01
retlw   '/'     ; 00 0001 10
retlw   80h     ; 00 0001 11   -ON-

retlw   '2'     ; 00 0010 00   col 2
retlw   '&'     ; 00 0010 01
retlw   '+'     ; 00 0010 10
retlw   81h     ; 00 0010 11   -FUNC(L)-

retlw   '3'     ; 00 0011 00   col 3
retlw   '$'     ; 00 0011 01
retlw   '-'     ; 00 0011 10
retlw   81h     ; 00 0011 11   -FUNC(R)-

retlw   '4'     ; 00 0100 00   col 4
retlw   '#'     ; 00 0100 01
retlw   '*'     ; 00 0100 10
retlw   ' '     ; 00 0100 11

retlw   '5'     ; 00 0101 00   col 5
retlw   '%'     ; 00 0101 01
retlw   '='     ; 00 0101 10
retlw   ' '     ; 00 0101 11

retlw   '6'     ; 00 0110 00   col 6
retlw   '<'     ; 00 0110 01
retlw   '"'     ; 00 0110 10
retlw   ' '     ; 00 0110 11

retlw   '7'     ; 00 0111 00   col 7
retlw   '>'     ; 00 0111 01
retlw   27h     ; 00 0111 10   single quote
retlw   82h     ; 00 0111 11   -LIST(L)-

retlw   '8'     ; 00 1000 00   col 8
retlw   '('     ; 00 1000 01
retlw   ':'     ; 00 1000 10
retlw   82h     ; 00 1000 11   -LIST(R)-

retlw   '9'     ; 00 1001 00   col 9
```

```
        retlw   ')'             ; 00 1001 01
        retlw   '.'             ; 00 1001 10
        retlw   83h             ; 00 1001 11   -SEND(L)-

        retlw   '0'             ; 00 1010 00   col 10
        retlw   'P'             ; 00 1010 01
        retlw   ','             ; 00 1010 10
        retlw   83h             ; 00 1010 11   -SEND(R)-


; Lookup tables for multi keys.
; W holds kslot entry (xxxxyytt).

upLeftLookup
        andlw 0fch              ; strip off scancnt
        movwf temp
        bcf     STATUS,C
        rrf     temp
        rrf     temp,W                  ; W = 00xxxxyy

        ; 6 bit offset, so 64 entries...BUT, we know that the
        ; column will never be > 10, so only 44 entries are actually
        ; here.
        ; column numbers start with 1, not 0
        ; row numbers start with 0

        ; UP/LEFT, so no: ROW 0
        ;               COL 1

        addwf PCL

        retlw   0               ; 00 0000 00   *UNUSED*
        retlw   0               ; 00 0000 01   *UNUSED*
        retlw   0               ; 00 0000 10   *UNUSED*
        retlw   0               ; 00 0000 11   *UNUSED*

        retlw   0               ; 00 0001 00   col 1
        retlw   0               ; 00 0001 01
        retlw   0               ; 00 0001 10
        retlw   0               ; 00 0001 11   -ON-
```

```
retlw  0      ; 00 0010 00   col 2
retlw  'Q'    ; 00 0010 01
retlw  'A'    ; 00 0010 10
retlw  'Z'    ; 00 0010 11   -FUNC(L)-

retlw  0      ; 00 0011 00   col 3
retlw  'W'    ; 00 0011 01
retlw  'S'    ; 00 0011 10
retlw  'X'    ; 00 0011 11   -FUNC(R)-

retlw  0      ; 00 0100 00   col 4
retlw  'E'    ; 00 0100 01
retlw  'D'    ; 00 0100 10
retlw  'C'    ; 00 0100 11

retlw  0      ; 00 0101 00   col 5
retlw  'R'    ; 00 0101 01
retlw  'F'    ; 00 0101 10
retlw  'V'    ; 00 0101 11

retlw  0      ; 00 0110 00   col 6
retlw  'T'    ; 00 0110 01
retlw  'G'    ; 00 0110 10
retlw  'B'    ; 00 0110 11

retlw  0      ; 00 0111 00   col 7
retlw  'Y'    ; 00 0111 01
retlw  'H'    ; 00 0111 10
retlw  'N'    ; 00 0111 11   -LIST(L)-

retlw  0      ; 00 1000 00   col 8
retlw  'U'    ; 00 1000 01
retlw  'J'    ; 00 1000 10
retlw  'M'    ; 00 1000 11   -LIST(R)-

retlw  0      ; 00 1001 00   col 9
retlw  'I'    ; 00 1001 01
retlw  'K'    ; 00 1001 10
retlw  '!'    ; 00 1001 11   -SEND(L)-
```

```
        retlw   0               ; 00 1010 00   col 10
        retlw   'O'             ; 00 1010 01
        retlw   'L'             ; 00 1010 10
        retlw   '?'             ; 00 1010 11   -SEND(R)-


downRightLookup
        andlw 0fch              ; strip off scancnt
        movwf temp
        bcf     STATUS,C
        rrf     temp
        rrf     temp,W                  ; W = 00xxxxyy

        ; 6 bit offset, so 64 entries...BUT, we know that the
        ; column will never be > 10, so only 44 entries are actually
        ; here.
        ; column numbers start with 1, not 0
        ; row numbers start with 0

        ; DOWN/RIGHT, so no: ROW 3
        ;                    COL 10

        addwf PCL

        retlw   0               ; 00 0000 00   *UNUSED*
        retlw   0               ; 00 0000 01   *UNUSED*
        retlw   0               ; 00 0000 10   *UNUSED*
        retlw   0               ; 00 0000 11   *UNUSED*

        retlw   'Q'             ; 00 0001 00   col 1
        retlw   'A'             ; 00 0001 01
        retlw   'Z'             ; 00 0001 10
        retlw   0               ; 00 0001 11   -ON-

        retlw   'W'             ; 00 0010 00   col 2
        retlw   'S'             ; 00 0010 01
        retlw   'X'             ; 00 0010 10
        retlw   0               ; 00 0010 11   -FUNC(L)-
```

```
retlw   'E'     ; 00 0011 00   col 3
retlw   'D'     ; 00 0011 01
retlw   'C'     ; 00 0011 10
retlw   0       ; 00 0011 11   -FUNC(R)-

retlw   'R'     ; 00 0100 00   col 4
retlw   'F'     ; 00 0100 01
retlw   'V'     ; 00 0100 10
retlw   0       ; 00 0100 11

retlw   'T'     ; 00 0101 00   col 5
retlw   'G'     ; 00 0101 01
retlw   'B'     ; 00 0101 10
retlw   0       ; 00 0101 11

retlw   'Y'     ; 00 0110 00   col 6
retlw   'H'     ; 00 0110 01
retlw   'N'     ; 00 0110 10
retlw   0       ; 00 0110 11

retlw   'U'     ; 00 0111 00   col 7
retlw   'J'     ; 00 0111 01
retlw   'M'     ; 00 0111 10
retlw   0       ; 00 0111 11   -LIST(L)-

retlw   'I'     ; 00 1000 00   col 8
retlw   'K'     ; 00 1000 01
retlw   '!'     ; 00 1000 10
retlw   0       ; 00 1000 11   -LIST(R)-

retlw   'O'     ; 00 1001 00   col 9
retlw   'L'     ; 00 1001 01
retlw   '?'     ; 00 1001 10
retlw   0       ; 00 1001 11   -SEND(L)-

retlw   0       ; 00 1010 00   col 10
retlw   0       ; 00 1010 01
retlw   0       ; 00 1010 10
retlw   0       ; 00 1010 11   -SEND(R)-
```

```
upRightLookup
        andlw  0fch            ; strip off scancnt
        movwf  temp
        bcf    STATUS,C
        rrf    temp
        rrf    temp,W                ; W = 00xxxxyy

        ; 6 bit offset, so 64 entries...BUT, we know that the
        ; column will never be > 10, so only 44 entries are actually
        ; here.
        ; column numbers start with 1, not 0
        ; row numbers start with 0

        ; UP/RIGHT, so no: ROW 0
        ;               COL 10

        addwf PCL

        retlw  0               ; 00 0000 00   *UNUSED*
        retlw  0               ; 00 0000 01   *UNUSED*
        retlw  0               ; 00 0000 10   *UNUSED*
        retlw  0               ; 00 0000 11   *UNUSED*

        retlw  0               ; 00 0001 00   col 1
        retlw  'Q'             ; 00 0001 01
        retlw  'A'             ; 00 0001 10
        retlw  'Z'             ; 00 0001 11   -ON-

        retlw  0               ; 00 0010 00   col 2
        retlw  'W'             ; 00 0010 01
        retlw  'S'             ; 00 0010 10
        retlw  'X'             ; 00 0010 11   -FUNC(L)-

        retlw  0               ; 00 0011 00   col 3
        retlw  'E'             ; 00 0011 01
        retlw  'D'             ; 00 0011 10
        retlw  'C'             ; 00 0011 11   -FUNC(R)-
```

```
        retlw  0            ; 00 0100 00   col 4
        retlw  'R'          ; 00 0100 01
        retlw  'F'          ; 00 0100 10
        retlw  'V'          ; 00 0100 11

        retlw  0            ; 00 0101 00   col 5
        retlw  'T'          ; 00 0101 01
        retlw  'G'          ; 00 0101 10
        retlw  'B'          ; 00 0101 11

        retlw  0            ; 00 0110 00   col 6
        retlw  'Y'          ; 00 0110 01
        retlw  'H'          ; 00 0110 10
        retlw  'N'          ; 00 0110 11

        retlw  0            ; 00 0111 00   col 7
        retlw  'U'          ; 00 0111 01
        retlw  'J'          ; 00 0111 10
        retlw  'M'          ; 00 0111 11   -LIST(L)-

        retlw  0            ; 00 1000 00   col 8
        retlw  'I'          ; 00 1000 01
        retlw  'K'          ; 00 1000 10
        retlw  '!'          ; 00 1000 11   -LIST(R)-

        retlw  0            ; 00 1001 00   col 9
        retlw  'O'          ; 00 1001 01
        retlw  'L'          ; 00 1001 10
        retlw  '?'          ; 00 1001 11   -SEND(L)-

        retlw  0            ; 00 1010 00   col 10
        retlw  0            ; 00 1010 01
        retlw  0            ; 00 1010 10
        retlw  0            ; 00 1010 11   -SEND(R)-


downLeftLookup
        andlw  0fch         ; strip off scancnt
        movwf  temp
        bcf    STATUS,C
```

```
rrf     temp
rrf     temp,W              ; W = 00xxxxyy

; 6 bit offset, so 64 entries...BUT, we know that the
; column will never be > 10, so only 44 entries are actually
; here.
; column numbers start with 1, not 0
; row numbers start with 0

; DOWN/LEFT, so no: ROW 3
;               COL 1

addwf PCL

retlw   0           ; 00 0000 00   *UNUSED*
retlw   0           ; 00 0000 01   *UNUSED*
retlw   0           ; 00 0000 10   *UNUSED*
retlw   0           ; 00 0000 11   *UNUSED*

retlw   0           ; 00 0001 00   col 1
retlw   0           ; 00 0001 01
retlw   0           ; 00 0001 10
retlw   0           ; 00 0001 11   -ON-

retlw   'Q'         ; 00 0010 00   col 2
retlw   'A'         ; 00 0010 01
retlw   'Z'         ; 00 0010 10
retlw   0           ; 00 0010 11   -FUNC(L)-

retlw   'W'         ; 00 0011 00   col 3
retlw   'S'         ; 00 0011 01
retlw   'X'         ; 00 0011 10
retlw   0           ; 00 0011 11   -FUNC(R)-

retlw   'E'         ; 00 0100 00   col 4
retlw   'D'         ; 00 0100 01
retlw   'C'         ; 00 0100 10
retlw   0           ; 00 0100 11

retlw   'R'         ; 00 0101 00   col 5
```

```
        retlw   'F'             ; 00 0101 01
        retlw   'V'             ; 00 0101 10
        retlw   0               ; 00 0101 11

        retlw   'T'             ; 00 0110 00   col 6
        retlw   'G'             ; 00 0110 01
        retlw   'B'             ; 00 0110 10
        retlw   0               ; 00 0110 11

        retlw   'Y'             ; 00 0111 00   col 7
        retlw   'H'             ; 00 0111 01
        retlw   'N'             ; 00 0111 10
        retlw   0               ; 00 0111 11   -LIST(L)-

        retlw   'U'             ; 00 1000 00   col 8
        retlw   'J'             ; 00 1000 01
        retlw   'M'             ; 00 1000 10
        retlw   0               ; 00 1000 11   -LIST(R)-

        retlw   'I'             ; 00 1001 00   col 9
        retlw   'K'             ; 00 1001 01
        retlw   '!'             ; 00 1001 10
        retlw   0               ; 00 1001 11   -SEND(L)-

        retlw   'O'             ; 00 1010 00   col 10
        retlw   'L'             ; 00 1010 01
        retlw   '?'             ; 00 1010 10
        retlw   0               ; 00 1010 11   -SEND(R)-


Init
        clrf    INTCON

        movlw b'11111111'       ; B is all col inputs
        tris    PORT_B

        movlw b'00000011'       ; 0&1 are col inputs, 2&3 are row sel,
        tris    PORT_A                  ; 4 is ser output
```

```
        movlw 10h           ; ser line hi, row sel 00
        movwf PORT_A

        movlw b'10000000'   ; 1:2 prescale, TMR0 clocked by I clk
        option              ; weak pullups DISABLED

        clrf    lastRow
        clrf    lastCol

        clrf    k0
        clrf    k1
        clrf    k2
        clrf    k3

        clrf    lastk0
        clrf    lastk1
        clrf    lastk2
        clrf    lastk3

        clrf    waitAllUp

;
; MAIN LOOP
;

scanKeypad
        movlw 10h           ; ser line hi, row sel 00
        movwf PORT_A                ; drive R0 hi
        movwf curRow                ; row cnt in bits 3:2

        call    scanColumns

        movlw 14h
        movwf PORT_A                ; drive R1 hi
        movwf curRow

        call    scanColumns

        movlw 18h
```

```
        movwf PORT_A            ; drive R2 hi
        movwf curRow

        call    scanColumns

        movlw 1ch
        movwf PORT_A            ; drive R3 hi
        movwf curRow

        call    scanColumns

;
; Now we have to evaluate the 4 kslots.
;
; Are all kslots zero?
;       YES-> clear waitAllUp
;       NO--> leave waitAllUp alone
;
; Are there any kslots of age 3?
;       NO--> Done.
;       YES-> Any Diagonal neighbor keys down?
;               YES-> Send multiKey, set waitAllUp, Done.
;               NO--> Any Side neighbor keys down?
;                       YES-> Do nothing now, let neighbors mature.
;                       NO--> Send singleKey, set waitAllUp, Done.
;
; If a kslot hasn't changed since the last scan, zero it.
;

; NOTE:  The "wait for all keys up" mechanism isn't quite right,
;     but it's probably close enough.  It will actually let keys
;        repeat.

evalKeySlots
        movf    k0,W            ; if all kslots are zero, clear waitAllUp
        btfss   STATUS,Z
        goto    upWaitCheck
        movf    k1,W
        btfss   STATUS,Z
        goto    upWaitCheck
```

```
        movf    k2,W
        btfss   STATUS,Z
        goto    upWaitCheck
        movf    k3,W
        btfss   STATUS,Z
        goto    upWaitCheck

        clrf    waitAllUp      ; all zero, so all must be up!

upWaitCheck

        ;
        ; ============== k0 ================
        ;
k0Age3Check
        movf    k0,W
        xorwf   lastk0,W       ; any change since last scan?
        btfsc   STATUS,Z       ; if Z, this entry is stale
        clrf    k0             ; clear it + last & go check k1
        movf    k0,W
        movwf   lastk0         ; update last
        andlw   3              ; mask all but cnt
        xorlw   3              ; both set?
        btfss   STATUS,Z       ; if Z, age == 3
        goto    k1Age3Check
                               ; it is 3, it is not stale.
        movf    waitAllUp,W    ; are we waiting for all up?
        btfss   STATUS,Z
        goto    k1Age3Check
                               ; look for diagonals.
        movf    k0,W
        movwf   diagTemp
        call    diagCheck
        btfss   STATUS,C       ; if there were diags, the appropriate char
        goto    k0SingleSend   ;  will have been sent, and C will be set

        clrf    k0
        clrf    lastk0

        incf    waitAllUp      ; set wait flag
```

```
        goto    waitAfterScan

k0SingleSend
        movf    waitAllUp,W ; are we waiting for all up?
        btfss   STATUS,Z
        goto    k1Age3Check

        movf    k0,W
        movwf   sideTemp
        call    sideCheck
        btfsc   STATUS,C    ; if a side key was down, C will be set
        goto    k1Age3Check ; a side key was down, so leave this one alone

        movf    k0,W
        clrf    k0
        clrf    lastk0
        call    singleLookup
        call    serXmtW

        incf    waitAllUp   ; set wait flag
        goto    waitAfterScan

        ;
        ; =============== k1 =================
        ;
k1Age3Check
        movf    k1,W
        xorwf   lastk1,W    ; any change since last scan?
        btfsc   STATUS,Z    ; if Z, this entry is stale
        clrf    k1          ; clear it + last & go check k2
        movf    k1,W
        movwf   lastk1      ; update last
        andlw   3           ; mask all but cnt
        xorlw   3           ; both set?
        btfss   STATUS,Z    ; if Z, age == 3
        goto    k2Age3Check
                            ; it is 3, it is not stale.
        movf    waitAllUp,W ; are we waiting for all up?
        btfss   STATUS,Z
        goto    k2Age3Check
```

```
                        ; look for diagonals.
        movf   k1,W
        movwf  diagTemp
        call   diagCheck
        btfss  STATUS,C      ; if there were diags, the appropriate char
        goto   k1SingleSend  ;  will have been sent, and C will be set

        clrf   k1
        clrf   lastk1

        incf   waitAllUp     ; set wait flag
        goto   waitAfterScan

k1SingleSend
        movf   waitAllUp,W   ; are we waiting for all up?
        btfss  STATUS,Z
        goto   k2Age3Check

        movf   k1,W
        movwf  sideTemp
        call   sideCheck
        btfsc  STATUS,C      ; if a side key was down, C will be set
        goto   k2Age3Check   ; a side key was down, so leave this one alone

        movf   k1,W
        clrf   k1
        clrf   lastk1
        call   singleLookup
        call   serXmtW

        incf   waitAllUp     ; set wait flag
        goto   waitAfterScan

        ;
        ; =============== k2 =================
        ;
k2Age3Check
        movf   k2,W
        xorwf  lastk2,W      ; any change since last scan?
        btfsc  STATUS,Z      ; if Z, this entry is stale
```

```
        clrf    k2              ; clear it + last & go check k3
        movf    k2,W
        movwf   lastk2          ; update last
        andlw   3               ; mask all but cnt
        xorlw   3               ; both set?
        btfss   STATUS,Z        ; if Z, age == 3
        goto    k3Age3Check
                                ; it is 3, it is not stale.
        movf    waitAllUp,W     ; are we waiting for all up?
        btfss   STATUS,Z
        goto    k3Age3Check
                                ; look for diagonals.
        movf    k2,W
        movwf   diagTemp
        call    diagCheck
        btfss   STATUS,C        ; if there were diags, the appropriate char
        goto    k2SingleSend    ;  will have been sent, and C will be set

        clrf    k2
        clrf    lastk2

        incf    waitAllUp       ; set wait flag
        goto    waitAfterScan

k2SingleSend
        movf    waitAllUp,W     ; are we waiting for all up?
        btfss   STATUS,Z
        goto    k3Age3Check

        movf    k2,W
        movwf   sideTemp
        call    sideCheck
        btfsc   STATUS,C        ; if a side key was down, C will be set
        goto    k3Age3Check     ; a side key was down, so leave this one alone

        movf    k2,W
        clrf    k2
        clrf    lastk2
        call    singleLookup
        call    serXmtW
```

```
        incf    waitAllUp      ; set wait flag
        goto    waitAfterScan

        ;
        ; =============== k3 =================
        ;
k3Age3Check
        movf    k3,W
        xorwf   lastk3,W       ; any change since last scan?
        btfsc   STATUS,Z       ; if Z, this entry is stale
        clrf    k3             ; clear it + last & that is all!
        movf    k3,W
        movwf   lastk3         ; update last
        andlw   3              ; mask all but cnt
        xorlw   3              ; both set?
        btfss   STATUS,Z       ; if Z, age == 3
        goto    waitAfterScan; OK, that is all!
                               ; it is 3, it is not stale.
        movf    waitAllUp,W ; are we waiting for all up?
        btfss   STATUS,Z
        goto    waitAfterScan
                               ; look for diagonals.
        movf    k3,W
        movwf   diagTemp
        call    diagCheck
        btfss   STATUS,C       ; if there were diags, the appropriate char
        goto    k3SingleSend   ;  will have been sent, and C will be set

        clrf    k3
        clrf    lastk3

        incf    waitAllUp      ; set wait flag
        goto    waitAfterScan

k3SingleSend
        movf    waitAllUp,W ; are we waiting for all up?
        btfss   STATUS,Z
        goto    waitAfterScan
```

```
        movf    k3,W
        movwf   sideTemp
        call    sideCheck
        btfsc   STATUS,C        ; if a side key was down, C will be set
        goto    waitAfterScan; a side key was down, so leave this one alone

        movf    k3,W
        clrf    k3
        clrf    lastk3
        call    singleLookup
        call    serXmtW

        incf    waitAllUp       ; set wait flag

        ;
        ; ========== Delay until next scan ============
        ;

waitAfterScan                   ; wait for a few ms before scanning again
        movlw b'10000111'       ; 1:256 prescale, TMR0 clocked by I clk
        option                  ; weak pullups DISABLED

        ; with 1:256, TMR0 ticks every 1.085us * 256 = 278us.
        ; to wait 20ms, wait: 20ms/278us = 72 ticks
        ; (could save power here by going to sleep?)

        movlw .128-.90
        movwf TMR0
wait20ms
        btfss   TMR0,7
        goto    wait20ms

        movlw b'10000000'       ; 1:2 prescale, TMR0 clocked by I clk
        option                  ; weak pullups DISABLED

        goto    scanKeypad


; sideTemp holds the xxxxyytt to check.
```

```
; Want to check: Up
;               Down
;               Left
;               Right
;
; If Col = 1,  don't check L
; If Row = 0,  don't check U
; If Col = 10, don't check R
; If Row = 3,  don't check D
;
; If a side key is detected:
;       Set C and return
; Else:
;       Clear C and return

sideCheck
        call    UpCheck
        btfsc   STATUS,C
        goto    sideDone

        call    DownCheck
        btfsc   STATUS,C
        goto    sideDone

        call    LeftCheck
        btfsc   STATUS,C
        goto    sideDone

        call    RightCheck
sideDone
        return



; If Row = 0,  don't check U

UpCheck
        movf    sideTemp,W
        andlw   0ch
        btfsc   STATUS,Z        ; if Z, row = 0
```

```
        goto    noUp

        addlw 0ch               ; decrement row select
        iorlw   10h             ; leave serial line hi
        movwf PORT_A                    ; drive row above key row hi

        movf    sideTemp,W
        andlw 0f0h              ; mask all but col select
        call    matchColumn ; if returns with C set, Up key is down
        return

noUp
        bcf     STATUS,C
        return



; If Row = 3,  don't check D

DownCheck
        movf    sideTemp,W
        andlw 0ch
        xorlw   0ch
        btfsc   STATUS,Z        ; if Z, row = 3
        goto    noDown

        movf    sideTemp,W
        andlw 0ch
        addlw 04h               ; increment row select
        iorlw   10h             ; leave serial line hi
        movwf PORT_A                    ; drive row below key row hi

        movf    sideTemp,W
        andlw 0f0h              ; mask all but col select
        call    matchColumn ; if returns with C set, Down key is down
        return

noDown
        bcf     STATUS,C
        return
```

```
; If Col = 1,  don't check L

LeftCheck
        movf   sideTemp,W
        andlw  0f0h
        xorlw  10h
        btfsc  STATUS,Z     ; if Z, col = 1
        goto   noLeft

        movf   sideTemp,W
        andlw  0ch          ; mask all but row select
        iorlw  10h          ; leave serial line hi
        movwf  PORT_A              ; drive row of key hi

        movf   sideTemp,W
        andlw  0f0h         ; mask all but col select
        addlw  0f0h         ; decrement col select
        call   matchColumn  ; if returns with C set, Left key is down
        return

noLeft
        bcf    STATUS,C
        return


; If Col = 10, don't check R

RightCheck
        movf   sideTemp,W
        andlw  0f0h
        xorlw  0a0h
        btfsc  STATUS,Z     ; if Z, col = 10
        goto   noRight

        movf   sideTemp,W
        andlw  0ch          ; mask all but row select
        iorlw  10h          ; leave serial line hi
```

```
        movwf PORT_A                ; drive row of key hi

        movf   sideTemp,W
        andlw  0f0h          ; mask all but col select
        addlw  010h          ; increment col select
        call   matchColumn   ; if returns with C set, Right key is down
        return

noRight
        bcf    STATUS,C
        return



; diagTemp holds the xxxxyytt to check.
; Want to check: Down & Right
;               Up & Left
;
; If Col = 1,  don't check U/L
; If Row = 0,  don't check U/L
; If Col = 10, don't check D/R
; If Row = 3,  don't check D/R
;
; If a diagonal is detected:
;         1. Send char via appropriate lookup table
;         2. Set C & return
; Else:
;         1. Clear C & return

diagCheck
        call   upLeft
        btfsc  STATUS,C      ; if that sent one, don't bother with D/R
        goto   checkDone

        call   downRight
        btfsc  STATUS,C      ; if that sent one, don't bother with U/R
        goto   checkDone

        call   upRight
        btfsc  STATUS,C      ; if that sent one, don't bother with D/L
```

```
        goto    checkDone

        call    downLeft
checkDone
        return


upLeft
        movf    diagTemp,W
        andlw   0f0h
        xorlw   10h
        btfsc   STATUS,Z        ; if Z, col = 1
        goto    noUpLeft

        movf    diagTemp,W
        andlw   0ch
        btfsc   STATUS,Z        ; if Z, row = 0
        goto    noUpLeft

        movf    diagTemp,W
        andlw   0ch             ; mask all but row select
        addlw   0ch             ; decrement row select
        iorlw   10h             ; leave serial line hi
        movwf   PORT_A                  ; drive row above key row hi

        movwf   lastRow

        movf    diagTemp,W
        andlw   0f0h            ; mask all but col select
        addlw   0f0h            ; decrement col select

        movwf   lastCol

        call    matchColumn ; if returns with C set, upleft key is down
        btfss   STATUS,C
        goto    noUpLeft

        call    dumpSlotsUL

        movf    diagTemp,W
```

```
        call    upLeftLookup
        call    serXmtW
        bsf     STATUS,C
        return

noUpLeft
        bcf     STATUS,C
        return


downRight
        movf    diagTemp,W
        andlw   0f0h
        xorlw   0a0h
        btfsc   STATUS,Z        ; if Z, col = 10
        goto    noDownRight

        movf    diagTemp,W
        andlw   0ch
        xorlw   0ch
        btfsc   STATUS,Z        ; if Z, row = 3
        goto    noDownRight

        movf    diagTemp,W
        andlw   0ch             ; mask all but row select
        addlw   04h             ; increment row select
        iorlw   10h             ; leave serial line hi
        movwf   PORT_A          ; drive row below key row hi

        movwf   lastRow

        movf    diagTemp,W
        andlw   0f0h            ; mask all but col select
        addlw   010h            ; increment col select

        movwf   lastCol

        call    matchColumn ; if returns with C set, downright key is down
        btfss   STATUS,C
        goto    noDownRight
```

```
        call    dumpSlotsDR

        movf    diagTemp,W
        call    downRightLookup
        call    serXmtW
        bsf     STATUS,C
        return

noDownRight
        bcf     STATUS,C
        return


downLeft
        movf    diagTemp,W
        andlw   0f0h
        xorlw   10h
        btfsc   STATUS,Z        ; if Z, col = 1
        goto    noDownLeft

        movf    diagTemp,W
        andlw   0ch
        xorlw   0ch
        btfsc   STATUS,Z        ; if Z, row = 3
        goto    noDownLeft

        movf    diagTemp,W
        andlw   0ch             ; mask all but row select
        addlw   04h             ; increment row select
        iorlw   10h             ; leave serial line hi
        movwf   PORT_A                  ; drive row below key row hi

        movwf   lastRow

        movf    diagTemp,W
        andlw   0f0h            ; mask all but col select
        addlw   0f0h            ; decrement col select

        movwf   lastCol
```

```
        call    matchColumn ; if returns with C set, upleft key is down
        btfss   STATUS,C
        goto    noDownLeft

        call    dumpSlotsDL

        movf    diagTemp,W
        call    downLeftLookup
        call    serXmtW
        bsf     STATUS,C
        return

noDownLeft
        bcf     STATUS,C
        return


upRight
        movf    diagTemp,W
        andlw   0f0h
        xorlw   0a0h
        btfsc   STATUS,Z        ; if Z, col = 10
        goto    noUpRight

        movf    diagTemp,W
        andlw   0ch
        btfsc   STATUS,Z        ; if Z, row = 0
        goto    noUpRight

        movf    diagTemp,W
        andlw   0ch             ; mask all but row select
        addlw   0ch             ; decrement row select
        iorlw   10h             ; leave serial line hi
        movwf   PORT_A                  ; drive row above key row hi

        movwf   lastRow

        movf    diagTemp,W
        andlw   0f0h            ; mask all but col select
```

```
        addlw 010h          ; increment col select

        movwf lastCol

        call    matchColumn ; if returns with C set, downright key is down
        btfss   STATUS,C
        goto    noUpRight

        call    dumpSlotsUR

        movf    diagTemp,W
        call    upRightLookup
        call    serXmtW
        bsf     STATUS,C
        return

noUpRight
        bcf     STATUS,C
        return



matchColumn
        movwf diagColTemp
        bcf     STATUS,C
        rrf     diagColTemp
        rrf     diagColTemp
        rrf     diagColTemp
        rrf     diagColTemp         ; diagColTemp = 0000xxxx

        clrf    colMatch

        clrf    curCol              ; xxxx0000
        incf    curCol              ; curCol is really col+1
        btfsc   PORT_B,COL_0
        call    compareCol

        incf    curCol
        btfsc   PORT_B,COL_1
        call    compareCol
```

```
        incf    curCol
        btfsc   PORT_B,COL_2
        call    compareCol

        incf    curCol
        btfsc   PORT_B,COL_3
        call    compareCol

        incf    curCol
        btfsc   PORT_B,COL_4
        call    compareCol

        incf    curCol
        btfsc   PORT_B,COL_5
        call    compareCol

        incf    curCol
        btfsc   PORT_B,COL_6
        call    compareCol

        incf    curCol
        btfsc   PORT_B,COL_7
        call    compareCol

        incf    curCol
        btfsc   PORT_A,COL_8
        call    compareCol

        incf    curCol
        btfsc   PORT_A,COL_9
        call    compareCol

        bcf     STATUS,C            ; assume no match
        movf    colMatch,W          ; was there a match?
        btfss   STATUS,Z            ; if no Z, match!
        bsf     STATUS,C
        retlw   0

compareCol
```

```
        movf    diagColTemp,W               ; fetch col we're looking for
        xorwf   curCol,W                ; match?
        btfsc   STATUS,Z                ; if Z clear, no match.
        incf    colMatch
        return



; A row has been driven.  Walk the 10 cols and see if any are hi.
; If so, see if the current row/col is already in a kslot.  If so,
; inc that kslot's scan count.  If not, put it in an empty kslot.
; If there are no free kslots, drop it on the floor.
; Note: scan counts are pinned at 3.  Only the kslot eval code
; clears kslots.
;
; curRow is already in bits 3:2.
; curCol is in 3:0 while the cols are scanned, then shifted up 4
; bits when it is time to update a kslot.
;
; kslot format:   7 6 5 4 3 2 1 0
;                 \     / \ / \ /
;                  col+1  row cnt

scanColumns
        clrf    curCol                  ; xxxx0000
        incf    curCol                  ; curCol is really col+1
        btfsc   PORT_B,COL_0
        call    keyDown

        incf    curCol
        btfsc   PORT_B,COL_1
        call    keyDown

        incf    curCol
        btfsc   PORT_B,COL_2
        call    keyDown

        incf    curCol
        btfsc   PORT_B,COL_3
        call    keyDown
```

```
        incf    curCol
        btfsc   PORT_B,COL_4
        call    keyDown

        incf    curCol
        btfsc   PORT_B,COL_5
        call    keyDown

        incf    curCol
        btfsc   PORT_B,COL_6
        call    keyDown

        incf    curCol
        btfsc   PORT_B,COL_7
        call    keyDown

        incf    curCol
        btfsc   PORT_A,COL_8
        call    keyDown

        incf    curCol
        btfsc   PORT_A,COL_9
        call    keyDown

        retlw   0


; The key at curRow,curCol was detected down.  Update a kslot
; accordingly.

keyDown
        bcf     STATUS,C
        rlf     curCol,W
        movwf   temp
        rlf     temp
        rlf     temp
        rlf     temp            ; temp = xxxx0000 (xxxx = col)

        movf    curRow,W
```

```
andlw 0ch           ; mask all but row bits
iorwf  temp         ; temp = xxxxyy00 (yy = row)

movf  k0,W
andlw 0fch
xorwf  temp,W
btfsc  STATUS,Z     ; if match, Z = 1
goto   k0Match

movf  k1,W
andlw 0fch
xorwf  temp,W
btfsc  STATUS,Z
goto   k1Match

movf  k2,W
andlw 0fch
xorwf  temp,W
btfsc  STATUS,Z
goto   k2Match

movf  k3,W
andlw 0fch
xorwf  temp,W
btfsc  STATUS,Z
goto   k3Match

; No match!  Look for empty kslot.  If none, bail.

movf  k0,W
btfsc  STATUS,Z
goto   k0Empty

movf  k1,W
btfsc  STATUS,Z
goto   k1Empty

movf  k2,W
btfsc  STATUS,Z
goto   k2Empty
```

```
        movf   k3,W
        btfsc  STATUS,Z
        goto   k3Empty

        retlw  0                ; no free slots!


k0Empty
        movf   temp,W
        movwf  k0
        retlw  0

k1Empty
        movf   temp,W
        movwf  k1
        retlw  0

k2Empty
        movf   temp,W
        movwf  k2
        retlw  0

k3Empty
        movf   temp,W
        movwf  k3
        retlw  0

; There was a match.
; Increment the scan count for that entry, pinning at 3.

k0Match
        movf   k0,W
        andlw  3                ; mask all but count
        xorlw  3                ; both set already?
        btfsc  STATUS,Z
        goto   k0MDone

        incf   k0               ; can safely just bump count
k0MDone
```

```
        retlw   0

k1Match
        movf    k1,W
        andlw   3
        xorlw   3
        btfsc   STATUS,Z
        goto    k1MDone

        incf    k1
k1MDone
        retlw   0

k2Match
        movf    k2,W
        andlw   3
        xorlw   3
        btfsc   STATUS,Z
        goto    k2MDone

        incf    k2
k2MDone
        retlw   0

k3Match
        movf    k3,W
        andlw   3
        xorlw   3
        btfsc   STATUS,Z
        goto    k3MDone

        incf    k3
k3MDone
        retlw   0
```

```
; Xmit byte in XmtReg at 9600,n,8,1.
; At 9600 baud, each bitcell is 104us.
; With a 1:2 prescale on TMR0, TMR0 ticks every 1.085us * 2 = 2.17us.
; 104us/2.17us = 50 TMR0 ticks.
; _______    _________________________
;       |_|0|1|2|3|4|5|6|7|S


serXmtW
        movwf XmtReg
serialXmt
        movlw 9             ; includes start bit
        movwf XmtCnt
        bcf   STATUS,C      ; this will be the start bit

xmtLoop
        movf  curRow,W      ; doesn't affect C
        btfsc STATUS,C      ; 1 or 0 to send?
        goto  xmtOne

xmtZero
        andlw 0fh
        goto  xmtCommon

xmtOne
        iorlw 10h

xmtCommon
        movwf PORT_A

        movlw .128-.39      ; wait 1 bit time
        movwf TMR0
spinBit
        btfss TMR0,7              ; wrap?
        goto  spinBit

        rrf   XmtReg              ; next...
        decfsz XmtCnt
        goto  xmtLoop
```

```
        movf   curRow,W
        movwf  PORT_A               ; be sure to leave with it hi

        movlw  .128-.39     ; wait 1 bit time
        movwf  TMR0
spinStop
        btfss  TMR0,7               ; wrap?
        goto   spinStop

        retlw  0



hexLookup
        movlw  03h
        movwf  PCLATH
        movf   dumpTemp,W
        andlw  0fh
        addwf  PCL

        retlw  '0'
        retlw  '1'
        retlw  '2'
        retlw  '3'
        retlw  '4'
        retlw  '5'
        retlw  '6'
        retlw  '7'
        retlw  '8'
        retlw  '9'
        retlw  'a'
        retlw  'b'
        retlw  'c'
        retlw  'd'
        retlw  'e'
        retlw  'f'


dumpW
```

```
	movwf dumpTemp

	swapf	dumpTemp
	call	hexLookup
	clrf	PCLATH

	call	serXmtW

	swapf	dumpTemp
	call	hexLookup
	clrf	PCLATH

	call	serXmtW

	movlw 0ah
	call	serXmtW
	movlw 0dh
	call	serXmtW

	return


dumpSlotsUL
	return			; comment out for debugging

	movlw 0ah
	call	serXmtW
	movlw 0dh
	call	serXmtW

	movlw 'U'
	call	serXmtW
	movlw 'L'
	call	serXmtW

	movlw '-'
	call	serXmtW
	movf	diagTemp,W
	call	dumpW
```

```
	call	dumpColStuff

	movf	lastRow,W
	call	dumpW
	movf	lastCol,W
	call	dumpW

	movf	k0,W
	call	dumpW
	movf	k1,W
	call	dumpW
	movf	k2,W
	call	dumpW
	movf	k3,W
	call	dumpW

	return


dumpSlotsUR
	return				; comment out for debugging

	movlw 0ah
	call	serXmtW
	movlw 0dh
	call	serXmtW

	movlw 'U'
	call	serXmtW
	movlw 'R'
	call	serXmtW

	movlw '-'
	call	serXmtW
	movf	diagTemp,W
	call	dumpW

	call	dumpColStuff

	movf	lastRow,W
```

```
        call    dumpW
        movf    lastCol,W
        call    dumpW

        movf    k0,W
        call    dumpW
        movf    k1,W
        call    dumpW
        movf    k2,W
        call    dumpW
        movf    k3,W
        call    dumpW

        return


dumpSlotsDL
        return                  ; comment out for debugging

        movlw 0ah
        call    serXmtW
        movlw 0dh
        call    serXmtW

        movlw 'D'
        call    serXmtW
        movlw 'L'
        call    serXmtW

        movlw '-'
        call    serXmtW
        movf    diagTemp,W
        call    dumpW

        call    dumpColStuff

        movf    lastRow,W
        call    dumpW
        movf    lastCol,W
        call    dumpW
```

```
	movf	k0,W
	call	dumpW
	movf	k1,W
	call	dumpW
	movf	k2,W
	call	dumpW
	movf	k3,W
	call	dumpW

	return


dumpSlotsDR
	return			; comment out for debugging

	movlw 0ah
	call	serXmtW
	movlw 0dh
	call	serXmtW

	movlw 'D'
	call	serXmtW
	movlw 'R'
	call	serXmtW

	movlw '-'
	call	serXmtW
	movf	diagTemp,W
	call	dumpW

	call	dumpColStuff

	movf	lastRow,W
	call	dumpW
	movf	lastCol,W
	call	dumpW

	movf	k0,W
	call	dumpW
```

```
        movf    k1,W
        call    dumpW
        movf    k2,W
        call    dumpW
        movf    k3,W
        call    dumpW

        return


dumpColStuff
        movf    colMatch,W
        call    dumpW

        movf    diagColTemp,W
        call    dumpW

        return

END
```