Chapter 4

The Touchback Keyboard Design

The driving point mechanical impedance of a very large class of mechanical systems can be simulated using an ODE solver as the primary computational workhorse. In the following, I describe the construction of a haptic interface controller from an ODE solver. Other approaches will be reviewed to provide perspective.

4.1 Construction of a human-in-the-loop simulator from an ODE solver

The construction of an interactive simula*tor* from an off-line, non-real-time simula*tion* is relatively straight-forward. Those coordinates whose motion is *specified* in the dynamical model are sampled in real-time from the interface hardware rather than being read from an input file or calculated using a pre-defined function. Additionally, response forces from the model simulation which correspond to the same driving point as the specified coordinate are displayed in real-time through the haptic interface simply by commanding those forces to the actuator. This scheme is closely related to that used in flight simulators, which of course have been around for many years. Rather than motion or visual display, however, haptic display is concerned with making apparent the variable which is conjugate to the one sensed by the controller, and displaying that conjugate variable at the same driving point.

This construction can be implemented using an inverse dynamics simulation (forces computed in response to specified kinematic state). However, the forward dynamics simulation may be used if a degree of freedom is introduced in the model at the driving point through the addition of a coupling spring. Indeed, the coupling spring-damper pair which links the driven manipulandum to the virtual key introduced in the previous chapter serves this purpose. The interaction forces may be read as this spring's extension and the specified motion may be appropriately applied to the model by driving the coupled body.

Our haptic interface controller based as described above on an ODE solver may be viewed as a kind of impedance controller, as depicted in Figure 4.1. To obey causality restrictions, the manipulandum must be viewed as an admittance and the human in turn as an impedance operator.

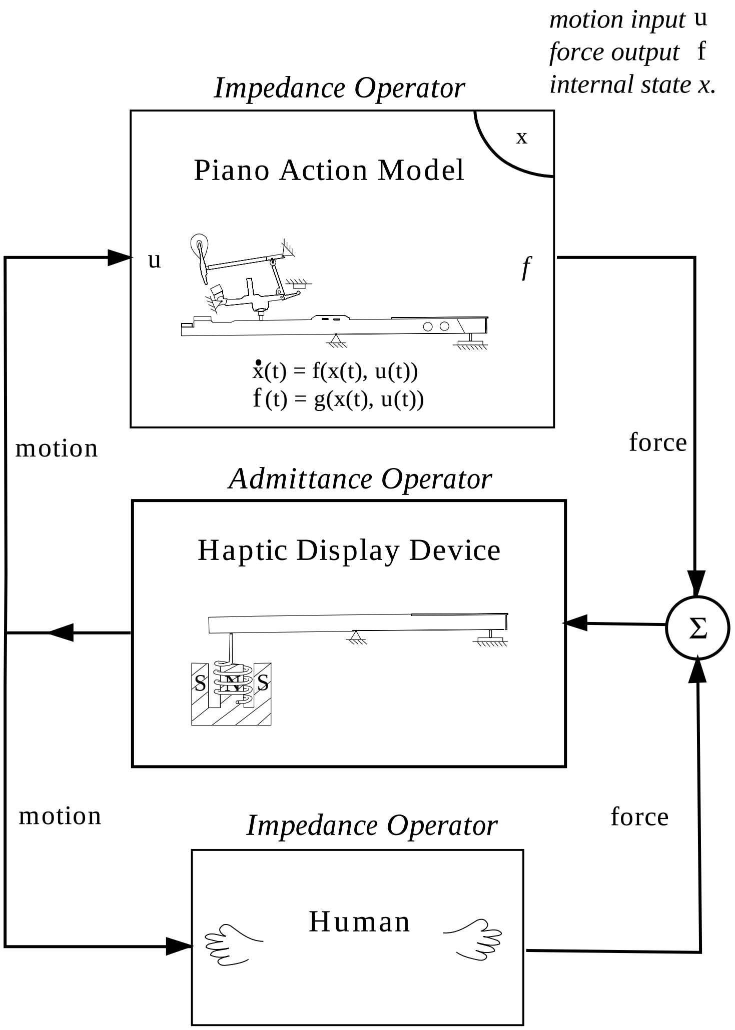

Figure 4.1: *Impedance Display through a Haptic Interface*

The impedance/admittance roles of controller and manipulandum may be reversed. In fact, the forward dynamics simulation may be used directly (without the coupling spring-damper pair) when forces are sensed from the manipulandum and kinematic state is imposed through the actuator. To impose kinematic state with an actuator, however, usually requires that an inner control loop be closed around the manipulandum. PID control can be used on the difference of desired and actual kinematic state, for example. See [64].

I prefer to call these implementations, in which an ODE-solver is involved, impedance *display* and admittance *display* rather than impedance and admittance control. But labels are somewhat hard to apply; it is usedful to relax definitions and explore overlap in display formulations. To further highlight the similarities and overlap, I will describe the haptic display of a simple sprung mass, as in Figure 4.2 using impedance control, impedance display, and admittance display.

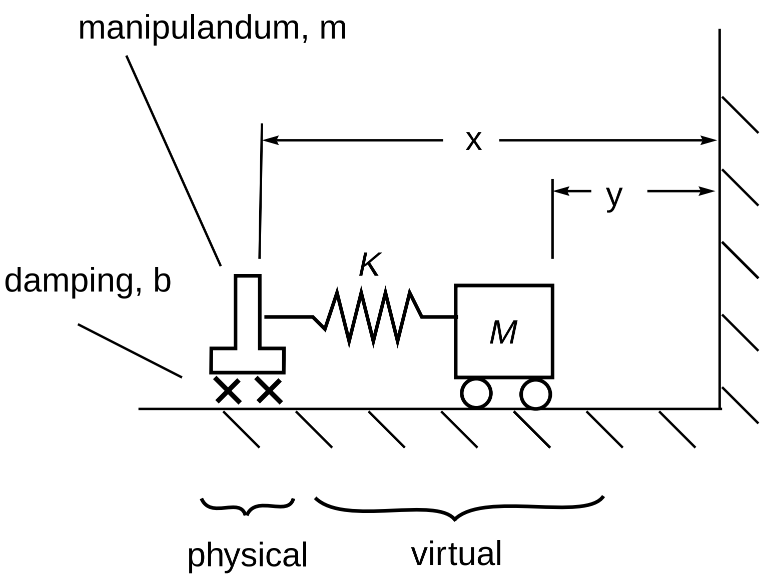

Figure 4.2: *Simple Model for Implementation in impedance and admittance display*

4.1.1 Impedance Control

Figure 4.3 shows the classic block diagram description of an impedance controller for haptic display. This diagram has been presented by Colgate in numerous papers [22], [25] [23]. $C(z)$ is simply a control law, usually $f = Kx_k + Bv_k$, which operates on sampled position x_k (and perhaps velocity v_k —not shown) to produce a force value for imposition on the manipulandum through a sample and hold and an amplifier. The manipulandum is modeled as a physical mass m coupled to ground through viscous damping of damping coefficient b.

It can be seen that impedance control is not capable of displaying the dynamics of a sprung mass. The operator $C(z)$ is memoryless, encoding no dynamics. It has no way to maintain the internal state y, the postion of the virtual mass.

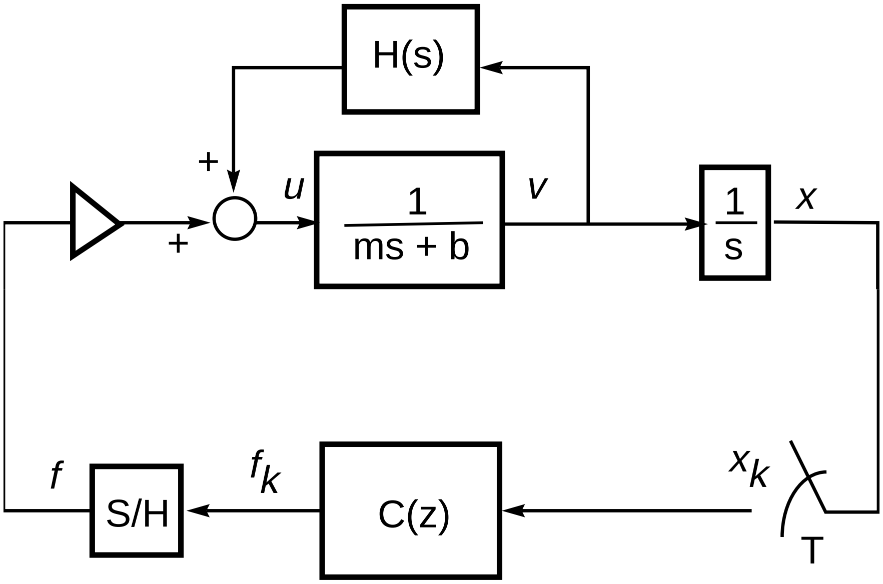

Figure 4.3: *The typical Z-Control Block Diagram*

4.1.2 Impedance Display

Figure 4.4 shows a detailed implementation of impedance display for a sprung mass. The sampled position x_k is differenced with the position y of a virtual mass, maintained by a numerical integration scheme. The basic formula for force display is: $f_k = K * (x_k - y)$. As this law is used to compute output force with each servo cycle, the solution to a second order model is computed with an ODE solver. In the simplest implementation, the state may be advanced through time as a function of input with the Euler method. The Euler method, applied to the equation of motion of a simple mass, $\ddot{y} = f/m$ reads:

$$\begin{aligned} v_{n+1} &= v_n - \tfrac{f_{n+1}}{m}\Delta t \\ y_{n+1} &= y_n + v_{n+1}\Delta t \end{aligned} \tag{4.1}$$

where Δt is the step size or servo period. The spring K in this model corresponds to that degree of freedom which must be added to the model to allow a forward dynamics simulation. The depiction of numerical solution of differential equations with integration blocks in Figure 4.4 is somewhat awkward, but made to draw parallels to the next scheme, admittance display.

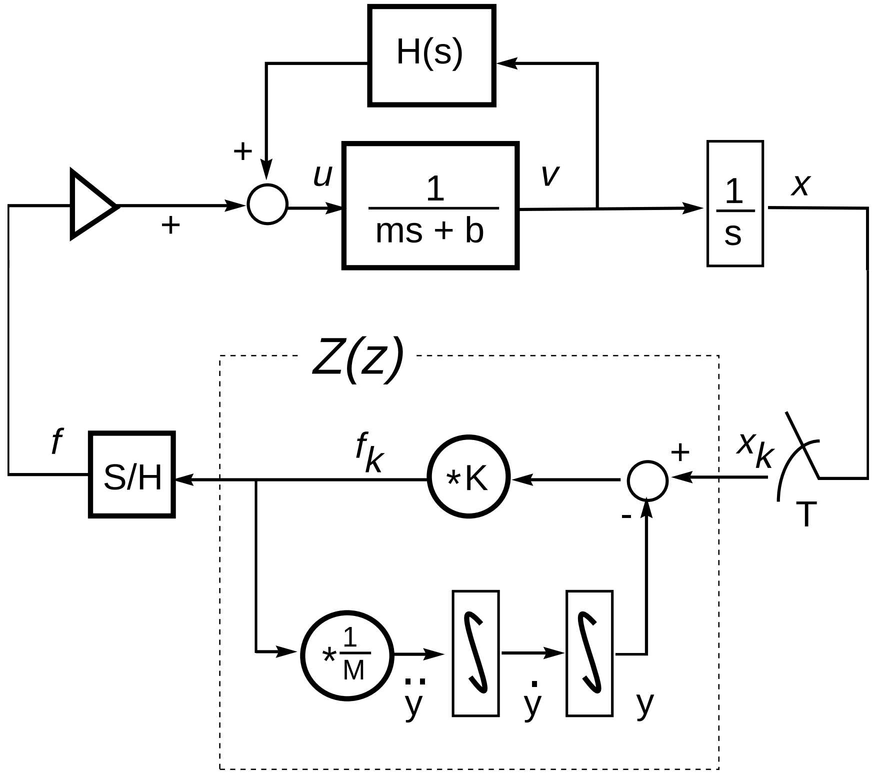

Figure 4.4: *The typical Z-Control (Impedance Display) Block Diagram*

4.1.3 Admittance Display

Figure 4.5 shows an implementation of a mass in admittance display. A force sensor is used to measure the interaction force f_h between manipulandum and human hand. This force is used in the simple constituent equation for a mass, $a = f/m$ and the resulting acceleration is integrated twice numerically to produce the position y. This position y is imposed on the manipulandum with proportional control (of gain K_p) in Figure 4.5. The gain K_p can be interpreted as a spring linking the simulated mass to the manipulandum. More generally, the position y is imposed on the manipulandum with a servo controller, such as a PID controller. See [64].

Note that in the above simple example of admittance control, the two integration blocks may be considered numerical integrators. In the impedance display implementation, the two integration blocks symbolize the numerical solution of a set of second order differential equations.

Boxes have been drawn and labeled Z(z) and Y(z) in Figures 4.4 and 4.5 respectively, to suggest why these schemes are sometimes called Z-control and Y-control. Rather loose interpretations of impedance and admittance are being applied in this case, since sampled position rather than velocity is involved.

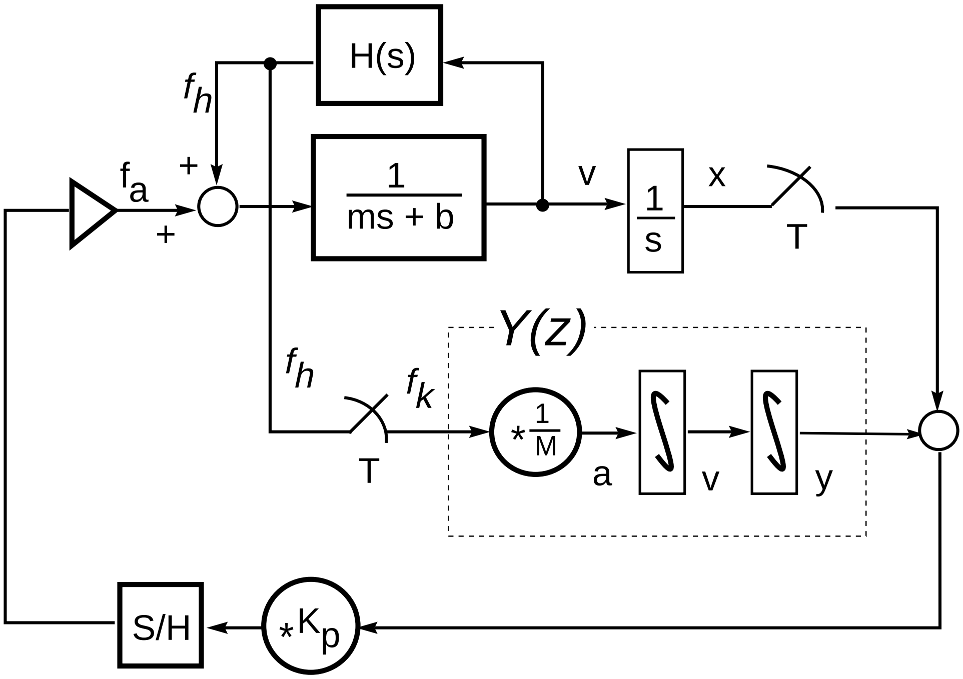

Figure 4.5: *The typical Y-Control (Admittance Display) Block Diagram*

4.1.4 Discussion

Issues to be considered in making a design choice between admittance display and impedance display include required sensors, noise sensitivity of those sensors, extensibility, maintenance, and so on. Supporting the choice between impedance display and admittance display (or impedance and admittance control) are surprisingly few guidelines and a rather small body of literature. The present-day lore is that the choice depends on whether the virtual environment impedance is dominated by inertia (in which case admittance control is recommended) or stiffness (in which case impedance control is recommended). Numerical properties, sensor resolution, and sensor noise must also be considered in these recommendations [64].

4.2 Design of the Touchback Keyboard

In this section, I will present the design of our Touchback Keyboard. The Touchback Keyboard is a haptic interface with a very particular application: re-creation of the mechanical impedance of the grand piano in a synthesizer keyboard. It thus takes on a particular form. The standard piano keyboard becomes is outer façade. It's inner design, however, is vastly different than that of the piano action. Each key is capstan (cable and pulley) driven by a small high quality ironless-core basket-wound motor and position sensored with an optical encoder.

Figure 4.6 shows an assembly drawing of the motor, pulley, motor mounting bracket, drum, and key mount. A highly flexible 0.012 in diameter steel cable with a 7x7x7 winding (not shown) couples the motor pulley to the drum. Note: hidden lines have not been removed in these drawings.

Perhaps the largest engineering challenge faced in this design was one of packing. Rather tight space restrictions inspired the eventual stacked and staggered design. Each key is mounted to a drum at one of seven angles. Figure 4.7 shows an assembly of four motorized keys. The drums are all bearing mounted to the same central shaft, but their radial placements around that shaft take on one of seven angles in a staggered fashion. Likewise, the drive components associated with each drum take on those same seven angles. In the end, all keys point forward and all motors, mounting plates, cables and other components fit neatly without interference. The keymounts rotate through small angles on the central shaft while the motor mounts are held securely in place by a box housing (not shown in the drawings). After seven keys, the arrangement may be repeated, for the width of seven keys is just over the length of one motor.

Our final selection for the mechanical advantage from motor to key was 24:1, which was based on a tradeoff of torque capacity against intertia as discussion below. 512-count per revolution encoders were incorporated with an additional advantage over the motor of 8:1, thus providing over 5,000

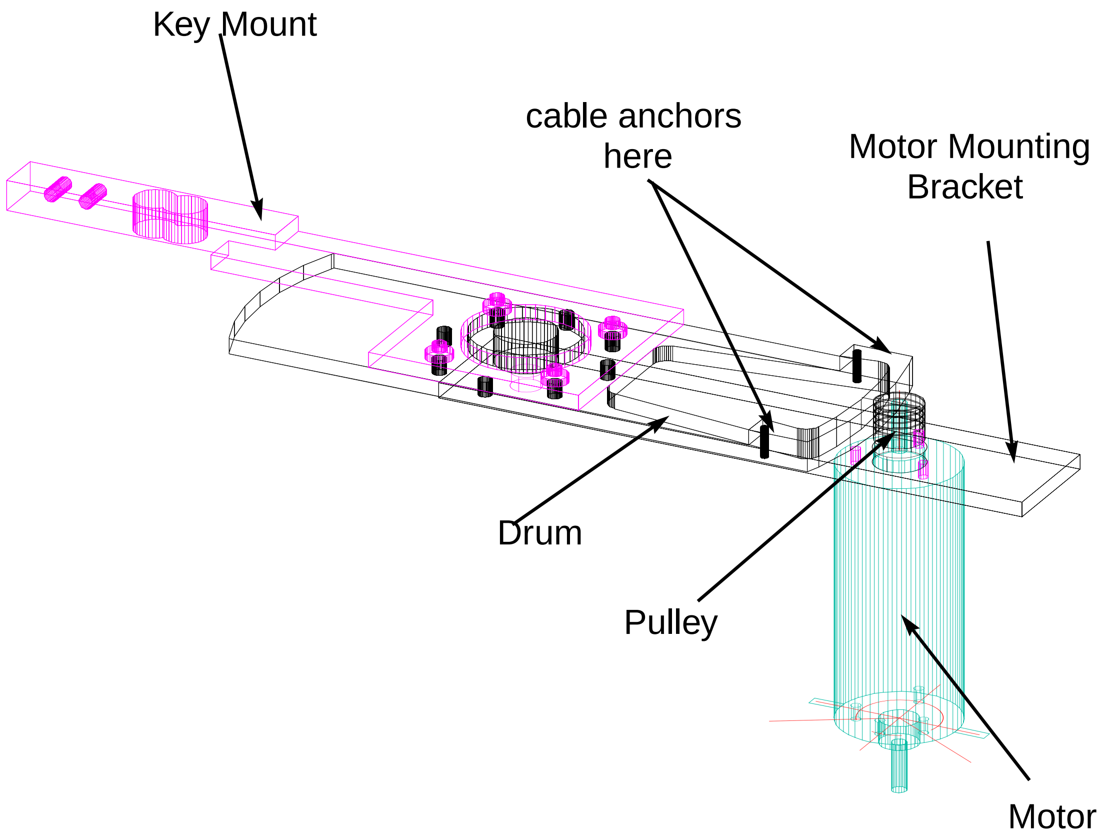

Figure 4.6: *One Key Assembly Drawing*

counts of encoder resolution (with quadrature counting) for the five degrees of key motion. This fine resolution was used to allow numerical differentiation of the position signal in lieu of a tachometer.

We have aimed to create a device which in its unpowered state has the mechanical properties of the key alone but when powered may be made to take on the impedance properties of the full piano action. All elements of the piano action apart from the key are to be rendered through the workings of the motor.

The determination of the target unpowered inertia was made with simple empirical studies on the inertia of the key (a bifilar pendulum experiment). The target force output capabilities were determined using simple models of the piano action (like those presented and reviewed in the previous chapter) and experimental data on piano playing forces available from the literature. The mechanical advantage of the motor was carefully sized to trade off reflected unpowered motor inertia to maximum force output.

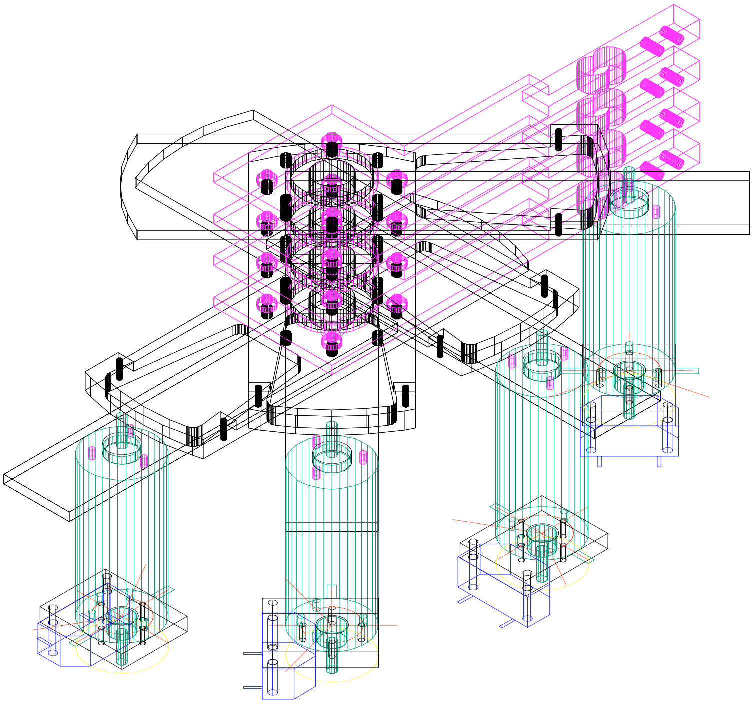

Figure 4.7: *Four-Key Assembly Drawing*

Figures 4.8 through 4.11 highlight the design in photographs. The electrical cables connecting each of the encoders and the motors to the computer are apparent in Figure 4.8. A cubical box with two open sides houses the assembly and secures each motor mount to its appropriate angular position. Figures 4.9 and 4.10 further document the mounting of each motor assembly to the box. Especially in Figure 4.10, looking down the central shaft, one sees that all but one of the eight possible 45-degree staggered positions is occupied by a motor assembly. The one angular position left out is occupied by the plane of keys itself. Figure 4.11 shows a view from above with the top plate of the box removed. The tight packing of all plates and motors can be appreciated in this view. The leftmost key features a mounted straingage and amplifier circuit for force sensing. Each of the key mounts are outfitted with a binocular-type strain concentrator, located at the area of mounting of the plastic key. So far, only the lowest key has been fully instrumented.

bfigureChapter4/IMAGES/Scans/topopenTouchback Keyboard: View with Top Open4.0

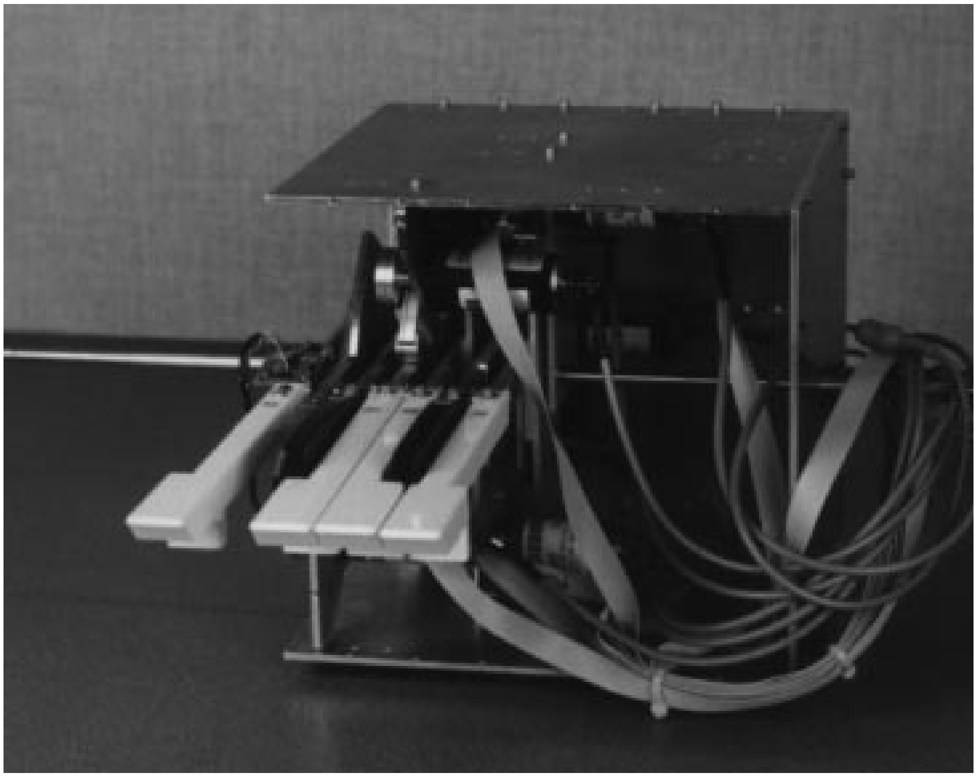

Figure 4.8: *Touchback Keyboard: Front View Showing Cables*

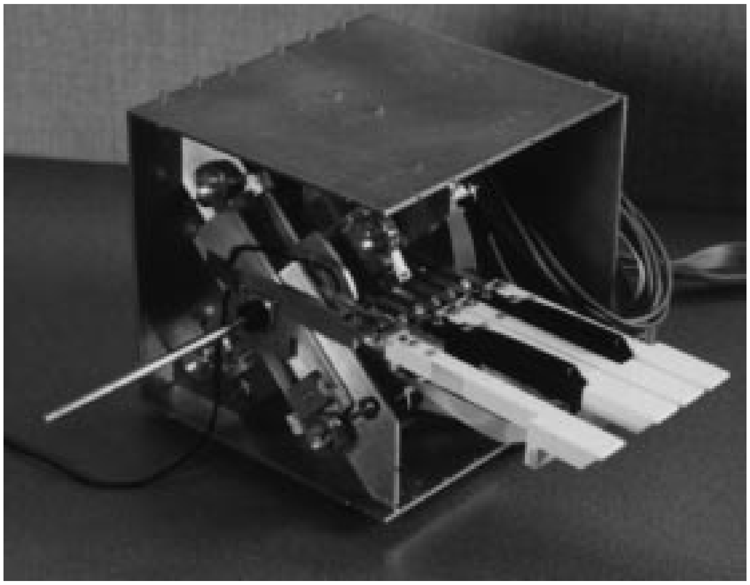

Figure 4.9: *Touchback Keyboard: Isometric View*

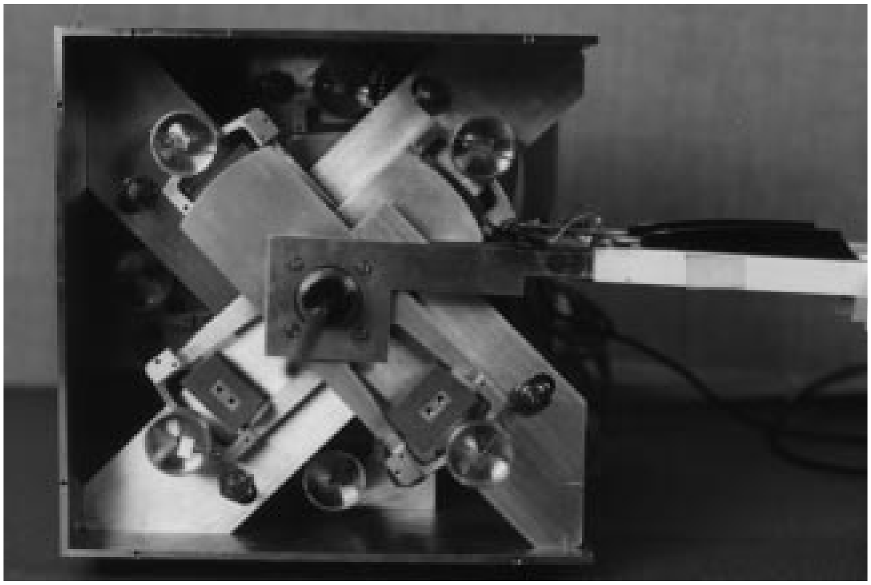

Figure 4.10: *Touchback Keyboard: View Looking Down Skewer*

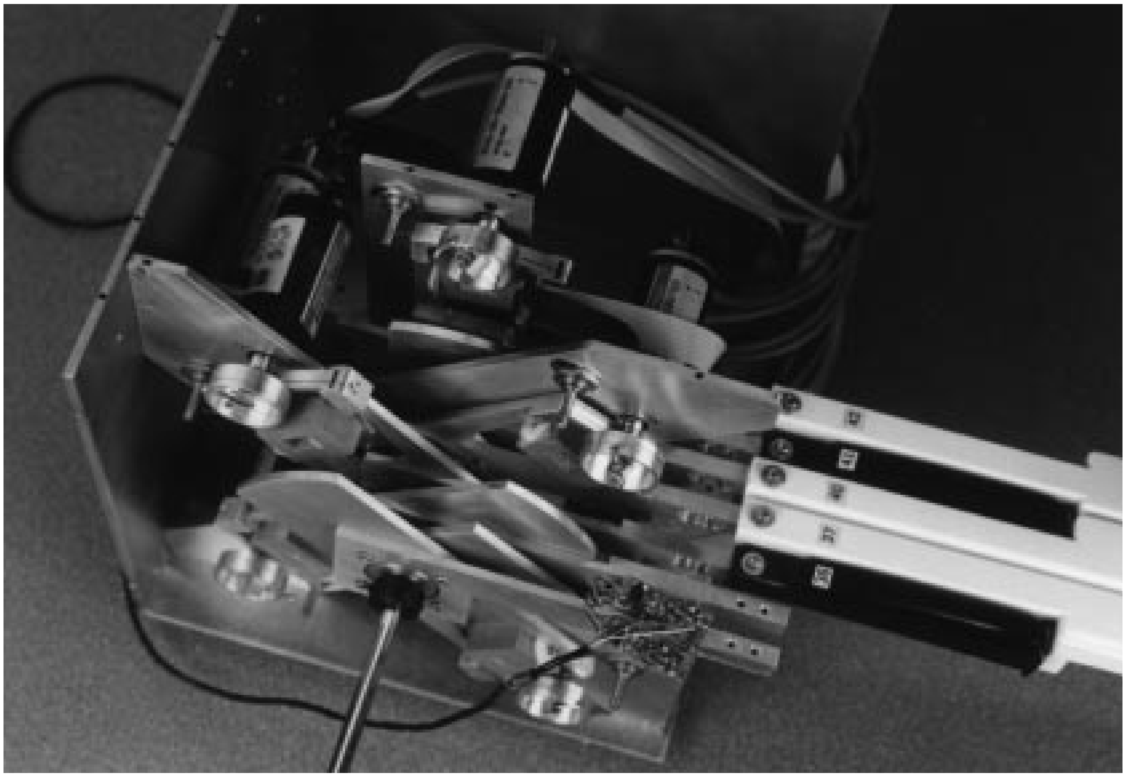

Figure 4.11: *Touchback Keyboard: View with Top Open*