POLYPHONIC VELOCITY-SENSITIVE KEYBOARD INTERFACE

Blair D. McKay, Barry L. Mills and David M. Carr
Department of Systems Design Engineering
University of Waterloo
Waterloo, Ontario, Canada
P.J. 301

ABSTRACT

The design of a music keyboard interface for a microcomputer-based music system is described and details are given for a prototype which incorporates a keyboard obtained from an acoustic piano. Variable inductors are used as key position sensors and a time-multiplexed digital circuit scans the keyboard for events which occur whenever a key is pressed or released. Fifteen bits of data, representing the key number, the direction of travel, and a measured time inversely proportional to key velocity, are sent to the microprocessor through two parallel ports each time an event is detected.

The use of inductive sensors eliminates the problem of wear and contamination associated with traditional contact switches. Furthermore, multiple contacts or sensors per key are unnecessary since a single low-cost device is able to measure the position of a key over its full range of travel. The sensor mechanism can be installed on a wide variety of keyboards, including those in place on existing instruments such as acoustic pianos.

A measure of each key's velocity is made by timing its travel between two reference positions. To allow for response variations in the inductive sensors, each key has its own set of reference positions, which are stored in a shared random access memory. The sensitivity of each key can thus be adjusted individually by changing the reference value stored for that key. This makes delicate mechanical adjustments of individual sensors unnecessary. An auto-calibrate programme on the microcomputer calculates all the key reference positions in a few seconds.

The paper also describes how the multiplexed digital hardware, developed initially for use with inductive sensors, can be used to provide a suitable microprocessor keyboard interface for several other sensor types.
INTRODUCTION

Musical keyboards are essential components in most performance-oriented synthesizers. Their design and implementation has attracted the attention of many musicians and engineers. (See for example [1,2,3,4,5] The challenge is to create systems which can extract all of the musically useful information transmitted to the keys by the musician so that various characteristics of the sound output can be controlled in an effective manner.

Fundamentally, a musical keyboard should be able to operate polyphonically, should be ergonomically comfortable and should be "touch-sensitive", i.e., sensitive to at least key depression rates. Other variables can also be measured and used in musically interesting ways. The importance of a touch-sensitive keyboard was described by LeCain[1] as follows:

"In general, phrasing, articulation, and general control of the broad structural features benefit from the touch sensitive (force controlled) action. Attack and decay can also be usefully controlled by the touch of the performer. The 'singing touch', which on the piano means merely an attitude, becomes a directly controllable quality on the touch-sensitive keyboard."

This paper describes the design of a polyphonic touch-sensitive keyboard which is used to control a microprocessor-based digital synthesizer. The sensor characteristics are discussed together with details on the necessary interface hardware which links the keyboard to the microcomputer.

CODING KEYBOARD ACTIVITY

The performance of music on a keyboard may be considered as a sequence of individual notes each of which has the following four parameters:

- Key number (which key)
- Start time (when hit)

---

1 Dr. Hugh LeCain (1914-1977) was a pioneer in the development of electronic music equipment in Canada. LeCain built a number of interesting devices including an electronic synthesizer (1946) and a touch-sensitive synthesizer keyboard (1954). Further information on this remarkable innovator is available in a series of newsletters published by The Hugh LeCain Project, 17 Davies Avenue, Toronto, Canada, M4W 2V4.
- key velocity (how hard)
- end time (when released).

This information can be extracted from the keyboard by detecting the
motion of the keys, and with reference to time, identifying the "KEY DOWN"
and "KEY UP" events for each of them. KEY DOWN events are represented by
numerically identifying the key that caused the event, the time of that
event, and information indicating how hard the key was hit. KEY UP events
normally require only the key number and time. A list of key events described
in this way can be used to represent and reconstruct a keyboard performance.

In a traditional keyboard instrument, such as the piano, the keyboard
is closely coupled to the sound generation "hardware". Energy applied
to the keyboard by the musician's fingers is quickly transferred to the
piano strings via the action. Loudness and, to some extent, the timbre
of a note is directly related to how much energy was applied to the key.
This energy is most easily measured in a synthesizer keyboard by measuring the
velocity of the key. Velocity, when measured as the key is nearing the
bottom of its travel, corresponds to the velocity with which the hammer (in
a piano) strikes the strings.

In an electronic instrument the coupling between the keyboard input
and the resulting musical output is not restricted and controlled by the
physical design of the instrument, to the same extent as the piano. Because
of this, the information provided by the keyboard may be used to control the
sound generation equipment in many different ways. Touch-sensitive information
is best used, however, to control the sound output in ways which are consistent
with the musician's expectations.

METHODS FOR SENSING KEYBOARD ACTIVITY

The motion of an individual key may be detected using an electrical
transducer or key sensor. The simplest form of sensor is the single pole
switch, and in monophonic analog synthesizers, key switches are connected to
a voltage divider circuit so that a precise voltage is generated which
corresponds to the key pressed. This voltage is chosen so that it can be used as an input to a voltage-controlled oscillator to produce the desired frequency. In addition to this control voltage, a trigger signal for initiating envelope generators is usually provided. One can only simulate touch sensitivity with such a keyboard by providing a secondary input device such as a slider, pedal, or touchwheel.

Touch sensitivity can be directly obtained by measuring key velocity. While it is possible to use a velocity sensitive transducer, the more common method is to measure the time taken for a key to travel a fixed distance. This time, which we will call the TRAVEL-TIME, is inversely proportional to key velocity. Two switches per key, mounted in such a way as to close at different points in the key’s travel, can provide the “fixed distance”. Care must be taken in mounting and adjusting the switches, however, so that all keys have the same “distance” and so that the “distance” is in the same part of each key’s travel. This must be done to ensure that the travel-time information will be consistent between keys.

An alternative to using two sensors per key is to use a transducer which provides an output signal related to key position. Conceptually, the simplest transducer of this sort is a variable resistor which can provide an output voltage directly proportional to key position. This output voltage can be differentiated to provide velocity information, or it can be compared to two reference voltages which correspond to two suitable positions in the key’s travel. This latter method provides sufficient information to determine whether the key is up, down, or in transit, and it also has other advantages which will be described later.

A resistive sensor, while conceptually simple, can be difficult to mechanically mount in a keyboard without interfering with the “feel” of the keys. This problem can be solved using capacitive (2,4) or inductive sensors. Moreover, contact wear and contamination, often associated with switch-type sensors, can be eliminated.

Inductive sensors have been incorporated into the keyboard described in this paper. In this design a ferrite rod attached to each key moves
freely through an air coil inductor as the key is depressed (see Fig. 1).

![Diagram of inductive key position sensor](image)

**Figure 1. Inductive Key Position Sensor Showing Mounted.**

The key position may be determined from the coil's inductance. This is done by measuring the peak voltage present across the inductor when a voltage pulse is applied to it through a series resistor. The inductor's peak voltage is then compared with two reference voltages previously obtained when the key was physically held at two different points in its travel. The results of these two comparisons are stored as two bits (the 'A' and 'B' flags) and identify the key's position as described in the next section.

**KEY STATES**

The position of each key may be represented by the values of the 'A' and 'B' flags (see Fig. 2). When the key is in the up position, neither flag is set. Flag 'A' becomes set when the key position moves below the
The six key states are now examined by stepping through them in sequence as a key is depressed and then released. The normal starting position is the "KEY UP" state (see Fig. 3). This changes to "GOING DOWN" when the 'A' flag becomes set as a result of the key physically moving below the upper reference position. The state will continue to be "GOING DOWN" until the 'B' flag is set by the key moving below the down reference.
position. If for some reason the key is never fully depressed and the key moves back above the upper reference position, the 'A' flag will be reset causing the "KEY UP" state to be re-entered. Usually, however, the 'B' flag becomes set and the state changes to "DOWN EVENT". At this time the
computer should be informed that a "DOWN EVENT" has occurred. The number of the key, and its travel-time must be passed to the computer before the state can advance to "KEY DOWN". If the computer is "RUN", the "DOWN EVENT" state will be maintained until the computer becomes "STOPED". The upward motion of the key causes the cycle to continue through the "GOING UP" and "UP EVENT" states to the "KEY UP" state in a similar manner to the downward motion.

The time "travel-time" may be calculated by timing the duration of the "GOING DOWN" state. This is easily accomplished by periodically incrementing a suitable counter during this state. If the counter reaches its maximum count before the "DOWN EVENT" occurs, the counter should be forced to maintain its maximum count. A convenient way of starting the counter at zero is to keep it "zeroed" while it is in the "KEY UP" state.

IMPLEMENTATION DETAILS

The conceptual design presented in the preceding for a single key has been implemented for an eighty-eight key keyboard obtained from an electronic piano. Some attempt has been made to maintain the characteristic "feel" of the keys.

A time-multiplexed circuit sequentially scans each of 128 possible keys (only 64 are active) in approximately one millisecond. Each scan of a single key consists of four sub-cycles. In the first sub-cycle the 'A' flag is set by comparing the key's current physical position (represented by a peak voltage), to the upper reference position (also a voltage). Flag 'B' is set during sub-cycle two. The third sub-cycle is used to update the key state as described above. The new state is determined from the old state plus the conditions of the 'A', 'B' and RUNY flags (see Fig. 3). In the fourth sub-cycle the travel-time is updated. This may require it to be incremented (GOING DOWN state), re-zeroed (KEY UP state), or left the same. A complete keyboard scan consists of repeating these four sub-cycles for all 128 possible keys.
Data necessary for the keyboard scanning operations is stored in a 1K by 8 bit random access memory (RAM). This memory keeps the upper and lower reference positions, the key state and the travel-time for each of the 128 possible keys. A total of 512 bytes of the RAM are currently being used.

The key reference positions stored in RAM are converted to voltages through a Digital to Analog Converter (DAC). The use of RAM to store key reference positions has several advantages, the primary one being that this allows each key to use its own set of "personalized" reference positions. A key sensor may thus be adjusted (calibrated) by simply changing the values stored in RAM for that key, rather than by physically adjusting the sensor. In this way sensors may be individually adjusted to a mechanical equivalent of approximately 0.05 mm (0.002 inches).

The entire keyboard may be automatically calibrated, using the computer, in a matter of seconds. The rest (up) position of each of the keys is measured first. The user is then requested to hold all the keys down simultaneously using a bar and the voltages at the down positions are measured. From the key position information obtained in this way, the computer can calculate appropriate values to be used as reference positions and load those values into the keyboard interface RAM.

A number, representing one of the six possible key states, previously described, is also stored in RAM. The key state is read from RAM and stored in the STATE LATCH in the third sub-cycle. It is then updated using its current value, plus the status of the 'A', 'B' and BUSY flags. The new state is then written back to RAM.

The RAM is also used to record the travel-time for each of the 128 possible keys. During the fourth sub-cycle the travel-time is read from RAM and either incremented (GOING DOWN state) or zeroed (KEY UP state) before being written back.
KEYBOARD INTERFACE OPERATION

Fig. 4 is a block diagram showing the complete keyboard—microcomputer interface. The COUNTER runs continuously, providing the RAM with addresses (key # and sub-cycle), and the CONTROL LOGIC with signals to control inter-sub-cycle timing. Data from the RAM may be sent to one of several places depending on the sub-cycle. During the first two sub-cycles, data from the RAM is sent to the DAC to generate the position reference voltages used by the COMPARATOR to set the A and B FLAG registers. The ANALOG MULTIPLEXER provides the other voltages used by the COMPARATOR. It selects the appropriate inductive sensor and allows it to be pulsed by the PULSE GENERATOR. In the third sub-cycle the state is read from the RAM and stored in the STATE LATCH. This may then be incremented or decremented by the CONTROL LOGIC before being written back into the RAM. Similarly the travel-time is read from RAM and stored in the T-TIME COUNTER. It too may be incremented or decremented by the CONTROL LOGIC before being returned to the RAM.

If an "event state" is detected by the CONTROL LOGIC and if the microprocessor PORTS are not busy, then the key number and the travel-time are loaded into the KEY PORT and T-TIME PORT respectively. This also sets the BUSY flag. When the microprocessor reads these PORTS the BUSY flag is automatically reset. While the PORTS are busy the keyboard interface continues to scan the keys. Any keys that enter an UP or DOWN EVENT state will remain in that state until they can transmit the event data to the PORTS. This makes it possible to accurately measure the travel-times for several keys that are depressed simultaneously, as is the case when a chord is played. The time that the microprocessor associates with these events will of course depend on how fast the microprocessor can deal with one event before accepting another. This may typically be on the order of five to ten milliseconds.

CONCLUDING REMARKS

The use of inductive sensors, and time-multiplexed digital circuitry makes it possible to build the polyphonic velocity-sensitive keyboard interface described above for less than one hundred dollars (electronic components and sensors, keyboard not included). A prototype interface, using forty-

593
Figure 4: Keyboard Interface Block Diagram
five integrated circuits, is currently being used as the primary input device
on an experimental microprocessor-based music synthesizer system, with
encouraging results (5,6,7). A second more portable keyboard (61 keys)
is currently under construction. It employs pre-mounted switch contacts
(inductive sensors may be added later) to select one of two voltages from
a voltage divider. These voltages are then connected to the same interface
hardware used by the present inductive sensor keyboard. Other input devices,
such as foot pedals (for sustain, etc.) and potentiometers, can be accommodated
to the interface circuitry by expanding the analog multiplexer, and treating
the additional devices as special "keys".

REFERENCES
Synthesizers", 1980 International Computer Music Conference, New York,
November 1980.
Music System", 1980 International Computer Music Conference, New York,
November 1980.
for Musically-Performing Skills", submitted to the World Conference on
Computers in Education, Lausanne, Switzerland, July 1982.