Digital polyphonic keyboard

Digipoly is a versatile keyboard instrument capable of producing many electronic organ, piano and synthesizer sounds. It is entirely digital, with a Midi interface, and includes an 8088-based control processor and a high-speed t.t.l. signal processor with other potential applications.

Most current electronic keyboard instruments use a combination of digital and analogue circuits. Output of these instruments is typically generated by sixteen conventional analogue oscillators whose frequency, and subsequent filtering/amplification, is digitally controlled.

Digipoly is the outcome of an investigation into the possibilities of producing a musical instrument using entirely digital note generation techniques. The result is a useful piece of equipment capable of synthesizing many conventional electronic organ sounds and many synthesizer and electronic piano sounds. However, the basic instrument is not capable of producing any voice for which the harmonic structure of the sound changes during the note.

The main advantage of Digipoly over older analogue designs is the simplicity and versatility of its digital circuits. A standard Midi (musical instrument digital interface) bus connection is included so that Digipoly can be used with other instruments, sequencers and under remote computer control.

Figure one shows the interconnection of Digipoly’s various parts. On its own board at the heart of the system is an 8088 microprocessor which controls all instrument functions. It has an 8Kbyte eprom for its program and look-up tables and 2Kbyte of cmos ram with battery back-up to retain user settings for user-defined voices when the main power source is removed.

The 8088 microprocessor controls all of the instrument through 16 eight-bit parallel i/o ports. This means that for development and debugging purposes, the complete microprocessor board can be removed from the instrument and replaced by a cable to a microcomputer which addresses the same 16 ports. Software for Digipoly was developed in this way on an 8088-based computer.

Front panel controls of the instrument are polled by the microprocessor. For economy, I chose a simple front panel with push buttons and leds, Fig.2. Two rotary controls are also used, one for the master volume setting and the second, connected to the a-to-d converter, whose function depends on the push-button selection.

The sprung-action keyboard is a standard 61-note C-to-C plastic one. It is scanned every 2ms by the microprocessor and appears as an array of 61 ones and zeros reflecting the state of

by D.J.Greaves B.A.

David Greaves is a postgraduate research student at St John's College Cambridge working with new computer architecture, in particular on natural language processing algorithms.

David is three time a winner of the Esso-sponsored Design Technology competition, holder of an IEE Jubilee Scholarship and a former contestant on BBC TV's Young Scientist of the Year programme. He plays the guitar and, being a lover of most types of music, is fascinated by anything that combines electronics, music and computer algorithms.

His many designs include a digital spectrum analyser, several digital sound effect units and an ever growing computer which he started building in 1977. This 200Kbyte computer with hard disc drives runs home-written software including a Basic Interpreter, BCPL compiler and assemblers.
Fig. 1. Block diagram of the digital polyphonic keyboard instrument. All control functions are handled by the 8088 microprocessor.

Fig. 2. Front panel legend. There is a variable-context button by each row in the two legend sections. A led under each column indicates when a column is in use.

the keys. Front-panel controls allow keyboard pitch to be stepped up or down over six octaves and also transposed up and down by six semitones. Absolute pitch can be varied over one semitone to allow tuning to other instruments. DIGIPOLY is equipped with a midi interface which is a fast serial data link to other instruments. Bit serialization is performed by the 8088 microprocessor and timed using software delays from the processor's 15MHz crystal. Received-data start bits are used to interrupt the microprocessor whose interrupt routine then assembles a byte which is stored in a queue in memory. This queue is polled by the main program for new commands which are then interpreted according to the Midi standard.

Sound generation is performed by a microcode program running on a simple processor built from discrete t.l.i.c.s. This processor, performing around five million instructions each second, simulates eight asynchronous oscillators whose frequency can be varied in 0.5Hz steps from zero to 16kHz. Waveforms of these oscillators are stored digitally in a 129-by-256 element array and their amplitude is adjustable over 64 linear steps. Each oscillator sends a new value to the output digital-to-analogue (d-to-a) converter at about 35kHz.

Having eight separate note channels within the instrument means that it is normally possible to hold eight keyboard keys together and hear all eight notes polyphonically. The 8088 microprocessor automatically assigns a new channel for each key. In 'double-up' mode, which is selected from the front panel, two channels are used for each key so only four keys may be used together. Reduction in the number of channels available is also sometimes caused by the automatic arpeggio effect, the Midi bus and the sustain pedal.

Output from the d-to-a converter is fed through a three-pole low-pass filter with a 5kHz cut-off frequency which removes steps from the digital waveform and sums the eight oscillators. The d-to-a converter is eight bits wide and used eight times for each sample, giving an effective dynamic range for all voices of 66dB; when no key is pressed, converter output is zero so that the signal-to-noise figure is generally better than this figure. Filter output is fed through the volume control to a ⅛in jack output socket.

Figure 3 shows some of the envelope possibilities. A variable 0.1 to 5Hz sinewave oscillator is included to apply vibrato and tremolo modulation to the output; speed and depth of these effects are controlled digitally by the 8088 microprocessor.

Frequency synthesizer

Figure 4 shows a notegeneration channel. Eight such channels are implemented by the t.l.i. microprocessor in microcode and each channel has two sixteen-bit registers, P and F. Frequency of the oscillator is determined by the F register and is set by the 8088 processor before a note starts. In each channel, the P register holds the current phase of the oscillator and has the contents of the F register added to it at regular intervals of 28μs.

The rate at which the P register overflows and returns to zero is the fundamental frequency of the note being
duced by the channel. Being 8 bits, the P register can count up to 65536; if the P register holds unity then the synthesized frequency is \((1/0.000028)/65536\) which is 0.5Hz. Hence the frequency of the channel is 0.5Hz multiplied by the value in the P register.

Indexing of the waveform table is carried out by the P register’s eight most-significant bits. The waveform table, held in an array in the t.t.l. processor address space,

---

**T.t.l. processor instructions**

All instructions are one byte long, the most significant four bits giving the instruction opcode and the lower four bits giving the addressing mode word where pertinent.

For instructions that do not specify an m value, the lower four bits of the opcode should be zero. The INCV instruction adds one to the V register modulo n where n is the number of sound channels that the Digipoly has. This is set by bits to eight. The INCV instruction has the same opcode as the HOST instruction so that the two can be performed at once. Other unused opcodes are decoded in hardware and so can easily be used to add new features to the instruction set.

**Bit patterns for the opcodes are**

<table>
<thead>
<tr>
<th>Opcode</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>LOAD m</td>
</tr>
<tr>
<td>1000</td>
<td>STORE m</td>
</tr>
<tr>
<td>0001</td>
<td>ADD m</td>
</tr>
<tr>
<td>0101</td>
<td>ADC m</td>
</tr>
<tr>
<td>0110</td>
<td>SUB m</td>
</tr>
<tr>
<td>0011</td>
<td>CLEAR</td>
</tr>
<tr>
<td>1010</td>
<td>LOOP</td>
</tr>
<tr>
<td>1011</td>
<td>INCV</td>
</tr>
<tr>
<td>1011</td>
<td>HOST</td>
</tr>
<tr>
<td>1111</td>
<td>NOP</td>
</tr>
</tbody>
</table>

The addressing mode bit patterns (m) are

<table>
<thead>
<tr>
<th>Mode</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>PL,V</td>
</tr>
<tr>
<td>0001</td>
<td>PH,V</td>
</tr>
<tr>
<td>0010</td>
<td>FL,V</td>
</tr>
<tr>
<td>0011</td>
<td>FH,V</td>
</tr>
<tr>
<td>0100</td>
<td>VOL,V</td>
</tr>
<tr>
<td>0101</td>
<td>Not used</td>
</tr>
<tr>
<td>0110</td>
<td>DAC0</td>
</tr>
<tr>
<td>0111</td>
<td>DAC1</td>
</tr>
<tr>
<td>1000</td>
<td>SQ,A</td>
</tr>
<tr>
<td>1010</td>
<td>WV,A</td>
</tr>
<tr>
<td>1100</td>
<td>E0</td>
</tr>
<tr>
<td>1101</td>
<td>E1</td>
</tr>
<tr>
<td>1110</td>
<td>E2</td>
</tr>
<tr>
<td>1111</td>
<td>E3</td>
</tr>
</tbody>
</table>

---

**Fig. 4.** Note generating system. Microcode in the t.t.l. processor runs eight of these systems. A pitch register is repeatedly added to an accumulator which is used to index a stored waveform. The waveform is multiplied by a volume factor using the ‘quarter squares’ method and then sent to the output d-to-a converter.

---

**Fig. 3.** Envelope profiles. The general a.d.s.r. (attack-decay-sustain-release) envelope is shown in (a), preset envelope profiles are shown in (b-e), and (f) is a non-a.d.s.r. profile which produces pulses. A pre-echo profile similar to the general a.d.s.r. form but with two attacks is shown in (g). Shaded portions indicate the time that a key is held.
contains values in the range -64 to +64. Contents of the array are initially loaded by the 8088.

Each channel has a volume register holding a value between zero and 63. It is necessary to multiply values from the waveform table by the volume value to control output-signal amplitude. This multiplication is performed using the 'quarter-squares' method. Use of a simple look-up table to multiply a seven-bit number by a six-bit number to produce an eight-bit result would take eight kilobytes of memory; with the quarter-squares method, only 256 bytes are needed. The identity

\[ A^2 - B^2 = (A+B)(A-B) \]

is used. Values from the waveform table and volume register are summed and differenced, then the difference between the squares of these values is computed. In order to keep the number of bits under control, the square table actually contains the function

\[ SQX = X^2/128 \]

which gives results in the range 0 to 128. When two values from the square table are differenced, a full eight-bit value is produced and this is the result of the computation.

Figure five shows the t.t.t.I. processor architecture. This processor was specifically designed for Digipoly but is general enough for use in many other applications. Different address spaces are used for the program and data so that the hardware can be more easily 'pipelined'.

A protocol allows the microcode program and sequential execution is controlled by the eight-bit program counter register. All data manipulation instructions use the eight-bit accumulator either as a source or destination (or both) of one of the operands. There are four general-purpose extension registers E0-E3 and an index register V which is normally used to select which synthesizer channel is being processed.

Main memory is partitioned in hardware into several regions. There are five arrays which are always indexed by the V register: PL, PH, PL, PH and VOL. These arrays contain eight-bit bytes and are 16 locations long. They are used for the low and high-order bytes of the P, F and volume registers respectively.

There are also two arrays of 256 bytes, the square table SQ and the waveform table WV, which are indexed by the accumulator. Output from the t.t.t.I. processor is achieved by writing to one of two locations, DAC0 and DAC1. Only DAC0 is used on the present Digipoly. Input to the processor is always performed by direct memory access (d.m.a) to the processor memory under the control of an external device.

Descriptions of the 8088 processor section and Midi bus are included in the next article.

---

**Fig. 5. Architecture of the t.t.t.I. processor which has ten instructions. The accumulator is used for all arithmetic results and there are four extension registers for temporary results. The main memory is partitioned in hardware and is accessed using several different addressing modes.**

**Fig. 6. Second harmonic at 36dB and sampling rate at 85dB of a 1kHz sine-wave (a), two sinewaves at 750 and 1000Hz with intermodulation product -50dB at 1750Hz (b), and 120Hz squarewave with low spectrum of odd harmonics (c).**
The instrument has two processors — an 8088 microprocessor for control functions and a t.t.l. processor for note generation. There are 18 instructions in Digipoly's microcode program, list 1, which execute sequentially and then start again. The final instruction, INCV, causes the program to be run on each sound channel in turn.

Frequency of the master clock is divided by the length of the program and by the number of voices to give the sample rate at the audio output d-to-a converter. With a 5MHz master clock rate this is

\[ 5 \, 000 \, 000/(18 \times 3) = 35 \text{kHz}. \]

An assembler written for the microcode language in BCPL produced the code in list 1, but microcode can easily be manually assembled using the instruction set described last month.

The first three instructions of the microcode increment the P-register low-order section and the next three the high-order section. At address six, the value from the waveform table is sampled and this is multiplied by the VOL, V register in the remaining instructions. At address 16, the computed result is sent to the output d-to-a converter.

Each channel sends its output to the same converter and the value is latched there until the next channel sends a value.

This gives a discontinuous waveform. Summation of the eight channels into a single continuous audio wave is performed by the integrating behaviour of the analogue low-pass filter following the converter.

List 1 Microcode of t.t.l. processor is only eighteen bytes long.

---

**Features and software availability**

Digipoly is an eight-note polyphonic digital musical instrument with a five-octave keyboard transposable over a nine-octave useful range. It includes:

- Comprehensive envelope generator controls
- Vibrato and tremolo control
- Midi interface
- Hundreds of front-panel selectable waveforms
- Dray-powered maintained memory for 16 user-defined voices
- Rotary control for adjustment of many parameters

Note frequencies are not rigidly locked as in divider type organs. A detune facility introduces a valuable amount of scale error.

Digipoly can be built for around £175 excluding case. Software is available in various forms from the author at 5 Grovely Way, Crampmoor, Romsey, Hampshire S05 9AX. A fifty-page listing of the 8088 source program is £3 and a 40-track disc for the BBC microcomputer, holding source, object and related files, is £4 (single density). Programmed 256 eeproms containing the 8088 object code and a bipolar prom containing the t.t.l. processor code are £8.50 and £4.00 respectively. Please include £1 for UK postage and make cheques payable to D.J. Greaves. Brave readers can obtain a copy of the hexadecimal listing by sending a large stamped addressed envelope and a cheque for £1.35 to our editorial offices. Please make this cheque payable to Business Press International.
This discrete high-speed t.l.l. processor was developed specifically for Digipoly but is general enough in design for many other applications. Its purpose is note generation. Microcode software for the ten-instruction processor is stored in a bipolar prom.
Polyphonic keyboard
part 3

Midi interface details are included in this third article on a versatile keyboard instrument using two processors.

Software for the 8088 micropro-
cessor was written in assembly
language and programmed into a
2764 8Kbyte eprom. There are
two devices in the processor
memory map, this eprom with
addresses between E000 and
FFFF and a 2Kbyte cnos ram
with battery backup using
addresses from zero to 7FF.

As mentioned last month, the
processor accesses the whole of
the rest of the instrument in an
address space of 16 parallel
ports. Names and functions of
used ports are given opposite.

From the user’s point of view,
there are two main aspects to
controlling Digipoly. First is
operation of the front panel con-
trols for setting up the instru-
ment’s sound and performance
characteristics, and second is the
response of the keys when
pressed, i.e. music.

Communication between these
aspects relies on a set of global
variables in the 8088 memory
map known as voxcons. These
voxcons are stored in a 32byte
array containing all information
about the current sound of the
instrument. Operating the front
panel controls changes some of
the voxcons in specific ways and
playing the instrument uses them
to create sounds.

Part of the random-access
memory forms a library of sixteen
sets of voxcons which are
retained by means of battery
backup while the main power is
switched off. Save and recall
functions issued from the front
panel copy the current voxcons to
a position in the library and back
respectively. Two other similar
variables, for fine and coarse
pitch settings, are not saved in the
library. It is important that
these too are saved when the
main supply is switched off.

When power is applied, the
8088 is given a power-on reset
which causes it to start execution
at address FFF0, which is in
the eprom. First it initializes
variables in its own memory map
and then sends the table of squares to the t.l.l. processor. The volume
registers are all zeroed and all
voxcons are set to default values
giving a simple reed organ type
sound, i.e. a triangle wave,
sudden envelope and all other fea-
tures off.

The software then enters the

Main polling loop which, in out-
line, performs the following:

— scan keyboard, turning on or
  off notes that have changed
  since the last scan.
— send values to the d-to-a con-
  verter influence register.
— advance each envelope regis-
  ter phase
— inspect the Midi bus queue for
  commands
— scan front panel controls
— read control knob if enabled

When there is no work to be
done, the polling loop takes just
under 1ms. Advancing of enve-
lope phases only takes place
every fifth time around the loop,
i.e. once in about 4ms.

Midi bus

Midi, short for musical instru-
ment digital interface, is a stan-
ard interface used on nearly all
modern keyboard instruments. It
can be used for controlling sound
generation sections of one musical
instrument from the keyboard of
another. Drum computers and
sequencers can also be intercon-
ected using the bus and there
are Midi adaptors which allow
instruments to be controlled by
computer.

Three 180° five-pin DIN socke-
tes are fitted to the instrument
-Midi-in, 'Midi-out' and 'Midi-
through'. Data is transmitted on
pins four (positives) and five
using a 10mA current loop to
drive opto-isolators at the receiv-
ing end. Cables used are shielded
twisted pairs with the shield connected to pin two at both ends; this pin is only earthed at the transmitting end. The connector chassis is not connected and pins one and three are unused.

Data is transmitted serially at 31.25 kilobaud. One start bit of zero is sent then eight data bits with the least-significant bit first followed by a stop bit of one.

Midi commands are normally three bytes long. First is the command byte with its most significant bit set on. Represented in the least-significant four bits of this byte is a channel number, n. Subsequent bytes specify parameters and have values between zero and 127.

The bus is logically divided into sixteen channels, referred to as one to sixteen, but of course the bit pattern of the n field in the command byte has values from zero to fifteen. Channel numbers that a particular instrument uses and responds to are set from the front panel of the instrument and Digipoly is no exception. Using different channel numbers allows several instruments to chained together using the Midi 'through' sockets and independently controlled from a single computer or sequencer.

In some commands, the second byte is 'kk' which corresponds directly to a key on the instrument keyboard. Byte kk is zero for lowest C and 3C for top C. If a number greater than 3C (60 denary) is received then the instrument will play that note as though the keyboard were continued to the right, provided that this is still within the instrument's pitch range. The highest valid value will depend on how the keyboard is currently transposed.

One restriction on Digipoly is that the Midi 'out' socket cannot be used at the same instant as the 'in' socket. This is because data is serialized and deserialized in software and not using a uart (universal asynchronous receiver and transmitter) i.e. One problem that this causes is that the useful loop test involving connecting the 'out' and 'in' sockets of the same instrument will not work.

Midi output

If transmit-on mode is selected on the front panel (Tx on), then information from the keyboard is not interpreted by Digipoly but is transmitted on the Midi 'out' socket. The channel number must also be selected from the front panel. Commands that may be transmitted are (Gn, kk 40) which notifies any device on channel n that key kk has just been pressed on the keyboard, (8n kk 40) which notifies any device on the channel that key kk has just been released, (Bn 0p yr) used to adjust sound-generating parameters of other instruments and (91 04) sent as a continuous stream of double bytes on the Midi 'out' socket.

In the third command, the parameter to be modified, p, is a value ranging from zero and 15 selected on the front panel. On selection of this

---

Names and functions of parallel ports used on the 8088 microprocessor.

**BUTTONS**, port 3, contains a bit-mapped image of front-panel control buttons.

**LEDS**, port 4, has various bit patterns written to it to control the front-panel indicator leds.

**BUT1**, port 6, is a bit-mapped image of general-purpose buttons 1-8.

**BUT2**, port 7, is a bit-mapped image of buttons 9-16. All of these buttons are push-to-make and produce a zero bit when pressed.

**CONTDAC**, Port 8, when written to latches an eight-bit control word for the auxiliary d-to-a converter. This converter determines various analogue parameters. The same latch is used for scanning the clavier keyboard.

**STATUS**, Port 9, is a read-only port with bit flags as follows:

Bit seven, reflecting output of the oscillator is used for arpeggio (TBASE), and is available to clock a sequencer which could be added as a software extension.

Bit 6 is one if voltage from the rotary control potentiometer on the front panel is higher than that of the control d-to-a converter. In conjunction with

**CONTDAC**, this bit allows anaogue-to-digital conversion under software control so that the 8088 can read the knob position.

Bit 2 is zero if the sustain foot switch is pressed.

Bit 1 is one if the clavier key indexed by the value last written to **CONTDAC** is currently held down.

Bit 0 reflects the state of the Midi in serial-data line.

**INFLEUTE**, port 11, when written to causes the voltage on the control converter output to be stored on a sample-and-hold capacitor. There are five of these capacitors. Writing numbers zero to four has the following effects:

Code 000 sets the track oscillator frequency. The track oscillator produces a sinewave for tremolo and vibrato and can be varied from about 0.2 to 10Hz.

Code 001, vibrato depth, varies coupling of the track oscillator to the master clock to change the degree of vibrato in the sound by frequency modulation.

Code 010, tremolo depth, varies coupling of the track oscillator to the multiply input of the main audio d-to-a converter. This changes the amount of tremolo in the sound by amplitude modulation.

Code 011, fine pitch, varies the steady frequency of the master clock to the t.t.i. processor and so pitches the whole instrument.

Code 100 sets the TBASE oscillator frequency over a range of about 0.2 to 10Hz.

**HOSTREG**, port 12, has its most significant bit inverted and connected directly to the Midi out socket at the back of the keyboard for communication with other instruments and computers. Other bits in this port address regions of the t.t.i. processor main memory. Values correspond directly to values of m in the microcode instruction set.

**INDEX**, port 13, is similar to the previous one, except that it provides the offset address within a memory region.

**DATAX**, port 14, is used for writing data to the t.t.i. processor. Values are set up in HOSTREG and INDEX first, then the required data is written to this port for passing on to the t.t.i. processor when it next executes a HOST instruction. Completion of the last operation can be detected by the 8088 using a handshake on its test input. The 8088 has a WAIT instruction which causes the processor to wait until the test input goes low; it is normally advisable to execute a WAIT instruction before updating ports 12 to 14.
parameter, the main control knob is enabled and any change in its setting causes the adjacent parameter command to be retransmitted with a new value of yv. Value yv ranges from zero to 127. In the final command, used as a hardware diagnostic aid, the two byte transmission is repeated about once every 400μs.

Midi Input

Any data received on the Midi ‘in’ socket will be retransmitted on the ‘through’ socket. If the data is one of the following commands, it will also be interpreted by Digipoly. Since the keyboard has no touch-sensitivity software, the third byte of these commands need not be present for correct operation. Normally, the command will be interpreted as soon as the first two bytes are received. Commands are ( 8n kk 40 ) , note on command, which plays a Digipoly note exactly as if a key had been pressed, ( 8n kk 40 ) which has the same effect as releasing key kk on the keyboard, ( Ch Op 00 ) for selecting a preset voice in the range 0-15, ( Dn 7D 00 ) for turning all notes off and ( 0F ) for reinitialising the power-on reset sequence.

While using the first command, the keyboard remains functional but pressing key kk will have no effect until the key is released, when it will silence the note. Normal polyphonic restrictions apply regarding the number of these commands that may be sent. The command for selecting a preset voice has the same effect as using the play button on Digipoly’s front panel and the turn-all-notes-off command has the same effect as sending note-off commands to all current notes.

With respect to the Omni and Poly modes of the Midi standard, Digipoly always behaves as though Omni is on and Poly is on.

The t.l.t and 8088 processor circuits are described next.

Software availability

Digipoly can be built for around £175 excluding case. Software is available in various forms from the author at 5 Grovely Way, Crampmoor, Romsey, Hampshire S05 9AX. A 50 page listing of the 8088 source program is £2 and a 40-track disc for the BBC microcomputer holding source, object and related files is £4 (single density). Programmed 2764 eeproms containing the 8088 object code and a bipolar prom containing the t.l.t processor code are £6.50 and £4 respectively. Please include £1 for UK postage and make cheques payable to D.J. Greaves. Brave readers can obtain a copy of the hexadecimal listing by sending a large stamped addressed envelope and a cheque for £1.50 to our editorial offices. Please make the cheque payable to Business Press International.

Digipoly’s main board.
The t.l.t. processor is in the upper left area. Microcode prom, op-code latch, op-code decoder, 100ns register memory and the two a.l.u.s are in the top row of ICs. Analogue circuits to the lower right include the output low-pass filter, vibrato and tremolo (note the glass encapsulated thermistor) sections and the d-to-a converter influence sample and hold circuits. To the right is a small perpendicular board holding the 10MHz master clock and at the left, a 14-pin keyboard socket and two 25-pin sockets for the 8088 board and front-panel controls.

Control-processing circuit (left) with battery-backed ram for storing user-defined voices when the power is removed. The 8088 microprocessor controls all instrument functions through 16 eight-bit parallel i/o ports.
Block-to-make switches are used on the front panel. These are wired in a rectangular array of three rows by eight columns. Since no two buttons in the same row need be pressed at once, multiplexing diodes are not necessary.

The columns connect directly to the Q-bus. This bus is terminated with 2.2kΩ pull-up resistors which reduce line ringing and provide a logical zero if a switch is not pressed. The rows are fed from open-collector buffers which pull the row line low when the 8088 wants to read the switches. If a switch is pressed, a line in the Q-bus will be pulled low when the row line is pulled low. Decoding of the switches is performed in software. The front panel LEDs are controlled by writing to a single eight-bit latch consisting of IC48,49. For economy, a single 8/13S734 could be substituted for these two integrated circuits. Ten sets of single leads or pairs of leads need to be addressed so decoder IC63 is used to extend the eight available bits.

### Analogue electronics

Component IC32 is the control digital-to-analogue converter. It is an eight-bit device and produces a voltage on pin two proportional to the number that the 8088 has put into IC30. Control-converter voltage is fed to the comparator, IC28, where it is compared with the front panel knob voltage to give a single above/below bit. This bit is gated onto the Q-bus by IC29 when the 8088 reads the status port. The control converter also feeds IC3, a CD4061 eight-way bilateral switch.

When the 8088 writes to its influence register, one of the switches in IC3 is enabled and this feeds the control voltage to one of five holding capacitors. These are the five sample-and-hold circuits described earlier. Holding capacitor C10 is buffered by IC44. This capacitor sets the tremolo and vibrato oscillator frequency. The oscillator is a standard phase-shifted sine wave oscillator tuned to the appropriate frequency. Its output feeds IC14,15, both CA3080 variable

### Polyphonic keyboard i.c. list

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TL701</td>
</tr>
<tr>
<td>2</td>
<td>74LS161</td>
</tr>
<tr>
<td>3</td>
<td>6349-1</td>
</tr>
<tr>
<td>4</td>
<td>74LS734</td>
</tr>
<tr>
<td>5</td>
<td>74LS184</td>
</tr>
<tr>
<td>6</td>
<td>74LS245</td>
</tr>
<tr>
<td>7</td>
<td>74LS244</td>
</tr>
<tr>
<td>8</td>
<td>74LS373</td>
</tr>
<tr>
<td>9</td>
<td>74LS374</td>
</tr>
<tr>
<td>10</td>
<td>74LS161</td>
</tr>
<tr>
<td>11</td>
<td>74LS126</td>
</tr>
<tr>
<td>12</td>
<td>74LS157</td>
</tr>
<tr>
<td>13</td>
<td>74LS244</td>
</tr>
<tr>
<td>14</td>
<td>74LS373</td>
</tr>
<tr>
<td>15</td>
<td>74LS245</td>
</tr>
<tr>
<td>16</td>
<td>74LS504</td>
</tr>
<tr>
<td>17</td>
<td>74LS502</td>
</tr>
<tr>
<td>18</td>
<td>74LS501</td>
</tr>
<tr>
<td>19</td>
<td>74LS387</td>
</tr>
<tr>
<td>20</td>
<td>74LS387</td>
</tr>
<tr>
<td>21</td>
<td>74LS387</td>
</tr>
<tr>
<td>22</td>
<td>74LS387</td>
</tr>
<tr>
<td>23</td>
<td>74LS387</td>
</tr>
<tr>
<td>24</td>
<td>74LS387</td>
</tr>
<tr>
<td>25</td>
<td>74LS387</td>
</tr>
<tr>
<td>26</td>
<td>74LS387</td>
</tr>
<tr>
<td>27</td>
<td>74LS387</td>
</tr>
<tr>
<td>28</td>
<td>74LS387</td>
</tr>
<tr>
<td>29</td>
<td>74LS387</td>
</tr>
<tr>
<td>30</td>
<td>74LS387</td>
</tr>
<tr>
<td>31</td>
<td>74LS387</td>
</tr>
</tbody>
</table>

**Notes:**
- Control d-to-a converter
- Carry-bit logic
- Carry-bit reg.
- Converter address detector
- Converter influence reg.
- Audio output filter
- Fine pitch control
- Vibrato depth control
- Master clock Schmitt trigger
- Tremolo depth amplifier
- Tremolo-depth/timbrebase-speed buffer
- Tremolo and vibrato-speed buffer
- Tremolo depth amplifier
- Tremolo and vibrato osc.
- Front panel o.c. buffer
- Upper led latch
- Lower led latch
- 8088 clock generator
- 2K x 8 n-mos memory
- 8K x 8 n-mos memory
- 8088 address decoding
- 8088 low address latch
- 8088 1/0 transceiver
- 8088 5V regulator
- 8088 logic
- 8088 I/O decoder
- 5V regulator
- 5V regulator
- Front-panel led controller
- Microcode logic regulator
- Midi data buffer

---

**Polyphonic keyboard i.c. list**

1. TL701
2. 74LS161
3. 6349-1
4. 74LS734
5. 74LS184
6. TMM2016P
7. 74LS181
8. 74LS181
9. 74LS74
10. 74LS161
11. 74LS126
12. 74LS157
13. 74LS244
14. 74LS373
15. 74LS245
16. 74LS374
17. 74LS245
18. 74LS387
19. 74LS504
20. 74LS502
21. 74LS501
22. 74LS387
23. 74LS387
24. 74LS387
25. 74LS387
26. 74LS387
27. 74LS387
28. 74LS387
29. 74LS387
30. 74LS387
31. 74LS387

---

**Analogue electronics**

Component IC32 is the control digital-to-analogue converter. It is an eight-bit device and produces a voltage on pin two proportional to the number that the 8088 has put into IC30.

Control-converter voltage is fed to the comparator, IC28, where it is compared with the front panel knob voltage to give a single above/below bit. This bit is gated onto the Q-bus by IC29 when the 8088 reads the status port. The control converter also feeds IC3, a CD4061 eight-way bilateral switch.

When the 8088 writes to its influence register, one of the switches in IC3 is enabled and this feeds the control voltage to one of five holding capacitors. These are the five sample-and-hold circuits described earlier. Holding capacitor C10 is buffered by IC44. This capacitor sets the tremolo and vibrato oscillator frequency. The oscillator is a standard phase-shifted sine wave oscillator tuned to the appropriate frequency. Its output feeds IC14,15, both CA3080 variable
transconductance amplifiers. Their transconductance is set by a current into pin five.

Similar circuits are used for both 3080 devices consisting of transistor constant-current generators whose current is set by one of the influence register sample and hold circuits.

Amplifier IC$_{48}$ modulates reference current in the audio-output converter IC$_{27}$ to provide a tremolo effect. Because the converter zero output is actually half way through its range, the tremolo modulates the d.c. level of the output. This effect can be nullied by adjusting the tremolo null potentiometer to add an opposite d.c. modulation directly to the output. The potentiometer is the only preset requirement in the whole instrument.

Audio output from the converter is filtered by IC$_{50}$ which is wired as a Sallen and Key three-pole low-pass filter with pass-band gain of three and cut-off frequency of 5kHz. Output level is about 200mV when a single key is pressed.

Component IC$_{48}$ feeds IC$_{49}$ which controls frequency of the master clock to provide a vibrato effect. Along with the vibrato signal, IC$_{48}$ sums the fine pitch setting from IC$_{50}$ and voltage from the pitch-bend front-panel control. Output of IC$_{48}$ biases a voltage-controlled capacitor in the tank circuit of the clock. For a swing from +5 to -10V in bias, clock frequency varies by about 1.5MHz in 10MHz. However, this range is not all needed and only a positive and negative swing of about 2V is produced by IC$_{48}$. A swing of 50kHz in the master clock changes the instrument pitch by one semitone.

Control-converter register IC$_{31}$ is also used to read the keyboard. The prototype keyboard has a built-in digital encoder provided with a 14-pin dill header connector. A five octave C-to-C keyboard has 61 notes which therefore need a six bit address.

The six least-significant bits of IC$_{31}$ are fed to the keyboard and a single bit is returned. Numbering the lowest C as zero, if the key addressed by the six-bit number is pressed, then a logical one is returned, otherwise a zero. The bit is buffered by IC$_{32}$ onto the Q-bus so that it appears as bit one in the 8088 status port.
Polyphonic keyboard

part 5

Hardware details of Digipoly's two processors are presented in this final article.

The t.t.l. processor is clocked by a 10MHz Hartley oscillator (December issue circuit). Output at 10MHz is converted to a squarewave using the self-biasing stage around Tr3 and then divided by two by Ic38, to give the operating frequency of 5MHz. This signal clocks the instruction counter made from IC24, as shown in October on the t.t.l. processor circuit.

Microcode for the processor is stored in IC8, a bipolar program-able read-only memory (prom) from the MMI or Signetics 636x series. The 6349 is organized as 512 locations of eight-bit bytes and has an access time of about 55ns; any non-volatile memory device with an access time of less than 150ns could be substituted.

Each op-code from the prom is latched in IC1, while it is being executed. The four most-significant bits drive a four-to-sixteen-line decoder IC5, outputs of which select the operation.

Least-significant bits feed the data selector IC12 to provide a page-select address for the processor’s memory device, IC6. Being n-mos, memory IC6 is the slowest element in the t.t.l. processor and limits operating speed to 5MHz. Access time of this 2K-by-eight-bit static memory is 100ns. As can be seen from Table 1, which shows address allocations for IC6, not all the memory locations are used.

The two 7415181 four-bit arithmetic logic Ics form the arithmetic unit. Accumulator output register IC7 has its outputs permanently enabled in order to drive the A-bus. Together with the D-bus, the A-bus feeds into the a.i.u. where the two eight-bit numbers are combined to produce a new value for the adds 128 to the number which the adds 128 to the number which adds 128 to the number which adds 128 to the number which adds 128 to the number.

Values for the D-bus are supplied from memory. The type of combination performed by the a.i.u. depends on the control number fed into the a.i.u. on pins three to six.

Arithmetic instructions in the t.t.l.-processor instruction set are ordered so that the a.i.u. control number can be derived directly from each op-code bit pattern.

Bistable device IC4 is a one-bit latch containing the carry bit. It is updated only as a result of an ADD instruction and its contents are fed into the a.i.u. through IC39, 34 during an analogue-to-digital conversion instruction.

For store operations, IC16 is enabled to feed the accumulator value onto the D-bus so that the memory can be updated. Some store operations feed the output ports in an addressing mode decoded by IC5. One of two output ports is selected by IC4.

Data for the output analogue converter is latched in IC15 in two's complement form. Since the digital-to-analogue converter expects unsigned numbers on its inputs, IC16 offsets the zero value to the middle of the converter's voltage range.

Addressing modes are decoded by random logic and determine what feeds the address inputs to the memory. Four-bit V-register IC32 is gated onto A3 by IC31 if indexed addressing is required for V instructions.

Zeros are fed onto the low address bits by IC2 and IC13 feeds zeros onto the high address bits if indexing is not required. If accumulator indexing is required, buffer IC18 is enabled to gate the A-bus onto A3-7.

Input to the t.t.l. processor occurs during the HOST instruction under control of the 8088 processor. The 8088 processor has an lO bus called the Q-bus which feeds three register ports in the t.t.l. processor. Latches and operations of these three ports are shown on Table 2.

Writing data to port 14 also sets two-bit shift register IC9 to all ones. Clocking of the shift register takes place at the end of each HOST instruction; if the output bit is one, data from IC9 is written into the t.t.l. processor memory.

8088 microprocessor board also contains an 8K eprom, 2Kbyte ram, an 8284 clock generator and crystal, NiCd battery for data retention, various support chips and an extra reset switch.

Feature summary

Digipoly is an eight-note polyphonic digital musical instrument with a five-octave keyboard transposable over a nine-octave useful range. It includes:

- Comprehensive envelope generator controls
- Vibrato and tremolo control
- Midi interface
- Hundreds of front-panel selectable waveforms
- Battery-maintained memory for 16 user-defined voices
- Rotary control for continuous adjustment of many parameters

Note frequencies are not rigidly locked as in divider type organs. A detune facility introduces a variable amount of scale error.

by D. J. Greaves
B.A.
Microcode for the high-speed TTL processor is contained in a 63xx-series programmable PROM (Monolithic Memories, Fairford). This circuit, published on page 39 in the April 1985 issue of *Electronic Engineering*, is for programming such ROMs. There are one or two minor errors in the previously published circuit.

The 6349 PROM may be programmed manually by addressing a byte then using this circuit to program all bits that must be zero. The program pulse is fed to the chip-enable input, pin 15, and the bit pulse to the output pin.

Table 1. Memory use in the TTL processor

<table>
<thead>
<tr>
<th>Address</th>
<th>Array</th>
</tr>
</thead>
<tbody>
<tr>
<td>000-0FF</td>
<td>low-order osc. phases</td>
</tr>
<tr>
<td>000-0FF</td>
<td>high-order osc. phases</td>
</tr>
<tr>
<td>100-1FF</td>
<td>low-order frequency values</td>
</tr>
<tr>
<td>180-1BF</td>
<td>high-order frequency values</td>
</tr>
<tr>
<td>200-26F</td>
<td>voltage values</td>
</tr>
<tr>
<td>400-4FF</td>
<td>table of squares</td>
</tr>
<tr>
<td>500-5FF</td>
<td>waveform table</td>
</tr>
<tr>
<td>600</td>
<td>E register</td>
</tr>
<tr>
<td>680</td>
<td>E register</td>
</tr>
<tr>
<td>700</td>
<td>E register</td>
</tr>
<tr>
<td>780</td>
<td>E register</td>
</tr>
</tbody>
</table>

Note: the first five arrays are indexed using the y register; the two tables are indexed using the accumulator.

Presence of the shift register causes the data to be written twice on successive HOST instructions. The second writing ensures that the correct data is written to the TTL processor memory when the HOST instruction is executing at the time that the 8088 writes to IC4.

For speed, and since the INCV instruction does not need access to the memory, INVC and HOST instructions are performed together using the same op-code. This is simply a matter of wiring outputs of IC5.

The LOOP instruction causes the address counter to load a zero value on the next instruction. Since there is one instruction delay through the prom and IC4, the instruction after LOOP is always executed before control resumes at the start of the microcode. If needed, an accumulator indexed branch could be added by simply feeding the A-bus to pins three to six of IC2,16.

**8088 processor hardware**

The 8088 microprocessor is mounted on a separate board with ten other ICs, forming a self-contained microcomputer module. This module is connected to the rest of Digipoly through a single 25-pin connector carrying address/data lines for the 16 ports, signals for reset, busy and interrupt and an 8V supply line. This is a general-purpose computer board; I have used the same circuit in other applications.

Clock generator IC6 is an Intel 8284 device. It divides a 14.7456 MHz crystal frequency by three to obtain the two-to-one duty cycle required by the 8088 processor. Power-on reset and wait-state circuits are also provided by the 8284, although wait states are not used in this design.

Component IC5 provides the 8088 processor with 2K-byte of ram. This static ram is an HMM6116LP-3 e-mos device requiring only about 4μA of current in standby mode. It is practical to run the memory from a small NiCd battery for long periods; when Digipoly is switched on, the battery charges automatically from the 5V supply.

Supply pin 24 of the ram is connected to the 5V supply but the ground pin is connected to

**ELECTRONICS & WIRELESS WORLD JANUARY 1986**
ground through T16. When the board is powered T16 is switched fully on, bringing the main and i.c. grounds to within 50mV of each other. This figure is too small to affect logic switching. When power is removed however, T16 turns off, causing the main ground pin to move to -4.8V relative to the main ground.

Put another way, all ram control signals move to +4.8V, or logic one. Since the ram control pins are active when at logic zero, this disables the ram, putting it into low-power mode. When T16 is off, its collector-base diode becomes forward biased so T16 is needed to stop the battery being discharged through the base of T16.

Input and output to the 8088 is performed through the Q-bus. This bus is a set of eight lines running through the whole instrument. Associated with the Q-bus is the MD-bus consisting of four lines for selecting one of the 16 i/o ports that the 8088 can address.

Both the MD-bus and the transceiver buffering the bus, IC66, are set up before master strobe line GO is pulsed low. This gives time for written data to propagate and settle. On input cycles, the data is returned to the 8088 before the trailing edge of the GO pulse. This pulse is always about 350ns long.

On arriving at the board, the active-low interrupt-request signal is inverted and fed to the 8088 maskable-interrupt input. The non-maskable input is not used. In response to an interrupt, the 8088 strobes INTA low twice at pin 24. The first strobe is a dummy one and the second reads an interrupt vector from the processor data bus. Eight diodes provide the vector by grounding each line during the acknowledge cycle, producing a zero value. Hence only vector zero is used and this sets software to read a serial byte from the MIDI-in port.

**Performance**

Figure 8a in the September issue shows the Digipoly output spectrum when synthesizing a single 1kHz sine wave. The noise floor is at -65dB relative to the single note. This is virtually all sampling noise which disappears when no notes are pressed since d-to-a converter

---

### Table 2. Port data latches

<table>
<thead>
<tr>
<th>IC</th>
<th>Port</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>26</td>
<td>12</td>
<td>affect type of memory to be modified</td>
</tr>
<tr>
<td>25</td>
<td>13</td>
<td>memory offset of type defined above</td>
</tr>
<tr>
<td>14</td>
<td>14</td>
<td>data to be stored</td>
</tr>
</tbody>
</table>

output is constant d.c.

Sampling noise can be heard quite clearly in Digipoly which may be recorded into a high-quality tape player and loudspeakers. This means that a further low-pass filter, for example on a mixing desk, has to be added before the sine wave can be used for other parts of the circuit. The output signal-to-noise ratio is excellent.

Owing to the note-generation technique, synthesis of high-frequency complex waveforms above about 1kHz is not good. Using higher frequencies, non-consecutive samples from the waveform table are taken with gaps of several samples. This causes small features in the waveform to become aliased, thus introducing the characteristic ringing spurious frequencies into the sound. Remember though, that the true harmonics that these small features are supposed to add to the output would be out of the audio spectrum and the desired sound can be achieved by using a simpler waveform.

Figure 6b in the September issue shows intermodulation product measured when two sine waves are produced by holding down two keys at once. This product is presumably created in the 741-based active filter used in Digipoly's output circuit since the digital circuit behaves totally linearly. Figure 6c shows harmonic spectrum from a single squarewave.

Performance and versatility of Digipoly are very good. The instrument is capable of producing several well-known organ sounds, a good Fender Rhodes sound, good chim and bell sounds and a large number of rich, raucous synthesizer sounds.

As usual with keyboard instruments, adding a few effects such as flanger and chorus can add a lot to the sound and the MIDI interface puts the instrument among the best as far as interfacing goes.