Convert all your synchro channels

to digital with a single μP-based system. For low speeds, you get an accurate, inexpensive, efficient and small unit.

Using a microprocessor, you can convert eight synchro or resolver channels to digital angles with four-arc-minutes accuracy in a package roughly the same size as a single-channel converter system—with only a bit more power consumption and at nearly the same cost. For example, in a complete modular system, the MN7200 from Micro Networks, a microprocessor controls sequencing and performs data conversions. The result is a converter capable of handling up to eight synchro or resolver inputs, whose accuracy is guaranteed over the full operating temperature range of 0 to 70 C.

There is one drawback, however. The μP-based instrument is limited to maximum transducer speeds of 10 deg/s.

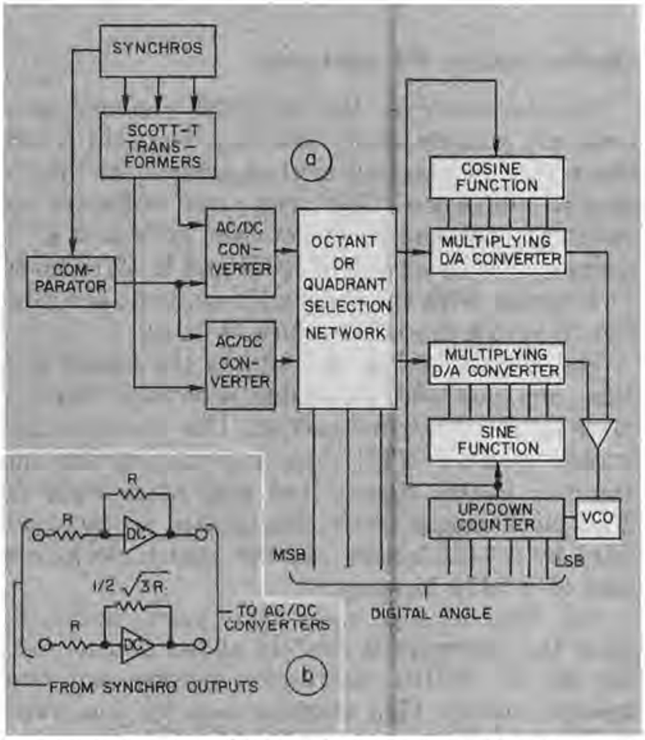
Synchro/resolver converter basics

The conventional converter system (Fig. 1) requires six distinct steps to convert either a synchro or resolver signal into digital angles:

- Converting the three-phase synchro signal into a two-phase resolver signal with either a Scott-T transformer or op amps. One phase of the resolver signal is proportional to the angle's sine, and the other phase to the angle's cosine.
- Demodulating both these signals with respect to the synchro's ac reference signal with an ac/dc converter, to get dc levels proportional to the sine and cosine of the input angle.
- Detecting, from the dc signal, the quadrant or octant in which the input angle lies, and digitizing the angle. Table 1 shows the bit weights for a binary sequence of angles.
- Feeding the binary-angle information to an up-down counter.
- Multiplying the counter's output by the dc signals with two multiplying d/a converters—one for the sine signal and one for the cosine signal—to get two results: sin θ cos ϕ, and cos θ sin ϕ, where θ is the angle in the counter and ϕ the input angle.

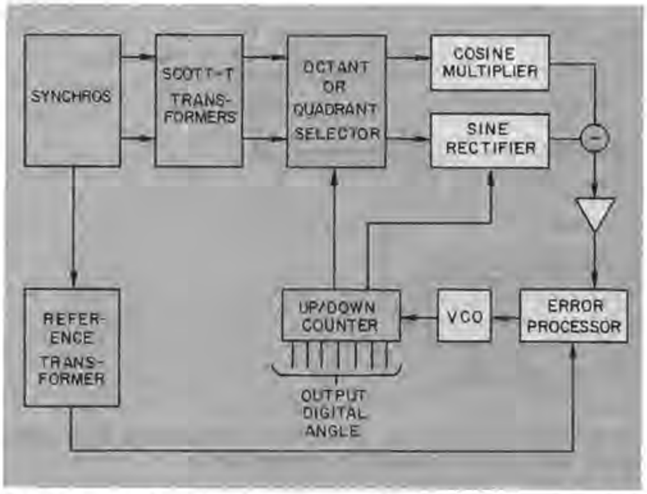
Binary weights of angles in the coding method employed in the MN7200

Bit number	Degrees	Degrees, minutes		Radians
1	180	180	0	3.141593
2	90	90	0	1.570796
3	45	45	0	0.785398
4	22.5	22	30	0.392699
5	11.25	11	15	0.196349
6	5.625	5	37.5	0.098175
7	2.8125	2	48.75	0.049087
8	1.40625	1	24.38	0.024544
9	0.70312	0	42.19	0.012272
10	0.35156	0	21.09	0.006136
11	0.17578	0	10.55	0.003068
12	0.08789	0	5.27	0.001534
13	0.04395	0	2.64	0.000767
14	0.02197	0	1.32	0.000383
15	0.01099	0	0.66	0.000192
16	0.00549	0	0.33	0.000096



- Subtracting, in an op amp, to get the difference between the two converter outputs, and adjusting the frequency of a gated voltage-controlled oscillator (VCO) with the result. The VCO's output pulses the up-down counter to the correct digital angle—the digital equivalent of nulling in an analog system. When the counter contains the exact input angle, the subtraction result is zero, and the process is complete.

The disadvantages inherent in this six-step process are obvious. Too much circuitry is required, principally for quadrant or octant selection and angle determination, and analog signals are carried too far—right up to the final counter.



2. **The popular tracking converters** are accurate but limited to converting only a few channels. Successive-approximation sampling converters are useful for converting several channels in a multiplexed system, but have serious accuracy problems.

Now there are three

The new μP technique competes with two other methods for converting synchro and resolver signals to digital angle data that, hitherto, have been used. These older types are the tracking converter and the successive-approximation sampling converter.

The tracking converter (Fig. 2) is noted for its high accuracy even with noisy signals. Noise cancels out because of ratio rather than amplitude detection. But, even though some recently available s/d and r/d tracking converters have improved operating speeds, most tracking converters can't track a synchro or resolver rotating at more than four rev/s. On the other hand, the successive-approximation sampling converter, although a higher-speed circuit than the tracking converter, becomes inaccurate with noise or distortion.

curately, and sampling converters when six or more multiplexed channels are to be converted at high speed.

The third method for s/d conversion, the microprocessor-based system, as used in the MN7200 is intended for multichannel applications. Accuracy (even in the presence of noise), low cost, small size and low power consumption make the μP-based system attractive for many applications, in spite of its low speed.

The new circuit (Fig. 3) is made up of eight dual ac/dc converters, a 16-channel data-acquisition system, and a microprocessor.

The microprocessor, a Fairchild Semiconductor F8, is a two-chip unit that consists of the CPU, and the Program Storage Unit (PSU). Besides being μP-based, the MN7200 circuitry differs from conventional s/d and r/d circuits because it has a minimum number of custom-linear circuits and, thus, is cheaper. Octant selection (0 to 45 degrees) is performed digitally with the microprocessor. And the hybrid circuits used, such as the ac/dc demodulators, multiplexers, and a/d converters are all standard products.

The system converts eight resolver (or with the addition of Scott-T transformers, synchro) channels into 14-bit digital form.

Multiplexing the inputs

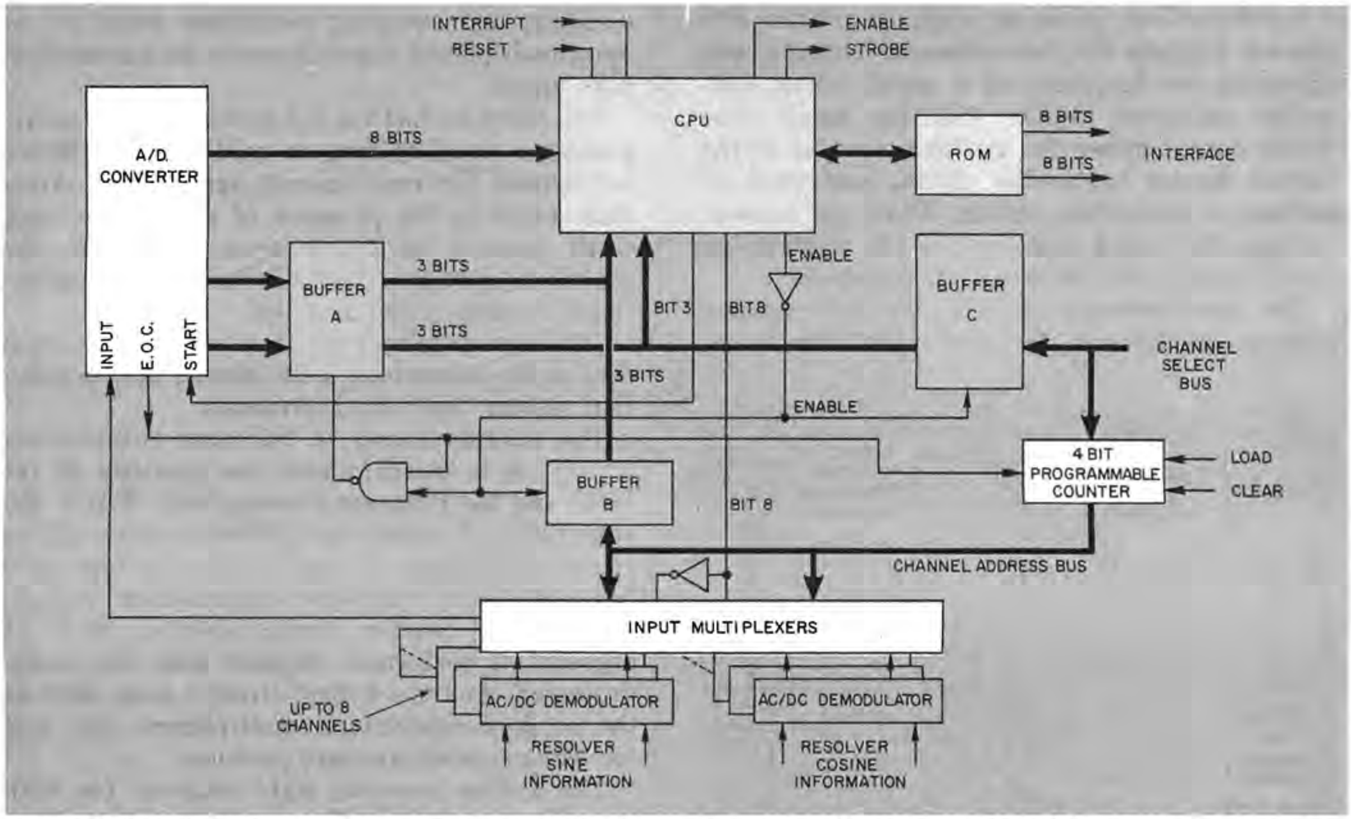
While conventional multichannel converters typically require an a/d for each channel, only a single a/d is used in this circuit: It accepts up to eight pairs of sequential inputs from the multiplexer.

Synchro outputs from the Scott-T transformers or resolver outputs V_x ac and V_y ac are converted to dc voltages V_x and V_y by two ac/dc demodulators—one each for sine and cosine (Fig. 3). A multiplexer connects these dc signals sequentially—one per conversion period (approximately 2 ms)—to the a/d converter that converts them into the binary signals X and Y. After conversion, the binary X and Y signals are stored in RAM. The microprocessor then executes the conversion equation,

$$\theta = \tan^{-1} \frac{Y}{X},$$

and the result is placed on the output-data lines and stored in RAM. Up to 8 channels are thus converted and stored in 16 RAM locations. A conversion cycle takes 2 ms.

A fetch cycle takes 50 μs from the time the data-output line is triggered. The system has two externally controlled channel-selection modes. In the sequential mode, a counter is incremented after each channel is converted. In the random mode, each channel may be selected by the chan-



3. **The MN7200 resolver-to-digital converter** circuit differs from conventional ones in that it allows for low-cost multichannel (eight, in this case) conversions in a compact, low-power-dissipating configuration.

r/d converters in that all its data are converted to digital form at the input a/d converters, whereas much of the information in conventional designs remains subject to error in analog form, right up to the final digital counter.

Canceling nonlinearities

The 12-bit a/d converter in the circuit (Fig. 3) performs both sine-proportional and cosine-proportional signal conversions, so errors caused by using separate a/d's are eliminated. By ratioing, the microprocessor eliminates the potential error source due to amplitude changes of the sine or cosine-input signals.

Two data I/O ports connect the 12-bit a/d converter's output and the microprocessor's CPU. Two 8-bit ports deliver 12 bits to the CPU in 2 bytes. One port inputs eight bits, the other only four.

With buffer A on and buffers B and C off, a/d data goes to the CPU. With buffers A & C off and B on, the address of the channel being converted is sent to the CPU. When operating in the interrupt or data-fetch modes with C on and A and B off, an external address can be sent to the CPU to address the memory for data.

In addition to the data bus that links the PSU and the CPU, two other 8-bit I/O ports are used This arrangement allows simple interfacing with either 16-bit minicomputers or 8-bit microcomputers.

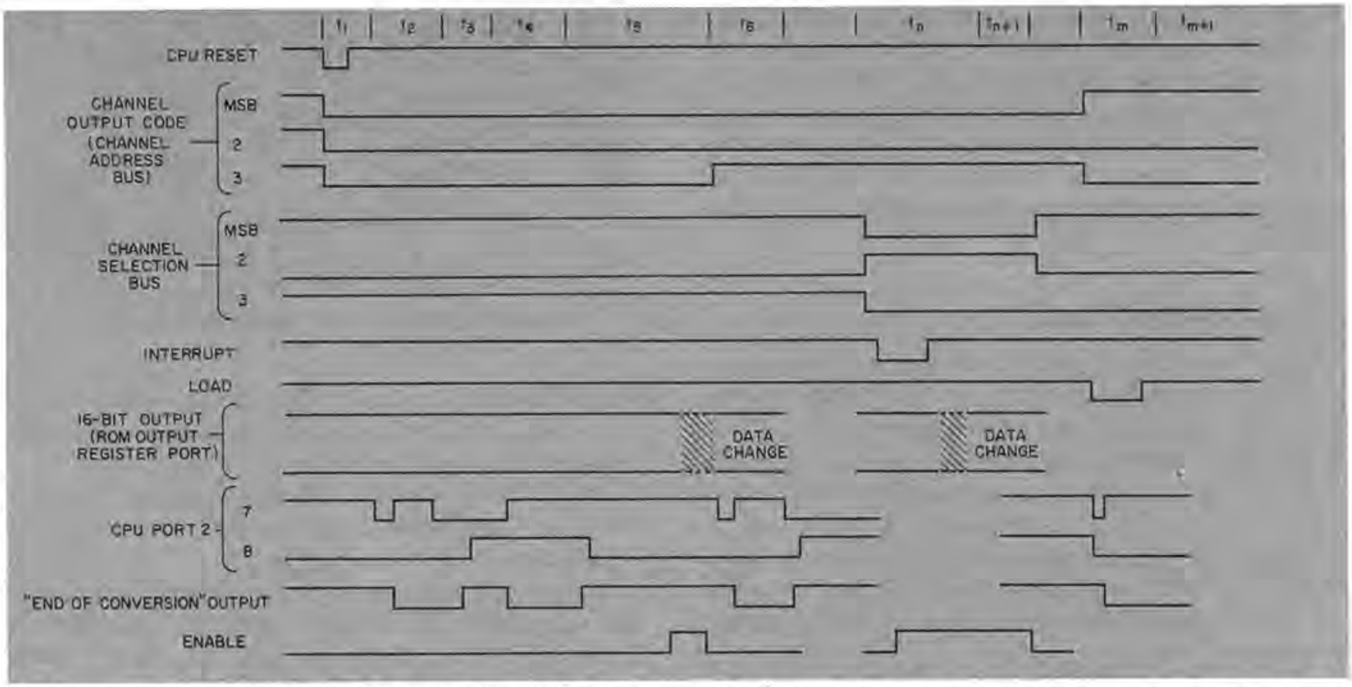
After the synchro or resolver angle is computed, the microprocessor system stores the resulting 14-bit word in the registers of the PSU's two I/O ports, until the word is replaced by a new result.

The 4-bit programmable counter, which addresses the input multiplexers sequentially, is connected to the channel-select bus to receive direct-set inputs from an external channel-selector circuit. The 3-bit, channel-address bus links the input multiplexers to the programmable counter and the CPU via buffer B.

It all starts with reset input

Operation begins when a reset input initializes the CPU. The system is clocked by bit 7 of the CPU's I/O port, which drives the start input of the a/d. The a/d then outputs EOC (end of clock), which drives the programmable address counter. Bit 8 enables the input multiplexers to pass the channel addressed by the programmable counter. In all, 6 bits are used for data entry to the CPU.

The multiplexer's sine inputs are enabled by bit 8. Its complement $\overline{\text{bit 8}}$ enables the cosine in-



4. **The converter's six timing states** are repeated as it cycles through each of eight channels. For fewer than eight channels, the last active channel is followed by channel 0 and the sequence repeats.

The CPU's enable-output line indicates that an interrupt signal has been received or data are changing. Normally, this line (low) disables buffers B and C. Buffer A is enabled after the a/d conversion is finished (EOC high). (An enabled buffer A permits data transfer from the a/d converter to the microprocessor's CPU.)

After an interrupt signal, the enable goes high, which activates buffers B and C and disables buffer A. The interrupt stops the normal operation of the microprocessor and requests that stored data from the addressed channel be transferred to the output lines.

The reset line to the CPU resets the program counter in the PSU. The load line allows the programmable counter to be set to the channel chosen by the channel-select data bus. The clear line resets the programmable counter.

Look at the timing diagram

Operation starts at t_1 with a pulse on the CPU's reset line (and the programmable counter's clear line). This pulse clears both the program counter in the ROM and the programmable counter to ZERO (see Fig. 4). The ROM's program counter is started by CPU clock signals. The channel-select line is still inactive, and all three bits on the channel-address bus are ZEROs.

The sine input of channel 0 then passes from its dedicated demodulator through the input multiplexer into the a/d converter.

At t_2, the a/d converter's start line and the blocks buffer stage A. During t_2, the EOC line clocks the programmable counter, but does not change the address to the multiplexers. This line changes only the LSB, which is not on the channel-address bus.

When the counter first accesses a channel, the address LSB is always ZERO. The multiplexer, therefore, always samples the selected channel's sine input first. The LSB is then toggled to a ONE, and the cosine input sampled. The LSB is then retoggled to ZERO. This time the 3-bit address is incremented, which accesses the next channel, starting with its sine input.

Also during t_2, the PSU's internal timer is set for a time interval slightly longer than the conversion time needed by the a/d. This timer stops the program counter in the CPU and restarts it at the end of the preset time interval. Meanwhile, the a/d converter completes its conversion cycle, and the EOC enables buffer A.

The a/d converter's 12 bits are routed in two parts—eight bits through data-port 1, and four bits through data-port 2. The first bit (MSB) indicates signal polarity. The remaining three bits are amplitude data and are stored in RAM by a program in ROM.

At t_3, bit 8 changes state, which both disables the sine multiplexer and enables the cosine multiplexer.

At t_4, conversion starts in the a/d and the EOC output blocks buffer stage A. The multiplexer connects the dc analog of the cosine input to the a/d converter and an all ZERO address is again concosine analog of input-channel 0 to digital form. At the conversion's end, EOC reverses the state of the buffers so that the cosine is at the CPU's I/O ports. At the end of the timer's interval, the program is restarted, and the cosine goes into RAM. Therefore, after t_4, values for both the sine and cosine of a channel's input angle are in the RAM.

A branch point can occur

Next, the μP computes the angle (answer) from its sine and cosine values. The MSBs of the sine and the answer are the same. For a ZERO MSB, the answer is between 0 and 180 degrees, and here the program has a branch point. If the sine's MSB is ZERO (sine positive), the cosine's MSB becomes the answer's next bit. For the sine's MSB, a ONE (sine negative), the complement of the cosine's MSB is the second MSB in the answer.

The answer's third MSB is determined by subtracting the sine's 11 amplitude bits from the cosine's. A positive result makes the third MSB a ONE, a negative result ZERO.

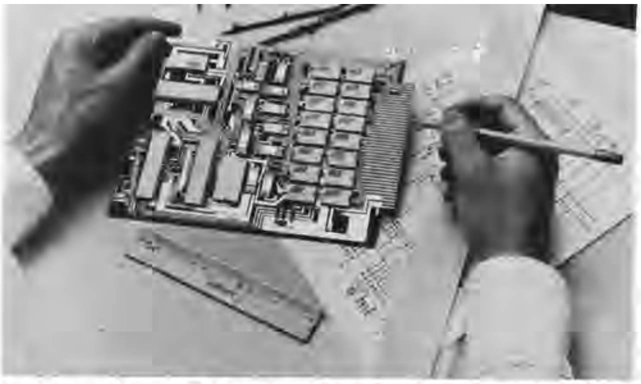
Now the μP can divide the larger 11-bit number into the smaller to get the angle's tangent. The ROM contains a tangent-to-angle table. To minimize the size of this table, only 64 values and the slopes to the next value are provided for the tangent function. Therefore, only the tangent's six MSBs are used to address the table; the remaining five bits are multiplied by the slope.

The final answer is the sum of two values; the first an 11-bit word that the tangent's six MSBs fetch from ROM. The second summed value is the product of the five remaining bits multiplied by a slope value that is also accessed by the six MSBs.

The enable line is held high to indicate changing output data. Buffers B and C are enabled, buffer A is disabled. The channel address, along with the program's instructions, select a pair of RAM addresses in which to store the 14-bit answer. The 14-bit answer also remains at the ROM's output register port.

At t_6, the programmable counter is clocked, which changes the second LSB of the three-bit address. The multiplexers pass to the next channel. The a/d converter starts, which begins the sequence for the next input channel. This process continues until the digital angles for all inputs are in storage, when the CPU's main program counter and the programmable counter are zeroed and started over again. Words stored in RAM are replaced by updated words as new information is processed and received.

The digital angle data are in binary angle form



5. **The entire μP-based r/d converter** consists of 32 DIPs on a 6 × 8-in. board.

which quarter, the next MSB indicates which octant, and so on. Code conversion, say, to binary or degrees is easily added here.

At t_n (Fig. 4), an interrupt signal inhibits buffer A, enables buffers B and C, and stops the ROM's program counter. The three bits that are now manually entered onto the channel-select bus address the RAM through buffer C and port 2. This 3-bit code addresses one of eight 14-bit angles in the RAM. The addressed angle appears at the ROM's output-register port. At the end of the interrupt, the main program counter picks up again, and the program continues from where it was interrupted.

At t_m, a load signal jams the address on the channel-select bus into the programmable counter. The CPU's reset line then resets the CPU and clears the main program counter to ZERO. The program then picks up by using the channel selected on the channel-select bus as the first channel, and proceeds in sequence thereafter.

Interrupt, therefore, provides the latest stored data for a selected channel, while load and reset cause new data to be processed starting with the selected channel.

The Scott-T transformers and ac/dc demodulators in the input circuit are conventional components. So are the IC multiplexers. The 12-bit a/d converter needs only a parallel output, something available in a variety of commercial components. Three-state buffers A, B, and C are merely used as on-off switches in the data paths to permit time-sharing of the CPU I/O ports. While the programmable counter is shown in Fig. 3 as a 4-bit unit, a 3-bit unit is adequate. The fourth bit is used to divide by 2, thereby maintaining the same channel address as the sine and cosine input multiplexers are successively enabled.

In the two-chip F8 microprocessor, one chip houses the CPU (including the control logic for RAM and ALU) and clock-generating circuitry, and the other chip, the PSU, houses the ROM. For a description of the F8 μP see "Microprocessor Basics, Part 3" (ED, No. 12, June 7, 1976, p. 126). A look-up table, for converting the tan-