ELECTRONOTES 103

NEWSLETTER OF THE MUSICAL ENGINEERING GROUP

1 PHEASANT LANE
ITHACA, NEW YORK 14850

VOLUME 11, NUMBER 103 JULY 1979

GROUP ANNOUNCEMENTS:

CONTENTS OF EN#103

In this issue, we discuss applications of the Vocal-Effects Waveform Animator (VEWA), pronounced "Vee-Waa" which was presented last issue. We also have two rather diverse contributions from Hal Chamberlin that we are sure our readers will find interesting.

NEWS AND NOTES:

In EN#99, pg. 3, the equation directly below the diagram should be:

$$V_1 = -V_{in} + Q[V_1(-1/s) + V_1(-1/s)^3] - R(V_1 \frac{1}{s^2}) - V_1/s^4$$

the error being the missing V_1 in the last term on the right. Thanks to Ken Reeves for picking up this error. Also, on page 12 of the same issue, in the diagram as viewed with the writing right side up, in the upper right corner, below the op-amp with its output marked "LP", there is an unmarked resistor. It should be 30k. Going way back to EN#72, on page 13, the diagram for the variable-slope filter, all the CA3080's, A13 through A18, have their (+) and (-) inputs reversed. It is the (+) input that should be at the top. Thanks to Lester Ludwig for catching this one.

A recent issue of J. Acoust. Soc. Amer., Vol. 66, No. 1, July 1979, has a paper by A. J. M. Houtsma on pp 87 - 99 titled "Musical Pitch of Two-Tone Complexes and Predictions by Modern Pitch Theories." In reading this paper I came across a reference to a paper by Gerson and Goldstein in a 1978 JASA paper which appears to present results and make suggestions similar to those presented in this newsletter EN#94 and EN#100. From the volume number, this would be a first-half 1978 paper, so their

[News and Notes continues, pg. 12]

APPLICATIONS OF A VOCAL-EFFECTS WAVEFORM ANIMATOR:

THE SYNTHESIS OF "ANIMATED" SOUNDS - PART 4 (ADDENDUM):

-by Bernie Hutchins

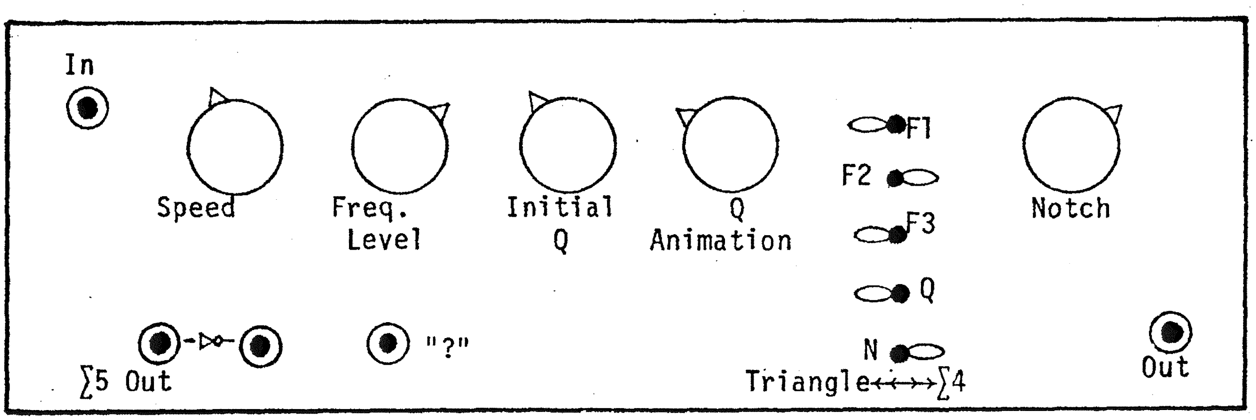
INTRODUCTION TO APPLICATIONS: The "Vocal Effects Waveform Animator" (VEWA) was described in its circuit details in EN#102. Here we will be looking at applications. The VEWA should be considered as basically a one-input, one-output device, with a few panel controls. In addition, of course, there are some other outputs for internally generated "summed LFO" signals, but these are output as a matter of convenience and economy - they could just as well be coming from another device.

We should start by saying what happens when you input a waveform, listen to the output, and then start turning a few knobs. First let's suppose we are inputting a series of sharp pulses at about a 1 Hz rate, and that the Q control is set in the medium to high region. The result is a ringing effect of three filters moving relative to each other. Thus the relationship of the three component rings is constantly changing, and also the ring frequency may be heard to change slightly.

Next, you make one change only - you turn the input frequency up into the audible range, say to about 100 - 150 Hz. A dramatic thing will happen. You will get the impression that the device has begun to talk. This impression will not last for more than a second or two, because after that, you will realize that the device is only producing vowel-like sounds, no fricatives (noisy sounds like "s" or "f") or any of the other inbetween sounds. Even though it is not "talking", you will not tire of playing with the device for some time, particularly if you feed back one of the summed LFO line to the driving VCO so that the device no longer is restricted to a monotone.

The general effects of the panel knobs can be noted. With the speed control in its slowest position, the formant motion can actually be made to stop, and in the maximum position, the formants are moving so rapidly that you may just start to lose tracking (sidebands just start to appear). Inbetween, there is a middle ground acceptable for a "normal" sort of speech. The frequency level control raises or lowers the formants above or below a "normal" level. While there might be some question as to whether the center of the range is normal, there is little question that the low range is somewhat muffled while the upper range is somewhat "Donald Duck." Thus, we have some evidence that the central region is in fact "normal." The Q control, as you might expect, has a dramatic effect. There is some loss of apparent energy as you change to higher Q, perhaps indicating that the constant peak idea is not a totally good idea (but probably necessary, to avoid clipping at times), but the main effect is to change from a sort of muffled speech (low-Q) to a "tinny" speech (at high Q) with a "normal" region inbetween. The Q animation control has

Fig. 1



Panel Diagram of Vocal Effects Waveform Animator a much more subtle effect, but this is probably because the effect is somewhat hidden in an overall mass of animation (formant positions as well as Q). The notch position and overall notch effect is somewhat more subtle, with the moving notch being the most essential feature.

A lot could be said about the five switches that control the selection of a triangle or a summed triangle LFO. First, there is a big difference, and the main one is that the summed LFO's tend to average each other out so that extreme values tend to be rare. Consequently, in the design we added a gain of two to the summers so that their extreme values are twice that of the triangles (±15 instead of ±7.5). This helps quite a bit. There is no real difference in "speed" between the two choices, even though there is more activity in the summed LFO case. Note that in the summed LFO case that when one of the LFO's rises for example, a rising component is added to all those to which this one is summed. Thus there is a degree of correlation between the various sums that is lacking in the independent case.

"TWO-TERMINAL" APPLICATIONS

We will begin our discussion of specific applications with uses of the VEWA as an in/out device, allowing for the possibility of changing some of the panel controls. Fig. 2 shows a "Pinger" type of application, the same as that discussed in the introduction. Fig. 3 shows a "Windsound"generator, Fig. 4 a "Rhythm Driver" and Fig.5 shows the more or less standard animator application:

Fig. 2 Complex Pinger

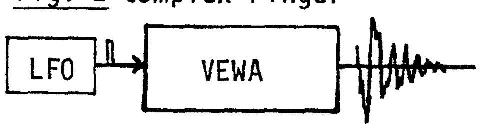
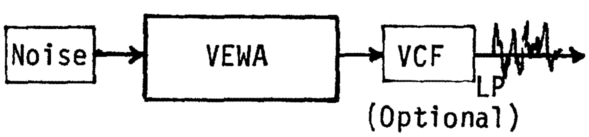


Fig. 3 Windsound Generator



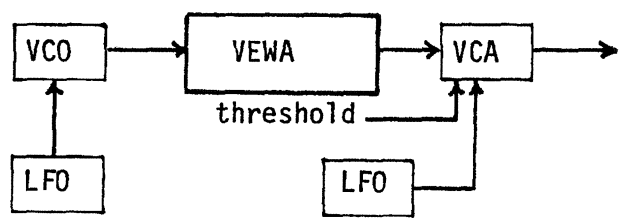


Fig. 4 Rhythm Driver

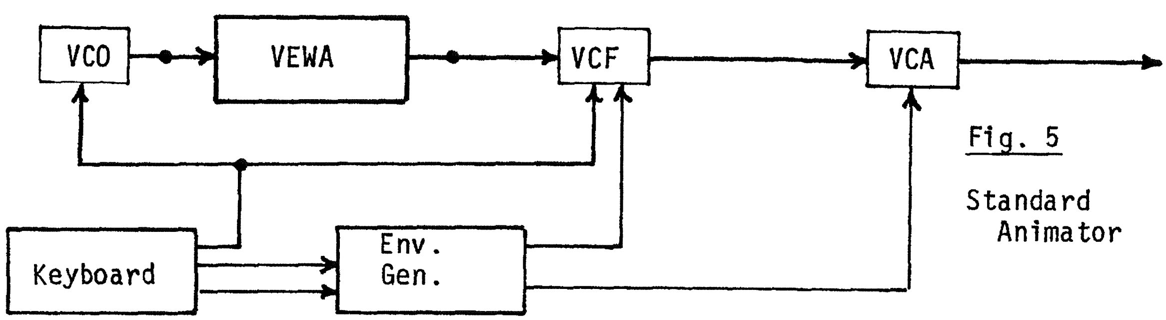


Fig. 5

Standard Animator

The "Complex Pinger" of Fig. 2 has been discussed in the introduction, and a similar setup is shown in Fig. 3, except now the VEWA is excited by a noise source rather than a pulse LFO. The effect is one of the best windsounds we have ever come across. You get not just the roar and whistle of the wind, but a rise and fall of the whistles. For this application, you will set the speed control to a very slow value, the frequency level control as low as possible, and the Q moderate to high with little if any animation to the Q. The switches should all be set to the ∑4 positions since we want no regular paterns in the wind. We also show a VCF as a LP at the end of the patch, and this while not absolutely necessary is a nice addition. It serves two functions. First, with the formants lowered by the frequency level control, the fixed formant at 3700 Hz tends to stand out too much, and it does not move. The VCF LP is used to cut this off to a degree. Secondly, the VCF is a very natural way of fading the wind in and out, much nicer than just letting the amplitude go up and down. All and all, it's a nice natural windsound that you will find interesting for quite a while.

Fig. 4 shows what I call a "Rhythm Driver", and what I have in mind here is a unit that produces a repeating pattern that could be used as a background of a composition or performance. Unlike other rhythm pattern generators that produce beats that stand out by accent or changes of tone color, this one produces individuality by repeating what we might call "mini-words." Perhaps something like AHH, EEE, OHH, UUU, EHH, AHH, EEE, OHH, UUU, EHH, AHH,. . . . and so on, a pattern of grunts and groans if you prefer. To get this exactly right, you have to have the switches in the triangle mode rather than the $\sum 4$ mode, and some effort must go into setting controls to values close to exact ratios. Probably the most important coordination is between the two LFO's shown specifically in the patch, and the triangle of F1, the first formant. The other formants are allowed to vary, and this, along with the inexact settings in the first place, means that the pattern, while repeating, will also be changing in some subtle ways. The possibilities are somewhat endless.

In general, when we think of an animator, we have in mind a device which can sit in a standard patch between the VCO and the VCF, and this is just what we have in Fig. 5. Here the VEWA is enriching the sound out of the VCO, and probably you will want to have the speed control set fairly slowly here for most applications where you would otherwise use just a fixed waveform from the VCO. Also, the Q will not be too sharp. Faster speeds and sharper Q's in the same patch will probably be useful for special effects. In particular, a very nice "bubbly" undercurrent can be achieved.

APPLICATIONS WITH SUMMED LFO FEEDOUT

The sum of all five LFO's ($\sum 5$) has been made available along with its inverted version, and this can be used to animate effects outside the animator. Most importantly, the input pitch levels and output amplitude levels can be controlled. Also, it is possible it alternately input noise and pulses, thus giving a sort of voiced/unvoiced speech effect. Two possible patches are shown in Fig. 6 and Fig. 7.

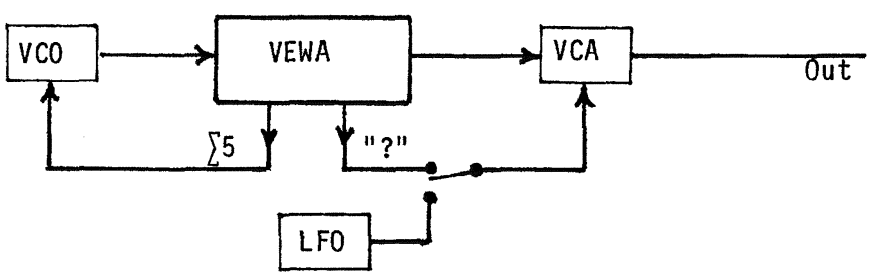


Fig. 6

Voiced Talker

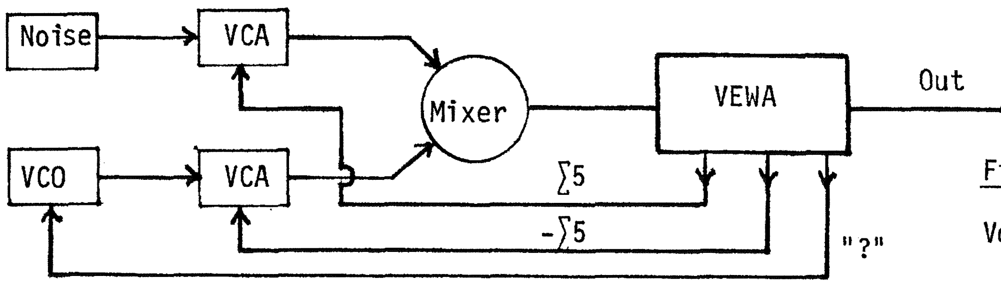


Fig. 7

Voiced/Unvoiced Talker

Both Fig. 6 and Fig. 7 are pretty good at giving a first impression of a talking person. While you might expect that Fig. 7 would be better, initial experiments show that the "fricative" part from the noise source is not at all natural and can be very distracting. Thus, it can be considered as being different from Fig. 6, and not necessarily an improvement. It must be emphasized however that the excitation of the talker with noise only is very interesting, it is the mix that there are problems with.

Fig. 6 shows a VCO signal being processed by the VEWA. For the moment consider the VCA to be out of the circuit and the $\Sigma 5$ line out of the VEWA to be disconnected. Thus the VCO is held at one fixed frequency, while the formants in the VEWA of course continue to move. The result is, logically enough, a simulated monotone, not unlike the sort of "robot voice" we often see in science fiction movies and TV shows. The "robot" of course does not say anything here - it just sounds as though he is trying to say something. Now, when we connect the $\Sigma 5$ line to the envelope input of the VCO, and start to turn up the associated control sensitivity, the pitch begins to change. Now the monotone is gone, and we have something more like speech. Next we look at the VCA, and see that we either feed it a "?" or a LFO. Clearly the LFO can be used to make the machine say a series of evenly spaced "words". This can be interesting, and can be similar to the "Rhythm Driver" (Fig. 4) if the switches are in the triangle positions instead of their $\Sigma 4$ positions. With the switches in the $\Sigma 4$ positions, the series of "words" does not repeat as it may in the triangle positions. What about the "?" output? This is another summed LFO signal, and we might suggest a weighted sum of all five LFO's, perhaps even parts of 1, 3, 5, and half of 2, and a quarter of 4. With the "?" fed to the VCA, we get random length words. The "duty cycle" of the talking vs. silence depends on the threshold level of the VCA.

Fig. 7 shows a similar setup where either a pitched signal from a VCO (as controlled by the "?" LFO sum), or random noise, may be entered into the VEWA. It is possible to adjust the threshold levels to the VCA's so that there is also a mix of VCO and noise drive at the same time. Note that one VCA receives the $\Sigma 5$ signal while the other receives $-\Sigma 5$, the inversion, so in general, when one VCA is on, the other will be off, or at least reduced from its full amplitude. This is a sort of classical "Voiced/Unvoiced" dichotomy, and is somewhat artificial in even the live speech case. Thus, we can not expect this to be too successful, and it is not. It is an interesting sound, but the noise "fricatives" tend to last too long, and if we speed things up to shorten the fricatives, the formants change too rapidly. If we were really concerned with imitating speech accurately, we would probably need one set of faster LFO's to turn the noise on and off. This is beyond what we hope to accomplish here.

A special case of Fig. 7 is where the noise VCA is on all the time, and the VCO VCA is off all the time. The result is just a noise input to the VEWA. Since the formants of the VEWA are moving as always, we get a sort of whispered speech effect that can be very useful. Note that this assumes that the speed, frequency level, and Q controls are set up for a natural sounding speech in the VCO case before the VCO is replaced by the noise source. As noted above, when we lower the speed and frequency level, we get into the wind effect of Fig. 3.

CONCLUSIONS AND POSSIBLE FUTURE REFINEMENTS

The initial results with this device seem more than enough to indicate that the moving formant approach is very useful. In fact, there are probably as many useful applications per "unit" of time and money invested with this device as any we have presented in the past. It is highly recommended.

As we said above, this unit was designed with the idea of a vocal effects simulator, and we did seem to achieve this goal to some degree. It is however interesting to speculate as to whether we can do better with slightly different data, and if we can in effect, do worse - getting the general effects without any overt attempt at vocal sounds. Certainly there is a lot we can do with adjustable frequency ratios and ranges of the formants. Also, through no intention on our part, the speaker seems to be male rather than female. Of course, it would be nice for musical purposes, if for no other reason, to have both. Adjusting the formant levels and input pitch upward does not seem to move in the direction of a female voice. It is probable that this is due in part to bias in the input data toward male speakers, but it is also generally acknowledged that the synthesis of a female voice, taken as a general problem, is more difficult than the synthesis of a male voice. Probably many of our readers will build this device, and interesting experiments may result.

LINEAR APPLICATIONS OF CMOS LOGIC:

-by Hal Chamberlin

What would you do if I told you that for 5 cents per amplifier you could purchase a hex op-amp that is internally compensated with a gain-bandwidth product of 10mHz, has a slew rate of 50 volts/microsecond, an input bias current less than a nanoamp, and it would do all these things on a single 5 volt power supply! Before you rush out and buy stock in my company you should know that there are disadvantages too. However before getting into all of the pros and cons, lets examine the device itself.

The "op-amp" I'm speaking of is the common CMOS hex inverter package known as either a 74C04 or a 4069. It sells for 20 to 30 cents and includes 6 well balanced logic inverters using fairly large transistors for a reasonable output drive capability. Each inverter is made simply from two complementary MOS transistors as shown in figure 1. A MOS transistor acts just like a regular JFET of the same polarity except that the gate bias - drain current curve is shifted to the right as in figure 2. In technical terms it is an <u>enhancement</u> mode device whereas the JFET is a <u>depletion</u> mode device but for now we will just use its characteristic curve. A P-channel MOS transistor has an exactly complementary curve as in figure 3. When the two transistors are connected in series opposing such that the drains are tied together and the gates are tied together, we have the classic Complementary Metal Oxide Semiconductor inverter or CMOS for short.

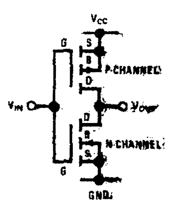
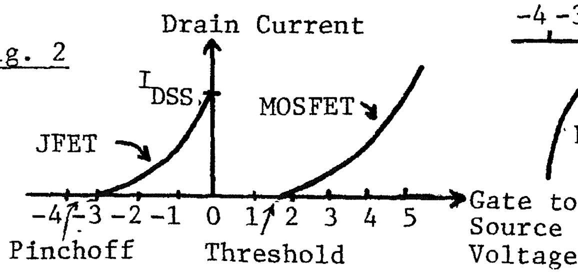


Fig. 1

Fig. 2



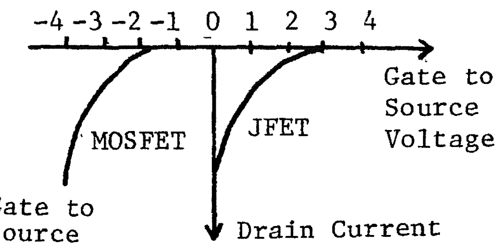


Fig. 3

It is interesting to plot the input/output voltage relationship of this structure. When the input voltage is close (within 2 volts) of either supply, the transistor connected to that supply is essentially cut off since its gate-source bias is less than its threshold voltage. Conversely, the opposite transistor is driven on fairly hard so the output voltage swings to the opposite supply voltage. This is the saturated state of the device and of course is the normal mode of operation for logic. When the input voltage is such that both transistors conduct, the output voltage will settle to some value between the supply voltages. The value it lands on depends on the relative channel conductivities (resistances) of the transistors at their respective bias voltages and is a sensitive function of the actual input voltage. Examination of the structure reveals that the lesser turned on transistor acts like a "load resistor" for the greater turned on device. If the transistors are matched (as they are in the 74C04 and 4069 devices), an input voltage midway between the supply voltages will yield an output voltage midway between the supplies as well. Figure 4 shows the input-output transfer function for different power supply voltages. Note that the function is considerably rounded for the higher voltages. This is because the device acting as a load resistor becomes very low in value and since the other transistor is acting as a common source amplifier (high output impedance), its gain becomes rather low. Figure 5 gives the gain-frequency characteristics at several supply voltages. Note the smooth 6dB per octave rolloff which implies good stability in feedback amplifiers.

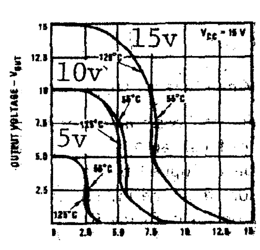


Fig. 4

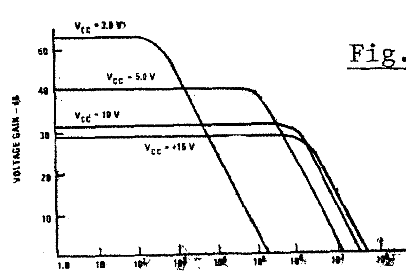


Fig. 5

By examining these curves we can draw several conclusions about the CMOS inverter used as an op-amp. First, the optimum power supply voltage is about 5 volts. Higher voltages are useful but power dissipation can become excessive so our discussion will be confined to operation on a single 5 volt supply. While restriction to a 5 volt supply may seem to be a prohibitive disadvantage, it is getting to the point in microprocessor systems where 5 volts is the only supply voltage available. If only a small amount of analog circuitry is needed in the system, it would be a great advantage if it would run from the same voltage (perhaps filtered a bit). The only alternative would be additional power supply circuitry or a noisy and expensive DC-to-DC converter.

The other apparent disadvantage is that the DC and low frequency voltage gain is only about 40dB (gain of 100). However the 10mHz unity gain frequency means that at frequencies above 10kHz it actually has higher gain than garden variety op-amps such as the 741 or LM307. In particular the open loop response is flat out to 100kHz whereas the internally compensated 741 is flat only to 10 Hertz open loop. This coupled with a high slew rate of 50 to 100 volts per microsecond means that transient intermodulation distortion does not occur in audio applications. Thus over the full audio range the CMOS amplifier actually has a higher gain than many general purpose op-amps.

Since the transfer curve is somewhat nonlinear near the extremes of output voltage, audio applications should be limited to about a 2.5 volt peak-to-peak swing or from 1.25 volts to 3.75 volts. Also feedback should be used to minimize distortion unless the swing is restricted even more. Even so, distortion figures of considerably less than 1% are possible which is not bad for a 5 cent device.

Perhaps the design task that requires most attention is biasing the device. For op-amps connected to bipolar power supplies, biasing is so simple that it is probably taken for granted and ignored. When operating from a single supply (no matter what kind of single supply amplifier is being used), biasing requires more attention. The rule for the CMOS inverter op-amp is that the input bias voltage will be very close 2.5 volts for linear operation. Although the N-channel and P-channel transistors are reasonably well matched, there will be a finite offset voltage. This means that a precise 2.5 volt input will not generate a precise 2.5 volt output. A good way to view this property is to imagine a non-inverting input internally connected to 1/2 the supply voltage. The offset voltage can then be defined just as with an ordinary 2-input op-amp. Thus an input offset voltage of 10MV would yield an output offset of 1 volt assuming an open loop gain of 100.

What this boils down to is that feedback from the output is necessary to stabilize the input bias, just as with any op-amp. For audio applications it is desirable (and usually possible) to have 100% DC feedback for bias stabilization while the AC feedback factor is much less as in figure 6 below. Pure DC amplifiers can also be constructed but the offset voltage restricts their closed loop gain to about 10 as shown in figure 7. Actually, the CMOS amplifier can be used successfully in just about any standard op-amp circuit in which the non-inverting input is grounded. If one looks through back issues of Electronotes and manufacturer's op-amp application notes it turns out that over one half of these circuits don't use the non-inverting input.

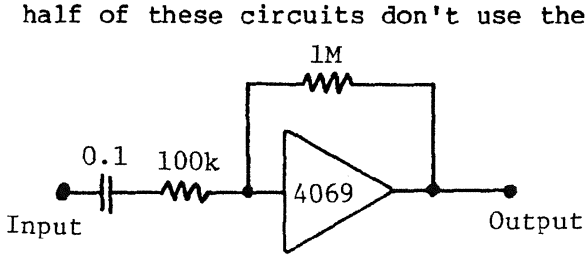


Fig. 6 Gain of 10

AC Amplifier

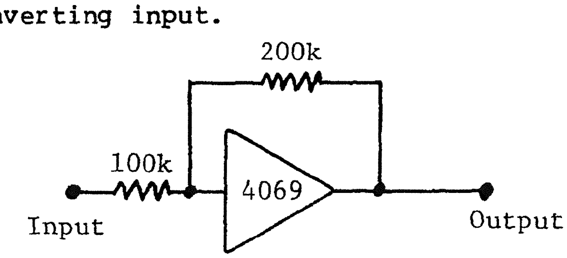


Fig. 7 Gain of 2

DC Amplifier

Probably the most advantageous application of linear CMOS amplifiers is in active filters. In figure 8 below is shown the three standard multiple feedback active filter circuits as applied to CMOS amplifiers. Note that the bandpass and highpass circuits both provide 100% DC feedback from output to input thus completely stabilizing the bias point. If the lowpass is designed for unity DC gain it too is well stabilized. These circuits are quite attractive when using more expensive standard op-amps but they do have drawbacks. Probably the worst is that when the design equations are solved one is likely to wind up with unequal and weird values for the capacitors, or at the very least, all of the resistors will be odd values. The other drawback is that high Q factors require very high gains. This restricts the Q with CMOS amplifiers to about 5 but don't forget that even standard op-amps will be restricted to low Q at high frequencies.

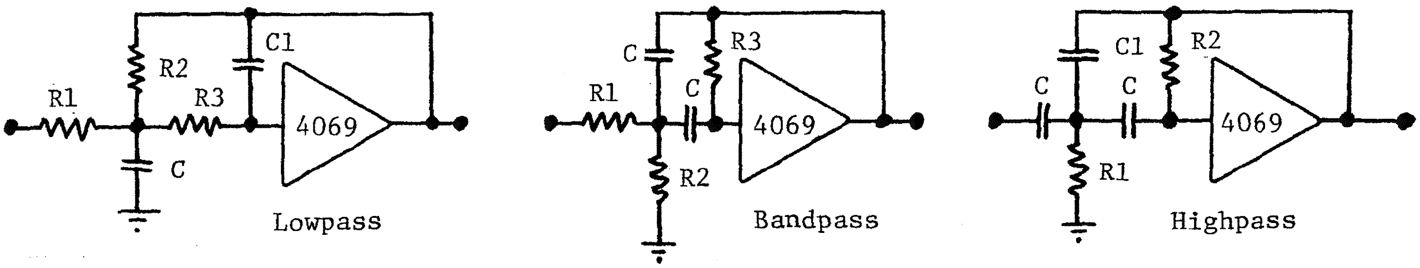
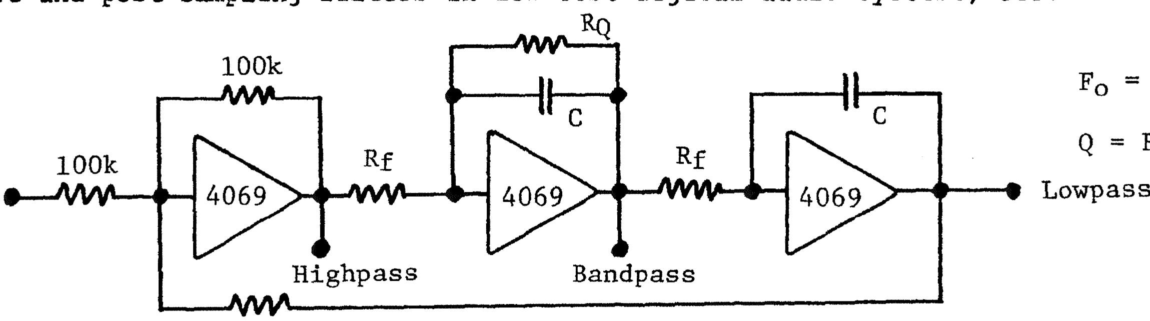


Fig. 8 Multiple Feedback CMOS Filters

As most of us know, the state variable or bi-quad filter has a list of advantages three feet long but is not used for plain filtering because it needs so many parts, particularly op-amps. The CMOS bi-quad filter in figure 9 below however is quite economical and overcomes all of the problems of the multiple feedback filters above. In particular, equal value capacitors can be used for widely varying cutoff frequencies and Q factors such as are encountered in multiple section Chebyshev filters. Since the response parameters are only affected linearly (first power rather than squared or worse) by shifts in component values, standard 5% resistor values may be used with confidence that the response error will be no larger than the component error. High Q's are easily attained as well and in fact if the damping resistor (R4 in the diagram) is removed altogether, a reasonably stable Q of several hundred is obtained. Since good stable filters are now so cheap, there is no reason why dozens of them can't be used for synthesis and analysis filter banks, 1/3 octave equalizers, real-time spectrum analyzers, pre and post sampling filters in low cost digital audio systems, etc.



$F_o = 1/2\pi R_f C$

$Q = R_Q/R_f$

Fig. 9 CMOS Biquad Filter

Figure 10 shows a 5 volt CMOS audio power amplifier that is cheap to build and outperforms by a wide margin all of the monolithic power amplifiers I've tested so far. The output will swing about 4.5 volts into a 4 ohm load yielding a continuous sine wave power output of about 600MW; quite adequate for a personal computer or arcade game. The reason for the wide output swing is that both output transistors are operated common emitter so the only voltage loss is the collector saturation voltage which is about .25 volt. No bootstrap circuit is needed to get a high positive swing. Note also the local feedback loop in the output stage and the overall feedback loop from output to input. Three CMOS gates in parallel are needed to adequately drive the bias network. In volume the parts come out to about 60 cents. The NSPU01 and NSPU51 output transistors are made by national semiconductor and contain a 1 amp chip in a plastic TO-18 package. An MPSU01 and MPSU51 (Motorola) will give you the same chip in a large power transistor package for more money.

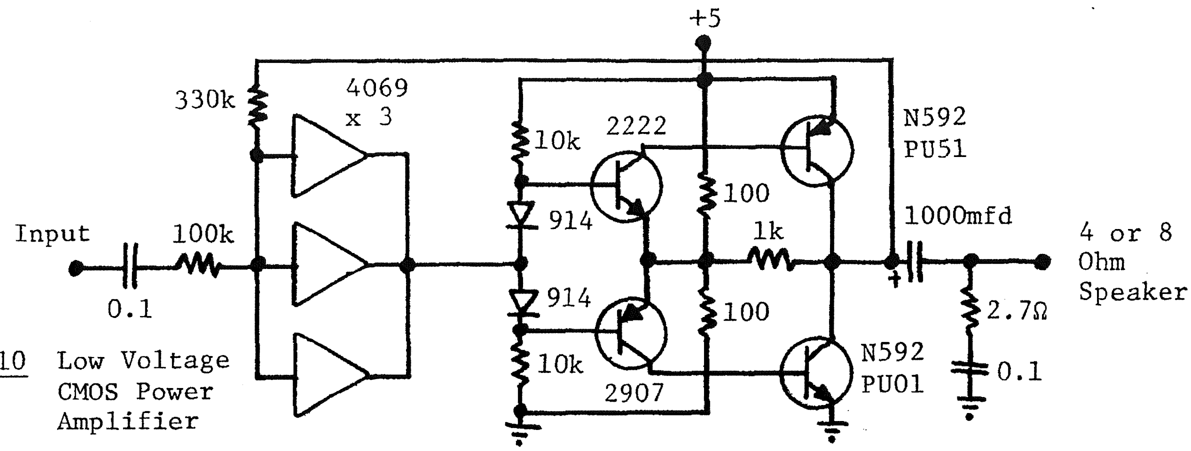


Fig. 10 Low Voltage CMOS Power Amplifier

At this point the hard-boiled engineers are probably wondering if the characteristics of CMOS are well enough controlled to base volume production commitments on. There are indeed a couple of pitfalls but they are easily circumvented. One is that some CMOS gates are internally buffered. For inverters this means that internally there are actually three inverters in series. While the gain would theorectically be boosted to 120dB, phase shift would be excessive thus requiring compensation. Buffered devices always have a B in their part number although not all B parts are buffered. This should only be a potential problem with 4069's; as far as I know the 74C04 device is never buffered. Another possible problem is that power supply noise rejection is not very good compared with standard op-amps. Fortunately the current drain is low so a simple RC filtering network (such as 47 ohms and 470uF at 6 volts) is usually adequate. When designing bias and feedback networks it is important that the output load impedance be kept high if the gain is to remain close to the 100 mark. A minimum combined load of 20K ohms is a good rule to observe. One nice characteristic is that temperature doesn't seem to bother things much. National Semiconductor shows an offset shift of only a few tens of millivolts over a -55 to +125 degree C range. As far as volume production goes, my company has commited to build several thousand units containing a 6 pole Chebyshev CMOS filter and the power amplifier above and we havn't had any problems yet.

* * * * * * * * * * * * * * * * *

COURSE REVIEW - TECHNIQUES OF COMPUTER SOUND SYNTHESIS:

-by Hal Chamberlin

Frequently I have seen record reviews, book reviews, and equipment reviews in the pages of Electronotes but not any course reviews. Although many of us may not care to admit it, taking a formal course in some aspect of musical engineering can be of great value. Such value is not necessarily in the acquisition of some new circuit technique or system organization concept but rather in extended contact with the instructor, laboratory staff members, and particularly with other attendees. A friend wondered why I would shell out the tuition money when I could probably have taught the course myself. The answer is that other people, especially the composers taking the course, can have a profound effect in the way engineering aspects of electronic music are viewed. Besides, it was an opportunity to use the MIT studio as well as a stimulating if not restful two week vacation.

The course that will be described ran from June 18-29, 1979 and was given at the Massachusetts Institute of Technology. It was immediately followed by a 4-week course entitled Workshop in Computer Music Composition which was oriented especially toward composers. The first is an intensive course comprising 4 to 5 hours of lecture and about 4 hours of studio time daily. After adding in time for eating (MIT is close to some very famous Boston resturants) and travel (by foot), there is very little time left for anything except sleeping. Studio time was also available on the weekend. The second course (which I did not attend) is primarly studio time with the goal of completing a substantial composition for public concert at the end of the term. Instruction during the composition course is on an individual basis as needed. Unfortunately these courses are not cheap - $900 for the first and $1100 for the second - but scholarships are available. This is the second year they have been offered (that I am aware of) and it is likely that they will continue to be given every summer in the future.

The primary lecturer as well as the driving force behind the entire MIT Experimental Music Studio is Barry Vercoe. Besides being an internationally known composer, he actually wrote the majority of the software used in the studio. In fact his Music 360 program, which runs on IBM 360 computers, is the most widely used professional computer synthesis program at this time. In addition, guest lecturers were brought in daily to discuss various specialized topics. These included such luminaries as Max Mathews (the originator of direct computer synthesis technique), Allan V. Oppenheim (author of several digital signal processing books), and Dr. F. Richard Moore (from Bell Labs, designer of the Moore Digital Synthesizer). Others from the MIT staff discussed speech synthesis, psychoacoustics, reverberation, cognitive processes in composition, and microprocessors. There were 27 attendees, half of whom were composers from music departments around the world. Several were from companies as diverse as Computer Automation, National Semiconductor, Hammond Organ, The Digital Group (hobby computer manufacturer), and Gremlin Industries. Makes one wonder what might be on the market in a couple of years doesn't it? There was also a handful of independents with no stated affiliation.

Although the brochure describing the course says that no previous experience with synthesis techniques or computers is required, it is doubtful that anyone without prior exposure to analog synthesis could keep up with the pace. Also, since the studio did not run in real-time, some knowledge of the phychoacoustics of timbre would be very helpful in making good "first guesses" and doing intelligent experimentation when setting up instrument definitions. Experience in computer operation was indeed not necessary (it always helps if one can touch type) to effectively use the studio equipment but most lectures required at least a basic understanding of computer science concepts such as constants, variables, initialization, program flow, conditionals, memory allocation, and concurrent processing. On the other hand, an engineer with no musical experience might have difficulty in comprehending the need for some the features of the system. In retrospect a course attendee should have some basic knowledge of physhoacoustics, be familiar with analog synthesis principles, and have had some experience with a computer programming language such as BASIC or FORTRAN. Note that one can buy a rather complete computer that runs BASIC for less than the course tuition!

The quality of instruction was in general quite good if for no other reason than the fact that all of the lecturers knew exactly what they were talking about. Unfortunately in such situations it is easy to slip into "high level" discussion of the topic at hand without realizing it and thus temporarily "lose" the audience. A particularly bad instance of this happened when the digital simulation of reverberation was being described. Rather than speaking of a delay as a shift register, it was described in programming terms of pointers and variable updating. It took several minutes of puzzlement before the fact that these operations merely performed a signal delay was apparent. Another case occured during the description of digital filters. Although the discussion started with an unusually lucid example (the filtering effect of a 2 point moving average), it later degenerated into the standard s-plane - Z-transform - complex variable - algebraic manipulation exercise. Although this was fine for the electrical engineers (if they had seen it before), I'm sure everyone else was left at the starting gate. In any case, filter blocks built into the system made that kind of discussion unnecessary. Instead, problems with numerical accuracy and non-ideal behavior close to the nyquist frequency should have been discussed since ignorance of these can lead to some strange problems when the canned filters are used.

The course organization, although adequate, could be improved. In particular, the entire basis for direct computer synthesis, namely sampling and quantization, was never discussed! This may have been intentional since it could not be assumed that everyone knew these very fundamental concepts but there was some puzzlement in the lab when a frequency sweep form 20Hz to 20kHz (at a sample rate of 20kHz) went up and then back down. A couple of the guest lectures were out of sequence but that was probably due to scheduling problems. Also some of them were dissapointing in that they were too shallow or in one case, on a topic without apparent connection to music synthesis. Max Mathews and Dr. Moore's presentations however were superb and enlightening for everyone. A fair amount of handout material was provided which included 14 technical paper reprints, user manuals for the computer system, and numerous class notes and examples. Guest lecturers also provided blurb sheets and catalogs if they were representing a company in the synthesis field.

One fact that must be understood is that the course is not intended as a complete coverage of music synthesis in general or even the computer music segment in particular. Instead it is oriented toward understanding and effective use of the MUSIC-11 software package that professor Vercoe has put together. This is not to say that the coverage was narrow however. MUSIC-11 is different from other software synthesis systems in that a number of synthesis techniques, each of which may be functionally complete in itself, are provided. Thus classic waveform table scanning, subtractive synthesis using digital filtering, additive synthesis using a separate envelope for each harmonic, and non-linear synthesis using frequency and amplitude modulation are all available and may be used simultaneously in any mix desired. Also available are delay functions, a reverberation function, and a linear prediction package. The latter, without going into detail, extracts the time-varying spectrum from an audio signal and then simulates a filter having a frequency response with the same time varying shape. One is then free to pass any signal desired through this filter. One may also change the time scale of the filter variation and move the entire response shape up or down at will. If the filter data is derived from speech analysis, one can effectively decouple the parameters of speed, timbre, and intelligence and vary them at will during synthesis.

The most impressive part of the two week experience was the MIT Experimental Music Studio itself. The system is based on a Digital Equipment PDP-11/50 (essentially the same as an 11/45) with a couple of small disks (RK-05) for the system software and a very large disk (100MB) for sound sample storage. A quad 16 bit digital-to-analog converter connected to Dynaco amplifiers and four full-sized Klipschorn speakers as well as two professional tape decks and a mixing console comprise the audio portion of the system. No fewer than 6 standard display terminals (DEC VT-52) were available for general usage by up to 6 people simultaneously. In addition, a high resolution graphic display (IMLAC PDS-4), music keyboard, digitizer tablet, and graphic printer were available for specialized uses although the vast majority of time was spent on the VT-52's. Strangely absent was any kind of analog-to-digital conversion facility. When required, such conversion is done in the speech processing lab in another building and the data transferred on magnetic tape.

An unusual feature of the system is that it is time-shared. This means that up to 6 people (7 if the IMLAC is used) can be entering scores and instrument definitions, editing them, computing sound samples for disk storage, and playing back sound samples all simultaneously. There is a limitation that only one sample stream can be played at a time but since it came from the speakers for everyone to hear, that was a desirable feature. Although the MUSIC-11 software is quite efficient and the PDP-11/50 is a fast minicomputer, timesharing 7 ways slows things down considerably. For a 5 second swatch of moderate complexity (say 5 FM voices simultaneously) it would take an average of 5 to 10 minutes to compute. Although this sounds like a lot, most users still spent more time with the editor than with the synthesis program. Since the UNIX operating system was used, it was also possible to edit one score while computing sound from another at the same terminal. There are still a few "bugs" in the system so every couple of hours it would "crash" and require a 10 minute restart procedure before being operational again.

Now why would anyone be interested in a non-real-time computer music system and a course that applies directly only to the MIT studio? One has only to hear the strange and complex sounds that eminated from those speakers during lab sessions and then examine the corresponding simple instrument and score definitions that were typed in by musicians with no previous computer experience to realize the power of the MUSIC-11 system and direct computer synthesis in general. Actually since the majority of the system is implemented on standard hardware (except for the DAC's) it will run on most any PDP-11. In fact, a version is being made available that will run on an LSI-11 (the microcomputer used in the Heathkit H-11 personal computer). Although a large disk (not the standard floppy disk) and D-A converter would have to be added, a system having full access to the power of MUSIC-11 could be put together for less than $10,000 and some engineering or less than $20,000 if all interfaces and DAC's are purchased ready-to-go. Although the LSI-11 is considerably slower than the PDP-11/50 at MIT, there is no time sharing overhead so the effective speed for a single user may be nearly as fast.

Of course the performance/cost ratio for all computers is constantly increasing at a dizzying rate. The soon to be available Z8000 and MC68000 microprocessors for example essentially equal the performance of the PDP-11/45 (except for floating point computations which, alas, MUSIC-11 uses heavily) and will eventually show up in personal systems. At the recent National Computer Conference in New York several firms were showing prototypes of 20MB disk drives that will cost right around $1000 when in production. Digital audio systems now being retrofitted to video disk players will bring the cost of super fidelity DAC's down to the $100 range within a year or two. In order to be prepared for the very practical computer music systems of the future it is advisable to become familiar with them now.

* * * * * * * * * * * * * * * * *

News and Notes (continued from pg. 1)

results certainly predate the ones we published. I regret that this paper was missed, and will have more to say about it once I have had a chance to review it. While the results and ideas published in this newsletter were entirely independent of the JASA paper by Gerson and Goldstein, we must apologize for missing this earlier paper.

Copeland Engineering, 3211 Torrance Ave., Sacramento, CA 95822 is offering a digital scanner for a 61 note keyboard. The design includes an 8 bit D/A for 1 volt/octave output, and is available at $100, with a diode matrix board at $25. This makes possible a complete keyboard system for about $200. Write them for more details.

Jerry Lindahl of the Public Access Synthesizer Studio (PASS) writes to tell us that the studio was robbed on the weekend of July 14, with tape recorders, speakers, turntable, microphones, and typewriter missing. Due to a low funding level, PASS is interested in receiving any aid possible in replacing equipment. All contributions are tax-deductable. The address is PASS, 16 w. 22 St. (902), New York, NY 10010.

In answer to a number of requests for more information on the TI-59 program used to solve for the complex roots of high-order polynomials in the EN#95 report on pole migration, a new supplement, S-018, has been prepared. This is "TI-59 Program For Roots of a Polynomial" by Walter Luke and Bernie Hutchins, and is available as listed on our order forms and mailing sheets. It consists of 5 pages, two of description, two of the program listing, and one of an example. The program will find the roots (real or complex) of a polynomial up to 20th order with real coefficients. While this is certain to be of most interest to TI-59 users (the printer is not absolutely necessary, but a big help), there may be enough information to get a similar program going on other calculators or on a small computer.

* * * * * * * * * * * * * * * * *

ELECTRONOTES, Vol. 11, No. 103 [July 1979](published Sept. 1979)
Published by B. A. Hutchins, 1 Pheasant Lane, Ithaca, NY 14850
Phone (607)-273-8030
Please use above address and phone for subscription information, technical questions, and for all other business. Best phone hours - Wednesday evenings, 8 PM through 2 AM, Eastern time. Bad times to call - any morning, all day Monday, Tuesday and Thursday afternoons. Feel free to call and evening other than Monday until 11:30 Eastern time.