Hear musical notes on your shoulder? Could be a computer!

BY JOEL SHURKIN

The clear sound of the flute, strong and breathy. The sound changes, it is a soprano singing one clear, lovely note. Now there are many women singing in harmony. Now they are gone.

A bell tinkles to your right. It moves into a position behind you. A larger bell chimes in the corner, then both bells seem to fly across the room, meeting and passing over your shoulder, exchanging places. A drum-like sound from everywhere announces the transposition.

Blue grass guitars, a solid foot-stomping rhythm. As soon as you catch the rhythm, it shatters. The plucked strings rise in crescendo, going higher than any guitar can go, faster than any fingers can pluck. The sound becomes organ-like. Now it sounds unlike anything you have heard before.

Computer music.

All of the impossible sounds were generated by a computer. The composer-player has done the work, not with the pen on lined composition paper, but at a keyboard hooked to a video display terminal and a digital computer. The computer not only produces the sounds but contains the software that enables the composer to produce the musical form desired.

Stanford is one of the world's leading centers for research in the use of computers to make music, and to analyze sound.

Computer music may be the music of the future. That computers should be used for musical composition is only logical; music is almost always mathematical (that's why Einstein so loved Mozart), and what instrument is better able to process the mathematics better than a computer?

The scope of the music produced by computers ranges from "Silicon Valley Breakdown," written by graduate student David Jaffee, a gentle satire on bluegrass, to Michael McNabb's "Dream Song" which makes use of some recorded natural sounds, such as people talking, or Dylan Thomas reading poetry, along with the artificial sounds made by the computer.

Another variant is "Turenus," an early work by Stanford Prof. John Chowning, full of bells, chirps, and flying whistles, what might be expected from a former percussionist.

To a large extent the history of Stanford's computer music program began with John Chowning.

Chowning came to Stanford in 1962 with a degree in composition, and after three years in Paris with Nadia Boulanger, the grande dame of modern composition teachers.

Contemporary music at the time was heavily into electronics, but made use of analog devices, electronic gadgets which produced sound by measuring the relationship between objects and transmitted electromagnetic waves. Chowning was interested in electronic music but he found that Stanford had no studios or synthesizers.

He read an article by Max Mathews of Bell Laboratories who had written a computer program to generate accoustical signals. Mathews' work involved digital computers, computers which count discrete objects, usually binary numbers.

"I thought, well, that sounds interesting, and looked around campus and found that the Artificial Intelligence project had acquired the first DEC (Digital Equipment Corp.) machine, a PDP-1," Chowning said. "They had this connected to an IBM 7090, (the main campus computer) through a shared disk drive."

Since Mathews' program was written for a 7090, Chowning acquired a copy and got it to work through the PDP-1.

"It was all very primitive; not much was known about acoustics or psycho-acoustics," he says.

(Psycho-acoustics is the study of what happens to sound once it is perceived by the ears, how it is transmitted to the brain, and what the brain does with it.)

All of this is important because it has to do with the efficient way one uses this machine, the richness of sound and all that. It became overwhelmingly important because the sounds (from the computer) were so dull, and one didn't know how to improve them.

"One of the things I got interested in right away was localization of sounds. Here you have these two speakers. The sound was dry and seemed to be always located at these two loudspeakers. I thought, well, our natural listening environment is much richer than that, we might be able to use computers to simulate this."

Modern recordings of concerts or in studios were in stereo and could simulate depth, but that was "free" stereo—the engineers did not have to produce it. All they had to do was accurately record the sound as it existed in the concert hall.

"There wasn't much known about how distances were perceived."

Chowning wanted to be able to place sounds, move them around, have the listener locate them in space. He experimented with writing his own programs in FORTRAN with the help of an undergraduate, David Poole. The programs enabled him to move the sounds about using either two or four speakers.

"It was reasonably successful; the illusions were compelling enough. . . . I was able to do this myself without having any technical background, and in a relatively short time."

But, the process yielded more than he asked for. It was clear to him that computer languages held the secret; there was no necessity to alter the hardware, to solder, connect, move, rewire machinery.

Programming languages allow conceptual contact with a whole world that I thought was denied me; namely this world of system design and processing. . . . All I had to do was learn to program. If a composer can write counterpoint. . . that's a lot harder than learning programming."

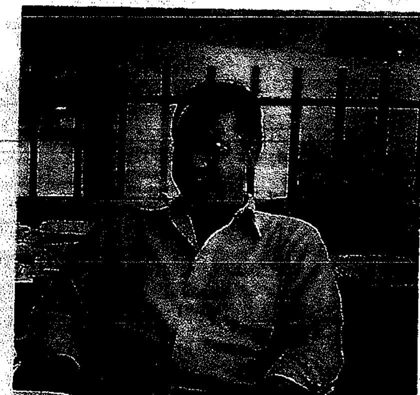
"All this was eye-opening," he says.

Chowning's enthusiasm lapped over to his adviser, Leland Smith, now professor of music. Smith wrote a program to aid in composition. The two programs—Smith's composition program and Chowning's systems program—enabled the two and their students to actually produce music on the computers. "Quite a bit of music—and good!" he adds.

"I think it was notable elsewhere because of the quality of the sound, the spatial aspects, plus the

continued on page 22

IN THE COMPUTER ROOM AT CCRMA: from left, music grad students David Jaffe and John Gordon, director John Chowning, and Music Department alumnus Bill Schottstaedt.



John Chowning

The computer not only produces the sounds but contains the software that enables the composer to produce the musical form desired.

Computer music...

continued from page 7

ease with which one could compose using the programs.

(Smith has since become involved in using computers to produce musical manuscripts.)

Chowning then turned to the nature of that computer-generated sound, the synthesis and modulation. He invented a synthesis technique which the University licenses. Originally it was just a program, but now there are hardware aspects to it.

Chowning earned his doctorate. The computer music project moved into the D.C. Power Lab, which then housed the Artificial Intelligence project. The lab was built by General Telephone and Electronics, but never completed. It is drafty, sometimes gloomy, but sits on a hill behind the campus with a lovely view.

Artificial Intelligence moved out, eventually, and the computer music people, all part of the Music Department, now use the building in isolated splendor.

Their project is now called the Center for Computer Research in Music and Acoustics, CCRMA, pronounced "Karma."

Along with the Institut de Recherche et de Coordination Acoustique-Musique (IRCAM), headed by Pierre Boulez, CCRMA is one of the world's leading centers in this field. There are also computer music centers at MIT, UC-San Diego, the University of Illinois and Colgate, most of them seeded by Stanford graduates.

CCRMA stays small deliberately, Chowning says.

"There's a critical size. If one gets beyond that, one loses contact with the people and the work that is being done. One of the most wonderful things about CCRMA is that we have this kind of natural interdisciplinary environment.

Half of the people who work here produce music, most of them composers, but the other half are electrical engineers, (or are in) speech and hearing, psychology, and computer sciences—mostly electrical engineering and speech and hearing sciences."

There are three engineers doing their Ph.D. work at CCRMA.

Chowning loves the environment.

"When I compose, that's all I do. I typically get up at one o'clock in the morning. I work up here until eight or nine in the morning. Then I take care of business nine to twelve and then I go home. That's the only way I can do it."

He does not work at home; he does not want a computer in his house. He only recently allowed a stereo, and then only because his children insisted.

"The last piece I did, I finished about a year and a half ago. I did that for six months, the only break was Thanksgiving and Christmas."

His music does not fall into classical forms. On the other hand, it is not the kind of minimalistic, or indeterminate music now popular among many modern composers and almost universally ignored by audiences. "The idea behind much of that music is more interesting than the music," he says.

"The computer can be made to sound like an instrument. This proved to be more complicated than it was first thought because there was more to the sound than was apparent. A pure, computer-generated sound did not sound natural; a certain random fuzziness was required to emulate nature, a very fine "random frequency fluctuation."

One instrument can be made to blend into another, the change almost inperceptible. A flute suddenly becomes a violin, which can, in turn, become a human soprano.

"It is also possible, of course, to produce a sound that has no counterpart in nature, but even here the sound requires something more than purity.

"We know how perfectly dull a sound is if it's perfectly periodic, it never sounds real," says Chowning.

"That's quantifying something which has never been quantified before, or even noticed as an important contribution in the perception of the timbre (tone color). If one wants to produce a sound that has no likeness to nature, there must be small changes in the sound in order to lead the ear.

"We've reached a fair degree of elegance in the kinds of sounds we can construct now that aren't necessarily like natural tones, but they all have a kind of liveliness because of having tried to model nature. It's not very easy to discover this sort of attribute to sound if one is working on some abstract notion."

"This little bit of randomness is like soy sauce in a Chinese restaurant; one wants to sprinkle a little bit everywhere."

The best test of a sound remains the human ear, Chowning says, and a musician's ear is the best of all.

Most computer musicians are now using high-level computer programs in their work as aids to composition. Many of the languages were developed by the artificial intelligence people, including the Stanford Artificial Intelligence Language (SAIL), and LISP, the seminal language of AI.

The languages are being used both for the overall composition and for the interior composition. Computers are helpful because they can remember rules and order, and much of music consists of rules and order.

"A canon is a rule, and to some extent, a fugue is a rule too," Chowning says.

No one is trying to produce Beethoven's Tenth Symphony (he only wrote nine), although artificial intelligence languages might be able to simulate one.

Programming musical structure, on the other hand, is quite useful, particularly complexes of notes or chords. The computer gives composers a great deal of power in this.

It would be theoretically possible to synthesize an entire symphony orchestra, Chowning says, but he does not see any point to doing it.

"People play because they want to play. We don't see here the computer as supplanting or replacing musicians. . . it's rather additive," he says.

Some of this music is finding its way into concert halls. A Loren Rush composition, "Song and Dance" for symphony orchestra and quadrophonic tape, was played by Seiji Ozawa when he was conductor of the San Francisco Orhcestra. A few recordings have been made.

Mostly, however, this music can be heard at computer music concerts, including the ones at Stanford.

Last summer some 2,000 people came to a set of two concerts held at Frost Amphitheater on campus.

CCRMA has received a large grant from the System Development Foundation of Palo Alto to continue its work. The money will go to buy new equipment, including a special computer built by David Poole's company, and the SUN (Stanford University Network) work station. CCRMA has also received support from Systems Concepts of San Francisco, which built CCRMA's digital synthesizer.