Every year a new crop of electronic wonders shows up on the local store shelves, where they are bought by a public that is hungry for new features. If new features and technology are not added to last year's products, fewer people will buy them. In truth, however, most of the so-called "new technology" each year isn't really new at all.

The same can be said of my project designs. Complex projects are really just combinations of simple projects. All the elements of the audio router I'm putting together are things that I've done before. Those building blocks are proven circuits that have worked well for me many times over the years, and using them will help ensure that this new project will work as I expect it to.

The router

With that out of the way, I can get back to the audio router. The circuitry I've laid out so far "saves" pressing an input switch on the keyboard in a latch. Now I need a circuit that will let me assign that switch selection to one of four outputs. Because I want to have four outputs, I need four latches. As with the inputs, I've chosen to use CD4508B CMOS dual 4-bit latches.

The layout for the output latches is shown on the right side of Fig. 1. The inputs to the latches form an eight-bit bus (one bit for each input). The latch outputs will be the control signals for the analog switches that will route the audio signals. As with the input latch, I don't need to control the enabling of the outputs because there's no possibility of bus contention at the output. Therefore I've grounded all of the 4508's enable (EN) inputs (pins 3 and 15 of each chip).

The clear (CL) inputs (pins **1** and 13) are also tied together so I have some way to clear all the output latches at once. In a similar manner, the store (ST) inputs (pins 2 and 14) of each chip are tied together and the resulting four lines are brought out to the four output selection switches previously mentioned.

Once you get this output section wired up, you can connect it to the input section you've already built. Five new switches have been added: four output select switches and a clear switch that clears both the output latches and the input latch as well.

Another function that was added to the circuit is a way to guarantee the state of all the latches when power is first applied. The 4508 is cleared when a high is put on the clear inputs, so a positive-going pulse must be generated at power up. Since there was one NAND gate left over in the 4011, I used it to build a half monostable multivibrator consisting of IC1-d, R5, and C2. When power is applied to the circuit, the brief positive-going pulse that appears at the output of IC1-d travels through R4 and clears all five of the 4508 latches in the circuit. Once that's done, the output of IC1-d stays logic low and prevents the clear inputs from floating.

You could use that same pulse to clear IC2, the heart of the keyboard section. When power is first applied to a 4017, there's no guarantee as to the state of its outputs. If you disable the keyboard clock, turn on power, and check the output pins of IC2, you'll see that the 4017 might have more than one output logic high-an illegal and somewhat confusing state. But that's corrected as soon as the chip gets a reset pulse. The reason I don't specifically reset
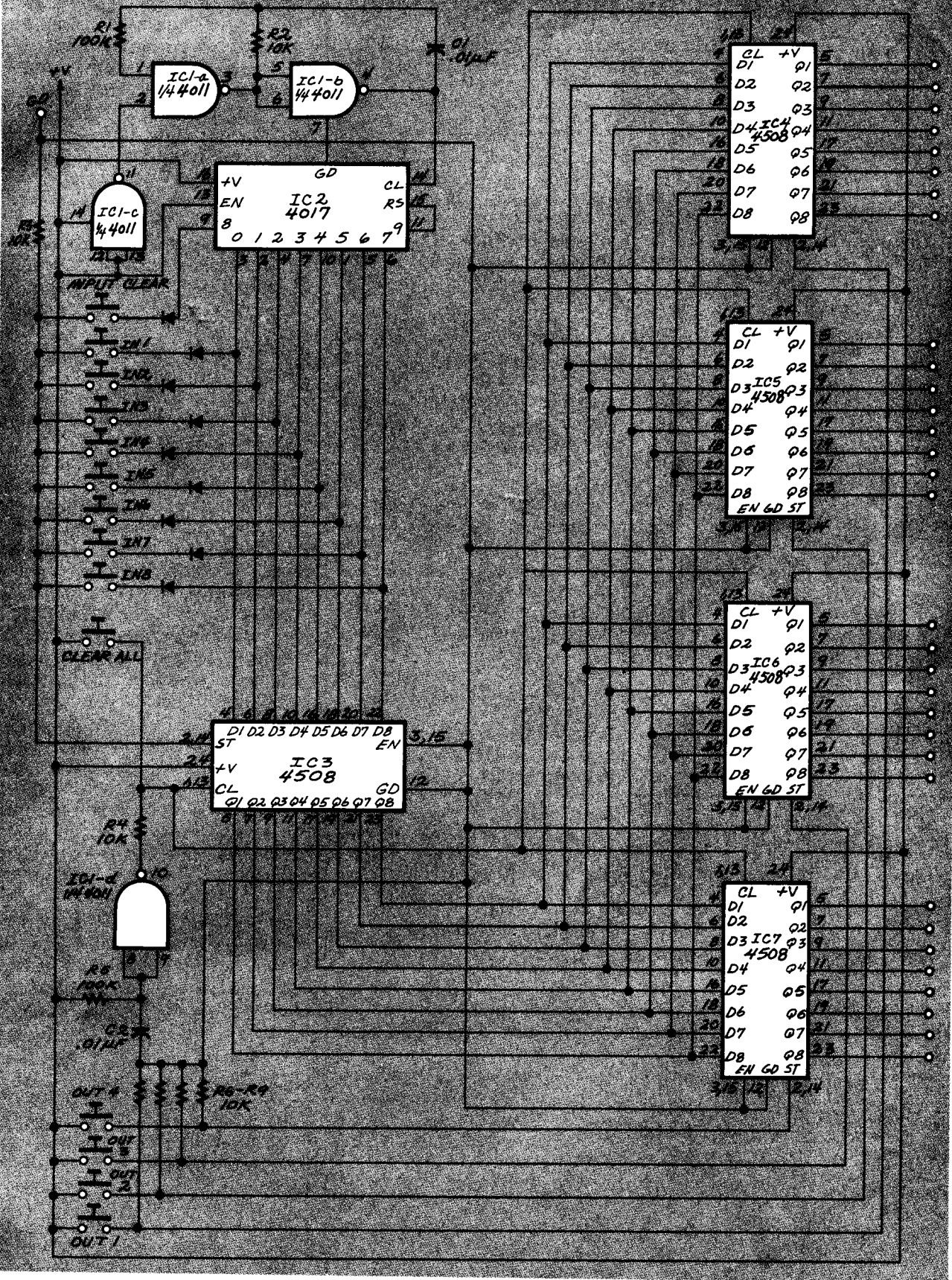
FIG. 2—THE COMPLETE CIRCUIT so far. Four output select switches and a clear switch that clears both the output latches and the input latch have been added.

the 4017 at power up is that it will be done automatically after a maximum of nine clock pulses. Since that takes only a couple of milliseconds, I'd rather just wait for it to reset than have to add an additional OR gate to do the job.

The four output selection switches connect one of the output latch's store inputs to + V, which is a cheap and easy way to generate the positive pulse needed to store the latch contents. I did not debounce the switches because it doesn't matter if the latches receive a short train of pulses. The state of the inputs doesn't change, so the number of pulses sent to the output latch is irrelevant. Resistors R6 to R9 are connected between the store inputs and ground to prevent those inputs from floating.

After it's wired up, test the operation of the circuit by selecting an input and an output and then, using a logic probe, check the output latches. Make sure that the expected output is logic high and that the other outputs remain low. Check both clear switches as well. If you get consistent errors, you've probably miswired the output bus. If you get random errors, you probably have left one of the control inputs floating.

The circuit now generates the signals needed to control the ICs that will be doing the actual audio switching. When we get together next time, I'll add this final section to the circuit and the project will be complete. I'll talk about a few extra features that can be added to the switcher to make it more useful, and I'll supply a PC board layout.