RANDOM RHYTHM MACHINE

By C. R. Fischer

Rhythm is a mainstay in the language of music. It adds essence and drive to the melody and harmony, and brings momentum to the groove. It's no surprise that drums were the first instrument invented by mankind.

For those looking to explore the twists and turns of rhythm, adding a randomizing process can create new patterns that a musician would not think of on his or her own. But playing with random music is a narrow tightrope: not enough chaos and your rhythms will be simple and cloying; too random, and the music will become anarchy.

What is needed is a process that allows control between the static and the random. This allows the user to determine the final results. And by using simple digital logic, we can put together a complex pseudo-random rhythm generator at a very reasonable cost. It includes 3 simple drum sound generators, so you can hear the results with any hi-fi or audio amplifier. If you already have some synthesizers sitting around, these can be triggered directly from the "Random Rhythm" for a variety of catchy new effects.

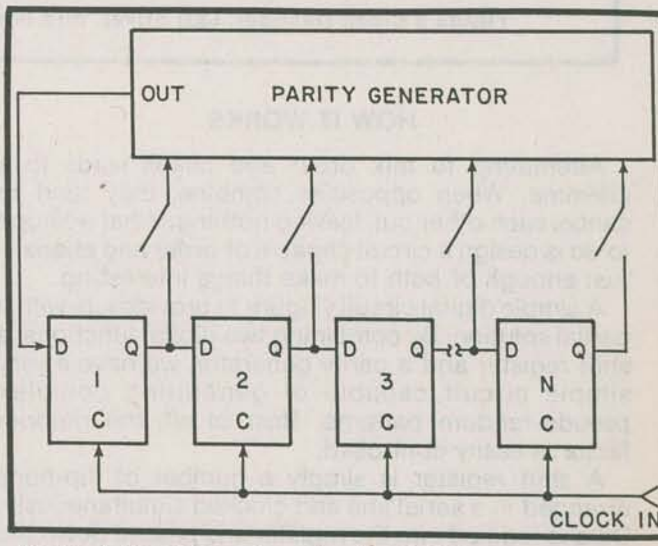
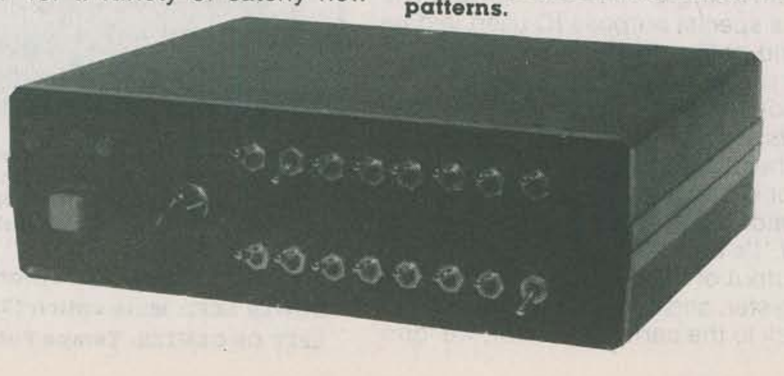


Figure 1 Combining two simple digital circuits, a shift register and a parity generator, gives us a circuit capable of generating a wide variety of complex patterns.



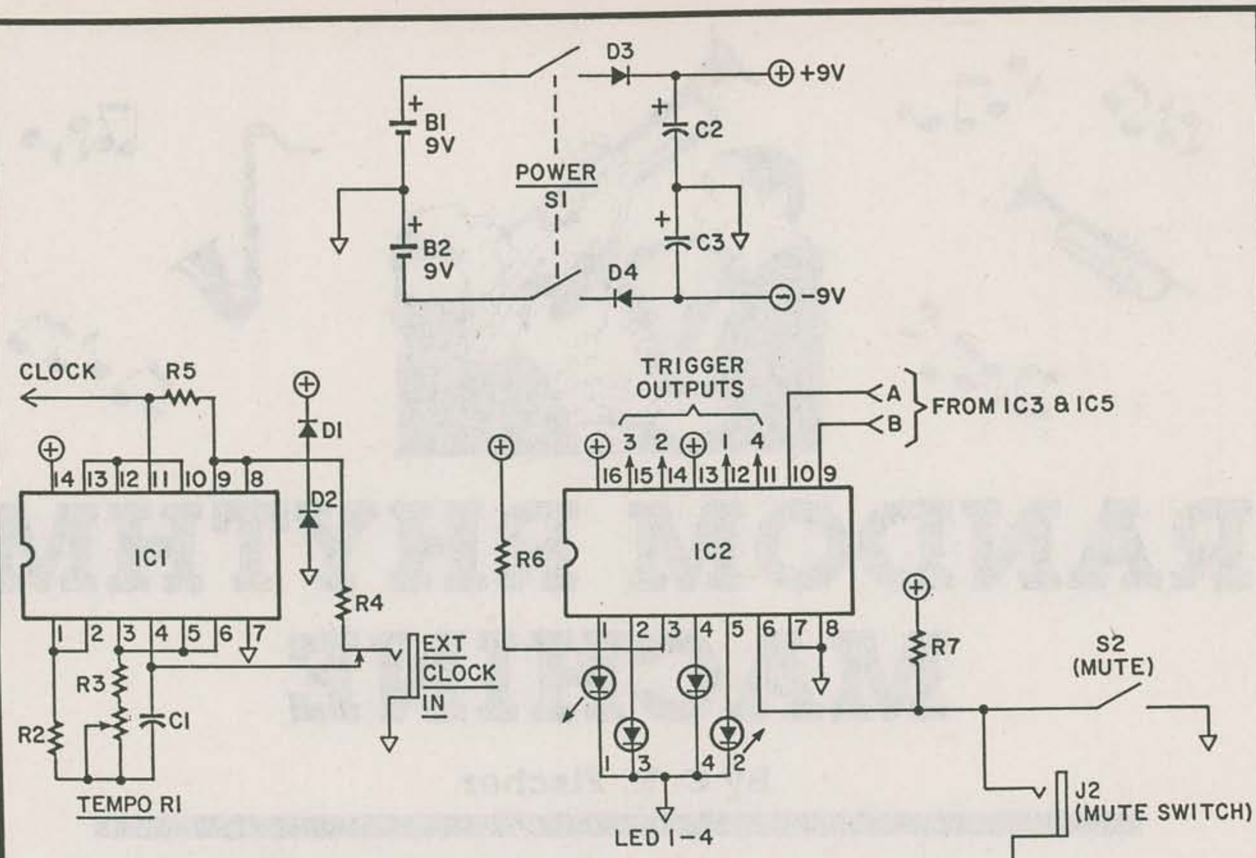


Figure 2 Clock Decoder, LED driver, and Power Supply schematics.

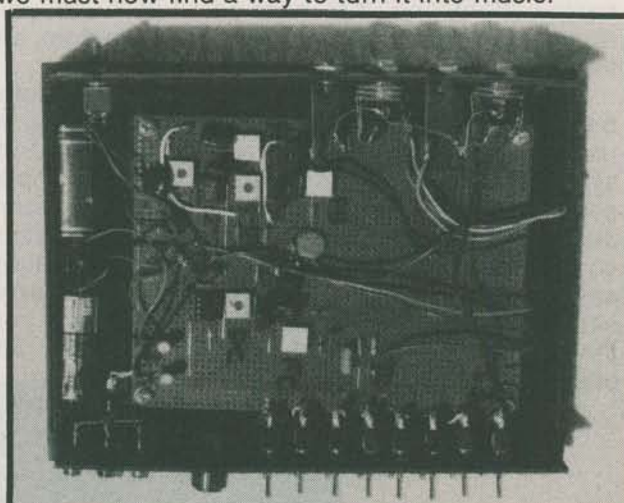
HOW IT WORKS

Attempting to mix order and chaos leads to a dilemma. When opposities combine, they tend to cancel each other out, leaving nothing. What we hope to do is design a circuit capable of **order and chaos**—just enough of both to make things interesting.

A simple digital circuit (Figure 1) provides us with a partial solution. By combining two digital functions, a shift register and a parity generator, we have a very simple circuit capable of generating complex pseudo-random patterns. Best of all, the random factor is easily controlled.

A shift register is simply a number of flip-flops arranged in a serial line and clocked simultaneously. Data is toggled into the register and passed down the line from one flip-flop to the next. A register, 4 flip-flops long, will provide a pattern of '0001', '0010', '0100', and '1000' from a single '1' in a chain of '0's. The parity generator is a special purpose IC used in data communications and pattern generation. It consists of a certain number of inputs and a single output. The parity generator looks at the combination of highs and lows at its inputs, and the output goes high or low depending on whether the number of highs are odd or even. The number or order of inputs are unimportant; all the parity generator cares about is whether or not the total number of '1's are odd or even.

By feeding the output of the parity generator to the input of the shift register, and connecting one or more register outputs back to the parity generator, we form a loop capable of generating complex feedback patterns. A single connection will provide a simple, repetitive sequence. Two or more paths create complex sequences, with each additional path increasing the complexity. Now that we've found a way of creating complex pseudo-random patterns, we must now find a way to turn it into music.



Here is an excellent view of the internal layout of the "Random Rhythm Machine". Note that there is ample room for all parts carefully arranged on the perfboard.

TOP LEFT: Trigger LEDs (barely visible)

LOWER LEFT: Mute switch (S2)

LEFT OF CENTER: Tempo Pot (R1)

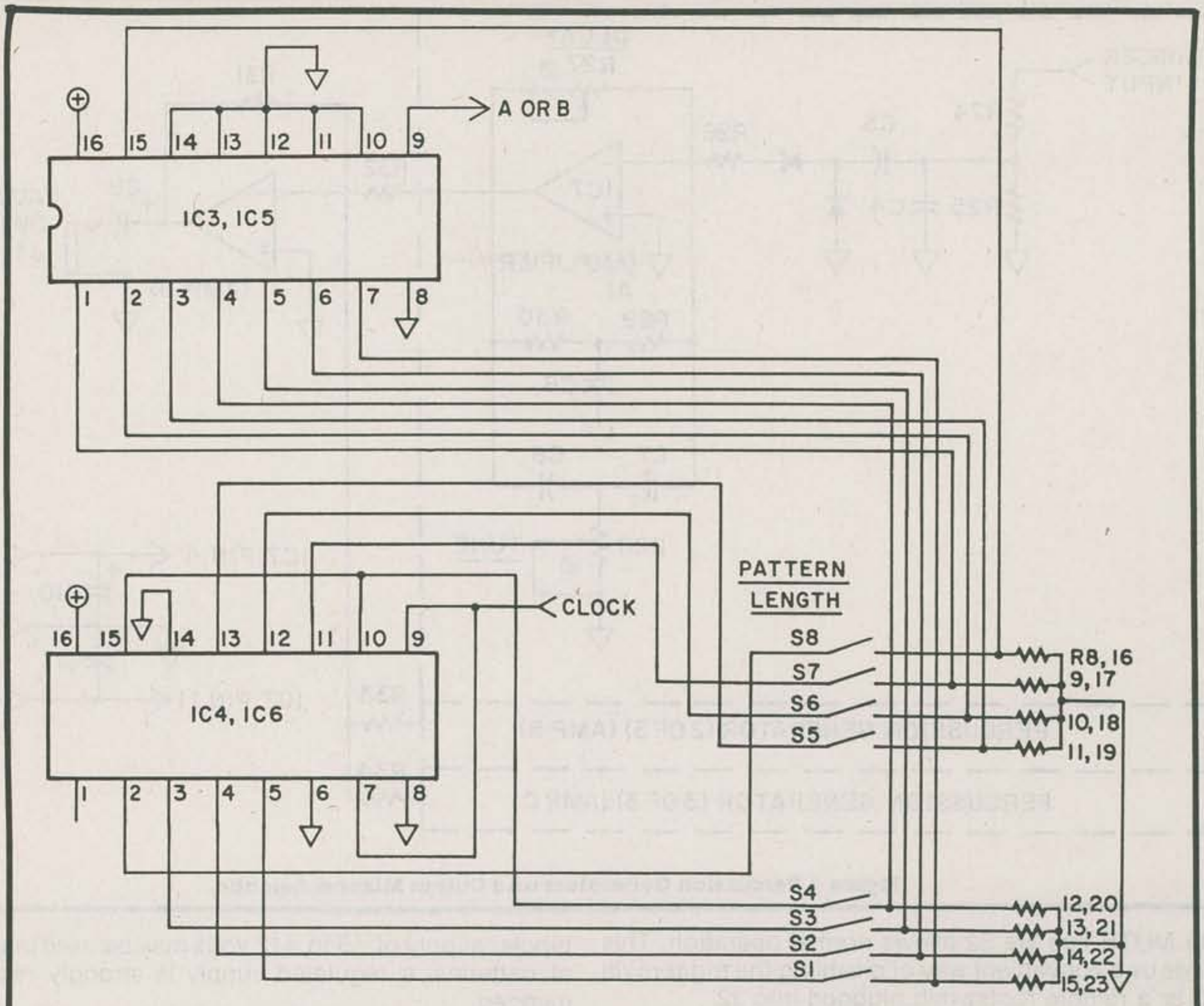


Figure 3 Pattern Generators (Build Two).

In the Random Rhythm Generator, we combine two independent register/parity circuits with a common clock to provide 4 trigger outputs. In the schematic (Figure 2), IC1 generates our master clock with R1 giving us control of the tempo. J1 allows us to use an external clock for synchronization purposes; it should be a 6 to 10V square wave. R4, D1, and D2 protect the circuit from excessive input levels; the top half of the IC adds hysterisis to clean up "sloppy" or "noisy" clock sources.

The pattern generators are illustrated in Figure 2. Note that there are two, each with independent outputs and switches. ICs 4 and 6 are the shift registers; ICs 3 and 5 are the parity generators. R8-23 are 'pull-down' resistors that force the parity generator inputs to ground when the switches are opened, and may be any value between 100k and 1 Megohm without problems. The switches allow various combinations of shift register outputs to be fed into the parity generator, which gives us a tremendous number of possible patterns.

IC2 (Figure 2) is a dual 1-of-4 analog switch that decodes the two pattern generators into usable triggers. The binary pattern at pins 9 and 10 of the chip determine which switches are closed; this gives us 4 possible output states. Since the IC is a dual device, we use the top half to generate triggers and the lower half to turn on several LEDS to show us what the Random Rhythm is doing visually. In addition, pin 6 of the CD 4052 gives us a master control of the circuit. When the pin is high, the chip is off; grounding

Front view of the "Random Rhythm Machine" demonstrates the neatness of the author's finished project.
The two rows of toggle switches are S3-20: (Top row for one parity circuit and Bottom row is for the other).

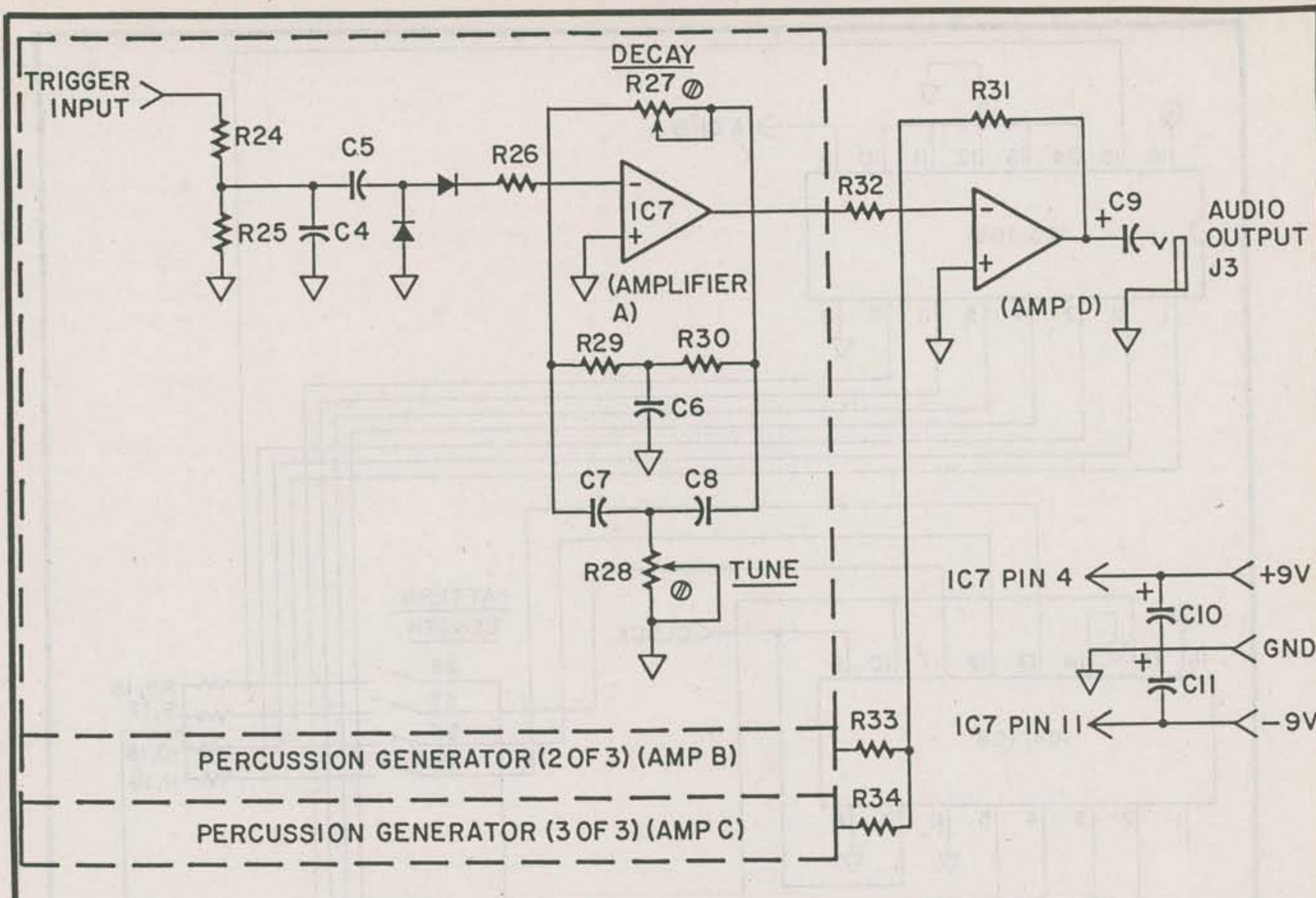


Figure 4 Percussion Generators and Output Mixer schematic.

the MUTE line via S2 allows normal operation. This gives us a convenient way of disabling the triggers via S2 or a remote footswitch plugged into J2.

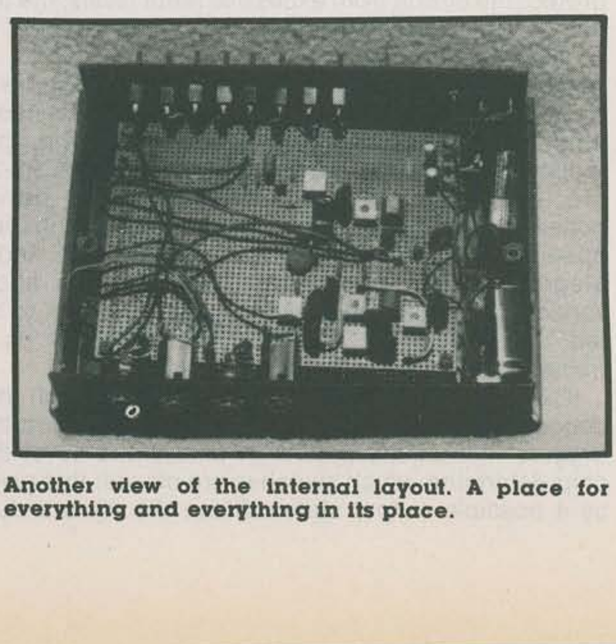
So we now have 4 triggers for use. If you've got some sound generators or synthesizers sitting around, you can look them up directly; for the rest of us, we need something to make musical sense of the thing.

Figure 4 consists of the sound generating portion of our circuit. A quad op amp (IC7) is used to generate 3 percussion sounds and as an output mixer. Each percussion generator takes a trigger and produces a sound somewhat similar to a bongo or conga. Each generator is a high-gain bandpass filter that is adjusted so that it is on the brink of oscillating. A trigger pulse causes the resonant filter to ring for a short period, giving us a percussive sound quality. R28 adjusts the drum's tuning, while R27 controls the decay, which gives us a variety of percussive sounds. Note that there are 4 triggers and only 3 sound generators. By leaving one trigger unused, we generate a 'rest' whenever the unused trigger turns on. This brief silence serves to break up the constant sound, and adds a bit of spice. The last portion of IC7 acts as a summing amplifier, and it mixes our signal for use with any audio amplifier.

By using CMOS ICs in our design, we have the advantage of being able to power the circuit from two 9-Volt batteries. S1 is the power switch; D3 and D4 prevent reverse polarity from wreaking havoc on our circuit. C2 and C3 are supply bypass capacitors. A bipolar supply of ±5 to ±12 volts may be used instead of batteries; a regulated supply is strongly recommended.

BUILDING IT

The Random Rhythm Generator uses a non-critical layout, and it may be built using any of the usual wirewrap or perfboard techniques. The author's prototype used 2 4½"×6" perfboards; the digital electronics (Figures 2 and 3) were on the bottom, with the analog electronics and trimpots mounted on the top for easy access.



Another view of the internal layout. A place for everything and everything in its place.

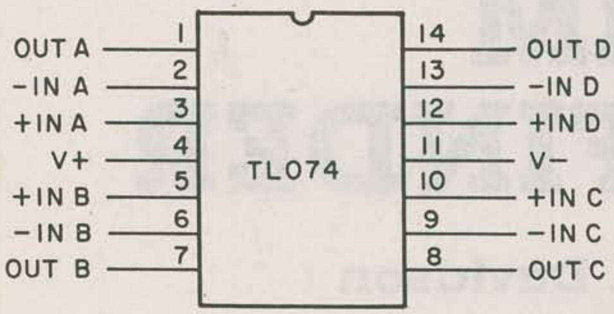


Figure 5 Pinout of IC7.

The repetitive circuitry between the 16 panel switches and the boards allow a lot of room for error, so some caution is called for. Ribbon cable would be a good idea here. If you're really on a budget, you might consider using DIP switches to economize (2 DIP switches run at around $ 3.50; 16 toggle switches might run at a buck or two each). Virtually any metal or plastic case may be used for the circuit.

IC sockets are strongly advised for all chips. All ICs except IC7 use CMOS technology, and are susceptible to static discharge. Avoid handling the ICs until all wiring is complete, and follow the usual guidelines. After the chips are safely installed, set all tune trimpots to midway, and all decay trimmers to minimum resistance. Connect J3 to an audio amplifier, and double check that the amplifiers volume is *completely down.* If one of the percussion generators is miswired or misbehaving, it could oscillate at full volume—which could damage your speakers, your hearing, or your sanity!

TEST AND CALIBRATION

Connect a pair of batteries to the circuit, and apply power via S1. One or more of the LEDS should turn on. If not, try S2 and see if they come on. If this has no effect, turn off the power and examine the circuit for wiring problems. Once the LEDs are working, adjust R1 and verify that the speed of the LED flashing varies.

Now it's time to raise the volume on your amplifier and listen for sounds. If you hear one or more constant tones, the sound generators are oscillating. Change the settings on the decay trimmers until this stops. Close a switch on both pattern generators and slowly raise the decay pot on a generator. The sound quality should go from silence, to muted, till it finally breaks into oscillation. Set the pot for a pleasing sound, then adjust the tuning trimmer and observe the effect. Note that there is some interaction between the two trimmers and you may have to spend a little time obtaining a pleasing sound quality. Then go on to the other two generators and duplicate the procedure. Your Random Rhythm is now complete and ready for use.

Now experiment with the switch settings. Try closing the number 4 switch for one pattern generator, and the number 8 for the other. You should hear a simple, repeating sequence (this is because 4 and 8 are multiples of 4/4 time). Now close any other switch; the pattern should suddenly become irregular. A few minutes with the Random Rhythm will provide plenty of unusual patterns.

While the Random Rhythm is fun by itself, it can be paired with any rhythm box with a clock output for a variety of special effects. While the rhythm box provides a regular beat, the Random Rhythm adds 'fills' and variations around the basic rhythm. In addition, a footswitch may be used with J2 to turn the rhythm on and off as desired.

Order and chaos make for an unusual pair, but the results can be useful, as well as unpredictable. Build the Random Rhythm and add a little fun (and chaos) to your music. ■

PARTS LIST FOR RANDOM RHYTHM GENERATOR

SEMICONDUCTORS
IC 1—CD 4001B (Quad CMOS NOR gate)
IC 2—CD 4052B (Dual 1-of-4 multiplexer)
IC 3,5—CD 4531B (12 input parity generator)
IC 4,6—CD 4015B (dual 4-stage shift register)
IC 7—TL 074 (Quad low-noise op amp)
LED 1—4 Inexpensive LEDs
D1, 2, 5, 6—1N914 diodes
D3,4—1N 4001 rectifiers
S1—DPDT slide/toggle switch
S2—SPST pushbutton or toggle switch
S3-20—SPST slide, toggle, or DIP switch assembly
B1,2—9 V transistor radio batteries
Capacitors—(all rated at 25V or better)
C1—.5 uF poly or film cap
C2, C3—10 uF electrolytic or tantalum
C4, 6,7,8—(12 total) .1 uF film or poly
C9,10,11—1 uF electrolytic or tantalum
C5—.005 uF disc (3 total)
Resistors—(¼W or greater)
R1—1 Megohm pot
R2,5—1 Meg
R3,—32,33,34,(31) 10K
R4—100K
R6—1K
R7—33K
R8-23—100K to 1 Meg (exact value not important)
R24—1K (3 total)
R25—39K (3 total)
R26—330K (3 total)
R27—200K trimpot (3 total)
R28—2K trimpot (3 total)
R29,30—100K (6 total)
Misc.—Perfboard, IC sockets, (3) ¼" phone jacks, (2) 9V battery clips, solder, mounting hardware, knob for R1, etc.

REFERENCES

Barbello, James. "Build This Percussion Synthesizer", Radio-Electronics, September 1979.
Fischer, C. R. "(Build The) $20 Drum Synthesizer", Modern Electronics, January 1987.
Chamberlin, Hal. "Musical Applications Of Microprocessors", Hayden Books, Pp. 311-315.
Lancaster, Don. "CMOS Cookbook", Howard Sams & Co.