Integrated circuit random foreperiod device and digital interval timer[1]

J. S. SOBOLEWSKI[2] *and* W. H. PAYNE,[3] *WASHINGTON STATE UNVIERSITY, Pullman, Washington 99163*

A digital integrated circuit computer that produces random (as opposed to pseudorandom) foreperiods is described. Also contained in the device is a precision digital interval timer.

Generation of random foreperiods for use in behavioral experiments involving variable foreperiod reaction time, or aperiodic stimulation to prevent neural "driving" in electrophysiological work, usually is accomplished by one of the following methods: (1) A multiple pole rotary stepping switch controls the time base of a counter (i.e., by varying capacitance or resistance). (2) Tone bursts or noise spikes are recorded at aperiodic intervals on magnetic tape and used to trigger external relay contacts. (3) Photographic film, through which light is passed, advances at a steady rate. When an opaque frame is encountered, relay contacts are closed. Paper tape variations of this principle are also used. (4) Rotating mechanical drums with pegs to trip switches are common. (5) The E initiates the stimulus directly "at random." This is probably the most common method. (6) If an on-line computer is available, a fixed-point pseudorandom number in an appropriate range can be computed. A memory location is set to zero and a first relay closed. The location is incremented by units until its contents compare equally to the pseudorandom number. At this point, a second relay is closed. The interval between the closing of the two relays determined by the sum of the cycle times of the increment loop generates the pseudorandom foreperiod.

Methods 1 through 4 produce pseudorandom periodic foreperiod sequences of variable length and known spectrum (time X probability of stimulus). Subsequences of foreperiods can often be identified by Ss. Method 5, while producing aperiodic sequences, gives an unknown spectrum. Also, most Es are unable to produce uniform distributions of foreperiods over time.

If a computer is available, there are two important methods for generating maximum period pseudorandom sequences. First, a Lehmer congruential generator, such as described by Payne, Rabung, and Bogyo (1969), can be used to generate pseudorandom numbers uniform in one dimension. Second, a
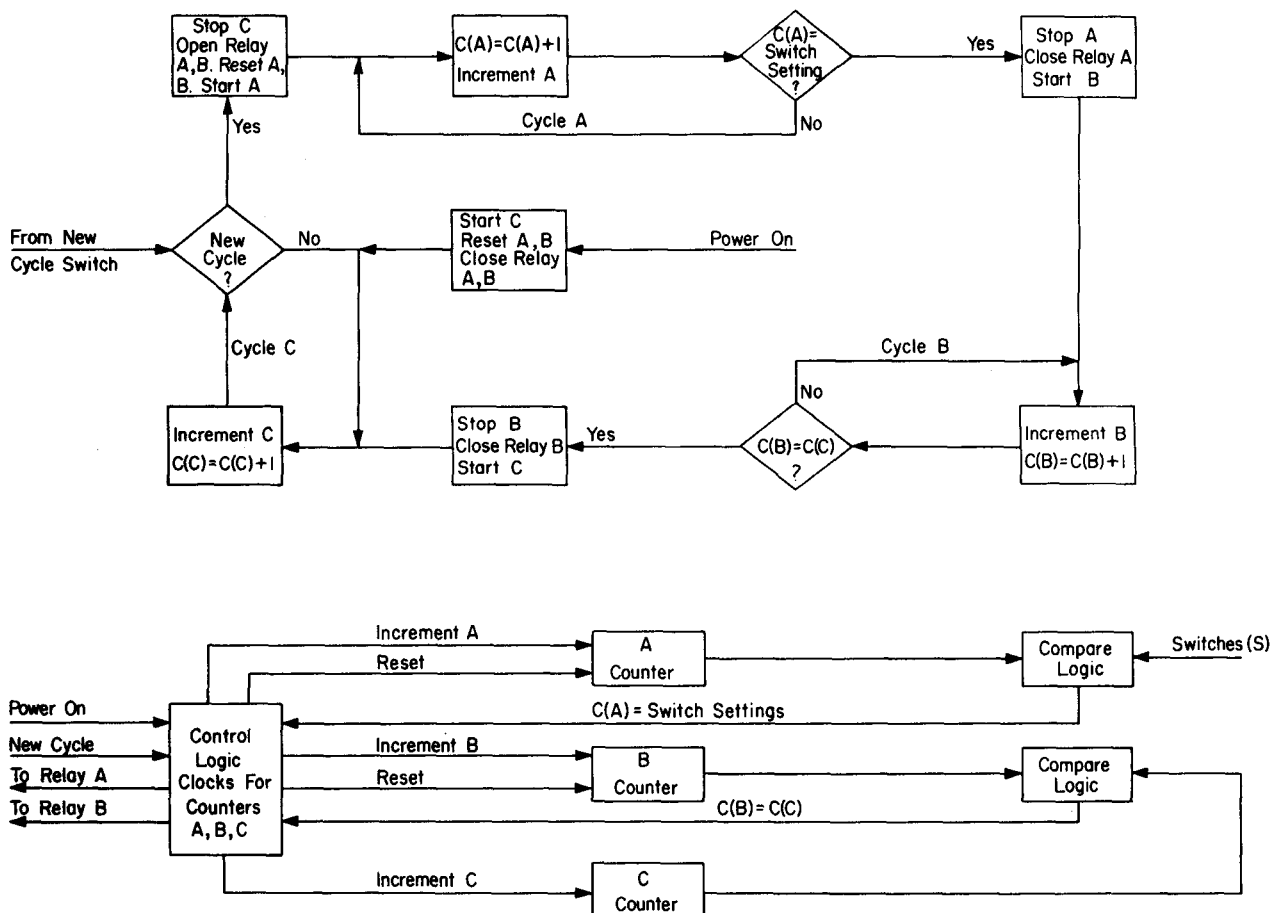

Fig. 1. (Top) Flowchart of random foreperiod generator. Counter A (12 bits, 100 cps) is the holdoff timer. Counter C (12 bits, 1.2 mcps) produces the random number. Counter B (12 bits, 10-100 cps) is incremented until its contents are equal to Counter C. (Bottom) Block diagram, showing the main components of the random foreperiod generator.

318

Behav. Res. Meth. & Instru., 1969, Vol. 1 (8)

linear recurring sequence [also called Tausworthe or pseudonoise (PN)] generator that produces maximum length pseudorandom numbers is applicable. Kendall's algorithm for this generator has been presented by Whittelsey (1968) and can be programmed in ASA standard FORTRAN for small computers. Values of M and N (see Whittlesey) for irreducible trinomials can be found in Golomb (1967). An example of coding the Tausworthe generator in ASA standard FORTRAN is given. The PN sequence method is relatively new compared to the Lehmer congruential method. Extreme caution (Fellen, 1969) should be exercised to test the randomness of PN sequences. These computer methods are excellent but require an expensive computer.

We present a relatively inexpensive digital integrated circuit random foreperiod generator with a built-in holdoff digital timer.

Principles of Operation

As shown in block diagram Fig. 1 (bottom), the device consists of three 12-bit counters, two sets of compare logic, and some control logic that also contains three clocks to run the counters. There are three inputs to the device; the POWER ON switch that is also used to initialize the system, the *New Cycle Switch* that is pressed every time a foreperiod is required, and a set of 12 switches that determine the length of the holdoff time. There are also two outputs from the control logic that drive Relays A and B. Relay A is closed at the end of the holdoff time, while Relay B is closed at the end of the foreperiod.

The principle of operation is best understood by referring to the flowchart at the top of Fig. 1. When the POWER ON switch is closed, the device is initialized and then enters the loop called Cycle C, where the C counter is incremented at a rate of about 1.2 mcps. This counter operates Modulo 4096, that is, it may contain any binary number from 0 to 4095, and when 4095 is reached, the next number in the sequence is 0. Counter C therefore recycles over 250 times a second. Relays A and B are both closed during Cycle C.

Counter C continues to run until the *New Cycle Switch* is closed, which signals a request for a random foreperiod. When this happens, Counter C is stopped (it will contain some number between 0 and 4095) and a new loop, called Cycle A, is entered. Here Counter A is incremented from 0 at a rate of 100 cps until its contents are equal to the settings on the 12 external switches. When this happens, the device enters into a third loop, called Cycle B. Thus, the duration of Cycle A, which is the holdoff time, can vary between 0 and 40.95 sec ±0.01 sec. Note that during the A cycle, Relays A and B are open.

In Cycle B, the B counter is incremented at a rate that can be varied from 10 to 100 cps by means of a potentiometer. The B counter will be incremented from 0 until its contents are equal to the C counter. Since Counter C contains a random number between 0 and 4095, the duration of Cycle B (the foreperiod) will be random and will vary between 0 and 40.95 sec in increments of 0.01 sec or between 0 and 409.5 sec in increments of 0.10 sec, depending on whether the frequency of the B clock is set at 100 or 10 cps, respectively. The frequency of the B clock can be changed easily to give different ranges. Note that during the B cycle, Relay A is closed while Relay B is open.

At the end of the B cycle, the device closes Relay B, goes to the C cycle and stays there until a new foreperiod is requested. Although not shown in the flowchart of Fig. 1, the control unit is set up so that pressing the *New Cycle Switch* at any time has the same result as pressing the switch when the device is in Cycle C. This is useful in cases where the device is in Cycles A or B and it is desired to abort the operation and start a new cycle immediately.

Construction

Much of the counter and compare circuit wiring is repetitive,

```
       FUNCTION RAND(Z)
       LOGICAL*4 A,B,ACOMP,BCOMP
       EQUIVALENCE (I,A),(J,B,REAL),(ACOMP,ICOMP),(BCOMP,JCOMP)
       DATA I/524287/
       N=262144
C      N=2**(WORD SIZE OF COMPUTER-SHIFT LENGTH(M)-1).
       M=8192
C      M=2**(SHIFT LENGTH(M)).
       J=I/M
       ICOMP=-I-1
       JCOMP=-J-1
       B=A.AND.BCOMP.OR.ACOMP.AND.B
       I=J
       J=J*N
       ICOMP=-I-1
       JCOMP=-J-1
       A=A.AND.BCOMP.OR.ACOMP.AND.B
       J=2147483647
C      J=2**(WORD SIZE OF COMPUTER-1)-1.  SET SIGN +.
       A=A.AND.B
       J=I/128
       J=J+1073741824
C      J=J+2**(WORD SIZE OF COMPUTER-2).  FLOATING-POINT EXPONENT.
       RAND=REAL
       RETURN
       END
```

Sample Program

making the device relatively simple to construct. Transistor-transistor logic (TTL) was used in preference to resistor-transistor logic (RTL) and diode-transistor logic (DTL), even though TTL is at present slightly more expensive. TTL is available in a greater variety of standard circuits, has faster switching capability, and an ever increasing demand will tend to lower prices in the future.

The construction is modular. Three separate 12-bit counters and two 12-bit digital comparators must be built. The entire random foreperiod device can be constructed on a 9 x 16 in. fiber glass matrix board. It is most important that the board be stiffened sufficiently so that there is no mechanical strain that can cause intermittent failures in the solder joints.

Although not shown in the circuit diagrams, care must be taken to ensure that the power supplies, especially the +5-V, are adequately bypassed to prevent noise on the supply lines. To bypass low-frequency noise, 10 microfarad electrolytic capacitors should be connected between the +5-V line and ground; there should be one such capacitor for every 4 in. of the +5-V line. To bypass any high-frequency noise, each electrolytic capacitor should have a 1000-picofarad Mylar capacitor in parallel with it.

Detailed Description

The most complicated part of the device is the control unit shown in Fig. 2. It consists of six SN7400N integrated circuits (quadruple two-input NAND gates) and some discrete components. Its function is to provide the clock and reset signals to drive the counters, to use the two outputs of the compare logic to set relays, and to display by means of two lamps the current state or cycle of the device. The waveforms at the various test points of the control unit are shown in Fig. 3.

The two low-frequency clocks are generated by unijunction transistor oscillators. The 100 clock (Test Point J) is fixed at 100 cps, the exact frequency being adjusted by the 25K ohm potentiometer. The frequency (and hence the maximum duration of the A cycle) can be varied over a large range by changing the 0.1-microfarad capacitor. The 100 clock, gated with the cycle A signal (Test Point B) generates the clock A signal (Test Point D). The 10-100 clock of Test Point J is variable between 10 and 100 cps, thus giving a wide range of foreperiods. With R2 set at 0 ohms, R1 is used to adjust the frequency to 10 cps. With R1 thus adjusted, R2 can now be used to vary the frequency from 10 to 100 cps. As before, these frequencies can also be varied by changing the 0.68-microfarad capacitor. When gated with the cycle B signal (Test Point F), the 10-100 clock forms the clock B signal (Test Point G). The 1-mcps (actually about 1.2 mcps) clock is obtained at Test Point M by connecting two of the NAND gates
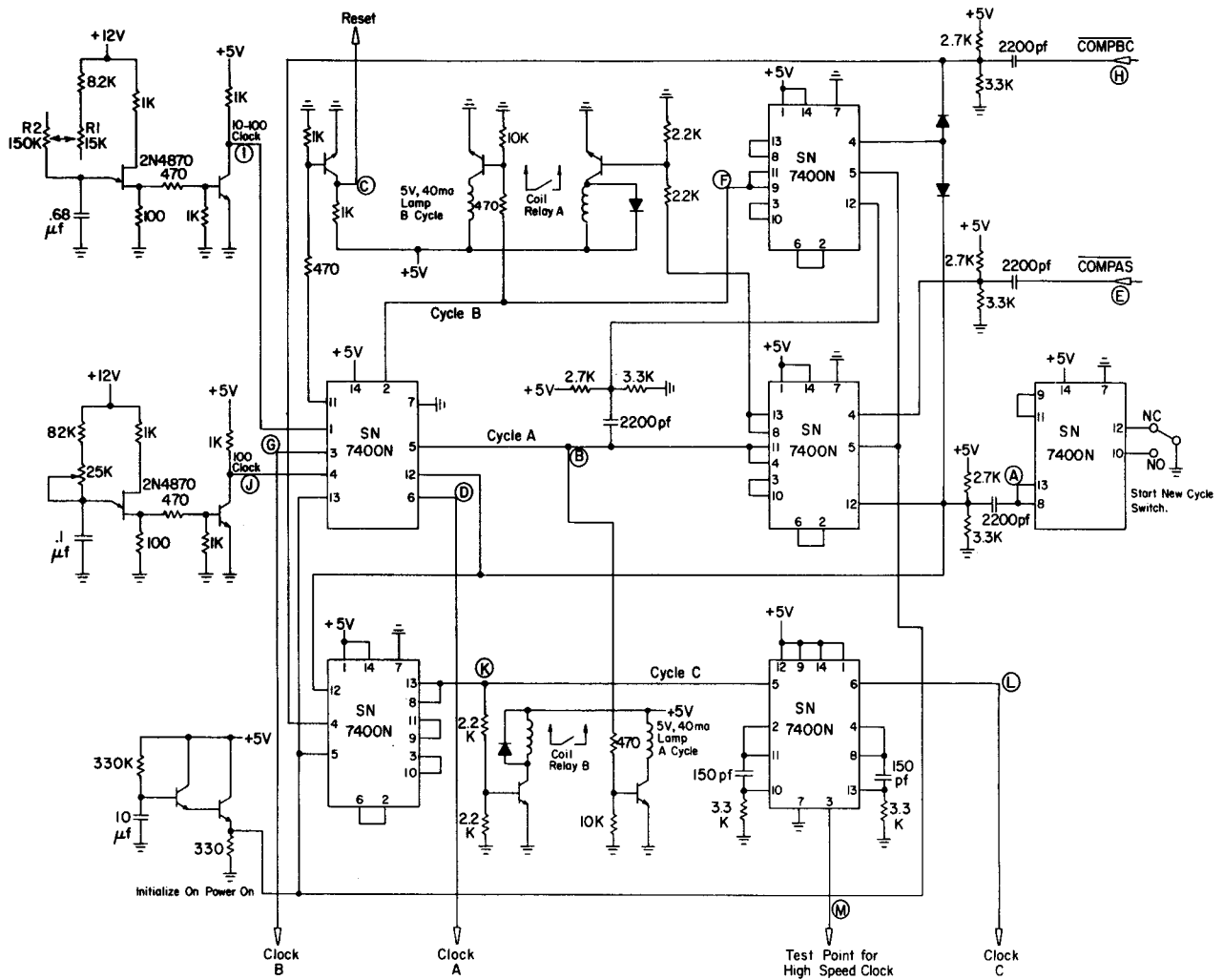
Fig. 2. Control, clocks, relay, and lamp drivers. Unless otherwise stated, all transistors are 2N5134 (or equivalent) and all diodes are 1N914 (or equivalent). The outputs to Fig. 4 are Reset, Clock A, Clock B, and Clock C. Inputs from Fig. 5 enter at COMPAS and COMPBC. The *New Cycle Switch* is a pushbutton type.

as an astable multivibrator. With the values of components shown, the frequency is about 1.2 mcps. Gating the 1-mcps clock with the cycle C signal (Test Point K) forms the clock C at Test Point L.

The device is normally in the C cycle. When the *New Cycle Switch* (a pushbutton switch that is normally in the NC position) is depressed momentarily to the NO position, it causes the following: (1) A 1-microsec wide pulse at Test Point C that resets the A and B counters to zero. (2) Cycle C lead goes low, causing Clock C to go high and thus stopping Counter C from incrementing. (3) The coils of Relays A and B are de-energized, opening the contacts. (4) Cycle A goes high, gating Clock A and causing Counter A to increment.

While Counter A is incrementing, the device is in the A cycle. The A cycle lamp is therefore on and the A counter continues incrementing until the compare logic senses that the contents of the counter are equal to the switch settings. When this happens, the lead COMPAS at Test Point E goes low, causing the following: (1) Cycle A signal goes low, turning off the cycle A lamp and closing Relay A. The clock A signal goes high and stops Counter A from incrementing. (2) Cycle B signal at Test Point F goes high, causing the cycle B lamp to turn on. At the same time, the 10-100 clock is gated through to form Clock B at Test

Point G, which causes the B counter to start immediately.

The device is now in Cycle B, with the B counter incrementing until its contents equal the contents of the C counter. When this happens, the signal COMPBC, at Test Point H, goes low, resulting in the following: (1) Cycle C, at Test Point K, goes high, closing Relay B and gating the Clock C signal, which causes the C counter to start incrementing again from its former value. (2) Cycle B signal goes low, causing Clock B to go high and thus stopping Counter B from incrementing. The cycle B lamp also turns off. Note that when both lamps are off, the device must be in Cycle C. The device will remain in Cycle C until the new cycle switch is depressed again, at which time the whole process will be repeated.

The circuit that initializes the system when power is turned on is shown at the bottom left hand corner of Fig. 2. When the +5-V supply comes on, the 10-microfarad capacitor charges through the 330K ohm resistor, maintaining the output low for several seconds before going high. This low output initializes the system as follows: (1) Cycle A signal is forced down, opening Relay A and causing Clock A to go high, hence preventing the A counter from incrementing. (2) Similarly, cycle B signal is forced down, opening Relay B and preventing Clock B from incrementing the B counter. (3) It causes the rest signal that resets the A and B

320

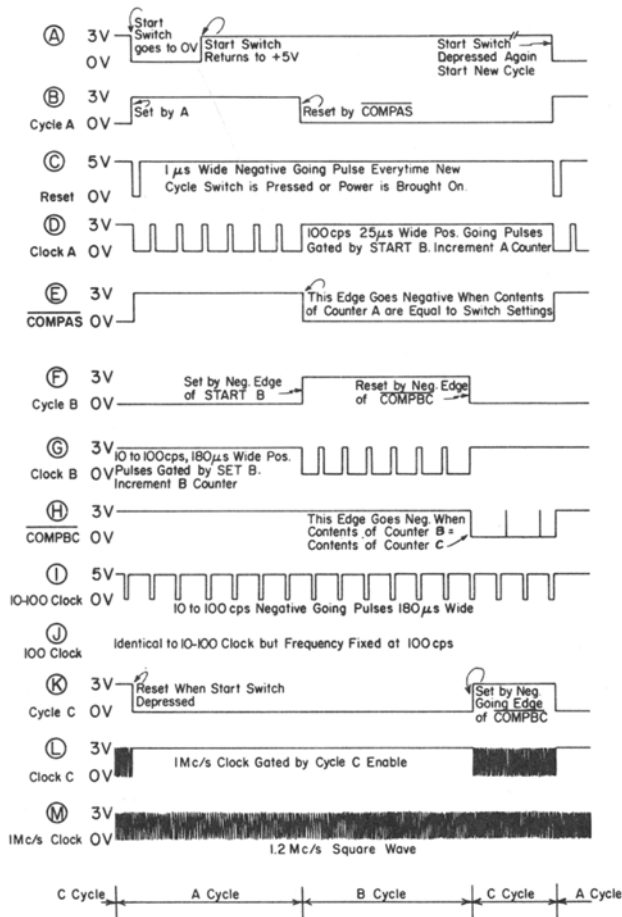Behav. Res. Meth. & Instru., 1969, Vol. 1 (8)

Fig. 3. Waveforms for test points indicated in Fig. 2. Timing cycles at the bottom of the figure should be compared with the flowchart at the top of Fig. 1.
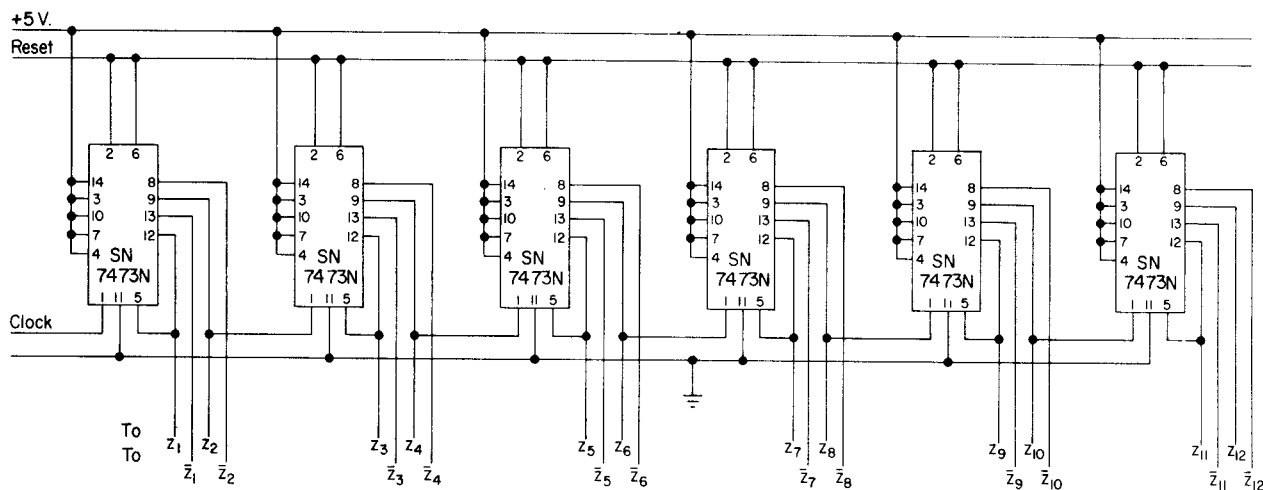
counters. (4) Cycle C signal is forced up, enabling the clock C that increments the C counter forcing the device into the C cycle. It should be noted at this point that the *New Cycle Switch* will be operative only some 5 to 10 sec after the power has been turned on. This is the time it takes the initialize signal to go high.

The three 12-bit counters are identical, each using six SN7473N dual J-K flip-flops, as shown in Fig. 4. The connections for these counters are explained in the caption. Note that the reset line for Counters A and B is connected to the reset line from the control unit, but for Counter C, the reset line is connected to +5 V.

The circuit for the compare logic is shown in Fig. 5. Each of the two identical circuits consists of six SN7451N circuits (dual two-input AND-OR INVERT gates), two SN7430N circuits (eight-input NAND gates) and a SN7400N. The inputs are again explained in the caption. The bottom right of the figure shows a switch wired as a flip-flop to give the required complementary outputs.

Power Requirements

The power required for the device is +12 V at 10 mA and +5 V at about 1 A. If such power supplies are not available, the +12-V power supply in Popular Electronics (Anon., 1969) can be used. The +5 V can be obtained from the +12 V by using an appropriately heat-sinked 6.8-V Zener diode (e.g., the 1N3305).

Tests

The device was tested by triggering the *New Cycle Switch* by a timer. The recycle interval of this timer was greater than the holdoff time plus maximum random foreperiod. Even though a new cycle was initiated by a fixed repetitive source, the random foreperiod device delivered uniform foreperiods where the foreperiod interval was subdivided into 10 time categories to test uniformity.

Conclusions

The device that has been described can generate random foreperiods preceded by a holdoff period. The B counter furnishes the random foreperiod for a variable time interval. The
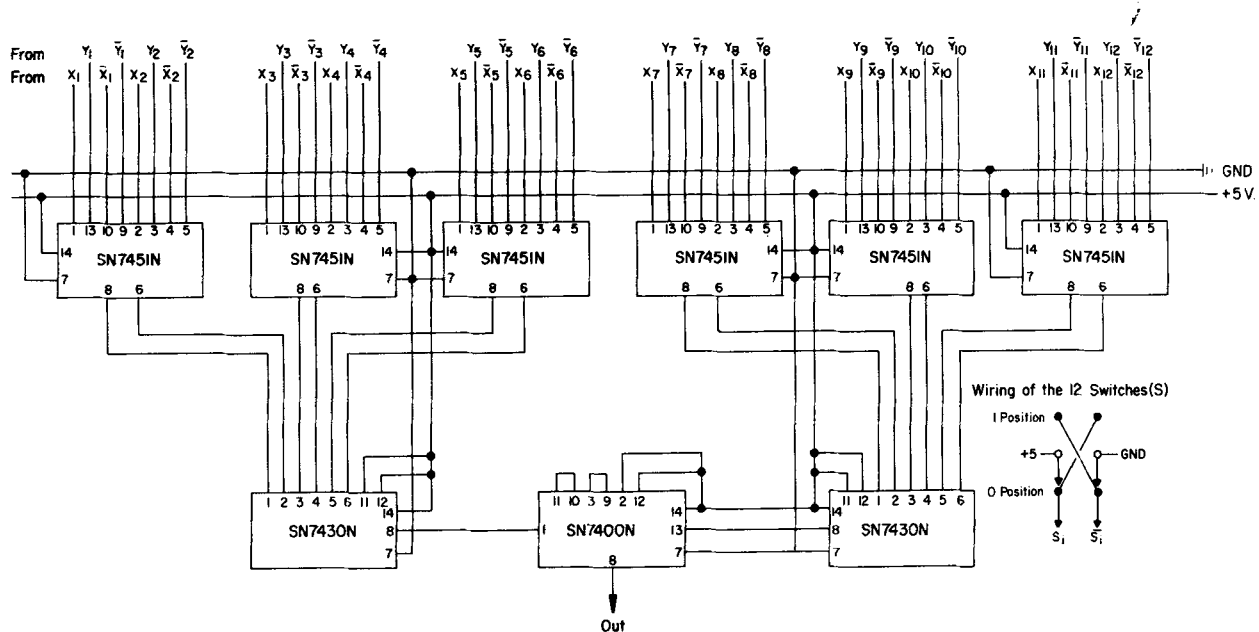

Fig. 4. Circuit for the A, B, and C counters. For Counter A: Clock=Clock A, $\overline{Z}_i = \overline{A}_i$, $Z_i = A_i$ (Z_i represents the complement of Z_i, i.e., not Z_i). For Counter B: Clock=Clock B, $Z_i = B_i$, $\overline{Z}_i = \overline{B}_i$. For Counter C: Clock=Clock C, $Z_i = C_i$, $\overline{Z}_i = \overline{C}_i$. For Counters A and B, the *Reset* line is connected to the Reset output of Fig. 2. For the C Counter the *Reset* line is connected to the +5-V supply.

Fig. 5. Circuit for the compare logic. For comparison of the A counter with the switch settings: $X_i = A_i$, $\overline{X}_i = \tilde{A}_i$, $Y_i = S_i$, $\overline{Y}_i = \overline{S}_i$, OUT=$\overline{COMPAS}$. Note how the 12 switches are wired to give complementary outputs S_i and $\overline{S}_i$. For comparison of the B and C counters: $X_i = B_i$, $\overline{X}_i = \overline{B}_i$, $Y_i = C_i$, $\overline{Y}_i = \overline{C}_i$, OUT=$\overline{COMPBC}$.

A counter provides the holdoff period whose duration may be set in increments of 0.01 sec by a set of 12 switches. It can thus be used as a digital interval timer. The complete instrument can be built for less than $200.00.

REFERENCES

ANONYMOUS. Build your own power supply. Popular Electronics, 1969, February, 72.

FELLEN, B. M. An implementation of the Tausworthe generation. Communications of the Association for Computing Machinery, 1969, 12, 413.

GOLOMB, S. W. Shift register sequences. San Francisco: Holden-Day, 1967.

PAYNE, W., RABUNG, J., & BOGYO, T. Coding the Lehmer pseudo-random number generator. Communications of the Association for Computing Machinery, 1969, 12, 85-86.

WHITTLESEY, J. R. B. A comparison of correlational behavior of random number generators for the IBM 360. Communications of the Association for Computing Machinery, 1968, 11, 641-644.