THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 52, No. 10, December, 1973
*Printed in U.S.A.*

# A Universal Digital Data Scrambler

By DAVID G. LEEPER

(Manuscript received May 22, 1973)

*Analyses in the literature of digital communications often presuppose that the digital source is "white," that is, that it produces stochastically independent equiprobable symbols. In this paper we show that it is possible to "whiten" to any degree all the first- and second-order statistics of any binary source at the cost of an arbitrarily small controllable error rate. Specifically, we prove that the self-synchronizing digital data scrambler, already shown effective at scrambling strictly periodic data sources, will scramble any binary source to an arbitrarily small first- and second-order probability density imbalance $\delta$ if (i) the source is first passed through the equivalent of a symmetric memoryless channel with an arbitrarily small but nonzero error probability $\epsilon$, and (ii) the scrambler contains $M$ stages where*

$$M \geqq 1 + \log_2[(\ln 2\delta)/\ln (1 - 2\epsilon)].$$

*Some interpretations and applications of this result are included.*

## I. INTRODUCTION AND SUMMARY

Digital transmission systems often have impairments which vary with the statistics of the digital source. Timing, crosstalk, and equalization problems usually involve source statistics in some way. While redundant transmission codes may be used to help isolate system performance from source statistics, the isolation is not always complete, and such codes generate additional problems by increasing the required symbol rate or the number of levels per symbol which must be transmitted. In addition, with or without transmission codes, it is always easiest to analyze or predict system impairments if we assume that the source symbols are stochastically independent and equiprobable. We shall refer to such a source as "white" because of the obvious analogy to white Gaussian noise. Methods for "whitening" the statistics of digital sources without using redundant coding generally come under the heading of *scrambling*.

We describe here a nonredundant scrambling/descrambling method which in principle will satisfactorily whiten the statistics of *any* binary source. The technique is based upon the self-synchronizing digital data scrambler. Savage has shown[1] that this device is very effective at scrambling strictly periodic digital sources. In this paper it is proven that the same device will scramble any binary digital source to an arbitrarily small first- and second-order probability density imbalance $\delta$ if (*i*) the source is first passed through the equivalent of a binary symmetric memoryless channel with an arbitrarily small but nonzero error probability $\epsilon$, and (*ii*) the scrambler contains $M$ stages where $M \geqq 1 + \log_2 [(\ln 2\delta)/\ln (1 - 2\epsilon)]$. In other words, *at the cost of an arbitrarily small controllable error rate, one can "whiten" to any degree all the first- and second-order statistics of any binary source.* This relaxes the restriction frequently found in the literature in which the source is assumed *a priori* to produce only independent equiprobable symbols. An auxiliary result is that the above relation for $M$ is useful when designing a standard self-synchronizing scrambler for a given application. Heuristically speaking, the relation expresses the "power" of the scrambler by linking the "randomness" of the input and output to the scrambler length, $M$.

In Sections II and III of this paper we examine some properties of scramblers, maximal length sequences, and mod-2 sums of binary random variables. With these discussions as background, we prove the main theorem in Section IV. In Section V we derive bounds for the autocorrelation of the scrambled sequence. Section VI contains some practical considerations involved in applying the theorem of Section IV. Beacuse they add insight, we give simple direct proofs for the lemmas and theorem of Sections III and IV.

## II. SCRAMBLERS AND MAXIMAL LENGTH SEQUENCES

Figure 1 shows a five-stage self-synchronizing scrambler and descrambler.[1] As seen, both are linear sequential filters, the scrambler utilizing feedback paths and the descrambler feedforward paths. Each cell represents a unit delay. We restrict our attention to the binary case and use the symbols $\oplus$ and $\sum$ to denote mod-2 addition. Representing the data as shown, we have

$$b_k = a_k \oplus b_{k-3} \oplus b_{k-5}$$

and

$$c_k = b_k \oplus b_{k-3} \oplus b_{k-5} = a_k,$$

which shows that the descrambled sequence is identically equal to the

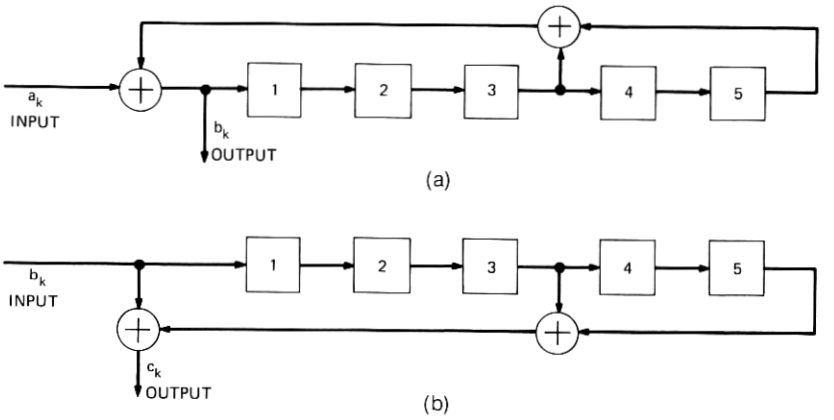


Fig. 1—(a) Five-stage scrambler. (b) Five-stage descrambler.

original data sequence. The descrambler is self-synchronizing because the effect of a channel error, insertion, or deletion lasts only as long as the total delay of the register, five bit-intervals in this example.

Let us consider the general scrambler of Fig. 2a with the input stream disconnected. Under such a condition, the scrambler becomes a sequence generator whose output must ultimately become periodic because (*i*) future states of the register are completely determined by the present state (the *state* of the register is the contents of its stages) and (*ii*) only the finite number $2^M$ states are possible, where $M$ equals the number of stages. One of these, the all-zeros state, simply leads to an all-zeros output. Discounting this state, we see that the longest possible period from the generator must be $2^M - 1$ bits. It is proven in the literature[2,3] that with the proper choice of feedback taps we can generate such a *maximal length sequence* for any $M$.

Registers which generate maximal length sequences make very effective scramblers because of their ability to *dissociate* one scrambler output bit from another. This property will enable us to show that two arbitrarily chosen output bits tend to be very weakly correlated. We state this essential property here in the form of a lemma.

*Lemma 1: From Fig. 2a it is evident that each "b" bit is equal to a lengthy mod-2 summation of selected "a" bits. Choose two bits, $b_m$ and $b_n$, $m > n$, and define $J_{mn}$ to be the number of "a" bits which enter the summation for $b_m$ but* not *the summation for $b_n$. That is, $b_m$ is dissociated from $b_n$ by the mod-2 sum of $J_{mn}$ "a" bits.*

*Then, if $n > 2^{M+1}$ (that is, the scrambler has processed at least $2^{M+1}$*

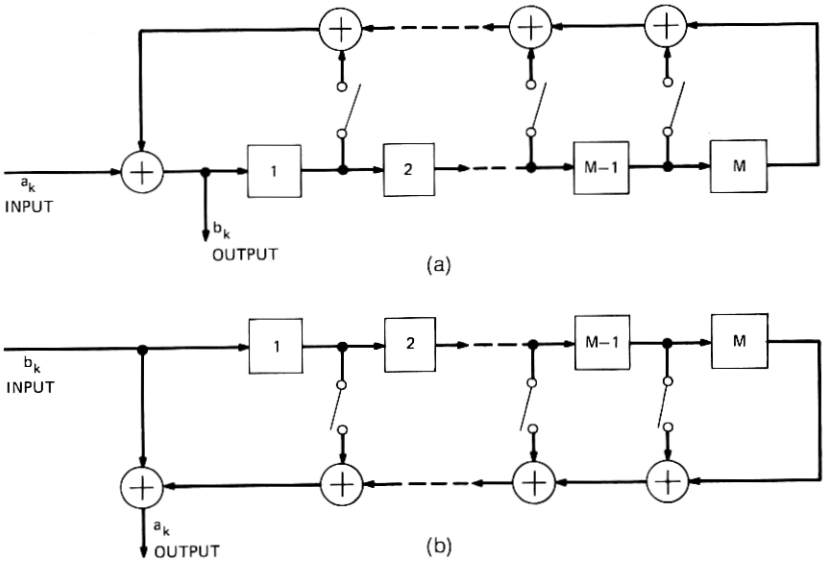


Fig. 2—(a) $M$-stage scrambler. (b) $M$-stage descrambler.

*"a" bits*),

$$J_o \equiv \min_{m,n} [J_{mn}] = 2^{M-1}. \tag{1}$$

*In other words, after a settling time of* $2^{M+1}$ *bits, any chosen pair of output bits will differ by the mod-2 sum of at least* $2^{M-1}$ *input bits.*
*Proof*: See appendix.

## III. MOD-2 SUMS OF BINARY RANDOM VARIABLES

Throughout this paper we assume that a data sequence may be modeled as a sequence of binary random variables defined on a suitable probability space. In this section we state as lemmas two essential properties of mod-2 sums of binary random variables. Since the scrambler output is formed from mod-2 sums of input bits, these properties play a key role in determining the scrambler output characteristics. We include the proofs in the text because the equations involved will be useful later on.

*Lemma 2: Consider two independent binary random variables* $r_1$ *and* $r_2$. *A third binary random variable* $r_3 = r_1 \oplus r_2$. *Let*

$$p_i = P(r_i = 1) = 1 - P(r_i = 0), \qquad i = 1, 2, 3.$$

*Then*

$$|p_3 - \tfrac{1}{2}| \leqq \min\left[|p_2 - \tfrac{1}{2}|, |p_1 - \tfrac{1}{2}|\right]$$

*with equality if and only if* $p_1$ *or* $p_2 = \tfrac{1}{2}$, 0, *or* 1. *In other words,* $r_3$ *is as close or closer to being equiprobable than either* $r_2$ *or* $r_1$.

*Proof*: Since $r_1$ and $r_2$ are independent,

$$p_3 = p_2(1 - p_1) + p_1(1 - p_2). \tag{2}$$

Let

$$d_i = p_i - \tfrac{1}{2} \qquad i = 1, 2, 3.$$

Then by substitution

$$|d_3| = 2|d_1||d_2|.$$

But since

$$|d_i| \leqq \tfrac{1}{2},$$

we have

$$|d_3| \leqq \min\left[|d_1|, |d_2|\right]$$

with equality if and only if $|d_1|$ or $|d_2| = 0$ or $\tfrac{1}{2}$.

*Corollary to Lemma 2: If* $p_1 = \tfrac{1}{2}$, *then* $p_3 = \tfrac{1}{2}$ *and* $r_3$ *is independent of* $r_2$.

*Proof*: Since $r_3 = r_1 \oplus r_2$, $P(r_3 = 1 | r_2 = 1) = 1 - p_1 = \tfrac{1}{2}$. But by eq. (2), $p_3 = \tfrac{1}{2}$. Thus, $P(r_3 = 1 | r_2 = 1) = P(r_3 = 1) = \tfrac{1}{2}$, which implies $r_3$ and $r_2$ are independent.

*Lemma 3: Consider now a sequence of independent binary random variables* $\{r_k, k = 1, 2, \cdots\}$ *with*

$$P(r_k = 1) = 1 - P(r_k = 0) = \epsilon \quad \text{for all } k.$$

*We form the mod-2 sum*

$$R_n = \sum_{k=1}^{n} r_k \tag{3}$$

*and let* $P_n = P(R_n = 1)$. *Then*

$$P_n = \tfrac{1}{2}[1 - (1 - 2\epsilon)^n]; \qquad n \geqq 1. \tag{4}$$

Note that, as $n \to \infty$, $P_n$ converges to $\tfrac{1}{2}$ for all $0 < \epsilon < 1$. However, we shall be concerned only with finite values for $n$.

*Proof*: By applying eq. (2) repeatedly, it is easily shown that the sequence $P_n$ satisfies

$$P_n = (1 - 2\epsilon)P_{n-1} + \epsilon; \qquad n \geqq 2,$$

and

$$P_1 = \epsilon.$$

The solution to this first-order linear difference equation is given by eq. (4).

## IV. A UNIVERSAL DIGITAL DATA SCRAMBLER

With the help of the lemmas, we may now derive the main result. We model the source as a device which generates a sequence of binary random variables $\{s_k\}$ with completely unknown statistics. Our goal is to find a scrambling/descrambling method such that the scrambled sequence $\{b_k\}$ will have statistics which approach those of the independent equiprobable ("white") sequence $\{w_k\}$. If we attempt to scramble $\{s_k\}$ directly as in Fig. 2, we are faced with a dilemma. The scrambler simply provides a one-to-one mapping between its input and output. As long as we have no knowledge or control of the statistics of $\{s_k\}$, the statistics of $\{b_k\}$ must likewise remain unknown and uncontrolled. Hence, the self-synchronizing scrambler *alone* cannot be universal.

Instead of scrambling directly, we proceed as shown in Fig. 3. The source output is first passed through the equivalent of a binary symmetric memoryless channel (BSC) with crossover probability $\epsilon > 0$. Remarkably, no matter how small $\epsilon$ may be, this modification of the source sequence is sufficient to guarantee that the first- and second-order probability densities for $\{b_k\}$ will approach those of $\{w_k\}$ to within an arbitrarily small difference $\delta$. The only requirement is that $M$, the length of the scrambler, be dependent upon the choice of $\epsilon$ and $\delta$. This is the essence of the theorem which we derive below. (We note in passing that the descrambled sequence will now differ from the original source sequence by the error rate $\epsilon$, but since $\epsilon$ may be chosen arbitrarily small, we assume for now that this is of no consequence.)

To begin, we observe that because of the BSC the scrambler input sequence may be written

$$a_k = s_k \oplus r_k, \qquad k = 0, 1, 2, \cdots, \tag{5}$$

where

$$P(r_k = 1) = 1 - P(r_k = 0) = \epsilon.$$

From Lemma 1 we have seen that the action of the $M$-stage scrambler is to dissociate any chosen pair of bits $(b_m, b_n)$ by the mod-2 sum of at least $2^{M-1}$ "$a$" bits. Let us assume that $b_m$ and $b_n$ are dissociated by *exactly* $2^{M-1}$ "$a$" bits and that they are related by

$$b_m = b_n \oplus \sum_{l=1}^{2^{M-1}} a_l. \tag{6}$$

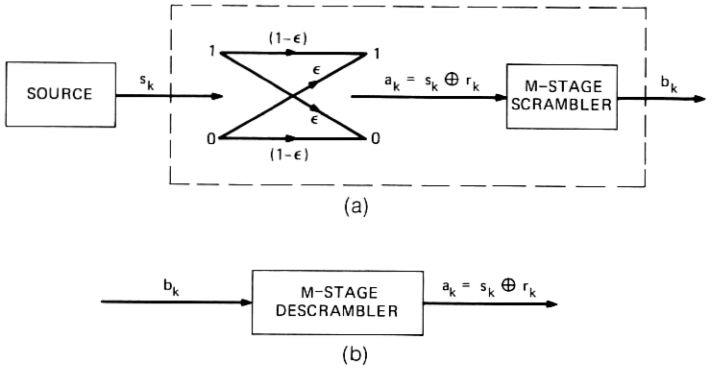


Fig. 3—(a) Universal scrambler. (b) Descrambler.

(Here the subscript $l$ is unrelated to the original position of $a_l$ in the scrambler input stream.) In what follows we show that $P(b_m = 1) \approx \frac{1}{2}$ and that $b_m$ and $b_n$ are nearly independent. For these purposes the use of eq. (6) represents a worst-case analysis. By substitution from eq. (5) we may write

$$b_m = \left[ b_n \oplus \sum_{l=1}^{2^{M-1}} s_l \right] \oplus \left[ \sum_{l=1}^{2^{M-1}} r_l \right] \tag{7}$$

$$\equiv \quad A \quad \oplus \quad R$$

where $A$ and $R$ equal the first and second bracketed terms, respectively. Since the bits comprising $R$ are independent from those comprising $A$, $R$ is independent of $A$. Furthermore, by Lemma 3,

$$P(R = 1) = \tfrac{1}{2}[1 - (1 - 2\epsilon)^{2^{M-1}}]. \tag{8}$$

Therefore, by Lemma 2, no matter what the value of $P(A = 1)$,

$$\delta' \equiv |P(b_m = 1) - \tfrac{1}{2}| \leqq \tfrac{1}{2}[(1 - 2\epsilon)^{2^{M-1}}] \equiv \delta. \tag{9}$$

It follows that so long as $\epsilon > 0$ we may force $\delta$ and $\delta'$ to be arbitrarily small by choosing a large enough $M$. Specifically, for a given $\delta$,

$$M \geqq 1 + \log_2 \left[ \frac{\ln 2\delta}{\ln (1 - 2\epsilon)} \right]; \qquad 0 < \epsilon < \tfrac{1}{2}. \tag{10}$$

Since $\delta$ may be made arbitrarily small, the density function $p(b_m)$ may be made nearly white, and it follows that all first-order statistics of the scrambled sequence may be made nearly white.

Our having shown $P(b_m = 1) \approx \frac{1}{2}$ does not by itself show that the source has been effectively scrambled. For example, consider a sequence

$\{x_n\}$ which consists of consecutive blocks of 100 symbols each. All the symbols in each block are alike; with probability $\frac{1}{2}$ they are all ones, and with probability $\frac{1}{2}$ they are all zeros. Here $P(x_n = 1) = \frac{1}{2}$ for all $n$, yet the sequence has a very "nonrandom" nature. The implication is that to determine the effectiveness of the scrambler, we must also evaluate the statistical dependence between scrambler output bits.

By definition, the variables $b_m$ and $b_n$ are independent if

$$p(b_m, b_n) - p(b_m)p(b_n) = 0.$$

Accordingly, we define the function

$$\begin{aligned} d(b_m, b_n) &\equiv p(b_m, b_n) - p(b_m)p(b_n) \\ &= p(b_m|b_n)p(b_n) - p(b_m)p(b_n) \end{aligned} \tag{11}$$

and show that the universal scrambler (Fig. 3) bounds the maximum value of $|d(b_m, b_n)|$.

We do a worst-case analysis by assuming that $b_m$ and $b_n$ are related by eq. (7). Further, we ignore the "$s$" bits appearing in eq. (7) because, being independent of $R$, they can only weaken the dependence between $b_m$ and $b_n$. Hence, we may compute the maximum value of $|d(b_m, b_n)|$ by assuming

$$b_m = b_n \oplus R. \tag{12}$$

From eqs. (8) and (9) we note $P(R = 1) = \frac{1}{2} - \delta$ and for convenience we temporarily let $P(b_n = 1) = b$. Substituting these relations and eq. (12) into eq. (11), we find that

$$|d(b_m, b_n)| = 2\delta[b(1 - b)];$$

for

$$b_m, b_n = 0, 1; \qquad m > n > 2^{M+1}.$$

Hence, for $b = \frac{1}{2}$ we obtain the general result

$$|d(b_m, b_n)|_{\max} = \delta/2.$$

Since $\delta$ may be forced arbitrarily small if $M$ is given by eq. (10), it follows that any pair of output bits may be made nearly independent, and we may whiten to any degree all the second-order statistics of the source sequence.

We may also show that the joint (second-order) density $p(b_m, b_n)$ approaches that for the white sequence. The derivation of eqs. (6) to (9) shows that both the density $p(b_i)$ and the conditional density $p(b_i|b_j)$ must have values on the interval $[(\frac{1}{2} - \delta), (\frac{1}{2} + \delta)]$ for all

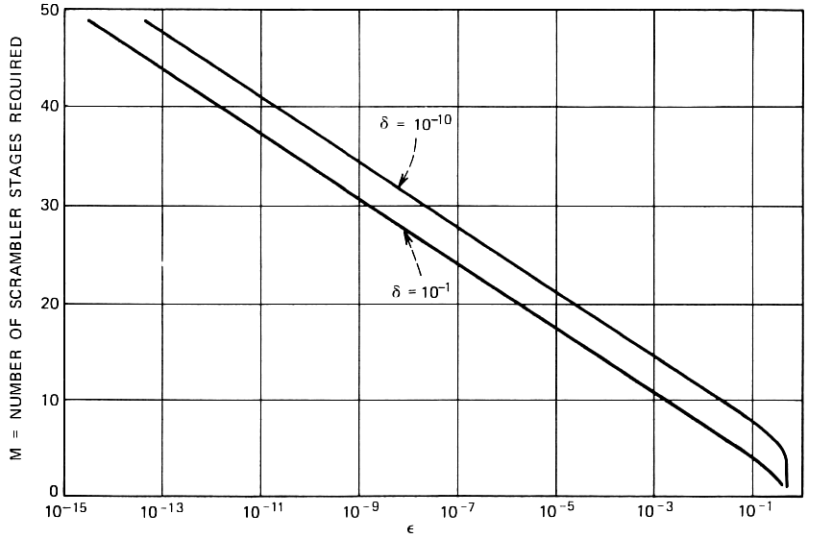


Fig. 4—Scrambler stages required as a function of $\epsilon$ and $\delta$.

possible values of $b_i$ and $b_j$. Hence,

$$(\tfrac{1}{2} - \delta)^2 \leqq p(b_m, b_n) = p(b_m|b_n)p(b_n) \leqq (\tfrac{1}{2} + \delta)^2,$$

or

$$|p(b_m, b_n) - \tfrac{1}{4}| \leqq \delta + \delta^2 \approx \delta,$$

where

$$b_m, b_n = 0, 1; \qquad m > n > 2^{M+1}.$$

For the white sequence $\{w_k\}$, we know $p(w_m, w_n) = \frac{1}{4}$ for $w_m, w_n = 0, 1$. Thus the joint density $p(b_m, b_n)$ may be whitened to any degree by choice of $\delta$, $\epsilon$, and $M$.

The discussion above constitutes a proof of the following theorem.

*Universal Scrambler Theorem: A binary source with unknown output statistics is connected to a binary symmetric memoryless channel and an M-stage self-synchronizing scrambler as shown in Fig. 3. The channel has error probability $\epsilon$ where $0 < \epsilon < \frac{1}{2}$. The scrambler output is represented by a sequence of random variables $\{b_n, n = 0, 1, \cdots\}$ and we define $p(b_n)$ to be the first-order and $p(b_m, b_n)$ the second-order density functions for $\{b_n\}$. Then for all $\delta > 0$; $m > n > 2^{M+1}$, and $b_m, b_n = 0, 1$,*

$$|p(b_n) - \tfrac{1}{2}| \leqq \delta, \tag{13}$$

*and*

$$|p(b_m, b_n) - \tfrac{1}{4}| \leqq \delta + \delta^2, \tag{14}$$

*provided that*

$$M \geqq 1 + \log_2 \left[ \frac{\ln 2\delta}{\ln (1 - 2\epsilon)} \right]. \tag{15}$$

Figure 4 shows the relation between $M$, $\epsilon$, and $\delta$. As seen, $M$ is primarily dependent upon $\epsilon$. This may be clarified by rewriting eq. (15) for small values of $\epsilon$. We then obtain

$$M \geqq \log_2 (1/\epsilon) + \log_2 [\ln (1/2\delta)]; \qquad \epsilon \ll \tfrac{1}{2}.$$

The primary importance of this theorem is conceptual. To avoid inordinate difficulties, many analyses in the literature of digital transmission must assume *a priori* that the digital source is white. The theorem relaxes this restriction by showing that in concept the first- and second-order statistics of *any* source may be made asymptotically white. The practical application of this theorem is discussed in Section VI.

## V. AUTOCORRELATION OF THE SCRAMBLED SEQUENCE

An important second-order statistic of the scrambled sequence is its autocorrelation. We define the autocorrelation as the expectation

$$R(k) = E[b_n b_{n+k}],$$

and for convenience we let the value of $b_n$ be $+1$ or $-1$. Clearly, $R(0) = 1$. For $k \neq 0$, we compute a bound on $|E[b_n b_{n+k}]|$. Following the argument which led to eq. (12), we have

$$|E[b_n b_{n+k}]| \leqq |E[b_n (b_n \oplus R)]|; \qquad n, n + k \geqq 2^{M+1}, \; k \neq 0.$$

By definition,

$$E[b_n b_m] = \sum_i \sum_j ij P(b_m = i | b_n = j) P(b_n = j).$$

We let $b_m = b_n \oplus R$ and for convenience

$$P(b_n = 1) = b = 1 - P(b_n = -1).$$

Substituting, the dependency on $b$ vanishes, leaving us with

$$E[b_n (b_n \oplus R)] = 2\delta.$$

Hence,

$$\begin{aligned} R(k) &= 1 \qquad \text{for} \quad k = 0, \\ |R(k)| &\leqq 2\delta \qquad \text{for} \quad k \neq 0. \end{aligned} \tag{16}$$

Note that, by forcing $\delta$ to a small value with proper choice of $M$ and $\epsilon$, this autocorrelation approaches that for a "white" digital source

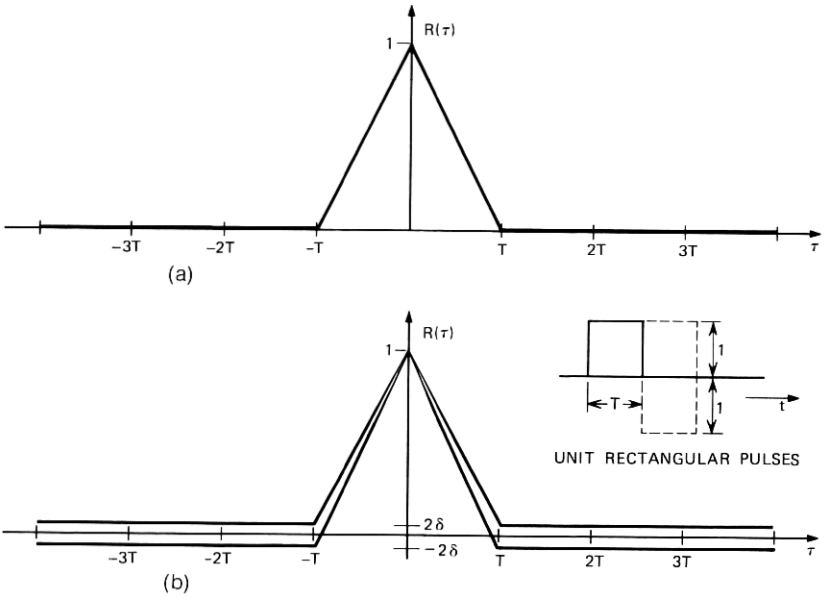


Fig. 5—(a) Autocorrelation for white sequence $\{w_k\}$. (b) Autocorrelation bound for scrambled sequence $\{b_k\}$.

which has $R(k) = 0$, $k \neq 0$. This is shown graphically in Fig. 5 which shows the autocorrelation of "white" and scrambled sources for unit rectangular pulses.

## VI. PRACTICAL CONSIDERATIONS

In practice, the binary symmetric channel required by the theorem might be implemented as shown in Fig. 6. The bit $r_k$ is a logic "one" only when the level from the noise generator exceeds some threshold. The threshold is set such that $P(r_k = 1) = \epsilon$. The noise source need not be white, but values of $n(t)$ separated by the baud interval should

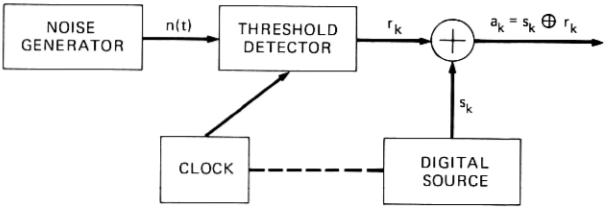


Fig. 6—One possible implementation of the BSC.

be independent. In principle, the combination of this simulated BSC and an $M$-stage self-synchronizing scrambler will form a universal scrambler capable of satisfactorily whitening the statistics of any binary source. Such a scrambling structure could be used wherever randomized bit statistics are essential and a small error rate can be tolerated (or perhaps corrected by an error-correcting code).

There are, of course, good reasons to avoid actual implementation of the BSC. First, it may be difficult to generate the $r_k$ sequence accurately if $\epsilon$ is very small. Second, the deliberate generation of errors, if not impractical, is at least unpalatable. Third, and most important, many commonly encountered sources do not need it. Self-synchronizing scramblers have been used successfully without any prior randomization of the source.[4] In this section we consider the operation of the scrambler without the BSC and show how a designer may use eq. (15) to estimate the required scrambler length for a given application.

From eqs. (2) and (5) we deduce that the net effect of the binary symmetric channel in Fig. 3 is

$$\epsilon < p(a_k \mid a_0, a_1, \cdots, a_{k-1}) < 1 - \epsilon; \qquad a_k = 0, 1, \tag{17}$$

for all $k$. In other words, because of the BSC there remains a small uncertainty as to the value of any "$a$" bit, even though all the other "$a$" bits might be known. As shown in the theorem, this and the dissociation property are sufficient to guarantee effective scrambling. Hence, if the designer knew to begin with that the source itself had the characteristic

$$\epsilon < p(s_k \mid s_0, s_1, \cdots, s_{k-1}) < 1 - \epsilon; \qquad s_k = 0, 1, \tag{18}$$

then no BSC would be necessary, and eq. (15) could be applied directly. For example, bit streams encoded from analog waveforms (such as frequency-division multiplexed speech) often have such a property, and a value for $\epsilon$ could be obtained from the coding rule and the amplitude distribution of the analog signal.

For those cases in which a value for $\epsilon$ cannot be computed, let us assume that the designer has at least some knowledge of the source pulse density. He could then proceed by estimating a nominal value for $\epsilon$ and then decreasing the value to allow some margin. For example, a source which produces bit streams known to vary from 10 to 90 percent "ones" over short periods (say, several hundred bits) would have a nominal $\epsilon = 0.1$. It seems reasonable to allow at least one order of magnitude "margin" in the estimate, resulting in $\epsilon = 0.01$. Then from Fig. 4 we see that an eight-stage scrambler should be sufficient.

Of course, estimating $\epsilon$ from the source pulse density does not guarantee that eq. (18) really holds, but if the source sequence is not strictly periodic (the case covered comprehensively by Savage), it is a reasonable procedure. The point here is that even when we are unwilling to commit deliberate errors to guarantee fixed source statistics, we may still use eq. (15) to estimate how large a scrambler is required. Heuristically speaking, eq. (15) is an expression for the "power" of the scrambler, relating the "randomness" of the input and output to the number of scrambler stages.

## VII. CONCLUSIONS

We have shown that at the cost of an arbitrarily small error rate it is possible to "whiten" to any degree all the first- and second-order statistics of any binary digital source. This relaxes the restriction frequently found in the literature in which the digital source is assumed *a priori* to produce only independent equiprobable symbols. The key equation in our result [eq. (15)] is useful when designing a standard self-synchronizing scrambler for a given application.

We leave unsolved the problem of whether universal scramblers exist for the $M$-ary source.

## VIII. ACKNOWLEDGMENTS

I wish to thank M. B. Romeiser for editorial suggestions and Miss J. M. Michel for assistance in computer programming. I also had the benefit of stimulating discussions on the subject of scrambling with T. M. Chien and R. J. Deaton.

## APPENDIX

### *Proof of Lemma* 1*

For convenience, we assume that in Fig. 2a the scrambler initially contains all zeros. Since each scrambler output bit is ultimately a mod-2 summation of selected input bits, we may write

$$b_n = \sum_{k=0}^{n} h_k a_{n-k}, \tag{19}$$

where the binary sequence $h_k$ performs the selection. We note that if $a_0 = 1$ and $a_i = 0$ for all $i > 0$, then $\{b_n\} = \{h_n\}$. But under these

* Independently of the author, U. Henriksson has developed[5] a proof of a similar lemma.

conditions, as described in Section II, $\{b_n\}$ will be a maximal length sequence. Hence, $\{h_n\}$ must itself be a maximal length sequence.

Now we consider the two output bits $b_n$ and $b_m$. We wish to count the number of "$a$" bits which entered the summation for $b_m$ but not $b_n$. We have

$$b_n = \sum_{k=0}^{n} h_k a_{n-k} \qquad b_m = \sum_{k=0}^{m} h_k a_{m-k}. \tag{20}$$

(a) (b)

Since $m > n$,

$$b_m = \sum_{k=0}^{m-n-1} h_k a_{m-k} \oplus \sum_{k=m-n}^{m} h_k a_{m-k}$$

$$= \sum_{k=0}^{m-n-1} h_k a_{m-k} \oplus \sum_{k=0}^{n} h_{m-n+k} a_{n-k} \tag{21}$$

Examination of the subscript range shows that all the "$a$" bits selected by the first summation in eq. (21) are unique to $b_m$. By comparing the second summation with eq. (20a) we see that the additional "$a$" bits which enter $b_m$ but not $b_n$ are those for which

$$h_{m-n+k} - h_{m-n+k} h_k = 1.$$

Hence,

$$J_{mn} = \sum_{k=0}^{m-n-1} h_k + \sum_{k=0}^{n} [h_{m-n+k} - h_{m-n+k} h_k],$$

or

$$J_{mn} = \sum_{k=0}^{m-n-1} h_k + \sum_{k=0}^{n} h_{m-n+k} - \sum_{k=0}^{n} h_{m-n+k} h_k, \tag{22}$$

where addition is now in the usual sense.

We examine this expression in detail, recalling that the sequence $\{h_k\}$ has period $p = (2^M - 1)$ and the given condition $n > 2^{M+1}$.

*Case (1)*: If $m - n = Kp$, $K = 1, 2, \cdots$, then

$$h_k = h_{m-n+k} = h_{m-n+k} h_k$$

for all values of $k$. Hence the second and third summations cancel. But then the first summation contains $K$ periods of a maximal length sequence. Since each period contains exactly $2^{(M-1)}$ ones,[6] the first summation totals at least $2^{(M-1)}$.

*Case (2)*: If $m - n \neq Kp$, then it is easily shown[6] that the sequence formed by the term-by-term product $h_{m-n+k} h_k$ has period $p$ and con-

tains $2^{(M-2)}$ ones per period. The sequence $\{h_{m-n+k}\}$ contains $2^{(M-1)}$ ones per period. Hence the net contribution of the second and third summations is $2^{(M-2)}$ ones per period. Since $n > 2^{M+1} > 2p$, the summations cover at least two periods. Thus their net total is at least $2^{(M-1)}$.

Thus for either case,

$$J_0 = \min_{m,n>2^{M+1}} [J_{mn}] = 2^{M-1}.$$

REFERENCES

1. Savage, J. E., "Some Simple Self-Synchronizing Digital Data Scramblers," B.S.T.J., *45*, No. 2 (February 1967), pp. 449–487.
2. Gallager, R. G., *Information Theory and Reliable Communication*, New York: John Wiley and Sons, 1968, pp. 225–238.
3. Peterson, W. W., *Error Correcting Codes*, Cambridge: M.I.T. Press, 1961, pp. 251–270.
4. Fracassi, R. D., and Tammaru, T., "Megabit Data Service with the 306A Data Set," Bell Laboratories Record, *49*, No. 10 (November 1971), pp. 310–315.
5. Henriksson, U., "On a Scrambling Property of Feedback Shift Registers," IEEE Trans. on Commun., *20*, No. 5 (October 1972), pp. 998–1001.
6. Golomb, S. W., *Shift Register Sequences*, San Francisco: Holden-Day, 1967, pp. 23–59, 88.