

OPTIMUM CODE PATTERNS FOR PCM SYNCHRONIZATION

Merwin W. Williard
Symetrics, Incorporated
Satellite Beach, Florida

Preface

The author published this article in 1962 as part of the Proceedings of the National Telemetry Conference held in Washington D. C. These Proceedings are not readily available to many members of IEEE 802.11. For their benefit, the following reprint was created by electronic scanning. I tried hard to correct all the errors introduced by the scanning process but it is likely I missed a few. The reader is therefore cautioned that (a) this reprint may contain errors not present in the original version; (b) the author, M. W. Williard, is not responsible for those new errors and (c) the reader uses any and all information here, be it wrong or right, entirely at his own risk. Mr. Williard subsequently performed computerized searches for longer words; these are included here as Table VI. He is now an independent consultant. His mailing address is P.O.B. 701967, St. Cloud, FL, 34770-1967 and his phone number is 1 407 892 1503 ---- John McKown

Introduction

Telemetry systems utilizing Pulse Code modulation for transmission require sync information intermixed with the data for reconstruction of the channel structure during data collection. The proper recognition and utilization of the sync information become a complex problem, where errors appear in the received information as a result of noise in the r-f transmission link.

The usual procedure for inserting sync information is to place a repetitive pattern of preselected polarity bits, which form a sync word, at fixed intervals in the transmitted signal. Nearly all present PCM systems utilize small patterns placed between each sample or word as the primary form of digital sync. Utilization of word sync, however, requires that additional secondary sync information be placed once per frame to define the beginning of the cycle of samples.

A recent study by the author [1] indicated that if the primary digital sync patterns are lengthened and spaced farther apart, increased speed of sync detection can be realized with a smaller percentage of the total transmitted bits devoted to sync, and with no degrading of the reliability of maintaining sync in the presence of noise.

One of the assumptions made in arriving at these conclusions is that the probability of the preselected pattern of sync bits

occurring in any group of consecutive bits, other than the group made up of the complete set of true sync bits, is $(0.5)^n$, where n is the number of bits in the pattern. This assumes that the occurrence of "1" and "0" bits in the data can be considered random, and equally likely. However, the assumption was extended to also include groups of consecutive bits which contain part of the group of true sync bits, and part data bits adjacent to the true sync pattern.

With the assumption that all groups of data bits are random, the length of the sync pattern is the only factor affecting the probability-of-occurrence of the pattern in a group of all data bits. With the assumption that errors produced by noise in detection of each individual bit received are random, the length of the sync pattern and the number of errors allowed in detection of the pattern are the only factors affecting the probability of finding the correct pattern in its true location.

The first objective of this study is to point out the problems associated with the detection of apparent sync patterns made up partly of random bits and partly of bits in the true sync pattern. The second objective is to define characteristics of patterns which minimize this problem. Finally, the choice of secondary sync patterns is discussed.

The Problem of Choice of Patterns

The problem of choice of patterns can be illustrated by an example. The diagram below represents part of a serial string of transmitted data with a 7-bit sync pattern inserted. The X's represent data bits on each side of a sync pattern made up of seven consecutive "0" bits.

X	X	X	X	X	X	X	0	0	0	0	0	0	0	X	X	X	X	X			
<-----A----->														<--	-----B----->						
													<-----C----->								

Before sync is established, all data must be reviewed to find the sync pattern --- in this case seven consecutive "0" bits. The seven bits in group A are considered random and, therefore, the probability-of-occurrence of seven consecutive "0" bits in this group is $(0.5)^7$. The seven bits in group B contain only six random data bits. The bit on the left end of group B is the right end bit of the true sync pattern. Since it is always transmitted as a "0", it is always identical to the "0" desired in the left end of

the sync pattern. In the absence of noise which might produce a "1" in place of this "0", only six random bits adjacent to the true sync pattern need to turn up "0"s. This can happen with a probability of $(0.5)^6$. This means the probability of group B containing all "0"s is double that of group A made up of all random bits.

Group C always contains six bits which are identical to the desired sync pattern. The probability-of-occurrence of all "0"s in group C is, therefore, 2^6 (or 64) times more probable than the probability-of-occurrence of a sync pattern in a group of all random data bits.

The term "overlap group" is used to describe groups of consecutive bits containing the same number of bits as the sync pattern, but made up partly of sync bits and partly of adjacent random data bits. Group B and C in the example above are considered overlap groups.

The problem indicated by the example above is that the probability-of-occurrence of what appears to be a true sync pattern in an overlap group of bits can be much larger than the probability-of-occurrence of the pattern in a set of random data bits.

In the absence of noise, which may cause errors in the detection of the received signal, the solution of this problem is simple. The pattern of bits 0000001 has the characteristic that for all overlap groups, at least one bit in the group conflicts with the pattern of the true sync bits. This is not true for a group of bits which begins and ends with the same bit, or group of bits, such as 0101001.

When random errors result in the false detection of the sync bits, the problem becomes more complex. The previous study by the author¹ indicated that it was quite feasible to design a sync system capable of detecting and maintaining sync in the presence of ten percent random bit errors. This figure was chosen on the basis that data received containing any more errors than this could be scarcely usable.

Criteria For Selection of Optimum Synchronization Patterns

The problem of optimum sync patterns has been considered by others. Barker [2] makes the statement that "the form of the pattern should be such that the probability of this type of error (referring to overlap occurrence of the pattern) is minimized." Then he proceeds to define patterns which, instead of minimizing the total probability-of-occurrence, minimize the maximum of the expected value.

Goode and Phillips [3] define a "sample variance" and state that, "for a given code length, n, the code with the smallest sample

variance will have the minimum probability of a false sync indication and, therefore, will be optimum."

This present study was pursued because there was much talk about the sacredness of patterns of bit length equal to one less than a multiple of four. The author believed that for any length pattern, n, there was a best pattern of length n+1 which was nearly twice as good as the best pattern of length n. This seemed obvious since adding a bit added another 0.5 factor to the probability-of-occurrence of the pattern in a set of random bits, and must aid also in the reduction of probability-of-occurrence of the pattern in overlap conditions if a good sync pattern is chosen.

Any criterion for selection of optimum sync patterns based on a particular detection technique is restrictive. Regardless of the detection technique, if the correct combination of "1" and "0" bits happens to occur, in the received signal, any conceived detector will define it as a possible true sync pattern. The criteria for selection of optimum sync patterns should be based on the minimum probability of false occurrence of the pattern in the received signal. This criterion is used throughout this study in the selection and evaluation of more optimum patterns.

Evaluation of Pattern

Consider the 7-bit sync pattern suggested by Barker, namely 0001101. With any degree of overlap, there is always a conflict in at least one bit. If, however, noise sufficient to produce 10% random errors in bit detection is considered, the group of seven bits which contain one of the sync bits and six data bits (one bit overlap) has a probability-of-occurrence of an apparent sync pattern of

$$(0.5)^6(0.1)$$

The factor (0.1) is the probability of the overlapped "1" being detected as a "0". Dividing by the probability-of-occurrence of the pattern made up of all random data bits results in

$$\frac{(0.5)^6(0.1)}{(0.5)^7} = 0.2$$

The probability-of-occurrence of an apparent sync pattern in the group containing the end sync bit and six random data bits is only 20 percent as great as the probability-of-occurrence of the pattern in a group of seven random data bits. When the overlap is two, the probability of the "1" in the overlap appearing as a "0" due to errors is 0.1 but there is only a 0.9 probability that the overlapped "0" in the data will appear as a "0". In conjunction with the random five bits, the probability-of-occurrence of an apparent sync pattern in the overlap-two condition relative to the

probability-of-occurrence of the pattern in a group of random data bits is

$$\frac{(0.5)^5(0.1)(0.9)}{(0.5)^7} = 0.36$$

The following definitions apply to the ensuing discussion:

n = number of bits in the sync pattern

m = number of bits in any group which are actually part of the true sync pattern (i.e., number of bits which overlap, or degree of overlap)

c = number of bits in the overlap which, as transmitted, are opposite to or conflict with bits expected in given bit positions.

I = number of bits in the overlap which, as transmitted, are identical to bits expected in given bit positions.
(therefore, c+I=m)

e = random bit error rate resulting from noise on the incoming signal.

In general, for any degree of overlap m and any length pattern n, the probability-of-occurrence of the sync pattern in the overlap group relative to the probability-of-occurrence of the pattern in n random data bits is defined as Rm.

$$R_m = \frac{(0.5)^{n-m}(1-e)^I e^c}{(0.5)^n}$$

$$R_m = 2^m(1-e)^I e^c = 2^m(1-e)^{m-c} e^c$$

Note that n cancels out of this equation. Therefore, regardless of the length of a pattern, the probability-of-occurrence of the pattern in any overlap group relative to the probability-of-occurrence of that pattern in any group of all random data bits is only a function of the number of bits which overlap (m), the number of these that, as transmitted, conflict with bits in the group under consideration (c), and the error rate (e). Henceforth Rm will be referred to as the relative probability-of-occurrence. Table I is a listing of Rm for m up to 7 and c from 0 to m, evaluated for an error rate of 10 percent. The table is easily constructed for greater m, since for any m the last entry at c=m is $2^m(0.1)^m$ and each succeeding entry above it is nine times the next lower entry.

Continuing the evaluation of the Barker 7-bit pattern of 0001101, it can be seen that for the six possible overlap groups the following is true:

OVERLAP	m	c	Rm
6	6	3	0.0467
5	5	3	0.0259
4	4	2	0.1296
3	3	2	0.072
2	2	1	0.36
1	1	1	0.2

It is rather interesting to note that in each group of bits containing part sync bits and part data bits the relative probability-of-occurrence is less than one for each degree of overlap. Since R1, R2, R3, each represent the relative probability-of-occurrence of a sync pattern in each degree of overlap, m = 1, 2, 3, respectively, then Rt will be defined as the sum of R1 through R(n-1) for any length pattern, n. Rt for the Barker 7-bit pattern results in 0.8342. This indicates not only that the relative probability-of-occurrence is less than 1 for each degree of overlap but that the total probability-of-occurrence of a false sync indication when scanning through all six degrees of overlap while approaching true sync is less than the probability-of-occurrence of the pattern in just one group of n random data bits.

Note that Rm is a function of whether bits are alike or different. Therefore, the complement of any pattern has an identical probability for each degree of overlap. Also, the reflected word, produced by reversing the pattern end-for-end, produces the same results. Therefore, the following set of patterns have identical overlap probabilities.

0001101	Basic
1110010	(complement)
1011000	(reflected)
0100111	(reflected complement)

As a means of terminology standardization, whenever reference is made to a pattern, the basic pattern will be the smallest binary number. It will be understood, however, that for the purposes of sync patterns, four sequences of "1"s and "0"s are implied.

Characteristics of Better Patterns

As stated earlier, Rm is independent of the length of the pattern (n), and only a function of degree of overlap (m), and number of conflicts in the overlap (c). A review of table I then indicates that a sync pattern which is to produce each Rm less than 1 must produce a minimum of one conflicting bit in one through three degrees of overlap, and at least two conflicting bits in three through seven degrees of overlap.

To meet the requirement that there is one conflict in one degree of overlap requires that the pattern begin and end with bits that

are not alike. Only patterns beginning with "0" and ending in "1" need to be considered since all patterns beginning in "1" and ending in "0" are complements of the pattern beginning in "0" and ending in "1".

What must the second and next-to-last bits be in order to develop at least one conflict in the overlap-2 condition?

```
0 - - - - 1
      0 - - - - 1
```

If the second bit is a "0" or if the next-to-last bit is a "1", one conflict will result. These two possibilities are represented by the patterns.

```
0 0 - - - X 1
0 X - - - 1 1
```

The dashes mean that any number of bits may be between these "start-and-end sets". The X's mean that the bit position may be either a "1" or a "0". Also notice that the second pattern is the reflected complement of the first pattern. Therefore, if all possible patterns which start with "00" and end in "1" are evaluated, and since the properties of a pattern and its reflected complement are identical, the second set is represented in evaluation of all members of the first set. The only 3-bit patterns which meet the condition of R1 and R2 each less than 1 are 001, its complement, reflection, and reflected complement. Note that the pattern 001 begins in 00 and ends in X 1. The X can be either "1" or "0" for longer patterns and still have R2 less than 1.

Now consider what the third and third-from-last bit in a pattern must be to make R3 less than 1 (at least one conflict).

```
0 0 - - - X 1
      0 0 - - - X 1
```

The condition is met with the third bit set at "0". If the third bit is a "1", then assigning the next-to-last as "1" will be sufficient. If the third is "1" and next-to-last "0", then the third-from-last better be a "1".

The three sets listed below represent all possible "start-and-end sets" which result in at least one conflict in each of the first three degrees of overlap.

```
0 0 0 - - - X X 1
0 0 1 - - - X 1 1
0 0 1 - - - 1 0 1
```

Note that when the X in the second set is assigned a "1", its reflected complement is a member of the first set. Thus, the second set need only be considered where X is assigned as a "0". The six possible 3-bit start-and-end sets are then:

```
0 0 0 - - - 0 0 1
0 0 0 - - - 0 1 1
0 0 0 - - - 1 0 1
0 0 0 - - - 1 1 1
0 0 1 - - - 0 1 1
0 0 1 - - - 1 0 1
```

Consider any possible sync pattern. Write down the pattern, its complement, reflection and reflected complement. Delete the three which have the highest equivalent binary value. If the remaining pattern does not begin and end with one of the above sets of bits, there will be less than one conflict in at least one of the first three degrees of overlap and either R1, R2, or R3 will be greater than one.

The only 4-bit patterns which have below-random autocorrelation at 10 percent bit error rate in the three possible degrees of overlap can be found from the 3-bit start-and-end sets. They must have the second and third bits identical to the third-from-last and second-from-last, respectively. This is only true of the first and fifth in the list. The below-random autocorrelation 4-bit patterns are then 0001 and 0011. Table II shows these and other best sync patterns.

The octal notation used here simplifies writing down patterns. The convention adopted is to start from the right end of any pattern of any length, break it into 3-bit groups and write down the octal equivalent of each three binary bit group. When very long patterns are to be listed, only the octal notation will be used. To convert from the octal notation to the actual pattern write out the binary coded equivalent of the octal number and fill in "0"s to the left until the specified pattern length is obtained.

Goode and Phillips [3] define sample variance as proportional to the square of the difference between the number of bits which are identical and the number which conflict. Minimum sample variance may well make the pattern look most like random data in overlap conditions but does not produce the pattern with minimum probability-of-occurrence of the sync pattern in noisy data in all degrees of overlap. The two 4-bit patterns listed in table II are a good example. The sample variance, calculated as specified by Goode and Phillips, is 0.176 for the first and only 0.093 for the second; but the simple fact that the first pattern has one more conflict in the overlap-2 condition makes it obvious that it is less likely to occur in noisy data conditions.

The next step is to find all 4-bit start-and-end sequences which meet the requirements that the numbers of conflicts in one, two, and three degrees of overlap are each at least one, and the conflicts in four degrees of overlap is at least two, to make R4 less than one. The work has been extended to find all start-and-end sequences up to six degrees of overlap for which the sequence of conflicts is a minimum of 111222. There are 15

four-bit start-and-end sequences, 47 five-bit start-and-end sequences, and 165 six-bit start-and-end sequences. Only one 5-bit pattern meets the requirements that the conflict sequence is below 1112. This pattern is 00101, with R_t equal to 1.2224. Considering all 5-bit start-and-end sequences produces two 6-bit patterns for which each R_m is less than one, while the 6-bit start-and-end sequences produce three 7-bit patterns, all of which have the characteristic that each R_m is less than one. These are shown in table II.

The work of extending the tables of start-and-end sequences became too laborious without a computer, but from the table of all 6-bit start-and-end sequences it is easy to pick out only those patterns in which the third, fourth, fifth and sixth bits match the sixth-, fifth-, fourth-, and third-from-the-last bits. These are all 8-bit patterns in which R_m for m equal to 1 through 6 is known to be less than one. It remains to find out if R_7 is also less than one. To get R_7 less than one requires two conflicts in overlap-7 position. Eight 8-bit patterns meet the requirement that each R_m is less than one. These are listed in table II.

An identical procedure was used to find all 9-, 10-, and 11-bit patterns for which each R_m was less than one. Sixteen 9-bit patterns, 31 ten-bit patterns and 57 eleven-bit patterns represent all patterns of each length for which each R_m is less than 1. Table III lists only the 9-, 10-, and 11-bit patterns for which R_t (sum of R_m from $m = 1$ to $m = n - 1$) is less than one.

The fact that R_t is less than one means that in approaching a sync pattern through all $n-1$ overlap conditions, the sum total of the probability-of-occurrence of the pattern in all degrees of overlap adds up to less than the probability-of-occurrence of the pattern in just one group of random data bits. As we can see, as the pattern length is increased, patterns can be found with smaller and smaller R_t , and the number of patterns for which each R_m and R_t is less than 1 also increases to great numbers.

Obviously, the sequence of conflicts for each degree of overlap of a pattern represents the quality of a pattern. As already stated, the number of conflicts required in each of the first three degrees of overlap is at least one, and the fourth through seventh degrees at least two, to make R_m less than 1 in each of the first seven degrees of overlap when the incoming signal contains 10 percent bit-errors due to noise.

Table I has been extended to over 33 degrees of overlap to allow evaluation of longer patterns. The table indicates that to maintain each R_m less than one requires the conflict-sequence to be equal to or better than the first sequence listed in table IV. More stringent requirements on a pattern may be possible, such as requiring that each R_m be less than 0.5, 0.2 or 0.1, requiring better conflict-sequences as indicated in table IV. Of course, the best that can be accomplished in one degree of overlap is 0.2.

A review of the conflict-sequences of the best patterns of length up to 11 bits indicates that no patterns exist up to 6 bit length which can equal the R_m less than the 0.5 sequence. Yet, the best 7-, 8-, 9- and 10-bit patterns better the R_m less than the 0.5 sequence. The best 11-bit pattern better the R_m less than the 0.2 sequence and the best 12- and 13-bit patterns better the R_m below the 0.1 sequence.

"Better", means that the number of conflicting bits is equal to or greater in every degree of overlap. This characteristic was used in narrowing the field of possible 6-bit start-and-end sets to find better patterns of 12, 13 and 14 bits length. In narrowing the field, not all R_m less than one, or even all R_t less than one, patterns were found. However, unlike 9-, 10-, and 11-bit patterns, which are known to have only 4, 4 and 5 patterns, respectively, with R_t 's of less than one, nine 12-bit, eighteen 13-bit, and nineteen 14-bit patterns have been found which have R_t 's less than one. Table III lists the best five patterns found for each of these length patterns.

Another interesting result of this study is represented in the quantity of "1" and "0" bits in best patterns. Up to pattern lengths of 13 bits, the best patterns have equal, or differ by one, ratios of "1" and "0" bits. It can easily be proven that this characteristic is necessary to maximize the total number of conflicts in all degrees of overlap. However, the quality of a sync pattern is dependent not only on the total number of conflicts but the distribution of these conflicts in the conflict-sequence. It has been determined that the first 14-bit pattern in table III is the best; yet, in the first three 14-bit patterns listed, the ratio of "1" and "0" bits is 6:8. This is also true of the fifth and sixth best 13-bit patterns where the ratio is 5:8.

Another characteristic of a good sync pattern is that a good pattern one or two bits longer can be generated simply by adding one or two bits to the pattern. This was suggested by Goode and Phillips. Referring to table II, note that the three 7-bit patterns listed (octal notation 13, 15, and 35) appear in a slightly different order of preference in the 8-bit patterns listed with a "0" added to the left end of the pattern. Adding a "1" bit to the right end of the three 7-bit patterns produces the 8-bit patterns with octal notation 27, 33, and 73, respectively. Both 27 and 33 occur in the best 8-bit patterns. Note also that taking the best 7-bit pattern and adding a "0" on the left puts it in fifth place in the 8-bit list, while adding a "1" on the right end results in the 8-bit pattern 27 which occurs as third best among the 8-bit patterns. The reverse is true in the conversion of the best 6-bit pattern to a 7-bit pattern. Adding a "0" on the right makes it the best 7-bit pattern while adding a "1" on the left produces a pattern for which R_m is not always less than one and does not exist in the table.

Extending this to patterns of two bits longer, it can be seen that the 9-bit pattern 73 appears in the 10- and 11-bit patterns listed and are thus the same as the 9-bit patterns with one and two "0"'s

added to the left end of the 9-bit pattern.

Longer Length Patterns

There is obviously a problem of obtaining all better long sync patterns in that the quantity which exist increases rapidly. Good 15- and 16-bit patterns have been found by adding bits to the end of previous length patterns.

The next technique used was based on the realization that so many do exist that trial and error should produce at least one of the very good ones. From the discussion of conflict sequences and quantity of patterns which fall within various restrictions on maximum R_m , it is obvious that as a pattern's length increases the restrictions can be tightened. The tighter the restrictions the more difficult will be the job of finding a pattern to meet the requirement. If the restrictions on maximum R_m are too tight, no pattern may exist. Nearly all patterns listed in table V have maximum R_m less than 0.1. All were found by trial-and-error construction.

Note in tables II, III and V that as the pattern lengths increase, patterns can be found which have lower and lower R_t , once past a 5-bit pattern. It is obvious that for very long patterns in the hundreds of bits, patterns exist starting and ending with large groups of "0"s and "1"s, respectively, for which $R_1 = 0.2$, $R_2 = 0.04$, $R_3 = 0.008$, $R_4 = 0.0016$, etc., resulting in R_t approaching 0.25 asymptotically.

It is of particular interest to note in table II that the Barker 7-bit pattern is only second best on the criterion of minimum R_t . This is also true of the Barker 11-bit pattern shown in table III. In both cases, the Barker words have the minimum sample variance discussed by Goode and Phillips. A look at the conflict-sequence for each pattern and the best patterns indicates the difference:

Degrees of Overlap	1	2	3	4	5	6	7	8	9	10
Barker 7-bit word conflict-sequence	1	1	2	2	3	3				
Best 7-bit word conflict-sequence	1	2	2	2	2	3				
Barker 11-bit word conflict-sequence	1	1	2	2	3	3	4	4	5	5
Best 11-bit word conflict-sequence	1	2	3	2	3	3	4	3	4	5

Since the number of "1" and "0" bits in both 7-bit and both 11-bit patterns are as nearly equal as possible, the total number of

conflicts are the same. In the 7-bit patterns the best has one more conflict in overlap-2 and one less in overlap-5 conditions.

Both comparisons emphasize the characteristic of Barker words, and which results from the criterion of minimum sample variance; namely, less probability-of-occurrence of false sync pattern close to the true sync position at the expense of a larger increase in the probability-of-occurrence displaced more from the true sync position.

Consideration at Other Error Rates

The figures calculated so far have always assumed that the bit error rate is random 10 percent. Both the Barker 7-bit pattern and the best 7-bit pattern, as well as a number of other patterns, meet the requirement that, in the absence of noise, there is no possibility of overlap occurrence of the pattern. For any given pattern, an expression can be written for R_t as a function of the bit error rate, e . R_t for the best 7-bit pattern (0001011) is the sum of R_1 through R_6 , and simplifies to the following equation:

$$R_t = 2e[(1+30e+80e^4) - 4e^2(9+10e+8e^3)]$$

A plot of this equation is not very exciting. It is an exponentially increasing quantity for increasing e . Writing a similar expression for the Barker 7-bit word results in a curve almost identical in shape. To emphasize the difference in the two words, the expression for the best 7-bit word was subtracted from the expression for the Barker 7-bit word. The resulting equation is the amount that R_t for the Barker word exceeds R_t for the best 7-bit word. The resulting equation is:

$$D = 4e(16e^4+1-40e^3-10e+32e^2)$$

Analyzing the equation, as e is increased it is found that the difference (D) is positive, increasing to a maximum at seven percent error rate, then decreases to zero at about 19 percent error rate, and then is negative to 50 percent error rate. It is interesting to note that the Barker word becomes better at error rates in excess of about 19 percent, but data is so poor under such conditions that there is no advantage. If an error rate of 20 percent had been chosen in calculating the quantities R_t for the 7-bit patterns, the Barker word would have come out better.

Since at this time, this line of analysis has not been pursued, the following is only conjecture. It is possible, particularly on larger patterns, that a group of two, three, or even more, patterns will have a very nearly equal R_t at one error rate, and that a crossover of advantage of one over the other will occur at a significantly smaller error rate than 19 percent, as is the case in the 7-bit patterns.

It should be quite obvious that suggested combinations of Barker

patterns - such as the 7-bit pattern and its complement - begin and end in the same bit; therefore, R_1 is always greater than one. A number of combinations have been analyzed such as Barker 7+7 complement with R_t of 3.45, Barker 7+7+7 complement with R_t of 3.49, and Barker 11+11 complement with R_t of 3.56. Since it has been shown that patterns exist which have R_t less than one, there is little reason not to use these better patterns.

Secondary Synchronization Patterns

All the preceding discussion about probability-of-occurrence of a pattern in all degrees of overlap is important only in the case where the sync detection circuits must scan through these possible patterns to find true sync. Every suggested technique for finding the first most repetitive sync pattern is some sort of threshold device. It requires scanning to find a likely pattern (some sort of search mode) then monitoring only the expected sync pattern until that pattern no longer meets some set minimum error condition (high-assurance or maintenance mode)

The author's first work in this area [1] analyzed the required amount of data which must be devoted to sync patterns to obtain various degrees of assurance of maintaining sync, and reasonable mean time to obtain sync. It is evident from this previous work that systems of PCM in existence which use word-sync patterns also have too little frame-sync information to allow minimum time to acquire sync if word-sync is ignored. Word-sync must be found to limit the number of non-sync groups of bits, which must be sampled in scanning for frame sync, to complete words. No system has yet been suggested for transmitting PCM where enough data is devoted to a "secondary" sync pattern (frame-sync if word-sync information is available, or sub frame-sync if only frame-sync is used) to allow minimum time to obtain secondary sync without prior search and establishment of primary sync.

In the process of establishing primary sync the data is grouped into words or frames. Generally, the grouping of data defined by primary sync acquisition uniquely defines the set of bits within the group which may contain the secondary sync pattern. This is true for all existing systems using frame-sync patterns of one word where word-sync is in the data. It is also true of all known proposed systems where sub frame-sync will be in same one or more prime channel locations when frame-sync patterns represent primary sync information. Therefore, establishment of prime sync allows the secondary sync detector to look for its pattern only in specified groups of bits which can never contain part of that sync pattern and part data bits. It is for this reason that the pattern chosen for secondary sync need not be a pattern with any particular autocorrelation properties. It is also the reason the same pattern of "1" and "0" bits finally chosen for sync of one or more prime channels of data containing subcommutated data can also be used as a sync pattern for a

different set of prime channels containing a different sub channel countdown, and possibly not even in known phase-relation to the first sub cycle. Use of only one pattern for insertion into more than one location in the group of data defined by prime sync will necessarily decrease complexity of both the airborne and ground equipment.

Is there any reason for a particular choice of secondary sync pattern then? If the pattern makes no difference at all why not make it the same as the primary sync pattern? Of course, this is ridiculous since it would then allow the definition of a secondary sync pattern as prime sync when attempting to acquire prime sync. This points up the fact that any secondary sync pattern, though not needing any special autocorrelation properties, should have minimum crosscorrelation characteristics with the primary sync pattern. This means that it should cross-correlate in each degree of overlap with the prime sync pattern, with an R_m less than 1.

Although the author knows of no equipment designed to search independently for prime sync and secondary sync (i.e., not require prior knowledge of prime sync in establishing secondary sync), if such equipment were planned it would be desirable to choose a secondary sync pattern with good auto-correlation properties and also good cross-correlation properties with the primary sync pattern. This imposes even more stringent properties on both the prime and secondary sync patterns and, as indicated by the author [1], would require that the same criterion be used in selecting the percentage data devoted to secondary sync as that used in selecting primary sync.

Complement As A Secondary Sync Pattern

What characteristics should a secondary sync pattern have to minimize the crosscorrelation with primary sync? First, consider, as is commonly discussed, the use of the complement of the prime sync pattern. It is true that this has maximum rejection when the patterns completely overlap, but consider the problem of scanning for primary sync through all overlap conditions of the secondary sync pattern. Since any good prime sync pattern starts and ends with opposite bits, the complement of this pattern will match up with bits identical in each overlap-1 condition. When a pattern is considered in self overlap the conflicting bits are maximized. Since $m-c$ is the number of bits which are identical, any pattern and its complement considered in cross-correlation has an "identical-sequence" and a conflict-sequence which are reversed. The sequence of conflicts which have been carefully maximized for minimum autocorrelation properties of the prime sync pattern just happens to be criterion for maximum cross-correlation of a pattern in all overlap conditions except the complete overlap, or coincidence, condition where all n bits are in conflict.

Complements are, therefore, never the answer. It is not

necessary to get n conflicts in complete overlap of a recognizer on its complement used as a sub-sync pattern at the expense of poor overlap conditions. It is only necessary to make each possible set of bits which the prime sync pattern recognizer must look at have a probability-of-occurrence of the prime sync pattern less than one.

Characteristics of Good Sub Sync Patterns

If the prime sync pattern starts and ends in "0" and "1", respectively, then the secondary sync patterns should do likewise to make each of the overlap-1 conditions ($R1$'s) equal to 0.2. Note that, in general, the cross-correlation of any two different patterns is not necessarily symmetrical.

For short patterns it is quite simple to find out if any patterns cross-correlate below random with a specified prime sync pattern and all such patterns if more than one exists. All eight of the 8-bit patterns listed in table II have been analyzed to find all possible 8-bit patterns which cross correlate below random with each of them. For the 8-bit patterns, octal notation 27, 15, 75, 65 and 53, no 8-bit patterns exist that will produce a conflict-sequence 1112223222111, which is that necessary to produce each R_m less than one. The patterns listed below at left cross-correlate with the patterns listed at right with R_m always less than one.

Cross-Correlation

Octal	Pattern	Pattern	Octal
33	00011011	01010111	127
35	00011101	00101111	57
13	00001011	01000111	107

None of the cross-correlation patterns seem to be good autocorrelation patterns, but remember that the eight 8-bit patterns listed in table II actually represent 32 possible patterns. The autocorrelation characteristics of each of the eight patterns is the same whether the pattern listed, or its complement, reflection or reflected complement is used. Since the complement or reflection of the patterns 127, 57, and 107 each begin in a "1" bit, they cannot be among the best 8-bit patterns listed. But consider their reflected complements. The reflected complement of 127 is 25; of 57, 13; and of 107, 35.

Thus we see that one pattern 25, which only cross-correlates below random with 33 in its reflected complement form of 127, is not itself among the eight best autocorrelation B-bit patterns. The fact that 35 cross-correlates below random with 57 is the same as the statement that 13 cross-correlates below random with 107. This is true since if any pair of patterns are evaluated for their crosscorrelation properties, these same properties will hold

if both patterns are complemented, reflected or reflected and complemented. We see now that if both 35 and 57 are reflected and complemented we obtain 107 and 13. It is important to note the word both. If one pattern only is reversed and/or complemented the second must be operated on in the same way to maintain the same cross-correlation properties.

Since the pattern 33 is highest in the list of good autocorrelation properties, it may be used for primary sync; pattern 127 makes the best 8-bit pattern for secondary sync if, as was discussed earlier, good autocorrelation properties are not required of the secondary sync pattern. If on the other hand, good autocorrelation properties are desired of the secondary sync pattern, then 35 should be used for prime sync since prime sync occurs most frequently, and pattern 15 is higher on the list. The pattern 13 in its reflected complemented form of 57 can be used for secondary sync. The best 11-bit pattern (227) was analyzed to find all 11-bit patterns which would cross correlate with each R_m less than one. Patterns 507, 315, 435, 515 and 1053 all meet this requirement. Of these, 315, 435, and 1053 have each R_m less than one in autocorrelation.

There is absolutely no reason why the secondary sync pattern should be the same number of bits as the prime sync pattern. When the patterns are different lengths, the minimum conflict-sequence is based on the number of bits in each degree of overlap which are common to the prime and secondary sync pattern. The best 9-bit pattern (47) was analyzed for good 9-bit cross-correlation patterns. None exist for which each R_m is below one. However, cross-correlation with 11-bit pattern 1055 meets this requirement.

It is reasonably easy to find good cross-correlation patterns. For example, the good 27-bit pattern listed in table V (11, 127, 347) was considered for a prime sync pattern. It was desirable to find a good 27-bit low cross-correlation pattern for secondary sync. From an extension of table I, the minimum conflict-sequence was chosen which would make each R_m below 0.1. With little more than a half-hour's work, a 27-bit pattern (15,645,267) was constructed in which the number of conflicts were in all degrees of overlap more than the minimum conflict-sequence demanded. This means that in 10 percent noisy input signal conditions the probability-of-occurrence of the pattern 11, 127, 347 in any group of 27 consecutive bits containing two or more bits of the 27-bit pattern 15, 645, 267 is below one-tenth the probability-of-occurrence of the pattern in 27 random data bits.

Evaluation of the pattern 15, 645, 267 for its autocorrelation properties indicates that each R_m is less than one except R_{11} where only two conflicts exist in self-overlap. It is a less than desirable prime sync pattern but an excellent pattern for secondary sync when the pattern 11, 127, 347 is used for primary sync.

is quite interesting to note that in the case of finding a cross-correlation pattern for 11, 127, 347 the conflict-sequence requiring each R_m to be below 0.1 was chosen and a pattern easily obtained. Curiously, it is found that only three of the eight best 8-bit patterns have any patterns which cross-correlate below R_m of 1.0, and the three 7-bit patterns have none.

It has been shown that as the length of patterns increase the number of patterns which exhibit sufficient autocorrelation characteristics increase rapidly. It is also no doubt true that not only do more of these good auto-correlation patterns have good cross-correlation patterns but that each good autocorrelation pattern has many good cross-correlation patterns. Only part of all good cross-correlation patterns are themselves good autocorrelation patterns if this characteristic is desirable.

Conclusions

The author has shown previously that the length of sync patterns and the spacing between patterns should be chosen on the bases of reliability of recognizing a true sync pattern when it arrives, and minimizing the probability of finding a sync pattern among groups of non-sync bits. In reaching these conclusions, it was assumed that a sync pattern of bits could be found which reduced the probability of false occurrence of a sync Pattern in each degree of overlap below the probability-of-occurrence of a sync pattern in each random group of data bits.

In this present study, it has been shown that the probability-of-occurrence of a sync pattern in each degree of overlap could not only be reduced below the random occurrence of a sync pattern in a group of data bits, but even at a 10 percent random bit error rate, the sum of the probabilities-of-occurrence of a sync pattern in all overlap conditions can be reduced below the probability-of-occurrence of the pattern in one set of random bits.

The general characteristics of sync patterns have been discussed. It has been found that for every length pattern a best pattern of "1" and "0" bits exists which minimizes the probability-of-occurrence of a false sync indication in incoming, and possibly noisy, data. There is nothing unique about patterns of length equal to one less than a multiple of four ($4K-1$) bits length. This is true since, given any good sync pattern, a pattern of one additional bit length has been found which reduces the probability-of-occurrence of a false sync pattern in random data by 2 to 1 and always reduces the probability-of-occurrence of the pattern in all groups of bits containing part sync and part data bits below that of the shorter pattern.

A final but extremely important conclusion to this study is that in choosing sync patterns for secondary sync, all secondary sync patterns located in different locations in groups of data defined by primary sync patterns may be made identical, requiring only one secondary sync pattern for all applications. This reduces

complexity in the air and on the ground. Secondary sync patterns should always be chosen to minimize cross-correlation with prime sync patterns. The autocorrelation properties of secondary sync patterns are of no concern.

References

1. Williard, M. W., "PCM Telemetry Synchronization", Proceedings of the National Telemetering Conference, Chicago, Illinois, May 1961.
2. Barker, R. H., "Group Synchronization of Binary Digital Systems", COMMUNICATIONS THEORY, by Willis and Jackson, Butterworth's Scientific Publications, P. 273.
3. Good, G. E. & Phillips, J. L., "Optimum PCM Frame Synchronization Codes and Correlation Detection", Proceedings of the National Telemetering Conference, Chicago, Illinois, May 1961.
4. Hill, F. R. & Weblemoe, J. L., "Synchronization Methods for PCM Telemetry", Proceedings of the National Telemetering Conference, Chicago, Illinois, May 1961.
5. Gilbert, E. N., "Synchronization of Binary Messages", IRE Transactions on Information Theory, Vol. IT-6, pp. 470-477, September 1960.
6. Williard, M. W., "Time to Establish Pattern Synchronization in PCM Telemetry", unpublished.

Table I

Relative Probabilities-of-Occurrence

OVERLAP	CONFLICTS	R_m
0	0	1.0
1	0	1.8
	1	0.2
2	0	3.24
	1	0.36
	2	0.04
3	0	5.832
	1	0.648
	2	0.072
	3	0.008
4	0	10.4976

	1	1.1664
	2	0.1296
	3	0.0144
	4	0.0016
5	0	18.8955
	1	2.0995
	2	0.2333
	3	0.02592
	4	0.00288
	5	0.00032
6	0	34.0122
	1	3.7791
	2	0.4199
	3	0.04665
	4	0.005184
	5	0.000576
	6	0.000064
7	0	61.2220
	1	6.8024
	2	0.7558
	3	0.08398
	4	0.009331
	5	0.001037
	6	0.0001152
	7	0.0000128

Table II

Best Synchronization Patterns

PATTERN LENGTH	PATTERN	OCTAL NOTATIONS	Rt
1	0	0	0.0
2	01	1	0.21
3	001*	1	0.56
4	0 011	3	0.888
	0 001	1	1.208
5	00 101	5	1.222
6	001 011	13	1.043
	001 101	15	1.248
7	0 001 011	13	0.722
	0 001 101*	15	0.832
	0 011 101	35	1.295

8	00 011 011	33	0.764
	00 011 101	35	0.895
	00 010 111	27	0.907
	00 001 101	15	1.010
	00 001 011	13	1.064
	00 111 101	75	1.379
	00 110 101	65	1.411
	00 101 011	53	1.464

*Barker

TABLE III

Best Synchronization Patterns

PATTERN LENGTH	OCTAL NOTATION	Rt
9	47	.82
	73	.84
	35	.91
	33	.93
10	73	.70
	67	.71
	47	.91
	173	.98
11	227	.65
	355 *	.87
	173	.93
	167	.94
	73	.96
12	153	.58
	273	.61
	533	.66
	573	.67
	267	.72
13	327	.54
	353	.61
	517	.68
	573	.69
	153	.69
14	547	.55
	327	.56
	1057	.59
	733	.60
	353	.61

* Barker

TABLE IV

Conflict Sequences

Degree of Overlap $m =$	1 2 3 4 5 6 7 8 9
Rm less than 1.0 $C \geq$	1 1 1 2 2 2 2 3 3
Rm less than 0.5 $C \geq$	1 1 2 2 2 2 3 3 3
Rm less than 0.2 $C \geq$	1 2 2 2 3 3 3 3 4
Rm less than 0.1 $C \geq$	1 2 2 3 3 3 3 4 4

TABLE V

Better Synchronization Patterns

PATTERN LENGTH	OCTAL NOTATION	Rt
15	1,347	0.449
16	2,717	0.487
17	12,667	0.511
18	26,567	0.405
21	155,367	0.424
22	332,757	0.523
23	447,347	0.381
27	11,127,347	0.368
29	44,567,347	0.360
31	222,253,347	0.361
33	1,454,265,557	0.331

TABLE VI

PATTERN LENGTH	OCTAL NOTATION	Rt
15	1347	.449
15	2467	.451
15	547	.454
15	3673	.515
15	2317	.539
16	5567	.410
16	4657	.418
16	4727	.420
16	4567	.427
16	5317	.454
17	5317	.389
17	11657	.402
17	3327	.410
17	5567	.412
17	6257	.418
18	5317	.348
18	12157	.359

18	12637	.379
18	11727	.384
18	12717	.389
19	24637	.323
19	14657	.323
19	26357	.338
19	23657	.348
19	12637	.350
20	43667	.330286
20	31537	.331193
20	24637	.331480
20	14657	.333547
20	44357	.338729
21	64567	.325172
21	53317	.326088
21	122637	.327425
21	66567	.328912
21	55637	.329302
22	53317	.294119
22	233657	.307726
22	122637	.311946
22	131367	.312068
22	215137	.314068
23	131657	.289854
23	152717	.295305
23	324737	.298981
23	461537	.301640
23	146657	.302171
24	1147537	.283710
24	611357	.284568
24	1067137	.285237
24	647357	.285943
24	331657	.286413
25	1421337	.279799
25	1147537	.279990
25	663657	.282320
25	2136357	.282582
25	1153477	.282627
26	2153137	.272114
26	3312757	.274766
26	1073237	.275803
26	2547477	.276029
26	2303277	.276093
27	4507337	.266231
27	3145537	.267268
27	2134557	.267846
27	4326277	.270273
27	3312757	.270768
28	11627277	.263117
28	5223637	.263452
28	10656477	.263958
28	5323637	.264533
28	3523557	.264858
29	21426277	.260960

29	12474677	.261478
29	5463657	.261845
29	31051737	.261923
29	6323657	.262155
30	13147537	.257032
30	25163337	.258728
30	32131677	.259325
30	20671357	.259418
30	41535477	.259862
31	104362677	.256297
31	53317177	.256701
31	113016737	.256956
31	52346677	.257503
31	43236277	.257758
32	46247537	.255375
32	53233477	.255497
32	124447177	.255648
32	210753337	.255724
32	120663277	.255832
33	223475277	.254357
33	303352677	.254558
33	111636537	.254678
33	126247637	.254848
33	225171577	.254962
34	223475277	.253355
34	523117177	.253579
34	152633637	.253855
34	311257177	.254026
34	521436677	.254114
35	325467477	.252330
35	323467277	.252510
35	544717277	.252660
35	1243066677	.252820
35	1105647177	.252960
36	2223432577	.252197
36	653157177	.252207
36	706515577	.252241
36	544717277	.252287
36	2223472577	.252299
37	646533477	.251734
37	1433132577	.251782
37	1247133177	.251969
37	2514457177	.251971
37	1226334377	.252001
38	1515267177	.251137
38	5231136377	.251347
38	1247133177	.251415
38	2514457177	.251437
38	4342332677	.251517
39	6151165377	.251046
39	20534751577	.251079
39	20710565577	.251115
39	10256364677	.251131
39	12433307577	.251140

40	13114357277	.250842
40	23072336577	.250850
40	24342355577	.250858
40	12470666377	.250883
40	11435157277	.250885
41	25451476377	.250707
41	16234556577	.250709
41	44650363577	.250714
41	13114357277	.250747
41	25106236377	.250783

PUBLICATIONS: of Merwin W. Williard

An Evaluation of Optional Timing/Synchronization Features to Support Selection of an Optimum Design for the DCS Digital Communication Network, Tenth Annual Precise Time and Time Interval meeting, co-authors Bradley, Cain, Naval Research Laboratory, November, 1978

Introduction to Redundancy Coding, IEEE Transactions on Vehicular Technology, August 1978, A Tutorial

TDM Over Poor Quality Circuits, Final Report to RADC, Rome, N.Y., Contract F30602-73-C-0358, September, 1975

Dynamic Behavior of a System of Mutually Synchronized Oscillators, IEEE Transactions on Communications Technology, November 1971

Maintaining Bit Integrity in Time Division Transmissions, N.A.E.C.O.M. Conference, co-author L.J. Horkan, Dayton, Ohio, June 1971

Analysis of a System of Mutually Synchronized Oscillators, IEEE Transactions on Communications Technology, October 1970

Timing and Framing Techniques for Troposcatter and Line of Sight, co-author R.E. Gilbreath, Final Report ECOM, Ft. Monmouth, N.J., 1968

Network Bit Synchronization Atomic Clock Versus Frequency Averaging, co-author W.O. Woolsey, Final Report ECOM, Ft. Monmouth, N.J., 1968

Automatic Real Time Data Quality Analysis, International Telemetry Conference, Washington, D.C., October 1967

Refinements on the Analysis of PCM Telemetry Synchronization, International Telemetry Conference, Los Angeles, CA, October 1966

Mean Time to Acquire PCM Synchronization, IRE Transactions, National Symposium on Space Electronics and Telemetry,

Miami Beach, Sept. 1962

Optimum Code Patterns for PCM Synchronization, Proceedings of the National Telemetry Conference, Washington, D.C., May 1962

Pulse Code Modulated Telemetry, Chapter VI, AEROSPACE TELEMETRY, edited by Harry Stiltz, Prentice-Hall Space Technology Series, 1961

Pulse Code Modulated Telemetry Synchronization, Proceedings of the National Telemetry Conference, Chicago, IL, May 1961

A Universal PCM Data Collection System, Proceedings of the National Telemetry Conference, Santa Monica, CA, May 1960

The Use of a Fractional Bi-Stable Multivibrator Counter in the Design of an Automatic Discriminator Calibrator, IRE Transactions of the Professional Group on Space Electronics and Telemetry, Sept. 1959

