REFINEMENTS ON ANALYSIS OF PCM SYNCHRONIZATION

MERWIN W. WILLIARD
Symetrics Engineering
Satellite Beach, Florida

ABSTRACT

In this paper, the author reconsiders some of his previously reported assumptions on an analysis of mean time to establish PCM synchronization, and he provides further insight into the effect of specific synchronization patterns and parameters of the synchronization process. An improvements in one assumption shows how the analysis can provide slightly more accurate results. Specific recommendations are made for standardization of PCM sync patterns.

INTRODUCTION

This material is an extension of work reported by the author in a previous paper entitled “Mean Time to Establish PCM Synchronization, “ (Reference 1) and presented in the Record of the 1962 National Symposium on Space Electronics & Telemetry, Miami Beach, Florida, October 1962.

This paper will discuss some of the assumptions made in the previous analysis of pattern synchronization in terms of the mean time to establish synchronization and the percent data lost due to loss in synchronization, and show that the analysis as previously reported is worst case. In addition some improvements in the analysis are justified which make the analysis more accurate and/or point up the significance of specific choice of sync patterns and their effect on sync performance.

Review of Previous Work In References 3, 2, and 1, in that order, the author has analyzed various aspects of PCM pattern synchronization. Specifically Reference 1 presents the mathematical tools to evaluate PCM synchronization in terms of the mean time to establish sync and the percentage data lost due to a loss in sync. It was shown that the analysis can be applied to more complex sync system instrumentation, however, it was shown that rapid and highly reliable pattern sync is possible with a relatively simple instrumentation technique.

The simple sync technique involves reviewing the incoming serial PCM bit stream as it passes through a shift register, looking at each set on N consecutive bits with a majority gate until a set of N bits containing e or less errors is found. This process is called the scan mode and when the scanner finds an acceptable set of N bits, it is assumed to be true sync. Counters are reset and the maintenance mode is initiated which ignores output of the scanner except to see that the pattern continues to reappear each proper interval in the data. In the maintenance mode the number of errors tolerated will in general be changed to a greater number (e'). The first time the maintenance mode does not find an acceptable pattern the system switches back to scan mode with e errors tolerated.

Equations are developed which provide the mean time to a scan decision, probability of a scan decision being correct or in error, but most important equations are developed which provide the mean time to true sync and the percentage data lost due to loss in synchronization in the presence of noise.

The equations are then used to evaluate and plot the expected performance of such instrumentation on various planned or in use PCM links.

The following is a list of symbols, their definitions, and the equations developed in Reference 1.

Pc	=	Probability of detecting a true sync pattern in the presence of noise when in the scan mode tolerating e errors.
Pc'	=	same but for the maintenance mode tolerating el errors.
Pi	=	Probability of occurance of an apparent sync pattern among random data when in the scan mode tolerating e errors.
Pi'	=	same but for the maintenance mode tolerating e' errors.
b	=	number of bits between consecutive sync patterns minus 1, referred to as a group or frame.
Pt	=	Probability of a false sync while scanning through one group of data.
T	=	Probability that a decision of the scan mode is true.
W	=	Probability that a decision of the scan mode is false.
M	=	mean number of groups or frames to a scan decision.
J	=	mean number of groups before maintenance mode can reject a false decision of the scanner.
K	=	mean number of true patterns which can pass before the maintenance mode rejects true sync because of noise errors.
L	=	mean time to reacquire true pattern synchronization.

$$Pt = 1-(1-Pi)^b$$

$$T = \frac{Pc(1-Pt)}{Pt + Pc(1-Pt)}$$

$$W = \frac{Pt}{Pt + Pc(1-Pt).}$$

$$M = \frac{1}{Pt + Pc(1-Pt)}$$

$$J = \frac{1}{1-Pi'}$$

$$K = \frac{1}{1-Pc'}$$

$$L = \frac{M}{T} + \frac{W}{T} J$$

$$\% \text{ data lost} = 100 \times \frac{L}{L+K}$$

Some of the assumptions made in the analysis reported in Reference I were discussed, but. their significance in some instances may need more clarification and/or evaluation. In addition, some suggestions will be made for further improving the analysis to make the evaluation more accurate. It should also be noted that where improvements are made in the analysis, the analysis as previously presented assumed worse conditions and therefore the original analysis was more on the safe side.

REVIEW and ANALYSIS of SOME PREVIOUS ASSUMPTIONS

Bit Synchronization This was mentioned in Reference 1, but the reader should be reminded that the analysis presented deals with pattern synchronization, which before it can operate must have a clean regenerated signal and clock. This means that a decision has been made on each received bit of information, and if noise is present that decision may well be in error. The performance of pattern sync is analyzed in terms of the bit error probability which is the average rate of bits which are detected in error, and it is assumed that the error rate is statistically random.

Random Data The assumption of data bits being random means that any time a data bit is received it has equal probability of being a “1” or a “0”. In other words, whether that bit comes up a “1” or a “0” bears absolutely no relation to the polarity of any other bit no matter if the other bit referred to is adjacent, one sample away, one frame away etc. In general, if data channels are not relatively static and/or sync samples are taken which overlap multiple channels, the assumption will be very good.

However, some of the dangers of this assumption become apparent. If a number of consecutive samples contain information predominantly below half scale, the most significant bits will be predominantly “0”. Even worse, if one channel or a few consecutive channels are reading static data, their transmitted code will remain fixed from one frame to the next.

If word synchronization is being analyzed, the problem of predominantly below half scale channels (or above half scale) becomes significant. If the sync cycle is from frame to frame, then the static channels are a problem. It would be a near impossible task to take these dependent probabilities into account even if the characteristic variations of each channel could be predicted in some specific operational test, and obviously impossible to include them in a general analysis.

Instead of attempting to take dependent probabilities into account, the designer of the transmission system can use some means to make their effect negligible. For example, primary frame synchronization patterns will usually be long enough to overlap two or three data samples. By intermixing expected static data channels with expected rapidly varying channels, at least a significant part of any erroneously detected pattern may be expected to change within a few frames or less. A much more certain way to handle this is to use alternating patterns such as a pattern and its complement or better yet, two patterns each with good autocorrelation properties and mutually with good cross-correlation properties. Reference 2 is a technical paper dealing solely with the problem of choice of synchronization pattern.

Using alternating patterns, if false sync occurs and the sync system requires that the next pattern must change to its complement or to some new pattern with a large number of bits different, the probability of these data channels changing to the same new code requirements is about as random as you can get. Actually if the data is near static, the probability of it changing to the second sync pattern is smaller than the Pi calculated as a statistically independent probability and, in fact, the assumption of random data becomes a safe-side assumption.

Synchronization Patterns The assumption made in the analysis is that <u>each and every</u> group of N data bits sampled by the pattern recognizer while scanning for the true sync pattern, other than the one true complete set of sync bits, have a probability of exactly Pi of providing a false sync indication. This is decidedly not true of those groups of N bits which contain part of the sync pattern and part data bits. Consider as an example a 30-bit sync pattern spaced at 300 bit intervals. While approaching the true pattern in a scan mode, each set of 30 bits is sampled, one new bit enters the shift register as all bits advance and another sample is then taken. As the true pattern is approached there are 29, or in general, (N- 1) groups of bits which contain part of the sync pattern and part data bits (hereafter called overlap groups).

Each time the scanner samples all patterns from one true pattern to the next it passes over (N-1) “overlap” groups while leaving the vicinity of one pattern and (N- 1) more overlap groups while approaching the next pattern. Therefore, if 30-bit patterns are spaced at 300 bit intervals, there are 299 non- sync patterns to be sampled, of which 2x(30-1) = 58 are not made up solely of statistically random data bits and 241 are “random” data.

Specific probabilities of false pattern occurrence in each overlap group can be calculated for each specific sync pattern and these probabilites can range from near 1 to well below Pi depending on the chosen pattern.

Reference 2 is one technical paper dealing solely with this problem. In fact, more has been written about patterns than about any other aspect of pattern synchronization. References 2, 6, 7, 8, and 9 and yet other papers referenced therein represent many papers which are wholly or predominantly devoted to the study of synchronization patterns. Nearly all recognize the probelm of overlap false detection of sync and suggest techniques for determining good sync codes. Our major concern here is how does the selection of a specific sync pattern affect the assumption that each overlap group has a probability of false occurrence equal to Pi. Referring to Reference 2, it has been shown that for all pattern lengths, a large number of patterns exist which have the property that, at a 10% bit error rate on the incoming signal, the probability-of-occurrence of the pattern in each overlap group is less than Pi (where Pi is the probability-of-occurrence of the pattern in one set of statistically random data). Actually, patterns have been found for which the sum of the probabilities-of-occurrence in all (N-1) overlap groups do not add up to one times Pi. If such good patterns are used, then the equation for Pt in Reference 1 will give a higher than actual probability of false synchronization. Thus, the assumption of all groups including the overlap groups having a probability of false sync indication equal to Pi will be on the safe or worst case side.

It is possible to evaluate just how safe this assumption actually is. Refer to equation 1, on page 4. This says that the probability of not getting a false sync (1-Pt) while scanning through b wrong patterns each of which has a probability of false sync equal to Pi is the quantity $(1\text{-Pi})^b$.

This is a simplification of the more general equation $(1\text{-}Pt)=(1\text{-}P1)^a (1\text{-}P2)^b (1\text{-} P3)^c$ --- $(1\text{-}Pn)^m$ which says that if in (a) positions the probability of false sync is P1, and in (b) positions the probability of false sync is P2, and in (c) positions the probability is P3, etc. , the probability of getting past all without a false sync indication is the above product. Now consider Table 1, of Reference 1. Take the value of Pi for e=5. Pi = .001625. Now let's assume the existence of some hypothetical 30-bit sync pattern such that the probability-of-occurrence of the pattern in all overlap groups vanishes to nothing. In other words, the probability of a false sync while scanning through any groups of data which contains part sync bits and part data bits is 0. Note the discussion in Reference Z which indicates that in the complete absence of noise, some patterns can have the probability-of-occurrence of false sync in all overlap groups equal to 0. Other patterns may not have this property.

Now the number of wrong patterns which must be scanned which contain only "random" data bits is 299-2(30-l)=24l.

$$(1-Pt)=(1-.0001625)^{241} = .9619$$

This is only 1% greater than (1 -Pt) if b = 299 and says that such a condition will only reduce the mean time the system spends in the scan mode by 1%.

$$\frac{M}{T} = \frac{1}{Pc(1-Pt)}$$

Now consider another hypothetical 30-bit pattern such that in each and every one of the 29 degrees of overlap the probability-of-occurrence of false sync is double that of the probability-of-occurrence of the pattern in random data.

$$(1-Pt)=(1-.0001625)^{241}(1-2x.0001625)^{58}=.94448$$

This is only a 0.8 percent decrease in the probability of getting completely through a group of non-sync patterns and will only increase the mean time spent in the scan mode of operation by 0.8 percent. Note again that this second calculation is for a pattern such that the probability of false sync in every overlap group is double that of random data. Nearly every pattern suggested by anyone who is familiar with the problem will be better than this.

Look at it another way. To get a 10% increase in the mean time to acquire sync, (1-Pt) would need to go down to .85732. Calculating backword,

$$.85732 = \left[1-n(.0001625)\right]^{58} \left[1-.0001625\right]^{241}$$

and n equals 12. Thus, each and every overlap group would need a probability of false sync 12 times greater than the probability of false sync in one group of random data to cause a mere 10% increase in the time to acquire sync.

This analysis then points up the relative insignificance of a specific pattern in its effect upon the mean time to establish pattern synchronization. This discussion should not be interpreted to mean that any pattern is acceptable. As is discussed in Reference 2, most patterns of any specific length will not be good sync patterns since they will have individual probabilities of occurrence of false sync in one or more overlap positions which will be well in excess of 12 times Pi.

Although the effect on mean time to acquire sync may be only a few percent on the average, when the system does lock onto this false sync in an overlap group, it will be more difficult to reject than a set of all data bits. This is due to the fact that those bits in the overlap group which are sync bits never change as transmitted. Thus, changes in the data bits must be relied upon to cause rejection of this false sync, and in an overlap group the number of data bits is smaller than N. The analysis of time to acquire assumes that all N bits of a falsely detected sync pattern have a random probability of reoccurring within the requirements of the sync pattern.

The analysis does show that a small difference between two good patterns will have negligible effect on the mean time to establish sync. Thus any relatively good pseudo random or minimum probability pattern will cause actual values of (1-Pt) to be greater than the analysis assumes, causing a resulting calculated average time in the scan mode (M/T) to be at most, a fraction of a percent on the high or safe side.

What are Good Synchronization Patterns Reference 2, 6, 7, 8 and 9 all describe good sync patterns. The author presented the best patterns selected by a minimum probability of false sync criteria. Others use different criteria but which also produce good patterns that differ little in actual performance and as shown above would have negligible effect on the mean time to establish synchronization.

It is interesting to observe that the “Best Synchronization Patterns” found by Maury and Styles and reported in Reference 6 are among the 3 best patterns reported by the author in Reference 2 for pattern lengths of 7 to 14 bits length. The criteria for selection was basically the same, however the author did not have the benefit of a computer to extend the work to much longer patterns. The properties of good patterns of all code lengths discussed by the author are characteristics of the Maury and Styles patterns.

Maury and Styles only extended the analysis to patterns of 30 bits length. Their patterns are listed as the patterns up to length 30 in Table I. The actual patterns are listed in the Octal notation defined by the author in Reference 2. Thus, the pattern defined as the 11 bit pattern 355 is the pattern.

00 01l 101 101

The octal characters are converted to binary groups of three bits and then zeros are added on the left until the specified pattern length is obtained. Maury and Styles report these patterns in the reverse order (end for end) but as pointed out by the author in Reference 2, a pattern, its complement, or the pattern reversed end for end or the complement of the reversed pattern all have identical sync pattern properties.

Table I also lists the summed relative probabilities of occurrence of the pattern in all degrees of overlap Rt defined in Reference 2. As predicted, better or best patterns have values of Rt which asymptomatically approach .25 as the length of pattern is increased.

It is easily shown that if Rt is less than one then the actual probability of occurrence of false sync in all degrees of overlap does not accumulate to be equal to the probability of occurrence of the pattern in just I sample of random data Pi. Specifically, the 7 bit pattern 15 has an accumulate probability of occurrence of a false pattern of .0060 while scanning through all six positions of overlap at a 10% bit error rate. The probability of occurrence of the pattern in just 1 group of random data bits is .0078. If such good patterns are used for sync patterns then in the analysis for mean time to establish sync, the number of patterns to be scanned in searching for sync (the quantity b in the analysis) can be reduced to b-2(N- 1) where N is the pattern length. This allows an assumption that the entire set of all overlap positions on one side of a true sync pattern has no greater probability of false sync than one sample of random data. As pointed out above however, this should change the results only about 1% or less in typical systems.

The author in Reference 2 reported that adding a few bits to either end of a good sync pattern, provided the polarity of the added bit is the same as the polarity of the last bit before adding, will produce a very good pattern one or two bits longer. Note in Table I that a 31, 32, and 33 bit pattern is presented which have the same octal notation as Maury and Styles' 30 bit pattern. These were obtained simply by adding 0's to the left end of the pattern (the end which already has a series of consecutive 0 bits). The patterns of length 31, 32 and 33 bits thus generated have Rt's which are lower than any previously found by the author. It is obvious that still better exist since the trend in Rt as pattern length increases is up and in general it should approach .25000 as a limit. However as evaluated above a little better pattern will have negligible effect on the mean time to establish synchronization.

The second 33 bit pattern listed in Table I was formed by adding a 1 bit to the right end of the listed 32 bit pattern. It is just slightly better than the first 33 bit pattern listed.

Mean Time to a Wrong Decision By Scanner In the scan mode analysis of Reference 1, the probability of detection of a false sync pattern is translated from the probability of detection on each look (Pi) to the probability of false detection in each group (Pt). In so doing, something is lost. The subsequent analysis referes to probabilities in terms of groups, and assumes a false decision of the scanner still requires at least 1 full group or frame. However, if Pi is very near 1, the scan mode will accept almost any random pattern as possibly being true sync.

Assume true sync has just been slipped. Only a few patterns (bits in this instance) go by in the scan mode before the scanner accepts a pattern as possibly being sync. Switching to the Maintenance mode shuts out all data for at least one frame length. When the maintenance mode again decides the scan decision was not right, it switches back to scan which, if Pi is very high, goes only a few more bits before picking out another possible sync position. Thus, the system in each scan operation only inches through a frame switching to Maintenance mode and leaps over frames and true sync patterns then back to inching in the scan mode. Thus where Pi is large, the system will switch many times back and forth between scan and maintenance modes with each scan lasting much less than one frame or group of data defined by consecutive sync patterns. Thus, when Pi is larger the average time to a false decision will be shorter than one group because the probability of a false decision is large. For example in Table I, Reference 1 for e = 9, note that T = .0016. Therefore, W = .9984. This says that there is a 99.84% chance that a wrong decision will be made in each scan of one group of data. It should be obvious from this that the average time to a scan decision will be much shorter than one group.

The mean time to a dicision expression is not realistic as presented in Reference 1. Therefore, reconsider a new expression for M based on the bit-by-bit samples made by the scanning circuit.

Figure 1 is a representation of the situation. Assuming true sync has just been slipped, the scanner begins just one bit after a true pattern is past. Pi is the probability that the first sample comes up wrong, and (1-Pi) that the scanner makes no decision on the first sample and thus gets to look again at a second sample. Pi(1-Pi) is thus the probability of false detection on the second sample, etc. After passing one group of data $(1-Pi)^b=1-Pt$, as defined in Reference 1, is the probability of reaching the next true sync without a false decision, and Pc(1-Pt) is the probability of a true decision on the next true pattern, etc.

The total probability of a true decision is the same sum of terms as calculated in Reference 1 and T is unchanged.

Now, let’s see if the total probability of a false decision comes out right. It must be 1-T if Pi and Pc are each between 0 and 1. Using the probability expressions in Figure I, where $(1-Pi)^b=(1-Pt)$

$$W = Pi+Pi(1-Pi)+Pi(1-Pi)^2+\text{-------}+Pi(1-Pi)^{b-1}$$
$$+(1-Pc)(1-Pt)\left[Pi+Pi(1-Pi)+\text{-----}+Pi(1-Pi)^{b-1}\right]$$
$$+(1-Pc)^2(1-Pt)^2\left[Pi+Pi(1-Pi)+\text{-----------------}\right]$$
$$+\text{-------------}$$

Note the common geometric progression of b terms in each line above. The common progression reduces to $1-(1-Pi)^b = Pt$ as defined in Reference 1.

$$W = Pt(1+r+r^2+r^3+\text{----------}+r^\infty)$$

Where

$r=(1-Pc)(1-Pt)$, thus:

$$W = \frac{Pt}{1-(1-Pc)(1-Pt)} = \frac{Pt}{Pt+Pc(1-Pt)}$$

This checks out as I-T and is unchanged from Reference 1.

The major change in the following analysis is that the average time to a decision by the scanner will be calculated in terms of average number of samples rather than assuming one full group must go by before a decision is made. The average number of <u>bits</u>, or scan samples, to a decision by the scanner will be the sum of the products of the probability of a decision, right or wrong, on any bit position times the number of bits to arrive at that position, divided by I (the probability that a decision will eventually be made). Since it is average number of bit rather than groups lets call it Mb such that Mb divided by (b+1) will be equivalent to the average groups to a decision.

$$Mb = Pi+2Pi(1-Pi)+3Pi(1-Pi)^2+\text{-------}bPi(1-Pi)^{b-1}$$
$$+(b+1)Pc(1-Pt)$$
$$+(1-Pc)(1-Pt)\left[(b+2)Pi+(b+3)Pi(1-Pi)+\text{----}+(2b+1)Pi(1-Pi)^{b-1}\right]$$
$$+2(b+1)Pc(1-Pc)(1-Pt)^2$$
$$+(1-Pc)^2(1-Pt)^2\left[(2b+3)Pi+\text{---}+(3b+2)Pi(1-Pi)^{b-1}\right]$$
$$+\text{----------------}$$

Scan Begins

Scan Mode Samples	1	2	3	---	b	b+1 (Sync Pattern)	b+2
Probability of False Decision	Pi	Pi(1-Pi)	$Pi(1-Pi)^2$	---	$Pi(1-Pi)^{(b-1)}$	0	$(1-Pc)\ (1-Pi)^b$ Pi
Probability of True Decision	0	0	0	---	0	$Pc(1-Pi)^b$	0

Scan Mode Samples	b+3	---	2b+1	2b+2 (Sync Pattern)	2b+3	2b+4
Probability of False Decision	$(1-Pc)\ (1-Pi)^b$ Pi(1-Pi)	---	$(1-Pc)(1-Pi)^b$ $Pi(1-Pi)^{(b-1)}$	0	$(1-Pc)^2(1-Pi)^{2b}$ Pi	$(1-Pc)^2(1-Pi)^{2b}$ Pi(1-Pi)
Probability of True Decision	0	---	0	$(1-Pc)\ (1-Pi)^b$ $Pc(1-Pi)^b$	0	0

Figure 1 Decision Probability Chart

Now consider the infinite set of lines containing Pc. The complete set of even lines above form the sum of terms:

$$Pc(1-Pt)\ (b+1)\quad \left[1+2(1-Pc)\ (1-Pt)\ +3(1-Pc)^2(1-Pt)^2+----\infty\right]$$

Using the rules of reduction of infinite geometric progressions, this reduces to:

$$H = \frac{Pc(1-Pt)\ (b+1)}{\left[\ 1-(1-Pc)\ (1-Pt)\right]^2} = \frac{Pc(1-Pt)\ (b+1)}{\left[Pt+Pc(1-Pt)\right]^2}$$

Now, breaking up the third and fifth lines above, as will be representative of the infinite number of additional odd number lines, into two lines each:

$$Mb = H$$

$$+Pi+2Pi(1-Pi)+3Pi(1-Pi)^2+-------------+bPi(1-Pi)^{b-1}$$

$$+(1-Pc)\ (1-Pt)\quad \left[(b+1)Pi+(b+1)Pi(1-Pi)+--+(b+1)Pi(1-Pi)^{b-1}\right]$$

$$+(1-Pc)\ (1-Pt)\quad \left[Pi+2Pi(1-Pi)+----+bPi(1-Pi)^{b-1}\right] = G$$

$$+(1+Pc)^2(1-Pt)^2\quad \left[\ (2b+2)Pi+---+(2b+2)Pi(1-Pi)^{(b-1)}\ \right]$$

$$+(1+Pc)^2(1-Pt)^2\quad \left[Pi+2Pi(1-Pi)+---+bPi(1-Pi)^{\ (b-1)}\right] = G$$

$$+-------------------$$

The even numbered lines in this expression for Mb all contain a common geometric progression containing just b terms. This can be reduced to

$$\frac{1-(1-Pi)^b}{1-(1-Pi)} \quad -b(1-Pi)^b$$

Using the symbols of Reference 1:

$$G = \frac{Pt}{Pi} \quad -b(1-Pt) \quad = \frac{Pt-bPi(1-Pt)}{Pi}$$

Now after factoring out the b+1, 2b+2, etc. , the odd numbered lines above starting with 3 have a common geometric progression of b terms which reduces to

$$1-(1-Pi)^b=Pt$$

Now, considering that there are an infinite number of lines in the above expression for Mb, it looks like this:

$$Mb = H+G\left[1+(1-Pc)(1-Pt)+(1-Pc)^2(1-Pt)^2+\text{-------------}\infty\right]$$
$$+Pt(b+1)(1-Pc)(1-Pt)\left[1+2(1-Pc)(1-Pt)+3(1-Pc)^2(1-Pt)^2--\infty\right]$$

The remaining two infinite geometric progressions reduce to:

$$\frac{1}{1-(1-Pc)(1-Pt)} \quad \text{and} \quad \frac{1}{\left[1-(1-Pc)(1-Pt)\right]^2} \quad \text{respectively.}$$

Thus

$$Mb=\frac{Pc(1-Pt)(b+1)}{[Pt+Pc(1-Pt)]^2}+\frac{Pt-bPi(1-Pt)}{Pi\,[Pt+Pc(1-Pt)]}+\frac{Pt(b+1)(1-Pc)(1-Pt)}{[Pt+Pc(1-Pt)]^2}$$

Since this was calculated in bits not groups, let’s divide by (b+1) to put it inequivalent average groups. Also, factor out the common term in the denominator. We can again refer to the average number of groups to a decision as M groups after dividing by (b+1).

$$M = \frac{1}{Pt+Pc(1-Pt)}\left[\frac{Pc(1-Pt)}{Pt+Pc(1-Pt)}+\frac{Pt-bPi(1-Pt)}{Pi(b+1)}+\frac{Pt(1-Pc)(1-Pt)}{Pt+Pc(1-Pt)}\right]$$

The first term in the brackets is equal to T. The last term reduces to W-Pt. Since W+T=1, it causes some simplification. Further combining and canceling produces:

$$M = \frac{1}{Pt+Pc(1-Pt)}\left[\frac{1-(1-Pt)(1-Pi)}{Pi(b+1)}\right]= \frac{1}{Pt+Pc(1-Pt)}\left[\frac{1-(1-Pi)^{b+1}}{Pi(b+1)}\right]$$

Here we see that the value for mean time to a scan decision M as derived in Reference 1 exists outside the brackets with a new modifying factor inside the brackets. It can easily be shown that as Pi approaches 0 the new factor approaches 1. The big question is when does this factor become significant? As an example, this factor was evaluated for values of e = 0 to 9 for Table I, Reference 1. Using this factor, it was found that the error produced in M/T was less than 1% for e = 4 or less, 2.5% for e = 5, 10% for e = 6, and over 80% for e = 9.

As Pi increases the modifying factor decreases below 1. Obviously the higher the probability of a wrong decision on random data, the quicker that wrong decision can be made. The M/T does not rise quite as rapidly as implied in Reference 1 as a function of errors tolerated in the scan mode. Using this more accurate expression for the mean time to a decision, M, may shift the minimum M/T for a given pattern length in the direction of one or two more errors tolerance, or in general, a little more error tolerance in the scan mode, to produce the minimum time to establish synchronization at a given error rate. At least in the more typical systems of synchronization pattern lengths and spacings analyzed to data, this additional factor had a small percentage effect.

Acquire Vs. Reacquire Note that in the analysis in Reference 1 it is assumed that true sync existed in the past and the equations determine mean time to reach new true sync. If there was a sudden increase in signal strength, then there is no telling how far away the first sync pattern is. <u>On the average</u>, the first pattern will be one-half group away and thus, <u>on the average</u>, initial acquisition will be about one-half group less than the mean time to reacquire as calculated here. Another way of looking at it is to simply realize that this figure is on the safe side and assume that each time signal strength comes up to a usable point, a sync pattern has just past and a scan through one complete group of data is necessary to reach the first true sync position. Thus, the mean time to reacquire true sync is what the equations actually produce, <u>and on the average</u>, the mean time to initial acquisition at any fixed noise error rate is one-half group less.

Mean Time to Reacquire in the Absence of Noise Note that the probabilities Pc were taken for an assumed bit error rate of 0.1. Thus, the results apply to an extremely noisy condition of barely usable data as defined in References 3, 4 and5. Note also that a separate column in Table I, Reference 1 is presented for bit error probability of 0.01 or 1%. In every case, M/T, L and percent data lost will be smaller for little or no noise. For one thing, if a good sync pattern is chosen, the probability of-occurrence of false sync in all overlap groups can be made zero. However, it should be emphasized that this only reduces Pt to: $(1\text{-Pi})^{b+1\,-2(N-1)}$ and not to 0 since the pattern can still appear among random data. Thus, the mean time to reacquire true sync in the absence of noise can only approach 1 group unless the sync pattern can be inhibited from appearing in all data groups of N bits length. This does not seem practical.

CONCLUSIONS

In this paper it has been further justified that a relatively simple approach to instrumenting for PCM Pattern Synchronization provides the ability to very rapidly synchronize on PCM sync patterns in the presence of noise while maintaining a very low level of percentage data lost due to loss in synchronization in a constant noisy condition. Some minor refinements are made to the analysis of Mean Time to Acquire synchronization and percentage data lost due to loss in synchronization as previously reported.

More importantly, it has been shown that the effect of different "good" PGM synchronization patterns of any specific length, or chosen by different criteria, is negligible in the overall problem of acquiring synchronization rapidly and maintaining sync in the presence of noise.

REFERENCES

1. Williard, M. W., "Mean Time to Establish PCM Synchronization," Proceedings of the National Symposium on Space Electronics and Telemetry, Miami Beach, Florida, October 1962.

2. Williard, M. W., "Optimum Code Patterns for PCM Synchronization," Proceedings of the National Telemetering Conference, Washington D. C. , May 1962.

3. Williard, M. W. , "Pulse Code Modulated Telemetry Synchronization," Proceedings of the National Telemetering Conference, Chicago, Illinois, May 1961.

4. Williard, M. W., and Ratner, V. A., "A Universal PCM Data Collection System," Proceedings of the National Telemetering Conference, Santa Monica, California, May 1960.

5. Williard, M. W. "Pulse Code Modulated Telemetry, " Chapter 6, Page 168 Aerospace Telemetry, edited by Harry Stiltz, Prentice Hall, 1961.

6. Maury, J. L., and Styles, F. J. , "Development of Optimum Frame Synchronization Code for Goddard Space Flight Center PCM Telemetry Standards," Proceedings of the 1964 National Telemetering Conference, Los Angeles, California, June 1964.

7. Masching, R. G. , " A Simplified Approach to Optimal PCM Frame Synchronization Formats," Proceedings of the 1964 National Telemetering Conference, Los Angeles, California, June, 1964.

8. Goode, G.E., and Phillips, J. L. , "Optimum PCM Frame Synchronization Codes and Correlation Detection," Proceedings of the National Telemetering Conference, Chicago, Illinois, May 1961.

9. Magnin, J. P. , and Codrington, R. S. , "Legendre PCM Synchronization Codes, PGSET Record, 1962 Symposium on Space Electronics and Telemetry, Miami Beach, Florida, October 1962.

TABLE I
EXCELLENT SYNCHRONIZATION PATTERNS

Pattern Length	Pattern Octal Notation	Rt	
7	15	.8342	
8	35	.8951	
9	35	.9127	
10	73	.6972	
11	355	.8697	
12	153	.5804	
13	327	.5397	
14	547	.5532	
15	2467	.4508	
16	4727	.4202	
17	5317	.3892	
18	5317	.3483	
19	24637	.3226	
20	43667	.3303	
21	64567	.3252	
22	53317	.2936	
23	131657	.2899	
24	1147537	.2847	
25	1073237	.2827	
26	2153137	.2730	
27	3145537	.2664	
28	6323657	.2659	
29	5463657	.2662	
30	13147537	.2570	
31	13147537	.2578	30 bit pattern add 0 on left
32	13147537	.2616	31 bit pattern add 0 on left
33	13147537	.2685	32 bit pattern add 0 on left
33	26317277	.2671	32 bit pattern add 1 on right