IEEE 7th Int. Symp. on Spread-Spectrum Tech. & Appl., Prague, Czech Republic, Sept. 2-5, 2002

Improved Synchronization Performance in DSSS Systems Using Concatenated Sequences

Bernard Grundlehner, Jaap Haartsen, Rajan Srinivasan
University of Twente
Telecommunication Engineering Group
7500AE Enschede, The Netherlands
j.c.haartsen@el.utwente.nl

Abstract— **This paper deals with the synchronization properties of a family of spreading codes, called concatenated codes. New synchronization algorithms are introduced, and the performance of these algorithms is discussed both analytically and by simulation. It turns out that with a minimum of extra hardware, the mean phase acquisition time can be considerably reduced, while maintaining the reliability of the synchronization process at low power consumption.**

Keywords— **Spreading sequences, concatenated sequences, rapid phase acquisition**

I. Introduction

SPREAD-SPECTRUM techniques can be used for high bandwidth efficiency in multiple access systems, data security, and resistance against multipath fading. Long spreading codes are useful to fully exploit these properties. Phase acquisition can be a difficult and time-consuming task when longer codes are used, as more candidate phase offsets have to be tested. Using matched filters instead of active correlators can minimize the acquisition time, but involves a higher degree of hardware complexity and higher power consumption. Therefore, in most commercial systems active correlators are used. This paper will deal with a family of spreading codes, which are referred to in literature as concatenated codes ([1], [2] and [3]), Kronecker sequences ([4], [5], [6], [7] and [8]) or Composite sequences ([9]). It will be shown that these codes may enable the use of matched filters, thus reducing the synchronization time, while the power consumption is kept small. It is known ([8]) that these codes have a comparable performance compared to conventional codes, like Gold codes, if synchronized.

II. Concatenated codes

Let the level-1 sequence $\{C_1\}$ and the level-2 sequence $\{C_2\}$ be repetitive sequences of binary elements $C_{1,2}(j) \in \{+1,-1\}$ of period N_1 and N_2, respectively, so that $C_{1,2}(j) = C(j + kN_{1,2})$ for all integers k and for all $j \bmod N_{1,2}$. The concatenated sequence $\{S\}$ with elements $S(j) \in \{+1,-1\}$ is then constructed by a process equivalent to sequence-inversion keying (SIK) or by calculating the Kronecker (or Tensor) product, with N_1 as inner code and N_2 as outer code, so that $\mathbf{S} = \mathbf{C_1} \otimes \mathbf{C_2}$. This process can be applied repetitive to a level-3 sequence $\{C_3\}$ until the level-n sequence $\{C_n\}$, thus creating a total code $\mathbf{S} = \mathbf{C_1} \otimes \mathbf{C_2} \otimes ... \otimes \mathbf{C_n}$ of length $N_{\text{tot}} = \prod_{k=1}^{n} N_k$. An implication of this construction is that the clock rate of the generator of the level-k sequence (with $2 \leq k \leq n$) is N_{k-1} times slower than that of the generator of the level-$(k-1)$ sequence.

III. New synchronization algorithms

The phase acquisition process of concatenated codes is done in n steps, every m^{th} step synchronizes to the level-m sequence. The first step involves an on-line correlation operation over N_1 chips. The resulting synchronization point will serve as a timing reference for the correlator of level 2, which may operate off-line. The second step involves a correlation operation over N_2 symbols, which are the output values of the level-1 correlator, at the given timestamp. The level-2 correlator has to operate at a clock rate which is N_1 times slower than the clock rate of the first correlator. Whenever the synchronization process of the level-1 sequence is successful, the SNR of the input values will be N_1 times higher than the input values of the level-1 correlator. This can be repeatedly employed to levels 3 until n. The synchronization process of the level-1 sequence can be seen to operate under the most stringent conditions: correlator 1 operates at the highest clock rate and the input values of synchronizer 1 have the lowest SNR. The processing gain is now achieved step by step, whereas it would have been achieved at once when using conventional codes. The staged correlators are smaller than the long correlator which has to be used with conventional codes, and all the correlators but one run at a lower clock-rate.

To synchronize at a level, the in-phase autocorrelation peak of the preceding correlator has to be detected. This peak appears periodically with an unknown sign, so the absolute values have to be considered. The amplitude of the peak is proportional to the sequence length of that level. The reliability can be improved by basing the detection of the in-phase autocorrelation peak on the k most promising candidate phase offsets, which are all correlated by level-2 correlators. This parallel verification scheme is depicted in Figure 1, and described in more detail in [3] and [10].

A. Synchronizer

The synchronizer can be implemented in two ways. In the *MAX-algorithm*, the maximum of the correlator output within a fixed time window is

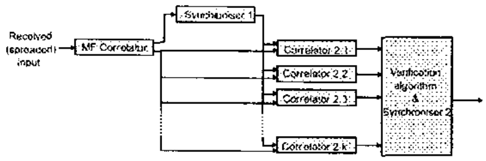

Fig. 1. Parallel verification scheme

taken. In the *threshold-algorithm*, the correlator output has to exceed a certain threshold. When using concatenated codes in this algorithm, every level should be correctly synchronized, otherwise the probability of exceeding a threshold on higher levels is very low. Therefore time-out is defined in the threshold algorithm, in order to prevent infinite loops to occur and to test the reliability of the preceding synchronization stages. A flow diagram of the threshold algorithm is given in Figure 2.

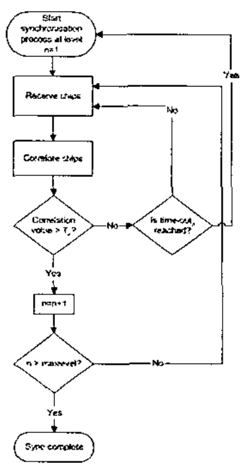

Fig. 2. Flow diagram of the threshold algorithm

B. Power consumption

The time-bandwidth product (TB), which is proportional to the number of taps N, can be used to calculate the power consumption. The relation between time T, the bandwidth B and the number of taps N is given by

$$N = B \cdot T \tag{1}$$

The power of the correlation operation per tap is relative to the bandwidth

$$P \triangleq N \times (\text{power per tap}) = N \cdot B = \frac{N^2}{T} \tag{2}$$

When comparing conventional codes to concatenated codes, codes of the same length with the same chiprate have to be compared. Therefore, we can define a normalized power $\tilde{P}$, with $T = 1$. When the system consists of more than one matched filter, the power from Equation (2) per filter can be summed to form $\tilde{P}$ of the complete system. Whenever conventional codes are used, there is only one correlator. The normalized power consumption is then equal to N^2. The normalized power consumption for concatenated codes can than be written as

$$\tilde{P}_{\text{concatenated}} = N_1 \cdot N_{\text{tot}} + \sum_{l=2}^{n} N_n \cdot \frac{N_{\text{tot}}}{\prod_{p=1}^{l-1} N_p} \tag{3}$$

When k_l parallel correlators are used at level l, as shown in Figure 1, $\tilde{P}$ can be written as

$$\tilde{P}_{\text{parallel}} = N_1 \cdot N_{\text{tot}} + \sum_{l=2}^{m+1} k_{(l-1)} \cdot N_k \cdot \frac{N_{\text{tot}}}{\prod_{p=1}^{l-1} N_p} + \sum_{l=m+2}^{n_{\max}} N_l \cdot \frac{N_{\text{tot}}}{\prod_{p=1}^{l-1} N_p} \tag{4}$$

It is clear that the power consumption of systems which make use of concatenated codes is always much lower than that of conventional systems. It must be noted that if parallel correlators are used, these correlators operate at a relatively low clock rate, so their contribution to $\tilde{P}$ is only limited.

IV. Theoretical performance of concatenated codes

In this section, the probability of correct synchronization as well as the mean phase acquisition time will be analytically evaluated for the AWGN channel, for systems without a parallel verification scheme, for both the MAX and the threshold algorithm. The derivation is explained in more detail in [10].

A. MAX synchronization algorithm

The MAX algorithm assumes that the maximum correlator output within a fixed time window is the in-phase correlation peak. Therefore, the time window is defined such that an in-phase correlation peak always occurs within it.

A.1 Probability of synchronization for the MAX algorithm

The input of the synchronizer consists of values from the odd and even correlation function. When level n is considered, there will be an in-phase correlation peak after every N_n input symbols, so the time window is assumed to have length N_n. Every combination of N_n succeeding correlation values is called a set. Let's define the event ξ to occur when synchronization is correct. When the maximum of

N_n succeeding input symbols is taken, the probability that this maximum corresponds with the correct synchronization point at level n is given by

$$\mathrm{P}_n(\xi) = \frac{1}{2(N_n-1)} \cdot \sum_{set=1}^{2(N_n-1)} (\mathrm{P}_{\mathrm{set},n}(\xi)) \cdot \prod_{k=1}^{n-1} (\mathrm{P}_k(\xi)) \quad (5)$$

with

$$\mathrm{P}_{\mathrm{set},n}(\xi) = \int_0^{\infty} \prod_{n=1}^{N_n-1} \left(Q\left(\frac{|\chi_{\mathrm{set}}(n)| - x}{\sigma \cdot \sqrt{N_n}}\right) - Q\left(\frac{|\chi_{\mathrm{set}}(n)| + x}{\sigma \cdot \sqrt{N_n}}\right)\right) \cdot (f_\nu(x - N_n) + f_\nu(-x - N_n))\, dx \quad (6)$$

where $\chi_{set}(n)$ denotes the n^{th} output value of the preceeding correlator, for that set, and with

$$\sigma = \frac{\sigma_1}{\prod_{l=1}^{n-1} \sqrt{N_l}} \quad (7)$$

and $f_\nu(x - N_n)$ the distribution of the noise around the peak value N_n, with a variance of $\sigma^2 \cdot N_n$. The standard deviation of the noise of the received chips is σ_1.

A.2 Mean phase acquisition time of the MAX algorithm

When using the MAX algorithm, the synchronization time only depends on N_n, for every level n. It can be written in terms of chip-times as

$$T_{\mathrm{acq,MF}}(n) = \begin{cases} 2 \cdot N_1 - 1 & , \text{for } n = 1 \\ 2 \cdot \prod_{j=1}^{n} (N_j) - \frac{1}{2} \prod_{k=1}^{n-1} (N_k), & \text{for } n > 1 \end{cases} \quad (8)$$

The total phase acquisition time can be determined by taking the sum over all n.

B. Threshold synchronization algorithm

The threshold algorithm is explained in Figure 2. For every level n, a threshold T_n and a time-out $t_{max,n}$ must be defined. The values of these parameters can be obtained by optimizing the equations which are presented in the next sections.

B.1 Probability of synchronization for the threshold algorithm

The analysis of the probability of correct synchronization will be done in two steps, see case A and B in Figure 2. In order to calculate the mean synchronization time, an event ζ_τ is defined, which denotes that the threshold is exceeded after processing τ input symbols. To calculate the probability of correct synchronization, an event ξ_τ is defined, which denotes that the threshold is exceeded at a correct phase offset, after τ input symbols.

Case A. Let's define N_{n+1} sets containing succeeding output values from correlator n, each set starting at an in-phase correlation peak, and each set consisting of values from the odd and even correlation functions (these correlation functions are jointly denoted by χ_n, see [10]). Within each set, there are N_n possible subsets. As the polarity of the values in the set is not known, the absolute values are considered. A set has a length between 1 and infinity, as it ends when the threshold is exceeded, which can occur after an arbitrary number of values.

The probability of exceeding the threshold of the level-n sequence (the event ζ_n) after receiving τ input symbols, $\mathrm{P}_\tau(\zeta_n)$, can be written as

$$\mathrm{P}_\tau(\zeta_n) = \mathrm{P}_\tau(\zeta_n | \xi_{1..n-1}) \cdot \prod_{j=1}^{n-1} (\mathrm{P}(\xi_j)) + \mathrm{P}_\tau(\zeta_n | \overline{\xi_{1..n-1}}) \cdot \left(1 - \prod_{j=1}^{n-1} (\mathrm{P}(\xi_j))\right) \quad (9)$$

this equation represents the above-mentioned probability, when the underlying levels are correctly synchronized, and the probability when this is not the case, respectively. The probabilities for ζ_n will be given in Equation (10) until (13), whereas the probability of *correct* synchronization, ξ_n, can be found in Equation (15) until (16).

For the case that the underlying levels are correctly synchronized, we can write for the probability of exceeding the threshold at level n after processing τ input values

$$\mathrm{P}_\tau(\zeta_n | \xi_{1..n-1}) = \frac{1}{N_{n+1}} \cdot \sum_{\mathrm{set}=1}^{N_{n+1}} \mathrm{P}_{\mathrm{set},\tau}(\zeta_n | \xi_{1..n-1}) \quad (10)$$

where the probability of exceeding the threshold for a specific set can be calculated by averaging over the possible subsets (offsets of that set)

$$\mathrm{P}_{\mathrm{set},\tau}(\zeta_n | \xi_{1..n-1}) = \frac{1}{N_n} \sum_{j=1}^{N_n} \left\{ \left(Q\left(\frac{T_n - |\chi_{\mathrm{set}}(j)|}{\sigma_n \cdot \sqrt{N_n}}\right) + Q\left(\frac{T_n + |\chi_{\mathrm{set}}(j)|}{\sigma_n \cdot \sqrt{N_n}}\right)\right) \cdot \left[\prod_{k=(N_i-2-\tau+j)}^{j-1} \left(Q\left(\frac{|\chi_{\mathrm{set}}(k \bmod N_i)| - T_n}{\sigma_n \cdot \sqrt{N_n}}\right) - Q\left(\frac{|\chi_{\mathrm{set}}(k \bmod N_i)| + T_n}{\sigma_n \cdot \sqrt{N_n}}\right)\right)\right]\right\} \quad (11)$$

Where $\chi_{\mathrm{set}}(j)$ denotes the j^{th} correlation value of the set. The standard deviation of the noise is given by

$$\sigma_n = \frac{\sigma_1}{\prod_{k=1}^{n-1} \sqrt{N_k}} \quad (12)$$

Whenever the synchronization process fails at level l, but has succeeded for levels 1 until $l-1$, the probability that the threshold is still exceeded before the time-out is reached, is given by

$$\mathrm{P}_\tau\left(\zeta_n|\overline{\xi_{1..n-1}}\right) = \sum_{l=1}^{n-1}\left[\left\{Q\left(\frac{T_n-0.5\cdot N_n}{\sigma_n\cdot\sqrt{N_n}}\right)+ Q\left(\frac{T_n+0.5\cdot N_n}{\sigma_n\cdot\sqrt{N_n}}\right)\right\}\cdot\left\{Q\left(\frac{0.5\cdot N_n-T_n}{\sigma_n\cdot\sqrt{N_n}}\right)- Q\left(\frac{0.5\cdot N_n+T_n}{\sigma_n\cdot\sqrt{N_n}}\right)\right\}^{\tau-1}\cdot\frac{1-\mathrm{P}(\xi_l)}{\prod_{k=1}^{l}(1-\mathrm{P}(\xi_k))}\right] \quad (13)$$

where σ_n is given by

$$\sigma_n = \begin{cases} \dfrac{\sqrt{\sigma_1^2+\frac{1}{2}}}{\prod_{k=1}^{n-1}\sqrt{N_k}} & \text{, for } l=1 \\ \dfrac{\sqrt{\sigma_1^2+\frac{1}{2}\cdot\prod_{k=1}^{l-1}\left(N_k^2\right)}}{\prod_{k=1}^{n-1}\sqrt{N_k}} & \text{, for } l>1 \end{cases} \quad (14)$$

The term $0.5\cdot N_n$ refers to the mean absolute correlator output.

The probability of *correct* synchronization of the level-n sequence can be written as

$$\mathrm{P}(\xi_n) = \sum_{\tau=1}^{t_{\max,n}-1}\left\{\frac{1}{N_{n+1}}\cdot\sum_{\mathrm{set}=1}^{N_{n+1}}\mathrm{P}_{\mathrm{set},\tau}(\xi_n)\right\} \quad (15)$$

where the time-out of level 1, $t_{\max,1}$, should be infinity. The probability of correct synchronization per set is given by

$$\mathrm{P}_{\mathrm{set},\tau}(\xi_n)=\frac{1}{N_n}\left\{\left(Q\left(\frac{T_n-N_n}{\sigma_n\cdot\sqrt{N_n}}\right)+Q\left(\frac{T_n+N_n}{\sigma_n\cdot\sqrt{N_n}}\right)\right)\cdot \prod_{k=(N_n-1-\tau)}^{0}\left(Q\left(\frac{|\chi_{\mathrm{set}}(k \bmod (N_n\cdot N_{n+1}))|-T_n}{\sigma_n\cdot\sqrt{N_n}}\right)- Q\left(\frac{|\chi_{\mathrm{set}}(k \bmod (N_n\cdot N_{n+1}))|+T_n}{\sigma_n\cdot\sqrt{N_n}}\right)\right)\right\} \quad (16)$$

where σ_n can be found in Equation(12).

Case B. Whenever the time-out is reached, the synchronization restarts at level 1, see Figure 2. The probability of correct synchronization, given that the higher levels are all synchronized, can be easily calculated using Bayes' rule, where Equations (9), (10) and (15) can be filled in.

B.2 Mean phase acquisition time of the threshold algorithm

The expected number of times $\mathrm{E}\{N_{\mathrm{TO},n}\}$ the time-out of level n will be reached is given by

$$\mathrm{E}\{N_{\mathrm{TO},n}\}=\sum_{i=1}^{\infty}\left(\mathrm{P}_{\mathrm{NoTO}}\cdot i\cdot(1-\mathrm{P}_{\mathrm{NoTO}})^i\right) \quad (17)$$

where the probability that the time-out is not reached can be written as

$$\mathrm{P}_{\mathrm{NoTO}} = \mathrm{P}(\zeta_2|\xi_1)\cdot\mathrm{P}(\xi_1)+\mathrm{P}\left(\zeta_2|\overline{\xi_1}\right)\cdot\mathrm{P}\left(\overline{\xi_1}\right) \quad (18)$$

The mean synchronization time for level n, in terms of chip-times, becomes

$$\overline{T_{\mathrm{acq},n,\mathrm{MF}}} = \mathrm{E}\{N_{\mathrm{TO},n}\}\cdot\left(t_{\max,n}\cdot N_n+\prod_{k=1}^{n}(N_k)-1+\sum_{k=1}^{n-1}\overline{T_{\mathrm{acq},k,\mathrm{MF}}}\right)+ \sum_{\tau=1}^{t_{\max,n}}(\tau\cdot\mathrm{P}(\zeta_\tau))+\prod_{k=1}^{n}(N_k)-1 \quad (19)$$

The total phase acquisition time can be calculated by simply summing the synchronization time per level for all levels.

V. Simulation results

In the previous section, no equations have been derived for the parallel verification scheme, as the equations would become very cumbersome. This algorithm has been implemented in a simulation program. The results of this simulation, as well as the calculation of the performance of a conventional code, will be presented here. The performance of a conventional code, an m-sequence of length 1023, is compared to that of a concatenated code, that consists of two levels, both m-sequences of length 31, with 11 parallel correlators. From equation (2), it can be seen that the normalized power consumption, $\tilde{P}$, is approximately $1\cdot10^6$ when using conventional codes and $40\cdot10^3$ when using concatenated codes with 11 parallel correlators.

The following values of the thresholds of level 1 and 2, and the time-out of level 2 were found after coarsely optimizing them: 36, 620 and 961, respectively.

In Figure 3, the probability of correct synchronization is shown as a function of the SNR. The conventional code has the best performance in all cases, with a difference up to 1.1 dB for the threshold algorithm. The MAX algorithm performs generally better than the threshold algorithm.

Figure 4 shows the performance in terms of mean phase acquisition time. It must be noted that the y-axes are differently scaled in the subplots. The conventional code is equipped with active correlators in this case, whereas the concatenated code is equipped with matched filters. It can be seen that

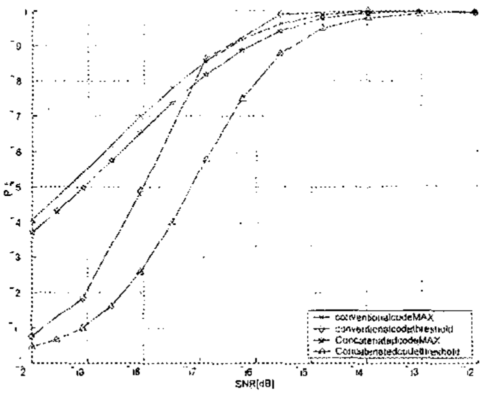

Fig. 3. Probability of correct synchronization as a function of SNR

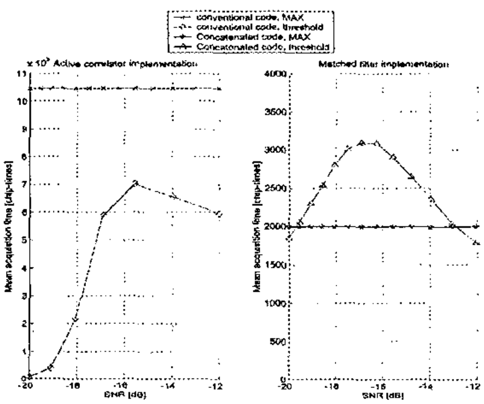

Fig. 4. Mean phase acquisition time as a function of the SNR

there is an enormous reduction in synchronization time when concatenated codes are used. With 11 parallel correlators, concatenated codes synchronize more than 300 times as fast at -17 dB SNR for the threshold algorithm. The MAX algorithm performs even better. It must be noted that it is also possible to use parallel active correlators in order to speed up the synchronization process when using conventional codes. Nevertheless, concatenated codes in combination with matched filters will always reduce the mean synchronization time, at the expense of minimal extra hardware complexity.

VI. Discussion and conclusions

From the preceding sections, it is clear that the probability of correct synchronization when using concatenated codes can approach that of conventional codes, when parallel correlators are used. The codes enable the use of matched filters, while keeping the power consumption minimal. The mean phase acquisition time can be drastically reduced, whereas there is only a small increase in hardware complexity. The advantage of concatenated codes, in comparison to conventional codes, is bigger as the total code-length increases. From literature, it is already known that, if receiver and transmitter are in synchronization, there is no difference in performance between conventional codes and concatenated codes. Another important property of the concatenated codes is that a number of adaptive algorithms can be introduced. The well-known RASE synchronizers ([11]) could, for example, be introduced at a point where the SNR is high enough to permit the use of this extremely fast synchronization algorithm, in order to speed up the synchronization process even more. The MAX algorithm is a very attractive choice when concatentated codes are used, not only because of the good performance, but also because there is no need to optimize parameters, the synchronization time is constant and the algorithm is less complex than the threshold algorithm.

References

[1] S.L. Maskara and J. Das, "Concatenated sequences for Spread Spectrum Systems," *IEEE Trans. on Acrospace and Electronic Systems*, vol. AES-17 No. 3, pp. 342-350, May 1981.

[2] M. Beale, "Comments on "Concatenated sequences for Spread Spectrum Systems"", *IEEE Trans. on Aerospace and Electronic Systems*, Vol.AES-20, no.6, p.829, Nov. 1984.

[3] J.C. Haartsen, *Method and Apparatus for Sequential Correlation*, patent SE-515911, Oct. 29, 2001.

[4] W.E. Stark and D.V. Sarwate, "Kronecker sequences for Spread-Spectrum Communications", *IEE Proc. Pt. F. Comm., Radar and Signal Proc.*, Vol. 128, Pt. F, No. 2, pp.104-109, April 1981.

[5] W.E. Stark, *Kronecker Sequences for Spread Spectrum Communications*, M. Sc. thesis, University of Illinois at Urbana-Champaign, Urbana IL, 31 p., July 1979

[6] K.H.A. Kärkkäinen and P.A. Leppänen, "Design of Kronecker and Combination Sequences and Comparison of Their Correlation, CDMA and Information security Properties", *IEICE Trans. Commun.*, VOL. E81-B, No. 9, pp. 1770-1778, September 1998.

[7] M. Beale, *Direct-Sequence Spread-Spectrum Multiple Access Systems*, Ph.D dissertation, University of Kent, Canterbury UK, 447 p., September 1982.

[8] K.H.A Kärkkäinen and P.A. Leppänen, "Performance of an asynchronous DS/SSMA (DS/CDMA) system with Kronecker sequences", *Proc. IEEE The Second International Symposium on Personal, Indoor and Mobile Radio Comm.*, *London*, UK, September 23-25, pp. 191-197, 1991.

[9] M. Beale and T.C. Tozer, "A class of composite sequences for Spread-Spectrum Communications", *IEE Journal on Comp. and Digital Techn.*, Vol. 2, pp. 87-92, April 1979.

[10] F.B. Grundlehner, *Concatenated Sequences for Spread Spectrum Systems*, M. Sc. thesis, University of Twente, Enschede, The Netherlands, September 2001.

[11] Roger L. Peterson, Roger E. Ziemer and David E. Borth, *Introduction to spread spectrum communications*, Prentice-Hall 1995.