ESA HQ, Paris (France), 3-7 May 2004

Document SLS-RNG_04-05

Clarifications on NASA/JPL PN ranging codes (AI_03-01 and AI_03-02 from CCSDS Fall Meeting)

Giovanni Boscagli (ESA)

1. Introduction

This paper summarises the status of the two following action items from the CCSDS Fall Meeting:

- AI_03-01. Contact J.Berner on probability of acquisition and integration time formulas
- AI_03-02. Contact J.Berner on other ranging codes (Massey, etc.)

In section 4 you can find the e-mail sent by G.Boscagli to J.Berner indicating questions and asking for clarifications.
In section 3 the J.Berner's answers are reported.

2. References Documents

[*REF-1*] *"Operation Comparison of Deep Space Ranging Types: Sequential Tone vs Pseudo-Noise''*, J.F.Berner, S.C.Bryan, IEEE 2002

[*REF-2*] *"Advantages of Regenerative Ranging for Deep Space Navigation"*, J.F.Berner, P.W.Kinman, J.M.Layland, presentation at DESCANSO Symposium

[*REF-3*] *"Regenerative Pseudo-noise Ranging for Deep Space Applications"*, J.F.Berner, J.M.Layland, P.W.Kinman, R.Smith, TMO Progress Report 42-137, May 1999

[*REF-4*] *"The NASA Spacecraft Transponding Modem"*, J.F.Berner, S.Kayalar, J.D.Perret, IEEE 2000

[*REF-5*]: "*Fast Acquisition Sequences"*, J.Ganz, A.P.Hilgen, J.L.Massey, Proc.6th Int Symp. on Comm. Theory and Appl., Ambleside, England, 15-20 July 2001, pp471-476

[*REF-6*] "*Study of PN Ranging Code for Future Mission*", DTOS-GS-SOW-1001-TOS-OW, 12 November 2003.

[*REF-7*] *"Regenerative Ranging"*, G.Boscagli, L.Simone, D.Gelfusa, CCSDS SLS-RFM_03-05, October 2003

[*REF-8*] *"Pseudo Noise Regenerative Ranging: Simplified Analysis of the Code Correlators Output"*, L.Simone (in ANNEX-1 in this paper)

3. E-mail received by J.Berner in date 12/01/2004

Giovanni,

I'm sorry that it has taken so long to respond. I was out for most of December (and the last part of November), and am just now finally getting caught up with all the email that came in that time period.

As some background, the regenerative ranging work was done as part of the development of the Spacecraft Transponding Modem (STM). Unfortunately, the STM funding was cancelled before we had any of our prototype hardware completed, so no testing or bread boarding was ever done. Note that the PN ranging (discussed in your Reference 1)[1] is independent of the regenerative work - that implementation is much more generic, allowing for different PN code combinations.

Let me see if I can answer some of your issues:

1. Why Pr/N0 of 27 dB-Hz? -
That was the STM requirement. As discussed above, we have no experimental data at this time.

2. Why was this code selected? -
We built upon the experience of John R.Smith and Robert Tausworth (nee Titsworth), who had been doing ranging in the DSN for 30 plus years. The code that they suggested seemed to meet all our requirements and was implementable, so we did not pursue the issue further. It is certainly possible that there are other codes out there, but this one did the job (and we had a very limited development time). Note that as was pointed out in Reference 1, this code is not good for a non-regenerative case.

3. DEFINITION OF "T" -
Tint is the time needed to accumulate the data for a measurement. Depending on the implementation, you could do the 23 measurements in parallel or series, so the final integration time depends on that.

4. *ACQUISITION TIME AND PROBABILITY*
You are probably correct that the integration time may be optimistic. I believe that the STM logic had a requirement of two consecutive successes or failures to change state, but it has been 5 years and I would have to dig up a lot of old notes. The original papers were planned as the first of a set (with implementation detail and test results planned for the later papers), but the cancellation of the STM ended that plan.

Jeff

[1] See REF-1

4. E-mail sent to J.Berner in date 28/11/2003

Dear J.Berner,

I am working on regenerative ranging and, as ESA technical officer, I started some activities in preparation to the future deep space missions. Both activities are with Alenia Spazio (Italy, Rome):

- **Activity 1 -** To develop the on-board regenerative ranging DSP inside the an FPGA
- **Activity 2 -** Pre-development of BepiColombo transponder (Eng. Model) including the regenerative ranging function.

The first activity (*) is already finished; we tested the modem (PN code acquisition and tracking) using a complete RF front-end (breadboard), while the second activity is started a few weeks ago.

Our present baseline is based on your papers, indeed we have implemented the same PN codes and the DSP approach is basically the same.
In order to generate a "*standard*", a working group has been organized inside the CCSDS and, during the Fall Meeting 2003, I made a presentation (see attached .ppt file)[2] including also the results of Activity 1 from Alenia Spazio.
The word document[3] reports a simplified analysis (performed by Alenia) on the PN acquisition performances.

There are some points that are not clear to me and that are underlined in my presentation as well. These points were formalized with the following two actions during the CCSDS 2003 Fall Meeting.

AI #	AI description	Actionee	Due date
AI_03-01	Contact J.Berner on probability of acquisition and integration time formulas	G.Boscagli	30.11.03
AI_03-02	Contact J.Berner on other ranging codes (Massey, etc.)	G.Boscagli	30.11.03

AI_03-01
Regarding this action you can refer to page 15 of my presentation (.ppt file)[4]. There are two bullets in this page. Please, could you reply to them?

1st bullet - ACQUISITION TIME AND PROBABILITY - The evaluation of Appendix B of your paper[5] seems to be optimistic because it seems that you do not consider the false detection probability. Alenia approach (please refer to the simplified analysis reported in the file .doc[6]) shows longer integration time (also validated via experimental results).
Note that:

---
[2] See REF-7
[3] See REF-8
[4] See REF-7
[5] See REF-1
[6] See REF-8

- Alenia made the analysis evaluating the integration time in order to select a proper detection threshold (given the detection probability and the false detection probability).
- Alenia experimentally checked also the statistics (average value and rms) at the detector output, confirming the results of the numerical evaluation.

2nd bullet - DEFINITION OF "T" IN YOUR PAPER[7] - Is "T" the integration time for each detection test (code phase)? If yes, considering T=18 sec (as indicated in your paper for Pr/No=27 dBHz) for the sequence number "6" (length = 23 chip), we can approximately assume a maximum acquisition time (for the whole sequence) of 23x18=414 seconds. Is it correct?

Other questions - Why are you assuming 27 dBHz as the minimum (worst case condition) value for the on-board ranging signal power over noise spectral density? Please, could you justify this requirement[8]: *minimum Pr/No 10 dB higher than the minimum signal over noise spectral density (17 dBHz)*. Have you characterized the PN sequence acquisition/tracking performances for different up-link signal power? Do you have experimental data?

AI_03-02

When you started to study the PN ranging and the On-Board Regenerative approach, did you perform a trade-off on the different available PN codes? Why have you selected the Titsworth's codes? In my presentation (starting from page 16), you can find a trade-off summary on different types of sequence. I have taken all these data from a paper[9] (see at page 16 of the presentation[10] for the reference). The trade off is based on the minimization of the sequential acquisition time.

At present, with the opportunity of using very large ASIC for on board application, parallel processing is possible and your and Alenia approach is based on it. However (as you can see in ALS block diagram at page 11 of the presentation), we have a sequential search for each of the 6 sub-sequences. Please could you comment this issue indicating if we can expect sequences with better performances with respect to the codes at present implemented by you and Alenia.

Best Regards
Giovanni

[7] See REF-3
[8] See REF-3
[9] See REF-5
[10] See REF-7

ANNEX-1

Pseudo-Noise Regenerative Ranging:
Simplified Analysis of the Code Correlators Output

Lorenzo Simone,
Alenia Spazio – via Marcellina 11, 00131 Rome (Italy)
l.simone@roma.alespazio.it

This note provides a simplified analysis of the PN code correlators output that is in agreement with the experimental results.
In what follows, we refer to the block diagram depicted in Figure 1, representing the signal processing performed by a code component correlator.

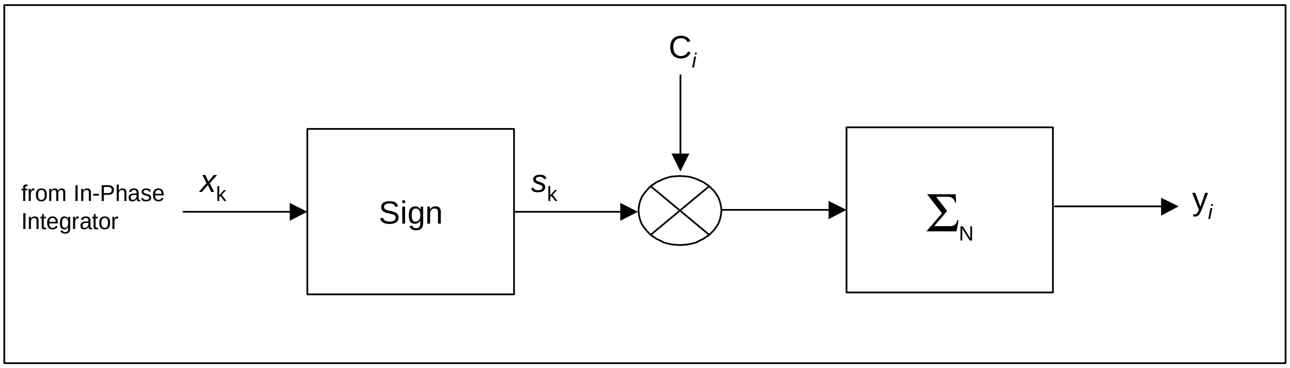

Figure 1 Code component correlator block diagram

Let's start from the C_1 component that consists in an alternating sequence of +1 and –1. Assuming that the chip synchronization has been already achieved, the in-phase integrator output can be written as:

$$s_k = \text{sign}[x_k] = \text{sign}[r_k + n_k] \tag{1}$$

where:

$$\begin{aligned} x_k &= \text{in - phase integrator input} \\ r_k &= \text{useful signal @ in - phase integrator input} \\ n_k &= \text{additive noise @ in - phase integrator input} \end{aligned}$$

In order to evaluate the statistic at the in-phase integrator output, let assume that a '+1' was sent. Exploiting the Theorem of Central Limit, x_k can be modelled as a Gaussian variable having probability density function $p(x_k)$ given by:

$$p(x_k) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(x_k - \mu)^2}{2\sigma^2}\right] \tag{2}$$

where:

$$\frac{\mu}{\sigma} = \sqrt{2} \cdot 10^{(E_c/N_0)/20} \tag{3}$$

being E_c/N_0 the energy chip-over-noise spectral density ratio in dB, i.e.:

$$\frac{E_c}{N_0} = \frac{P_r}{N_0} - 10 \cdot \log_{10}(R_c) - L \tag{4}$$

being:

- P_r/N_0 = ranging power-over-noise spectral density ratio (dBHz)
- R_c = chip rate (chip/s)
- L = carrier demodulation loss equal to 2 dB

The in-phase integrator output s_k has mean and variance given by:

$$E(s_k) = \int_{-\infty}^{+\infty} \text{sign}(x_k)\, p(x_k) dx_k = \int_{0}^{+\infty} p(x_k) dx_k - \int_{-\infty}^{0} p(x_k) dx_k = \frac{1}{2}\text{erfc}\left(-\frac{\mu}{\sqrt{2}\sigma}\right) - \frac{1}{2}\text{erfc}\left(\frac{\mu}{\sqrt{2}\sigma}\right) = \text{erf}\left(\frac{\mu}{\sqrt{2}\sigma}\right) \tag{5}$$

$$Var(s_k) = E(s_k^2) - \left[E(s_k)\right] = 1 - \text{erf}^{\,2}\left(\frac{\mu}{\sqrt{2}\sigma}\right) \tag{6}$$

In order to simplify the analysis, we approximate the PN code with a squarewave. In this case the output y_1 of the integrate-&-dump has the following mean:

$$E(y_1) = N \cdot E(s_k) \tag{7}$$

when the local code component is in phase with the transmitted code component, and:

$$E(y_1) = -N \cdot E(s_k) \tag{8}$$

when the local code component is in phase opposition with respect to the transmitted code component. The variance of the output y_1 of the integrate-&-dump is given by:

$$Var(y_1) = N \cdot Var(s_k) \tag{9}$$

Likewise, the mean and variance of the output y_i (i=2,3,…,6) of the other correlators can be expressed as:

$$E(y_i) = \frac{N \cdot E(s_k)}{10^{13/10}} \tag{10}$$

$$Var(y_i) = N \cdot Var(s_k) \tag{11}$$

The previous relationships have been obtained considering N equal to a multiple of the PN sequence period and using the fact that the correlation peak of the component C_2, …, C_6 is 13 dB below that of C_1 component.

The following table shows a perfect matching between the theoretical results and the experimental results (the expected values are relevant to the correlation peak).

P_r/N_0	Chip Rate	n	Theoretical Results			Test Results		
			E (y_1)	E (y_i), i =2,…,6	Var (y_i), i =2,…,6	E (y_1)	E (y_i), i =2,…,6	2×Var (y_i), i =2,…,6
26	1 Mchip/s	5	85297	4275	2235	~80000	Not measurable	~4000
33	1 Mchip/s	5	190893	9567	2234	~200000	Not measurable	~4000
33	1 Mchip/s	16	610859	30615	3997	~530000	~28000	~8000
35	1 Mchip/s	16	768837	38533	3994	~700000	~32000	~8000
39	1 Mchip/s	5	380406	19065	2228	~400000	~18000	~4000

Table 2 Theoretical results vs. Test results
(*n* is the integration length expressed as number of period of the overall PN sequence, i.e.: N=1,009,470×n)

Note that the value of 'n' in the previous table is relevant to the number of period of the entire sequence which are needed for each correlation. As an example, considering n=16 and assuming a chip rate equal to 1 Mchip/s leads to a maximum acquisition time which is equal to 23×16 sec = 368 sec, being 23 the length of the longest subsequence of the incoming code.

Using the statistics at the correlator branches output allows to estimate the correct detection $P_d(i)$ and false detection $P_{fd}(i)$ probabilities according to the following relationships:

$$P_d(i) = \frac{1}{2}\operatorname{erfc}\left[\frac{T(i)\text{-}E(y_i)}{\sqrt{2Var(y_i)}}\right] \qquad P_{fd}(i) = \frac{1}{2}\operatorname{erfc}\left[\frac{T(i)}{\sqrt{2Var(y_i)}}\right] \tag{12}$$

being $T(i)$ the user defined thresholds and i=1,2,…,6.

The equations presented in this note have been implemented in the MATLAB program reported in Table 2.

```
% ----------------------------
% C1 Code Correlation Analysis
%    Lorenzo Simone, ALS
%          19/10/03
% ----------------------------

clear all
Pr_No=input('Pr/No (dBHz) = ');
Rc =input ('Chip Rate (chips/s) = ');
N = input ('Integration Window (chips) = ');
L=2; % implementation loss (dB)
Ec_No=Pr_No-10*log10(Rc)-L;

s=1; % normalized standard deviation
m=s*sqrt(2)*(10^(Ec_No/20));

% In-Phase Integrator
% -------------------
m1=erf(m/(sqrt(2)*s)); % mean
s1=sqrt(1-m1^2); % standard deviation

% Integrator Output
% -----------------
m2=0.954*N*m1; % mean (th 0.954 factor takes into account the correlation loss for
the C1 component)
s2=sqrt(N)*s1; % standard deviation
SNR=20*log10(m2/s2); % output SNR
fprintf('\n')
disp ('C1 Component')
disp ('------------')
disp (['Correlator Output (mean) = ',num2str(m2)])
disp (['Correlator Output (std) = ',num2str(s2)])
disp (['Correlator Output (SNR) = ',num2str(SNR),' dB'])
T1=input('C1 Component Threshold = ');
Pd1=.5*erfc((T1-m2)/(sqrt(2)*s2));
Pfd1=.5*erfc((m2)/(sqrt(2)*s2));
disp (['C1 Correct Detection Prob. = ',num2str(Pd1)])
disp (['C1 False Detection Prob. = ',num2str(Pfd1)])

fprintf('\n')
disp ('C2-C6 Components')
disp ('----------------')
m3=m2/10^(13/10);
s3=s2;
SNR=20*log10(m3/s3); % output SNR
disp (['Correlator Output (mean) = ',num2str(m3)])
disp (['Correlator Output (std) = ',num2str(s3)])
disp (['Correlator Output (SNR) = ',num2str(SNR),' dB'])
T2=input('C2-C6 Components Threshold = ');
Pd2=.5*erfc((T2-m3)/(sqrt(2)*s3));
Pfd2=.5*erfc((m3)/(sqrt(2)*s3));
disp (['C2-C6 Correct Detection Prob. = ',num2str(Pd2)])
disp (['C2-C6 False Detection Prob. = ',num2str(Pfd2)])
```

Table 2 MATLAB simulation software for code correlators output estimation

The MATLAB simulation software provided in Table 2 has been used to evaluate the integration time needed to acquire the PN code components in the following operative conditions:

- P_r/N_0 = 27 dBHz
- Chip Rate = 2 Mchip/s
- Required $P_d > 99.5$ %
- Required $P_{fd} < 10^{-6}$

The simulation software output is reported in the Table 3, which shows that an integration time of **30 s** is required to acquire the each PN code components with the specified correct detection and false detection probabilities. **It means that the worst-case acquisition time is equal to 23×30 sec = 690 sec, being 23 the length of the longest subsequence of the incoming code**.

```
Pr/No (dBHz) = 27   (user input)
Chip Rate (chips/s) = 2e6 (user input)
Integration Window (chips) = 60e6 (user input)

C1 Component
---------------------
Correlator Output (mean) = 812113.6267
Correlator Output (std) = 7745.187
Correlator Output (SNR) = 40.4117 dB
C1 Component Threshold = 100000 (user input)
C1 Correct Detection Prob. = 1
C1 False Detection Prob. = 0

C2-C6 Components
----------------------------
Correlator Output (mean) = 40702.0982
Correlator Output (std) = 7745.187
Correlator Output (SNR) = 14.4117 dB
C2-C6 Components Threshold = 20000 (user input)
C2-C6 Correct Detection Prob. = 0.996
C2-C6 False Detection Prob. = 7.3953e-008
```

Table 3 MATLAB simulation software: output example