

# ANALYSIS AND DESIGN OF THE SECOND ORDER DELAY-LOCK LOOP IN A CDMA SYSTEM

J.J.Olmos, R.Agustí

Dept. de Teoria del Senyal i Comunicacions  
Universitat Politècnica de Catalunya (UPC)  
Apdo. 30.002, 08080 Barcelona, SPAIN

## ABSTRACT

In CDMA systems a DLL is used to keep a fine alignment between sequences. The differential equation of an incoherent second-order DLL has been programmed. By numerical simulation we have obtained the *Mean Time to Lose Lock* of the DLL as a function of the signal to noise ratio in the data bandwidth. This result is important in a mobile environment where a deep fading may result in a practically instantaneous loss of lock.

## INTRODUCTION

CDMA systems require synchronization between the received sequence and the locally generated sequence. Once the coarse alignment of the codes has been achieved, the received code phase has to be tracked in order to compensate for the doppler effect and (or) clock rate mismatch. This is done by means of a Delay-Lock Loop (DLL). Second order DLL's are able to track a frequency error without a constant phase error. This capability is important in CDMA mobile communications, where a first order DLL would easily lose lock, [1].

The differential equation of an incoherent second-order DLL has been programmed. The program allows us to obtain the maximum frequency offset that the DLL can absorb in terms of the natural frequency of the loop. Knowing this we design the optimum loop filter. We have found a formula that gives the minimum achievable timing jitter in terms of the signal to noise ratio, the product of the VCO frequency stability times the system processing gain and the time offset between the early and late codes in the DLL.

The Mean Time to Lose Lock (MTLL) of the DLL is a very important design parameter. By numerical simulation we have obtained this parameter as a function of the timing jitter. Furthermore, an exponential regression formula has been obtained from the sample points in order to easily predict the MTLL. Combining this formula with the formula of the jitter we are able to plot the MTLL as a function of the signal to noise ratio in the data bandwidth. This result is again important in a mobile environment where a deep fading may result in a practically instantaneous loss of lock.

## DLL DESCRIPTION

The block diagram of an incoherent DLL is shown in figure 1.

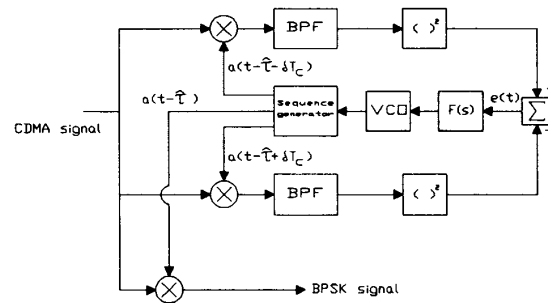


Figure 1: Block diagram of an incoherent DLL

The received CDMA signal can be written as,

$$y(t) = \sqrt{2P} b(t) a(t-\tau) \cos(\omega_0 t + \theta) + n(t) \quad (1)$$

where  $a(t)$  is the transmitted PN sequence,  $b(t)$  is the data sequence and  $\omega_0$  and  $\theta$  are, respectively, the carrier pulsation and phase.  $a(t)$  takes on values from the set

$\{\pm 1\}$  each  $T_c$  seconds (chip period), and  $b(t)$  takes on values from the set  $\{\pm 1\}$  each  $T$  seconds (bit period), where  $T \gg T_c$ .  $P$  is the received signal power and  $\tau$  is the propagation delay to which the DLL must lock.  $n(t)$  is white gaussian noise with spectral density  $G_n(f) = N_0/2$ . In figure 1,  $2\delta$  is the offset between the early and late codes, and the bandwidth of the band-pass filters,  $B \approx 1/T$  allows the filtering of the data signal without distortion. The behavior of the DLL can be described by a differential equation, [2]:

$$\frac{d\epsilon}{dt} = \frac{1}{T_c} \frac{d\tau}{dt} - KP f(t) * (S(\epsilon, \delta) + N(t)) \quad (2)$$

In (2),  $\epsilon \Delta (\tau - \tau)/T_c$  is the shift between the local and the received sequence in chips,  $d\tau/dt$  is the initial clock frequency offset,  $K$  is the VCO constant,  $f(t)$  is the low-pass filter impulse response,  $S(\epsilon, \delta)$  is the "S"-curve and  $N(t)$  is noise with spectral density near the origin given by, [2]:

$$G_N(0, \epsilon) = \frac{2N_0}{P} \left[ \lambda(\epsilon, \delta) + \frac{1-g^2(\delta)}{\gamma} \right] \quad (3)$$

where  $\gamma = P/N_0B$  is the signal to noise ratio in the data bandwidth, and

$$\lambda(\epsilon, \delta) \Delta R_a^2[(\epsilon - \delta)T_c] + R_a^2[(\epsilon + \delta)T_c] - 2g(\delta)R_a[(\epsilon - \delta)T_c]R_a[(\epsilon + \delta)T_c]$$

$$g(\delta) = \begin{cases} 1 - 2\delta, & (0 \leq \delta \leq 0.5) \\ 0, & (\delta > 0.5) \end{cases}$$

$$R_a(\tau) = \begin{cases} 1 - |\tau/T_c|, & (|\tau| \leq T_c) \\ 0, & (|\tau| > T_c) \end{cases} \quad (4)$$

The loop "S"-curve is shown in figure 2. It can be shown that its slope at the origin is  $4(1-\delta)$ . The second order DLL parameters: natural frequency, damping factor and equivalent noise bandwidth are, respectively:

$$\omega_n = \sqrt{\frac{4(1-\delta)KP}{\tau_1}} ; \quad \xi = \frac{\omega_n \tau_2}{2} ; \quad B_L = \frac{\omega_n}{2} \left[ \xi + \frac{1}{4\xi} \right] \quad (5)$$

where  $\tau_1$  and  $\tau_2$  are the time constants of the first order active loop filter. Using the linear model, which is valid for  $|\epsilon| \approx 0$ , is easy to show that the residual jitter in  $\epsilon$  is

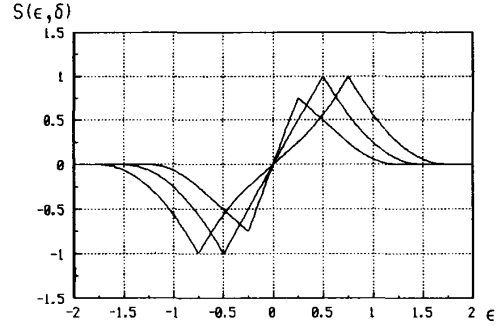


Figure 2: "S" curve for  $\delta = 0.25, 0.50$  and  $0.75$ .

given by:

$$\sigma_\epsilon^2 = \frac{B_L}{B} \frac{1}{4(1-\delta)^2\gamma} \left[ \lambda(0, \delta) + \frac{1-g^2(\delta)}{\gamma} \right] \quad (6)$$

## SIMULATION DESCRIPTION

Assuming an active first-order loop filter, with a transfer function given by:

$$F(s) = \frac{1}{s\tau_1} + \frac{\tau_2}{\tau_1} \quad (7)$$

the differential equation (2) can be written as:

$$\frac{d\epsilon(t)}{dt} = \frac{1}{T_c} \frac{d\tau(t)}{dt} - \frac{KP}{\tau_1} \left( \int_{-\infty}^t v(u) du + \tau_2 v(t) \right) \quad (8)$$

$$v(t) = S(\epsilon(t), \delta) + N(t)$$

In order to simulate by computer equation (8), we can envisage a recursive method to update  $\epsilon$  every  $T_0$  seconds, where  $T_0$  is an arbitrary time interval much shorter than  $B_L^{-1}$ . So, the main loop in the simulation program is equivalent to  $T_0$  seconds of real time. The recursive equation is:

$$\epsilon(k+1) = \epsilon(k) + T_0 \dot{\tau} - \frac{KPT_0}{\tau_1 T_0} \left[ \sum_{i=-\infty}^{k+1} v(i) + \frac{\tau_2}{T_0} v(k+1) \right] \quad (9)$$

$$v(k) = S(\epsilon(k), \delta) + N(k)$$

where  $k$  is the present update instant and  $\tau$  is the clock frequency offset.

## SIMULATION RESULTS

To check the validity of our program we have first obtained the simulated DLL behavior during the acquisition process. For example, taking  $\xi=0.7071$ ,  $\delta=0.5$  and  $\gamma=100\text{dB}$ , figure 3 shows some trajectories of the alignment error and its derivative in the phase plane.

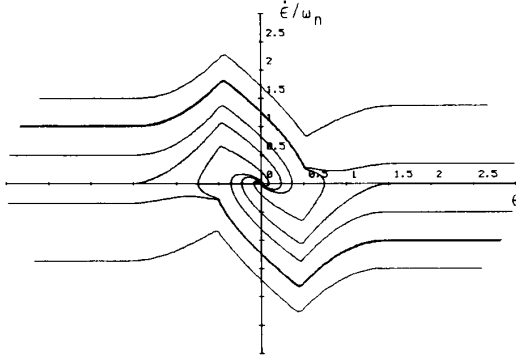


Figure 3: DLL acquisition in the phase plane

Similar plots have been obtained for different values of  $\delta$ . From these results we can state that in the presence of an initial clock frequency offset  $\tau$  and in a noiseless situation, the lock-in of the DLL is not guaranteed unless  $\tau/\omega_n \leq 2\delta$ . This is quite exact for  $\delta < 0.5$  and somewhat pessimistic for  $\delta > 0.5$ . As the initial clock frequency offset is mainly due to the VCO, we have that with an active loop filter and  $\xi=0.7071$  there is a lower bound for the noise bandwidth of the loop given by:

$$B_L \geq \frac{0.53 \cdot \Delta f_{VCO} \cdot R_c}{2\delta} = \frac{0.53 \cdot \Delta f_{VCO} \cdot B \cdot PG}{2\delta} \quad (10)$$

where  $\Delta f_{VCO}$  is the VCO stability and  $R_c = T_c^{-1}$  is the sequence chip rate. Notice that  $R_c$  can be expressed as  $B$  times the system processing gain ( $PG$ ). Now, taking (10) with equality and substituting  $B_L/B$  in (6) we get the expressions for the minimum jitter.

$$\sigma_\epsilon^2 = \begin{cases} \frac{0.53 \cdot \Delta f_{VCO} \cdot PG}{2\gamma} \left[ 1 + \frac{1}{\gamma(1-\delta)} \right] & (\delta \leq 0.5) \\ \frac{0.53 \cdot \Delta f_{VCO} \cdot PG}{2\gamma} \left[ \frac{1}{2\delta} + \frac{1}{4\gamma\delta(1-\delta)^2} \right] & (0.5 < \delta < 1) \end{cases} \quad (11)$$

where  $\lambda(0, \delta)$  and  $1-g^2(\delta)$  have been evaluated according to (4). Expression (11) gives the minimum achievable jitter unless some special action is taken to reduce the loop bandwidth once the DLL is locked. We have not considered this possibility. Figure 4 shows the jitter as a function of  $\gamma$  for different values of  $\delta$  and for  $\Delta f_{VCO} \cdot PG = 10^{-3}$ . Dashed curves are for  $\delta < 0.5$  and continuous line means  $\delta \geq 0.5$ . The step in  $\delta$  is 0.05.

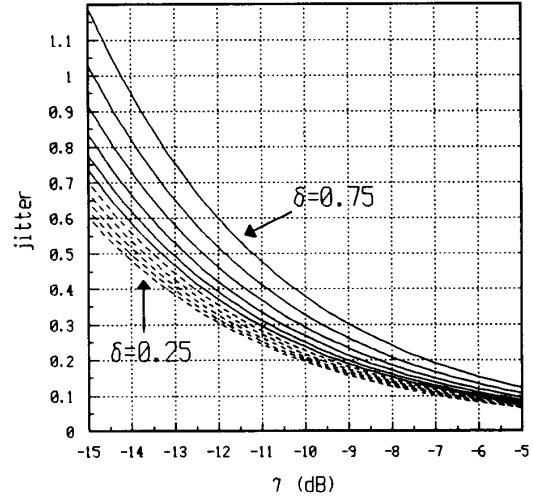


Figure 4: Jitter (in chips) for  $\Delta f_{VCO} \cdot PG = 10^{-3}$

As the loop "S"-curve is not periodic, due to the presence of noise the DLL will sooner or later lose lock. The Mean Time to Lose of Lock (MTLL) is a very important design parameter. We have obtained this magnitude, for the second order DLL, by computer simulation. To obtain the MTLL we run the program until  $|\epsilon|$  lies outside of the range of values for which the "S"-curve is not zero. This is considered as an out-of-lock condition and then  $\epsilon$  is reset to zero and the program is started again. When 100 out-of-lock situations have been counted, the MTLL is approximated by the total processed time (in terms of  $B_L^{-1}$ ) divided by 100. Taking  $\Delta f_{VCO} \cdot PG = 10^{-3}$  we need  $B_L/B = 5.3 \cdot 10^{-4}$  (for  $\delta=0.5$ ) or  $B_L/B = 3.533 \cdot 10^{-4}$  (for  $\delta=0.75$ ) in order to guarantee the lock of the DLL. Assuming these data in figure 5 we plot the logarithm of the obtained MTLL (in terms of  $B_L^{-1}$ ) versus the jitter. The square and the triangle marks are the points obtained by simulation. The curves presented correspond to an exponential regression that fits quite well to the simulation points. The regression equations are:

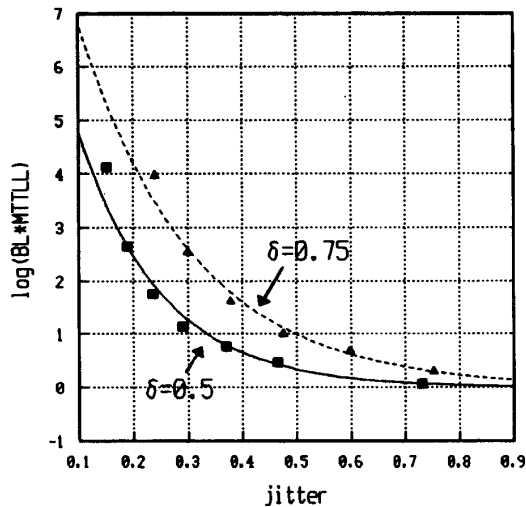


Figure 5: MTLL, in terms of  $B_L^{-1}$ , as a function of the jitter

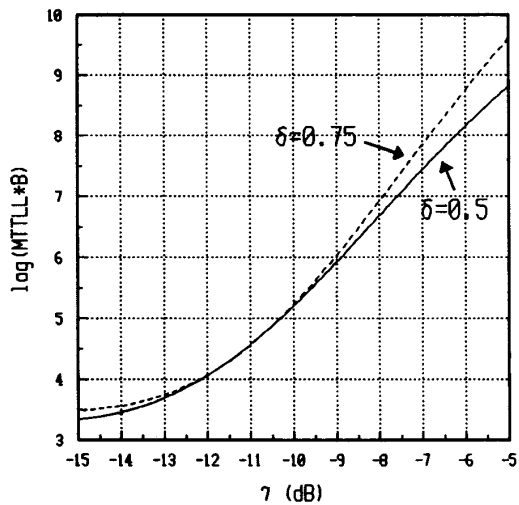


Figure 6: MTLL, in terms of  $B^1$ , as a function of the signal to noise ratio

$$\log(t_L B_L) = 9.34 \cdot (1.27 \cdot 10^{-3})^\gamma \quad (\delta=0.5) \quad (12)$$

$$\log(t_L B_L) = 11.02 \cdot (8.11 \cdot 10^{-3})^\gamma \quad (\delta=0.75)$$

Now, as  $B_L/B$  is known, combining (12) with the expressions of the jitter (11), we can plot the logarithm of the MTLL in bits as a function of  $\gamma$ . Figure 6 shows the results obtained. In figure 6 it can be noticed that the MTLL is very sensitive to the signal to noise ratio, since a change of one dB is sufficient to have the MTLL divided by ten. Another conclusion that can be derived from figure 6 is that the DLL with  $\delta=0.75$  has a better performance than the DLL with  $\delta=0.5$  only for  $\gamma > -10dB$ . This is due to the fact that, although for the same jitter the DLL with  $\delta=0.75$  has higher MTLL than the DLL with  $\delta=0.5$ , for the same signal to noise ratio it also has a higher residual jitter. The considered value of  $\Delta f_{VCO} \cdot PG = 10^{-3}$  may correspond, for example, to a system with an VCO stability of  $10^{-5}$  and 20dB of processing gain.

### CONCLUSIONS

A simulation program for the second order DLL has been developed. The program allows an optimum design of the loop low-pass filter in order to guarantee the right clock frequency acquisition and minimum residual jitter. The MTLL of the DLL has also been obtained by computer simulation for two different shifts between early and late codes. Results show a remarkable sensitivity to the signal to noise ratio in the data bandwidth.

### ACKNOWLEDGMENT

This work has been supported by ALCATEL SESA.

### REFERENCES

- [1] R.KARMY, B.Z.BOBROVSKY, Z.SCHUSS, "Loss of Lock Induced by Doppler or Code Rate Mismatch in Code Tracking Loops", MILCOM 1987.
- [2] A.POLYDOROS, C.L.WEBER, "Analysis and Optimization of Correlative Code-Tracking Loops in Spread-Spectrum Systems", IEEE Trans. on Comm. Vol COM-33, NO.1, January 1985.