All digital phase-locked loop: concepts, design and applications

Y.R. Shayan
T. Le-Ngoc

Indexing terms: Digital communication systems, Signal processing

Abstract: The concepts of an all digital phase-locked loop (DPLL), which contains a purely digital phase detector, loop filter and voltage-controlled oscillator, are explained. A second order DPLL is considered and analysed using the Z-transform technique. Implementation of the DPLL, based on the CMOS digital signal processor TMS 320C25, and the experimental results, are presented. Potential applications are also discussed.

1 Introduction

The progress in increasing the performance, speed, reliability, and the simultaneous reduction in size and cost of integrated circuits has resulted in strong interest being shown in the implementation of control and communication systems in the digital domain. Aside from the general advantages associated with digital systems, a digital version of the phase-locked loop solves some of the problems associated with its analogue counterpart; namely, sensitivity to DC drifts and component saturations, difficulties encountered in building higher order loops, and the need for initial calibration and periodic adjustments. In addition, with the ability to perform elaborate real-time processing on the signal samples, the DPLLs can be made more flexible and versatile, especially by the use of microprocessors.

The earliest efforts on DPLLs concentrated on partially replacing the analogue PLL (APLL) components with digital ones. The first all DPLL was reported by Drogin [1] in 1967. Since then, different authors [2–4] have suggested many kinds of all digital phase-locked loops and have discussed various aspects of implementing them. All DPLLs are categorised into two classes as follows:

(i) uniform sampling DPLL, in which the input signal is sampled at a fixed rate (Nyquist rate)

(ii) nonuniform sampling DPLL in which the sampling rate is variable.

In this paper, the concepts and design of a uniform sampling DPLL suitable for coherent (all digital) communications are presented. This DPLL contains a purely digital phase detector, loop filter and voltage-controlled oscillator, providing an extremely wide tracking frequency range and a completely linear behaviour. Experimental results and potential applications are also discussed.

Paper 6453F (E16), first received 10th March and in revised form 17th October 1988

The authors are with the Department of Electrical and Computer Engineering, Concordia University, 1455 de Maisonneuve Blvd., West Montreal, Quebec, Canada H3G 1M8

2 System description

The block diagram of the DPLL is given in Fig. 1. The A/D and D/A convertors are used to convert an analogue signal to a sampled, digitised form and vice-versa. The DPLL (on the right of these convertors) consists of three major functional units:

(*a*) phase detector (PD)
(*b*) digital loop filter
(*c*) voltage-controlled oscillator (VCO).

Other than these major units, there are two look-up tables to perform arcsine and sine functions. The arcsine

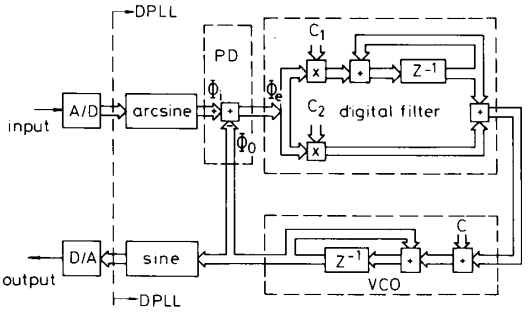

Fig. 1 *Second order DPLL*

function is used to find the phase of the input signal (Φ_i) and the sine function is used for constructing the output signal, knowing Φ_0. Note that for proper operation of the arcsine function, the input of the DPLL should have a fixed amplitude. The constant amplitude can be achieved by the use of an automatic gain control (AGC) at the input.

In this DPLL, after finding the phase of the input signal (Φ_i), the phase detector (PD) calculates the phase error (Φ_e). This phase error is filtered and used to control the instantaneous output phase of the VCO (Φ_0). Then for constructing the output sinusoidal signal, Φ_0 is applied to the sine function block. In Fig. 1, the coefficients C_1 and C_2 are parameters of the digital filter, and C is a constant value which determines the central frequency of the DPLL.

This second order DPLL is equivalent to an analogue PLL with linear phase detector which has a first order active filter as loop filter [5]. The advantage of this DPLL compared to other uniform sampling DPLLs [4] is in having a linear phase detector. The linear phase

detector results in a shorter acquisition time and larger lock-in range.

3 Analysis

3.1 Transfer function

Using the Z-transform technique, the transfer function of the proposed DPLL $H(Z)$ is

$$H(Z) = \frac{\Phi_0(Z)}{\phi_i(Z)} = \frac{C_2(Z-1) + C_1}{(Z-1)^2 + C_2(Z-1) + C_1} \tag{1}$$

To find the values of C_1 and C_2, the equivalent second order analogue system with transfer function $H(S)$ is considered [5], where

$$H(S) = \frac{2\eta\omega_n S + \omega_n^2}{S^2 + 2\eta\omega_n S + \omega_n^2} \tag{2}$$

The values of C_1 and C_2 can be found when the sampling frequency f_s is high compared to the natural frequency f_n of the loop ($\omega_n T \ll 1$) and

$$\begin{aligned} C_2 &= 2\eta\omega_n T \\ C_1 &= C_2^2/4\eta^2 \end{aligned} \tag{3}$$

where

$$T = \frac{1}{f_s}, \quad \omega_n = 2\pi f_n$$

and η is the damping factor.

3.2 Phase error response

The phase error response due to an input phase $\Phi_i(Z)$ is

$$\Phi_e(Z) = \frac{(Z-1)^2}{(Z-1)^2 + C_2(Z-1) + C_1}\Phi_i(Z) \tag{4}$$

Using the final value theorem it can be shown that the steady-state phase error is zero for both frequency ($\Delta\omega$) and phase step (θ) inputs. The transient response can be found by taking the inverse Z-transform of eqn. 5.

$$\begin{aligned} \Phi_e(Z) &= \frac{\theta Z(Z-1)}{(Z-1)^2 + C_2(Z-1) + C_1} \\ \Phi_e(Z) &= \frac{\Delta\omega T Z}{(Z-1)^2 + C_2(Z-1) + C_1} \end{aligned} \tag{5}$$

3.3 Tracking range

The loop DC gain K_v is given by

$$K_v = \lim_{z\to 1}\left[\frac{C_1 + C_2(Z-1)}{(Z-1)^2}\right] \to \infty \tag{6}$$

As the tracking range of DPLL is proportional to the loop DC gain [5] we can expect a very wide tracking range. The tracking range is only limited by $f_s/2$.

3.4 Stability

For a stable DPLL, the poles of $H(Z)$ must be inside the unit circle in the Z-plane, i.e.

$$\left|\frac{2 - C_2 \pm [(C_2-2)^2 - 4(C_1 - C_2 + 1)]^{1/2}}{2}\right| < 1 \tag{7}$$

After solving the above inequalities we can find the condition for a stable DPLL, which is

$$2C_2 - 4 < C_1 < C_2, \quad C_1 > 0 \tag{8}$$

It is shown in Reference 6 that because of quantisation errors, sometimes an oscillation can be seen at the output of a digital filter, even in the stable region of the coefficients of the filter. As the model in Fig. 1 is a second order recursive filter, we can expect the oscillation (limit cycle) for some values of C_1 and C_2. Using the results given in Reference 7, the region of limit cycles can be calculated for this DPLL. The triangle in Fig. 2 shows

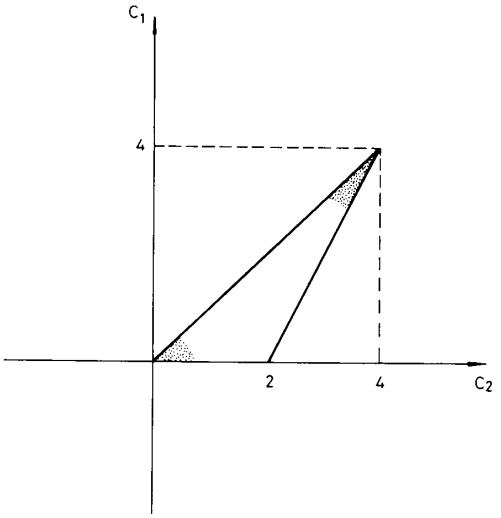

Fig. 2 *Stable region of the DPLL*

the stable region of the DPLL, and the dotted area represents the limit cycle region.

3.5 Noise bandwidth

Let us consider an additive white Gaussian noise present at the input of the DPLL. It is shown in Reference 8 that if the noise samples are stationary with zero mean, the steady state variance of Φ_e is

$$\sigma_\Phi^2 = \frac{1}{2\pi j}\oint_{|Z|=1} H(Z)H(Z^{-1})Z^{-1}\frac{R_n(Z)}{2P}\,dZ \tag{9}$$

where $R_n(Z)$ is the Z-transform of the noise autocorrelation function. In many practical cases, we can model the noise to be of uncorrelated zero mean with variance $\sigma_n^2 = N_0 B_i$ (B_i is the input noise bandwidth). The one-sided loop bandwidth B_L is defined to be

$$2B_L/B_i = \frac{1}{2\pi j}\oint_{|Z|=1} H(Z)H(Z^{-1})Z^{-1}\,dZ \tag{10}$$

This yields $\sigma_\Phi^2 = N_0 B_L/P$, which is identical to the analogue result. Note that P is the power of the input signal.

4 TMS320C25 based design and experimental results

The implementation of DPLL is achieved by using a CMOS digital signal processor, TMS320C25. The TMS320C25 is a 16-bit processor and can perform 16-bit addition, subtraction and 16-bit by 16-bit multiplication, each in a single cycle of 100 nanoseconds. It has 544 words of on-chip data RAM and 4K words of on-chip masked ROM [9]. These features make it possible to design the DPLL by using only a TMS320C25 chip and 4 SSI chips. The implemented second order DPLL needs 420 words of ROM to store the look-up tables and the program itself. The maximum sampling frequency of this DPLL implementation is 120 kHz.

To design the DPLL, coefficients of the filter are chosen by using eqn. 3 inside the stable region shown in

Fig. 2. They are then normalised to 16 bits such that the normalised values are in the range of 2^{14} to $2^{15} - 1$. To represent all the variables inside the loop, 16 bits are used (except for multiplication results). As the values of C_1 and C_2 are very small, double-precision multiplication, and a long register in the digital filter, are used to preserve their effect on the phase error. The length of this register is 48 bits. The constant C, which represents the central frequency of the DPLL, and the register inside the VCO are represented in 16 bits.

Based on the above discussion, a second order DPLL, with the following conditions, was tested:

(*a*) central frequency $f_0 = 4.8$ kHz
(*b*) sampling frequency $f_s = 8f_0$
(*c*) damping factor $\eta = 0.707$
(*d*) natural frequency $f_n = 10$ Hz

In the following paragraphs, real-time measurement results of this DPLL are compared with theoretical ones, and also with the characteristics of the analogue PLL.

In Fig. 3, the experimental transfer function of the DPLL is given. This result shows that the experimental

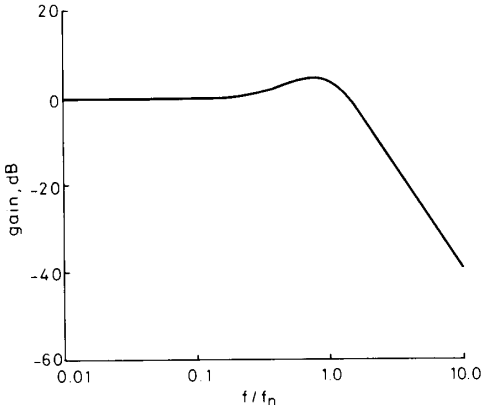

Fig. 3 *Transfer function of the DPLL*

transfer function of the DPLL agrees with the theoretical one. This agreement is due to the high sampling frequency of $f_s = 8f_0 = 38.6$ kHz.

The steady-state phase error is measured to be almost zero. This amount is due to quantisation, which decreases by increasing the number of bits n representing Φ_i. The steady-state phase error is $2\pi/2^n$. Fig. 4 represents the phase error transient response, with an input frequency step of 87 Hz when the loop was initially out of lock. This Figure shows four skipping of cycles. Owing to quantisation, the acquisition time for this case is about 10% longer than the theoretical value.

In Fig. 5, the measured acquisition time of the DPLL is compared to the acquisition time of the analogue counterpart. This shows a very fast acquisition time for the DPLL, even with the quantisation effect discussed above.

For measuring the hold-in range, the input frequency was set at 4.8 kHz (central frequency), and then it was continuously varied until the loop went out-of-lock. The measured hold-in range was up to 19.3 kHz ($f_s/2$), in agreement with the analytical results, and is much larger than the hold-in range of analogue PLLs.

To find the lock-in range, the DPLL was initially brought to an out-of-lock condition, and then an input signal with a frequency step was applied to the DPLL. From the measurement results, no cycle skips occurred when the frequency step was up to 65 Hz, and so the lock-in range is 65 Hz, which is about 4.5 times that of an analogue PLL. When there was skipping of cycles, the

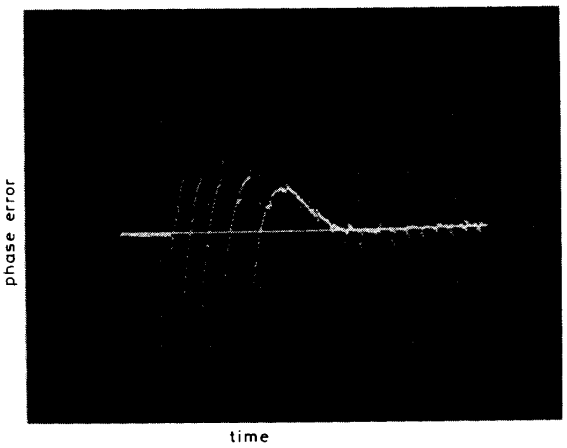

Fig. 4 *Transient response of the DPLL*
Horizontal axis: 40 ms/division
Vertical axis: 90 degrees/division

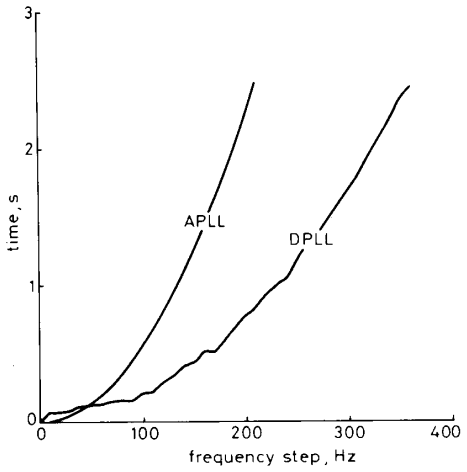

Fig. 5 *Acquisition behaviour*

pull-in range was measured to be about 405 Hz, which is much higher than the case of an analogue PLL.

The noise behaviour of the DPLL is measured in the presence of additive white Gaussian noise at the input, and the root-mean-square (RMS) phase jitter at the output of the system is found. Fig. 6 shows the noise behaviour of DPLLs and APLLs, respectively.

The results in Fig. 6 indicate that for a high input signal-to-noise ratio (SNR), the digital PLL outperforms the analogue PLL. This effect can be explained by considering the limiting effect of an A/D convertor. In practice, this effect is also present in APLLs because of a hard limiter in front of the phase detector.

5 Conclusions and applications

Analytical derivations of a second order DPLL have been presented using the Z-transform technique. Implementation of the DPLL based on the CMOS digital signal processor TMS320C25, and the corresponding

experimental results have been discussed. The advantages of a digital PLL over the analogue counterpart can be summarised as follows:

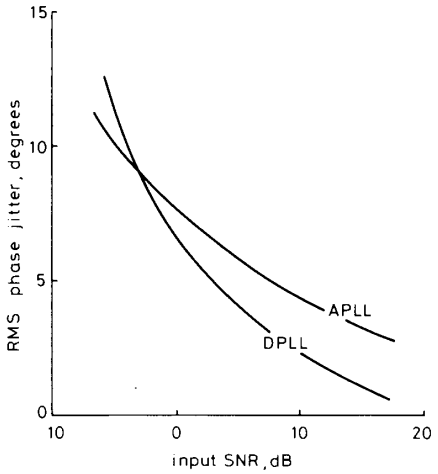

Fig. 6 *Noise behaviour*

(i) The steady-state phase error, which is determined by the number of bits used for computations, is very small.

(ii) The hold-in range extends from DC to $f_s/2$.

(iii) It has a large pull-in range.

(iv) The linear phase detector yields a shorter acquisition time and larger lock-in range.

Owing to the flexibility and high performance of digital systems, modems are mainly designed by a digital approach. The carrier recovery in digital modems is achieved by phase-locked loops, and as the DPLL is purely digital, it can be used in these systems.

As fast acquisition is one of the important factors in the design of frequency synthesisers for frequency hopping, this DPLL can be used in mobile communications.

In the digital PLL it is possible to change the central frequency simply by changing the coefficient C in Fig. 1. We can therefore use this DPLL in frequency synthesiser applications and change C by software in microprocessor-based implementations.

6 References

1 DROGIN, E.M.: 'Steering on course to safer air travel', *Electronics*, November 1967, pp. 95–102

2 LINDSEY, W.C., and CHIE, C.M.: 'A survey of digital phase-locked loops', *Proc. IEEE*, 1981, **69**, (4), pp. 410–431

3 TROHA, D.G.: 'Digital phase-locked loop design using SN54/74LS297'. Application report, Texas Instruments, 1988

4 GARODNICK, J., GRECO, J., and SCHILLING, D.L.: 'Response of an all digital phase-locked loop', *IEEE Trans.*, 1974, **COM-22**, pp. 751–764

5 GARDNER, F.M.: 'Phase lock techniques' (John Wiley & Sons, New York, 1966)

6 RABINER, L.R., and GOLD, B.: 'Theory and application of digital signal processing' (Bell Telephone Labs., New York, 1975), Chap. 5

7 MECKLENBRAUKER, W.F.G., and PEEK, J.B.H.: 'Remarks on the zero-input behaviour of second order digital filters designed with one magnitude truncation quantiser', *IEEE Trans.*, April 1975, ASSP

8 OPPENHEIM, A.V., and SCHAFER, R.W.: 'Digital signal processing' (Prentice-Hall, Englewood Cliffs, NJ, 1975)

9 TMS320C25 User's Guide, Texas Instruments, 1986