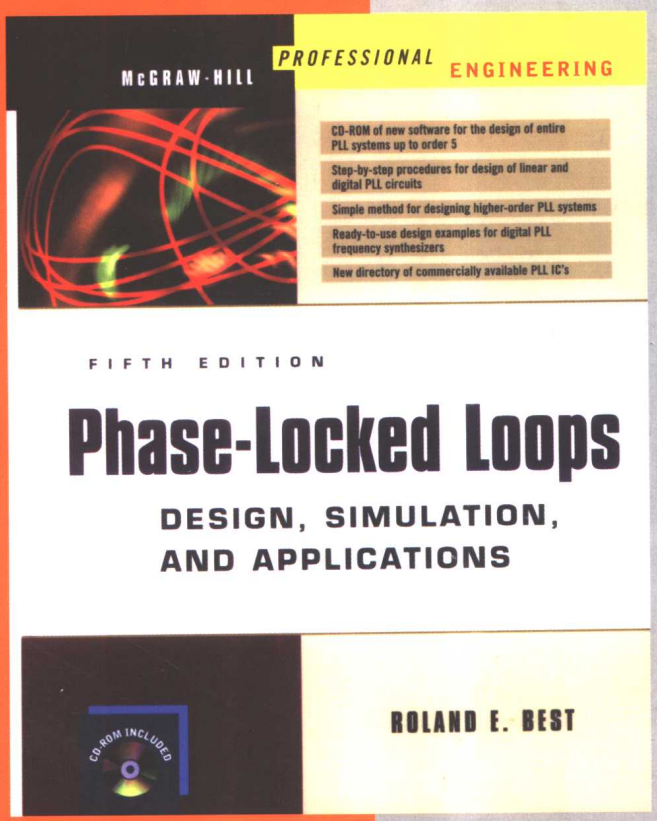
Education

国外大学优秀教材 — 微电子类系列（影印版）

Roland E. Best

锁相环设计、仿真与应用



清华大学出版社

国外大学优秀教材 —— 微电子类系列（影印版）

锁相环设计、仿真与应用

Phase-Locked Loops Design, Simulation, and Applications

Roland E. Best

清华大学出版社
北京

Roland E. Best
Phase-Locked Loops Design, Simulation, and Applications (5th Edition)
EISBN: 0-07-141201-8

北京市版权局著作权合同登记号　图字：01-2003-7773

图书在版编目(CIP)数据

锁相环设计、仿真和应用 = Phase-Locked Loops Design, Simulation, and Applications / 贝斯特（Best, R. E.）著. —影印本. —北京：清华大学出版社，2003
（国外大学优秀教材——微电子类系列）
ISBN 7-302-07729-0

Ⅰ. 锁…　Ⅱ. 贝…　Ⅲ. 锁相技术－高等学校－教材－英文　Ⅳ. TN911.8

中国版本图书馆 CIP 数据核字（2003）第 112312 号

出 版 者：清华大学出版社　　**地　　址**：北京清华大学学研大厦
http://www.tup.com.cn　　**邮　　编**：100084
社总机：（010）6277 0175　　**客户服务**：（010）6277 6969
责任编辑：田志明
印 刷 者：北京牛山世兴印刷厂
装 订 者：三河市新茂装订有限公司
发 行 者：新华书店总店北京发行所
开　　本：185×230　**印张**：27.25
版　　次：2003 年 12 月第 1 版　2003 年 12 月第 1 次印刷
书　　号：ISBN 7-302-07729-0/TN・157
印　　数：1~3000
定　　价：68.00 元（含光盘）

出 版 前 言

微电子技术是信息科学技术的核心技术之一，微电子产业是当代高新技术产业群的核心和维护国家主权、保障国家安全的战略性产业。我国在《信息产业“十五”计划纲要》中明确提出：坚持自主发展，增强创新能力和核心竞争力，掌握以集成电路和软件技术为重点的信息产业的核心技术，提高具有自主知识产权产品的比重。发展集成电路技术的关键之一是培养具有国际竞争力的专业人才。

微电子技术发展迅速，内容更新快，而我国微电子专业图书数量少，且内容和体系不能反映科技发展的水平，不能满足培养人才的需求，为此，我们系统挑选了一批国外经典教材和前沿著作，组织分批出版。图书选择的几个基本原则是：在本领域内广泛采用，有很大影响力；内容反映科技的最新发展，所述内容是本领域的研究热点；编写和体系与国内现有图书差别较大，能对我国微电子教育改革有所启示。本套丛书还侧重于微电子技术的实用性，选取了一批集成电路设计方面的工程技术用书，使读者能方便地应用于实践。本套丛书不仅能作为相关课程的教科书和教学参考书，也可作为工程技术人员的自学读物。

我们真诚地希望，这套丛书能对国内高校师生、工程技术人员以及科研人员的学习和工作有所帮助，对推动我国集成电路的发展有所促进。也衷心期望着广大读者对我们一如既往的关怀和支持，鼓励我们出版更多、更好的图书。

清华大学出版社
理工出版事业部
2003.9

Phase-Locked Loops

Design, Simulation, and Applications

影 印 版 序

Roland E. Best 的这本经典参考书《锁相环设计、仿真与应用》（Phase-Locked Loops Design, Simulation, and Applications）第 5 版终于面世了。作者从事锁相环设计数十年，具有丰富的理论和实际经验。本书系统的讲述了基本的锁相环理论和电路模块和锁相环系统结构以及详细的设计过程。包括锁相环的概况，混合信号锁相环，锁相环频率综合器，高阶锁相环路，混合信号频率综合器的计算机辅助设计和模拟，全数字锁相环，全数字锁相环的计算机辅助设计和模拟，软件锁相环，锁相环在通信中的应用，最先进的商业化集成锁相环，锁相环参数的测试等，涉及到控制理论，通信理论，信号处理理论等多个学科的内容。

第 5 版的特色是从实际应用的角度在锁相环系统的层次上解释理论方面的问题，给出了锁相环系统在实际设计中的具体过程和步骤，并且提供了作者自己开发的，在行为级对锁相环系统的性能进行仿真的软件程序。

本书是进行锁相环系统设计的入门教材，可以作为本科生和研究生的课程教材或参考书，也可以作为从事锁相环电路设计的工程师和技术管理人员的参考手册。对于已经具有一定经验的工程师和研究人员，也可以从本书中学习到一些全面的和最新的有关锁相环系统的知识。

李永明

2003 年 11 月

于清华大学

Preface to the Fifth Edition

Historically the PLL has been a linear circuit. While the first PLLs were realized with discrete components, they became available as ICs in about 1965. The first of these were linear devices (LPLLs), built in semiconductor technologies similar to the operational amplifiers of that era. A few years later (about 1970) the first "digital" PLLs (DPLLs) became available, but when looking at their schematics, we recognize that only the phase detector was built from logic circuits, while the remaining parts (voltage-controlled oscillator (VCO), loop filter) stayed analog; hence these PLLs must be considered hybrid systems. In this new edition of the book we combined the two categories LPLL and DPLL into one single class named "mixed-signal PLL." Doing so greatly simplifies the analysis, because both classes now can be treated by a unified theory.

Chapter 1 is a short introduction into the PLL domain, and Chap. 2 deals with theory, design, and applications of mixed-signal PLLs. The discussion includes different types of phase detectors (linear and digital), phase-frequency detectors with charge-pump outputs, loop filters (active and passive), and VCOs. Typical mixed-signal PLL applications are given, such as circuits for retiming and clock recovery, motor speed control, and the like.

Because frequency synthesis is one of the most important applications of DPLLs, a separate chapter (3) is devoted to digital PLL frequency synthesizers. Because phase jitter and spurious sidebands are the most disturbing phenomena with frequency synthesizers, different methods are presented to overcome those problems, e.g., antibacklash circuits and higher-order loop filters. Moreover, the analysis covers both integer-N and fractional-N synthesizers and demonstrates that the latter can acquire "lock" much faster, a feature that is used with great benefit in frequency-hopping (spread-spectrum) applications. Spread-spectrum techniques will become increasingly important in the newest generations of mobile phones. It is demonstrated furthermore that simple synthesizers can be realized as single loops, but that multiloop configurations become mandatory in high-performance systems.

Because higher-order systems (filters) are a must in many synthesizer applications, Chap. 4 deals with the design of such systems, i.e., of PLLs up to fifth

order. In designing higher-order loops, the placement of poles and zeros can be a difficult task. The design is greatly facilitated by a novel method developed by the author, based on simple Bode diagrams. To ease system realization even further, a program developed by the author is included on a CD-ROM, which automatically designs and analyses PLLs up to fifth order; this topic is discussed in Chap. 5, which also includes many design examples. Having synthesized a PLL circuit, the program can be used to simulate dynamic performance of the system, e.g., lock-in and lock-out processes. To investigate PLL performance in the presence of noise—which is the normal scenario in practice—the user can superimpose narrowband or broadband noise of any desired level. Finally, the program displays Bode diagrams of the synthesized PLL and loop filter schematics, including the component values.

Chapter 6 treats theory, design and applications of the "all-digital" PLL (ADPLL), a PLL category introduced later than the preceding ones. In contrast to LPLLs and DPLLs, which are continuous-time systems, the ADPLL is a discrete-time device and therefore exhibits relatively large ripple (phase jitter). The applications of the ADPLL are therefore restricted to those cases where ripple can be tolerated, e.g., frequency-shift keying (FSK) decoders and the like. Chapter 7 describes computer-aided design and simulation of ADPLLs, using the already mentioned computer program.

Because the speed of microcontrollers and digital signal processors increased dramatically in recent years, many PLL applications can now be implemented by software. Chapter 8 discusses the hardware/software tradeoff in the domain of the PLL and describes a number of software algorithms that can be used to implement software PLLs (SPLLs).

In Chap. 9 a review of PLL applications in the field of communications is given. It includes the most important digital modulation schemes such as BSK, QPSK, FSK, and QAM and describes a number of special PLL circuits used for carrier and symbol synchronization (e.g., Costas loop, early-late gate, integrate-and-dump circuit) and measures to prevent intersymbol interference (ISI), e.g., root-raised-cosine filters. Among other topics, this chapter also explains how to increase the symbol rate of digital communications without increasing system bandwidth.

Chapter 10 contains a list of PLL ICs currently available from American, European, and Japanese manufacturers, including short descriptions of the circuits. The list includes full PLL systems on a chip, parts of PLLs such as phase detectors and VCOs, and complex systems based on PLL-like frequency synthesizers or radio/TV chips. It also includes many single- and dual-modulus prescalers.

Finally, Chap. 11 demonstrates how the parameters of PLLs can be measured with conventional lab instruments such as oscilloscopes and signal generators.

Three appendixes provide additional information on selected topics. In App. A the analysis of the acquisition process is developed for the mathematically interested reader. Appendix B is a primer on the Laplace transform, which is

frequently used in this book, and App. C is an overview on digital filtering, which has become an increasingly important issue in the design of high-complexity PLL systems.

Roland Best

Contents

Chapter

1

Introduction to PLLs

1.1 Operating Principles of the PLL

The phase-locked loop (PLL) helps keep parts of our world orderly. If we turn on a television set, a PLL will keep heads at the top of the screen and feet at the bottom. In color television, another PLL makes sure that green remains green and red remains red (even if the politicians claim that the reverse is true).

A PLL is a circuit that causes a particular system to track with another one. More precisely, a PLL is a circuit synchronizing an output signal (generated by an oscillator) with a reference or input signal in frequency as well as in phase. In the synchronized—often called *locked*—state the phase error between the oscillator's output signal and the reference signal is zero, or remains constant.

If a phase error builds up, a control mechanism acts on the oscillator in such a way that the phase error is again reduced to a minimum. In such a control system the phase of the output signal is actually *locked* to the phase of the reference signal. This is why it is referred to as a *phase-locked loop*.

The operating principle of the PLL is explained by the example of the linear PLL (LPLL). As will be pointed out in Sec. 1.2, there exist other types of PLLs, e.g., digital PLLs (DPLLs), all-digital PLLs (ADPLLs), and software PLLs (SPLLs). Its block diagram is shown in Fig. 1.1*a*. The PLL consists of three basic functional blocks:

1. A voltage-controlled oscillator (VCO)
2. A phase detector (PD)
3. A loop filter (LF)

In this simple example, there is no down-scaler between output of VCO [$u_2(t)$] and lower input of the phase detector [ω_2]. Systems using down-scalers are discussed in the following chapters.

In some PLL circuits a current-controlled oscillator (CCO) is used instead of the VCO. In this case the output signal of the phase detector is a controlled

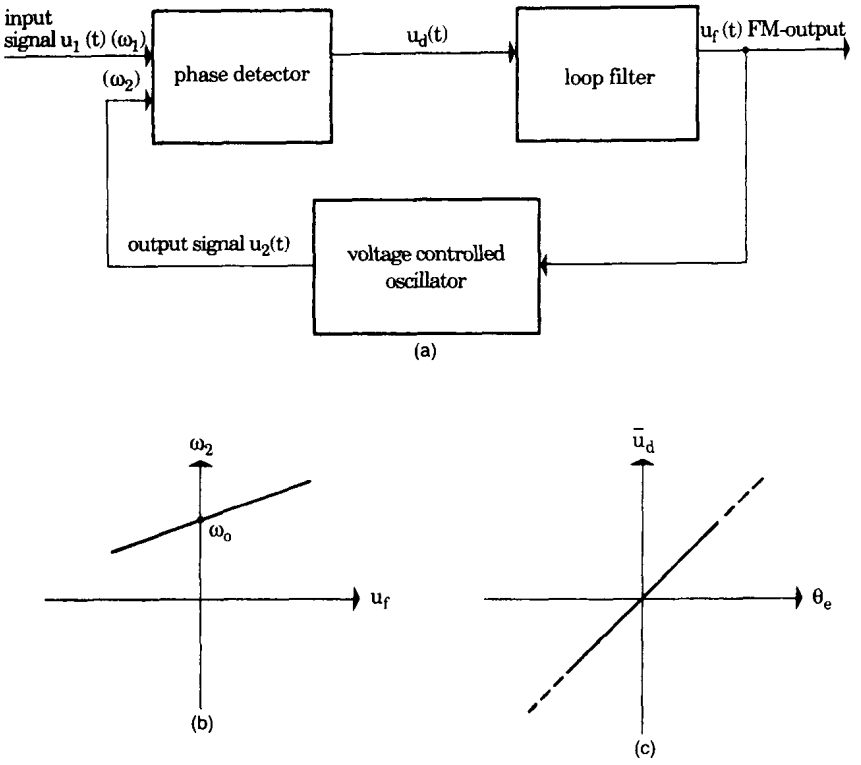


Figure 1.1 (*a*) Block diagram of the PLL. (*b*) Transfer function of the VCO. (u_f = control voltage; ω_2 = angular frequency of the output signal.) (*c*) Transfer function of the PD. ($\overline{u_d}$ = average value of the phase-detector output signal; θ_e = phase error.)

current source rather than a voltage source. However, the operating principle remains the same. The signals of interest within the PLL circuit are defined as follows:

- The reference (or input) signal $u_1(t)$
- The angular frequency ω_1 of the reference signal
- The output signal $u_2(t)$ of the VCO
- The angular frequency ω_2 of the output signal
- The output signal $u_d(t)$ of the phase detector
- The output signal $u_f(t)$ of the loop filter
- The phase error θ_e, defined as the phase difference between signals $u_1(t)$ and $u_2(t)$

Let us now look at the operation of the three functional blocks in Fig. 1.1*a*. The VCO oscillates at an angular frequency ω_2, which is determined by the output signal u_f of the loop filter. The angular frequency ω_2 is given by

$$\omega_2(t) = \omega_0 + K_0 u_f(t) \tag{1.1}$$

where ω_0 is the center (angular) frequency of the VCO and K_0 is the VCO gain in rad $\cdot$ s^{-1} $\cdot$ V^{-1}.

Equation (1.1) is plotted in Fig. 1.1*b*. Because the rad (radian) is a dimensionless quantity, we will drop it mostly in this text. (Note, however, that any phase variables used in this book will have to be measured in radians and not in degrees!) Therefore, in the equations a phase shift of 180° must always be specified as a value of π.

The PD—also referred to as phase comparator—compares the phase of the output signal with the phase of the reference signal and develops an output signal $u_d(t)$ that is approximately proportional to the phase error θ_e, at least within a limited range of the latter:

$$u_d(t) = K_d \theta_e \tag{1.2}$$

Here K_d represents the gain of the PD. The physical unit of K_d is volts per radian. Figure 1.1*c* is a graphical representation of Eq. (1.2).

The output signal $u_d(t)$ of the PD consists of a dc component and a superimposed ac component. The latter is undesired; hence it is canceled by the loop filter. In most cases a first-order, low-pass filter is used. Let us now see how the three building blocks work together. First we assume that the angular frequency of the input signal $u_1(t)$ is equal to the center frequency ω_0. The VCO then operates at its center frequency ω_0. As we see, the phase error θ_e is zero. If θ_e is zero, the output signal u_d of the PD must also be zero. Consequently the output signal of the loop filter u_f will also be zero. This is the condition that permits the VCO to operate at its center frequency.

If the phase error θ_e were not zero initially, the PD would develop a nonzero output signal u_d. After some delay the loop filter would also produce a finite signal u_f. This would cause the VCO to change its operating frequency in such a way that the phase error finally vanishes.

Assume now that the frequency of the input signal is changed suddenly at time t_0 by the amount $\Delta\omega$. As shown in Fig. 1.2, the phase of the input signal then starts leading the phase of the output signal. A phase error is built up and increases with time. The PD develops a signal $u_d(t)$, which also increases with time. With a delay given by the loop filter, $u_f(t)$ will also rise. This causes the VCO to increase its frequency. The phase error becomes smaller now, and after some settling time the VCO will oscillate at a frequency that is exactly the frequency of the input signal. Depending on the type of loop filter used, the final phase error will have been reduced to zero or to a finite value.

The VCO now operates at a frequency that is greater than its center frequency ω_0 by an amount $\Delta\omega$. This will force the signal $u_f(t)$ to settle at a final value of $u_f = \Delta\omega/K_0$. If the center frequency of the input signal is frequency modulated by an arbitrary low-frequency signal, then the output signal of the loop filter is the demodulated signal. The PLL can consequently be used as an

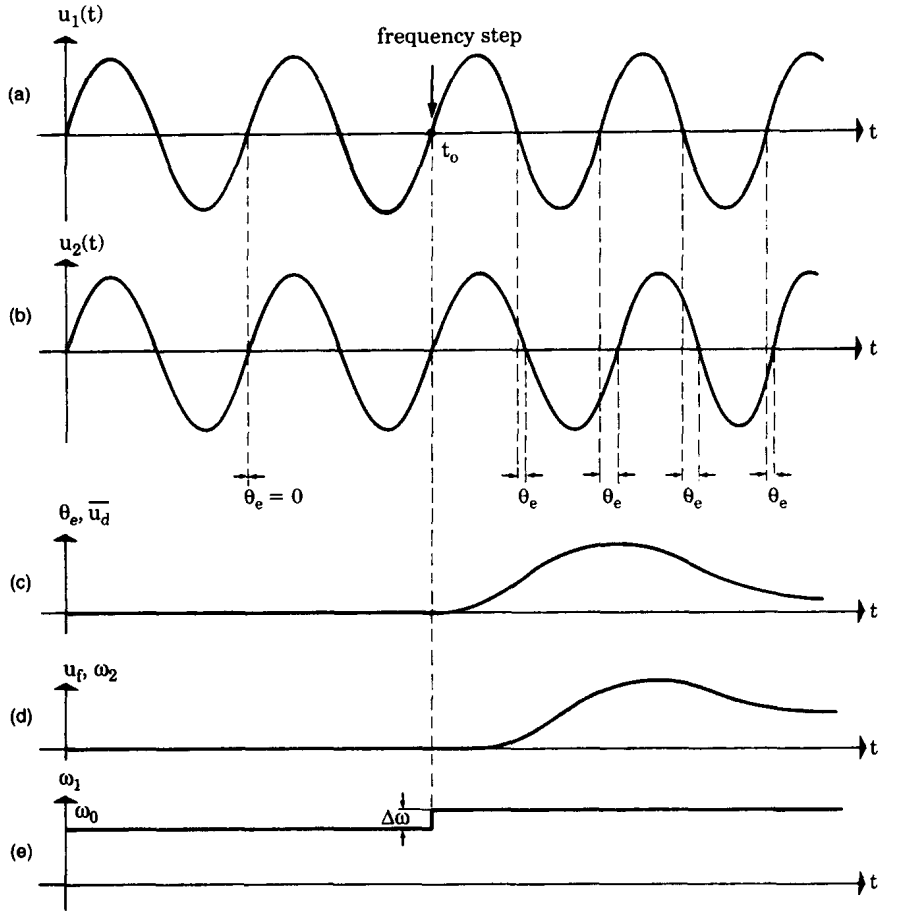


Figure 1.2 Transient response of a PLL onto a step variation of the reference frequency. (*a*) Reference signal $u_1(t)$. (*b*) Output signal $u_2(t)$ of the VCO. (*c*) Signals $\overline{u_d}(t)$ and $\theta_e(t)$ as a function of time. (*d*) Angular frequency ω_2 of the VCO as a function of time. (*e*) Angular frequency ω_1 of the reference signal $u_1(t)$.

FM detector. As we shall see later, it can be further applied as an AM or PM detector.

One of the most intriguing capabilities of the PLL is its ability to suppress noise superimposed on its input signal. Let us suppose that the input signal of the PLL is buried in noise. The PD tries to measure the phase error between input and output signals. The noise at the input causes the zero crossings of the input signal $u_1(t)$ to be advanced or delayed in a stochastic manner. This causes the PD output signal $u_d(t)$ to jitter around an average value. If the corner frequency of the loop filter is low enough, almost no noise will be noticeable in the signal $u_f(t)$, and the VCO will operate in such a way that the phase of the signal $u_2(t)$ is equal to the average phase of the input signal $u_1(t)$. Therefore, we can state that the PLL is able to detect a signal that is buried in noise. These

simplified considerations have shown that the PLL is nothing but a servo system that controls the phase of the output signal $u_2(t)$.

As shown in Fig. 1.2, the PLL was always able to track the phase of the output signal to the phase of the reference signal; this system was locked at all times. This is not necessarily the case, however, because a larger frequency step applied to the input signal could cause the system to "unlock." The control mechanism inherent in the PLL will then try to become locked again, but will the system indeed lock again? We shall deal with this problem in the following chapters. Basically two kinds of problems have to be considered:

- The PLL is initially locked. Under what conditions will the PLL remain locked?
- The PLL is initially unlocked. Under what conditions will the PLL become locked?

If we try to answer these questions, we notice that different PLLs behave quite differently in this regard. We find that there are some fundamentally different types of PLLs. Therefore, we first identify these various types.

1.2 Classification of PLL Types

The very first phase-locked loops (PLLs) were implemented as early as 1932 by de Bellescize[22]; this French engineer is considered the inventor of "coherent communication." The PLL found broader industrial applications only when it became available as an integrated circuit. The first PLL ICs appeared around 1965 and were purely analog devices. An analog multiplier (four-quadrant multiplier) was used as the phase detector, the loop filter was built from a passive or active *RC* filter, and the well-known voltage-controlled oscillator (VCO) was used to generate the output signal of the PLL. This type of PLL is referred to as the "linear PLL" (LPLL) today. In the following years the PLL drifted slowly but steadily into digital territory. The very first digital PLL (DPLL), which appeared around 1970, was in effect a hybrid device: only the phase detector was built from a digital circuit, e.g., from an EXOR gate or a JK-flipflop, but the remaining blocks were still analog. A few years later, the "all-digital" PLL (ADPLL) was invented. The ADPLL is exclusively built from digital function blocks and hence doesn't contain any passive components like resistors and capacitors.

Analogous to filters, PLLs can also be implemented by software. In this case, the function of the PLL is no longer performed by a piece of specialized hardware, but rather by a computer program. This last type of PLL is referred to as SPLL.

Different types of PLLs behave differently, so there is no common theory that covers all kinds of PLLs. The performance of LPLLs and DPLLs is similar, however; hence we can develop a theory that is valid for both categories. We will deal with LPLLs and DPLLs in Chap. 2, "Mixed-Signal PLLs." The term