

通过在LTspice中应用拉普拉斯变换对传递函数进行建模

Joseph Spencer, 现场应用工程师

摘要

传递函数用于电子系统的设计，例如滤波器、电源和其他控制系统。传递函数的频域分析涉及到拉普拉斯变换。本文探讨LTspice®中的传递函数实现方案，将理想的响应与建模的实现方案进行比较，并提供多个有用示例。有一些文件可以帮助理解和实践本主题。

简介

当我们想要分析一个系统的输出如何随输入变化时，会使用传递函数。这个系统可以是电阻分压器的输出电压，也可以是您在踩油门时汽车的速度。为了分析系统，我们需要了解，与频率范围内的输入相比较，其输出振幅和相位如何。为此，我们使用拉普拉斯变换。

$$F(s) = \int_0^{\infty} f(t)e^{-st} dt$$

目前有许多资源对拉普拉斯变换的数学逻辑和直观性都展开了探讨。为了达到我们的目的，我们可以把它看作一个数学工具，它让我们能够操作增益、频率、相位和系统的指数响应。

图1显示了传递函数的基本思路。

$$\text{In} \xrightarrow{H(s)} \text{Out} \quad \frac{\text{Out}}{\text{In}} = H(s)$$

图1. 传递函数。

图2显示了两个不同的传递函数。电阻分压器可以简单表述为：

$$H(s) = \frac{1}{2} \quad (1)$$

RC电路则使用稍复杂一些的公式2进行表述：

$$H(s) = \frac{1}{sC} = \frac{1}{R + \frac{1}{sC}} = \frac{1}{1 + sRC} = \frac{1}{1 + \frac{s}{\omega_p}} \quad (2)$$

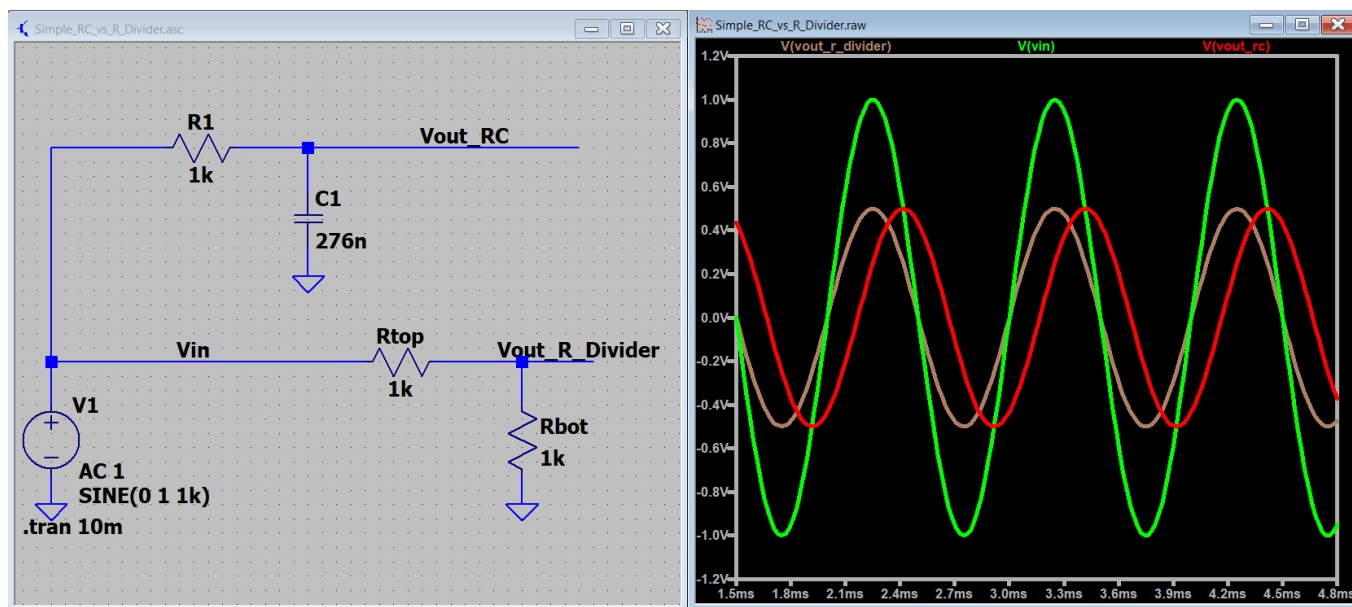


图2. 电阻分压器和RC电路都将振幅降低一半，其中 $f=1\text{kHz}$ 。RC电路还会导致延迟。

通过用这种形式表述传递函数，我们可以从极点与零点角度来讨论。这里，在 $\omega_0 = 1/RC$ 对应的位置，有一个极点。虽然RC电路（红色曲线）的1 kHz输出与电阻分压器的振幅相同，但其相位延迟较为显著。通过使用拉普拉斯变换，我们可以表述RC电路的增益和相位如何随频率变化。可以参考示例文件 [Simple_RC_vs_R_Divider.asc](#)。¹

LTspice中的拉普拉斯变换语法

要在LTspice中实现拉普拉斯变换，首先要在示意图中加入一个与电压相关的电压源。其对话框如图3所示。

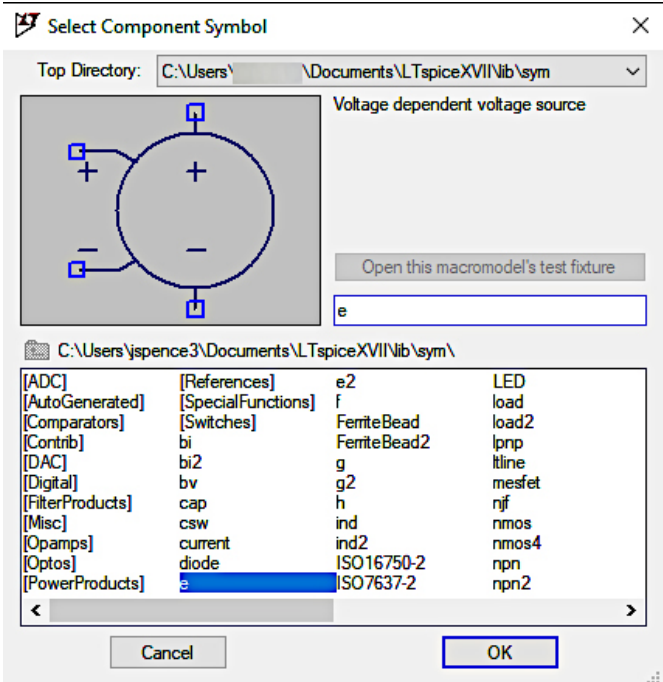


图3. 加入一个与电压相关的电压源。

右键点击电压源元素，打开**元件属性编辑器**。在**值**字段中，输入“Laplace =”，后接您的公式，其中应包含复变量“s”。

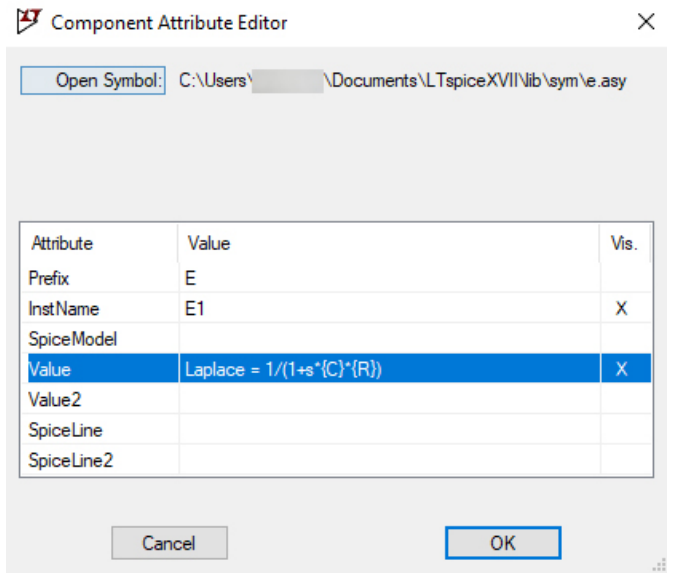


图4. 在**值**字段中输入公式。

在图4中，{C}和{R}包含在大括号中，所以我们可以使用.step spice命令来重复使用某些值。

图5仿真显示一个RC阻抗分压器的阶跃响应，以及使用拉普拉斯变换表述的对等传递函数。运行仿真，²我们可以看到，两个输出是匹配的。

在传递函数增大的过程中，我发现使用代码编辑器非常有用，它可以自动高亮显示匹配的括号。在文本编辑器中输入公式，然后将其复制到LTspice对话框中。参见图6。

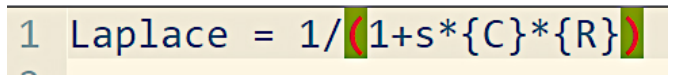


图6. 利用文本编辑器的语法高亮功能。

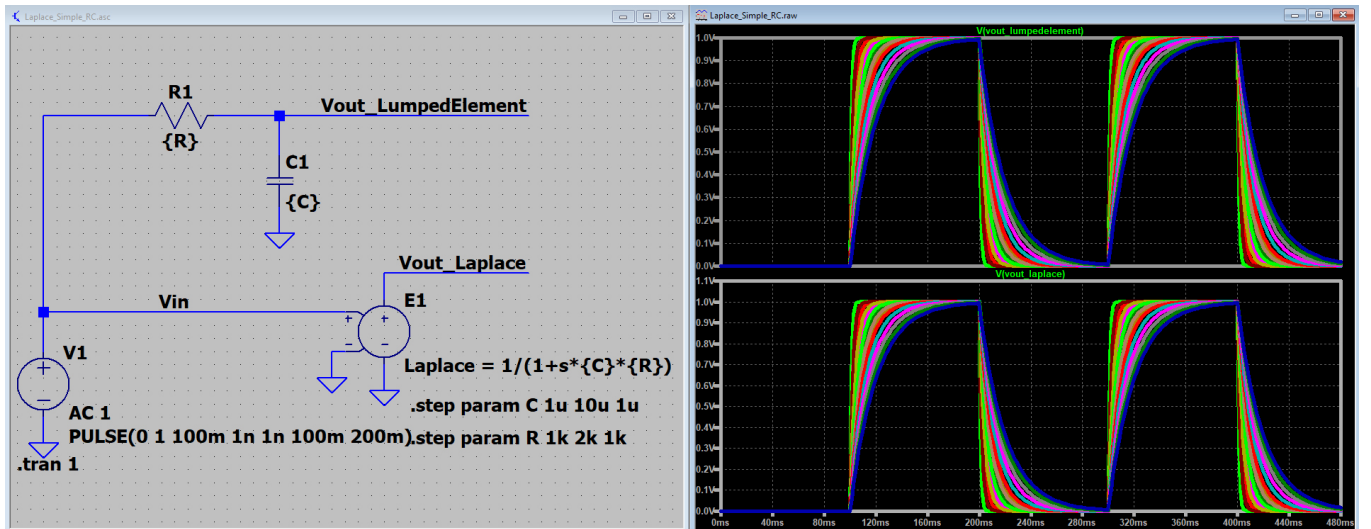


图5. 一个简单的RC电路，以及对等的拉普拉斯变换。

关于极点和零点的回顾

公式3给出了一个传递函数示例。

$$H(s) = \frac{\omega_{p0}}{s} \times \frac{1 + \frac{s}{\omega_z}}{1 + \frac{s}{\omega_{p1}}} \quad (3)$$

拆解公式3的各个部分，我们首先会看到 ω_{p0}/s 项。这表示一个原极点。原极点与积分器同义，用以在低频率下提供极高增益，从而消除控制系统中的稳态误差。原极点产生恒定的 90° 相位延迟，以及 -20 dB/十倍频程的增益响应。 ω_{p0} 项表示增益因为原极点滚降至 0 dB时的频率。

求分子和分母的根可以分别得出零点和极点频率。注意极点/零点频率是由这些根的幅度（即绝对值）给出的。零点导致产生 90° 相位增幅，且其频率周围的增益增大。极点导致产生 90° 相位延迟，且其频率周围的增益降低。

如果将示例传递函数表述为：

$$H(s) = \frac{s}{\omega_{z0}} \times \frac{1 + \frac{s}{\omega_z}}{1 + \frac{s}{\omega_{p1}}} \quad (4)$$

打头的“s”项现在表示原点处的零点。原点处的零点是纯微分器的代表。它具有恒定的 90° 相位增幅，增益随频率增大，在频率为 ω_{z0} 时达到 0 dB增益。

在图7所示为这些不同极点和零点的实现。每个实现方案的频率都为 1 kHz。得出的每个实现方案的增益和相位如图8所示。可以参考示例文件[PoleZeroExamples.asc](#)。³

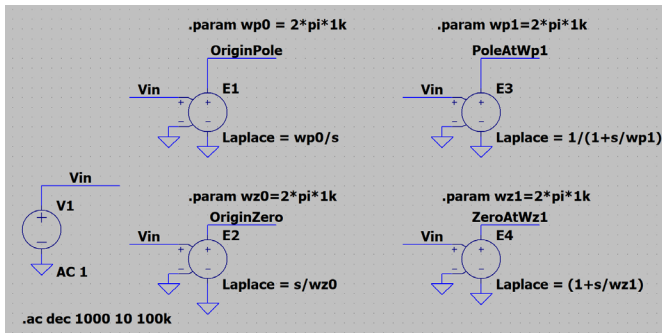


图7. 极点和零点的实现。

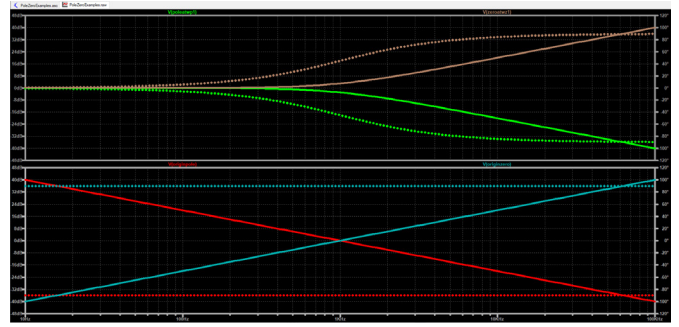


图8. 极点和零点（均在1 kHz频率下）的频率和相位响应。

比较理想和建模的实现方案

通过使用LTSpice来对传递函数进行建模，您可以利用庞大的建模元件库。在第一个示例中，我们来看看提供比例增益的反相运算放大器。在理想实现中， $H(s) = -R_p/R_i$ 。这应该会导致简单的输入电压缩放，以及 180° 相移。但如图9所示，在约 320 kHz时，对比理想增益，增益滚降 3 dB，且相位发生了显著偏移。其推断依据是LT6015的增益带宽为 3.2 MHz，图9所示电路的增益为 -10 V/V。

在LTSpice中，您可以在波形查看器中输入您自己的公式。如果我们想要查看理想公式与运算放大器电路的增益和相位差，那么我们可以添加新波形。操作方式为，右键点击波形查看器，然后选择添加曲线。然后，可以添加表达式，以得到一个节点除以另一个节点（即 $V(\text{vout_pro})/V(\text{vout_laplace})$ ）的绘图。参见图10。

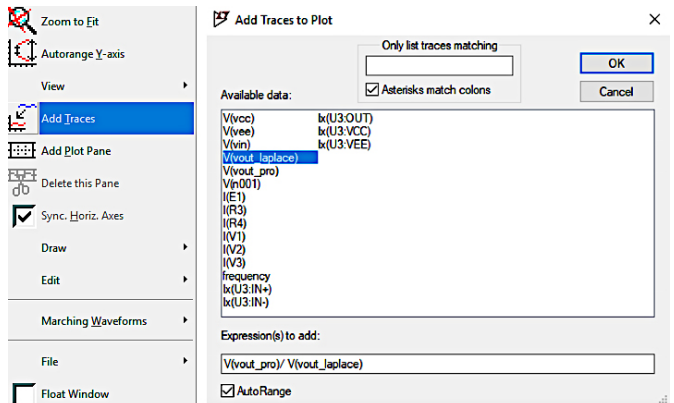


图10. 添加计算出的波形。

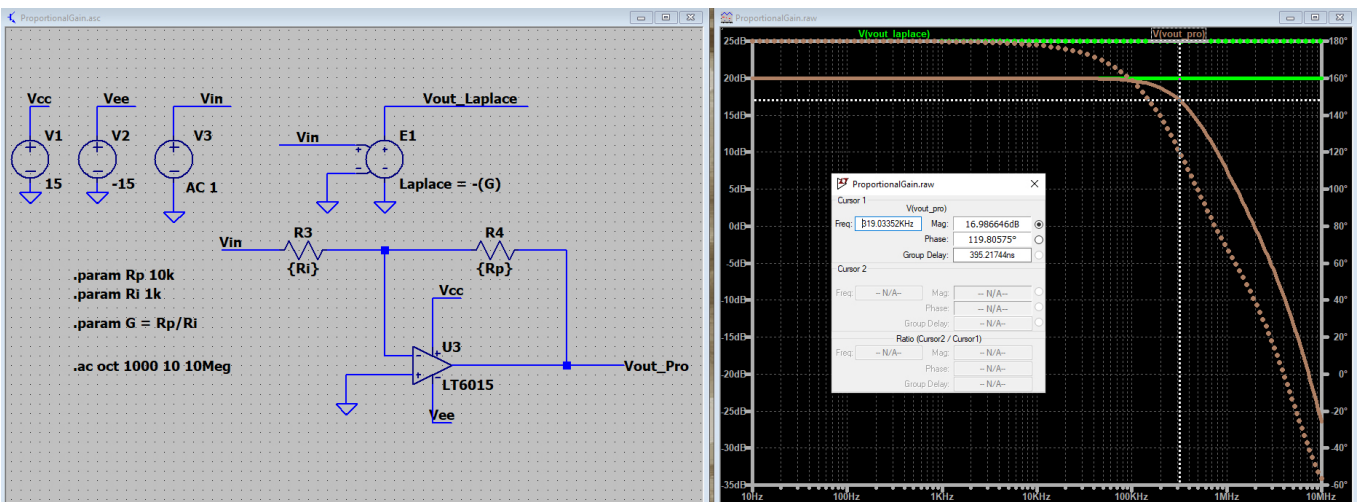


图9. 比例增益。

其增益和相位比较如图11所示，从图中很明显可以看出，在低频率下，增益差为0 dB，无相位差。可以参考示例文件 [ProportionalGain.asc](#)。⁴

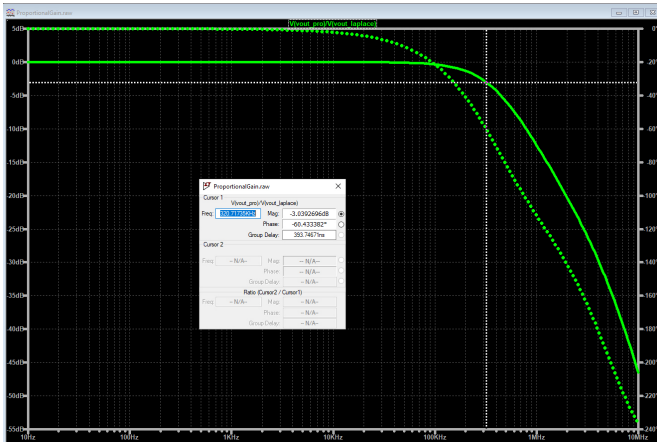


图11. 比较两个波形的增益和相位。

二阶RLC系统

Simon Bramble的文章“[二阶系统电子控制理论：为工程师提供的实用分析](#)”给出了二阶系统的一般形式。⁵

$$H(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (5)$$

当表示为LRC时，我们可以使用以下公式：

$$\omega_n = \frac{1}{\sqrt{LC}}, \zeta = \frac{R}{2} \times \sqrt{\frac{C}{L}}$$

LTspice实现方案如图12所示。在该示例中，您可以指定R、L和C，或者，您可以指定阻尼因数 ζ 和谐振频率 ω_n 。

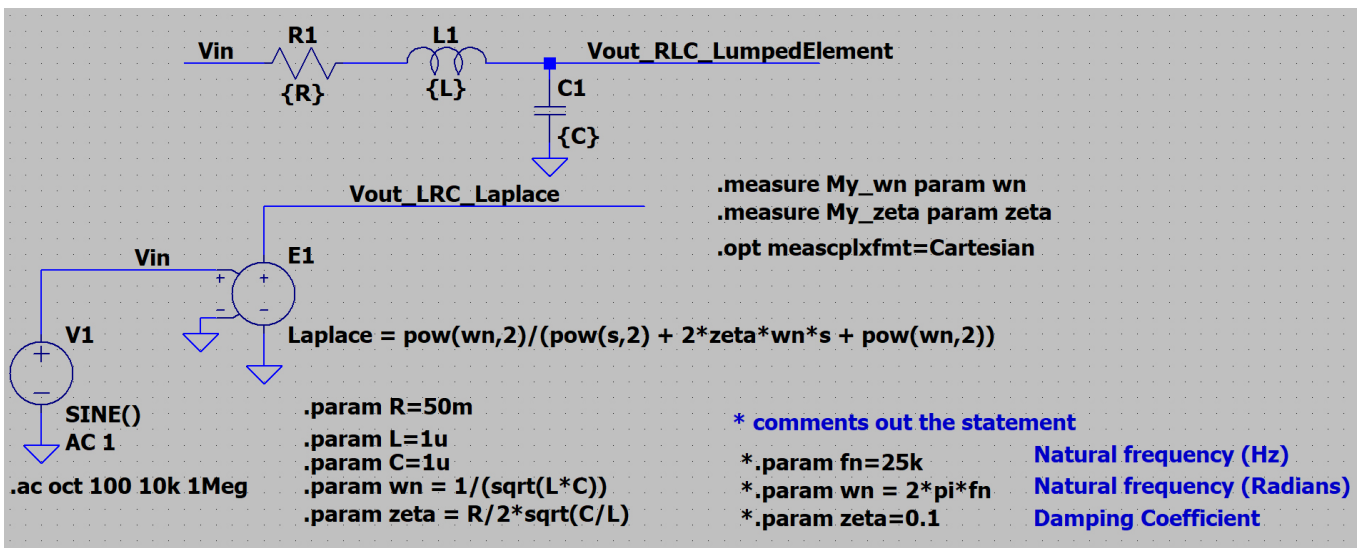


图12. 二阶RLC系统。

这个二阶系统的频率响应如图13所示。图中所示的峰化通常不可取，在进行设计时，例如，为降压稳压器设计输入EMI滤波器时，通常需要对这一峰化进行衰减。可以参考示例文件 [LRC_AC_Analysis.asc](#)。⁶

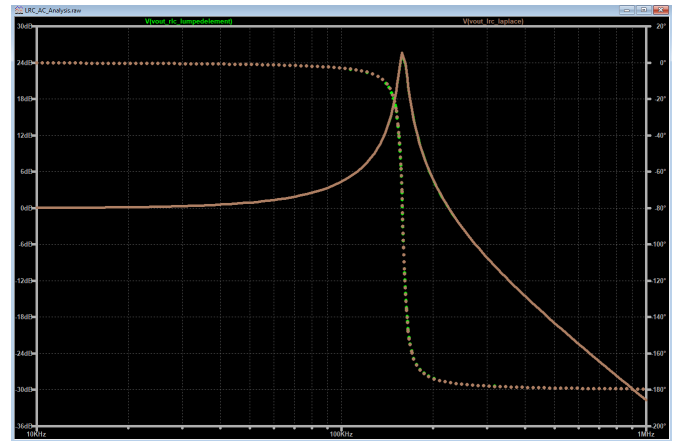


图13. RLC频率响应。

微分器和安全极点

微分器如图14所示。

和预期的一样，原点处的零点导致增益在频率范围内增加20 dB/十倍频程。微分器用于提高对高频率信号的响应。实际上，我们并不会使用纯微分器，而是会在实现过程中加入一个安全极点。加入安全极点是为了使系统不会对高频率噪声太过敏感。此外，请注意运算放大器实现方案中显示的谐振峰值。在尝试控制系统时，这个峰值最终可能会造成问题。

图14所示的实现方案会在以下频率时达到零（增益达到0 dB）

$$f_z = \frac{1}{2\pi \times 100 \text{ nF} \times 10 \text{ k}\Omega} = 159 \text{ Hz} \quad (6)$$

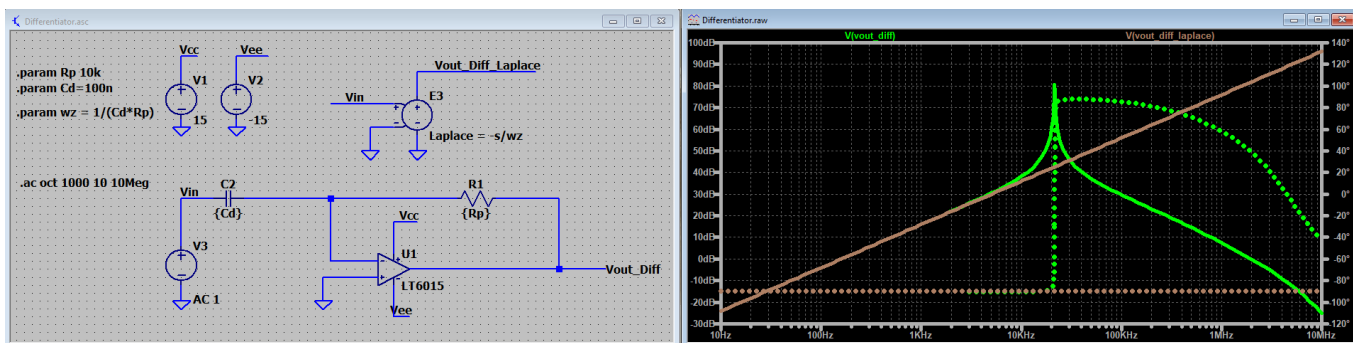


图14. 微分器。

在我们的示例中，安全极点在此值的5倍处实现。图15显示包含安全极点的微分器实现方案。我们所需的极点频率为 $5 \times 159 \text{ Hz} = 796 \text{ Hz}$ 。为了实现这个频率，我们会使用电容值为输入电容的 $1/5$ 的反馈电容。计算整个传递函数的公式为：

$$H(s) = \frac{-s}{\omega_z} \times \frac{1}{1 + \frac{s}{\omega_p}} \quad (7)$$

其中 $\omega_z = 1/R_p C_d$, $\omega_p = 1/R_p C_f$ 。仿真（参见图11）显示，在接近极点频率时，该安全极点会将增益和相位双双降低。由运算放大器导致的峰化也会降低。

一阶加延迟(FOPD)

在对系统的行为建模时，FOPD模型是一个有用工具。有关更完整的说明，请阅读John D. Hedengren博士的“[动态过程建模](#)”报告。⁷简而言之，过程可以用固定延迟、满量程响应和时间常数来描述。要实现固定延迟，我们会利用下方的拉普拉斯变换：

$$L\{\delta(t - \tau)\} = e^{-\tau \times s} \quad (8)$$

注意，时间常数由 $1 - 1/e = 63.2\%$ 定义。FOPD模型表示为：

$$H(s) = \frac{k_{pF}}{1 + s \times \tau_F} e^{-s\theta_F} \quad (9)$$

k_p = 满量程输出

τ_F = 时间常数

θ_F = 时间延迟

图16显示的是满量程为100、时间常数为10 s、延迟为1 s的系统实现方案。因为是在 $t = 1 \text{ s}$ 时应用输入阶跃，所以紧接着在图17中 $t = 12 \text{ s}$ 时，输出为63.2 V。可以参考模拟文件 [FOPD.asc](#)。⁸

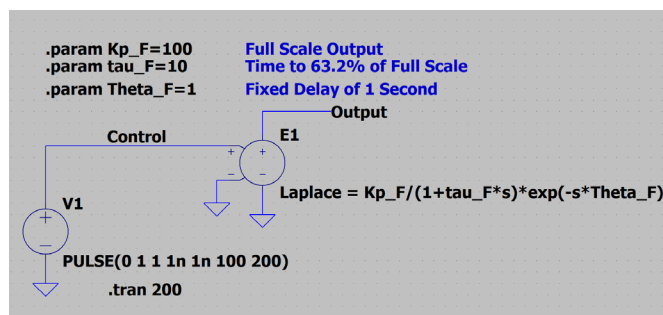


图16. 一阶加延迟模型。

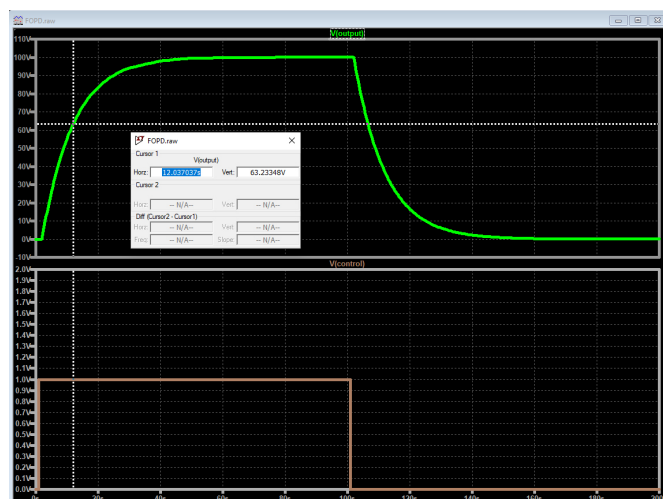


图17. 对输入阶跃的一阶加延迟响应。

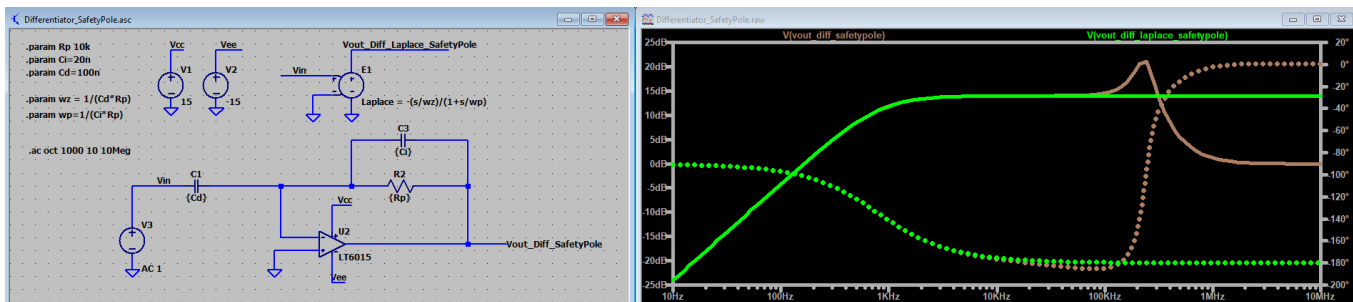


图15. 包含安全极点的微分器。

在LTspice中设计补偿器

控制系统设计是跨越多个学科的复杂领域。补偿器是闭环控制系统的一个关键组成部分。在得出被控系统的传递函数之后，就需要设计补偿器来调节增益和相位，使闭环电路实现稳定性和高性能。

PI (2a型) 补偿器

许多系统可以使用2a型补偿器进行补偿，它由原极点（积分器）和一个零点构成。负反馈提供180°相位延迟，而原极点另外导致90°固定延迟。通过加入一个零点，可以在更高频率下消除这个额外的90°相移。通过使用这种方法，我们可以确定在特定频率下需要多少相移，并据此增加零点。

基于运算放大器的2a型控制器如图18所示。2a型传递函数可以表示为：

$$H(s) = -\frac{1 + \frac{s}{\omega_z}}{\frac{s}{\omega_p}} \quad (10)$$

原极点频率公式为 $\omega_p = 1/R_2 C_i$ ，零点频率为 $\omega_z = 1/R_1 C_i$ 。

在此情况下，运算放大器的作用很明显，导致增益和相位在更高频率下相比理想值双双滚降。可以参考示例文件Type2a.asc。⁹

2型补偿器

2型补偿器由一个原极点和一个极点/零点对构成，表现形式如图19所示。

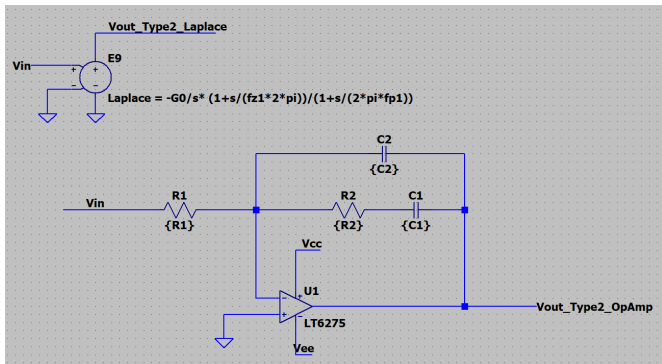


图19. 2型补偿器。

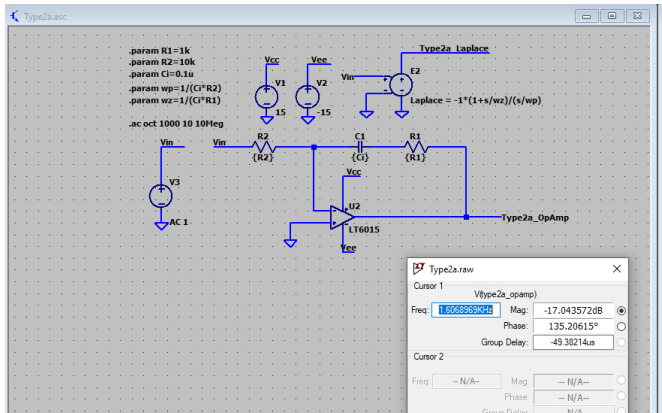


图18. 基于运算放大器的2a型补偿器。

计算传递函数的公式为：

$$H(s) = -\frac{G_0 \left[1 + \frac{s}{\omega_{z1}} \right]}{S \left[1 + \frac{s}{\omega_{p1}} \right]} \quad (11)$$

极点/零点的位置为：

$$\omega_{z1} = \frac{1}{R_2 C_2}, \omega_{p1} = \frac{1}{R_2 \frac{C_1 C_2}{C_1 + C_2}}, G_0 = \frac{1}{R_1 (C_1 + C_2)} \quad (12)$$

此时，设计人员会面临一个两难的情况。R和C的选择对极点和零点都会产生影响。此外，还需要确定加入极点和零点的位置。《设计线性和开关电源的控制回路》一书推导了如何完成设计过程，并给出了一个相关示例。为了帮助读者理解，我们在LTspice示例文件Type2_autoCalcs.asc中采用并说明了此方法。¹⁰ 该示例的输出如图20所示。该示例达成了在10 kHz时实现10 dB衰减和30°相位增幅的目标。和之前一样，运算放大器在更高频率下会显示出其非理想特性。

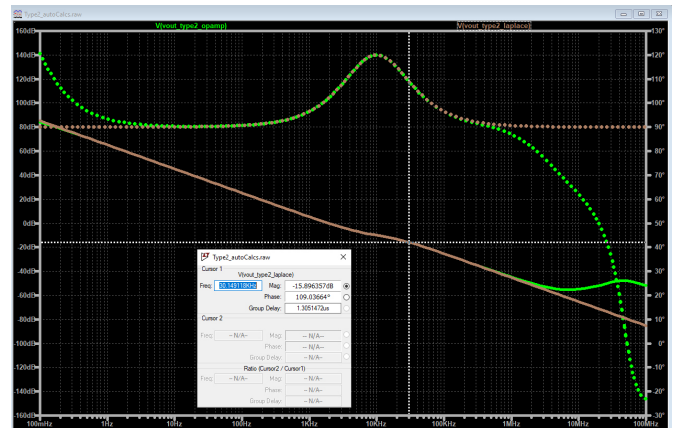
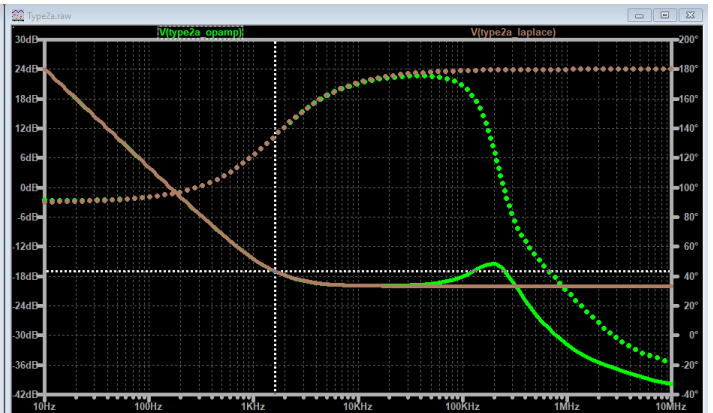


图20. 2型相位和频率响应。



3型补偿器

3型补偿器的表现形式如图21所示。

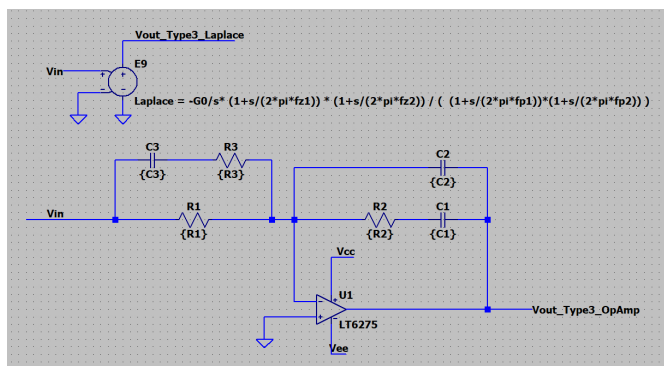


图21. 3型补偿器。

实现这个更复杂的传递函数是为了获得更大的相位增幅。负反馈提供180°初始延迟。但是现在，我们能够使用两个零点，可以将相位增加180°。通过加入两个极点使高频率增益和相位出现滚降。计算传递函数的公式为：

$$H(s) = - \frac{G_0 \left[1 + \frac{s}{\omega_{z1}} \right] \left[1 + \frac{s}{\omega_{z2}} \right]}{S \left[1 + \frac{s}{\omega_{p1}} \right] \left[1 + \frac{s}{\omega_{p2}} \right]} \quad (13)$$

极点/零点的位置为：

$$G_0 = \frac{1}{R_1(C_1 + C_2)} \quad \omega_{z1} = \frac{1}{R_1 C_2} \quad \omega_{z2} = \frac{1}{(R_1 + R_3)C_3}$$

$$\omega_{p1} = \frac{1}{R_2 C_1} \quad \omega_{p2} = \frac{1}{R_3 C_3} \quad (14)$$

与2a型补偿器一样，涉及到选择极点/零点位置，然后计算R和C的过程。《设计线性和开关电源的控制回路》一书推导了如何完成设计过程，并给出了一个相关示例。为了帮助读者理解，我们在LTspice示例文件Type3_autoCalcs.asc中采用并说明了此方法。¹¹ 该示例的输出如图22所示。该示例达成了在5 kHz时实现10 dB衰减和145°相位增幅的目标。和之前一样，运算放大器在更高频率下会显示出其非理想特性。

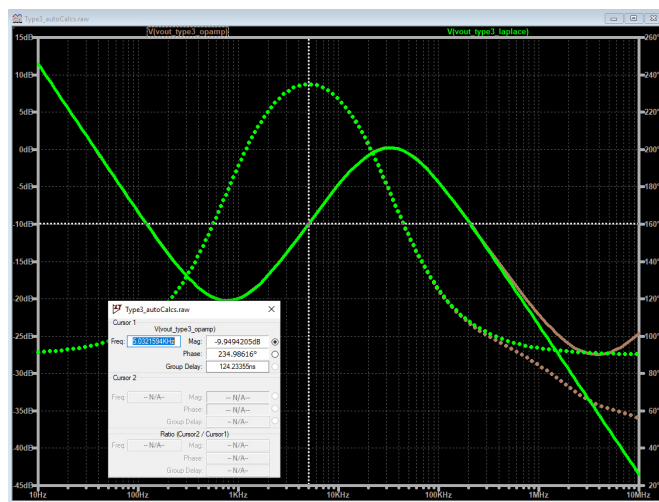


图22. 3型补偿器响应。

时域中的原极点

到目前为止，我们已经展示了LTspice如何利用电路元件和拉普拉斯变换来实现传递函数。所展示的示例采用的均是频域。接下来自然可以在时域（即阶跃响应）中分析这些传递函数。

为此，我们需要解决原极点和如何实现的问题。包含原极点（即纯积分器）的传递函数为 $H(s) = 1/s$ 。这个传递函数在频域（图23）中运行正常，但在时域中会出错（图24）。

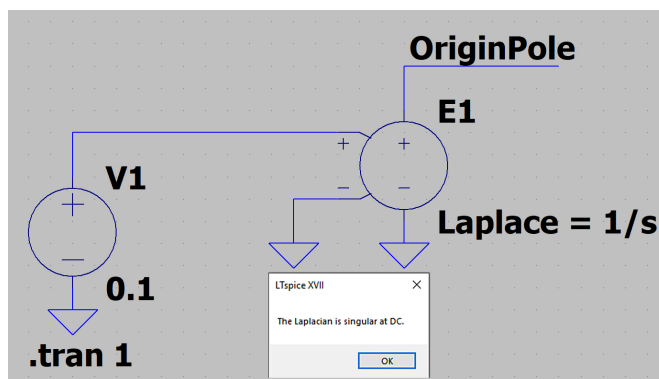


图24. 原极点瞬态分析失败。

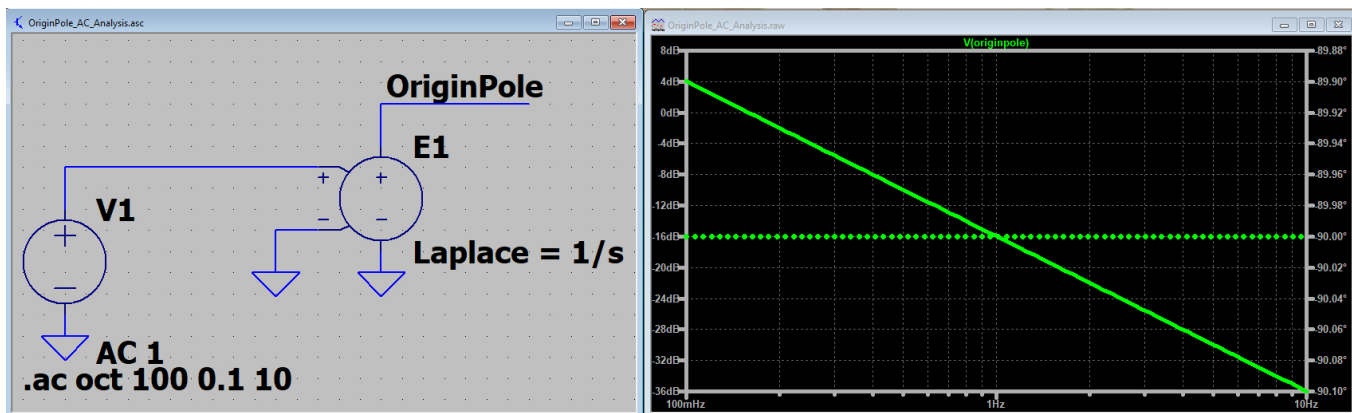


图23. 原极点的AC分析运行正常。

为了解决这个问题，我们可以改为使用行为电压源和idt()函数。此方法如图25所示。图25还给出了反馈系统的开头部分，它比较了对静态反馈电压的阻碍作用。可以参考示例文件OriginPole_TransientAnalysis_idt.asc。¹²

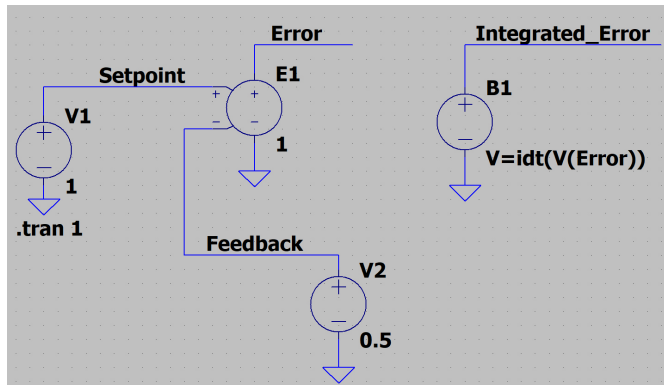


图25. 可在时域中正常运行的积分。

滤波器设计

《线性电路设计手册》¹³只是ADI公司提供的庞大教育材料库中的一部分。该书第8章探讨了模拟滤波器。本章中探讨的滤波器拉普拉斯表示法已作为示例文件Laplace_Filters_LinearSystemsBook.asc给出。¹⁴请参考Precision Toolbox中关于滤波器设计的部分，获取有关设计有源滤波器的帮助。¹⁵

结论

本文通过比较集总元件表示法和基于拉普拉斯变换的公式，就如何实现传递函数提供了大量示例。这些示例作为教育材料，旨在将课堂理论和电路实现结合起来，为各种补偿器的实际设计提供帮助。需要注意的是，对于开关模式电源，LTpowerCAD是一种强大（且首选的）的进行补偿设计的资源。¹⁶虽然LTspice可能会被用于设计和理解某些补偿器，但在设计电源时，LTpowerCAD才是适合此项工作的工具。

致谢：

仿真均是在LTspice中完成的。感谢科罗拉多大学博尔德分校的Shalom D. Ruben博士对本文的指导。

参考资料

- ¹ Simple_RC_vs_R_Divider.asc。ADI公司
- ² Laplace_Simple_RC.asc。ADI公司
- ³ PoleZeroExamples.asc。ADI公司
- ⁴ ProportionalGain.asc。ADI公司

- ⁵ Simon Bramble。“二阶系统电子控制理论：为工程师提供的实用分析”。《模拟对话》，第54卷第3期，2020年9月。
- ⁶ LRC_AC_Analysis.asc。ADI公司
- ⁷ John D. Hedengren。“动态过程建模。”
- ⁸ FOPD.asc。ADI公司
- ⁹ Type2a.asc。ADI公司
- ¹⁰ Type2_autoCalcs.asc。ADI公司
- ¹¹ Type3_autoCalcs.asc。ADI公司
- ¹² OriginPole_TransientAnalysis_idt.asc。ADI公司
- ¹³ 《线性电路设计手册》。ADI公司，2008年。
- ¹⁴ Laplace_Filters_LinearSystemsBook.asc。ADI公司
- ¹⁵ ADI Precision Studio。ADI公司
- ¹⁶ Henry Zhang。“应用笔记149：开关模式电源的模型和回路补偿设计。” 凌力尔特，2015年1月。

Basso, Christophe。设计线性和开关电源的控制回路。Artech House，2012年10月。

Differentiator_SafetyPole.asc。ADI公司

Type3_autoCalcs_steps.asc。ADI公司

作者简介

Joseph Spencer自2013年以来一直在凌力尔特（现在已成为ADI公司的一部分）担任现场应用工程师。Joseph为犹他州和爱达荷州客户提供支持，主要负责高性能模拟和RF解决方案。在担任现场应用工程师之前，Joseph在L3 Communications公司工作，负责软件定义无线电的硬件设计和管理。Joseph最初就职于Maxim Integrated公司，曾担任应用和测试工程方面的职位。Joseph拥有犹他大学电气工程学士和硕士学位。联系方式：joseph.spencer@analog.com。

在线支持社区



访问ADI在线支持社区，中文技术论坛
与ADI技术专家互动。提出您的
棘手设计问题、浏览常见问题
解答，或参与讨论。

请访问ez.analog.com/cn

