

学校代码 10530

学 号 201330101249

分 类 号 TN713

密 级 公开

湘潭大学

硕 士 学 位 论 文

一种流水线ADC数字校准算法的实现 与研究

学 位 申 请 人 张文杰

指 导 教 师 谢亮副教授

学 院 名 称 物理与光电工程学院

学 科 专 业 集成电路工程

研 究 方 向 数字集成电路设计

二〇一六年五月

Implement And Research On a Pipelined ADC Digital Calibration Algorithm

Candidate_____Wenjie Zhang_____

Supervisor_____Associate Professor Xieliang_____

College_____Facilty of Physics and Optoelectronic Engineering_____

Program_____Integrated Circuit Engineering_____

Specialization_____IC CAD and ASIC Design_____

Degree_____Master of Engineering_____

University_____Xiangtan University_____

Date_____May, 2016_____

摘要

流水线 ADC(Pipeline ADC)兼顾高速度和高精度被广泛地运用于视频成像系统、通信系统和测试设备中。而在实际应用中,流水线 ADC 的性能会受到电容失配、比较器失调和放大器的有限直流增益等误差的限制。随着 CMOS 工艺的发展,通过复杂的模拟电路来补偿流水线 ADC 的误差已经变得越来越困难。数字校准技术使用数字电路校准流水线 ADC 模拟电路的误差,降低了流水线 ADC 模拟电路设计的难度,成为进一步提高流水线 ADC 性能新的手段。

本文首先根据 CMOS 工艺的发展情况,阐述了使用数字电路辅助设计流水线 ADC 的优点,随后介绍了国内外主流的流水线 ADC 校准算法,在分析了它们的优缺点的基础上提出本文所研究的校准算法。

其次,本文系统的分析了流水线 ADC 模拟电路结构上存在的各种误差,并且全部建立了对应的误差模型,为研究基于参考 ADC 与自适应滤波器的流水线 ADC 校准算法打下基础。

然后,本文从流水线 ADC 的误差出发,利用 MDAC 的传递函数介绍了基于参考 ADC 与自适应滤波器的流水线 ADC 校准算法的基本原理,并且提出了变步长的 LMS 算法改进这种校准算法。

最后,利用流水线设计方法优化校准系统的电路结构,大幅度提高了校准系统的工作频率,同时使用 Xilinx 的 V5-110T 型 FPGA 验证了变步长 LMS 校准算法的校准效果,结果表明,与基本校准算法相比,变步长的校准算法拥有更好的校准性能。

关键词: 流水线 ADC; 误差模型; 流水线设计; 数字校准算法

Abstract

Pipelined ADC with high speed and high accuracy is widely used in video imaging system, communication system and test equipment. In practical applications, the performance of pipelined ADC is limited by the error of capacitor mismatch, comparator offset, and finite DC gain of the amplifier. With the development of CMOS process, it has become more and more difficult to compensate the error of pipelined ADC through the complex analog circuits; The use of digital calibration circuit can reduce the analog circuit's design difficulty, and be a step to improve the performance of the pipelined ADC new means.

Firstly, according to the development of CMOS technology, this article firstly expounds the advantage of using the digital circuit aided design of the pipelined ADC; then introduces the abroad mainstream pipelined ADC calibration algorithm, and analysis their advantages and disadvantages based on the calibration algorithm.

Subsequently, this paper analysis all kinds of errors of the pipelined ADC analog circuit structure, and established the corresponding error model, which is the basic of reference ADC adaptive filter and the pipelined ADC calibration algorithm research.

Thirdly, it starts with the errors of the pipelined ADC and uses the MDAC transfer function to introduce the basic principle based on the reference ADC and adaptive filter pipelined ADC calibration algorithm. Then the paper put forward the variable step size LMS algorithm to improve the calibration algorithm.

Finally, the circuit structure of the calibration system optimized by the pipelining design greatly improves the working frequency of system calibration. And use Xilinx V5-110T type FPGA simultaneously verifies the calibration results of the variable step size LMS calibration algorithm. The results show that compared with the basic calibration algorithm, variable step calibration algorithm has better calibration performance.

Key words: pipelined ADC; error model; pipeline design; digital calibration algorithm

目 录

第 1 章 绪论.....	1
1.1 研究背景与意义.....	1
1.2 国内外研究现状.....	3
1.3 本文的组织架构.....	5
第 2 章 流水线 ADC 的误差模型与 LMS 算法原理.....	7
2.1 流水线 ADC 的测试指标	7
2.2 流水线 ADC 的工作原理与电路结构	9
2.2.1 流水线 ADC 工作原理	9
2.2.2 流水线子级的电路结构.....	10
2.3 流水线 ADC 的误差分析与建模	13
2.3.1 流水线 ADC 的误差来源	13
2.3.2 流水线 ADC 的建模	17
2.4 基于 LMS 算法自适应滤波器原理与建模	21
2.5 本章总结.....	23
第 3 章 基于参考 ADC 的流水线 ADC 校准算法.....	24
3.1 基于参考 ADC 的校准原理	24
3.2 校准算法的建模与仿真.....	28
3.3 校准算法的改进.....	33
3.4 本章总结.....	37
第 4 章 基于参考 ADC 的流水线 ADC 校准算法的硬件设计.....	38
4.1 硬件电路设计说明.....	38
4.2 硬件电路细节说明.....	39
4.2.1 校准模块的流水线设计	41
4.2.2 输入延时模块与误差产生模块的设计	43
4.2.3 抽头系数迭代模块的设计	45
4.2.4 其他模块的设计	49
4.3 板级验证.....	50
4.4 本章总结.....	56
第 5 章 总结与展望.....	57
参考文献.....	58

致谢.....	62
附录 A 个人简历	63
附录 B 在校期间发表的学术论文及研究成果	64

第 1 章 绪论

1.1 研究背景与意义

模数转换器(Analog-to-digital,ADC)是将模拟信号转换为数字信号的电子器件。图 1.1 给出了 ADC 的功能框图,在如今的电子系统中,ADC 首先对模拟信号进行采样,随后将采样信号量化成数字信号,并对数字信号进行编码得到数字码输出完成模拟信号向数字信号的转换^[1-2]。ADC 的输出信号将由数字系统进行处理和加工,所以作为模拟世界和数字系统的桥梁,ADC 的性能制约着整个电子系统的性能。

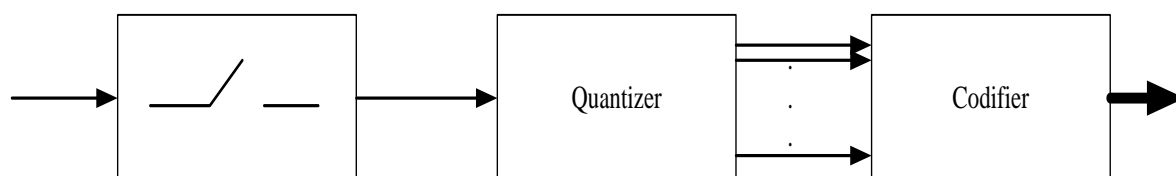


图 1.1 ADC 功能框图

按照架构 ADC 可以分为：增量-累加型 ADC(Σ - Δ ADC)、逐次逼近型 ADC(SAR ADC)和流水线 ADC(Pipelined ADC)。 Σ - Δ ADC 由 Σ - Δ 调制器和数字抽取滤波器组成,其中 Σ - Δ 调制器是以过采样与噪声整形为基础,以速度换取精度的电路结构。 Σ - Δ ADC 工作时,首先以较高的采样频率完成模拟信号向数字信号的转换,接下来由调制器拓扑实现噪声整形功能,将低频的量化噪声转移到高频中,最后由低频数字抽取滤波器滤掉高频噪声,将 Σ - Δ ADC 的输出信号频率降低到某个可用的频率^[3-4]。 Σ - Δ ADC 在各种 ADC 当中分辨率是最高的,所以被广泛地运用在测量和测试、科学仪器和工业过程控制当中。

SAR ADC 由 DAC、逐次逼近寄存器和一个电压比较器组成。SAR ADC 工作时,首先令模拟输入信号与 DAC 的输出通过电压比较器进行比较,然后由逐次逼近寄存器根据比较结果控制 DAC 的输出不断逼近模拟输入电压,通过这样不断的比较和逼近,最后将逐次逼近寄存器存储的数据作为 SAR ADC 的输出^[5]。SAR ADC 转换速度快、功耗低,能以比较低的制造成本获得较高的分辨率,被广泛地运用在各种中、低速场合中。

流水线 ADC 由前端采样保持电路、流水线子级电路和 Fash ADC 组成。流水线 ADC 的时序由受采样时钟和保持时钟的控制,工作时模拟输入信号逐次通过各级流水线子级电路和 Fash ADC,所有流水线子级和 Fash ADC 都会对模拟输入信号进行量化,最后将所有的数字输出信号综合在一起就是流水线 ADC 的

输出^[6-7]，流水线结构让流水线 ADC 获得良好的速度和精度，所以被广泛的运用于视频成像系统、通信系统和测试设备中。

如今随着消费电子的不断发展，人们对移动电子设备的需求激增。仅仅从 2012 年到 2013 年期间，移动数据量就增加了百分之七十，是 2000 年的十二倍以上，这其中百分之五十的增长量来自手机^[8]。这就迫切要求 ADC 往低功耗，高精度和高速度方向发展，流水线 ADC 的速度和精度有着良好的折中，所以全面提高流水线 ADC 的性能成为了研究热点。

伴随着电子技术的不断发展，集成电路的制造技术也得到很大的提高。在过去的数十年中，CMOS 工艺的发展一直受到摩尔定律的指导，图 1.2 给出了近些年 COMS 工艺发展情况^[9]。

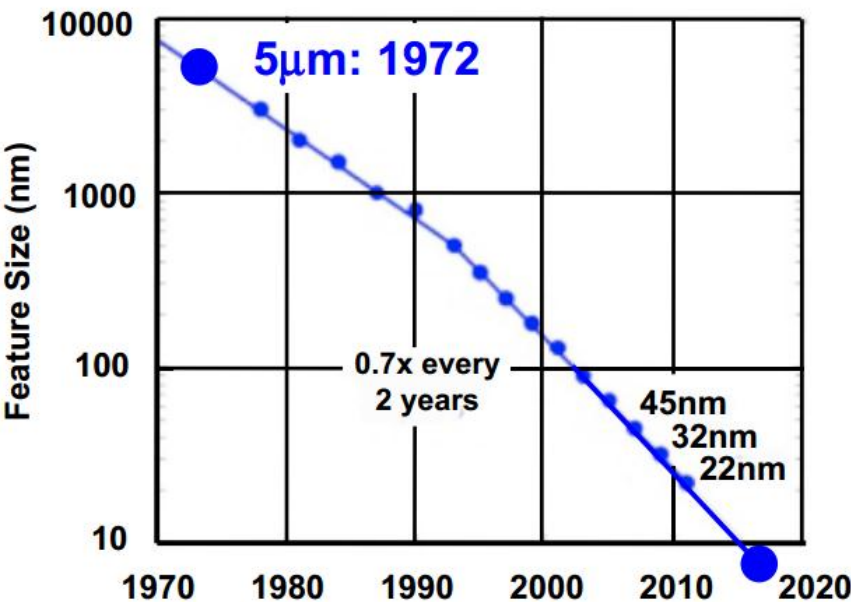


图 1.2 近些年 COMS 工艺发展情况

从图 1.2 可以看出 1972 年 COMS 工艺尚停留在 5 μ m 的水平，而 2020 年将达到 5nm。器件尺寸的缩小极大的降低了功耗，例如一个有两个输入端的与非门，在 0.5- μ m 工艺下每次逻辑操作需要消耗 1pJ 的功耗；而在 27-nm 的 COMS 工艺下，功耗是前者的一千分之一^[10-11]。与此同时，微型处理器的时钟频率每 2.3 年提高一倍，整体性能每 1.5 年提高一倍，这些都极大地促进了数字电路的发展，令数字电路在系统中的地位越来越重要。然而这一系列的进步却给模拟电路设计带来了挑战。首先电源电压不断降低，在亚微米工艺时代，电源电压由 10-5V 降低 1.8-3.3V，如今工艺从亚微米进入纳米时代电源电压将至 1V。电源电压降低使得信号的摆幅减小，这样对电路中的信号比(SNR)造成了冲击。另外由于电源电压

的降低,运算放大器的增益越来越难以提高。所以利用数字电路设计流水线 ADC 成为了研究重点。

1.2 国内外研究现状

数字校准技术就是使用数字电路快速处理和分析数据的优势来减低模拟电路的设计难度,它的思路一般是首先采取某种方法观察或者跟踪模拟电路的误差,然后在数字域中通过某种算法补偿模拟电路的误差。采用数字校准技术设计 ADC 的方法早在 1981 年便被提出^[12],随着 CMOS 工艺的不断进步,数字电路的性能得到了显著的提高,越来越多的数字校准技术被运用在 ADC 设计当中。如图 1.3 所示,近几年的 ISSCC 论文中百分之四十的数据转换器设计采用数字校准技术^[13],数字校准技术已经成为 ADC 设计的主要潮流。下面主要阐述流水线 ADC 的数字校准技术的发展历程。

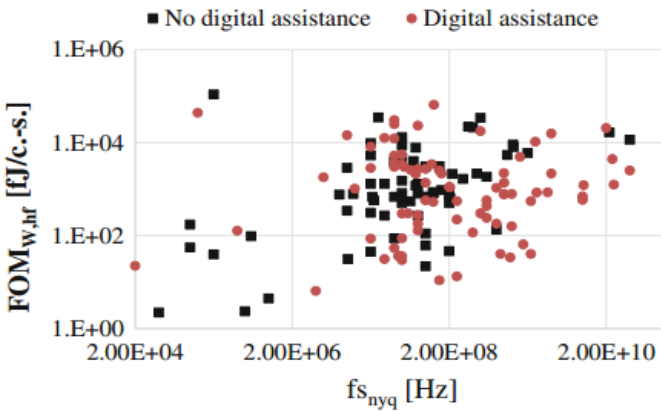


图 1.3 数字校准技术的使用情况

流水线 ADC 的数字校准技术一般分为前台校准技术和后台校准技术,前台校准技术在校准过程中需要打断流水线 ADC 正常工作,例如“skip-and-fill”校准算法^[14],这种校准算法周期性的利用一个已知信号代替输入信号,通过观察这个信号在电路中的变化提取模拟电路的误差。根据这种思路文献[15]提出使用正弦波估计电容失配和运放的直流增益误差。

流水线 ADC 的前台校准技术拥有较快的校准速度,但是会影响 ADC 的正常工作,所以运用范围受到了较大的限制。与前台校准技术相比,流水线 ADC 的后台校准技术在校准过程中不需要打断 ADC 正常工作,且可完全由数字电路实现,所以受到人们关注。下面介绍几种常用的后台校准技术:

(1)PN 注入法

利用 PN 码校准流水线 ADC 被广泛的运用在基于统计学与基于相关性的校准算法中, 文献[16]中提出利用 PN 码校准 ADC 的方法; 文献[17]将 PN 码运用在校准流水线 ADC 当中, 这种校准技术首先将 PN 码注入到子 DAC 中, 然后在流水线 ADC 的输出端注入相同的 PN 码, 通过做自相关运算提取了级间增益误差, 然后运用自适应的思想根据提取的误差校准流水线 ADC 的非线性误差, 这类校准算法统称为基于相关性校准算法。文献[18]使用基于统计学与 PN 码的数字校准算法设计了一款 12bit 75M 的流水线 ADC, 该 ADC 第一级是一个有两种工作模式的残差放大器, 在工作时由 PN 码控制两种工作模式的转换, 然后在流水线 ADC 的后端比较两种工作模式的输出码密度曲线以提取误差, 最后利用查找表校准放大器的非线性误差。这两种校准方法都过于依赖 PN 码, 且校准速度缓慢。文献[19]指出, 使用基于统计学的校准方法校准 N bit 的流水线 ADC 的增益误差, 需要很多时钟周期完成校准, 一个 11b 的流水线 ADC 就需要将近四百万个时钟周期完成校准。在文献[20]中 Jipeng Li 和 Mu-Ku 指出对于基于相关性的校准算法来说, 无论是将 PN 注入到子 DAC 中, 还是将 PN 注入到子 ADC 中, PN 码的长度都极大的制约了此类校准方法的准确性, 这因为输入信号影响 PN 自相关运算的精度。为此 Jipeng Li 和 Mu-Ku 提出一种新的 ADC 结构, 即设计两个同样的流水线 ADC 代替双通道 ADC, 给予它们相同的但极性相反的输入信号, 然后在输出信号端口让两个数字输出信号综合一起, 以消除输入信号对自相关运算的影响, 但 PN 码的长度依然在很大程度上制约着校准的准确性。随后 Bang-Sup Song 根据输入信号的幅度来选择 PN 序列的幅度^[21], 使得伪随机码的幅度影响进一步减小。

(2)基于参考 ADC

基于参考 ADC 的校准算法主要是利用低速高精度的参考 ADC 提供期望信号, 通过对比流水线 ADC 的输出信号与期望信号提取模拟电路的误差。

Yun Chiu 与 Cheongyuen W 在文献[22]中提出根据码域均衡的原理, 即使用低速高精度的参考 ADC, 结合 LMS 自适应滤波器的方法校准 1.5bit/级的流水线 ADC 的校准方法; 这种方法根据流水线 ADC 的结构和工作原理, 首先利用参考 ADC 的输出与流水线 ADC 的输出作差提取误差参数, 然后利用自适应滤波器根据误差参数处理流水线子级的数字码输出, 在后端补偿流水线 ADC 模拟电路的误差。这种校准方法不会改变模拟电路原来的结构, 可完全由数字电路实现, 能校准多种误差, 但还不能校准流水线 ADC 运放的非线性误差。Xiao yue Wang 也于文献[23]提出基于参考 ADC 与自适应滤波器的流水线校准算法, 并且基于 0.35-um CMOS 工艺设计了校准 20M/s 12bit 流水线 ADC 的实际电路, 此后 Xiao

yue Wang 在原来的基础上优化了电路结构, 去掉了前端 S/H 电路, 从而降低了功耗^[24]。

(3) 基于“split ADC”结构

文献[21]在改进基于相关性的流水线 ADC 校准算法时已经提出了这种结构, John 在文献[25]中使用“split ADC”结构设计了一款 16bit 1-Ms/s ADC。该 ADC 由两个一模一样的子 ADC 组成, 通过比较两个 ADC 的输出提取误差因子, 然后采用自适应的思想校准 ADC。John 在文献[26]中再次使用“split ADC”结构校准流水线 ADC 的开环残差放大器非线性误差, 随后 John 将这种校准思想运用在时间交织 ADC 与 SAR ADC 当中^[2-28]。这种方法不会增加模拟电路的复杂度, 且拥有较快的校准速度, 但是为了减小电路面积, 两个子 ADC 的精度都不高, 无法提供比较纯净的参考信号。

近几年很多流水线 ADC 校准算法都受上述三种类型校准算法的影响。如文献[29]中为了减少模拟电路中的功耗, 使用开关电容放大器设计流水线 ADC, 然后根据流水线子级的传递函数利用自适应算法校准流水线 ADC, 文献[30]采用前台数字校准技术估计级间误差, 使用后台数字校准技术补偿这些误差; 文献[31]则结合了 PN 注入, split ADC 和多项式来校准流水线 ADC 中 MDAC 的非线性误差。从近些年的文献可以看出流水线 ADC 的数字校准呈以下趋势:

1. 全数字化, 最大程度的利用数字电路优异的性能减轻模拟电路的设计压力, 如在使用数字校准前提下, 降低高增益运放的设计指标;
2. 后台校准, 校准必须在不打断 ADC 正常工作的情况下进行;
3. “split ADC”结构成为研究热点, 模拟电路的设计重点偏向配合数字校准算法的设计。

与国外的研究相比。国内对于校准技术的研究起步比较晚, 但也已经相继发表了一些论文^[32-34];可总体落后于国外。

1.3 本文的组织架构

本文所研究的是基于参考 ADC 的流水线 ADC 校准算法, 第一章作为绪论, 首先介绍了 COMS 工艺的发展对模拟集成电路与数字集成电路的影响, 然后介绍了比较常见的流水线 ADC 数字校准算法, 并且分析了它们的优缺点。

第二章首先分析流水线 ADC 模拟电路中存在的各种误差, 以及这些误差对 ADC 性能的影响, 同时根据这些误差建立了流水线 ADC 的误差模型。接下来介绍了基于 LMS 的自适应滤波器的基本原理, 并且通过仿真验证了基于 LMS 算法的自适应滤波器的功能。

第三章在第二章的基础上介绍了基于参考 ADC 与自适应滤波器的流水线 ADC 校准算法的原理,并且通过 Simulink 模型验证了这种校准技术的校准效果,最后根据 LMS 算法原理使用变步长的 LMS 算法改进了原始的校准技术。

第四章分析了基于参考 ADC 与自适应滤波器的流水线 ADC 校准技术移植到数字电路上的难点,并针对这些难点进行了流水线设计,优化了校准系统的电路结构,提高了校准系统的工作频率,最后通过利用 Xilinx 的 FPGA 完成了校准算法的板级验证。

第五章总结了文本的工作,并为下一阶段工作指明方向。

第2章 流水线 ADC 的误差模型与 LMS 算法原理

本章简要的介绍流水线 ADC 的模拟电路结构和性能指标，同时根据电路结构分析了流水线 ADC 的误差来源，在 Matlab/Simulink 模型上建立流水线 ADC 的误差模型。最后介绍基于 LMS 算法的自适应滤波器的工作原理。

2.1 流水线 ADC 的测试指标

ADC 的性能指标分为静态性能指标和动态性能指标，静态性能指标由微分非线性(DNL)和积分非线性(INL)组成；动态性能参数由信噪比(SNR)、总谐波失真(THD)、无杂散动态范围(SFDR)和信纳比(SNDR)等性能参数组成。

(1)微分非线性

微分非线性是指理想的量化码宽与实际的量化码宽的差别，微分非线性定义的数学表达式如式 2.22 所示

$$V_{LSB} = \frac{\Delta V_{(2^N-1)}}{2^N - 1} \quad (2.1)$$

$$DNL(n) = \frac{V_{n+1} - V_n}{V_{LSB}} - 1 \quad (2.2)$$

式 2.1 与式 2.2 中 $\Delta V_{T(2^N-1)}$ 代表数字信号为 2^N-1 时对应的模拟输入电压， V_{n+1} 代表数字输出为 $n+1$ 时对应的模拟电压， V_n 代表数字输出为 n 时对应的模拟电压。由式 2.1 与 2.2 可得如果一个 N bit 的 ADC，它的 DNL 大于 $1LSB$ ，那么这个 ADC 的数字码输出可能不会随着模拟输入电压的增大而增大，也就是说这个 ADC 不具有单调性。

(2)积分非线性

积分非线性是指 ADC 的数字码对应的理想模拟值和实际模拟值的区别，它是微分为线性累积的结果，数学表达式如式 2.3 所示

$$INL = \sum_{n=1}^{2^N-1} DNL(n) \quad (2.3)$$

(3)信噪比

信噪比是指 ADC 的信号能量与噪声能量之比，这里的噪声包括 ADC 的量化噪声和电路产生的噪声，但是不包括谐波。对于一个 N bit 分辨率的 ADC 的 SNR 如式 2.4 所示

$$SNR_{\max} = 6.02 \cdot N + 1.76 \quad (2.4)$$

信噪比的大下与输入电压幅度成正比，所以在测量 ADC 的信噪比时，一般采用满摆幅输入。

(4)总谐波失真

总谐波失真是指频谱中前几次谐波与信号振幅的比值，如式 2.5 所示

$$THD = \frac{\sqrt{\sum_{k=2}^7 A_{(k \cdot f_{in})}^2}}{A_{f_{in}}} \quad (2.5)$$

式 2.4 中 $A_{f_{in}}$ 代表输入频率为 f_{in} 幅值为 A 的输入信号， $A_{n \cdot f_{in}}$ 为频率为 f_{in} 的谐波，计算 THD 时一般取前六次谐波。

(5)无杂散动态范围

SFDR 是指幅值最大的谐波与信号振幅之比。

(6)信纳比

信纳比是指所有谐波能量与噪声能量和信号能量之比，往往 SNDR 与 SNR 越接近越好。

(7)有效位数(ENOB)

有效位数由信噪比决定，如式 2.6 所示

$$ENOB = \frac{SNR - 1.76}{6.02} \quad (2.6)$$

此外失调是指 ADC 的数字码输出从 0 跳变成 1 时，实际对应的模拟电压转换点与理想的模拟电压转换点的差值。增益误差是指 ADC 理想的输入输出曲线与实际的输入输出曲线的差别。量化噪声是指由于 ADC 分辨率有限所带来的误差，ADC 的分辨率越高量化噪声越小。

上述 ADC 的性能参数都是对 ADC 输出数据作相关处理得到，ADC 的静态参数是通过码密度测试法得出，码密度测试法一般使用满摆幅正弦信号作为输入，然后统计输出信号中所有数字码出现的概率，对于信号 $A \sin(\omega \cdot t)$ ，它的电压密度函数如式 2.7 所示

$$p(V) = \frac{1}{\pi \cdot \sqrt{A^2 - V^2}} \quad (2.7)$$

公式 2.8 给出了电压区间($V, V+dV$)在整个正弦波电压摆动范围中的概率。

$$p(V, V + \Delta V) = \frac{1}{\pi} \cdot \left\{ \sin^{-1} \frac{V + dV}{A} - \sin^{-1} \frac{V}{A} \right\} \quad (2.8)$$

通过式 2.8 可以计算中 ADC 某个电压区间在整个输出电压区间出现的概率，也就能求出所有码字出现的概率，这样通过比较输出码字在输出信号中的实际概率和理想概率求出 DNL 与 INL。

为了保证性能参数的精确度，所以对用于计算 DNL 与 INL 的输出信号点数必须满足一定要求，具体如式 2.9 所示

$$N \geq \frac{Z_{\sigma}^2 \cdot \pi \cdot 2^{n-1}}{\beta^2} \quad (2.9)$$

式 2.9 中 Z_{σ} 代表测试精度为 $(1-\sigma/2)$ 时的可信度，代表容许的误差， n 为 ADC 的位数，在实际检测当中，在计算静态参数的精确程度越高或 ADC 的分辨率较高的情况下，计算静态参数所需要的输出点数也就越多。例如对于一个 10bit 的 ADC 要求测试精度以 99% 的可信度控制在 0.1LSB，那么用于计算的输出数据至少需要 1.07 个。

ADC 的动态参数是通过对输出信号作 FFT 分析而得到的，这里的输出信号一般是 ADC 对一个满量程正弦波量化得到的。为了避免频谱泄露，提高动态参数的可信度，用于 FFT 分析的输出信号的频率、采样点数等如式 2.10 所示

$$\frac{f_{in}}{f} = \frac{m_t}{m} \quad (2.10)$$

f_{in} 是输入信号频率， f 是 ADC 的采样频率，是用于 FFT 分析的输出信号周期数， m 为用于 FFT 分析的输出信号点数， m_t 为采样信号周期数， m 与 m_t 互为质数。为了增加 FFT 分析的精确度，数据在 FFT 分析之前还可以使用窗函数处理。

2.2 流水线 ADC 的工作原理与电路结构

2.2.1 流水线 ADC 工作原理

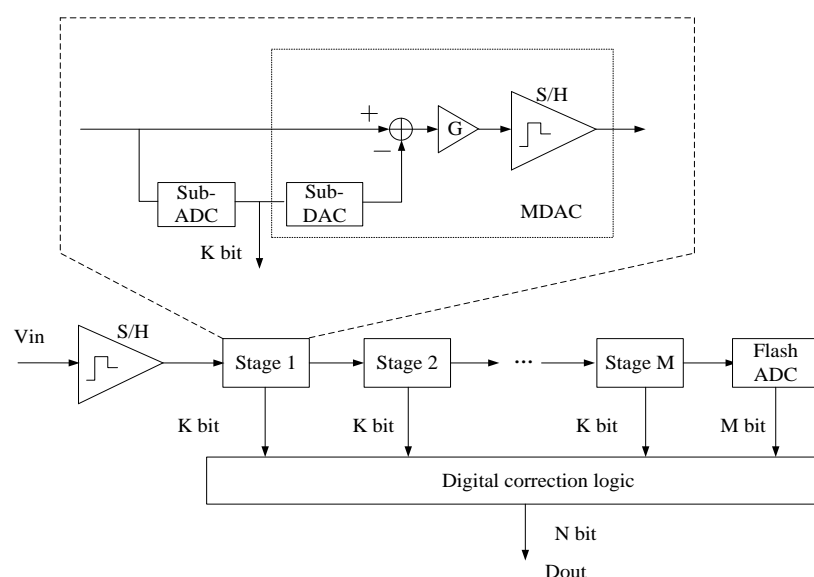


图 2.1 流水线 ADC 的原理图

图 2.1 为流水线 ADC 的原理框图，流水线 ADC 一般使用冗余编码结构，它由前置采样保持(S/H)电路、流水线子级电路与 Flash ADC 组成，流水线子级由 ADC(Sub-ADC)与残差放大器组成(MDAC)，MDAC 由子 DAC(Sub-DAC)、运算放大器和下一级的采样保持电路组成。

流水线 ADC 的所有动作由两个双相非交叠时钟控制，即采样时钟和保持时钟。在流水线 ADC 工作时，采样时钟和保持时钟控制所有流水线子级交替处理各自的输入信号。首先流水线子级在采样时钟的控制下完成对输入信号的采样，然后由子 ADC 对输入信号进行粗略的量化得到 K bit 的数字信号，保持时钟控制子 DAC 将子 ADC 的数字输出转换为模拟信号，并且将这部分信号从输入信号中分离出去得到残差信号。为了提高信号的动态范围，由残差放大器将残差信号放大，放大后的残差信号作为下一级流水线子级的输入信号，最后将所有流水线子级中的子 ADC 的输出综合在一起作为流水线 ADC 的输出信号。

尽管模拟信号从前置采样保持电压输入，到经过最后的 Flash ADC 完成量化需要几个时钟。但是流水线 ADC 的后端有专门的延时单元，对不同的流水线子级的数字码输出作不同的延时处理，这样就同步了所有子 ADC 的数字输出信号，保证了流水线 ADC 准确地将模拟信号转换成数字信号。流水线子级一般使用同样的电路结构，但设计难度并不一样。如图 2.2 所示，框图的大小代表设计的难度，第一级流水线子级的数字码输出代表流水线 ADC 的 MSB，在 ADC 输出中占有最大的权重，所以必须拥有最好的性能，设计要求也最高。模拟信号经过某一级流水线子级处理后马上传递到下一级流水线子级，流水线子级中子 ADC 的数字码输出随着模拟信号的不传递在整个输出中所占权重也越来越低，那么设计要求也就越来越低。最后一级流水线子级只需要执行采样和量化，不需要 MDAC 所以使用 Flash ADC 代替。

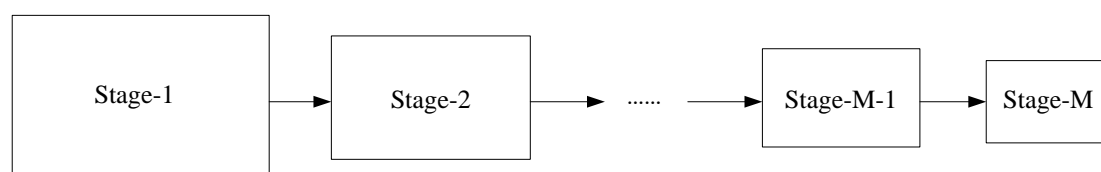


图 2.2 流水线子级的设计要求示意图

2.2.2 流水线子级的电路结构

(1)子 ADC

上一节已经指出，子 ADC 负责对各流水线子级的模拟输入信号做初步量化，通常使用 Flash 结构设计。不同结构的 MDAC 对应着不同分辨率的子 ADC，

1.5bit/stage、2.5bit/stage 和 3.5bit/stag 的流水线子级分别对应着 2bit、3bit 和 4bit 的子 ADC。

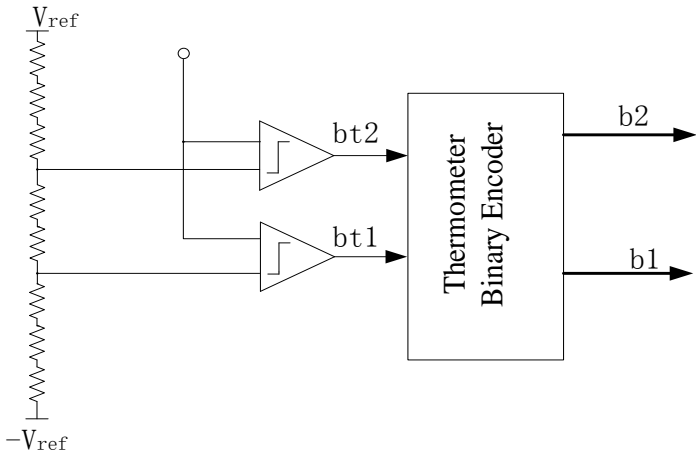


图 2.3 2 bit 的子 ADC

图 2.3 为一个 2bit 的子 ADC，它通过使用电阻串对参考电压进行分压来产生比较电压。图 2.3 将参考电压分为 V_1 和 V_2 ， V_1 和 V_2 分别对应着一个比较器，当模拟输入信号输入到子 ADC 后通过比较器分别与 V_1 和 V_2 比较，比较结果输入到编码器当中经过编码完成模拟信号向数字信号的转换。

子 ADC 理想情况下的量化结果如式 2.11 所示

$$b_1b_2 = \begin{cases} 00 & -V_{ref} \leq V_{in} \leq -0.25V_{reg} \\ 01 & -0.25V_{ref} \leq V_{in} \leq 0.25V_{reg} \\ 10 & 0.25V_{ref} \leq V_{in} \leq V_{reg} \end{cases} \quad (2.11)$$

(2)MDAC

图 2.4 为 MDAC 的原理图，图中 V_{in} 代表模拟输入电压， V_o 代表输出电压。MDAC 由子 DAC、运放放大器与下一级流水线子级采样保持电路组成，负责产生残差信号，并且将其放大，1.5bit/stage MDAC 的残差电压放大倍数为两倍。

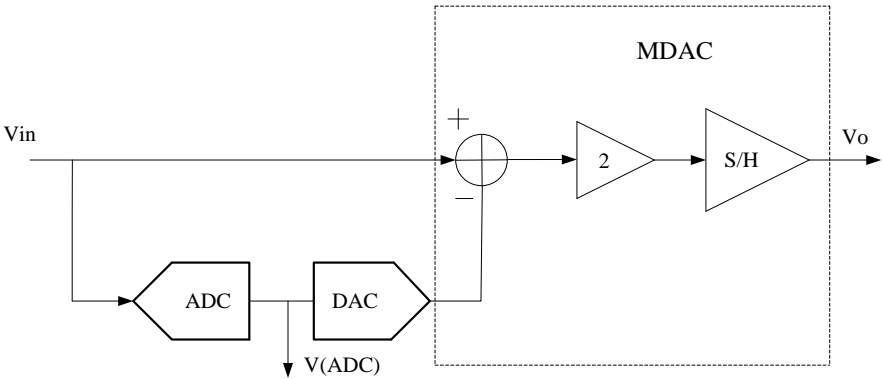


图 2.4 1.5bit MDAC 原理图

The diagram shows a two-stage CMOS op-amp. The first stage is a differential pair with NMOS transistors M_{n1} and M_{n2} and PMOS load transistors M_{p1} and M_{p2} . The gates of M_{n1} and M_{n2} are connected to a tail current source I_{SS} . The gates of M_{p1} and M_{p2} are connected to a common-mode feedback network consisting of a PMOS transistor M_{p3} and a resistor R_{CMF} . The output of the first stage is V_{out1} . The second stage is a common-source amplifier with an NMOS transistor M_{n3} and a PMOS load transistor M_{p3} . The gate of M_{n3} is connected to the output of the first stage, and the gate of M_{p3} is connected to a common-mode feedback network consisting of a PMOS transistor M_{p4} and a resistor R_{CMF} . The output of the second stage is V_{out2} .

图 2.5 中 V_{os} 代表失调电压, C_x 代表寄生电容, d 代表子 ADC 的输出, 当 Φ_1 闭合 Φ_2 打开时, C_1 与 C_2 对输入信号进行采样; 当 Φ_1 打开 Φ_2 闭合时, 由于 ADC 的数字输出信号 d 选择连接到 C_2 上的模拟电压, 完成数模转换的功能(这里 00、01 和 10 分别对应模拟输出电压是 V_{ref} 、0 与 $-V_{ref}$)。接下来根据电荷守恒, MDAC 会将采样的模拟输入电压与子 DAC 输出进行减法处理产生残差电压, 并且放大, 最后将残差电压传输到下一级流水线子级电路。

$$V_o = 2 \cdot V_{in} - (d-1) \cdot V_{ref} \quad (2.12)$$

现在常用的 MDAC 结构有 1.5bit/stage、2.5bit/stage 与 3.5bit/stage，图 8 为 2.5bit/stage 电路原理图，2.5bit/stage MDAC 电路结构与工作原理都与 1.5bit/stage MDAC 相似，都是在保持相完成模拟输入信号的采样和数模转换，在放大相产生擦差电压并放大，2.5bit/stage MDAC 在理想状态下的传递函数如式 2.13 所示

$$V_o = 4 \cdot V_{in} - (d - 1) \cdot V_{ref} \quad (2.13)$$

(3)前置采样保持电路

12

转型结构有更高的反馈因子，并且没有电容失配误差，一般应用于高速设计；电荷重分配结构适用于共模电平变化大的场合。

2.3 流水线 ADC 的误差分析与建模

2.3.1 流水线 ADC 的误差来源

(1) 热噪声

热噪声是 ADC 随机噪声的重要组成部分，它是由电子器件中电子的热运动引起的，热噪声主要来自电路中的电阻器件。在流水线 ADC 中，由于开关电容在时钟的控制下不断的进行充电和放电，这就相当于 MOS 电阻。MOS 电阻与对应的采样电容将热噪声引入系统中，它们产生的噪声功率谱如式 2.14 所示^[37]

$$S(f) = 4KTR \cdot \frac{1}{4\pi^2 R^2 C^2 f^2 + 1} \quad (2.14)$$

式 2.14 中 K 为波尔兹曼常量，T 为温度，R 为 MOS 电阻，C 为采样电容。ADC 的热噪声功率谱密度是均匀分布的，根据式 2.14 可得热噪声总功率，具体如 2.15 所示

$$P = \frac{K \cdot T}{C} \quad (2.15)$$

从式 2.15 可以看出热噪声与电容成反比。流水线 ADC 的模拟电路分为若干个流水线子级模块，所有流水线子级模块都有热噪声产生。这些噪声会一级一级的积累起来，其中第一级流水线子级的热噪声对系统影响最大，如式 2.16 所示

$$P_m = \frac{K \cdot T}{C_{s/h}} + \sum_{n=1}^m \frac{K \cdot T}{2^n \cdot C_n} \quad (2.16)$$

式 2.16 给出了流水线 ADC 中前置采样保持电路与所有流水线子级产生热噪声的计算公式，其中 $C_{s/h}$ 代表前置采样保持电路中的采样电容， C_n 代表第 n 级流水线子级电路中的采样电容。

由式 2.14 与 2.15 可知，采样电容越小，产生的热噪声就越多，但是如果使用较小的采样电容又容易造成电容较大的电容失配误差。使用数字校准技术设计流水线 ADC 可以校准电容失配误差，更好的解决上述的模拟电路设计存在的矛盾。

(2) 时钟抖动

流水线 ADC 的前置采样和级间采样都是由时钟沿控制，但是实际电路的非理想因素造成时钟沿到达的时刻不确定，所以产生了时钟抖动^[38]。如图 2.6 所示，系统对模拟输入信号正确的采样应在 t 时刻发生，但实际却是 t+Δt 时刻对模拟

信号进行采样，原来系统是理想的采样结果应该是 x ，但是因为时钟沿的不稳定采样结果变成 $x + \Delta x$ ，这样就形成了采样误差。

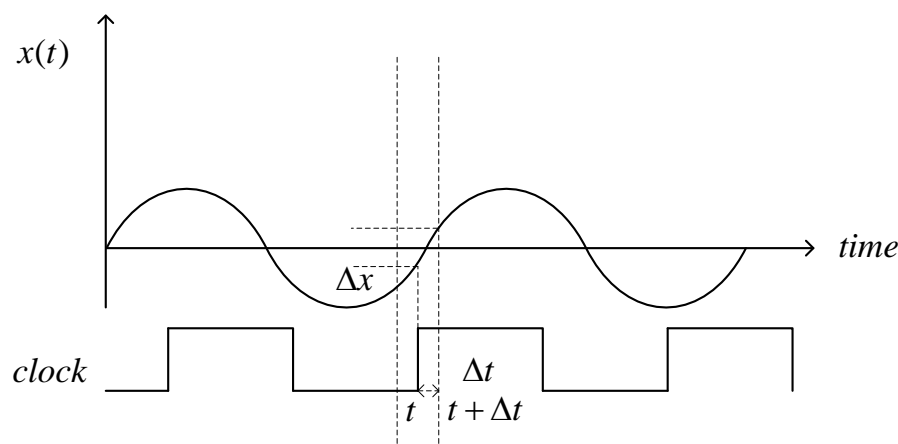


图 2.6 时钟抖动

假设有一个幅度为 A ，频率为 F 的正弦信号，那么采样保持电路对它进行采样时因为时钟抖动的所造成的误差如式 2.17 所示

$$\Delta X(t) = \Delta t \cdot \frac{d \sin(A \cdot 2\pi \cdot F)}{dt} \quad (2.17)$$

公式 2.17 表明时钟抖动误差主要由时钟频率和输入信号幅值决定，时钟抖动影响了 ADC 对输入信号幅值的判断，继而影响了 ADC 输出信号的 SNR^[39]。如式 2.18 所示

$$SNR = -20 \log(2 \cdot \pi \cdot F_{analog} \cdot \Delta t) \quad (2.18)$$

式 2.18 中 F_{analog} 代表模拟输入信号的频率，在实际情况中时钟抖动不能避免，也不能在电路中观察到，所以一般在流水线 ADC 中采用高性能的采样保持电路减少它的影响，或者在数字校准的补偿下去掉前置采样保持电路。

(3) 子 ADC 的误差

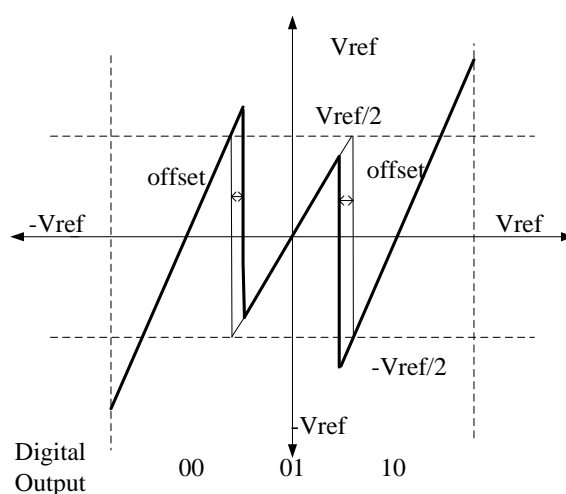


图 2.7 电压失调误差

上一节已经指出在流水线 ADC 的设计中,子 ADC 一般采用 Flash 结构。Flash ADC 的核心是电压比较器,比较器的参考电压是由电源电压和分压电阻串提供。如果分压电阻失配造成比较器的比较结果失误,那么就会影响子 ADC 的输出码字,从而影响残差电压的幅度,接下来很可能造成残差放大器的输出电压超出下一级流水线子级的输入电压范围。如图 2.7 所示,细线为理想的子 ADC 转换曲线,粗线为受参考电压失调影响的转换曲线。

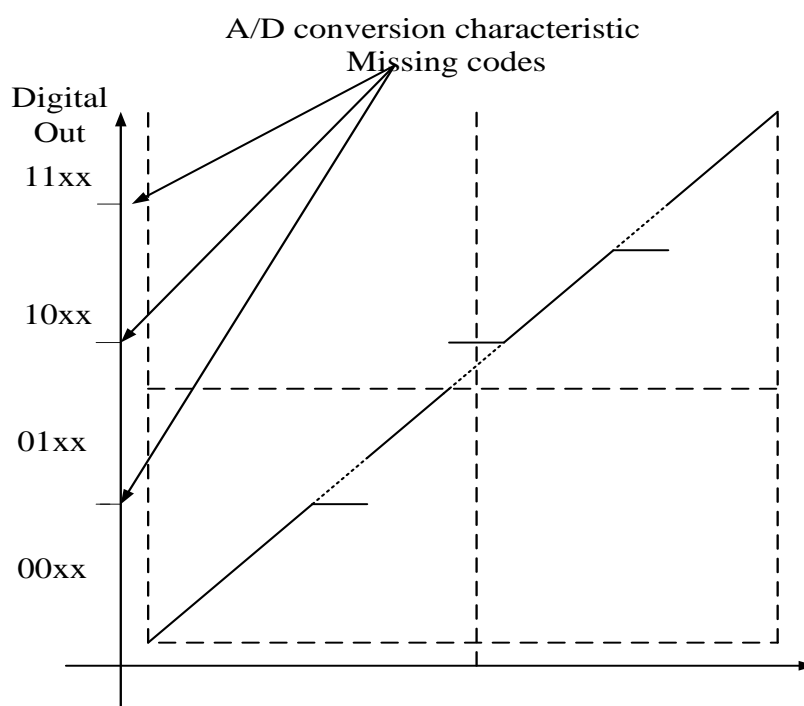


图 2.8 ADC 输出失码示意图

与此同时,因为比较器比较结果的失误,子 ADC 输出了错误的数字码,这可能导致整个流水线 ADC 的输出信号出现失码的情况,对 ADC 数字输出信号的线性度造成很大的影响,具体如图 2.8 所示。

如今在流水线 ADC 中一般使用冗余编码技术,这极大的减小了子 ADC 因为参考电压失调所造成的误差。例如在 1.5bit/stage 的 12bit 流水线 ADC 中,子 ADC 增加了 0.5bit 冗余位,假设在某一级流水线子级中出现了子 ADC 的参考电压失调,并产生了错误的数字码,那么后续的流水线子级与数字校准电路会将这个误差纠正,使流水线 ADC 输出正确数字信号。对于采用冗余编码技术的子 ADC,若比较器的参考电压失调在 $0.25V_{ref}$ 以内,那么失码将不会对输出造成影响,这大大减轻了子 ADC 的设计压力。

(4)MDAC 的误差

MDAC 是流水线 ADC 的核心电路,它的性能决定着整个流水线 ADC 的性能。图 2.5 给出了电容翻转式 1.5bit MDAC 的电路图。由于使用冗余编码技术设

计流水线 ADC，子 ADC 的误差被大大降低。所以现在减小 MDAC 的误差成为了进一步提高流水线 ADC 性能的关键。根据电量守恒 MDAC 的输入与输出关系如式 2.19 所示

$$V_{out} = \frac{(C_1/C_2 + 1) \cdot V_{in} - C_1/C_2 \cdot (d-1) \cdot V_{ref} + (C_1 + C_2 + C_x)/C_2(1+A) \cdot V_{os}}{C_1 + C_2 + C_x/C_2 \cdot A + 1} \quad (2.19)$$

电容失配是指采样电容与保持电容不相等，由式 2.19 可知电容失配会影响残差电压的放大倍数，同时也影响了输入输出曲线的线性度，这些误差会随着流水线子级传递而积累，最终造成失码和谐波失真。电容失配与电容面积的关系如式 2.20^[40]所示

$$capacitor_mismatch \propto \frac{1}{\sqrt{capacitor_area}} \quad (2.20)$$

由式 2.20 可知，提高电容匹配度的方法就是增大电容面积，例如使用单位电容阵列或者使用金属电容来提高两个电容的匹配度^[41]，但这无疑增加芯片面积提高了成本。若要兼顾芯片成本和 ADC 性能，那么只能将模拟电路的设计压力转移一部分至数字电路，使用数字校准计数设计流水线 ADC。

根据 MDAC 的传递函数，运放的直流增益 A 也是主要误差来源之一，由于 A 在实际电路中不可能无穷大，那么残差放大器也不可能实现理想的放大倍数，这样便形成了增益误差，具体如图 2.9 所示。

从图 2.9 可以看出，运放的增益误差最终造成了流水线 ADC 输出信号的失码，其中 $(1 - (A \cdot \beta)^{-1}) \cdot 2$ 代表运放的增益误差。

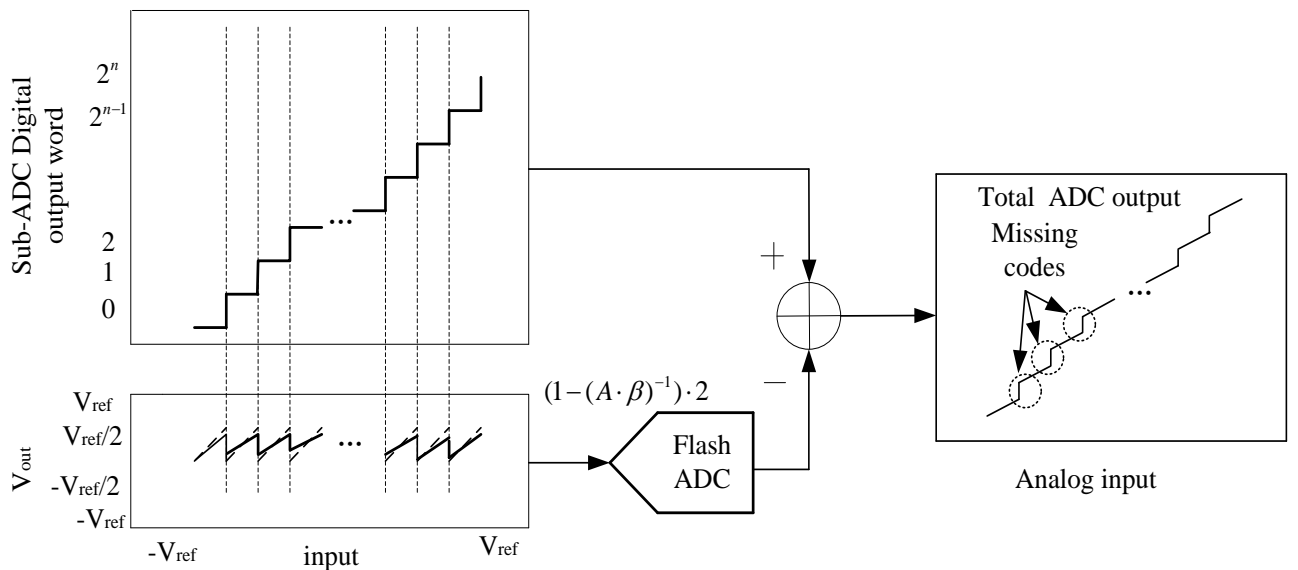


图 2.9 运放的误差

对于一个 n bit 的 MDAC，运放的有限直流增益带来的误差如下^[42]

$$\Delta_{input} \approx \frac{1}{A \cdot \beta \cdot 2^n} \quad (2.21)$$

式 2.21 中 A 为运放的直流增益，为反馈系数，为了不产生失码，那么 Δ_{input} 必需小于量化噪声。如式 2.22 所示

$$\frac{1}{2^N} > \frac{1}{A \cdot \beta \cdot 2^n} \quad (2.22)$$

即如式 2.23 所示

$$A > \frac{2^{N-n}}{\beta} \quad (2.23)$$

对于一个 12bit 1.5bit/stage 的流水线 ADC 的第一级流水线子级来说，等于 0.5，那么根据式 2.23 运放的增益至少需要达到 66dB，而且考虑到电容失配与后面的流水线子级的误差，实际上第一级流水线子级的运放需要更高的增益。在如今的 CMOS 工艺下，晶体三极管有源区的本征增益越来越低，这更加提高了高直流增益运放的设计难度。为了达到运放的性能要求，可以使用增益提高技术，或者采用多级运放结构^[43]，但同时会提高运算放大的功耗和复杂度。

另外运算放大器的输入输出曲线并非是非线性的，运放的增益会随着输入或输出变化，MDAC 实际的输入输出关系如式 2.24 所示

$$V_{in} = V_o \cdot a_1 + V_o^2 \cdot a_2 + \dots + (d-1) \cdot V_{ref} \cdot \frac{C_2}{C_1 + C_2} + V_{os} \cdot \frac{C_1 + C_2 + C_x}{C_1 + C_2} \quad (2.24)$$

式 2.24 中 a_n ($n=1,2,3,\dots$) 是与输出信号有关的系数，运算放大器的非线性增益影响了 ADC 的总谐波失真。

2.3.2 流水线 ADC 的建模

为了验证本文所研究的校准算法，必须建立流水线 ADC 的误差模型，根据上一节所介绍的流水线 ADC 的各种误差，在 matlab/simulink 平台上建立流水线 ADC 的误差模型：

(1) 前置采样保持电路的模型

前置采样保持电路的误差主要来自热噪声和时钟抖动，热造成和时钟抖动主要是影响模拟输入信号的幅值，所以在传输通路使用加法单元和白噪声源模拟这两种误差，如图 2.10 所示

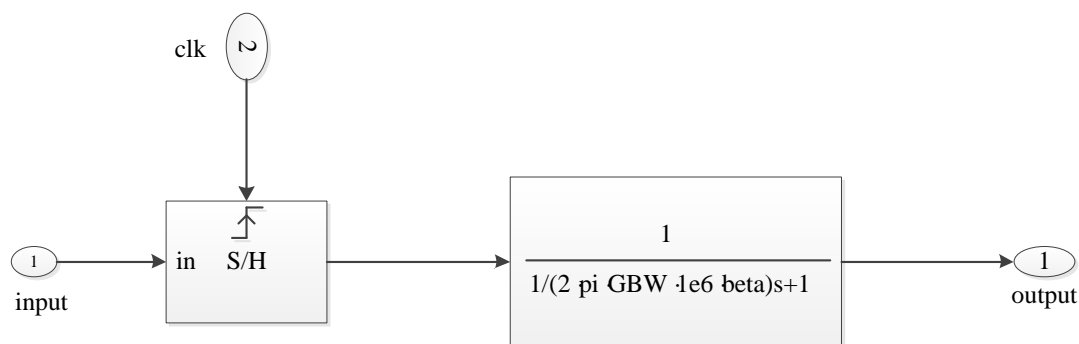


图 2.10 采样保持电路模型

前置采样保持电路中也存在运放的误差, 所以使用两个乘法单元模拟运放的增益误差和带宽误差。

增益误差系数为 $1 - (10^{-A/20})/\beta$; 带宽误差系数为 $1 / \{1 / [(2 \cdot \pi \cdot 1e6 \cdot GBW \cdot \beta) \cdot s + 1]\}$, 其中 β 代表反馈系数, GBW 代表带宽取 1000Mhz。

(2) 流水线子级的模型

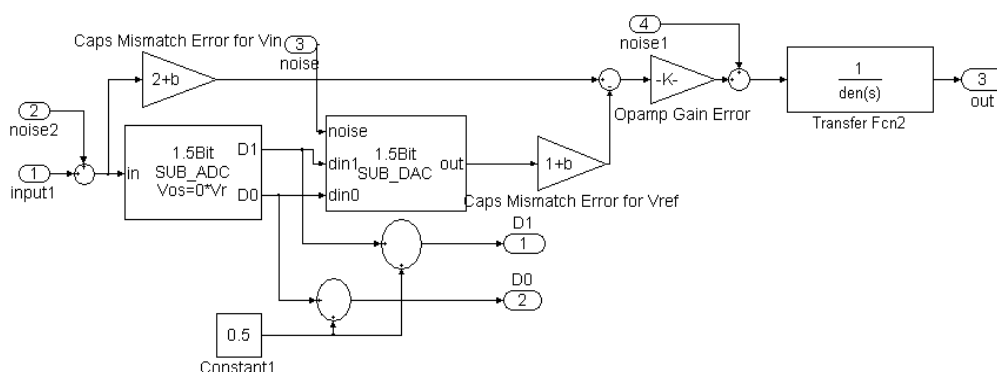


图 2.11 流水线子级模型

流水线子级包括子 ADC 与 MDAC 电路, 按照上一节的误差分析建立的 1.5bit 子级电路模型如 2.11 所示。

如采样保持电路一样, 模型使用白噪声源模拟各类随机噪声源, 其中模拟带宽误差的乘法单元与模拟增益误差的乘法单元, 与前置采样保持电路中对应的乘法单元一致, 另外使用乘法单元 $(2+b)$ 与 $(1+b)$ 模拟电容失配误差, b 代表电容的匹配度。这里 A 为 simulink 子系统封装后可自由设置的运放增益, 它的单位为 dB, β 为 MDAC 的反馈系数。

另外在子 ADC 使用乘法单元 $offset \cdot V_{ref}$ 模拟参考电压失调, $offset$ 为失调参数可自由设置, 在子 DAC 中使用加法单元 $V_{ref} + noise$ 模拟参考电压失调, $noise$ 为可自由设置的参数, Flash ADC 的模型建立比较简单在此不再赘述。

流水线 ADC 的输出由各流水线子级的数字输出码决定，如式 2.25 所示

$$D_{out} = d_1 \cdot 2^{n-1} + d_2 \cdot 2^{n-2} + d_3 \cdot 2^{n-3} + \dots + d_k \quad (2.25)$$

各流水线子级的输出并不同步，必须在后端增加延时单元，如图 2.12 所示

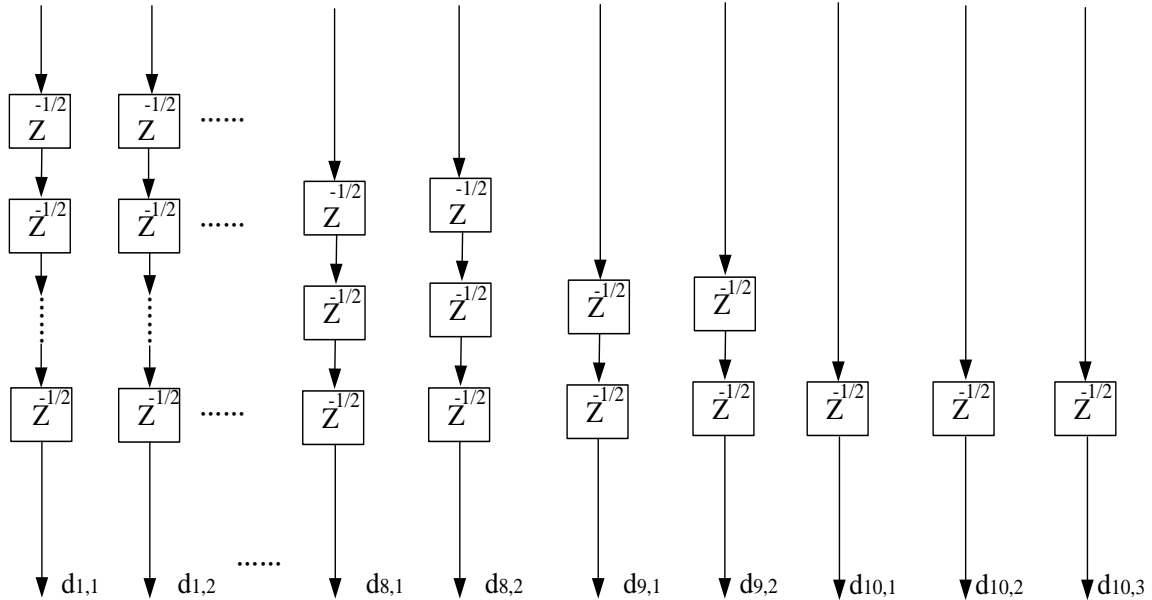


图 2.12 流水线后端延时单元

图 2.12 中每个流水线子级的数字码输出对应一个延时单元，其中第一级输出延时十个时钟，第二级输出延迟 9 个单元，最后一级 Flash ADC 延时一个单元，同时为保持数据的同步，所有延时单元的最后一个延时模块都由图 2.11 中的 CLK2 控制。数据经过延时单元同步以后进行错位相加模块，如图 2.13 所示

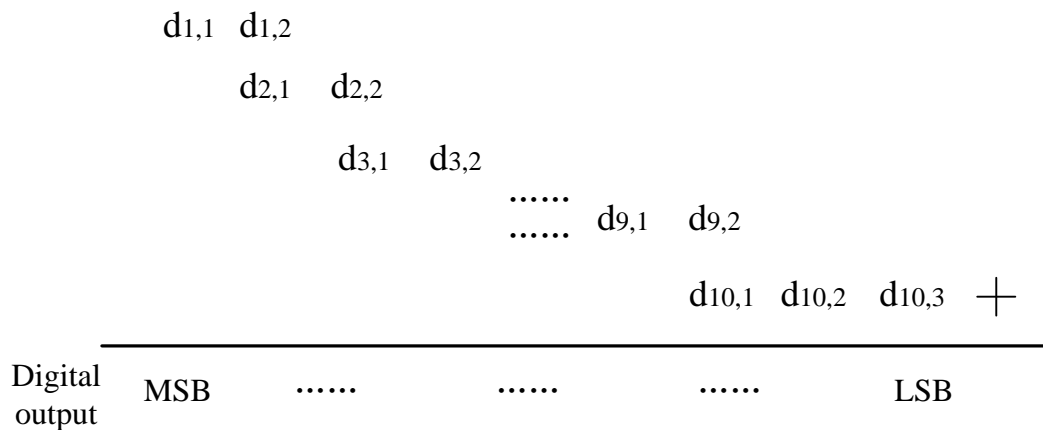


图 2.13 流水线后端错位相加单元

利用流水线子级模型与前置采样保持电路模型可建立整个 1.5bit/stage 12bit 流水线 ADC 模型，在流水线 ADC 的模型中设置 120M 的采样时钟，令为 0.5V，

给予它一个频率为 11.1475M，幅度为 0.495V 的正弦模拟输入信号，然后设置 0.05% 的电容失配，90dB 的运放增益，启动仿真。

在 matlab 中提取输出信号，并对按照相关采样原则输出信号做 FFT 分析得到图 2.14 所示的频谱图。

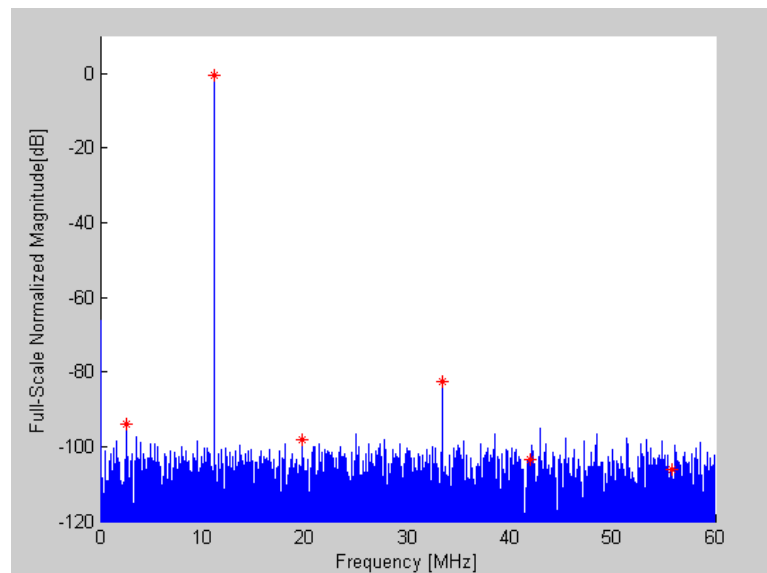


图 2.14 流水线 ADC 输出信号的频谱图

流水线 ADC 输出信号的静态参数如图 2.15 所示

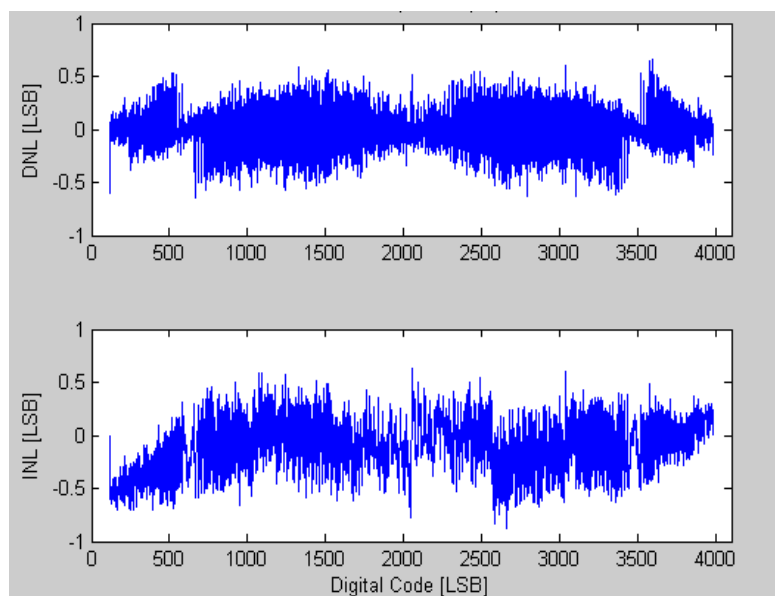


图 2.15 流水线 ADC 输出信号的 INL 与 DNL

ADC 的性能如表 2.1 所示

表 2.1 流水线 ADC 性能参数

F_s	120M	THD	-81.6652 dB
F_{in}	11.1475M	SNR	73.4978 dB
DNL	- 0.55/+0.63 LSB	SNDR	72.8814 dB
INL	-0.77/+0.52 LSB	ENOB	11.9232 Bit
SFDR	82.1422 dB		

2.4 基于 LMS 算法自适应滤波器原理与建模

适应滤波器是一种参数可调的特殊维纳滤波器,它由参数可调的数字滤波器与自适应算法组成。如图 2.16 所示,输入信号 $x(n)$ 经过参数可调的数字滤波器处理得到输出信号 $y(n)$,然后将 $y(n)$ 与期望信号 $d(n)$ 比较得到误差信号 $e(n)$,接下来自适应算法会根据 $e(n)$ 调整数字滤波器的参数使得输出信号 $y(n)$ 不断趋向于 $d(n)$,理论上最终可以使 $e(n)$ 为 0,令 $y(n)$ 完全等于 $d(n)$ 。

参数可调数字滤波器可以是有限长冲激响应(FIR)数字滤波器或无限长冲激响应(IIR)滤波器,但一般使用结构较为简单的 FIR 数字滤波器。自适应算法中应用最为广泛的是 LMS 算法与递归最小二乘算(LMS)法,其中 LMS 算法计算量小,容易硬件实现,且拥有较好的鲁棒性,所以本文使用基于 LMS 算法的自适应滤波器校准流水线 ADC。

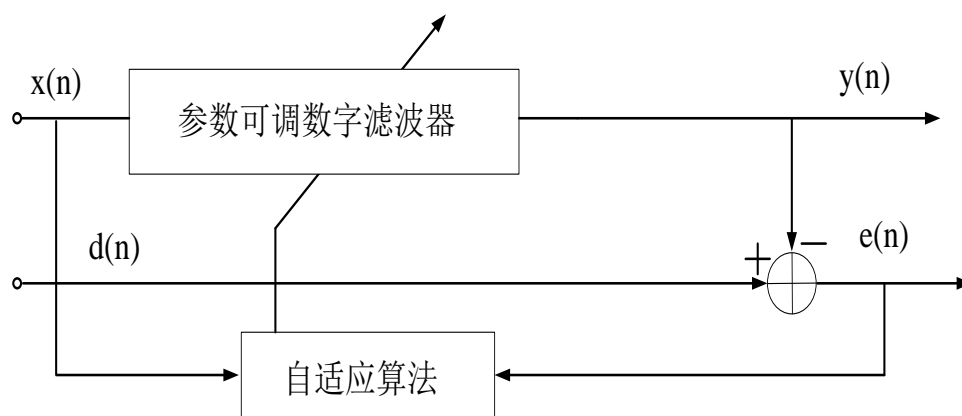


图 2.16 自适应滤波器原理图

LMS 算法由 Widrow 与 Hoff 于 1960 年提出,图 2.17 为基于 LMS 算法的自适应滤波器原理框图,输入信号为 $x(n)$,滤波器的阶数为 k , $W_1(n) \sim W_k(n)$ 为 k 阶 FIR 数字滤波器对应的抽头系数。输出信号如式 2.26 所示

$$y(n) = \sum_{m=1}^k W(n) \cdot x(n-k) \quad (2.26)$$

误差信号如式 2.27 所示

$$e(n) = d(n) - W^T(n) \cdot x(n) \quad (2.27)$$

定义均方误差因子为式 2.28 所示

$$\Delta J = E\{e^2(n)\} \quad (2.28)$$

由式 2.28 可得^[44]式 2.29

$$\begin{aligned} \Delta J &= E\{e^2(n)\} = E\{(d(n) - W^T \cdot x(n)) \cdot (d(n) - W^T \cdot x(n))\} \\ &= E\{d^2(n) + W^T \cdot E\{x(n) \cdot x^T(n)\} \cdot W - 2 \cdot E\{d(n) \cdot x^T(n)\} \cdot W\} \end{aligned} \quad (2.29)$$

若输入信号 $x(n)$ 与期望信号 $d(n)$ 都为平稳随机信号，可令

$R = E\{x(n) \cdot x^T(n)\}$ ， $P = E\{d(n) \cdot x(n)\}$ ，式 2.24 可改写为式 2.30。

$$\Delta J = E\{d^2(n)\} + W^T \cdot R \cdot W - 2 \cdot P^T \cdot W \quad (2.30)$$

为了求出使 ΔJ 获得最小值的一组抽头系数 $W(n)$ ，式 2.30 对 $W(n)$ 求导得到公式 2.31。

$$d(\Delta J)/dW = 2 \cdot R \cdot W - 2P \quad (2.31)$$

由式 2.31 可知，使误差因子获得最小值的这一组抽头系数也称作最优维纳解，但求逆矩阵在硬件中较难实现，且运算量太大，所以采用最速下降法优化式 2.31，优化后如式 2.32 所示

$$W(n+1) = W(n) + u \cdot x(n) \cdot e(n) \quad (2.32)$$

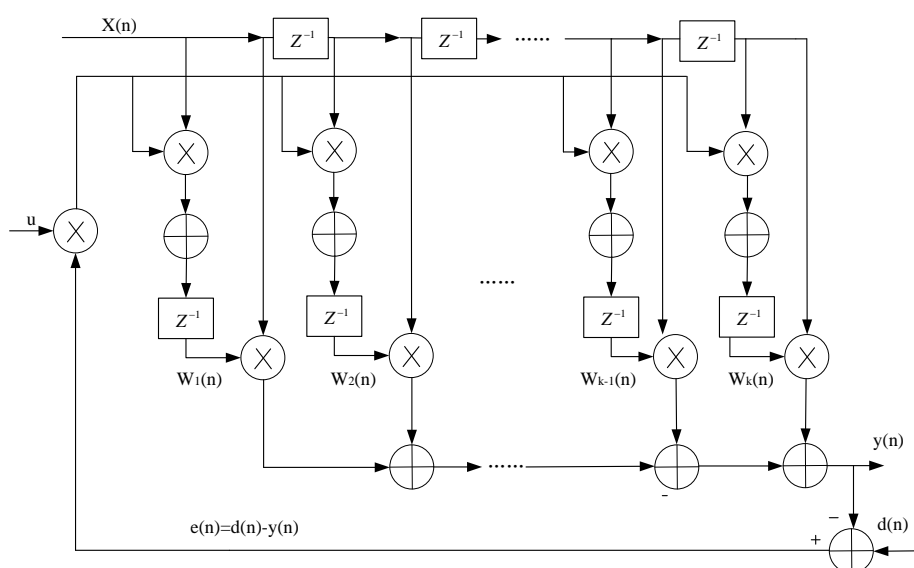


图 2.17 基于 LMS 算法的自适应滤波器原理图

为了验证基于 LMS 算法的自适应滤波器的基本功能，在 matlab 平台上给予 32 阶自适应滤波器一个理想的正弦波作为期望信号，令正弦波与噪声组合成自适应滤波器的输入信号，设置不同的步长因子进行仿真。

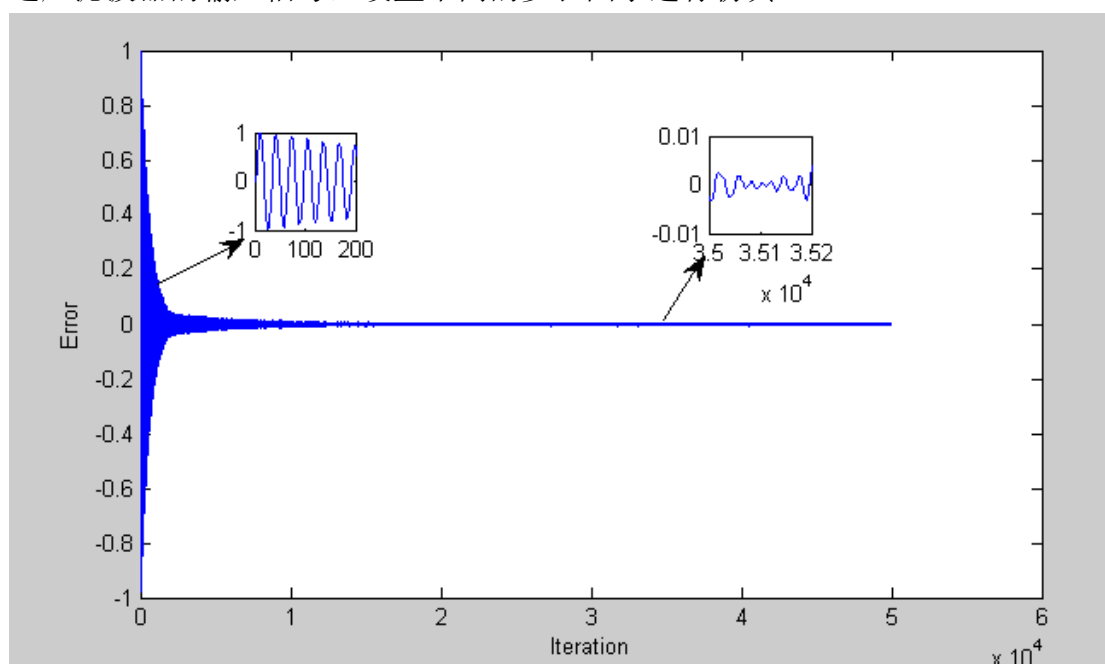


图 2.18 误差曲线

图 2.18 为误差曲线，它是自适应输出与期望信号作差的结果，从图中可以看出，误差由大变小，最后误差的取值稳定在一个区间之类，这表示自适应滤波器的纠正误差能力已经到极限，这个系统收敛了。

2.5 本章总结

本章首先介绍流水线 ADC 的性能指标和工作原理，并分析了流水线 ADC 的模拟电路结构和相对应误差，通过数学公式和图形展现了各类误差对流 ADC 的影响。本章还在 Matlab/simulink 平台上建立流水线 ADC 的模型，为以后验证校准算法打下基础。最后，本章介绍了基于 LMS 算法自适应滤波器的基本原理，并且通过仿真展示了自适应滤波器的基本功能。

第3章 基于参考 ADC 的流水线 ADC 校准算法

本章首先根据电路原理和 MDAC 的传递函数介绍整个流水线 ADC 的传递函数，然后结合自适应滤波器介绍使用参考 ADC 校准流水线 ADC 的基本理论。另外，本章还在 Matlab/Simulink 平台上利用建立的流水线 ADC 模型验证算法的校准效果，并根据变步长 LMS 算法理论对校准算法进行优化。

3.1 基于参考 ADC 的校准原理

第2章已经指出，目前流水线 ADC 的设计难点在于 MDAC 的设计，下面从 MDAC 的电路结构和误差出发介绍本文所研究的校准算法。

根据图 2.15 1.5bit/stage 电容翻转型 MDAC 的电路结构可得到式 3.1。

$$V_{in} \cdot (C_1 + C_3) = V_o \cdot \left(\frac{C_1 + (C_1 + C_2 + C_x)}{A} \right) + V_{ref} \cdot (d-1) \cdot C_2 - V_{os} (C_1 + C_2 + C_x) \quad (3.1)$$

式 3.1 为流水线 ADC 子级的模拟域传递函数，与为流水线子级的模拟输入电压与输出电压，A 为运放的直流增益， V_{os} 为失调电压。校准的流水线 ADC 为 12bit，等号两边同时除以 $V_{ref}/2048$ 后，可得式 3.2。

$$D_i = D_o \cdot \frac{(C_1 + \frac{C_1 + C_2 + C_x}{A})}{C_1 + C_2} + 2048 \cdot (d-1) \cdot \frac{C_2}{C_1 + C_2} - D_{os} \cdot \left(\frac{C_1 + C_2 + C_x}{C_1 + C_2} \right) \quad (3.2)$$

式 3.2 就是流水线子级传递函数，但是式中所有模拟电压量转换成了数字电压量，d 为子 ADC 的输出信号， D_i 为模拟输入信号对应数字码， D_o 为流水线子级输出信号对应的数字码， D_{os} 为失调电压的数字码，简化式 3.2 可得式 3.3。

$$D_i = D_o \cdot a + 2048 \cdot (d-1) \cdot b + D_{os} \cdot c \quad (3.3)$$

$$\text{在式 3.3 中 } a = \left(\frac{c_1}{c_1 + c_2} \right) + \frac{c_1 + c_2 + c_x}{A \cdot (c_2 + c_1)}, \quad b = \frac{c_2}{c_1 + c_2}, \quad c = \frac{c_1 + c_2 + c_x}{c_1 + c_2};$$

根据式 3.3，若模拟电路处于理想状态，则 a 等于 0.5，b 等于 0.5，c 等于 1，如式 3.4 所示

$$D_o = 2 \cdot D_i + (d-1) \cdot 2048 \quad (3.4)$$

式 3.4 只是将式 2.11 中模拟电压量转换为数字电压量，但因为模拟电路的各种非理想因素使得参数 a，b 和 c 偏离了理想值，继而影响了流水线子级的输入和输出曲线，基于参考 ADC 的校准算法将参数 D_o ，d 和 V_{os} 作为校准模块输入信

号，通过三个基于 LMS 算法的自适应滤波器来校准流水线 ADC 子级模拟电路中存在的电容失配、运放误差和电压失调误差。

对于整个 12bit 流水线 ADC，它有九个 1.5bit 流水线子级和一个 3bit Flash ADC 组成，每一级流水线子级的模拟电路都是相同的，所以可得到式 3.5。

$$\begin{aligned} D_{i,1} &= D_{o,1} \cdot a_1 + 2048 \cdot (d_1 - 1) \cdot b_1 + D_{os,1} \cdot c_1 \\ D_{i,2} &= D_{o,2} \cdot a_2 + 2048 \cdot (d_2 - 1) \cdot b_2 + D_{os,2} \cdot c_2 \\ &\vdots \\ D_{i,9} &= D_{o,9} \cdot a_9 + 2048 \cdot (d_9 - 1) \cdot b_9 + D_{os,9} \cdot c_9 \end{aligned} \quad (3.5)$$

式 3.5 中，第九级流水线子级的模拟输出信号作为 Flash ADC 的输入信号，并由 Flash ADC 采样量化得到的数字输出信号，式 3.5 中所有流水线子级的参数 a , b , c 在理想状态下都是相等的，如式 3.6 所示

$$a_1 = a_2 = a_3 = \dots = a_8 = a_9 \quad (3.6)$$

流水线 ADC 所有流水线子级都是串行排列，输入信号会流经每一个流水线子级，所以每一级流水线 ADC 的模拟输出信号就是下一级流水线子级的模拟输入信号，如式 3.7 所示

$$D_{i,n+1} = D_{o,i} \quad (3.7)$$

根据这个递归关系可得到整个流水线 ADC 在数字信号域的传递函数，如式 3.8 所示

$$D_{input} = D_{i,1} = \{[\dots] \cdot a_2 + 2048 \cdot (d_2 - 1) \cdot b_2 - V_{os} \cdot c_2\} \cdot a_1 + 2048 \cdot (d_1 - 1) \cdot b_1 + D_{os,1} \cdot c_1 \quad (3.8)$$

式 3.8 中，由于第一级流水线子级的输入信号等于整个流水线 ADC 的输入信号，由前置采样保持电路对模拟信号采样所得，所以有 $D_{input} = D_{i,1}$ ， D_{input} 是流水线 ADC 模拟输入信号对应的数字信号。

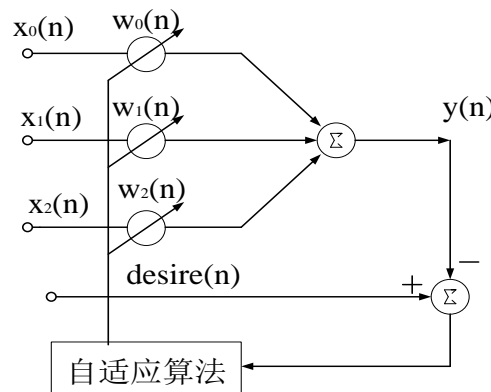


图 3.1 三输入自适应滤波器

式 3.8 定量的描述了模拟电路的非理想因素对流水线 ADC 输入输出曲线的影响,同时公式 3.3 与多输入自适应滤波器的传递函数相似,三输入自适应滤波器的结构如图 3.1 所示。

图 3.1 为三输入自适应滤波器, $W_0(n)$ 、 $W_1(n)$ 与 $W_2(n)$ 为输入信号, $x_0(n)$ 、 $x_1(n)$ 与 $x_2(n)$ 对应的抽头系数, $\text{desire}(n)$ 为期望信号, 这个三输入自适应滤波器的传递函数如式 3.9 所示

$$y(n) = x_0 \cdot w_1(n) + x_1 \cdot w_2(n) + x_2 \cdot w_3(n) \quad (3.9)$$

式 3.9 与式 3.3 类似, 根据式 3.3 将流水线子级的输入输出信号作为三输入自适应滤波器的输入信号, 具体如式 3.10 所示

$$D_i = D_o \cdot w_1 + (d - 1) \cdot w_2 + D_{os} \cdot w_3 \quad (3.10)$$

式 3.10 中 w_1 、 w_2 与 w_3 分别代表三输入自适应滤波器三个抽头系数。根据流水线子级之间的递归关系, 将每一个流水线子级的输出都用一个三输入的自适应滤波器处理, 这样可以得到整个校准系统的传递函数, 如式 3.11 所示

$$D_{input} = D_{i,1} = \{[\dots] \cdot w_{2,1} + 2048 \cdot (d_2 - 1) \cdot w_{2,2} + D_{os,2} \cdot w_{2,3}\} \cdot w_{1,1} + 2048 \cdot (d_1 - 1) \cdot w_{1,2} + D_{os,1} \cdot w_{1,3} \quad (3.11)$$

根据式 3.11, 后端可以使用自适应滤波器拟合流水线 ADC 的传递函数。如果外界提供一个高精度的基准信号, 将流水线 ADC 的输出与基准信号比较得到误差因子 $e(n)$, 这里 $e(n)$ 是模拟电路的非理想因素造成的, 从公式 3.8 当中来看, $e(n)$ 是由于各流水线子级传递函数中的三个参数(a、b 和 c)偏离理想值所造成的, $e(n)$ 可以根据 LMS 算法中的抽头系数迭代公式驱动式 3.11 中所有抽头系数变化, 使得流水线 ADC 的输出信号与基准信号的差别越来越小, 这样 ADC 的输出精度就得到了提高, 根据上述思路可得整个校准系统框图如图 3.2 所示

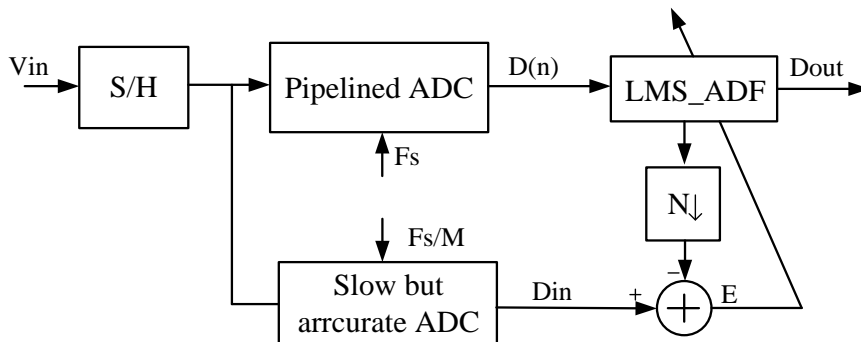


图 3.2 校准原理

图 3.2 中 $D(n)$ 是流水线 ADC 未经校准的输入信号, 它由九个流水线子级中子 ADC 的输出与 Flash ADC 的输出组成, D_{in} 是一个低频率高精度参考 ADC 的输出, LMS_ADF 模块是九个子级校准模块, D_{out} 为校准输出信号。每一个子级

校准模块的输出就是上一级子级校准模块的输入。给予两个 ADC 同样的模拟输入信号 V_{in} ，为了输入信号同步两个 ADC 必须共用一个采样保持电路，另外由于 D_{out} 的频率比 D_{in} 大，所以在提取误差因子 E 之间 D_{out} 需要经过降频处理。

基于参考 ADC 与自适应滤波器的校准方法是建立公式 3.3 的基础上的，但公式 3.3 没有考虑运放的非线性误差，如今人们对流水线 ADC 的性能越来越高，这就要求一种优异的校准算法不仅能校准线性误差还能校准非线性误差，下文介绍如何用参考 ADC 法校准运放的非线性误差。

公式 2.21 比公式 2.11 多了输出信号的高次项，按照这一节介绍的校准理论，只需在原来的流水线子级校准模块中增加一阶自适应滤波器的个数，如公式 3.12 所示。图 3.3 给出了 MDAC 非线性误差校准系统的组成结构。

$$D_i = D_0 \cdot w_1 + 2048 \cdot (d-1) \cdot w_2 + V_{os} \cdot w_3 + D_0^2 \cdot w_4 + D_0^3 \cdot w_5 \quad (3.12)$$

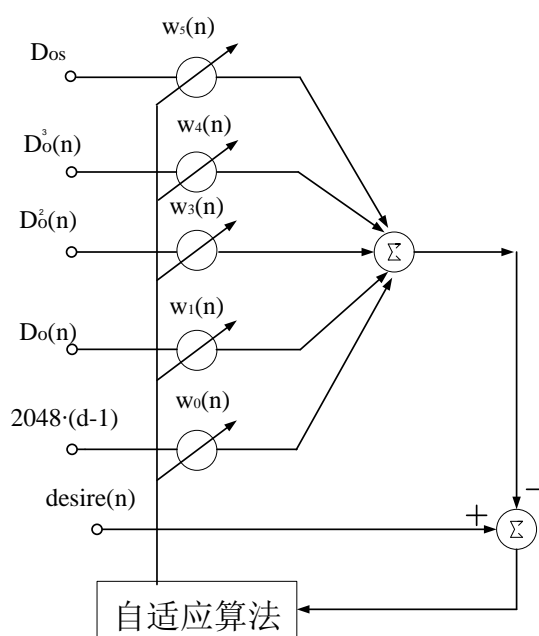


图 3.3 MDAC 非线性误差校准结构

这一节描述了基于参考 ADC 的 1.5bit/stage 流水线 ADC 的校准思路，这类校准算法是建立在 MDAC 的电路结构上的，1.5bit/stage 与多比特 MDAC 的电路结构和工作原理相似，基于参考 ADC 与自适应滤波的校准算法适用于各种 MDAC 结构。

3.2 校准算法的建模与仿真

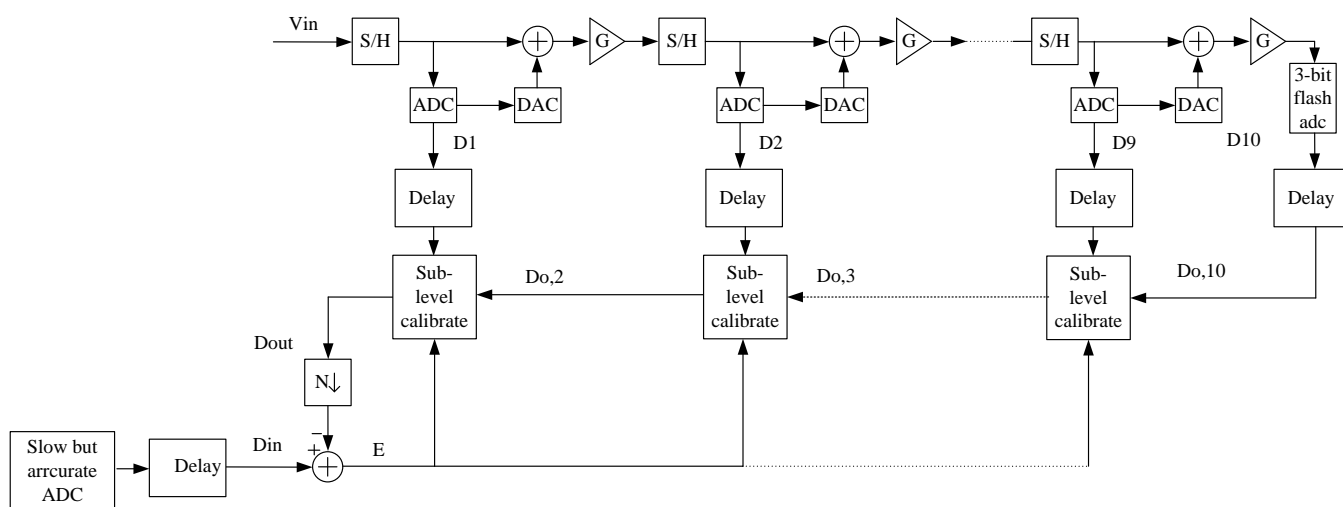


图 3.4 带校准系统的流水线 ADC 原理图

上一节中详细描述了基于参考 ADC 校准流水线 ADC 的基本理论，本节将在 Matlab/Simulink 平台上建立校准算法的模型，通过比较校准前后 ADC 的性能参数检测校准算法。

流水线 ADC 后端是延时模块与错位相加模块，校准系统与流水线 ADC 拼接以后，需要删除错位相加模块，带校准的流水线 ADC 的原理图如图 3.4 所示，图 3.4 中 D1~D10 代表流水线子级的输出， $D_{o,2} \sim D_{o,10}$ 代表流水线子级校准模块的输出， D_{out} 代表经过校准的流水线 ADC 输出信号，Din 代表期望信号，流水线子级校准的模型如图 3.5 所示

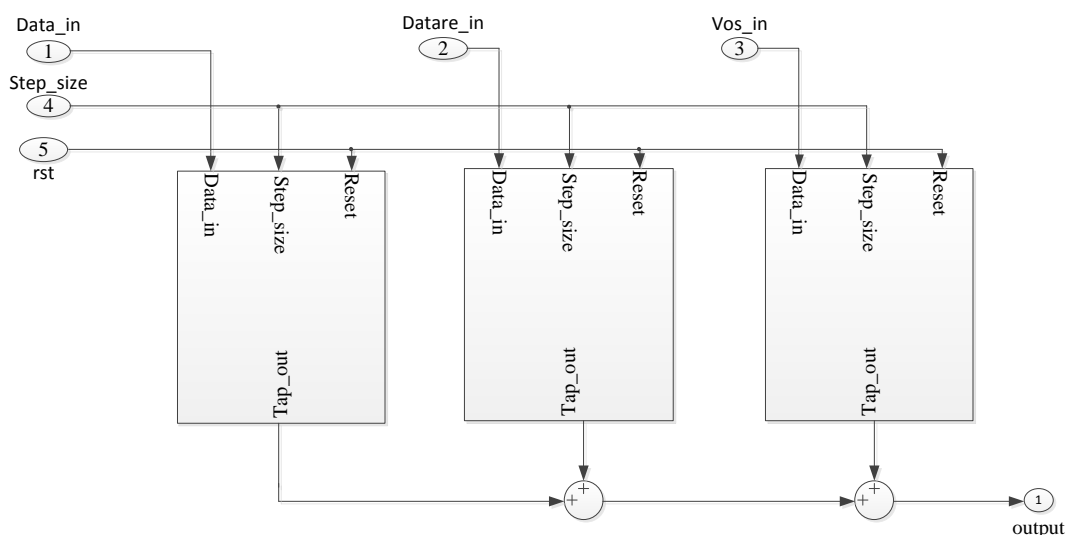


图 3.5 流水线 ADC 子级校准模块

仿真之前按照表 3.1 设置流水线 ADC 的误差，另外在每级采样保持电路模型、子 DAC 和子 ADC 处引入相应的白噪声，模拟热噪声等其他误差，同时将流水

线 ADC 的时钟设置为 120M，输入信号设置为 11.1475M，参考 ADC 的时钟为流水线 ADC 的十八分之一。启动仿真以后，按照相关采样原则采集 571 个周期 4096 个输出码作 FFT 分析。去 3 个输出信号保证计算 DNL/INL 时，测试精度以 95%的可信度控制在 0.1LSB 中。这里没有引入运算放大器的非线性误差，所以使用线性校准算法。

表 3.1 流水线 ADC 模型参数

参数	1 到 3 级设置值	参数	4 到 9 级设置值
电容失配	10%	电容失配	5%
运放增益	60dB	运放增益	60dB
参考电压失调	0.01	参考电压失调	0.005
寄生电容	$0.1 \cdot C_1$	寄生电容	$0.1 \cdot C_1$

图 3.6 给出了线性校准模块误差曲线，误差曲线在校准开始时取值较大，抽头系数变化较快，随着校准的进行误差信号逐渐变小，抽头系数取值趋于平稳，这代表经过校准模块的 ADC 信号精度得到较大提高。

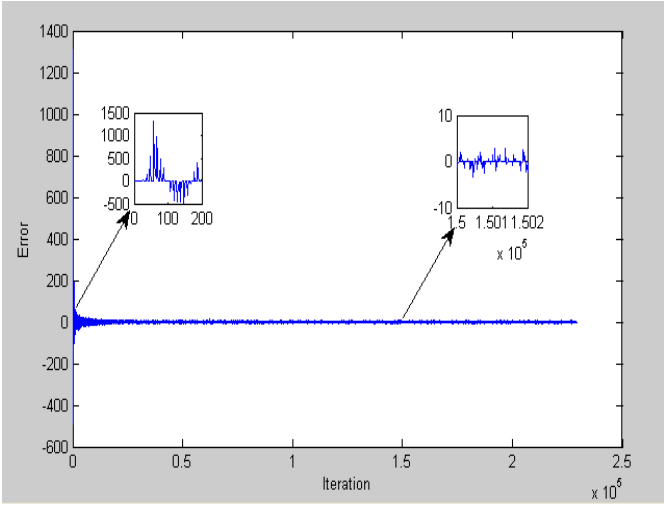


图 3.6 误差曲线

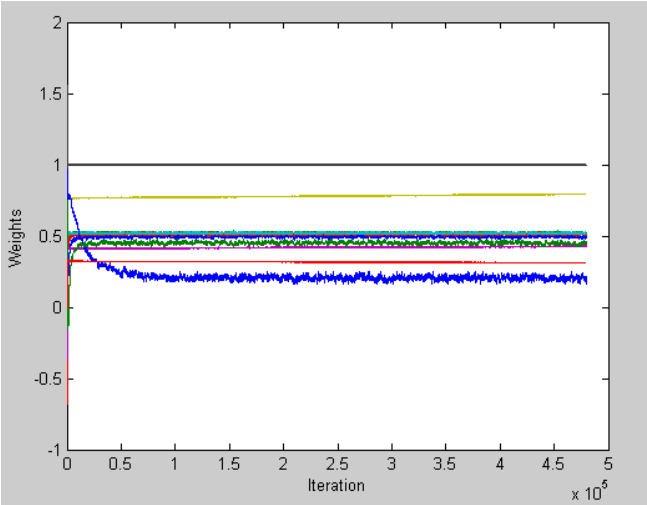


图 3.7 抽头系数迭代曲线

图 3.7 给出了校准系统抽头系数的迭代曲线，所有曲线在仿真的后期都趋向与稳定，这代表校准收敛，大概经过 5000 次迭代校准系统的误差取值稳定在一定范围内。图 3.8 与图 3.9 给出了 ADC 输出信号在未经校准和校准两种情况下的频谱图，从图中可以看过经过校准的 ADC 信号的噪底得到显著的降低，谐波噪声也得到了大幅度的降低。

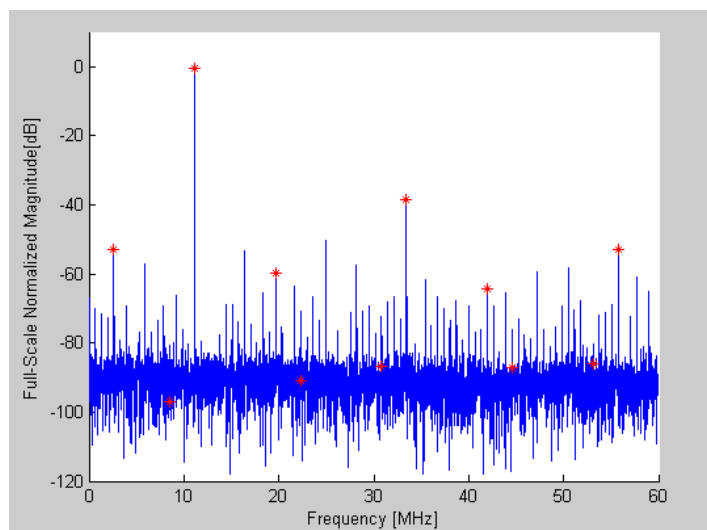


图 3.8 未经校准的流水线 ADC 输出频谱图

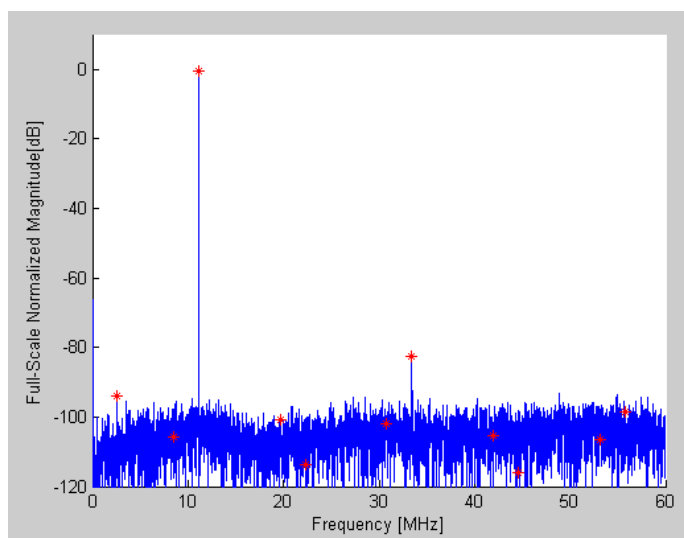


图 3.9 经过校准的流水线 ADC 输出频谱图

表 3.2 给出了 ADC 输出信号在未经校准与校准两种情况下的 FFT 分析结果；从表 3.2 中可以看出，经过校准的 ADC 信号的动态参数全部得到有效的提高，其中 ENOB 从 7.094bit 提高到 10.810bit。

表 3.2 FFT 分析结果

性能参数	未经校准	经过校准	性能参数	未经校准	经过校准
SNR	44.423dB	66.797 dB	ENOB	7.094 Bit	10.810Bit
THD	-37.853 dB	-80.972 dB	SNDR	36.989dB	66.634 dB
SFDR	38.202 dB	81.702dB			

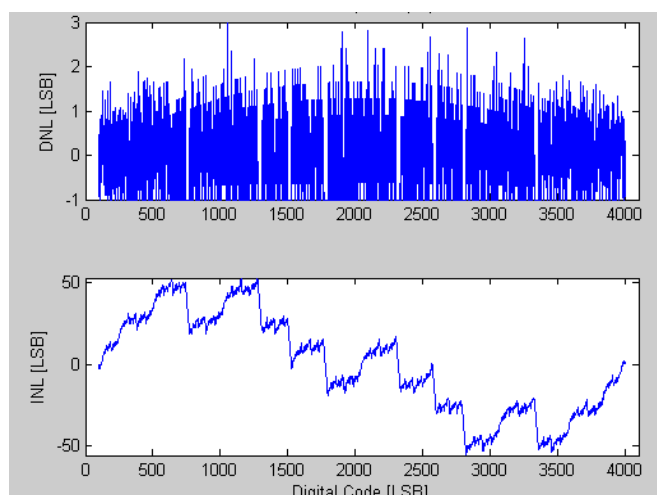


图 3.10 未经校准的流水线 ADC 静态参数

图 3.10 与图 3.11 给出了 ADC 输出信号在未经校准和校准两种情况下的 DNL/INL 统计情况，从图中可以看出，经过校准 ADC 的静态参数也得到提高，其中 DNL 由 $-1/+3$ LSB 提高到 $-0.77/+0.71$ LSB, INL 由 $-55/+53$ LSB 提高到 $-0.89/+0.95$ LSB。上述结果充分说明基于参考 ADC 的校准方法对于电容失配、运放的有限直流增益和失调电压等误差有一定校准效果。

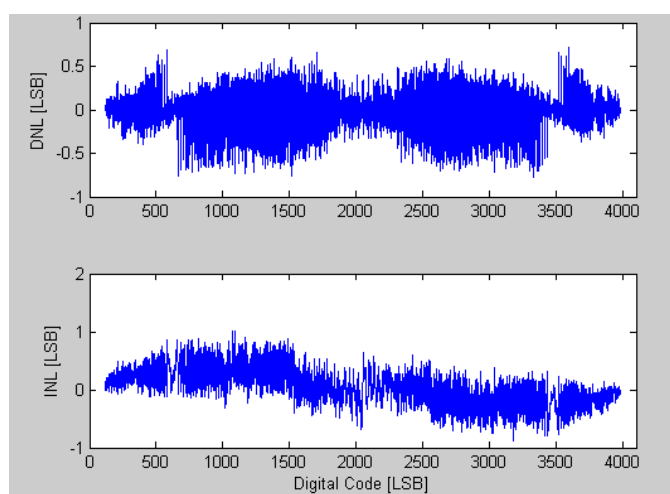


图 3.11 经过校准的流水线 ADC 静态参数

由表 3.1 可知，流水线 ADC 模型没有加入运放的非线性误差，所以在模块中没有使用非线性校准模块。运放的非线性误差是指运放的直流增益随输入或者输出变化，为了模拟运放的非线性误差，令每一级的运放直流增益 A 在工作时随输入波动，如式 3.13 所示

$$A_{real} = A_{ideal}(1 + V_i \cdot \alpha_1 + V_i^2 \cdot \alpha_2 + V_i^3 \cdot \alpha_3) \quad (3.13)$$

式 3.13 给出了 MDAC 中运算放大器的直流增益与输入信号之间的关系，其中 A_{ideal} 为基础增益，设置为 60dB， α_1 、 α_2 和 α_3 为输入电压一次项到三次项的系数。按照式 3.13 在原有的流水线 ADC 模型的基础上增加运放增益产生模块，这样整个流水线 ADC 误差模型就具备了非线性误差。图 3.12 给出了第一级流水线子级中运放的直流增益随输入变化的曲线。

在引入线性误差模型以后，依然按照表 3.1 设置电容失配等误差，同时给予与上次仿真同样的输入信号来启动仿真。

表 3.3 仿真参数

性能参数	未经校准的参数	经过线性校准的参数	经过非线性校准的参数
SNR	38.195 dB	63.244 dB	66.515 dB
SNDR	34.983 dB	63.166dB	66.371 dB
THD	-39.632 dB	-80.651dB	-81.245dB
SFDR	40.540 dB	82.010 dB	82.208dB
ENOB	6.059Bit	10.220Bit	10.763 Bit

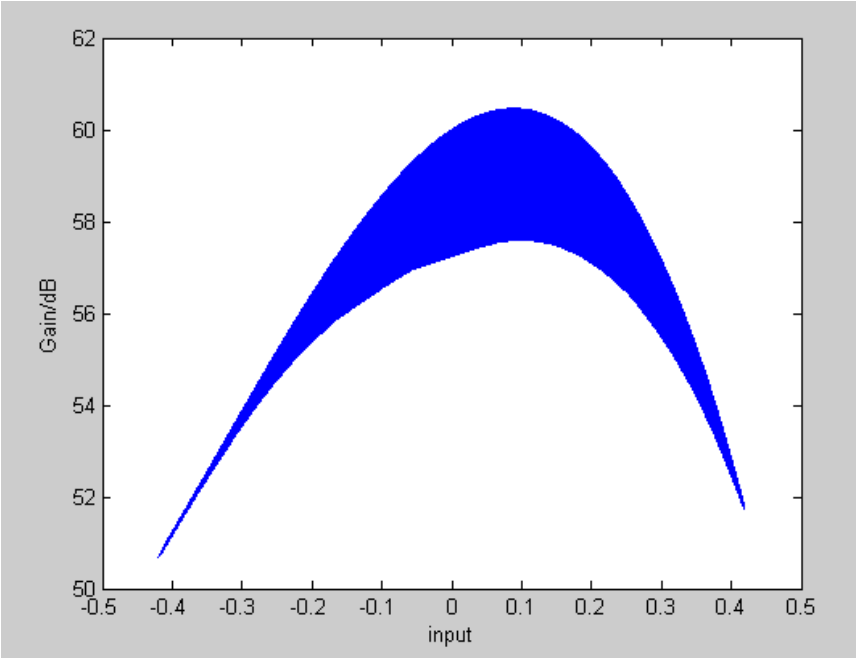


图 3.12 增益变化曲线

表 3.3 第二列给出了未经任何校准的 ADC 信号动态参数取值，没有带校准系统的流水线 ADC 在引入运算放大器的非线性误差以后性能更低；表 3.3 第三列给出了 ADC 信号经过线性校准模型校准后的动态参数，这里信号的动态性得到有效的提高，但线性校准模型是根据式 3.4 设计，没有如式 3.12 针对 MDAC 的输入输出曲线的非线性引入高次项，所以对非线性误差不敏感；表 3.3 第四列给出了 ADC 信号经过非线性校准模块校准后的动态参数，非线性校准模块是以式 3.12 为基础，引入了输出信号的二次项和三次项，可以校准 MDAC 中运算放大器的非线性误差，所以与第三列相比，经过非线性校准模块的动态性能得到进一步的提高。

与此同时，表 3.4 给出了三种情况下静态参数的分析结果，DNL 与 INL 情况基本与动态参数的结果一样，经过非线性校准模块校准的信号静态参数也稍优于经过线性校准模块校准的信号。

表 3.4 静态参数分析

静态参数	未经校准的参数	经过线性校准的参数	经过非线性校准的参数
DNL	- 1/+2.7 LSB	-0.86/+0.78 LSB	-0.78/+0.73 LSB
INL	-64/+65 LSB	-1.06/+0.98 LSB	-0.89/+0.95 LSB

3.3 校准算法的改进

基于参考 ADC 的流水线校准算法，首先是依靠低速高精度的流水线 ADC 提供期望信号，然后利用期望信号与校准输出做差提取误差因子，这样就能实时的监控流水线 ADC 的输出信号，自动的提高输出信号的精度。同时，校准算法是利用基于 LMS 算法的自适应滤波器减小误差因子，并且根据误差因子的大小校准流水线 AD。所以，基于 LMS 算法的自适应滤波器的收敛速度和收敛精度就代表校准算法的校准速度和校准精度。

LMS 算法校准速度和校准精度取决于抽头系数趋向于最优维纳解的速度与精度。上一节已经指出，自适应滤波器在实际应用中难以直接得到最优维纳解，在实际应用中 LMS 算法采用最速下降法更新抽头系数，所以要提高校准算法的性能就必须分析最速下降法的工作特点和性能特点。

最速下降法的工作状态由抽头系数的初始值、自相关矩阵 \mathbf{R} 和步长因子 μ 决定。若抽头系数的初始值设置得当，则可以减少抽头系数的收敛时间，但实际中则不好把握初始值，所以一般设置为零。实际上最速下降法的性能和状态最终是由输入信号的自相关矩阵 \mathbf{R} 与步长因子 μ 决定。

为了说明输入信号的自相关矩阵 \mathbf{R} 与步长因子 u 对算法的影响, 令^[45-46]

$$Error(k) = w(k) - w_o \quad (3.14)$$

抽头系数在理想情况下会无限趋向于最优维纳解 w_o , 所以如式 3.15 所示

$$\lim_{n \rightarrow \infty} E(Error(k)) = 0 \quad (3.15)$$

式 3.15 中 $E(Error(k))$ 代表 $Error(k)$ 的均值, 根据自适应滤波器理论可得式 3.16。

$$E(e(k) \cdot x(k)) = \mathbf{R} \cdot E(w(k)) - \mathbf{P} \quad (3.16)$$

按照式 3.14、式 3.15 和式 3.16 可得式 3.17。

$$\begin{aligned} E(Error(k+1)) &= E(w(k+1) - w_o) = E(w(k) + u \cdot e(k) \cdot x(k) - w_o) \\ &= (I - u \cdot \mathbf{R}) \cdot E(error(k)) \end{aligned} \quad (3.17)$$

式 3.17 给出了抽头系数迭代计算的另外一种表达式, 若抽头系数要收敛于最优维纳解则如式 3.18 所示

$$-1 < I - u \cdot \mathbf{R} < 1 \quad (3.18)$$

从式 3.18 可知当时 $0 < u < 2/\gamma_{\max}$, 系统才会收敛, 这里为矩阵 \mathbf{R} 的最大特征值。从校准系统来看, 只有设置的步长因子 u 满足这个条件, 校准模块才可以产生校准效果。

另外公式 3.18 可以看出, 随着迭代的进行, 抽头系数与最优维纳解的差越来越小, 所以可以根据迭代次数来估计收敛的时间, 如式 3.19^[47]所示

$$\sigma_T \approx \frac{1}{u \cdot \gamma_m} \quad (3.19)$$

公式 3.19 给出了收敛时间 σ_T 与步长因子 u 以及矩阵 \mathbf{R} 的第 m 个特征值的关系, 当一定时, 收敛时间与步长因子 u 成反比, 这个公式说明了当 u 比较小时收敛速度慢, u 较大时收敛速度慢。

公式 3.18 和 3.19 指出了 LMS 算法性能和功能的决定因素, 这其中矩阵 \mathbf{R} 是输入信号的自相关矩阵, 我们在实际情况中只能通过控制步长因子 u 来控制 LMS 算法的性能。

提高 LMS 算法的关键就在能否同时获得较快收敛速度和较快的收敛精度, 目前人们一般使用变步长的思想来改进 LMS 算法, 变步长的 LMS 算法中的步长因子 u 不是固定的, 在基于 LMS 算法的自适应滤波器工作之初, 误差比较大, 这时候变步长的 LMS 算法会设置较大的 u 以提高收敛速度, 当基于 LMS 算法的自适应滤波器工作一段时间收, 误差减小到一定程度的时候, 变步长的 LMS 算法会设置较小的步长因子以提高最终的收敛精度, 进一步减小误差。

目前已经衍生出很多种变步长的 LMS 算法^[47-49]，这些算法基本上是在已有算法的基础上修改步长因子 u 的计算公式，然后通过仿真验证算法的性能。 u 的计算公式一般如式 3.20 所示

$$u(n+1) = \begin{cases} f(u(n), e(n)) \\ f(u(n), e(n), x(n)) \end{cases} \quad (3.20)$$

由公式 3.16 可知， u 的变化曲线一般由输入信号 $x(n)$ 和误差信号 $e(n)$ 驱动，但并不一定全部适用于改进本文所研究的流水线 ADC 后台校准算法。由式 3.12 可以看出，校准系统是一个由九个三输入的自适应滤波器组成，其中所有的输入信号并不相关。如果使用 $f(u(n), e(n), x(n))$ 型变步长算法，那么必须根据所有不同的输入信号增加步长迭代模块的输入端和计算单元。而且输入信号中包括子 ADC 的输出 d 、失调电压源和上一级的子级校准输出，这其中 d 只有 00、01 和 10 三种信号，是一个用于校准失调电压误差的常量，的取值情况远远的超过上两个输入信号。所以对于不同的输入信号可能要设计不同步长迭代算法，这样无疑增加了数字电路复杂度，所以本文使用的是基于 $f(u(n), e(n))$ 型步长迭代算法。

本设计使用的变步长算法最终需要设计成数字电路，所以必须考虑算法的复杂度，为了避免在步长迭代算法中使用过于复杂的函数，本文将抽头系数的迭代函数修改为式 3.21 与式 3.22。

$$\theta(n) = \varepsilon \cdot \theta(n-1) + f(e(n), e(n-1)) \quad (3.21)$$

$$u(n) = \rho \cdot u(n-1) + \varphi \cdot \theta(n) \cdot \theta(n) \quad (3.22)$$

公式 3.21 和 3.22 给出了本文的步长变化公式，其中 $f(e(n), e(n-1))$ 根据实际情况设置。为了防止系统出现发散的情况，在实际应用中还给 u 设置了阈值，防止 u 过大或过小。这样在校准过程中，步长 u 会随误差信号大小的变化而变化，误差较大时，步长 u 变大增快校准速度；误差变小时，步长 u 也减小，减小收敛后误差信号的变化范围，而从提高校准输出与高精度期望信号的相似度。

在原有的待非线性误差的仿真模型中增加步长迭代模块，然后根据表 3.1 设置各种误差，给予 11.1525M 的正弦输入信号，再次启动仿真。

图 3.13 给出了变步长校准算法与固定步长校准算法的误差曲线，其中红色曲线为变步长校准算法的误差曲线，蓝色为固定步长校准算法的误差曲线。从两者的对比可以看出变步长的校准算法拥有更快的校准速度和校准精度。

图 3.14 给出了步长迭代曲线，从图中可以看出，步长因子 u 确实如本章分析中一样，先增大再减小，使得校准系统的校准速度和校准精度得到了良好的折中。

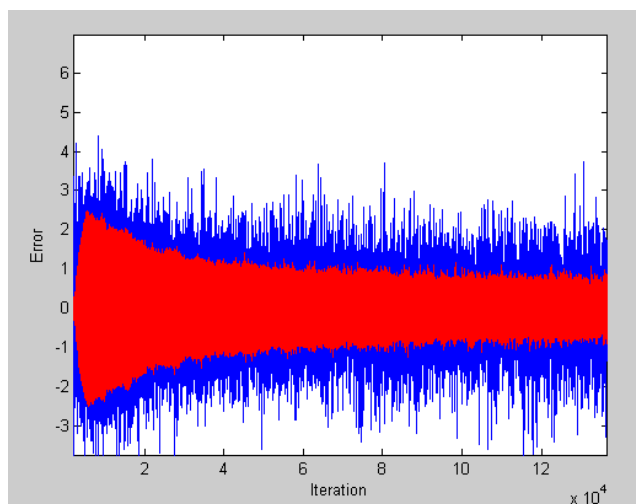


图 3.13 误差曲线

图 3.15 给出了 ADC 信号经过变步长校准后的频谱图，图 3.16 给出了校准后的静态参数示意图，具体参数取值如表 3.4 所示

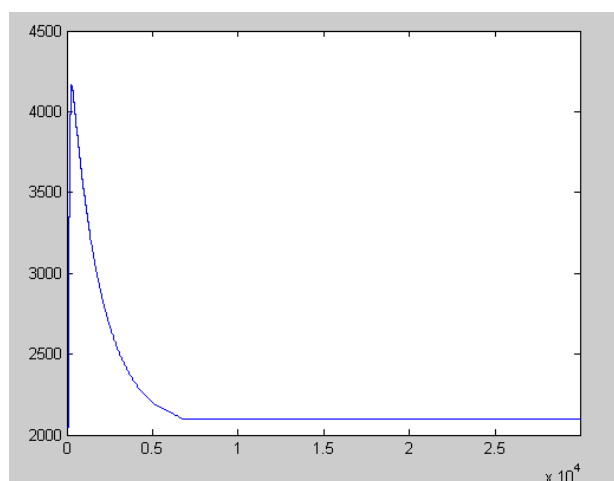


图 3.14 步长迭代误差曲线

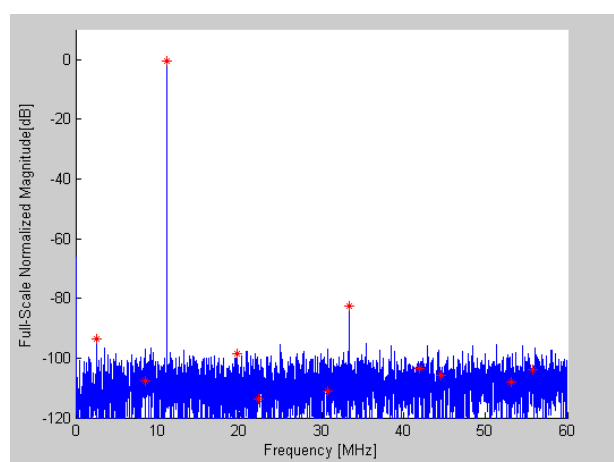


图 3.13 经过变步长算法校准的 ADC 信号

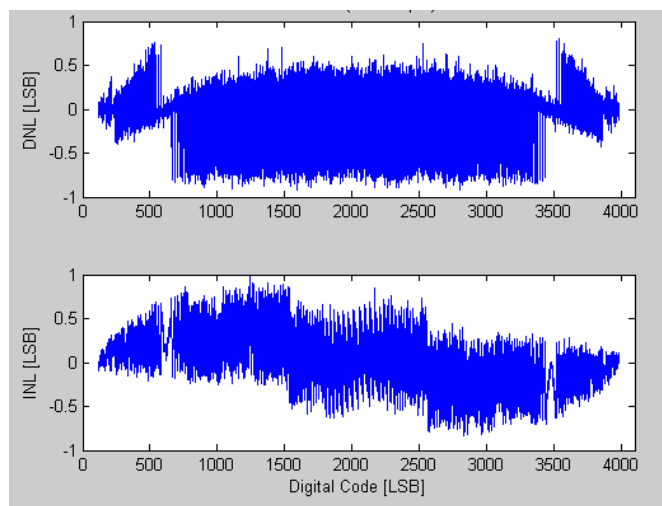


图 3.14 经过变步长算法校准的 ADC 信号 DNL/INL

表 3.4 经过变步长算法校准后的 ADC 性能参数

性能参数	经变步长算法校准的参数
SNR	69.926dB
SNDR	69.605dB
THD	-81.079dB
SFDR	81.6179dB
ENOB	11.329 Bit
DNL	-0.72/+0.80 LSB
INL	-0.84/+0.95 LSB

3.4 本章总结

本章介绍了基于参考 ADC 与自适应滤波的流水线 ADC 后台校准算法的原理，然后根据给定的 ADC 误差模型，验证了线性校准模型与非线性校准模型的校准效果。本章还分析了步长因子 u 对校准系统的影响，提出了改进的变步长迭代方法。从仿真结果可以看出，与固定步长的校准算法相比，变步长的校准算法拥有较好的校准速度和精度。本章基于 Matlab 的模型与仿真结果，为校准算法移植到数字电路上，打下了扎实的基础。

第 4 章 基于参考 ADC 的流水线 ADC 校准算法的硬件设计

本章基于第三章的校准理论和仿真结果设计了校准算法的数字电路，在设计数字电路时，充分考虑了软件实现与硬件实现的相同点和不同点。在保证校准算法功能的同时，运用流水线设计思想优化了电路结构，使校准电路拥有了更快的工作频率；并且对设计的校准电路进行了硬件验证，分析了验证结果。

4.1 硬件电路设计说明

图 4.1 给出了基于参考 ADC 与自适应滤波器的流水线 ADC 校准算法的硬件电路结构，其中带校准的 ADC 是由 9 级 1.5bit/stage 的流水线子级与 3bit Flash ADC 组成的 12bit 流水线 ADC，参考 ADC 是一个 12bit，频率只有流水线 ADC 十八分之一的高精度理想 ADC。

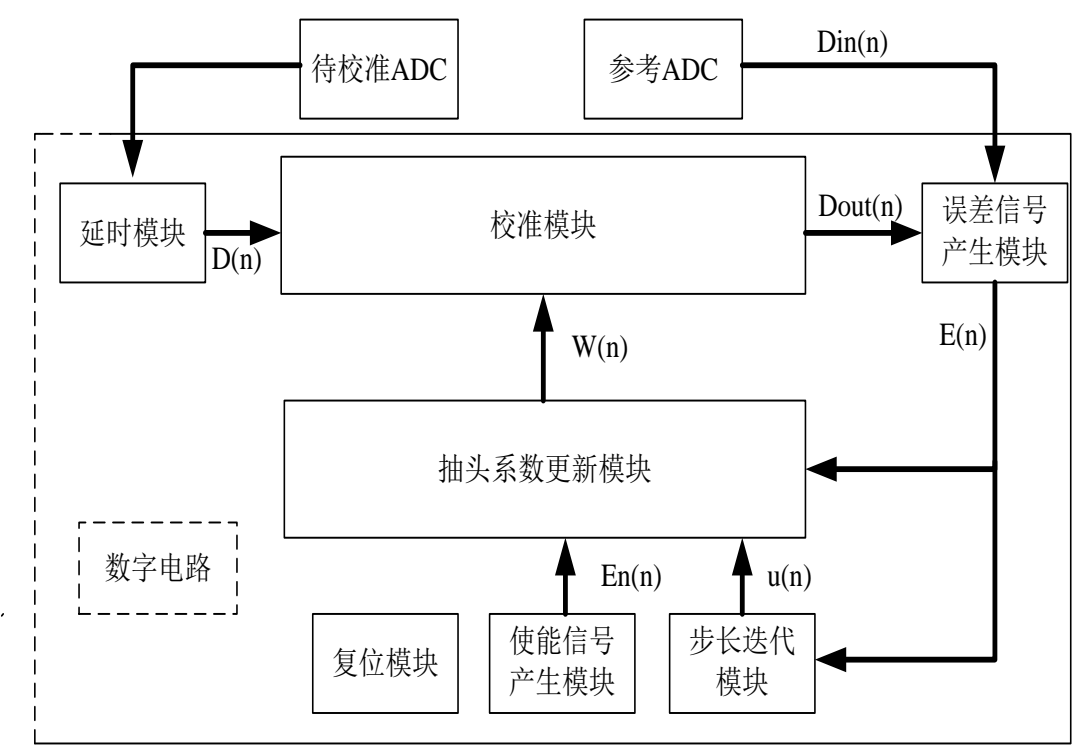


图 4.1 校准电路结构

表 4.1 给出了图 4.1 中各模块的功能介绍，下文将详细介绍各模块的电路结构和所要实现的功能。

表 4.1 各模块功能介绍

功能模块	功能描述
延时模块	控制各子级校准模块的输入延时
误差产生模块	对流水线 ADC 的校准输出作降频处理，在同步校准输出与参考 ADC 的输出后，输出误差信号
使能模块	控制抽头系数的更新频率
步长迭代模块	控制步长因子的变化
校准模块	校准算法的实现
抽头系数更新模块	根据步长因子与输入信号，控制抽头系数的迭代
复位模块	控制系统复位

4.2 硬件电路细节说明

在介绍硬件电路细节之前，先说明校准电路的设计难点。第三章已经详细的介绍了校准电路的理论基础和实现方式，此类校准算法实质就是一个多输入迭代的基于 LMS 算法的自适应滤波器，如式 3.11 所示。

基于 LMS 算法的自适应器实现简单，硬件电路无任何难点。每一个流水线 ADC 的输出是由所有子级与 Flash ADC 的数字码输出决定的，由式 3.11 可得，整个校准结构是由九个子级校准模块组成，每一级的校准输出都是下一级校准输入。所以要实现整体的校准效果，必须从最后一级开始校准。对于最后一级校准模块，它的输入信号是第九级流水线子级的子 ADC 输出与 Flash ADC 输出，这里为了简化设计，失调误差的校准只在前三级子级校准模块。如果不做任何处理，完全按照 Simulink 平台上的模型结构，所有输入信号都使用同样的时序，那么这两个信号就必须要在一个时钟内，经过九个子级校准模块，其中要经过九次有符号乘法和九次加法计算，如图 4.2 所示

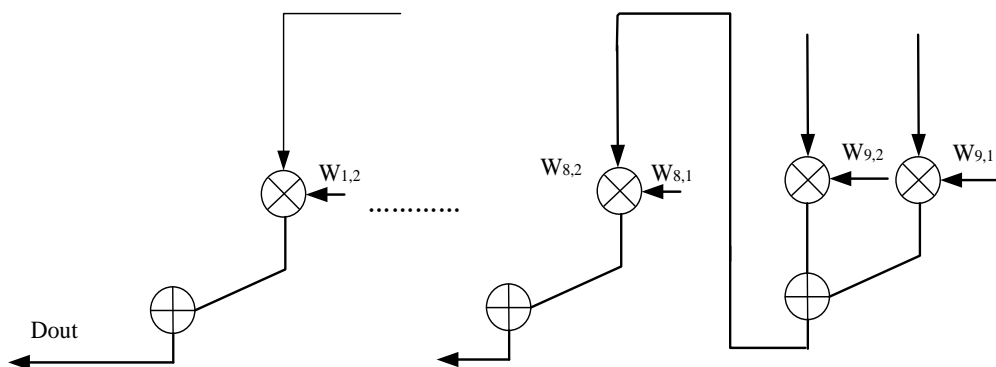


图 4.2 关键信号传输路径

图 4.2 所示的信号传输路径在 Simulink 模型中很容易实现，但是在数字电路中，并且处于在频率较高的环境下，根本无法满足时钟的约束。

若必须在较高时钟频率下实现校准功能，就必须以算法的 Simulink 模型为框架，对数字电路进行优化，根据数字电路中“面积”与“频率”互换原则使用流水线设计校准电路^[50]。

流水线设计可以提高数字电路数据的吞吐量，从而提高数字电路的工作频率。在时序逻辑电路中，组合逻辑制约着系统的最大工作频率。如图 4.3 所示，在一个较大的组合逻辑块中插入寄存器，使得原本在一个时钟内必须完成的组合逻辑操作，分成两个或多个时钟完成。图中信号本来经过 Reg_1 到 Reg_2 输出需要一个时钟，插入寄存器以后，信号从 Reg_3 到 Reg_5 需要两个时钟。但经过这两个时钟以后，可以不间断的输出数据。

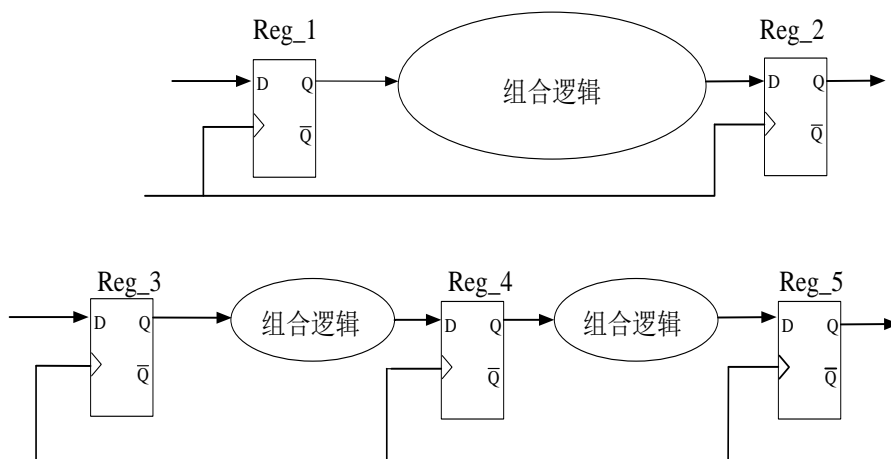


图 4.3 流水线设计的基本思想

下面利用静态时序分析的方法说明数字电路流水线设计的原理，如图 4.4 所示，假设时钟的周期是 T ，时钟信号传输到寄存器 reg1 的延时是 T_{clk1} ，时钟信

号传输到寄存器 reg2 的延时是 T_{clk2} 。信号通过组合逻辑的传输延时是 T_{delay} 。根据静态时序分析的基本理论可得式 4.1^[51]

$$T - (T_{clk2} - T_{clk1}) > T_{delay} + T_{su} \quad (4.1)$$

即如式 4.2 所示

$$T > T_{delay} + T_{su} + (T_{clk2} - T_{clk1}) \quad (4.2)$$

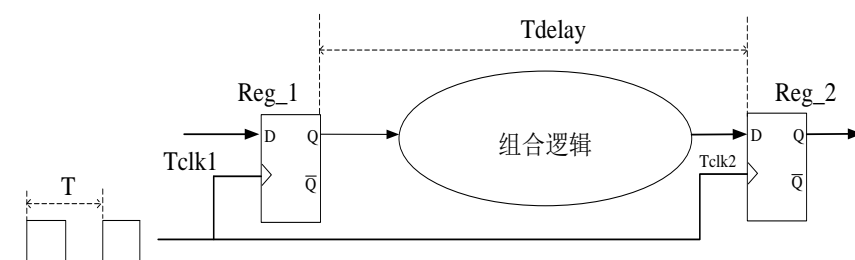


图 4.4 流水线设计的静态时序分析

式 4.2 中 T_{su} 代表建立时间需求，式 4.3 中， $(T_{clk2} - T_{clk1})$ 是由于时钟网络延时而产生的， T_{su} 是由具体器件和工艺决定的，所以若要减小时钟周期 T ，就必须减小。减小的最直接方法就是在组合逻辑中插入寄存器，减小两个寄存器之间的传输延时。

流水线设计的代价就是需要使用较多的寄存器，但在如今的 CMOS 工艺下，数字电路的集成度越来越高，功耗越来越低，增加一定数量的寄存器而提高电路工作频率成为一个很好的折中方案。至于多出的处理时间，可以通过系统的总体优化协调好，并不响应系统的总体功能。

除了使用流水线设计以外，本文在设计数字电路的同时对算法也进行了一定的优化，首先采用的是线性校准模块，因为非线性校准模块必须增加多个乘法器与加法器，数字电路本身的误差令线性校准算法与非线性校准算法的实际校准效果相差无几。

另外流水线 ADC 的设计难度主要集中前几级流水线子级，所以处理前几级以外不再校准失调电压误差，减少自适应滤波器个数。

4.2.1 校准模块的流水线设计

图 4.5 给出了校准模块的流水线设计原理图，其中 $W_{n,x}$ 代表第 n 级子级校准第 x 个抽头系数。 $D_{o,n}$ 代表第 n 级的校准输出。待校准的 ADC 是 12bit 但为了防止输出数据溢出，将 $W_{n,x}$ 与 $D_{o,n}$ 设置为 14bit 有符号数，但最终的校准输出 D_{out} 还是设置为 12bit。

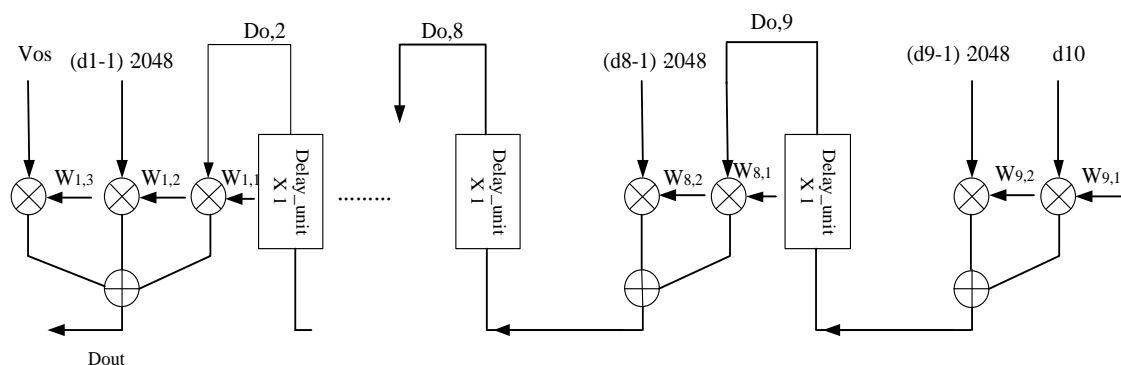


图 4.5 校准模块的流水线结构

图 4.5 在每两级校准输出之间都插入了一个寄存器，这样对于最后一级流水线子级的数字码输出信号与 Flash ADC 输出信号来说，就有九个时钟的时间从最后一级子级校准模块传输到第一级子级校准模块。经过这样的处理，在校准电路中所有的乘法器和加法器都被寄存器隔开，具体如图 4.6 所示

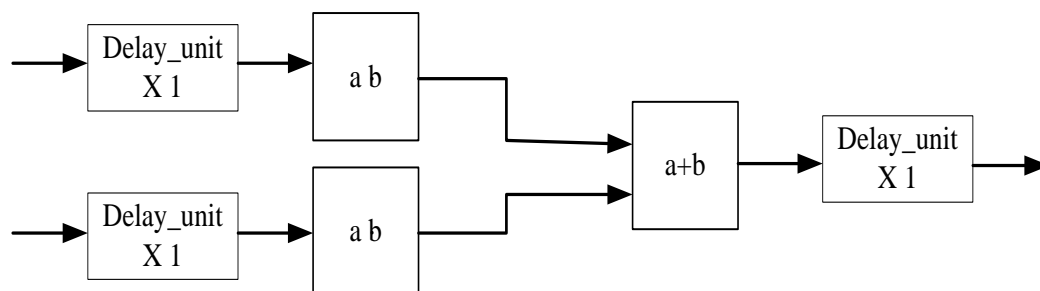


图 4.6 校准系统中的组合逻辑单元

所有子级校准输出的时序关系如图 4.7 所示

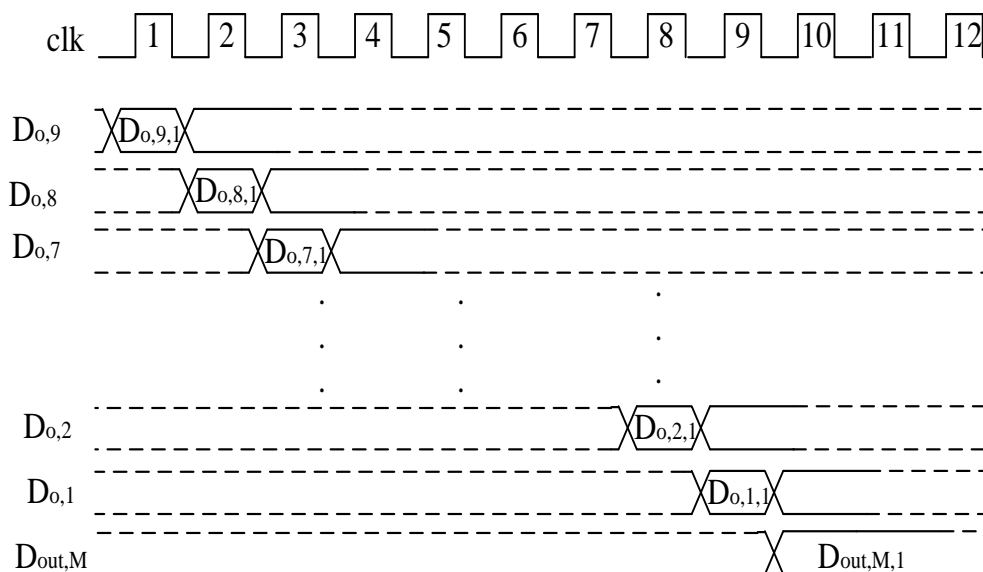


图 4.7 各子级校准输出的时序图

图 4.7 中 $D_{o,n,c}$ 代表第 n 级校准数据的第 c 个输出数据, $D_{out,M}$ 代表经过降频 M 倍的流水线 ADC 输出信号。图中给出了第一个流水线校准输出信号的产生过程, 第一个时钟产生第九级的校准输出信号, 第九个时钟产生第一级校准输出信号, 然后在第十个时钟产生经过降频的 ADC 信号。

但按照本节的中思路, 在校准模块中插入寄存器以后, 整个校准系统的所有信号都必须作相应的延迟处理。这里流水线子级有九级, 上一章已经介绍过参考 ADC 的降频倍数 M 就是 18, 这因为参考 ADC 的频率是待校准流水线 ADC 的十八分之一, 抽头系数更新频率也是待校准流水线 ADC 频率的十八分之一。

4.2.2 输入延时模块与误差产生模块的设计

(1) 输入延时模块的设计

由于九个子级校准模块的输出信号是分九个时钟依次产生的, 所以它们对应输入信号也必须做相应的延时处理, 具体如图 4.8 所示

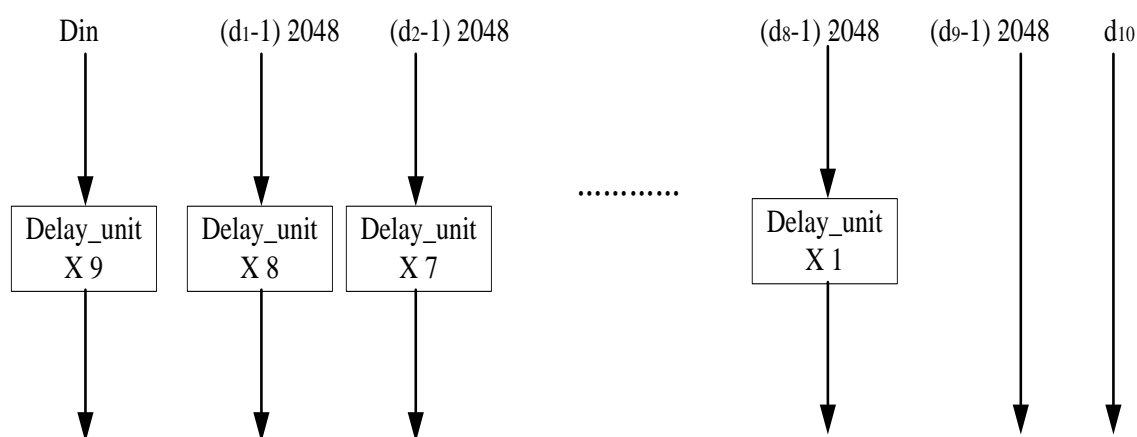


图 4.8 输入延时

图 4.8 中 d_n ($n=1,2,\dots,9,10$) 代表第 n 级子 ADC 的数字码输出, 按照流水线 ADC 的工作流程, d_1 首先产生, d_{10} 最后产生, 经过不同的延时后 $d_1 \sim d_{10}$ 得到同步。这里在这些信号经过同步后, 因为子级校准模块之间插入了寄存器, 为了配合校子级校准输出的流水线处理, 在输入到校准模块之前还需要进行一次延时处理。根据图 4.6 与图 4.7, 校准系统的所有输入信号经过图 4.8 所示的延时处理, 九个子级校准系统的输入与输出就可以对应, 这里将 $(d_x - 1) \cdot 2048$ 设置为有符号的 13bit 数, 因为 $(d_x - 1)$ 会取到负数。

(2) 误差产生模块的设计

图 4.9 给出了校准系统的误差产生模块的原理框图, 这个模块有三个输入信号和一个输出信号, 具体如表 4.2 所示

表 4.2 端口介绍

端口名	端口类型	数据位
D_{in}	输入端口	12bit 无符号型
D_{out}	输入端口	12bit 无符号型
$u(n)$	输入端口	13bit 有符号型
$e(n) \cdot u(n)$	输出端口	14bit 有符号型

误差产生模块首先要实现 ADC 信号与期望信号的同步，经过校准的 ADC 信号首先由一个带使能端的延时单元处理实现降频，使能信号 En 由计数器与比较器控制，计数器的计数周期是 18，比较器的比较值是 8。

其中为经过降频的 ADC 信号，从图 4.9 可以看出，误差信号产生以后经过一个延时单元才输入到乘法器，这里使用的是流水线设计的思想，目的是减少误差信号 $e(n)$ 所在路径的时序压力。

图 4.10 给出了误差产生模块的时序图，校准刚开始时，校准系统将在第 9 个时钟获得第一个 ADC 信号，然后由计数器和比较器联合发出使能信号，将第一个 ADC 信号的长度变为以前的十八倍，在第 10 个时钟与期望信号同步计算出第一个误差信号 $e(1)$ ，误差信号的长度为 18 个时钟，最后在第 11 个时钟得到误差信号与步长因子的乘积。以此类推，误差产生模块将在第 28 个得到第二个误差信号 $e(2)$ ，第 29 个时钟得到第二个误差信号与第二个步长因子 $u(2)$ 的乘积。

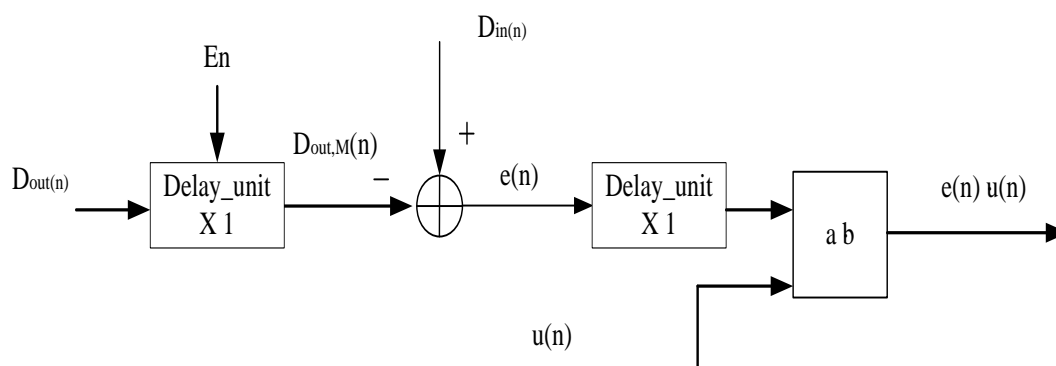


图 4.9 误差产生模块

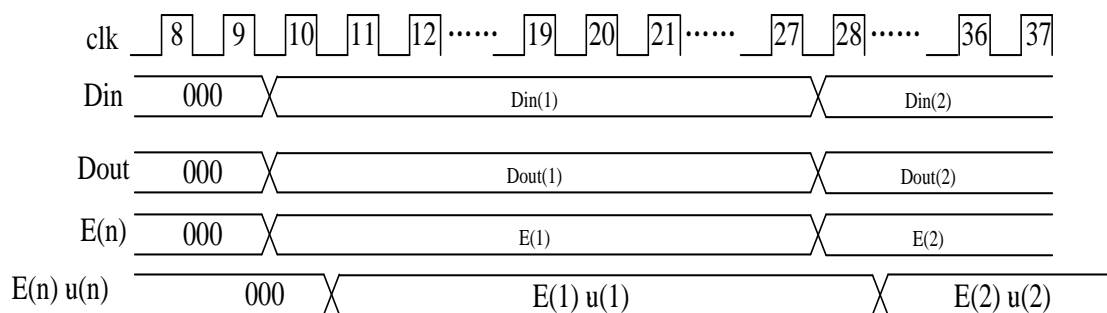


图 4.10 误差产生模块时序

4.2.3 抽头系数迭代模块的设计

抽头系数迭代模块不仅要保证所有抽头系数准确的迭代，还要根据输入信号的时序在正确的时刻完成抽头系数的更新。校准系统在增加了步长迭代模块以后，抽头系数迭代如式 4.3 所示

$$w(n+1) = w(n) + u(n) \cdot x(n) \cdot e(n) \quad (4.3)$$

式 4.3 说明， $w(n+1)$ 的取值由 $w(n)$ 、 $u(n)$ 、 $x(n)$ 与 $e(n)$ 决定，这四个信号相互之间存在因果关系，其中误差信号 $e(n)$ 是由输入信号 $x(n)$ 与抽头系数 $w(n)$ 的乘积 $Y(n)$ 与对应的期望信号做差而得到， $u(n)$ 信号的迭代也是由 $e(n)$ 驱动。所以在抽头系数迭代模块的设计时，必须使用一定的延迟单元来保证这四个同步在运算时是同步的。

为了设计公式 4.3 的硬件电路，误差产生模块将误差信号与步长因子的乘积产生后还需要作不同的延时处理，再输入到不同的子级校准模块中，以满足不同子级校准模块的同步要求。误差的延时单元如图 4.11 所示，误差与步长的乘积要延时一个时钟传输到第九级和第八级子级校准模块中，沿延时八个时钟传输到第一级子级校准模块中，采取这样的延时有两个目的，第一个是同步输入信号、误差信号与步长信号，第二就是减小时序压力。

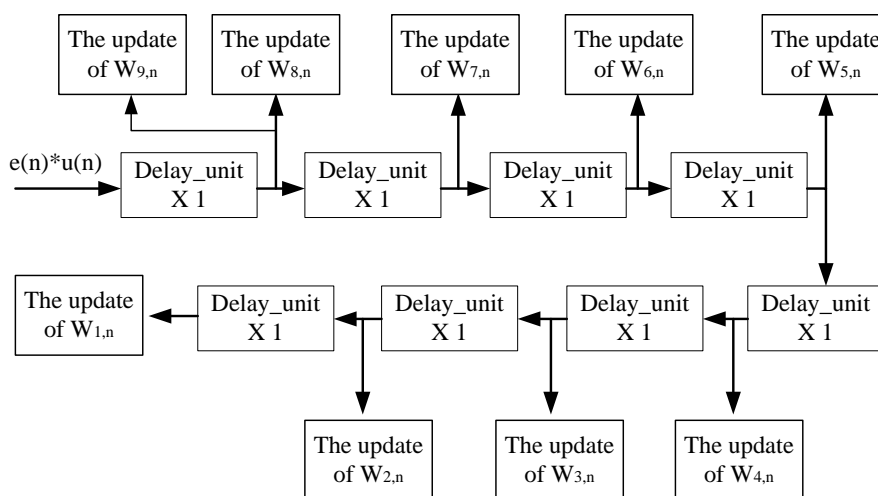


图 4.11 误差信号与步长乘积的延时

假设 $d_1(1) \sim d_{10}(1)$ 与 $D_{in}(1)$ 是待校准的流水线 ADC 与参考 ADC 对第一个模拟信号量化所得, $d_1(2) \sim d_{10}(2)$ 与 $D_{in}(2)$ 是待校准的流水线 ADC 与参考 ADC 对第二个模拟信号量化所得。根据图 4.6 和图 4.10, $d_9(1)$ 与 $d_{10}(1)$ 将比 $D_{in}(1)$ 信号提前九个时钟到达校准模块, 第 10 个时钟产生第一个误差信号 $e(1)$, 第 11 个时钟产生信号 $u(1)$ 与 $e(1)$ 乘积, 这些信号是要用于实现如下迭代

$$w(2) = w(1) + u(1) \cdot x(1) \cdot e(1) \quad (4.4)$$

$w(1)$ 是抽头系数的初始值为零, $w(2)$ 必须在 $d_1(2) \sim d_{10}(2)$ 到达个子级校准模块时之前产生。 $d_9(2)$ 与 $d_{10}(2)$ 将在第 19 个时钟到达, 其他输入信号将按照相应的时序逐步到达。系统在第 13 个时钟完成第九级子级校准模块中抽头系数的第一次更新, 随后其他流水线子级的抽头系数依次完成更新。按照这样的设计, 所有的抽头系数更新都在对应的输入信号到来之前完成, 保证了校准系统正确的工作。

图 4.11 中, $e(n) \cdot u(n)$ 信号是同时传输到第九级与第八级子级校准系统中, 但是第八级的输入信号要比第九级的输入信号晚到一个时钟。按这个思路, $e(n) \cdot u(n)$ 信号应该与第九级的输入信号一样不经过延时直接传输, 可根据图 4.9, $e(n) \cdot u(n)$ 信号是由乘法器产生, 它传输到第九级子级校准系统中也是作为乘法器的输入, 如图 4.12 所示

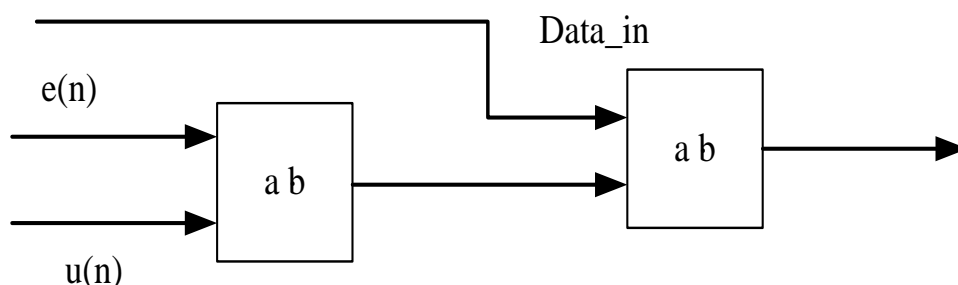


图 4.12 误差信号与步长因子的输出路径

图 4.12 中信号 $e(n)$ 与 $u(n)$ 需要经过两个乘法器, 那么这里的延时将大于图 4.6 中所示的乘法器与加法器的组合, 成为制约系统工作频率的关键, 所以图 4.11 让 $e(n) \cdot u(n)$ 信号经过一个延时单元再传输到第九级子级校准模块中, 其中 $Data_in$ 代表子级校准的输入信号, 可以是子 ADC 的输出信号也可以是上一级子级校准的输出, 如图 4.13 中所示。

图 4.13 中的电路结构避免了两个乘法器的直接级联, 所有子级校准中的抽头系数迭代模块基本一样, 下面介绍第九级和第一级子级校准抽头系数迭代模块。

图 4.15 给出了第九级子级校准的抽头系数迭代的时序图，从图中可以看出，抽头系数在第 13 时钟时完成第一次更新，这个时候 $d_9(2)$ 还未输入，只需要对更新后的第二个抽头系数做一定延时处理就能使 $d_9(2)$ 与它同步。只有输入信号经过正确的延时和分频，加法器由正确的使能信号控制，那么抽头系数就可以在输入信号延时和误差信号延时的情况下完成正确的迭代，同时整个校准系统也可以正确的提出误差信号，使流水线 ADC 的信号不断趋向于高精度的参考 ADC 信号。

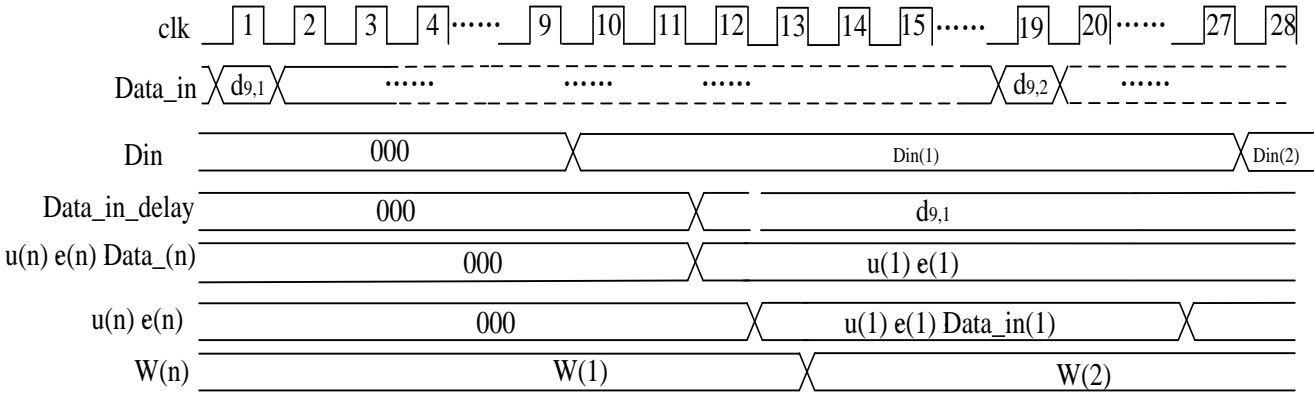


图 4.15 第九级子级校准抽头系数更新模块的时序图

第一级抽头系数迭代模块与第九级类似，区别在于输入信号达到的时刻不一样，所以个别延时单元不一样，另外两个使能信号达到的时刻也不一样，具体结构如图 4.16 所示。

由于误差信号与步长信号根据输入信号延时做了相应的延时，第一级与第二级到第八级的抽头系数迭代模块一模一样，区别只在于两个使能信号到达时刻的不同。整个校准系统在输入端以及内部数据传输中插入相应的寄存器，使得整个系统的动作都流水线化，只是首先需要一定的操作时间。

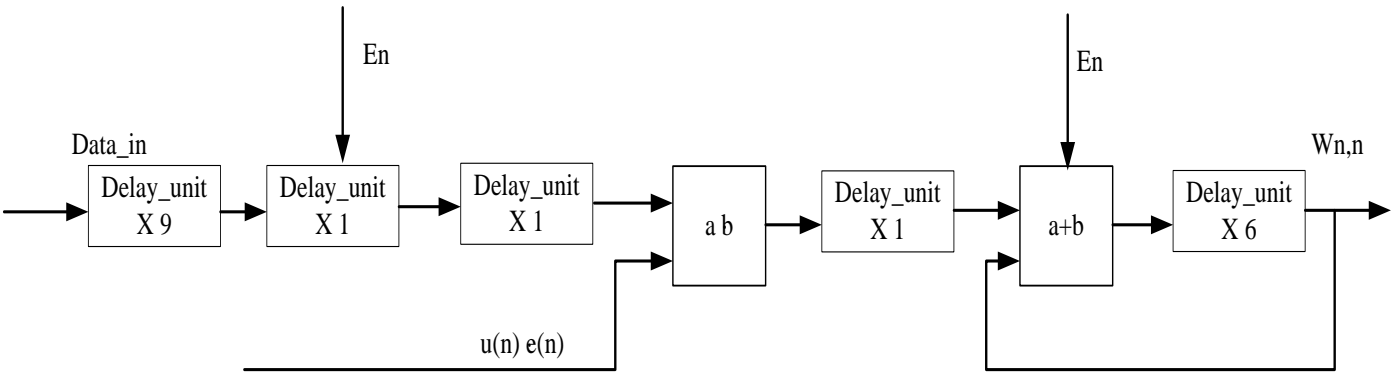


图 4.16 第一级子级校准抽头系数更新模块

4.2.4 其他模块的设计

(1) 步长因子迭代模块的设计

根据公式 3.16, 步长因子 u 的迭代更新过程与抽头系数 W 的迭代更新过程类似, 都是根据误差信号 $e(n)$ 更新, 而且步长因子 u 的更新频率与误差信号 $e(n)$ 的更新频率一样, 所以步长因子迭代模块具有充足的时钟完成乘法和加法操作。而且步长因子迭代模块的结构与抽头系数迭代模块的结构相似, 故在此不再赘述。

(2) 使能信号产生模块的设计

上一节对使能信号产生模块已经有过介绍, 使能信号产生模块由一个计数器和若干比较器组成, 它要按照校准系统的各输入信号的时序, 发出相应时序的使能信号, 以控制抽头系数迭代模块正确的更新, 同时还控制待校准的流水线 ADC 信号以正确的时序降频, 保证正确的提取误差信号。

(3) 复位模块的设计

复位模块负责将复位信号传输到校准系统中所有的寄存器上, 常用的复位模式有同步复位和异步复位, 同步复位是指复位信号到达寄存器以后, 还需要等待时钟沿才让寄存器复位。异步复位则不需要等待时钟沿, 复位信号到达后直接将寄存器复位。

校准系统内部既有时序逻辑电路又有组合逻辑电路, 但进行了流水线设计以后, 系统中所有的动作都由系统时钟控制。复位信号的产生不由校准系统时钟控制属于异步时序, 这样就必须减小复位端口出现亚稳态的概率。

亚稳态是指信号在传输过程中不满足保持时间与建立时间而造成信号不能成功被寄存器抓取或输出^[52], 以致寄存器输出信号呈现不确定的状态, 具体如图 4.17 所示。

图 4.17 中, 一个信号从 0 向 1 跳变时, 因为不满足寄存器的保持时间和建立时间造成寄存器的输出错误, 使输出信号没有向 1 跳变, 而是经过一定时间的不定态以后又跳回 0。

若使用异步复位, 复位信号在信号进行保持时间和建立时间检查时到达寄存器, 那么同样会产生亚稳态, 使得这个寄存器的复位信号失效。若使用同步复位, 复位信号在两个时钟域传输时, 可能不满足校准系统的保持时间和建立时间要求, 同样导致复位信号失效。

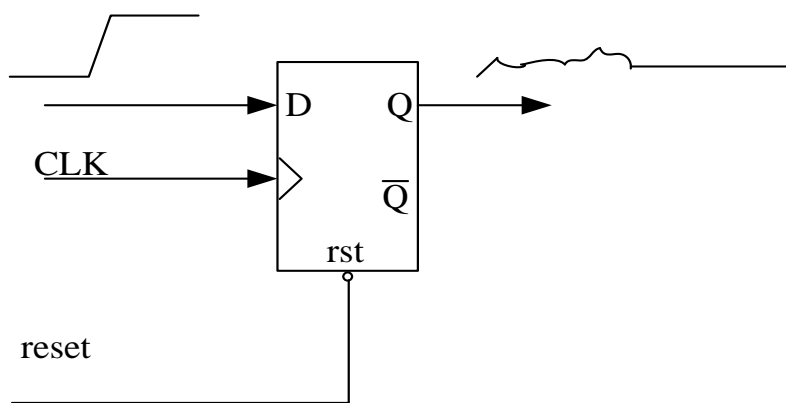


图 4.17 复位信号的亚稳态

为了减小亚稳态出现的概率，本文使用同步寄存器的办法设计复位模块，具体如图 4.18 所示

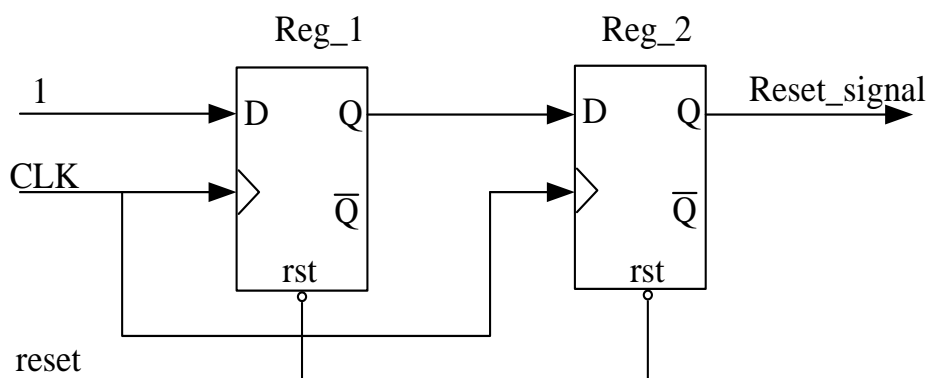


图 4.18 复位模块原理图

根据图 4.18，当复位信号达到 Reg_1 与 Reg_2 时，首先 Reg_1 复位输出置零，然后在下一个时钟，Reg_2 抓取到这个零信号并将它作为系统的复位信号传输出去。这样就算 Reg_1 产生了亚稳态，复位模块依然拥有一个时钟的时间令复位信号稳定，在同一个时钟域中亚稳态的危害会大大降低，这样就保证复位信号正常的功能。

4.3 板级验证

按照上一节的描述完成校准系统的 RTL 级电路设计后，进行板级验证。在板级验证中，使用的输入信号是由第二章介绍的流水线 ADC 的 Simulink 模型提供，流水线 ADC 的参数设置也与第二章一致。这次验证是基于 Xilinx 的 FPGA 型号是 v5-110T，表 4.4 给出了校准系统的输入输出端口。

表 4.4 校准系统的端口介绍

端口名	端口简介	端口类型	位数
D1	第一级子 ADC 的输出	输入	2
D2	第二级子 ADC 的输出	输入	2
D3	第三级子 ADC 的输出	输入	2
D4	第四级子 ADC 的输出	输入	2
D5	第五级子 ADC 的输出	输入	2
D6	第六级子 ADC 的输出	输入	2
D7	第七级子 ADC 的输出	输入	2
D8	第八级子 ADC 的输出	输入	2
D9	第九级子 ADC 的输出	输入	2
D10	Flash ADC 的输出	输入	3
Din	参考 ADC 的数据	输入	12
Clk	时钟信号	输入	1
Rst	复位信号	输入	1
Dout	校准输出信号	输出	12

这里步长迭代模块的将迭代参数固化在数字电路中，不设置专门的输入端口，另外设置输入时钟为 120Mhz。图 4.19 给出了校准系统的 FPGA 验证方案示意图，本文所研究的校准算法不需要改变模拟电路的结构，校准系统可以完全由数字电路实现，所以可以将模拟部分和数字部分分开验证，利用 Simulink 平台上模拟电路模型提供输入信号，将这些输入信号写入 FPGA 上的 RAM 中，然后启动 FPGA 上的板级验证，使用 Xilinx 的 shiscope 工具观察内部信号的变化，并将相关信号。例如误差信号与校准输出信号传输到计算机当中，利用 Matlan 软件计算出校准系统的校准结果。

基于流水线设计的校准系统对于输入信号的时序有着严格的要求,如果输入信号发送端与校准系统的读取端不对应,那么校准系统将无法收敛。所以信号输入端与校准系统采用同样的复位信号。为了能正确的捕捉到相关信号,首先设置相应的计数器和比较器,使 RAM 中的输入信号全部输出完毕以后将整个系统复位。复位以后,所有寄存器恢复初始值,重新开始上一次的过程,如此循环就能捕捉到校准的过程中所有数据,验证的流程如 4.20 所示。

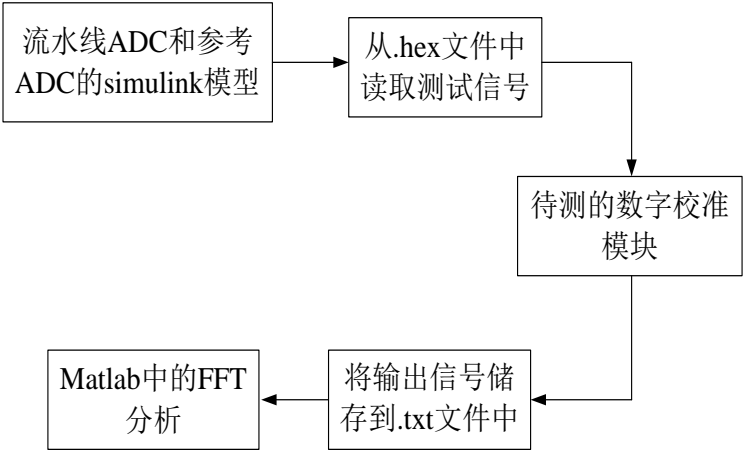


图 4.19 验证平台

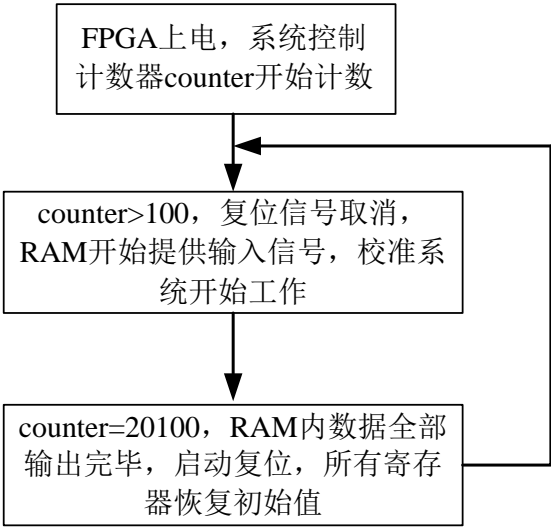


图 4.20 验证流程示意图

板级验证开始后, 依靠 Xilinx 公司的线逻辑分析仪抓取相关数据, 图 4.21 给出了校准刚开始时误差信号的取值; 图 4.22 给出了校准进行一段时间后误差信号的取值。从图 4.22 中可以看误差信号正在不断减小, 说明校准系统工作正常。但是根据软件仿真结果来看, 校准系统一般需要数十万组输入信号才能完全收敛, 可型号为 V5-110T 的 FPGA 资源有限, 只能容纳 20000 组左右的输入数据, 输入数据不足, 令校准系统无法完全收敛。所以这里采取 Xilinx 的 System

Generator 工具，使用软硬件联合仿真的策略进行完全的校准验证。

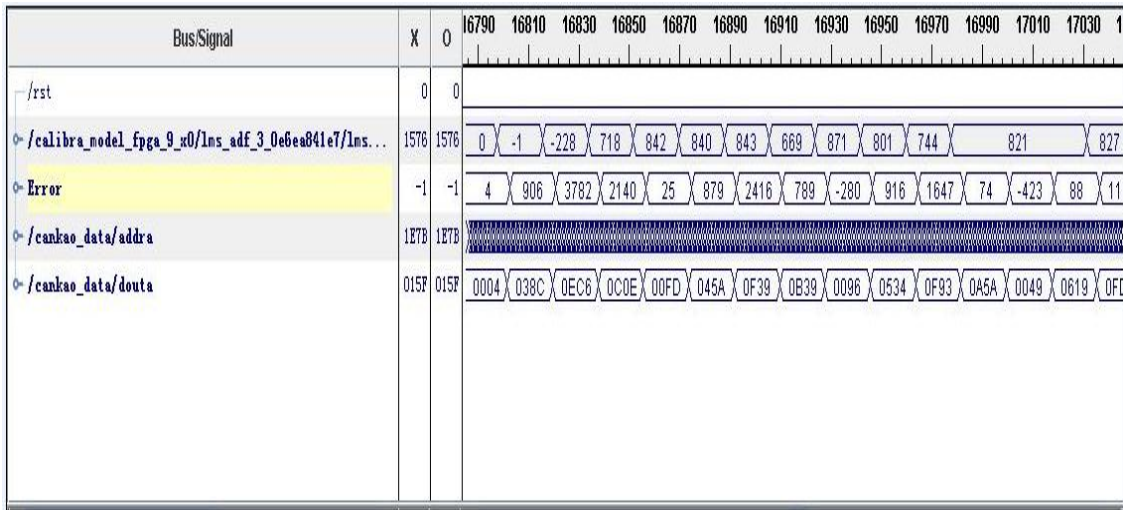


图 4.21 校准开始时的误差

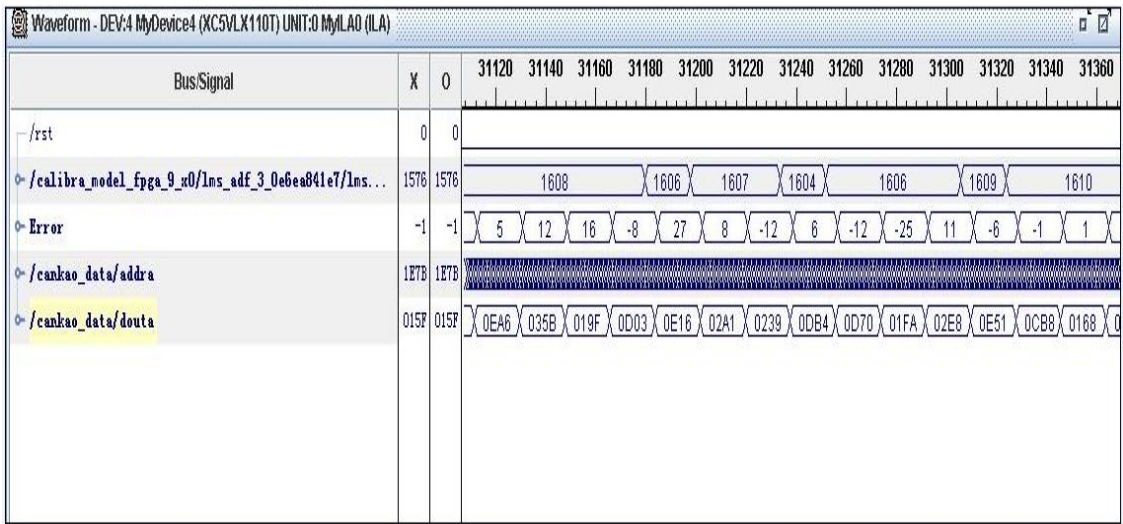


图 4.22 校准一段时间后的误差

System Generator 可以通过 USB 端口与网线完成软件与硬件之间的数据传输，将校准系统的 HDL 模块移植到 System Generator 平台上，通过 System Generator 上的工具完成综合、布局与布线，然后继续进行板级验证。

图 4.23 给出了固定步长校准算法的误差曲线，图 4.24 给出了变步长校准算法的误差曲线。由图 4.23 与图 4.24 可知，与固定步长的校准算法相比，变步长的校准算法拥有更快的收敛速度和收敛精度。

图 4.25 给出了步长迭代曲线，图中步长因子的变化趋势与本章的分析基本一致，使用变步长的校准算法，可以提高校准系统的校准速度和校准精度。

图 4.26 给出了经过变步长校准算法校准后的流水线 ADC 信号频谱图，表 4.5 给出了对应的动态参数。

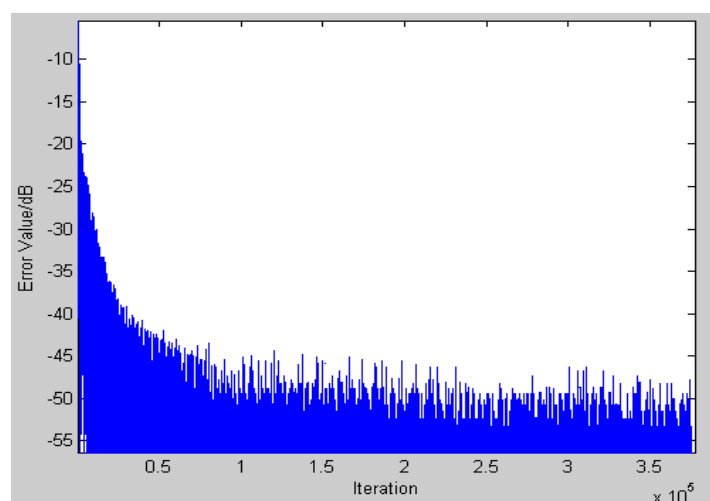


图 4.23 固定步长校准算法的误差曲线

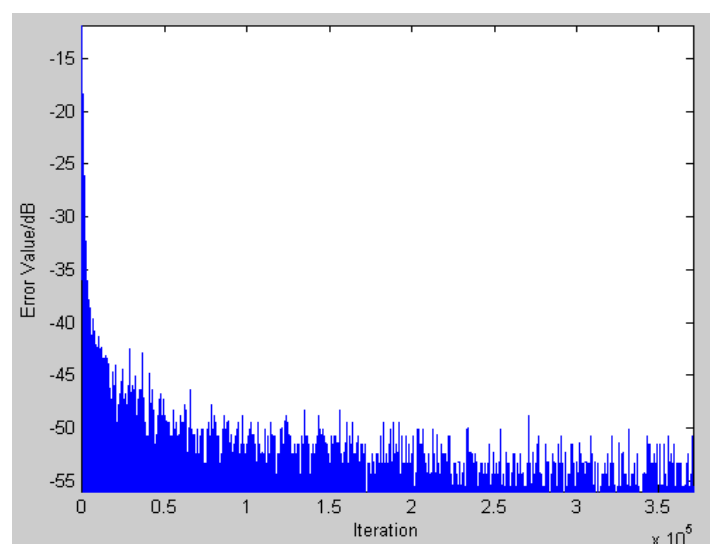


图 4.24 变步长校准算法的误差曲线

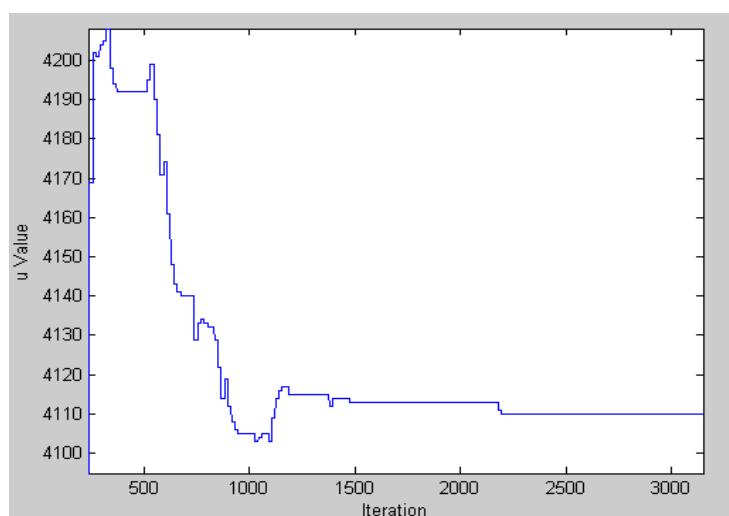


图 4.25 步长迭代曲线

表 4.5 经过变步长算法校准后的 ADC 性能参数

性能参数	未经校准的参数
SNR	53.138dB
SNDR	52.985dB
THD	-67.701dB
SFDR	71.804dB
ENOB	8.541 Bit

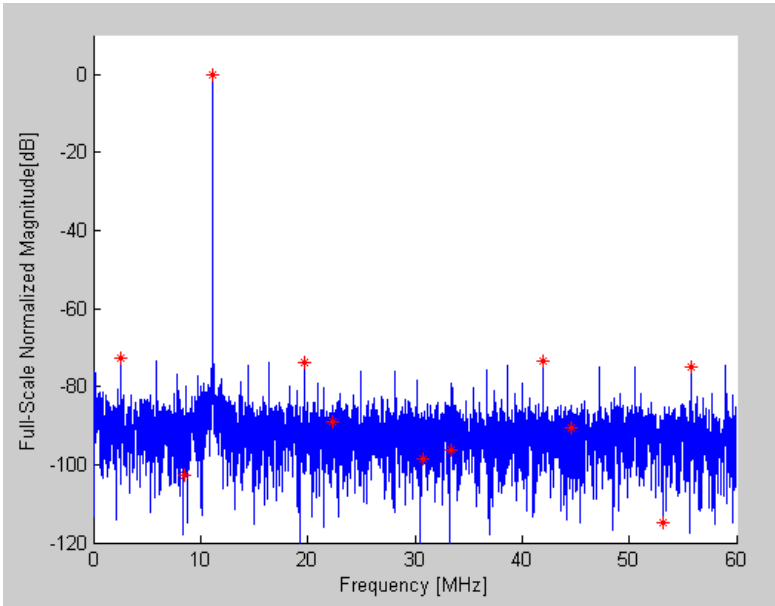


图 4.26 经过变步长校准算法校准的流水线 ADC 频谱图

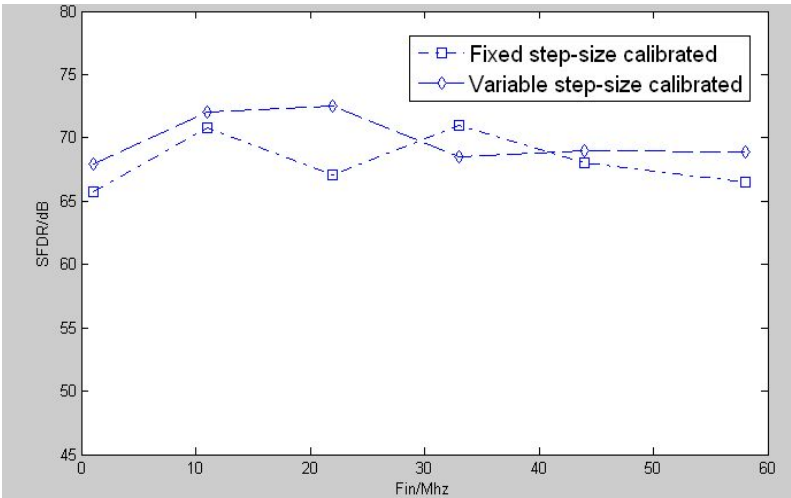


图 4.27 SFDR 的分析结果

为了进一步验证校准算法，给予校准系统多组输入信号，图 4.27 给出了固定校准算法与变步长校准算法的 SFDR 分析结果，图 4.28 给出了固定步长校准算法与变步长校准算法的 SNDR 的分析结果。

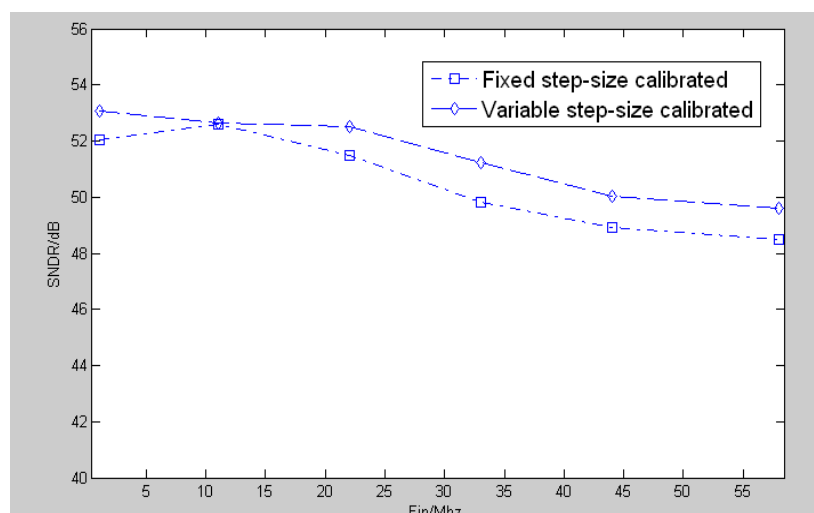


图 4.28 SNDR 的分析结果

由图 4.27 与 4.28 可知，校准系统拥有一定校准效果，但硬件仿真的校准效果与软件仿真的校准效果相差较大，这主要是由于数字电路中有符号小数作乘法时会产生截断误差。单从一次乘法计算的来看，截断误差比较小，但经过数千次乃至数万次迭代以后，截断误差会积累下来影响抽头系数的收敛。

若要提高乘法的精度，就必须增加乘法操作时钟数和数据宽度。对于本文所设计的电路来说，进一步降低参考 ADC 的工作频率(比如设置参考 ADC 的频率为流水线 ADC 的三十六分之一)将会增加单次校准的时钟操作数，这样就可以对乘法进行优化，但是同时校准速度将成倍下降。

4.4 本章总结

本章介绍了校准系统的流水线设计，并且设计了校准系统的 RTL 级电路。为了验证校准系统的 RTL 级电路，使用 Xilinx 的 FPGA 和相关 EDA 工作完成板级验证，最后对板级仿真结果进行了分析。

第 5 章 总结与展望

本文在基本校准算法的基础上,进行了校准算法的改进,并且优化了校准系统的数字电路结构,完成了校准算法的软硬件验证。主要工作内容与成果可总结如下:首先从 CMOS 工艺的发展出发,阐述了使用数字校准技术设计流水线 ADC 的优势,同时介绍了多种不同的流水线 ADC 校准算法,在分析了它们的优缺点的基础上选择了本文研究的校准算法。随后,根据流水线 ADC 的模拟电路结构介绍了流水线 ADC 的误差来源以及这些误差所造成的影响,在此基础上介绍了基于参考 ADC 的流水线 ADC 后台校准算法的基本理论,并且使用变步长的 LMS 算法对其进行改进。接下来,将改进后的校准算法移植到数字电路中,并且使用流水线设计的思想对校准系统的数字电路结构进行了优化,最后完成了校准算法的软硬件验证,并且对软硬件仿真结果的不同之处进行了分析。

随着流水线 ADC 校准技术的进步,校准算法将拥有更全面误差校准能力,更优良的校准性能。本课题还可以在如下方面进一步改进:1.本文完成了校准系统的前端设计,接下来还需要完成版图设计和后端验证等后端设计;2.本文所提到的数字电路中的截断误差,对校准系统的校准性能造成了很大的影响,需要进一步提高数字电路中有符号小数运算的精度。

参考文献

- [1] Maloberti. F, Data converters[M]. Springer Science & Business Media, 2007
- [2] Scheier. R, Temes. R, CMOS Analog Circuit Design[M].北京: 电子工业出版社, 2005.
- [3] Zeller, S, Muenker, C. Weigel R, et al. A 0.039 mm Inverter-Based 1.82 mW 68.6 dB-SNDR 10 MHz-BW CT--ADC in 65 nm CMOS Using Power-and Area-Efficient Design Techniques[J]. Solid-State Circuits, IEEE Journal of, 2014, 49(7): 1548-1560.
- [4] Schreier. R, Temes. G. C. Understanding delta-sigma data converters[M]. Piscataway, NJ: IEEE press, 2005.
- [5] Van De Plassche R J. CMOS integrated analog-to-digital and digital-to-analog converters[M]. Springer Science & Business Media, 2013.
- [6] Payne,Robert,et al. A 16-bit 100 to 160MS/s SiGe BiCMOS pipelined ADC with 100 dBFS SFDR[J]. IEEE Journal of Solid-State Circuits, 2010,45(2):2613-2622.
- [7] Ali,Ahmed,et al.A 14b 1GS/s RF sampling pipelined ADC with background calibration,Solid-State Circuits Conference Digest of Technical Papers(ISSCC),2014 IEEE International.IEEE,2014.
- [8] Todd.Brooks, digitally assisted analog design enables mobile soc evolution[C]. Solid-State Circuits Conference(ISSCC), 2013.2.
- [9] Sansen W, 1.3 Analog CMOS from 5 micrometer to 5 nanometer[C] Solid-State Circuits Conference(ISSCC), 2015 IEEE International. IEEE, 2015: 1-6.
- [10] Murmann.B,Digitally assisted data converter design[M] Proceedings of the ESSCIRC (ESSCIRC). 2013.
- [11] R. Jewett, K. Poulton, K. Hsieh, and J. Doernberg, A 12-b 128 Msamples/s ADC with 0.05LSB DNL,inProc Int. Solid-State Circuits Conf.,Feb. 1997, pp. 138–139
- [12] Z. G. Boyacigiller, B. Weir, P. D. Bradshaw, ‘An Error-Correcting 14b/20ls CMOS A/DConverter,IEEE ISSCC 1981
- [13] 梁上泉.流水线数模转换器伪随机序列注入后台快速数字校准技术研究 [D] .合肥: 合肥工业大学, 2011.
- [14] Un-Ku Moon, Bang-Sup Song. Background Digital Calibration Techniques for Ppelined ADC's[J]. IEEE Transactions on Circuits and Systems II, 1997, 44(2): 102-109.
- [15] Roy S, Sahoo B, Banerjee S. Radix based digital calibration technique for pipelined ADC using Nyquist sampling of sinusoid[C].Circuits and Systems (ISCAS), 2012 IEEE International Symposium on. IEEE, 2012: 2985-2988.
- [16] R. Jewett, K. Poulton, K. Hsieh, and J. Doernberg, “A 12-b 128 Msamples/s ADC with 0.05LSB DNL,” inProc Int. Solid-State Circuits Conf.,Feb. 1997, pp. 138–139
- [17] J. Ming and S. Lewis, “An 8-bit 80-Msample/s pipelined analog-to-digital converter with background calibration,”IEEE J. Solid-Sate Circuits,vol. 36, pp. 1489–1497, Oct. 2001.

- [18] B. Murmann and B. Boser, "A 12-b 75MS/s pipelined ADC using open-loop residue amplifier," ISSCC Dig. Tech. Papers, pp. 330–331, Feb. 2003.
- [19] McNeill J, Coln M C W, Larivee B J. " Split ADC" architecture for deterministic digital background calibration of a 16-bit 1-MS/s ADC[J]. Solid-State Circuits, IEEE Journal of, 2005, 40(12): 2437-2445.
- [20] Li. J ,Moon. U. K, Background calibration techniques for multistage pipelined ADCs with digital redundancy[J]. IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, 2003, 50(9): 531-538.
- [21] Shu. Y. S, Kyung M J, Lee W M, et al. A 15-bit 60-MS/s Floating-Point ADC With Digital Gain and Offset Calibration[J]. IEEE Journal of solid-state circuits, 2009, 44(9): 2356-2365.
- [22] Yun Chiu, Cheongyuen W. Tsang, Paul R. Gray. Least mean square adaptive digital background calibration of pipelined analog-to-digital converters[J]. IEEE Transactions on Circuits and Systems I, 2004, 51(1): 38-45.
- [23] Wang. X, Hurst. P. J, Lewis. S. H, A 12-bit 20-Msample/s pipelined analog-to-digital converter with nested digital background calibration[J]. IEEE Journal of Solid-State Circuits, 2004, 39(11): 1799-1808.
- [24] Wang. H, Wang. X, Hurst. P. J, Nested digital background calibration of a 12-bit pipelined ADC without an input SHA[J]. IEEE Journal of Solid-State Circuits, 2009, 44(10): 2780-2789.
- [25] J. McNeill, M. Coln, B. Larivee, A split-ADC architecture for deterministic digital background calibration of a 16b 1 MS/s ADC, Solid-State Circuits Conference, 2005. Digest of Technical Papers. ISSCC. 2005 IEEE International, vol. 48, 6–10, pp. 276–598, Feb. 2005.
- [26] J. McNeill, M. Coln, B. Larivee, Nair, A " Split-ADC" Digital Background Correction of Open-Loop Residue Amplifier Nonlinearity Errors in a 14b Pipeline ADC[C]. ISCAS. 2007: 1237-1240.
- [27] McNeill. J. A, David. C, Coln. M, et al. "Split ADC" Calibration for All-Digital Correction of Time-Interleaved ADC Errors[J]. Circuits and Systems II: Express Briefs, IEEE Transactions on, 2009, 56(5): 344-348.
- [28] McNeill. J. A, Chan. K. Y, Coln M C W, et al. All-digital background calibration of a successive approximation ADC using the "split ADC" architecture[J]. Circuits and Systems I: Regular Papers, IEEE Transactions on, 2011, 58(10): 2355-2365.
- [29] Fang B N, Wu J T. A 10-bit 300-MS/s pipelined ADC with digital calibration and digital bias generation[J]. Solid-State Circuits, IEEE Journal of, 2013, 48(3): 670-683.
- [30] Ding L, Sin S W, Seng-Pan U, et al. A background gain-calibration technique for low voltage pipelined ADCs based on nonlinear interpolation[C]. Circuits and Systems (MWSCAS), 2013 IEEE 56th International Midwest Symposium on. IEEE, 2013: 665-668.
- [31] Montazerolghaem M A, Moosazadeh T, Yavari M. A Predetermined LMS Digital Background Calibration Technique for Pipelined ADCs[J]. Circuits and Systems II: Express Briefs, IEEE Transactions on, 2015, 62(9): 841-845.

- [32] Li W, Sun C, Li F, et al. A 14-bit pipelined ADC with digital background nonlinearity calibration[C].Circuits and Systems (ISCAS), 2013 IEEE International Symposium on. IEEE, 2013: 2448-2451.
- [33] Hou Y, Li Q, Lin S, et al. An LMS-based adaptive digital calibration algorithm for CMOS pipelined analog-to-digital converters[C].Microelectronics and Electronics (PrimeAsia), 2010 Asia Pacific Conference on Postgraduate Research in. IEEE, 2010: 127-130.
- [34] Huang P, Hsien S, Lu V, et al. SHA-less pipelined ADC with in situ background clock-skew calibration[J]. Solid-State Circuits, IEEE Journal of, 2011, 46(8): 1893-1903.
- [35] 付凯.14 位 200Mhz 流水线 ADC 关键模块设计 [D] .上海: 复旦大学, 2014.
- [36] Ruiz-Amaya J, Delgado-Restituto M, Rodríguez-Vázquez A. Device-level modeling and synthesis of high-performance pipeline ADCs[M]. Springer Science & Business Media, 2011.
- [37] Razavi B, 罗扎. Design of analog CMOS integrated circuits[M]. 清华大学出版社有限公司, 2001.
- [38] Brannon B, Barlow A. Aperture uncertainty and ADC system performance[J]. Applications Note AN-501. Analog Devices, Inc.(September), 2000.
- [39] Brannon B. 采样系统以及时钟相位噪声和抖动的影响[J].
- [40] Ahmed I. Pipelined ADC design and enhancement techniques[M]. Springer Science & Business.Media, 2010.
- [41] Razavi B, 罗扎. Design of analog CMOS integrated circuits[M]. 清华大学出版社有限公司, 2001.
- [42]Johns D A, Martin K. Analog integrated circuit design[M]. John Wiley & Sons, 2008.
- [43] You F, Embabi S H K, Sanchez-Sinencio E. Multistage amplifier topologies with nested G m-C compensation[J]. Solid-State Circuits, IEEE Journal of, 1997, 32(12): 2000-2011.
- [44] Haykin S. 自适应滤波器原理: 英文原版[M]. 电子工业出版社, 2002.
- [45] 李宁. LMS 自适应滤波算法的收敛性能研究与应用[D]. 哈尔滨: 哈尔滨工程大学, 2009.
- [46] Manolakis D G, Ingle V K, Kogon S M. Statistical and adaptive signal processing: spectral estimation, signal modeling, adaptive filtering, and array processing[M]. Norwood: Artech House, 2005.
- [47] Bershad N J, Bermudez J C M, Tournier J Y. Stochastic analysis of the LMS algorithm for system identification with subspace inputs[J]. Signal Processing, IEEE Transactions on, 2008, 56(3): 1018-1027.
- [48] Mathews V J, Xie Z. A stochastic gradient adaptive filter with gradient adaptive step size[J]. Signal Processing, IEEE Transactions on, 1993, 41(6): 2075-2087.
- [49] Shi K, Ma X. A variable-step-size NLMS algorithm using statistics of channel response[J]. Signal Processing, 2010, 90(6): 2107-2111.
- [50] 夏宇闻. Verilog 数字系统设计教程[M]. 北京航空航天大学出版社, 2008.
- [51] Bhasker J, Chadha R. Static timing analysis for nanometer designs: a practical approach[M]. Springer Science & Business Media, 2009.

[52] Altera. Metastability in Altera Devices, Application Note 42, v.4.0, May 1999, pp.1-10,
www.altera.com.

致谢

自2013年9月入学，到现在已经快三年了，我的研究生学习生涯马上就结束了，我的学生生涯也接近尾声。在小学到研究生这么多年的学习生活中，我慢慢学会了如何为人处世、如何学习和如何思考问题，特别是在这三年之中，在老师和同学的帮助下，我意识到自己各方面不足，找到了自己努力的方向。

首先我要感谢我的指导老师谢亮副教授，谢老师在我的课题上给予耐心的指导和帮助，令我在科研上慢慢的掌握了正确的思维方式和科研方法，养成了独立思考的习惯；同时谢老师有着严谨的治学态度，广博的学识，对于科学研究孜孜不倦的进取精神，这不仅无时无刻的感染着我，而且值得我终身学习。感谢IMOS实验室主任金湘亮教授，是金老师建立起IMOS实验室这个大家庭，为一届又一届的同学提供了一个良好的学习平台。

然后我要感谢IMOS课题组的罗志国、刘杨、段志坚和聂拓等师兄师姐，他们在生活和学习上给予我很大的帮助，时常与我交流分享他们的学习经验，使我快速的从一个本科生转变成为一个研究生，衷心的感谢他们！

感谢邓准、覃仕成、周梦嵘和李彬等同学，我们作为一个团队在这三年中一起生活，相互学习，共同进步；是你们让我感受到了团队的温暖，让我的研究生生活变得丰富多彩。感谢田也、路序长、刘惠君、汪杰等师弟师妹，你们为我的研究生生活带来了更多欢乐。

最后我要感谢我的父母，是他们的耐心、理解和无私的奉献让我可以心无旁骛的学习，是他们无微不至的关怀让我感觉到了家庭的幸福和美好，是他们的豁达和热情让我有了开阔的心胸，前进的动力，万分感谢他们！

附录 B 在校期间发表的学术论文及研究成果

- [1] 张文杰, 邓准, 谢亮, 金湘亮. 一种流水线 ADC 后台数字校准算法的实现[J]. 太赫兹科学与电子信息学报.