An Integrated Circuit Pressure Sensing System with Adaptive Linearity Calibration

by

Jennifer Anne Lloyd

Bachelor of Science in Electrical Engineering
Massachusetts Institute of Technology, June 1989

Master of Science in Electrical Engineering and Computer Science
Massachusetts Institute of Technology, June 1992

Submitted to the Department of Electrical Engineering and Computer Science in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy
in Electrical Engineering and Computer Science
at the
Massachusetts Institute of Technology

June 1997

© 1997 Massachusetts Institute of Technology. All rights reserved.

Signature of Author ____________________________

Department of Electrical Engineering and Computer Science
May 1, 1997

Certified by ________________________________ 5/19/97

Hae-Seung Lee
Professor of Electrical Engineering
Thesis Supervisor

Accepted by ________________________________

Arthur C. Smith
Professor of Electrical Engineering
Graduate Officer

JUL 24 1997
An Integrated Circuit Pressure Sensing System with Adaptive Linearity Calibration
by
Jennifer Anne Lloyd

Submitted to the Department of Electrical Engineering and Computer Science on May 1, 1997 in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Electrical Engineering and Computer Science

Abstract

Micromachined capacitive sensors are typically very small, with small associated changes in capacitance, and the resulting kT/C noise often limits the resolution of the sensing system. Oversampling ΣΔ analog-to-digital converters (ADCs) are used to overcome this limitation via the decimator that follows the modulator. The oversampling converter samples data faster than the Nyquist rate so that thermal noise is averaged in the low-pass filtering, resulting in a speed-resolution tradeoff. There is also a tradeoff between resolution, speed, and linearity. Because the sensing system runs open-loop, which is desirable for avoiding large force rebalance voltages and limitations on the sensing medium, the readout is subject to the mechanical nonlinearity of the sensor. A sensor with a large area is mechanically more flexible and gives a larger change in capacitance for a given pressure change, providing high resolution, but yielding more nonlinearity than a smaller sensor over the same pressure range.

A sensing system is demonstrated which uses two ΣΔ ADCs and linearity calibration to relax the resulting trade-off between speed, resolution and linearity. The system incorporates both a large and a small sensor, and exploits the features of each. The large sensor provides the desired signal-to-noise ratio at a given data rate, while the small sensor is used to perform continuous linearity calibration at a lower data rate. This calibration compensates both the nonlinearity of the large sensor as well as slow environmental changes such as temperature. The calibration algorithm is a recursive least squares technique based on a Chebyshev polynomial model of the sensor.

An integrated circuit including the sensor interfaces and the ADCs has been designed and fabricated. The system is demonstrated using micromachined pressure sensors developed at MIT. Measured data indicates successful linearity calibration of the sensors when operated at a 1 Msamples/s sampling rate, with an SNR of about 70 dB for a 1 kHz decimation filter cutoff frequency. The integrated circuit is built in a 0.6 μm, double-poly CMOS technology and uses 150 mW from a 5 V supply.

Thesis Supervisor: Hae-Seung Lee
Title: Professor of Electrical Engineering
Acknowledgments

Funding for this research was provided by the Semiconductor Research Corporation (contract #96-SC-309.002). Support for the author was provided in part by an Intel Foundation Graduate Fellowship.

I am indebted to my thesis advisor, Professor Hae-Seung Lee, for his invaluable assistance throughout the course of this project. His impeccable technical advice and constant encouragement is greatly appreciated. I also thank my thesis committee members for their efforts and helpful feedback. Professor Martin Schmidt has been instrumental particularly in defining the early goals of this work. Professor James Roberge deserves thanks for several key technical discussions, as well as for career advice throughout my stay at MIT. This research has also benefited from the practical guidance of Professor Charles Sodini.

I greatly enjoyed my collaboration with Lalitha Parameswaran. I appreciate her efforts and contributions to the fabrication and packaging for this project.

It has been a pleasure to be a member of the Lee/Sodini research team at MIT. I have enjoyed both the social and technical interactions with the excellent students of those groups. In particular, Ken Szajda has provided extremely helpful technical advice. I also extend thanks to Dan McMahill, Michael Ashburn, Steve Decker and Andrew Karanickolas, who have provided criticisms and suggestions via technical discussions. Joel Phillips and Professor Jacob White have also given helpful technical advice.

Many thanks go to Jeff Gealow for his tremendous contributions to the computing environment in the MTL. In particular his work on our technology file database, and his many awk scripts, have saved us all hours of work.

I would like to thank my industrial mentors, in particular Ian Young of Intel Corporation. I also thank both Rob Weiner and Ted Tewksbury at Analog Devices for assistance with fabrication and packaging of the test chips.

Thanks to the computing, administrative and fabrication staff of the MTL, especially Carolyn Zaccaria who provided assistance with many day to day matters.

I am grateful for all those whose friendship has livened my years at MIT. For this, I extend a special thanks to Melanie Sherony, Ignacio McQuirk and Greg Troxel.

I owe special thanks to my husband Joe for his love and understanding, as well as his companionship during these years at MIT. Finally thanks to Mom, Dad and my sisters, Gwen and Pam, for providing support and encouragement for this and all my endeavors.
## CONTENTS

4.5.1 Polynomial Model .................................................. 56
4.5.2 Block Least Squares ................................................. 58
4.5.3 Recursive Least Squares ........................................... 59
4.6 Wideband Calibration ................................................. 62
  4.6.1 Calibration/Decimation ......................................... 63
  4.6.2 Simulations/Examples .......................................... 65
4.7 Stochastic Search Techniques ...................................... 68
4.8 Hardware and Computational Requirements ...................... 69

5 Sigma-Delta Modulation ............................................... 71
  5.1 Motivation .......................................................... 71
    5.1.1 Basic Operation ............................................... 72
    5.1.2 Noise Shaping ................................................ 73
    5.1.3 SNR ............................................................ 74
  5.2 Modulator Architectures .......................................... 76
    5.2.1 Selection Based on Sensing System Constraints ........... 76
  5.3 Modulator Coefficient Selection .................................. 78
    5.3.1 Design Procedure ............................................ 78
    5.3.2 Second-order Modulator ..................................... 79
    5.3.3 Third-order Modulator ...................................... 79
    5.3.4 Simulator ..................................................... 80

6 Integrated Circuit Design ............................................ 83
  6.1 Operational Amplifier ............................................ 83
    6.1.1 Opamp Dynamics .............................................. 88
    6.1.2 Simulation Results .......................................... 93
  6.2 Integrating/Summing Block ....................................... 96
  6.3 Quantizer .......................................................... 97
  6.4 Modulator .......................................................... 98
  6.5 Sensor Interface .................................................. 103
  6.6 Sources of Error .................................................. 104
    6.6.1 Capacitor Nonlinearity ..................................... 104
    6.6.2 Comparator Offset ........................................... 106
    6.6.3 Component Matching ......................................... 107
    6.6.4 Finite Operational Amplifier Gain ......................... 107
    6.6.5 Operational Amplifier Settling and Slew Rate .......... 108
    6.6.6 Operational Amplifier Offset ............................... 108
    6.6.7 Leakage Current ............................................ 109
    6.6.8 Parasitic Capacitance of the Sensor ..................... 109
    6.6.9 Temperature Coefficient Mismatch ......................... 110
List of Figures

2.1 Wheatstone bridge used for piezoresistive sensing. ........................................ 20
2.2 Simple RC circuit and and associated noise generator. ............................ 21
2.3 Switched capacitor scheme for capacitance to voltage conversion. ............ 22
2.4 Alternate switched capacitor scheme for capacitance to voltage conversion. ........................................ 23
2.5 Switched capacitor scheme with noise voltage sources indicated. .............. 23
2.6 High-frequency capacitor network resulting beyond opamp bandwidth. ...... 25
2.7 Oscillator circuit for capacitance to frequency conversion. ...................... 26
2.8 Pressure sensor fabrication sequence. ......................................................... 29
2.9 Pressure sensor cross-section. ....................................................................... 30
2.10 Membrane deflection profiles under uniform load for sensors with radii 130 μm through 250 μm. ......................................................... 32
2.11 Sensor capacitance versus pressure. ......................................................... 33
2.12 Sensor capacitance normalized to zero-pressure capacitance $C_R$ for membranes with radius 130 μm and 240 μm. ................................. 34
2.13 Sensor nonlinearity versus pressure. ......................................................... 36
2.14 Sensor nonlinearity versus radius for full-scale pressure ranges of 20 and 35 psi. ......................................................... 37
2.15 Electrical model for the wafer-bonded pressure sensor. .......................... 38
3.1 Digital low-pass filtering to remove kT/C noise. ........................................... 43
3.2 Block diagram of capacitive pressure sensing system. ............................... 44
4.1 Sensor characteristics with temperature. .................................................... 50
4.2 Example of adaptive sensor calibration of $C_L$ with reference signal $C_S$. .... 51
4.3 Adaptive linear combiner. ................................................................. 52
4.4 Simple adaptive structure for sensor calibration. ...................................... 54
4.5 Adaptive structure for sensor calibration that uses polynomial sums of pressure to generate estimates of $C_L$ at data rate $f_2$. ...................... 55
4.6 Both maximum and mean-square error versus polynomial order for least-squares polynomial fitting of the 240 μm sensor capacitance over 35 psi. ......................................................... 57
4.7 Wideband adaptive structure for sensor calibration. .............................. 62
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.8</td>
<td>Wideband calibration algorithm - Part I</td>
<td>64</td>
</tr>
<tr>
<td>4.9</td>
<td>Wideband calibration algorithm - Part II</td>
<td>65</td>
</tr>
<tr>
<td>4.10</td>
<td>Simulated transfer function of electrical nonlinearity.</td>
<td>66</td>
</tr>
<tr>
<td>4.11</td>
<td>Simulated spectrum of third-order modulator output, serving as calibration standard.</td>
<td>66</td>
</tr>
<tr>
<td>4.12</td>
<td>Simulated, uncalibrated spectrum of third-order modulator output.</td>
<td>67</td>
</tr>
<tr>
<td>4.13</td>
<td>Simulated, calibrated spectrum of third-order modulator.</td>
<td>67</td>
</tr>
<tr>
<td>5.1</td>
<td>General block diagram of a sigma-delta modulator.</td>
<td>72</td>
</tr>
<tr>
<td>5.2</td>
<td>Generic single-bit modulator.</td>
<td>72</td>
</tr>
<tr>
<td>5.3</td>
<td>Additive white-noise model for modulator.</td>
<td>73</td>
</tr>
<tr>
<td>5.4</td>
<td>Illustration of first-, second- and third-order noise shaping.</td>
<td>74</td>
</tr>
<tr>
<td>5.5</td>
<td>Uniform quantizer.</td>
<td>75</td>
</tr>
<tr>
<td>5.6</td>
<td>Block diagram of second-order modulator.</td>
<td>77</td>
</tr>
<tr>
<td>5.7</td>
<td>Block diagram of third-order modulator.</td>
<td>77</td>
</tr>
<tr>
<td>5.8</td>
<td>Sigma-Delta Block Diagram Simulator</td>
<td>81</td>
</tr>
<tr>
<td>6.1</td>
<td>Regular and active cascode circuits.</td>
<td>83</td>
</tr>
<tr>
<td>6.2</td>
<td>Gain-enhanced operational amplifier using single-ended amplifiers for cascode feedback.</td>
<td>85</td>
</tr>
<tr>
<td>6.3</td>
<td>Fully-differential gain-enhanced opamp using differential amplifiers for cascode feedback.</td>
<td>86</td>
</tr>
<tr>
<td>6.4</td>
<td>Additional gain-enhancement opamp with PMOS input stage.</td>
<td>87</td>
</tr>
<tr>
<td>6.5</td>
<td>Common-mode sensing circuit.</td>
<td>88</td>
</tr>
<tr>
<td>6.6</td>
<td>Small-signal model for the folded-cascode amplifier.</td>
<td>89</td>
</tr>
<tr>
<td>6.7</td>
<td>Folded-cascode amplifier with significant capacitors noted.</td>
<td>89</td>
</tr>
<tr>
<td>6.8</td>
<td>Small-signal model for the gain-enhanced folded cascode amplifier.</td>
<td>90</td>
</tr>
<tr>
<td>6.9</td>
<td>Open-loop gain of the gain-enhanced folded-cascode amplifier.</td>
<td>91</td>
</tr>
<tr>
<td>6.10</td>
<td>Load impedance versus frequency for the gain-enhanced opamp.</td>
<td>91</td>
</tr>
<tr>
<td>6.11</td>
<td>Small-signal model for the gain-enhanced folded cascode amplifier including the auxiliary amplifier dynamics.</td>
<td>93</td>
</tr>
<tr>
<td>6.12</td>
<td>Gain-enhanced opamp open-loop differential-mode characteristic (a) gain and (b) phase.</td>
<td>94</td>
</tr>
<tr>
<td>6.13</td>
<td>Internal transient response at auxiliary amplifier input with (a) single-ended and (b) fully-differential gain-enhancement.</td>
<td>95</td>
</tr>
<tr>
<td>6.14</td>
<td>Switched capacitor integrator/summer.</td>
<td>96</td>
</tr>
<tr>
<td>6.15</td>
<td>Clocked comparator circuit.</td>
<td>97</td>
</tr>
<tr>
<td>6.16</td>
<td>Switch-level schematic of the second-order modulator.</td>
<td>99</td>
</tr>
<tr>
<td>6.17</td>
<td>Switch-level schematic of the third-order modulator.</td>
<td>100</td>
</tr>
<tr>
<td>6.18</td>
<td>Schematics showing switched DAC connections assuming (a) $\phi_P = \phi_2$, and (b) the general case.</td>
<td>101</td>
</tr>
<tr>
<td>6.19</td>
<td>Sensor interface circuit.</td>
<td>103</td>
</tr>
</tbody>
</table>
6.20 Closed-loop pole location of (a) second- and (b) third-order modulator for 25% change in sensor capacitance. ........................................ 105
6.21 Schematic indicating modulator first stage input summing node \( A \). 106
6.22 Schematic indicating leakage current in modulator first stage. .......... 109

7.1 Hybrid sensor integrator layout. ........................................... 112
7.2 Hybrid sensor chip layout. .................................................. 113
7.3 Photograph of integrated circuit. ........................................... 115
7.4 Integrated sensor integrator layout. ....................................... 117
7.5 Integrated sensor chip layout. ............................................. 118
7.6 Hybrid sensor package. ...................................................... 119
7.7 Hybrid sensor package with vertical chip stacking. ....................... 120
7.8 Test setup. ........................................................................... 121
7.9 Reference voltage circuit. ..................................................... 122
7.10 Bias current circuit. ............................................................ 123
7.11 Pressure testing setup. ......................................................... 124
7.12 Circuit for generating electrical nonlinearity. ............................. 125
7.13 Test setup for electrical calibration test. ................................... 126

8.1 Third-order modulator idle-channel noise shown up to \( f_s/2 \). ............. 129
8.2 Third-order modulator idle-channel noise shown up to \( f_s/100 \). ............ 129
8.3 Second-order modulator idle-channel noise shown up to \( f_s/2 \). ............ 130
8.4 Second-order modulator idle-channel noise shown up to \( f_s/100 \). ............ 130
8.5 Third-order modulator wideband spectrum for sine-wave input. .......... 131
8.6 Third-order modulator baseband spectrum for sine-wave input. .......... 131
8.7 Measured transfer function of electrical nonlinearity. ...................... 134
8.8 Measured, uncalibrated spectrum of third-order modulator output. ....... 134
8.9 Measured, uncalibrated spectrum of second-order modulator output, serving as calibration standard. ........................................... 135
8.10 Calibrated spectrum of third-order modulator. ............................ 135
8.11 Convergence of weight \( w_1 \) to its final value. ............................ 136
8.12 Measured transfer function of electrical nonlinearity. ..................... 137
8.13 Measured, uncalibrated spectrum of third-order modulator output. ...... 137
8.14 Measured spectrum of second-order modulator output with residual distortion. ................................................................. 138
8.15 Calibrated spectrum of third-order modulator. ............................ 138
8.16 Idle-channel spectrum of second-order modulator with sensor capacitors, shown up to \( f_s/2 \). .................................................. 140
8.17 Idle-channel spectrum of third-order modulator with sensor capacitors, shown up to \( f_s/2 \). .................................................. 140
8.18 Measured dc characteristic of the large sensor \( C_L \). ....................... 141
# List of Tables

3.1 Pressure sensors parameters. .................................................. 46  
3.2 Calculation of sensor oversampling ratio (OSR) for SNR improvement to 86 dB. .................................................. 46  

4.1 Storage requirements for implementing adaptive calibration. .... 70  
4.2 Floating point operations required to implement adaptive calibration. .................................................. 70  

5.1 Second- and third-order modulator coefficient values. ............ 80  

6.1 Summary of operational amplifier simulation results. .............. 93  

8.1 Calibration weights to correct distortion (maximum INL) of 16%. .. 133  
8.2 Calibration weights to correct distortion (maximum INL) of 10%. .. 133
Chapter 1

Introduction

Micromechanical devices are finding increased application in automotive sensing, medical instrumentation, and industrial control. These devices include sensors and actuators that range from accelerometers, pressure sensors, and displacement meters to miniature motors, generators, and valves. Their appeal is due to several factors. First, because of their small size, which is on the order of several microns, they are essential where minimal invasion or perturbation of the measuring environment is necessary [1]. Their size also allows them to be compactly packaged with signal processing and conditioning circuits. Recent trends have been in investigating the integration of sensors and actuators with signal processing circuitry on the same die by leveraging existing integrated circuit (IC) manufacturing technology [2]. Built together on a single IC chip, the potential exists for providing a low-cost and manufacturable sensor package [3, 4, 5, 6].

Capacitive sensors have shown recent promise in applications where high resolution and low power are critical. Compared with more widely manufactured piezoresistive sensors, capacitive sensors provide higher sensitivity to their input signal at a given power level and are also significantly more temperature stable [2, 7, 8]. However, the output from a capacitive sensor is a small capacitance change. This tiny signal can easily be corrupted by noise introduced by readout circuitry as well as that coupled via packaging and bonding wires, ultimately limiting the resolution of the sensing system. A variety of methods have been used to measure this very small capacitance. For example, digital averaging and oversampling techniques have been used to improve the minimum resolvable capacitance changes [9, 10, 11, 12].

For further improvement in signal-to-noise ratio, the physical structure of the sensor can be altered so that it exhibits more dynamic range over the pressure range of interest. This usually corresponds to more nonlinearity in the sensor characteristic. Force-rebalance techniques can be used to linearize the sensor in a closed-loop configuration. Alternately the sensor must be calibrated, particularly in open-loop systems. Standard calibration techniques include look-up and power-on type calibration [13, 14], but these methods can be costly in terms of the time it takes to calibrate
the sensors before they leave the factory and the silicon real estate that the on-chip memory consumes. In addition, it may be necessary to perform re-calibration due to aging and environmental effects. The memory requirement can be relaxed using numerical interpolation techniques [15, 16]. Self-calibration techniques are useful for compensating environmental effects, but may require a separate calibration cycle in which the measurement system goes off-line [17].

The pressure calibration techniques described compensate for nonlinearity in the sensor characteristic, and have two components: the linearization of the sensor characteristic and the endpoint calibration. To calibrate the endpoints, two known pressure test signals are applied to the sensor and the resulting outputs are measured. Once the sensor characteristic has also been linearized, a linear relationship allows extrapolation of the pressure for any subsequent measurement. Because calculation of the endpoints is necessary for a full pressure calibration, the linearization of the interior points of the characteristic is referred to as linearity calibration.

This dissertation is concerned with the development a generic system which can provide high-resolution digital output from capacitive sensors. The system uses oversampling techniques, as well as continuous sensor linearity calibration, to provide a decoupling of the tradeoffs that normally persist in measurement from small capacitive sensors. The continuous nature of the calibration alleviates the need for large on-chip memory and the application of test signals for calibration. The system is demonstrated with a wafer-bonded silicon capacitive pressure sensor developed at MIT. The research examines packaging solutions in which the readout electronics and the sensor are combined as both a hybrid system and a one-chip solution. To fully exploit the promising characteristics of capacitive sensors, on-chip electronics be used to transduce the physical input to an electrical signal that can be further processed with digital signal processing circuits. The signal processing circuits are also applicable to the calibration of nonlinearity in analog-to-digital converters.

This dissertation presents the design and demonstration of an integrated circuit pressure sensing system. In Chapter 2, background material is presented that influences the system architecture design. The limitations of capacitive sensing are described and provide motivation for the oversampling system design. In addition, the sensor and its electrical characteristics are described, providing the reasons for the linearity calibration in the system design. Chapter 3 describes the pressure sensing system design. Chapter 4 provides details on the sensor calibration. Both the calibration structure and algorithms are motivated and described. Chapter 5 presents an introduction to oversampling analog-to-digital conversion and describes the behavioral level design. Chapter 6 provides the details of the integrated circuit blocks used to construct the sensing system. Chapter 7 presents the test system. The fabrication, packaging and testing issues associated with interfacing with the pressure sensors, and the electrical system used to drive and test the electronic portion of the system, are described. Results from the electrical and pressure testing, along with calibration results are presented in Chapter 8. Chapter 9 provides a conclusion.
Chapter 2

Background

2.1 Pressure Sensors

Several types of sensing techniques are possible for measuring pressure from micro-machined devices. These include piezoresistive techniques [18, 19], capacitor-based techniques [20, 21, 22], as well as optical [23] and resonant methods [24, 25]. The discussion in this section focuses on a motivation for capacitive sensing techniques over more traditional piezoresistive techniques.

2.1.1 Piezoresistive Sensors

Most of the early micromachined pressure sensors used the piezoresistive properties of silicon in order to measure applied pressure. Change in the applied strain in the silicon material causes its resistivity to change, so that deflection of a silicon membrane can be determined by any changes in resistance of one or several membrane-mounted resistors. These resistors are typically formed by a diffusion or ion-implantation in the membrane. This change is often detected using a Wheatstone bridge configuration, similar to that shown in Figure 2.1 [26, 27].

Piezoresistive pressure sensors have several disadvantages. They exhibit significant temperature sensitivity due to the temperature coefficient of the piezoresistivity of the resistors in the bridge. Because of this, these sensors require either temperature compensation schemes [28, 29] or resistor trimming [30] to reduce the overall temperature sensitivity of the output signal. Scaling of piezoresistive sensors, making them smaller but less sensitive to pressure, is limited since the resolution of strain in very small sensors is difficult[31]. While these sensors are still widely produced because they are easy to fabricate, they lack the sensitivity to physical forces that can be achieved with other types of sensors [32, 33].
2.1.2 Capacitive Sensors

Capacitive sensing systems measure applied pressure by sensing the displacement of a membrane as a change in capacitance of the air gap formed between the membrane and a second electrode. Capacitive based sensors are attractive because they offer the potential for sensitive and low-power measurements with low temperature sensitivity. Pressure measurement via the air-gap capacitance change is less dependent on material properties, and therefore temperature, than measurement using piezoresistors. Instead, the sensor characteristics are primarily dependent on the dimensions of the structure.

Readout from these sensors is often based on measuring a capacitance difference or ratio between a pressure-sealed reference capacitor and a signal capacitor, and converting it to a voltage [34, 35, 36, 37] or frequency [38, 13, 39, 40, 41]. This conversion is often an intermediate step in converting the physical signal to a digital output, so an ADC often follows the readout and digital techniques are used for subsequent processing of the signal [42, 43, 44].

Typical changes in capacitance for these structures are about 10% of the full-scale capacitance, and thus are scaling limited by the fact that the small capacitance change may be difficult to measure. The nominal sensor capacitance can be small, on the order of 100 fF to several picofarads, requiring parasitic insensitive schemes. In addition, high-resolution readout requires circuit or signal-processing techniques to overcome the noise limitation, including digital averaging [37, 45] or oversampling techniques [14, 16, 46].

2.2 Noise Limitations in Capacitance Readout

While a variety of techniques can be utilized to improve the resolution of a sensor front-end, a fundamental electrical limitation persists. The limitation is examined in
this section, using two common sensing techniques as examples. Both a switched-capacitor capacitance-to-voltage conversion circuit and a continuous capacitance-to-frequency conversion (oscillator) circuit are used to demonstrate a generalized noise limitation for capacitive sensors.

The noise sources considered in the analysis are the thermal noise generated by resistors, switches, and amplifier input-stage devices. The thermal noise generated by a resistor $R$ is modeled as a voltage source having noise spectral density

$$\overline{v_n^2} \over \Delta f = 4kTR,$$

(2.1)

where $k$ is Boltzmann’s constant and $T$ is the temperature of the device. The thermal noise generated by the input devices of an MOS operational amplifier is represented by input-referred noise voltage sources with noise spectral density

$$\overline{v_n^2} \over \Delta f = 4kT \left( \frac{2}{3g_m} \right),$$

(2.2)

where $g_m$ is the transconductance of the input stage devices. This noise source is attributed to the equivalent channel resistance of $3/2g_m$ in saturation. Differential techniques, as well as chopper stabilization, are utilized in practice to reduce the effects of $1/f$ noise and offsets.

### 2.2.1 Switched-Capacitor Schemes

![Figure 2.2: Simple RC circuit and associated noise generator.](image)

Consider the noise generated by the resistor $R_{on}$ in the $RC$ circuit of Figure 2.2. This circuit is similar to the configuration of a sampling capacitor and associated switch in its on state. The thermal noise of the resistor is bandwidth limited by the $RC$ circuit, so that the total squared noise voltage on the capacitor is

$$\overline{v_{n,\text{switch}}^2} = \int_0^\infty 4kTR_{on} \cdot \frac{1}{1 + j2\pi CR_{on}f} \cdot 2df$$

$$= 4KTR_{on} \cdot \frac{\pi}{2} \cdot \frac{1}{2\pi R_{on}C}$$

$$= \frac{kT}{C}.$$

(2.3)
In a sampling circuit, the total noise is more complex. In general there will be two components to the noise: noise appearing directly on the output when the switch is closed, and a sample and hold noise due to the opening of the switch. Both appear as a $kT/C$ noise, with the sample and hold noise often dominating the total noise[47].

![Switched capacitor scheme for capacitance to voltage conversion.](image)

**Figure 2.3:** Switched capacitor scheme for capacitance to voltage conversion.

Figure 2.3 shows a single-ended signal detection scheme. This circuit measures the change in capacitance of $C_x$, a variable sense capacitor. Via charge redistribution, the difference between the sense capacitance, $C_x$, and a reference capacitance, $C_R$, is measured on phase $\phi_2$ of the two-phase clock

$$v_o = \frac{C_x - C_R}{C_I} V_R = \frac{\Delta C}{C_I} V_R. \quad (2.4)$$

The dominant sources of noise in this scheme are the thermal noise associated with the on-resistance $R_{on}$ of the switches and the input-referred noise of the operational amplifier. Figure 2.4 shows a more practical implementation of this circuit, using parasitic-insensitive switching and offset cancellation while achieving the same functionality.

The total noise generated in this circuit is composed of both the direct noise and the sample and hold noise, which are considered separately. Figure 2.5 shows the direct noise sources for clock phase $\phi_2$. (During phase $\phi_1$, none of the input switches are connected to the opamp, and only the operational amplifier noise contributes to the output. The output is not valid on this clock phase and this noise contribution can be ignored.) Assuming that the switch $RC$ time constants are significantly faster than the opamp, the opamp will bandwidth limit the direct noise sources. The opamp noise $v_o$ will most likely dominate this noise contribution, and the corresponding output referred noise can be calculated.
Figure 2.4: Alternate switched capacitor scheme for capacitance to voltage conversion with offset cancellation.

Figure 2.5: Switched capacitor scheme shown during clock phase $\phi_1$ with noise voltage sources indicated.
It is assumed that the input-referred noise of the opamp is dominated by the thermal noise associated with the input devices

$$\frac{v_o^2}{\Delta f} = 8kT \left( \frac{2}{3g_m} \right). \quad (2.5)$$

Single-pole amplifier dynamics, with unity-gain frequency $g_m/C_C$, are also assumed, where $C_C$ is the opamp compensation capacitor and $g_m$ is the first stage transconductance.

The transfer function between $v_a$ and $v_o$ is approximately

$$H(f) = \frac{v_o}{v_a} \approx \frac{C_x + C_R + C_I}{C_I} \cdot \frac{1}{1 + j\frac{f}{f_1}} \quad (2.6)$$

where $f_1$ is approximately equal to

$$f_1 = \frac{g_m}{2\pi C_C} \cdot \frac{C_I}{C_x + C_R + C_I}. \quad (2.7)$$

The output-referred noise results from the following integral

$$\overline{v_{o,\text{amp}}}^2 = \int_0^\infty 8kT \left( \frac{2}{3g_m} \right) |H(f)|^2 df$$

$$= 8kT \left( \frac{2}{3g_m} \right) \left( \frac{C_x + C_R + C_I}{C_x + C_R} \right)^2 \pi f_1. \quad (2.8)$$

The total noise is thus

$$\overline{v_{o,\text{amp}}}^2 = \frac{4kT}{3C_C} \left( \frac{C_x + C_R + C_I}{C_I} \right). \quad (2.9)$$

The compensation capacitor $C_C$ can be selected so that it is much larger than the reference capacitor $C_R$, in which case this noise source will be negligible compared with the sample and held noise contributed by the switches.

The more significant source of noise in this circuit is that which is sampled and held as each of the switches are opened. Opening switches S1 and S2 will result in a sampled noise that is bandwidth limited only by the $RC$ time constant of the switch, since these switches are isolated from the opamp in their on-state. The same is true for switches S5, S6 and S11. The resulting noise on capacitors $C_x$ and $C_R$ can be referred to the output by the respective sampled-data transfer functions. For example, switch S5 will contribute a noise voltage of

$$\overline{v_{o,\text{switch}}^2} = \frac{kT}{C_R} \left( \frac{C_R}{C_I} \right)^2 \quad (2.10)$$
at the amplifier output.

Switches S3, S4, S7 and S8 will contribute less noise at the output, since associated noise sources will be bandwidth limited by the opamp. In the worst case, the bandwidth of the opamp is faster than the bandwidth of the switches and $kT/C$ noise is sampled. In the more likely scenario, the bandwidth of the opamp is slower than the switches, and the noise will be sampled onto a network of capacitors. For example, Figure 2.6 shows the capacitor network resulting in clock phase $\phi_2$ when the opamp is an open circuit and no longer holds the virtual ground at its inverting terminal. In this case, the switch noise of S3, for example, is sampled onto a capacitance comprised of $C_x$ in series with the parallel combination of $C_R + C_P$ and the series combination of $C_I$ and $C_L$. $C_P$ is the input parasitic capacitance of the opamp and $C_L$ is the load capacitance. The noise at the output will be divided by this network, resulting in only a fraction of $kT/C_x$. In general, an integral including the dynamics of the opamp must be computed, however the assumption here is that the noise will be dominated by the opamp input noise and $kT/C$ noise onto $C_x$ and $C_R$.

![Figure 2.6: High-frequency capacitor network resulting beyond opamp bandwidth.](image)

Assuming that each of the described individual noise sources are uncorrelated, the noise powers are added to calculate the total thermal noise. The total noise for this switched-capacitor scheme is dominated by thermal noise of the switches, and is at a minimum equal to

$$\bar{v}_{SC}^2 \approx \frac{kT}{C_R} \cdot \left[ 2(1 + m) \left( \frac{C_R}{C_I} \right)^2 + 2(1 + m) \left( \frac{C_R}{C_I} \right)^2 \frac{C_x}{C_R} \right],$$

(2.11)

where $m < 1$ is determined by the relative bandwidth of the opamp and the switches. Given the total noise in Equation 2.11, and the transfer function in Equation 2.4, the
signal-to-noise ratio (SNR) is

\[ SNR = \frac{v_o^2}{v_{3C}^2} \approx \left( \frac{\Delta C}{C_R} \right)^2 \left( \frac{kT}{C_R} \right)^{-1} \cdot (2 \cdot (1 + m))^{-1} \cdot \left[ 1 + \frac{C_x}{C_R} \right]^{-1} \cdot v_R^2. \]  

(2.12)

The SNR of a fully-differential version of this circuit will be the same. The number of switches and capacitors is twice as large, although the signal level doubles to compensate. The differential circuit provides the advantage that it can employ chopper stabilization for 1/f noise and offset reduction.

### 2.2.2 Frequency Modulated Readout

![Oscillator circuit](image)

**Figure 2.7:** Oscillator circuit for capacitance to frequency conversion.

An alternate readout technique converts the sensor capacitance change to a frequency change. An example of this type of scheme is shown in Figure 2.7, which is a relaxation oscillator. The circuit has two basic components: a linear integrator and a nonlinear Schmitt trigger. The variable capacitance, \( C_x \), is a time-control element that dictates an \( RC \) product that in turn determines the oscillation frequency. A change in \( C_x \) relative to \( C_R \) will produce a frequency shift from the nominal operating frequency that can be measured with a digital counter and subtracter circuit.

This loop produces a square wave output with period

\[ T = \frac{4V_H}{V_M} RC_x \]  

(2.13)

and thus a frequency of oscillation that is inversely proportional to \( C_x \). The output of the oscillator is assumed to be a square-wave with amplitude \( 2V_M \), where the value of \( V_M \) is the output voltage swing of the Schmitt trigger, and is usually close to the supply voltage. The output square-wave is fed back to the input of the integrator, and, in conjunction with a resistor \( R_1 \), provides a current into a virtual ground that alternately charges and discharges the sense capacitor. Alternatively, \( V_M \) may be the input to a set of switches that turn on charging and discharging currents, creating the
same effect. Thus the integrator output is a triangle wave, which causes the Schmitt trigger to switch when the integrator output voltage reaches the hysteresis voltage of the Schmitt trigger, $V_H$. This loop runs self-consistently, with an up integration time $T_1$ equal to $\frac{2V_M}{V_H} RC_z$ in the steady-state, where $V_H$ is the hysteresis voltage of the Schmitt trigger and $C_z$ is the value of the variable sense capacitor.

The operation performed by the integrator on the capacitance signal can be described by the time-convolution of the integrator input voltage with a pulse of width $T_1$ and height $\frac{1}{RC_z}$ [48]. Equivalently, in the frequency domain, the frequency representation of the input voltage is multiplied by the Fourier transform of the pulse, which is

$$H(f) = \frac{1}{RC_z} \frac{\sin \pi f T_1}{\pi f}.$$  \hspace{1cm} (2.14)

The thermal noise of the opamp and the resistors (or equivalently the charging current) can be lumped into an equivalent resistance $R$, and the total associated thermal noise power is found by integrating the thermal noise spectral density $4kTR$ to yield

$$\overline{v_n^2} = 4kTR \left( \frac{T_1}{RC_z} \right)^2 \int_0^\infty \left| \frac{\sin \pi f T_1}{\pi f T_1} \right|^2 df$$

$$= 4kTR \left( \frac{T_1}{RC_z} \right)^2 \frac{1}{T_1}.$$  \hspace{1cm} (2.15)

This expression can be broken down into three components: the spectral density of the thermal noise $4kTR$, the low-frequency gain of the loop, and the noise bandwidth $f_N = 1/T_1$. Since $T_1 = 2\frac{|V_H|}{V_M} RC_z$, it follows directly that the quantity $\frac{T_1}{RC_z}$ is just $2\frac{|V_H|}{|V_M|}$. The total noise can thus be reduced to the following

$$\overline{v_n^2} = 8\frac{kT}{C_z} \frac{|V_H|}{|V_M|}.$$  \hspace{1cm} (2.16)

To relate this noise voltage to a frequency error at the oscillator output, consider the addition of a voltage error equal to $\sqrt{\overline{v_n^2}}$ to the ramp output of the integrator. The slope of the ramp is $\frac{V_M}{RC_z}$, so that the switching point of the Schmitt trigger can move by $\Delta t = \frac{v_n}{V_M/(RC_z)}$. This corresponds to a frequency error of

$$f_{err} = \frac{1}{T_1} - \frac{1}{T_1 + \Delta t} \approx \frac{\Delta t}{T_1^2}.$$  \hspace{1cm} (2.17)

The frequency error squared is

$$f_{err}^2 = \frac{1}{2C_z} \frac{kT}{(RC_z)^2} \frac{|V_M|}{|V_H|^3}.$$  \hspace{1cm} (2.18)
Since the frequency of operation is \( f_0 = \frac{|V_M|}{|V_H|/4RC_x} \), for a small change in capacitance \( C_x \) of \( \Delta C \), the incremental change in frequency will be

\[
\Delta f = \frac{|V_M|}{|V_H|/4RC_x} \frac{1}{C_x} \Delta C.
\]  \( \text{(2.19)} \)

Thus the signal to noise power ratio is

\[
\frac{\bar{f}^2}{\sigma^2} = \frac{1}{8} \left( \frac{\Delta C}{C_R} \right)^2 \left( \frac{kT}{C_R} \right)^{-1} C_R C_x |V_M| |V_H|.
\]  \( \text{(2.20)} \)

For \( C_x \) nominally equal to \( C_R \), this expression reduces to

\[
SNR = \frac{\bar{f}^2}{\sigma^2} \approx \frac{1}{8} \left( \frac{\Delta C}{C_R} \right)^2 \left( \frac{kT}{C_R} \right)^{-1} |V_M| |V_H|.
\]  \( \text{(2.21)} \)

The result is very similar to the SNR of the switched-capacitor scheme, particularly when the voltages \( V_H \) and \( V_M \) are replaced by a reference voltage \( V_R \):

\[
SNR = \frac{\bar{f}^2}{\sigma^2} \approx \frac{1}{8} \left( \frac{\Delta C}{C_R} \right)^2 \left( \frac{kT}{C_R} \right)^{-1} V_R^2.
\]  \( \text{(2.22)} \)

### 2.2.3 Summary of Noise Calculations

In the previous sections, the impact of the thermal noise on the readout SNR was summarized for two readout schemes and shown in Equations 2.12 and 2.22. As expected, both expressions are similarly dependent on \( kT \) on \( C \), where \( C \) is the nominal sense capacitance \( C_R \), and the reference voltage \( V_R \). Because integrated sensors have very small static capacitances, and even smaller full-scale changes in capacitance, \( kT \) on \( C \) noise dominates the total noise. The maximum full-scale change in capacitance \( \Delta C \) also limits the SNR of the readout.

From Equations 2.12 and 2.22, a typical SNR can be calculated. Assuming a reference voltage of 1 V, a maximum change in capacitance of 10%, and a reference capacitance of 100 fF, the resulting resolution limit is approximately 45 dB, or about 8 bits. To overcome this limitation, a variety of techniques can and have been used. Oversampling, in conjunction with decimation filtering, alleviates the \( kT \) on \( C \) noise limitation, as can correlated double sampling [45].

The readout schemes may also be improved for circuit imperfections other than the \( kT/C \) noise. Using fully-differential implementations reduces the power supply and substrate noise, and allows for the implementation of techniques such as chopper modulation, which can reduce offsets and low frequency noise or correlated double sampling, which also removes low frequency noise.
2.3 Sealed-Cavity Pressure Sensor

This section presents details of the sealed-cavity pressure sensor used as the test vehicle for the capacitive sensing system. The fabrication sequence for the sensor and its final cross-section will be briefly reviewed. The characteristics of the sensor and its linearity will be described. This is followed by a description of the electrical model for the sensor including its parasitic elements. Finally, a discussion of force feedback requirements will be presented.

2.3.1 Fabrication Sequence

The pressure sensor is fabricated in a process which combines micro-electromechanical (MEMS) processing via wafer-bonding and a conventional CMOS process [49]. The wafer-bonded process has been used to successfully build several microstructures [50, 51]. One of its advantages is allowing the use of single-crystal silicon for the mechanical element, therefore providing good reproducibility of the mechanical properties of the structure.

Figure 2.8 illustrates the fabrication sequence for the sensor. The initial steps in the fabrication sequence have been designed to allow for the subsequent high-
temperature steps necessary to complete integrated circuit processing without plastically deforming the membrane [49]. First a 1 µm cavity is formed in an n-type handle wafer by plasma etching. A p-type device wafer is implanted to a 10 µm depth to form an n-type layer. This 10 µm n-type layer eventually forms the membrane and the substrate for CMOS processing, so the dopant and its concentration are selected appropriately. The device and the handle wafer are joined together using a silicon wafer-bonding technique. The handle wafer is thinned using a chemo-mechanical polish etch. This is followed by an electrochemical etch, which uses the p-n junction as an etchstop for thickness control, leaving a 10 µm silicon membrane. In the case where a p-type substrate is desired for CMOS processing, a silicon-on-insulator (SOI) device wafer can be used instead. In this case the buried oxide layer, instead of the p-n junction, is used as an etchstop.

Standard IC processing occurs next in a 1.75 µm twin-well CMOS process. This is followed by back-end low-temperature mechanical processing used to etch the pressure inlet port and free the membrane. The back-end processing is designed so that the devices formed in the IC part of the process flow are not adversely affected by temperature cycling or other effects. Metal for the top-electrode of the sensor is patterned on a Pyrex glass wafer. The glass and silicon wafers are subsequently bonded together using a thermo-compression bond. The glass wafer is useful for aligning these steps, and also provides an over-pressure stop for the sensor. The final processing step is the release of the back side of the membrane. The back side of the wafer is etched using KOH, and uses a etchstop layer formed inside the cavity before wafer bonding. Figure 2.9 shows a completed cross section of the sensor.
2.3.2 Capacitance Model

Using the wafer-bonded sensor, pressure is applied through the back inlet port. The silicon membrane deflects, changing the depth of the air gap and hence the capacitance of the structure. This capacitance is electrically measured and calibrated to produce analog or digital readout of the signal.

The sealed-cavity pressure sensor is modeled as a clamped circular membrane under uniform load. A nonlinear partial differential equation, in conjunction with boundary conditions for the membrane, determines the deflection surface of the plate. For a clamped membrane, the deflection \( w \) is zero at the edges, and the slope of the deflection surface in the radial direction, \( dw/dr \), is zero at the center and the edges of the membrane. For deflections that are small compared with the plate thickness, the strain in the middle plane of the plate is negligible. As a result, the differential equation becomes linear and the solution to the plate problem is simplified. In the case of large deflections greater than about half the thickness of the membrane, a nonlinear solution must be found to the differential equation [52].

Assuming small deflection of the clamped circular membrane, the deflection profile \( w(r) \), described as a function of radius, is given by

\[
w(r) = \frac{q}{64D} (a^2 - r^2)^2, \tag{2.23}
\]

where \( a \) is the radius of the sensor and \( q \) is the applied load per unit area, or pressure. \( D \), the plate constant, is equal to \( \frac{Eh^3}{12(1-\nu^2)} \), where \( E \) is Young's modulus for silicon, \( h \) is the membrane thickness and \( \nu \) is the Poisson ratio for silicon [52]. The maximum deflection of the plate occurs at its center and is equal to

\[
w(r) = \frac{q}{64D} a^4. \tag{2.24}
\]

The deflection of several 10 \( \mu m \) thick membranes with radii varying between 130 \( \mu m \) and 250 \( \mu m \) is shown in Figure 2.10 for a uniform load of 35 psi. These calculations assume a Young's modulus \( E \) of 160 GPa and a Poisson ratio \( \mu \) of 0.25.

Given this analytical model for the deflection, the total capacitance as a function of \( w \), and therefore of \( q \), can be calculated by integrating infinitesimally small concentric parallel capacitances over the total area of the membrane. Assuming an air gap \( d \) between the membrane and the bottom electrode, the total capacitance can be calculated from the following integral

\[
C = \int_{r=0}^{r=a} \frac{\epsilon 2\pi r \cdot dr}{d - w(r)}, \tag{2.25}
\]

where \( \epsilon \) is the permittivity of the material in the gap. The resulting total capacitance is

\[
C = \frac{\epsilon \pi}{2\sqrt{dKq}} \cdot \left[ \log \frac{\sqrt{d} + \sqrt{Kqa^2}}{\sqrt{d} - \sqrt{Kqa^2}} \right], \tag{2.26}
\]
Figure 2.10: Membrane deflection profiles under uniform load for sensors with radii 130 \( \mu \text{m} \) through 250 \( \mu \text{m} \).
Figure 2.11: Sensor capacitance versus pressure for membranes with radius (a) 130 μm and (b) 240 μm.
Figure 2.12: Sensor capacitance normalized to zero-pressure capacitance $C_R$ for membranes with radius 130 $\mu$m and 240 $\mu$m.
for $q > 0$ where $K = (64D)^{-1}$ is a material-dependent constant.

Figures 2.11(a) and (b) show the capacitance versus pressure for two sensors of thickness $h$ of 10 μm and radii $a$ of 130 μm and 240 μm, respectively. In Figure 2.12, the capacitances are shown normalized to the zero-pressure capacitances $C_R$ of 0.48 pF and 1.6 pF, respectively. This figure illustrates the pronounced difference in the relative change in capacitance and the nonlinearity over the given pressure range. The deflections for the large and small sensors under maximum pressure are 0.75 μm and 0.065 μm, respectively. In both cases, the deflection in microns is less than 20% of the 10 μm thickness of the material, so the small deflection solution condition holds [52]. Notice that the larger sensor has a much greater nominal capacitance and change in capacitance in the pressure range of interest. However, the large sensor is more nonlinear than the small sensor.

### 2.3.3 Linearity

The nonlinearity of the sensor over a given pressure range is found by calculating the deviation from a linear capacitance characteristic. Specifically, if the sensor has a capacitance at a maximum pressure $q_{FS}$ of $C_{FS}$, and a zero-pressure capacitance of $C_R$, then if it is perfectly linear, the capacitance as a function of pressure is

$$C_{lin} = C_R + q \cdot \left[ \frac{C_{FS} - C_R}{q_{FS}} \right].$$  \hspace{1cm} (2.27)

The integral nonlinearity (INL) of the sensor is the deviation of the actual capacitance $C$ from this ideal characteristic, referenced to the maximum capacitance, $C_{FS}$

$$INL(q) = \frac{C_{lin} - C}{C_{FS}}.$$ \hspace{1cm} (2.28)

Figure 2.13 shows the nonlinearity versus pressure for the two sensors while in small deflection. The maximum nonlinearity over a 35 psi pressure range is about 13% for the 240 μm sensor and about 0.03% for the 130 μm sensor. Figure 2.14 indicates the dependence of the maximum INL on sensor radius for sensors operating under the same full-scale pressure. The characteristic is shown for full-scale pressure ranges of 20 psi and 35 psi. This graph emphasizes the strong dependence of linearity on the radius of the sensor. Physically this dependence is related to the fact that the deflection of the sensor is proportional to the fourth power of the radius.

### 2.3.4 Equivalent Circuit Model

One of the difficulties with sensing from micromachined devices is not only that the devices themselves are small, but that the parasitic capacitances formed by layers associated with the sensor are significant compared with the nominal capacitance
Figure 2.13: Sensor nonlinearity versus pressure for membranes with radius (a) 130 μm and (b) 240 μm.
Figure 2.14: Sensor nonlinearity versus radius for full-scale pressure ranges of 20 and 35 psi.
of the sensor. In addition, parasitics associated with the implementation and packaging, which can include bond-pad and metal-line capacitances, metal and contact series resistances and bond-wire inductance, can be important for modeling the device sensitivity.

A lumped model indicating the parasitic elements associated with the sensor is shown in Figure 2.15. $C_x$ represents the capacitance of the sensor. $D_1$ is the diode formed by the membrane and the silicon substrate, and $C_D$ is the associated junction capacitance. $C_{ox}$ is the field oxide overlap capacitance formed between the top electrode and the bottom plate. $C_T$ and $C_B$ represent the metal over field oxide to substrate capacitance of the metal traces connecting to the top and bottom plates, respectively.

2.3.5 Force Feedback

In some applications, the sensor is packaged in a closed-loop system. A feedback voltage is used to move the membrane to a zeroed position by creating an electrostatic force between the membrane and another electrode. The voltage creates a force on the membrane at position $r$ of

$$F(r) = \frac{1}{2} \epsilon_o E(r)^2 A, \quad (2.29)$$

where $E$ is the electric field between the electrodes and $A$ is the area of the plate. This can be related to an equivalent pressure

$$q(r) = \frac{1}{2} \epsilon_o E^2 = \frac{1}{2} \epsilon_o \frac{V^2}{d - w(r)}, \quad (2.30)$$

where $d$ is the gap thickness, $w$ is the displacement of the membrane, and $V$ is the voltage across the electrodes. This force is always an attractive force, since it is proportional to the voltage squared.
In a closed-loop sensing system, it may be necessary to apply some fraction of the full-scale physical signal, in this case pressure, to rebalance the sensor. The worst-case voltage necessary to apply a full-scale pressure to the sensor can be calculated for a given sensor displacement and gap thickness. The worst-case assumption, which yields the largest feedback voltage, is that the sensor displacement is small compared with the gap thickness \( d \). Assuming a 1 \( \mu \text{m} \) air gap depth, a voltage of 233 V is required to apply a full-scale pressure range of 35 psi, or 241 kPa, on the membrane. A voltage of 5 V would apply a pressure of 110 Pa, or 0.016 psi, which is only 0.04% of the full-scale pressure range. For this reason, force feedback for this type of pressure sensor is incompatible with a low-power integrated circuit interface for a full-scale pressure range of 35 psi.
Chapter 3

Sensing System Architecture

3.1 Motivation

The sensing system design serves two primary purposes. The first is to address the need for open-loop sensing. The second is to develop a technique to overcome the fundamental trade-off between the speed, the linearity and the resolution of the sensor readout. Each of these purposes is described in this section.

3.1.1 Open-loop Sensing

Many sensing systems use force rebalancing techniques. For example, accelerometers typically require feedback to keep the proof mass at a zeroed position in the air gap. The advantage of using a force-rebalance technique is that it places the sensor in a closed-loop configuration. To first order, the output of the system is then insensitive to the mechanical properties of the sensor. In the case of the pressure sensor, this is desirable for reducing effects of the sensor nonlinearity. The disadvantage of using force rebalance is that it may require large feedback voltages. In particular, pressure sensors with a pressure range of more than a few psi require excessively large force-rebalance voltages. In low-voltage systems, it is desirable instead to operate sensors open-loop to avoid supplying these large voltages.

It is also desirable that the sensing system be capable of measuring pressure in both conducting as well as non-conducting media. Force feedback of the membrane in a non-conducting medium is possible by applying a voltage on an electrode below the membrane, since this will create an attractive force to counteract the effects of the pressure. In a conducting media however, the two electrodes in this scheme will short together. Either a more complex type of feedback or more elaborate sensor fabrication sequence is required to avoid the need to apply a voltage across the conducting medium. To allow the flexibility in sensing in any media, an open-loop sensing solution was sought for the system design.
3.1.2 Speed, Resolution and Linearity Trade-offs

Micro-machined capacitive sensors are typically very small with small associated changes in capacitance. As a result, the kT/C noise of the sensor often limits the resolution of the readout. This limitation can be overcome by using oversampling and decimation techniques to average the noise of the sensor. The amount of oversampling required to achieve a given signal-to-noise ratio (SNR) at a given data rate is directly related to the size and the full-scale change in capacitance of the sensor. The result is a trade-off between the speed and the resolution of the system when oversampling techniques are used.

There also exists a trade-off between the resolution and the linearity of the sensor. A sensor with a large area will have a bigger nominal capacitance and show higher sensitivity to changes in pressure than a sensor with a smaller diameter (over the same pressure range), and is therefore able to provide high SNR at a given data rate. At the same time, a large sensor exhibits more nonlinearity than a smaller sensor over the same pressure range. The small sensor, which is mechanically less flexible, can be extremely linear over the measurement range, but will exhibit less sensitivity to pressure. Because of its nominal size and maximum change in capacitance, the small sensor requires a higher oversampling ratio to achieve the desired SNR than the larger sensor.

For these reasons, there exist trade-offs between the linearity, the resolution and the readout speed of the sensing system. To relax the trade-offs, the system is implemented using sigma-delta (ΣΔ) oversampling readout from two sensors, so that the linearity of a small sensor can be exploited at a low data rate while still providing SNR at a higher data rate with a large sensor. An adaptive linearity calibration technique allows the features of the two sensors to be combined.

3.2 Decimation

Oversampling techniques are desirable for the sensor readout because they are coupled with decimation filtering. One or more decimation stages follow the oversampling modulator. The function of the decimator is to both digitally filter and downsample the output bitstream from the modulator. Both the quantization noise of the modulator plus any white noise in the output spectrum is filtered by the decimator, so that any noise above \( f_b \), the cutoff frequency of the filter, is removed. Figure 3.1 shows the frequency domain representation of a system with white-noise which has been sampled at a sampling rate \( f_s \). The total noise power is distributed between zero and the sampling frequency. The digital low-pass filter removes the fraction of the noise that resides above \( f_b \). Since the output spectrum will always have the same total noise power from dc to \( f_s \), the noise power remaining after filtering is related to the ratio of \( f_s \) to \( f_b \), which is called the oversampling ratio (OSR). In the case of white noise, the reduction in noise voltage is equal to the square root of the
OSR, representing a 3 dB/octave noise improvement with oversampling ratio. Notice that the improvement in the total noise with oversampling ratio can be greater than 3 dB/octave when the noise is shaped, as is the case with the quantization noise introduced in the modulator loop.

The digital low-pass filtering of the output signal is the process by which the sensor $kT$ on $C$ noise is removed. Since the noise is injected at the input to the modulator, this noise is shaped only by the low-pass signal transfer function of the modulator, which is ideally flat within the baseband. This key feature of the decimation filters for oversampling systems is exploited in the use of $\Sigma\Delta$ modulation as the readout technique.

![Diagram](image.png)

**Figure 3.1**: Digital low-pass filtering to remove $kT/C$ noise.

### 3.3 System Block Diagram

The following section presents a sensing system based on the motivations and principles just described. The sensing scheme exploits the characteristics of differently sized sensors to optimize both the resolution and linearity of an open-loop system. A small sensor that is very linear is used to calibrate a larger sensor, which is more sensitive to changes in pressure and yields the desired resolution when used in conjunction with oversampling techniques.

Figure 3.2 shows a block diagram of the sensing system. The system utilizes two sense capacitors, $C_L$ and $C_S$. Oversampling techniques are used to reduce the $kT$ on $C$ noise of the sensing interface. The amount of noise reduction is proportional to the oversampling ratio, which is the ratio of the sampling clock frequency $f_s$ to the desired output data rate of the sensor. Oversampling of each sensor is performed with a single-bit $\Sigma\Delta$ modulator. Both signal paths contain identical decimators, $D_1$ and $D_2$ so identical phase delay is introduced to both signal paths.

The area of sensor $C_L$ is larger than the area of sensor $C_S$. $C_L$ is selected to produce large capacitance change and the desired SNR at a given oversampling ratio
Figure 3.2: Block diagram of capacitive pressure sensing system.
and data rate $f_1$. However, it is more nonlinear when compared with $C_S$. $C_S$ provides a more linear characteristic and is hence used as a linearity calibration standard. The change in capacitance of $C_S$ is significantly smaller than that of $C_L$, and must be compensated for by further decimation of the output to a slower rate $f_2$ to maintain the desired SNR. The main output is provided from the signal path containing $C_L$, and is continuously calibrated against the output from the small sensor $C_S$.

The calibration correction is determined by comparing the output from sensor $C_S$, at the slow rate $f_2$, with the main output further decimated to $f_2$ to match the output speed of the calibration standard. An adaptive algorithm is used to update the correction to the main output at the slow data rate $f_2$, while the calibrated main output is available at much higher rate $f_1$. Although the calibration speed may be much lower than the main output speed, the update rate of the calibration need only be fast enough to capture the static or very slowly-varying nonlinearity. The calibration operates continuously, instead of off-line, using the actual input signal to drive the calibration, therefore avoiding a separate calibration period. The effects of temperature and aging on linearity are automatically removed. This calibration procedure is detailed in a later chapter.

Because the oversampling ratio of the calibration loop $f_s/f_2$ is large, the order of the modulator in that loop can be lower than that in the main loop. A second-order and a third-order modulator were chosen for the calibration loop and the main loop, respectively. The orders are determined from the desired SNR, $f_1$, $f_2$, and the full-scale capacitance changes of $C_L$ and $C_S$.

### 3.4 Selection of System Parameters

The sensing system is designed to operate using a large sensor $C_L$ optimized for operation over a pressure range of 0-35 psi. Based on the membrane thickness and gap dimension of the process, the sensor for such operation is a 240 $\mu$m circular membrane. The nominal capacitance of this sensor is 1.6 pF. The small sensor $C_S$ was selected to provide a desirable linearity (< 0.05%) over the pressure range of the large sensor. This sensor is a 130 $\mu$m radius circular membrane, and has a nominal capacitance of 0.48 pF. Table 3.1 shows the relevant parameters of these two sensors. Since the gap dimension can vary, the results are shown for gap thicknesses of both 1.0 $\mu$m and 1.7 $\mu$m.

Based on the fundamental signal to noise limitation described in the previous section, the maximum SNR with no oversampling of these sensors can be calculated from the following equation which implies a reference voltage $V_R$ of 1 V

$$ SNR \approx 10 \log_{10} \left[ \frac{1}{8} \cdot \left( \frac{kT}{C_R} \right)^{-1} \cdot \left( \frac{\Delta C}{C_R} \right)^2 \right]. \tag{3.1} $$

Table 3.2 indicates the SNR before oversampling for the two sensors, as well as the
oversampling ratio necessary to achieve 86 dB of resolution from each. A second set of SNRs is calculated based on assumed ΔC/C_R of 0.1, as well as smaller nominal capacitances. These calculations are shown because they represent early assumptions about the sensors and led to original system design parameters. In practice the sensors fabricated were larger than originally planned.

For sensors designed for use over a 35 psi range, the maximum resolution of the large sensor with no oversampling is about 75 dB, or 56 dB with a 10% full-scale change in capacitance. The small sensor is limited to 39 dB of resolution over this range. Oversampling ratios of at least 1000 and 4000 are necessary to achieve the desired SNR. The oversampling ratios were conservatively implemented as 2048 and 32768 for the main and calibration loops, respectively. To accomplish these oversampling ratios with an output data rate f_1 of 1 kHz, the system clock rate is greater than 2 MHz. The calibration rate f_2 is 16 times lower than f_1.

### Table 3.2
Calculation of sensor oversampling ratio (OSR) for SNR improvement to 86 dB.

<table>
<thead>
<tr>
<th>Sensor</th>
<th>Radius</th>
<th>Gap</th>
<th>Thickness</th>
<th>ΔC/C_R</th>
<th>SNR</th>
<th>OSR for 86 dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>C_L</td>
<td>240 μm</td>
<td>1.0 μm</td>
<td>10 μm</td>
<td>1.3/1.6</td>
<td>75 dB</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>1.0 μm</td>
<td>10 μm</td>
<td>0.1</td>
<td>56 dB</td>
<td></td>
<td>1024</td>
</tr>
<tr>
<td>C_S</td>
<td>130 μm</td>
<td>1.0 μm</td>
<td>10 μm</td>
<td>0.012/0.48</td>
<td>40 dB</td>
<td>32768</td>
</tr>
<tr>
<td></td>
<td>1.0 μm</td>
<td>10 μm</td>
<td>0.1</td>
<td>51 dB</td>
<td></td>
<td>4096</td>
</tr>
</tbody>
</table>
3.5 Summary

The sensing system architecture presented has several important features. By using oversampling and linearity calibration, necessitating the use of two sensors, the trade-off between speed, resolution and linearity is relaxed. The resulting system is programmable, so that speed at the output can be traded off for resolution. This is a direct result of the decimation filtering that occurs after oversampling. The linearity calibration of the sensor is continuous, and there is no need for separate calibration periods. The calibration can compensate aging and temperature effects, and eliminates the need for costly calibration of the sensor in the factory. The linearity calibration can be extended to a full calibration by the use of training signals, or by an additional small look-up table to store information about the endpoints.
Chapter 4

Linearity Calibration

The goal of the adaptive linearity calibration is to provide a high-speed calibrated output from the sensing system without necessitating off-line correction. The calibration technique corrects for static and slowly-varying nonlinearity of the sensor, and also corrects for static nonlinearity in the ADC. The calibration is performed with a nonlinear filtering process. This chapter includes an introduction to adaptive calibration, and describes in detail the calibration structures and algorithms applied to the sensing system.

4.1 Motivation for Adaptive Calibration

Calibration of a sensor is necessary to properly correlate the output of the sensing system with the magnitude of the physical variable being sensed. For linear sensors, this calibration process may be as simple as calculating the offset and the gain error of the output over a certain range of inputs, thus calibrating the end-points of the sensor characteristic [34]. For sensors with nonlinear characteristics, information at more than two points is required. To perform this type of calibration, the sensor is driven at a number of known reference points, and the resulting sensor output is stored in on-chip memory, typically in look-up table format. Intermediate points can be interpolated from the calibration data [13, 53, 54].

To compensate for aging, or long-term shifts on the sensor characteristic, the sensor may have to be periodically re-calibrated. In addition, environmental or temperature changes will also cause the sensor to require re-calibration. One solution to the temperature calibration is to add another dimension to the look-up table type calibration, so that the sensor characteristic shift with temperature can be interpolated from that data using an on-chip temperature sensor [55, 19]. Obviously for a sensor that has linear temperature sensitivity, the amount of data that needs to be stored is minimal. Attempts to reduce the amount of nonlinearity of in the temperature sensitivity of the sensor are therefore worthwhile.
The cost of calibrating a sensor can be a large fraction, up to one-third, of the total cost of a sensor product [19]. Both the hardware and the time required to perform calibration are factors in this cost. To reduce the costs of calibration, we explore the possibility of using an adaptive and continuous calibration, that not only reduces the hardware required for full sensor calibration over a temperature range, but compensates for aging effects which would otherwise require removing a sensor from its environment for re-calibration.

4.2 Sensor Linearity Calibration

![Sensor Characteristics with Temperature](image)

**Figure 4.1:** Sensor characteristics with temperature.

Figure 4.1 shows an example of a nonlinear sensor characteristic drawn for several temperatures. Look-up type calibration of the sensor requires applying signal inputs $q_0 \cdots q_N$ to the sensor at temperatures $T_0 \cdots T_M$, resulting in approximately $N \times M$ calibration points $(q_i, T_j, C_{ij})$. While the sensor is in operation, both $C$ and $T$ are measured on-chip, and $q$ is interpolated from this data. This type of calibration is typical of many sensors.

The motivation for continuous sensor calibration is to avoid look-up table calibration to linearize the sensor. An adaptive system can be used to perform the linearizing function in a continuous manner. Figure 4.2 shows a simple block diagram example of an adaptive system used to calibrate a sensor. The input $C_L$ is corrected by an adjustable-coefficient filter, producing a linearized output $C_{lin}$. This output is compared with a reference signal $C_S$ to produce an error signal $\varepsilon_k$. This error is used in conjunction with an adaptive algorithm to continuously update the correction coefficients. This example implies that $C_L$ and $C_S$ are at equal data rates. The advantage in using this technique is that the correction can be determined at a low data rate where the calibration standard $C_S$ is available, but can be applied to $C_L$ at a much higher data rate, as described in the previous chapter.
Figure 4.2: Example of adaptive sensor calibration of $C_L$ with reference signal $C_S$.

Unlike look-up calibration, the adaptive calibration does not require the application of the test points $(q_i, T_j)$. Instead, a sensor characteristic $C_{lin}$, corresponding to a range of pressure inputs, continuously results from the calibration. A very important point is that because $C_{lin}$ results from a linearity calibration only, it does not provide an absolute mapping to the sensor input $q$. Full calibration, which will give the absolute value of $q$, requires knowledge of the gain and offset in the measurement of $C_{lin}$. To completely calibrate the signal so that the measurement can be related to the absolute value of the input $q$, the endpoints of the characteristic must be measured. The endpoints of $C_{lin}$, $C_{lin}(q_0)$ and $C_{lin}(q_{max})$ are measured at two known inputs, $q = q_0$ and $q = q_{max}$, respectively. The fully calibrated pressure output $q_{cal}$ is calculated from this information

$$q_{cal} = [C_{lin} - C_{lin}(q_0)] \times \frac{q_{max} - q_0}{C_{lin}(q_{max}) - C_{lin}(q_0)}.$$

To compensate for temperature and aging effects, the endpoints must be periodically re-calibrated.

Because calculation of the endpoints is necessary for a full calibration, the calibration of the interior points is referred to as linearity calibration. This distinguishes the calibration explored in this thesis with a full calibration of the sensor.

4.3 Methods for Adaptive Linearity Calibration

In general, an adaptive system is used to continuously monitor or filter a time-varying input, as described in the example of Figure 4.2. The monitoring is accomplished using a combination of a filter structure and an adaptive algorithm. The structure of an adaptive system is the way in which it is realized in terms of its parameters. Depending on the order of operations in the filter, effects such as numerical rounding
error can be minimized. Associated with the structure and the internal parameters is a method, or adaptive algorithm, for continuously updating the parameters. A variety of structures and algorithms can be applied to the adaptive calibration of the pressure sensor. Tradeoffs can be made between complexity in the filter structure and the convergence requirements of the algorithm. Both of these aspects of an adaptive calibration system will be described in this section.

**Figure 4.3:** Adaptive linear combiner.

Figure 4.3 shows an example of a simple adaptive linear combiner, which is a basic structure used in adaptive systems. The inputs are denoted \( x_0 \cdots x_N \), where \( N \) is the number of weights in the combiner. When the inputs represent sequential discrete time samples of an input signal, the combiner implements a transversal \( N \)-tap FIR filter. In this case, the output \( y_k \) is

\[
y_k = y(kT) = \sum_{i=0}^{N} w_i x((k - i)T)
\]  

where \( k \) denotes a time index and \( T \) is the discrete sampling interval. The \( x_i \) inputs to the weights are delayed samples of the input \( x \). This type of structure is typical in channel equalization problems [56]. The inputs to the linear combiner may also represent spatial samples, meaning that they represent simultaneous inputs from several sources at timepoint \( k \), so that the output is

\[
y_k = y(kT) = \sum_{i=0}^{N} w_i x_i(kT).
\]

This case arises for adaptive beamforming problems or sensor array problems, where signals from several sources are adaptively weighted [56].

The algorithm associated with the adaptive combiner continuously changes the filter weights \( w_0 \cdots w_N \). The rate of change of the filter weights relative to the
bandwidth of the input signal determines the stability of the system, the speed of
adaptation, and the resulting accuracy of the filter. The complexity of the algorithm
ultimately determines the performance of the system in terms of these figures of merit.

Two competing trade-offs in the adaptive algorithm are the speed of adaptation
and final accuracy of the weights [57]. The specific application determines which of
these is more important. In some cases, it is important that the weights \( w_i \) accurately
model the system after some time, but this means that the adaptive algorithm may
have to adjust the weights more slowly. Conversely, the system may need to adapt
very quickly, but the accuracy of the weights will not be as good as in the slowly
adapting case. If the system adapts too quickly, the adaptive system may become
numerically unstable.

Linked closely with the performance of the adaptive system is the history main-
tained in the system [57]. A system which retains a large amount of past information
will not be able to adapt very quickly, however it will be able to accurately model the
statistics of the signal. Alternately, a system which has a short memory will be able
to adapt very quickly, but will compromise its ultimate accuracy.

Systems with short memory lend themselves well to block least squares type algo-
rithms. Block least squares involves solving a matrix inverse problem of size \( M \), once
every \( M \) samples. Because \( M \) is presumably small, this matrix inverse computation
is not very costly. Systems which retain a longer memory instead lend themselves to
recursive algorithms, such as the traditional least mean squares (LMS) algorithms.
These type of algorithms use only current information and iteratively solve the error
minimization problem, making small steps at every iteration, but never actually solv-
ing a full matrix inverse. These type of algorithms may include a windowing factor
so that past information eventually fades.

In summary, several features of the adaptive system control its ultimate perfor-
mance. They include the structure of the adaptive filter and the adaptive algorithm
used to update the filter coefficients, as discussed. Several other factors also impact
the performance, including the performance cost function and the statistics of the
input signal. Each of these features will be discussed in the derivation of the adaptive
structure and the adaptive algorithm used to implement the sensing system.

### 4.4 Calibration Structure

Figure 4.4 shows the application of the simple adaptive combiner to the sensor cali-
bration. In this figure, \( D_L \) and \( D_S \) represent the resultant outputs of the large and small
sensors, respectively, after one stage of decimation by \( D1 \) as shown in Figure 3.2. The
adaptive system adds \( N + 1 \) weighted functions of \( D_L \) to generate a desired response
\( C_{\text{lin}} \) as follows:

\[
C_{\text{lin}} = \sum_{i=0}^{N} w_i f_i(D_L).
\]  

\[ (4.4) \]
Figure 4.4: Simple adaptive structure for sensor calibration.

The feedback in the adaptive loop minimizes the error between $C_{\text{lin}}$ and $G \times D_S$, where $G$ is an implicit gain factor which compensates for the difference in full-scale capacitance of the two sensors. In this case the inputs to the adaptive combiner, $f_1(D_L)$, represent a spatial array of samples because the input $D_L$ at time $k$ is shared, and only the functions $f_i$ are spatially variant.

A simple example is used to discuss this structure. Assume that the characteristics of the sensors are described by

\begin{align}
D_L &= K_1 q^2 \\
D_S &= K_2 q \tag{4.5}
\end{align}

Using the adaptive functions

\begin{align}
f_0(x) &= 1 \\
f_1(x) &= x^{\frac{1}{2}} \tag{4.7}
\end{align}

and selecting the samples $q_k$ from a random distribution, we find that the convergence using a very simple algorithm is quite good. This is because the square-root function can precisely invert the second-order nonlinearity of the large sensor. In reality however the sensor is not described by a simple polynomial, and this simple structure does not perform well when realistic data is used. The reason is that fractional powers of the input are required for correction. These functions may be difficult to predict and implement.

To alleviate the difficulties with such a simple structure, the adaptive structure is implemented as shown in Figure 4.5. In this figure $D_L$ and $D_S$ represent data from
both the large and small sensors respectively after one stage of decimation by $D_1$. The corresponding data rate is $f_1$. $DD_L$ and $DD_S$ represent the output of both the large and small sensors, respectively, after two stages of decimation by both $D_1$ and $D_2$. The corresponding data rate is $f_2$, where $f_2 < f_1$. In this implementation, the loop tries to minimize the error between $DD_L$ and $C_{L,est}$, an estimate of the large sensor capacitance generated from polynomial sums of pressure. The notion of pressure is generated from the small sensor. To within the linearity of this small sensor, $DD_S/A$ represents $q_{DD}$, the pressure at the slow data rate $f_2$. By minimizing this error, the loop creates a mechanism for generating $DD_L$ from $q_{DD}$, which also allows it to find $q_D$ from $D_L$ at the higher data rate $f_1$. This requires a second search loop. The output of the second loop is that fast data rate pressure $q_D$ and it can in turn be used to generate the linearized version of $D_L$, $C_{cal}$.

This structure has the advantage that the functions of $DD$, or implicitly $q$, are polynomials. Therefore, the final resolution of the calibration will depend on the order of the polynomials used, and the selection of the adaptive functions requires only the selection of the polynomial type and order. Clearly, however, we exchange the complexity of the implementation with the ease of the algorithm, since this new structure requires two feedback loops.

In this particular structure, the input variables corresponding to $x_0 \cdots x_N$ in the linear combiner are deterministic, since they are all functions of $D_L$ or $D_S$ at a particular timepoint $k$. Because of this, the solution of the adaptive problem
itself well to least squares techniques, which do not assume any a priori statistics of the input samples nor do they attempt to find them.

4.5 Adaptive Algorithm

In solving the numerical problem associated with this structure, we seek a least sum of squares fit of the last \( M \) samples of data to an \( N \)th order polynomial. Because the information at each time point is not stochastic, past information must be used in the curve fitting. The least square fitting can be performed in a block or recursive algorithm. When performed recursively, old data can be maintained in the system, and the amount of computation may be reduced. Additionally, least squares computation allows for incorporating decimation into the adaptation for a wideband calibration, as will be discussed.

4.5.1 Polynomial Model

There are two principal methods for least square computation, block and recursive least squares. In either case, at each time step we are solving the least squares fitting problem over \( M \) samples. This is represented by the following matrix equation

\[
\begin{bmatrix}
  P_0(x_k) & P_1(x_k) & \cdots & P_N(x_k) \\
  P_0(x_{k-1}) & P_1(x_{k-1}) & \cdots & P_N(x_{k-1}) \\
  \vdots & \vdots & \ddots & \vdots \\
  P_0(x_{k-M}) & P_1(x_{k-M}) & \cdots & P_N(x_{k-M})
\end{bmatrix}
\begin{bmatrix}
  w_k \\
  y_k
\end{bmatrix}
= w_k = y_k
\] (4.9)

where \( w_k \) is a vector of length \( N \) and \( y_k \) is a vector of length \( M \). \( x_k \) represents the input sampled at time \( kT \). The most simple polynomials \( P_N(x) \) are:

\[
P_0(x) = 1 \\
P_1(x) = x \\
\vdots \\
P_N(x) = x^N
\] (4.10)

This equation can be re-written

\[
Vw_k = y_k,
\] (4.11)

where the matrix \( V \) is a Vandermonde matrix of the form

\[
V = \begin{bmatrix}
  1 & x_0 & x_0^2 & \cdots & x_0^N \\
  1 & x_1 & x_1^2 & \cdots & x_1^N \\
  \vdots & \vdots & \vdots & \ddots & \vdots \\
  1 & x_M & x_M^2 & \cdots & x_M^N
\end{bmatrix}.
\] (4.12)
This matrix is notoriously ill-conditioned. The condition number of this matrix, or ratio of maximum to minimum entry, may become quite large because the columns are not orthogonal. For polynomial order of five or less, this matrix may be invertible but becomes near singular for higher order polynomials. Unfortunately, the sensor capacitance data requires polynomials of at least sixth order to accurately model their behavior. Figure 4.6 shows the relative mean square error ($L_2$) and relative maximum error ($L_{\infty}$) in the sensor characteristic after least-squares polynomial fitting.

![Graph showing error versus polynomial order](image)

**Figure 4.6:** Both maximum and mean-square error versus polynomial order for least-squares polynomial fitting of the 240 µm sensor capacitance over 35 psi.

Solutions using these types of polynomials fit better for low values of the input $x_k$, but it is advantageous to use other types of orthonormal polynomials to formulate the problems. Chebyshev polynomials were selected instead. The matrix formation is much better conditioned over pre-defined range of inputs. Additionally the polynomials themselves are defined recursively so higher order polynomials can be quickly calculated.

The Chebyshev polynomials are orthogonal with respect to following weighting
function
\[ g(x) = \frac{1}{\sqrt{1 - x^2}} \] \hspace{1cm} (4.13)

over the interval \( x \in [-1, 1] \). The first few polynomials are
\[ T_0(x) = 1, \]
\[ T_1(x) = x, \]
\[ T_2(x) = 2x^2 - 1, \]
\[ T_3(x) = 4x^3 - 3x, \text{ and} \]
\[ T_4(x) = 8x^4 - 8x^2 + 1. \] \hspace{1cm} (4.14)

In general they are defined by the following recursion
\[ T_0(x) = 1, \]
\[ T_1(x) = xT_0(x) = x, \]
\[ T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x), n \geq 0. \] \hspace{1cm} (4.15)

Because these polynomials are orthogonal, they form a better basis set for solving the least squares problem. Now the coefficient matrix formed by these polynomials is much more well-conditioned (invertible).

### 4.5.2 Block Least Squares

The least squares computation can be performed using a block of data, meaning that the optimal weights are computed over a finite number of samples. Solving the least squares problem for every block of data entails solving the normal equations
\[ A^T A w = A^T y \] \hspace{1cm} (4.16)

where \( A \in \mathbb{R}^{M \times N} \) is of the form
\[
\begin{bmatrix}
T_0(x_k) & T_1(x_k) & \cdots & T_N(x_k) \\
T_0(x_{k-1}) & T_1(x_{k-1}) & \cdots & T_N(x_{k-1}) \\
\vdots & & & \\
T_0(x_{k-M}) & T_1(x_{k-M}) & \cdots & T_N(x_{k-M})
\end{bmatrix}
\] \hspace{1cm} (4.17)

and \( y \) is a vector of the form \([f(x_k) f(x_{k-1}) \cdots f(x_{k-M})] \)\(^T\).

To solve the normal equations, a \( QR \) decomposition of \( A \) is performed. \( Q \) is an orthogonal matrix and \( R \) is an upper triangular matrix, such that \( A = QR \). Since \( Q \) is orthogonal, \( Q^{-1} = Q^T \), and the normal equations can be solved by taking the transpose of \( Q \), multiplying it by \( y \) and back substituting into the upper triangular matrix \( R \) so that
\[ w_{LS} = R Q^T y. \] \hspace{1cm} (4.18)
4.5. ADAPTIVE ALGORITHM

Here it is implicit that \( w_{LS} \) represents the optimal weights at timepoint \( k \). The QR solution method is preferred over Gaussian elimination for a number of reasons, including numerical stability.

4.5.3 Recursive Least Squares

The block least squares technique requires computing the matrix inverse or QR decomposition for every \( M \) samples, where \( M \) is the block length. Computing the matrix inverse can be costly. In addition, it may be useful to compute new weights at each time step, instead of waiting until a sufficient block of data can be acquired. To accomplish this, the matrix inverse can be computed recursively, so that for each new data point, new weights are computed.

Assume that a QR factorization for the matrix \( A \) exists, so that \( QR = A \). A new data point \( x_k \) increases the number of data points in the least square problem of Equation 4.16 by one, and causes a row to be appended to both \( A \) and \( y \) as follows:

\[
\begin{bmatrix}
T_0(x_k) & T_1(x_k) & \cdots & T_N(x_k)
\end{bmatrix}
\begin{bmatrix}
w_0 \\
\vdots \\
w_N
\end{bmatrix} =
\begin{bmatrix}
f(x_k) \\
y_{k-1}
\end{bmatrix}
\]  

(4.19)

The matrix on the left hand side of this equation can be represented as

\[
\tilde{A} = 
\begin{bmatrix}
T_0(x_k) & T_1(x_k) & \cdots & T_N(x_k)
\end{bmatrix}
\begin{bmatrix}
a^T \\
A
\end{bmatrix}
\]  

(4.20)

and the new right hand side is

\[
\tilde{y} = 
\begin{bmatrix}
f(x_k) \\
y
\end{bmatrix}
\]  

(4.21)

so that the new least squares solution involves the QR decomposition of \( \tilde{A} \) in

\[
\tilde{A}w = \tilde{y}.
\]  

(4.22)

It can be shown that

\[
\begin{bmatrix}
1 & 0 & \cdots & 0 \\
0 & Q^T & & \\
\vdots & & Q^T & \\
0 & & & Q^T
\end{bmatrix} \cdot \tilde{A} = 
\begin{bmatrix}
q^T \\
R
\end{bmatrix} = H
\]  

(4.23)

where \( H \) is has the form

\[
H = 
\begin{bmatrix}
X & X & X & X \\
X & X & X & X \\
0 & X & X & X \\
0 & 0 & X & X
\end{bmatrix}.
\]  

(4.24)
Since \( H \) is an upper Hessenberg matrix having elements on the sub-diagonal, it can be converted to an upper triangular matrix \( \tilde{R} \) by the use of Givens rotations. Givens rotations are rank-two orthogonal matrices of the form

\[
G = \begin{bmatrix}
\cos \theta & \sin \theta \\
-\sin \theta & \cos \theta
\end{bmatrix}.
\] (4.25)

For a vector \( r = G^T x \), \( r \) is obtained by rotating \( x \) counter clockwise by \( \theta \) radians. Givens rotations can be used to zero out a column element by selecting the angle of rotation. For example for a vector \( x = [a \ b]^T \),

\[
G^T x = \begin{bmatrix}
c & -s \\
s & c
\end{bmatrix} \begin{bmatrix}
a \\
b
\end{bmatrix} = \begin{bmatrix}
z \\
0
\end{bmatrix},
\] (4.26)

where \( s = \sin \theta \) and \( c = \cos \theta \) for the rotation angle \( \theta \). It is simple to show that the values of \( s \) and \( c \) can be selected to zero out one element in the right hand side of the equation as indicated. The resulting values are:

\[
c = \frac{a}{\sqrt{a^2 + b^2}}, \text{ and}
\]
\[
s = \frac{-b}{\sqrt{a^2 + b^2}}.
\] (4.27)

In general the Givens rotation \( G_{ik} \) are rank-two corrections to the identity matrix, but their two-dimensional size can be exploited in a product of Givens rotations.

The Givens rotations, \( G_1 \cdots G_N \), are applied to the matrix \( H \) of Equation 4.24 so that \( G_N^T \cdots G_1^T \cdot H = \tilde{R} \) is upper triangular. \( G_N^T \cdots G_1^T \) is a product of Givens rotations. The new \( QR \) factorization becomes

\[
\tilde{A} = \tilde{Q} \tilde{R}
\] (4.28)

where \( \tilde{Q} \) is also computed from the Givens rotations as follows

\[
\tilde{Q} = \begin{bmatrix}
1 & 0 & \cdots & 0 \\
0 & & & \\
\vdots & & Q \\
0 & & &
\end{bmatrix} \cdot [G_1 \cdots G_N].
\] (4.29)

The new value of the least-squares solution can be computed at timepoint \( k \), by computing

\[
w_k = \tilde{R} \tilde{Q}^T \tilde{y}.
\] (4.30)

It is important to note that the Givens rotations can be directly applied to \( Q^T y \) to yield \( \tilde{Q}^T \tilde{y} \) so that there is no need to store \( Q \) and \( y \) separately, only the vector \( Q^T y \).
4.5. ADAPTIVE ALGORITHM

It may be useful to weight the data in a least squares computation, so that in general, the problem to be solved is

\[ A \hat{w} = y \rightarrow W A \hat{w} = W y, \]  

(4.31)

where \( W \) contains weighting terms on the main diagonal and cross-weighting terms off-diagonal. In particular, windowing the data with respect to the time index is important for removing the effects of the old data. Since the recursive least squares algorithm presented above retains all the data history, the effects of old data will take a long time to fade out. Instead, a geometric weighting can be applied. A typical weighting has the form

\[ g(i, n) = \lambda^{i-n}. \]  

(4.32)

This means that the error being minimized now has the following form

\[ \sum_{i=1}^{N} \lambda^{i-N} \varepsilon^2(f_i). \]  

(4.33)

The optimum weight vector is a solution to the modified normal equations

\[ A^T(W^TW)A \hat{w}_{LS} = A^T(W^TW)y \]  

(4.34)

for \( W \) invertible.

Since the type of weighting in Equation 4.32 is geometric, rows can be added to the least square problem by multiplying the old matrix \( A \) and \( y \) by \( \lambda \). For example, each row update now involves solving the following equation

\[ \begin{bmatrix} T_0(x_k) & T_1(x_k) & \cdots & T_N(x_k) \\ \lambda A & \lambda y \end{bmatrix} = \begin{bmatrix} f(x_k) \\ \lambda y \end{bmatrix}. \]  

(4.35)

For solving the recursive least squares problem, the \( QR \) factorization must be updated. More specifically \( R \) and \( Q^T y \) are adjusted at each step. The \( QR \) factorization of \( \lambda A \) is simply \( Q(\lambda R) \), \( R \) and \( Q^T y \) are updated as follows

\[ Q^T y \rightarrow \begin{bmatrix} f(x_k) \\ \lambda Q^T y \end{bmatrix}, \quad \text{and} \]

\[ R \rightarrow \begin{bmatrix} a^T \\ \lambda R \end{bmatrix}. \]  

(4.36)

(4.37)
4.6 Wideband Calibration

The calibration functions only when signal is present at data rate $f_2$ or lower. When no information is present, the calibration must remain idle until enough signal power is present at that frequency or lower. Depending on the application, the system may be trained using a known electrical signal at low frequencies. Instead, the order in which operations are performed in the calibration can be reversed, so that any signal below $f_1$, the higher data rate, will allow the calibration to function. In this more wideband calibration, a coarse calibration is performed at data rate $f_2$, and the resulting calibration weights are digitally low-pass filtered to provide the desired accuracy. This is demonstrated in Figure 4.7.

![Diagram of Wideband Calibration](image)

**Figure 4.7:** Wideband adaptive structure for sensor calibration.
4.6.1 Calibration/Decimation

The wideband calibration can be understood by first considering the comparison between least squares fitting to a constant and averaging. When we fit data to polynomials of order zero, the least squares problem reduces to

\[
\begin{bmatrix}
1 \\
\vdots \\
1
\end{bmatrix} \cdot \begin{bmatrix} w_0 \\
\vdots \\
w_N \end{bmatrix} = \begin{bmatrix} f_1 \\
\vdots \\
f_N \end{bmatrix}.
\]

(4.38)

In this case, \(w_{LS}\) reduces to

\[
w_{LS} = (A^T A)^{-1} \cdot A^T y = \frac{1}{N} \sum_{i=1}^{N} f_i
\]

(4.39)

which is the average value of the \(f_i\)s. Least squares fitting of \(N\) data points is therefore the same as accumulate and downsample (sinc) filtering of the data to provide data rate reduction (decimation) by \(N\).

The approach to a combined decimation/least-squares filtering can be readily accomplished using the recursive least-squares algorithm as follows: equally weight the samples within a block \(N\), where \(N\) is the downsampling ratio, \(f_1/f_2\). After the least squares fit of the \(N\)th data point, keep only that sample. \(\lambda\) is changed only every \(N\) samples, but is equal to 1 for samples \(1 \cdots (N - 1)\).

The wideband recursive least-squares algorithm is described in Figures 4.8 and 4.9. The first algorithm, compute-model(), is used to compute the model for the sensor at data rate \(f_1\). \(D_L\) and \(D_S\) are the calibration inputs, also indicated in Figure 4.7. The iterative loop continuously generates the best weights \(w_{LS}\) at data rate \(f_1\). The second algorithm, calibrate() is used to update the nonlinear output \(D_L\). It performs a search function using decimated weights, \(w_{1D}\), and an numerical gradient estimate of the error surface. The decimation and filtering functions in both of these algorithms are shown as block operations, but are computed continuously in practice.
Figure 4.8: Wideband calibration algorithm - Part I

```plaintext
compute-model()
{
    compute Chebyshev filter taps for first decimator;
    filter and decimate both input channels to $f_1$;
    compute scale factors so that $D_L, D_S \in [-1,1]$;
    select polynomial order $N$;
    select weighting parameter $\lambda$;
    $k = 0$;
    while ($k < \text{Data\_Length}$) {
        $x_k \leftarrow D_S(k)$;
        $f_k \leftarrow D_L(k)$;
        compute $N$ polynomials of $T_i(x_k)$
        if ($k < N$) {
            add row $T_i(x_k)$ to $\lambda A$ and $f_k$ to $\lambda y$;
        }
        if ($k = N$) {
            compute initial $QR$ decomposition;
            store $R$ and $Q^T b$;
        }
        if ($k > N$) {
            add row $T_i(x_k)$ to $\lambda R$ and $f_k$ to $\lambda Q^T y$;
            compute Givens rotations;
            apply Givens rotations to $R$ and $Q^T y$;
            compute new least squares solution $w_{LS} \leftarrow R \backslash (Q^T b)$;
        }
        $k = k + 1$;
    }
}
```
4.6. WIDEBAND CALIBRATION

Figure 4.9: Wideband calibration algorithm - Part II

```c
calibrate()
{
    determine error tolerance errtol;
    filter and decimate weights \( w_i \) to \( w_{iD} \) at \( f_2 \);
    \( k = 0 \);
    while (k < Data.Length) {
        \( f_k \leftarrow D_L(k) \);
        estimate \( q_D \);
        recursively compute data estimate \( c_k = w_{iD} T_i(q_D) \);
        compute error \( \varepsilon_k = f_k - c_k \);
        while (|\varepsilon_k| > errtol) {
            compute numerical gradient of error surface;
            use Newton's method to update estimate \( c_k \);
            \( k \leftarrow k + 1 \);
        }
    }
}
```

4.6.2 Simulations/Examples

An example calibration is performed using data generated by the \( \Sigma \Delta \) simulator. A 1.01 kHz sinewave and a corresponding distorted sinewave are input to the block-diagram simulator. A transfer characteristic of the nonlinearity is shown in Figure 4.10. The distorted input has a maximum INL of about 10%. The corresponding outputs of the simulator for the undistorted and the distorted cases are shown in Figures 4.11 and 4.12, respectively. Using the calibration algorithm with 6 weights, the corrected output corresponds to the spectrum shown in Figure 4.13.
Figure 4.10: Simulated transfer function of electrical nonlinearity with linear input indicated.

Figure 4.11: Simulated spectrum of third-order modulator output, serving as calibration standard.
Figure 4.12: Simulated, uncalibrated spectrum of third-order modulator output.

Figure 4.13: Simulated, calibrated spectrum of third-order modulator.
4.7 Stochastic Search Techniques

Stochastic search techniques may be used as an alternative solution technique for adaptive problems. These techniques are based on estimating the search direction based on the input statistics, and are very popular because of their ease of implementation. A brief review of these techniques is provided here so that they may be contrasted with the least squares, or exact, techniques which were implemented in the sensing system.

Stochastic techniques may be applied to the calibration structures described in this chapter. For the adaptive linear combiner shown in Figure 4.3, the output $y_k$ is formed from the product $x_k^T w_k$, where $w_k$ is the time-varying adaptive weight vector. For sampling interval $k$, the error between $y_k$ and a desired response $\tilde{y}_k$ is

$$\varepsilon_k = \tilde{y}_k - y_k = \tilde{y}_k - x_k^T w_k. \quad (4.40)$$

 Generally the most efficient form of the error to minimize is the mean square error (MSE), which is defined as

$$MSE = E[\varepsilon_k^2] = E[\tilde{y}_k^2] + w^T E[x_k x_k^T]w - 2E[\tilde{y}_k x_k^T]w$$

$$= E[\tilde{y}_k^2] + w^T R w - 2P w, \quad (4.41)$$

where $R = E[x_k x_k^T]$ represents the input correlation matrix, and $P = E[\tilde{y}_k x_k^T]$ is the cross-correlation matrix. It can be shown that for a quadratic performance surface and a statistically stationary input, only one minimum, $w^*$, exists.

Several stochastic search techniques can be used to find the minimum of the mean square error surface, $w^* = R^{-1}P$. Many of these are based on gradient search techniques, which try to minimize the gradient of the error surface:

$$\nabla(MSE) = \frac{\partial(MSE)}{\partial w} = 2R w - 2P. \quad (4.42)$$

Often one approximates the input signal as statistically stationary, so that $R = x_k x_k^T$ can be used to approximate the expected value of $E[R]$ in the gradient. As an example of a search algorithm, Newton's method changes $w$ in the direction of the error surface minimum, so that one iteration can be described as:

$$w_{k+1} = w_k - \mu R^{-1} \nabla_k, \quad (4.43)$$

where $\mu$ is a relaxation factor which can accelerate or slow the convergence rate and $\nabla_k$ is the gradient of the error surface. In this case the Jacobian matrix is approximated as the identity matrix, and this method is therefore very similar to the steepest descent method. The steepest descent (SD) algorithm changes $w$ in the direction of the negative gradient of the error surface, so that

$$w_{k+1} = w_k + \mu (-\nabla_k). \quad (4.44)$$
Alternate algorithms exist, including conjugate gradient techniques, random searches, which are useful for non-quadratic error surfaces, and the sequential regression algorithm (SER), which estimates the input correlation matrix \( R \) using past information instead of only the current information. In general, however, the stability and convergence speed of all these algorithms is determined by the eigenvalues \( \lambda_i \) of \( R \), and the damping factor \( \mu \).

For adaptive systems, the exact gradient of the performance surface is generally not known, since the error surface is not analytically described. Critical to the search algorithm is the technique for estimating this gradient. Least Mean Squares (LMS) is the most common form of gradient estimation

\[
\hat{\nabla}(MSE) \approx \nabla(\varepsilon_k^2) \approx -2\varepsilon_k x_k
\]

which can be implemented in conjunction with any gradient search algorithm.

Although these stochastics techniques may be applied to the sensor problem with some success, the input signal is not statistically stationary, and these techniques do not provide the best convergence properties. Because they are so widely used, however, they are important to contrast with the least squares techniques. Least squares or exact solutions are more applicable to the adaptive structure chosen for the sensor problem because the inputs to the adaptive algorithm are functionally dependent, and do not have a statistically stationary distribution.

### 4.8 Hardware and Computational Requirements

The hardware required to implement the wideband calibration will be composed of that necessary to store the intermediate variables, such as the weights vector, as well as that required to perform the floating point operations. Since the area of such hardware is so technology dependent, the memory required and the floating point operations (flops) will be listed without reference to area. The calibration runs at a relatively low data rate, which allows much of the digital computation to be performed in a pipelined, or recursive, fashion.

The memory requirements are listed in Table 4.1 and the number of flops per computation are listed in Table 4.2. The notation \( O(n) \) indicates "order \( n \)". Note that the matrix used to store the initial matrix \( A \) can be reused for in-place computation of the initial \( QR \) decomposition and for subsequent storage of \( R \). Table 4.2 lists the amount of computation for one data point \( x_k \) at the fast data rate \( f_1 \). The computation of \( q_D \) is not included in this table, but the search algorithm requires a number of iterations proportional to the amount of nonlinearity.
## Table 4.1
Storage requirements for implementing wideband adaptive calibration with \( n \) weights.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Storage requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single variables:</td>
<td></td>
</tr>
<tr>
<td>( x_k, f_k, \lambda, D_{L,est}, C_{cal}, q_D, \varepsilon_k \cdots )</td>
<td>( 10 \ O(1) )</td>
</tr>
<tr>
<td>Weights vector ( w_{LS} )</td>
<td>( O(n) )</td>
</tr>
<tr>
<td>Initial matrix ( A )</td>
<td>( O(n^2) )</td>
</tr>
<tr>
<td>Initial vector ( y )</td>
<td>( O(n) )</td>
</tr>
<tr>
<td>Coefficients for ( T_i(x_k) ) computation</td>
<td>( 2O(1) )</td>
</tr>
<tr>
<td>Rank-one matrix update ( T_i(x_k) )</td>
<td>( O(n) )</td>
</tr>
<tr>
<td>Upper-triangular matrix ( R )</td>
<td>( (1/2) \ O(n) )</td>
</tr>
<tr>
<td>( Q^T y )</td>
<td>( O(n) )</td>
</tr>
<tr>
<td>Givens rotations</td>
<td>( 4 \ O(1) )</td>
</tr>
</tbody>
</table>

## Table 4.2
Floating point operations required to implement wideband adaptive calibration with \( n \) weights.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Flops</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compute ( T_i )</td>
<td>( 3 \ O(n) )</td>
</tr>
<tr>
<td>Compute Hessenberg QR</td>
<td>( 3 \ O(n^2) )</td>
</tr>
<tr>
<td>Compute ( Q^T \tilde{y} )</td>
<td>( O(n) )</td>
</tr>
<tr>
<td>Back substitute ( w_k = \tilde{R} \ Q^T \tilde{y} )</td>
<td>( O(n^2) )</td>
</tr>
</tbody>
</table>
Chapter 5

Sigma-Delta Modulation

5.1 Motivation

Unlike a conventional analog-to-digital converter (ADC), which performs one data conversion for every input sample, an oversampling converter digitally processes many samples of an input signal to achieve high resolution from many lower-resolution samples. Often these samples represent a single-bit quantization of the input. With a maximum sampling rate imposed, the number of samples that are filtered and downsampled determines the output frequency. Therefore, oversampling converters trade off resolution of the output signal with the speed of the output.

Sigma-Delta (ΣΔ) modulation is a technique for implementing an oversampling ADC which uses noise shaping quantization, as well as digital filtering, in order to provide more resolution than a simple oversampling technique. The achievable resolution ideally depends on many factors, including the number of integrators in the loop, the number of quantization levels, the order and type of the decimation filters, and the oversampling ratio (OSR). In practice many nonidealities also enter the picture [58].

A ΣΔ ADC is well suited for the sensing system because of the oversampling properties that allow filtering of input-injected kT/C noise with the digital decimators. In addition, nonidealities in the analog domain are traded with processing in the digital domain, and in most cases, precise component matching is not required [58]. The modulator, which forms the analog portion of the ADC, produces a quantized bitstream that can be processed completely in the digital domain. Potentially this processing can be performed off-chip, which is attractive for some sensor applications [59]. Finally, the relatively low-frequency of the mechanical sensor bandwidth makes ΣΔ modulation suitable for this application, since high clock rates are not required to achieve reasonable oversampling ratios and resultant averaging of kT on C noise.
5.1.1 Basic Operation

Figure 5.1: General block diagram of a sigma-delta modulator.

Figure 5.1 shows a block diagram of a generalized $\Sigma\Delta$ modulator. The sampled input is integrated and fed to a quantizer, or ADC. The output from the quantizer, the modulator output, is then converted to an analog quantity with a digital-to-analog converter (DAC) and subtracted from the input. The resulting error is accumulated in the integrators. This discrete feedback process forces the average value of the modulator output to track the input. The order of the modulator is determined by $N$, the number of integrators comprising the loop filter $H(z)$. An important parameter of the system is the oversampling ratio, which is the ratio of the clock frequency $f_s$ to the Nyquist rate $f_b$ of the input signal.

Figure 5.2: Generic single-bit modulator.

Often the internal ADC is implemented with comparator, or simple one-bit quantizer, and the DAC subconverter is implemented with direct feedback, as illustrated in Figure 5.2. In general, however, these subcomponents can be multi-bit, or $M$ bit, increasing the high-frequency starting wordlength and decreasing the amount of oversampling required to reach a resolution target. The ADC subconverter quantizes the signal into an $M$ bit word.
The quantizing function may be viewed as an addition of another signal onto the output, so that the output signal can be represented as

\[ y[n] = x(nT) + q[n], \]  

(5.1)

where \( y[n] \) is the discrete output word, \( x(nT) \) is the sampled analog input signal and \( q[n] \) is the quantization error. For high-order or multibit modulators the quantization error is often approximated as a white noise source that is uncorrelated with the input [58]. For low-order modulators this approximation is less accurate, and the noise may be significantly correlated with the input signal, leading to tones, or limit cycle oscillations, in the output signal [60].

### 5.1.2 Noise Shaping

![Additive white-noise model for modulator.](image)

**Figure 5.3:** Additive white-noise model for modulator.

The additive white noise model of the modulator is useful for designing the loop filter, provided that the modulator is fully simulated to verify its behavior. Assuming that the noise source \( q(t) \) is uncorrelated with the input, the modulator can be represented with the linearized block diagram shown in Figure 5.3. The loop filter is represented by \( H(z) \). The \( z \)-transform of the output can be expressed in terms of the two linear inputs, \( X(z) \) and \( Q(z) \)

\[ Y(z) = STF(z)X(z) + NTF(z)Q(z), \]  

(5.2)

where STF and NTF are the signal transfer function and noise transfer function described by the following equations

\[ STF(z) = \frac{H(z)}{1 + KH(z)}, \]  

(5.3)

and

\[ NTF(z) = \frac{1}{1 + KH(z)}. \]  

(5.4)
The effect of the loop filter $H(z)$ is to shape the quantization noise, or move it higher frequencies than the cutoff frequency of the digital low-pass filter that follows the modulator. The energy distribution achieved by the noise shaping is determined by the number of integrators in the loop, as well as the placement of the poles and zeros of the noise-shaping transfer function. Figure 5.4 shows a sketch of a typical noise-shaping transfer function from a first-, second- and third-order modulator, indicating that the amount of noise shaping improves for higher order modulators.

![Figure 5.4: Illustration of first-, second- and third-order noise shaping.](image)

In Equations 5.3 and 5.4, $K$ represents the effective gain of the comparator. Because the quantizer is performing a nonlinear operation, it will have an associated statistical gain. Interestingly, a gain preceding the quantizer will not effect the decision of the comparator. However it will affect the stability of the linear model. In practice the statistical comparator gain will change depending on the input signal, so that there is no single transfer function describing the quantization noise shaping. For example, as the input level becomes larger, the comparator gain will drop due to larger inputs into the quantizer, until eventually the loop loses stability due to a low overall gain [61].

### 5.1.3 SNR

Assuming that the additive white noise is valid, we can derive the theoretical expected improvement in SNR with oversampling. A uniform quantizer with level spacing $\Delta$ is shown in Figure 5.5. The output $y$ is related to the input $x$ by the following equation

$$y = Gx + e$$  \hspace{1cm} (5.5)
where $Gx$ is the ideal output and $e$ represents the quantization error. If the error is modeled as a white noise source lying between $\pm \Delta/2$, then the rms error is

$$e_{\text{rms}}^2 = \frac{1}{\Delta} \int_{-\Delta/2}^{+\Delta/2} e^2 de = \frac{\Delta^2}{12}.$$ (5.6)

Since this noise is sampled, the total noise power folds entirely into the frequency range $0 < f_s < f_s/2$, so that the noise spectral density is

$$E(f) = e_{\text{rms}} \left(\frac{2}{f_s}\right)^{1/2}.$$ (5.7)

Generic oversampling of the white noise with no noise shaping produces a total noise power in the baseband of

$$n^2 = \int_{f=0}^{f=f_s} E^2(f) df = e_{\text{rms}}^2 \frac{f_s}{f_s} = \frac{e_{\text{rms}}^2}{OSR},$$ (5.8)

or a noise voltage of $e_{\text{rms}}/\sqrt{OSR}$, providing 3 dB of improvement in SNR per octave of oversampling.

Noise shaping can improve this. In a first-order $\Sigma\Delta$ the quantization error in the modulator is differentiated by the loop, so that the modulation noise is $n_i = e_i - e_{i-1}$. The spectral density of this noise is

$$N(f) = E(f)|1 - e^{-j\omega\tau}| = 2e_{\text{rms}}\sqrt{2\tau} \sin\left(\frac{\omega\tau}{2}\right).$$ (5.9)

As illustrated in Figure 5.4, the noise is shaped by the differentiation so that more of the total noise power is at frequencies close to half the sampling frequency. The resulting total noise power in the signal band is

$$n^2 = \int_{f=0}^{f=f_s} |N(f)|^2 df = e_{\text{rms}}^2 \frac{\pi^2}{3}(OSR)^{-3},$$ (5.10)
providing 9 dB of improvement in SNR with every octave of oversampling.

For additional improvements in noise shaping, more integrators are used to shape the noise. In general, for $L$-order noise shaping, the noise spectral density is

$$N(f) = E(f) \cdot 2\sin \left( \frac{\omega T}{2} \right)^L$$

and the equivalent total noise voltage in the baseband is

$$n = e_{\text{rms}} \frac{\pi^L}{\sqrt{2L + 1}} (\text{OSR})^{-(L + \frac{1}{2})}.$$  \hspace{1cm} (5.12)

The noise voltage thus drops by $3(2L + 1)$ dB per octave of oversampling. For second- and third- order modulators, an improvement of 15 dB per octave and 21 dB per octave of oversampling is expected. The high SNR is compromised for stability in higher-order loop filters.

Multibit subconverters can improve the SNR of the modulator. By further quantizing the output into $M$ levels, the rms error is reduced. Since the step size changes ($\Delta \rightarrow \Delta \frac{\Delta}{2^{M-1}}$), the rms error is

$$e_{\text{rms}} = \left( \frac{\Delta}{\sqrt{12}} \right) \frac{1}{2^M - 1}.$$  \hspace{1cm} (5.13)

For large values of $M$, this improves the SNR by roughly $6M$ dB.

## 5.2 Modulator Architectures

### 5.2.1 Selection Based on Sensing System Constraints

The sensing system described in Chapter 3 incorporates two $\Sigma\Delta$ modulators. The calibration loop employs a single-bit, second-order modulator. A single-bit architecture was selected for two reasons. First, the design of the ADC and DAC subconverters is greatly simplified over multibit architectures. Secondly, and most importantly, single-bit modulators offer good linearity since the one-bit DAC has perfect differential nonlinearity (DNL). Since the output from this modulator is used as the linearity calibration standard, the linearity of this modulator is critical.

The implemented loop filter in this converter is second-order. Since the oversampling ratio for this half of the circuit is large, it was not necessary to implement a third- or higher-order converter. A second-order modulator is sufficient for removing the quantization noise because of the very high oversampling ratio. Although a first-order loop would be simpler, it would be extremely likely to have tones, or limit cycle oscillations. This effect improves with converter order, therefore a second-order converter is better from this standpoint. The second-order converter will still have
some tones, but they are less likely to be a problem at the baseband for converters such as this one with high oversampling ratios. If necessary, dithering can be used to improve the randomization of the quantization noise [62].

The block diagram of the modulator architecture, shown in Figure 5.6, is well-known [63]. The topology is both simple and unconditionally stable [64]. The zeros of the quantization noise-shaping transfer function are located at dc. The circuit uses distributed feedback to implement the desired loop filter poles, similar to [65, 66, 67]. Alternate topologies use feedforward techniques [68, 69, 70, 71, 55]. The advantage of the distributed feedback is that it is possible to achieve a very flat passband response. However, the implementation may be larger and use more power than other topologies [72].

![Second-order modulator block diagram](image)

**Figure 5.6:** Block diagram of second-order modulator.

![Third-order modulator block diagram](image)

**Figure 5.7:** Block diagram of third-order modulator.

The primary modulator, which encodes the signal from the large main sensor, is implemented as a third-order loop with single-bit feedback. The third-order architecture was necessary to remove quantization noise at the given oversampling ratio of this
loop (about $10^3$). The selected architecture is shown in Figure 5.7. This architecture was selected because it can be implemented as a simple extension of the second order modulator. This architecture allows for the placement of a complex zero-pair via the feedback path $b$ from the output of the third to the input of the second integrator (the third zero is at dc). Potentially this can improve the noise-shaping by creating a quantization noise null in the baseband.

Because both the second- and third-order modulators are implemented as pipelined structures, they will have an associated delay. This must be compensated either in hardware or software so that the delays of both modulators are identical ensuring proper adaptive calibration. We corrected this delay in the implementation of the decimation filters.

To optimize the delay matching of the two signal paths, the main converter can be implemented as a second-order loop with multi-bit feedback, where the multi-bit feedback compensates for the lower oversampling ratio and provides the ancillary benefit of reduced susceptibility to tones [60]. A disadvantage of this technique however is the added complexity as well as the additional nonlinearity of the DAC subconverter. The $M$-bit DAC must typically be linear to within the resolution of the output of the converter, requiring either very high-accuracy components or some form of digital linearity error correction to relax this constraint [73]. However in this case the adaptive calibration can compensate for this nonlinearity error. Although this is an interesting result, single-bit feedback was selected for the implementation of this modulator for simplicity and consistency with the design of the other modulator.

### 5.3 Modulator Coefficient Selection

#### 5.3.1 Design Procedure

Modulator coefficients are selected for a given architecture based on the desired quantization noise transfer function. The resulting feedback loop must be stable over the input range of interest. Stability can be verified with a number of techniques, including nonlinear analysis [74] and describing functions [75, 61]. However, the stability analysis is most often accomplished by assuming that the modulator can be modeled by linear additive white-noise model. Since the loop is inherently nonlinear due to the quantization function, the mapping procedure from the nonlinear model to the linear one is often achieved in software [76]. In other words, the loop filter can be designed with a standard filter package, using an assumed linear gain for the quantizer, often unity. The resulting poles and zeros are mapped into loop coefficients which are used to perform a behavioral simulation yielding the actual quantizer gain. This leads to an iterative procedure in which the coefficients can be modified in such a way that the basic function of the loop filter is the same, but the comparator gain changes. This process eventually converges when the comparator gains match between the linear
model and the simulated system. A second iteration is required to properly scale the integrators outputs in the modulator so that they fall within the output swing of the operational amplifiers.

The measurement of the comparator gain is critical to the analysis just described. Again, a number of techniques can be used. In this case, the mean of the comparator output and the mean of the comparator input indicate the statistical comparator gain. In this analysis, the comparator gain is measured over a range of input values, and stability is ensured for all corresponding input level and comparator gain pairs, since the stability will degrade as the comparator gain drops. Simulations were performed with a block diagram simulator developed in collaboration with Szajda [77].

### 5.3.2 Second-order Modulator

Figure 5.6 shows the block diagram of the selected second-order modulator architecture. The The z-transform of the output can be expressed in terms of the two linear inputs, $V_{in}(z)$ and $Q(z)$

$$V_o(z) = STF(z)V_{in}(z) + NTF(z)Q(z),$$  \hspace{1cm} (5.14)

where

$$STF(z) = \frac{V_o}{V_{in}}(z) = \frac{c_1c_2}{z^2 + (a_2c_2 - 2)z + (1 - a_2c_2 + a_1c_1c_2)},$$ \hspace{1cm} (5.15)

and

$$NTF(z) = \frac{V_o}{Q}(z) = \frac{(z - 1)^2}{z^2 + (a_2c_2 - 2)z + (1 - a_2c_2 + a_1c_1c_2)}.$$ \hspace{1cm} (5.16)

Note that the dc gain of the second-order modulator is

$$\frac{V_o(z)}{V_{in}(z)}(z \to 1) = \frac{1}{a_1}.$$ \hspace{1cm} (5.17)

The noise-shaping transfer function has two-zeros at dc, which suppress the quantization noise at low frequencies. The poles were matched to those of a Butterworth high-pass filter. Table 5.1 summarizes the final feedback gains and integrator gains for this modulator after scaling and stability verification with a block diagram simulator.

### 5.3.3 Third-order Modulator

For the third-order modulator shown in Figure 5.7, the transfer functions relating the input and quantization noise to the output are

$$STF(z) = \frac{V_o}{V_{in}}(z) = \frac{c_1c_2c_3}{z^3 + \alpha_2z^2 + \alpha_1z + \alpha_0},$$ \hspace{1cm} (5.18)
Table 5.1
Second- and third-order modulator coefficient values.

<table>
<thead>
<tr>
<th>Coefficient</th>
<th>Second-order Modulator</th>
<th>Third-order Modulator</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_1$</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>$a_2$</td>
<td>0.6</td>
<td>0.3113</td>
</tr>
<tr>
<td>$a_3$</td>
<td>0</td>
<td>0.2170</td>
</tr>
<tr>
<td>$c_1$</td>
<td>0.3</td>
<td>0.068</td>
</tr>
<tr>
<td>$c_2$</td>
<td>0.5</td>
<td>0.3727</td>
</tr>
<tr>
<td>$c_3$</td>
<td>0</td>
<td>0.5</td>
</tr>
</tbody>
</table>

\[
NTF(z) = \frac{V_o(z)}{Q(z)} = \frac{z^3 + (bc_1c_3 - 3)z^2 + (3 - bc_1c_3)z - 1}{z^3 + \alpha_2 z^2 + \alpha_1 z + \alpha_0},
\]

(5.19)

where

\[
\begin{align*}
\alpha_2 &= a_3c_3 + bc_2c_3 - 3 \\
\alpha_1 &= -2a_3c_3 - bc_2c_3 + a_2c_2c_3 + 3 \\
\alpha_0 &= a_3c_3 - a_2c_2c_3 + a_1c_1c_2c_3 - 1
\end{align*}
\]

(5.20)

Note that the dc gain of the third-order modulator is

\[
\frac{V_o(z)}{V_{in}(z)}(z \to 1) = \frac{1}{a_1}.
\]

(5.21)

The feedback coefficient $b$ allows for placement of a zero-pair at a finite frequency which may further suppresses quantization noise in the baseband. The numerator may be factored into the following expression:

\[
(z - 1)(z^2 + (bc_1c_3 - 4)z + 1),
\]

(5.22)

corresponding to a zero at dc and a complex zero pair. In the case where $b = 0$, all three zeros of the noise-shaping transfer function reside at $z = 1$ corresponding to dc. The selected coefficients for the third-order modulator are also summarized in Table 5.1.

5.3.4 Simulator

The design procedure outlined in the last section requires a behavioral simulator. The basic simulator algorithm is displayed in Figure 5.8. In this algorithm, $i_{out_i}$ are the
simulate()
{
    compute $M$ ADC and $M - 1$ DAC quantization levels;
    compute starting value for DAC output $y_{DAC}$;
    compute timestep $h$ based on sampling frequency;
    $t \leftarrow 0$;
    $k \leftarrow 1$;
    while ($t < \text{Simulation.Time}$) {
        compute sampled input value $x_k$;
        for $i \leftarrow N, N-1, \ldots 2$ {
            $iout_{ik} \leftarrow c_{ik} \times (iout_{i-1,k} - a_{ik} \times y_{DAC})$;
            check for saturated integrator outputs;
        }
        $iout_{ik} \leftarrow c_{1k} \times (x - a_{1k} \times y_{DAC})$;
        compute output $y \leftarrow \text{adc}(iout, adclevels, M)$;
        compute DAC output $y_{DAC} \leftarrow \text{dac}(y, daclevels, M)$;
        store histogram data;
        $t \leftarrow t + h$;
        $k \leftarrow k + 1$;
    }
}

N integrator outputs, $c_i$ are the integrator gains and $a_i$ are the feedback coefficients. Functions $\text{adc}$ and $\text{dac}$ represent generic functions to perform M-bit quantization of the ADC input and reconstruction of the DAC output signal from the digital output $y_{DAC}$. Nonidealities may be added to this simulator via the coefficient values and the ADC and DAC functions.
Chapter 6

Integrated Circuit Design

6.1 Operational Amplifier

The core circuit employed by the system is the operational amplifier, which is used to construct both the modulator and the sensor interface. The performance of the opamp affects the overall performance of both of these circuits. An opamp with large dc gain is desirable for minimizing the impact of parasitics in the front end of the sensing scheme, as well as for minimizing the integrator leakage in the ΣΔ modulators. The dc gain in the first integrator of the modulator must be at least as high as the oversampling ratio for first-order converters [58], although this constraint is somewhat relaxed for higher order converters. The system clock requirements dictate the speed of the opamp, since it must be able to slew and enter linear settling within half a clock cycle to prevent distortion in the modulator [78, 79].

![Diagram of operational amplifiers](image)

FIGURE 6.1: Regular and active cascode circuits.

Both a large open-loop gain and high unity-gain frequency are required in order to meet both accuracy and fast settling requirements of the system. The large oversampling ratio, on the order of $10^5$ for the small-sensor, requires an open-loop gain on the order of 100 dB. In addition, the system is designed for clock speeds...
below 4 MHz, so that in the worst case the opamp must settle in 125 ns. Satisfying the gain requirement is difficult in short-channel CMOS processes, since the intrinsic gain of the devices is limited. A technique for opamp open-loop gain-enhancement is therefore necessary, and is applied to a folded-cascode topology.

The active cascode circuit shown in Figure 6.1 provides a small-signal output resistance approximately \( A \) times larger than that of a regular cascode, also shown in the figure, where \( A \) is the open-loop gain of the feedback amplifier [80]. The amplifier is used to create a minor loop that fixes the voltage at the drain of the lower NMOS transistor \( M_1 \) such that changes in the output current produce very small changes in drain voltage of \( M_2 \).

The active cascode technique can be used to enhance the gain of an operational amplifier. Figure 6.2 gives an example of a fully-differential folded-cascode topology in which this technique is applied [81]. The differential input is applied to \( M_1 \) and \( M_2 \), and the resulting differential current is injected into the low impedance node formed at the source of the common-gate transistors \( M_5 \) and \( M_6 \). The differential output is taken from the high-impedance node of the cascode devices \( M_{3-6} \) and \( M_{7-10} \). To first order, the dc gain of the main circuit is proportional to the input stage transconductance \( g_{m1} \) times the output resistance of the cascodes. By substituting the regular cascodes, \( M_{3-6} \) and \( M_{7-10} \), with active cascodes, the open-loop dc gain of the main amplifier is improved by a factor equal to the dc gain of the added amplifiers. This occurs because the overall output resistance of the folded cascode is raised by the same factor [81] as indicated in Figure 6.1.

The disadvantage of this particular technique for gain-enhancement is that it requires implementation of four additional amplifiers to achieve the active cascode feedback. The added complexity increases the power consumption of the overall amplifier. Additionally, since the gain-enhancement amplifiers utilize current mirrors for single-ended conversion, the dynamic performance of the main amplifier is degraded due to ringing at the outputs of the gain-enhancement amplifiers. For example, differential ringing at the gates of transistors \( M_5 \) and \( M_6 \) due to various mismatches between the left and right halves of the circuit translates directly into an increased settling time at the output. These mismatches may be due to device mismatches and also due to current level differences caused by the differential signal.

The fully-differential gain-enhanced opamp shown in Figure 6.3 is an alternate topology for implementing the gain-enhancement technique of [81]. Since improvement of only the differential signal is important, the four gain-enhancement amplifiers can be replaced by two fully-differential ones. The advantage of the fully-differential gain-enhancement method is in the area and power savings. This technique has the ancillary benefit that internal node dynamics are improved, since current mirrors are eliminated from the additional amplifiers. Any mismatches cause only common-mode settling errors which are rejected by the high common-mode rejection of the fully-differential feedback amplifier. This yields a faster-settling transient response at the output.
Figure 6.2: Gain-enhanced opamp using single-ended amplifiers for cascode feedback.
Figure 6.3: Fully-differential gain-enhanced opamp using differential amplifiers for cascode feedback.
Figure 6.4: Additional gain-enhancement opamp with PMOS input stage.

One of the auxiliary gain-enhancement amplifiers is shown in detail in Figure 6.4. Both of the additional amplifiers use single transistor common-mode feedback for simplicity. Because the amplifiers have local unity-gain feedback in the main amplifier, setting the input common-mode level is equivalent to setting the output common-mode level. This common-mode control can be achieved utilizing only one additional device $M_{10}$, making the fully-differential gain-enhancement amplifier only slightly more complex than its single-ended counterpart. For the amplifier shown in Figure 6.4, the requisite gate voltage for $M_{10}$ is $V_{DS,SAT}$ of an NMOS device, $\Delta V_N$, so that as long as $M_{10}$ and $M_1$ have the same $V_{gs}$, the input common-mode level will also be $\Delta V_N$. The $\Delta V$ needed for each of these amplifiers is generated with little additional complexity, requiring no additional circuitry to generate $\Delta V_P$, the $V_{DS,SAT}$ of the PMOS device, and only two additional devices to generate $\Delta V_N$ from an improved cascode circuit [82]. High open-loop gain for these feedback amplifiers is achieved using non-minimum channel length devices as the output stage load. As an alternative, cascode current sources can be used.

The opamp utilizes PMOS input devices since they generate less $1/f$ noise than NMOS devices. The assumption here is that the devices $M_3$ and $M_4$, which also contribute $1/f$ noise at the input, are large enough so that the higher $1/f$ noise of these NMOS devices, coupled with the ratio of $g_m3$ to $g_{m1}$, makes it worthwhile to use PMOS devices at the input. In addition, an isolated n-well was provided in the process, allowing for isolated PMOS devices with the wells connected to the sources for better common mode rejection.
Common-mode feedback of the main amplifier is achieved with the differential pair formed by $M_{15}$ and $M_{16}$, which senses a difference between the measured common-mode level, $V_{cm}$ and the desired common-mode output level, $V_{cmref}$, and appropriately adjusts the input stage current to set the output common-mode level. The output common-mode level is sensed using the switched-capacitor circuit shown in Figure 6.5. The unity-gain frequency of the common-mode loop is designed to be a factor of four lower than the unity-gain frequency of the overall amplifier.

6.1.1 Opamp Dynamics

The dc gain of the main folded-cascode amplifier without gain-enhancement can be derived from the small-signal model shown in Figure 6.6 and is

$$\frac{V_o}{V_{in}} = -\frac{g_{m1} R_L r_{oc}(1 + g_{m5} r_{o5})}{r_{o5} + R_L + r_{oc}(1 + g_{m5} r_{o5})},$$  \hspace{1cm} (6.1)$$

where $r_{oc} = r_{o1}||r_{o3}$ is the combined output resistance of $M_1$ and $M_3$, and $R_L$ is the output resistance of the load current source.

The dynamics of this amplifier are predicted from the model in Figure 6.7 using open circuit time constants \cite{83}. The dominant time constant is created by the load capacitance at the output, $C_L$, and is equal to $C_L$ times the effective output impedance

$$\tau_1 = C_L \left[ R_L || (r_{o5} + r_{oc}(1 + g_{m5} r_{o5})) \right].$$  \hspace{1cm} (6.2)$$
Figure 6.6: Small-signal model for the folded-cascode amplifier.

Figure 6.7: Folded-cascode amplifier with significant capacitors noted.
The unity-gain frequency can be calculated from Equations 6.1 and 6.2

\[
\omega_c = \left( \frac{V_o}{V_{in}} \right) \frac{1}{\tau_1} = \frac{g_{m1}r_{oc}(1 + g_{m5}r_{o5})}{C_L\left[r_{o5} + r_{oc}(1 + g_{m5}r_{o5})\right]}. \tag{6.3}
\]

For \(g_{m5}r_{oc} >> 1\), the unity-gain frequency is dominated by the first stage transconductance divided by the load capacitance, \(\omega_c \approx g_{m1}/C_L\). The second pole for this amplifier is created by the total capacitance at the drain node of the input devices, which is dominated by \(C_{gs5}\). The time constant associated with this pole is

\[
\tau_2 = \frac{C_{gs5}r_{o5}r_{oc}}{r_{o5} + r_{oc}(1 + g_{m5}r_{o5})}. \tag{6.4}
\]

The second pole is roughly at a radian frequency \(\omega_2 \approx g_{m5}/C_{gs5}\), and contributes phase shift at the unity-gain frequency of the amplifier.

![Figure 6.8: Small-signal model for the gain-enhanced folded cascode amplifier.](image)

When the auxiliary amplifier is included to improve the cascaded structures, as indicated in the small-signal model in Figure 6.8, the dc gain of the amplifier is

\[
A_{dc} = \frac{V_o}{V_{in}} = -\frac{g_{m1}R_Lr_{oc}(1 + g_{m5}r_{o5}(a + 1))}{r_{o5} + R_L + r_{oc}(1 + g_{m5}r_{o5}(a + 1))}. \tag{6.5}
\]

The total gain can be approximated as

\[
\frac{V_o}{V_{in}} \approx -g_{m1}(R_L||r_{oc}(1 + g_{m5}r_{o5}(a + 1))). \tag{6.6}
\]

In this case, \(R_L\) is also increased by the gain of the additional amplifier, therefore the dc gain of the main amplifier is improved by the dc gain, \(A\), of the additional amplifier. This is shown schematically in Figure 6.9, which indicates the open-loop gain versus frequency, both with and without the additional amplifier. Since the output impedance of the amplifier is increased by \(A\), the first pole of the improved amplifier moves to a lower frequency \(\omega_1\), such that \(\tau_1 = \tau_1 A\). The gain-enhancement
Figure 6.9: Open-loop gain of the gain-enhanced folded-cascode amplifier.

Figure 6.10: Load impedance versus frequency for the gain-enhanced opamp.
will be effective so long as the bandwidth of the auxiliary amplifier \( \omega_2 \) lies above the first pole of the main amplifier, \( \omega_1 \). This is equivalent to saying that the crossover frequency of the additional amplifier \( \omega_4 \) must reside above the 3 dB frequency of the original amplifier. The additional amplifier Miller multiplies the capacitance between the gate and source of \( M_5 \), so that the effective capacitance seen at the drain node of the input devices is \( C_{gs2}(a + 1) \). However, the amplifier also decreases the impedance at that node by \( a \), so that the overall effect on the second pole of the main amplifier is minimal.

The unity-gain frequency and the dynamics of the main amplifier remain relatively unchanged by the addition of the auxiliary amplifier, however care must be taken in the design of the frequency response of the additional amplifier so that the settling time is not degraded. The gain-enhancement adds a doublet to the frequency response of the main amplifier. This doublet must be carefully placed to avoid deteriorating the settling time of the amplifier. Assume that the additional amplifier stage has the following single-pole dynamics

\[
a = \frac{A}{\tau s + 1}.
\]

The output impedance of the main amplifier, ignoring the dominant capacitance \( C_L \) at the output, rolls off so that at high frequencies the impedance is that of the original folded cascode with no gain-enhancement. In other words, the impedance of the cascode structure has the form

\[
Z_{OUT} = R_{OUT} \frac{\tau_z s + 1}{\tau s + 1},
\]

where

\[
\tau_z = \tau \frac{1 + g_{m5}r_{ds}}{1 + g_{m5}r_{ds}(A + 1)} \approx \frac{\tau}{A}.
\]

The time constant of the zero \( \tau_z \) corresponds to the crossover frequency of the gain-enhancement amplifier \( \omega_c = A/\tau \). When combined with the impedance of the load capacitance \( C_L \), the output impedance, and therefore the gain of the amplifier, has a pole-zero doublet that resides around the crossover frequency of the auxiliary amplifier. This effect on the load impedance is indicated in Figure 6.10. For unity gain settling, it is important that this doublet frequency be kept away from the crossover frequency of the main amplifier. (The doublet can be placed so that the magnitude of the open-loop gain at the doublet frequency is larger than \( 1/\beta \), the feedback factor of the closed-loop opamp configuration.)

The auxiliary amplifier forms a closed minor loop whose dynamics must be considered. The loop transmission of this minor loop is evident from Figure 6.11, in which the additional amplifier is modeled as a transconductance source \( g_{ma} \), a load resistance \( r_a \) and a compensation capacitor \( C_a \). The dominant pole of the auxiliary amplifier is formed by its output resistance and load capacitance, which includes an intentionally introduced compensation capacitance \( C_a \) plus the total capacitance at
the gate of $M_5$, dominated by $C_{gs5}$. The second pole will be that of the main amplifier, and it is therefore necessary to keep the crossover frequency of the auxiliary amplifier below the second pole of the main amplifier to stabilize the internal feedback loop.

### 6.1.2 Simulation Results

Figure 6.12 shows the simulated open-loop differential mode magnitude and phase of the gain-enhanced opamp. These HSPICE simulations use BSIM Level 13 models extracted from HP 1.2 µm n-well process data offered through the MOSIS service. The dc gain is enhanced by 60 dB, corresponding to the gain of the additional amplifiers. In practice, the gain is limited by stray conductance at the output node.

#### Table 6.1
Summary of operational amplifier simulation results.

<table>
<thead>
<tr>
<th></th>
<th>Gain Enhancement</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Differential</td>
<td>Single-Ended</td>
<td>None</td>
</tr>
<tr>
<td>Dc gain</td>
<td>133 dB</td>
<td>128 dB</td>
<td>73 dB</td>
</tr>
<tr>
<td>Unity gain frequency</td>
<td>100 MHz</td>
<td>100 MHz</td>
<td>100 MHz</td>
</tr>
<tr>
<td>Load Cap</td>
<td>10 pF</td>
<td>10 pF</td>
<td>10 pF</td>
</tr>
<tr>
<td>Phase margin</td>
<td>83°</td>
<td>82°</td>
<td>85°</td>
</tr>
<tr>
<td>Power</td>
<td>15 mW</td>
<td>17 mW</td>
<td>14 mW</td>
</tr>
<tr>
<td>Supply Voltage</td>
<td>±2.5 V</td>
<td>±2.5 V</td>
<td>±2.5 V</td>
</tr>
<tr>
<td>Settling time (1 ppm of final value)</td>
<td>93 ns</td>
<td>106 ns</td>
<td>128 ns</td>
</tr>
<tr>
<td>Active area (estimated)</td>
<td>16174 µm²</td>
<td>19896 µm²</td>
<td>13039 µm²</td>
</tr>
</tbody>
</table>
Figure 6.12: Gain-enhanced opamp open-loop differential-mode characteristic (a) gain and (b) phase.
Figure 6.13: Differential mode internal transient response at PMOS auxiliary amplifier input with (a) single-ended and (b) fully-differential gain-enhancement.
Transient simulations of the opamp in a unity-gain configuration demonstrate the improved dynamic performance of the amplifier. Figure 6.13 shows the differential transient response at the output of the gain-enhancement amplifier, corresponding to the differential voltage at the gates of the actively cascaded transistors, $M_7$ and $M_8$ in Figure 6.3. For this simulation, the gate lengths of the following device pairs are mismatched by 0.1 μm: $M_1$ and $M_2$, $M_5$ and $M_6$, and $M_7$ and $M_8$. The improved dynamic performance of the fully-differential circuit translates into a somewhat faster settling time, as indicated in the summary of simulation results in Table 6.1. For these simulations, the final value refers to the voltage that the output would settle to if an infinite amount of time is allowed.

### 6.2 Integrating/Summing Block

![Switched capacitor integrator/summer.](image)

**Figure 6.14:** Switched capacitor integrator/summer.

Figure 6.14 shows a simplified switch-level diagram of the basic building block used in the modulator. The circuit is shown single-ended for simplicity, although the actual implementation is fully-differential and also requires several more switches to accommodate a switched-reference scheme. This circuit implements both the summing and integrating functions required in the loop filter. Charge conservation at the virtual ground node of the operational amplifier yields

$$
(v_o(\phi_2) - v_o(\phi_1))C_1 = v_{in}(\phi_1)C_S - v_{fb}(\phi_2)C_F.
$$

(6.10)

This relation can be expressed as a difference equation. For a two-phase clock, we can define time increments at the rising edge of clock phase $\phi_2$. The time-step $h$ is defined as the total clock period including both phases, $\phi_1$ and $\phi_2$, so that Equation 6.10 can be rewritten as follows

$$
v_o(t) = v_o(t - h) + \frac{C_S}{C_i} v_{in}(t - h) - \frac{C_F}{C_I} v_{fb}(t).
$$

(6.11)
6.3 QUANTIZER

Assuming that the outputs change only at discrete-time intervals, this equation is equivalent to the difference equation

\[ v_o[n] = v_o[n-1] + \frac{C_S}{C_I} v_{in}[n-1] - \frac{C_F}{C_I} v_{fb}[n]. \]  

(6.12)

The z-transform of this equation produces the following transfer functions for the input signal \( v_{in} \) and the feedback signal \( v_{fb} \)

\[
\frac{V_o}{V_{in}}(z) = \frac{C_S}{C_I} \cdot \frac{z^{-1}}{1 - z^{-1}}, \text{ and} \\
\frac{V_o}{V_{fb}}(z) = \frac{C_S}{C_I} \cdot \frac{1}{1 - z^{-1}}. 
\]

(6.13)

These transfer functions are used to design the loop filter of the \( \Sigma\Delta \) modulator. In practice, finite opamp gain, finite switch resistance and finite opamp bandwidth can cause the transfer functions to deviate from these ideal equations [79].

6.3 Quantizer

![Clocked comparator circuit.](image)

\textbf{Figure 6.15:} Clocked comparator circuit.

The internal one-bit ADC, or quantizer, of the \( \Sigma\Delta \) is implemented using the bistable latch shown in Figure 6.15. The input to the quantizer is the output of the last integrator in the loop filter. On clock phase \( \phi_1 \), the input to the quantizer is sampled. On phase \( \phi_2 \) the circuit latches depending on the magnitude of the differential input compared with the offset of the latch. Because the modulator output
is only necessary to compute at discrete time intervals, the clocked latch provides a means of implementing a power-efficient comparator.

The latched output of the comparator, valid on phase $\phi_2$ of the clock, is used to drive on-chip buffers, providing digital output from the modulator. The buffers are implemented with cascaded scaled inverter circuits. The comparator outputs are also used to switch feedback references into the modulator summing nodes, thus implementing the internal one-bit DAC.

### 6.4 Modulator

Figures 6.16 and 6.17 show the switched-capacitor implementations of the modulator block diagrams of Figures 5.6 and 5.7, respectively. The circuit is implemented fully-differentially with the integrating and summing units shown in Figure 6.14 as the basic building block. A fully-differential architecture was implemented so that power supply and substrate noise can be rejected, and clock feedthrough errors can be minimized. The internal ADC is implemented using the fully-differential latch and operational amplifier circuits previously described.

The one-bit DAC is implemented using a switched reference [84]. The reference voltages, $V_{DAC1}$ and $V_{DAC2}$, are either subtracted or added to the integrator input at each of the summing nodes, depending on the polarity of the output from the latch and their connection to either the inverting or non-inverting signal path. Since the circuit is fully-differential, each DAC reference is connected to two sets of switches and capacitors.

This is explained in Figure 6.18(a), which shows the desired connections for a logical low latch output. On phase $\phi_1$, $V_{DAC}$ charges capacitor $C_{FP}$. The resulting charge is redistributed at node $A$ on clock $\phi_2$ when switch $S_1$ closes. This operation effectively adds the reference voltage $V_{DAC}$, multiplied by the capacitor ratio $C_{FP}/C_{iP}$, to the output of the integrator. At the same time $V_{DAC}$ also charges capacitor $C_{IN}$ and redistributes that charge at node $B$ using switch $S_2$, thus subtracting the feedback charge from the integrator output, as desired in the inverting signal path of a fully-differential circuit. In this manner a "positive" reference is connected to node $A$ and a "negative" reference is connected to node $B$ when the latch output is low. To implement the desired connections for a logical high latch output, we can instead use switch $S_1$ to connect to node $B$ and switch $S_2$ to connect to node $A$. This added functionality is accomplished with the circuit shown in Figure 6.18(b). In this circuit, the latch output is gated with clock phase $\phi_2$ to produce clocks $\phi_N$ and $\phi_P$, as indicated in Figure 6.16. These clocks control the summing node to which the "positive" and "negative" references are connected. The polarity of this connection obviously depends on the sign of the output voltage.

The switched-reference technique is preferred over direct feedback from the comparator in which those outputs are latched to the supplies. Although the latch outputs
Figure 6.16: Switch-level schematic of the second-order modulator.
Figure 6.17: Switch-level schematic of the third-order modulator.
Figure 6.18: Schematics showing switched DAC connections assuming (a) $\phi_P = \phi_2$, and (b) the general case.
can be buffered, it is preferable to provide a very low-noise reference voltage to the summing nodes. This is particularly critical in the first stage of the modulator where any noise on the feedback reference will alias into the baseband, causing idle tones in the spectrum of the modulator [79].

By implementing the DAC with a switched reference, only one reference voltage is needed since the polarity of the reference can be changed by introducing the charge on different clock phases (therefore implementing addition or subtraction of the reference). Alternate schemes require the use of an inverting buffer to create a negative reference or the use of two references.

An ancillary benefit of the switched reference scheme is that separate DAC reference voltages can be provided to each stage. In particular this is implemented to accommodate the mismatch in nominal capacitance per unit area of the capacitors in the first stage. Since this modulator is eventually combined with the sensor interface, it is impossible to implement known capacitor ratios.

The coefficients of the modulator are determined by the capacitor ratios. For the second-order modulator circuit shown in Figure 6.16, the first two block diagram coefficients of Figures 5.6 and 5.6 are

\[
\begin{align*}
c_1 &= \frac{C_{S1}}{C_{I1}} \\
c_2 &= \frac{C_{S2}}{C_{I2}} K \\
a_1 &= r \frac{C_{F1}}{C_{S2}} \\
a_2 &= r \frac{C_{F2}}{C_{S2}}
\end{align*}
\]

where K is the effective gain of the comparator and r the ratio of the magnitude of the feedback reference voltage \(V_{DAC}\) to \(V_{SS}\), the comparator reference. The capacitances are selected based on the slewing and settling requirements of the operational amplifier. The capacitance seen at the output of the opamp, which is the series combination of the integrating capacitor and sum of the feedback and input capacitor in parallel with the input capacitance of the next stage. This total capacitance should be on the order of 10 pF for the opamp unity gain frequency to be as designed. If the capacitance is significantly too low, the modulator will be unstable. Therefore where needed, additional capacitance is added. All capacitors are connected so that their top plates attach to the summing node of the opamp. This strategy is implemented for minimal substrate coupling, and so that the voltage sources drive the associated parasitic capacitance between the bottom plate and the substrate.

A two-phase clocking scheme is used to control the modulator. In general, the switches are implemented using transmission gates, although single NMOS switches
are used where possible. For example, switches connecting the top plate of the sampling capacitor between ground and the virtual ground of the opamp can be implemented with a single NMOS switch. Since the switch does not connect to the input, cancelling signal dependent charge injection is not an issue. Additionally, the signal swing of the switch is minimal. PMOS devices in the transmission gates are sized twice as wide as the NMOS devices to accommodate the factor of two mobility difference between them. To prevent signal-dependent charge injection a sliding clock scheme is used [85, 86]. Clock phase $\phi_{1d}$ is used to control the connection between the input and the sampling capacitor $C_S$. In this case clock $\phi_{1d}$ is derived on-chip from a four-inverter chain whose input is clock $\phi_1$.

### 6.5 Sensor Interface

Many sensing schemes measure the sensor capacitance with a capacitance to voltage converter, then use an ADC to generate a digital signal. A switched-capacitor gain stage provides a convenient interface to the switched-capacitor modulator in an ADC. This gain stage can be merged with the first stage of the $\Sigma\Delta$ modulator to eliminate the need for the operational amplifier, switches and capacitors required to implemented the interface circuit.

![Sensor interface circuit](image)

**Figure 6.19:** Sensor interface circuit.

A switch-level schematic of the combined sensor interface and modulator is shown in Figure 6.19. The gain of the modulator, which is determined by the ratio of the feedback capacitor and reference capacitor, is in this case

$$\frac{1}{a_1} = \frac{C_x}{C_R} = 1 + \frac{\Delta C}{C_R}. \quad (6.15)$$

The changing capacitor $C_x$ modulates the dc gain of the $\Sigma\Delta$. Although the changing capacitance of $C_x$ will modify the coefficients of the modulator, the anticipated...
full scale changes in $C_x$ are small enough that the resulting changes in loop-filter characteristics and loop stability can be tolerated.

The savings of this scheme are in the total power and area, since the number of opamps, capacitors and switches is minimized. In addition, this scheme also reduces the amount of noise introduced by switches and opamps. The disadvantage of this method is that the size of the sensor capacitance is fixed, so there is less flexibility in the choice of capacitors used to implement the loop-filter coefficients. In addition, the capacitors fabricated in the sensor layers have different capacitance per unit area than the capacitors used in the integrated circuit, which are poly-metal in the first stage. This will cause a wider processing spread in the expected values for the modulator coefficients.

With the sensor capacitances in place at the input to the modulator, the capacitor values used to implement the coefficients of the loop changed. In fact, the loop filter was redesigned for a more conservative nominal loop to ensure stability of the loop over a 25% change in sensor capacitance. Figure 6.20(a) and (b) show the location of the modulator poles over a 25% change in sensor capacitance for the second- and third-order modulator, respectively. The nominal position of the poles is indicated.

### 6.6 Sources of Error

Modulators can be very tolerant of component mismatch and nonidealities within the loop, particularly when the oversampling ratios are large. For the sensing system, both oversampling ratios are greater than 1000, so that the loop is relatively tolerant of mismatches and noise injected beyond the first stage. However, nonidealities in the first stage of the modulator can cause a significant deviation from the ideal behavior of the \( \Sigma \Delta \). The following section describes some of these nonidealities and how they were addressed in the modulator design.

#### 6.6.1 Capacitor Nonlinearity

The linearity of the capacitors in the first stage of the modulator is important. Figure 6.21 shows a diagram indicating the critical capacitors. If $C_S(V) = C_0(1 + \epsilon V')$ is nonlinear, then the sampled charge at node $A$ is

$$Q_A = V_{in} \times C_S = V_{in} C_0(1 + \epsilon V_{in}).$$

This is equivalent to a nonlinear voltage input of $V'_{in} = V_{in}(1 + \epsilon V_{in})$. Clearly this nonlinearity will produce harmonic distortion at the output of the modulator.

The linearity of $C_I$ is also important. The input voltage $V_{in}$ is coupled to the first-stage integrator output by the ratio of $C_S$ to $C_I$. The change in the output voltage of that stage is

$$\Delta V_{o1} = \frac{C_S V_{in} \pm C_F V_{DAC}}{C_I}.$$
Figure 6.20: Closed-loop pole location of (a) second- and (b) third-order modulator for 25% change in sensor capacitance.
Figure 6.21: Schematic indicating modulator first stage input summing node A.

If $C_I$ is nonlinear such that $C_I(V) = C_0(1 + \epsilon V)$, then

$$\frac{C_S V_{in}}{C_I} = \frac{C_S V_{in}}{C_0(1 + \epsilon V_{in})} \approx \frac{C_S V_{in}}{C_0}(1 - \epsilon V_{in}).$$ \hspace{1cm} (6.18)

Again, the nonlinearity in $C_I$ will appear like a nonlinearity in $V_{in}$, and add unwanted harmonic distortion to the output.

The linearity of $C_F$ is not significant in a single-bit architecture, since the feedback charge is discrete, and $Q_A$ at the summing node is described by

$$Q_A = \begin{cases} \frac{C_F(V_{DAC}) * V_{DAC}}{C_F(-V_{DAC}) * (-V_{DAC})} = Q^+ \\ \frac{C_F(V_{DAC}) * V_{DAC}}{C_F(-V_{DAC}) * (-V_{DAC})} = Q^- \end{cases}$$ \hspace{1cm} (6.19)

A nonlinearity in this capacitance is equivalent to a $V_{DAC}$ with a fixed error. This will only affect the loop in that the loop coefficients will change depending on the sign of the feedback, but will not cause harmonic distortion in the output. The feedback capacitor is still implemented with the same physical structure as the input capacitor so that relative matching is optimized.

Capacitor nonlinearity in the second and subsequent stages is not as significant due to the large in-band gain of the integrators which precede them. In the original design of the circuit, all of the first stage capacitors were implemented using poly-metal capacitors, which has the lowest voltage coefficient, but lowest capacitance per unit area, of the available capacitors. The remaining capacitors were implemented with poly-n+ structures. In the sensor-embedded design, the first stage capacitances were implemented with the sensors and the integrating capacitor with a poly-metal structure.

6.6.2 Comparator Offset

The $\Sigma\Delta$ modulator is able to tolerate a significant amount of comparator offset. The feedback loop causes the statistical distribution of samples at the input to the com-
parator to adjust so that they are centered around the comparator offset. Therefore the comparator is able to make the correct decisions for the given samples. The difficulty with this arises for low-swing amplifiers, where the comparator offset may be large enough to cause the distribution of inputs to shift too far, so that the operational amplifier output goes out of range. This will cause the integrator output to saturate resulting in harmonic distortion.

### 6.6.3 Component Matching

Capacitor mismatch in the modulator will affect both the stability and gain of the loop. Several types of capacitor mismatch are possible. For example, variation in the nominal values of capacitors $C_S$ and $C_F$ will yield a gain error in the modulator since it is determined by their ratio $a_1 = C_F/C_S$. Variations in the capacitances of $C_S$ and $C_I$, which cause the first integrator gain to vary, will only affect the dynamics of the loop since only the scaling of the first integrator output is modified. Mismatches in $C_S$ and $C_F$ between the two sides of the fully differential circuit may also occur. If the relative error is $\epsilon$ in $a_1$, the common-mode feedback causes the error to look like an effective error of $\epsilon/2$ in a balanced circuit. A similar result occurs for mismatches in $V_{DAC}$ where relative mismatch of $\epsilon$ results in an equivalent error of $\epsilon/2$ in the coefficient $a_1$. These effects are weakened for later stages of the modulator.

Because the circuit is implemented with sensors in the first stage as $C_S$ and $C_F$, it is impossible to create a reliable ratio with on-chip capacitors $C_I$. The adjustable DAC reference feedback to the first stage compensates this. However, capacitors forming the two sides of the differential circuit architecture are created using common centroid configurations to provide the best possible matching. A possible technique to create reliable ratios is to build capacitors using unit capacitors. However, this was not implemented since the ratios demanded by the modulator are not amenable to ratioing and because the matching in later stages is not as critical.

### 6.6.4 Finite Operational Amplifier Gain

Finite operational amplifier gain will decrease the dc gain of the integrators in the loop filter. The transfer function for the gain can be rewritten in the following form

$$\frac{V_o(z)}{V_{in}} = C_S \frac{\beta z^{-1} - 1}{1 - \alpha z^{-1}} \cdot \beta z^{-1}. \quad (6.20)$$

The dc gain of the integrator is degraded to

$$\frac{V_o}{V_{in}} (z \to 1) = \frac{C_S}{C_I} \frac{\beta}{1 - \alpha} \text{ for } \alpha < 1. \quad (6.21)$$

It can be shown that the leakage due to finite opamp gain will cause less dc gain in the loop and therefore less attenuation of the quantization noise at low frequencies.
A dc gain equal to the oversampling ratio of the modulator will cause less than 1 dB degradation in the noise at low frequencies [63, 58]. Large dc gain is also necessary to suppress harmonic distortion [79].

Because the oversampling ratios in the sensor circuit may be large due to the low-frequency calibration of the sensor, the operational amplifier gain must be larger than possible with a conventional CMOS amplifier. For this reason, a gain enhanced amplifier was used in the first stage of the modulator. However, the second and subsequent stages of the modulator do not require this gain, and simple folded cascode amplifiers were used for these stages, therefore conserving power and area.

6.6.5 Operational Amplifier Settling and Slew Rate

During one clock phase of the modulator, it is possible that the operational amplifier may have to switch over its entire output range. The opamp will most likely slew before entering linear settling. This nonlinear settling will result in significant harmonic distortion. However, if the amplifier enters linear settling before the end of a clock cycle, incomplete settling will result in a gain error only. In [79], it is suggested that the opamp be designed to fully settle in one clock cycle to avoid the resulting errors. For these reasons, the opamp was designed to slew for approximately 20% of the clock cycle, and the remaining time is used for linear settling. The slew rate is determined by the bias current of the folded cascode and the load capacitance.

6.6.6 Operational Amplifier Offset

The offset of the operational amplifier will have its most pronounced effect in the first stage of the modulator. Ideally, if $C_s$ and $C_F$ are equal (typically they are), then the offset voltage will be double-sampled on the two clock phases and will cancel. However in general, the change in the integrator output will reflect the offset voltage as follows

$$\Delta V_{o1} = \frac{C_s}{C_I} \left( V_{in} \pm \frac{C_R}{C_s} V_{DAC} + V_{off} \left( \frac{C_R - C_s}{C_s a_1} \right) \right). \quad (6.22)$$

The effect of the opamp offset voltage is related to the difference this gain has from unity. When the sensor capacitors are used at the input to the modulator, and used to modify the dc gain of the modulator, this effect will be more pronounced. However, correlated double sampling or chopper stabilization techniques can be used to alleviate this effect.
Figure 6.22: Schematic indicating leakage current in modulator first stage.

6.6.7 Leakage Current

Gate leakage current at sensitive nodes may be an issue for the modulator when operating at low speeds. Given a leakage current $i_L$ at the summing junction in Figure 6.22, the charge conservation equation for one clock cycle is

$$V_{in}C_S \pm V_{DAC}C_F = C_I \Delta V_{o1} + i_L \Delta T.$$  \hfill (6.23)

Dividing through by $C_S$ yields

$$V_{in} \pm a_1 V_{DAC} = c_1 \Delta V_{o1} + \frac{i_L \Delta T}{C_S}.$$  \hfill (6.24)

The leakage current integration appears like an effective input voltage change of $i_L \Delta T / C_S$. Assuming a clock frequency of 1 MHz and an input capacitance of 0.25 pF, the leakage current necessary to cause a 100 mV effective input shift is 25 nA. Even for a clock rate as low as 100 kHz, the leakage current can be as large as 2.5 nA before significantly affecting the output of the first stage.

6.6.8 Parasitic Capacitance of the Sensor

Large parasitic capacitances associated with the sensor will affect the modulator. Capacitances on the top plate of the sensor will increase the parasitic capacitance at the input node of the first operational amplifier. This capacitance will affect the feedback ratio of the closed-loop operational amplifier, and will decrease the bandwidth of the opamp and therefore its settling time. If significant parasitic capacitance is present, the modulator may not function at the desired speeds. Large parasitic capacitance may thus demand the need for larger open-loop gain in the opamp. Capacitances on the bottom plate will increase the dynamic load on the reference sources, but will not otherwise affect the performance of the modulator.
6.6.9 Temperature Coefficient Mismatch

The most pronounced temperature effects will be in the temperature coefficient mismatch between the capacitors in the first stage. In this implementation, the first stage input and feedback capacitors are formed by the sensor air gap, whereas the integrating capacitor is a poly-metal capacitor. The temperature coefficient mismatch of these capacitors will cause a further statistical spread in the coefficients of the modulator. This temperature effect on the gain of the modulator can be calibrated by the adaptive calibration, however the loop stability can not be guaranteed. The modulator must be designed with enough margin to ensure stability. However because the temperature coefficient of the sensors was not known at the time, this was not addressed.
Chapter 7

Test Setup

Several integrated circuits were built and tested. The first one is an integrated circuit specifically designed for standalone testing. This chip contained standard capacitors in place of sensors. A second chip, in which the electronics are identical to that of the first, was also built. This chip is designed to be packaged with external connections to sensors. Both were built in a standard IC process. A third integrated circuit was fabricated at MIT and includes both sensors and signal-processing circuits on the same substrate. Each of these three chips will be described, followed by a discussion of the test setups and circuit used to test the performance of each of them.

7.1 Integrated Circuit Layout and Fabrication

The integrated circuit was originally designed for a 1.2 μm digital CMOS process offered by the MOSIS service. The integrated circuit was ultimately fabricated in a 0.6 μm TSMC process through Analog Devices. Only critical devices were scaled and in general the 1.2 μm design rules were maintained. The TSMC process is a two-level metal process with two layers of polysilicon.

The layout of the integrated circuit is hierarchical so that consistency and matching can be maintained between the blocks of the modulators. Because large fractions of the design are fully-differential, symmetric layout techniques were used so that only one half of the circuit layout needed to be created manually. The primary concern in the integrated circuit is to keep clocks and other digital signals as far away from the analog signal lines as possible. In particular, care was taken to keep the opamp input summing nodes shielded from these digital signals.

The core of the modulators is formed by the discrete-time summing and integrating block shown schematically in Figure 7.1. In this case the digital power supplies and clock lines form a bus at the outer extreme of the layout. The transmission gates and pass gates form the connection between the clocks and the analog signals, and thus lie between these two sets of signals. The analog signals, which include the
Figure 7.1: Hybrid sensor integrator layout.
Figure 7.2: Hybrid sensor chip layout.
opamp summing nodes and outputs, are each shielded by an analog ground signal running between them. The opamp forms the core of the integrator block. The first stage integrator includes the layout of the additional amplifiers, and compensation capacitors for those amplifiers must also be included. Clocks for the opamp common mode feedback and an additional power supply bus complete the layout. The opamp clocks are not of particular concern, since that clock frequency is at least an order of magnitude lower than that of the main clock. Since they are synchronized, these clocks should not be switching when the amplifiers are settling.

The original process for which this circuit was designed did not contain two layers of polysilicon, therefore first-layer metal to polysilicon capacitors were used for the first stage where capacitor linearity is critical. Polysilicon to n+ capacitors were used in the remainder of the layout, for both signal capacitors, compensation capacitors and decoupling capacitors, since their capacitance per unit area is much larger than that of the metal to polysilicon capacitors. All signal capacitors were formed in a common-centroid configuration, to improve matching despite process variations in the oxide thickness. Capacitors connecting to the summing nodes were designed so that the top plates connect to the summing node, reducing the effects of any substrate coupling from the bottom plate. The top plates were shielded using a grounded upper-metal layer.

The overall layout of the chip is shown in Figure 7.2. The integrator/summer blocks are cascaded together to form the modulators, so that the main axis of symmetry in the opamp core is maintained. Large power-supply and signal buses run in the East-West direction, and are generally routed in the second layer of metal, which has a lower series resistance and lower capacitance to the substrate than the first-level metal. The pads which ultimately connect to the sensors are found on the West side of the chip, and interface to the switches with as short as possible lead lengths to reduce the parasitic capacitance to the substrate.

The integrated circuit uses 48 pads. These pads are concentrated in a U-shape, with the intention that they connect to three sides of a 64-pin pin grid array (PGA) package with the ease in bonding. The fourth side, consisting of 16 pads, is devoted to connections with the sensor elements. The large pin count makes many signals accessible. For testing purposes, the pads are created without ESD protection for complete flexibility in the applied voltages.

Since the integrated circuit is pad-limited, there is a large amount of unused area. The remaining area is used for decoupling capacitors between the power supplies when convenient, as well as decoupling capacitors from the bias lines to the appropriate voltage supplies to reduce the impedance of these nodes. The overall area of the chip, including the pad frame, is approximately 3800 μm square. A die photo of the integrated circuit is shown in Figure 7.3.
Figure 7.3: Photograph of integrated circuit.
7.2 Fully-Integrated Sensor Chip Layout

The fully-integrated sensor chip is designed for a 2 \( \mu \text{m} \) MIT twin-well CMOS process. The CMOS processing steps form the integrated-circuit core of the MEMS-CMOS process designed at MIT to integrate sensors and CMOS on the same substrate. The baseline CMOS process has one level of metal, and one polysilicon layer. Significant modifications were made to the original 1.2 \( \mu \text{m} \) design to accommodate the new design rules and layers. Polysilicon jumpers were used in lieu of a second-level of metal, and the design was rerouted almost entirely in the first-level metal. Great care was taken to minimize the amount of polysilicon used.

Figure 7.4 shows the integrator/summer layout for the MTL process. Because the switches in the new process are significantly larger, they became the limiting factor in the spacing of the integrator blocks. As such, they were relocated. Other rearranging was necessary to most conveniently handle the reduction to one level of metal, so that although the pad frame for this chip is identical to that of the previous chip, the layout is somewhat different. Figure 7.5 shows an illustration of the layout of the entire chip.

Again, the integrated circuit uses 48 pads, concentrated in a u-shape, and is compatible with the hybrid sensor chip. The 16 remaining pads, which previously connected with the sensor, are unused since the sensors reside on the chip. The area of the circuits and their corresponding bond pads is about 5500 \( \mu \text{m} \) by 9800 \( \mu \text{m} \). Approximately 15\% of that area is completely unused because the chip is pad-limited. The sensors consume an area that is 1100 \( \mu \text{m} \) by 8500 \( \mu \text{m} \), or about 15\% of the total area.

Both the hybrid chip and the integrated chip were verified using layout versus schematic tools in Cadence. Design rules were also verified using the Cadence design rule checker Diva.

7.3 Hybrid System Packaging

When the integrated circuit and the sensors reside on separate substrates, great care must be taken in packaging them together. Bond wires between the sensors and the IC, as well as from the IC to the package, must be kept to a minimum to reduce inductive and capacitive coupling.

One possible packaging solution is to place the sensor and the IC in the same package cavity, bond between the joint 16 pads on both die, and bond the circuit out to the lead frame on the package. Because of the relatively small size of the IC compared with the discrete sensor chip, this solution has several difficulties. First, the IC is not able to be centered in the cavity. Therefore, extremely long bond wires are necessary to bond to the upper leftmost portion of the PGA lead frame. This is illustrated in the diagram shown in Figure 7.6. In addition to the obvious concerns
Figure 7.4: Integrated sensor integrator layout.
Figure 7.5: Integrated sensor chip layout.
Figure 7.6: Hybrid sensor package.
Figure 7.7: Hybrid sensor package with vertical chip stacking.
about wire inductance of the long wires, the wires are difficult to create. They must reach over the sensor chip, which is thicker, and some of the wires are very close together. For a more reliable solution, the long wires should be rested on epoxy posts to keep them from sagging.

The sensor chip is formed with a glass overlayer. This glass layer forms a flat surface on which the integrated circuit can be mounted. Now the IC can be optimally placed within the package cavity so that the bond wires are all of more consistent length[87]. The sensor chip is rotated by 180 degrees, which was possible because the pad arrangement on the sensor chip was mirror-symmetric. With the IC mounted on top of the sensor chip, the bonds from the IC to the sensor chip are similar to downbonds from the IC to the level of the pads on the sensor substrate. This vertical distance is relatively significant since the wires must traverse two layers: from IC to glass and from glass to substrate. A schematic view of the packaging scheme is shown in Figure 7.7.

Pressure inlet ports for the sensor are drilled into the PGA before the sensor chip is glued to the package cavity. The holes are aligned to back-side pressure inlets on the sensor substrate. After the sensor substrate is secured, four thin plastic tubes are glued to the back of the package. These small tubes are packaged together in one large tube, which connects to the pressure test setup.

7.4 Test Setup

The main purpose of the test setup is to provide a flexible testing environment in which the integrated circuit can be evaluated both with and without the sensors. The test setup includes the printed circuit board (PCB) containing current sources, references, and clocks for the integrated circuit, the data acquisition system, and the pressure testing setup. The overall test system is depicted in Figure 7.8.
7.4.1 PCB Design

The integrated circuit is mounted in a (PCB). The board is composed of four layers, including one internal ground plane. Signals are routed on the component and solder side of the board, and power is routed on a separate plane. The layout of the board closely matches the physical location of signals on the PCB, and is intended to separate digital clock signals from analog supplies, reference currents and voltage sources.

A ground plane provides a wide area, and thus a low-impedance path, for return currents. The ground plane is notched so that current is guided into the appropriate areas of the board. Split power supply pairs are routed adjacent to each other so that return currents to the supplies flow in close proximity to the corresponding outgoing source.

The integrated circuit is mounted in a low-insertion force (LIF) socket. PGA pins are bypassed as close as possible to the chip for the lowest-possible impedance. Both 10 μF tantalum capacitors for low-frequency bypassing and 0.1 μF ceramic capacitors for high-frequency bypassing are utilized. Reference voltages and current sources are also appropriately bypassed close to the chip so that a low-impedance may be maintained over a wide frequency range.

Four reference voltages are provided on the PCB. Two of the references provide the DAC references for the modulators, and others provide voltages which can be used as dc inputs to the modulators. In either case, the reference voltage is critical to the ultimate accuracy of the system. The references are provided by a bandgap reference (AD780) and buffered with a low-noise opamp, as shown in Figure 7.9. Potentiometers are used so that the references can be adjusted anywhere from 0-3 V.

The ΣΔ modulators each have one main external bias current source. Current sources for each of the modulators are provided by n-p-n transistors. The bias circuit is shown in Figure 7.10. The nominal bias current for each modulator is 0.5 mA.

Two non-overlapping clocks are provided from a dual pulse generator, and terminated on board with 50 Ω resistors. These clocks provide the φ1 and φ2 clocks for the modulators. The modulator clocks are divided to provide the opamp common-mode refresh clocks. The division factor is programmable. The digital division circuitry is
performed in gate array logic.

Sinewave inputs to the modulator are provided from a low-distortion sinewave source. The inputs to the modulator are filtered immediately where the signal arrives on the board using a low-pass RC filter. The outputs from the modulators are buffered on-chip, and connect to a ribbon cable at the edge of the board for further processing. The modulator outputs connect to capacitive isolators, which are used to level shift from split supplies, nominally $\pm2.5$ V to TTL levels, 0-5 V.

Separate analog and digital power supplies are employed by the sensing chip. This is to minimize the interference on the analog supply from transients on the digital supply and vice versa. Additionally, this scheme provides the flexibility that the analog and digital supplies may be set to different levels. Power is provided to the chip via banana jack connectors on the board, and connect to external power supplies via twisted-pair wires.

### 7.4.2 Data Acquisition System

The data-acquisition system is designed to simultaneously store data from both the on-chip modulators. The data is stored in eight 128k by 8 bit SRAM chips. At 1 MHz this corresponds to 0.5 seconds worth of data. The data acquisition board contains the memory and corresponding address registers, as well as handshaking logic for communicating with a personal computer, where the data is uploaded at a much slower rate that it is loaded into memory. Final signal processing takes places in the computer.

### 7.4.3 Pressure Testing System

The test setup for applying pressure to the sensors is shown in Figure 7.11. The pressure regulator is a Bellofram type 1000 pressure transducer, with an output of
3-120 psi for a corresponding dc input of 0-10 V. The pressure sensor is a Honeywell high pressure gage sensor, model #242PC60G with a pressure range of 0-60 psi. The maximum pressure possible with in-house plumbing is about 45 psi.

7.4.4 Integrated Circuit Standalone Testing System

The adaptive linearity calibration can be demonstrated using the test chip with on-chip capacitors in place of sensors. A nonlinearity can be electrically introduced at the input of the third-order modulator to emulate the nonlinear sensor. Since the charge at the first summing node is equal to $C \times V$, nonlinearity introduced either through $C$ or $V$ can be calibrated. The circuit which generates the nonlinearity is adjustable so that different amounts of nonlinearity can be tested. A schematic of this circuit is shown in Figure 7.12. The nonlinearity results from the exponential nonlinearity of a diode-connected transistor $Q_1$. Applying voltage $V_{in}$ across $Q_1$ results in a collector current $I \propto e^{V_{in}}$ that is mirrored through $Q_3$ and $Q_4$ to produce a voltage $V_{out} = V_B + R_4I$. Resistor $R_1$ is used to weaken the nonlinearity. The test setup for standalone testing is shown schematically in Figure 7.13.
Figure 7.12: Circuit for generating electrical nonlinearity.
Figure 7.13: Test setup for electrical calibration test.
Chapter 8

Experimental Results

The experimental results are based on testing the integrated circuits both with and without sensors. Two versions of the basic integrated circuit were fabricated, one for use in the hybrid sensor and one containing on-chip capacitors to replace the sensors. The latter of the two was intended for integrated circuit testing only. The functionality of the integrated circuits are demonstrated using the stand-alone chip. The calibration is also demonstrated using this chip. A nonlinearity was electrically generated to mock the nonlinearity of the sensor, and the calibration of this nonlinearity is shown. The hybrid system, including sensors, was dc tested for basic functionality of the sensors and the circuit, and idle-channel measurements are shown.

In all cases, 1 MByte of data is collected by the data acquisition system. This data is uploaded into a workstation and data windowing and FFT tests, as well as calibration, are performed in combined c-code and MATLAB software. The following sections outline the experimental results.

8.1 Integrated Circuit Functionality

The basic functionality of the integrated circuits was determined by operating the modulators as stand-alone ADCs. For the purposes of demonstration, a test chip is used in which sensors are replaced by on-chip capacitors. These on-chip capacitors are of the same nominal capacitance as the sensors, therefore the loop coefficients and resulting characteristics of the modulators should be approximated under these conditions. The nominal size of the large sensor, $C_L$, is 1.6 pF and the nominal size of the small sensor, $C_S$, is 0.48 pF. The feedback references $V_{DAC1}$ and $V_{DAC2}$ are both nominally 1.2 V and 1.0 V, respectively.

The inputs to the modulator are driven with the low-distortion Audio Precision System One signal generator. A Blackman window is used to window the output bitstream from each modulator, and a 65536-point FFT was performed on the resulting data. Both the wideband and baseband characteristics of the modulator are
examined. The data shown is these experiments is taken a clock rate of 1 MHz.

When the inputs to the modulator are grounded, the spectrum of the output signal represents the idle channel noise. The spectrum of the third-order modulator idle-channel output is shown in Figure 8.1, from \( f = 0 \) to \( f = f_s / 2 \), half the sampling frequency. These results indicate noise-shaping functionality of the modulator. The high-frequency spikes indicate tones which are related to the dc input value. Figure 8.2 shows the low-frequency portion of the spectrum, taken from the same data, indicate the lower noise floor in the baseband. Both the wideband and low-frequency idle-channel noise for the second-order modulator are shown in Figures 8.3 and 8.4, respectively.

The wideband and narrowband characteristics of the third-order modulator are shown in Figures 8.5 and 8.6, for a 998 Hz sine-wave input. This data has not been decimated. For this test, the input sine-wave was 500 mV peak-to-peak. The DAC feedback voltage was 1.2 V for the first integrator and 1.0 V for the second and subsequent stages.
8.1. INTEGRATED CIRCUIT FUNCTIONALITY

**Figure 8.1:** Third-order modulator idle-channel noise for $f_s$ of 1 MHz, shown up to $f_s/2$.

**Figure 8.2:** Third-order modulator idle-channel noise for $f_s$ of 1 MHz, shown up to $f_s/100$. 
Figure 8.3: Second-order modulator idle-channel noise for $f_s$ of 1 MHz, shown up to $f_s/2$.

Figure 8.4: Second-order modulator idle-channel noise for $f_s$ of 1 MHz, shown up to $f_s/100$. 
Figure 8.5: Third-order modulator output spectrum for $f_s$ of 1 MHz, with sine-wave input at 988 Hz, shown from $f = 0$ to $f = f_s/2$.

Figure 8.6: Third-order modulator output spectrum for $f_s$ of 1 MHz, with sine-wave input at 988 Hz, shown from $f = 0$ to $f = f_s/100$. 
8.2 Calibration Test

The adaptive linearity calibration can also be demonstrated using the test chip with on-chip capacitors in place of sensors. An intentionally large nonlinearity is electrically introduced at the input of the third-order modulator to emulate the nonlinear sensor. Since the charge at the first summing node is equal to $C \times V$, nonlinearity introduced either through $C$ or $V$ can be calibrated.

8.2.1 Example One

Figure 8.7 shows the nonlinearity introduced to the third-order modulator. The maximum INL of this characteristic is 16%, commensurate with the nonlinearity of the large sensor. Figure 8.8 shows the corresponding output spectrum from the third-order modulator. As expected, the nonlinearity results in a large amount of harmonic distortion, indicated by the harmonic peaks in the spectrum. The increased noise in the baseband is introduced by the nonlinearity-generating circuit.

Figure 8.9 shows the spectrum of the output from the second-order modulator which is used as the calibration standard. Figure 8.10 shows the third-order modulator output spectrum after calibration. This is the main output of the system. The result indicates successful calibration.

For this particular example, the calibration utilizes seven calibration weights. Figure 8.11 shows the error in the first calibration weight $w_1$ versus iteration number, indicating the convergence properties of the calibration algorithm. Table 8.1 lists the final calibration weights after 1024 iterations. These weights correspond to data which has been scaled by approximately 2 to yield data on a [-1,1] interval, which is optimal when using Chebyshev polynomials. Because orthogonal polynomials are used, the weights other than the dominant term $w_1$ are of the same order of magnitude. Beyond $N = 7$, the weights become insignificant, as indicated by the higher-order coefficients shown for models with $N = 8$ and $N = 9$. For comparison, the weights corresponding to the calibration of a 10% nonlinearity are also shown in Table 8.2.

All data shown in Figures 8.7 through 8.11 are obtained with a 1 MHz clock rate and a 1 kHz, 500 mV amplitude, single-ended sinewave at the input to the second-order modulator. The same sinewave input passes through a nonlinearity generator before being applied to the third-order modulator. Although a sinewave was used as the input signal for convenience in testing, the calibration is effective with any input signal with enough energy below $f_2/2$. 
### Table 8.1
Calibration weights to correct distortion (maximum INL) of 16%.

<table>
<thead>
<tr>
<th>Weight</th>
<th>N=7</th>
<th>N=8</th>
<th>N=9</th>
</tr>
</thead>
<tbody>
<tr>
<td>$w_0$</td>
<td>0.1670</td>
<td>0.1672</td>
<td>0.1671</td>
</tr>
<tr>
<td>$w_1$</td>
<td>1.0009</td>
<td>1.0014</td>
<td>1.0013</td>
</tr>
<tr>
<td>$w_2$</td>
<td>-0.1638</td>
<td>-0.1636</td>
<td>-0.1638</td>
</tr>
<tr>
<td>$w_3$</td>
<td>-0.0017</td>
<td>-0.0012</td>
<td>-0.0013</td>
</tr>
<tr>
<td>$w_4$</td>
<td>-0.0039</td>
<td>-0.0036</td>
<td>-0.0038</td>
</tr>
<tr>
<td>$w_5$</td>
<td>0.0028</td>
<td>0.0033</td>
<td>0.0032</td>
</tr>
<tr>
<td>$w_6$</td>
<td>-0.0019</td>
<td>-0.0017</td>
<td>-0.0018</td>
</tr>
<tr>
<td>$w_7$</td>
<td>0.0019</td>
<td>0.0019</td>
<td>0.0019</td>
</tr>
<tr>
<td>$w_8$</td>
<td></td>
<td></td>
<td>-0.0007</td>
</tr>
</tbody>
</table>

### Table 8.2
Calibration weights to correct distortion (maximum INL) of 10%.

<table>
<thead>
<tr>
<th>Weight</th>
<th>N=7</th>
<th>N=8</th>
<th>N=9</th>
</tr>
</thead>
<tbody>
<tr>
<td>$w_0$</td>
<td>0.0814</td>
<td>0.0814</td>
<td>0.0815</td>
</tr>
<tr>
<td>$w_1$</td>
<td>1.0290</td>
<td>1.0289</td>
<td>1.0287</td>
</tr>
<tr>
<td>$w_2$</td>
<td>-0.0987</td>
<td>-0.0987</td>
<td>-0.0984</td>
</tr>
<tr>
<td>$w_3$</td>
<td>-0.0211</td>
<td>-0.0212</td>
<td>-0.0214</td>
</tr>
<tr>
<td>$w_4$</td>
<td>0.0053</td>
<td>0.0054</td>
<td>0.0056</td>
</tr>
<tr>
<td>$w_5$</td>
<td>-0.0012</td>
<td>-0.0013</td>
<td>-0.0015</td>
</tr>
<tr>
<td>$w_6$</td>
<td>0.0027</td>
<td>0.0028</td>
<td>0.0030</td>
</tr>
<tr>
<td>$w_7$</td>
<td>0.0006</td>
<td>-0.0008</td>
<td>0.0013</td>
</tr>
<tr>
<td>$w_8$</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure 8.7: Measured transfer function of electrical nonlinearity with linear input indicated.

Figure 8.8: Measured, uncalibrated spectrum of third-order modulator output.
Figure 8.9: Measured, uncalibrated spectrum of second-order modulator output, serving as calibration standard.

Figure 8.10: Calibrated spectrum of third-order modulator.
Figure 8.11: Convergence of weight $w_1$ indicated by error from its final value versus iteration number.

8.2.2 Example Two

A second example is presented in which the second-order modulator output has residual harmonic distortion. Figure 8.14 shows the spectrum of the output from the second-order modulator, which is used as the calibration standard. The source of the harmonic distortion in this output was caused by the test setup, but is used to demonstrate the limitations of the calibration. Figure 8.15 shows the third-order modulator output spectrum after calibration. This is the main output of the system. The result indicates successful calibration to within the linearity of the calibration standard.

All data shown in Figures 8.12 through 8.15 are obtained with a 1 MHz clock rate and a 911 Hz, 500 mV amplitude, single-ended sinewave at the input to the second-order modulator. The same sinewave input passes through a nonlinearity generator before being applied to the third-order modulator.
Figure 8.12: Measured transfer function of electrical nonlinearity.

Figure 8.13: Measured, uncalibrated spectrum of third-order modulator output.
Figure 8.14: Measured spectrum of second-order modulator output, serving as calibration standard, with residual distortion.

Figure 8.15: Calibrated spectrum of third-order modulator for case where calibration standard has residual distortion.
8.3 Idle Channel Tests: Hybrid Sensor

Noise-shaping functionality of the hybrid system, which uses sensors in the modulator first stage, is shown in Figures 8.16 and 8.17. These spectra show the output of the third- and second-order modulators with the inputs grounded. This data is taken at a clock rate of 1 MHz.

8.4 DC Pressure Tests: Hybrid Sensor

DC pressure measurements are used to verify the functionality of the sensing system. The following data results from the hybrid sensing system, with sensors and circuits on separate substrates, but within the same package. Pressure is applied to the system via a pressure regulator, and measured with a pressure sensor. The corresponding output bitstream from each of the modulators is averaged to provide an uncalibrated measurement of the pressure.

Figure 8.18 shows the characteristic of the large sensor, $C_L$, which is the output of the third-order modulator. For this measurement, a 500 mV input voltage reference is applied to each of the modulators. The system is clocked at 1 MHz, and 65536 samples are averaged to provide each data point.

8.5 Sensor Calibration

To demonstrate calibration of the sensor, it is necessary to provide an ac pressure signal to the sensor. The signal must be at or below the calibration frequency $f_2$ (or at or below frequency $f_1$ for wideband calibration). However, if the signal is at too low a frequency, the number of samples which must be processed by the signal processing is too large. The limit on the number of samples is set by the SRAM memory size, in this case 1 Mb. Although the clock frequency can be lowered, leakage current in the system places a limitation on the lower bound, and this is not enough to compensate. The pressure regulator is not able to provide ac signals above a very small fraction of a Hertz. Signals measured at 0.1 Hz were of very small pressure amplitude, only a few psi. Additionally, the signal is extremely distorted. Although the voltage-controlling input can be generated from a low-distortion source, the mechanical element produces significant nonlinearity and hysteresis in the signal.
Figure 8.16: Idle-channel spectrum of second-order modulator with sensor capacitors for $f_s$ of 1 MHz, shown up to $f_s/2$. 

Figure 8.17: Idle-channel spectrum of third-order modulator with sensor capacitors for $f_s$ of 1 MHz, shown up to $f_s/2$. 
Figure 8.18: Measured dc characteristic of the large sensor $C_L$. 
Chapter 9

Conclusions

9.1 Summary

Fundamentally, the readout from capacitive microsensors is limited by kT/C noise. Oversampling and decimation of the sensor can alleviate this limitation, providing resolution dependent on the desired output data rate. The achievable speed-resolution product is related to the linearity of the sensor, since in general large sensors, which have significant dynamic range, are more nonlinear. The resolution, speed and linearity are therefore tied together in a tradeoff such that the output speed times the SNR divided by the INL is approximately a constant.

The sensor calibration techniques described in this thesis provide a powerful means for decoupling the resolution, linearity and speed tradeoffs associated with capacitive sensors. By using multiple sensors, the wide dynamic range of a large sensor and the linearity of a small sensor can be simultaneously exploited. This thesis has demonstrated this, using a capacitive sensing system applied to the measurement of pressure. Nonlinearity can be calibrated out of the system, to within the linearity of the calibration standard. The resulting tradeoff is mainly between the speed and resolution of the system.

The continuous nature of the calibration provide significant advantage over typical power-up or look-up table calibration which is costly due to the time to calibrate and the associated memory. In addition, the adaptive calibration techniques are applicable for calibrating temperature and environmental effects as well as the drift of the sensor characteristic.

9.2 Future Work

The calibration techniques described in this thesis have shown the possibility for continuously calibrating nonlinearity in an open-loop system. The adaptive technique described may also calibrate out the nonlinearity of the DAC in an oversampling
system which uses multibit feedback. To date power-up type calibration of this nonlinearity has been demonstrated [88] as well as continuous calibration of the DAC capacitor mismatch [89]. An important future direction for this work is exploration of the adaptive calibration techniques applied to the calibration of general nonlinearities in oversampling converters as well as in other ADC architectures. The extension of these calibration techniques to adaptive methods for full-calibration of the sensor may also be explored.

The demonstrated digital calibration technique is only one way in which the nonlinearity of the sensor can be compensated in an open-loop manner. Other potential open-loop sensing methods may include the use of the translinear principle to decode the input signal to the nonlinearity. The sensing front-end demonstrated in this thesis incorporates the sensors directly into the first stage of the modulator. ADC techniques which can correct nonlinearity in the conversion process itself may be interesting to explore. (For example ADCs for image sensing directly incorporate the gamma and color correction into the ADC.)

On the simulation side, the author is interested in exploring the optimization of the modulator implementation in hardware. The coefficients required to implement third- and higher-order modulators can often be difficult to implement based on unit capacitors. It is possible that a slightly less aggressive loop-filter design may yield an implementation that affords the use of unit capacitors. The idea is to write an optimization procedure that wraps around the basic behavioral simulator for the modulator, and whose goal is to find the best fit unit capacitor multiples that implement something close to the loop filter.
References


REFERENCES


[77] K. Szajda. Personal communication.


[87] M. Perrott. Personal communication.
