Design and Modeling of a 16-bit 1.5MSPS Successive Approximation ADC with Non-binary Capacitor Array

Jianhua Gan
Cirrus Logic, Inc.
2901 Via Fortuna
Austin, Texas 78746, USA

Shouli Yan
Department of ECE
University of Texas at Austin
Austin, Texas 78712, USA

Jacob Abraham
Department of ECE
University of Texas at Austin
Austin, Texas 78712, USA

ABSTRACT
The design and modeling of a high performance successive approximation analog-to-digital converter (ADC) using non-binary capacitor array are presented in this paper. A non-binary capacitor array with 20 capacitors is used to design a 16-bit, 1.5 mega samples per second (MSPS) successive approximation ADC. A perceptron learning rule, originally developed for Artificial Intelligence applications, is used as the capacitor calibration algorithm. The system architecture and the circuit design for the capacitor array, the sampling network and the high performance comparator are discussed. The capacitor weights are adaptively calibrated to match the physical capacitors with better than 22-bit accuracy. Capacitor matching is not a limiting factor to the accuracy. Various sources of noise, interference and distortion are modeled to evaluate their effects and to ensure the robustness of the calibration algorithm. This architecture is especially suitable for mixed-signal VLSI in the Nanometer Era because it relaxes the matching requirement on analog circuitry.

Categories and Subject Descriptors
B.7.1 [**Integrated Circuits**]: Types and Design Styles – *VLSI (very large scale integration).*

General Terms
Design

Keywords
Analog-to-Digital Converter, Successive Approximation, Non-binary Capacitor Array, Calibration.

1. INTRODUCTION
While the fabrication technology in sub-micron and nanometer era renders very high performance digital circuitry, it becomes more and more difficult to build high performance analog circuitry using the same technology. New architecture must be explored to design high performance analog and mixed signal circuitry. The flourish of digital signal processing requires high performance analog-to-digital converter (ADC) and digital-to-analog converter (DAC) to interface with the physical world of analog nature. High-accuracy, high-speed ADCs with greater than 16-bit accuracy and greater than 1 mega-samples per second (MSPS) sampling rates are in high demand.

GLSVLSI '03, April 28-29, 2003, Washington, DC, USA.

Successive approximation register (SAR) converters offer the combination of resolution and speed unmatched by delta-sigma, pipeline or flash type ADCs. SAR's have no latency, and can be multiplexed. Furthermore, the power consumption is relatively low. These features make SAR converters suitable for data acquisition and fiber-optic applications.

A conventional capacitor based charge redistribution successive approximation ADC uses a binary-weighted capacitor array [3] and a comparator as the internal DAC. The advantage of the binary-weighted DAC is that the back-end digital complexity is low. An accuracy of 10 bits can be easily achieved with straightforward design techniques using capacitor based charge redistribution converters because capacitor matching better than 0.1% is common [3]. However, building a binary weighted DAC with greater than 16-bit accuracy and greater than 1MSPS speed is not trivial. Every decision in the converting steps is critical and affects the final accuracy. In order to get better than 10 bits accuracy in charge-redistribution DAC, production laser trimming or other trimming methods are used to guarantee tighter capacitor matching. The production laser trimming is expensive. The accuracy of other trimming methods is usually limited to the size of the smallest capacitor.

Instead of matching the physical capacitors themselves, we take a different approach. We build the capacitor array and adaptively adjust the digital representation of the capacitor: capacitor weights to match the fabricated capacitor array. In our SAR ADC based on non-binary capacitor array, we take advantage of the redundancy in the non-binary capacitor array and the adaptive calibration algorithm to greatly relax the capacitor matching requirement.

2. CAPACITOR ARRAY CALIBRATION ALGORITHM
We want to calibrate the capacitor weights to reflect the physical capacitor values. Then we can use this set of capacitor weights in the successive approximation conversion process by adding the corresponding capacitor weight only when the comparator output is 1. We use the perceptron learning rule [4] to calibrate the capacitor weights. In order to create learning cases, we have to create redundancy in the system. A radix less than 2 capacitor array in the charge-redistribution SAR converter creates redundancy. Based on simulation results, radix 1.8 is a good choice. The exact radix is immaterial and will be calibrated out. In order to get 16-bit resolution, the largest capacitor in the array should be at least 2^16 times greater than the smallest capacitor. We need 20 capacitors in the capacitor array for 16-bit ADC.

The capacitor weight corresponding to capacitor j is denoted as $W(j)$ (j = 19 to 0). The system offset is W_{off}. The capacitor weights are set to initially guessed values. The calibration is not sensitive to these initial weights.

We sample the reference voltage using a random vector $\{A(j)\}$. The bottom plate of capacitor j is switched to the reference voltage Vref if $A(j)$ = 1. The bottom plate of capacitor j is switched to the signal ground Sgnd if $A(j)$ = 0. We can use a linear feedback register (LFSR) to generate this random vector. This will create a charge on the top plate of the array. Add the corresponding capacitance weights and offset W_{off} to get a digital result D_a. Performing successive approximation for 20 cycles, we can get a result in the SAR register $\{B(j)\}$. Add the corresponding capacitor weights to get D_b. D_a and D_b may be different because of the redundancy and the noise in the system. After the SAR conversion process, we get the sign of the residual analog voltage on the top plate of the array: S_a by checking the comparator output. We can also get the sign of D_a minus D_b which is the digital sign: S_d. Now we have a learning case, then we correct the capacitance weights according to the perceptron learning rule:

$$W(j) \leftarrow W(j) + \alpha \times [A(j) - B(j)] \times (S_a - S_d)$$

$$Woff \leftarrow Woff + \alpha \times (S_a - S_d)$$

where α is called the learning rate.

This process is repeated while we decrease α until α is smaller than the desired accuracy level. It has been proven that the weight space has no local minimum [4]. If we calibrate long enough, we will calibrate the capacitance weights to an accuracy level better than the noise level in the system because of the averaging effect in the calibration process. The calibration is only performed at system startup.

3. SYSTEM ARCHITECTURE

We propose a mixed-signal micro-controller architecture to implement the ADC. The block diagram of the SAR ADC is shown in Figure 1. It uses one unified control structure. The address unit, the program ROM and the instruction decoding block generate the control signals. There are separate decoding blocks for the analog and digital blocks to generate the control signals for the respective blocks.

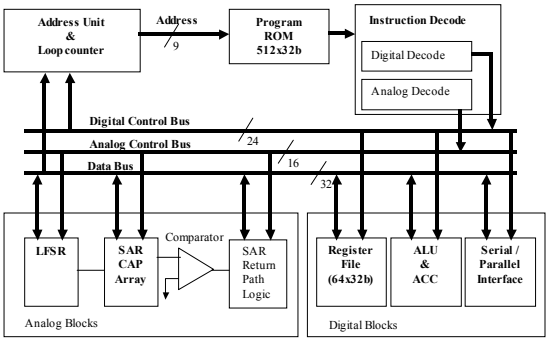

Figure 1. **The block diagram of the ADC**

The analog blocks in the SAR converter include a 20-bit linear feedback shift register (LFSR), a non-binary capacitor array, a comparator and SAR return path logic. The random vector generated by the LFSR is used to control the bottom plate connection of the capacitor array during the sampling phase. The capacitor is connected to the reference voltage Vref if the corresponding bit in LFSR is 1. The capacitor is connected to the signal ground Sgnd if the corresponding bit in LFSR is 0.

The digital blocks include a 64-word by 32-bit register file, a 32-bit arithmetic and logic unit (ALU) and accumulator, and serial or parallel interface. The main block in the ALU is a 32-bit adder. Other peripheral or arithmetic blocks can be added as needed.

This system provides the infrastructure to implement the calibration algorithm as the instructions running on the mixed-signal micro-controller. The system can also handle normal successive approximation conversion process if we can sample the analog input voltage. We only need change the program.

4. CIRCUIT DESIGN

4.1 System Clock

We design a 16-bit, 1.5MSPS successive approximation A/D converter using the non-binary capacitor array. We allocate 12 clock cycles for sampling. The successive approximation conversion takes 20 cycles because there are 20 capacitors in the array. The total number of clock cycles for one complete conversion is 32. The system clock frequency should be 48MHz.

4.2 Non-binary Capacitor Array

The analog power supply voltage is 3.3V. The reference voltage is 2.5V. The target signal-to-noise ratio (SNR) is 92dB. The peak to peak input voltage $V_{in(p-p)}$ is 2.5V. The total root-mean-square (RMS) noise voltage V_n can be calculated as

$$V_n = \frac{V_{in(p-p)}}{2\sqrt{2} \times 10^{92/20}} = 22.2\mu V$$

Two main noise sources: the kT/C noise V_{n_cap} and the comparator noise V_{n_comp}. Assume that the kT/C noise voltage is twice the comparator noise voltage. Then the kT/C noise is $V_{n\_cap} = 19.86\mu V$. The comparator noise is $V_{n\_comp} = 9.93\mu V$.

We know that the kT/C noise is 64μV for a capacitor of 1pF. So the total effective capacitance of the capacitor array should be

$$C_{total} = \left(\frac{64}{V_{n\_cap}} \right)^2 = 10.4\,pF$$

The smallest capacitance is

$$C_{LSB} = \frac{C_{total}}{\sum_{k=0}^{19} 1.8^k} = 0.065\,fF$$

The largest capacitance is $C_{MSB} = C_{LSB} \times 1.8^{19} = 4.6\,pF$.

The smallest capacitor is too small to implement, we divide the array into 3 sections and use two bridge capacitors between adjacent sections to implement this capacitor array. There are 8, 6, and 6 capacitors in the most significant bit (MSB) section, middle section (MID) and the least significant bit (LSB) section,

respectively. The total effective capacitance at the input node of comparator is about 10.4pF. Both bridge capacitors Cb1 and Cb2 are 96fF. The system can tolerate a lot of capacitance variation. The actual capacitance ratio between two adjacent capacitors can deviate from 1.8. The calibration algorithm will make the capacitor weights match the fabricated capacitors.

Assume that the RMS error of capacitor weight is σ. The worst case RMS error is $\sqrt{20}\sigma$. We choose 6 sigma error to be smaller than LSB.

$$6\sqrt{20}\sigma \le 2^{-16} \Rightarrow \sigma \approx 2^{-21}$$

We need calibrate the capacitor weights to better than 21-bit accuracy in order to get 16-bit accuracy for the ADC.

4.3 Sampling Network

The sampling network for one capacitor is shown in Figure 2. The top plates of all the capacitors are connected to one input of the comparator. The other input of the comparator is connected to Vcm. Vcm can be set to half of the power supply voltage. A tracking switch connects the top plate to Vcm during tracking phase. The tracking switch is open in sampling phase. The bottom plate of each capacitor can be switched to the analog input Vin, the reference voltage Vref or the signal ground Sgnd.

PMOS can be used as the switch connected to the reference voltage Vref. NMOS can be used as the switch connected to the signal ground Sgnd. Transmission gate consisting of both NMOS and PMOS can be used as the switch connected to the input voltage Vin. The size of the switch can be derived from the settling requirement and the corresponding capacitance that the switch connects to. In general, the switch of large size is required for larger capacitor. SPICE simulation is used to verify that the sampling network meets the speed, distortion and signal-to-noise requirement.

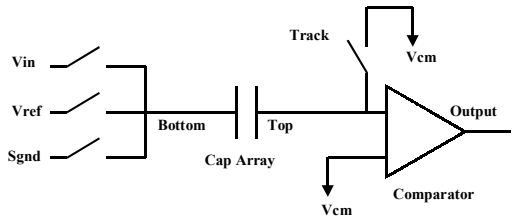

Figure 2. **The sampling network connected to each capacitor**

4.4 High Speed Comparator Design

A comparator architecture consists of 5 preamplifiers and a latch L as shown in Figure 3. It has two modes of operation: tracking and latching. Preamplifiers are enabled to amplify the input difference while the latch is disabled in the tracking mode. In the latching mode, preamplifiers are disabled and the latch is enabled so that the voltage difference is regeneratively amplified in the latch. A logic level will be generated at the latch output during latching mode. Autozeroing techniques [1] can be used to cancel the offset of the comparator. The comparator offset is part of the system offset. There is an offset register corresponding to the system offset. The system offset can be calibrated automatically during the capacitor array calibration.

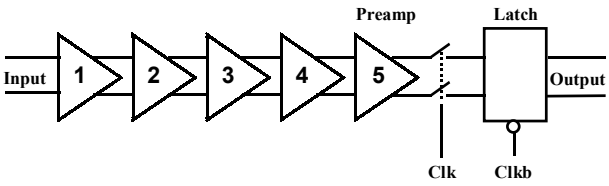

Figure 3. **A comparator with preamplifiers and latch**

The first 4 preamplifier stages are simple low gain and high bandwidth differential amplifier stages. The fifth stage of the preamplifier is high gain and relatively low bandwidth stage. As shown in Figure 4, it is a folded cascode amplifier followed by a source follower as the output stage. PMOS transistors M1 and M2 and NMOS transistors M3 and M4 are the core of the folded cascode amplifier. M5 and M6 make up the output stage. A quasi-autozero technique [2] can remove signal hysteresis by shorting the output momentarily before comparison. M7 is used as the quasi-autozero switch. The common mode feedback circuitry is not shown in Figure 4.

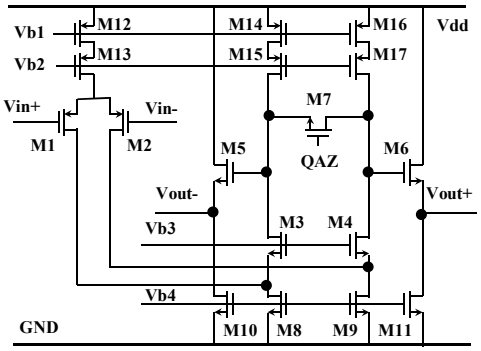

Figure 4. Preamplifier stage 5

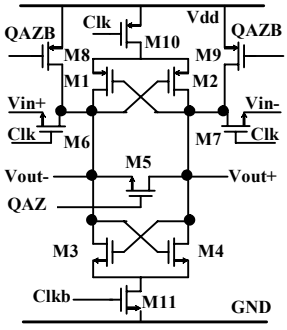

Figure 5. Latch in the comparator

The latch is shown in Figure 5. Transistors M1, M2, M3 and M4 form the cross-coupled latch. M6 and M7 are used to isolate the latch from the preamplifiers when the comparator goes into latching state. They are controlled by clock signal Clk. M5, M8 and M9, controlled by QAZ or QAZB, are used as quasi-autozero switches. Transistors M10 and M11 are controlled by Clk and Clkb, respectively. The latch is disabled when Clk is high and Clkb is low. When Clk is low and Clkb is high, the latch is enabled and it is isolated from the preamplifiers by M6 and M7. The comparator is designed and simulated at transistor level to ensure that it meets the requirement.

4.5 Digital Circuit Design

The digital circuitry is designed and synthesized with Verilog. Estimation shows that the area overhead for implementing the calibration algorithm is about 20%. This overhead is comparable to the overhead in other calibration methods like trimming.

5. SIMULATION RESULTS

We need make sure that the calibration is robust under different conditions. So we consider the effects of the non-idealities in the system such as noise, interference and nonlinearity. We analyze them and model them using high level language in detail. We also simulate them to evaluate their effects and make design trade-off recommendations for the implementation in the future. The noise sources are the kT/C noise in the capacitor array, the charge injection noise from the track switch, the noise from the reference voltage and the noise in the comparator. They affect the dynamic range in the ADC. They also affect the calibration, but the noise effect is averaged in the calibration process.

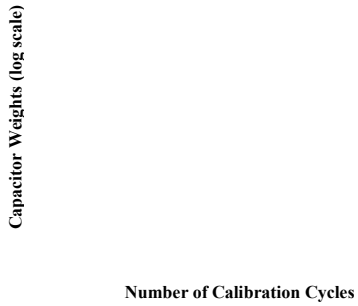

Figure 6. The convergence of capacitor weights in calibration

We implement the capacitor array calibration algorithm based on the perceptron learning rule. The simulation is done in high level language: Verilog. The reference voltage is 2.5V. The offset is assumed to be 10mV. Thermal noise and comparator noise are included in the model. The analog values are represented using floating point numbers. The 20 capacitor weights and one offset converge to the correct values as shown in Figure 6. The capacitor weights are scrambled to a large extent in the early stage. The calibration corrects the capacitor weights based on the learning cases. This adaptive correction mechanism makes the capacitor weights converge nicely to the correct values. The capacitor weights are calibrated with better than 22-bit accuracy consistently under different conditions.

We also perform Verilog simulation for the whole ADC using the capacitor weights after calibration. The simulated ADC output spectrum after the calibration for an input sine wave of -60dB, 150kHz is shown in Figure 7. This demonstrates that the capacitor weights reflect the true capacitor sizes. The capacitor weights converge under different noise, interference and distortion conditions. It can tolerate at least 10% deviation from the nominal value for each capacitor. This is a much relaxed requirement for capacitor matching for a 16-bit ADC. This shows the effectiveness and the robustness of the calibration algorithm. The calibration time is about 50ms.

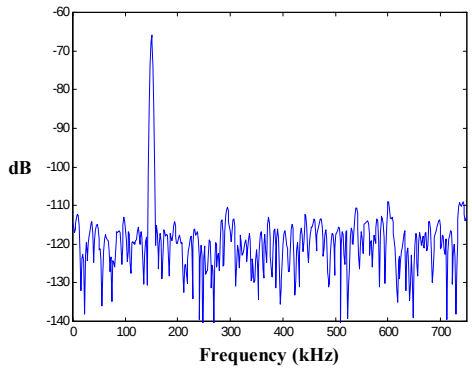

Figure 7. Simulated ADC output spectrum for -60dB input

With 22-bit accuracy for the calibrated capacitor weights, we can potentially get even higher resolution with the same architecture. The limiting factor is the noise in the circuit. Capacitor matching is no longer a limiting factor. There is a trade-off between speed and noise level. We can use larger capacitance in the array if we want to lower the noise level. Larger capacitance will slow down the conversion. Smaller radix will give us more redundancy and will be more forgiving for initial incorrect decisions but it will require more capacitors in the array and longer conversion time.

6. CONCLUSION

The design and modeling of a high performance successive approximation ADC using non-binary capacitor array is presented. A mixed-signal micro-controller architecture is used to implement the relatively complex capacitor array calibration algorithm for SAR converter. The capacitor weights are calibrated with better than 22-bit accuracy after the capacitor array calibration process. Capacitor matching is no longer a limiting factor. It demonstrates the robustness of the calibration algorithm and the flexibility of the mixed-signal micro-controller. It is essential to the design of high speed and high resolution A/D converters. It has even more advantages for Nanometer Era VLSI since it relaxes the matching requirement.

We can use the same architecture to get even higher accuracy. This algorithm may also be used to calibrate other analog circuits such as pipeline ADC and relax the matching requirement.

7. REFERENCES

[1] Allen, P., and Holberg, D., CMOS Analog Circuit Design, Second Edition, Oxford University Press, 2002, pp.483-487.

[2] Kerth, D., U.S. Patent 4748418, Quasi auto-zero circuit for sampling amplifiers, May 31, 1988.

[3] Razavi, B., *Principles of Data Conversion System Design*, IEEE Press, 1995.

[4] Russell, S., and Norvig, P., *Artificial Intelligence: A Modern Approach*, Prentice Hall, 1995, pp.573-578.