SAR ADC Architecture with Digital Error Correction

Masao HOTTA † Akira HAYAKAWA Nan ZHAO
Yosuke TAKAHASHI Haruo KOBAYASHI
† Department of Electronics & Communication Eng., Musashi Institute of Technology
Electronic Engineering Department, Faculty of Engineering, Gunma University
e-mail: mhotta@sc.musashi-tech.ac.jp

Abstract - This paper describes a high-performance successive approximation (SAR) ADC using three comparators operating in parallel, instead of just one as in conventional ADCs. This comparator redundancy enables higher resolution, potentially faster operation, higher reliability and comparator-error correction. We describe a reliability-enhancement and error-correction algorithm that we developed. Embedded microcontrollers containing this SAR ADC would be ideal for automotive applications.
Keywords: Successive Approximation ADC, Error Correction, Automotive Electronics, Microcontroller

1. Introduction

Recently much attention is being paid to automotive electronics [1, 2]. Successive Approximation Register (SAR) ADCs in embedded microcontrollers are widely used in such applications where reliability, speed, accuracy, and low power consumption are required.

This paper describes the architecture and error correction algorithm of a high-reliability SAR ADC suited to automotive applications. Design targets were higher resolution (12-14bit), higher speed (20MS/s), and higher reliability than existing SAR ADCs, as well as low power and low cost (implementation with digital CMOS processes). The proposed SAR ADC has three comparators. We describe how this comparator redundancy can enhance reliability, enable correction of comparator errors, and enable higher speed and accuracy.

Note that error correction is not possible in conventional binary search SAR ADCs using only one comparator. In [3], a non-binary SAR ADC is proposed which can perform digital error correction, but it takes extra steps and hence sampling time becomes slower. In [4], comparison decision time is adjusted to be different at each step for overall higher sampling rate, but it requires complicated time-base circuitry.

2. SAR ADC

2.1 Features of SAR ADC

SAR ADCs are widely used for high resolution (10-12bit), medium speed (≈ 5MS/s), low-power, low-cost applications such as automotive, factory automation, and pen digitizer applications. [3, 5, 6, 7, 8]. SAR ADCs with improved performance, lower cost, and higher reliability can make a significant impact in industry.

2.2 Configuration of SAR ADC

A conventional SAR ADC consists of a track-hold circuit, a comparator, a DAC, SAR logic and time-base circuits (Fig.1). The track-hold circuit, and ensuring linearity of DAC input-output characteristics, are the most critical parts of the design. Usually a ring counter is used in the time-base circuitry to provide accurate timing signals.

2.3 Operation of SAR ADC

The SAR ADC operates according to a binary search algorithm as follows (Fig.2):

- The track-hold circuit samples and holds the voltage of the analog input Vin (full-scale input is Vref).
- The comparator compares the voltages of Vin (held by the track-hold circuit) and Vref /2 (where Vref /2 is generated by the DAC).
- In case Vin > Vref /2:
  The comparator outputs logic ”1”.
  – The comparator then compares the voltage Vin with (3/4)Vref (where (3/4)Vref is generated by the DAC).
  – If Vin > (3/4)Vref, then (7/8)Vref is used for the

next comparison. Else if Vin < (3/4)Vref, then (5/8)Vref is used.

– This binary search continues in this manner.

• In case Vin < Vref /2:

The comparator outputs logic ”0”.

– The comparator then compares the voltages Vin with (1/4)Vref (where (1/4)Vref is generated by the DAC).

– If Vin > (1/4)Vref, then (3/8)Vref is used for the next comparison. Else if Vin < (1/4)Vref, then (1/8)Vref is used.

The SAR ADC performs N comparisons, then outputs a digital value corresponding to the N-bit binary comparison result. Fig.2 illustrates the case for N = 4.

3. Proposed SAR ADC Architecture

We propose using an SAR ADC that has three comparators (or, equivalently, a 2-bit flash ADC) and a DAC with three reference voltage outputs (Fig.3). A conventional SAR ADC has only one comparator, so a wrong comparator decision in any comparison step can not be corrected in a later comparison step (Fig.2). However, our proposed ADC has three redundant comparators; this has the following advantages:

(1) Reliability: Even if the comparator decision is wrong in any comparison step, comparator redundancy allows this error to be corrected in later comparison steps; this is ideal for (e.g. automotive) applications that require high reliability.

(2) High Speed: When such error correction is used, we do not have to wait for DAC comparator-reference voltage outputs to settle completely, hence the converter can operate at a higher frequency.

4. High Reliability SAR Algorithm

The proposed SAR ADC has three comparators (Fig.3), and it obtains 2-bit data in the first comparison step and 1-bit data in each subsequent step. Thus N comparisons result in (N + 1) bit digital data, and the redundancy of the three comparators is used to improve reliability.

Hereafter, let the analog input range of the SAR ADC be from 0 to Vref , and the internal 2-bit ADC output provided by the three comparators at the n-th step be Dout(n).

4.1 Digital Error Correction Algorithm

This section describes our digital error correction algorithm (Fig.4) in the proposed SAR ADC (Fig.3).

(1) In case Vin = (6.1/8)Vref :

Suppose that the outputs of the internal 2-bit ADC which consists of the three comparators are **correct** in the first and second comparison steps.

In the first step: Set the three comparator reference voltages to (6/8)Vref , (4/8)Vref , (2/8)Vref respectively (Fig.3). When the output of the internal 2-bit ADC is correct, it outputs Dout(1) = 11.

In the second step: Set the three reference voltages to (8/8)Vref , (7/8)Vref , (6/8)Vref. When the output of the internal 2-bit ADC is correct, it outputs Dout(2) = 01

Overall ADC output: Dout(1) = 11 and Dout(2) =01, and then the 3 MSBs of overall ADC output becomes 110. (The rule of obtaining overall ADC output is described in Section 4.3.)

(2) In case Vin = (6.1/8)Vref :

Suppose that the output of the internal 2-bit ADC is **incorrect** in the first step, but is correct in the second step.

In the first step: Set the three reference voltages to (6/8)Vref , (4/8)Vref , (2/8)Vref. Consider the case that the output of the internal 2-bit ADC is incorrect and it outputs Dout(1) = 10.

In the second step: Set the three reference voltages to (6/8)Vref , (5/8)Vref , (4/8)Vref. When the output of the internal 2-bit ADC is correct, it outputs Dout(2) = 11.

Overall ADC output: Dout(1) = 10 and Dout(2) =11, and then the 3 MSBs of overall ADC output becomes 110.

4.2 Three Reference Voltages in Proposed Architecture

Let us denote Vr3=Vh(n), Vr2=Vm(n), Vr1=Vl(n) as the three reference voltages of the three comparators in the n-th step (n = 1, 2, 3, ..), and also define

Vr(n) = Vh(n) − Vm(h) (= Vm(n) − Vl(n)).

These reference voltages will be as follows:

• Vh(1) = (3/4)Vref, Vm(1) = (1/2)Vref ,

Vl(1) = (1/4)Vref, Vr(1) = (1/4)Vref .

• Vr(n + 1) = Vr(n)/2.
• In case Dout(n) = “11” :
Vm(n + 1) = Vh(n) + Vr(n + 1).
In case Dout(n) = “10” :
Vm(n + 1) = Vm(n) + Vr(n + 1).
In case Dout(n) = “01” :
Vm(n + 1) = Vm(n) − Vr(n + 1).
In case Dout(n) = “00” :
Vm(n + 1) = Vl(n) − Vr(n + 1).
• Vh(n + 1) = Vm(n + 1)+Vr(n + 1),
Vl(n + 1) = Vm(n + 1) − Vr(n + 1).

4.3 Rigorous Description of Proposed Digital Error Correction Algorithm

Let us consider the 4-step case, because its extension to the N-step case is straightforward. In this 4-step case the ADC digital output is 5 bits and we denote it as b5 b4 b3 b2 b1. Here, b5 is the MSB and b1 is the LSB.

Assumption :

• The output of the internal 2-bit ADC (Dout(n) (n = 1, 2, 3)) in each of the first, second and third steps is either the correct value, the correct value plus one, or the correct value minus one.
• The output of the internal 2-bit ADC in the fourth step Dout(4) is the correct value.

Then we can obtain correct 5-bit output data “b5 b4 b3 b2 b1” and data for correction “c1 c2 c3” from Dout(1), Dout(2), Dout(3), Dout(4) as follows:

In the fourth step :

• In case Dout(4) = “11” : c1 = 1, b1 = 0.
• In case Dout(4) = “10” : c1 = 0, b1 = 1.
• In case Dout(4) = “01” : c1 = 0, b1 = 0.
• In case Dout(4) = “00” : c1 = −1, b1 = 1.

In the third step :

• In case Dout(3) = “11” :
If c1 = 1, then c2 = 1, b2 = 1.
Else if c1 = 0, then c2 = 1, b2 = 0.
Else if c1 = −1, then c2 = 0, b2 = 1.
• In case Dout(3) = “10” :
If c1 = 1, then c2 = 1, b2 = 0.
Else if c1 = 0, then c2 = 0, b2 = 1.
Else if c1 = −1, then c2 = 0, b2 = 0.
• In case Dout(3) = “01” :
If c1 = 1, then c2 = 0, b2 = 1.
Else if c1 = 0, then c2 = 0, b2 = 0.
Else if c1 = −1, then c2 = −1, b2 = 1.
• In case Dout(3) = “00” :
If c1 = 1, then c2 = 0, b2 = 0.
Else if c1 = 0, then c2 = −1, b2 = 1.
Else if c1 = −1, then c2 = −1, b2 = 0.

In the second step :

• In case Dout(2) = “11” :
If c2 = 1, then c3 = 1, b3 = 1.
Else if c2 = 0, then c3 = 1, b3 = 0.
Else if c2 = −1, then c3 = 0, b3 = 1.
• In case Dout(2) = “10” :
If c2 = 1, then c3 = 1, b3 = 0.
Else if c2 = 0, then c3 = 0, b3 = 1.
Else if c2 = −1, then c3 = 0, b3 = 0.
• In case Dout(2) = “01” :
If c2 = 1, then c3 = 0, b3 = 1.
Else if c2 = 0, then c3 = 0, b3 = 0.
Else if c2 = −1, then c3 = −1, b3 = 1.
• In case Dout(2) = “00” :
If c2 = 1, then c3 = 0, b3 = 0.
Else if c2 = 0, then c3 = −1, b3 = 1.
Else if c2 = −1, then c3 = −1, b3 = 0.

In the first step :

• In case Dout(1) = “11” :
If c3 = 1, then b5 = 1, b4 = 1 (overflow).
Else if c3 = 0, then b5 = 1, b4 = 1.
Else if c3 = −1, then b5 = 1, b4 = 0.
• In case Dout(1) = “10” :
If c3 = 1, then b5 = 1, b4 = 1.
Else if c3 = 0, then b5 = 1, b4 = 0.
Else if c3 = −1, then b5 = 0, b4 = 1.
• In case Dout(1) = “01” :
If c3 = 1, then b5 = 1, b4 = 1.
Else if c3 = 0, then b5 = 0, b4 = 1.
Else if c3 = −1, then b5 = 0, b4 = 0.
• In case Dout(1) = “00” :
If c3 = 1, then b5 = 0, b4 = 1.
Else if c3 = 0, then b5 = 0, b4 = 0.
Else if c3 = −1, then b5 = 0, b4 = 0 (underflow).

4.4 Assumption (Limitation) of Digital Error Correction

Let us denote Dout(n) as actual 2-bit ADC output at n-th step, and Dout(n)’ as its correct output. Then the proposed digital error correction algorithm works correctly if and only if

• Dout(4) = Dout(4)’ and

• |Dout(n) − Dout(n)'| < 2, (n = 1, 2, 3).

When this condition is satisfied, the value of b5b4b3b2b1 is correct. ADC output errors that occur in several steps – such as Dout(1) − Dout(1)' = 1, Dout(2) − Dout(2)' = −1 and Dout(3) − Dout(3)' = 1 – are acceptable if the above condition is satisfied.

Example 1 : In the case where analog input Vin = (129/256)Vref : The correct digital output is b5b4b3b2b1 = 10000, and we will obtain the correct result when "Dout(1)""Dout(2)""Dout(3)""Dout(4)" is any of the following:

- "10""01""01""01"
- "10""01""00""11"
- "10""00""11""01"
- "10""00""10""11"
- "01""11""01""01"
- "01""11""00""11"
- "01""10""11""01" etc.

Example 2 : In the case where analog input Vin = (161/256)Vref : The correct digital output is b5b4b3b2b1 = 10100, and we will have the correct result when "Dout(1)""Dout(2)""Dout(3)""Dout(4)" is any of the following:

- "10""10""01""01"
- "11""00""00""11"
- "11""00""01""01"
- "01""11""10""11"
- "10""01""11""01"
- "10""01""10""11"
- "10""10""00""11" etc.

As mentioned above, the proposed digital error correction algorithm allows large redundancy for the upper and lower reference voltages, Vh(n) and Vl(n), of the three comparators, and the middle reference voltage Vm(n) may be settled to final value within +/- 1/2LSB error by the last clock of conversion sequence.

4.5 Conversion time

When proposed error correction is used, we do not need to wait for the DAC outputs as comparator-references to settle completely, hence a higher conversion rate can be achieved. For example, in the case of 10-bit conventional SAR ADC, a settling time of internal DAC into +/- 1/2LSB should be 6.9τ, where τ is time constant of the DAC output. This means that the conversion time of the ADC becomes 69τ at least. On the other hand, when the proposed error correction algorithm is used, the settling time of the DAC output for the middle reference voltage Vm(n) decides the conversion time and it is reduced to 2 τ because the settling error of the DAC can be relaxed to $Vref/2^{n+2}$ as shown in Fig.5. Consequently, the conversion time of the 10-bit ADC can be reduced to only 18τ because the 2 MSBs are decided at the first clock.

5. Conclusions

We have proposed a high-performance SAR ADC architecture that achieves both high speed and high reliability – ideal for automotive applications and the like – by using triple (redundant) comparators and a new error correction algorithm we have developed.

Next we plan to implement this architecture and algorithm efficiently on an IC.

Acknowledgement

We would like to thank STARC (Semiconductor Technology Academic Research Center) which is supporting this research. Thanks are also due to T. Matsuura, A. Abe, M. Kondo, K. Mashiko, N. Takai, H. San and K. Wilkinson for valuable discussions.

References

[1] H. Casier, P. Moern, K. Appeltans, "Technology Consideration for Automotive," Proc. of ESSCIRC, pp.37-41, Leuven, Belgium (Sept. 2004).

[2] ISSCC Short Course, Automotive Technology and Circuits, San Francisco (Feb. 2005).

[3] F. Kuttner, "A 1.2V 10b 20MS/S Non-Binary Successive Approximation ADC in 0.13μm CMOS," Tech. Digest of ISSCC, San Francisco (Feb. 2002).

[4] M. G. Kim, P. K. Hanumolu, U.-K. Moon "A 10MS/s 11b 0.19mm2 Algorithmic ADC with Improved Clocking," Tech. Digest of VLSI Circuit Symposium, Honolulu (June 2006).

[5] M. Banihashemi, Kh. Hadidi, A. Khoei, " A Low-Power, Small-Size 10-Bit Successive-Approximation ADC," IEICE Fundamentals,

vol.E88-A, no.4, pp.996-1006 (April 2005).

[6] T. Komuro, N. Hayasaka, H. Kobayashi, H. Sakayori, " A Practical Analog BIST Cooperated with an LSI Tester", IEICE Trans. Fundamentals, E89-A, no.2, pp.465-468 (Feb. 2006).

[7] B. Razavi, Principles of Data Conversion System Design, IEEE Press (1995).

[8] N. Verma, A. Chandrakasan, "A 25µW 100kS/s 12b ADC for Wireless Micro-Sensor Applications," Tech. Digest of ISSCC, pp.222-223, San Francisco (Feb. 2006).

[9] R. van de Plassche, CMOS Integrated Analog-to-Digital and Digital-to-Analog Converters, 2nd Edition,Kluwer Academic Publishers (2003).

[10] T. Tsukada, K. Takagi, Y. Kita, M. Nagata, "An Automatic Calibration Technique for High Accuracy Converters," IEEE J. of Solid-State Circuits,vol.19, no.2, pp.266-268 (April 1984).

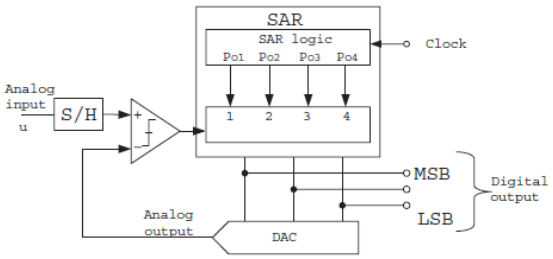

Fig.1: Block diagram of conventional SAR ADC.

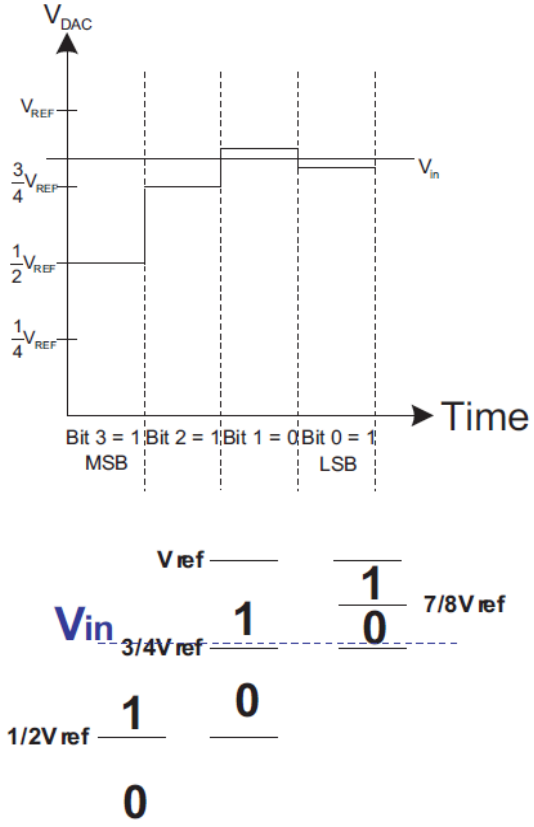

Fig.2: Operation of conventional SAR ADC.

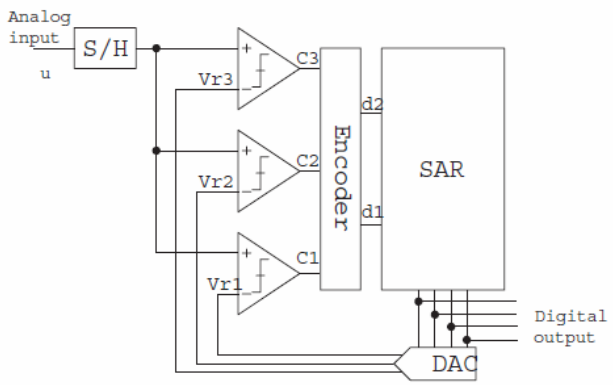

Fig.3: Proposed SAR ADC with three comparators.

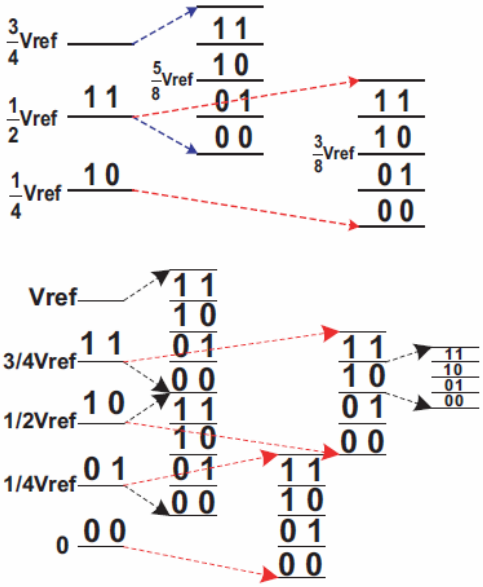

Fig.4: Operation and reliability-improvement algorithm of the proposed SAR ADC.

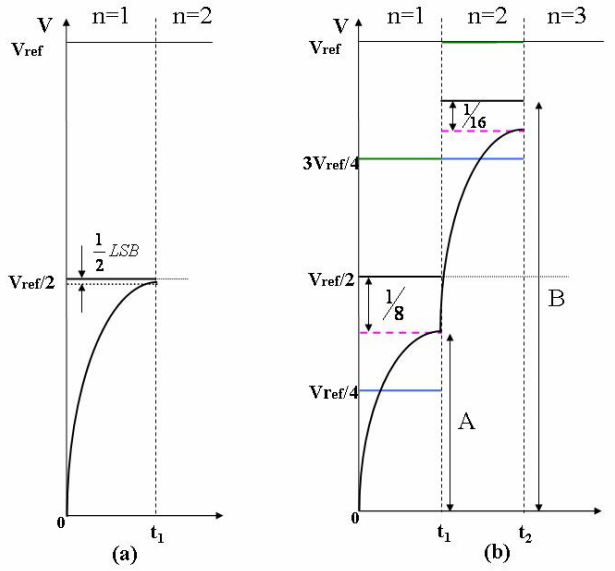

Fig.5: Settling time of the internal DAC as the comparator reference voltage for (a) conventional SAR ADC and (b) proposed SAR ADC.