SAR ADC Algorithm with Redundancy Based on Fibonacci Sequence

Yutaro Kobayashi, Haruo Kobayashi
Division of Electronics and Informatics, Gunma University
1-5-1 Tenjin-cho Kiryu 376-8515 Japan
t14804039@gunma-u.ac.jp k_haruo@el.gunma-u.ac.jp

***Abstract*— This paper describes redundancy algorithm design methods to improve reliability and conversion speed by digital error correction for Successive Approximation Register analog-to-digital converter (SAR ADC), based on number theory. Especially we show that using Fibonacci and Lucas sequences (which have interesting properties such as the closest terms ratio called "golden ratio" and realization of all terms with integers), we can design well-balanced redundancy algorithms for SAR ADC. We also present some derived equations and beautiful properties related to the redundancy design for SAR ADC using Fibonacci sequence.**

Keywords*— *Successive Approximation; ADC; Redundancy; Digital Error Correction; Fibonacci Sequence; Golden ratio

I. INTRODUCTION

Recently, automotive electronics are gathering attention for industry competitiveness of vehicles. Therefore, SAR ADCs embedded in micro-controller for automotive electronics application need better performance such as high reliability, high speed and high resolution, and we study here about redundancy design of SAR ADCs to realize them.

Redundancy design enables digital error correction for SAR ADC [1, 2]. One redundancy design method is to use a non-binary search algorithm instead of a binary search algorithm. There, extra comparison steps and a non-binary weighted DAC are needed for a redundant SAR ADC and we have to determine its non-binary weighted values. Generally, their values are determined using a non-binary radix or selected flexibly by the SAR ADC designer. These methods may not be efficient enough; in other words, the efficient and systematic redundant SAR ADC algorithm design has not been studied well yet.

In this paper we discuss several methods to design redundant SAR algorithms based on number theory, and especially, we show that we can obtain well-balanced non-binary weight values by applying properties of Fibonacci sequence such as the closest terms ratio called "golden ratio" and realization of all terms with integers (without fraction) [3]. Accordingly we show several important properties and their proofs of the redundancy design for SAR ADC utilizing Fibonacci sequence.

II. SAR ADC

SAR ADCs are used for medium sampling speed and high-resolution applications. Since they have features of low power, small chip area, they are widely applied to such as automotive, factory automation. Furthermore it does not require operational amplifiers, which is suitable for nano-CMOS implementation.

The SAR ADC consists of a sample-and-hold circuit, a comparator, a DAC, SAR logic and a timing generator (Fig 1). For precise AD conversion, enough accuracy of the sample-and- hold circuit and the DAC is required.

Conversion of the SAR ADC is based on principle of balance and generally it uses the binary search algorithm. Firstly, the sample-and-hold circuit acquires analog input voltage. Secondly, the comparator compares the input analog voltage and the reference voltage that is generated by the DAC and decides 1bit digital output. Thirdly, SAR logic provides DAC input based on the comparator output. The input voltage and the updated DAC output voltage are compared by the comparator. This operation is repeated and finally SAR ADC can obtain the whole digital output.

Fig.2 shows the binary search algorithm of a 4bit SAR ADC. The bold line in Fig.2 indicates the reference voltage value to compare with the analog input at each step. Their values are calculated by either sum or difference between the last step reference voltage and the weighted voltage p(k) of each step as shown in Fig.2. The comparator outputs 1 if the input voltage is larger than the reference voltage; otherwise outputs 0. Then we obtain the digital output. In many cases, the weighted voltage p(k) is a binary weighted value because the binary search algorithm is popular as an efficient algorithm.

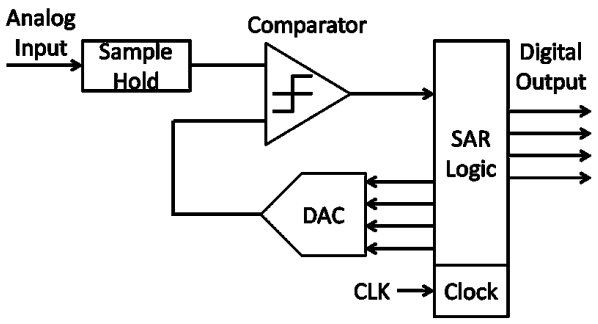

Fig. 1. Block diagram of a 4bit SAR ADC.

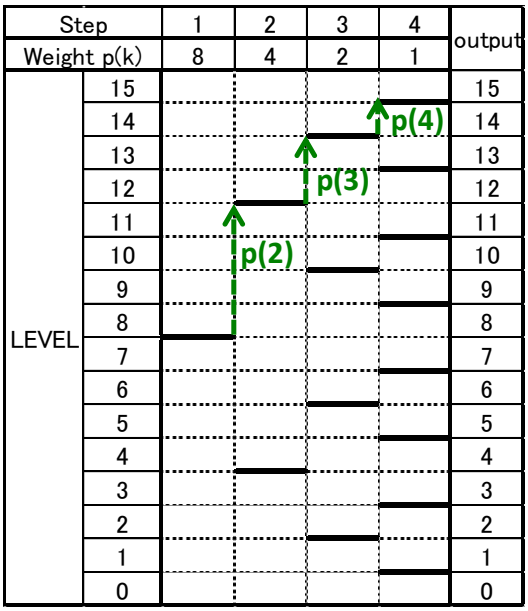

Fig. 2. Binary search algorithm of a 4-bit 4-step SAR ADC.

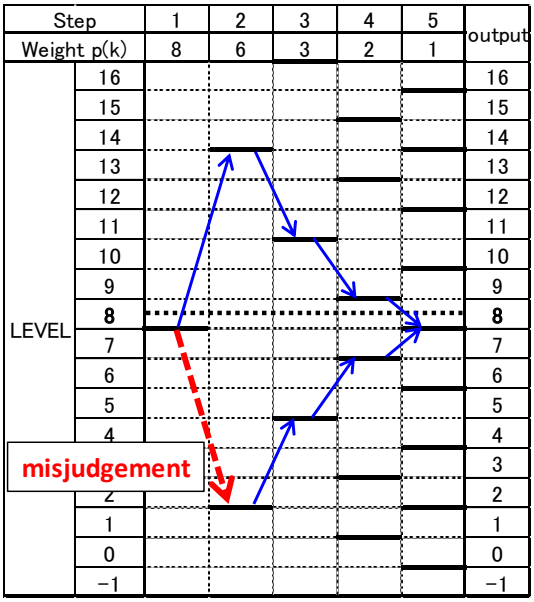

Fig. 3. Operation of a redundant search algorithm of a 4-bit 5-step SAR ADC in case of correct judgment and incorrect judgment.

However in actual applications there is possibility of comparator misjudgment due to DAC incomplete settling, and sample-and-hold circuit incomplete settling as well as noise. In the binary weighted SAR ADC (where binary and decimal codes are mapped to each other with one-to-one), one misjudgment of the comparator leads to incorrect output and low reliability. Hence this paper investigates redundancy design of SAR ADC to enable digital error correction for misjudgment of the comparator.

III. Redundancy Design of SAR ADC

A. Summary of SAR ADC redundancy design

Redundancy design is a popular technique to improve circuit and system performance. To apply the redundancy design to the SAR ADC means adding extra comparison [1, 2]. This method changes binary weights to non-binary weights for the DAC and realizes digital error correction with redundancy property.

Fig.3 shows an example of two redundant search operations of a 4bit 5step SAR ADC. There, the input voltage is 8.6LSB and the reference voltage weights p(k) are 1, 2, 3, 6 and 8. The one operation (solid arrows) assumes that the comparison is correct, whereas the other (dotted arrows) assumed that it is incorrect. However both obtain the correct digital output of 8 by digital error correction. In the 4bit 5step SAR ADC as shown in Fig.3, there are 2^5 comparison patterns and 2^4 output patterns. In other words, a given output level can be expressed by multiple comparison patterns. Therefore even if comparator decision is wrong at some steps, the correct ADC output may be obtained. This is the basic principle digital error correction, which can contribute reliability improvement. In addition, even if the number of the comparison steps is increased, the digital error correction enables high-speed AD conversion as a whole, because the digital error correction can take care of the DAC incomplete settling [1]; thus redundancy design has potential for reliable and/or high-speed SAR AD conversion.

B. Generalization of redundant SAR ADC design

We generalize SAR ADC redundancy design from using some equations [1]. If we realize an N-bit resolution SAR ADC by M-step comparison(M ≥ N), the reference voltage $V_{ref}(k)$ at k-th step and ADC output D_{out} are defined by Eq.(1) and Eq.(2) respectively. Here k = 1,2,3,4, … , M and p(k) is the reference voltage weight value for addition to (or subtraction from) the DAC input in the previous step. Moreover, each d(k) is decided by the comparator output. If the comparator digital output at k-th step is 1, then d(k) = 1, and if the comparator digital output at k-th step is 0, then d(k) = -1.

$$\mathbf{V_{ref}(k)} = \sum_{\mathbf{i=1}}^{\mathbf{k}} \mathbf{d(i-1)p(i)} \tag{1}$$

$$\mathbf{D_{out}} = \mathbf{0.5d(M) - 0.5} + \sum_{\mathbf{i=1}}^{\mathbf{M}} \mathbf{d(i-1)p(i)} \tag{2}$$

$$\textbf{where } \mathbf{d(0) = 1}$$

We can also define “the redundancy at k-th step q(k)” as Eq.(3).

$$\mathbf{q(k)} = \mathbf{-p(k+1) + 1} + \sum_{\mathbf{i=k+2}}^{\mathbf{M}} \mathbf{p(i)} \tag{3}$$

Here q(k) indicates correctable difference between the input voltage and the reference voltage [1]. Even if the comparator result is wrong in the k-th step, we can obtain the correct output as long as $q(k) > |V_{in} - V_{ref}(k)|$ is satisfied. Fig.4 shows q(k) as an example of Fig.3. In Fig.4, one-way arrows indicate q(k), while two-way arrows show correctable input ranges. As shown in Fig.4, since $q(1) > |V_{in} - V_{ref}(1)|$ is satisfied, the SAR ADC can obtain the correct output in

Fig.3. Therefore q(k) expresses the digital error correction capability. Moreover q(k) is defined by only the reference voltage weight p(k) in Eq.(3), and thus p(k) is the most important parameter in the redundancy SAR ADC algorithm design.

C. Conventional method to decide reference voltage weight

Only reference voltage weight p(k) decides correction capability of the redundant SAR ADC, which needs extra circuit elements to realize redundancy. However if the design of the reference voltage weight p(k) is not appropriate, the SAR ADC cannot have the maximum compensation ability.

The ratio of the reference voltage weights p(k+1)/p(k) must be between 1(unary) to 2(binary). In conventional methods, we can obtain the k-th step reference voltage weight p(k) based on the radix $\boldsymbol{x}$ in Eq.(4). Here, N is the ADC resolution, and M is the number of the whole steps.

$$\mathbf{p(k) = x^{M-k}} \tag{4}$$

Here

$$\mathbf{p(1) = 2^{N-1}} \tag{5}$$

$$\mathbf{1 < x < 2}$$

Additionally, the total number of steps M has to satisfy Eq.(6) to enable all output level expression.

$$\mathbf{2^{N-1} - 1 \le \sum_{i=0}^{M-2} p(M-i)} \tag{6}$$

We can systematically decide conditions for redundancy design based on the above equations.

D. Conventional method Issues

Conventional methods may have some issues.

First, the reference voltage weight p(k) in Eq.(4) is not an integer which is not suitable for the circuit design. Since the reference voltage weights p(k) must be integers for conversion accuracy, its rounding to an integer is needed to determine p(k). However rounding operation causes changing of the radix and variability of the correction capability q(k) at each step, and they may disturb performance improvement.

In addition, there is difficulty of appropriate radix choice. Fig.3 shows an example in case of radix 1.80 and using rounding. However in Fig.4, two-way arrows indicate that correctable input range cannot cover all input range, which means that there are some ranges that cannot be corrected. In Fig.4, if ADC input is not within the range of 1~3, 7~9, 13~15[LSB], redundancy design becomes meaningless. Thus the inappropriate selection of a radix loses redundancy design effectiveness. On the other hand, the selection of a small radix to increase value of q(k) induces increasing of comparison steps and hence conversion time. In this way, there is a trade-off between correction capability and conversion speed, and the SAR ADC designer is forced to search a radix that is the most suitable for SAR ADC; these are causes of design difficulty.

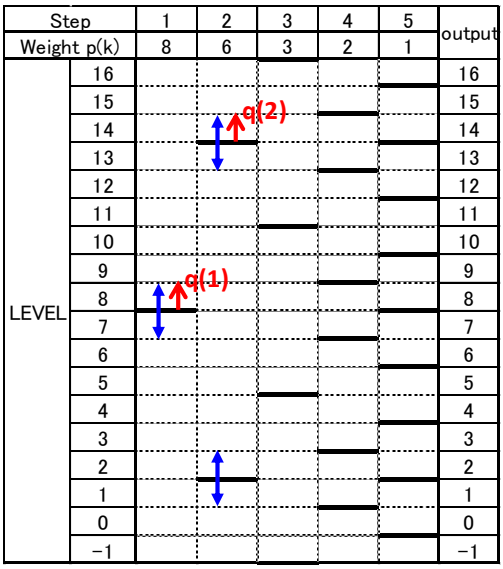

Fig. 4. Redundant search algorithm of a 4-bit 5-step SAR ADC and definition of correctable difference q(k).

IV. Redundancy Design Based on Fibonacci Sequence

Redundancy design has possibility of a high reliability and high speed SAR ADC, but we need further investigation of designing the redundant algorithm. Then we propose here a redundancy design method based on "Fibonacci sequence".

A. Fibonacci sequence

Fibonacci sequence is defined by a recurrence relation as shown in Eq.(7), where $\boldsymbol{n}$ in Eq.(7) is an integer greater than or equal to 0. Fibonacci sequence is presented in 1202 by Leonardo Fibonacci, who was a mathematician in Italy and it is known as one of the most famous number theories [3].

$$\boldsymbol{F_{n+2} = F_n + F_{n+1}} \tag{7}$$

$$\textbf{where } \boldsymbol{F_0 = 0,\ F_1 = 1}$$

Fibonacci numbers are expressed as the following by calculating Eq.(7).

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181…

In short, the sum of neighboring two terms is next term.

In addition, the closest terms ratio of Fibonacci sequence converges at about 1.62 as shown Eq.(8).

$$\boldsymbol{\lim_{n\to\infty} \frac{F_n}{F_{n-1}} = 1.618033988749895 = \varphi} \tag{8}$$

This ratio is called "Golden ratio" and it is widely recognized as the most beautiful ratio.

Fibonacci sequence and Golden ratio are based on very simple rules like the above. However we can find them in various places of our surroundings such as nature and humanity, and they have many interesting and unique properties. Thus they have been studied by many researchers for more than 800 years.

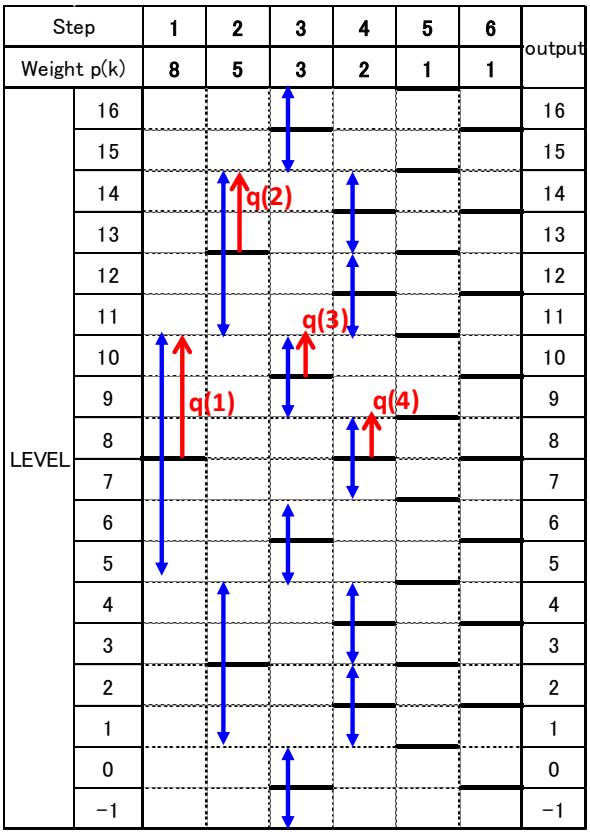

Fig. 5. Non-binary search algorithm using Fibonacci sequence of a 4-bit 6-step SAR ADC.

B. *Fibonacci sequence application to redundant SAR ADC design*

Eq.(7) indicates that Fibonacci sequence numbers are integers, and Eq.(8) indicates that the closest term ratio of Fibonacci number converges at about 1.62 called Golden ratio. In other words, Fibonacci sequence can generate a number string at radix 1.62 with only integer terms. In general, multiplication result of an integer and a decimal fraction is a decimal fraction, nevertheless multiplication result of an integer and a decimal fraction (1.62) is an integer in Fibonacci sequence. Therefore we can apply Fibonacci sequence to the redundancy algorithm design of the SAR ADC using effective properties of the fixed rate and integer terms.

We select the reference voltage weight p(k) by using the Fibonacci sequence method as shown in Eq.(9).

$$\mathbf{p(k)} = \boldsymbol{F_{M-k+1}} \tag{9}$$

$$\mathbf{where\ p(1)} = \mathbf{2^{N-1}}$$

In short, we set p(k) to Fibonacci number in ascending order. Since p(k) follows the property of Fibonacci sequence, the proposed method can realize radix 1.62 by using only integers. Here the total number of steps M is satisfies Eq.(6).

Fig.5 shows a redundant search operation of a 4-bit 6-step SAR ADC using Fibonacci sequence as shown in Eq.(9). One-way arrows indicate q(k) and two-way arrows show correctable input range as shown in Fig.4.

C. *Discovery and proof of the proposed method*

We have discovered two interesting properties in Fig.5 as follows:

1) Correctable difference q(k) is always Fibonacci number F_{M-k-1}.

2) q(k) of k-th step is exactly in contact with q(k+1) of k+1-th step without overlap. In other words, the tips of two-way arrows of k-th step and k+1-th step point exactly the same level as shown in Fig.5. This means that Fibonacci weight is q(k) boundary between overlap and separating, and Fibonacci number weight is the fastest weight to contact with q(k) of adjacent two steps.

Here we prove two above properties.

(Proof of property 1)

First, we determine sum of n ($n = 1,2,3,\dots$) terms of Fibonacci numbers from the first term. Since n is an integer of 1 or more, it follows from Eq.(7) that

$$\boldsymbol{F_n} = \boldsymbol{F_{n+2}} - \boldsymbol{F_{n+1}} \tag{10}$$

Thus we can obtain the sum of Fibonacci numbers from 1 to n as follows:

$$\begin{aligned}\sum_{i=1}^{n} \boldsymbol{F_i} &= \boldsymbol{F_1} + \boldsymbol{F_2} + \boldsymbol{F_3} + \cdots + \boldsymbol{F_n} \\ &= \boldsymbol{F_{n+2}} - \boldsymbol{F_2} \\ &= \boldsymbol{F_{n+2}} - \mathbf{1}\end{aligned} \tag{11}$$

The sum of Fibonacci numbers from 1 to n is equal to the value obtained by subtracting 1 from the Fibonacci number after two terms.

Second, we assume the step number k value as shown in Eq.(12).

$$\mathbf{1} \le \mathbf{k} < \mathbf{M} - \mathbf{1} \tag{12}$$

Since the correctable difference q(k) is defined by Eq.(3) and the reference voltage weight p(k) is defined by Fibonacci number as shown in Eq.(9), we have following:

$$\mathbf{q(k)} = -\boldsymbol{F_{M-k}} + \mathbf{1} + \sum_{i=1}^{M-(k+1)} \boldsymbol{F_i} \tag{13}$$

Here we transform Eq.(13) using Eq.(7) and Eq.(11) as follows:

$$\begin{aligned}\mathbf{q(k)} &= -\boldsymbol{F_{M-k}} + \mathbf{1} + \left(\boldsymbol{F_{\{M-(k+1)\}+2}} - \mathbf{1}\right) \\ &= -\boldsymbol{F_{M-k}} + \mathbf{1} + \boldsymbol{F_{M-k+1}} - \mathbf{1} \\ &= \boldsymbol{F_{M-k+1}} - \boldsymbol{F_{M-k}} \\ &= \boldsymbol{F_{M-k-1}}\end{aligned}$$

$F_0 = 0$ is the minimum number of Fibonacci number. Thus we can define considered area of k value as following:

$$\mathbf{M - k - 1 > 0}$$
$$\mathbf{k < M - 1}$$

It shows that Eq.(12) is right.

Therefore we can obtain the correctable difference q(k) as Eq.(14) and we know that q(k) is always Fibonacci number.

$$\mathbf{q(k)} = \boldsymbol{F_{M-k-1}} \tag{14}$$

(Q.E.D)

(Proof of property 2)

Reference voltage weight of k-th step ($k \geq 2$) is expressed by Eq.(9). Consequently, the reference voltage V_{ref} has the difference of F_{M-k+1} for each step. We obtain Eq.(15) by transforming of Eq.(9) using Eq.(7).

$$\begin{aligned}\mathbf{p(k)} &= \boldsymbol{F_{M-k+1}} \\ &= \boldsymbol{F_{M-k}} + \boldsymbol{F_{M-k-1}} \\ &= \boldsymbol{F_{M-(k-1)-1}} + \boldsymbol{F_{M-k-1}}\end{aligned} \tag{15}$$

Here we replace Fibonacci number of Eq.(15) to the correctable difference q(k) by using Eq.(14), and we obtain Eq.(16) as follows:

$$\mathbf{p(k)} = \mathbf{q(k-1)} + \mathbf{q(k)} \tag{16}$$

Reference voltage difference p(k) between (k-1)-th step and k-th step is expressed by sum of q(k-1) in (k-1)-th step and q(k) in k-th step. Thus q(k) of k-th step comes in contact with q(k+1) of k+1-th step. Moreover, since we can recognize that Eq.(16) is realized regardless of the number of steps, Fibonacci number weight is the fastest weight to contact with q(k) of adjacent two steps.

(Q.E.D)

D. Application of Lucas sequence

We consider application of Lucas sequence as development of using Fibonacci sequence to SAR ADC redundancy design.

Lucas sequence and Lucas number [3] are defined by Luca Edouard as Eq.(17).

$$\boldsymbol{L_{n+2}} = \boldsymbol{L_n} + \boldsymbol{L_{n+1}} \tag{17}$$

$$\textbf{where } \boldsymbol{L_0} = \mathbf{2}, \boldsymbol{L_1} = \mathbf{1}$$

Lucas sequence is the one where the initial value of Fibonacci sequence is changed, and it is only the sequence that can realize golden ratio besides Fibonacci sequence. Property 2 of the redundancy design using Fibonacci sequence is valid on the condition that the radix is golden ratio 1.62. Hence we came up with the usage of Lucas number.

Fig.6 shows a redundant search operation example of a 4-bit 6-step SAR ADC using Lucas sequence as in Eq.(17). There the reference voltage weight p(k) is Lucas number in ascending order. We can discover two properties in Fig.6 that q(k) is Lucas number and q(k) of k-th step comes exactly in contact with q(k+1) of k+1-th step. Their properties are the same as the properties of Fibonacci sequence redundancy design, which is due to the factor that formula of sum of terms from 1 to n is the same as Fibonacci number: therefore their properties can be proven in the same way. However the results re-confirm us that the golden ratio is boundary condition.

E. Effectiveness of redundancy design using number theorys

We have obtained two properties by applying Fibonacci sequence and Lucas sequence to the redundancy design. In particular, the property 2 is important for design of redundant SAR ADC algorithm due to the following two reasons:

First, the property can be a standard of all redundancy design in the viewpoints of the radix of Fibonacci sequence and Lucas sequence which is golden ratio 1.62, and the boundary condition of q(k). Hence, we can assume that q(k) becomes overlap, non-overlap or separation by using golden ratio. If the radix value is larger than the golden ratio, the redundancy is small and q(k) boundaries are separated as shown in Fig.4. On the other hand, if the value of the radix is smaller than the golden ratio, the redundancy is large and q(k) boundaries are overlapped. Thus we can easily select the radix by considering the golden ratio as the standard.

Second, the redundancy design using Fibonacci sequence and Lucas sequence can be considered as the most efficient design. The property 2 indicates that q(k) covers wide input range by minimum extra comparison steps. Therefore, we can realize the redundancy design without waste by only integer terms. Moreover even if we change the first step reference voltage, property 2 holds based on Eq.(16), which means that the redundancy design using Fibonacci sequence and Lucas sequence is flexible. By comparing Fibonacci and Lucas sequence weights, Lucas number weights are slightly bigger. Therefore Lucas number weights can realize fewer-step SAR ADC. However Lucas number weights cannot compensate the third step from the last. So Fibonacci sequence is more suitable for redundancy design.

The above two statements show that proposed method using Fibonacci sequence and Lucas sequence can solve the problems of conventional methods and contribute the efficient redundant algorithm design.

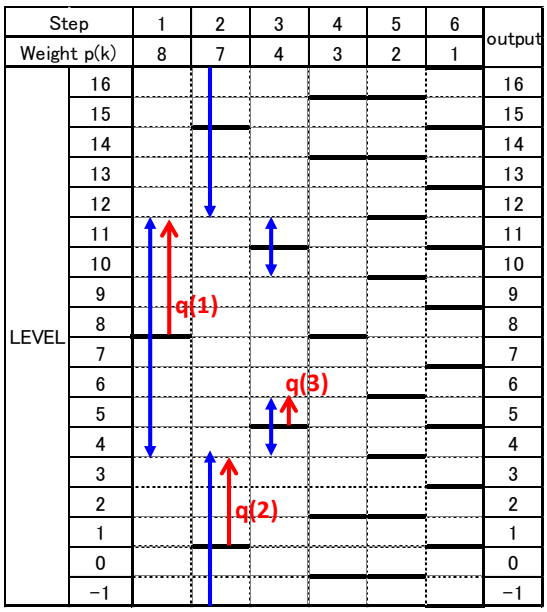

Fig. 6. Non-binary search algorithm using Lucas sequence of a 4-bit 6-step SAR ADC.

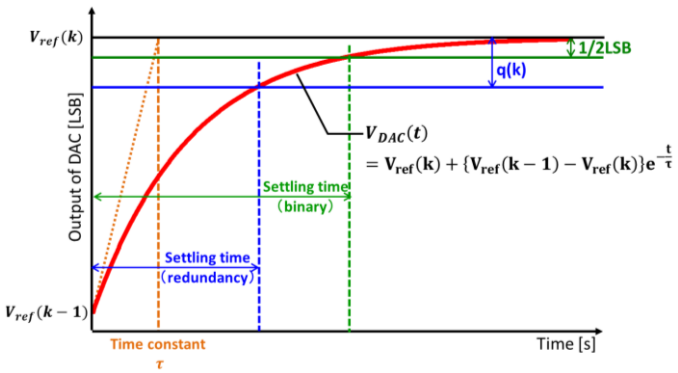

Fig. 7. Principle of settling time acceleration with incomplete settling.

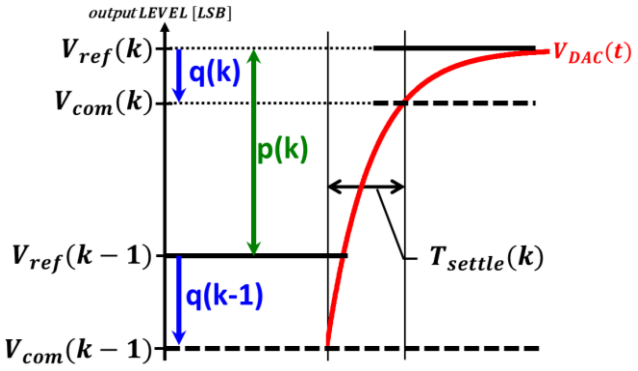

Fig. 8. Schematic diagram to derive settling time equation.

V. DAC Incomplete Settling

A. Summary and Generalization of DAC Incomplete Settling

An SAR ADC contains a DAC that outputs reference voltage by result of comparison at previous step. Since the DAC output must change from previous reference voltage to next one, the DAC output takes time to settle. In binary search algorithm which does not have correctable difference q(k), the DAC must take time to settle between output voltage of DAC and next reference voltage within 0.5LSB for accurate conversion. This DAC settling time often dominates the SAR ADC conversion time. Besides, this settling time is much longer for high resolution SAR ADC. On the other hand, in non-binary search algorithm which has correctable difference q(k), the DAC can decrease settling time, thanks to redundancy and digital error correction at the following steps as shown in Fig.7. Difference between the DAC output voltage and the next reference voltage can be smaller than q(k) to accurate conversion when conversion step has correctable difference q(k).

We generalize SAR ADC incomplete settling by using a simple RC model (a first-order system) as shown Fig.7. (However, here we do not consider settling time to the first step reference voltage, or half-scale reference voltage.) Firstly, we can get output voltage of DAC as Eq.(18) from Fig.7.

$$\mathbf{V_{DAC}(t) = V_{ref}(k) + \{V_{ref}(k-1) - V_{ref}(k)\}e^{-\frac{t}{\tau}}} \qquad (18)$$

Here, τ is time constant of DAC output.

To satisfy correctable condition at the redundant SAR ADC, difference between input voltage of the comparator and the reference voltage has to be smaller than q(k). Thus we can use comparison voltage V_{com}, that has distance q(k) from the original reference voltage, to compare the input voltage. Consequently settling time $T_{settle}(k)$ which is the time to make k-th step comparison voltage can be the time to change the last comparison voltage $V_{com}(k-1)$ into next comparison voltage $V_{com}(k)$. As we should consider the longest settling time to decide each step settling time, we obtain settling time $T_{settle}(k)$ as Eq.(19) in Fig.8.

$$\mathbf{T_{settle}(k) = \tau \ln\left(\frac{p(k) + q(k-1)}{q(k)}\right)} \qquad (19)$$

Note that if correctable difference q(k) is less that 1LSB, we can regard q(k) as 0.5LSB.

Finally, a variable clock SAR ADC takes sum of T_{settle} as total settling time. However, for a fixed clock SAR ADC total settling time is equal to the longest span of T_{settle} multiplied by the number of steps of SAR ADC used.

B. Analysis of Fibonacci SAR ADC settling time

We consider settling time of the redundant SAR ADC using Fibonacci sequence in theory. In Fibonacci sequence SAR ADC, we can transform Eq.(19) to Eq.(20) by using Eq.(9) and property 1.

$$\mathbf{T_{settle}(k) = \tau \ln\left(\frac{F_{M-k+1} + F_{M-k}}{F_{M-k-1}}\right)} \qquad (20)$$

Here we transform Eq.(20) using Eq.(7) and Eq.(8) as follows:

$$\begin{aligned}\mathbf{T_{settle}(k)} &= \mathbf{\tau \ln\left(\frac{(F_{M-k} + F_{M-k-1}) + F_{M-k}}{F_{M-k-1}}\right)} \\ &= \mathbf{\tau \ln\left(2\frac{F_{M-k}}{F_{M-k-1}} + 1\right)} \\ &= \mathbf{\tau \ln(2\varphi + 1)}\end{aligned}$$

Therefore we obtain settling time of k-th step at SAR ADC using Fibonacci sequence as shown Eq.(21).

$$\mathbf{T_{settle}(k) = 1.444\tau} \qquad (21)$$

Eq.(21) indicates that settling time is constant regardless of step number k or usage of variable clock. Likewise, Lucas sequence also realizes constant settling time owing to properties that are the same as properties of Fibonacci sequence. On the other hand, conventional method using radix cannot realize constant settling time, because reference voltage weight p(k) does not have relationship for correctable difference q(k).

C. Simulation of Fibonacci sequence SAR ADC settling time

We compare the method of Fibonacci sequence and the method of radix in terms of redundant SAR ADC settling time. We carry out comparison at 8bit SAR ADC under the condition that variable clock and fixed clock, and get Fig.9 by using Eq.(19). In Fig.9, the vertical axis indicates total settling time and the horizontal axis indicates radix of conventional method. And total settling time of Fibonacci sequence is shown as a horizontal straight line in Fig.9 for comparison but radix of Fibonacci sequence is about 1.618.

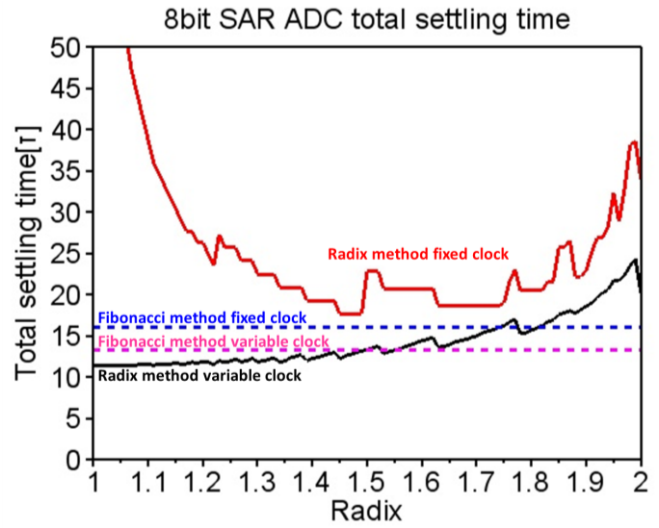

Fig. 9. Result of each method settling time simulation with 0.001 increment of radix.

TABLE I. SAR ADC TOTAL SETTLING TIME

Total settling time			Resolution[bit]				
			6	8	10	12	14
Total settling time[τ]	variable clock	Binary	11.20	20.26	32.06	46.61	63.95
		Radix	8.31	11.63	14.26	17.04	19.81
		Fibonacci	9.00	13.33	17.66	21.99	26.33
	fixed clock	Binary	17.33	33.96	56.14	83.87	117.14
		Radix	12.47	17.70	22.53	28.97	33.27
		Fibonacci	11.27	16.09	20.92	25.75	30.58

From Fig.9, the lower radix ADC has, the shorter time ADC needs at variable clock. On the other hand, surprisingly, Fibonacci sequence can realize the shortest settling time at fixed clock. Results of total settling time at some resolution are shown in Table I. We see from Table I that total settling time using Fibonacci sequence is shorter than total settling time using radix in any resolutions. In case of Lucas sequence, settling time of method using radix is the shortest at both of the two-type clock. This result indicates that the proposed method of Fibonacci sequence is superior to conventional method of radix method from the perspective of conversion time.

VI. CONCLUSIONS

In this paper we have proposed a redundancy SAR ADC algorithm design method with applying properties of Fibonacci sequence such as the closest terms ratio called golden ratio and realization of all terms with integer. As a result, we have obtained some important and beautiful properties of the radix, the error correctable range with digital calibration for the SAR ADC and incomplete settling time. Besides, we found that design method using Fibonacci sequence improves settling time compared to conventional redundant SAR ADCs. These results indicate that the proposed method using Fibonacci sequence contribute to realization reliable and high-speed SAR AD conversion.

ACKNOWLEDGMENTS

We would like to thank T. Matsuura, K. Asami, and K. Kato for valuable discussions. Thanks are also due to STARC which is supporting this project.

REFERENCES

[1] T. Ogawa, H. Kobayashi, Y. Takahashi, N. Takai, M. Hotta, H. San, T. Matsuura, A. Abe, K. Yagi, T. Mori,"SAR ADC Algorithm with Redundancy and Digital Error Correction", IEICE Trans. Fundamentals, vol.E93-A, no.2, pp.415- 423 (Feb. 2010).

[2] F.Kuttner, "A 1.2V 10b 20MSample/s non-binary successive approximation ADC in 0.13μm CMOS," Tech. Digest of ISSCC, (Feb. 2007)

[3] Alfred S.Posamentier, Ingmar Lehmann, Syunsuke Matsuura: "*Husigina suuretsu fibonacci no himitsu*[The Fabulous FIBONACCI Numbers]", NikkeiBP, (Aug.2010)