# A Digital-to-Frequency Converter Using Redundant Signed Binary Addition

Wickham Chen, Mitchell A. Thornton, Ping Gui
Department of Electrical Engineering, SMU
Dallas, TX 75275-0338

*Abstract*— **Redundant signed binary addition (RSBA) has been used to create high performance arithmetic circuits. This paper presents an accumulator-based digital-to-frequency (DFC) converter employing redundant signed binary addition (RSBA). RSBA is advantageous in that no carry propagation occurs resulting in constant delay regardless of operand word size. Utilizing RSBA in the proposed DFC resolves the performance bottleneck in the DFC's conventional implementation and achieves extremely high frequency resolution. In addition, a new RSBA-based 8:1 Multiplexer is introduced for a complete RSBA implementation of the DFC. Experimental results show an increase of more than 3.5 times in the speed of the accumulator compared to the conventional implementation regardless of bit size of the adder.**

*Index Terms*— **Redundant Signed Binary Addition, Flying- Adder(FA), Digital Frequency Converter(DFC)**

## I. Introduction

The Flying-Adder (FA)-based frequency synthesizer can be viewed as a Digital-to-Frequency Converter (DFC) for on-chip digital controllable frequency synthesis [1]. Due to its wide frequency range, instant response, and easy integration, the FA-DFC has been widely used in system-on-chip applications [5]. The FA-DFC consists of two units, a simple integer-N PLL providing the reference signals, and digital processing unit (mainly adder, multiplexer and D-flip-flop), capable of synthesizing a wide range of frequencies instantaneously. As its name states, the core of the FA-DFC is the function of addition which generates the control signals for the multiplexer in the FA-DFC to construct the final clock output. This addition operation is crucial for the creation of the output frequency, in that it determines the frequency resolution as well as the highest frequency synthesizable using this architecture [4]. High resolution in the FA-DFC clock output requires a large number of bits in the fractional part of the frequency control word. As a result, the adder/accumulator becomes the bottleneck of the FA-DFC because the higher the frequency resolution, the larger the adder/accumulator is required and hence an increased propagation delay results in the adder circuit due to carry propagation.

In conventional binary addition circuits using the digit set {0, 1}, there have been many published methods of reducing the delay of large adders by manipulating the way a carry-ripple is propagated and minimizing its occurrence as described in [7, 8, 9].One can conjecture that minimizing/eliminating the carry-ripple digit propagation would also in fact minimize/eliminate the respective contributing delay. As discussed in [6], redundant signed digit number systems can be used to implement fast addition circuits by taking advantage of the redundancy in the system. This redundancy can be used to create addition circuits that do not experience carry-ripples that, in the worst case, propagate through all bits of the addend and augend.

Previous research in the community has looked at increasing the performance in terms of jitter and demonstrated its use on a FPGA [10]. This paper focuses on incorporating an addition circuit based on RSBA into the FA-DFC on the transistor level. The focus is increasing the speed of the adders, thereby enabling the FA-DFC to achieve a higher frequency resolution. In Section II, we provide the details of the FA-DFC, introduce RSBA and present a new RSBA multiplexer. Section III explains the new implementation of the FA-DFC using RSBA. Section IV provides simulation results. We conclude in Section V.

## II. Architecture Background

### A. Flying-Adder Digital-to-Frequency Converter

The FA-DFC operates according to the concept of time average frequency [1]. The FA-DFC is able to periodically introduce an additional pro-longed cycle so that the desired frequency is the result of an average of the output created by the FA. Frequency synthesis is accomplished by using multiple phase delayed outputs generated from the VCO/PLL. Figure 1 depicts the FA-DFC architecture used in our design. Specifically a 24-bit accumulator would give the desired frequency resolution (24-bit FA-DFC model)

Eight VCO outputs are used to construct the final output clock. These VCO outputs are input into two 8:1 multiplexers. Since there are 8 inputs available in each path, this provides a total of 16 signals to construct the output.

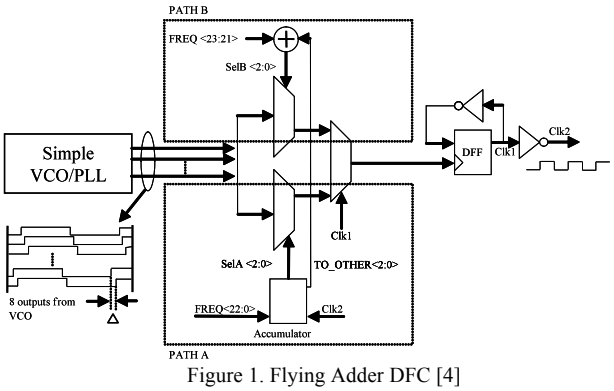


Figure 1. Flying Adder DFC [4]

This work is done in conjunction with funding from the SRC Corporation, task number 1620.001

The adjacent VCO output signals differ in phase by Δ, as shown in Equation (1).

$$\Delta = T_{Vco} / K \quad (1)$$

In Equation (1) $T_{Vco}$ is the period of the VCO output and $K$ is the number of output stages, in our case $K$=8. The FA-DFC transfer function is defined in Equation (2) as

$$T_s = 1/f_s = FREQ*\Delta \quad (2)$$

where $T_s$ is the desired output frequency period and *FREQ* is the frequency control word [5].

To generate the final output waveform, *CLK*2, the FA-DFC selects the appropriate VCO output signal at the time needed to trigger the flip-flop [2]. It is these VCO output waveforms that are input to the FA-DFC that make up the final output combined with PATHA and PATHB working together. These paths are synchronized by sending the address output in PATHA to the input in PATHB. A large accumulator is used in PATHA to accumulate the fractional part of the control word, *FREQ*. The most significant bits denoted as *FREQ*<23:20> represent the integer part while the remaining bits, <19:0> are the fractional part of *FREQ*. This accumulation is performed for accuracy and guarantees that all the frequencies in the operation range can be generated. The integer portion of *FREQ* is used as control inputs to the 8:1 multiplexers, which in turn select the appropriate signals from the VCO. When working together, the rising edge of the output signal is generated by PATHA while, conversely, the falling edge is generated by PATHB, or vice versa. The output CLK2 signal is created by the 2:1 multiplexer toggling between each path and thereby triggering the DFF [4].

### *B. Redundant Signed Binary Arithmetic*

Traditionally, digital logic designers are accustomed to binary base-2 systems with the digits {1, 0}. In RSBA arithmetic, a base-2 system is still used, however the digits $\{-1,0,1\}$ are incorporated into the system and denoted by $\{\overline{1},0,1\}$. This three digit set is then encoded as a two-bit string for implementation in digital circuitry. In the work described here, $\{\overline{1},0,1\}$ is encoded as {00, 01, 10} where the string 11 is not used. The particular encoding can have an effect on the resultant adder area, performance, and power characteristics as described in [6] and is dependent upon the standard cell library used in the overall circuit implementation. A chief difference of the RSBA system is that each digit now contains individual sign information instead of more common approaches employing a magnitude and overall sign such as the commonly used two's complement representation. Because the digit set is comprised of three values and the radix is 2, the resultant number system is redundant in that some quantities can be represented with more than one unique digit string. As an example, 5 can be represented as both $10\overline{1}\,\overline{1}$ and $1\,\overline{1}01$ as shown below in the radix polynomial expansions.

*Conventional Binary System:*

$$0101 = (0\times2^3) + (1\times2^2) + (0\times2^1) + (1\times2^0) = 5 \quad (3)$$

*Redundant Signed Binary System:*

$$10\,\overline{1}\,\overline{1} = (1\times2^3) + (0\times2^2) + (-1\times2^1) + (-1\times2^0) = 5 \quad (4)$$

Or

$$1\,\overline{1}01 = (1\times2^3) + (-1\times2^2) + (0\times2^1) + (1\times2^0) = 5 \quad (5)$$

This redundancy can be used to eliminate the propagation of carry digits during the addition of two operands *W*+X by transforming them into an equivalent addition of *Y*+*Z* [6]. The word *Y is* restricted to a digit string consisting only of $\{\overline{1},0\}$ and Z is constrained to a digit string consisting only of $\{0,1\}$. With this transformation, the final sum can be calculated and a carry propagation will never occur since the two cases that can possibly generate a carry $(1+1 \text{ or } \overline{1}+\overline{1})$ can never occur.

In order to determine the overall negative value of *Y* and the overall positive value of *Z*, the concept of a "borrow" word is used in order to form these values. The digits forming these values are found by satisfying Equation (6) where the subscript *i* represents the power of 2 in the radix polynomial expansion.

$$w_i + x_i = 2(z_{i+1} + b_{i+1}) + y_i - b_i \quad (6)$$

The digits $b_i$ are restricted to $\{\overline{1},0\}$ and form the borrow word *B* allowing for the values *Y* and *Z* to be formed subject to the digit set constraints explained above.

This allows carry-free addition to be performed in three steps:

1) Determine the borrow digits $b_{i+1}$ based on the $w_i$ and $x_i$ digits (note that $b_0$=0)
2) Determine the $z_{i+1}$ and $y_i$ digits based on $w_i$, $x_i$, and $b_i$ (note that $z_0$=0)
3) Combine each $z_i$ and $y_i$ through addition to form the final sum.

Each of steps 1) through 3) can be performed in parallel digit by digit, particularly step 3) where no carry propagation can occur since the cases of $z_i$+$w_i$ can never be $1+1$ or $\overline{1}+\overline{1}$ due to the digit set restrictions of *Z* and *Y*. Figure 2 illustrates the structure of an addition circuit based on the three steps described above. The subcircuits of Figure 1 depend on the signed digit encoding values and Table I where the values of the *B*, *Y*, and *Z* words are enumerated for the RSBA addition operation [6].

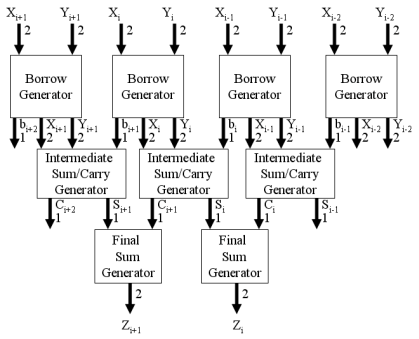


Figure 2. Fast Adder using SBA with borrow

Table I. Signed Binary Addition Table with borrow

| $w_i+x_i$ | $b_{i+1}$ | $b_i$ | $z_{i+1}$ | $y_i$ |
|---|---|---|---|---|
| $\bar{1}+\bar{1}$ | $\bar{1}$ | $\bar{1}$ | 0 | $\bar{1}$ |
| $\bar{1}+\bar{1}$ | $\bar{1}$ | 0 | 0 | 0 |
| $\bar{1}+0$ | $\bar{1}$ | $\bar{1}$ | 0 | 0 |
| $\bar{1}+0$ | $\bar{1}$ | 0 | 1 | $\bar{1}$ |
| 0+0 | 0 | $\bar{1}$ | 0 | $\bar{1}$ |
| 0+0 | 0 | 0 | 0 | 0 |
| $1+\bar{1}$ | $\bar{1}$ | $\bar{1}$ | 1 | $\bar{1}$ |
| $1+\bar{1}$ | $\bar{1}$ | 0 | 1 | 0 |
| 1+0 | 0 | $\bar{1}$ | 0 | 0 |
| 1+0 | 0 | 0 | 1 | $\bar{1}$ |
| 1+1 | 0 | $\bar{1}$ | 1 | $\bar{1}$ |
| 1+1 | 0 | 0 | 1 | 0 |

As an example, consider the addition of the two values $W = 110\bar{1}\,\bar{1}1010$ and $X = 11\,\bar{1}0110\,\bar{1}0$.

| | |
|---|---|
| $110\bar{1}\,\bar{1}1010$ | $W = 346$ |
| $11\bar{1}0110\bar{1}0$ | $X = 342$ |
| $00\bar{1}\,\bar{1}\,\bar{1}00\bar{1}00$ | $B = -228$ |
| $0\bar{1}0000\,\bar{1}0\,0$ | $Y = -132$ |
| $110\,0110\,10\,0$ | $Z = 820$ |
| $11\bar{1}0110\,000$ | $SUM = 688$ |

### C. Redundant Multiplexer

Although using RSBA produces fast adders, the caveat to utilizing RSBA is that the number system should be kept consistent throughout the design. Otherwise, converting from RSBA back to normal binary involves carry ripples in conversion arithmetic. To solve this problem, we designed a new RSBA-based multiplexer (an 8:1 one for the FA-DFC) for use with the adder. While a standard 2:1 multiplexer requires one select line, a RSBA-based 2:1 multiplexer requires two select lines (SEL_S for the sign bit and SEL_M for the magnitude bit). Likewise, for an RSBA 8:1 multiplexer, six select lines in total are required, three for SEL_S and three for SEL_M. Figure 3 depicts the difference between a conventional 8:1 multiplexer and its RSBA counterpart.

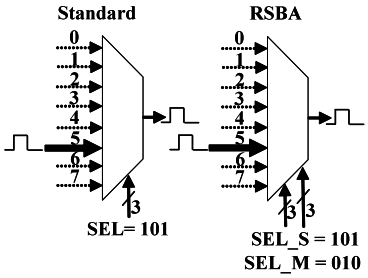


Figure 3. Depicting Multiplexer selects of Standard and RSBA designs

Figure 4 depicts the internal architecture of a RSBA-based 8:1 multiplexer. The RSBA-multiplexer consists of control logic and a tree of logic gates. The role of the control logic is to select the appropriate output path. When one path is selected, the control logic outputs a logical '1' to enable the respective AND gate. The tree of OR gates that follow propagate the signal to the output. The control logic composes of a table of values that maps the select lines to an appropriate output path selection.

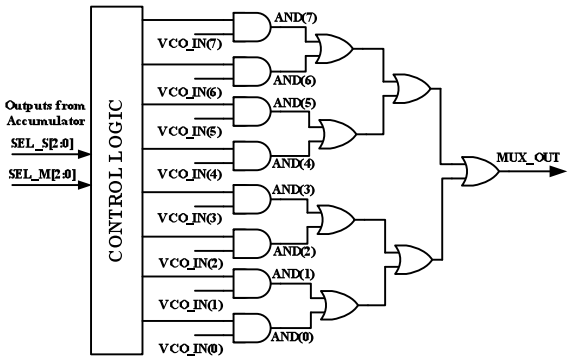


Figure 4. Theoretical RSBA 8:1 Multiplexer

## III. Implementation

A FA-DFC with the new adder/accumulator is described and validated utilizing VHDL and Cadence schematic simulations. Utilizing Table I, the adder/accumulator inside the FA Synthesizer was synthesized to a RSBA system. The original 8:1 multiplexers were replaced with the new RSBA-based 8:1 Multiplexers. Figure 5 depicts the complete FA-DFC using the RSBA addition circuitry. Since the RSBA system utilizes twice the number of bits as opposed to a conventional base-2 system for signed digit encoding, the frequency control word and intermediate select lines are also twice the size.

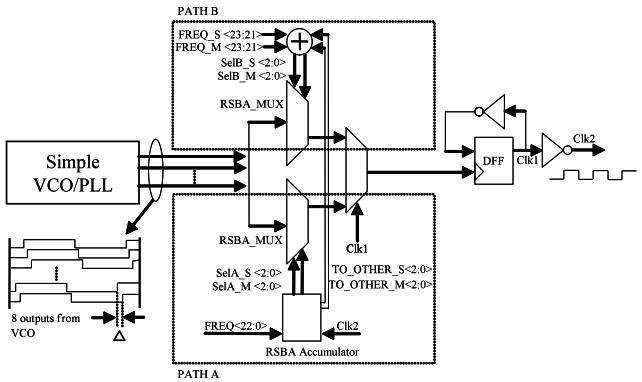

Figure 5. Flying-Adder Using RSBA System

## IV. Experimental Results

A conventional and RSBA-based adder/accumulators are synthesized using a standard cell library from IBM 0.13µm (CMRF8SF) process and the results reported from the Synopsys® Design Compiler® tool on the two designs are compared. We report in Table II and Table III the performance, area and power for various word sizes for the adder/accumulator block. The area was taken by using the area of a standard 1X drive strength NAND2 gate. The power consumption is reported from Design Compiler® based on a 1.2V power supply. In Table IV, we compare the report on the critical path, area and power of a conventional and new RSBA 8:1 Multiplexer. Synthesized results show that in the RSBA design, we obtain constant speed regardless of the word size as expected due to properties of RSBA. By contrast, for the conventional adder, as word size increases, speed decreases and area and power consumption increase. Specifically for a 24-bit adder, the RSBA has a 3.7 times performance improvement over the conventional one. As a result the bandwidth is also expected to increase since the adder is the speed bottleneck of the FA-DFC. Even if we account for 100% wire and routing delay, we still see almost a 2.0 times improvement in performance. The improved performance does come with increase in area (2 times) and power dissipation (3.5 times). A Cadence™ schematic simulation was conducted on the whole DFC and the experimental results show that the total power dissipation of the adder is approximately 0.355mW. This accounts for only (<10%) of the whole power consumption of the DFC (3.84 mW total from Cadence™). This means replacing the conventional adder in the DFC with RSBA-based results in an increase of only 25% of the total power consumption of the DFC. Therefore the increase in power is reasonable when a higher frequency resolution is desired. Note the power consumption reported by Synopsys® assumes 100% activity, which in practice will be much lower. This is confirmed by circuit-level Cadence simulation results: 1.2mA for the accumulator and 13mA for the complete DFC. The increase in area is less a concern for deep sub-micron or nano-meter technology as the devices sizes become smaller. In addition, the increase in area and power consumption in the RSBA multiplexer is negligible because the total power dissipation of the multiplexer only accounts for a small fraction of the total power dissipation of the synthesizer (<1%). Frequency resolution remains the same as the normal Flying Adder since it is solely dependent on the bit size of the adder used.

## V. Conclusion

This paper introduces an RSBA based adder/accumulator in the FA-DFC that resolves the performance bottleneck caused by the adder. The new RSBA Multiplexer is presented to allow for the full integration of the RSBA adder into the FA-DFC. The RSBA adder demonstrates superior speed performance regardless of the bit/word size, thus a very fine frequency resolution is achieved in the frequency synthesizer.

Table II. Adder/accumulator used in Flying Adder. Multiple bit size synthesized performance results as reported from Design Compiler.

| | **Conventional Adder/Accumulator FA** | | |
|---|---|---|---|
| **Adder Size** | Speed (ns) | Area (Units of NAND2) | Dynamic Power Consumption (mW) |
| **24-bit** | 0.81 | 517.25(324.25192.99) | 4.2842(3.6496,0.6346) |
| **32-bit** | 0.83 | 636.75(444.50,192.25) | 4.6412(3.8652,0.776) |
| **48-bit** | 1.1 | 1227.75(870.00,357.75) | 6.4727(5.3746,1.0981) |
| **64-bit** | 1.2 | 1491.25(1034.25,456.99) | 7.2895(6.1152,1.1744) |

Table III. Adder/accumulator used in Flying Adder. Multiple bit size synthesized performance results as reported from Design Compiler.

| | **NEW RSBA Based Adder/Accumulator FA** | | |
|---|---|---|---|
| **Adder Size** | Speed (ns) | Area (Units of NAND2) | Dynamic Power Consumption (mW) |
| **24-bit** | 0.22 | 1047.50(518.50,528.99) | 15.9265(13.7137,2.128) |
| **32-bit** | 0.22 | 1978.25(1257.25,720.99) | 25.8423(20.593,5.2496) |
| **48-bit** | 0.22 | 2897.743(1905.75,991.99) | 32.6018(24.7033,7.8985) |
| **64-bit** | 0.22 | 3861.75(2557.75,1303.99) | 42.1860(31.6046,10.5813) |

TABLE IV. Standard Multiplexer vs. RSBA Based Multiplexer

| | **8:1 Multiplexer Comparison** | | |
|---|---|---|---|
| **Type** | Critical Path Limit (ns) | Area (Units of NAND2) | Dynamic Power Consumption (uW) |
| **Standard** | 0.52 | 12.5(72,0) | 8.1187(5.4736,2.6450) |
| **RSBA** | 0.50 | 59.5(37.5,22) | 28.1325(18.7376,9.9349) |

## References

[1] L.Xiu, "The Concept of Time-Average-Frequency and Mathematical Analysis of Flying-Adder Frequency Synthesis Architecture," *IEEE Circuit and Systems Magazine*, pp.27-51, Third Quarter 2008.

[2] H. Mair and L. Xiu, "An Architecture of High-performance Frequency and Phase Synthesis," *IEEE J. Solid-State Circuits*, vol. 35, no. 6, pp. 835–846, Jun. 2000.

[3] L. Xiu, "A Flying-Adder PLL Technique Enabling Novel Approaches for Video/Graphic Applications," *IEEE Trans. on Consumer Electronic*, vol.54, no.2, pp. 591-599, May, 2008.

[4] L. Xiu and Z. You, "A Flying-Adder Architecture of Frequency and Phase Synthesis With Scalability," *IEEE Trans. Very Large Scale Integrat. (VLSI) Syst.*, vol. 10, no. 5, pp. 637–649, Oct. 2002.

[5] L. Xiu, "A 'Flying-Adder' On-Chip Frequency Generator for Complex SoC Environment," *IEEE Tran. On Circuits and Systems,* vol 54, Issue 12, pp. 1067 – 1071, Dec. 2007.
[6] M. Thornton, "A Signed Binary Addition Circuit Based on an Alternative class of Addition Tables," *Computers & Electrical Engineering*, vol.29, no.2, pp.303-315, March 2003.
[7] M.Lehman and N.Burla, "Skip Techniques for High-Speed Carry Propagation in Binary Arithmetic Circuits*," IRE Transactions on Electronic Computers*, vol.10, pp.691-698, December 1962.
[8] H.Ling, "High Speed Binary Adder", *IBM Journal of Research Development*, vol. 25, pp.156-166, May 1981.
[9] T.F. Ngai, M.J. Irwin and S.Rawat. Regular, "Area-Time Efficient Carry-Lookahead Adders," Journal of Parallel and Distributed Computing, vol. 3, pp. 92-105,1986.
[10] H. Gharaee, E. Tathesari, "A New High Resolution Frequency and Phase Synthesis Method based on Flying-Adder Architecture," *IEEE International Conference on Semiconductor Electronics,* pp. 520-523, Dec.1 2006.