Hindawi Publishing Corporation
Journal of Electrical and Computer Engineering
Volume 2011, Article ID 871385, 11 pages
doi:10.1155/2011/871385

*Research Article*

# An Interpolated Flying-Adder-Based Frequency Synthesizer

**Pao-Lung Chen and Chun-Chien Tsai**

*Department of Computer and Communication Engineering, National Kaoshiung First University of Science and Technology, No. 2, Jhuoyue Road, Nanzih District, Kaohsiung City 811, Taiwan*

Correspondence should be addressed to Pao-Lung Chen, plchen@nkfust.edu.tw

Received 8 June 2011; Revised 25 August 2011; Accepted 25 August 2011

Academic Editor: Jae-Yoon Sim

Copyright © 2011 P.-L. Chen and C.-C. Tsai. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This work presents an interpolated flying-adder- (FA-) based frequency synthesizer. The architecture of an interpolated FA, which uses an interpolated multiplexer (MUX) to replace the multiplexer in conventional flying adder, improves the cycle-to-cycle jitter and root-mean-square (RMS) jitter performance. A multiphase all-digital phase-locked loop (ADPLL) provides steady reference signals for the interpolated flying adder. This paper reveals implementation skills of a multiphase ADPLL, as well as an interpolated flying adder. In addition, analytical details of the jitter performance are derived. A test chip for the proposed interpolated FA-based frequency synthesizer was fabricated in a standard 0.18 $\mu$m CMOS technology, and the core area was 0.143 mm$^2$. The output frequency had a range of 33 MHz ~ 286 MHz at 1.8 V with peak-to-peak ($P_k$-$P_k$) jitter 215.2 ps at 286 MHz/1.8 V.

## 1. Introduction

The frequency synthesizer is a key component for numerous systems to generate a desired frequency for frequency conversion in digital communication or in system on chip (SoC) for signal synchronization [1]. Conventional approaches in [2–4] have utilized phase-locked loop (PLL) to generate different high-frequency outputs with a low-frequency crystal clock by setting the frequency control word that is widely used in the industry. While PLL provides flexible frequency synthesis, the loop parameters, such as damping factor and loop bandwidth, must be adjusted to minimize jitter and to ensure that each output frequency and frequency control word is stable. The loop bandwidth should be approximately 1/20 of the reference frequency. To reduce cost and enhance loop stability, an all-digital PLL (ADPLL) in [5] utilized a digital loop filter with a seven-cycle lock time. However, the output frequency range is less than 100 MHz. With a fractional-N synthesis technique [6], finer frequency control can be achieved. But the system typically has a narrow bandwidth that has tight control on loop parameters.

Direct digital synthesis (DDS) in [7] uses memory and logic to generate the desired output frequency. It normally consists of a phase accumulator, an ROM lookup table and a linear digital-to-analog converter (DAC). The performance of a DAC is the bottleneck of this technique. In recent years, flying-adder frequency synthesis has provided a new way of generating frequency on chip [8–16]. The flying-adder-based frequency synthesizer solves problem that cannot be dealt easily with conventional PLL-based or DDS-based frequency synthesizer. The flying-adder frequency synthesizer is also called a direct digital period synthesizer (DDPS) in [17]. The flying-adder-based frequency synthesizer has been applied in many commercial chips such as the triple DAC graphics digitizer, digital speaker, LCD monitors, HDTV chips, and NTSC video decoders [11–13].

The basic structure of a flying adder is shown in Figure 1. The structure consists of an N equally spaced phases, a multiplexer (MUX), a D flip-flop (DFF), and an $n$-bit accumulator. The $n$-bit accumulator is composed of an $n$-bit adder and an $n$-bit register. The $n$-bit register is divided into the integer (Int) and fractional (Fract) parts. The frequency control word (FREQ) is inputted to the flying adder. The Fract is used for accumulating the fractional part. The multiplexer selects one of the input signals from the N equally spaced phases according to Int value. The output signal from the multiplexer triggers a D flip-flop. The function of D flip-flop is similar to a frequency divider

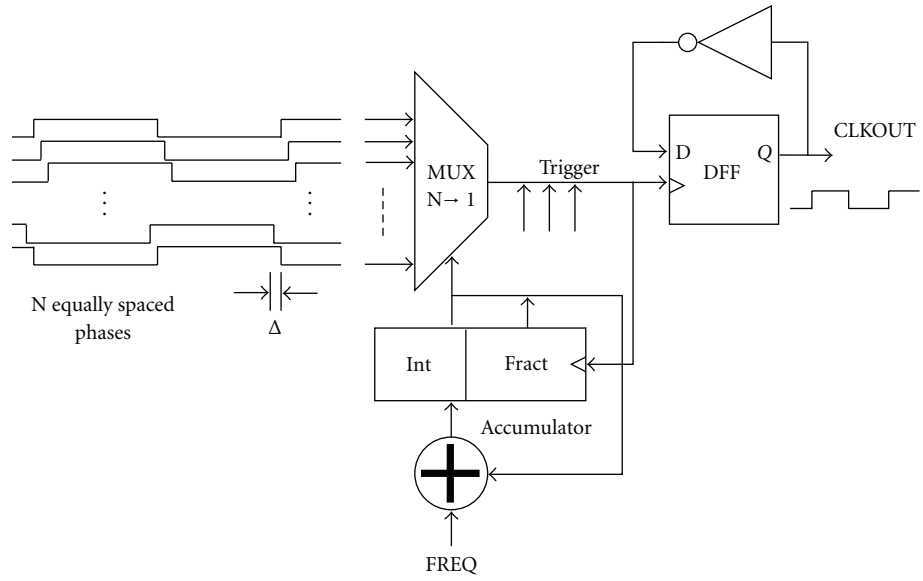



Figure 1: Block diagram of conventional flying adder [8].

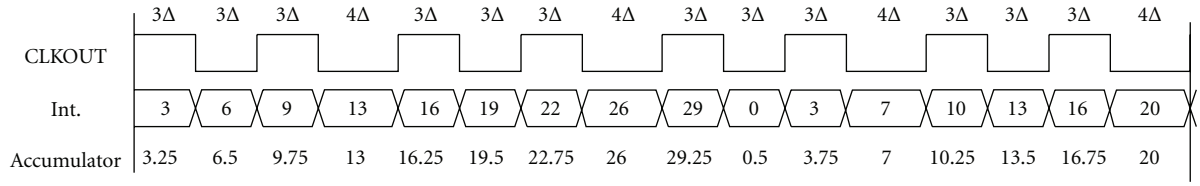



Figure 2: Numerical operation of [8]'s flying adder.

by-2 circuit. Meanwhile, the $n$-bit register for the MUX address is updated. Let us assume that FREQ is a 10-bit frequency control word with 5 bits for Int and 5 bits for Fract, respectively. The timing resolution of a 32 equally spaced phases with "Δ" is 0.2 ns. If a 769 MHz (1.3 ns) signal is the desired output, then the FREQ[9 : 0] can be calculated as follows:

$$\text{FREQ}[9:0] \cdot 2 \cdot 0.2\,\text{ns} = 1.3\,\text{ns}. \quad (1)$$

The FREQ[9 : 0] is equal to $3.25 = 00011.01000_{\text{b}}$. As illustrated in this numerical example, the clock cycle at that particular time is 0.2 ns (i.e., Δ) longer than a normal cycle whenever the value of fractional part is propagated to the integer part. This is called the cycle prolong in a flying-adder frequency synthesizer.

To explain clearly the cycle prolong of the flying-adder-based frequency synthesizer of [8], we assume that the FREQ = Int + Fract = 3.25 as in (1). The Int part is equal to 3, and the fractional part is 0.25.

The numerical operation of Figure 1's flying adder is illustrated in Figure 2. Let us assume that the phase resolution of N equally spaced signals is Δ. Whenever there is a rising edge from the MUX output, the value in the accumulator is accumulated with FREQ. The values of the accumulator are 3.25, 6.5, 9.75, and 13. The values of Fract part are 0.25, 0.5, 0.75, and 0. The generated trigger intervals are 3Δ, 3Δ, 3Δ, and 4Δ. The trigger interval between 3Δ and 4Δ is 1Δ because the accumulation of Fract part is propagated into the Int part. Therefore, the cycle-to-cycle jitter is 1Δ when the cycle is prolonged, as indicated in [8, 11, 14, 15]. The poor cycle-to-cycle jitter degrades the output clock's performance. Reference [11] solved the prolonged cycle using a post divider. However, the generated frequency is divided down. We propose an interpolated flying-adder structure to reduce jitter which is caused by the cycle prolong.

The rest of this paper is arranged as follows: Section 2 describes the proposed interpolated flying adder. The implementation of a frequency synthesizer using an interpolated flying adder is described in Section 3. In Section 4, the jitter analysis of proposed interpolated flying adder is provided. The experimental result is in Section 5. Finally, Section 6 presents a summary and conclusions.

## 2. Proposed Interpolated Flying Adder

We propose an interpolated flying-adder structure, which can reduce the cycle-to-cycle jitter to one-half, to solve the cycle prolong. The interpolated flying-adder structure is an improvement of [9] two path's flying-adder structure. The two-path flying adder is interlocked through two AND gates and XOR/XNOR gates between path A and path B as shown in Figure 3.

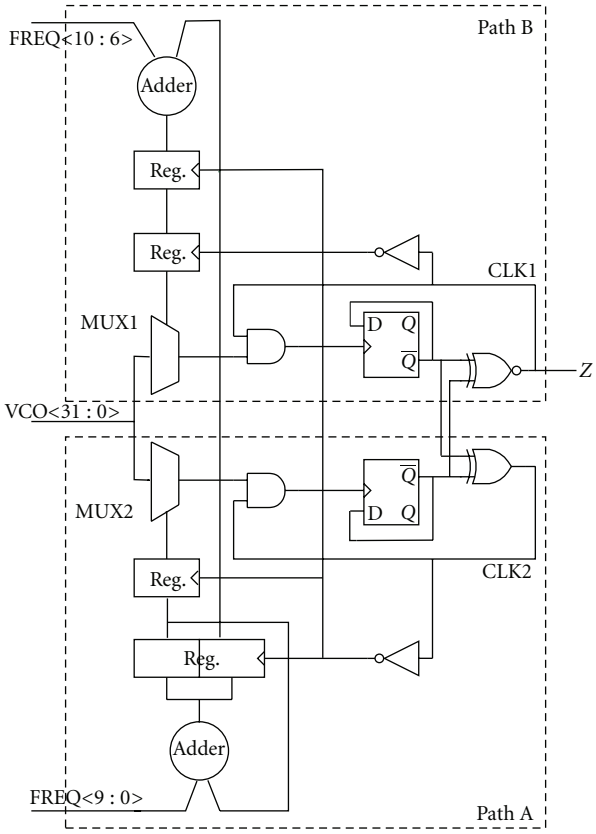



Figure 3: Two-path flying-adder frequency synthesizer [9].

The two AND gates and the feedback self-clocking of the registers ensure that at any given time, there is only one path switched on and one path switched off. The two-path flying-adder makes the 32 VCO ticks look like 64 ticks [19]. utilized an all-digital PLL with delay chain for generation of the 32 VCO ticks for two-path flying-adder frequency synthesizer. A cell-based implementation of two-path flying adder with dual resolution is indicated in [20]. However, the problem of cycle prolong is still unsolved in both one-path and two-path flying adders. Figure 4 is the block diagram of proposed interpolated flying adder. We modify the original two-path flying-adder structure. A digital interpolator is applied to interpolate the output from path A and path B. Both paths are simultaneously switched on in interpolated flying adder which is different from the two-path operation.

Path A is similar to path B. However, the difference is the bit width of accumulator. The input signal of Reg. A is the integer (Int) part of accumulator. The value of Reg. B is accumulated with the value in Reg. A and input phase control. The phase control is used to control the phase relationship between path A and path B. The output signals of path A and path B are interpolated by an interpolator. The interpolator receives two signals, path A_out and path B_out, to produce third output signal, CLKOUT. The rising edge of the out signal is in the midpoint of path A_out and path B_out as illustrated in Figure 5.

When FREQ = 3.25 is the same as in Figure 2 for the flying adder of [8], the PHASE = 3 is equal to the integer part of FREQ. Figure 6 shows the numerical operation of the proposed interpolated flying adder. The Int part value of the accumulator is stored in Reg. A. The Reg. A's value will be accumulated with phase into Reg. B triggered by the output MUX B. The output of path A and path B will be interpolated at the output of interpolator. Then the interpolated intervals are 3Δ, 3Δ, 3.5Δ, and 3.5Δ, as calculated in Figure 6. The trigger interval between 3Δ and 3.5Δ is 0.5Δ. Therefore, the cycle-to-cycle jitter of the proposed interpolated flying adder is improved 50% as compared with the flying adder in Figure 2.

## 3. Implementation of Interpolated Flying-Adder-Based Frequency Synthesizer

The interpolated flying-adder-based frequency synthesizer consists of two subsystems: one is the multiphase all-digital phase lock loop (ADPLL) to generate 32 equally spaced phases; the second subsystem is the interpolated flying-adder as described in Figure 4.

The operation of the system is based on the FREQ < 28 : 0 > which is similar to [8]. The FREQ < 28 : 0 > is the input frequency control word. The 5 MSB bits are used directly for selecting the MUX 32 inputs. The 24 LSB bits are the fractional part. Let us assume that the output frequency of multiphase ADPLL is running at 156.25 MHz (6.4 ns).

The time delay between the two adjacent outputs is 6.4 ns/32 = 0.2 ns. If a 161.29 MHz (6.2 ns) is the desired output, then the FREQ < 28 : 0 > can be calculated as follows: FREQ < 28 : 0 > ·2 · 0.2 ns = 6.2 ns $\Longrightarrow$ FREQ < 28 : 0 > = 15.5 = 01111.1000 0000 0000 0000 0000 $0000_2$.

*3.1. Multiphase All-Digital Phase Locked Loop.* The function of the multiphase all-digital phase locked loop (ADPLL) is to generate 32 equally spaced phases for the proposed interpolated flying-adder-based frequency synthesizer. The block diagram of multiphase ADPLL is indicated in Figure 7.

The multiphase ADPLL consists of two major function units. The first function unit is the timing control unit and SAR, and the second part is the digital-to-voltage converter (DVC) and multiphase voltage-controlled oscillator (VCO). The timing control unit performs timing selection of reference frequency as well as output-generated frequency. Successive approximation register (SAR) control is the key unit for frequency searching. The major advantages of the SAR unit are a compact structure as well as a regular layout [21].

The digital-to-voltage converter (DVC) generates the Vtune signal to control VCO, and detailed DVC can be found in [22]. The VCO has 32 NAND gates that are illustrated in Figure 8. The VCO has 16 stages, and each stage has two NAND gates. The output of each NAND gate is connected to the next stage, and the feedback is connected to the input of the same stage's NAND gate [11]. The structure of the VCO is highly regular. The output frequency of VCO ranges

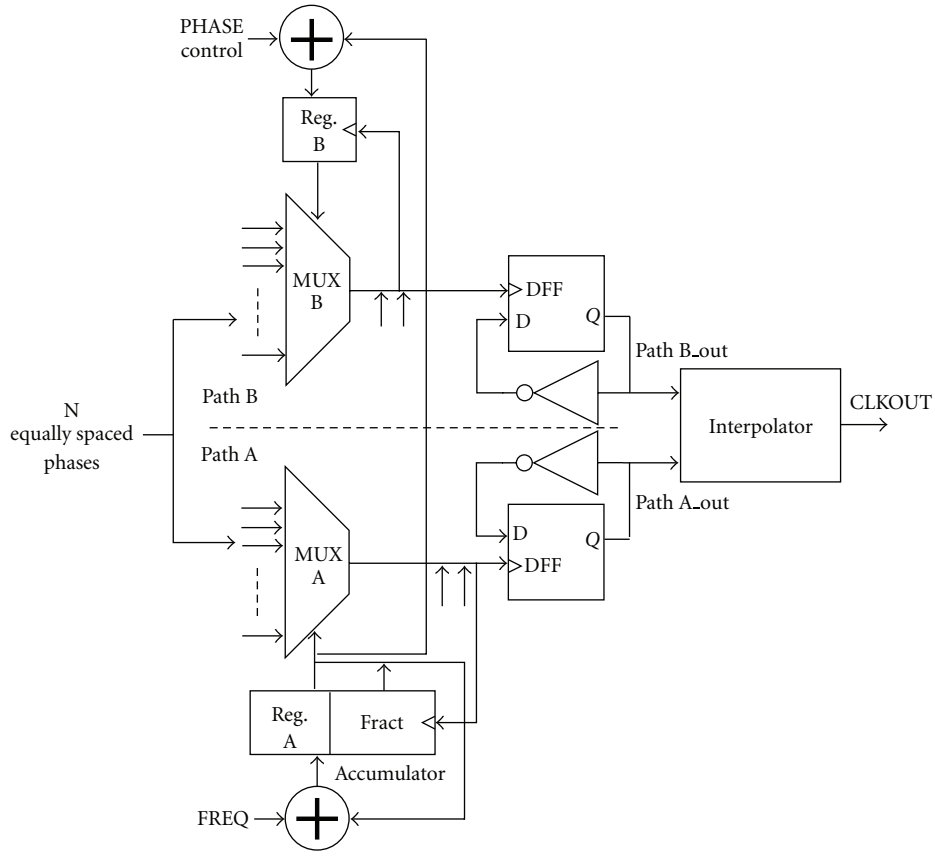



Figure 4: Block diagram of proposed interpolated flying adder.

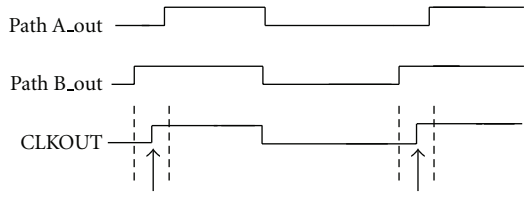



Figure 5: Function of the interpolator.

from 35 MHz to 235 MHz as shown in Figure 9. The output frequency of the multiphase ADPLL can be represented as

$$f_{\text{ADPLL}} = f_{\text{Ref}} \cdot \frac{N}{2}, \tag{2}$$

where $f_{\text{ADPLL}}$ is the output frequency and $f_{\text{Ref}}$ is the input reference frequency. The $N$ is controlled by $m_0 \sim m_5$. To generate 50% duty cycle of the output frequency, the VCO output frequency passes a divide-by-2 circuit. Therefore, the output frequency of a multiphase ADPLL is $N/2$ times of the reference frequency.

*3.2. Interpolated Flying Adder.* The subsystem of an interpolated flying adder consists of path A, path B, and an interpolator as shown in Figure 4. Each path has one 32-to-1 MUX, accumulator, flip-flop, and an inverter. The bit width of accumulator in the bottom is 29 bits, and the bit width of upper accumulator is 5 bits. The core element of an accumulator is the adder design. For simplicity of the adder design and taking advantage of signal synchronization, we adopted a modified carry-select adder with linear increasing the length of each subadder. The first 2 bits are a simple ripple adder, and followed by a carry-save adder with the length of 2 bits, 3 bits, 4 bits, 5 bits, 6 bits, and 7 bits. Therefore, the total length of the adder is 29 bits as shown in Figure 10.

The MUX design is a key issue and the implementation of an interpolator. The 32-to-1 MUX can be constructed by the tree structure of 2-to-1 MUX [24]. However, this tree structure creates delay mismatch as indicated in Figure 11. $T_S$ and $T_D$ denote the delays for control inputs (CS0 ~ CS4) to the MUX and for the data, respectively. The data have different delay phases to the output depending on their control inputs. If the control input only CS0 changes, the delay is $T_S + 4T_D$. In contrast, the delay is $T_S$ when only CS4 changes. The delay mismatch creates deterministic jitter.

To simplify the MUX design, we use a parallel transmission gate to reduce the switching delay as well as signal delay. To reduce the loading of the parallel-connected transmission, we combine 16 parallel transmission gates in one group. Then, we used a 2-to-1 MUX to select the signal. Figure 12 shows the implemented structure of a 32-to-1 MUX. Furthermore, we simplified the encoder circuit design, as shown in Figure 13. We divide the encoder circuit into 16 groups. Each group has one P-MOS load and four NMOSs.

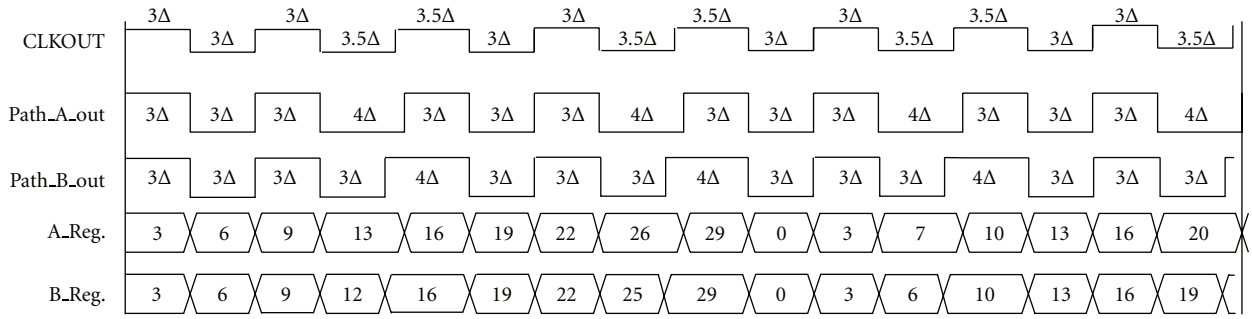



Figure 6: Numerical operation of proposed interpolated flying adder.

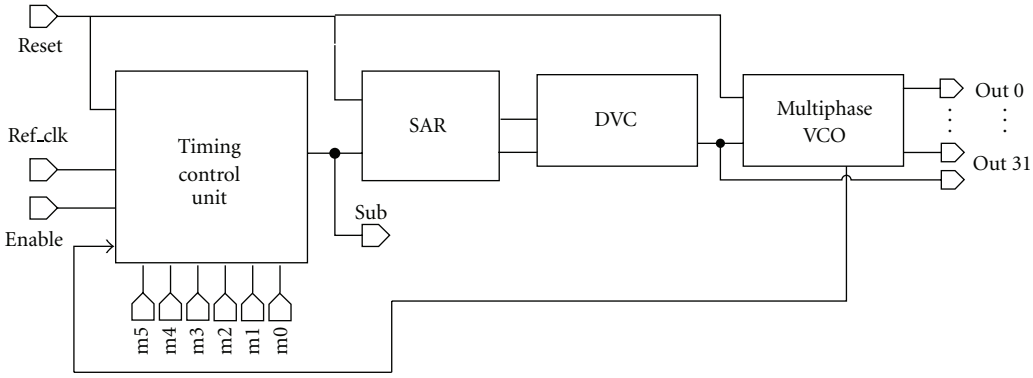



Figure 7: Block diagram of multiphase ADPLL.

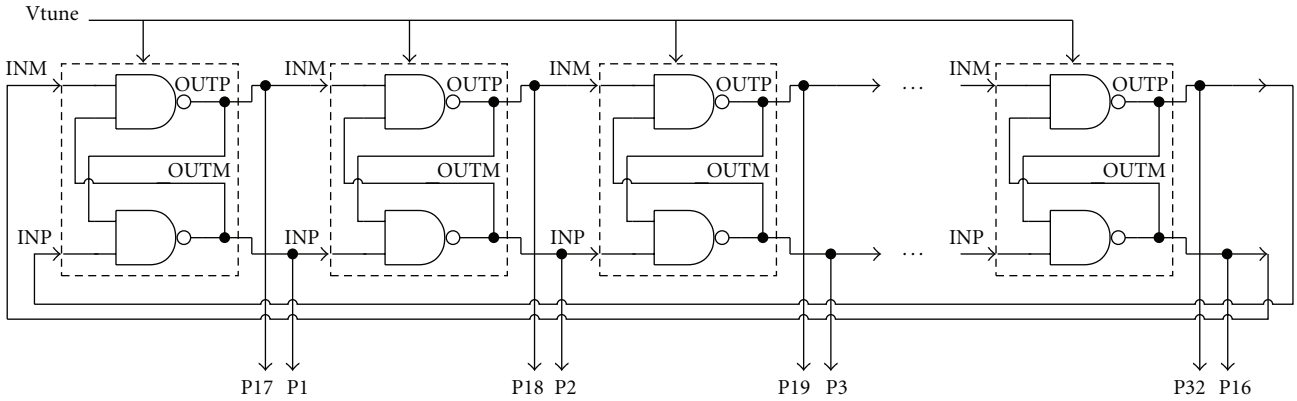



Figure 8: Block diagram of multiphase VCO.

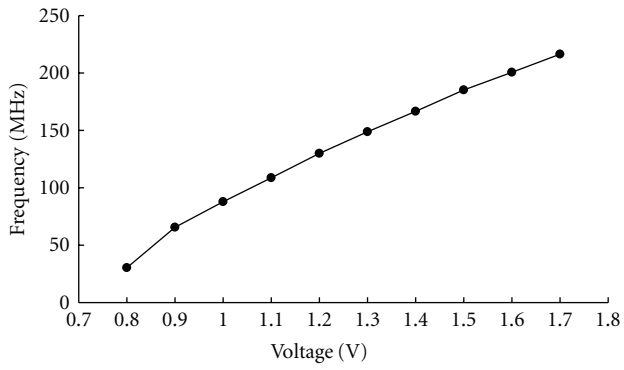



Figure 9: VCO output frequency.

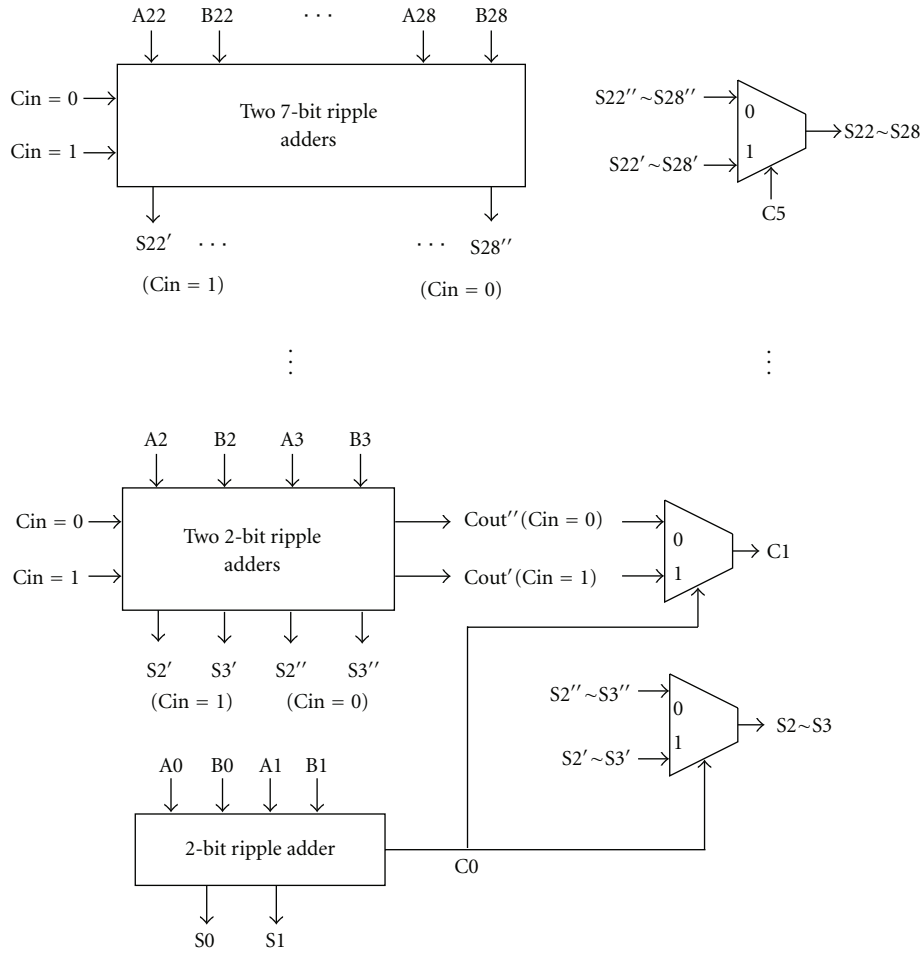



Figure 10: A modified 29-bit carry-select adder.

A digital interpolator in [18] has been used to adjust the delay that is smaller than the propagation delay of inverter in the delay-locked loop (DLL) as indicated in Figure 14(a). The digital interpolator can provide the function of interpolating two input rising edges as shown in Figure 14(b). The two input signals path A_out and path B_out are interpolated by the interpolator to achieve the out signal. Therefore, the interpolator performs the function in Figure 5.

## 4. Jitter Analysis of Interpolated Flying Adder

The basic architecture of interpolated flying adder is discussed in Section 2. A simple numerical example involving FREQ = 3.25 is illustrated in Figure 6. The interpolated flying adder can reduce the cycle-to-cycle jitter from 1Δ to Δ/2 as explained in Figures 2 and 6. However, more detailed analysis should be provided to compare the difference between the flying adder and interpolated flying adder. For the flying adder, jitter performance analysis has been intensively investigated in [8, 25, 26].

If a frequency $f = 1/T$ must be generated by the interpolated flying-adder-based frequency synthesizer and the multiphase signals are equally spaced with Δ resolution, then the frequency control word FREQ can be calculated as

$$T = \text{FREQ} \cdot \Delta, \tag{3}$$

$$\text{FREQ} = \frac{T}{\Delta} = I + r, \tag{4}$$

where the $I$ is the integer part and $r$ is the fractional part. Two types of cycles, short cycle $T_S$, and long cycle $T_L$ are in the synthesized signal. The short cycle $T_S$ and long cycle $T_L$ are represented as

$$\begin{aligned} T_S &= I \cdot \Delta, \\ T_L &= \left(I + \frac{1}{2}\right) \cdot \Delta. \end{aligned} \tag{5}$$

The peak-to-peak jitter can be shown as

$$J_{P_k\text{-}P_k} = T_L - T_S = \frac{\Delta}{2}. \tag{6}$$

The possibility of a long cycle and a short cycle can be divided into two cases: $0 < r < 0.5$ and $0.5 \leqq r < 1$.

Table 1: Jitter performance comparison.

| | Interpolated flying adder | Reference [8]'s flying adder |
|---|---|---|
| $J_{\text{mean}}$ | 0 | 0 |
| $J_{\text{rms}}$ (maximum) | $\Delta/4$ | $\Delta/2$ |
| $J_{P_k\text{-}P_k}$ (maximum) | $\Delta/2$ | $\Delta$ |

Table 2: Measured results versus jitter analysis.

| Items | Measured results | | |
|---|---|---|---|
| Synthesized output frequency | 33 MHz | 120 MHz | 286 MHz |
| Power consumption (includes 28 I/O pads) | 16.2 mW | 21.3 mW | 28.2 mW |
| Frequency, cycle time ($\Delta$), and peak-to-peak jitter of multiphase ADPLL | 42.7 MHz, $\Delta$ = 732 ps, 213 ps | 171 MHz, $\Delta$ = 183 ps, 150.3 ps | 171 MHz $\Delta$ = 183 ps, 150.3 ps |
| Peak-to-peak jitter | 435.6 ps | 248.7 ps | 215.2 ps |
| Peak-to-peak jitter, jitter of multiphase ADPLL | 222.6 ps < $\Delta/2$ | 98.4 ps ~ $\Delta/2$ | 64.9 ps < $\Delta/2$ |
| RMS jitter | 62.6 ps < $\Delta/4$ | 41.5 ps < $\Delta/4$ | 40.2 ps < $\Delta/4$ |

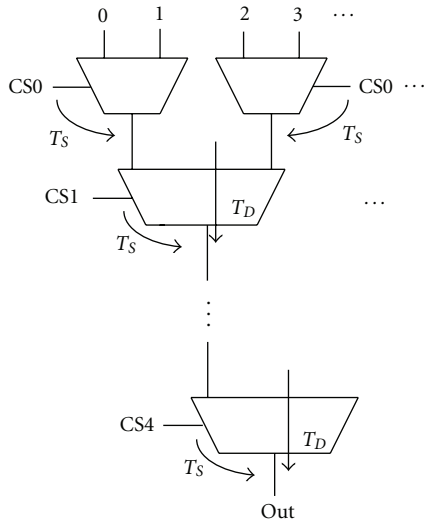



Figure 11: Delay mismatch in 32-to-1 MUX based on tree structure.

*Case 1* ($0 < r < 0.5$). The possibility of a long cycle and a short cycle is

$$\begin{aligned} P_L &= 2r, \\ P_S &= 1 - P_L = 1 - 2r. \end{aligned} \tag{7}$$

Then, the synthesized average cycle time is calculated as

$$\begin{aligned} T_{\text{avg}} &= (P_L \cdot T_L) + (P_S \cdot T_S) \\ &= (2r) \cdot \left(I + \frac{1}{2}\right)\Delta + (1 - 2r) \cdot (I)\Delta \\ &= (I + r) \cdot \Delta \end{aligned} \tag{8}$$

and is the same as (4). The mean jitter can be derived as

$$\begin{aligned} J_{\text{mean}} &= P_L(T_L - T) + P_S(T_S - T) \\ &= 2r \cdot \left[\left(I + \frac{1}{2}\right) \cdot \Delta - (I + r) \cdot \Delta\right] \\ &\quad + (1 - 2r) \cdot [I \cdot \Delta - (I + r) \cdot \Delta] \\ &= 0. \end{aligned} \tag{9}$$

Finally, the root-mean-square (RMS) jitter can be shown as

$$\begin{aligned} J_{\text{rms}} &= \sqrt{P_L(T_L - T)^2 + P_S(T_S - T)^2} \\ &= \sqrt{2r \cdot \mathcal{A} + (1 - 2r) \cdot [I \cdot \Delta - (I + r) \cdot \Delta]^2} \\ &= \Delta\sqrt{\frac{r}{2} - r^2}. \end{aligned} \tag{10}$$

We donate that $[(I + (1/2)) \cdot \Delta - (I + r) \cdot \Delta]^2 = \mathcal{A}$.

When $r = 1/4$, the $J_{\text{rms}}$ has maximum jitter value $\Delta/4$ and is one-half of the conventional flying adder of [8].

*Case 2* ($0.5 \leqq r < 1$). The possibility of a long cycle and a short cycle is

$$\begin{aligned} P_L &= 2r - 1, \\ P_S &= 1 - P_L = 2 - 2r \end{aligned} \tag{11}$$

and is different from $0 < r < 0.5$. However, the $T_{\text{avg}} = (I + r)\,\Delta$ and $J_{\text{mean}} = 0$ are the same as in Case 1.

The rms jitter can also be derived as

$$\begin{aligned} &J_{\text{rms}} \\ &= \sqrt{P_L(T_L - T)^2 + P_S(T_S - T)^2} \\ &= \sqrt{(2r - 1) \cdot [(I + 1) \cdot \Delta - (I + r) \cdot \Delta]^2 + (2 - 2r) \cdot \mathcal{A}} \\ &= \Delta\sqrt{-r^2 + \frac{3}{2}r - \frac{1}{2}}. \end{aligned} \tag{12}$$

When $r = 3/4$, the $J_{\text{rms}}$ has maximum jitter value $\Delta/4$ and is also one-half of the flying adder of [8].

Table 1 is the summary of the jitter performance comparison between the proposed interpolated flying adder and conventional flying adder [8]. Both of the proposed interpolated flying adder and the flying adder of [8] have

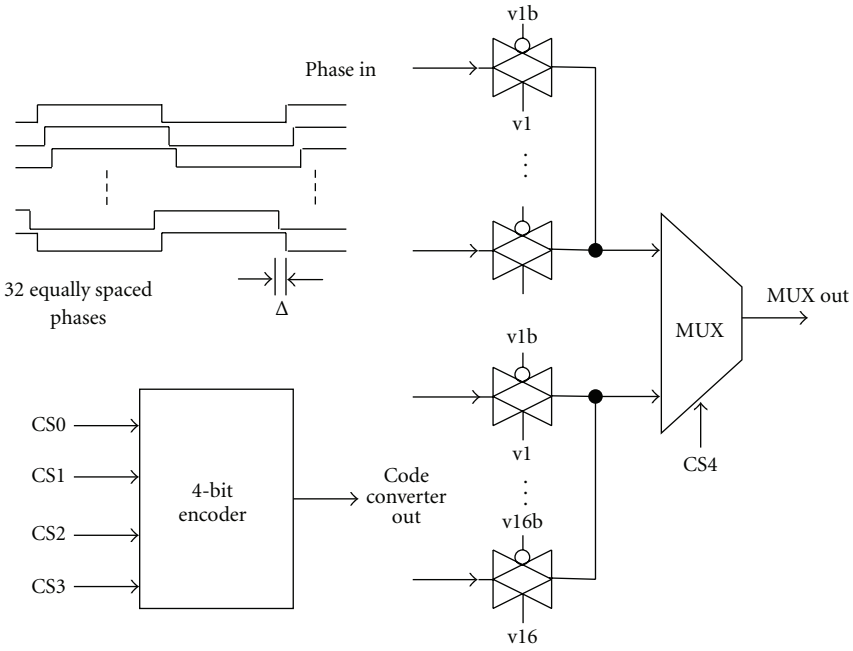



Figure 12: Transmission gate-based 32-to-1 MUX.

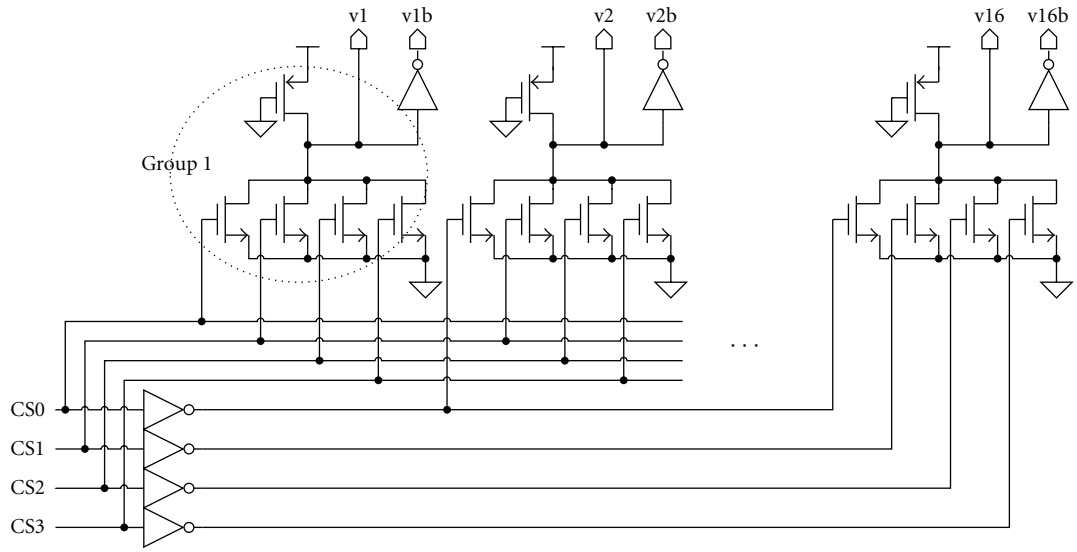



Figure 13: Simplified 4-bit encoder circuit.

the same average cycle time ($T_{\text{avg}}$) and mean jitter ($J_{\text{mean}}$). However, the RMS jitter and peak-to-peak jitter of the interpolated flying adder were improved to one-half of [8]'s flying adder.

## 5. Experimental Results

The test chip was fabricated in a standard 0.18 $\mu$m CMOS process. Figure 15 displays the microphotograph of the proposed interpolated flying adder. It includes two subsystems. The first subsystem is the multiphase ADPLL. The area of multiphase ADPLL is 283 $\mu$m $\times$ 132 $\mu$m. The reference clock of multiphase ADPLL input is 10 MHz. The output frequency of multiphase ADPLL is 171 MHz when the multiplication factor is 34 $(m_5, m_4, \ldots, m_0)$. Figure 16 shows the measured frequency of multiphase ADPLL at 171 MHz with 150.3 ps peak-to-peak jitter.

The other subsystem is the interpolated flying adder whose area is 295 $\mu$m $\times$ 353 $\mu$m as shown in Figure 15. When the frequency control word FREQ $< 28 : 0 >$ is 22.75 and the frequency of multiphase ADPLL is 171 MHz, the measured interpolated flying-adder output frequency is 120 MHz as indicated in Figure 17(a). The power consumption is 21.3 mW including 28 I/O pads. The peak-to-peak jitter is

Table 3: Performance comparison.

| | Reference [8]'s FA frequency synthesizer | Reference [19]'s two-path FA synthesizer | Reference [20]'s two-path FA synthesizer | Reference [23]'s two-path ADPLL | This work |
|---|---|---|---|---|---|
| Process | 0.6 $\mu$m, 3.3 V | 0.18 $\mu$m, 1.8 V | 0.18 $\mu$m, 3.3 V | 0.13 $\mu$m, 1.3 V | 0.18 $\mu$m, 1.8 V |
| Area (flying-adder) | 1350 $\mu$m $\times$ 1260 $\mu$m | 400 $\mu$m $\times$ 400 $\mu$m | 690 $\mu$m $\times$ 630 $\mu$m | 300 $\mu$m $\times$ 300 $\mu$m | 295 $\mu$m $\times$ 353 $\mu$m |
| Output frequency range | 57.27 MHz ~ 130 MHz | 39.38 MHz ~ 226 MHz | 62 KHz ~ 62.5 MHz | 10 MHz ~ 500 MHz | 33 MHz ~ 286 MHz |
| Accumulator width | 32 bits | 11 bits | 11 bits | | 29 bits |
| Power consumption (multiphase PLL) | ~40 mW | 3.6 mW | 32.4 $\mu$W | | 28.2 mW (I/O pads: ~14 mW, core:14.2 mW) |
| Power consumption (flying adder) | ~150 mW | | | | |
| Peak-to-peak jitter | 1132 ps at 120.05 MHz | 130 ps at 187.5 MHz | 410 ps | 288 ps at 191.4 MHz | 215.2 ps at 286 MHz |
| RMS jitter | 165 ps at 120.05 MHz | 50 ps at 187.5 MHz | | 39 ps at 191.4 MHz | 40.2 ps at 286 MHz |
| FoM(GHz/W) | 0.63 | 17.8 | | | 18.2 |

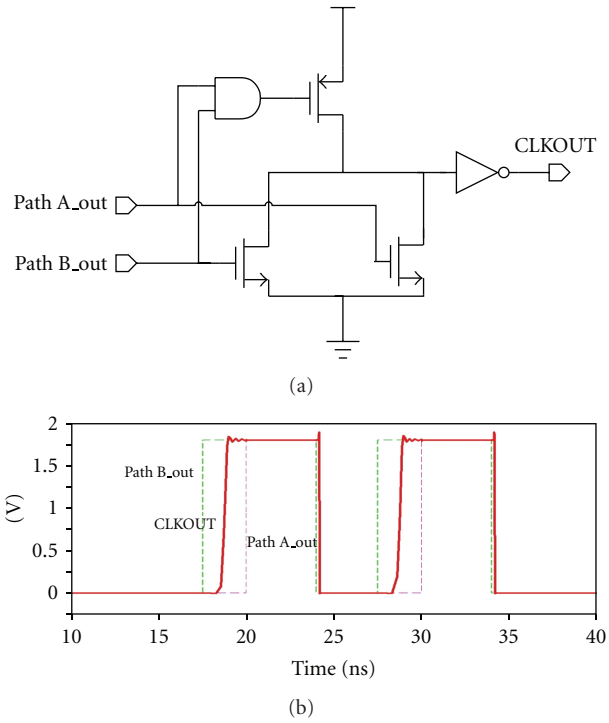



Figure 14: (a) Digital interpolator [18], (b) Hspice simulation.

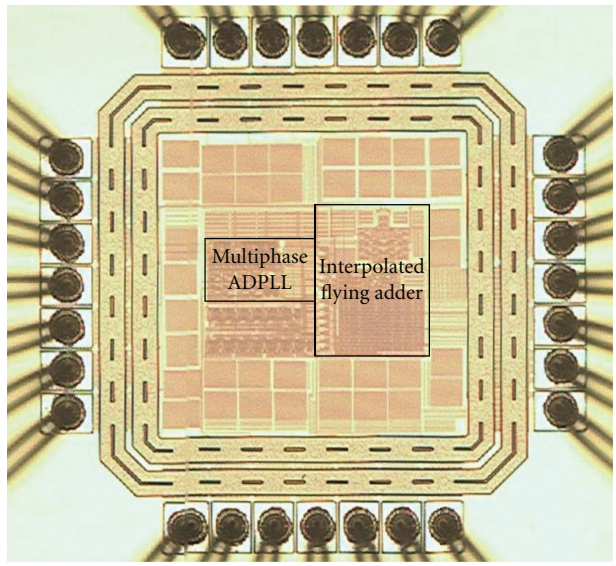



Figure 15: Microphotograph of the proposed all-digital frequency synthesizer using an interpolated flying adder.

249 ps, and the RMS jitter is 41.4 ps. Figure 17(b) is the measured result of interpolated flying adder at 286 MHz with 215.2 ps peak-to-peak jitter and 40.2 ps RMS jitter when the frequency control word FREQ < 28 : 0 > is 9.57. Only when the frequency of multiphase ADPLL is 42.7 MHz with 213 ps peak-to-peak jitter, the measured result of minimum output frequency is 33 MHz. Figure 18 shows that the output peak-to-peak jitter is 435 ps, and the RMS jitter is 60.5 ps.

The peak-to-peak jitter 150.3 ps of the multiphase signals will directly contribute jitter to output frequency as indicated in [17]. The output jitter caused by the interpolated flying adder is 98.4 ps and 64.9 ps at 120 MHz and 286 MHz, respectively. The RMS jitter is roughly equal to $\Delta/4 = 41$ ps in (12). When the output frequency is 33 MHz with multiphase ADPLL's cycle time ($\Delta$) 732 ps, the jitter caused by the interpolated flying adder is 222.6 ps. Table 2 is the jitter measurements as compared with jitter analysis. Therefore, the measurement of peak-to-peak jitter and RMS jitter verifies the jitter analysis in Section 4.

To evaluate the performance of flying-adder-based frequency synthesizer, an easy measured and calculated figure

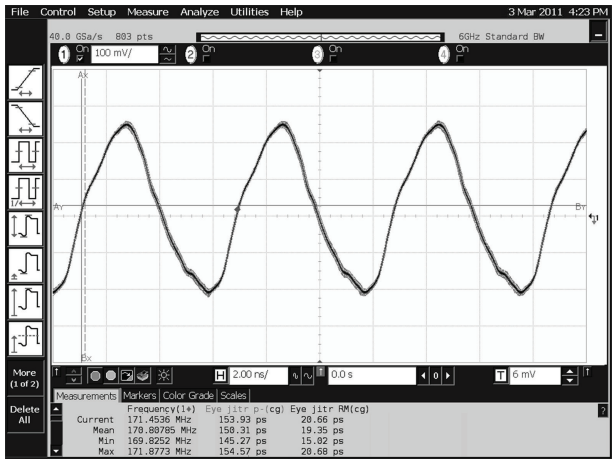
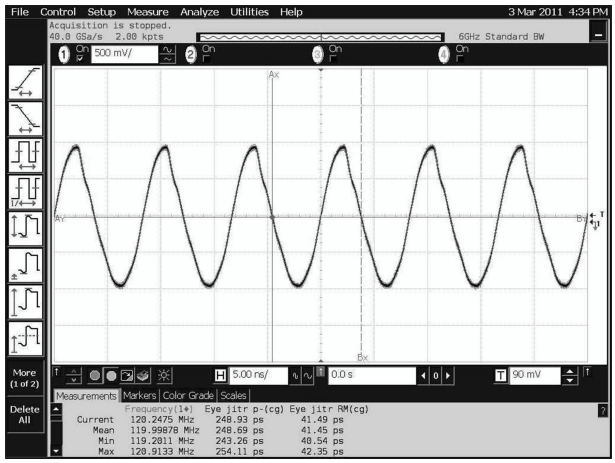
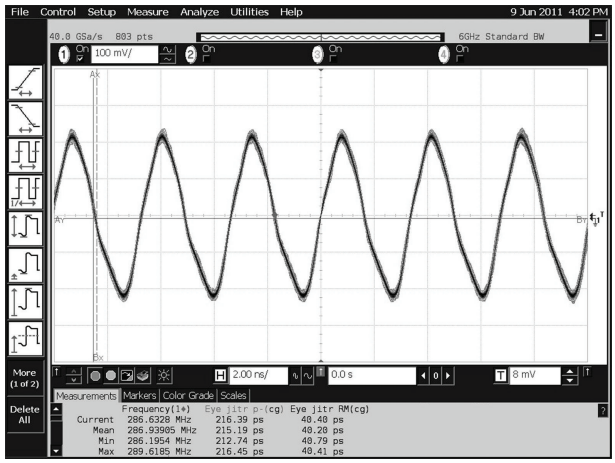



Figure 16: Measured results of multiphase ADPLL's output at 171 MHz with $P_k$-$P_k$ = 150.3 ps.




(a)




(b)

Figure 17: Measured results of interpolated flying adder (multiphase ADPLL at 171 MHz), (a) Output at 120 MHz with $P_k - P_k$ = 248.7 ps, (b) Output at 286 MHz with $P_k$-$P_k$ = 215.2 ps.

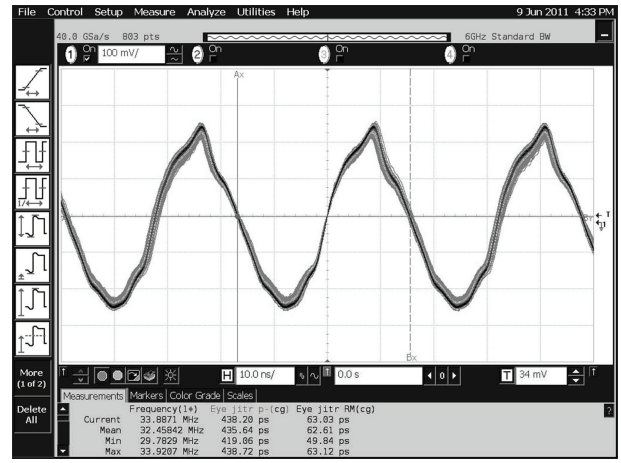



Figure 18: Measured results of interpolated flying-adder output at 33 MHz with $P_k$-$P_k$ = 435.6 ps (multiphase ADPLL at 42.7 MHz).

of merit (FoM) must be defined from a combination of performance parameters. The power efficiency FoM including the effective number of bit (ENOB) is defined in [7]. The bit width of accumulator is important to frequency tuning resolution as indicated in [8]. Similarly, we define the power efficiency FoM related with normalized bit width of accumulator as follows:

$$\text{FoM} = \frac{[\text{Output frequency (GHz)} \cdot \text{ normalized bit width}]}{\text{power(W)}}. \tag{13}$$

The normalized bit width is the bit width of accumulator divided by bit width of [8]. According to Table 2's measurement results, the power consumption of the 28 I/O pads is approximately 14 mW, and the core power is 14.2 mW at 286 MHz. The FoM of proposed frequency synthesizer is 18.2 (GHz/W).

Table 3 is the performance comparison among flying-adder-based frequency synthesizers. The proposed frequency synthesizer achieved the smallest area and wide output frequency range as compared with [19, 20] in 0.18 $\mu$m process technology. By setting the worst case in the fractional part of frequency control word, we accomplish in peak-to-peak jitter and RMS jitter performance improvement as compared with [8]'s flying-adder structure. This works also has better jitter performance than [23]'s two-path ADPLL in 0.13 $\mu$m process.

## 6. Conclusion

This paper presents an all-digital frequency synthesizer using an interpolated flying adder with improved jitter performance. The 32 equally spaced phases are generated by a multiphase ADPLL which consists of a 32-phase NAND gate-based VCO, a DVC circuit, and an SAR controller. The interpolated flying adder is implemented with two low-cost pseudo-N MUXs, a modified 29-bit carry-select adder as well as a digital interpolator. The jitter performance

of the proposed interpolated flying-adder-based frequency synthesizer is also obtained. The output frequency of the prototype chips has the range of 33 MHz ~ 286 MHz at 1.8 V. Moreover, the peak-to-peak ($P_k$-$P_k$) jitter of the output clock at 286 MHz is 215.2 ps, and the RMS jitter is 40.2 ps.

## Acknowledgments

The authors would like to thank L. Xiu, NovaTek, for useful discussions. They also would like to thank the Chip Implementation Center (CIC) of the National Science Council in Taiwan for chip fabrication. This work was supported by National Science Council of Taiwan, Taiwan, under Grant no. Nsc99-2221-E-327-047.

## References

[1] W. F. Egan, *Frequency Synthesis by Phase Lock*, John Wiley & Sons, New York, NY, USA, 2nd edition, 1999.

[2] P. Park, D. Park, and S. Cho, "A low-noise and low-power frequency synthesizer using offset phase-locked loop in 0.13 $\mu$m CMOS," *IEEE Microwave and Wireless Components Letters*, vol. 20, no. 1, Article ID 5339111, pp. 52–54, 2010.

[3] A. Yamagishi, M. Ugajin, and T. Tsukahara, "A 2.4-GHz PLL synthesizer for a 1-V Bluetooth RF transceiver," *IEICE Transactions on Electronics*, vol. E87-C, no. 6, pp. 895–900, 2004.

[4] K.-Y. Lee, H. Ku, and Y. B. Kim, "A fast switching low phase noise CMOS frequency synthesizer with a new coarse tuning method for PHS applications," *IEICE Transactions on Electronics*, vol. E89-C, no. 3, pp. 420–428, 2006.

[5] T. Watanabe and S. Yamauchi, "An all-digital PLL for frequency multiplication by 4 to 1022 with seven-cycle lock time," *IEEE Journal of Solid-State Circuits*, vol. 38, no. 2, pp. 198–204, 2003.

[6] S. Dosho, T. Morie, K. Okamoto, Y. Yamada, and K. Sogawa, "A -90 dBc@10 kHz phase noise fractional-N frequency synthesizer with accurate loop bandwidth control circuit," *IEICE Transactions on Electronics*, vol. 89, no. 6, pp. 739–744, 2006.

[7] X. Geng, F. F. Dai, J. D. Irwin, and R. C. Jaeger, "An 11-bit 8.6 GHz direct digital synthesizer MMIC with 10-bit segmented sine-weighted DAC," *IEEE Journal of Solid-State Circuits*, vol. 45, no. 2, Article ID 5405153, pp. 300–313, 2010.

[8] H. Mair and L. Xiu, "Architecture of high-performance frequency and phase synthesis," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 6, pp. 835–846, 2000.

[9] L. Xiu and Z. You, "A flying-adder architecture of frequency and phase synthesis with scalability," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 10, no. 5, pp. 637–649, 2002.

[10] L. Xiu and Z. You, "A new frequency synthesis method based on flying-adder architecture," *IEEE Transactions on Circuits and Systems II*, vol. 50, no. 3, pp. 130–134, 2003.

[11] L. Xiu, "A Flying-Adder On-chip frequency generator for complex SoC environment," *IEEE Transactions on Circuits and Systems II*, vol. 54, no. 12, pp. 1067–1071, 2007.

[12] L. Xiu, "A novel DCXO module for clock synchronization in MPEG2 transport system," *IEEE Transactions on Circuits and Systems I*, vol. 55, no. 8, pp. 2226–2237, 2008.

[13] L. Xiu, "A Flying-Adder PLL technique enabling novel approaches for video/graphic applications," *IEEE Transactions on Consumer Electronics*, vol. 54, no. 2, pp. 591–599, 2008.

[14] L. Xiu, "The concept of time-average-frequency and mathematical analysis of flying-adder frequency synthesis architecture," *IEEE Circuits and Systems Magazine*, vol. 8, no. 3, Article ID 4609962, pp. 27–51, 2008.

[15] L. Xiu, "Some open issues associated with the new type of component: digital-to-frequency converter," *IEEE Circuits and Systems Magazine*, vol. 8, no. 3, Article ID 4609966, pp. 90–94, 2008.

[16] L. Xiu, "A fast and power-area-efficient accumulator for flying-adder frequency synthesizer," *IEEE Transactions on Circuits and Systems I*, vol. 56, no. 11, Article ID 4785490, pp. 2439–2448, 2009.

[17] D. E. Calbaza and Y. Savaria, "A direct digital period synthesis circuit," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 8, pp. 1039–1045, 2002.

[18] T. Saeki, M. Mitsuishi, H. Iwaki, and M. Tagishi, "1.3-cycle lock time, non-PLL/DLL clock multiplier based on direct clock cycle interpolation for 'clock on demand'," *IEEE Journal of Solid-State Circuits*, vol. 35, no. 11, pp. 1581–1590, 2000.

[19] G. N. Sung, S. C. Liao, J. M. Huang, Y. C. Lu, and C. C. Wang, "All-digital frequency synthesizer using a flying adder," *IEEE Transactions on Circuits and Systems II*, vol. 57, no. 8, Article ID 5545381, pp. 597–601, 2010.

[20] Y. A. Chau, Y.-Y. Yang, and J.-F. Chen, "All-Digital frequency synthesizer with dual resolution," in *Proceedings of the International Symposium on Intelligent Signal Processing and Communication Systems*, pp. 630–633, Tottori, Japan, 2006.

[21] A. Rossi and G. Fucili, "Nonredundant successive approximation register for A/D converters," *Electronics Letters*, vol. 32, no. 12, pp. 1055–1057, 1996.

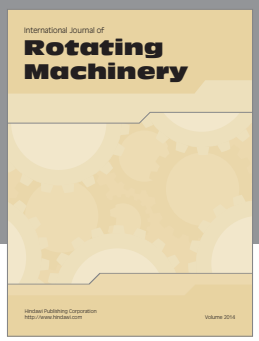
[22] P.-L. Chen, C.-F. Liu, and T.-H. Lin, "Amultiphase digital controlled oscillator with DVC techniqu," in *Proceedings of the 15th Workshop on Synthesis and System Integration of Mixed Information Technologies*, pp. 473–476, Okinawa, Japan, March 2009.

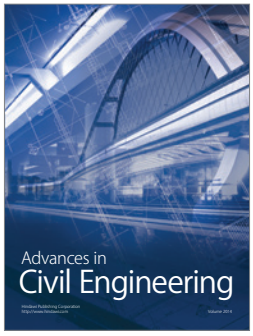
[23] W. Liu, W. Li, P. Ren, C. Lin, S. Zhang, and Y. Wang, "A PVT tolerant 10 to 500 MHz all-digital phase-locked loop with coupled TDC and DCO," *IEEE Journal of Solid-State Circuits*, vol. 45, no. 2, Article ID 5405141, pp. 314–321, 2010.

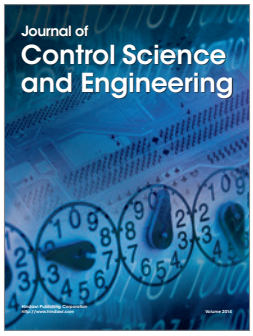
[24] H. Lu, C. Su, and C. N. J. Liu, "A tree-topology multiplexer for multiphase clock system," *IEEE Transactions on Circuits and Systems I*, vol. 56, no. 1, pp. 124–131, 2009.

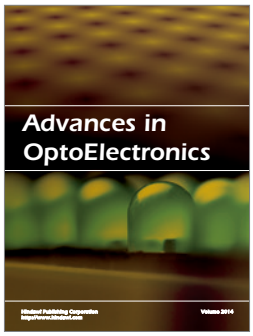
[25] P. P. Sotiriadis, "Theory of flying-adder frequency synthesizers—Part I: modeling, signals' periods and output average frequency," *IEEE Transactions on Circuits and Systems I*, vol. 57, no. 8, Article ID 5424107, pp. 1935–1948, 2010.

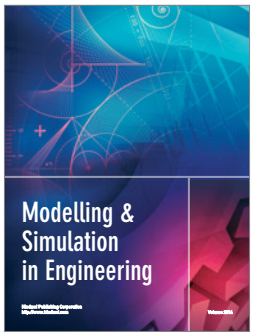
[26] P. P. Sotiriadis, " Theory of flying-adder frequency synthesizers—Part II: time- and frequency-domain properties of the output signal," *IEEE Transactions on Circuits and Systems I*, vol. 57, no. 8, pp. 1949–1963, 2010.

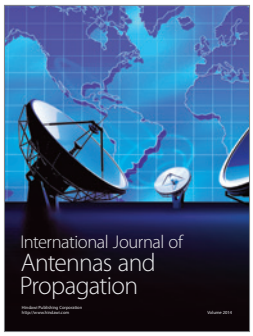


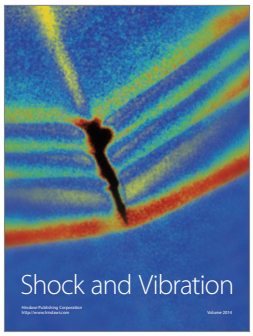


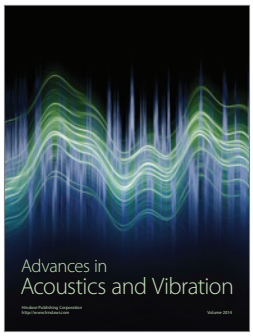













Hindawi

Submit your manuscripts at
http://www.hindawi.com