An All-Digital Large-N Audio Frequency Synthesizer for HDMI Applications

Ching-Che Chung, *Member, IEEE*, Duo Sheng, Chia-Lin Chang, Wei-Da Ho, Yang-Di Lin, and Fang-Nien Lu

***Abstract*—In this brief, a novel all-digital and large-frequency-multiplication-ratio audio frequency synthesizer for high-definition multimedia interface applications is presented. The proposed large-N frequency synthesizer is designed in an all-digital manner to reduce circuit complexity and design efforts in advanced CMOS process technology, as compared with prior studies. The proposed frequency synthesizer does not require an extra high-frequency reference clock source but employs a single locking loop to reduce lock-in time and enhance loop stability. Based on the proposed frequency search algorithm and the high-resolution digitally controlled oscillator, the frequency synthesizer cannot only provide a large frequency multiplication ratio, but it also achieves low-jitter performance. Measurement results show that the frequency multiplication ratio has a range of 4096 to 25 088 and that the power consumption of the proposed frequency synthesizer can be improved to 591 μW (at 24.576 MHz) with a peak-to-peak jitter of 1.23%. In addition, the proposed frequency synthesizer can be implemented with standard cells, making it easily portable to different processes and very suitable for system-on-a-chip applications.**

***Index Terms*—Frequency multiplication, high-definition multimedia interface (HDMI), jitter, phase-locked loops (PLLs).**

I. Introduction

As THE DATA rate demanded by multimedia system increases, the high-speed and high-quality interface is required. High-definition multimedia interface (HDMI) is an all-digital compact audio/video interface for transmitting uncompressed digital data. HDMI, which is the replacement for the consumer analog standards, will connect digital audio/video sources (such as HD digital versatile disc players, Blu-ray disc players, set-top boxes, personal computers, etc.) to compatible digital audio/video devices [1]. Because HDMI can transmit audio/video signals by a single cable with high definition, it is widely used for multimedia applications.

When audio data are carried across the HDMI link, the audio data is driven by a transition minimized differential signaling (TMDS) clock running at a rate corresponding to the video pixel clock rate, but it does not maintain the original audio sampling clock. Thus, the audio clock has to be regenerated at the HDMI sink device [1]. Fig. 1 illustrates the block diagram of the HDMI audio clock regeneration system. The HDMI source device determines the fractional relationship between the TMDS clock (TMDS_CLK) and the audio clock (AUDIO_CLK), which is equal to 128 times the audio sampling clock (Fs), and the N and cycle time stamp (CTS) values will be then transmitted to the HDMI sink device. The relationship between the frequency of TMDS_CLK and AUDIO_CLK is expressed by

$$f_{\mathrm{AUDIO\_CLK}} = f_{\mathrm{TMDS\_CLK}} \times N \div \mathrm{CTS}. \tag{1}$$

After the HDMI sink device receives TMDS_CLK, CTS, and N from the source device, it regenerates the audio clock by means of a frequency synthesizer.

In the frequency synthesizer, the TMDS clock divided by CTS (TMDS_CLK_divCTS) is around 1 kHz, the audio clock ranges from 4.096 to 24.576 MHz with different sampling clocks, and the maximal root-mean-square (RMS) period jitter should be smaller than 200 ps. Accordingly, the frequency multiplication ratio N is always larger than 4096 and up to over 20 000 in this application. Conventional frequency synthesizers are implemented with charge-pump-based architectures [2], [3]. However, the serious MOS capacitor leakage current problems in the advanced CMOS technology and a very low loop refresh rate due to a low-frequency reference clock will result in large frequency drifting. In addition, since the loop bandwidth inversely scales with the frequency multiplication ratio, the tracking jitter linearly scales with the frequency multiplication ratio [2]. As a result, the conventional charge-pump-based frequency synthesizer architecture cannot be directly applied to implement the audio frequency synthesizer in HDMI applications.

In contrast to the charge-pump-based frequency synthesizer, the all-digital frequency synthesizer (ADFS), which does not utilize any passive components, uses the digital design approaches, which allows it to be easily integrated into digital systems under the advanced CMOS technology [4], [5].

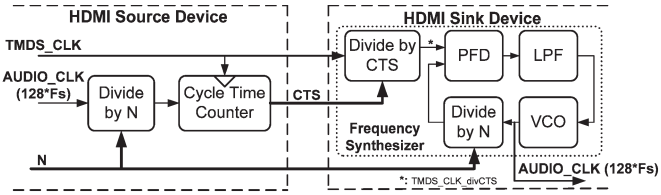

Fig. 1. Block diagram of the HDMI audio clock regeneration system.

Manuscript received December 14, 2011; revised February 23, 2012; accepted April 21, 2012. Date of publication June 1, 2012; date of current version July 13, 2012. This work was supported in part by the National Science Council of Taiwan under Grant NSC100-2221-E-194-051. The shuttle program supported by United Microelectronics Corporation is acknowledged as well. This brief was recommended by Associate Editor C.-Y. Lee.

C.-C. Chung, C.-L. Chang, W.-D. Ho, Y.-D. Lin, and F.-N. Lu are with the Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi 621, Taiwan (e-mail: wildwolf@cs.ccu.edu.tw).

D. Sheng is with the Department of Electrical Engineering, Fu Jen Catholic University, New Taipei City 24205, Taiwan.

Color versions of one or more of the figures in this brief are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSII.2012.2198980

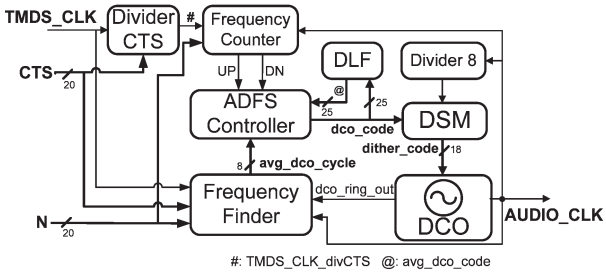

Fig. 2. Proposed ADFS architecture.

Recently, different architectural solutions have been proposed to implement ADFS [4]–[9]. A direct-digital frequency synthesizer (DDFS) employs a mathematical operation to generate the desired output frequency directly. Although DDFS can switch output frequency very quickly, it requires a high-speed and high-resolution digital-to-analog converter to convert the digital control code to the analog signal [6]. The flying-adder-based frequency synthesizer is another important ADFS type that can synthesize output frequency with large frequency multiplication ratios and switch output frequencies rapidly. However, it requires an extra phase-locked loop (PLL) to provide a high-frequency and multiphase reference clock source, resulting in increasing power consumption and chip area [6], [7]. The dual-loop-type ADFS cascades two PLLs to increase the frequency multiplication ratio. However, it is not only difficult to arrange a divider ratio to obtain an arbitrary multiplication ratio, but the power consumption is also increased [8].

In this brief, an ADFS with a large frequency multiplication ratio for HDMI applications is presented. In the proposed ADFS, the single-loop architecture is used without an extra high-frequency reference clock source to reduce power consumption. The proposed frequency search algorithm and high-resolution digitally controlled oscillator (DCO) increase the frequency multiplication ratio and meet the requirements of system jitter performance. Moreover, because the proposed ADFS had a good portability as a soft intellectual property (IP), it would be very suitable for system-on-a-chip (SoC) applications, as well as system-level integration.

II. Proposed ADFS Design

A. Proposed ADFS Architecture

Fig. 2 illustrates the architecture of the proposed ADFS, which is composed of a frequency finder (FF), an ADFS controller, a DCO, a frequency counter (FC), a digital loop filter (DLF), a delta–sigma modulator (DSM), and two frequency dividers (divider CTS and divider 8). TMDS_CLK is taken as the reference clock, and TMDS_CLK_divCTS is TMDS_CLK divided by CTS through divider CTS. AUDIO_CLK is the output of the DCO. Based on the frequency search algorithm, the FF can search the target frequency very quickly and then provide the initial DCO control code (avg_dco_cycle) for the ADFS controller. The FC detects the frequency differences between N times TMDS_CLK_divCTS and AUDIO_CLK. When the ADFS controller receives UP from the FC, the DCO control code (dco_code) will be decreased to speed up the output frequency of the DCO. Oppositely, when the ADFS controller receives DN from the FC, the ADFS controller adds the DCO control code to decrease the output frequency of the DCO. The DLF [4] generates a baseline DCO control code (avg_dco_code) to filter out the jitter effect of TMDS_CLK. In addition, in order to enhance the frequency resolution of the DCO, a DCO dithering scheme is employed through a 7-bit first-order DSM. Therefore, the integral part of the DCO control code has 18 bits, and the fractional part of the DCO control code has 7 bits. The operation speed of the DSM is the output audio clock frequency divided by 8.

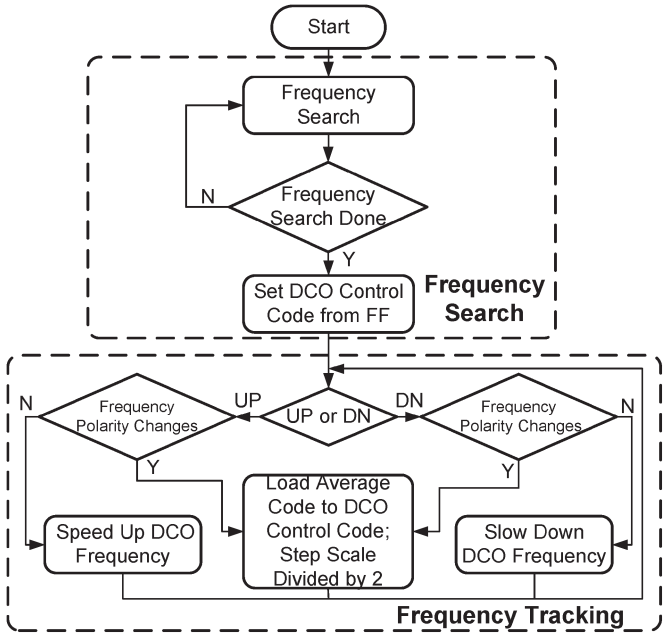

Fig. 3. Flowchart of the proposed lock-in algorithm.

B. Lock-in Algorithm

Fig. 3 illustrates the flowchart of the proposed lock-in algorithm. The locking procedure of the proposed frequency synthesizer is divided into frequency search and tracking states. First, in the frequency search state, the FF searches the target output frequency very quickly and then determines the initial DCO control code (avg_dco_cycle). Second, the ADFS enters into the frequency tracking state. A binary search scheme is used to reduce the lock-in time to search for the target DCO control code. Based on UP and DN outputted by the FC, the DCO control code will be changed to track the frequency of TMDS_CLK. Once the frequency polarity changes, the search step will be divided by 2, and the baseline DCO control code will be restored to the current DCO control code to converge the frequency tracking [4].

III. Circuit Design

A. Frequency Finder

In order to reduce the lock-in time of the ADFS, the proposed locking algorithm employs an FF to obtain the initial output frequency in the beginning. The block diagram of the proposed FF is shown in Fig. 4(a). The frequency of TMDS_CLK is divided by 256 to enlarge the clock period and then quantized by a time-to-digital converter (TDC). To increase the accuracy, the period

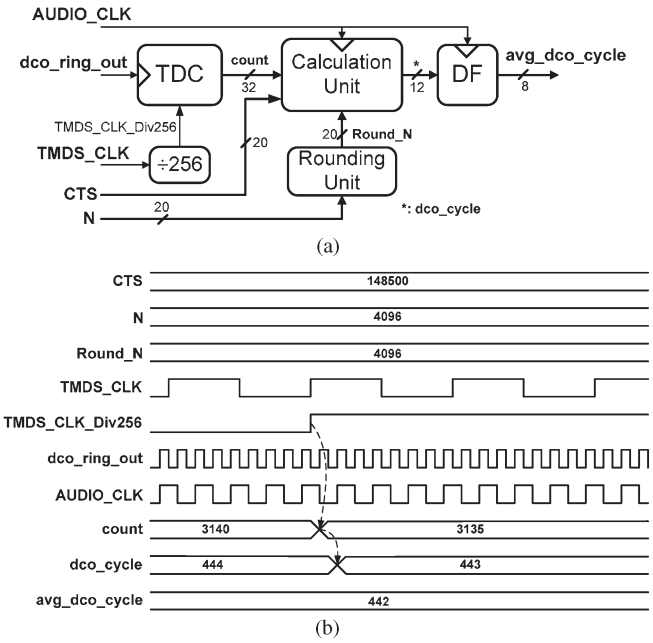

Fig. 4. Proposed FF. (a) Block diagram. (b) Timing diagram.

of TMDS_CLK needs to enlarge 256 times. The TDC uses a high-speed clock (dco_ring_out), which generated inside the proposed DCO to quantize the enlarged TMDS_CLK's period. The calculation unit takes the quantized value (count), CTS, and Round_N, which generated from the rounding unit to produce the DCO cycle value (dco_cycle). The calculation equation is expressed by

$$\text{dco\_cycle} = \text{count} \times \text{CTS} \div (N \times 256). \tag{2}$$

To reduce the circuit complexity, the rounding unit is used to round N to 2^N, where 2^N is the nearest value of N. In addition, a digital filter [4] is used to suppress the jitter effect of TMDS_CLK and generate an average DCO cycle value (avg_dco_cycle). The avg_dco_cycle value will determine the initial DCO control code and the output frequency. After the frequency searching state is completed, the binary search algorithm will further reduce the frequency error. The timing diagram of the proposed FF is also shown in Fig. 4(b).

B. DCO

Basically, DCO dominates the major performance of the all-digital clock generator, such as power consumption and jitter performance, and, hence, is the most important component of such clocking circuits [10]. To achieve the high portability of the proposed ADFS, all of its components, including DCO, are implemented with standard cells. Fig. 5 illustrates the architecture of the proposed DCO, which employed cascading structure for a cyclic-controlled stage (CCS), a coarse-tuning delay stage (CTDS) and a fine-tuning delay stage (FTDS) to achieve a fine frequency resolution and a wide operation range. Because the requested lowest output frequency in the system is 4.096 MHz, the conventional delay line structure is not suitable for this application. Thus, the proposed DCO employs a cyclic-controlled delay line (CCDL) to generate a low-frequency clock with a low hardware cost [11]. The

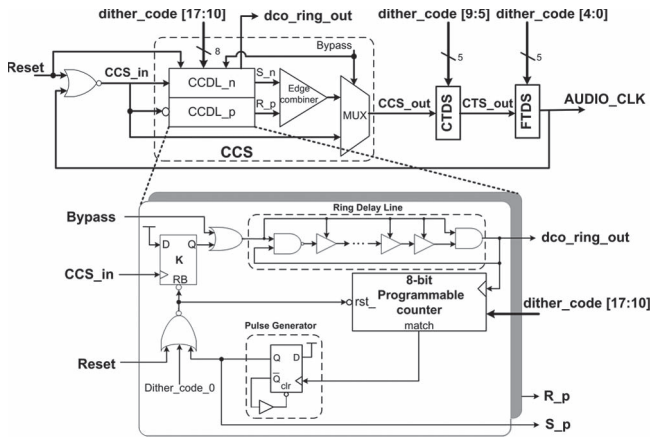

Fig. 5. Proposed DCO and CCS architecture.

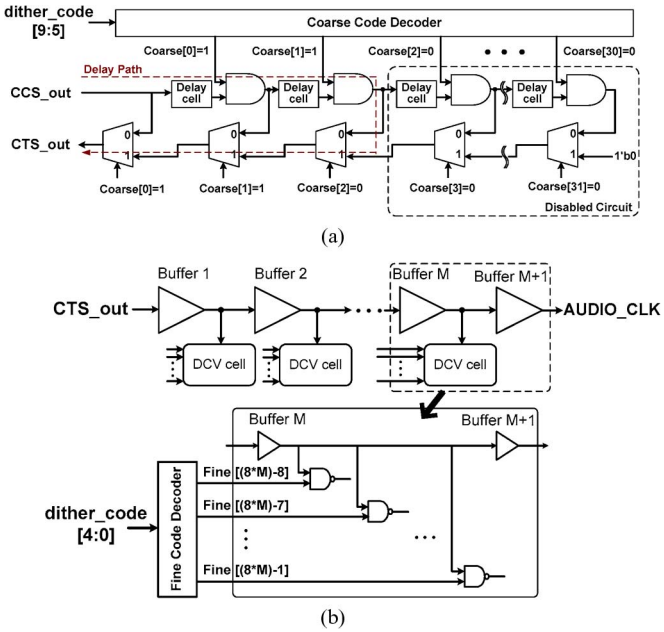

Fig. 6. Block diagram of (a) the CTDS and (b) the FTDS.

CCS is composed of two CCDLs, an edge combiner, and a multiplexer. The edge combiner receives S_n and R_p from CCDL_n and CCDL_p, respectively, and then generates the complete cycle clock signal. The multiplexer will determine whether the DCO bypasses the CCS. In the CCDL, once the flip-flop K is triggered by CCS_in, the ring delay line generates dco_ring_out continuously until the flip-flop K has been reset. When the count value of the 8-bit programmable counter is equal to dither_code [17:10], the pulse generator generates a short pulsewidth signal (S_n and R_p) and resets the flip-flop K. Because the period of the output clock can be easily enlarged by the cycle control code extension, the proposed DCO could achieve a wide operating frequency range.

The CTDS, which has 31 delay cells with 32 multiplexers, can provide 32 different delays, as shown in Fig. 6(a). In addition, two-input AND gates are added to each delay cell's output to disable unused cells and save power consumption [12]. To achieve better DCO resolution, digital-controlled varactors [10], [12] are used in the FTDS. Fig. 6(b) shows the DCO's FTDS. The FTDS has M buffers, and each buffer

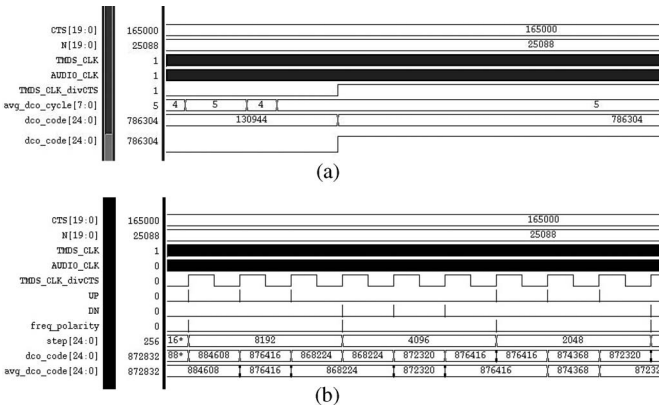

Fig. 7. Simulation results of (a) the frequency search stage and (b) the frequency tracking state.

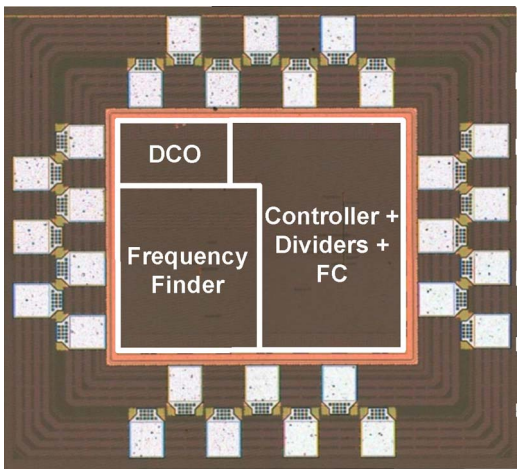

Fig. 8. Microphotography of ADFS test chip.

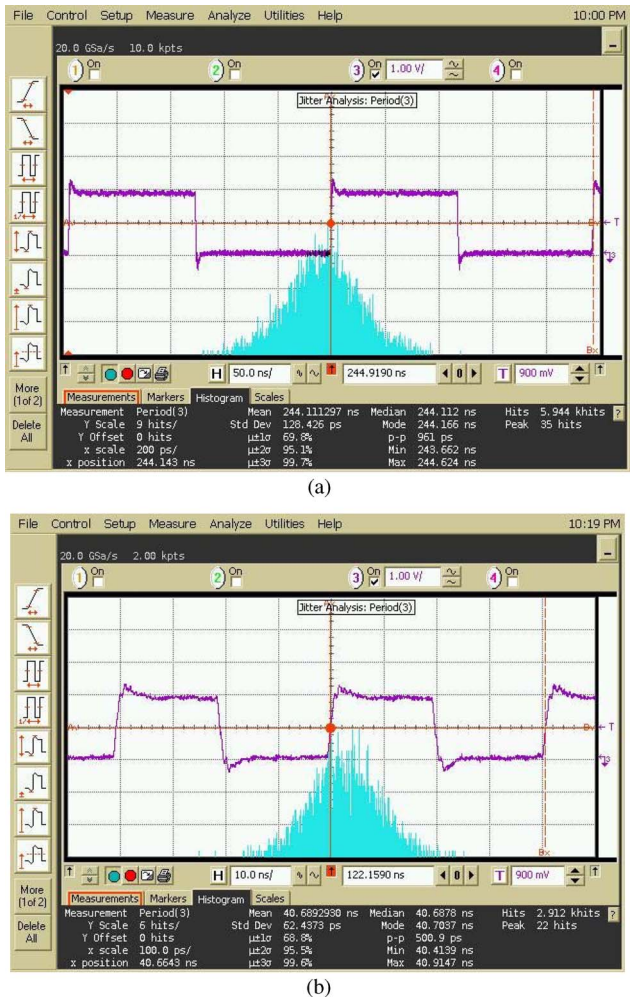

Fig. 9. Jitter histogram of ADFS at (a) 4.096 and (b) 24.576 MHz.

connects to eight NAND gates. When the fine-tuning control code $(\text{Fine}[(8*M)-1:(8*M)-8])$ is changed, the capacitance in the buffer's output node is also changed. As a result, a high-resolution fine-tuning delay stage with good linearity could be created. Based on the requested operating frequency in the system application, the design parameter M is equal to 4.

IV. Implementation and Experimental Results

Fig. 7 shows the simulation waveform of the proposed ADFS. The locking procedure of the proposed frequency synthesizer is divided into the frequency search state and tracking states. In the beginning, the proposed frequency synthesizer starts from the frequency search state. In this state, the FF employs the TDC and the digital filter to generate the initial DCO control code to approach the target output frequency very quickly, as shown in Fig. 7(a). After the frequency searching is finished, the ADFS enters into the frequency tracking state. A binary search scheme is used in the ADFS controller to reduce the lock-in time to search for the target DCO control code. Based on UP and DN, outputted by FC, the DCO control code is changed to track the frequency of TMDS_CLK, as shown in Fig. 7(b). The test chip is fabricated in a standard-performance 65-nm CMOS technology, where the chip microphotography of the ADFS chip is shown in Fig. 8. The chip size is 812×722 μm^2, and the core size is 362×269 μm^2.

Fig. 9 shows the measured jitter histogram of the output clock (AUDIO_CLK) at the lowest and highest frequencies, respectively. Fig. 9(a) shows the RMS and peak-to-peak output period jitters as 128 and 961 ps, respectively, at 4.096 MHz, and Fig. 9(b) shows the RMS and peak-to-peak output period jitters as 62 and 501 ps, respectively, at 24.576 MHz. The measurement results for different frequency multiplication ratios N and output frequencies are summarized in Table I. The measurement results show that the range of the frequency multiplication ratio is from 4096 to 25 088 and that the maximum peak-to-peak output period jitter is 1.23%, which shows that the period jitter is effectively reduced by the proposed locking algorithm and DCO dithering scheme. The power consumption values are 0.566 and 0.591 mW at 4.096 and 24.576 MHz, respectively.

Table II lists the comparison results for the state-of-the-art frequency synthesizers. In terms of the frequency multiplication ratio, the proposed ADFS has the highest frequency multiplication ratio as compared with other frequency synthesizers. Furthermore, the proposed ADFS does not induce any performance loss, including jitter performance. In addition, since the proposed architecture is a single locking loop that does not require an extra high-frequency reference clock source and

TABLE I
JITTER PERFORMANCE SUMMARY

TMDS Clock (MHz)	Sampling Clock (kHz)	CTS	N	Audio Clock (MHz)	P2P Jitter (ps)/%	RMS Jitter (ps)/%
25.2	32	25200	4096	4.096	961/0.39	128.4/0.05
148.5	32	148500	4096	4.096	906.4/0.37	134.5/0.06
25.2	176.4	28000	25088	22.5792	439.9/0.99	55.1/0.12
25.2	192	25200	24576	24.576	500.9/1.23	62.4/0.15
148.5	192	148500	24567	24.576	494.1/1.21	54.4/0.13

TABLE II
PERFORMANCE COMPARISONS

Performance Indices	Proposed	JSSC'03 [2]	JSSC'03 [9]	JSSC'06[8]	TCASII'10[6]
Process	65nm CMOS	0.13μm CMOS	0.65μm CMOS	0.18μm CMOS	0.18μm CMOS
Design Approach	All-Digital	Analog	All-Digital	All-Digital	All-Digital
Structure	Single Loop	Charge-Pump	TDC-Based	Dual Loop	Flying Adder
Input Range	25.2 MHz ~ 148.5 MHz (TMDS Clock)	NA	11.2 ~ 339.7 kHz	19.26 kHz ~ 60 MHz	NA
Output Range (MHz)	4.096 ~ 24.576	30 ~ 650	0.0449 ~ 61.3	2.4 ~ 378	39.38 ~ 226
Multiplication Factor	4096 ~ 25088	1 ~ 4096	4 ~ 1022	4 ~ 13888	NA
P2P Period Jitter	439.9ps (0.99%) @22.5792MHz, N=25088	1.7% @240MHz, N=4096	4.8% @30.4MHz, N=1022	2.8% @134.7MHz, N=13888	2.4%@187.5MHz
Power Consumption (mW)	0.591 @24.576MHz	7 @240MHz	NA	15 @378MHz	3.6
Area (mm^2)	0.097	0.182	1.17	0.16	0.16
Portability	Yes	No	Yes	Yes	Yes

without passive components such as loop capacitors, it achieves low-complexity and lower power consumption as compared with other designs. Moreover, since the proposed design can be implemented with standard cells, it has a good portability and is very suitable for SoC integration. As a result, the proposed ADFS has the benefits of a higher frequency multiplication ratio, better jitter performance, less power consumption, and greater portability.

V. CONCLUSION

In this brief, we have proposed an all-digital and large-frequency-multiplication-ratio audio frequency synthesizer with cell-based design for HDMI applications. As compared with conventional designs, the proposed ADFS employs the single-locking-loop architecture and the all-digital design approach in order to enhance loop stability and reduce circuit complexity and design efforts in advanced CMOS process technology. A DCO dithering scheme has been applied to improve the frequency resolution of the DCO. The proposed ADFS not only provides a large frequency multiplication ratio but also achieves low-jitter performance. Moreover, because the proposed ADFS has a good portability as a soft IP, it would be very suitable for SoC applications, as well as system-level integration.

ACKNOWLEDGMENT

The authors would like to thank their colleagues in the Silicon Sensor and System (S3) Laboratory of the National Chung Cheng University for many fruitful discussions.

REFERENCES

[1] HDMI, HDMI specification version 1.3, Jun. 2006.
[2] J. G. Maneatis, J. Kim, I. McClatchie, J. Maxey, and M. Shankaradas, "Self-biased high-bandwidth low-jitter 1-to-4096 multiplier clock generator PLL," *IEEE J. Solid-State Circuits*, vol. 38, no. 11, pp. 1795–1803, Nov. 2003.
[3] C.-C. Hung and S.-I. Liu, "A leakage-compensated PLL in 65-nm CMOS technology," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 56, no. 7, pp. 525–529, Jul. 2009.
[4] C.-C. Chung and C.-Y. Ko, "A fast phase tracking ADPLL for video pixel clock generation in 65nm CMOS technology," *IEEE J. Solid-State Circuits*, vol. 46, no. 10, pp. 2300–2311, Oct. 2011.
[5] S.-Y. Yang, W.-Z. Chen, and T.-Y. Lu, "A 7.1 mW, 10 GHz all digital frequency synthesizer with dynamically reconfigured digital loop filter in 90 nm CMOS technology," *IEEE J. Solid-State Circuits*, vol. 45, no. 3, pp. 578–586, Mar. 2010.
[6] 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 Trans. Circuits Syst. II, Exp. Briefs*, vol. 57, no. 8, pp. 597–601, Aug. 2010.
[7] W. Chen, P. Gui, and L. Xiu, "A low-jitter digital-to-frequency converter based frequency multiplier with large N," in *Proc. IEEE Int. MWSCAS*, Aug. 2011, pp. 1155–1158.
[8] P.-L. Chen, C.-C. Chung, J.-N. Yang, and C.-Y. Lee, "A clock generator with cascaded dynamic frequency counting loops for wide multiplication range applications," *IEEE J. Solid-State Circuits*, vol. 41, no. 6, pp. 1275–1285, Jun. 2006.
[9] T. Watanabe and S. Yamauchi, "An all-digital PLL for frequency multiplication by 4 to 1022 with seven-cycle lock time," *IEEE J. Solid-State Circuits*, vol. 38, no. 2, pp. 198–204, Feb. 2003.
[10] D. Sheng, C.-C. Chung, and C.-Y. Lee, "An ultra-low-power and portable digitally controlled oscillator for SoC applications," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 54, no. 11, pp. 954–958, Nov. 2007.
[11] H.-H. Chang, S.-M. Lee, C.-W. Chou, Y.-T. Chang, and Y.-L. Cheng, "A 1.6-880 MHz synthesizable ADPLL in 0.13 μm CMOS," in *Proc. Int. Symp. VLSI-DAT*, Apr. 2008, pp. 9–12.
[12] C.-C. Chung, C.-Y. Ko, and S.-E. Shen, "A built-in self calibration circuit for monotonic digitally controlled oscillator design in 65 nm CMOS technology," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 58, no. 3, pp. 149–153, Mar. 2011.