A fast-locking all-digital delay-locked loop for phase/delay generation in an FPGA*

Chen Zhujia(陈柱佳)[1, 2], Yang Haigang(杨海钢)[1, †], Liu Fei(刘飞)[1], and Wang Yu(王瑜)[1, 2]

[1]Institute of Electronics, Chinese Academy of Sciences, Beijing 100190, China
[2]Graduate University of the Chinese Academy of Sciences, Beijing 100049, China

Abstract: A fast-locking all-digital delay-locked loop (ADDLL) is proposed for the DDR SDRAM controller interface in a field programmable gate array (FPGA). The ADDLL performs a 90° phase-shift so that the data strobe (DQS) can enlarge the data valid window in order to minimize skew. In order to further reduce the locking time and to prevent the harmonic locking problem, a time-to-digital converter (TDC) is proposed. A duty cycle corrector (DCC) is also designed in the ADDLL to adjust the output duty cycle to 50%. The ADDLL, implemented in a commercial 0.13 μm CMOS process, occupies a total of 0.017 mm^2 of active area. Measurement results show that the ADDLL has an operating frequency range of 75 to 350 MHz and a total delay resolution of 15 ps. The time interval error (TIE) of the proposed circuit is 60.7 ps.

Key words: all digital DLL; DDR SDRAM controller; time-to-digital converter; duty cycle corrector; DCDL; FPGA

1. Introduction

Delay-locked loops (DLLs) are widely used as phase shifters in clock de-skew buffers, multiphase clock generators and DRAM interfaces in high speed field programmable gate arrays (FPGAs). Compared with the phase-lock loop (PLL), a DLL has advantages of better jitter performance, stability, improved phase tracking ability and ease of design[1]. Digital DLLs are a good alternative in applications such as phase shifting in the DRAM interfaces of FPGAs.

In double data rate (DDR) SDRAM controller designs, data transfers are based on the bidirectional data strobe (DQS) signal transmitted with the output data (DQ)[2]. Figure 1 shows the timing budget in read operation. Ideally, both signals are edge aligned by a DDR SDRAM. However, due to PCB board skew and pin-to-pin mismatch between the DQS and the DQ, skew exists between the DQS and DQ when they arrive at the controller, making the data valid window smaller. In order to enlarge the data valid window, the DQS needs to be delayed by a 90° phase shift to the center of the data window. One way to implement the phase shift is to use a buffer chain. The disadvantages of this method are that the buffer chain cannot meet a wide input frequency range and its delay value varies with process, voltage and temperature (PVT) variations. Compared with this method, digital DLLs are preferred due to their fast locking time and ease of migration over different processes in high speed DDR memory interface controllers[3−7].

To tolerate wide variations of clock frequency and PVT, a DLL needs to be able to operate in a wide frequency range. The highest frequency of a DLL is decided by the delay of a single delay unit and the lowest frequency is decided by the length of the delay line. All of the above demand that the delay line of a DLL has a high-bandwidth. To meet this requirement, conventional digital DLLs[9, 10] consume large numbers of delay units and increase supply-induced jitter due to the longer delay line.

Conventional digital DLLs[9] employ counters to adjust the delay lines, which makes the locking time increase exponentially as the number of control bits increases. Thus a short locking time is required for the ADDLL to generate a phase-shifted clock signal when the controller switches from power down mode to active mode. A binary searching algorithm was adopted in Refs. [11, 12] to reduce the locking time in proportion to the control bits, but the open-loop characteristic makes it hard to track PVT variations.

In this work, an improved all-digital DLL (ADDLL) is proposed. Compared with conventional digital DLLs[9, 10], which use a single delay line, the digital control delay line (DCDL) of the proposed ADDLL is composed of a coarse delay unit and a fine delay unit. The proposed ADDLL meets the short locking time requirement by utilizing a novel time-to-digital converter (TDC) to coarse lock the ADDLL by tuning the coarse delay unit in one clock cycle. The proposed structure achieves a short locking time and prevents the possibility of harmonic locking.
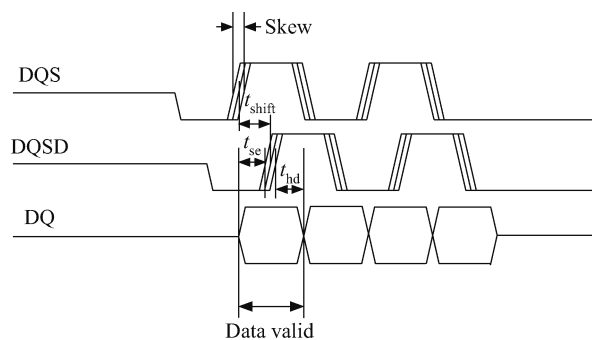

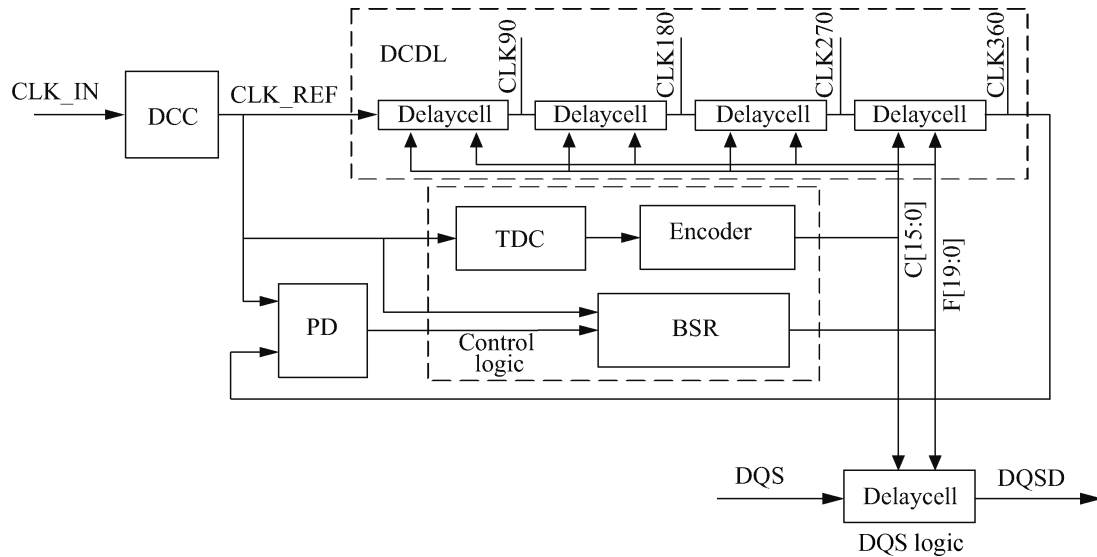Fig. 1. DDR SDRAM Read operation timing budget.

Fig. 2. Architecture of the proposed ADDLL.

A duty-cycle correction (DCC) circuit is designed to adjust the duty cycle of the input clock signal close to 50%.

2. Architecture of the all digital DLL

Figure 2 shows the architecture of the proposed ADDLL. It is composed of a phase detector (PD), a bidirectional shift register (BSR), a TDC, a DCDL and a DCC. Like DLL-based multi-phase clock generators, the ADDLL has a multi-stage delay line with the same digital control code to generate an equally spaced multi-phase clock output[7]. In this design, four duplicated delay cells are utilized in the DCDL to generate a 90° phase shift, each of which contains a coarse delay unit and a fine delay unit. The delay cell in the DQS logic is the same as in the DCDL. When the ADDLL is in the locked state, the delay time of the DCDL is one clock cycle, which is averagely divided into quarters by the four delay cells. Then, the 90° phase shift of the DQS can be tuned by the delay component in the DQS logic.

The locking procedure of the ADDLL is divided into 3 steps: coarse locking by TDC, fine locking by the PD and BSR, and state holding. Initially, the signal RESET_N is low, which resets control codes C[15:0] and F[19:0] to zero for minimum delay. When RESET_N goes to high, the ADDLL enters into the locking procedure. At the beginning, the DCC takes 6 clock cycles to adjust the input clock duty cycle to near 50% and generate the output signal CLK_REF fed into the DCDL and TDC. After the DCC finishes its adjustment, the ADDLL enters into the coarse tuning procedure. The PD and BSR are disabled in this period. The TDC estimates the input clock cycle by a multiple of the coarse delay value and generates the coarse delay control code C[15:0] for the DCDL. The TDC completes coarse tuning in one clock period and then generates a control signal to enable the PD and BSR, and the ADDLL begins fine tuning. After this step, the delay difference between the reference phase and the output phase of the DCDL is less than a step size of the coarse delay unit.

After coarse locking, the DCDL will be fine tuned by the BSR. The PD compares the reference phase with the output phase of the delay line and then generates an UP/DOWN signal for the BSR. The state of the BSR is controlled by the UP/DOWN signal from the PD. The BSR generates a thermometer control code F[19:0], which helps to reduce dithered switching of the control code. The 4-stage duplicated delay cells in the DCDL generate 4 equally spaced signals: CLK90, CLK180, CLK270 and CLK360. When the output clock lags or leads the reference clock, the PD generates an UP or DOWN signal for the BSR. When the phase difference is within the lock range of the lock detector, a LOCK signal is created by the lock detector to turn the ADDLL into the hold state.

3. Circuit implementation

3.1. Digital control delay line (DCDL)

The DCDL is the most important part of an ADDLL design, as it decides the operating frequency range, delay resolution and delay linearity of the ADDLL. The DCDL in this design is divided into two parts to extend its tunable range and to reduce the locking time: a coarse delay unit and a fine delay unit. According to the requirements of the ADDLL, the delay line needs 4 duplicated delay cells to generate a 90° phase shift. To track PVT variations, the minimum delay of each delay cell in a worst case scenario (slow, 1.35 V, 125 ℃) is supposed to be shorter than 1/4 of the input reference clock period, and larger than 1/4 of the input reference clock period in the best case scenario (fast, 1.65 V, − 40 ℃).

The conventional digital-controlled delay unit is shown in Fig. 3(a). It has two different delays controlled by a multiplexer. The tunable range can be increased by cascading the delay units, but this also increases the intrinsic delay. So the maximum operating frequency is restricted by the large intrinsic delay. The proposed CDU shown in Fig. 3(b) is a MUX-based structure mirror delay line. The length of the delay chain can be unlimited and enables a wide range of frequencies and delays. In this design, a 16-stage delay line is designed for the coarse delay cell.

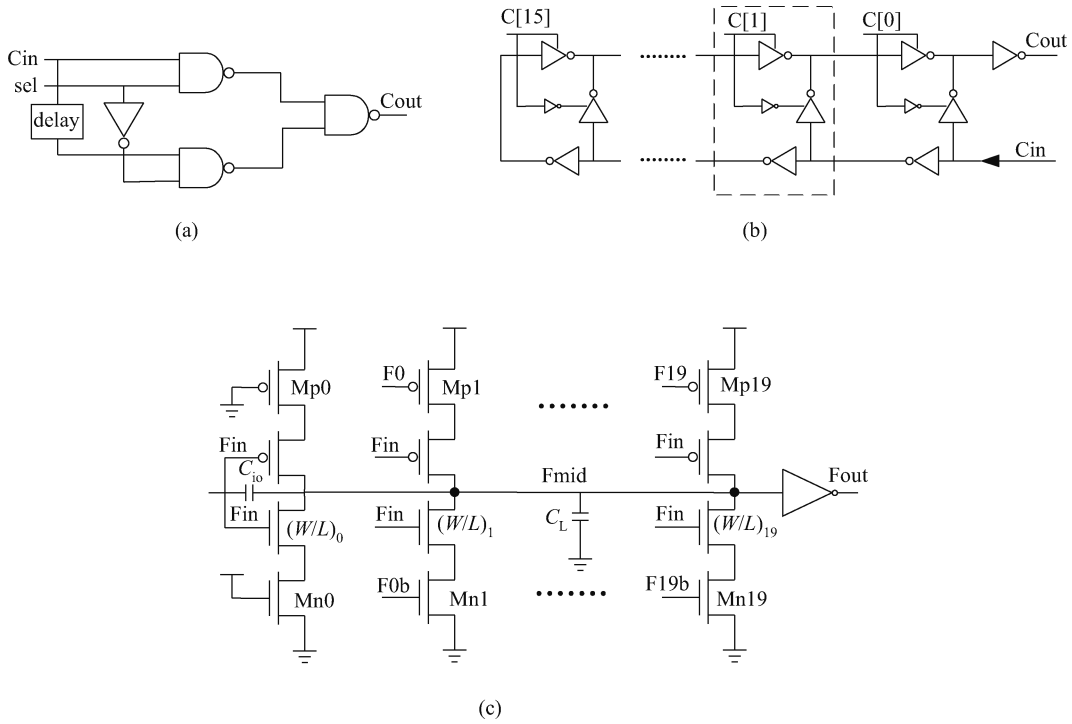A fully digital fine delay cell is implemented by adopting

Fig. 3. (a) Conventional digital delay unit. (b) Coarse delay unit. (c) Fine delay unit.

an inverter-based structure, as shown in Fig. 3(c). The delay cell has a minimum delay when all the control bits F[19:0] are low and a maximum delay when all the bits are high. To cover the delay of a coarse control bit, the tuning range of the fine delay cell is designed to be larger than the step size of the coarse delay cell.

To obtain a uniform delay granularity of the FDU, the size of the transistors in the FDU need to be tuned carefully. The delay time of the FDU is analyzed in this design as follows. The switch-controlling transistors Mn1–Mn19 and Mp1–Mp19 are used as switching transistors to turn on/off the respective inverters. With more switching transistors turned on, the charging and discharging currents through the output capacitance of the first inverter become larger, thus the delay value of fine delay cell becomes smaller. As analyzed in Ref. [15], the falling delay T_{fd} of an inverter is that:

$$T_{\mathrm{fd}} = R_{\mathrm{n}} C_{\mathrm{tot}} \left[-\ln \frac{1}{2} \left(1 + \frac{C_{\mathrm{io}}}{C_{\mathrm{tot}}} \right) \right], \quad (1)$$

where C_{io} is the capacitance between input node Fin and the inverter's output node Fmid. C_{tot} is the sum of output node capacitance C_{L} and C_{io}. R_{n} is the equivalent resistance, which is in inverse proportion to the W/L ratio of the NMOS transistor. When the switching transistors Mn1–Mnk are turned on, the equivalent W/L ratio of the NMOS transistor is:

$$\left(\frac{W}{L} \right)_k^{\mathrm{eq}} = \left(\frac{W}{L} \right)_0 + \cdots + \left(\frac{W}{L} \right)_k. \quad (2)$$

To obtain a linear delay increment in the FDU, the W/L ratio of the current controlling transistors should meet Eq. (3). By carefully tuning the size of transistors in the DCDL, the DCDL can obtain a timing resolution of less than 15 ps and a total tunable delay range from 2.8 to 13.2 ns.

$$\left[\left(\frac{W}{L} \right)_{k+1}^{\mathrm{eq}} \right]^{-1} - \left[\left(\frac{W}{L} \right)_k^{\mathrm{eq}} \right]^{-1} =$$
$$\left[\left(\frac{W}{L} \right)_k^{\mathrm{eq}} \right]^{-1} - \left[\left(\frac{W}{L} \right)_{k-1}^{\mathrm{eq}} \right]^{-1}. \quad (3)$$

3.2. Phase detector (PD)

A three-state PD with a lock-state window is shown in Fig. 4. The lock-state window width is $2\Delta t$, where Δt is the delay difference of delay elements D1 and D2. The PD shown in Fig. 4(a) is a two-state bang-bang PD based on cross-coupled RS latch. The bang-bang PD detects the phase difference between the reference clock CLK_REF and the feedback clock CLK_OUT, and then changes the UP/DOWN signal periodically. Figure 4(b) shows a three-state PD with lock detecting, which consists of three two-state PDs[13]. The first PD compares the phases of the feedback signal to the reference, and the second and third PDs compare the delayed version of the reference and feedback signal. When the feedback signal leads the reference signal, an UP signal is generated; when it lags, a DOWN signal is generated. When the feedback signal falls into the lock-state window, a LOCKED signal is generated and the phase detector is considered to be in a locked state. The characteristics of the PD are shown in Fig. 5. The lock-state window width is designed to be larger than the delay resolution of the DCDL.

3.3. Duty cycle corrector

Due to the large-scale clock networks in the FPGA, the input reference clock of the ADDLL coming from it will propa-

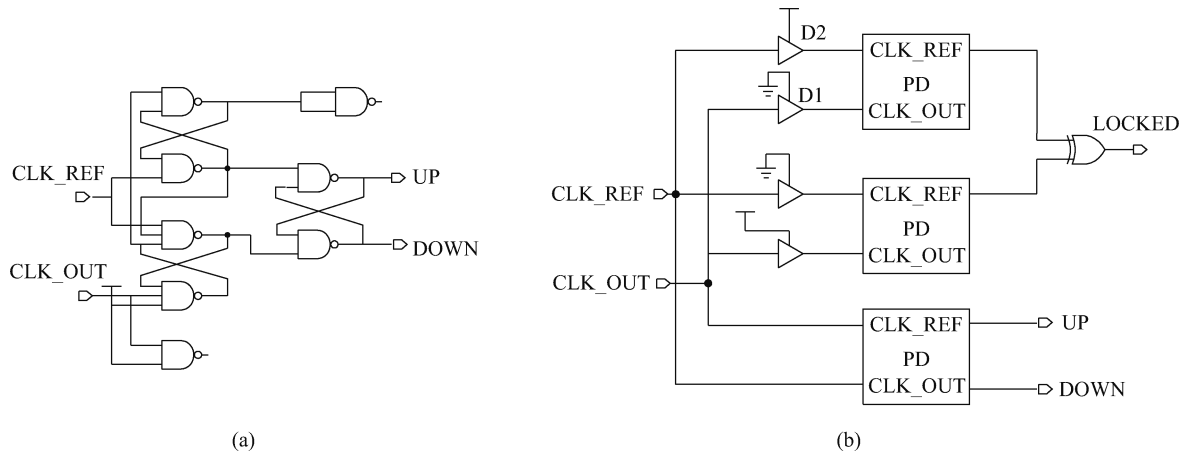(a)   (b)
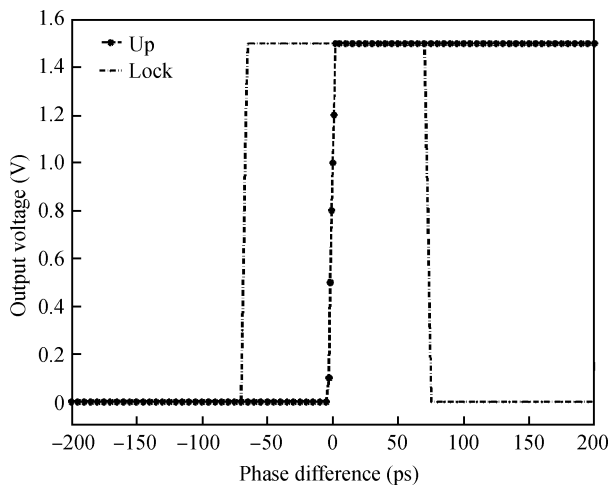
Fig. 4. Phase detector.

Fig. 5. Simulated characteristics of the PD and lock detector.

gate a long way, which causes duty cycle degradation. To maintain a 50% duty cycle, a duty cycle corrector (DCC) is embedded in the proposed ADDLL. As Figure 6(a) shows, the proposed DCC is composed of two edge detectors, a keeper, two delay chains and a successive approximation register (SAR). The two delay chains have the same delay time and the same control word DutyCode[5:0] generated by the SAR. The SAR is utilized to reduce the adjustment time of the DCC. The rising edge delay of Cout is a NAND's constant delay, while the falling edge delay is decided by the delay chain. As depicted in Fig. 6(b), t_r is the rising edge delay, t_f is the delay between the rising edge of B and the falling edge of Cout, and t_{delay} is the delay of the delay chain. Thus the output duty cycle is:

$$t_{duty} = t_{delay} + t_r - t_f, \qquad (4)$$

when t_r and t_f are tuned to be the same, the output duty cycle is equal to t_{delay}. The DCC works as follows: the output clock Cout is generated by the edge detectors, whose duty cycle is t_{delay}. Cdelay is the output of the delay line. A D flip-flop is used as a phase comparator to compare the phase of Cdelay and Cout. A 6-bit SAR is used in the circuit, which requires 6 clock cycles to finish the DCC adjustment, as depicted in Ref. [14].

The timing diagram of duty cycle detection is shown in Fig. 6(c). Two cases are shown in this waveform. As shown in Fig. 6(c), the duty cycle of Cout and the delay between Cdelay and Cout are both t_{delay}, thus the delay between the falling edge of Cdelay and the rising edge of Cout is $2t_{delay}$. When the duty cycle of Cout is under 50% (meaning that t_{delay} is less than $T/2$), the input data of D flip-flop is always low and output *Comp* is set to a low value. However, over 50% of the duty cycle of Cout will drive the output of Comp to high.

The DCC is adjusted by the delay time of the delay chain. Using a more precise delay chain could obtain a better DCC adjustment resolution, but needs longer delay chain, which means more chip area, more power dissipation and a longer adjusting time of the DCC. A tradeoff between resolution and area and power dissipation needs to be considered.

3.4. Time-to-digital converter

Since a conventional digital DLL utilizes a sequential search scheme to obtain the control words, the lock time depends on the length of the delay line, and is exponentially proportionate to the number of control bits. The proposed ADDLL adopts a novel time-to-digital converter (TDC) to reduce locking time and to prevent the harmonic locking problem. The proposed TDC in the ADDLL shown in Fig. 7 comprises a pulse generator, a TDC delay chain and an encoder. The TDC delay chain is a rearrangement of the DCDL and it is composed of 4 fine delay units and 15 TDC delay units. The 4 fine delay units generate the intrinsic delay of the DCDL, and each TDC delay unit contains 4 coarse delay cells. According to the transfer characteristic of the PD, the DCDL delay should meet the following requirement in order to prevent false locking:

$$0.5T_{REF} < T_{DCDL} < 1.5T_{REF}, \qquad (5)$$

where T_{REF} is the period of reference clock and T_{DCDL} is the delay of the digital delay line. In a TDC, the period of the reference clock is quantized by TDC delay unit and then converted to a TDC control code to control the coarse delay of the DCDL. The TDC control code guarantees that the DCDL delay meets Eq. (5). The timing diagram of the proposed TDC is shown in Fig. 8. The TDC works as follows: a PULSE_START signal is generated by the pulse generator at the first rising edge of the effective clock. PULSE_START then passes through TDC
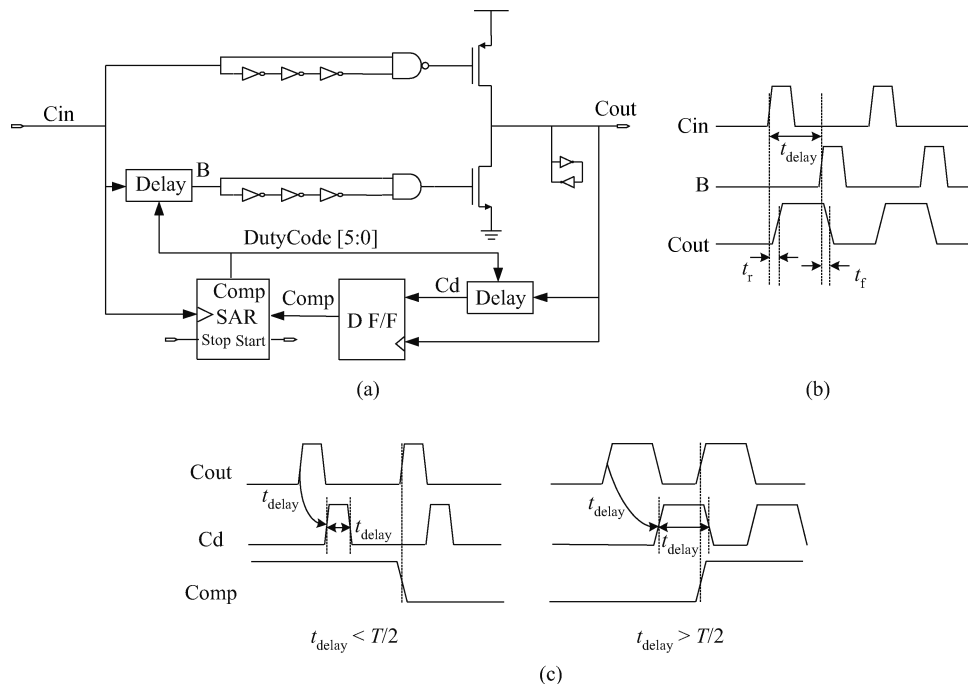
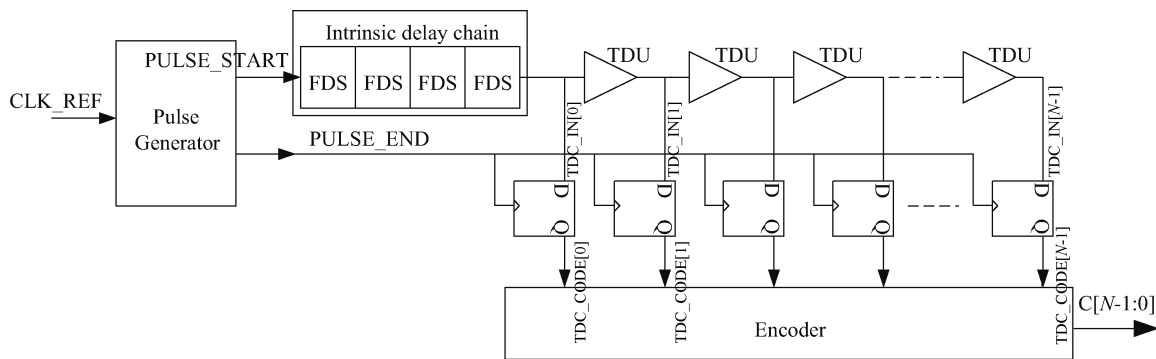Fig. 6. (a) Duty cycle corrector. (b) Output duty cycle. (c) DCC timing diagram.
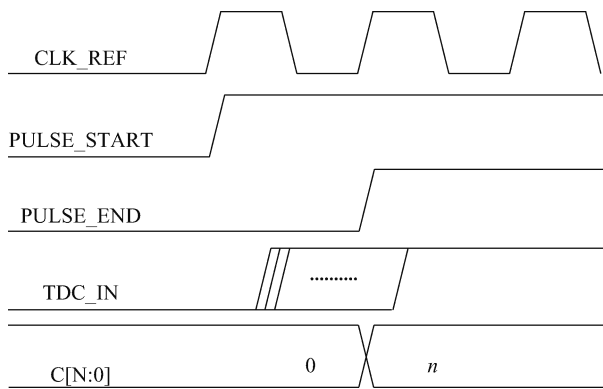

Fig. 7. Time-to-digital converter.

Fig. 8. Timing diagram of TDC.

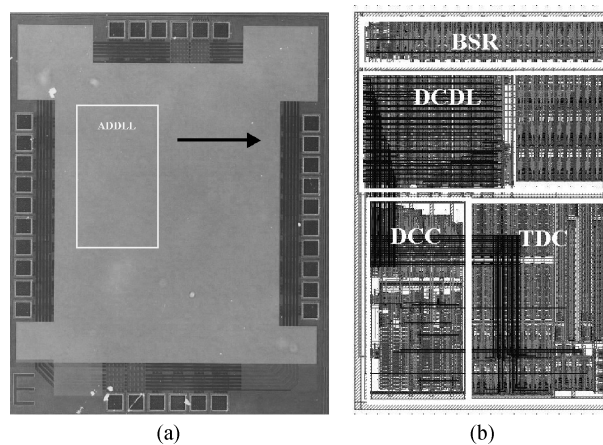(a)   (b)

Fig. 9. (a) Chip microphotograph and (b) layout of the proposed AD-DLL.

delay chain and generates equally spaced signals TDC_IN[N-1:0]. The delay spacing of the generated signals is the same as the coarse delay step size of the DCDL. A PULSE_END signal generated at the second rising edge of the effective clock
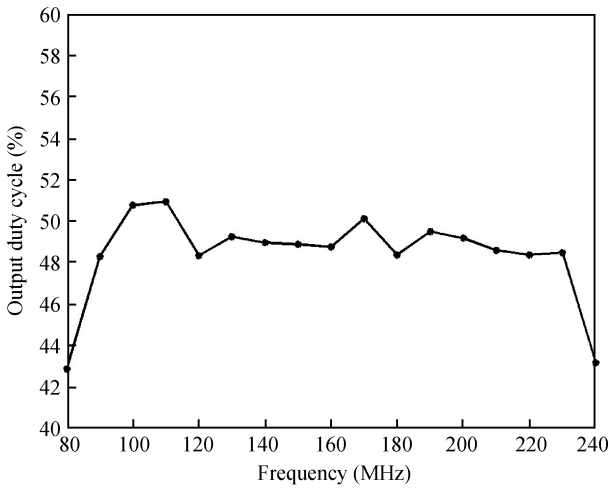
is used to sample TDC_IN[N-1:0] and generates a control code TDC_CODE[N-1:0]. An encoder converts the TDC_CODE[N-
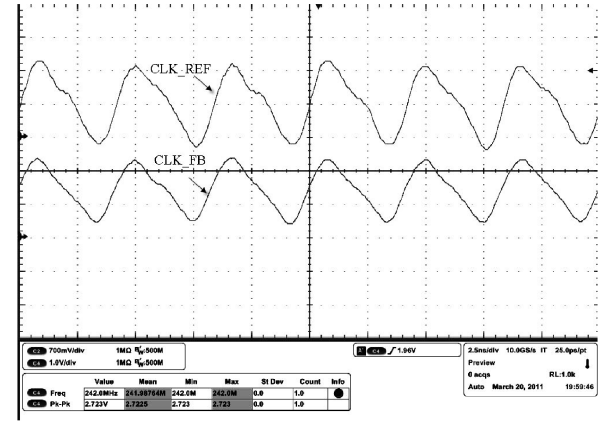
(a)

(a)

(b)

Fig. 10. Output duty cycle of the DCC as function of input frequency.
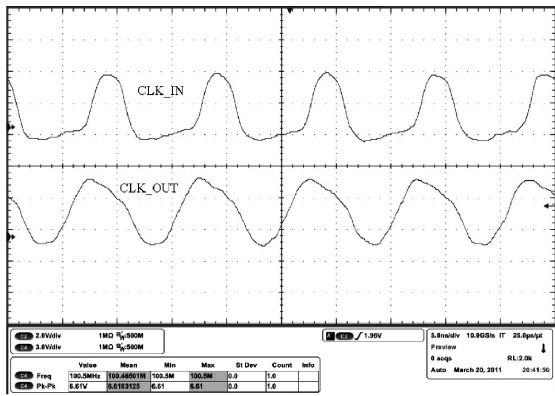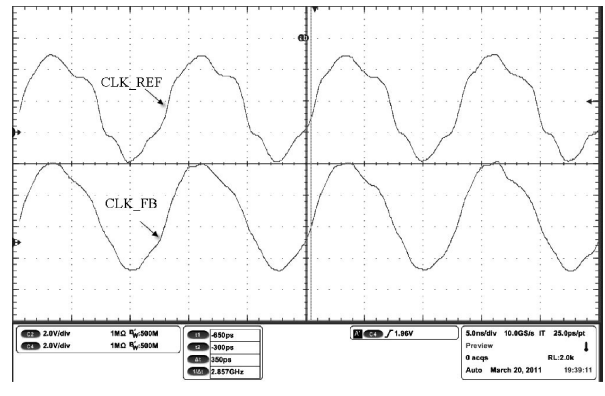
(b)

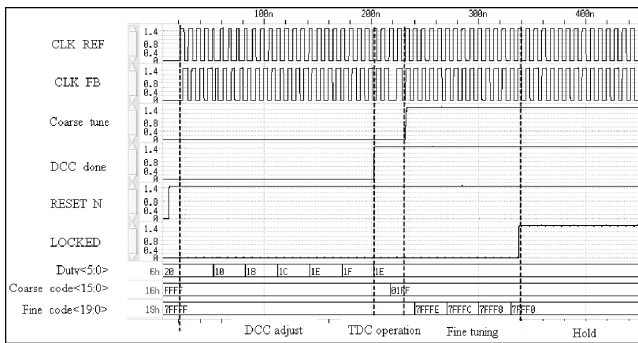Fig. 12. Measurement results at (a) 80 MHz and (b) 240 MHz.

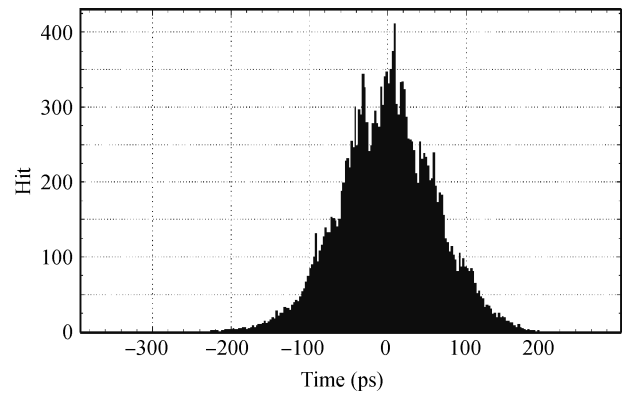Fig. 11. Locking procedure of the proposed ADDLL.

Fig. 13. Measured TIE histogram of ADDLL.

1:0] code into the coarse delay control code of the DCDL C[N-1:0], thus the proposed TDC can turn the ADDLL into a coarse locked state in one clock cycle.

4. Implementation and measurement results

The proposed ADDLL is fabricated in a chartered 0.13 μm CMOS standard process with a 1.5 V supply voltage. The active area of the ADDLL is 0.017 mm^2. The chip microphotograph of the ADDLL and the layout of the core circuits are shown in Fig. 9.

Figure 10(a) shows the measured output duty cycle of the DCC circuit for an input clock of a 25% duty cycle at 100 MHz. The duty cycle of the output clock is 51%. Figure 10(b) shows the output duty cycle of the DCC circuit at different input frequencies. Results show that for an input frequency range of 100 to 200 MHz, the output duty cycle of the DCC is within 48% to 51.5%. The duty cycle error is less than 2%.

The locking procedure of the proposed ADDLL is shown in Fig. 11. The results show that the minimum delay resolution of one fine delay cell is 15 ps, thus the total DCDL delay resolution is 60 ps. The ADDLL takes 6 clock cycles to the finish DCC adjustment, 1 clock cycle to finish the TDC operation and less than 10 clock cycles for fine tuning. Thus the total lock-

Table 1. Performance comparison.

Parameter	Proposed	Ref. [6] (simulation)	Ref. [7]	Ref. [4]	Ref. [5]	Ref. [8]
Process	0.13 μm CMOS	0.13 μm CMOS	0.13 μm CMOS	0.13 μm CMOS	0.18 μm CMOS	0.13 μm CMOS
Supply voltage (V)	1.5	1.2	1.2	1.2	1.8	1.2
Locking time (clock cycles)	< 17	13	N/A	40	80	42
Operation range (MHz)	75–350	200–400	100–200	333.5–800	510–1100	30–1000
Phase error (degree)	2.59	1.3	5.47 (7.6%)	2	N/A	N/A
Power consumption (mW)	1.74 @ 240 MHz	5.5 @ 400 MHz	9 @ 200 MHz	19.2 @ 800 MHz	12 @ 800 MHz	1.5 @ 30 MHz
Delay resolution (ps)	15	4	1.4	10	5.9	10
Active area (mm^2)	0.017	0.026	0.207	0.074	0.023	0.02

Table 2. The measured performance summary of the proposed AD-DLL.

Parameter	Value
Operating frequency range	75–350 MHz
Delay resolution	15 ps
Locking time	< 17 clock cycles
Output duty cycle	48%–51.5%
Power @ 240 MHz	1.74 mW
Input duty cycle range	15%–80%
Active area	0.1 × 0.17 mm^2
Time interval error	60.7 ps

ing time of the ADDLL is less than 17 clock cycles. When the ADDLL is in a locked state, the LOCKED signal is activated to put the ADDLL into power down mode, which reduces the dithering phenomenon and power consumption of the ADDLL.

Figures 12(a) and 12(b) show the output clock signal when the proposed ADDLL is locked at 80 and 240 MHz, respectively. The measured signals in Figs. 12(a) and 12(b) are the reference clock and output clock signal, separately. According to the measurement results, the proposed ADDLL can operate at a tuning range of 75 to 350 MHz.

Figure 13 shows the measured time-interval-error (TIE) histogram of the output signal when the ADDLL operates at 240 MHz. The standard deviation of the TIE of the ADDLL is 60.7 ps. The measured power consumption is 1.74 mW when the ADDLL is locked at 240-MHz clock input.

Table 1 lists comparison results with the state-of-the-art DLLs in DDR controller applications. The measured performance summary of the proposed ADDLL is listed in Table 2. The proposed ADDLL using a time-to-digital converter, shortens the locking time to less than 17 clock cycles. The proposed ADDLL also has the widest operating range and the lowest power consumption compared with other DLL designs.

5. Conclusion

A fast-locking all-digital DLL has been proposed in this paper. By using a novel structure TDC, the locking time of the ADDLL can be reduced to less than 17 clock cycles. The BSR helps to reduce dithering during switching of fine control code. The fabricated chip occupies 0.017 mm^2 active area in 0.13 μm CMOS technology and consumes 1.74 mW from a 1.5 V supply. The proposed ADDLL can be used in the interface of a DDR SDRAM controller for phase shifting.

References

[1] Kim B, Weigandt T C, Gray P R. PLL/DLL system noise analysis for low-jitter clock synthesis design. Proceedings of the International Symposium on ISCAS, 1994, 4: 31

[2] JEDEC Standard, Double Data Rate (DDR) SDRAM specification, JESD79E, May 2005

[3] Yoshimura T, Nakase Y, Watanabe N, et al. A delay-locked loop and 90-degree phase shifter for 100 Mbps double data rate memories. Symposium on VLSI Circuits Digest of Technical Papers, 1998: 66

[4] Bae J H, Seo J H, Yeo H S, et al. An all-digital 90-degree phase-shift DLL with loop embedded DCC for 1.6 Gbps DDR interface. CICC Dig Tech Papers, 2007: 373

[5] Oh K I, Kim L S, Park K I, et al. Low-jitter multi-phase digital DLL with closest edge selection scheme for DDR memory interface. Electron Lett, 2008, 44(19): 1121

[6] Sheng D, Chung C C, Lee C Y. Fast-lock all-digital DLL and digitally-controlled phase-shifter for DDR controller applications. IECIE Electronics Express, 2010, 7(9): 634

[7] Chung C C, Chen P L, Lee C Y. An all-digital delay-locked loop for DDR SDRAM controller applications. Symposium on VLSI Design, Automation and Test, 2006: 1

[8] Wang L, Liu L, Chen H. An implementation of fast-locking and wide-range 11-bit reversible SAR DLL. IEEE Trans Circuits Syst II: Express Briefs, 2010, 57(6): 421

[9] Jeon Y J, Lee J H, Lee H C, et al. A 66–333-MHz 12-mW register-controlled DLL with a single delay line and adaptive-duty-cycle clock dividers for production DDR SDRAMs. IEEE J Solid-State Circuits, 2004, 39(11): 2087

[10] Alvandpour A, Krishnamurthy R K, Eckerbert D, et al. A 3.5 GHz 32 mW 150 nm multiphase clock generator for high-performance microprocessors. IEEE International Solid-State Circuits Conference Digest of Technical Papers, 2003: 112

[11] Wang J S, Wang Y M, Chen C H, et al. An ultra-low-power fast-lock-in small-jitter all-digital DLL. IEEE International Solid-

State Circuits Conference Digest of Technical Papers, 2005: 422

[12] Yang R J, Liu S I. A 40–550 MHz harmonic-free all-digital delay-locked loop using a variable SAR algorithm. IEEE J Solid-State Circuits, 2007, 42(2): 361

[13] Dehng G K, Hsu J M, Yang C Y, et al. Clock-deskew buffer using a SAR-controlled delay-locked loop. IEEE J Solid-State Circuits,

2000, 35(8):1128

[14] Rossi A, Fucili G. Nonredundant successive approximation register for A/D converters. Electron Lett, 1996, 32: 1055

[15] Maymandi-Nejad M, Sachdev M. A digitally programmable delay element: design and analysis. IEEE Trans VLSI Syst, 2003, 11(5): 871