A 5.3-GHz Programmable Divider for HiPerLAN in 0.25-μm CMOS

Nagendra Krishnapura and Peter R. Kinget, *Member, IEEE*

***Abstract*—A 5.3-GHz low-voltage CMOS frequency divider whose modulus can be varied from 220 to 224 is presented. Programmability is achieved by switching between different output phases of a D-flip-flop (DFF). An improved glitch-free phase switching architecture through the use of retimed multiplexer control signals is introduced. A high-speed low-voltage DFF circuit is given. The programmable divider fabricated in 0.25-μm technology occupies 0.09 mm^2; it consumes 17.4 mA at 1.8 V and 26.8 mA at 2.2 V. Operation of 5.5 GHz with 300-mV$_{\rm pk}$ single-ended input is achieved with a 2.2-V supply. The residual phase noise at the output is −131 dBc/Hz at an offset of 1 kHz from the carrier while operating from a 5.5-GHz input.**

***Index Terms*—Frequency divider, phase switching, prescaler, retiming.**

I. Introduction

HIPERLAN is a wireless data networking standard that operates in the 5.3-GHz band and consists of five channels separated by 23.5294 MHz. Fig. 1 shows the carrier synthesizer used in a HiPerLAN transceiver. The programmable divider presented in this paper is intended to be used in the phase-locked loop shown in Fig. 1. Several requirements that the divider needs to satisfy are marked on the figure. Its division factor has to be programmable from 220 to 224 in order to generate the required carriers and operate from a high input frequency of 5.3 GHz. The input sensitivity of the divider must be about 300 mV$_{\rm pk}$ or less, since that is what can be reasonably expected from an on-chip oscillator fabricated in a 0.25-μm CMOS technology and operating from a low supply voltage of 1.8 V.

In Section II, we discuss phase switching, which is the chosen architecture for this programmable divider. In Section III, we discuss a potential problem with existing phase switching circuits and introduce a retimer block that eliminates this problem. Section IV deals with the high-speed divide-by-2 ($\div 2$) stage. Section V discusses the overall circuit and some implementation details. Experimental results are given in Section VI.

II. Architecture

Conventionally, programmable dividers are implemented using a high-speed dual-modulus prescaler along with low-speed programmable counters, which implement arbitrary division factors by "swallowing" pulses [1]. In [2], an asynchronous divider is presented in which pulse swallowing is accomplished by switching between different output phases of a $\div 2$ stage implemented using a master–slave flip-flop. In this scheme, only one flip-flop (the first $\div 2$) stage operates at full speed ($f_{\rm in}$), as opposed to several flip-flops as in a dual-modulus prescaler. Additionally, the elimination of high-speed feedback loops around multiple flip-flops that would be present in a dual-modulus prescaler and the reduced load on the voltage-controlled oscillator (VCO) result in a higher maximum speed of operation and lower power consumption in a given technology. This architecture was therefore chosen for our 5.3-GHz programmable divider. Since programmable division using phase switching [2] is a relatively new technique, we describe it in some detail below. On a related note, [3] presents a circuit in which multiphase clocks are used to generate periodic outputs with an arbitrary pulse pattern in each period.

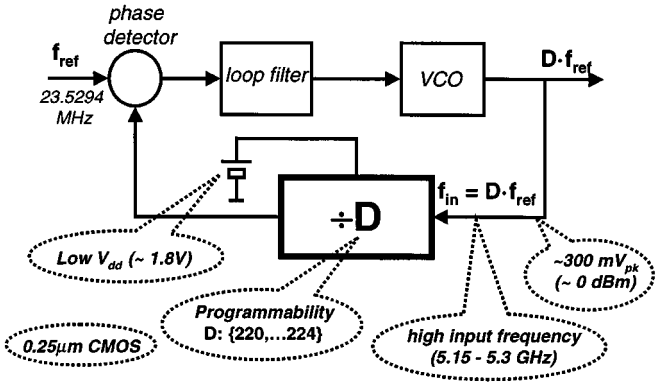


Fig. 1. Carrier synthesizer for HiPerLAN.

The block diagram of the phase switching divider is shown in Fig. 2. A quarter cycle (90°) at $f_{\rm in}/4$ is the same duration as a full cycle (360°) at $f_{\rm in}$. A cascade of two $\div 2$ stages using master–slave flip-flops has four outputs (X, Y, XB, YB) whose rising edges are separated by 90° (Fig. 2, waveforms in Fig. 3). At any instant, only one of these outputs is connected to the subsequent $\div N$ stage through a multiplexer (MUX), as shown in Fig. 2. The MUX output is labeled OUT in Fig. 3. In order to swallow a cycle and augment the division factor of the frequency divider by one, the input of the $\div N$ stage is switched to a waveform that is lagging the current waveform by 90° (e.g., from X to Y in Fig. 2). The arrows in Fig. 3 indicate these transitions. As shown, the MUX output is switched to successively lagging phases to obtain a $\div 5$ output. The thick lines in each waveform indicate the periods when that particular

Manuscript received December 20, 1999; revised February 6, 2000.

N. Krishnapura was with Bell Laboratories, Lucent Technologies, Murray Hill, NJ 07974 USA. He is now with Columbia University, New York, NY 10027 USA (e-mail: nagi@cisl.columbia.edu).

P. R. Kinget was with Bell Laboratories, Lucent Technologies, Murray Hill, NJ 07974 USA. He is now with Broadcom Corporation, Irvine, CA 92619 USA.

Publisher Item Identifier S 0018-9200(00)03881-6.

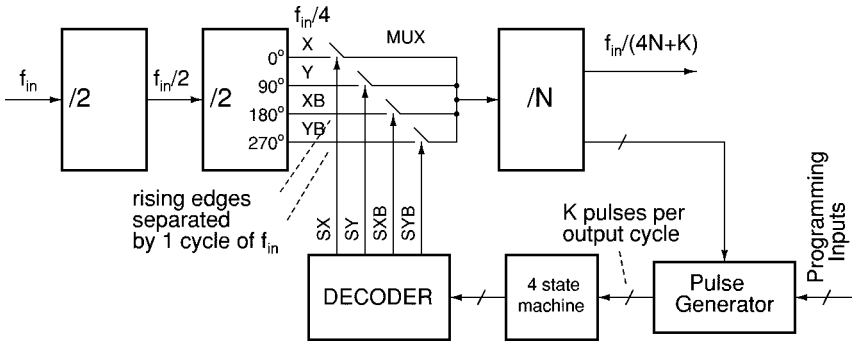


Fig. 2. Programmable frequency divider employing phase switching.

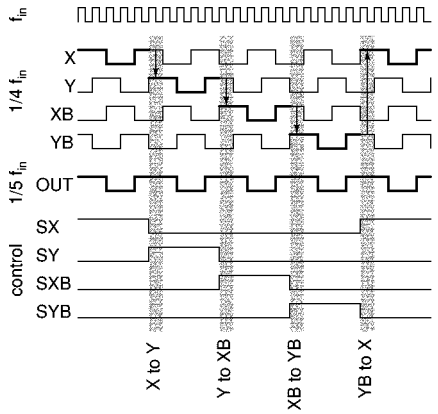


Fig. 3. Waveforms illustrating phase switching.

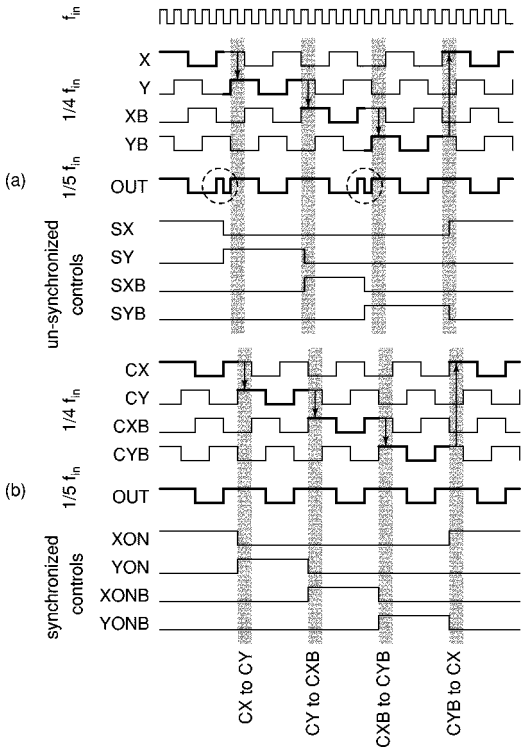


Fig. 4. (a) Glitches during phase switching due to improper signal timing. (b) Glitch elimination by retiming the MUX control signals.

signal is directed to the MUX output. SX, SY, SXB, SYB in Figs. 2 and 3 are the MUX control signals.

For an arbitrary division factor, input cycles can be swallowed by changing the control inputs of the MUX appropriately. In the absence of phase switching, the divider chain in Fig. 2 has a division factor of $4N$. If the phases are switched K times in each output cycle of the divider chain, K input cycles are swallowed, and consequently, the division factor is augmented by K and becomes $4N + K$. For the case illustrated in Fig. 3, $N = K = 1$. By varying K, programmability is achieved. The pulse generator (Fig. 2) generates K pulses per output cycle, where K is set by the programming inputs. A $\div 4$ counter can be used as a four-state machine (Fig. 2) that is clocked by these pulses and cycles through four states. Each state corresponds to one of the four possible connections in the MUX. A decoder decodes the state and turns the appropriate switch ON (through one of SX, SY, SXB, SYB) in the MUX.

III. Glitch Free Phase Switching

A potential problem with the architecture of Fig. 2 is shown in Fig. 4(a). The input clock, the four waveforms (X, Y, XB, YB) differing by 90°, and the four control signals SX, SY, SXB, SYB are shown. Compared to Fig. 3, the rising edge of SY arrives earlier. Consequently, the transition from X to Y occurs when X is high and Y is still low, resulting in a glitch as marked on the waveform OUT. Another glitch due to the premature arrival of SYB is also indicated on the figure. These extra transitions cause the divider to miscount, which in turn would result in an erroneous output frequency in the synthesizer. The timing window for glitch-free switching is shown shaded in Fig. 3 and the top part of Fig. 4(a) for each of the four transitions (marked with arrows). The circuit controlling the MUX must enforce this correct timing.

If no precautions are taken, glitches are almost certain to occur for the following reasons.

1) The delay of the divider and the logic circuit in the feedback path ($/N$ stage, pulse generator, four-state machine, and decoder in Fig. 2) that provide the phase switching signals cannot be determined accurately over process and temperature variations.
2) As mentioned earlier, X must be changed to Y only when $X \cdot Y = 1$. Y must be changed to XB only when $XB \cdot Y = 1$, and so on [see the shaded regions in Fig. 4(a)]. So the correct "windows" are different for each of the transitions in the MUX.

In a previous implementation, glitches were avoided by using a long rise time for the MUX control signals [2], which is not

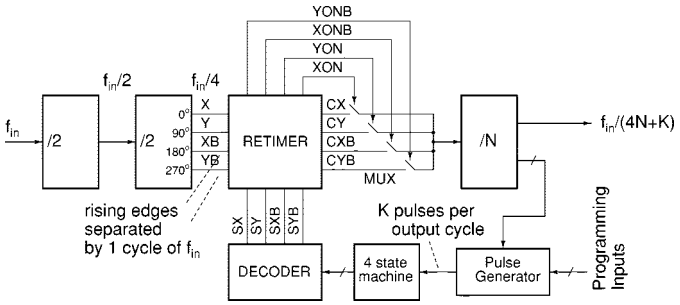


Fig. 5. Programmable divider employing glitch-free phase switching.

a robust solution; or by using feedback from the output of the MUX [4], which limits the speed due to the delay in the feedback loop.

Here we describe a retimer circuit that takes as its input the four clocks X, Y, XB, YB and the control signals SX, SY, SXB, SYB (decoder outputs) and generates four clocks CX, CY, CXB, CYB and corresponding control signals XON, YON, $XONB$, $YONB$ whose transitions are synchronized to the respective clocks so that glitches are eliminated. This retimer is inserted before the MUX, as shown in Fig. 5. Consistent with the high speed requirement, the retimer operates in a feed-forward fashion as described below.

The logic diagram and a pseudo-nMOS implementation of the retimer are shown in Fig. 6. The output of a cross-coupled latch (YON) directs CY to the output of the MUX. The cross-coupled latch is controlled by gated inputs such that the timing "zones" marked on Figs. 3 and 4 are enforced. YON is pulled high if both X and Y are high and SY is high (i.e., the decoder selects Y). YON is pulled low when both Y and XB are high and SXB is high (the decoder selects XB). In order to have both the clocks CX, CY, CXB, CYB) and the control signals (XON, YON, $XONB$, $YONB$) arrive synchronously at the MUX inputs, a buffer whose delay is the same as that of the control signal generator is used on the clock line (lower part of Fig. 6). The waveforms in Fig. 4(b) illustrate its operation. The new control signals (XON, YON, $XONB$, $YONB$) are synchronized to the new clocks (CX, CY, CXB, CYB) even though both are skewed from the original clocks and controls depicted in Fig. 4(a). SY goes high when Y is still low. But due to YON's being synchronized to CY, a glitch is avoided (OUT in the lower part of Fig. 4).

For each of the other three clocks (X, XB, and YB), identical circuits are used that generate the control signal for a particular clock in the proper "window."

A pseudo-nMOS implementation of the retimer is shown in Fig. 6(b). From simulations, this circuit is found to work reliably with variations in process and temperature. The arrival of the clocks and the control signals at the inputs of the MUX is found to be practically coincident in all cases.

IV. Divide-by-Two Stage

The block most difficult to design is the first ÷2 stage, which should operate at 5.3 GHz or more. Fig. 7 shows some published latches intended for high-speed operation. Both a conventional CMOS latch and a single-phase latch [5] [Fig. 7(a)] are too slow

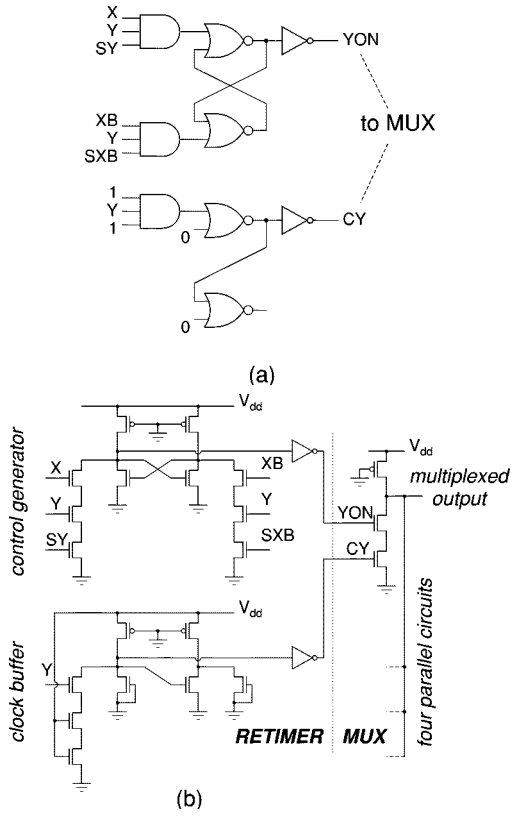


Fig. 6. Retiming circuit used to achieve glitch-free phase switching: (a) Logic and (b) pseudo-nMOS implementation.

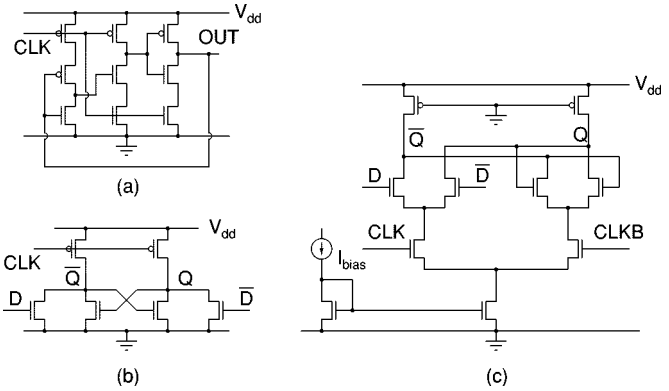


Fig. 7. (a) True single-phase ÷2 stage [5], (b) latch proposed in [6], and (c) latch using source-coupled logic.

for our purposes because they have a large input capacitance due to the parallel connection of pMOS and nMOS gates. Due to its lower mobility and larger threshold voltage (in our case, $V_{\mathrm{TN}} = 0.6$ V and $V_{\mathrm{TP}} = 0.9$ V), the pMOS transistor contributes little to the current drive and much to the capacitances, considerably slowing down the circuit. The latch proposed in [6] [Fig. 7(b)] uses pMOS transistors in the clock path and was found to work only up to ≈2 GHz in our technology. Also, the 25% duty cycle of the output signals is less convenient for phase switching. The source-coupled latch (Fig. 7(c), e.g., [2]) has a reduced output swing that facilitates high speed, but due to the stacking of many devices it cannot be accommodated in the intended low supply voltage.

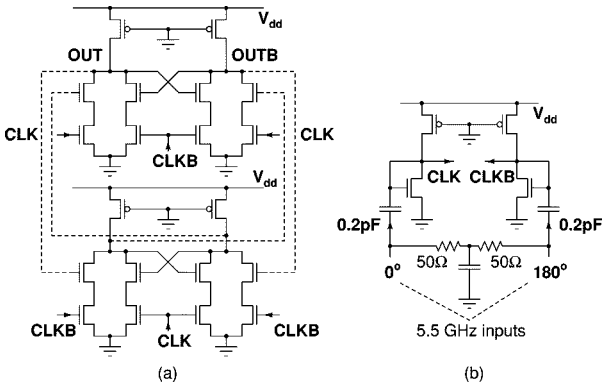


Fig. 8. (a) Pseudo-nMOS divide-by-two stage and (b) input ac coupling.

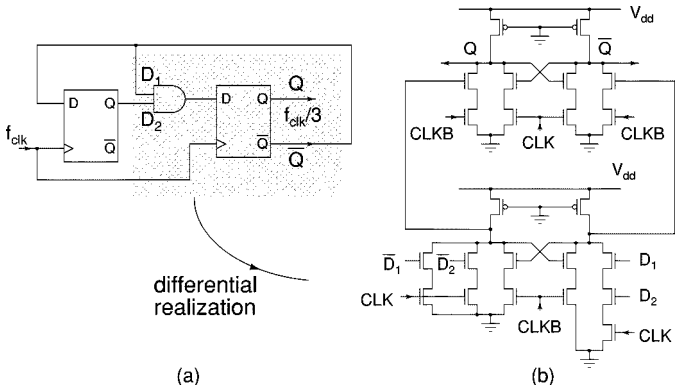


Fig. 9. Divide-by-three stage: (a) logic and (b) implementation.

Using pseudo-nMOS gates enables high-speed operation while providing large output swing. For comparison, we observe that in this technology, with a 1.8-V supply, a three-stage CMOS ring oscillator oscillates at 2.5 GHz, whereas a three-stage pseudo-nMOS ring oscillator oscillates at 6 GHz. This led to our choice of pseudo-nMOS logic despite its high power consumption.

Fig. 8(a) shows a pseudo-nMOS D-flip-flop (DFF) whose outputs are connected back to its inputs (shown in dashed lines) to form a ÷2 stage. NAND gates are used to form the latch since they enable a compact layout where node parasitics can be minimized. In order to have the inputs vary around the switching threshold of the gates, CLK and $CLKB$ are ac coupled through 0.2-pF capacitors. An inverter whose input and output are tied together [Fig. 8(b)] biases the DFF inputs to the correct dc level over process and temperature. A DFF whose inputs are biased near the switching threshold self-oscillates in the absence of an input signal. A disable mode in which $CLK, CLKB$ inputs to the ÷2 stage are tied to the opposite supply rails prevents self-oscillations. This mode is used if and when the synthesizer needs to be turned off.

From simulations, the ÷2 circuit (extracted from layout) was found to operate satisfactorily over process and temperature variations with 300 mV_{pk} inputs (single ended) at 5.5 GHz while operating from a 1.8-V supply and driving the second ÷2 stage.

V. The Complete Programmable Divider

Fig. 5 shows the block diagram of the complete programmable divider. The division factor in absence of phase switching is 216 ($4N = 2^3 \times 3^3$) and is implemented as a cascade of three ÷2 and three ÷3 stages. The ÷2 stages use the circuit shown in Fig. 8(a). Successive stages use smaller transistors, since they are operating at a reduced speed. Pseudo-nMOS inverters are used for interstage buffering in the divider chain.

A ÷3 stage needs two DFF's, one of which has AND gated inputs. The logic diagram is shown in Fig. 9(a). The combination of the AND gate and the flip-flop is implemented as shown in Fig. 9(b). The parallel branches implement the AND function.

To realize division factors from 220 to 224, four to eight pulses are required, and they are derived from the outputs of the last three ÷3 stages. Combinational logic is used to obtain the desired number of pulses from the three programming bits. The retimer circuit is shown in Fig. 6. The four-state counter, decoder, and MUX shown in Fig. 5 are pseudo-nMOS implementations. Pseudodifferential logic stages are used throughout the divider. The output is measured single-ended, for convenience. For the sake of measurement, a string of successively larger CMOS inverters is used to buffer the output. The buffer can provide about 0.6 V_{pp} across a 50-Ω load. Broad-band matching for the clock inputs is provided by on-chip ac-coupled 50-Ω termination resistors.

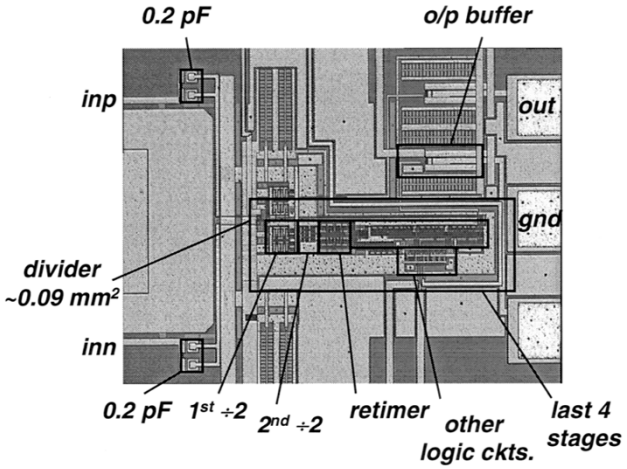


Fig. 10. Chip photograph.

Fig. 10 shows the photograph of the chip. The different stages of the divider are marked on the figure. The entire test chip measures 0.9 mm^2 out of which the programmable divider (excluding the buffers) occupies 0.09 mm^2. Separate supply pins are used for the output buffers. On-chip bypassing is provided using MOS capacitors.

VI. Measurement Results

The chips were tested by probing on wafer as well as by wirebonding the dice to a printed circuit board. The differential inputs were provided using a hybrid. The output was monitored on an oscilloscope and a spectrum analyzer.

Fig. 11(a) shows the minimum input amplitude required for proper operation with three supply voltages (1.8, 2.0, and 2.2 V). The curves are qualitatively similar but shifted to the right

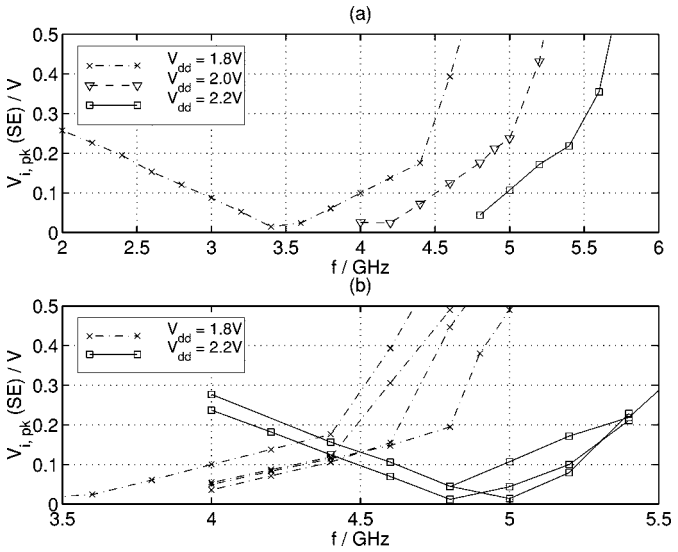


Fig. 11. (a) Divider sensitivity versus input frequency at different supply voltages. (b) Several samples with $V_{dd} = 1.8$ V and 2.2 V.

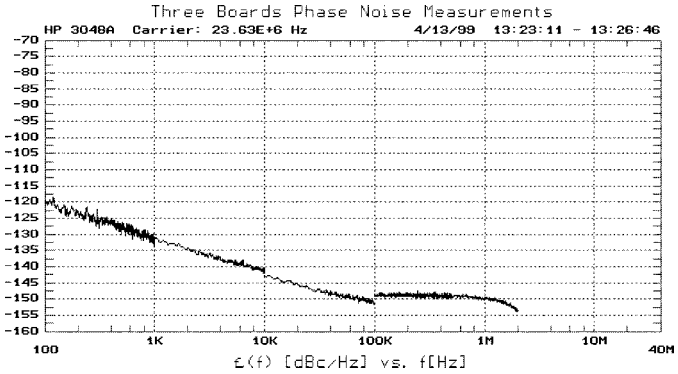


Fig. 12. Sum of residual phase noise o/p from two dividers.

with increasing supply voltage. Assuming 300-mV peak (single ended) to be a reasonable upper limit for the drive available from an on-chip VCO, the corresponding upper limit on the frequency can be ascertained from Fig. 11(a). Operation of 5.5 GHz is possible with a supply voltage of 2.2 V. Operation at 5.5 GHz of 1.8 V (with a 300-mV peak input) was expected from simulations over process and temperature variations but was not achieved because some of the parameters of the experimental process shifted severely after submission of the layout. Measurement of several chips across the wafer showed a consistent performance, as can be seen from Fig. 11(b). Consistent performance over all possible division factors was also verified.

The current drawn from the supply varies slightly with input frequency and division factor. The divider draws a maximum[1] of 17.4, 21.7, and 26.8 mA, respectively, with supply voltages of 1.8, 2.0, and 2.2 V.

For residual phase noise measurement, two dividers were run from a common input clock. The outputs were fed to the phase detector in an HP3048A phase noise analyzer. A phase shifter was used at the output of one of the dividers. There is a phase difference of 90° between the two inputs of the phase detector. The

[1]The current consumption reported in [7] included the current consumed in an extra DFF that was fabricated along with the divider.

TABLE I
SUMMARY OF MEASURED PERFORMANCE

Technology	0.25μm CMOS	
Chip area	0.09 mm^2	
V_{dd}	2.2 V	1.8 V
$I(V_{dd})$	26.8 mA[1]	17.4 mA[1]
$f_{in,\,max}$	5.5 GHz	4.5 GHz
Sensitivity	300 mVpk., SE	300 mVpk., SE
o/p phase noise	-131 dBc / Hz	-133 dBc / Hz
(with $f_{in} = f_{in,\,max}$)	@ 1 kHz offset	@ 1 kHz offset

TABLE II
COMPARISON TO OTHER PUBLISHED DIVIDERS

		Tech.	$f_{in,\,max}$ GHz	V_{dd} V	P_d mW	sens. V_{pk}	phase noise dBc/Hz @ 1 kHz input ref.
This work	÷220 to ÷224	0.25μm	5.5	2.2	59[1]	0.3	-83.2
[9]	÷8/9	0.7μm	1.5	5.0	55	0.16	-93.9
[10]	÷4	0.15μm	11.8	2.0	20	1.0	—
[2]	÷128	0.7μm	1.7	3.0	24	—	-87.9
[6]	÷2	0.1μm	13.4	2.6	26	1.3	—
[11]	÷16	1.2μm	1.5	5.0	13	0.35	—
[12]	÷4/5	0.4μm	4.2	3.5	—	0.5	—
[13]	÷256/258	0.2μm GaAs	14.5	0.6	22	1.0	—

measured noise is twice the output noise of each of the dividers (assuming identical dividers), and the jitter in the input clock is canceled. Details of this measurement technique can be found in [8]. The output phase noise of an otherwise noiseless synthesizer using a noisy divider is equal to the *input-referred* phase noise of the divider (inside the loop bandwidth). The input-referred phase noise is the square of the division factor times the output phase noise. In this case, it is 47 dB [$=20\log(224)$] above the output phase noise.

Fig. 12 shows the results of the measurement. The residual phase noise at the output is shown. This consists of phase noise added by two dividers and the respective on-chip output buffers. The sum of the residual phase noise of the two dividers is -131 dBc/Hz at 1 kHz offset. As can be seen from Fig. 12, 10 dB/decade behavior is maintained down to very low offset frequencies. The absence of $1/f^3$ regions in the residual phase noise is good for the overall synthesizer since the divider contributes to the synthesizer's jitter only at low offset frequencies that are inside the PLL's bandwidth.

The phase noise measurement setup outside the chip had a noise floor of about -155 dBc/Hz, which is well below the measured value shown in Fig. 12. However, crude simulations of the on-chip buffers used to drive the 50-Ω load indicated that their noise could be large enough to interfere with the measurement, in which case the divider's phase noise could be below the value shown in Fig. 12. Unfortunately, at this time, we do not have a

new chip without output buffers to verify by measurement if the output buffer's noise is indeed masking the noise from the divider.

The measured performance of the divider is summarized in Table I. Table II compares this divider to several published frequency dividers (all CMOS, except the last one). The fabricated stage, maximum operating frequency, supply voltage, power dissipation, input sensitivity, and residual phase noise are shown in each case. For purposes of comparison, the residual phase noise is shown referred to the input at an offset frequency of 1 kHz from the carrier. Good sensitivity and noise performance are achieved at a low supply voltage.

VII. Conclusion

A high-speed programmable divider operating at a low supply voltage is presented. A DFF capable of high speed operation is described. A retiming technique for reliable implementation of phase switching programmable division is given. Measurement results demonstrate the high-speed capability and programmability of the presented circuits. A good noise performance compared to other published circuits is achieved while operating at a high frequency from a low supply voltage.

Acknowledgment

The authors would like to thank W. Fischer for his help with the layout of the chip and V. Boccuzzi for the excellent help with the measurements. The authors would also like to thank A. Dunlop's support and the loan of equipment by M. Banu, R. C. Melville, and H. Wang.

References

[1] U. Rohde, *Digital PLL Frequency Synthesizers: Theory and Design*. Englewood Cliffs, NJ: Prentice-Hall, 1983.

[2] J. Craninckx and M. S. J. Steyaert, "A 1.75-GHz/3-V dual-modulus divide-by-128/129 prescaler in 0.7-μm CMOS," *IEEE J. Solid-State Circuits*, vol. 31, no. 7, pp. 890–897, July 1996.

[3] F. Mu, *et al.*, "Digital multiphase clock/pattern generator," *IEEE J. Solid-State Circuits*, vol. 34, pp. 182–191, Feb. 1999.

[4] M. H. Perrott, "Techniques for high data rate modulation and low power operation of fractional-N frequency synthesizers," Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, Sept. 1997.

[5] J. Yuan and C. Svensson, "High-speed CMOS circuit technique," *IEEE J. Solid-State Circuits*, vol. 24, pp. 62–70, Feb. 1989.

[6] B. Razavi, *et al.*, "Design of high-speed, low-power frequency dividers and phase locked loops in deep submicron CMOS," *IEEE J. Solid-State Circuits*, vol. 30, pp. 101–109, Feb. 1995.

[7] N. Krishnapura and P. Kinget, "A 5.3 GHz programmable divider for HiPerLAN in 0.25 μm CMOS," in *Proc. 1999 Eur. Solid State Circuits Conf.*, 1999, pp. 144–147.

[8] W. F. Egan, "Modeling phase noise in frequency dividers," *IEEE Trans. Ultrason., Ferroelect., Freq. Contr.*, vol. 37, pp. 307–315, July 1990.

[9] B. De Muer and M. Steyaert, "A single ended 1.5 GHz 8/9 dual modulus prescaler in 0.7μm CMOS technology with low phase noise and high input sensitivity," in *Proc. 1998 Eur. Solid State Circuits Conf.*, 1998, pp. 256–259.

[10] M. Kurisu, *et al.*, "An 11.8 GHz 31-mW CMOS frequency divider," in *1997 Symp. VLSI Circuits Dig. Tech. Papers*, 1997, pp. 73–74.

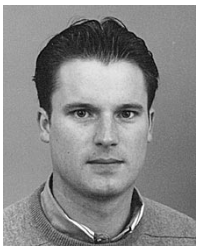
[11] N. Foroudi and T. Kwasniewski, "CMOS high-speed dual-modulus frequency divider for RF frequency synthesis," *IEEE J. Solid-State Circuits*, vol. 30, pp. 93–100, Feb. 1995.

[12] H.-I. Cong, *et al.*, "A 2 GHz CMOS dual-modulus prescalar IC," *IEEE J. Solid-State Circuits*, vol. 23, pp. 1189–1194, Oct. 1988.

[13] T. Maeda, *et al.*, "An ultra low power consumption high-speed GaAs 256/258 dual-modulus prescaler IC," in *Proc. 1997 IEEE GaAs IC Symp.*, 1997, pp. 175–178.

Nagendra Krishnapura received the B.Tech. degree in electronics and communications engineering from the Indian Institute of Technology, Madras, in 1996. He is currently pursuing the Ph.D. degree in electrical engineering at Columbia University, New York.

He was with Bell Laboratories, Lucent Technologies, Murray Hill, NJ, during the summer of 1998. His research interests are analog and RF circuits.



Peter R. Kinget (M'96) received the engineering degree in electrical and mechanical engineering and the Ph.D. degree in electrical engineering from the Katholieke Universiteit Leuven, Belgium, in 1990 and 1996, respectively.

From 1991 to 1996, he was a Research Assistant at the ESAT-MICAS Laboratory, Katholieke Universiteit Leuven. From October 1996 to July 1999, he was with Bell Laboratories, Lucent Technologies, in Murray Hill, NJ, as a Member of Technical Staff in the Design Principles Department. Since July 1999, he has been with Broadcom Corp., Irvine, CA, as a Senior Staff Scientist in the Analog and RF Microelectronics group. His interests are in analog and RF integrated circuits.

Dr. Kinget received a fellowship from the Belgian National Fund for Scientific Research from 1991 to 1995. He is a member of the Technical Program Committee of the IEEE Custom Integrated Circuits Conference.