

TABLE VI
ESTIMATED AREA

(mm^2)	EX1	EX2	EX3	EX4	EX5	EX6
FASIDC	0.945	1.838	2.612	3.510	4.253	7.076
SIDC	1.102	1.761	2.987	3.234	4.481	6.694

TABLE VII
RUNTIME (SEC)

(sec)	EX1	EX2	EX3	EX4	EX5	EX6
FASIDC	18.00	40.00	71.00	92.00	194.00	397.00
SIDC	18.00	40.00	71.00	92.00	194.00	397.00

the nearest resource in floorplan, we can reduce interconnect delay of the critical-path. Also, by balancing fan-outs of the reused components, performance of the filters are improved. Floorplan and architecture synthesis are performed simultaneously to achieve this goal. Compared to the traditional SIDC approach, floorplan-aware SIDC achieves 15% average reduction in critical-path delay.

REFERENCES

- [1] H. Samuelli, "An improved search algorithm for the design of multiplierless FIR filters with powers-of-two coefficients," *IEEE Trans. Circuits Syst.*, vol. 36, no. 7, pp. 1044–1047, Jul. 1989.
- [2] Y. C. Lim and S. R. Parker, "FIR filter design over a discrete powers-of-two coefficient space," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-31, no. 3, pp. 583–591, Jun. 1983.
- [3] A. Nishihara, M. Yagyu, and N. Fujii, "Fast FIR digital filter structures using minimal number of adders and its application to filter design," *IEICE Trans. Fundam.*, vol. E79-A, no. 8, pp. 1120–1128, Aug. 1996.
- [4] R. I. Hartley, "Subexpression sharing in filters using canonic signed digital multipliers," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 43, no. 10, pp. 677–688, Oct. 1996.
- [5] M. B. Srivastava, M. Potkonjak, and A. P. Chandrakasan, "Multiple constant multiplications: Efficient and versatile frame-work and algorithms for exploring common subexpression elimination," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 15, no. 2, pp. 151–165, Feb. 1996.
- [6] R. P. Pasko, P. Schaumont, V. Derudder, S. Vernalde, and D. Durackova, "A new algorithm for elimination of common subexpressions," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 18, no. 1, pp. 58–68, Jan. 1999.
- [7] A. G. Dempster and M. D. Macleod, "Use of minimum adder multiplier blocks in FIR digital filters," *IEEE Trans. Circuits Syst.*, vol. 42, no. 9, pp. 569–577, Sep. 1995.
- [8] I. Park and H. Kang, "Digital filter synthesis based on minimal signed digit representation," in *Proc. Des. Autom. Conf.*, 2001, pp. 468–473.
- [9] K. Muhammad and K. Roy, "A graph theoretic approach for synthesizing very low-complexity high-speed digital filters," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 21, no. 2, pp. 204–216, Feb. 2002.
- [10] —, "A graph theoretic approach for design and synthesis of multiplierless FIR filters," in *Proc. 12th Int. Symp. Syst. Synthesis*, Nov. 1999, pp. 94–99.
- [11] H. Choo, K. Muhammad, and K. Roy, "MRPF: An architectural transformation for synthesis of high-performance and low-power digital filters," in *Proc. Des., Automat. Test Europe Conf. Exhibition*, vol. 203, Mar. 2003, pp. 700–705.
- [12] —, "Complexity reduction of digital filters using shift inclusive differential coefficients," *IEEE Trans. Signal Process.*, vol. 52, no. 6, pp. 1760–1772, Jun. 2004.
- [13] J. D. Meindl, "Interconnect limits on gigascale integration," *Elec. Perform. Electron. Packag.*, pp. 25–27, Oct. 1999.
- [14] V. G. Moshnyaga, H. Mori, H. Onodera, and K. Tamaru, "Layout-driven module selection for register-transfer synthesis of sub-micron ASICs," in *IEEE/ACM Int. Conf. Comput.-Aided Des., Dig. Tech. Papers*, Nov. 1993, pp. 100–103.

- [15] Y. Chen, W. K. Tsai, and F. J. Kurdahi, "Layout driven logic synthesis system," *Inst. Elect. Eng. Proc. Circuits, Devices Syst.*, vol. 142, pp. 158–164, Jun. 1995.
- [16] H. M. Murata, K. Fujiyoshi, S. Nakatake, and V. Kajitani, "VLSI module placement based on rectangle-packing by the sequence-pair," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 15, no. 12, pp. 1518–1524, Dec. 1996.
- [17] X. Tang and D. F. Wong, "Fast-SP: A fast algorithm for block placement based on sequence pair," in *Proc. ASP-DAC*, Feb. 2001, pp. 521–526.
- [18] Y. Fang and D. Wong, "Simultaneous functional-unit binding and floorplanning," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Des. I, Nov.* 1994, pp. 317–321.
- [19] J. Cong and Z. Pan, "Interconnect delay estimation models for logic and high level synthesis," in *Proc. Asia South Pacific Des. Automat. Conf.*, Jan. 1999, pp. 97–100.
- [20] *Pathmill, Version 2002.03*, Synopsys Inc., Allentown, PA.

A Low-Power Correlation-Derivative CMOS VLSI Circuit for Bearing Estimation

Pedro Julián, Andreas G. Andreou, and David H. Goldberg

Abstract—We present a CMOS integrated circuit (IC) for bearing estimation in the low-audio range that performs a correlation derivative approach in a 0.35- μm technology. The IC calculates the bearing angle of a sound source with a mean variance of one degree in a 360° range using four microphones: one pair is used to produce the indication and the other to define the quadrant. An adaptive algorithm decides which pair to use depending on the direction of the incoming signal, in such a way to obtain the best estimate. The IC contains two blocks with 104 stages each. Every stage has a delay unit, a block to reduce the clock speed, and a 10-bit UP/DN counter. The IC measures 2 mm by 2.4 mm, and dissipates 600 μW at 3.3 V and 200 kHz. It is purely digital and uses a one-bit quantization of the input signals.

Index Terms—Correlation, CMOS digital integrated circuits, direction of arrival estimation, low-power consumption.

I. INTRODUCTION

This paper presents a CMOS integrated circuit (IC) for the task of sound source-bearing estimation. The IC was originally conceived to work as a node in a sensor network and for this reason the minimization of power consumption is one of its main concerns. Methods to do sound-source localization are basically coherent or noncoherent [1]. An example of a coherent method is the correlation between signals arriving at different microphones [2]. Another example is the gradient flow [3] algorithm that estimates the bearing angle calculating the spatial gradient of the sound field. A mixed-signal IC implementing this

Manuscript received December 8, 2004. This work was supported in part by the National Science Foundation (NSF) under Grant IIS-0434161, by ANPCyT of Argentina under Grant PICT 2003 #14628, and by CONICET under Grant PIP 5048.

P. Julián is with the Universidad Nacional del Sur, Departamento de Ingeniería Eléctrica y Computadoras, 8000 Bahía Blanca, Argentina, and also with Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET), 1033 Capital Federal, Argentina (e-mail: pjulian@ieee.org).

A. G. Andreou is with the Electrical and Computer Engineering Department, The Johns Hopkins University, Baltimore, MD 21210 USA.

D. H. Goldberg is with the Department of Physiology and Biophysics, Weill Medical College, Cornell University, New York, NY 10021 USA.

Digital Object Identifier 10.1109/TVLSI.2005.863740

algorithm in a 3 mm × 3 mm 0.5- μ m CMOS technology has been presented in [4]. This method is based on analog processing at a sampling rate of 16 kHz, and discriminates 250 ns with a power consumption of 54 μ W. Other coherent approaches include cochlea based schemes, like the one originally proposed in [5] and neuromorphic inspired approaches, like those presented in [6]–[8]. Recently, an IC based on an analog cochlea was presented in [9] using a 0.5- μ m process in an area of 5 mm². In this case, the power dissipation depends strongly on the input signals. With no activity, the cochlear channels dissipate 400 μ W; for a time delay of 100 μ s (corresponding to a 77° angle incoming signal), the power dissipated is 1.85 mW. Coherent methods require precise synchronization among the nodes. Noncoherent methods include triangulation based on sound pressure level, and are less sensitive to synchronization but more dependent on sensor and channel characteristics [10].

The IC presented in this paper implements a coherent method at the sensor level, previously proposed by the authors in [11], which is based on the measurement of the interaural time delay (ITD) between signals. The IC is composed of 208 stages that perform the correlation derivative of delayed versions of the input signals in a 0.35- μ m technology process. The IC calculates the bearing angle of a sound source using four microphones located in the node in a 360° range with a mean variance of one degree. The microphones are used in pairs: one pair produces the indication and the other defines the quadrant. Given the relation between the bearing angle and the ITD, the accuracy is a nonlinear function of the incoming signal angle. Therefore, an adaptive algorithm decides which pair to use, depending on the direction of the incoming signal, in such a way to obtain the best possible estimate. The IC contains two blocks with 104 stages each. Every stage has a delay unit, a block to reduce the clock speed, and a 10-bit UP/DN counter. Thanks to the use of the correlation derivative instead of the correlation, the activity, and, therefore, the power consumption is reduced by processing at the speed of the incoming signal (300 Hz max) instead of the CLK speed (200 kHz). This reduces the power budget by a factor over 600. The IC measures 2 mm × 2.4 mm, consumes 180 μ A at 3.3 V, and has a clock frequency of 200 kHz. It is purely digital and uses a one-bit quantization of the input signals.

The paper is organized as follows. Section II briefly describes the motivating problem and the method developed for solving the problem in an efficient way. Section III presents the circuit architecture used. Section IV presents experimental results of a field test.

II. PROBLEM DESCRIPTION

The problem that originally motivated this project is the measurement of the bearing angle of a sound source in the low-audio range (10 Hz < f < 300 Hz) using four microphones located in an acoustic enclosure (see [12] and [11] for constructive details). If the distance between microphones is L then the inter-aural time delay (ITD) is given by $\text{ITD} = L/v \cos \beta$, where v is the speed of sound, and β is the angle between the sound source and the imaginary line that passes along the microphones. The acoustic enclosure is circular with a diameter of 11 cm and produces an effective distance between microphones of $L = 15.6$ cm so that the maximum ITD is 460 μ s. The method chosen to perform the estimation is based on the correlation method [2], [13]. If we assume that $x_1(k)$, $x_2(k)$ are samples of the signals entering a pair of microphones, then

$$\begin{aligned} x_1(k) &= s(k) + n_1(k) \\ x_2(k) &= s(k - D) + n_2(k) \end{aligned} \quad (1)$$

where $s(\cdot)$ is the signal emitted by the source, $n_1(\cdot)$ and $n_2(\cdot)$ are uncorrelated noise signals, and D is the time delay between microphones. The discrete-time correlation function is

$$\tilde{R}_{x_1 x_2}(i) = \sum_{k=0}^K x_1(k)x_2(k-i) \quad (2)$$

where K is the time window under consideration. Operation (2) can be implemented in a digital fashion after quantization of the signals. The study of simulations based on naturally recorded signals revealed that a quantization of the signal with more than one bit did not produce a change in accuracy (see [11]); therefore, a one-bit quantization was used.¹ From a hardware prospective, coding the signal with just one bit produces a dramatic reduction in the size and complexity of the design. Regarding the sampling time, the smaller it is, the more resolution can be achieved at the expense of power consumption. The target resolution imposed by the application was one degree. That resolution can be achieved in an angle range of $[-90, -40] \cup [+40, +90]$ using $T_s = 5 \mu$ s (see [11] for a theoretical justification). The setup included four microphones in quadrature, therefore, the specified accuracy can be achieved with a sampling time $T_s = 5 \mu$ s in the full range by switching microphone pairs.

The associated structure is composed of a number of stages

$$y(i) = \sum_{k=0}^K x_1(k)x_2(k-i) \quad (3)$$

where i is an index to the stage number. As the sampling time is 5 μ s and the maximum ITD is 460 μ s, the number of stages is 92. From a hardware viewpoint, the digital implementation of (3) requires shift registers to generate the delayed versions of x_2 , a counter implementing the correlation operation, and, finally, one block to determine where the maximum has occurred. In the worst case, considering a sampling time of 5 μ s and a time window of 1 s, a counter could reach a maximum count of 200 000 (e.g., 17.6 bits). However, once the signals are one-bit quantized, the information of the ITD is contained solely in the changes of the signal. Accordingly, no information is contained in those parts of the signal without state changes. However, every stage (3) is counting all the time at the speed set by the clock, regardless of input values. As the frequency of the clock is much higher than the frequency of the signal (200 kHz versus 300 Hz), this architecture will dissipate more power than is actually needed. An additional factor to consider in this case, is the need to actually calculate the maximum (18 bits) among all stages. In view of this, a much more efficient approach based on the correlation derivative approach was followed [11].

A. Correlation Derivative Approach

The maximum of the correlation occurs when the delay produced by the shift register chain coincides with the relative delay between signals. Mathematically, detecting the maximum of the correlation function is equivalent to detecting the zero-crossing of its derivative when the second derivative is negative. The discrete difference between adjacent stages in (3) is

$$\begin{aligned} \Delta y(i) &:= y(i) - y(i-1) \\ &= \sum_{k=0}^i x_1(k) \left[x_2(k-i) - x_2(k-(i-1)) \right]. \end{aligned} \quad (4)$$

¹This is a natural consequence of the used detection method that computes the time delay using the zero crossing of the signals. As a consequence, a quantization of the input with more than two levels has no effect on the estimation

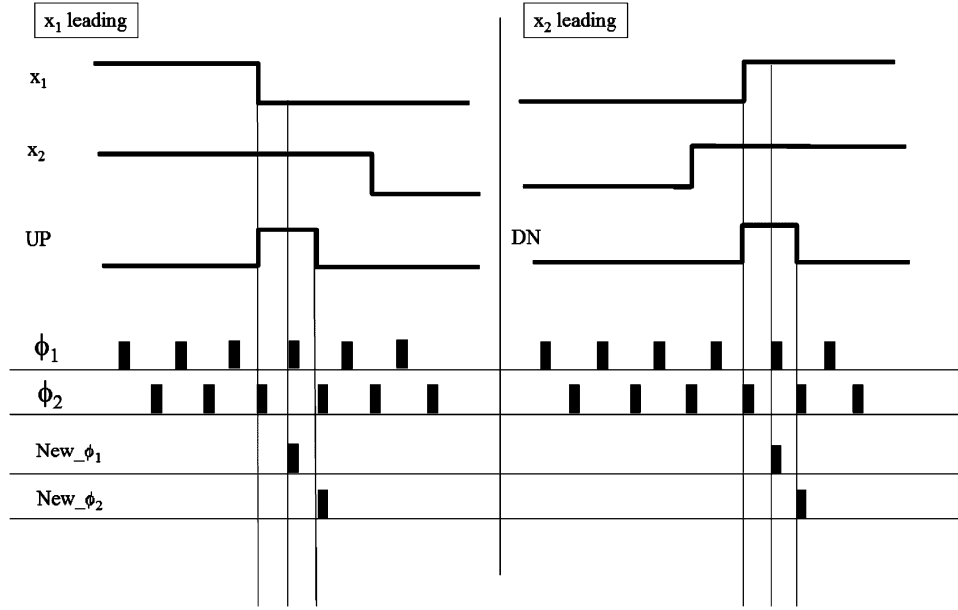


Fig. 1. Time behavior of signals UP and DN, original and modified clocks.

Equation (4) corresponds to an UP/DN counter. The counter counts up when $x_1(k) = 1$, and the other signal satisfies $x_2(k - i) = 1$ and $x_2(k - (i - 1)) = 0$; it counts down when $x_1(k) = 1$, and the other signal satisfies $x_2(k - i) = 0$ and $x_2(k - (i - 1)) = 1$. Accordingly, the signals UP and DN driving the counter can be written as

$$\begin{aligned} \text{UP} &= x_1(k) \cdot \left(x_2(k - i) \cdot \overline{x_2(k - i + 1)} \right) \\ \text{DN} &= x_1(k) \cdot \left(\overline{x_2(k - i)} \cdot x_2(k - i + 1) \right). \end{aligned}$$

In this case, the counter only operates when one of the signals changes state. This reduces the activity of the circuit and consequently implies a reduced power consumption. In addition, the counters are also smaller. The maximum possible count in this case corresponds to a signal of 300 Hz in a 1-s time window, that is 300 counts or 9 bits. Finally, the reading of the output is greatly simplified due to the value of the delay as given by the position of the stage where the zero-crossing has occurred; this can be done with a decoder.

III. CIRCUIT ARCHITECTURE

The localizer circuit can be divided into two main blocks. One block calculates the time delay between two signals and the other block is the control unit that determines the timing and order in which the different operations in the chip are performed.

A. Control Unit

The algorithm has a time window of 1 s to determine the direction and angle of the sound source. We have divided the time window into two periods. During the first, or main sweep, that occupies 75% of the time window the ITD is calculated using one pair of microphones. During the second, or secondary sweep, that occupies the remaining 25% of the time window, the other pair of microphones is used to determine the quadrant. An internal variable keeps track of the value of the pair of microphones used during the main sweep. The choice of this variable is *not* arbitrary and obeys the following fact. The precision of the algorithm in measuring the ITD depends strongly on the angle. For the chosen frequency of operation, the algorithm achieves, theoretically, a one degree error in the interval $[-90, -40] \cup [+40, +90]$. However, if the angle is outside this interval (i.e., it is smaller than 40°

in absolute value) the precision drops down fast. In order to avoid this loss of accuracy, the algorithm determines whether the sound source is in range during the main sweep. If the source is not in range, the internal variable is changed at the end of the secondary sweep. This way, during the next time window, the pair of microphones used in the main sweep will be the other and the signal will be in the good range again.

The state machine was implemented using an 18-bit counter and has seven different states.

- State 0: The main calculation is evaluated in this period. It lasts 163 840 tics of the clock, equivalent to 0.8192 s.
- State 1: The end of the main calculation. The output of the counters and the variable that determines if the reading was in range are latched internally (1 clock period).
- State 2: The counters are reset (1 clock period).
- State 3: The secondary calculation is evaluated in this period. It lasts 212 994 tics of the clock, equivalent to 0.2460 s.
- State 4: The end of the secondary calculation. The variables that determine the quadrant are latched internally (1 clock period).
- State 5: The variable defining the microphone input pair to be used in the next main period is set. The data with the measured time delay plus orientation are sent to the output pads of the chip (1 clock period).
- State 6: The counters are cleared and the count is reset (1 clock period).

B. Time Delay Estimator

The time delay estimator is composed of two identical circuit blocks. In each of these blocks, one of the signals, namely x , is fed to a delay chain consisting of 104 master-slave D flip-flops (FF). Associated to each FF there is one stage based on a 10-bit UP/DN counter. This block processes the delayed input x and the other input, namely y , which is not delayed, to produce the correlation derivative function. It is clear that such a block can measure the relative time delay between the signals only if signal y is delayed with respect to x . Accordingly, one block has the x_1 input connected to the delay chain and x_2 to the other input, while the other block has the x_2 input connected to the delay chain and x_1 to the other input.

In order to achieve power efficiency, the calculation of the correlation derivative is worked as follows. An auxiliary block, called signal

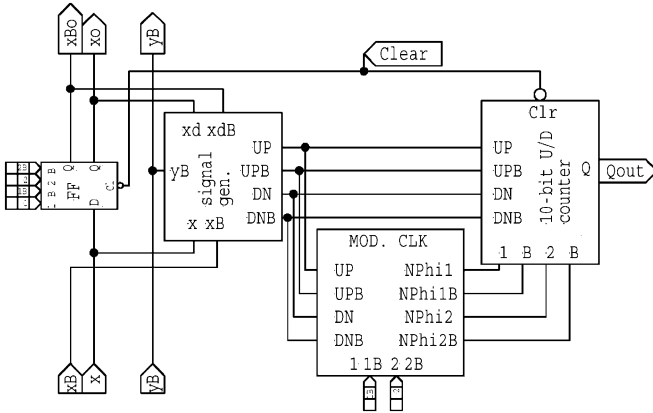


Fig. 2. Block diagram of the basic correlator block.

generator, generates two signals UP and DN. Both signals are pulses generated only when x changes, either from 0 to 1 or from 1 to 0. Signal UP goes high in the first clock pulse when x goes from 1 to 0 with $y = 1$, indicating that signal x is leading. Signal DN goes high in the first clock pulse when x goes from 0 to 1 with $y = 1$, indicating that signal y is leading. The system runs on a biphasic clock of 200 kHz generated from a 400-kHz base clock. This is the clock used by the delay chain. The 10-bit UP/DN counters, however, use a modified clock. Since these counters only need to count whenever there is a change in signal x , the modified clock is a replica of the system clock but is only active when either $UP = 1$ or $DN = 1$. This reduces the activity of the counters to the frequency of the input signal instead of to the frequency of the clock, providing a reduction factor of more than 600 in activity. The signals are illustrated in Fig. 1. The block diagram is shown in Fig. 2.

The UP and DN signals, together with the new clock signals, are then fed to a synchronous 10-bit UP/DN counter. The output of this block is the most significant bit (the 10th bit), which is also the sign bit. As all counters above the stage where the coincidence occurs have a count of a given sign, and all counters below have a count of the opposite sign, the zero-crossing is detected by connecting the sign bit of every pair of adjacent blocks (4) to an XOR gate, in such a way that it becomes active when two adjacent cells have a count of different signs. The XOR gates are connected to 8-bit input priority encoders that convert the location of the zero-crossing to a binary number that gives the reading of the delay in multiples of the sampling time T_s . Notice here that the use of the derivative in the calculation of the correlation eliminates the need to search for the maximum of the outputs, and instead provides a straightforward architecture to read the value of the delay.

The layout of an 8-stages block, each containing the FF delay chain, the signal generator, the 10-bit UP/DN counter, and the output priority encoder, is shown in Fig. 3. The photograph of the complete IC is shown in Fig. 4.

IV. EXPERIMENTAL SETUP AND RESULTS

The complete circuit was implemented in the TSMC 0.35- μm process; it occupies an area of 2 mm \times 2.4 mm and has 140 000 transistors. Working with a 3.3-V power supply, it features a power consumption of 600 μW (180 μA). For the experiment, a board with four microphones and preamplifiers was built. The microphones are miniature Knowless Sysonic MEMS, with a sensitivity of $-42 \text{ dB} \pm 4 \text{ dB}$ (0 dB = 1 V/Pa measured at 1 kHz) and a noise level of 35 dBA (SPL). Because the accuracy of the circuit relies on a precise measurement of the time delay between signals, the channels need to be matched in order to minimize phase mismatch.



Fig. 3. Layout of an 8-stages block containing the FF delay chain, the signal generator, the 10-bit UP/DN counter, and the output priority encoder.

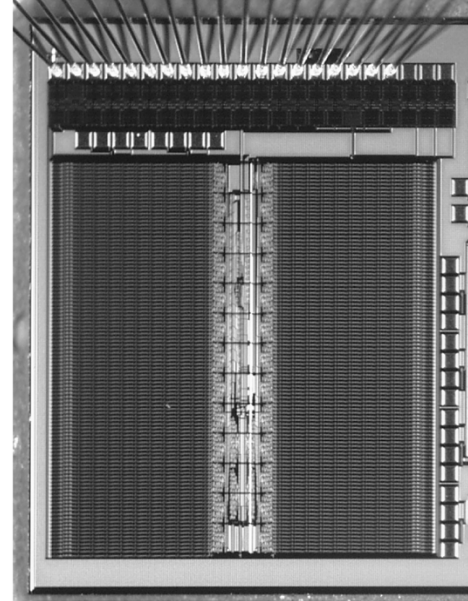


Fig. 4. Photograph of IC.

The preamplifier is a bandpass filter implemented using two TLV2382 low-power operational amplifiers (7 μA at 3.3 V) and matched discrete components. The low-cutoff frequency is $f_L = 10.6 \text{ Hz}$ and the high-cutoff frequency is $f_H = 473 \text{ Hz}$. The power consumption of the discrete part is dominated by the microphones; there are four of them and each one dissipates 500 μW . In addition, there are a total of eight operational amplifiers that consume 184.4 μW . Low-power integrated front ends have been reported in the literature [14]–[16] displaying a power consumption of less than 100 μW per channel, including the power of the microphone.

The circuit was tested outdoors using a recorded signal and a speaker. Two different signals were used: one of them was a narrow-band signal ($f = 200 \text{ Hz}$), and the other one was broad-band noise ($16 \text{ Hz} < f < 300 \text{ Hz}$). The speaker was placed at different angles, and for every angle ten readings were recorded (every reading is the output of the correlator derivative IC after 1 s of operation). Using these ten values, the mean and the STD were calculated for each reading. Fig. 5 shows the mean outputs for both types of signals. The narrow-band signal exhibits a mean deviation from the linear relationship of 3.95° and is below 6° for all angles, except the range $[60^\circ, 80^\circ]$. In the broad-band case, the mean deviation from the linear relationship is 5° and is below 7° for all angles, except in the range $[50^\circ, 80^\circ]$. These deviations can be compensated because they are monotonic, so the variable of interest to determine the accuracy is the STD [11]. Fig. 6 shows the STD deviation for both signals. For the narrow-band signal, the STD is less than 1.6° in the whole range, and the mean value of the STD in the whole range is 0.79° . For the broad-band noise the standard deviation (STD) of the output is greater. The maximum STD is 2.2° and the mean value in the whole range is 1.20° . In both cases, the asymmetric characteristics are mainly due to the environment. During the experiment, a signal-to-noise ratio between 25–30 dB was measured.

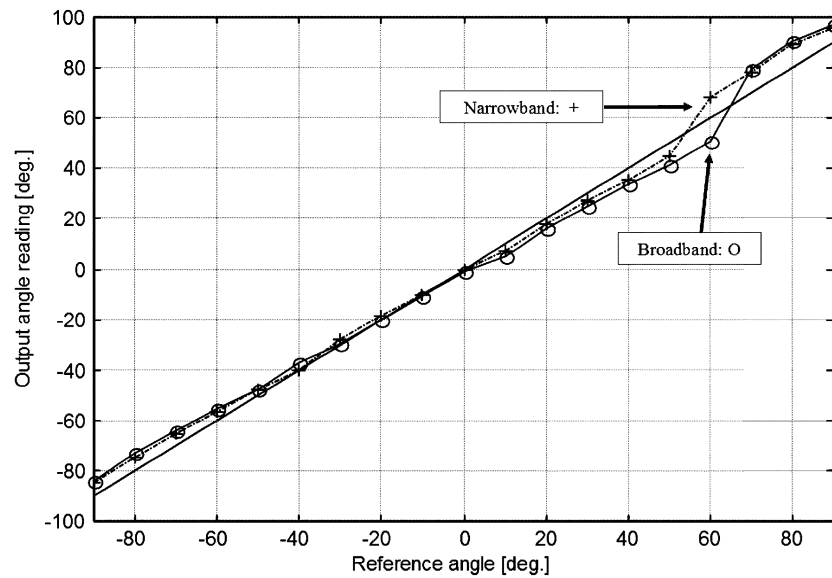


Fig. 5. Experimental field results: Mean values.

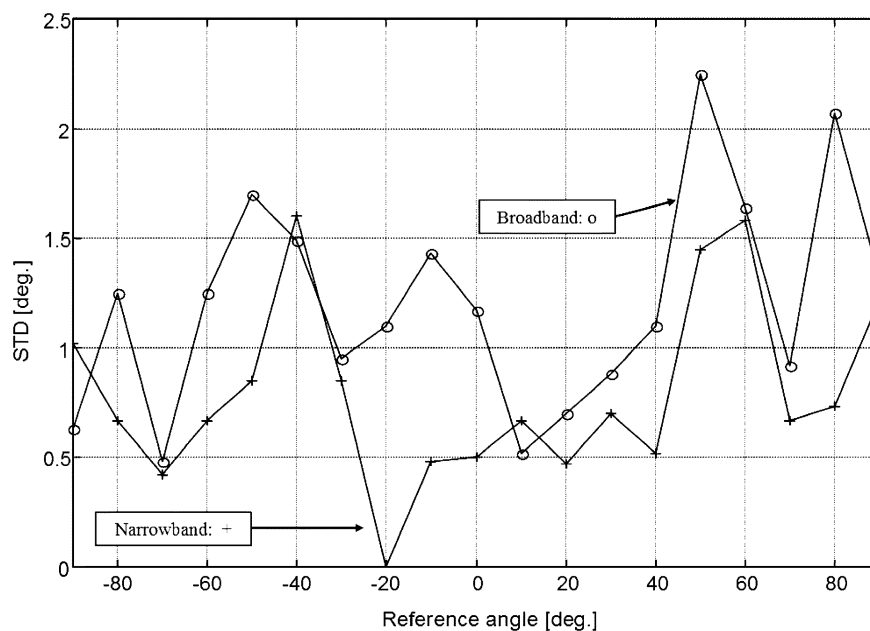


Fig. 6. Experimental field results: Standard deviations.

V. CONCLUSION

We have presented a CMOS VLSI circuit for bearing estimation in a $0.35\text{-}\mu\text{m}$ technology. The circuit implements a modification on the standard correlation approach, based on its derivative, which permits to obtain a dramatic reduction in the activity and, therefore, in power consumption. The integrated circuit has a current consumption of $180\ \mu\text{A}$ at $3.3\ \text{V}$ and a clock frequency of $200\ \text{kHz}$. The circuit also has a control logic that implements an adaptive algorithm to permanently select the best pair of microphones for the estimation. An experiment in a natural environment was setup in order to test the IC. The results show a mean STD of 0.79° and 1.20° , respectively, depending on whether the signal is narrow-band or broad-band noise.

ACKNOWLEDGMENT

The authors would like to thank P. Pouliquen for his helpful discussions and also P. Mandolesi and R. Rosasco for their help in the debugging and experimental tests.

REFERENCES

- [1] J. C. Chen, K. Yao, and R. E. Hudson, "Source localization and beamforming," *IEEE Signal Process. Mag.*, vol. 19, no. 2, pp. 30–39, Mar. 2002.
- [2] G. C. Carter, "Coherence and time delay estimation," *Proc. IEEE*, vol. 75, no. 2, pp. 236–255, Feb. 1987.
- [3] M. Stanacevic and G. Cauwenberghs, "Mixed-signal gradient flow bearing estimation," in *Proc. IEEE Int. Symp. Circuits and Syst. (ISCAS)*, vol. 1, 2003, pp. 777–780.

- [4] —, “Micropower gradient flow acoustic localizer,” *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 52, no. 10, pp. 2148–2156, Oct. 2005.
- [5] J. P. Lazzaro and C. Mead, “Silicon models of auditory localization,” *Neural Comput.*, vol. 1, pp. 41–70, 1989.
- [6] T. Horiuchi, “An auditory localization and coordinate transform chip,” *Adv. Neural Inform. Process. Syst.*, vol. 7, pp. 787–794, 1995.
- [7] J. G. Harris, C. J. Pu, and J. C. Principe, “A neuromorphic monaural sound localizer,” *Adv. Neural Inform. Process. Syst.*, vol. 11, pp. 692–698, 1999.
- [8] I. Grech, J. Micalef, and T. Vladimirova, “Experimental results obtained from analog chips used for extracting sound localization cues,” in *Proc. 9th Int. Conf. Electron., Circuits, Syst.*, vol. 1, 2002, pp. 247–251.
- [9] A. van Schaik and S. Shamma, “A neuromorphic sound localizer for a smart MEMS system,” *Analog Integr. Circuits Signal Process.*, vol. 39, pp. 267–273, 2004.
- [10] C. Savarese, J. M. Rabaey, and J. Beutel, “Locating in distributed ad-hoc wireless sensor networks,” in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Process. (ICASSP)*, 2001, pp. 2037–2040.
- [11] P. Julián, A. G. Andreou, G. Cauwenberghs, L. Riddle, and A. Shamma, “A comparative study of sound localization algorithms for energy aware sensor network nodes,” *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 4, pp. 640–648, Apr. 2004.
- [12] L. Riddle, “VLSI acoustic surveillance unit,” in *Proc. GOMAC*, Mar. 2004, pp. 12–13.
- [13] G. Carter, “Time delay estimation for passive sonar signal processing,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-29, no. 3, pp. 463–470, Jun. 1981.
- [14] M. W. Baker and R. Sarpeshkar, “A low-power high-PSRR current-mode microphone preamplifier,” *IEEE J. Solid-State Circuits*, vol. 38, no. 10, pp. 1671–1678, Oct. 2003.
- [15] W. A. Serdijn, A. C. van der Woerd, J. Davidse, and A. H. van Roermund, “A low-voltage low-power fully-integratable front-end for hearing instruments,” *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 42, no. 11, pp. 920–932, Nov. 1995.
- [16] J. Silva-Martinez and J. Alcedo-Suner, “CMOS preamplifier for electret microphones,” in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, vol. 3, 1995, pp. 1868–1871.

Comments on “Carry Checking/Parity Prediction Adders and ALUs”

José J. Rodríguez-Navarro

Abstract—In this brief, it is shown that the checking or comparison of normal carries versus duplicated carries in a carry checking/parity prediction adder can be partially avoided, making it feasible to implement a less complex checker when using a robust logic style.

Firstly, and for completeness, the basic Boolean difference concept is briefly introduced and later used systematically to prove theoretical aspects. For a logic network, the Boolean difference of its output, represented by a function $F(X) = F(x_1, \dots, x_n)$ of its input logic variables x_1, \dots, x_n , is given by

$$\frac{dF(X)}{dx_i} \triangleq F(x_1, \dots, x_i, \dots, x_n) \oplus F(x_1, \dots, \bar{x}_i, \dots, x_n)$$

where \oplus stands for the modulo-2 addition and \bar{x} represents the complementary of x . Now, our interest is the following consequence; that if

$dF(X)/dx_i = G(X)$, then an error in x_i will cause an error in $F(X)$ if and only if $G(X) = 1$. From this, it is apparent that the Boolean difference represents for a given logic network the dependencies of an output signal with respect to an input signal.

In conventional implementations of binary adders, we find and distinguish two sets of carries: the set of propagating carries and the set of nonpropagating carries. The set of propagating carries includes those carries c_i in which $dc_{i+1}/dc_i \neq 0$, where the index i represents the bitweight. In other words, a single error in c_i may be propagated to c_{i+1} . The rest of the carries form the set of nonpropagating carries, in which a single error in c_i is never propagated to c_{i+1} , i.e., $dc_{i+1}/dc_i = 0$.

The approach to implement totally self-checking adders for single faults presented by Nicolaidis [1] makes use of duplicated carries $c_i^{\text{dup}} = G_i^{\text{dup}} + P_i c_{i-1}$, where G_i^{dup} and P_i are the duplicated-generate and propagate signals for bitweight i , respectively, and where it is clear that c_i^{dup} has no dependency on normal carry c_i , i.e., $dc_i^{\text{dup}}/dc_i = 0$. The parity prediction scheme duplicate carry with parity check II (DCPC-II), introduced by Sellers *et al.* [2] and used in [1], consists of computing (or predicting) the parity of the sum P_s as $P_a \oplus P_b \oplus P_{c_{\text{dup}}} \oplus c_{\text{in}}$, where P_a is the parity of the addend, P_b is the parity of the augend, c_{in} is the input carry, and $P_{c_{\text{dup}}}$ is the parity of the duplicated carries. The carry-checking scheme [1] makes use of a two-rail checker in order to indicate an error, if any, c_i and c_i^{dup} are different and to compute $P_{c_{\text{dup}}}$. The following theorems show that partial checking of the carries is possible.

Theorem 1: A single error in a nonpropagating carry is detected by comparing the normal carry with its duplicated counterpart.

Proof: First of all, it is noted that a single error in a nonpropagating carry c_{i-1} ends up in a sum bit s_i being in error since $ds_i/dc_{i-1} = 1$, and possibly in a duplicated carry c_i^{dup} being in error since $dc_i^{\text{dup}}/dc_{i-1} = P_i$. Therefore, the parity prediction method DCPC-II may fail to detect the error if propagated to the duplicated carry. Therefore, a comparison between these nonpropagating carries and their duplicated counterparts is necessary to detect these errors. ■

Theorem 2: A single error in a propagating carry is detected by a parity prediction DCPC-II.

Proof: It is obvious that a single error in a propagating carry c_i that does not propagate to the next carries c_{i+1}, \dots, c_{i+q} is detected by parity prediction DCPC-II, since it results in a single sum bit in error. In the case of this error being propagated to the next carries c_{i+1}, \dots, c_{i+q} , it results in sum bits $s_{i+1}, \dots, s_{i+q+1}$ in error and duplicated carries $c_{i+1}^{\text{dup}}, \dots, c_{i+q}^{\text{dup}}$ in error. The total number of errors is $2q + 1$, which is an odd number, and, therefore, the error is detected by parity check. ■

Thus, the fault-secureness of a carry checking/parity prediction adder can be achieved just by the use of a two-rail checker applied to those nonpropagating carries and duplicated counterparts, and by the use of an XOR-tree applied to the rest of the duplicated carries. Since the output of the two-rail checker corresponds to the odd and even parity of its inputs [3], these can be used in combination with the output of the XOR-tree for the final parity generation of $P_{c_{\text{dup}}}$. It should be noted that depending on the carry generation structure of the adder, groups of adjacent nonpropagating carries may appear. In this case, it is sufficient to compare nonpropagating carries with their duplicated ones at alternate stages, resulting in lower overhead. This is due to the fact that the nonpropagating carry and the duplicated carry differ in pairs when carry errors occur. In other words, an error in c_{i-1} can cause an error in c_i^{dup} without causing an error in c_i , and can be detected by comparing c_{i-1} and c_{i-1}^{dup} or c_i and c_i^{dup} . If the error in

Manuscript received December 8, 2004.

The author was with RWTH Aachen University, D-52056 Aachen, Germany. He is now with the MED-EL Medical Electronics, A-6020 Innsbruck, Austria (e-mail: jjr.navarro@gmail.com).

Digital Object Identifier 10.1109/TVLSI.2005.863739