

A mixed analog–digital fast hamming-weight filtering circuit using switched-capacitor arrays

Saleh Abdel-hafeez, Behrooz Parhami & Mohammad Al-Hammouri

Analog Integrated Circuits and Signal Processing

An International Journal

ISSN 0925-1030

Volume 83

Number 1

Analog Integr Circ Sig Process (2015)
83:35-44

DOI 10.1007/s10470-015-0502-6

ANALOG INTEGRATED CIRCUITS AND SIGNAL PROCESSING

An International Journal

Volume 83 · Number 1 · April 2015

RF AND MILLIMETER WAVE ICs

Low-power, small-size transmitter module with metamaterial antenna

K. Hiratah · T. Wada · K. Kubo · Y. Otsu · M. Ikebe · E. Sano 1

Physical design optimization of MOSFETs for millimeter wave and sub-millimeter wave circuits

Q.J. Gu · Z. Xu · J.Y.-C. Liu 11

A 40 GHz wireless link for chip-to-chip communication in 65 nm CMOS

T. Tikka · O. Viitala · J. Rynnänen 23

A mixed analog–digital fast hamming-weight filtering circuit using switched-capacitor arrays

S. Abdel-hafeez · B. Parhami · M. Al-Hammouri 35

An ultra-high speed monolithic clock recovery circuit in 0.2- μ m GaAs process

L. Tang · Z. Wang · Y. Qiu · C. Zhang · J. Xu 45

A 90-GHz receiver in 40-nm CMOS for plastic waveguide links

M. Tytgat · N. Van Thienen · P. Reynaert 55

MIXED SIGNAL LETTERS

A compact curvature corrected bandgap reference in 0.35 μ m CMOS process

P.B. Basu · D.Y. Aksin 65

A new implementation for the logarithmic/exponential function generator

M.T. Abuelma'atti · N.A. Tassaduq 75

An energy-efficient high data-rate 915 MHz FSK wireless transmitter for medical applications

A. Moradi · M. Sawan 85

Interference rejection in quadrature band-pass $\Sigma\Delta$ modulators

Y.B. Nithin Kumar · E. Bonizzoni · A. Patra · F. Maloberti 95

(continued on back cover)

 SPRINGER

ISSN: 0925-1030

 Springer

Your article is protected by copyright and all rights are held exclusively by Springer Science +Business Media New York. This e-offprint is for personal use only and shall not be self-archived in electronic repositories. If you wish to self-archive your article, please use the accepted manuscript version for posting on your own website. You may further deposit the accepted manuscript version in any repository, provided it is only made publicly available 12 months after official publication or later and provided acknowledgement is given to the original source of publication and a link is inserted to the published article on Springer's website. The link must be accompanied by the following text: "The final publication is available at link.springer.com".

A mixed analog–digital fast hamming-weight filtering circuit using switched-capacitor arrays

Saleh Abdel-hafeez · Behrooz Parhami ·
Mohammad Al-Hammouri

Received: 11 November 2013/Revised: 2 December 2014/Accepted: 2 February 2015/Published online: 11 February 2015
© Springer Science+Business Media New York 2015

Abstract Many circuit design applications rely on an intermediate sequence to carry a decision to the next circuit stage. The decision may be carried by a weighted pattern of N bits, with the weights being selected in a way that optimizes the circuit implementation or some aspect of performance. For example, when the weights are consecutive powers of 2 beginning with $1 = 2^0$, we have the standard binary representation. As another example, when all the weights are 1, we have the unary representation that encodes a value k by k asserted bits and $N-k$ unasserted bits (a weight- k bit vector of length N). In this paper, we present the design of a circuit that screens a unary representation to verify that the represented value falls between preset lower and upper limits l and u , passing through any string that represents a value in the interval $[l, u]$ and outputting the all-0 s bit pattern otherwise. Our mixed analog–digital circuit implementation, based on switched-capacitor arrays, provides a decision output within a clock cycle of 4 ns for 16-bit unary representation, when realized with $0.15\ \mu\text{m}$ TSMC technology. The latter results were obtained with normal, per-bit capacitance of 200 fF and single-clock-cycle operation. As an added benefit, our filtering circuit can form the basis for designing a cost-effective Hamming decoder circuit.

Keywords Decision circuit · Hamming decoder · Hamming filter · Hamming weight · Mid-pass filter · Switched-capacitor array

S. Abdel-hafeez · M. Al-Hammouri
Jordan University of Science and Technology, Irbid 22110,
Jordan

B. Parhami (✉)
University of California, Santa Barbara, CA 93106-9560, USA
e-mail: parhami@ece.ucsb.edu

1 Introduction

Decision circuits such as controllers, counters, and Vernier delay lines usually generate a 0–1 bit pattern which is carried to subsequent operation stages. Such bit patterns can be associated with two kinds of semantics. The first class, involving the use of fixed position weights (such as powers of a radix), requires digital arithmetic circuitry [1–3], such as XOR logic, decoders, or comparators, for reaching appropriate decisions regarding matching values. The second class, relying on unweighted bit patterns, treats all bit positions the same way and requires the detection of Hamming weight of a bit pattern to deduce its associated value. The second class has received much recent attention [4–8]. Many circuits in the latter class have been designed to cater to specific applications or application domains involving Hamming vectors and weights. Among the many available options, one finds all-digital designs [9–12], digital designs based on the technology-ratioed approach [13–15], designs using lookup tables and approximations [16–18], and, in the case of our prior contributions, mixed digital–analog designs that are more compact and use high-speed parallel evaluation [19, 20]. A general view, covering both radix-based weights and unit weights, is the assignment of arbitrary “importance” to various features [21].

Our focus in this paper is on determining the number of active bits in, or the Hamming weight of, a bit pattern. We refer to such bit patterns as Hamming sequences, so as to distinguish them from the more commonly used “positional” patterns in which bit positions are associated with different power-of-2 weights. We derive an efficient structure for a Hamming filtering circuit that solves the just-stated problem and also constitutes a cost-effective building block for the design of Hamming decoder circuits,

which in turn form commonly used blocks in a variety of important applications dealing with Hamming sequences [4–8].

Any representation of numerical values is prone to inaccuracies when errors cause some bits to flip. With a very small number of bit-flips (the common case), the inaccuracy is less problematic with unary than binary representation: an important advantage for the unary code. To overcome this difficulty, one may forego weighted representation in favor of constant-weight codes. A weight- k code can be used to represent $\binom{N}{k}$ options using N bits.

The representational efficiency is lower than that of binary representation, but we gain the ability to tolerate any number of bit-flips, provided they are all in the same direction (the so-called unidirectional errors). To maximize the representational efficiency, one picks the single weight k (or a range of weights) to be as close to $N/2$ as possible. Ever since the 1970s, unidirectional errors have been found to be quite common in VLSI circuits [22–26], so making circuits resilient to them is highly desirable.

The rest of this paper is organized as follows. Basic concepts related to the use of switched-capacitor arrays are reviewed in Sect. 2. CMOS circuit design with related equations and event timings for $N = 4$ is presented in Sect. 3 and generalized in Sect. 4. Section 5 reports on the results of our HSPICE simulation [27] based on 0.15 μm and 1.5 V integrated-circuit TSMC technology [28] for $N = 4, 8,$ and 16 bits, offering detailed analyses on accuracy and power consumption. Section 6 offers speed and cost comparisons with alternative state-of-the-art solutions. Section 7 concludes the paper.

2 Basic concepts and functionality

Consider a circuit that examines an N -bit pattern to ensure that its Hamming weight lies within the allowed interval (l, u) . Please note that we use the open-interval notation, where values included in the interval are strictly greater than l and strictly less than u ; in other words, in contrast with the closed-interval notation $[l, u]$, the end values l and u are not part of the interval (l, u) . The circuit accomplishes this goal via producing an output decision signal that allows all such valid patterns to pass through and any invalid pattern (having a weight that is less than or equal to l or greater than or equal to u) to be blocked by forcing the output to the all-0s state. We call such a circuit a Hamming mid-pass filter with parameters l and u . Taking $N = 4$ as an example, of the 16 possible 4-bit patterns, one has the Hamming weight 0, four have weight 1, six have weight 2, four have weight 3, and one has weight 4. So, a filtering

circuit with the parameters $(l, u) = (1, 3)$ will let 6 of the 16 patterns through unmodified and will pass on 0000 for the other 10 patterns, including the input 0000. The all-0s pattern may be viewed as a safe state that causes no harm at output.

Switched-capacitor arrays have become attractive in working with Hamming sequences. A good example is in circuits that compare Hamming weights of two different sequences or the weight of one sequence against a fixed threshold [20]. Unfortunately, however, circuit realization through switched-capacitor array usually implies high design cost and large power consumption due to the need for a high-resolution analog comparator. Furthermore, large capacitor mismatches due to the usage of single threshold capacitance can lead to inaccuracies and requires as many as 3–4 non-overlapping phased clock signals, usually generated by a delay-locked loop circuit [29, 30]. Layout design is also riddled with pitfalls that may complicate testing and compromise reliable operation.

Being aware of the drawbacks just enumerated, we hope to use a switch-capacitor-based design, taking advantage of novel structural features to mitigate the aforementioned problems and to gain speed advantages in a particular application. Our application of interest is to determine whether the Hamming weight of a bit pattern falls within an allowed range (l, u) , as outlined at the beginning of this section. Our proposed circuit uses only a 2-phase operation, which can be associated with clock high and low periods, thus obviating special circuitry for generating non-overlapping clock phases. Using a mathematical series arrangement, we distribute the threshold capacitive ratio used to compare the arrival of bit patterns in specific weight ranges among all positions, thus reducing the capacitance mismatches and balancing all feed-through voltages during the switching of inside capacitances.

Note that in hardware implementation, we can take advantage of the discrete nature of Hamming weights to implement a $(1, 3)$ Hamming mid-pass filter with the analog thresholds 1.5 and 2.5, thus allowing some noise tolerance without rendering an incorrect decision.

3 Circuit and operation

Let XH be the Hamming weight of a bit-vector X of length N . Normalizing the weight to the real (closed) interval $[0, 1]$, we have the normalized Hamming weight:

$$WH = XH/N \quad (1)$$

Since analog comparators work on the basis of less-than or greater-than relationships, without including the equal case, we associate a range with each Hamming weight:

$$(XH - 1/2)/N < WH < (XH + 1/2)/N \tag{2}$$

The lower and upper bounds are designated as:

$$YL = (XH - 1/2)/N \tag{3}$$

$$YU = (XH + 1/2)/N \tag{4}$$

For simplicity and clarity, we first describe the design of the circuit for $N = 4$ bits, later extending the design to $N = 8$ and $N = 16$ bits.

As shown in Fig. 1, our proposed 4-bit circuit contains three switched-capacitor arrays and two comparators. The switched-capacitor arrays receive the Hamming input sequence XH via the associated switch settings SXB_i , where i is bit position in XH . Internal to the circuit, the bounds YL and YU are associated with the switch settings $SYLB_i$ and $SYUB_i$. The Hamming input XH is compared against YL by the lower analog comparator, which produces the outputs VPL and VNL . Symmetrically, XH is compared against YU by the upper analog comparator, which yields the outputs VPU and VNU . The names of the comparator output signals incorporate the letters P and N to indicate positive and negative outcomes and the letters L and U to designate the lower and upper comparator.

Elaborating further, the top comparator module checks the Hamming input weight against the YL lower bound, while the bottom comparator module checks the Hamming input against the YU upper bound. If the two comparators outputs VPL and VNU are asserted, then the Hamming sequence weight is within the specified range; in this case, the sequence is forwarded to the next stage unchanged; otherwise, the all-0s pattern is forwarded to output. On occasion, we may be interested in the lowest-weight all-0s pattern, that is, in the smallest lower-bound value (when VNU and VNL are asserted), or in the largest-weight all-1s pattern, that is, in the largest upper-bound value (when VPU and VPL are asserted). Thus, the sequence is forward to next stage unchanged; otherwise all-0s pattern being forward to output.

The switched capacitors in the array pertaining to XH all have the common capacitance C_0 , while those in the switched-capacitor array for YU and YL are given the following sequence of capacitances, with the only difference between them being the switching activities, as presented in Table 1 and Eqs. (5)–(8).

$$SYUB_0 = SYLB_0 = B_0 = \frac{1}{4} C_0 \tag{5}$$

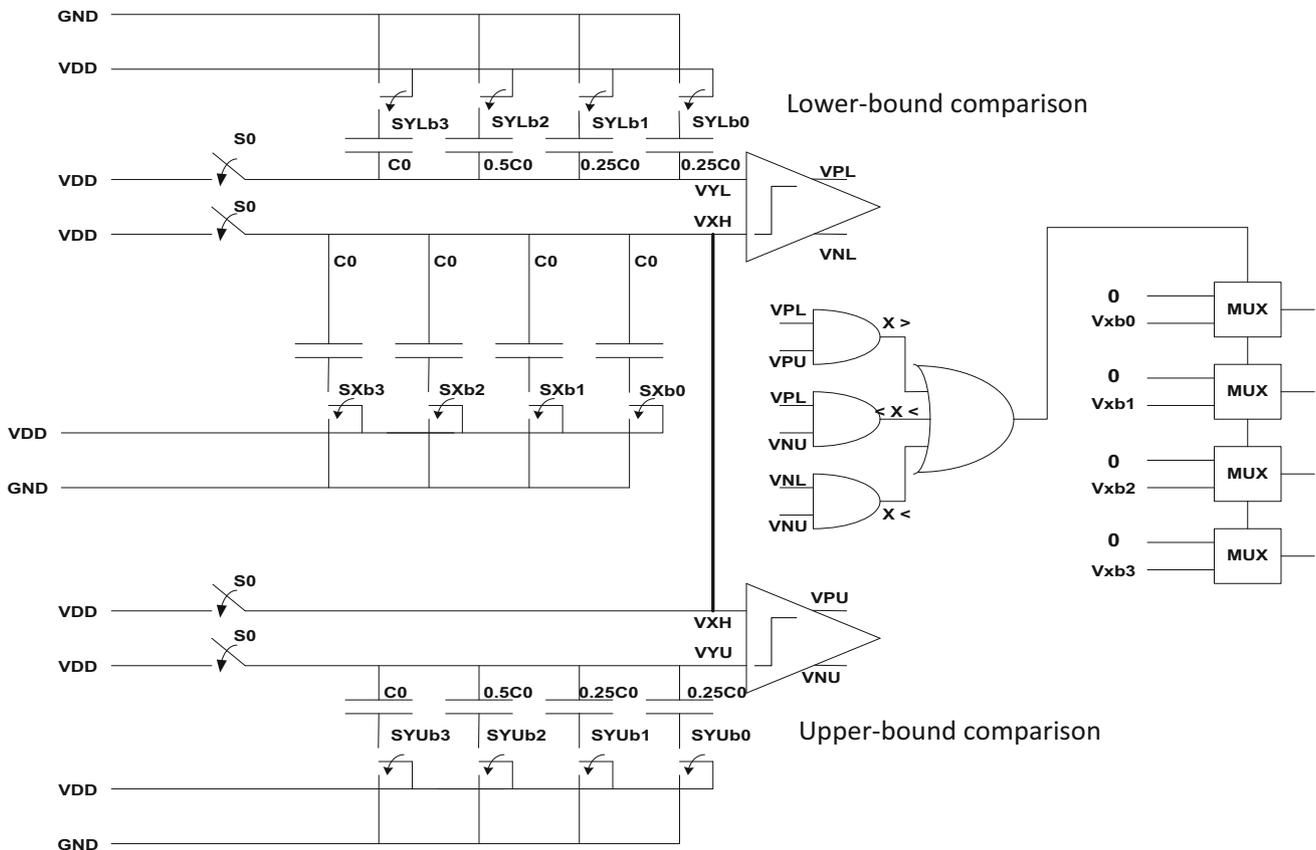


Fig. 1 A CMOS realization of our Hamming mid-pass filtering circuit

Table 1 Normalized Hamming weights and the associated lower- and upper-bound values

Hamming weights	Hamming values (<i>VXH</i>)	Lower bound (<i>VYL</i>)	Upper bound (<i>VYU</i>)
0	0	–	0.125
1	0.25	0.125	0.375
2	0.5	0.375	0.625
3	0.75	0.625	0.875
4	1.0	0.875	–

Table 2 Capacitor switching activities for different weight ranges

Lower bound (<i>VYL</i>)	Upper bound (<i>VYU</i>)	<i>YL</i> switched-capacitor	<i>YU</i> switched-capacitor
–	0.125	All OFF	1/4
0.125	0.375	1/4	1/4 + 1/2
0.375	0.625	1/4 + 1/2	1/4 + 1
0.625	0.875	1/4 + 1	1/4 + 1/2 + 1
0.875	–	1/4 + 1/2 + 1	All On

$$SYUB_1 = SYLB_1 = B_1 = \frac{1}{4}C_0 \tag{6}$$

$$SYUB_2 = SYLB_2 = B_2 = \frac{1}{2}C_0 \tag{7}$$

$$SYUB_3 = SYLB_3 = B_3 = C_0 \tag{8}$$

We can verify the operation of switched-capacitor arrays and associated equations in determining the lower- and upper-bound comparisons by means of examples, using Table 2.

In the case of $N = 4$ and with range parameters (l, u) = (2, 4), satisfied when $XH = 1011$ or for any of the tree input sequences having weight 3, the entries in Table 2 confirm proper operation, as follows. For this example, the upper bound is evaluated using (4) to be $0.875 \cdot VDD$, and the lower bound is evaluated using (3) to be $0.625 \cdot VDD$, where VDD is the voltage supply. In addition, the Hamming input yields the voltage $0.75 \cdot VDD$ based on (1). Now, using the switch activities of YU and YL from Table 2, the accumulated charges VXH , VYU and VYL are evaluated within the two phases of system clock (CLK_{sys}).

During Phase 1, all the following equations are satisfied simultaneously. For the lower bound voltage VYL , the relationship $C_0 \cdot [VDD - VDD] + 0.5C_0 \cdot [VDD - VDD] + 0.25C_0 \cdot [VDD - VDD] + 0.25C_0 \cdot [VDD - 0] = QTL$ specifying the total charge yields:

$$QTL = 0 \tag{9}$$

For the upper bound voltage VYU , we have $C_0 \cdot [VDD - VDD] + 0.5C_0 \cdot [VDD - VDD] + 0.25C_0 \cdot [VDD - VDD] + 0.25C_0 \cdot [VDD - VDD] = QTU$ as the total charge, resulting in:

$$QTU = 0 \tag{10}$$

For the Hamming input XH , the total charge $C_0 \cdot [VDD - VDD] + C_0 \cdot [VDD - VDD] + C_0 \cdot [VDD - VDD] + C_0[VDD - VDD] = QTX$ simplifies to:

$$QTX = 0 \tag{11}$$

On the other hand, during Phase 2, all the following equations are satisfied simultaneously and every switch is connected to ground. For the voltage lower bound VYL , the charge balance equation $C_0[VYL - VDD] + 0.5C_0[VYL - 0] + 0.25C_0[VYL - 0] + 0.25C_0[VYL - VDD] = 0$ yields:

$$VYL = 0.625 \cdot VDD \tag{12}$$

For the upper bound voltage VYU , the charge balance equation $C_0[VYU - VDD] + 0.5C_0[VYU - VDD] + 0.25C_0[VYU - 0] + 0.25C_0[VYU - VDD] = 0$ yields:

$$VYU = 0.875 \cdot VDD \tag{13}$$

For the Hamming input voltage VXH , the charge balancing equation $C_0[VXH - VDD] + C_0[VXH - VDD] + C_0[VXH - VDD] + C_0[VXH - VDD] = 0$ results in:

$$VXH = 0.75 \cdot VDD \tag{14}$$

During this phase, the comparators start comparing the input voltages and activate the outputs. The signals VPL and VNU are asserted, and thus, the decoder logic selects the multiplexer to pass through the Hamming input XH (in our example, 1011).

Different boundary ranges can be chosen which is based on the boundary sequence derived in Table 2. This boundary sequence, which is of the following form, can exploit all possible voltage ranges for VYL and VYU , using Table 1 along with switching values in Eqs. (5)–(8).

$$BS_4 = C_0 \cdot B_3 + 0.5C_0 \cdot B_2 + 0.25C_0 \cdot B_1 + 0.25C_0 \cdot B_0 \tag{15}$$

We now proceed to extend our design, explained in detail for $N = 4$, to other values of N .

4 Extending to arbitrary values of N

The boundary sequence can be simply extended to larger size of Hamming input. For $N = 8$ and $N = 16$, we have the following boundary sequences:

$$BS_8 = C0 * B_7 + C0 * B_6 + 0.5C0 * B_5 + 0.5C0 * B_4 + 0.25C0 * B_3 + 0.25C0 * B_2 + 0.25C0 * B_1 + 0.25C0 * B_0 \tag{16}$$

$$BS_{16} = C0 * B_{15} + C0 * B_{14} + C0 * B_{13} + C0 * B_{12} + 0.5C0 * B_{11} + 0.5C0 * B_{10} + 0.5C0 * B_9 + 0.5C0 * B_8 + 0.25C0 * B_7 + 0.25C0 * B_6 + 0.25C0 * B_5 + 0.25C0 * B_4 + 0.25C0 * B_3 + 0.25C0 * B_2 + 0.25C0 * B_1 + 0.25C0 * B_0 \tag{17}$$

Extending to an arbitrary power-of-2 value of N is straightforward. In general, one quarter of the capacitors in the switched-capacitor array associated with the lower or upper bound have capacitance $C0$, one quarter have capacitance $0.5C0$, and the remaining half have capacitance $0.25C0$. This leads to the total capacitance of $(N/2)C0$, thus allowing all ratios between $1/(2N)$, corresponding to a single $0.25C0$ capacitance selected, and $(2N - 1)/(2N)$, associated with all but one of the $0.25C0$ capacitances selected, to be synthesized. The preceding description leads to the boundary sequence formula:

$$BS_N = (N/4)C0 * \sum_{3N/4 \leq i \leq N-1} B_i + (N/4)0.5C0 * \sum_{N/2 \leq i \leq 3N/4-1} B_i + (N/2)0.25C0 * \sum_{0 \leq i \leq N/2-1} B_i \tag{18}$$

So far, we have assumed that N is a power of 2. For a non-power-of-2 value of N , we use the next larger value that is a power of 2 for our design. For example, given $N = 10$, we build our circuit for $N = 16$ in order to keep the same structure of uniform capacitances that reduces mismatches. Then, we employ a combination of input voltages similar to those in Table 2 to provide the boundary values for any Hamming weight within $N = 10$. This strategy works because the boundary sequence formula guarantees any boundary for any Hamming weight from 0 to 16.

5 Assessment via HSPICE simulation

The proposed circuit depicted in Fig. 1 has been synthesized based on $0.15 \mu\text{m}$ CMOS technology for three different sizes of $N = 4, 8$, and 16 bits. All possible waveform weights of the Hamming input HX , and all possible boundaries (YL, YU) using the BS_4 of (15) are depicted in

Fig. 2. The corresponding waveforms for $N = 8$ and 16 , using BS_8 and BS_{16} , respectively, are depicted in Figs. 3 and 4, respectively. Note that as the size N increases, the comparator input resolutions become smaller, where this resolution might be predicted by the difference between (1) and (2) or (1) and (4) leading to the ideal theoretical comparator resolution difference:

$$ICRD = 1/(2N) \tag{19}$$

Table 3 summarizes ICRDs margin resolution based on Eq. (19) from HSPICE simulations presented in Figs. 2, 3, and 4. The worst-case HSPICE results deviate from the theoretical data, leading to error differences reported in the rightmost column of Table 3. This error difference is due to pass-gate switch mismatches [31, 32], feed through charges [33, 34], and other second-order effects related to non-overlapping signals [35]. However, the two lines of input comparator are subjected to the same error, and thus, provide the same ratio of charge mismatches, which reduces the overall error.

Besides, the switched-capacitor array of YU and YL distribute the weight charges on all capacitances instead of single capacitor that usually has the offset threshold value. More details on reducing the mismatches between capacitances in switched-capacitor arrays can be found in several of our references [19, 20, 36–38]. Analysis results show that the worst-case capacitance mismatches are such that the layout can support $N = 64$ or higher, with mismatch error not exceeding 22 %, while providing a resolution on the order of 8 mV. Most current ADC comparators can support up to 0.1 mV resolution [31, 32, 35, 39].

The limited speed is determined during Phase 1 of charge accumulation since during this phase the VDD need to pre-charge the comparators input (YL, XH, YU) lines through the $S0$ switches. These lines carry sum of total switched-capacitance array capacitances. Therefore, the time delay is limited by the time constant [40, 41], which may be predicted by the following form:

$$T_{\text{delay}} = K * RTG * C_{\text{sum}} \quad \text{where } 3 < K < 8 \tag{20}$$

That is, K is the accuracy order of time constant, RTG is the equivalent transmission gate switch resistance, and C_{sum} is the lumped sum of all switched capacitances in the array associated with the input line of comparators. In our simulations, we proceed to about $K = 6$ in order to provide large margin for our basic analog comparators which can compare an input value of order 10 mV. In addition, the nominal $C0 = 200 \text{ fF}$ and T_{delay} shows the value of 2, 2, and 4 ns for the $N = 4, 8$, and 16 , respectively. We use $0.15 \mu\text{m}$ TSMC technology with voltage supply 1.5 V.

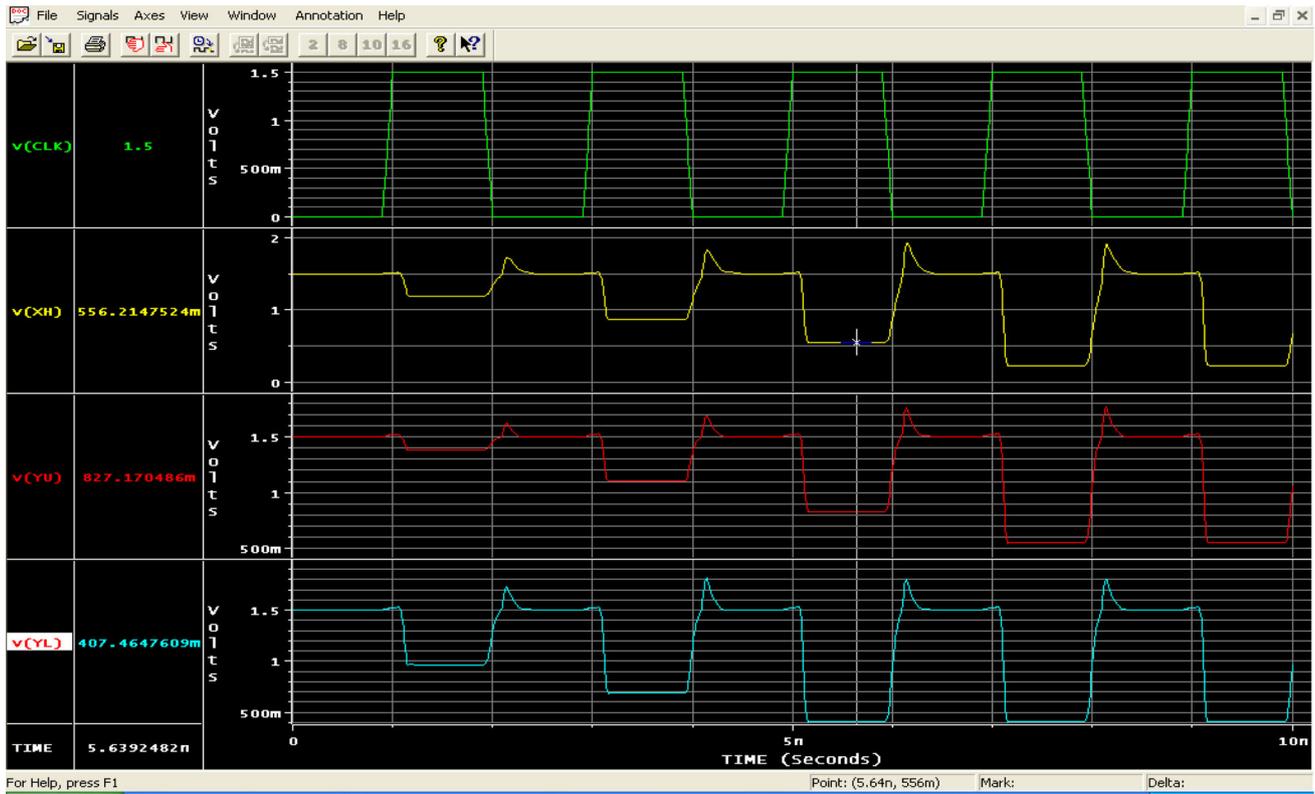


Fig. 2 HSPICE waveforms when the 4-bit Hamming weight XH is sequentially increased, while YL and YU define the ranges for each new XH . Horizontal time scale is in nanoseconds and vertical output scale is in volts ($CLK = 0.5\text{ GHz}$, $V_{dd} = 1.5\text{ V}$), using TSMC $0.15\text{ }\mu\text{m}$ technology

6 Speed and cost comparison

The design proposed in this paper is unique in functionality and implementation, making comparisons with state-of-the-art modules having similar objectives rather difficult. However, it is still possible to compare our design with alternative proposals where the functionalities overlap. More specifically, consider that our proposed design deals with an incoming bit-vector within the following cases:

$$XH_{out} = \begin{cases} 0 & XH < YHL \\ XH & YHL < XH < YHU \\ 0 & YHU < XH \end{cases} \quad (21)$$

Thus, as evident from Eq. (21), our design covers comparisons, Hamming distance measurements, and filtering of pre-selected Hamming weights. Most previously proposed Hamming-related module designs [9, 10, 13, 16, 17], by contrast, are aimed at measuring distances between, or comparing, bit-vectors, entailing only one comparison instead of two parallel ones. Additionally, the aforementioned functionalities of our design are applicable to N -bit input XH , with YHL and YHU potentially being of same width or narrower.

Recent work [10] offers several Hamming-distance analyzer modules based on look-up tables and counting

networks, which are primarily optimized for FPGA resources. A re-usable counting network module similar to the one in [10] is proposed in [13]. The resulting designs are fast and take into account all Hamming-weight possibilities in network routing, but they suffer from circuit-level design difficulties related to large fan-in and fan-out capacitances as well as substantial gate-count and pipeline register cost. Attempts to reduce the number of gates, and thus the fan-in and fan-out capacitance loading for routing network, have led to the proposal for an inexact-decision Hamming-weight threshold voting scheme [16]. However, the resulting circuit still suffers from many of the same design problems, with the further drawback of limited and narrow applications. Parhami [9] has proposed a more general Hamming-weight comparator using the notion of signed bits. His design, though improving on earlier proposal, still entails large gate count, delay, and power.

Our design uses capacitance weights as lumped Hamming threshold values, passing them to high-resolution and low-power analog comparators [31, 32, 35, 39]. Analog comparators [32] are widely known in terms of positive-feedback switched capacitor circuits. Our design avoids known problems of such circuits (namely, large capacitive loading at the comparator inputs and multiphase clock operation necessitating the use of DLL circuits) by using a



Fig. 3 HSPICE waveforms when the 8-bit Hamming weight XH is sequentially increased, while YL and YU define the ranges for each new XH . Horizontal time scale is in nanoseconds and vertical output scale is in volts ($CLK = 0.5$ GHz, $V_{dd} = 1.5$ V), using TSMC 0.15 μm technology

single-cycle operation and uniformly weighted capacitances associated with each bit of the input bit-vector.

We next report on the results of HSPICE evaluation of our design and a representative prior design [9] suitable for 0.15 μm CMOS technology. Similar comparisons based on different CMOS technologies have been reported in [9, 10, 13, 16, 17]. Our HSPICE comparison results are shown in Table 4. Note, however, that the figures in adjacent columns of Table 4 are not directly comparable, as we have to add various elements to the figures shown for our design. These elements vary within a range, so we have chosen not to incorporate them in the table; rather we discuss them in the paragraphs that follow.

The transistor counts should be viewed as rough indicators of relative circuit complexities. The silicon area is also affected by details of the capacitance array geometry in our design and by routing in any alternate design based on the choice of the number of metal layers. The latency (transient delay) of our design excludes the contributions of the comparator and mux front circuit shown in Fig. 1. The delay of a comparator with the needed resolution and technology factor is in the range 0.74–1.38 ns. Finally, we note that power consumption figures for our design must be augmented by the power requirements of the two

comparators, which with low-power design consume in the range of 0.27–0.43 mW using comparable technology [31, 32].

Our comparison results show a close match, with minor characteristic advantages of the proposed design. Most recent digital Hamming-based designs [9, 10, 13, 16, 17] also have cost-performance figures in the same general range. However, the uniqueness of our design and its broader functionality have the potential of leading to further application domains, as reported in [10]. Furthermore, our novel approach can widen the domain of use for switched-capacitor arrays and low-power comparators to applications beyond mere data conversion.

7 Conclusion

In this paper, we have proposed a digital-analog design for a Hamming weight filtering circuit that classifies an input sequence of length N between two Hamming weight boundaries, thus essentially acting as a Hamming mid-pass filter. In other words, a 4-bit input sequence can be checked against one of the ranges of $0 < XH < 4$; $1 < XH < 4$; $2 < XH < 4$; $0 < XH < 3$; $1 < XH < 3$; $0 < XH < 2$;

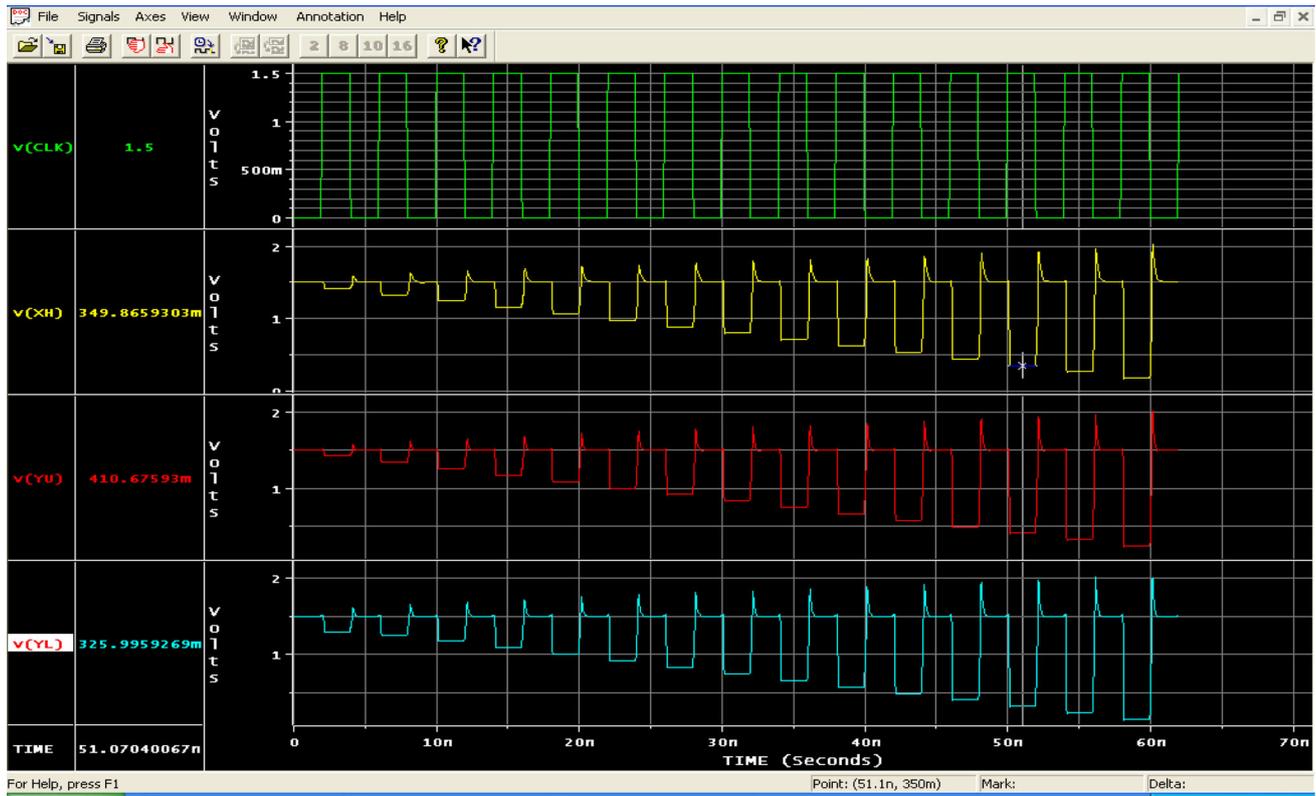


Fig. 4 HSPICE waveforms when the 16-bit Hamming weight XH is sequentially increased, while YL and YU define the ranges for each new XH . Horizontal time scale is in nanoseconds and vertical output scale is in volts ($CLK = 0.25$ GHz, $V_{dd} = 1.5$ V), using TSMC 0.15 μm technology

Table 3 Circuit margin resolution for comparators for different values of N

	Theoretical	HSPICE simulation	Error in percent
4-bit	0.125	0.119	4.8
8-bit	0.0625	0.0576	7.7
16-bit	0.03125	0.02702	13.4

$XH > 4$. If the input sequence has a Hamming weight in the specified range, then the circuit becomes transparent and forwards the input to output; otherwise the all-0s pattern appears at the circuit output.

Our circuit is implemented using the switched-capacitor array structure and screens the Hamming input signal to

uniform capacitances. At the output side, a series of weighted capacitances are tailored to our desired ranges. Mathematical relationships allow us to derive the capacitance weights to fit any input length N . The switched-capacitor array accumulates the charges through only two clocking phases, corresponding to low and high parts of a system clock, thus obviating the need for any extra non-overlap circuit and phase generator, such as DLL circuit.

Our design has been implemented using 0.15 μm TSMC technology for $N = 4, 8,$ and 16 bits, and the result simulated at 500 and 250 MHz, showing a power consumption of 1.37 mW. The design draws power due to the use of two analog comparators in order to classify the Hamming input signal range. The comparators are required

Table 4 Assessment of our design against that of Parhami [9] for Hamming-weight comparisons

Width (bits)	CMOS transistor count		Latency (ns)		Power (mW @4ns)	
	Parhami	Proposed*	Parhami	Proposed*	Parhami	Proposed*
4	152	48	1.65	0.56	2.1	0.94
8	280	96	2.33	0.64	3.8	1.9
12	392	144	2.80	0.91	7.2	3.2
16	496	192	3.12	1.32	10.3	4.1
24	704	288	3.42	1.58	15.6	6.8
32	896	384	3.71	1.83	20.4	9.9

* Please see the text for exclusions and other factors to be taken into account for a fair comparison

to compare between two accumulated voltage charges on the order of 10 mV. The HSPICE simulation results show that the circuit can produce a value at 500 MHz for $N = 4$ or 8 and at 250 MHz for $N = 16$.

References

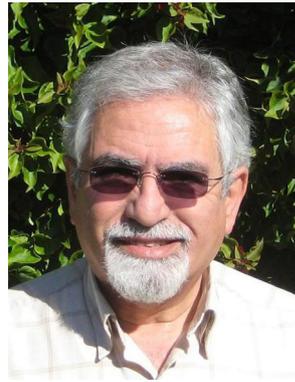
- Abdel-Hafeez, S., Gordon-Ross, A., & Parhami, B. (2013). Scalable digital CMOS comparator using a parallel prefix tree. *IEEE Transactions on VLSI Systems*, 21(11), 1989–1998.
- Abdel-Hafeez, S., & Gordon-Ross, A. (2011). A gigahertz digital CMOS divide-by- N frequency divider based on a state look-ahead structure. *Journal of Circuits, Systems, and Signal Processing*, 30(6), 1549–1572.
- Kohavi, Z., & Jha, N. K. (2009). *Switching and Finite Automata Theory*. Englewood Cliffs: Prentice Hall.
- Ikeda, M., and Asada, K. (1998). Time-domain minimum-distance detector and its application to low power coding scheme on chip interface. In *Proceedings of the 24th European Solid-State Circuits Conference*, pp. 464–467.
- Sklar, B. (1988). *Digital communications fundamentals and applications*. Englewood Cliffs: Prentice Hall.
- Gaitanis, N., Kapogianopoulos, G., and Karras, DA. (1993). Pattern classification using a generalized hamming distance metric. In *Proceedings of International Joint Conference on Neural Networks*. (Vol. 2, pp. 1293–1296).
- Sorokine, V., and Pasupathy, S. (1998). On the hamming weight of binary sequences and linear complexity. In *Proceedings of IEEE International Symposium on Information Theory*. pp. 387–394.
- Ngai, C.-K., Yeung, R. W., & Zhang, Z. (2011). Network generalized hamming weight. *IEEE Transactions on Information Theory*, 57(2), 1136–1143.
- Parhami, B. (2009). Efficient hamming weight comparators for binary vectors based on accumulative and up/down parallel counters. *IEEE Transactions on Circuits and Systems II*, 56(2), 167–171.
- Sklyarov, V., & Skliarova, I. (2013). Digital hamming weight and distance analyzers for binary vectors and matrices. *International Journal of Innovative Computing, Information and Control*, 9(12), 4825–4849.
- Fujino, M., and Moshnyaga, V. G. (2002) An efficient hamming distance comparator for low-power applications. In *Proceedings of 9th International Conferences on Electronics, Circuits and Systems*, (Vol. 2, pp. 641–644).
- King, D. B. S., Simpson, R. J., Moore, C., & MacDiarmid, I. P. (1998). Digital n -tuple hamming comparator for weightless systems. *Electronics Letters*, 34(22), 2103–2104.
- Piestrak, S. J. (2007). Efficient hamming weight comparators for binary vectors. *Electronics Letters*, 43(11), 611–612.
- Chang, J.-C. (2006). Distance-increasing mappings from binary vectors to permutations that increase hamming distances by at least two. *IEEE Transactions on Information Theory*, 52(4), 1683–1689.
- Pedroni, V. A. (2003). Compact fixed-threshold and two-vector hamming comparators. *Electronics Letters*, 39(24), 1705–1706.
- Bharghava, R., Abinash, R., Purini, S., Regeti, G. (2010). Inexact decision circuits: an application to hamming weight threshold voting. In *Proceedings of International Conferences on VLSI Design*. pp. 158–163.
- Pappalardo, F., Pennisi, M., Motta, S., Calonaci, C., & Mastriani, E. (2009). HAMFAST: fast hamming distance computation. In *Proceedings of World Congress on Computer Science and Information Engineering*, (Vol. 1, pp. 569–572).
- Asada, K., Kumatsu, S., and Ikeda, M. (1999). Associative memory with minimum hamming distance detector and its application to bus data encoding. In *Proceedings of IEEE Asia-Pacific Application-Specific Integrated Circuits Conferences*, (pp. 161–166).
- Abdel-hafeez, S. (2010) A new high-speed SAR ADC architecture. In *Proceedings of 11th International Workshop Symbolic and Numerical Methods, Modeling and Applications to Circuit Design*, (pp. 1–5).
- Abdel-Hafeez, S., & Parhami, B. (2013). High-speed and low-power scalable hamming weight comparator based on a non-weighted switched-capacitor array. *Analog Integrated Circuits and Signal Processing*, 75(3), 417–434.
- Gaitanis, N., Kapogianopoulos, G., & Karras, D. A. (1993). Pattern classification using a generalised hamming distance metric. In *Proceedings of International Joint Conference on Neural Networks*, (Vol. 2, pp. 1293–1296).
- Parhami, B., & Avizienis, A. (1978). Detection of storage errors in mass memories using arithmetic error codes. *IEEE Transactions on Computers*, 27(4), 302–308.
- Barral, C., Coron, J.-S., & Naccache, D. (2004). Externalized fingerprint matching. In *Proceedings on 1st International Conferences on Biometric Authentication*, (pp. 309–315).
- Abdel-hafeez, S., Harb, S. M., & Lee, K. M. (2011). “On-chip jitter measurement architecture using a delay-locked loop with Vernier delay line, to the order of giga hertz. In *Proceedings of 18th International Conferences on Mixed Design on Integrated Circuits and Systems*, (pp. 502–506).
- Piestrak, S. J. (1997). Design of encoders and self-testing checkers for some systematic unidirectional error detecting codes. In *Proceedings of International Symposium on Defect and Fault Tolerance in VLSI Systems*, (pp. 119–127).
- Komatsu, S., Ikeda, M., & Asada, K. (2001). Bus data encoding with coupling-driven adaptive code-book method for low power data transmission. In *Proceedings of 27th European Solid-State Circuits Conferences*, (pp. 297–300).
- HSPICE, Synopsys, 2010. <http://www.synopsys.com>.
- Taiwan Semiconductor Manufacturing Corp., 0.15 μm CMOS ASIC Process Digests. 2002.
- Hsieh, M.-H., Chen, L.-H., Liu, S.-I., & Chen, C. C. (2012). A 6.7 MHz-to-1.24 GHz 0.0318mm² fast-locking all-digital DLL in 90 nm CMOS. In *Proceedings of IEEE International Solid-State Circuits Conferences*, (pp. 244–246).
- Kim, Y.-S., Lee, S.-K., Park, H.-J., & Sim, J.-Y. (2011). A 110 MHz to 1.4 GHz locking 40-phase all-digital DLL. *IEEE Journal of Solid-State Circuits*, 46(2), 435–444.
- Agens, A., Bonizzoni, E., Malcovati, P., & Maloberti, F. (2010). An ultra-low power successive approximation A/D converter with time-domain comparator. *Analog Integrated Circuits and Signal Processing*, 64(2), 183–190.
- Fiorenza, J. K., Sepke, T., Holloway, P., Lee, H.-S., & Sodini, C. G. (2006). Comparator-based switched-capacitor circuits for scaled CMOS technologies. *IEEE Journal of Solid-State Circuits*, 41(12), 2658–2668.
- Ogawa S., & Watanabe, K. (1992). Clock-feedthrough compensated switched-capacitor circuits. In *Proceedings of IEEE International Symposium on Circuits and Systems*, (Vol. 3, pp. 1195–1198).
- Lee, S.-H., & Song, B.-S. (1992). Digital-domain calibration of multistep analog-to-digital converters. *IEEE Journal of Solid-State Circuits*, 27(12), 1679–1688.
- Masuzawa, A. (2007) Design challenges of analog-to-digital converters in nanoscale CMOS. *IEICE Transactions on Electronics*, (Vol. E90-C, No. 4, pp. 779–785).

36. Hastings, A. (2001). *The art of analog layout*. Englewood Cliffs: Prentice Hall.
37. McNutt, M. J., LeMarquis, S., & Dunkley, J. L. (1994). Systematic capacitance matching errors and corrective layout procedures. *IEEE Journal of Solid-State Circuits*, 29(5), 611–616.
38. Sayed, D., and Dessouky, M. (2002). Automatic generation of common-centroid arrays with arbitrary capacitor ratio. In *Proceedings of Design, Automation and Test in Europe Conference*, (pp. 1530–1591).
39. Siyu, Y., Hui, Z., Wenhui, F., Ting, Y., & Zhiliang, H. (2011). A low power 12-bit 200-kS/s SAR ADC with a differential time domain comparator. *Journal of Semiconductors*. Vol. 32, No. 3, pp. 035002-1:035002-6.
40. Uyemura, J. P. (1999). *CMOS logic circuit design*. Netherlands: Kluwer.
41. Razavi, B. (2001). *Design of analog CMOS integrated circuits*. New York: McGraw-Hill.



Saleh Abdel-hafeez received a Ph.D. in computer engineering from the University of Texas at El Paso and M.S. from New Mexico State University. He was a senior member of technical staff at S3 Inc. and Viatechnologies.com in the area of mixed-signal IC design. He also was an Adjunct Professor of Computer Engineering at Santa Clara University from 1998 to 2002. He has two US patents, numbered 6,265,509 and 6,356,509, with S3 Inc. His current research interests are in the

areas of high speed ICs, computer arithmetic algorithms, and mixed-signal design. Dr. Abdel-hafeez is currently the Chairman of Computer Engineering Department at Jordan University of Science and Technology.



Behrooz Parhami (Ph.D., 1973, University of California, Los Angeles) is Professor of Electrical and Computer Engineering, and former Associate Dean for Academic Personnel, College of Engineering, at University of California, Santa Barbara, where he teaches and does research in computer arithmetic, parallel processing, and dependable computing. A Fellow of IEEE, IET, and British Computer Society, and recipient of several other awards (including a most-cited

paper award from *J. Parallel & Distributed Computing*), he has written six textbooks and more than 280 peer-reviewed technical papers. Professionally, he serves on journal editorial boards and conference program committees and is also active in technical consulting.



Mohammad Al-Hammouri received the B.Sc. Degree in Computer Engineering in 2004 and the M.S. degree in Computer Engineering in 2007, both from Jordan University of Science and Technology, Irbid, Jordan. He is currently a full-time lecturer at the same University. His research interests include VLSI components design and computer architecture.