A Charge Recycling Differential Noise Immune Perceptron

Jabulani Nyathi, Valeriu Beiu, Suryanarayana Tatapudi, and David J. Betowski

School of Electrical Engineering and Computer Science
Washington State University
Pullman, Washington 99164-2752, USA
Email: {jabu, vbeiu, statapud, dbetowsk}@eecs.wsu.edu

Abstract—**This paper proposes a new differential neural inspired gate with improved noise immunity. The charge recycling differential noise-immune threshold logic (CRD-NTL) perceptron is based on combining the split-level precharge differential logic, with a technique for enhancing noise immunity of threshold logic gates: noise suppression logic. Another idea included in the design of the CRD-NTL gate is the use of two threshold logic banks implementing the function *(f)* and its inverse *(f_bar)*, and working in conjunction with the noise suppression logic blocks for enhanced performance. Characterization of the new gate has been performed by extensive simulation in 0.25 μm CMOS technology at 2.5 V.**

Index Terms—**Charge recycling, differential, noise immune, perceptron, threshold logic, parameter variations.**

I. INTRODUCTION

The work of Warren McCulloch and Walter Pitts entitled *A Logical Calculus of the Ideas Immanent in Nervous Activity*, published in 1943 is considered to be the starting point into research in neural networks [1]. For modeling a neuron, they introduced the threshold logic gate (TLG) represented by the function:

$$f(x_1,\ldots,x_\Delta) = \operatorname{sgn}\sum_{i=1}^{\Delta}(w_i x_i - \theta) \tag{1}$$

where w_i is the synaptic weight associated with input x_i, θ is the threshold, and Δ is the fan-in. In order to allay concerns that a neuron is a TLG, the threshold logic (TL) model has been tested on a spike train generated by the Hodgkin-Huxley model with a stochastic input [2]. The result was that the TL model correctly predicted nearly 90% of the spikes, justifying the description of a neuron as a TLG.

The tremendous impetus of VLSI technology has made neuro-computer design a lively research topic. There are many theoretical circuit complexity results showing that TL circuits (TLCs) are more powerful than classical Boolean circuits [3]. Beside, TLGs (or their variations) have been used in MIPS R2010 [4], SUN Sparc V9 [5], a CMOS fingerprint sensor array [6], and very recently in the Itanium 2 microprocessor [7]. Finally, the emerging nano devices (e.g., resonant tunneling, single electron, and molecular) have been used for quite some time for implementing TLGs and TLCs. These emerging nano devices have led to many different TLG implementations such as those presented in [8], [9], [10].

In this paper we shall focus primarily on charge recycling differential TLGs, with Section II providing a review of some implementations of these gates, Section III focusing on a new noise-immune differential charge recycling perceptron and some concluding remarks in Section IV. The charge recycling differential noise-immune perceptron is based on combining the split-level precharge differential logic (SPDL) [11], with a technique for enhancing the noise immunity of TLGs [12], [13], [14], known as noise suppression logic (NSL). Another idea included in the design of the charge recycling perceptron is the use of two TL banks implementing the function *f* and its inverse *f_bar* [15]. This technique works very well in conjunction with the NSL enhancing the speed performance of the TLG. The terms *perceptron* and *TLG* will be used interchangeably in this paper.

II. REVIEW OF HIGH SPEED LOW POWER GATE IMPLEMENTATIONS

A. Differential and Differential Charge Recycling Gates

Energy efficiency design has been the driving force behind many of the differential gate implementations. These differential gates dissipate less power, while operating at very high frequencies. A list of differential Boolean gates includes: differential cascode voltage switch DCVS (1984), differential split-level logic DSLL (1985), sample-set differential logic SSDL (1986), differential pass transistor logic DPTL (1987), enable/disable CMOS differential logic ECDL (1988), latched CMOS differential logic LCDL (1991), differential current switch logic DCSL (1996), current sensing differential logic CSDL (1998), no-race charge-recycling differential logic NCDL (1999), and many more.

The ongoing challenge to maximize device-switching speed while minimizing the power consumption has seen further improvements on differential gates. Some of the designs enumerated above use fully differential dynamic logic with precharge and evaluate phases. While negligible static

0-7803-8359-1/04/$20.00 ©2004 IEEE

power is dissipated, the dynamic power dissipation increases significantly over traditional static CMOS, since the output must swing from rail to rail during each clock cycle. A novel method of reducing the delay and power consumption is charge recycling. The technique was first proposed in 1995 as a time-multiplexing scheme to enable sharing of bus wires [16], [17], [18].

The operation of charge recycling differential gates involves precharging several nodes of the gate during which both differential outputs switch to $V_{DD}/2$ by distributing the charge. During the evaluation phase the outputs only need to swing up or down by $V_{DD}/2$, rather than V_{DD}, decreasing the propagation delay and expectedly the power consumption. Charge recycling is achieved by placing a transistor "switch" between the output signal and its complement. The transistor placed between the differential outputs is not conducting during the evaluation phase, allowing the outputs to be driven to their full voltage levels, by charging or discharging the load and parasitic capacitances. When in the precharge phase, the switch closes, short-circuiting the differential outputs, which must be isolated from V_{DD} and GND so that the charge stored on the load and parasitic capacitances is redistributed. We list a few gates that employ the charge recycling technique, and highlight the improvements reported.

- The first charge recycling differential logic gate (CRDL) utilized a complementary pass transistor network and a differential acceleration buffer [18], [19]. The pass transistor network realizes the Boolean function and drives both the acceleration buffer and the output load. The acceleration buffer, which expedites the rise and fall times, basically is a pair of cross-coupled inverters isolated from ground during the precharge phase. While offering higher speeds and lower power dissipation than differential cascode voltage switch (DVCS) logic [20], [21], the design loses the recycled charge during the precharge phase. This occurs when the output and its inverse become $V_{DD}/2$ and $V_{GSP} = V_{DD}/2 > |V_{TP}|$, turning on the pMOS transistors, and creating a direct path from V_{DD} to GND. This problem is overcome by connecting the substrate of the pMOS transistors to a voltage higher than V_{DD} (known as body bias), requiring two voltage sources, so that V_{TP} is greater than $V_{DD}/2$.
- The dual voltage obstacle was overcome with the introduction of half-rail differential logic (HRDL) in 1997 [22], [23], [24]. This design improves the CRDL, as isolating the output nodes from V_{DD} and GND during precharge, by using a multiple phase clock and eliminating the requirement for multiple bias voltages. Although the speed degrades slightly, power and reliability improve over the original CRDL design. The output circuitry, however, has reduced driving capability.
- A significant improvement to charge recycling logic came in 2001 with the introduction of SPDL [11]. As opposed to CRDL or HRDL, the functional logic network drives only the acceleration buffer, resulting in a reduction of the logic circuitry. Using only a single-phase clock and its complement, the output nodes are completely isolated from V_{DD} and GND. The power consumption, speed, and reliability are reported to be improved compared to CRDL or HRDL designs.

Since the introduction of CRDL and HRDL logic, several improvements have been made to increase the speed of the function logic, including nMOS [23], [24], [25], complementary pass transistor logic (CPL) [23], [26]–[31], and CMOS.

B. Differential TLG Implementations

TLGs have also employed the differential gate design approach and we present a brief review of such TLGs in this sub-section, while the interested reader should consult references [10], [32]. Two basic approaches for implementing TLGs are: capacitive and conductance. The concept underlying capacitive TLGs is the use of an array of capacitors implementing the weighted sum of inputs. The idea was introduced as early as 1966 [33]. Capacitive TLGs can be divided into two classes, namely: capacitive threshold logic (CTL), and neuron MOS (or vMOS). A few comparisons [34], [35], [36], draw the following conclusions:

- the vMOS operation is simpler than that of the CTL;
- the maximum attainable fan-in by vMOS is an order of magnitude less than that of CTL gate;
- the delay has a logarithmic dependence with respect to large fan-ins (fan-in ≤ 255 in [35], fan-in ≤ 64 in [36], while for small fan-ins (fan-in ≤ 20 [34]) the behavior looks linear: $1 + 0.35\Delta$ (where Δ is the fan-in).

The idea of using switched capacitors, switches, and inverters, and taking advantage of the inherent saturation of the inverters to implement the perceptron's non-linearity was originally introduced in 1987 [37]. This first approach required a somewhat complex three-phase clock. It has quickly evolved into a simpler two-phase clock solution [35]: the capacitive threshold logic (CTL). This has large fan-in capability (up to 255), but also large delays and area, and DC power consumption. A differential version is the balanced-CTL (B-CTL) [38]. The requirement for a highly precise reference voltage is eliminated here by implementing functions with thresholds equal to zero. Two banks of capacitors connected to a differential amplifier form the basic structure, with one additional half-capacitor unbalancing the voltage level. B-CTL gates are reported to be faster than CIAL gates [39]. Neuron MOS TLGs are based on an idea introduced in the mid 60s [33]. It was rediscovered in 1991 [40]. The static vMOS TLG is very simple and compact, but has DC power consumption. This static power can be eliminated and the speed increased by a current comparison between a vMOS transistor and a reference device, using a positive feedback circuit.

- One configuration is the sense-amplifier vMOS TL [41]. It employs a current controlled latch-sense amplifier circuit. Variations can be found in [42]. This solution is similar to the digital comparator from [43]. Speed improvements of five times, and power savings over the static vMOS were reported [42].
- Another variation, CMOS capacitor coupling logic (C^3L), uses the capacitor coupling technique and a current sense amplifier [44]. Fluctuations of the device parameters are compensated by the differential configuration.
- The charge recycling threshold logic (CRTL) gate [45], is based on CRDL [19]. CRTL gates exhibit high speed (even for large fan-ins), while also having low power consumption. CRTL gates achieve the highest speed and 15-20% lower power consumption when compared with clocked vMOS [41], C^3L [44], and LCTL [46].
- A self-timed threshold logic (STTL) has been proposed in [47]. The gate is based on a cross-coupled nMOS transistor pair. The enable signals are passed to the next stage, being propagated in a self-timed fashion. The solution is low power, and eliminates the clock at the expense of a double rail signaling and the additional "enable generate" block.

The other class of differential TLG implementations is the current/conductance category. Two parallel connected banks of nMOS transistors are used for implementing the weighting operation, followed by a current CMOS comparator for the threshold operation.

- The operation of cross-coupled inverters with asymmetrical loads (CIAL) was exploited to implement digital (bus) comparators [43], a particular TLG.
- A generic latch-type TL (LCTL) gate was proposed in [46]. It consists of a CMOS current controlled latch providing both the output and its complement, and two input arrays having an equal number of parallel transistors whose gates are the inputs of the TLG. Two extra transistors guarantee correct operation for the case when the weighted sum of inputs is equal to the threshold value. Current flows only during transitions.
- The speed of LCTL was improved in [39], where the nMOS banks are external to the latch. It is called cross-couple inverters with asymmetrical loads threshold logic (CIALTL), but it is different from CIAL.
- The circuit arrangement for realizing logic elements that can be represented by threshold value equations patented by Prange et al. [48] is a simplified CIAL.
- Single input current-sensing differential logic (SCSDL) is based on CSDL [50]. Yield analyses in 0.35 μm [49] CMOS has shown that fan-in ≤ 14 are achievable.
- Differential current-switch threshold logic (DCSTL) [51] is based on DCSL [52]. It restricts the voltage swing of the internal nodes for lowering the power consumption. Reported experiments show that DCSTL exhibits better power-delay product than: LCTL [46] and CIALTL [39].
- Current-mode threshold logic (CMTL) [53] achieves low power by limiting the voltage swing on interconnects and on the internal nodes. Various clocked cross-coupled loads have led to discharged CMTL (DCMTL) and equalized CMTL (ECMTL).

All the TLGs based on current comparisons are relatively *sensitive to noise and mismatch of process parameters.* Reliability can be improved by known layout and circuit techniques where the devices behavior is matched (substrate voltage control, shield and isolations, centroid layout) for reducing statistical parameter variations.

III. A NEW DIFFERENTIAL CHARGE RECYCLING PERCEPTRON

All the differential TL solutions cited compare the sum of weights with a threshold [41], [44], [45], [47], [48], [49] or compare two weighted sums [38], [39], [43], [46], corresponding to the positive and the negative weights. The functions implemented in the second case have threshold $\theta = 0$. Additional transistors are needed to differentiate the case when the two weighted sums are equal. A quite simple but efficient idea is to implement the function *f* with one TL bank, while implementing *f_bar* with another TL bank. It is well known that if *f* is a TL function, *f_bar* is a TL function having the same weights w_i (the inputs x_i being the inverted x_{i}*_bar*), and the threshold changed (for the TL bank implementing *f_bar*). The fact that *f* and *f_bar* always have transitions in opposite directions leads to increased speed, and also to better noise margins [15]. The noise margins can be enhanced even more, while only marginally increasing the speed, by increasing the gap using a non-linear solution for both *f* and *f_bar*, like the noise suppression logic (NSL) reported in [12], [13], [14]. The resulting solution is noise immune by design (and also very fast). That is why this technique is called noise-immune threshold logic (NTL). It can be used in conjunction with any differential implementation.

In this section we describe the new charge recycling differential noise-immune perceptron, providing details on

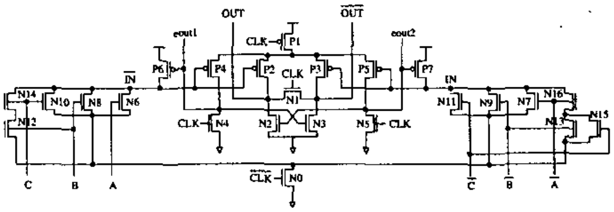

Fig. 1. Differential charge recycling perceptron.

how NTL works in conjunction with SPDL [11]. SPDL is an improvement on other charge recycling differential logic such as HRDL and CRDL in terms of power dissipation, propagation delay, increased reliability, avoiding metastable states, and large fan-out. The 3-input charge recycling differential noise immune perceptron shown in Figure 1 has a total of 24 appropriately sized transistors.

It is a special case of the differential gate in that it lacks the traditional cross-coupled pull-up network and thus requires a few additional transistors to produce the differential outputs. The gate consists of the SPDL block, and two evaluate and two NSL blocks. The SPDL part is represented by transistors *N0-N5* and *P1-P9* (for further details see [11]).

The perceptron is initialized when the clock is high, resulting in transistors N4, N5, P6 and P7 being turned ON. This state sets nodes IN and IN*_bar* to logic 1 while transistors N2, N3, P2, P3, P4 and P5 are turned OFF. Of particular interest during this initialization stage is that transistor N1 conducts, enabling charge recycling between the differential outputs labeled OUT and OUT*_bar*. Charge recycling makes these outputs switch from V_{DD} or GND to $V_{DD}/2$. It is also during this initialization stage that the inputs can be changed.

The output function is implemented by the n-network comprising transistors N6, N8 and N10 for one bank and N7, N9 and N11 for the other bank. The NSL devices are N12 and N14 for one TL bank, and N13, N15 and N16 for the other. Transistor N0 provides a path to GND for either bank. The evaluation state involves turning transistors N0 and P1 ON after the inputs settle. Both the n- and p-network devices are sized to ensure that the weights (w_i) are represented appropriately resulting in the TLG functioning properly. The proper sizing of the transistors N6-N11 for encoding the weights associated with the inputs is such that the *W/L* of N6 is four times that of N8, and N10. The same is true for the right bank, where the *W/L* of N11 is four times that of N7, and N9. The NSL logic blocks (N12 and N14 along with N13 and N15-N16) have to be sized at least as large as N6 (making them larger will always improve on the noise margins [14], but might degrade the speed). Table I summarizes the transistor sizes for one possible implementation.

TABLE I
PERCEPTRON TRANSISTOR DIMENSIONS

Device Level	The Width-Length Ratio (W/L)
N0	64
N1, N2, N3, N4 and N5	4
N6, N12, N13, N14, N15 and N16	12
N7 and N9	1.5
N8 and N10	3
N11	6
P1	64
P2, P3, P4 and P5	8
P6 and P7	21
P8 and P9	Not implemented

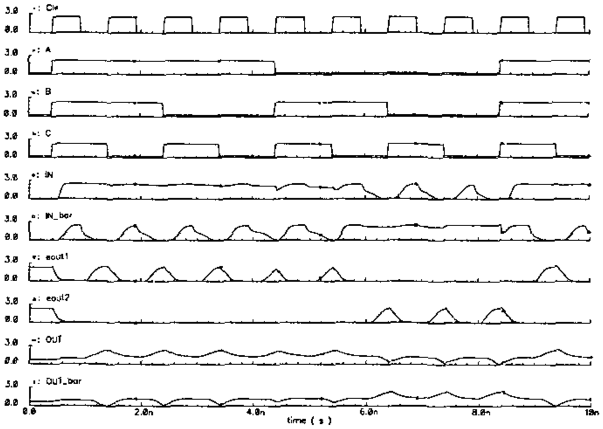

Fig. 2. Critical nodes of perceptron running at 1000 MHz.

The perceptron has been evaluated for performance in terms of power dissipation, and switching delays. The results reported here are based on a layout of the gate done in standard CMOS 0.25 μm technology at 2.5 V. The gate is subjected to a load of eight minimum sized inverters on both differential outputs. Figure 2 depicts traces of the perceptron's critical nodes. In this figure we show results obtained at a frequency of 1000 MHz, while simulations at 100 MHz and 500 MHz have been performed to ensure proper functionality. Different simulations factoring environmental parameter (temperature and power supply voltage) variations have also been performed to fully characterize the gate. The average current values at 100 MHz, 500 MHz, and 1000 MHz (when running continuously) were unexpectedly high: 289 μA, 739 μA, and 1.04 mA, respectively. This can only partly be explained by the load and high temperature we have used.

Removing transistor N1 allows the differential outputs to have a full swing from GND to V_{DD}, and we have taken measurements for such a configuration and determined that the perceptron with charge recycling capability reduces switching delays by 38%. It is much faster to switch the output node from $V_{DD}/2$ to V_{DD} or from $V_{DD}/2$ to GND as opposed to having a rail-to-rail switching activity at the output nodes during evaluation. Figure 3 shows simulations of the two configurations, one with charge recycling capability, and one without.

The average current over all possible input combinations shows that the charge recycling perceptron draws nearly the same current as the one without charge recycling capability. The current values are high and there is only a slim 6% difference in favor of the perceptron with charge recycling capability. This small difference in average current is due to the fact that even when some input patterns result in the output remaining at logic 1 or logic 0, the differential outputs still have to be driven from $V_{DD}/2$ to either V_{DD} or GND, while the outputs remain constant for the design without

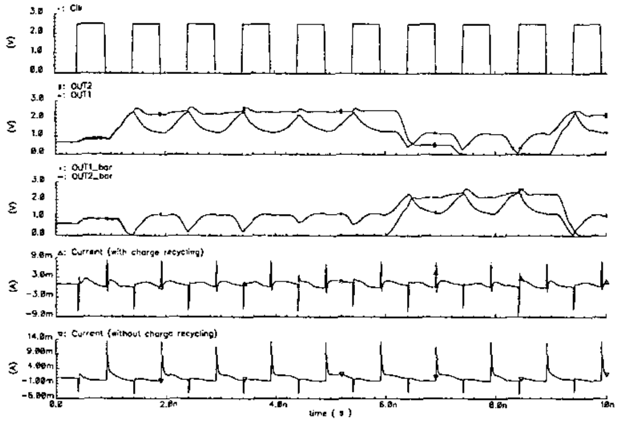

Fig. 3. Differential outputs comparison.

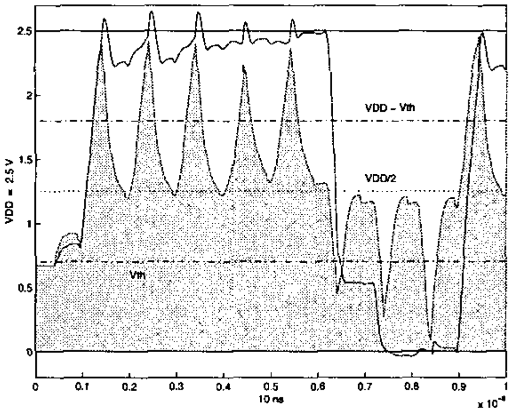

Fig. 4. Switching activity on output nodes.

charge recycling capability. We note that the current simulations (shown in Figure 3) for the two configurations look very similar. This is due to the fact that dynamic switching occurs at several internal nodes even when the input patterns being evaluated result in no changes at the output nodes. Figure 4 shows overlapped plots of the output nodes for the two configurations. Values of interest such as V_{th}, $V_{DD}/2$ and $V_{DD} - V_{th}$ are clearly marked. The delays reported here are measured at 10% to 90% of the steady state voltage. *The perceptron with the charge recycling capability shows a slight disadvantage*, by repeatedly switching from V_{DD} to $V_{DD}/2$ for input patterns that evaluate to logic 1 while the perceptron with no charge recycling capability's outputs show no significant changes. This behavior also takes place when input patterns that evaluate to logic 0 are applied, the charge recycling perceptron repeatedly switches from GND to $V_{DD}/2$. The other perceptron's output remains steady at logic 0. The only gain in performance occurs when output nodes have to be switched from logic 1 to logic 0 or vise versa. In this case, the charge recycling perceptron discharges or charges the nodes faster as depicted in the plot of Figure 4.

This work demonstrates that the charge recycling perceptron can be implemented using two banks for *f* and *f_bar* in conjunction with NSL blocks. An ideal approach in measuring the efficiency of the perceptron in terms of power dissipation, and speed entails comparing the design with purely CMOS, pseudo-NMOS, and Domino implementations. Most of the design examples cited were implemented in older technology nodes making it difficult to directly compare them in terms of delay and power dissipation. A fair comparison would dictate that the circuits be simulated at the same technology node.

IV. CONCLUDING REMARKS

The present state-of-the-art shows a large variety of differential TLGs. Some of these are quite advanced [11], [32], [47] and they allow for drastic reductions of the dissipated power (when compared to earlier implementations). A novel differential charge recycling TLG with NSL has been proposed. It incorporates two new ideas: using *f* and *f_bar*, and adding nonlinear data dependent terms. For the selected frequencies it shows speed improvements of 34-38% over its counterpart dynamic perceptron without charge recycling. This work has shown that differential logic gates are fast but not necessarily suitable for low power dissipation. The basic disadvantages include: the need to have two networks (for *out* and *out_bar*) resulting in increased switching activity, and the need to reduce the pull-down stack for improved delays (which unfortunately increases the leakage currents). As far as we know, the best solution for low power differential logic was that presented in [54]. It uses an inverter's short-circuit current to drive critical nodes. We expect that further scaling [55] will accentuate differential logic gates' need for power, as opposed to simpler gates—irrespective of the techniques employed—due to increased leakage and higher switching activity. We conclude that differential structures are seemingly not the best solution for scaled CMOS. Simpler gates with fewer transistors might do better. These could employ techniques such as adaptive body biasing and sub-threshold power supply voltages for enhanced overall performance.

REFERENCES

[1] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *Bull. Math. Biophysiol*, Vol. 5, 1943, pp. 115–133.

[2] W. M. Kistler, W. Gerstner, and J. L. van Hemmen, "Reduction of the Hodgkin-Huxley equations to a single variable threshold model," *Neural Computation*, vol. 9, 1997, pp. 1015–1045.

[3] V. Beiu, "A survey of perceptron circuit complexity results," *Proc. IJCNN*, Jul. 2003, vol. 2, pp. 989–994.

[4] M. G. Johnson, "A symmetric CMOS NOR gate for high-speed applications," *IEEE J. Solid-State Circ.*, Vol. 23, 1988, pp. 1233–36.

[5] L. A. Lev, "Fast static cascode logic gate," U.S. Patent 5,438,283 (Aug. 1, 1995).

[6] S. Jung, R. Thewes, T. Scheiter, K. F. Goser, and W. Weber, "A low-power and high-performance CMOS fingerprint sensing and encoding architecture," *IEEE J. Solid-State Circ.*, vol. 34, 1999, pp. 978–984.

[7] S. D. Naffziger, G. Colon-Bonet, T. Fischer, R. Reidlinger, T. J. Sullivan, and T. Grutkowski, "The implementation of the Itanium 2 processor," *IEEE J. Solid-State Circ.*, vol. 37, 2002, pp. 1448–1460.

[8] R. Compañó, L. Molenkamp, and D.J. Paul (Eds.), *Technology Roadmap for Nanoelectronics*, European Commission, IST Programme, Future and Emerging Technologies (FET), 2000. Available: http://www.cordis.lu/ esprit/src/melna-rm.htm.

[9] K.F. Goser, C. Pacha, A. Kanstein, and M.L. Rossmann, "Aspects of system and circuits for nanoelectronics," *Proc. IEEE*, vol. 85, Apr. 1997, pp. 558–573.

[10] V. Beiu, J. M. Quintana, and M. J. Avedillo, "VLSI implementations of threshold logic: A comprehensive survey," *IEEE Trans. Neural Networks*, vol. 14, Sep. 2003, pp. 1217–1243.

[11] J. Lee, J. Park, B. Song, and W. Kim, "Split-level precharge differential logic: A new type of high-speed charge-recycling differential logic," *IEEE J. Solid-State Circ.*, vol. 36, 2001, pp. 1276–1280.

[12] V. Beiu, "Ultra-fast noise immune CMOS threshold gates." *Proc. MWSCAS*, 2000, pp. 1310–1313.

[13] V. Beiu, "On higher order noise immune perceptrons," *Proc. IJCNN* vol. 1, 2001, pp. 246–251.

[14] V. Beiu, "Noise tolerant conductance-based logic gate and methods of operation and manufacturing thereof," U.S. Patent 6,430,585 (Aug. 6, 2002).

[15] V. Beiu, "Low-power differential conductance-based logic gate and method of operation thereof," U.S. Patent 6,580,296 (Jun. 17, 2003).

[16] H. Yamauchi, and A. Matsuzawa, "A low power signal-swing suppressing strategy using time-multiplexed differential data-transfer scheme," *Proc. Symp. Low Power Electronics*, 1995, pp. 48–49.

[17] H. Yamauchi, and A. Matsuzawa, "A signal-swing suppressing strategy for power and layout area savings using time-multiplexed differential data-transfer scheme," *IEEE J. Solid-State Circ.*, vol. 31, Sep. 1996, pp. 1285–1294.

[18] H. Morimura, and N. Shibata, "A 1-V 1-Mb SRAM for portable equipment," *Proc. Intl. Symp. Low Power Electronics & Design, ISLPED*, 1996, pp. 61–66.

[19] B.-S. Kong, J.-S. Choi, S.-J. Lee, and K. Lee, "Charge recycling differential logic for low-power application," *Proc. ISSCC*, 1996, pp. 302–303, 462.

[20] B.-S. Kong, J.-D. Im, Y.-C. Kim, S.-J. Jang, and Y.-H., Jun, "Asynchronous sense differential logic," *Proc. ISSCC*, 1999, pp. 284–285.

[21] B.-S. Kong, J.-D. Im, Y.-C. Kim, S.-J. Jang, and Y.-H. Jun, "CMOS differential logic family with self-timing and charge-recycling for high-speed and low-power VLSI," *IEE Proc. Circ. Dev. Sys.*, vol. 150, 2003, pp. 45–50.

[22] S. Y. Choe, G. Rigby, and G. Hellestrand, "Dynamic half rail differential logic for low power," *Proc. ISSCC*, 1997, pp. 1936–1939.

[23] B.-S. Kong, Y.-H. Jun, and K. Lee, "A true single-phase clocking scheme for low-power and high-speed VLSI," *ISSCC*, 1997, pp. 1904–1907.

[24] J.-H. Won, and K. Choi, "Modified half rail differential logic for reduced internal logic swing," *Proc. ISCAS*, 1998, vol. 2, pp. 157–160.

[25] K.Y. Cheung, "CRRDL: A novel charge recovery-recycling differential logic," *Proc. ISCAS*, 2001, vol. 4, pp. 152–153.

[26] S.-M. Yoo, and S.-M. Kang, "CMOS pass-gate no-race charge-recycling logic (CPNCL)," *Proc. ISSCC*, 1999, vol. 1, pp. 226–229.

[27] S.-O. Jung, and S.-M. Kang, "Modular charge recycling pass transistor logic (MCRPL)," *Electr. Lett.*, vol. 36, 2000, pp. 404–405.

[28] H. Lin, Y.-F. Chen, and H.-C. She, "A low-power 3-phase half rail pass-gate differential logic," *Proc. ISCAS*, 2001, vol. 4, pp. 148-151.

[29] A. Inoue, V. Oklobdzija, W. Walker, M. Kai, and T. Izawa, "A low power SOI adder using reduced-swing charge recycling circuits," *Proc. ISSCC*, 2001, pp. 316–317, 451.

[30] A. Abbasian, S.H. Rasoulil, J. Derakshandeh, A. Afzali-Kushal, and M. Nourani, "Race-free CMOS pass-gate charge recycling logic (FCPCL) for low power applications," *Proc. Southwest Symp. Mixed-Signal Design SSMSD*, 2003, pp. 87–89.

[31] A. Abbasian, S.H. Rasoulil, A. Afzali-Kushal, and M. Nourani, "No-race charge recycling complementary pass transistor logic (NCRCPL) for low power applications," *Proc. ISCAS*, 2003, vol. 5, pp. 289–292.

[32] V. Beiu, J.M. Quintana, M.J. Avedillo, and R. Andonie, "Differential implementations of threshold logic gates," *Proc. Intl. Symp. Signal, Circ. & Sys. SCS*, 2003, vol. 2, pp. 489–492.

[33] J. R. Burns, N. J. Trenton, and R. A. Powlus, "Threshold circuit utilizing field effect transistors," U.S. Patent 3,260,863, (Jul. 12 1966).

[34] P. Celinski, S. Al-Sarawi, and D. Abbott "Delay analysis of neuron-MOS and capacitive threshold-logic," *Proc. ICECS*, 2000, vol. 2, pp. 932–935.

[35] H. Özdemir, A. Kepkep, B. Pamir, and Y. Leblebici, and U. Çilingiroglu, "A capacitive threshold-logic gate," *IEEE J. Solid-State Circ.*, vol. 31, 1996, pp. 1141–1150.

[36] M. Pădure, C. Dan, S. D. Coţofană, M. Bodea, and S. Vassiliadis, "Capacitive threshold logic: A designer perspective" *Proc. CAS*, 1999, vol. 1, pp. 81–84.

[37] Y. P. Tsividis, and D. Anastassiou, "Switched-capacitor neural networks." *Electron. Lett.*, vol. 23, 1987, pp. 958–959.

[38] J. L. García, J. L. Ramos, and A. G. Bohórquez, "A balanced capacitive threshold logic gate," *Proc. DCIS*, 2000. Available: http://www.el.uma.es/Ppepefer/DCIS2000.pdf.

[39] J. F. Ramos, J. A. H. López, M. J. Martín, J. C. Tejero, and A. G. Bohórquez, "A threshold logic gate based on clocked coupled inverters," *Intl. J. Electronics*, vol. 84, 1998, pp. 371–382.

[40] T. Shibata, and T. Ohmi, "An intelligent MOS transistor featuring gate-level weighted sum and threshold operations." *IEDM Tech. Digest*, 1991, pp. 919–922.

[41] K. Kotani, T. Shibata, M. Imai, and T. Ohmi, "Clocked-controlled neuron-MOS logic gates," *IEEE Trans. Circ. Sys. II*, vol. 45, 1998, pp. 518–522.

[42] W. Weber, S. J. Prange, R. Thewes, E. Wohlrab, and A. Luck, "On the application of neuron MOS transistor principle for modern VLSI design." *IEEE Trans. Electr. Dev.*, vol. 43, 1996, pp. 1700–1708.

[43] J. A. H. López, J. G. Tejero, J. F. Ramos, and A. G. Bohórquez, "New types of digital comparators," *Proc. ISCAS*, 1995, vol. 1, pp. 29–32.

[44] H. Y. Huang, and T. N. Wang, "CMOS capacitor coupling logic (C^{3}L) logic circuits," *Proc. AP-ASIC*, 2000, pp. 33–36.

[45] P. Celinski, J. F. López, S. Al-Sarawi, and D. Abbott, "Low power, high speed, charge recycling CMOS threshold logic gate," *Electron. Lett.*, vol. 37, 2001, pp. 1067–1069.

[46] M. J. Avellido, J. M. Quintana, A. Rueda, and E. A. Jiménez, "A low-power CMOS threshold-gate." *Electron. Lett.*, vol. 31, Dec. 1995, pp. 2157–2159.

[47] P. Celinski, J. F. López, S. Al-Sarawi, and D. Abbott, "Compact parallel (m, n) counters based on self-timed threshold logic," *Electron. Lett.*, vol. 38, 2002, pp. 633–635.

[48] S. Prange, R. Thewes, E. Wohlrab, and W. Weber, "Circuit arrangement for realizing logic elements that can be represented by threshold value equations," U.S. Patent 5,991,789 (Nov. 23, 1999).

[49] R. Strandberg, and J. Yuan, "Single input current-sensing differential logic (SCSDL)," *Proc. ISCAS*, 2000, vol. 1, pp. 764–767.

[50] J. Park, J. Lee, and W. Kim, "Current sensing differential logic: A CMOS logic for high reliability and flexibility," *IEEE J. Solid-State Circ.*, vol. 34, 1999, pp. 904–908.

[51] M. Pădure, S. D. Coţofană, C. Dan, M. Bodea, and S. Vassiliadis, "A new latch-based threshold logic family," *Proc. CAS*, 2001, vol. 2, pp. 531–534.

[52] D. Somasekhar and K. Roy, "Differential current switch logic: a low power DCVS logic family," *IEEE Journal of Solid-State Circuits*, vol. 31, Jul. 1996, pp 981–991

[53] S. Bobba, and I. N. Hajj, "Current-mode threshold logic gates," *Proc. ICCD*, 2000, pp. 235–240.

[54] A. M. Fahim and M. I. Elmasry, SC2L; "A low power high-performance dynamic differential logic family," *Proc. Intl. Symp. Low Power Electronics & Design ISLPED*, 1999, pp. 88–90.

[55] *International Technology Roadmap for Semiconductor* ITRS, 2003. Available: http://public.itrs.net/Files/2003ITRS/Home2003.htm.