VLSI Implementations of Threshold Logic A Comprehensive Survey

Valeriu Beiu, *Senior Member, IEEE*, José M. Quintana, and María J. Avedillo

***Abstract*—This is an in-depth review paper on silicon implementations of threshold logic gates, covering several decades (*i.e.*, from the early days till now). The paper starts by describing early MOS implementations followed by different VLSI solutions including: capacitive (switched capacitor and floating gate with their variations), conductance/current (pseudo-nMOS and output-wired-inverters—including a plethora of solutions evolved from them—as well as many differential solutions), and shortly mentions other implementations (*e.g.*, based on negative resistance devices and on single electron technologies).**

***Index Terms*— Integrated circuits, neural network hardware, threshold logic, VLSI.**

I. Introduction

RESEARCH on neural networks (NNs) goes back sixty years ago. The seminal year for the development of the "science of mind" was 1943 when the article *A Logical Calculus of the Ideas Immanent in Nervous Activity* by Warren McCulloch and Walter Pitts was published [13]. They introduced the first, very simplified, mathematical model of a neuron operating in an all-or-none fashion: the threshold logic (TL) gate (TLG). It computes the sign of the weighted sum of its inputs:

$$f(x_1, \ldots, x_n) = \operatorname{sgn}\left(w_1 x_1 + \ldots + w_n x_n - \theta\right) = \operatorname{sgn}\left(\sum_{i=1}^{n} w_i x_i - \theta\right) \quad (1)$$

w_i being the synaptic weight associated to x_i, θ the threshold, and n the fan-in of the TLG.

It did not take too long for a hardware implementation to be developed. In 1951 Marvin Minsky teamed with Dean Edmonds and designed the first 40-neuron "neurocomputer" *Snark* [14]. Although it was an electro-mechanical implementation built of tubes, motors, and clutches, it successfully modeled the behavior of a rat searching for food in a maze. In 1957 Frank Rosenblatt generalized the McCulloch-Pitts neuron inventing the perceptron [20]. During 1957 and 1958, Rosenblatt together with Charles Wightman and others constructed and successfully demonstrated the Mark I Perceptron. The *Mark I Perceptron* had 512 adjustable weights implemented as an 8×8×8 array of potentiometers. Because of the successful presentation of the Mark I Perceptron, the neurocomputing field became a subject of intensive research. Shortly afterwards, Bernard Widrow together with his students developed another type of neural computational element: the ADALINE (ADAptive LINear Element) [21]. They used an electrically adjustable resistor called a memistor. *This can be considered the first electrical implementation of threshold logic circuits* (TLCs), *i.e.* circuits made of TLGs. Widrow also founded the first neurocomputer hardware company: Memistor Corporation, producing neurocomputers during the early to mid 1960s. More details can be found in Nils Nilsson's book *Learning Machines* [17]. The neurocomputer industry was born.

The general belief that a neuron is a threshold element (or TL element, or TLG), which fires when some variable reaches a threshold, can be questionable as to whether such a drastic simplification can be justified. For answering that, the precise four-dimensional neuron model of Hodgkin and Huxley has been used, and the threshold model has been tested on a spike train generated by the Hodgkin-Huxley model with a stochastic input current. The result was that the threshold model correctly predicts nearly 90% of the spikes, *justifying the description of a neuron as a TLG* [11].

In the last decade the tremendous impetuous of VLSI technology has made neurocomputer design a really lively research topic. Research on hardware implementations of NNs in general, and TL in particular, has recently been very active. While there is a large body of literature on hardware implementations of NNs (see for example *Part E: Neural Network Implementations* in [2], and the many references therein), to the knowledge of the authors there are no up-to-date review papers on hardware implementations of TL since [10] and [7]. Books on TL have been written some time ago [9], [16], with only one recent chapter as an exception [1]. Particular TL implementations using either currents [4], or a few capacitive solutions [3], [18], are the exception rather then the rule, and they have covered only particular subclasses of solutions. Even more, nanoelectronics devices like those based on single electron technology (SET) or on negative resistance devices (NRDs)—*e.g.*, resonant tunnelling devices (RTDs)—have not been included [6], [15], [19]. Besides, there are many theoretical results showing that TLCs are more

Draft: September 15, 2002

V. Beiu is with the School of Electrical Engineer and Computer Science, Washington State University, Pullman, WA 99164-2752, USA (phone: +1 (509) 335-5223; fax: +1 (509) 335-3818; e-mail: vbeiu@ eecs.wsu.edu).

J.M. Quintana and M.J. Avedillo are with the Centro National de Microelectrónica (CNM), Universty of Sevilla, Edificio CICA, Avda. Reina Mercedes s/n, 41012–Sevilla, Spain (phone: +34 (95) 505-6666, fax +34 (95) 505-6686, e-mail: {josem, avedillo}@imse.cnm.es).

powerful/efficient than classical BCs. These have been another motivation to investigate various VLSI implementations. One important aspect for NNs is their adaptive behaviour, but in this in-depth review paper we shall focus only on the many different approaches that have been tried for implementing TL in silicon. Effectiveness of TL as an alternative for modern VLSI design is determined by the availability, cost and capabilities of the basic building blocks. In this sense, many interesting circuit concepts for developing standard-CMOS compatible TLGs have been explored.

As the number of different proposed solutions and fabricated chips reported in the literature is on the order of hundreds, we cannot mention all of them here. Instead, we shall try to cover important types of architectures and present only a few representative examples—although some readers might at times disagree with our choice. The paper is structured in four main Sections II–V. Each of these Sections is dedicated to a different design approach. Section II is covering a few CMOS solutions. Section III is dedicated to capacitive implementations, dealing both with switched capacitor solutions and with floating gate approaches. Section IV details many conductance/current implementations starting from pseudo-nMOS and the output-wired-inverters. It presents many solutions which have evolved from these two, as well as a large variety of differential solutions. Finally, Section V is dedicated to several other implementations including those based on SET and RTD. In most cases the various solutions discussed are sorted chronologically by order of their publication date, although in some cases the order would be somehow different by submission date. For keeping the paper's length reasonable, the very early days of TL implementations (*i.e.*, when the technologies were TTL, ECL, I^2L, and nMOS) will not be covered. However, it is worth mentioning here that there have been many different proposals. A few representative MOS ones are [8] (Fig. 1a), [12] (Fig. 1b), and [5] (Fig. 2). In conclusions we discuss and compare the different implementations, and comment on the future directions of research.

II. CMOS Solutions

Most of the early solutions have represented each distinct weighted sum (of inputs) by an analogue value (voltage or current). This implies static power dissipation, which is hardly acceptable anymore. Currently, low power solutions are at a premium, and three different low power CMOS solutions will be detailed further.

Probably *the first pure CMOS solution* is due to Hampel [22] (Fig. 3). The CMOS devices form a plurality of TLG configurations having MAJORITY logic functions with near symmetrical switching delay times. MAJORITY functions are TLF having identical (unit) weights. Any TLF can be represented as a MAJORITY function by repeating/complementing the inputs. Hence, the gate can implement arbitrary TLFs by tying together several inputs. Corresponding gate terminals of individual MOS devices within the identical nMOS and pMOS complementary stacks are commonly connected to the input signals. The fact that the nMOS and pMOS stacks are alike leads to symmetrical switching delay times. The gate has low power consumption and large noise margins. A variation of this type of gate can be found in almost any textbook on VLSI: the "mirrored adder." The only disadvantage is that larger fan-in gates are quite slow due to the large number of series transistors and the larger capacitance. Even more, if implementing arbitrary TLF, the fan-in is reduced because several inputs have to be tied together for implementing weights different from the unit weight.

A NULL Convention Logic™ (**NCL™**) gate [24], [25], [26], receives a plurality of inputs, each having an asserted state and a NULL state. The TLG switches its output to an asserted state when the number of asserted inputs exceeds a threshold number. The TLG switches its output to the NULL state only after all inputs have returned to NULL. Signal states may be implemented as distinct voltage or current levels. This approach implements *m*-of-*n* TLGs with hysteresis. This gate is a generalization of both a Muller C-element (*n*-of-*n*) and a Boolean OR (1-of-*n*) gate. NCL™ is an asynchronous delay-insensitive logic design methodology. Several implementations are possible: static (shown in Fig. 4), semi-static, and dynamic. The gate has low power and large noise margins, being reasonably fast for small fan-ins (the large number of transistors in series slows it down for larger fan-ins).

Recently, another low power solution based on a pass-transistor logic style has been detailed [23]. It offers an attractive alternative to CMOS solutions. In particular, Complementary Pass-transistor Logic (**CPL**) is a well-known low-power logic design style. A steering circuit, which produces all the TLFs for an *n*-input logic function was presented in [23]. Weights different from 1 are implemented by modifying the diagonal connection pattern in the steering circuit (instead of tying together as many inputs as given by w_i). Fig. 5 shows the steering logic circuit realizing all the six possible TL functions that can be obtained with the set of weights [1, 1, 2, 2]. A distinguishing characteristic differentiating this approach from others TLG realizations is that pass-transistor-based ones depend only on the number of variables, not on their associated weights. However, as the CPL-based design is a class of static pass-transistor logic, it inherits the problems that are specific to this class of circuits.

III. Capacitive Implementations

The concept underlying capacitive TLGs is the use of an array of capacitors to implement the weighted sum of inputs. Distinct circuits structures have been proposed which differ in the way the value of the threshold is set, and in the circuit techniques used to carry out the comparison involved in determining the output value. Capacitive threshold-logic gates

can be classified into two major groups: Capacitive Threshold Logic (**CTL**), and Neuron MOS (**νMOS)**, also known as multi input floating gate transistor (MIFG or MFMOS). Although closely related, these two original approaches were different at the beginning: static versus clocked and different mechanisms for setting the threshold value, while their current developments have become increasingly similar. Several comparisons [18], and [3] (see also [33]) draw the following conclusions:

- the operation of the νMOS is simpler than that of the CTL;
- the maximum fan-in attainable by νMOS is an order of magnitude less than that of CTL gate (the CTL gate is less limited by process variations);
- both solutions have large power consumptions (as the floating gate voltage of the primary inverter in the comparator chain causes DC current);
- the delay has a logarithmic dependence with respect to large fan-ins (fan-in ≤ 255 in [33], fan-in ≤ 64 in [18]), while for small fan-ins (fan-in ≤ 20 [3]) the behaviour of the normalized delay looks linear: 1 + 0.35*n* (*n* being the fan-in).

A. The switched capacitor

Originally introduced in 1987 the main idea was to use switched capacitors, switches and inverters, and to take advantage of the inherent saturation of the inverters to implement the neuron non-linearity without additional elements [36], [37]. This first approach required a somehow complex three-phase clock, as shown in Fig. 6.

The principle of capacitive synapse was presented also in [27], [28], with the same three-phase clock. It has quickly evolved into a simpler two-phase clock solution [33], known as the capacitive threshold logic (**CTL**) gate. Its conceptual circuit schematic is shown in Fig. 7 for an *n*-input gate. It consists of a row of capacitors C_i, $i = 1, 2, \ldots n$, with capacitances proportional to the corresponding input weight, $C_i = w_i \times C_u$, and a chain of inverters which functions as a comparator to generate the output. This TLG operates with two non-overlapping clock phases Φ_R and Φ_E. During the reset phase, Φ_R is high and the row voltage V_R is reset to the first inverter threshold voltage, while the capacitors' bottom plates are precharged to a reference voltage V_{ref}. Evaluation begins when Φ_E is at a logic 1, connecting the gate inputs to the capacitor bottom plates. As a result, the change of voltage in the capacitor top plates is given by $\Delta V_R = [\sum_{i=1}^{n} C_i (V_i - V_{ref})] / C_{tot}$ where C_{tot} is the total row capacitance including parasitics. Choosing adequate definitions for V_{ref} and C_i as functions of the input weight and threshold values, the above relationship can be expressed as $\Delta V_R = [\sum_{i=1}^{n} (w_i x_i - \theta) C_u V_{DD}] / C_{tot}$. Together with the comparison function of the chain of inverters, this gives the TL operation: $V_o = V_{DD}$ if $\sum_{i=1}^{n} w_i x_i \geq \theta$, and V_o = GND if $\sum_{i=1}^{n} w_i x_i < \theta$. Between two consecutive reset phases, a large number of input vectors can be processed.

Experimental results from different CTL gates fabricated in standard-CMOS technology [33], [29], [30], [31], have shown the proper functionality of this type of TLG and its large fan-in capability (gates with fan-in = 255 have been simulated). This later feature is due to the auto-offset cancellation technique widely used in chopper-type CMOS comparators. Originally, CTL gates required a double-poly process, but some developments (like dynamic and differential CTL [29]) use the MOS cap with a small penalty on the fan-in (fan-in ≤ 64). CTL gates have a simple regular structure, and are able to implement large fan-ins, while their main drawbacks are: large delays, large area, DC power consumption, and the threshold value programming mechanism. The reset time grows quickly with the fan-in of the gate—due to the high capacitance—and can be quite large (thousands of evaluation phases) [18]. Propagation delay is logarithmic in the number of inputs, and has a strong dependence on the unit capacitor [33]. The estimated area of the unit capacitor is equivalent to several minimum sized inverters, making the capacitor array area large. Due to the linear operation of the sense amplifier, the power consumption is high. Several developments proposed for overcoming CTL's limitations are summarized below. The fact that the threshold value is set by an analogue reference voltage complicates its integration. In addition, each CTL gate may require a different reference voltage, thus it is practically impossible to build circuits with a large number of CTL gates. This problem is solved by the improved CTL gate [29] which operates exclusively with binary input logic levels. Another solution to this problem is the Capacitor Programmable CTL gate (**CP-CTL**) [34], [35], which does not rely on the presence of additional external voltages. Fig. 8 depicts its circuit schematic. The original CTL gate is augmented with a number of capacitors. The programming of the gate is now achieved by setting V_{ref}, V_{eval1}, V_{eval2}, and V_{reset} to readily available voltage levels. Different combinations of GND, V_{DD} and V_{DD} /2 (programming method) can be used.

Finally, another variation called Balanced-CTL (**B-CTL**) [32] is shown in Fig. 9. The requirement for a highly precise reference voltage is eliminated by implementing functions with thresholds equal to 0. This is not a restriction on the class of TLFs that can be implemented, since any TLF can be converted into an equivalent TLF with threshold equal to zero by inverting certain inputs, and changing the sign of their associated weights [16]. The basic structure is formed by two banks of capacitors (Bank A and Bank B in Fig. 9). Both banks are connected to a differential amplifier that determines which bank has a larger number of inputs at logic one. That bank has a higher voltage level on its common line. This gate implements TLFs, with thresholds equal to zero, if the inputs having positive weights are connected to one bank and the inputs having negative weights are connected to the other one. One additional half capacitor unbalances the voltage level at the amplifier inputs in case both banks have an identical

number of high-level inputs. B-CTL gates operate from one clock that switches the gate between two states: reset and evaluate. B-CTL gates are reported to be faster than CIAL gates [122] (to be described in Section IV.D). Their main characteristics are high fan-in and low power consumption.

B. The neuron-MOS transistor

Neuron-MOS (**νMOS**) TLGs are based on the νMOS transistor [54]. This transistor has a buried floating polysilicon gate and a number of input polysilicon gates capacitively coupled to the floating gate (Fig. 10a). The voltage of the floating gate becomes a weighted sum of the voltages on the input gates, and controls the current in the transistor channel. The most simple νMOS-based TLG is the complementary inverter using both pMOS and nMOS νMOS devices [55], [56], [57], [58]. A schematic of this TLG is shown in Fig. 10b. The floating gate is common to both the pMOS and nMOS transistors, and the input gates correspond to the TLG inputs $x_1, x_2, \ldots, x_n$. The weights are proportional to the ratio between the corresponding input capacitance C_i (between the floating gate and each of the input gates), and the total capacitance (including the transistor channel capacitance) between the floating gate and the substrate C_{chan}. Without using the extra control inputs, the voltage in the floating gate is given by $V_F = \left(\sum_{i=1}^{n} C_i \cdot V_{x_i}\right) / C_{tot}$, where $C_{tot} = C_{chan} + \sum_{i=1}^{n} C_i$. When V_F becomes higher than the inverter threshold voltage, the output switches to logic 0.

In the case of the simple **static νMOS**, the gate's threshold is adjusted via additional threshold-setting capacitors. It is obvious that this νMOS TLG is very simple and very compact. However, there are a number of problems. Degradation in the long-term stability is anticipated due to the use of a floating gate. Sensitivity to parasitic charges in the floating gate and to process variations could limit its effective fan-in, unless adequate control is provided. In particular, ultraviolet light erasure is required for initialization/reprogramming. Static νMOS gates have DC power consumption, and different schemes have been proposed to alleviate at least some of these problems.

In the **clock-controlled νMOS** TLG [48], [49], a clock-driven switch is attached to the floating gate to initialize the floating-gate charge (reset phase). This switch short-circuits the floating gate and the inverter output, thus biasing the inverter at the most sensitive point of the inverting characteristics (see Fig. 11). This is the same auto-offset cancellation technique used for CTL gates (and in chopper type CMOS comparators). At the same time, each input capacitors is fixed to either GND or V_{DD}, such that the logical threshold of the gate is correlated with the physical threshold of the inverter. This means that, in each reset phase, the floating-gate charge is refreshed, avoiding the problems due to parasitic capacitances and long-term stability. The inverter threshold is also automatically readjusted, reducing the sensitivity to process and ambient parameters variations, and increasing the fan-in of the gates. As an example, static νMOS TLGs for MAJORITY with up to 9 inputs are possible (typically the fan-in of a static νMOS is limited by 12), while clocked νMOS can reach up to 30 inputs. The gate is not very fast: a neuron with 32 synapses of 5-bit accuracy in 0.8 μm CMOS exhibit delays in the 3–17 ns range [50]. The same concept is used in Controlled Floating-Gate Devices (**CFGDs**) [38]. These dynamic versions have relatively high static power and might require multiple phase clocks.

The static power consumption of the basic νMOS TG can be eliminated, and its speed increased, by a current comparison between a νMOS transistor and a reference device, using a positive feedback circuit. Many different configurations taking advantage of this concept have been reported [48]. One example is the configuration called **sense-amplifier νMOS** TL [49] (Fig. 12a). It applies a current-controlled latch-sense amplifier circuit to the basic νMOS TLG. Variations can be found in [60], [61], followed by several patents [59], [62], [51], [52] (Fig. 12b). They use a solution similar to the digital comparator based on the clock-coupled inverters introduced in [123] (this is a differential conductance solution to be described in Section IV.D). In [61], significant speed improvements (100 MHz to 500 MHz) and power savings for the νMOS gate from Fig. 12a over the static νMOS gate are reported. In [52] a very thorough analysis with respect to parameter variations, namely coupling capacitances of the floating gate and the sensing amplifiers of νMOS TLGs using a dynamic comparator latch for sensing, is carried out. The dominant mismatch originates from the input offset voltage variations of the sensing circuits. Measured results show that the most critical components are the comparators circuits. Improved noise margins can be traded off for increased layout areas and increased power consumption (due to increased capacitances). The conclusion is that this is a problem that will be exacerbated by future CMOS technologies, since lower supply voltages and increased device mismatch will have a diminishing effect on the threshold window, sensing margins, signal to noise ratio, and reliability. In addition, it is claimed that a careful comparison with the area consumption of a standard CMOS logic circuit is absolutely necessary, and that the use of νMOS gates is not always advisable. However, they explicitly mention that there are applications where floating gate MOS devices can be employed advantageously, like TLCs with low logic depth implemented in fault tolerant architectures requiring high functional densities (*e.g.*, data processing architectures in image sensors).

Another variation, called CMOS Capacitor Coupling Logic (**C^3L**), uses the capacitor coupling technique and a current sense amplifier [45] (Fig. 13). Although these circuits do not have an offset cancellation mechanism, fluctuation in device parameters can be compensated by the differential configuration.

Fig. 14a shows the structure of another TLG based on a charge recycling differential sense-amplifier. It is called

charge recycling threshold logic (**CRTL**) gate [42], [41]. The inputs are capacitively coupled onto the floating gate of transistor M_5, and the gate voltage of transistor M_6 sets the threshold. A CRTL gate has two operation phases controlled by a single-phase clock. When $\overline{E}$ is high the output voltages are equalized. When E is high, the outputs are disconnected and the differential circuit (transistors M_5, M_6 and M_7) draws different currents from the OUT and $\overline{OUT}$. The sense amplifier is activated and amplifies the difference of potential between OUT and $\overline{OUT}$, accelerating the transition. Thus, it evaluates if the weighted sum of the inputs is greater or less than the threshold. It is based on a charge recycling asynchronous sense differential amplifier (ASDL) [46], [47]. The symmetry of the layout is important. CRTL gates exhibit high-speeds, and are suitable for high fan-ins, while also having low power consumption. In fact, CRTL gates achieve the highest speed and 15-20% lower power consumption when compared with clocked νMOS [49], C^3L [45], and LCTL [118] (to be described in Section IV.D). CRTL gates have been tested for process variations at 45 corners, and seem to be robust. A 4-bit carry look-ahead adder using CRTL gates with fan-in up to 9, was implemented in a 0.25 μm double poly CMOS process [41]. It runs at 100-200 MHz at $V_{DD} = 2$ V, dissipating 0.5 mW @ 100 MHz.

Very recently, a novel self-timed threshold logic (**STTL**) has been proposed [43], [44]. The self-timing idea comes from asynchronous circuits, the goal being to eliminate the clock and thus reduce power consumption (a self-timed power-down mechanism applied to conductance TLGs [94], [95], [96], [98] will be detailed in Section IV.C). The gate is based on the cross-coupled nMOS transistor pair, M3 and M4 (Fig. 14b). Precharge and evaluate are specified by an enable signal: E and $\overline{E}$. Two current mirrors are used M8-M1 and M9-M2. Because the capacitances of node A and B have to be matched, the two buffering inverters have to be identically sized. The enable signals E and $\overline{E}$ are generated from the outputs and passed to the next stage, being propagate in a self-timed fashion. The solution is low-power (as being differential), and eliminates the clock at the expense of a double-rail signalling and the additional "enable generate" block. It is too early to say if the power reduction due to the elimination of the clock and its distribution is off-balanced by the "enable generate" block required by each gate. Obviously, low power solutions have to be used in designing this block. The only results reported so far are for a (7,3) counter, a fundamental building block for binary multipliers (used for reducing the partial products). In a 0.25 μm double poly CMOS, the (7,3) counter has a delay of 1.4 ns and dissipates 870 μW @ 2 V when driven by a 300 MHz enable signal.

IV. Conductance / Current Implementations

A. Early solutions

The first conductance/current based implementations of TLG's were made in the mid-1950's using resistive circuits. Later, bipolar realizations [8], [12], [66] were proposed, and MOS solutions followed [5] (see Fig. 2). In this section we shall discuss two early TL solutions in CMOS [1)], which time has proved to be enduring: the pseudo-nMOS (also know as grounded-pMOS), and the output-wired-inverters (also known as ganged CMOS).

The nMOS technology was suitable for high fan-in gates. A depletion nMOS transistor was used as a load (pull-up), making NOR gates very fast (the pull-down network has only parallel transistors). In CMOS, the solution was to use a pMOS with its gate grounded. This is the **pseudo-nMOS** (also known as grounded pMOS) solution: fast, having DC power, and using ratio rules. The reduced output voltage swing and gain makes the gate more susceptible to noise. That is why, instead of just grounding the pMOS load, its current should track the nMOS device (making the gate less sensitive to process variations), *e.g.*, by using a current mirror. This also accelerates the rise time. In time critical signal paths, pseudo-nMOS logic wisely combined with static CMOS, led to substantial speed improvements at the cost of only slightly increasing the power consumption. Furthermore, because the gate of the pull-up pMOS transistor can be turned off, pseudo-nMOS supports a power-down mechanism at no extra cost. Large fan-in gates with very fast switching times and almost without static power can be built. One last advantage of such gates is their low transistor count. The ratio rules make it possible to implement TLFs, but these TLGs exhibit all the advantages and disadvantages mentioned above. In the particular case of pseudo-nMOS TLG, the noise margins are reduced as the common output node has meaningful analog voltages. That is why these gates are limited to small fan-in values and an inverter is used both for buffering and for recovering the voltage.

The second solution is based on a plurality of inverters with their outputs hard-wired together. The first TLG implementation based on **output-wired-inverters** followed by a recovering buffer inverter (see original drawing in Fig. 15a) was detailed in 1973 by Lerch [71]: "*A threshold gate comprising a plurality of complementary-symmetry, field-effect transistor inverters, each inverter receiving at its common gate connection a different input signal and each connected at its output terminal to a common circuit output terminal* [another inverter]. *The gate may have inputs all of the same weight or, with appropriately chosen values of transistor conduction channel impedance or parallel connected inverters, may have inputs of different weight.*" It produces a non-linear voltage divider that drives a restoring inverter (or a chain of inverters) whose purpose is to quantize the non-binary (analogue) signal at the common node v_g. Fig.

[1)] These are not pure CMOS solutions like those detailed in Section II.

15b shows the circuit structure for these output-wired-inverters TLGs. Each input x_i drives a ratioed CMOS inverter with only one transistor conducting at a time (because the input is either logic "1" or "0"). Both the pMOS and the nMOS transistors are operated as resistors (conductance). That is why the voltage on v_g depends on how many pMOS and nMOS transistors are conducting, being proportional to $\sum_{i=1}^{n} w_i x_i$. The output inverter is designed to switch when this sum is greater than θ, and as an output buffer for recovering the signal. It also provides additional driving capability. The design process for these threshold gates involves sizing only two different inverters [70]. Assuming the same length for all the transistors, the widths $[W_p, W_n]_{i,b}$ for each inverter are chosen taking into account the permissible sum of the weights $\sum_{i=1}^{n} w_i$ and the θ value to be implemented. Only positive and integer weight and threshold values are allowed when using this technique. Still, this is not a limiting factor because any TLG can be implemented using only positive integer weights and threshold. Moreover, non-unit weight values $w_i = k > 1$ can be realized by simply connecting in parallel k basic inverters (one inverter corresponding to $w_i = 1$). The threshold value θ is determined by the output inverter's threshold voltage V_{th}. The v_g node is effectively isolated from external circuitry, thereby tolerating some local noise. Unfortunately, due to the sensitivity of the voltage on v_g and of the V_{th} of the output inverter to process variations, the output-wired-inverter TLGs are fan-in limited. A good study of this limitation can be found in [63], while in [69] upper and lower bounds on the channel width ratio were obtained analytically. These prove that process variations and operating conditions drastically limit the fan-in. A different approach for the determination of the W/L-ratios of the transistors uses an evolutionary algorithm [65]. Still, these TLGs are extremely fast, while exhibiting high power consumption (assumable when traded-off for speed), as well as narrow noise margins. Two very similar solutions were shortly proposed [64], [76]. Afterwards, output-wired-inverters TLGs have been rediscovered several times. In [67], a very fast CMOS NOR gate is presented which is Lech's construction [71] without the final restoring inverter. Later, Schultz et al. [75] rediscovered Lerch's original construction [71], and called it "ganged-CMOS logic" (**GCMOS**), the name under which it became well-known. The design was extended to multiple-valued logic [74]. The output-wire-inverters technique has been employed to build TLCs for non-linear filtering [68], [69], [70], Muller C-elements [73], Losq's voters with multi-threshold TLGs [73], or TLCs for D flip-flops [72].

Both pseudo-nMOS and output-wired-inverters solutions are very fast. By the time they were introduced, the DC power consumption was not such a stringent concern/limitation as it is today. Even more, the higher supply voltage made their reduced noise margins acceptable for TLGs with small fan-ins. As an example of that era, output-wired inverters implementing NOR functions with 2 and 3 inputs have been used in the MIPS R2010, the FPU of MIPS R2000 [67]. Both solutions have represented the starting points of two long series of variations/modifications, which made incremental enhancements on their two major drawbacks: the DC power consumption and their reduced noise margins. An almost exhaustive enumeration follows in the next two sections.

B. Beyond pseudo-nMOS

The DC power consumption was the major drawback of pseudo-nMOS gates when implementing BL, while noise margins became a concern only when the gates were used to implement TL. As many applications have focused on very fast BGs, pseudo-nMOS was an attractive alternative. Especially for large fan-in they are much faster than equivalent CMOS gates (slowed down by long series of transistors). That is why a lot of effort has been devoted to reducing the power consumption of large fan-in (wide) pseudo-nMOS gates (*e.g.*, implementing NOR function). Although TLGs are not always mentioned explicitly, the results obtained are immediately applicable to TLGs. The other drawback, the reduced noise margins, was left unresolved as an open question for TL research.

The main idea for reducing the DC power was to replace the pMOS load transistor (which was always 'on') with a more or less complex load circuit. Such solutions rely on: using asynchronous feedback and/or feedforward, reducing the voltage swings (unfortunately, this reduces the noise margins even more), using a clock signal (dynamic solutions), using controlled current mirrors, or even data-dependent solutions. As we shall see, combinations of several of such techniques have also been proposed.

The original pseudo-nMOS has DC current in $2^n - 1$ of the 2^n possible states (where n is the fan-in of the gate), being a data-dependent DC power consumption. For uniformly distributed random inputs, an approximation is given by the ratio $(2^n - 1)/2^n$. Even for relatively small fan-in values, this ratio is close to 1, and will be considered as "100% DC power." The data-dependent DC power consumption of the different solutions will be estimated as a percent of this "100% DC power," or the exact percent will be given when known.

One of the first solutions is due to [88] (see Fig. 16a). It is a pseudo-nMOS design with feedback: an inverter receives the output of the gate and drives the pMOS load. On average, the power is reduced to 50% (supposing that the output is also a uniformly distributed random variable). This solution is now considered as granted and included in many textbooks. A similar solution was presented later by Raza and Nazarian [84], the main differences being that the feedback loop has two inverters (instead of one), and that an additional reference voltage was used to control a second parallel load transistor. A solution for a MAJORITY gate using a current load was presented in [85]. A nonthreshold logic (**NTL**) was derived from its bipolar counterpart [92], being by that time speed-comparable to I^2L and ECL. The power-delay product is

nearly the same as that of conventional CMOS operated at high frequencies. Reduced voltage swings decrease the power consumption, but also degrade the noise margins.

An enhancement over [88] is detailed in [91] (Fig. 16b). It is "*a high speed low power dissipation, all parallel FET logic circuit*." The basic improvement is that the inverter is used both for controlling the active pull-up (load) transistor, as well as recovering the voltage and buffering the output. The output is recovered by an inverter, and latched through the pMOS load. The voltage transfer function of the inverter is deliberately skewed for improving the speed. On average the power is reduced to 50% (as in the previous solutions). A multigate serial load transistor may further reduce power consumption, unfortunately also slowing down the gate.

A precharged dynamic (clocked) load design, with feedforward for increased speed is presented in [79] (Fig. 17). It has a screening transistor (Q20) and clocking circuitry (Q17, Q18, Q19). The clocking circuitry alternately precharges nodes (30) and (34) to V_{DD}, and evaluates the voltage on node (30) to output a logic level. Two latching transistors (Q21, Q22) improve the behavior with respect to process variations and circuit instabilities. The outputs are buffered by inverters, isolating node (30). On average the power is reduced to about 25%.

Another method for reducing the DC current uses both feedback and feedforward [83] (Fig. 18a). This design is self-timed, *i.e.*, it does not use a clock. This circuit has both a strong (310) and a weak (309) pull-up pMOS. The weak pull-up device (309) is always 'on' and holds the node high if the pull-down device is in an 'off' state. However, if the pull-down device is in an 'on' state, the strong pull-up device (310) is also turned 'on', thereby providing a stable intermediate voltage on the node. A feedback path from the output (317, 316, 315 and 314) controls the state of the strong pull-up device (310). The feedback path can be made sensitive to both the temperature of the circuit and the supply voltage through a control input (320). The power reduction is difficult to estimate as depending on the sizing of the transistors, but should be better than 50% (probably as low as 25% with a carefully designed layout).

A similar solution, using both a weak and a strong pull-up, is the asynchronous high-speed large fan-in NOR gate, inspired by pseudo-NMOS and dynamic designs, and introduced in [93] (Fig. 18b). The basic idea is to use feedback from the output to control the load, and also to cut the DC current (Q9B). A regulator (VREF) is coupled to the strong pull-up transistor (Q7B) for regulating the drive current in response to temperature and power supply voltage variations (for maintaining the speed). Four different versions allow for: (*i*) high speed; (*ii*) reduced voltage swings on the inputs; (*iii*) temperature and voltage compensation; and (*iv*) limited low voltage on the output (using a feedback technique). Power reduction is difficult to estimate, but should be better than 50%, while the regulator providing the reference voltage VREF complicates the design.

A method for significantly reducing the DC power consumption of clocked pseudo-nMOS (ratioed) gates is presented in [81], [82] (Fig. 19). A sensing circuit (#1) analyzes the voltage transitions of the ratioed node, and controls the DC current flow (#2) through the entire circuit. Simulations have shown that DC power is reduced to 14%, making it one of the best solution with respect to DC power reduction (for pseudo-nMOS/ratioed circuits).

A simple improved pseudo-nMOS design for minimizing power is described in [80] (see Fig. 20a). The solution is using a clock to control a current mirror. Voltage floating of the output is also eliminated. On average power is reduced to 50%. Another more complex version of the gate uses both a clock signal (508) and a power-up signal (806). The power-up signal is a delayed version of the clock signal and, together with the feedback from the output, further diminishes the power consumption. This second solution (Fig. 20b) is complex and requires a demanding timing scheme, but could be rewarded by a DC power reduction even better than the previous one ([81], [82]).

By far the simplest solution for reducing the power consumption to 50% (on average) is presented in [89], [90] (see Fig. 21a). This is a data-dependent pseudo-nMOS gate, the pull-up transistor being controlled by one of the input variables. The idea has been used also in [87] (see Fig. 21b) for an *m*-of-*n* TLG [86].

Finally, we mention a hybrid solution presented by [77], [78] (see Fig. 22). It corresponds to the category of conductance with one pMOS transistor driven by the threshold and all the nMOS transistors driven by inputs through floating gates, *i.e.*, instead of setting the weights by the width to length ratio (W/L) of the transistors, the weights are encoded as charges on the floating gates. These charges modify the transistor threshold voltage and therefore its current. Hence, weights are programmable and can be quadratic or exponential in the voltage stored on the floating gate resulting in a large dynamic range. A 16-input programmable gate is reported. Programming is achieved through tunneling and injection of hot electrons. The solution is sensitive to noise, relatively slow, and has data dependent static power dissipation, but would allow for larger fan-ins.

C. *Beyond output-wired-inverters*

Output-wired-inverters suffer from the same disadvantages as pseudo-nMOS solutions: DC power consumption and reduced noise margins, so solutions for trying to overcome either one or the other of these disadvantages have been on the research agenda for quite some time.

A first enhancement can be seen in Fig. 23. It showed how to connect the inputs only to the nMOS transistors [103]. It reduces input capacitance by having the input signals connected only to the nMOS stack. The threshold of the function to be implemented is set by pre-wiring all the pMOS transistors to either V_{DD} or GND. The solution slightly increases the speed due to reduced capacitance, but does not improve on either power consumption or on noise margins.

A modification to the basic idea was introduced in [110], [111], where a new class of logic gates called Source Follower Pull-up Logic (**SFPL**) is described. The pull-up and pull-down structures are separated and connected through an inverter. A high fan-in gate implemented following this technique is shown in Fig. 24a. The power dissipation is similar to that of pseudo-nMOS implementations. It is mentioned that SFPL has acceptable noise margins. An enhancement over SFPL is detailed in [108] (see Fig. 24b), and used in custom comparators to speedup critical stages in a superscalar processor [109]. Other comparators specifically designed for low-power are described in [107], and compared against SFPL.

The original output-wire-inverters have two transistors per input. Using only one transistor per input was shown for particular BFs in [104], [105], [102], [106], [101]. The formal proof, and the systematic method on how to design TLGs having one transistor per input (either nMOS or pMOS), led to the β-driven threshold element (**βDTE**) [112]. The computing block is a voltage divider (the β-comparator) formed by pMOS and nMOS transistors, and can be seen in Fig. 25a. The feasibility of such an implementation follows from the fact that any threshold function can be represented in a ratio form as:

$$y = \mathrm{sgn}\left(\sum_{i=1}^{n} w_i x_i - \theta\right) = \mathrm{sgn}\left(\sum_{j\notin S} w_j x_j - \sum_{j\in S} w_j \overline{x_j}\right) = \mathrm{sgn}\left(\frac{\sum_{j\notin S} w_j x_j}{\sum_{j\in S} w_j \overline{x_j}} - 1\right) \quad (2)$$

where S is a certain subset of indices such that $\sum_{j\in S} w_j = \theta$.

The voltage on node v_β is determined by the ratio of sums of β's of pMOS and nMOS transistors. A remarkable feature of βDTE is that its implementability depends only on the threshold value θ, and does not depend on the number of inputs and their weights. The βDTE solution reduces the input capacitance and the internal node capacitance, making the gate even faster, but does not tackle any of the disadvantages: high power consumption and narrow noise margins. An improved β-comparator having higher non-linearity in the threshold zone improves on the noise margin [115], [116] (Fig. 25b). This is achieved using three additional highly stable reference voltages (a quite demanding condition). SPICE simulations for 0.8 μm CMOS have proven that the fan-in is limited to about 10. Artificial learnable neurons based on βDTEs have also been reported [113], [114], [116], [117].

Another method for enhancing the noise margins of TLGs implemented by a β-comparator is presented in [94], [95], [96], [99]. The method is data-dependent, being simpler than [115], [116]. It adds data-dependent non-linear terms to the βDTEs, practically converting the TLG into a "high order perceptron." The non-linear terms form a noise suppression logic (**NSL**), which can always be determined from the Boolean form of the TLF by subtracting the minterms implemented by the pMOS stack: $f_{NSL} = f \setminus (\prod_{j\in S} x_j)$. Fig. 26a shows the βDTE implementation of $f_4 = g_i \vee (p_i \wedge g_{i-1})$, which can be expressed as $f_4 = \mathrm{sgn}(2g_i + p_i + g_{i-1} - 1.5)$. Fig. 26b shows the implementation when the additional NSL has been added. By properly sizing the transistors the noise immunity can be improved (*i.e.*, better noise margins are traded off for larger area), and the speed can be increased (at the expense of higher DC power consumption). NSL has been tested for gates with fan-in ≤ 7. The TLG with NSL implementing f_4 in 0.5 μm CMOS has a delay of less than 80 ps at V_{DD} = 3.3 V (when driving four identical gates). A 5-layer 32-bit adder using f_4, and three other BFs $f_6 = g_i \vee (p_i \wedge g_{i-1}) \vee (p_i \wedge p_{i-1} \wedge g_{i-2})$, $h_4 = (a_i \wedge b_i) \vee [(a_i \vee b_i) \wedge (a_{i-1} \wedge b_{i-1})]$, and $g_{i:i+3} = g_{i+2:i+3} \vee (p_{i+3} \wedge p_{i+2} \wedge g_{i:i+1})$ (see [97], [99], [100]) has been implemented using TLGs with NSL in 0.18 μm CMOS. It achieves a delay of less than 300 ps dissipating 142 mW @ 2.5 GHz (running continuously).

For reducing the DC power a data-dependent self-timed power-down mechanism (**STPD**) has been recently developed [94], [96], [97], [98]. It uses either one or two additional transistors isolating the gate from V_{DD} and/or GND (Fig. 27a). Each of these transistors is driven by a control logic having as inputs the incoming data, the output of the gate and an asynchronous external signal *PW*. One of the solutions reduces the DC power to about 50% (Fig. 27b), while another solution reduces the DC power to about 25% (see [97], [98]). For the 32-bit adder mentioned above, the power can be reduced from 142 mW to 46 mW @ 2.5 GHz. A differential version of the STPD has also been developed (patent application pending), and is expected to reduce the power consumption to about 10%. It is currently tested in the design of a 4-layer 32-bit adder [100]. We anticipate that in 0.18 μm CMOS this adder will have an overall delay of less than 250 ps, while dissipating about 11 mW @ 3 GHz (running continuously). The power reduction comes both from having fewer TLGs (than the 5-layer adder) and from using the differential STPD [2].

D. Differential solutions

Many of the differential TLG implementations in the current/conductance category have in common two parallel connected sets of nMOS transistors implementing the weighting operation, and a current CMOS comparator for the threshold operation, *e.g.*, the CMOS cascode nonthreshold [92]. The main advantage over the previous solutions is their low power consumption (only dynamic power).

The operation of cross-coupled inverters with asymmetrical loads (**CIAL**) was exploited to implement digital bus comparators [123], a particular example of a TLG (see Fig. 28a). At the same time, a generic latch-type threshold gate (**LCTL**) was proposed in [118] (Fig. 28b). Its consists of a

[2] All the TLGs used in these adders include NSL.

CMOS current-controlled latch (transistors M_2/M_5 and M_7/M_{10}) providing both the output and its complement, and two input arrays ($M_{4_1} - M_{4_n}$) and ($M_{9_1} - M_{9_n}$) having an equal number of parallel transistors whose gates are the inputs of the TLG. Transistor pairs M_1/M_3 and M_6/M_8 specify the precharge or evaluate phase, and two extra transistors $M_{4_{n+1}}/M_{9_{n+1}}$ ensure correct operation for the case when the weighted sum of inputs is equal to the threshold value. Precharging occurs when the reset signal Φ_R is at logic 0. M_1 and M_6 are 'on', while M_3 and M_8 'off', and both OUT and $\overline{\text{OUT}}$ are at logic 1. Evaluation begins when Φ_R is at logic 1. M_1 and M_6 are turned 'off', while M_3 and M_8 are turned 'on', and nodes OUT and $\overline{\text{OUT}}$ begin to be discharged. In this situation, depending on the logic values at the inputs of the two transistor arrays, one of the paths will sink more current than the other. This accelerates the falling of its corresponding output voltage (either OUT or $\overline{\text{OUT}}$). When the output node of the path with the highest current value falls below the threshold voltage of either M_5 or M_{10}, it turns it off, fixing the latch situation completely. Supply current only flows during transitions and, consequently this TG does not consume static power. Input terminal connections and input transistor sizes in this TLG implementation must be established according to the threshold value θ to be implemented. When all transistors M_{4_i} and M_{9_i} ($i = 1, 2, \ldots, n$) have the same dimensions and the same voltage is applied to their gates, $I_{in} > I_{ref}$ due to $M_{4_{n+1}}$.

The speed performance of this gate is improved by the solution proposed in [122] where the nMOS banks are external to the latch (see Fig. 28c), avoiding the large long feedback chain of LCTL. It is called Cross-couple Inverters with Asymmetrical Loads Threshold Logic (**CIALTL**). Note that in spite of using the same name, circuit topologies in [122] and [123] are different. In this gate, the input transistor arrays ($M_{x_i} - M_{y_i}$, $i = 1, 2, \ldots, n$) are connected directly to the latch's output nodes, and precharging occurs when Φ_1 and Φ_2 are at logic 0, putting nodes D, OUT and $\overline{\text{OUT}}$ at logic 1. For the evaluation phase, both Φ_1 and Φ_2 are at logic 1, but Φ_2 must return to a low level before Φ_1 in order to allow the latch to switch. CIALTL needs two control signals, which have to be obtained from a general clock. Still, a great deal of power is dissipated in the internal clock front end. The circuit arrangement for realizing logic elements that can be represented by threshold value equations patented by Prange et al. [128] is a simplified version of CIAL (see Fig. 28d).

More recently, a number of TLGs have been proposed based on advanced clocked CMOS differential logic structures by implementing the pull-down network with two banks of parallel nMOS transistors, instead of using nMOS complementary logic trees. Examples are:

- Single Input Current-Sensing Differential Logic (**SCSDL**) [131], [132] after the Current Sensing Differential Logic (CSDL) [124]. Fig. 29 shows its schematic for a generic pull-down tree and the circuit structure for an *n*-input MAJORITY gate.
- Differential Current-Switch Threshold Logic (**DCSTL**) [125], [126], [127] (Fig. 30) after the Differential-Current Switch Logic (DCSL) [129], [130], a Differential Cascove Voltage Swing (DCVS) approach which restricts the voltage swing of the internal nodes for lowering the power consumption. DCSTL requires a single clock. Reported experiments from a 31-input AND show that DCSTL exhibits better power-delay product than the other two latch-based TLG implementations described above: LCTL [118] and CIALTL [122].
- Current-Mode Threshold Logic (**CMTL)** [119] also uses two banks of parallel transistor for inputs and threshold followed by sensing. Low power is achieve by limiting the voltage swing on interconnects and the internal nodes of the CMTL gates. Various clocked cross-coupled loads have led to Discharged CMTL (**DCMTL**) and Equalized CMTL (**ECMTL**).

These TLGs based on current comparisons are relatively sensitive to noise and mismatch of process parameters. Clearly, increasing the number of inputs reduces the allowed mismatch. Reliability can be improved by known layout and circuits techniques where the devices behaviour is matched (substrate voltage control, shield and isolations, layout of transistors with the same orientation, use the same size for transistors, *i.e.*, use multiple smaller transistors connected together to realize a larger device with reduced statistical parameter variations). Yield considerations limit the fan-in. As an example, yield analysis for SCSDL implemented in 0.35 μm CMOS showed that the fan-in ≤ 14 [131], [132].

All the solutions detailed above fall under one of the following two cases: either compare the sum of weights with a threshold [128], [132], [119], or compare two weighted sums [123], [118], [122]. For the second case, additional transistors are needed to differentiate when the two weighted sums are equal. A slightly better solution (patent application pending) is to implement function f with one bank, while implementing $\overline{f}$ with the other. It is well known that inverting a TL function requires only to invert the inputs and to change the threshold. The fact that f and $\overline{f}$ always have transitions in opposite directions leads to increased speed and better noise margins. By using the NSL scheme both for f and $\overline{f}$ one can do even better.

Finally a conceptually different implementation is proposed in [120], [121]. The key computational concept is to use a floating-gate device as a programmable-switched conductance. By storing one analogue value as the threshold of a floating gate device and applying a second digital value on the gate of the device, the conductance can be either zero or a pre-programmed analogue value. These conductances store the weights associated to each input. Fig. 31 depicts the

circuit schematic. Two parallel Flash-EEPROM banks implement: the weighted sum of inputs with positive weights, and the weighted sum of inputs with negative weights. The rest of the circuit, called the conductance comparator, provides for measuring conductance based on the current through the 'memory' cells. The precision to which the threshold of a floating gate can be programmed determines the bit equivalent precision of the weights.

V. Other Implementations

Many other approaches have been used for implementing TLG. As early as 1966 Jones [138] has looked into superconducting implementations. In particular research on Josephson TLs has been well published [137], [135], [136]. Other researchers have tried charge-coupled devices. They reach very low power while being very low speed also. A survey can be found in [134], [133]. Even biological TLGs have been investigated [139].

Currently, the emerging devices are: single electron devices, resonant tunnelling devices, double layer tunnelling transistors, and Schottky barrier MOSFET.

A. Single Electron Tunnelling Solutions

Single Electron Tunnelling technology (SET) has been receiving increased attention because it combines large integration and ultra low power dissipation. Operation of a SET device is based on the quantum-mechanical tunnelling phenomena. This allows controlling the current flowing through SET devices per individual electron if desired. The fundamental physical principle of SET devices is the Coulomb Blockade, which results from the quantization of the elementary charge in an isolated node of a double junction structure. To observe the blocking of the electron tunnelling through the island, the charging energy of the island has to exceed the thermal energy, which at room temperature requires ultra-fine structures. Recently, applications of SET technology to TLCs have been proposed [140], [142], [143], [144], [150], [153], [149]. In particular, several implementations of TLGs have been presented. Many of them use a capacitor array for input summation, similar to the solutions reported in Section III, but differ in the way the thresholding operation is realized.

The SET implementation of MAJORITY gates equivalent of the static νMOS TLGs is presented in [145]. The circuit consists of a capacitor array for input summation and a SET inverter for threshold operation. Fig. 32a shows a 3-input MAJORITY gate. It consist of an input capacitor array (six capacitors C) for input summation and a Tucker type [151] inverter (tunnel junctions C_{j1}, ..., C_{j4}, and capacitors C_1, C_2, and C_3) for threshold operation. The input nodes P and Q of the inverter are coupled to V_1, V_2, and V_3 through the input capacitances (C), so the potential of each input node is changed proportionally to the mean value of the inputs.

Klunder and Hoekstra [146] have proposed to use the electron box as a programmable logic circuit (NAND and NOR functions) and, although not explicitly mentioned, arbitrary TLGs. It consists of an electron box (see Fig. 32b) in which the non tunnelling capacitor has been divided into $n+1$ capacitors. For this circuit, $V_{out} = \left(V_c C_c + \sum_{k=1}^{n} V_{in,k} C_{in,k} + q_i\right) / C_{tot}$ with $C_{tot} = C_c + C_j + \sum_{k=1}^{n} C_{in,k}$, and q_i the total charge of the island (equal to ne, with n the number of electrons that left the island), assuming that the background charge and the initial charge are both zero. The circuit naturally compares V_{out} to $e/(2C_{tot})$, as an electron can tunnel through the junction if $-e/(2C_{tot}) \geq V_{out} \geq e/(2C_{tot})$. The values for V_c and for the capacitors can be selected to implement a given TL function. Correct operation at $T = 0$°K, logic input swing of 0.2 mV, and logic output swing of 1 mV, was validated through simulation with SIMON [152].

The n-input TLG proposed in [147] requires one tunnel junction and $n+2$ true capacitors (Fig. 32c). The TL function performed by this circuit is the comparison of the voltages V_j across the tunnel junction, and the critical voltage needed to enable tunnelling. Both positive and negative weights can be implemented with this structure. Correct operation was also validated using SIMON [152], assuming temperature $T = 0$°K and no background charge effects. It exhibits voltages levels of 0 V and 16 mV for the 0-logic and 1-logic respectively. A full adder is reported as having a delay of 2 ns. When such TLGs are placed in a network structure, strong feedback effects occur, which could result in erroneous behaviour. For solving this problem an active buffer is used after the TLG [148].

B. Resonant Tunneling Devices

Negative resistance devices (NRDs) have been used for quite some time for implementing TLGs [159], [155], [170]. The transistors currently in use are in fact potential barriers. If the width of such a 'potential barrier' at the base becomes smaller than the wavelength of the electron (about 10 nm), the electrons will tunnel through (tunnel effect discovered by Esaki). Such a small transistor will leak, and it will not be possible to use it as a switch. Nanaoelectronic devices in general, and resonant tunnelling devices (RTD) in particular, are designed to take advantage of exactly this effect. The simplest such device is the resonant tunnelling diode, which consists of an emitter and collector region and a double tunnel barrier structure. This contains a narrow quantum (about 5 nm), which allows electrons to travel through only at the resonant energy level. The characteristic of this device is similar to the Esaki tunnel diode, and exhibits a region of negative resistance, having a peak B and a valley C (see Fig. 33a). A resonant tunnelling transistor (RTT) is a transistor exhibiting a similar current-voltage characteristic. Here are two examples of RTT:

- the resonant-tunnelling hot-electron transistors

(RHET) is a conventional hot-electron transistor with an RTD between emitter and base;

- the quantum excited state transistor (QuESTT) is an RTD-like structure which makes direct electrical contact to the quantum well (this being the main difficulty), treating it as its base;
- to add a few more!

For implementing logical operations a bistable pair is used, having a load device and a driving device. If both devices are RTTs, the configuration corresponds to the ideal form of MOBILE: monostable-bistable transition logic element [154]. Other configurations are possible, but the major advantage comes from the fact that the NDR characteristic directly supports multiple value logic style, making TL an ideal candidate. An example is given in Fig. 33b [164]. As can be seen, the similarity with the β-comparator (Fig. 25a) is striking. Other structures are similar to pseudo-nMOS [160], [165], while more complex ones are possible (*e.g.*, a differential amplifier has been shown in [171]).

Such ideas have already been tested, and a prototyping technique based on a four transistor MOS-NDR has been reported [160], [157], [158].

VI. Conclusions

The present state of the art of TL shows a large diversity of solutions for coping with their two major drawbacks: power dissipation and reduced noise margins. Several implementation results (representing only a fraction) are reported in Table 1. Some of the solutions presented in this survey are highly advanced: differential and even asynchronous (data-dependent, self-timed, etc.). TLG have clearly benefited from developments in the more general field of differential logic structures and will continue to do so (see [177], [182]). Practically, the power dissipation should not be a problem anymore. Solutions for enhancing the noise margins have also been proposed, and could be used together with differential designs, but TLGs are more sensitive to noise than standard CMOS. Still, for a fair comparison TLGs should be evaluated against advanced BGs like, *e.g.*, domino logic, and not against standard CMOS.

It follows that fast and low-power TLGs are implementable. The major differences between one particular solution and another are the power-delay tradeoffs, conductance implementations being in general faster [181]. Slow and very low-power solutions (capacitive, differential, data-driven, asynchronous) are also possible. Lastly, the other design parameter to consider is the fan-in. Only a few solutions allow for really large fan-ins, the majority being somehow limited with respect to fan-in. Still, the claim that TLG should have a large fan-in comes from their original goal of mimicking the brain. Theoretical results [172] have shown that small fan-ins (fan-in = 6 … 9) can lead to VLSI-optimal solutions. If this were the case, almost all the solutions presented in this survey could qualify.

In addition to hardware neurons, potential applications for TLCs start from general microprocessors, DSPs, and cores where addition, multiplication and multiply-accumulate are at a premium. Others are floating point units for gaming workstations and graphics accelerators, which could clearly benefit from a boost in speed and/or reduced power. Among the dedicated applications, those which are computationally intensive are immediately coming to mind: encryption/decryption (RSA, ECC—elliptic curves cryptosystems, AES—Rijndael, …), convolution/deconvolution (FFT, DFT, DCT, etc.), compression/decompression (MPEG, etc.), and non-linear filtering. As an example, weighted order statistic (WOS) filters can be efficiently implemented with flip-flops and TLGs and they are widely used in image processing [175]. Still, with only a few exceptions TLGs have practically not been used: MIPS R2010 [67], SUN Sparc V9 [109], and a CMOS fingerprint sensor array for image processing [51], [52], [180], being the few commercial products we are aware of. Obviously this is not because TLGs are having poor performances. As the results presented in this survey have shown, advanced TLGs can compete with BGs. So why are they not used? The answer to this question has its roots in the TL design approach, namely the fact that TLGs need full custom design and that they lack high-level synthesis tools. The usefulness of TL as a design alternative, in general, will be determined not only by the availability, the cost and the high capabilities of the basic building blocks (the TLGs), but significantly more by the existence of automatic synthesis tools that could take advantage of them. Many logic synthesis algorithms exist targeting conventional BGs, but few have been developed for TLGs. The problem was addressed as early as the beginning of the 70's [16], and several PhD theses have investigated the topic [185], [176], [179], [178], [184], [173], [174]. Unfortunately, it seems like almost nothing has been done since the 70's. The two-level (depth-2) LSAT algorithm [183] inspired from techniques used in classical two-level minimization of logic circuits is one remarkable exception. As long as the effort will be put only into improving the gates, there will be few chances for TLCs, except in some dedicated applications and maybe inside a few cores.

Lastly, because quantum and reconfigurable computing will probably get central stage positions in the (near) future, TL will surely benefit from that. As RTDs are already operating at room temperature (as opposed to SET or quantum dot architectures), they appear to hold the most promise as a short to medium-term solution. The fact that TL is a perfect fit for RTDs will certainly help. This trend is proven by projects funded both by NSF (quantum and biological inspired computing, or QuBIC), and by the European Union (LOCOM and QUDOS for example).

TABLE 1. CIRCUITS FABRICATED USING THRESHOLD LOGIC GATES.

FUNCTION	CATEGORY	COMMENTS	REF.
Full adder	Static νMOS	double-poly process	[57]
Multiplier cell (plus FA)	Static νMOS	30% area of conventional worse power and speed	[61]
8-input C-Muller	Static νMOS	single nine input gate with feedback area, speed and power advantages over conventional CMOS 0.8 μm process	[53]
3-input EXOR	Clocked-νMOS	3 μm process two stages of floating gate inverters	[49]
Multiplier cell (plus FA)	Clocked-νMOS	50% are of conventional better power over 50 MHz. Better speed	[52]
Sensor array	νMOS	0.65 μm CMOS triple-metal double-poly process	[178]
3 input programable gate (NOR3, NAND3 and CARRY)	2-FGUVMOS	for low voltage applications transistor in subthreshold (its effective threshold is change by UV)	[39]
16 input programable neuron	Conductance	weighths codified as charge in a floating gate 2 μm process only qualitative characterization 1 MHz	[77]
8-input Muller C-element	Conductance	multi-output TLG 1.6 μm process	[73]
Analog rank order filter	CTL	1.2 μm process compares favourably with other reported designs	[175]
MAJORITY	CTL	ORBIT 1.2 μm double-metal double-poly process	[33]
(31,5) parallel counter	CTL	1.2 μm double-poly AMS process Input rate: 16Mvectors/s Area: 0.08mm^2	[30]
(3x3) image filter	CTL	1.2μm double-metal double-poly process	[31]

REFERENCES

I. INTRODUCTION

[1] M.J. Avedillo, J.M. Quintana, and A. Rueda, "Threshold logic," in *Wiley Encyclopedia of Electrical and Electronics Engineering*, J.G. Webster, Ed., New York: J. Wiley & Sons, 1999, vol. 22, pp. 178–190. Available: www.mrw.interscience.wiley.com/eeee/90/2290/W.2290-toc.html

[2] R. Beale, and E. Fiesler, Eds., *Handbook of Neural Computation*, New York: Inst. of Physics (IoP), 1996. Available: http://www.iop.org/Books/CIL/HNC/html/hnc_toce.htm

[3] P. Celinski, S. Al-Sarawi, and D. Abbott, "Delay analysis of neuron-MOS and capacitive threshold-logic," *Proc. ICECS'2000*, Beirut (Lebanon), 2000, vol. 2, pp. 932-935.

[4] K.W. Current, "Current-mode CMOS multiple-valued logic circuits," *IEEE J. Solid-State Circuits*, vol. SC-29, pp. 95–107, Feb. 1994.

[5] M.I. Elmasry, "Field effect current mode logic gate," U.S. Patent 4 445 051, Apr. 24, 1984.

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

[7] D. Hampel, and R.O. Winder, "Threshold logic," *IEEE Spectrum*, vol. 8, pp. 32–39, May 1971.

[8] J.D. Heightley, "Threshold logic gate," U.S. Patent 3 597 626, Aug. 3, 1971.

[9] S. Hu, *Threshold Logic*, Berkeley, CA: Univ. of California Press, 1965.

[10] S.L. Hurst, "An introduction to threshold logic: A survey of present theory and practice," *The Radio and Electronic Engineer*, pp. 339–351, Jun. 1969.

[11] 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, no. 5, pp. 1015-1045, mo. 1997

[12] H.M. Martin, "Threshold logic for integrated full adder and the like," U.S. Patent 3 609 329, Sep. 28, 1971.

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

[14] M.L. Minsky, "Neural nets and the brain-model problem," Ph.D. dissertation, Princeton Univ., Princeton, USA, 1954.

[15] C. Moffat, "A survey of nanoelectronics," TR94/2, Univ. College London, UK, 1994.

[16] S. Muroga, *Threshold Logic and Its Applications*, New York: Wiley, 1971.

[17] N.J. Nilsson, *Learning Machines*, New York: McGraw-Hill, 1965.

[18] M. Pădure, C. Dan, S. Coțofană, M. Bodea, and S. Vassiliadis, "Capacitive threshold logic: a designer perspective," *Proc. CAS'99*, Sinaia (Romania), 1999, vol. 1, pp. 81–84.

[19] W. Prost, U. Auer, F.-J. Tegude, C. Pacha, K.F. Goser, R. Duschl, K. Eberl, and O.G. Schmidt, "Tunneling diode technology," *Proc. ISMVL'2001*, Warsaw (Poland), 2001, pp. 49–58.

[20] F. Rosenblatt, "The perceptron a probabilistic model for information storage and organization," *Brain Psych. Revue*, vol. 62, pp. 386–408, mo. 1958.

[21] B. Widrow, and M.E. Hoff, Jr., "Adaptive switching circuits," *IRE WESCON Convention Record*, part 4, pp. 96–104, 1960.

II. CMOS SOLUTIONS

[22] D. Hampel, K.J. Prost, and N.R. Scheinberg, "Threshold logic using complementary MOS device," U.S. Patent 3 900 742, Aug. 19, 1975.

[23] J.M. Quintana, M.J. Avedillo, R. Jiménez, and E. Rodríguez-Villegas, "Practical low-cost CPL implementations of threshold logic functions," *Proc. GLSVLSI'01*, West Lafayette (USA), 2001, pp. 139–144.

[24] G.E. Sobelman, and K.M. Fant, "Current mode null convention threshold gate," U.S. Patent 5 640 105, Jun. 17, 1997.

[25] G.E. Sobelman, and K.M. Fant, "Null convention threshold gate," U.S. Patent 5 656 948, Aug. 12, 1997.

[26] G.E. Sobelman, and K. Fant, "CMOS circuit design of threshold gates with hysteresis", *Proc. ISCAS'1998*, Monterey (USA), 1998, pp. 61-64.

III.A. THE SWITCHED CAPACITOR

[27] U. Çilingiroğlu, "Capacitive synapses for microelectronic neural networks," *Proc. ISCAS'90*, New Orleans (USA), 1990, pp. 2982–2985.

[28] U. Çilingiroğlu, "A purely capacitive synaptic matrix for fixed-weight neural networks," *IEEE Trans. Circuits Syst. II*, vol. CAS2-38, pp. 210–217, Apr. 1991.

[29] Y. Leblebici, F.K. Gürkaynak, and D. Mlynek, "A compact 31-input programmable majority gate based on capacitive threshold logic," *Proc. ISCAS'98*, Monterey (USA), vol. 2, 1998, pp. 105–108.

[30] Y. Leblebici, H. Özdemir, A. Kepkep, and U. Çilingiroglu, "A compact high-speed (31,5) parallel counter circuit based on capacitive threshold-logic gates," *IEEE J. of Solid-State Circuits*, Vol. 31, n. 8 , Aug. 1996, pp. 1177–1183.

[31] Y. Leblebici, A. Kepkep, F.K. Gürkaynak, H. Özdemir, and U. Çilingiroglu, "A fully pipelined programmable real-time (3×3) image filter based on capacitive threshold-logic gates," *Proc. ISCAS'97*, Hong Kong, pp. 2072–2075.

[32] J. López García, J. Fernández Ramos, and A. Gago Bohórquez, "A balanced capacitive threshold logic gate," *Proc. DCIS'2000*, Montpellier (France), 2000, pp. ??–??. Available: www.el.uma.es/Ppepefer/DCIS2000.pdf

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

[34] A. Stokman, Implementation of threshold logic, M.S. thesis, Delft Univ. of Technology, Tech. Rep. TR 1-68340-28(1998)01, 1998. Available: ce-serv.et.tudelft.nl/orsres/1998/cont_STOK_2355.ps.gz or ce-serv.et.tudelft.nl/~sorin/ms/ sander/Main.ps.gz

[35] A. Stokman, S. Coţofană, and S. Vassiliadis, "A versatile threshold logic gate," *Proc. CAS'98*, Sinaia, (Romania), 1998, pp. 163–166.

[36] Y.P. Tsividis, "Switched neural networks," U.S. Patent 4 873 661, Oct. 10, 1989.

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

III.B. THE NEURO MOS TRANSISTOR

[38] K. Aoyama, "A reconfigurable logic circuit based on threshold elements with a controlled floating gate," *Proc. ISCAS 2002*, Scottsdale (USA), 2002, to be published.

[39] S. Aunet, Y. Berg, and T. Saether, ""Floating-gate low-voltage/low-power linear threshold element for neural computation," *Proc. ISCAS 2002*, Scottsdale (USA), 2002, to be published.

[40] R. Baumgartner, "Capacitive circuit architectures for realization of compact ripple-flash analog-digital converters," Ph.D. dissertation, Worcester Polytechnic Inst., USA, 2001.

[41] P. Celinski. S. Al-Sarawi, D. Abbot, and J.F. Lopez, "Low depth carry lookahead adder addition using charge recycling threshold logic," *Proc. ISCAS'2002*, Scottsdale (USA), 2002, to be published.

[42] 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, no. 17, pp. 1067–1069, Aug. 2001.

[43] 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, no. 13, pp. 633–635, Jun. 2002.

[44] P. Celinski, "Implementation of high performance digital VLSI systems using threshold logic" Ph.D. in progress, Dept. E&EE, Univ. of Adelaide, Australia (www.eleceng.adelaide.edu.au/Personal/celinski).

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

[46] B.-S. Kong, J.-S. Choi, S.-J. Lee, and K. Lee, "Charge-recycling differential logic (CRDL) for low-power application," *IEEE J. Solid-State Circuits*, vol. 31, pp. 1267–1276, Sep. 1996.

[47] B.-S. Kong, J.-D. Im, Y.-C. Kim, S.-J. Jang, and Y.-H. Jun, "Asynchronous sense differential logic," *Proc. ISSCC'99*, San Francisco (USA), 1999, pp. 284–285.

[48] K. Kotani, T. Shibata, M. Imai, and T. Ohmi, "Clocked-neuron-MOS logic circuits employing auto-threshold-adjustment," *Proc. ISSCC'95*, San Francisco (USA), 1995, pp. 320–321, 388.

[49] K. Kotani, T. Shibata, M. Imai, and T. Ohmi, "Clocked-controlled neuron-MOS logic gates," *IEEE Trans. Circuits Syst. II*, vol. CAS2-45, pp. 518–522, April 1998.

[50] R. Lashevsky, K. Takaara, and M. Souma, "Neuro MOSFET as a way to design a threshold gate with the threshold and the input weights alterable in real time," *Proc. APCCAS'98*, Chiangmai (Thailand), 1998, pp. 263–266.

[51] A. Luck, R. Thewes, and W. Weber, "Threshold logic circuit with low space requirement," U.S. Patent 5 986 464, Nov. 16, 1999.

[52] A. Luck, S. Jung, R. Brederlow, R. Thewes, K, Goser, and W. Weber, "On the design robustness of threshold logic gates using multi-input floating gate MOS transistors," *IEEE Trans. Electron Devices*, vol. ED-47, pp. 1231–1240, Jun. 2000.

[53] E. Rodríguez-Villegas, G. Huertas, M.J. Avedillo, J.M. Quintana, and A. Rueda, "A practical floating-gate Muller-C element using νMOS threshold gates," *IEEE Trans. Circuits Syst. II*, vol. CAS2-48, pp. 102–106, Jan. 2001.

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

[55] T. Shibata, and T. Ohmi, "A functional MOS transistor featuring gate-level weighted sum and threshold operations," *IEEE J. Solid-State Circuits*, vol. SC-39, pp. 1444–1455, Jun. 1992.

[56] T. Shibata, and T. Ohmi, "Real-time reconfigurable logic circuits using neuron MOS transistors," *Proc. ISSCC'93*, San Francisco (USA), Feb. 1993, pp. 238–239.

[57] T. Shibata, and T. Ohmi, "Neuron MOS binary-logic integrated circuits—Part I: Design fundamentals and soft-hardware-logic circuit implementation," *IEEE Trans. Electron Devices*, vol. ED-40, pp. 570–576, Mar. 1993.

[58] T. Shibata, and T. Ohmi, "Neuron MOS binary-logic integrated circuits—Part II: Simplifying techniques of circuit configuration and their practical applications," *IEEE Trans. Electron Devices*, vol. ED-40, pp. 974– 979, May 1993.

[59] R. Thewes, S. Prange, E. Wohlrab, and W. Weber, "Circuit for comparing two electrical quantities provided by a first neuron MOS field effect transistor and a reference source," U.S. Patent 5 990 709, Nov. 23, 1999.

[60] W. Weber, S. Prange, R. Thewes, and E. Wohlrab, "A neuron MOS transistor-based multiplier cell," in *IEDM Technical Digest*, 1995, pp. 21.5.1–21.5.4.

[61] 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. Electron Devices*, vol. ED-43, pp. 1700–1708, Oct. 1996.

[62] W. Weber, R. Thewes, and A. Luck, "Threshold logic with improved signal-to-noise ratio," U.S. Patent 6 078 190, Jun. 20, 2000.

IV.A. EARLY SOLUTIONS

[63] N.-E. Belabbès, A.J. Guterman, Y. Savaria, and M.R. Dagenais, "Ratioed voter circuit for testing and fault-tolerance in VLSI processing arrays," *IEEE Trans. Circuits Syst. I*, vol. CAS1-43, pp. 143–152, Feb. 1996 ["Ratioed voter circuit for testing and fault-tolerance," *Proc. ISCAS'92*, San Diego (USA), 1992, pp. 1125–1128].

[64] J.S. Britton, "CMOS comparator," U.S. Patent 4 031 511, Jun. 21, 1977.

[65] C. Burwick, M. Thomas, J. Dienstuhl, and K.F. Goser, "Threshold-gates in arithmetic circuits," *Proc. ICECS'01*, Valletta (Malta), 2001, pp. 909–912.

[66] R.P. Forester, "Threshold and majority gate elements and logical arrangements thereof," U.S. Patent 3 644 923, Feb. 22, 1972.

[67] M.G. Johnson, "A symmetric CMOS NOR gate for high-speed applications," *IEEE J. Solid-State Circuits*, vol. SC-23, pp. 1233–1236, Oct. 1988.

[68] Ch.L. Lee, and Ch.-W. Jen, "Bit-sliced median filter design based on majority gate," *IEE Proc.-G*, vol. 139, pp. 63–71, Feb. 1992.

[69] Ch.L. Lee, and Ch.-W. Jen, "Device programmable CMOS threshold gate for order statistic filtering," *IEEE Winter Workshop on Nonlinear Digital Signal Processing*, Tampere (Finland), 1993, pp. 5.2-1.1–5.2-1.5.

[70] Ch.L. Lee, and Ch.-W. Jen, "CMOS threshold gates and networks for order statistics filtering," *IEEE Trans. Circuits Syst. I*, vol. CAS1-41, pp. 453–456, Jun. 1994.

[71] J.B. Lerch, "Threshold gate circuits employing field-effect transistors," U.S. Patent 3 715 603, Feb. 6, 1973.

[72] J.M. Quintana, M.J. Avedillo, A. Rueda, and S. Baena, "Hazard-free edge-triggered D flipflop based on threshold gates, " *Electron. Lett.*, vol. 30, pp. 1390–1391, (17), 1994.

[73] J.M. Quintana, M.J. Avedillo, A. Rueda, and C. Baena, "Practical low-cost CMOS realization of complex logic functions, " *Proc. ECCTD'95*, Istanbul (Turkey), 1995, pp. 51–54.

[74] K.J. Schultz, and K.C. Smith, "A CMOS binary adder using quaternary ganged-logic internal node," *Proc. ISMVL'89*, Guangzhou (China), 1989, pp. 356–359.

[75] K.J. Schultz, R.J. Francis, and K.C. Smith, "Ganged CMOS: trading standby power for speed," *IEEE J. Solid-State Circuits*, vol. SC-25, pp. 870–873, June 1990.
[76] R. Zuleeg, and J.K. Notthoff, "Symmetrical input NOR/NAND gate circuit," U.S. Patent 4 038 563, Jul. 26, 1977.

IV.B. BEYOND PSEUDO-NMOS

[77] V. Bohossian, P. Hasler, and J. Bruck, "Programmable neural logic," *IEEE Trans. Component, Packaging, and Manufacturing Techniology Part B*, vol. 21, pp 346–351, Nov. 1998 [Preliminary version in *Proc. Int. Conf. Innovative Systems in Silicon*, Austin (USA), 1997, pp. 13–21].
[78] V. Bohossian, "Neural logic: theory and implementation," Ph.D. dissertation, Caltech, USA, 1998. Available: www.paradise.caltech.edu/papers/vincent_thesis.ps
[79] J.T. Bridges, J.E. Maguire, and P.C. Rossbach, "Decoder/comparator and method of operation," U.S. Patent 5 291 076, Mar. 1, 1994.
[80] E. Fang, "Low-power, compact digital logic topology that facilitates large fan-in and high-speed circuit performance," U.S. Patent 5 670 898, Sep. 23, 1997.
[81] P.D. Kartschoke, and N.J. Rohrer, "Low power pre-discharged ratio logic," U.S. Patent 5 572 150, Nov. 5, 1996.
[82] P.D. Kartschoke, and N.J. Rohrer, "Techniques for reduced power and increased speed in dynamic- and ratio-logic circuits," *Proc. MWSCAS'96*, Ames (USA), 1996, pp. 175–178.
[83] B.Y. Nguyen, "Low power consumption and high speed NOR gate integrated circuit," U.S. Patent 5 408 145, Apr. 18, 1995.
[84] S.B Raza, and H. Nazarian, "High-speed ratio logic structures with static and dynamic pull-ups and/or pull-downs using feedback," U.S. Patent 5 654 652, Sep. 27, 1995.
[85] W.C. Seelbach, and B.K. Hansen, "Majority logic gate," U.S. Patent 4 423 339, Dec. 27, 1983.
[86] G.E. Sobelman, and K.M. Fant, "Null convention threshold gate," U.S. Patent 5 656 948, Aug. 12, 1997.
[87] G.E. Sobelman, and J.J. Hinze, "Semi-dynamic and dynamic threshold gates with modified pull-up structures," U.S. Patent 6 262 593, Jul. 17, 2001.
[88] T. Takemoto, "MOS type semiconductor IC device," U.S. Patent 3 911 289, Oct. 7, 1975.
[89] R.N. de Tavares, M. Berkelaar, and J. Jees, "Low Power PLA," *Proc. IEEE/ACM IWLS'98*, Lake Tahoe (USA), 1998, pp. 436–439.
[90] R.N. de Tavares, and M. Berkelaar, "Reducing power consumption for PLAs," *Proc. ProRISC/IEEE CSSP'98*, Mierlo (The Netherlands), 1998, pp. 535–539.
[91] A.W. Vinal, "High speed complementary filed effect transistor logic circuits," U.S. Patent 5 001 367, Mar. 19, 1991.
[92] J.-S. Wang, C.-Y. Wu, and M.-K. Tsai, "CMOS nonthreshold logic (NTL) and cascode nonthreshold logic (CNTL) for high-speed applications," *IEEE J. Solid-State Circuits*, vol. SC-24, pp. 779–786, Jun. 1989.
[93] J.T. Wong, F. Fontana, and H. Law, "High speed NOR gate with small output voltage swings," U.S. Patent 5 432 463, Jul. 11, 1995.

IV.C. BEYOND OUTPUT-WIRED INVERTERS

[94] V. Beiu, "Ultra-fast noise immune CMOS threshold gates," *Proc. MWSCAS'2000*, Lansing (USA), Aug. 2000, vol. 3, pp. 1310–1313.
[95] V. Beiu, "High-speed noise robust threshold gates" (Best Paper Award), *Proc. CAS'2000*, Sinaia (Romania), Oct. 2000, vol. 1, pp. 79–82.
[96] V. Beiu, "On higher order noise immune perceptrons," *Proc. IJCNN'01*, Washington (USA), 2001, pp. 246–251.
[97] V. Beiu, "Adder and multiplier circuits employing logic gates having discrete, weighted inputs and methods of performing combinatorial operations therewith," U.S. Patent 6 205 458, Mar. 20, 2001.
[98] V. Beiu, "Logic gate having reduced power dissipation and method of operation thereof," U.S. Patent 6 259 275, Jul. 10, 2001.
[99] V. Beiu, "Noise tolerant conductance-based logic gate and methods of operation and manufacturing thereof," U.S. Patent 6 430 585, Aug. 6, 2002.
[100] V. Beiu, "Adder having reduced number of internal layers and method of operation thereof," U.S. Patent 6 438 572, Aug. 20, 2002.
[101] H. Chung and K. Lee, "Programmable multilayer neural network," U.S. Patent 5 448 682, Sep. 5, 1995.
[102] H. Chung and S. Pack, "Floating point adder circuit using neural network," U.S. Patent 5 086 405, Feb. 4, 1992.
[103] S. Goodwin-Johansson, "Circuit to perform variable threshold logic," U.S. Patent 4 896 059, Jan. 23, 1990.
[104] H. Jeong, "Neural network implementation of a binary adder," U.S. Patent 5 016 211, May 14, 1991.
[105] H. Jeong, "Adaptive associative memory comprising synapses of CMOS transistors," U.S. Patent 5 034 918, Jul. 23, 1991.
[106] H. Jeong, "Digital multiplier employing CMOS transistors," U.S. Patent 5 095 457, Mar. 10, 1992.
[107] N. Ko, A. Pua, A. Hill, and P. Srivastava, "Hybrid dual-threshold techniques for high-performance processors with low-power features," *Proc. ISLPED'97*, Monterey (USA), 1997, pp. 307–311.
[108] L.A. Lev, "Fast static cascode logic gate," U.S. Patent 5 438 283, Aug. 1, 1995.
[109] L.A. Lev, A. Charnas, M. Tremblay, A.R. Dalal, B.A. Frederick, C.R. Srivatsa, D. Greenhill, D.L. Wendell, D.D. Pham, E. Anderson, H.K. Hingarh, I. Razzack, J.M. Kaku, K. Shin, M.E. Levitt, M. Allen, P.A. Ferolito, R.I. Bartolotti, R.K. Yu, R.J. Melanson, S.I. Shah, S. Nguyen, S.S. Mitra, V. Reddy, V. Ganesan, and W.J. de Lange, "A 64-b microprocessor with multimedia support", *IEEE J. Solid-State Circuits*, vol. SC-30, pp. 1227–1238, Nov. 1995.
[110] T.D. Simon, and T.F. Knight, "A fast static logic gate," MIT Tr. Note #57. Available: www.cs.caltech.edu/research/ic/transit/tn57/tn57.html
[111] T.D. Simon, "A fast static CMOS NOR gate," *Proc. of the 1992 Brown/MIT Conf. on Advanced Research in VLSI and Parallel Systems* (T. Knight, and J. Savage (Eds.), Cambridge, Mass.: MIT Press, 1992, pp. 180–192.
[112] V.I. Varshavsky, "β-driven threshold elements," in *Proc. GLSVLSI'98*, Lafayette (USA), 1998, pp. 52–58.
[113] V.I. Varshavsky, " CMOS artificial neuron on the base of β-driven threshold element," *Proc. IEEE SMC'98*, San Diego (USA), vol. 2, 1998, pp. 1857–1861.
[114] V.I. Varshavsky, "Simple CMOS learnable threshold element," *Int. ICSC/IFAC NC'98*, Vienna (Austria), 1998. pp. ???–???.
[115] V.I. Varshavsky, and V. Marakhovsky, "β-CMOS artificial neuron and implementability limits," *Proc. IWANN'99*, Alicante (Spain), vol. 2, 1999, pp. 117–128.
[116] V.I. Varshavsky, and V. Marakhovsky, "Implementability restrictions of the β-CMOS artificial neuron," *Proc. ICECS'99*, Pafos (Cyprus), vol. 1, 1999, pp. 401–405.
[117] V.I. Varshavsky, and V. Marakhovsky, "β-CMOS implementation of artificial neurons," *Proc. AeroSense'99*, Orlando (USA), vol. SPIE-3722, 1999, pp. 210–221.

IV.D. DIFFERENTIAL SOLUTIONS

[118] M.J. Avellido, J.M. Quintana, A. Rueda, and E. Jiménez, "A low-power CMOS threshold-gate," *Electron. Lett.*, vol. 31, pp. 2157–2159, Dec. 1995.
[119] S. Bobba, and I.N. Hajj, "Current-mode threshold logic gates," *Proc. ICCD'2000*, Austin (USA), 2000, pp. 235–240.
[120] V. Fabbrizio, F. Raynal, X. Mariaud, A. Kramer, and G. Colli, "Low-power, low-voltage conductance mode CMOS analog neuron," *Proc. of the MicroNeuro'96*, pp. 111-115.
[121] V. Fabbrizio, G. Colli, and A. Kramer, "Low-voltage, very-low-power conductance mode neuron," U.S. Patent 6 032 140, Feb. 29, 2000.
[122] J. Fernández Ramos, J.A. Hidalgo López, M.J. Martín, J.C. Tejero, and A. Gago Bohórquez, "A threshold logic gate based on clocked coupled inverters," *Intl. J. Electronics*, vol. 84, 371–382, Apr. 1998.
[123] J.A. Hidalgo López, J.G. Tejero, J. Fernández Ramos, and A. Gago Bohórquez, "New types of digital comparators," *Proc. ISCAS'95*, Seattle (USA), vol. 1, 1995, pp. 29–32.
[124] J. Park, J. Lee, and W. Kim, "Current sensing differential logic (CSDL)," *Proc. ISCAS'98*, Monterey (USA), 1998, pp. 129–132.
[125] M. Pădure, S. Coțofană, C. Dan, M. Bodea, and S. Vassiliadis, "A new latch-based threshold logic family," *Proc. CAS'01*, Sinaia (Romania), 2001, vol. 2, pp. 531–534.

[126] M. Pădure, S. Coţofană, S. Vassiliadis, C. Dan, M. Bodea, "A low-power threshold logic family," *Proc. ICECS 2002*, Dubrovnik (Croatia), 2002, to be published. Available: ce.et.tudelft.nl/~marius/icecs2002.pdf
[127] M. Pădure, "Threshold logic arithmetic units," PhD in progress, EE Dept. Technische Universiteit Delft, The Netherlands (ce.et.tudelft.nl/~marius/).
[128] 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.
[129] D. Somasekhar and K. Roy, "Differential current switch logic: A low DCVS logic family," *IEEE J. of Solid State*, Vol. 31, 7, pp. 981-991, 1996.
[130] D. Somasekhar, K. Roy, J. Sugisawa, "Differential current switch logic gate." U.S. Patent 6 014 041, Jan. 11, 2000.
[131] R. Strandberg, and J. Yuan, "Single input current-sensing Differential Logic (SCSDL)," *Proc. NORCHIP'99*, Oslo (Norway), 1999, pp. 328–333.
[132] R. Strandberg, and J. Yuan, "Single input current-sensing differential logic (SCSDL)," *Proc. ISCAS'2000*, Geneva (Switzerland), 2000, vol. 1, pp. 764–767.

V. Other Implementations

[133] M.H. Abd-El-Barr, "On the design of multi-valued CCD logic circuits," PhD dissertation, Univ. of Toronto, Toronto, Canada, 1986.
[134] M.H. Abd-El-Barr, and Z.G. Vranesic, "Charge-coupled device implementation of multivalued logic systems," *IEE Proc. E*, vol. 136, no. 4, pp. 306–315, Jul. 1989.
[135] N. Fujimaki, S. Kotani, T. Imamura, and S. Hasuo, "Josephson modified variable threshold logic gates for use in ultra high speed LSI," *IEEE Trans. Electron Devices*, vol. ED-36, pp. 433–446, Feb. 1989.
[136] Y. Harada, and E. Goto, "Artificial neural network circuits with Josephson devices," *IEEE Trans. Magn.*, vol MAG-27, pp. 2863–2866, Mar. 1991.
[137] Y. Hatano, Y. Harada, K. Yamashita, Y. Tarutani, and U. Kawabe, "A 4-bit×4-bit multiplier and 3-bit counter in Josephson threshold logic," *IEEE J. Solid-State Circuits*, vol. SC-22, pp. 606–612, Aug. 1987.
[138] L.J. Jones II, "Superconductive threshold logic circuitry," Ph.D. dissertation, Duke Univ., Durham, USA, 1966.
[139] M. Okamoto, "Development of biochemical threshold-logic device capable of storing short-memory," *Proc. IEEE Engineering in Medicine and Biology Society* (EMBS'90), Philadelphia (USA), 1990, pp. 1770–1771.

V.A. Single Electron Technology (SET)

[140] S. Bandyopadhyay, V.P. Roychowdhury, and X. Wang, "Computing with quantum dots: novel architectures for nanoelectronics," *Phys. Low-Dim. Struc.*, 8/9, pp. 29–82, mo. 1995.
[141] M.J. Goossens, "Analog neural networks in single-electron tunneling technology," Ph.D. dissertation, Delft Institute of Microelectronics and Submicron Technology (DIMES), The Netherlands, 1998.
[142] M.J. Goossend, C.J.M. Verhoeven, and A.H.M. van Roermund, "Concepts for ultra low-power and very-high-density single-electron neural networks," *Proc. NOLTA'95*, Las Vegas (USA), pp. 679–682, 1995.
[143] M.J. Goossend, C.J.M. Verhoeven, and A.H.M. van Roermund, "Single electron tunneling technology for neural networks," *Proc. MicroNeuro'96*, Lausanne (Switzerland), pp. 125–130, 1996.
[144] M.J. Goossens, J.H. Ritskes, C.J.M. Verhoeven, and A.H.M. van-Roermund, "Neural networks with periodic single-electron tunnelling transistors," *Proc. ECCTD'97*, Budapest (Hungary), 1997, pp. 936–941.
[145] H. Iwamura, M. Akazawa, and Y. Amemiya, "Single-electron majoriy logic circuits," *IEICE Trans. Electron*, vol. E-81C, no. 1, pp. 42–48, Jan. 1998.
[146] R.H. Klunder, and J. Hoekstra, "Programmable logic using a SET electron box," *Proc. ICECS*, Valletta (Malta), 2001, vol. 1, pp. 185–188.
[147] C. Lageweg, S. Coţofană, and S. Vassiliadis, "A linear threshold gate implemented in single electron technology," *Proc. WVLSI'01*, Orlando (Florida), pp. 93–98, 2001.
[148] C. Lageweg, S. Coţofană, and S. Vassiliadis, "A full adder implementation using SET based linear threshold gates", *Proc. ICECS 2002*, Dubrovnik (Croatia), 2002, to be published.
[149] A.H.M. van Roermund, and J. Hoekstra, "From nanotechnology to nanoelectronic systems, for sets to neural nets," *Proc. ISCAS'00*, Geneva (Switzerland), 2000, vol. 1, pp. 8–11.
[150] E. Rouw, R. van de Haar, A.H.M. van Roermund, R. Klunder, and J. Hoekstra, "Neural nets using SET technology," *Proc. ProRISC'99*, Mierlo (The Netherlands), 1999, pp. 685–689.
[151] J.R. Tucker, "Complementary digital logic based on the Coulomb blockade", *J. Appl. Phys*., vol. 72, no. 9, pp. 4399–4413, mo. 1992.
[152] C. Wasshuber, H. Kosina, and S. Selberherr, "SIMON: a simulator for single-electron device and circuit", *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 16, No. 9, pp. 937–944, Sep. 1997.
[153] T. Yamada, and Y. Amemiya, "A Multiple-valued Hopfield Network device using Single-Electron Circuits," *IEICE Trans. Electron*, vol. E-82C, no. 9, pp. 1615–1622, Sep. 1999.

V.B. Resonant Tunnelling Devices

[154] T. Akeyoshi, K. Maezawa, and T. Mizutani, "Weighted sum threshold logic operation of MOBILE's (monostable-bistable transition logic element) using resonant-tunnelling transistors," *IEEE Electron Device Lett.*, vol. ED-14, pp. 475–477, Oct. 1993.
[155] J.J. Amodei, "Electrical circuits using negative resistance diode-transistor combination," US Patent 3 315 100, Apr. 18, 1967.
[156] M. Bhattacharya, and P. Mazumder, "Noise margins of threshold gates containing resonant tunneling diodes," *IEEE Trans. Circuit Syst. II*, vol. CAS2-47, pp. 1080–1085, mo. 2000 [Preliminary version in *Proc. ISCAS'98*, Monterey (USA), 1998, vol. 2, pp. 61–64].
[157] A.F. González, M. Bhattacharya, S. Kulkarni, and P. Mazumder, "CMOS implementation of a multiple-valued logic signed-digit full adder based on negative-differential-resistance devices," *IEEE J. Solid-State Circuits*, vol. SC-36, pp. 924–932, Jun. 2001.
[158] A.F. González, "Analysis and design of ultrafast digital circuits based on resonant-tunnelling quantum electronic devices," Ph.D. dissertation, Univ. of Michigan, Ann Arbor, USA (www.eecs.umich.edu/~algonz/).
[159] T.A. Jeeves, "High-speed tunnel diode adder," US Patent 3 280 316, Oct. 18, 1966.
[160] P.M. Kelly, C.J. Thompson, T.M. McGinnity, and L.P. Maguire, "Investigation of a programmable threshold logic gate array," *Proc. ICECS'2002*, Dubrovnik (Croatia), 2002, to be published.
[161] P. Mazumder, S. Kulkarni, M. Bhattacharya, J.P. Sun, and G.I. Haddad, "Digital circuit applications of resonant tunnelling devices," *Proc. IEEE*, vol. 86, pp. 664–686, Apr. 1998.
[162] A.S. Nazimudeen, J. Deng, and M.S. Shur, "Application of heterodimensional transistors for weighted sum threshold logic operations," *Proc. IEEE/Cornell Conf. Advanced Concepts on High Perf. Devices*, Ithaca (USA), 2000, pp. 101–109.
[163] C. Pacha, and K. Goser, "Design of arithmetic circuits using resonant tunneling diodes and threshold logic," *Proc. 2nd Workshop on Innovative Circuits and Systems for Nano Electronics*, Delft (The Netherlands), 1997, pp. 83–93.
[164] C. Pacha, U. Auer, C. Burwick, P. Glösekötter, A. Brennemann, W. Prost, F.-J. Tegude, and K.F. Goser, "Threshold logic circuit design for parallel adders using resonant tunneling devices," *IEEE Trans. VLSI Syst.*, vol. VLSI-8, pp. 558–572, Oct. 2000.
[165] C. Pacha, P. Glösekötter, K. Goser, W. Prost, U. Auer, and F.-J. Tegude, "Resonant tunneling transistors for threshold circuit applications," *Proc. GLSVLSI'99*, Ann Arbor (USA), Mar. 1999, pp. 344–345.
[166] C. Pacha, K. Goser, A. Brennemann, and W. Prost, "Resonant tunneling based threshold logic circuits," *Proc MEL-ARI Workshop on Nano-Scale Integrates Circuits*, Lille (France), 1998, pp. ???–???. Available: www-be.e-technik.uni-dortmund.de/~pacha/locom/locom2c.html

[167] C. Pacha, K. Goser, A. Brennemann, and W. Prost, "A threshold logic full adder based on resonant tunnelling transistors," *Proc. ESSCIRC'98*, The Hague (The Netherlands), Sept. 1998, pp. 428–431.

[168] W. Prost, U. Auer, J. Degenhardt, A. Brennemann, C. Pacha, K.F. Goser, and F.-J. Tegude, "Technology of a depth-2 full-adder circuit using the InP RTD/HFET MOBILE," *Proc. IPRM'2001*, Nara (Japan), 2001, pp. 228–231.

[169] W. Prost, S.O. Kim, P. Glösekötter, C. Pacha, H. van Husen, T. Reimann, K.F. Goser, and F.-J. Tegude, "Experimental threshold logic implementations based on resonant tunneling diodes," *Proc. ICECS'2002*, Dubrovnik (Croatia), 2002, to be published.

[170] L.K. Russel, T.T. Dao, and R.S. Muller, "Threshold switching integrated circuit and method of forming the same," US Patent 3 943 554, Mar. 9, 1976.

[171] T. Waho, K. Hattori, and K. Honda, "Novel resonant-tunneling multiple-threshold logic circuit based on switching sequence detection," *Proc. ISMVL'2000*, Portland (USA), 2000. pp. 317–322.

VI. Conclusions

[172] V. Beiu, and H.E. Makaruk, "Deeper sparser nets can be optimal," *Neural Proc. Lett.*, vol. 8, pp. 201–210, Dec. 1998.

[173] B.D. Carroll, "Two-level realization of switching functions with a minimum number of threshold gates," Ph.D. dissertation, University of Texas at Austin, USA, 1969.

[174] G. T.-H. Cheng, "A study of the synthesis and the decomposition of threshold gate networks," Ph.D. dissertation, Columbia Univ., USA, 1971.

[175] T. Dake and U. Çilingiroğlu, "Rank-order filter using multiple-winners-take-all," *Proc. MWSCAS'2000*, Lansing (USA), 2000, pp. 482–485.

[176] M.L. Dertouzos, "Threshold logic: A synthesis approach," Ph.D. dissertation, EE Dept., Massachusetts Institute of Technology, USA, 1964.

[177] E.S. Gayles, "Data enabled logic circuits," US Patent 6 078 196, Jun. 20, 2000.

[178] R. Gonzales, "Synthesis problems in linear threshold logic," Ph.D. dissertation, Univ. of Michigan, USA, 1966.

[179] J.E. Hopcroft, "Synthesis of threshold logic networks," Ph.D. dissertation, Stanford Univ., USA, 1964.

[180] 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 Journal of Solid-State Circuits*, vol. 34, no. 7 , pp. 978–984, Jul. 1999.

[181] A.H. Kramer, "Array-based analog computation; principles, advantages and limitations," *Proc. MicroeNeuro'96*, Lausanne (Switzerland), 1996, pp. 68–79.

[182] 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 Circuits*, vol. SC-36, pp. 1276–1280, Aug. 2001.

[183] A.L. Oliveira, and A. Sangiovanni-Vincentelli, "LSAT – An algorithm for the synthesis of two level threshold gate networks," *Proc ICCAD'91*, Santa Clara (USA), 1991, pp. 130–133.

[184] R.T. Sha, "Threshold logic: A simplified synthesis by algebraic approach," Ph.D. dissertation, Univ. of Pittsburgh, USA, 1968.

[185] R.O. Winder, "Threshold logic," Ph.D. dissertation, Princeton Univ., Princeton, USA, 1962.

Fig. 1. MOS implementation from [8], and from [12].

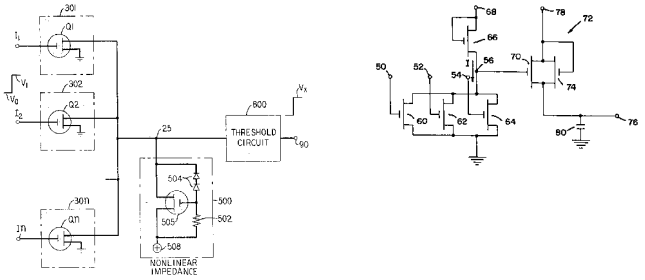


Fig. 2. nMOS solution from [5].

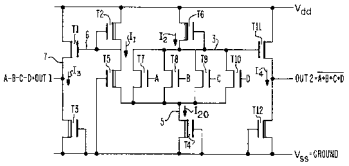


Fig. 3. CMOS solution for MAJORITY functions from [22].

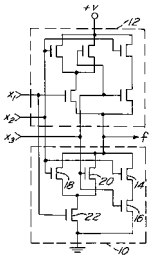


Fig. 4. "NULL convention threshold logic" from [24].

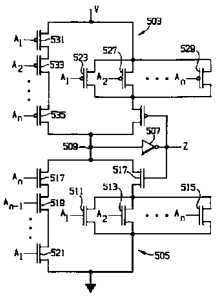


Fig. 5. Steering logic network implementing $[1, 1, 2, 2; k]$ for $k = 1, \ldots, 6$, from [23].

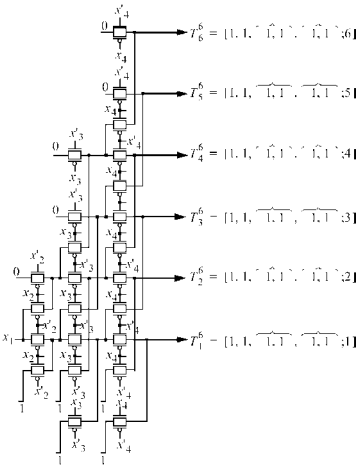


Fig. 6. Switched capacitor from [36], [37].

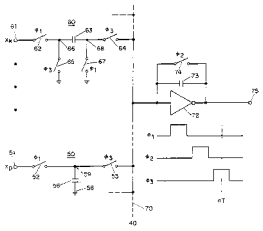


Fig. 7. Capacitive threshold logic (**CTL**) from [33].

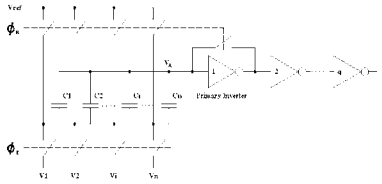


Fig. 8. Capacitor-programmable capacitive threshold logic (**CP-CTL**) from [34], [35].

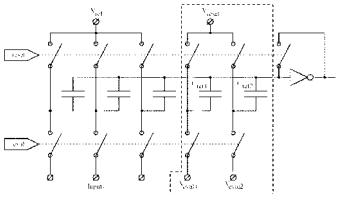


Fig. 9. Balanced capacitive threshold logic (**B-CTL**) from [32].

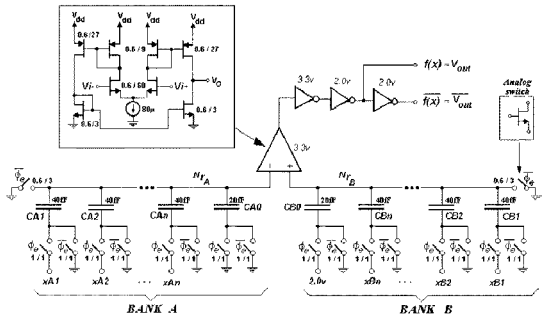


Fig. 10. Original neuron MOSFET (neuMOS, or **vMOS**) from [54], and complementary vMOS (**C-vMOS**) inverter (static gate) [55], [56], [57], [58].

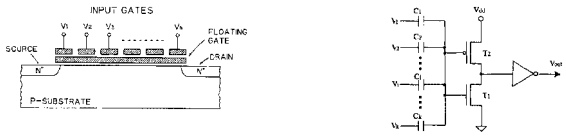


Fig. 11. Clock-controlled neuron-MOS (with reference voltage) from [48], and without reference voltage from [50].

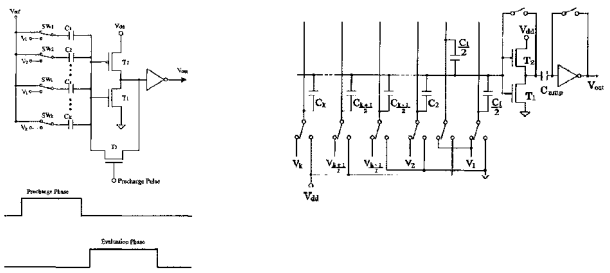


Fig. 12. Dynamic latched-sense-amplifier (comparator) neuron MOS from [49], and from [51], [52], [59], [62].

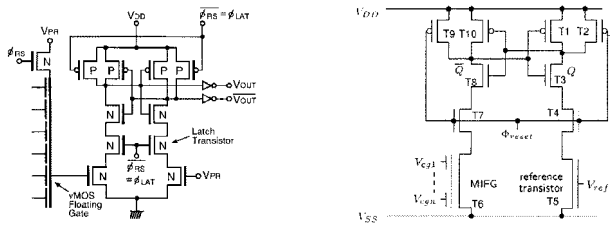


Fig. 13. CMOS capacitor coupling logic (**C^{3}L**) from [45].

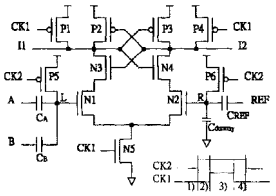


Fig. 14. Charge recycling threshold logic (**CRTL**) from [42], and self-timed threshold logic (**STTL**), another asynchronous sense amplifier differential logic with self-timed enable signalling from [43].

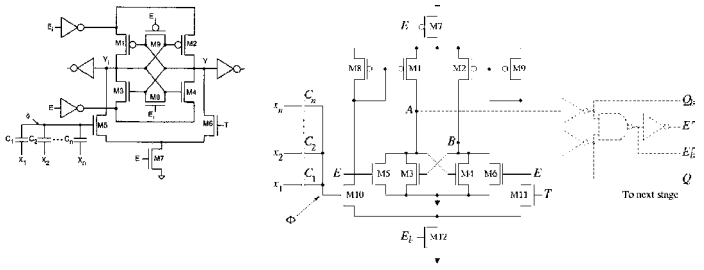


Fig. 15. The output wired inverters discovered by Lerch [71].

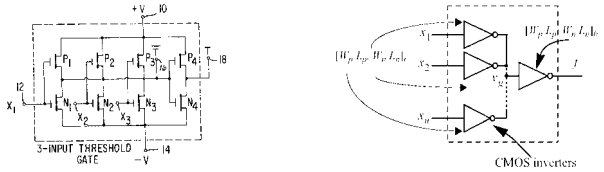


Fig. 16. Dynamic load for lowering the power from [88], and variation from [91].

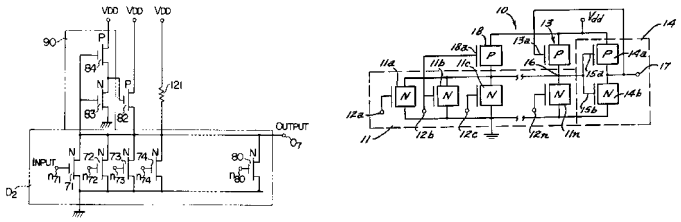


Fig. 17. Clocked (dynamic) solution from [79].

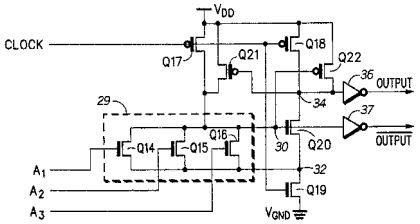


Fig. 18. Self-timed feedback solution for low power consumption from [83], and self-timed feedback solution limiting the voltage from [93].

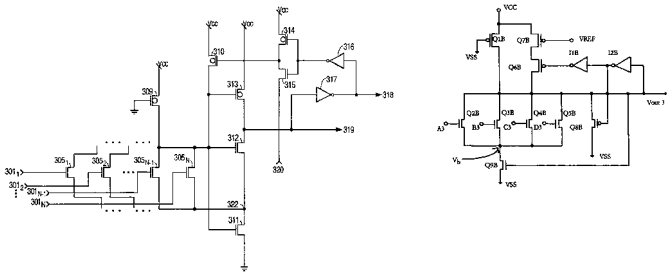


Fig. 19. Pre-discharged ratio logic from [81], [82].

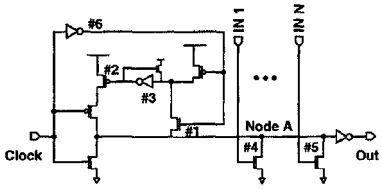


Fig. 20. Clocked (dynamic) solutions from [80].

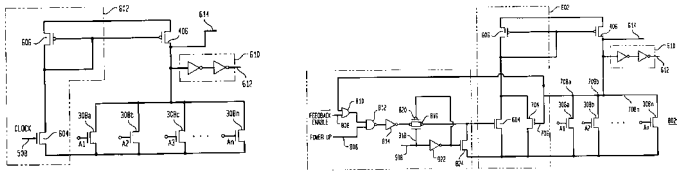


Fig. 21. Modified data dependent pseudo-NMOS gates from [89], [90], and data dependent *m*-of-*n* threshold gate from [87].

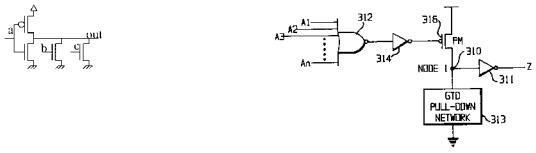


Fig. 22. Solution using floating gates for the inputs from [77], [78].

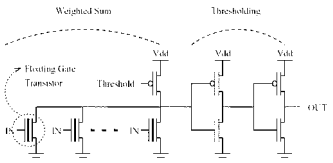


Fig. 23. CMOS solution from [103].

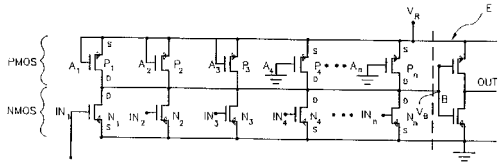


Fig. 24. Source follower pull-up logic (**SFPL**) from [110], [111], and an enhanced version from [108], [109].

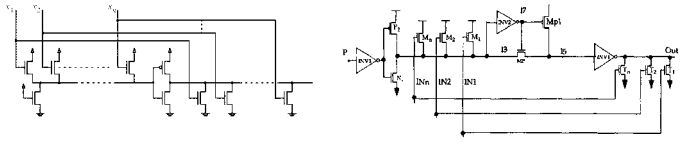


Fig. 25. Beta-driven threshold element (**βDTE**) from [112] (add and mention patents: 2001, 2002), and modified beta-driven threshold element (**βDTE**) from [115], [116].

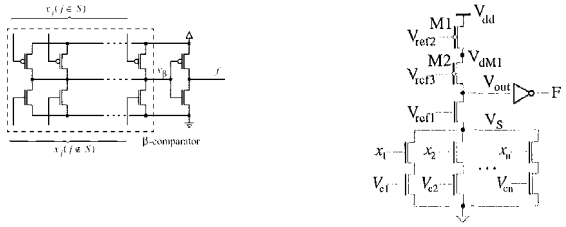


Fig. 26. Data dependent noise suppression logic (**NSL**) from [94], [95], [96], [99].

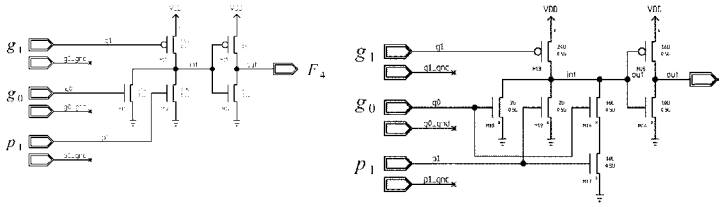


Fig. 27. Data dependent self-timed power-down (**STPD**) mechanism from [94], [96], [98].

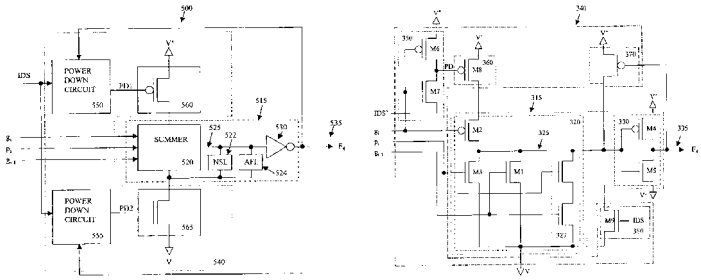


Fig. 28. Digital comparators based on clocked cross-coupled inverters with asymmetrical load (**CIAL**) from [123]; latch type low power threshold logic (**LCTL**) from [118]; (**CIAL-TL**) from [122]; and the differential implementation from [128].

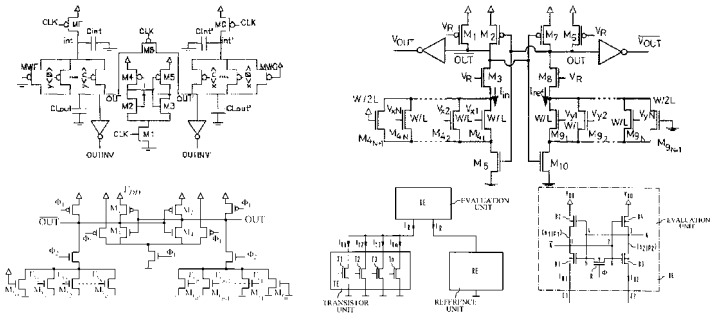


Fig. 29. Single input current-sensing differential logic (**SCSDL**) form [131], [132].

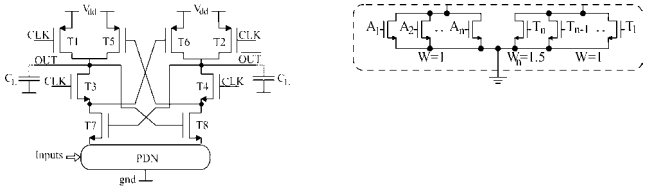


Fig. 30. Differential current-switch threshold logic (**DCSTL**) from [125], [126].

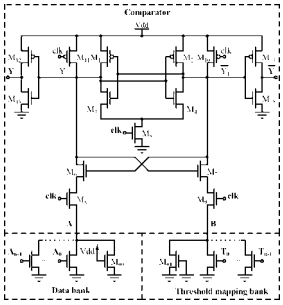


Fig. 31. Conductance sensing using floating gates for the inputs from [120], [121].

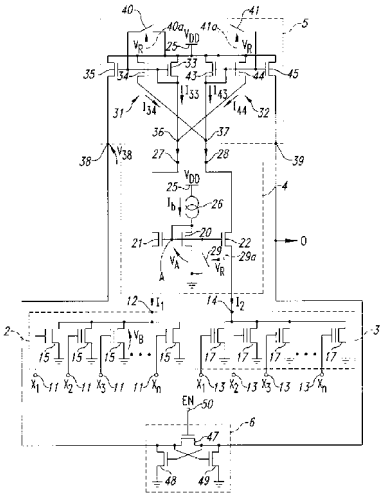


Fig. 32. SET: 3-input MAJORITY gate from [145], programmable gate from [146], and n-input TLG from [147].

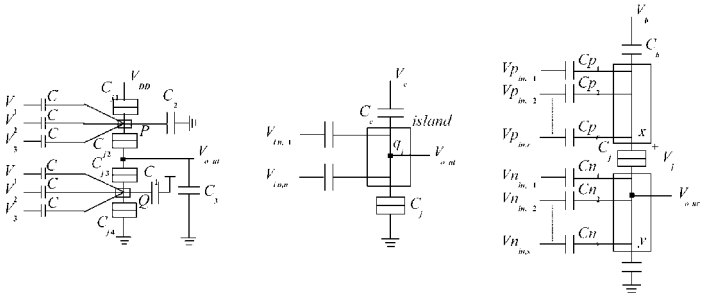
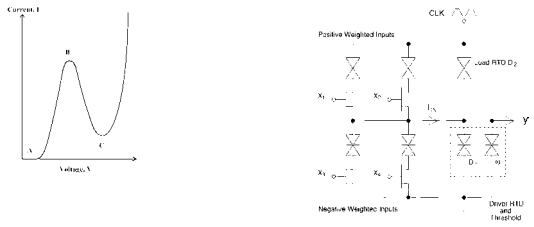


Fig. 33. Current-voltage characteristic of a RTD, and RTD-HFET threshold gate circuit from [164].



Valeriu Beiu (S'92–M'95–SM'96) received the M.Sc. in computer engineering from the "Politehnica" University of Bucharest (Romania), in 1980, and the Ph.D. with "Summa cum Laude" in electrical engineering from the Katholieke Universiteit Leuven (Belgium), in 1994.

After graduating, he has worked for two years on high-speed CPUs and FPUs at the Research Institute for Computer Techniques (Bucharest), before returning to the "Politehnica" University of Bucharest. He has received three fellowships: Fulbright (1991), Human Capital and Mobility (1994-1996) with King's College London (project title: "Programmable Neural Arrays"), and Director's Funded Postdoc (1996-1998) with Los Alamos National Laboratory (project title: "Field Programmable Neural Arrays," under the Deployable Adaptive Processing Systems initiative), before becoming the Head of R&D and Fellow of RN2R/Rose Research (1998-2001). Since 2001 he is an Associate Professor with the School of EE&CS, Washington State University (Pullman, USA). He was the PI of 34 contracts, holds 10 patents (19 are pending), received 23 grants, gave over 60 invited talks, and authored over 180 publications, among which a chapter on digital integrated circuit implementations (of neural networks) in the *Handbook of Neural Computation*, and a forthcoming book on the VLSI complexity of neural networks. His main research interests are on VLSI-efficient design of high-performance (neural inspired) architectures for computational intensive tasks.

Dr. Beiu is a member of INNS, ENNS, ACM, and MCFA, has organized 10 conferences and 18 conference sessions, has received a Best Paper Award (CAS'2001), and has been the Program Chairman of the IEEE Los Alamos Section (1997).

José M. Quintana joined the Department of Electronics and Electromagnetism at the University of Seville in 1983 as Assistant Professor, and obtained the Ph.D. degree with "Summa cum Laude" in 1987. Since 1990 he is Associate Professor in that Department. In 1989 he became researcher at the Department of Analog Design of the National Microelectronics Center (CNM), now Institute of Microelectronics at Seville (IMSE). He has leaded and participated in several research projects financed by the Spanish CICYT and in the ESPRIT Projects ADCIS and AD-2000. He has authored over 60 technical papers in main international journals and conferences. He has organized special sessions on threshold logic and also is co-author of a chapter titled "Threshold Logic" in the *Wiley Encyclopaedia of Electrical and Electronics Engineering*. He won the KELVIN Premium of "The Council of the Institution of Electrical Engineers" for two articles published in 1994. His current research interests include design of threshold logic circuits, non-linear signal processing, computer arithmetic and development of CAD tools for FSM synthesis.

María J. Avedillo joined the Department of Electronics and Electromagnetism at the University of Seville in 1988 as Assistant Professor, and obtained the Ph.D. degree with "Summa cum Laude" in 1992. Since 1995 she is Associate Professor in that Department. In 1989 she became researcher at the Department of Analog Design of the National Microelectronics Center (CNM), now Institute of Microelectronics at Seville (IMSE). She has participated in several research projects financed by the Spanish CICYT and in ESPRIT Projects. She has authored over 60 technical papers in main international journals and conferences. She has organized special sessions on threshold logic and also is co-author of a chapter titled "Threshold Logic" in the *Wiley Encyclopaedia of Electrical and Electronics Engineering*. She won the KELVIN Premium of "The Council of the Institution of Electrical Engineers" for two articles published in 1994. Her current research interests include design of threshold logic circuits, non-linear signal processing, development of CAD tools for FSM synthesis and design for testability.