[8] W.-C. Tseng, Y.-H. Chen, and R.-B. Lin, "Router and cell library codevelopment for improving redundant via insertion at pins," in *Proc. IEEE Int. Conf.*, 2008, pp. 646–651.
[9] A. Diaz and O. Sigmund, "Checkerboard patterns in layout optimization," *Structural Multidisciplinary Optimization*, vol. 10, pp. 40–45, 1995.
[10] UMC, Hsinchu, Taiwan, "L65 SP UM065LSCSPMVBBR logic and mixed-mode standard performance cell library," 2010. [Online]. Available: http://www.umc.com
[11] Synopsys, Inc., Taipei, Taiwan, "Zroute, IC Compiler," 2010. [Online]. Available: http://www.synopsys.com

Scaling Energy Per Operation via an Asynchronous Pipeline

Bo Marr, Brian Degnan, Paul Hasler, and David Anderson

***Abstract*—Statistical analysis of computations per unit energy in processors over the last 30 years is given that illustrates a sharp reduction in the rate of energy efficiency improvements over the last several years resulting in the formation of an asymptotic "wall" with our dataset; we use the measure of giga multiply accumulates per Joule. We have developed an energy model which takes into account the realities of scaling, specifically for asynchronous systems. Studies of an energy efficient asynchronous pipeline show fabricated results of 17 Giga Operations per Joule in 0.6 μm at subthreshold when fully pipelined, and simulations at a more modern 65 nm process show a further order of magnitude improvement on that.**

***Index Terms*—Asynchronous circuits, digital integrated circuits, energy efficiency, integrated circuit modeling, performance analysis, power consumption.**

I. Energy as a Figure of Merit

NEXT-GENERATION *embedded* applications need vast improvement in energy per operation to implement features that are currently envisioned for both commercial and military markets, down to 10 pJ per operation, equivalent to the performance per Watt metric of 100 GOPS/W. This need is urgent and this need is in danger of not being met with current trends because data suggests Moore's law is no longer allowing exponential growth in energy efficiency as shown in Fig. 1.

Fig. 1 shows this trend supported by production chip data and was generated by normalizing a "computation" as a 32-bit multiply accumulate (MAC) operation [1]–[10]. Note that a quadratic relationship between bit-width and energy consumption for multiplication is assumed to normalize computations of different bit-widths. The MAC was chosen because it represents the most important operation in dataflow and signal processing applications, which are the critical applications in embedded systems of interest. The data suggests that we are approaching an asymptote in computational efficiency of digital processors.

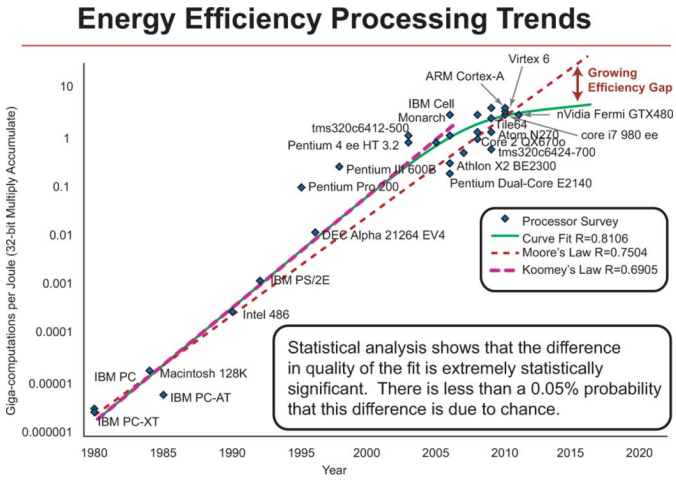

Fig. 1. Energy efficiency (giga multiply-accumulates per Joule) versus year for commercial digital processors. Note that Koomey's law, which assumes operations per energy increases at a faster rate than the Moore's law rate of 2× per every 1.5 years no longer holds as of about 2005 and operations per unit energy starts to fall off creating a larger gap between practice and theory.

A statistical analysis was done on Koomey's and Moore's curves as well as a curve with one additional degree of freedom from a linear fit: a quadratic curve that shows a reduction in energy efficiency improvements over time. The goodness-of-fit of each curve was calculated by the least sum-of-squares method. An F-test was used to verify that the additional variable (degree of freedom) used in the quadratic curve allows a significantly greater quality of fit. The F-test results show that an extremely significant 99.95% confidence exists in this case. The analysis results in the quadratic fit as by far the best fit showing an asymptotic-like behavior on the performance per Watt axis.

This paper presents results of an in-depth analysis of performance per Watt, or equivalently computations per Joule, for recent state-of-the-art processors presented in Section II. Following this, a more accurate fundamental model for energy efficiency in devices is given in Section III as well as possible solutions to the problem in Section IV.

II. Energy Efficiency in Dataflow Systems

The MAC is the classical figure of merit for dataflow-based systems, such as image and natural-signal processing, communications and military-related systems. A critical feature of dataflow systems is that information is stream based, and must be processed in real-time, reducing the effectiveness of post-processing and multi-core approaches. For these reasons, the energy efficiency of a processor in dataflow applications is correlated to inter-related MAC operations in sequence. This accounts for the asymptotic behavior in the energy efficiency curve seen in Fig. 1 and the more detailed and zoomed in version of this plot in Fig. 2. The reasons for this reduction in energy efficiency improvement are both architectural and physical. Architecturally, a *decrease* in efficiency is delivered by multi-core systems. Fig. 3 below shows the energy efficiency of a dataflow processor as the number of cores are increased assuming different levels of parallelization where P represents the fraction of instructions that are parallelizable.

The mathematical basis and assumptions for Fig. 3 are detailed in [11], where the base core equivalent (BCE) model is used to account for different architectures using Amdahl's Law. This model allows for heterogeneous cores of different sizes where a chip is assumed to have

Manuscript received March 15, 2011; revised August 01, 2011; accepted November 10, 2011. Date of publication January 17, 2012; date of current version December 19, 2012. This work was supported in part by a National Science Foundation Fellowship and in part by Raytheon Company.

B. Marr is with the Raytheon Company, El Segundo, CA 90245 USA (e-mail: harry.b.marr@raytheon.com).

B. Degnan, P. Hasler, and D. Anderson are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30308 USA (e-mail: degs@ece.gatech.edu; phasler@ece.gatech.edu; dva@ece.gatech.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVLSI.2011.2178126

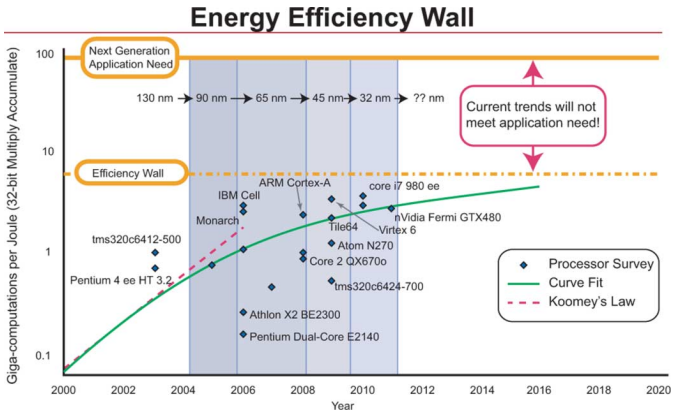

Fig. 2. Apparent energy efficiency asymptote, the gap between processing capability and application need, and general trends of feature size used in leading edge systems.

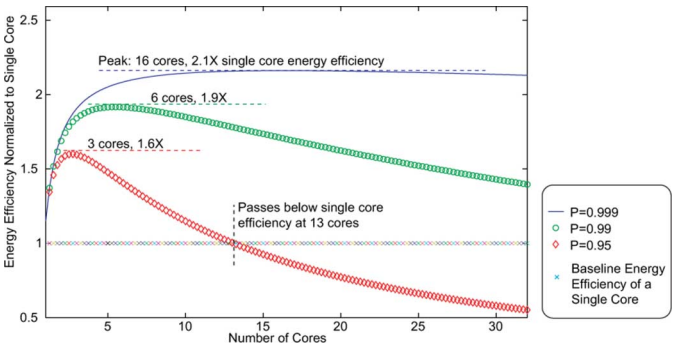

Fig. 3. Energy efficiency normalized to a single core versus number of cores. Plot assumes amount of parallelism in the program is P according to Amdahl's Law. Note that operations per unit energy peaks and then decreases as number of cores increases.

total resources of n BCEs, monolithic cores consisting of r BCEs, and small cores of a single BCE resource. Hence the chip contains n cores in the many-core model; the model also allows for the multi-core model where each core has r resources so that the chip contains n/r powerful cores. Amdahl's law takes the form shown in

$$\text{Performance} = \frac{1}{\frac{1-p}{\text{perf}(r)} + \frac{p}{\text{perf}(r)+n-r}} \quad (1)$$

where $\text{perf}(r)$ is the performance of a core with r BCEs, using Pollack's rule we assume this to be $\text{perf}(r) = \sqrt{r}$ here. Additional assumptions are made here that only a single thread can be used as per the definition of a dataflow application and that eventually a voltage scaling limit [12] does exist where further lowering the supply voltage *increases* energy per operation. Thus assuming each core is at the optimal voltage for energy efficiency, adding more cores increases the total power envelope. The notable result here is that the number of cores for optimal energy efficiency in dataflow applications is quite low even for high levels of parallelization $(P > 0.95)$.

Physically, the electrical consequences and realities of digital design at the nanometer scale result in a decrease in energy efficiency that can be seen in Fig. 2. The asymptotic curve falls off from the linear trend at approximately the 90 to 65 nm minimum feature size node due to the failing of Dennard's Law [13], which predicts that voltage, capacitance, delay, and power of digital circuits all scale according to a well-defined rate as the device dimensions scale. Results predicted by Dennard's Law, observed historically, and observed in recent nodes are shown in Table I. As Table I shows, Dennard's law fails from the 90 nm minimum feature size node to the 65 nm node. The primary reason is that the supply voltage stopped scaling and has remained constant in recent commercial processes [14]. The approximation for the channel current in an ideal nMOS device at saturation is $I_{\text{DSAT}} \propto (W_{\text{eff}}/L_{\text{eff}})((V_G - V_T) - V_S)^2$. Thus with the supply voltage not scaling, $V_G = V_{DD}$ in saturation, and with second order effects such as velocity saturation, saturation currents are not scaling either.

TABLE I
FAILING OF DENNARD'S LAW: FIRST-ORDER EFFECTS FOR SUB-100 nm CMOS

Parameter	Dennard's Factor	500nm→ 350nm	90nm→ 65nm
Dimension T_{ox}, L, W	α	0.7	0.7
Voltage	α	0.7	1
Current	α	0.7	1
Capacitance	α	0.7	0.7
Delay(VC/I)	α	0.7	0.7
Power Dissipation(VI)	α^2	0.5	1

Certainly process improvements such as metal gates, FinFETs, and architecture improvements such as turning off cores, have enabled power dissipation per gate to decrease in subsequent chip generations, even in sub-100 nm nodes, but not nearly as fast as Dennard's law predicts. Secondly, Dennard's law also assumed the wire delay and capacitance are negligible compared to logic delay and capacitance of logic, and this assumption is no longer valid at sub-100 nm nodes. In [15], the authors showed the average chip-wide wire length actually increases super-linearly as a function of the number of gates in the chip, impacting large interconnect networks. A third factor causing the energy efficiency asymptote phenomenon is manufacturing (process) variations. As devices have scaled down, manufacturing variations have become extreme, delay of a gate can vary up to 50% or more of the intended value [16]. It is no coincidence that troublesome process variation effects are being seen at the same nodes as the beginning of the asymptotic behavior of energy efficiency.

III. Energy Delay Model

This work uses the energy delay product (EDP), expressed as $E\tau$ to quantify energy efficiency. The $E\tau^{\alpha}$ metric is useful to compare technologies because $E\tau^{\alpha}$ reveals not only how much electrical energy is used, but how efficiently this energy is used towards performing computation where different values of α allow different design points to be explored. In this work, $\alpha = 1$ was chosen because this metric results in measuring energy per operation [17] as opposed to power per operation or some other metric, and we are concerned with minimizing energy. Much work has been done to model EDP minimization with several limiting assumptions on operating behavior and region, for example the seminal low power CMOS work given in [18], [19]. The traditional model for delay is based on the dynamic charging of a capacitive load through a single network.

This simple model upon which much literature assumes for energy efficiency calculation, where the computation time $T_d = V_{dd}/(V_{dd} - V_t)^2$ ignores leakage current draining charge off of the load. The simple model also assumes supply voltage is much greater than threshold voltage $V_{dd} \gg V_{th}$. Once again, this model holds historically where leakage is insignificant during dynamic charging and V_{dd} is large, but not at sub-100 nm nodes where leakage is significant or with technologies with near-threshold supply voltages. We augment this previous work with an EKV model. This model not only takes leakage into account, which is a simple addition of the current drawn in the pull-down network, but is correct throughout

supply operating point including near and subthreshold voltages. The resulting description for delay is shown in

$$T_d = \frac{KC_{\rm load}V_{dd}}{I_{\rm pFET} - I_{\rm nFET}} \tag{2}$$

where $I_{\rm FET}$ is independent of operating point described in the Appendix. In (2), the leakage term is subtracted in the denominator, and therefore T_d increases.

Suppose V_{dd} is close to the threshold voltage, then the increase in T_d compared to nominal operation is significant. T_d ties closely into energy because energy is modeled as the electrical work to charge and discharge the capacitive load, $C_{\rm load}$ taking into account the electrical work wasted to leakage during the time it takes to perform this charging or discharging. Mathematically, the energy for a gate-level operation is

$$E_{\rm Total} = \alpha C_{\rm load}V_{dd}^2 + I_{\rm leak}V_{dd}t_{\rm cycle} \tag{3}$$

where α is the activity factor. In (3), we assume that the static energy is the leakage power of the digital gate times the delay of the computation, $t_{\rm cycle}$. $t_{\rm cycle}$ is highly related to T_d in that T_d is the delay to charge a gate while $t_{\rm cycle}$ is the cycle time for an entire operation for which a given gate is a part. Thus, if one wishes to reduce the total static energy used, one needs to minimize $I_{\rm leak}$ or $t_{\rm cycle}$.

IV. Asynchronous Design as a Possible Solution

One possible solution to this trend in energy efficiency is asynchronous design. Fine grained asynchronous pipeline work by Singh *et al.* [20] shows that dynamic logic, where an asynchronous logic cell can accept new input values while still being precharged and thus acting implicitly as a latch, can be used to remove much of the overhead involved with handshaking. The delay of a pipeline stage $t_{\rm pipestage}$ is reduced to about 8 FO4 inverter gate delays with this technology, which is far less than is achievable with most synchronous designs which typically range from 16 to as many as 40 gate delays per pipeline stage in current designs. When $t_{\rm pipestage} = t_{\rm cycle}$ in asynchronous logic this results in reduced leakage energy per operation.

Variation tolerant subthreshold asynchronous design was explored in [21] where the authors showed using asynchronous logic yields significant advantages with sub-100 nm processes that are subject to delay variability.

It has been well studied that the optimal operating point for EDP is when the supply voltage is near threshold, $V_{dd} \approx V_{th}$ [12]. Whether this operating point is above, at, or below threshold is dependent on the ratio of active devices to leaky devices *and the speed of the given operation*. In modern architectures, the optimal EDP point is slightly above threshold because this allows the active gates to perform computation faster, thus reducing leakage energy consumed per computation. With asynchronous logic, the $t_{\rm cycle}$ can be reduced compared to clocked logic, and the optimum EDP point is at a supply voltage slightly lower than threshold, allowing for a lower energy per operation.

An experiment was done to verify this with the energy consumed by a 32-bit ripple-carry add operation. The time to complete this add operation is $t_{\rm add32}$, which is data-dependent in asynchronous adders, and can be given for an entire system as the average $t_{\rm add32}$ over N uniformly distributed random inputs

$$t_{\rm add32,async} = \frac{1}{N}\sum_{i}^{N} t_{\rm add32,i}. \tag{4}$$

However, for synchronous logic, $t_{\rm add}$ must by definition be clocked at the worst case, a ripple through all 32 bits in this case, resulting in

$$t_{\rm add32,sync} = \max(t_{{\rm add32},i}). \tag{5}$$

Consider a 32-bit adder. If $t_{\rm fulladd}$ is the delay of a single full adder, then in the synchronous case each full adder will consume energy due to leakage for a time equal to $t_{\rm add32} = 32 \cdot t_{\rm fulladd}$. Meanwhile the average critical path length of the adder is $\log N$, the asynchronous ripple carry adder is able to take advantage of this and leaks for a time equal to $t_{\rm add32} = 5 \cdot t_{\rm fulladd}$, now a reset must occur equal to the delay of approximately $2 \cdot t_{\rm fulladd}$ as per the design used in [20], for a total delay of $7 \cdot t_{\rm fulladd}$. In general the asynchronous full adder leaks for $(\log N + 2) \cdot t_{\rm fulladd}$ in an N-bit adder whereas the synchronous case leaks for $N \cdot t_{\rm fulladd}$. Thus we can set $t_{\rm cycle} = t_{\rm add32}$ in (3) to calculate energy per 32-bit ripple carry add operation.

In [22], the authors dispute average case asynchronous timing. However, they present no complexity analysis confirming the simulated results, and refer to the average case *delay* of components and this paper refers to the average case *critical path* through a given arithmetic unit.

Interestingly with this experiment, the optimal supply voltage is lower with asynchronous logic than with synchronous logic. The V_{dd} at which minimum EDP occurs drops from 350 to 260 mV shown in Fig. 4(a) with the resulting static energy reduction shown in Fig. 4(b).

The resulting asynchronous energy delay product for a gate in this adder becomes

$$\text{EDP} = E_{\rm Total}T_d = \frac{KC_{\rm load}\left(\alpha C_{\rm load}V_{dd}^3 + I_{\rm leak}V_{dd}\log(t_{\rm cycle})\right)}{I_{\rm pFET} - I_{\rm nFET}}. \tag{6}$$

V. Energy Efficient Asynchronous Pipeline

We designed and fabricated several asynchronous arithmetic units surrounded by a register-file considering the issues of voltage scaling, EDP minimization and dataflow efficiency. The core of the IC shaded in blue in Fig. 4(e) consists of a single timing plane, where all timing is generated by the asynchronous datapath logic; the registers are clocked by the bit-wise datapath acknowledgement signals, which can be initiated by an enable signal when data is present. The reservation station registers when enabled, shown in Fig. 4(d), feed the asynchronous pipeline where handshaking is done at the full-adder level in parallel within the pipeline. The datapath involves a 16×16-bit array multiplier and a 16-bit carry-skip adder where the multiplier MSBs are dropped. The reorder buffers collect results at the full width of the datapath and generate a completion signal for the entire 16-bit result. The reorder buffer contains multiplexor logic such that the jth result from a given datapath output bit is stored in the jth buffer; when the last buffer slot gets filled for any of the 16-bits, the reservation station stalls the pipeline until an entire 16-bit word is completed, put onto the bus, and the cycle starts over from the first buffer slot.

The logic in yellow in Fig. 4(e) is off-chip, implemented with a breadboard and a PC such that an external clock and enable/disable signal is used to write into the reservation stations on the IC.

This logic style allows for computations to be calculated on a bit-by-bit basis independent of previous and future computations, assuming a full pipeline. For example, if bit numbers 0 and 1 of an addition finish in the first operation, the computation in these bits for the second operation starts immediately, even while bits 2 to $N-1$ are still computing from the first operation. This is possible due to the reservation stations filling the processing blocks, and the reorder buffers sorting the result. This results in a $\mathcal{O}(\text{constant})$ T_d as the bit-width of an operation scales, shown in Fig. 4(c).

VI. Chip Results and Conclusion

A test chip was designed and fabricated in 0.6 μm, where initial design stages are discussed in [23] with asynchronous pipeline results shown for a style following most closely after the PS0 style from [24]. Dual-rail domino logic was chosen where an inverter follows the inverted dynamic logic for each function for robustness against

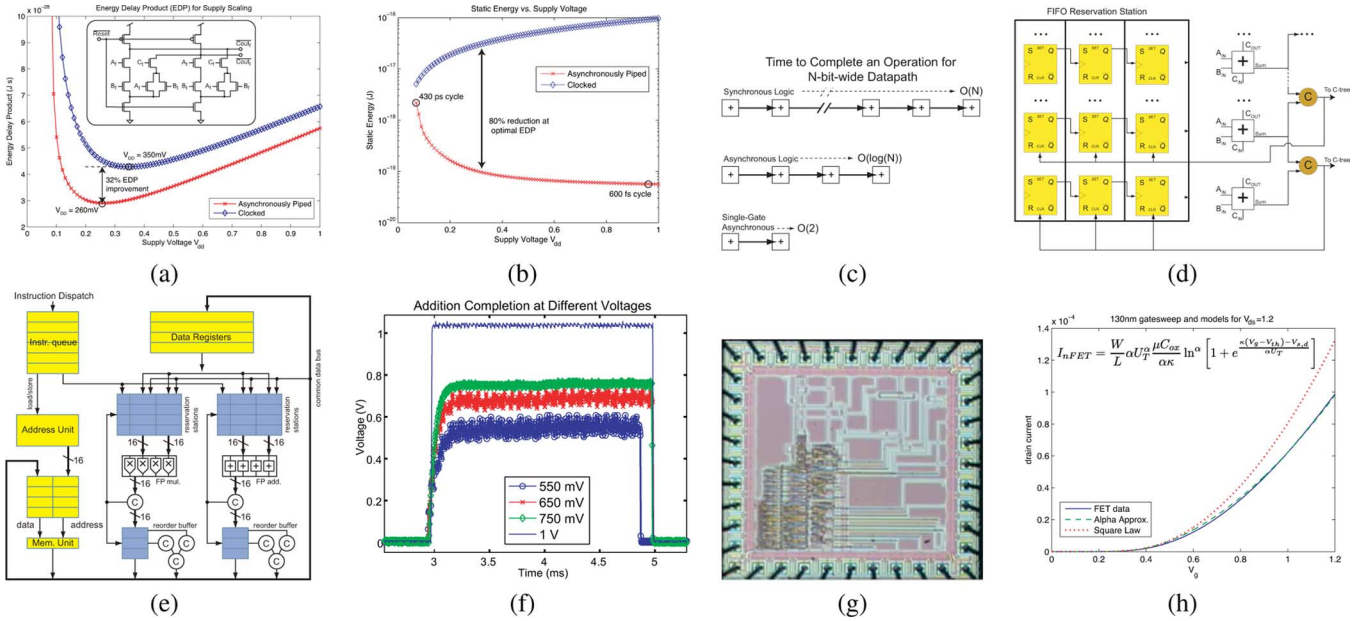

Fig. 4. (a) EDP versus V_{dd} for a digital gate in an arithmetic unit of both synchronous and asynchronous systems. The schematic of the asynchronous full adder used in this work is the inset. (b) Leakage energy per operation versus supply voltage for clocked versus asynchronous digital gates used in an arithmetic computation for an inverter was simulated in a 32-nm process. (c) Illustration of the complexity analysis of the delay for clock, asynchronous, and single-gate-asynchronous logic, assuming a full pipeline. (d) Asynchronously clocked reservation station. (e) The architecture of the test IC where the asynchronous arithmetic units are embedded in a clocked architecture through the reservation stations and reorder buffers. (f) Voltage output of the completion indicator across voltages showing operation well into subthreshold. (g) Micrograph of the die of the test IC. (h) "Square-Law" compliance compared to measured data combined with the α-fit that was used to calculate EDP independent of operating region.

charge sharing, voltage drop issues, and floating voltage node issues. A restorative inverter was also needed to allow subthreshold voltage scaling. A footer transistor was only used to implement fully controlled asynchronous logic only when the pull down stack could be kept less than or equal to three transistors. Pull down stacks larger than this resulted in slow logic and failures when voltage scaling. Experiments to take the inverter out of the carry-chain resulted in non-robust voltage levels when the output node of the asynchronous logic floated such as during an evaluation phase where valid inputs had not yet arrived.

A result of one Giga operations per second (GOp/s) was measured at the full voltage of 5 V when the design is fully pipelined; note that the arithmetic units are pipelined at the full adder level, such that a full adder pipeline stage is completed at this rate. Also note our results do not consider effects due to stalling. The chip was also designed to be tested at subthreshold voltages where in Fig. 4(f) an enable switch was used such that the rise and fall times could be compared at different voltages. The optimal EDP voltage was at a subthreshold voltage of 550 mV, achieving a measured 17 Giga Ops per Joule, 17(GOps/J). The V_{th} = 700 mV in this process. The latches in the reservation station and the reorder buffers were the first to fail in attempts to scale below 550 mV. The observation was made that the optimal EDP is lower than the at-threshold value predicted for synchronous logic. This translates into a performance of 56 pJ per operation in a 0.6-μm process. The simulation of this architecture yielded a power performance of 1.58 pJ per operation at 65 nm, which is below the 10 pJ per operation target.

The risks and limitations of this design are that it is predicted that performance will drop off significantly with pipeline stalls. In a real design such as a large FIR filter, it is recommended that the designer insert *identity* elements to prevent stalls; an identity element simply propagates the input to the output and implements an asynchronous evaluation cycle for this operation. Thus identity elements can serve as a buffer of data to prevent stalls. An elastic pipeline with a variable number of identity elements could be inserted using a multiplexor would be ideal. Permanent stalls downstream from a fork in the datapath, such as if one of the arithmetic units was fed by I/O that hadn't arrived yet actually violates the isochronic fork assumption and could cause this logic to fail, we assumed the isochronic fork assumption held in this work.

This work was semi-custom layout as no commercially available tools exist on the open market for this type of design outside of academic groups and proprietary tools such as used by Achronix. However, if these hurdles are crossed, asynchronous design continues to become a leading candidate for a solution to continuing to scale energy efficiency.

Appendix

The Enz-Krummenacher-Vittoz (EKV) model describes the transistor's operation continuously between the subthreshold region of diffusion-based charge movement to the above-threshold region of drift-based charge movement [25]. A variant of the EKV model is the compact EKV model which interpolates around the threshold current [26]. The "square law" form of drift movement no longer holds due to higher order effects [27]–[30]. The compact EKV model is easily modified for an empirical "α" fit [26]. The form of $ln^{\alpha}\left(1+e^{x/\alpha}\right)$ degrades above threshold region while leaving the subthreshold region untouched. The resulting modification is

$$I_{\mathrm{nFET}} = \frac{W}{L} 2U_T^2 \frac{\mu C_{ox}}{2\kappa} \ln^{\alpha}\left[1 + e^{\kappa\left(V_g - V_{th}\right) - V_{s,d}/\alpha U_T}\right] \quad (7)$$

where α is a fit to the above threshold behavior. Fig. 4(h) shows this fit compared to measured nFET data from a commercially available 130-nm process. The current through the device is a function of the forward and reverse current, and as the drain voltage increases, the reverse contribution to the current decreases. For a V_{ds} greater than 125 mV, the reverse current is approximate 1% of the channel current and the device can be considered to be in saturation.

ACKNOWLEDGMENT

The authors would like to thank K. Prager, M. Trainoff, and B. Pierce from Raytheon for their contributions. They would also like to thank the anonymous reviewers.

REFERENCES

[1] TI, Dallas, TX, 2011. [Online]. Available: http://power.ti.com
[2] AMD, Sunnyvale, CA, 2011. [Online]. Available: http://www.amd.com/us/products/Pages/products.aspx
[3] Intel. [Online]. Available: http://ark.intel.com/Default.aspx
[4] D. Patterson and J. Hennessy, *Computer Architecture: A Quantitative Approach*, 3rd ed. Norwell, MA: Morgan Kaufmann, 2003.
[5] J. Koomey, S. Berard, M. Sanchez, and H. Wong, "Assessing trends in the electrical efficiency of computation over time," *IEEE Annals History Comput.*, to be published.
[6] D. Pham, S. Asano, M. Bolliger, M. N. Day, H. P. Hofstee, C. Johns, J. Kahle, A. Kameyama, J. Keaty, and Y. Masubuchi *et al.*, "The design and implementation of a first-generation CELL processor," in *Proc. Int. Solid-State Circuits Conf. (ISSCC) 2005–2010*, 2005, pp. 184–592.
[7] D. Krueger, E. Francom, and J. Langsdorf, "Circuit design for voltage scaling and SER immunity on a quad-core Itanium processor," in *IEEE Int. Dig. Tech. Papers Solid-State Circuits Conf.*, 2008, pp. 94–95.
[8] R. Kumar and G. Hinton, "A family of 45nm IA processors," in *Dig. Tech. Papers Solid-State Circuits Conf.*, 2009, pp. 58–59.
[9] J. Friedrich, B. McCredie, N. James, B. Huott, B. Curran, E. Fluhr, G. Mittal, E. Chan, Y. Chan, and D. Plass *et al.*, "Design of the Power6 microprocessor," in *Dig. Tech. Papers Solid-State Circuits Conf.*, 2007, pp. 96–97.
[10] S. Bell, B. Edwards, J. Amann, R. Conlin, K. Joyce, V. Leung, J. MacKay, M. Reif, L. Bao, and J. Brown *et al.*, "Tile64-processor: A 64-core SOC with mesh interconnect," in *Dig. Tech. Papers Solid-State Circuits Conf.*, 2008, pp. 88–598.
[11] D. H. Woo and H.-H. Lee, "Extending amdahl's law for energy efficient computing in the multi-core era," *IEEE Comput.*, vol. 41, no. 12, pp. 24–31, Dec. 2008.
[12] S. Hanson, , B. Zhai, K. Bernstein, D. Blaauw, A. Bryant, L. Chang, K. K. Das, W. Haensch, E. J. Nowak, and D.M. Sylvester, "Ultralow-voltage, minimum energy cmos," *IBM J. Res. Develop.*, vol. 50, no. 4, 2006.
[13] T. H. Ning, "A perspective on theory of mosfet scaling and its impact," *IEEE Solid State Circuit News: The Impact of Dennard's Scaling Theory*, vol. 12, no. 1, pp. 27–30, 2007.
[14] MOSIS, Marina Del Rey, CA, 2011. [Online]. Available: http://www.mosis.com
[15] J. Meindl, J. Davis, and V. K. De, "A stochastic wire-length distribution for giga-scale integration (GSI)—Part II: Applications to clock frequency, power dissipation, and chip size estimation," *IEEE Trans. Electron Devices*, vol. 45, no. 3, pp. 580–589, 1998.
[16] K. Bowman, J. Tschanz, C. Wilkerson, S. L. Lu, T. Karnik, V. De, and S. Borkar, "Circuit techniques for dynamic variation tolerance," in *Proc. 46th Annu. Design Autom. Conf.*, 2009, pp. 4–7.
[17] J. Teifel, D. Fang, D. Biermann, C. Kelly, and R. Manohar, "Energy-efficient pipelines," in *Proc. Int. Symp. Asynch. Circuits Syst. (ASYNC)*, 2002, pp. 23–33.
[18] A. P. Chandrakasan, S. Sheng, and R. W. Brodersen, "Low-power CMOS digital design," *IEEE J. Solid-State Circuits*, vol. 27, no. 4, pp. 473–484, 1992.
[19] B. Calhoun, A. Wang, and A. Chandrakasan, "Modeling and sizing for minimum energy operation in subthreshold circuits," *IEEE J. Solid-State Circuits*, vol. 40, no. 9, pp. 1778–1786, Sep. 2005.
[20] M. Singh and S. Nowick, "The design of high-performance dynamic asynchronous pipelines: High-capacity style," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 15, no. 11, pp. 1270–1283, Nov. 2007.
[21] I. J. Chang, S. P. Park, and K. Roy, "Exploring asynchronous design techniques for process-tolerant and energy-efficient subthreshold operation," *IEEE J. Solid-State Circuits*, vol. 45, no. 2, pp. 401–410, 2010.
[22] M. Greenstreet and B. Alwis, "How to achieve worst case performance," in *Proc. Symp. Asynch. Circuits Syst. (ASYNC)*, 2001, pp. 206–216.
[23] B. Marr, B. Degnan, P. Hasler, and D. V. Anderson, "An asynchronously embedded datapath for performance acceleration and energy efficiency," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, 2009, pp. 3046–3049.
[24] T. E. Williams, "Self-timed rings and their application to division," Ph.D. dissertation, Stanford Univ., Stanford, CA, 1992.
[25] C. C. Enz, F. Krummenacher, and E. A. Vittoz, "An analytical MOS transistor model valid in all regions of operation and dedicated to low-voltage and low-current applications," *Analog Integr. Circuits Signal Process.*, vol. 8, no. 1, pp. 83–114, 1995.
[26] S. C. Liu, *Analog VLSI: Circuits and Principles*. Cambridge, MA: MIT Press, 2002.
[27] C. Mead and V. Analog, *Neural Systems*. Reading, MA: Addison Wesley, 1989.
[28] T. A. Fjeldly and M. Shur, "Threshold voltage modeling and the subthreshold regime of operationof short-channel MOSFETs," *IEEE Trans. Electron Devices*, vol. 40, no. 1, pp. 137–145, Jan. 1993.
[29] J. T. Watt and J. D. Plummer, "Universal mobility-field curves for electrons and holes in MOS inversion layers," in *Symp. VLSI Technol. Dig. Techn. Papers.* , 1987, pp. 81–82.
[30] R. van Langevelde and F. M. Klaassen, "Effect of gate-field dependent mobility degradation on distortion analysis in MOSFETs," *IEEE Trans. Electron Devices*, vol. 44, no. 11, pp. 2044–2052, Nov. 1997.

A High Speed Low Power CAM With a Parity Bit and Power-Gated ML Sensing

Anh-Tuan Do, Shoushun Chen, Zhi-Hui Kong, and Kiat Seng Yeo

***Abstract*—Content addressable memory (CAM) offers high-speed search function in a single clock cycle. Due to its parallel match-line (ML) comparison, CAM is power-hungry. Thus, robust, high-speed and low-power ML sense amplifiers are highly sought-after in CAM designs. In this paper, we introduce a parity bit that leads to 39% sensing delay reduction at a cost of less than 1% area and power overhead. Furthermore, we propose an effective gated-power technique to reduce the peak and average power consumption and enhance the robustness of the design against process variations. A feedback loop is employed to auto-turn off the power supply to the comparison elements and hence reduce the average power consumption by 64%. The proposed design can work at a supply voltage down to 0.5 V.**

***Index Terms*—CMOS, content addressable memory (CAM), match-line.**

I. INTRODUCTION

Content addressable memory (CAM) is a type of solid-state memory in which data are accessed by their contents rather than physical locations. It receives input search data, i.e., a search word, and returns the address of a similar word that is stored in its data-bank [1].

In general, a CAM has three operation modes: READ, WRITE, and COMPARE, among which "COMPARE" is the main operation as CAM rarely reads or writes [4]. Fig. 1(a) shows a simplified block diagram of a CAM core with an incorporated search data register and an output encoder. It starts a compare operation by loading an n-bit input search word into the search data register. The search data are then broadcast into the memory banks through n pairs of complementary search-lines (SLs) and directly compared with every bit of the stored words using comparison circuits. Each stored word

Manuscript received April 29, 2011; revised July 25, 2011; accepted November 10, 2011. Date of publication January 23, 2012; date of current version December 19, 2012.

The authors are with Virtus, IC Design Centre of Excellent, School of Electrical and Electronics Engineering, Nanyang Technological University, Singapore 639798 (e-mail: atdo@ntu.edu.sg; eechenss@ntu.edu.sg; zhkong@ntu.edu.sg; eksyeo@ntu.edu.sg).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVLSI.2011.2178276