The Magic School Bus Travels to Sub-threshold Voltage

By Dr. Henry Jekyll, Caviendish Laboratory, Cambridge

How an obscure, 22 year old circuit can limit leakage near the theoretical minimum where just the electrons that count- count

Five years ago, Electronic Design covered a new, low-power microcontroller that used an analog concept found in integrated circuits as far back as the 1968 CMOS-based RCA CD4007 (now made by TI). It described how the conventional wisdom regarding leakage currents does not apply at the lowest sub-threshold voltages. The article did a thorough history on the research and progress of the technology from analog to digital circuit applications.[26] The latter application, in digital circuit design, remains a widely unknown technology. It has even taken myself over 3 years to better understand. A simple analogy might work, but from experience, one analogy may not fit all point of views. The concept of drowsy logic has a connection to sub-threshold voltage, although that link will be detailed more later in the article. What is first important to review, is why these techniques are used- the ultimate goal may not be energy efficiency itself, but what new products or cost savings energy efficiency can allow. The Electronic Design article cited a 2016 dissertation[33] by Dr. Sunhil Kim, who cited 0.55V as the optimum Power Delay Product of 7.2 (p.28/78), achieving a 99% reduction in power:

Table 3.2. Power and delay for various voltages of the multiplier

Supply Voltage (V)	Total Power (μW)	Delay (ns)	Power-Delay Product
0.20	0.05	4700	235
0.40	0.15	120	18.0
0.45	0.60	41	24.6
0.50	1.40	12	16.8
0.55	1.80	4	7.20
0.60	6.00	1.60	9.60
0.80	110	0.18	19.8
1.00	640	0.07	44.8

To start with this table, a 2020 text defined energy efficiency not by a power-delay-product[75], but by an energy delay product. A 2013 text states, "However, subthreshold voltage devices may prefer PDP metrics where leakage can present an operational or medical risk."[61]

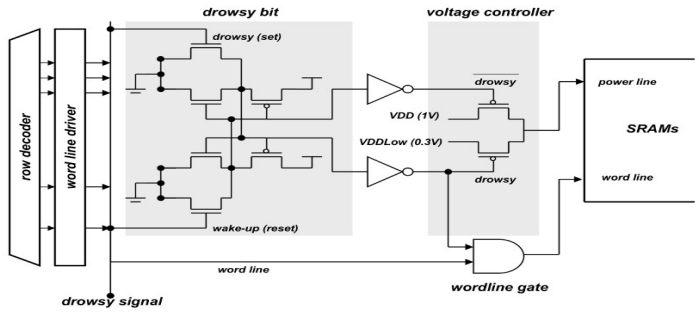
"The proposed work uses energy-delay product (EDP), where energy the total energy consumption of cores and delay is the amount of time for executing applications. Spiliopoulos et al. [6] used EDP and ED^2P, where execution time was computed as the average execution time over multiple programs that run simultaneously on their multicore systems. Murray et al. [9] used EDP. Tarplee et al. [12] defined Pareto frontiers to tradeoff energy consumption versus execution time. Jung et al. [13] measured energy consumption for different classifiers. Lai et al. [14] computed EDP for each phase of their applications to optimize an EDP target. Wang et al. [10] used profit as explained above. Chen et al. [15] used power consumption."[62]
EDP efficiency could also be analogized to computational benchmarks, such as MIPS or DMIPS(Dhrystones), divided by Power(W)[54]. Depending on the instruction sets and application (e.g. HCI-MIPJ/glyph[76]), synthetic benchmarks may not always be comparable to other architectures[77].

Nonetheless, now that 0.55V as a local minima can be selected as point-of reference for sub-threshold logic, it is important to contrast this circuit from the point of view of the user, rather the circuit-side. In short, digital logic is like a light switch turning on and off, communicating as a Morse lamp, but glowing afterwards for a few seconds. While there are various types of bulbs that have multiple reasons for this[65], such as a hot filament in an incandescent, or phosphorous coating interacting with residual ions of mercury in a CFL, or even phantom voltage on LEDs with poor insulation or lacking earthing wire, the most analogous example to a computer transistor is the capacitance inside an LED circuit in a normal operation. Instead of needing a 120V AC current to power a 60 Watt incandescent, an LED running at 12V DC could power a 0.75 Amp bulb that uses 9 Watts whilst outputting the same number of lumens (~ 800 @2700K).
Isolating the cells is important, to not only reduce leakage and soft-memory errors from capacitive cross-talk[36], but also to improve speed by the same factor[26]. With an ultra low leakage foundry process node such as 22nm ULL,[37] low-voltage transistors still operate like transistors or switches, but may operate in weird and idiosyncratic ways, requiring more non-Gaussian LVF modeling and prediction in the EDA stage[38].

In optimal conditions, setting a voltage that maintains the operations of transistors at the lowest power level would only be limited by the amount of performance needed by the application or operating system. IoT devices have utilized commercial sub-threshold microcontrollers for just over 10 years. More complex processors are just starting to be developed. Alternative energy-efficient chips such as static cores[39] utilize other techniques to save power, although might consume approximately the same amount if needed to run a heavy workload. Furthermore, past static designs were more common in the 1980s-90s, typically operate at higher threshold voltages, and new process nodes have produced few, if any new designs compared to dynamic logic.

A drowsy circuit can be analogized to a light switch being turned off and on repeatedly by a playful child. While the voltage switch itself appears to be exploited for fun, the circuit itself has significant amperage waiting as the circuit is technically "on," with sufficient electrons to maintain state, IPC and the instruction cache utilized. Thus the average voltage using a 0.55V threshold setting might be ½ of that, or 0.275V, if the voltage switch was on 50% of the time. The slide on next page depicts Ambiq's threshold voltage example as 0.5V[34]. However, to the human eye, a flickering LED might appear more like a strobe light, even with residual glow from capacitance leaking into the LED. The flashes are like "accessed" drowsy bit lines. Capacitance builds up before it reaches the transistor, but waits less time in a series-based short-channel MOS with lower leakage when set to a lower voltage.[56] The 2 extra transistors needed for the drowsy circuit add complexity but save power[3]:

FIGURE 2. Implementation of the drowsy cache line



Note that, for simplicity, the word line, bit lines, and two pass transistors in the drowsy bit are not shown in this picture.

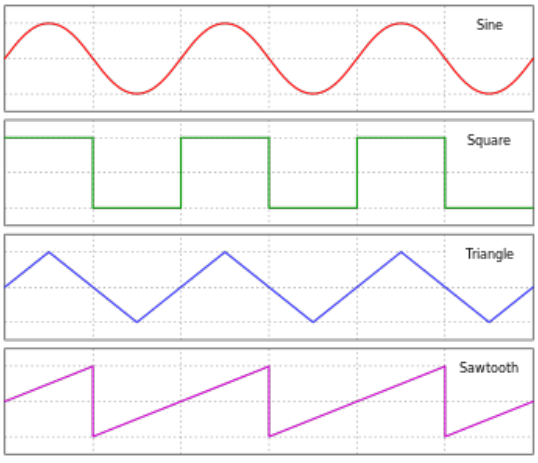
To better understand how the analogy works at the circuit side, it is more apt to use an expressway analogy during rush hour traffic. An observer on an overpass may be able to see cars on an otherwise 65 mph speed limit expressway traveling at 30 miles per hour. Assume, for the sake of simplicity, that is a passenger car commuting or traveling for leisure in between every 18 wheeler fully loaded with commercial goods (e.g. en route to a Big Box store that arrived from a container ship yard). The passenger cars represent electrons that are bit padding, not doing useful work necessarily (unless the instructions are CISC/Out-of-Order), whereas the trucks are carrying very important or valuable data for the user or customers on a network (shoppers). Thus the tracelines in the chip's IPC are where electrons are traveling with significant amperage-not as much as 500mV (65mph), but still a lot with an average net transfer of imports to an inland location over a given amount of time (e.g. 12 hours). The switch that is turned off effectively introduces traffic, causing 4 lanes to run at 30mph, also similar to the throughput of two lanes being open in non-rush hour traffic at 65 mph. Since amperage is not based on width or height, the analogy of 2 lanes running parallel or above or besides two others is less relevant than the number of semi-trucks per hour carrying "bytes" of data at a given speed. Thus, to an observer on an overpass, the 30mph traffic may appear to be slow, but they are typically focused on the speed of the individual vehicles rather than the throughput and cumulative number of bytes (goods) of all the vehicles on that route.

The speed of electrons in a wire travel signal velocity, which is "somewhat close to speed of light in a vacuum"[39] or 60%[41], rather than the drift velocity. In any case, it is not possible for an observer to see the individual electrons, and a classical text book chemical bond diagram of H_2O is not suggesting that 2 electrons of Oxygen are permanently bonded to Hydrogen or even temporarily necessarily for longer than a very small fraction of a second. In that context, the location an electron occupies in atom or molecule is more accurately represented by a valence orbital, which describes the *probability* that an electron will be found: s, p, d, and f orbitals[43]. Drowsy logic, however, is not anywhere near a probabilistic processor[44], which calculates the probability of a bit being 1 instead of 0, rather than a certainty. As stated above, Ambiq Micro microcontrollers (32-bit) still operates like a CMOS transistor and switch, even with some variance or yield losses. It is by this analogy that measurements in most, if not all of engineering uses SI units instead of Gaussian cgs, and Heaviside-Lorentz[45] units instead of factors of 4π. SI units operate on the concept of stochastic probability, rather than exact quantities as in Gaussian units.

To get a close-up view of an actual drowsy circuit, one might try to use an electron ptychrography microscope[46], which reconstructed the highest resolution of an atom crystal, $PrScO_3$ in 2021. However, electrons are 100 million times smaller than atoms, and due to Heisenberg's uncertainty principle, cannot be found. A multimeter would measure electrons far more practically.

If one were to use the Magic School Bus as an example, shrinking to the size of the individual bits transported at signal velocity, one might be able to determine the properties of the circuit's amperage, and whether programmed dynamic voltage[47] drops in fixed intervals timed to reliably renew the circuit before decaying provides enough throughput for IPC and cache. In other words, the fundamental properties of the electrons are relied upon for optimizing the lowest leakage circuit. Electrons are not sentient like observers, and cannot know that a voltage switch caused a threshold to be reached for power on/off state.

Therefore, the Magic School Bus example might make more sense if the drowsy circuit were operating in a SeaLab in the Mariana Trench, and the light switch was being frequently turned on and off aboard a research boat above and connected via a marine-grade cable to the Sea-Lab. Ms. Frizzle's class at the bottom of the Mariana trench would not be able to hear the student's light switch atop the boat turn off (which would travel much slower than the optical cable, if it could even reach audible levels) nor when the switch was being flipped, but they would be able to see the downstream effect on the ebbing and cresting waveform[51]:



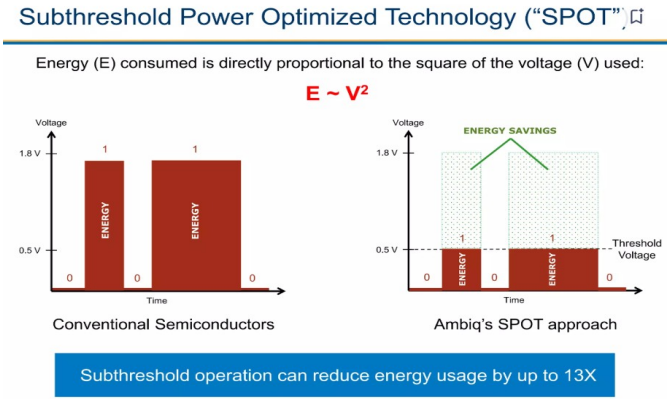
(Source: Wikimedia)

By this analogy, then, the dynamic scaling of voltage transforms analog controls on a circuit into a digital and predictable operation, despite the feared wave function collapse[74]. Like the old joke, if 4 pies are brought to a Heaviside-Lorentzian forest, does anyone eat them? If there is an infinitely small gap in a mathematical set, it is still known as a continuous function[52].

At most, it might create a soft-error in a redundant and/or inconsequential bit. At worst, as in genetics, it could cause a missense mutation. Thus exploring whether drowsy logic is on the threshold of "discontinuous" functions can better classify the limits of sub-threshold technologies. Like an underwater cave or SeaLab, the presence of an air pocket is the exception rather than the rule in the ocean of high impedance sub-threshold voltage.

Since there is not an infinite amount of oxygen (supply voltage) in an underwater cave, unique applications need to be designed, much like calculating the maximum biomass of animals(s) or insects that could survive in a hypothetical cave under the Mariana trench, as would a sealed terrarium would need to allow sufficient photosynthesis from sunlight to replenish the oxygen consumed by aerobic bacteria and other insects. The added difficulty, is that at the bottom of the sea, there is no light, although that analogy does not need to be taken literally or be completely congruous. It could also be that there are enough anaerobic bacteria that are able to produce oxygen as a byproduct for other species in the air pocket. Part of the reason IoT is used as an experimental, and low-cost test bed for new and energy-scarce devices is that they can be marketed for non-essential applications such as wearables until the technology improves its error-correction techniques.

Early computer power supplies were not designed with undervoltage protection. Inexpensive and unsafe electronics may also be missing certain voltage loss detection circuitry, which could lead to damaged components. Improved power supplies also have circuitry that can also prevent data loss, but also depend on software that is designed to save data either to disk either as soon as it is written to RAM, or directly accessed and modified in place (XiP). Static processors, and some emerging research, uses ferrostatic FRAM[48, 49] and magnetoresistive RAM (MRAM) to save data. Whether this is crucial to facilitating the use of drowsy and sub-threshold voltage processors remains to be seen. Exploring whether sub-threshold processors can operate without interruption, despite their idiosyncratic quirks and potential voltage drops, may be able to extend the life of conventional memory technologies despite having higher power-consumption. Potentially controlled "leakage" could be designed to recover lost capacitance using



supervisory circuits[50], although most leakage is lost as heat.

Slide from a 2015 Presentation (Source: Slideshare[34])

If that $E \sim V^2$ resembles Einstein's relativity equation $E = mc^2$, that's because Ohm's law is as important as that in Electricity[68], and follows analogous special relativity variances for non-inertial frames of reference[28,29]. Rewritten as a measure of resistance, V^2=IR, which also means that that $\sqrt{IR} = V$ [70]. Amperage decrease exponentially when resistance remains the same and voltage decreases linearly. The trade-off allows, for example, 1GHz microprocessors to be underclocked at 1/5th of the speed but achieve more than 10x the energy savings (sometimes 13x). In other words, more performance for a non-linear (exponential) reduction in power. This also means that as amperage approaches a zero asymptote, resistance increases exponentially, and the Power Delay-Product at a given voltage (e.g 200mV), would operate instructions far less efficiently (235) than at a PDP at 0.55V (e.g. 7.2). Some concepts: "Power increases as the square of the voltage with P = IV" [69,71]

“The power by definition* is the product of the current and the voltage across a one-port (a two-terminal device). The power is only proportional to the square of the voltage if the \$I(V)\$ relationship is linear.”[67] “you need to understand that this law is not a linear equation, it's rather an instantons fact that stands, meaning at every differential time step, the product of resistance and current would give you the Voltage.”[67]
By determining what a given circuit's threshold voltage should be, the voltage rail can be allowed to decay without providing conventional state retention:

“Drowsy logic, an alternative to putting the processor into sleep mode, is applicable where leakage currents are managed. Muller said that it is possible to turn off the power and let the power rail voltage decay. But it is not necessary to provide explicit state retention as long as the voltages on flipflops do not go too low. Essentially, it is a method of intermittently providing voltage sufficient to maintain state and operations.”[35]

In *Computer Architecture Techniques for Power-Efficiency*, by Stefanos Kaxiras and Margaret Martonosi (2008), Chapter 5.3 confirms that drowsy circuits are a new class of “*state-preserving*” leakage reduction techniques.[60]

In *Recent Progress in Boolean Logic* (2013), Bernd Steinbach similarly describes sub-threshold logic: “In this case, voltage scaling may be used quite aggressively, i.e., to a point where the supply voltage is so low that transistors are never really ON.” [61]

How do these sub-threshold transistors operate, if they are not really “on”? Electrons propagate in one direction alongside a conductor in an electromagnetic (EM) wave once current is applied to a wire.[63] The “state-preserving” aspect of this propagation apparently would not immediately get shutdown or revert to drift velocity[64], as long as the voltage is flipped on again before the current runs out.

The quadratic and exponential decrease in power below sub-threshold voltage is less immediately useful for higher performance microprocessors, at least without significant code/instruction optimization and parallelization. The number of microcontrollers that use just a few Megahertz can achieve the biggest energy consumption gains in non time sensitive applications. That too, reflects on the similarity of $E \sim V^2$ to the Special Relativity equation.

By applying this concept, another analogy can be made: If a 2nd Earth were positioned in the same orbit as Earth, but at the March Equinox while the Earth was a the Summer Solstice, perpendicular to the axis that it intersects with the sun, it would have a clear view of the Earth in the dawn sky. Suppose a light ray left the Sun and, was somehow tagged with a very bright and cartoonishly large fluorescent arrow. A person on the 2nd earth could use a telescope and track the movement of this arrow pointing to that single light ray. The earth is 93 million miles away from the Sun. Einstein's 2nd postulate in Special Relativity states:

“The speed of light in vacuum is the same for all observers, regardless of the motion of light source or observer.”[64] It takes 8 and 1/3 minutes for a light ray to reach (both) Earths. Thus, from a distance of 93 million miles, even the speed of light might appear to be traveling slowly. A 5 Gigahertz processor today might sound much faster than a 50Mhz processor from 1994. However, the speed of individual electrons in both circuits travel at the same speed, like a feather and a hammer being dropped on the moon with no atmosphere. Less throughput (weight/cargo~bytes), but no difference in wind resistance, as there is none. To the telescope observer, the light ray is traveling at the same speed for both observers on Earth and the 2nd Earth.

Hypothetically, if the observer were the size of the Magic School Bus and millimeters away from the ray's trajectory, the electron would pass the observer far too quickly to be observed. Returning to the Mariana Trench analogy, the child aboard the boat knows when the light switch is being turned off and on, since he/she is the analog control of the switch. But he/she/they cannot see the effects on the circuit that the rest of the class is observing (assuming with a multimeter or microscope that can capture moving electrons).

Thus, the light switch (DVFS) in sub-threshold does not have the same effect as on above threshold/conventional circuits, because due to impedance and threshold requirements to maintain the circuit, the circuit in effect operates with a small variation in frequency, or could even be set to a fixed clock speed. Designing a microprocessor with “complexity-effective” IPC[53] throughput is the next challenge of the future power efficient designs.

Power First already using Drowsy Logic?

Power-efficiency has been an increasing design consideration in virtually all new silicon in the past 15 years. Power-first[1] designs, however, typically appear only in niche applications such as IoT. A recent paper describing a researcher's 2002 circuit, called drowsy logic, reviews[2] historical strategies to limit leakage in the context of foundries' recent implementations of low-leakage FinFET and Gate-All-Around technologies.

The original 2002 technique[3] involves a strategy, called the “simple” policy of placing all lines in a drowsy mode using a single global counter, awakening only when it is accessed. The performance trade off was known to reduce leakage up to 85% while increasing run-time by just 0.62% in certain conditions (using 93% drowsy lines). The paper focused on advanced drowsy strategies in reducing latency due to L1's time-critical cache, but suggested L2 strategies could use the simpler techniques.

A 2008 paper[7] titled “BTB Access Filtering: A Low Energy and High Performance Design” describes lowering branch target buffers and using direct-mapped BTBs in superscalar processors with drowsy techniques in the filter buffer to limit predictor energy consumption by 92.7% with up to a 10.8% performance trade-off. Early ARM processors[8], such as ARM7 v3 and ARM9 v4, did not use superscalar architecture and would likely not need large buffers. Static branch prediction was used, however, in power-conscious processors such as the ARM810[9,10].

Direct-mapping of hardware registers appears to have a similarity to MMU-less operating systems, such as μClinux[11], developed by Jeff Dionne and Ken Albanowski in 1998, which used flat memory addresses and was further developed by companies such as EmCraft[12].

Direct mapping of both hardware and kernel resources offers a path to limit energy-intensive memory management caches. While a chip can be designed to operate in a more deterministic manner, such as in real-time operating systems, use cases involving known application resource limits can be increasingly factored into software-defined hardware (SDH[13]), a decades old design goal. One design concept that monolithic chips use is “holistic timing.”[14] This is where multiple systems fit on a single die and operate within a certain window, including breaking partitions in the clock.

A Power-First design would most certainly benefit from opting towards fewer Die-to-die interfaces, which typically increase power consumption. An advantage to using chiplets, such as Bunch of Wires, however, would be the parallel/co-design and integration of peripherals such as lower power radios and I/O by third parties. While sub and near-threshold processors are a major focus of power reduction, memory can occupy 2-8x[15] the size of a single microcontroller such as Zero-riscy (now Ibex) or RI5CY (cv32e40p), according to PULP Platform of ETH Zürich. General purpose operating systems require far more memory and cache than microcontrollers. Efficient design hardware and software can lessen the amount of memory (and thus power) needed.

In software, Cortex-M processors feature interrupt/exception handling, where instead of automatically putting floating point registers onto the stack, can be configured to do so in a “lazy” way[4,5,6]. While most processors are designed to be fast, or to reduce latency, they also benefit from using tricks like that to make them less resource intensive.

Compared to Vdd gating, Cache Decay, and Adaptive Mode-Control

The 2002 paper and the 2023 retrospective paper contrasts drowsy logic with three previous techniques to reduce leakage developed in 2000 and 2001. The first, called *gated-Vdd*[16], uses a circuit-level technique to gate

supply voltage to reduce leakage in unused SRAM. That technique, paired with a novel resizable cache architecture (DRI i-cache), is said to reduce leakage by 62% with a minimal impact on performance. The second, Cache Decay, uses a time-based strategy to turn off a cache line after a pre-set number of cycles have elapsed since its last access[17]. This method is said to reduce L1 leakage up to 70% using competitive algorithms. The third, Adaptive Mode Control (AMC), uses tags that are always active, tracks which cache lines are missed, and can adjust the number of intervals to turn off cache lines based on previous misses[18]. Cache decay was also recently reviewed by their authors in the ISCA@50 retrospective[19].

Variation in process can be attributed to environmental factors such as temperature, but techniques to adapt voltage to temperature formed one of the first strategies hybridize drowsy and cache decay[60]. A significant amount of variation can also be found in the instruction cache: “This is because instructions exhibit strong temporal and spacial locality and any delay in fetch shows up immediately on performance.”[60]

The aforementioned research in BTBs[7] is also mentioned, using the hybrid approach. An advantage of microcontrollers having an immediate application for simple drowsy policies is that most of their operations use real-time operating systems with known scheduling, which can optimize the instruction cache and IPC buffer.

The 2008 text interestingly cites a 2000 paper by Zyuban and Kogge to describe power-inefficiency in out-of-order architectures (4.5.1)[60, 73], but also describes some of the innovative approaches to resource partitioning (4.5.2) and examines some of the disadvantages of earlier research in filter caches (4.10.2). as concurrently researched by Ziavras et al[7].

An avenue of research that could be worth exploring more are the “idle” cycles of instructions.[66] While modern operating systems have developed advances in idle-task scheduling for background processes such as SSD garbage collection, anti-virus scans, often using an idle core(s), in low-power microcontrollers and microprocessors, these processes, if used at all, would constitute a larger fraction of processors’ clock cycles. Nonetheless, the co-development of software and hardware from the ground up remains an important consideration to optimize towards the most energy and power efficiency.[54,56]

Compared to fast-to-wake, fast-to-compute, fast-to-sleep

While drowsy modes have been developed for both instruction and data cache, in high-performance computing (HPC), it may not yet have some of the advantages of state-of-the art memory such as spin-transfer torque (STT) MRAM and spin-orbital torque (SOT) MRAM[29]. That is because they are designed to operate in low-data modes where speed, including fast-to-compute and fast-to-sleep is not as crucial to operation in applications such as remote sensors with fixed interval telemetry. That said, the fast-wake up is known to be 1-2 cycles[32].

Drowsy logic is renamed to...dynamic voltage & frequency scaling?

A review in the ICSA@50 retrospective paper published in June of 2023 implemented drowsy logic into “a form of voltage scaling”:
“We proposed a design in which one can choose between two different supply voltages in each cache line, corresponding to normal supply voltage and a drowsy lower voltage. In effect we used a form of voltage scaling to reduce static power consumption. Due to short-channel effects in deep-sub micron processes, leakage current reduces significantly with voltage scaling.”[2]

By 2008, the term drowsy had already been used in textbooks as form of voltage scaling, but it is usually paired with other techniques such as Cache decay to form DVFS. Even the 2002 paper mentions it, but its novel feature at the time was that it was applied to static power:

“Such a dynamic voltage scaling or selection (DVS) technique has been used in the past to trade off dynamic power consumption and performance [6][7][8]. In this case, however, we exploit voltage scaling to reduce static power consumption.” (p.2/10)

New techniques utilizing hybrid or novel implementations of drowsy transistors in SRAM continue to be researched in academic labs.[20, 21,78]

TABLE 3. Comparison of various low-leakage circuit techniques

	Advantages	Disadvantages	Leakage power in low power mode
DVS	• Retains cell information in low-power mode. • Fast switching between power modes. • Easy implementation. • More power reduction than ABB-MTCMOS.	• Process variation dependent. • More SEU noise susceptible.	6.24nW
ABB-MTCMOS	• Retains cell information in low-power mode.	• Higher leakage power. • Slower switching between power modes.	13.20nW
Gated-V_{DD}	• Largest power reduction. • Fast switching between power modes. • Easy implementation.	• Loses cell information in low-power mode.	0.02nW

Techniques sometimes have multiple industry names. As early as 1995, multi and dual-threshold voltage techniques have been described[22, 23].

The 2002 paper also cites a 1994 PARC paper, “Scheduling for Reduced CPU Energy,” which also pioneered the concept of MIPJ (or Millions-of-instructions-per-joule) and describes software-based scheduling techniques that allow more granular control of the system clock speed by the operating system scheduler[53]. The PARC paper cites an early wave of low-power design developed in the late 1980s as the first Notebooks and PDAs were commercialized[54-59].

A difference that remains is that drowsy logic offers some of the best energy savings: “Moreover, since the penalty for waking up a drowsy line is relatively small (it requires little energy and only 1 or 2 cycles) and there are less frequent accesses to the lower memory hierarchy than Gated-V_{DD} schemes, cache lines can be put into drowsy mode more aggressively to save more power:”[2,3(Table 3)]

The theoretical minimum leakage should not be confused with the theoretical minimum of energy computation. In *Recent Progress in the Boolean Domain* (2013), Bernd Steinbach cites how the Landauer bound described zero-energy as requiring a reversible process, and that “any bit that is destroyed incurs an energy cost of $k_BTln(2)$. We also reported a strikingly similar result: that the lowest supply voltage at which a logic gate can operate while still acting as an amplifer is only a few times larger than k_BT /q.” [61] Granted, the sub-threshold frontier is much like the border between math and physics.

Industry news articles appear to have decreased mentions of “drowsy” circuits in the mid 2010s. A keyword search of one well-known site turned up only two mentions in 2014 and 2015 for “drowsy”[24] “For memories, people are building additional operational modes for them, such as drowsy modes.”[25] Furthermore, an aforementioned 85% reduction in leakage power can be found in IoT devices which are designed for batteryless operation and energy-harvesting. Microcontrollers by companies such as Ambiq Micro are known to achieve a 13-fold reduction compared to other chips[26].

Analog IoT device-makers such as ONiO feature an integrated solar/RF/thermoelectric harvester and power management integrated circuit with low-power, asynchronous ROM/RAM[27]. Since the latter uses just 2KB of RAM, it may not need to operate in sub-threshold mode to create “digital zeros and 1s,” as much as Ambiq's 2MB SRAM cells to achieve ultra low energy consumption[30]. Nonetheless, the ultra low power of advanced power saving techniques such as sub-threshold voltage, hybrid drowsy cache and memory suggests mobile phones could one day run on solar power.

Footnotes

[1] https://semiengineering.com/a-power-first-approach/

[2] “Drowsy Caches: Simple Techniques for Reducing Leakage Energy—A Retrospective” Krisztián Flautner, Nam Sung Kim, Steve Martin, David Blaauw, Trevor Mudge; ISCA@50 25-Year Retrospective: 1996-2020 https://sites.coecis.cornell.edu/isca50retrospective/papers/ https://bpb-us-w2.wpmucdn.com/sites.coecis.cornell.edu/dist/7/587/files/2023/07/drowsy_retro.pdf

[3] “Drowsy Caches: Simple Techniques for Reducing Leakage Power” Krisztián Flautner, Nam Sung Kim, Steve Martin, David Blaauw, Trevor Mudge (2002) https://web.eecs.umich.edu/~manowar/publications/drowsy-caches-ISCA2002.pdf

[4] https://developer.arm.com/documentation/dai0298/a

[5] https://docs.zephyrproject.org/latest/hardware/arch/arm_cortex_m.html

[6] https://docs.zephyrproject.org/latest/kernel/services/other/float.html

[7] “BTB Access Filtering: A Low Energy and High Performance Design” Shuai Wang, Jie Hu, and Sotirios G. Ziavras Department of Electrical and Computer Engineering New Jersey Institute of Technology (2008) https://web.archive.org/web/20090920084956id_/http://web.njit.edu:80/~sw63/pub/ISVLSI_BAF_2008.pdf
[8] https://en.wikipedia.org/wiki/List_of_ARM_processors
[9]https://web.archive.org/web/20130926155924/http://www.eetimes.com/document.asp?doc_id=1208831 “VLSI Technology Now Shipping ARM810” 08/1996
[10]https://web.archive.org/web/20181224080542/https://www.hotchips.org/wp-content/uploads/hc_archives/hc08/2_Mon/HC8.S4/HC8.4.1.pdf “ARM810: Dancing to the Beat of a Different Drum” 07/1996
[11] https://en.wikipedia.org/wiki/%CE%9CClinux
[12] https://www.emcraft.com/
[13] https://semiengineering.com/why-chiplets-dont-work-for-all-designs/ “Why Chiplets Don’t Work For All Designs” 09/2023
[14] https://semiengineering.com/software-defined-hardware-architectures/ “Software-Defined Hardware Architectures” 05/2023
[15] https://pulp-platform.org/community/showthread.php?tid=229&pid=650#pid650 PULP Platform, forum post, 2021
[16] ˝Gated-Vdd: A Circuit Technique to Reduce Leakage in Deep-Submicron Cache Memories,” 2000 Michael Powell, Se-Hyun Yang, Babak Falsafi, Kaushik Roy, and T. N. Vijaykumar https://engineering.purdue.edu/~vijay/papers/2000/gatedvdd.pdf
[17] ˝Cache Decay: Exploiting Generational Behavior to Reduce Cache Leakage Power” 2001 Stefanos Kaxiras, Zhigang Hu, Margaret Martonosi https://mrmgroup.cs.princeton.edu/papers/hzg-isca2001.pdf
[18] ˝Adaptive Mode Control:A Static-Power-Efficient Cache Design” 2001 https://prod.tinker.cc.gatech.edu/journal/zhou03adaptive.pdf https://pdfs.semanticscholar.org/ba53/2536b7d1890f967d1aac7485544f6abebe39.pdf
[19] “RETROSPECTIVE: Cache Decay: Exploiting Generational Behavior to Reduce Cache Leakage Power” 06/2023 https://bpb-us-w2.wpmucdn.com/sites.coecis.cornell.edu/dist/7/587/files/2023/06/Hu_2001_Cache.pdf
[20] Design and Implementation of Low Power SRAM Using Highly Effective Lever Shifters er Shifters, 2021 https://scholar.uwindsor.ca/cgi/viewcontent.cgi?article=9892&context=etd
[21] ˝Hybrid Drowsy SRAM and STT-RAM Buffer Designs for Dark-Silicon-Aware NoC, 2016 Jia Zhan, Student Member, IEEE, Jin Ouyang, Member, IEEE, Fen Ge, Member, IEEE, Jishen Zhao, Member, IEEE, and Yuan Xie, Fellow, IEEE https://cseweb.ucsd.edu/~jzhao/files/darlsilicon-noc-tvlsi2016.pdf
[22] “1 -V Power Supply High-speed Digital Circuit Technology with Multithreshold-Voltage CMOS” 1995 https://ieeexplore.ieee.org/document/400426 http://www.facweb.iitkgp.ac.in/~apal/LPC_2010/MTCMOS.pdf
[23] ˝Design and Optimization of Low Voltage High P erformance Dual Threshold CMOS Circuits” Liqiong W ei, Zhanping Chen, Mark Johnson, Kaushik Roy & Vivek De, 1998 https://dl.acm.org/doi/pdf/10.1145/277044.277179
[24] “S-L Power Modeling Gains Steam” 08/2014 https://semiengineering.com/system-level-power-modeling-activities-get-rolling/
[25] “With Responsibility Comes Power,” 02/2015 https://semiengineering.com/with-responsibility-comes-power/
[26] “What’s All This Subthreshold Stuff, Anyhow?” 02/2019 https://www.electronicdesign.com/technologies/analog/article/21807652/whats-all-this-subthreshold-stuff-anyhow
[27] “What if You Never Had to Charge Your Gadgets Again?” 01/2024 https://www.wsj.com/tech/personal-tech/what-if-you-never-had-to-charge-your-gadgets-again-955ea960
[28] https://www.fluke.com/en-us/learn/blog/electrical/what-is-ohms-law
[29] https://en.wikipedia.org/wiki/Special_relativity
[30] “Interview With Scott Hanson - Founder and CTO at Ambiq” 01/2024 https://www.safetydetectives.com/blog/scott-hanson-ambiq/
[31] “TSMC tandem builds exotic new MRAM-based memory with radically lower latency and power consumption” 01/2024 https://www.tomshardware.com/pc-components/dram/tsmc-tandem-builds-exotic-new-memory-with-radically-lower-latency-and-power-consumption-mram-based-memory-can-also-conduct-its-own-compute-operations
[32] “Circuit and microarchitectural techniques for processor on-chip cache leakage power reduction” (Dissertation), 2004 Nam Sung Kim, Trevor N. Mudge https://dl.acm.org/doi/10.5555/1023306
[33]https://www.eng.auburn.edu/~agrawvd/THESIS/KIM_S/S.Kim_HonorsThesis_FINAL.pdf May, 2016 Sungil Kim
[34] https://www.slideshare.net/AmbiqMicro/apollo-ultra-low-power-mcu-from-ambiq-micro
[35] https://www.eetimes.com/arm-preps-near-threshold-processor-for-iot/
[36] https://en.wikipedia.org/wiki/Soft_error#Other_causes https://en.wikipedia.org/wiki/Crosstalk#Other_examples
[37]https://www.tsmc.com/english/dedicatedFoundry/technology/logic/l_22nm
[38] https://semiengineering.com/near-threshold-computing-gets-a-boost/
[39] https://en.wikipedia.org/wiki/Geode_(processor)#Geode_GXm
[40] https://www.wtamu.edu/~cbaird/sq/2014/02/19/what-is-the-speed-of-electricity/
[41] https://electronics.stackexchange.com/questions/464689/what-is-the-speed-of-electricity
[42] “What are valence orbitals?” https://www.lanl.gov/orgs/nmt/nmtdo/AQarchive/04spring/VO.html
[44] “Probabilistic Processors” https://www.cs.uaf.edu/2011/spring/cs641/proj1/rarutter/
[45] https://en.wikipedia.org/wiki/Centimetre%E2%80%93gram%E2%80%93second_system_of_units#Advantages_and_disadvantages
[46] https://www.scientificamerican.com/article/see-the-highest-resolution-atomic-image-ever-captured/
[47] Dynamic voltage frequency scaling (DVFS) for microprocessors power and energy reduction December 2005 https://www.emo.org.tr/ekler/035226640b6b89f_ek.pdf
[48] “Battery-Free Game Boy” 09/2020 https://dl.acm.org/doi/pdf/10.1145/3411839
[49] “Dynamic Voltage and Frequency Scaling for Intermittent Computing” 01/2024 https://arxiv.org/pdf/2401.08710.pdf
[50]https://en.wikipedia.org/wiki/Supervisory_circuit#Overvoltage_and_undervoltage_protection
[51] https://en.wikipedia.org/wiki/Waveform
[52] https://en.wikipedia.org/wiki/Continuous_function
[53] RETROSPECTIVE: “Complexity-Effective Superscalar Processors”, Subbarao Palacharla, Norman P. Jouppi, J. E. ICSA@50, Smith3 06/2023 https://bpb-us-w2.wpmucdn.com/sites.coecis.cornell.edu/dist/7/587/files/2023/06/Palacharla_1997_Complexity.pdf
[54] “Scheduling for Reduced CPU Energy” Mark Weiser, Brent Welch, Alan Demers, Scott Shenker, Proceedings of the First USENIX Symposium on Operating Systems Design and Implementation, November 1994 https://www1.icsi.berkeley.edu/pubs/networking/ICSI_schedulingforreduced94.pdf
[55] “A Scheduling Model for Reduced CPU Energy” Frances Yao Alan Demers Scott Shenker”FOCS, 1995 https://graal.ens-lyon.fr/~lmarchal/scheduling/Yao-FOCS95-Energy.pdf
[56] A JSSC classic paper: Low-power CMOS digital design, 04/2003 https://chandrakasan.mit.edu/wp-content/uploads/2021/04/Chandrakasan_06499960.pdf
[57] “Low-power CMOS Digital Design” A. P. Chandrakasan, S. Sheng, and R. W. Brodersen , JSSC, 04/1992 https://mtlsites.mit.edu/researchgroups/icsystems/pubs/journals/1992_chandrakasan_jssc.pdf
[58] “A JSSC Classic Paper: The Simple Model of CMOS Drain Current” 10/2004 https://www.eng.auburn.edu/~agrawvd/COURSE/READING/LOWP/alpha_power_law.pdf
[59] “Alpha-power law MOSFET model and its applications to CMOS inverter delay and Other Formulas” 04/1990 https://courses.ece.ucsb.edu/ECE125/125_W11Banerjee/Lectures/SAK90a.pdf
[60] *Computer Architecture Techniques for Power-Efficiency*, Stefanos Kaxiras & Margaret Martonosi (Ch. 4.5-4.10),(Ch. 5.3) Morgan & Claypool Press 2008
[61] *Recent Progress in Boolean Logic*, Bernd Steinbach, 2013 (4.1.3, 203), (4.1.5, 211)
[62] "A study of DVFS methodologies for multicore systems with islanding feature *(p.42) Shervin Hajiamini, Behrooz A. Shirazi, in Advances in*

Computers, 2020

[63] https://electronics.stackexchange.com/questions/462524/propagation-speed-of-em-waves-in-air-versus-through-conductive-material/

[64] https://en.wikipedia.org/wiki/Speed_of_electricity#Electric_drift

[65] https://en.wikipedia.org/wiki/Vacuum

[66] https://www.ledyilighting.com/why-do-light-bulbs-glow-when-switched-off/

[67] “Idleness is not sloth” Richard Golding, Peter Bosch, Carl Staelin, Tim Sullivan, and John Wilkes https://dl.acm.org/doi/proceedings/10.5555/1267411 TCON'95: Proceedings of the USENIX 1995 Technical Conference Proceedings January 1995 https://john.e-wilkes.com/papers/idleness.pdf

[68] “Is power proportional to V or V^2?” https://electronics.stackexchange.com/questions/543610/is-power-proportional-to-v-or-v2

[69] Ohm's Law http://avstop.com/ac/apgeneral/ohm%27slaw.html

[70] https://www.malcolmchisholm.net/painless-decibels-almost

[71] https://science-education-research.com/when-is-vir-the-formula-for-ohms-law/

[72] https://www.physicsforums.com/threads/understanding-the-relationship-between-power-voltage-current-and-resistance.876880/

[73] "Optimization of high-performance superscalar architectures for energy efficiency" V. Zyuban, P. Kogge, 08/2000 https://dl.acm.org/doi/pdf/10.1145/344166.344522

[74]https://chem.libretexts.org/Bookshelves/Physical_and_Theoretical_Chemistry_Textbook_Maps/Supplemental_Modules_(Physical_and_Theoretical_Chemistry)/Quantum_Mechanics/03._The_Tools_of_Quantum_Mechanics/Collapsing_Wavefunctions

[75] https://en.wikipedia.org/wiki/Power%E2%80%93delay_product#cite_note-Gaudet_2014-1

[76] "energy autonomous computing" https://github.com/kragen/dernocua/blob/master/text/energy-autonomous-computing.md#estimating-the-necessary-performance-for-basic-interactive-computation-01-dmips

[77] "My visit to Bletchley Park and The UK Computing History Museum in November 2017” https://www.koomey.com/post/670564790101524480

[78] "Design the efficient SRAM circuit using 4transistor with sleepy logic" International Journal of Pure and Applied Mathematics Volume 118 No. 20 2018, 115-123 https://acadpubl.eu/hub/2018-118-21/articles/21b/15.pdf