HACKADAY.IO

[Edit project details]

Project Details

For The Open Source Autarkic Motherboard

Seeing shall be believing: A solar-powered, Raspberry Pi-like Board Design project that runs on 5mW

Giovanni • 02/15/2021 at 00:35

12-30-2023

New Name idea: Project Sorites? Or, Plato and the Sorites Paradox.

https://en.wikipedia.org/wiki/Sorites_paradox I've liked this term for building PCs, and my solar motherboard project, because, a lot of the abstract brainstorming in defining a standard, is very much dependent on vague, open source standards which may or may not need to be adopted. Thus, a Lego-like system with modular and interoperable parts, resembles the slow disappearance of a form factor/laptop as it is slowly disassembled and reassembled, each part like a grain of sand.

"A typical formulation involves a heap of sand, from which grains are removed individually. With the assumption that removing a single grain does not cause a heap to become a non-heap, the paradox is to consider what happens when the process is repeated enough times that only one grain remains: is it still a heap? If not, when did it change from a heap to a non-heap?[3]

Paradox of the heap

The word *sorites* (*Greek*: ***σωρείτης***) derives from the Greek word for 'heap' (*Greek*: ***σωρός***).[4] The paradox is so named because of its original characterization, attributed to Eubulides of Miletus.[5] The paradox is as follows: consider a heap of sand from which grains are removed individually. One might construct the argument, using premises, as follows:[3]

1,000,000 grains of sand is a heap of sand (Premise 1)*A heap of sand minus one grain is still a heap.* (Premise 2)

Repeated applications of Premise 2 (each time starting with one fewer grain) eventually forces one to accept the conclusion that a heap may be composed of just one grain of sand.[6] Read (1995) observes that "the argument is itself a heap, or sorites, of steps of *modus ponens*":[7]

1,000,000 grains is a heap.If 1,000,000 grains is a heap then 999,999 grains is a heap.So 999,999 grains is a heap.If 999,999 grains is a heap then 999,998 grains is a heap.So 999,998 grains is a heap.If ...... So 1 grain is a heap."

So is this a laptop, phone or tablet?

Well, if you have carefully analyzed the Sorites Paradox, it can be any of the three, and you are the decider of what defines the "completed" assembly of the mobile device. There is not a wrong answer, since one person may use the a different combination of standards to build a different system using some of the building blocks.

Also see Andreas Erikksen's prototype for a simpler (and functional) demo: https://hackaday.io/project/184340-potatop :)

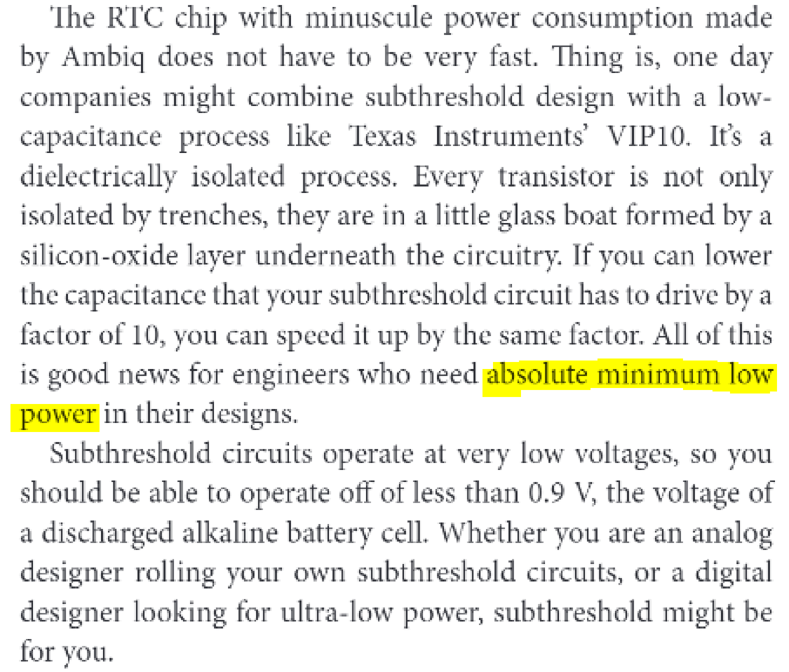
Update 10/22/2023: See post on laptop design could benefit from Plato's Ideal World of 3rd Party components: https://hackaday.io/page/21232-platos-ideal-world

Update 9/28/2023: https://www.raspberrypi.com/news/introducing-raspberry-pi-5/ It cost $25 million to build the Raspberry Pi 5, plus $15 million to develop the RP1 I/O controller. "Under development since 2016, RP1 is by a good margin the longest-running, most complex, and (at $15 million) most expensive program we've ever undertaken here at Raspberry Pi."

And, "However, the much higher performance ceiling means that for the most intensive workloads, and in particular for pathological "power virus" workloads, peak power consumption increases to around 12W, versus 8W for Raspberry Pi 4."

The first Raspberry Pi, Model B, released in 2012, used between 0.9W and 1.1W. The Model A and Pi Zero used as little as 0.4W. The Rpi 5 watt ceiling consumption has crossed into the double digits.

With no indication that the Raspberry Pi seeks to develop a solar powered single-board computer anytime soon, this project can serve as that low power alternative, in the absolutely lowest possible terms:



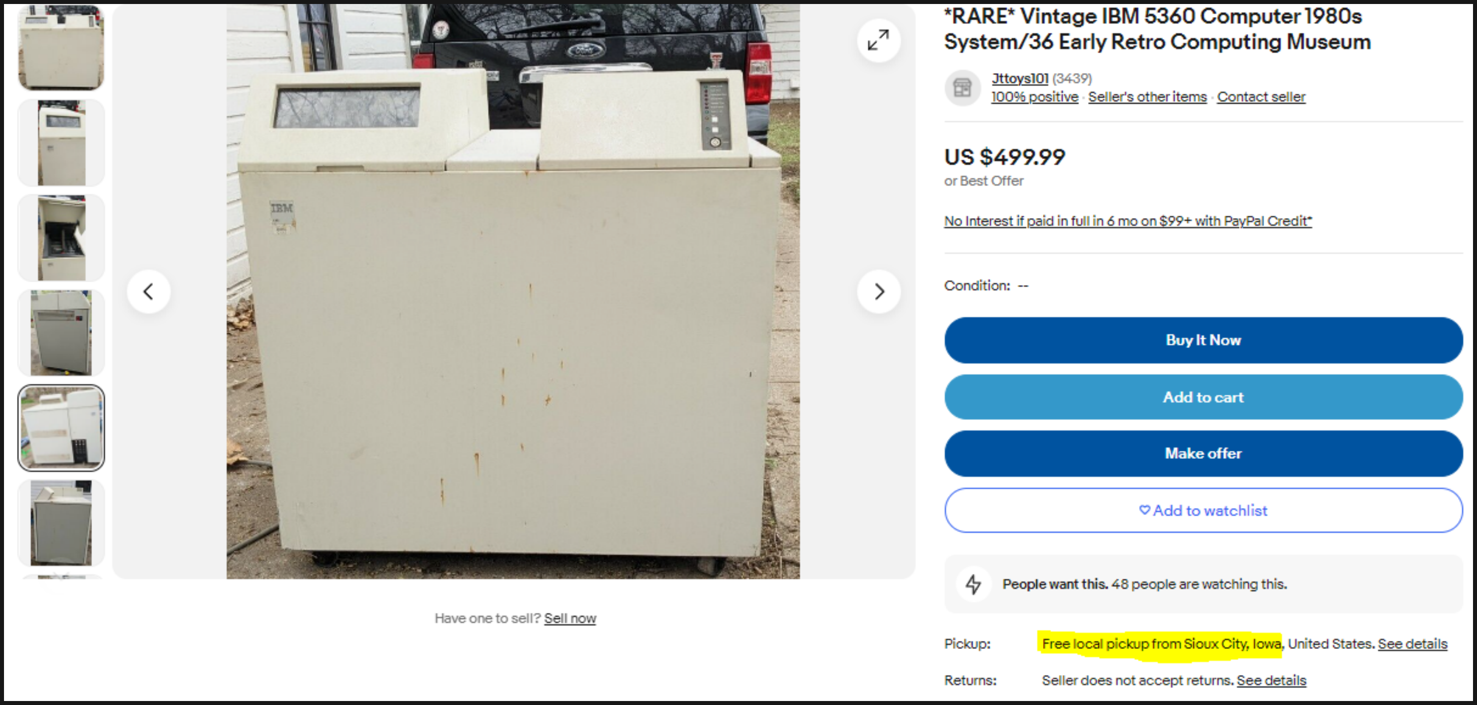
The RTC chip with minuscule power consumption made by Ambiq does not have to be very fast. Thing is, one day companies might combine subthreshold design with a low-capacitance process like Texas Instruments' VIP10. It's a dielectrically isolated process. Every transistor is not only isolated by trenches, they are in a little glass boat formed by a silicon-oxide layer underneath the circuitry. If you can lower the capacitance that your subthreshold circuit has to drive by a factor of 10, you can speed it up by the same factor. All of this is good news for engineers who need absolute minimum low power in their designs.

Subthreshold circuits operate at very low voltages, so you should be able to operate off of less than 0.9 V, the voltage of a discharged alkaline battery cell. Whether you are an analog designer rolling your own subthreshold circuits, or a digital designer looking for ultra-low power, subthreshold might be for you.

To give you an idea how much this would cost, add $25 million (which includes $7-12 million alone for the CPU) to display integration- be it E-ink, Ynvisible (formerly Rdot Displays) or Azumo, then include low-power wireless bandwidth by telecoms interested in supporting NB-IoT LTE, Cat M2, NB2 & DSSS modulation, and the cost of this project exceeds $50 million. Still interested? Most venture startups don't have the funds for that kind of research and development. Save for a few, but it is being invested in AI, or something else. Can one put a price on connectivity? A product like this could connect 2 billion people who do not have any internet access, in regions with no electricity. It could also be a useful emergency cell phone in places where power outages hit utilities. Portable Bentocells could be a practical complement.

Another upside is, this wouldn't be an investment that requires redesigning the entire board for subsequent generations. The Raspberry Pi moved their Ethernet port back to the bottom right. Perhaps this could be a nod to wanting to keep a defacto standard to limit the disposal of older cases so that they can be re-used in newer systems. So designing an ATX standard, even for tiny motherboards would set the standard for decades of phones, as there is no indication anyone wants to start using Google or Apple AI glasses for their communications anytime soon. The ATX standard saved Intel money, but it also saved consumers money too. It just so happened to be a *de facto* standard that no one minded adopting. It is a standard that hasn't gone away for almost 30 years. Are phones really more personable that they need to be more special (at the expense of non-reusable outer shells), or is it that smartphone marketing is just really good at selling us different phones that make us feel like we have many personable options? Visit distrowatch.com and you can see thousands of operating systems, yet they all work on a handful of architectures.

What this era could use is a modular platform for upgradeable mobile hardware.



RARE Vintage IBM 5360 Computer 1980s System/36 Early Retro Computing Museum

Jttoys101 (3439)
100% positive · Seller's other items · Contact seller

US $499.99
or Best Offer

No Interest if paid in full in 6 mo on $99+ with PayPal Credit*

Condition: --

People want this. 48 people are watching this.

Pickup: Free local pickup from Sioux City, Iowa, United States. See details

Returns: Seller does not accept returns. See details

When a PC outlives its usefulness, it gets tossed to the side of a curb.

There is a reason certain products can't offer a shipping option. Obsolete hardware, back in the day, was rare/specific enough that modularity was not a practical idea- there was no need to standardize form factors because the next generation was certain to have a radically different operating system (PDP-11 vs. GE 645) and more manufacturable, and smaller components. Phones cannot get any smaller than a flip phone, because the human interaction with the inputs reaches a physical limit with vision (font size) and tactile interaction. Therefore miniaturization of *user-interface,* not transistors, has reached a plateau.

""As far as machines are concerned, machines may be classified into two general types- those with dimensions determined by their function and those whose size is limited only by economic factors"

-Kenneth Shoulders, 1961 in "Microelectronics Using Electron-Beam-Activated Machining Techniques"

Design of standardization can be adopted. The main reason for rejecting standardization is that it creates the impression of freedom. But freedom itself is somewhat subjective. Freedom to exclude interoperability, or freedom to save money? The value of a unique design certainly may increase compared to a generic, mass produced one, but at the same time, a mass produced one will have a lasting value, which can lead to more freedom of use and utility. Is aesthetic priceless? Is the material representation of a design more important than a digital rendering? Especially in an era of abundant digital content, material goods are far scarcer.

The Open Source Autarkic Motherboard project is a concept seeking an ISO standard and a network effect: https://en.wikipedia.org/wiki/Network_effect (funding would also help!). This is a placeholder for a concept. I am not funded by any of the products that I promote here. They are listed here as concepts. If you want to make this product as inexpensive as possible, create a community project and agree upon a standard that meets as many people's criteria as possible. If you want to make this product as expensive as possible, develop it in a "limited edition" and make Gen 2.0 not backwards compatible. The only amount of influence I have in this project is keeping the idea alive. Once any number of developers agree on the concept, it is no longer in my control. You become the architect, and is it is forever in your hands. Be careful what you wish for! :) But if you want to hire me, I'm happy to be the conceptual architect/project manager. I don't have a degree from Parsons School of Design, but Steve Jobs was a dropout after 1 semester- and I managed eight semesters. I recommend keeping the kernel engineers and the hardware architects somewhat separated in the early stages, as one is most likely going to have opposing principles (maximalism vs minimalism, when there is a specific formula involved related to market needs, not an engineer's utopia). It took a lot of conceit to create the Apple I, but it also took a lot of modesty. Think 1984 Macintosh with TCP/IP (which wasn't integrated until 1988) (also TCP/IP is just a concept- a placeholder for an internet protocol, thus it is not a preference for that over some other protocol-it's merely suggesting the device should be able to connect to the internet, rather than offline- thus you can choose every protocol- or have support for a number of protocols), rather than something like a modern kernel with millions of lines of code. A modern kernel is going to use more than 1 watt of power. So power consumption is the first thing you'd need to agree upon before working on your respective domains. 10mW max is probably my estimate of average consumption. In theory, if the ISO standard can be developed, then like the ATX motherboard, it can last 30 years+ without needing to change it. Which is why old, yellowing ATX cases from 1997 can be used with the newest AM4/Intel LGA sockets. If this standard can last even 10 years, it would be a success, since it would extend the time of a phone's lifespan.

A natural analogy to an autarkic system's "battery" could be like water is to a cactus: efficient at storing lots of water, and very resistant to allowing water to evaporate.

Update 8/31/2022: This project is more like a portable motherboard development project. By emphasizing less the exterior and more the interior, modular boards can become a core platform for phones, tablets, and laptops. In a way, by calling this project an open source autarkic laptop would be limiting it's applicability. While a phone may have less processing power than a laptop (not always), the concept is to design a board with the most scalability, without needing completely different drivers. Maximizing the number of modular adapters without causing overcomplexity is the goal here. A clear distinction should be made. Maximum scalability is **NOT** infinite scalability. To quote a physics academic: "Infinite growth isn’t possible, but superlinear for an extremely long time horizon definitely is."
https://twitter.com/lachlansneff/status/1552493268113321984

Update 8/30/2022

To make this project a little easier to follow, I'm splitting up development into several areas, as each component would require a different specialty, and any expertise is certainly welcome! Feel free to claim a section (or sections) of the board you want to work on. This is your project! I'm just a facilitator of ideas. You can also fork my github page if you'd like-there is no code- just ideas and links! If forking is an undesirable term, you can spoon it... elseware (not here!)
it: https://getpocket.com/explore/item/ten-surprising-facts-about-everyday-household-objects

In broad categories, the board wouldn't be a one-size fits all laptop, but it's intended to be somewhat utilitarian for text editing and displaying a terminal shell, at least initially, in that it seeks to find a commonality in laptop use-case that enough users would want to contribute to.

The board itself does not have a single component that must be used. It is 100% modular. That said, if a project were to gain traction, some amount of consensus would need to be made, so that the parts can "talk" to each other and not have power consumption incompatibilities.

I kind of see this project like an "All-Star Team" or "Supergroup" in that it's recruiting hackers, tinkerers, developers, (or a term you prefer to be called) for the:

Lowest power CPU- Ambiq Apollo 4

https://en.wikipedia.org/wiki/SuperH#J_Core (possibly on 22/28nm?)

Lowest Power Display- MIP (memory in pixel/e-ink)? TFDs use 3mW (1.8")

Lowest Power Keyboard: https://www.ti.com/lit/an/slaa139a/slaa139a.pdf?ts=1674338500549

https://www.ti.com/tool/TIDM-BATTERYLESSNFCKEYBOARD?HQS=ti-null-null-productcentre_refdes-manupromo-rd-ElectronicSpecifier-eu#overview https://www.ti.com/lit/ug/tidu398/tidu398.pdf

https://www.mouser.com/ProductDetail/ZF/AFIG-0007?qs=KnNyCueLKRVo%252BCLj1DZFyQ%3D%3D

Lowest Power Memory- (integrated/on chip as with 2MB Apollo4 or RPC-DRAM: https://iis-projects.ee.ethz.ch/index.php?title=An_RPC_DRAM_Implementation_for_Energy-Efficient_ASICs_(1-2S)

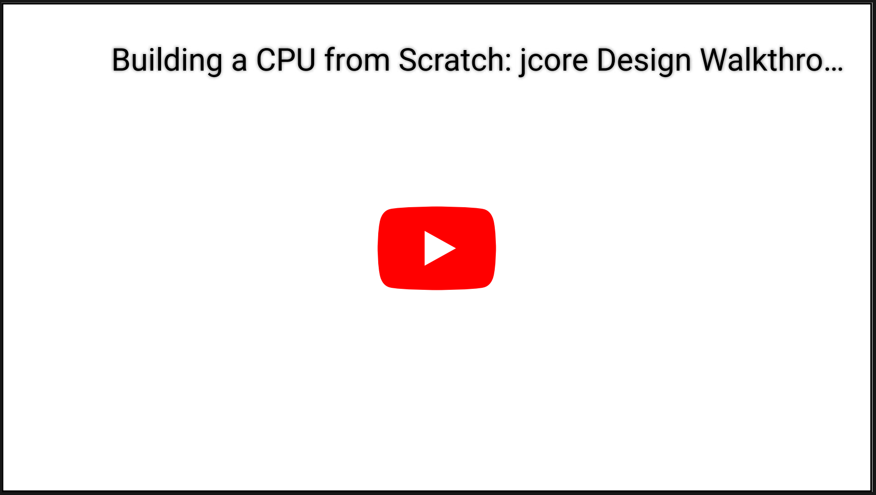
Lowest Power Operating System & Language- Assembly (such as Uxn Tal), C, or nesC, Symbian/EKA2kernel

https://en.wikipedia.org/wiki/SuperH#J_Core "

- Existing compiler and operating system support (Linux, Windows Embedded, QNX[11])"

https://j-core.org/

https://github.com/CoreSemi/jcore-jx



Building a CPU from Scratch: jcore Design Walkthro...

"The SuperH processor is a Japanese design developed by Hitachi in the late 1990's. As a second generation hybrid RISC design it was easier for compilers to generate good code for than earlier RISC chips, and it recaptured much of the code density of earlier CISC designs by using fixed length 16 bit instructions (with 32 bit register size and address space), using microcoding to allow some instructions to perform multiple clock cycles of work. (Earlier pure risc designs used one instruction per clock cycle even when that served no purpose but to make the code bigger and exhaust the encoding space.)

Hitachi developed 4 generations of SuperH. SH2 made it to the United states in the Sega Saturn game console, and SH4 powered the Sega Dreamcast. They were also widely used in areas outside the US cosumer market, such as the japanese automative industry.

But during the height of SuperH's development, the 1997 asian economic crisis caused Hitachi to tighten its belt, eventually partnering with Mitsubishi to spin off its microprocessor division into a new company called "Renesas". This new company did not inherit the Hitachi engineers who had designed SuperH, and Renesas' own attempts at further development on SuperH didn't even interest enough customers for the result to go ito production. Eventually Renesas moved on to new designs it had developed entirely in-house, and SuperH receded in importance to them... until the patents expired."

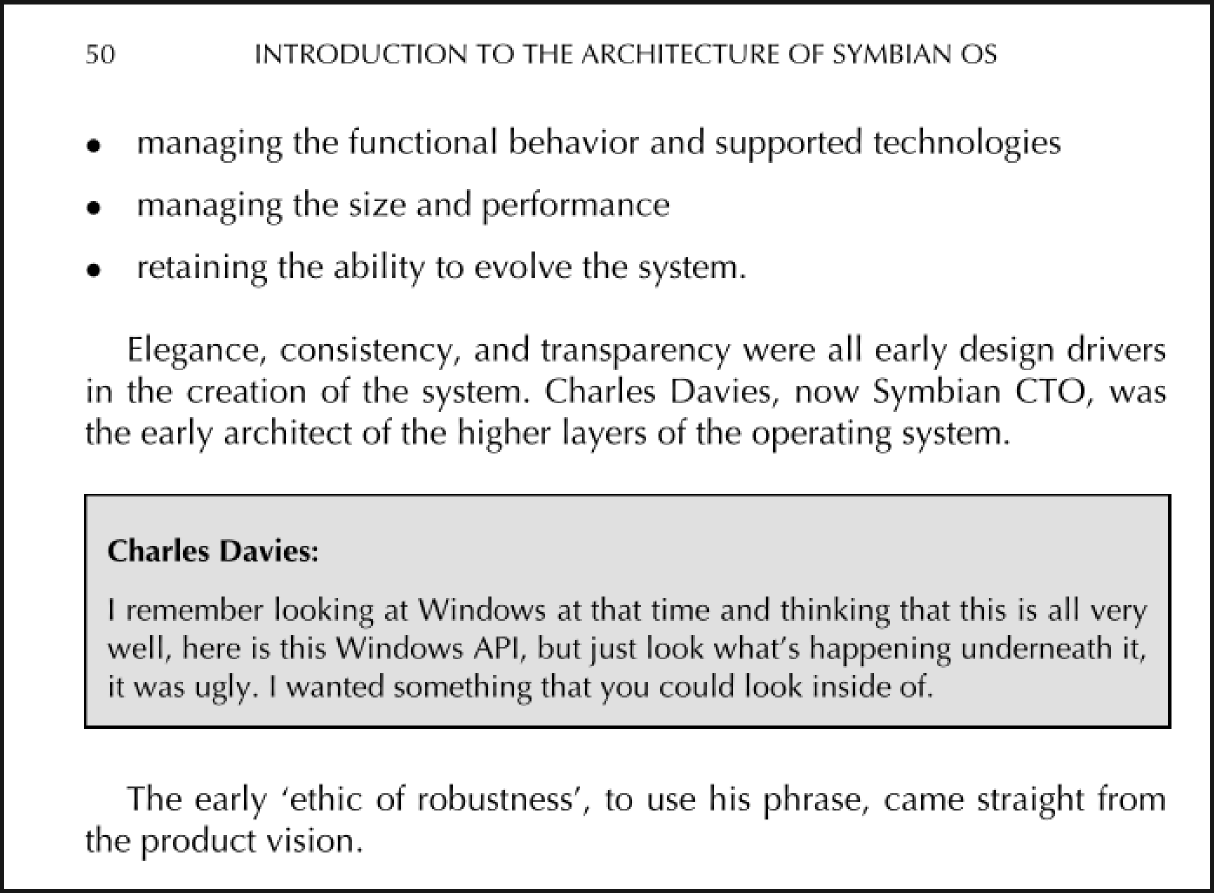
https://www.qnx.com/developers/docs/6.3.0SP3/neutrino/sys_arch/kernel.html

I highly recommend checking out:

"The Symbian OS Architecture Sourcebook: Design and Evolution of a Mobile Phone OS (Symbian Press) 1st Edition by Ben Morris"

https://www.amazon.com/Symbian-OS-Architecture-Sourcebook-Evolution/dp/0470018461/

Chapter 3 goes into the Philosophy of an OS Architecture, and why it matters.



50 INTRODUCTION TO THE ARCHITECTURE OF SYMBIAN OS

- managing the functional behavior and supported technologies
- managing the size and performance
- retaining the ability to evolve the system.

Elegance, consistency, and transparency were all early design drivers in the creation of the system. Charles Davies, now Symbian CTO, was the early architect of the higher layers of the operating system.

> **Charles Davies:**
>
> I remember looking at Windows at that time and thinking that this is all very well, here is this Windows API, but just look what's happening underneath it, it was ugly. I wanted something that you could look inside of.

The early 'ethic of robustness', to use his phrase, came straight from the product vision.

One idea, which I think could work https://github.com/hatonthecat/ENGAGE-GEOS copying the technique ENGAGE used: porting STM32F7 (3.84Mhz) NES emulator to CortexM4 Apollo3, by using another STMF4 C64 emulator (1Mhz), and porting it to Apollo3:

#C64 Emulator Implementation To port GEOS, a C64 emulator for ARM would need to include it. A C64 emulator has been ported to STM32F4: https://github.com/Staringlizard/memwa3 https://hackaday.com/2014/10/23/a-complete-c64-system-emulated-on-an-stm32/

Thus to load GEOS, one would need to include a method for loading GEOS onto the emulator. A useful app would be GeoWrite- a feature rich text editor: https://github.com/mist64/geowrite#description

https://github.com/vvaltchev/tilck

https://hackaday.com/2021/11/18/c-is-the-greenest-programming-language/

https://dl.acm.org/doi/epdf/10.1145/3274783.3274839

Lowest data Internet Protocol: Low data protocol: MQTT / LwIP

Lowest Power Keyboard & Trackpad/Mouse-

Lowest Power Wireless- https://arxiv.org/abs/1611.00096 (tunnel diode oscillator (TDO) or LoReA.

LTE IoT-NB w/ eDRX)

And a Battery/Supercapacitor/Li-Ion Capacitor Management to run it, powered by:

A Solar Panel that fits on the lid, around the display or on the back lid.

There are definitely more components to this, such as removable flash memory, USB, and the likes. But with each component claimed by the experts, the development of this project can progress in parallel. Perhaps someone with knowledge of each part who can designate a component or draw a schematic that is known to work with all the other components would certainly be helpful. However one is free to work on their choice for component and submit it as a candidate for the initial release. If there is more than one candidate, it would be voted on by all the contributing group members. If not, you're the first to join. So far, Andreas Eriksen has done some very interesting things with his PotatoP project and took some inspiration from this and I think it is very interesting see how ideas can freely permeate through a collection of similar projects. He also calculated the power consumption of the Artemis module (which has an Apollo3, and found it uses around 22mW (update 1/19/2023: 6.6mW! on typing and refreshing screen) when typing with constant refreshes-which is really great. Those are the kinds of discoveries I seek in this project).

I also discovered from his Andreas' links this website: http://www.technoblogy.com which has some really interesting portable devices with keyboards and displays, such as the LISP Badge: http://www.technoblogy.com/show?2AEE and some other boards like on r/Cyberdeck.

What I think is remarkable about these gadgets is how lightweight they are. Phones today have huge screens and weigh a lot, due to their batteries, yet a solar laptop could be developed that weighs less, with its main weight being plastic enclosure, rather than the battery. So one goal is that large laptops, with comfortable, full sized keyboards could weigh less than phones.

If a solar panel is able to successfully power an MCU, display, and keyboard 99% of the time under a single lightbulb*, for peace of mind, one might only need a 350mAh backup battery (or less) instead of 2000Mah battery that I've use to run a Raspberry Pi 3B+. The goal of this project is to select components that could one day be programmed to boot a general purpose OS. Testing keyboard outputs and programming is certainly an integral process, though. I've also recently found an amazing programmer who develops U-boot and hopefully uclinux can be ported. So your expertise is highly sought!

* Why a lightbulb and not the sun? Well, for one, I believe in setting an extremely high benchmark as a way to set a clear limit on power consumption, while also having a ubiqitous ability to charge in both indoor and outdoor environments. For example, let's say you're in a college auditorium with 120 seats and you're taking notes in class- but prefer to type them. Every seat doesn't have it's own outlet for charging, and you don't want to browse the web while you're supposed to be taking notes in class. A laptop with just note taking capabilities would actually be able to charge from the auditorium ceiling lights, much like a solar panel could power a calculator. At worst, it might not result in a net *increase* of charge while using it, but *slow* down the depletion of the battery if you take the solar laptop to class and forgot to charge it, but it has 5% battery. It might not charge the laptop the entire hour (esp if the lights dim during a presentation/video) so it might trickle charge only 1-2%, so if you're taking notes, it might use 5% of the battery in one hour without any charging and it might shutoff before the end of the hour, but only 3% with indoor lighting (i.e 2% remaining, enough to save the document and transfer the file to another PC in the dorm and recharge the laptop on the campus quad or once back at the dorm).

Update 8-9-22

This project has always been idea that needed to get off the drawing board before if could realistically be prototyped. As mentioned before, I am not an engineer with programming or PCB capabilities. This projects exists solely to explore *how*, *not if*, it can be done.

[update 12-25-22- as this section was written in August of 2022, and pre-birdsite takeover, I should add, that "not a huge Tesla fan, even in August, was a polite understatement] I am not a huge Tesla fan, but I found this story very interesting. The GigaPress example: Tesla reached out to the six largest 6 die-casting makers in the world and asked them if they could make a single press for a Tesla. 5 of them said no, One of them said, "maybe."

Source:

https://www.teslarati.com/tesla-new-giga-press-supplier-rumor/

https://www.facebook.com/watch/?v=628017608199814З

https://en.wikipedia.org/wiki/Giga_Press

“When we were trying to figure this out, there were six major casting manufacturers in the world. We called six. Five said’ no,’ one said ‘maybe.’ I was like ‘that sounds like a yes.’ So with a lot of effort and great ideas from the team, we’ve made the world’s biggest casting machine work very efficiently to create and radically simplify the manufacturing of the car,” Musk said."

So this project isn't here to debate why something can't be made, or why I should look into something with more performance and, in *your* words, "marginally increased power consumption". I understand there are countless projects out there that can deliver that. My communication skills are not always the best, but I try to clarify and revise whenever I can to help drive such a point. Part of why I somewhat gave up on this project a year ago is because I realize that solar-powered phones/laptops are not in demand and I needed to take a break since I didn't have any idea what to do with it. Breaks can be good and refreshing.

What has changed since 2021? Well, there are a handful of more battery-less microcontrollers. No 10mW Microprocessors that can run linux though.

I have followed technology news for over 15 years. I am a occasionally a consummate news junkie. In 2011 I first read about the solar powered Claremont demo by Intel. Since then, I have never found any any chipmakers that released NTV-capable chips, save for microcontrollers like Ambiq Apollo 4. The only application processor that was mentioned to use NTV is/was the MicroMagic 10mW processor over a year ago, and hasn't been released or updated. The Quark x86 line of processors was discontinued in 2016, after the Raspberry Pi and Soc market gained dominance. But what I find fascinating (and frustrating) is that no one is clamoring over the potential for minimum (ntv) power chips that could be run in fully solar-powered laptops if only they were accompanied by a low-power screen like memory in pixel or e-ink. The Claremont was a 32nm optimization of a Pentium P54C.

https://en.wikipedia.org/wiki/Instructions_per_second#Millions_of_instructions_per_second_(MIPS)

Intel Pentium	188 MIPS at 100 MHz	1.88	1.88	1994	[54]

Without necessarily seeking the bloatware that followed it, it is a goal of mine to see a portable Qwerty-based device that can run user-space applications on solar, not just microcontrollers that can on a device using 1990s software with no need to have the same security protections of network stacks since it is primarily designed for an offline leisure device. The MicroMagic is said to be run between a 10nm and a 20nm FinFet process, which would make running an application processor on solar quite feasible. Lower process nodes routinely are used to run chips at faster clocks to consume the same amount of power. This is the norm. That doesn't really interest me- it bores me in fact. The Claremont was rare in that it didn't always try run faster because it could. It set a limit and delivered it. Partly why the Quark was discontinued- One news site said that Intel didn't know what to do with it. But slow doesn't mean useless. It just means "niche" in market terms.

(Update 8-31-22. Without having any proof, I have speculated for years that the reason Intel or Samsung didn't release a solar power able processor isn't that it wouldn't necessarily be unsellable or an unpopular product, but because it could eat into sales of the market segment above it- I.e Celeron processors)- if the processor was feature rich enough -"good enough computing" -see https://rhombus-tech.net/whitepapers/ecocomputing_07sep2015/

Cold fusion was once ridiculed because of some poor publishing issues. Today cold fusion is a respected and serious endeavor that scientists are quietly researching, hoping to optimize the yields of fusion power. Edit 12/17/22: net fusion power discovered: https://www.science.org/content/article/historic-explosion-long-sought-fusion-breakthrough

You don't hear computer engineers quietly researching solar powered computers for the same reason. There no stigma nor shame in announcing a press release once a year or two, along the lines of, "We're still working on developing a solar powered cpu that's fast enough to run Windows 11 but we're not there yet." So if no one is doing that, maybe it's a secret weapon -that is- a mature, ready to release product on the backburner that would get launched if Intel lost market share for whatever reason (i.e ARM processors/RISC). Some of these speculations could be extremely unlikely. But I am just devising a laundry list of scenarios that could ever happen if market share ever shifted in such a way that favored a startup fabless company that seeked to take share away from Intel. In a recent development, last year, Intel announced that they would open up their foundries to other businesses. So in a way, they joined the pure-play business in that they'd cede some of their exclusive foundry space to other ISA developers. Would competitors be able to rent the EUVs? Maybe... but then again, wouldn't this just accelerate Intel's development of solar powerable NTV processors?

9/7/22: The critique here is that few are asking, why was the demo just a demo? When someone says ""Nothing to see here, move along." or "no it's not for sale" That raises more questions than answers. It's not about whether the technology was ready to run Windows 7 in 2011. I know it wasn't ready for that. It's what the demo represents in terms of potentially new ways of doing computing.

https://en.wikipedia.org/wiki/Abductive_reasoning

"It starts with an observation or set of observations and then seeks the simplest and most likely conclusion from the observations. This process, unlike deductive reasoning, yields a plausible conclusion but does not positively verify it. Abductive conclusions are thus qualified as having a remnant of uncertainty or doubt, which is expressed in retreat terms such as "best available" or "most likely". One can understand abductive reasoning as **inference to the best explanation**,[3] although not all usages of the terms *abduction* and *inference to the best explanation* are exactly equivalent.[4][5]"

To that effect, abductive reasoning not only creates a new set of plausible possibilities missing from the blind spots of both inductive and deductive reasoning, but creates a limited set of avenues to explore:

1. Those who do not understand the technology, and do not think it is possible.

2. Those who do not understand the technology and do think it is possible.

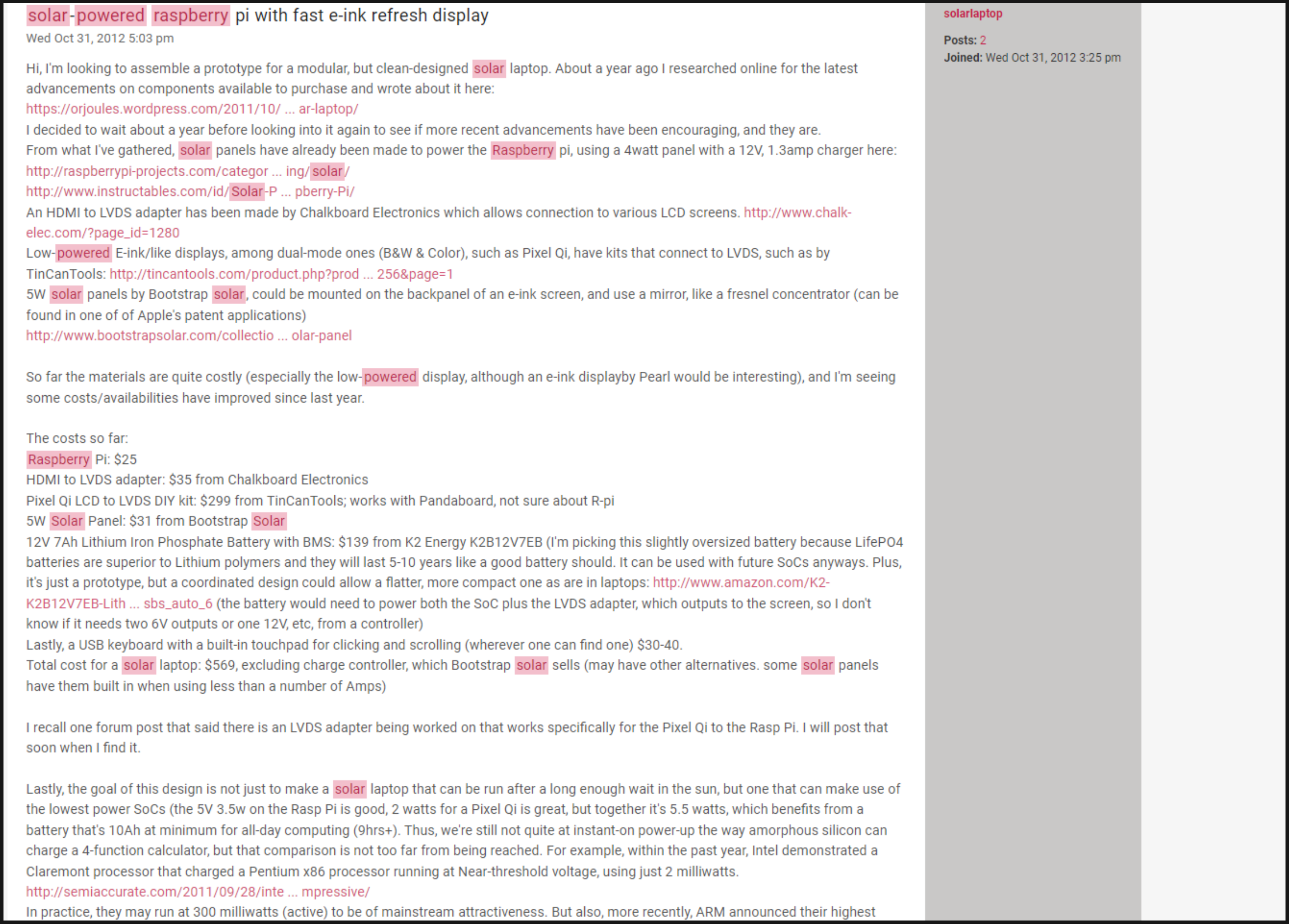
3. Those who do understand the technology, and think or know it is possible but have a conflict of interest- money, ideology, compromise, ego

4. Those who do understand the technology, but have not enough money or power to develop it and wish to promote it.

5. Those who do understand the technology and do have the power to promote or develop it.

There could be a sixth category too. But that is all I can think of. I don't have a sixth sense ;)

I understand something isn't for sale, but tech news readers who enjoy reading about the latest Threadripper or 64 core Xeon processors should pause and think for a second, that what was a demo has effectively turned into a leak. My question is, why wasn't anyone else asking about it in 2011? I was talking about it in 2012, where I referenced the same Semi-Accurate article on the Raspberry Pi forum: https://forums.raspberrypi.com/viewtopic.php?p=206323#p206323



solar-powered raspberry pi with fast e-ink refresh display

Wed Oct 31, 2012 5:03 pm

solarlaptop

Posts: 2
Joined: Wed Oct 31, 2012 3:25 pm

Hi, I'm looking to assemble a prototype for a modular, but clean-designed solar laptop. About a year ago I researched online for the latest advancements on components available to purchase and wrote about it here:
https://orjoules.wordpress.com/2011/10/ ... ar-laptop/
I decided to wait about a year before looking into it again to see if more recent advancements have been encouraging, and they are.
From what I've gathered, solar panels have already been made to power the Raspberry pi, using a 4watt panel with a 12V, 1.3amp charger here:
http://raspberrypi-projects.com/categor ... ing/solar/
http://www.instructables.com/id/Solar-P ... pberry-Pi/
An HDMI to LVDS adapter has been made by Chalkboard Electronics which allows connection to various LCD screens. http://www.chalk-elec.com/?page_id=1280
Low-powered E-ink/like displays, among dual-mode ones (B&W & Color), such as Pixel Qi, have kits that connect to LVDS, such as by TinCanTools: http://tincantools.com/product.php?prod ... 256&page=1
5W solar panels by Bootstrap solar, could be mounted on the backpanel of an e-ink screen, and use a mirror, like a fresnel concentrator (can be found in one of of Apple's patent applications)
http://www.bootstrapsolar.com/collectio ... olar-panel

So far the materials are quite costly (especially the low-powered display, although an e-ink displayby Pearl would be interesting), and I'm seeing some costs/availabilities have improved since last year.

The costs so far:
Raspberry Pi: $25
HDMI to LVDS adapter: $35 from Chalkboard Electronics
Pixel Qi LCD to LVDS DIY kit: $299 from TinCanTools; works with Pandaboard, not sure about R-pi
5W Solar Panel: $31 from Bootstrap Solar
12V 7Ah Lithium Iron Phosphate Battery with BMS: $139 from K2 Energy K2B12V7EB (I'm picking this slightly oversized battery because LifePO4 batteries are superior to Lithium polymers and they will last 5-10 years like a good battery should. It can be used with future SoCs anyways. Plus, it's just a prototype, but a coordinated design could allow a flatter, more compact one as are in laptops: http://www.amazon.com/K2-K2B12V7EB-Lith ... sbs_auto_6 (the battery would need to power both the SoC plus the LVDS adapter, which outputs to the screen, so I don't know if it needs two 6V outputs or one 12V, etc, from a controller)
Lastly, a USB keyboard with a built-in touchpad for clicking and scrolling (wherever one can find one) $30-40.
Total cost for a solar laptop: $569, excluding charge controller, which Bootstrap solar sells (may have other alternatives. some solar panels have them built in when using less than a number of Amps)

I recall one forum post that said there is an LVDS adapter being worked on that works specifically for the Pixel Qi to the Rasp Pi. I will post that soon when I find it.

Lastly, the goal of this design is not just to make a solar laptop that can be run after a long enough wait in the sun, but one that can make use of the lowest power SoCs (the 5V 3.5w on the Rasp Pi is good, 2 watts for a Pixel Qi is great, but together it's 5.5 watts, which benefits from a battery that's 10Ah at minimum for all-day computing (9hrs+). Thus, we're still not quite at instant-on power-up the way amorphous silicon can charge a 4-function calculator, but that comparison is not too far from being reached. For example, within the past year, Intel demonstrated a Claremont processor that charged a Pentium x86 processor running at Near-threshold voltage, using just 2 milliwatts.
http://semiaccurate.com/2011/09/28/inte ... mpressive/
In practice, they may run at 300 milliwatts (active) to be of mainstream attractiveness. But also, more recently, ARM announced their highest

The hacker ethic is to turn what was never intended by an inventor's purpose to deliberately announce a new functional purpose. How a technology is used faces the same dilemmas as any other technology. Are they afraid the technology will fall into the wrong hands? Don't be evil. Use technology for good. No doubt about that. That isn't the focus of this hack- that is something that only comes after a functional prototype. But feel free to educate yourself:
that: https://en.wikipedia.org/wiki/Hacker_ethic Hackers are not "Crackers"

https://stallman.org/articles/on-hacking.html

https://www.educba.com/hackers-vs-crackers/

----

https://www.tomshardware.com/news/intels-foundry-services-lands-mediatek-as-a-16nm-customer)

[only parenthetical section between dashed lines is part of 8-31-22 update. Below this line is from an earlier post and resumes from above dashed lines.]

I have tons of ideas what I could do with a processor running windows or linux at 90mhz. Partly it is nostalgia. Often times when I contact someone for engineering design questions, I often get this blank response, like, "why would I want to spend all that money to do that, or why not use something more powerful?" In the customer service world, the "customer is always right". I dislike using that phrase, because I have worked in customer service before, but I understand it. There are many SBCs out there that have a specific function or niche advantage over other SBCs, yet it would be just as innovative to see more chips that are designed based on their power consumption first, and not how much horsepower they have.

Designed for writers first. A scenic detour from an all-purpose laptop. With an OS like Nanolinux but can run on 8MB RAM:

https://sourceforge.net/projects/nanolinux/ , https://arm.slitaz.org/rpi/ or http://www.tinycorelinux.net/welcome.html

Not designed to replace your productivity/performance-geared laptop, but to supplement it. (Though for writers, typing without distraction *is the productivity).*

Unless this turns into the OLPC, in which it would be someone's 1st laptop: https://www.theverge.com/2018/4/16/17233946/olpcs-100-laptop-education-where-is-it-now

Inspiration:

https://medium.com/this-should-exist/prose-a-distraction-free-e-ink-laptop-for-thinkers-writers-4182a62d63b2

https://ploum.net/the-computer-built-to-last-50-years/

https://gemini.circumlunar.space/

This project seems realistically feasible within 1-2 yrs. However, I cannot do it alone due to the amount of engineering, money, and time involved. I don't know how many people it would take, but I imagine it would be somewhere on the scale of the developers for the original Raspberry Pi. Since this is intended to be a free and open-source project, I have no interest in owning the IP for this. I would rather see the project develop and then various homebrew manufacturers capable of producing the laptop's STL's designs. I don't expect anyone to know how the entire laptop motherboard works if they specialize in only one or two technologies. But by recruiting specialists in each area, the product could be realized. Therefore, this project is designed to be interdisciplinary and multidisciplinary. See this post on how I think this project is like designing a space shuttle: https://hackaday.io/page/9864-3-quotes-from-the-big-three

This is a niche product, understandably, at first glance. But the value of this product aims to look deeper and a longer view towards exploiting the power efficiency of a microcontroller and an IoT power management towards general purpose userspace applications. Indoor solar panels could power an entire laptop, if the thermal design power (TDP) were limited to 5mW. I think it is better to set a limit and creatively determine how many apps could be designed under that, rather than allocating an "unlimited" or unconsidered TDP (above 100mW-30W).

This design concept could be called "PowerFirst." That is, TDP is the first consideration of the laptop, when it is usually the last priority. There are many options when shopping for a laptop. Power consumption being the #1 priority is virtually non existent. By developing in this manner, PowerFirst is a starting point for "building-up"- that is, determining an efficient module or package of core applications- text editing, pdf reading, and then implementing more efficient solar panels and more efficient procesors (i.e Apollo4, 5nm Arm-A processors) as consumer solutions become available. Therefore, this laptop concept represents a platform, much like an ATX motherboard form factor doesn't change shape or basic I/O expectations- maybe PS/2 getting phased out after 20 yrs for all USB, but very long-term specifications, rather than ephemeral.

Uninstalling unused apps and storing them on an uSD card, rather than loading all the apps into RAM, might be a way to save on disk space. Ideally this could boot the entire OS in RAM, but only one or two apps at a time. This is an exercise in constraints, and as my avatar suggests, I enjoy solving puzzles. The technologies (Moore's Law/Dennard's Scaling in terms of performance efficiency already exists at this power consumption level to run user-space applications, but they haven't been designed, assembled and programmed yet to work together. At the core, a mobile ITX-like motherboard with solar power manager & reflective/eink display connectors could be prototyped, using off the shelf components, but once the software and hardware is tested for functionality, a converged system in a chip integrating all the components on a PCB could be developed.

A few comments about this project. The overall concept has been established. What hasn't been done is integrating these six blocks- energy harvesting, energy management, microcontroller, memory, display, and software. Again, this would definitely be in the level of investment that the Raspberry Pi had to do in order to develop the first $25 PC. The frequently asked question I already get is, why would anyone want to pay more for a chip that does less? The answer to that is has more to do with the philosophical preferences limiting information and sensory overload, rather than an industrious or rational answer. This concept may appear to resemble https://en.wikipedia.org/wiki/Reductio_ad_absurdum but after reading that, a solar powered laptop sounds much more rational than a system built for the sake of being different. That is, I could suggest an Overton Window alternative further to the left of this idea, but I believe this idea is enough food for a day's thought.

If you are still interested in this project after reading this all, thank you. Feel free to contact me, whether you agree or disagree, I welcome all feedback.

⇡ Going up?