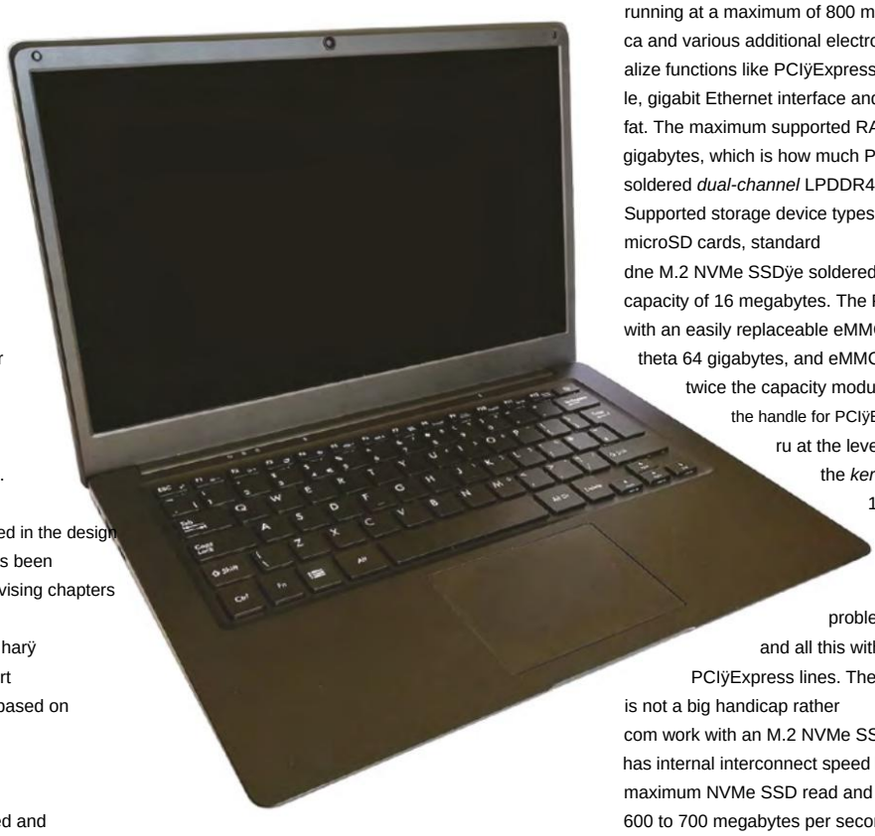
Pine64 PineBook Pro

A very **open** laptop

S vaki functional laptop represents a whole consisting of hardware and software, and *Piÿ The nebook Pro* is one quite uniquey on the whole. What makes it so is the way in which the hardware and software that make it up were created, so to begin with, to clarify thatÿ mo. The hardware was created on a relatively common jen way with SBC manufacturers: the parent company hires engineers who create electrical diagrams for the device, then the design thus created is forwarded to the selected OEM *(original equipment manufacturer,* read: factory), and the parent company's team, through cooperation with the OEM, follows up production, performs various tests



and finally realizes the sale itself of final products. In the case of *Piÿ nebook Pro* (hereinafter: PBP), as with other *Pine64* products, the situation is somewhat more complicated, because the *Piÿ The ne64* community is involved to some extent in the design and testing of hardware. ra, for example, by giving suggestions for the correction of problems that have been observed in the design jnu. For example, the author of this article has been working for a long time on suggestions for revising chapters not printed circuit boards for PBP.

Everything described so far was valid for harÿ the door part, and the entire software support ka for all *Pine64* devices, including PBP, is based on community work. In other words thus, there are no official *Pine64* inÿ installation *image* files, that is, not it says some official *Linux* distribution created and maintained by *Pine64* as a company, or something similar. Everything that powers PBP is the result of community work, be it *Pine64* forÿ unit or, for example, the *Linux kernel* or the KDE community. Of course, certain *Linux* disÿ tributions make an extra effort, providing specific support for PBP (as, at priÿ mer, *Manjaro* ARM, whose collaborator and author of this article), which is often necessary, since all *kernel* patches required for funcÿ functional PBPs have not yet become part of the *mainline Linux kernel.* Yes, you read that right, PBP runs a *mainline Linux kernel* *nel,* which means that there are no ancient BSPs in circulation *(board support package,* that is, the *kernel* that the SoC *vendor* provides) year for its product), but it also means that certain software functions are still missing. The community works on it, and give backÿ we will talk about this in the rest of the article.

Obviously, this display of devices does not follow any usual flow, and the goal is to avoid repeating common things and use the space for what *sets Pine64* and PBP apart from other manufacturers and devices on the SBC market. Namely, the vast majority of SBC manufacturers are oriented towards offering functional SBCs to end users through a kind of "rounded system". ma" that makes it possible, either through some sort of "standard way" of doing certain things or through sets of inÿ installation *images* that often use some ancient versions of the BSP *kernel* and *userspace* software. You will ask yourself what is wrong with that, and the answer is that this is how "moldable- it" of the users that hinders their creation tivity, that is, that in this way, timeÿ nom, the death of software support is inevitable. In other words, the only real way to support it in the *Linux kernel* for some hardware save from the teeth of time isÿ is to become part of the *mainline Linux kernel la.* After turning on support for the nor hardware in the *mainline Linux kernel,* the concern for continued support is largely left to the *Linux kernel* community.

As for the hardware specifications of PBP, they can be called relatively solid. nim for today. Essentially, PBP is, forÿ right, modified *RockPro64* SBC, upaÿ forged into the body of the laptop, with accessories necessary for its operation, such as additional hardware logic that serves the built-in battery. The SoC on which PBP is based is *Rockchip RK3399,* which uses ARM "big.LIÿ TTLE" architecture with two "big" *ARM Coreÿ texÿA72* core which, according to the official specÿ fications, they work at a maximum of 1.8 gigaherÿ ca and four "LITTLE" *ARM CortexÿA53* cores operating at a maximum of 1.4 gigahertz. The SoC contains a total of slightly less than two megabytes that cache for CPU cores, the MaliÿT860 integrated GPU running at a maximum of 800 megaherÿ ca and various additional electronics that reÿ alize functions like PCIÿExpress masterÿ le, gigabit Ethernet interface and USB port fat. The maximum supported RAM capacity is four gigabytes, which is how much PBP comes with, in the soldered *dual-channel* LPDDR4 variant. Supported storage device types include eMMC modules, microSD cards, standard dne M.2 NVMe SSDÿe soldered SPI chip, with a capacity of 16 megabytes. The PBP comes standard with an easily replaceable eMMC module, covers theta 64 gigabytes, and eMMC is also available twice the capacity module. Per- the handle for PCIÿExpress is in the hardwareÿ ru at the level of version 2.0, but in the *kernel* it is limited to version 1.0 (because, according to *Rockcÿ hipa,* unexplainedÿ of possible proÿ problem in operation as 2.0), and all this with the available four PCIÿExpress lines. The limitation to version 1.0 is not a big handicap rather com work with an M.2 NVMe SSD, as *the RK3399* has internal interconnect speed limits that result in maximum NVMe SSD read and write speeds of around 600 to 700 megabytes per second, which is still pretty good for a fairly old ARM SoC. Maximum read and write speeds with eMMC modules range from 150 to 180 megabytes per second, which is still quite but it's fast and in good balance with the overall level of performance that the *RK3399* can offer. For more details about *RK3399 ,* see its *datasheet* at **i.sk.rs/ 358744,** and for more details about *RK3399* within PBP, see put *Device Tree (DT) kernels* behind PBP, doÿ entered at the address **i.sk.rs/358745.**

What gives the above hardware specifications additional power or, boÿ in other words, the real strength is the openness of the hardware in terms of specifications and software Tver view. If we go "into the small intestine", the only closed segments are softwareÿ of support for the *RK3399* SoC are the so-called *Maÿ skROM,* machine code that is written directly sleep in the ROM part of the chip during its manufacture, support for HDCP *(Highÿbanÿ dwidth Digital Content Protection)* i poÿ handle for some additional functions of USBÿC ports. So far, the community has implemented i

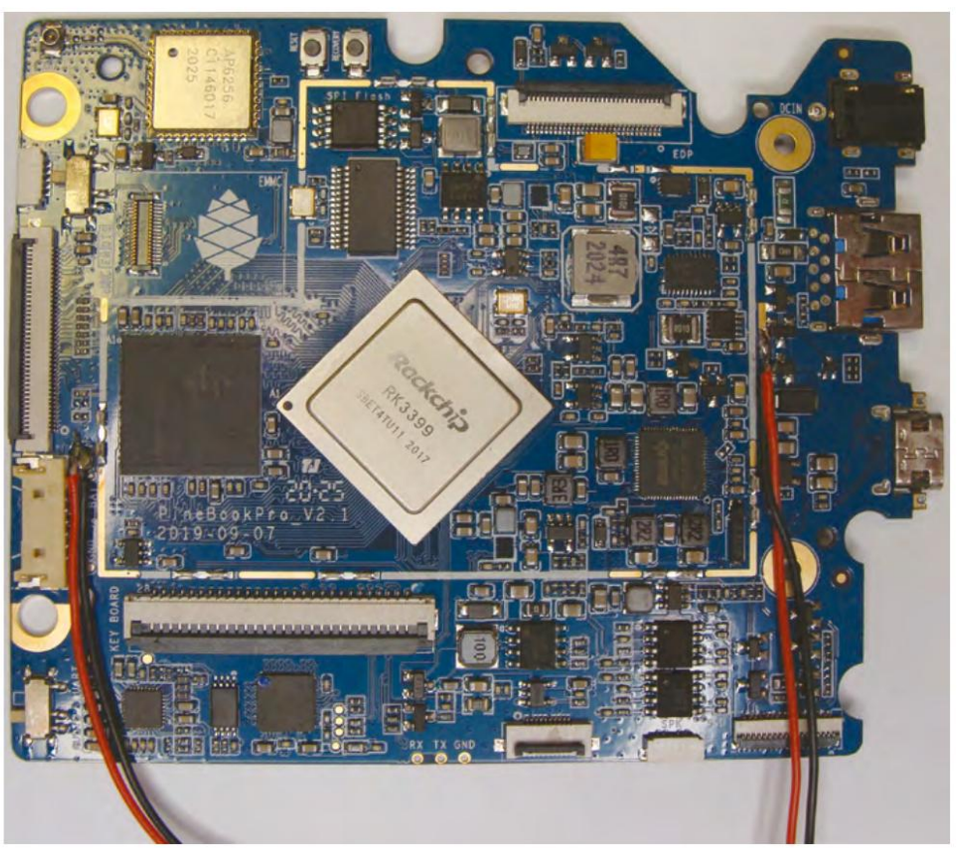
partial *reverse engineering MaskROM* code, with the aim of insight into its operation, which, by the way, is limited to the basic *poweryon* iniy *cialization of the RK3399* to the level required to load the *bootloader* from one of the support of storage devices. Without the remaining two of the previously mentioned *blobs, RK3399* it can be used without problems, moreover, the *default build of the bootloader* does not include hears HDCP support. Speaking of *bootloader,* it is used *de fay cto* standard for the ARM platform, from bow, *upstream UyBoot,* in vany nation with *upstream TFyA (Trusted Firy mwareyA),* for which, of course, upy provided complete source code as part of *upstream* projects. She makes the fact that *upstream* variants of the *bootloader* and *kernel* are used is enough to put a huge smile on the faces of those who want to know and find out exactly what and how the hardware they use works, without exaggerationy

no reliance on posty janje *black boxova* i *bloy buoys* who "do something", and maybe in addition "work de and a little more". Admittedly, when it comes to tra PBP as a whole, there are a few more *blobs* that include firmware to embedy nu USB keyboard, for the built-in *touchpad* and for the soldered SDIO *WiFi/ Buetooth* module using the *Broadcom BCM4345* chip. For- the unit also carried out a partial *rey verse engineering* of the keyboard firmware, with the aim of correcting certain problems in in operation and to enable changes to the keyboard's operating mode, for example key remapping at the firmware level.

Of course, not everything is ideal, so, that's it. reliability and use of *upstream* versions of *bootloy skin* and *kernel* at the same time represent the main source of defects and "mushyness" of PBP. For example, there is no software poy videoyforyhardware decoding handle writes despite the existence of *IP cores* for that in *RK3399.* As another example, support for the USByC port is partial and, as of yet, unproven. for some reason, support for *DisplayPort Alt Mode* via USByC port, which allows connecting an external monitor via USByC port, and is implemented in the form of an *outyofytree kernel* patch, simply refuses to work on all PBPs. Also, there are certain hardware problems, such as overheating of the *Texas Instruments BQ24171* integrated circuit that realizes the charging of the built- in battery. is of 10 thousand milliampereyhours, which temporarily causes the battery to stop charging even when the charger is connected. Solving this problem requires configuring a restrictive nog *thermal throttling* for the CPU core and GPU, in order to reduce the *heat soak* that gey is not released by RK3399 , and is "absorbed" by *BQ24171,* which is enhanced by the passive nature of cooling sty of that PBP. Although, some owners of PBP report that they do not have these problems with *BQ24171,* and some have successfully solved this problem by running the cooling system. All this is part of the real nature of PBP, which is more of a development platform than a final product for an "ordinary" user.

Let's finally get back to what was can be called the usual description of a laptop. There is not much to say here, since the PBP looks and behaves like a completely ordinary laptop, and from the external ports there are connecy tor power supply, which delivers three amps at five volts, one USB 3.0 port, one USB 2.0 port and one USByC 3.0 port, as well as a microySD card reader and a 3.5ymm TRRS port for sluy cups. The keyboard is very comfortable to use. which is contributed by the rather long key stroke for a laptop keyboard. for more the author of the article considers the keyboard on the PBP much more comfortable to type than, for example, the keyboard on the *HP ProBook G8* laptop pu. The screen on the PBP is excellent, 14 inches diagonally, made in IPS technology and with a notch with a resolution of 1920×1080 pixels. It offers nony the worse the range of colors it can reproduce mints, but it makes up for it with a fairly high contrast ratio (1000:1), making the displayed image look really great, especially considering the price range the PBP belongs to. Ugray a webcam is also provided, usually position above the screen, but the quality of the video it produces is, frankly, quite poor. Two speakers and two microphones are built in, all of mediocre quality. Spoy the smooth surfaces of the PBP's case are a pleasant surprise, as they are made of magnesium alloy and make the exterior of the PBP have a certain amount of *premium* feel. remain the laptop is made of plastic and is well fitted, and as a big drawback, because of the lens ness, we can state that the first series of PBPs had somewhat worse plastic, which often caused the plastic part of the case to crack in the area of the screen joints.

This, obviously, is not a common occurrence. laptop model, since PBP is not "just another laptop", but is, in practice, a development environment for the *RK3399* platform that can also be used as a laptop, in case you are ready to accept some of its "flies"



ness" and limitations, and for the sake of what ney di on the issue of openness. What other laptop offers, for example, a serial port for *debugging* as a second function for the 3.5ymm coy nector for headphones, and there are *solder pads* for RX, TX and GND on the main PCB for that same serial port(!?). Or, what other laptop offers, if you put in a little effort, access to two unused USB ports on the internal, fory to a soldered USB 2.0 *hub* chip? Few laptops offer such a level of openness, and we change again the availability of complete wiring diagrams and the use of *upstream booy tloadera* i *kernela.* For all additional information information about PBP, see the official page on *Pine64* **(i.sk.rs/358746).**

As a final curiosity, this article is on written on one PBP that the author in the last he has been using them every day for a year or two, as a primary computer, with satisfactory performance performance level, and whose *uptime* is at the time of writing this article was around 330 days. Not bad for a laptop.

Dragan SIMIy