Show Notes

Certified transcription #000000E2 provided by NetcastExtreme™ - 26 pages - edited by content creator for: [clarity, intent, terms, names] revision #0000001B.

Show: Single Board Computer and Virtual Private Server Show™
Title: "Stuff and Things"
Number: 0001 rev(0000)
Date Recorded: 2016-06-05
Date Released: 2016-07-20
Date Last Revised: none
Running time: 1 hr 13 minutes 05 seconds - (74 minutes)
Media Type: mp3 audio, 52.6 MB (52,643,840 bytes)

Joe C. Hecht

Digitally signed by Joe C. Hecht
DN: cn=Joe C. Hecht, o=Code4Sale, LLC, ou=Code4Sale, LLC, email=code4salellc@code4sale.com, c=US
Date: 2016.08.02 06:41:41 -05'00'

Sections

1:13:05 Show End

Links
Indexed Links
Show's Media Links
Show's Media Hashes

Welcome to Single Board Computers from Code4Sale.com.

With your host, code guru Joe C. Hecht, and special guest, armbian chief architect, Igor Pečovnik.

Brought to you by NetcastReserve™ providing no compromise Netcasting, all for Peanuts™,

and by armbian the universal Linux operating system, for ARM development boards.

Hey there, this is TJoe™, coming to you from the SwampWërks™ here at Code4Sale, LLC, in sunny Destin, Florida.

It's Sunday, June 5, 2016 and welcome to the first boot of the Single Board Computer and Virtual Private Server Show™ entitled "Stuff and Things".

We have a bit packed show for you today, including the start of a great project, a fantastic interview with Igor Pečovnik, the chief architect and creator of the armbian universal Linux operating system, and we have a giveaway for a ODROID C1+ ARM development board.

Since this the first show, I guess I'll start out by telling you a little about myself, and why I started the Single Board Computer Show™.

I got into computers backs in the early 1980's, when I was in the publishing business. I started writing code for graphics and went on to develop some graphics and gaming engines, drivers, publishing applications, and software developer kits. I'm a former Borland guy, so I suppose I know a little about compilers.

I got into single board computers when I got my first Raspberry Pi, and just like if you give a mouse a cookie, I've been collecting them ever since.

I love my Raspberry Pi, but I quickly found out there was other boards too, and some of them were pretty cool. I started feeling like the Raspberry Pi was getting all of the spotlight in the news. I started sending out boards like the BeagleBone Black, and the Banana Pi, to netcasters like DoorToDoor Geek, asking them to share a little bit of limelight with the other boards.

I started this netcast because there seems to a void in the discussion in single board computers.

You have The Pi Podcast and they talk about the Raspberry Pi, but there's other boards.

You have The MiniPC Show coming out of podnutz.com. Man I love that show. I'm a big supporter, and you just got love DoorToDoor Geek. He's a very close friend of mine, and a huge supporter of computer community. I highly recommend the show. You should go over to podnutz.com and subscribe.

It's a wide ranging in show about MiniPC's. They read specs sheets from some the thousands of boards that become available, and talk about general use cases like using one for a MediaPC, an OwnCloud server, or router.

I looked far and wide for show that wanted to talk technical about single board computers. Discuss projects, get your hands dirty. I wanted the show that appealed to both the complete novice, right up to the guru level, embedded system developers.

So here we. Welcome aboard!

I guess I better warn you that I tend to speak my mind about things, and I hope you do too. I'm quick to find fault, and I hold little back. Some people call me a loose cannon. I like that. At the same time I like to think that I accept criticism as well as I give it. If get something wrong, call me out on it, and I'll proudly stand corrected. Around here is not who's right, but what's right that really counts.

Now this is my first netcast, so expect that there will be few kinks along the way. I got to get the website up and running, get proper show notes, get the audio quality up where it needs to be. Please bear with me, turns out this is a lot harder than I ever imagined.

By the way, I don't want this to be my netcast, I want it to be yours too!

If you're interested in co-hosting a show, being a guest, presenting a project, don't be shy! Come be part of the team!

Now I'm not pretend to know everything about single board computers, I may have been working with these boards for a few years, but I feel like a complete novice, and I'm a software guy. I really need a technical lead for the show. Someone that really knows these systems from hardware perspective, and chime in with some technical knowledge when we fall a bit short. If you think you are that person, please let us know.

OK! Let's get started!

I'm going to ask the gurus out there that are listening, to skip ahead by 16 minutes, so we can bring our single board computer newbies up to speed by defining "what is a single board computer", and then tell a little bit about the different operating systems that are available, and give out a few little shopping hints.

Now just like a regular computer, we have the basic pieces: a CPU, some RAM, a video subsystem and a way to hook things up to it. With most computers, these things exist on separate boards, and the boards can be changed in and out, mixed and matched, and upgraded. It's all very flexible.

With a single board computer, all those things are generally built on to a single chip, so you can't change the components in and out.

Sounds kind limiting, eh? The advantage is thay can be made to be very very small, and often very inexpensive. These days, they can put an entire system on a single chip. That system on a chip is so small, you would have trouble finding a way to connect something to it. So they mount the chip on carrier board that they mount connectors on.

Now, these chips are really small. I've got one this size of postage stamp, and that even includes the carrier board. Now generally you get one of these boards for cheap. Perhaps as low as even five or ten dollars, and it has just about everything that you need, but the keyboard, the monitor, perhaps a SD card, and something the power of the board with (you can use something like your phone charger).

And what you get out of the deal? A pretty complete computer, capable of doing and all sorts of things from being web server or a router, a media PC, a play toy, right up to driving a car, flying a plane, or being a complete desktop computer replacement.

These little boards get faster, and cheaper, and more capable every day, and at some point they will even overtake that expensive desktop computer you may have spent thousands of dollars for. In fact, it's happening today.

I have one of these little systems that's the size of a credit card, and it's a fully blown supercomputer. I hear that the next generation of these chips may have the power of 1500 MacBook Air's, all in a single chip.

At minimum, all you need to get started with one of these little boards is a phone charger to power the board with, and a SD card to burn an operating system to. You can often run these boards remotely, with no keyboard and monitor. We call that "headless", and you login into it from your main PC, a phone, or a tablet.

But folks generally like to add in a keyboard and a mouse, along with a cable that attach it to a monitor or a TV, and often times an Ethernet cable, although these boards can be made to run over Wi-Fi.

Now most of these little boards are of the ARM variety. There are just like your phone, and they are made with low cost, power efficient chips that have something called a "reduced instruction set".

For some things, they may have to work a little harder than the chip on your desktop, but that's OK, because they cost a whole lot less, and they use a whole lot less power. While they probably won't run your standard version of Windows, but they will run something that's probably even better. Linux, Android and ya, there are some flavors of Windows that might run, but we are not going to talk much about that. Some people have had some very good success running Windows programs under Linux using some auxiliary software that you can purchase.

It seems like that there are hundreds and hundreds of these little boards available, so the problem is figuring which one of these you want to buy. The Raspberry Pi runs about 35 dollars, but they have versions that go as low as five dollars. Consumer boards range from five dollars to maybe a hundred dollars, with commercial boards going up into the hundreds or even thousands of dollars.

I think the sweet spot's about thirty-five dollars, and you can get a really really good board from between nineteen to forty-five dollars.

I want you to know that we are board agnostic here. It's really an interesting term. Joseph Normandin over at militaryaerospace.com is reported to have coined the phrase. It's getting some wide use these days. When we say it, we mean that were not fan boys or fan girls

of a given board. We weigh every board for its merits and its capabilities, and what it can be used for. There is no perfect board, but there may be a perfect use for a given board.

Currently I have a pretty good range of boards that I work with. I have several versions of the Raspberry Pi. I have Banana Pi's, ODROID XU4's, ODROID C2's, ODROID C1+'s, Pine64's and the coveted Jetson TK1 and TX1 boards.

I sure have been through a lot of boards over the years. I've had BeagleBones, CUBoxes, and, well, you get the idea. I'm no export either, but one thing I've learned about these boards is they are only as good as the operating system provided. In short, if your OS is flaky, then your board might as well be a brick.

I've tossed plenty of boards for that reason, and when it comes to OS for a given board, I believe in at least two things:

1) The manufacture should supply at least one officially supported OS image, and,

2) That image should be immediately available as open source, so that the code can be inspected, and other developers can chime in and fix things when the manufacturer fails to.

A lot of folks run Android on these little boards. I'm not sure I'm fully understand why either. I get the development side, but I want to run Android, I have my phones. Maybe they are lacking good GPIO header. If you have an opinion on that, feel free to fill me in. I'm curious.

For our novices out there, you are going to hear the word "GPIO" a lot. And it's really cool. It stands for "General Purpose Input and Output". In short, it means you can hook-up some wires to your board, and program them to turn stuff on and off, or find-out if something is turned on or off.

Believe it or not, you can get pretty fancy with that. Do things like drive cars, fly planes, check the weather, know when your plants needs watering.

Pretty much any purpose you can dream up, they have some sort of servo-sensor thingy to do it with. And if you hook that up to the internet, you get the internet of stuff, and things.

OK! Back to operating systems. Now generally for these little boards, you are going to be running some flavor of Linux or Android. Now some folks are going to be quick to point out that Android is actually a flavor of Linux, and they would be correct in that. But Android feels a lot differently than your run of the mill Linux distro, so for the sake of our sanity, we're just going to talk about them like they're two different types of systems.

Now for most boards you will be able to find several different versions of Linux to tryout, and perhaps a version of Android as well. For Android, depending on the version, you may even have some luck at getting some of software that you've purchased on some play stores to work on a board.

You start out by downloading an OS image for your board from somewhere, you probably start out with a official image from the manufacturer, and often there's third party images available as well.

Linux images generally come in three flavors: A minimal server image, a desktop image, or something else, where someone has takin some sort of image and adopt it for some specialized purpose, like being router, or some sort of sever, or perhaps a MediaPC. When they say an image is a minimal server image, it's probably is not actually setup to be to be server. It's just a minimal system image with very little pre-installed on it, making an excellent starting point, if you care to build a system from the ground up.

Desktop images comes with some sort of windowing system pre-installed. With Linux there is a ton of choices, some of them are very similar, and some of them are quite different. You will need to try out a few of these systems to see which desktop you like. Choosing a Linux desktop is kind of like shopping for a book, you shouldn't judge it by cover. You may not like some desktops right off the bat, but don't give up too early because they can usually be highly customized to please almost anyone.

People tend to use lightweight Desktops on these little boards to keep things snappy. The MATE desktop is becoming a popular choice, because it provides a lot of features with some of the heavier desktops, but its trim down enough to give a nice user experience on these little boards.

There's many flavors Linux and they call them distros. Some of them are very similar, and others are not. The most obvious thing that sets them part is how you go about installing software on them, using something call a package manager.

You will want to try out a few of these distros to see which ones are your favorite. Popular choices include Debian and Ubuntu, they are very similar and they are very easy to setup. Then there is something like Arch. It's a really minimal system where you have to setup almost everything from scratch. It may not be exactly easy for a newbie, but you are in total control about how your system gets built.

Distros and desktops can often be mixed and matched too. For one of these desktop images, someone starts out with a minimal system image, and then they pre-install some desktop system that they like. They may add some drivers that are specific to a given board, that allow things like the sound system to work, or to accelerate the desktop by making use of the board's graphics processing unit. And they will probably install a collection of third party software that they think you might like.

One of the problems of these little boards it that it's often difficult to find a system with the desktop you like, installed on distro you like, and get the accelerated drivers that you need to take advantage of specific hardware that make these boards run really great. There is some really neat tricks to do that, and we will be rolling up our sleeves and exploring some cool ways to get that done.

Just be forewarned that getting accelerated drivers on these little boards often as not as easy as just downloading the drivers and installing them, and many of the third party images don't actually come with accelerated drivers. There is a lot to it, and your choice of OS image can make all the difference in the world. After all, if you buy some great board for hardware capabilities that require a driver, and your OS doesn't actually have that driver, then you are paying for capabilities that you can't actually use. On the other hand, you might find the manufacture's image may not be all that great either.

Much of this hardware is really fancy, and experimental at the same time. Don't just believe the spec sheets, you may have to scour around on the internet to find out how well given board's hardware is actually supported. And hardware support can make all the difference in the world between having a great board and a brick.

Be especially careful when a new board hits the market. Many of the reviews are paid for, and almost certainly based on quick opinion,

rather than experience.

In the early reviews you often hear the term "it runs like butter". But I gotta ask, does the reviewer even know what butter is? They might be talking about margarine. It's a great descriptive term, but it does not tell me anything about the board.

The point I'm trying to make is, no matter how great the review, or how great the board sounds on paper, you really need to go to the forums, and take a look and see what the real life users are reporting, and what sort of issues there are having. Remember, no boards are going to be perfect, but you at least you know what to expect, based on real life experience.

You have to be careful when shopping for these boards for the words and terms that are used. I often joke about the Banana Pi Pro. I mean, what's so professional about it? Its got a Wi-Fi adapter that's hard to be proud of. It's worth perhaps 87 cents, including shipping from China. In no way does it compete with a brand name Wi-Fi chip, but someone in marketing came up with the word "Pro", slapped on the box, because they know some people buy labels.

To be totaly fair, it did have a couple of other miner upgrades made to it, but I think the marketing department deserves a gold star on this, because when you hear people talk about the Banana, you hear them talk about the Banana Pro, and unless you actually need that embedded Wi-Fi chip, you are probably paying a premium price for a premium label.

You have to be really careful when you are looking into the specifications of these boards. A good example is serial ATA ports. A few boards have real onboard SATA controllers. Others just mount a USB chip to the board and call it good. There is a huge difference it that. One gets you a true high speed connection, and the other is equivalent to a low performance USB dongle, that's worth less than a dollar. If you find a board like that, know that the intent is probably to mislead you.

The same goes with gigabit Ethernet controllers. These days, most of them are true gigabit controllers, although for some broads it's just a chip strapped to the USB port and will never achieve gigabit speeds, and sit around and sap the USB bus.

The true number of usable CPU cores a board has is another area to watch out for. Many of these boards have two sets of cores. One is for low power and runs at a slow speed, and the other one is for high performance and runs at a high speed.

Some boards can only use one set of cores at a time, while other boards like the ODROID XU4 may be able to use all of the cores, although it may not buy you very much in performance. The point here is that some manufactures will advertise the total number of CPU cores, and others will advertise only the number of usable CPU cores. The NVidia Jetson TX1 is a good example of this. It actually has nine CPU cores, but they only advertise the board as having four.

So is that board you are buying really dual-core, quad-core, or octa-core? Maybe not! Buyer beware.

The number GPU cores can be really important, or not. If you can get an OS with good drivers installed and the system tweaked up properly to use them, or you have some specialized GPU software to make use of them, they can make all the difference in the world. And if you end up using some other OS, and have no GPU support, it may not make any difference at all.

Finally I would like to touch on cases. Generally for any board, you'll probably find a wide variety of really cool cases to use, but one thing about cases is, they trap heat, and these little boards, they don't like heat. As a matter a fact, they take their temperature

constantly, and they can adjust their speed in fractions of a second to adjust for heat. It's called throttling. It is a good thing, because it keeps your board from getting damaged, and it is a bad thing, because it slows your board down. As a matter of fact, a really fast board that heats up really fast, may end up running slower than your average run of the mill board.

Cases are a personal decision. Some people go with fancy ones, some people go with no case at all, and some people go exactly the opposite direction and start adding heat syncs and fans to their boards, so that they can run them as fast as possible, sometimes even faster than they were designed to run. That's called "overclocking".

OK! I hope our gurus have caught back up with us, because it's time to move on to the next part of the show. I guess if I was a true professional, I'd be calling this a monologue, but I do believe in the truth of advertising, so I will just call this what it is, a bunch of single board computer rhetoric.

I'll just climb up on my soapbox and get started. Here we go!

Turns out I'm a little bit depressed about the state of the operating systems for these little boards. I try out a lot of the third party images that are available. First off, it seems like finding the password for given image usually requires a Google search. Few actually put the password anywhere close to the download link. I just spent twenty minutes going around in circles over at raspberrypi.org, in a test to see just how easy it should be to find. It's not.

Seems like the password is usually buried under ten pages of instructions of how to burn an image to an SD card. Ya know, that part's easy, and after you have done it a couple of times, you are an expert. And you tend to just skip that part, and just want to get on to booting your board.

With most of these images, I am really freaked out with all preconfigured user accounts, preconfigured passwords, premade SSH keys, and pre-installed software. It doesn't not have to be that way. And it causes trouble too.

For example, every Raspberry Pi in the world running Raspbian, starts out with a user account of Pi, and a password of Raspberry. Sure you can change that, and you should, but as soon as you do, programs like that fine raspi config program you use to configure up your board is probably broken, because it is hard coded to believe that no-one ever changes their user name for the account, and that seems like it is the result of some very poor coding, built upon the very poor concept of setting up a Linux system.

I think the only reason why a user account should ever be named Pi, or ODROID, or even alarm, is because the admin setting up the system, input that user name during the install, because they were asked about it.

And preconfigured SSH accounts? My brain says that if someone really wants a server, then let them turn it on.

And the same goes with resizing the OS to fill SD card. Some of these installs are just sloppy. I have seen many a partitioned SD card get resized for me, wiping out my stuff. I don't like that!

You know, they can ask about these things! There is easy ways to code these things up to happen, or not. And preferably by default, the probably shouldn't.

Auxiliary software installs should only happen after asking. The default should be zilch. Just ask me!

Would you like Kodi with that?

I honestly wonder if sometimes if this stuff is added by the marketing department somewhere. Libra office is a given. Let me decide!

You know I complained recently on a manufacturer's forum about an update that rewrote my boot.ini and fstab without warning me. It ended up booting somewhere else, wrecking total havoc around here.

You know that they replied that I should be happy because I was getting an update for free. And I asked them about that! I said, "This is an official manufactured supplied image. Am I wrong to think that I paid for that as part of the board? I mean does the manufacturer just make boards and then expect the open source community to be there to provide official images and updates free of charge, and then let the manufacturers stamp their logo on it and advertise it? Or are these guys actually getting paid for all their hard work and support?"

I certainly hope so!

Ya know, they didn't come back with an answer, and I didn't like that!

I saw an image labeled Ubuntu 16.04, but it was modified and came with a lot of stuff added. Is it still Ubuntu 16.04 after you modify it, add this, that, and the next thing, with Kodi on top?

I think not!

Perhaps there should be some sort of labeling standard, and be applicable in at least 37 countries and Antarctica.

It should come with a label like we label things here in the U.S. in the grocery stores "organic". It really does not mean that much, but at least you know that it's not made of alien moon dust or something.

If an image says it's Ubuntu 16.04, then I feel like that it should be a clean Ubuntu image, and nothing more.

Now speaking of labeling, I was looking at the latest DietPi images, and they sure seem interesting.

I've been following it since it's beginnings. It is kind of like tasksel, but perhaps maybe a little more specialized to single board computers. Tasksel, (I think that's how you say it), is a Linux package that can automate the installation of many different setups, like everything you need to make a web server for WordPress, or whatever.

DietPi's tag line started out as "Lightweight Justice for your Raspberry Pi".

Webster's defines "justice" as "the result of using laws to fairly judge and punish crimes and criminals"!

I did not know I needed any justice for my little Raspberry Pi's!

DietPi has moved on to support other boards, so they changed the tag line to be:

"Lightweight justice for your single board computer".

Well I don't think that I have any boards that need any justice!

I did have a Banana Pi M2 and M3, but I got rid of those boards, and I don't think I am going back.

I don't know. DietPi sounds like it should be a minimal OS, put on a diet.

But what does that give you? Less than a minimal operating system?

That doesn't sound good!

Now that they've moved on to support other boards, why the name "Pi"?

Why not "Diet Debian" or some such?

I mean I do get it, that it is all about marketing and everybody wants to ride the coattails of the Raspberry Pi. But that damn marketing department is controlling things again, right down to the name "diet".

I mean, to me DietPi comes up as too much stuff preinstalled anyway. I gotta uninstall stuff after I install it!

It auto updates, it auto expands, it adds Dropbear? And a web-server? I can't even find a "no" option for that. You're gonna get a web-server!

Some of the options seem unclear if they are going to install or not. And there are some menu loops that seem to demand that you install stuff. It will tell you that "foo bar is not installed" and ask if you would like to install it. Then they give you this menu, the only choice being, "OK install it" or "go back".

There is no escaping it! I mean, that's not asking me if I want to install something. That's demanding it!

To me, DietPi seems to violate every rule of a diet install. It is made for people that don't know Linux, that don't care to learn Linux, yet they want to run web-servers and the like. There must be nine different web stack server choices that it can install.

Believe me, you need to set up a web-server stack, you need to learn how to do it! And it is pretty easy. Web-servers are not something that you should let somebody else set up for you and then just forget it.

I don't know. DietPi just doesn't seem to a diet image.

Seems to come with a carbohydrate laden menu that's loaded down with every known sugar snack you would ever want to bloat a system with.

And the only thing that's "Pi" about it is one of the nine boards they support.

Perhaps it is all about optimization, like these guys know more than the developers on what to yank out of an install. As a developer, I understand some of that, and as a developer, it makes me nervous.

It is like getting one of those cleaners for your OS. On the one hand, you hear folks like DoorToDoor Geek, promoting Linux as a nice clean operating system. And then you hear of them suggest getting a cleaner. Ya that might take out some of the unused language files, then again, I have tried some of these cleaners and ended up with an unbootable system.

I don't know who to believe and on what day of the week to believe um on. I really don't know.

DietPi sounds like marketing to me.

If security matters to you, it seems like it would be a bad idea to use it for a base OS install. I mean, it might be good to quickly try something out, but would you really want to trust a point shoot interface to install a bunch of software and servers and stuff on your computer? You need to know what is going on underneath the hood! This isn't something that you just hand over to somebody else and trust them on it. I don't think!

Perhaps I am just taking this just a little too seriously. If that's even possible these days? DoorToDoor Geek highly recommends the DietPi on his show, and he is a big Linux guy, he knows great security practices, he knows how Linux works, he's got thousands of followers, many of them are highly respected, and nobody else is saying anything. So it might be OK?

I guess I wish they separated it out a little bit more.

As I understand it they have got a pretty smooth operating system, and we certainly need those.

Maybe I just error on the side of caution. Or perhaps I am just crazy! I mean, I hear all the lectures on good security practices these days.

I want to think that these little single board computers deserve as much respect as we give to any other server, or desktop, or laptop that we put on our network. And somehow, it just seems like with these single board computers, good security practices just go out the window.

Now I will say I found DietPi provides a really nice operating system, and it makes it easy to install some packages.

But at the end of the day I have to question if it is a good idea for an OS install to start out with the default installation of complex server software, and I fail to understand why that software has to be installed by default in the first place.

New users choosing DietPi, may not be knowledgeable enough to even understand what is going on, or be knowledgeable to properly maintain the system. And it's easy enough for developers to put up a prompt and ask the user "do you really want to install that"? Personally I find no excuse for it, I think that's the first thing that should have gone in there.

Now what I really want to ask is: "Why aren't we hearing more about this kind of stuff? Who is steering this ship? Does anybody care that we are heading for the rocks?"

I would like to think that the responsible members of the publishing community would be speaking up on these matters.

Then again, maybe I am crazy! Tell me what you think!

And it is interview time!

Today, we are honored to have as our special guest, Igor Pečovnik, the creator and chief architect of the armbian operating system.

I'll tell you what. If you are running Linux on an ARM board, your probably running some of this guy's code! There must be countless third party images out there that end up getting their start with armbian, or entirely based on armbian, and there's some boards out there that might not even run if it wasn't for this guy and his team. We owe this guy a huge debt of gratitude, no doubt, and if there is ever an ARM board project worth supporting, this is the one!

Now, you might not realize it today, but this interview may become a piece of history!

This was recorded earlier, so without further ado, here we go!

TJoe™: Here we are with Igor Pečovnik, and he is the chief architect of the armbian universal operating system that's used by many ARM boards. So what got you into computers Igor?

Igor Pečovnik: When I was a child. I got this ZX Spectrum. It was an era of home computers in the '80's. Yeah, it was '85 or less, '83? I do not know exactly, but ZX Spectrum was my first computer that I owned actually. Then I started with, Z80 I was playing with but I did not own it, my neighbour had it, and we had it in our school, so.

TJoe™: Now when did you start out with armbian?

Igor Pečovnik: With the project I started in the fall of 2013, but it was a project that I hosted on my personal website and GitHub. It wasn't named armbian at that time. That happened one year later, let's say, something like that.

TJoe™: So you got into building operating systems originally for ARM boards?

Igor Pečovnik: Yeah, yeah. Actually I owned an ARM board, and I figured out that the operating system, the support is really really bad, lousy, I would say. That is why I just started, it was my personal motivation to have some nice operating system.

TJoe™: I was looking at your download page here, and I lost count at how many boards you support. How many boards are there here?

Igor Pečovnik: A little more than thirty. I am not sure how much. Some of them are still in the development phase, and they will be

supported in a week or two. I don't know.

TJoe™: How do you decide what board that it is that you are going to support next?

Igor Pečovnik: Actually there are no special rules. We usually... OK, this situation changed. At the beginning, I need to buy a board. I need to bought it and then I start to play with it. That was, lets say one year ago or something like that. Now it is a little bit different since manufacturers usually send samples, usually before they release the board. So I got them, and some of them never came out of the drawer, that is also possible, because some of them are supported so poorly, so badly, no-one is working on it, and I cannot do it alone. It is impossible, so.

TJoe™: I think you mainly support three different types of ARM chips, is that right?

Igor Pečovnik: I count up to ten different chips that are different enough that they need to have their own kernel or U-Boot. At least up to ten we need to handle, but they are similar. So, we could optimise or rewrite the code to nail down to let us say six or something like that.

TJoe™: I noticed that you have had a new release of armbian. Want to tell us a little bit about what is going on there? What is new?

Igor Pečovnik: It is one of the major releases actually, we fixed quite a lot of things. First of all I have to mention we pushed Docker support to all boards which had kernel hardened 3.10 which is the minimum for Docker. What else? Then we added ARM64, bit support for board that came out was ODROID C2, and Pine64 is also on the way. It will be released in a matter of weeks. It's actually, it's already working, but I am not aware of exactly how far really we've come, because I am not working with that particular board. There are too many boards now so I cannot work with all them. It's a core team now let's say few people which are sharing this development, don't worry, there is strong community support. We cannot work without it. They are supplying patches, and ideas, and also they are adding boards, some cloned boards mostly, because the basic board support is quite difficult to add at this point. We have to... there are too many parameters which you have to consider how to bring things to armbian built system. So, this is a bit complex. But when some board is added, clones which usually came not long after. Let's say Banana had a lot of clones, actually all boards eventually got some clones which are not that much different.

TJoe™: armbian is like totally open source? And you encourage your users to build it from scratch is that right?

Igor Pečovnik: Exactly yes. Our build system is oriented that it grabs the code together from GitHub. So, actually it builds from a single C code down to bootable image. So we take care of everything what's in-between. We usually don't, I think we don't have any binaries included. Except OK firmware(s) for those wireless chips, those are BLOBS. The rest is open source.

TJoe™: So how difficult is it for a new guy to come in and build them up an armbian system from scratch.

Igor Pečovnik: Ahh you have to try it. ha ha.. I think you are too skilled already. I think that you are more skilled then, our tools are designed at entry level Linux user could use it. You boot to host and you run it inside, on the PC let's say, or with a virtual machine or whatever, and our script makes the rest. So, you have menu driven system, you select the board, and you wait half an hour, or, I don't know, it depends on your computer, and your image is done.

TJoe™: You seem to take security really seriously. I want to thank you for that. I noticed that almost every distro for all the ARM boards that I have gotten have premade user accounts, and I feel like that that is a security issue that is going to come back around and bite most of the manufacturers. People are going to figure out you really don't want something like that. Yet, armbian seems to be one, if the only system out there that doesn't come preconfigured with user accounts. Do you want to tell us a little bit more about your security model?

Igor Pečovnik: I think in the early days of building images, you could find some leftovers. Because companies they build up a system on the SD card and when they were satisfied they made a snapshot of that image, and that became a downloadable image. Of course sometimes they forgot some user settings, some passwords, or whatever, inside it is was quite funny. But with our approach you cannot actually, it can not come to that kind of things, because we built, the script builds everything from nothing. People, I said others, other companies use this approach. Not at this level. They still use packed root file system, prepacked file system. They glue images together, and we don't do that. We bootstrap. We compile everything, from scratch.

TJoe™: I think that it is good that you didn't go the extra step to throw all that stuff in there. I think it is wonderful to be able to start out with a nice clean image, without premade user accounts, no premade passwords, no premade keys, and what not. I think that you have done right , and I think that the other manufacturers are really going to have to go back to the drawing board, and are going to have to follow your example on that eventually.

Igor Pečovnik: They could simply use our tool I think.

TJoe™: I wish more of them would, because you have got a nice solid system. I am very impressed with it. While we are on security, would you mind commenting, or can you comment on that security alert we recently heard from the Allwinner forums?

Igor Pečovnik: I think that was a little bit exaggerated, because it was a security hole, but it was not a hidden back door, let's say. It was easy to find. Yeah it was some leftover from the debugging process. So, the code was not cleaned well before releasing. And I think nowadays because they are rushing too much and such things can easily happen. But the good thing is we fix it in a matter of hours or minutes, I don't know. When it was discovered we pushed an update and armbian is backdoor free.

TJoe™: I noticed that some of your distros, you have a desktop that comes with accelerated drivers. Want to tell us a little bit about your hardware support?

Igor Pečovnik: Yeah, accelerated drivers are actually also based on open source driver, it was developed some time ago, and it's so so supported, so not all codecs are supported, but eventually it works. Let's say on Allwinner 820, 810, and H3, and it is possible to run an accelerated video with an open source driver. And also Mali which is good for gaming for example, or I do not know what all they are using it for. We also added Mali drivers, so from this point graphics or desktops are much more usable. But that is just Allwinner for now. But we could add on other boards. But for other boards there are usually binarys so it is a little bit complicated.

TJoe™: Is it really tough to add in drivers for a user if they start out with like the server edition?

Igor Pečovnik: It is. So it's not a simple task, but when our work will be packed it will be very easy to install. So currently it is not packed so you need to download desktop version if you want to have all of this functionality.

TJoe™: I think a lot of people miss the fact that the GPU's usable for so much than just accelerating the desktop. These little boards they're driving cars, they're flying planes, they're doing all sorts of things that we never even dreamt of, and a lot of that is made possible by, you know, the accelerated GPU, and it's not being used to accelerate a desktop it's just doing calculations in parallel. The GPU on these little boards are fairly powerful, and they certainly are usable for a lot of things besides just making your desktop snappy.

Igor Pečovnik: Desktop is the most what we will use our GPU's, so.

TJoe™: I have taken your server image and placed on a number of our boards different desktops on it, and it seems to run as smooth or smoother than some of the manufacturerâ€™s images that claim to be accelerated. I am very impressed, you have got a nice lean system and it runs very solid.

Igor Pečovnik: Yes, there are many optimizations. I made many of these optimizations. I did not just extract some system and, they usually just put some Debian or Ubuntu root file system and glue it with a kernel. But that is not enough if you want to have a well working image, you need to tune up the file system operations and also the CPU clocks and that kind of stuff. So we fine tune all boards.

TJoe™: That is one of the most solid images I have ever run, and I have run them on quite a few boards. I think that you support every board, except for maybe the NVidia Jetsons, I think those are the only boards that we have that aren't.

Igor Pečovnik: I told you we don't support Raspberry Pi.

TJoe™: That's OK, there is enough support for that. I know that you get in there and you support boards that a lot of people would not even have a choice in the matter. I know that the Banana M2 and M3, have been problematic boards.

Igor Pečovnik: Yes they are. M2 hmmm. They are now quite usable for server case, and I think for light desktop it so is usable up to now. But M3, we do not have support for M3. Actually I just got the board last week, I think, yeah. The main line support is so so. The kernel was never cleaned up. There is a lot of work to be done.

TJoe™: Now I know that you get a lot of support from your users, but what about the manufacturers? How good are they at lending support to you?

Igor Pečovnik: They send a board, one of them sent some cash as a donation, and that is about it. I mean we have some relations, but it is not that deep that it could be. We could do more let's say.

TJoe™: Do you generally find that the manufacturers somewhat cooperative?

Igor Pečovnik: They are, they are.

TJoe™: I'd think that they'd really would want to support an operation like you've got there. You make a great product, and I know that it is not easy, but it is also not cheap either to do this sort of work. Now, do you ever get users that kick in and try to supply boards or...

Igor Pečovnik: Yes they do. Actually I also got a few boards I got from people, from community. They saw that I was struggling, that I

was solving problems on the theoretical level which is usually hard. When you have a board here you can try a few things, and you quickly find a solution. But, if you tell people try that, and try that, and try that. It takes a lot of time before you get some useful feedback.

TJoe™: It must be really expensive to support all those boards, take an awful lot of work. I can't imagine keeping up with all that. I have got a few boards. We've got 21 boards here, and I can't even keep them updated. I cannot imagine you creating an operating system. I don't see how you do it.

Igor Pečovnik: I am not the only one doing it. There are a few people around. So, everybody is doing something, so.

TJoe™: You have got a good team, and I have noticed that they get in there and they help support the users too. It seems to be a very friendly forum that you've got.

Igor Pečovnik: Yeah the community is really good. People are polite. So more or less, we do not have problems. It's technically oriented of course. So a lot of information is shared.

TJoe™: Your a volunteers supported organization, you have got people helping out. What sort of things do you need people to help out with?

Igor Pečovnik: Support, to help to code. I am always happy that someone takes some notes when we talk about coding. There is a lot of stuff which is waiting to be done for desktop, so in this area we would certainly appreciate help. Then it is help with some donations and so on, that we could pay our bills let's say. I am more or less full time here now, and friend is also doing almost full time, and so it is costly.

TJoe™: It is a great project I have always promoted it as a project very much worthwhile in supporting. OK I have got a personal question I've just got to ask you. What is your favorite board?

Igor Pečovnik: I think the old Banana, the original Banana would be one of the most interesting boards. Even if it is not the best one, but it is simple.

TJoe™: That's a good choice. Its one of my favorites as well. So how can somebody contact you?

Igor Pečovnik: armbian.com you have at the bottom contact center there are, there is my email. You have a technical question, go to the forum ask there, I will answer there or somebody else will, but for anything else it's email. Let's say for business communication. Whatever, that's not related to technical issues.

TJoe™: I would like to thank you for taking the time to talk with us. We would like to have you back, and maybe become a permanent fixture here at the Single Board Computer Show™.

Igor Pečovnik: OK, Joe. Thank you for the invite, and for all your efforts. I know it is hard to make all this happen. Yeah, I would like to be here more often as well. It was fun talking about things we both like to play with. So I wish you good luck, and all the best with the show.

TJoe™: Great, thanks so much Igor. We certainly appreciate it.

OK! That was Igor Pečovnik, chief architect and creator of the armbian universal operating System.

Wow! What an interview! You know, we sure owe that guy and his team a lot!

You remember when I asked how easy it was to build armbian from scratch and he replied, "Just try it'? Well I did!

Wow! Was I ever amazed!

It is no wonder that so many operating systems are actually based on armbian.

I downloaded the script into a Linux VM. Fired it off. They asked me what kind of board I wanted to build it for, and If I wanted a desktop, or minimal system image. In no time at all it built me up a complete system from scratch. All the latest security updates, compete new kernel. Everything! Wow! How easy is that?

I find that just amazing. You know, I built some images for some of my boards using different versions of Debian and Ubuntu. But then I went into the build script, and tried playing around a little bit. I changed the kernel config line to say yes, ran the script again and wow! I couldn't believe all the configurable choices that became available. Seemed like every option under the sun, moon, and stars just lit up.

It started to become clear why certain things in some of these other images, never really worked. They aren't enabled in the kernel! Man there's some cool stuff in there too. I mean, have you ever wondered why that DVD would never play on your ARM boards? Support may not have been enabled in the kernel!

You know with armbian it is just a simple matter of selecting what you want, and telling it to build you one. You should go over to armbian.com and check it out, that's A R M B I A N dot com.

You know you don't have to build your own image, they supply prebuilt ones, but Igor and his team have made it so easy to build your own system, I would encourage even the complete newbie to give it a try. It really is that simple.

Now I have to admit, they've got a pretty fine desktop, but I like some other ones too. I have found it is really simple to add in your favorite desktop to either their basic vanilla image or their desktop image. Now what could be better than that? Get your own version of Linux built from scratch, it is all nice and clean, nothing else added, built up from the ground up your way.

And you should go on over to armbian.com and check it out. And think about getting involved with the project too, and if you do be sure to tell um TJoe sent you.

OK! It's project time!

The next part of the show is at your own risk. We accept no liability for anything you get off the show, and if things get messed up, we are not going to be responsible for it. OK?

One of the most popular options included with an OS install, using up all the space on the SD card and giving it over to the OS, and to that, I say: wait a second! Do we really want to do that?

Linux is a really lean operating system, and it takes up hardly any space at all. Like, you can do a full blown install of Linux and add a ton of apps in with just a handful of gigabytes. Instead of giving the whole the SD card over to the OS, we could actually put several different OS's on that one single SD card, and then boot between them.

We could make a little common area on this SD card to store stuff, and all the different OS's we boot to would be able to get to it.

We could even partition up the cards to make it easy to do back-ups and snapshots, and if you are a fan of using swap space on your OS, we could have all the different operating systems use the same chunk of the SD card, and conserve disk space.

Sound like a good idea to you? That's what we do here at Code4Sale SwampWërks™ everyday, using a simple system that we have appropriately named "BootCrazy", and it saves our bacon!

See. We are constantly installing stuff and experimenting with OS images, we're just destroying these things. We have got a lot of boards, and we boot a lot of different images. The boards are stacked up, and the SD cards are hard to get too, and we don't have time to sit around and burn SD cards all day long. So we invented BootCrazy to get around all that. All I can say is that it works.

Now BootCrazy is going to be an ongoing project here at The Single Board Computer Show™. Today we are going to start out by describing how it works, and tell you where we are going to take it. We are going to post more to our website as soon, as we get it up and running. Scripts or not, you are going to be able to use this system because we are going to roll up our sleeves and learn it.

Now, I don't want this to be some sort of "download and install it" sort of project. I am from the old school. For it to be a true project, you have to actually get involved, and learn how to make it work. That's what a real project is, and it's going to be easy.

Now, the BootCrazy system will work with almost all of the Linux operating system images out there. You will need a USB SD card reader, and two or three SD cards, the larger the better, and at least one or more of your favorite Linux operating system images. Here is how it works:

Almost all of these Linux operating systems out there use a two partition system. Now each partition is a separate area of the SD card. There's a really small boot partition, and that holds your boot files. And a larger partition called "the system file partition" or "root file system", that holds your OS.

Now the board boots whatever it finds in the first partition of the SD card, and that's where it gets the location of the root file system to run. Now there is generally no rule on where the root file system has to be located. It can be almost anywhere. Somewhere else on the SD card, it can be on a USB drive, an SSD drive, even out on the network.

The only thing that really matters is that the boot files are in that first partition on the SD card. So what we're going to do it split the SD card up into several partition areas.

We'll have an area to hold the current boot files, that will be the first partition. Then we are going to make a common area to hold any

data that we want to be accessible by all the OS's that we are going to boot from the SD card. And finally we are going to put two partitions for each operating system we want to boot on the card. A really small one for its boot files. And the other one is going to be its root file system.

The idea is pretty simple. For any OS we want to boot, we're just going to simply copy its boot files to the first partition, and reboot the board. There's not a lot more to it than that. But we can make things a bit more fancy if we want to, and we plan to do just that!

We are going to add some little scripts to the common area to let you boot between the different OS's all at the click of a mouse. It's really that easy.

Now, to make it work, we do need to do a little bit of planning.

Now the first thing is we have to choose a disk format to use, it turns out there is many to choose from. But most all of these boards use something called a "master boot record disk", we'll just call it MBR for short. And it can do exactly what we need, so we are going to keep things simple and just use that.

Now a MBR disk has one little issue, it can only have four primary partitions.

Now we the ability to have more than four partitions, but we only need one primary partition to boot from. That's OK because MBR disk will allow us to create something called an extended partition. It is like a container partition that can hold even more partitions, but what really matters is that the first partition is always large enough to hold all of the boot files that we will be using for any OS that we want to boot too. I make mine 256 megabytes in size, and so far I have never seen anything that uses anything more than that.

Next in our planning stage, you may want to have a dedicated swap area.

Now, there is a lot to be said about swap space, especially when it comes to SSD's and SD cards. It's a personal choice, and I have seen many arguments break out over its use.

Now, I'll tell you what I do, so long as you promise to allow me to have my personal choice. I don't need a lecture, so please do not give me one. And I encourage you to seek out the facts and make your own personal decision, and I won't lecture you on it either. OK?

Here is what I do. I use a dedicated swap space in the second partition. Generally equal to the size of the on board RAM. Some operating systems like Raspbian, come with a file based swap system called "DH Physical Swap File", and if so, I will turn it off and uninstall it. And some come with no swap enabled at all enabled in the kernel. You can't add swap to those. But, I go ahead and add a swap partition anyway, because you never know when that might change. Since we are going to be putting multiple OS's on these SD cards. Well, at a later time we might add a new version to the operating system, and it might have swap enabled.

I am not going to get into why I do it this way, or even try to defend some of the seemingly legitimate reason I have for doing it this way, and I am not going to listen to any lectures, so, you can make up your own mind, and we can all play nice together on it.

Now next I add a third partition called "common", and that's where I put the files I want to access from any of the OS's I'll be installing too. You may want to skip that, but I feel like it's kind of important, because among other things I put there, will be some scripts that will

allow me to easily copy the boot files to the first partition for any OS I want to boot, and it reboots the board. This partition doesn't have to be very large, mine are usually pretty small, but it does need to be large enough to hold whatever it is that you want to be able to share between the OS's. I use a minimum of 256 megabytes, but there is no upward limit. You might even want to make this common partition, and possibly even the swap partition, last on the SD card to make it easier to resize. It really doesn't matter to me. The location of all but the first partition can be according to your own personal taste.

Finally I create an "extended partition" to hold all the other partitions we're going to put on the SD card. Now I usually make it take up the rest of SD card, unless I am using an SSD, and then I will leave about 10% at the end for an over partition area.

Now to create all of these partitions I use a program called GParted. Its a GUI program so you can run it from a desktop, and it's really easy to use. If you prefer we can also use command line tools to do the same thing. Now I am going to put this all in the show notes, as soon as we get some show notes, and then we will cover the finer details on using these programs in the next boot of The Single Board Computer Show™.

Today we're just going to tell you about BootCrazy, and where we are going to take it. If you're feeling adventurous, you may want to go ahead and get started, and you'll need to install GParted on your ARM board, along with mtools, and dosfstools, to give you the support you will need to label up the different partitions, and work with the different partition types.

So you need GParted. That's G P A R T E D, dosfstools D O S F S T O O L S, and mtools M T O O L S.

OK!

So as I said we will be covering the finer step by step details, on getting all this done on the next boot of The Single Board Computer Show™. Here's the overhead to give you something to think about in the meantime.

Here's the plan. We will install a couple of the Linex OS's to two different SD cards and then use GParted to resize all the different partitions to fit, and combine all our different operating systems onto one single SD card. Finally we will add a couple of cool shell scripts to our common area, that we'll use to copy the right boot files to the boot partition for the OS we want to boot, and then reboot the board.

Pretty cool eh?

There's one last little detail we'll need to do. Since the partition's position of whatever OS we are going to be booting will have moved, we'll need to do a quick edit of the boot files and the fstab file to point to the right place.

OK! So where are we going to take this? Well we'll be adding the ability to update the files we copied to the boot partition, so whenever we get new kernel updates they don't get overridden, and the ability to utilize compression for backup and restoration of system images, boot partitions, and user directories.

So how many OS's can you put on an SD card?

Well of course it depends on the size of the SD card and the size of the OS partitions. And you are not just limited to using an SD card

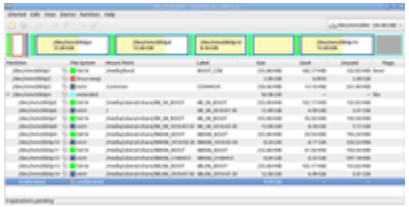
either. Here at the SwampWërks™, we use the system boot all over the place. Flash drives, SSD drives, and even over the network.

So how many can you get?

Well, with a 64 gig SD card, we often get between four and six OS images on a SD card, and I have a couple of boards with SSDs that have over forty images on them.

Once you get the SD card initially set up to multi boot, it also gets a lot easier to add new images to it. This system's really sweet, especially for backing up. It's really nice to be able to instantly reboot to almost any image or OS, all with just a mouse click, and if you do a lot of experimentation and you know you are going to wreck the OS image, man, this system is for you! We commonly use it to take snapshots of the system right before we do a major update.

I really like using the BootCrazy system to compare manufacturer images. For example, it is really interesting to be able to instantly boot back in time and see how the ODROID C2 Ubuntu images have evolved since the board originally shipped.



BootCrazy - Illustrated Multiboot Disk Layout - GParted
(Click image for larger view)

D.O.

Oh my!

I guess it is time to wash up from our project, put away our tools, and move onto the VPS section of the show.

If you're not familiar with virtual private servers, here's how it works.

It turns out that you can rent time on a virtual computer for almost nothing these days. In short, you go to a online control panel, you tell it what sort of computer you want, like how much RAM, and how many cores, and many network cards you want, and how you want it preconfigured. You hit the start button and, in less than a minute, there it is, ready to log onto. You can keep it going as long as you want, and you can make copies of it, duplicate it, destroy it when you are done. And it is cheap, like really really cheap. It's so cheap that if you decide to keep it running for less than an hour they might not even bill you for it. Because for all that, your bill might not even add up to a single penny.

But why would someone want to do this? Well let's say that you started thinking about how cool it might be to shoot together some sort of WordPress site, or some sort of fancy server software setup, an e-commerce site perhaps, add a forum, there's a lot of choices for all

that. Perhaps you would like to try a few? Then you might start asking, well how does this work with that?

You know, with real server hardware, that might get expensive really fast. It might take a lot of time too. But what if you could just push a button and it magically appeared. You could play with it for as long as you like, and put it down whenever you are done. And all it costs you is less than a penny per hour to play with. Well that's what virtual servers are all about. At least it should be, if you are at the right place. I will tell you, virtual private server hosting companies are not all created equal. At all.

Let me tell you a little bit about a company called Digital Ocean.

Now. They're not currently a sponsor of the show. In fact, I suppose it is the other way around. So far we have had a great experience with Digital Ocean. We have no other choice than highly recommend them as the best darn solution for virtual private servers we have ever come across. And so far, they may even be the most upfront and honest company we have dealt with, so far.

You know, I'll tell you, we really like how their jib's cut. Businesses arn't supposed to make you feel that way. Somehow we always come away feeling like the folks at Digital Ocean are shooting straight with us. And I'll tell you something, even if you are not interested in getting a virtual private server, if you want to learn about setting up and working with servers. You know, perhaps for that little single board computer you have. You can't beat the great resources over at Digital Ocean for first rate learning experience.

You don't even have to be a customer to take advantage of all the great technical articles available at Digital Ocean, to help you get that job done right. Now I am serious on that.

I am going to tell you about our promo at Digital Ocean. It's not because we might make anything of it, I could really care less about that. But, it does say that as of right now, you get like a ten dollar credit by using it. Now that might change some day, and this net cast might be forever, so I have added my disclaimers. You should be able to tell that I am shooting straight with you. This isn't about money, honestly. So far Digital Ocean really rocks.

Now, I do remember a promo link is how I got started at Digital Ocean, and if you would like to give it a try, well, there you go.

Now, for what it's worth, I just logged into my account to see what my promo link would get you, and noticed I have had a virtual server running over there for over two years straight with 100% uptime. Wow, I'm really amazed with Digital Ocean so far.

So here's the promo link, and again if you are not interested in virtual private servers, you are setting up any sort of server for your single board computer, you really need to go check out their community for some fantastic resources.

I know it is our number one resource for getting serious answers for setting up servers here. So far.

If you want a free credit, check out our promo link at code4sale.com/do.

OK and I do want to make it clear that if you use the promo code and you stay with Digital Ocean and you spend some amount of money with them, we may get some money back, and this is not meant to be a commercial for Digital Ocean. This is truly a great resource that is our number one resource that we use around here. We have had a good experience with them and we are sharing that with you. Just know that discussing virtual private servers and what you can do with them is going to be a regular part of the show, and

we'll be bringing you more on virtual private servers, on the next boot of The Single Board Computer and Virtual Private Server Show™.

So, there you go, just know, hey, we love Digital Ocean. Really, they seem to be awesome folks.

Well what do you know, we have reached the part of the show that everybody has been waiting on. The giveaway for the ODROID C1+ ARM development board to one of the lucky members of our g+ community.

Mrs. Single Board Computers, aka Little Kitten has preselected one of the members of our g+ community as the winner.

She printed off a page of all the members, cut them up in little squares, and put them in a basket, and pulled out the winner. So here it goes, I am going to open up the envelope now. The winner is, I hope I say this right, Ingmar Stapel. OK Ingmar, you've got 14 days from the first day that we post a link to our show on our g+ community to get hold of us, and claim your board.

Let's see where Ingmar's from. MÃ¼nchen! Munich, Germany! Oh this is great! My family is actually from there. Well that's going to cost a few dollars to ship! I might as well deliver this board in person, it is a beautiful city, I love Munich I love visiting there. Well congratulations Ingmar! I hope you enjoy the board. Let us know the shipping address and we'll try to get this board shipped out to you ASAP.

Well what do you know? That brings us to the end of the first boot of The Single Board Computer and Virtual Private Server Show™. I would like to thank our g+ community, you guys rock! Don't forget if you are interested in helping out, drop us a line!

We can be found on the web at code4sale.com/sbc/. You can leave us a voice message on the How Stuff's Broke™ listener line at 850-659-2986 right here in the US.

Next show will be continuing on with our Boot crazy project, I can't wait!

OK until next time... Reboot!

Products mentioned may be trademarks of the respective trademark holders.

Some links provided may be sponsored.

If you would like to contact the show please visit us on the web at howstuffsbroke.com, or leave a message at (850) 659-2986, tolls and surcharges may apply.

The Single Board Computer Show™ is a production of SwampWërks™ Studios, Florida.

Executive producer Lynda A. Hecht.

Music by Steve Cherubino

Indexed Links:

Allwinner
Allwinner 820, 810
Allwinner H3
Arch Linux
ARM64
armbian
Banana Pi
BeagleBone
Code4Sale, LLC
CUBox
DietPi
Debian
Digital Ocean
Docker
DoorToDoor Geek
dosfstools
Dropbear
GParted
GPU
How Stuff's Broke™
Kodi
Loose Cannon
MacBook Air
Mali
MATE

Merriam-Webster
militaryaerospace.com
mtools
NetcastExtreme
NetcastReserve
NVidia Jetson
ODROID
OwnCloud
Pine64
podnutz.com
Raspbian
Raspiberry Pi
Tasksel
The MiniPC Show
The Pi Podcast
The Single Board Computer and Virtual Private Server™ g+ community
Steve Cherubino
Ubuntu
U-Boot
WordPress
Z80
ZX Spectrum

Show's Media Links:

Show's Website: https://code4sale.com/sbc/
This Show's Website: https://code4sale.com/sbc/?go=2016%2fsbcvps-0001-2016-06-05
Audio: https://shows.code4sale.com/sbcvps/2016/sbcvps-0001-2016-06-05.mp3?rev=0000
Show's Feed's page: https://feeds.code4sale.com/sbcvps/

Show's Media Hashes:

File: sbcvps-0001-2016-06-05.mp3
GUID: 1ACB0869-E058-4D9E-A0EE-2A26033157FC
crc32: be75b8f7
md5sum: 65f46a0897ad07b99cd6da553829f02e
sha1sum: 51cab6d8e443d44323b797179b3d3e36fec9ad02
sha256sum: 76ab8b7cad0547cc7cd8a432292c14327c35c3b3886ab2fa827c114740d0c3a5
