Die wirre Welt der kleinen Computer

Daniel Maslowski

tübix

Agenda

- Introduction
- Understanding hardware
- Looking at the SoC
- Tracking upstream
- Cool projects
- Finding a community

Introduction

So many options…

So many options…

There are hundreds of “single board computers” (SBCs).

So many options…

There are hundreds of “single board computers” (SBCs).

There are compute modules (SoMs) and carrier boards.

So many options…

There are hundreds of “single board computers” (SBCs).

There are compute modules (SoMs) and carrier boards.

There are add-ons, lots of form factors.

So many options…

There are hundreds of “single board computers” (SBCs).

There are compute modules (SoMs) and carrier boards.

There are add-ons, lots of form factors.

There are microcontrollers and application processors.

So many options…

There are hundreds of “single board computers” (SBCs).

There are compute modules (SoMs) and carrier boards.

There are add-ons, lots of form factors.

There are microcontrollers and application processors.

There are different instruction sets: RISC-V, Arm 32bit + 64-bit, MIPS…

So many options…

There are hundreds of “single board computers” (SBCs).

There are compute modules (SoMs) and carrier boards.

There are add-ons, lots of form factors.

There are microcontrollers and application processors.

There are different instruction sets: RISC-V, Arm 32bit + 64-bit, MIPS…

People often look at computers and ask:

Does/can it run “Linux”?

tübix

So many options…

There are hundreds of “single board computers” (SBCs).

There are compute modules (SoMs) and carrier boards.

There are add-ons, lots of form factors.

There are microcontrollers and application processors.

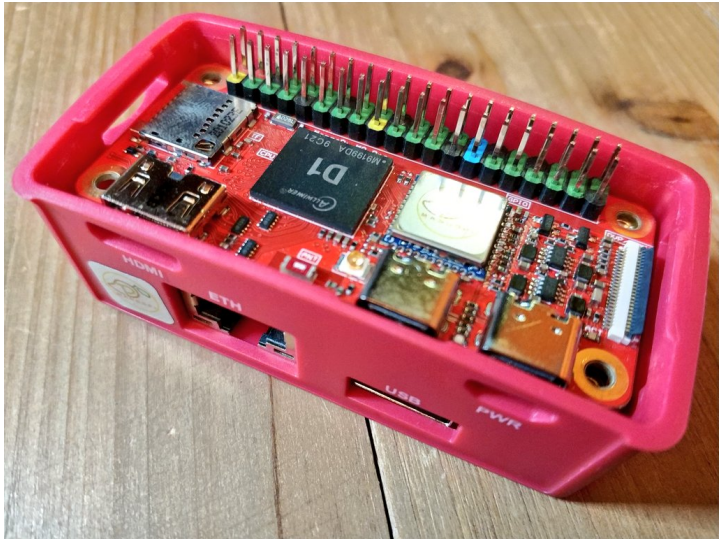
There are different instruction sets: RISC-V, Arm 32bit + 64-bit, MIPS…

People often look at computers and ask:
Does/can it run “Linux”?

That is a very tough question. Let’s see! :-)

tübix

Compatibility



tübix

Confusion

wiki.radxa.com/zero/downloads

Official ROCK Pi system images can also be downloaded from **ROCK Pi BaiduPan** or **Radxa Github Release**.

For user names and passwords please check the FAQ.

Tools

Description	Linux	MacOS
Etcher - A user friendly Image Writer	Linux 64bit | Linux 32bit	balenaEtcher-1.5.76.dmg

Official Images

	**Android 9**	**Android Install Wiki.**
	Ubuntu Focal	**Ubuntu Product. Install Wiki. Ubuntu Focal (20.04).**
	Debian Buster	**Debian Product. Install Wiki. Debian Buster(10).**

Third Party Images

	openSUSE Tumbleweed JeOS	**Build 2022-07-04.** **Install WiKi** **user:root password:linux**

Peripherals

Peripherals

Drivers and special component firmware can be very nasty.

- graphics / GPU
- audio
- wireless / Wi-Fi + Bluetooth

Peripherals

Drivers and special component firmware can be very nasty.

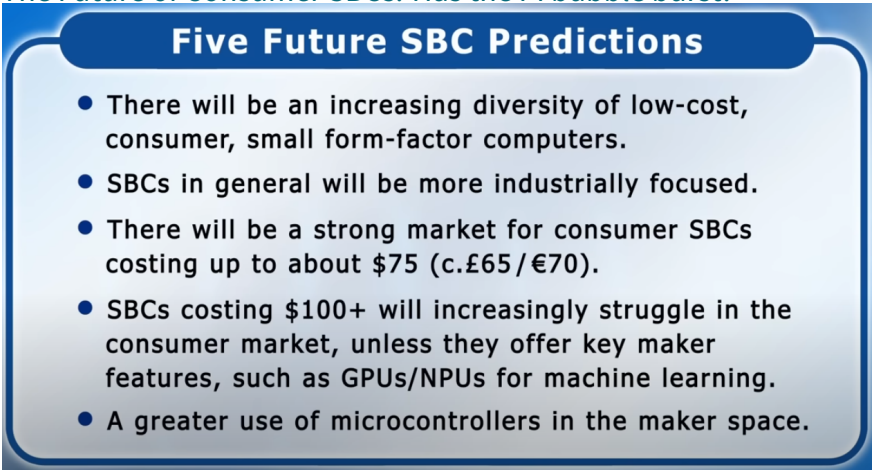
- graphics / GPU
- audio
- wireless / Wi-Fi + Bluetooth

Fun story:

We had to install `bluez-firmware` to get Wi-Fi working on a Radxa Zero.

Prediction

The Future of Consumer SBCs: Has the Pi bubble burst?

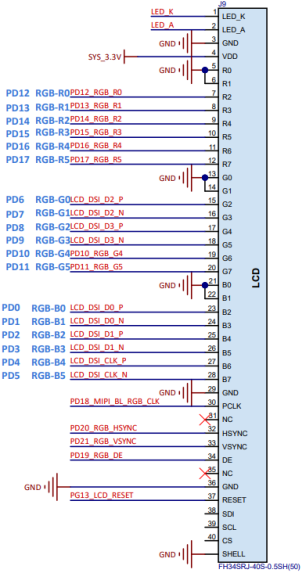


https://www.youtube.com/watch?v=Hjb3bx6vxnc

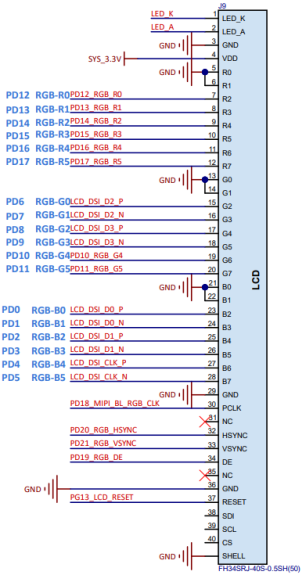
tübix

Understanding hardware

Reading Schematics



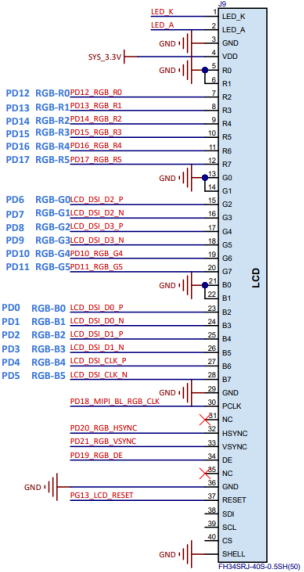
Reading Schematics



Why does my LCD not work?

tübix

Reading Schematics

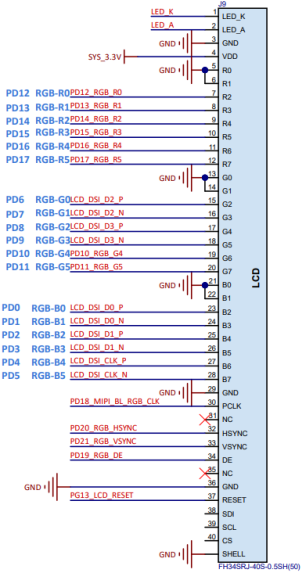


Why does my LCD not work?

Look very, very closely at the interface…

No.	Symbol	Description
1	VBL-	Backlight LED Cathode
2	VBL+	Backlight LED Anode.
3	GND	System Ground
4	VCC	Power supply for logic operation
5~12	R0~R7	Data bus
13~20	G0~G7	Data bus
21~28	B0~B7	Data bus
29	GND	System Ground
30	CLK	Pixel clock signal
31	DISP	Display on/off control
32	HSYNC	Horizontal Sync signal
33	VSYNC	Vertical Sync signal
34	DEN	Data Enable
35	NC	No connect
36	GND	System Ground
37	NC/XR	TP pin XR
38	NC/YD	TP pin YD
39	NC/XL	TP pin XL
40	NC/YU	TP pin YU

Reading Schematics



Why does my LCD not work?

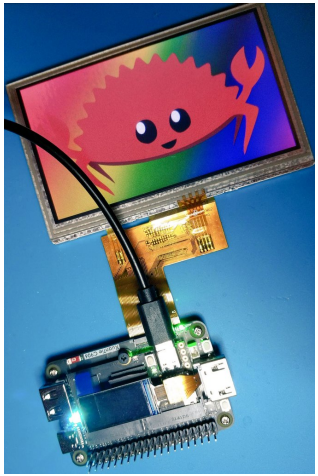
Look very, very closely at the interface…

No.	Symbol	Description
1	VBL-	Backlight LED Cathode
2	VBL+	Backlight LED Anode.
3	GND	System Ground
4	VCC	Power supply for logic operation
5~12	R0~R7	Data bus
13~20	G0~G7	Data bus
21~28	B0~B7	Data bus
29	GND	System Ground
30	CLK	Pixel clock signal
31	DISP	Display on/off control
32	HSYNC	Horizontal Sync signal
33	VSYNC	Vertical Sync signal
34	DEN	Data Enable
35	NC	No connect
36	GND	System Ground
37	NC/XR	TP pin XR
38	NC/YD	TP pin YD
39	NC/XL	TP pin XL
40	NC/YU	TP pin YU

Aha, Pin 31 enables the display.
But it’s not connected on my board.
Let’s fix it! :-)

tübix

Hardware Hacks



https://github.com/adamgreig/d1rgb

tübix

TRM / SoC manual / datasheets

TRM / SoC manual / datasheets

To understand a chip, you need its *Technical Reference Manual (TRM)*.

It may also be called *SoC (System on Chip) manual* or just *datasheet*.

TRM / SoC manual / datasheets

To understand a chip, you need its *Technical Reference Manual (TRM)*.

It may also be called *SoC (System on Chip) manual* or just *datasheet*.

Note that some datasheets may be only brief summaries.

TRM / SoC manual / datasheets

To understand a chip, you need its *Technical Reference Manual (TRM)*.

It may also be called *SoC (System on Chip) manual* or just *datasheet*.

Note that some datasheets may be only brief summaries.

Manuals are rarely public, often contain “confidentiality” notes.

They may have errata and/or be incomplete.

A Primer on Embedded Linux

So you want to build an embedded Linux system?

> *The first step is to architect your system. This is hard to do unless what you're building is trivial or you have a lot of experience, so you'll probably start by buying some reference hardware, trying it out to see if it can do what you're trying to do (both in terms of hardware and software), and then using that as a jumping-off point for your own designs.*

https://jaycarlson.net/embedded-linux/

tübix

Looking at the SoC

Capabilities

Not every SoC is general purpose.

Many SoCs are designed for narrow tasks, yet *seem* generic.

Common FruitPi SoC Vendors

Common FruitPi SoC Vendors

OrangePi, BananaPi, CherryPi, MangoPi, … you get the idea.

Common FruitPi SoC Vendors

OrangePi, BananaPi, CherryPi, MangoPi, … you get the idea.

Common FruitPi SoC Vendors

OrangePi, BananaPi, CherryPi, MangoPi, … you get the idea.

Loader Tools

`sunxi-fel` / `xfel`

`rkflashtool`

`pyamlboot`

tübix

Common FruitPi SoC Vendors

OrangePi, BananaPi, CherryPi, MangoPi, … you get the idea.

Loader Tools

`sunxi-fel` / `xfel`

`rkflashtool`

`pyamlboot`

Note: These compete on the multimedia device market, e.g., TV boxes.

tübix

Common FruitPi SoC Vendors

OrangePi, BananaPi, CherryPi, MangoPi, … you get the idea.

Loader Tools

`sunxi-fel` / `xfel`

`rkflashtool`

`pyamlboot`

Note: These compete on the multimedia device market, e.g., TV boxes.

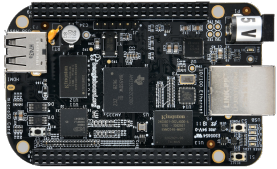
https://linux-sunxi.org/Comparison_of_chip_maker_openness

tübix

More SoC vendors

TEXAS INSTRUMENTS

AM{3,4,5,6}xx series
used in Beaglebone Black



NXP

i.MX application processors
used in MNT Reform laptop

tübix

Even more SoC vendors

Tracking upstream

Firmware

tübix

Firmware

a boot loader for
Embedded boards

https://u-boot.readthedocs.io/

tübix

Firmware

a boot loader for
Embedded boards
https://u-boot.readthedocs.io/

a fork of coreboot, with
C removed, written in Rust.
https://github.com/oreboot/oreboot

Firmware

a boot loader for
Embedded boards
https://u-boot.readthedocs.io/

a fork of coreboot, with
C removed, written in Rust.
https://github.com/oreboot/oreboot

tübix

Linux

tübix

Linux

Protocol	Location
HTTP	https://www.kernel.org/pub/
GIT	https://git.kernel.org/
RSYNC	rsync://rsync.kernel.org/pub/

Latest Release
6.4

mainline:	**6.4**	2023-06-25	[tarball] [pgp] [patch]	[view diff]	[browse]	
stable:	**6.3.10**	2023-06-28	[tarball] [pgp] [patch] [inc. patch]	[view diff]	[browse]	[changelog]
longterm:	**6.1.36**	2023-06-28	[tarball] [pgp] [patch] [inc. patch]	[view diff]	[browse]	[changelog]
longterm:	**5.15.119**	2023-06-28	[tarball] [pgp] [patch] [inc. patch]	[view diff]	[browse]	[changelog]
longterm:	**5.10.186**	2023-06-28	[tarball] [pgp] [patch] [inc. patch]	[view diff]	[browse]	[changelog]
longterm:	**5.4.249**	2023-06-28	[tarball] [pgp] [patch] [inc. patch]	[view diff]	[browse]	[changelog]
longterm:	**4.19.288**	2023-06-28	[tarball] [pgp] [patch] [inc. patch]	[view diff]	[browse]	[changelog]
longterm:	**4.14.320**	2023-06-28	[tarball] [pgp] [patch] [inc. patch]	[view diff]	[browse]	[changelog]
linux-next:	**next-20230629**	2023-06-29			[browse]	

https://kernel.org

tübix

Specialized distros for Arm

tübix

Specialized distros for Arm

Arch Linux Arm

https://archlinuxarm.org/

openSUSE

https://en.opensuse.org/Portal:Arm

Fedora

https://fedoraproject.org/wiki/Architectures/ARM

Armbian

https://www.armbian.com/

Ubuntu

https://ubuntu.com/download/server/arm

Specialized distros for Arm

Arch Linux Arm

https://archlinuxarm.org/

openSUSE

https://en.opensuse.org/Portal:Arm

Fedora

https://fedoraproject.org/wiki/Architectures/ARM

Armbian

https://www.armbian.com/

Ubuntu

https://ubuntu.com/download/server/arm

Problem: Many of these have specific images per board.
Why? (many reasons)

tübix

Cool projects

Does it have to be…

tübix

Does it have to be…

… a general purpose distro?

tübix

Does it have to be…

… a general purpose distro?
Make your own system!

tübix

Does it have to be…

… a general purpose distro?
Make your own system!

Frameworks

- Yocto/OpenEmbedded
- Buildroot
- OpenWrt
- AOSP (Android)

tübix

Does it have to be…

… a general purpose distro?
Make your own system!

… an application processor?

Frameworks

- Yocto/OpenEmbedded
- Buildroot
- OpenWrt
- AOSP (Android)

tübix

Does it have to be…

… a general purpose distro?
Make your own system!

… an application processor?
There are many microcontrollers, too.
In general, they are more open.

Frameworks

- Yocto/OpenEmbedded
- Buildroot
- OpenWrt
- AOSP (Android)

tübix

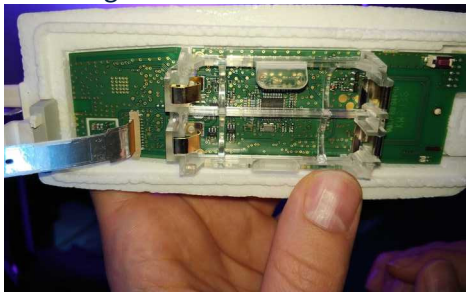
Does it have to be…

… a general purpose distro?
Make your own system!

Frameworks

- Yocto/OpenEmbedded
- Buildroot
- OpenWrt
- AOSP (Android)

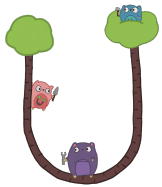
… an application processor?
There are many microcontrollers, too.
In general, they are more open.
You can get one for free: Wettersonde



https://github.com/arnobert/rs41_rust

tübix

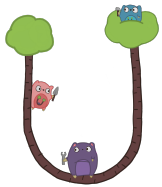
u-root



an initramfs builder with Busybox-like tools written in Go

https://u-root.org

u-root



an initramfs builder with Busybox-like tools written in Go

https://u-root.org

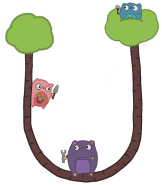
`cpu`

cpu command in Go, inspired by the Plan 9 cpu command

https://github.com/u-root/cpu

tübix

u-root



an initramfs builder with Busybox-like tools written in Go

https://u-root.org

`cpu`

cpu command in Go, inspired by the Plan 9 cpu command

https://github.com/u-root/cpu

How about USB CPU? Demo time!

tübix

Gokrazy

what if we massively reduced the overall system complexity by getting rid of all software we don't strictly need, and instead built up a minimal system from scratch entirely in Go

https://gokrazy.org/

Gokrazy

what if we massively reduced the overall system complexity by getting rid of all software we don't strictly need, and instead built up a minimal system from scratch entirely in Go

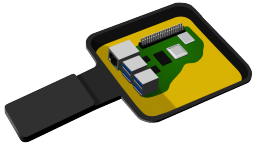
https://gokrazy.org/

Build Go appliances for the Raspberry Pi using gokrazy!

https://media.ccc.de/v/gpn21-78-build-go-appliances-for-the-raspberry-pi-using-gokrazy-

tübix

Racklet



Racklet is a fully-integrated, miniature server rack.

https://racklet.io/

tübix

Finding a community

Communication channels

Communication channels

There are wikis, forums, IRC, Matrix, Slack, Telegram groups…

Communication channels

There are wikis, forums, IRC, Matrix, Slack, Telegram groups…

Examples

https://linux-sunxi.org/
https://en.opensuse.org/openSUSE:IRC_list
https://slack.osfw.dev/

Communication channels

There are wikis, forums, IRC, Matrix, Slack, Telegram groups…

Examples

https://linux-sunxi.org/
https://en.opensuse.org/openSUSE:IRC_list
https://slack.osfw.dev/

In person

Have you visited your local fablab, hackerspace or makerspace yet?

Thank you! :)

Related

Repurposing Gadgets (FOSSASIA Summit 2021)

https://metaspora.org/repurposing-gadgets-fossasia2021.pdf

Drivers from Outer Space (CLT 2022)

https://chemnitzer.linux-tage.de/2022/en/programm/beitrag/226

Speedy Distro Porting via the cpu Command

https://media.ccc.de/v/3802-speedy-distro-porting-via-the-cpu-command

Platform System Interface - Design und Evaluation holistischer Computerarchitektur (rC3 2022)

https://media.ccc.de/v/fire-shonks-2022-49154-platform-system-interface-design-und-evaluation-holistischer-computerarchitektur

Follow Me

Daniel Maslowski

https://github.com/orangecms
https://twitter.com/orangecms
https://mastodon.social/@cyrevolt
https://youtube.com/@cyrevolt
https://twitch.tv/cyrevolt

https://metaspora.org/sbcs-and-socs-tuebix-2023.pdf

tübix