

Kerneländerungen n testen

Wie verifiziert man Änderungen auf einer
großen Zahl verschiedener Systeme

Heiko Stübner

- Kernel-Entwickler bei BQ (eReader, Smartphones, Tablets)
- embedded-Einstieg 2008 mit Openmoko Freerunner
- erste Kernelschritte 2010 auf dem Thalia Oyo
- Maintainer der Rockchip ARM-SoC im Mainline-Kernel

Motivation

- Anzahl unterschiedlicher ARM-SoCs (und anderer Chips) wächst jedes Jahr
- Auch mein in den letzten 2 CLTs vorgestellter Rockchip support wächst
- Mehr SoCs (rk3288, rk3368, rk3399, ...), Mehr Boards
- => Schubladen voll mit Rockchip boards
- => Posteingang voll mit Änderungen
- Skalierungsproblem - Testen ohne wahnsinnig zu werden

Rockchip-Status 2014

	RK28xx ARM9	RK29xx Cortex-A8 Vivante	RK292x Cortex-A9 (1)	RK3066 Cortex-A9 (2)	RK3188 Cortex-A9 (4)
INTC	DW APB ICTL				
Timer					
Pinctrl					
Clocks					
Serial					
SMP					
DMA	dw_dma				
MMC					
NAND					
I2C					
SPI					
Ethernet					
USB					
HDMI					
LCDC					

	funktioniert
	benötigt Anpassung
	sollte funktionieren
	nicht unterstützt

Rockchip-Status März 2015

	RK28xx ARM9	RK29xx Cortex-A8 Vivante	RK292x Cortex-A9 (1)	RK3066 Cortex-A9 (2)	RK312x Cortex-A7	RK3188 Cortex-A9 (4)	RK3288 Cortex-A17 (4)	RK3368 Cortex-A53 (8)
INTC	DW APB ICTL							
Timer								
Pinctrl								
Clocks								
Serial								
SMP								
DMA	dw_dma							
MMC								
NAND								
I2C								
SPI								
Ethernet								
USB								
HDMI								
LCDC								

funktioniert
 benötigt Anpassung
 sollte funktionieren
 nicht unterstützt

Rockchip-Status 2016

[illegible]

Rockchip-Status aktuell

[illegible]

Kerneländerungen vs. normale Software

- Software: ändern, kompilieren, starten, läuft (oder nicht)
- rinse, repeat
- Unit-Tests helfen Regressions zu fangen
- Kernel: Testgerät jedesmal neustarten
- Hardware verhält sich in Teilen unberechenbar
- Entwicklungsumgebung weg, evtl startet Rechner gar nicht mehr

Exkurs: Kernel-Patch

- normale OS-Projekte nutzen ggf. einfach github pull-requests
- Kernel: Teilbereiche haben eigene Maintainer
- `scripts/get_maintainer.pl` als Hilfe
- Aufteilung in logische (leichtverdauliche) Einheiten ("und" als Indikator)
- jeder Patch muss allein funktionieren und keine Warnungen/Fehler erzeugen
- immer als Mail (inline)
- `git send-email` statt regulärer Mail-Clients

[PATCH] serial: 8250_dw: Honor clk_round_rate errors in dw8250_set_termios

Von: Heiko Stuebner <heiko@sntech.de>

An: gregkh@linuxfoundation.org

Kopie: jslaby@suse.com, linux-serial@vger.kernel.org, linux-kernel@vger.kernel.org, Heiko Stuebner <heiko@sntech.de>

clk_round_rate returns a signed long and may possibly return errors in it, for example if there is no possible rate.

Till now dw8250_set_termios ignored any error, the signedness and would just use the value as input to clk_set_rate. This of course falls apart if there is an actual error, so check for errors and only try to set a rate if the value is actually valid.

Signed-off-by: Heiko Stuebner <heiko@sntech.de>

drivers/tty/serial/8250/8250_dw.c | 7 +++++--
1 file changed, 5 insertions(+), 2 deletions(-)

diff --git a/drivers/tty/serial/8250/8250_dw.c b/drivers/tty/serial/8250/8250_dw.c

index 6ee55a2d47bb..223ac234ddb2 100644

--- a/drivers/tty/serial/8250/8250_dw.c

+++ b/drivers/tty/serial/8250/8250_dw.c

@@ -257,7 +257,7 @@ static void dw8250_set_termios(struct uart_port *p, struct ktermios *termios,

```
{  
    unsigned int baud = tty_termios_baud_rate(termios);  
    struct dw8250_data *d = p->private_data;  
-    unsigned int rate;  
+    long rate;  
    int ret;
```

```
    if (IS_ERR(d->clk) || !old)
```

@@ -265,7 +265,10 @@ static void dw8250_set_termios(struct uart_port *p, struct ktermios *termios,

```
    clk_disable_unprepare(d->clk);  
    rate = clk_round_rate(d->clk, baud * 16);  
-    ret = clk_set_rate(d->clk, rate);  
+    if (rate < 0)  
+        ret = rate;  
+    else  
+        ret = clk_set_rate(d->clk, rate);  
    clk_prepare_enable(d->clk);
```

```
    if (!ret)
```

--

2.11.0

Einfache Lösung: Virtuelle Maschine

- kann mit deutlich weniger Aufwand neugestartet werden
- Entwicklungsumgebung bleibt erhalten
- lässt sich vermutlich recht gut automatisieren
- also: bauen, starten, sehen ob es funktioniert

oder doch nicht

- VMs emulieren immer nur bestimmte Hardwarekomponenten
- nur bei hardwarefernen Komponenten
- Kernbereiche von Subsystemen, Dateisysteme, Scheduler
- vermutlich aber auch da noch Hardware-Quirks

Embedded Sichtweise: einfacher

- Board auf dem Tisch
- serielle Schnittstelle
- einfach Kernel einspielen und testen
- also keine Beeinträchtigungen des Workflows

oder doch nicht

- bei einem Board noch handhabbar
- Tisch voller Boards - wenn Hobby, wohnqualität beachten ;-)
- Boardspezifische Mechanismen um Kernel zu aktualisieren
- Transportproblem, wenn man woanders arbeitet

schlechte Alternative: Absenden und hoffen

- "simple Änderung kann unmöglich Fehler haben"
- maximal Kompilation testen
- darauf hoffen, dass schon nix schief geht
- klappt fast nie und passiert trotzdem viel zu oft
- andere Entwickler mit entsprechender Hardware testen lassen

[PATCH v3 1/2] clk: rockchip: add "," to mux_pll_src_apll_dppll_gppll_usb480m_p on rk3036

Von: Heiko Stuebner <heiko@sntech.de>

An: sboyd@codeaurora.org, mturquette@baylibre.com

Kopie: linux-rockchip@lists.infradead.org, linux-clk@vger.kernel.org, Heiko Stuebner <heiko@sntech.de>

The mux_pll_src_apll_dppll_gppll_usb480m_p parent list was missing a "," between the 3rd and 4th parent names, making them fall together and thus lookups fail. Fix that.

Fixes: 5190c08b2989 ("clk: rockchip: add clock controller for rk3036")

Signed-off-by: Heiko Stuebner <heiko@sntech.de>

drivers/clk/rockchip/clk-rk3036.c | 2 +-
1 file changed, 1 insertion(+), 1 deletion(-)

diff --git a/drivers/clk/rockchip/clk-rk3036.c b/drivers/clk/rockchip/clk-rk3036.c

index 924f560dcf80..dcde70f4c105 100644

--- a/drivers/clk/rockchip/clk-rk3036.c

+++ b/drivers/clk/rockchip/clk-rk3036.c

@@ -127,7 +127,7 @@ PNAME(mux_ddrphy_p) = { "dppll_dds", "gppll_dds" };
PNAME(mux_pll_src_3ppll_p) = { "apll", "dppll", "gppll" };
PNAME(mux_timer_p) = { "xin24m", "pclk_peri_src" };

-PNAME(mux_pll_src_apll_dppll_gppll_usb480m_p) = { "apll", "dppll", "gppll", "usb480m" };
+PNAME(mux_pll_src_apll_dppll_gppll_usb480m_p) = { "apll", "dppll", "gppll", "usb480m" };

PNAME(mux_mmc_src_p) = { "apll", "dppll", "gppll", "xin24m" };
PNAME(mux_i2s_pre_p) = { "i2s_src", "i2s_frac", "ext_i2s", "xin12m" };
--
2.11.0

Automatische Testsysteme

- über die Zeit entstanden einige automatische Testsysteme
- finden die größten Probleme in Einsendungen
- auf verschiedenen Ebenen von Mailingliste bis Boot

kbuild test robot

- liest Mailinglisten
- liest (und findet) Kernel git-Repositories
- baut gefundene Sachen (verschiedene Configs) und sucht Fehler und Warnungen
- manchmal 1 Minute nach Absenden eines Patches schon Rückmeldung
- Patches mit kbuild-Antworten werden meist von Maintainern ignoriert

[phy:next 11/40] include/linux/kernel.h:831:48: error: initialization from incompatible pointer type

Datum: 11.09.16 07:27

Von: kbuild test robot <fengguang.wu@intel.com>

An: Frank Wang <frank.wang@rock-chips.com>

Kopie: kbuild-all@01.org, Kishon Vijay Abraham I <kishon@ti.com>, Heiko Stuebner <heiko@sntech.de>, Guenter Roeck <linux@roeck-us.net>

tree: <https://git.kernel.org/pub/scm/linux/kernel/git/kishon/linux-phy.git> next

head: 84371ce018465a80661301c8d96d5779ecb00dfe

commit: 0e08d2a727e68bbe426457dc61ec11a5c6a76ed6 [11/40] phy: rockchip-inno-usb2: add a new driver for Rockchip usb2phy

config: powerpc-allmodconfig (attached as .config)

compiler: powerpc64-linux-gnu-gcc (Debian 5.4.0-6) 5.4.0 20160609

reproduce:

```
wget https://git.kernel.org/cgit/linux/kernel/git/wfg/lkp-tests.git/plain/sbin/make.cross -O ~/bin/make.cross
chmod +x ~/bin/make.cross
git checkout 0e08d2a727e68bbe426457dc61ec11a5c6a76ed6
# save the attached .config to linux build tree
make.cross ARCH=powerpc
```

Note: the phy/next HEAD 84371ce018465a80661301c8d96d5779ecb00dfe builds fine.

It only hurts bisectibility.

All errors (new ones prefixed by >>):

```
drivers/phy/phy-rockchip-inno-usb2.c:124:16: error: field 'clk480m_hw' has incomplete type
struct clk_hw clk480m_hw;
```

^

In file included from include/linux/clock.h:16:0,

from drivers/phy/phy-rockchip-inno-usb2.c:17:

drivers/phy/phy-rockchip-inno-usb2.c: In function 'rockchip_usb2phy_clk480m_enable':

| | *include/linux/kernel.h:831:48: error: initialization from incompatible pointer type [-Werror=incompatible-pointer-types]*

```
const typeof( ((type *)0)->member ) *__mptr = (ptr); \
^
```

drivers/phy/phy-rockchip-inno-usb2.c:159:3: note: in expansion of macro 'container_of'

linux-next

- Vereint (fast) all Maintainer Git-Trees
- simuliert also das nächste Merge-Window
- (fast) tägliche Generierung
- Findet merge-Konflikte zwischen verschiedenen Trees
- Basis für weitere Tests



index : kernel/git/next/linux-next.git

The linux-next integration testing tree

master  switch

Linux Next Group

about **summary** refs log tree commit diff

log msg  search

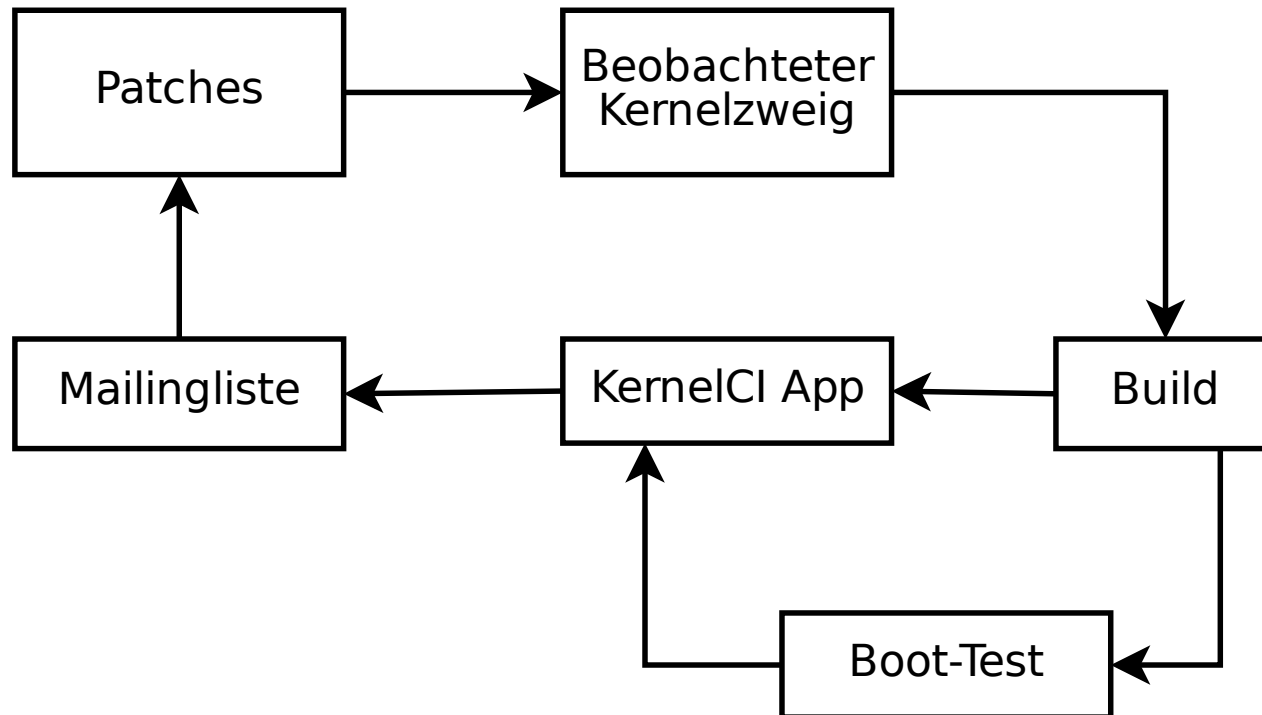
Branch	Commit message	Author	Age
akpm	treewide: decouple cacheflush.h and set_memory.h	 Laura Abbott	2 days
akpm-base	Merge remote-tracking branch 'rtc/rtc-next'	 Stephen Rothwell	2 days
master	Add linux-next specific files for 20170310	 Stephen Rothwell	2 days
stable	Merge branch 'core-urgent-for-linus' of git://git.kernel.org/pub/scm/linux/ke...	 Linus Torvalds	3 days

Tag	Download	Author	Age
next-20170310	commit 5be4921c99...	 Stephen Rothwell	2 days
next-20170309	commit 2e62da1e53...	 Stephen Rothwell	3 days
next-20170308	commit 56b8bad5e0...	 Stephen Rothwell	4 days
next-20170307	commit 3c90632cc1...	 Stephen Rothwell	4 days
next-20170306	commit 9d51244afb...	 Stephen Rothwell	6 days
v4.11-rc1	commit c1ae3cfa0e...	 Linus Torvalds	6 days
next-20170303	commit c0b7b2b33b...	 Stephen Rothwell	9 days
next-20170302	commit 9bf0327920...	 Stephen Rothwell	10 days
next-20170301	commit 51788aeb7...	 Stephen Rothwell	11 days
next-20170228	commit 8d01c06948...	 Stephen Rothwell	12 days
[...]			

Age	Commit message	Author
2 days	Add linux-next specific files for 20170310 HEAD next-20170310 master	 Stephen Rothwell
2 days	Merge branch 'akpm/master'	 Stephen Rothwell
2 days	treewide: decouple cacheflush.h and set_memory.h akpm	 Laura Abbott
2 days	drivers/staging/media/atomisp/pci/atomisp2: use set_memory.h	 Andrew Morton
2 days	drivers/video/fbdev/vermilion/vermilion.c: use set_memory.h header	 Laura Abbott
2 days	drivers/misc/sram-exec.c: use set_memory.h header	 Laura Abbott
2 days	alsa: use set_memory.h header	 Laura Abbott
2 days	kernel/power/snapshot.c: use set_memory.h header	 Laura Abbott

KernelCI

- automatische Build und Boot Tests
- verteiltes System - Boards verteilt auf Labs (Firmen, etc)
- Pengutronix, BayLibre, Free Electrons, Collabora, Embedded Bits, Linaro, TI, khilman, tbaker
- Zentrale baut verschiedene Kernel-Repos (+configs)
- Booten auf den passenden Boards
- Rückmeldung an Zentrale ob Boot klappt



Available Boot Reports

The results shown here cover the last **14 days** of available data starting from **Tue, 07 Mar 2017** (time is UTC based).

25

boot reports per page

Q

rk3288 next-

Tree	Branch	Kernel	Board Model	Defconfig	Arch.	Lab Name	Date	Status	
next	local/master	next-20170306	rk3288-veyron-jerry	multi_v7_defconfig+CONFIG_EFI=y	arm	lab-baylibre-seattle	2017-03-06	✓	Q
next	local/master	next-20170306	rk3288-veyron-jerry	multi_v7_defconfig+CONFIG_ARM_LPAE=y	arm	lab-baylibre-seattle	2017-03-06	✓	Q
next	local/master	next-20170306	rk3288-veyron-jerry	multi_v7_defconfig+CONFIG_PROVE_LOCKING=y	arm	lab-baylibre-seattle	2017-03-06	✓	Q
next	local/master	next-20170306	rk3288-veyron-jerry	multi_v7_defconfig+CONFIG_LKDTM=y	arm	lab-baylibre-seattle	2017-03-06	✓	Q
next	local/master	next-20170306	rk3288-veyron-jerry	multi_v7_defconfig	arm	lab-baylibre-seattle	2017-03-06	✓	Q
next	local/master	next-20170306	rk3288-veyron-jerry	multi_v7_defconfig+CONFIG_THUMB2_KERNEL=y+CONFIG_ARM_MODULE_PLTS=y	arm	lab-baylibre-seattle	2017-03-06	✓	Q
next	local/master	next-20170303	rk3288-veyron-jerry	multi_v7_defconfig	arm	lab-baylibre-seattle	2017-03-03	✓	Q
next	local/master	next-20170303	rk3288-veyron-jerry	multi_v7_defconfig+CONFIG_THUMB2_KERNEL=y+CONFIG_ARM_MODULE_PLTS=y	arm	lab-baylibre-seattle	2017-03-03	✓	Q
next	local/master	next-20170303	rk3288-veyron-jerry	multi_v7_defconfig+CONFIG_EFI=y	arm	lab-baylibre-seattle	2017-03-03	✓	Q
next	local/master	next-20170303	rk3288-veyron-jerry	multi_v7_defconfig+CONFIG_ARM_LPAE=y	arm	lab-baylibre-seattle	2017-03-03	✓	Q

Boot details for board «rk3288-veyron-jerry» (lab-baylibre-seattle)

Lab name	lab-baylibre-seattle Q	Status	
Board	rk3288-veyron-jerry Q	Architecture	arm
Board instance		SoC	rockchip — Q
Tree	next — Q	Endianness	little
Git branch	local/master	Warnings	0
Git describe	next-20170306 — Q	Boot time	14 sec. 310 mill.
Defconfig	multi_v7_defconfig — Q	Boot log	txt Q — html Q
Date	2017-03-06 04:37:22 UTC		
Compiler	gcc		
Compiler version	5.3.1		
Compiler string	gcc version 5.3.1 20160412 (Linaro GCC 5.3-2016.05)		
Cross-compile	arm-linux-gnueabihf-		
		Boot retries	0
Dtb	dtbs/rk3288-veyron-jerry.dtb Q		
Dtb address			
Load address	0x01000000		
Initrd address	0x02000000		
Kernel image	zImage Q (6.99 MiB)		

Other Boot Reports

Exact Match

Lab Name	Failure Reason	Boot Log	Date	Status
No similar boot reports found.				

Nicht Nachhaltig :-)

- Testen auslagern, erzeugt böses Blut bei den "Opfern"
- z.B. KernelCI-Maintainer versuchen meist Probleme zu identifizieren
- tinydrm-Aufreger für 4.11
- Zeit für Testumlauf bis zum Ergebnis
- Also doch wieder lokales Testen

Lösung für viele Boards: Netboot

- Standard: Root-Dateisystem einspielen + Massenspeicher allgemein
- Standard: Module pro Kernel-Build einspielen
- Umgehen mittels TFTP (Kernel) + NFS (RootFS)
- Zentraleinheit kann alle Kernel + RootFS vorhalten

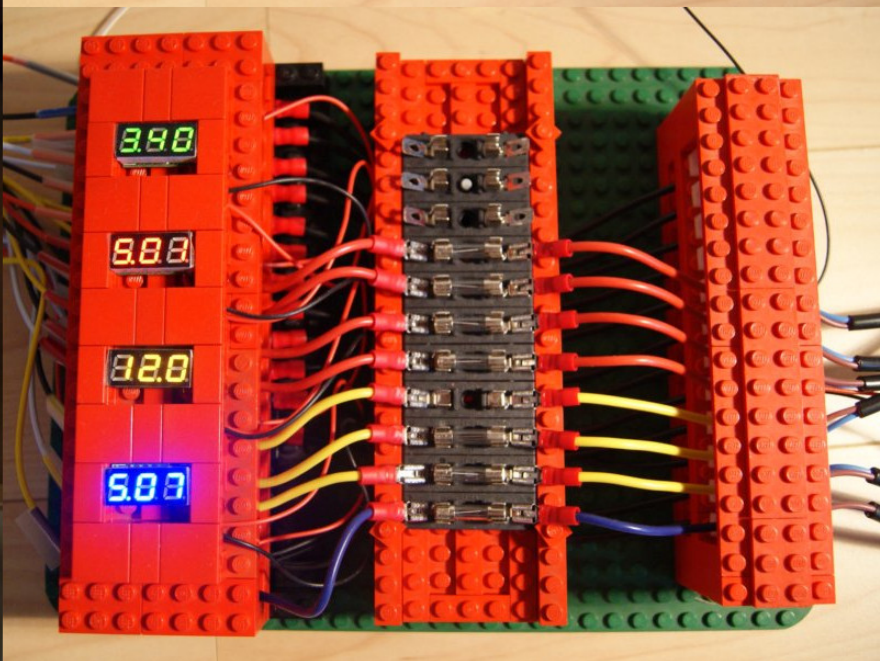
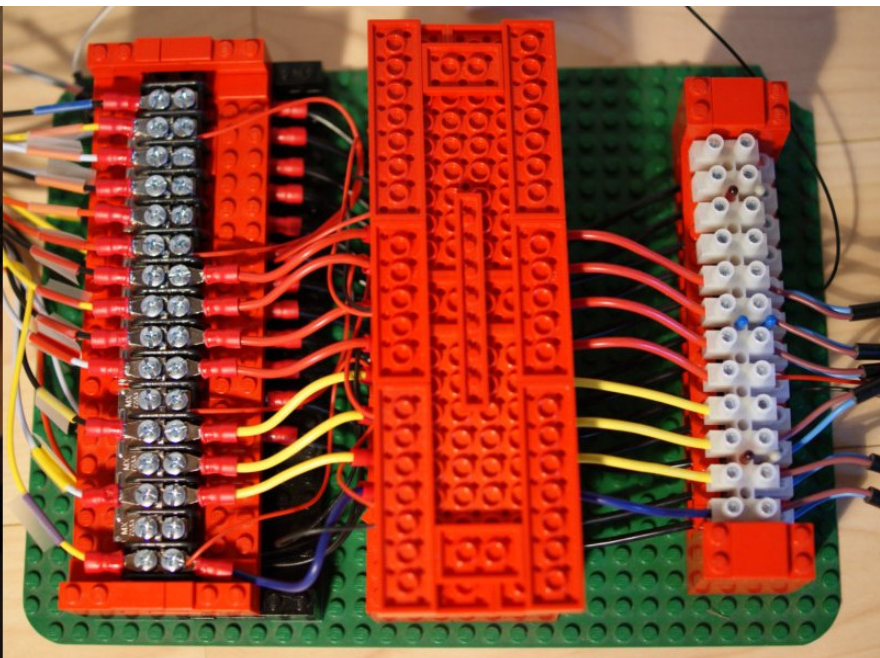
Netboot Probleme

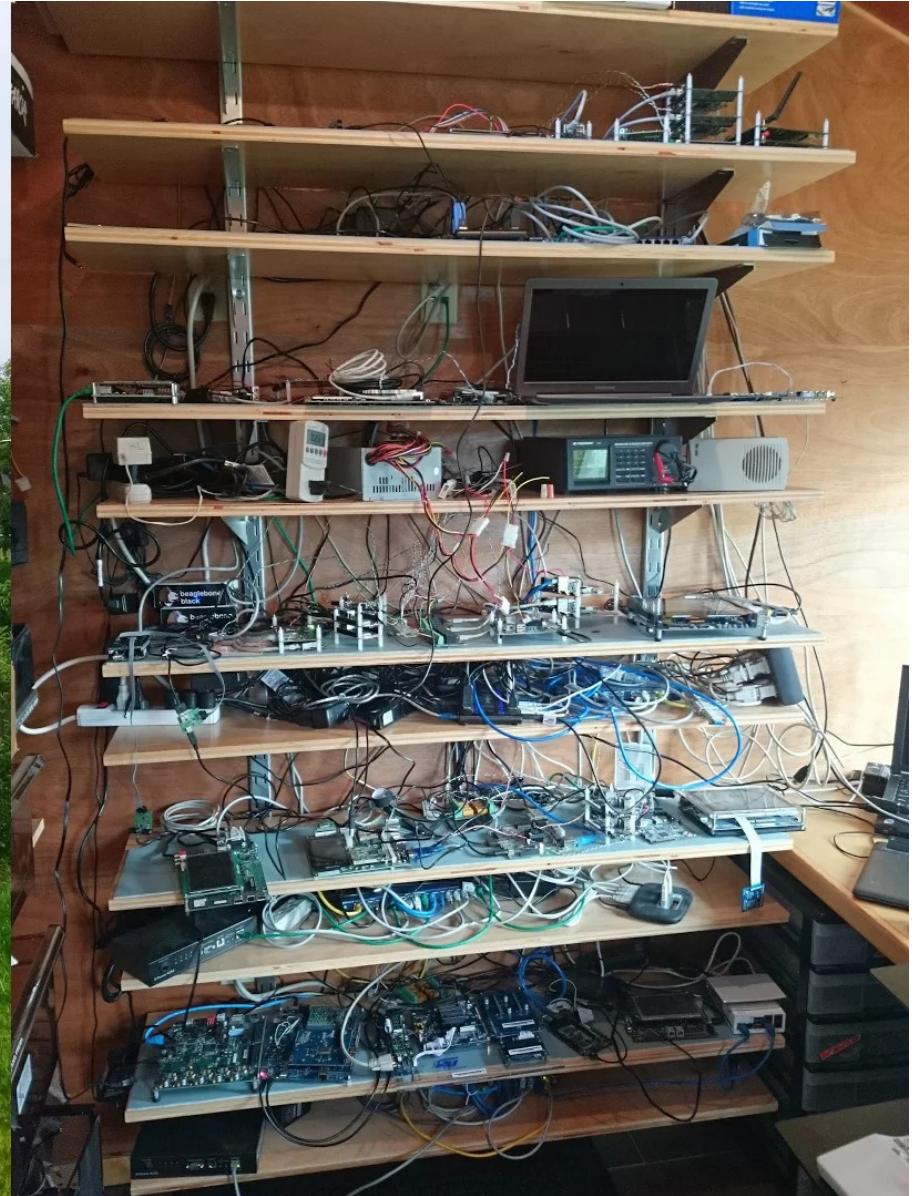
- zwar kein Massenspeicher nötig, dafür Netz
- proprietäre Bootloader bringen oft kein Netboot mit
- Ethernet-Controller im Bootloader
- USB-Ethernet erfordert USB im Bootloader
- Netzwerkconfig wenn der Arbeitsrechner auch TFTP-Host ist
- Boards müssen Server-IP kennen; ändern, wenn sich der Arbeitsort ändert

Lösung: Boardfarm

- Zentrale Aufbewahrung
- statische Infrastruktur
- automatisiertes Ein-/Ausschalten
- keine Knöpfe drücken / Netzteile stecken
- Über das Netz erreichbar
- Ggf. durch mehrere Entwickler nutzbar
- Integration in KernelCI möglich
- Viele verschiedene Bauformen









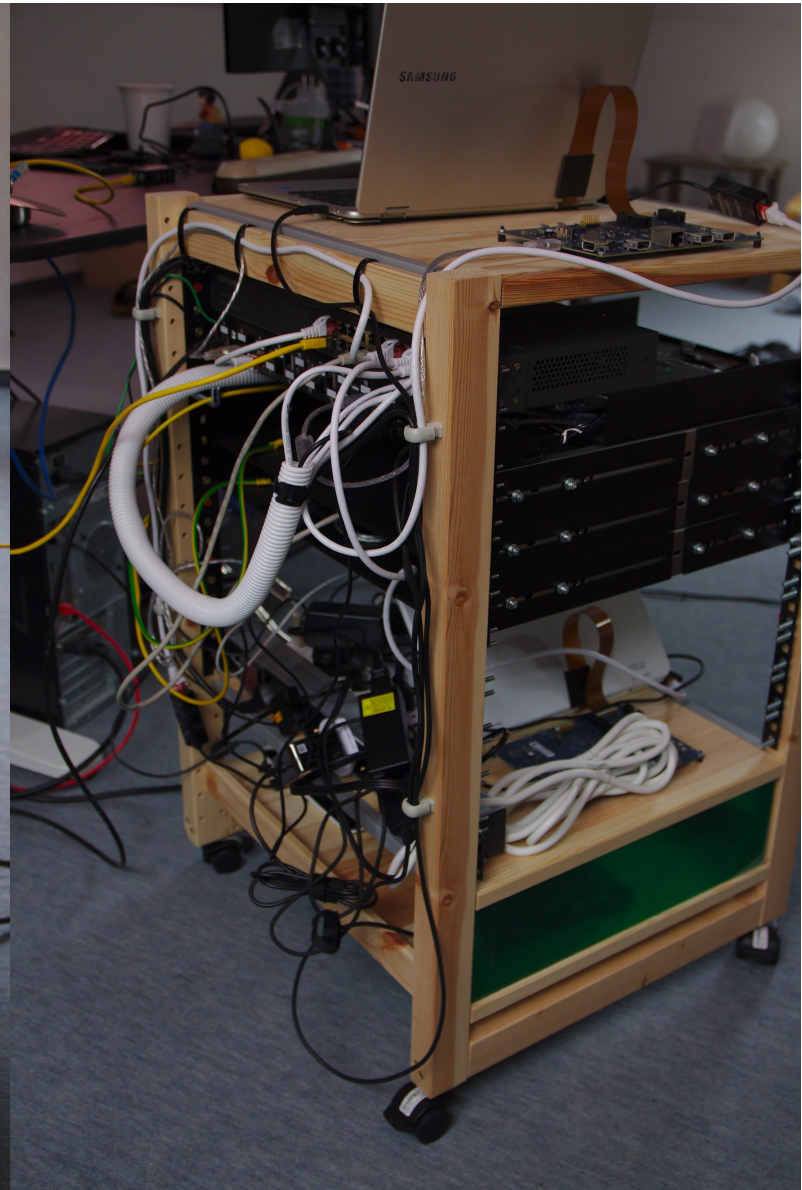
Meine Boardfarm

- zuerst einfaches Regalfach zum probieren
- Kabelgewirr unschön
- Angst vor angekorkelten Boards, wenn alles auf Holzregalen liegt
- Zentraleinheit ein überzähliges embedded-Board
- SATA-HDD an SATA-USB-Bridge an USB2.0 Controller



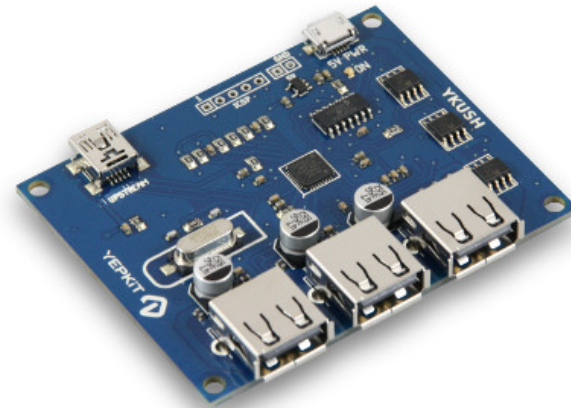
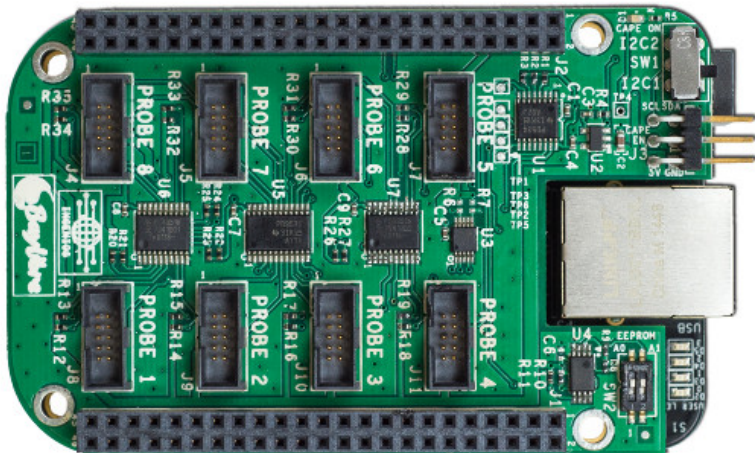
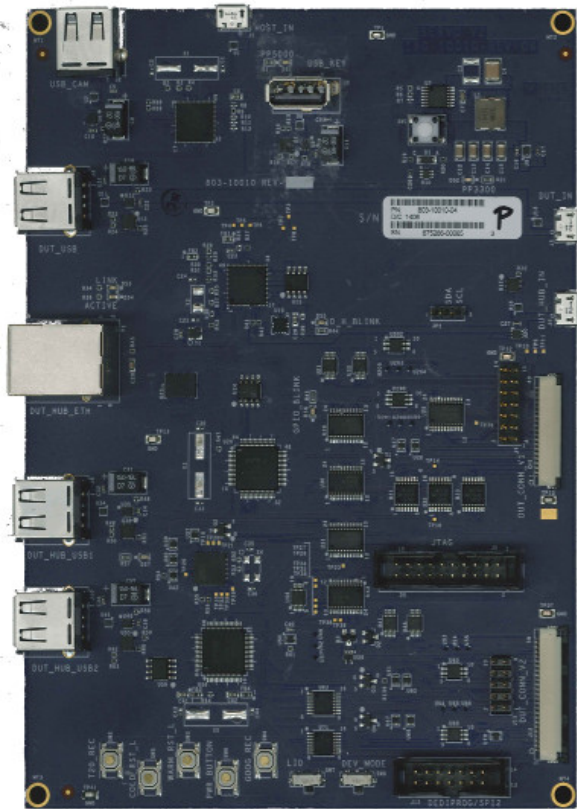
Meine Boardfarm - Neubau

- Ivar - mit ein bisschen Sägen, 19 Zoll Einschub
- 19 Zoll Standardkomponenten
- Metallauszüge mit Erdung - hoffentlich verringerte Brandgefahr
- WIP: vernünftiges Kabelmanagement
- WIP: Acrylglasverkleidung für Wohnlichkeit und Farbtupfer



Boardfarm-Management: Stromverwaltung

- Hauptschalter: FritzDect 200
- soll nicht dauerhaft laufen
- 5V über USB-Hub, manche können einzelne Ports schalten / YKush
- ChromeOS-Geräte haben standardisiertes Debug-Board (Servo)
- Später: BayLibre ACME: Board mit Aufsätzen für 5V, 12V Rundstecker und mehr
- TODO: Lastspitzen

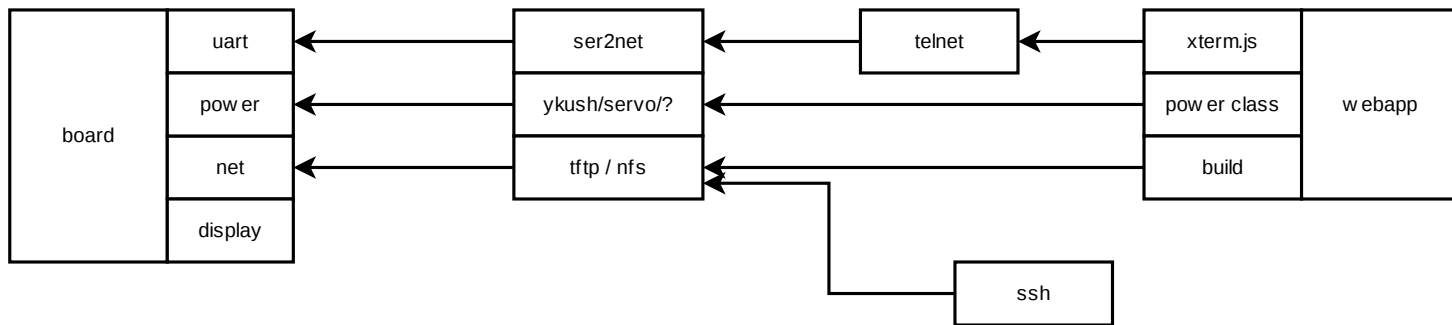


Boardfarm-Management: LAVA

- Verwendet in vielen KernelCI-Labs
- Verteilt BootImages auf Boards / kümmert sich um Boots
- schwierig für direkte Tests zu verwenden (Job-Queue, Priorität)

Boardfarm-Management: Webapp

- soll primär meine Arbeitsweise abbilden
- Applikation auf Basis von Qooxdoo + node.js Backend
- abstrahiert Board-Liste, Powermanagement, Konsole
- direkter Zugriff auf die serielle Konsole



Temperature: 27.50 °C Power: 23.88 W Load: 1.10 1.11 0.72 8/228 3133

▶ Power On

■ Power Off

↺ Reset

Build

servo:9900(1/1)

■ pinky

servo:9901(1/1)

■ gru

servo:9902(1/1)

■ kevin

ykush:YK21466(1/3)

■ firefly

ykush:YK21544(3/3)

■ kyllin

■ r88

■ rock

```
1.853367] mmc1: SDHCI controller on fe330000.sdhci [fe330000.sdhci] using ADMA
1.866081] ledtrig-cpu: registered to indicate activity on CPUs
1.873968] usbcore: Registered new interface driver usbhcd
1.880224] usbhcd: USB HID core driver
1.886777] NET: Registered protocol family 17
1.891847] Spnet: Installing 9P2000 support
1.896667] Key type dns_resolver registered
1.902035] registered taskstats version 1
1.922157] input: gpio-keys as /devices/platform/gpio-keys/input/input0
1.923644] hctosys: unable to open rtc device (rtc0)
1.967575] mmc1: power class selection to bus width 8 ddr 0 failed
1.982705] mmc1: error -110 whilst initialising MMC card
2.140768] mmc1: power class selection to bus width 8 ddr 0 failed
3.155893] mmc1: error -110 whilst initialising MMC card
3.327717] mmc1: power class selection to bus width 8 ddr 0 failed
3.342840] mmc1: error -110 whilst initialising MMC card
3.55363] mmc1: power class selection to bus width 8 ddr 0 failed
3.570485] mmc1: error -110 whilst initialising MMC card
14.233831] ppi800_pcie: disabling
14.237657] wlan_pdn: disabling
14.241273] pp3000_sd_slot: disabling
14.245368] pp3300_disp: disabling
14.249165] pp3300_wifi_bt: disabling
14.253306] p3.3v_dig: disabling
14.256930] ppvar_sd_card_io: disabling
14.261236] ALSA device list:
14.264568]   No soundcards found.
14.268538] ttyS2 - failed to request DMA
111.705599] VFS: Unable to mount root fs via NFS, trying floppy.
111.712541] VFS: Cannot open root device "nfs" or unknown-block(2,0): error -6
111.720656] Please append a correct "root=" boot option; here are the available partitions:
111.730019] Kernel panic - not syncing: VFS: Unable to mount root fs on unknown-block(2,0)
111.735270] CPU: 0 PID: 1 Comm: swapper/0 Not tainted 4.10.0-10773-gl4625bbb6de47 #305
111.746031] Hardware name: Google Kevin (DT)
111.752803] Call trace:
111.755544] [] dump_backtrace+0x0/0x238
111.761582] [] show_stack+0x14/0x20
111.767233] [] dump_stack+0x9c/0xc0
111.772882] [] panic+0x118/0x288
111.778241] [] mount_block_root+0x178/0x250
111.784667] [] mount_root+0x11c/0x134
111.790500] [] prepare_namespace+0x13c/0x184
111.797024] [] kernel_init_freeable+0x214/0x238
111.803842] [] kernel_init+0x10/0x100
111.809685] [] ret_from_fork+0x10/0x50
111.815627] SMP: stopping secondary CPUs
111.820013] Kernel Offset: disabled
111.823909] Memory Limit: none
111.827325] ---[ end Kernel panic - not syncing: VFS: Unable to mount root fs on unknown-block(2,0)
331.111105] usb 1-1: USB disconnect, device number 2
```


Temperature: 32.50 °C Power: 47.49 W Load: 1.28 1.06 0.75 7/244 3234

▶ Power On

■ Power Off

↺ Reset

Build

servo:9900(1/1)

■ pinky

servo:9901(1/1)

■ gru

servo:9902(1/1)

■ kevin

ykush:YK21466(1/3)

▶ firefly

ykush:YK21544(3/3)

■ kylin

■ r88

■ rock

```
[ 16.425340] hub 1-1:1.0: 4 ports detected
[ OK ] Reached target System Initialization.
[ OK ] Started Daily Cleanup of Temporary Directories.
[ OK ] Reached target Basic System.
Starting System Logging Service...
Starting Getty on tty2-tty6 if dbus and login are not available...
Starting LSB: Load kernel modules needed to enable cpufreq scaling...
[ OK ] Reached target System Time Synchronized.
[ OK ] Started Daily apt activities.
[ OK ] Reached target Timers.
[ OK ] Started Regular background program processing daemon.
[ OK ] Started System Logging Service.
Starting Daily apt activities...
[ OK ] Started Raise network interfaces.
[ OK ] Reached target Network.
[ OK ] Started boardfarm-checkin.service.
Starting Permit User Sessions...
[ OK ] Reached target Network is Online.
Starting /etc/rc.local Compatibility...
Starting LSB: exim Mail Transport Agent...
[ OK ] Started getty on tty2-tty6 if dbus and login are not available.
[ OK ] Started /etc/rc.local Compatibility.
[ OK ] Started Permit User Sessions.
[ OK ] Started LSB: Load kernel modules needed to enable cpufreq scaling.
Starting LSB: set CPUFreq kernel parameters...
[ OK ] Started Getty on tty4.
[ OK ] Started Getty on tty2.
[ OK ] Started Getty on tty3.
[ OK ] Started Getty on tty6.
[ OK ] Started Getty on tty5.
[ OK ] Started Serial Getty on ttyS2.
[ OK ] Started Getty on tty1.
[ OK ] Reached target Login Prompts.
[ OK ] Started LSB: set CPUFreq kernel parameters.
```

Debian GNU/Linux 9 192.168.140.54 ttyS2

192 login: root

Password:

Last login: Thu Mar 9 09:16:17 UTC 2017 on ttyS2

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

root@192:~#

root@192:~#

root@192:~#

root@192:~#

Resourcen

- www.kernelci.org
- <https://github.com/mmind/boardfarm-scripts>
- <https://github.com/mmind/boardfarm-mgmt>

Fragen?