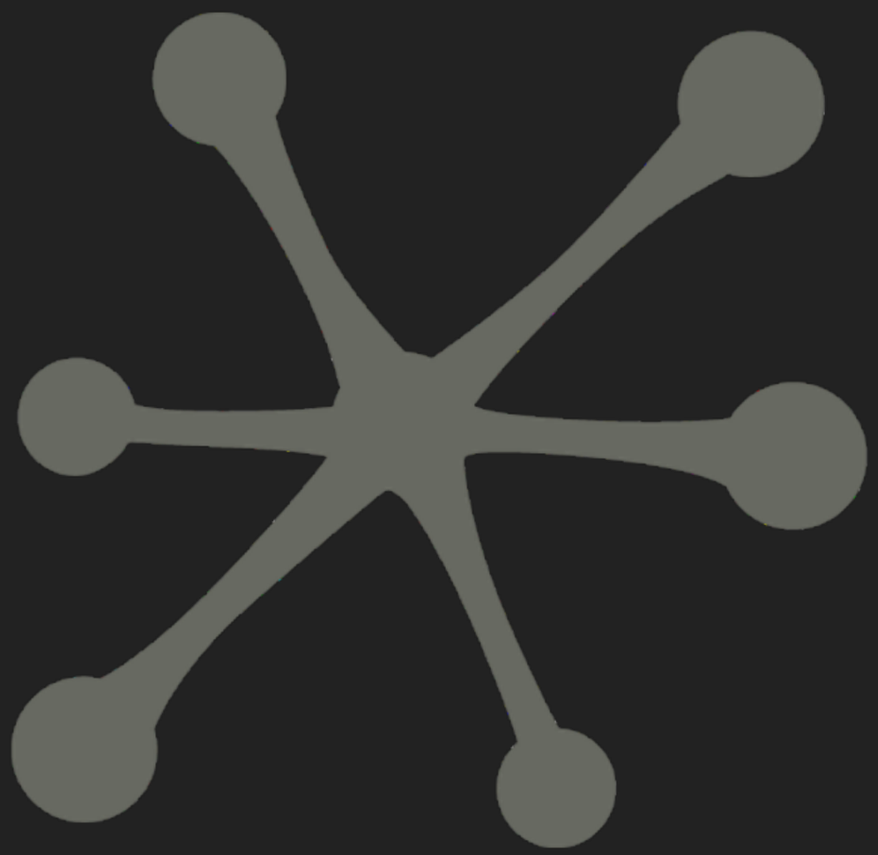
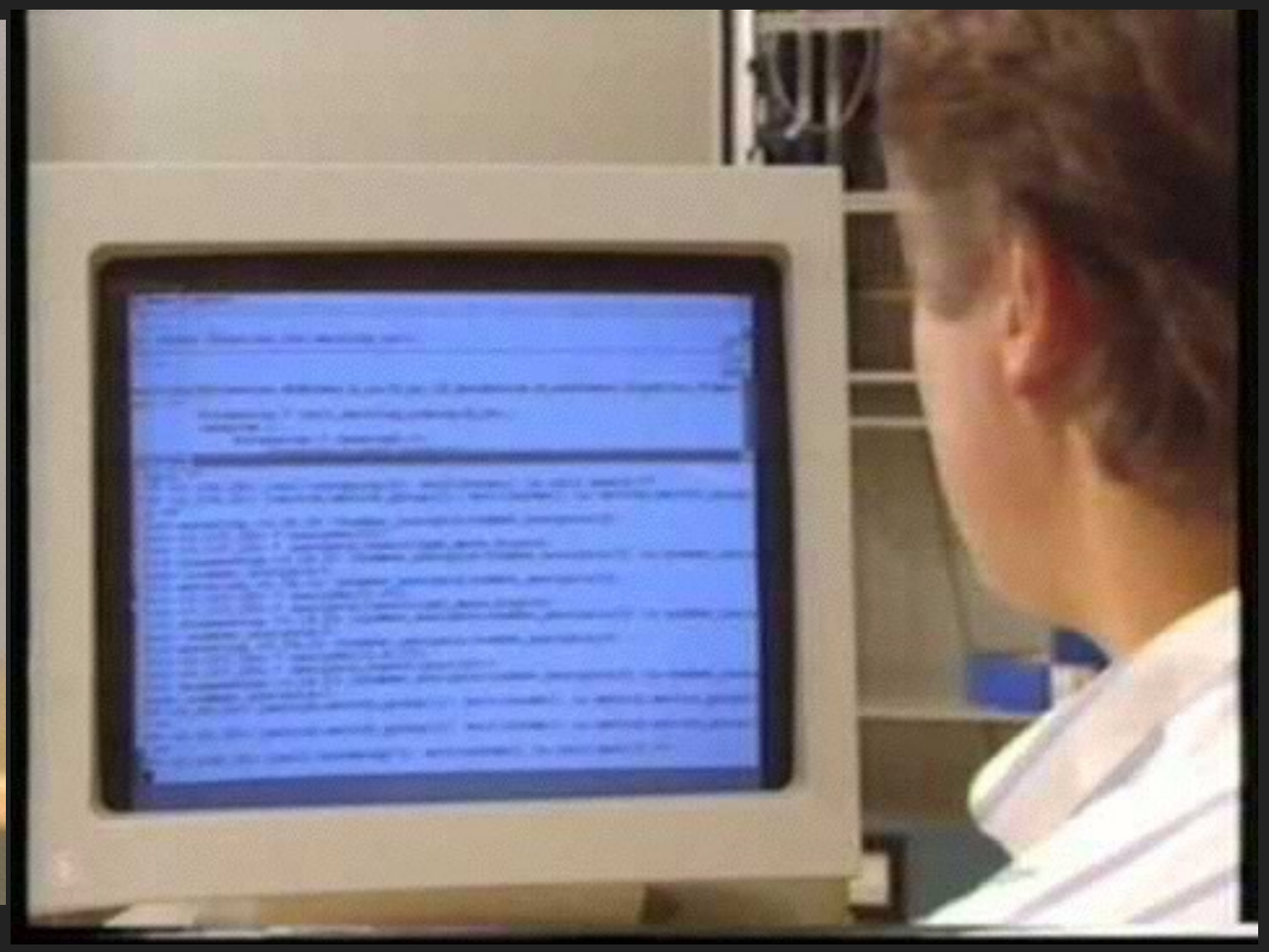


ENHANCED EMBEDDED SYSTEMS

NERVES PROJECT

1980



SMALLER!

EASIER!

Monochrome (1.101w 07-May-08) (Last on Wed May 14 13:36)

New streamlined layout! **Easier to use!** New files! **Extra exclamation marks!**

Dish some dirt at <MTO> today!

archon

Menu [**ESC**] = Utilities (inc. Talker & **EXIT**)

You don't use ssh. Booo!

Welcome to
the new
version of
Monochrome!
(version 1.101w)

Menu [**I**] = Help and **I**nformation on Monochrome

Menu [**N**] = **N**ews and Media
Menu [**T**] = Science, **T**echnology and Medicine
Menu [**E**] = **E**ntertainment
Menu [**C**] = Society and **C**ulture
Menu [**R**] = **R**ecreation

Menu [**M**] = **M**onochrome Users

Hello 'SexDrugs&DrumMachinesForAgRaveGeneration'. (evilandi:4)
<< 22 other users at Sun Jan 11 19:30 BST >>


Version 1.5 - Install Disk
for PC's
Insert this disk in your floppy drive (A or B); set the DOS prompt
to the correct drive (A or B), and type INSTALL to boot.
America Online
Just Us
New Version
To Try A
F
NE


Google!
Search the web using Google!
10 results
Google Search
I'm feeling lucky
Index contains ~25 million pages (soon to be much bigger)
About Google!
Stanford Search Linux Search
Get Google! updates monthly!
your e-mail
Subscribe
Archive
Copyright ©1997-8 Stanford University


RAILS

Phoenix Framework

I NEED AN EMBEDDED SYSTEM AND I NEED IT IN 1 WEEK.

Element ID

this page intentionally left blank


ELEMENT

NERVES LET US CREATE AND DELIVER A PRODUCTION PRODUCT IN 5 DAYS WITHOUT SACRIFICING PERFORMANCE OR RELIABILITY.

Element ID

. . . UNLIKE OTHER EMBEDDED PLATFORMS WE'VE USED, MODIFICATIONS AND FEATURE ENHANCEMENTS ARE GOING TO BE EASY TO DO IN THE FUTURE.

Element ID

2 Web Developers

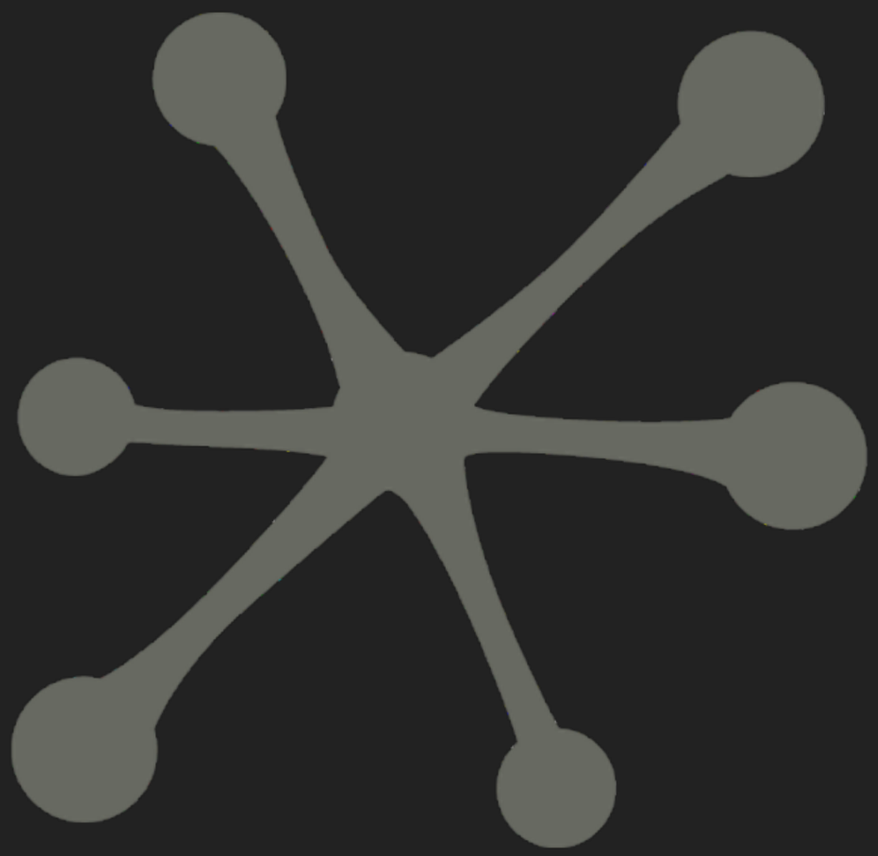
5 Days

Frank
Hunleth

Justin
Schneck

Garth
Hitchens

Community
elixir-lang slack #nerves



WHAT IS

NERVES

FRAMEWORKS

PLATFORM

TOOLING

FRAMEWORKS

nerves_led

nerves_networking

nerves_uart

elixir_ale

nerves_io_neopixel

nerves_ssdp_server

nerves_ssdp_client

nerves_hub

PLATFORM

nerves_system_ag150

nerves_system_alix

nerves_system_bbb

nerves_system_rpi

nerves_system_rpi2

nerves_system_rpi3

nerves_system_br

nerves_toolchain

nerves_toolchain_...

TOOLING

mix tasks

- mix nerves.new
- mix nerves.loadpaths
- mix nerves.precompile
- mix firmware
- mix firmware.burn

utilities

- fwup
- cell

SUPPORTED TARGETS	TARGET NAME
Raspberry Pi B / A+ /B+ / Zero	rpi
Raspberry Pi 2	rpi2
Raspberry Pi 3	rpi3
BeagleBone Black	bbb
Alix	alix
AG150	ag150
Intel Galileo 2	galileo
Lego EV3	ev3
QEmu Arm	qemu_arm

LINUX SINGLE BOARD COMPUTERS

GETTING STARTED

SANDBOX

RASBIAN / DEBIAN LINUX

- Update system
- Establish network
- SSH
- Install Erlang (ESL)
- Install Elixir
- Checkout Blinky
- mix run

RASBIAN / DEBIAN LINUX

THERE HAS TO BE A BETTER WAY!

GETTING STARTED

NERVES PLATFORM

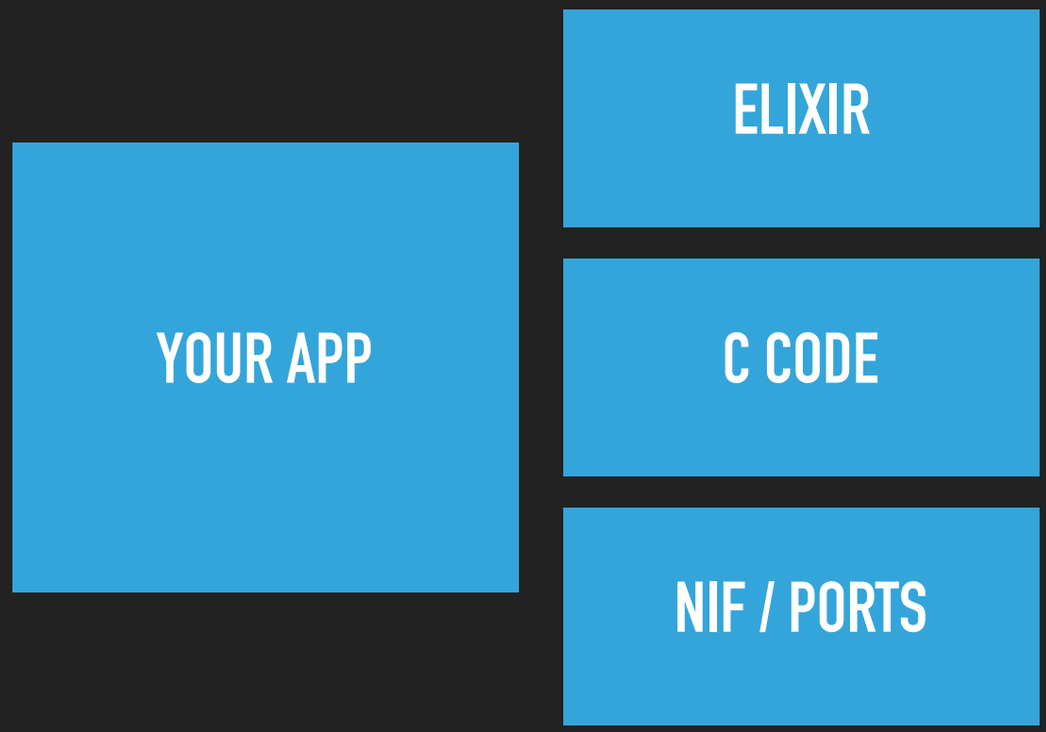
LETS MAKE THIS EASY

```
# install bake
Bakefile…
bake system get –target
bake toolchain get –target
bake firmware
bake burn
```

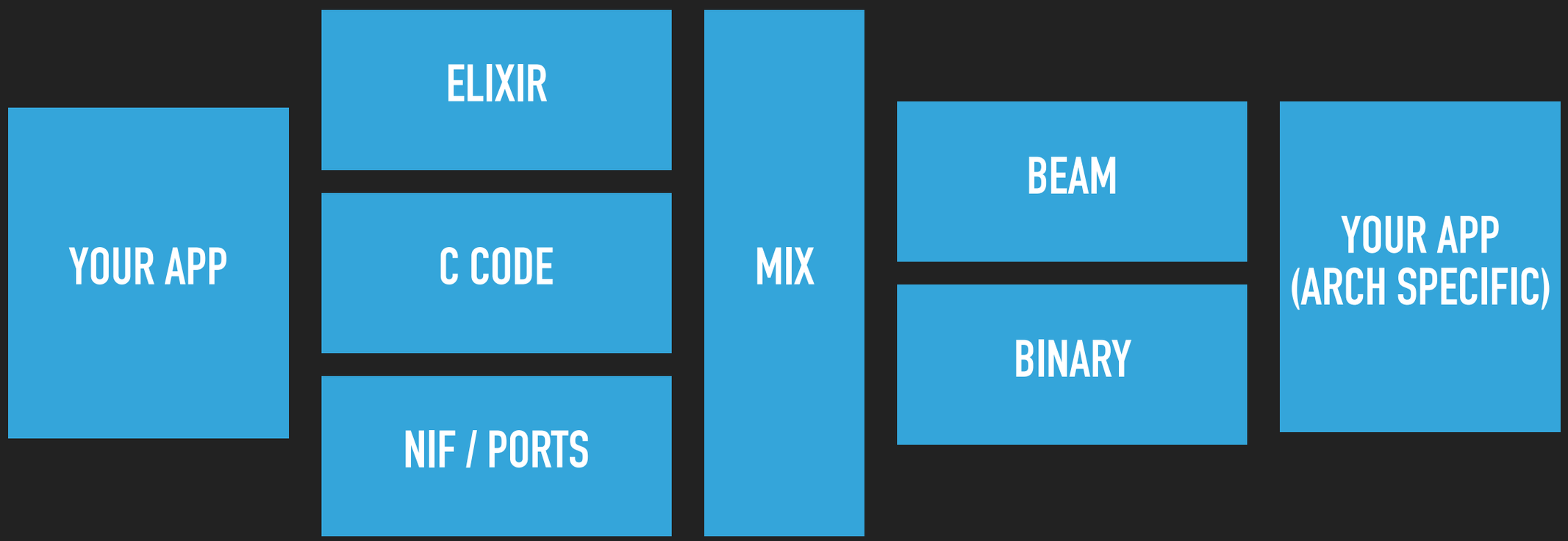
LETS MAKE THIS EASY

```
mix deps.get
mix firmware
mix firmware.burn
```

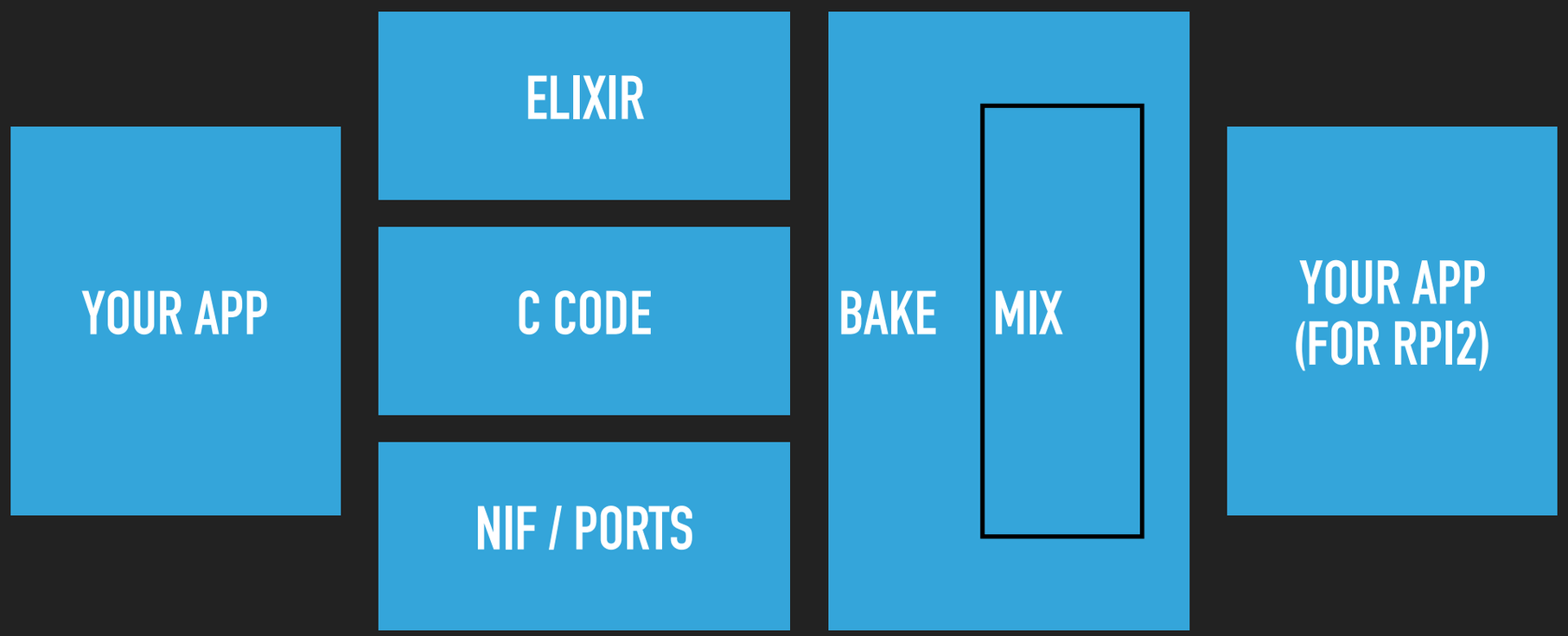
MIXING FIRMWARE



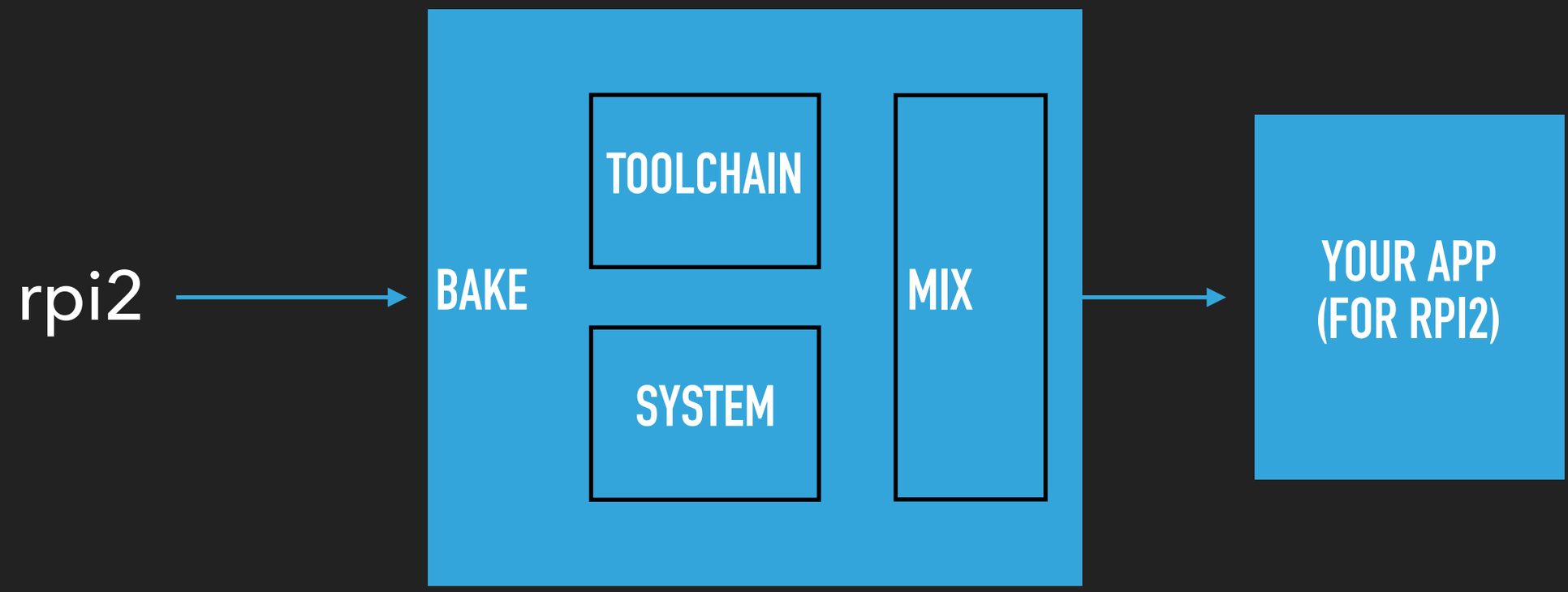
COMPILING ON YOUR MACHINE



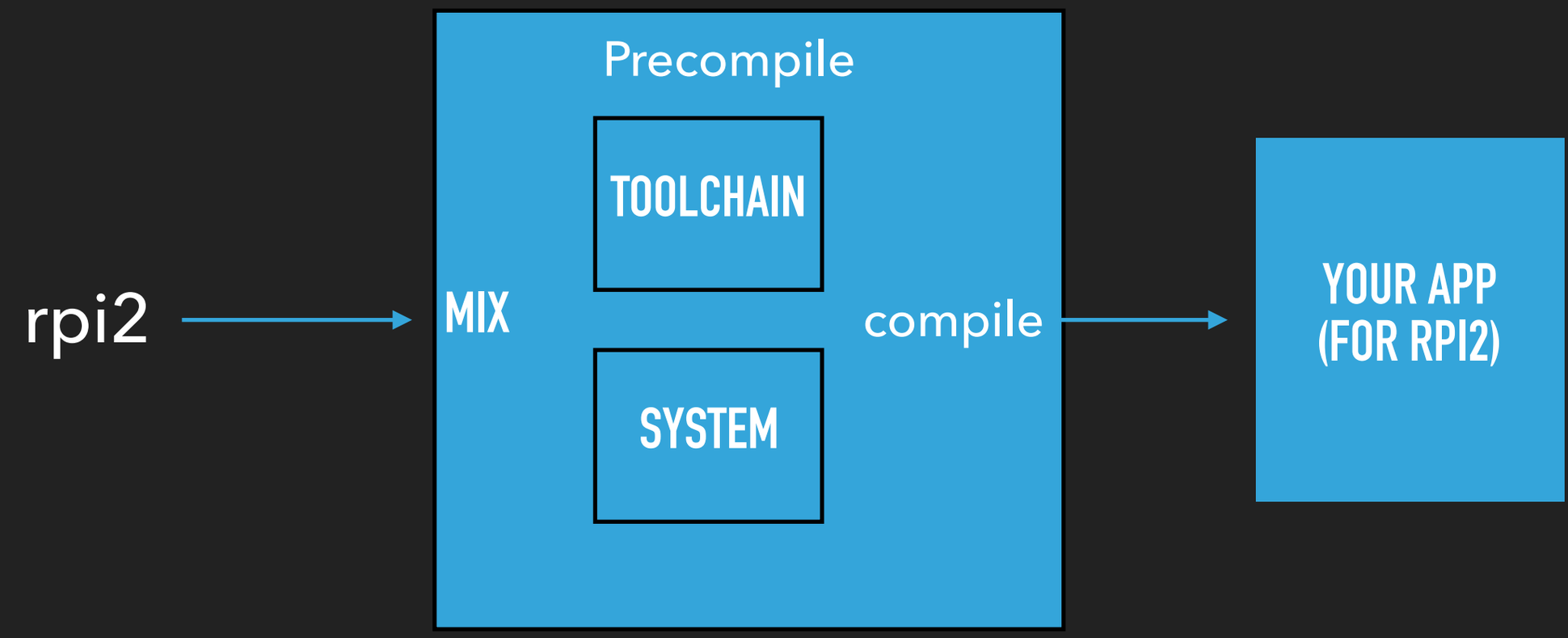
MIXING FIRMWARE WITH BAKE



MIXING FIRMWARE WITH BAKE



MIXING FIRMWARE



TOOLCHAINS

TOOLCHAIN CONFIG

- crosstool-ng
- for target
- host configs

TOOLCHAIN

- compilers
- run on host
- compile for target

TOOLCHAIN CONFIG

```
CT_LOCAL_TARBALLS_DIR="${CT_TOP_DIR}/../dl"
CT_SAVE_TARBALLS=y
CT_PREFIX_DIR="${CT_TOP_DIR}/../x-tools/${CT_TARGET}"
# CT_REMOVE_DOCS is not set
CT_LOG_EXTRA=y
CT_ARCH_FLOAT_HW=y
CT_ARCH_arm=y
CT_KERNEL_linux=y
CT_KERNEL_V_3_4=y
CT_BINUTILS_LINKER_LD_GOLD=y
CT_BINUTILS_GOLD_THREADS=y
CT_BINUTILS_LD_WRAPPER=y
CT_BINUTILS_PLUGINS=y
…
```

SYSTEMS

SYSTEM CONFIG

- buildroot
- defconfig
- rootfs-additions

SYSTEM

- bootfoles
- rootfs
- linux kernel

SYSTEM CONFIG

```
BR2_arm=y
BR2_cortex_a7=y
BR2_ARM_FPU_NEON_VFPV4=y
BR2_TOOLCHAIN_EXTERNAL=y
BR2_TOOLCHAIN_EXTERNAL_CUSTOM=y
BR2_TOOLCHAIN_EXTERNAL_DOWNLOAD=y
BR2_PACKAGE_NERVES_CONFIG_APPS="crypto"
BR2_PACKAGE_NERVES_CONFIG_EXTRA_MOUNTS="/dev/mmcblk0p3:/root:vfat
BR2_PACKAGE_NERVES_CONFIG_HANG_ON_EXIT=y
BR2_PACKAGE_NERVES_CONFIG_UNIQUEID_PROG="\"/usr/bin/boardid -b rp
4\""
BR2_PACKAGE_NERVES_CONFIG_HOSTNAME_PATTERN="nerves-%.4s"
BR2_PACKAGE_HOST_ERLANG_RELSYNC=y
BR2_PACKAGE_HOST_ERLANG_RELX=y
BR2_PACKAGE_HOST_FWUP=y
…
```

MIX LIFECYCLE

MIX deps.precompile

BOOTSTRAP

nerves_system

DEPS

phoenix

cowboy

nerves_system_rpi2

nerves_system_br

...

nerves.exs

```
config :nerves_system_rpi2, :nerves_env,
  type:  :system,
  mirrors: [
    "https://github.com/nerves-project/
nerves_system_rpi2/releases/download/
v#{version}/nerves_system_rpi2-
v#{version}.tar.gz"],
  build_platform: Nerves.System.Platforms.BR,
  build_config: [
    defconfig: "nerves_defconfig"
  ]
```

MIX LIFECYCLE

MIX deps.precompile deps.loadpaths compile

BOOTSTRAP

nerves_system

DEPS

phoenix

cowboy

nerves_system_rpi2

nerves_system_br

...

BOOTSTRAP

nerves_system

SYSTEM ENV

system

toolchain

NERVES ENV

toolchain

NERVES BOOTSTRAP

```
mix archive.install https://github.com/nerves-project/archives/raw/
master/nerves_bootstrap.ez
```

MIX FILE

```
defmodule Blinky.Mixfile do
  use Mix.Project

  @target System.get_env("NERVES_TARGET") || “rpi2"
  …
end
```

MIX FILE

```
defmodule Blinky.Mixfile do
…
def project do
    [app: :blinky,
     version: "0.1.0",
     archives: [nerves_bootstrap: "~> 0.1"],
     target: @target,
     deps_path: "deps/#{@target}",
     build_path: "_build/#{@target}",
     config_path: "config/#{@target}/config.exs",
     aliases: aliases,
     deps: deps ++ system(@target)]
  end
end
```

MIX FILE

```
defmodule Blinky.Mixfile do
…

  def system("rpi2") do
    [{:nerves_system_rpi2, “~> 0.4.0”}]
  end

  def aliases do
    ["deps.precompile": ["nerves.precompile", "deps.precompile"],
     "deps.loadpaths":  ["deps.loadpaths", "nerves.loadpaths"]]
  end

end
```

MIX FILE

```
defmodule Blinky.Mixfile do
…
  def system("rpi") do
    [{:nerves_system_rpi, "~> 0.4.0"}]
  end

  def system("rpi2") do
    [{:nerves_system_rpi2, “~> 0.4.0”}]
  end

  def system("rpi3") do
    [{:nerves_system_rpi3, “~> 0.4.0”}]
  end
…
end
```

CHANGING TARGETS

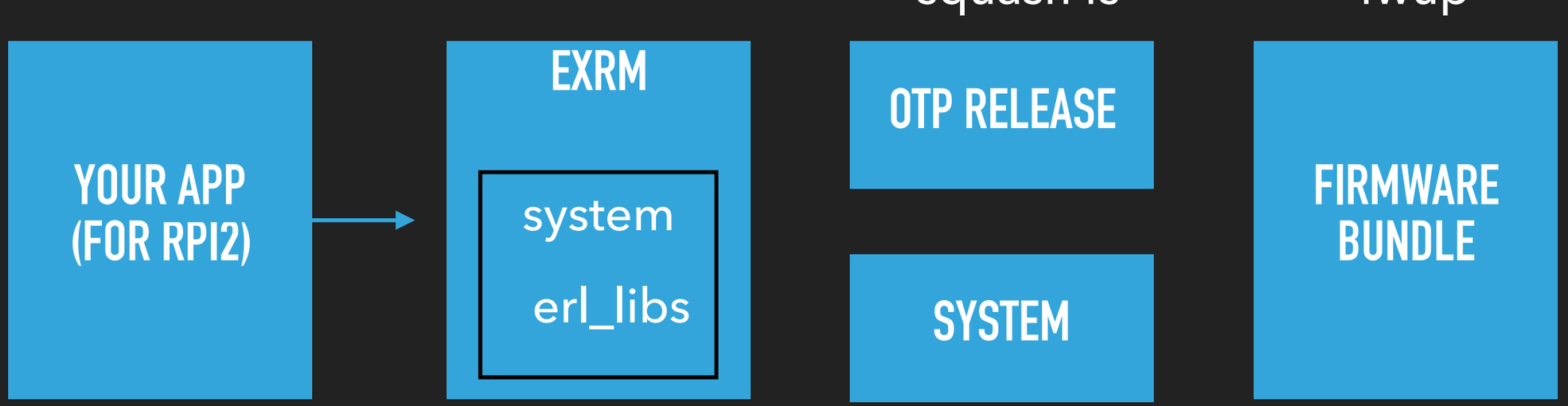
```
NERVES_TARGET=rpi3 mix deps.get
```

```
export NERVES_TARGET=rpi3
mix deps.get
```

```
mix deps.get
# @target System.get_env("NERVES_TARGET") || "rpi2"
```

MIX FIRMWARE

MIX ... compile firmware



MIX FIRMWARE

```
MIX ... compile firmware firmware.burn
```

fwup fwup

YOUR APP (FOR RPI2)

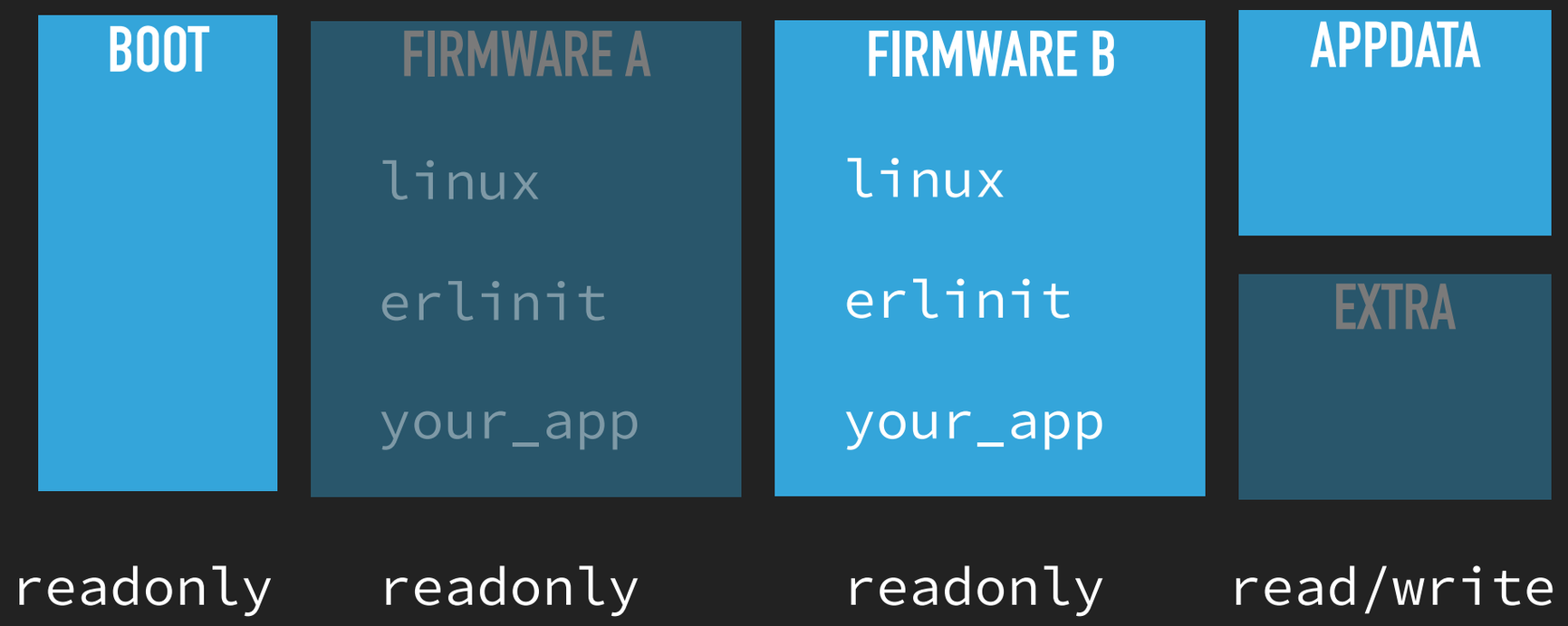
FIRMWARE BUNDLE

SD CARD/ FIRMWARE IMAGE

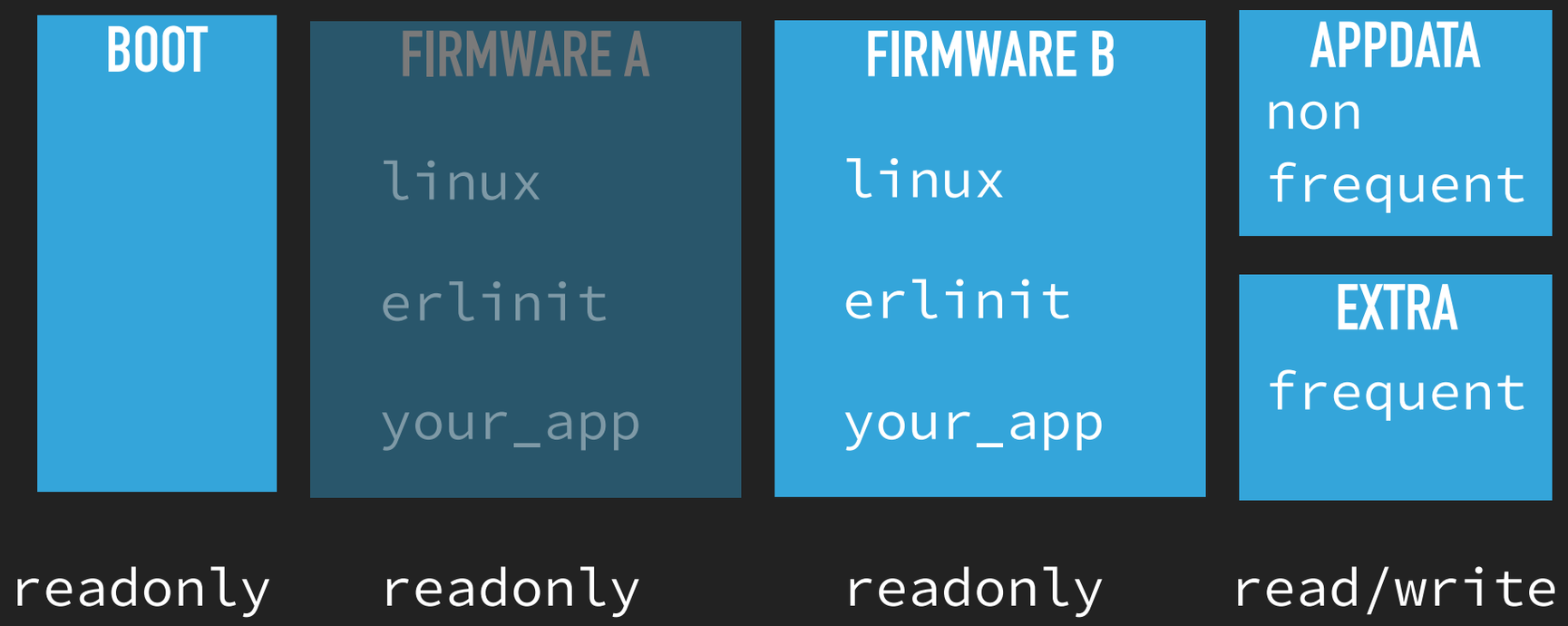
THE RESULT


BOOT
FIRMWARE A
linux
erlinit
your_app
FIRMWARE B
linux
erlinit
your_app
APPDATA
EXTRA
readonly
readonly
readonly
read/write

THE RESULT


BOOT
FIRMWARE A
linux
erlinit
your_app
FIRMWARE B
linux
erlinit
your_app
APPDATA
EXTRA
readonly
readonly
readonly
read/write

THE RESULT


BOOT
FIRMWARE A
linux
erlinit
your_app
FIRMWARE B
linux
erlinit
your_app
APPDATA
non
frequent
EXTRA
frequent
readonly
readonly
readonly
read/write

GETTING STARTED

NERVES FRAMEWORK

NETWORKING

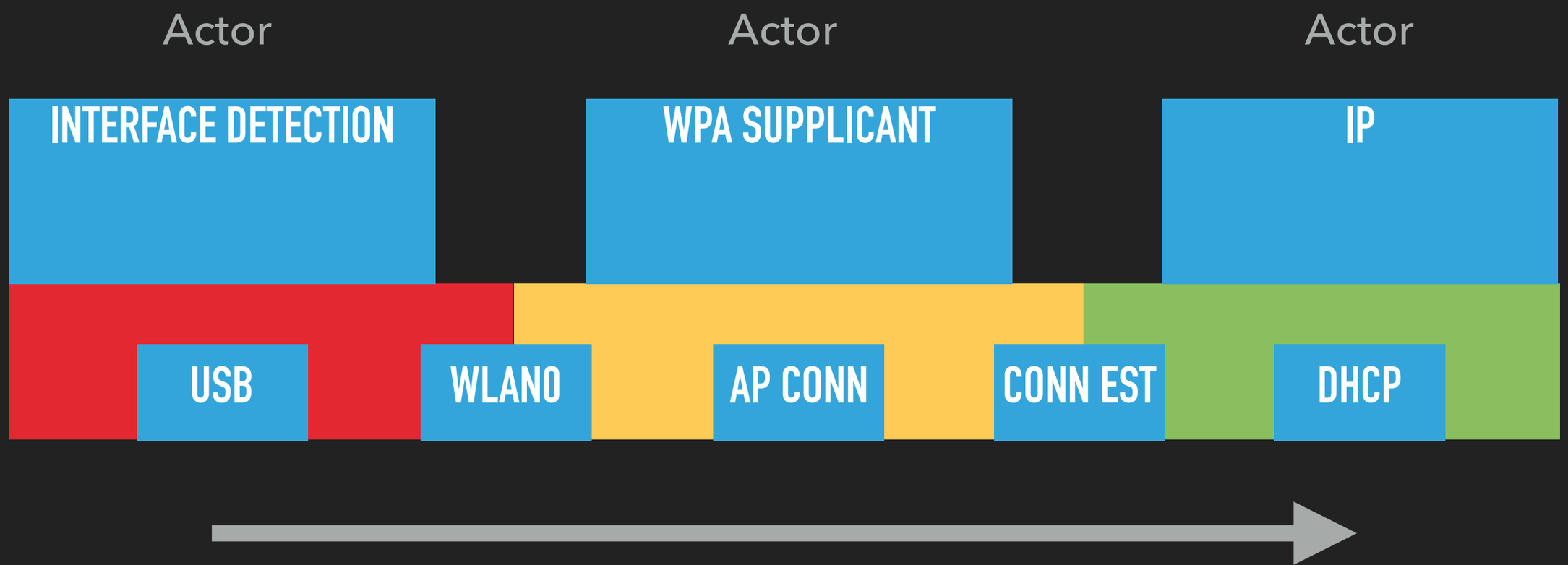
```
{:nerves_networking, “~> 0.5.0”}

{:ok, _} = Networking.setup :eth0
```

```
mode: "static",
ip: "10.0.0.5",
router: "10.0.0.1",
mask: "16",
subnet: "255.255.0.0",
mode: "static",
dns: "8.8.8.8 8.8.4.4",
hostname: "myhost"
```

WIFI

```
{:nerves_wifi, “~> 0.1.0”}
```

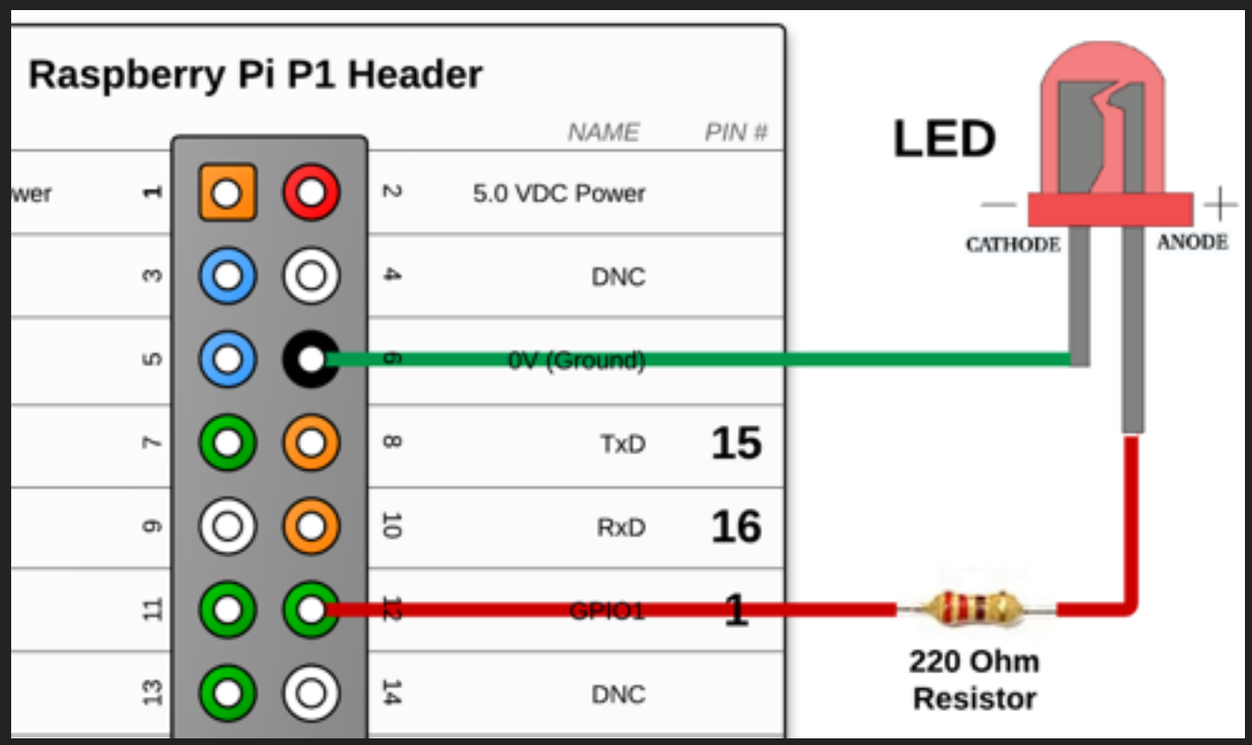
WIFI



Initialization Transitions

INTERACTING WITH HARDWARE

```
{:elixir_ale, “~> 0.4.0”}

{:ok, pid} = Gpio.start_link(1, :output)
Gpio.write(pid, 1)
```

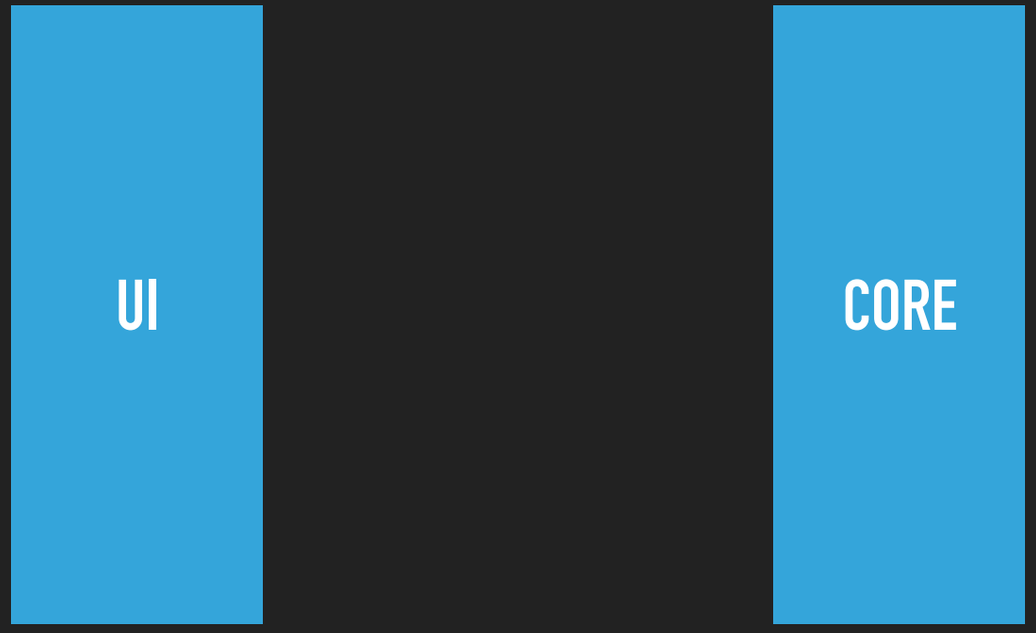


GETTING STARTED

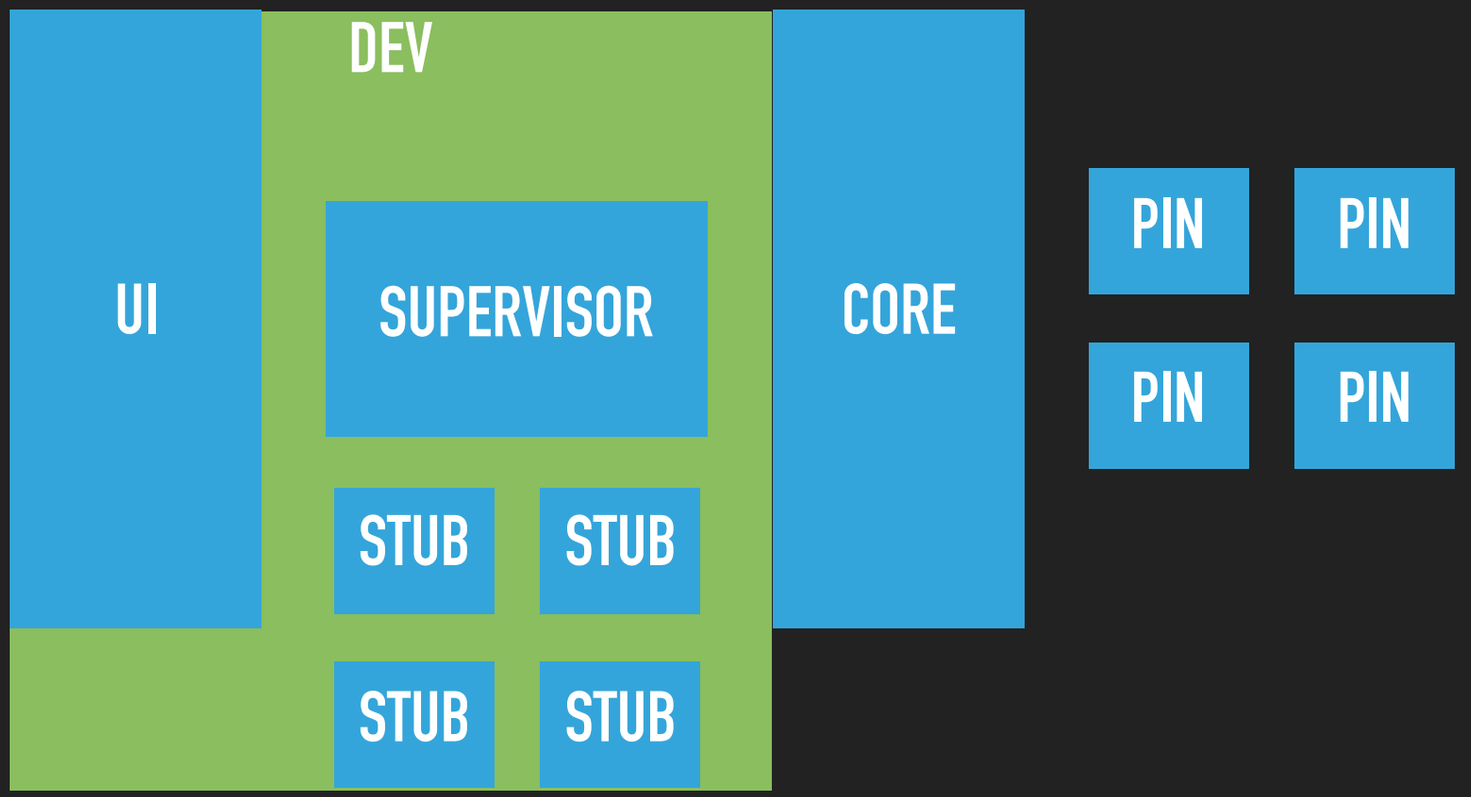
USER INTERFACES

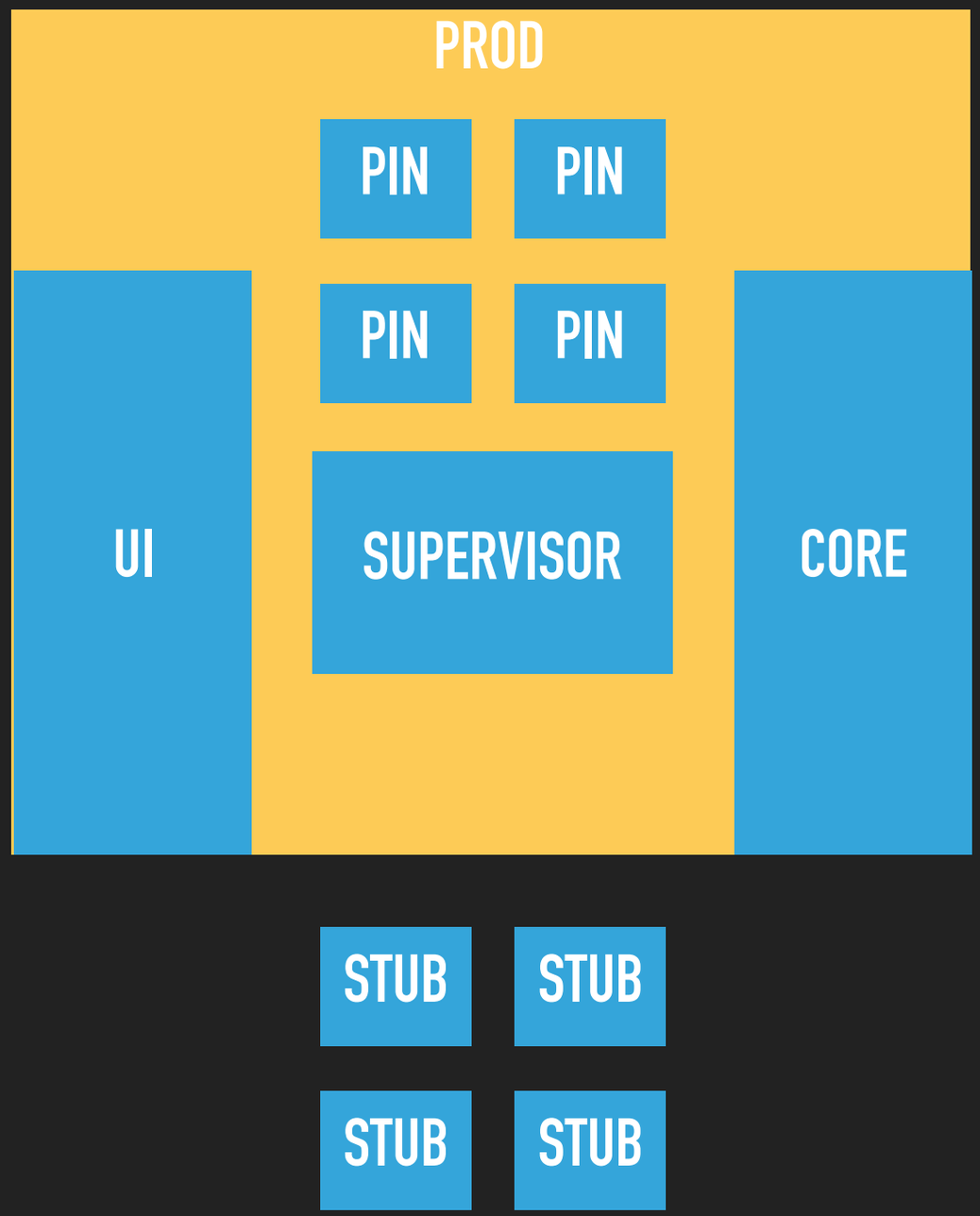
PHOENIX FOR WEB ADMIN

```
your_app_umbrella
|- your_app_nerves
|- your_app_ui
```



GETTING STARTED - USER INTERFACES




PROD
PIN
PIN
PIN
PIN
UI
SUPERVISOR
CORE
STUB
STUB
STUB
STUB

Element IO

- Home
- Configuration
- Pinmuxing
- Actions
- Digital Events
- Interval Events

Reboot Device

Download Log | Clear Log

ELEMENT ID®

Uptime: 0:0:01:56

Pin 1 ON
Pin 1 OFF

Counters

pin 1 monitor: 1

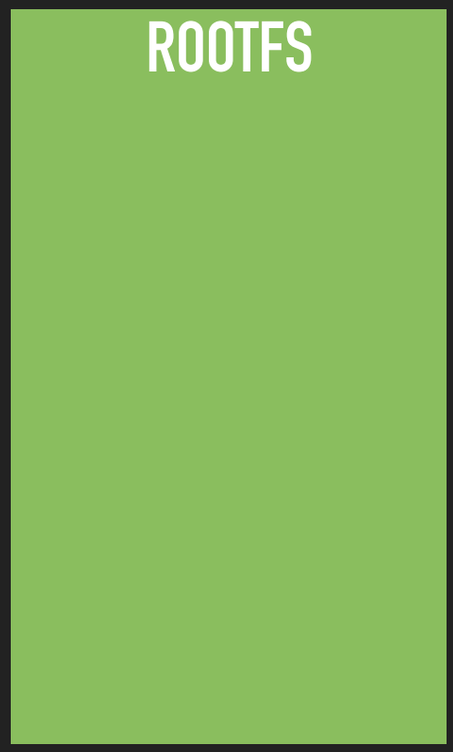
ADVANCED

NERVES FIRMWARE

ADDING FILES TO THE ROOT FILE SYSTEM

```
config :nerves, :firmware,
  rootfs_additions: "config/rpi2/rootfs-additions"
```

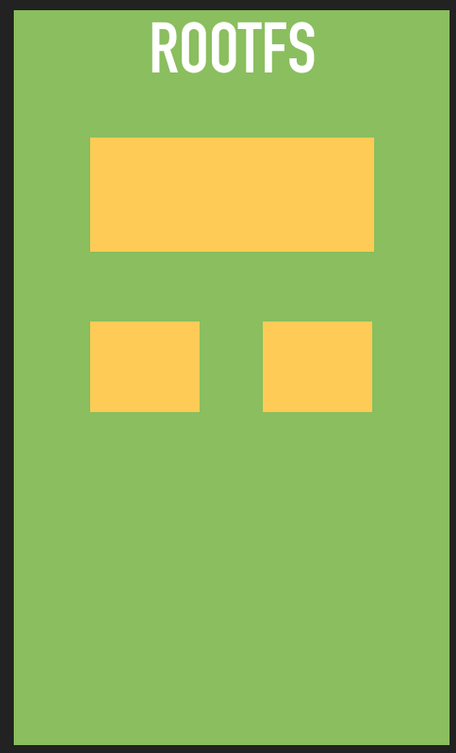
```
rootfs-additions
|- etc
  |- my_utility.conf
```



ADDING FILES TO THE ROOT FILE SYSTEM

```
config :nerves, :firmware,
  rootfs_additions: "config/rpi2/rootfs-additions"
```

```
rootfs-additions
|- etc
  |- my_utility.conf
```

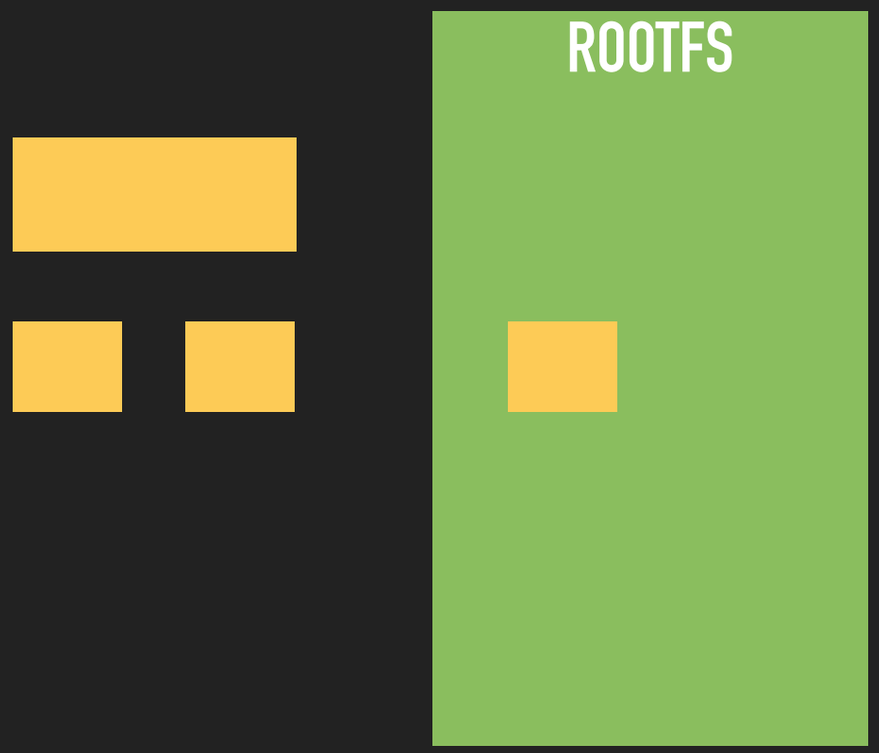


CHANGING FILES ON ROOT FILESYSTEM

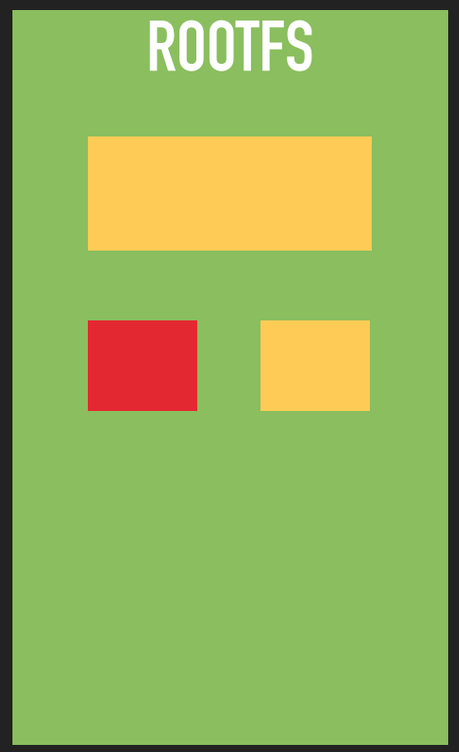
```
config :nerves, :firmware,
  rootfs_additions: "config/rpi2/rootfs-additions"
```

```
rootfs-additions
|- etc
  |- erlinit.conf

# Uncomment to hang the board rather than rebooting when
Erlang exits
#--hang-on-exit
```

CHANGING FILES ON ROOT FILESYSTEM



CHANGING FILES ON ROOT FILESYSTEM



CHANGING FILES ON BOOT PARTITION

CHANGING FILES ON BOOT PARTITION

```
# config/rpi2/config.exs
config :nerves, :firmware,
  fwup_conf: "config/rpi2/fwup.conf",
```

```
# config/rpi2/fwup.conf
file-resource cmdline.txt {
    host-path = “${NERVES_APP}/config/rpi2/cmdline.txt"
}

# config/rpi2/cmdline.txt
console=tty1 console=ttyS0,115200 root=/dev/mmcblk0p2 rootwait
```

CHANGING FIRMWARE PARTITIONS

```
+-----------------------------+
| MBR                         |
+-----------------------------+
| p0: Boot partition (FAT32)  |
| zImage, bootcode.bin,       |
| config.txt, etc.            |
+-----------------------------+
| p1*: Rootfs A (squashfs)    |
+-----------------------------+
| p1*: Rootfs B (squashfs)    |
+-----------------------------+
| p2: Application (FAT32)     |
+-----------------------------+
```

CHANGING FIRMWARE PARTITIONS

```
# Log partition
define(LOG_PART_OFFSET, 1643048)
define(LOG_PART_COUNT, 1048576)

partition 3 {
        block-offset = ${LOG_PART_OFFSET}
        block-count = ${LOG_PART_COUNT}
        type = 0x83 # Linux
    }
```

CONNECTING TO REMOTE NODES

WHATS AHEAD

NERVES PROJECT

THE FUTURE-ISH

Display / Touch Screen support

Network Firmware Update

Develop Lifecycle

- Always connected target nodes
- Target Distributed ExText
- Development Kits

Easier IO for connecting to arduinos

Video camera support

Better Docs

Tutorials

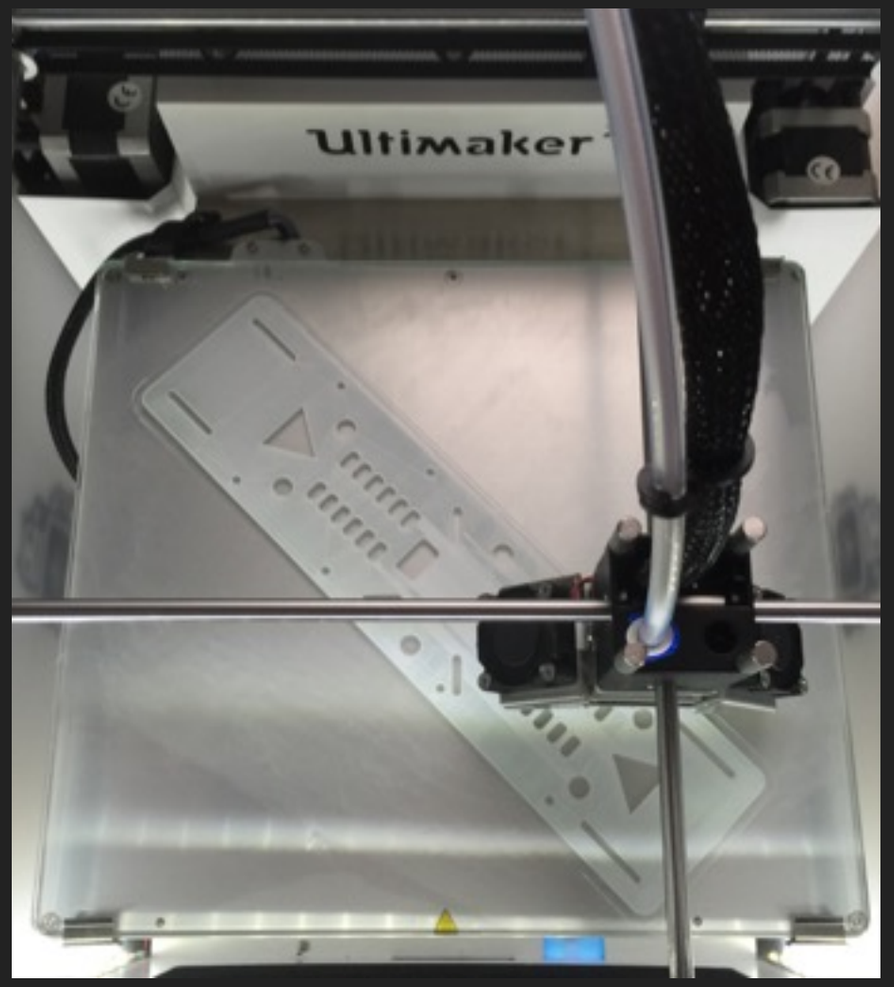
Videos

Books

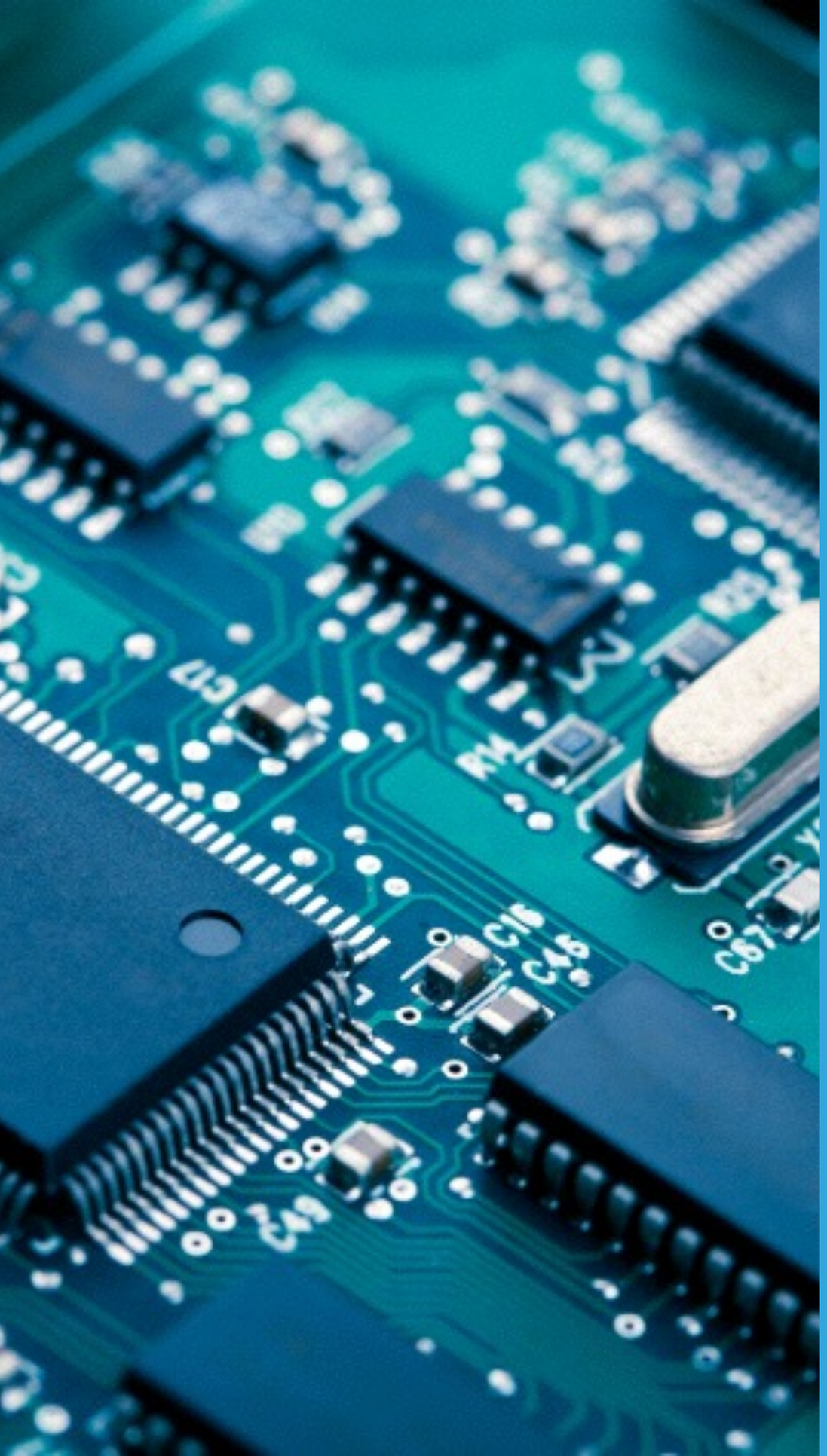
NERVES EMBEDDED SYSTEMS

THE REVOLUTION

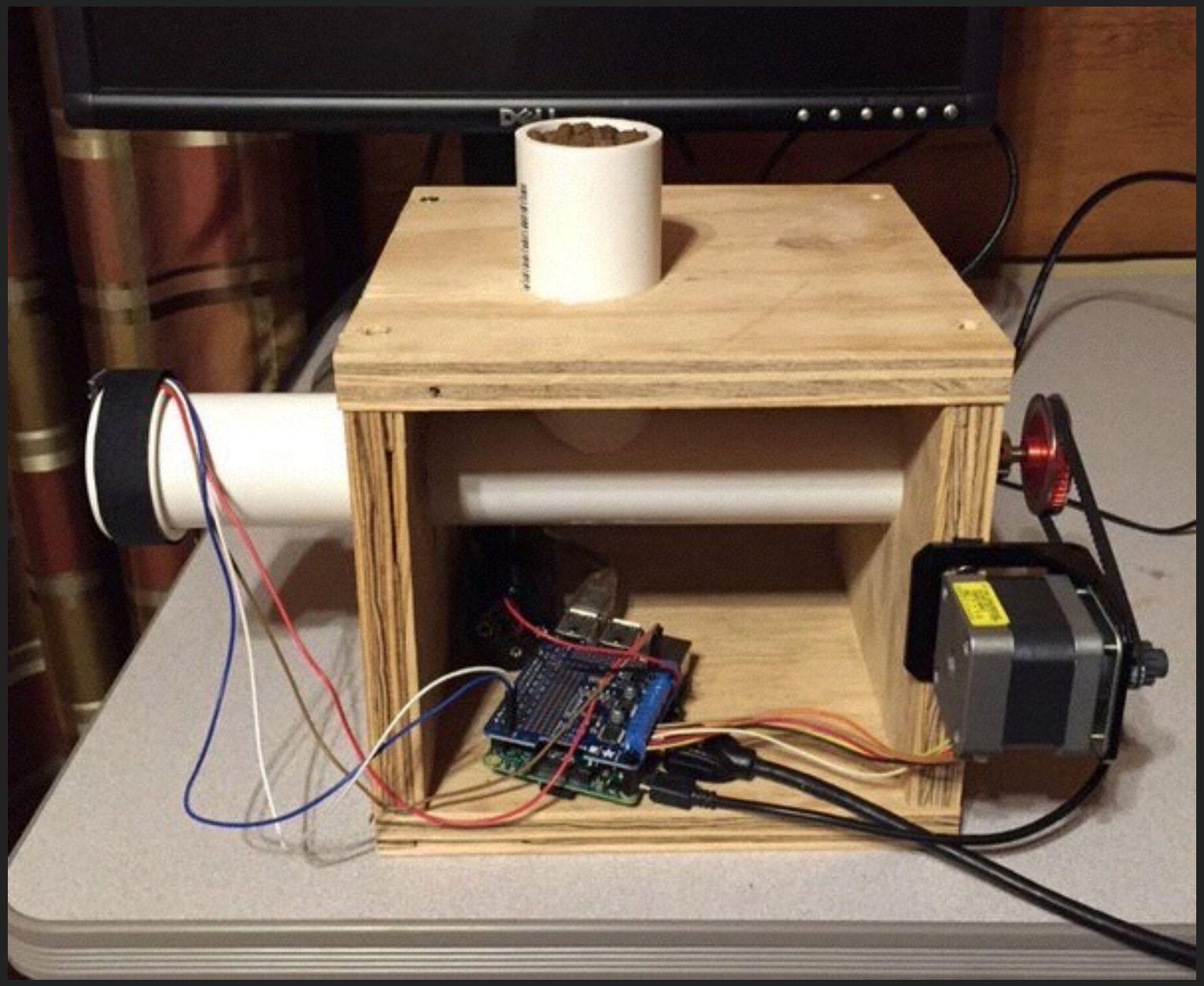
CHANGE THE WORLD




STOP



LETS REWRITE EMBEDDED



@wsmoak

@diptimmo

2016

2016

Justin Schneck

@mobileoverlord

@nervesproject