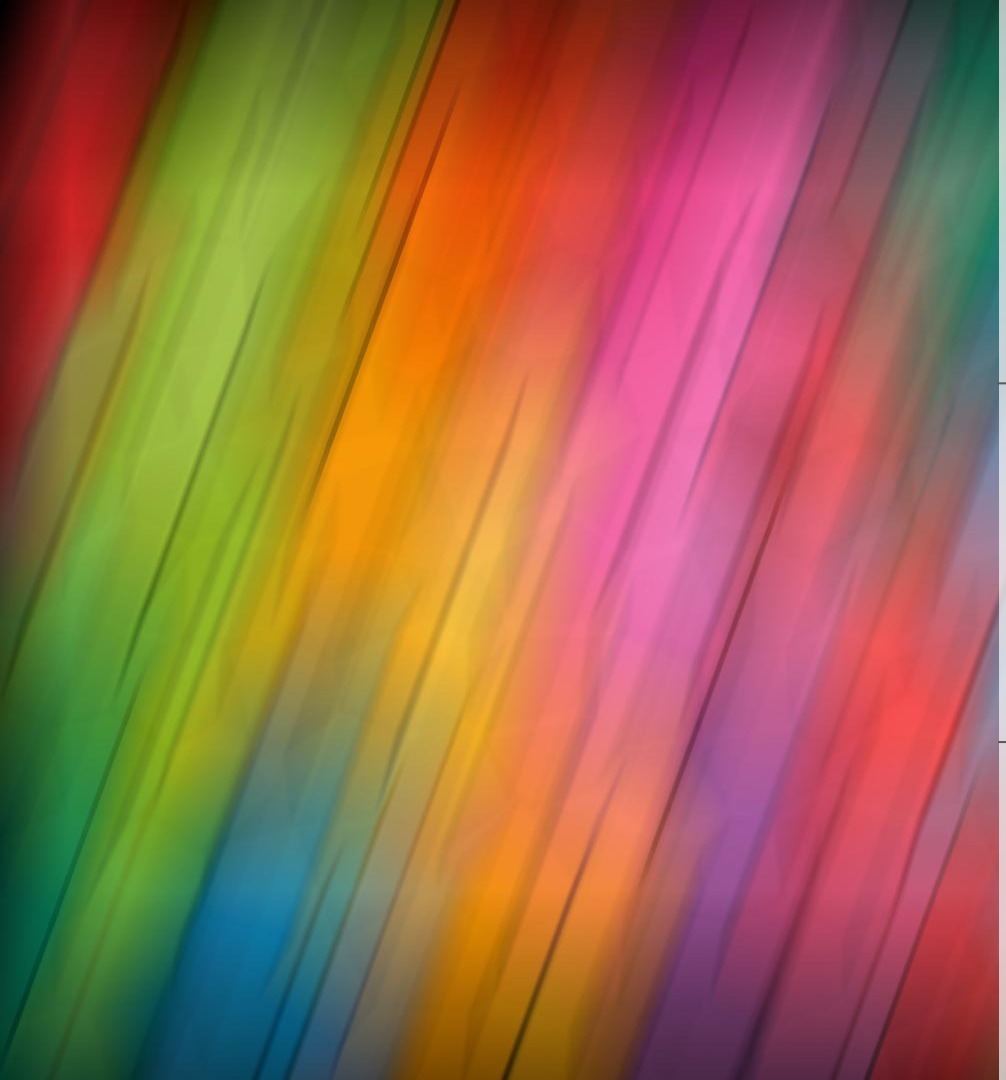


AmbiLight

SoC2018

B. Gigerl • F. Kargl • P. Nasahl

A. Trinker • M. Lipp

L. Macan • S. Steinegger

YOUR TV IS SO BORING THAT YOU ARE READING THE NEWSPAPER?


ON
Carol Perez Trevino
Kris Frazier
is watching 'Built' on Verizon FiOS TV
Orly Angelo
Ryan Trees
Renee Tidwell
OR CHECKING FACEBOOK?

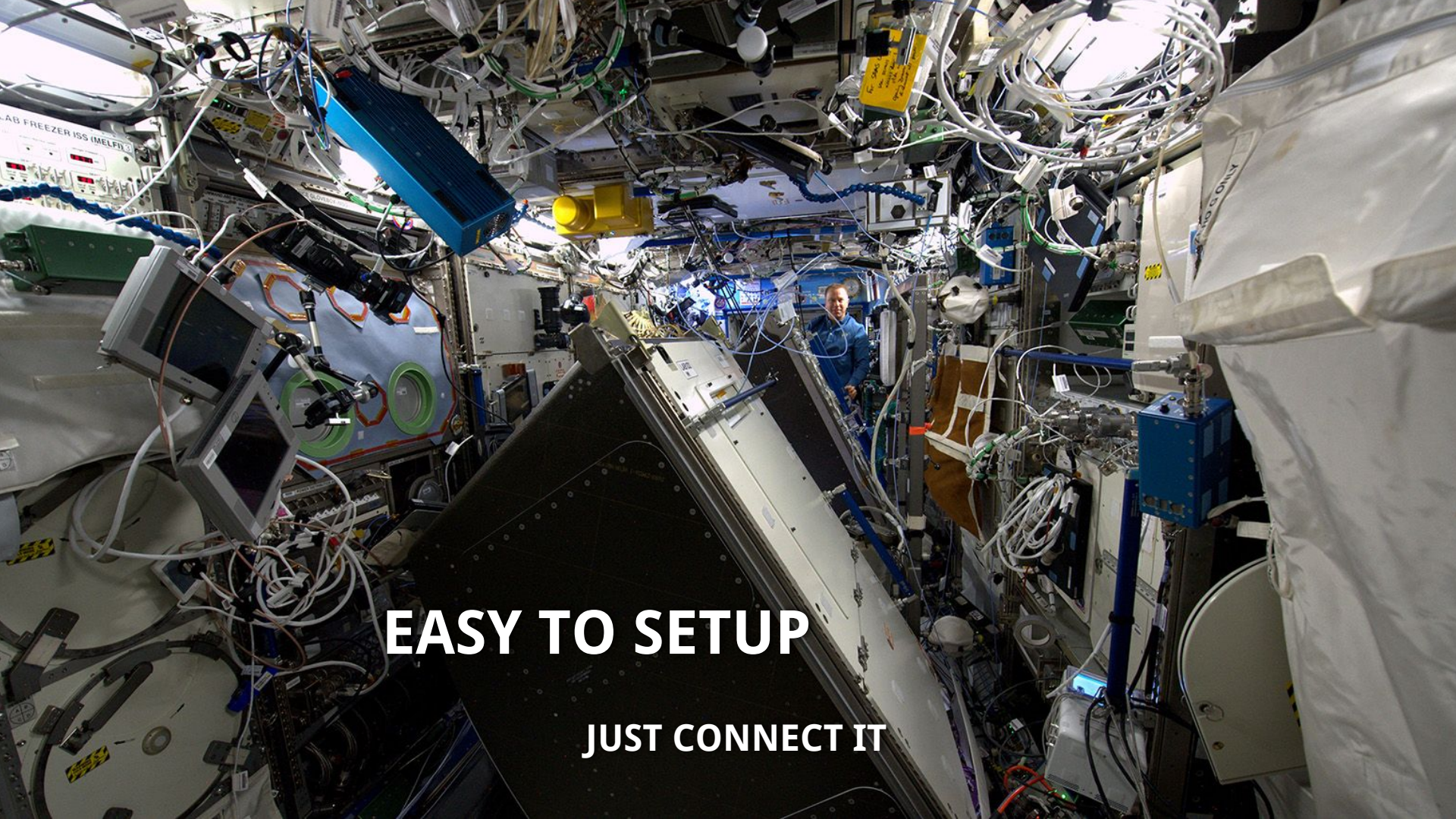
GET READY …

WE MAKE TV GREAT AGAIN

AMBILIGHT

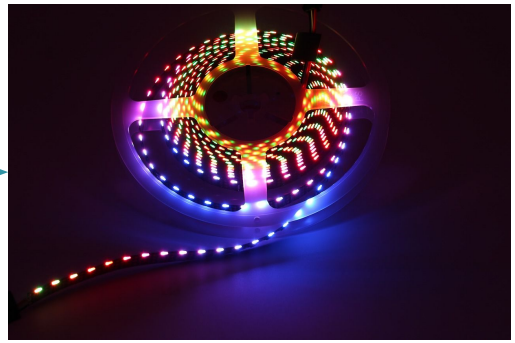
WATCH TV AS YOU NEVER DID BEFORE

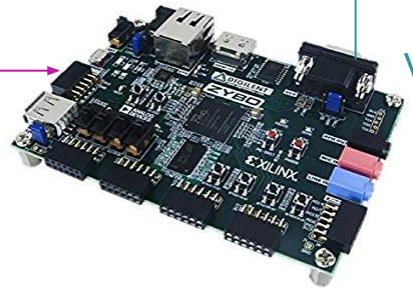

EASY TO USE
CONTROL IT WITH YOUR SMARTPHONE


EASY TO SETUP
JUST CONNECT IT


PURE JOY
YOU NEVER WANT TO GET UP AGAIN

Overview

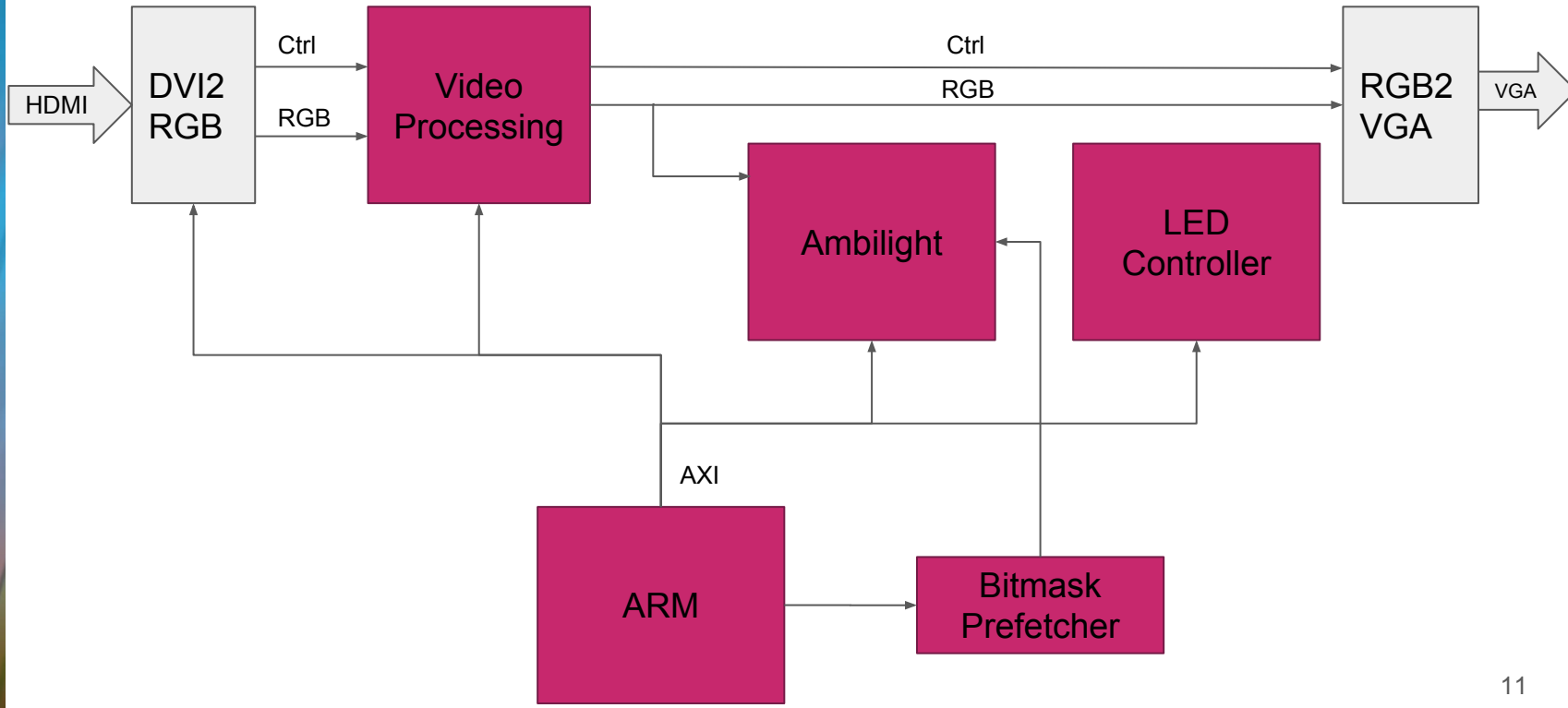




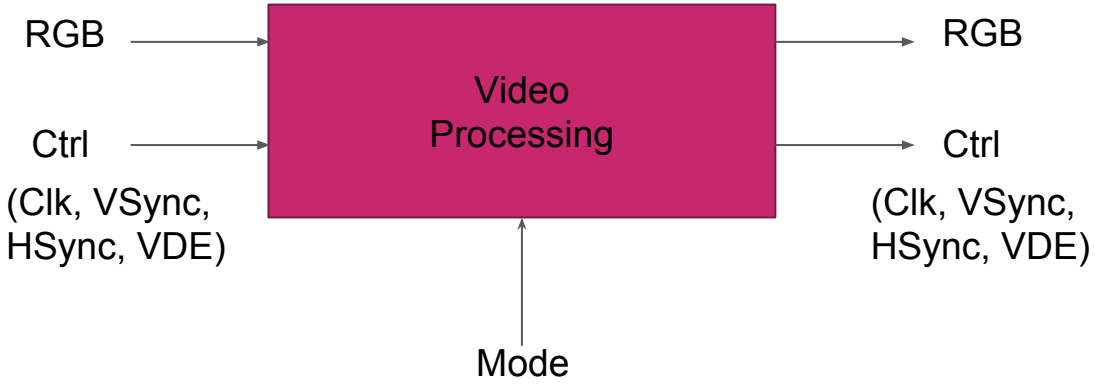
HDMI

VGA

Architecture Overview



HW: Video Processing



1. Passthrough
2. Grayscale
3. Dreamy Frame Effect
4. Image Negative
5. Color correction
6. Pink Frame
7. Color binning

HW: Video Processing

3. Dreamy Frame Effect

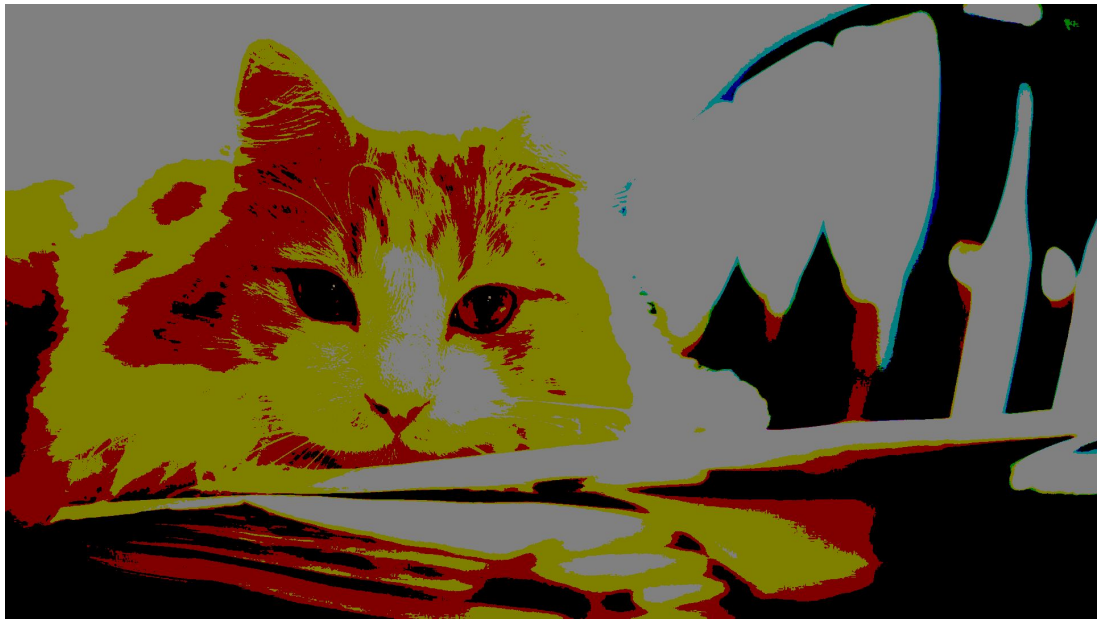
Alpha blending: `out = b * in + (255-b) * 255`

HW: Video Processing

5. Color correction

```
out = c * in + b
```

HW: Video Processing



7. Color binning

HW: AmbiLight

- Frame divided into zones
- Average pixels per zone
- Pixel to zone mapping with mask
  - Any shape
  - Zone transitions
- ~16 MiB for 8-bit mask
- Only 240 KB Block RAM on Zybo

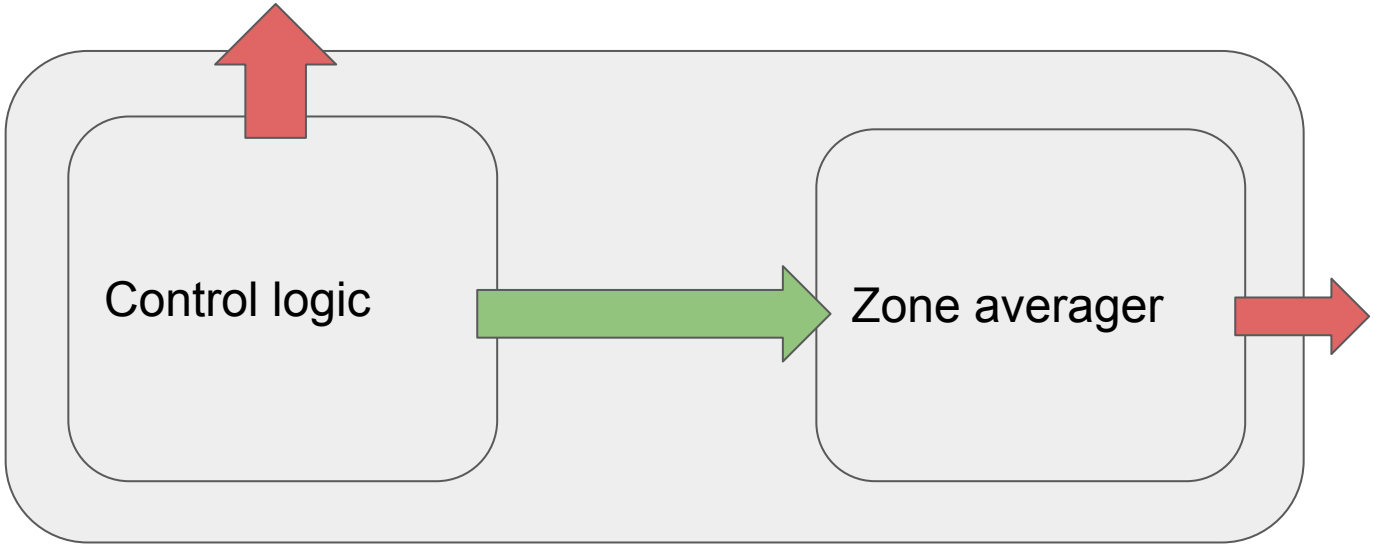
1	1	1	0	0	2	2	2
1	1	1	0	0	2	2	2
1	1	1	0	0	2	2	2
0	0	0	0	0	0	0	0
3	3	3	4	3	4	4	4
3	3	3	4	3	4	4	4
3	3	3	4	3	4	4	4

HW: AmbiLight

- Solution:
- Runtime encoding 16 bit
  - Zone (8 bit) | Length (7 bit) | Zone_End (1 bit)
  - 1 pixel zones unlikely
  - Saves Memory
- Stored in on-board DRAM
  - 512MiB available
- DRAM accessible with AXI3
- Prefetch and cache 16 bit words
  - 16 encoded words
  - DRAM and AXI latency

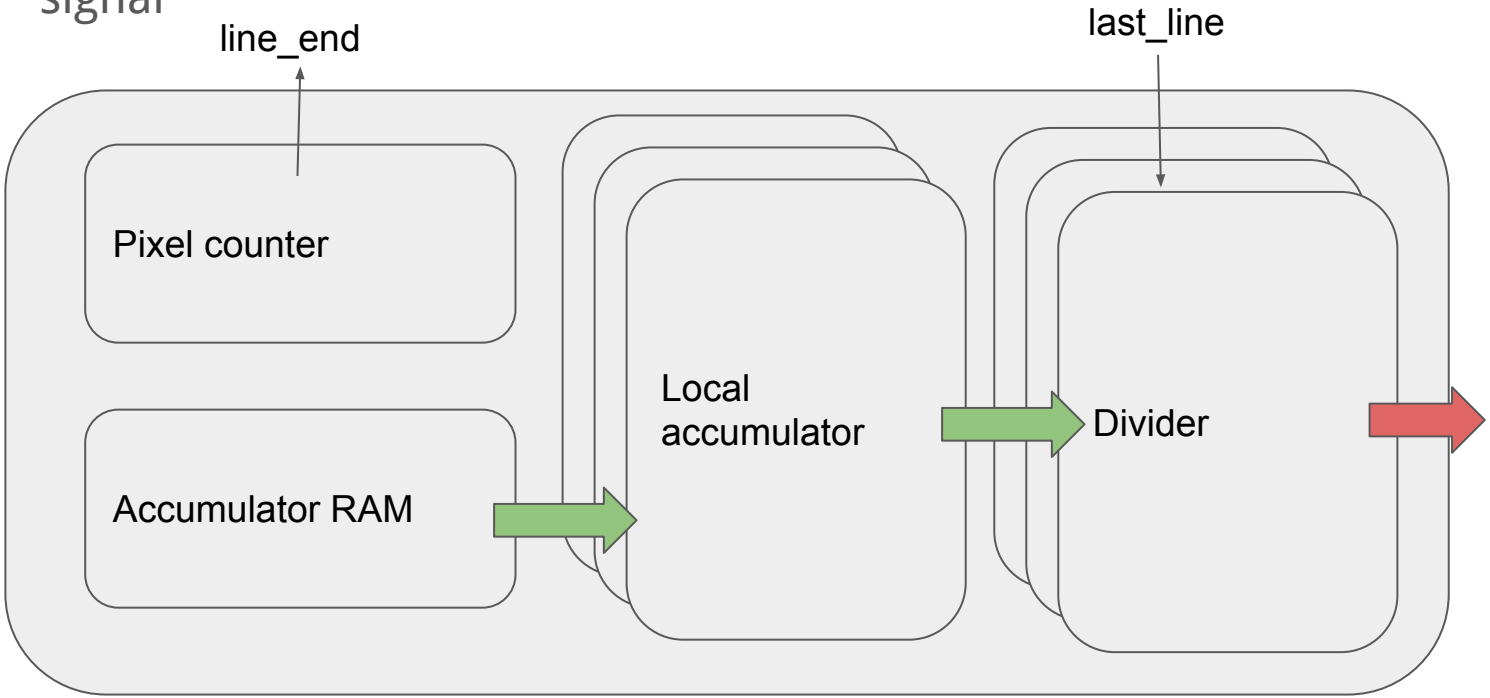
HW: AmbiLight

- Tasks:
  - Average the zones pixel value
  - Control the LED controller and DRAM prefetcher



HW: Zone averager

- Produces average value of the zone and additional handshaking signal



HW: Ambilight - Challenges

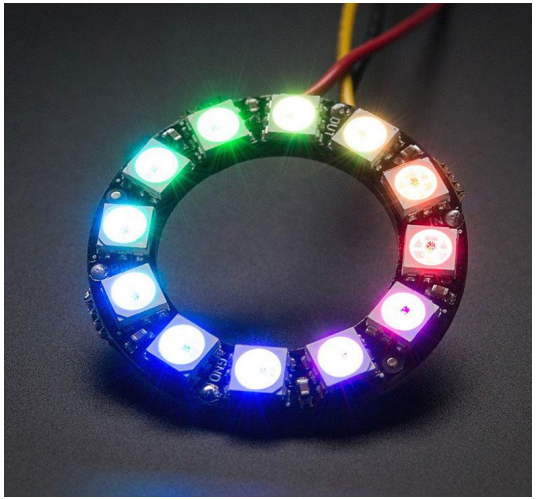
- Writing control logic that synchronises and initializes all the modules
  - When to initiate next read so that the data is ready in time for the Zone averager?
- Storing data about previous accumulations in a way that we can access it in 1 clock

HW: Ambilight - Issues

- Using 2 clocks between modules
  - Possible timing issues

HW: LED controller

- “framebuffer”
  - up to 250 RGB(W) LEDs
  - dual-port BRAM
- output encoder
  - PWM-based single-wire encoding
- CTRL core
  - state machine
  - “start” signal, “done” flag
- AXI4-lite interface
  - framebuffer + CTRL

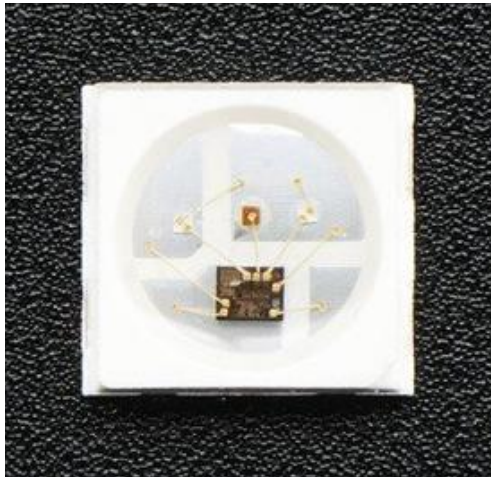


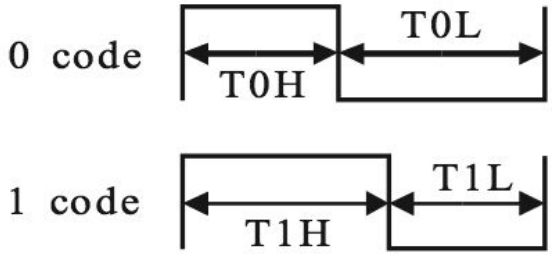
HW: LED controller

- “NEOPIXEL” WS2812B
  - R+G+B LEDs + controller die
  - +5V, GND, SI, SO, no CLK!
  - 800kHz PWM interface
  - strict timing requirements !
  - or not...
  - daisy chained

- 144 LED/m strip
  - 1.25us per PWM cycle → 30us per LED
  - ~5ms per strip → 200Hz refresh rate

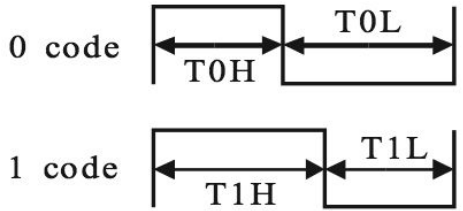
  - 20mA per color → 60mA per LED
  - up to 9A per strip ! (5V)





HW: LED controller

- 1bit per PWM cycle
- “GRB” LED, MSb first
- first 24bit → first LED, rest ist sent on
- refresh: disable PWM for ~”48bits”



Composition of 24bit data:

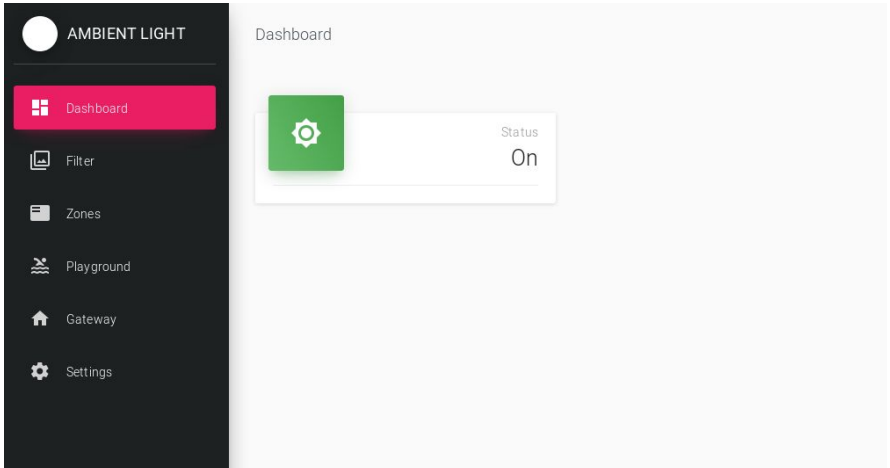
G7	G6	G5	G4	G3	G2	G1	G0	R7	R6	R5	R4	R3	R2	R1	R0	B7	B6	B5	B4	B3	B2	B1	B0

Data transfer time (TH+TL=1.25μs±300ns)

T0H	0 code ,high voltage time	0.4us	±150ns
T1H	1 code ,high voltage time	0.8us	±150ns
T0L	0 code , low voltage time	0.85us	±150ns
T1L	1 code ,low voltage time	0.45us	±150ns
RES	low voltage time	Above 50μs	

SW: Web Interface

- Modern web frontend using Vue.js
  - Mobile or desktop application
- Allows to control the AmbientLight
  - Set filters
  - Set zones
  - Control light bulb



SW: RESTful API - Backend

- Provides a RESTful API
  - Implemented in Python using Flask
  - Runs on the AmbientLight
- Handles requests of the frontend
- Stores settings in a database
- Communicates with the kernel module

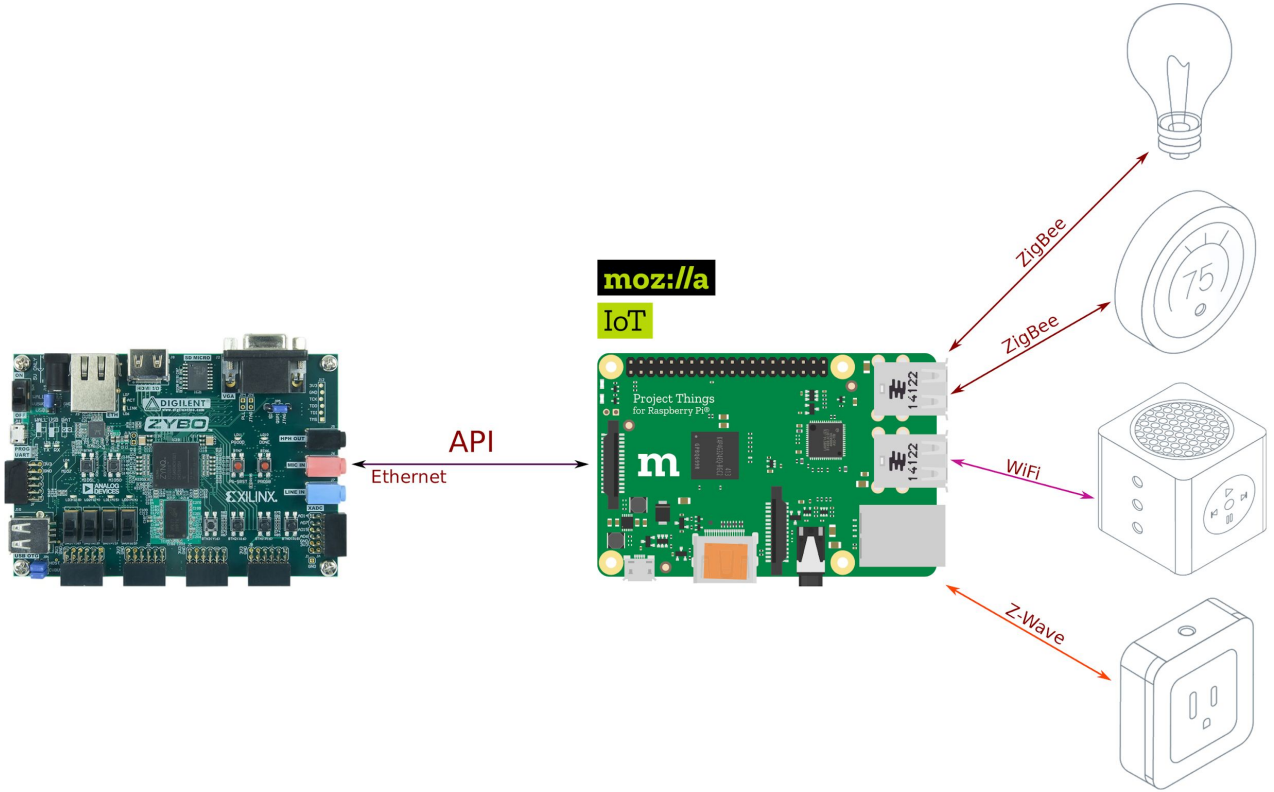
SW: Kernel Module

- Kernel module that can be controlled using IOCTL
- Dynamically detects which controllers of the AmbientLight are available
  - Allows easier implementation and testing
- Handles the requests of the backend or example applications
- Offers commands to
  - Configure everything
  - Set lights manually
  - ...

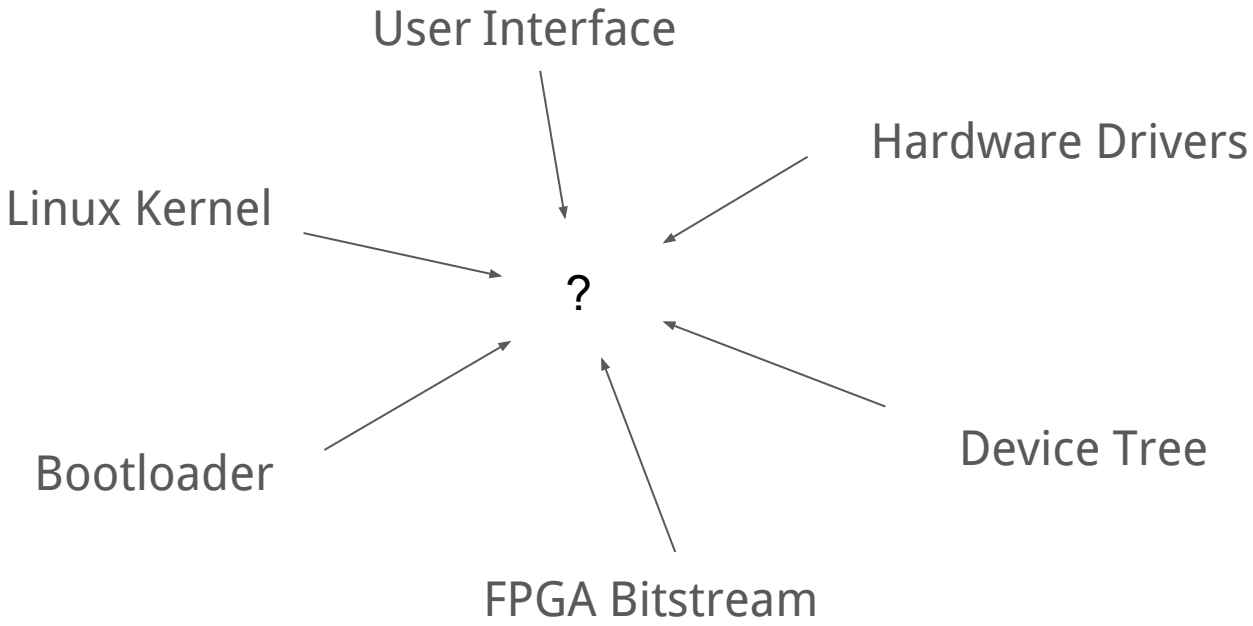
SW: Home Gateway

- Goal: Switch off bulbs when movie starts
- Using proprietary gateways:
  - APIs (often) available
  - Supported devices?
  - Privacy, security?
- Implementing ZigBee stack is not that easy

SW: Home Gateway

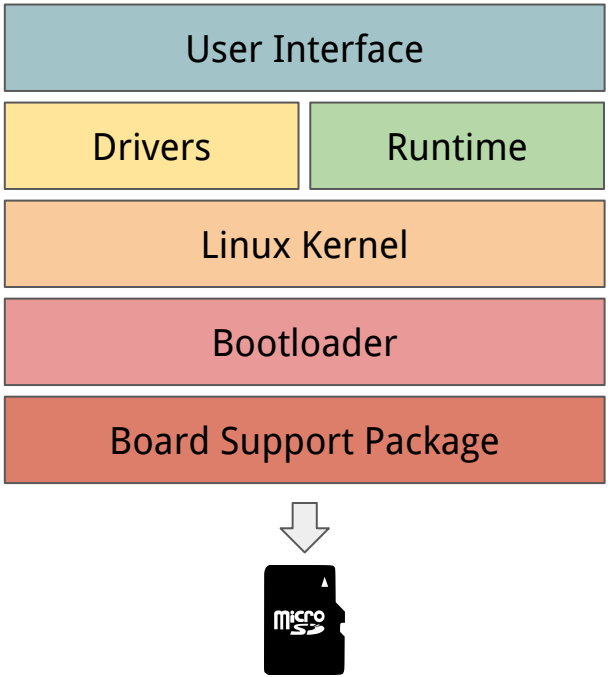


Putting it all together

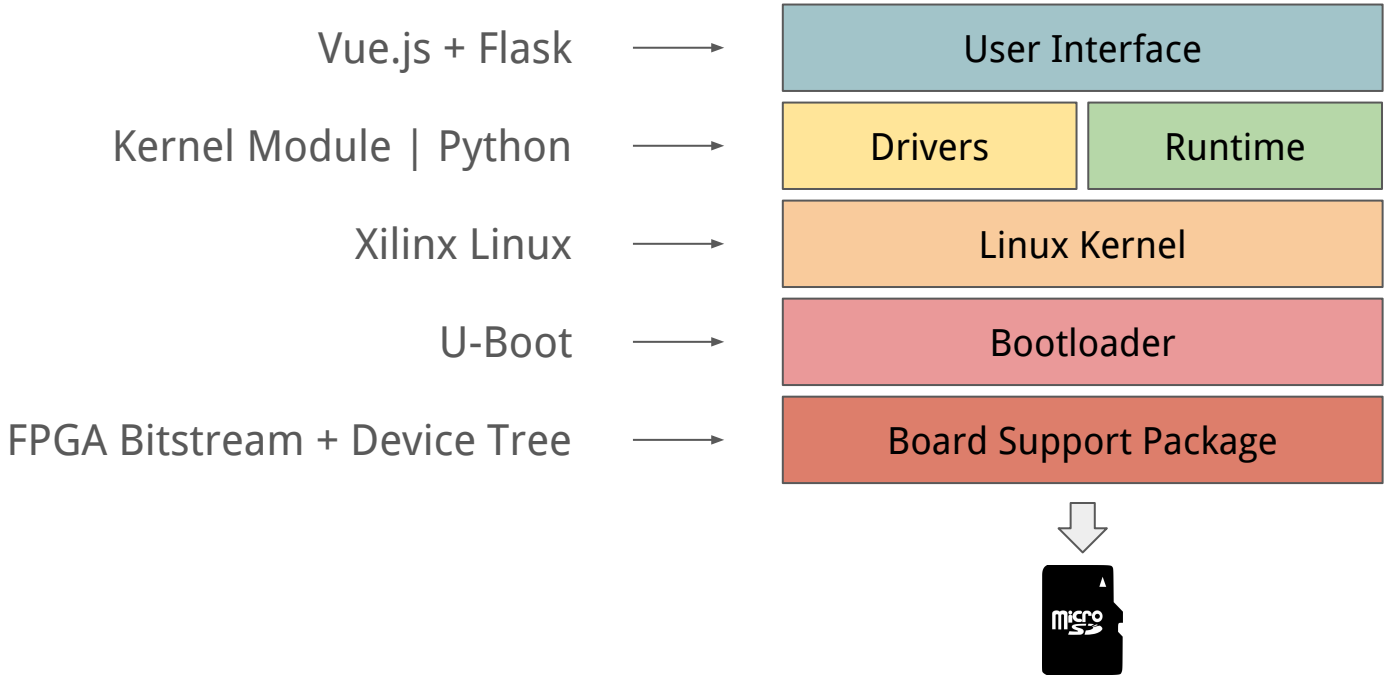


Yocto - “LEGO for Embedded Systems”

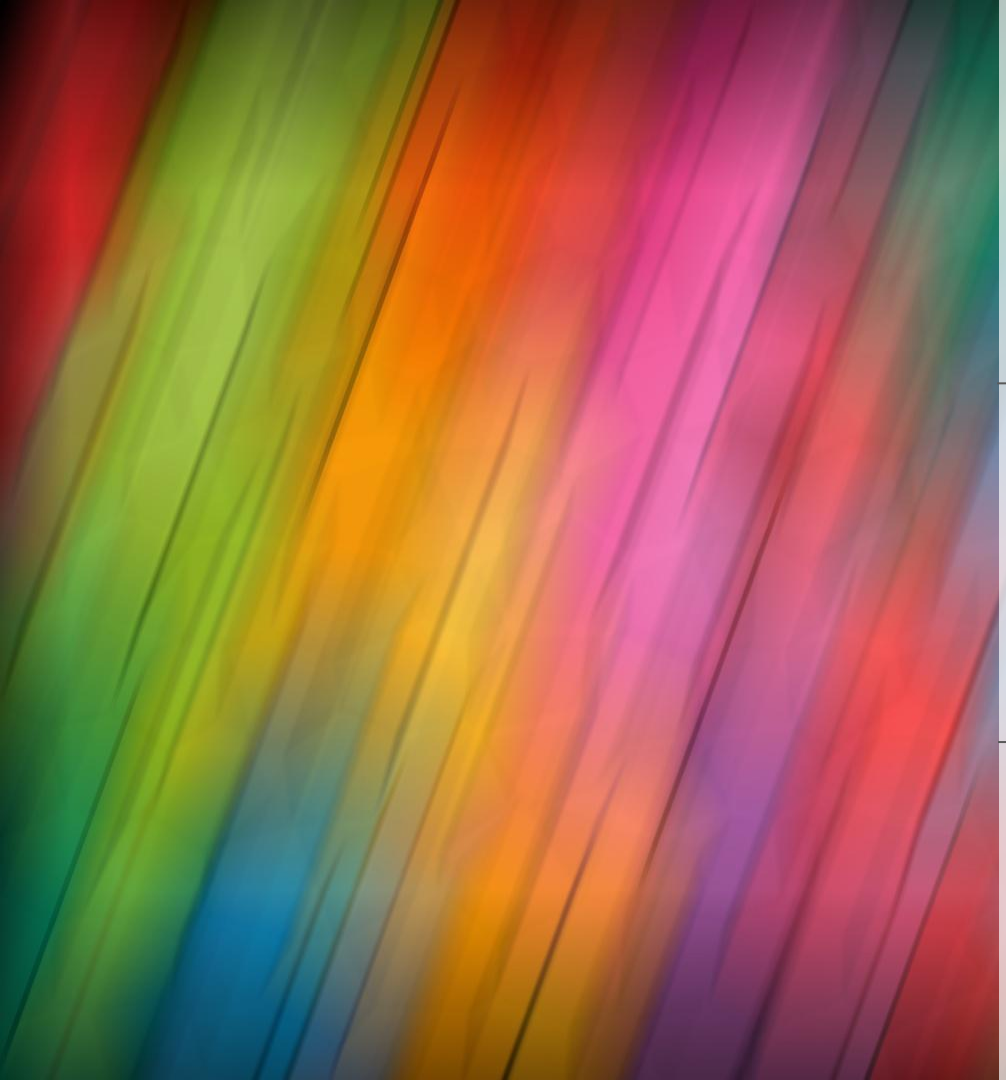
- Build system for the whole stack
- Configurable via modular “layers”
- Combines all parts into a final bootable image



Yocto - “LEGO for Embedded Systems”



Live Demo



AmbiLight

SoC2018

B. Gigerl • F. Kargl • P. Nasahl

A. Trinker • M. Lipp • L. Macan

S. Steinegger