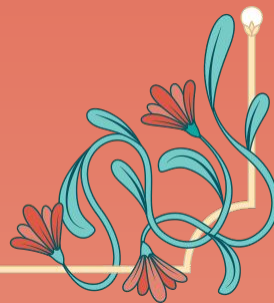


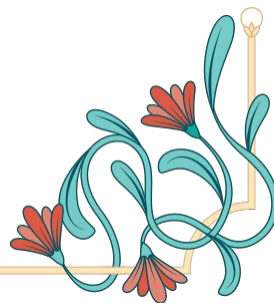
Fully Describe Your Regulators and Supplies

Chen-Yu Tsai, Google LLC



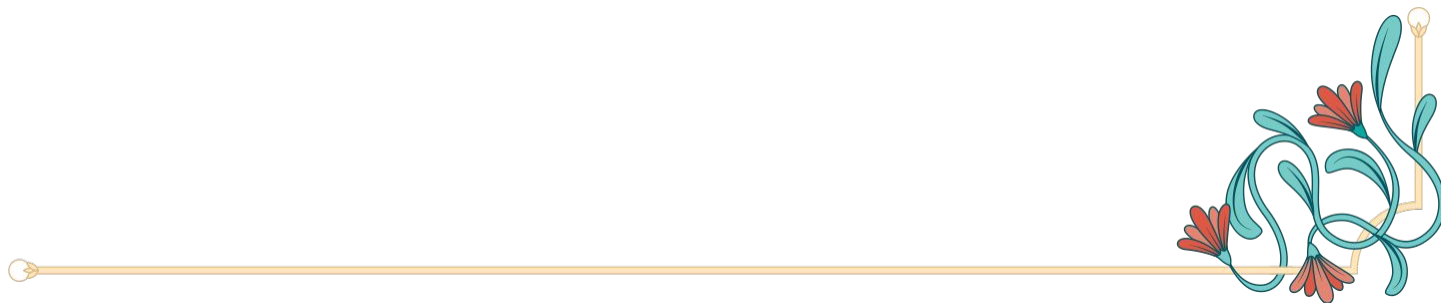
Chen-Yu Tsai

- Works on ARM-based Chromebooks at Google
- Current maintainer of wireless-regdb project
- Maintainer of Allwinner SoC within the kernel on the side



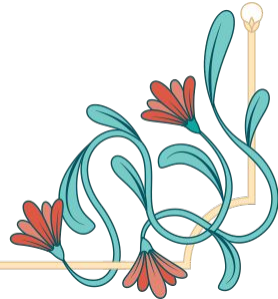
Disclaimer

- All opinions are my own, not my employer's, nor the maintainer's
- Focus more on device tree platforms / representations

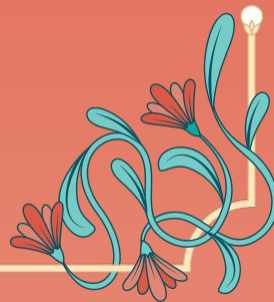


Outline

- Fully Describe Regulators
- Regulator / Power Tree
- Regulators
- Regulators in Linux

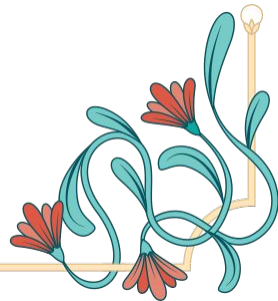


Fully Describe Regulators



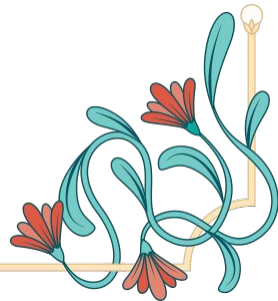
Fully Describe Regulators - Why?

- Correctness
 - Tree matches the hardware design
 - Names match the rail names in the schematic
 - Constraints match what hardware needs
- Dependencies
 - No dangling regulators
 - No (unused) regulators accidentally turned off



Fully Describe Regulators - How?

- All regulators should have inputs given
- All regulators should have operational voltage limits
 - Describe limits of devices fed by the regulator
 - A device accepting 5V should not be fed 12V
- Except load switches
 - No regulation - no limit
- Include any other characteristics or limitations of the board design
 - Enable ramp up time
 - Voltage change ramp rate
 - etc.



Regulator / Power Tree



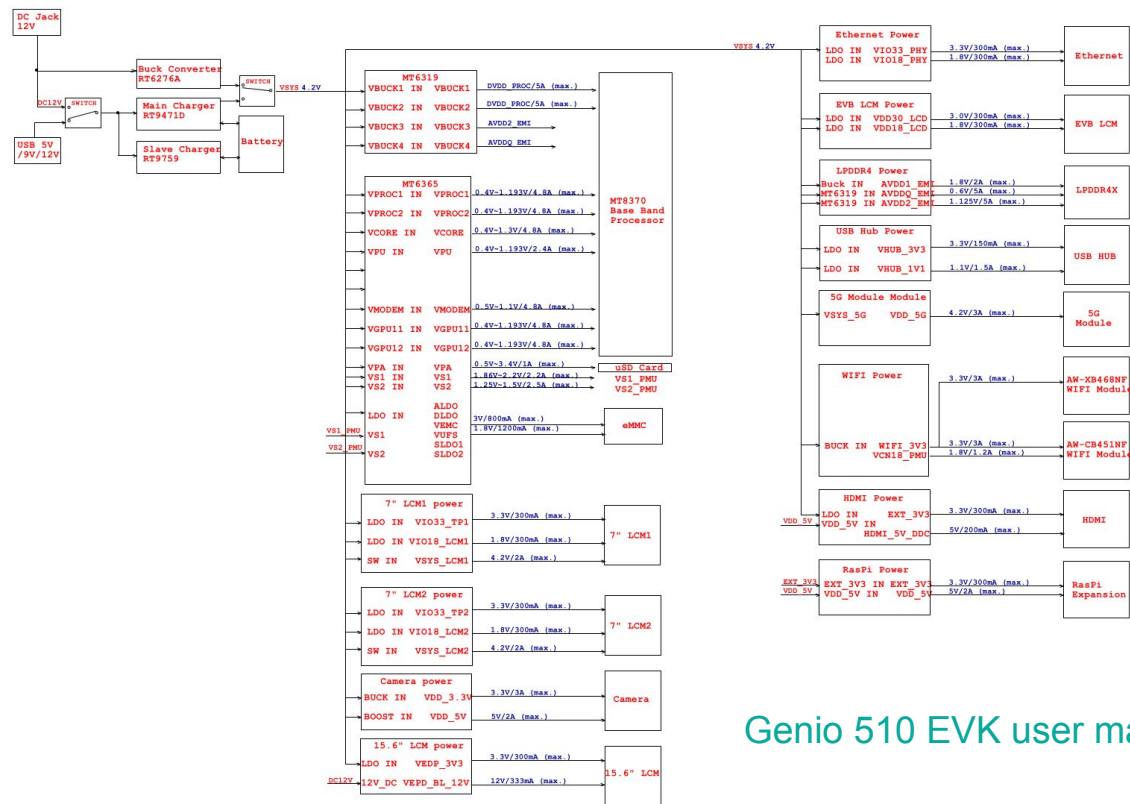
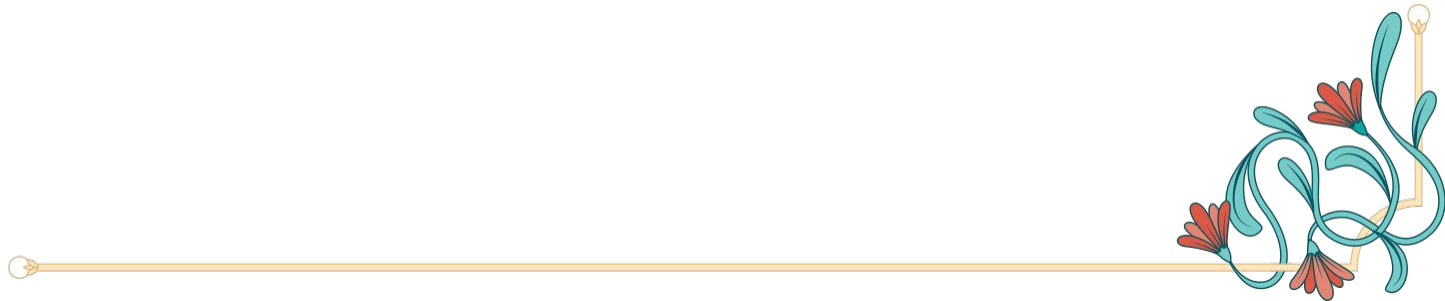


Figure 5-1. Power Distribution

Genio 510 EVK user manual

Regulator Tree in Hardware

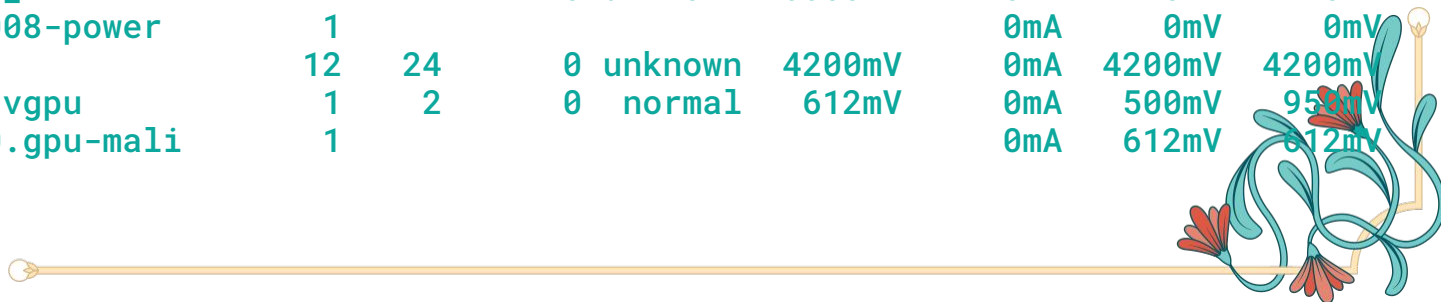
- System typically has one or two main power sources blended into one root power rail
- Regulators (either discrete or within a PMIC) regulate this power rail to provide a few main power rails at different voltages
- Load switches control power from these power rails to individual components or peripherals



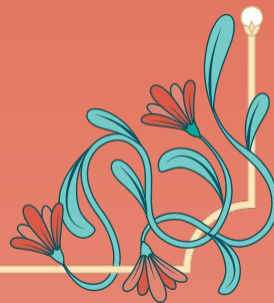
Regulator Tree (snippet) in Linux

```
# cat /sys/kernel/debug/regulator/regulator_summary
```

regulator	use	open	bypass	opmode	voltage	current	min	max
regulator-dummy	7	6	0	unknown	0mV	0mA	0mV	0mV
1-0010-vccio	1					0mA	0mV	0mV
11280000.usb-vusb33	1					0mA	0mV	0mV
spi2.0-vcc	1					0mA	0mV	0mV
ppvar_sys	6	5	0	unknown	0mV	0mA	0mV	0mV
backlight-lcd0-power	1					0mA	0mV	0mV
pp3300_ldo_z5	1	0	0	unknown	3300mV	0mA	3300mV	3300mV
pp3300_z2	4	3	0	unknown	3300mV	0mA	3300mV	3300mV
5-001a-MICVDD	1					0mA	0mV	0mV
pp3300_disp_x	1	1	0	unknown	3300mV	0mA	0mV	0mV
aux-0-0008-power	1					0mA	0mV	0mV
pp4200_z2	12	24	0	unknown	4200mV	0mA	4200mV	4200mV
ppvar_dvdd_vgpu	1	2	0	normal	612mV	0mA	500mV	950mV
13040000.gpu-mali	1					0mA	612mV	612mV

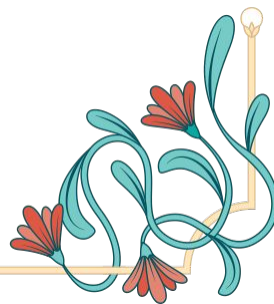


Regulators



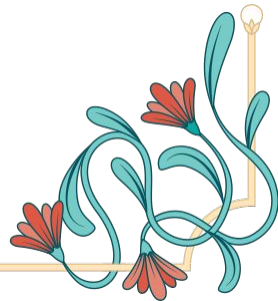
Regulators

- Regulators "regulate" voltage or "limit" current
- Load switches only control on/off
- All regulators have an input supply
 - Some regulators have separate control plane supplies
 - Not directly modelable in the regulator framework
- Controllable regulators have some control interface, typically I2C
 - GPIO controlled designs also exist



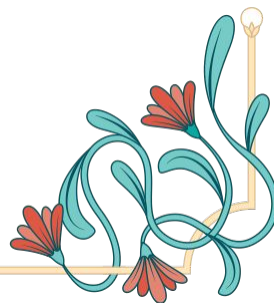
Regulator Constraints - Requirements of Consumers

- Each device has different tolerances for input voltage
- Some also require voltage rise times to be in a certain range
- Check your component datasheets
- Decoupling capacitors and inductors also affect voltage rise time and slew rates
- Constraints are not the characteristics of the regulator itself
- Regulator voltage ranges are already known based on the component model



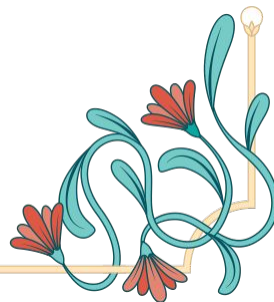
Regulator Types in Linux

- Fixed regulators
 - Provide power at fixed voltage
 - Optional GPIO for on/off
 - Also used to model load switch
- GPIO controlled regulators
 - Extra GPIO(s) typically change feedback resistor network of a variable regulator
- Programmable regulators
 - PMICs and other regulator ICs with programmable interface
 - I2C, SPI, SPMI, etc.



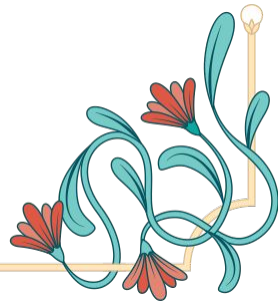
Regulators in Linux

- `struct regulator_desc` describes regulator characteristics
 - Fixed regulator has one voltage, not a range
 - Load switch has no defined voltage
 - Voltage inherited from supply
- Fixed regulators and load switches only implement `.enable()` / `.disable()` / `.is_enabled()`
- Configurable regulators also implement `.get_voltage()` / `.set_voltage()`
 - Or `.get_current_limit()` / `.set_current_limit()`



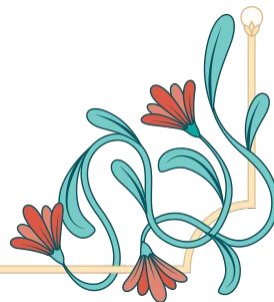
Discrete Fixed Regulator in Device Tree

```
/* system wide LD0 3.3V power rail */
pp3300_z5: regulator-pp3300-ldo-z5 {
    compatible = "regulator-fixed";
    regulator-name = "pp3300_ldo_z5";
    regulator-always-on;
    regulator-boot-on;
    regulator-min-microvolt = <3300000>;
    regulator-max-microvolt = <3300000>;
    vin-supply = <&ppvar_sys>;
};
```



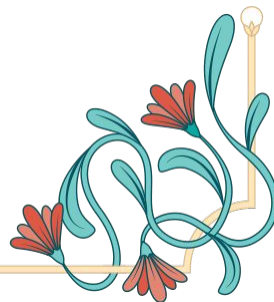
PMIC Regulators in Device Tree

```
pmic {  
    vsys-gpu-supply = <&ppvar_sys>;  
  
    ppvar_gpu: regulator-ppvar-gpu {  
        regulator-name = "ppvar-gpu";  
        regulator-min-microvolt = <500000>;  
        regulator-max-microvolt = <950000>;  
        regulator-ramp-delay = <6250>;  
        regulator-enable-ramp-delay = <200>;  
    };  
};
```



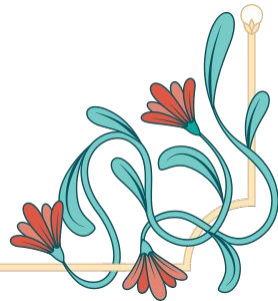
PMIC Regulators in Device Tree

```
pmic {  
    compatible = "...";  
    pvdd1-supply = <&ppvar_sys>;  
    pvdd2-supply = <&ppvar_sys>;  
  
    regulators {  
        vbuck1 { ... };  
        vbuck2 { ... };  
    };  
};
```

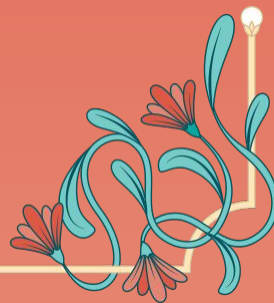


PMIC Regulators with Shared Supply in Device Tree

```
pmic {  
    compatible = "...";  
    vsys-vs1-supply = <&ppvar_sys>;  
    vs1-ldo1-supply = <&vs1>; # for vs1-ldo1-*  
  
    regulators {  
        vs1 { ... };  
        vs1-ldo1-foo { ... };  
        vs1-ldo1-bar { ... };  
    };  
};
```

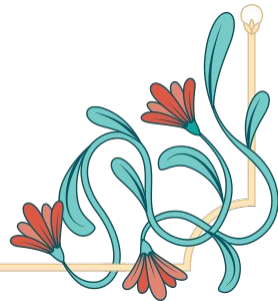


Summary



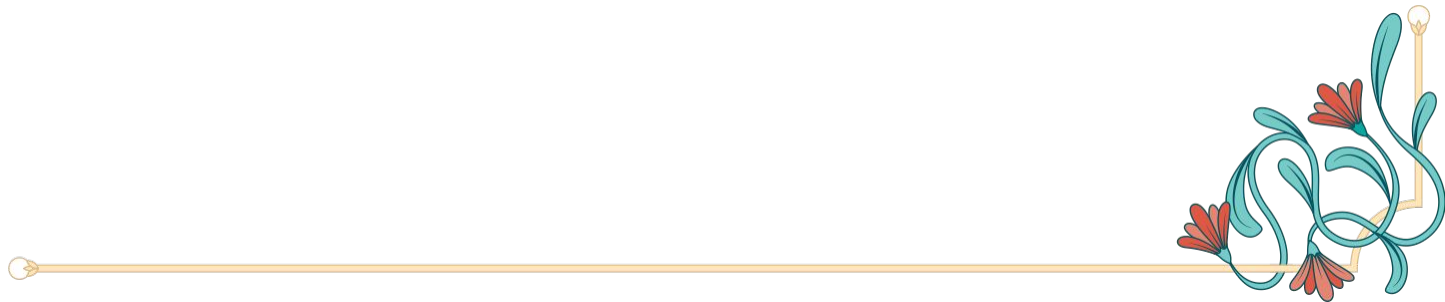
What You Should Do

- All regulators should have input supplies
 - Except the root power rail
- All regulators should have voltage constraints
 - Except load switches
- Voltage constraints should be the intersection of the regulator's consumer's operating ranges

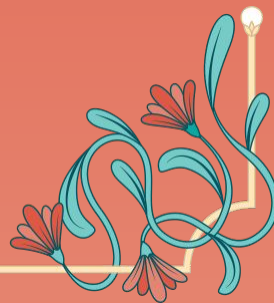


What to Avoid - pmic.dtsi file with defaults

- Default names don't make sense and get overridden anyway
- Voltage defaults are almost always specified with the regulators' full range
 - Not valid for consumers and will get overridden
 - Full range, a characteristic of the regulator, is already known by the driver
 - Not a constraint
- Why have a nearly empty .dtsi file?



Questions?



THE LINUX FOUNDATION



**EMBEDDED
LINUX
CONFERENCE**

EUROPE

