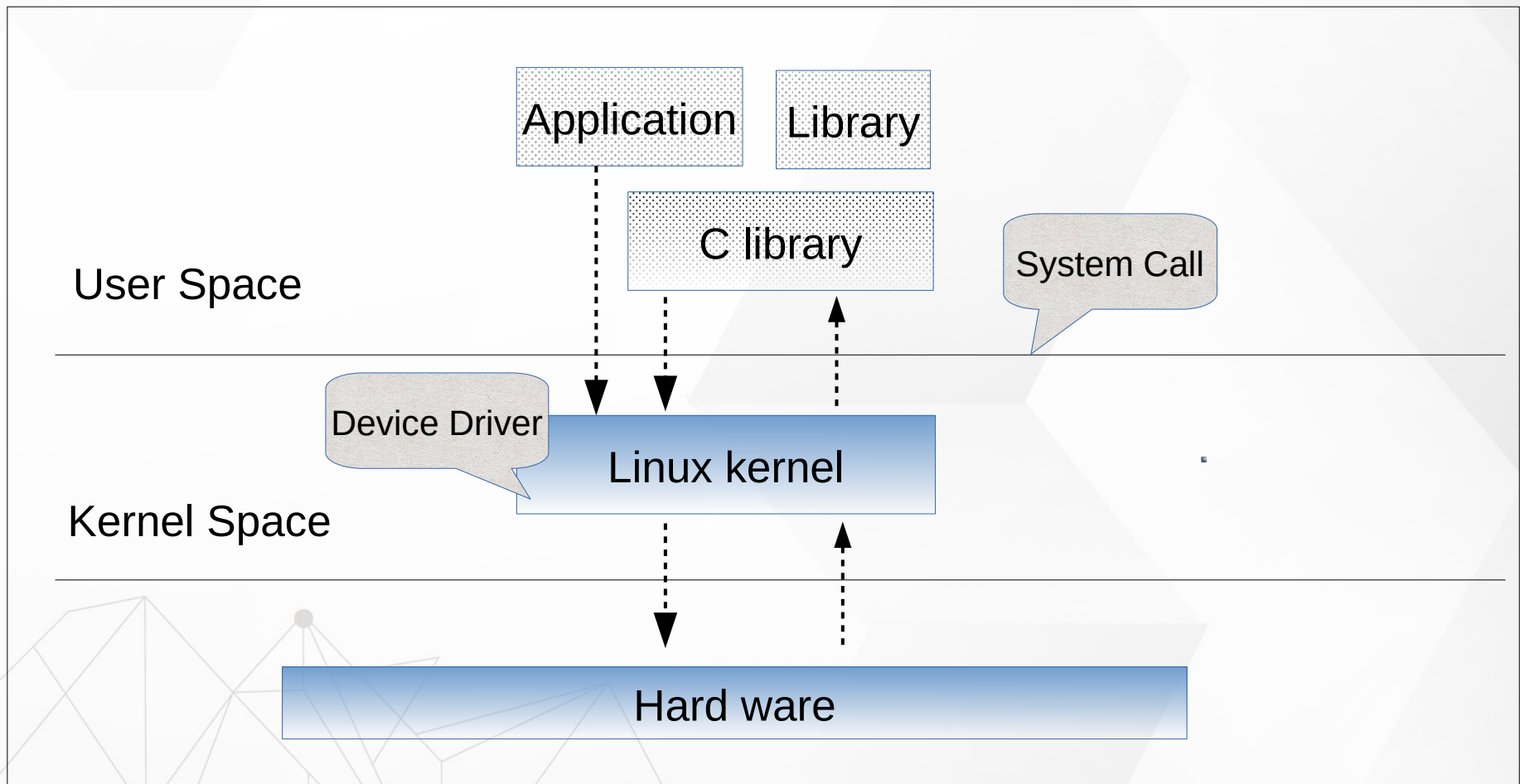
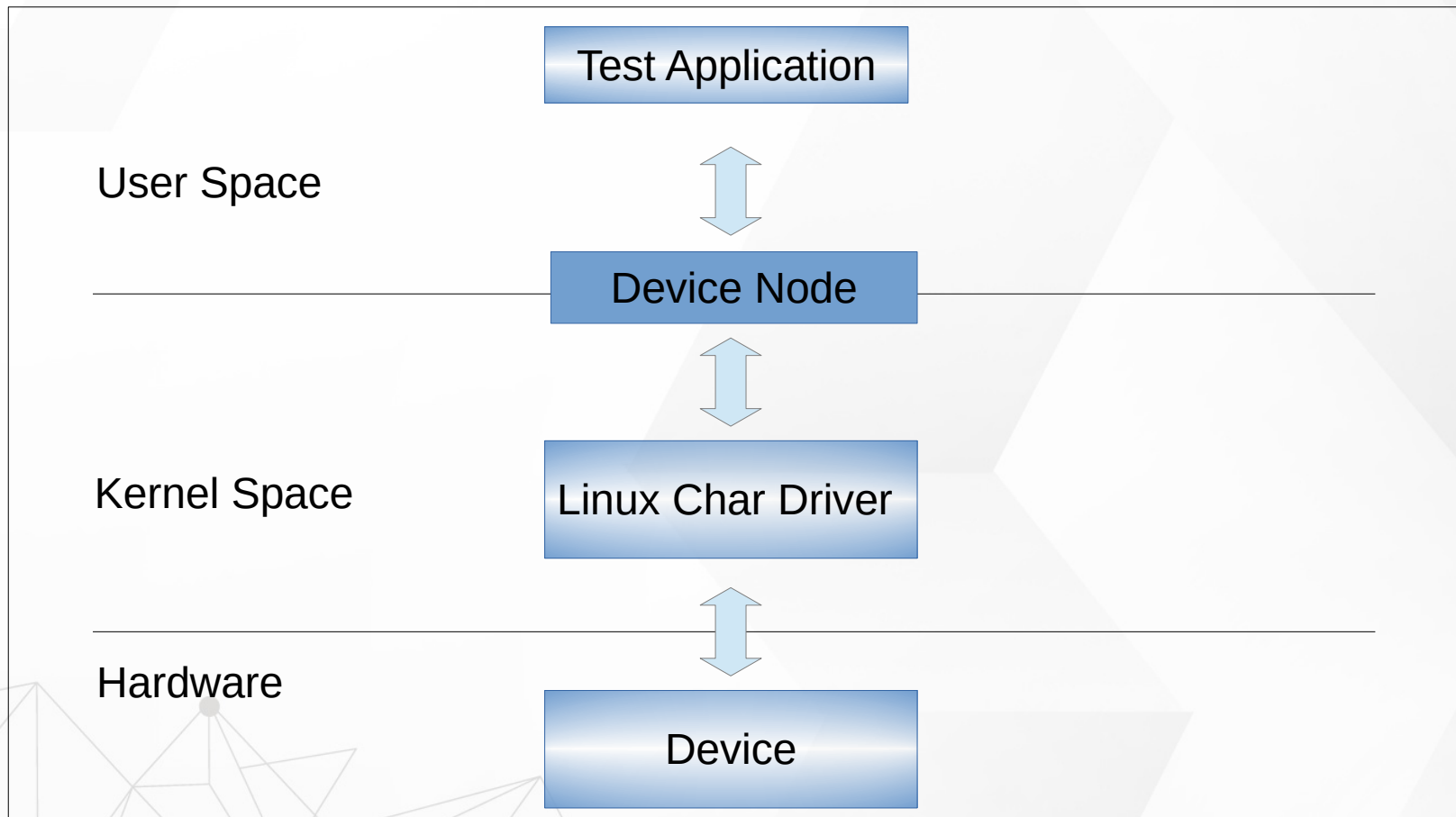


CH9 Control Hardware Driver

Application and Hardware



User Space and Kernel Space

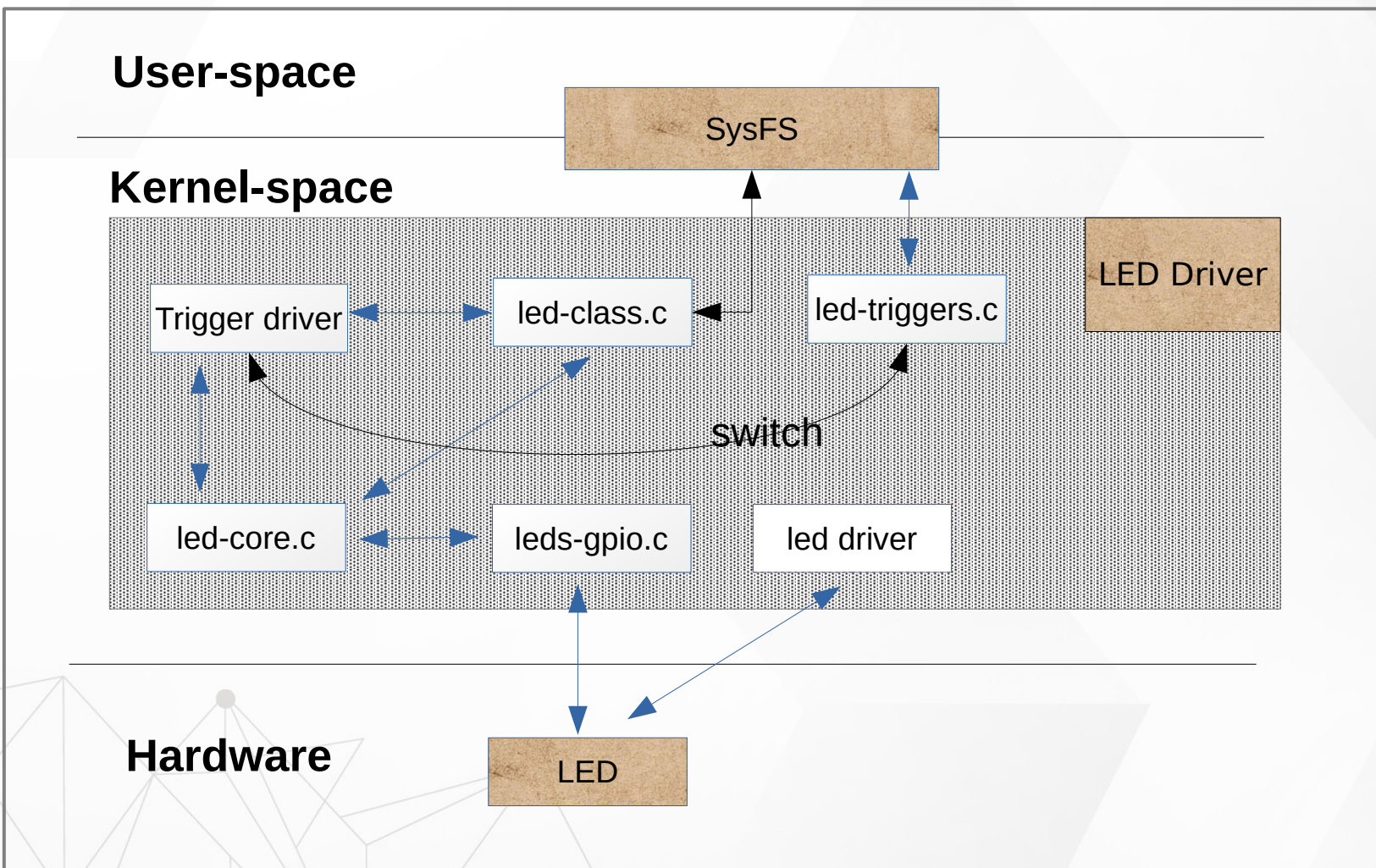




SysFS

- Allows kernel code to export information to user processes
- SysFS is an in-memory file system
- It provides two components
 - A kernel programming interface for exporting these items via sysfs
 - User interface to view and manipulate these items that maps back to the kernel objects which they represent

LED and SysFS



Sys File System

```
# tree -L 1 /sys/
```

```
/sys/  
├── block  
├── bus  
├── class  
├── dev  
├── devices  
├── firmware  
├── fs  
├── hypervisor  
├── kernel  
├── module  
└── power
```

```
# tree -L 1 /sys/class/i2c-dev/i2c-0/
```

```
/sys/class/i2c-dev/i2c-0/  
├── dev  
├── device -> ../../../../i2c-0  
├── name  
├── power  
├── subsystem -> ../../../../../../class/i2c-dev  
└── uevent
```

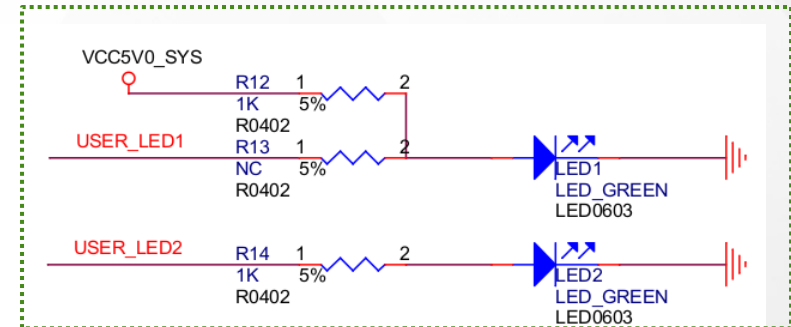
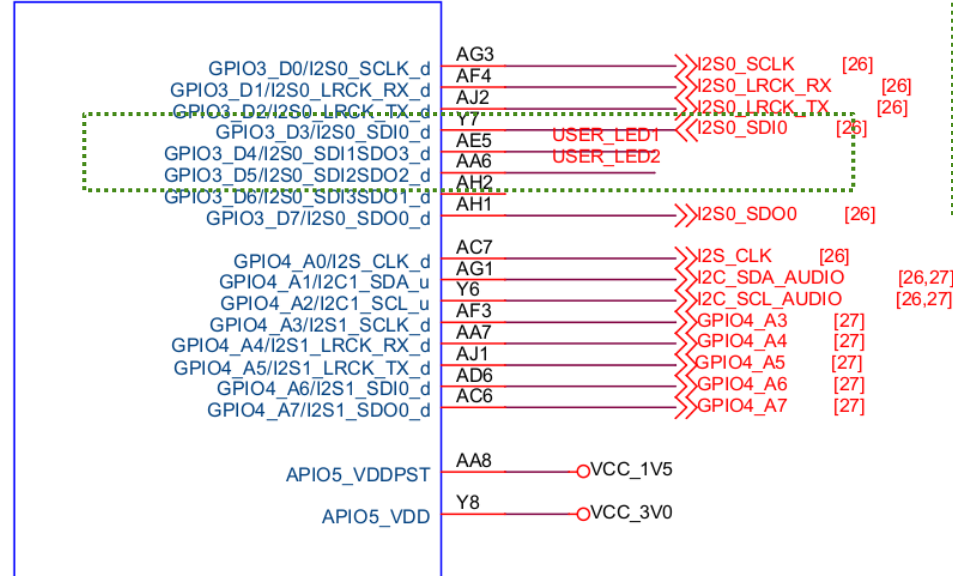
```
# tree -L 1 /sys/class/i2c-dev
```

```
/sys/class/i2c-dev/  
├── i2c-0 -> ../../devices/pci0000:00/0000:00:02.0/i2c-0/i2c-dev/i2c-0  
├── i2c-1 -> ../../devices/pci0000:00/0000:00:02.0/i2c-1/i2c-dev/i2c-1  
├── i2c-2 -> ../../devices/pci0000:00/0000:00:02.0/i2c-2/i2c-dev/i2c-2  
├── i2c-3 -> ../../devices/pci0000:00/0000:00:02.0/i2c-3/i2c-dev/i2c-3  
├── i2c-4 -> ../../devices/pci0000:00/0000:00:02.0/i2c-4/i2c-dev/i2c-4  
├── i2c-5 -> ../../devices/pci0000:00/0000:00:02.0/i2c-5/i2c-dev/i2c-5  
├── i2c-6 -> ../../devices/pci0000:00/0000:00:02.0/drm/card0/card0-DP-1/i2c-6/i2c-dev/i2c-6  
├── i2c-7 -> ../../devices/pci0000:00/0000:00:02.0/drm/card0/card0-DP-2/i2c-7/i2c-dev/i2c-7  
└── i2c-8 -> ../../devices/pci0000:00/0000:00:02.0/drm/card0/card0-DP-3/i2c-8/i2c-dev/i2c-8
```

LED Drivers

LED Schematic

U1J RK3399 Note:RK3399 part J is 1.8V/3.0V mode



GPIO3_D4/I2S0_SDI1SDO3_d → LED1

GPIO3_D5/I2S0_SDI2SDO2_d → LED2



LED Subsystem

» Control LED convenient with SysFS

» For example

- `echo 1 > /sys/class/leds/user-led2/shot`

» Switch different LED trigger type in SysFS

» For example

- `echo "gpio" > /sys/class/leds/user-led2/trigger`
- `echo "1" > /sys/class/leds/user-led2/brightness`
- `echo "0" > /sys/class/leds/user-led2/brightness`

LED SysFS

```
root@rockpi4b:/sys/class/leds/user-led2# ls -l
```

```
brightness
device -> ../../../../gpio-leds
max_brightness
power
subsystem -> ../../../../../../class/leds
trigger
uevent
```

Check trigger type

```
root@rockpi4b:/sys/class/leds/user-led2# cat trigger
```

```
none rc-feedback kbd-scrolllock kbd-numlock kbd-capslock
kbd-kanalock kbd-shiftlock kbd-altgrlock kbd-ctrllock kbd-altlock
kbd-shiftllock kbd-shiftrlock kbd-ctrllllock kbd-ctrlrlock m
timer oneshot heartbeat backlight [gpio] cpu0 cpu1 cpu2 cpu
```

Switch trigger type

```
root@rockpi4b:/sys/class/leds/user-led2# echo heartbeat > trigger
```

```
root@rockpi4b:/sys/class/leds/user-led2# cat trigger
none rc-feedback kbd-scrolllock kbd-numlock kbd-capslock
kbd-kanalock kbd-shiftlock kbd-altgrlock kbd-ctrllock kbd-altlock
kbd-shiftllock kbd-shiftrlock kbd-ctrllllock kbd-ctrlrlock mmc0 mmc1
timer oneshot [heartbeat] backlight gpio| cpu0 cpu1 cpu2 cpu3 cpu4 cpu5
```

GPIO Control



Driver LED in User Space

➤ Paths in Sysfs

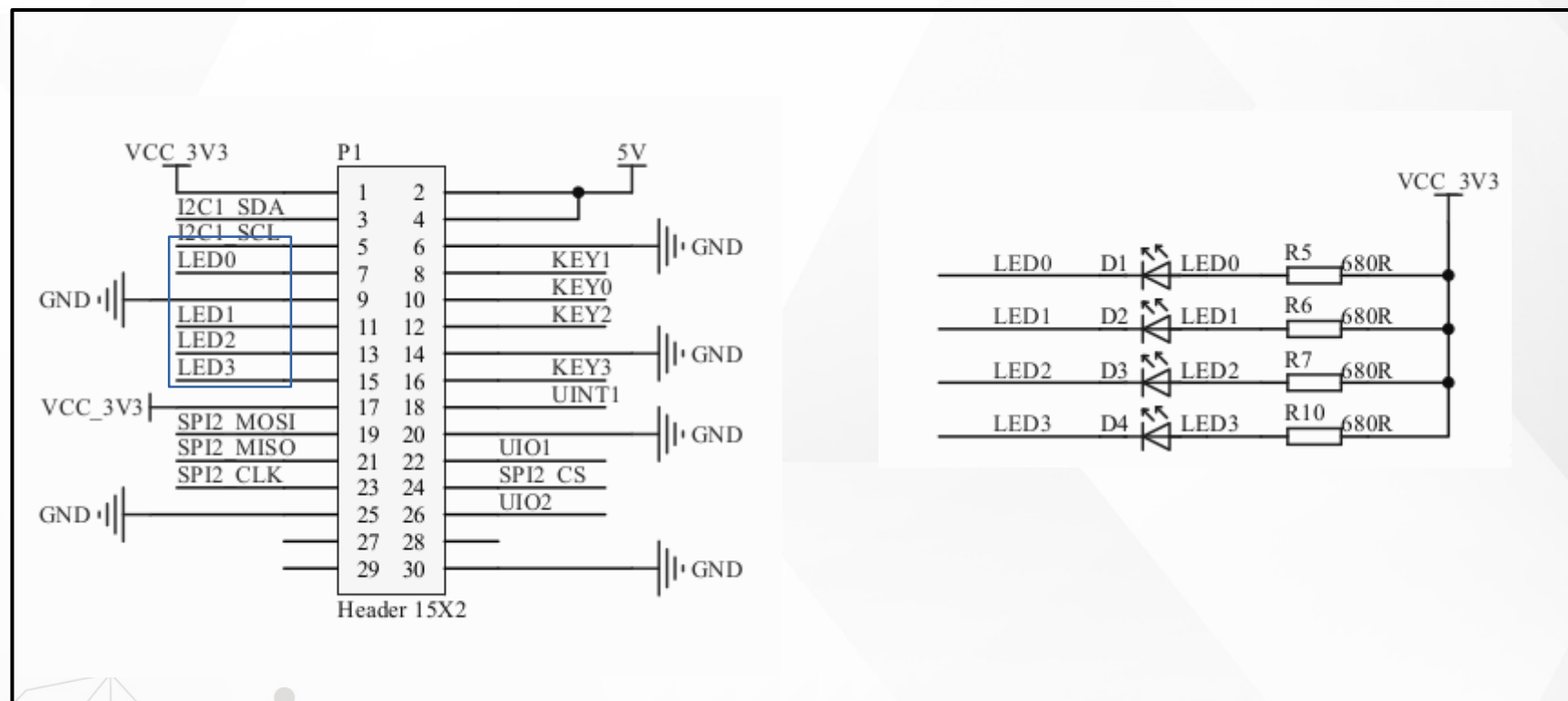
➤ /sys/class/gpio:

- Control interfaces used to get userspace control over GPIOs;
- GPIOs themselves
- GPIO controllers("gpio_chip" instances)

➤ /sys/class/gpio/

- "export" : ask the kernel to export GPIO to userspace by writing
 - "echo 19 > export"
 - create a "gpio19" node in /sys/class/gpio
- "unexport" : Reverses the effect of exporting to userspace
 - "echo 19 > unexport"
 - remove "gpio19" node from /sys/class/gpio

Cadtc Ext Board LED



RockPi4B HEAD

Rock Pi 4 A/B/C general purpose input-output (GPIO) connector

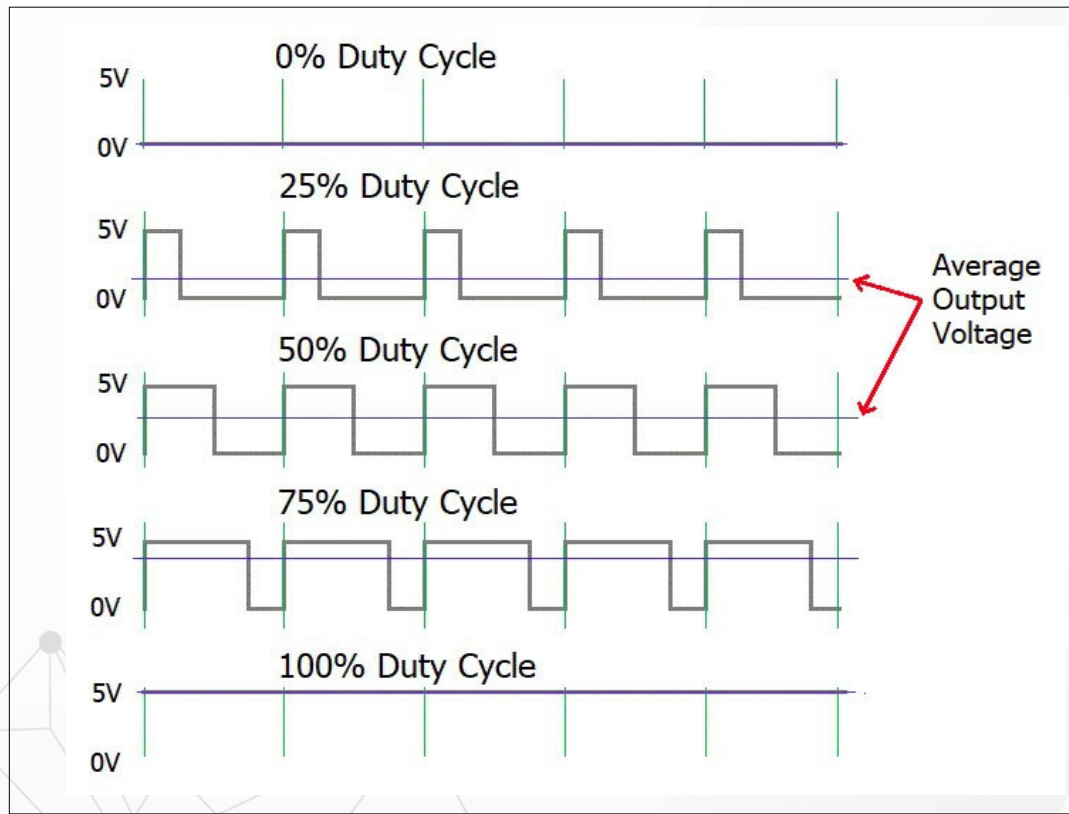
ROCK Pi 4 has a 40-pin expansion header. Each pin is distinguished by color.

GPIO number	Function2	Function1	GPIO	Pin#	Pin#	GPIO	Function1	Function2	GPIO number
		+3.3V		1	2		+5.0V		
71		I2C7_SDA	GPIO2_A7	3	4		+5.0V		
72		I2C7_SCL	GPIO2_B0	5	6		GND		
75		SPI2_CLK	GPIO2_B3	7	8	GPIO4_C4	UART2_TXD		148
		GND		9	10	GPIO4_C3	UART2_RXD		147
146		PWM0	GPIO4_C2	11	12	GPIO4_A3	I2S1_SCLK		131
150		PWM1	GPIO4_C6	13	14		GND		
149		SPDIF_TX	GPIO4_C5	15	16	GPIO4_D2			154
		+3.3V		17	18	GPIO4_D4			156
40	UART4_TXD	SPI1_TXD	GPIO1_B0	19	20		GND		
39	UART4_RXD	SPI1_RXD	GPIO1_A7	21	22	GPIO4_D5			157
41		SPI1_CLK	GPIO1_B1	23	24	GPIO1_B2	SPI1_CSn		42
		GND		25	26		ADC_IN0		
64		I2C2_SDA	GPIO2_A0	27	28	GPIO2_A1	I2C2_CLK		65
74	I2C6_SCL	SPI2_TXD	GPIO2_B2	29	30		GND		
73	I2C6_SDA	SPI2_RXD	GPIO2_B1	31	32	GPIO3_C0	SPDIF_TX	UART3_CTSn	112
76		SPI2_CSn	GPIO2_B4	33	34		GND		
133		I2S1_LRCK_TX	GPIO4_A5	35	36	GPIO4_A4	I2S1_LRCK_RX		132
158			GPIO4_D6	37	38	GPIO4_A6	I2S1_SDI		134
		GND		39	40	GPIO4_A7	I2S1_SDO		135

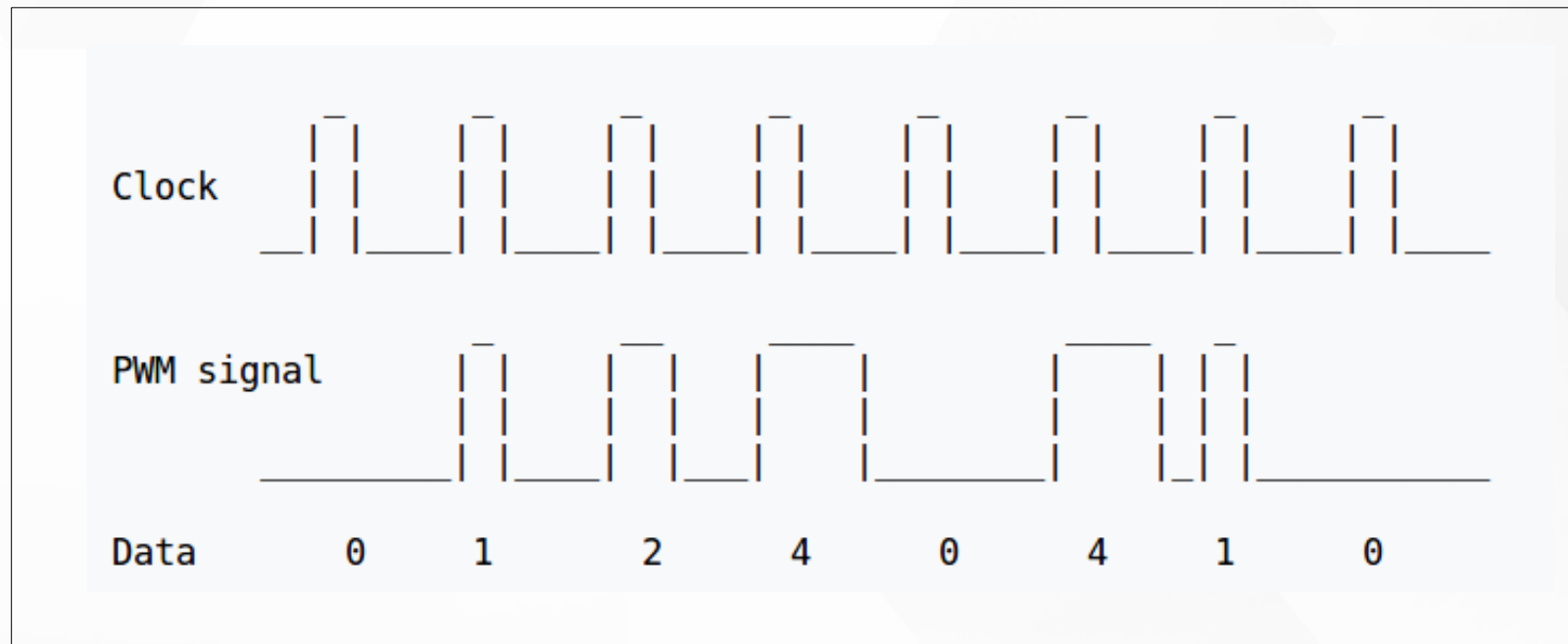
PWM Sub System

PWM

PWM : Pulse Width Modulation



PWM



https://en.wikipedia.org/wiki/Pulse-width_modulation



PWM Parameter in Linux



Period

- The total period of the PWM signal
- Value is in nanoseconds
- sum of the active and inactive time of the PWM



duty_cycle

- The active time of the PWM signal
- Value is in nanoseconds
- must be less than the period.



PWM Parameter in Linux

Polarity

→ The polarity of the PWM signal

Enable

→ Enable PWM Signal



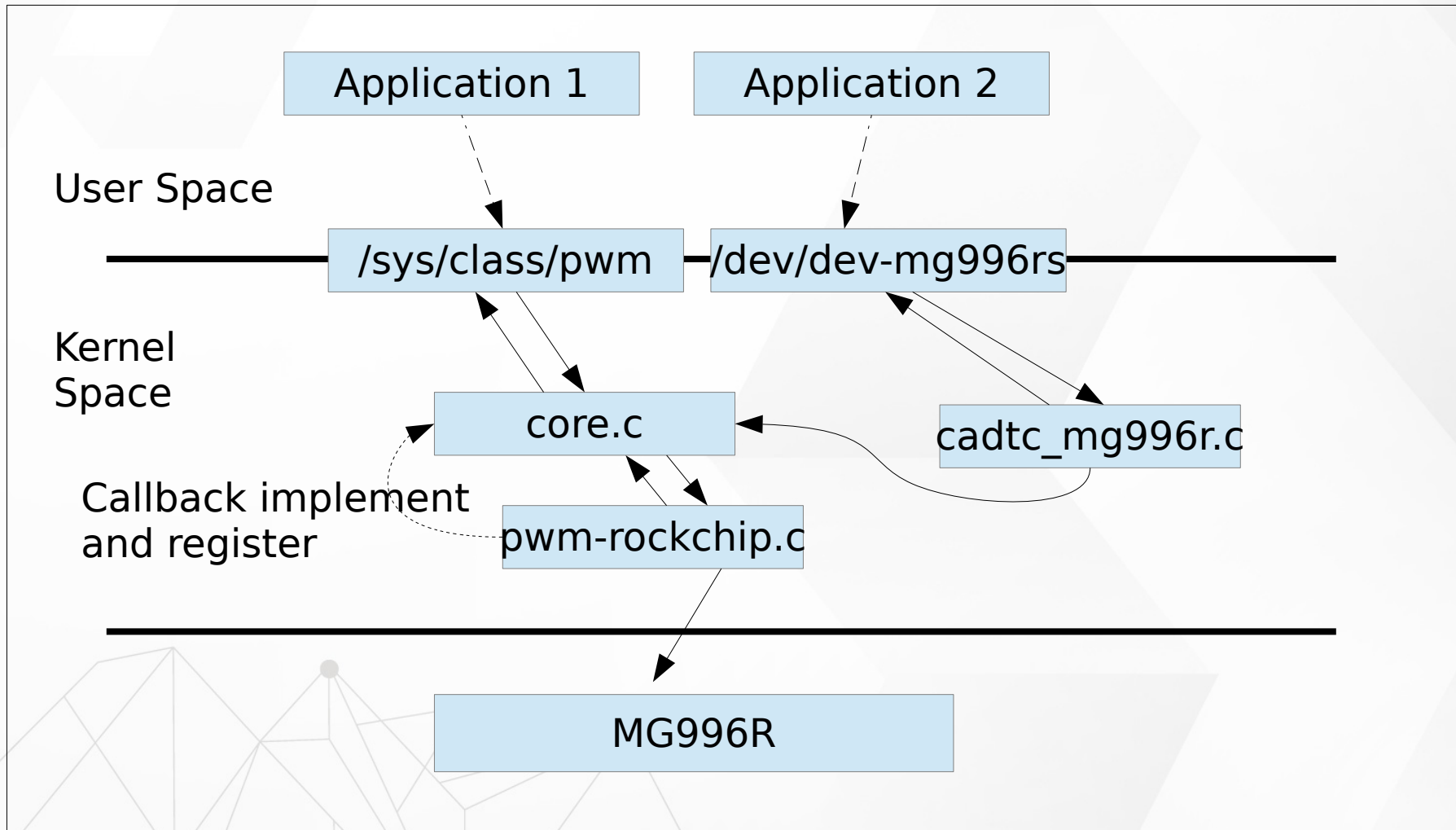
PWM Driver

➤ \$(KERNEL_SRC)/Documentation/pwm.txt

➤ Platform Driver

- drivers/pwm/
- drivers/pwm/core.c
- drivers/pwm/pwm-rockchip.c

PWM Subsystem



RockPi4B HEAD

Rock Pi 4 A/B/C general purpose input-output (GPIO) connector

ROCK Pi 4 has a 40-pin expansion header. Each pin is distinguished by color.

GPIO number	Function2	Function1	GPIO	Pin#	Pin#	GPIO	Function1	Function2	GPIO number
		+3.3V		1	2		+5.0V		
71		I2C7_SDA	GPIO2_A7	3	4		+5.0V		
72		I2C7_SCL	GPIO2_B0	5	6		GND		
75		SPI2_CLK	GPIO2_B3	7	8	GPIO4_C4	UART2_TXD		148
		GND		9	10	GPIO4_C3	UART2_RXD		147
146		PWM0	GPIO4_C2	11	12	GPIO4_A3	I2S1_SCLK		131
150		PWM1	GPIO4_C6	13	14		GND		
149		SPDIF_TX	GPIO4_C5	15	16	GPIO4_D2			154
		+3.3V		17	18	GPIO4_D4			156
40	UART4_TXD	SPI1_TXD	GPIO1_B0	19	20		GND		
39	UART4_RXD	SPI1_RXD	GPIO1_A7	21	22	GPIO4_D5			157
41		SPI1_CLK	GPIO1_B1	23	24	GPIO1_B2	SPI1_CSn		42
		GND		25	26		ADC_IN0		
64		I2C2_SDA	GPIO2_A0	27	28	GPIO2_A1	I2C2_CLK		65
74	I2C6_SCL	SPI2_TXD	GPIO2_B2	29	30		GND		
73	I2C6_SDA	SPI2_RXD	GPIO2_B1	31	32	GPIO3_C0	SPDIF_TX	UART3_CTSn	112
76		SPI2_CSn	GPIO2_B4	33	34		GND		
133		I2S1_LRCK_TX	GPIO4_A5	35	36	GPIO4_A4	I2S1_LRCK_RX		132
158			GPIO4_D6	37	38	GPIO4_A6	I2S1_SDI		134
		GND		39	40	GPIO4_A7	I2S1_SDO		135

PWM SYSFS

```
/sys/class/pwm/pwmchip0
```

```
device    export    npwm    power    subsystem uevent    unexport
```

```
echo 0 > export
```

```
capture    enable    polarity    uevent    duty_cycle    period    power
```

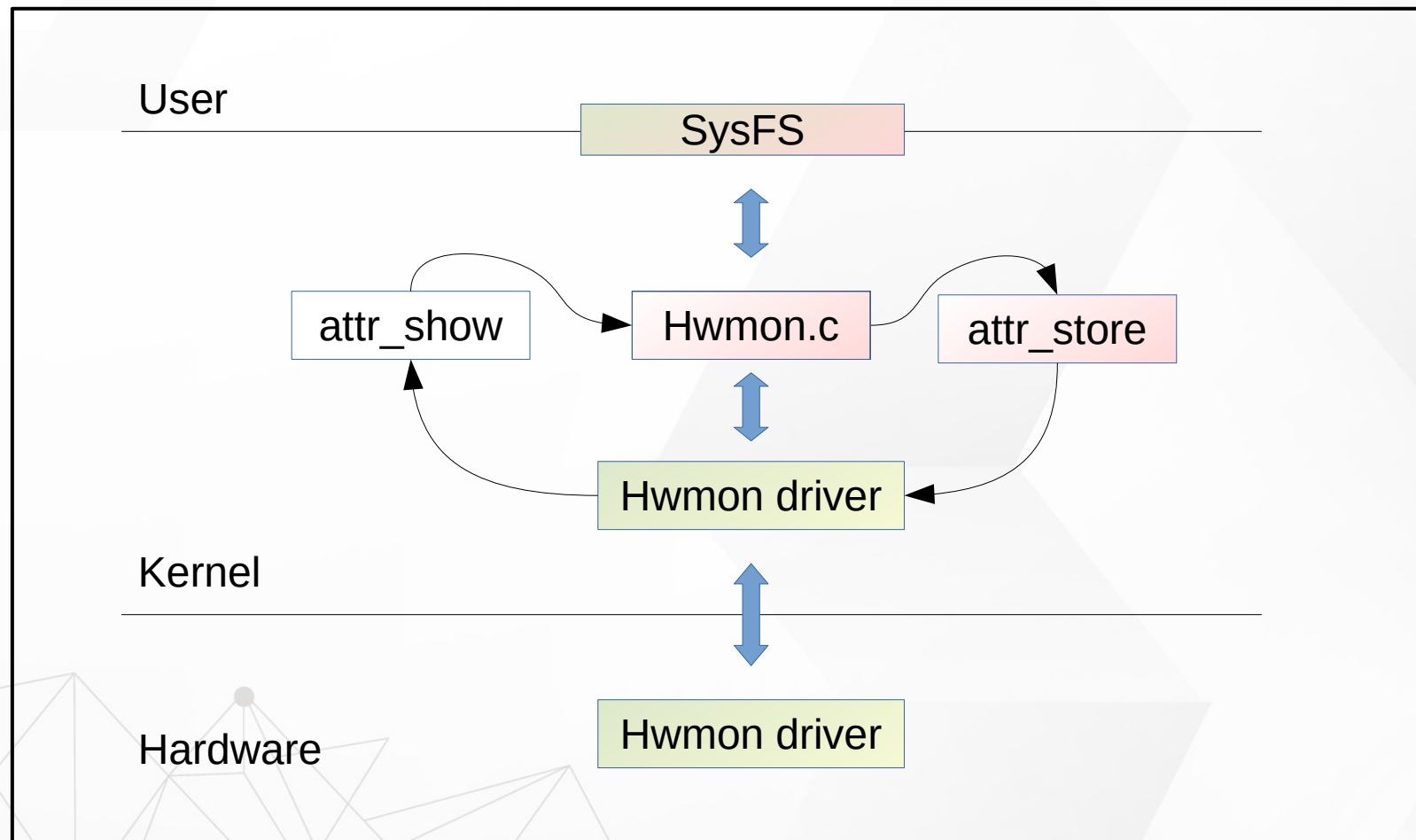
```
echo "20000000" > period    //20ms, 50 Hz
```

```
echo "2000000" > duty_cycle    //2ms
```

```
echo 1 > enable    //Enable
```

Hwmon Subsystem

Hwmon Subsystem



SHT21

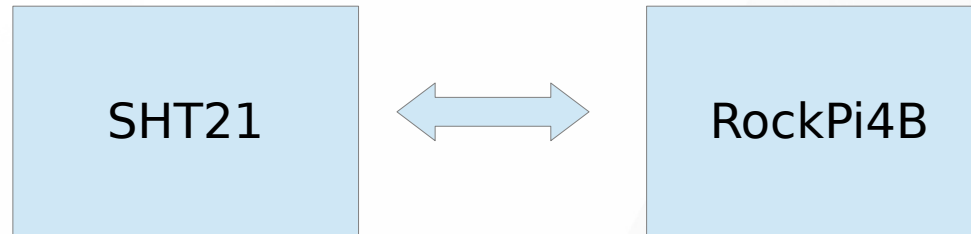
 Simple interface Bus interface

- I2C, GPIO, SPI

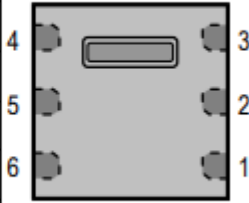
 Sensors

- Temperature
- Voltage
- Humidity
- Fan speed
- PWM control

SHT21



Pin	Name	Comment
1	SDA	Serial Data, bidirectional
2	VSS	Ground
5	VDD	Supply Voltage
6	SCL	Serial Clock, bidirectional
3,4	NC	Not Connected



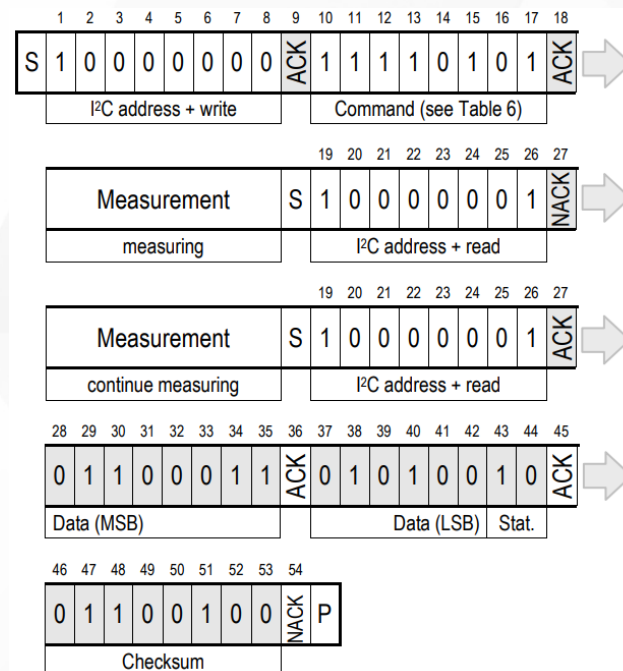
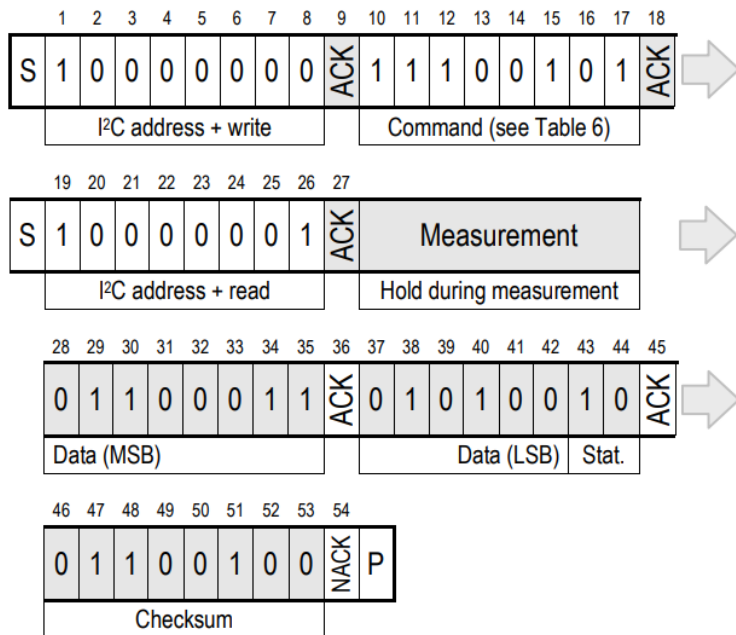
A small diagram of the SHT21 sensor package with pins numbered 1 through 6. Pin 1 is on the right, pin 2 is below it, pin 3 is on the top right, pin 4 is on the top left, pin 5 is on the bottom left, and pin 6 is on the bottom right.

Command	Comment	Code
Trigger T measurement	hold master	1110'0011
Trigger RH measurement	hold master	1110'0101
Trigger T measurement	no hold master	1111'0011
Trigger RH measurement	no hold master	1111'0101
Write user register		1110'0110
Read user register		1110'0111
Soft reset		1111'1110

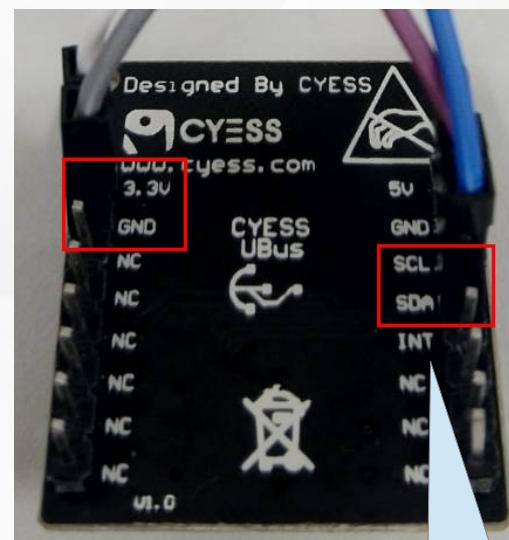
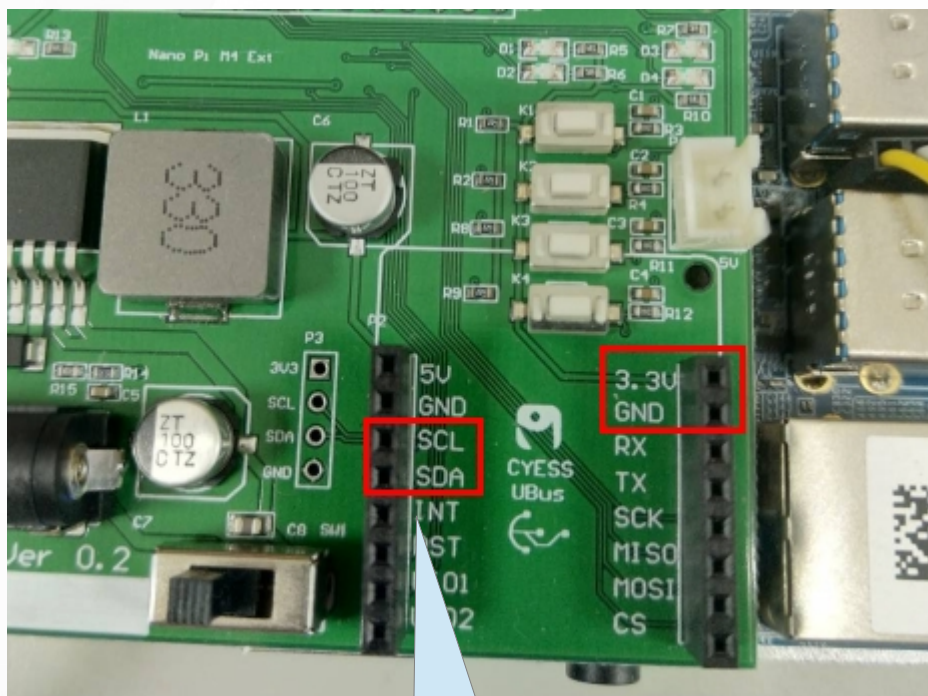
SHT21

Hold master communication sequence

No Hold master communication sequence



SHT21



SHT21

Hwmon Sysfs

```
# ls /sys/class/hwmon/hwmon0
device          name    subsystem  uevent
humidity1_input power   temp1_input
```

temperature

```
# cat /sys/class/hwmon/hwmon0/temp1_input
32279
```

humidity

```
# cat /sys/class/hwmon/hwmon0/humidity1_input
34512
```