kvmd-armbian GPIO ATX Interface

Christoph, St.Gallen, 24 Jan 2023

ATX Interface Board (Hardware)

You should find the hardware with google: aliexpress pikvm atx board

At the moment it says in Aliexpress: ATX CM4 Raspberry pikvm HDMI CSI KVM IP pi

(I don't put the URL here, the don't long live :))

Understanding GPIO Pin´s

Intro

https://iot4beginners.com/difference-between-bcm-and-board-pin-numbering-in-raspberry-pi/

Quote:
GPIO BOARD– This type of pin numbering refers to the number of the pin in the plug, i.e, the numbers printed on the board, for example, P1. The advantage of this type of numbering is, it will not change even though the version of board changes.

GPIO BCM– The BCM option refers to the pin by "Broadcom SOC Channel. They signify the Broadcom SOC channel designation. The BCM channel changes as the version number changes.

kvmd-armbian
Our GPIO python file is /usr/local/lib/python3.10/dist-packages/kvmd/plugins/atx/gpio.py

It is best to write your values directly in this gpio.py.

BCM numbers are used, the addressing addresses from the *Linux Core* to the chip

And "*/etc/kvmd/override.yaml*" only to be used for activation, and no pin to be defined there.

BCM Calculator Allwinner Chips

If you have an OrangePi Zero (Cpu's H2/H3/H5) you will not have the BCM numbers in the table from the manual.

Then there is an online converter that you can use. If PA03 don't write PA3 NO. You have to write the corrct full Name PA03 into the Calculator! (it works also for H616)

 https://jsfiddle.net/tuav7f6q/1/

Below I show my OrangePi Zero 2 (CPU H616), where the BCM number GPIO is also shown in the user manual.

GPIO Configuration

Example Raspberry 4

(Pin is the Plug on Board)

Function	BCM	pin#	pin#	BCM	Function
		Physical Pins			
Function	**BCM**	**pin#**	**pin#**	**BCM**	**Function**
3.3 Volts		1	2		5 Volts
GPIO/SDA1 (I2C)	2	3	4		5 Volts
GPIO/SCL1 (I2C)	3	5	6		GND
GPIO/GCLK	4	7	8	14	TX UART/GPIO
GND		9	10	15	RX UART/GPIO
GPIO	17	11	12	18	GPIO
GPIO	27	13	14		GND
GPIO	22	15	16	23	GPIO
3.3 Volts		17	18	24	GPIO
MOSI (SPI)	10	19	20		GND
MISO(SPI)	9	21	22	25	GPIO
SCLK(SPI)	11	23	24	8	CEO_N (SPI)
GND		25	26	7	CE1_N (SPI)
RESERVED		27	28		RESERVED
GPIO	5	29	30		GND
GPIO	6	31	32	12	GPIO
GPIO	13	33	34		GND
GPIO	19	35	36	16	GPIO
GPIO	26	37	38	20	GPIO
GND		39	40	21	GPIO

Original in kvmd-arbian from Raspberry Pi4:

nano /usr/local/lib/python3.10/dist-packages/kvmd/plugins/atx/gpio.py

```
 def get_plugin_options(cls) -> dict:
     return {
        "device": Option("/dev/gpiochip0", type=valid_abs_path, unpack_as="device_path"),

        "power_led_pin":     Option(23,   type=valid_gpio_pin),
        "power_led_inverted": Option(False, type=valid_bool),
        "power_led_debounce": Option(0.1,   type=valid_float_f0),

        "hdd_led_pin":     Option(22,   type=valid_gpio_pin),
        "hdd_led_inverted": Option(False, type=valid_bool),
        "hdd_led_debounce": Option(0.1,   type=valid_float_f0),

        "power_switch_pin": Option(24,  type=valid_gpio_pin),
        "reset_switch_pin": Option(27,  type=valid_gpio_pin),
        "click_delay":     Option(0.1, type=valid_float_f01),
        "long_click_delay": Option(5.5, type=valid_float_f01),
    }
```
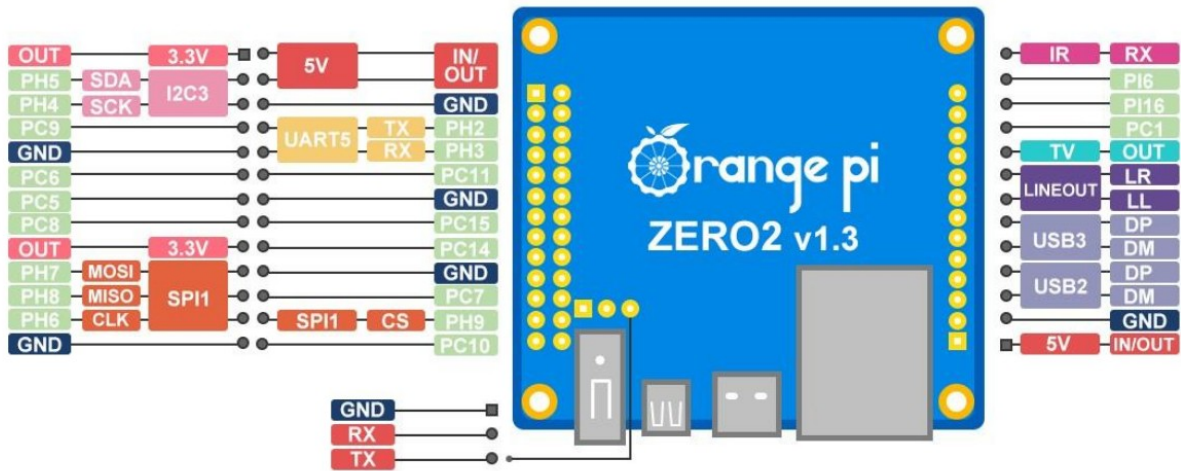
My OrangePi Zero 2 (CPU H616)

Pinout Diagramm

On the Pinout Diagramm, I only use the pins that have no special function. but direct.

So by me: PC6/PC11/PC5/PC8

The connection cables have double connectors, so the correct contacts are next to each other for the connection to the ATX interface board.

I would plan and choose that in advance so that the pins Power_LED,HDD_LED are next to each other. And Power/Rest are next to each other.

Configuration OrangePi Zero2 (H616)

So now i change to my Board and activate it. (Pin is the Plug on Board)

The function of the 26 pin of the Orange Pi Zero2 development board is shown in the table below

GPIO No.	GPIO	Function	Pin	Pin	Function	GPIO	GPIO No.
		3.3V	1	2	5V		
229	PH5	TWI3-SDA	3	4	5V		
228	PH4	TWI3-SCK	5	6	GND		
73	PC9	PC9	7	8	UART5_TX	PH2	226
		GND	9	10	UART5_RX	PH3	227
70	PC6	PC6	11	12	PC11	PC11	75
69	PC5	PC5	13	14	GND		
72	PC8	PC8	15	16	PC15	PC15	79
		3.3V	17	18	PC14	PC14	78
231	PH7	SPI1_MOSI	19	20	GND		
232	PH8	SPI1_MISO	21	22	PC7	PC7	71
230	PH6	SPI1_CLK	23	24	SPI1_CS	PH9	233
		GND	25	26	PC10	PC10	74

nano /usr/local/lib/python3.10/dist-packages/kvmd/plugins/atx/gpio.py

```
def get_plugin_options(cls) -> dict:
    return {
        "device": Option("/dev/gpiochip0", type=valid_abs_path, unpack_as="device_path"),

        "power_led_pin":     Option(70,   type=valid_gpio_pin),
        "power_led_inverted": Option(False, type=valid_bool),
        "power_led_debounce": Option(0.1,   type=valid_float_f0),

        "hdd_led_pin":     Option(75,   type=valid_gpio_pin),
        "hdd_led_inverted": Option(False, type=valid_bool),
        "hdd_led_debounce": Option(0.1,   type=valid_float_f0),

        "power_switch_pin": Option(69,  type=valid_gpio_pin),
        "reset_switch_pin": Option(72,  type=valid_gpio_pin),
        "click_delay":     Option(0.1, type=valid_float_f01),
        "long_click_delay": Option(5.5, type=valid_float_f01),
    }
```

Activate ATX Function

nano /etc/kvmd/override.yaml

    atx:

        type: gpio

Change just from disabled to gpio, save it

maybe make a sync command, reboot.

Or restart all kvmd services, i prefer reboot.

Electric

You need the 4 lines that you have defined. (Power_LED,HDD_LED,Reset,Power)

Additionally a GND (Ground) and 3.3 volts

So total 6 lines from your GPIO Header to the ATX Interface Board.

Be careful, I don't show pictures because there are different Pi boards