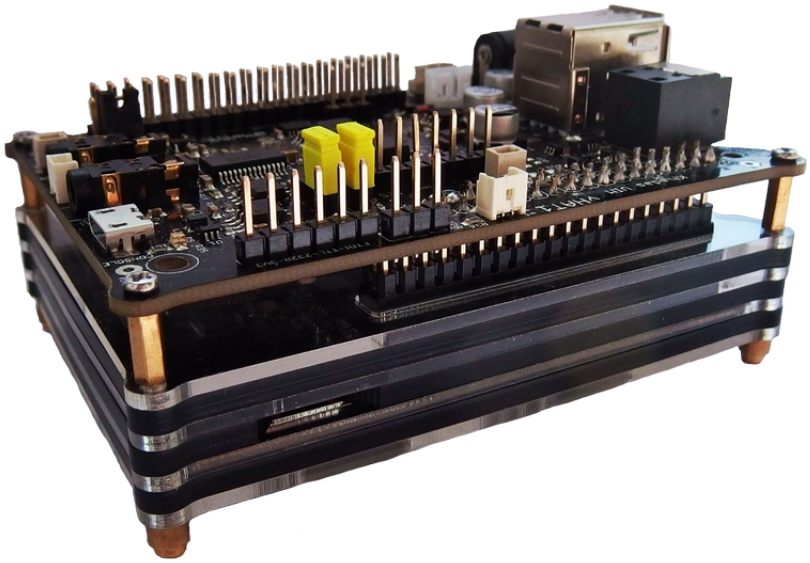
KeyHas



vHAT1
Khadas
VIM extension board

Technical Documentation

Document Revision	1.0
Approved	Filip

Revision History

Revision	Date	Author	Changes
1.0	6/18/11	FTo	Inital release

Preface

Intended Audience

This document is intended for technically qualified personnel. It is not intended for general audiences.

Symbols

The following symbols are used in this user's guide:

Warning

Warnings indicate conditions that, if not observed, can cause personal injury.

Caution

Cautions warn the user about how to prevent damage to hardware or loss of data.

Note

Notes call attention to important information that should be observed.

Terminology

Term	Description
GB	Gigabyte (1,073,741,824 bytes)
GHZ	Gigahertz (one billion hertz)
kB	Kilobyte (1024 bytes)
MB	Megabyte (1,048,576 bytes)
Mbit	Megabit (1,048,576 bits)
kHz	Kilohertz (one thousand hertz)
MHz	Megahertz (one million hertz)
N.C.	Not connected
N.A.	Not available
TBD	To be determined

Copyright Notice

Trademarks

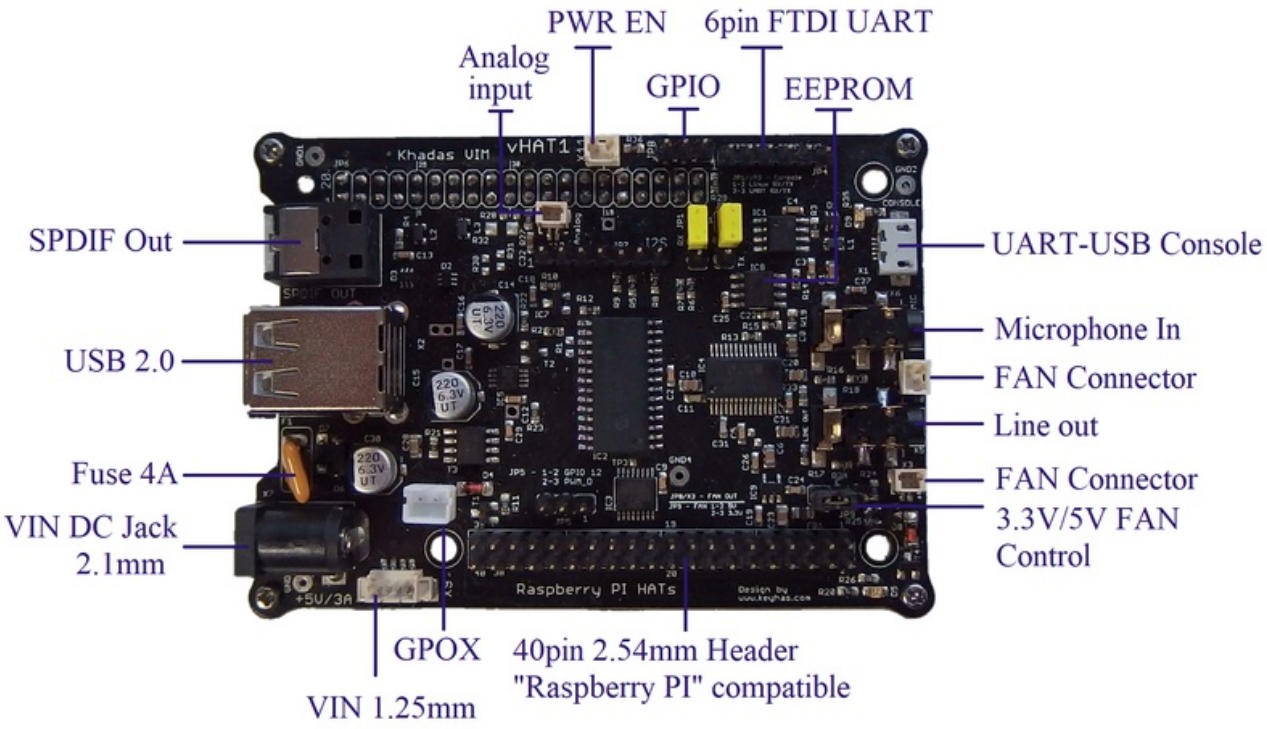
Disclaimer

This product is not intended for use in medical, life saving, or life sustaining applications. Circuit diagrams and other information relating to products are included as a means or illustrating typical applications. KeyHas makes no representations or warranties with respect to the accuracy or completeness of the contents presented in this document.

Contents

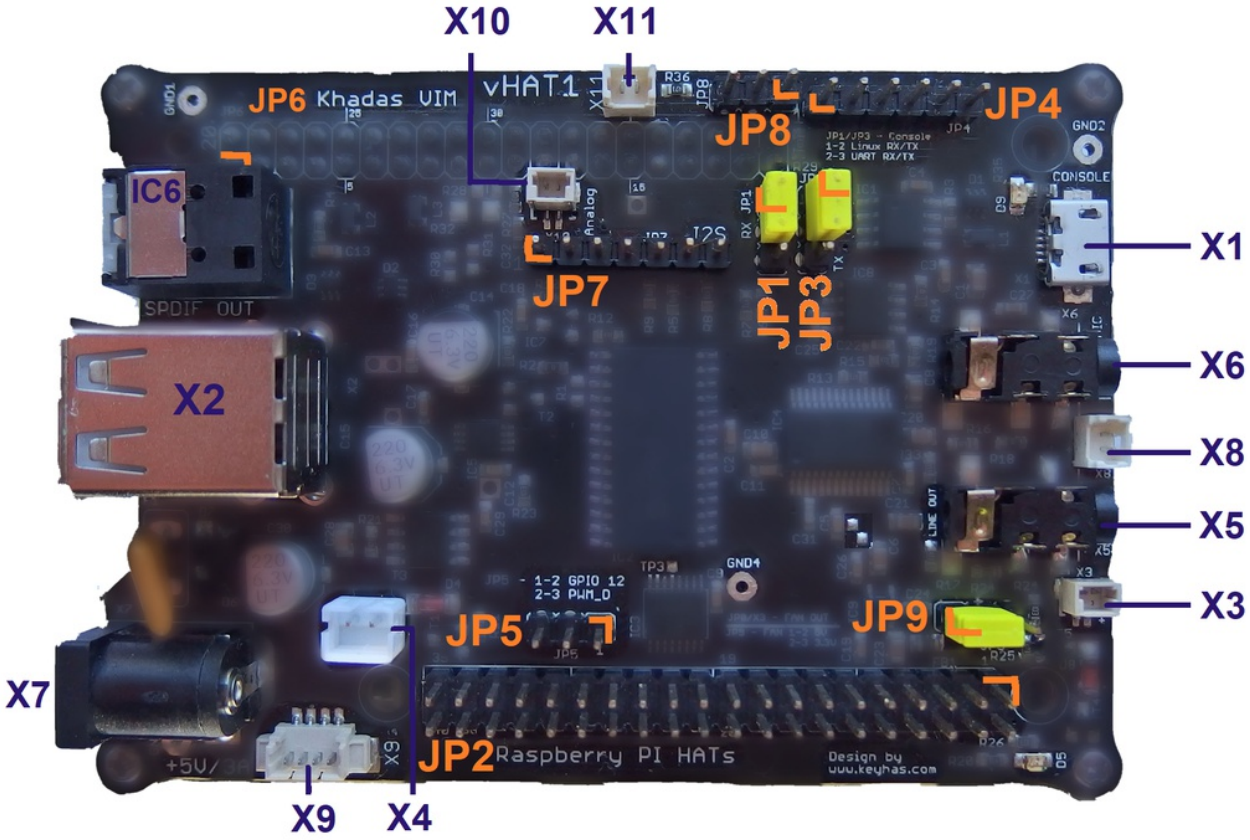
1 Main Features

- Extension to Khadas VIM/VIM2 board
- S/PDIF Output
- up-to 2x USB 2.0 Host
- DC In 2.1mm Power Jack
- Raspberry PI Hat's connections
- FAN Connector controlled by GPIO
- I2S Audio codec with Microphone In and Line Out
- UART Debug Console via USB
- and many more supports connections from Khadas VIM



vHAT1 Specifications Summary		
	VIM1	**VIM2**
VIM1/VIM2 PWR EN/ PWR ON	PWR_EN – Power Enable signal for external peripherals	PWR_ON – Power On signal for VIM2
USB 2.0 HOST from VIMx USB Host HUB	**x2 USB2.0** (1.5A load limit)	**x1 USB 2.0** (1.5A load limit)
Analog Input ADC_CH2	1.00mm Connector for analog input (0 to 1.8V) connected directly to VIM	
Input Voltage +5V/4A	DC Jack + 4pin 1.25mm (same type and pin out like VIM1/VIM2). Input power can power VIM. **note:** All peripherals on vHAT1 are controlled by +3.3V from VIM so are turn off when system is in standby, including USB connectors.	
Input Voltage protections	5V +/- 10% (maximum 5.6V / 4A).	
General Output Power Control	General output power connector	
“Raspberry PI” compatible HATs	40Pin 2.54mm Header compatible with Raspberry PI Hats. Included GPIO, SPI (from I2C->SPI), I2C, I2S, UART	
Fan Output Voltage	+5V or 3.3V selected by jumper	
Fan Control	Fan is controlled by PWM_D signal from VIM1 or VIM2	
Fan Connectors	Two Box Headers 2pin 1.25mm or 2 pin 1.00mm	
Audio Codec	WM8731 I2S Audio Codec, 96kbit/s, 90dB ADC/100dB DAC Mic In/Line out, MIC Bias support, 2x Audio 3.5mm Jack	
UART RX/TX or LNX_RX/TX (selected by JP1/JP3)	6pin FTDI Header (compatible with FTDI-TTL-232R-3.3V) and UART → USB (PL2303SA) to micro USB 2.0 output	
EEPROM	24LC64, I2C 64kBIT Electrically Erasable PROM	
I2S 7pin output	Extra I2S connections (+5V, SDIN, SDOUT, CLKs)	
S/PDIF Optical Output	16 Mbits S/PDIF optical out from VIM1/VIM2	
GPIOH_5	General Interrupt signal from all internal devices from vHAT1	
RTC_CLK (pin 36) from VIMx	Not connected	
Board Dimension	82 x 57.5 x 17 mm (board shape like VIM1/VIM2)	
Mounting Holes for vHAT1	4x 2.2mm hole (with 4x 30mm, spacer M2x8 to VIM1/VIM2)	
Mounting for Raspberry HAT	3x 3.2mm holes	
Mounting Holes for inner VIMx holes	1x 2.2mm hole 1x 3.2mm hole (shared with Raspberry HAT)	
Operations Temperature	0-50 Celsius Degree (TBD)	

2 Jumper Settings/Connectors



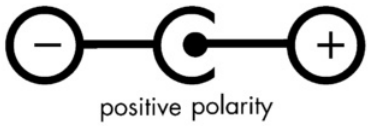
Jumpers

Jumper	*Description*	*Default*
JP1	**1-2:** Linux RX Signal to USB console/FTDI connector **2-3:** UART_RX_AO_B to USB console/FTDI connector	1-2
JP2	Raspberry PI Compatible header	No jumpers assembled
JP3	**1-2:** Linux TX Signal to USB console/FTDI connector **2-3:** UART_TX_AO_B to USB console/FTDI connector	1-2
JP4	Header for FTDI-TTL-232R-3V3 adapter or any other. 6x 2.54mm pitch	No jumpers assembled Pin1: GND Pin4: RXD Pin5: TXD
JP5	GPIO12 signal route pin32/JP2 **1-2**: GPIO12 from GPIO Expander is used for GPIO12 **2-3**: PWM_D (VIM) is used for GPIO12	1-2
JP6	Khadas VIM connections	No jumpers assembled

Jumper	*Description*	*Default*
JP7	I2S Header 1x7x 2.54mm pitch	No jumpers assembled Pin1 - GND Pin2 - I2S DIN Pin3 - +5V Pin4 – I2S MCLK Pin5 – I2S LRCK Pin6 – I2S BCK Pin7 – I2S DOUT
JP8	PWM_F / GPIODV_13 (General Input/Output) **1-2**: pin 2 to 10K Pull-up to 3.3V **2-3**: pin 2 connected to GND	No jumpers assembled
JP9	Fan Control **1-2**: +5V to both FAN connectors **2-3**: +3.3 to both Fan connectors	1-2

Connectors

Connector	*Description*	*Pinning*
X1	UART TLL to USB converter Transfer RX/TX lines from JP1/JP3 to USB for console terminal connections.	Micro USB 2.0 Connector
X2	USB 2.0 connections from VIM HUB, provide 2x/1x USB2.0 Host connections based on VIM1/VIM2. USB Lines are directly connected to Khadas VIM connector header. Secured by ESD protections. Current Limit is 1.5A per each usb port.	2x USB 2.0 A Connector
X3	Fan Connector controlled by PWM_D signal. Support off/on control (PWM TBD). Output Voltage is controlled by jumper JP9.	2pin 1.25mm (X8) or 2 pin 1.00mm (X3) (box headers) **Pin 1**: +V output from T5 **Pin 2**: GND
X4	General output connector can be use for example for relay/power transistors. Controlled by GPOX signal from GPIO chip IC2 (GPA1). In connector PIN2 is controlled by mosfet, when is active, PIN 2 is connected to ground pin.	**Pin 1**: +5V (main +5V) **Pin 2**: Mosfet 30V/3A to ground
X5	I2S Audio Codec Stereo Line out. 0 dBV output. Suitable for driving typical line loads of impedance 10K and capacitance 50pF. No really suitable for headphones.	1x Audio Jack 3.5mm

Connector	*Description*	*Pinning*
X6	I2S Audio Codec Microphone mono input in with bias support. The inputs includes programmable volume adjustments and a mute function.	1x Audio Jack 3.5mm
X7	DC Jack 2.1mm/5.5 mm provide input voltage for peripherals system including VIM1/2 board 	DC Jack 2.1mm/5.5mm Tip: +5V Sleeve: GND
X8	Second Fan connector with same functions and connections like X3	2pin 1.25mm **pin 1**: +V output from T5 **pin 2**: GND
X9	Internal connectors to provide VIN inputs like DC jack X7. Another way how to provide power into the VIM board and vHAT.	4pin 1.25mm Box header **Pin 1,2** : +5V **Pin 3,4** : GND
X10	Analog Input ADC_CH2* from VIM1/VIM2 VIM1: 10K pull-down to gnd * * possible by pull-up set signal to 3.3V/5V or make divider for 1.8V inputs. VIM2: 10K pull-up into 1.8V	2pin 1.0mm Box Header **Pin 1**: Analog ADC_CH2 **Pin 2**: GND
X11	PWR EN/PWR ON. Depends VIM1/VIM2 supports either PWR EN or PWR ON signal. Signal has pull-down 47K to GND.	2pin 1.25mm Box Header **Pin 1**: PWR_EN **Pin 2**: GND
IC6	S/PDIF Output, support up-to 16Mbit optical output to toslink optical cable.	Toslink Optical cable

3 Description

vHAT1 is extension board for VIM1 or VIM2 product and is providing connections from internal 2x20 header to various connectors and peripherals.

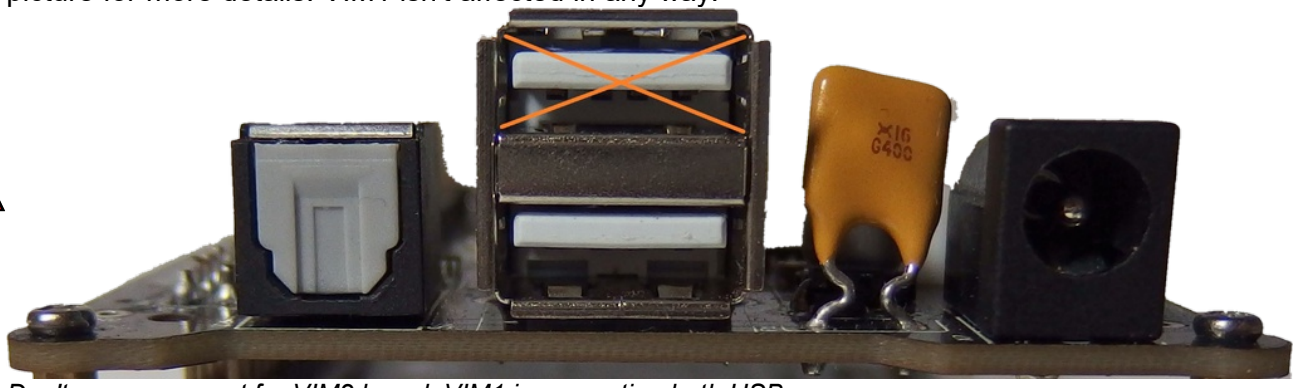
Some connections like USB, S/PDIF, I2S, UART are directly connected to the output connectors. Other connections like Audio Codec, Raspberry PI Hat, FAN control are generated internally.

Input voltage, provided either DC Jack or internal 4pin connector is limited to +5V +/- 10% and is protected by poly fuse 4A for VIM and internal peripheral, except USB Power. USB power switch is connected directly from DC input. Voltage Inputs is protected against reverse voltage by diode and over voltage by simple zener diode. Wrong polarization can caused HW damage of that protection diode, but should not damage others circuit.

All internal voltages of peripherals are controlled by VIM by +3.3V, so all are off in case when VIM is in standby mode, including USB ports. For permanent functions of USB please use internal USB connectors from VIM.

USB Connectors using 1.5A current limiter per port. Over current signal is shared by both ports and is carried by internal GPIO Expander into the system. Both USB ports are supported only by VIM1. Where VIM2 hasn't second USB connections and connections to the USB cannot be used. So USB over-current signal is valid for one port in VIM2.

Due to compatibility of vHAT1 for VIM1 and VIM2, please do not use a upper USB port for VIM2 due to connections to the IO ports directly to the VIM2. Improper use can damage VIM2. See picture for more details. VIM1 isn't affected in any way.



Don't use upper port for VIM2 board. VIM1 is supporting both USB

S/PDIF using standard toslink transceiver supports any optical toslink cable use by audio equipment. The signal from VIM is not in any way manipulated so capability of S/PDIF out is fully controlled by SW/HW running in the VIM. Hardware limits is reported as 16Mbit/s. By demand is possible to use 24Mbit/s transceiver.

UART to MicroUSB converter using PL2303SA chip, requiring driver installed in host system. That chip providing virtual UART running in host system. vHAT1 and virtual com is connected in cross connections. RX signals from virtual com going to TX signals from internal UART and vice versa. User can either select internal connections to debug connections or UART_xx_AO_B UART of VIM1. Connections from vHAT1 to Host system is provide by microUSB

cable. Another capability for connections is used internal header (JP4) for example FTDI device FTDI-TTL-232R-3V3, all signals are 3.3V voltage capable.

vHAT1 contains I2C Serial EEPROM, that eeprom is delivered empty and can be used for any internal purpose, it is connected to the I2C_xxx_B channel. Currently is assembled 64kbit version. Address is fixed by resistors, but address A2 bit can be changed by removing R14.

FAN on/off Control is controlled by PWM_D signal from VIM1, real PWM functions needs to be tested. Both fan output FAN connectors (X3, X8) support same functions. The output voltage use for fan can be switched between 3.3V or 5V. Output current is up-to 3A, so can support almost any FAN or multiple fans in one time.

The board contains universal output transistor controlled by internal GPIO signal. That output is going to connector and can be used for any application where signal needs to be tide to ground (like relay). First pin of output connector is providing +5V for inputs like +5V relay. The second pin is controlled by mosfet. GPIO signal is provide by internal GPIO converter and is connected to the signal GPA1 (bit1 of PORTA). That general output not provide any PWM functionality.

Raspberry PI HAT 40pin connections (supports RPI2 B, RPI3..etc) is provide by internal circuits or directly connected from VIM. The GPIO Signals are generated by I2C->GPIO internal chip. The SPI signals are generated from I2C->SPI converter. Buses like I2C, I2S and UART connections are provided by VIM. The VIM's I2C channel A is used as main I2C channel for the connectors, where second channel B is used only for internal Raspberry HAT eeprom connections.

GPIO Expander (IC2) from Microchip, MCP23017 is providing GPIO signals for 40pin output connectors, bit 0 of port A is using as signal of USB over signaling and bit 1 of port A is use for general output transistors, others signals are routed to the 40pin output connectors. All signals are 3.3V capable with pull-up capability, each bit can configured as output or input. All GPIO signals aren't +5V tolerant.

SPI Bridge (IC3) from NXP, SC18IS602B is designed to serve as an interface between the standard SPI of a VIM. Two slave select signals are routed to 40pin output connectors. Third slave selected is available as Test Point TP3 on the board.

Audio Codec (IC4) from Cirrus Logic, WM8731 is provide line out and microphone in and use I2S connections. Chip is controlled by I2C bus and needs to be configured before first use.

Line Out output has 100R resistors in series and is limiting to drive headphones outputs. External amplifier needs to be used. Headphones outputs from the chip are connected to internal Test Points. Requiring at least 220uF dc coupling capacitors, what was not possible to add into the boards due space limitation. For experimental testing, is possible to remove resistors R33/R34 and C20/C21 and use external cap to drive headphones output.

Analog DC Input choose signal ADC_CH2 (pin 11) connected directly from the VIM. Due compatibility between VIM1 and VIM2 there are different behavior between both versions. For VIM1 there is pull down resistors (R27) to ground. This resistors can be used and part of non assembled pull -up resistors (R28/R32) and create divider to achieve maximum 1.8V. For VIM2 is this resistors (R27) acting as pull-up to 1.8V, so no necessary dividers are necessary.

Extra details about internal Signals from VIM, like:

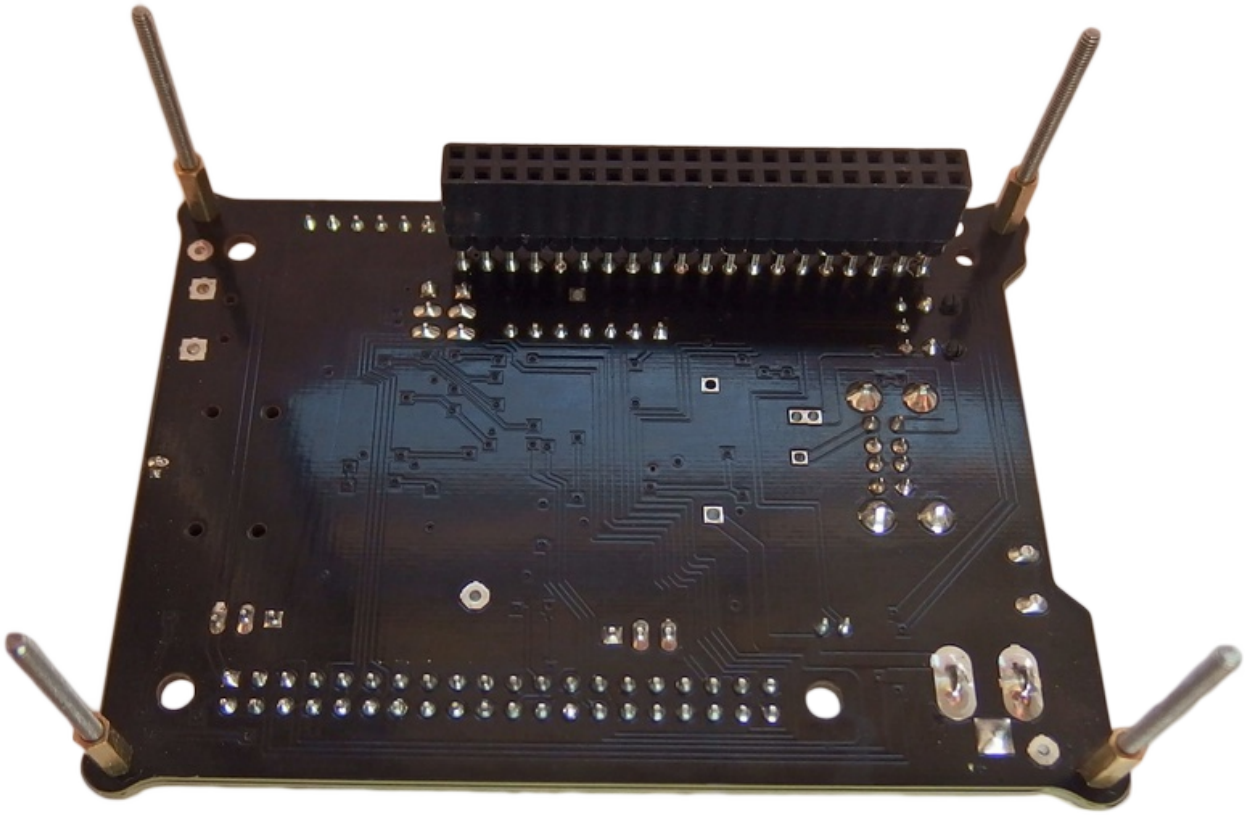
VIM	***Pin***	***Description***
I2C_xxx_A	22/23	Routed directly to 40pin output connector, not used in vHAT1
I2C_xxx_B	25/26	Used for all I2C Chips in vHAT1 (GPIO, SPI, Audio Codec, EEPROM) and connected to 40pin output connector
I2S Signals	29-33	I2S Audio Codec + I2S Header + 40pin output connector
PWM_D	35	Used for FAN control and can be routed to 40pin output connectors. Has pull-up 10K to 3.3V
RTC_CLK	36	Isn't routed and used in vHAT1
GPIOH_5	37	It is used as general interrupt signals from all peripherals (USB, I2C->SPI bridge, GPIO Expander). Has pull-up 10K to 3.3V
PWR_EN PWR_ON	38	Signals is going to connectors X11 and has internal weak pull-down to the ground. Can be use for external usage.
PWM_F GPIODV_13	39	In VIM1 is routed as PWM_F signal in VIM2 is routed GPIODV_13.

4 Assembly Instructions

You need take:

4x M2x30 mm screws
4x M2x8 mm spacer
1x vHAT1
1x Khadas VIM

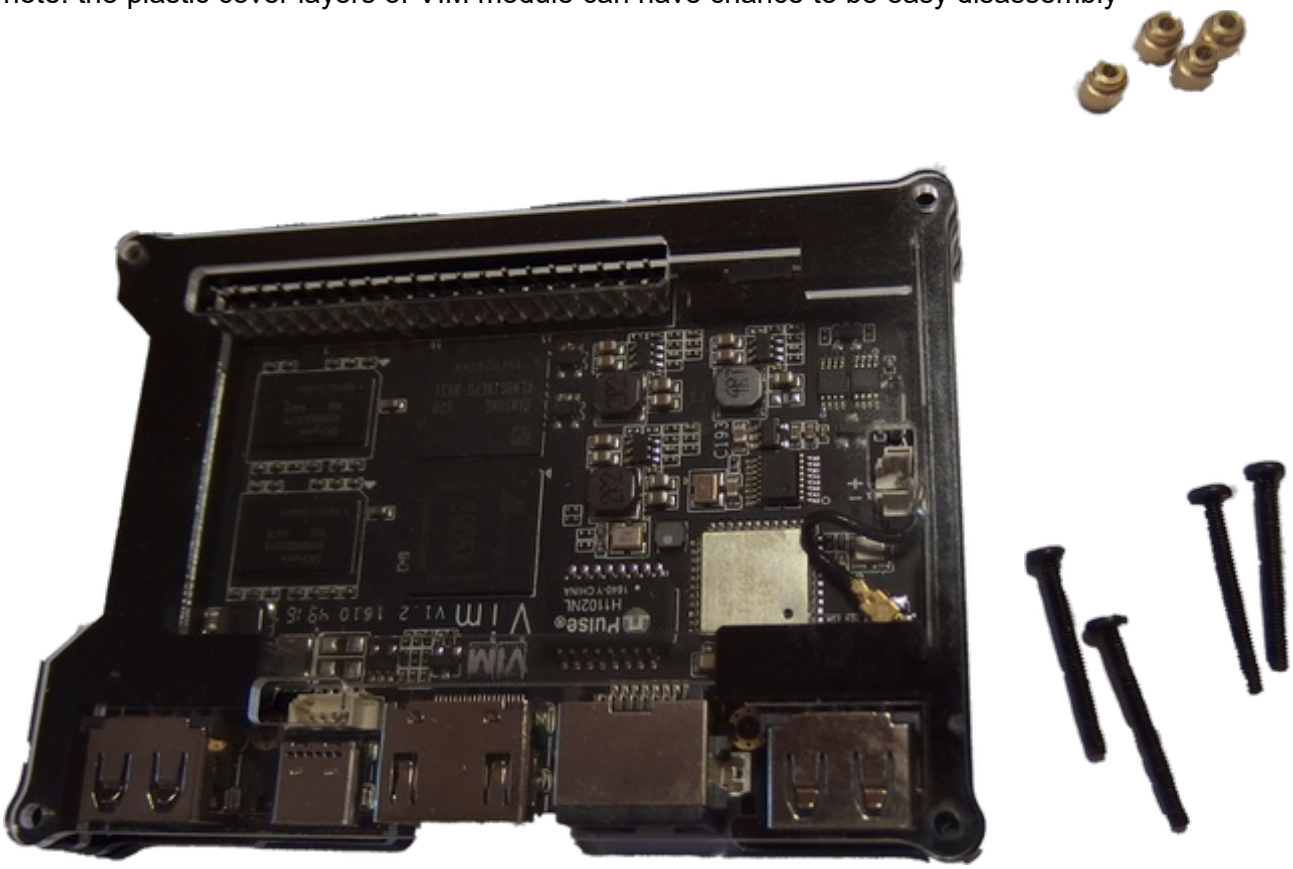
1. Combine vHAT1 with M2x30 mm screws and spacer like in picture



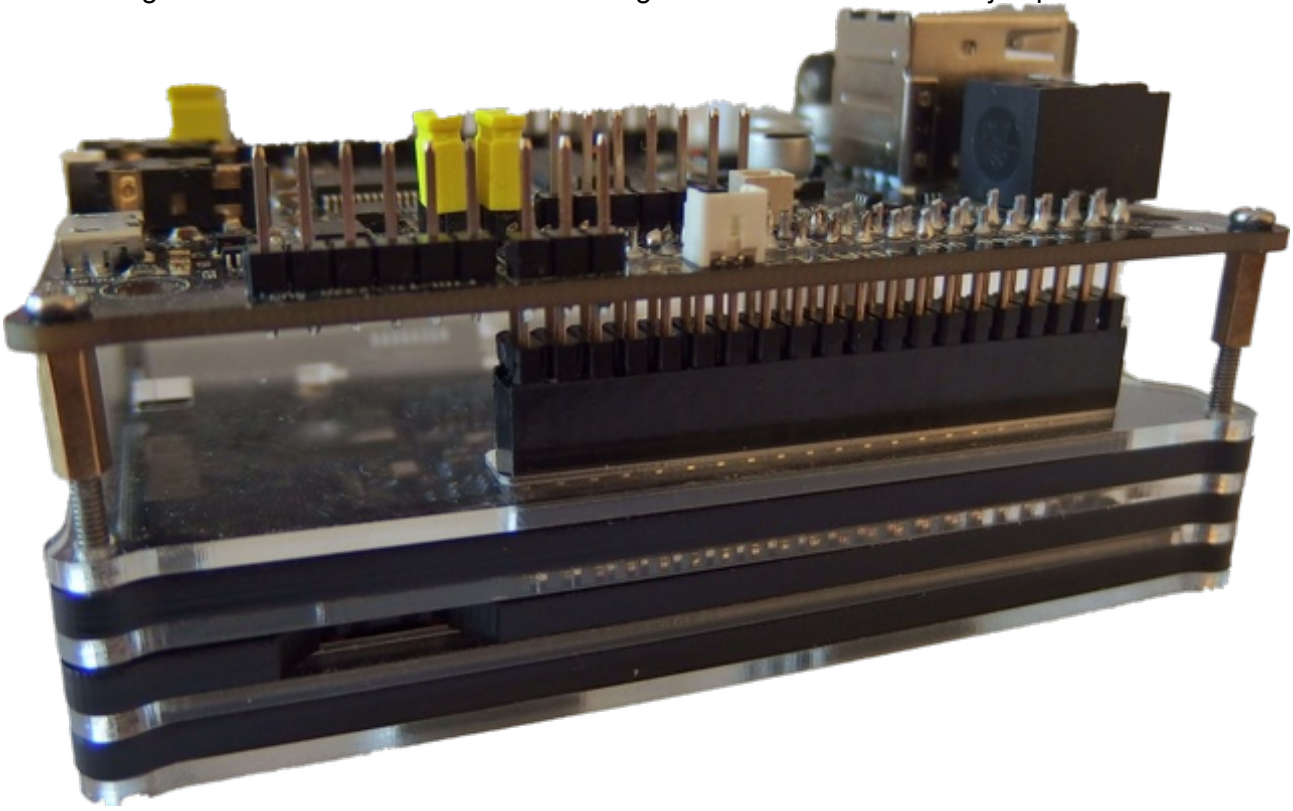
2. Unscrew original nut from vHAT1 (don't lose them)

3. Carefully remove original blacks screws from the VI

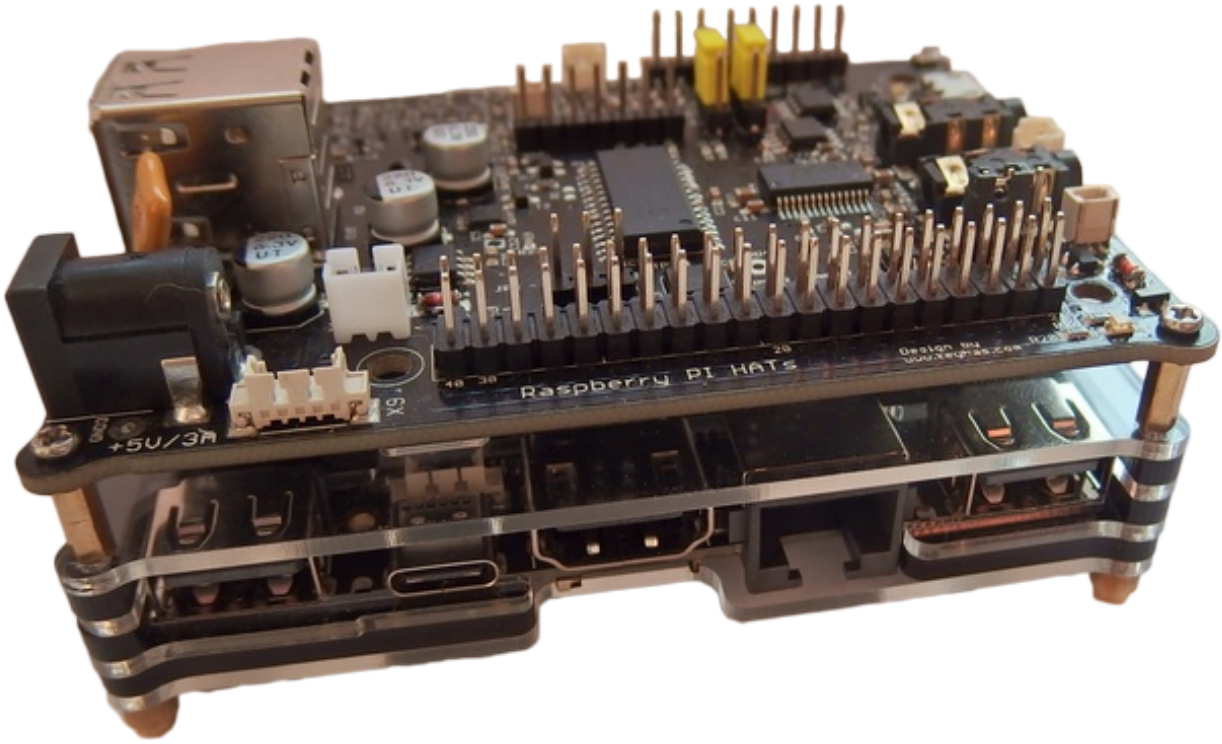
note: the plastic cover layers of VIM module can have chance to be easy disassembly



4. Plug the vHAT1 into holes of vHAT1 till signal connectors will hit the jumper in VIM



5. Slightly plug the vHAT1 into VIM
6. Screws M2x30 screws with nuts from VIM1



5 Brings Up/ First use

vHAT1 doesn't require any special actions for brings up, all devices are pretest and jumpers settings is set to default. To check all features of vHAT1 is need support from running OS.

The USB ports, USB-UART console and FAN are supported by any build. SPDIF can be limited in some builds, successfully was tested under Android 6 and 7. Audio codec and internal I2C devices needs bus support of I2S and I2C in OS.

5.1 USB-UART Console

In case of using, is recommend to start with UART-USB console. (how-to present use under Microsoft windows and Xubuntu/Ubuntu , but idea can be apply in similar way in others operations systems)

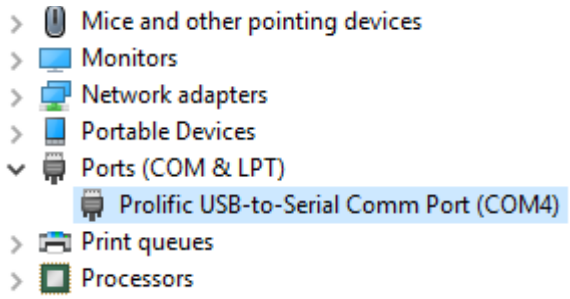
By console you can control VIM. Console program offer same access to the system like by console program running in OS. Connections not provide any graphical outputs. It's mainly helpful for debugging or quick tests or change settings during boot.

5.1.1 USB-UART under Microsoft Windows:

- Install driver support in host system for virtual USB support
  see: Prolific PL2303SA webpage

- Install/Download propriety console program (in examples is used putty for Windows)
  see: http://www.putty.org/ - Windows based console program

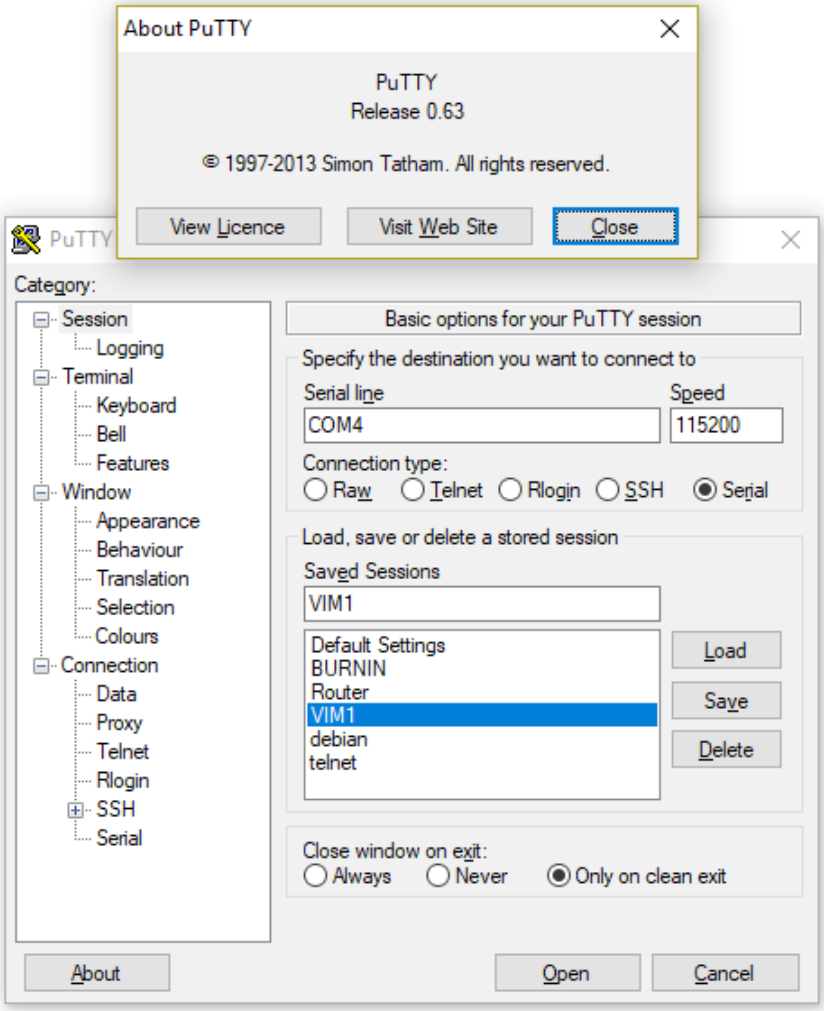
- Plug micro usb cable into X1 and identify serial ports from virtual-usb (or reinstall drives if is needed),



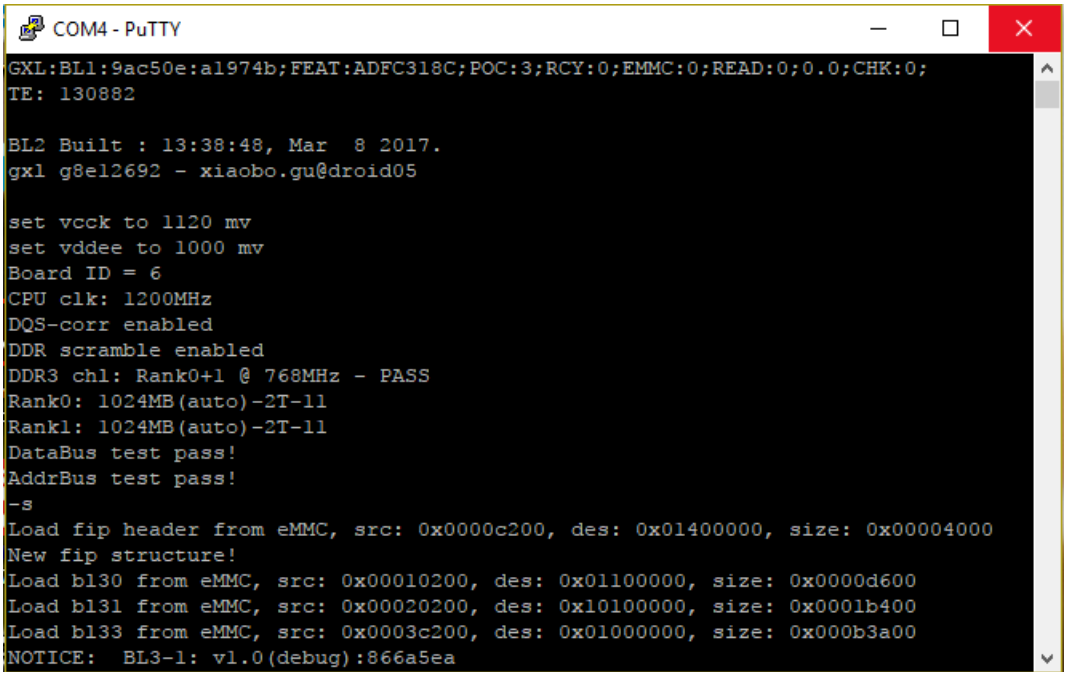
  (the serial port can be vary in every system, is recommend to have serial ports between com 1-9 to keep support for some tools)

  In vHAT1 is D9 green led indicate power for UART-USB, from that point Host system should find a new virtual ports

- establish communications to see console output. Default settings is 115200bps, 8bits, choose right virtual port COM. Click on “Open” button to run console.



- Power VIM by USB-C cable, in the console should be appear output from booting of VIM.



5.1.2 USB-UART under Xubuntu 16.04 LTS

- Default Xbuntu installations already support PL2303 drivers, so is require Plug micro usb cable into X1 and identify serial ports from pl2303 driver. In vHAT1 is green led D9 indicate power for UART-USB, from that point Host system should find a new virtual ports, please call

```
dmesg | grep pl2303

[  118.838904] usbcore: registered new interface driver pl2303
[  118.838941] usbserial: USB Serial support registered for pl2303
[  118.838964] pl2303 1-2:1.0: pl2303 converter detected
[  118.841124] usb 1-2: pl2303 converter now attached to ttyUSB0
```

- in my example ttyUSB0 device file is assigned to converter

- install any console program, for example screen or putty

```
sudo apt-get install screen or
sudo apt-get install putty
```

- run command to open console window

```
sudo screen /dev/ttyUSB0 115200 or
sudo putty
```

5.2 Audio Codec Initialization

In webpages can be found a image of Android 7 supporting I2S and I2C bus, please use before use of codec. This require re flash images in vim: see tutorial in

http://docs.khadas.com/bootcamp/UpgradeViaUSBCable/

The Audio codec needs to be initialize to support audio output. Please copy that commands to console for codec initialization:

```
su                          # get root access
i2cset -y 2 0x1a 0x1E 0x0   #safe to makes reset
i2cset -y 2 0x1a 0x00 0x40  #optional line in mute
i2cset -y 2 0x1a 0x02 0x40  #optional line in mute
i2cset -y 2 0x1a 0x04 0xE5  #optional=default
i2cset -y 2 0x1a 0x06 0xE5  #optional=default
i2cset -y 2 0x1a 0x08 0x16  #mutemic=1,Insel=1(Mic),Bypass=0,DACSEL=1
i2cset -y 2 0x1a 0x0a 0x00  #DAC SoftMute=0
i2cset -y 2 0x1a 0x0C 0x61  #Line Input=off(1), CLKOUT=off
#CLKOUT=off(1), OSCPD=1 (use MCLK as input), rest on
i2cset -y 2 0x1a 0x0e 0x02  #I2S, 16bit,
i2cset -y 2 0x1a 0x10 0x20  #Normal Mode=0,BOSR=0, SR=1000 44.1kHz,256fs
i2cset -y 2 0x1a 0x12 0x01  #Active codec
```

note: until first sound, codec can generate a digital noise on output.

5.3 Others I2C Devices

The devices in vHAT1 are connected to I2C_xxx_B bus, what is I2C-2 bus in VIM. I2C detect command, can find all device except Audio Codec (not responding on read commands), see

i2cdetect -y -r 2

```
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- 24 -- -- -- 28 -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- 51 -- -- 54 -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- -
```

i2cdetect can flooded lines before table appears. Issues is under investigations, but I2C bus can works properly.

```
[  279.791141@0] i2c i2c-2: [aml_i2c_xfer] error ret = -5 (-EIO)
[  279.796764@0] i2c i2c-2: token 1, master_no(2) 300K addr 0x76
```

The I2C address are follow:

```
0x24    IC2:MCP23017S0 (I2C ==>GPIO)
0x28    IC3:SC18IS602B (I2C ==>SPI)
0x1A    IC4:WM8731 (I2C,I2S ==>DAC)
0x54    IC8:24C64N (I2C ==> EEPROM)
-----
0x51    YM8563 (RTC chip from VIM, similar to PCF8563)
```