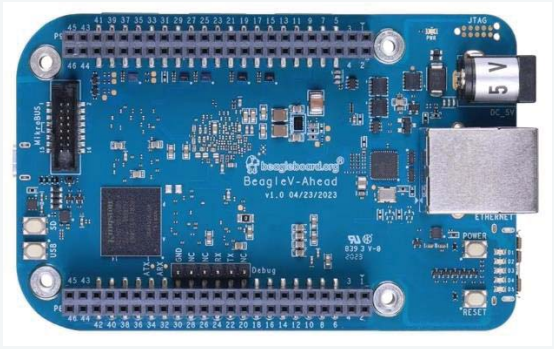
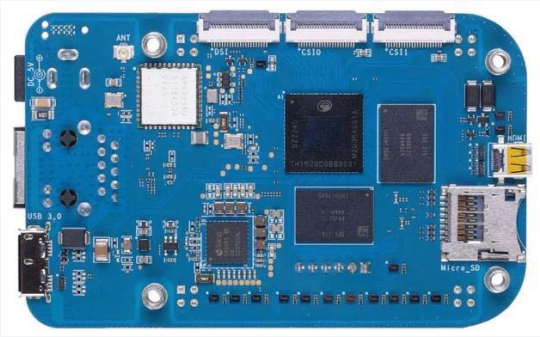
Introduction

BeagleV Ahead is an open-source RISC-V Single Board Computer (SBC) in the form factor of BeagleBone Black. It has the same P8 & P9 cape header pins as BeagleBone Black allowing you to stack your favourite BeagleBone cape on top to expand it's capability. Featuring a powerful quad-core RISC-V processor BeagleV Ahead is designed as an affordable RISC-V enabled pocket-size computer for anybody who want's to dive deep into the new RISC-V ISA.





Detailed overview

BeagleV Ahead is build around T-Head TH1520 RISC-V SoC with quad-core Xuantie C910 processor clocked at 1.85GHz with a 4 TOPS NPU, support for 64-bit DDR, and audio processing using a single core C906.

Table 64 BeagleV Ahead features

Feature	Description
Processor	T-Head TH1520 (quad-core Xuantie C910 processor)
PMIC	DA9063
Memory	4GB LPDDR4
Storage	16GB eMMC
WiFi/BlueTooth	• PHY: AP6203BM • Antennas: 2.4GHz & 5GHz

Table 64 BeagleV Ahead features

Feature	Description
Ethernet	• PHY: Realtek RTL8211F-VD-CG Gigabit Ethernet phy • Connector: integrated magnetics RJ-45
microUSB 3.0	• Connectivity: USB OTG, Flash support • Power: Input: 5V @ <To-Do>, Output: 5V @ <To-Do>
HDMI	• Transmitter: TH1520 Video out system • Connector: Mini HDMI
Other connectors	• microSD • mikroBUS connector (I2C/UART/SPI/MCAN/MCASP/PWM/GPIO) • CSI connector compatible with BeagleBone AI-64, Raspberry Pi Zero / CM4 (22-pin) • DSI connector

Board components location

This section describes the key components on the board, their location and function.

Front components location

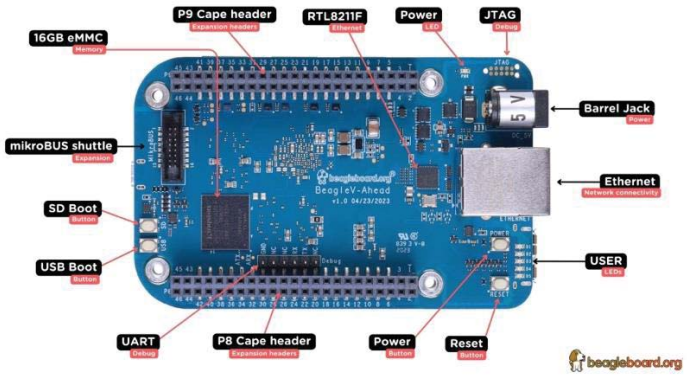


Fig. 178 BeagleV Ahead board front components location¶

Table 65 BeagleV Ahead board front components location

Feature	Description
Power LED	Power (Board ON) indicator
JTAG (TH1520)	TH1520 SoC JTAG debug port
Barrel jack	Power input
GigaBit Ethernet	1Gb/s Wired internet connectivity
User LEDs	Five user LEDs, Power and boot section provides more details. These LEDs are connect to the AM6254 SoC
Reset button	Press to reset BeagleV Ahead board (TH1520 SoC)
Power button	Press to shut-down (OFF), hold down to boot (ON)
P8 & P9 cape header	Expansion headers for BeagleBone capes.
UART debug header	6 pin UART debug header
USB boot button	Hold and reset board (power cycle) to flash eMMC via USB port
SD boot button	Hold and reset board (power cycle) to boot from SD Card
mikroBUS shuttle	16pin mikroBUS shuttle connector for interfacing mikroE click boards
16GB eMMC	Flash storage
RTL8211F	Gigabit IEEE 802.11 Ethernet PHY

Back components location

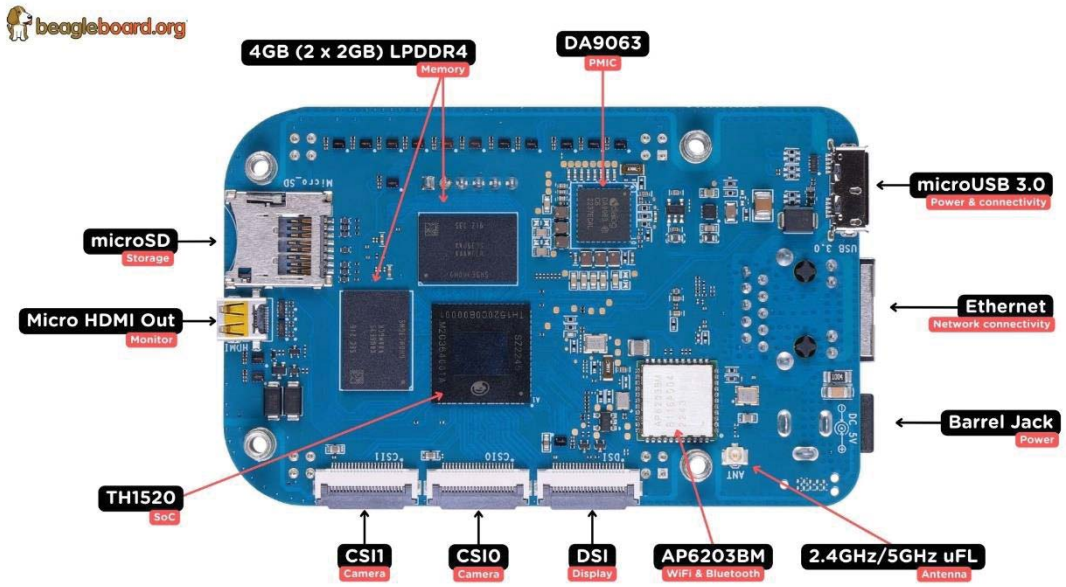


Fig. 179 BeagleV Ahead board back components location¶

Table 66 BeagleV Ahead board back components location

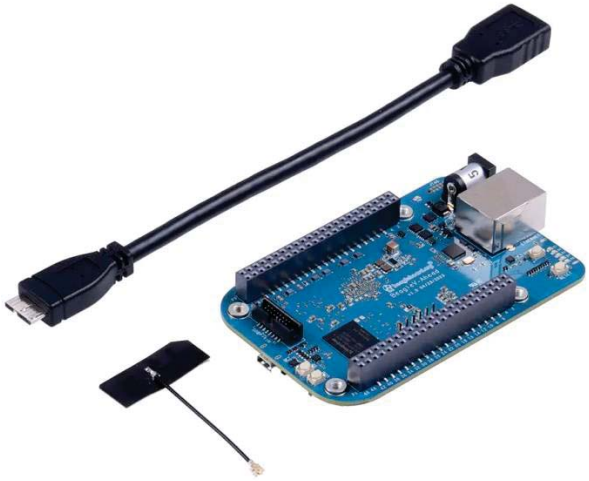
Feature	Description
DA9063	Dialog semi Power Management Integrated Circuit (PMIC)
microUSB 3.0	Power & USB connectivity as client or Host (OTG)
Antenna connector	2.4GHz/5GHz uFL connector
AP6203BM	Ampak WiFi & BlueTooth combo
DSI	MIPI Display connector
CSI0 & CSI1	MIPI Camera connectors
TH1520	T-Head quad-core C910 RISC-V SoC
Mini HDMI	HDMI connector
microSD	SDCard holder
4GB RAM	2 x 2GB LPDDR4 RAM

Quick Start

What's included in the box?

When you purchase a brand new BeagleV Ahead, In the box you'll get:

1. BeagleV Ahead board
2. One (1) 2.4GHz/5GHz antennas
3. microUSB OTG cable
4. Quick-start card



Design & specifications

If you want to know how the BeagleV Ahead board is designed and what are it's high-level specifications then this chapter is for you. We are going to discuss each hardware design element in detail and provide high-level device specifications in a short and crisp form as well.

Block diagram

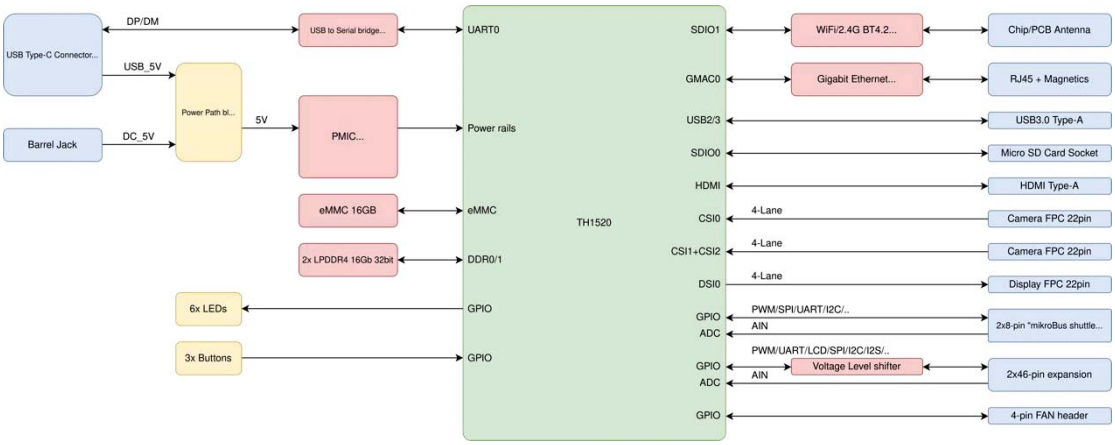


Fig. 180 System block diagram¶

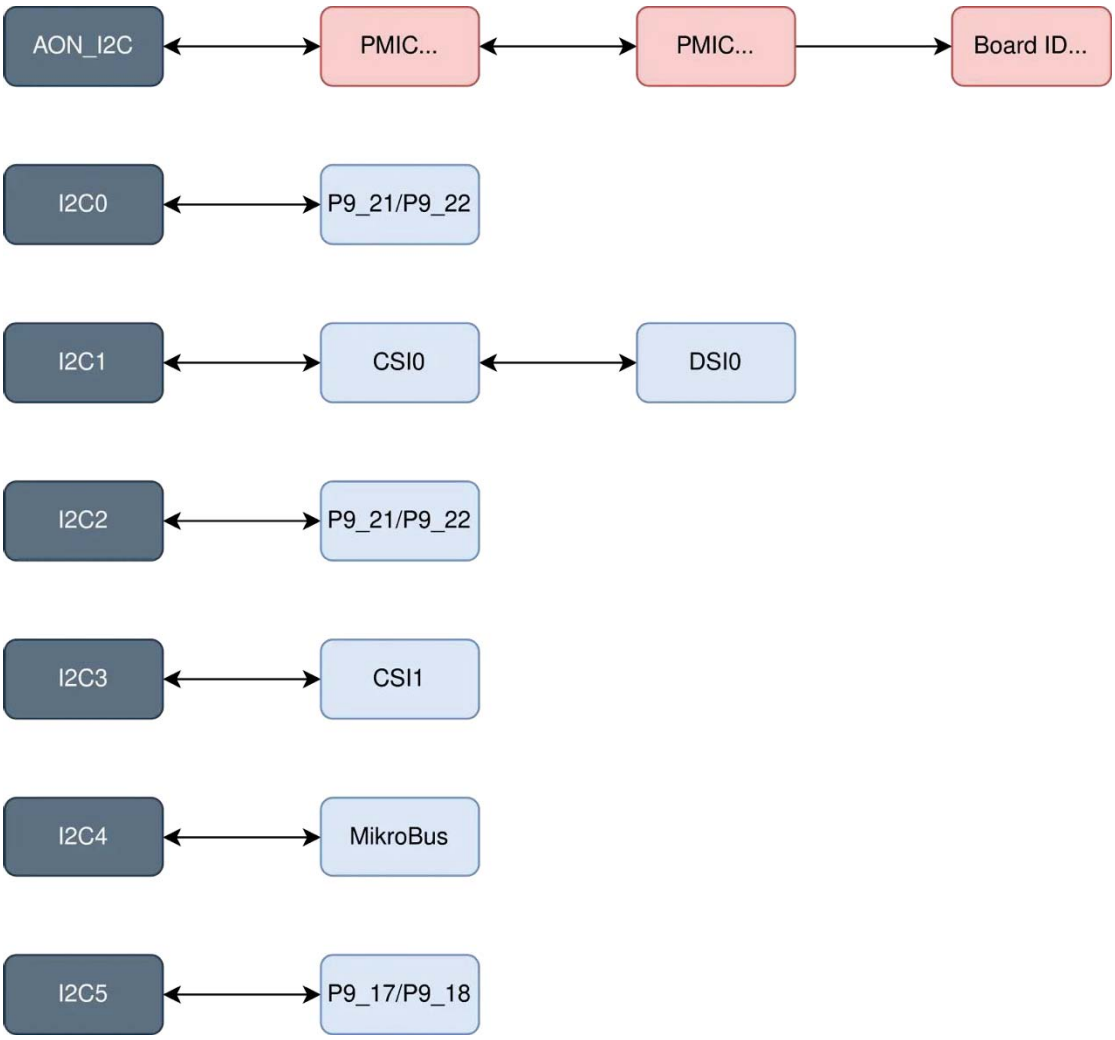


Fig. 181 I2C-Usage diagram¶

System on Chip (SoC)

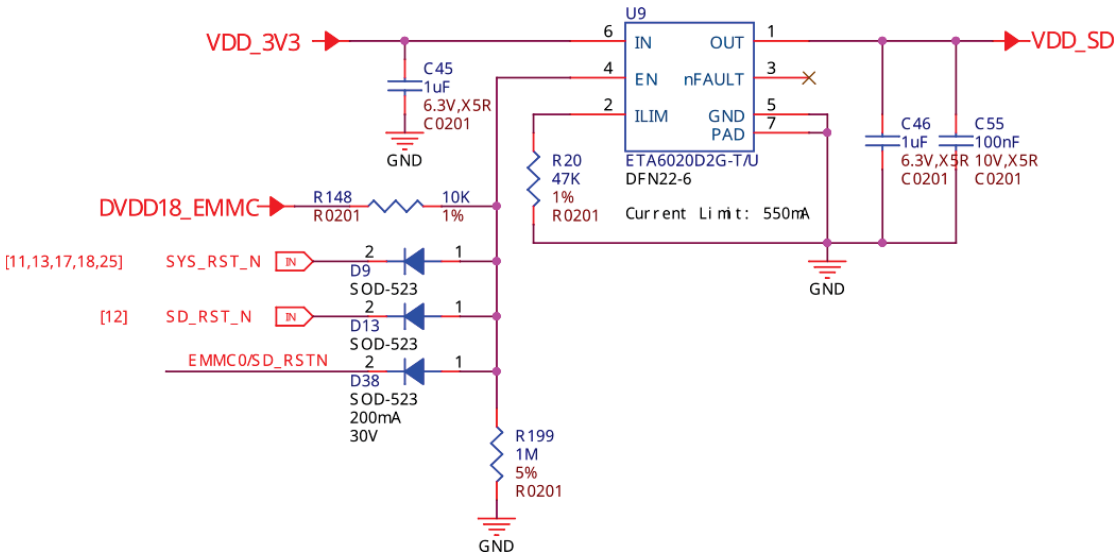


Fig. 182 SoC eMMC power switch¶

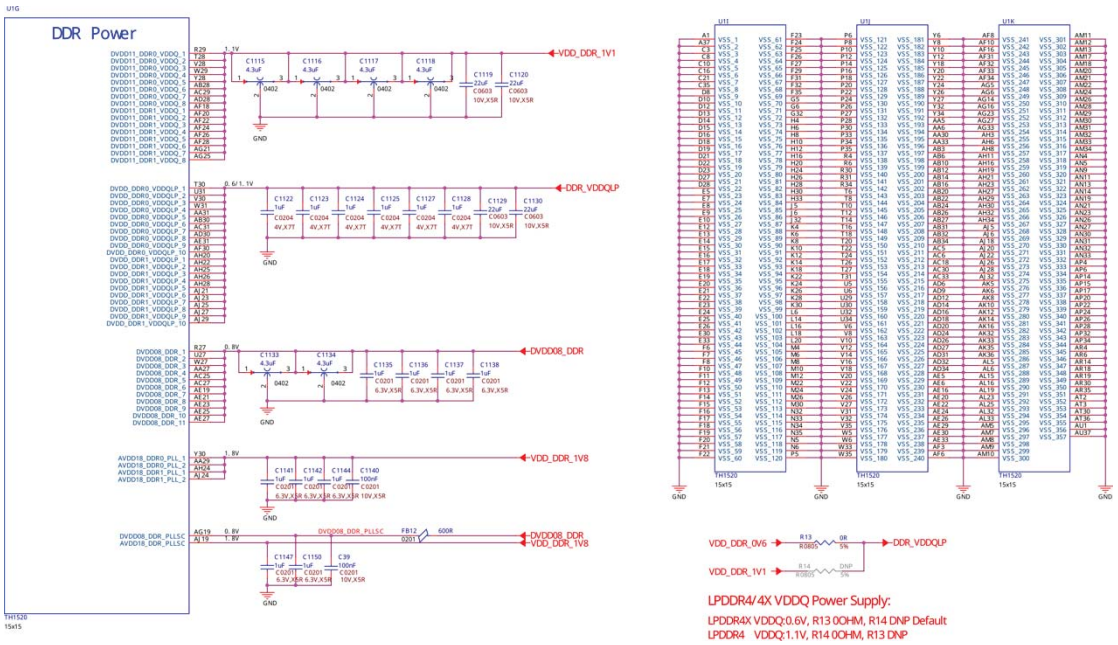


Fig. 183 SoC DDR Power¶

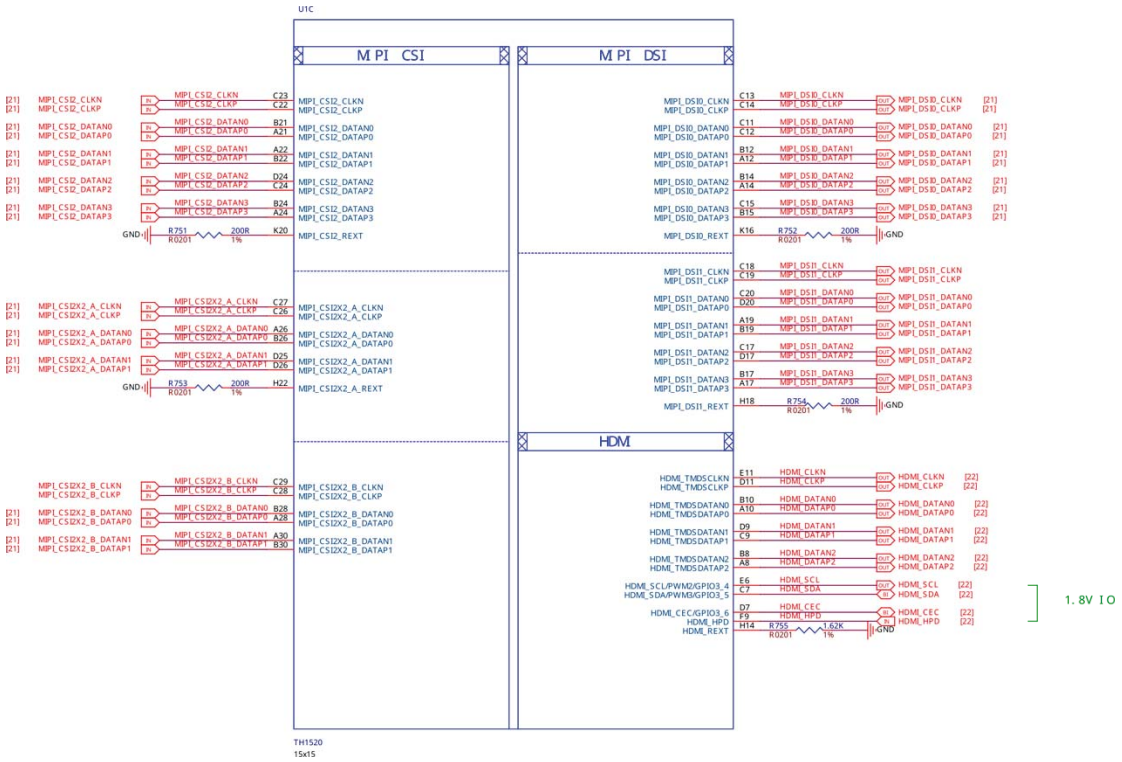


Fig. 184 SoC MIPI CSI DSI HDMI¶

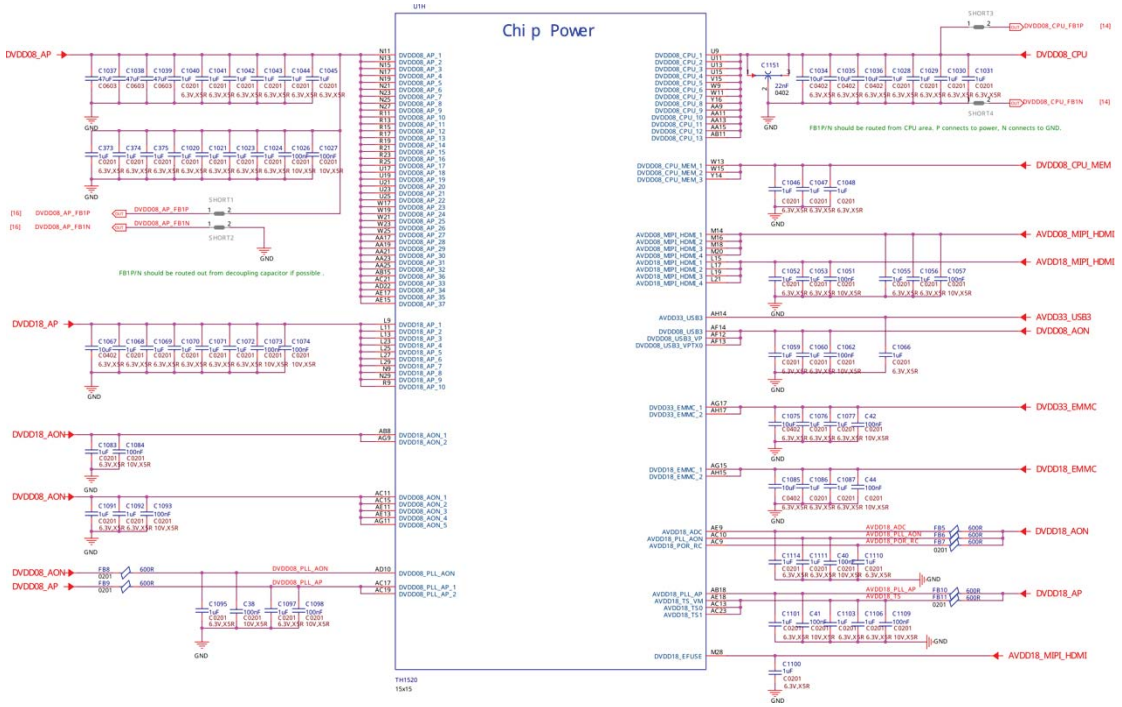


Fig. 185 SoC power¶

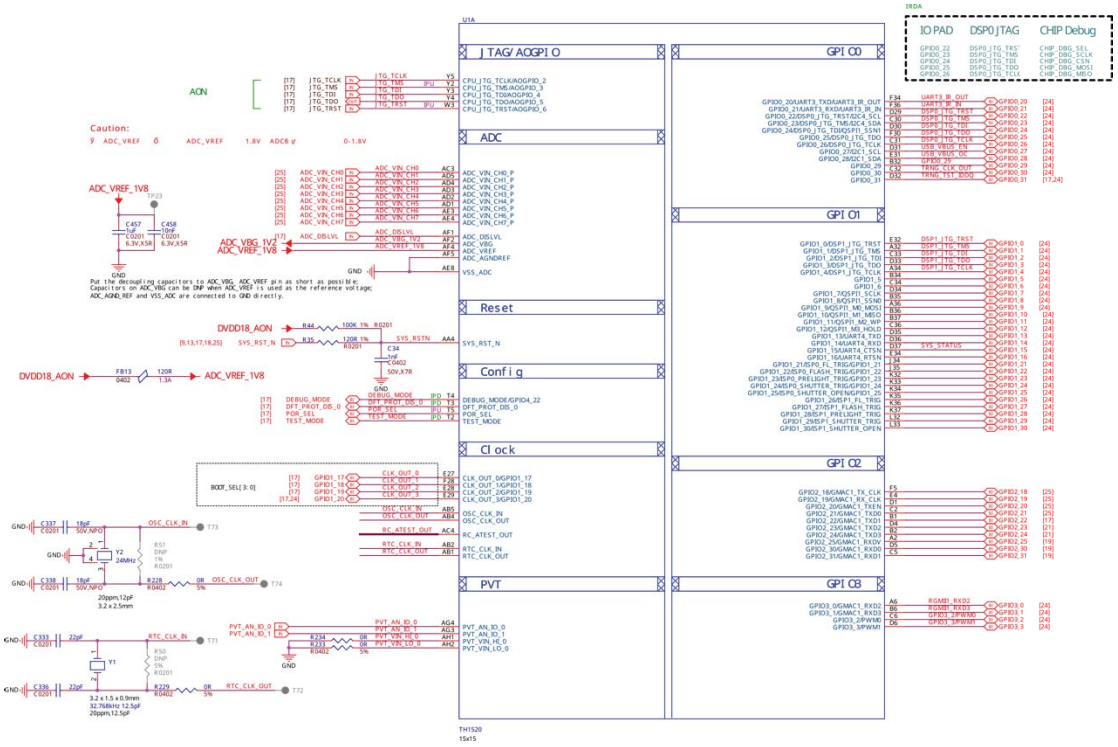


Fig. 186 SoC sys, ADC, and Clock¶

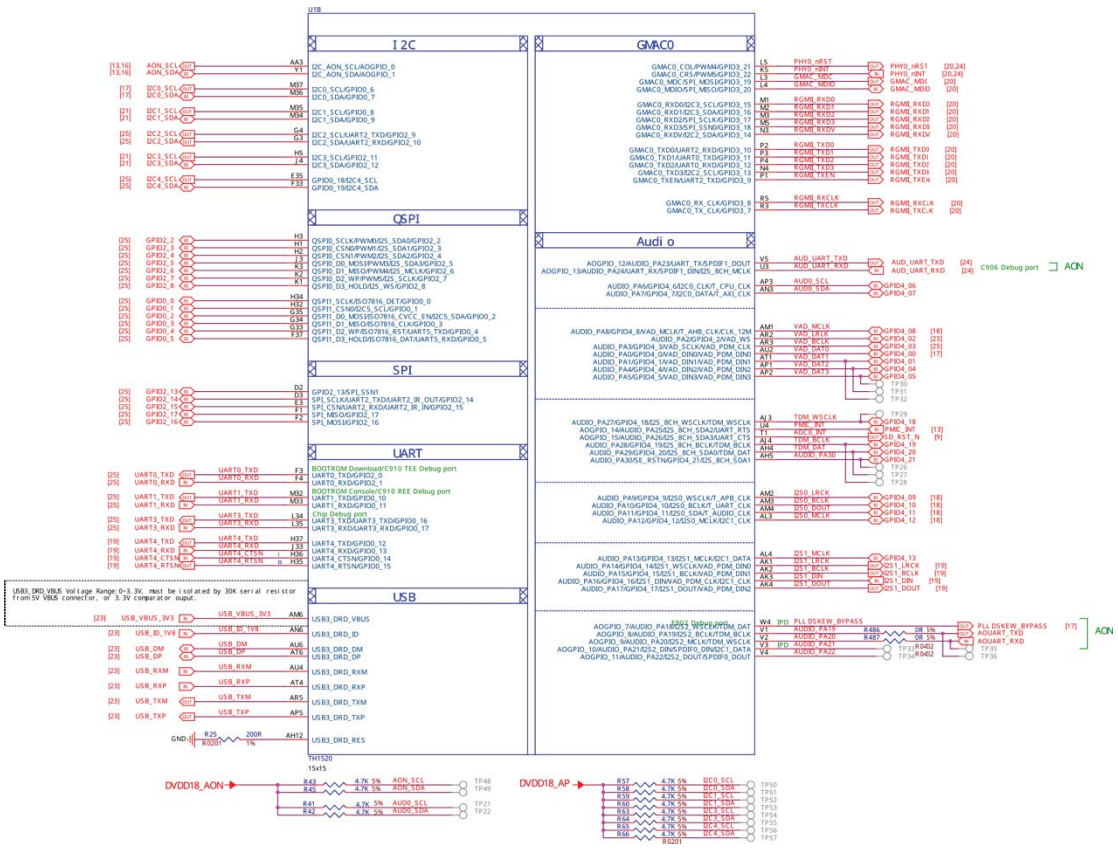


Fig. 187 SoC USB GMAC Audio¶

Power management

Barrel jack

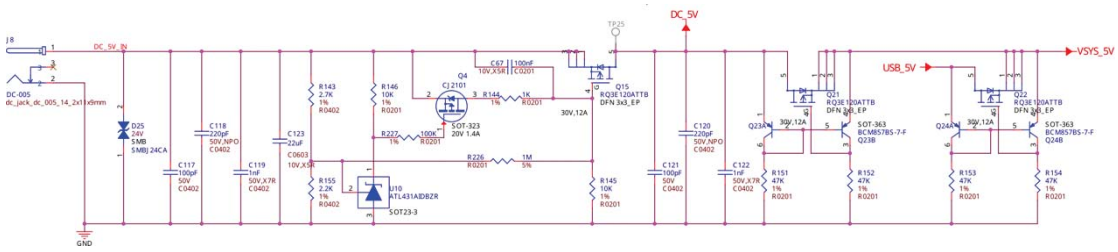


Fig. 188 Barrel jack power input¶

0.8V DCDC buck

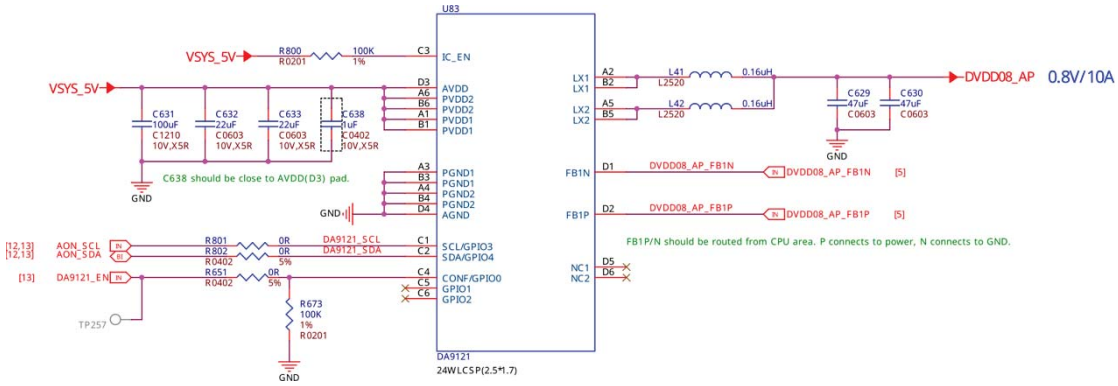


Fig. 189 0.8V DCDC buck converter¶

3.3V DCDC buck

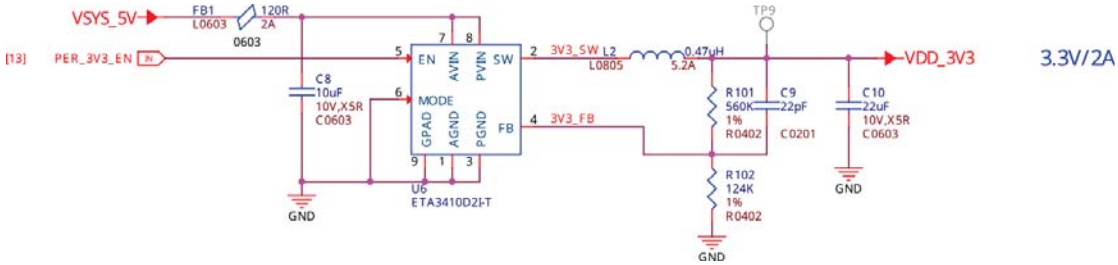


Fig. 190 3.3V DCDC buck converter¶

1.8V LDO

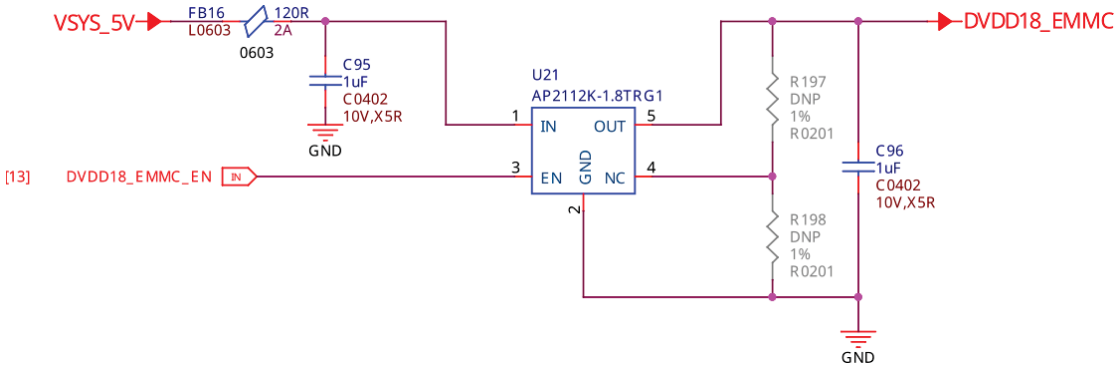


Fig. 191 1.8V LDO regulator¶

PMIC

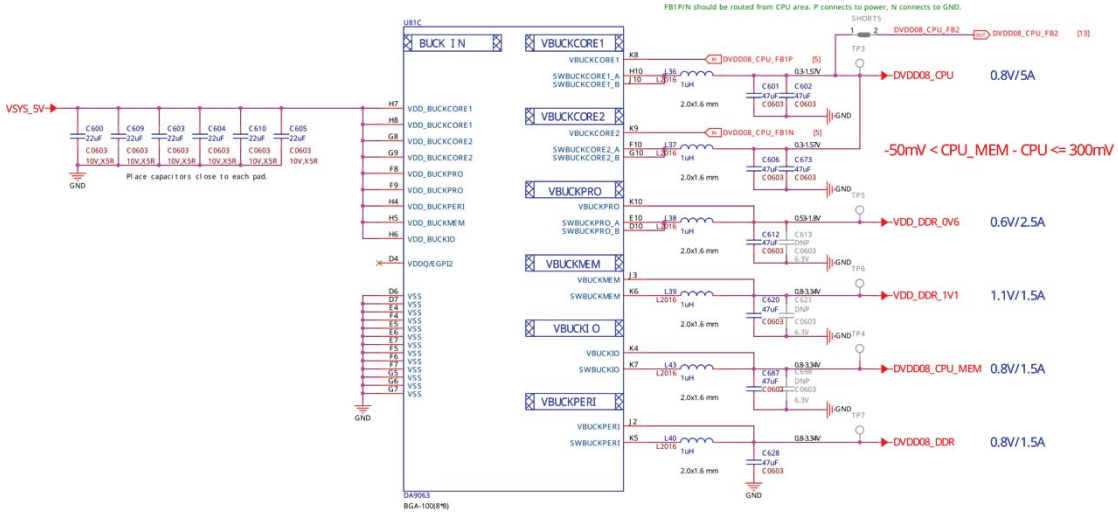


Fig. 192 PMIC Buck¶

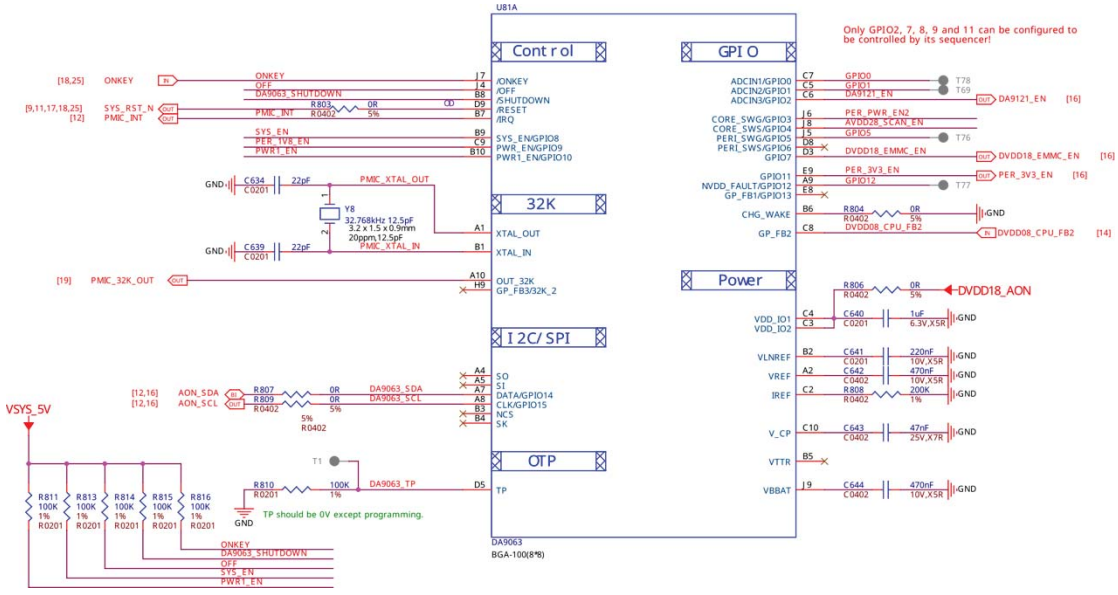


Fig. 193 PMIC Control¶

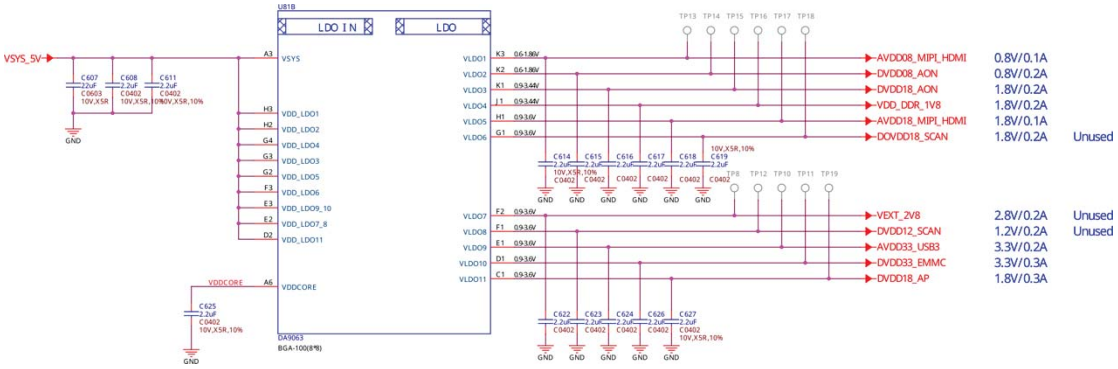


Fig. 194 PMIC LDO¶

General Connectivity and Expansion

microUSB 3.0 port

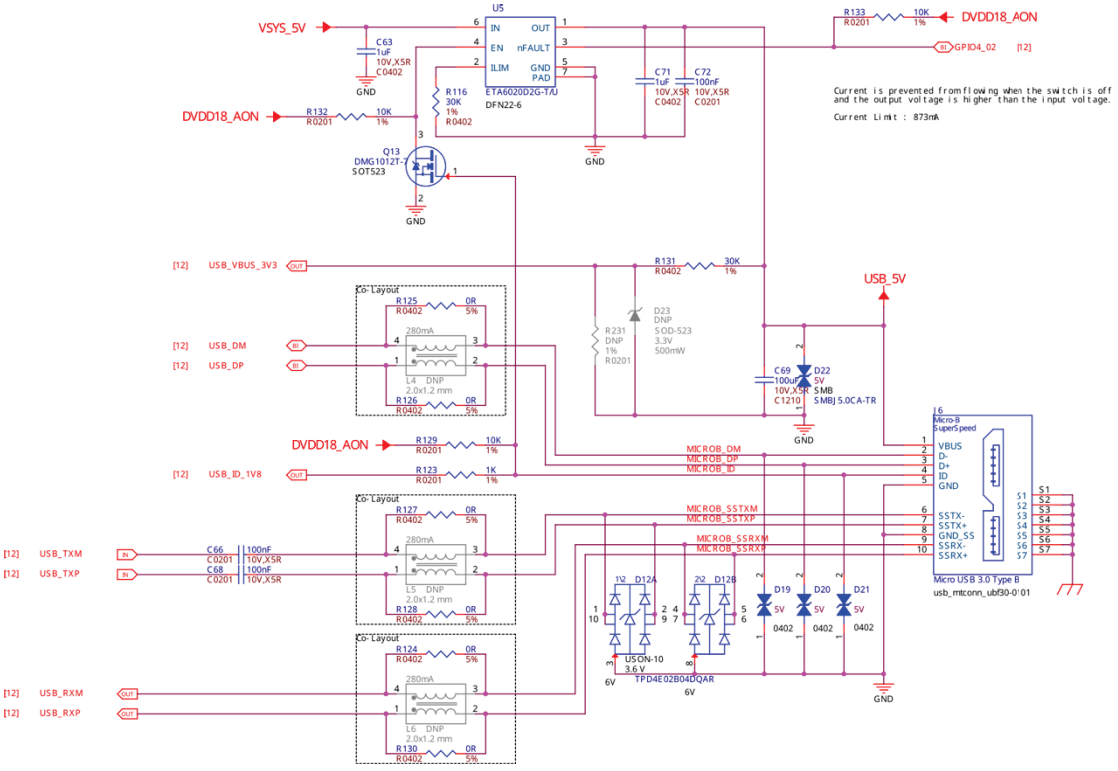


Fig. 195 microUSB 3.0 port¶

P8 & P9 cape header pins

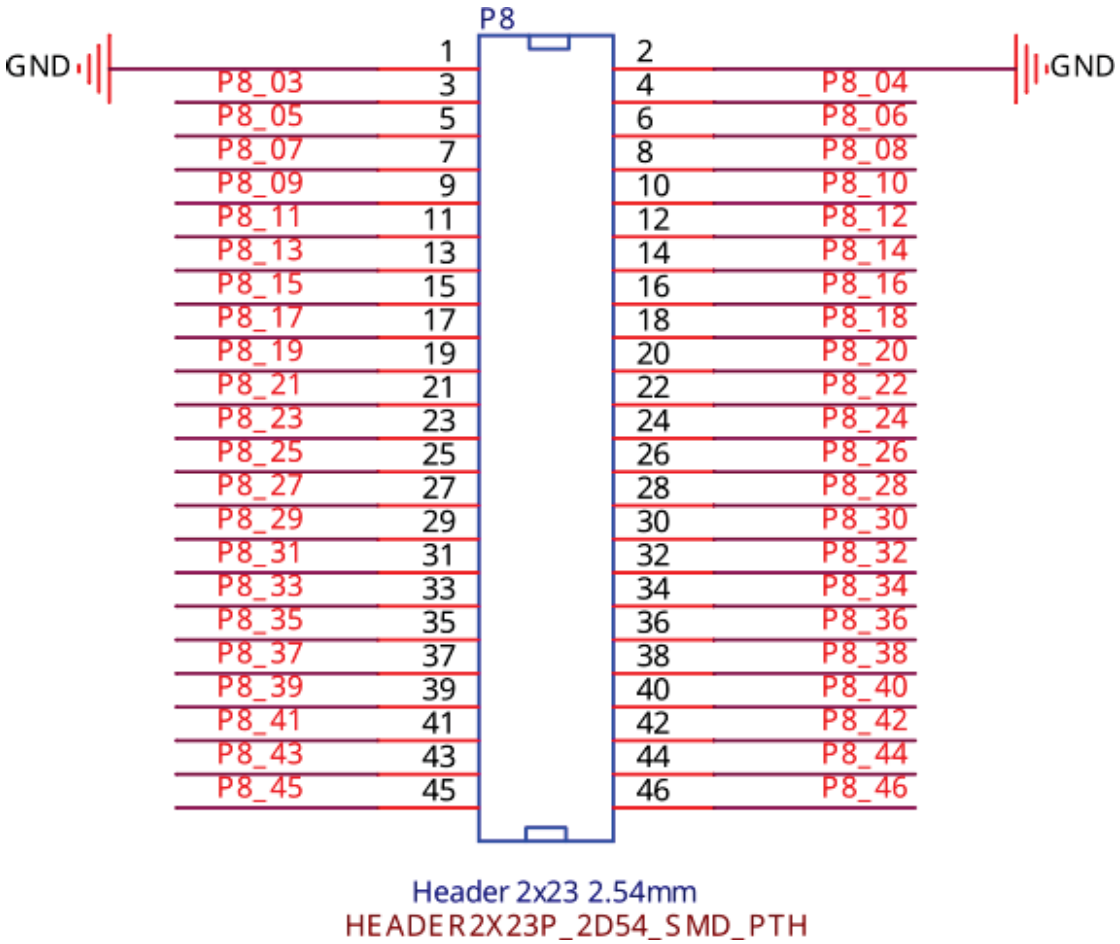


Fig. 196 P8 cape header¶

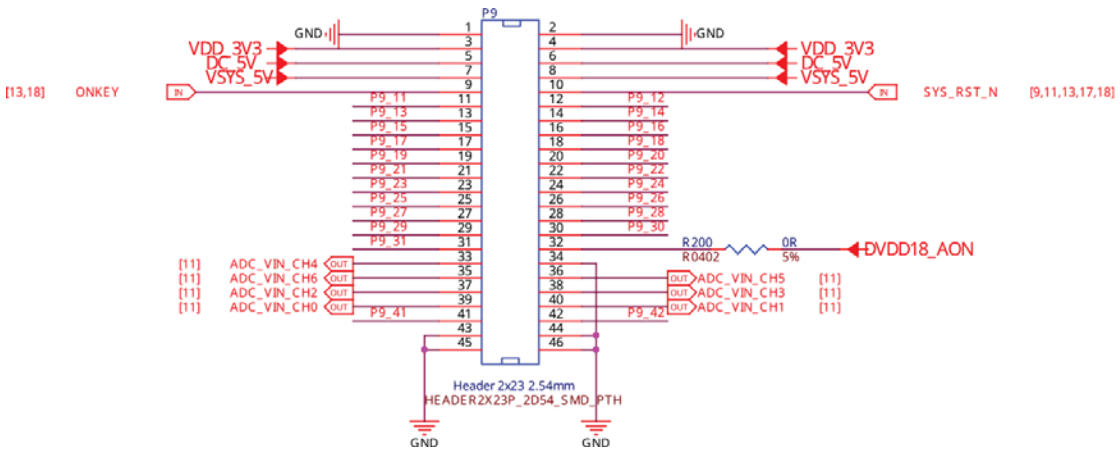
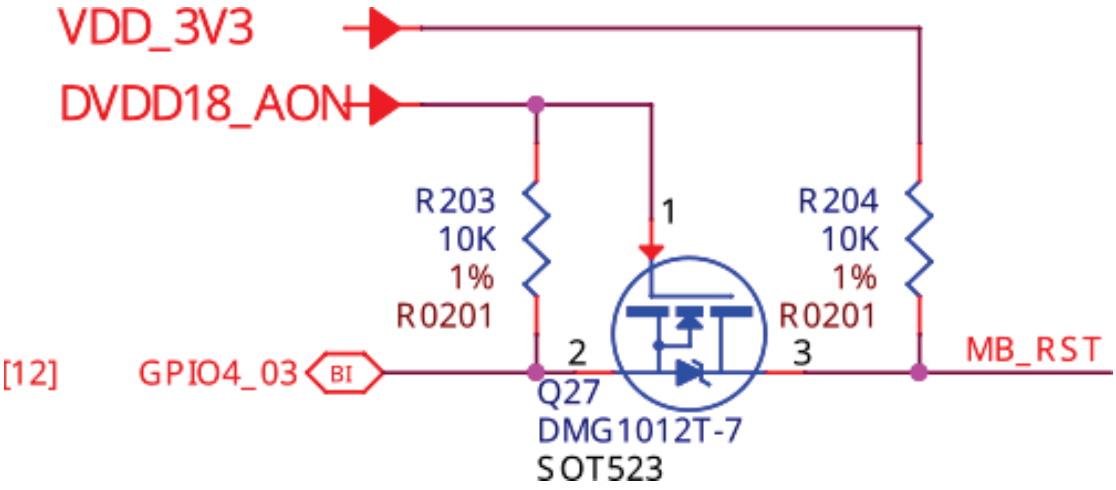


Fig. 197 P9 cape header¶

mikroBUS shuttle connector

P8, P9, and mikroBUS helper circuitry



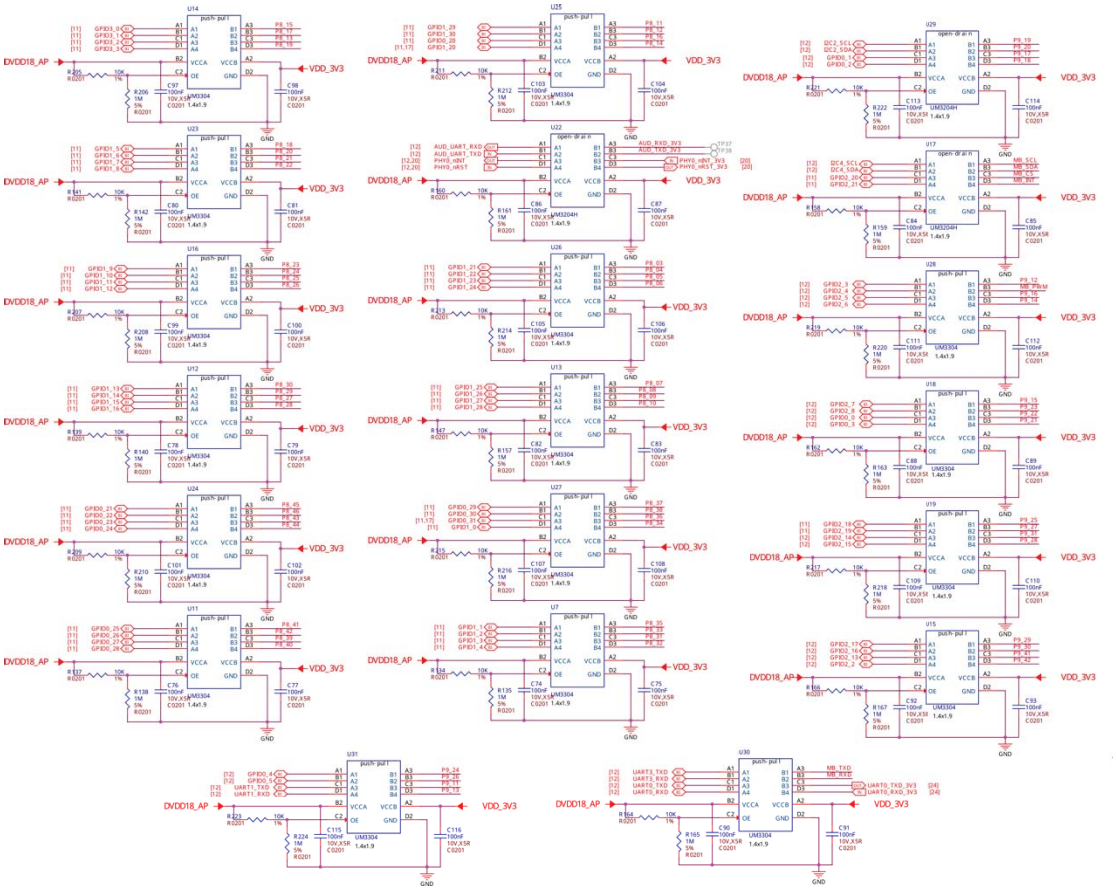
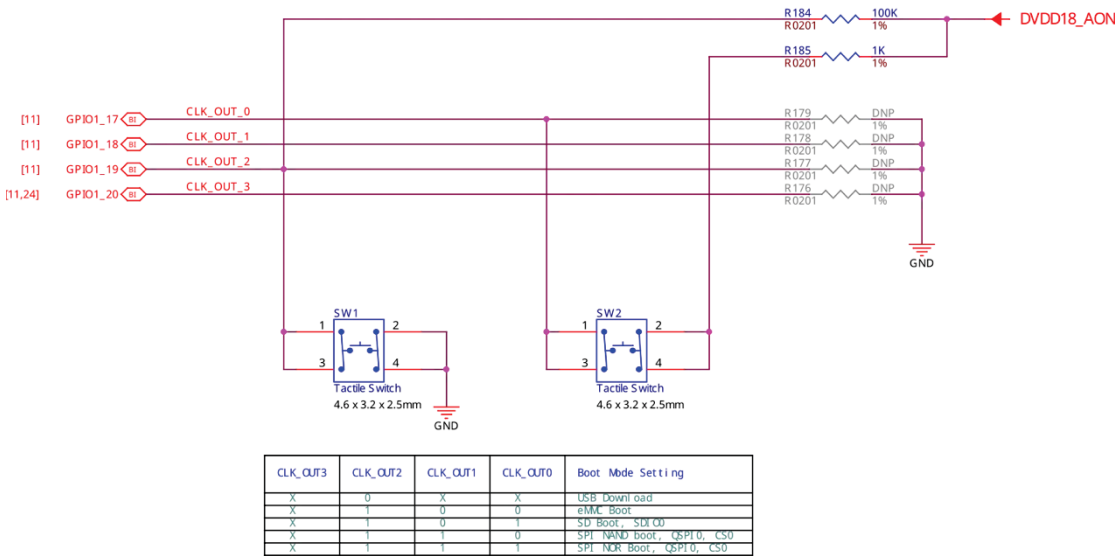


Fig. 198 P8, P9, and mikroBUS level shifters¶

Buttons and LEDs

Boot select buttons



CLK_OUT3	CLK_OUT2	CLK_OUT1	CLK_OUT0	Boot Mode Setting
X	0	X	X	USB Download
X	1	0	0	eMMC Boot
X	1	0	1	SD Boot, SDIO0
X	1	1	0	SPI NAND boot, QSPI0, CS0
X	1	1	1	SPI NOR Boot, QSPI0, CS0

Fig. 199 Boot select buttons¶

User LEDs and Power LED

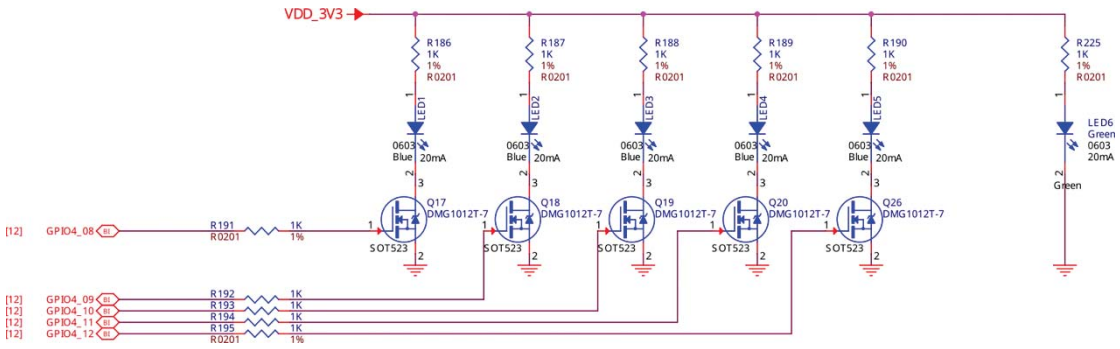


Fig. 200 User LEDs and power LED¶

Power and reset button

Fig. 201 Power and reset button¶

Wired and wireless connectivity

Ethernet

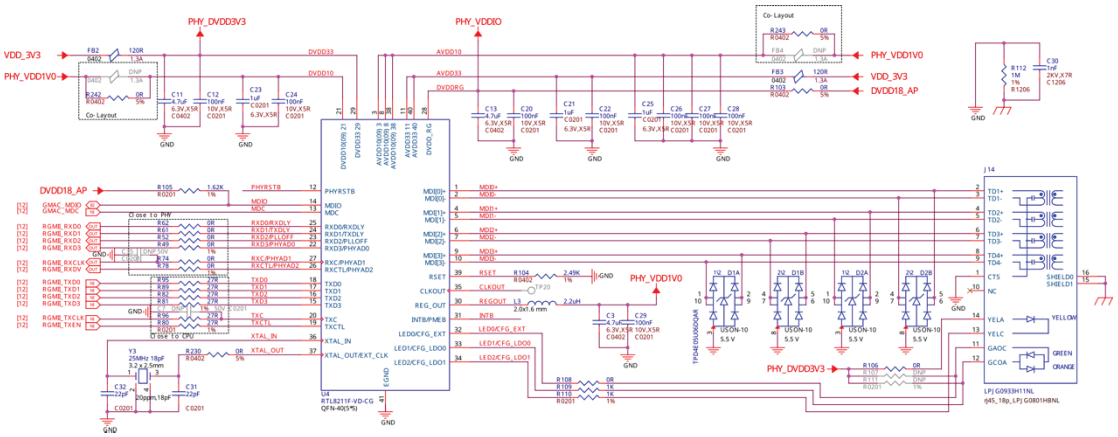


Fig. 202 Ethernet¶

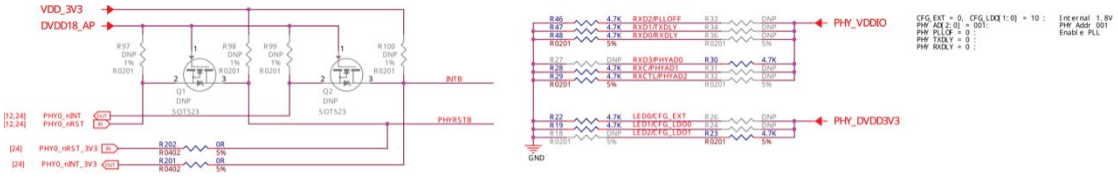


Fig. 203 Ethernet LevelShifter and Strapping¶

WiFi & Bluetooth

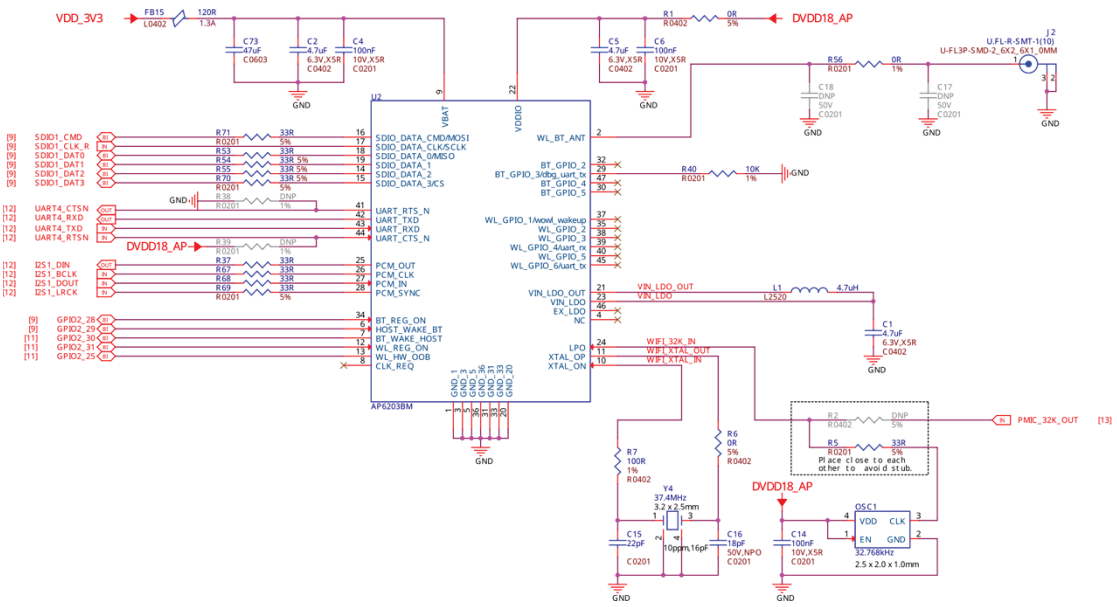


Fig. 204 WiFi and Bluetooth¶

Memory, Media and Data storage

DDR memory

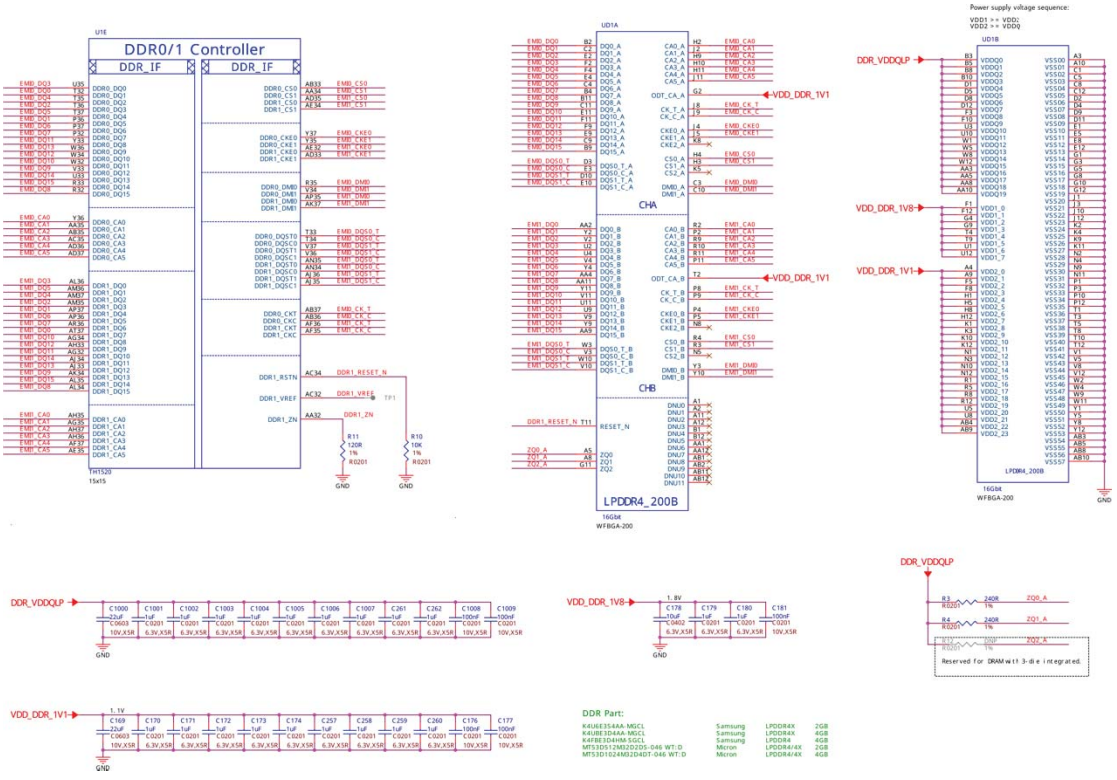


Fig. 205 2GB DDR4 Memory chip1¶

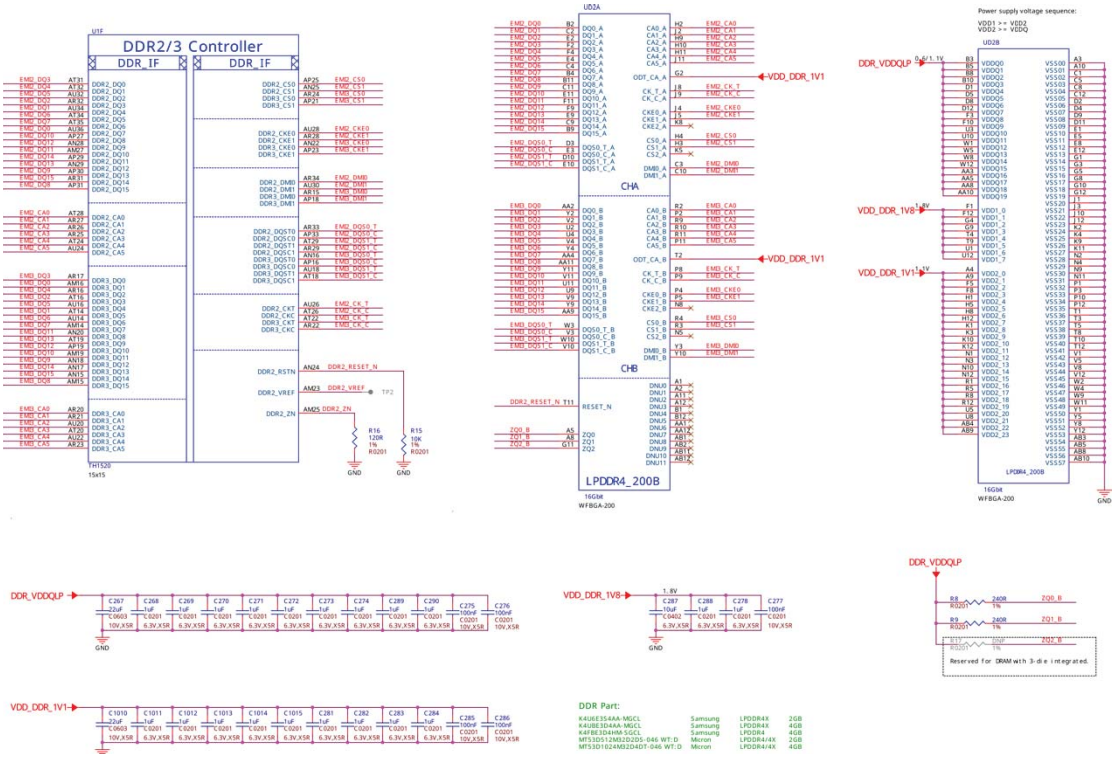


Fig. 206 2GB DDR4 Memory chip2¶

eMMC

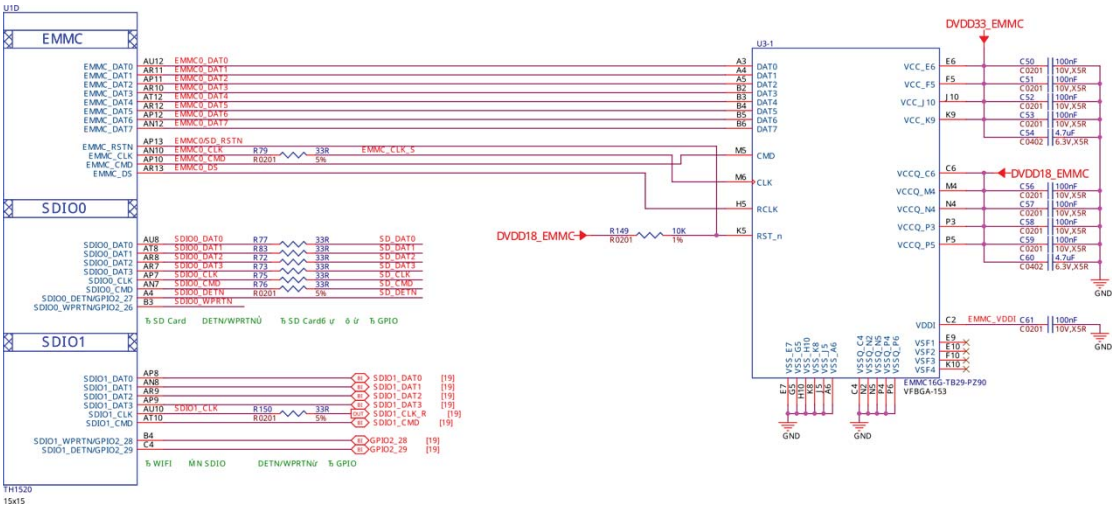


Fig. 207 16GB eMMC¶

microSD

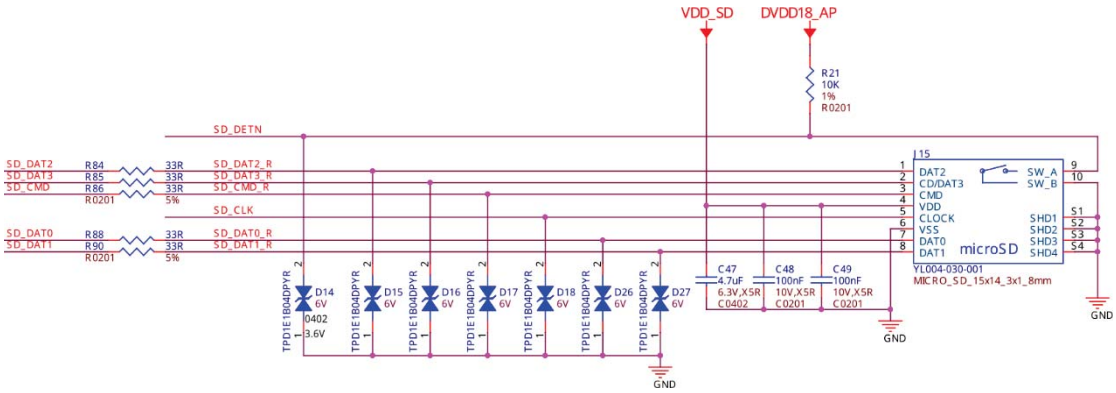


Fig. 208 microSD card connector¶

EEPROM

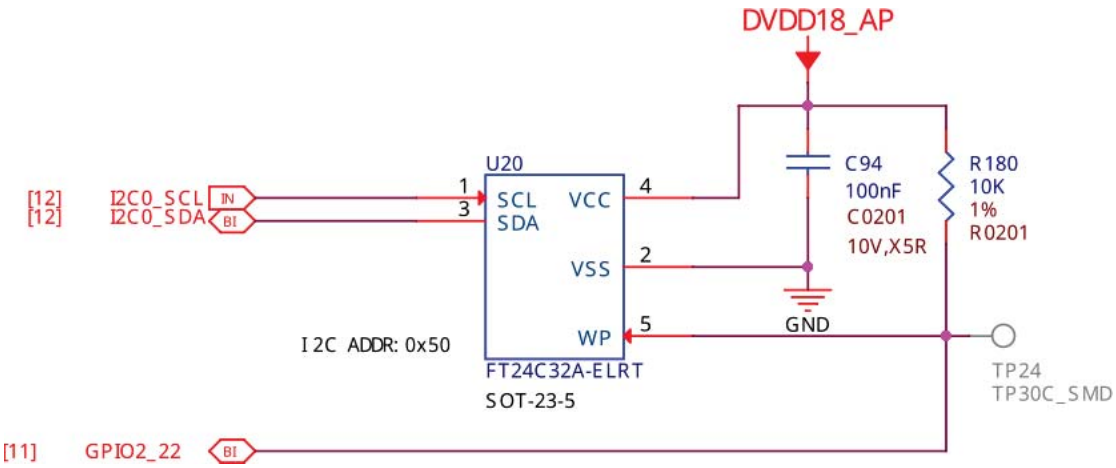


Fig. 209 16GB EEPROM¶

Multimedia I/O

CSI0

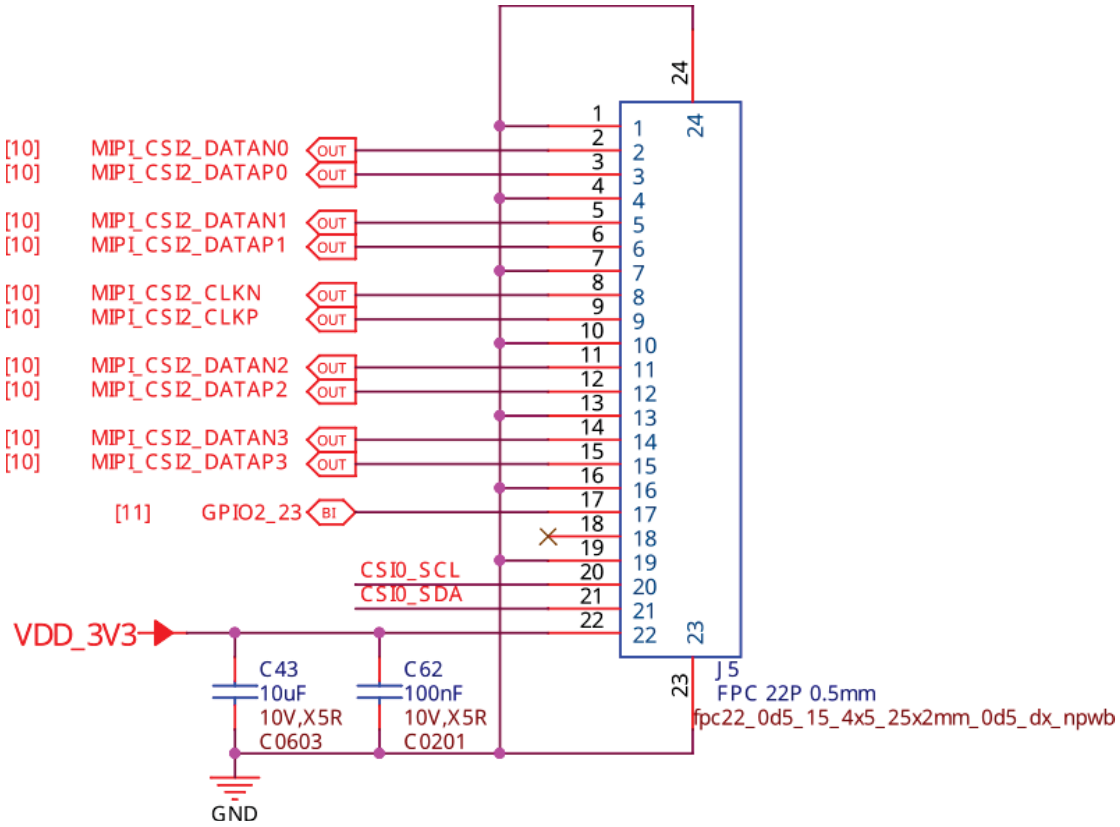


Fig. 210 CSI0 camera interface¶

CSI1

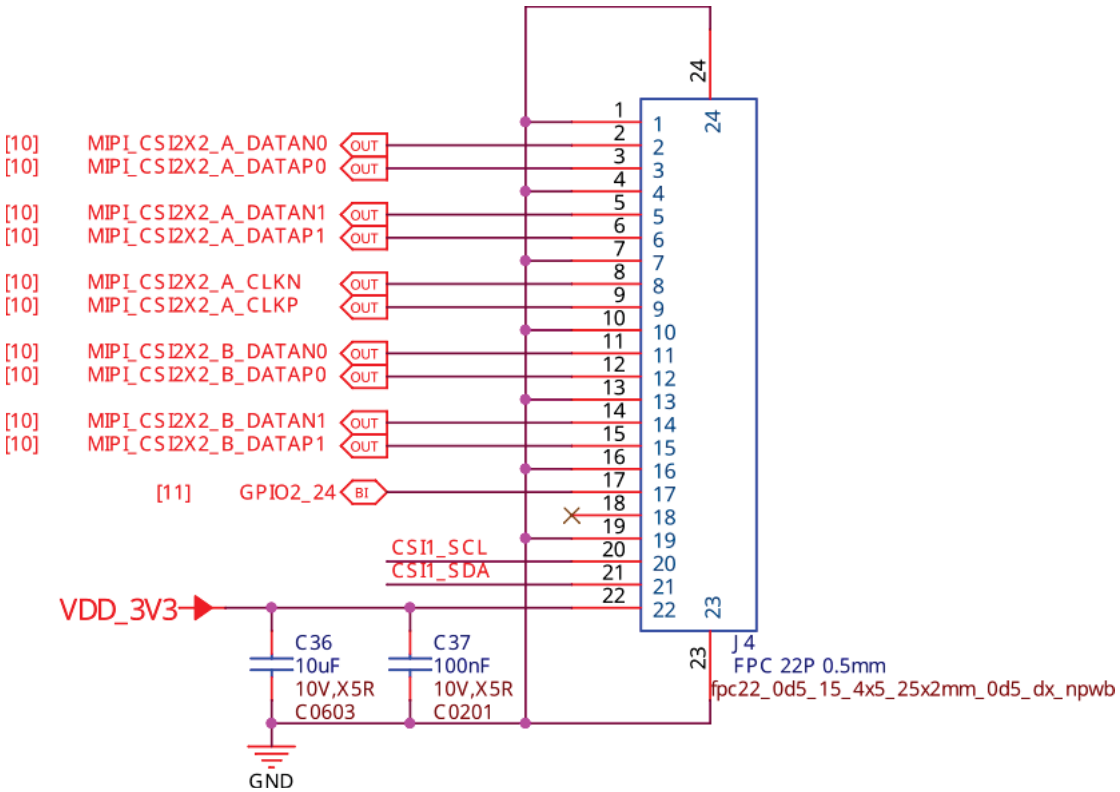


Fig. 211 CSI1 camera interface¶

DSI

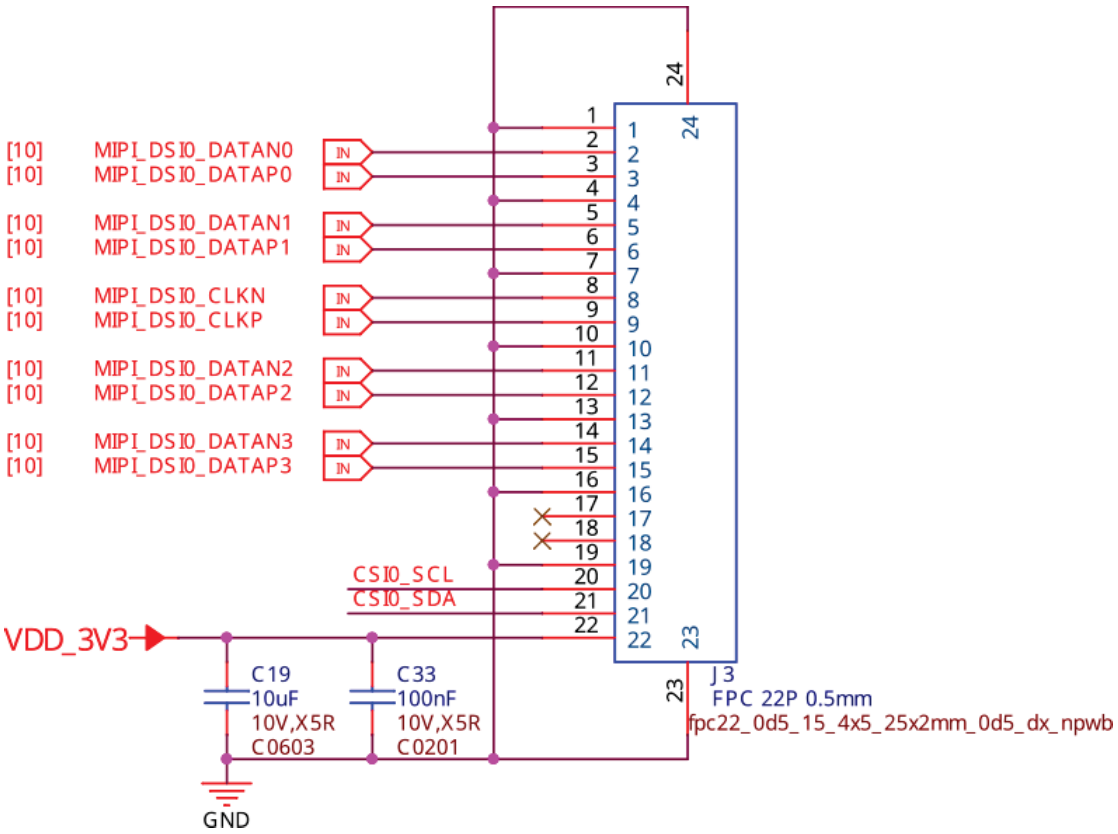


Fig. 212 DSI display interface¶

CSI & DSI level shifter

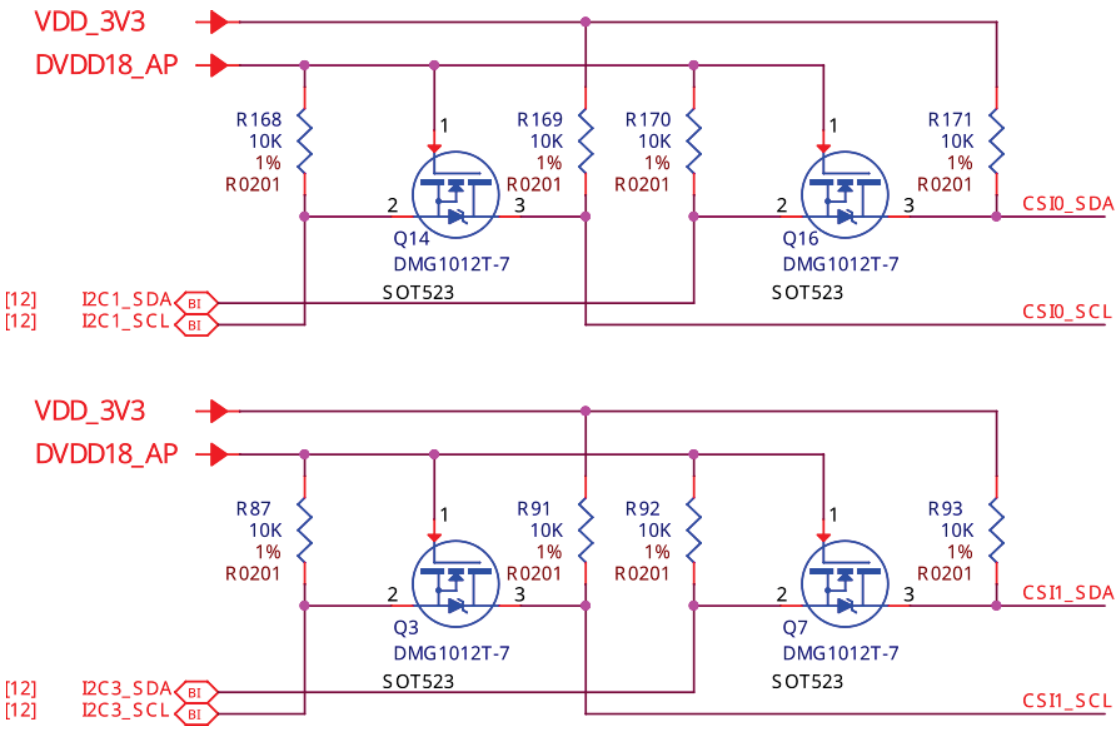


Fig. 213 CSI & DSI level shifter¶

HDMI

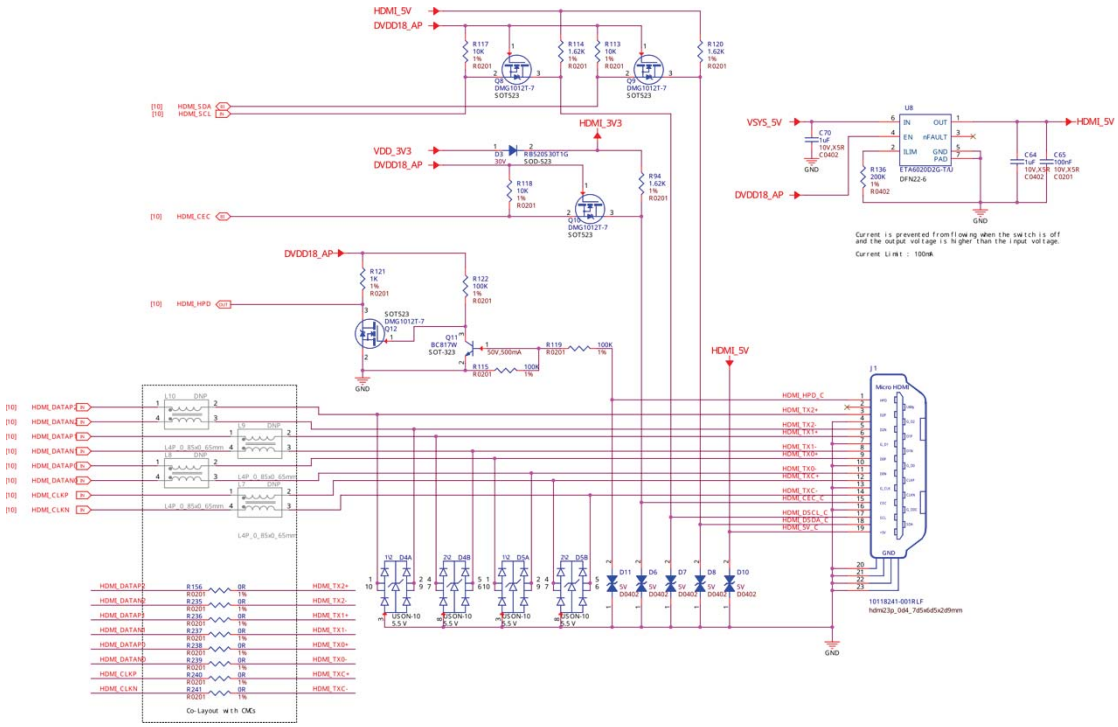
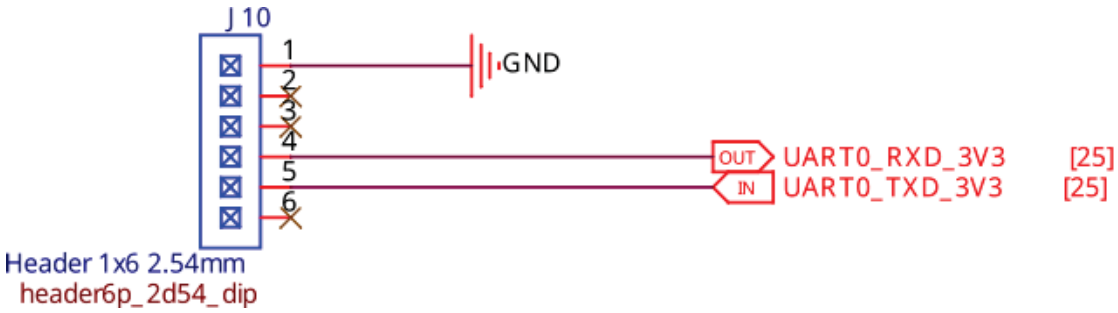


Fig. 214 HDMI display interface¶

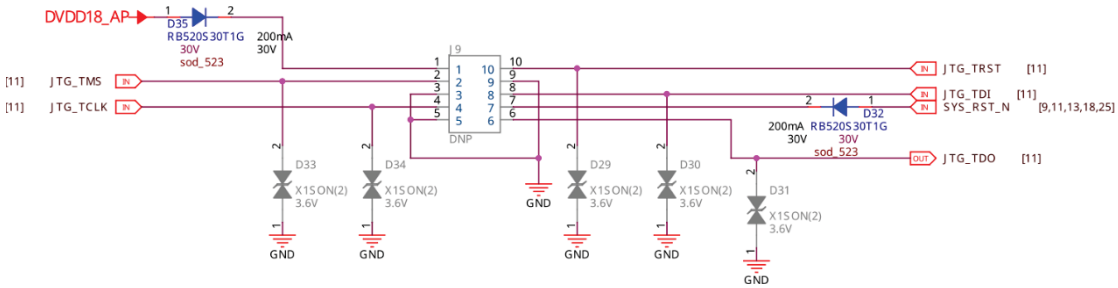
Debug

UART debug port



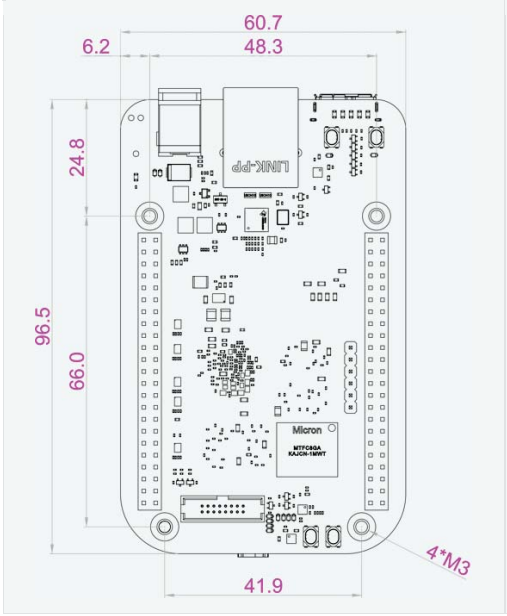
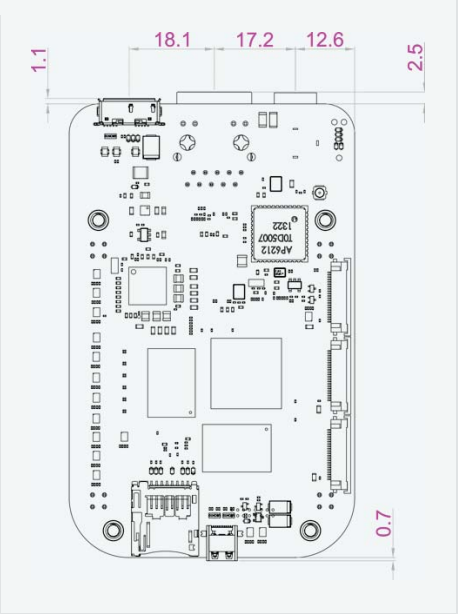
UART Debug port¶

JTAG debug port

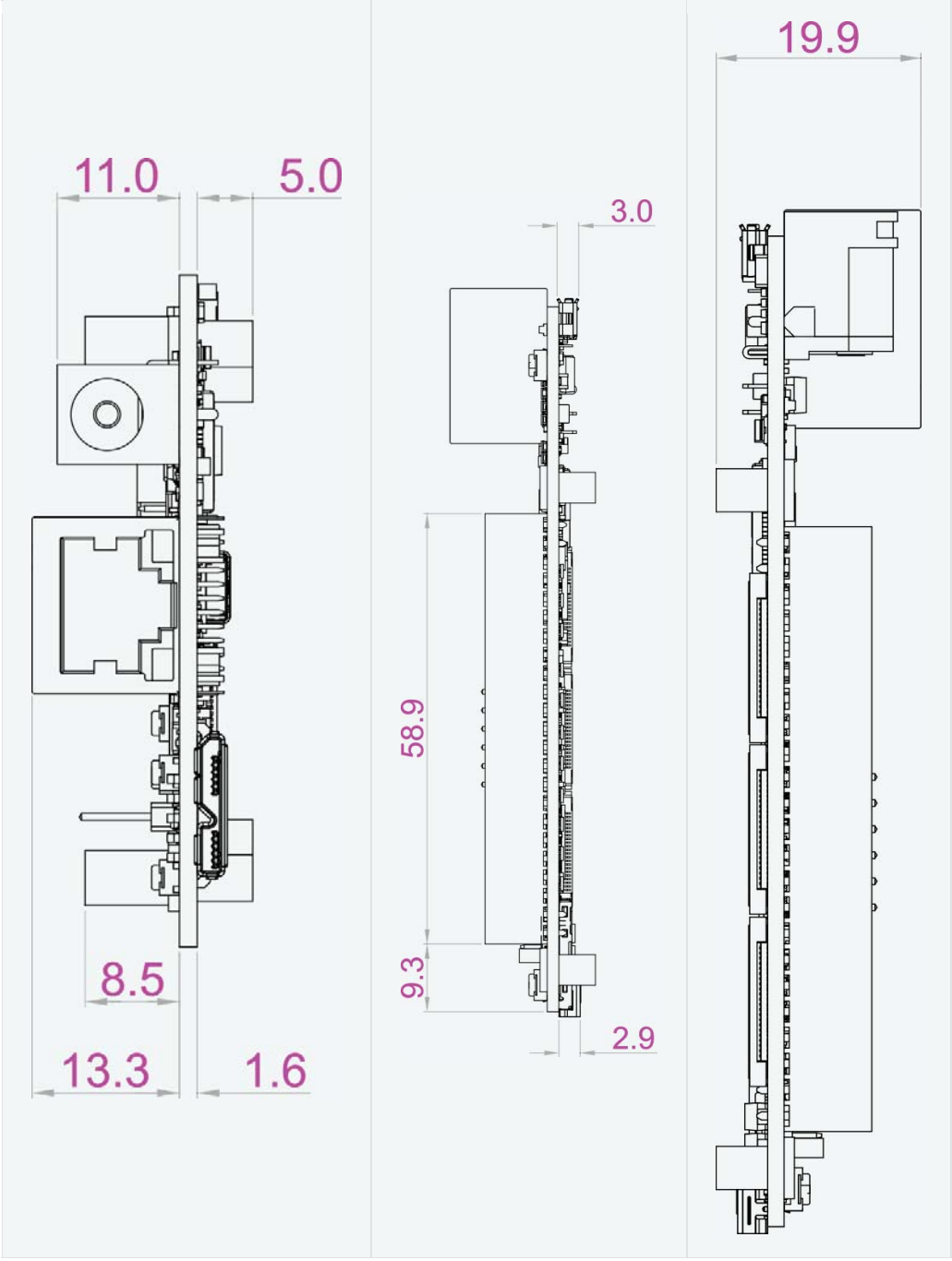


JTAG debug port¶

Mechanical Specifications

Top	Bottom
	

60.7
6.2
48.3
24.8
96.5
66.0
41.9
4*M3
LINK-PP
Micron
18.1
17.2
12.6
1.1
2.5
0.7

Front	Left	Right
		

Dimensions & weight

Parameter	Values
Size	96.5×60.7×19.9mm
Max heigh	21.1mm
PCB Size	96.5x60.5*1.6mm

Dimensions & weight

Parameter	Values
PCB Layers	10 layers
PCB Thickness	1.6mm
RoHS compliant	yes
Gross Weight	128.8g
Net weight	49.7g

Expansion

Cape Headers

The expansion interface on the board is comprised of two headers P8 (46 pin) & P9 (46 pin). All signals on the expansion headers are 3.3V unless otherwise indicated.

Note

Do not connect 5V logic level signals to these pins or the board will be damaged.

Note

DO NOT APPLY VOLTAGE TO ANY I/O PIN WHEN POWER IS NOT SUPPLIED TO THE BOARD. IT WILL DAMAGE THE PROCESSOR AND VOID THE WARRANTY.

NO PINS ARE TO BE DRIVEN UNTIL AFTER THE SYS_RESET LINE GOES HIGH.

Connector P8

The following tables show the pinout of the P8 expansion header. The SW is responsible for setting the default function of each pin. Refer to the processor documentation for more information on these pins and detailed descriptions of all of the pins listed. In some cases there may not be enough signals to complete a group of signals that may be required to implement a total interface.

The column heading is the pin number on the expansion header.

The **GPIO** row is the expected gpio identifier number in the Linux kernel.

Each row includes the gpiochipX and pinY in the format of *X Y*. You can use these values to directly control the GPIO pins with the commands shown below.

```
# to set the GPIO pin state to HIGH
debian@BeagleBone:~$ gpioset X Y=1

# to set the GPIO pin state to LOW
debian@BeagleBone:~$ gpioset X Y=0

For Example:

+---------+----------+
| Pin       | P8.03      |
+==========+==========+
| GPIO      | 1 20       |
+---------+----------+

Use the commands below for controlling this pin (P8.03) where X = 1 and Y = 20

# to set the GPIO pin state to HIGH
debian@BeagleBone:~$ gpioset 1 20=1

# to set the GPIO pin state to LOW
debian@BeagleBone:~$ gpioset 1 20=0
```

The **BALL** row is the pin number on the processor.

The **REG** row is the offset of the control register for the processor pin.

The **MODE #** rows are the mode setting for each pin. Setting each mode to align with the mode column will give that function on that pin.

NOTES:

DO NOT APPLY VOLTAGE TO ANY I/O PIN WHEN POWER IS NOT SUPPLIED TO THE BOARD. IT WILL DAMAGE THE PROCESSOR AND VOID THE WARRANTY.

NO PINS ARE TO BE DRIVEN UNTIL AFTER THE SYS_RESET LINE GOES HIGH.

P8.01-P8.02

P8.01	P8.02
GND	GND

P8.03-P8.05

Pin	P8.03	P8.04	P8.05
Name	GPIO1_21	GPIO1_22	GPIO1_23
BALL	J34	J35	K32
GPIO	1 21	1 22	1 23
REG	GPIO1_21_MUX	GPIO1_22_MUX	GPIO1_23_MUX
Mode 0	GPIO1_21	GPIO1_22	GPIO1_23
MODE 1	~	~	~
MODE 2	ISP0_FL_TRIG	ISP0_FLASH_TRIG	ISP0_PRELIGHT_TRIG
MODE 3	GPIO1_21	GPIO1_22	GPIO1_23
MODE 4	~	~	~
MODE 5	~	~	~

P8.06-P8.09

Pin	P8.06	P8.07	P8.08	P8.09
Name	GPIO1_24	GPIO1_25	GPIO1_26	GPIO1_27

Pin	P8.06	P8.07	P8.08	P8.09
BALL	K33	K34	K35	K36
GPIO	1 24	1 25	1 26	1 27
REG	GPIO1_24_MUX	GPIO1_25_MUX	GPIO1_26_MUX	GPIO1_27_MUX
Mode 0	GPIO1_24	GPIO1_25	GPIO1_26	GPIO1_27
MODE 1	~	~	~	~
MODE 2	ISP0_SHUTTER_TRIG	ISP0_SHUTTER_OPEN	ISP1_FL_TRIG	ISP1_FLASH_TRIG
MODE 3	GPIO1_24	GPIO1_25	~	~
MODE 4	~	~	~	~
MODE 5	~	~	~	~

P8.10-P8.13

Pin	P8.10	P8.11	P8.12	P8.13
Name	GPIO1_28	GPIO1_29	GPIO1_30	GPIO3_2
BALL	K37	L32	L33	C6
GPIO	1 28	1 29	1 30	3 2
REG	GPIO1_28_MUX	GPIO1_29_MUX	GPIO1_30_MUX	GPIO3_2_MUX

Pin	P8.10	P8.11	P8.12	P8.13
MODE 0	GPIO1_28	GPIO1_29	GPIO1_30	GPIO3_2
MODE 1	~	~	~	PWM0
MODE 2	ISP1_PRELIGHT_TRIG	ISP1_SHUTTER_TRIG	ISP1_SHUTTER_OPEN	~
MODE 3	~	~	~	~
MODE 4	~	~	~	~
MODE 5	~	~	~	~

P8.14-P8.16

Pin	P8.14	P8.15	P8.16
Name	CLK_OUT_3	GPIO3_0	GPIO0_20
BALL	E29	A6	F34
GPIO	1 20	3 0	0 20
REG	CLK_OUT_3_MUX	GPIO3_0_MUX	GPIO0_20_MUX
MODE 0	BOOT_SEL3	GPIO3_0	GPIO0_20
MODE 1	CLK_OUT_3	GMAC1_RXD2	UART3_TXD
MODE 2	~	~	UART3_IR_OUT
MODE 3	GPIO1_20	~	~
MODE 4	~	~	~
MODE 5	~	~	~

P8.17-P8.19

Pin	P8.17	P8.18	P8.19
Name	GPIO3_1	GPIO1_5	GPIO3_3
BALL	B6	B34	D6
GPIO	3 1	1 5	3 3
REG	GPIO3_1_MUX	GPIO1_5_MUX	GPIO3_3_MUX
MODE 0	GPIO3_1	GPIO1_5	GPIO3_3
MODE 1	GMAC1_RXD3	~	PWM1
MODE 2	~	~	~
MODE 3	~	~	~
MODE 4	~	DPU_COLOR_16	~
MODE 5	~	DPU1_COLOR_16	~

P8.20-P8.22

Pin	P8.20	P8.21	P8.22
Name	GPIO1_6	GPIO1_7	GPIO1_8
BALL	C34	D34	B35
GPIO	1 6	1 7	1 8
REG	GPIO1_6_MUX	GPIO1_7_MUX	GPIO1_8_MUX
MODE 0	GPIO1_6	GPIO1_7	GPIO1_8
MODE 1	~	QSPI1_SCLK	QSPI1_SSN0
MODE 2	~	~	~
MODE 3	~	~	~
MODE 4	DPU_COLOR_17	DPU_COLOR_18	DPU_COLOR_19
MODE 5	DPU1_COLOR_17	DPU1_COLOR_18	DPU1_COLOR_19

P8.23-P8.26

Pin	P8.23	P8.24	P8.25	P8.26
Name	GPIO1_9	GPIO1_10	GPIO1_11	GPIO1_12
BALL	A36	B36	B37	C36
GPIO	1 9	1 10	1 11	1 12
REG	GPIO1_9_MUX	GPIO1_10_MUX	GPIO1_11_MUX	GPIO1_12_MUX
MODE 0	GPIO1_9	GPIO1_10	GPIO1_11	GPIO1_12
MODE 1	QSPI1_M0_MOSI	QSPI1_M1_MISO	QSPI1_M2_WP	QSPI1_M3_HOLD
MODE 2	~	~	~	~
MODE 3	~	~	~	~
MODE 4	DPU_COLOR_20	DPU_COLOR_21	DPU_COLOR_22	DPU_COLOR_23
MODE 5	DPU1_COLOR_20	DPU1_COLOR_21	DPU1_COLOR_22	DPU1_COLOR_23

P8.27-P8.29

Pin	P8.27	P8.28	P8.29
Name	GPIO1_15	GPIO1_16	GPIO1_14
BALL	D37	E34	D36
GPIO	1 15	1 16	1 14
REG	GPIO1_15_MUX	GPIO1_16_MUX	GPIO1_14_MUX
MODE 0	GPIO1_15	GPIO1_16	GPIO1_14
MODE 1	UART4_CTSN	UART4_RTSN	UART4_RXD

Pin	P8.27	P8.28	P8.29
MODE 2	~	~	~
MODE 3	~	~	~
MODE 4	DPU_VSYNC	DPU_PIXELCLK	DPU_HSYNC
MODE 5	DPU1_VSYNC	DPU1_PIXELCLK	DPU1_HSYNC

P8.30-P8.32

Pin	P8.30	P8.31	P8.32
Name	GPIO1_13	GPIO1_3	GPIO1_4
BALL	D35	D33	A34
GPIO	1 13	1 3	1 4
REG	GPIO1_13_MUX	GPIO1_3_MUX	GPIO1_4_MUX
MODE 0	GPIO1_13	GPIO1_3	GPIO1_4
MODE 1	UART4_TXD	DSP1_JTG_TDO	DSP1_JTG_TCLK
MODE 2	~	~	~
MODE 3	~	~	~
MODE 4	DPU_COLOR_EN	DPU_COLOR_14	DPU_COLOR_15
MODE 5	DPU1_COLOR_EN	DPU1_COLOR_14	DPU1_COLOR_15

P8.33-P8.35

Pin	P8.33	P8.34	P8.35
Name	GPIO1_2	GPIO1_0	GPIO1_1
BALL	C33	E32	A32
GPIO	1 2	1 0	1 1
REG	GPIO1_2_MUX	GPIO1_0_MUX	GPIO1_1_MUX

Pin	P8.33	P8.34	P8.35
MODE 0	GPIO1_2	GPIO1_0	GPIO1_1
MODE 1	DSP1_JTG_TDI	DSP1_JTG_TRST	DSP1_JTG_TMS
MODE 2	~	~	~
MODE 3	~	~	~
MODE 4	DPU_COLOR_13	DPU_COLOR_11	DPU_COLOR_12
MODE 5	DPU1_COLOR_13	DPU1_COLOR_11	DPU1_COLOR_12

P8.36-P8.38

Pin	P8.36	P8.37	P8.38
Name	GPIO0_31	GPIO0_29	GPIO0_30
BALL	D32	B32	C32
GPIO	0 31	0 29	0 30
REG	GPIO0_31_MUX	GPIO0_29_MUX	GPIO0_30_MUX
MODE 0	GPIO0_31	GPIO0_29	GPIO0_30
MODE 1	~	~	~
MODE 2	~	~	~
MODE 3	~	~	~
MODE 4	DPU_COLOR_10	DPU_COLOR_8	DPU_COLOR_9
MODE 5	DPU1_COLOR_10	DPU1_COLOR_8	DPU1_COLOR_9

P8.39-P8.41

Pin	P8.39	P8.40	P8.41
Name	GPIO0_27	GPIO0_28	GPIO0_25
BALL	D31	E31	F30

Pin	P8.39	P8.40	P8.41
GPIO	0 27	0 28	0 25
REG	GPIO0_27_MUX	GPIO0_28_MUX	GPIO0_25_MUX
MODE 0	GPIO0_27	GPIO0_28	GPIO0_25
MODE 1	~	~	DSP0_JTG_TDO
MODE 2	I2C1_SCL	I2C1_SDA	~
MODE 3	~	~	~
MODE 4	DPU_COLOR_6	DPU_COLOR_7	DPU_COLOR_4
MODE 5	DPU1_COLOR_6	DPU1_COLOR_7	DPU1_COLOR_4

P8.42-P8.44

Pin	P8.42	P8.43	P8.44
Name	GPIO0_26	GPIO0_23	GPIO0_24
BALL	C31	C30	D30
GPIO	0 26	0 23	0 24
REG	GPIO0_26_MUX	GPIO0_23_MUX	GPIO0_24_MUX
MODE 0	GPIO0_26	GPIO0_23	GPIO0_24
MODE 1	DSP0_JTG_TCLK	DSP0_JTG_TMS	DSP0_JTG_TDI
MODE 2	~	I2C4_SDA	QSPI1_SSN1
MODE 3	~	~	~
MODE 4	DPU_COLOR_5	DPU_COLOR_2	DPU_COLOR_3
MODE 5	DPU1_COLOR_5	DPU1_COLOR_2	DPU1_COLOR_3

P8.45-P8.46

Pin	P8.45	P8.46
Name	GPIO0_21	GPIO0_22
BALL	F36	D29
GPIO	0 21	0 22
REG	GPIO0_21_MUX	GPIO0_22_MUX
MODE 0	GPIO0_21	GPIO0_22
MODE 1	UART3_RXD	DSP0_JTG_TRST
MODE 2	UART3_IR_IN	I2C4_SCL
MODE 3	~	~
MODE 4	DPU_COLOR_0	DPU_COLOR_1
MODE 5	DPU1_COLOR_0	DPU1_COLOR_1

Connector P9

The following tables show the pinout of the P9 expansion header. The SW is responsible for setting the default function of each pin. Refer to the processor documentation for more information on these pins and detailed descriptions of all of the pins listed. In some cases there may not be enough signals to complete a group of signals that may be required to implement a total interface.

The column heading is the pin number on the expansion header.

The GPIO row is the expected gpio identifier number in the Linux kernel.

Each row includes the gpiochipX and pinY in the format of *X Y*. You can use these values to direcly control the GPIO pins with the commands shown below.

```
# to set the GPIO pin state to HIGH
debian@BeagleBone:~$ gpioset X Y=1
```

```
# to set the GPIO pin state to LOW
debian@BeagleBone:~$ gpioset X Y=0

For Example:

+---------+----------+
| Pin       | P9.11      |
+=========+==========+
| GPIO      | 1 1        |
+---------+----------+

Use the commands below for controlling this pin (P9.11) where X = 1 and Y = 1

# to set the GPIO pin state to HIGH
debian@BeagleBone:~$ gpioset 1 20=1

# to set the GPIO pin state to LOW
debian@BeagleBone:~$ gpioset 1 20=0
```

The **BALL** row is the pin number on the processor.

The **REG** row is the offset of the control register for the processor pin.

The **MODE #** rows are the mode setting for each pin. Setting each mode to align with the mode column will give that function on that pin.

If included, the **2nd BALL** row is the pin number on the processor for a second processor pin connected to the same pin on the expansion header. Similarly, all row headings starting with **2nd** refer to data for this second processor pin.

NOTES:

DO NOT APPLY VOLTAGE TO ANY I/O PIN WHEN POWER IS NOT SUPPLIED TO THE BOARD. IT WILL DAMAGE THE PROCESSOR AND VOID THE WARRANTY.

NO PINS ARE TO BE DRIVEN UNTIL AFTER THE SYS_RESET LINE GOES HIGH.

P9.01-P9.05

P9.01	P9.02	P9.03	P9.04	P9.05
GND	GND	VOUT_3V3	VOUT_3V3	VIN

P9.06-P9.10

P9.06	P9.07	P9.08	P9.09	P9.10
VIN	VOUT_SYS	VOUT_SYS	ONKEY#	RESET#

P9.11-P9.13

Pin	P9.11	P9.12	P9.13
Name	UART1_TXD	QSPI0_CSN0	UART1_RXD
BALL	M32	H1	M33
GPIO	0 10	2 3	0 11
REG	UART1_TXD_MUX	QSPI0_CSN0_MUX	UART1_RXD_MUX
MODE 0	UART1_TXD	QSPI0_SSN0	UART1_RXD
MODE 1	~	PWM1	~
MODE 2	~	I2S_SDA1	~
MODE 3	GPIO0_10	GPIO2_3	GPIO0_11
MODE 4	~	~	~
MODE 5	~	~	~

P9.14-P9.16

Pin	P9.14	P9.15	P9.16
Name	QSPI0_D1_MISO	QSPI0_D2_WP	QSPI0_D0_MOSI
BALL	K3	K2	J3
GPIO	2 6	2 7	2 5
REG	QSPI0_D1_MISO_MUX	QSPI0_D2_WP_MUX	QSPI0_D0_MOSI_MUX

Pin	P9.14	P9.15	P9.16
MODE 0	QSPI0_M1_MISO	QSPI0_M2_WP	QSPI0_M0_MOSI
MODE 1	PWM4	PWM5	PWM3
MODE 2	I2S_MCLK	I2S_SCLK	I2S_SDA3
MODE 3	GPIO2_6	GPIO2_7	GPIO2_5
MODE 4	~	~	~
MODE 5	~	~	~

P9.17-P9.19

Pin	P9.17	P9.18	P9.19
Name	QSPI1_CSN0	QSPI1_D0_MOSI	I2C2_SCL
BALL	H32	G35	G4
GPIO	0 1	0 2	2 9
REG	QSPI1_CSN0_MUX	QSPI1_D0_MOSI_MUX	I2C2_SCL_MUX
MODE 0	QSPI1_SSN0	QSPI1_M0_MOSI	I2C2_SCL
MODE 1	~	ISO7816_CVCC_EN	UART2_TXD
MODE 2	I2S_MCLK	I2C5_SDA	~
MODE 3	GPIO0_1	GPIO0_2	GPIO2_9
MODE 4	EFUSE_SPI_NSS	EFUSE_SPI_SI	~
MODE 5	~	~	~

P9.20-P9.22

Pin	P9.20	P9.21	P9.22
Name	I2C2_SDA	QSPI1_D1_MISO	QSPI1_SCLK
BALL	G3	G34	H34

Pin	P9.20	P9.21	P9.22
GPIO	2 10	0 3	0 0
REG	I2C2_SDA_MUX	QSPI1_D1_MISO_MUX	QSPI1_SCLK_MUX
MODE 0	I2C2_SDA	QSPI1_M1_MISO	QSPI1_SCLK
MODE 1	UART2_RXD	ISO7816_CLK	ISO7816_DET
MODE 2	~	~	~
MODE 3	GPIO2_10	GPIO0_3	GPIO0_0
MODE 4	~	EFUSE_SPI_SO	EFUSE_SPI_CLK
MODE 5	~	~	~

P9.23-P9.25

Pin	P9.23	P9.24	P9.25
Name	QSPI0_D3_HOLD	QSPI1_D2_WP	GPIO2_18
BALL	K1	G33	F5
GPIO	2 8	0 4	2 18
REG	QSPI0_D3_HOLD_MUX	QSPI1_D2_WP_MUX	GPIO2_18_MUX
MODE 0	QSPI0_M3_HOLD	QSPI1_M2_WP	GPIO2_18
MODE 1	~	ISO7816_RST	GMAC1_TX_CLK
MODE 2	I2S_WS	UART5_TXD	~
MODE 3	GPIO2_8	GPIO0_4	~
MODE 4	~	EFUSE_BUSY	~
MODE 5	~	~	~

P9.26-P9.28

Pin	P9.26	P9.27	P9.28
Name	QSPI1_D3_HOLD	GPIO2_19	SPI_CSN
BALL	F37	E4	E3
GPIO	0 5	2 19	2 15
REG	QSPI1_D3_HOLD_MUX	GPIO2_19_MUX	SPI_CSN_MUX
MODE 0	QSPI1_M3_HOLD	GPIO2_19	SPI_SSN0
MODE 1	ISO7816_DAT	GMAC1_RX_CLK	UART2_RXD
MODE 2	UART5_RXD	~	UART2_IR_IN
MODE 3	GPIO0_5	~	GPIO2_15
MODE 4	~	~	~
MODE 5	~	~	~

P9.29-P9.31

Pin	P9.29	P9.30	P9.31
Name	SPI_MISO	SPI_MOSI	SPI_SCLK
BALL	F1	F2	D3
GPIO	2 17	2 16	2 14
REG	SPI_MISO_MUX	SPI_MOSI_MUX	SPI_SCLK_MUX
MODE 0	SPI_MISO	SPI_MOSI	SPI_SCLK
MODE 1	~	~	UART2_TXD
MODE 2	~	~	UART2_IR_OUT
MODE 3	GPIO2_17	GPIO2_16	GPIO2_14
MODE 4	~	~	~
MODE 5	~	~	~

P9.32-P9.40

P9.32	P9.34
VDD_ADC	GND

P9.33	P9.35	P9.36	P9.37	P9.38	P9.39	P9.40
ADC_VIN_CH4	ADC_VIN_CH6	ADC_VIN_CH5	ADC_VIN_CH2	ADC_VIN_CH3	ADC_VIN_CH0	ADC_VIN_CH1

P9.41-P8.42

Pin	P9.41	P9.42
Name	GPIO2_13	QSPI0_SCLK
BALL	D2	H3
GPIO	2 13	2 2
REG	GPIO2_13_MUX	QSPI0_SCLK_MUX
MODE 0	GPIO2_13	QSPI0_SCLK
MODE 1	SPI_SSN1	PWM0
MODE 2	~	I2S_SDA0
MODE 3	~	GPIO2_2
MODE 4	~	~
MODE 5	~	~

P9.43-P9.46

P9.43	P9.44	P9.45	P9.46
GND	GND	GND	GND

mikroBUS

Pin	mikroBUS port		Pin
ADC_VIN_CH7	AN	PWM	QSPI0_CSN1 (MODE1:PWM2)

Pin	mikroBUS port		Pin
AUDIO_PA3 (MODE3:GPIO4_3)	RST	INT	GPIO2_21 (MODE0:GPIO2_21)
GPIO2_20 (MODE0:GPIO2_20)	CS	RX	UART3_RXD (MODE1:UART3_RXD)
SPI_SCLK (MODE0:SPI_SCLK)	SCK	TX	UART3_TXD (MODE1:UART3_TXD)
SPI_MISO (MODE0:SPI_MISO)	MISO	SCL	GPIO0_18 (MODE1:I2C4_SCL)
SPI_MOSI (MODE0:SPI_MOSI)	MOSI	SDA	GPIO0_19 (MODE1:I2C4_SDA)
3.3V supply	3V3	5V	5V supply
Ground	GND	GND	Ground

FCC statement:

When using the product, maintain a distance of 20cm from the body to ensure compliance with RF exposure requirements.

This device complies with part 15 of the FCC rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

NOTE: The manufacturer is not responsible for any radio or TV interference caused by unauthorized modifications or changes to this equipment. Such modifications or changes could void the user's authority to operate the equipment.

NOTE: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Caution: Any changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate this equipment.

ORIGINAL EQUIPMENT MANUFACTURER (OEM) NOTES

The OEM must certify the final end product to comply with unintentional radiators (FCC Sections 15.107 and 15.109) before declaring compliance of the final product to Part 15 of the FCC rules and regulations. Integration into devices that are directly or indirectly connected to AC lines must add with Class II Permissive Change.

The OEM must comply with the FCC labeling requirements. If the module's label is not visible when installed, then an additional permanent label must be applied on the outside of the finished product which states: "Contains transmitter module FCC ID: Z4T-BVA23060V1.

Additionally, the following statement should be included on the label and in the final product's user manual: "This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interferences, and (2) this device must accept any interference received, including interference that may cause undesired operation."

The module is allowed to be installed in mobile and portable applications.

A module or modules can only be used without additional authorizations if they have been tested and granted under the same intended end-use operational conditions, including simultaneous transmission operations. When they have not been tested and granted in this manner, additional testing and/or FCC application filing may be required. The most straightforward approach to address additional testing conditions is to have the grantee responsible for the certification of at least one of the modules submit a permissive change application. When having a module grantee file a permissive change is not practical or feasible, the following guidance provides some additional options for host manufacturers. Integrations using modules where additional testing and/or FCC application filing(s) may be required are: (A) a module used in devices requiring additional RF exposure compliance information (e.g., MPE evaluation or SAR testing); (B) limited and/or split modules not meeting all of the module requirements; and (C) simultaneous transmissions for independent collocated transmitters not previously granted together.

This Module is full modular approval, it is limited to OEM installation ONLY. Integration into devices that are directly or indirectly connected to AC lines must add with Class II Permissive Change. (OEM) Integrator has to assure compliance of the entire end product include the integrated Module. Additional measurements (15B) and/or equipment authorizations (e.g. Verification) may need to be addressed depending on co-location or simultaneous transmission issues if applicable. (OEM) Integrator is reminded to assure that these installation instructions will not be made available to the end user。

CE statement:

5150–5250 MHz can be used indoor only.

	AT	BE	BG	CH	CY	CY	DE	DK
	EE	EL	ES	FI	FR	HR	HU	IE
	IS	IT	LI	LT	LU	LV	MT	NL
	PL	PT	RO	SE	SI	SK	TR	

Frequency bands and power

	Bands	Operation Frequency	Max. Power
Bluetooth	2.4GHz	2402-2480 MHz	EIRP 9.67 dBm
Wi-Fi	2.4GHz	2412-2472MHz	EIRP 14.06 dBm
	5GHz	5180-5240MHz	EIRP 12.02 dBm
		5745-5825MHz	EIRP 13.66 dBm