

Introduction

This User Guide (UG) for the OSD3358-SM-RED platform is intended to serve as a resource to find information regarding the design and usage of the hardware and software aspects of the board. It will cover topics ranging from getting started with the platform to using different sensors and peripherals that are present on board.

This document will be updated as required to improve or add information. Please make sure to look for updates and sign up for document change notifications on [Octavo Systems website](https://octavosystems.com) to get up to date info.

This user guide is relevant for both OSD3358-SM-RED revision 3 and revision 4. The board revision is indicated on the board at the designated location shown below:



Some relevant resources for support and development are given below:

- OSD3358-SM-RED board design files – v4:
<https://octavosystems.com/files/osd3358-sm-red-eagle-files/>
- OSD3358-SM-RED board design files – v3:
<https://octavosystems.com/files/osd3358-sm-red-v3-eagle-files/>
- Sign up for OSD3358-SM-RED latest software updates:
<https://octavosystems.com/doc-change-signup/>
- OSD3358-SM datasheet:
<https://octavosystems.com/docs/osd335x-sm-datasheet/>
- OSD3358-SM RED platform getting started guide:
<https://octavosystems.com/docs/osd3358-sm-red-quick-start-guide/>
- OSD3358-SM Application notes:
 - a. OSD335x-SM layout guide:
https://octavosystems.com/app_notes/osd335x-sm-layout-guide/
 - b. OSD335x Family Pin Assignments Compared to AM335x:
https://octavosystems.com/app_notes/osd335x-family-pin-assignments/

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



Revision History

Revision	Details	Date	Author
1.0	Initial Release	09-19-2017	Neeraj Dantu
2.0	Updates for Rev 4 hardware changes	11-26-2021	Neeraj Dantu

Notice: The information provided within this document is for informational use only. Octavo Systems provides no guarantees or warranty to the information contained.

Contents

1	Powering the OSD3358-SM-RED.....	6
1.1	Power up procedure.....	7
2	OSD3358-SM-RED Use Cases.....	10
2.1	Standalone Boot-Up	10
2.2	Bootting as USB Client Connected to a Computer/Laptop.....	14
2.3	Boot Messages Through UART	17
2.4	User LEDs	21
2.5	Powering Down / Power Cycling.....	21
3	Overview of OSD3358-SM-RED	22
3.1	Specification	22
3.2	Processor (U1).....	25
3.3	PC USB Interface (X1)	26
3.4	Serial Debug Interface (J3).....	27
3.5	EMMC (U7)	28
3.6	Cape Headers (J5 & J6).....	29
3.7	JTAG (J4)	30
3.8	Micro-SD Card Slot (X3)	31
3.9	HDMI Interface (X6)	32
3.10	Gigabit Ethernet (X4)	33
3.11	Four (4) Port USB Hub (X4 & X5).....	34
3.12	Inertial Measurement Unit (IMU) (U23) - V3	35
3.13	Inertial Measurement Unit (IMU) (U23) - V4	35
3.14	Temperature sensor hub (U17)	36
3.15	Pressure sensor (U22) - V3	37
3.16	Pressure sensor (U22) - V4	37
3.17	Trusted Platform Module (TPM) (U16)	38
3.18	NOR Flash (U21).....	39
4	Detailed Hardware Design	40
4.1	Power Management	40
4.2	Additional Functions of the PMIC.....	42

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



4.3	Input Power.....	42
4.4	PMIC-AM3358 Interface	45
4.5	Clamping Circuit	48
4.6	Clock Input to the Processor	49
4.6.1	OSC0:.....	49
4.6.2	OSC1:.....	49
4.7	Boot Configuration	51
4.8	UART0 Interface	53
4.9	eMMC	54
4.10	MicroSD Card Slot.....	55
4.11	JTAG.....	56
4.12	ADC	57
4.13	USB Client Port.....	58
4.14	Four (4) port USB 2.0 Hub	59
4.15	Ethernet.....	60
4.16	HDMI Interface	61
4.17	9-Axis IMU.....	62
4.18	Temperature Sensor Hub	63
4.19	Barometer	64
4.20	Trusted Platform Module (TPM)	65
4.21	EEPROM	66
4.22	Expansion Headers.....	66
4.23	On-board Connectors and Jumpers	67
4.24	Test Points.....	69
5	Board Images and Software Features	70
5.1	Boot Sequence	70
5.2	Device Tree.....	71
5.2.1	Device Tree Overlays	71
5.3	Using On-Board Sensors - V3	72
5.3.1	Accessing Peripherals using the Terminal	73
5.3.2	Using Peripheral Library to Write Applications	77
5.4	Using On-Board Sensors - V4	78

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021

5.4.1	Temperature Sensor - TMP468 (U17)	78
5.4.2	Barometer - LPS22HB (U22).....	79
5.4.3	Accelerometer - ICM-20948 (U23).....	80
6	Cape Support	81
7	Mechanicals	82
8	Support and Troubleshooting.....	84
9	Reference Documents	85

1 Powering the OSD3358-SM-RED

Figure 1 shows the OSD3358-SM-RED and its power inputs. The board can be powered through any of three inputs:

1. Micro-USB cable connected to the micro-USB port
2. 5V AC adapter connected to the barrel jack
3. Single cell Li-Ion or Li-Polymer battery connected to the battery header

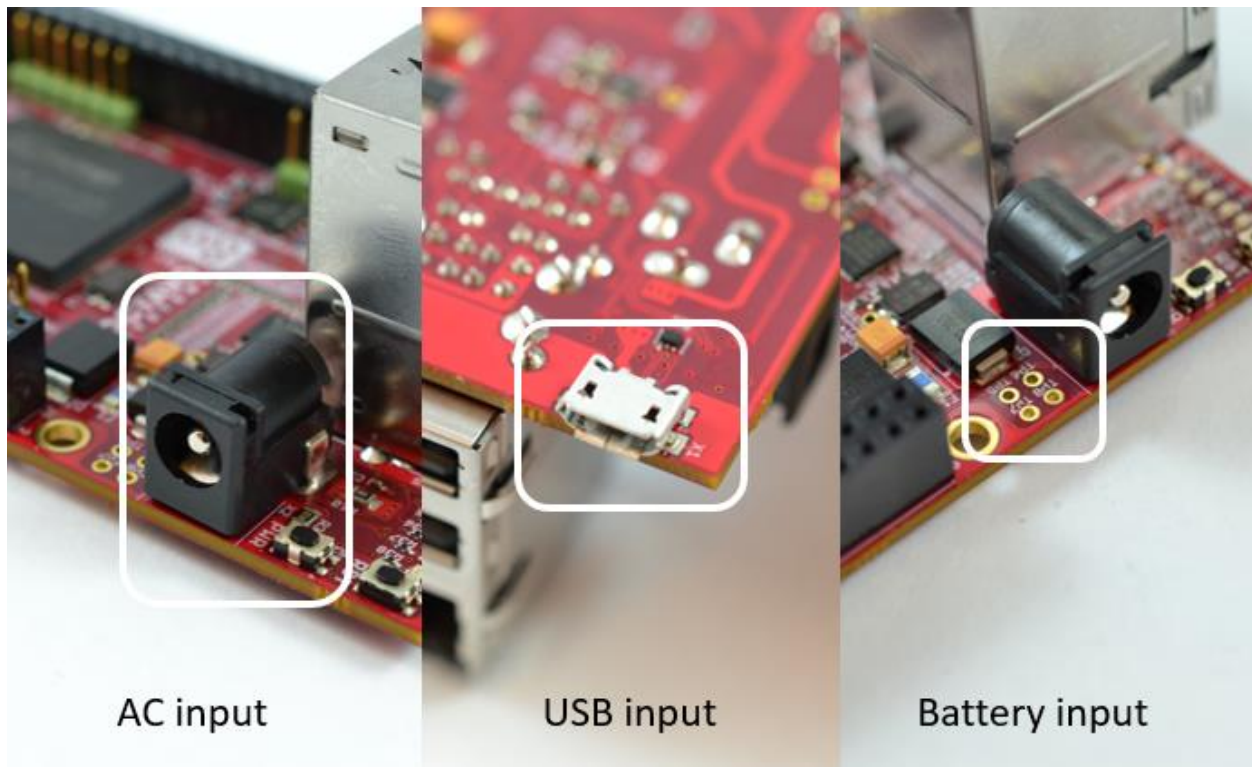


Figure 1: Picture of OSD3358-SM-RED Highlighting Power Inputs

1.1 Power up procedure

The following steps describe the power up sequence of the board when powered via USB, AC Adapter, or battery inputs.

1. Connect your preferred power source:
 - a. USB:
 - i. Connect the micro-USB side of the USB cable to the micro-USB port X1
 - ii. Connect the USB side to one of the USB ports on a laptop/computer or other USB power source.



Figure 2: Powering the Board using a USB Cable

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



- b. AC adapter:
 - i. Plug in the 5V AC adapter with 2.5 mm x 5.5 mm barrel into the power connector X2.



Figure 3: Powering the Board using an AC-DC Adapter

- c. Battery input terminals:
 - i. Connect the battery input terminals to the battery terminals as shown below.

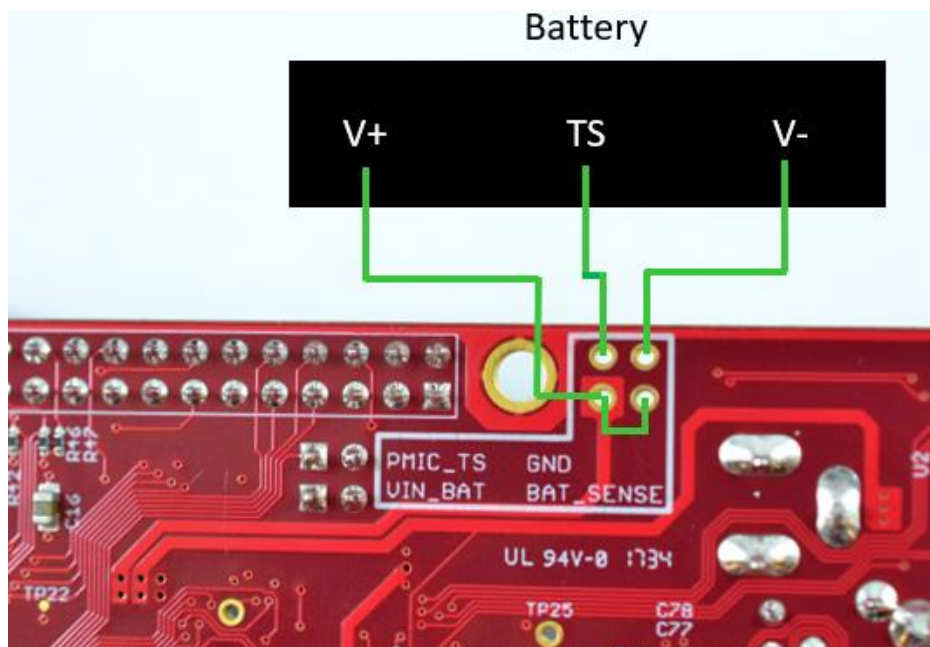


Figure 4: OSD3358-SM-RED Battery Connections (back-side of board)

Octavo Systems LLC
Copyright 2017-2021

NOTE: Notes on connecting the PMIC_TS input of the SiP can be found in "Battery pack temperature monitoring" section of the TPS65217C PMIC datasheet. If the battery does not have a temperature output, the battery temperature monitoring mechanism of the PMIC can be bypassed by connecting a 10KOhm resistor between PMIC_TS (TP7) of the board and GND.

2. LED D1 will turn on immediately after power is applied to the board.



Figure 5: OSD3358-SM-RED Board Boot - LED D1 ON

3. LEDs D4 – D7 turn on in the sequence indicated below indicating Linux is booting.



Figure 6: OSD3358-SM-RED Board Boot - LEDs D4 - D7 turn ON

2 OSD3358-SM-RED Use Cases

The OSD3358-SM-RED Platform can be used in 3 different modes. A description of each mode and instructions on how to set up each mode is described below.

2.1 Standalone Boot-Up

The board can be used as a standalone platform for software development. To boot it up as a standalone system, the following components will be needed.

1. 5V AC adapter
2. HDMI/DVI-D monitor
3. Micro HDMI to HDMI cable and/or HDMI to DVI-D adapter
4. Wired/wireless USB keyboard and mouse
5. Ethernet cable or WiFi USB adapter connected to a network for internet access

The following steps describe the procedure to setup the board as a standalone software development tool.

1. Connect the micro HDMI connector of the HDMI cable to X6 connector on board.
2. Connect the HDMI connector of the HDMI cable into the monitor. HDMI to DVI-D convertor may be necessary if the monitor only has a DVI-D input.

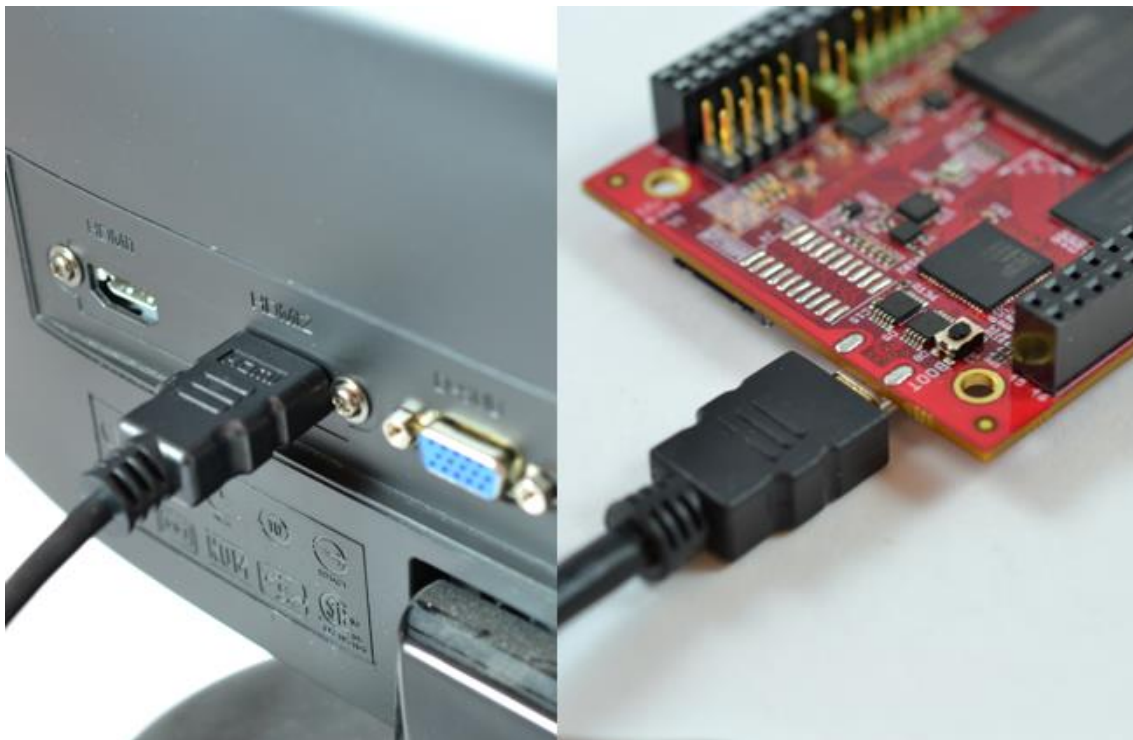


Figure 7: Connecting OSD3358-SM-RED to a Monitor via HDMI Cable

3. Connect the wired/wireless USB keyboard and mouse inputs to any of the four USB ports available on X4 and X5 connectors of the board.



Figure 8: Wireless Adapter for Keyboard and Mouse plugged in to the Board

4. If internet connectivity is required:
 - a. a WiFi-USB adapter can be plugged into one of the available USB ports on X4 or X5 connector. A list of compatible adapters can be found Compatible WiFi adapters: http://www.elinux.org/Beagleboard:BeagleBoneBlack#WIFI_Adapters OR
 - b. An ethernet cable can be plugged into the ethernet connector receptacle on X4.

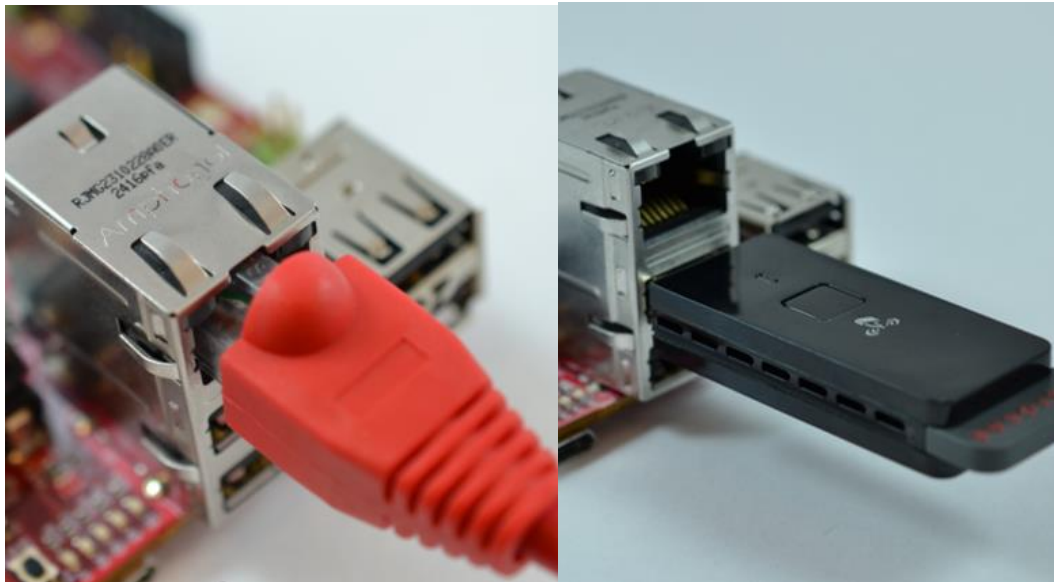


Figure 9: Ethernet Cable / Wi-Fi Adapter plugged in to the Board

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



5. Plug the 5V AC adapter into the power connector X2.



Figure 10: AC-DC Adapter Plugged in to the Board

6. The board will go through the default boot up process
7. After the board boots up, the monitor screen should show a Linux desktop environment as shown below.



Figure 11: OSD3358-SM-RED Desktop Environment

8. Click on the *Start* → *System Tools* → *Q Terminal* to open up a command terminal.



Figure 12: Shell Command Terminal on Desktop

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



2.2 Booting as USB Client Connected to a Computer/Laptop

The OSD3358-SM-RED platform can also be powered using a computer/laptop's USB port. The micro-USB to USB cable provided in the box can be used to connect the RED board to a computer/laptop. Unlike the standalone mode, this setup does not require additional hardware. The following steps describe the procedure to power the board as a USB client and set up the development environment.

1. Connect the micro-USB side of the USB cable to the micro-USB port X1
2. Connect the USB side to one of the USB ports on a laptop/computer.



Figure 13: Powering OSD3358-SM-RED using a USB Cable

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021

3. Wait for the board to show up as a mass storage device on the computer/laptop

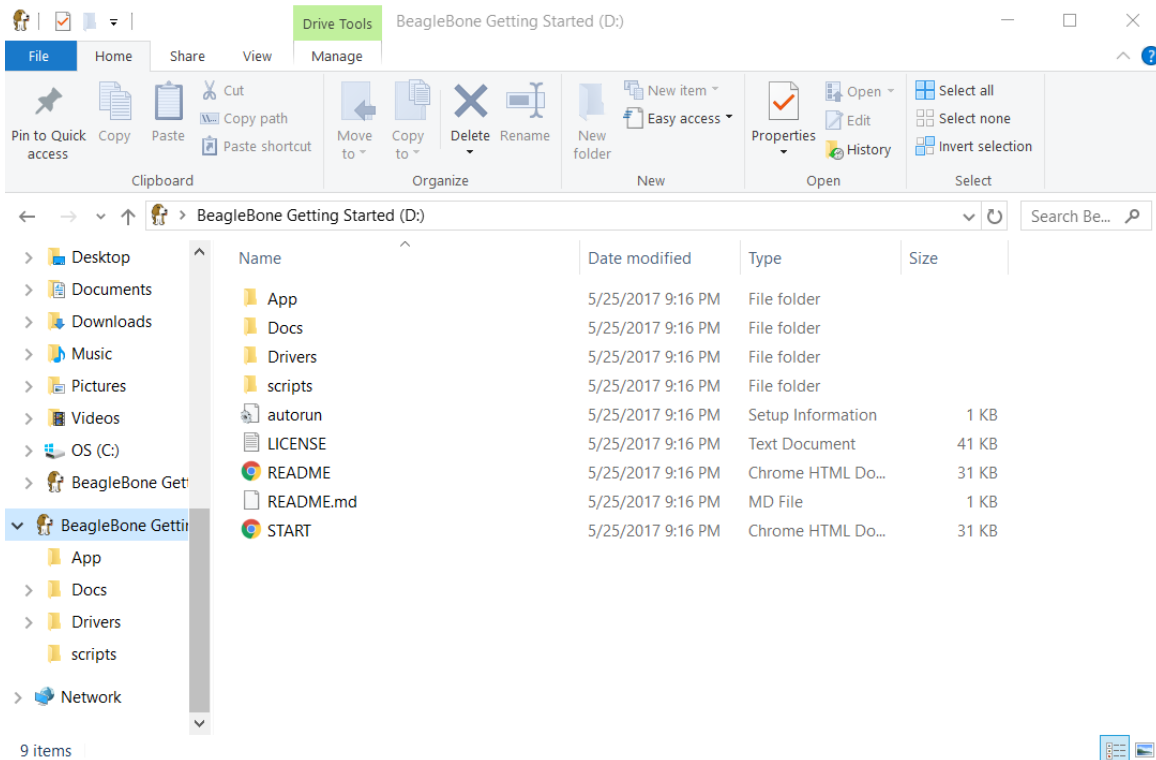


Figure 14: OSD3358-SM-RED USB Mass Storage Filesystem Window

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



- Open a browser (Firefox/Chrome) and access the url: <http://192.168.7.2>. The webpage hosted on the webserver of the board should indicate that the board is connected as shown below.

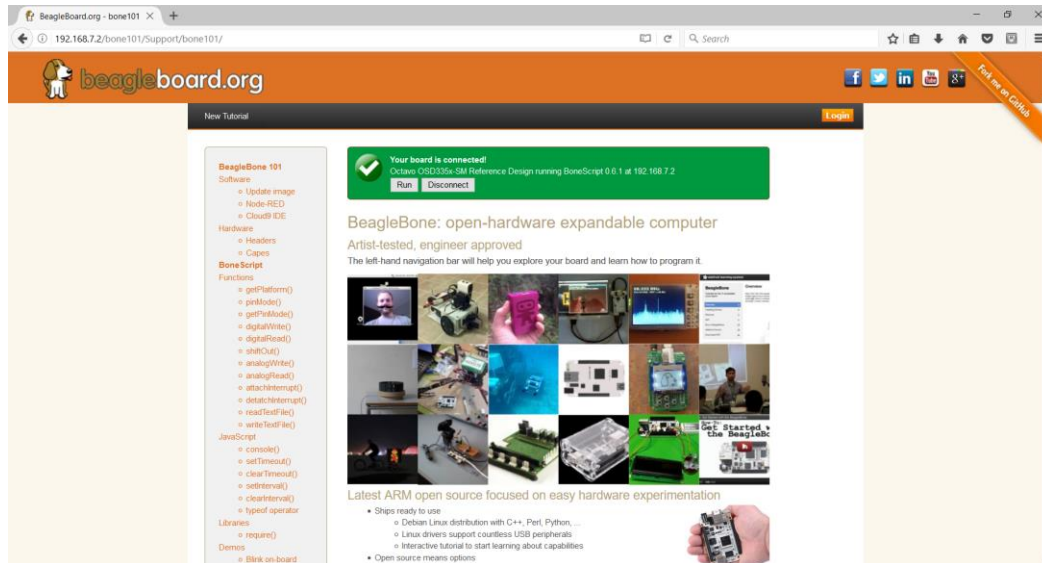


Figure 15: OSD3358-SM-RED Home Webpage - Indicates a Connection to the Board

- Access the cloud9 IDE through the url: <http://192.168.7.2:3000/>. Cloud9 IDE is an open-source web-based development environment supporting several languages. The environment should look like the following Figure with workspace containing examples that can be executed. Cloud9 also loads a shell terminal that can be seen at the bottom of the following figure.

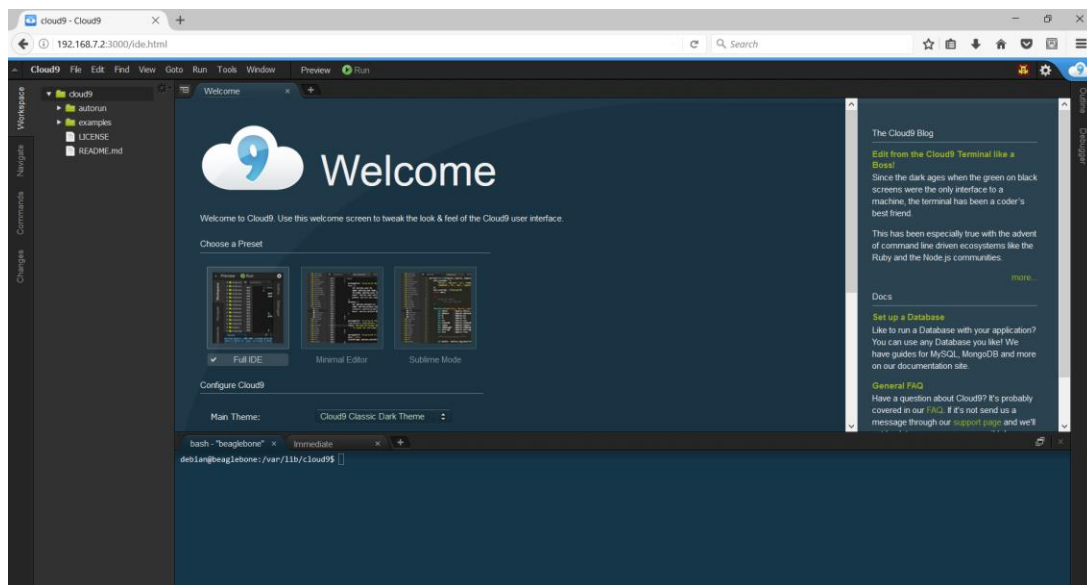


Figure 16: Cloud9 IDE

2.3 Boot Messages Through UART

The processor on the OSD3358-SM-RED platform (RED board) sends boot messages through the UART0 port, J3. This is useful in embedded Linux development or if there are boot problems to verify proper initialization of all components on the board. After boot, the UART0 port will function as a Linux console terminal. Accessing this requires a serial connection to a host device with a serial port. A USB-UART serial adapter can also be used for host devices with a USB port. A terminal program, such as Putty is required for communication.

The following steps describe the procedure to access the boot messages during boot and the terminal through UART0 port using a standard computer/laptop as a host device (Note: Given the computer/laptop has no native serial ports, a USB-UART serial adapter must be used):

1. Connect the UART side of the adapter to header J3 of the RED board. Make sure to check for proper orientation of the cable to connect Tx and Rx signals of the UART0 port appropriately to the adapter's signals.

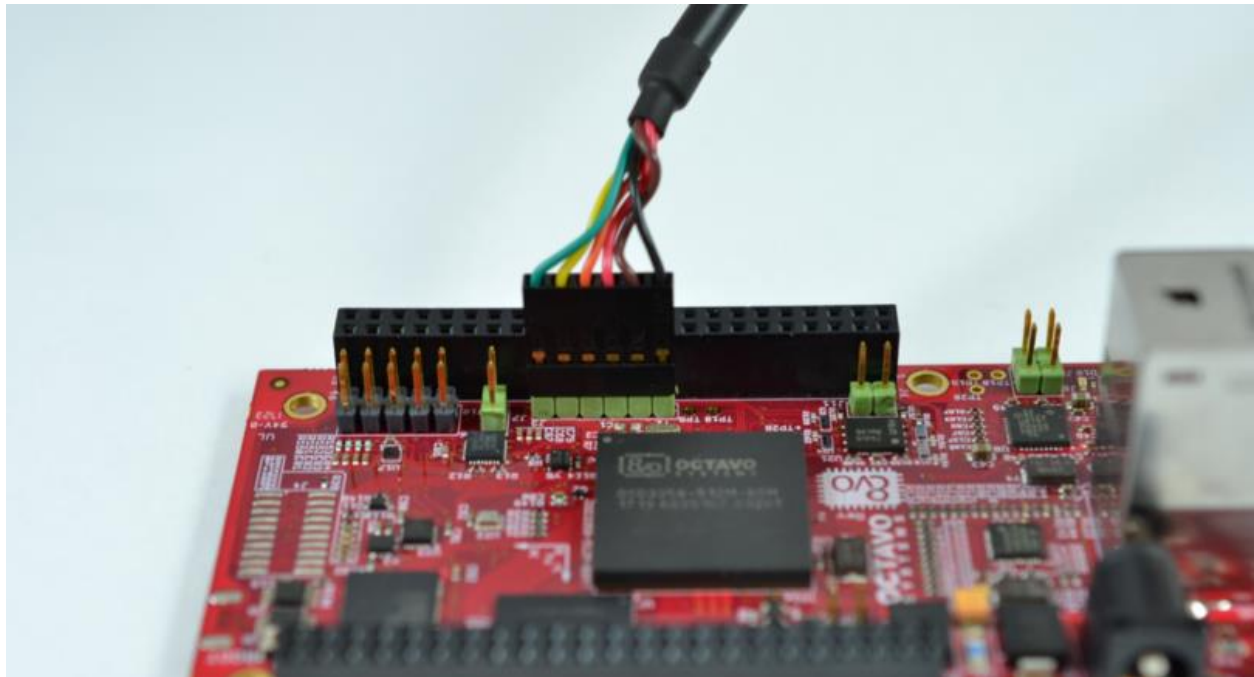


Figure 17: USB-UART Adapter Cable Plugged in to the Board

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



2. Connect the USB side of the adapter to the computer/laptop



Figure 18: UART-USB Adapter Plugged in to the Host Computer

3. Check which COM port of the computer/laptop the adapter is connected to using the device manager.

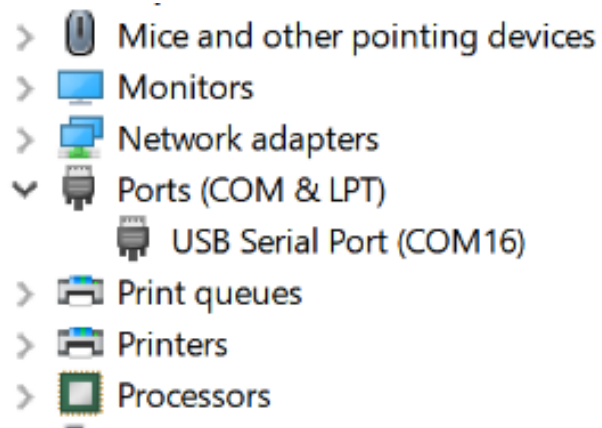


Figure 19: USB-UART Adapter Detected as COM Port 16

4. Open a serial console (open-source applications like Putty come with one). Select the serial communication option and specify the port and speed of the protocol as shown below and click on open.

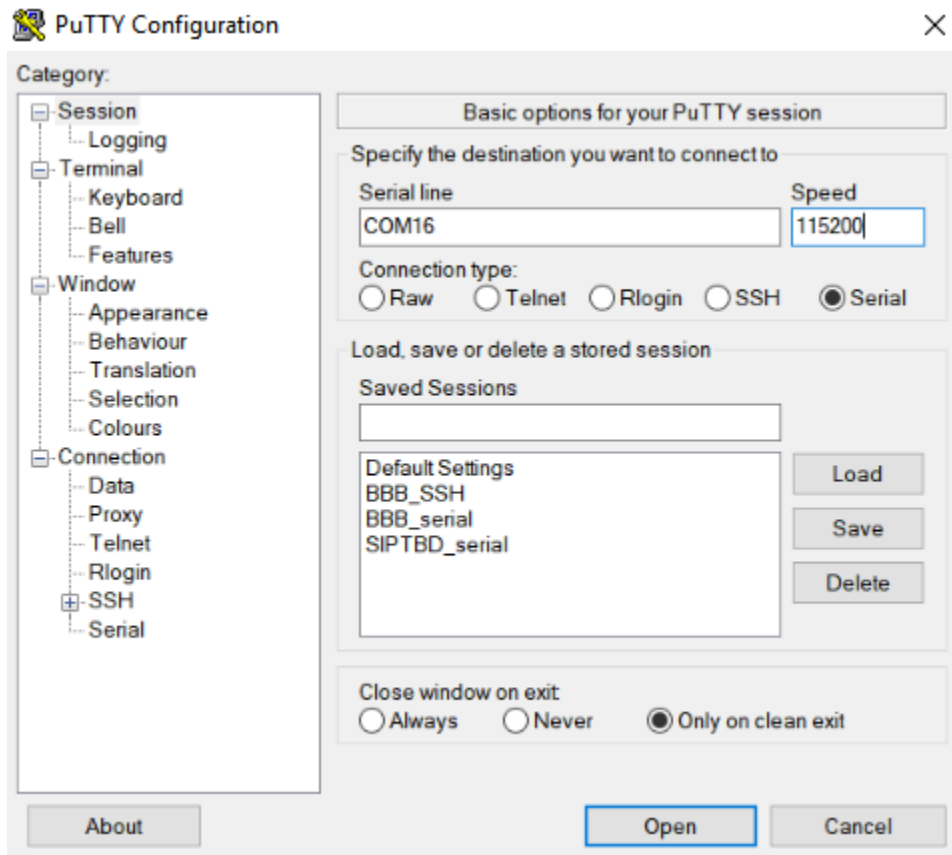


Figure 20: PUTTY Configuration for Serial Communication using USB-UART Adapter

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



5. After a new serial console is open, connect the board to a power supply through one of the three power inputs of the board. The boot messages should begin to scroll on the console. After boot, the console will ask for secure shell login. You can login to the shell using the credentials – **debian:temppwd** .

A screenshot of a PuTTY terminal window titled 'COM16 - PuTTY'. The terminal displays the following text:

```
Debian GNU/Linux 9 beaglebone ttyS0
BeagleBoard.org Debian Image 2017-08-31
Support/FAQ: http://elinux.org/Beagleboard:BeagleBoneBlack_Debian
default username:password is [debian:temppwd]

beaglebone login: debian
Password:
Linux beaglebone 4.9.45-ti-r57 #1 SMP PREEMPT Fri Aug 25 22:58:38 UTC 2017 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
debian@beaglebone:~$
debian@beaglebone:~$
debian@beaglebone:~$
debian@beaglebone:~$
```

Figure 21: Debian Linux Terminal after Boot

2.4 User LEDs

While the LEDs D4 through D7 can be controlled as desired, they have the following standard function when the system is booted.

- D4: Activity indicator for MMC1 interface (eMMC)
- D5: Activity indicator for the kernel
- D6: Activity indicator for MMC0 interface (SD card)
- D7: Heartbeat indicator from Linux kernel

2.5 Powering Down / Power Cycling

Once the system has completely booted, the board can be safely powered down in the following way:

1. Execute the shutdown command on Linux command line. Wait for LEDs D5 and D7 to turn off. Unplug the power source.
2. Press the PWR button S2 for less than 8 seconds

The board can be reset using one of the following methods:

1. Reset Only: Press the RST button S1
2. Power Cycle: Press and hold the PWR button S2 pushed for 8 seconds

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



3 Overview of OSD3358-SM-RED

The OSD3358-SM-RED is a Reference, Evaluation, and Development platform for the OSD335x-SM family of devices from Octavo Systems. The OSD335x-SM integrates the AM335x processor from Texas Instruments (TI) along with the TI TPS65217C PMIC, the TI TL5209 LDO, up to 1 Gigabyte (GB) of DDR3 Memory, a 4 Kilobyte (KB) EEPROM for non-volatile configuration storage and resistors, capacitors, and inductors into a single 21mm x 21mm design-in-ready package. The AM335x is a low cost, low power ARM® Cortex® A8 based processor that is widely used in the industry as well as the open source and hobbyist communities.

The platform offers access to many peripherals of the processor and functions as a foundation to swiftly design and develop applications based on the OSD335x-SM. It comes out of the box with software compatible with Beagleboard.org® which has an active open-source community and support.

3.1 Specification

Figure 22 shows the block diagram of the OSD3358-SM-RED.

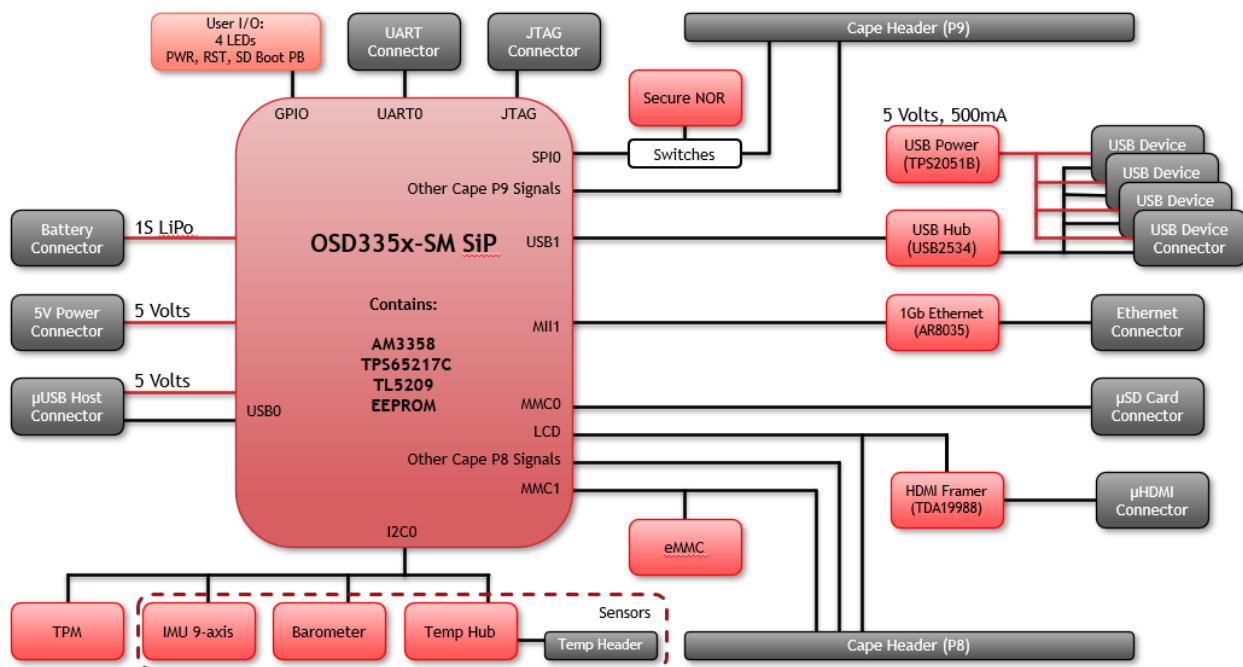


Figure 22: OSD3358-SM-RED Block Diagram

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021

The following table provides an overview of the specification of the board.

Table 1: OSD3358-SM-RED Specification

PART	PROPERTY	VALUE
OSD3358-512M-BSM (U1)	Processor	TI Sitara AM3358 ZCZ 1GHz SGX530 3D Graphics accelerator
	Real-time processor	2 Real-time Programmable Units (PRU)
	Memory size	512MB DDR3L, 400MHz
	Power Management	TI TPS65217C TI TL5209 LDO
	Non-volatile memory	4KB EEPROM
	Power inputs	AC (from AC-DC adapter), USB & Battery
Power Jack (X2)	Dimensions	2.5 mm x 5.5 mm barrel jack
Power button (S2)	Function	Power cycle the board/power the board ON/OFF
Reset button (S1)	Function	Holds the processor in reset
Boot button (S3)	Function	Used for SD-card boot
User LEDs (D4 – D7)	Number	4
eMMC (U7)	Part number	SDIN8DE2-16G (or equivalent)
	Memory Size	16GB
4 port USB host ports (U8)	Max. current output	500mA
Micro-USB client port (X1)	Default Max. input current	500mA
HDMI Framer (U10)	Audio supported resolutions	• 1920 x 1080 • 1280 x 720 • 720 x 480 • 640 x 480
Ethernet port (X5)	Supported speeds	10BASE-Tx/100BASE- Tx/1000Base-T IEEE 802.3 compliant
Debug (J3 and J4)	ports	JTAG, UART (Boot messages)
JTAG connector (J4)	Part number	FTR-110-03-G-D-06
ADC	Number of inputs	7 available. 1 PMIC voltage monitoring input
	Reference voltage	1.8V
	Resolution	12 bit
NOR Flash (U21)	Part number	S25FL127S (or equivalent)
	Memory size	128MB
	Supply voltage	3.3 V
	Sector size	64KB

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



	Package	8-pin W-PDFN
TPM (U16)	Part number	AT97SC3205T (or equivalent)
	I2C bus	I2C0
	I2C address	0x29
	Supply voltage	3.3V
	Package	32-pin QFN
Temperature hub (U17)	Part number	TMP468
	Number of channels	9: 1 board monitoring channel, 4 channels available through expansion header and 3 channels unused
	Local and remote diode accuracy	±0.75°C
	Local sensor accuracy	±0.35°C
	Supply voltage	3.3V
	I2C bus	I2C0
	I2C address	0x48
Barometer (U22)	Part number (V3)	BMP280
	Part number (V4)	LPS22HBTR
	Pressure range	300 – 1100 hPa (V3) 20 – 1260 hPa (V4)
	Relative accuracy	±0.16 hPa (V3) ±0.1 hPa (V4)
	Supply voltage	3.3V
	I2C bus	I2C0
	I2C address	0x76 (V3) 0x5C (V4)
IMU (U23)	Part number (V3)	MPU-9250
	Part number (V4)	ICM-20948
	Gyroscope word length	16-bits
	Accelerometer word length	16-bits
	Magnetometer word length	14-bits
	Supply voltage	3.3V (V3) 1.8V (V4)
	I2C bus	I2C0
	I2C address	0x68
PCB	Layers	4
	Minimum trace width	6mils

Board	Minimum via drill	12mils
	Minimum via diameter	24mils
	Size	108 mm x 55 mm
	Maximum height	32.15 mm
	Weight	60 grams

NOTE: The NOR flash and eMMC parts are replaceable with equivalent footprint compatible parts.

3.2 Processor (U1)

The OSD3358-SM-RED board uses the OSD3358-512M-BSM (U1) System-in-Package (SiP) device from Octavo Systems. The SiP package integrates TI's AM3358 processor, TI power management system, 512MB of DDR3 memory and 4KB EEPROM in 21mm x 21mm BGA package. A list of references for each of the main components integrated in the OSD3358-SM is given below:

- AM335x datasheet: <http://www.ti.com/lit/ds/sprs717j/sprs717j.pdf>
- AM335x Technical Reference Manual: <http://www.ti.com/lit/ds/sprs717j/sprs717j.pdf>
- TPS65217C Datasheet: <http://www.ti.com/lit/ds/symlink/tps65217.pdf>
- Powering AM335x with TPS65217x: <http://www.ti.com/lit/ug/slvu551i/slvu551i.pdf>
- TL5209 LDO Datasheet: <http://www.ti.com/lit/ds/symlink/tl5209.pdf>
- 4KB EEPROM Datasheet: <http://www.microchip.com/wwwproducts/en/24LC32A>

NOTE: The 4KB EEPROM sits on the I2C0 line of the AM3358 processor. Along with the board information that is needed to boot Linux, it is pre-programmed with 256 bytes of device specific information described in OSD3358-SM datasheet (<https://octavosystems.com/docs/osd335x-sm-datasheet/>).

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



3.3 PC USB Interface (X1)

The board has a micro-USB OTG port X1 (shown below) that can connect to the USB port of a laptop/computer. In addition to powering the board, this port can also be used to access the board as a storage device or through an RNDIS ethernet connection. SSH clients such as PUTTY (<http://www.putty.org/>) can be used to connect and run programs through a secure shell command prompt. In addition, the Beagleboard.org® compatible Linux image allows a web browser (Chrome, Firefox, etc.) to browse to the web server running on the board and access Cloud9 IDE, an easy-to-use web-based programming platform can also be used for software development.



Figure 23: USB OTG Port of the Board

NOTE: OSD3358-SM-RED Platform is compatible specifically with Linux images that can be installed on the boards BeagleBone® Black, BeagleBone® Black Wireless, BeagleBone® Blue, PocketBeagle®, etc. Suitable modifications to the device tree need to be made for complete functionality. Please see Section 5.2 for details.

3.4 Serial Debug Interface (J3)

The UART0 port of the processor is exposed as a 1x6 header J3 (shown below). This interface serves as a debug port that can be accessed via USB port of a laptop/computer using a USB to Serial interface adapter. The processor sends status messages to this port during boot up allowing for debug and development.

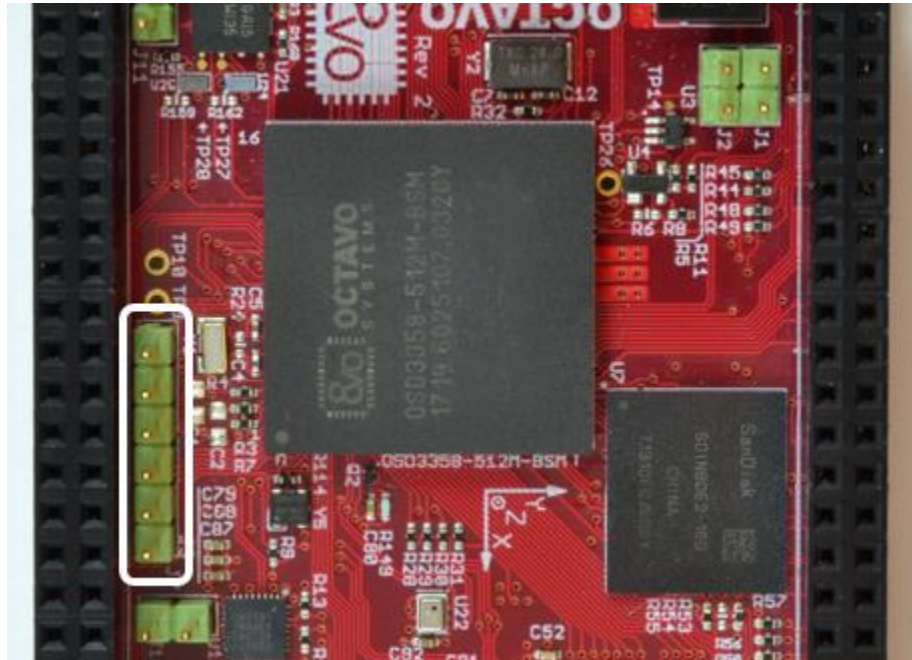


Figure 24: UART Header (boxed)

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



3.5 EMMC (U7)

A 16GB embedded Multi-Media Card (eMMC), U7, is connected to 8-bit wide MMC1 port of the processor (shown below). It contains the Linux image that the board will boot from by default. The board can be made to boot from other sources and the Linux image on the eMMC can be updated as needed.



Figure 25: eMMC (boxed)

3.6 Cape Headers (J5 & J6)

Two 2x23 cape headers J5 and J6 (shown below) serve as expansion interfaces to the processor exposing a wide range of peripherals of the processor. They are compatible with the BeagleBone® Black headers P8 & P9

(http://elinux.org/Beagleboard:Cape_Expansion_Headers) meaning capes for the BeagleBone® Black are compatible and work with the OSD3358-SM-RED board. More information on these headers can be found in Section 4.22.

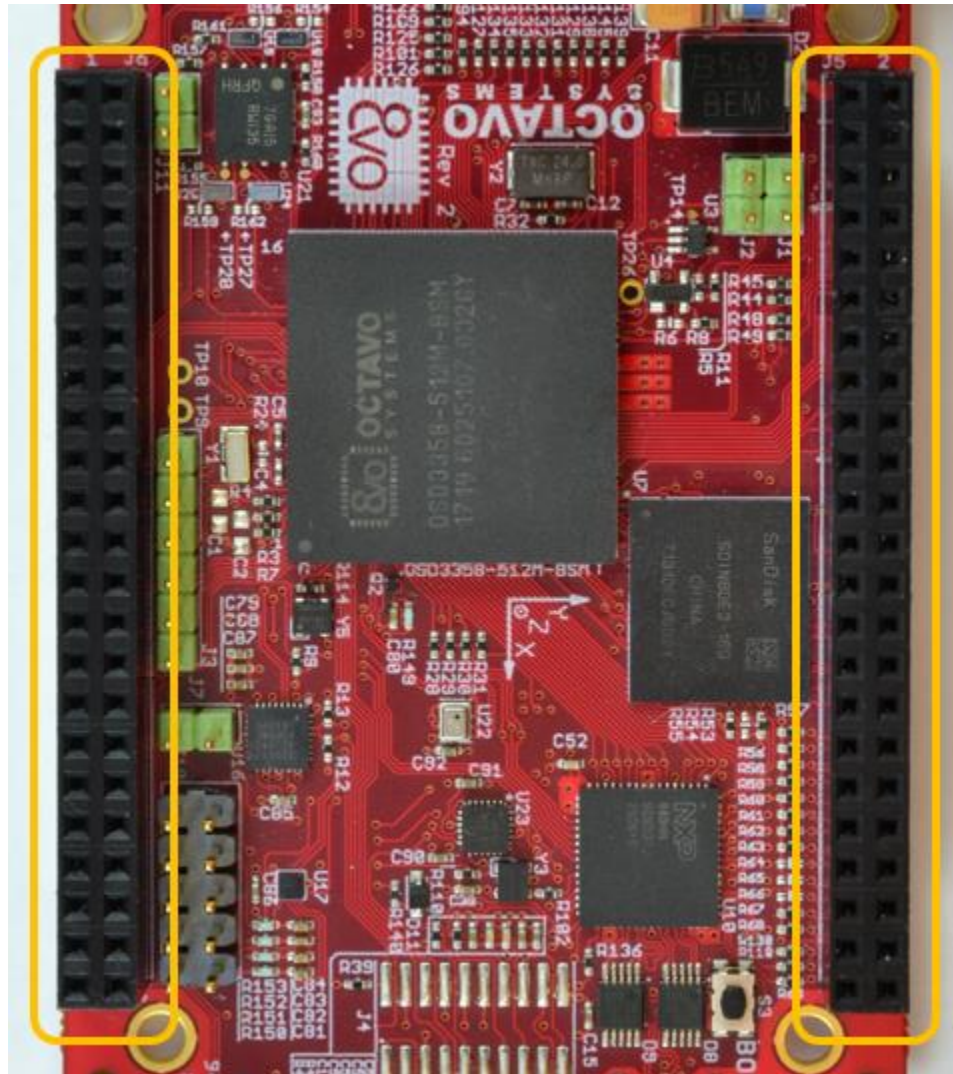


Figure 26: Cape Headers (boxed)

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



3.7 JTAG (J4)

A 20 pin CTI JTAG footprint J4 (shown below) is exposed for mounting a JTAG header on the board. This port can be used for software development and debug by connecting to the board through a JTAG emulator.

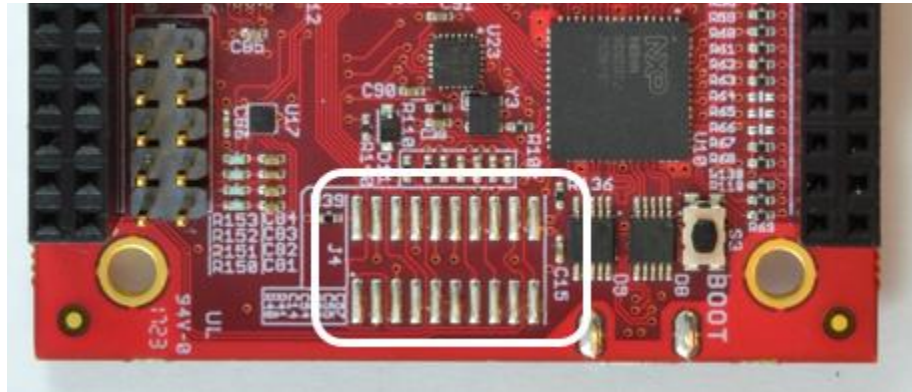


Figure 27: JTAG Header Footprint (boxed)

To use the JTAG footprint it is recommended to use the emulators and JTAG header parts listed below. The JTAG footprint comes with solder on the pads to make it easy to solder the JTAG connector.

JTAG Header:

1. Samtec Part No: FTR-110-03-G-D-06

Emulators:

1. Blackhawk XDS2xx USB debug probe (<http://www.blackhawk-dsp.com/products/USB200.aspx>)
2. Blackhawk XDS100v2 USB debug probe (<http://www.blackhawk-dsp.com/products/usb100.aspx>)

Software:

1. TI Code Composer Studio (<http://www.ti.com/tool/ccstudio>)
2. IAR Embedded Workbench (<https://www.iar.com/iar-embedded-workbench/>)

3.8 Micro-SD Card Slot (X3)

A Micro-SD card slot X3 (shown below) interfaces the processor with a micro-SD card. The slot is connected to 4-bit wide MMC0 interface of the processor. A micro-SD card inserted in the slot can be used as a storage device, a boot source or can be used to flash the on board eMMC to update the software image of the board.

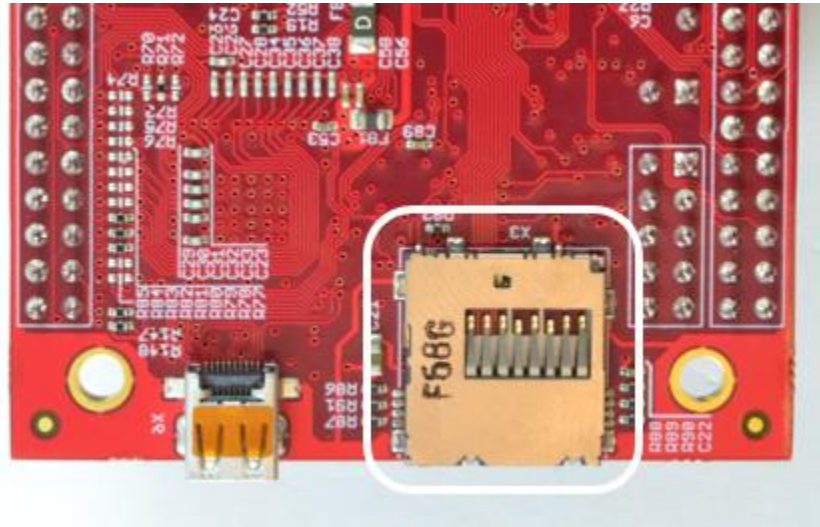


Figure 28: Micro-SD Card Slot (boxed)

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



3.9 HDMI Interface (X6)

An HDMI port X6 (shown below) provides an HDMI connection interface to the processor's 16-bit LCD controller through an HDMI framer (U10).

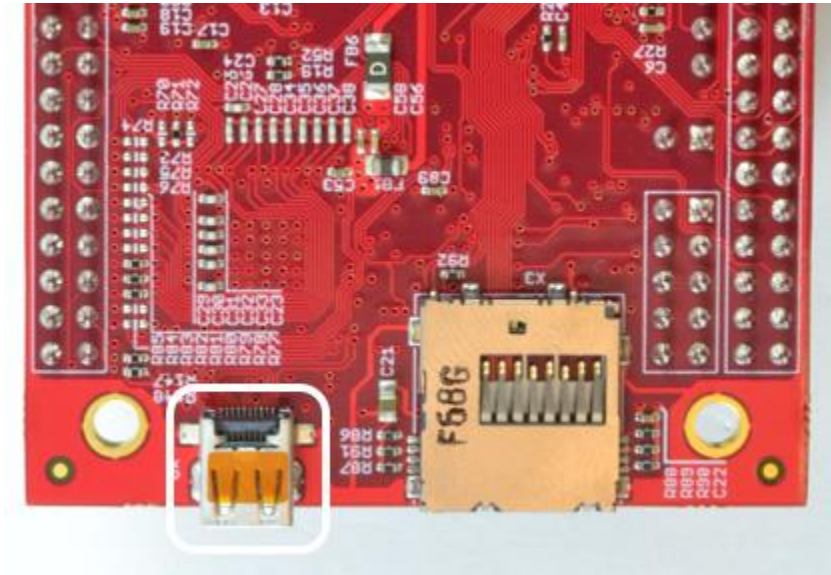


Figure 29: micro-HDMI Port (boxed)

The following resolutions are supported via the preloaded software image.

- 1280 x 1024
- 1440 x 900
- 1024 x 768
- 1280 x 720

HDMI Framer TDA199988 Datasheet:

https://media.digikey.com/pdf/Data%20Sheets/NXP%20PDFs/TDA19988_DS_21_july_2011.pdf

NOTE: When the HDMI interface is used, the LCD controller pins that are multiplexed with other functions and pinned out to the headers should not be used. When the HDMI interface is not used, the LCD pins can be used for their multiplexed functions through the headers J4 and J5.

3.10 Gigabit Ethernet (X4)

The board has a Gigabit Ethernet interface port X4 (Shown below). The ethernet PHY is capable of 10/100/1000 Mbps speeds and is connected to the RGMII port of the processor. The Cloud9 IDE, web-server and the secure shell command prompt can be accessed by connecting an ethernet cable to this port in addition to the micro-USB port.

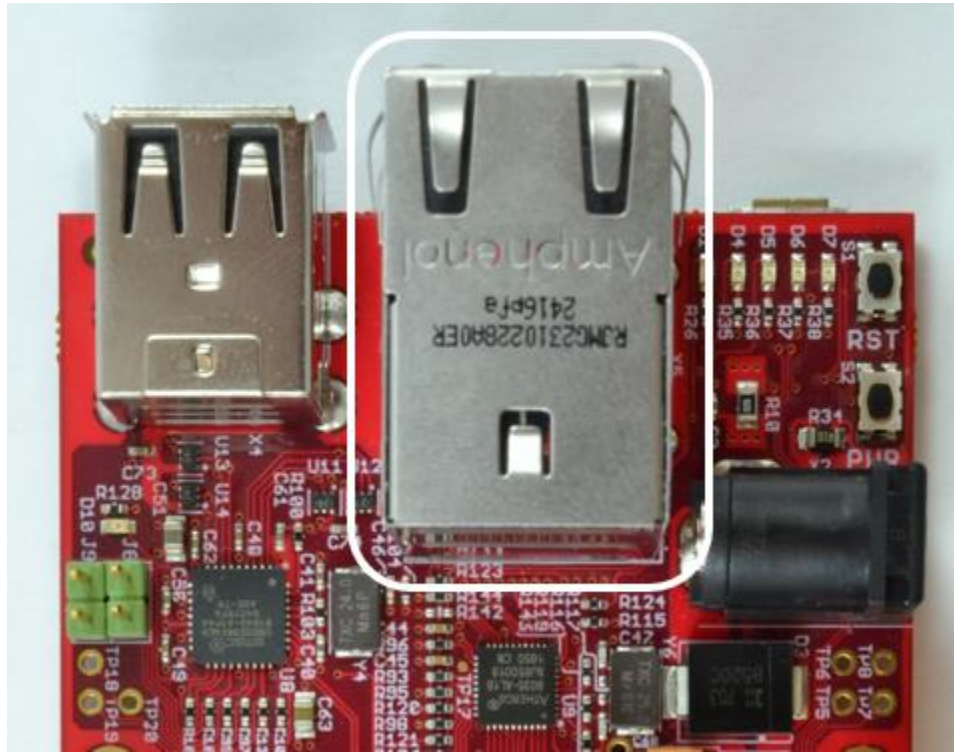


Figure 30: Gigabit Ethernet Port (boxed)

Ethernet PHY AR8035 Datasheet:

[https://media.digikey.com/pdf/Data%20Sheets/CSR%20PDFs/AR8035_DS_\(Atheros\)_Mar2011.pdf](https://media.digikey.com/pdf/Data%20Sheets/CSR%20PDFs/AR8035_DS_(Atheros)_Mar2011.pdf)

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



3.11 Four (4) Port USB Hub (X4 & X5)

The board comes with a total of four (4) USB 2.0 Type-A ports in connectors X4 and X5 (shown below) facilitated by a USB hub controller that is connected to USB1 port of the processor. These ports can be used to connect USB storage devices and other USB gadgets such as WiFi adapters and wireless/wired keyboard and mouse. The 4 ports can supply a combined current of 500mA to the devices that are connected to them when the board is powered through AC input. While they cannot supply 500mA when the board is powered using the USB port of a laptop/computer, low powered USB devices can still be used.

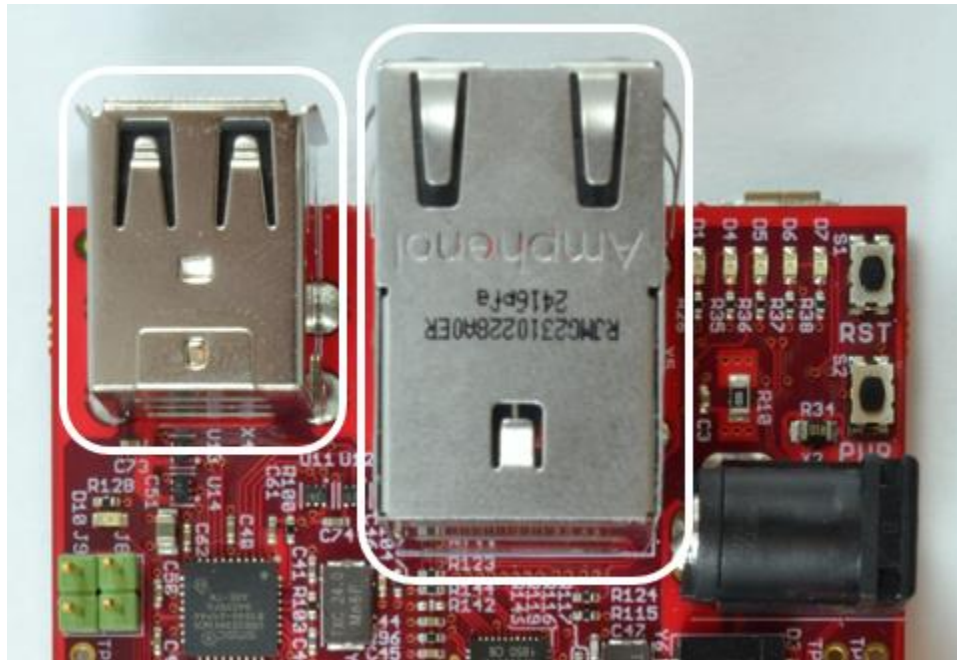


Figure 31: USB 2.0 Ports (boxed)

USB port hub USB2534-1080AEN datasheet:

<http://ww1.microchip.com/downloads/en/DeviceDoc/00001713A.pdf>

3.12 Inertial Measurement Unit (IMU) (U23) - V3

The board comes equipped with a 9 axis IMU (shown below). The MPU-9250 (U23) sits on the I2C0 line of the processor. It has a 3-Axis gyroscope, a 3-Axis accelerometer and a 3-Axis magnetometer inside and functions as a 9-Axis motion tracking device. More information on how to use it is given in Section 5.3.

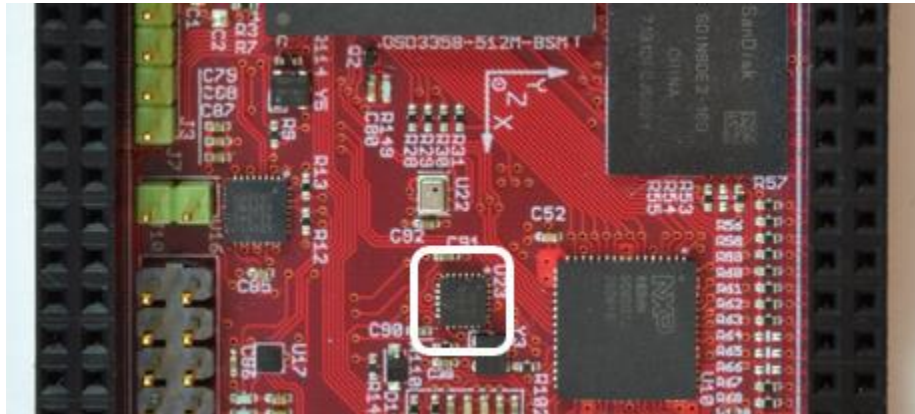


Figure 32: IMU - Board Revision V3 (boxed)

IMU MPU9250 datasheet: <https://www.invensense.com/wp-content/uploads/2015/02/PS-MPU-9250A-01-v1.1.pdf>

3.13 Inertial Measurement Unit (IMU) (U23) - V4

The IMU on board revision V4 is shown below. ICM-20948 (U23) sits on the I2C0 line of the processor. It has a 3-Axis gyroscope, a 3-Axis accelerometer and a 3-Axis magnetometer inside and functions as a 9-Axis motion tracking device. More information on how to use it is given in Section 5.3/5.4.



Figure 33: IMU - Board Revision V4 (boxed)

ICM-20948 datasheet: <https://invensense.tdk.com/wp-content/uploads/2016/06/DS-000189-ICM-20948-v1.3.pdf>

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



3.14 Temperature sensor hub (U17)

A high accuracy 9 channel temperature sensor TMP468 (U17) (shown below) is connected to I2C0 interface of the processor. 4 channels of the sensor are exposed using a 2x5 header P3 (shown below). 1 remote channel is utilized to measure temperature on board and 1 channel internal to the temperature sensor gives the temperature of the sensor itself. More information on how to use it can be found in Section 5.3/5.4.

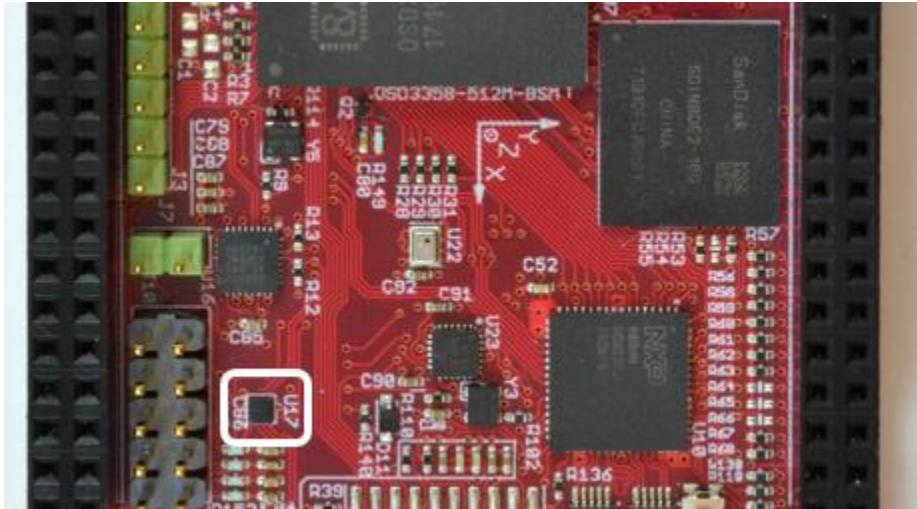


Figure 33: Temperature Sensor (boxed)

Temperature sensor TMP468 datasheet: <http://www.ti.com/lit/ds/symlink/tmp468.pdf>

3.15 Pressure sensor (U22) - V3

A Piezo-resistive pressure sensor BMP280 (U22) (shown below) is connected to the processor through I2C0 interface. The pressure sensor can deliver high precision pressure measurements that aid in altitude measurements. More information on this device and how to use it can be found in Section 5.3.



OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



3.17 Trusted Platform Module (TPM) (U16)

A Trusted Computing Group (TCG) compliant TPM v1.2 U16 (shown below) on board facilitates development of security software. Applications supported by the TPM include secure cryptographic key generation, secure boot, authentication, and random number generation. The TPM sits on I2C0 interface of the processor. While the board does not come with any software support for secure applications out of the box, inclusion of the TPM enables development of such software solutions.

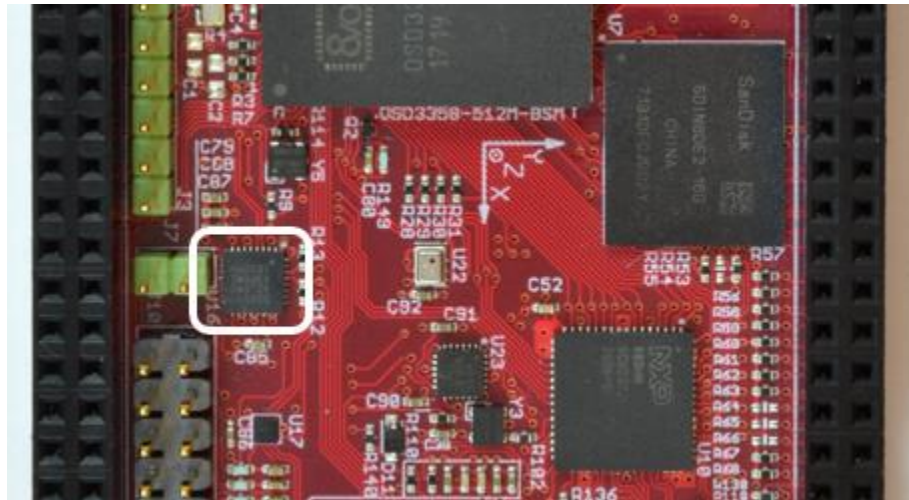


Figure 36: TPM (boxed)

TPM AT97SC3205T datasheet: <http://www.atmel.com/Images/Atmel-8883S-TPM-AT97SC3205T-Datasheet-Summary.pdf>

3.18 NOR Flash (U21)

A 128MB Serial NOR Flash U21 (shown below) is connected to the processor via SPI0. This non-volatile storage space can be utilized by applications running on the processor when needed and can also be used for secure boot in conjunction with the TPM. The SPI0 serial interface used to communicate with the Flash is multiplexed with the I2C1 and UART2 interfaces on header J6. When using the NOR Flash, the 2 pins of the jumper J11 should be shorted together or the FLASH_EN signal controlled by the processor should be pulled high. This will disable the use of the I2C1 and UART2 interfaces through J6. If the Flash is not used, the pins of the jumper J11 must not be populated to use the I2C1 and UART2 interfaces of the processor through header J6.



Figure 37: NOR Flash (boxed)

NOR flash S25FL127S datasheet: <http://www.cypress.com/file/177961/download>

4 Detailed Hardware Design

The following sections describe the hardware design of the OSD3358-SM-RED platform.

4.1 Power Management

The following figure shows the power system of the OSD3358-SM-RED board. The voltage rails, their current sourcing capabilities and the components that are powered by them are also shown.

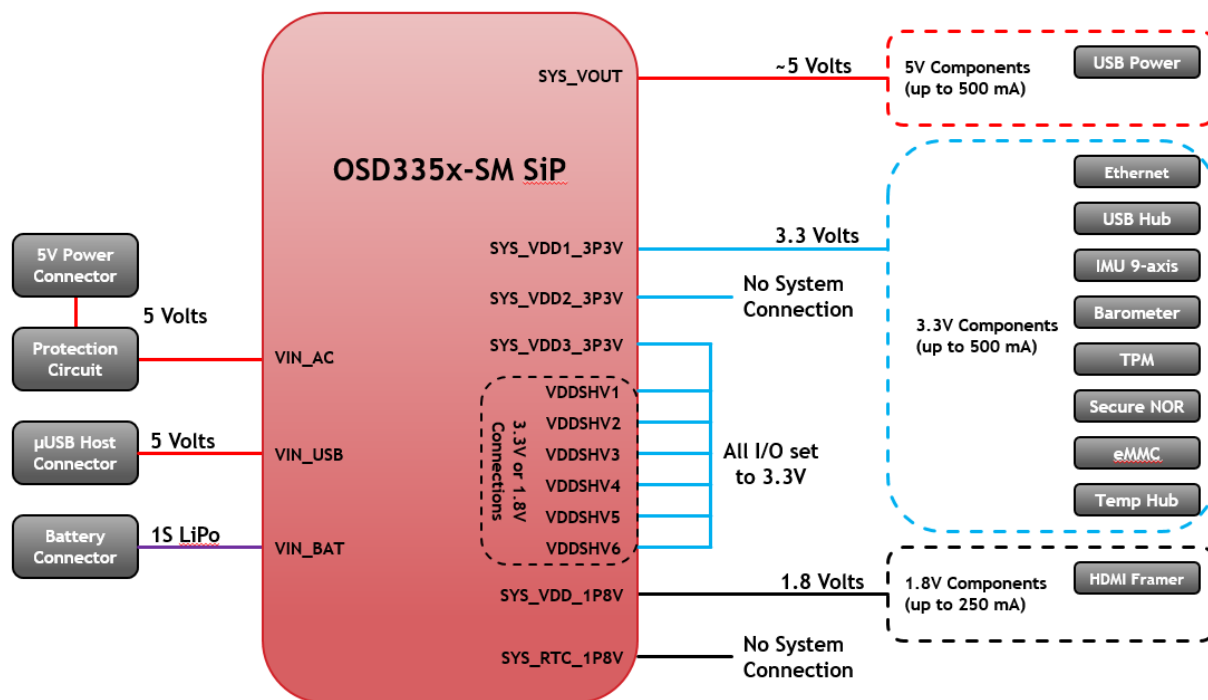


Figure 38: OSD3358-SM-RED Power System Block Diagram

The power system of the platform is comprised of:

1. AC, USB, battery inputs
2. Protection circuit on AC input
3. The TPS65217C PMIC and TL5209 LDO inside the OSD3358-512M-BSM.

The PMIC is responsible for powering the AM335x processor and the DDR3 as well as provide output power for other system needs. It provides configurable power-up and power-down sequencing required by the processor and monitors the processor input voltage levels. It contains 3 DC-DC power converters, 2 LDOs and 2 load switches that are configured as LDOs and used as power supplies. It can be powered by any combination of a 5V AC adapter, USB

port, or Single Cell Li-Ion battery. Figure 39 shows the power system of OSD335x-SM including connections between the PMIC and various power domains of the processor.

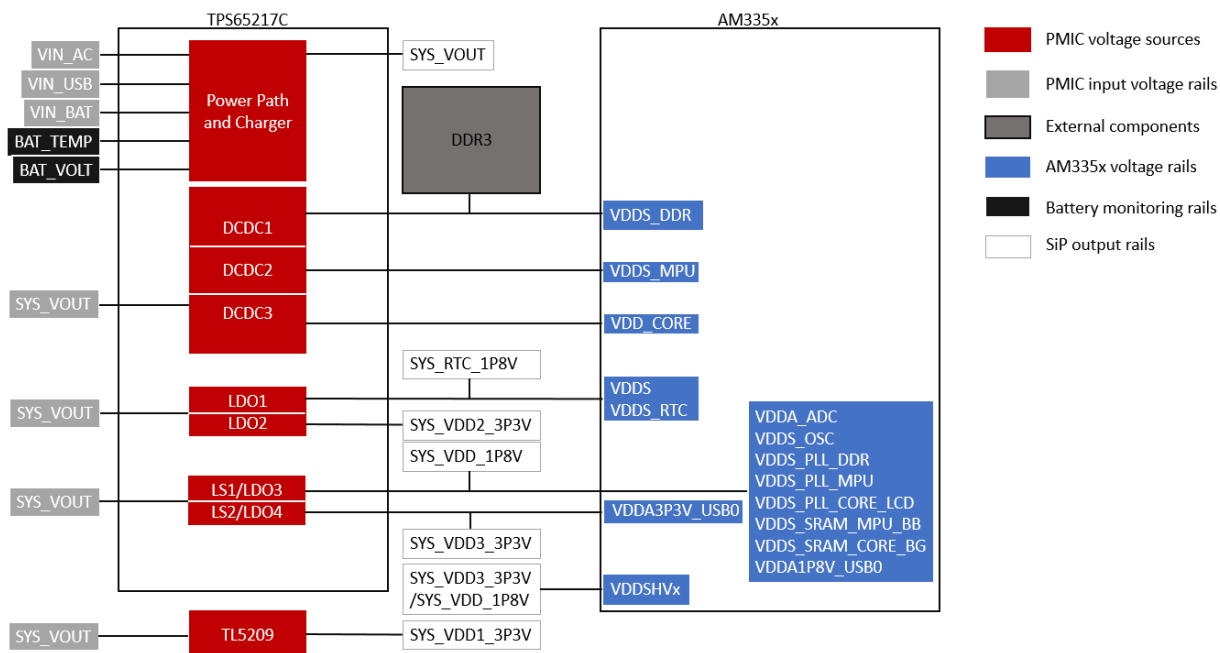


Figure 39: OSD335x-SM Power System Block Diagram

Table 2 shows the voltage output levels of each of the voltage sources which can be measured on the respective test point. Each of the output voltages can be changed dynamically using I2C commands when the PMIC is in active mode.

Table 2: PMIC Output Voltages

TPS65217C Voltage Source	OSD335x-SM Voltage rail	Test Point Label	Voltage (V)
DCDC1	VDDS_DDR	TP1	1.5
DCDC2	VDD_MPU	TP3	1.1
DCDC3	VDD_CORE	TP2	1.1
LDO1	SYS_RTC_1P8V	TP14	1.8
LDO2	SYS_VDD2_3P3V	TP13	3.3
LDO3	SYS_VDD_1P8V	TP15	1.8
LDO4	SYS_VDD3_3P3V	TP16	3.3

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



4.2 Additional Functions of the PMIC

The PMIC also performs some tertiary functions. A few of the key functions are listed below. For a complete explanation of all the functions provided by the TPS65217C PMIC refer to the datasheet (<http://www.ti.com/product/TPS65217/datasheet>).

1. The PMIC retrieves the processor from OFF or SLEEP mode upon detecting a falling edge on PWR_BUT signal. It also power cycles the processor if PWR_BUT signal is held low for more than 8 seconds by holding the power button S2 down.
2. It provides an active low wake up signal (PMIC_NWAKEUP) which is de-asserted when a wakeup event is detected and is connected to EXT_WAKEUP pin of the processor.
3. The PMIC has an interrupt pin (PMIC_NINT) to signal an event or fault condition to the processor via EXTINT pin. The pin is released when the processor reads the INT register.
4. PMIC provides a linear charger for Single Cell Li-Ion batteries and allows charging of the battery and powering of the system at the same time. More information on this feature can be obtained from the datasheet.
5. The TPS65217C provides protection to AM335x and itself in the event of catastrophic situations like an unexpected short or excessive current leakage. Please refer to the OSD335x Power application note (https://octavosystems.com/app_notes/osd335x-power-application-note/) for information on how to design the power system for an application with OSD335x.

4.3 Input Power

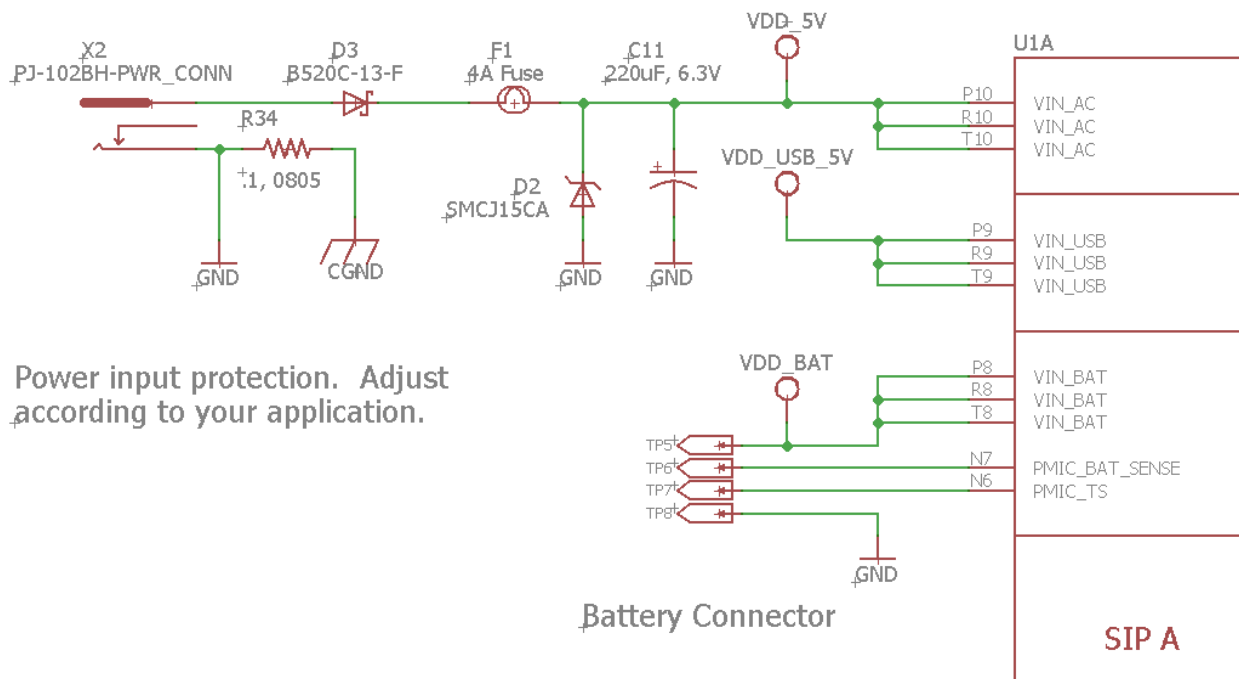


Figure 40: OSD3358-SM-RED V3 Input Power Circuitry

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021

Figure 40 shows power inputs of the OSD3358-SM-RED platform V3. A combination of the AC, USB and battery inputs can be used to power the board. To protect the board from power surge, reverse polarity and transient input voltage behavior, the AC adapter input signal goes through an input protection circuit shown in Figure 40.

The following figure shows power inputs for OSD3358-SM-RED platform V4. In addition to the protection circuitry in V3, a load switch TPS22958 (U26) and Voltage monitor BU4240 (U4) are used for VIN_AC input. The voltage monitor monitors the voltage input VIN and controls the load switch to deactivate VIN_AC when VIN decreases below 4.2V. This acts as protection against a brownout condition on VIN_AC. For more information, please see <https://www.ti.com/lit/an/slva901/slva901.pdf>.

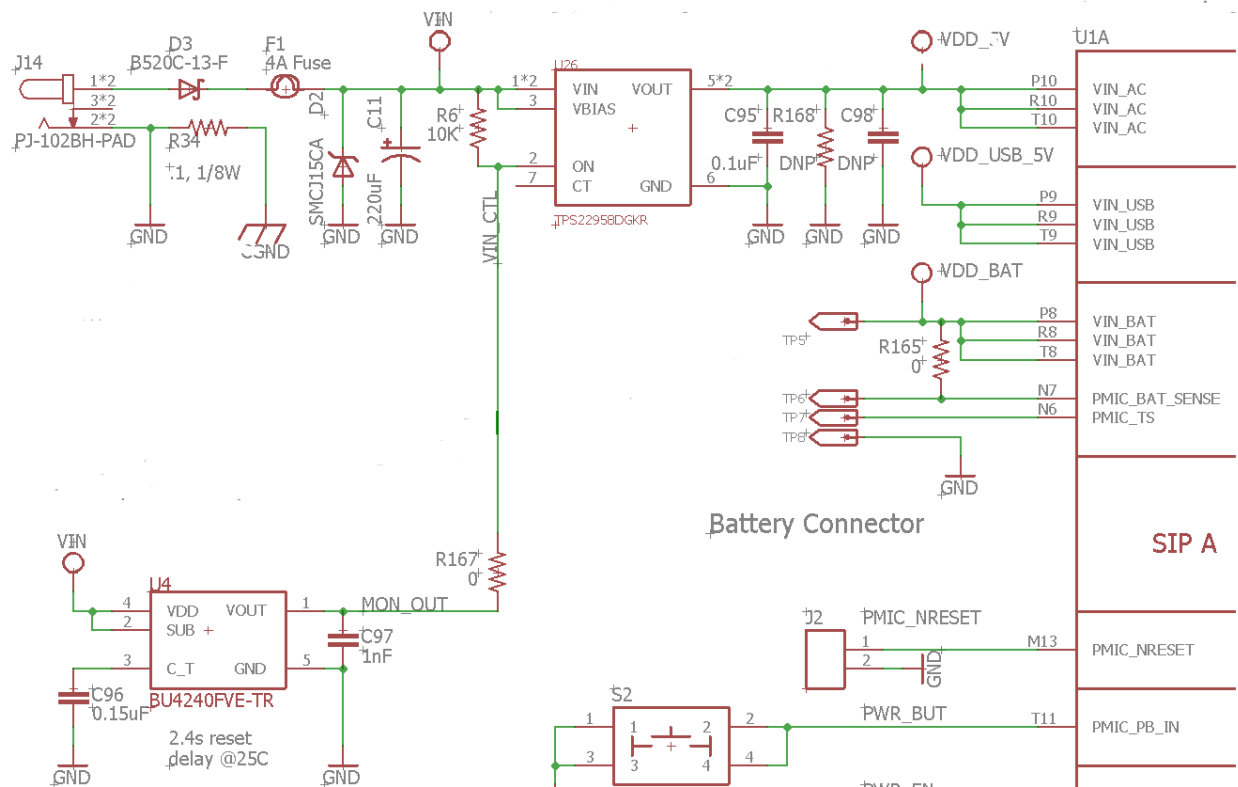


Figure 41: OSD3358-SM-RED V4 Input Power Circuitry

To power the board using a 1S Li-Po battery, the following hook up is required.

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



Table 3: OSD3358-SM-RED Battery Connections

Pin	Function	Hook up
VIN_BAT (TP5)	Battery input	Positive battery terminal
BAT_SENSE (TP6)	Battery voltage sense input	Positive battery terminal
PMIC_TS (TP7)	Battery temperature sense input	NTC thermistor to sense battery temperature. Works with 10k and 100k thermistors
GND (TP8)	Digital system ground	Battery negative terminal

The connections are shown in the following Figure:

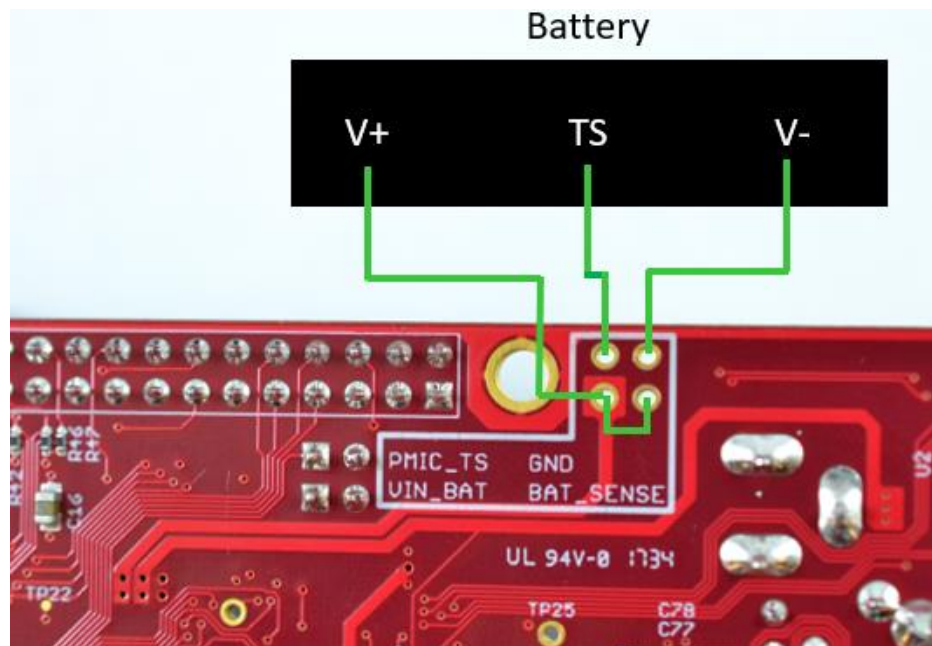


Figure 42: Battery Connections

NOTE: Due to the input voltage requirements of the TL5209 LDO that supplies SYS_VDD1_3P3V of the OSD335x-SM, the voltage of the battery that will power the board needs to be greater than 3.7 V. If the voltage of the battery falls below this threshold, SYS_VDD1_3P3V begins to go out of regulation resulting in some of the on-board components powered by this voltage rail not working in spec. This includes the SD card slot and the eMMC, which are the main boot sources for the board.

4.4 PMIC-AM3358 Interface

As discussed in sections 4.1 and 4.2, functions of the PMIC include: proper power up of the processor, monitoring of the AM3358 power rails, interrupt generation and reset management of the processor. To perform these functions, the PMIC and AM335x must be connected. These connections can be found in the datasheet (<https://octavosystems.com/docs/osd335x-sm-datasheet/>). Figure 41 shows some of these connections in OSD335x-SM-RED schematics.

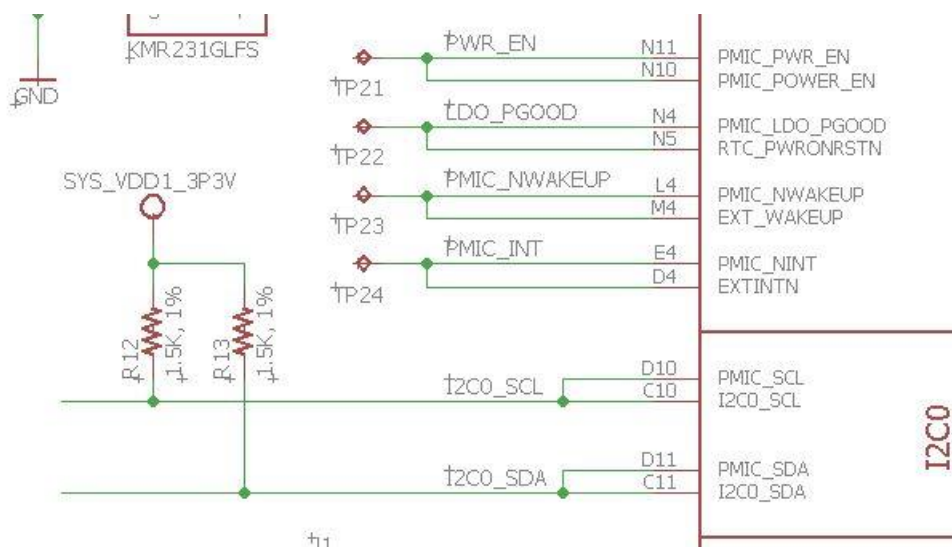


Figure 43: OSD3358-SM-RED PMIC - AM3358 Interface

NOTE: I2C0 lines are internally pulled up to 3.3 V using 4.7K resistors. In addition, 1.5K resistors were used to pull up the I2C lines in the RED board. Please see: (<http://www.ti.com/lit/an/slva689/slva689.pdf>) for more information on how to calculate I2C pull ups.

NOTE: Test points TP21, TP22, TP23 and TP24 are provided to monitor these signals for debug purposes.

Figure 44 shows the reset circuit in the design schematic. The PMIC's PGOOD output controls the PWRONRST input of the processor. So, the processor is held in reset until PMIC_PGOOD comes up which means all the power rail outputs of the PMIC are up and stable.

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021

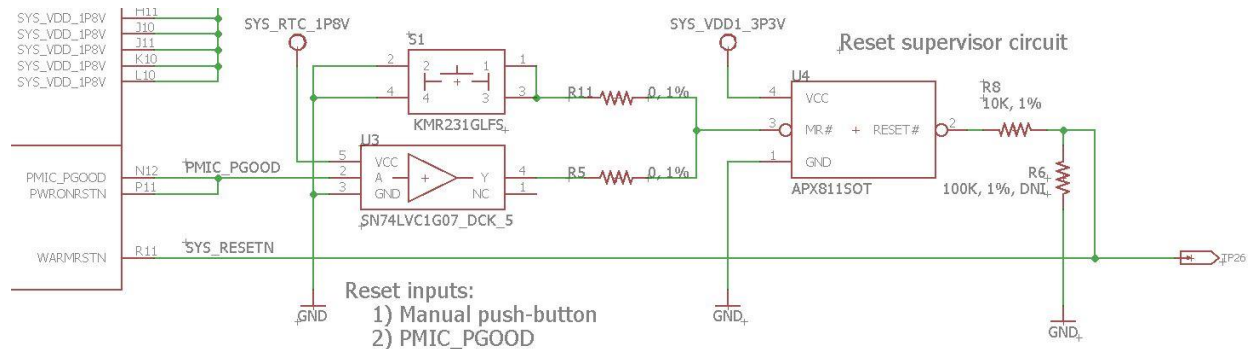


Figure 44: OSD3358-SM-RED Reset Circuit

NOTE: PWRONRSTN holds all the processor systems under reset except for RTC module. The RTC module has its own power-on-reset input, the RTC_PWRONRSTN pin of OSD335x.

Figure 44 also shows the reset mechanism for the WARMRSTN input of the processor. WARMRSTN input is a soft reset and does not affect most of the power/boot systems of the processor unlike PWRONRSTN. While PMIC_PGOOD and PWRONRSTN operate on 1.8V logic, WARMRSTN is 3.3V logic input. Therefore, an open drain buffer U3 is used to drive the WARMRSTN input. WARMRSTN can also be pulled low using the reset switch S1. As the switch output is susceptible to ground bounce, causing multiple or partial reset, a reset supervisor U4 is used in between the output of the switch and WARMRSTN input. For more information on the reset circuit and the functions of each signals, please refer to the 'OSD335x Design Tutorial series' 'OSD335x Reset Circuitry' article (https://octavosystems.com/app_notes/osd335x-design-tutorial/bare-minimum-boot/reset-circuitry/)

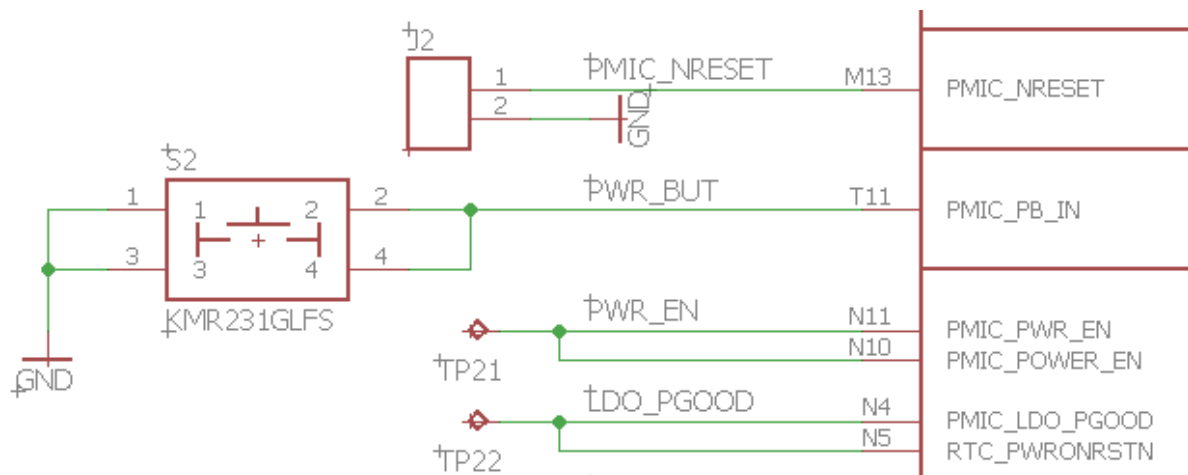


Figure 45: OSD3358-SM-RED Power Button

The OSD335x-SM-RED board also provides a power button S2 connected to the PMIC_PB_IN (active low push button input of PMIC). PMIC_PB_IN has the following functions:

- The PMIC will be powered up from OFF or SLEEP mode upon detecting a falling edge on PWR_BUT signal (hold the power button, S2 down).
- PMIC is power cycled/reset when PWR_BUT (Switch S2) is held low for more than 8 s. All rails will be shut down by the sequencer and all register values are reset to their default values. Rails not controlled by the sequencer are shut down immediately. After 1 sec of RESET state, the board powers up again.
- If the PMIC_IN_PB_IN pin is kept low for an extended amount of time, the device will continue to cycle between ACTIVE and RESET state, entering RESET every 8 seconds.

NOTE: Holding the power button low for 8 seconds on some boards will power off the system instead of power cycling it depending on current draw.

Figure 45 also shows a 2-pin header J2. Pin 1 of the jumper is connected to PMIC_NRESET and pin 2 is connected to GND. The PMIC can be put in reset state by using a jumper to connect the 2 pins of the header. The PMIC will remain in reset until the PMIC_NRESET input is released from the pull down to GND and for a minimum of 1 sec.

4.5 Clamping Circuit

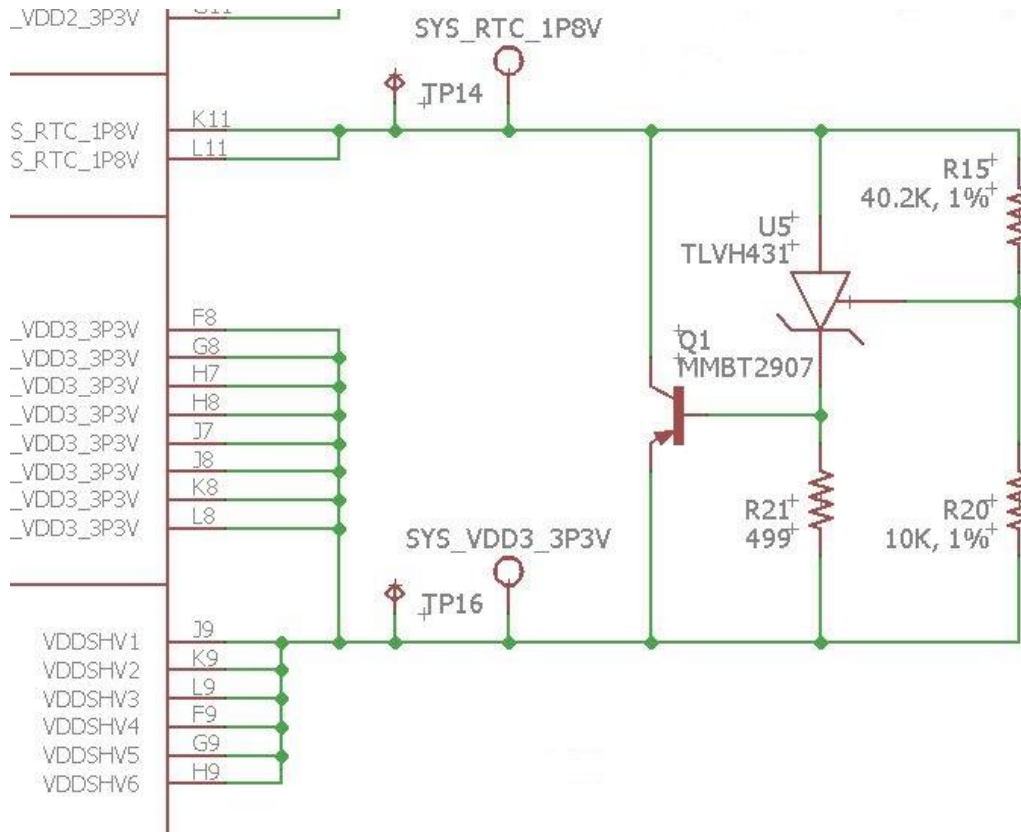


Figure 46: OSD3358-SM-RED Clamping Circuit

Figure 46 shows a clamping circuit designed to prevent a PMIC shutdown fault. The AM335x datasheet requires that the voltage difference between the power rails VDDS (1.8V) and VDDSHVx [1-6] (3.3V) of the AM335x processor be less than 2V during the entire power-down sequence. A detailed description of the clamping circuit and its operation can be found in “The OSD335x Reference Design Lesson 1 Power Circuitry Part 4: Clamping Circuit”

(https://octavosystems.com/app_notes/osd335x-reference-design-lesson-1-power-circuitry-part-4-clamping-circuit/).

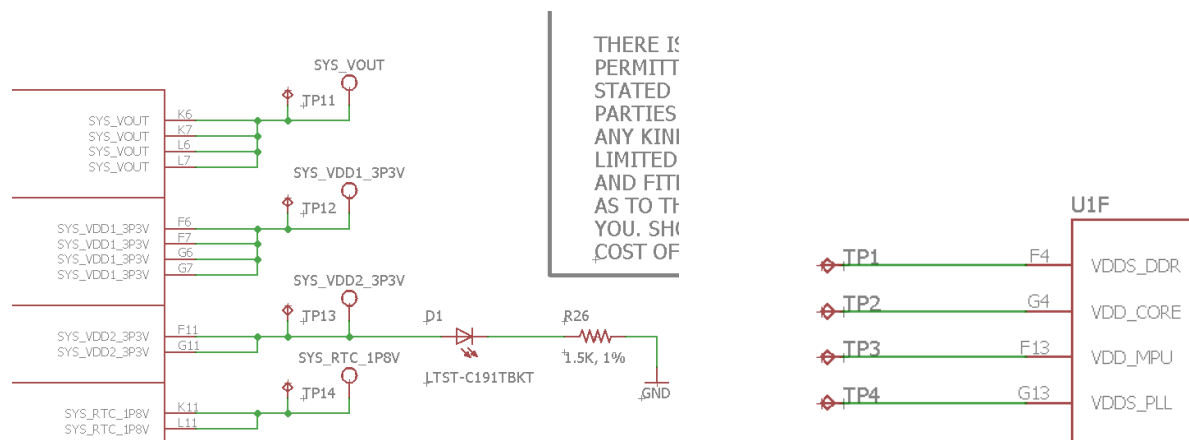


Figure 47: OSD3358-SM-RED Output Power Rails

Figure 47 shows the main power rails of the PMIC connected to test points on the OSD3358-SM-RED board schematic. This facilitates debug of the hardware system. The test points allow probing of the power rails and verification of the boot up sequence. To give an initial indication that the board is booting up, LED D1 is connected to SYS_VDD2_3P3V, since this power output will come up as part of the boot sequence.

4.6 Clock Input to the Processor

OSD335x-SM has 2 clock inputs: OSC0, the main system clock, at 24MHz and OSC1, the Real-Time Clock (RTC) clock, at 32kHz.

4.6.1 OSC0:

This input is the main clock source for the processor and is the reference clock to all subsystems except RTC. This clock input is a requirement for all designs with the OSD335x-SM. For the OSD3358-SM-RED, this clock is 24MHz.

4.6.2 OSC1:

This input is the clock source for Real-Time Clock (RTC) system of the processor. The clock input for OSC1 should be 32.768KHz generated by a single ended clock source or crystal oscillator. This clock input is optional and can be removed if the RTC subsystem of the processor is not used.

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021

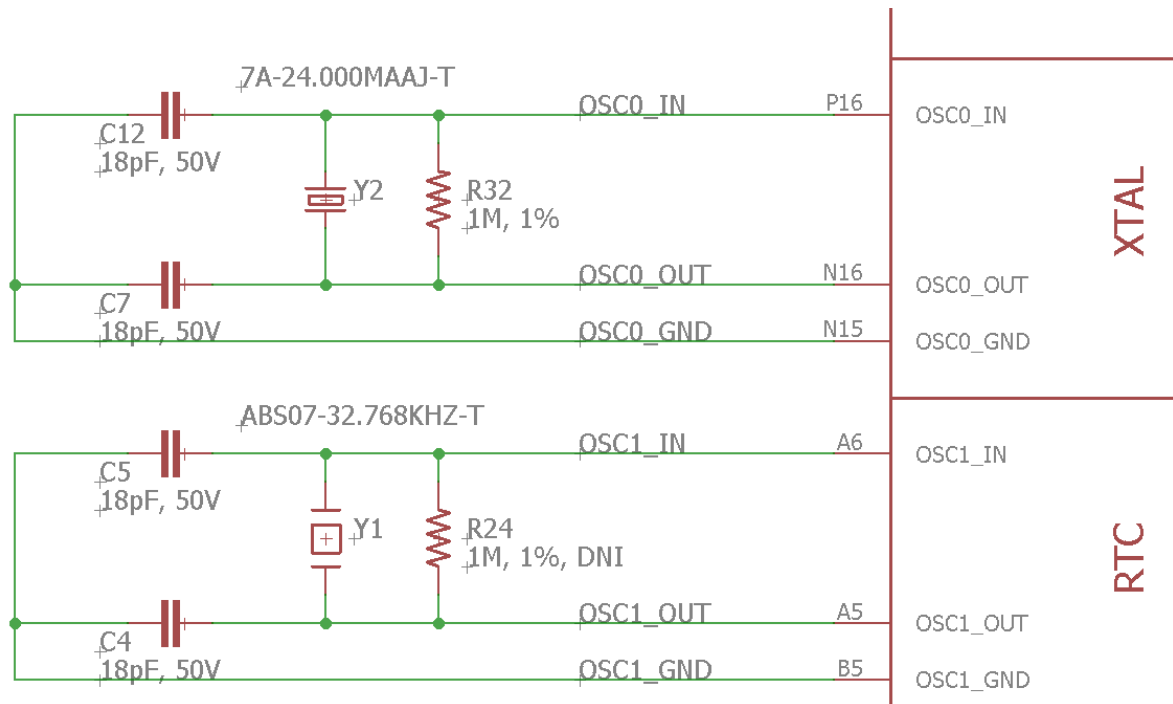


Figure 48: OSD3358-SM-RED Clock Circuitry

Figure 48 shows the circuit for OSC0 and OSC1 clock inputs of OSD335x-SM in the OSD3358-SM-RED Board. More information on the inputs and circuit design can be found in AM335x TRM (<http://www.ti.com/lit/ug/spruh73p/spruh73p.pdf>), AM335x datasheet (<http://www.ti.com/lit/ds/sprs717j/sprs717j.pdf>) and Reference Design Clock circuitry article (https://octavosystems.com/app_notes/osd335x-design-tutorial/bare-minimum-boot/clock-circuitry/).

NOTE: The version of the PMIC used within the OSD335x-SM is not capable of supporting RTC-only low power mode. However, the internal RTC LDO is still used to power the RTC module by pulling RTC_KALDO_ENN (M16) to GND through a 10K resistor.

4.7 Boot Configuration

The ROM code checks the SYSBOOT configuration pins to communicate information to the processor such as, the OSC0 clock frequency, system boot source and boot sequence of the board. The boot configuration functionality is multiplexed with the LCD interface pins of the processor. The SYSBOOT pins are latched on the rising edge of PWRNONRSTN signal. After boot, the pins can be used for LCD interfacing or other multiplexed functions. Table 26-7 of the TRM (<http://www.ti.com/lit/ug/spruh73p/spruh73p.pdf>) describes the function of each boot configuration pin. For the OSD3358-SM-RED Platform, the following boot configuration was chosen.

Table 4: OSD3358-SM-RED SYSBOOT Pin Functions

Pins	Value	Function
SYSBOOT[15:0]	0x401C	
SYSBOOT[15:14]	01b	Input crystal frequency of OSC0 input is 24MHz
SYSBOOT[13:12]	00b	Reserved
SYSBOOT[11:6]	000000b	Don't care for ROM code for the OSD3358-SM-RED boot modes
SYSBOOT[5]	0b	CLKOUT1 disabled
SYSBOOT[4:0]	See Table 5	

SYSBOOT[4:0] controls the boot sequence. The OSD3358-SM-RED has the ability to select multiple boot sequences. The boot button S3 can be used to modify SYSBOOT[2] to change the boot sequence of the processor. The boot button S3 must be pressed before power is applied so that the correct value can be latched during power up. The boot sequence changes are described in the following table allowing the processor to boot from the SD card slot even when there is a valid image in the eMMC.

Table 5: OSD3358-SM-RED Boot Order

System change	Default	Boot Button (S3) Pressed
SYSBOOT[4:0]	11100b	11000b
Boot sequence priority	• MMC1 • MMC0 • UART0 • USB0	• SPI0 • MMC0 • USB0 • UART0

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



Figure 49 shows the boot configuration circuit in the OSD3358-SM-RED Board schematics.

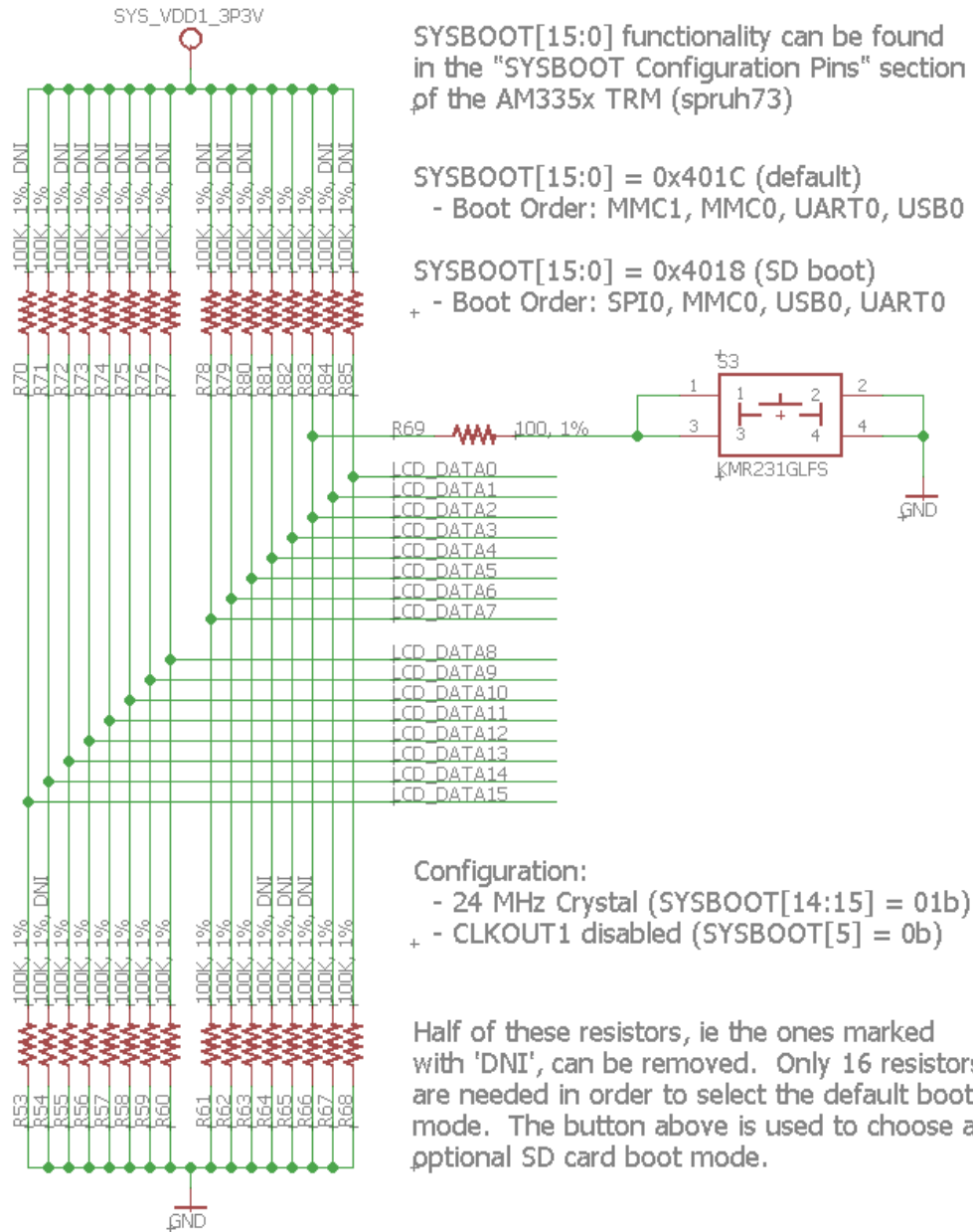


Figure 49: OSD3358-SM-RED Boot Configuration

4.8 UART0 Interface

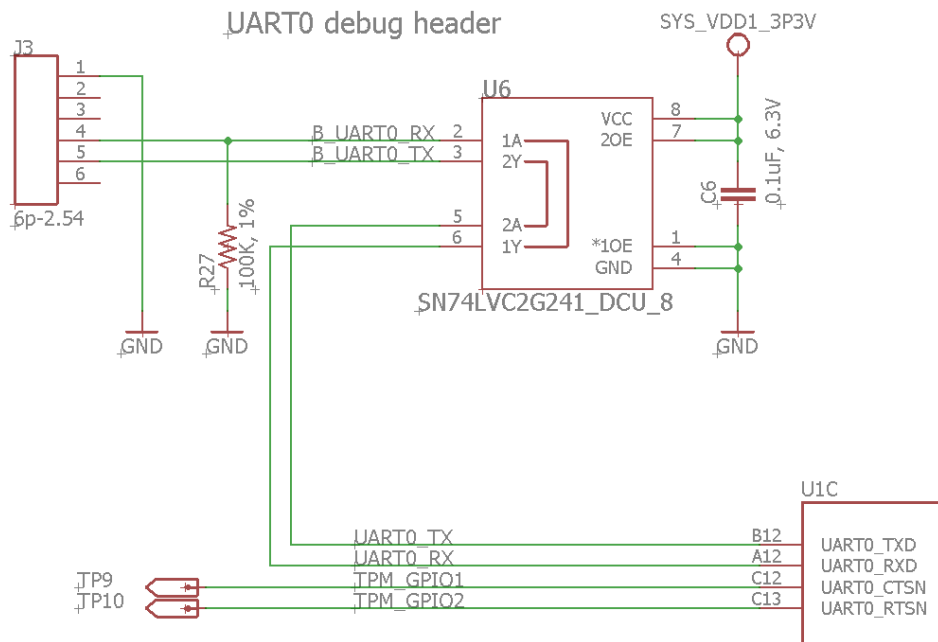


Figure 50: OSD3358-SM-RED UART Interface

Figure 50 shows UART0 interface of OSD3358-512M-BSM connected to header J3 on the board through a dual buffer. This port can be used for debug by monitoring boot messages the processor sends through the UART0 interface. A USB to TTL adapter may be required to communicate to the processor through this port using a laptop/computer.

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



4.9 eMMC

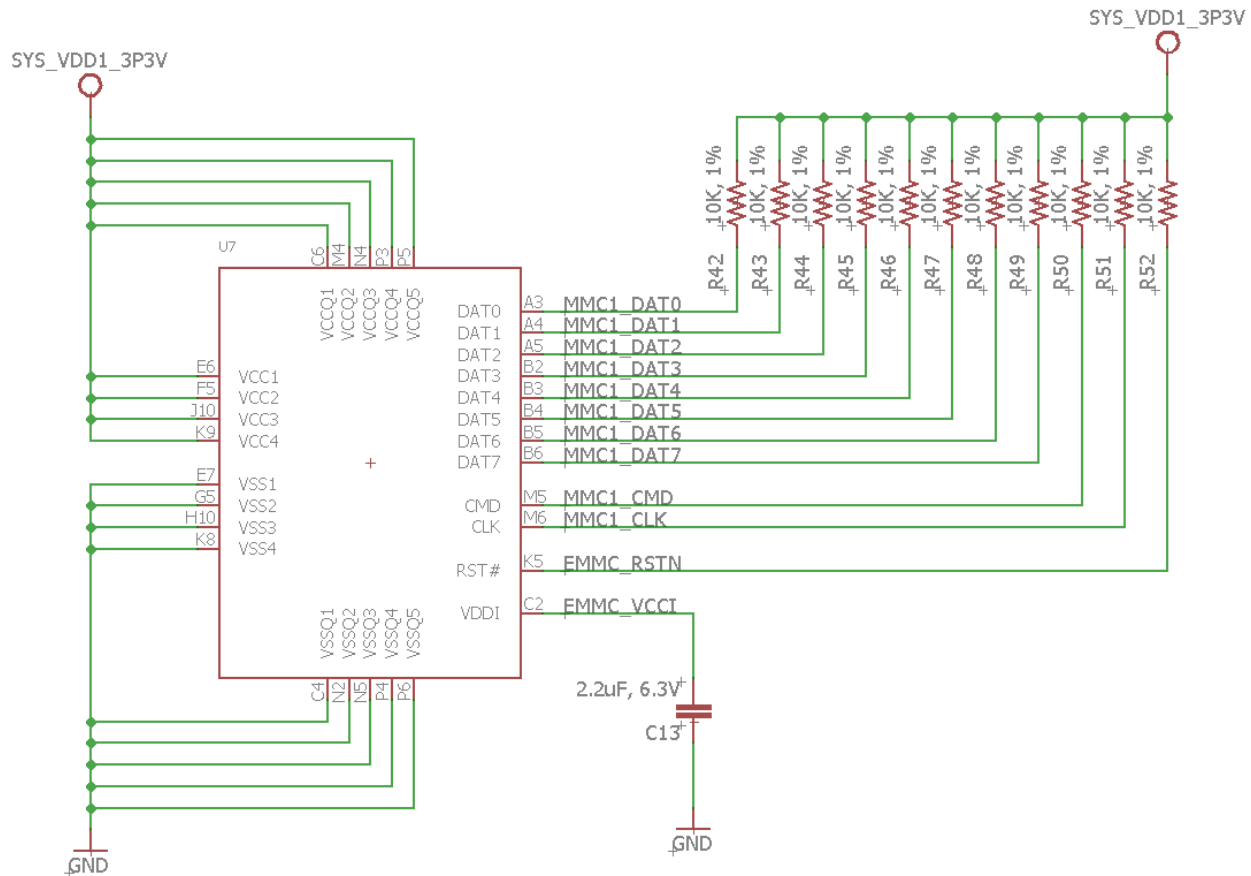


Figure 51: OSD3358-SM-RED eMMC Circuitry

Figure 51 shows eMMC U7 connected to the processor through MMC1 interface. As discussed in section 3.5, eMMC is the primary boot source for the processor. The Linux kernel and file system needed to boot Linux are stored in here. The ROM code reads the second stage bootloader in the eMMC and passes control to it. The ROM only supports 4-bit mode. But, after initial boot, the eMMC can be interfaced in 8-bit mode. MMC1_CLK is the clock input and MMC1_CMD is the command input to the eMMC. Pull up resistors are used on all the signals to increase rise time on the signals and compensate for any capacitance on the board. More information on booting of the board and board software can be found in Section 5.

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



4.11 JTAG

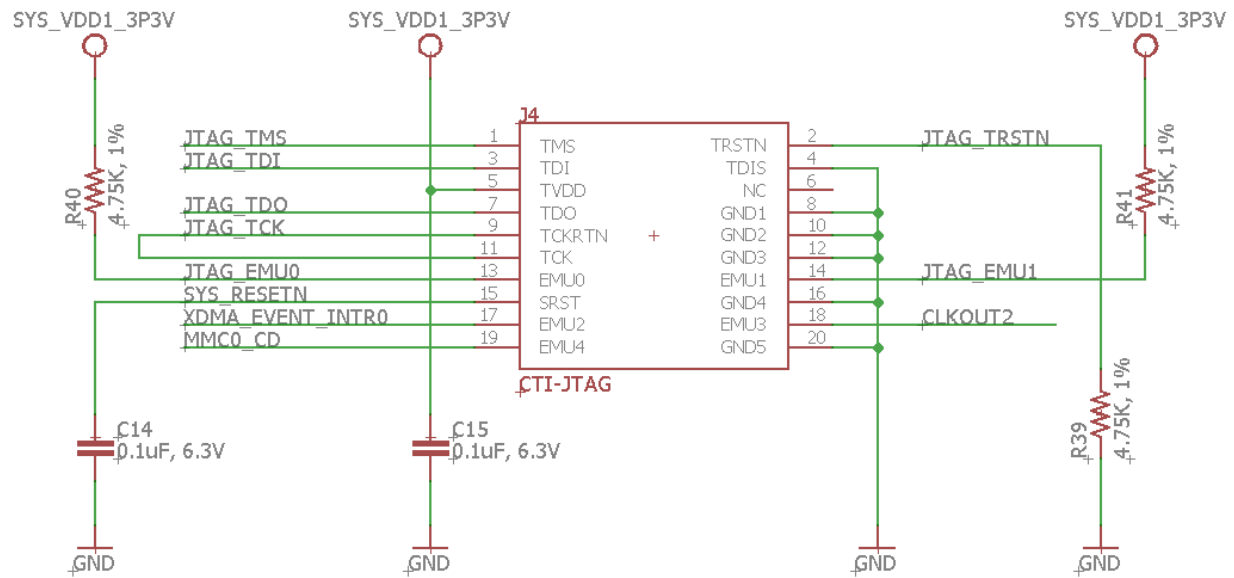


Figure 53: OSD3358-SM-RED JTAG Circuitry

A JTAG header footprint J4 is provided on the board to facilitate software development and debug using JTAG emulators. More information on how to use this is provided in Section 3.7.

4.12 ADC

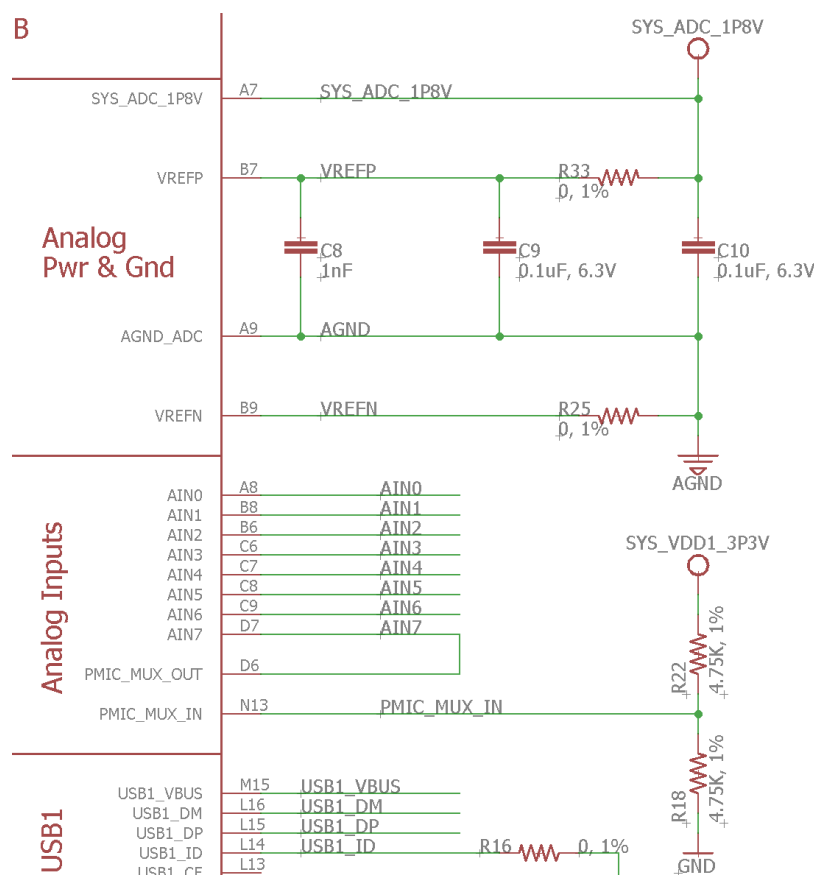


Figure 54: OSD3358-SM-RED ADC Circuitry

Figure 54 shows ADC circuit on the OSD3358-SM-RED platform schematics. On the RED Board, the positive reference voltage VREFP of the ADC is connected to SYS_ADC_1P8V and the negative reference voltage VREFN of the ADC is connected to AGND (Analog GND). This provides a 1.8V range for the 12-bit ADC module in the OSD335x-SM. The external resistors and capacitors are used to filter the noise on the reference voltages.

Of the total 8 analog inputs (AIN0 – AIN7), seven (AIN0 – AIN6) are brought out to expansion headers on the board and can be used. AIN7 is connected to PMIC_MUX_OUT to monitor the internal voltages of the PMIC. The PMIC has an analog multiplexer that allows critical voltages of the system to be monitored. An additional PMIC_MUX_IN input can be used to input any critical system level voltage that needs to be monitored. The PMIC's I2C interface can be used to select the voltage that needs to be monitored and AIN7 can be used to read the voltage. More information on the analog multiplexer can be found in the PMIC datasheet (<http://www.ti.com/lit/ds/symlink/tps65217.pdf>).

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



4.13 USB Client Port

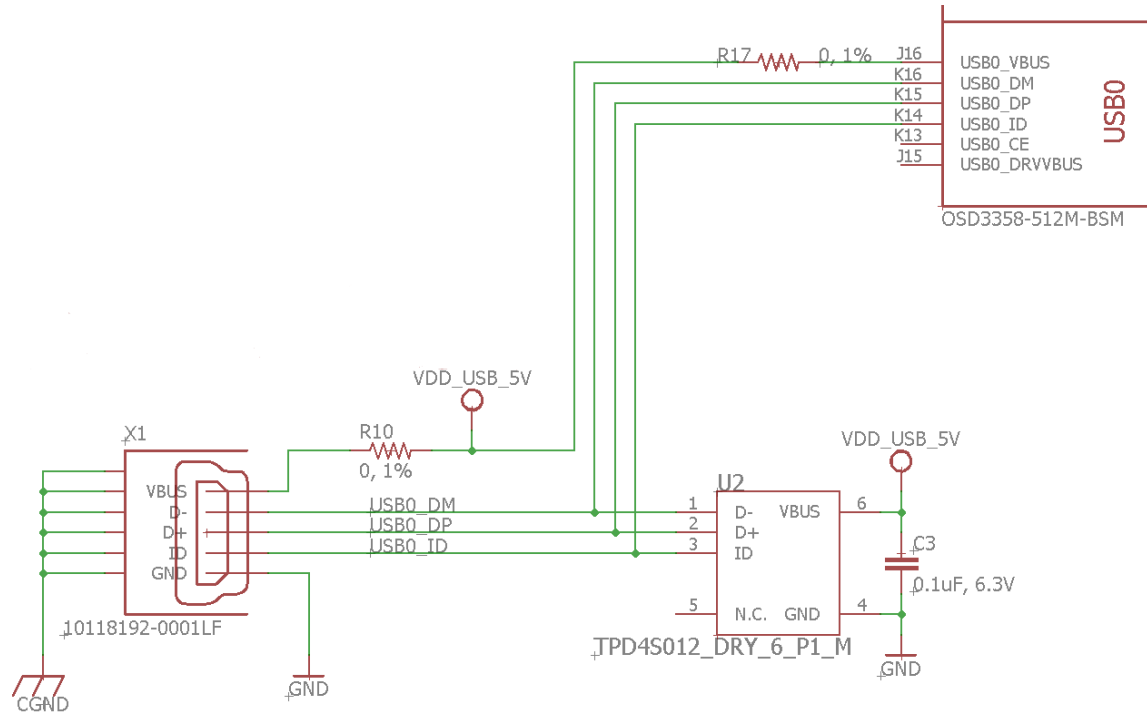


Figure 55: OSD3358-SM-RED USB Client Circuitry

Figure 55 shows USB client port X1 connected to USB0 interface of the processor in the OSD3358-SM-RED Board schematic. As described in Section 2, this port can be used to power the board through a laptop/computer's USB port. A secure shell program can be used to get to the Linux command prompt from a laptop/computer. U2 provides ESD protection for the USB signals.

4.14 Four (4) port USB 2.0 Hub

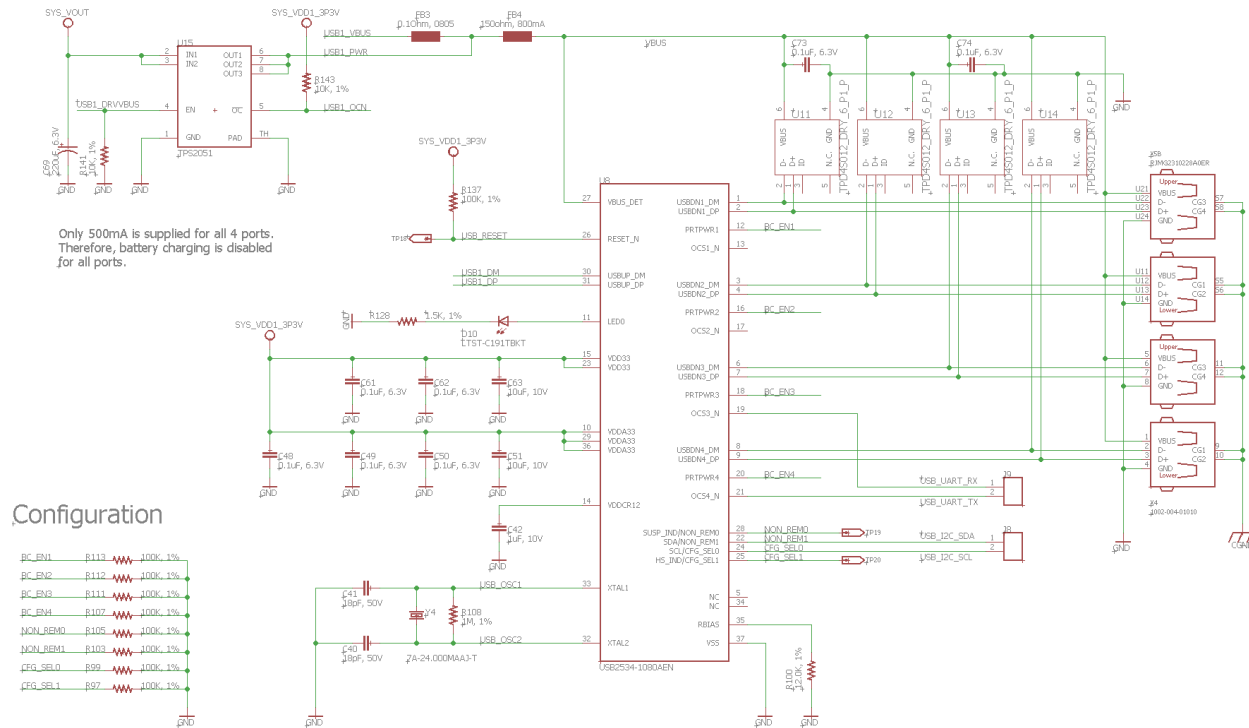


Figure 56: OSD3358-SM-RED USB 2.0 Host Hub Circuitry

Figure 56 shows the 4 port USB hub U8, 4 USB ports part of X4 and X5 connectors and a power distribution switch U15 that combine to form the USB host interface for the OSD3358-SM-RED Board. The USB hub chip converts USB1 interface of the processor into 4 independent USB connection ports each of which have ESD protection and can be used to connect USB peripherals simultaneously. The hub chip has UART and I2C interfaces which can be used to configure it. Battery charging is disabled on all the ports because of the 500mA limit that can be supplied from U15. A 24 MHz crystal is used to supply clock to the hub controller USB2534 (U8) that is powered by SYS_VDD1_3P3V. The four ports can be used simultaneously, however the combination of the attached peripherals cannot exceed the 500mA current draw limit.

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



4.15 Ethernet

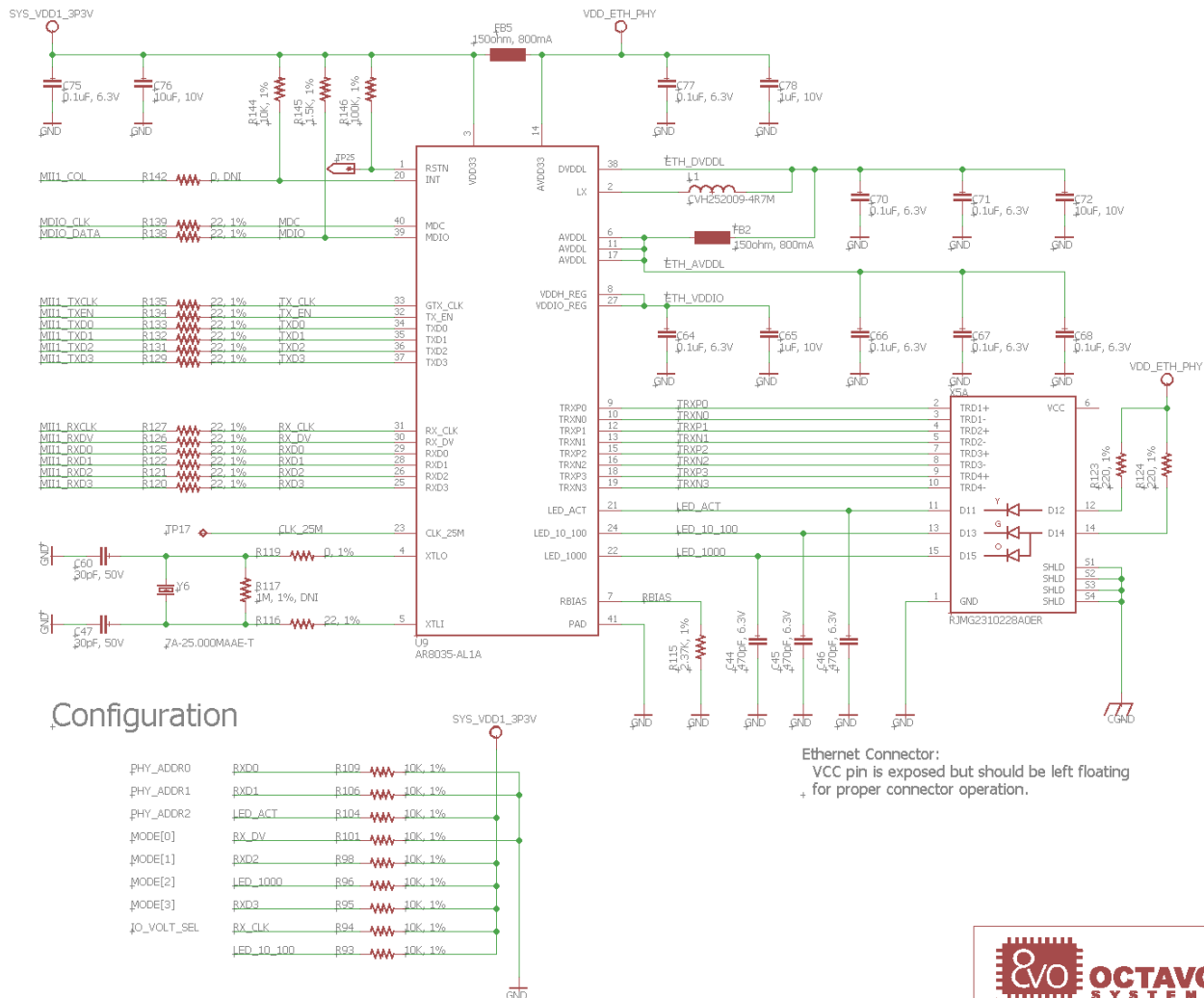


Figure 57: OSD3358-SM-RED Ethernet Circuitry

The OSD3358-SM-RED Platform is capable of Gigabit ethernet with the AR8035 Gigabit Ethernet PHY connected to OSD335x-SM via RGMII1 port. The PHY interface with the processor consists of 4 data lines for transmission and reception along with clock and control lines for each. MDIO interface is used by the processor for instruction/configuration of the PHY. A 25MHz crystal oscillator, Y6, supplies clock to the PHY. X5 contains an ethernet connector with integrated magnetics with 2 LEDs indicating mode of operation.

More information on ethernet interface can be found at the following resources:

1. AR8035-AL1A Datasheet
([https://media.digikey.com/pdf/Data%20Sheets/CSR%20PDFs/AR8035_DS_\(Atheros\)_Mar2011.pdf](https://media.digikey.com/pdf/Data%20Sheets/CSR%20PDFs/AR8035_DS_(Atheros)_Mar2011.pdf))
2. RJMG2310228A0ER Datasheet
(https://media.digikey.com/pdf/Data%20Sheets/Amphenol%20PDFs/RJMG2310228A0ER_Dwg.pdf)

4.16 HDMI Interface

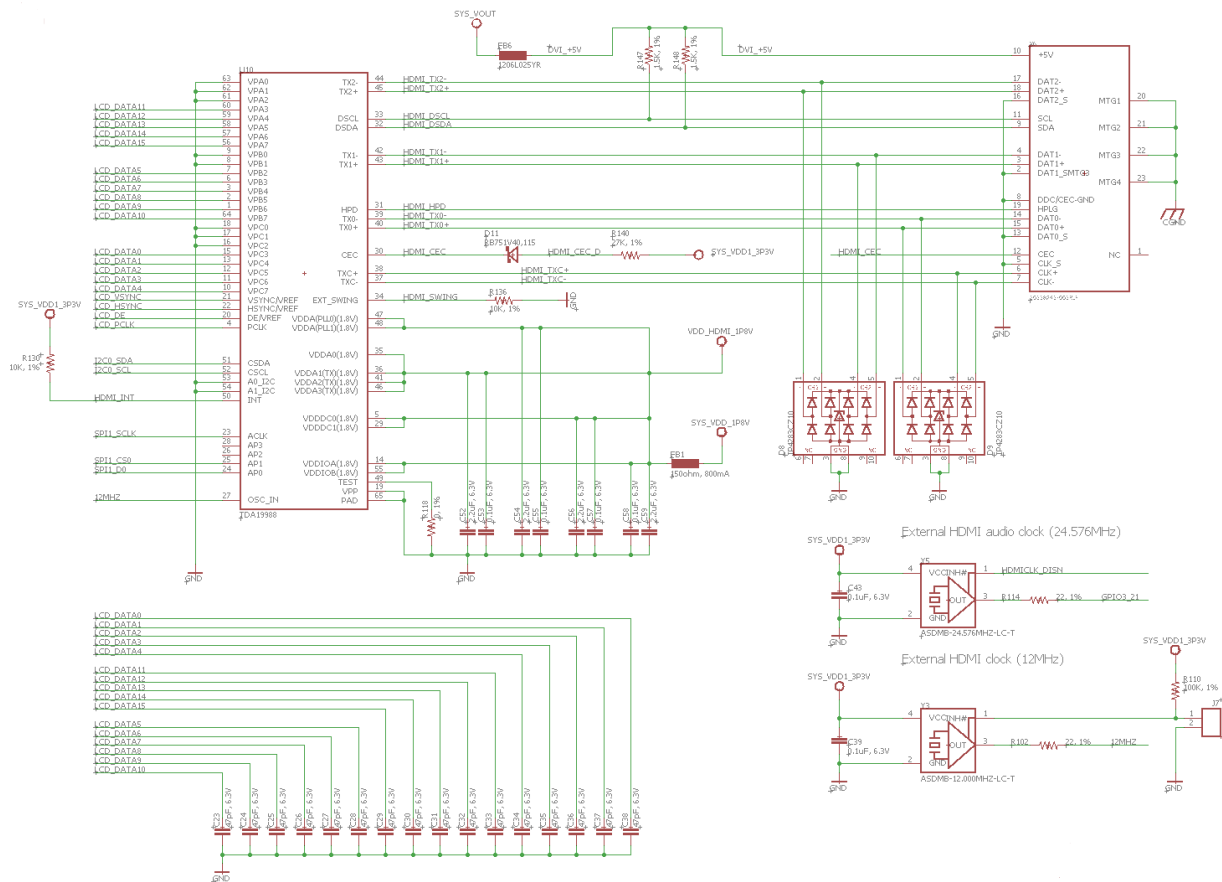


Figure 58: OSD3358-SM-RED HDMI Circuitry

The OSD3358-SM-RED Platform can drive an HDMI monitor with the on-board TDA1998 HDMI framer U10 and micro-HDMI connector X6. The framer converts the processor 16-bit LCD interface to drive HDMI monitors. The framer is powered by SYS_VDD1_3P3V and can be configured using the I2C0 interface. A 12 MHz oscillator Y3 supplies the main clock input to the framer while the 24.576MHz crystal provides audio clock. The interface between the micro-HDMI connector X6 and the HDMI framer U10 is ESD protected using D8 and D9.

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



4.17 9-Axis IMU

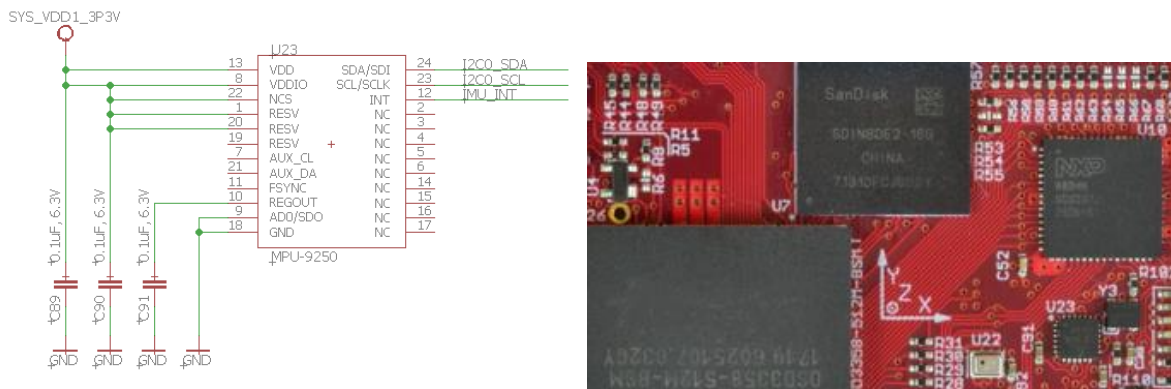


Figure 59: OSD3358-SM-RED V3 IMU Circuitry

Figure 59 shows a 9-Axis IMU U23 powered by SYS_VDD1_3P3V rail and connected to the processor on the I2C0 interface in board revision V3. This sensor sits on the horizontal axis of the board that cuts the board in half. The chip is located at coordinates (3485 mil, 1075 mil) from the bottom left of the board oriented in the same way as shown in Figure 59. It is comprised of a 3-axis gyroscope, a 3-axis accelerometer and a 3-axis magnetometer making the device a 9-axis motion tracking sensor. Each of the axes have a 16-bit ADC to digitize the sensor outputs.

Figure 60 shows the 9-Axis IMU U23 used in board revision V4. It has the same specifications as MPU-9250. The IO voltage of ICM-20948 is 1.8V. So, a bidirectional voltage translator (U25) is used to interface between I2C0 interface of OSD335x and I2C interface of the IMU.

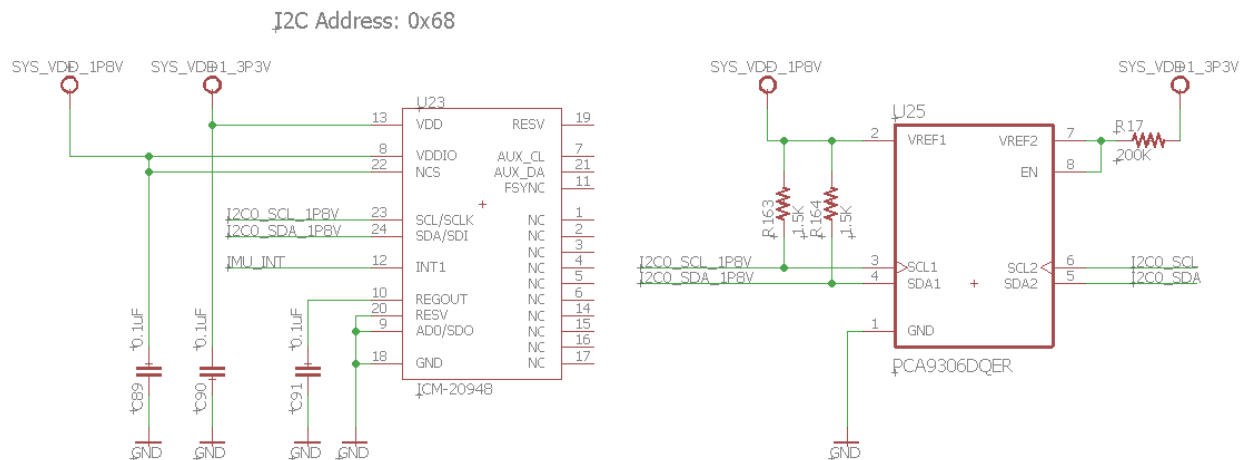


Figure 60: OSD335-SM-RED V4 IMU Circuitry

4.18 Temperature Sensor Hub

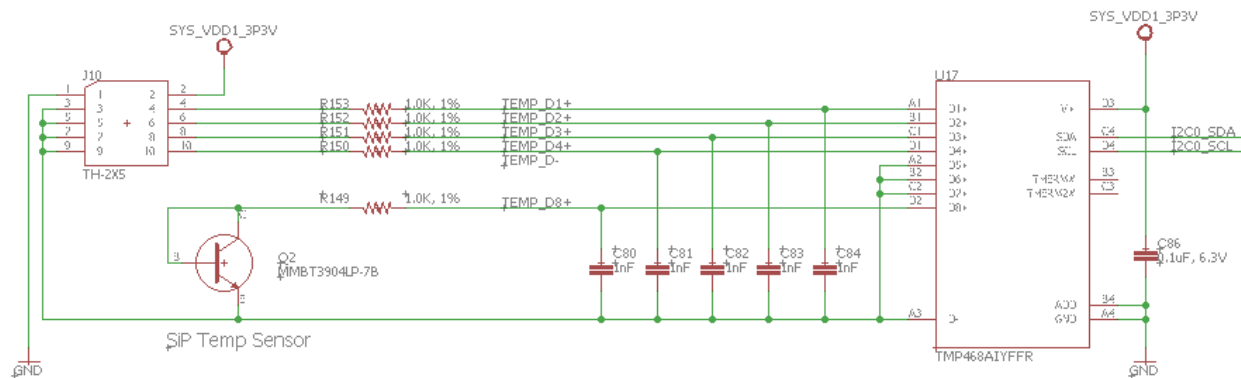


Figure 61: OSD3358-SM-RED Temperature Sensor Hub Circuitry

The OSD3358-SM-RED Platform also has a temperature sensor hub (shown in Figure 61) sitting on the I2C0 interface of the processor. The hub has 8 sensor inputs, out of which 4 (D1 – D4) are brought out to header J10. An additional input D8 is used to measure temperature on board near the SiP using Q2. The 3 remaining inputs are not routed out due to the size of the package of the sensor hub. The hub is powered by SYS_VDD1_3P3V and can be used to monitor thermal behavior across multiple locations of the system. More information setting up remote measurements can be found in the temp sensor datasheet (<http://www.ti.com/lit/ds/symlink/tmp468.pdf>). The sensor has a resolution of 13 bits (0.0625C) and can measure temperatures ranging from -40C to 125C.

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



4.19 Barometer

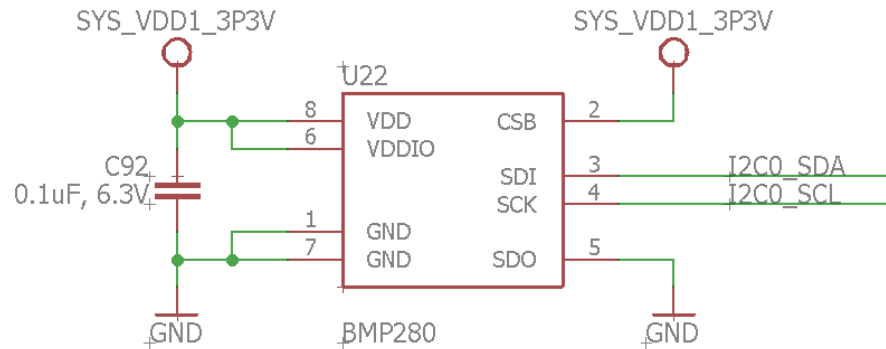


Figure 62: OSD3358-SM-RED V3 Pressure Sensor Circuitry

Figure 62 shows BMP280 (U22) connected to the processor on the I2C0 interface in board revision V3. This is a barometric pressure sensor with a range of 300 – 1100 hPa. Along with an additional temperature sensor, it has noise filtering circuitry that can be used to reject short term disturbances. The sensor can measure with a 1m altitude accuracy.

Figure 63 shows LPS22HB (U22) connected to the processor in I2C0 interface in board revision V4.

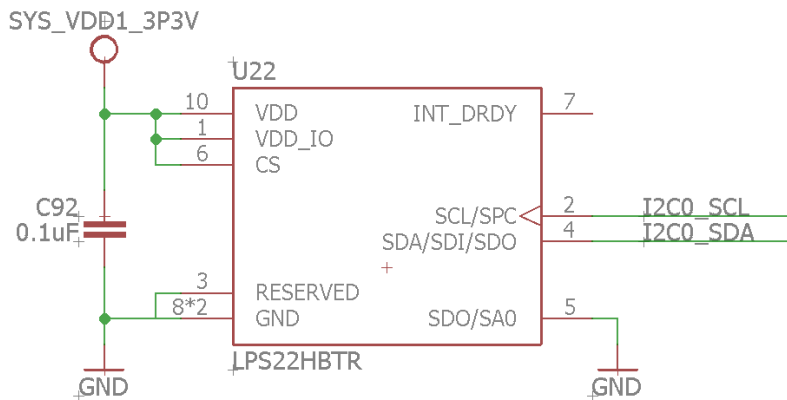


Figure 63: OSD3358-SM-RED V4 Pressure Sensor Circuitry

4.20 Trusted Platform Module (TPM)

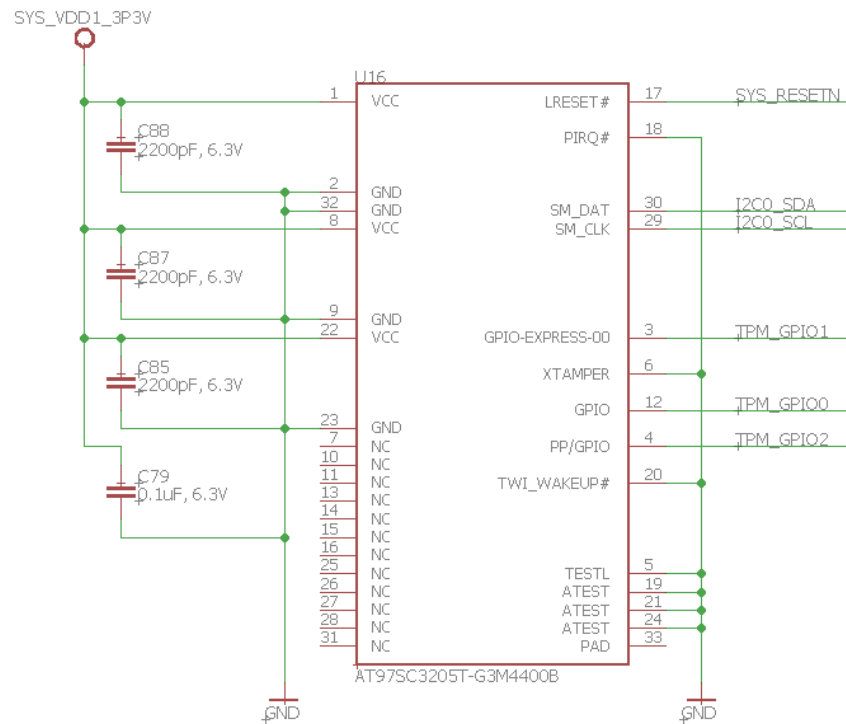


Figure 64: OSD3358-SM-RED TPM Circuitry

A Trusted Platform Module (TPM) is present on I2C0 interface of the processor. It is powered by SYS_VDD1_3P3V and can be used to develop security applications for the processor. Coupled with the NOR Flash device on SPI0 bus, the TPM can be used to implement secure boot and other secure application environments. The board does not come with any security applications out of the box.

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



4.21 EEPROM

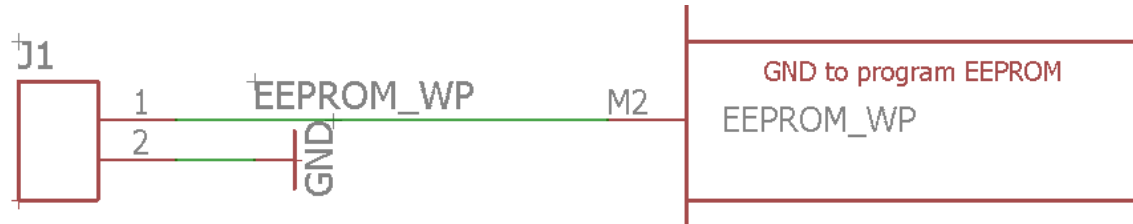


Figure 65: OSD3358-SM-RED EEPROM Write Protect Circuitry

The OSD335x-SM device has an EEPROM integrated inside the OSD335x-SM on the I2C0 line. EEPROM's write protect pin is brought out as a ball M2 (Shown in Figure 65) and can be used to write to the EEPROM. By default, the EEPROM holds some board identification and device configuration information. The board specific contents are given below:

Name	Description	Size (bytes)	Contents
Header	Header Value designates section is valid	4	0xAA, 0x55, 0x33, 0xEE
Board Name	Board name	8	Name in ASCII: A335BNLT
Octavo Systems HW Revision	HW Revision	4	Name in ASCII: Rev 3: OS00 Rev 4: OS01

The device specific information and its format can be found in the OSD3358-SM datasheet(<https://octavosystems.com/docs/osd335x-sm-datasheet/>)

4.22 Expansion Headers

The OSD3358-SM-RED board has 2 main expansion headers J5 and J6 that have the same pinout as BeagleBone® Black headers P8 and P9 if jumper J11 is unpopulated. The pinout is shown in the Figure 66. Detailed information regarding the pinout and multiplexing of each pin is described in section 'Connectors' of the BeagleBone® Black reference manual.

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021

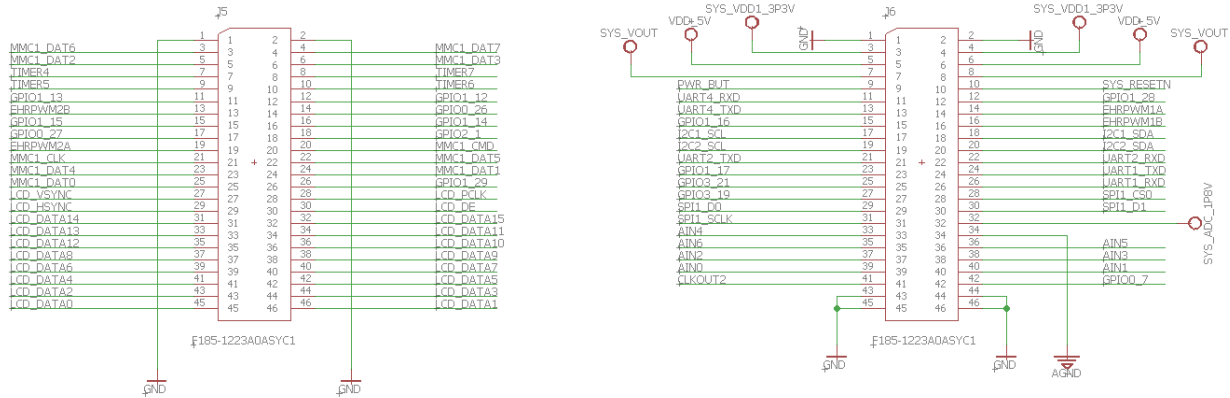


Figure 66: OSD3358-SM-RED Expansion Headers

4.23 On-board Connectors and Jumpers

The following table details the functions of all the headers and jumpers.

Table 6: Connector / Jumper Description

Designation	Function	
J1	Populating this jumper grounds the EEPROM write protect	
J2	Populating this jumper grounds the PMIC_NRESET pin of the OSD3358-SM	
J3	Function	UART0 Header
	Pin 1	GND
	Pin 2	NC
	Pin 3	NC
	Pin 4	UART0_RX
	Pin 5	UART0_TX
	Pin 6	NC
J4	Footprint for JTAG header is provided. Populate with the recommended part in section 3.7	
J5	Same pin map as header P8 of the Beaglebone Black	
J6	Same pin map as header P9 of the Beaglebone Black	
J7	Populating this jumper disables the 12MHz external HDMI clock from Y3	
J8	Function	I2C communication port for the USB port hub U8
	Pin 1	SDA pin of U8
	Pin 2	SCL pin of U8
J9	Function	UART communication port for the USB port hub U8
	Pin 1	UART_RX pin of U8
	Pin 2	UART_TX pin of U8
J10	Function	Temp sensor channel header
	Pin 1	GND
	Pin 2	SYS_VDD1_3P3V
	Pin 3	D- pin of U17
	Pin 4	D+ of channel 1
	Pin 5	D- pin of U17
	Pin 6	D+ of channel 2

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



	Pin 7	D- pin of U17
	Pin 8	D+ of channel 3
	Pin 9	D- pin of U17
	Pin 10	D+ of channel 4
J11	Populating this jumper enables the NOR flash SPI communication with the OSD3358-SM *NOTE: populating this header breaks BeagleBoard.org® cape compatibility of the OSD3358-SM-RED board	

4.24 Test Points

There are several test points provided on the OSD3358-SM board in order to facilitate debug. The following table provides a description of the available test points and their signal names.

Table 7: OSD3358-SM-RED Test Point Description

Designator/Label	Signal name
TP1	VDDS_DDR
TP2	VDD_CORE
TP3	VDD_MPU
TP4	VDDS_PLL
TP5 (VIN_BAT)	VIN_BAT (PMIC Battery voltage input)
TP6 (BAT_SENSE)	BAT_SENSE (PMIC Battery sense input)
TP7 (TEMP_SENSE)	TEMP_SENSE (PMIC Temperature sense input)
TP8	GND
TP9	TPM_GPIO1
TP10	TPM_GPIO2
TP11	SYS_VOUT
TP12	SYS_VDD1_3P3V
TP13	SYS_VDD2_3P3V
TP14	SYS_RTC_1P8V
TP15	SYS_VDD_1P8V
TP16	SYS_VDD3_3P3V
TP17	CLK_25M (Ethernet PHY)
TP18	USB_RESET (USB port hub)
TP19	NON_REM0 (USB port hub)
TP20	CFG_SEL1 (USB port hub)
TP21	PWR_EN
TP22	LDO_PGOOD
TP23	PMIC_NWAKEUP
TP24	PMIC_INT
TP25	RSTN (Ethernet PHY)
TP26	SYS_RESETN
TP27	SPI0_SCLK
TP28	SPI0_D1(MOSI)
TP29	SPI0_D0(MISO)
TP30	SPI0_CS0



5 Board Images and Software Features

The OSD3358-SM-RED board is loaded with a Beagleboard.org® compatible Linux image out of the box. This pre-installed Linux image on the board is a modified version of Debian LXQT image available on Beagleboard.org® website (<https://beagleboard.org/latest-images>). Modifications to the default image were made to customize the image to work with the hardware on the board. The following changes were made to the default Linux image:

1. A new device tree specific to the OSD335x-SM-RED platform is used to make Linux aware of the board specific hardware
2. The Robotiscape library (<http://www.strawsondesign.com/#!/manual-install>) that comes installed by default is modified to work with the sensors on board (hardware revision V3 only). Note that the present version of the Robotiscape library is not compatible with the modified library that comes installed out of the box on the RED board.

5.1 Boot Sequence

The board goes through multiple stages during boot to a Linux environment. These stages are described below:

1. TI Boot ROM: The processor contains a boot ROM that performs some peripheral configuration required for boot up and passes the control to the next stage bootloader
2. MLO: This stage sets up pin-muxing, initializes the clocks and memory, and loads the next stage bootloader
3. U-boot: This is the primary Linux bootloader. It looks for a boot source and reads the boot configuration text file (uEnv.txt) for boot arguments from the boot source. It specifies the root file system, performs additional initialization, loads and passes control to the Linux kernel. A device tree binary file (hardware specific board configuration file) specified in the configuration text file, is also passed to the kernel for additional setup.
4. Linux kernel: Decompresses and loads the kernel into memory, performs peripheral initializations and mounts the file system that contains all the Linux applications.

Useful resources:

1. Linux core U-Boot User's Guide: http://processors.wiki.ti.com/index.php/Linux_Core_U-Boot_User%27s_Guide
2. Das U-Boot home page: <http://www.denx.de/wiki/U-Boot/WebHome>
3. Building U-Boot, Root Filesystem and Linux Kernel: <https://eewiki.net/display/linuxonarm/BeagleBone+Black>

5.2 Device Tree

The OSD3358-SM-RED board has a device tree file that tells the Linux kernel about the hardware on board. Derivative designs from the OSD3358-SM-RED will need to modify the device tree appropriately. A Flattened Device Tree (FTD) is a data structure that describes the hardware on the board to the Linux kernel. A compiled version, .dtb (Device Tree Binary) of this file specified in the boot configuration file uEnv.txt located in /boot/ folder communicates which peripherals are available and enabled, device drivers that need to be loaded, default pin mux settings after boot, etc. The device Tree Binary file is a compiled version of a .dts(Device Tree Source) file. A Device Tree Compiler (dtc) is necessary to convert the source file into the binary file. The device tree source files for the OSD335x devices and the OSD3358-SM-RED board are linked below. The best way to write custom device trees is by using the dtb-rebuilder and is also linked below.

Useful resources:

1. OSD3358-SM-RED device tree files: <https://github.com/octavosystems/OSD335x-Device-Tree>
2. Device tree usage: http://elinux.org/Device_Tree_Usage
3. Device tree reference: http://elinux.org/Device_Tree_Reference
4. Device tree rebuilder: <https://github.com/RobertCNelson/dtb-rebuilder>

5.2.1 Device Tree Overlays

Similar to the capes of Beaglebone Black, capes can be added to the OSD3358-SM-RED platform for enhanced functionality. The board is cape compatible with existing Beaglebone Black capes. So, capes for the Beaglebone Black can be used with the OSD3358-SM-RED board. The additional cape specific setup for Linux is done using Device Tree Overlays. BeagleBone Black compatible expansion capes also normally include an I2C EEPROM (must be on I2C2 in the address range of 0x54 to 0x57), that provides information such as the name of the cape, cape version, part number, and other relevant run-time information. The information required to load the correct Device Tree Overlay for a specific cape is read from the I2C EEPROM by the Cape Manager (<http://elinux.org/Capemgr>). See the BeagleBone Black SRM (<https://github.com/beagleboard/beaglebone-black/wiki/System-Reference-Manual>) “Cape Board Support” section for details. Device Tree Overlays can also be loaded (cape is enabled) or unloaded (cape is disabled) at run-time through the cape manager’s /sys interface.

Useful resources:

1. Cape Manager: <http://elinux.org/Capemgr>
2. Using cape manager and device tree overlays: <https://github.com/jadonk/validation-scripts/tree/master/test-capemgr>
3. Beaglebone Black capes: http://elinux.org/Beagleboard:BeagleBone_Capes
4. Device tree overlays tutorial: <https://learn.adafruit.com/introduction-to-the-beaglebone-black-device-tree/exporting-and-unexporting-an-overlay>

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



5.3 Using On-Board Sensors - V3

As described in section 3 & 4, there are several sensors on OSD3358-SM-RED board. The board is equipped with a software library and examples to interface with the sensors. For board revision V3, the library package 'Redperipheralib' is pre-installed on Linux and allows commands line access to these components on the board. The Robotcsape library (<http://strawsonedesign.com/#!/manual-install>) that comes pre-installed on the Beagleboard.org® images was used as a base to build the Redperipheralib library. The library is located in **/home/debian/** folder of the Linux environment. The structure of the library is given below:

```

└─ Redperipheralib
   └─ debian
   └─ examples
      ├── rc_benchmark_algebra
      ├── rc_calibrate_gyro
      ├── rc_calibrate_mag
      ├── rc_check_model
      ├── rc_cpu_freq
      ├── rc_kill
      ├── rc_test_adc
      ├── rc_test_algebra
      ├── rc_test_barometer
      ├── rc_test_drivers
      ├── rc_test_filters
      ├── rc_test_imu
      ├── rc_test_polynomial
      ├── rc_test_time
      ├── rc_test_tmp
      ├── rc_test_vector
      ├── rc_uart_loopback
      └── rc_version
   └─ libraries
      ├── bmp280
      ├── gpio
      ├── math
      ├── mmap
      ├── mpu9250
      ├── other
      ├── pwm
      ├── serial_ports
      └── tmp468

```

Figure 63: Structure of Redperipheralib

- **debian:** Contains Debian specific configuration parameter files
- **examples:** Contains example programs that show how to use the peripherals onboard
- **libraries:** Contains libraries necessary to exercise the peripherals onboard
 - **bmp280:** Contains functions and definitions to interface with the pressure sensor
 - **gpio:** Contains functions and definitions to control the GPIOs of the board
 - **math:** Contains functions and definitions to perform mathematical operations required
 - **mmap:** Contains functions and definitions to perform memory mapping operations
 - **mpu9250:** Contains functions and definitions to interface with the IMU

- **other**: Contains functions and definitions for additional features of the library
- **pwm**: Contains functions and definitions to generate a PWM
- **serial_ports**: Contains functions and definitions to perform I2C and SPI protocol communications
- **tmp468**: Contains functions and definitions to interface with the temperature sensor

5.3.1 Accessing Peripherals using the Terminal

The RED peripheral library facilitates command line access to board features. The commands, options and output formats are given below.

1. *rc_benchmark_algebra*

Function: This command performs several matrix and vector related benchmark tests and returns the time taken to perform each test.

Options:

- d: use default matrix size
- s {size}: specify matrix size
- h: Display help message

Result:

```
root@beaglebone:~$ rc_benchmark_algebra -d
Starting
  3888us Time to make random matrix & vector
   474us Time to duplicate matrix
 10849us Time to multiply matrices
   505 MFLOPS multiplying matrices
 20217us Time to find matrix determinant
228666us Time to invert matrix
 97589us Time to do LUP decomposition
950841us Time to do QR decomposition
 11431us Time to solve linear system
DONE
```

2. *rc_calibrate_gyro*

Function: This command performs calibration of the gyro sensor in the IMU

Options: None

Result:

```
root@beaglebone:~$ rc_calibrate_gyro
Starting calibration routine

gyro calibration file written
run rc_test_imu to check performance
```

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021

3. *rc_calibrate_mag***Function:** This command calibrates the magnetometer in the IMU**Options:** None**Result:**

```
root@beaglebone:~$ rc_calibrate_mag
```

```
This will sample the magnetometer for the next 15 seconds
Rotate the cape around in the air through as many orientations
as possible to collect sufficient data for calibration
Press ENTER to continue or anything else to quit
spin spin spin!!!
```

```
keep spinning
you're doing great
keep spinning
you're doing great
keep spinning
you're doing great
```

```
Okay Stop!
Calculating calibration constants.....
```

```
Offsets X:  71.924 Y:  -1.718 Z: -24.156
Scales  X:   1.956 Y:   0.929 Z:   2.485
```

```
magnetometer calibration file written
run rc_test_imu to check performance
```

4. *rc_check_model***Function:** This command returns the board name that it is currently running on**Options:** None**Results:**

```
root@beaglebone:~$ rc_check_model
```

```
Currently running on a:
OCT_OSD3358-SM-RED
```

5. *rc_kill*

Function: This command kills any existing peripheral library programs that is running onboard

Options: None

Results:

```
root@beaglebone:~$ rc_kill
No existing roboticscape program is running.
```

6. *rc_test_adc*

Function: This command returns the ADC results of 7 channels of the ADC module

Options: None

Results:

```
root@beaglebone:~$ rc_test_adc
adc_0: 3973
adc_1: 3693
adc_2: 3930
adc_3: 2242
adc_4: 2217
adc_5: 3304
adc_6: 3328
```

7. *rc_test_barometer*

Function: This command runs a test on the barometer and returns the temperature, pressure, altitude and filtered altitude sensor values

Options: None

Results:

```
root@beaglebone:~$ rc_test_barometer
temp   | pressure | altitude | filtered |
35.08C | 101.57kpa | -20.04m  | -19.88m  |
```

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



8. *rc_test_imu*

Function: This command tests and returns the IMU measurements

Options:

- None : print sensor values with default settings
- r: print raw values instead of radians
- d: print gyro in deg/s instead of radians
- h: print this help message

Results:

```
root@beaglebone:~$ rc_test_imu
try 'test_imu -h' to see other options
  Accel XYZ(m/s^2) |   Gyro XYZ (rad/s)   |  Mag Field XYZ(uT)  |
Temp (C)
  0.43   0.18   9.74 |   0.0   -0.0   0.0 |  -2.2  -78.4  -33.8 |
37.9
```

9. *rc_test_time*

Function: This command tests the rc time functions of the peripheral library

Options: None

Results:

```
root@beaglebone:~$ rc_test_time

testing rc time functions
time to call rc_nanos_since_epoch: 1081ns
time to call rc_nanos_since_boot: 1102ns
time to call rc_nanos_thread_time: 1901ns
```

10. *rc_test_tmp*

Function: This command tests the temperature sensor and returns the temperature value on channel 8

Options: None

Results:

```
root@beaglebone:~$ rc_test_tmp
temperature sensor read value in celsius: 28.750000
```

11. *rc_uart_loopback*

Function: This command tests the UART protocol capabilities of the library by running a loopback test on UART0 port. UART0_TX and UART0_RX need to be connected for this test to run.

Options: None

Results:

```
root@beaglebone:~$ rc_uart_loopback 0
testing UART bus 0
Sending 11 bytes: Hello World
Received 11 bytes: Hello World
```

12. *rc_version*

Function: This command returns the version of the peripheral library being run on the board

Options: None

Results:

```
debian@beaglebone:~$ rc_version
REDPeripherallib 0.3.4 built on Roboticscape 0.3.4
```

5.3.2 Using Peripheral Library to Write Applications

All the peripheral access supporting libraries are present in the libraries folder. The Makefile will compile a single shared object `robotics_cape.so` and install it to `/usr/lib`. It will also move all the header files to `/usr/include/`. To use the `.so`, it is enough to add `#include <robotics_cape.h>`.

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



5.4 Using On-Board Sensors - V4

This section described the process of exercising the on-board sensors of OSD3358-SM-RED V4. The sensors on OSD3358-SM-RED are instantiated in the device tree and are exposed to the user via SYSFS interface. The following commands can be used to exercise each of these sensors.

5.4.1 Temperature Sensor - TMP468 (U17)

The temperature sensor is present on address 0x48 on bus I2C0. The SYSFS path to access the functions exposed by the driver for TMP468 is shown below:

```
debian@osd3358-sm-red:~$cd /sys/bus/i2c/devices/0-0048/hwmon/hwmon0
```

The following files are available in this folder to interact with the temperature sensor:

```
debian@osd3358-sm-red:/sys/bus/i2c/devices/0-0048/hwmon/hwmon0$ ls
device          temp1_max      temp2_max_alarm temp3_max_alarm
temp4_max_alarm temp5_max_alarm temp6_max_alarm  temp7_max_alarm
temp8_max_alarm temp9_max_alarm
name            temp1_max_alarm temp2_max_hyst   temp3_max_hyst
temp4_max_hyst  temp5_max_hyst  temp6_max_hyst   temp7_max_hyst
temp8_max_hyst  temp9_max_hyst
of_node         temp1_max_hyst  temp2_offset     temp3_offset
temp4_offset    temp5_offset    temp6_offset     temp7_offset
temp8_offset    temp9_offset
power           temp2_crit     temp3_crit       temp4_crit
temp5_crit      temp6_crit     temp7_crit       temp8_crit
temp9_crit      uevent
subsystem       temp2_crit_alarm temp3_crit_alarm temp4_crit_alarm
temp5_crit_alarm temp6_crit_alarm temp7_crit_alarm temp8_crit_alarm
temp9_crit_alarm
temp1_crit      temp2_crit_hyst temp3_crit_hyst  temp4_crit_hyst
temp5_crit_hyst temp6_crit_hyst  temp7_crit_hyst  temp8_crit_hyst
temp9_crit_hyst
temp1_crit_alarm temp2_fault     temp3_fault      temp4_fault
temp5_fault     temp6_fault     temp7_fault      temp8_fault
temp9_fault
temp1_crit_hyst temp2_input     temp3_input      temp4_input
temp5_input     temp6_input     temp7_input      temp8_input
temp9_input
temp1_input     temp2_max       temp3_max        temp4_max
temp5_max       temp6_max       temp7_max        temp8_max
temp9_max
```

A total of 9 configurable temperature measurement channels are available. Temp1 is internal to the temperature sensor. Temp2 – temp9 are for external channels. On OSD3358-SM-RED, channel 8(temp9) is used to measure board temperature using a transistor. The following commands show how to read the internal temperature sensor and temperature at the transistor Q2.

```
debian@osd3358-sm-red:/sys/bus/i2c/devices/0-0048/hwmon/hwmon0$ cat
temp1_input
32743
debian@osd3358-sm-red:/sys/bus/i2c/devices/0-0048/hwmon/hwmon0$ cat
temp9_input
36878
```

The sensor reports the reading in degrees Celsius and without a decimal point. In the above case, the temperature reading on channel temp1 should be interpreted as $32743/1000 = 32.743$ degree Celsius.

5.4.2 Barometer - LPS22HB (U22)

The Barometer is present on address 0x5c on bus I2C0. The SYSFS path to access the functions exposed by the driver for LPS22HB is shown below:

```
debian@osd3358-sm-red:~$ cd /sys/bus/i2c/devices/0-005c/iio:device1
```

The following files are available in this folder to interact with the pressure sensor:

```
debian@osd3358-sm-red:/sys/bus/i2c/devices/0-005c/iio:device1$ ls
buffer          dev              in_pressure_scale  in_temp_scale
of_node         sampling_frequency  scan_elements      trigger
current_timestamp_clock  in_pressure_raw    in_temp_raw        name
power           sampling_frequency_available  subsystem          uevent
```

The data from the sensor can be read using the following command:

```
debian@osd3358-sm-red:/sys/bus/i2c/devices/0-005c/iio:device1$ cat
in_pressure_raw
4179345
```

The reading needs to be divided by the scaling factor (4096) to get the pressure data in hPa. Here in the above case, the reading is $4179345/4096 = 1020.34$ hPa.

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



5.4.3 Accelerometer - ICM-20948 (U23)

The accelerometer is present on address 0x68 on bus I2C0. The SYSFS path to access the functions exposed by the driver for ICM-20948 is shown below:

```
debian@osd3358-sm-red:~$cd /sys/bus/i2c/devices/0-0068/iio:device0
```

The following files are available in this folder to interact with the accelerometer:

```
debian@osd3358-sm-red:/sys/bus/i2c/devices/0-0068/iio:device0$ ls
buffer                      in_accel_scale              in_accel_y_raw
in_anglvel_scale_available  in_anglvel_z_calibbias     in_temp_scale
sampling_frequency_available
current_timestamp_clock    in_accel_scale_available   in_accel_z_calibbias
in_anglvel_x_calibbias     in_anglvel_z_raw          name
scan_elements
dev                        in_accel_x_calibbias       in_accel_z_raw
in_anglvel_x_raw           in_gyro_matrix             of_node
subsystem
in_accel_matrix            in_accel_x_raw             in_anglvel_mount_matrix
in_anglvel_y_calibbias     in_temp_offset            power
trigger
in_accel_mount_matrix      in_accel_y_calibbias       in_anglvel_scale
in_anglvel_y_raw           in_temp_raw                sampling_frequency
uevent
```

The following data can be read from the sensor:

1. Accelerometer data (x, y, z directions)
2. Gyro data (x, y, z directions)
3. Temperature data

For example, the acceleration data in z direction can be determined by using the following command:

```
debian@osd3358-sm-red:/sys/bus/i2c/devices/0-0068/iio:device0$ cat
in_accel_z_raw
16217
```

The scaling factor for accelerometer is 16,384 (LSB/g). So, the acceleration of the board in z direction can be determined by dividing the sensor reading by the scaling factor: 16217/16384: 0.98 g

6 Cape Support

The OSD3358-SM-RED board is capable of supporting all the capes of the BeagleBone® Black. See the BeagleBone® Black SRM (<https://github.com/beagleboard/beaglebone-black/wiki/System-Reference-Manual>) “Cape Board Support” section for details on capes for BeagleBone® Black. The capes can be dynamically loaded/unloaded using device tree overlays.

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



7 Mechanicals

Figure 64 show the dimensions of the board.

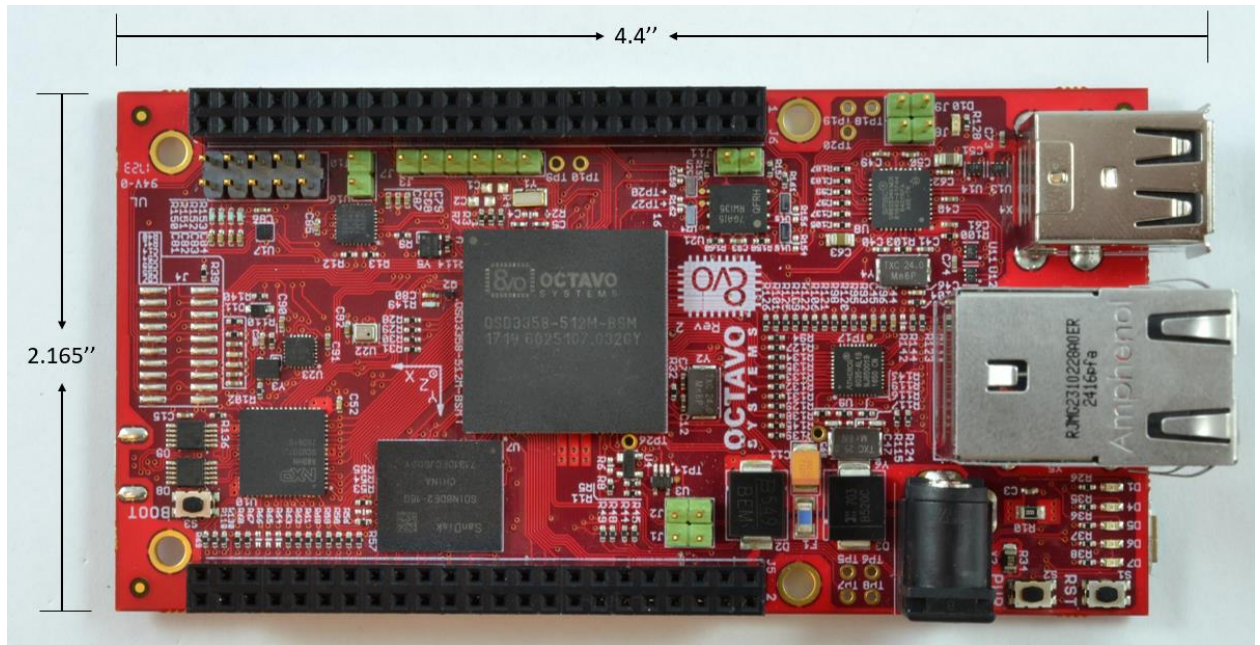


Figure 64: Dimensions of the OSD3358-SM-RED

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021

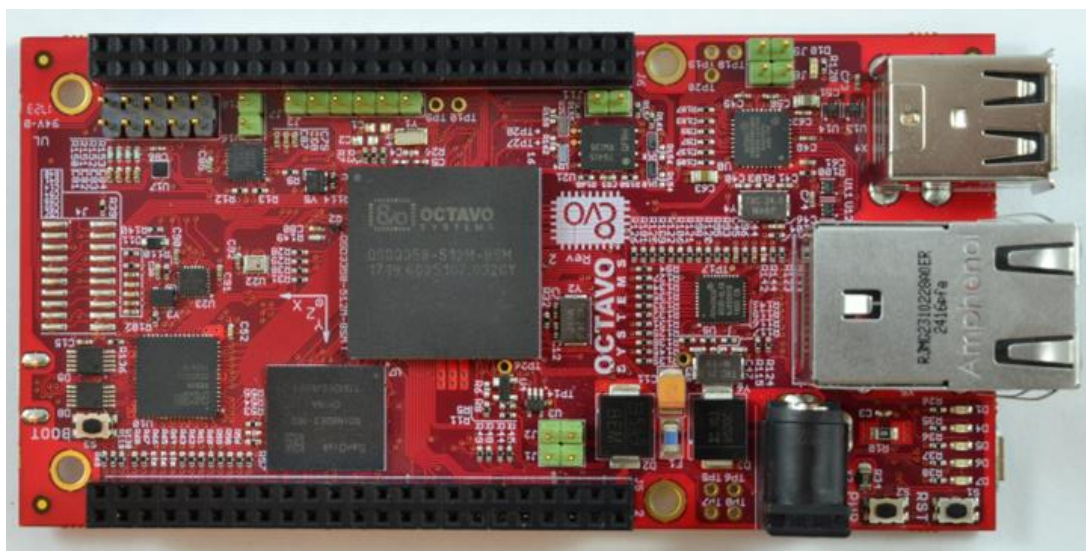


Figure 65: OSD3358-SM-RED - Top View

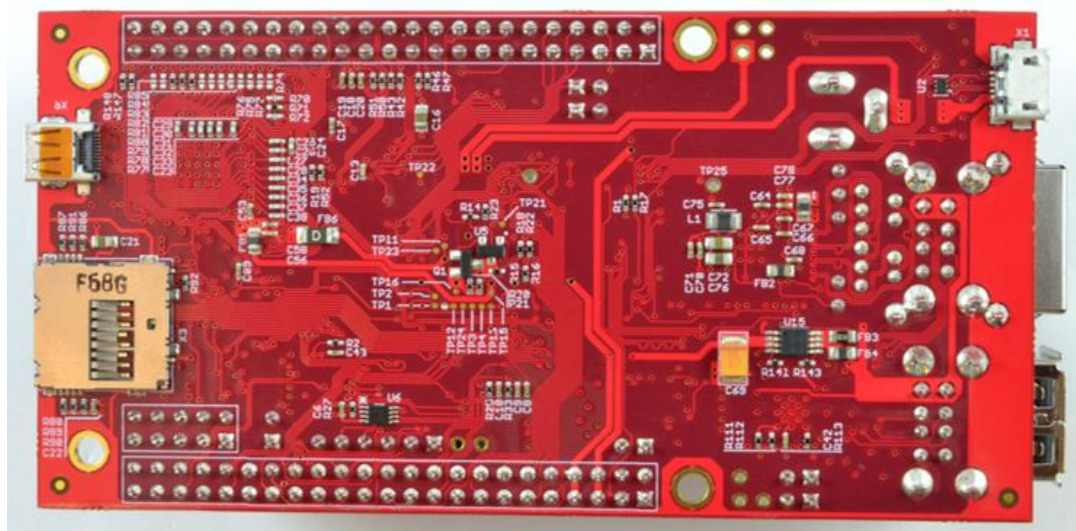


Figure 66: OSD3358-SM-RED - Bottom View

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



8 Support and Troubleshooting

A list of useful support resources is given below:

- OSD3358-SM-RED board design files:
<https://octavosystems.com/files/osd3358-sm-red-eagle-files/>
- Sign up for OSD3358-SM-RED latest software updates:
<https://octavosystems.com/doc-change-signup/>
- OSD3358-SM datasheet:
<https://octavosystems.com/docs/osd335x-sm-datasheet/>
- OSD3358-SM RED platform getting started guide:
<https://octavosystems.com/docs/osd3358-sm-red-quick-start-guide/>
- OSD3358-SM Application notes:
 - a. OSD335x-SM layout guide:
https://octavosystems.com/app_notes/osd335x-sm-layout-guide/
 - b. OSD335x Family Pin Assignments Compared to AM335x:
https://octavosystems.com/app_notes/osd335x-family-pin-assignments/
- OSD3358-SM-RED RMA request page: <https://octavosystems.com/contact/>

9 Reference Documents

1. OSD335x datasheet: <https://octavosystems.com/docs/osd335x-sm-datasheet/>
2. AM335x datasheet: <http://www.ti.com/lit/ds/sprs717j/sprs717j.pdf>
3. AM335x Technical Reference Manual: <http://www.ti.com/lit/ds/sprs717j/sprs717j.pdf>
4. TPS65217C Datasheet: <http://www.ti.com/lit/ds/symlink/tps65217.pdf>
5. Powering AM335x with TPS65217x: <http://www.ti.com/lit/ug/slvu551i/slvu551i.pdf>
6. TL5209 LDO Datasheet: <http://www.ti.com/lit/ds/symlink/tl5209.pdf>
7. 4KB EEPROM Datasheet: <http://www.microchip.com/wwwproducts/en/24LC32A>
8. Compatible WiFi adapters:
http://www.elinux.org/Beagleboard:BeagleBoneBlack#WIFI_Adapters
9. PUTTY tool: <http://www.putty.org/>
10. Beaglebone Black expansion headers:
http://elinux.org/Beagleboard:Cape_Expansion_Headers
11. HDMI Framer TDA199988 Datasheet:
https://media.digikey.com/pdf/Data%20Sheets/NXP%20PDFs/TDA19988_DS_21_july_2011.pdf
12. Ethernet PHY AR8035 Datasheet:
[https://media.digikey.com/pdf/Data%20Sheets/CSR%20PDFs/AR8035_DS_\(Atheros\)_Mar2011.pdf](https://media.digikey.com/pdf/Data%20Sheets/CSR%20PDFs/AR8035_DS_(Atheros)_Mar2011.pdf)
13. USB port hub USB2534-1080AEN datasheet:
<http://www1.microchip.com/downloads/en/DeviceDoc/00001713A.pdf>
14. IMU MPU9250 datasheet: <https://www.invensense.com/wp-content/uploads/2015/02/PS-MPU-9250A-01-v1.1.pdf>
15. Temperature sensor TMP468 datasheet: <http://www.ti.com/lit/ds/symlink/tmp468.pdf>
16. Pressure sensor BMP280 datasheet: https://ae-bst.resource.bosch.com/media/tech/media/datasheets/BST-BME280_DS001-11.pdf
17. TPM AT97SC3205T datasheet: <http://www.atmel.com/Images/Atmel-8883S-TPM-AT97SC3205T-Datasheet-Summary.pdf>
18. NOR flash S25FL127S datasheet: <http://www.cypress.com/file/177961/download>
19. OSD335x Power application note: https://octavosystems.com/app_notes/osd335x-power-application-note/
20. I2C pullup calculation: <http://www.ti.com/lit/an/slva689/slva689.pdf>
21. OSD335x Design Tutorial series - OSD335x Reset Circuitry article
https://octavosystems.com/app_notes/osd335x-design-tutorial/bare-minimum-boot/reset-circuitry/
22. OSD335x Reference Design Lesson 1 Power Circuitry Part 4: Clamping Circuit:
https://octavosystems.com/app_notes/osd335x-reference-design-lesson-1-power-circuitry-part-4-clamping-circuit/
23. Reference Design Tutorial series - Clock circuitry article:
https://octavosystems.com/app_notes/osd335x-design-tutorial/bare-minimum-boot/clock-circuitry/

OSD3358-SM-RED – User Guide

Rev. 2.0 11/26/2021



24. RJMG2310228A0ER Datasheet
(https://media.digikey.com/pdf/Data%20Sheets/Amphenol%20PDFs/RJMG2310228A0ER_Dwg.pdf)
25. Beagleboard.org® latest images: <https://beagleboard.org/latest-images>
26. Roboticscape library: <http://www.strawsondesign.com/#!/manual-install>
27. Linux core U-Boot User's Guide: http://processors.wiki.ti.com/index.php/Linux_Core_U-Boot_User%27s_Guide
28. Das U-Boot home page: <http://www.denx.de/wiki/U-Boot/WebHome>
29. Building U-Boot, Root Filesystem and Linux Kernel:
<https://eewiki.net/display/linuxonarm/BeagleBone+Black>
30. OSD3358-SM-RED device tree files: <https://github.com/octavosystems/OSD335x-Device-Tree>
31. Device tree usage: http://elinux.org/Device_Tree_Usage
32. Device tree reference: http://elinux.org/Device_Tree_Reference
33. Device tree rebuilder: <https://github.com/RobertCNelson/dtb-rebuilder>
34. BeagleBone® Black SRM: <https://github.com/beagleboard/beaglebone-black/wiki/System-Reference-Manual>