



Multiscale Observation Networks for Optical  
monitoring of Coastal waters, Lakes and Estuaries

*SOS-box: legacy sensor interface*

*Operating guide*

### **Project Description**

**Funded by EU H2020 MONOCLE creates sustainable *in situ* observation solutions for Earth Observation (EO) of optical water quality in inland and transitional waters.** MONOCLE develops essential research and technology to lower the cost of acquisition, maintenance, and regular deployment of *in situ* sensors related to optical water quality. The MONOCLE sensor system includes handheld devices, smartphone applications, and piloted and autonomous drones, as well as automated observation systems for e.g. buoys and shipborne operation. The sensors are networked to establish interactive links between operational Earth Observation (EO) and essential environmental monitoring in inland and transitional water bodies, which are particularly vulnerable to environmental change.



Authors	Name	Organisation	Role / Title
Deliverable Lead	Stefan Simis	PML	Scientific coordinator
Contributing Author(s)	Jaime Kershaw Brown	PML	Lead developer / author
	Darren Snee	PML	WP5 Lead developer

**Document History:**

Release	Date	Reason for Change	Status	Distribution
1.0		First release	Final	Public

**To cite this document:**

Simis, SGH, Kershaw Brown, J, Snee, D. (2019). Operators handbook for the MONOCLE legacy sensor interface (SOS-box). Plymouth Marine Laboratory, UK. DOI: DOI: 10.5281/zenodo.3558533

**© The authors**



## TABLE OF CONTENTS

1. Hardware requirements.....	4
2. Hardware setup.....	4
3. Software setup .....	5



# 1. Hardware requirements

The SOS-box is designed to work on the low-cost, low-power Raspberry Pi zero platform. To get started we recommend you acquire the following hardware components:

- Raspberry Pi Zero wireless
- Power supply for the Raspberry Pi Zero
- Micro SD card (16GB recommended)
- A converter cable to connect the micro HDMI output of the Pi to a computer monitor
- A three-port USB hub with Ethernet and a micro USB connector

Additionally you will need a computer with an SD card reader, or a portable SD card reader to initially set up the pi zero, but these are no longer needed after the first software setup.

You may additionally require a USB drive to transfer setup files onto the Pi zero, if you do not connect the pi to a network.

# 2. Hardware setup

Follow the steps below to set up your Raspberry Pi Zero.

Install the Raspbian operating system software onto the micro SD card. To do this, place micro SD card into a card reader and connect to a PC. Download raspbian from the following address:

<https://www.raspberrypi.org/downloads/raspbian/>

We recommend you install the “Raspbian Buster with desktop and recommended software” option onto the Pi, provided you have at least an 8GB SD card. Follow this link for a full guide on how to install Raspbian:

<https://howchoo.com/g/ywmxmza2ndf/raspbian-buster-install-or-upgrade>

A simpler option may be to download NOOBS instead of Raspbian but we have not teste this approach. A guide on how to install NOOBS can be found here:

<https://www.raspberrypi.org/documentation/installation/noobs.md>

Now that your Pi is set up with an operating system we can add the SOS-box software. Use a micro-HDMI adaptor cable to connect the Pi to a monitor. Then attach the USB hub with Ethernet to the pi, and plug in a mouse and keyboard. Connect the Pi to a power supply and wait for the Pi to boot up, which may take a couple of minutes. Apply your preferred operating settings (language etc), which will be followed by several software updates, which may take a while.

## 3. SOS-box software setup

### Software dependencies

Software dependencies should be installed prior to adding the SOS-box software. At this time, only installation of `pyudev` is needed. Open a terminal, and type

```
pip3 install pyudev
```

If for any reason this does not work, try instead

```
sudo install pyudev
```

### Network setup

You will need to set up network services to run permanently. Do this by following the steps below:

First create the following file (e.g. using `vi`, `vim`, `nano` or another editor)

```
/etc/network/interfaces.d/wlan0
```

Enter the following lines into this file

```
auto wlan0

iface wlan0 inet static
address 192.168.0.1
netmask 255.255.255.0
wireless-channel 11
wireless-essid sosbox
wireless-mode ad-hoc
```

In the above configuration, if you plan to have multiple sensor interfaces on the same platform, you may modify the `wireless-essid` value to make it individually recognisable. For example, number your devices (`sosbox01-10`) or name them according to the sensor-serial number you will connect to each interface.

Now run the following commands:

```
apt-get install udhcpd
systemctl enable udhcpd
systemctl disable wpa_supplicant
mv /etc/rc3.d/S01udhcpd /etc/rc3.d/S50udhcpd
```

Next, modify `/etc/init.d/udhcpd` so that the section marked `Required-Start` has `$network` as its final element:

```
# Required-Start:
```

```
$remote_fs $syslog $network
```

## Download and configure SOS-box software

Next, we need to download the SOS-box software itself, which is found at this link:

<https://github.com/monocle-h2020/sosbox>

Add the following line to cron for user pi. To do this open up a terminal, enter `crontab -e`, and then enter the text below on one line.

```
@reboot /bin/bash home/pi/Documents/sensor_recorder/always-run.sh >>
/home/pi/Documents/sensor_recorder/sensor_recorder.log
```

**Your SOS box is now set up to accept sensor input. You may disconnect mouse, keyboard and monitor and mount the interface in a suitable (e.g. weather-proof) enclosure.**

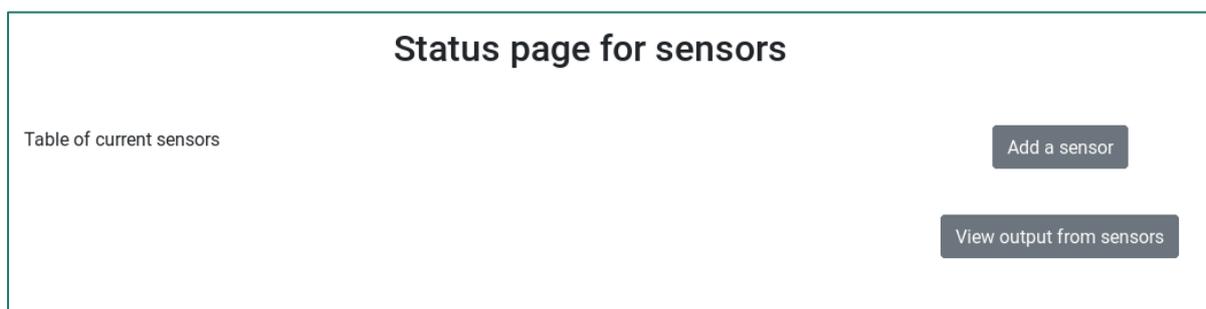
## 4. Connecting a sensor

If the sensor has a USB connector it may be plugged into the USB-hub connected to the SOS-box. Please verify that your sensor is compatible with the Linux operating system.

Most of the legacy sensors for which the SOS box is developed will have a non-USB serial output. You will need to connect the appropriate USB-to-serial interface for your sensor. For example, a sensor using the RS232 protocol requires a different USB adaptor cable than a sensor providing TTL (low voltage) level serial connection. Please refer to the manual of the sensor manufacturer to choose the right interface cable.

With the sensor connected to USB, it should be recognized as an external device by the Linux operating system. The SOS-box interface allows us to inspect any connected device through its configuration pages, which are available on its WiFi interface. Using a laptop, tablet or phone, connect to the WiFi network that your interface is broadcasting (default: sosbox). Then, point an internet browser to address `192.168.0.1`.

You will be presented with a home page which also serves as the status page for any active devices that you have set up and connected.



To add a sensor, click on the "Add a sensor" button in the top right of the page. This will re-direct you to a new page where you can see all attached devices, and allows you to select one, and then

select the sensor type that it is from the list below. Please see image below for reference where there are no sensor devices attached.

### Add a new sensor

100 bytes of data from the sensor

[Return to status page](#)

Sensor identification information

Select the sensor you have attached from the list below.  
Enter your unique sensor name for identification

[BB3 sensor](#) ▾

[Add sensor](#)

When a sensor device has been attached, a table listing the devices will appear on the right hand side.

[Return to status page](#)

Sensor identification information

Device is 154601a7, the port used is: /dev/ttyACM0,

Select the sensor you have attached from the list below.  
Enter your unique sensor name for identification

[BB3 sensor](#) ▾

[Add sensor](#)

To start reading data from the device follow these steps:

1. Select the device using the button on the left of the device description.
2. Select the sensor type from the drop down list.
3. Click the button "Add sensor".

[Return to status page](#)

**Sensor identification information**

Device is 154601a7, the port used is: /dev/ttyACM0,

Select the sensor you have attached from the list below.  
Enter your unique sensor name for identification

GPS ▼

[Add sensor](#)

Once you have completed the above steps, click on the “Return to status page” button, and you will see the newly added device.

**Status page for sensors**

<p>Table of current sensors</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">GPS_UBLOX7</td> </tr> </table>	GPS_UBLOX7	<p><a href="#">Add a sensor</a></p> <p><a href="#">View output from sensors</a></p>
GPS_UBLOX7		

To view the output of the device, click on the “View output from sensors” button in the top right corner. This will redirect you to the page seen in the image below.

**View the latest output from the sensors**

<p>Latest 10 outputs from the sensor</p>	<p><a href="#">Return to status page</a></p>	
<p>Table of current sensors</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;"><input type="radio"/> GPS_UBLOX7,</td> </tr> </table>		<input type="radio"/> GPS_UBLOX7,
<input type="radio"/> GPS_UBLOX7,		

You will see a list of sensors that you are reading from in a table on the right. To view the output of an individual sensor, click on the button next the to device name.

A table will appear with the latest 10 outputs from that device. To refresh the outputs, you must click on the button next to the device again. See figure below for an example.

## View the latest output from the sensors

Latest 10 outputs from the sensor

3440,,,,,0,\$GPRMC,,V,,,,,,,,,N*53,\$GPVTG,,,,,,N*30,\$GPGGA,,,,,0,00,99.99,,,,,*48,\$GPGSA,A,1,,,,,,,,,99.99,99.99,99.99*30,\$GPGSV,1,1,01,18,,,22*71,\$GPGLL,,,,,V,N*64,
3439,,,,,0,\$GPRMC,,V,,,,,,,,,N*53,\$GPVTG,,,,,,N*30,\$GPGGA,,,,,0,00,99.99,,,,,*48,\$GPGSA,A,1,,,,,,,,,99.99,99.99,99.99*30,\$GPGLL,,,,,V,N*64
3438,,,,,0,\$GPRMC,,V,,,,,,,,,N*53,\$GPVTG,,,,,,N*30,\$GPGGA,,,,,0,00,99.99,,,,,*48,\$GPGSA,A,1,,,,,,,,,99.99,99.99,99.99*30,\$GPGLL,,,,,V,N*64
3437,,,,,0,\$GPRMC,,V,,,,,,,,,N*53,\$GPVTG,,,,,,N*30,\$GPGGA,,,,,0,00,99.99,,,,,*48,\$GPGSA,A,1,,,,,,,,,99.99,99.99,99.99*30,\$GPGLL,,,,,V,N*64
3436,,,,,0,\$GPRMC,,V,,,,,,,,,N*53,\$GPVTG,,,,,,N*30,\$GPGGA,,,,,0,00,99.99,,,,,*48,\$GPGSA,A,1,,,,,,,,,99.99,99.99,99.99*30,\$GPGLL,,,,,V,N*64
3435,,,,,0,\$GPRMC,,V,,,,,,,,,N*53,\$GPVTG,,,,,,N*30,\$GPGGA,,,,,0,00,99.99,,,,,*48,\$GPGSA,A,1,,,,,,,,,99.99,99.99,99.99*30,\$GPGSV,1,1,01,16,,,20*7D,\$GPGLL,,,,,V,N*64,
3434,,,,,0,\$GPRMC,,V,,,,,,,,,N*53,\$GPVTG,,,,,,N*30,\$GPGGA,,,,,0,00,99.99,,,,,*48,\$GPGSA,A,1,,,,,,,,,99.99,99.99,99.99*30,\$GPGLL,,,,,V,N*64
3433,,,,,0,\$GPRMC,,V,,,,,,,,,N*53,\$GPVTG,,,,,,N*30,\$GPGGA,,,,,0,00,99.99,,,,,*48,\$GPGSA,A,1,,,,,,,,,99.99,99.99,99.99*30,\$GPGSV,1,1,03,01,,,10,09,,,21,25,,,10*76,\$GPGLL,,,,,V,N*64,
3432,,,,,0,\$GPRMC,,V,,,,,,,,,N*53,\$GPVTG,,,,,,N*30,\$GPGGA,,,,,0,00,99.99,,,,,*48,\$GPGSA,A,1,,,,,,,,,99.99,99.99,99.99*30,\$GPGSV,1,1,04,01,,,11,08,,,28,19,,,21,25,,,09*7B,\$GPGLL,,,,,V,N*64,
3431,,,,,0,\$GPRMC,,V,,,,,,,,,N*53,\$GPVTG,,,,,,N*30,\$GPGGA,,,,,0,00,99.99,,,,,*48,\$GPGSA,A,1,,,,,,,,,99.99,99.99,99.99*30,\$GPGSV,1,1,03,01,,,17,19,,,21,25,,,11*71,\$GPGLL,,,,,V,N*64,

[Return to status page](#)

Table of current sensors

● [GPS\\_UBLOX7](#)

To return to the status page, click the button in the top right corner.