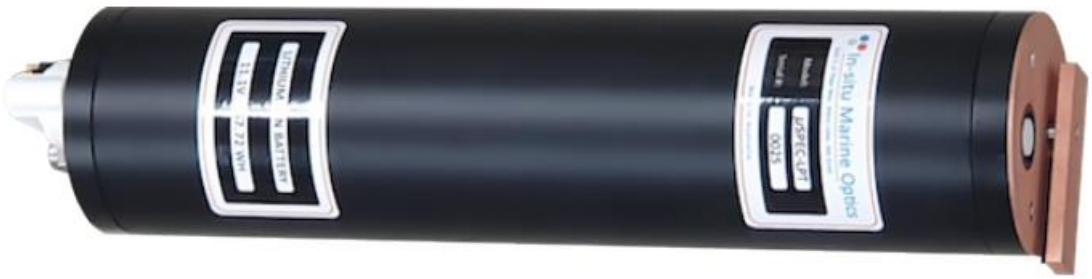
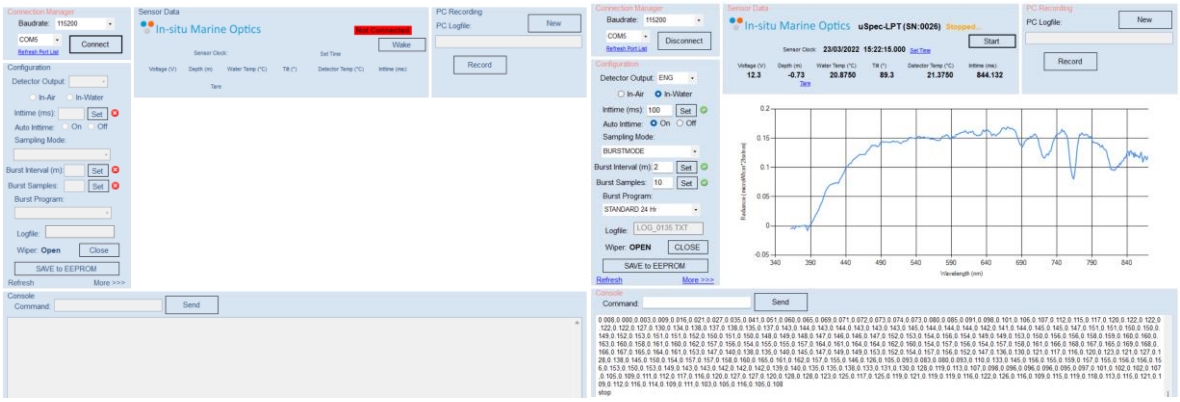
µSpec-LPTBW



Settings and Software

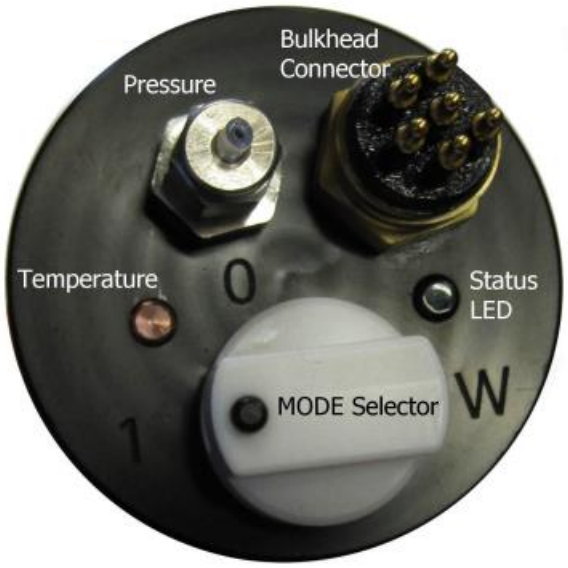
1. Connect instrument to computer using Serial to USB adaptor cable
2. Open the uSpec software (uSPEC-LPT_view_V3)
3. Click “Connect”, followed by “Wake”, followed by “Set Time” NOTE: the instrument needs to be powered on to wake.



4. Once connected the device will start streaming data to the chart and console.
5. Ensure “ENG” units is selected as Detector Output
6. Turn Auto Inttime “On”
7. Select whether the instrument will be used In-Air or In-Water
8. Settings for mooring deployment:
    a. Select Burst Sampling mode
    b. Set a time interval in minutes
    c. Set a number of measurements per burst
    d. Select burst program: Standard (24 hour), Sunrise/Sunset (only measures after local sunrise and until sunset), User Defined (only measures between user defined times)
    e. Select Save to EEPROM
9. Settings for profile deployment:
    a. Select Continuous Sampling mode
    b. Select Save to EEPROM
10. Select “Disconnect” and disconnect the instrument from the USB cable

Basic Operation

1. To switch on the instrument turn the mode selector counter clockwise until the Status LED lights up green



2. The wiper will move and the device is now measuring with the settings applied.
3. Ensure the dummy connector is securely attached to the bulkhead connector before submerging the instrument

Charging the Instrument

1. Connect the cables as shown

2. The LED on the “turtle” will flash traffic light colours before settling on orange to indicate charging.
3. The LED will turn green when the instrument is fully charged.
4. Should the LED turn red or flash consult manufacturer’s handbook.

Download Data

1. Turn the switch clockwise until aligned with ‘W’, the LED should light up green then orange
2. Connect to the broadcast wifi network ‘’uSpec-LPT (SN:0026)”
3. Navigate to the I.P. address of the device: http://192.168.1.1/
4. Click on the file you wish to download.
5. If there are issues with the wifi download feature, a serial download can be carried out (not advised if it can be avoided as it is very time consuming)
    a. Connect the instrument to the computer using the USB cable
    b. Type “LS” into the console command box to see all files stored on the instrument
    c. Select “New” in the PC Recording section and type in a filename to save the data
    d. Select “Record”

e. Type “GET LOG_XXXX.TXT” into the console command box to start streaming the desired file to the pc from the instrument
f. Once streaming is complete the new log file will contain the contents of the file on the instrument.
g. Streaming will display every measurement on the chart, meaning large files will take a very long time to stream

Python Code (See Below to Copy and Paste)

Most up to date version available at: https://github.com/Stephen-Grant/uSpec

1. Open uSpec_simple_V1.0.py in your IDE of choice
2. Type in filepath (Line 16)
3. Select Run
4. Three files will be saved containing, all collected Spectra, all measurements with a calculated Epar measurement, averaged measurements from burst mode measurements.

Contact stephen.grant@ntnu.no if there are any issues.

```
#uSPEC Data Reading and Processing
#S.Grant


#Import Libraries
import numpy as np
from matplotlib import pyplot as plt
import pandas as pd
from datetime import datetime
import scipy.integrate as integrate

#Today's date, the date you are processing data, not when it was
collected!
date = datetime.now().strftime("%Y%m%d_%H%M%S")

#Open the Log file and import all data in
with open(r"C:\Users\stepheg\Downloads\LEDstrip.log") as f: #Enter path
for LOG file here <----
   lis = [line.split() for line in f]

   wl=(lis[37][2])
#Extract Wavelengths and convert to array
   wlarray=(np.array(wl.split(",")))
   wlarray[0]=wlarray[0].translate({ord('{'): None})
   wlarray[268]=wlarray[268].translate({ord('}'): None})
   wlarray=wlarray.astype(np.float64)

   #Empty variables for holding data, pre allocated for speed

spectraEpar=[];specdate=[];spectime=[];specsig=[];specinttime=[];specde
pth=[];Eparlist=[];

Eparredlist=[];Epargreenlist=[];Eparbluelist=[];Eparbandstotallist=[];t
imestamp=[];

    #Saves spectra, date, integration time, signal %, depth.
    #Save integrated Epar in W/m^2
   for i in range(54,(len(lis)-1)):
      spec=np.array(lis[i][0].split(","))
      specdate.append(spec[2]);
      dati=spec[2]+" "+spec[3]
      timestamp.append(dati)
      spectime.append(spec[3]);
      specsig.append(spec[10]);
      specinttime.append(float(spec[9]))
      specdepth.append(float(spec[5]))
      spectrum=spec[11:280].astype(np.float64)
      Epar=(integrate.trapezoid(spectrum[15:146],wlarray[15:146]))*1E-
      6; #Convert from uW to W
      Epar=Epar*10000; #Convert from cm^-2 to m^-2
      Eparlist.append(Epar)
      spectraEpar.append(spectrum[15:146])

spectraEpararray=np.vstack(spectraEpar[1:len(spectraEpar)])

#First Epar Plot
plt.figure(0)
plt.plot(Eparlist,'k',marker=".")
```

```
plt.xlabel("Measurement Number")
plt.ylabel("Epar")

#Second Epar vs Depth Plot
plt.figure(1)
plt.scatter(Eparlist,specdepth)
plt.axhline(0,0,1)
plt.ylim(75,-5)
plt.xlabel("Epar")
plt.ylabel("Depth")

#Create dataframe to export all spectra
df0=pd.DataFrame(spectraEpararray)
df0.columns=wlarray[15:146]
df0.insert(0,'timestamp', timestamp[1:len(spectraEpar)])
df0.to_csv(lis[4][3][0:8]+'Spectra_'+date+'.csv')

#Create dataframe to export Epar data
result=[timestamp,specinttime,specdepth,Eparlist,specsig]
data=np.array(result)
df = pd.DataFrame(data=data.T, index=[np.linspace(1,len(data.T) ,
len(data.T))],
columns=["datetime","IntegrationTime","Depth","Epar","Signal %"])
df.to_csv(lis[4][3][0:8]+'DataFrame_'+date+'.csv')

#Remove empty data points
indices=np.argwhere(np.isnan(Eparlist))
timestamp=np.delete(timestamp,indices,0)
Eparlist=np.delete(Eparlist,indices,0)

#Create dataframe for average of burst measurements
result2=[timestamp,Eparlist]
data2=np.array(result2)
df2 = pd.DataFrame(data=data2.T, index=[np.linspace(1,len(data2.T) ,
len(data2.T))], columns=["datetime","Epar"])
df2['datetime'] = pd.to_datetime(df2['datetime'])
df2["Epar"] = pd.to_numeric(df2["Epar"], downcast="float")
burstsum=df2.resample('T', on='datetime').Epar.sum()
burstsum = burstsum.replace(0, np.nan).dropna()
avg=df2.resample('T', on='datetime').Epar.mean()
avg = avg.dropna()

#Third Average Epar plot
plt.figure(2)
avg.plot(y='Epar', use_index=True)
#burstsum.plot(y=Epar, use_index=True)
plt.show()
avg.to_csv(lis[4][3][0:8]+'Average_'+date+'.csv')
```