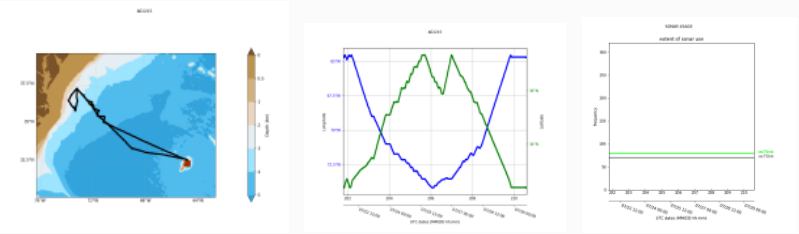
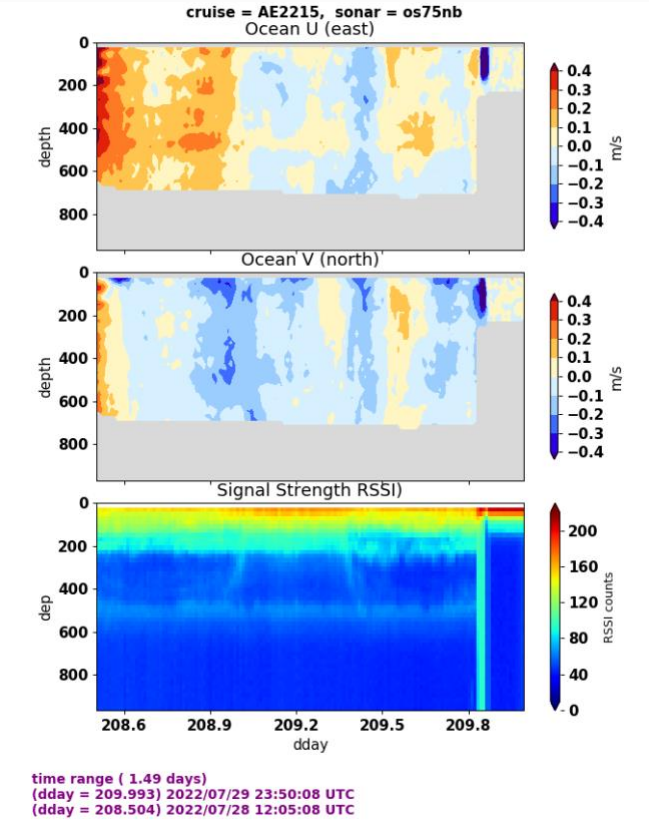
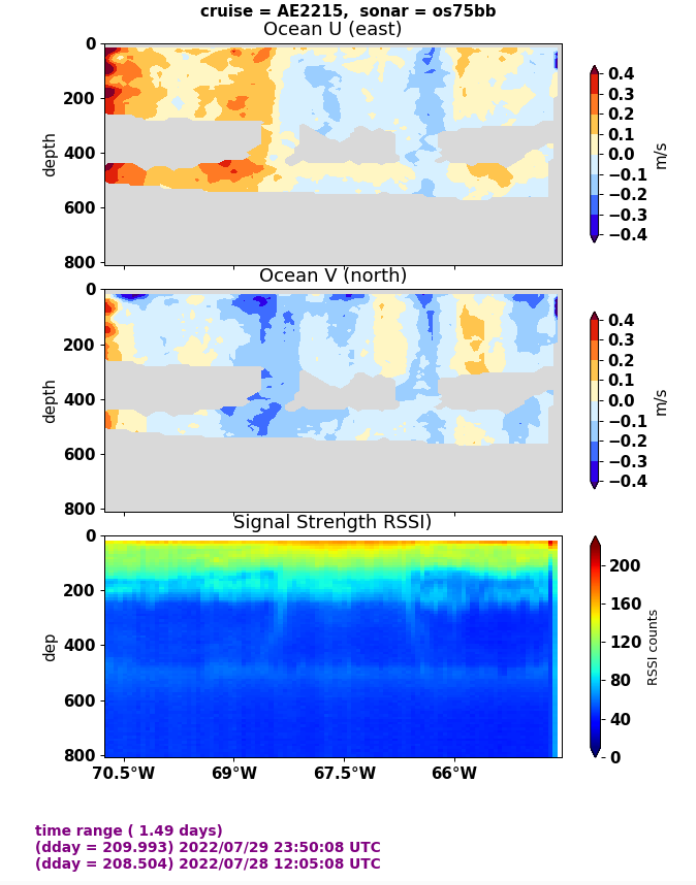
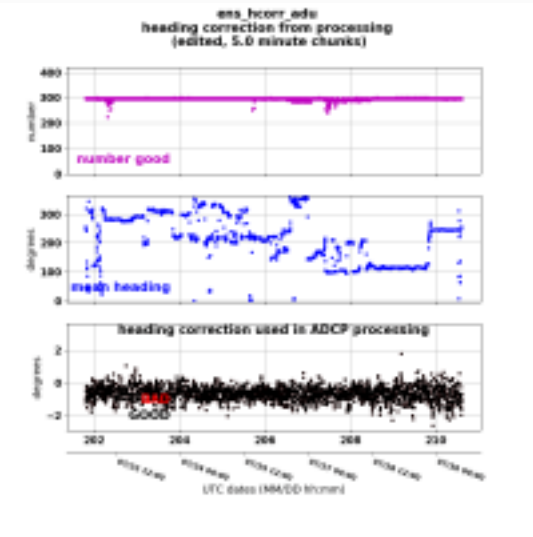
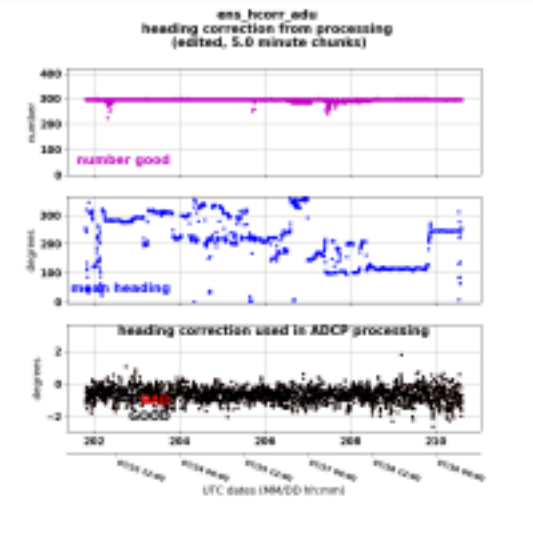
FIGURE-CARING EUROFLEETS (AE2215) SADCP summary report

CATEGORY	DESCRIPTION	FILE
overview		nav_plot_all_topo.png nav_plot_all_txy.png db_timerange.png (Appendix I)
overview	========== ======= start of report for /home/data/AE2215 ======= ========== ------raw------- raw: abxtwo 107 files (ae2022_201_68107 - ae2022_210_50400) raw: adu800 107 files (ae2022_201_68107 - ae2022_210_50400) raw: jrc_hdg4800 107 files (ae2022_201_68107 - ae2022_210_50400) raw: jrc_port 107 files (ae2022_201_68107 - ae2022_210_50400) raw: jrc_stbd 107 files (ae2022_201_68107 - ae2022_210_50400) adcp: os75 .raw 107 files (ae2022_201_68107 - ae2022_210_50400) adcp: os75 .raw.log 107 files (ae2022_201_68107 - ae2022_210_50400) adcp: os75 .raw.log.bin 107 files (ae2022_201_68107 - ae2022_210_50400) -----rbin------- rbin: abxtwo:adu 107 files (ae2022_201_68107 - ae2022_210_50400) rbin: abxtwo:gps 107 files (ae2022_201_68107 - ae2022_210_50400) rbin: adu800:adu 107 files (ae2022_201_68107 - ae2022_210_50400) rbin: adu800:gps 107 files (ae2022_201_68107 - ae2022_210_50400) rbin: jrc_hdg4800:hdg 107 files (ae2022_201_68107 - ae2022_210_50400) rbin: jrc_port:gps 107 files (ae2022_201_68107 - ae2022_210_50400) rbin: jrc_port:hdg 107 files (ae2022_201_68107 - ae2022_210_50400) rbin: jrc_stbd:gps 107 files (ae2022_201_68107 - ae2022_210_50400) rbin: jrc_stbd:hdg 107 files (ae2022_201_68107 - ae2022_210_50400) -----gbin------- gbin: os75 (abxtwo,jrc_hdg4800,adu800,time,jrc_port,jrc_stbd)	uhdas_overview.txt

	`` ------ database time ranges -------- proc: os75bb 201.792 - 210.608 (2022/07/21 to 2022/07/30) proc: os75nb 201.792 - 210.608 (2022/07/21 to 2022/07/30) ---------- end of report for /home/data/AE2215 ------ ```	
processed ADCP		Example figure (full dataset os75nb.nc)
processed ADCP		Example figure (full dataset os75bb.nc)

processed ADCP	========== ======= start of report for /home/data/AE2215 ======= ========== ================ ADCP calibrations ================ cal: os75nb BT none cal: os75nb WT Number of edited points: 35 out of 42 cal: os75nb WT median mean std cal: os75nb WT amplitude 1.0020 1.0009 0.0095 cal: os75nb WT phase 0.0970 0.0173 0.7728 ---------- cal: os75nb DXDY guessing ADCP (dx=starboard, dy=fwd) meters from GPS cal: os75nb DXDY positions from a_ae.gps cal: os75nb DXDY calculation done at 2022/07/30 14:35:33 cal: os75nb DXDY xducer_dx = 0.348232 cal: os75nb DXDY xducer_dy = 0.386815 cal: os75nb DXDY signal = 982.958890 ---------- cal: os75bb BT none cal: os75bb WT Number of edited points: 28 out of 40 cal: os75bb WT median mean std cal: os75bb WT amplitude 1.0040 1.0011 0.0143 cal: os75bb WT phase -0.0275 0.0214 0.8076 ---------- cal: os75bb DXDY guessing ADCP (dx=starboard, dy=fwd) meters from GPS cal: os75bb DXDY positions from a_ae.gps cal: os75bb DXDY calculation done at 2022/07/30 14:35:23 cal: os75bb DXDY xducer_dx = 2.280338 cal: os75bb DXDY xducer_dy = 5.609641 cal: os75bb DXDY signal = 982.960916 ---------- ---------- end of report for /home/data/AE2215 ------	ADCP_cals.txt

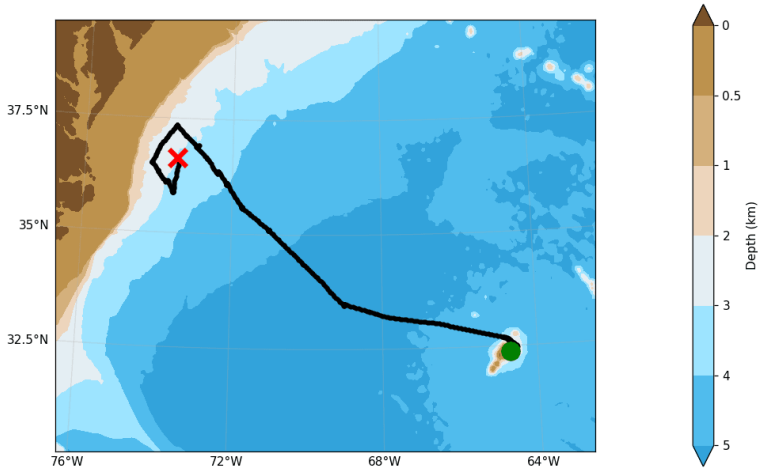
quality	 	os75nb_hcorr.png os75bb_hcorr.png
raw ADCP	``` ========== ======= start of report for /home/data/AE2215 ======= ========== ======= os75 settings: chunked by configuration ==== index num startdday enddday BT (ping, nbins,binsize, blank, pulse) (...) ---- --- --------- -------- --- 0 107 201.788327 210.608986 off (bb, 100, 8.0, 8.0, 8.0) (nb, 60, 16.0, 8.0, 16.0) ------- end of report for /home/data/AE2215 --------- ```	ADCP_settings.txt
UHDAS settings	ADCP CODAS processing settings	AE2215_proc.py.txt (Appendix II)
UHDAS settings	UHDAS serial acquisition settings	AE2215_sensor.py.txt (Appendix III)

References

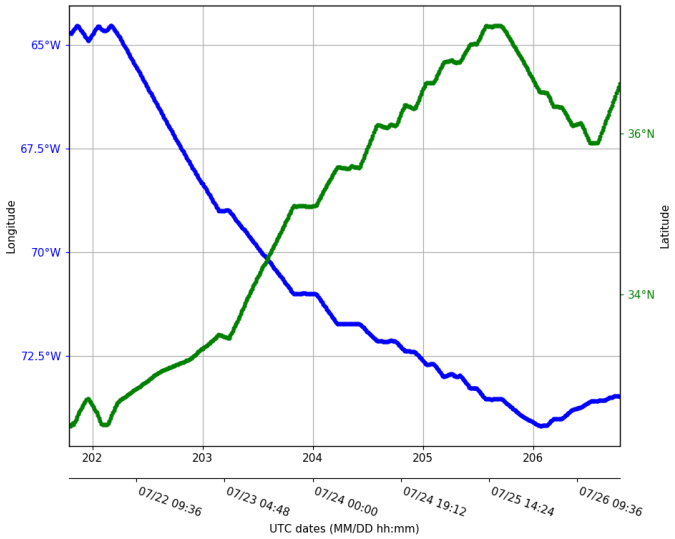
Firing, E and Hummon, J. M. (2010) Shipboard ADCP Measurements. In, The GO-SHIP Repeat Hydrography Manual: A Collection of Expert Reports and Guidelines. Version 1, (eds Hood, E.M., C.L. Sabine, and B.M. Sloyan). 11pp. (IOCCP Report Number 14; ICPO Publication Series Number 134). Available online at: http://www.go-ship.org/HydroMan.html. DOI: https://doi.org/10.25607/OBP-1352

More info about the FIGURE-CARING cruise at https://doi.org/10.17600/18002940.

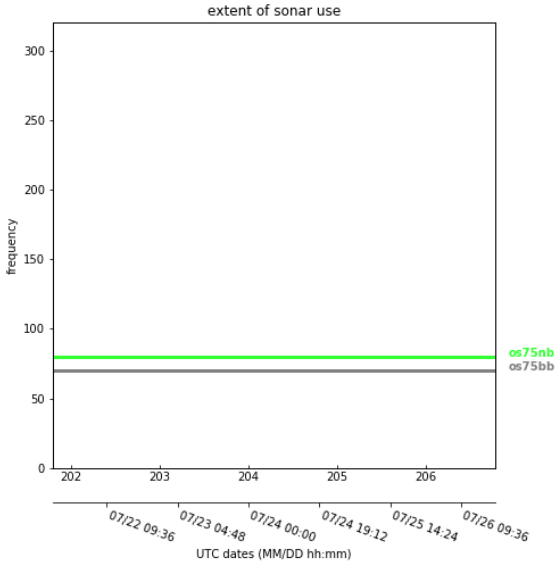
Appendix I



nav_plot_all_topo.png



nav_plot_all_txy.png



db_timerange.png

Appendix II

AE2215_proc.py.txt

```
shipname = 'Atlantic Explorer'
cruiseid = 'AE2215'
yearbase = 2022
uhdas_dir = '/home/data/AE2215'

# from proc_cfg.*:

## for processing
##----------------
## ship name: shipname = "Atlantic Explorer"
## at-sea "proc_cfg.*" initialized date = "2022/02/22 03:35:19"
##
## This file starts as /home/adcp/config/proc_cfg.py or .toml and
## includes the following information.  Uncomment, left-justify
## and fill these in if you are attempting to generate proc_cfg.*
## from this template.  The file must be named {cruiseid}_proc.py or *.toml
## or for this example, kk1105_proc.py or kk1105_proc.toml.
##
## example values: fill in for your cruise...
#
# yearbase = 2011               # usually year of first data logged
# uhdas_dir = "/home/data/kk1105"  # path to uhdas data directory
# shipname = "Ka`imikai O Kanaloa" # for documentation
# cruiseid = "kk1105"            # for titles
#

#======== serial inputs =========

# choose position instrument (directory and rbin message)

pos_inst = "jrc_port"
pos_msg = "gps"

# choose attitude instruments (directory and rbin message)

pitch_inst = "abxtwo"     # pitch is recorded, but NOT used in transformation
pitch_msg = "adu"      # disable with "" (not None)

roll_inst = "abxtwo"      # roll is recorded, but NOT used in transformation
roll_msg = "adu"       # disable with "" (not None)

hdg_inst = "jrc_hdg4800"       # reliable heading, used for beam-earth transformation
hdg_msg = "hdg"


## heading correction
## all heading+msg pairs, for hbin files
hdg_inst_msgs = [
   ('jrc_hdg4800', 'hdg'),
```

```
   ('abxtwo', 'adu'),
   ('adu800', 'adu'),
   ('jrc_port', 'hdg'),
   ('jrc_stbd', 'hdg'),]

## instrument for heading correction to ADCP data (dir and msg)
hcorr_inst = "abxtwo"       # disable with "" (not None)
hcorr_msg = "adu"       # disable with "" (not None)
hcorr_gap_fill = -1   ## fallback correction for hcorr gaps
                ## calculate hdg_inst - hcorr_inst, eg gyro - ashtech
                ## SAME SIGN CONVENTION as cal/rotate/ens_hcorr.ang

## if there is a posmv
acc_heading_cutoff = 0.02

# ========== ADCP transformations========

# heading alignment:  nominal - (cal/watertrack)
h_align = dict(
    os75 = 45.2,)

# transducer depth, meters
ducer_depth = dict(
    os75 = 3,)

# velocity scalefactor
# see SoundspeedFixer in pycurrents/adcp/pingavg.py
scalefactor = dict(
   os75bb = 1.0,
   os75nb = 1.0,)

# soundspeed
# Soundspeed is usually None, and should ALWAYS be left as None for Ocean Surveyor
# (it is remotely possible that soundspeed for a WH, BB, or NB might need to
#         be set to a number, but usually that just results in an erroneous
#         scale factor.
soundspeed = dict(
   os75bb = None,
   os75nb = None,)

# salinity
salinity = dict(
   os75bb = None,
   os75nb = None,)

#=================================================================
# =========          values for quick_adcp.py         ==========
# ========= These are set here for at-sea procesing,    ==========
# ========= but are REQUIRED in quick_adcp.py control   ==========
# =========  file for batch mode or reprocessing.      ==========

## choose whether or not to use topography for editing
## 0 = "always use amplitude to guess the bottom;
##         flag data below the bottom as bad"
```

```
## -1 = "never search for the bottom"
## positive integer: Only look for the bottom in deep water, where
##      "deep water" is defined as "topo database says greater than this".

max_search_depth = dict(
   os75bb = 2000,
   os75nb = 2000,)

# special: weakprof_numbins
weakprof_numbins = dict(
   os75bb = None,
   os75nb = None,)

# set averaging intervals
enslength = dict(
   os75bb = 300,
   os75nb = 300,)

# Estimate of offset between ADCP transducer and gps:
# - Specify integer values for 'xducer_dx' and 'xducer_dy' for each instrument
# - xducer_dx = ADCP's location in meters, positive starboard with the GPS
#   location as origin
# - xducer_dy = ADCP's location in meters, positive forward with the GPS
#   location as origin
#
# There should be one set of xducer_dx, xducer_dy values per instrument
# Ex. (python version):
#   xducer_dx = dict(
#   wh300 = -2,
#   os38 = 16, )
# Ex. (toml version)
#   xducer_dy = { wh300 = 1, os38 = 6 }
#
# Note that estimates of xducer_dx, xducer_dy can be found in
# cal/watertrk/guess_xducerxy

xducer_dx = dict(
     os75 = 0,)
xducer_dy = dict(
     os75 = 0,)

## If there is a bad beam, create a dictionary modeled after
## enslen (i.e. Sonar-based, not instrument based) and use the
## RDI number (1,2,3,4) to designate the beam to leave out.
```

Appendix III

AE2215_sensor.py.txt

```
# This configuration file is Python code.  You should not
# need to change it; but if you do, you need to know that
# in Python, the *indentation matters*.

# The following will normally be empty lists, like this:
#
#ignore_ADCPs = []
#ignore_other_sensors = []
#
# But if you want to run with only a subset of the ADCPs
# and/or ancillary sensors that are defined in this file,
# you can list the ones you want to ignore like this:
#
#ignore_ADCPs = ['wh300', 'os75']
#ignore_other_sensors = ['GPS']
#
# In this case, you are listing the 'instrument' field of each
# ADCP or sensor you wish to exclude.
#

ignore_ADCPs = []
ignore_other_sensors = []
use_publishers = False
#use_publishers = True

# 2-letter abbreviation for logging file prefix and constructing dbase name;
# read by procsetup.py
shipabbrev = "ae"

ADCPs = [
      { 'instrument'  :  'os75',
        'setup'      :  'rdi_setup',     ## RENAME ?? rdi_setup.py
        'terminal'    :  'oswh_term',
        'defaultcmd'  :  'os75_default.cmd',
        'commands'    :  ('EA04500',),
        'datatypes'   :  ('os75bb', 'os75nb'),
        'wakeup_baud' :  9600 }, ]

# Do not edit the next three lines:
common_opts = '-f %s -F -m 1 -H 2 ' % (shipabbrev,)
nb_opts =   '-lE -c -I'         # raw data, write a log file, log errors
oswh_opts = '-lE -c -O -I'      # -O for OS/WH data format
                        # -c for checksum, -I to initiate pinging

sensors = [
      { 'instrument'  :  'os75',      # Passive logging of OS
        'device'     :  'ttyUSB3',
        'baud'       :   19200,      # Some errors at 115200.
        'format'     :  'binary',
        'subdir'     :  'os75',
```

```
  'ext'       : 'raw',       # zmq for speedlog:
  'opt'       : oswh_opts, },


{ 'instrument'  :  'ADU800',
  'device'     :  'ttyUSB5',
  'baud'       :   115200,
  'format'     :  'ascii',
  'subdir'     :  'adu800',   ## do not change dir name
  'ext'        :  'paq',
  'strings'    :  ('$GPGGA', '$PASHR,ATT'),
  'messages'   :  ('gps',  'adu',),
  'opt'        :  '-c -Y2'},    # $PYRTM

{ 'instrument'  :  'ABXTWO',
  'device'     :  'ttyUSB1',
  'baud'       :   115200,
  'format'     :  'ascii',
  'subdir'     :  'abxtwo',   ## do not change dir name
  'ext'        :  'adu',
  'strings'    :  ('$GPGGA', '$PASHR,ATT'),
  'messages'   :  ('gps',  'adu',),
  'opt'        :  '-c -Y2'},    # $PYRTM


{ 'instrument'  :  'JRC_hdg_4800',
  'device'     :  'ttyUSB4',
  'baud'       :   4800,
  'format'     :  'ascii',
  'subdir'     :  'jrc_hdg4800',   #reliable heading device
  'ext'        :  'hdg',
  'strings'    :  ('$GPHDT',),
  'messages'   :  ('hdg',),
  'opt'        :  '-c -Y2'},    # $PYRTM

{ 'instrument'  :  'JRC_JLR21_port',
  'device'     :  'ttyUSB0',
  'baud'       :   38400,
  'format'     :  'ascii',
  'subdir'     :  'jrc_port',    # primary position device
  'ext'        :  'hdg',
  'strings'    :  ('$GPHDT', '$GPGGA',), # others were not useful
  'messages'   :  ('hdg','gps'),
  'opt'        :  '-c -Y2'},   #JRC JLR-21 GPS


{ 'instrument'  :  'JRC_JLR21_stbd',  # spare
  'device'     :  'ttyUSB7',
  'baud'       :   38400,
  'format'     :  'ascii',
  'subdir'     :  'jrc_stbd',
  'ext'        :  'hdg',
  'strings'    :  ('$GPHDT', '$GPGGA',), # others were not useful
  'messages'   :  ('hdg','gps'),
```

```
        'opt'        :  '-c -Y2'},   #JRC JLR-21 GPS  ]


## enabling or disabling this occurs in uhdas_cfg.py
## speedlog: enabled in /home/adcp/config/uhdas_cfg.py
speedlog_config = {
   'instrument'       : 'os75',
   'serial_device'    : '',    # no serial out (else: '/dev/ttyUSB1',)
   'baud'             : 9600,
   'eth_port'         : 'eno2',   # this is the NIC to serve web_speedlog.py
   'zmq_from_bin'     : "tcp://127.0.0.1:38010",  # port to read os75 indexing
   'pub_addr'         : "tcp://127.0.0.1:38020",  # port to publish speedlog #NO SPACES
#    'heading_offset'    : xx.x,  # uses h_align; better would be from EA_estimator.py
   'scale'            : 1.0,     # multiplies velocity measurement
   'bins'             : (1,3),   # zero-based; input to python slice()
   'navg'             : 5,       # pings to average}

## this section describes...
publishers = [
           {
           'subdir'     :  'jrc_port',
           'pub_addr'    :  'tcp://127.0.0.1:38000', # uses this port
           'sample_opts' :  '-c -Y2 -s60',     # $PYRTM}, ]

#### DO NOT CHANGE the following ################

ADCPs = [A for A in ADCPs if A['instrument'] not in ignore_ADCPs]
sensors = [S for S in sensors if S['instrument'] not in ignore_ADCPs]
sensors = [S for S in sensors if S['instrument'] not in ignore_other_sensors]

from uhdas.uhdas import make_publishers

# append zmq_from_bin port to opts in appropriate ADCP sensor block
make_publishers.modify_sensors_for_speedlog(sensors, speedlog_config)

# substitute publishers into sensors for zmq publishing
if use_publishers==True:
   make_publishers.modify_sensors_and_publishers(sensors, publishers)
```