|  |  |  |  |
| --- | --- | --- | --- |
| **SNo** | **Name of the Function/Subroutine** | **Purpose of the Function/Subroutine in the Argo RT processing** | **Calls which other routine(s)** |
| 1. | addIridiumPositions.m | addIridiumPositions(wmoID,pn)  this function takes the wmoID and profile number for a float and asks you for the iridium position for that profile. It then does all the bookkeeping required to use this position. | Calls: getargo.m  argoprofile\_nc.m  interpolate\_locations.m  locationplots.m  web\_float\_summary.m  Called by: None |
| 2. | argoprofile\_Bfile\_nc.m | ARGOPROFILE\_BFILE\_NC Create or add to a netCDF Argo RAW profile file. May be used for single or multi-profile files (but for now expect only the former)  INPUT :  dbdat - master database record for this float  fp - float struct array containing ONLY the profiles to be added.  OUTPUT  To a single netCDF file. If only one profile, the file name will include  its profile number, otherwise it will just have float WMO number. | Calls:  netCDF toolbox  overall\_qcflag.  Called by:  argoprofile\_nc.m |
| 3. | argoprofile\_nc.m | ARGOPROFILE\_NC Create or add to a netCDF Argo RAW profile file. May be used for single or multi-profile files (but for now expect only the former)  INPUT : dbdat - master database record for this float. fp - float struct array containing ONLY the profiles to be added.  OUTPUT : To a single netCDF file. If only one profile, the file name will include its profile number, otherwise it will just have float WMO number. | Calls:  netCDF toolbox  overall\_qcflag.m  argoprofile\_Bfile\_nc.m  Called by:  Fixoxy.m  interpolate\_locations.m  process\_iridium.m  process\_profile.m  rejectpoints.m  rerunoxygenprofiles.m |
| 4. | best\_est\_ascent\_end.m | BEST\_EST\_ASCENT\_END Fill in any gaps in jday\_ascent\_end (time float surfaced) fields.  INPUT: fpp - float structure array  OUTPUT: t0 - estimate of "ascent end" for every profile in fpp. Only differ from jday\_ascent\_end when there are some good and some missing values in that field, or if all missing [then replaced by jday(1) values.] | Calls: None  Called by:  trajectory\_nc.m |
| 5. | betasw\_ZHH2009.m | volume scattering at angles defined by theta. | Calls:  RInw(), BetaT(),rhou\_sw(), dlnasw\_ds.m  Called by:  convertBbsig.m  metadata\_nc.m |
| 6. | bin2num.m | BIN2NUM Convert vector of 0 and 1 to decimal (differs from BIN2DEC which works from string binary) | Calls: Round, Reshape  Called by:  decode\_provor.m |
| 7. | blankfloat.m | blankfloat creates a blank entry in the float mat files for empty profiles. | Calls: fieldnames() |
| 8. | bphase\_conv.m | Function to convert bphase to dissolved oxygen (old script) | Calls: None  Called by:  convertBphase.m |
| 9. | calc\_ascent\_end.m | CALC\_ASCENT\_END Calculate Ascent End time - method differs according to make of float.  INPUT b1t - Block1 time info (b1tim.dat)  maxblk - number of blocks in profile message  dbdat - db record for this float  fp - profile structure  verbose - 1=dump descriptive stuff  0=shut up [default]  OUTPUT jae - estimate julian day ascent end time. | Calls:  datestr()  Called by:  process\_profile.m |
| 10. | calcFSIsalt.m | Calculate salinity from FSI sensor | Calls:  sw\_salt()  Called by: None |
| 11. | calc\_ice\_cv.m | calculate ice detection critical values... | Calls:  getdbase.m  getargo.m  Called by: None |
| 12. | calc\_TST\_JD.m | This Dunn method just fits a regression through all the repeat\_num/juld points, rather than estimating the rep rate from all pairs of points. This seems less prone to first point error and other analysis errors. However it could stuff up when have points from two cycles - but if that occurs we have major problems anyway! | Calls: None  Called by:  load\_traj\_apex\_argos.m  load\_traj\_apex\_iridium.m |
| 13. | calc\_TST.m | Compute TST according to argo\_DAC\_cookbook\_V1.4, Annex B, sec 6.2.  INPUT juld - block 1 julian days  nb - msg numbers corresp to juld  OUTPUT TST - single TST value  stat - 0=good, 2=dodgy | Calls: None  Called by:  load\_traj\_apex\_argos.m  load\_traj\_apex\_iridium.m |
| 14. | calibrate\_p.m | Calibrate pressure  INPUT fpp - struct array of profiles for one float.  ll - profile to calibrate [default - the last one]  OUTPUT fpp - with p\_calibrate field loaded, and for Webb floats,  the .surfpres\_used and .surfpres\_qc fields loaded. | Calls: qc\_apply(),sw\_cndr(), sw\_ptmp(), sw\_salt()  Called by:  argoprofile\_nc.m  process\_iridium.m  process\_profile.m |
| 15. | calsal\_clim.m | Calibrate salinity in raw (near-realtime) Argo float, and load variables 's\_calibrate', 'cndc\_raw', 'cndc\_qc', 'cndc\_calibrate'  INPUT  float - struct array of profiles for one float. It is assumed that  profiles have already been QCed.  ical - index to those to be calibrated [if not supplied, will calibrate  all profiles after the last with a non-empty 'c\_ratio' field  OUTPUT  nfloat - copy of 'float', but with 's\_calibrate' and all conductivity  profile variables loaded.  cal\_report 6 diagnostic values for the last profile calibrated  1 - "theta" (min theta of near-bottom values)  2 - num profile potential T values in cal range  3 - num reference CTDs used  4 - range of c\_ratio estimates [large value (>.0005?) may indicate  problems in profile data]  5,6 supplied if can calculate a calibration  5 - correction as applied to S value at top of cal range  6 - threshold: median STD(S) [plus measure of local spatial  variability in clim S estimates at theta?] | Calls: None  Called by:  calsal.m |
| 16. | calsal.m | Same as calsal\_clim.m | Calls: None  Called by:  argoprofile\_nc.m  process\_iridium.m |
| 17. | calsal\_Profile2.m | Calibrate salinity in raw (near-realtime) Argo float, and load variables 's\_calibrate', 'cndc\_raw', 'cndc\_qc', 'cndc\_calibrate'  this has been modified to apply the master calibration to a subset  % of salinity associated with a second, lower resolution profile | Calls: sw\_cndr(), sw\_salt()  Called by:  argoprofile\_nc.m  argoprofile\_Bfile\_nc.m |
| 18. | celltm\_sbe41.m | Given a profile of SBE-41 (or SBE-41CP) data consisting of salinity  % (salt, pss-78), temperature (temp, deg. C, ITPS-68), pressure (pres,  % dbar), and the elapsed time of the samples (e\_time, seconds), this  % function returns a corrected salinity (salt\_cor, pss-78). | Calls: sw\_cndr(),sw\_salt()  Called by:  thermal\_lag\_calc.m |
| 19. | checkOAfailures.m | Checks for the corrections suggested by Coriolis and applied the correction if accepted. | Calls: qc\_apply.m  Called by: None |
| 20. | check\_profile.m | CHECK\_PROFILE Apply simple checks on new profile  DEV First thing to add to this is a lenient T/S outlier check  INPUT:  fp - profile structur | Calls: qc\_apply.m  Called by:  argoprofile\_nc.m  process\_iridium.m |
| 21. | combineDandRfileV3\_1LW6.m | Combine the V3.1 R files with the DModed old file. only the Delayed mode values will be added, Dfile is created. | Calls: getdbase(), getargo()  Called by: None |
| 22. | continents.m | Draw continents as patches with color COLOR edge color EDGE and NMIN points in coastline | Calls : None  Called by:  locationplots.m  set\_argo\_sys\_params.m |
| 23. | convertArgotoFloat.m | convertArgo\_desc.mat to float structure take the /home/reg2/argo/usmirror/argo\_desc.mat file and turn it into the Indian float structures, including all available metadata!!! (but not  technical data). though it could be modified to convert the technical  files as well if they exist.  usage: convertArgotoFloat  Note: countrycode must be set before you run this : 9=india | Calls: None  Called by: None |
| 24. | convertBbsig.m | To convert the raw backscattering values to scientific backscattering values using the coefficients.  INPUT  Bbsig - the raw BackScattering measurement from the float  wmoid - used to find the Backscattering coefficients for the float  being processed  BS - the derived Backscattering value | Calls: betasw\_ZHH2009()  Called by:  process\_iridium.m  fixoxyQC.m |
| 25. | convertBphase.m | Convert Bphase value to Dissolved oxygen in mm/kg  INPUT  Bphase - the raw Bphase measurement from the float  t - the temperature from the Seabird CTD (not the oxygen  temperature)  s - the salinity from the CTD  wmoid - used to find the O2 calibration coefficients for the float  being processed  pp - the pressure from the CTD - converted to depth for the final  calculation  lat - required for the pressure to depth conversion  OUTPUT  O2 - the derived oxygen saturation | Calls: getO2caldbase.m  Called by:  decode\_webb.m  fixoxy.m  process\_iridium.m  rerunoxygenprofiles.m |
| 26. | convertFREQoxygen.m | CALO2\_CLIM Calibrate oxygen in from the raw values reported by the Argo float, and load  variables 'o\_calibrate', and 'o\_raw'  INPUT  freqO2 - the raw oxygen measurement from the float  t - the temperature from the Seabird CTD (not the oxygen  temperature)  s - the salinity from the CTD  wmoid - used to find the O2 calibration coefficients for the float  being processed  pp - the pressure from the CTD -  OUTPUT  O2 - the derived oxygen saturation | Calls: getO2caldbase.m  Called by:  decode\_webb.m  metadata\_nc.m |
| 27. | convertFsig.m | Convert Fsig value to Chlorophyll values mg/l  INPUT  Fsig - the raw Fsig measurement from the float    wmoid - used to find the O2 calibration coefficients for the float  being processed  CHL - the derived chlorophyll value | Calls: getO2caldbase.m  Called by:  process\_iridium.m  fixoxyQC.m |
| 28. | convert\_matfiles.m | CONVERT\_MATFILES Convert an Argo pre-2006-format matfile to new format.  INPUT wmo - WMO id of float  ipth - path to old (input) files (include trailing '/')  opth - path to new (output) files (include trailing '/')  OUTPUT  new matfile created: [opth 'float' num2str(WMO)] | Calls: None  Called by: Stand alone |
| 29. | convertSBE63Oxy.m | Decodes the SBE 63 Oxygen sensors on the Navis floats  INPUT  Phase - the raw measurements from the float  t - the temperature from the Seabird CTD (not the oxygen  temperature)  s - the salinity from the CTD  wmoid - used to find the O2 calibration coefficients for the float  being processed  pp - the pressure from the CTD - converted to depth for the final  calculation  lat - required for the pressure to depth conversion  OUTPUT  O2 - the derived oxygen concentration (umol/kg) | Calls: getO2caldbase.m  Called by:  process\_iridium.m  metadata\_nc.m  fixoxy.m |
| 30. | convertSBEOxyfreq.m | this decodes the SBE Oxygen sensors  INPUT  Freq - the raw frequency measurement from the float  t - the temperature from the Seabird CTD (not the oxygen  temperature)  s - the salinity from the CTD  wmoid - used to find the O2 calibration coefficients for the float  being processed  pp - the pressure from the CTD - converted to depth for the final  calculation  lat - required for the pressure to depth conversion  OUTPUT  O2 - the derived oxygen saturation | Calls: None  Called by:  process\_iridium.m  rerunoxygenprofiles.m  process\_iridiumseparateDoxy.m |
| 31 | convertTphase.m | Optode type='Aanderaa 4330',  INPUT  Tphase - the raw Tphase measurement from the float  t - the temperature from the Seabird CTD (not the oxygen  temperature)  s - the salinity from the CTD  wmoid - used to find the O2 calibration coefficients for the float  being processed  pp - the pressure from the CTD - converted to depth for the final  calculation  lat - required for the pressure to depth conversion  OUTPUT  O2 - the derived oxygen saturation | Calls: None  Called by:  process\_iridium.m |
| 32. | convert\_uMolar.m | convert\_uMolar - takes raw dissolved oxygen in uM/l and converts to uM/kg for Argos processing  usage:  function [oxy]=convert\_uMolar(O2a,pp,lat);  where O2a = oxygen concentration in uM/l,  pp = pressure  s = salinity  t = temperature  lat = latitude of the profile | Calls: None  Called by:  decode\_webb.m  fixoxy.m  metadata\_nc.m  rerunoxygenprofiles.m |
| 33. | CRCcheck.m | CRCcheck - Carry out Cyclic Redundancy Check on a single message  INPUT In - a vector of number, the first being the CRC value  computed (according to make-dependant algorithms) for all  the rest.  dbdat float database struct including maker and subtype  OUTPUT ok 0=failed CRC check 1=CRC is correct | Calls: None  Called by:  find\_best\_msg.m |
| 34. | CSIRO\_copy\_iridium\_data.m | Moves files when successfully processed. |  |
| 35. | cullAllMissions.m | cullAllMissions  run this to create all mission files for all floats (only works for files  in the iridium\_processed directory) | Calls: getdbase(), cullMissions\_iridium()  Called by: None |
| 36. | cullInitialMission.m | cullInitialMissions  for floats where the mission is contained in the log files, the first  mission must be obtained from the .000.log file which is not usually  read. You must ensure that htis file is in the iridium\_processed  directory.  this script uses the data from cullMissions\_iridium and only obtains the data from the 000 file. | Calls: None  Called by:  cullMissions\_iridium.m |
| 37. | cullMissions\_iridium.m | cullMissions\_iridium  designed to gather mission information from iridium floats  this is then stored in float#####aux.mat so the mission number can calculated.  It can be used either to cull all missions from existing floats or add  missions from new profiles to an existing file.  Usage: cullMissions\_iridium(fname)  where fname contains the path to the iridium msg file for one profile  fname=-1 regenerates all missions from the iridium\_processed directory for a given flaot | Callls:  getargo()  cullInitialMission()  Called by:  cullInitialMission.m  cullAllMissions.m  process\_iridium.m  fixmissionFloatTech5.m |
| 38. | cullSingleMission.m | cullSingleMission  run this to create all mission files for a single floats (only works for files in the iridium\_processed directory | Calls: getdbase(), cullMissions\_iridium(). |
| 39. | date\_ftp\_file.m | DATE\_FTP\_FILE Scan an ftp download file and find newest ARGOS date-time (to estimate the UTC time of the ftp transfer)  INPUT  fid - FID of open argos message dump file (created by 'ftp\_get\_argos')  verbose - 1=give stats of dates found [default 0]  OUTPUTS  lstdate - datenum of last date in the file (ie newest UTC fix time)  NOTE: \*\*\* The input file is left open and at the start (ie is rewound) | Calls: None  Called by:  strip\_argos\_msg.m |
| 40. | decode\_provor.m | DECODE\_PROVOR reads block data from Provor float Argos transmissions and decode.  INPUT prof - decimal profile (in blocks and lines) from STRIP\_ARGOS\_MSG  dbdat- details for this float from the float master database  pos - the header structure containing the current profile  date/time  OUTPUT pro - structure containing profile and tech data | Calls: new\_profile\_struct()  Called by:  Process\_profile()  Trajectory\_nc() |
| 41. | decode\_webb.m | DECODE\_WEBB reads block data from Webb R1/Apex palace Argos transmissions and converts to real units.  INPUT prof - decimal profile (in blocks and lines) from STRIP\_ARGOS\_MSG  dbdat- details for this float from the float master database  OUTPUT fp - structure containing profile and tech data (empty if a fatal error has occurred.)  Also in THIS FILE: functions H2B CALC\_TEMP CALC\_SAL CALC\_VOLT | Calls: new\_profile\_struct()  Called by:  Process\_profile()  Process\_iridium()  Trajectory\_nc() |
| 42. | delete\_bom\_ftp.m | Blank | Calls: None |
| 43. | Dirc.m | DIRC Directory listing with cell output and optional arguments.  DIRC(DIR\_NAME) uses the DIR function to return the directory listing of directory DIR\_NAME, with usage of optional arguments, and with the following differences in output:  - The output is in a 'cell' format instead of a 'structure' format.  - The output includes the following columns (in the stated order):  Full Item Name, Item Name, Item Extension, Date, Bytes, IsDir | Calls: None  Called by:  extract\_Iridium\_data.m |
| 44. | distance\_lpo.m | Computes distance and bearing between points on the earth using WGS 1984 ellipsoid  [range, A12, A21] = distance\_lpo(lat, long) computes the ranges RANGE between points specified in the LAT and LONG vectors (decimal degrees with positive indicating north/east). Forward and reverse bearings (degrees) are returned in AF, AR.  Ellipsoid formulas are recommended for distance d<2000 km, but can be used for longer distances. | Calls: None  Called by: jamstec\_position\_test.m |
| 45. | drive\_argo\_manual.m | Script to drive ArgoRT in various test modes  Expected that the user will modify this script | Calls:  strip\_argos\_msg.m  web\_processing\_report.m  getargo.m  web\_float\_summary.m  Called by:  None |
| 46. | drive\_rt\_argo.m | This MUST be run within the Argo RT root directory  Plotting routines especially can throw up warnings - so make sure the session doesn't halt in the debugger if such a warning occurs | Calls:  set\_argo\_sys\_params.m  strip\_argos\_msg.m  web\_processing\_report.m  export\_argo.m  Called by: None |
| 47. | drive\_web\_report.m | Generate the web reports. | Calls:  web\_processing\_report.m  called by: None |
| 48. | edit\_workfile.m | RUN\_FROM\_WORKFILE Interactively edit, test and process Ago profiles from 'workfiles'. The only editing allowed is setting or clearing the bad-data flags on any line of input message (and the special case of auto-correcting message-number-wrap.) Changes are not permanent until either the 'save' or 'accept' commands are used.  If have a block of profiles to just run through the processing system, better to just use RUN\_FROM\_WORKFILE Command "?" provides a detailed Help message.  INPUT:  n\_or\_p 0 = "np" specified in matfile record index (n)  1 = "np" specified in Argo profile number (p) [default]  showhex 0 = dispaly raw messages as decimal [default]  1 = display raw messages in hex  rt\_prec 0 = Processing records to "reprocessing\_records.mat" [default]  1 = " " RT records file.  OUTPUT:  edited workfiles, and the usual products if any profiles submitted for processing. | Calls:  set\_argo\_sys\_params.m  pos\_fix\_check.m  getdbase.m  called by: None |
| 49. | export\_argo.m | EXPORT\_ARGO Read float processing records to find which TESAC messages and GDAC netCDF files to transmit, organise transmission, update records.  INPUT  reads Argo\_proc\_records.mat, finds .tesac and .nc files  OUTPUT  updates Argo\_proc\_records.mat, emails tesac, calls writeGDACfiles to ftp the netCDF files | Calls:  set\_argo\_sys\_params.m  getdbase.m  called by:  drive\_rt\_argo.m  strip\_argos\_msg.m  trajectory\_nc.m |
| 50. | export\_text\_files.m | exporttextfiles - reads the mat files and generates a text file for archiving in ./textfiles/R590????? | Calls: None  Called by:  process\_iridium.m  process\_profile.m |
| 51. | extract\_argos\_msg.m | EXTRACT\_ARGOS\_MSG Extract the messages for specified floats from an ftp download from Argos, or just make a copy of an ftp download file, but adding UTC date as the first line.  INPUT  fnm - name of argos message dump file (created by script 'ftp\_get\_argos')  Do not give path - file is assumed to be in the usual spot.  flist - list of \*ARGOS\* id numbers for the float(s) to extract  If empty, will just make a copy of the input file, but add the  UTC last-fix-time as the first record (if not already in file)  OUTPUT  a single file, named with the parent file prefix and the first requested  id number, ie fnm '\_' flist(1) '.log', and placed in 'work/'  If no 'flist' then "\_new" will the appended to the fnm prefix. | Calls: None  Called by:  None (stand alone) |
| 52. | extract\_Iridium\_data.m | extract Iridium data - checks the delivery directories and, if there is  new data, processes it before sending strip\_argos\_msg | Calls:  BOM\_retrieve\_Iridium.m  mail\_out\_iridium\_log\_error.m  idcrossref.m  getdbase.m  process\_iridium.m  new\_proc\_rec\_struct.m  mail\_out\_dead\_float.m  interpolate\_locations.m  CSIRO\_copy\_iridium\_data.m  BOM\_send\_iridium\_notify.m  Called by:  Strip\_argos\_msg.m |
| 53. | extract\_phy\_data.m | extract phy data (Solo Polynya data) - checks the delivery directories and, if there is new data, processes it before ending strip\_argos\_msg | Calls:  Idcrossref.m  Getdbase.m  process\_phyfiles.m  new\_proc\_rec\_struct.m  BOM\_send\_iridium\_notify.m  Called by:  Strip\_argos\_msg.m |
| 54. | extract\_Solo2\_data.m | extract Iridium data - checks the delivery directories and, if there is  new data, processes it before sending strip\_argos\_msg | Calls:  Getdbase.m  new\_profile\_struct.m  new\_tech\_struct.m  addSolo2Iridium.m  fill\_float\_from\_solo2.m  calibrate\_p.m  qc\_tests.m  calsal.m  web\_profile\_plot.m  argoprofile\_nc.m  metadata\_nc.m  techinfo\_nc.m  time\_section\_plot.m  waterfallplots.m  locationplots.m  tsplots.m  web\_float\_summary.m  called by:  extract\_Iridium\_data.m  strip\_argos\_msg.m |
| 55. | fetchloginfo.m | note - you must set i before starting - use 1 if the first time you  run... you also need to setwmoID, i=1 | Calls:  Getargo.m  find\_argos\_download.m  called by:  None (Stand alone) |
| 56. | find\_argos\_download.m | [file]=find\_argos\_download(wmo\_id,pn)  This function is designed to scan the argos\_downloads directory and find the correct argos download to process a given profile from a given float.  It works by doing a grep of what it 'guesses' is the right download and then working forward until it sees two successive downloads with the same number of lines for that float. It then uses the second. This might take time but should save in the end. The downside is that a download needs to be present and UNCOMPRESSED before it will work. AT  usage: [file]=find\_argos\_download(wmo\_id,pn)  where: file is the filename returned - can be array if you supply an array of pn's wmo\_id is the identifier for a flaot and pn is the profile number you're after. | Calls:  Getdbase.m  Called by:  fetchloginfo.m  remake\_workfile.m  rerunsinglefloats.m |
| 57. | find\_best\_msg.m | FIND\_BEST\_MSG For each array of repeated lines in a cell-array, find lines with correct CRC and, if necessary, construct a line of most common values for each nibble (ie character). Used to reduce error-prone ARGOS- delivered Argo float transmissions to a single best profile message.  INPUT rawdat .dat [nlin maxlinelength] 31 or 32 byte decimal lines  .blkno [nlin 1] block numbers  dbdat float database struct including float maker and subtype codes  opts struct from decoded Argos message including field "nocrc"  OUTPUT best [maxblk linelength] single best profile message  fbm\_report 8 activity counts: (M\_C\_B = Most\_Common\_Byte)  1 - total number of all repetitions of all blocks  2 - total number with good CRC  3 - number of blocks, no ok CRC, used M\_C\_B  4 - number of blocks, no ok CRC, did NOT use M\_C\_B  5 - number of blocks with only one ok CRC  6 - number of blocks where all ok-CRC messages are identical  7 - number of blocks, differing ok-CRC messages - used M\_C\_B  8 - where no ok-CRC, used M\_C\_B, number of bytes rejected  because no repeated values.  rawdat Input structure, but with .crc & .maxblk added. | Calls:  CRCcheck.m  Called by:  load\_traj\_apex\_argos.m  load\_traj\_apex\_iridium.m  process\_profile.m  strip\_for\_workfile.m |
| 58. | findspike.m | FINDSPIKE Argo prescribed spike tests (tests 9 & 10)  Presently algorithm straight from 'addQC.m'  INPUT: vv - a profile of T or S  p - corresponding pressures  var - 't' or 's'  OUTPUT: b1 - index to spikes by test 9  b2 - index to spikes by test 10 (obsolete and useless) | Calls:  None  Called by:  qc\_tests2.m  qc\_tests\_Profile2.m |
| 59. | fixallfloatmatfiles.m | script to run over float.mat directory to ensure all fields are present: | Calls:  Getdbase.m  Fixfields.m  Called by:  None (stand alone) |
| 60. | fix\_all\_matfiles.m | FIX\_ALL\_MATFILES View or perform some action (you add your lines below) for all or a selection of matfiles. Should be rarely needed.  \*\*\* Use this rather than fiddling with matfiles in Matlab!  Note: good idea to keep an appropriately named and documented version of this - as a record of what was done!  INPUT: wmo - WMO ID [defaults to all listed in database]  arg1 - ?  OUTPUT Argo matfiles modified according to your code, added below | Calls:  Getdbase.m  Called by:  rework\_flag\_set.m |
| 61. | Fixfields.m |  | Calls:  new\_profile\_struct.m  called by:  fixallfloatmatfiles.m |
| 62. | fixmissionFloatTech.m | script to fix the empty missions for iridium floats | Calls:  cullMissions\_iridium.m  called by:  None (stand alone). |
| 63. | fixNSTdata.m | function fixNSTdata  this splits out the unpumped data from the pumped data and stroes it back to the float.mat file | Calls:  Getargo.m  Getdbase.m  Called by:  None (stand alone) |
| 64. | Fixoxy.m | fix oxygen concentration units: | Calls:  Getdbase.m  Getargo.m  convert\_uMolar.m  argoprofile\_nc.m  web\_profile\_plot.m  tsplots.m  locationplots.m  techinfo\_nc.m  webUpdatePages.m  web\_float\_summary.m  time\_section\_plot.m  waterfallplots.m  called by:  fixallfloatmatfiles.m |
| 65. | fixoxyQC.m | fix oxygen concentration units: | Calls:  Getdbase.m  Getargo.m  convertFsig.m  convertBbsig.m  qc\_tests.m  argoprofile\_nc.m  called by:  None (stand alone) |
| 66. | fixtechnicalprocesses.m | look for string matches- need to do this every profile | Calls:  None  Called by:  None |
| 67. | fix\_things\_cells.m | regenerate technical data files to fix inconsistencies.... | Calls:  Getdbase.m  Getargo.m  techinfo\_nc.m  web\_float\_summary.m  trajectory\_nc.m  called by:  None (stand alone) |
| 68. | generate\_all\_plots.m | run\_allArgoRTplots and re-generate web files: | Calls:  Getdbase.m  Getargo.m  Waterfallplots.m  time\_section\_plot.m  locationplots.m  tsplots.m  called by:  None (stand alone) |
| 69. | generic\_fields\_nc.m | GENERIC\_FIELDS\_NC Define or fill a few fields which are common to several Argo netCDF file types.  NOTE: Not used for Profile files (typ 1), because although all of these  fields are in those files, they are differently dimensioned there.  INPUT:  nc - the netcdf file handle  dbdat - master database record for this float  typ - Argo filetype: 1=Profile 2=Metadata 3=Trajectory 4=Technical  stage - 1=define 2=fill | Calls:  None  Called by:  metadata\_nc.m  techinfo\_nc.m  trajectory\_nc.m |
| 70. | genV31profs.m | Regenerate the old profile with Version 3.1 | Calls:  Getdbase.m  Getargo.m  argoprofile\_nc.m  called by:  None (stand alone) |
| 71. | getadditionalinfo.m | GETDBASE On first call for a session, all float details are loaded from  % the master database (massaging some fields in the process) into global  % variables. If 'fnum' provided, extracts record for that float. | Calls:  None  Called by:  Argoprofile\_nc.m  Argoprofile\_Bfile\_nc.m  Metadata\_nc.m  Trajectory\_nc.m  Web\_float\_summary.m |
| 72 | getargo.m | GETARGO Get a float array, or just one profile, from mat-files | Calls:  Getdbase.m  Called by:  addIridiumPositions.m  argoprofile\_nc.m  calc\_ice\_cv.m  combineDandRfileV3\_1final4.m  cullMissions\_iridium.m  drive\_argo\_manual.m  fetchloginfo.m  fixNSTdata.m  fixoxy.m  fixoxyQC6.m  fix\_things\_cells.m  generate\_all\_plots.m  interpolate\_locations.m  plot\_all\_profiles.m  regenerate\_profile\_files.m  regenerate\_techfiles.m  regenNSTProfs.m  rejectpoints.m  rerunoxygenprofiles.m  rerunsinglefloats.m  runall\_metadata.m  runall\_techfiles.m  runall\_trajfiles.m  runsingTraj.m  view\_profs.m  web\_processing\_report.m |
| 73. | get\_Argos\_config\_params.m | this script makes it easier to modify and add to various configuration  % parameters for the Argos versions of the floats. this is based on  % getdbase | Calls:  None  Called by:  metadata\_nc.m  trajectory\_nc.m |
| 74. | Getcaldbase.m | GETCALDBASE On first call for a session, all float CALIBRATION details are  % loaded from the master cal database into a global variable struct array. | Calls: none  Called by:  metadata\_nc.m |
| 75. | getconfig\_names.m | Blank |  |
| 76. | getCrc.m | Function used to find the CRC value for a given  % message using the 16 bit CRC-CCITT standard polynomial.  % The Modulo-2 Binary Division is implemented and the  % remainder of the division will be the CRC. This  % implementation is rather slow and can be speeded up  % using the well known technique of generating first a  % look-table. (Will do when I get the time!). | Calls: none  Called by:  CRCcheck.m |
| 77. | Getdbase.m | % GETDBASE On first call for a session, all float details are loaded from the master database (massaging some fields in the process) into global variables. If 'fnum' provided, extracts record for that float. | Calls:  Called by:  calc\_ice\_cv.m  combineDandRfiles.m  convert\_matfiles.m  cullAllMissions.m  drive\_argo\_manual.m  edit\_workfile.m  export\_argo.m  extract\_Iridium\_data.m  extract\_Solo2\_data.m  find\_argos\_download.m  fixallfloatmatfiles.m  fix\_all\_matfiles.m  fixNSTdata.m  fixoxy.m  fixoxyQC.m  fix\_things\_cells.m  generate\_all\_plots.m  genV31profs.m  getadditionalinfo.m  getargo.m  get\_Argo\_config\_params.m  getcaldbase.m  getO2caldbase.m  idcrossref.m  load\_float\_to\_traj.m  matfile\_edit.m  mychlconv.m  plot\_all\_profiles.m  proc\_recs\_rebuild.m  regenerate\_profile\_files.m  regenerate\_techfiles.m  rework\_flag\_set.m  runall\_metadata.m  runall\_techfiles.m  run\_from\_workfile.m  strip\_argos\_msg.m  strip\_for\_workfile.m  test\_new\_formats\_nospreadsheet.m  testpadding.m  testprovorinput.m  web\_database.m  web\_processing\_report.m  web\_select\_float.m |
| 78. | Getfloatspecs.m | %run with mypark\_depths2 | Calls: none  Called by: none |
| 79. | Getmission\_number.m | %function[mn,config]=getmission\_number(wmo\_id,pn,all,dbdat)  %  % function to retrieve the current mission number from the aux.mat files for iridium floats: | Calls:  Called by:  argoprofile\_Bfile\_nc.m  argoprofile\_nc.m  metadata\_nc.m  trajectory\_nc.m |
| 80. | GetO2caldbase.m | % GETO2CALDBASE On first call for a session, all float CALIBRATION details are loaded from the master cal database into a global variable struct array. | Calls: none  Called by:  convertBphase.m  convertFREQoxygen.m  metadata\_nc.m  mychlconv.m  strip\_argos\_msg.m |
| 81. | Get\_ocean\_depth.m | % GET\_OCEAN\_DEPTH Find deepest point within +/- .25 deg of each position | Calls: Called by:  Process\_iridium.m  Process\_profile.m  Qc\_tests.m  Strip\_for\_workfile.m  Test\_process.m |
| 82. | Idcrossref.m | % IDCROSSREF Return all IDs for the given id(s) of one type. | Calls:  Called by:  Extract\_Iridium.m  Extract\_phy\_data.m  Extract\_Solo2\_data.m  Getadditionalinfo.m  Getdbase.m  Strip\_argos\_msg.m  Strip\_for\_workfile.m  Test\_new\_formats\_nospreadsheet.m  Testprovorinput.m |
| 83. | interpolate\_locations.m | % function [latarr,lonarr]=interpolate\_locations(dbdat)  % this script takes a float structure and checks to see if locations from previous profiles are missing. if so, it interpolates from teh current location and previous locations and adds them to the mat structure. It then generates the tesac and netcdf files for the float for delivery to the GDACs and GTS. | Calls:  Called by:  addIridiumPositions.m  extract\_Iridium\_data.m  extract\_phy\_data.m  process\_iridium.m |
| 84. | iridiumprofileformat.m | Profile format | Calls: none  Called by: None |
| 85. | isapprox.m | % function isapprox - checks whether the input is within tolerance of a target: | Calls:  Called by:  Profile\_rollover.m |
| 86. | jamstec\_position\_test.m | % JAMSTEC\_POSITION\_TEST Carry out quality test of trajectory position fixes  % float.position\_accuracy contains the Argos position accuracy class, ie G 1 2 3 0 A B Z | Calls:  Called by:  load\_traj\_apex\_argos |
| 87. | load\_ctd\_refset.m | % LOAD\_CTD\_REFSET Load pre-arranged CTD reference set | Calls: none  Called by: None |
| 88. | load\_float\_to\_traj.m | % LOAD\_FLOAT\_TO\_TRAJ This is not part of normal processing, but a utility/test program to build the whole traj workfile and traj netcdf file for a float, when required by the operator.  % | Calls:  Called by:  Runall\_trajfiles.m  runsingleTraj.m |
| 89. | load\_traj\_apex\_argos.m | % Collect Trajectory data from one cycle for an Argos Apex float (APF8,9), and analyse accumulated time-series for float to deterine/update all values needed for trajectory files. | Calls:  Median.m  Polyfit.m  Julian.m  Hist.m  Find\_best\_msg.m  Calc\_TST.m  Trim\_LMT.m  Calc\_TET.m  Called by:  Process\_profile.m  Jamstec\_position\_test.m  Load\_traj\_to\_traj.m  Load\_traj\_apex\_argos.m |
| 90. | load\_traj\_apex\_iridium.m | % Collect Trajectory data for one cycle for an Apex float, and analyses accumulated time-series for float to deterine/update all values needed for trajectory files. | Calls:  Median.m  Polyfit.m  Julian.m  Hist.m  Find\_best\_msg.m  Calc\_TST.m  Trim\_LMT.m  Calc\_TET.m  Called by:  None |
| 91. | locationplots.m | % LOCATIONPLOTS Create location plot for one float. | Calls:  Called by:  addIridiumPositions.m  extract\_Solo2\_data.m  fixoxy.m  generate\_all\_plots.m  interpolate\_locations.m  process\_iridium.m  process\_profile.m |
| 92. | Logerr.m | % LOGERR Update global activity report structure and write msg to report file | Calls:  Called by:  Argoprofile\_Bfile\_nc.m  Argoprofile\_nc.m  Bin2num.m  Calc\_ascent\_end.m  Calibrate\_p.m  Calsal\_clim.m  Calsal.m  Celltm\_sbe41.m  Check\_profile.m  convertBbsig.m  convertBphase.m  convertFREQoxygen.m  convertFsig.m  convertSBE63Oxy.m  convertSBEOxyfreq.m  convertTphase.m  CRCcheck.m  Date\_ftp\_file.m  Decode\_provor.m  Decode\_webb.m  Edit\_workfile.m  Export\_argo.m  Export\_text\_file.m  Extract\_Iridium\_data.m  Extract\_phy\_data.m  Extract\_Solo2\_data.m  Find\_best\_msg.m  Getdbase.m  Metadata\_nc.m  Process\_iridium.m  Process\_profile.m  Profile\_rollover.m  Qc\_test.m  Qc\_test\_oxygen.m  Qc\_test\_profile2.m  Report\_bad\_prof.m  Run\_from\_workfile.m  Save\_workfiles.m  Strip\_argos\_msg.m  Strip\_for\_workfile.m  Techinfo\_nc.m  Trajectory\_nc.m  Web\_float\_summary.m  Write\_BUFR.m  Write\_tesac.m |
| 93. | mail\_out\_ArgoRT\_report.m | % mail out function for sending ArgoRT reports to operators | Calls: none  Called by:  Strip\_argos\_msg.m |
| 94. | mail\_out\_dead\_float.m | % mail out function for notifying 'dead float talking' | Calls: none  Called by:  Extract\_iridium\_data.m  Process\_iridium.m  Strip\_agos\_msg.m |
| 95. | mail\_out\_iridium\_log\_error.m | % mail out function for sending .log or .msg | Calls:  Called by:  Extract\_Iridium\_data.m  Extract\_phy\_data.m |
| 96. | mail\_out\_overdue\_floats.m | % mail\_out\_overdue\_floats  % mail out function for sending overdue float list to operators | Calls: none  Called by: none |
| 97. | mail\_out\_TESAC.m |  | Calls: none  Called by: export\_argo.m |
| 98. | matfile\_edit.m | % MATFILE\_EDIT Perform some limited actions on one Argo matfile. Should never be needed - use with great reservation (but do use this rather than directly fiddling with matfiles in Matlab! | Calls: none  Called by: none |
| 99. | Metadata\_nc.m | % METADATA\_NC Create a new Argo metadata file  %  % INPUT: dbdat - master database record for this float  % fpp - struct array containing the profiles for this float  % (presently get a couple of times from this) | Calls:  Generic\_fields\_nc.m  netCDF toolbox functions  julian.m  Called by:  Extract\_Solo2\_data.m  genV31metatech.m  getcaldbase.m  getO2caldbase.m  plot\_all\_profiles.m  process\_iridium.m  process\_profile.m  runall\_metadata.m  trajectory\_nc.m |
| 100. | my\_save\_fig.m | %save\_fig: Save an image of the current Figure window in a file. | Calls:  Called by:  Locationplots.m  Time\_section\_plot.m  Tsplots.m  Waterfallplots.m  Web\_plot\_generation.m  Web\_profile\_plot.m |
| 101. | new\_profile\_struct.m | % NEW\_PROFILE\_STRUCT Create a blank standard raw Argo profile structure | Calls:  Called by:  Convert\_matfiles.m  Decode\_provor.m  Decode\_webb.m  Extract\_Solo2\_data.m  Fix\_all\_matfiles.m  Fixfields.m  Matfile\_edit.m  Metadata\_nc.m  Process\_iridium.m  Process\_profile.m |
| 102. | O2sol.m | O2sol Solubility of O2 in sea water | Calls:  Called by:  Bphase\_conv.m  convertBhpase.m  convertFREQoxygen.m  convertSBEOxyfreq.m  metadata\_nc.m |
| 103. | Overall\_qcflag.m | %OVERALL\_QCFLAG Compute the Argo QC status flag given the QC flags for a single profile. We assume that profile has been QC-ed. | Calls:  Called by:  Argoprofile\_nc.m  Argoprofile\_Bfile\_nc.m  combineDandRfileV3\_1final4.m |
| 104. | Plot\_all\_profiles.m | % plot\_all\_profiles - this cycles through all the floats and regenerates the web plots for all profiles. Edit this to plot only selected profiles or selected floats.... | Calls:  Called by:  None |
| 105. | plotbathyArgo.m | % plotbathy adds the bathymetry to the map plot window... | Calls: none  Called by: none |
| 106. | Plot\_depths\_iota.m | % run this when you are running quest to check bottom depths from the databases. | Calls: none  Called by: none |
| 107. | Pos\_fix\_check.m | % POS\_FIX\_CHECK Investigate apparent high speeds between fixes for the surface drift period in one profile. | Calls:  Called by: edit\_workfile.m |
| 108. |  |  |  |

BOM\_retrieve\_Iridium.m - blank.

BOM\_send\_iridium\_notify.m - blank

BOM\_write\_BUFR.m - blank

Codes for processing SOLO floats which transmit data in SBD format. Slightly modified version of these can used for processing Arvor-I floats data.

|  |  |  |  |
| --- | --- | --- | --- |
| **SNo** | **Name of the Function/Subroutine** | **Purpose of the Function/Subroutine in the Argo RT processing** | **Calls which other routine(s)** |
| 1. | InputArgoIMEI.m | [IMEI\_dbdat,HullID] = InputArgoIMEI  inputArgoIMEI - gets the IMEIs for the argo floats that report by SBD so we can process them when we process the XBT SBDs this works with extract\_SBD\_data to determine which of the SBDs are Argo and which are XBT data. | Calls: none  Called by: extract\_Solo2\_data.m |
| 2. | decodeSolo2Config.m | Mission = decodeSolo2Config(sensor)  SBD - SOLO2 DECODE THE ASCII DUMP OF MISSION CONFIG PARAMETERS | Calls: none  Called by: decodeSolo2Config.m  extract\_Solo2\_data.m |
| 3. | decodeSolo2PumpSeries.m | RiseRate = decodesolo2PumpSeries(sensor)  SBD - SOLO2 DECODE THE PUMP SERIES DATA MESSAGE | Calls: none  Called by:  extract\_Solo2\_data.m |
| 4. | solo2\_GetHeader.m | [X, nn, Hull, dive, id, valid] = solo2\_GetHeader(sbdfile)  READ THE FIRST 6 BYTES OF THE sbd FILE RETURNING:  % -----------------------------------------------------------------------------  % X = b(1) -is the character "X"  % nn = 256\*b(2)+b(3) -is the length of the message - 7 in bytes  % Hull = 256\*b(4)+b(5) -is the serial number of the solo2 float ex: 7070  % dd = 256\*b(6)+b(7) -is the dive number -1:startup, 0:test dive, >1 profile  % id = b(8) -is the packet id counter 0-255  % valid = true/false; -verifies the integrity of the packet | Calls: none  Called by:  extract\_Solo2\_data.m  solo2\_GetHeader.m |
| 5. | decodeSolo2EngE3.m | Eng = decodeSolo2EngE3(sensor)  SBD - SOLO2 DECODE THE ENGINEERING 0xE0 DIAGNOSTIC MESSAGE | Calls: none  Called by:  decodeSolo2EngE3.m  extract\_Solo2\_data.m |
| 6. | decodeSolo2RiseRate.m | RiseRate = decodeSolo2RiseRate(sensor)  SBD - SOLO2 DECODE THE ENGINEERING 0xE0 DIAGNOSTIC MESSAGE | Calls: none  Called by:  extract\_Solo2\_data.m |
| 7. | extract\_Solo2\_data.m | % extract Iridium data - checks the delivery directories and, if there is new data, processes it before sending strip\_argos\_msg | Calls:  getdbase()  inputArgoIMEI  solo2\_GetHeader()  decodeSolo2GPS()  decodeSolo2PumpSeries()  decodeSolo2Config()  decodeSolo2EngE0()  decodeSolo2EngE2()  decodeSolo2EngE3()  decodeSolo2EngE5()  decodeSolo2Mission()  calibrate\_p()  qc\_tests()  web\_profile\_plot()  argoprofile\_nc()  techinfo\_nc()  time\_section\_plot()  waterfallplots()  locationplots()  tsplots()  web\_float\_summary()  Called by:  extract\_Iridium\_data.m  strip\_argos\_msg.m |
| 8. | decodeSolo2EngE2.m | Eng = decodeSolo2EngE2(sensor)  SBD - SOLO2 DECODE THE ENGINEERING 0xE2 DIAGNOSTIC MESSAGE PORTION | Calls: none  Called by:  extract\_Solo2\_data.m |
| 9. | decodeSolo2Mission.m | Mission = decodeSolo2Mission(sensor)  SBD - SOLO2 DECODE THE ARGO DATA MESSAGE  % -----------------------------------------------------------------------------  % ID: 'f0'=240 After each dive, specific dive information for creating the PHY file for the ARGO community is sent. Dive information for creating the PHY file. Mission configuration information. | Calls: none  Called by:  decodeSolo2EngE2.m  extract\_Solo2\_data.m |
| 10. | decodeSoloP.m | [pres] = decodeSoloP(sbdm)  take a raw sbd messge and extract pressure, returning it in a vector to be sorted later %  % usage: decodeSoloP  % where sbdm is a binary attachment from a single file  % and pres is a set of pressure values from that file | Calls: none  Called by:  decodeSoloS.m  decodeSoloT.m  extract\_Solo2\_data.m |
| 11. | decodeSoloS.m | [psal] = decodeSoloS(sbdm)  take a raw sbd messge and extract salinity, returning it in a vector to be sorted later %  % usage: decodeSoloS  % where sbdm is a binary attachment from a single file  % and sal is a set of salinity values from that file | Calls: none  Called by:  extract\_Solo2\_data.m |
| 12. | decodeSoloT.m | [temp] = decodeSoloT(sbdm)  take a raw sbd messge and extract temperature, returning it in a vector to be sorted later %  % usage: decodeSoloT  % where sbdm is a binary attachment from a single file  % and temp is a set of temperature values from that file | Calls: none  Called by:  extract\_Solo2\_data.m |
| 13. | decodeSolo2EngE5.m | Eng = decodeSolo2EngE5(sensor)  SBD - SOLO2 DECODE THE ENGINEERING 0xE0 DIAGNOSTIC MESSAGE  % -----------------------------------------------------------------------------  % Engineering 0xe5 (229)  % ID=0xe5, Engineering message following BITest | Calls: none  Called by:  extract\_Solo2\_data.m |
| 14. | decodeSolo2GPS.m | U = decodesolo2GPS(sensor)  %SBD - SOLO2 DECODE THE X-01 GPS MESSAGE:  % --------------------------------------------------------------------------------  % SENSOR ID Code: 0x01-0x05 (ALL GPS DATA FORMATS)  % GPS 0x01 Fix at before leaving surface in Surface Drift | Calls: none  Called by:  extract\_Solo2\_data.m |
| 15. | fill\_float\_from\_solo2.m | Fp = fill\_float\_from\_solo2(fp,flstruct,argosid)  this function takes the structure decoded from the binary SBD messages and combines it with the metadata and profile data for that profile and fills in the float.mat structure required by the plotting and netcdf scripts of the Argo RT processing. | Calls: none  Called by:  extract\_Solo2\_data.m |
| 16. | decodeSolo2EngE0.m | Eng = decodeSolo2EngE0(sensor)  SBD - SOLO2 DECODE THE ENGINEERING 0xE0 DIAGNOSTIC MESSAGE  % -----------------------------------------------------------------------------  % Engineering 0xe0 (224)  % Diagnostic data in first diagnostic dive at start of mission | Calls: none  Called by:  extract\_Solo2\_data.m |