DESCRIPTION OF THE VALIDATION FLAGS

A detailed description of the validation procedure applied to the original data is given in the "Hydrographic Atlas of the Southern Ocean" (Olbers, Gouretski, Seiss, Schröter, 1992)

The main idea of the validation procedure applied to the data was to put them through a number of filters. The data which fail to pass a filter (check) are marked with the respective value of the validation flag.

There are two validation flags for every hydrographic station:

Station_Validation_Flag is given in the station header and is applied for the whole station. It characterizes the quality of the information within the station header or/and the quality of all temperature or/and salinity or/and oxygen data at the station.

Standard_Data_Validation_Flag is given for every standard level. It characterizes the quality of temperature or/and salinity or/and oxygen data at every standard level.

In the data base the validation flags are represented by an INTEGER*2 variable (32 bits). It allows to keep information about up to 32 checks simultaneously. INTEGER data types are represented internally in binary twos complement notation and a bit in a binary pattern (Validation_Flag) has a value of 0 (data passed the check) or 1 (data failed the check).

A description of the validation flags and examples of VAX FORTRAN programs for manipulations with validation flags are given below.

<table>
<thead>
<tr>
<th>N</th>
<th>VAL Flag</th>
<th>Meaning of the value</th>
<th>Abbreviation used by PANGAEA</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>No Validation applied</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Cruise_Number changed</td>
<td>Cruise number changed</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>Date_Time corrected</td>
<td>Date/Time corrected</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>Date_Time doubtful</td>
<td>Date/Time doubtful</td>
</tr>
<tr>
<td>4</td>
<td>8</td>
<td>Longitude corrected</td>
<td>Longitude corrected</td>
</tr>
<tr>
<td>5</td>
<td>16</td>
<td>Latitude corrected</td>
<td>Latitude corrected</td>
</tr>
<tr>
<td>6</td>
<td>32</td>
<td>Longitude and Latitude corrected</td>
<td>Longitude and Latitude corrected</td>
</tr>
<tr>
<td>7</td>
<td>64</td>
<td>Coordinates doubtful</td>
<td>Coordinates doubtful</td>
</tr>
<tr>
<td>8</td>
<td>128</td>
<td>Out of temperature range</td>
<td>Out of T range</td>
</tr>
<tr>
<td>9</td>
<td>256</td>
<td>Out of salinity range</td>
<td>Out of S range</td>
</tr>
<tr>
<td>10</td>
<td>512</td>
<td>Out of oxygen range</td>
<td>Out of O2 range</td>
</tr>
<tr>
<td>11</td>
<td>1024</td>
<td>Statistical check for Temperature</td>
<td>Statchk for T</td>
</tr>
<tr>
<td>12</td>
<td>2048</td>
<td>Statistical check for Salinity</td>
<td>Statchk for S</td>
</tr>
<tr>
<td>13</td>
<td>4096</td>
<td>Statistical check for Oxygen</td>
<td>Statchk for O2</td>
</tr>
<tr>
<td>14</td>
<td>8192</td>
<td>Special case of salinity correction through the comparison with high-quality data</td>
<td>Special case of S correction</td>
</tr>
<tr>
<td>15</td>
<td>16384</td>
<td>Unrealistic salinity profile</td>
<td>Unrealistic S profile</td>
</tr>
<tr>
<td>16</td>
<td>32768</td>
<td>Unrealistic temperature profile</td>
<td>Unrealistic T profile</td>
</tr>
<tr>
<td>17</td>
<td>65536</td>
<td>Misleading dummy value for temperature (t=0)</td>
<td>t=0 removed</td>
</tr>
</tbody>
</table>
Two examples of the usage of the validation flags are given below.

DESCRIPTION OF THE VARIABLES in the FORTRAN program:

C VFSTATION - Value of the Validation_Flag in the Station header
C Array VFSTNDATA of the Validation_Flags for each standard level
C Array of temperature TEM(k) for the station under consideration
C Array of salinity SAL(k) for the station under consideration
C Array of standard level depths Z (maximum 42 levels)
C Integer array NV. Sequential numbers of the elements of array NV
C are equal to the sequential numbers of Validation_Flags as given in
C the Description of Validation Flags
C
C..................
    integer*2 NV(32)
    integer*4 vfstation,vfstndata(42)
    real*4 tem(42), sal(42), z(42)
C..................

EXAMPLE 1.

C SELECTION OF TEMPERATURE AND SALINITY DATA WHICH PASSED QUALITY CHECKS
C BOTH FOR TEMPERATURE AND SALINITY
C
    call bit(vfstation,NV)
    ns=nv(7)+nv(25)+nv(28)+nv(29)+nv(30)+nv(31)  ! reject station
    if(ns.gt.0) go to 1

C
    kk=0
    do 2 k=1,NLEVEL
    call bit(vfstndata, NV)
    ns=nv(8)+nv(9)+nv(11)+nv(12)+nv(15)+nv(16)+nv(17)+nv(18)+nv(19)
* +nv(20)+nv(23)

C
EXAMPLE 2.

C SELECTION OF OXYGEN DATA WHICH PASSED QUALITY CHECKS FOR OXYGEN

call bit(vfstation, NV)
nS=nv(7)+nv(22)+nv(24)+nv(26)
if(ns.gt.0) go to 1 !! reject station
C
  kk=0
  do 2 k=1,NLEVEL
  call bit(vfstndata, NV)
nS=nv(10)+nv(13)+nv(21)
C
  if(ns.ne.0)go to 2 ! reject standard level
  kk=kk+1
  OX(KK)=OX(K)
z(kk)=z(k)
  2 continue
  1 continue
C..................................

The following subroutine BIT uses VAX intrinsic function BTEST
(see "Programming in VAX FORTRAN", Digital Equipment GmbH, 1987)

  subroutine BIT(M,IA)
C
  C   M - value of the Validation_Flag to check
  C   IA - integer array, where IA(i)=1 when i-th bit of M equal 1 (the value
  C         has not passed i-th check)
  C   or IA(i)=0 when i-th bit of M equal 0 (the value has passed i-th check)
  C   Index "I" for the element of array "IA" is the number of the
  C   corresponding validation procedure.
  C (For meaning of Flags see Description of Validation Flags)
C
  C   IA(i)=1 means that after the validation procedure the previous value
  C   of the Validation_Flag was replaced by the new value according to:
  C   New_Valid_Flag=Old_Valid_Flag + 2**I
C
  integer*2 ia(32)
  integer*4 N,M,IA1
  logical*2 T
  IA1=M
  do 2 ib=1,32
ia(ib)=0
ia2=ib-1
ia1=M
T=BTEST(ia1,ia2)
if(T.eq..TRUE.,)ia(ib)=1
2 continue
   return
end