

Supplementary Information B

A guide for the PC interface to extract the tie points

1. General information

A PC interface used to search for tie points is explained. Based on preliminary tie points, a detailed search can be conducted easily. Figure B1 is the window of the interface. Procedures are as follows.

- (i) Preparation of the data files. Each set of ice core data (ECM, DEP, ACECM, sulfate or other data) should have three columns; column of its original depth, data values and tentative depth equivalent to a single reference core (DF1 core in case of this study). The data of tentative depth equivalent to a single reference core must be made prior to the use of the PC interface.
- (ii) We must prepare the computer program for your data. Necessary procedure include setting of the input/output directory, name of variables for your data, number of data. In this document, source code is provided (see below). Lines indicated with light yellow are lines that you need to modify. Once preparation of data and the source code is done, you can execute the program. The program requires environment of Microsoft Visual Basic 6.0 which runs on Windows PC.
- (iii) Loading of the data. The entire data should be loaded in the program.
- (iv) Display graphs on the PC interface. We should display depth-dependent profile of each set of data on the PC windows. Like the example of Figure B1, multiple windows should be aligned vertically, so that we can compare features of each data easily. Importantly, for x axis, the tentative depth equivalent to a single reference core must be used in order that user can easily examine synchronicity between multiple sets of data. In the windows, data should be scalable both in depth (x) directions and data value (y) directions. In addition, the x -axis should be adjustable for offset of the depth scales for each core data.
- (v) Extraction of local maxima from each set of data. In the data profiles, the candidates for tie points should be found by extracting local maxima. Importantly, the operator should be careful to keep synchronization among graphs by adjusting the offset. To observe pattern of the appearance of peaks is very important.
- (vi) The operator should choose each datum or not (1/0 switches in the right side of the image, in case of this study). By clicking “Record” on the right, the data – depth of peak, peak height and background level – should be recorded only for the chosen data.
- (vii) By shifting the depth range of windows, the operator should scan further candidate of the tie points.

2. View of the PC interface



Figure B1: PC interface when program is started.

Explanation of buttons, text boxes and their functions

Left top: “Start” is to start the program. “Cancel” and “End” are to cancel and to stop, respectively.

Left top: Text box at “Initial z” is to set depth in meters at which processing start.

Left top: “Span” means depth span (width of each graph) for analysis.

Left top: “Maximum detection range” sets width at the center of graphs to search for the local maxima. With smaller numbers, range for search is narrower.

Left: Text boxes at “x offset” set offset of the graphs in x direction.

Left: Two text boxes at each graph are for setting the range of the y-values.

Center: 8 windows are for displaying the graphs.

Center: Buttons with numbers “+1, +10, -1, -10” are to find peaks manually. Numbers mean shift numbers of data point.

Center: Switches “Auto” sets automatic detection of the candidate peak or not.

Center: Switches “1” and “0” is to choose the peak for recording (1) or not (0).

Right: Buttons with an indication of “Shift window” are to shift the depth range for searching.

Right: A button “Record” is for recording information of peaks.

Note: When numbers in the text boxes are modified, click the button “RE” on the right. Then changes of the settings are reflected and updated on the graph.

Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 kyr

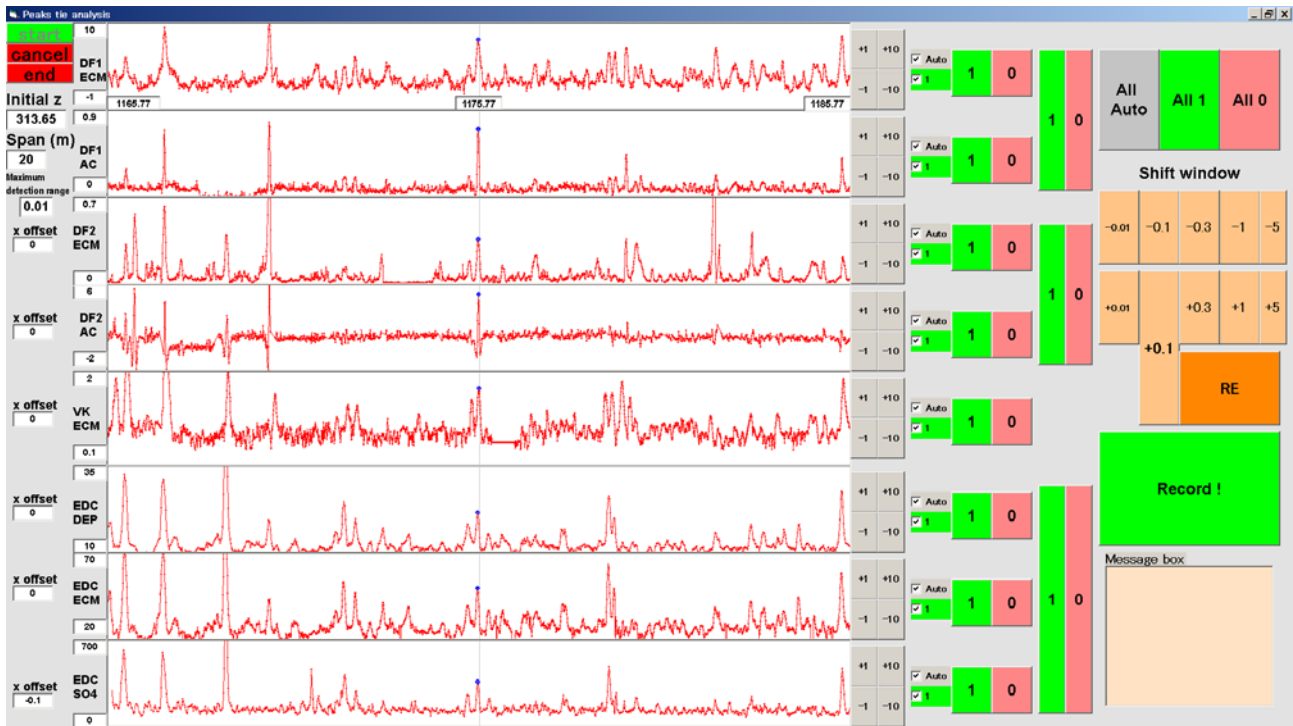


Figure B2: PC interface data are loaded. At the center of the graphs, local maxima are chosen automatically. If the operator click a button “Record”, information of the peaks are recooded in PC.

3. Source code

Name of the vbp (Visual Basic Project) file: synchronization.vbp

Name of the frm (Visual Basic Form) file: synchronization.frm

```
'-----
' A program for PC interface to extract tie points from ice core data
' written by Shuji Fujita, National Institute of Polar Research, Tokyo, Japan
' Date: May 13th, 2015
'-----

'Declarations (Variables that are commonly used in multiple routines)

'Dimensions for ice core data
'Note that we use depth of DF1 core as reference depth.
'For cores except DF1 core, we prepare data of (i) original depth, (ii) ice core
data and (iii) reference depth equivalent to DF1 core.
'This reference depth is used to make multiple graphs and compare between them.

Dim df_depth(242620), df_dc(242620), df_ac(242620) 'DF1 core ECM and ACECM data
Dim edc_acdepth(122689), edc_ac(122689), edc_acdepth_df1(122689) 'EDC99 core DEP
data
Dim df2_acdepth(159598), df2_ac(159598), df2_acdepth_df1(159598) 'DF2 core AC-
ECM data
Dim df2_dcdepth(167888), df2_dc(167888), df2_dcdepth_df1(167888) 'DF2 core ECM
data
Dim edc_dc99depth(162824), edc_dc99(162824), edc_dc99depth_df1(162824) 'EDC99
core EDC data
Dim edc_dc96depth(68859), edc_dc96(68859), edc_dc96depth_df1(68859) 'EDC96 core
ECM data
Dim edc_sl99depth(65526), edc_sl99(65526), edc_sl99depth_df1(65526) 'EDC99 core
Sulfate data
Dim edc_sl96depth(19133), edc_sl96(19133), edc_sl96depth_df1(19133) 'EDC96 core
Sulfate data
Dim vk_dcdepth(163341), vk_dc(163341), vk_dcdepth_df1(163341) 'Vostok core
ECM data

'These variables are for to define coordinates of 8 graphs.
Dim xend_1, xstart_1, yend_1, ystart_1
Dim xend_2, xstart_2, yend_2, ystart_2
Dim xend_3, xstart_3, yend_3, ystart_3
Dim xend_4, xstart_4, yend_4, ystart_4
Dim xend_5, xstart_5, yend_5, ystart_5
Dim xend_6, xstart_6, yend_6, ystart_6
Dim xend_7, xstart_7, yend_7, ystart_7
Dim xend_8, xstart_8, yend_8, ystart_8

'These are prepared for offset of graphs in x axis. For DF1 core data (ID 1 and
2), offset was not necessary.
Dim xoffset3, xoffset4, xoffset5, xoffset6, xoffset7, xoffset8

Dim centerpixel 'Number of data around which search of the tie points was done.
Dim search_range 'Range of the tie point search
Dim row, column 'row is arbitrary. column is (2500)+(20 as header)

Dim maxx_1, maxy_1, maxb_1, prev_maxx_1, prev_maxy_1, maxj_1
Dim maxx_2, maxy_2, maxb_2, prev_maxx_2, prev_maxy_2, maxj_2
```

```

Dim maxx_3, maxy_3, maxb_3, prev_maxx_3, prev_maxy_3, maxj_3
Dim maxx_4, maxy_4, maxb_4, prev_maxx_4, prev_maxy_4, maxj_4
Dim maxx_5, maxy_5, maxb_5, prev_maxx_5, prev_maxy_5, maxj_5
Dim maxx_6, maxy_6, maxb_6, prev_maxx_6, prev_maxy_6, maxj_6
Dim maxx_7, maxy_7, maxb_7, prev_maxx_7, prev_maxy_7, maxj_7
Dim maxx_8, maxy_8, maxb_8, prev_maxx_8, prev_maxy_8, maxj_8
Dim maxx2_1, maxx2_2, maxx2_3, maxx2_4, maxx2_5, maxx2_6, maxx2_7, maxx2_8

Dim background

Dim bottomlevel      'A threshold to distinguish between noise and meaningful data.
Dim kagen_sikii
Dim sdev_bottomlevel  'standard deviation of the bottom level
Dim prev_value        'previous value as 1 or 0
Dim sa

Dim Dire$, Infile$(9), Outfile$      'directory, input file (multiple files),
output file

Dim PLN

Dim Graphx_1, pGraphx_1
Dim Graphyl_1, pGraphyl_1  'We use these in graph subroutine. X axis and two
kinds of Y values

Dim Graphx_2, pGraphx_2
Dim Graphyl_2, pGraphyl_2  'Ditto

Dim Graphx_3, pGraphx_3
Dim Graphyl_3, pGraphyl_3  'Ditto

Dim Graphx_4, pGraphx_4
Dim Graphyl_4, pGraphyl_4  'Ditto

Dim Graphx_5, pGraphx_5
Dim Graphyl_5, pGraphyl_5  'Ditto

Dim Graphx_6, pGraphx_6
Dim Graphyl_6, pGraphyl_6  'Ditto

Dim Graphx_7, pGraphx_7
Dim Graphyl_7, pGraphyl_7  'Ditto

Dim Graphx_8, pGraphx_8
Dim Graphyl_8, pGraphyl_8  'Ditto

'Various flags
Dim Dummy
Dim CancelFlag, Cancelled 'If user choose cancel, this flag is set.
Dim YesFlag
Dim NoFlag

Dim P10Flag_1, P10Flag_2, P10Flag_3, P10Flag_4, P10Flag_5, P10Flag_6, P10Flag_7,
P10Flag_8
Dim P1Flag_1, P1Flag_2, P1Flag_3, P1Flag_4, P1Flag_5, P1Flag_6, P1Flag_7,
P1Flag_8
Dim M10Flag_1, M10Flag_2, M10Flag_3, M10Flag_4, M10Flag_5, M10Flag_6, M10Flag_7,
M10Flag_8
Dim M1Flag_1, M1Flag_2, M1Flag_3, M1Flag_4, M1Flag_5, M1Flag_6, M1Flag_7,
M1Flag_8

Dim RedoFlag, Redo2Flag
Dim NFB 'Number of times to run Fwd or Bwd. Default is 1.

```

Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 kyr

```
Dim GoFlag
```

```
Dim N, NN, NNN, NID
```

```
Dim NNSTART, NNEND
```

```
Dim switch1
```

```
Private Sub Command10_Click()  
    P10Flag_2 = True  
    Check2.Value = 0  
End Sub
```

```
Private Sub Command11_Click()  
    M10Flag_2 = True  
    Check2.Value = 0  
End Sub
```

```
Private Sub Command12_Click()  
    P1Flag_3 = True  
    Check3.Value = 0  
End Sub
```

```
Private Sub Command13_Click()  
    M1Flag_3 = True  
    Check3.Value = 0  
End Sub
```

```
Private Sub Command14_Click()  
    P10Flag_3 = True  
    Check3.Value = 0  
End Sub
```

```
Private Sub Command15_Click()  
    M10Flag_3 = True  
    Check3.Value = 0  
  
End Sub
```

```
Private Sub Command16_Click()  
    P1Flag_4 = True  
    Check4.Value = 0  
  
End Sub
```

```
Private Sub Command17_Click()  
    M1Flag_4 = True  
    Check4.Value = 0  
End Sub
```

```
Private Sub Command18_Click()  
    P10Flag_4 = True  
    Check4.Value = 0  
  
End Sub
```

```
Private Sub Command19_Click()  
    M10Flag_4 = True  
    Check4.Value = 0  
  
End Sub
```

Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 kyr

```
Private Sub Command20_Click()  
    P1Flag_5 = True  
    Check5.Value = 0  
End Sub  
  
Private Sub Command21_Click()  
    P10Flag_5 = True  
    Check5.Value = 0  
End Sub  
  
Private Sub Command22_Click()  
    M1Flag_5 = True  
    Check5.Value = 0  
End Sub  
  
Private Sub Command23_Click()  
    M10Flag_5 = True  
    Check5.Value = 0  
  
End Sub  
  
Private Sub Command24_Click()  
    P1Flag_6 = True  
    Check6.Value = 0  
End Sub  
  
Private Sub Command25_Click()  
    P10Flag_6 = True  
    Check6.Value = 0  
  
End Sub  
  
Private Sub Command26_Click()  
    M1Flag_6 = True  
    Check6.Value = 0  
  
End Sub  
  
Private Sub Command27_Click()  
    M10Flag_6 = True  
    Check6.Value = 0  
  
End Sub  
  
Private Sub Command28_Click()  
    P1Flag_7 = True  
    Check7.Value = 0  
End Sub  
  
Private Sub Command29_Click()  
    P10Flag_7 = True  
    Check7.Value = 0  
  
End Sub  
  
Private Sub Command30_Click()  
    M1Flag_7 = True  
    Check7.Value = 0  
  
End Sub  
  
Private Sub Command31_Click()  
    M10Flag_7 = True  
    Check7.Value = 0  
End Sub
```

Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 kyr

```
Private Sub Command32_Click()  
Check9.Value = 1  
RedoFlag = True  
End Sub
```

```
Private Sub Command33_Click()  
Check10.Value = 1  
RedoFlag = True  
End Sub
```

```
Private Sub Command34_Click()  
Check11.Value = 1  
RedoFlag = True  
End Sub
```

```
Private Sub Command35_Click()  
Check12.Value = 1  
RedoFlag = True  
End Sub
```

```
Private Sub Command36_Click()  
Check13.Value = 1  
RedoFlag = True  
End Sub
```

```
Private Sub Command37_Click()  
Check14.Value = 1  
RedoFlag = True  
End Sub
```

```
Private Sub Command38_Click()  
Check8.Value = 0  
Check9.Value = 0  
Check10.Value = 0  
Check11.Value = 0  
Check12.Value = 0  
Check13.Value = 0  
Check14.Value = 0  
Check16.Value = 0  
RedoFlag = True  
End Sub
```

```
Private Sub Command39_Click()  
Check8.Value = 1  
Check9.Value = 1  
Check10.Value = 1  
Check11.Value = 1  
Check12.Value = 1  
Check13.Value = 1  
Check14.Value = 1  
Check16.Value = 1  
RedoFlag = True  
End Sub
```

```
Private Sub Command4_Click()  
Picture3.Cls  
RedoFlag = True  
End Sub
```

```
Private Sub Command40_Click()  
Check8.Value = 0  
RedoFlag = True  
End Sub
```


Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 kyr

```
Private Sub Command41_Click()  
    Check9.Value = 0  
    RedoFlag = True  
End Sub  
  
Private Sub Command42_Click()  
    Check10.Value = 0  
    RedoFlag = True  
End Sub  
  
Private Sub Command43_Click()  
    Check11.Value = 0  
    RedoFlag = True  
End Sub  
  
Private Sub Command44_Click()  
    Check12.Value = 0  
    RedoFlag = True  
End Sub  
  
Private Sub Command45_Click()  
    Check13.Value = 0  
    RedoFlag = True  
End Sub  
  
Private Sub Command46_Click()  
    Check14.Value = 0  
    RedoFlag = True  
End Sub  
  
Private Sub Command47_Click()  
Picture3.Cls  
    RedoFlag = True  
    Text8.Text = Val(Text8.Text) + 0.01  
  
End Sub  
  
Private Sub Command48_Click()  
Picture3.Cls  
RedoFlag = True  
    Text8.Text = Val(Text8.Text) - 0.01  
  
End Sub  
  
Private Sub Command49_Click()  
Check1.Value = 1  
Check2.Value = 1  
Check3.Value = 1  
Check4.Value = 1  
Check5.Value = 1  
Check6.Value = 1  
Check7.Value = 1  
Check15.Value = 1  
RedoFlag = True  
End Sub  
  
Private Sub Command5_Click()  
    M1Flag_1 = True  
    Check1.Value = 0  
End Sub  
  
Private Sub Command3_Click()  
    P1Flag_1 = True  
    Check1.Value = 0  
End Sub
```

Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 kyr

```
Private Sub Command50_Click()  
Check8.Value = 1  
Check9.Value = 1  
RedoFlag = True  
End Sub  
  
Private Sub Command51_Click()  
Check8.Value = 0  
Check9.Value = 0  
RedoFlag = True  
End Sub  
  
Private Sub Command52_Click()  
Check11.Value = 1  
Check12.Value = 1  
RedoFlag = True  
End Sub  
  
Private Sub Command53_Click()  
Check11.Value = 0  
Check12.Value = 0  
RedoFlag = True  
End Sub  
  
Private Sub Command54_Click()  
Check10.Value = 1  
Check13.Value = 1  
Check14.Value = 1  
RedoFlag = True  
End Sub  
  
Private Sub Command55_Click()  
Check10.Value = 0  
Check13.Value = 0  
Check14.Value = 0  
RedoFlag = True  
End Sub  
  
Private Sub Command56_Click()  
RedoFlag = True  
Text8.Text = Val(Text8.Text) + 0.3  
End Sub  
  
Private Sub Command57_Click()  
RedoFlag = True  
Text8.Text = Val(Text8.Text) - 0.3  
End Sub  
  
Private Sub Command58_Click()  
RedoFlag = True  
Text8.Text = Val(Text8.Text) + 5  
End Sub  
  
Private Sub Command59_Click()  
RedoFlag = True  
Text8.Text = Val(Text8.Text) - 5  
End Sub  
  
Private Sub Command6_Click()  
PiFlag_2 = True  
Check2.Value = 0  
End Sub  
  
Private Sub Command60_Click()
```

Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 kyr

```
        P1Flag_8 = True
        Check15.Value = 0
End Sub

Private Sub Command61_Click()
    M1Flag_8 = True
    Check15.Value = 0
End Sub

Private Sub Command62_Click()
    P10Flag_8 = True
    Check15.Value = 0
End Sub

Private Sub Command63_Click()
    M10Flag_8 = True
    Check15.Value = 0
End Sub

Private Sub Command64_Click()
    Check16.Value = 1
    RedoFlag = True
End Sub

Private Sub Command65_Click()
    Check16.Value = 0
    RedoFlag = True
End Sub

Private Sub Command7_Click()
    M1Flag_2 = True
    Check2.Value = 0
End Sub

Private Sub Command8_Click()
    P10Flag_1 = True
    Check1.Value = 0

End Sub

Private Sub Command9_Click()
    M10Flag_1 = True
    Check1.Value = 0

End Sub

Private Sub Form_Load()
    'This subroutine is executed initially.
    'All the data are loaded at this stage.

    StartButton.Enabled = True
    CancelButton.Enabled = False
    EndButton.Enabled = True

    'User of this program should modify this part.

    Dire$ = "C:\Users\Shuji Fujita\Desktop\DF_DC_ECmsynchronization\" 'Directory of
data storage

    'Data sets to be loaded
    Infile$(1) = "DF1_depth_dc_ac242620.csv" 'Number of data is 242620
    Infile$(2) = "EDC_depth_dep_dfdepth122689.csv" 'Number of data is 122689
    Infile$(3) = "DF2_depth_ac_depthdf159598.csv" 'Number of data is 159598
    Infile$(4) = "DF2_depth_dc_depthdf167888.csv" 'Number of data is 167888
    Infile$(5) = "EDC_depth_dc99_dfdepth162824.csv" 'Number of data is 167888
```

Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 kyr

```
Infile$(6) = "EDC_depth_dc96_dfdepth68859.csv" 'Number of data is 68859
Infile$(7) = "EDC_depth_sul_dfdepth65526.csv" 'Number of data is 65526
Infile$(8) = "EDC_depth_sl96_dfdepth19133.csv" 'Number of data is 65526
Infile$(9) = "VKdepth_ecm_dfdepth163341.csv" 'Number of data is 163341

'Name of output file
Outfile$ = "tie_points_20150510.txt"
```

End Sub

Private Sub StartButton_Click()

'This subroutine is executed when StartButton is clicked.

```
StartButton.Enabled = False
CancelButton.Enabled = True
EndButton.Enabled = True
```

'Load necessary files

```
Open Dire$ + Infile$(1) For Input As #1
Open Dire$ + Infile$(2) For Input As #2
Open Dire$ + Infile$(3) For Input As #3
Open Dire$ + Infile$(4) For Input As #4
Open Dire$ + Infile$(5) For Input As #5
Open Dire$ + Infile$(6) For Input As #6
Open Dire$ + Infile$(7) For Input As #7
Open Dire$ + Infile$(8) For Input As #8
Open Dire$ + Infile$(9) For Input As #9
```

'Open output file

```
Open Dire$ + Outfile$ For Append As #10
Print #10, "df1_dc_dp, df1_dc, df1_dc_bg, df1_ac_dp, df1_ac, df1_ac_bg,
df2_dc_dp, df2_dc, df2_dc_bg, df2_ac_dp, df2_ac, df2_ac_bg, edc_ac_dp, edc_ac, edc_ac_bg,
edc_dc_dp, edc_dc, edc_dc_bg, edc_sl_dp, edc_sl, edc_sl_bg, vk_dc_dp, vk_dc, vk_dc_bg"
```

For j = 1 To 242620 'Loading all the data

Input #1, df_depth(j), df_dc(j), df_ac(j)

Next j

For j = 1 To 122689 'Loading all the data

Input #2, edc_acdepth(j), edc_ac(j), edc_acdepth_df1(j)

Next j

For j = 1 To 159598 'Loading all the data

Input #3, df2_acdepth(j), df2_ac(j), df2_acdepth_df1(j)

Next j

For j = 1 To 167888 'Loading all the data

Input #4, df2_dcdepth(j), df2_dc(j), df2_dcdepth_df1(j)

Next j

For j = 1 To 162824 'Loading all the data

Input #5, edc_dc99depth(j), edc_dc99(j), edc_dc99depth_df1(j)

Next j

For j = 1 To 68859 'Loading all the data

Input #6, edc_dc96depth(j), edc_dc96(j), edc_dc96depth_df1(j)

Next j

For j = 1 To 65526 'Loading all the data

Input #7, edc_sl99depth(j), edc_sl99(j), edc_sl99depth_df1(j)

Next j

```
For j = 1 To 19133      'Loading all the data
Input #8, edc_sl96depth(j), edc_sl96(j), edc_sl96depth_df1(j)
Next j
```

```
For j = 1 To 163341    'Loading all the data
Input #9, vk_dcdepth(j), vk_dc(j), vk_dcdepth_df1(j)
Next j
```

```
fromD = Val(Text11.Text) 'Define initial depth and depth span to investigate.
span = Val(Text12.Text)
```

```
Text8.Text = fromD      'Graph of the 1st row from the top
xstart_1 = fromD
Text2.Text = xstart_1 + span
```

```
Text8.Text = fromD      'Graph of the 2nd row from the top
xstart_2 = fromD
Text2.Text = xstart_2 + span
```

```
Text8.Text = fromD      'Graph of the 3rd row from the top
xstart_3 = fromD
Text2.Text = xstart_3 + span
```

```
Text8.Text = fromD      'Graph of the 4th row from the top
xstart_4 = fromD
Text2.Text = xstart_4 + span
```

```
Text8.Text = fromD      'Graph of the 5th row from the top
xstart_5 = fromD
Text2.Text = xstart_5 + span
```

```
Text8.Text = fromD      'Graph of the 6th row from the top
xstart_6 = fromD
Text2.Text = xstart_6 + span
```

```
Text8.Text = fromD      'Graph of the 7th row from the top
xstart_7 = fromD
Text2.Text = xstart_7 + span
```

```
Text8.Text = fromD      'Graph of the 8th row from the top
xstart_8 = fromD
Text2.Text = xstart_8 + span
```

```
Text7.Text = 10          'Graph of the 1st row from the top
Text23.Text = -1
yend_1 = Val(Text7.Text)
ystart_1 = Val(Text23.Text)
```

```
Text3.Text = 0.7         'Graph of the 2nd row from the top
Text5.Text = 0
yend_2 = Val(Text3.Text)
ystart_2 = Val(Text5.Text)
```

```
Text15.Text = 0.7        'Graph of the 5th row from the top
Text16.Text = 0
yend_5 = Val(Text15.Text)
ystart_5 = Val(Text16.Text)
```

```
Text6.Text = 5           'Graph of the 4th row from the top
Text14.Text = -2
yend_4 = Val(Text6.Text)
ystart_4 = Val(Text14.Text)

Text10.Text = 35          'Graph of the 3rd row from the top
Text13.Text = 10
yend_3 = Val(Text10.Text)
ystart_3 = Val(Text13.Text)

Text17.Text = 70          'Graph of the 6th row from the top
Text18.Text = 20
yend_6 = Val(Text17.Text)
ystart_6 = Val(Text18.Text)

Text19.Text = 700         'Graph of the 7th row from the top
Text20.Text = 0
yend_7 = Val(Text19.Text)
ystart_7 = Val(Text20.Text)

Text26.Text = 2           'Graph of the 8th row from the top
Text27.Text = 0.1
yend_8 = Val(Text26.Text)
ystart_8 = Val(Text27.Text)
```

7000

```
span = Val(Text12.Text)

'We define here widths of the graphs.

xstart_1 = Val(Text8.Text)
xend_1 = xstart_1 + span

xstart_2 = Val(Text8.Text)
xend_2 = xstart_2 + span

xstart_3 = Val(Text8.Text)
xend_3 = xstart_3 + span

xstart_4 = Val(Text8.Text)
xend_4 = xstart_4 + span

xstart_5 = Val(Text8.Text)
xend_5 = xstart_5 + span

xstart_6 = Val(Text8.Text)
xend_6 = xstart_6 + span

xstart_7 = Val(Text8.Text)
xend_7 = xstart_7 + span

xstart_8 = Val(Text8.Text)
xend_8 = xstart_8 + span

'Scales of the graphs from 1st graph to 8th graph.
```

```
ystart_1 = Val(Text23.Text)
yend_1 = Val(Text7.Text)

ystart_2 = Val(Text5.Text)
yend_2 = Val(Text3.Text)

ystart_3 = Val(Text13.Text)
yend_3 = Val(Text10.Text)

ystart_4 = Val(Text14.Text)
yend_4 = Val(Text6.Text)

ystart_5 = Val(Text16.Text)
yend_5 = Val(Text15.Text)

ystart_6 = Val(Text18.Text)
yend_6 = Val(Text17.Text)

ystart_7 = Val(Text20.Text)
yend_7 = Val(Text19.Text)

ystart_8 = Val(Text27.Text)
yend_8 = Val(Text26.Text)
```

```
DisplayGraph1 'Subroutine to draw graph
```

```
If Check1.Value = 1 Then
DisplayGraphAscopeMax_1
End If
```

```
If Check2.Value = 1 Then
DisplayGraphAscopeMax_2
End If
```

```
If Check3.Value = 1 Then
DisplayGraphAscopeMax_3
End If
```

```
If Check4.Value = 1 Then
DisplayGraphAscopeMax_4
End If
```

```
If Check5.Value = 1 Then
DisplayGraphAscopeMax_5
End If
```

```
If Check6.Value = 1 Then
DisplayGraphAscopeMax_6
End If
```

```
If Check7.Value = 1 Then
DisplayGraphAscopeMax_7
End If
```

```
If Check15.Value = 1 Then
DisplayGraphAscopeMax_8
End If
```

Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 kyr

```
7500
7550     YesNo
7600
        If RedoFlag Then
            GoTo 7000
        End If

        If YesFlag = False And NoFlag = False Then
            DisplayGraphAscopeMax_1
            DisplayGraphAscopeMax_2
            DisplayGraphAscopeMax_3
            DisplayGraphAscopeMax_4
            DisplayGraphAscopeMax_5
            DisplayGraphAscopeMax_6
            DisplayGraphAscopeMax_7
            DisplayGraphAscopeMax_8

            GoTo 7500
        End If

7800

Close #1
Close #2
Close #3
Close #4
Close #5
Close #6
Close #7
Close #8
Close #9

Close #10

8000     PLN = PLN + 1: Win1clear

        Picture3.Cls
        Picture3.Print row; " data processed"

        StartButton.Enabled = True
        CancelButton.Enabled = False
        EndButton.Enabled = True
End Sub

Private Sub Command1_Click()
'When "Record" command is chosen, data will be saved with an order below.

'DF1 DC    depth, value, backgroud level
'DF1 AC    depth, value, backgroud level
'DF2 DC    depth, value, backgroud level
'DF2 AC    depth, value, backgroud level
'EDCDEP    depth, value, backgroud level
```



```
'EDC DC    depth, value, backgrould level
'EDC SL    depth, value, backgrould level

If maxb_1 > 0 Then maxb_1 = 0.01 * Int(maxb_1 * 100)
If maxb_2 > 0 Then maxb_2 = 0.001 * Int(maxb_2 * 1000)
If maxb_3 > 0 Then maxb_3 = 0.01 * Int(maxb_3 * 100)
If maxb_4 > 0 Then maxb_4 = 0.01 * Int(maxb_4 * 100)
If maxb_5 > 0 Then maxb_5 = 0.01 * Int(maxb_5 * 100)
If maxb_6 > 0 Then maxb_6 = 0.01 * Int(maxb_6 * 100)
If maxb_7 > 0 Then maxb_7 = 0.01 * Int(maxb_7 * 100)
If maxb_8 > 0 Then maxb_8 = 0.01 * Int(maxb_8 * 100)

If Check8.Value = 0 Then
maxx2_1 = iran0: maxy_1 = iran0: maxb_1 = iran0
End If

If Check9.Value = 0 Then
maxx2_2 = iran0: maxy_2 = iran0: maxb_2 = iran0
End If

If Check10.Value = 0 Then
maxx2_3 = iran0: maxy_3 = iran0: maxb_3 = iran0
End If

If Check11.Value = 0 Then
maxx2_4 = iran0: maxy_4 = iran0: maxb_4 = iran0
End If

If Check12.Value = 0 Then
maxx2_5 = iran0: maxy_5 = iran0: maxb_5 = iran0
End If

If Check13.Value = 0 Then
maxx2_6 = iran0: maxy_6 = iran0: maxb_6 = iran0
End If

If Check14.Value = 0 Then
maxx2_7 = iran0: maxy_7 = iran0: maxb_7 = iran0
End If

If Check16.Value = 0 Then
maxx2_8 = iran0: maxy_8 = iran0: maxb_8 = iran0
End If

    iran = "iran"
    Write #10, maxx2_1; maxy_1; maxb_1; maxx2_2; maxy_2; maxb_2; maxx2_5; maxy_5;
maxb_5; maxx2_4; maxy_4; maxb_4; maxx2_3; maxy_3; maxb_3; maxx2_6; maxy_6; maxb_6;
maxx2_7; maxy_7; maxb_7; maxx2_8; maxy_8; maxb_8

    Picture3.Print "Recorded"

    'ID number 1 is for DF1 ECM.
    'ID number 2 is for DF1 AC-ECM.
    'ID number 5 is for DF2 ECM.
    'ID number 4 is for DF1 AC-ECM.
    'ID number 3 is for EDC DEP.
    'ID number 6 is for EDC ECM.
    'ID number 7 is for EDC Sulfate.
    'ID number 8 is for VK ECM.

End Sub
Private Sub Command2_Click()
Check8.Value = 1
```

```
RedoFlag = True
End Sub

Private Sub DisplayGraph1()

    Picture2.Cls
    Picture8.Cls
    Picture10.Cls
    Picture12.Cls
    Picture14.Cls
    Picture4.Cls
    Picture5.Cls
    Picture1.Cls

    Text4.Text = (xstart_1 + xend_1) / 2
    Text2.Text = xend_1

    Graphline1 'Draw grid of the graphs

    If Check1.Value = 1 Then
        maxy_1 = -140
    Else

    End If

    If Check2.Value = 1 Then
        maxy_2 = -140
    Else
    End If

    If Check3.Value = 1 Then
        maxy_3 = -140
    Else
    End If

    If Check4.Value = 1 Then
        maxy_4 = -140
    Else
    End If

    If Check5.Value = 1 Then
        maxy_5 = -140
    Else
    End If

    If Check6.Value = 1 Then
        maxy_6 = -140
    Else
    End If

    If Check7.Value = 1 Then
        maxy_7 = -140
    Else
    End If

    If Check15.Value = 1 Then
        maxy_8 = -140
    Else
    End If
```

```
For j = 2 To 242619
```

Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 kyr

```
Graphx_1 = df_depth(j): pGraphx_1 = df_depth(j - 1)
Graphx_2 = df_depth(j): pGraphx_2 = df_depth(j - 1)

Graphyl_1 = df_dc(j): pGraphyl_1 = df_dc(j - 1)
Graphyl_2 = df_ac(j): pGraphyl_2 = df_ac(j - 1)

If xstart_1 < Graphx_1 And xend_1 > Graphx_1 Then
If Graphx_1 > xstart_1 + (xend_1 - xstart_1) * (1 - Val(Text9.Text)) * 0.5
And Graphx_1 < xend_1 - (xend_1 - xstart_1) * (1 - Val(Text9.Text)) * 0.5 Then

If Check1.Value = 1 Then
If Graphyl_1 > maxy_1 Then
maxy_1 = Graphyl_1
maxx_1 = Graphx_1
maxj_1 = j
maxx2_1 = df_depth(j)

End If
End If
Else
sekisan_Graphyl_1 = sekisan_Graphyl_1 + Graphyl_1
sekisan_Graphyl_1N = sekisan_Graphyl_1N + 1

End If
DisplayGraphAscope_1
End If

If xstart_2 < Graphx_2 And xend_2 > Graphx_2 Then
If Graphx_2 > xstart_2 + (xend_2 - xstart_2) * (1 - Val(Text9.Text)) * 0.5
And Graphx_2 < xend_2 - (xend_2 - xstart_2) * (1 - Val(Text9.Text)) * 0.5 Then
If Check2.Value = 1 Then
If Graphyl_2 > maxy_2 Then
maxy_2 = Graphyl_2
maxx_2 = Graphx_2
maxj_2 = j
maxx2_2 = df_depth(j)

End If
End If
Else
sekisan_Graphyl_2 = sekisan_Graphyl_2 + Graphyl_2
sekisan_Graphyl_2N = sekisan_Graphyl_2N + 1

End If
DisplayGraphAscope_2
End If

Next j
If sekisan_Graphyl_1N > 0 Then
maxb_1 = sekisan_Graphyl_1 / sekisan_Graphyl_1N
sekisan_Graphyl_1N = 0: sekisan_Graphyl_1 = 0
End If

If sekisan_Graphyl_2N > 0 Then
maxb_2 = sekisan_Graphyl_2 / sekisan_Graphyl_2N
sekisan_Graphyl_2N = 0: sekisan_Graphyl_2 = 0
End If
```

```
For j = 2 To 122689

    'edc_acdepth(j), edc_ac(j), edc_acdepth_df1(j)
    Graphx_3 = edc_acdepth_df1(j): pGraphx_3 = edc_acdepth_df1(j - 1)
    Graphyl_3 = edc_ac(j): pGraphyl_3 = edc_ac(j - 1)

    If xstart_3 < Graphx_3 And xend_3 > Graphx_3 Then
        If Graphx_3 > xstart_3 + (xend_3 - xstart_3) * (1 - Val(Text9.Text)) * 0.5 +
xoffset3 And Graphx_3 < xend_3 - (xend_3 - xstart_3) * (1 - Val(Text9.Text)) * 0.5 +
xoffset3 Then
            If Check3.Value = 1 Then
                If Graphyl_3 > maxy_3 Then
                    maxy_3 = Graphyl_3
                    maxx_3 = Graphx_3
                    maxj_3 = j
                    maxx2_3 = edc_acdepth(j)

                End If
            End If
            Else
                sekisan_Graphyl_3 = sekisan_Graphyl_3 + Graphyl_3
                sekisan_Graphyl_3N = sekisan_Graphyl_3N + 1

            End If
            DisplayGraphAscope_3
        End If

    Next j
    If sekisan_Graphyl_3N > 0 Then
        maxb_3 = sekisan_Graphyl_3 / sekisan_Graphyl_3N
        sekisan_Graphyl_3N = 0: sekisan_Graphyl_3 = 0
    End If

For j = 2 To 159598

    Graphx_4 = df2_acdepth_df1(j): pGraphx_4 = df2_acdepth_df1(j - 1)
    Graphyl_4 = df2_ac(j): pGraphyl_4 = df2_ac(j - 1)

    If xstart_4 < Graphx_4 And xend_4 > Graphx_4 Then
        If Graphx_4 > xstart_4 + (xend_4 - xstart_4) * (1 - Val(Text9.Text)) * 0.5 +
xoffset4 And Graphx_4 < xend_4 - (xend_4 - xstart_4) * (1 - Val(Text9.Text)) * 0.5 +
xoffset4 Then
            If Check4.Value = 1 Then
                If Graphyl_4 > maxy_4 Then
                    maxy_4 = Graphyl_4
                    maxx_4 = Graphx_4
                    maxj_4 = j
                    maxx2_4 = df2_acdepth(j)

                End If
            End If
            Else
                sekisan_Graphyl_4 = sekisan_Graphyl_4 + Graphyl_4
                sekisan_Graphyl_4N = sekisan_Graphyl_4N + 1

            End If
            DisplayGraphAscope_4
        End If

    Next j
    If sekisan_Graphyl_4N > 0 Then
        maxb_4 = sekisan_Graphyl_4 / sekisan_Graphyl_4N
        sekisan_Graphyl_4N = 0: sekisan_Graphyl_4 = 0
    End If
```

```

For j = 2 To 167888

    Graphx_5 = df2_dcdepth_df1(j): pGraphx_5 = df2_dcdepth_df1(j - 1)
    Graphyl_5 = df2_dc(j): pGraphyl_5 = df2_dc(j - 1)

    If xstart_5 < Graphx_5 And xend_5 > Graphx_5 Then
        If Graphx_5 > xstart_5 + (xend_5 - xstart_5) * (1 - Val(Text9.Text)) * 0.5 +
xoffset5 And Graphx_5 < xend_5 - (xend_5 - xstart_5) * (1 - Val(Text9.Text)) * 0.5 +
xoffset5 Then
            If Check5.Value = 1 Then
                If Graphyl_5 > maxy_5 Then
                    maxy_5 = Graphyl_5
                    maxx_5 = Graphx_5
                    maxj_5 = j
                    maxx2_5 = df2_dcdepth(j)

                End If
            End If
            Else
                sekisan_Graphyl_5 = sekisan_Graphyl_5 + Graphyl_5
                sekisan_Graphyl_5N = sekisan_Graphyl_5N + 1

            End If
            DisplayGraphAscope_5
        End If

    Next j
    If sekisan_Graphyl_5N > 0 Then
        maxb_5 = sekisan_Graphyl_5 / sekisan_Graphyl_5N
        sekisan_Graphyl_5N = 0: sekisan_Graphyl_5 = 0
    End If

For j = 2 To 162824

    Graphx_6 = edc_dc99depth_df1(j): pGraphx_6 = edc_dc99depth_df1(j - 1)
    Graphyl_6 = edc_dc99(j): pGraphyl_6 = edc_dc99(j - 1)

    If xstart_6 < Graphx_6 And xend_6 > Graphx_6 Then
        If Graphx_6 > xstart_6 + (xend_6 - xstart_6) * (1 - Val(Text9.Text)) * 0.5 +
xoffset6 And Graphx_6 < xend_6 - (xend_6 - xstart_6) * (1 - Val(Text9.Text)) * 0.5 +
xoffset6 Then
            If Check6.Value = 1 Then
                If Graphyl_6 > maxy_6 Then
                    maxy_6 = Graphyl_6
                    maxx_6 = Graphx_6
                    maxj_6 = j
                    maxx2_6 = edc_dc99depth(j)

                End If
            End If
            Else
                sekisan_Graphyl_6 = sekisan_Graphyl_6 + Graphyl_6
                sekisan_Graphyl_6N = sekisan_Graphyl_6N + 1

            End If
            DisplayGraphAscope_6
        End If

    Next j
    If sekisan_Graphyl_6N > 0 Then
        maxb_6 = sekisan_Graphyl_6 / sekisan_Graphyl_6N
        sekisan_Graphyl_6N = 0: sekisan_Graphyl_6 = 0
    End If

```

```
For j = 2 To 68859

    Graphx_6 = edc_dc96depth_df1(j): pGraphx_6 = edc_dc96depth_df1(j - 1)
    Graphyl_6 = edc_dc96(j): pGraphyl_6 = edc_dc96(j - 1)

    If xstart_6 < Graphx_6 And xend_6 > Graphx_6 Then
        If Graphx_6 > xstart_6 + (xend_6 - xstart_6) * (1 - Val(Text9.Text)) * 0.5 +
xoffset6 And Graphx_6 < xend_6 - (xend_6 - xstart_6) * (1 - Val(Text9.Text)) * 0.5 +
xoffset6 Then
            If Check6.Value = 1 Then
                If Graphyl_6 > maxy_6 Then
                    maxy_6 = Graphyl_6
                    maxx_6 = Graphx_6
                    maxj_6 = j
                    maxx2_6 = edc_dc96depth(j)

                End If
                End If
                Else
                    sekisan_Graphyl_6 = sekisan_Graphyl_6 + Graphyl_6
                    sekisan_Graphyl_6N = sekisan_Graphyl_6N + 1

                End If
                DisplayGraphAscope_6
                End If

    Next j

    If sekisan_Graphyl_6N > 0 Then
        maxb_6 = sekisan_Graphyl_6 / sekisan_Graphyl_6N
        sekisan_Graphyl_6N = 0: sekisan_Graphyl_6 = 0
    End If

For j = 2 To 65526

    Graphx_7 = edc_sl99depth_df1(j): pGraphx_7 = edc_sl99depth_df1(j - 1)
    Graphyl_7 = edc_sl99(j): pGraphyl_7 = edc_sl99(j - 1)

    If xstart_7 < Graphx_7 And xend_7 > Graphx_7 Then
        If Graphx_7 > xstart_7 + (xend_7 - xstart_7) * (1 - Val(Text9.Text)) * 0.5 +
xoffset7 And Graphx_7 < xend_7 - (xend_7 - xstart_7) * (1 - Val(Text9.Text)) * 0.5 +
xoffset7 Then
            If Check7.Value = 1 Then
                If Graphyl_7 > maxy_7 Then
                    maxy_7 = Graphyl_7
                    maxx_7 = Graphx_7
                    maxj_7 = j
                    maxx2_7 = edc_sl99depth(j)

                End If
                End If
                Else
                    sekisan_Graphyl_7 = sekisan_Graphyl_7 + Graphyl_7
                    sekisan_Graphyl_7N = sekisan_Graphyl_7N + 1

                End If
                DisplayGraphAscope_7
                End If

    Next j

    If sekisan_Graphyl_7N > 0 Then
```

Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 kyr

```
maxb_7 = sekisan_Graphyl_7 / sekisan_Graphyl_7N
sekisan_Graphyl_7N = 0: sekisan_Graphyl_7 = 0
End If

For j = 2 To 19133

    Graphx_7 = edc_sl96depth_df1(j): pGraphx_7 = edc_sl96depth_df1(j - 1)
    Graphyl_7 = edc_sl96(j): pGraphyl_7 = edc_sl96(j - 1)

    If xstart_7 < Graphx_7 And xend_7 > Graphx_7 Then
        If Graphx_7 > xstart_7 + (xend_7 - xstart_7) * (1 - Val(Text9.Text)) * 0.5 +
xoffset7 And Graphx_7 < xend_7 - (xend_7 - xstart_7) * (1 - Val(Text9.Text)) * 0.5 +
xoffset7 Then
            If Check7.Value = 1 Then
                If Graphyl_7 > maxy_7 Then
                    maxy_7 = Graphyl_7
                    maxx_7 = Graphx_7
                    maxj_7 = j
                    maxx2_7 = edc_sl96depth(j)

                End If
            End If
            Else
                sekisan_Graphyl_7 = sekisan_Graphyl_7 + Graphyl_7
                sekisan_Graphyl_7N = sekisan_Graphyl_7N + 1

            End If
            DisplayGraphAscope_7
        End If

    Next j
    If sekisan_Graphyl_7N > 0 Then
        maxb_7 = sekisan_Graphyl_7 / sekisan_Graphyl_7N
        sekisan_Graphyl_7N = 0: sekisan_Graphyl_7 = 0
        End If

For j = 2 To 163341

    Graphx_8 = vk_dcdepth_df1(j): pGraphx_8 = vk_dcdepth_df1(j - 1)
    Graphyl_8 = vk_dc(j): pGraphyl_8 = vk_dc(j - 1)

    If xstart_8 < Graphx_8 And xend_8 > Graphx_8 Then
        If Graphx_8 > xstart_8 + (xend_8 - xstart_8) * (1 - Val(Text9.Text)) * 0.5 +
xoffset8 And Graphx_8 < xend_8 - (xend_8 - xstart_8) * (1 - Val(Text9.Text)) * 0.5 +
xoffset8 Then
            If Check15.Value = 1 Then
                If Graphyl_8 > maxy_8 Then
                    maxy_8 = Graphyl_8
                    maxx_8 = Graphx_8
                    maxj_8 = j
                    maxx2_8 = vk_dcdepth(j)

                End If
            End If
            Else
                sekisan_Graphyl_8 = sekisan_Graphyl_8 + Graphyl_8
                sekisan_Graphyl_8N = sekisan_Graphyl_8N + 1

            End If
            DisplayGraphAscope_8
        End If

    Next j
```

Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 kyr

```
        If sekisan_Graphyl_8N > 0 Then
            maxb_8 = sekisan_Graphyl_8 / sekisan_Graphyl_8N
            sekisan_Graphyl_8N = 0: sekisan_Graphyl_8 = 0
        End If

End Sub
Private Sub P10Button_Click()
Picture3.Cls
    RedoFlag = True
    Text8.Text = Val(Text8.Text) + 1
End Sub
Private Sub P1Button_Click()
Picture3.Cls
    RedoFlag = True
    Text8.Text = Val(Text8.Text) + 0.1
End Sub
Private Sub M10Button_Click()
Picture3.Cls
    RedoFlag = True
    Text8.Text = Val(Text8.Text) - 1
End Sub
Private Sub M1Button_Click()
Picture3.Cls
    RedoFlag = True
    Text8.Text = Val(Text8.Text) - 0.1
End Sub
Private Sub RedoButton_Click()
    RedoFlag = True
    Text8.Text = Val(Text8.Text) + 50
End Sub

Private Sub CancelButton_Click()
    CancelFlag = True
End Sub

Private Sub EndButton_Click()
    End
End Sub

Private Sub YesNo()
    'This subroutine will return "true" either to YesFlag or NoFlag.

    'Initial values
    CancelFlag = False
    YesFlag = False
    NoFlag = False
    P10Flag_1 = False
    P10Flag_2 = False
        P10Flag_3 = False
        P10Flag_4 = False
        P10Flag_5 = False
        P10Flag_6 = False
        P10Flag_7 = False
        P10Flag_8 = False

    P1Flag_1 = False
    P1Flag_2 = False
        P1Flag_3 = False
        P1Flag_4 = False
        P1Flag_5 = False
        P1Flag_6 = False
        P1Flag_7 = False
        P1Flag_8 = False
```



```
M10Flag_1 = False
M10Flag_2 = False
    M10Flag_3 = False
    M10Flag_4 = False
    M10Flag_5 = False
    M10Flag_6 = False
    M10Flag_7 = False
    M10Flag_8 = False

M1Flag_1 = False
M1Flag_2 = False
    M1Flag_3 = False
    M1Flag_4 = False
    M1Flag_5 = False
    M1Flag_6 = False
    M1Flag_7 = False
    M1Flag_8 = False

RedoFlag = False
Redo2Flag = False

GoFlag = False
switch1 = 0

StartButton.Enabled = False
CancelButton.Enabled = True
EndButton.Enabled = True

While switch1 = 0

    'This section checks if user chose cancel or not.
    Dummy = DoEvents()

    If CancelFlag Then
        CancelFlag = False
        Cancelled = 1
        switch1 = 1

    PLN = PLN + 1: Win1clear
        Picture3.Print " cancelled "
    End If

    'This section checks if user chose YES or NO.
    If YesFlag Then
        switch1 = 1
    End If

    If NoFlag Then
        switch1 = 1
    End If

    'This section checks if user chose plus or minus.
    If PlFlag_1 Then
        EraseMax_1
        Graphx_1 = df_depth(maxj_1 + 1): Graphy1_1 = df_dc(maxj_1 + 1)
        maxj_1 = maxj_1 + 1
        maxy_1 = Graphy1_1
        maxx_1 = Graphx_1
        maxx2_1 = df_depth(maxj_1 + 1)
        DisplayGraph1
        switch1 = 1
    End If
```

```

If PlFlag_2 Then
  EraseMax_2
  'EraseMax3
  Graphx_2 = df_depth(maxj_2 + 1): Graphy1_2 = df_ac(maxj_2
+ 1)

  maxj_2 = maxj_2 + 1
  maxy_2 = Graphy1_2
  maxx_2 = Graphx_2
  maxx2_2 = df_depth(maxj_2 + 1)
  DisplayGraph1
  switch1 = 1
End If

If PlFlag_3 Then
  EraseMax_3
  'EraseMax3
  Graphx_3 = edc_acdepth_df1(maxj_3 + 1): Graphy1_3 =
edc_ac(maxj_3 + 1)

  maxj_3 = maxj_3 + 1
  maxy_3 = Graphy1_3
  maxx_3 = Graphx_3
  maxx2_3 = edc_acdepth(maxj_3 + 1)
  DisplayGraph1
  switch1 = 1
End If

If PlFlag_4 Then
  EraseMax_4
  'EraseMax3
  Graphx_4 = df2_acdepth_df1(maxj_4 + 1): Graphy1_4 =
df2_ac(maxj_4 + 1)

  maxj_4 = maxj_4 + 1
  maxy_4 = Graphy1_4
  maxx_4 = Graphx_4
  maxx2_4 = df2_acdepth(maxj_4 + 1)
  DisplayGraph1
  switch1 = 1
End If

If PlFlag_5 Then
  EraseMax_5
  'EraseMax3
  Graphx_5 = df2_dcdepth_df1(maxj_5 + 1): Graphy1_5 =
df2_dc(maxj_5 + 1)

  maxj_5 = maxj_5 + 1
  maxy_5 = Graphy1_5
  maxx_5 = Graphx_5
  maxx2_5 = df2_dcdepth(maxj_5 + 1)
  DisplayGraph1
  switch1 = 1
End If

If PlFlag_6 Then
  EraseMax_6
  'EraseMax3
  Graphx_6 = edc_dc99depth_df1(maxj_6 + 1): Graphy1_6 =
edc_dc99(maxj_6 + 1)

  maxj_6 = maxj_6 + 1
  maxy_6 = Graphy1_6
  maxx_6 = Graphx_6
  maxx2_6 = edc_dc99depth(maxj_6 + 1)
  DisplayGraph1
  switch1 = 1
End If

```

```

edc_sl99(maxj_7 + 1)

If PlFlag_7 Then
    EraseMax_7
    'EraseMax3
    Graphx_7 = edc_sl99depth_df1(maxj_7 + 1): Graphy1_7 =

    maxj_7 = maxj_7 + 1
    maxy_7 = Graphy1_7
    maxx_7 = Graphx_7
    maxx2_7 = edc_sl99depth(maxj_7 + 1)
    DisplayGraph1
    switch1 = 1
End If

vk_dc(maxj_8 + 1)

If PlFlag_8 Then
    EraseMax_8
    'EraseMax3
    Graphx_8 = vk_dcdepth_df1(maxj_8 + 1): Graphy1_8 =

    maxj_8 = maxj_8 + 1
    maxy_8 = Graphy1_8
    maxx_8 = Graphx_8
    maxx2_8 = vk_dcdepth(maxj_8 + 1)
    DisplayGraph1
    switch1 = 1
End If

df_dc(maxj_1 + 10)

If Pl0Flag_1 Then
    EraseMax_1
    'EraseMax3
    Graphx_1 = df_depth(maxj_1 + 10): Graphy1_1 =

    maxj_1 = maxj_1 + 10
    maxy_1 = Graphy1_1
    maxx_1 = Graphx_1
    maxx2_1 = df_depth(maxj_1 + 10)
    DisplayGraph1
    switch1 = 1
End If

df_ac(maxj_2 + 10)

If Pl0Flag_2 Then
    EraseMax_2
    'EraseMax3
    Graphx_2 = df_depth(maxj_2 + 10): Graphy1_2 =

    maxj_2 = maxj_2 + 10
    maxy_2 = Graphy1_2
    maxx_2 = Graphx_2
    maxx2_2 = df_depth(maxj_2 + 10)
    DisplayGraph1
    switch1 = 1
End If

edc_ac(maxj_3 + 10)

If Pl0Flag_3 Then
    EraseMax_3
    'EraseMax3
    Graphx_3 = edc_acdepth_df1(maxj_3 + 10): Graphy1_3 =

    maxj_3 = maxj_3 + 10
    maxy_3 = Graphy1_3

```

Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 k yr

```
maxx_3 = Graphx_3
maxx2_3 = edc_acdepth(maxj_3 + 10)
DisplayGraph1
switch1 = 1
End If

If P10Flag_4 Then
  EraseMax_4
  'EraseMax3
  Graphx_4 = df2_acdepth_df1(maxj_4 + 10): Graphy1_4 =
df2_ac(maxj_4 + 10)
  maxx_4 = Graphx_4
  maxx2_4 = df2_acdepth(maxj_4 + 10)
  DisplayGraph1
  switch1 = 1
End If

If P10Flag_5 Then
  EraseMax_5
  'EraseMax3
  Graphx_5 = df2_dcdepth_df1(maxj_5 + 10): Graphy1_5 =
df2_dc(maxj_5 + 10)
  maxx_5 = Graphx_5
  maxx2_5 = df2_dcdepth(maxj_5 + 10)
  DisplayGraph1
  switch1 = 1
End If

If P10Flag_6 Then
  EraseMax_6
  'EraseMax3
  Graphx_6 = edc_dc99depth_df1(maxj_6 + 10): Graphy1_6 =
edc_dc99(maxj_6 + 10)
  maxx_6 = Graphx_6
  maxx2_6 = edc_dc99depth(maxj_6 + 10)
  DisplayGraph1
  switch1 = 1
End If

If P10Flag_7 Then
  EraseMax_7
  'EraseMax3
  Graphx_7 = edc_sl99depth_df1(maxj_7 + 10): Graphy1_7 =
edc_sl99(maxj_7 + 10)
  maxx_7 = Graphx_7
  maxx2_7 = edc_sl99depth(maxj_7 + 10)
  DisplayGraph1
  switch1 = 1
End If

If P10Flag_8 Then
  EraseMax_8
  'EraseMax3
  Graphx_8 = vk_dcdepth_df1(maxj_8 + 10): Graphy1_8 =
vk_dc(maxj_8 + 10)
  maxx_8 = Graphx_8
  maxx2_8 = vk_dcdepth(maxj_8 + 10)
  DisplayGraph1
  switch1 = 1
End If
```

Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 kyr

```

maxy_8 = Graphy1_8
maxx_8 = Graphx_8
maxx2_8 = vk_dcdepth(maxj_8 + 10)
DisplayGraph1
switch1 = 1
End If

'edc_acdepth(122689), edc_ac(122689), edc_acdepth_df1(122689)

If M1Flag_1 Then
    EraseMax_1
    'EraseMax3
    Graphx_1 = df_depth(maxj_1 - 1): Graphy1_1 =
df_dc(maxj_1 - 1)

    maxx_1 = Graphx_1
    maxy_1 = Graphy1_1
    maxx2_1 = df_depth(maxj_1 - 1)
    DisplayGraph1
    switch1 = 1
End If

If M1Flag_2 Then
    EraseMax_2
    'EraseMax3
    Graphx_2 = df_depth(maxj_2 - 1): Graphy1_2 =
df_ac(maxj_2 - 1)

    maxx_2 = Graphx_2
    maxy_2 = Graphy1_2
    maxx2_2 = df_depth(maxj_2 - 1)
    DisplayGraph1
    switch1 = 1
End If

If M1Flag_3 Then
    EraseMax_3
    'EraseMax3
    Graphx_3 = edc_acdepth_df1(maxj_3 - 1): Graphy1_3 =
edc_ac(maxj_3 - 1)

    maxx_3 = Graphx_3
    maxy_3 = Graphy1_3
    maxx2_3 = edc_acdepth(maxj_3 - 1)
    DisplayGraph1
    switch1 = 1
End If

If M1Flag_4 Then
    EraseMax_4
    'EraseMax3
    Graphx_4 = df2_acdepth_df1(maxj_4 - 1): Graphy1_4 =
df2_ac(maxj_4 - 1)

    maxx_4 = Graphx_4
    maxy_4 = Graphy1_4
    maxx2_4 = df2_acdepth(maxj_4 - 1)
    DisplayGraph1
    switch1 = 1

```

```

End If

If M1Flag_5 Then
    EraseMax_5
    'EraseMax3
    Graphx_5 = df2_dcdepth_df1(maxj_5 - 1): Graphy1_5 =
df2_dc(maxj_5 - 1)
    maxj_5 = maxj_5 - 1
    maxy_5 = Graphy1_5
    maxx_5 = Graphx_5
    maxx2_5 = df2_dcdepth(maxj_5 - 1)
    DisplayGraph1
    switch1 = 1
End If

If M1Flag_6 Then
    EraseMax_6
    'EraseMax3
    Graphx_6 = edc_dc99depth_df1(maxj_6 - 1): Graphy1_6 =
edc_dc99(maxj_6 - 1)
    maxj_6 = maxj_6 - 1
    maxy_6 = Graphy1_6
    maxx_6 = Graphx_6
    maxx2_6 = edc_dc99depth(maxj_6 - 1)
    DisplayGraph1
    switch1 = 1
End If

If M1Flag_7 Then
    EraseMax_7
    'EraseMax3
    Graphx_7 = edc_sl99depth_df1(maxj_7 - 1): Graphy1_7 =
edc_sl99(maxj_7 - 1)
    maxj_7 = maxj_7 - 1
    maxy_7 = Graphy1_7
    maxx_7 = Graphx_7
    maxx2_7 = edc_sl99depth(maxj_7 - 1)
    DisplayGraph1
    switch1 = 1
End If

If M1Flag_8 Then
    EraseMax_8
    'EraseMax3
    Graphx_8 = vk_dcdepth_df1(maxj_8 - 1): Graphy1_8 =
vk_dc(maxj_8 - 1)
    maxj_8 = maxj_8 - 1
    maxy_8 = Graphy1_8
    maxx_8 = Graphx_8
    maxx2_8 = vk_dcdepth(maxj_8 - 1)
    DisplayGraph1
    switch1 = 1
End If

If M10Flag_1 Then
    EraseMax_1
    'EraseMax3
    Graphx_1 = df_depth(maxj_1 - 10): Graphy1_1 =
df_dc(maxj_1 - 10)
    maxj_1 = maxj_1 - 10
    maxy_1 = Graphy1_1

```

Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 k yr

```

maxx_1 = Graphx_1
maxx2_1 = df_depth(maxj_1 - 10)
DisplayGraph1
switch1 = 1
End If

If M10Flag_2 Then
EraseMax_2
'EraseMax3
Graphx_2 = df_depth(maxj_2 - 10): Graphy1_2 =
df_ac(maxj_2 - 10)

maxj_2 = maxj_2 - 10
maxy_2 = Graphy1_2
maxx_2 = Graphx_2
maxx2_2 = df_depth(maxj_2 - 10)
DisplayGraph1
switch1 = 1
End If

If M10Flag_3 Then
EraseMax_3
'EraseMax3
Graphx_3 = edc_acdepth_df1(maxj_3 - 10): Graphy1_3 =
edc_ac(maxj_3 - 10)

maxj_3 = maxj_3 - 10
maxy_3 = Graphy1_3
maxx_3 = Graphx_3
maxx2_3 = edc_acdepth(maxj_3 - 10)
DisplayGraph1
switch1 = 1
End If

If M10Flag_4 Then
EraseMax_4
'EraseMax3
Graphx_4 = df2_acdepth_df1(maxj_4 - 10): Graphy1_4 =
df2_ac(maxj_4 - 10)

maxj_4 = maxj_4 - 10
maxy_4 = Graphy1_4
maxx_4 = Graphx_4
maxx2_4 = df2_acdepth(maxj_4 - 10)
DisplayGraph1
switch1 = 1
End If

If M10Flag_5 Then
EraseMax_5
'EraseMax3
Graphx_5 = df2_dcdepth_df1(maxj_5 - 10): Graphy1_5 =
df2_dc(maxj_5 - 10)

maxj_5 = maxj_5 - 10
maxy_5 = Graphy1_5
maxx_5 = Graphx_5
maxx2_5 = df2_dcdepth(maxj_5 - 10)
DisplayGraph1
switch1 = 1
End If

If M10Flag_6 Then
EraseMax_6
'EraseMax3
Graphx_6 = edc_dc99depth_df1(maxj_6 - 10): Graphy1_6 =
edc_dc99(maxj_6 - 10)

```

Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 kyr

```

        maxj_6 = maxj_6 - 10
        maxy_6 = Graphy1_6
        maxx_6 = Graphx_6
        maxx2_6 = edc_dc99depth(maxj_6 - 10)
        DisplayGraph1
        switch1 = 1
    End If

    If M10Flag_7 Then
        EraseMax_7
        'EraseMax3
        Graphx_7 = edc_sl99depth_df1(maxj_7 - 10): Graphy1_7 =
edc_sl99(maxj_7 - 10)

        maxj_7 = maxj_7 - 10
        maxy_7 = Graphy1_7
        maxx_7 = Graphx_7
        maxx2_7 = edc_sl99depth(maxj_7 - 10)
        DisplayGraph1
        switch1 = 1
    End If

    If M10Flag_8 Then
        EraseMax_8
        'EraseMax3
        Graphx_8 = vk_dcdepth_df1(maxj_8 - 10): Graphy1_8 =
vk_dc(maxj_8 - 10)

        maxj_8 = maxj_8 - 10
        maxy_8 = Graphy1_8
        maxx_8 = Graphx_8
        maxx2_8 = vk_dcdepth(maxj_8 - 10)
        DisplayGraph1
        switch1 = 1
    End If

    'This section checks if user clicked REDO or not.
    If RedoFlag Then
        switch1 = 1
    End If

    If Redo2Flag Then
        switch1 = 1
    End If

Wend

StartButton.Enabled = False
CancelButton.Enabled = True
EndButton.Enabled = True

End Sub

Private Sub Graphline1() 'Draw grids of graphs
    Picture2.Cls
    Picture2.Scale (xstart_1, yend_1)-(xend_1, ystart_1)
    Picture2.Line ((xstart_1 + xend_1) / 2, ystart_1)-((xstart_1 + xend_1) /
2, yend_1), &HC0C0C0 'Center line

```



```

Picture8.Cls
Picture8.Scale (xstart_2, yend_2)-(xend_2, ystart_2)
Picture8.Line ((xstart_2 + xend_2) / 2, ystart_2)-((xstart_2 + xend_2) /
2, yend_2), &HC0C0C0 'Center line

Picture10.Cls
xstart_3mod = xstart_3 + xoffset3
xend_3mod = xend_3 + xoffset3

Picture10.Scale (xstart_3mod, yend_3)-(xend_3mod, ystart_3)
Picture10.Line ((xstart_3 + xend_3) / 2 + xoffset3, ystart_3)-((xstart_3
+ xend_3) / 2 + xoffset3, yend_3), &HC0C0C0 'Center line

Picture12.Cls
xstart_4mod = xstart_4 + xoffset4
xend_4mod = xend_4 + xoffset4

Picture12.Scale (xstart_4mod, yend_4)-(xend_4mod, ystart_4)
Picture12.Line ((xstart_4 + xend_4) / 2 + xoffset4, ystart_4)-((xstart_4
+ xend_4) / 2 + xoffset4, yend_4), &HC0C0C0 'Center line

Picture14.Cls
xstart_5mod = xstart_5 + xoffset5
xend_5mod = xend_5 + xoffset5
Picture14.Scale (xstart_5mod, yend_5)-(xend_5mod, ystart_5)
Picture14.Line ((xstart_5 + xend_5) / 2 + xoffset5, ystart_5)-((xstart_5
+ xend_5) / 2 + xoffset5, yend_5), &HC0C0C0 'Center line

Picture4.Cls
xstart_6mod = xstart_6 + xoffset6
xend_6mod = xend_6 + xoffset6
Picture4.Scale (xstart_6mod, yend_6)-(xend_6mod, ystart_6)
Picture4.Line ((xstart_6 + xend_6) / 2 + xoffset6, ystart_6)-((xstart_6 +
xend_6) / 2 + xoffset6, yend_6), &HC0C0C0 'Center line

Picture5.Cls
xstart_7mod = xstart_7 + xoffset7
xend_7mod = xend_7 + xoffset7
Picture5.Scale (xstart_7mod, yend_7)-(xend_7mod, ystart_7)
Picture5.Line ((xstart_7 + xend_7) / 2 + xoffset7, ystart_7)-((xstart_7 +
xend_7) / 2 + xoffset7, yend_7), &HC0C0C0 'Center line

Picture1.Cls
xstart_8mod = xstart_8 + xoffset8
xend_8mod = xend_8 + xoffset8
Picture1.Scale (xstart_8mod, yend_8)-(xend_8mod, ystart_8)
Picture1.Line ((xstart_8 + xend_8) / 2 + xoffset8, ystart_8)-((xstart_8 +
xend_8) / 2 + xoffset8, yend_8), &HC0C0C0 'Center line

End Sub

Private Sub DisplayGraphAscope_1()
'Draw data points on Window2 -----
dotsize = 1 * (xend_1 - xstart_1) / 1200
Picture2.Circle (Graphx_1, Graphy1_1), dotsize, &HFF&
Picture2.Line (pGraphx_1, pGraphy1_1)-(Graphx_1, Graphy1_1), &HFF&

End Sub

Private Sub DisplayGraphAscope_2()
'Draw data points on Window2 -----
dotsize = 1 * (xend_2 - xstart_2) / 1200
Picture8.Circle (Graphx_2, Graphy1_2), dotsize, &HFF&

```

Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 kyr

```
Picture8.Line (pGraphx_2, pGraphyl_2)-(Graphx_2, Graphyl_2), &HFF&

End Sub
Private Sub DisplayGraphAscope_3()
'Draw data points on Window2 -----
dotsize = 1 * (xend_3 - xstart_3) / 1200
Picture10.Circle (Graphx_3, Graphyl_3), dotsize, &HFF&
Picture10.Line (pGraphx_3, pGraphyl_3)-(Graphx_3, Graphyl_3),
&HFF&

End Sub
Private Sub DisplayGraphAscope_4()
'Draw data points on Window2 -----
dotsize = 1 * (xend_4 - xstart_4) / 1200
Picture12.Circle (Graphx_4, Graphyl_4), dotsize, &HFF&
Picture12.Line (pGraphx_4, pGraphyl_4)-(Graphx_4, Graphyl_4),
&HFF&

End Sub
Private Sub DisplayGraphAscope_5()
'Draw data points on Window2 -----
dotsize = 1 * (xend_5 - xstart_5) / 1200
Picture14.Circle (Graphx_5, Graphyl_5), dotsize, &HFF&
Picture14.Line (pGraphx_5, pGraphyl_5)-(Graphx_5, Graphyl_5),
&HFF&

End Sub
Private Sub DisplayGraphAscope_6()
'Draw data points on Window2 -----
dotsize = 1 * (xend_6 - xstart_6) / 1200
Picture4.Circle (Graphx_6, Graphyl_6), dotsize, &HFF&
Picture4.Line (pGraphx_6, pGraphyl_6)-(Graphx_6, Graphyl_6), &HFF&

End Sub
Private Sub DisplayGraphAscope_7()
'Draw data points on Window2 -----
dotsize = 1 * (xend_7 - xstart_7) / 1200
Picture5.Circle (Graphx_7, Graphyl_7), dotsize, &HFF&
Picture5.Line (pGraphx_7, pGraphyl_7)-(Graphx_7, Graphyl_7), &HFF&

End Sub
Private Sub DisplayGraphAscope_8()
'Draw data points on Window2 -----
dotsize = 1 * (xend_8 - xstart_8) / 1200
Picture1.Circle (Graphx_8, Graphyl_8), dotsize, &HFF&
Picture1.Line (pGraphx_8, pGraphyl_8)-(Graphx_8, Graphyl_8), &HFF&

End Sub

Private Sub DisplayGraphAscopeMax_1()
'Draw data points on Window2 -----
dotsize = 1 * (xend_1 - xstart_1) / 1200
Picture2.Circle (maxx_1, maxy_1), dotsize * 1, &HFF0000
Picture2.Circle (maxx_1, maxy_1), dotsize * 2, &HFF0000
Picture2.Circle (maxx_1, maxy_1), dotsize * 3, &HFF0000

End Sub
Private Sub DisplayGraphAscopeMax_2()
'Draw data points on Window2 -----
dotsize = 1 * (xend_2 - xstart_2) / 1200
Picture8.Circle (maxx_2, maxy_2), dotsize * 1, &HFF0000
Picture8.Circle (maxx_2, maxy_2), dotsize * 2, &HFF0000
Picture8.Circle (maxx_2, maxy_2), dotsize * 3, &HFF0000

End Sub
Private Sub DisplayGraphAscopeMax_3()
```

```

        'Draw data points on Window2 -----
        dotsize = 1 * (xend_3 - xstart_3) / 1200
        Picture10.Circle (maxx_3, maxy_3), dotsize * 1, &HFF0000
        Picture10.Circle (maxx_3, maxy_3), dotsize * 2, &HFF0000
        Picture10.Circle (maxx_3, maxy_3), dotsize * 3, &HFF0000
End Sub
Private Sub DisplayGraphAscopeMax_4()
    'Draw data points on Window2 -----
    dotsize = 1 * (xend_4 - xstart_4) / 1200
    Picture12.Circle (maxx_4, maxy_4), dotsize * 1, &HFF0000
    Picture12.Circle (maxx_4, maxy_4), dotsize * 2, &HFF0000
    Picture12.Circle (maxx_4, maxy_4), dotsize * 3, &HFF0000
End Sub
Private Sub DisplayGraphAscopeMax_5()
    'Draw data points on Window2 -----
    dotsize = 1 * (xend_5 - xstart_5) / 1200
    Picture14.Circle (maxx_5, maxy_5), dotsize * 1, &HFF0000
    Picture14.Circle (maxx_5, maxy_5), dotsize * 2, &HFF0000
    Picture14.Circle (maxx_5, maxy_5), dotsize * 3, &HFF0000
End Sub
Private Sub DisplayGraphAscopeMax_6()
    'Draw data points on Window2 -----
    dotsize = 1 * (xend_6 - xstart_6) / 1200
    Picture4.Circle (maxx_6, maxy_6), dotsize * 1, &HFF0000
    Picture4.Circle (maxx_6, maxy_6), dotsize * 2, &HFF0000
    Picture4.Circle (maxx_6, maxy_6), dotsize * 3, &HFF0000
End Sub
Private Sub DisplayGraphAscopeMax_7()
    'Draw data points on Window2 -----
    dotsize = 1 * (xend_7 - xstart_7) / 1200
    Picture5.Circle (maxx_7, maxy_7), dotsize * 1, &HFF0000
    Picture5.Circle (maxx_7, maxy_7), dotsize * 2, &HFF0000
    Picture5.Circle (maxx_7, maxy_7), dotsize * 3, &HFF0000
End Sub
Private Sub DisplayGraphAscopeMax_8()
    'Draw data points on Window2 -----
    dotsize = 1 * (xend_8 - xstart_8) / 1200
    Picture1.Circle (maxx_8, maxy_8), dotsize * 1, &HFF0000
    Picture1.Circle (maxx_8, maxy_8), dotsize * 2, &HFF0000
    Picture1.Circle (maxx_8, maxy_8), dotsize * 3, &HFF0000
End Sub

Private Sub EraseMax_1()
    'Draw data points on Window2 -----
    dotsize = 1 * (xend_1 - xstart_1) / 1200
    Picture2.Circle (maxx_1, maxy_1), dotsize * 1, &HFFFFFF
    Picture2.Circle (maxx_1, maxy_1), dotsize * 2, &HFFFFFF
    Picture2.Circle (maxx_1, maxy_1), dotsize * 3, &HFFFFFF
End Sub
Private Sub EraseMax_2()
    'Draw data points on Window2 -----
    dotsize = 1 * (xend_2 - xstart_2) / 1200
    Picture8.Circle (maxx_2, maxy_2), dotsize * 1, &HFFFFFF
    Picture8.Circle (maxx_2, maxy_2), dotsize * 2, &HFFFFFF
    Picture8.Circle (maxx_2, maxy_2), dotsize * 3, &HFFFFFF
End Sub
Private Sub EraseMax_3()
    'Draw data points on Window2 -----
    dotsize = 1 * (xend_3 - xstart_3) / 1200
    Picture10.Circle (maxx_3, maxy_3), dotsize * 1, &HFFFFFF

```

Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 k yr

```
Picture10.Circle (maxx_3, maxy_3), dotsize * 2, &HFFFFFFF
Picture10.Circle (maxx_3, maxy_3), dotsize * 3, &HFFFFFFF

End Sub
Private Sub EraseMax_4()
'Draw data points on Window2 -----
dotsize = 1 * (xend_4 - xstart_4) / 1200
Picture12.Circle (maxx_4, maxy_4), dotsize * 1, &HFFFFFFF
Picture12.Circle (maxx_4, maxy_4), dotsize * 2, &HFFFFFFF
Picture12.Circle (maxx_4, maxy_4), dotsize * 3, &HFFFFFFF

End Sub
Private Sub EraseMax_5()
'Draw data points on Window2 -----
dotsize = 1 * (xend_7 - xstart_7) / 1200
Picture14.Circle (maxx_7, maxy_7), dotsize * 1, &HFFFFFFF
Picture14.Circle (maxx_7, maxy_7), dotsize * 2, &HFFFFFFF
Picture14.Circle (maxx_7, maxy_7), dotsize * 3, &HFFFFFFF

End Sub
Private Sub EraseMax_6()
'Draw data points on Window2 -----
dotsize = 1 * (xend_6 - xstart_6) / 1200
Picture4.Circle (maxx_6, maxy_6), dotsize * 1, &HFFFFFFF
Picture4.Circle (maxx_6, maxy_6), dotsize * 2, &HFFFFFFF
Picture4.Circle (maxx_6, maxy_6), dotsize * 3, &HFFFFFFF

End Sub
Private Sub EraseMax_7()
'Draw data points on Window2 -----
dotsize = 1 * (xend_7 - xstart_7) / 1200
Picture5.Circle (maxx_7, maxy_7), dotsize * 1, &HFFFFFFF
Picture5.Circle (maxx_7, maxy_7), dotsize * 2, &HFFFFFFF
Picture5.Circle (maxx_7, maxy_7), dotsize * 3, &HFFFFFFF

End Sub
Private Sub EraseMax_8()
'Draw data points on Window2 -----
dotsize = 1 * (xend_8 - xstart_8) / 1200
Picture1.Circle (maxx_8, maxy_8), dotsize * 1, &HFFFFFFF
Picture1.Circle (maxx_8, maxy_8), dotsize * 2, &HFFFFFFF
Picture1.Circle (maxx_8, maxy_8), dotsize * 3, &HFFFFFFF

End Sub

Sub Win1clear()
If PLN > 39 Then
If Check3.Value = 0 Then Picture3.Cls: PLN = 0
End If

End Sub

Private Sub Text28_Change()
xoffset8 = Val(Text28.Text)
RedoFlag = True
End Sub

Private Sub Text1_Change()
xoffset7 = Val(Text1.Text)
RedoFlag = True
End Sub
```

Fujita, Parrenin et al.: Volcanic synchronization of Dome Fuji and Dome C Antarctic deep ice cores over the past 216 kyr

```
Private Sub Text21_Change()  
xoffset6 = Val(Text21.Text)  
RedoFlag = True  
End Sub
```

```
Private Sub Text22_Change()  
xoffset5 = Val(Text22.Text)  
RedoFlag = True  
End Sub
```

```
Private Sub Text24_Change()  
xoffset4 = Val(Text24.Text)  
RedoFlag = True  
End Sub
```

```
Private Sub Text25_Change()  
xoffset3 = Val(Text25.Text)  
RedoFlag = True  
End Sub
```

```
Private Sub Text8_Change()  
RedoFlag = True  
End Sub
```