GPR_SAMP

Version: 1.08.09.01
Last revision date: 8-9-2001
Description: ; GPR_SAMP re-samples GPR data. The number of traces and/or samples can be reduced and the number of samples per trace can be increased or decreased. The following computer storage formats are recognized: GSSI SIR-10A version 3.x to 5.x, Sensors and Software pulseEKKO, and SEG-Y. If the storage format does not conform to any of the above or this program is having trouble reading the file correctly, there are options in the CMD file for the user to specify required parameters.

After the data have been manipulated, they are stored to disk in the same format as they were read in. An exception is made for early-version GSSI DZT files that had 512-byte headers. A current (version 5.x) 1024-byte header is written into the output file. File extensions are forced to DZT for GSSI files and to DT1 for pulseEKKO files.

To convert from one GPR storage format to another, use program GPR_CONV.EXE. Because some storage formats maintain information about the data in each trace header, both the trace header and trace data block are stored as a unit as one column in the program's internal storage grid. If the size of the trace header is not an even multiple of the data element (sample) size, the program will stop and report an error. This should only be a problem with some user-defined storage formats.

The input to this program is a "CMD" file, an ASCII text file containing keywords (or parameters) describing how to re-sample the radar data. The CMD file specifies the data file name (and optionally the storage format). Inspect the example file GPR_SAMP.CMD for usage.

NOTES:

- Only 1 to 4 headers are supported in DZT files.
- There is no graphic display of the data.
- To change storage formats use GPR_CONV.EXE.
- To process the data use GPR_PROC.EXE.
- To display the processed data, use programs such as GPR_DISP.EXE or FIELDVEW.EXE.

THE KEYWORDS

Following is the list of keywords and their default values. The documentation format is:
"KEYWORD: **keyword** = default value".
Look at GPR_SAMP.CMD as an example command file with correct usage and default keyword values. The file GPR_SAMP.CMD has most comments stripped out, and GPR_SAMP.CM_ has all comments removed.

*************************** PROGRAM CONTROL *******************************

KEYWORD: **batch** = "FALSE"
Place the program in batch mode (no pauses) if "TRUE". The program will normally pause at times before ending.

KEYWORD: **debug** = "FALSE"
Place the program in debug mode if "TRUE" (for developers)

******************** SPECIFICATION OF INPUT DATA ************************

The storage format is determined by inspecting the file. If the program cannot recognize a flavor of the three formats below then an error message may be issued.

Recognized storage formats are:

DZT	GSSI SIR-10A file with embedded (512- or 1024-byte) info header
DT1	Sensors & Software pulseEKKO file with matching HD info file
SGY	SEG SEG-Y format

DT1 and HD files are assumed paired, i.e. both have same filename with different extensions. So, if a data file with a ".DT1" extension is specified, the ".HD" filename will be assumed. Only DT1/HD files must have those filename extensions.

KEYWORD: **dat_infilename** = ""
This is the input GPR binary data file name

KEYWORD: **channel** = 0
This keyword is for use with multiple channel DZT files only. It is the channel number in multi-channel data sets and is indexed from 0. GSSI data can have up to 4 channels (channel = 0, 1, 2, or 3).

KEYWORD: **first_trace** = -1
This is the first trace to use from the file. Traces are indexed from 0 (that is the first trace in the file is trace 0).

KEYWORD: **last_trace** = -1
This is the last trace to use from the file. Traces are indexed from 0 (that is the first trace in the file is trace 0).

RAMAC and user-defined data files can be read by assigning correct values to the next five keywords.

############ NOTE #############
IF the GPR format DOES NOT CONFORM to any of the above formats then the next six parameters (other_format, file_header_bytes, trace_header_bytes, samples_per_trace, total_time, and input_datatype) MUST be specified. Otherwise, IGNORE THEM. If you want to convert the storage format then GPR_CONV is the program to use. GPR_INFO will report this basic information for recognized storage formats.
################################

KEYWORD: **file_header_bytes** = 0
Replace with number of bytes in the file header. PulseEKKO data files do not have a file header - the information is held in another file with a .HD extension. GSSI files have either a 512-byte (old style) or 1024-byte (current style) header. However, DZT files can have up to 4 file headers - one for each channel. SEG-Y files have a 3600-byte header. RAMAC data files have no file header.

KEYWORD: **trace_header_bytes** = 0
Replace with number of bytes in each trace header. For pulseEKKO files, a 128-byte header precedes each GPR trace. For GSSI files, no header precedes each trace, but the first 2 samples (not necessarily bytes) are reserved. SEG-Y files have a 240-byte trace header. RAMAC data files have no trace headers.

KEYWORD: **samples_per_trace** = 0
Replace with the number of samples per trace. For pulseEKKO data, the number of samples per trace is recorded in the HD file (NUMBER OF PTS/TRC). For GSSI data, the number of samples per trace is a power of 2, from 128 to 2048, typically 256, 512, or 1024. The information is recorded in the DZT file

header in the rh_nsamp field. For RAMAC files, the RAD text file records the number of samples. For SEG-Y files, look in the comment area of the file header.

KEYWORD: **total_time** = 0
Replace with total number of nanoseconds per trace. For pulseEKKO data, look at the "TOTAL TIME WINDOW" field in the .HD file. For GSSI data the value is recorded in the file header. For SEG-Y files, look in the comment area of the file header. For RAMAC files, the TIMEWINDOW parameter records the time per trace in microseconds (multiply by 1000 to get ns).

KEYWORD: **input_datatype** = 0
This defines the type of input data element. Replace with one of the following element types:

```
     1 for 1-byte signed characters
    -1 for 1-byte unsigned characters            (GSSI)
     2 for 2-byte signed short integers          (pulseEKKO, RAMAC, SEG-Y)
    -2 for 2-byte unsigned short integers        (GSSI)
    -5 for 2-byte unsigned short integers, but only first 12-bits used
     3 for 4-byte signed long integers           (SEG-Y)
    -3 for 4-byte unsigned long integers
    -6 for 4-byte unsigned long integers, but only first 24-bits used
     4 for 4-byte floats                         (SEG-Y)
     8 for 8-byte doubles
```

For example: 8-bit GSSI data are unsigned characters (values from 0 to 255), use -1 for input_datatype. Use –2 for 16-bit GSSI data (values from 0 to 65535). PulseEKKO and RAMAC data are typically 16-bit signed integers (values from -32768 to 32767), use 2 for input_datatype. For SEG-Y data, the input_datatype can be 2 (signed short integers), 3 (signed long integers), or 4 (4-byte floating point reals). Data types are stored in the file header of DZT and SGY files. PulseEKKO and RAMAC do not record the data type.

********************* SPECIFICATION OF OUTPUT FILE ************************
The processed GPR data are stored in the SAME FORMAT as the input data. Use GPR_CONV.EXE to convert between storage formats.

KEYWORD: **dat_outfilename** = ""
This is the name of the binary GPR data file that is written to disk. For S&S files, an "HD" file will also be created. Note that output files are single-channel only.

**************************** RESAMPLING OPTIONS ****************************
This group determines how data are resampled;

If "first_samp" or "last_samp" are not specified or are both less than 0, then all trace samples will be used. If all trace samples are not used, i.e. first_samp or last_samp are greater than 0, then the total time of the trace will be reduced ONLY for a non-GSSI DZT file and a low-cut filter will be used. The sample rate is not changed. If you want to "expand" the new piece of the trace out to the original length, then assign "resample" the appropriate value (that is, the number of original samples) and then the sample rate will change (and a high-cut filter will be used based on the largest sampling rate encountered).

KEYWORD: **first_samp** = -1
Start processing at this sample number (inclusive).
;

KEYWORD: **last_samp** = -1
Stop processing at this sample number (inclusive).
;

KEYWORD: **resample** = 0
If assigned a value greater than or equal to 3, then this is the new number of samples for each trace. The minimum value is 3 (for splining) and the maximum is 65536. A cubic spline is used for the interpolation. FFT filtering is applied to low-pass filter if you are reducing or increasing the number of samples. The total time (which determines the lowest frequency in the trace) does not change. The highest frequency in the trace is determined by the sample rate, which is determined from the number of samples. Hence, a high-cut filter is always applied to avoid aliasing (when reducing samples) or to avoid introducing frequencies not present in the original trace (when increasing samples).

NOTE: For DZT files, "resample" will be forced to the next higher power of 2 (if not already a power of 2) such as 128, 256, 512, 1024, 2048, 4096, etc.

############ NOTE #############
The re-sampling operation DOES NOT change the range gain records in DZT headers (or in the comment area of other storage types). To keep the DZT header information correct, first remove range gain from the DZT files using GPR_PROC, run this program, then using GPR_PROC add gain back to the data (it can be the same or different). To calculate new dB values, linear interpolation can be used between the old gain dB values. Use program S10_EDHR.EXE to change the range gain stored in the DZT file header.
################################

KEYWORD: **preprocFFT** = "TRUE"
If set to "TRUE" then the scans will be pre-processed. This operation tapers the start and end of each trace to the median value. A Hanning cosine taper is always applied to edges of the filter pass block.

############ NOTE #############
If the total time per trace has changed (that is, if "first_samp" or "last_samp" are not the default values) or the sample rate has changed (if "resample" is greater than or equal to 3), then the data will be FFT filtered.
################################

Usage: `GPR_SAMP cmd_filename`
Required command line arguments:
cmd_filename - The name of the keyword file.
Optional command line arguments (do not include brackets): none
Examples:

```
gpr_samp sfile1.cmd
```