

## BIOSIG History File Structure

### **Introduction**

This document describes the structure of data files that are created by TDT's BioSig program. You will need the following source files to successfully read and manipulate BioSig history files (.ARF)

<i>log.h</i>	.ARF data file structures
<i>doslog.h</i>	Dos version of log.h

The following C modules contain routines that read to and write from BioSig data files. *log.cpp* was written for Microsoft Visual C++ and may not be compatible with your compiler.

<i>log.cpp</i>	MSVC++ routines to read and write .ARF files
<i>doslog.cpp</i>	DOS version of log.cpp

IMPORTANT: You must use `open()` and `read()` rather than `fopen()` and `fread()` for file indexes to work correctly.

### **Format of .ARF Files**

#### Overview of structure:

At the top of the file there will be a group structure, which has information about this group, such as number of records in this group, subject, Siggen file, units, sampling rate, and other information. Several records will follow this group structure. Each record has a header as defined in `RecType`, which is the record information. After the header, there will be the data. In the record header the number of data points is specified. After the first group, there may be several other similar groups. The number of groups are specified in the file header.

#### Typical .arf file structure:

```

File header (RecHeadType)
Group 1 (GroupType)
  Record 1 (RecType) record data
  Record 2 (RecType) record data
  Record 3      record data
  ... ..
  Record n      record data
Group 2
  Record 1      record data
  ... ..
Group 3
  ... ..
Group m
    
```

**Data Structures**

```

#define GROUP_VER2      2
#define GROUP_VER3      3
#define GROUP_VER4      4
#define GROUP_VER10     10

#define CURRECTYPE      1
#define MAXRECS         2000
#define MAXGRPS         200

#define MAXCURSORS      10

struct RecHeadType
{
    int ftype;
    int ngrps;
    int nrecs;
    long grpseek[MAXGRPS];
    long recseek[MAXRECS];
    FILE *fhand;
};

struct SubInfoType
{
    char id[16];
    char ref[2][16];
    char memo[50];
};

struct GroupType
{
    int grpn;
    int frecn;
    int nrecs;
    SubInfoType SI;
    time_t beg_t;
    time_t end_t;
    char sgfname[2][100];
    char vname[MAXVARS][15];
    char units[MAXVARS][5];
    float srate;
    // char logfname[100];
    long cc_t;
    int version;
    long postproc;
    char dump[92];
};

struct CursorType
{
    float tmark;    //tmark =-1 if no cursors set. Important to set this
                  flag

```

```
float val;
char desc[20];
int xpos,ypos;
int hide;
int lock;
};

#define TF_TIME 0
#define TF_FREQ 1
#define TF_PHASE 2

struct RecType
{
    int recn;
    int grp_id;
    time_t grp_t;
    int newgrp;
    int sgi;
    char chan;
    char rtype;

    int npts;
    float osdel;
    float dur;
    float srate;

    float arthresh;
    float gain;
    int accouple;

    int navgs;
    int narts;

    time_t beg_t;
    time_t end_t;

    float vv[MAXVARS];

    CursorType CUR[MAXCURSORS];
};
```