BRAIN QUICK SOFTWARE EXCHANGE EEG TRACE FILE STRUCTURE DESCRIPTION

Version: 1.00
Dated: 18/07/2017

Via Giotto 2 - 31021
Mogliano Veneto (TV) - Italy
tel. +39.041.5937000
fax. +39.041.5937011
e-mail micromed@micromed.eu
web: www.micromed.eu

WARNING

This document is strictly confidential and must not be distributed without permission. Contact Micromed to have permissions or any other information.

INTRODUCTION

This is a description of the Micromed Exchange EEG Trace file that is files generated by Brain Quick Software for exchange purposes.

Any Micromed Exchange EEG file contains a data file structure that can be easily interpreted. Data file is broken down into two main sections: the ***header, which*** contains the patient and setup data and a variable length ***trailer*** that contains the digitised trace data. The sequence of this data can be interpreted with the header block. Files have the capability to store up to **256** individual channels of signal data, the number stored in each file may vary and this information may be found in the header. To find the file of the relative trace that you are examining, use the "Properties" function provided by the System 98 software.

FILE STRUCTURE

Offset	Label	Data Type	Description
0	**FileDescriptor**	char[32]	Micromed Descriptive String
32 (0x20)	**ID**	unsigned char[16]	Guid of file
48 (0x30)	**Version**	UInt16	Version number
50 (0x32)	**PatientData**	Byte[69]	Data of Patient
119 (0x77)	**RecordingDateTime**	Int64	Recording DateTime in ticks
127 (0x7F)	**SampleLength**	Uint16	Float / double
129 (0x81)	**DerivationNumber**	Uint16	Number of Derivation
131 (0x83)	**SamplingFrequency**	Uint16	Sampling Frequency in Hz
133 (0x85)	**Derivations**	Byte[20 * MAX_DERIVATION]	Derivation struct array
20613 (0x5085)	**Data**	Byte[]	Interlaced derivation data

MAX_DERIVATION ***1024***

The following is the structure of the data file. The first column contains the effective address in the file.

0 FILE_DESCRIPTION

This descriptive string specifies that the file is a "MICROMED_EXCHANGE_TRACE". The number of possible ASCII characters is 30, with a 0x00,0x1A used as terminators of the string.

32 ID

This Guid is unique identifier of analyser external data file.

48 Version

Version number of file format. Value = 1

50 PATIENT DATA (Fixed Size of 69)

This is data pertaining to patient identification. The structure of this record is the following:

Offset	Label	Data Type	Description
50 (0x32)	**Surname**	char[22]	Surname, without any terminator
72 (0x48)	**Name**	char[20]	First Name, without any terminator
92 (0x5C)	**DOB**	Int64	Birthdate in ticks
100 (0x64)	**Reserved**	unsigned char[19]	Internally used by the Brain Quick Software. They are not meaningful for the user and can't be used.

119 RecordingDateTime

DateTime of the recording in ticks. (see Appendix A)

127 SampleLength

This represent the number of bytes of each sample. value = 4 means float, value = 8 means double

129 DerivationNumber

This represent the number of Derivation

131 SamplingFrequency

Sampling frequency in Hz. It's common to all derivations

133 Derivations (Fixed Size of 20)

This is data pertaining to derivations up to MAX_DERIVATION. The structure of this record is the following:

Offset	Label	Data Type	Description
+0 (0x00)	**Label**	char[16]	Label of Derivation (Inverting Label – Non Inverting Label)
+16 (0x10)	**InvertingID**	UInt16	ID of the Inverting Electrode
+18 (0x12)	**NonInvertingID**	UInt16	ID of the Non-Inverting Electrode

20613 Data

Contains the data of the trace file. The data are interlaced and the unit of measure is microvolt

APPENDIX A

DateTime is serialized to Int64 in ticks.
Ticks represents the number of 100-nanosecond intervals that have elapsed since 12:00:00 midnight, January 1, 0001, which represents DateTime.MinValue. A single tick represents one hundred nanoseconds or one ten-millionth of a second. There are 10,000 ticks in a millisecond.

```
// Convert C# DateTime to C++ date time
{
    // 624376461050000000 (dec)  07/28/1979 10:35:05 PM
    unsigned __int64 DTFromCDull = 0x8AA3B154F003280;
   // Thursday, January 01, 1970 12:00:00 AM
   unsigned __int64 UnixEpoch = 0x089f7ff5f7b58000;

   time_t t;
   struct tm *tm;

   t=(time_t)((DTFromCDull-UnixEpoch)/10000000);
   tm=gmtime(&t);
   printf("%s \n",asctime(tm));
}
```

References:
http://www.morgantechspace.com/2013/08/convert-datetime-to-ticks-and-ticks-to.html
https://www.codeproject.com/Tips/694874/Exchange-Date-Time-between-Native-Cplusplus-and-Cs