

Gigaset M740 AV

Recording File Formats

1. General data formats:

All files are stored in big endian format.

(**long long** is signed 64 bit, **int** is 32 bit, **short** is 16 bit and **char** is 8 bit)

2. Transport streams are recorded in .fmpg .mpg .midx and .idx files

For one recording there are the following files:

```
test.fmpg                // main index file, references all test.fmpg.*.mpg
test.fmpg.000.mpg       // transport stream 0, 10 minutes long
test.fmpg.000.mpg.midx  // meta index (index for index)
test.fmpg.000.mpg.idx   // index
test.fmpg.001.mpg       // transport stream 1
test.fmpg.001.mpg.midx
test.fmpg.001.mpg.idx
test.fmpg.002.mpg       // transport stream 2
test.fmpg.002.mpg.midx
test.fmpg.002.mpg.idx
.....
```

3. Structure of the .fmpg file

The .fmpg file is the **main file** for one recording and it consists of the following 256 byte blocks:

```
typedef struct {
    long long timestamp;        // one second is 90000
    short pcr_pid,vpid;        // pcr pid, video pid
    short pmt_pid,apid;        // pmt pid, audio pid
    char filename[256-16];      // for example test.fmpg.000.mpg
} FragmentIndexEntry;
```

4. Structure of the .idx file

The .idx file is used to support the trick modes **fast forward** and **fast reverse**. It is used for quickly finding the iframes and consists of the following 24 byte blocks:

```
/* Entry of StreamIndex */
typedef struct {
    long long timestamp;    /* 90 khz timestamp */
    long long offset;      /* byte-offset of I-Frame in transport stream */
    int size;              // size of iframe
    int flags;             // not used
} IndexEntry;
```

5. Structure of the .midx file

The .midx file is for **quickly jumping** to an arbitrary time position in the stream and it consists of the following 24 byte blocks:

```
/* Entry of MetaIndex */
typedef struct {
    long long timestamp;    /* 90 khz timestamp */
    long long offset;      /* byte-offset in stream */
    int ioffset;           /* byte-offset in .idx file */
    int pad;               // unused
} MetaIndexEntry;
```

6. Structure of the .crid file

Structure of the .crid file:

```
long CRID-Version;          /* always 2 */
long long CRID-ID;         /* unique resource identifier */
long Recording-State;      /* 1 scheduled for recording */
                           /* 2 is currently recorded */
                           /* 3 was recorded o.k. */
                           /* 4 recording damaged */
long epg start time;       /* UTC in sec. since 1970/01/01 */
long epg end time;        /* UTC in sec. since 1970/01/01 */
long user access data;     /* not used, always 0 */
long recording pre-offset; /* not used, always -1 */
```

long recording post-offset;	<i>/* not used, always -1 */</i>
long recording type;	<i>/* 1, single epg recording */</i>
	<i>/* 2, single timer recording */</i>
	<i>/* 4, series epg recording */</i>
	<i>/* 8, series timer recording */</i>
long series ID;	<i>/* unique series identifier */</i>
short protected flag;	<i>/* 0, unprotected */</i>
	<i>/* 1, protected */</i>
long length of crid title;	
char[] crid title;	<i>/* string of char of crid title */</i>
long number of recorded pieces;	<i>/* (normally 1, more when hard- */</i>
	<i>/* powered off and restarted) */</i>
long length of recording control file;	<i>/* (fmpg) name */</i>
char[] name of recording control file;	<i>/* string of char of fmpg name */</i>
long absolute start time of recording;	<i>/* possibly start of timeshift ! */</i>
long long start-timestamp on recording in 90 kHz;	<i>/* 0, bigger than 0 when recorded */</i>
	<i>/* from timeshift, or previous */</i>
	<i>/* recording on same channel */</i>
long long end-timestamp on recording in 90 kHz;	<i>/* 0 when recording was killed */</i>
	<i>/* (hard-powered off) or when */</i>
	<i>/* recording is still running */</i>
	<i>/* >0, when recording was finished */</i>
long length of epg short text;	
char[] epg short text;	<i>/* string of char of epg short text */</i>
long length of epg long text;	
char[] epg long text;	<i>/* string of char of epg long text */</i>
long playback timestamp;	<i>/* position on recording in seconds */</i>
	<i>/* from begin of recording where last */</i>
	<i>/* playback was stopped. */</i>
	<i>/* 0 when never played back */</i>