GCF Reference

Güralp Compressed Format

Specification

Part No. GCF-RFC-GCFR

Designed and manufactured by
Güralp Systems Limited
Midas House, Calleva Park
Aldermaston RG7 8EA
England

Issue C 2011-01-05

Table of Contents

1 Introduction

GCF is a block-based format for storing seismic data. Each block is 1024 bytes long, including a 16-byte header, and is completely self contained. There are currently six distinct types of block: data (containing sample differences), textual status, CD status (a deprecated binary status format), unified status, strong motion results and byte pipe.

The 16 byte header contains a system ID and a stream ID, which, taken together, generally identify the instrument producing the data and the channel with which it is associated. It also contains a start time in seconds (possibly with a fractional component), a sample rate (0 if not seismic data), a number of samples (or bytes), the format of the samples and some further ancillary information.

Connected with the GCF format (although not directly part of it) are various transport layers: BRP (block recovery protocol, for use over serial links; see SWA-RFC-BRPR[1]) and Scream! UDP and TCP (for use over IP networks; see SWA-RFC-SCRM[2]).

1.1 Byte order

Unless otherwise stated, multi-byte integers are stored in big-endian (most significant byte first) format.

1.2 GCF Time

The time-stamp for the packet is always in UTC. It is given as two fields: the number of days elapsed since 1989-11-17, the first day of operation of the GCF format, and the number of seconds elapsed since midnight (00:00:00Z). This allows it to correctly represent negative and positive leap seconds (a value of 86400 in the seconds field means 23:59:60Z, the clock time of a positive leap second). The time-stamp refers to the start of the data in the packet (i.e. it is the time-stamp of the first sample); it is left up to applications to compute the end time.

This format for time is used throughout the GCF standard and in many programs dealing with it; it is referred to as 'GCF time'.

1.3 Base 36 IDs

Take the integer read from the ID field in the header and consider it as a base 36 number (i.e. each digit spans the decimal range 0_{10} to 35_{10}).

1 http://www.guralp.com/documents/SWA-RFC-BRPR.pdf
2 http://www.guralp.com/documents/SWA-RFC-SCRM.pdf

The base 36 digits are mapped to ASCII characters as follows: 0_{10} →0, 9_{10} →9, 10_{10} →A, 35_{10} →Z. Alternatively stated, the base 36 digit may be used as a (zero-referenced) character position offset into the string "`0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ`".

Having ascertained the character associated with the least significant digit, the integer can be divided by 36 (with the fractional remainder discarded) and the next least significant digit used to determine another character of the ID. Once the integer itself hits zero, there are no more characters to decode (i.e. you can never have an ID that starts with the character 0).

2 Common Header Fields

Each GCF packet has a 16-byte header. The meaning of fields in the header depends on the type of block, but there is a lot of commonality.

For any type of block, the system ID and stream ID are determined in the same manner. The stream ID is required to further differentiate between certain types of block. The sample rate, date, compression and number of samples are also determined in a similar manner for different types of block, although the details vary slightly.

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td>Ex</td><td colspan="31">System ID</td></tr>
<tr><td>0</td><td colspan="31">Stream ID</td></tr>
<tr><td colspan="15">Timestamp (days)</td><td colspan="17">Timestamp (seconds)</td></tr>
<tr><td colspan="8">TTL</td><td colspan="8">Sample rate</td><td colspan="8">Compression</td><td colspan="8">N° of records</td></tr>
</table>

"`Ex`" is the extended sysid flag. Bit 31 of the "`Stream ID`" word is reserved and must be set to 0. The "`N° of records`" field states the number of 4-byte data records: this can be multiplied by the compression value to give the number of samples in the packet.

2.1 No extended sysid

Test:

- '`Ex`' (the extended sysid flag, bit 31 of the `sysid` word) is clear

There is no information in the `sysid` word other than the ID itself. Decoding the base 36 IDs is explained in section 1.3 on page 3. The type of instrument that produces this sort of block cannot be determined from the GCF block.

2.2 Extended sysid

Test:

- '`Ex`' (the extended sysid flag, bit 31 of the `sysid` word) is set

If the extended sysid flag is set, then some of the `sysid` word is used to represent system type and gain information. A further "double-extended" flag is available for future use. A normal extended sysid comprises the following fields:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	Gain			Type	*Sysid*																									

Bits 29-27	Gain setting
000	No gain front end fitted
001	x1
010	x2
011	x4
100	x8
101	x16
110	x32
111	x64

Table 2.1: Interpretation of variable gain bits

Decoding the base 36 IDs is explained in section 1.3 on page 3. The 6 most significant bits must be masked off, giving only 5 characters.

A future expansion "double-extended" GCF header would be decoded as follows (with the 11 most significant bits masked off before decoding the ID, giving only 4 characters):

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	Gain			Type	Future use					Sysid																				

2.2.1 Instrument type

The instrument type is a CMG-DM24mk3 if the "`Type`" bit is 0 or a CMG-CD24 if the "`Type`" bit is 1[1]. Given the instrument type, it is possible to determine (for data packets) the sequence of decimation filters through which the sampled analogue data has passed. This

1 Early versions of the CMG-CD24 firmware set the extended sysid bit but left the 'Type' bit as 0, so the digitiser would be misidentified as a CMG-DM24mk3. This was fixed in release 273 of the CMG-CD24 firmware.

information is contained in the tap table lookup byte, which provides an index into a per-instrument table of filter sequences (see section 3.1 on page 8).

2.2.2 Gain

Some digitiser units are equipped with a variable gain front end. Bits 29–27 of the sysid word identify whether a gain front end is fitted and, if so, what its current gain value is. Table 2.1 can be used to interpret these values.

3 Decoding Blocks

With the common header details determined, the next step is to categorise the block.

3.1 Data block

Test:

- Sample rate (byte 13) is not equal to zero

Data blocks contain raw waveform data, encoded as integer differences. The data samples follow immediately after the header. The value of the sample rate field (byte 13) generally indicates the number of samples per second. However, as it is an unsigned 8-bit integer, it is not possible to directly represent sample rates higher than 255sps or lower than 1sps. Therefore, some unlikely values have been used to represent such sample rates. These are shown in table 3.1.

When a sample rate higher than 250sps is in use, it is possible that a block will need to be started at a time that does not lie on an integral second boundary. To represent such times, a time fractional offset (in the top 4 bits of byte 14, the compression code) is used. The offset is the numerator of a fraction whose denominator is given in table 3.1.

This gives a fraction between 0–1 which is the offset (in seconds) from the start time given in bytes 8–11.

Header value (byte 13)	Sample rate	Time fractional offset denominator
157	0.1	unused
161	0.125	unused
162	0.2	unused
164	0.25	unused
167	0.5	unused
171	400	8
174	500	2
176	1000	4
179	2000	8
181	4000	16

Table 3.1: Special sample rates

Format field (byte 14)	Description	Number of samples
1	Signed 32-bit differences	1×Nº of records(byte 15)
2	Signed 16-bit differences	2×Nº of records(byte 15)
4	Signed 8-bit differences	4×Nº of records(byte 15)

Table 3.2: Difference formats and number of samples.

Sample data immediately follows the GCF header. In order to efficiently represent seismic waveforms, the data is not presented as raw samples but as one initial (absolute) value followed by a sequence of signed differences. The number of samples in a packet is computed from the compression format (byte 14) and another value in the header (byte 15), as shown in table 3.2.

In order to decode the raw samples, the first 4 bytes of the data are read and interpreted as a signed 32-bit integer, representing the absolute value of the first sample in the block (whose time will be given in the header). This value is the "Forward Integration Constant" or FIC.

Following the FIC are $n-1$ signed differences (where n is the number of samples in the block, computed from table 3.2). The absolute value at each time step is computed simply by accumulating the differences with the FIC.

As a final check, the block contains the absolute value of the last sample (the "Reverse Integration Constant", or RIC). The RIC is a signed 32-bit integer stored immediately after the last difference. The value in the accumulator should match the RIC. If it does not, a transmission error has occurred.

3.1.1 TTL values for CMG-DM24mk2

Test:

- '`Ex`' (the extended sysid flag, bit 31 of the `sysid` word) is clear
- TTL (byte 12) is not equal to zero

The TTL value will be between 17–59 inclusive. The TTL specifies the sequence of decimation filters through which data passes to get to the lowest tap; the sample rate can be used to determine how many of these decimation filters were applied. Consult SWA-D24-2D00[1] for more details.

1 http://www.guralp.com/documents/SWA-D24-2D00.pdf

3.1.2 TTL values for CMG-DM24mk3

Test:

- '`Ex`' (the extended sysid flag, bit 31 of the `sysid` word) is set
- Digitiser type (bit 26 of sysid word) is DM24mk3
- TTL (byte 12) is not equal to zero

The TTL value will be between 1–95 inclusive. The TTL specifies the sequence of decimation filters through which data passes to get to the lowest tap; the sample rate can be used to determine how many of these decimation filters were applied. Consult SWA-D24-3D00[1] for more details.

3.1.3 TTL values for CMG-CD24 (and CMG-EDU)

Test:

- '`Ex`' (the extended sysid flag, bit 31 of the `sysid` word) is set
- Digitiser type (bit 26 of sysid word) is CD24
- TTL (byte 12) is not equal to zero

The TTL value will be between 1–226 inclusive. The TTL specifies the sequence of decimation filters through which data passes to get to the lowest tap; the sample rate can be used to determine how many of these decimation filters were applied. Consult SWA-C24-D000[2] for more details.

3.2 Status block

Test:

- Sample rate (byte 13) is equal to zero
- Compression code (byte 14) is 4
- The rightmost two characters of the stream ID are "00" (i.e. the value of `strid`, modulo 1296, is zero)

A status block contains information such as state of health in ASCII (although there are various control codes and other nasties that may

1 http://www.guralp.com/documents/SWA-D24-3D00.pdf
2 http://www.guralp.com/documents/ SWA-C24-D000.pdf

occur, so the data must be sanitised before displaying on a terminal etc.). The status information occurs immediately after the header.

The length of the status text, in bytes, is determined by multiplying the unsigned 8 bit value of byte 15 (number of samples) by 4.

3.3 Unified status block

Test:

- Sample rate (byte 13) is equal to zero
- Compression code (byte 14) is 4
- The rightmost two characters of the stream ID are “01” (i.e. the value of `strid`, modulo 1296, is one)

A unified status block is a binary, machine-readable block of status information that is emitted every second by a digitiser with this mode enabled. The length of the data, in bytes, is determined by multiplying the unsigned 8 bit value of byte 15 (number of samples) by 4.

See the document SWA-RFC-UNIS[1] for the full specification of unified status blocks.

3.4 Strong motion block

Test:

- Sample rate (byte 13) is 0
- Compression code (byte 14) is 4
- The rightmost two characters of the stream ID are “SM” (i.e. the value of `strid`, modulo 1296, is 1030).

A strong motion block contains the results of strong motion calculations, computed every second by a digitiser with this mode enabled. The length of the data, in bytes, is determined by multiplying the unsigned 8 bit value of byte 15 (number of samples) by 4.

See the document SWA-RFC-STMN[2] for the full specification of unified status blocks.

1 http://www.guralp.com/documents/ SWA-RFC-UNIS.pdf
2 http://www.guralp.com/documents/ SWA-RFC-STMN.pdf

3.5 CD status block (deprecated)

Test:

- Sample rate (byte 13) is zero
- The rightmost two characters of the stream ID are "CD" (i.e. the value of `strid`, modulo 1296, is 445).

This format is deprecated. Use unified status instead. When CD status packets are enabled, the digitiser will regularly emit one packet a second.

3.6 Byte pipe

Test:

- Sample rate (byte 13) is zero
- Compression code (byte 14) is 4
- The rightmost two characters of the stream ID are "BP" (i.e. the value of `strid`, modulo 1296, is 421).

Some digitisers have a 'byte pipe' mode, used to package arbitrary binary data read from an auxiliary serial port into a GCF packet for transmission to a host. The length of the packaged data, in bytes, is given by multiplying the number of samples field (byte 15) by 4. The data is transmitted raw, with no encoding.

4 Revision history

2007-09-28	A	New document
2009-09-22	B	Update formatting to Güralp standard. Document variable gain header fields. Remove detailed CD status documentation.
2011-01-03	C	Clarification of "Nº of records"