BCF2 Quick Reference (r198)

In BCF2, each key in the FILTER, INFO and FORMAT fields is required to be defined in the VCF header. For each record, a key is stored as an integer which is the index of its first appearance in the header. '`PASS`' is always indexed at 0, which is special cased as VCF does not require the presence of this word.

In BCF2, a typed value consists of a typing byte and the actual value with type mandated by the typing byte. In the typing byte, the lowest four bits give the atomic type. If the number represented by the higher 4 bits is smaller than 15, it is the size of the following vector; if the number equals 15, the following typed integer is the array size. The highest 4 bits of a Flag type equals 0 and in this case, no assumptions can be made about the lower 4 bits. The table below gives the atomic types and their missing values:

Bit 0–3	C type	Missing value	Description
1	`int8_t`	`0x80`	signed 8-bit integer
2	`int16_t`	`0x8000`	signed 16-bit integer
3	`int32_t`	`0x80000000`	signed 32-bit integer
5	`float`	`0x7F800001`	IEEE 32-bit floating pointer number
7	`char`	'`\0`'	character

A genotype (GT) is encoded as an integer vector with each integer describing an allele and its phase w.r.t. the previous allele. The first allele does not carry the phase information. In the vector, each integer is organized as '`(allele+1)<<1|phased`' where `allele` is set to -1 if the allele in GT is a dot '.' (thus the higher bits are all 0). The vector is padded with missing values if the GT having fewer ploidy.

A BCF2 file is BGZF compressed and all multi-byte value are little endian.

Field	**Description**	**Type**	**Value**
magic	BCF2 magic string	`char[5]`	`BCF\2\1`
l_text	Length of the header text, including any NULL padding	`uint32_t`	
text	NULL-terminated plain VCF header text	`char[l_text]`	
	List of VCF records (until the end of the BGZF section)		
l_shared	Data length from CHROM to the end of INFO	`uint32_t`	
l_indiv	Data length of FORMAT and individual genotype fields	`uint32_t`	
CHROM	Reference sequence ID	`int32_t`	
POS	0-based leftmost coordinate	`int32_t`	
rlen	Length of reference sequence	`int32_t`	
QUAL	Variant quality; `0x7F800001` for a missing value	`float`	
n_allele_info	`n_allele<<16|n_info`	`uint32_t`	
n_fmt_sample	`n_fmt<<24|n_sample`	`uint32_t`	
ID	Variant identifier	`typed str`	
	List of alleles in the REF and ALT fields (n=n_allele)		
allele	A reference or alternate allele	`typed str`	
FILTER	List of filters; filters are defined in the dictionary	`typed vec`	
	List of key-value pairs in the INFO field (n=n_info)		
info_key	Info key, defined in the dictionary	`typed int`	
info_value	Value	`typed val`	
	List of FORMATs and sample information (n=n_fmt)		
fmt_key	Format key, defined in the dectionary	`typed int`	
fmt_type	Typing byte of each individual value, possibly followed by a typed int for the vector length	`uint8_t+`	
fmt_value	Array of values. The information of each individual is concatenated in the vector. Every value is of the same fmt_type. Variable-length vectors are padded with missing values; a string is stored as a vector of `char`.	(by fmt_type)	