

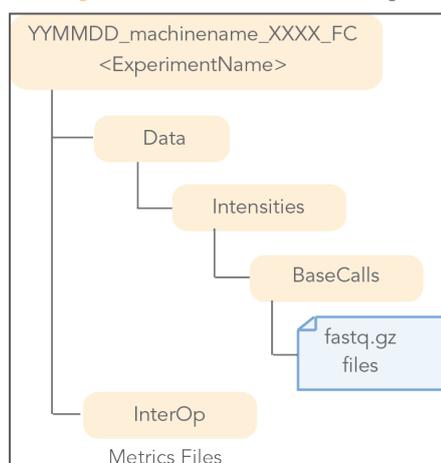
BCL Conversion Output Folder

The BCL Conversion output directory has the following characteristics:

- ▶ By default, the FASTQ output files are located in the Data/Intensities/BaseCalls directory; names of the files start with the sample name as derived from the sample sheet.
- ▶ The Undetermined files contain the reads with an unresolved or erroneous index.
- ▶ If no sample sheet exists, the software generates an Undetermined_S0 FASTQ file for each lane and read number.

For each sample, there is one FASTQ file per lane per read number (if reads exist for that sample, lane, and read number).

Figure 5 BCL Conversion Output



FASTQ Files

bcl2fastq2 Conversion Software v2.17 converts *.bcl, *.bcl.gz, and *.bcl.bgzf files into FASTQ files, which can be used as input for secondary analysis. The files are located in the Data/Intensities/BaseCalls directory. If no sample sheet is provided, the software generates one Undetermined_S0 FASTQ file for each lane and read number combination.

Naming

FASTQ files are named with the sample name and the sample number, which is a numeric assignment based on the order in the sample sheet that a sample ID first appeared in a given lane. Example:

Data/Intensities/BaseCalls/<SampleName>_S1_L001_R1_001.fastq.gz

- **<Sample_Name>**—The sample name provided in the sample sheet. If a sample name is not provided, the file name includes the sample ID, which is a required field in the sample sheet and must be unique within a lane.
- **S1**—The sample number based on the order in one lane that samples are first listed in the sample sheet starting with 1. In this example, S1 indicates that the sample ID is the first listed in the sample sheet. If this sample ID appears in another lane, the same sample number is used.



NOTE

Reads that cannot be assigned to any sample are written to a FASTQ file for sample number 0, and excluded from downstream analysis.

- **L001**—The lane number.
- **R1**—The read. In this example, R1 means Read 1. For a paired-end run, there is at least one file with R2 in the file name for Read 2. When generated, Index Reads are **I1** or **I2**.
- **001**—The last segment is always 001.

Compression

FASTQ files are saved in the compressed GNU zip format (an open source file compression program), indicated by the .gz file extension. By default, the BGZF variant of the GNU zip format is used. The BGZF variant facilitates parallel decompression of the FASTQ files by downstream applications. While BGZF is compliant with the GNU zip standard, if a downstream application cannot handle this variant, it can be turned off with the command line option `--no-bgzf-compression`.

Format

Each entry in a FASTQ file consists of 4 lines:

- ▶ Sequence identifier
- ▶ Sequence
- ▶ Quality score identifier line (consisting only of a +)
- ▶ Quality score

```
@<instrument>:<run number>:<flowcell ID>:<lane>:<tile>:<x-
  pos>:<y-pos>:<UMI> <read>:<is filtered>:<control
  number>:<index>
```

The following table describes the elements:

Element	Requirements	Description
@	@	Each sequence identifier line starts with @.
<instrument>	Characters allowed: a-z, A-Z, 0-9 and underscore	Instrument ID.
<run number>	Numerical	Run number on instrument.
<flowcell ID>	Characters allowed: a-z, A-Z, 0-9	
<lane>	Numerical	Lane number.
<tile>	Numerical	Tile number.
<x_pos>	Numerical	X coordinate of cluster.
<y_pos>	Numerical	Y coordinate of cluster.
<UMI>	Restricted characters: A/T/G/C/N	Optional, appears when UMI is specified in sample sheet. UMI sequences for Read 1 and Read 2, separated by a plus [+].
<read>	Numerical	Read number. 1 can be single read or Read 2 of paired-end.
<is filtered>	Y or N	Y if the read is filtered (did not pass), N otherwise.
<control number>	Numerical	0 when none of the control bits are on, otherwise it is an even number. On HiSeq X and NextSeq systems, control specification is not performed and this number is always 0.
<index>	Restricted characters: A/T/G/C/N	Index of the read.

An example of a valid entry is as follows; note the space preceding the read number element:

```
@SIM:1:FCX:1:15:6329:1045:GATTACT+GTCTTAAC 1:N:0:ATCCGA
TCGCACTCAACGCCCTGCATATGACAAGACAGAATC
+
<>;##=><9=AAAAAAAAAAA9#:<#<;<<<?????#=#
```

Control Values

If the read is not identified as a control, then the 10th column (<control number>) is 0. If the read is identified as a control, the number is greater than 0, and the value specifies what type of control it is. The value is the decimal representation of a bit-wise encoding scheme. In that scheme bit 0 has a decimal value of 1, bit 1 a value of 2, bit 2 a value of 4, and so on.

Quality Scores

A quality score (or Q-score) expresses an error probability. In particular, it serves as a convenient and compact way to communicate very small error probabilities.

Given an assertion, A, the quality score, Q(A), expresses the probability that A is not true, P(~A), according to the relationship:

$$Q(A) = -10 \log_{10}(P(\sim A))$$

where $P(\sim A)$ is the estimated probability of an assertion A being wrong.

The relationship between the quality score and error probability is demonstrated with the following table:

Quality score, Q (A)	Error probability, P (~A)
10	0.1
20	0.01
30	0.001



NOTE

On the systems we currently support, Q-scores are automatically binned. The specific binning applied depends on the current Q-table. See the white paper *Reducing Whole Genome Data Storage Footprint* for more information, available from www.illumina.com.

Quality Scores Encoding

In FASTQ files, quality scores are encoded into a compact form, which uses only 1 byte per quality value. In this encoding, the quality score is represented as the character with an ASCII code equal to its value + 33. The following table demonstrates the relationship between the encoding character, its ASCII code, and the quality score represented.



NOTE

When Q-score binning is in use, the subset of Q-scores applied by the bins is displayed.

Table 1 ASCII Characters Encoding Q-scores 0–40

Symbol	ASCII Code	Q-Score	Symbol	ASCII Code	Q-Score
!	33	0	6	54	21
"	34	1	7	55	22
#	35	2	8	56	23
\$	36	3	9	57	24
%	37	4	:	58	25
&	38	5	;	59	26
'	39	6	<	60	27
(40	7	=	61	28
)	41	8	>	62	29
*	42	9	?	63	30
+	43	10	@	64	31
,	44	11	A	65	32
-	45	12	B	66	33
.	46	13	C	67	34
/	47	14	D	68	35
0	48	15	E	69	36
1	49	16	F	70	37
2	50	17	G	71	38
3	51	18	H	72	39
4	52	19	I	73	40
5	53	20			

InterOp Files

The InterOp files can be found in the directory: <run directory>/InterOp. This directory contains binary files used by the Sequencing Analysis Viewer (SAV) software to summarize various analysis metrics such as cluster density, intensities, quality scores, and overall run quality.

The index metrics are stored in the file *IndexMetricsOut.bin*, which has the following binary format:

- ▶ Byte 0: file version (1)
- ▶ Bytes (variable length): record:
 - 2 bytes: lane number (uint16)
 - 2 bytes: tile number (uint16)
 - 2 bytes: read number (uint16)
 - 2 bytes: number of bytes Y for index name (uint16)
 - Y bytes: index name string (string in UTF8Encoding)