PacBio

Obc2fastq reference guide (v5.11)

Research use only. Not for use in diagnostic procedures.

P/N 103-086-100 Version 02 (August 2023)

Information in this document is subject to change without notice. PacBio assumes no responsibility for any errors or omissions in this document.

PacBio
1305 O’Brien Drive
Menlo Park, CA 94025
www.pacb.com

Obc2fastq reference guide (v5.11)

Introduction

The `obc2fastq` utility is a command-line software tool that converts OBC (Onso™ Base Call) files generated by PacBio® Onso sequencers into FASTQ files. The utility extracts read sequences and quality scores.

Optionally, `obc2fastq` can perform sample demultiplexing if a sample sheet is provided. The sample sheet file contains the mappings between individual samples and the index barcode sequences. See “Sample sheet” on page 3 for details.

The `obc2fastq` utility is typically used by bioinformaticians, data analysts, or any researchers who handle PacBio Onso sequencing data for downstream analysis.

Hardware and software requirements

Hardware requirements
• **PC with 8 cores and hyperthreading capability**
• **Minimum 64 GB RAM**
We recommend up to 8 threads per lane with a run time of 2.8 hours at 1.37 M spots x 336 cycles x 550 tiles and 16 SIDs per lane. At 16 threads per lane and 96 SIDs per lane, the RAM usage goes up to 121 GB and the run time is 2.37 hours.
Software requirements
• **Centos 7, Centos 8, Rocky 8, or Ubuntu operating systems**
• **gcc version 5.0 or greater, with C++17 support**

Installation

`obc2fastq` packages are available here and can be installed using standard Linux tools:

- **Centos 7 package**: Use the `yum` tool to install.
- **Centos 8 package**: Use the `dnf` tool to install.
- **Ubuntu package**: Use the `dpkg` tool to install.
- **Rocky 8 package**: Use the `dnf` tool to install.

Running the obc2fastq utility

Open a command-line window and enter the following command:

```
obc2fastq --input <run folder> \
--output <output folder name> \
--flowcellid <flow cell ID> \
--samplesheet <sampleSheet.csv file name> \
--designsheet <obc2fastq_params file name> \
--threadlanes <int> \
--threadsperlane <int> \
--controlsfile <control fasta file name> \
--barcodeallowedmismatches <int> \
```

where:

- `<run folder>` is the path to the sequencing run output. **(Required)**
- `<output folder name>` is the folder name where output files will be written. **(Required)**
  Note: If this folder does **not** exist, the software will create it.
- `<flow cell ID>` is the unique barcode associated with the flow cell scanned and loaded on the instrument. (**Optional**)
- `<sampleSheet.csv file name>` is the full path to the run's `sampleSheet.csv` file. (**Optional, see** "Sample sheet" on page 3 for details.)
- `<obc2fastq_params file name>` is the full path to the `obc2fastq_params` design sheet file. (**Optional**)
  Note: Settings in the `sampleSheet.csv` file (if provided) will **override** settings in the `obc2fastq_params` file.
- `<int>` is `0` (No) or `1` (Yes), specifying whether or not to process the lanes concurrently. (**Optional**, default = `1`)
- `<int>` is an integer between 8 and 16 representing the threads per lane. (**Optional**. If not specified, an estimated number of threads is used based on system information.)
- `<control fasta file name>` is the path to the control `fasta` file. (**Optional**)
- `<int>` is an integer representing the allowed number of barcode mismatches when demultiplexing. (**Optional**, default = `0`)

obc2fastq output files

- `Control_Library.fastq.gz`: The gzipped FASTQ file for the controls. One file for each lane/read combination. (This is generated if `OutputControlFASTQ` is specified.)
- `Control_Library_Metrics.csv`: The comma-separated metrics file for the controls. One file for each lane/read combination. (This is generated if `OutputControlFASTQ` is specified.)
- `Sample_Library<_sampleID>.fastq.gz`: The gzipped FASTQ file for the samples. One file for each lane/read combination.
- `Sample_Library_Metrics.csv`: The comma-separated metrics file for the samples. One file for each lane/read combination.

- `{FlowCellBarcode}_Logs\Analysis\{FlowCellBarcode}_obc2fastq.log`: The run log.
- `{FlowCellBarcode}_Logs\Analysis\Metrics\{FlowCellBarcode}_Control_Library_Metrics.csv`: The comma-separated metrics file for the controls. One for each lane/read combination. This file is a duplicate of the one in the root folder.
- `{FlowCellBarcode}_Logs\Analysis\Metrics\{FlowCellBarcode}_Sample_Library_Metrics.csv`: The comma-separated metrics file for the samples. One for each lane/read combination. This file is a duplicate of the one in the root folder.

FASTQ file output format description

Sequence data are represented in FASTQ format, with each sequence represented by four lines of data:

- **Line 1**: Read ID:

```
@PSQ01:25:FB0012915-ABB:1:01001:35:104 1:N:0:GTACTTCTACGTT:JB?EDBKGEIHB>
1 UMI:
@<instrumentID>:<runID>:<flowcell>:<lane>:<swathtile>:<x>:<y>
<read>:<filtered>:<0>:<UMI>:<UMI_qscores>
2 UMI:
@<instrumentID>:<runID>:<flowcell>:<lane>:<swathtile>:<x>:<y>
<read>:<filtered>:<0>:<UMI1>:<UMI2>:<UMI1_qscores>:<UMI2_qscores><_sampleID>
```

- **Line 2**: Sequence data (such as `CCAGT`...)
- **Line 3**: Comment line, which always begins with a plus sign (`+`).
- **Line 4**: Quality score data, which are Phred-scale quality scores encoded in ASCII-33 characters.

Sample sheet

The sample sheet is a file containing sample information about a given sequencing run; it contains the mappings between individual samples and the index barcode sequences.

- A sample sheet is **required** if using the demultiplexing feature of `obc2fastq`, but it is **not** required to run `obc2fastq`.
- If a sample sheet is **not** provided, `obc2fastq` will generate a single `.fastq` file for **each** lane, containing all reads for that lane.

Following is a description of the sample sheet format, along with the elements in the sample sheet. Examples of elements and their constituent data are included in table format, followed by the CSV representation of the same element and data.

Sample sheet format

The sample sheet is a comma-delimited text file (`.csv`) that consists of the following elements. (See “Appendix A - Example sample sheet” on page 8 for an example.)

Sections - Sections represent a group of data and contain the following records:

- **Field labels** - Used to identify the specific values for each section.
- **Field values** - Each field value is tied to a field label and represents the sample-specific information that corresponds to a sequencing run and is to be filled in for a given sequencing experiment.

Sections are identified within brackets such that each section name precedes the data for that section.

Example: `[<SectionName>] section data …`

Allowed values for `SectionName` are: `Run Information`, `Flow Cell Settings`, `obc2fastq Settings`, and `Samples`.

Note: Every sample sheet **must** include these four sections. Within a section, some settings are optional.

Run Information section

The **Run Information** section contains metadata about the run and can be used by downstream analysis.

1. **(Required) FileFormatVersion** –Currently `2`.
2. **(Required) InstrumentPlatform** – Currently `Onso`.
3. **(Required) FlowCellBarcode** - The unique barcode associated with the flow cell that is scanned and loaded on the instrument.
4. **(Required) RunName** – User-specified text string.

[Run Information]	
FileFormatVersion	2
InstrumentPlatform	Onso
FlowcellBarcode	FC123456
RunName	MyRunName

```
[Run Information]
FileFormatVersion,2
InstrumentPlatform,Onso
FlowcellBarcode,FC123456
RunName,MyRunName
```

Flow Cell Settings section

The **Flow Cell Settings** section mirrors the Instrument Control Software (ICS) settings used by the Onso instrument.

Allowed field labels and values:

1. **(Required) Read1Cycles** – Integer, specifies the number of cycles run for insert 1. See “Appendix C - Cycle masks” on page 9 for details.
2. **(Required) Read2Cycles** – Integer, specifies the number of cycles run for insert 2. See “Appendix C - Cycle masks” on page 9 for details.
3. **(Optional) Index1Cycles** –Integer, specifies the number of cycles run for index 1.
4. **(Optional) Index2Cycles** – Integer, specifies the number of cycles run for index 2.
5. **(Required) CustomPrimer** - Must be `TRUE` or `FALSE`. Specifies if custom primers were used for the run.
6. **(Required) OBC2FASTQ** - Must be `TRUE` or `FALSE`. Specifies if a FASTQ file should be generated.

[Flow Cell Settings]	
Read1Cycles	150
Read2Cycles	150
Index1Cycles	8
Index2Cycles	8
CustomPrimer	F
OBC2FASTQ	TRUE

```
[Flow Cell Settings]
Read1Cycles,150
Read2Cycles,150
Index1Cycles,8
Index2Cycles,8
CustomPrimer,FALSE
OBC2FASTQ,TRUE
```

obc2fastq Settings section

The **obc2fastq Settings** section include the settings supported by the `obc2fastq` software.

- All of the settings in this section are **optional**.
- For boolean settings, if the setting is included, it is set to `TRUE`. If the setting is **not** included, it is set to `FALSE`.

Allowed field labels and values:

1. **IncludeTiles** – If specified, all tiles are included in the data processing. (**Note**: This setting **cannot** be used together with the **ExcludeTiles** setting.)
2. **ExcludeTiles** – If specified, all tiles are **excluded** from the data processing. (**Note**: This setting **cannot** be used together with the **IncludeTiles** setting.)
3. **OutputControlFASTQ** – If specified, a FASTQ file for the control reads is generated. If **not** specified, the FASTQ file is **not** generated.

4. **I1Mismatches** – Must be 1 or 2. Sets the maximum number of mismatches allowed in index 1 for performing demultiplexing.
5. **I2Mismatches** – Must be 1 or 2. Sets the maximum number of mismatches allowed in index 2 for performing demultiplexing.
6. **OutputIndexFASTQ** – If specified, a FASTQ file for the index reads is generated. If **not** specified, the FASTQ file is **not** generated.
   Note: Reads in the I1/I2 FASTQ files should be written in the same order as the R1/R2 FASTQ files.
7. **MergeLanes** – If specified, R1 and R2 reads with the same Sample_ID in different lanes are merged together into the same R1 and R2 FASTQ files. If **not** specified, the reads are **not** merged.
8. **R1CycleUsage** – Specifies the cycle masks for R1. The masks used specify which cycles of the .obc data tracks (T1,T2,T3,T4) to pull data from for demultiplexing into the Read 1 FASTQ output files. See “Appendix C - Cycle masks” on page 9 for more information.
9. **R2CycleUsage** – Specifies the cycle masks for R2. The masks used specify which cycles of the .obc data tracks (T1,T2,T3,T4) to pull data from for demultiplexing into Read 2 FASTQ output files. See “Appendix C - Cycle masks” on page 9 for more information.
10. **I1CycleUsage** – Specifies the cycle masks for I1. The masks used specify where the cycles are located for Index 1 demultiplexing. See “Appendix C - Cycle masks” on page 9 for more information.
11. **I2CycleUsage** – Specifies the cycle masks for I2. The masks used specify where the cycles are located for Index 2 demultiplexing. See “Appendix C - Cycle masks” on page 9 for more information.
12. **U1CycleUsage** – Specifies the cycle masks to use for UMI data. See “Appendix C - Cycle masks” on page 9 for more information.
13. **U2CycleUsage** – Specifies the cycle masks to use for UMI data. See “Appendix C - Cycle masks” on page 9 for more information.

[Obc2fastq Settings]	
IncludeTiles	L01/S01_T001,L01/S01_T002
OutputControlFASTQ	TRUE
I1Mismatches	1
I2Mismatches	1
OutputIndexFASTQ	FALSE
MergeLanes	FALSE

```
[Obc2fastq Settings]
IncludeTiles,L01/S01_T001,L01/S01_T002
OutputControlFASTQ,TRUE
I1Mismatches,1
I2Mismatches,1
OutputIndexFASTQ,FALSE
MergeLanes,FALSE
```

Samples section

The **Samples** section includes sample information and specifies whether or not demultiplexing is performed if the sample number 2 or more.

Note: The minimum entries required for the [**Samples**] section are the `[Samples]` and the `Biosample Lane Index Index2` lines**.**

Allowed field labels and values: Any number of user-specified names for individual samples sequenced on the instrument. **Note**: Only printable characters (ASCII 32-126) are allowed.

Note: `Index` and `Index2` are the index barcode sequences used to demultiplex samples that are on the same lane.

[Samples]			
Biosample	Lane	Index	Index2
BioSample1	1	AATTGCC	TGCTATC
BioSample2	1	TCGACTCA	TCATCGG
BioSample2	2	TCGACTCA	TCATGGG
BioSample3	2	AAATTGCC	TGCTATC
BioSample4	2	AAGACTCC	ACTAGCA

```
[Samples]
Biosample Lane Index Index2
BioSample1,1,AATTGCC,TGCTATC
BioSample2,1,TCGACTCA,TCATCGG
BioSample2,2,TCGACTCA,TCATGGG
BioSample3,2,AAATTGCC,TGCTATC
BioSample4,2,AAGACTCC,ACTAGCA
```

Appendix A - Example sample sheet

```
[Run Information]
FileFormatVersion,2
InstrumentPlatform,Onso
FlowcellBarcode,FC123456
RunName,MyRunName

[Flow Cell Settings]
Read1Cycles,150
Read2Cycles,150
Index1Cycles,8
Index2Cycles,8
CustomPrimer,FALSE
OBC2FASTQ,TRUE

[Obc2fastq Settings]
IncludeTiles,L01/S01_T001,L01/S01_T002
OutputControlFASTQ,TRUE
I1Mismatches,1
I2Mismatches,1
OutputIndexFASTQ,FALSE
MergeLanes,FALSE

[Samples]
Biosample Lane Index Index2
BioSample1,1,AATTGCC,TGCTATC
BioSample2,1,TCGACTCA,TCATCGG
BioSample2,2,TCGACTCA,TCATGGG
BioSample3,2,AAATTGCC,TGCTATC
BioSample4,2,AAGACTCC,ACTAGCA
```

Appendix B - Error messages

The file is not found:

```
2022-11-04 12:13:58,422 [22536] [ERROR] Open
d:\demo\20220914_FB0030734-BCC_PSQ004_65\L01\S02_T002\obc/
Read2/called_bases.obc failed.
```

The file is corrupt:

```
2022-11-04 12:50:38,789 [24032] [ERROR] File 1:02002 not all
tracks contain the same number of reads.
```

Appendix C - Cycle masks

A **cycle mask** specifies a set of cycles for a demultiplexing operation. Within a cycle mask, a series of operators indicates whether cycles are either **included** or **skipped**.

A positive integer or asterisk follows each operator to indicate a count of how many cycles are referenced.

- A `Y` (yes) operator indicates that a cycle is to be **used**.
- A `N` (no) operator indicates that a cycle is to be **skipped**.
- A positive integer indicates the number of cycles to include or exclude.
- An **asterisk** functions as a wild card, matching any remaining cycles in the read.

Examples:

- `Y4N*` - Indicates that only the first four cycles are to be used.
- `N3Y2N*` - Skips the first three cycles, uses the fourth and fifth cycles, and skips the remaining cycles.

Track identifiers

A cycle mask begins with a **track identifier** that specifies one of the `.obc` files produced by the base caller. Depending on the sequencing run, there can be between 1 and 4 files produced, such as `R1.obc`, `R2.obc`, `R3.obc`, and `R4.obc`. Each track identifier is followed by a colon (such as `T3:`).

- Example cycle mask that references the first 50 cycles of track 4 (`R4.obc`) and skips the reset of the cycles: `T4:Y50N*`

Cycle lengths

A cycle mask must define the full cycle length of a read, regardless of whether you are masking select bases in the read or all bases. For example, if the Track 1 produced by `callbase` consists of 30 bases and you want to mask the first 15, end the base mask with the total number of cycles. The base mask `T1:Y15N15` masks the first 15 bases (`Y15`) of Track 1 (`T1:`) and leaves the remaining 15 bases unmasked (`N15`).

Alternatively, `T1:Y15N*` achieves the same goal, but uses an asterisk to cover the remaining number of cycles.

Example cycle masks

- `T1:Y2N*`– Matches the first two cycles of track 1.
- `T3:N3Y100N3` – Matches 100 cycles of track 3 skipping the first and last 3 cycles.
- `T3:N2Y*N2` – Matches all but the first two and last two cycles of track 3.

Use of cycle masks in the obc2fastq section

The **[obc2fastq Settings]** section of the sample sheet uses cycle masks for settings `R1CycleUsage`, `R2CycleUsage`, `I1CycleUsage`, `I2CycleUsage`, `U1CycleUsage` and `U2CycleUsage`.

- The masks in `R1CycleUsage` and `R2CycleUsage` specify which cycles of the `.obc` data tracks (`T1`,`T2`,`T3`,`T4`) to pull data from for demultiplexing into the `Read 1` and `Read 2` FASTQ output files.
- The masks in `I1CycleUsage` and `I2CycleUsage` specify where the cycles are located for Index 1 and Index 2 demultiplexing.

The masks in `U1CycleUsage` and `U2CycleUsage` specify the cycles to use for UMI data.