Cresko Laboratory Procedures and Protocols

Cresko Laboratory

Thursday, December 7, 2023

Table of contents

How to use this book

This is a Quarto book that contains all of the Procedures and Protocols for the Cresko Laboratory in the Institute of Ecology and Evolution at the University of Oregon.

The book is organized into major section that contain

- General Laboratory Protocols or the lab
- More detailed Laboratory Protocols
- Husbandry protocols for vertebrate animals primarily stickleback and pipefish, but also zebrafish
- Husbandry protocols for *Daphnia*
- Bioinformatic protocols including how to get on to **Talapas**

You can scroll through the book using the index on the left, but also use the search field to find all relevant protocols.

There are also useful appendices at the end, as well as a section for the references cited throughout the book.

This book was written in Markdown using Quarto. To learn more about Quarto books visit https://quarto.org/docs/books.

Part I

General Laboratory Protocols

This section of the book contains general protocols for working in the laboratory.

1 Contact Information

Col1	Col2	Col3
Cresko Laboratory	541-346-5189	Phone
Bill Cresko	541-285-5446	Cell
Mark Currey	541-505-0006	Cell
Susie Bassham	xxxx	Cell

Part II

Molecular Protocols

This section of the book contains common protocols used for molecular biology and genomics in the laboratory. These include standard protocols such as setting up creating reagents, setting up PCRs and running gels, as well as advanced protocols such as creating constructs.

2 cDNA basic

2.1 Introduction

- **Purpose**: This procedure describes how to synthesis cDNA for use with PCR.
- **Procedure Type**: Molecular
- **Species**: N/A

2.2 Materials:

- 2 µl Oligo d(T)23 VN (50 µM, NEB; anchored-dT primer)*
- X µl up to 5 µg total RNA
- 1 µl 10 mM dNTP
- water
- 2 µl 10x RT buffer (Invitrogen)
- 4 µl 25 mM MgCl2
- 2 µl 0.1 mM DTT – Invitrogen
- 1 µl RNase inhibitor – e.g., RNAseOUT (Invitrogen)
- 1 µl Superscript III reverse transcriptase (200 u/µl – Invitrogen)

2.3 Solutions:

NONE

2.4 Procedure:

First strand synthesis

Combine:

- 2 µl Oligo d(T)23 VN (50 µM, NEB; anchored-dT primer)*
- X µl up to 5 µg total RNA

- 1 µl 10 mM dNTP mix
- Water (if necessary) to bring total to 10 µl

Heat to 65°C for 5 min., then ice

Collect contents at bottom of tube by brief centrifugation.

Add:

- 2 µl 10x RT buffer (Invitrogen)
- 4 µl 25 mM MgCl2
- 2 µl 0.1 mM DTT – Invitrogen
- 1 µl RNase inhibitor – e.g., RNAseOUT (Invitrogen)
- 1 µl Superscript III reverse transcriptase (200 u/µl – Invitrogen)

Mix by gentle aspiration

- 25°C for 5 min.

Reaction can be scaled up to accommodate more starting RNA

Synthesis: Incubate at 50°C for 50 min.

Inactivation: 85°C for 5 min. Chill on ice, collect contents to bottom by short spin.

Destroy RNA template: 1 µl RNase H (2 u/µl), incubate at 37°C for 20 min.

Proceed to PCR. Depending on expression level, may be able to use a dilution of cDNA as template – try 1:50 dilution in EB, use 2 µl as template in a 20 µl reaction. Don't dilute your entire amount of cDNA, as some products may require a higher concentration of template.

3 2x Turbo

3.1 Introduction

- **Purpose**: This procedure describes how to create 2x Turbo PCR mix.
- **Procedure Type**: Molecular
- **Species**: N/A

3.2 Materials:

- 33,000 µl npH2O
- 2000 µl MgSO4 (100mM)
- 1600 µl 1M Tris-HCl (pH 8.6)
- 800 µl 1M KCl
- 800 µl 1M (NH4)2SO4
- 800 µl Triton-X 100 (10%)
- 400 µl DMSO (100 %)
- 120 µl dATP (100mM)
- 120 µl dGTP (100mM)
- 120 µl dTTP (100mM)
- 120 µl dCTP (100mM)
- 80 µl 100mg/ml BSA

Total = 40 ml of buffer

3.3 Solutions:

NONE

3.4 Procedure:

- Mix above reagents together
- Place in 1.5 ml ependorph tubes
- Store at -20C

4 Paraformaldehyde

4.1 Introduction

- **Purpose**: This procedure describes how to make 8% paraformaldehyde. This protocol is the one I have used and makes use of pH to get the PFA into solution relatively quickly - then you readjust the pH. It's for 8% - then you can add 1:1 2x PBS.
- **Procedure Type**: Molecular
- **Species**: N/A

4.2 Materials:

- xxx

Total = 40 ml of buffer

4.3 Solutions:

NONE

4.4 Procedure:

> ⚠ HUMAN HEALTH WARNGING
>
> Paraformaldehyde can be hazardous to your health - make sure you prepare in the fume hood.

- Add 40 g Paraformaldehyde to 450 ml distilled water (or scale for desired final volume).
- Add 1 ul of 10 N NaOH per ml of water (i.e. 500 ul for 500 ml).
- Apply medium heat while stirring at medium speed to dissolve - approx 15-20 min.
- Solution should not go above 60º C.

- Eventually, granules will fully dissolve and the solution will become translucent.

> **DO NOT LET THE SOLUTION STIR BEYOND THIS POINT**
>
> It will form a fuzzy precipitate that reduces the solution strength after filtering.

- Once the granules have dissolved and the solution clears, turn off the heat and equilibrate to pH 7.4 with approx 1.5 ml of 20% HCl (or scale, depending on target volume).
- Bring volume to 500 ml (or scaled volume) with distilled water.
- Filter while still warm to 0.45 um (or 0.2 um). Aliquot and store at -20^{o} C.

Part III

Bioinformatic

This ection of the book contains protocols for basic bioinformatic skills such as using our laboratory cluster 'Genome', as well as our account Nereus on the UO supercomputer Talapas.

Note that there are several appendices that contain greater details and training on things such as the use of command line, R and Python, markdown and literature programming, and documentation using Quarto and Jupyter notebooks.

See Knuth (1984) for additional discussion of literate programming.

5 Placeholder_Molecular

5.1 xxx

xxxx

xxx

xxxxx

Part IV

Vertebrate Husbandry

This section of the manual contains protocols for the safe and ethical husbandry and use of vertebrate animals, particular the fish models stickleback, zebrafish and syngnathids.

6 Twenty Gallon Aquarium Cleaning

6.1 Introduction

- **Purpose**: This procedure describes how to clean 20 gallon glass tanks.
- **Procedure Type**: Husbandry
- **Species**:
  - Threespine stickleback, (*Gasterosteus aculeatus*),
  - Gulf pipefish (*Syngnathus scovelli*)

> ⚠ Schedule for Cleaning
>
> Tank cleaning is to be done ONLY Monday - Friday

6.2 Materials:

- Scrub pad or sponge
- Cart (you may or may not want to use)
- Old clothes (this can be messy)
- Personal protection equipment (Splash proof glasses or face shield).

6.3 Solutions:

- **Bleach solution**: Make a 10% bleach solution in a 2 gallon bucket. Add 4.5 L of water. Add 0.5 L of bleach and gently stir.
- **Sodium thiosulfate**: Make a 3% solution of sodium thiosulfate in a separate 2 gallon bucket. Add 5 L of water (to line) and 150g (marked on dispenser) of sodium thiosulfate. Mix

Note: When using bleach and/or sodium thiosulfate. Eye protection is required. Please use splash proof glasses or a face shield when using bleach and sodium thiosulfate.

6.4 Procedure:

- Complete bleaching and cleaning of tank. This needs to be done to each tank every 2 months.
- Remove fish from tank and put them into a clean tank. Tanks that are emptied of fish need to be cleaned and sterilized before another batch of fish can be introduced.
- Drain the tank and remove it from the rack. Clean air diffuser as instructed below.
  - Clean the tank and all parts thoroughly with a scrub pad, taking care not to damage the silicon water seals on the inside (algae should be left if very gentle rubbing will not remove it.
  - Squirt about 10 – 20 mls of bleach into the tank. Wash the bleach water thoroughly around the inside of the tank by hand using a pad or sponge exposing all inside portions of the tank to bleach.
  - Rinse the tank thoroughly with hot tap water. Rinse the tank with sodium thiosulfate, and then rinse it again with hot water. Put a few thiosulfate crystals into the tank and leave it.
  - Reassemble the tank and put it back on the rack. Fill with system water and allow water to recirculate for about 30 minutes before adding fish. Watch fish for 15min to look for any signs of distress.
  - Using a dry erase marker record date/time on the front of the tank when system water is turned back on.
- Initial the check list that you have completed the tank cleaning.

6.5 Air difuser cleaning:

- Remove dirty air diffusers from tanks and rinse with tap water to remove excess algae and debris.
- Place in 10% bleach solution for 15-30 minutes.
- Rinse the corner filters with hot water for 5 and then place into 3% sodium thiosulfate for 5 minutes.
- Rinse with hot water for 5 minutes.
- When cleaned air diffusers are placed back into aquaria, observe fish for 15 min for signs of distress.

7 Artemia Decapsulation

7.1 Introduction

- **Purpose**: This procedure describes standard practices for decapsulating brine shrimp. Although brine shrimp can be hatched, collected and then fed to fish, the cysts are often hard to separate from the newly hatched brine shrimp and can be ingested by stickleback and pipefish. To reduce this phenomenon we can decapsulate brine shrimp in advance, and then leave them in a suspended state in the freezer for an extended period of time (~ xxx weeks) before they are hatched.
- **Procedure Type**: Husbandry
- **Species**:
  - Threespine stickleback, (*Gasterosteus aculeatus*),
  - Gulf pipefish (*Syngnathus scovelli*)
- **Authors**
  - Mark C. Currey

7.2 Materials:

- 15 oz can of dried Artemia cysts (approximately 430 g)
- 4.3 L ~6% laundry grade bleach
- Rock Salt (NaCl)
- 125 ml 40% Lye (NaOH) solution
- 30.0 g Sodium thiosulfate (Na_{2}S_{2}O_{3})
- 16 L Hatching Cone with aeration
- 125 ?m mesh bag (Aquatic Eco-Systems PMB3, 125 micron x 18")
- Several 3-5 L beakers
- (1-2) Squirt bottles - squeeze type

7.3 Solutions:

> ⚠ Be ready
>
> Solutions should be prepared in advance.

- Bleach, ~6% laundry grade
  - 25 ppt Salt Solution
  - Combine: 50 g Rock Salt (NaCl) To 2.0 L with tap water
  - Stir to dissolve completely.
- 40% Lye (NaOH) solution
  - Combine: 200 g Lye (NaOH) To 500 mL with tap water
  - Stir to dissolve completely.
  - Store in refrigerator (4°C)
- Buffered Salt Solution
  - Combine: 2L, 25 ppt Salt Solution
  - 125 mL 40% Lye Solution, pre-chilled to 4°C
- 1.0% Sodium Thiosulfate
  - Combine: 30 g sodium thiosulfate To 3.0 L with tap water
  - Stir to dissolve.
- Saturated Brine
  - Combine: ~25g Rock Salt to 4.0 L with tap water
  - Aerate to dissolve.

7.4 Procedure:

1. **Cyst hydration**: Hydrate one full can of dried cyst in 5 L of tap water in a hatching cone with aeration for 1 hour at room temp. Examine the cyst under a dissecting scope with top lighting before proceeding. Dry cysts are dimpled, resembling a deflated basketball, whereas fully hydrated cysts are completely spherical in shape. The cysts must be fully hydrated prior to the de-capsulation step. If cysts are not completely spherical after 1 hour, continue the hydration process (for a maximum of 2 hours), checking the progress of the cysts under a microscope every 15 min.
2. **Filter and rinse cysts**: Collect the hydrated cyst in a 125 um mesh bag and rinse with cool tap water.

3. **Transfer cysts back to the cone**: Add the Buffered Salt Solution to the cone and aerate (save back a filled squirt bottle of salt solution to help transfer cysts to cone). Transfer cysts into cone.

4. **De-capsulation**: Add the bleach (4.3 L) to the cone and continue aeration. Watch the cysts turn from brown to grey to orange, When the cysts are 90% orange, stop the reaction by quickly siphoning the cysts through a 125 um mesh bag and rinsing well with cool tap water.

5. **Neutralization residual chlorine**: To neutralize any residual chlorine transfer the mesh bag to a clean 4 L beaker and pour the 1.0% Sodium Thiosulfate (3L) into the bag. Soak the cysts in the sodium thiosulfate solution for ~1 min, then rinse the cysts with de-ionized tap water. Rinse until discharge turns clear.

6. **Dehydration for long-term storage**: Transfer the cysts back to the cone with 4 L of saturated brine and aerate until salt is dissolved. Transfer dehydrated cyst to (5 or 6) 1 L Nalgene bottles filled with 200 - 300 grams of salt. Add enough salt so that it does not dissolve when de-capsulated brine is added. Fill the bottles with de-capsulated brine. Store in refrigerator. The de-capsulated brine will store for at least 1 month. Hatch brine as you would capsulated brine (see Hatching and Feeding Brine SOP).

Part V

Daphnia Husbandry

This section of the manual contains protocols for the safe and ethical husbandry and use of invertebrate animals, particular the nematode worm *C. remanei* and water fleas of the genus *Daphnia*

8 Placeholder_Daphnia

8.1 xxx

xxxx

xxx

xxxxx

Part VI

Bioinformatic

This ection of the book contains protocols for basic bioinformatic skills such as using our laboratory cluster 'Genome', as well as our account Nereus on the UO supercomputer Talapas.

Note that there are several appendices that contain greater details and training on things such as the use of command line, R and Python, markdown and literature programming, and documentation using Quarto and Jupyter notebooks.

See Knuth (1984) for additional discussion of literate programming.

9 A field guide to base R

9.1 Introduction

To finish off the programming section, we're going to give you a quick tour of the most important base R functions that we don't otherwise discuss in the book. These tools are particularly useful as you do more programming and will help you read code you'll encounter in the wild.

This is a good place to remind you that the tidyverse is not the only way to solve data science problems. We teach the tidyverse in this book because tidyverse packages share a common design philosophy, increasing the consistency across functions, and making each new function or package a little easier to learn and use. It's not possible to use the tidyverse without using base R, so we've actually already taught you a **lot** of base R functions: from `library()` to load packages, to `sum()` and `mean()` for numeric summaries, to the factor, date, and POSIXct data types, and of course all the basic operators like `+`, `-`, `/`, `*`, `|`, `&`, and `!`. What we haven't focused on so far is base R workflows, so we will highlight a few of those in this chapter.

After you read this book, you'll learn other approaches to the same problems using base R, data.table, and other packages. You'll undoubtedly encounter these other approaches when you start reading R code written by others, particularly if you're using StackOverflow. It's 100% okay to write code that uses a mix of approaches, and don't let anyone tell you otherwise!

In this chapter, we'll focus on four big topics: subsetting with `[`, subsetting with `[[` and `$`, the apply family of functions, and `for` loops. To finish off, we'll briefly discuss two essential plotting functions.

Prerequisites

This package focuses on base R so doesn't have any real prerequisites, but we'll load the tidyverse in order to explain some of the differences.

```
library(tidyverse)
```

9.2 Selecting multiple elements with `[`

`[` is used to extract sub-components from vectors and data frames, and is called like `x[i]` or `x[i, j]`. In this section, we'll introduce you to the power of `[`, first showing you how you can use it with vectors, then how the same principles extend in a straightforward way to two-dimensional (2d) structures like data frames. We'll then help you cement that knowledge by showing how various dplyr verbs are special cases of `[`.

Subsetting vectors

There are five main types of things that you can subset a vector with, i.e., that can be the `i` in `x[i]`:

1. **A vector of positive integers**. Subsetting with positive integers keeps the elements at those positions:

   ```
   x <- c("one", "two", "three", "four", "five")
   x[c(3, 2, 5)]
   ```

   ```
   [1] "three" "two"   "five"
   ```

   By repeating a position, you can actually make a longer output than input, making the term "subsetting" a bit of a misnomer.

   ```
   x[c(1, 1, 5, 5, 5, 2)]
   ```

   ```
   [1] "one"  "one"  "five" "five" "five" "two"
   ```

2. **A vector of negative integers**. Negative values drop the elements at the specified positions:

   ```
   x[c(-1, -3, -5)]
   ```

   ```
   [1] "two"  "four"
   ```

3. **A logical vector**. Subsetting with a logical vector keeps all values corresponding to a `TRUE` value. This is most often useful in conjunction with the comparison functions.

   ```
   x <- c(10, 3, NA, 5, 8, 1, NA)

   # All non-missing values of x
   x[!is.na(x)]
   ```

```
[1] 10  3  5  8  1
```

```
# All even (or missing!) values of x
x[x %% 2 == 0]
```

```
[1] 10 NA  8 NA
```

Unlike `filter()`, `NA` indices will be included in the output as `NA`s.

4. **A character vector**. If you have a named vector, you can subset it with a character vector:

```
x <- c(abc = 1, def = 2, xyz = 5)
x[c("xyz", "def")]
```

```
xyz def
  5   2
```

As with subsetting with positive integers, you can use a character vector to duplicate individual entries.

5. **Nothing**. The final type of subsetting is nothing, `x[]`, which returns the complete `x`. This is not useful for subsetting vectors, but as we'll see shortly, it is useful when subsetting 2d structures like tibbles.

9.3 Summary

In this chapter, we've shown you a selection of base R functions useful for subsetting and iteration. Compared to approaches discussed elsewhere in the book, these functions tend to have more of a "vector" flavor than a "data frame" flavor because base R functions tend to take individual vectors, rather than a data frame and some column specification. This often makes life easier for programming and so becomes more important as you write more functions and begin to write your own packages.

This chapter concludes the programming section of the book. You've made a solid start on your journey to becoming not just a data scientist who uses R, but a data scientist who can *program* in R. We hope these chapters have sparked your interest in programming and that you're looking forward to learning more outside of this book.

10 Summary

In summary, this book has no content whatsoever.

```
1 + 1
```

```
[1] 2
```

References

Knuth, Donald E. 1984. “Literate Programming.” *Comput. J.* 27 (2): 97–111. https://doi.org/10.1093/comjnl/27.2.97.

A Sbf1 Barcodes in 96 Well Plate

well	Barcode	Name (top)	Final top sequence	well	Name (bottom)	Final bottom sequence
A1	AAACGG	SbfI-AAACGG-top	ACACTCTTTCCCTACACGACGCTCT	A1	SbfI-AAACGG-bot	/5Phos/[illegible]
A2	AACGTT	SbfI-AACGTT-top	ACACTCTTTCCCTACACGACGCTCT	A2	SbfI-AACGTT-bot	/5Phos/[illegible]
A3	AACTGA	SbfI-AACTGA-top	ACACTCTTTCCCTACACGACGCTCT	A3	SbfI-AACTGA-bot	/5Phos/[illegible]
A4	AAGACG	SbfI-AAGACG-top	ACACTCTTTCCCTACACGACGCTCT	A4	SbfI-AAGACG-bot	/5Phos/[illegible]
A5	AAGCTA	SbfI-AAGCTA-top	ACACTCTTTCCCTACACGACGCTCT	A5	SbfI-AAGCTA-bot	/5Phos/[illegible]
A6	AATATC	SbfI-AATATC-top	ACACTCTTTCCCTACACGACGCTCT	A6	SbfI-AATATC-bot	/5Phos/[illegible]
A7	AATGAG	SbfI-AATGAG-top	ACACTCTTTCCCTACACGACGCTCT	A7	SbfI-AATGAG-bot	/5Phos/[illegible]
A8	ACAAGA	SbfI-ACAAGA-top	ACACTCTTTCCCTACACGACGCTCT	A8	SbfI-ACAAGA-bot	/5Phos/[illegible]
A9	ACAGCG	SbfI-ACAGCG-top	ACACTCTTTCCCTACACGACGCTCT	A9	SbfI-ACAGCG-bot	/5Phos/[illegible]
A10	ACATAC	SbfI-ACATAC-top	ACACTCTTTCCCTACACGACGCTCT	A10	SbfI-ACATAC-bot	/5Phos/[illegible]

well	Barcode	Name (top)	Final top sequence	well	Name (bottom)	Final bottom sequence
A11	ACCATG	SbfI-ACCATG-top	ACACTCTTTACCCTACACGACGCTCTTCCGATCTACCATG	A11	SbfI-ACCATG-bot	/5Phos/ATGGTAGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
A12	ACCCCC	SbfI-ACCCCC-top	ACACTCTTTACCCTACACGACGCTCTTCCGATCTACCCCC	A12	SbfI-ACCCCC-bot	/5Phos/GGGGGTAGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
B1	ACTCTT	SbfI-ACTCTT-top	ACACTCTTTACCCTACACGACGCTCTTCCGATCTACTCTT	B1	SbfI-ACTCTT-bot	/5Phos/AAGAGTAGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
B2	ACTGGC	SbfI-ACTGGC-top	ACACTCTTTACCCTACACGACGCTCTTCCGATCTACTGGC	B2	SbfI-ACTGGC-bot	/5Phos/GCCAGTAGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
B3	AGCCAT	SbfI-AGCCAT-top	ACACTCTTTACCCTACACGACGCTCTTCCGATCTAGCCAT	B3	SbfI-AGCCAT-bot	/5Phos/ATGGCTAGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
B4	AGCGCA	SbfI-AGCGCA-top	ACACTCTTTACCCTACACGACGCTCTTCCGATCTAGCGCA	B4	SbfI-AGCGCA-bot	/5Phos/TGCGCTAGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
B5	AGGGTC	SbfI-AGGGTC-top	ACACTCTTTACCCTACACGACGCTCTTCCGATCTAGGGTC	B5	SbfI-AGGGTC-bot	/5Phos/GACCCTAGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
B6	AGGTGT	SbfI-AGGTGT-top	ACACTCTTTACCCTACACGACGCTCTTCCGATCTAGGTGT	B6	SbfI-AGGTGT-bot	/5Phos/ACACCTAGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
B7	AGTAGG	SbfI-AGTAGG-top	ACACTCTTTACCCTACACGACGCTCTTCCGATCTAGTAGG	B7	SbfI-AGTAGG-bot	/5Phos/CCTACTAGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
B8	AGTTAA	SbfI-AGTTAA-top	ACACTCTTTACCCTACACGACGCTCTTCCGATCTAGTTAA	B8	SbfI-AGTTAA-bot	/5Phos/TTAACTAGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
B9	ATAGTA	SbfI-ATAGTA-top	ACACTCTTTACCCTACACGACGCTCTTCCGATCTATAGTA	B9	SbfI-ATAGTA-bot	/5Phos/TACTATAGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
B10	ATCAAA	SbfI-ATCAAA-top	ACACTCTTTACCCTACACGACGCTCTTCCGATCTATCAAA	B10	SbfI-ATCAAA-bot	/5Phos/TTTGATAGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
B11	ATGCAC	SbfI-ATGCAC-top	ACACTCTTTACCCTACACGACGCTCTTCCGATCTATGCAC	B11	SbfI-ATGCAC-bot	/5Phos/GTGCATAGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT

well	Barcode	Name (top)	Final top sequence	well	Name (bottom)	Final bottom sequence
B12	ATGTTG	SbfI-ATGTTG-top	ACACTCTTTCCCTACACGACGCTCTT[illegible]	B12	SbfI-ATGTTG-bot	/5Phos/[illegible]
C1	ATTCCG	SbfI-ATTCCG-top	ACACTCTTTCCCTACACGACGCTCTT[illegible]	C1	SbfI-ATTCCG-bot	/5Phos/[illegible]
C2	CAAAAA	SbfI-CAAAAA-top	ACACTCTTTCCCTACACGACGCTCTT[illegible]	C2	SbfI-CAAAAA-bot	/5Phos/[illegible]
C3	CAATCG	SbfI-CAATCG-top	ACACTCTTTCCCTACACGACGCTCTT[illegible]	C3	SbfI-CAATCG-bot	/5Phos/[illegible]
C4	CACCTC	SbfI-CACCTC-top	ACACTCTTTCCCTACACGACGCTCTT[illegible]	C4	SbfI-CACCTC-bot	/5Phos/[illegible]
C5	CAGGCA	SbfI-CAGGCA-top	ACACTCTTTCCCTACACGACGCTCTT[illegible]	C5	SbfI-CAGGCA-bot	/5Phos/[illegible]
C6	CATACT	SbfI-CATACT-top	ACACTCTTTCCCTACACGACGCTCTT[illegible]	C6	SbfI-CATACT-bot	/5Phos/[illegible]
C7	CCATTT	SbfI-CCATTT-top	ACACTCTTTCCCTACACGACGCTCTT[illegible]	C7	SbfI-CCATTT-bot	/5Phos/[illegible]
C8	CCCGGT	SbfI-CCCGGT-top	ACACTCTTTCCCTACACGACGCTCTT[illegible]	C8	SbfI-CCCGGT-bot	/5Phos/[illegible]
C9	CCCTAA	SbfI-CCCTAA-top	ACACTCTTTCCCTACACGACGCTCTT[illegible]	C9	SbfI-CCCTAA-bot	/5Phos/[illegible]
C10	CCGAGG	SbfI-CCGAGG-top	ACACTCTTTCCCTACACGACGCTCTT[illegible]	C10	SbfI-CCGAGG-bot	/5Phos/[illegible]
C11	CCGCAT	SbfI-CCGCAT-top	ACACTCTTTCCCTACACGACGCTCTT[illegible]	C11	SbfI-CCGCAT-bot	/5Phos/[illegible]
C12	CCTAAC	SbfI-CCTAAC-top	ACACTCTTTCCCTACACGACGCTCTT[illegible]	C12	SbfI-CCTAAC-bot	/5Phos/[illegible]

well	Barcode	Name (top)	Final top sequence	well	Name (bottom)	Final bottom sequence
D1	CGAGGC	SbfI-CGAGGC-top	ACACTCTTTCCCTACACGAC	D1	SbfI-CGAGGC-bot	/5Phos/GTCTTCGGATCC
D2	CGCAGA	SbfI-CGCAGA-top	ACACTCTTTCCCTACACGAC	D2	SbfI-CGCAGA-bot	/5Phos/ATCTGCGGATCC
D3	CGCGTG	SbfI-CGCGTG-top	ACACTCTTTCCCTACACGAC	D3	SbfI-CGCGTG-bot	/5Phos/ACTGCGGATCC
D4	CGGTCC	SbfI-CGGTCC-top	ACACTCTTTCCCTACACGAC	D4	SbfI-CGGTCC-bot	/5Phos/GGACCGGATCC
D5	CGTCTA	SbfI-CGTCTA-top	ACACTCTTTCCCTACACGAC	D5	SbfI-CGTCTA-bot	/5Phos/TAGACGGATCC
D6	CGTGAT	SbfI-CGTGAT-top	ACACTCTTTCCCTACACGAC	D6	SbfI-CGTGAT-bot	/5Phos/ATCACGGATCC
D7	CTACAG	SbfI-CTACAG-top	ACACTCTTTCCCTACACGAC	D7	SbfI-CTACAG-bot	/5Phos/CTGTAGGATCC
D8	CTCGCC	SbfI-CTCGCC-top	ACACTCTTTCCCTACACGAC	D8	SbfI-CTCGCC-bot	/5Phos/GGCGAGGATCC
D9	CTGCGA	SbfI-CTGCGA-top	ACACTCTTTCCCTACACGAC	D9	SbfI-CTGCGA-bot	/5Phos/TCGCAGGATCC
D10	CTGGTT	SbfI-CTGGTT-top	ACACTCTTTCCCTACACGAC	D10	SbfI-CTGGTT-bot	/5Phos/AACCAGGATCC
D11	CTTATG	SbfI-CTTATG-top	ACACTCTTTCCCTACACGAC	D11	SbfI-CTTATG-bot	/5Phos/CATAAGGATCC
D12	CTTTGC	SbfI-CTTTGC-top	ACACTCTTTCCCTACACGAC	D12	SbfI-CTTTGC-bot	/5Phos/GCAAAGGATCC
E1	GAAATG	SbfI-GAAATG-top	ACACTCTTTCCCTACACGAC	E1	SbfI-GAAATG-bot	/5Phos/CATTTCGATCC

well	Barcode	Name (top)	Final top sequence	well	Name (bottom)	Final bottom sequence
E2	GAACCA	SbfI-GAACCA-top	ACACTCTTTCCCTACACGAC	E2	SbfI-GAACCA-bot	/5Phos/ATGTTACCGATGG
E3	GACGAC	SbfI-GACGAC-top	ACACTCTTTCCCTACACGAC	E3	SbfI-GACGAC-bot	/5Phos/GTCGTACGAATGG
E4	GACTCT	SbfI-GACTCT-top	ACACTCTTTCCCTACACGAC	E4	SbfI-GACTCT-bot	/5Phos/AGAGTACGATGG
E5	GAGAGA	SbfI-GAGAGA-top	ACACTCTTTCCCTACACGAC	E5	SbfI-GAGAGA-bot	/5Phos/TCTCTACGAATGG
E6	GATCGT	SbfI-GATCGT-top	ACACTCTTTCCCTACACGAC	E6	SbfI-GATCGT-bot	/5Phos/ACGATACGATGG
E7	GCAGAT	SbfI-GCAGAT-top	ACACTCTTTCCCTACACGAC	E7	SbfI-GCAGAT-bot	/5Phos/ATCTGCGAATGG
E8	GCATGG	SbfI-GCATGG-top	ACACTCTTTCCCTACACGAC	E8	SbfI-GCATGG-bot	/5Phos/CCATGCGAATGG
E9	GCCGTA	SbfI-GCCGTA-top	ACACTCTTTCCCTACACGAC	E9	SbfI-GCCGTA-bot	/5Phos/TACGGCGATGG
E10	GCGACC	SbfI-GCGACC-top	ACACTCTTTCCCTACACGAC	E10	SbfI-GCGACC-bot	/5Phos/GGTCGCGATGG
E11	GCGCTG	SbfI-GCGCTG-top	ACACTCTTTCCCTACACGAC	E11	SbfI-GCGCTG-bot	/5Phos/CAGCGCGATGG
E12	GCTCAA	SbfI-GCTCAA-top	ACACTCTTTCCCTACACGAC	E12	SbfI-GCTCAA-bot	/5Phos/TTGAGCGATGG
F1	GGACTT	SbfI-GGACTT-top	ACACTCTTTCCCTACACGAC	F1	SbfI-GGACTT-bot	/5Phos/AAGTCCGATGG
F2	GGCAAG	SbfI-GGCAAG-top	ACACTCTTTCCCTACACGAC	F2	SbfI-GGCAAG-bot	/5Phos/CTTGCCGATGG

well	Barcode	Name (top)	Final top sequence	well	Name (bottom)	Final bottom sequence
F3	GGGCGC	SbfI-GGGCGC-top	ACACTCTTTCCCTACACGACGCTCTTCCGATCT	F3	SbfI-GGGCGC-bot	/5Phos/AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
F4	GGGGCG	SbfI-GGGGCG-top	ACACTCTTTCCCTACACGACGCTCTTCCGATCT	F4	SbfI-GGGGCG-bot	/5Phos/AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
F5	GGTACA	SbfI-GGTACA-top	ACACTCTTTCCCTACACGACGCTCTTCCGATCT	F5	SbfI-GGTACA-bot	/5Phos/AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
F6	GGTTTG	SbfI-GGTTTG-top	ACACTCTTTCCCTACACGACGCTCTTCCGATCT	F6	SbfI-GGTTTG-bot	/5Phos/AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
F7	GTAAGT	SbfI-GTAAGT-top	ACACTCTTTCCCTACACGACGCTCTTCCGATCT	F7	SbfI-GTAAGT-bot	/5Phos/AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
F8	GTATCC	SbfI-GTATCC-top	ACACTCTTTCCCTACACGACGCTCTTCCGATCT	F8	SbfI-GTATCC-bot	/5Phos/AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
F9	GTCATC	SbfI-GTCATC-top	ACACTCTTTCCCTACACGACGCTCTTCCGATCT	F9	SbfI-GTCATC-bot	/5Phos/AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
F10	GTGCCT	SbfI-GTGCCT-top	ACACTCTTTCCCTACACGACGCTCTTCCGATCT	F10	SbfI-GTGCCT-bot	/5Phos/AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
F11	GTGTAA	SbfI-GTGTAA-top	ACACTCTTTCCCTACACGACGCTCTTCCGATCT	F11	SbfI-GTGTAA-bot	/5Phos/AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
F12	GTTGGA	SbfI-GTTGGA-top	ACACTCTTTCCCTACACGACGCTCTTCCGATCT	F12	SbfI-GTTGGA-bot	/5Phos/AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
G1	TAAGCT	SbfI-TAAGCT-top	ACACTCTTTCCCTACACGACGCTCTTCCGATCT	G1	SbfI-TAAGCT-bot	/5Phos/AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
G2	TAATTC	SbfI-TAATTC-top	ACACTCTTTCCCTACACGACGCTCTTCCGATCT	G2	SbfI-TAATTC-bot	/5Phos/AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT
G3	TACACA	SbfI-TACACA-top	ACACTCTTTCCCTACACGACGCTCTTCCGATCT	G3	SbfI-TACACA-bot	/5Phos/AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT

well	Barcode	Name (top)	Final top sequence	well	Name (bottom)	Final bottom sequence
G4	TACGGG	SbfI-TACGGG-top	ACACTCTTTCCCTACACGACGCTCT [illegible]	G4	SbfI-TACGGG-bot	/5Phos/[illegible]
G5	TAGTAT	SbfI-TAGTAT-top	ACACTCTTTCCCTACACGACGCTCT [illegible]	G5	SbfI-TAGTAT-bot	/5Phos/[illegible]
G6	TATCAC	SbfI-TATCAC-top	ACACTCTTTCCCTACACGACGCTCT [illegible]	G6	SbfI-TATCAC-bot	/5Phos/[illegible]
G7	TCAAAG	SbfI-TCAAAG-top	ACACTCTTTCCCTACACGACGCTCT [illegible]	G7	SbfI-TCAAAG-bot	/5Phos/[illegible]
G8	TCCTGC	SbfI-TCCTGC-top	ACACTCTTTCCCTACACGACGCTCT [illegible]	G8	SbfI-TCCTGC-bot	/5Phos/[illegible]
G9	TCGATT	SbfI-TCGATT-top	ACACTCTTTCCCTACACGACGCTCT [illegible]	G9	SbfI-TCGATT-bot	/5Phos/[illegible]
G10	TCGCCA	SbfI-TCGCCA-top	ACACTCTTTCCCTACACGACGCTCT [illegible]	G10	SbfI-TCGCCA-bot	/5Phos/[illegible]
G11	TCGGAC	SbfI-TCGGAC-top	ACACTCTTTCCCTACACGACGCTCT [illegible]	G11	SbfI-TCGGAC-bot	/5Phos/[illegible]
G12	TCTCGG	SbfI-TCTCGG-top	ACACTCTTTCCCTACACGACGCTCT [illegible]	G12	SbfI-TCTCGG-bot	/5Phos/[illegible]
H1	TCTTCT	SbfI-TCTTCT-top	ACACTCTTTCCCTACACGACGCTCT [illegible]	H1	SbfI-TCTTCT-bot	/5Phos/[illegible]
H2	TGAACC	SbfI-TGAACC-top	ACACTCTTTCCCTACACGACGCTCT [illegible]	H2	SbfI-TGAACC-bot	/5Phos/[illegible]
H3	TGACAA	SbfI-TGACAA-top	ACACTCTTTCCCTACACGACGCTCT [illegible]	H3	SbfI-TGACAA-bot	/5Phos/[illegible]
H4	TGCCCG	SbfI-TGCCCG-top	ACACTCTTTCCCTACACGACGCTCT [illegible]	H4	SbfI-TGCCCG-bot	/5Phos/[illegible]

well	Barcode	Name (top)	Final top sequence	well	Name (bottom)	Final bottom sequence
H5	TGCTTA	SbfI-TGCTTA-top	ACACTCTTTCCCTACACGAC	H5	SbfI-TGCTTA-bot	/5Phos/ATCGTATGCCGTCTTCTGCTTG
H6	TGGGGA	SbfI-TGGGGA-top	ACACTCTTTCCCTACACGAC	H6	SbfI-TGGGGA-bot	/5Phos/ATCCTCCCCAGGATCGTATGCCGTCTTCTGCTTG
H7	TTATGA	SbfI-TTATGA-top	ACACTCTTTCCCTACACGAC	H7	SbfI-TTATGA-bot	/5Phos/ATCATAATGATCGTATGCCGTCTTCTGCTTG
H8	TTCCGT	SbfI-TTCCGT-top	ACACTCTTTCCCTACACGAC	H8	SbfI-TTCCGT-bot	/5Phos/ATCGTACGGATCGTATGCCGTCTTCTGCTTG
H9	TTCTAG	SbfI-TTCTAG-top	ACACTCTTTCCCTACACGAC	H9	SbfI-TTCTAG-bot	/5Phos/ATCTAGTAGATCGTATGCCGTCTTCTGCTTG
H10	TTGAGC	SbfI-TTGAGC-top	ACACTCTTTCCCTACACGAC	H10	SbfI-TTGAGC-bot	/5Phos/ATGCTCAAGATCGTATGCCGTCTTCTGCTTG
H11	TTTAAT	SbfI-TTTAAT-top	ACACTCTTTCCCTACACGAC	H11	SbfI-TTTAAT-bot	/5Phos/ATTAATAAATCGTATGCCGTCTTCTGCTTG
H12	TTTGTC	SbfI-TTTGTC-top	ACACTCTTTCCCTACACGAC	H12	SbfI-TTTGTC-bot	/5Phos/AGACAAAGATCGTATGCCGTCTTCTGCTTG

B Appendix 2

Hah Hah