SpiNNer Documentation

Release v2.5.0

Jonathan Heathcote

March 23, 2016

SpiNNer is a collection of tools for assisting with all stages of the process of wiring up large SpiNNaker machines, from design to installation verification.
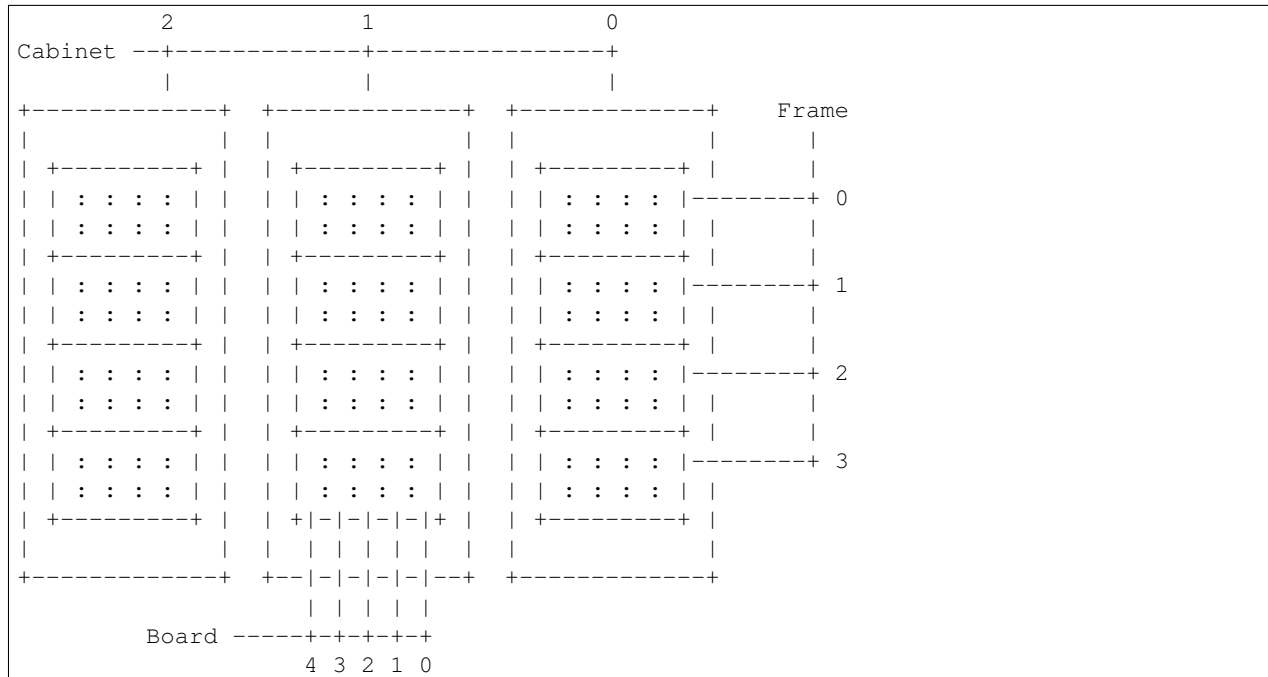
For installation and setup, see the README, for everything else, continue reading!

Terminology & Conventions

The following terminology and conventions are used throughout SpiNNer's scripts and code.

1.1 Cabinets, Frames and Boards

When discussing SpiNNaker systems, it is assumed that the system is comprised of SpiNN-5 boards arranged in standard 19" cabinets and frames like so:

```
           2                1                 0
Cabinet --+------------+---------------+
          |            |               |
+------------+  +------------+  +------------+     Frame
|            |  |            |  |            |       |
| +--------+ |  | +--------+ |  | +--------+ |       |
| | : : : | |  | | : : : | |  | | : : : |-------+ 0
| | : : : | |  | | : : : | |  | | : : : | |       |
| +--------+ |  | +--------+ |  | +--------+ |       |
| | : : : | |  | | : : : | |  | | : : : |-------+ 1
| | : : : | |  | | : : : | |  | | : : : | |       |
| +--------+ |  | +--------+ |  | +--------+ |       |
| | : : : | |  | | : : : | |  | | : : : |-------+ 2
| | : : : | |  | | : : : | |  | | : : : | |       |
| +--------+ |  | +--------+ |  | +--------+ |       |
| | : : : | |  | | : : : | |  | | : : : |-------+ 3
| | : : : | |  | | : : : | |  | | : : : | |
| +--------+ |  | +|-|-|-|-|+ |  | +--------+ |
|            |  | | | | | | | |  |            |
+------------+  +--|-|-|-|-|--+  +------------+
                   | | | | |
        Board ----+-+-+-+-+
                  4 3 2 1 0
```

A convention used throughout SpiNNer is to refer to individual boards by their position expressed in the form of (cabinet, frame, board) triples.
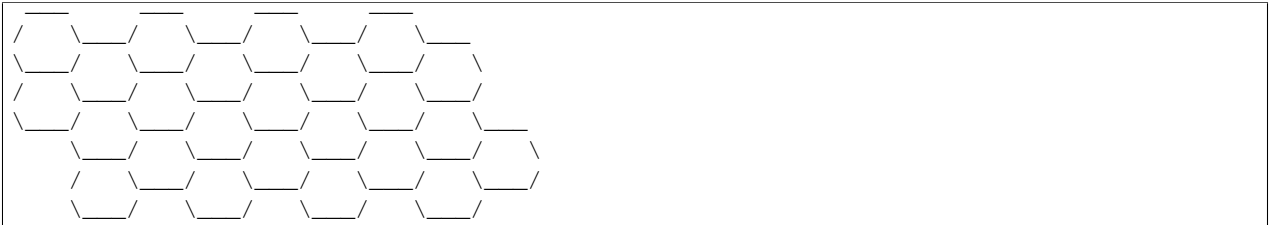
1.2 Triads and Boards

When describing large systems, their size is referred to either as a number of SpiNN-5 boards or in terms of their width/height in triads. A triad is a group of three SpiNN-5 boards connected like so:

```
   ___
  /   \___
  \___/   \
  /   \___/
  \___/
```

This pattern is repeated horizontally:

```
   ___     ___     ___     ___
  /   \___/   \___/   \___/   \___
  \___/   \___/   \___/   \___/   \
  /   \___/   \___/   \___/   \___/
  \___/   \___/   \___/   \___/
```

And then vertically:

```
   ___     ___     ___     ___
  /   \___/   \___/   \___/   \___
  \___/   \___/   \___/   \___/   \
  /   \___/   \___/   \___/   \___/
  \___/   \___/   \___/   \___/   \___
      \___/   \___/   \___/   \___/   \
      /   \___/   \___/   \___/   \___/
      \___/   \___/   \___/   \___/
```

Thus, in this example, we have a 4x2 triad, or 24 board system.

When specifying systems just in terms of numbers of boards, the squarest-possible arrangement of triads with that number of boards is assumed (preferring landscape arrangements when a square arrangement is not possible).

1.3 Physical Dimensions

All real-world physical dimensions are given in meters.

Physical coordinates are supplied in 3D space with the axes going from left-to-right, top-to-bottom and front-to-back for X, Y and Z respectively.

```
    _   Z
   /|
  /
 /
o-------> X
|
|
|
V
Y
```

1.4 Folding

For definitions of the terms used to describe the folding process (e.g. shear, slice and uncrinkling), please refer to "Bringing the Hexagonal Torus Topology into the Real-World" by Jonathan Heathcote et. al. which is available on request and is expected to be published at some future point.

SpiNNer Command-Line Utilities

SpiNNer's functions are broken down into a number of individual command-line tools, all starting with `spinner-*`. Each tool accepts arguments which describe the SpiNNaker system of interest and a complete listing of these can be found using the `-h` option.

2.1 Standard Arguments

All the SpiNNer commands accept a common set of arguments to specify the size of the system to be built.

All commands require that you specify the size of the system you're working with:

`--num-boards N` or **`-n N`** The number of boards to use. The 'squarest' possible system with this many boards will be created.

`--triads W H` or **`-t W H`** The dimensions of system in terms of triads of boards.

You can also optionally specify how these boards should be 'folded' to reduce the maximum wire-length required when assembling such systems. These options are largely for backward compatibility: the tool will automatically select a sensible folding scheme.

When working with commands which deal with real-world physical dimensions, the exact dimensions and positions of sockets, boards, frames and cabinets can be defined using additional options (see `spinner-wiring-guide -h`, for example, for a list of these). The default values represent the dimensions of standard cabinets and frames and can generally be left unchanged.

2.2 Command Documentation

Example of common uses for each command can be found below:

2.2.1 `spinner-topology-stats`

Prints basic statistics about the network topology which will result from a given system.

```
$ spinner-topology-stats -h
usage: spinner-topology-stats [-h] [--version] (--num-boards N | --triads W H)
                              [--transformation {shear,slice}]
                              [--uncrinkle-direction {columns,rows}]
                              [--folds X Y]
```

```
Print basic topological statistics for a specified configuration of boards.

optional arguments:
  -h, --help            show this help message and exit
  --version, -V         show program's version number and exit

machine topology dimensions:
  --num-boards N, -n N  build the 'squarest' system with this many boards
  --triads W H, -t W H  build a system with the specified number of triads of
                        boards in each dimension (yielding 3*W*H boards)

topology folding options:
  --transformation {shear,slice}, -T {shear,slice}
                        the transformation function to use from hexagonal
                        torus to rectangular Cartesian grid (selected
                        automatically if omitted)
  --uncrinkle-direction {columns,rows}
                        direction in which to uncrinkle the hexagonal mesh to
                        form a regular grid (default: rows)
  --folds X Y, -F X Y   the number of pieces to fold into in each dimension
                        (default: (2, 2)) ignored if --transformation is not
                        given
```

Finding sytstem dimensions

The most common use-case for `spinner-topology-stats` is determining the network dimensions of a given system of N boards.

For example, want to know about a 120-board system?

```
$ spinner-topology-stats -n 120
Topology Statistics
===================

| Measurement        | Value  | Unit  |
| ------------------ | ------ | ----- |
| Network dimensions | 96x60  | chips |
| Board array        | 8x5    | triad |
| Number of boards   | 120    |       |
| Number of cables   | 360    |       |
| Number of chips    | 5760   |       |
| Number of cores    | 103680 |       |
```

In this example, the network of SpiNNaker chips in this system will be 96x60 and the system is made up of 8x5 triads.

2.2.2 `spinner-wiring-stats`

Prints basic statistics about the wiring of a specified configuration of boards.

```
$ spinner-wiring-stats -h
usage: spinner-wiring-stats [-h] [--version] (--num-boards N | --triads W H)
                            [--transformation {shear,slice}]
                            [--uncrinkle-direction {columns,rows}]
                            [--folds X Y] [--histogram-bins N]
                            [--wire-length L [L ...]] [--minimum-slack H]
                            [--board-dimensions W H D]
```

```
                              [--board-wire-offset-south-west X Y Z]
                              [--board-wire-offset-north-east X Y Z]
                              [--board-wire-offset-east X Y Z]
                              [--board-wire-offset-west X Y Z]
                              [--board-wire-offset-north X Y Z]
                              [--board-wire-offset-south X Y Z]
                              [--inter-board-spacing S]
                              [--boards-per-frame BOARDS_PER_FRAME]
                              [--frame-dimensions W H D]
                              [--frame-board-offset X Y Z]
                              [--inter-frame-spacing S]
                              [--frames-per-cabinet FRAMES_PER_CABINET]
                              [--cabinet-dimensions W H D]
                              [--cabinet-frame-offset X Y Z]
                              [--inter-cabinet-spacing S] [--num-cabinets N]
                              [--num-frames N]


Print basic wiring statistics for a specified configuration of boards.


optional arguments:
  -h, --help            show this help message and exit
  --version, -V         show program's version number and exit

machine topology dimensions:
  --num-boards N, -n N  build the 'squarest' system with this many boards
  --triads W H, -t W H  build a system with the specified number of triads of
                        boards in each dimension (yielding 3*W*H boards)

topology folding options:
  --transformation {shear,slice}, -T {shear,slice}
                        the transformation function to use from hexagonal
                        torus to rectangular Cartesian grid (selected
                        automatically if omitted)
  --uncrinkle-direction {columns,rows}
                        direction in which to uncrinkle the hexagonal mesh to
                        form a regular grid (default: rows)
  --folds X Y, -F X Y   the number of pieces to fold into in each dimension
                        (default: (2, 2)) ignored if --transformation is not
                        given

wire length histogram options:
  --histogram-bins N, -H N
                        number of bins to pack wire lengths into in the
                        histogram of wire lengths (default: 5)

available wire lengths:
  --wire-length L [L ...], -l L [L ...]
                        specify one or more available wire lengths in meters
  --minimum-slack H     the minimum slack to allow in a wire connecting two
                        boards in meters

board physical dimensions:
  --board-dimensions W H D
                        physical board dimensions in meters (default: (0.014,
                        0.233, 0.24))
  --board-wire-offset-south-west X Y Z
                        physical offset of the south-west connector from board
                        left-top-front corner in meters (default: (0.008,
```

```
                            0.013, 0.0))
  --board-wire-offset-north-east X Y Z
                            physical offset of the north-east connector from board
                            left-top-front corner in meters (default: (0.008,
                            0.031, 0.0))
  --board-wire-offset-east X Y Z
                            physical offset of the east connector from board left-
                            top-front corner in meters (default: (0.008, 0.049,
                            0.0))
  --board-wire-offset-west X Y Z
                            physical offset of the west connector from board left-
                            top-front corner in meters (default: (0.008, 0.067,
                            0.0))
  --board-wire-offset-north X Y Z
                            physical offset of the north connector from board
                            left-top-front corner in meters (default: (0.008,
                            0.085, 0.0))
  --board-wire-offset-south X Y Z
                            physical offset of the south connector from board
                            left-top-front corner in meters (default: (0.008,
                            0.103, 0.0))
  --inter-board-spacing S
                            physical spacing between each board in a frame in
                            meters (default: 0.00124)

frame physical dimensions:
  --boards-per-frame BOARDS_PER_FRAME
                            number of boards per frame (default: 24)
  --frame-dimensions W H D
                            frame physical dimensions in meters (default: (0.43,
                            0.266, 0.25))
  --frame-board-offset X Y Z
                            physical offset of the left-top-front corner of the
                            left-most board from the left-top-front corner of a
                            frame in meters (default: (0.06, 0.017, 0.0))
  --inter-frame-spacing S
                            physical spacing between frames in a cabinet in meters
                            (default: 0.133)

cabinet physical dimensions:
  --frames-per-cabinet FRAMES_PER_CABINET
                            number of frames per cabinet (default: 5)
  --cabinet-dimensions W H D
                            cabinet physical dimensions in meters (default: (0.6,
                            2.0, 0.25))
  --cabinet-frame-offset X Y Z
                            physical offset of the left-top-front corner of the
                            top frame from the left-top-front corner of a cabinet
                            in meters (default: (0.085, 0.047, 0.0))
  --inter-cabinet-spacing S
                            physical spacing between each cabinet in meters
                            (default: 0.0)
  --num-cabinets N, -c N
                            specify how many cabinets to spread the system over
                            (default: the minimum possible)
  --num-frames N, -f N      when only one cabinet is required, specifies how many
                            frames within that cabinet the system should be spread
                            across (default: the minimum possible)
```

Selecting wire lengths to buy

By default `spinner-wiring-stats` shows a histogram of wire lengths required to wire up the supplied system in the *Wire length histogram* section. This histogram gives a basic overview of the lengths of wires required:

```
$ spinner-wiring-stats -n 120
...snip...
| Range (meters)   | Count | Histogram       | Min slack (meters) | Max slack (meters) |
| --------------- | ----- | --------------- | ------------------ | ------------------ |
| 0.00 < x <= 0.17 | 200   | ############## | 0.11               | 0.15               |
| 0.17 < x <= 0.35 | 0     |                 | 0.35               | 0.00               |
| 0.35 < x <= 0.52 | 64    | #####           | 0.10               | 0.14               |
| 0.52 < x <= 0.69 | 0     |                 | 0.69               | 0.00               |
| 0.69 < x <= 0.87 | 96    | ########        | 0.05               | 0.09               |
```

If the available wire lengths are known, these can be listed using the `--wire-length` option and the histogram will bin the wires in the system accordingly:

```
$ spinner-wiring-stats -n 120 --wire-length 0.15 0.30 0.50 1.00
...snip...
| Range (meters)   | Count | Histogram       | Min slack (meters) | Max slack (meters) |
| --------------- | ----- | --------------- | ------------------ | ------------------ |
| 0.00 < x <= 0.15 | 200   | ############## | 0.09               | 0.13               |
| 0.15 < x <= 0.30 | 0     |                 | 0.30               | 0.00               |
| 0.30 < x <= 0.50 | 64    | #####           | 0.08               | 0.12               |
| 0.50 < x <= 1.00 | 96    | ########        | 0.18               | 0.22               |
```

When deciding the length of wire to use, spinner automatically includes at least the slack specified by the `--minimum-slack` argument which defaults to 0.05 m.

The histogram table also indicates the minimum and maximum slack for each wire length. This may be important when building systems whose wires are installed in an enclosed space.

Determining the folding process

Generally, SpiNNer automatically makes all the decisions required to 'fold' the system to remove long wires. If you wish to see what decisions have been made, refer to the *Folding Parameters* section of `spinner-wiring-stats`'s output:

```
$ spinner-wiring-stats -n 120
...snip...
Folding Parameters
-----------------

| Parameter                  | Value | Unit   |
| -------------------------- | ----- | ------ |
| Number of boards           | 120   |        |
| System dimensions          | 8x5   | triads |
| Transformation             | shear |        |
| Uncrinkle Direction        | rows  |        |
| Folds                      | 2x2   | pieces |
| Number of cabinets         | 1     |        |
| Number of frames-per-cabinet | 5   |        |
| Number of boards-per-frame | 24    |        |
...snip...
```

Perfect-world wire-length measurements

The *Non-cabinetised measurements* section of `spinner-wiring-stats` gives the wire-lengths of the folded system before the boards are mapped into real-world cabinets. This section is useful when comparing alternative folding schemes since the results are not distorted by the cabinet mapping process.

The numbers in this section assume all boards are laid out in large rectangular grid and distance measures are given in units of the size of a board.

```
$ spinner-wiring-stats -n 1200
...snip...
Non-cabinetised measurements
---------------------------

| Parameter           | Value         | Unit   |
| ------------------- | ------------- | ------ |
| System dimensions   | 60.00 x 20.00 | boards |
| Mean wire length    | 2.91          | boards |
|    NE/SW            |    4.02       | boards |
|    N/S              |    2.49       | boards |
|    W/E              |    2.22       | boards |
| Maximum wire length | 4.47          | boards |
|    NE/SW            |    4.47       | boards |
|    N/S              |    2.83       | boards |
|    W/E              |    2.83       | boards |
...snip...
```

2.2.3 `spinner-wiring-diagram`

Generates illustrations of SpiNNaker machine wiring.

```
$ spinner-wiring-diagram -h
usage: spinner-wiring-diagram [-h] [--version]
                              [--wire-thickness {thick,normal,thin}]
                              [--highlight CABINET [FRAME [BOARD [{east,south-west,north,south,north-
                              [...]] [--hide-labels]
                              (--num-boards N | --triads W H)
                              [--transformation {shear,slice}]
                              [--uncrinkle-direction {columns,rows}]
                              [--folds X Y] [--board-dimensions W H D]
                              [--board-wire-offset-south-west X Y Z]
                              [--board-wire-offset-north-east X Y Z]
                              [--board-wire-offset-east X Y Z]
                              [--board-wire-offset-west X Y Z]
                              [--board-wire-offset-north X Y Z]
                              [--board-wire-offset-south X Y Z]
                              [--inter-board-spacing S]
                              [--boards-per-frame BOARDS_PER_FRAME]
                              [--frame-dimensions W H D]
                              [--frame-board-offset X Y Z]
                              [--inter-frame-spacing S]
                              [--frames-per-cabinet FRAMES_PER_CABINET]
                              [--cabinet-dimensions W H D]
                              [--cabinet-frame-offset X Y Z]
                              [--inter-cabinet-spacing S] [--num-cabinets N]
                              [--num-frames N] [--subset SUBSET [SUBSET ...]]
                              filename [width] [height]
```

```
Generate illustrations of SpiNNaker machine wiring.

optional arguments:
  -h, --help            show this help message and exit
  --version, -V         show program's version number and exit
  --wire-thickness {thick,normal,thin}
                        set the thickness of wires drawn (default: normal)
  --highlight CABINET [FRAME [BOARD [{east,south-west,north,south,north-east,west}]]] [ ...]
                        highlight a particular cabinet/frame/board/socket with
                        a red border
  --hide-labels, -L     hide socket/board/frame/cabinet number labels

image file parameters:
  filename              filename to write the output to (.pdf or .png)
  width                 width of the image in mm for PDF and pixels for PNG
                        (defaults to 280 mm if PDF and 1000 px for PNG)
  height                height of the image in mm for PDF and pixels for PNG
                        (if only width is given, output will be at most width
                        wide and width tall)

machine topology dimensions:
  --num-boards N, -n N  build the 'squarest' system with this many boards
  --triads W H, -t W H  build a system with the specified number of triads of
                        boards in each dimension (yielding 3*W*H boards)

topology folding options:
  --transformation {shear,slice}, -T {shear,slice}
                        the transformation function to use from hexagonal
                        torus to rectangular Cartesian grid (selected
                        automatically if omitted)
  --uncrinkle-direction {columns,rows}
                        direction in which to uncrinkle the hexagonal mesh to
                        form a regular grid (default: rows)
  --folds X Y, -F X Y   the number of pieces to fold into in each dimension
                        (default: (2, 2)) ignored if --transformation is not
                        given

board physical dimensions:
  --board-dimensions W H D
                        physical board dimensions in meters (default: (0.014,
                        0.233, 0.24))
  --board-wire-offset-south-west X Y Z
                        physical offset of the south-west connector from board
                        left-top-front corner in meters (default: (0.008,
                        0.013, 0.0))
  --board-wire-offset-north-east X Y Z
                        physical offset of the north-east connector from board
                        left-top-front corner in meters (default: (0.008,
                        0.031, 0.0))
  --board-wire-offset-east X Y Z
                        physical offset of the east connector from board left-
                        top-front corner in meters (default: (0.008, 0.049,
                        0.0))
  --board-wire-offset-west X Y Z
                        physical offset of the west connector from board left-
                        top-front corner in meters (default: (0.008, 0.067,
                        0.0))
  --board-wire-offset-north X Y Z
```

```
                              physical offset of the north connector from board
                              left-top-front corner in meters (default: (0.008,
                              0.085, 0.0))
  --board-wire-offset-south X Y Z
                              physical offset of the south connector from board
                              left-top-front corner in meters (default: (0.008,
                              0.103, 0.0))
  --inter-board-spacing S
                              physical spacing between each board in a frame in
                              meters (default: 0.00124)

frame physical dimensions:
  --boards-per-frame BOARDS_PER_FRAME
                              number of boards per frame (default: 24)
  --frame-dimensions W H D
                              frame physical dimensions in meters (default: (0.43,
                              0.266, 0.25))
  --frame-board-offset X Y Z
                              physical offset of the left-top-front corner of the
                              left-most board from the left-top-front corner of a
                              frame in meters (default: (0.06, 0.017, 0.0))
  --inter-frame-spacing S
                              physical spacing between frames in a cabinet in meters
                              (default: 0.133)

cabinet physical dimensions:
  --frames-per-cabinet FRAMES_PER_CABINET
                              number of frames per cabinet (default: 5)
  --cabinet-dimensions W H D
                              cabinet physical dimensions in meters (default: (0.6,
                              2.0, 0.25))
  --cabinet-frame-offset X Y Z
                              physical offset of the left-top-front corner of the
                              top frame from the left-top-front corner of a cabinet
                              in meters (default: (0.085, 0.047, 0.0))
  --inter-cabinet-spacing S
                              physical spacing between each cabinet in meters
                              (default: 0.0)
  --num-cabinets N, -c N
                              specify how many cabinets to spread the system over
                              (default: the minimum possible)
  --num-frames N, -f N  when only one cabinet is required, specifies how many
                              frames within that cabinet the system should be spread
                              across (default: the minimum possible)

wire subset selection:
  These arguments allow the specificiation of subsets of wires to install,
  for example, selecting only particular wires within a particular cabinet
  or frame. If no subsets are specified, all wires will be included,
  otherwise the union of all specified subsets are included. Use '1.2.*' to
  select all wires between boards in cabinet 1, frame 2. Use '1.*.*' to
  select all wires between boards in cabinet 1. Use '1-2.*.*' to select all
  wires which cross between cabinets 1 and 2.

  --subset SUBSET [SUBSET ...]
                              specify the subset of wires to include
```
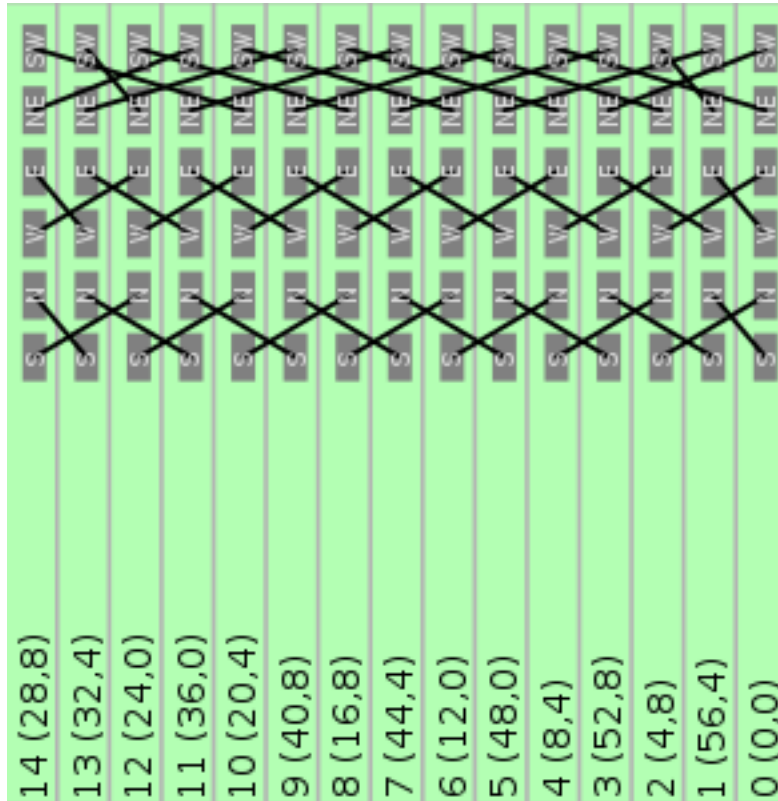
Wiring diagrams

Small SpiNNaker machines can be assembled by following a printed wiring diagram produced by `spinner-wiring-diagram`:
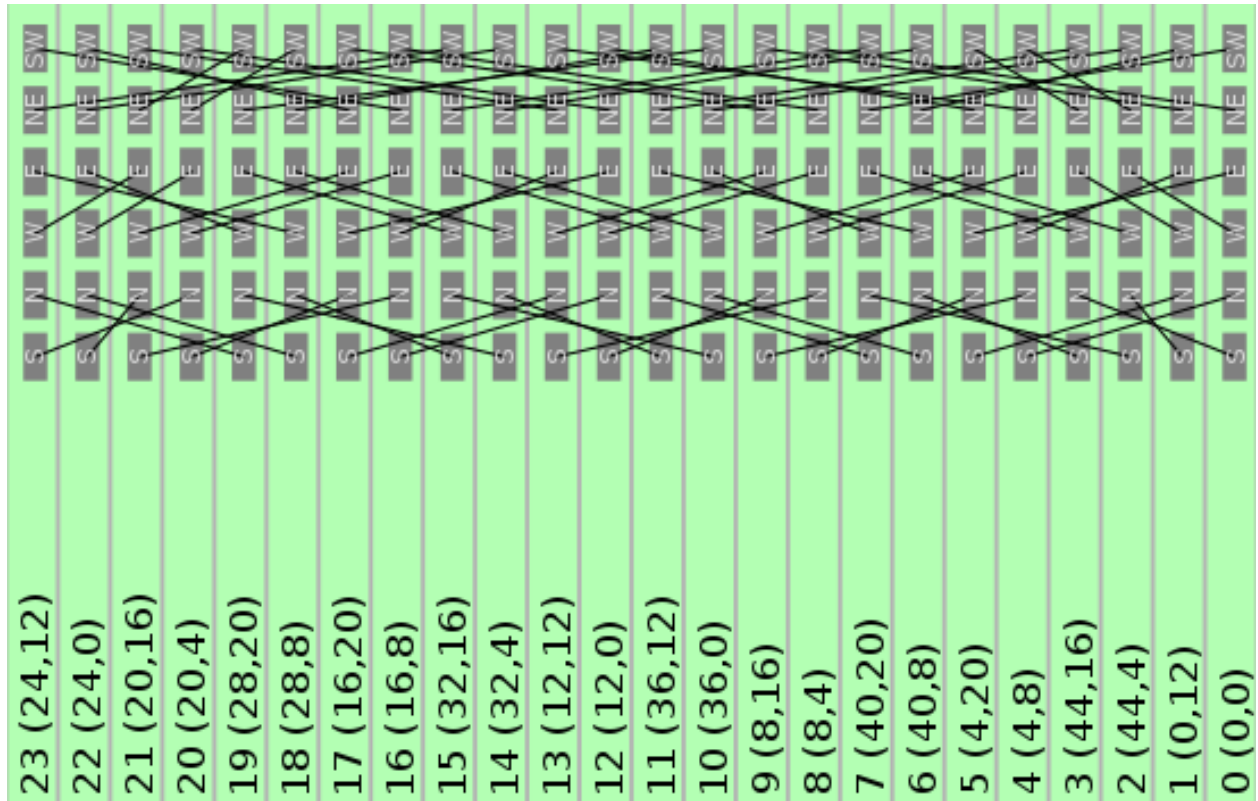
```
$ spinner-wiring-diagram -n 15 out.png
```


The diagrams can be produced as either PNG or PDF files.

For diagrams with a large number of wires, the thickness of the wires drawn can be modified with the `--wire-thickness` option to make them easier to follow:

```
$ spinner-wiring-diagram -n 24 out.png --wire-thickness thin
```

Diagrams showing just a subset of the wires can also be shown using the `--subset` argument (see *more detailed introduction*). For example, the command below generates a diagram showing only wires crossing between cabinets:
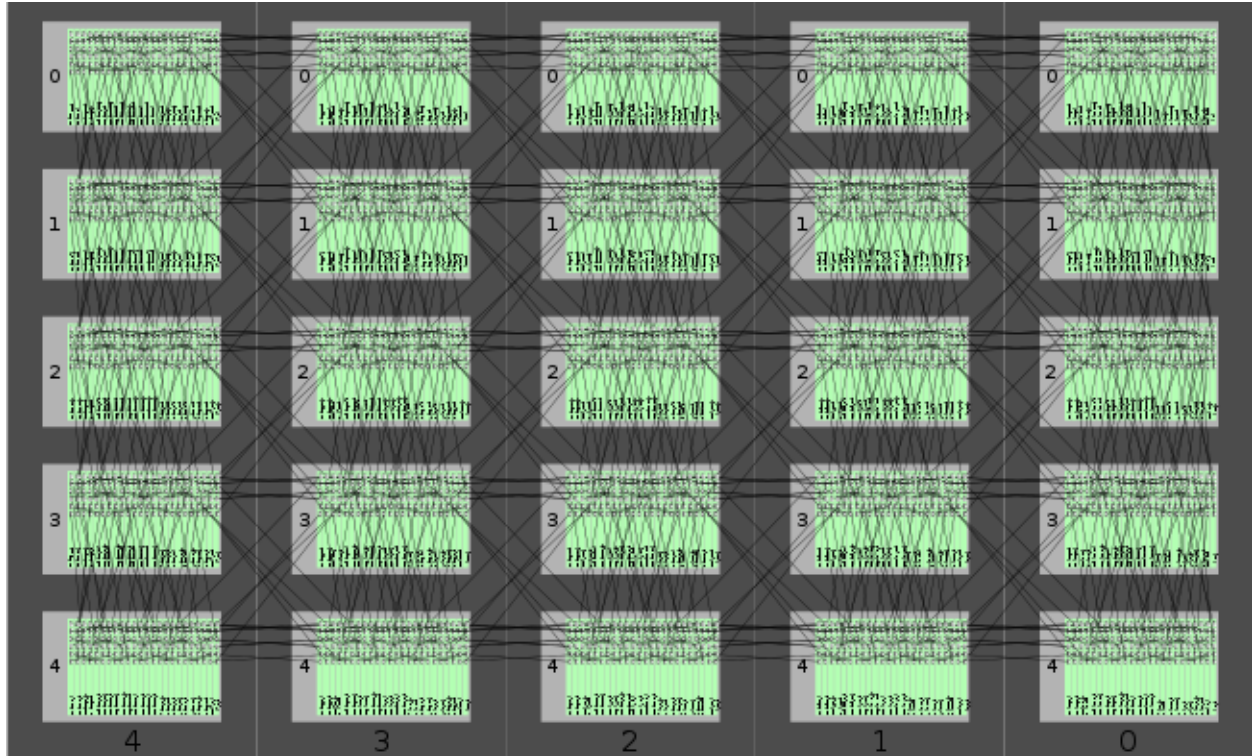
```
$ spinner-wiring-diagram -n 600 out.png --subset 0-1.*.* 1-2.*.* 2-3.*.* 3-4.*.*
```
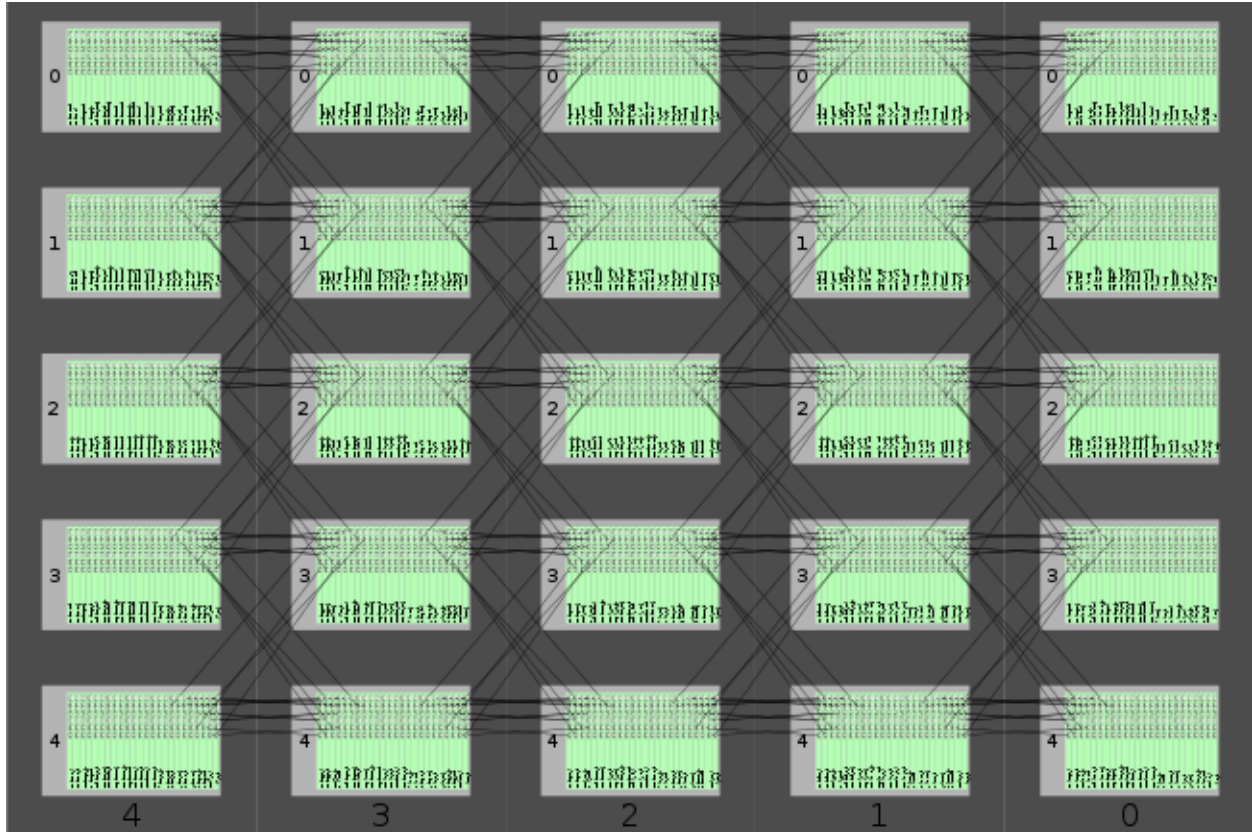
Board-to-Chip Maps

By default, all boards, frames and cabinets are numbered with their physical positions and also with the (X,Y) coordinate of the Ethernet connected chip on each board:

```
$ spinner-wiring-diagram -n 600 out.png
```

For a complementary visual mapping from SpiNNaker chip coordinates to physical board positions, see the *spinner-machine-map* command. Alternatively, for a machine-readable mapping, see the *spinner-ethernet-chips* command.

Images for presentations

If a diagram is required for illustrative purposes, the labels can be omitted using the `--hide-labels` option:

```
$ spinner-wiring-diagram -n 1200 out.png --hide-labels
```
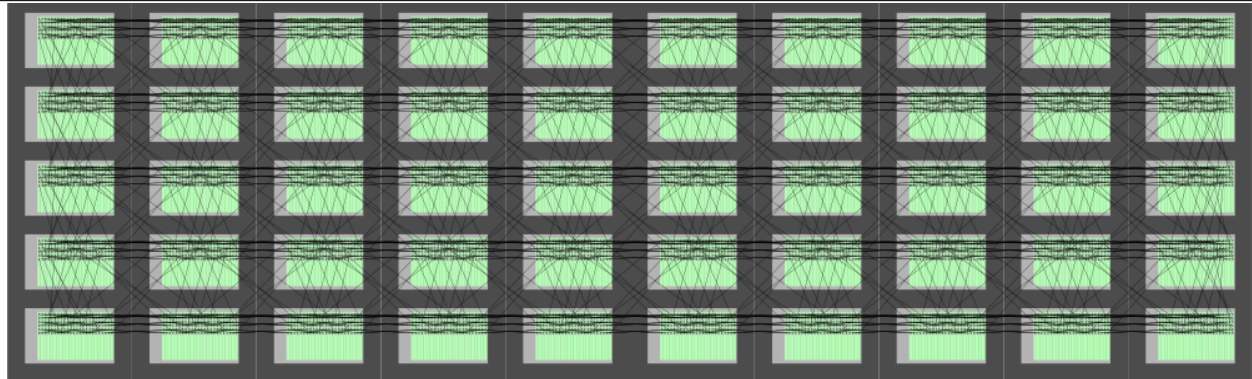

2.2.4 `spinner-wiring-guide`

Interactively guides the user through the process of wiring up a SpiNNaker machine and correcting wiring errors.

```
usage: spinner-wiring-guide [-h] [--version] [--no-tts] [--no-auto-advance]
                            [--fix] [--log LOGFILE]
                            (--num-boards N | --triads W H)
                            [--transformation {shear,slice}]
```

```
                              [--uncrinkle-direction {columns,rows}]
                              [--folds X Y] [--board-dimensions W H D]
                              [--board-wire-offset-south-west X Y Z]
                              [--board-wire-offset-north-east X Y Z]
                              [--board-wire-offset-east X Y Z]
                              [--board-wire-offset-west X Y Z]
                              [--board-wire-offset-north X Y Z]
                              [--board-wire-offset-south X Y Z]
                              [--inter-board-spacing S]
                              [--boards-per-frame BOARDS_PER_FRAME]
                              [--frame-dimensions W H D]
                              [--frame-board-offset X Y Z]
                              [--inter-frame-spacing S]
                              [--frames-per-cabinet FRAMES_PER_CABINET]
                              [--cabinet-dimensions W H D]
                              [--cabinet-frame-offset X Y Z]
                              [--inter-cabinet-spacing S] [--num-cabinets N]
                              [--num-frames N] [--wire-length L [L ...]]
                              [--minimum-slack H] [--bmp CABINET FRAME HOSTNAME]
                              [--proxy HOSTNAME] [--proxy-port PORT]
                              [--subset SUBSET [SUBSET ...]]

Interactively guide the user through the process of wiring up a SpiNNaker
machine.

optional arguments:
  -h, --help            show this help message and exit
  --version, -V         show program's version number and exit
  --no-tts              disable text-to-speech announcements of wiring steps
  --no-auto-advance     disable auto-advancing through wiring steps
  --fix                 detect errors in existing wiring and just show
                        corrective steps
  --log LOGFILE         record the times at which each cable is installed

machine topology dimensions:
  --num-boards N, -n N  build the 'squarest' system with this many boards
  --triads W H, -t W H  build a system with the specified number of triads of
                        boards in each dimension (yielding 3*W*H boards)

topology folding options:
  --transformation {shear,slice}, -T {shear,slice}
                        the transformation function to use from hexagonal
                        torus to rectangular Cartesian grid (selected
                        automatically if omitted)
  --uncrinkle-direction {columns,rows}
                        direction in which to uncrinkle the hexagonal mesh to
                        form a regular grid (default: rows)
  --folds X Y, -F X Y   the number of pieces to fold into in each dimension
                        (default: (2, 2)) ignored if --transformation is not
                        given

board physical dimensions:
  --board-dimensions W H D
                        physical board dimensions in meters (default: (0.014,
                        0.233, 0.24))
  --board-wire-offset-south-west X Y Z
                        physical offset of the south-west connector from board
                        left-top-front corner in meters (default: (0.008,
```

```
                              0.013, 0.0))
  --board-wire-offset-north-east X Y Z
                        physical offset of the north-east connector from board
                        left-top-front corner in meters (default: (0.008,
                        0.031, 0.0))
  --board-wire-offset-east X Y Z
                        physical offset of the east connector from board left-
                        top-front corner in meters (default: (0.008, 0.049,
                        0.0))
  --board-wire-offset-west X Y Z
                        physical offset of the west connector from board left-
                        top-front corner in meters (default: (0.008, 0.067,
                        0.0))
  --board-wire-offset-north X Y Z
                        physical offset of the north connector from board
                        left-top-front corner in meters (default: (0.008,
                        0.085, 0.0))
  --board-wire-offset-south X Y Z
                        physical offset of the south connector from board
                        left-top-front corner in meters (default: (0.008,
                        0.103, 0.0))
  --inter-board-spacing S
                        physical spacing between each board in a frame in
                        meters (default: 0.00124)

frame physical dimensions:
  --boards-per-frame BOARDS_PER_FRAME
                        number of boards per frame (default: 24)
  --frame-dimensions W H D
                        frame physical dimensions in meters (default: (0.43,
                        0.266, 0.25))
  --frame-board-offset X Y Z
                        physical offset of the left-top-front corner of the
                        left-most board from the left-top-front corner of a
                        frame in meters (default: (0.06, 0.017, 0.0))
  --inter-frame-spacing S
                        physical spacing between frames in a cabinet in meters
                        (default: 0.133)

cabinet physical dimensions:
  --frames-per-cabinet FRAMES_PER_CABINET
                        number of frames per cabinet (default: 5)
  --cabinet-dimensions W H D
                        cabinet physical dimensions in meters (default: (0.6,
                        2.0, 0.25))
  --cabinet-frame-offset X Y Z
                        physical offset of the left-top-front corner of the
                        top frame from the left-top-front corner of a cabinet
                        in meters (default: (0.085, 0.047, 0.0))
  --inter-cabinet-spacing S
                        physical spacing between each cabinet in meters
                        (default: 0.0)
  --num-cabinets N, -c N
                        specify how many cabinets to spread the system over
                        (default: the minimum possible)
  --num-frames N, -f N  when only one cabinet is required, specifies how many
                        frames within that cabinet the system should be spread
                        across (default: the minimum possible)
```

```
available wire lengths:
  --wire-length L [L ...], -l L [L ...]
                        specify one or more available wire lengths in meters
  --minimum-slack H     the minimum slack to allow in a wire connecting two
                        boards in meters

SpiNNaker BMP connection details:
  --bmp CABINET FRAME HOSTNAME
                        specify the hostname of a BMP to use to communicate
                        with SpiNNaker boards in the given frame

SpiNNaker proxy connection details:
  --proxy HOSTNAME      specify the hostname of a spinner-proxy-server
                        instance to use to communicate with the SpiNNaker
                        system
  --proxy-port PORT     specify the port to connect to spinner-proxy-server
                        with (default: 6512)

wire subset selection:
  These arguments allow the specificiation of subsets of wires to install,
  for example, selecting only particular wires within a particular cabinet
  or frame. If no subsets are specified, all wires will be included,
  otherwise the union of all specified subsets are included. Use '1.2.*' to
  select all wires between boards in cabinet 1, frame 2. Use '1.*.*' to
  select all wires between boards in cabinet 1. Use '1-2.*.*' to select all
  wires which cross between cabinets 1 and 2.

  --subset SUBSET [SUBSET ...]
                        specify the subset of wires to include
```
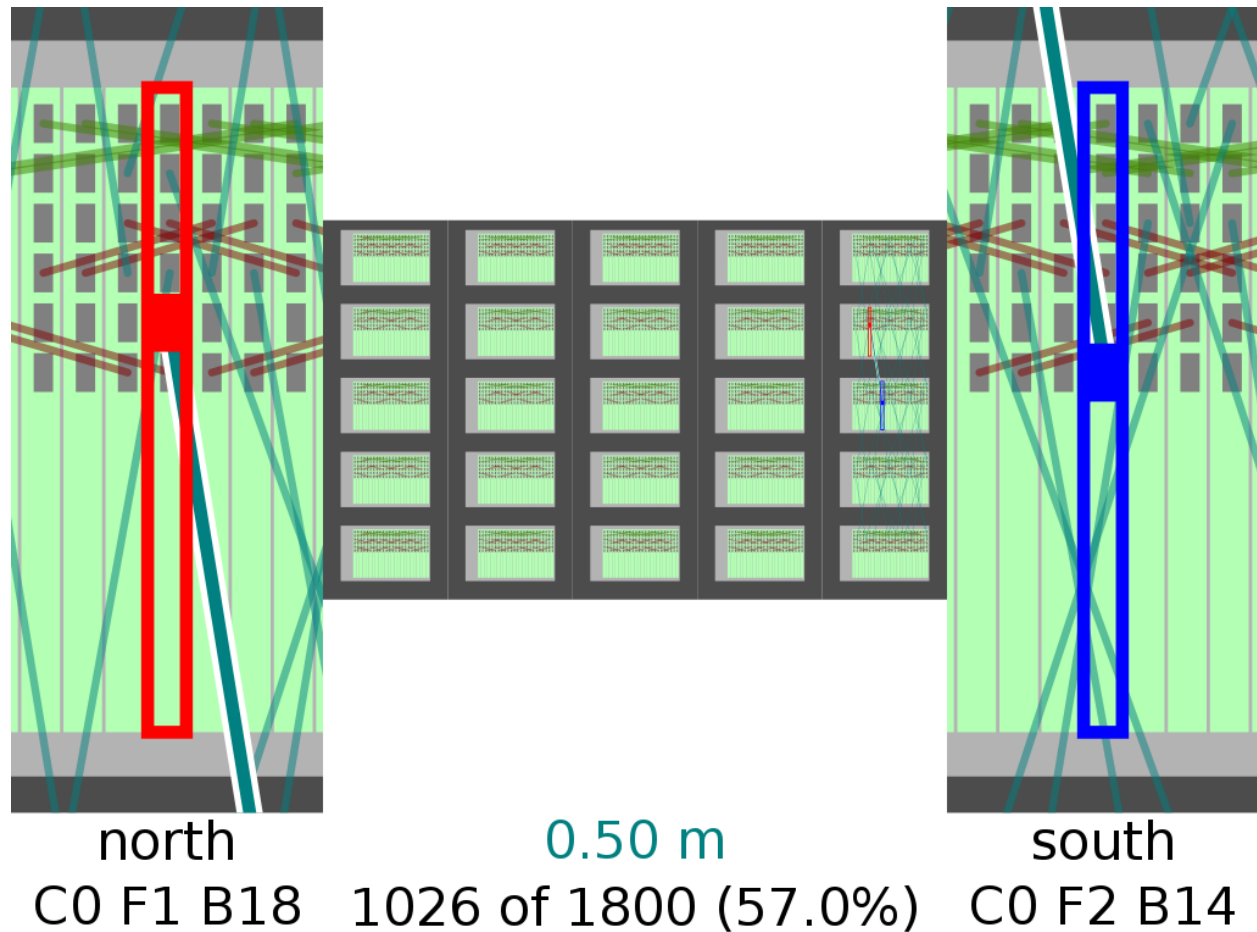
User Interface

north
C0 F1 B18 1026 of 1800 (57.0%) south C0 F2 B14

The screen shot above shows the wiring guide being used to wire up a 600 board system.

The centre of the display shows an overview of the whole machine and indicates wires that have already been installed and are coloured based on the length of the cable used. The boards and sockets to be connected are highlighted and a close-up view of the two endpoints is shown on the left- and right-hand side of the display.

Underneath the diagrams, the central caption indicates the length of wire to use for the current connection and the current progress through the whole job. The left and right captions indicate the positions of the specific boards to be connected (in terms of their cabinet, frame and board numbers).

When each wire is installed, SpiNNer verbally announces the next connection to be made using text-to-speech (TTS). Additionally, when appropriate command-line options are given, SpiNNer also illuminates a red LED on the pair of boards which are to be connected. Further, SpiNNer can automatically check for the correct installation of each wire and automatically advance to the next wire when a successful connection is made.

The length of each wire to be installed is selected by the same method as the `spinner-wiring-stats` tool. The order of wire installation proceeds in three broad phases:

- Within frames
- Between frames
- Between cabinets

Additionally, the tightest wires are installed first such that newly inserted wires can be installed over the top of existing wires.

The following keys are used to interact with the wiring guide:

Purpose	Key
Move to next wire	Left-click, Down, Space
Skip forward multiple wires	Page-Down
Move to previous wire	Right-click, Up, Backspace
Skip backward multiple wires	Page-Up
Go to first wire	Home
Go to last wire	End
Toggle Text-to-Speech	t
Toggle Auto-Advance	a
Pause logging	p

See the *spinner-proxy-server documentation* to see how this tool may be used by multiple people at once to share the task of installing cables in large machines.

Standalone Usage

To run the wiring tool stand-alone without connecting to the SpiNNaker machine being assembled, simply supply a system size and a set of available wire lengths:

```
$ spinner-wiring-guide -n 1200 -l 0.15 0.30 0.50 1.00
```

Illuminating LEDs and On-the-Fly Wire Checking

`spinner-wiring-guide` can illuminate a red LED on boards whose wires are to be connected and also check wires are inserted on-the-fly. To enable this feature, ensure all boards are powered on (e.g. using `rig-power BMP_HOSTNAME on -b 0-23` for each frame) and then use:

```
$ spinner-wiring-guide -n 24 -l 0.15 0.30 0.50 1.00 --bmp 0 0 BMP_HOSTNAME
```

Note that the `--bmp` argument must be given once for each frame in the system.

Parallel, multi-person cable installation

Several instances of `spinner-wiring-guide` can be used in parallel along with `--proxy` argument and the `spinner-proxy-server` tool to share the work of installing cables in very large SpiNNaker machines between multiple people. See the *proxy server documentation* for a complete example of how to do this.

Just Illuminating LEDs

If you do not wish to power-up the system while wiring it up, LEDs can still be illuminated while disabling the wire-checking feature using the `--no-auto-advance` argument:

```
$ spinner-wiring-guide -n 24 -l 0.15 0.30 0.50 1.00 --bmp 0 0 BMP_HOSTNAME --no-auto-advance
```

Repairing wiring errors

Adding the `--fix` option will check all installed wires in the machine and guide you through any corrections which must be made:

```
$ spinner-wiring-guide -n 24 -l 0.15 0.30 0.50 1.00 --bmp 0 0 BMP_HOSTNAME --fix
```

Installing subsets of machines

If installation is to be split into multiple phases focusing on one subsection at a time, the `--subset` argument may be used to filter the wires displayed or repaired by the wiring guide.

The `--subset` argument takes a set of arguments in the form `c.f.b` where `c`, `f`, and `b` are described below and specify a range of cabinets, frames or boards. The three parts must be in one of the following formats:

- A number (e.g. `3`) specifying a specific cabinet, frame or board.
- A pair (e.g. `1-2`) specifying a specific pair of cabinets, frames or boards.
- A wildcard (`*`).

For example:

- Subset `0.1.*` would select wires going between any boards within cabinet 0, frame 1.
- Subset `0-1.*.*` would select any cable which connects between cabinet 0 and cabinet 1.

If multiple subsets are defined, cables matched by at least one of the subsets will be selected. For example, the screenshot below was produced by the following command-line which defines two subsets:

```
$ spinner-wiring-guide -n 360 -l 0.15 0.3 0.5 0.9 --subset 0.*.* 0-1.*.*
```
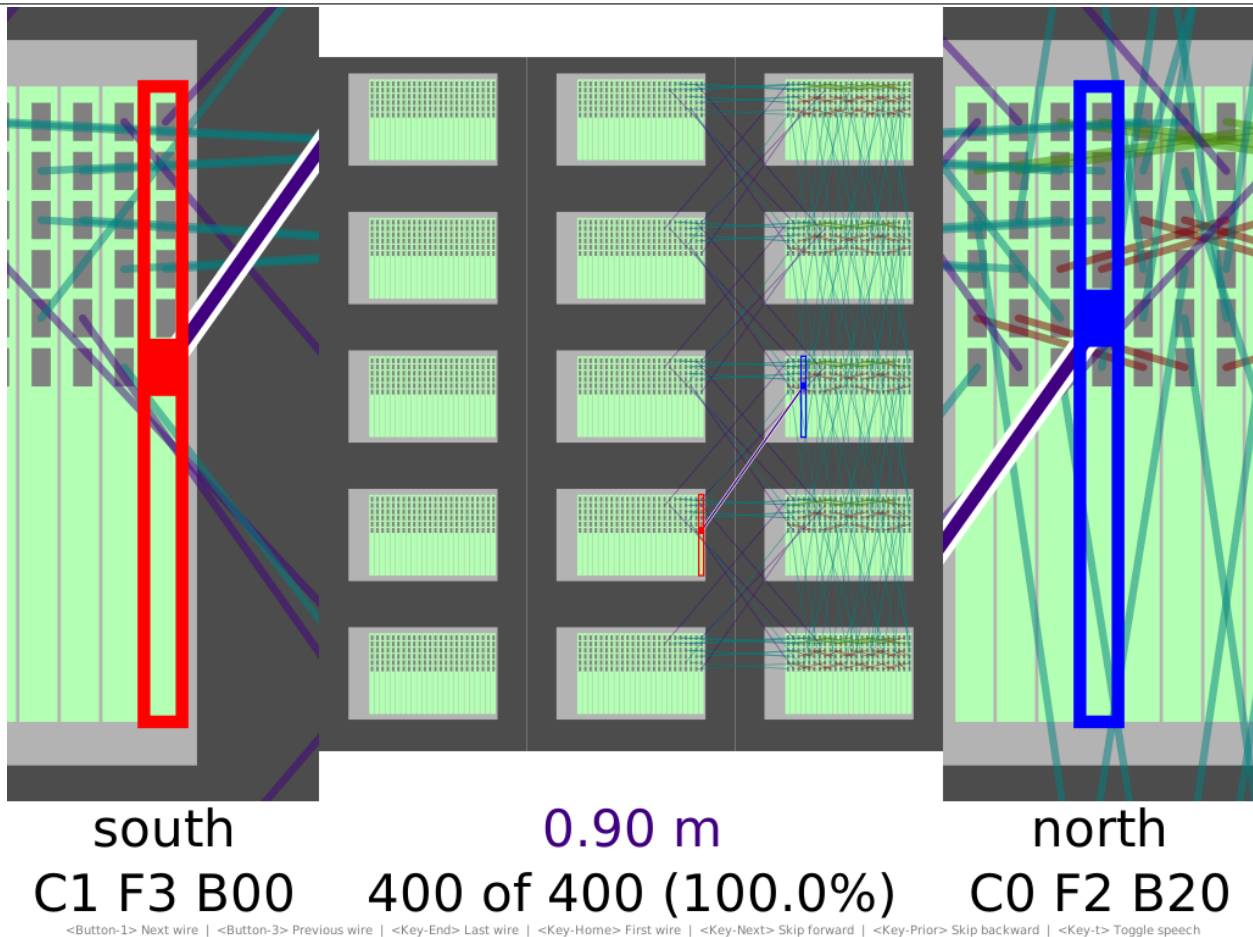
Logging

The `--log FILENAME` argument causes the wiring guide to log (into a CSV file) how long it took to install each ceable. This may be useful for research comparing ease of installation and maintainance of a SpiNNaker system. Note that this system does not currently log cables being removed.

The CSV file contains the set of columns defined below. Various types of events are recorded in the log and not every event has a sensible value for every column. Columns without a sensible value are set to NA.

event_type The type of event being logged (see list below).

realtime The real time and date the event occurred.

time Time that the event occurred, in seconds since the start of logging and excluding any time spent paused.

sc, sf, sb, sd, dc, df, db, dd Source and destination cabinet, frame, board and direction of a cable being installed.

duration Overall time, in seconds, to connect a cable correctly (or time spent paused for pause events).

attempt_duration Time since last attempt to connect the cable, in seconds.

num_attempts Number of attempts made to install the current cable.

The following event types are defined:

logging_started This event is produced when a new wiring session begins. All relative times are measured in seconds from this point.

logging_stopped Produced when logging ceases.

connection_started Produced when a new cable to install is displayed on the screen.

connection_error Produced each time a cable is connected incorrectly according the the wiring probe.

connection_complete Produced when the wiring probe detects that the cable has been installed correctly.

pause Produced *after* logging has been paused for some period of time. Relative timings reported by other events will not include any time spent paused.

2.2.5 `spinner-wiring-list`

Textually enumerates every connection required in a machine.

```
$ spinner-wiring-list -h
usage: spinner-wiring-list [-h]
                           [--sort-by {installation-order,board,wire-length}]
                           [--version] (--num-boards N | --triads W H)
                           [--transformation {shear,slice}]
                           [--uncrinkle-direction {columns,rows}]
                           [--folds X Y] [--board-dimensions W H D]
                           [--board-wire-offset-south-west X Y Z]
                           [--board-wire-offset-north-east X Y Z]
                           [--board-wire-offset-east X Y Z]
                           [--board-wire-offset-west X Y Z]
                           [--board-wire-offset-north X Y Z]
                           [--board-wire-offset-south X Y Z]
                           [--inter-board-spacing S]
                           [--boards-per-frame BOARDS_PER_FRAME]
                           [--frame-dimensions W H D]
                           [--frame-board-offset X Y Z]
```

```
                              [--inter-frame-spacing S]
                              [--frames-per-cabinet FRAMES_PER_CABINET]
                              [--cabinet-dimensions W H D]
                              [--cabinet-frame-offset X Y Z]
                              [--inter-cabinet-spacing S] [--num-cabinets N]
                              [--num-frames N] [--wire-length L [L ...]]
                              [--minimum-slack H]

Textually enumerate every connection required in a machine.

optional arguments:
  -h, --help            show this help message and exit
  --sort-by {installation-order,board,wire-length}, -s {installation-order,board,wire-length}
                        Specifies the order the connections should be listed
                        in the file: installation-order sorts in the most
                        sensible order for installation, board lists wires on
                        a board-by-board basis, wire-length lists in order of
                        wire length.
  --version, -V         show program's version number and exit

machine topology dimensions:
  --num-boards N, -n N  build the 'squarest' system with this many boards
  --triads W H, -t W H  build a system with the specified number of triads of
                        boards in each dimension (yielding 3*W*H boards)

topology folding options:
  --transformation {shear,slice}, -T {shear,slice}
                        the transformation function to use from hexagonal
                        torus to rectangular Cartesian grid (selected
                        automatically if omitted)
  --uncrinkle-direction {columns,rows}
                        direction in which to uncrinkle the hexagonal mesh to
                        form a regular grid (default: rows)
  --folds X Y, -F X Y   the number of pieces to fold into in each dimension
                        (default: (2, 2)) ignored if --transformation is not
                        given

board physical dimensions:
  --board-dimensions W H D
                        physical board dimensions in meters (default: (0.014,
                        0.233, 0.24))
  --board-wire-offset-south-west X Y Z
                        physical offset of the south-west connector from board
                        left-top-front corner in meters (default: (0.008,
                        0.013, 0.0))
  --board-wire-offset-north-east X Y Z
                        physical offset of the north-east connector from board
                        left-top-front corner in meters (default: (0.008,
                        0.031, 0.0))
  --board-wire-offset-east X Y Z
                        physical offset of the east connector from board left-
                        top-front corner in meters (default: (0.008, 0.049,
                        0.0))
  --board-wire-offset-west X Y Z
                        physical offset of the west connector from board left-
                        top-front corner in meters (default: (0.008, 0.067,
                        0.0))
  --board-wire-offset-north X Y Z
```

```
                                physical offset of the north connector from board
                                left-top-front corner in meters (default: (0.008,
                                0.085, 0.0))
  --board-wire-offset-south X Y Z
                                physical offset of the south connector from board
                                left-top-front corner in meters (default: (0.008,
                                0.103, 0.0))
  --inter-board-spacing S
                                physical spacing between each board in a frame in
                                meters (default: 0.00124)

frame physical dimensions:
  --boards-per-frame BOARDS_PER_FRAME
                                number of boards per frame (default: 24)
  --frame-dimensions W H D
                                frame physical dimensions in meters (default: (0.43,
                                0.266, 0.25))
  --frame-board-offset X Y Z
                                physical offset of the left-top-front corner of the
                                left-most board from the left-top-front corner of a
                                frame in meters (default: (0.06, 0.017, 0.0))
  --inter-frame-spacing S
                                physical spacing between frames in a cabinet in meters
                                (default: 0.133)

cabinet physical dimensions:
  --frames-per-cabinet FRAMES_PER_CABINET
                                number of frames per cabinet (default: 5)
  --cabinet-dimensions W H D
                                cabinet physical dimensions in meters (default: (0.6,
                                2.0, 0.25))
  --cabinet-frame-offset X Y Z
                                physical offset of the left-top-front corner of the
                                top frame from the left-top-front corner of a cabinet
                                in meters (default: (0.085, 0.047, 0.0))
  --inter-cabinet-spacing S
                                physical spacing between each cabinet in meters
                                (default: 0.0)
  --num-cabinets N, -c N
                                specify how many cabinets to spread the system over
                                (default: the minimum possible)
  --num-frames N, -f N  when only one cabinet is required, specifies how many
                                frames within that cabinet the system should be spread
                                across (default: the minimum possible)

available wire lengths:
  --wire-length L [L ...], -l L [L ...]
                                specify one or more available wire lengths in meters
  --minimum-slack H     the minimum slack to allow in a wire connecting two
                                boards in meters
```

Enumerating required wires

To get a list of all connections required when assembling a given machine simply run `spinner-wiring-list` specifying a number of boards and available wire lengths:

```
$ spinner-wiring-list -n 24 -l 0.15 0.3 0.5
C  F  B  Socket        C  F  B  Socket       Length
-- -- -- ----------    -- -- -- ----------    ------
 0  0  0 east           0  0  5 west          0.15
 0  0  0 north          0  0  4 south         0.15
 0  0  0 south west     0  0  7 north east    0.30
 0  0  1 east           0  0  4 west          0.15
 0  0  1 north          0  0  5 south         0.15
 0  0  1 south west     0  0  6 north east    0.15
 0  0  2 east           0  0  0 west          0.15
 0  0  2 north          0  0  1 south         0.15
[...snip...]
```

Each connection is listed with its Cabinet, Frame and Board (C, F and B) number and the socket to use. The suggested
wire length to use is also given.

By default the list is ordered by board position (top-right to bottom-left) but this can be changed to either installation-
order or ordered by wire length using the `--sort-by` arguments.

2.2.6 `spinner-wiring-validator`

Validate the wiring of a SpiNNaker system.

```
$ spinner-wiring-validator -h
usage: spinner-wiring-validator [-h] [--version] [--verbose]
                                (--num-boards N | --triads W H)
                                [--transformation {shear,slice}]
                                [--uncrinkle-direction {columns,rows}]
                                [--folds X Y] [--board-dimensions W H D]
                                [--board-wire-offset-south-west X Y Z]
                                [--board-wire-offset-north-east X Y Z]
                                [--board-wire-offset-east X Y Z]
                                [--board-wire-offset-west X Y Z]
                                [--board-wire-offset-north X Y Z]
                                [--board-wire-offset-south X Y Z]
                                [--inter-board-spacing S]
                                [--boards-per-frame BOARDS_PER_FRAME]
                                [--frame-dimensions W H D]
                                [--frame-board-offset X Y Z]
                                [--inter-frame-spacing S]
                                [--frames-per-cabinet FRAMES_PER_CABINET]
                                [--cabinet-dimensions W H D]
                                [--cabinet-frame-offset X Y Z]
                                [--inter-cabinet-spacing S] [--num-cabinets N]
                                [--num-frames N]
                                [--bmp CABINET FRAME HOSTNAME]

Validate the wiring of a SpiNNaker system.

optional arguments:
  -h, --help            show this help message and exit
  --version, -V         show program's version number and exit
  --verbose, -v         list all incorrect and missing wires

machine topology dimensions:
  --num-boards N, -n N  build the 'squarest' system with this many boards
  --triads W H, -t W H  build a system with the specified number of triads of
```

```
                          boards in each dimension (yielding 3*W*H boards)

topology folding options:
  --transformation {shear,slice}, -T {shear,slice}
                          the transformation function to use from hexagonal
                          torus to rectangular Cartesian grid (selected
                          automatically if omitted)
  --uncrinkle-direction {columns,rows}
                          direction in which to uncrinkle the hexagonal mesh to
                          form a regular grid (default: rows)
  --folds X Y, -F X Y     the number of pieces to fold into in each dimension
                          (default: (2, 2)) ignored if --transformation is not
                          given

board physical dimensions:
  --board-dimensions W H D
                          physical board dimensions in meters (default: (0.014,
                          0.233, 0.24))
  --board-wire-offset-south-west X Y Z
                          physical offset of the south-west connector from board
                          left-top-front corner in meters (default: (0.008,
                          0.013, 0.0))
  --board-wire-offset-north-east X Y Z
                          physical offset of the north-east connector from board
                          left-top-front corner in meters (default: (0.008,
                          0.031, 0.0))
  --board-wire-offset-east X Y Z
                          physical offset of the east connector from board left-
                          top-front corner in meters (default: (0.008, 0.049,
                          0.0))
  --board-wire-offset-west X Y Z
                          physical offset of the west connector from board left-
                          top-front corner in meters (default: (0.008, 0.067,
                          0.0))
  --board-wire-offset-north X Y Z
                          physical offset of the north connector from board
                          left-top-front corner in meters (default: (0.008,
                          0.085, 0.0))
  --board-wire-offset-south X Y Z
                          physical offset of the south connector from board
                          left-top-front corner in meters (default: (0.008,
                          0.103, 0.0))
  --inter-board-spacing S
                          physical spacing between each board in a frame in
                          meters (default: 0.00124)

frame physical dimensions:
  --boards-per-frame BOARDS_PER_FRAME
                          number of boards per frame (default: 24)
  --frame-dimensions W H D
                          frame physical dimensions in meters (default: (0.43,
                          0.266, 0.25))
  --frame-board-offset X Y Z
                          physical offset of the left-top-front corner of the
                          left-most board from the left-top-front corner of a
                          frame in meters (default: (0.06, 0.017, 0.0))
  --inter-frame-spacing S
                          physical spacing between frames in a cabinet in meters
```

```
                              (default: 0.089)

cabinet physical dimensions:
  --frames-per-cabinet FRAMES_PER_CABINET
                        number of frames per cabinet (default: 5)
  --cabinet-dimensions W H D
                        cabinet physical dimensions in meters (default: (0.6,
                        1.822, 0.25))
  --cabinet-frame-offset X Y Z
                        physical offset of the left-top-front corner of the
                        top frame from the left-top-front corner of a cabinet
                        in meters (default: (0.085, 0.047, 0.0))
  --inter-cabinet-spacing S
                        physical spacing between each cabinet in meters
                        (default: 0.0)
  --num-cabinets N, -c N
                        specify how many cabinets to spread the system over
                        (default: the minimum possible)
  --num-frames N, -f N  when only one cabinet is required, specifies how many
                        frames within that cabinet the system should be spread
                        across (default: the minimum possible)

SpiNNaker BMP connection details:
  --bmp CABINET FRAME HOSTNAME
                        specify the hostname of a BMP to use to communicate
                        with SpiNNaker boards in the given frame
```

Validating machine wiring

To check that all board-to-board links are operational and the wiring of a machine matches the wiring plans produced by SpiNNer, `spinner-wiring-validator` can be used:

```
$ spinner-wiring-validator -n 24 --bmp 0 0 BMP_HOSTNAME
All 72 wires correctly connected.
```

Note a `--bmp` argument for every frame in the system must be supplied giving the cabinet number, frame number and BMP hostname.

This test requires that all boards are powered on (e.g. using `rig-power BMP_HOSTNAME on -b 0-23` for each frame).

Note: `spinner-wiring-validator` only checks that the high-speed-serial board-to-board links are active and connected to the correct boards. They do not test the integrity of the actual SpiNNaker chip-to-chip network.

If any wiring errors are detected, the missing connections can be listed by enabling the `--verbose` option.

To aid in making any required corrections, you can also use *spinner-wiring-guide –fix option*.

2.2.7 `spinner-machine-map`

Generate visual maps from the SpiNNaker network topology to board locations.

```
$ spinner-machine-map -h
usage: spinner-machine-map [-h] [--version] (--num-boards N | --triads W H)
                           [--transformation {shear,slice}]
```

```
                        [--uncrinkle-direction {columns,rows}]
                        [--folds X Y] [--board-dimensions W H D]
                        [--board-wire-offset-south-west X Y Z]
                        [--board-wire-offset-north-east X Y Z]
                        [--board-wire-offset-east X Y Z]
                        [--board-wire-offset-west X Y Z]
                        [--board-wire-offset-north X Y Z]
                        [--board-wire-offset-south X Y Z]
                        [--inter-board-spacing S]
                        [--boards-per-frame BOARDS_PER_FRAME]
                        [--frame-dimensions W H D]
                        [--frame-board-offset X Y Z]
                        [--inter-frame-spacing S]
                        [--frames-per-cabinet FRAMES_PER_CABINET]
                        [--cabinet-dimensions W H D]
                        [--cabinet-frame-offset X Y Z]
                        [--inter-cabinet-spacing S] [--num-cabinets N]
                        [--num-frames N]
                        filename [width] [height]

Generate visual maps from the SpiNNaker network topology to board locations.

optional arguments:
  -h, --help            show this help message and exit
  --version, -V         show program's version number and exit

image file parameters:
  filename              filename to write the output to (.pdf or .png)
  width                 width of the image in mm for PDF and pixels for PNG
                        (defaults to 280 mm if PDF and 1000 px for PNG)
  height                height of the image in mm for PDF and pixels for PNG
                        (if only width is given, output will be at most width
                        wide and width tall)

machine topology dimensions:
  --num-boards N, -n N  build the 'squarest' system with this many boards
  --triads W H, -t W H  build a system with the specified number of triads of
                        boards in each dimension (yielding 3*W*H boards)

topology folding options:
  --transformation {shear,slice}, -T {shear,slice}
                        the transformation function to use from hexagonal
                        torus to rectangular Cartesian grid (selected
                        automatically if omitted)
  --uncrinkle-direction {columns,rows}
                        direction in which to uncrinkle the hexagonal mesh to
                        form a regular grid (default: rows)
  --folds X Y, -F X Y   the number of pieces to fold into in each dimension
                        (default: (2, 2)) ignored if --transformation is not
                        given

board physical dimensions:
  --board-dimensions W H D
                        physical board dimensions in meters (default: (0.014,
                        0.233, 0.24))
  --board-wire-offset-south-west X Y Z
                        physical offset of the south-west connector from board
                        left-top-front corner in meters (default: (0.008,
```

```
                              0.013, 0.0))
  --board-wire-offset-north-east X Y Z
                              physical offset of the north-east connector from board
                              left-top-front corner in meters (default: (0.008,
                              0.031, 0.0))
  --board-wire-offset-east X Y Z
                              physical offset of the east connector from board left-
                              top-front corner in meters (default: (0.008, 0.049,
                              0.0))
  --board-wire-offset-west X Y Z
                              physical offset of the west connector from board left-
                              top-front corner in meters (default: (0.008, 0.067,
                              0.0))
  --board-wire-offset-north X Y Z
                              physical offset of the north connector from board
                              left-top-front corner in meters (default: (0.008,
                              0.085, 0.0))
  --board-wire-offset-south X Y Z
                              physical offset of the south connector from board
                              left-top-front corner in meters (default: (0.008,
                              0.103, 0.0))
  --inter-board-spacing S
                              physical spacing between each board in a frame in
                              meters (default: 0.00124)

frame physical dimensions:
  --boards-per-frame BOARDS_PER_FRAME
                              number of boards per frame (default: 24)
  --frame-dimensions W H D
                              frame physical dimensions in meters (default: (0.43,
                              0.266, 0.25))
  --frame-board-offset X Y Z
                              physical offset of the left-top-front corner of the
                              left-most board from the left-top-front corner of a
                              frame in meters (default: (0.06, 0.017, 0.0))
  --inter-frame-spacing S
                              physical spacing between frames in a cabinet in meters
                              (default: 0.089)

cabinet physical dimensions:
  --frames-per-cabinet FRAMES_PER_CABINET
                              number of frames per cabinet (default: 5)
  --cabinet-dimensions W H D
                              cabinet physical dimensions in meters (default: (0.6,
                              1.822, 0.25))
  --cabinet-frame-offset X Y Z
                              physical offset of the left-top-front corner of the
                              top frame from the left-top-front corner of a cabinet
                              in meters (default: (0.085, 0.047, 0.0))
  --inter-cabinet-spacing S
                              physical spacing between each cabinet in meters
                              (default: 0.0)
  --num-cabinets N, -c N
                              specify how many cabinets to spread the system over
                              (default: the minimum possible)
  --num-frames N, -f N   when only one cabinet is required, specifies how many
                              frames within that cabinet the system should be spread
                              across (default: the minimum possible)
```
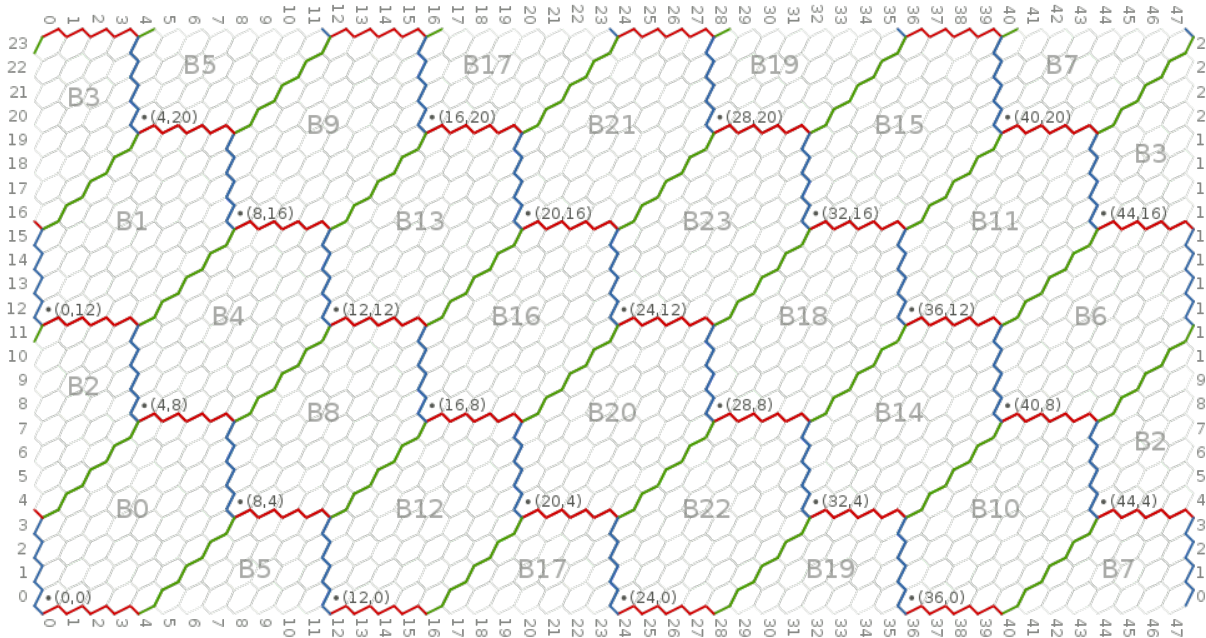
Chip-to-Board Location Maps

To aid in diagnosing hardware-related issues in large SpiNNaker applications, SpiNNer can generate maps which show where each board boundary and board-to-board link in the system lies:

```
$ spiner-machine-map -n 24 out.png
```


PDF and PNG output are supported.

In the diagram, each hexagon represents a SpiNNaker chip and board boundaries are clearly marked. The coordinates of each Ethernet connected chip is indicated in the bottom-left corner of each board. The physical location of each board is labelled in large grey text. Depending on the size of the system, the cabinet, frame and board numbers are given (prefixed with 'C', 'F' and 'B' respectively).

Board boundaries are coloured to indicate which board-to-board link is responsible for each chip-to-chip link:

Direction	Colour
North/South	Red
East/West	Green
North-East/South-West	Blue

For a complementary visual mapping from physical board positions to SpiNNaker chip coordinates, see the *spinner-wiring-diagram* command. Alternatively, for a machine-readable mapping, see the *spinner-ethernet-chips* command.

2.2.8 `spinner-ethernet-chips`

Produce CSV listings of Ethernet connected chip physical and network positions.

```
$ spinner-ethernet-chips -h
usage: spinner-ethernet-chips [-h] [--version] (--num-boards N | --triads W H)
                              [--transformation {shear,slice}]
                              [--uncrinkle-direction {columns,rows}]
                              [--folds X Y] [--board-dimensions W H D]
                              [--board-wire-offset-south-west X Y Z]
                              [--board-wire-offset-north-east X Y Z]
```

```
                                [--board-wire-offset-east X Y Z]
                                [--board-wire-offset-west X Y Z]
                                [--board-wire-offset-north X Y Z]
                                [--board-wire-offset-south X Y Z]
                                [--inter-board-spacing S]
                                [--boards-per-frame BOARDS_PER_FRAME]
                                [--frame-dimensions W H D]
                                [--frame-board-offset X Y Z]
                                [--inter-frame-spacing S]
                                [--frames-per-cabinet FRAMES_PER_CABINET]
                                [--cabinet-dimensions W H D]
                                [--cabinet-frame-offset X Y Z]
                                [--inter-cabinet-spacing S] [--num-cabinets N]
                                [--num-frames N]

Produce CSV listings of Ethernet connected chip physical and network
positions.

optional arguments:
  -h, --help            show this help message and exit
  --version, -V         show program's version number and exit

machine topology dimensions:
  --num-boards N, -n N  build the 'squarest' system with this many boards
  --triads W H, -t W H  build a system with the specified number of triads of
                        boards in each dimension (yielding 3*W*H boards)

topology folding options:
  --transformation {shear,slice}, -T {shear,slice}
                        the transformation function to use from hexagonal
                        torus to rectangular Cartesian grid (selected
                        automatically if omitted)
  --uncrinkle-direction {columns,rows}
                        direction in which to uncrinkle the hexagonal mesh to
                        form a regular grid (default: rows)
  --folds X Y, -F X Y   the number of pieces to fold into in each dimension
                        (default: (2, 2)) ignored if --transformation is not
                        given

board physical dimensions:
  --board-dimensions W H D
                        physical board dimensions in meters (default: (0.014,
                        0.233, 0.24))
  --board-wire-offset-south-west X Y Z
                        physical offset of the south-west connector from board
                        left-top-front corner in meters (default: (0.008,
                        0.013, 0.0))
  --board-wire-offset-north-east X Y Z
                        physical offset of the north-east connector from board
                        left-top-front corner in meters (default: (0.008,
                        0.031, 0.0))
  --board-wire-offset-east X Y Z
                        physical offset of the east connector from board left-
                        top-front corner in meters (default: (0.008, 0.049,
                        0.0))
  --board-wire-offset-west X Y Z
                        physical offset of the west connector from board left-
                        top-front corner in meters (default: (0.008, 0.067,
```

```
                          0.0))
  --board-wire-offset-north X Y Z
                          physical offset of the north connector from board
                          left-top-front corner in meters (default: (0.008,
                          0.085, 0.0))
  --board-wire-offset-south X Y Z
                          physical offset of the south connector from board
                          left-top-front corner in meters (default: (0.008,
                          0.103, 0.0))
  --inter-board-spacing S
                          physical spacing between each board in a frame in
                          meters (default: 0.00124)

frame physical dimensions:
  --boards-per-frame BOARDS_PER_FRAME
                          number of boards per frame (default: 24)
  --frame-dimensions W H D
                          frame physical dimensions in meters (default: (0.43,
                          0.266, 0.25))
  --frame-board-offset X Y Z
                          physical offset of the left-top-front corner of the
                          left-most board from the left-top-front corner of a
                          frame in meters (default: (0.06, 0.017, 0.0))
  --inter-frame-spacing S
                          physical spacing between frames in a cabinet in meters
                          (default: 0.089)

cabinet physical dimensions:
  --frames-per-cabinet FRAMES_PER_CABINET
                          number of frames per cabinet (default: 5)
  --cabinet-dimensions W H D
                          cabinet physical dimensions in meters (default: (0.6,
                          1.822, 0.25))
  --cabinet-frame-offset X Y Z
                          physical offset of the left-top-front corner of the
                          top frame from the left-top-front corner of a cabinet
                          in meters (default: (0.085, 0.047, 0.0))
  --inter-cabinet-spacing S
                          physical spacing between each cabinet in meters
                          (default: 0.0)
  --num-cabinets N, -c N
                          specify how many cabinets to spread the system over
                          (default: the minimum possible)
  --num-frames N, -f N  when only one cabinet is required, specifies how many
                          frames within that cabinet the system should be spread
                          across (default: the minimum possible)
```

Making Machine-Readable Physical-Position-to-Chip-Coordinate Mappings

A CSV format mapping between physical cabinet, frame and board position to the SpiNNaker network coordinates of each Ethernet connected chip can be produced with the `spinner-ethernet-chips` command:

```
$ spinner-ethernet-chips -n 3
cabinet,frame,board,x,y
0,0,0,0,0
0,0,2,4,8
0,0,1,8,4
```

For a complementary visual mapping from physical board positions to SpiNNaker chip coordinates, see the *spinner-wiring-diagram* command. For the inverse mapping see *spinner-machine-map*.

2.2.9 `spinner-proxy-server`

A proxy server which sits between very large SpiNNaker machines and multiple instances of `spinner-wiring-guide` to enable multiple people to work on a large machine at once. See below for a tutorial.

```
usage: spinner-proxy-server [-h] [--version] [--host HOST] [--port PORT]
                            [--verbose] (--num-boards N | --triads W H)
                            [--transformation {shear,slice}]
                            [--uncrinkle-direction {columns,rows}]
                            [--folds X Y] [--board-dimensions W H D]
                            [--board-wire-offset-south-west X Y Z]
                            [--board-wire-offset-north-east X Y Z]
                            [--board-wire-offset-east X Y Z]
                            [--board-wire-offset-west X Y Z]
                            [--board-wire-offset-north X Y Z]
                            [--board-wire-offset-south X Y Z]
                            [--inter-board-spacing S]
                            [--boards-per-frame BOARDS_PER_FRAME]
                            [--frame-dimensions W H D]
                            [--frame-board-offset X Y Z]
                            [--inter-frame-spacing S]
                            [--frames-per-cabinet FRAMES_PER_CABINET]
                            [--cabinet-dimensions W H D]
                            [--cabinet-frame-offset X Y Z]
                            [--inter-cabinet-spacing S] [--num-cabinets N]
                            [--num-frames N] [--bmp CABINET FRAME HOSTNAME]

Start a proxy server to enable multiple interactive wiring sessions to
interact with the same SpiNNaker machine.

optional arguments:
  -h, --help            show this help message and exit
  --version, -V         show program's version number and exit
  --host HOST, -H HOST  Host interface to listen on (default: any)
  --port PORT, -p PORT  Port listen on (default: 6512)
  --verbose, -v         Increase verbosity.

machine topology dimensions:
  --num-boards N, -n N  build the 'squarest' system with this many boards
  --triads W H, -t W H  build a system with the specified number of triads of
                        boards in each dimension (yielding 3*W*H boards)

topology folding options:
  --transformation {shear,slice}, -T {shear,slice}
                        the transformation function to use from hexagonal
                        torus to rectangular Cartesian grid (selected
                        automatically if omitted)
  --uncrinkle-direction {columns,rows}
                        direction in which to uncrinkle the hexagonal mesh to
                        form a regular grid (default: rows)
  --folds X Y, -F X Y   the number of pieces to fold into in each dimension
                        (default: (2, 2)) ignored if --transformation is not
                        given
```

```
board physical dimensions:
  --board-dimensions W H D
                        physical board dimensions in meters (default: (0.014,
                        0.233, 0.24))
  --board-wire-offset-south-west X Y Z
                        physical offset of the south-west connector from board
                        left-top-front corner in meters (default: (0.008,
                        0.013, 0.0))
  --board-wire-offset-north-east X Y Z
                        physical offset of the north-east connector from board
                        left-top-front corner in meters (default: (0.008,
                        0.031, 0.0))
  --board-wire-offset-east X Y Z
                        physical offset of the east connector from board left-
                        top-front corner in meters (default: (0.008, 0.049,
                        0.0))
  --board-wire-offset-west X Y Z
                        physical offset of the west connector from board left-
                        top-front corner in meters (default: (0.008, 0.067,
                        0.0))
  --board-wire-offset-north X Y Z
                        physical offset of the north connector from board
                        left-top-front corner in meters (default: (0.008,
                        0.085, 0.0))
  --board-wire-offset-south X Y Z
                        physical offset of the south connector from board
                        left-top-front corner in meters (default: (0.008,
                        0.103, 0.0))
  --inter-board-spacing S
                        physical spacing between each board in a frame in
                        meters (default: 0.00124)

frame physical dimensions:
  --boards-per-frame BOARDS_PER_FRAME
                        number of boards per frame (default: 24)
  --frame-dimensions W H D
                        frame physical dimensions in meters (default: (0.43,
                        0.266, 0.25))
  --frame-board-offset X Y Z
                        physical offset of the left-top-front corner of the
                        left-most board from the left-top-front corner of a
                        frame in meters (default: (0.06, 0.017, 0.0))
  --inter-frame-spacing S
                        physical spacing between frames in a cabinet in meters
                        (default: 0.133)

cabinet physical dimensions:
  --frames-per-cabinet FRAMES_PER_CABINET
                        number of frames per cabinet (default: 5)
  --cabinet-dimensions W H D
                        cabinet physical dimensions in meters (default: (0.6,
                        2.0, 0.25))
  --cabinet-frame-offset X Y Z
                        physical offset of the left-top-front corner of the
                        top frame from the left-top-front corner of a cabinet
                        in meters (default: (0.085, 0.047, 0.0))
  --inter-cabinet-spacing S
                        physical spacing between each cabinet in meters
```

```
                                 (default: 0.0)
  --num-cabinets N, -c N
                          specify how many cabinets to spread the system over
                          (default: the minimum possible)
  --num-frames N, -f N    when only one cabinet is required, specifies how many
                          frames within that cabinet the system should be spread
                          across (default: the minimum possible)

SpiNNaker BMP connection details:
  --bmp CABINET FRAME HOSTNAME
                          specify the hostname of a BMP to use to communicate
                          with SpiNNaker boards in the given frame
```

Parallel, multi-person cable installation

To enable multiple people to install the cables in a very large machine simultaneously, a `spinner-proxy-server` may be set up which allows multiple instances of the `spinner-wiring-guide` tool to be started, one instance per person (ideally running on seperate computers and headphones).

The server should be started (on a single machine) like so:

```
$ spinner-proxy-server -n 600 --bmp 0 0 192.168.1.0 ...
Proxy server starting...
```

Note that the arguments specifying the size of the machine, along with BMP IP addresses for all frames, must be given.

Next, each installer must start the interactive wiring guide, substituting the ususal `--bmp` arguments for the hostname of the machine running the proxy server. In addition the `--subset` argument (see *the documentation*) should be used to assign each installer a non-overlapping portion of the machine.

For example, if the first installer will install all cables within cabinet 0 and between cabinets 1 and 0, the following command would be used to start the wiring guide on their machine:

```
$ spinner-wiring-guide -n 600 -l 0.15 0.3 0.5 0.7 1.0 --proxy hostname --subset 0.*.* 0-1.*.*
```

If the second installer will install cables in cabinet 1 and going between cabinets 1 and 2, the following command would be used on their machine:

```
$ spinner-wiring-guide -n 600 -l 0.15 0.3 0.5 0.7 1.0 --proxy hostname --subset 1.*.* 1-2.*.*
```

...and so on!

Since SpiNNer will install cables staying within cabinets before moving to inter-cabinet cabling, the scheme described above should result in minimal overlap between installers' activity. An alternative scheme would be for each installer to first complete the wiring within their assigned cabinet and, once complete, restart the tool to guide installation of the cables between cabinets.

2.3 Backwards compatibility

Options which recreate wiring plans from old versions of SpiNNer can be found here:

2.3.1 Backward Compatible Options

The following arguments can be used to create wiring plans equivalent to those produced by old versions of SpiNNer.

Note: These values are only provided for backwards compatibility (i.e. working with existing systems). It is highly recommended that new systems are built using the plans generated by `--num-boards N`.

Old inter-frame spacing

Older multi-frame SpiNNaker machines use a different inter-frame spacing of 0.089 meters. Prior to SpiNNer version 2.3.0 this was the default. From Version 2.3.0 onwards a new inter-frame spacing default is 0.133 meters. To use the old inter-frame spacing add the following argument to SpiNNer commands:

```
--inter-frame-spacing=0.089
```

`spin_103` (3 boards)

```
--triads 1 1 --transformation slice --uncrinkle-direction rows --folds 1 1
```

`spin_1037` (6 boards)

```
--triads 2 1 --transformation slice --uncrinkle-direction rows --folds 1 1
```

`spin_104` (12 boards)

```
--triads 2 2 --transformation slice --uncrinkle-direction rows --folds 2 1
```

`spin_1043` (24 boards)

```
--triads 4 2 --transformation slice --uncrinkle-direction rows --folds 1 1
```

`spin_105` (120 boards)

```
--triads 8 5 --transformation slice --uncrinkle-direction columns --folds 1 2
```

`spin_1057` (600 boards)

```
--triads 10 20 --transformation slice --uncrinkle-direction rows --folds 2 2
```

`spin_106` (1200 boards)

```
--triads 20 20 --transformation slice --uncrinkle-direction rows --folds 4 2
```

Indices and tables

- genindex
- modindex
- search