

# Chapter 1

## Star + Placer

### README

Starplacer is an analytic placer created for the FPGA placement problem. It works by solving / minimizing a system of linear equations describing the estimated wirelength cost. It uses variations of [Conjugate Gradient](#) and [Successive Over Relaxation](#) to iteratively improve the placement.

### Contents

- **arch:** [Architecture](#) files describe the hardware configuration a [Netlist](#) is to be mapped onto.
- **benchmarks:** contains [Netlist](#) files of various sizes, ex5p.net is the smallest and clma.net is one of the largest.
- **src:** the source code directory, a combination of C++ and C.
- **include:** the header files, most documentation is contained here.
- **docs:** generated documentation extracted from the source code.
- **references:** a couple of papers related to VPR and analytic placement.
- **Makefile:** used to compile the source code.

### Starplacer

- [Netlist](#) and [Architecture](#) parsing code is from [Versatile Place and Route](#)
- Original starplacer author: Ming Xu [camingxu@gmail.com](mailto:camingxu@gmail.com)
- Translation from C#: Ryan Pattison [rpattiso@uoguelph.ca](mailto:rpattiso@uoguelph.ca)

### Documentation Generation

Compiling the HTML/LaTeX documentation requires some additional tools and packages. All the actual documentation is inlined in the `.cpp` and `.h` files so compiling to the HTML version is not necessary if that is sufficient.

```
sudo apt-get install -i doxygen msc-gen graphviz-dot markdown
```

For the LaTeX documentation the following packages are required in addition to the above:

```
sudo apt-get install -i pdflatex
```

In the root directory:

```
cd docs
doxygen
```

After that, the html folder will be present in the docs directory:

```
gnome-open docs/html/index.html
```

For the LaTeX PDF file:

```
cd docs/latex && make
gnome-open docs/latex/refman.pdf
```

## Compiling The Placer

In the root directory type `make` and the executable will be in put in the root and named `starplacer`. By default this is the optimized release version, to compile without optimizations and with assertions enabled use:

```
make debug
```

## Running The Placer

In the root directory:

```
./starplacer benchmarks/mcnc/tseng.net arch/4lut_sanitized.arch place.out --place_algorithm
SOR
```

For the `--place_algorithm` option, you may choose Conjugate Gradient (CG) or Successive over Relaxation (SOR)

The intermediate costs are printed to the screen and the output placement written to the file specified as the third argument (`place.out`).

The output file is a tab separated file with some additional syntax listing block x, y, and z positions.

```
Netlist file: ./benchmarks/mcnc/tseng.net Architecture file: ./arch/4lut_sanitized.
arch
Array size: 33 x 33 logic blocks

#block name      x      y      subblk  block number
#-----
tin_pv10_4_4_    6      34      0      #0
tin_pv11_4_4_    0      24      0      #1
tin_pv6_7_7_     16     0       0      #2
tin_pv2_0_0_     34     16     0      #3
tin_pv10_3_3_    7      34     0      #4
tin_pv1_2_2_     34     16     0      #5
tin_pv11_3_3_    0      27     0      #6
...
```

## References

In the `references` directory there are some papers which you may find useful.

`references/manual_430.pdf` : The manual for VPR v4.3 which explains the `.arch` `.net` and placement file types.

`references/ming_xu-phd_thesis.pdf` : The original Star+Placer authors Ph. D thesis, covers the algorithms used at a high level.

Papers not included, but great references which covers VPR and is a good overview of the placement problem in general.

Betz, Vaughn, and Jonathan Rose. "VPR: A new packing, placement and routing tool for FPGA research." *Field-Programmable Logic and Applications*. Springer Berlin Heidelberg, 1997.