Chrystian Guth

Curriculum Vitae

Experience

05/2019–now **Senior Software Engineer**, *Communication Security Group*, Florianópolis, Brazil.
- Cross-platform SIP UA development targeting Windows, Android, Mac and Linux.
  - C, C++11, C++14, C#, Kotlin
  - PJSIP and OpenSSL low-level C programming
  - Cross-compiling for Android
  - Research on Codecs and TURN server placement and allocation
  - Git, TDD and Continuous Integration
- Asterisk channel-driver development

09/2017–04/2019 **Software Engineer**, *Communication Security Group*, Florianópolis, Brazil.
- CellCrypt Classified 2.0 (NIAP Certificated)
  - PJSIP and OpenSSL low-level C programming
  - Cross-compiling for Android

01/2017–07/2017 **Software Engineering Intern**, *Cadence Design Systems*, Cambridge, United Kingdom.
- Development of tools for Electronic Design Automation with emphasis on Timing Analysis.
  - C++03 and C++11
  - Code reviews
  - Daily commits

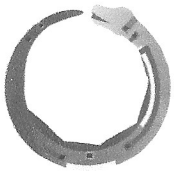
2016 **Software Engineer**, *Cellcrypt*, Florianópolis, Brazil.
- Development of tools for secure online communication with audio and video.
  - Modern C++
  - Cross-platform (MacOS, Win32)
  - Code reviews
  - Daily commits

Rod. João Paulo - 435 – Florianópolis – Brazil
+55 (48) 99606 6659 • *csguth@gmail.com* • *csguth.com*

CG

2014–2016 **Researcher on EDA & PD**, *Embedded Computing Laboratory - Federal University of Santa Catarina*, Florianópolis, Brazil.

- Research and development of algorithms for Timing-Driven Placement [3] [4] [5], Timing-Analysis and Timing-Aware Power-Driven Placement
- Ophidian Project [`https://github.com/eclufsc/ophidian`]



Ophidian

2011–2013 **Undergrad. Research Assistant**, *Embedded Computing Laboratory - Federal University of Santa Catarina*, Florianópolis, Brazil.

- Research and development of algorithms for Timing-Analysis and Gate-Sizing [6] [7] [8] [9]

Computer skills

C++	Modern C++, Conan, STL, Catch, Boost, OpenMP, Lemon, Qt, SFML, range-v3
Programming	Test-Driven Development, Data-Oriented Design, Object-Oriented Programming, Autotools, CMake, QMake, gcc, clang
Python	Matplotlib, Pandas, Scipy, NumPy, iPython Notebook, regex
Project Managment	Git, Github, Gitlab, Trello, SVN, Perforce, Travis-CI, Jira, CCMS
Other	Linux, Latex, Inkscape (vector graphics), Typescript, Javascript, PHP, Java, Node.js, Electron, NW.js

Languages

Portuguese	**Advanced**	*Native Speaker*
English	**Advanced**	*Can listen, read, write and communicate. Lived in the UK for 6 months.*
Spanish	**Moderate**	*Can listen and read.*

Education

2016 **Ph.D. Abandoned**, *Federal University of Santa Catarina*, Florianópolis – Brazil, *Computer Science*.

2014–2015 **Master's degree**, *Federal University of Santa Catarina*, Florianópolis – Brazil, *Computer Science*.

2010–2013 **Bachelor's degree**, *Federal University of Santa Catarina*, Florianópolis – Brazil, *Computer Science*.

Master thesis

title Exploiting Non-Critical Steiner Tree Branches for Post-Placement Timing Optimization

Rod. João Paulo - 435 – Florianópolis – Brazil
+55 (48) 99606 6659 • *csguth@gmail.com* • *csguth.com*

CG

advisor	Prof. Dr. José Luís Almada Güntzel
description	This work proposes and evaluates an incremental timing-driven placement (TDP) technique that moves a subset of cells to optimize the delay of the most critical interconnections in the circuit, while trying to preserve the initial placement quality. The technique explicitly models the interconnections as Steiner trees, which are able to capture information on the interconnection topologies in the final routing. The main contributions of this work are presented in [4].

EDA Skills

Industrial File Formats	Library Exchange Format (.lef), Design Exchange Format (.def), Liberty (.lib), Verilog (.v)
Modeling	Circuit, Interconnection, Timing-Graph
Interconnection	Delay Estimation, Elmore Delay, Effective Capacitance
Timing	Static and Incremental Timing Analysis
Optimization	Gate Sizing, Timing-Driven Placement
Legalization	On-the-fly Legalization, Tetris, Abacus

Contests and Awards

ICCAD'15	First Place in the Incremental Timing-Driven Placement (Problem C) Contest
ICCAD'14	Fifth Place in the Incremental Timing-Driven Placement (Problem B) Contest
ISPD'13	First Place in the Discrete Gate Sizing Contest

Interests

Programming	Opensource, Mobile, Games
Music	Electric Guitar, Music Production, Metal (Genre), Bad News Bad News (Band)
Crypto Currencies	Bitcoin, DECRED, *etc.*
Sports	Martial Arts (Karate, Taekwondo), Cycling

Publications

[1] Tiago Fontana, Renan Netto, Vinicius Livramento, Chrystian Guth, Sheiny Almeida, Laércio Pilla, and José Luís Güntzel. How game engines can inspire eda tools development: A use case for an open-source physical design library. In *Proceedings of the 2017 ACM on International Symposium on Physical Design*, pages 25–31. ACM, 2017.

[2] Tiago Augusto Fontana, Sheiny Almeida, Renan Netto, Vinicius Livramento, Chrystian Guth, Laércio Pilla, and José Luís Güntzel. Exploiting cache locality to speedup register clustering. In *Proceedings of the 30th Symposium on Integrated Circuits and Systems Design: Chip on the Sands*, pages 191–197. ACM, 2017.

[3] Chrystian Guth, Vinicius Livramento, Renan Netto, Renan Fonseca, José Luís Güntzel, and Luiz Santos. Timing-driven placement based on dynamic net-weighting

CG

for efficient slack histogram compression. In *Proceedings of the 2015 Symposium on International Symposium on Physical Design*, pages 141–148. ACM, 2015.

[4] Vinicius Livramento, Chrystian Guth, Renan Netto, José Luís Güntzel, and Luiz CV dos Santos. Exploiting non-critical steiner tree branches for post-placement timing optimization. In *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pages 528–535. IEEE Press, 2015.

[5] Vinicius Livramento, Renan Netto, Chrystian Guth, José Luís Güntzel, and Luiz CV Dos Santos. Clock-tree-aware incremental timing-driven placement. *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, 21(3):38, 2016.

[6] Vinicius dos S Livramento, Chrystian Guth, Jose Luis Guntzel, and Marcelo O Johann. Evaluating the impact of slew on delay and power of neighboring gates in discrete gate sizing. In *2012 IEEE 3rd Latin American Symposium on Circuits and Systems (LASCAS)*, pages 1–4. IEEE, 2012.

[7] Vinicius dos S Livramento, Chrystian Guth, José Luís Güntzel, and Marcelo O Johann. Lagrangian relaxation-based discrete gate sizing for leakage power minimization. In *2012 19th IEEE International Conference on Electronics, Circuits, and Systems (ICECS 2012)*, pages 468–471. IEEE, 2012.

[8] Vinicius S Livramento, Chrystian Guth, José Luís Güntzel, and Marcelo O Johann. Fast and efficient lagrangian relaxation-based discrete gate sizing. In *2013 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pages 1855–1860. IEEE, 2013.

[9] Vinicius S Livramento, Chrystian Guth, José Luís Güntzel, and Marcelo O Johann. A hybrid technique for discrete gate sizing based on lagrangian relaxation. *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, 19(4):40, 2014.

[10] Renan Netto, Tiago Augusto Fontana, Sheiny Fabre, Bernardo Ferrari, Vinicius Livramento, Thiago Barbato, João Souto, Chrystian Guth, Laércio Pilla, and José Luís. Ophidian: an open-source library for physical design research and teaching.

[11] Renan Netto, Chrystian Guth, Vinicius Livramento, Marcio Castro, Laércio Lima Pilla, and José Luís Güntzel. Exploiting parallelism to speed up circuit legalization. In *2016 IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, pages 624–627. IEEE, 2016.

[12] Renan Netto, Vinicius Livramento, Chrystian Guth, Luiz CV dos Santos, and José Luís Güntzel. Evaluating the impact of circuit legalization on incremental optimization techniques. In *Proceedings of the 29th Symposium on Integrated Circuits and Systems Design: Chip on the Mountains*, page 5. IEEE Press, 2016.

[13] Renan Netto, Vinicius Livramento, Chrystian Guth, Luiz CV dos Santos, and Jose Luis Guntzel. Speeding up incremental legalization with fast queries to multidimensional trees. In *2016 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, pages 36–41. IEEE, 2016.

CG