Evolution of Heat Flow Prediction Models for FPGA Devices

Hendrik Hangmann

November 24, 2014

1 Introduction for Heat Monitoring and Prediction for VLSI Devices

On today's integrated circuits there is a considerable potential of generating high temperatures, due to shrinking devices sizes and increasing frequencies. Especially FPGAs can generate local temperature hot spots and alarmingly high temperatures are reached within a small area. For many cases the accurate temperature monitoring and prediction is becoming increasingly important. Many FPGAs, like the Xilinx Virtex-5 or Virtex-6, provide the opportunity to measure the thermal effects of a circuit with the help of a built-in temperature sensor. As [5] pointed out, the outputs of the System Monitor, offered by Xilinx FPGAs, may be inaccurate. Hence [2] proposed a grid of several thermal sensors, consisting of ring oscillators and counters, rather than the single temperature diode. This approach uses the built-in temperature sensor in order to calibrate its sensors, as the correlation between measured temperatures and sensor frequencies is almost linear. Beside the temperature sensor grid, the system also contains heat-generating circuits. These heat-cores generate spatial thermal gradients up to $6,5°$C and are used to calibrate the sensors. Besides the importance of monitoring the temperature of VLSI, the prediction of temperature distributions may also be crucial. Especially for FPGAs, the obtained temperature models may lead to a pro-active thread remapping. In general there are two possibilities for temperature prediction. First, the temperature distribution can be predicted at design time, by regarding the die's structure and several layers. This approach will be introduced in Section 2. Furthermore, it is possible to predict the temperature distribution on-line at runtime, which may be crucial for reconfigurable FPGA devices. This approach will be introduced at Section 3.

2 Post-Fabrication Heat Flow Modeling Approaches

One popular approach to model heat flow on VLSI is to make use of the duality between thermal and electrical phenomena. HotSpot [3] for example is a tool, that simulate temperature of VLSI designs by modeling the die by an resistance capacitance (RC) network, which is an electric circuit consisting of (thermal) resistors and capacitors (heat

absorption capability). Many of the post-fabrication heat flow modeling approaches know precisely about several layers' properties of the chip, like the layers' thickness and temperature conductivity. Figure 1 depicts a simple RC network with only two layers, heat sources and a heat sink. Usually a chip contains about nine layers, such as heat sink, heat spreader, silicon bulk, interconnect layer, etc. Each of these layers is divided in to an arbitrary number of blocks, which are laterally and vertically interconnected. The temperature distribution is then derived out of the calculated vertical and lateral resistances.

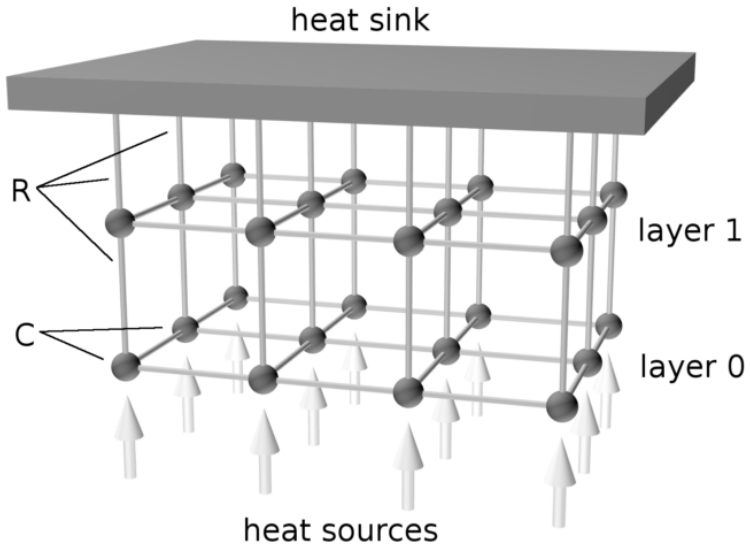

Figure 1: RC-Network with two layers [2]

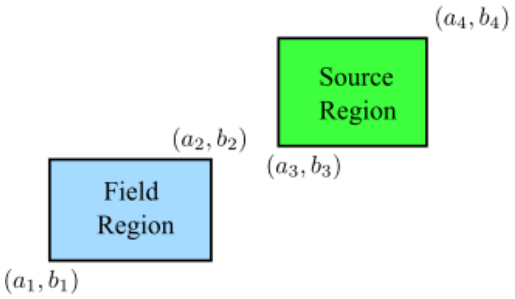

Figure 2: The temperature in the field region is calculated from the power of the source field [4]

Another approach for post-fabrication heat flow modeling is proposed in [4]. Just as HotSpot, this approach requires knowledge about the chip's structure, the layers' thickness and thermal conductivity. However, it does not model the thermal behavior as an RC network. On the contrary, the temperature distribution in a field region located between the chip's coordinates (a_1, b_1) and (a_2, b_2) is derived from the power source located between (a_3, b_3) and (a_4, b_4). Figure 2 depicts these regions located on the chip.

This approach faces the problem of solving the heat diffusion equation, which is the differential equation

$$\rho c_p \frac{\partial T(x,y,z,t)}{\partial t} = \nabla \cdot [k(x,y,z,T)\nabla T(x,y,z,t)] + g(x,y,z,t) \tag{1}$$

where ρ is the density of the material ($\frac{kg}{m^3}$), c_p is the specific heat ($\frac{J}{kg \cdot ^\circ C}$), T is the temperature ($^\circ C$) , k is the thermal conductivity ($\frac{W}{m \cdot ^\circ C}$), g is the volume power density ($\frac{W}{m^3}$) and (x, y, z) are the die's coordinates [4].

3 Goals of the Thesis

3.1 On-Line Heat Flow Modeling Approach

In many application areas of today's VLSI the on-chip temperature prediction is becoming increasingly important. The heat flow modeling approaches at design time introduced in Section 2 are solely suitable devices where the internal chip structure is known. This does not apply to all devices. The on-line heat flow modeling approach is based on this problem. In contrary to the post-fabrication heat flow modeling approaches, the on-line approach benefits from dropping the requirements of knowing the internal layer structure of a chip, such as heat conductivity. However, the proposed system [2] also takes advantage of an RC network. But, for the sake of performance, the accuracy is slightly decreased in this model. That is, the number of layers is reduced to two, as Figure 1 depicts. This trade-off reduces the complexity of the system in order to be executed efficiently on embedded devices. Furthermore, increasing the amount of layers only marginally improve the prediction, whereas a greater workload for the learning algorithm is generated.

The approach resorts to a black box heat and internal layer structure. The RC network's parameters are tuned online by a heuristic. That is, instead of having fixed parameters for the thermal model, the on-line approach contains a set of free parameters P, which are initialized randomly at the beginning of the learning process and improved at each step t_i. The improvement happens according to the deviation in the model's temperature prediction from the actual measured heat distribution. Technically, the thermal model is improved by comparing the actual temperature of the die $T_m(t_i, i)$ and the predicted temperature $T_s(P, t_i, j)$ for each node j. The learning is done by randomized hill climbing in this approach. The goal is the minimization of the mean square error mse dependent on the parameters P.

$$mse(P) = \frac{1}{|N||M|} \cdot \sum_{i \in M} \sum_{j \in N} (T_s(P, t_i, j) - T_m(t_i, j))^2 \tag{2}$$

3.2 Thesis Objectives

As proposed above there are several ways to model heat flow on integrated devices. The thesis gives an overview on heat models using physical device models and other post-

fabrication approaches like the RC-network. Furthermore the on-line extension of these models, which were briefly introduced above, will be explained in detail.

The main goal of the thesis is to improve the on-line approach of generating a heat model. Besides the general methodology all necessary approaches will be explained, like temperature generation and measurement and learning and optimizing algorithms, like Simulated Annealing and Evolutionary Algorithms. These are the parts where the improvement to existing approaches may occur. Learning and optimizing algorithms are used to redefine the model's set of parameters in order to achieve an efficient and precise heat model on-line. Each of these approaches will be tested and afterwards evaluated by discussing and comparing its effort for creating a the model and the accuracy of heat prediction.

4 Structure of the Thesis

The thesis will be structured as follows:

- **Introduction** Introduction to heat modeling and the reason why it becomes eminent for new many-core and reconfigurable devices
- **Heat Modeling of integrated Devices**
  - Heat Models using physical device models
  - Other Methods (e. g. RC- Networks)
  - Extension to On-Line Evolution of Heat Models
  - Temperature Generation Methods
  - Temperature Measurement Methods
    - Discussion on Accuracy
- **Online Evolution of Heat Models**
  - Methodology
    - Definition of Heat Model
    - Measurement modes and Accuracy
  - Learning and Optimzing Algorithms
    - Methodology of used Algorithms
- **Experiments and Evaluation**
  - Experiments on RC-network
  - Results for several Learning Algorithms
  - Discussion of Convergence
  - Discussion of Accuracy
  - Discussion of On-Line Suitability
  - Discussion of Embedded Implementation

- **Conclusion**

5 Time Schedule

Figure 3 depicts the intended time schedule of the thesis.

6 Thesis Organization

Besides the thesis' contents, there are several organizational tasks which will be performed during the thesis.

- **Kick-Off Presentation** The thesis' topic and time schedule will be introduced in a 15 minutes presentation.
- **Frequent Meetings** Every (second) week there will be a meeting with the supervisor, in order to keep track of the thesis' progress and potential issues.
- **Documentation** The thesis will be documented accurately. Care is given to clarity and completeness.
- **Final Presentation** Referring to the target agreement, the obtained results will be presented after the submission of the thesis.
- **Submission** The complete compilable and synthesizable code, experimental data, slides of both presentation and the thesis itself will be submitted in digital form.

References

[1] Andreas Agne, Hendrik Hangmann, Markus Happe, Marco Platzner, and Christian Plessl. Seven recipes for setting your FPGA on fire – A cookbook on heat generators. *Microprocessors and Microsystems*, 2013.

[2] Markus Happe, Andreas Agne, and Christian Plessl. Measuring and predicting temperature distributions on FPGAs at run-time. *Reconfigurable Computing and ...*, pages 55–60, 2011.

[3] Wei Huang and Shougata Ghosh. HotSpot: A compact thermal modeling methodology for early-stage VLSI design. *Very Large Scale ...*, 14(5):501–513, 2006.

[4] S.S. Sapatnekar and Yong Zhan. Fast Computation of the Temperature Distribution in VLSI Chips Using the Discrete Cosine Transform and Table Look-Up. *Proc. of Design Automation Conference (ASP-DAC)*, 1:87–92, 2005.

[5] Moinuddin a. Sayed and Phillip H. Jones. Characterizing Non-ideal Impacts of Reconfigurable Hardware Workloads on Ring Oscillator-Based Thermometers. *2011 International Conference on Reconfigurable Computing and FPGAs*, pages 92–98, November 2011.

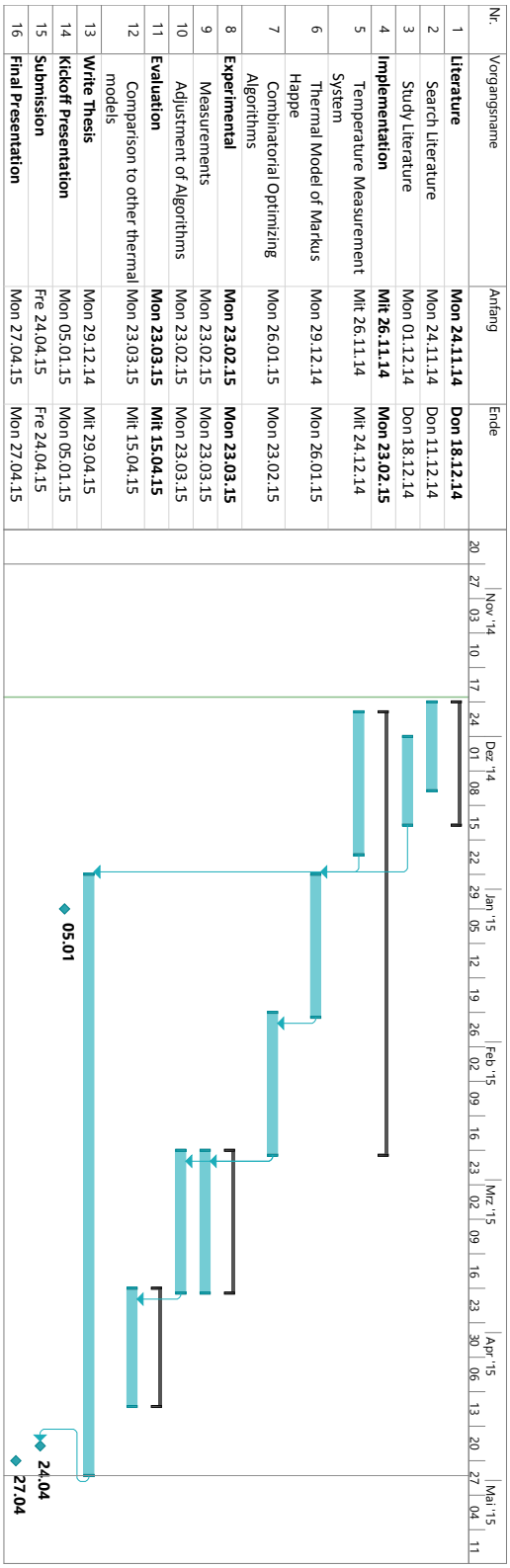

Figure 3: Time schedule