Placement and Routing of RF Embedded Passive Designs In LCP Substrate*

Mohit Pathak, Souvik Mukherjee, Madhavan Swaminathan, Ege Engin, and Sung Kyu Lim
School of Electrical and Computer Engineering
Georgia Institute of Technology
mohitp@ece.gatech.edu

Abstract

Physical layout generation of RF embedded passive design is not an easy task since the response of a given layout is tightly coupled with the response of the individual components and the effect of interconnect parasitics. In this paper we propose a methodology for automatic layout generation of embedded passive RF circuits. We make use of circuit models to represent and optimize a given layout and use non-linear optimization at various stages of the methodology to obtain the desired goals. Full-wave EM simulations is completely out of the design loop, so our methodology significantly reduces the design time for RF embedded passive circuits. The proposed approach has been used successfully to generate layout for band-pass filters of varying sizes.

1 Introduction

Passive elements are an important part of microelectronic devices. The number of passive components in handheld devices and computers is greater than 80% of the total part counts. Moreover, the passive to active ratio continues to grow [1]. Embedded passive is an emerging technology that has a potential for increased reliability, improved electrical performance, size shrinkage and reduced cost [2]. Using this technology, surface-mount passive components used in systems can be integrated into packaging substrate via multiple layers. However, the design of circuits with embedded passives is non-trivial due to the electromagnetic interactions that cause parasitics, leading to non-ideal frequency behavior. In this paper we target embedded passives using liquid crystalline polymer (LCP) substrate (see Figure 1). LCP is a low-loss material ($\tan\delta = 0.002$) with relative permittivity ($\varepsilon_r$) of 2.95. The material properties are invariant up to 20 GHz with negligible moisture absorption (0.04%). The process is also known to be low cost and low temperature [2]. Thus, LCP-based embedded passives promise high quality passives implemented in the packaging substrate.

Layout generation for RF embedded passive design is not an easy task. The desired response of a given layout is tightly coupled with the response of each of the individual components and the effect of parasitics due to interconnects between them. The manual design cycle for generation of such layouts can be extremely time-consuming. This is because the circuit response can be very sensitive to the parasitics of individual components and the interconnects. Thus, a minor change in the layout may cause a drastic change in the frequency response. A conventional design flow tries to optimize circuit performance at the layout level at the pre-

---

*This work was supported by the Mixed Signal Design Tools Consortium (MSDT) at the Packaging Research Center, Georgia Tech under project number 2126Q0R.
mium cost of time-consuming EM iterations (using a full-wave EM simulation tool like SONNET) for entire layouts.

Our goal in this paper is to develop a tool that quickly generates and optimizes RF embedded passive circuits in LCP substrate automatically. An early work [3] targets primarily CMOS technology, not packaging substrate such as LCP or LTCC. They do not describe a way of estimating parasitics for more than 2-pin nets. Such an approach would not work for RF embedded passive circuit since knowing the exact nature of interconnect parasitics is critical. Sommer et al. [4] presented a layout synthesis algorithm for embedded passive components such as capacitor, resistor and inductor. But, they did not discuss how to use them to construct an entire circuit. Mukherjee et al. [2] discuss another technique for automating the design of passive components. They also suggest how to analyze an entire embedded passive circuit, but their approach is limited to optimizing a given layout instead of constructing one.

Our contribution is a design methodology that automates the design process of LCP-based embedded passive designs from its circuit model to the layout. We make use of circuit models to represent and optimize a given layout and use non-linear optimization at various stages of the methodology to obtain the desired goals. Full-wave EM simulations is completely out of the design loop. Thus, our methodology significantly reduces the design time for RF embedded passive circuits. In addition, we provide the designer with an initial layout solution that closely matches the desired response. Since it is easier to tweak a given good solution to meet the desired goals rather than starting from scratch, our design flow can help reduce design times significantly.

2 Preliminaries

2.1 Problem Formulation

We assume that the following are given: (i) a netlist of the given RF circuit consisting of a set of passive components\(^1\) and nets, (ii) initial value of the \(k\) components, (iii) a set of \(r\) design goals\(^2\) to be achieved, (iv) a parameterized library that consists of inductors, capacitors, interconnects, and coupling models. The goal of Embedded Passive PHYSICAL DESIGN is to automatically generate a layout of the given circuit (= placement and routing of the components) such that the performance objectives are achieved and the area of the layout is minimized. Our target packaging substrate consists of two metal layers separated by a LCP layer for component placement (= minimum required to place a capacitor) and routing [2].

\(^1\)We restrict ourselves to inductors and capacitors for RF designs.
\(^2\)Typical goals include center/resonant frequency, bandwidth, etc.

2.2 Design Flow

Our method named EMplace is shown in Figure 2. The basic approach is to optimize the component placement using Simulated Annealing and component routing using maze-routing. We perform non-linear optimization at various steps of the layout generation process to meet the desired performance objectives while minimizing the area of the layout.

- **Step 1:** component shapes are chosen from the library based on their initial values.
- **Step 2:** these components are resized during our pre-placement optimization step by ADS (Advanced Design System by Agilent), a circuit-level simulator and optimizer. This is necessary to consider the effects of various component parasitics such as vias connecting to ground, wire connection from the core to boundary, etc. The geometric shapes change slightly during this step.
- **Step 3:** we perform placement and routing using the optimized components. Component coupling and wire parasitics are introduced during this step. The objectives during this step include layout area, wirelength, and routability. We save \(K_1\)-best layouts based on these objectives.
- **Step 4:** for each of the \(K_1\)-best layouts, we derive the circuit model and evaluate using ADS. The goal is to select \(K_2\)-best layouts that achieve the responses that are closest to our goal. The circuit model of the components and their coupling are extracted from the placement, whereas the wire and via parasitics are derived from the routing.
- **Step 5:** for each of the \(K_2\)-best layouts, we perform non-linear circuit optimization again using ADS for the entire layout. The components and wires are again re-sized during this step so that the overall circuit response best meets our goals.
- **Step 6:** The white space introduced during the post-placement optimization step is removed during our layout compaction step. The goal is to preserve the circuit response while optimizing the layout area. The component placement and routing change slightly during this step.

In case the final compacted layout passes our final circuit-level verification, we perform a full-wave EM simulation using SONNET for final verification. If not, we repeat the entire process starting from placement and routing. Compared to the traditional design flow where “manual placement and routing” and “EM simulation” are iterated, our
new design flow constructs a high-quality layout in a fraction of the time since the time-consuming EM simulation is out of the loop. Our final solution can be used for further manual touch-up if necessary.

3 Pre-layout Optimization

In this section we describe the details of pre-layout optimization. The objective is to find new component dimensions which meet the desired circuit objectives while considering intra-component parasitics. Consider Figure 3, where it shows the circuit model of a capacitor and an inductor cascaded together. The circuit elements in each component are grouped into two categories, namely, dominating elements, and parasitic elements. The dominating element values reflect the main inductance or the capacitance value of a given component.

During initial selection, the components were selected based on their dominating element values. Due to the effect of parasitics in each component, the initial selection does not always meet the desired circuit response. The goal of our pre-layout optimization is to find new components (possibly with new dimensions) such that they meet the desired circuit responses while considering both the dominating and parasitic elements. The dominating element values reflect the main inductance or the capacitance value of a given component.

During initial selection, the components were selected based on their dominating element values. Due to the effect of parasitics in each component, the initial selection does not always meet the desired circuit response. The goal of our pre-layout optimization is to find new components (possibly with new dimensions) such that they meet the desired circuit responses while considering both the dominating and parasitic elements. In addition to intra-component parasitics, additional parasitics such as vias connecting to ground, core-to-boundary elongation are also modeled. Due to these changes in the circuit model, the overall shape of the component after the optimization may change. Note that the optimization is applied not to individual components but to the circuit model of the entire design. Since the optimization is performed before placement and routing, the effect of interconnect parasitic and component coupling are not considered in this case.³

To optimize a given unplaced/unrouted circuit, we use Agilent’s Advanced Design System (ADS) engine to perform non-linear optimization. The basic idea is that during the optimization process, the parasitic element values of all components are fixed while the dominating element values are changing. The optimization process finds the optimal values of dominating elements. For every component with new dominating element value, we replace it using our library. Since the new component has different dominating value, the parasitics are also different in this new component compared to the old component. We then repeat the overall process until the response of the overall design meets our goal or the change in the response is minimal.

The pseudocode for preliminary resizing is shown in figure 4.

³Our post-layout optimization step does consider these layout-based parasitics.
the dominating element values as variables for the optimization engine. In line 4 we find the maximum and minimum value for dominating elements of each component based on the available values in library. The optimization engine is called in line 5. Line 6-7 updates the current solution based on the values (both dominating and parasitic elements) obtained from the optimization engine. The above process is repeated while the maximum change in dominating element values is below a threshold or the maximum number of iterations is reached.

4 Placement and Routing

The main objective of our placement and routing step is to find good candidate layouts which are optimized for area, wirelength, and routability. From a given candidate placement solution, we actually perform routing to accurately decide whether the given placement is routable. Since the routing resource is restricted to two metal layers, and the wire/via parasitics add significant parasitics to the overall design, wirelength and routing completion are very important goals. In addition, the area objective also play an important role in determining wirelength as well as coupling among the components.

The ultimate goal in our embedded passive layout is to meet the desired goals in terms of frequency responses, which can be judged by a circuit simulator such as ADS or even EM simulator such as SONNET. However, layout optimization using these tools is extremely time-consuming if not impossible. Thus, we sample “good initial layouts” that are optimized in terms of area, wirelength, and routability, and then choose the best one using ADS as the verification engine. As discussed in Section 2.2, we save the $K_1$-best layouts based on area, wirelength, and routability objectives. For each of the $K_1$-best layouts, we derive the circuit model and evaluate using ADS. The goal is to select $K_2$-best layouts that achieve the responses that are closest to our goal. Lastly, for each of the $K_2$-best layouts, we perform non-linear circuit optimization again using ADS for the entire layout. The components and wires are again resized during this step so that the overall circuit response best meets our goals. The layout that closely matches our goal and is the smallest becomes our final solution.

The dimensions of each component is obtained from our library based on their optimized dominating/parasitic element values. The placement is obtained and optimized using the well-known Sequence Pair [5] combine with Simulated Annealing. Routing is performed on each candidate placement solution, which is based on the well-known maze routing [6]. The main objective of this phase is to route all nets and reduce the overall wirelength. The nets are routed in the order of their decreasing weights, where the weight of a net is equal to the number of pins in the net. This method focuses on hard-to-route nets first. The routing graph is based on the channel intersection graph [7].

5 Post-layout Optimization

5.1 Overview

The goal of the place and route phase was to minimize the parasitic effects of interconnect so as to produce candidate solutions which may be close to the desired circuit response. In addition, adding white space around components helps reduce coupling between components. However, we note that the layout obtained from the placement and routing stage does not usually meet the desired. This occurs because of the considerable amount of (i) component parasitics, (ii) parasitics added due to interconnects and vias, and (iii) coupling among components and interconnects. The idea of post-layout optimization is to choose new components and their layouts such that the desired goals are achieved considering the parasitic effect of interconnects. This process usually introduces changes in the overall layout, and we attempt to keep the change small so that the overall structure and quality of the layout are preserved.

5.2 Resizing Algorithm

We perform post-layout optimization by using circuit models to represent the entire layout and then optimize it using ADS to obtain new component values which meet the desired goals. Since the initial layout is given, this circuit model not only includes the individual component-level models but also the interconnect wires, vias, and coupling information among wires and components. Such a circuit representation is not highly accurate but shows high fidelity and matches the response of a given layout closely. The use of circuit models is necessary since using a full wave EM-solver to optimize a given layout can be prohibitively time-consuming.

ADS-based circuit optimization causes the components to change their dominating element values, which in turn change the shapes. In order to prevent any drastic change in the layout during this resizing process—which may damage the initial placement and routing solutions—we perform the post-layout optimization in two steps.

- During the first step, we impose no restriction on the size of resized components. The new component sizes obtained may degrade the initial routing solution, so we perform placement compaction and rip-up-and-route on the affected nets. In this case, the Sequence Pair (= relative position among the components in the placement solution) remains fixed, but only the component dimensions and routing change.
Post-layout optimization

input: netlist $NL$, layout layouts $PR_i$, goals $G$
output: optimized layouts $PR_f$ with resizing

1. $num = 0$;
2. while ($num < max_itr$)
3.   generate $ckt$ from given $PR_i$ solution;
4.   optimize $ckt$ to meet $G$;
5.   compute new dimensions $c_{dim}^i \forall c_i \in ckt$;
6.   compute new $PR_i$ based on $c_{dim}^i$;
7.   $num++$;
8.   generate $ckt$ from $PR_i$;
9.   add dimension constraints $\forall c_i \in ckt$;
10. optimize $ckt$ to meet $G$;
11. find $PR_f$ based on new dimensions and $PR_i$;

Figure 5. Pseudocode for post-layout optimization algorithm

Figure 6. Impact of post-layout resizing and compaction

- During the second step, we ensure that a given routing solution does not change due to component resizing, which is achieved by adding restrictions on the allowed values for component resizing. The maximum allowed size for each component during optimization is chosen such that it does not alter the routing solution. In this case, only the connection from the core of the component to the pins located on the boundary will change. This minor change in the intra-component layout sometimes help improve the overall quality of the design.

The pseudocode of our post-layout optimization is given in Figure 5. Resizing based on fixed Sequence Pair, i.e., the first resizing step, is shown in lines (2-7). In each iteration, the circuit model is generated from a layout (line 3), the component values are recomputed based on current routing solution (line 4), and a new place and route solution is obtained based on new component dimensions (line 5-6). The entire process is repeated $max_itr$ times. Fixed routing resizing, i.e., the second resizing step, is shown in lines (8-11). The entire process is similar to the one described above except for line 9, where we add additional size constraints for component resizing. The constraints ensure that the component dimensions do not make a given routing solution invalid.

In both steps of the post-layout optimization, the size of each component typically reduces. Our post-layout optimization plus re-place and re-route basically performs layout compaction, where the whitespace among the components as well as intra-component whitespace are both removed, causing the overall layout are to reduce. Figure 6 shows an illustration of the impact of post-layout resizing and compaction.

Figure 8. Layout of the 8-component band-pass filter

Figure 9. Frequency response for the 8-component filter. The line with square, circle, and triangle respectively denote the desired, SONNET, and ADS response.
6 Experimental Results

We implemented our algorithms in C++/STL and ran our experiments on a Linux PC running 3GHz. We ran our algorithm on two bandpass filter circuits with varying numbers of lumped components. The response of the final layout for each circuit obtained was found by running SONNET on the final layout.

The initial circuit model for the 8-component RF bandpass filter circuit is shown in Figure 7. The final layout for the circuit is shown in Figure 8. The total runtime needed for this circuit was 1813 seconds. Figure 9 shows the S21 comparison. In this figure, the line containing squares shows the desired response. The line containing circles shows the SONNET response from our final layout. Lastly, the line containing triangles show the ADS response of the circuit model of our final layout. We see that our layout generates response that is very close to the desired response. We also see that the SONNET plot and ADS plot closely match, showing the accuracy of our circuit model.

The initial circuit model for the 12-component RF bandpass filter design is shown in Figure 10. The final layout obtained for the circuit is shown in Figure 11. Figure 12 shows the S21 comparison between the required initial response and the response of the final layout obtained. The total runtime needed for this circuit was 2546 seconds. We see that the mismatch between the desired response and our layout response becomes larger as the circuit complexity increases. In this case, a post-layout manual touch-up can improve the response of the layout. However, we emphasize that the
2546 seconds is considerably smaller than the time it usually takes to design a 12-component filter manually with the aid of EM simulation.

7 Conclusion

In this paper we described a methodology for automatic layout generation of RF embedded passive circuits. We made use of circuit models to optimize layouts and performed non-linear optimization at various stages of the methodology to obtain the desired goals. Full-wave EM simulations is completely out of the design loop, so our methodology significantly reduces the design time. We plan to improve our method by employing divide and conquer methodology in both layout and circuit analysis steps.

References


