TechRxiv
Powered by IEEE

# ViTaL: Verifying Trojan-Free Physical Layouts through Hardware Reverse Engineering

This paper was downloaded from TechRxiv (https://www.techrxiv.org).

LICENSE

CC BY 4.0

SUBMISSION DATE / POSTED DATE

09-11-2021 / 22-02-2022

CITATION

Ludwig, Matthias; Bette, Ann-Christin; Lippmann, Bernhard (2021): ViTaL: Verifying Trojan-Free Physical Layouts through Hardware Reverse Engineering. TechRxiv. Preprint.
https://doi.org/10.36227/techrxiv.16967275.v2

DOI

10.36227/techrxiv.16967275.v2

https://t.me/learningnets

# ViTaL: Verifying Trojan-Free Physical Layouts through Hardware Reverse Engineering

Matthias Ludwig*, Ann-Christin Bette*, Bernhard Lippmann*
*Infineon Technologies AG, Munich, Germany
{matthias.ludwig, ann-christin.bette, bernhard.lippmann}@infineon.com

***Abstract*—The semiconductor industry is heavily relying on outsourcing of design, fabrication, and testing to third parties. The threat of possibly malicious actors in this ramified supply-chain poses a risk for the integrity of integrated circuits (ICs) and hardware Trojans (HTs) are a heavily discussed topic in academia and the industry. A variety of pre- and post-silicon HT prevention and detection techniques has been suggested in prior works. Hardware reverse engineering has the potential to detect potential modification in physical layouts. Yet, there is no model to qualitatively and quantitatively rate the complex and expensive reverse engineering (RE) process addressing its inherent process aberrations and consequently provide a tool for layout verification. The *ViTaL* framework introduces a statistical validation technique, based on physical layout verification through RE and considers all potential sources of errors. The golden-model based framework is technology-agnostic, scaleable, and user input is optional. For the first time, results of fine pitch metallization layers of a CMOS 40 nm process node IC are presented quantitatively and the limitations and possibilities are discussed.**

***Index Terms*—hardware trust, hardware RE, physical layout verification, layout HT detection, design-for-manufacturability, feature extraction**

## I. Introduction

To sustain profitability in the investment-intensive semiconductor industry, companies are heavily depending on the outsourcing of design, fabrication, and testing to third parties. Threats like counterfeiting [1] or malicious modifications and insertions [2] during any design-stage are existent and potentially endanger trust, safety, and security. Mentioned modifications – i.e. hardware Trojans – can lead to information leakage, to a denial-of-service, or compromise the reliability of systems. At the defensive end of HTs pre- and post-silicon countermeasures and detection strategies have been introduced. Several detection strategies exist which rely on non-destructive testing methods of the specific changes in the *golden* foot-prints during IC testing. Delay and timing characteristics [3], power dissipation characteristic through laser voltage imaging [4], electro-magnetic emission profiles [5], or an examination of the logic outputs [6] can be tested statistically. Nonetheless, these methods do have the shortcoming of being unable to deal with extremely small HTs without any significant logic, analog, or electro-magnetic indication. In general, RE has the potential to unveil these HTs on the layout level. This has already been shown by previous research: In their most recent work, Bao *et al.* [7] have presented a robust way to identify HT-free ICs. Their method is based on feature extraction of geometrical properties of the layout which was reformulated to a clustering problem, completely independent of design parameters. Their theoretical set-up has been verified through publicly available benchmarks from *ISCAS89* [8] and *ITC99* [9]. These were synthesized with commercial tools and noise was synthetically added to simulate manufacturing and layout RE variations. This work can be extended by considering the difficulties of a *real* RE process beyond a simulation of the same. Another related work is the contribution from Vashishta *et al.* [10]. In their *Trojan Scanner*, computer vision algorithms for feature extraction are combined with a supervised machine learning model. In detail, they argue that a delayering of the active area (AA) from the back-side suffices to extract a unique descriptor for each type of standard cell. Via this approach, possible modifications are detectable on the AA. Nonetheless, an extension of this approach is possible when extending this work towards all layers with a layout foot-print. Similar approaches focusing on the idea of standard cell alterations are Courbon *et al.* [11] or Shi *et al.* [12]. In our prior works, a figure of merit for RE has been introduced [13] and scanning speed improvements were quantitatively shown based on this figure of merit [14]. Yet, a general way to verify the layout integrity via RE, considering all potential variances, that arise through manufacturing and the RE process is still pending. Based on this motivation, we introduce a framework for the **V**er**i**fication of **T**roj**a**n-free **L**ayouts (ViTaL) with following new contributions:

- A methodology for the qualitative and quantitative assessment of the performance of a hardware RE process for layout extraction is introduced which addresses all occurring distortions during the RE process.
- For the validation of a non-existence of malicious alterations, the framework is extended towards a post-fabrication verification tool. The framework avoids a traversing of the entire netlist extraction and only requires the steps until the extracted layout. We show the opportunities and limitations of the model to detect layout-

This work was partly funded by the German Ministry of Education and Research in the project RESEC under Grant No.: 16KIS1009. © 2021 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. DOI: 10.1109/PAINE54418.2021.9707702

https://t.me/learningnets

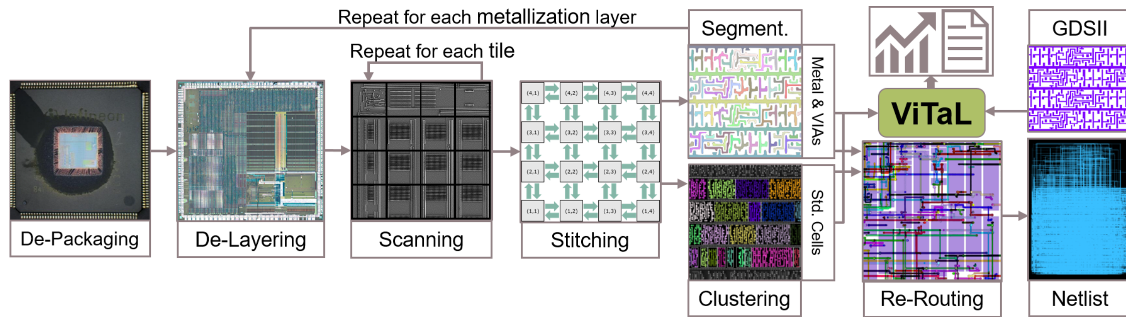



Figure 1: Process steps of an integrated hardware reverse engineering flow for integrated circuits. The *ViTaL* framework is shown as the verification tools after the image processing steps.

bound modifications and take IC fabrication, design-for-manufacturing, and RE induced errors into consideration.

- Finally, the RE process is performed on a 40 nm CMOS test sample and results of the golden-model verification are presented quantitatively.

## II. Background and Motivation

### A. Hardware Reverse Engineering

A circuit RE process is a complex semi-automated process which aims to recover the netlist of an analyzed device [15], [13], [16], or [17] for further abstraction. A full IC RE flow until the netlist generation, as shown in Figure 1, consists of the following sequential steps:
*1. De-Packaging:* The die is removed from its package.
*2. De-Layering:* The layers of the silicon die are sequentially de-processed via etching and polishing techniques.
*3. Image Scanning:* Each layer is scanned with an ultra-high resolution scanning electron microscope (SEM).
*4. Image Stitching:* A geometrically-undistorted mosaic is reconstructed via overlapping, adjacent tiles.
*5. Image Processing:* All metallization layers are vectorized by image processing techniques. The repeatedly placed standard cells are clustered and interpreted manually per class.
*6. Netlist Re-Routing:* Standard cells are connected by assigning overlapping conductive tracks for generation of a netlist.
*7. Netlist Abstraction:* Abstraction is done, for instance, by structural graph-analysis [18] or structural block analysis [19].

This multi-stage process requires experienced analysts on all parts of the process. Additionally, the requirements regarding the specialized RE hardware and software are extremely high.

### B. Attack Scenario

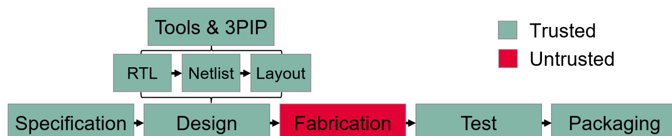



Figure 2: Attack scenario for manufacturing time attacks at an untrusted foundry.

The general design flow for integrated devices is shown in Figure 2. In the attack scenario covered by *ViTaL*, malicious alterations are implemented in the fabrication phase, while all other steps are assumed trustworthy. The industrial relevance for this attack is given: For modern ICs, the fabrication is conducted in possibly non-trustworthy foundries distributed across the globe. Also, from an academic point of view it is assumed that untrusted foundries are the most likely adversaries [12]. There are two ways for an insertion of alterations during this stage: First, through a polygon-based manipulation of the design files. Second, the photomasks can be manipulated through electron-beam lithography. Sophisticated players with extreme capabilities – regarding expertise and finances – could be able to execute such insertions, beyond changing random geometries on the layout. The difficulty of an insertion is followed by a difficulty of detection [20]: Physical characteristics of an altered layout can be distinguished via distribution and size, and might be extremely tiny. A further possibility of distinction is the type of HT inserted. Layout-bound HTs can be either functional or parametric [21]. Parametric HTs can pose a real threat: for instance, the random behavior of physical unclonable functions (PUFs) could be compromised by tiny layout changes.
An appropriate RE-based golden model must meet following characteristics: The first, by far most important, is that modifications in the layout are reliably recognized as such. The second has a practical reason. In case of an over-sensitivity, the manual detection effort is expected to become unrealistically high. All discussed challenges must be addressed adequately, so that the optimal trade-off can be reached. To our best knowledge, academic literature does not provide models which link the existing difficulties of design-for-manufacturability (DfM) measures, manufacturing variances, and RE process distortions with this verification task.

### C. Challenges

These issues are now outlined in detail.
**Design-for-Manufacturability and Manufacturing Variances.** It could be argued that after extracting the layout, checking the layout integrity could be possible with conventional EDA tools (i.e. layout XORing, layout Diff.). Yet, such a comparison would be useless: Even before manufacturing of an IC, layout post-processing for yield enhancement is carried out. Normally, the owner of the original GDSII layout does

https://t.me/learningnets

not have access to the post-processed mask data. The post-processed design files represent the actual layout imprinted and have a visible divergence to the GDSII. Where applicable, the distance between some of the tightly packed design objects is relaxed on the layout as measure of DfM. Another option is the widening of power lines which can potentially double the polygon areas compared to the original design file. For technology nodes of $130\,\mathrm{nm}$ [22] and below, additional processing in the form of optical proximity correction is necessary which might have further impact on the layout if the underlying models are not perfect. In addition, the manufacturing process itself has an influence on the appearance of the layout. Normal variances during manufacturing (e.g. etching rates) influence the eventual physical layout and will vary between dies, wafers, and lots.

**RE Process Variances.** As discussed, the design layout and the physical layout will have discrepancies. Yet, the potential distortions the layout will yield after the physical RE process may be even bigger. This error function is shown in Figure 3: Starting from the decapsulated IC, the defects accumulate during the intermediate processes. Preparation tends to be the most error-prone, with extreme requirements towards the analysts. Besides violating the tight planarity requirements, particles, dents, or scratches are a big issue. Over- or under-etching is another source for imperfectly processed layers. In extreme cases, vertical interconnect accesses (VIAs) or metal lines may be completely dislodged. During scanning, distortions occur through inadvertent charging of the probe or, and even more harmful, intra-image distortions and drifting appearances render further processing hard. An improper selection of detectors, a sub-optimal mixing of multiple detectors, or simply inadequate scanner settings for a specific sample complicate the image processing. Finally, image processing has immense requirements towards analysts and the software. Geometrical distortions during 2D stitching may hinder the correct layout extraction. Widely available image processing tools and algorithms are not suited and more sophisticated segmentation techniques are mandatory. All these steps lead to an imperfectly extracted layout and propel the manual RE effort towards the extreme. Consequently, these factors must be taken into account for a layout verification model.

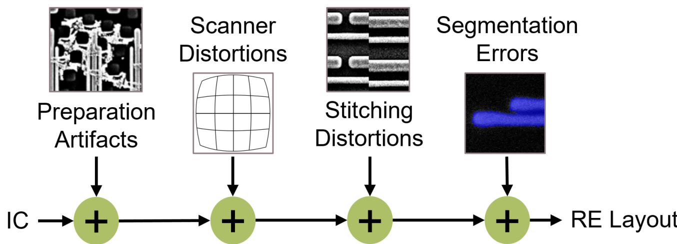



Figure 3: Simplified error function of the reverse engineering process, which accumulated from the decapsulated integrated circuit to the extracted layout. Examples of potential errors are visualized respectively.

## III. Methodology – The ViTaL Framework

In the following section, the golden model approach is introduced. The basic idea is shown in Figure 1 in the top right corner: The framework shows the superposition of two layouts.

### A. Pre-Requisites

*a) Notations:* Following notations will be used: the two layers are denoted $R$ for the reverse engineered layout, $D$ for the design layout. $P_D, P_R$ denote a set of polygons and $p_D, p_R$ describe single polygons of the respective layer. $f$ denotes a numeric feature value of two polygons, $F$ denotes a matrix of extracted features, and $X$ indicates a list of paired polygons.

*b) Layer Alignment:* First, the reverse engineered layout data is written to a common polygon format. For the ensuing comparison, the heavily nested GDSII file is parsed into the same polygon format. This can be done through the algorithms by Singla *et al.* [23]. Initially, both layers cannot be compared: they are completely un-aligned regarding scale, rotation, and translation. The alignment of the two layouts is done through a semi-automated process: on both layers, a minimum of three *equal* polygons are selected. Via the corresponding coordinates, a transformation matrix [24] is calculated. The affine transformation is executed on every polygon of the design layer and results in an approximate overlay as shown in the excerpt of Figure 4a.

*c) Scoring Figures:* To rate the process quantitatively, figures for a subsequent rating are necessary which have initially been introduced in [14]. Three generic types of extracted polygons can be observed in the layout: First, a *match* between extracted and design polygons (true positive, `TP`). Second, polygons that are only existing in the reverse engineered layout (false positive, `FP`). Finally, design polygons that are not represented by an extracted polygon (false negative, `FN`). From these generic classes, a binary classification and statistical figures [25] can be derived:

$$\text{Precision: } P = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{1}$$

$$\text{Recall: } R = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{2}$$

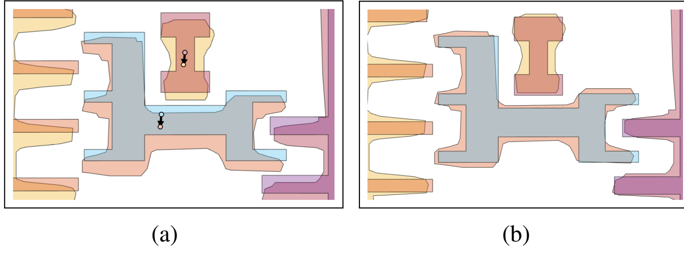
$$F_1 \text{ Score: } F_1 = 2 \cdot \frac{P \cdot R}{P + R} \tag{3}$$

The mapping of the polygons into these specific classes is outlined accordingly.

*d) Feature Extraction:* To determine these classes, adequate polygon-based feature descriptors are necessary to describe the similarity and the feature set of [7] is extended. All possibilities of modified polygons must be assessed. For a pair of polygons (e.g. Figure 4b), it is possible that equal centroid (C) coordinates yield completely different areas. Also, if the area (A) and centroid are close to equal, the polygons can still have a big discrepancy regarding their aspect ratio. Malicious modifications could be specifically trimmed if these features are not carefully selected.

Consequently, three inter-polygon features are defined: The

https://t.me/learningnets



(a) (b)

Figure 4: (a) Overlay of design and reverse engineered layout before translation. The vectors of the translated centroids illustrate the existing shift. (b) is the overlay of both polygons after translation.

Euclidean distance between two centroids $f_C$, the difference of the aspect ratios $f_{AR}$ of two bounding boxes (BB), and the relative difference of the area intersection $f_A$ form a robust model to detect discrepancies between two polygons in the two layers.

$$f_C = \parallel C(p_D), C(p_R) \parallel \tag{4}$$

$$f_{AR} = \frac{\mid \mathrm{BB}_x(p_R) \mid / \mid \mathrm{BB}_y(p_R) \mid}{\mid \mathrm{BB}_x(p_D) \mid / \mid \mathrm{BB}_y(p_D) \mid} \tag{5}$$

$$f_A = \frac{A(p_R) \cap A(p_D)}{A(p_D)} \tag{6}$$

### B. The ViTaL Algorithm

*a) Initial Polygon-Pair Assignment and Translation:* After the alignment of *R* and *D*, it cannot be assured that no single tiles are geometrically shifted due to stitching aberrations (see Figure 4a). The compensation of this aberration – preceded by an initial polygon-pair assignment – can be described as follows: For all polygons $P_R$ (count: $n_i$) and $P_D$ (count: $n_j$) of a single tile, the centroids are calculated. The layer with lesser polygons is defined as the codomain, the other as the domain. If $n_i = n_j$ hold true, *D* is selected as codomain. For the following steps, it is assumed $n_i < n_j$. The Euclidean distance between the centroids of $P_D$ to all centroids of layer $P_R$ is computed $f_C(p_{D_j}, p_{R_i}); \forall\, i \in n_i,\ j \in n_j$. The minimum distance of every vector ($P_{R_i}$) of the matrix yields the best-matching polygons for both layers and a pair-wise mapping between both layer exists: $X_1 : P_R \rightarrow P_D$. Polygon indices of *D* which are not present in the list can be omitted beforehand. Next, the median absolute deviation (MAD) is calculated for $X_1$:

$$\varepsilon = \mathrm{median}(\mid f_{C_{R_i,D_j}} - \mathrm{median}(F_{C_{R,D}}) \mid) \tag{7}$$

Outlying polygon-pairs above the MAD are removed from the mapping $X_1$. For a compensation of the aberration or the average translation in this tile, the mapping can be utilized. The average translation vector is defined as the median translation between the centroids of the polygon-pairs in $X_1$: $\vec{T} = \mathit{median}(F_C(X_1))$. The error is adjusted through a translation of $-\vec{T}$ of $P_R$ and can be seen in Figure 4b.

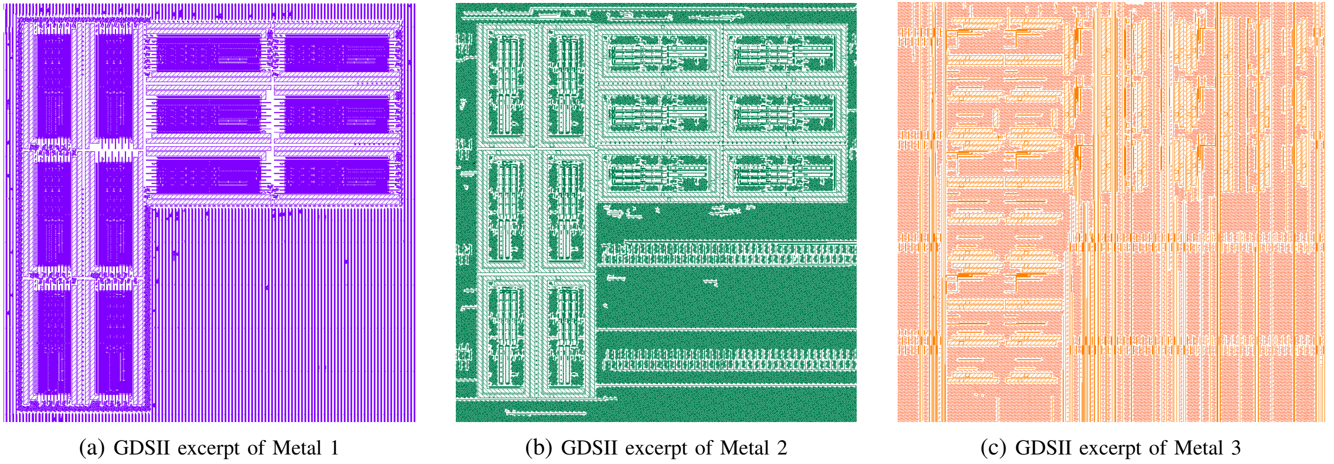
This method yields two advantages: an improved subsequent processing for the binary classification, since errors, which are not relevant for the actual layout extraction, are eradicated, and the possibility to report the stitching performance and possible aberrations (see Figure 9).

**Algorithm 1** Simplified algorithm of the *ViTaL* framework.

1: **for** $m$ to NumTiles **do**
2: TileCoords $\leftarrow$ GetTileCoords($m$)
3: [$P_D$, $P_R$] $\leftarrow$ GetTilePolygons(TileCoords)
▷ Feature extraction 1:
4: **for** $i$ to $n_i$ **do**
5: **for** $j$ to $n_j$ **do**
6: $f_{C,i,j} \leftarrow$ ExtractFeatures($p_{D_j}$, $p_{R_i}$)
7: **end for**
8: **end for**
▷ Initial assignment and translation:
9: $X_1 \leftarrow$ GetPolygonPairings($F_C$)
10: $\varepsilon \leftarrow$ CalculateMAD($X_1$)
11: $X_1 \leftarrow$ UpdatePairs($X_1$, $\varepsilon$)
12: $\vec{T} \leftarrow$ GetAverageTranslation($X_1$)
13: ReportStitchingError($\vec{T}$)
14: $P_R \leftarrow$ TranslatePolygons($P_R, \vec{T}$)
▷ Feature extraction 2:
15: **for** $i$ to $n_i$ **do**
16: **for** $j$ to $n_j$ **do**
17: $f_{C,i,j} \leftarrow$ ExtractFeatures($p_{D_j}$, $p_{R_i}$)
18: **end for**
19: **end for**
▷ Final assignment and scoring:
20: $X_2 \leftarrow$ GetPolygonPairings($F_C$)
21: $\varepsilon \leftarrow$ CalculateMAD($F_C$)
22: $X_2 \leftarrow$ UpdatePairs($X_2$, $\varepsilon$)
23: $F_C, F_{AR}, F_A \leftarrow$ ExtractFeatures($X_2$)
24: $\varepsilon_C, \varepsilon_{AR}, \varepsilon_A \leftarrow$ CalculateMAD($F_C, F_{AR}, F_A$)
25: $X_2 \leftarrow$ UpdatePairs($X_2, F_C, \varepsilon_C, F_{AR}, \varepsilon_{AR}$, $F_A, \varepsilon_A$,[DfM])
26: ReportScores($X_2, n_i, n_j$)
27: **end for**

*b) Final Assignment and Scoring:* Based on the translated polygons, the mapping ($X_2 : P_R \rightarrow P_D$) is calculated through the procedure introduced in III-B0a. For a final assignment, a centroid difference below a certain threshold is a necessary condition, yet not sufficient. Differences in area intersections and varying aspect ratios (Eq. 4) are the other features for a certain assignment. For all pairs in $X_2$, the features ($F_C, F_{AR}, F_A$) are extracted and the MAD is calculated respectively. Based on this threshold outlying pairings are removed if a single feature reports an outlier. As discussed in Section IV-C, a manual adjustment of these thresholds is needed to deal with technology-specific DfM measures. Since these factors are not technology-agnostic, the threshold will vary and it must be manually adjusted. A good practice is to run the procedure twice. During the first run, no *DfM* factor should be selected, so that the results will be purely based on statistics. Through manual assessment of the scores, the user can adapt individual thresholds for a run with *DfM* measures. In a last step, remaining outliers are assigned to `FNs` ($P_D$)

https://t.me/learningnets



(a) GDSII excerpt of Metal 1 (b) GDSII excerpt of Metal 2 (c) GDSII excerpt of Metal 3

Figure 5: Physical layout of the evaluated test sample.

TABLE I: Properties of the investigated layers of the 40 nm TD and the corresponding RE settings. The settings which are identical for all three layers are: Scanner type: *Raith CS150 Two*; Detector: *ET-SE*; Field of view: 16.0 $\mu$m; Pixel Size: 4.0 nm (square); Pixel dwell time: 6.0 $\mu$s; Image resolution: 16 Mpx (4000 x 4000); Bit depth: $2^8$; Grid: *12 x 12*.

| | Scanner Settings | | Image Processing | | | |
|---|---|---|---|---|---|---|
| **Layer** | Acc. Volt. | Focus | Stitching Method | Pre-Processing | Segmentation Method | Post-Processing |
| **Metal 1** | 3.5 kV | 4.96 mm | NCC (local), MLE (global) | Median Filter (1 x), Kernel: 5 x 5 | Edge Detection, gradient oriented flood-fill | Vertex simplification, Deletion (<70 $\text{Px}^2$) |
| **Metal 2** | 7.0 kV | 4.95 mm | NCC (local), MLE (global) | Median Filter (7 x), Kernel: 5 x 5 | Thresholding | Vertex simplification, Deletion (<100 $\text{Px}^2$) |
| **Metal 3** | 7.0 kV | 5.00 mm | NCC (local), W-LMS (global) | Median Filter (3 x), Kernel: 5 x 5 | Edge Detection, gradient oriented flood-fill | Vertex simplification, Deletion (<100 $\text{Px}^2$) |

and FPs ($P_R$). The length of the assignment matrix yields the number TPs. The absolute numbers define the statistical figures, which can be derived now.

The discussed model is summarized in Algorithm 1. The overall time-complexity scales with $O(n_i \cdot n_j) \approx O(n_j^2)$ per tile. Processing time is influenced by the number of polygons per tile which in turn is impacted by the field of view, the technology node, and the complexity of the extracted polygons. The presented golden-model is completely free of any *magic* threshold in the *normal* mode. The optional DfM threshold is a potential user input for a user *assisted* mode. Furthermore, it does not require any machine learning steps which frequently introduce uncertainties or aggravate interpretability.

## IV. Case Study – Results of a 40 nm device

### A. Test Device in Detail

In this section, the theoretically introduced figures are demonstrated on a test device (TD). The TD has been manufactured in an ultra-low-$\kappa$ CMOS 40 nm process. The metallization stack (Copper) consists of 7 metal layers. The test area for evaluation is approx. 190 µm x 190 µm or in total $36.1 \times 10^3\,\mu\text{m}^2$. We limit the layers to metal 1, 2, and 3 (M1, M2, M3). These first layers of the metallization have the closest pitches and far tinier vertical dimensions than the upper metal layers. Also, processing the highly porous ultra-low-$\kappa$ materials is more challenging than ordinary dielectrics.

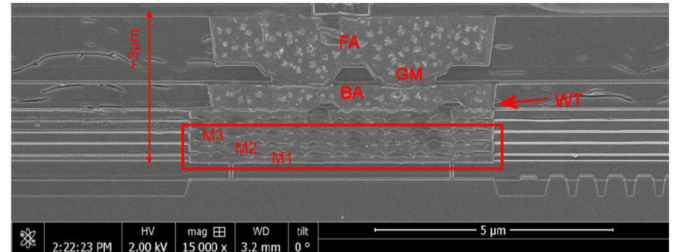
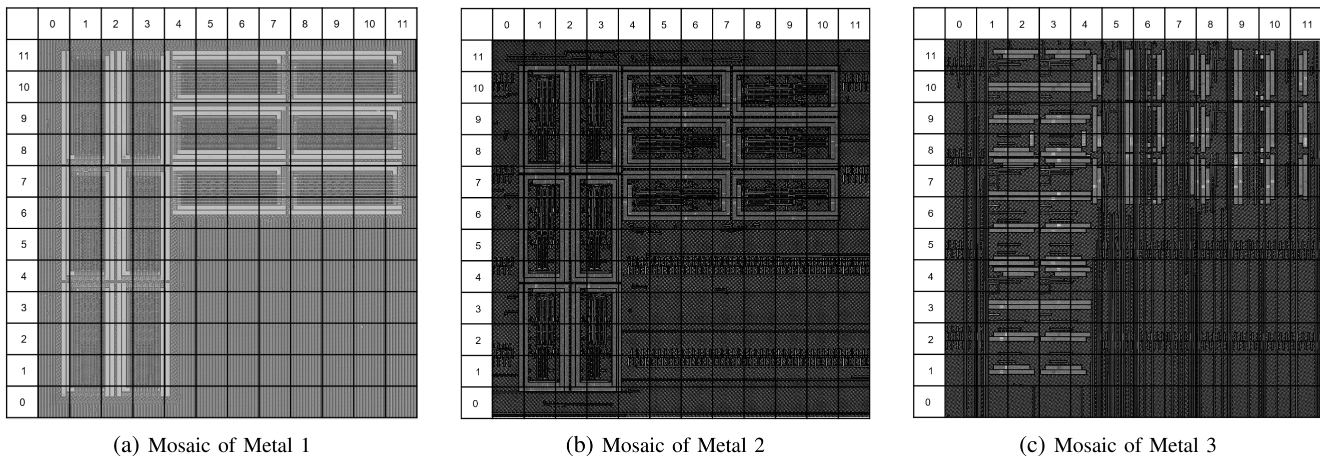



Figure 6: SEM cross-section of the 40 nm test IC.

### B. RE of the Test Device

De-processing starts with a removal of the package material through wet chemical etching with sulfur oxide or other eligible acids. To get an improved overview of the technological features, a mechanical cross-section is produced (see Figure 6). The resulting SEM cross-section image shows M1, M2, and M3 with vertical dimensions of the three layers varying approx. between 150 to 180 nm. The ROIs of the TD are shown in Figure 5, wherein a distinctive, rotated *L*-pattern is visible. In the bottom right corner and the small surrounding parts, only dummy fill patterns are placed. The vertical dimensions

https://t.me/learningnets




(a) Mosaic of Metal 1 (b) Mosaic of Metal 2 (c) Mosaic of Metal 3

Figure 7: Scanned layers of the 40 nm test sample. The grid indicates the individual SEM scans.

TABLE II: Quantitative results of the tested layers. The binary ratios are distinguished between the a technology-agnostic assessment (without *DfM*) and an assessment that takes the TD-specific *DfM* measures into account.

| | Without *DfM* | | | With *DfM* | | | No. of polygons | |
|---|---|---|---|---|---|---|---|---|
| **Layer** | F1 Score(%) | Precision(%) | Recall(%) | F1 Score(%) | Precision(%) | Recall(%) | GDS | RE |
| **Metal 1** | 92.54 | 89.55 | 95.73 | 95.70 (+3.18) | 92.63 (+3.08) | 99.03 (+3.30) | 30,386 | 32,486 |
| **Metal 2** | 95.00 | 95.13 | 94.87 | 97.92 (+2.92) | 98.05 (+2.92) | 97.79 (+2.92) | 46,429 | 46,302 |
| **Metal 3** | 97.16 | 97.15 | 97.18 | 99.66 (+2.50) | 99.64 (+2.49) | 99.67 (+2.49) | 44,680 | 44,693 |

set the boundaries for the following delayering process. Over the complete ROI, the maximum planarity variance must be kept far below 150 nm. Problems like warpage or total thickness variations must be addressed accordingly. The de-layering of the sample was done through the process introduced in [13]: An application of alternating etching techniques and testing procedures is required. Anisotropic plasma etching is performed for removal of the dielectrics (inter-layer oxides and passivation). Plan-parallel mechanical polishing with a grained diamond suspension is applied to the conducting materials. To improve scanning, it can be of advantage to leave a small remaining di-electric layer over the conducting metal lines. Finally, the IC is de-processed to the active regions which are treated with hydrofluoric acid. All steps are combined with optical and metrological machinery for assurance of the delayering quality. Still, this part of the RE process is often called an *art* [26] and requires years of experience and expensive specialized equipment. An overview of the properties of the remaining RE steps for all three layers is listed in Table I. Some of these settings require a more detailed explanation: For the image stitching a local registration is the first step necessary, where all adjacent tiles are aligned accordingly. A template matching in the spatial domain via a normalized cross-correlation (NCC) yielded adequate results. For M1 and M2, a maximum likelihood estimation (MLE) and for M3 weighted least mean square (W-LMS) were the respective optimal solutions for the global registration. The stitched mosaics of the three layers are depicted in Figure 7. Layout extraction was conducted through a three step approach: Pre-processing only required median filtering over all three layers. Image segmentation was done with simple thresholding for M2. M1 and M3 needed an edge detection approach with a subsequent flood fill which direction was determined by filling the areas with the lower gradients. In the post-processing step, only a vertex simplification and a deletion of objects below a certain area boundary was conducted.

### C. ViTaL Results

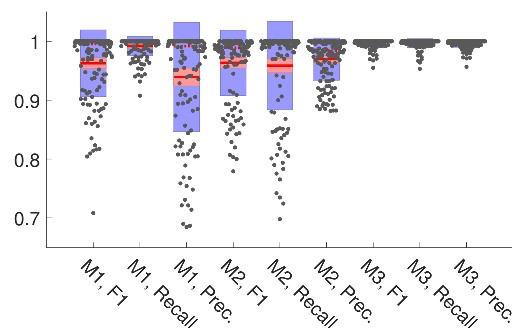



Figure 8: F1 score, precision, and recall of the three evaluated layers. The scores of the individual tiles are shown *with DfM* measures in grey. The *mean* is indicated in red, while the blue boxes indicate the *standard deviation*.

The experimental results of the RE performance are summarized in Table II. The binary statistics, with and without addressing DfM measures are outlined for the TD. Over every layer an F1 score of over 90% has been reported

https://t.me/learningnets

which was increased through a DfM-specific adjustment. DfM measures where manually addressed by an adjustment of the accepted area difference (power line widening) which was altered during layout post-processing.
The average score is not the only figure of importance. Single tiles yielding massive outliers can render an analysis useless. Therefore, the binary scores of the individual tiles are shown in Figure 8. While M3 had an excellent overall yield, with almost all tiles close to perfect scores, M1 and M2 had some outlying tiles. More specifically, the precision of M1 is impacted. The root cause are mostly tiny particles or dents on the layer. Subsequently, these can be deleted either manually or automatically via special particle deletion algorithms [14, p.12]. A bigger issue is the low recall of M2. Consequently, the `FN`s remain in the layer and these cannot be rejected automatically. Regarding this layer, the recall of 97.79% leads to 1,026 polygons to be checked manually. In the presented example, the root-cause mostly originated from errors during the preparation.

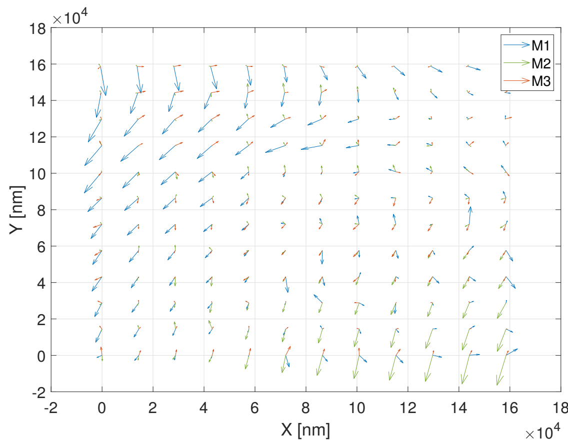



Figure 9: Accumulated distortions between the centroids of assigned polygon-pairs per tile. The vector lengths are not true to scale. The bottom right green vector yields a length of ∼65 nm. All other vectors are scaled in relation to this vector.

The per tile deviation compared to the ground truth is illustrated in Figure 9. The average aberration after stitching of M1 is $19\,\mathrm{nm}$ with a maximum of $56\,\mathrm{nm}$, for $\mathrm{M2}_{Avg.}$ $13\,\mathrm{nm}$, $\mathrm{M2}_{max}$ $65\,\mathrm{nm}$, and for $\mathrm{M3}_{Avg.}$ $8\,\mathrm{nm}$, $\mathrm{M3}_{max}$ $18\,\mathrm{nm}$. It should be remembered that the TD is a $40\,\mathrm{nm}$ technology node wherein the minimum distance of M1, M2, and M3 is $70\,\mathrm{nm}$. Consequently, the deviations of this example lead to the situation that an automated generation of the netlist is not possible. The layers M1 and M2, connected through VIA1 will – in certain areas – not coincide in the re-routing process. Here the advantage of a pure layout-based verification is obvious: even larger stitching errors can be compensated and a layout verification is still possible.
The overall time per tile for the given TD was on average $5\,\mathrm{Mins}$ on a Intel® Core™ i7-8700 CPU with a memory requirement of $4\,\mathrm{GB}$. Assuming a total area of $1\,\mathrm{mm}^2$ was examined, the cumulative amount is 3,600 tiles. Given this area, the time span will approximately be $12.5\,\mathrm{days}$ which is easily improvable by parallelization.

## V. Discussion

The strength of *ViTaL* is obvious: Due to the potentially extreme number of polygons in a ROI, the manual effort can be decreased to an absolute minimum. The remaining work is a check of remaining `FP`s and `FN`s for a integrity validation. Potential layout modifications can be detected with the optimal trade-off between certainty and decreased manual effort. Besides the improved time and financial aspects, the advantages are clear: The golden model allows an integrity verification without execution of a full RE flow. Yet, a full layout extraction is only feasible on a very limited number of dies. Thus, when only a small number of shipped devices are maliciously modified, they will evade detection with a high probability. The model is approachable on all layers with a layout foot-print. Consequently, the feasibility of an evaluation of gate lengths and widths or on the contact and VIA layers is given. Yet, HTs implemented during manufacturing are not limited to layout modifications. Malicious, local alterations of dopant concentrations or changed etching rates could affect the properties of ICs and possibly circumvent detection strategies via imaging. A first approach has been shown by Becker *et al.* [27] and reversed by Sugawara *et al.* [28].

## VI. Conclusion

We have introduced a comprehensive, technology-agnostic flow for layout integrity verification based on a RE process. The golden-model incorporates the robustness necessary to deal with DfM measures, manufacturing, and reverse engineering process variances. *ViTaL* is a novel tool for trust validation via RE processes and supports subsequent netlist generation and RE process stability. Our experiments have shown these capabilities on a $40\,\mathrm{nm}$ test IC. Additionally, with the whole approach being technology-agnostic, also PCBs or other devices with physical layouts can be evaluated through this approach. The framework is a step towards reproducible and expressive layout verification and RE process benchmarking.

## VII. Future Research

Several questions for future research that have not been addressed in this work are open:

- Strategies for the detection of non-layout-bound modifications in given attack scenario which may evade imaging strategies are a task for future research.
- For the delayering of large areas ($\gtrsim 1\,\mathrm{mm}^2$) on ever-shrinking nodes, innovative preparation, scanning, image processing, and netlist abstraction techniques are needed.
- For netlist generation and overall cost saving, SEMs often are the bottleneck. Aberration-free scanning in minimum time-spans is of high importance. Optimizing RE towards maximum field of views with minimum dwell times is a vital task for shrinking technology nodes.

https://t.me/learningnets

## REFERENCES


[1] M. M. Tehranipoor, U. Guin, and D. Forte, *Counterfeit Integrated Circuits: Detection and Avoidance*. Springer Publishing Company, Incorporated, 2015.

[2] S. Bhunia and M. M. Tehranipoor, *The Hardware Trojan War: Attacks, Myths, and Defenses*, 1st ed. Springer Publishing Company, Incorporated, 2017.

[3] B. Cha and S. K. Gupta, "Trojan detection via delay measurements: A new approach to select paths and vectors to maximize effectiveness and minimize cost," in *2013 Design, Automation Test in Europe Conference Exhibition (DATE)*, 2013, pp. 1265–1270.

[4] U. Kindereit, G. Woods, J. Tian, U. Kerst, R. Leihkauf, and C. Boit, "Quantitative investigation of laser beam modulation in electrically active devices as used in laser voltage probing," *IEEE Transactions on Device and Materials Reliability*, vol. 7, no. 1, pp. 19–30, mar 2007.

[5] F. Stellari, P. Song, A. J. Weger, J. Culp, A. Herbert, and D. Pfeiffer, "Verification of untrusted chips using trusted layout and emission measurements," in *2014 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, 2014, pp. 19–24.

[6] H. Salmani, M. Tehranipoor, and J. Plusquellic, "A novel technique for improving hardware trojan detection and reducing trojan activation time," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 20, no. 1, pp. 112–125, 2012.

[7] C. Bao, D. Forte, and A. Srivastava, "On reverse engineering-based hardware trojan detection," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 1, pp. 49–57, jan 2016.

[8] F. Brglez, D. Bryan, and K. Kozminski, "Combinational profiles of sequential benchmark circuits," in *IEEE International Symposium on Circuits and Systems*. IEEE, 1989, pp. 1929–1934 vol.3.

[9] F. Corno, M. Reorda, and G. Squillero, "Rt-level itc'99 benchmarks and first atpg results," *IEEE Design & Test of Computers*, vol. 17, no. 3, pp. 44–53, 2000.

[10] N. Vashistha, H. Lu, Q. Shi, M. T. Rahman, H. Shen, D. L. Woodard, N. Asadizanjani, and M. Tehranipoor, "Trojan Scanner: Detecting Hardware Trojans with Rapid SEM Imaging Combined with Image Processing and Machine Learning," vol. ISTFA 2018: Conference Proceedings from the 44th International Symposium for Testing and Failure Analysis, pp. 256–265, nov 2018. [Online]. Available: https://doi.org/10.31399/asm.cp.istfa2018p0256

[11] F. Courbon, P. Loubet-Moundi, J. J. A. Fournier, and A. Tria, "A high efficiency hardware trojan detection technique based on fast sem imaging," in *2015 Design, Automation Test in Europe Conference Exhibition (DATE)*, 2015, pp. 788–793.

[12] Q. Shi, N. Vashistha, H. Lu, H. Shen, B. Tehranipoor, D. L. Woodard, and N. Asadizanjani, "Golden gates: A new hybrid approach for rapid hardware trojan detection using testing and imaging," in *2019 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, 2019, pp. 61–71.

[13] B. Lippmann, N. Unverricht, A. Singla, M. Ludwig, M. Werner, P. Egger, A. Duebotzky, H. Graeb, H. Gieser, M. Rasche, and O. Kellermann, "Verification of physical designs using an integrated reverse engineering flow for nanoscale technologies," *Integration*, vol. 71, pp. 11–29, mar 2020. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167926019302998

[14] M. Ludwig, B. Lippmann, and N. Unverricht, "Enabling trust for advanced semiconductor solutions based on physical layout verification," in *Intelligent System Solutions for Auto Mobility and Beyond*, C. Zachäus and G. Meyer, Eds. Cham: Springer International Publishing, 2021, pp. 87–103.

[15] R. Torrance and D. James, "The state-of-the-art in semiconductor reverse engineering," in *Proceedings of the 48th Design Automation Conference*, ser. DAC '11. ACM, 2011, pp. 333–338.

[16] Q. R., D. R., and P. J., "Large-area automated layout extraction methodology for full-ic reverse engineering," in *J Hardw Syst Secur (2018) Springer International Publishing*, vol. 2, no. 4. Springer Science and Business Media LLC, oct 2018, pp. 322–332.

[17] A. Kimura, J. Scholl, J. Schaffranek, M. Sutter, A. Elliott, M. Strizich, and G. Via, "A decomposition workflow for integrated circuit verification and validation," *Journal of Hardware and Systems Security*, vol. 4, 03 2020.

[18] M. Werner, B. Lippmann, J. Baehr, and H. Gräb, "Reverse engineering of cryptographic cores by structural interpretation through graph analysis," in *3rd IEEE International Verification and Security Workshop, IVSW 2018, Costa Brava, Spain, July 2-4, 2018*. IEEE, jul 2018, pp. 13–18.

[19] J. Couch, E. Reilly, M. Schuyler, and B. Barrett, "Functional block identification in circuit design recovery," in *2016 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*. IEEE, may 2016, pp. 75–78.

[20] P. Laackmann and M. Janke. (2015) Hardware-trojaner in security-chips, eine reise auf die dunkle seite. [online; (accessed March 30, 2019)]. [Online]. Available: https://media.ccc.de/v/32c3-7146-hardware-trojaner{_}in{_}security-chips

[21] M. Tehranipoor and F. Koushanfar, "A survey of hardware trojan taxonomy and detection," *IEEE design & test of computers*, vol. 27, no. 1, pp. 10–25, 2010.

[22] P. Gupta, A. B. Kahng, C.-H. Park, K. Samadi, and X. Xu, "Wafer topography-aware optical proximity correction," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, no. 12, pp. 2747–2756, dec 2006.

[23] A. Singla, B. Lippmann, and H. Graeb, "Verification of physical chip layouts using gdsii design data," in *2019 IEEE 4th International Verification and Security Workshop (IVSW)*. IEEE, jul 2019, pp. 55–60.

[24] E. W. Weisstein. Affine transformation. [Online]. Available: https://mathworld.wolfram.com/AffineTransformation.html

[25] T. Fawcett, "An introduction to roc analysis," *Pattern Recogn. Lett.*, vol. 27, no. 8, pp. 861–874, Jun. 2006.

[26] J. Scholl, Y. Patel, J. Baur, A. Kimura, A. Waite, J. Kelley, and G. D. Via, "Sample mounting methods for precision delayering of 130 nm integrated circuit devices," in *2020 IEEE Physical Assurance and Inspection of Electronics (PAINE)*. IEEE, dec 2020, pp. 1–5.

[27] G. Becker, F. Regazzoni, C. Paar, and W. Burleson, "Stealthy dopant-level hardware trojans: Extended version," *Journal of Cryptographic Engineering*, vol. 4, pp. 19–31, 04 2014.

[28] S. T., D. S., R. F., S. T., R. Hori, M. S., and T. F., *Reversing Stealthy Dopant-Level Circuits*, L. Batina and M. Robshaw, Eds. Springer, Berlin, Heidelberg, 2014, vol. vol 8731.


https://t.me/learningnets