

UNCLASSIFIED

Reverse Engineering Complex Application-Specific Integrated Circuits (ASICs)

L. R. Avery, J. S. Crabbe,
S. Al Sofi, Sarnoff Corporation, Princeton, NJ
H. Ahmed, J. R. A. Cleaver, D. J. Weaver, Cambridge University, England

Summary: Many military systems contain complex ASICs built by companies that no longer exist, and where there is little documentation beyond a part number. Even where documentation exists, it is inadequate to faithfully reproduce the intended function. To solve this problem an ASIC characterization capability is being developed as part of the Advanced Microcircuit Emulation (AME) Program.

Using a combination of optical and electron beam microscopy, pattern recognition algorithms and netlist extraction techniques, it is possible to regenerate the original logic diagram and reproduce the original form, fit and function of the part using modern gate-array technology developed under the AME program. When fully debugged, the goal of this prototype capability is to reverse engineer the connectivity of a 50k gate array in about one month, and all layers of a similar sized, full-custom chip in about 3-4 months.

Acknowledgements: Much of the integrated circuit reverse engineering work described in this paper has been developed in collaboration with the Cavendish Laboratory, Microelectronics Research Centre, at Cambridge University, Cambridge, England. Cambridge has developed e-beam and optical imaging technology, chemical staining and plating techniques, and image distortion correction algorithms to ensure layer to layer and optical to e-beam image alignment.

This work is part of the AME program sponsored by the Defense Logistics Agency (DLA)

Background: Early integrated circuits were designed at the transistor level, and the full circuit schematic, including component values, published. As ICs became more complex, block-function-level diagrams replaced the transistor-level schematics, and these were often only *representations* of the logic function, and not a true schematic of the logic. Today, the use of VHDL is widespread, and logic ICs are designed algorithmically using sophisticated software.

The IC fabrication process can be conveniently split into two parts: the base layers, resulting in the formation of transistors and other components, and the “back end” or interconnect layers, connecting the silicon components to form the required circuit function. In logic circuits using Gate-Array technology, including Field-Programmable Gate-Arrays (FPGAs), the base layers are the same for all circuits fabricated with that array, and only the interconnection layers (or fuse/antifuse links in the case of FPGAs) vary. Therefore, for these types of circuit, once the base layers are determined, it is only

necessary to extract the interconnect layers. For full custom circuits, such as digital signal processing and microprocessor chips, all layers must be extracted.

The Reverse Engineering Process: The process begins with an optical examination of the IC. From this it is often possible for an experienced engineer to determine the fabrication technology (CMOS, ECL, TTL, Schottky etc.), the number of metal interconnect layers, minimum feature size, and the input and output ports. Unfortunately, the data provided by the optical image, although readily discernable to the human eye, is very confusing to pattern recognition software, since all metal layers often appear the same, and thus are interpreted as being “shorted together”. This is where the use of e-beam imaging techniques has proved invaluable. The back-scattered electrons from the incident beam vary according to the depth of their back-scattering, and the material type. The top metal layer (the second metal in the circuits of interest) is underneath a single dielectric layer, whereas all other layers are under two or more dielectric layers. The number of backscattered electrons reaching the detectors is therefore significantly higher from those layers near the surface of the IC. Using commercially available image processing software it is a simple matter to extract the second metal layer (M2) from the other layers.

Before imaging the other layers it is necessary to remove the protective overcoat glass and M2. This is normally done using wet and/or dry-etch techniques, depending on the layers. Removal of the overcoat glass must be done carefully to avoid removal of the inter-level dielectric layer, but must be adequate to completely remove all dielectric from the surface of the M2 layer.

The top metal layer connects to the lower metal layer (M1) through holes (called vias) in the inter-level dielectric. When the M2 layer is etched, the M1 layer under the via, connected by M2, is also etched. It is necessary to identify all the M2-M1 via interconnects to determine the circuit connectivity. Currently, this can be accomplished in two ways: by pattern recognition of the M2 layer, and by enhancing the vias using an electroless plating technique. A close observation of the M2 image reveals a doughnut-like image wherever a via is present. Using a suitable pattern-recognition algorithm, the majority of vias are easily recognized. Also, the vias can be enhanced using an electroless zinc plating technique on the via holes after M2 is removed to produce a similar doughnut-shaped image, with high contrast to other layers, when imaged in the scanning electron microscope. These two images are compared, and a composite via image generated.

Satisfactory imaging of the first-level metal has proved the most challenging. The various etch processes result in either excessive removal of material, or unwanted material remaining, causing the generation of artifacts when imaged by conventional e-beam or optical microscopy techniques. There is also a lack of image contrast between M1 and the transistor poly silicon gates due to removal of the dielectric over the poly silicon. Additionally, the dry-etch removal of the M2 and overcoat layers leaves the glass layer on the sidewalls of the M2, causing apparent “breaks” in the M1 layer.

Many approaches have been tried to eliminate these artifacts. The most successful approach involves dry-etching all the dielectric from above and along side the M1 layer, coating the chip with a layer of photoresist, and taking a dark field image of the M1. This provides a very high contrast image with adequate detail for the reverse engineering process.

Similar image-enhancing techniques are used to derive the other layers. This is a necessary step the first time a particular gate array is imaged, and necessary for all custom IC layouts. Preconditioning software adjust the relative sizes and distortion of individual images to ensure correct alignment and ensure full connectivity between layers. The photo-like images are then enhanced and filtered to separate the layer of interest, and converted into high-contrast filtered images further processing.

Building the Schematic: When the SEM image processing is complete, the resulting database is parsed to identify blocks of data that are topographically similar, with the purpose of identifying groups of identical blocks, or macros. In order for any two blocks to qualify as instances of the same macro, this topographical similarity must extend to all layers. A first-pass parse will attempt to match blocks with identical via patterns; since via is a relatively sparse layer, it provides the most efficient elimination test for potential members of any single macro category. Next, each member of the resulting groups are compared with one another by analyzing contact, the next sparsest layer, and then metal 2, metal 1, poly and so forth. For gate-array derived-blocks, identification of each unique logic macro can be performed with only metal, contact and via layers, but for full custom circuits it is necessary to analyze both the poly silicon gate and the active silicon areas as well.

As this procedure of grouping together identical blocks progresses, those blocks with differences are separated out, until all identical blocks have been identified. Each group of identical blocks becomes a unique macro. These macros constitute the basic components of the design library for a particular IC, and become part of a general library of macros for the part manufacturer. Any one IC may contain over 50 different macros. These can be entered in a library database and referenced in interpreting other similar circuits from the same manufacturer. The more designs of a family analyzed according to this procedure, the more comprehensive the macro library and the less time it takes to complete this stage of the process for subsequent chips.

After the macros have been defined, the location, rotation and orientation of each macro instance is then recorded in a CAD database. Once the macros have been located within the database, the problem of extracting a schematic for the design is reduced to determining the function of each macro, and extracting the interconnection between each instance.

To determine the function of a macro, the following three steps are undertaken. First, an instance of each macro is translated, layer by layer, into the IC physical design standard format of GDSII. Then an algorithm similar to that used by IC layout verification software is applied to the GDSII database, yielding a transistor-level netlist.

Finally, commercially available software is used to translate the transistor-level netlist into a logic function netlist. At the end of this procedure, all layers of all macro instances will be identified in the design.

While the macros are being processed, the interconnection between the macro instances is extracted. Metal routing and via are then translated into GDSII, in a manner similar to the extraction of the macro layers. Once this is completed, there is a GDSII database with full layer information for all macros, a cell-based hierarchy for the design, and top-level interconnect between the ports of the macro instances. By virtue of the macro identification steps undertaken previously, this GDSII database contains hierarchical design information (in the cell-based hierarchy) that can be utilized for the final step of extracting a hierarchical netlist for the top-level design. By leveraging this macro information, the cumbersome procedure of transistor-level verification has been avoided, thereby greatly increasing the speed and reliability of the top-level netlist extraction procedure. Once a top-level hierarchical netlist for the design has been obtained, proprietary software is used to extract higher levels of functionality, which aids in the human-readability of the final schematic.

Recreating the IC: Although it is theoretically possible to recreate the original chip by assembling data from individual layers, in a manner similar to the way the original chip was constructed, this technique does not work well in practice due to the additional cost and time required to produce masks and process wafers for each part type. Instead, it is preferable to emulate the logic function using the AME Cell Library and one of the AME arrays designed for this purpose. This approach ensures the AME Program does not require or utilize proprietary data in its emulation process.

Suitable test vectors can be obtained either by interrogating the ASIC in the original system, and recording input and output test vectors, or by algorithmic generation, based on the extracted logic function. Obviously the former is the preferred method, as it allows loop closure between the performance of the original chip in the system, including timing, and the reverse engineering extracted schematic.

Conclusions: A reverse engineering technique has been developed to extract the detailed schematic diagram from ASICs, to allow full Form, Fit and Function emulation of the circuit using AME technology. To date, gate-array-based circuits up to 36 mm² have been successfully reverse engineered, and work on a full custom ASIC is nearly complete.

Although this work has concentrated on digital circuits, similar techniques can be applied to analog circuits.