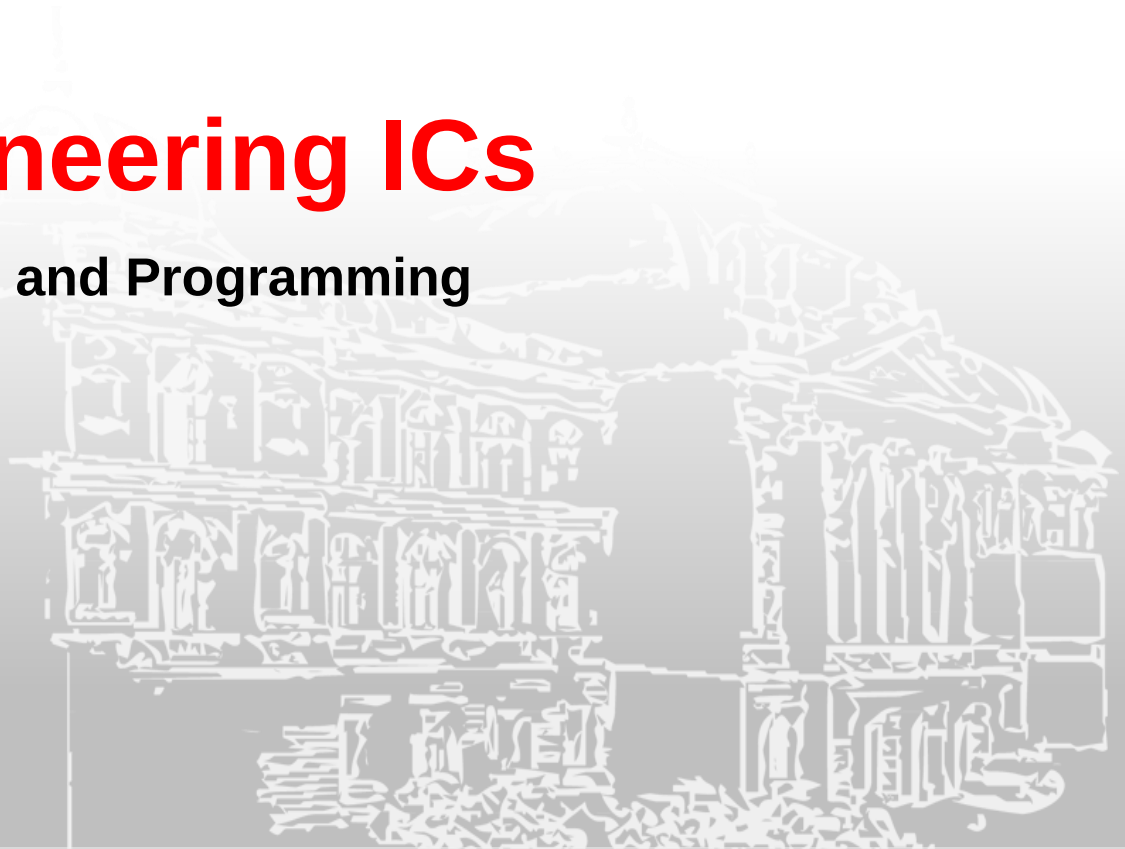
Reverse Engineering ICs

Digital System Integration and Programming

Klemens Armstorfer

23.11.2022



Agenda

- Background
- Reverse Engineering
- Netlist Extraction
- Specification Discovery
- Available Tools
- Conclusion and References

Overview of ICs

- Electronic circuit on a small flat piece of semiconductor (chip, die)
- Also called "monolithic circuit"
- Consists of MOSFETs and other semiconductor – parts
- Manufactured on a wafer
- Size of transistors up to 7nm
- From a few dozen transistors
  in 1960 to billions of transistors now

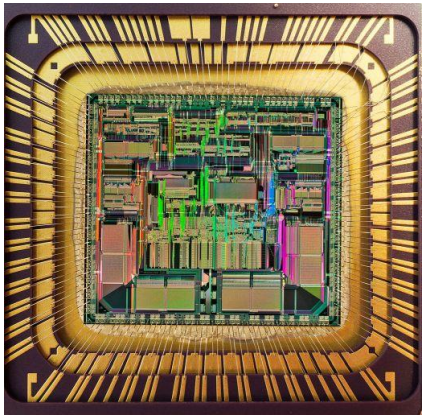


Fig. 1: Picture of the die of an Motorola 68040 chip[1]

Advantages of ICs

- Integrate complex circuits
- Replace recurring parts of circuits
- Cost and size effective
- Standardized parts with defined behavior
- Example: 4x NAND-gate in TTL logic

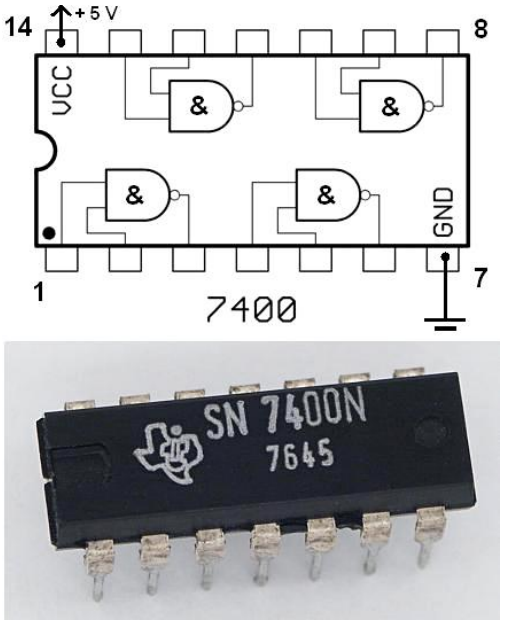


Fig. 2: SN7400N chip with its circuit diagram[2]

Types of ICs

- Field programmable gate array (FPGA)
- Application specific integrated circuit (ASIC)
- Analog ICs (amplifiers)
- Digital ICs
- Mixed-signal ICs
- System on a chip (SoC)
  - Apple M1, Microcontroller MSP430

Fig 3: Picture of the microcontroller MSP430[1]

Reverse Engineering

- Indigent peeking ("unverschämter Blick")
- Gain information about the chip
- Deductive process to understand how an already made device works
- Problem: ethics and compliance with law
- In 1984 US SemicondutorChip Protection Act, for educational purposes
- Tools: Ghidra (NSA), Cutter

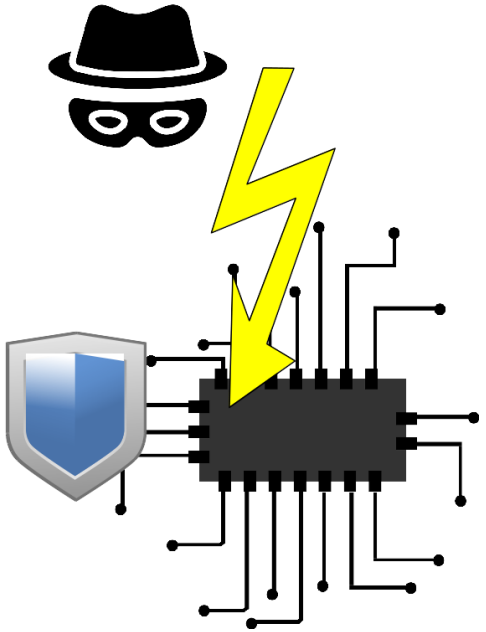
Fig. 4: Logo of Ghidra[4]

Motivation for Reverse Engineering of ICs

- Competitive analysis of competitors
- Check if competitors steal your design
- Monitor semiconductor suppliers
- Detect hardware trojans
- Detect counterfeit devices
- Failure analysis

Threat Model

- Attacker has unlimited physical access
- Assets:
  - IP on the device
  - Data in memory
  - The chip design itself
- Process of reverse engineering:
  - Input is a physical device
  - Output is a human-readable specification



General Steps

- 2-step approach
  - Netlist extraction
  - Specification discovery

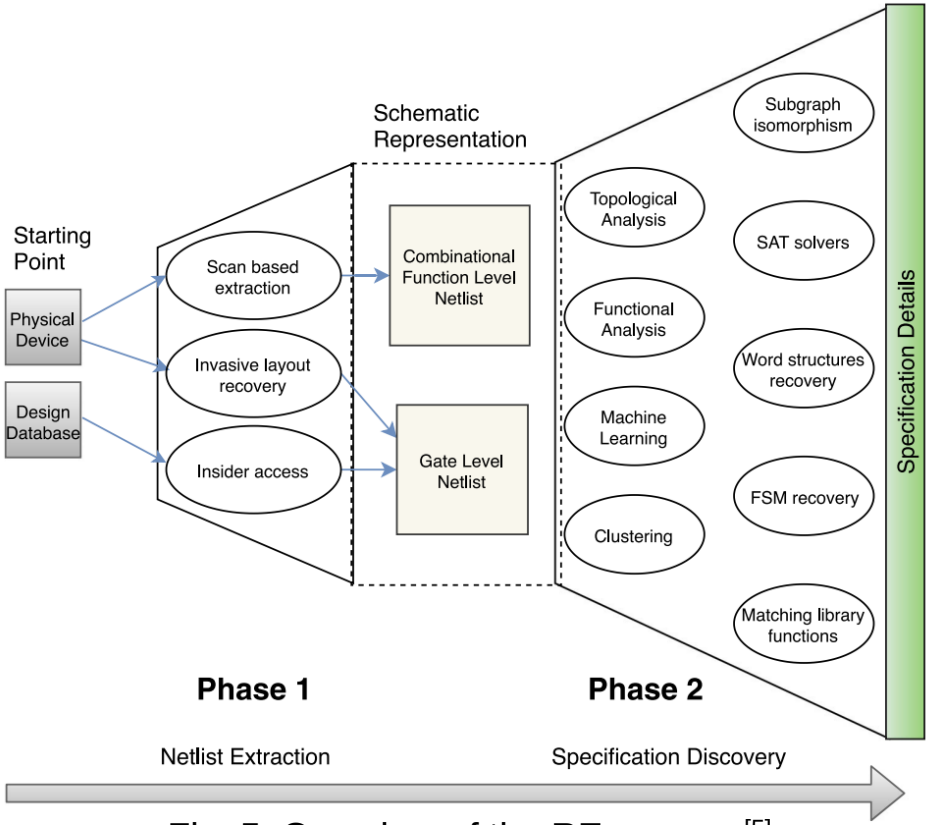


Fig. 5: Overview of the RE process[5]

The Netlist

- Describes the connectivity of the circuit
- Consists of components and connected nodes
- Well defined goals, metrics and processes
- Mostly written in HDL

- Netlist Extraction process
  - Input is the chip to examine
  - Output is a human readable netlist

```
P  UNITS CUST 1
P  VER IPC-D-356A
P  IMAGE PRIMARY
327+VIN        R260  -1      A04X+128397Y+299720
327+VIN        C16   -2      A04X+016510Y+288163
317+VIN        U18   -2   D1321PA00X+016510Y+292100
327+VIN        R26   -1      A04X+019050Y+288163
```

Fig. 6: Example Netlist

Netlist Extraction on ASIC

- Invasive method, harder with shrinking gate sizes
- Process:
  - Decapsulation (remove package)
  - Delayering (etching, milling)
  - Imaging (record gates and connections)
  - Processing (stitch images together)

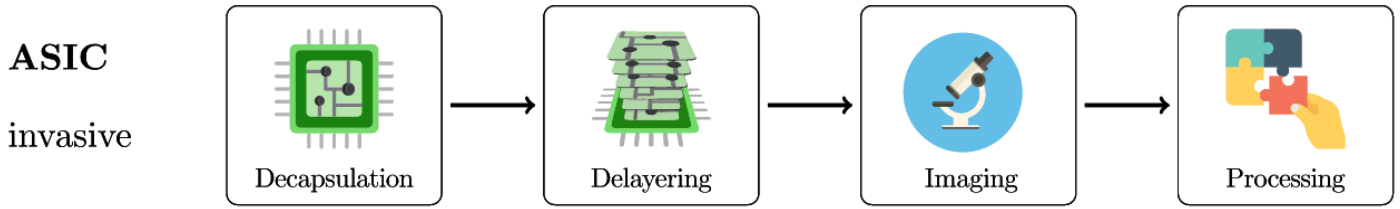


Fig. 6: Overview of the invasive Netlist Extraction on ASIC[5]

Netlist Extraction on ASIC

- Extracting via scan chain, non-invasive
- Less resources, but limited in accuracy
- Exploit common design-for-test technique
  - Arrange internal registers as shift registers
  - Capture and probe cycle
  - Calculate boolean function between registers

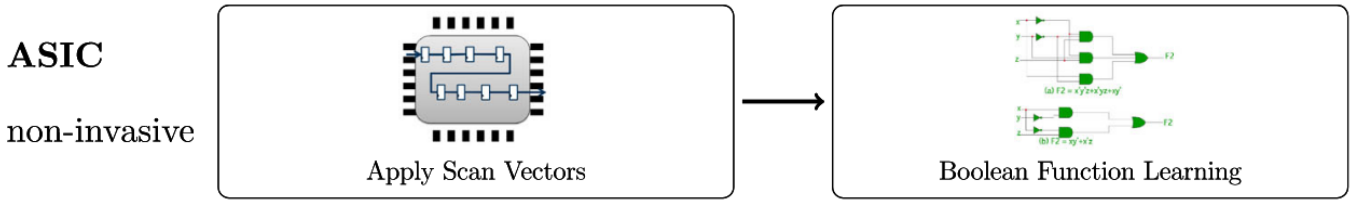


Fig. 7: Overview of the non-invasive Netlist Extraction on ASIC[5]

Netlist Extraction on FPGA

- Extract bitstream from memory
  - Wiretap configuration lines
  - Read flash memory
  - Break bitstream encryption with Side-channel attack
- Understand the bitstream by correlating with example bitstreams

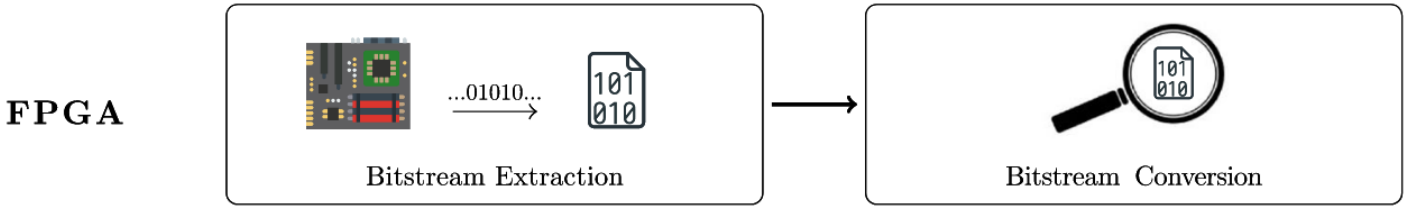


Fig. 8: Overview of the non-invasive Netlist Extraction on FPGA[5]

Specification Discovery

- Many different approaches
- Same procedure for ASIC and FPGA
- Process
  - Input is gate-level netlist
  - Output is full understanding of the functionality
- Problems
  - What is the full understanding of the chip?
  - On which abstraction level?

Specification Discovery

- Combine fundamental algorithms from different areas
  - Matching Library modules (matching patterns)
  - Repeated modules
  - Common names (netlist names provided)
  - Control functions and bus structures
  - Partitioning
  - Functional und structural analysis

Early Work

- First comprehensive studies on ISCAS-85
- Benchmark circuits from 1985 ISCAS (International Symposium on Circuits And Systems)
- Small and custom built circuits
- Basics for modern reverse engineering
- Example: ISCAS – 85 C17

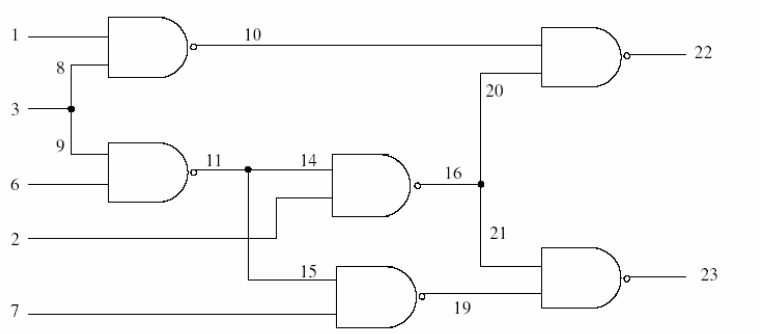


Fig. 9: Schematic of the ISCAS – 85 C17[6]

Partitioning

- Partition netlist into design hierarchy
- Top-Down
  - Translate netlist into directed graph and partition it into subgraphs
  - Min-cut algorithms, NP-complete problem
  - Structural matching with known cells
- Bottom-Up
  - Start with small subcircuit and add elements
  - Shared Nearest Neighbor
  - Problem: same implementation, different structures

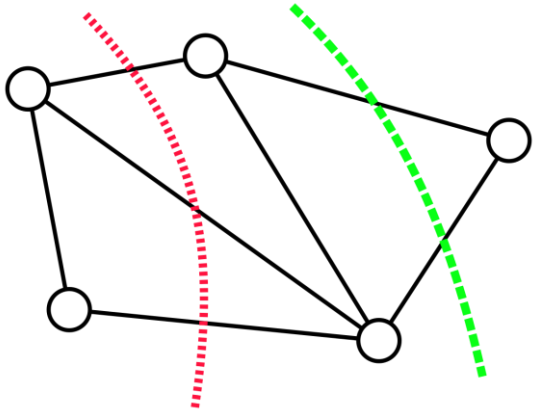


Fig. 10: Min-Cut Algorithm[7]

Structural Analysis

- Only care about topological properties
- Mostly Graph-based algorithms
  - Label circuit and library cells concerning surroundings
  - Use nonlinear optimization algorithm on the match matrix
  - Construct subcircuits out of this matrix
- Problem: Size of the subcircuits
- Problem: Errors in the netlist
  - Aim for lowest error vector

Functional Analysis

- Behavioral analysis
  - Utilized for logic equivalence verification
  - Combinational matching algorithms
  - Match subcircuits with library component
- Monitoring optical emissions during operation may helps
- Problem: Amount of different subcircuits!
  - Match subcircuit with Templates or subcircuit type

Putting it All Together

Methods for revealing the functionality as a whole

- Extracting finite state machines
  - Split control logic and data processing part
  - Try to identify state registers and state transitions
- Combine structural analysis and formal verification
  - Convert output of structural analysis into a standardized form
  - Solve it with SAT solver

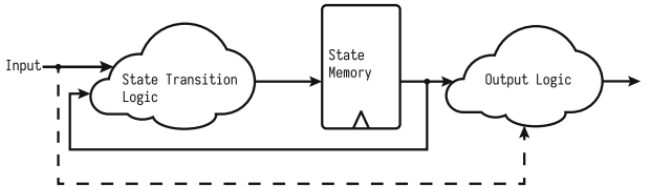


Fig. 11: Example overview of a state-machine[5]

Putting it All Together

- Machine learning
  - Learning techniques (clustering)
  - Open for future work
- World-level identification
  - High-level register reconstruction and dataflow analysis
  - DANA algorithm (Dataflow ANAlysis)

Available Tools

- Hardware analyzer (HAL)
  - Framework to create tools
  - Convert netlist into multi-graph representation
- ChipWorks from TechInsigths (Ca)
- ChipJuice from Texplained (Fr)
  - Processes layer images of ICs
  - Generates a gate-level netlist

Fig. 12: Texplained Logo[8]

Conclusion

- Internal information of ICs becomes more and more important
- Extracting the inner structure of the chip becomes harder with shrinking gate sizes
- Analyzing the netlist is rather easy
- Analyzing the functionality is hard
- No “jack of all trades” exist, we need to combine different techniques

➢ Much room for future work

References and Image Sources

- [1] Motorola 68040-Die, Link: https://en.wikipedia.org/wiki/File:Motorola68040die.jpg, Author: Gregg M. Erickson, Licence: CC BY 3.0
- [2] SN7400N chip with circuit diagram, Link: https://klexikon.zum.de/wiki/Benutzer:Hans_Haase/Digitaltechnik, Author: Hans Haare, License: unknown
- [3] MSP430 aufgelötet auf einer Leiterplatte, Link: https://commons.wikimedia.org/wiki/File:Aktivmed_GlucoCheck_Comfort_-_Texas_Instruments_M430FG438-7739.jpg, Author: Raimond Spekking, Licence: CC BY-SA 4.0
- [4] Ghidra Logo, Link: https://ghidra-sre.org/images/GHIDRA_1.png, Author: Ghidra, License: unknown
- [5] Azriel, L., Speith, J., Albartus, N. et al. A survey of algorithmic methods in IC reverse engineering. J Cryptogr Eng 11, 299–315 (2021). https://doi.org/10.1007/s13389-021-00268-5
- [6] Pereira, Luís & Lamma, Evelina & Risorgimento, Viale & Riguzzi, Fabrizio. (2001). Logic Aided Lamarckian Evolution
- [7] Min cut example, Link: https://commons.wikimedia.org/wiki/File:Min_cut_example.svg, Author: Kilom691, License: CC BY-SA 3.0
- [8] Texplained Logo, Link: https://www.texplained.com/, Author: Texplained, License: unknown

Reverse Engineering ICs

Digital System Integration and Design

Klemens Armstorfer

23.11.2022