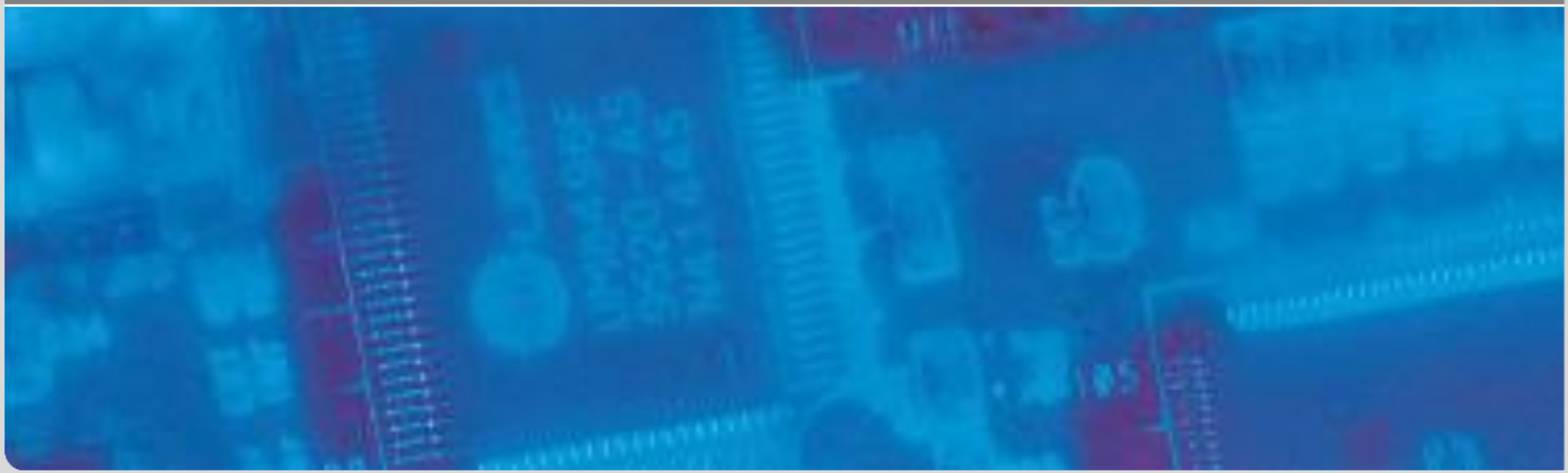
Karlsruher Institut für Technologie

Institut für Technik der Informationsverarbeitung

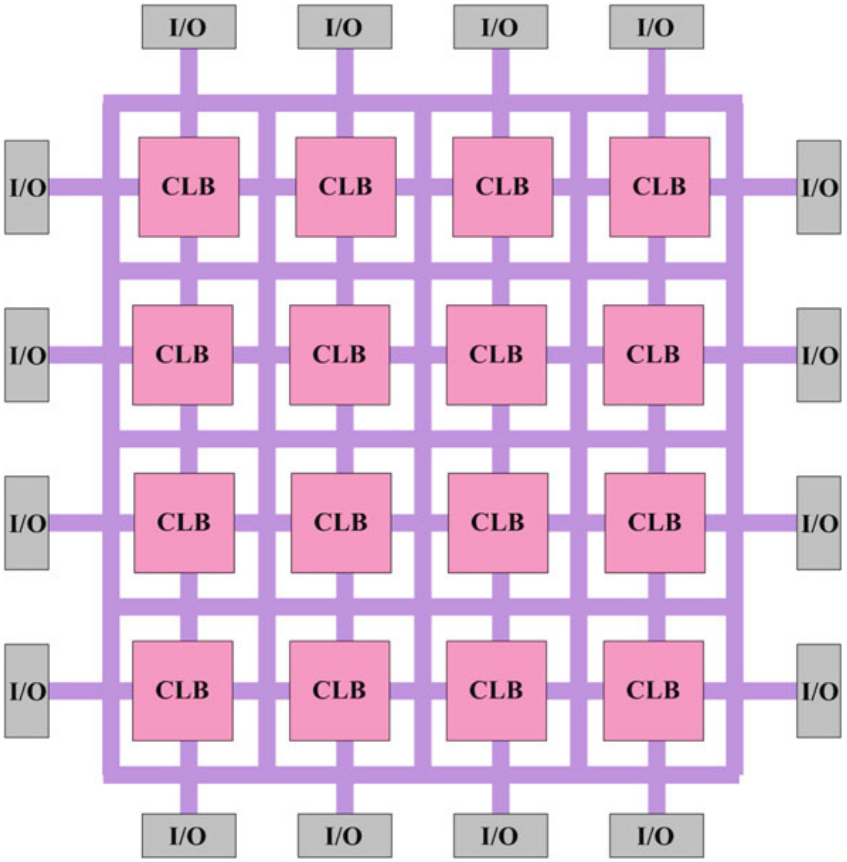
Institutsleitung
Prof. Dr.-Ing. Dr. h. c. J. Becker
Prof. Dr.-Ing. Eric Sax
Prof. Dr. rer. nat. W. Stork

A Tutorial on FPGAs

M. Sc. Simon Reder, M. Sc. Augusto Hoppe



Topology of FPGAs

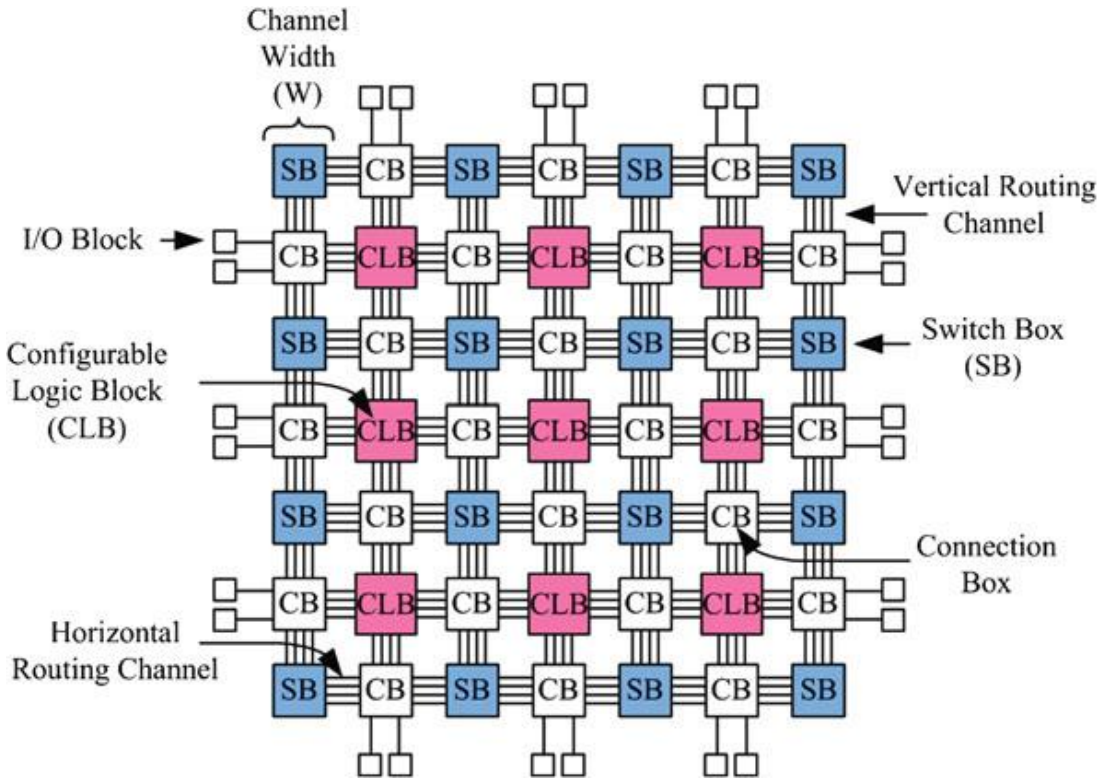


- **Configurable Logic Block (CLB):** Contain Programmable Logic Cells of FPGA
- **Input/Output (I/O) Blocks:** Connect FPGA to its Periphery

- General FPGA (2D FPGA Structure)

U. Farooq et al., *Tree-Based Heterogeneous FPGA Architectures*, Chapter 2 FPGA Architectures: An Overview, Springer 2012

Routing Network of FPGAs

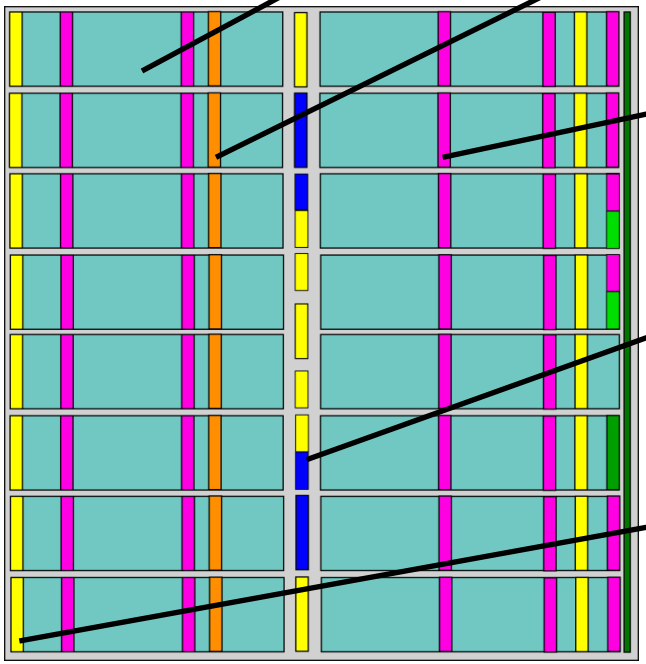


- **Switch Box (SB):** general FPGA-wide horizontal and vertical Routing
- **Connection Box (CB):** Connect CLBs to Routing Network

- Traditional island-style FPGA (Xilinx uses this structure)

U. Farooq et al., *Tree-Based Heterogeneous FPGA Architectures*, Chapter 2 FPGA Architectures: An Overview, Springer 2012

Structure of FPGAs

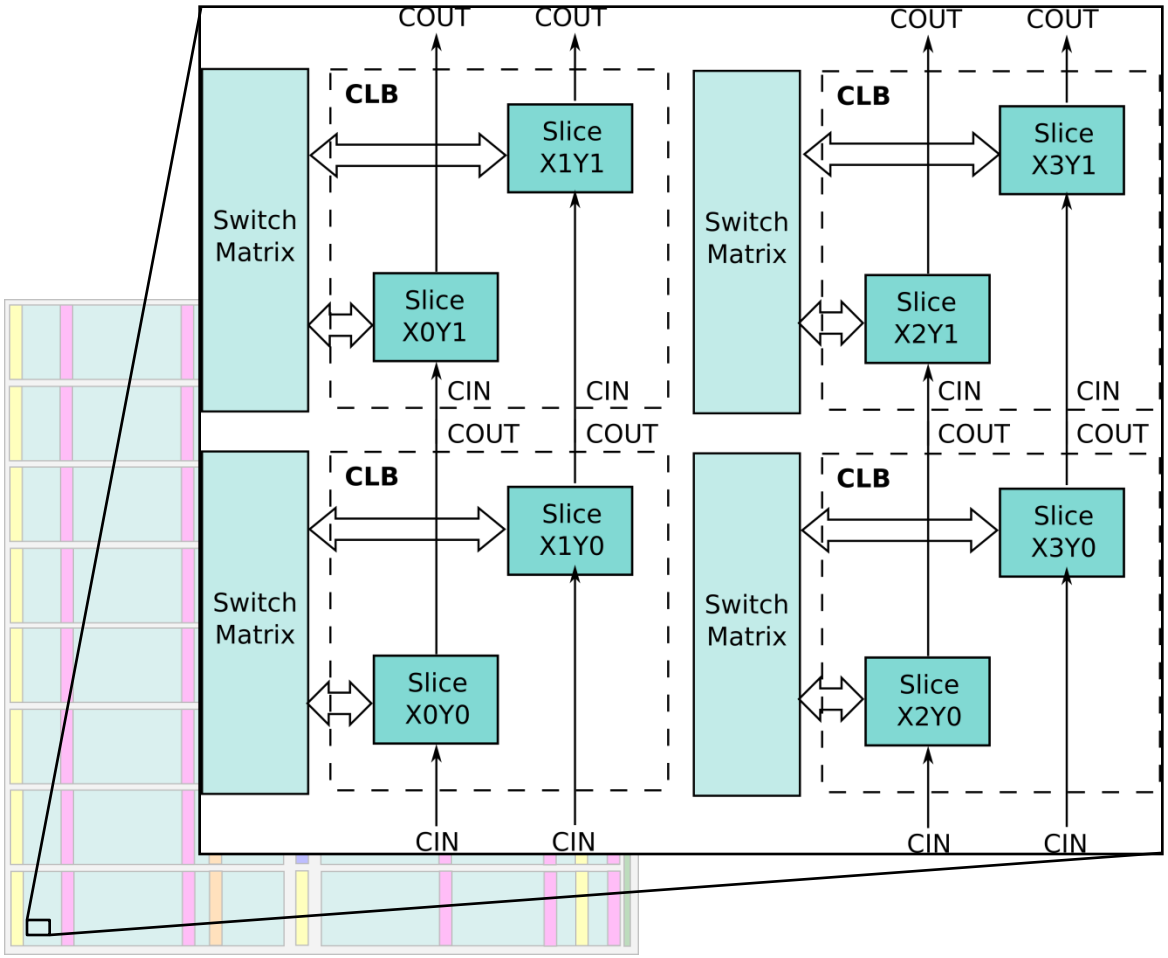


- Configurable Logic Block (CLB)
- Digital Signal Processor (DSP)
- Block RAM (BRAM)
- Digital Clock Memory (DCM)
- I/O Bank

These Blocks are arranged in *Columns*

Example: Xilinx Virtex5 110T

Structure of FPGAs - CLBs

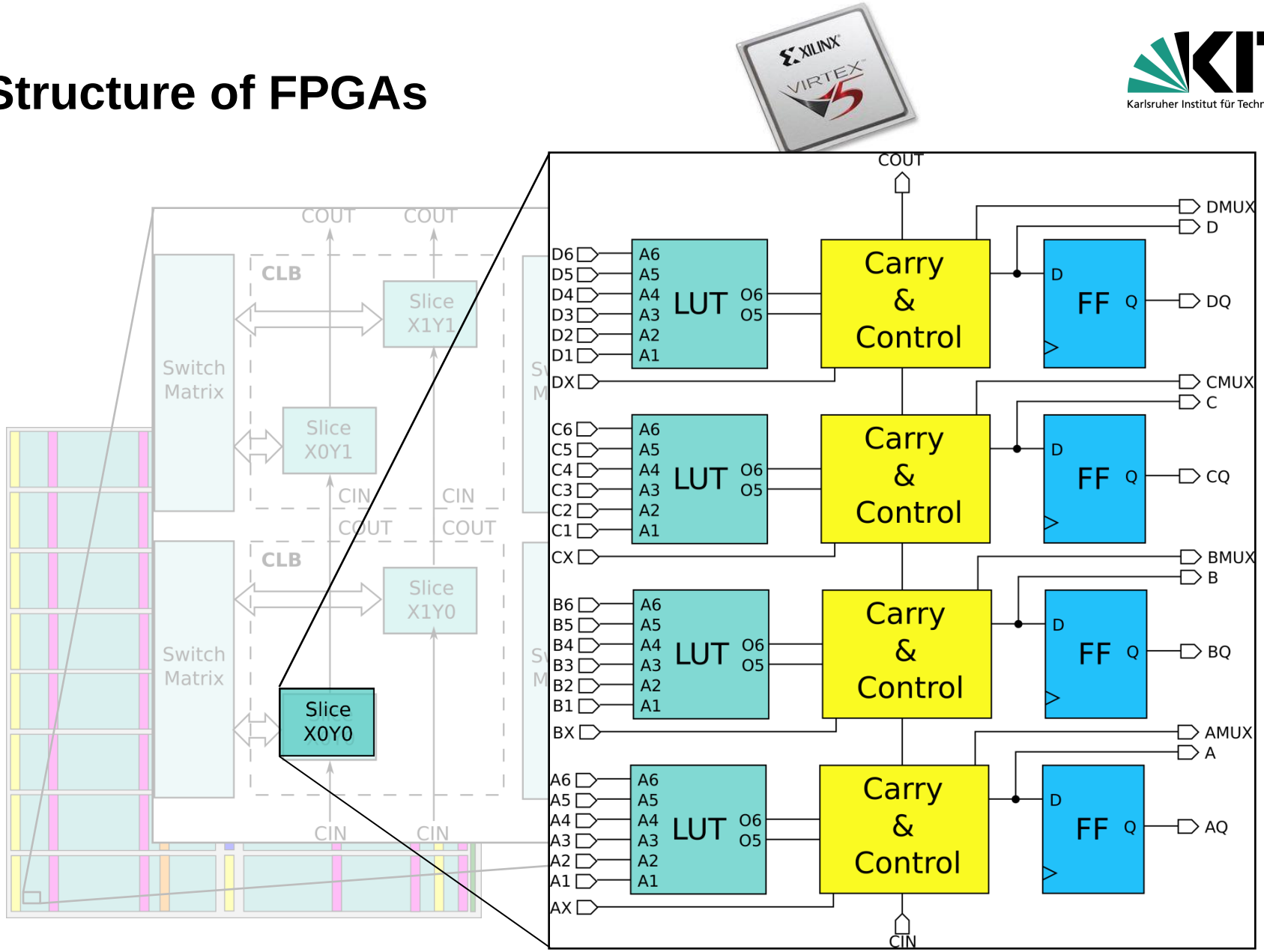


- One CLB consists of two Slices
- Programmable Switch Matrix
- Fast local routing inside a CLB
- Global routing between CLBs

Example: Xilinx Virtex5 110T

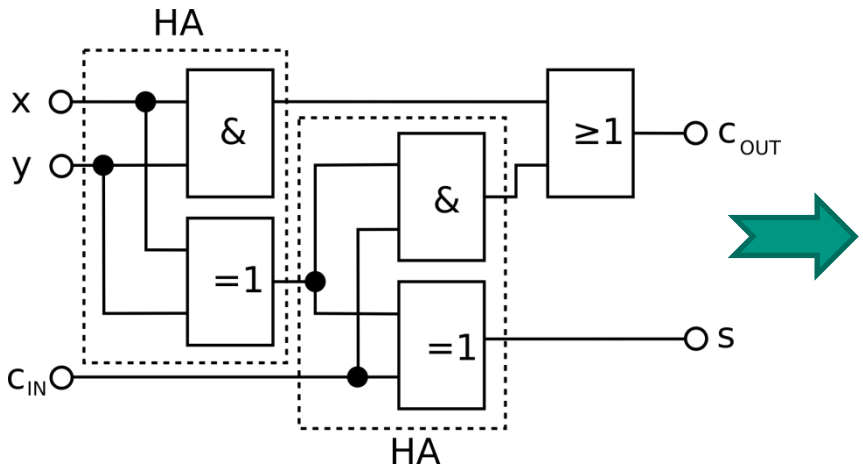
Dipl.-Inform. Tanja Harbaum, KSETA Workshop 2013

Structure of FPGAs



How to map a full adder into a Slice

- Step 1: Create a truth table

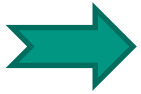


X	Y	Cin	S	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

How to map a full adder into a Slice

- Step 2: Produce Disjunctive Normal Form (DNF)

X	Y	Cin	S	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



$$s = (\overline{x} \wedge \overline{y} \wedge Cin) \vee (\overline{x} \wedge y \wedge \overline{Cin}) \vee (x \wedge \overline{y} \wedge \overline{Cin}) \vee (x \wedge y \wedge Cin)$$

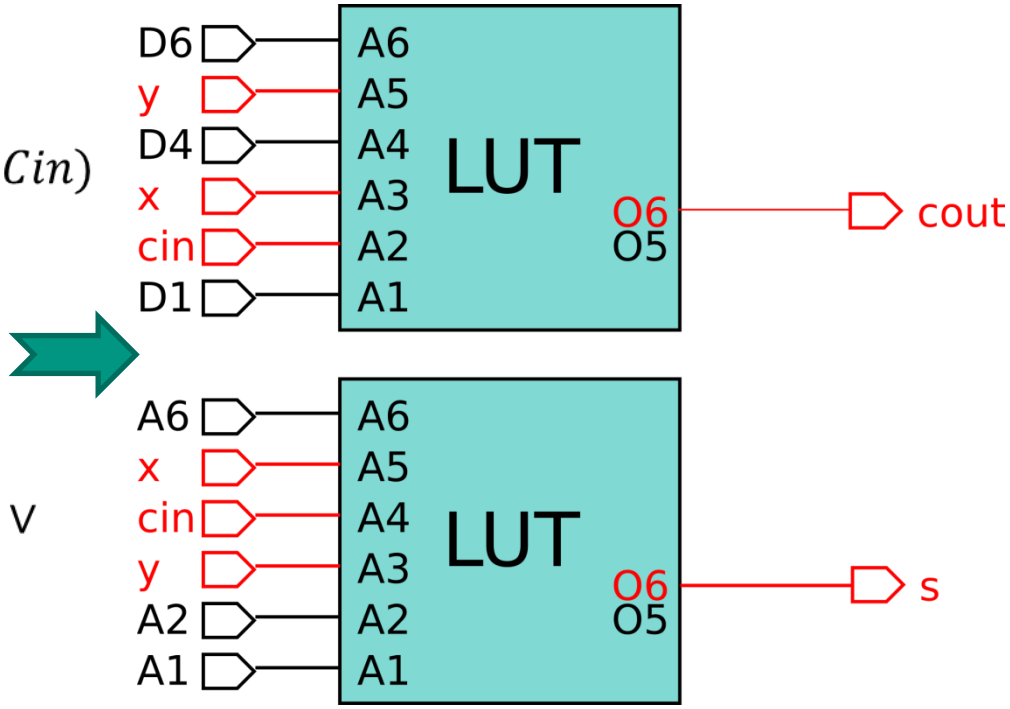
$$Cout = (\overline{x} \wedge y \wedge Cin) \vee (x \wedge y) \vee (x \wedge Cin)$$

How to map a full adder into a Slice

- Step 2: Realization with two Lookup Tables

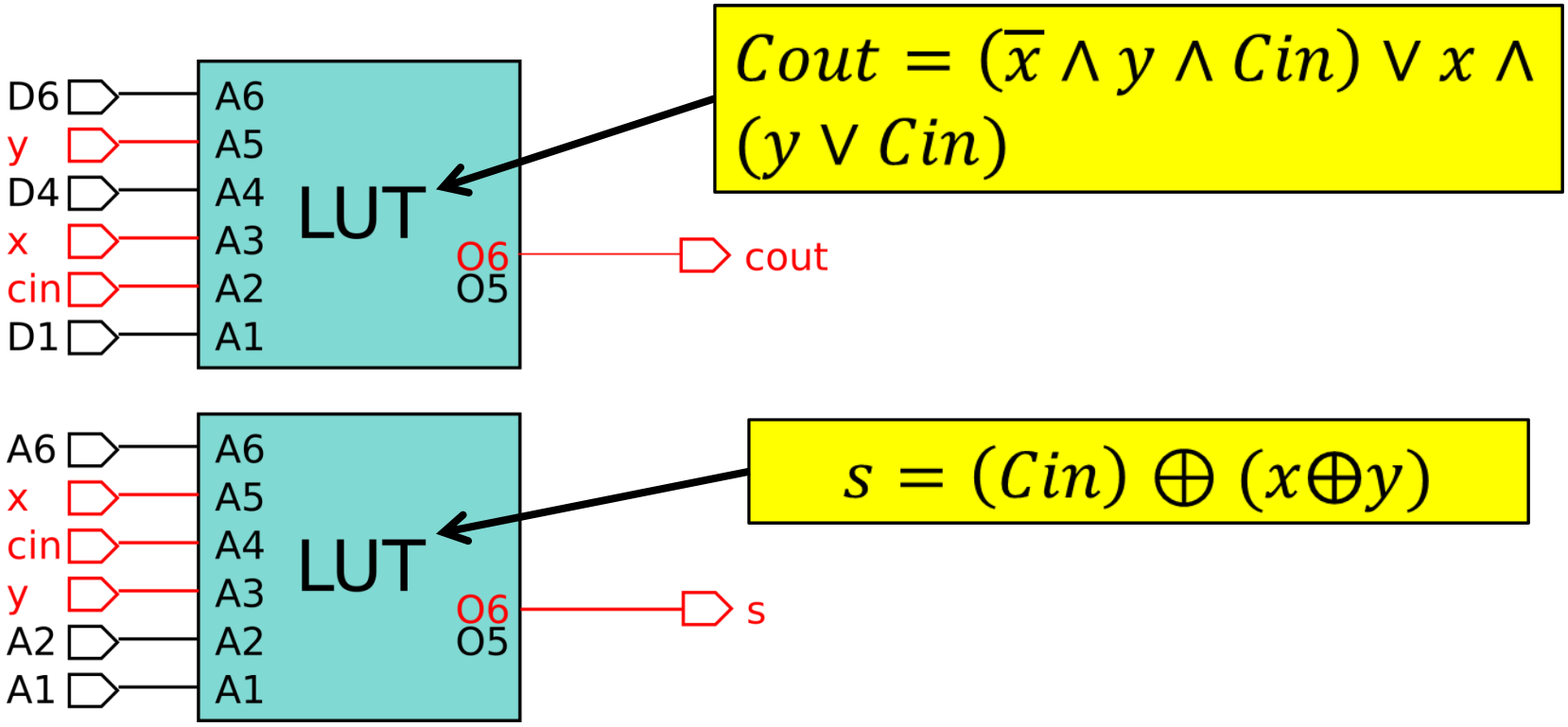
$$Cout = (\overline{x} \wedge y \wedge Cin) \vee (x \wedge y) \vee (x \wedge Cin)$$

$$s = (\overline{x} \wedge \overline{y} \wedge Cin) \vee (\overline{x} \wedge y \wedge \overline{Cin}) \vee (x \wedge \overline{y} \wedge \overline{Cin}) \vee (x \wedge y \wedge Cin)$$



How to map a full adder into a Slice

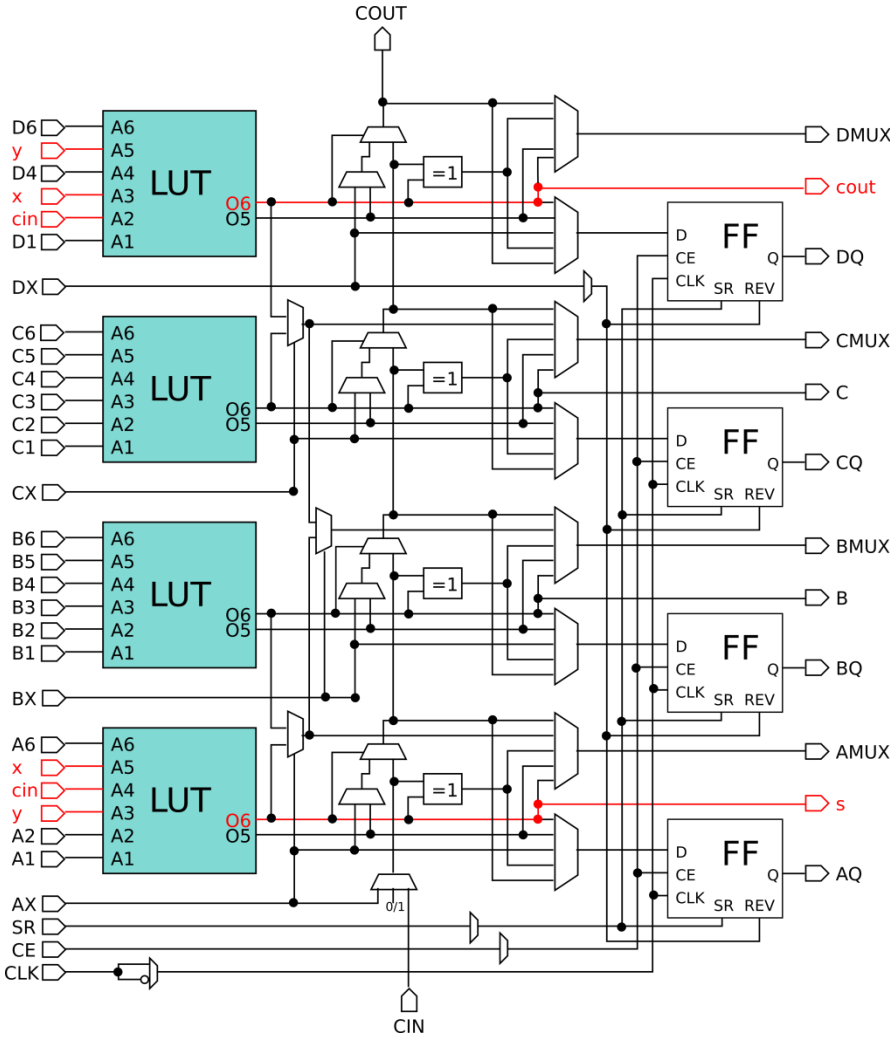
- Step 2: Realization with two Lookup Tables



Generated by Xilinx ISE 14.2

Dipl.-Inform. Tanja Harbaum, KSETA Workshop 2013

Full adder in a Virtex5 SliceL



Dipl.-Inform. Tanja Harbaum, KSETA Workshop 2013