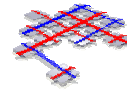


Computer Science II

(Summer Semester 2003)

Prof. Dr. Dieter Hogrefe
Dr. Xiaoming Fu
Kevin Scott, M.A.

Telematics group
University of Göttingen, Germany



Part I Review

- Number systems
- Gates
- Boolean algebra & K-maps
- Latches and flip-flops

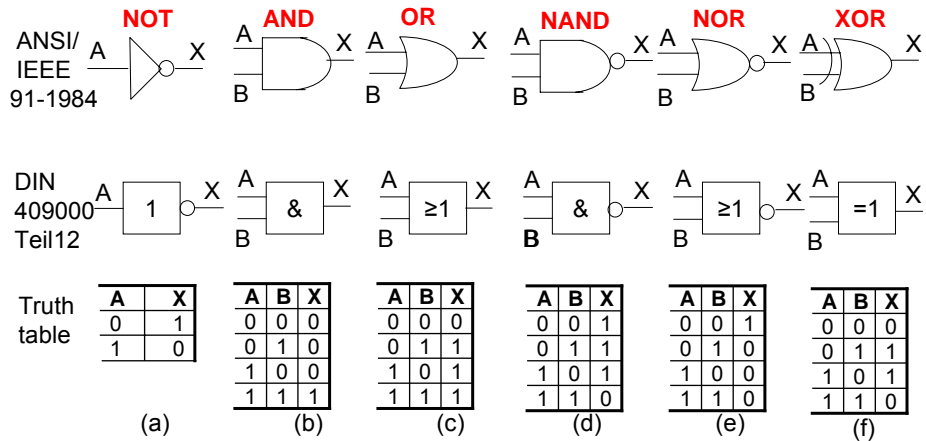
Number Systems

- 4 number systems
 - Decimal: base-10
 - Binary: base-2
 - Octal: base-8
 - Hexadecimal: base-16
- Basic computing unit in digital computers: binary data
- Octal & hexadecimal: better presentation
- Conversion of different number systems

Logical Gates

- Logic circuits allow computer engineers to develop digital circuits in a **logical manner**.
- The building blocks of a logic circuit are called **gates**.
- Q:
 - *How many types of gates do you know?*

6 Gates and Their Notations



SS 2003 Computer Science II

5

General Remarks for Gates

- Gates are used to describe digital logics, which generally convert from input voltage(s) to output voltage.
 - Logic **1**: 2V-5V
 - Logic **0**: 0-0.8V
- Gates are built in units: Integrated Circuits (**ICs**)
 - SSL (1-10gates), MSI(10-100gates), LSI(100-100,000gates), VLSI(>100,000gates)
- Gate **delay**: 1-10ns
- A principle for circuit design: high **gate/pin ratio**

SS 2003 Computer Science II

6

Implementation of a Boolean Function

- Any Boolean function can be expressed as **sum of products (SOP)**.
- *How to get a sum of products?*
 1. Create a truth table that describes the desired logic circuit.
 2. Find a minterm with a truth-value of "1" for each true row of the truth table.
 3. Take the logical sum of the minterms found in step 2
- *How to implement it? Generally,*
 1. Look at the truth table and the SOP
 2. For any complement of inputs, provide inverters
 3. Draw an AND gate for result item with 1
 4. Wire the AND gates to appropriate inputs
 5. Feed the output of all the AND gates into an OR gate
- *Boolean functions can be simplified:*
 - **Boolean algebra**
 - **K-map**

SS 2003 Computer Science II

7

Boolean Algebra

- Defines operation rules upon binary numbers 0 and 1
 - Single variable: rules 1-8
 - Multiple variables: rules 9-15
 - DeMorgan's theorem: rules 16-19
- Based on Boolean algebra, Boolean expressions can be written into *sum of products*, and *sum of minterms*

SS 2003 Computer Science II

8

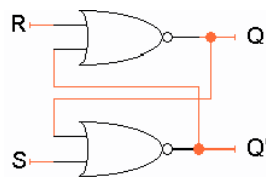
Karnaugh Maps

- **Karnaugh Maps** is an alternative, more "graphical" simplification technique that enables us to build and study larger, more realistic circuits.
- It can reduce expressions to a minimal sum of products (MSP) form
- 4 steps:
 - Start SOP or truth table
 - Plot into K-Map
 - **Group the items:**
 - Number of groups: the fewer, the better
 - Size of each group: the larger, the better
 - Reduce each group into one product term

SS 2003 Computer Science II

9

Latches and Flip-flops

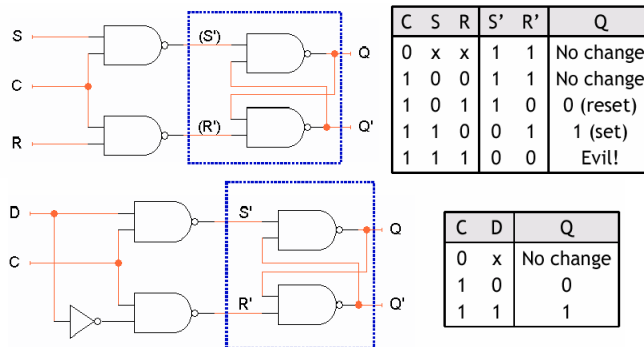


- Latches are circuits that are able to remember states
- E.g., **SR latch** remembers whether S or R was last on
 - If S was on, $Q=1$
 - If R was on, $Q=0$
- Using this feature we can build computer memories

SS 2003 Computer Science II

10

Latches and Flip-Flops



- A SR Latch & a D Latch with Control Input
- If C → edge signal of a clock:
 - It becomes a **flip-flop!**

SS 2003 Computer Science II

11

General Remarks Regarding Latches and Flip-Flops

- Combinational circuits can be used for composing ALU (Arithmetic Logic Unit)
- Latches and flip-flops are sequential circuits
- The difference between a latch and a flip-flop is a flip-flop uses edge-triggered clock to trigger the state change.
- Flip-flops can be used to built registers and memories

SS 2003 Computer Science II

12