

Digitalelektronik

Beginn 10:00

Hintergrund

- **1977** Beginn mit Digitalelektronik
- **1979** Fortsetzung mit Mikroprozessoren
- **1985** Studium E-Technik/Informationstechnik
- **1990** Promotion Entwurfsautomation
- **1995** Siemens Halbleiter/Infineon
- **2002 FH München (FK06)**
 - Digitalelektronik
 - Mikroprozessortechnik
 - (Informatik, Digitale Signalverarbeitung, Embedded Systems, Netzwerktechnik)

Ablauf

- **Vorlesung online**
 - Zoom
 - Video (Film)
- **Praktikum im „home office“**
 - Hardware-Kit zum Ausleihen (Pfand 50€) Ende Oktober
 - Aktuell 64 Tln, 52 Kits -> Zweiergruppen
 - Entwurfssoftware kostenlos
 - „Aufgabenkorrektur“ per E-Mail

Entwicklung der Digitalelektronik I

- **Digitalwecker neben dem Bett**
 - Vorgefertigte einfache Grundelemente (Gatter)
 - Wenige Elemente (ca. 10)
 - Entwurf manuell (Schaltplan)
 - Standardspannungen und Standardbauteile
 - Oft hoher Ruhestromverbrauch (Geräte am Netz)

Entwicklung der Digitalelektronik II

- **„IBM PC“**
 - Schaltkreise mit komplexen Funktionen verfügbar
 - Mikroprozessor
 - Speicher
 - Grafikgenerator (Bildschirmanzeige)
 - ...
 - „Glue Logic“: Zusammenschaltung der komplexen Elemente mit Hilfe der weiterhin vorhandenen einfachen Grundgatter
 - 100 und mehr Elemente, aber Entwurf weiterhin manuell mittels Schaltplan

Entwicklung der Digitalelektronik III

- **Industriesteuerung (1990)**
 - Einsatz erster frei programmierbarer Schaltkreise als Ersatz vieler einfacher Grundelemente (PAL)
 - Übergang zu niedrigeren Spannungen (Energieverbrauch)
 - Entwurf weiterhin im wesentlichen manuell, sowohl im großen (Schaltplan) als auch für die PALs
 - Übergang zu CMOS-Technologie beendet (nahezu ohne Ruhestrom) -> mobile Geräte werden möglich

Entwicklung der Digitalelektronik IV

- **SoC (System on a Chip)**
 - Rechner auf einem Chip aus komplexen Funktionen
 - Zusätzlich See von einfachen Grundelementen
 - von 1000 bis zu 1.000.000 (jedes Jahr steigend)
 - Entwurf durch „Hardwarebeschreibungssprache“, entspricht einem Compiler (z.B. C-Compiler) für Software
 - Extrem schnelle Reaktionen bzw. Berechnungen möglich
 - Interne Kopplung selbst entworfener Funktionseinheiten an den vorgefertigten Teil (Mikroprozessor, Speicher, ...)
- **Typische Einsatzgebiete**
 - Bildverarbeitung
 - KI

Inhalt

- **Grundlagen**
 - Rechenregeln, Verknüpfungsregeln
 - Boolesche Algebra, Funktionsdarstellungen
- **Schaltungsentwurf**
 - Kombinatorik
 - Sequentielle Schaltungen
- **Bauteile**
 - CPLD/FPGA
- **Entwurfswerkzeuge**
 - Struktur (Schaltplan)
 - Hardwarebeschreibungssprache (VHDL)

Ziele vs. Erwartungen

- **Ziele**
 - 50% **Grundlagen** der Digitalelektronik
 - 50% **Grundlagen** von VHDL
- **Erwartungen an das Modul**
 - Bitte bis zum Sonntag per E-Mail

Grundlagen – Eigenschaften

	Analog	Digital
Signale	Beliebige Werteverläufe	Zwei diskrete Werte
Einheiten	Spannung (Strom), Zeit	wahr, falsch, Zeit
Bauelemente	Widerstände, Kondensatoren, Spulen, Transistoren (Dutzende / Hunderte)	Transistoren, Gatter (Tausende / Millionen)
Entwurf	manuell	automatisch

Grundlagen – Signale und Werte

- **Signal: physikalisch messbar (z.B.: 10 Volt)**
- **Wert: abstrahiertes Signal (z.B.: wahr)**

- **Werte** werden in der Digitalelektronik für die Beschreibung und den Entwurf der Schaltung benötigt.

- **Signale** werden bei der Umsetzung (Realisierung) der entworfenen Schaltung mit Bauelementen benötigt.

Grundlagen – Verknüpfungen

- **Mathematische Grundlage der Digitalelektronik ist die Aussagenlogik**
- **Aussagen können wahr (w) oder falsch (sein)**
- **Aussagen können miteinander verknüpft werden**
- **Resultat kann wieder nur „wahr“ oder „falsch“ sein**
- **Beispiel: UND-Verknüpfung**
 - A1: Schalter S1 ist geschlossen (wahr/falsch)
 - A2: Schalter S2 ist geschlossen (wahr/falsch)
 - A3: Die Lampe brennt (wahr/falsch)

A3 ist **genau dann** wahr, wenn A1 wahr **und** A2 wahr sind.

Grundlagen – Verknüpfungen

- **Darstellung als Wahrheitstabelle (UND)**

A1	A2	A3

Grundlagen - Verknüpfungen

- Eine Wahrheitstabelle stellt eine **Abbildung** von Eingangsvariablen auf Ausgangsvariablen **dar**
- Daher ist auch eine **Darstellung als Funktion** möglich: $A_3 = f(A_1, A_2)$
- Die **Rechenregeln** für diese zweiwertige Funktionen werden als **Boolesche Algebra** bezeichnet

Grundlagen – Begriffe

- In der **Funktionsdarstellung** treten **Variablen** an die Stelle der **Aussagen**
- Die Variablen werden statt mit „w“ und „f“ mit „0“ und „1“ belegt
- Eine Funktion, die von **n Variablen abhängt** wird als **n-stellig** bezeichnet
- Eine Funktion bzw. Variable, die **m Werte** annehmen kann, wird als **m-wertig** bezeichnet
- Eine **Belegung** ist die Zuweisung von Werten an Variablen

Grundlagen – Boolesche Algebra

- Die **UND-Verknüpfung** wird auch als **Konjunktion** bezeichnet
- Die **UND-Verknüpfung** entspricht der **Multiplikation** in der „üblichen“ Algebra (siehe Wahrheitstabelle mit „0“ und „1“ statt „w“ und „f“)
- **Übliche UND-Verknüpfungszeichen sind :**
der „Multiplikationspunkt“ $y = x_1 \bullet x_2$
gar kein Zeichen $y = x_1 x_2$
ein „^“ $y = x_1 \wedge x_2$

Grundlagen – Boolesche Algebra

- **Erweiterung der UND-Verknüpfung auf n Stellen**

- Das Ergebnis ist nur wahr, wenn alle Teilaussagen wahr sind

- Funktionsdarstellung: $y = x_1 x_2 x_3 x_4$

- Wahrheitstabelle:

- Symbol „-“ (don't care)
bedeutet entweder 0 oder 1

x1	x2	x3	x4	y
0	-	-	-	0
-	0	-	-	0
-	-	0	-	0
-	-	-	0	0
1	1	1	1	1

Grundlagen – Boolesche Algebra

- **Einstellige Verknüpfungen**

– Identität: $y=x$

x	y
0	0
1	1

– Negation: $y=\bar{x}$

x	y
0	1
1	0

– Die Negation (Inversion) entspricht in etwa einem Vorzeichenwechsel

Grundlagen – Boolesche Algebra

- Die **ODER-Verknüpfung** wird auch als **Disjunktion** bezeichnet
- Die **ODER-Verknüpfung** entspricht der **Addition** in der „üblichen“ Algebra.
- **Übliche ODER-Verknüpfungszeichen sind :**
das Additionszeichen $y=x_1+x_2$
ein „v“ $y=x_1 \vee x_2$

Grundlagen – Boolesche Algebra

Absorptionen

$$x \cdot x = x$$

$$x \cdot \bar{x} = 0$$

$$x \cdot 1 = x$$

$$x \cdot 0 = 0$$

$$x + x = x$$

$$x + \bar{x} = 1$$

$$x + 1 = 1$$

$$x + 0 = x$$

Doppelte Negation

$$\overline{\bar{x}} = x$$

Distributionen

$$x_1 (x_2 + x_3) = x_1 x_2 + x_1 x_3$$

$$x_1 + (x_2 x_3) = (x_1 + x_2)(x_1 + x_3)$$

Boolesche Algebra

- **Gesetz von Shannon zur Negation einer Funktion**
 - Ersetze alle bisherigen Konjunktionen durch Disjunktionen
 - Ersetze alle bisherigen Disjunktionen durch Konjunktionen
 - Negiere alle Variablen

$$\overline{f}(x_1, \dots, x_n, \bullet, +) = f(\overline{x}_1, \dots, \overline{x}_n, +, \bullet)$$