

# Eingebettete Systeme WS 14/15

- Vorlesung: Di 12:00-13:30h, Do 12:00-13:30h, Zuse HS
- Dozent: Prof. Dr. Reiner Kolla
- Assistent: M.Sc. Johannes Mühr
- Übungen: Di 14:00-16:00h ÜR I, Do 14:00h-16:00h ÜR I



## Bücherliste

- Jürgen Teich, Christian Haubelt  
*Digitale Hardware Software Systeme -- Synthese und Optimierung.*  
Springer – Lehrbuch 2. Auflage 2007.
- Jane W. S. Liu  
*Real-Time Systems*  
Prentice Hall 2000.

## 1. Modellierung eingebetteter Systeme

Zur Vorlesung  
Eembedded Systems  
WS 14/15  
Reiner Kolla



## 1. 1 Einführung

Zur Vorlesung  
Eembedded Systems  
WS 14/15  
Reiner Kolla



## Was sind Eingebettete Systeme?

### Definitionsversuch

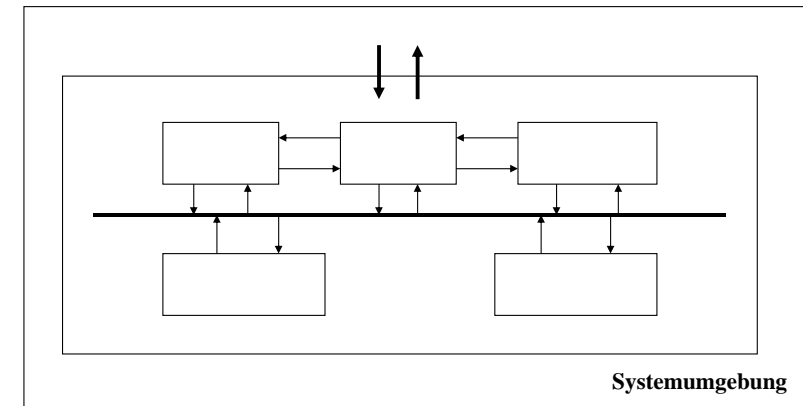
Ein **eingebettetes System** ist ein informationsverarbeitendes System bestehend aus Hardware und Software, das in einem technischen Gesamtsystem eingebettet ist, dessen primärer Zweck nicht das Verarbeiten und Aufbereiten von Information ist.

Durch die hohe Integrationsdichte haben komplexe Hardware / Software Systeme den Einzug in viele Produkte des täglichen Lebens gefunden: Automobile, Flugzeuge, Unterhaltungselektronik, medizinische Geräte, Kommunikationsgeräte ....

Durch die zunehmende Komplexität hat sich der Entwurf eingebetteter Systeme von analogen Schaltungen und Assemblerprogrammen auf einfachen Mikrocontrollern weg zum Problem hin entwickelt, komplexe Hardware/Software Systeme mittels leistungsfähiger Werkzeuge zu entwickeln und zu optimieren.

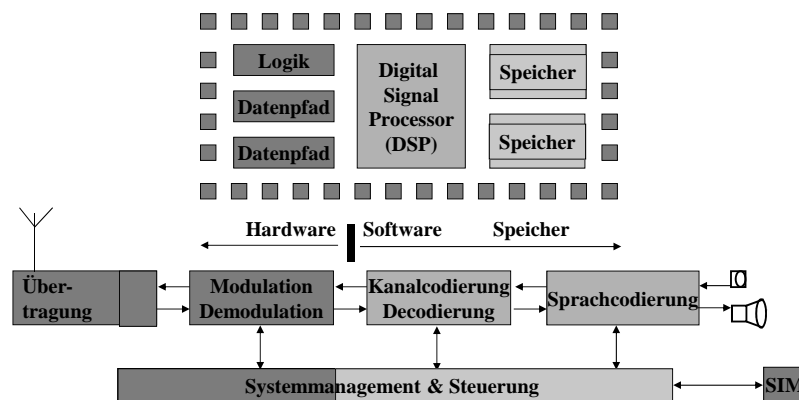
5

## Allgemeine Struktur eines Eingebetteten Systems



6

## Beispiel: Handy nach dem GSM Standard



7

## Eingebettete Systeme vs. konventionelle Rechner

### feste Funktion:

Zum Zeitpunkt des Entwurfs ist die Funktion, die das System übernehmen soll, bekannt.

### reaktives Verhalten:

Das System muss auf Aktionen der Umgebung unter festen zeitlichen Bedingungen reagieren.

### Co-Design:

Hardware und Software werden aufeinander abgestimmt entwickelt und optimiert.

### universell und offen:

Zum Zeitpunkt des Entwurfs sind die Funktionen, die das System übernehmen soll, weitgehend offen.

### interaktives Verhalten:

Das System sollte mit der Umgebung unter sehr losen zeitlichen Bedingungen interagieren.

### unabhängiges Design:

Hardware und Software werden getrennt voneinander entwickelt und unabhängig voneinander optimiert.

8

## Was sind Eingebettete Systeme -- ff

### Fazit

Der Begriff **eingebettetes System** ist sehr weit gestreut. Betrachtet man die Gegenüberstellung auf der letzte Folie genauer, so sieht man, dass auch konventionelle Computer eingebettete Systeme in sich integriert haben (Grafik-, Netzwerkkarten), was der eingangs getroffenen Definition ein wenig widerspricht.

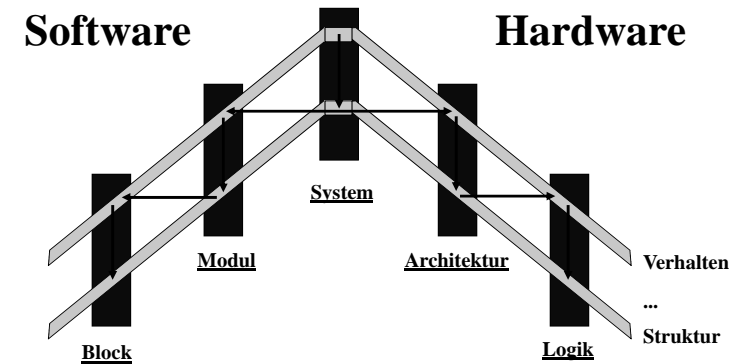
Wir können aber die beiden wichtigsten Merkmale herausstellen:

- reaktives Verhalten unter Echtzeitbedingungen
- gleichzeitige Entwicklung von Hardware und Software

Die Entscheidung, welche Funktionen in Hardware und welche in Software auf einem oder mehreren Mikrocontrollern realisiert werden fällt oft sogar erst während der Entwicklung und Optimierung des Entwurfs (HW/SW Codesign).

9

## Abstraktionsebenen und Sichten des HW/SW Co-Design



10

## Abstraktionsebenen -- Software Bereich

- **System:** die **Modelle** der Systemebene beschreiben das zu entwerfende Gesamtsystem auf der Ebene von Netzwerken aus komplexen, miteinander kommunizierenden Teilsystemen.
  - **Modul:** die Modulebene gehört zum Softwarebereich. Die Modelle der Modulebene beschreiben komplexe Funktionen und ihre Interaktionen
  - **Block:** die Blockebene gehört ebenfalls zum Softwarebereich. Die entsprechenden Modelle beschreiben Programme bis hin zu Instruktionen, die auf der zugrundeliegenden Rechnerarchitektur elementare Operationen ausführen.

11

## Abstraktionsebenen -- Hardware Bereich

- **System:** die **Modelle** der Systemebene beschreiben das zu entwerfende Gesamtsystem auf der Ebene von Netzwerken aus komplexen, miteinander kommunizierenden Teilsystemen.
  - **Architektur:** die Architecturebene gehört zum Hardwarebereich. Die Modelle dieser Ebene beschreiben kommunizierende Blöcke, die alle nebenläufig komplexe Operationen ausführen.
  - **Logik:** die Logikebene gehört ebenfalls zum Hardwarebereich. Die Modelle dieser Ebene beschreiben verbundene Gatter und Register, die Boolesche Funktionen berechnen.

12

## Entwurfsrepräsentation

- Modell: unter einem Modell versteht man die formale Beschreibung eines Systems (oder Teilsystems). Wie genau ein Modell ein System beschreibt, hängt von der entsprechenden Abstraktionsebene ab.

Man unterscheidet in der Regel zwischen

- **zustandsorientierten** (z.B. auf endlichen Automaten basierende)
- **aktivitätsorientierten** (z.B. auf Datenflussgraphen basierende)
- **struktuorientierten** (z.B. auf Blockschaltbildern basierende)
- **datenorientierten** (z.B. als Kollektion von Datenobjekten mit Relationen)

Modellen.

13

automatischen

## Aufgabe der Systemsynthese

... aktuelles Forschungsgebiet

- Festlegung der Komponententypen, die in der Implementierung verwendet werden (Allokation), z.B. Mikroprozessor, ASIC, Speicherbausteine...
- Festlegung der Anzahl der jeweiligen Komponenten, Auswahl und Dimensionierung der Verbindungsstruktur (Allokation)
- Zuordnung der Variablen zu Speicherbausteinen, Operationen zu Funktionsbausteinen und Kommunikationen zu Bussen (Binding). Hierbei sind Realisierung in Hardware und in Software gegeneinander abzuwägen (Hardware/Software-Partitionierung).
- Erstellung eines Ablaufplans: Wann wird welche Aufgabe durch seine Ressource ausgeführt?
- Schätzung von Systemeigenschaften, um eine vernünftige Exploration des Entwurfsraumes zu ermöglichen. Gerade auf den obersten Entwurfsebenen werden grundlegende Entwurfsentscheidungen getroffen, die die Leistungsfähigkeit und die Kosten des ganzen Systems bestimmen.

14

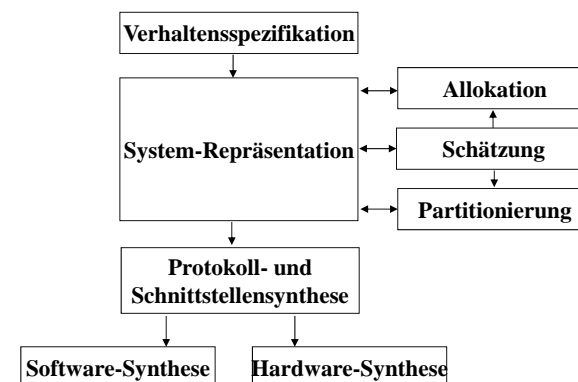
## Mögliche Zielarchitekturen

- ein einziger Mikroprozessor, Signal- oder Bildprozessor
- mehrere parallel arbeitende programmierbare Prozessoren
- eine Erweiterung der oben genannten Architekturen mit spezialisierten funktionalen Einheiten, (z.B. für die Bewegungsschätzung bei Bildverarbeitung)
- eine reine spezialisierte Hardware-Lösung, die an den Algorithmus genau angepasst ist.

hohe Kosten >>> niedrige Kosten

15

## Entwurfsablauf auf Systemebene



16

## Aufgabe der Architektursynthese

... ist es, eine strukturelle Sicht aus einer Verhaltensbeschreibung, zum Beispiel formuliert in VHDL, zu generieren.

➤ Bestimmung des Grads der Parallelität.

### Wesentlichste Aufgaben

- Identifikation von Hardware-Elementen, die die spezifizierten Operationen ausführen können (Allokation)
- Ablaufplanung zur Bestimmung der Zeitpunkte, an denen die Operationen ausgeführt werden.
- Binding, d.h. Zuordnung von
  - Variablen zu Speichern
  - Operationen zu funktionalen Einheiten
  - Kommunikationskanäle zu Bussen

17

## Beispiel zur Architektursynthese:

Verhaltenbeschreibung:

Differentialgleichungslöser für

- ⊙  $y' + 3xy' + 3y = 0$
- ⊙ im Intervall  $[x_0, a]$
- ⊙ mit der Schrittweite  $dx$  und
- ⊙ den Anfangswerten  $y(x_0) = y_0$  und  $y'(x_0) = u$

mit Hilfe der Euler-Methode, die iterativ den Lösungswert durch

```
x:=x0; y:=y0; ypalt:=u
loop ypneu := ypalt - 3*(x*ypalt + y)*dx;
  y := y + ypalt*dx;
  x := x + dx;
  ypalt := ypneu;
until x>a;
return y;
```

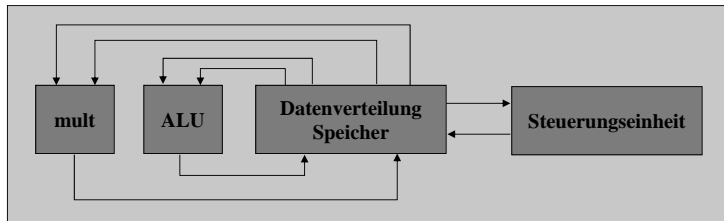
berechnet.

18

## Architektursynthese:

... einer der zentralen Punkte der Vorlesung

```
x:=x0; y:=y0; ypalt:=u
loop ypneu := ypalt - 3*(x*ypalt + y)*dx;
  y := y + ypalt*dx;
  x := x + dx;
  ypalt := ypneu;
until x>a;
return y;
```



19

## Aufgabe der Logiksynthese

... siehe Vorlesung: Logiksynthese

... ist, ausgehend von

- ⊙ Booleschen Ausdrücken (bei kombinatorischem Verhalten)
- ⊙ endlichen Automaten (bei sequentiellm Verhalten),

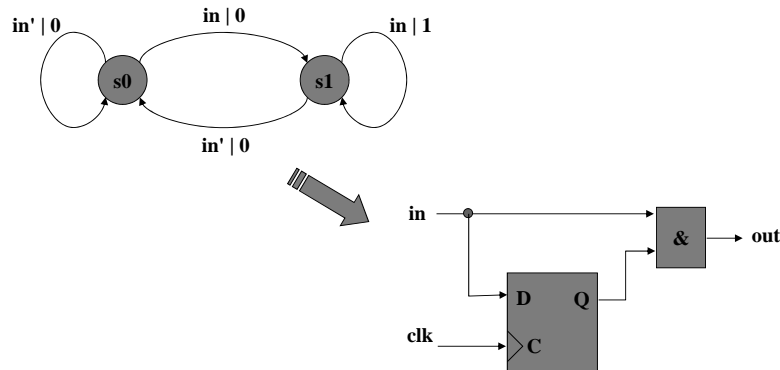
die Generierung einer strukturellen Sicht auf Logikebene, also eine Realisierung durch Gatter / FF.

Die zu lösenden Teilprobleme sind

- ⊙ Zustandsminimierung und Zustandskodierung
- ⊙ Optimierung Boolescher Ausdrücke
- ⊙ Bindung an eine Zellenbibliothek

20

## Logiksynthese: Beispiel



21

## Aufgabe der Modulsynthese

..... ist es, eine strukturelle Sicht aus einer Verhaltensbeschreibung (Menge von untereinander kommunizierender Prozesse), zum Beispiel in C++ formuliert, zu generieren.

➤ (insbesondere) Bestimmung des Grads der Parallelität.

### Wesentlichste Aufgaben

- Prozessablaufplanung hinsichtlich einer guten Ausnutzung paralleler funktionaler Einheiten des Zielprozessors

Bemerkung: Allokation und Binding ist hier weniger ein Thema, da die Maschinenhardware in der Regel weitgehend fest vorgegeben ist.

22

## Aufgabe der Blocksynthese

... siehe auch Vorlesung: Compilerbau

... ist es ausgehend von

- einem Programm(stück) in einer höheren Programmiersprache die Generierung eines Maschinenprogramms.

### Wesentlichste Aufgaben

- Programmtransformationen zur optimalen Ausnutzung der Fließbandverarbeitung innerhalb des Zielprozessors und zur Minimierung der Speicherzugriffe (Ablaufplanung)
- Optimale Befehlsauswahl und Registervergabe (Binding)
- Minimierung der Programmlänge

Bemerkung: Allokation ist hier auch kein Thema, da die Maschinenhardware in der Regel fest vorgegeben ist.

23

## Ausblick zum Inhalt der Vorlesung

- Modelle zur Spezifikation eingebetteter Systeme (Petri-Netze, Datenflussgraphen, Sequenzgraphen)
- Verifikation eingebetteter Systeme
- Implementierungsplattformen für eingebettete Systeme
  - ASICs, FPGAs für Hardware
  - Mikrocontroller, Echtzeitbetriebssysteme für Software
- Grundlegende Algorithmen zur Ablaufplanung, Allokation und Binding
  - Statisch (HW)
  - Dynamisch (SW)
- Architektursynthese
- Softwaresynthese

24