

# Embedded Systems / Eingebettete Systeme

BSc-Studiengang Informatik  
Fachbereich Campus Minden

Matthias König



**FH Bielefeld**  
University of  
Applied Sciences

# Organisatorisches: Daten

- Verantwortlich: Matthias König (Kontaktdaten, s. FH Bielefeld Seiten)
- Vorlesung
  - Dienstag, 09:45-11:15, H 10
- Praktika
  - Gruppe 1, Mittwoch, 09:45-11:15, D 328 (Sprute)
  - Gruppe 2, Mittwoch, 14:00-15:30, D 328 (Sprute)
  - Gruppe 3, Mittwoch, 11:30-13:00, D 327 (Rasch)
  - Gruppe 4, Dienstag, 11:30-13:00, D 327 (Rasch)

I. Praktikum KW 18 2018

# Organisatorisches: Bewertung

- Vorlesung mit Abschlussklausur (Prüfungsleistung)
  - min. 50% der Punkte müssen erreicht werden.
  - max. 3 Versuche
  - keine Hilfsmittel
- Praktikum (Studienleistung)
  - Anwesenheitspflicht > 75%
  - min. 50% der Punkte müssen erreicht werden.
  - Abnahme und Punkte im Praktikum gemäß Abgabetermin
- Note: arithmetisches Mittel aus Klausur- und Praktikumsnote

# Voraussetzungen

- Insbesondere Wissen aus
  - Technischer Informatik
  - Systemprogrammierung
  - Mathematik für Informatiker
- Generell wichtig
  - Textverständnis (Deutsch und Englisch)

# Inhalt

- Vorlesung
  - Grundlagen eingebetteter Systeme
- Praktikum
  - Programmierung eingebetteter Systeme

# Literatur

- Zur Vorlesung:
  - u.a. Marwedel, Eingebettete Systeme, URL: <http://www.springerlink.com/content/978-3-540-34048-5/> (online aus FH Netz)
- Zum Praktikum:
  - TI Launchpad, URL: <http://www.ti.com/launchpad>
  - Fritzing, <http://fritzing.org/>

# Veranstaltungsinhalt

- Überblick und Anwendungsgebiete
  - Besonderheiten und Anforderungen
  - Ausflug in die Grundlagen der Elektronik
  - Zusammenspiel Software und Hardware
  - Entwurfsschritte
  - Spezifikation und Modellierung
  - Realisierung und Implementierung
  - Validierung und Evaluation
- 
- heute

# Eingebettetes System / embedded system

- bezeichnet einen elektronischen Rechner, der *in einen technischen Kontext eingebunden ist*. Dabei hat der Rechner entweder die Aufgabe, das System, *in das er eingebettet ist*, zu **steuern**, zu **regeln** oder zu **überwachen**.

[Quelle: Wikipedia, Eingebettetes System, URL: [de.wikipedia.org/wiki/Eingebettetes\\_System](http://de.wikipedia.org/wiki/Eingebettetes_System), Zugriff 26.03.2014]

# Wozu? - Beispiele

- **Automotive:** Steuergerät
- **Avionik:** Bordelektronik
- **Raumfahrt:** Satellitensteuerung
- **Haushaltsgeräte:** Waschmaschine
- **Unterhaltungselektronik:** Fernseher
- **Medizintechnik:** Insulinpumpe
- **Robotik:** autonomer Staubsauger
- **Steuerungssysteme:** Fabriksteuerung
- **Mess- und Regelsysteme:** Kraftwerksregelung
- **Haustechnik:** Heizungssteuerung
- **Kommunikation:** Internet of Things

# Warum Eingebettete Systeme?

- Eingebettete Systeme sind Schlüsseltechnologie für Innovation
- Daher für Industrie- und Wirtschaftsentwicklung bedeutsam
- Tragende Rolle bei den Technologietrends
  - Energieeffizienz (z.B. *energy harvesting*)
  - Vernetzung (z.B. *Things* des *Internet of Things*)
  - Mobilität
  - Gesundheitsvorsorge (z.B. *Telemedizin*)

# Aus Sicht der Industrie

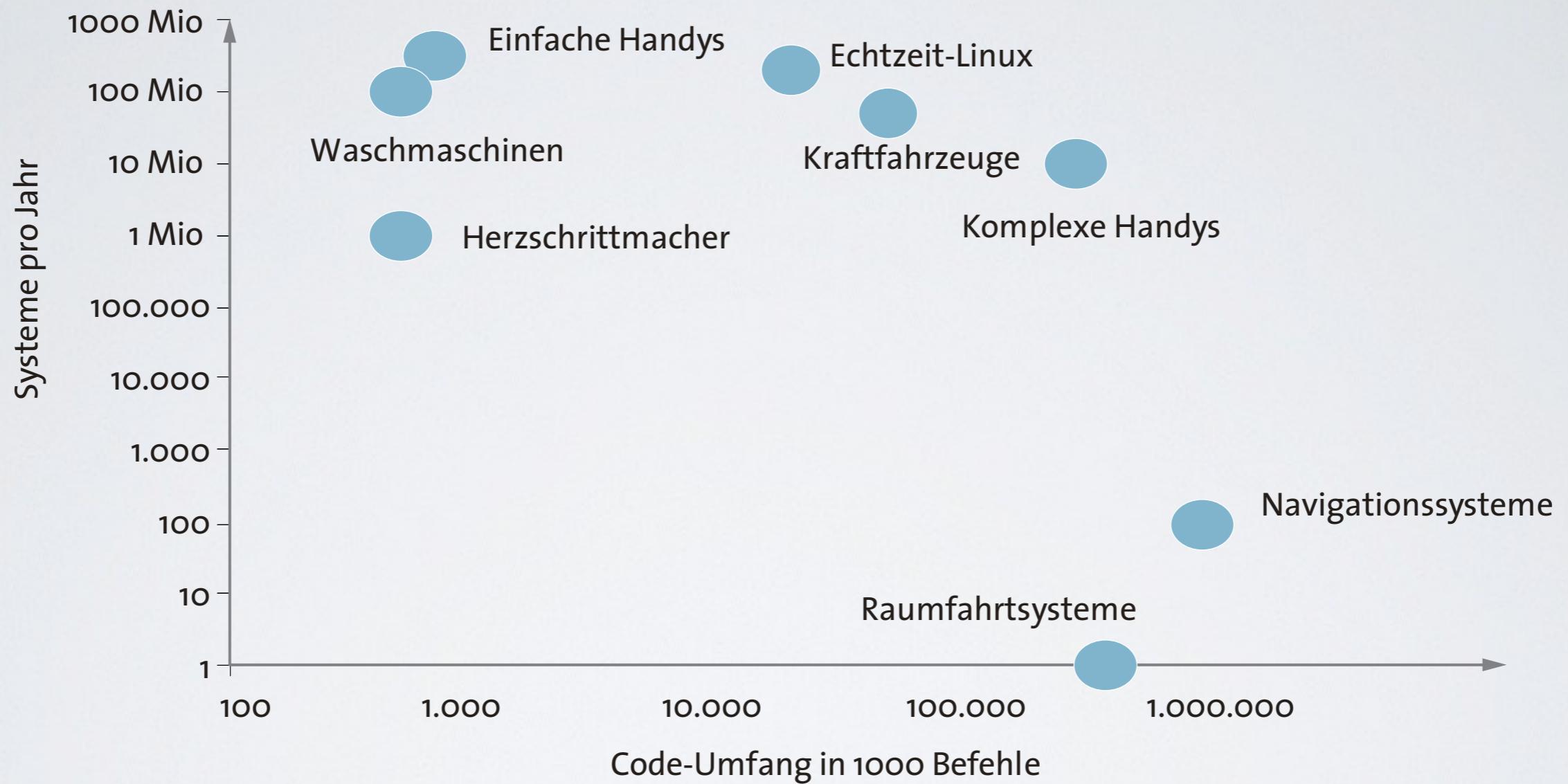
*“Der Umsatz mit Eingebetteten Systemen (ES) wird in Deutschland 2012 erstmals die Grenze von 20 Milliarden Euro überschreiten. Der Markt wird um rund 6 Prozent wachsen,...”*

[Quelle: Bitkom, Pressemitteilung 27.02.2012, URL: [www.bitkom.org/de/presse/8477\\_71318.aspx](http://www.bitkom.org/de/presse/8477_71318.aspx), Zugriff 26.03.2014]

*“... wir denken an PCs, große IT- Systeme und online Internet-Anwendungen. Derartige IT-Systeme machen aber nur weniger als zwei Prozent der produzierten Mikroprozessoren aus. Die meisten Mikroprozessoren sind in Systemen für Autos, mobile Kommunikation, Haushaltsgeräte, Flugzeuge, Roboter, Verkehrsmanagement, Cameras oder Medizintechnik verbaut.*

*...vor allem der hohe Bedarf an Ingenieuren in dieser Branche ist deutlich sichtbar.”*

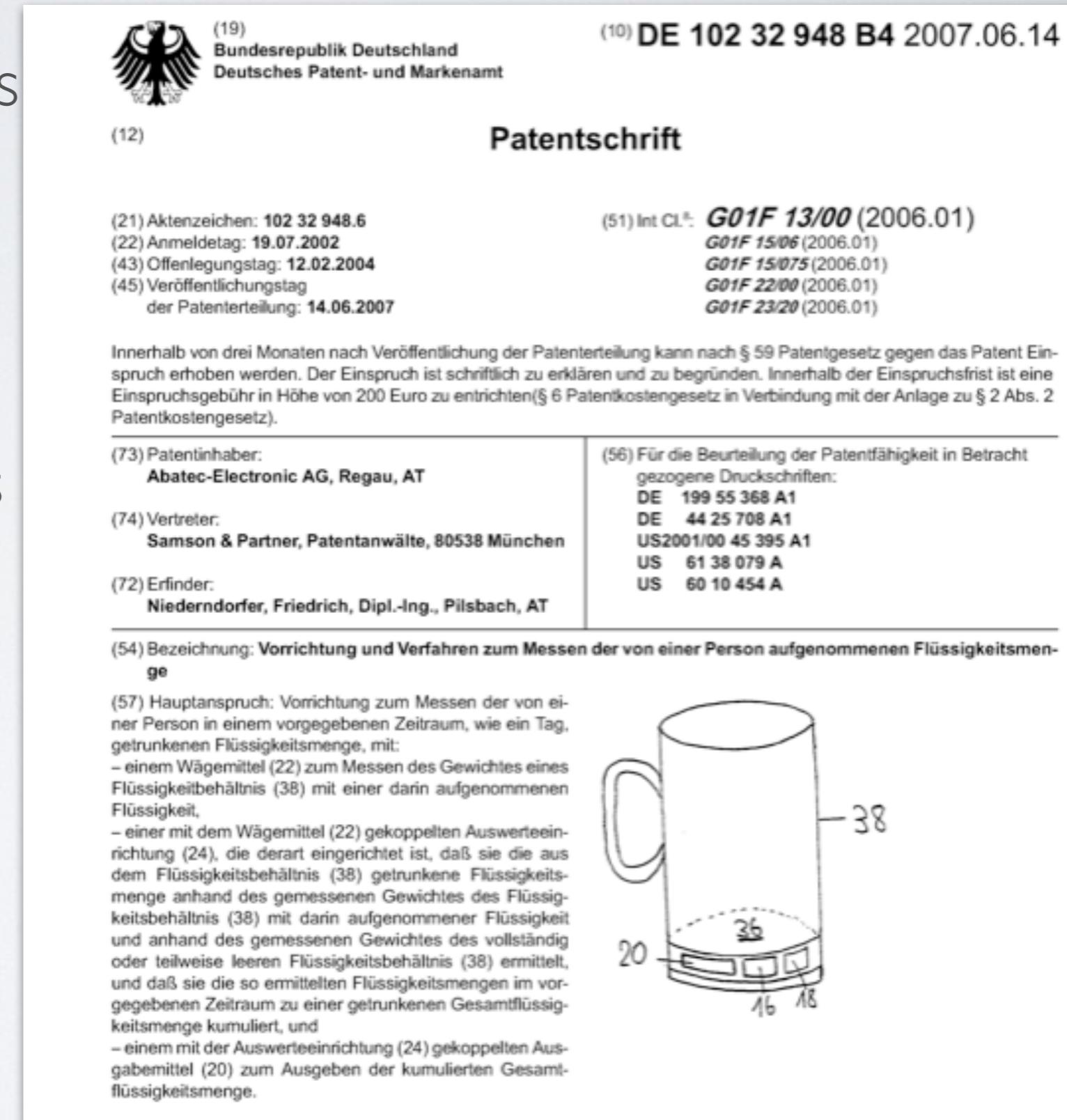
[Quelle: Bitkom, Eingebettete Systeme – Ein strategisches Wachstumsfeld für Deutschland, 2010, URL: <https://www.bitkom.org/Publikationen/2010/Leitfaden/Eingebettete-Systeme-Anwendungsbeispiele-Zahlen-und-Trends/EingebetteteSysteme-web.pdf>, Zugriff: 26.03.2014]



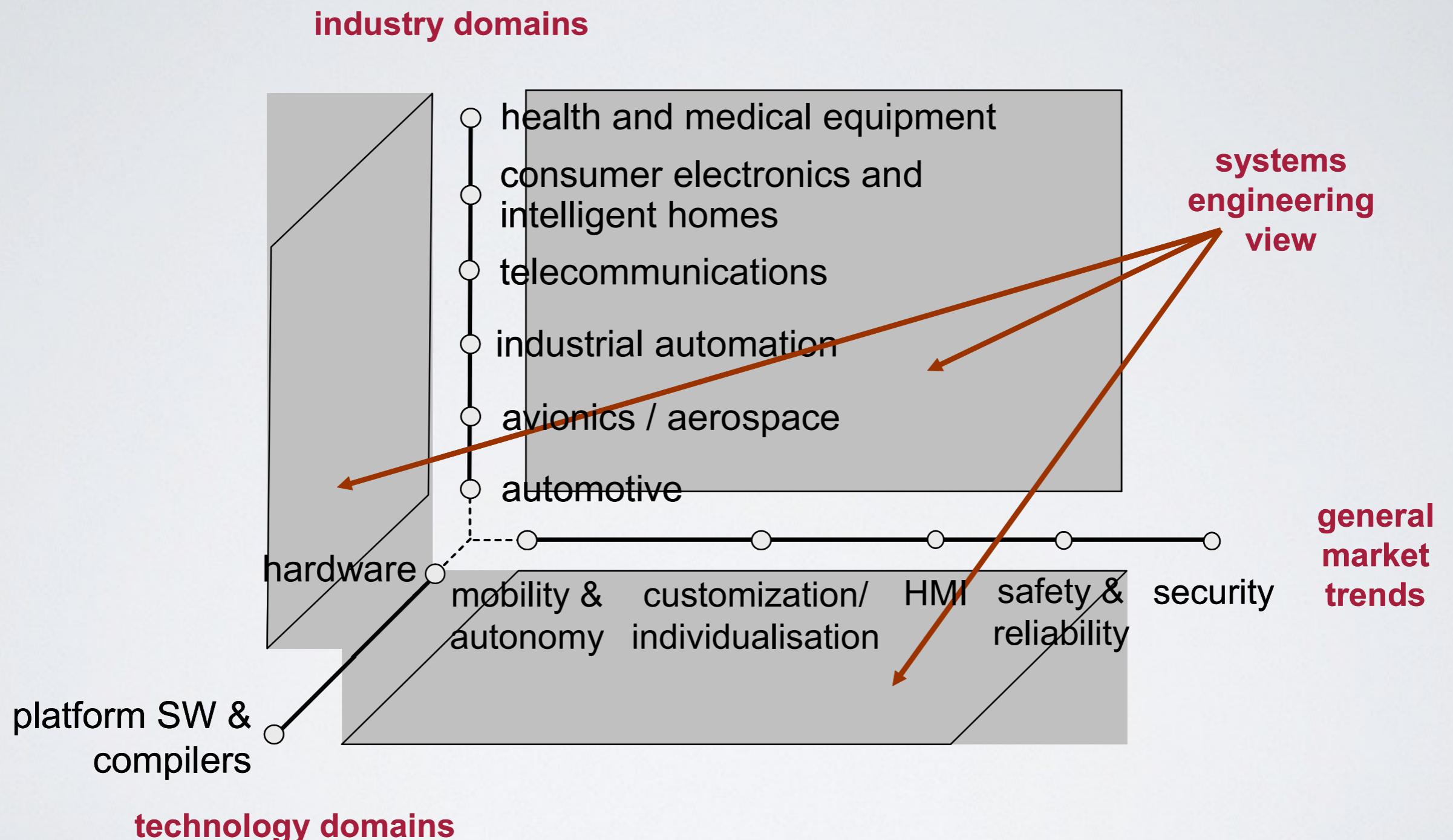
[Quelle: Bitkom, Eingebettete Systeme – Ein strategisches Wachstumsfeld für Deutschland, 2010, URL: <https://www.bitkom.org/Publikationen/2010/Leitfaden/Eingebettete-Systeme-Anwendungsbeispiele-Zahlen-und-Trends/EingebetteteSysteme-web.pdf>, Zugriff: 26.03.2014]

# Eingebettete Systeme und Patente

- Sogenannte **Technizität** als eine Bedingung (neben Neuheit, Erfinderischer Tätigkeit, gewerblicher Anwendbarkeit) der Patentierbarkeit ist für das (gesamte) Eingebettete System **gegeben**.
- Viele Erfindungen von Eingebetteten Systemen patentiert.



# Treiber für Eingebettete Systeme



[Quelle: Braun et al, Study of Worldwide Trends and R&D Programmes in Embedded Systems in View of Maximising the Impact of a Technology Platform in the Area, 2005: URL: [http://www.artemis-austria.net/uploads/media/FAST\\_final-study-181105\\_en.pdf](http://www.artemis-austria.net/uploads/media/FAST_final-study-181105_en.pdf), Zugriff 26.03.2014]

# Weltweiter Markt für Embedded Systems

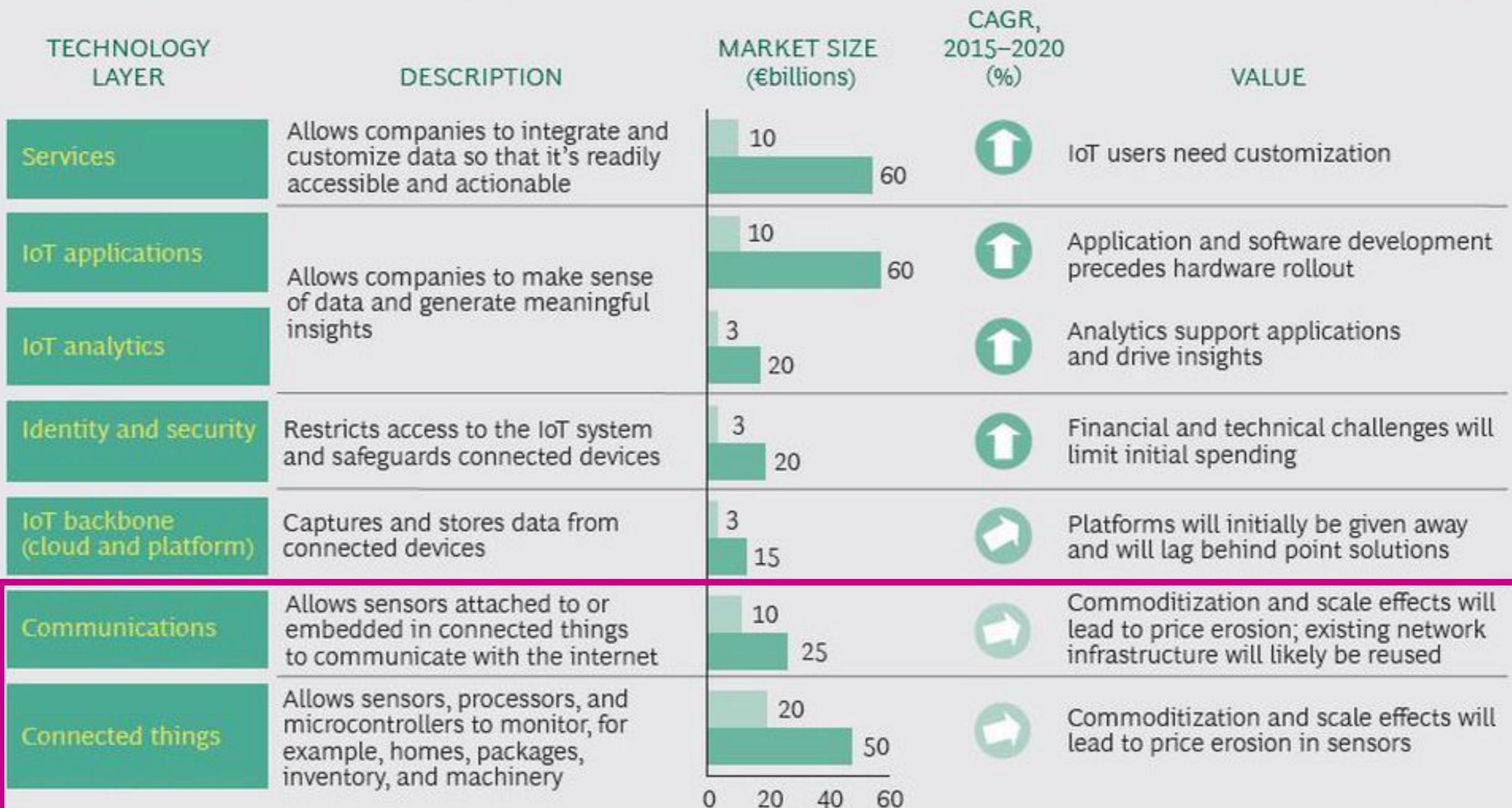
Industrie	2012	2015
Automotive	78	103
Industrie	159	207
Gesundheit	54	75
Energie	65	181
Kommunikation	459	614
Consumer	285	333
<b>Gesamt</b>	<b>1.100</b>	<b>1.514</b>

in Milliarden US\$

[Quelle: Petriisans et. al, "Design of Future Embedded Systems Toward Systems of Systems", 2012, URL: [cordis.europa.eu/fp7/ict/embedded-systems-engineering/documents/idc-study-presentation.pdf](http://cordis.europa.eu/fp7/ict/embedded-systems-engineering/documents/idc-study-presentation.pdf), Zugriff: 06.04.2016]

# Prediction of IoT-markets

## EXHIBIT 1 | Services and IoT Applications and Analytics Will Capture Some 60% of IoT Spending

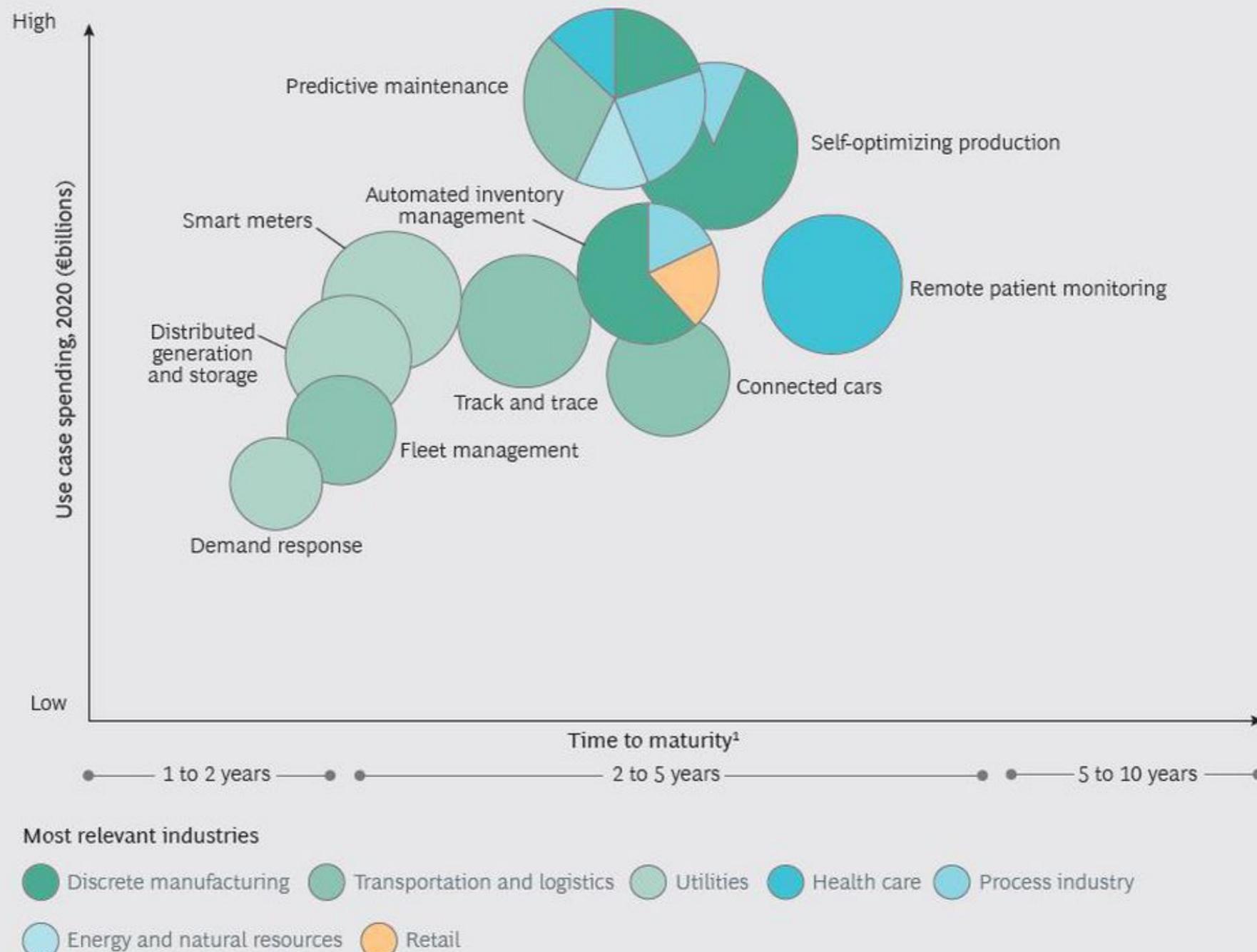


2015 2020 CAGR, 2015–2020 = ~40% CAGR, 2015–2020 = ~30% CAGR, 2015–2020 = ~20%

Sources: IDC; Gartner; ABI Research; BCG Internet of Things buyer survey; expert interviews; BCG analysis.

# Prediction of IoT-markets

EXHIBIT 2 | Ten Use Cases Will Drive IoT Growth Through 2020



# Was charakterisiert Eingebettete Systeme?

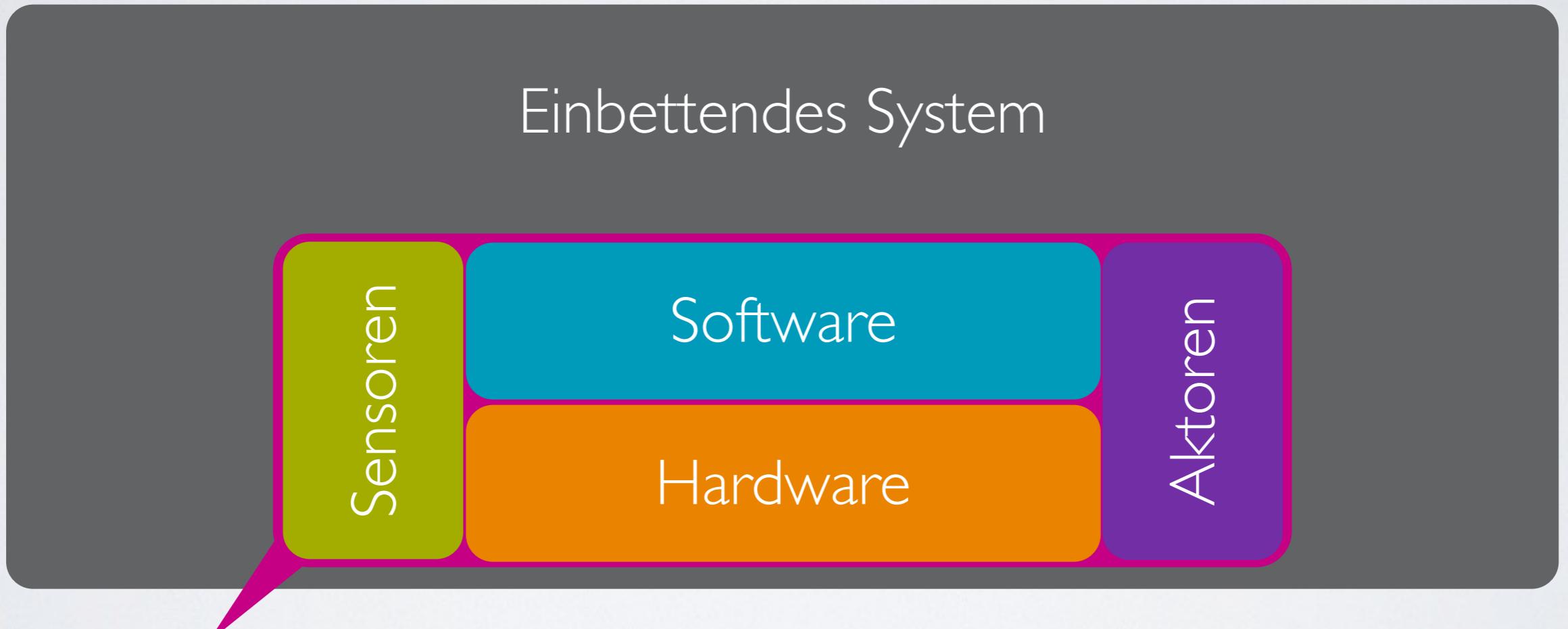
- “Eingebettete Systeme sind **informationsverarbeitende Systeme**, die in ein größeres Produkt **integriert** sind.”

[Quelle: Marwedel, Eingebettete Systeme, Springer-Verlag, 2008]

- “Ein HW- und SW-System / Eingebettetes System besteht im Allgemeinen aus **Hardware, Software und eingebetteten Komponenten**. Ein Projekt, welches als Projektgegenstand ein HW- und SW-System / Eingebettetes System hat, wäre also zum Beispiel die Entwicklung des Eurofighters oder eines Schiffes. Weiterhin wird ein HW- und SW-System / Eingebettetes System charakterisiert durch die **Erfassung der Umwelt über Sensoren und Aktoren zur Interaktion mit seiner physischen Umgebung**. Dadurch werden auch kleinere Systeme adressiert, wie z.B. ein Mikrocontroller, der mit Hilfe seines Programms die Airbagauslösung im Kraftfahrzeug steuert.”

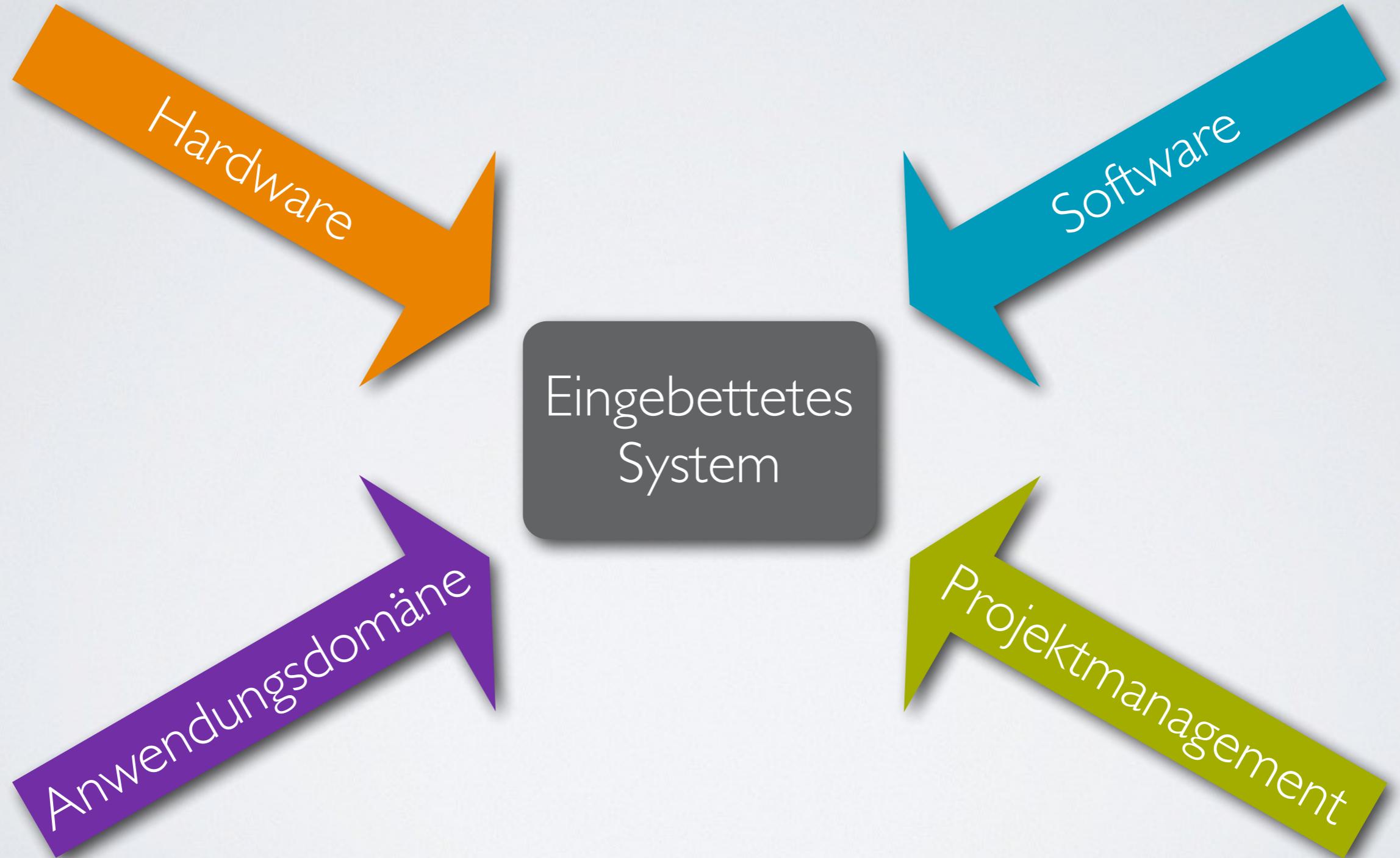
[Quelle: V-Modell® XT, Version 1.3, URL: <ftp://ftp.tu-clausthal.de/pub/institute/informatik/v-modell-xt Releases/1.3/V-Modell%20XT%20HTML/7a0a11a076fa61b.html#toc67>, Zugriff: 26.03.2014]

# Teil eines Gesamtsystems



Eingebettetes System

# Umsetzung interdisziplinäres Wissens



# Welche Anforderungen?

## Verlässlich hinsichtlich

- Zuverlässigkeit (nicht ausfallen)
- Wartbarkeit (schnell repariert bei Ausfall)
- Verfügbarkeit (hohe Zuverlässigkeit, hohe Wartbarkeit)
- Sicherheit (keinen Schaden bei Ausfall)
- Integrität (Gewährleistung von Daten- und Kommunikationssicherheit)

[Quelle: Marwedel, Eingebettete Systeme, Springer-Verlag, 2008]

# Beispiel: Fehlende Sicherheit

Heise online 2013, *Vaillant-Heizungen mit Sicherheits-Leck:*

“Die Vaillant-Heizungsanlagen des Typs ecoPower 1.0 enthalten ein hochkritisches Sicherheitsloch, durch das ein Angreifer die Anlage über das Internet ausschalten und potenziell beschädigen kann. In einem Informationsschreiben rät der Hersteller seinen Kunden daher zu einem drastischen Schritt: Sie sollen den Netzwerkstecker ziehen und auf den Besuch eines Servicetechnikers warten.”

Quelle: <http://www.heise.de/newsticker/meldung/Vaillant-Heizungen-mit-Sicherheits-Leck-1840919.html> [15.04.2013]

# Keine Verlässlichkeit birgt Schadenspotential

Beispiel: Trägerrakete Ariane 5, V88

- Exception in Software führt zur Selbstzerstörung
- Fehler im Softwaremodul der Inertialplattform
- Arithmetischer Überlauf bei Umwandlung einer 64-Bit-Gleitkomma-Variablen in eine vorzeichenbehaftete 16-Bit-Ganzzahl
- (Unbehandelte) Fehlerbehandlung lieferte Diagnosedaten anstelle Flugdaten.
- Falsche Interpretation der Daten durch Bordcomputer führt zur Kursabweichung und letztlich Selbstzerstörung.
- Frühere Softwaretest mit Parametern, die Fehlerbedingungen nicht umfassten.
- Software wurde von Vorgängermodell Ariane 4 übernommen.
- Gesamtverlust: ca. 290 Millionen Euro

# Welche Anforderungen?

**Effizient** hinsichtlich

- Energieverbrauch (Energie kostet...)
- Codegröße (z.B. Festplatte oft ungewünscht wg. Verschleiß)
- Laufzeit (Einhalten von Zeitbedingungen)
- Gewicht (z.B. leicht trag- oder verbaubar)
- Preis (nur Teil eines System)

[Quelle: Marwedel, Eingebettete Systeme, Springer-Verlag, 2008]

# Welche Anforderungen?

- **Einzwecksystem:** Eingebettetes System übernimmt oft genau eine bestimmte Aufgabe
  - vermindert Ressourcenanforderungen
  - verbessert Effizienz
- “Nicht mehr als nötig” für die dedizierte Aufgabe
- oft keine Änderungen nach Fertigstellung möglich
- oft **Echtzeitbedingungen**

# Welche Arten unterscheidet man?

- **Transformative** und **reaktive** Systeme
  - transformative: Eingabe, Berechnung, Ausgabe
  - reaktive: dauerhafte Interaktion mit seiner Umwelt
- Systeme mit Echtzeit-Anforderungen
  - **harte Zeitbedingungen** (Nichteinhalten ist Fehler)
  - **weiche Zeitbedingungen** (z.B. statistisch im Rahmen)

[Quelle: Lee and Seshia, Introduction to Embedded Systems, A Cyber-Physical Systems Approach, <http://LeeSeshia.org>, ISBN 978-0-557-70857-4, 2011]

# Unterschiede bei der Komplexität

## Hardware

Large Embedded Systems  
(32 Bit)

Small Embedded Systems  
(8 und 16 Bit)

## Software

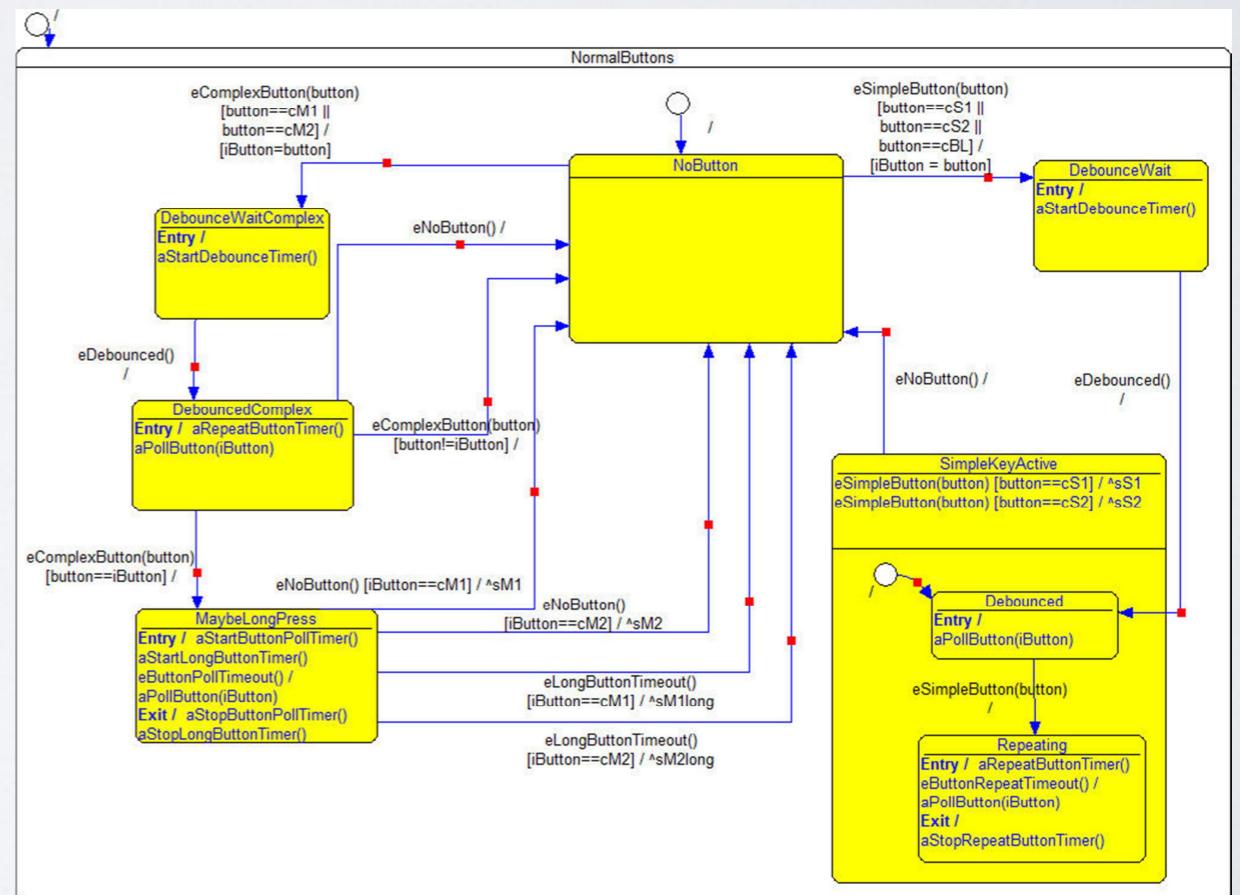
High level programming,  
use of operating systems

Low level programming



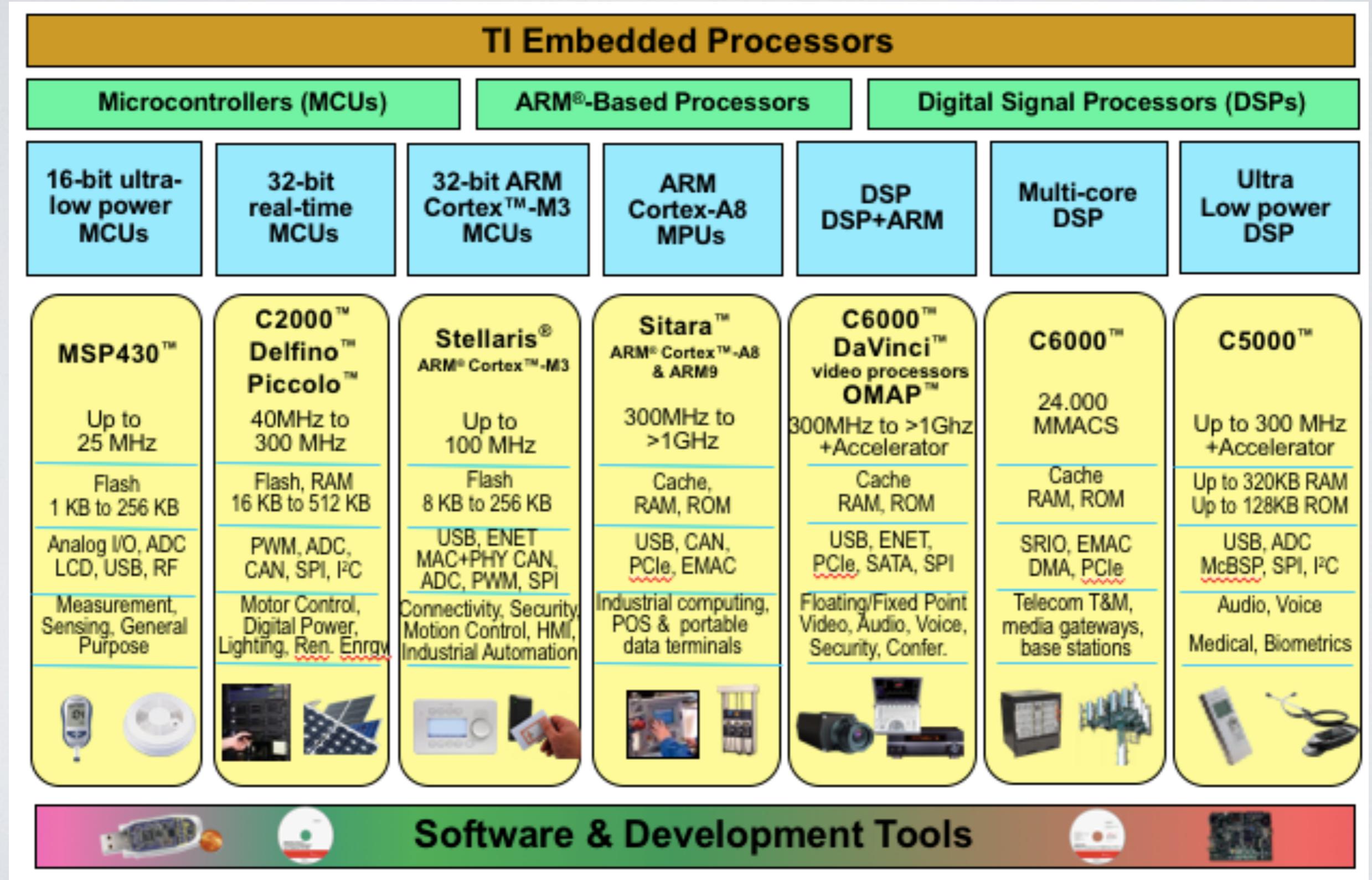
# Bandbreite - Beispiel

- iPhone 6S, 32Bit, z.B. 64 GByte Flash
- ARM Cortex A9-Dual, 512 MByte
- SDK mit Simulator
- Chronos, 16Bit, 32 KByte Flash
- CC430, 4KByte
- verschiedene IDEs



[Quelle: Holmberg, IAR Systems, IAR visualSTATE and TI eZ430-Chronos, URL: <http://www.iar.com/Products/IAR-visualSTATE/Design-your-eZ430-Chronos-watch-with-IAR-visualSTATE/TI-Chronos-Watch-Application-Example/>, Zugriff: 26.03.14]

# Beispiel: Produktvielfalt eines Anbieters



# Beispiele für Einsteigerplattformen: LaunchPad

 **TEXAS INSTRUMENTS**     Everything ▾ Search     Login / Register

Products Applications & designs Tools & software Support & community Sample & buy About TI     Cart English myTI

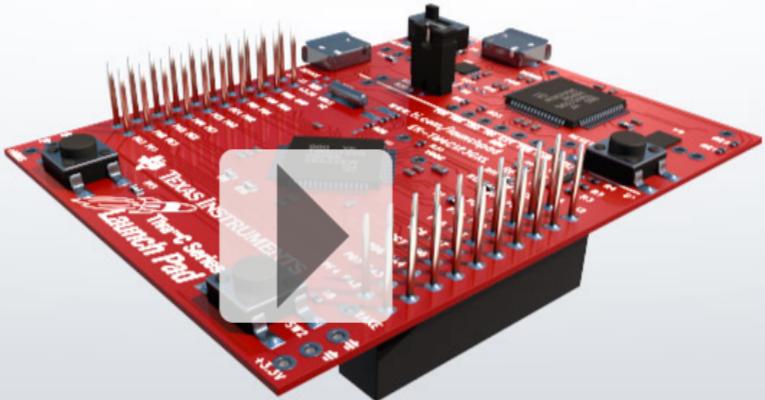
## TI LaunchPad

### Develop. Make. Innovate.

Get started with microcontroller LaunchPad Evaluation Kits from Texas Instruments.

Choose from a variety of low-cost kits & BoosterPack plug-in modules.

Scalable software tools provide multiple points of entry for programming your LaunchPad.



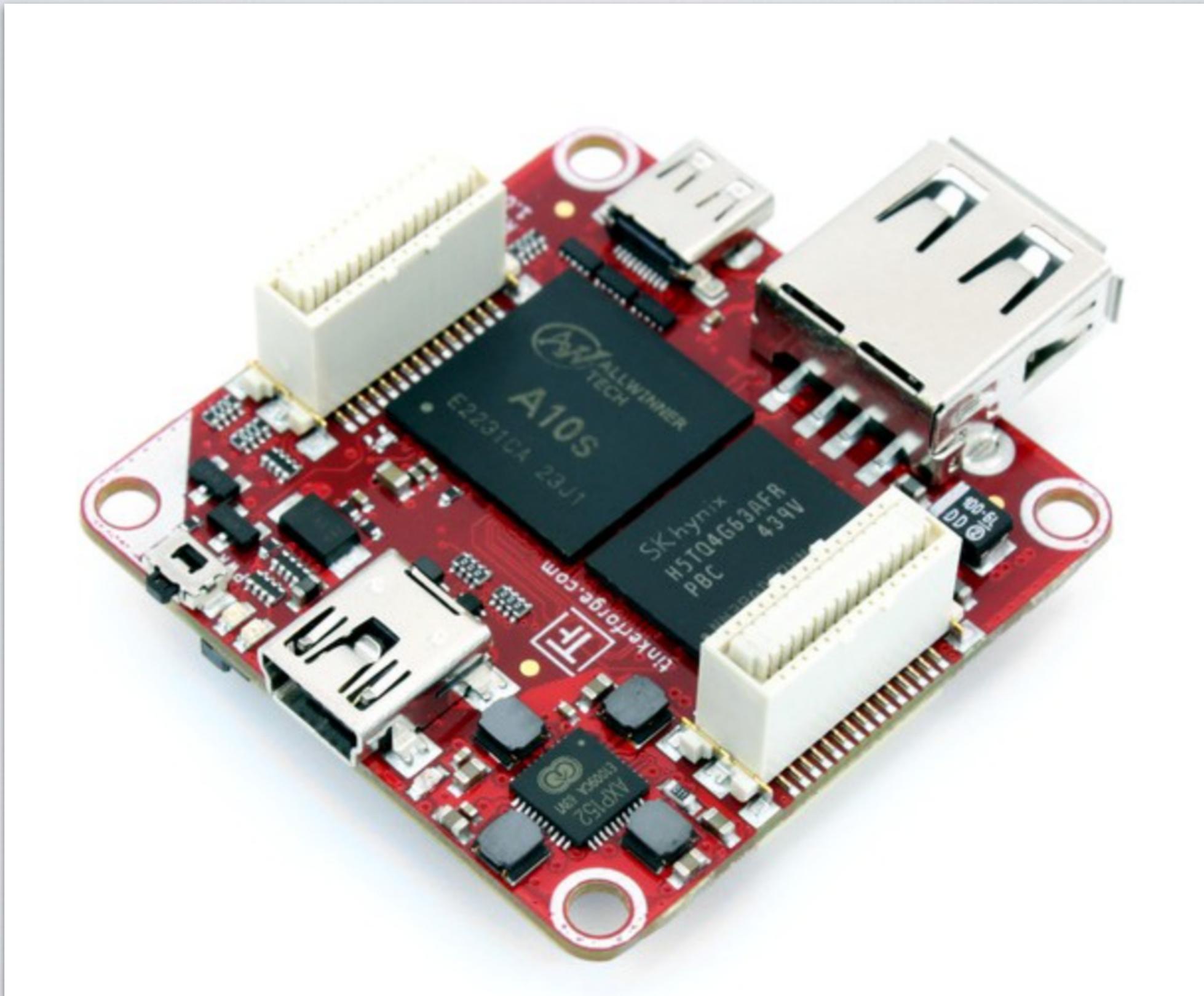
**Home** | **About** | **LaunchPads** | Software | BoosterPacks | BYOB | Projects | Community & Support

**3** Choose Your LaunchPad    **2** Download Software    **1** Plug in BoosterPack

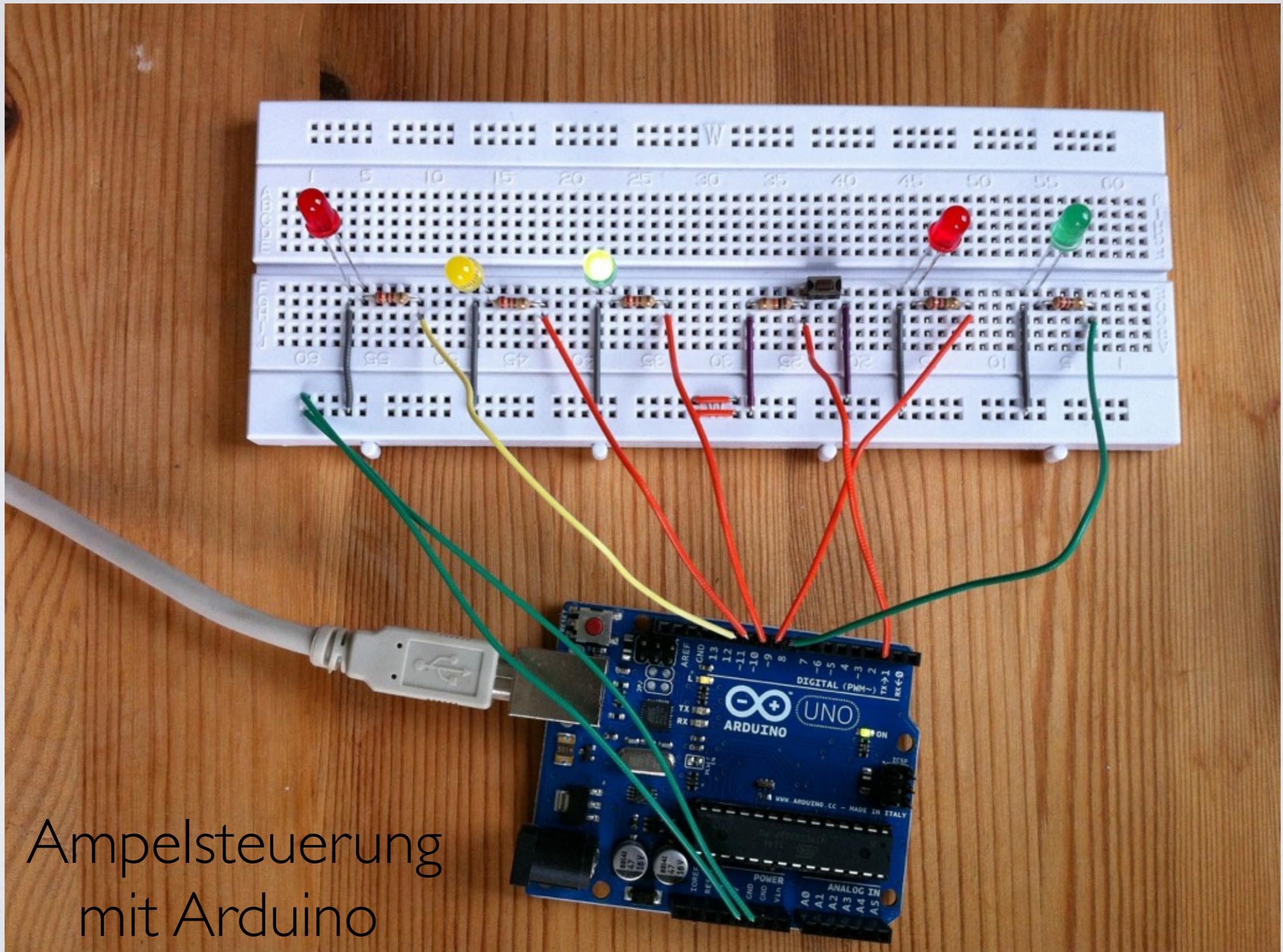
MSP LaunchPads 	C2000 LaunchPads 	Connected LaunchPads 	Hercules LaunchPads 	LaunchPad Brochure 
Ultra-Low Power	Real-Time Control	Get connected	Safety	Brochure
- Starting at \$9.99 <a href="#">Buy Now</a>	\$17.00 <a href="#">Buy Now</a>	\$12.99 <a href="#">Buy Now</a>	\$19.99 <a href="#">Buy Now</a>	
- Specifications				
16-bit ultra low power MCUs 32-bit ultra low power ARM® MCUs	32-bit real-time digital control MCUs	32-bit ARM® Cortex-M4	32-bit ARM® Cortex-R4 lock-step CPUs	
Up to 256 KB Flash	Up to 256 KB Flash	Up to 1 MB flash	Up to 384 KB Flash	
Up to 64 KB RAM	Up to 96 KB RAM	Up to 256 KB RAM	Up to 32 KB RAM	
Up to 48 MHz	Up to 90 MHz	Up to 120 MHz	Up to 100 MHz	
+ Key integration				
+ Applications				

[Quelle: <http://www.ti.com/ww/en/launchpad/launchpads.html>, Zugriff: 26.03.2014]

# Beispiele für Einsteigerplattformen: Tinkerforge RED Brick



# Prototyping Embedded Systems



ZUM PRAKTIKUM

# Praktikum

- An Anwendungsbeispielen u.a. mit den Plattformen
  - TI Tiva-C Launchpad
- Kennenlernen von
  - Hardware-Abstraction-Layers
  - Framework und Tool-Chain
  - einer RTOS-Umgebung (FreeRTOS)
- Programmierung in Assembler, C/C++
- Umsetzung wesentlicher Methoden, wie Interrupts, Powermanagement

# Warum diese Plattformen?

- Geringe Kosten (erschwinglich für fast jeden)
- Open Source / nicht proprietär
  - viel Dokumentation, viele Beispiele
  - viele Nutzer und Nutzerforen
- Einfache IDE mit “geringer Hemmschwelle” und weitergehende IDE mit Debugger etc.
- Entwicklungsumgebung unabhängig vom System (Linux, Windows, OS X)
- Grundlegendes demonstrierbar und Erlerntes transformierbar auf andere Plattformen
- Vielzahl an Möglichkeiten für weitergehende Ideen
  - über “Arduino/Energia”-kompatible und ARM-Cortex-basierte Plattformen
  - s. beispielsweise TI Innovation Challenge (TIIC)

# Arduino/Energia und Embedded Softwareentwicklung

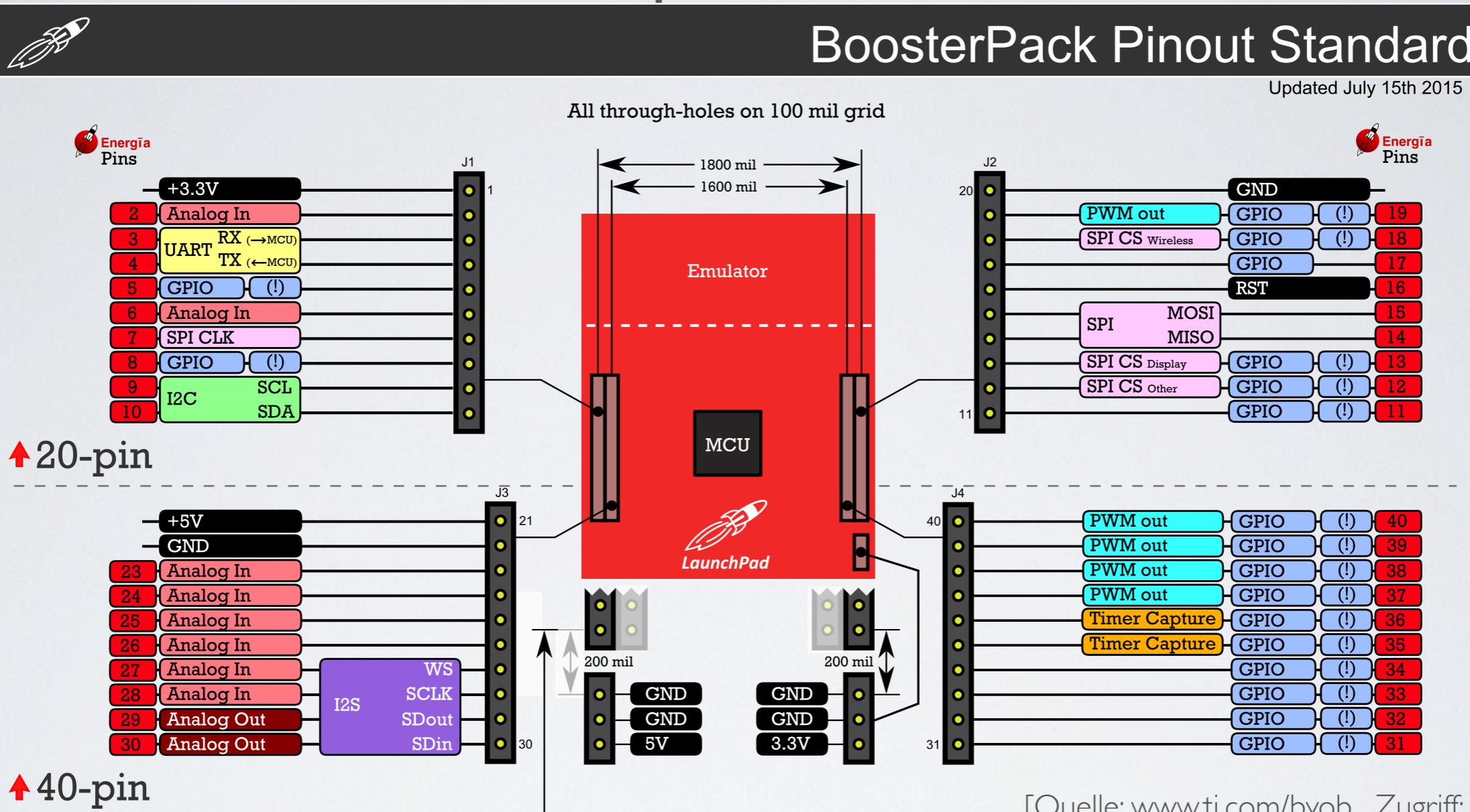
- Hohe Abstraktion von Hardware durch IDE und Bibliotheken
  - pro:
    - schnelle Ergebnisse
    - Kreativität im Fokus
  - contra:
    - Verständnis des Mikrocontrollers bei Arduino/Energia nicht notwendig, aber für professionelle Programmierung wichtig
    - zu wenig Wert auf technische Aspekte (z.B. Effizienz)

Daher: Tiefere Betrachtung des Mikrocontrollers im Praktikum

# Launchpad Tiva-C

## BoosterPack Pinout Standard

Updated July 15th 2015



- Tina TM4C123GH6P Mikrocontroller, ARM Cortex-M4F Prozessor, 80 MHz (100 DMIPS)
- 256KB interner Flash, 32 KB SRAM, 2 KB EEPROM
- 14 physische GPIO Blöcke (144 I/Os)

# Praktikumsaufgaben

- Umsetzung von Aufgaben / Konzepten in 2er-Teams  
(gemäß XP Pair-Programming)
- Feste Teams
- Augenmerk auf Qualität der Software-Lösung / Umsetzung
  - Funktion
  - Struktur (Lesbarkeit, Modularität, keine Magic Numbers...)
  - Kommentare

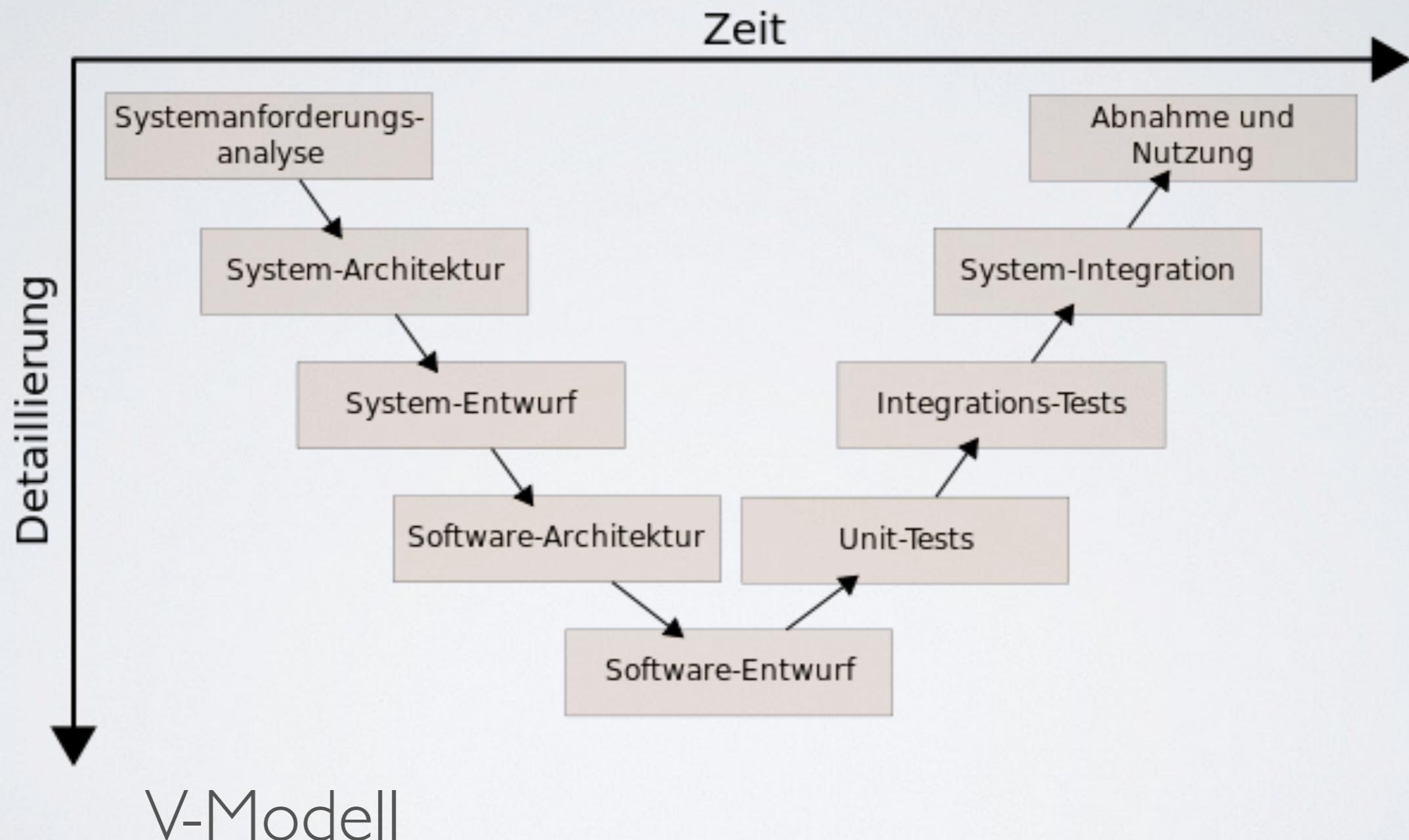
# Entwicklung Eingebetteter Systeme

verwendet Wissen aus den Gebieten

- Technische Informatik
- Programmiermethoden
- Software Engineering
- Betriebssysteme
- Systemprogrammierung
- Usability und Datenvisualisierung

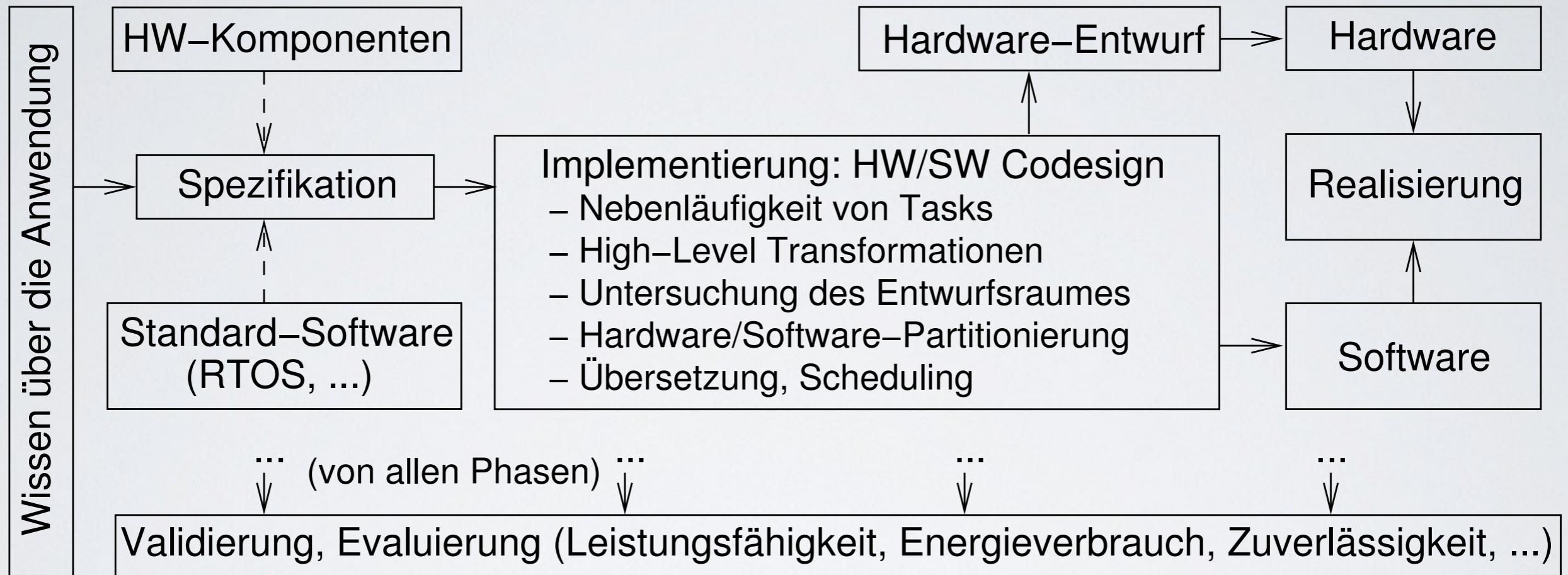
# Entwicklungsmodelle

- Häufig V-Modell, derzeit noch wenig andere (agile) Ansätze



[Quelle: Pätzold, Seyfert, <http://upload.wikimedia.org/wikipedia/de/3/3a/V-Modell.svg>, 2010, Zugriff: 26.03.2014]

# Hardware/Software Codedesign



Entwurf Eingebetteter Systeme (nach Marwedel)

[Quelle: Marwedel, Eingebettete Systeme, Springer-Verlag, 2008]

# Besonderheiten bei Entwicklung Eingebetteter Systeme

- Enges Zusammenspiel von Entwicklungsteams für
  - Hardware
  - Software  **Fokus**
- Spezielle Programmiersprachen und Anforderungen (standardisiert) an diese, z.B.
  - Ada
  - MISRA-C
- Normen und Standards für Entwicklung, z.B.
  - IEC 61508 ‘Funktionale Sicherheit sicherheitsbezogener elektrischer/elektronischer/programmierbarer elektronischer Systeme’, Teil 3: Anforderungen an Software

# Besonderheiten bei Entwicklung Eingebetteter Systeme

Entwicklung nicht auf dem Target-System, daher

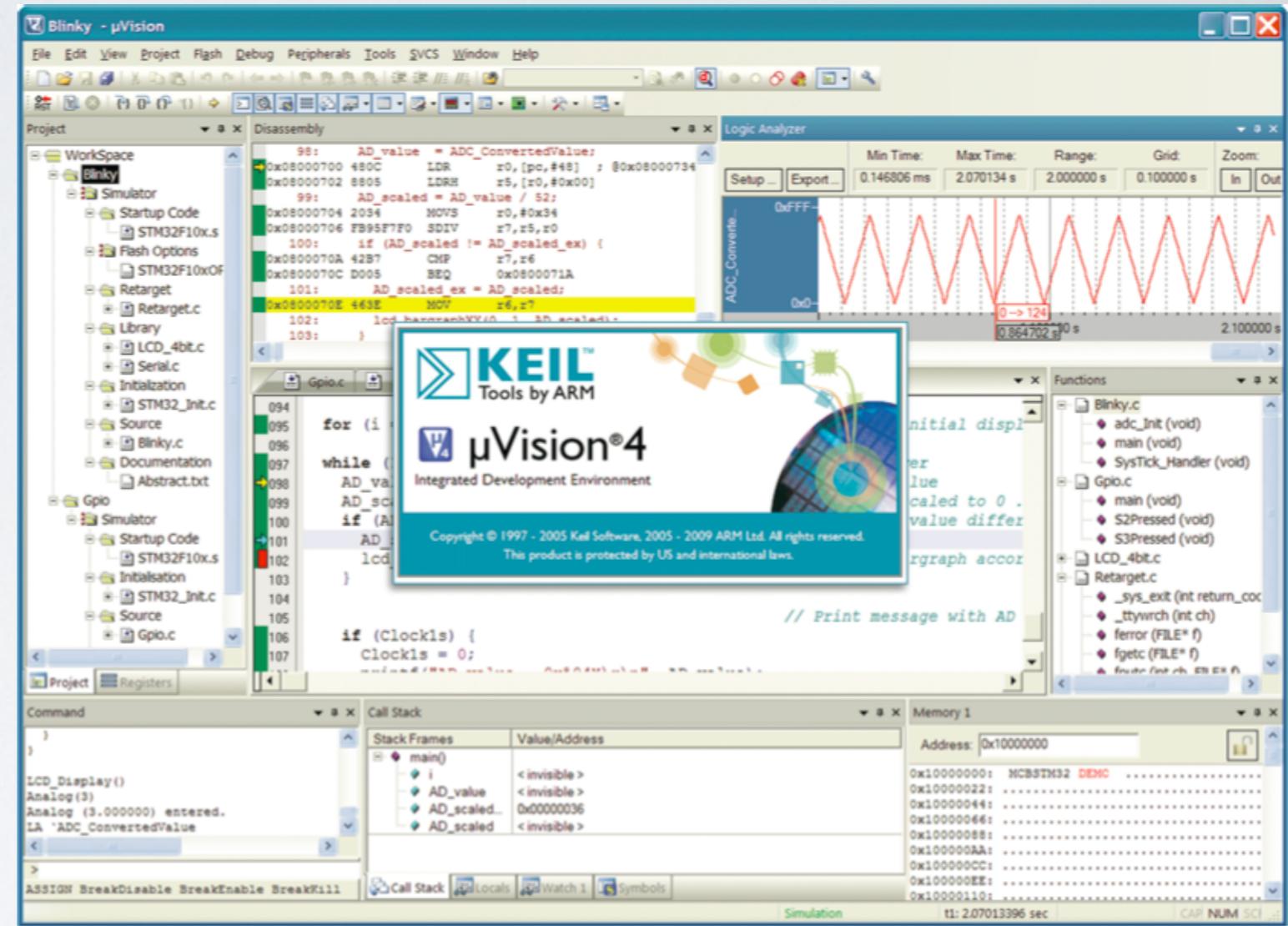
- Cross-Compiler
- Debugging erfordert Umwege, z.B.
  - mittels Simulatoren
  - über JTAG
  - Problem bei Hardwarebugs...

...This version of the board was mainly to provide a new SD connector due to supply issues. Also added was R219 in an attempt to make the Yellow LED on the RJ45 turn on when at 100M operation. This caused the Ethernet not to work on power up but did work after a board reboot....

[Quelle: BeagleBoard.org, Getting started with your new BeagleBone, [beagleboard.org/static/beaglebone/latest/README.htm](http://beagleboard.org/static/beaglebone/latest/README.htm), Zugriff 26.03.2014]

# Entwicklungswerkzeuge

- Vielfältig
  - Konsolen-Toolchains
  - IDEs verschiedener Abstraktionsebenen, u.a. Visual Programming
  - Libraries
  - RTOS
  - OS, z.B.  
Embedded Linux



Beispiel einer IDE

[Quelle: Keil.com, Getting Started Creating Applications with μVision®4, URL: <http://www.keil.com/product/brochures/uv4.pdf>, Zugriff: 26.03.2014]

# Trends bei Eingebetteten Systemen

- Modellbasierte Entwicklung
  - Entwurf von Modellen (hoher Abstraktionsgrad)
  - Automatische Codegenerierung anhand Modellen
  - Verifikation anhand Modellen
  - Plattformunabhängig und anpassbar für jeweilige Plattform
- Wiederverwendbare Komponenten, Frameworks, COTS

[Quelle: Braun et al, Study of Worldwide Trends and R&D Programmes in Embedded Systems in View of Maximising the Impact of a Technology Platform in the Area, 2005: URL: [http://www.artemis-austria.net/uploads/media/FAST\\_final-study-181105\\_en.pdf](http://www.artemis-austria.net/uploads/media/FAST_final-study-181105_en.pdf), Zugriff 26.03.2014]

# Wohin geht's mit Embedded Systems?

- Internet of Things / Cyber-physische Systeme / Industrie 4.0
  - Vernetzung vieler Geräte bzw. smarter Objekte
  - Mehr Berechnungen auf Gerät (IoT-Device oder Edge)
- Energieeinsparung
  - Intelligente Umgebungen
  - E-Ink
- Robotik
- Software wird entscheidendes Differenzierungsmerkmal
  - Schwerpunkt bei Softwareentwicklung für Eingebettete Systeme

# Embedded World Conference 2018

- IoT (OS & Platforms, Solutions, Connectivity...Payment): 9 Sessions
- Embedded Vision (and Augmented Reality): 4 Sessions
- Autonomous Systems (...Machine Learning): 5 Sessions
- Security and Safety (...Functional Safety...Hacking): 9 Sessions
- Embedded OS (...Linux...RTOS...): 8 Sessions
- Hardware Engineering (Multicore... FGPA...): 7 Sessions
- Software and Systems Engineering (...Modeling, Design Pattern, Quality, Standards, C++...): 13 Sessions

# Zusammenfassung

- Eingebettete Systeme treiben und werden von Technologieentwicklung getrieben.
- Softwareentwicklung für Eingebettete Systeme
  - nimmt an Bedeutung zu, und
  - hat besondere Anforderungen an Entwicklung.
- Plattformen-Vielfalt erfordert generelle Herangehensweisen.

ENDE