

# Fuzzy Control Teaching Models

**Klaus-Dietrich Kramer**

**Department of Automation and Computer Science  
Harz University, Wernigerode, Germany, and  
Institute of Automation and Informatics (IAI),  
Wernigerode, Germany**

[kkramer@hs-harz.de](mailto:kkramer@hs-harz.de); [k.kramer@iai-wr.de](mailto:k.kramer@iai-wr.de)

**Annedore Söchting**

**Institute of Automation and  
Informatics (IAI),  
Wernigerode,  
Germany**

[a.soechting@iai-wr.de](mailto:a.soechting@iai-wr.de)

**Thomas Stolze**

**Department of Automation and  
Computer Science  
Harz University, Wernigerode,  
Germany**

[tstolze@hs-harz.de](mailto:tstolze@hs-harz.de)

## Abstract

Many degree courses at technical universities include the subject of control systems engineering. As an addition to conventional approaches Fuzzy Control can be used to easily find control solutions for systems, even if they include nonlinearities. To support further educational training, models which represent a technical system to be controlled are required. These models have to represent the system in a transparent and easy cognizable manner. Furthermore, a programming tool is required that supports an easy Fuzzy Control development process, including the option to verify the results and tune the system behavior. In order to support the development process a graphical user interface is needed to display the fuzzy terms under real time conditions, especially with a debug system and trace functionality. The experiences with such a programming tool, the Fuzzy Control Design Tool (FHFCE Tool), and four fuzzy teaching models will be presented in this paper. The methodical and didactical objective in the utilization of these teaching models is to develop solution strategies using Computational Intelligence (CI) applications for Fuzzy Controllers in order to analyze different algorithms of inference or defuzzification and to verify and tune those systems efficiently.

**Keywords:** Fuzzy Control, FHFCE-Tool, Teaching Models, FC-Debugging, FC-Tracing

---

Material published as part of this publication, either on-line or in print, is copyrighted by the Informing Science Institute. Permission to make digital or paper copy of part or all of these works for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage AND that copies 1) bear this notice in full and 2) give the full citation on the first page. It is permissible to abstract these works so long as credit is given. To copy in all other cases or to republish or to post on a server or to redistribute to lists requires specific permission and payment of a fee. Contact [Publisher@InformingScience.org](mailto:Publisher@InformingScience.org) to request redistribution permission.

## Introduction

Over the last years several teaching systems have been developed at the Harz University for different applications. One particular group are teaching systems for Fuzzy Control Methods (FC Methods) in automation (Becker, 1996;

**Editor: Eli Cohen**

Submitted: December 1, 2015; Revised: March 16, 2016; Accepted: May 11, 2016

Blankenberg, 2004; GUNT Gerätebau GmbH, 2007). To program these models a graphical user interface (FHFCE Tool) was introduced. This tool allows to define all structures of the Fuzzy Controller, selecting different functions of processing, and finally generating the machine code for the target microcontroller which controls the model. For tuning the Fuzzy Controller, a Fuzzy Debug System with a trace option is available (Blankenberg, 2004; Söchting, Stolze, Kramer, & Braune, 2009).

The aim of using the Fuzzy Teaching Models is, on the one hand, the transfer of knowledge about Fuzzy Control (Mamdani Type) and the algorithms used (Lee, 1990; Driankov, Hellendoorn, & Reinfrank, M., 2013), and, on the other hand, to get to know techniques of optimization and tuning of Fuzzy Controllers in real technical applications. This is achieved through an implemented online debugging with a trace function for selected values.

The experiences with these systems shall be described below.

## Models and Software

### Overview

At the Harz University 4 Fuzzy Control teaching models with different skill levels were developed:

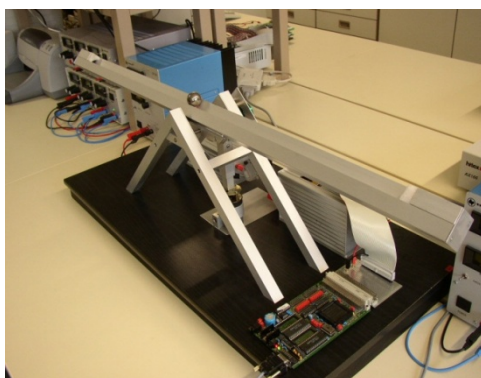
- Ball-on-Beam (1-dimensional seesaw model)
- Inverted Pendulum (propeller model)
- Ball-on-Plate (2-dimensional seesaw model)
- Carrier Vehicle with Inverted Pendulum (stick balance car model)

### Aspects of Using the FC Teaching Models

At all 4 models the technical problem is easily cognizable. There are 2 or 3 process inputs and 1 or 2 process outputs, so it is easy to define the rule base and to test the system. Differences between the models exist at the real time demands, the structures of the Fuzzy Controllers, and the different FC processes (different inference processes and different defuzzyfication functions).

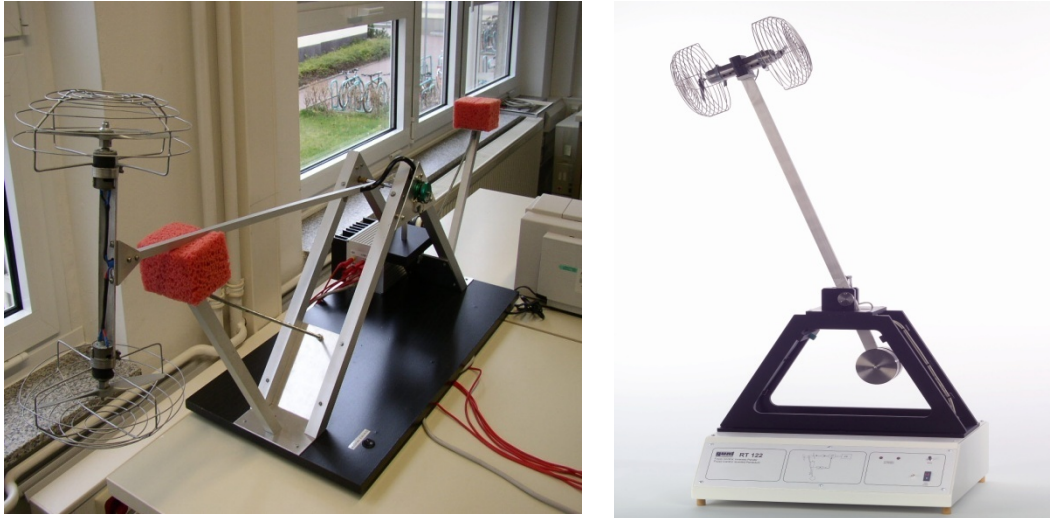
### Examples

The first model is a 1-dimensional seesaw, shown in Figure 1. Based on three process inputs, position of the ball, speed of the ball and angle of the seesaw, the ball shall be balanced at the middle of the seesaw. The special topic of this experiment is to learn about the essential FC functions, the handling of the FHFCE-Shell and about the online debugging and trace functions.



**Figure 1: Ball-on-Beam (left: prototype of Harz University; right: industrial reproduction)**  
(Kramer & Braune, 2001; GUNT Gerätebau GmbH, 2007)

A typical use case for Fuzzy Control is the inverse pendulum (Figure 2). Here the position (technically speaking, the angle of the pendulum) is manipulated with two opposite rotating air twist-ers. The process inputs are the position of the pendulum, the speed of the pendulum and, if necessary, the position of a disturbance load.



**Figure 2: Inverted Pendulum (left: prototype of Harz University; right: industrial reproduction)** (Kramer & Braune, 2001; GUNT Gerätebau GmbH, 2007)

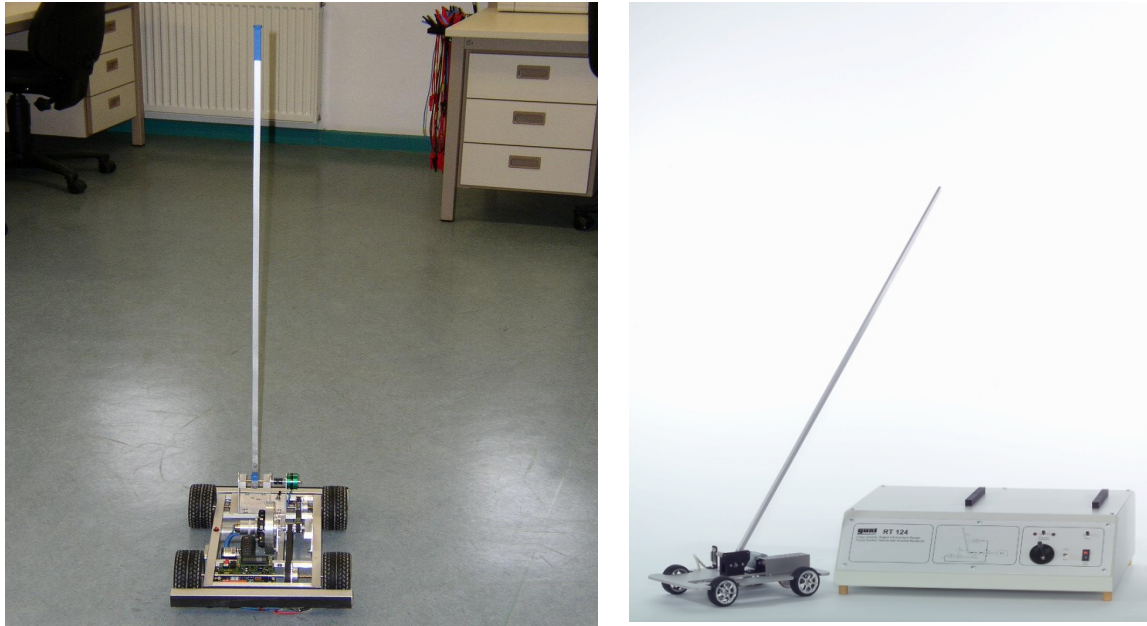
Two process outputs (air twister I and II) are controlled to stabilize the pendulum in a vertical position. The special topic of this model is the realization of two independent Fuzzy Controllers where each of them has an independent rule base and special demands on tuning and verification of the whole system.



**Figure 3: Ball-on-Plate (left: prototype of Harz University; right: industrial reproduction)** (Kramer & Braune, 2001; GUNT Gerätebau GmbH, 2007)

The third model is a 2-dimensional seesaw (Ball-on-Plate - Figure 3). The topic of this model is to realize two completely independent Fuzzy Controllers with separated inputs (ball position and ball speed in x and y direction and the table position) and outputs (motor control in x and y direction).

The main problem of this model is to balance the ball in the middle of the table with the two independent cooperating Fuzzy Controllers. A possibility to exercise the position of the ball or to train a complex motion is given in teach-in-mode.



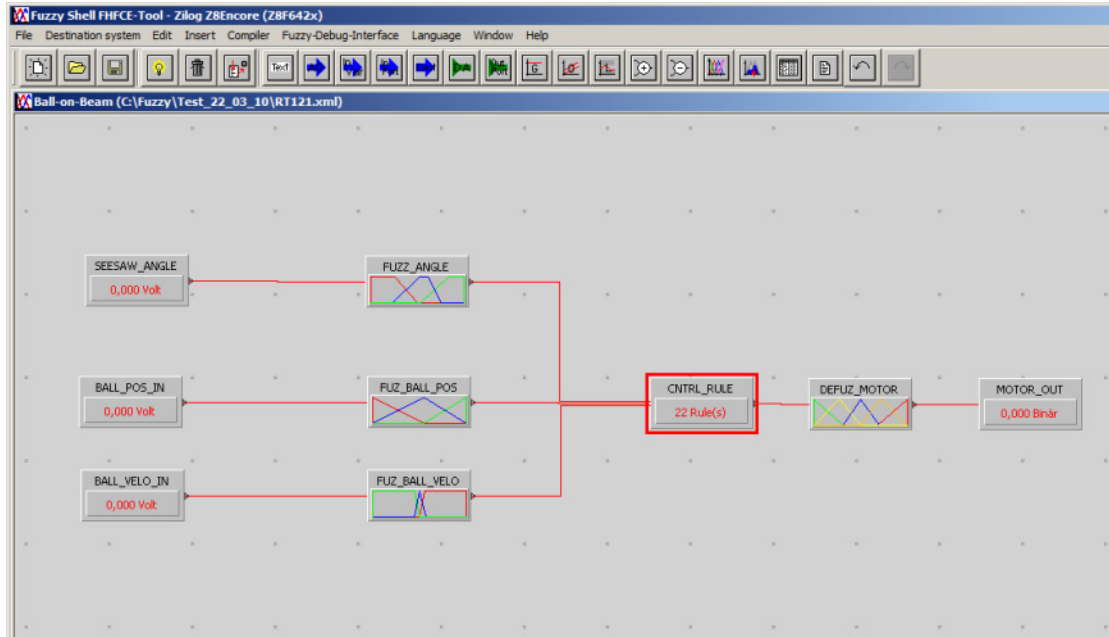
**Figure 4: Carrier Vehicle with Inverted Pendulum (left: prototype of Harz University; right: industrial reproduction)** (Kramer & Braune, 2001; GUNT Gerätebau GmbH, 2007)

The fourth model is the stick balance car model (Figure 4), a special version of the inverted pendulum. With two or three process inputs (angle of the stick, angle velocity of the stick, position of the car), the pendulum must be held in vertical position. The extremely high real time conditions require a special system design and the definition of special tuning strategies.

## FHFCE-Tool

### ***Project Editing of FCs***

The structure of the design process of a Fuzzy Control System is realized analogously to the Fuzzy Control Process, consisting of the sub-processes **FUZZYFICATION**, **INFERENCE** (with the **RULE BASE**) and **DEFUZZYFICATION**. All the definitions of the structure and the algorithms of the fuzzy process have to be determined, including all data conversions and data manipulations. To achieve this, all relevant functions and algorithms are deposited as icons in a drag and drop toolbar. All icons include software macros, written in assembler, to generate an optimized code and to upload the compiled machine code to the target microcontroller (MC). The structure of the program functions of the FC is defined by the red lines (Figure 5). In the code generation process the machine code of the MC is generated depending on this structure (Blankenberg, 2004).



**Figure 5:** Graphical Editing Window of FHFCE Tool

### ***FC-Debugging***

After the process of code generation and uploading the machine code to the target controller, a verification and tuning phase is necessary. This verification and tuning phase can be realized directly (online or offline) or indirectly (offline or with a simulator). The modification of parameters or rules has to be carried out in this phase. Additionally, a change of system functions (e.g., inference functions like MAX-MIN or MAX-PROD, or defuzzification algorithms like Singleton-, Center of Maxima- or Center of Area-Method), is possible, too.

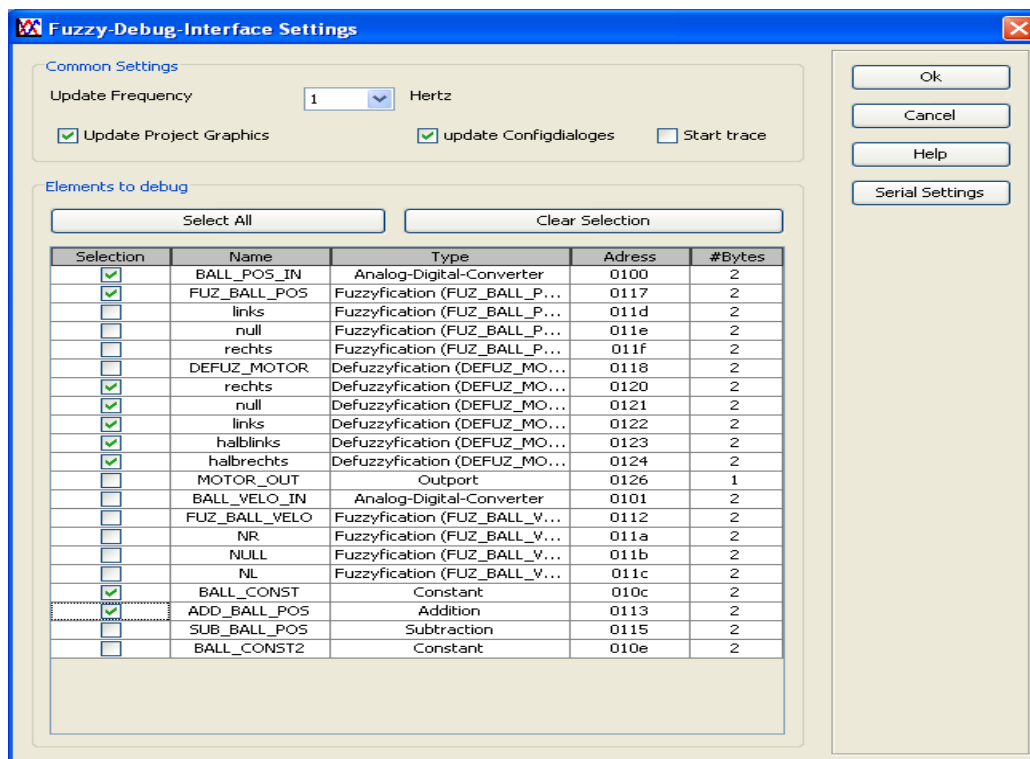
There are different goals in the verification and tuning phase:

- improvement of system dynamics
- improvement of system exactness
- general system optimization

All these processes which are described as tuning and optimizing processes require efficient system support:

- online debugging (realized by the system debugger)
- online debugging with management of the INPUT /OUTPUT data (e.g. realized by a real-time system)
- complex analysis of performance (measuring methods, monitoring, fuzzy benchmarks, trace of relevant data)





**Figure 6:** Value selection at the Debugging Window

To use the online debugging function included in the FHFCE Tool all data (INPUTs, Fuzzyfication terms, Inference results, Defuzzyfication values, etc.) that should be analyzed, can be assigned in a special pop up window. Data is transferred to the host computer through an USB (virtual RS232) interface. In the process of Fuzzy Control all selected data is updated with 1, 2, 5, 10, 15 or 20 cycles per second – depending on the number of selected trace data and the complexity of the computations.

### ***FC-Tracing***

To trace special data for a period of time, the values are logged and displayed (Figure 7). Therefore, values to be traced have to be selected manually. These values are a subset of the selected debug data. After this, the selected data is updated in each execution cycle (Söchting et al., 2009).

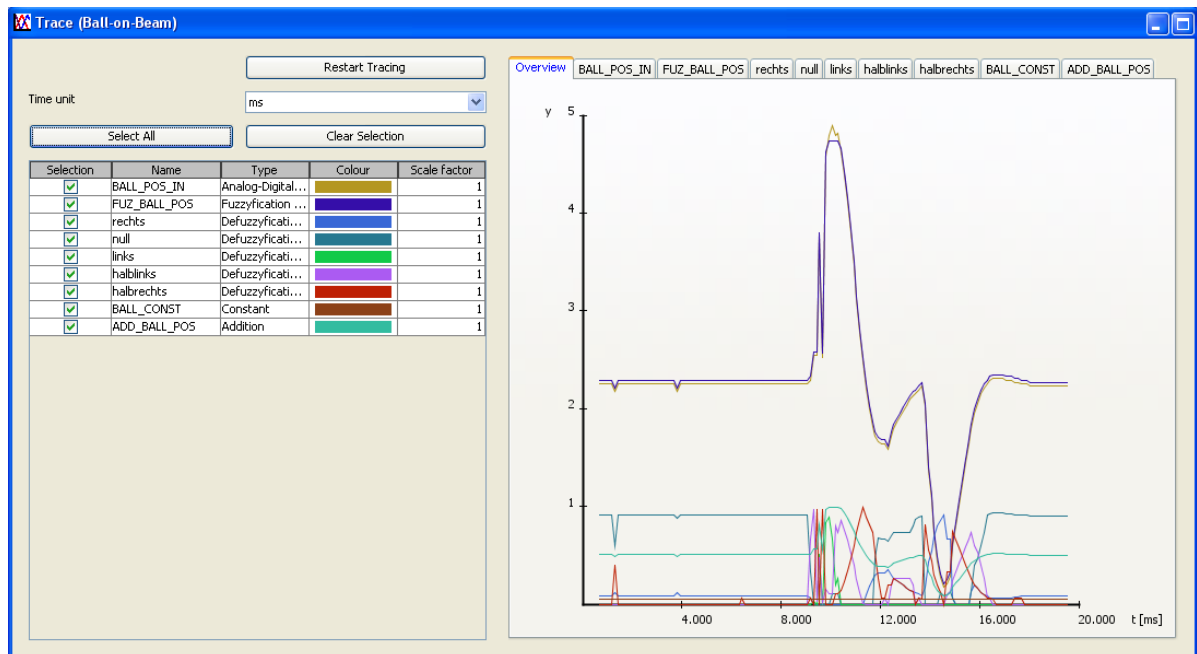


Figure 7: Trace Window

## Aspects of Experiences

### Compilation of Projects

At the end of the graphical project editing the machine code for the microcontroller calculating the fuzzy algorithms has to be generated. All functions (INPUT, FUZZYFICATION, INFERENCE, DEFUZZYFICATION, etc.) are deposited as assembler macros in a library. The interfaces between the functions are well defined registers of the microcontroller (RAM addresses).

All defined functions, embedded into the whole project, are compiled and linked by one step. So it is possible to generate memory and time optimized machine code.

### Special Experiences

The aims of the development of Fuzzy Teaching Models (get to know about the fuzzy design process with gradated requirements, test of different fuzzy computing algorithms (INFERENCE, DEFUZZYFICATION), get knowledge about tuning strategies) are well imparted through the FHFCE Tool and the Fuzzy Teaching Models. The results of using these tools show that students are able to understand the real technical problem quickly and they are also able to find solutions for it (fuzzyfication of inputs, define the rule base and find the defuzzyfication algorithm). Generally said, the fuzzy models are well accepted by the students, since they are able to let the models perform actions by setting up the Fuzzy Controller. Furthermore, they can easily improve the behavior of the models using the online debugging functions. In fact, there is often a quite interesting competition between different student groups concentrating on building the most effective and precise Fuzzy Controller. That keeps students focused on learning about Fuzzy Control and optimizing their control strategies. The ability of remembering the knowledge learned while doing exercises with the fuzzy models is very high. Learning only fuzzy theories without practical training is not that effective.

## System Tuning

The development process consists not only of the system design and the programming of the FC but also in a particular way of adjustment and optimization of FC. This process, called system tuning, requires special tools like a debug system that is able to show the current relevant data and enables the developer to supervise the process outputs for given inputs or to display which rules are active at the moment.

In further development of the FHFCE Tools another function was created: the trace function. This allows logging of selected data for a certain period of time. This period depends on the selected data transfer rate (1 – 20 Hz, see section FC-Debugging). So it is possible to log short term or long term courses of the function's special process data (see Figure 7).

These tools allow optimal system tuning and education of students in this field.

## Conclusion

The Fuzzy Teaching Models presented in this paper give an overview of the huge variety of using FC applications. The models developed together with the FHFCE Tool represent graduated requirements for the user (student). Important conditions for a successful education are on the one hand the easy design process and on the other hand effective tools for system tuning. The experiences of the recent years show that FC Teaching Models are very useful in efficient education processes. This is achieved through the graphical design process of the FC which lets the user focus on the parameters of the fuzzy process instead of implementing the FC in source code (and need to know how to write efficient source code for the underlying microcontroller). For further developments the system can be extended to Neuro-Fuzzy methods to optimize the FC (optimization of fuzzification and/or of rule base) and test it by the FHFCE debug system.

## References

- Becker, C. (1996). *Entwicklung einer Programmieroberfläche für PMS500IF*. [Development of a programming interface for PMS500IF]. Diploma Thesis. Harz University.
- Blankenberg, C. (2004). *Entwicklung einer Fuzzy-Shell für den Mikrocontroller Z8ENCORE*. [Development of a fuzzy shell for the Z8ENCORE microcontroller]. Diploma Thesis: Harz University
- Driankov, D., Hellendoorn, H., & Reinfrank, M. (2013). *An introduction to fuzzy control* (2nd ed.). Berlin: Springer-Verlag.
- GUNT Gerätebau GmbH. (2007). *RT121 – RT124 teaching systems for fuzzy methods in automation*. IM-PRINT, GUNT Gerätebau GmbH, Hamburg. Retrieved from [http://www.unhas.ac.id/rhiza/arsip/kuliah/Sistem-dan-Tekn-Kendali-Proses/GUNT%20Catalog/introduction%20to%20equipment%20series/RT12x\\_english.pdf](http://www.unhas.ac.id/rhiza/arsip/kuliah/Sistem-dan-Tekn-Kendali-Proses/GUNT%20Catalog/introduction%20to%20equipment%20series/RT12x_english.pdf)
- Kramer, K.-D., & Braune, S. (2001). Fuzzy design tool for low-cost microcontrollers. *Proceedings of ISIC-2001*, Singapore, NTU, p.473-475.
- Lee, C.C. (1990). Fuzzy logic in control systems: Fuzzy logic controller – Part I. *IEEE Transactions on Systems, Man, and Cybernetics*, 20(2).
- Söchting, A., Stolze, T., Kramer, K.-D., & Braune, S. (2009). *Trace function at the FHFCE-Tool*, Harz University. internal paper.



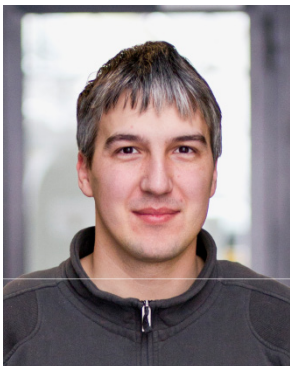
## Biographies



**Prof. Dr.-Ing. Klaus-Dietrich Kramer** is an Application Engineer in an engineering institute, Lecture at the Ingenieurschule Eisleben, since 1998 Professor for Microprocessorsystems at the Harz University, Wernigerode, since 2004 President of the Institut of Automation and Informatics (IAI) in Wernigerode. His research interests are Microcontroller Applications, Benchmarks of Microcontrollers, Microprocessors and Digital Signal Processors, CI-Applications (Fuzzy-Control an Low-Cost-MC, Real Time CI-Systems, etc.), Automotive Applications.



**Annedore Söchting** is a software engineer at the Institut für Automatisierung und Informatik GmbH in Germany. She received her Master of Computer Science, mobile Systems, from the Fachhochschule Harz, Wernigerode (University of Applied Studies and Research), Germany



**Thomas Stolze** was a software engineer at the Institut für Automatisierung und Informatik GmbH from 2005 to 2008 and is now a research associate in the Microcontroller Application Center (MCAC) at Harz University, Germany. He is currently a doctoral Candidate at the Technical University of Ilmenau in Germany.