

Karsten Becker

KB@Karsten-Becker.de
http://Karsten-Becker.de

EDUCATION

TU Hamburg Harburg (TUHH), Hamburg, Germany	— PhD candidate 2010-2015
KTH Stockholm, Stockholm, Sweden	— Master of science 2009
HAW Hamburg (University of applied science), Hamburg, Germany	— Bachelor of science 2007
TU Hamburg Harburg (TUHH), Hamburg, Germany	— aborted in 2003

WORK EXPERIENCE

Senior Embedded Software Developer / Team architect — 2020 - present
Working on Siprotec Protection Software at Siemens AG in Siemensstadt

CTO / Head of electronics — 2010 - 2019
Leading the electronics team and represent the team in public. See the section "Part-Time Scientists".

PhD Candidate at Technical University Hamburg — 2010 - 2015
As PhD candidate at the institute of computer technology, I am researching on hardware description languages and FPGA architectures. I also work as teaching assistant for (hands-on) lectures. Please refer to section "The Plain Simple Hardware Description Language" below for more information.

Internship at IBM Research; Zurich, Switzerland — 5 months in 2007
I worked on the Eclipse JDT team and implemented two new refactorings, some quickfixes and added JUnit test cases for each. I also updated the documentation for the new release.

Java tutor HAW; Hamburg, Germany — Summer semester in 2005, 2006
For our Java lecture, I held tutorials to help students understand the course contents.

Internship at Siemens research; Munich, Germany — 6 months in 2005 through 2006
I helped to develop several prototypes on the Eclipse RCP environment, including an authentication and authorization framework, an MP3 manager and a database tool. I also helped with a workshop for the OOP conference.

VHDL lab assistant TUHH; Hamburg, Germany — Summer semester 2003-2005, Winter semester 2011-2014
For the VHDL courses, I worked as assistant for a mandatory lab.

Development of WAMP portal for TUHH; Hamburg Germany — 3 months in 2002 through 2003
For a department at TUHH, I developed a web portal based on WAMP (Windows, Apache, MySQL, PHP).

PROFILE

I began programming with Turbo Pascal at 16 and soon transitioned to Delphi, where I developed numerous small applications. In high school, I learned C and completed a project for my graduation called "robot_control." This project involved a Delphi application connected to an Atmel controller via a serial port. I programmed the controller in C and designed, manufactured, and soldered a double-sided PCB myself. I also completed the coursework for CCNA and CCDA certifications, scoring over 95% in the tests, though I did not obtain the certifications.

After high school, I worked as an administrator, leveraging my strong Linux and networking skills. At TUHH, I learned VHDL and created a 3D accelerator. After transferring to HAW Hamburg, I started programming in Java and using Eclipse as an IDE. During my studies, I developed two Eclipse features: the PL_Editor, which optimized the copy/paste/replace method, and 3D Democreator, a collaborative project for creating 3D demos using a visual representation of the rendering tree. My bachelor's thesis focused on Aspect-Oriented Programming and compared weaving technologies in Java-based AOP languages.

For my master's degree at KTH Stockholm, I worked on the pricerat.com concept, which participated in the Android Developer Challenge, and developed SequenceIDE, a description language for sequences that translates to bytecode and generates an interpreter. My master's thesis aimed to enhance the public transit system by designing an architecture that improved user convenience and information accessibility.

Part-Time Scientists

At Part-Time Scientists (now PTScientists), I was responsible for the technical development of the ALINA spacecraft. This included system architecture development, PCB design coordination, review, manufacturing, and low-level software development. As a member of the steering board and CTO, I was involved in

strategic decisions and partnership setups. Additionally, I served as a public representative for the company at conferences and in media interviews.

The Plain Simple Hardware Description Language

When I began my PhD at TUHH, my initial focus was on FPGA architectures. I soon realized that developing a useful architecture required better tooling for creating benchmarks. This led me to develop a user-friendly language and a web-based IDE in Dart with Polymer. Recognizing a market opportunity, I continued to develop and market the IDE to makers and hackers, aiming to extend it to an online course for self-learning hardware development. The primary features of PSHDL are user-friendliness, extremely fast simulation speeds, and openness. The web IDE also provides feedback for improving the language and tools (cf. paper: doi: 10.1109/EDUCON.2014.6826073).

CONFERENCES AND PROJECTS

On several occasions I gave presentations at the CCC camp and congress. I am also author and co-author of multiple papers presented at IAF events. Feel free to contact me for a list.

A selection of Fun/study projects from <http://karsten-becker.de/projects> :

SmartHome	I deployed several ESP32 around my house to monitor humidity, pressure, temperature, air quality, light and motion. I reverse engineered some protocols to automate my TV and my heating unit and started to develop a binding for those for OpenHab.
Game of Waves	A gamification of my online PSHDL tool where you have to write efficient code to recreate the output trace. Try it: http://waves.pshdl.org
The welcome Stockholm experience	A system architecture to provide a personalized experience for the users of a public transit system. The personalization is of special interest for foreign and disabled people. It also allows new business opportunities.
SequencelIDE	A domain-specific language for describing sequences that can be translated to byte code. It also generates the interpreter needed for the byte code. The interpreter is designed to run on an embedded platform and could be updated during runtime with new byte code. This allowed very quick turn-around times to find the best timing possible.
Pricerat.com	A location-based pricing service contributed to the Android Developer Challenge.
PL_Editor	A domain-specific language for generating C code of finite state machines for automation processes written as Eclipse feature. PL_Editor was used by several students and allowed them to focus on the state machine rather than the copy-pasted error-prone C code.

Thesis's:

Bachelor: <http://karsten-becker.de/BachelorThesis.pdf>
Master: <http://karsten-becker.de/master.pdf>

Overview of technologies used

Below I collected a list of technologies that I used to work with. This list is likely not complete and may miss several smaller projects and other things I did. But I hope that it provides a good overview over my expertise in the embedded domain.

Processors/FPGAs

Used for an extended period of time

- PPC440 (Virtex5) - Porting and development of drivers in QNX for rover
- ARM Cortex M3 (SmartFusion2) - Used for video processing and rover control
- ATMega64M1 - Used as a motorcontroller in rover
- ARM-Cortex-A53 (RaspberryPi 3b+) - Used for my SmartHome
- ESP32 - Also used for my SmartHome
- MicroSemi ProASIC3 - PSHDL Board and rover
- Spartan6 - Used for video processing

Used at some point

- Altera NIOS 2 - Just experimenting
- PPC405 (Virtex4) - Controlling the rover (baremetal)
- Microblaze - (Used in some university projects)
- ATXMega - Used in programmer for PSHDL, for simple rovers
- STM32F4 (Discovery board)- Used in university projects
- ARM Cortex-A8 (BeagleBone XM) - Ported QNX and used in simple rover; also with students
- MIPS (softcore) - Used for 3D Accelerator (customized ISA output of GCC for speedup)

Operating Systems

Used for an extended period of time

- MacOS - I use it every day
- Linux - used extensively in server environment with k8s
- QNX - Adjusted BSP to our requirements, developed custom drivers
- Android - Developed apps for controlling the rover

Used at some point

- FreeRTOS - Used it on SmartFusion for testing
- IOS - developed some non serious apps

Programming languages

Frequently used

- Rust
- C++

Occasional use

- Python
- Bash / Fish
- VHDL
- SQL
- C
- Java
- XTend
- PSHDL
- JavaScript

Used at some point

- Dart
- Verilog
- Go
- SystemC

IDEs

Frequently used

- VSCode
- PlatformIO

Occasional use

- XCode
- Eclipse CDT (SoftConsole)
- Android Studio
- Eclipse (Java)
- JetBrains IDEs