

Freelancer Viktor Hildebrand

Persönliche Daten

Name: Viktor Hildebrand
Adresse: Struthweg 16, 57578 Elkenroth
Geburtsdatum: 13.09.1984
Tel: +49 171 / 6015233
E-Mail: viktor@hildebrand.software
Website: <https://hildebrand.software>

Fachliche Kenntnisse

- **Programmiersprachen:** C MISRA-C:2012 C++ Python Java C#
- **Microcontroller:** Microchip Renesas NXP STM TI ARM-basierte-Controller
- **Bussysteme:** CAN LIN USB Ethernet UART SPI I²C 1-Wire
- **Plattformen:** Embedded-Systeme Maschinensteuerung Windows Linux
- **Test:** SonarQube VectorCAST PC-Lint CppCheck Python-Unittest GoogleTest
- **Architektur:** Rhapsody Cameo
- **Versionskontrolle:** Git SVN
- **Verwaltung:** Jira Polarion
- **CI:** GitLab CMake Conda
- **Dokumentation:** Doxygen MkDocs Latex
- **Office:** MS Office LibreOffice

Sprachen

- Deutsch
- Verhandlungssicheres Englisch

Zulassungen

- Ü2-Sicherheitsüberprüfung

Ausbildung

- **2001-08 - 2003-07:** Berufsbildende Schule Betzdorf/Kirchen
Höhere Berufsfachschule für Informatik
Schwerpunkt Technische Informatik und Automatisierungstechnik
- **2004-09 - 2010-01:** Universität Siegen
Studiengang: Angewandte Informatik - Anwendungsfach Elektrotechnik
Diplomarbeit: Bulk-Transfer und Firmware-Update über PLC-Netze

Gesamtnote Hauptstudium (D II): 1,5

Werdegang

- **2006-01 – 2010-01:** ROTH + WEBER GmbH in Niederdreisbach als Werkstudent
- **2009-09 – 2009-11:** Renesas Technology Europe GmbH als Diplomant
- **2010-02 – 2016-07:** ROTH + WEBER GmbH in Niederdreisbach als Softwareentwickler
- **2016-08 – 2019-12:** Thomas Magnete GmbH in Herdorf als Softwareentwickler
- **2020-01 – 2020-10:** Rohde & Schwarz SIT GmbH in Stuttgart als Freiberuflicher Softwareentwickler
- **2020-10 – 2022-05:** ROTH + WEBER GmbH in Niederdreisbach als Freiberuflicher Softwareentwickler
- **2021-03 – 2025-12:** Miele & Cie. KG in Gütersloh als Freiberuflicher Softwareentwickler
- **2022-12 – 2023-10:** Thomas Magnete GmbH in Herdorf als Freiberuflicher Softwareentwickler

Weiterbildung

- **2007-09:** Xilinx Professional VHDL
- **2016-09:** Functional Safety ISO26262 – System, Hardware & Software
- **2017-08:** Polarion Basic Training
- **2018-03:** VectorCAST C/C++ QuickStart Training
- **2018-08:** Willert UML Start-Up Training
- **2019-02:** Vector UDS Diagnose & Candela Studio
- **2019-09:** Jira Basic Training
- **2019-09:** ISTQB Certified Tester Foundation Level
- **2019-11:** Willert UML Follow-Up Training

Projekte

2024-06 – 2025-05 | Miele

Entwicklung einer Embedded-Software für eine Spülmaschine mit Cloud-Anbindung & App-Bedienung.

Aufgaben:

- Implementierung neuer Software-Komponenten in C++ für die Spülmaschinen-Steuerung
- Integration eines Kamera-Treibers für die Bildaufnahme des Spülmaschinen-Innenraums und Upload der aufgenommenen Bilder in die Cloud
- Implementierung von Unit-Tests mit GoogleTest
- Implementierung von Integrationstests mit PyTest
- Überwachung der Softwarequalität mit SonarQube

Hardware:

- MCU: STM32U575
- OS: Embedded Linux auf Arm Cortex-M4

Tools:

- IDE: Visual Studio
 - Projektverwaltung: GitLab
 - Statische Code-Analyse: SonarQube
 - Test-Framework: GoogleTest & PyTest
-

2022-12 - 2023-10 | Thomas Magnete

Entwicklung einer Embedded-Software für einen Ventil-Prototypen mit LIN-Interface.

Aufgaben:

- Erweiterung der Software-Architektur in Rhapsody
- Implementierung der Ventilsteuerung als State-Machine in Rhapsody
- Implementierung eines LIN-Stacks für die Kommunikation mit PLIN-LDF-Testumgebung
- Integration & Inbetriebnahme des Built-in-Bootstrap-Loaders

Hardware:

- MCU: Infineon TLE9867

Tools:

- IDE: Keil µVision5
 - Projektverwaltung: GitLab
 - Architecture: Rhapsody Architect
 - CAN-Adapter: PEAK PLIN-USB
-

2022-09 - 2024-06 | Miele

Entwicklung einer Embedded-Software für eine Bluetooth-Fernbedienung zur Steuerung eines Staubsaugers.

Aufgaben:

- Erstellung der Software-Architektur in Cameo
- Implementierung neuer Software-Komponenten in C für die Grundfunktionen der Fernbedienung
- Implementierung neuer Software-Komponenten in C für Bluetooth-Datenübertragung
- Anbindung an ein externes Bluetooth-Kommunikationsmodul und Test der Datenübertragung
- Überwachung der Softwarequalität mit SonarQube

Hardware:

- MCU: TI CC2651R3SIPA

Tools:

- IDE: Code Composer Studio
- Projektverwaltung: GitLab
- Architecture: Cameo Systems Modeler

- Statische Code-Analyse: SonarQube
-

2021-03 - 2024-06 | Miele

Entwicklung einer Embedded-Software für einen Staubsauger mit Cloud-Anbindung & App-Bedienung.

Aufgaben:

- Erstellung der Software-Architektur in Cameo
- Integration von Plattform-Software-Komponenten, u.a. Runtime-Environment mit EmbOS-Betriebssystem
- Integration & Test eines externen Kommunikationsmoduls mit WLAN & Bluetooth-Funktionalität
- Implementierung neuer Software-Komponenten in C++ für die Staubsauger-Steuerung
- Implementierung neuer Software-Komponenten in C++ für die Cloud-Anbindung
- Überwachung der Softwarequalität mit SonarQube

Hardware:

- MCU: STM32L496 / STM32U575

Tools:

- IDE: Visual Studio
 - Projektverwaltung: GitLab
 - Architecture: Cameo Systems Modeler
 - Statische Code-Analyse: SonarQube
-

2020-10 - 2022-05 | ROTH + WEBER

Entwicklung einer Embedded-Software für ein Farbdruckersystem und Einführung einer Unit-Testumgebung für die Windows-Entwicklungsumgebung und CI-Build-Umgebung.

Aufgaben:

- Implementierung neuer Software-Komponenten für das Farbdruckersystem in C
- Implementierung einer Unit-Testumgebung basierend auf CMake & GoogleTest
- Integration der Tests in die GitLab-CI-Pipeline

Hardware:

- MCU: NXP LPC178x

Tools:

- IDE: LPCXpresso IDE
 - Projektverwaltung: GitLab
 - Test-Framework: GoogleTest
-

2020-01 - 2020-10 | Rohde & Schwarz SIT

Test einer Embedded-Kryptomodul-Software eines Funkgeräts auf Integrationsebene und Weiterentwicklung der vorhandenen Testumgebung.

Aufgaben:

- Spezifikation von Test-Cases auf Integrationsebene in Python und XML
- Implementierung eines Test-Adapters auf Basis von Apache Thrift in C++
- Portierung & Integration einer vorhandenen C-Bibliothek in die Python-Testumgebung
- Generierung von Testreports in LaTeX
- Integration der Tests in die GitLab-CI-Pipeline

Hardware:

- MCU: NXP i.MX8 ARM
- OS: Embedded Linux

Tools:

- IDE: Eclipse
 - Projektverwaltung: GitLab & Jira
 - Test-Framework: Python-Unititest
 - Test-Report: Latex
-

2018-07 - 2019-12 | Thomas Magnete

Entwicklung einer Embedded-Software mit CAN-Interface zur Ansteuerung eines Schrittmotors.

Aufgaben:

- Betrachtung der funktionalen Sicherheit gemäß ISO 26262
- Definition von Anforderungen und Testspezifikationen
- Adaptierung vorhandener MCAL-Module
- Implementierung einer Schrittmotoransteuerung in MISRA-C:2012 mit Stromregelung, Positionsregelung und Geschwindigkeitssteuerung
- Implementierung von Sicherheitsfunktionen
- Betreuung eines externen Dienstleisters bei der Entwicklung eines 2-stufigen UDS-Bootloaders

Hardware:

- MCU: Microchip dsPIC33CH128MP505
- Driver: TI DRV8703

Tools:

- IDE: MPLAB X + PICkit 3
- Requirements: Polarion ALM
- Unit-Tests: VectorCAST
- Configuration: EB tresos
- Architecture: Rhapsody Architect

- UDS-Flasher: Vector vFlash
 - CAN-Adapter: PEAK PCAN-USB
 - CAN-Adapter: Vector VN1610
-

2017-02 - 2017-04 | Thomas Magnete

Entwicklung einer Embedded-Software für ein UART-Multiplexer-System.

Aufgaben:

- Erstellung der Software-Architektur
- Implementierung eines UART-Moduls für die Kommunikation mit einem SPS-Steuergerät
- Implementierung eines SPI-Moduls für den Zugriff auf 5 angeschlossenen MAX14830-ICs

Hardware:

- MCU: Atmel ATxmega32E5
- Peripherals: Maxim MAX14830

Tools:

- IDE: Atmel Studio + Atmel-ICE
-

2016-08 - 2018-06 | Thomas Magnete

Entwicklung einer universellen Build-Toolchain für Embedded-Software-Projekte in Python.

Funktionsumfang der Toolchain:

- Analyse der Formatierung und verwendeten Bezeichner hinsichtlich Naming-Conventions
- Statische Code-Analyse und MISRA-C:2012
- Automatisierte Ausführung von Unit-Tests
- Generierung der Projektdokumentation
- Kompilierung der Embedded-Software
- Ausführbarkeit auf einem Build-Server & am Entwickler-PC

Tools:

- Formatierung: LLVM Clang
 - Statische Code-Analyse: PC-Lint & CppCheck
 - Unit-Tests: VectorCAST
 - Projektdokumentation: Doxygen
 - Standard-Compiler: MinGW
 - Build-Umgebung: Jenkins
-

2016-08 - 2018-06 | Thomas Magnete

Entwicklung einer Embedded-Software in MISRA-C:2012 mit LIN-Interface zur Ansteuerung eines BLDC-Motors. Zusätzliche Entwicklung für den Systemtest: Eine Testumgebung in Python für das Embedded System mit zusätzlicher UART-Kommunikation für Event-Logging und u.a. Fault-Injection-Tests.

Aufgaben:

- Betrachtung der funktionalen Sicherheit nach ISO26262 auf System und Software-Ebene
- Definition von Requirements und Testspezifikationen
- Erstellung der Software-Architektur
- Implementierung einer BLDC-Motoransteuerung mit Stromregelung und Geschwindigkeitssteuerung
- Implementierung einer Single-Shunt-Current-Reconstruction für die Stromregelung.
- Implementierung eines SPI-Moduls für den Zugriff ein externes EEPROM und den Motortreiber
- Implementierung eines LIN-Stacks für die Kommunikation mit der Python Testumgebung
- Implementierung von Sicherheitsfunktionen

Hardware:

- MCU: Microchip dsPIC33EV128GM004
- Driver: TI DRV8305

Tools:

- IDE: MPLAB X + PICkit 3
- Unit-Tests: VectorCAST
- LIN-Adapter: PEAK PCAN-USB Pro FD

2015 | ROTH + WEBER

Entwicklung einer Embedded-Software für ein Falmmaschinensystem.

Aufgaben:

- Erstellung der Software-Architektur
- Implementierung eines USB-Kommunikations-Moduls
- Integration vorhandener Software-Module für CAN, I²C, UART & 1Wire Kommunikation
- Erweiterung eines vorhandenen Bootloaders um die USB-Kommunikation
- Integration der Schrittmotoransteuerung
- Implementierung der Faltablaufsteuerung

Hardware:

- MCU: Renesas RX600

Tools:

- IDE: Renesas e²Studio + J-Link

2014 | ROTH + WEBER

Entwicklung einer Falmmaschinen-Simulationssoftware mit Benutzeroberfläche in Java.

Aufgaben:

- Implementierung eines Lagenrechners für Längs- und Querverfaltung mit Visualisierung
 - Implementierung eines Rechners der die Dauer der Faltprozedur für verschiedene Parameter (u.a. Geschwindigkeit, Beschleunigung) berechnet und visualisiert.
-

2013 | ROTH + WEBER

Portierung der Embedded-Software eines Druckersystems auf eine neue Hardware-Plattform, Refaktorisierung der Software-Architektur und Entwicklung neuer Software-Module.

Aufgaben:

- Entwicklung der neuen Software-Architektur
- Modularisierung vorhandener, wiederverwendbarer Software-Komponenten
- Implementierung von neuen Software-Modulen für CAN, I²C, UART & 1Wire Kommunikation
- Integration des vorhandenen Bootloaders
- Implementierung einer Schrittmotoransteuerung
- Integration und Optimierung der Druckablaufsteuerung

Hardware:

- MCU: Renesas RX600

Tools:

- IDE: Renesas e²Studio + J-Link
-

2013 | ROTH + WEBER

Weiterentwicklung einer Software für die Generierung von Typenschildern für verschiedene Maschinentypen in Java.

Aufgaben:

- Erweiterung der Software für neue Maschinentypen
-

2013 | ROTH + WEBER

Weiterentwicklung eines vorhandenen Scannersystems inklusive Portierung der Embedded-Software und Anpassung des Windows-USB-Treibers.

Aufgaben:

- Adaptierung der Embedded-Software an die neue Hardware-Plattform
- Erweiterung des USB-Treibers für die Kommunikation mit der angepassten Embedded-Software

Hardware

- MCU: Cypress EZ-USB FX3
-

2012 | ROTH + WEBER

Entwicklung einer Client-Server-basierten Lizenzverwaltung-Software in Python für verschiedene Maschinentypen.

Aufgaben:

- Implementierung einer Schnittstelle für den Datentransfer mit dem ERP-System
 - Implementierung einer Trial-Lizenzen-Funktionalität
-

2011 | ROTH + WEBER

Entwicklung einer Embedded-Software für ein neues Scannersystem.

Aufgaben:

- Entwicklung der Software-Architektur
- Implementierung eines CAN-Stacks mit ISO-TP-Kommunikation
- Implementierung eines Bootloader mit CAN-Interface
- Implementierung der Scan-Ablaufsteuerung

Hardware:

- MCU: Renesas M16C
-

2010 | ROTH + WEBER

Entwicklung eines User-Mode-Treibers für die Kommunikation eines Scannersystems mit einem Windows PC via USB2.

Aufgabe:

- Adaptierung der LibUSB für die USB-Kommunikation
- Implementierung der performancerelevanten Software-Komponenten als DLL in C.
- Implementierung eines socketbasierten Kommunikationsdienstes in Python mit XML-RPC-Interface, der den Zugriff auf die Treiber-DLL via ctypes realisiert.

Hardware:

- MCU: Cypress EZ-USB FX2

Tools:

- Compiler: MinGw
-

2010 | ROTH + WEBER

Entwicklung einer Java-Applikation mit Benutzeroberfläche für die Konfiguration und Verwaltung eines Scannersystems.

Aufgaben:

- Implementierung einer Benutzerverwaltung
 - Implementierung einer Parameterverwaltung
 - Implementierung einer Firmware-Update-Funktion
 - Implementierung einer XMLRPC-basierten Kommunikation mit dem Scannersystem
-

2010 | ROTH + WEBER

Betreuung und Weiterentwicklung einer Embedded-Software für ein Falmmaschinensystem.

Aufgaben:

- Implementierung einer Endlosfaltprozedur
- Bugfixing

Hardware:

- MCU: Philips 80C592
-

Elkenroth, 11.08.2025