

*o p e n*

*MATTHIAS THÖMEL*

# **Projektliste**

Kundeninformation

Version DE 1.27

© copyright by matthias thömel – hannover – germany 1999-2024

date: 10-02-2024

author: m.thoemel

o p e n

## 1. Quality Assurance

Document	
Title	Projektliste DE
ID	
Related Documents	
Author / Department	M.Thömel
File Position	/media/d/MATTHIAS/Jobsuche_2024/docs/de/Projektliste Matthias Thoemel DE v1.27.docx

History	Reason for change	Name / Department	Date
Version 1.19	Added ESP32 IoT Projekt	M. Thömel	06-Nov-2017
Version 1.20	Added Android, EFM32	M. Thömel	06-Jul-2019
Version 1.21	Update EFM32	M. Thömel	02-Okt-2019
Version 1.22	Added: AILO Ki Projekt	M. Thömel	30-Dec-2019
Version 1.23	Added: DPRE Smart ESP32	M. Thömel	06-Sept-2021
Version 1.24	Added: ESP32 Audio	M. Thömel	02-Feb-2022
Version 1.25	Added: Azure IOT HUB	M. Thömel	17-Mar-2022
Version 1.26	Added: Small Projects 2022	M. Thömel	28-Feb-2023
Version 1.27	Update last projects	M. Thömel	10-Feb-2024

*o p e n*

**Contents**

1. Quality Assurance ..... 2

2. Projektliste..... 4

## 2. Projektliste

Es folgen alle relevanten Projekte, in der Matthias Thömel mitgewirkt hat. Der letzte Eintrag ist der aktuellste. Die genauen Tätigkeiten sind unter dem Punkt Tätigkeiten aufgeführt:

### **Programmierung eines universellen grafischen Visualisierungssystems für die Darstellung unterschiedlichster Meßdaten, m+p PLOT 1000.**

Zeitraum: 1986 (6 MM)

Hardware: HP1000 (RTE-A), HP9000 (HP-UX)

Software: FORTRAN, C

Tätigkeiten: \* Herstellung verschiedener Treiber für Plotter und Bildschirm

\* Umstellung der Software von RTE-A auf HP-UX

### **Herstellung einer universellen, masken- und menügestützten Benutzeroberfläche für die HP2392-Terminals an einer HP1000.**

Zeitraum: 1986 (4 MM)

Hardware: HP1000 (RTE-A/L), HP9000 (HP-UX)

Software: FORTRAN, C, HP2392A-Escape-Sequenzen

Tätigkeiten: \* Optimierung der I/O-Treiber für die MUX-Schnittstelle

\* Implementierung des gesamten Terminal-Funktionsumfangs in die SW

\* Entwicklung weiterer Treiber für andere Terminals und für HP-UX-Console

\* Umstellung der Software auf HP-UX

### **Auftragsdatenverarbeitung Stahlwerk Klöckner, Georgsmarienhütte mit Stangenzählung und Qualitätsüberwachung.**

Zeitraum: 1987 (4 MM)

Hardware: HP1000 (RTE-A)

Software: FORTRAN

- Tätigkeiten:
- \* Realisation von Modulen der maskengestützten Auftragsverwaltung
  - \* Anbindung externer Meßsysteme für Qualitätssicherung (X/R-Regelkarte)
  - \* Hilfe bei Installation der Soft- und Hardware vor Ort
  - \* Dokumentation und Handbucherstellung
  - \* Schulung und telefonische Betreuung
  - \* Erweiterungen der Software nach Auslieferung

### **Auftragsdatenverwaltung Blankbetrieb des Stahlwerks Klöckner, Georgsmarienhütte mit Qualitätskontrolle per X/R-Regelkarte. Meßmittel: Laserdiameter.**

Zeitraum: 1987 (3 MM)

Hardware: HP1000 (RTE-A), Laserdiameter

Software: FORTRAN, IMAGE/1000

- Tätigkeiten:
- \* Planung des Softwaredesignes für ca. 70% der Software
  - \* Herstellung eines Bedingungs- und Formelinterpreter für die parametrisierte Arbeitseinsatzplanung im gesamten Betrieb
  - \* Verwaltungsprogramme für die Datenverarbeitung

## o p e n

- \* Anbindung des Laserdiameters zur Meßwerterfassung
- \* Reportgenerator für den X/R-Regelkartenausdruck
- \* Anbindung der IMAGE/1000-Datenbank an alle Programmteile
- \* Dokumentation und Handbucherstellung
- \* Installation und Wartung vor Ort
- \* Schulung und telefonische Betreuung
- \* Planung und Realisation von nachfolgenden Softwareerweiterungen

**Datenzwischenspeicherung einer Meßboje vor der Insel Sylt, Uni Hannover.**

Zeitraum: 1988 (3 MM)

Hardware: HP85 (BASIC), HP1000 (RTE-A), Modem

Software: BASIC, FORTRAN

- Tätigkeiten:
- \* Planung und Realisation der gesamten Software in HP85 und HP1000
  - \* Erstellung eines einfachen Datentransferprotokolls
  - \* Installation der Hardware und Software in der Uni Hannover

**Meßdatenerfassung und Auswertung am Großen Wellenkanal des Franziusinstituts der Uni Hannover.**

Zeitraum: 1988 (3 MM)

Hardware: HP1000 (RTE-A), als Massenspeicher: 4 UMATIC-Videorecorder per PCM-Modul angebunden,

## o p e n

Software: FORTRAN

- Tätigkeiten: \*
- Anbindung der Videorecorder an HP1000
  - Erstellung eines Spezialtreibers für m+p PLOT 1000 für Endlosdruck
  - \* Analyse HP eigener Vektor-Fonts zur Verwendung in eigenen Programmen
  - \* Installation, Schulung

**Meßdatenaufnahme und -verarbeitung an Feldspritzen, Landesamt für Landwirtschaft, Braunschweig.**

Zeitraum: 1989 (3 MM)

Hardware: HP 9000 (HP-UX), HP-Frontend (BASIC, 68000 Assembler)

Software: C, BASIC, 68000-Assembler.

- Tätigkeiten: \*
- Konzeption der gesamten Software
  - \* Realisation der Software (im Frontend auch als Assemblerprogramm)
  - \* Installation der Software
  - \* Dokumentation und Handbucherstellung

**Steuerung eines Planheitsmeßtisches und Meßdatenaufnahme von 11 Lasermeßköpfen. ALCAN Deutschland.**

Zeitraum: 1990 (4 MM)

Hardware: HP1000 (RTE-A), eigene Centronics-Schnittstelle an HP1000,

## o p e n

eigene Timerhardware auf Breadboard in HP1000 hergestellt,  
Videokarte für HP1000

Software: FORTRAN, IMAGE/1000

- Tätigkeiten:
- \* Konzeption der Software
  - \* Konzeption der Hardware
  - \* Herstellung der Software
  - \* Schneller Höhenlinienalgorhythmus
  - \* Anbindung IMAGE/1000-Datenbank
  - \* Herstellung der Hardware
  - \* Implementation der Hardware und Software in England
  - \* Handbuch- und Dokumentationsherstellung
  - \* Auslieferung bei ALCAN Göttingen
  - \* Schulung

### **Weiterentwicklung des Schallanalysesystems SAVEM auf der HP1000 für VW-Wolfsburg.**

Zeitraum: 1990 (6 MM)

Hardware: HP1000 (RTE-A), Meßdatenaufnahme mit HP1000-  
Erweiterungskarten

Software: FORTRAN, Image/1000

- Tätigkeiten:
- \* Neustrukturierung der gesamten Oberflächenstruktur
  - \* Entwicklung eines Formelinterpreters
  - \* Erweiterung der grafischen Darstellungsmöglichkeiten
  - \* Installation in Wolfsburg
  - \* Handbuch- und Dokumentationsherstellung

## o p e n

- \* Schulung
- \* Wartung und Pflege des Programms

**Allgemeine Entwicklung grafischer Benutzeroberflächen**

- Zeitraum: 1991 (2 MM)
- Hardware: HP9000 (HP-UX)
- Software: C, X-Windows
- Tätigkeiten: \* Einführung der Mitarbeiter in die Programmierung mit Xt-Intrinsics und mit den X-Windows-Funktionen

**Analyseprogramm zur Messung von Einspritzdüsen bei Firma L'Orange.**

- Zeitraum: 1990 (9 MM; 3 MA)
- Hardware: HP1000 (RTE-A)
- Software: FORTRAN, Image/1000
- Tätigkeiten: \* Projektleiter und Koordinator für m+p und dem Subunternehmer
- \* Treiberherstellung für Meßdatenerfassung
  - \* Softwareimplementation
  - \* Installation vor Ort

**Schulung für HP-Paragon Meßdatenerfassungssystem mit 68000- und 56001-Prozessoren in USA.**

- Zeitraum: 1991 (6 MM; 2 MA)
- Hardware: HP-Paragon: Schnelles Meßdatenerfassungssystem

## o p e n

Software: C, Assembler 56001 / 68000

Tätigkeiten: \* Schulung in USA  
\* eine Woche arbeiten mit dem Paragon in den USA um die Einsatzmöglichkeiten des Meßinstruments für m+p-Vibcontrol beurteilen zu können

**Herstellung eines Personalverwaltungs- und Einsatzplanungs-Instruments für das Personal von mehrtägigen Stadionkonzerten für Firma Hannover Concerts und Firma KKS.**

Zeitraum: 1992 (6 MM)

Hardware: PC (MS-DOS)

Software: DBase 4, Clipper.

Tätigkeiten: \* Konzeption und Herstellung der Datenbankstruktur  
\* Erstellung des Softwarekonzepts zusammen mit dem Kunden  
\* Herstellung der Software  
\* Wartung der Software vor Ort  
\* Notdienst

**Herstellung eines Crosscompilers von DBase 4 zu Clipper 5.0.**

Zeitraum: 1993 (2 MM)

Hardware: PC (MS-DOS)

Software: C, DBase4, Clipper

Tätigkeiten: \* Konzeption des Crosscompilers und Herstellung unter Borland Turbo-C

## o p e n

- \* Besondere Aufgabe: Realisation von Fenstertechnik unter Clipper.

**Herstellung einer MIDI-Out-Schnittstelle für ein Tasteninstrument.**

Zeitraum: 1993 (1 MM)

Hardware: 8031-Prozessor

Software: 8031-Assembler

- Tätigkeiten: \*
- Konzeption der Hardware und Software
  - \* Realisation der Hard- und Software
  - \* Test und Einbau in das Instrument

**Herstellung von div. Spielesoftware für den Promotion-Sektor.**

Zeitraum: 1993 (18 MM; 3 MA)

Hardware: Z80-Hardware (eigene Produktion)

Software: Z80-Assembler, GNU-C-Compiler

- Tätigkeiten: \*
- GNU-Compiler auf Target anpassen
  - \* "Multi-Tasking" System entwickeln
  - \* Spielvorschriften in C codieren
  - \* Dokumentation, Softwarewartung

**Projektierung, Realisation und Autorenarbeit für das Wellenreiterprojekt der Zeitschrift ELRAD (siehe ELRAD Artikel).**

Zeitraum: 1992-1993 (15 MM; 3 MA)

Hardware: 56001-Signalprozessor, 68008-Prozessor, PC, div.

## o p e n

## Schnittstellenbausteine

Software: C, C++, 56001-, 68008-Assembler

- Tätigkeiten: \*
- \* Konzeption der Hard- und Software
  - \* Herstellung der Software auf allen Prozessoren
  - \* Dokumentation der Software
  - \* Herstellung der Artikelserie für ELRAD

**Herstellung von Anwendersoftware auf der Basis AutoCAD 12 und 13 für eine Architekturapplikation. Darunter: Layerverwaltung, Treppenkonstruktionsmodul.**

Zeitraum: 1995 (8 MM; 2 MA)

Hardware: PC

Software: AutoCAD12/13, DOS4GW-Extender, C

- Tätigkeiten: \*
- \* Analyse der Treppenbau-Vorschrift
  - \* Herstellung des Softwarekonzept
  - \* Herstellung der Software
  - \* Dokumentation der Software
  - \* Weitere Softwarekomponenten für Tools für Architekten

**Herstellung eines Crosscompilers von AutoLISP (Programmiersprache von AutoCAD) zu C bis zur Version AutoCAD 13C2, für die C-Compiler WATCOM 9.5, 10.5 und Microsoft Visual C++ 2.0 und 4.0.**

Zeitraum: 1995 (8 MM; 2 MA)

Hardware: PC

Software: C / C++ (diverse Compiler)

## o p e n

- Tätigkeiten:
- \* Analyse der Programmiersprache LISP
  - \* Umsetzung der Strukturen auf C-Sprachumfang
  - \* Realisation des Crosscompilers selbst
  - \* Realisation der Emulationsbibliotheken für die C-Programme
  - \* Erstellen des Benutzerhandbuchs
  - \* Vermarktungskonzeption und Vertriebstätigkeiten

**Programmierung umfangreicher Leitstandfunktionen für ein BDE-System unter Windows 3.11. Darunter: Herstellung eines BASIC-Interpreters mit Datenbankfunktion und Formelinterpreter zur Erzeugung kundenindividueller Auswertungen für den Bildschirm und den Windows-Drucker.**

Zeitraum: 1996 (16 MM; 4 MA)

Hardware: PC

Software: C / C++ unter Windows 3.11

- Tätigkeiten:
- \* Realisation fremder Softwarekonzepte unter Windows 3.11
  - \* Konzeption eines Listengenerators inkl. BASIC-Interpreter
  - \* Anwenderschulungen
  - \* Projektierung für Teilaufgaben innerhalb des Leitstands
  - \* Realisierung dieser Teilaufgaben

**Konzeption und Realisation eines IO-Systems für den 68302-Prozessor auf der Basis von pSOS+. Darunter BASIC-Interpreter, Filesystem auf FLASH-PROM, usw.**

## o p e n

Zeitraum: 1996 (4 MM; 2 MA)

Hardware: 68302-Prozessor, FLASH- und andere Hardwarekomponenten

Software: pSOS+, MRI-C-Compiler

- Tätigkeiten:
- \* Konzeption eines BDE-Betriebssystems auf dem 68302-Target
  - \* Konzeption eines eigenständigen I/O-Systems auf der Basis pSOS+
  - \* Realisation diverser Treiber für das I/O-System
  - \* Implementation eines eigenen BASIC-Interpreters unter pSOS+

**Analyse und Programmierung eines Boot-Daemons für HP-UX 9.5 um über LAN angeschlossene HP1000-Rechner per Memory-File zu booten.**

Zeitraum: 1997 (3 MM)

Hardware: LAN-Interface HP9000, LAN-Interface HP1000

Software: C (HP-UX 9.5)

- Tätigkeiten:
- \* Analyse des Bootvorgangs der HP1000-LAN-Karte, wenn eine HP1000 als Server angeschlossen ist
  - \* Konzeption eines Multitasking-Boot-Server-Daemons, der 16 HP1000-Rechner parallel booten kann
  - \* Realisation der Software per C auf HP-UX 9.5 und der Low-Level-Befehle Steuerung des HP9000-LAN-Treibers (/dev/lan0)
  - \* Implementation des Daemons in einer Industrieumgebung um ca. 150 HP1000-Rechner an unterschiedlichen Servern zu booten.

o p e n

**Planung, Realisation und Vertrieb im Internet eines Netzwerk-Spions, der die Daten, die über ein Netzwerk-Kabel gehen, auf unterschiedliche Weise sichtbar machen kann und speichert (als Dateien auf der lokalen Festplatte) für den Einsatz zur automatischen Erfassung von redaktionellen Texten innerhalb einer Rundfunk-Redaktion.**

Zeitraum: 1998 (2 MM)

Hardware: Windows 95, NDIS3-Interface (LAN-Karte oder DFÜ-Adapter)

Software: C (LCC Win32)

Tätigkeiten: \* Analyse der Protokolle: TCP, IP, NETBIOS, FTP, HTTP, SMTP, NNTP, UDP

\* Konzeption der Oberflächensoftware und des Aufzeichnenden Tasks (Realtimemodus bei Windows 95)

\* Realisation der Software per C auf Windows 95 und der Low-Level-Zugriffe auf den NDIS-Adapter

**Planung, Realisation und auch Arbeitsgruppenleitung für ein Datenbanksystem integrierende C-Programmiers-API, die jegliche Art von Datenbanken (gestützt durch eigenentwickelte Treiber: SQL, CDM, u.a.) zur Benutzung in C-Programmen zur Verfügung stellt.**

Zeitraum: 1998-1999 (27 MM; 3 MA)

Hardware: PC, Sun, andere Workstations (NT und Unix)

Software: Unix - C-Compiler, MSVS 6.0 C++

Tätigkeiten: \* Einarbeitung in die DB-Programmierschnittstellen

\* Planung des allgemeinen Interfaces und der Treiber-Steuer-Strukturen

## o p e n

- \* Erstellung der API und der Treiber
- \* Erstellung einer transparenten Transportschicht (sockets) unter Unix und NT 4.0
- \* Implementation dieser API in Applikationen
- \* Vorstellung der Möglichkeiten vor Abteilungsleitern
- \* Leitung der Arbeitsgruppe rund um diese API zur Weiterentwicklung
- \* Entwicklung hierarchischer Strukturanalyse- und Pflegefunktionen

**Planung, Realisation und Arbeitsgruppenleitung für ein Databroker-System, das in einem großen Intranet arbeitet. Leistungsanforderung: mindestens 30.000 Zugriffe pro halbe Stunde mit maximaler Reaktionszeit von 3 Sekunden pro Anfrage bei 60 MB Datenbestand. Ein Unix-Cluster mit verteilten, autonom arbeitenden Data-Brokern wurde realisiert**

Zeitraum: 1998-1999 (10 MM; 3 MA)

Hardware: PC, HP-Workstations

Software: Unix - C-Compiler

- Tätigkeiten:
- \* Design des Systems
  - \* Planung der Arbeit des Teams
  - \* Realisation der Cluster-Bildung per Socket-IO (TCP/IP)
  - \* Installation beim Kunden

**Realisation einer Treibersoftware für ein OROS-Messdatenerfassungssystem, daß in der Schwingregeltechnik eingesetzt wird. Einsatzgebiet der Komponenten: Luft- und Raumfahrt / Militär.**

## o p e n

Zeitraum: 1999.8 – 1999.10

Hardware: OROS-Frontend, PC

Software: MS-Visual Studio 6.0

Tätigkeiten: \* Design des Treibers  
\* Realisierung unter C (PC und OROS – TMS320)

**Mitarbeit bei der Erstellung von Software für den Paketdatentransfer vom Internet zum Mobile Phone (Handy) und zurück für japanische Nutzer (PPDC).**

Zeitraum: 1999.11 – 2002.4

Hardware: Multiprozessorsystem (diverse Power-PC Prozessoren), T1-Interface (US-Standard), Ethernet-Interfaces, Ericsson eigene Hardware, Sun-Workstations

Software: VxWorks (PPC), GNU-Compiler (Unix), ClearCase-Development-Environment (Unix), Streams (Unix und Sun)

Tätigkeiten: \* C-Design des T1-Streams-Drivers  
\* C-Design des L2-Protokoll-Streams-Modules  
\* C-Softwaredesign zur Erhöhung der Performance  
\* Realisierung der Flow-Control über drei Protokollebenen hinweg  
\* Dokumentation (Englisch)

**Softwareentwicklung zur Synchronisation von multiplen Video-Screens eines Multimedia-Video-Systems (Reality.Engine)**

Zeitraum: 2000.11 – 2001.4

Hardware: PC – Pentium, N-VIDIA-Grafikkarte

## o p e n

Software: MS-Windows 98, NEMO (Multimedia-System)

Tätigkeiten: \* C-Design des TCP/IP-Synchronisations-Systems  
\* Design einer virtuellen Welt mit Synchronisation in NEMO

**Softwaredesign und -entwicklung für einen verkehrstechnischen Systemanbieter. Sehr verschiedene Teilaufgaben, inklusive Forschungsprojekt.**

Zeitraum: 2001.5 – 2003.12

Hardware: Motorola DSP56826, PPC604, PPC860, Pentium

Software: Windows-XP, VxWorks, eigenes RTOS, JAVA, CORBA, VisiBroker (auf PPC), TCL/TK, MS-DEV 6.0

Tätigkeiten: \* Entwicklung eines eigenen DSP Multitasking Realtime Operating Systems (miniRTOS)  
\* Realisation eines alten Siemens Modemstandards auf dem DSP  
\* Realisation eines normalen FSK Modems auf dem DSP  
\* Realisation eines sehr alten Siemens-Modemstandards unter Verwendung einer FFT und eines FIR's auf dem DSP  
\* Entwicklung und Anwendung eines DSP-System Testsystems mittels TCL/TK und einer eigenen TK-Lib zur Ansteuerung des DSP.  
\* Realisation einer Sprachanalyse, Datenkompression und Verschlüsselung, Datenübertragung über Telefon, Datendekompression und Entschlüsselung, Sprachsynthese auf dem DSP  
\* Entwicklung einer eigenen Video-Komprimierung / -Dekomprimierung für Videoübertragungen via ISDN Leitung.

## o p e n

Läuft auf dem PPC860 und PPC604. Mit JAVA Client

- \* Entwicklung eines Multi-Messaging-Systems auf VxWorks zum Senden von Fax (inklusive GDI), SMS (via GSM-Modem), Sprache und Mail und zum Empfang von SMS, DTMF und Mail. Angesteuert über VisiBroker (CORBA)
- \* Entwicklung diverser Devices und Treiber auf VxWorks

**Systemdesign, Softwaredesign und Softwareprogrammierung für den C55 DSP im OMAP5912 / OMAP5946 (Trinity) Chip von Texas Instruments als Erweiterung des DSP/BIOS.**

Zeitraum: 2004.4 – 2007.4

Hardware: TI OMAP 5912 / OMAP 5946

Software: DSP/BIOS, Chip Support Library, Code Composer Studio, TCONF, ClearCase, ClearMake, ClearAudit

- Tätigkeiten:
- \* Systemdesign eines multi-Applikations Environments auf Basis des DSP/BIOS mit Dynamischen Applikationen
  - \* Entwicklung des Applikationsmanagements für den DSP C55
  - \* Entwicklung des neuen, schnelleren Memory Managements für alle Speicherbereiche (Ersatz der TI MEM-Funktionen)
  - \* Entwicklung der Inter-Prozessor-Kommunikation innerhalb der Prozesse des DSP und nach extern zum ARM-Core (mit Zero-Copy Eigenschaften)
  - \* Entwicklung eines DMA Management für die einfach Anwendung des C55-DMA-Controllers
  - \* Erweiterung des DSP/BIOS um Analysefunktionen

## o p e n

(Task-Tree, Mem-Usage, etc...)

- \* Zusätzliche Optimierungen für alle C55 DSP Programme (RAMSET Nutzung, Stackmove to internal memory)
- \* Aufsetzen einer Buildumgebung für das DSP Software Projekt, das verschiedene Targets und verschiedene Projekte unterstützt
- \* Technischer Support aller Entwickler im Team
- \* Softwareentwicklung für den ARM Prozessor im OMAP im Bereich Low-Level-Treiber und Startup
- \* Design und Entwicklung eines übergreifenden Treibermanagement-Frameworks im ARM Core für den Startup, den Shutdown und die Power-Drop Ereignisse (Stop-Start-Situation)
- \* Systemdesign für das gesamte Projekt (ARM und DSP)

### **Teamleitung Widget-Team für die Entwicklung eines HMI für ein Radio-Navigations-System auf Basis von generiertem Code aus einem GUIDE-Model**

Zeitraum: 2007.5 – 2008.1

Hardware: TI OMAP 5912 / OMAP 5946

Software: C / C++, LUA-Scripting, ASM-Compiler

- Tätigkeiten:
- \* Teamsetup (Teamgröße 6 Entwickler) und Rampup
  - \* Anforderungsanalyse und Design der Software
  - \* Auswahl der Tools und LUA als embedded Script-Sprache für die Widgets
  - \* Teilprojektleitung und –planung
  - \* Konzept eines universellen Audio-Management für ein

o p e n

RNS

- \* Enger Kontakt zum Auftraggeber mit technischen Projektmanagement

### **Team- und Projektleitung vom Team Applikationsschicht-Entwicklung für ein Radio-Navigations-System und Systemdesign**

Zeitraum: 2008.6 – 2010.1

Hardware: PC, SH4 (und andere)

Software: JAVA (J9 / Java 1.6), Eclipse, SVN, Tortoise, FindBugs, Checkstyle, und andere

- Tätigkeiten:
- \* Teamsetup (Teamgröße 15 Entwickler, Projektgröße 50 Entwickler und Modellierer) und schnelles Rampup einzelner Mitarbeiter
  - \* Prozessentwicklung für die Entwicklung im gesamten Projekt, das Fehlermanagement, Integrations-Strategie, Konzern-Schnittstellen-Strategie und -Verfahren
  - \* Teamleitung und Teilprojektleitung für 15 Entwickler
  - \* Codereviews
  - \* Technische Konzernabsprachen
  - \* Beratung im Bereich Systemdesign, Headunit-Funktionalität, BAP, Targets, Entwicklungsmethodik, Testmethodik, Projektleitung insgesamt
  - \* Technische Workshops

### **Systemdesigner und Senior-Developer / Architekt für ein HMI- und Navigationsmodul (embedded Linux und t-kernel auf ARM-Multicore System on Chip)**

## o p e n

- Zeitraum: 2010.2 – 2013.10
- Hardware: Triton Prozessor (3 Core ARM Architektur, embedded Linux), MOST-Bus
- Software: Embedded Linux (Monta Vista), SQLite embedded, uBuntu X86 Linux, t-kernel, C/C++, STL, Eclipse-CDT, GNU EABI Toolchain, K2L MOST Tooling, Trace-Tool, perf für Linux, Lauterbach-Debugger, ClearCase
- Tätigkeiten: \*
- \* Erste Portierung vorhandener Software auf X86 Linux
  - \* Erste Portierung vorhandener Software auf embedded Linux für GNU EABI gcc inklusive der Buildsystem Anpassungen für Cross-Compile mit GNU EABI
  - \* Einführung von Eclipse remote CDT Debugging für Target unter X86 Linux mit GNU EABI gdb/gdbserver
  - \* Entwicklerworkshops für C/C++, Buildsystem, Download-System, Remote-Debugging
  - \* Zwei Geräte-Muster Präsentationen in Detroit/USA (2011)
  - \* Entwicklung eines Memory-Tracers als Preload für embedded Linux
  - \* Entwicklung eines Function-Tracers als Preload für embedded Linux
  - \* Redesign der Mediaplayer Datenbank im embedded Linux
  - \* C/C++ Code Review der gesamten Modulsoftware
  - \* Coaching und Steuerung der Mediaplayer Entwicklergruppe bei Monta Vista
  - \* Fehlersuche und Fehleranalyse im Gesamtsystem

## o p e n

(MOST, externe Komponenten, HMI-Modul)

- \* Leitung der Performance Task-Force für den t-kernel Teil des Systems
- \* Support aller Entwickler im Bereich Fehlersuche, Debugging, Codedesign, Architektur, Systemdesign
- \* Mitarbeit im Bereich Performance Analyse im embedded Linux
- \* Performance Analyse einiger Prozesse im embedded Linux
- \* Analyse von Audio Problemen im Gesamtsystem (MOST, externe Komponenten, t-kernel, Linux)
- \* Betreuung aller Entwickler und Fachteamleiter in technischen und systemischen Fragen
- \* Fachteamleitung für Media Player Core Entwicklungsteam
- \* Entwicklung eines State Machine Frameworks für generierte State Machines mit ArgoUML
- \* Entwicklung eines sehr schnellen SQLite3 Abstraktionslayers für eine Mediaplayer Applikation

### **Privates Parallelprojekt: Hochparallele Bilddatenverarbeitung auf Nvidia GPU Stream Prozessoren mit CUDA**

Zeitraum: 2012.12 – 2013.02

Hardware: Nvidia GPU (GT-260)

Software: Windows, CUDA 5.0, WT (WebToolkit), Videoinput

Tätigkeiten: \* Entwicklung eines Thread-Control und -Scheduling Frameworks um tausende Threads auf den Stream

## o p e n

Prozessoren zeitgesteuert zu starten und zu unterbrechen

- \* Entwicklung eines Webfrontends für die CUDA Software
- \* Entwicklung von Filter und Bilderkennungsalgorithmen für die Stream Prozessoren

### **Entwickler für mobile Geräte im Bereich öffentlicher Nahverkehr**

Zeitraum: 2013.10 – 2014.02

Hardware: ARM CPU

Software: WindowsCE 4.2, Subversion, MS Developer Studio

- Tätigkeiten:
- \* Design und Entwicklung eines UDP Scramblers der das Verhalten von UDP Paketen auf einer Luftschnittstelle simuliert (als Testtool)
  - \* Design und Entwicklung eines Kommunikationslayers für ein kundenspezifisches Protokoll zwischen Leitstelle und den Bussen im öffentlichen Nahverkehr
  - \* Design und Entwicklung von Tools (perl/tk) für einen Software Produktionsprozess für ein mittelständisches Unternehmen auf Basis von SVN

### **Systemdesigner und Senior-Developer (embedded Linux auf iMX6 Multicore)**

Zeitraum: 2014.02 – 2015.09

Hardware: iMX6, Kundenhardware

Software: Embedded Linux, SQLite3 embedded, uBuntu X86 Linux, C/C++, STL, Eclipse-CDT, GNU EABI Toolchain, perf für Linux, ClearCase

- Tätigkeiten:
- \* Kernteam-Member für Media Player Core

## o p e n

## Entwicklungsteam

- \* Bugfixing, Weiterentwicklung Mediaplayer
- \* Design und Implementierung für Content Sharing

## Funktion

- \* Einführung des StateMachine-Framework für Phone Komponente, Schulung der Mitarbeiter, Designmeetings für diese neue Komponente

**Private Parallelprojekte: Android Apps für Phone und Wear**

Zeitraum: 2015.02 – 2015.07

Hardware: diverse Android Phones und Tables; Android Wear

Software: JAVA, Android JAVA API, Open GLES, Android Studio, Corel Draw, Videoschnitt

- Tätigkeiten:
- \* Entwicklung von Android Spielen und Apps:
    - \* TopLoad: Casual Game
    - \* Adjustment: Casual Game
    - \* Nexty – Advanced Bluetooth Chat für Phone und Android Wear (Watch)
    - \* Phone Remote Control für Android Wear
  - \* Komplette Produktionskette:
    - \* Planung, SW-Design, Grafik-Design, Implementierung, Test, Vermarktung (mit Funktionsvideos)

Link <https://play.google.com/store/apps/developer?id=Matthias+Credits>

**Linux Treiberentwicklung: DMA Treiber / Device Tree**

## o p e n

Zeitraum: 2016.10 – 2016.12

Hardware: Altera SocKIT (ARM CPU + FPGA)

Software: Linux kernel, C, gcc toolchain, make, Eclipse CDT, perl, GoogleTest

- Tätigkeiten:
- \* Entwicklung eines DMA Treibers unter Linux als Teil einer Plattform-Entwicklung für Magnet-Resonanz-Spektrografen
  - \* Entwicklung eines Tracing Systems mit minimalem Overhead, einsetzbar in Linux Treibern und Userland
    - \* Mit bash und perl Scripten zur Visualisierung der Traces
  - \* Analyse des vorhandenen Treibers
  - \* Integration von Google Test Framework und Herstellung von Google Testcases
  - \* Softwaredesign des DMA Treibers
  - \* Konfiguration des device tree für neuen DMA Treiber und vorhandene Produkt Treiber
  - \* Entwicklung eines Character Device Drivers zur rekursiven Interpretation des spezifischen Plattform device tree
  - \* Entwicklung, Test, Dokumentation

### Linux Middleware: FileObserver für Business-Software

Zeitraum: 2017.02 – 2017-02

Hardware: Linux PC

Software: Linux, C++, g++, make, Eclipse CDT, inotify, boost, TCP/IP-sockets, GoogleTest

## o p e n

- Tätigkeiten: \*
- Software Design eines Linux File-Observers und File-Manipulators für eine Windows Business-Software
  - \* Entwicklung des Designs mit C++, inotify und boost
  - \* Entwicklung von 80 Google Tests (Funktion und Regression)
  - \* Test, Dokumentation

**Bilderkennung-KI: Projekt mit TensorFlow**

- Zeitraum: 2017.05 – 2017-06
- Hardware: Windows PC
- Software: Windows, python, Eclipse for python, TensorFlow, cv
- Tätigkeiten: \*
- Privates Projekt zur Einarbeitung in TensorFlow, einem Framework für Programmierung von Künstlicher Intelligenz (KI)
  - \* Inhalt: Objekterkennung in Bildern mit gleichzeitiger Positionsbestimmung der Objekte im Erkennungsschritt
  - \* Anwendung eines eigenen Convolutional Neuronal Networks (CNN) mit 3 Convolution- und 2 Function-Netzen
  - \* Ergebnis: Unterschiedliche Objekte werden erkannt und ihre Position im Bild wird angezeigt.

**Komplettes Daten-Routing-Modul zwischen UART und WIFI für IoT**

- Zeitraum: 2017.07 – 2017-11
- Hardware: ESP32, Linux-PC
- Software: Linux, xtensa-Cross-Compiler, freeRTOS, ESP-IDF, Eclipse CDT, GoogleTest, python
- Tätigkeiten: \*
- Architektur und Design eines Daten-Routers auf einem ESP32 Chip für ein IoT Consumer-Gerät. Fokus schnellste

o p e n

- Datenverarbeitung, höchste Zuverlässigkeit (keine Resets)
- \* Implementierung/Erweiterung der Treiber-Software für die Schnittstellen UART und WIFI AP/Station
  - \* Implementierung des Router-kernels und der Routing State-Machine.
  - \* Implementierung der Web Protokolle: WebSocket, UPnP, SSL
  - \* Design/Implementierung 75 High-Level Modultests für das GoogleTest Framework
  - \* Implementierung eines Simulators der Außenwelt für die GoogleTests
  - \* Dokumentation und Testing

### **Android App für Messdatenerfassung und Visualisierung**

Zeitraum: 2017.11 – 2019-07

Hardware: Android Tablet/Mobile

Software: Android(X), SQLite, JAVA

- Tätigkeiten: \*
- Entwerfen und Entwickeln der gesamten App als Einzelpersonen-Projekt. Inklusive Simulation der Messgeräte und Testfunktionen.
  - \* JAVA unter AndroidX
  - \* GPU Unterstütztes Grafik-Rendering der Daten-Plots
  - \* Extrem schnelle sqllite Anbindung (für mehrere Millionen Datensätze pro Sekunde)
  - \* ProGuard
  - \* Multilingual

## o p e n

- \* Portrait und Landscape Unterstützung
- \* WebViews / JavaScript Anbindung an JAVA
- \* USB-Host Schnittstelle
- \* GPS Anbindung
- \* Graddle-Programmierung (Mehrere Flavors für die App)
- \* Dynamische Views (zur Laufzeit erzeugt)
- \* Foreground-Service (für Messungen im Hintergrund)
- \* Unit-Tests / Demo-Mode der App
- \* Obfusking eines Teils der Datenbank (Maßnahme gegen Reverse Engineering)

Google <https://play.google.com/store/apps/details?id=com.ibp.btl.biometer.hdm>  
Play [ter.hdm](https://play.google.com/store/apps/details?id=com.ibp.btl.biometer.hdm)

**Migration/Weiterentwicklung von Geräte-Software auf EFM32**

Zeitraum: 2019.05 – 2019-10

Hardware: EFM32

Software: C

- Tätigkeiten:
- \* Migration einer alten C-Software auf den neuen EFM32
  - \* Anpassung / Neuentwicklung der Low-Level Treiber für USART, I<sup>2</sup>C, A/D-Wandler, Flash-Speicher, LDMA Anbindung, USB Treiber, LowPower-Mode
  - \* Anbindung eines Bluetooth-LE-Modems zur zusätzlichen Datenübertragung
  - \* Inbetriebnahme und Test

## Entwicklung einer Künstlichen Intelligenz / Maschinen-Lernen

Zeitraum: 2019.10 – 2019-12

Hardware: Intel / NVIDIA GPU RTX2070

Software: C++14, Nvidia-Cuda-C, WebToolkit, OpenCV, OpenPort

Tätigkeiten: \* Entwicklung des Maschinen-Lernens auf der GPU mittels CUDA von NVidia: Schnelles, paralleles Vergleichen von Erinnerungen (1,5 GB pro 62,5 ms)

\* Aufbereitung durch DSP Routinen des Eingangssignals auf dem Intel Core

\* Anbindung von Audio und Midi-Geräten

\* Entwicklung der Benutzeroberfläche, die in einem Browser läuft mittels WebToolkit (C++14)

\* Aufsetzen eines MSI Installers für Kundenlieferung

\* Design und Implementierung einer Webseite für die Vermarktung: <http://ailo.show>

## Entwicklung einer Steuerungssoftware für ein Dach-Lüftersystem

Zeitraum: 2021.01 – 2021.12

Hardware: ESP32 / Kundeneigene Hardware

Software: ESP-IDF-4.2, C, Eclipse-IDE, mbettls, AES, HTTP-Server, WIFI, HTML, JavaScript, JSON, SMTPAuth, OTA, MQTT, MODBUS, OneWire, diverse HW-Interfaces, SecureBoot, Encrypted App

Tätigkeiten: \* Entwicklung einer Steuerungs- und Management-Software für Dach-Zwangsentlüfter-Systeme auf ESP32 mit angeschlossener Kunden-Hardware

## o p e n

- \* Architektur der gesamten Software
- \* Design aller Komponenten
- \* Patching Espressif IDF um 24/7 Stabilität zu erreichen und memory leaks zu fixen
- \* Entwicklung der interaktiven maschineninternen HTML Seite inklusive JavaScript Anteile
- \* Entwicklung fehlender oder instabiler Treiber (zum Beispiel: OneWire, Modbus-Master-Seriell)
- \* Migration von OpenSource Bibliotheken (Zum Beispiel: MQTTServer)
- \* Design und Implementation eines verschlüsselten OTA Update-Wegs auf einem Secure Boot ESP32 mit Auto-Verschlüsselung aktiv
- \* Ansteuerung sehr vieler verschiedener Bauelemente über diverse HW-Schnittstellen

**Entwicklung einer Echtzeit-Audio-Datenverarbeitung und Analyse**

Zeitraum: 2022.01 – 2022.03

Hardware: ESP32, ES8388 Codec auf LyraT Board (Espressif)

Software: ESP-IDF-4.3.1, C, Eclipse-IDE, HTTP-Server, WIFI, HTML, JavaScript, diverse HW-Interfaces, JPEG, FFT

- Tätigkeiten:
- \* Entwicklung einer Software zur Echtzeit-Audio-Datenerfassung auf dem ESP32 (WROVER-E)
  - \* Ansteuerung des ES8388 Codex für Input und Output (I2C und I2S)
  - \* Implementierung von Optimierung, Vorverstärkung,

## o p e n

## Automatic Level Control

- \* Entwicklung der Nebengeräusch-Unterdrückung inkl. Phase-Shift
- \* Implementierung eines Software Audio-Kompressors
- \* Implementierung einer FFT in Echtzeit
- \* Echtzeit-Weiterleitung von Frequenz-Wasserfall-Bildern an das Web-Interface
- \* Architektur der gesamten Software
- \* Design aller Komponenten
- \* Patching Espressif IDF um 24/7 Stabilität zu erreichen und memory leaks zu fixen
- \* Entwicklung der interaktiven maschineninternen HTML Seite inklusive JavaScript Anteile
- \* Migration von OpenSource Bibliotheken (Zum Beispiel: FFT, JPEG)
- \* Anbindung der Microsoft Azure IOT HUB Cloud für sekundliche Telemetrie und Command (auf Basis der ESP32-MQTT Komponente)

**(Weiter-)Entwicklung einer Machine-Learning Software für einen autonomen Roboter**

Zeitraum: 2022.03 – 2022.05

Hardware: RosBOT 2 (<https://husarion.com/manuals/rosbot/>), Kamera, 3D-Tiefen-Kamera, Rad-Ansteuerungen, Positionsbestimmung, Entfernung-Sensoren, Nvidia-GPU

Software: C auf Rosbot 2.0 OS (embedded), C++/C unter Windows 10, CUDA unter Windows

## o p e n

Tätigkeiten: \* Weiterentwicklung einer Machine-Learning-Software für die autonome Bilderkennung und Klassifizierung, autonome Roboter-Bewegungssteuerung, Tiefenkamera-Auswertung (Entfernungsmessung), autonome Verhaltenssteuerung des Roboters, autonome Exploration der Umgebung durch den Roboter und vieles mehr.

**Entwicklung einer JAVA Software für ein LTE-Terminal/Modem**

Zeitraum: 2022.08 – 2022.09

Hardware: Gemalto ELS61 LTE Terminal, Serielle Schnittstellen

Software: JAVA, AT-Befehle, Win32 C Entwicklung, Microsoft Visual Studio

Tätigkeiten: \* Entwicklung eines transparenten Datenrouters in JAVA mit gutem Reconnect-Verhalten  
\* Entwicklung eines Download-Tools für das LTE-Terminal in Win32 C

**Weiter-Entwicklung einer Steuerungssoftware für ein Dach-Lüftersystem**

Zeitraum: 2023.01 – 2023.02

Hardware: ESP32 / Kundeneigene Hardware

Software: ESP-IDF-4.2, C, Eclipse-IDE, mbedtls, AES, HTTP-Server, WIFI, HTML, JavaScript, JSON, SMTPAuth, OTA, MQTT, MODBUS, OneWire, diverse HW-Interfaces, SecureBoot, Encrypted App

Tätigkeiten: \* Einbindung des ESP32 MQTT Clients

## o p e n

- \* Implementierung eines Betriebs-Daten-Recorders auf SD-Karte
- \* Implementierung eines Betriebs-Daten-Plotters auf dem ESP32 internen http-Server in Javascript und C
- \* Stabilisierung der WIFI-Software in Bezug auf Mesh-Netze und der Connection-State-Machine
- \* Anpassungen für automatisierte Produktion der Maschinen

### **Weiter-Entwicklung der Android App für Messdatenerfassung und Visualisierung**

Zeitraum: 2023.02 – jetzt

Hardware: Android Smartphones

Software: JAVA, gradle, SQLite3, bash scripting, Ubuntu, Android-Studio

- Tätigkeiten:
- \* Anpassung der App an Android 13
  - \* Erweiterung der App um einen Daten-Recorder und -Exporters
  - \* Erweiterung der App um ein Messdaten-Dokumentations-System zum Prüf-Zertifikats-Export mit Kundenspezifischen Templates
  - \* Import der neuesten SQLite3 AAR Bibliothek in die App
  - \* Python-Scripting für automatisierte Tests auf Endgeräten
  - \* Performance- und Stabilitäts-Tests und -Optimierungen

<end of document>