

Entwicklung eines echtzeitfähigen softwaredefinierten Rundfunkempfängers für Desktop-Computer

Von der Fakultät für Maschinenbau, Elektrotechnik und
Wirtschaftsingenieurwesen der Brandenburgischen Technischen
Universität Cottbus - Senftenberg

zur Erlangung des akademischen Grades eines
Doktors der Ingenieurwissenschaften

vorgelegt von
Master of Science
Piotr Szegvári

geboren am 12. November 1979 in Cottbus (Deutschland)

Vorsitzender:	Prof. Dr.-Ing. habil. Matthias Wolff
Gutachter:	Prof. Dr.-Ing. habil. Christian Hentschel
Gutachter:	Prof. Dr.-Ing. habil. Uwe E. Kraus

Tag der mündlichen Prüfung: 09.10.2014

Berichte aus der Informationstechnik

Piotr Szegvári

**Entwicklung eines echtzeitfähigen
softwaredefinierten Rundfunkempfängers
für Desktop-Computer**

Shaker Verlag
Aachen 2015

Bibliografische Information der Deutschen Nationalbibliothek

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

Zugl.: Cottbus-Senftenberg, BTU, Diss., 2014

Copyright Shaker Verlag 2015

Alle Rechte, auch das des auszugsweisen Nachdruckes, der auszugsweisen oder vollständigen Wiedergabe, der Speicherung in Datenverarbeitungsanlagen und der Übersetzung, vorbehalten.

Printed in Germany.

ISBN 978-3-8440-3582-7

ISSN 1610-9406

Shaker Verlag GmbH • Postfach 101818 • 52018 Aachen

Telefon: 02407 / 95 96 - 0 • Telefax: 02407 / 95 96 - 9

Internet: www.shaker.de • E-Mail: info@shaker.de

„Computer sind nutzlos, sie können uns nur Antworten geben.“

(Pablo Picasso, geb. 25.10.1881, ges. 08.04.1973)

Dankeswort

Die vorliegende Dissertation entstand im Rahmen einer vierjährigen Tätigkeit am Lehrstuhl Medientechnik der Fakultät Maschinenbau, Elektrotechnik und Wirtschaftsingenieurwesen an der Brandenburgischen Technischen Universität Cottbus. Daher gilt mein erster Dank dem Lehrstuhlinhaber Herrn Prof. Dr.-Ing. Christian Hentschel, der mit seinem Vertrauen in meine Person die Entstehung dieser Arbeit erst ermöglicht hat.

Die Grundlage für das Thema war indessen meine Masterarbeit, die ich am selben Lehrstuhl im Rahmen eines Drittmittelprojekts in Zusammenarbeit mit der Firma Pinnacle® Systems¹ bearbeiten durfte. In diesem Zusammenhang danke ich Herrn Dr. Alexander Roy. Er interessierte sich schon frühzeitig für das Thema des softwaredefinierten Empfängers im Bereich der Unterhaltungselektronik. Durch dieses Interesse war er an unseren Lehrstuhl herantreten, um eine Machbarkeitsstudie für die Realisierung von Rundfunkempfängern [211] auf Desktop-Computern in Auftrag zu geben. Auch für die daran anschließende dreimonatige Zusammenarbeit, die für das Programmgerüst, für weitere wichtige Impulse und für die finanzielle Unterstützung des Projekts über die eigentliche Laufzeit hinaus verantwortlich war, sei an dieser Stelle recht herzlich gedankt.

Ein knappes Jahr später konnte dann aufgrund eines zweijährigen Stipendiums, das im Rahmen der Internationalen Graduiertenschule² an der BTU Cottbus verliehen wurde, auf den vorhandenen Grundlagen die Weiterführung der Dissertation in Angriff genommen werden. Für diese Möglichkeit gilt stellvertretend für die Verantwortlichen der Graduiertenschule mein Dank Herrn Prof. Dr.-Ing. Heinrich-Theodor Vierhaus vom Lehrstuhl Technische Informatik der BTU Cottbus.

Die Beendigung der Dissertation war nach Ablauf der Graduiertenschule nur durch ein weiteres Drittmittelprojekt möglich, welches als Teilzeitprojekt eingeplant war und mir damit eine parallele Weiterarbeit an der Dissertation ermöglichte. In diesem Zusammenhang gilt mein besonderer Dank meinem Kollegen Dr.-Ing. Dirk Höpfner, der die entscheidende Kraft bei der Akquirierung dieses Projektes war. Neben ihm gilt mein kollegialer Dank auch Herrn Dr.-Ing. Stefan Schiemenz. Beide Kollegen standen nicht nur für fachliche Diskussionen immer zur Verfügung, sodass die Arbeitszeit manchmal viel zu schnell verging.

¹ Ehemals eine Tochtergesellschaft von Avid® und aktuell von Hauppauge® – www.pctvsystems.com

² Im Rahmen der Klasse A – DEDIS Nano von November 2007 bis Dezember 2009

Weiterhin danke ich allen an dieser Dissertation beteiligten studentischen Hilfskräften, Bachelor- und Masterstudenten, die einiges und in speziellen Fällen Beachtliches mehr zum Gelingen dieser Dissertation beigetragen haben.

Mein größter Dank allerdings gilt den Menschen, ohne die ich nicht hätte studieren, geschweige denn meine Dissertation schreiben können. Zu Ihnen zählen die, die mir täglich so viel Alltagsarbeit abgenommen haben, dass ich 24 Stunden am Tag hätte forschen können. Zu Ihnen zählen auch die, die durch ablenkende Aktivitäten meine immer knapper werdenden Ressourcen schnellst möglichst aufzufüllen versuchten. Deshalb gilt der größte Dank meinen Eltern, meinem Bruder nebst Schwägerin, meiner Freundin, meiner Tochter und meinen Freunden.

Danke!

Kurzfassung

Unter einem echtzeitfähigen softwaredefinierten Rundfunkempfänger versteht diese Arbeit einen Rundfunkempfänger, dessen Signalverarbeitung durch Software auf programmierbaren Hardwareplattformen ausgeführt wird, anstatt mithilfe fest verdrahteter Hardwarekomponenten. Die Echtzeitfähigkeit wird durch das in dieser Arbeit entwickelte Gesamtkonzept und innerhalb des neu entwickelten Programmgerüsts durch sogenannte skalierbare Module erreicht. Das Gesamtkonzept berücksichtigt, dass jeder derzeitige und zukünftige Rundfunkstandard durch eine einfache XML-Datei beschrieben und sofort ausgeführt werden kann. Die skalierbaren Module sorgen dafür, dass selbst komplexe Empfängertypen auf leistungsschwachen Plattformen mithilfe des skalierbaren Ansatzes auf Umgebungseinflüsse lokaler aber auch globaler Art reagieren können. Zu den globalen Einflüssen zählen schlechte Empfangsverhältnisse, die durch schlechtes Wetter, eine stark bebaute Umgebung oder Störsignale hervorgerufen werden. Zu den lokalen Umgebungseinflüssen zählen in ihrer Verfügbarkeit schwankenden Ressourcen, was ein softwaredefinierter Empfänger zur Ausführung in Echtzeit kompensieren können muss. Diese Ressourcen setzen sich in erster Linie aus der Rechenleistung, der vorhandenen Prozessoren und dem Arbeitsspeicher zusammen. Bei einem kurzzeitigen oder länger anhaltenden Mangel an Ressourcen ist ein echtzeitfähiger Rundfunkempfänger - im Gegensatz zu einem herkömmlichen Rundfunkempfänger - in der Lage, durch eine dynamische Reduzierung der Ausgabequalität des Ausgangssignals den Empfang und die Wiedergabe nahezu störungsfrei fortzuführen. Ein Überschuss an verfügbaren Ressourcen führt hingegen zu einer dynamischen Erhöhung der Ausgabequalität.

Die Untersuchung und Entwicklung eines realistischen Konzepts für einen solchen Empfänger zur Ausführung auf einem Desktop-Computer bildet den Schwerpunkt dieser Arbeit. Dabei soll die Theorie durch entsprechende praktische Messungen und daraus resultierenden Erkenntnissen validiert werden. Deshalb wurden in dieser Arbeit drei Empfängertypen (FM-Stereo inklusive RDS, ATV-PAL-B/G und DVB-T) in Simulink®, teilweise im GNURadio sowie auf dem neuen Konzept basierenden und eigens implementierten PhiloRadio modelliert und untersucht. Die Ergebnisse zeigen, ob und inwieweit eine Ausführung der genannten Empfänger unter Echtzeitanforderungen auf einem beliebigen Desktop-Computer überhaupt möglich ist. Die zu diesem Ziel führenden Werkzeuge und Bedingungen werden erläutert und am Ende der Arbeit entsprechend zusammengefasst und diskutiert.

Neben der Entwicklung eines echtzeitfähigen softwaredefinierten Rundfunkempfängers für Desktop-Computer soll diese Arbeit aber auch als eine Übersicht über den derzeitigen Stand der Technik im Bereich des *Software Radio* allgemein dienen.

Abstract

A real-time software defined broadcasting receiver combines a general purpose programmable hardware architecture with software-based digital signal processing functions instead of hard-wired functions. Real-time capability can be achieved with a complete new application concept based on a new framework using so-called scalable modules. The new concept considers that every existing and future developed broadcasting service can be described by an xml-file. With such a description every broadcasting service could be executed immediately. Even complex receiver types running on underperforming platforms can react on local or global environment influences with the assistance of scalable modules. Global environment influences are poor receiving conditions caused by bad weather, massive coverage of buildings or transients. Local environment influences are fluctuating resources which have to be compensated by a real-time software defined receiver. These resources are computing power of one or all available processors and memory. If one of these resources were missing a non-adaptable real-time broadcasting receiver would not work without conflicts. A real-time software defined radio receiver will adapt dynamically its output quality dependent on available resources. Thus an interruption free reception and reproduction will be possible. If more resources are available output quality will be increased.

The design and development of a realistic concept for such a receiver running on desktop computers is the focus of this study. The theory behind the solution will be verified by practical measurements and the derived knowledge. To get this knowledge three receiver types (FM-Stereo including RDS-Decoding, ATV PAL-B/G and DTV DVB-T) were implemented and evaluated within the proposed new framework called PhiloRadio. In addition, all receivers were modeled inside Simulink® and some parts inside GNU Radio. The results show for each of them whether real-time execution on desktop computers is possible or not. All necessary tools and conditions will be explained and at the end of this thesis summarized and discussed.

To show the development of real-time software defined broadcasting receivers for desktop computers is only one intention of this thesis. It should also represent the state of the art within the scope of software radios.

Symbole

δ	-	Örtliche Auflösungsgrenze des Auges in Winkelminuten
$f_{Bild_{min}}$	-	Untere Bildfrequenz
$f_{Bild_{max}}$	-	Obere Bildfrequenz
f_B	-	Vollbildfrequenz
f_{BAS}	-	Theoretische Bandbreite des analogen BAS-Signals
ω_0	-	Kreisfrequenz
f_H	-	Horizontale Zeilenfrequenz
f_V	-	Vertikale Teilbildfrequenz
f_a	-	Abtastfrequenz
f_g	-	Grenzfrequenz
f_{Nyq}	-	Nyquistfrequenz
τ	-	Verzögerungszeit
φ	-	Phasenwinkel
η	-	Modulationsindex
Δf	-	Frequenzhub
f_{ZF}	-	Zwischenfrequenz
f_{FT}	-	Farbträgerfrequenz
f_{BT}	-	Bildträgerfrequenz
f_{TT}	-	Tonträgerfrequenz
f_{PI}	-	Pilotton bei Stereo-Audio-Übertragung
f_O	-	Oszillatorfrequenz
$H_c(f)$	-	Übertragungsfunktion Chrominanzfilter für SECAM
$H(f)$	-	Allgemeine Übertragungsfunktion
ε	-	Bandbreiteneffizienz
$v^{Bit_{Brutto}}$	-	Bruttodatenrate
$v^{Bit_{Netto}}$	-	Nettodatenrate
f_E	-	Abstimmfrequenz
f_{Offset}	-	Frequenzoffset zwischen Mittenfrequenz und Abstimmfrequenz
f_M	-	Mittenfrequenz
A	-	Farbsättigung
a	-	Farbton

$a(t)$	-	Diracimpulsfolge
b	-	Bildbreite
B	-	Blauanteil oder Bandbreite
D	-	Dezimationsfaktor
D_B, D_R	-	SECAM-Farbdifferenzsignale
E	-	Anzahl der Nullen in einem Block
F	-	Farbartvektor
G	-	Grünanteil
h	-	Bildhöhe
I	-	Interpolationsfaktor oder Interleavingtiefe
I, Q	-	NTSC Farbdifferenzsignale oder Quadraturkomponenten
k	-	Kell-Faktor
K	-	Anzahl der Filterkoeffizienten
L	-	Linker Tonkanal
M	-	FFT-Blockgröße
N	-	Anzahl der Abtastwerte
n	-	Abtastwert n
O	-	Überlappungsbereich
p	-	Kantenlänge eines Pixels
P	-	Pixel pro Zeile
P_{Bild}	-	Pixelanzahl pro Vollbild
$P/^\circ$	-	Perioden pro Grad
R	-	Rotanteil oder Rechter Tonkanal
r	-	Roll-Off-Faktor
R_S	-	Übertragungsbandbreite
R_{Symbol}	-	Symbolrate
S	-	Segmentgröße
$S(f)$	-	Fouriertransformierte des Zeitsignals
$s(t)$	-	Zeitsignal
t	-	Zeit einer bestimmten Operation in Sekunden
T	-	Zeit eines bestimmten Softwaremoduls in Sekunden
T_0	-	Periodendauer
T_G	-	Guardintervall-Dauer
U, V	-	reduzierte Farbdifferenzsignale
$U_{Filter}(t)$	-	Spannung am Ausgang eines Filters
$U_{Input}(t)$	-	Eingangssignal
$U_{Output}(t)$	-	Ausgangssignal

$U_{PD}(t)$	-	Spannung am Ausgang eines Phasendetektors
$x(n)$	-	Eingangssignalfolge
Y	-	Luminanz/Leuchtdichte
$y(n)$	-	Ausgangssignalfolge
z	-	Zeilenzahl

	-	Addition		-	Multiplikation
	-	Division		-	Interpolationsfilter
	-	Subtraktion		-	Integrator
	-	EXOR		-	Argument
	-	Mischer		-	Konjugiert Komplexe Zahl
	-	Oscillator		-	Frequenzteiler
	-	Allgemeines Filter		-	Look-Up-Table
	-	Hochpassfilter		-	Einweggleichrichter
	-	Tiefpassfilter		-	A/D-Umsetzer
	-	Bandpassfilter		-	33° Phasenverschiebung
	-	Bandstopppfilter		-	Frequenzvervielfacher
	-	Preemphasis		-	Regelbare Phasenverschiebung
	-	Deemphasis		-	Verzögerungsglied
	-	90° Phasenverschiebung		-	Filterfunktion
	-	Amplituden-(De)modulator		-	Automatic Gain Control



- Frequenz-(De)modulator



- Summierer



- Phasen-(De)modulator



- Dezimationsfilter



- Differenzierer



- Wurzel

Abkürzungen

ABSOC	-	Advanced Broadcasting Systems of Canada
ACML	-	AMDs Core Math Library
ADC	-	Analog to Digital Converter
ADU	-	Analog zu Digital Umsetzer
AFE	-	Analog Front End
AGC	-	Automatic Gain Control
AM	-	Amplituden Modulation
AMD	-	Advanced Micro Devices
API	-	Application Programming Interface
ARI	-	Auto Fahrer Rundfunk Information
ASIC	-	Application Specific Integrated Circuit
ASIP	-	Application Specific Instruction Set Processor
ATSC	-	Advanced Television Systems Committee
ATV	-	Analog Television
AVR	-	Automatic Voltage Control
BAS	-	Bild-, Austast-, Synchronsignal
BAT	-	Bouquet Association Table
BB	-	Basisband
BCH	-	Bose, Ray-Chaudhuri und Hocquenghem
BCJR	-	Bahl, Cocke, Jelinek und Raviv
BE	-	Bildelement
BER	-	Bit Error Rate
BFA	-	Bidirektionale Fano Algorithmus
BGL	-	Boost Graph Library
BMA	-	Berlekamp-Massey-Algorithmus
BNC	-	Bayonet Neill Concelman
BPF	-	Bandpass Filter
BPSK	-	Binary Phase Shift Keying
BTU	-	Brandenburgische Technische Universität
CAT	-	Conditional Access Table
CCIR	-	Comité Consultativ International des Radiocommunications
CDMA	-	Code Division Multiple Access
CFO	-	Carrier Frequency Offset
CIC	-	Cascaded Integrated Comb

COFDM	-	Coded Orthogonal Frequency Division Multiplex
CORBA	-	Common Object Request Broker Architecture
CORDIC	-	Coordinate Rotation Digital Computer
CP	-	Continual Pilot
CPFIFO	-	Chasing Pointer First In First Out
CPLD	-	Complex Programmable Logic Device
CPO	-	Carrier Phase Offset
CPU	-	Central Processing Unit
CUDA	-	Compute Unified Device Architecture
CW	-	Continuous Wave
DA	-	Data Aided
DAB	-	Digital Audio Broadcast
DAG	-	Directed Acyclic Graph
DARPA	-	Defense Advanced Research Projects Agency
dB	-	Dezibel
DBPSK	-	Differential Binary Phase Shift Keying
DC	-	Direct Current
DCT	-	Discrete Cosinus Transform
DDC	-	Digital Down Converter
DDR	-	Double Data Rate
DFE	-	Digital Front End
DFT	-	Discrete Fourier Transformation
DIN	-	Deutsches Institut für Normung
DLL	-	Dynamic Link Library
DMA	-	Direct Memory Access
DQPSK	-	Differenzielle Quadratur Phase Shift Keying
DRM	-	Direct Memory Access
DSB	-	Double Sideband
DSP	-	Digital Signal Prozessor
DTV	-	Digital Television
DVB-C	-	Digital Video Broadcasting Cable
DVB-S	-	Digital Video Broadcasting Satellite
DVB-T	-	Digital Video Broadcasting Terrestrial
EBU	-	European Broadcasting Union
EDF	-	Earliest Deadline First
EDTV	-	Extended Definition Television
EEA	-	Erweiterter Euklidischer Algorithmus

EHF	-	Extremely High Frequencies
EIT	-	Event Information Table
EME	-	Electromagnetic Energy
EN	-	Euro Norm
ESB	-	Einseitenband
FAR	-	Flexible Architecture Radio
FBAS	-	Farb-, Bild-, Austast-, Synchronsignal
FCC	-	Federal communications Commission
FDA	-	Filter Design and Analysis
FEC	-	Forward Error Correction
FFmpeg	-	Fast Forward MPEG
FFT	-	Fast Fourier Transform
FFTW	-	Fastest Fourier Transform in the West
FIFO	-	First In First Out
FIR	-	Finite Impulse Response
FLOPS	-	Floatingpoint Operation Per Second
FM	-	Frequenz Modulation
FPA	-	Filed Programmable Analog Array
FPGA	-	Field Programmable Gate Array
FPU	-	Floating Point Unit
FU	-	Function Unit
GFLOPS	-	Giga Floating Point Operations Per Second
GNU	-	<u>G</u> NU ist <u>N</u> ot <u>U</u> nix
GP	-	General Purpose
GPGPU	-	General Purpose Graphics Processing Unit
GPL	-	General Public License
GPP	-	General Purpose Processor
GPS	-	Global Positioning System
GRC	-	GNU Radio Companion
GS/s	-	Giga Sample Per Second
GSA	-	Generalisierte Stack Algorithmus
GSM	-	Global System For Mobil Communications
GTL	-	Graph Template Library
GUI	-	Graphical User Interface
HD	-	High Definition
HDR	-	Hardware Defined Radio
HDTV	-	High Definition Television

HF	-	High Frequency
HL	-	High Level
HP	-	High Profile
HPF	-	Highpass Filter
HTK	-	Hidden Markov Model Toolkit
IC	-	Integrated Circuit
ID	-	Identifier
IDCT	-	Inverse Discrete Cosinus Transform
IDE	-	Integrated Development Environment
IDFT	-	Inverse Diskrete Fourier Transformation
IEC	-	International Electrotechnical Commission
IEEE	-	Institute of Electrical and Electronics Engineers
IF	-	Intermediate Frequency – (Zwischenfrequenz)
IFA	-	Internationale Funkausstellung
IFFT	-	Inverse Fast Fourier Transform
IIR	-	Infinite Impulse Response
IP	-	Internet Protocoll
IPP	-	Integrated Performance Primitives
IQ	-	Inphase Quatratur
IRQ	-	Interrupt Request
IRQL	-	Interrupt Request Level
ISO	-	International Standards Organization
ISR	-	Ideal Software Radio
ITU	-	International Telecommunication Union
JK	-	Jack Kilby bzw. Jump Kill
JTRS	-	Joint Tactical Radio System Program Office
KSPS	-	Kilo Samples Per Second
LA	-	Line Averaging
LAN	-	Local Area Network
LDTV	-	Low Definition Television
LGPL	-	GNU Lesser General Public License
LL	-	Low Level
LLC	-	Limited Liability Company
LNA	-	Low Noise Amplifier
LO	-	Local Oscillator
LPF	-	Lowpass Filter
LPT	-	Line Print Terminal

LSB	-	Lower Sideband
LTI	-	Linear Time Invariant
LUT	-	Look Up Table
LW	-	Langwelle
MAC	-	Media Access Control
MAP	-	Maximum Aposteriori Probability
MB	-	Mega Byte
MCPS	-	Mega Cycles Per Second
MFLOPS	-	Mega Floating Point Operations Per Second
MIMD	-	Multiple Instruction Multiple Data
MIPS	-	Mega Instructions Per Second
MISD	-	Multiple Instruction Single Data
MIT	-	Messachuetts Institute Of Technology
MKL	-	Math Kernel Library
ML	-	Main Level
MMITS	-	Modular Multifunction Information Transfer System
Mod	-	Modular
MOPS	-	Mega Operations Per Second
MP	-	Main Profile
MPEG	-	Motion Pictures Experts Group
MPI	-	Message Passing Interface
MPX	-	Multiplex
MS	-	Microsoft
MS/s	-	Mega Samples Per Second
MSRC	-	Modular Software Programmable Radio Consortium
MST	-	Multi-Standard-Terminal
MUSE	-	Multiple Subsampling Encoding
MW	-	Mittelwelle
NCA	-	Numerical Controlled Amplifier
NCO	-	Numeric Controlled Oscillator
NDA	-	Non Data Aided
NF	-	Noise Figure
NICAM	-	Near Instantaneous Companded Audio Multiplex
NIT	-	Network Information Table
NOP	-	No Operation
NP	-	Nichtdeterministisch Polynomiell
NTSC	-	National Television System Committee

OEF	-	Ossie Eclipse Feature
OFDM	-	Orthogonal Frequency Division Multiplex
OIRT	-	Organisation Internationale de Radiodiffusion et de Télévision
OS	-	Operating System
OSI	-	Open Systems Interconnections
OSSIE	-	Open-Source SCA Implementation
PA	-	Power Amplifier
Pac	-	Parameter Controlled
PAL	-	Phase Alternation Line
PAT	-	Program Association Table
PC	-	Personal Computer
PCI	-	Peripheral Component Interconnect
PCR	-	Program Clock Reference
PD	-	Phasendetektor
PES	-	Packetized Elementary Streams
PFD	-	Phasen Frequenz Detektor
PI	-	Program Identification
PID	-	Program Identifier
PLL	-	Phase Locked Loop
PM	-	Phase Modulation
PMT	-	Program Map Table
PPL	-	Parallel Pattern Library
PPU	-	Physics Processing Unit
PRBS	-	Pseudo Random Binary Sequence
PSA	-	Pilot Symbol Assisted
PSI	-	Program Specific Information
QAM	-	Quadratur-Amplituden-Modulation
QM	-	Quality Manager
QoS	-	Quality Of Service
QPSK	-	Quadratur Phase Shift Keying
RAID	-	Redundant Array of Independent Disks
RBDS	-	Radio Broadcast Data System
RC	-	Resistor Capacitor
RCB	-	Radio Control Board
RDS	-	Radio Data System
RDTSC	-	Read-Time-Stamp-Counter-Register
RF	-	Radio Frequency

RGB	-	Red Green Blue
RL	-	Resource Level
RM	-	Resource Manager
RMS	-	Root Mean Square
RS	-	Reed Solomon
RSB	-	Restseitenband
RSP	-	Ring Segment Puffer
RST	-	Running Status Table
RTTY	-	Radioteletype
RX	-	Receive
SA	-	Skalierbarer Algorithmus
SATA	-	Serial Advanced Technology Attachment
SAW	-	Surface Acoustic Wave
SC	-	Suppressed Carrier
SCA	-	Software Communication Architecture
SCR	-	Software Controlled Radio
SD	-	Secure Digital
SDK	-	Software Development Kit
SDL	-	Simple DirectMedia Layer
SDR	-	Software Defined Radio
SDRPHY	-	Software Defined Radio Physical Layer
SDT	-	Service Description Table
SDTV	-	Standard Definition Television
SECAM	-	Séquentielle à mémoire
SF	-	Spiegelfrequenz
SFML	-	Simple and Fast Multimedia Library
SFO	-	Sample Frequency Offset
SHF	-	Super High Frequencies
SI	-	Service Information
SIMD	-	Single Instruction Multiple Data
SINAD	-	Signal To Interference Ratio Including Noise And Distortion
SISD	-	Single Instruction Single Data
SNR	-	Signal To Noise Ratio
SNRP	-	SNR Scalable Profile
SOS	-	Second Order Section
SOVA	-	Soft Output Viterbi Algorithmus
SP	-	Simple Profile or Scattered Pilot

SPLD	-	Simple Programmable Logic Devices
SPLL	-	Software Phase Locked Loop
SRC	-	Samplerate Converter
SSB	-	Single Sideband
SSP	-	Spatial Scalable Profile
STL	-	Standard Template Library
STO	-	Symbol Timing Offset
SVA	-	Skalierbare Video Algorithmen
SVHS	-	Super Video Home System
TBB	-	Thread Building Blocks
TCP	-	Transport Control Protocoll
TDT	-	Time & Date Table
TPF	-	Tiefpass Filter
TPS	-	Transmission Parameter Signalling Pilot
TS	-	Transport-Strom
TSR	-	Transport-Strom-Rahmen
TTF	-	TrueType-Fonts
TV	-	Television
TVRX	-	Television Receiver
TVRX2	-	Television Receiver Version 2
TX	-	Transmit
UDP	-	Uni Direction Protocoll
UFA	-	Unidirektionale Fano Algorithmus
UHD	-	Universal Software Radio Peripheral Hardware Driver
UHF	-	Ultra High Frequency
UMS	-	User Mode Scheduling
UMTS	-	Universal Mobile Telecommunications System
USA	-	United States of America
USB	-	Upper Sideband oder Universal Serial Bus
USR	-	Ultimate Software Radio
USRP	-	Universal Software Radio Peripheral
VBS	-	Video Baseband Signal
VCA	-	Voltage Controlled Amplifier
VHF	-	Very High Frequency
VLIW	-	Very Long Instruction Word
VR	-	Volksrepublik
VSF	-	Vestigial Sideband

WBX	-	Wideband Transceiver
XML	-	Extensible Markup Language
ZF	-	Zwischenfrequenz
ZJ	-	Zigangirov-Jelinek
ZSB	-	Zweiseitenband

Inhaltsverzeichnis

DANKESWORT.....	V
KURZFASSUNG.....	VII
ABSTRACT	IX
SYMBOLS	XI
ABKÜRZUNGEN	XV
INHALTSVERZEICHNIS	XXV
1 EINLEITUNG	1
1.1 MOTIVATION	1
1.2 STAND DER FORSCHUNG	5
1.3 GLIEDERUNG DER ARBEIT	6
2 SOFTWARE DEFINED RADIO.....	9
2.1 EINLEITUNG	9
2.2 ALLGEMEINE DEFINITIONEN	10
2.3 IM DETAIL	12
2.4 TOPOLOGIE.....	23
2.5 HARDWARE	26
2.5.1 Allgemein	26
2.5.2 PhiloRadio-Board	42
2.5.3 USRP2-Board	43
2.5.4 Sonstige	44
2.6 SOFTWARE	46
2.7 ZUSAMMENFASSUNG.....	50
3 SYSTEM ARCHITEKTUREN.....	55
3.1 EINLEITUNG	55
3.2 ECHTZEITSYSTEME.....	55
3.2.1 Graphen	60
3.2.2 Ablaufpläne	62
3.2.3 Pufferspeicher	70
3.2.4 Laufzeit-Messung	71

3.3	SKALIERBARE SYSTEME	74
3.3.1	<i>Hardwarebasiert</i>	75
3.3.2	<i>Softwarebasiert</i>	82
3.4	EXISTIERENDE ARCHITEKTUREN	90
3.4.1	<i>Matlab®</i>	90
3.4.2	<i>GNU Radio</i>	92
3.4.3	<i>Sora</i>	94
3.4.4	<i>Sonstige SDRs</i>	96
3.5	ZUSAMMENFASSUNG	97
4	DAS PHILORADIO	99
4.1	EINLEITUNG	99
4.2	PROGRAMMGERÜST	100
4.3	IMPLEMENTIERUNG	123
4.4	ZUSAMMENFASSUNG	128
5	ANALOG UND DIGITALE RUNDUNKDIENSTE	131
5.1	EINLEITUNG	131
5.2	FM-STEREO MIT RDS-DECODER	133
5.3	ANALOGES FERNSEHEN IN PAL-B/G	138
5.4	DIGITALES FERNSEHEN MIT DVB-T	149
5.5	ZUSAMMENFASSUNG	156
6	NICHT SKALIERBARE SOFTWARE MODULE	157
6.1	EINLEITUNG	159
6.2	QUELLEN UND SENKEN	163
6.3	GRUNDOPERATIONEN	165
6.4	ANALOG MODULE	167
6.4.1	<i>Verzögerer</i>	167
6.4.2	<i>Sättiger</i>	168
6.4.3	<i>Normalisierer</i>	169
6.4.4	<i>ATV Dematrizierer</i>	171
6.5	FOURIER TRANSFORMATOR	174
6.6	DECODER	176
6.6.1	<i>Radio-Data-System-Decoder</i>	176
6.6.2	<i>Transportstrom-Demultiplexer</i>	178
6.6.3	<i>Sync-Byte-Detektor</i>	182

6.7	DIGITALE MODULE	184
6.7.1	<i>Symbol-Demapper</i>	184
6.7.2	<i>Symbol-Deinterleaver</i>	187
6.7.3	<i>Bit-Deinterleaver</i>	189
6.7.4	<i>Faltungs-Deinterleaver</i>	190
6.7.5	<i>Energieverwischung</i>	192
6.8	HILFSMODULE	193
6.9	ZUSAMMENFASSUNG	196
7	SKALIERBARE SOFTWARE MODULE	201
7.1	EINLEITUNG	201
7.2	QUELLEN UND SENKEN	202
7.3	DIGITALE FILTER	203
7.3.1	<i>Infinite Impulse Response Filter</i>	203
7.3.2	<i>Finite Impulse Response Filter</i>	207
7.3.3	<i>Abtastatenkonverter</i>	216
7.3.4	<i>Kammfilter</i>	225
7.3.5	<i>Translationsfilter</i>	233
7.4	DEMODULATOREN	236
7.4.1	<i>Mischer</i>	236
7.4.2	<i>Frequenzdemodulator</i>	242
7.4.3	<i>Amplitudendemodulator</i>	248
7.4.4	<i>I/Q-Demodulator</i>	251
7.5	SYNCHRONISATOREN	253
7.5.1	<i>Verstärkungsregelung</i>	253
7.5.2	<i>COFDM Synchronisator</i>	259
7.5.3	<i>Phasenregelschleife</i>	272
7.5.4	<i>H/V-Pulse-Synchronisator</i>	280
7.6	DECODER	282
7.6.1	<i>PAL-Decoder</i>	282
7.6.2	<i>TPS-Decoder</i>	291
7.6.3	<i>Viterbi-Decoder</i>	292
7.6.4	<i>Reed-Solomon-Decoder</i>	297
7.6.5	<i>MPEG-2-Decoder</i>	299
7.7	FREQUENZTEILER	301
7.8	KANALSCHÄTZER	302
7.9	DEINTERLACER	304

7.10	ZUSAMMENFASSUNG	305
8	SKALIERBARE SOFTWAREDEFINIERT FUNKEMPFÄNGER	309
8.1	EINLEITUNG	309
8.2	MESSAUFBAU	310
8.3	FM-STEREO MIT RDS-DECODER	312
8.3.1	<i>Parameter</i>	313
8.3.2	<i>Messergebnisse</i>	318
8.3.3	<i>Aufbau</i>	324
8.4	ATV IN PAL-B/G	326
8.4.1	<i>Parameter</i>	326
8.4.2	<i>Messergebnisse</i>	332
8.4.3	<i>Aufbau</i>	336
8.5	DTV MIT DVB-T	338
8.5.1	<i>Parameter</i>	338
8.5.2	<i>Messergebnisse</i>	344
8.5.3	<i>Aufbau</i>	348
8.6	AUSWERTUNG	349
9	BENUTZERSCHNITTSTELLE DES PHILORADIOS	357
10	ZUSAMMENFASSUNG	361
11	AUSBLICK	365
	ANHANG	368
(A)	ÜBERSICHT VON SDR-Projekten	369
(B)	PARAMETER DER REALISIERTEN MODULE	376
(C)	BEISPIELRECHNUNG EINER PARTITIONIERUNG	397
(D)	EMPFÄNGER UNTER SIMULINK	401
(Da)	<i>Skalierbarer FM-Stereo-Empfänger inkl. RDS-Decoder</i>	401
(Db)	<i>Skalierbarer ATV PAL-Empfänger</i>	405
(Dc)	<i>Skalierbarer DVB-T-Empfänger</i>	409
(E)	WEITERE MESSUNGEN	411
	ABBILDUNGSVERZEICHNIS	419
	TABELLENVERZEICHNIS	427
	LITERATURVERZEICHNIS	431
	STICHWORTVERZEICHNIS	453