

Increasing the Spectral Efficiency of Continuous Phase Modulation Applied to Digital Microwave Radio: A Resource Efficient FPGA Receiver Implementation

A thesis presented in partial fulfilment of the
requirements for the degree of

**Master of Engineering
in
Electronics and Computer Systems
Engineering**

at Massey University, Palmerston North,
New Zealand

Andrew B. Bridger
B.E. (Hons 1)

November 2, 2009

Abstract

In modern point to point microwave radio systems used to backhaul cellular voice and data traffic, quadrature amplitude modulation (QAM) is the norm. These systems require a highly linear power amplifier which is expensive and has relatively low power efficiency. Recently, continuous phase modulation (CPM) has been deployed in this market. The CPM transmitted waveform has a constant envelope and so a non-linear RF power amplifier can be used. This significantly reduces cost and improves power efficiency.

Two important disadvantages of CPM are receiver complexity and inferior spectral efficiency compared to QAM. This thesis demonstrates a 50% spectral efficiency improvement over an existing CPM configuration without loss of detection efficiency. This is achieved by moving to coherent demodulation and extending the duration of the CPM phase pulse to 3 symbol periods.

This new CPM configuration of $h=1/4, M=4, L=3$, is evaluated against ETSI requirements for a 28 MHz channel carrying 24 E1 circuits. Simulation of the receiver floating point model demonstrates all requirements are met. The detection efficiency requirement is exceeded by 4.7 dB. Carrier recovery, phase and timing synchronisation are assumed to be ideal.

The 50% increased symbol rate, coherent reception and a longer smoother phase pulse, conspire to increase receiver complexity substantially. The Viterbi algorithm is used to perform maximum-likelihood detection resulting in a 128 state trellis. This application has a stringent cost requirement that limits the implementation target to a Field Programmable Gate Array (FPGA) costing less than US\$30. To demonstrate this demanding cost target is met, the two most computationally expensive receiver functions, the branch metric unit and path metric processing unit, are implemented in VHDL and targeted to a Xilinx Spartan 3A-DSP 1800 FPGA. The implementation uses 67% of the available logic resources, thus meeting the cost requirement.

The branch metric unit is implemented using a distributed arithmetic technique that performs the equivalent of 27.6 giga-multiplies/s, consuming only 23% of the available FPGA logic cells. This is very efficient compared to a conventional approach using all the FPGA's embedded multipliers which combined can only achieve 21 giga-multiplies/s.

The Viterbi path metric processing unit is implemented using a more conventional state-parallel architecture. To reduce state metric routing complexity, states are grouped into radix-4 units comprising dual add-compare-select (ACS) units. By utilising a spare cycle in the deep ACS pipeline, each ACS unit processes two output state metrics, thus halving the number of ACS units required. This implementation uses 44% of the available FPGA resources and meets timing at 204.5 MHz, exceeding the throughput requirement of 54 Mbit/s.

Copyright is owned by the Author of the thesis. Permission is given for a copy to be downloaded by an individual for the purpose of research and private study only. The thesis may not be reproduced elsewhere without the permission of the Author.

Acknowledgements

I would like to acknowledge and thank those who have helped me during my research and made it all possible.

This thesis was completed under a Technology Industry Fellowship (TIF) Education scholarship in conjunction with Harris Stratex Networks (NZ) LTD and Massey University. This support from FRST is much appreciated.

A big thanks to Dr Philip Secker, my project mentor from Harris Stratex Networks, for his role in kick starting the project and securing funding from FRST and Harris Stratex Networks. Also for taking the time out from a busy schedule to offer guidance, read my long emails and answer my many questions throughout the year, and provide feedback on this thesis.

I would like to thank Harris Stratex Networks for their commitment to the project in the form of Philip's time and a stipend contribution.

Many thanks to Dr Xiang Gui, my academic supervisor in Palmerston North, for his time spent listening and offering guidance during our weekly progress meetings, for ensuring I had access to the university's resources I needed, and for taking the time to review my thesis.

I would also like to thank Dr Edmund Lai, my co-supervisor based in Wellington, for his wise words, excellent thesis writing resources on his website, and for taking the time to provide feedback on my thesis.

Colin Plaw and Patrick Rynhart helped me with VPN access to the university labs and ensured I had access to a source code repository. Managing all the material created throughout the year would have been much much harder without it, thank you.

To my parents, Mum, Dad, Ma and Baba, thank you for all your encouragement and support, and for all the dinners that have been cooked for me over the last 6 months while I have been writing up this thesis. Also, thank you Mum for proof reading the whole thesis.

Finally, a very special thank you to my lovely wife Mandira, for convincing me to pursue my FPGA and digital signal processing passion in the form of a Masters in Engineering. Thank you for your constant encouragement and understanding during the thesis writeup phase.

Contents

Abstract	iii
Acknowledgements	v
List of Figures	xii
List of Tables	xiii
Glossary	xvi
1 Introduction	1
1.1 Introduction	1
1.2 Scope	2
1.3 Summary of Thesis Contributions	4
1.4 Overview	4
2 Background and Related Work	7
2.1 Introduction	7
2.2 Notation	7
2.3 Continuous Phase Modulation Signal Model	7
2.4 Maximum Likelihood Receiver	10
2.4.1 Rational h	11
2.4.2 Viterbi Trellis Decode	12
2.5 Viterbi Decoder Architecture	12
2.6 Literature Review	13
2.6.1 CPM Receiver Implementations	13
2.6.1.1 Viterbi Path Metric Processing	13
2.6.1.2 Viterbi Path Metric Normalisation	14
2.6.2 CPM Configurations and their Energy/Bandwidth Consumption	15
2.6.3 Complexity Reduction	15
2.6.4 Literature Review Summary	16
3 CPM Parameter Selection	17
3.1 Introduction	17
3.2 CPM Candidates	18
3.2.1 Choice of Symbol Alphabet Size (M) and Phase Pulse Shape	19
3.2.2 Modulation Index (h) and Phase Pulse Duration (L) Candidates	21

3.3	ETSI Requirements	23
3.3.1	Specific Application	23
3.3.2	Bandwidth: Transmit Power Spectral Density (PSD) . . .	24
3.3.3	Detection Efficiency: Bit Error Rate as a function of Re- ceive Signal Level	24
3.3.4	Interference Rejection	24
3.4	Simulation Results	25
3.4.1	Simulation System Model	25
3.4.2	Bandwidth: Transmit Power Spectral Density (PSD) . . .	25
3.4.3	Detection Efficiency and Interference Rejection Performance: h=1/4, L=3	27
3.4.3.1	Detection Efficiency: Bit Error Rate as a func- tion of SNR	27
3.4.3.2	1st Adjacent Channel Interference	29
3.4.3.3	Co-channel Interference	30
3.4.4	Detection Efficiency and Interference Rejection Performance: h=1/5, L=2	30
3.5	Conclusion	32
4	Fixed Point Modelling	35
4.1	Introduction	35
4.2	Implementation Target	35
4.3	Sampling Rate	37
4.4	Fixed Point Modelling	38
4.4.1	Floating Point vs Fixed Point Numeric Representation .	39
4.4.2	Quadrature and In-phase Received Signal Word-length .	39
4.4.3	Branch Metric Filter Bank Coefficient Word-length	39
4.4.4	Branch Metric Word-length	41
4.4.5	State Metric Normalisation	42
4.5	Survivor Path History	43
4.6	Conclusion	45
5	Branch Metric Implementation	47
5.1	Introduction	47
5.2	Computation Complexity	48
5.3	CPM and Distributed Arithmetic	50
5.3.1	Phase State Symmetry	51
5.4	FPGA Implementation	52
5.4.1	Throughput Requirements	52
5.4.2	Efficient Mapping of a DA Filter into FPGA Hardware .	53
5.4.2.1	DALUT	53
5.4.2.2	Scaling Accumulator	54
5.4.2.3	FPGA Resource Use Summary	54
5.4.2.4	Additional Resource Use Required to Meet Tim- ing	54
5.4.3	Implementation Results	55
5.4.4	Functional Verification	56
5.5	DA vs Embedded Multipliers	56
5.6	Conclusion	57

6 Path Metric Implementation	59
6.1 Introduction	59
6.2 Background	60
6.3 Proposed Solution	61
6.3.1 State-Parallel Radix-4 Decomposition	61
6.3.2 Add-Compare-Select Unit	64
6.3.2.1 Resource Use Estimate	65
6.4 Implementation Results	66
6.4.1 Functional Verification	67
6.5 Conclusion	67
7 Conclusions and Future Work	69
Appendix	72
A VHDL Implementation Functional Verification	75
A.1 VHDL Functional Verification Architecture	75
B Receive Signal Level to SNR Conversion	77
C Baseband I/Q Modulator Derivation	79
D VHDL Source Code	81
D.1 Branch Metric Filter Bank	81
D.1.1 Synthesis Top Level	81
D.1.2 Top Level	81
D.1.3 4-Tap Distributed Arithmetic Filter	87
D.1.4 Filter Coefficients	92
D.1.5 Testbench	92
D.1.6 Test Vectors Package	96
D.2 Viterbi Trellis Path Metrics	96
D.2.1 Synthesis Top Level	96
D.2.2 Top Level	96
D.2.3 Radix-4 Add-Compare-Select Unit	105
D.2.4 Viterbi Trellis Package	121
D.2.5 Testbench	121
D.2.6 Test Vectors Package	122
D.3 Branch Metrics Filter Bank and Viterbi Trellis Path Metrics	122
D.3.1 Synthesis Top Level	122
D.3.2 Top Level	122
D.3.3 Testbench	122
D.4 Placed Primitives Modules	122
D.4.1 Adder with Subtract and Clear Controls	122
D.4.2 Adder with Subtract and 2 Input Operand Mux	122
D.4.3 16 Deep ROM using Distributed Ram	122
D.4.4 2 Input Mux	122
D.4.5 Shift Register	122
D.4.6 Register	122
D.4.7 Relative Location Constraint(RLOC) Helper Package	122
D.5 Sundry Packages	123
D.5.1 Key Project Constants	123

E	Matlab Source Code	125
E.1	Analytical CPM Code Performance	125
E.1.1	CPM Code Minimum Euclidean Distance Upper Bound	125
E.1.2	CPM Code Baseband Double Sided Bandwidth	125
E.2	Floating and Fixed Point M-file Models	125
E.3	VHDL Trellis Representation and Test Vector Generation	125
E.3.1	Export Matlab Data to VHDL Writer	126
E.3.2	VHDL Writer	126
E.4	Miscellaneous	126
E.4.1	Hekstras Method Bound Calculation	126
F	Implementation Results	127
F.1	Branch Metric Filter Bank	127
F.2	Viterbi Trellis Path Metrics	129
G	Software Tool Versions	133
G.1	High Level Modelling: Matlab and Simulink	133
G.2	FPGA Implementation: Xilinx ISE 10.1	133
G.3	VHDL Simulation: Modelsim	133
	Bibliography	135

List of Figures

1.1	CPM Receiver Functionality Studied in this Thesis	3
1.2	Modelling and Implementation at Complex Baseband	4
2.1	Raised Cosine (RC) and Rectangular (REC) Frequency Pulse and Phase Pulse	9
2.2	Standard Viterbi Decoder Implementation Architecture	12
3.1	Effect of Symbol Alphabet Size (M) on Detection Efficiency, L=1, Raised Cosine Phase Pulse	19
3.2	Effect of Phase Pulse Duration (L) on Detection Efficiency, M=4	20
3.3	Relative Detection Efficiency and Spectral Efficiency for several CPM Configurations	22
3.4	Simulation System Model	26
3.5	Simulated Transmit Power Spectral Density of Candidate CPM Configurations, Various (h, L), M=4, 27 Msymbols/s)	26
3.6	Zoomed in version of Figure 3.5	27
3.7	Simulated Bit Error Probability with and without Reed Solomon FEC, AWGN Channel, No ACI Reject Filter, h=1/4, L=3RC, 27 Msymbols/s	28
3.8	Simulated Bit Error Rate with Adjacent Channel Interference, h=1/4, L=3, M=4, 27 Msymbols/s	29
3.9	Simulated Bit Error Rate Demonstrating Effect of ACI Reject Filter and Adjacent Channel Interference, h=1/4, L=3, M=4, 27 Msymbols/s	30
3.10	Simulated Bit Error Rate with Co-Channel Interference, h=1/4, L=3, M=4, 27 Msymbols/s	31
3.11	Simulated Bit Error Rate with and without Adjacent Channel Interference, h=1/5, L=2, M=4, 27 Msymbols/s	32
4.1	Approximations to the maximum-likelihood receiver	36
4.2	Effect of Sampling Rate on Detection Efficiency	37
4.3	Effect of Quantised Received Signal Word-length on Receiver Detection Efficiency	40
4.4	Effect of Quantised Branch Metric Filter Bank Coefficient Word-length on Receiver Detection Efficiency	41
4.5	Effect of Branch Metric Word-length on Receiver Detection Efficiency	42
4.6	Effect of Survivor Path History Depth on Detection Efficiency	44

5.1	4-Tap Distributed Arithmetic FIR Filter Block Diagram	52
6.1	CPM Viterbi Detection	59
6.2	Add-Compare-Select Processing Required per State	61
6.3	128 State CPM Viterbi Detector Comprising 32 Radix-4 Add-Compare-Select Units	62
6.4	CPM Radix-4 Trellis ($h=1/4$, $L=3$, $M=4$)	63
6.5	Add-Compare-Select (ACS) Unit Detail and Pipeline	64
6.6	Radix-4 Unit Comprising Dual ACS Units	65
A.1	VHDL Implementation Test Architecture	76

List of Tables

2.1	Notation	7
3.1	Candidate Modulation Indices (h)	21
3.2	Candidate CPM Configurations	23
3.3	ETSI Co-Channel and 1st Adjacent Channel Interference Performance [1, Table D.7]	24
4.1	Comparison of DSP and FPGA Multiplication Capability	36
4.2	Detection Efficiency Degradation due to Branch Metric Quantisation	42
5.1	CPM Branch Metric Filter Bank Complexity	49
5.2	Distributed Arithmetic Filter Fmax Requirement	53
5.3	Estimated Branch Metric Filter Bank FPGA Resource Use	54
5.4	Branch Metric Filter Bank Implementation Results	55
6.1	Single Add-Compare-Select Unit FPGA Resource Use Estimate	66
6.2	Path Metric Processing Unit Implementation Results	67
B.1	ETSI Received Signal Level Converted to SNR and $\frac{E_b}{N_o}$	77
G.1	Matlab and Simulink Software Versions	133
G.2	Xilinx Synthesis and Implementation Software Versions	134

Glossary

ACI	Adjacent Channel Interference
ACS	Add-Compare-Select
ACSU	Add-Compare-Select Unit
ADC	Analog to Digital Converter
AGC	Automatic Gain Control
ASIC	Application Specific Integrated Circuit
ASSP	Application Specific Standard Product
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BMU	Branch Metric Unit
BRAM	Block Random Access Memory
CCI	Co-Channel Interference
CFO	Carrier Frequency Offset
CPFSK	Continuous Phase Frequency Shift Keying
CPM	Continuous Phase Modulation
DA	Distributed Arithmetic
DALUT	Distributed Arithmetic Look-Up Table
DSP	Digital Signal Processor
DUT	Device Under Test
ETSI	European Telecommunications Standards Institute
FEC	Forward Error Correction
FPGA	Field Programmable Gate Array
GMSK	Gaussian Minimum Shift Keying
IF	Intermediate Frequency
LC	Logic Cell
LUT	Look-Up Table
ML	Maximum-Likelihood
MLSD	Maximum-Likelihood Sequence Detector

MSK	Minimum Shift Keying
PAM	Pulse Amplitude Modulation
PSD	Power Spectral Density
QAM	Quadrature Amplitude Modulation
RC	Raised Cosine
REC	Rectangular
RF	Radio Frequency
RLOC	Relative Location
RSL	Received Signal Level
SER	Symbol Error Rate
SMU	Survivor Management Unit
SNR	Signal to Noise Ratio
SRAM	Static Random Access Memory
SRC	Spectrally Raised Cosine
STA	Static Timing Analysis
TFM	Tamed Frequency Modulation
UHF	Ultra-High Frequency
VHDL	VHSIC hardware description language
VHSIC	Very-High-Speed Integrated Circuit
VME	VERSA-module Europe