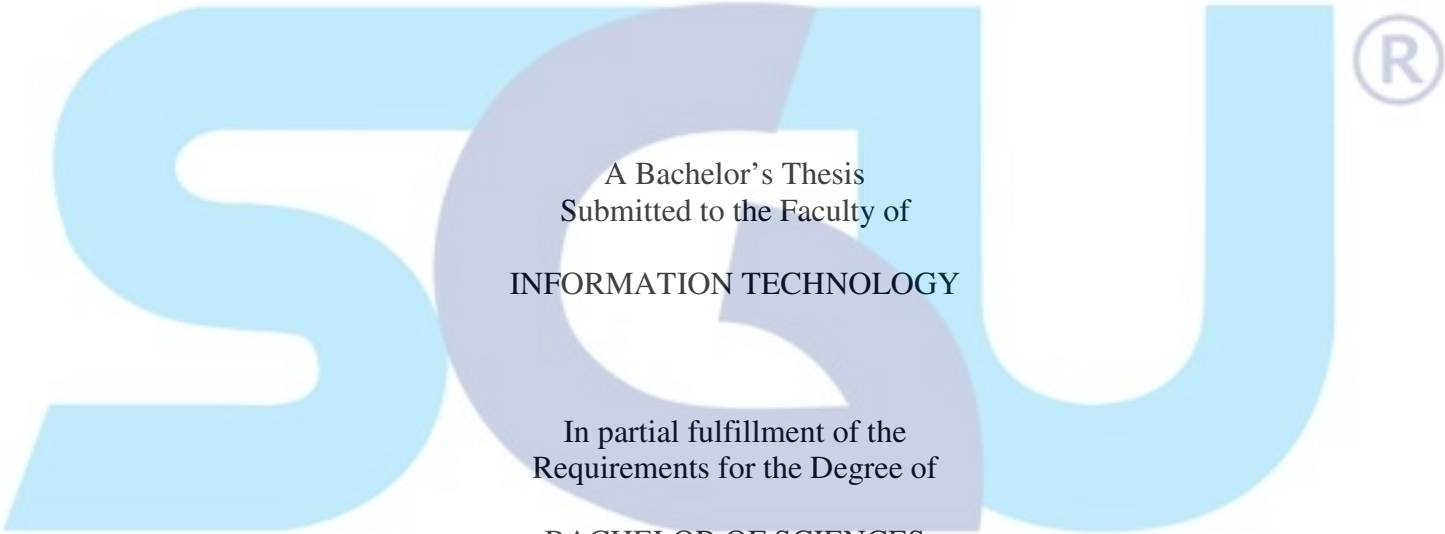


**WHITE-BOX AND BLACK-BOX SOFTWARE TESTING OF
CONNECTIONLESS NETWORK PROTOCOL (CLNP)
FUNCTIONS IN AERONAUTICAL TELECOMMUNICATION
NETWORK (ATN)**

By

Tjeuw Alvin Felix

The logo for Swiss German University (SGU) features the letters 'S', 'G', and 'U' in a large, light blue, stylized font. The 'G' is partially overlapping the 'S' and 'U'. A registered trademark symbol (®) is located to the right of the 'U'.

A Bachelor's Thesis
Submitted to the Faculty of
INFORMATION TECHNOLOGY

In partial fulfillment of the
Requirements for the Degree of

BACHELOR OF SCIENCES
WITH A MAJOR IN INFORMATION TECHNOLOGY

SWISS GERMAN UNIVERSITY

SWISS GERMAN UNIVERSITY
Campus German Centre
Bumi Serpong Damai – 15321
Island of Java, Indonesia
www.sgu.ac.id

July 2008

Revision after the Thesis Defense on August 12, 2008

STATEMENT BY THE AUTHOR

I hereby declare that this submission is my own work and to the best of my knowledge, it contains no material previously published or written by another person, not material which to a substantial extent has been accepted for the award of many other degree or diploma at any educational institution, except where due acknowledgement is made in the thesis.

Tjeuw Alvin Felix

Date

Approved by:

Husni Fahmi, Ph.D

Date

Chairman of the Examination Steering Committee

Date

ABSTRACT

WHITE-BOX AND BLACK-BOX SOFTWARE TESTING OF CLNP FUNCTIONS IN ATN

By

Tjeuw Alvin Felix

SWISS GERMAN UNIVERSITY

Bumi Serpong Damai

Husni Fahmi, Ph. D.

Charles Lim, M. Sc.

This thesis is aimed to conduct white-box and black-box software testing and provide software metrics for the CLNP code. The white-box testing methodologies will provide flowcharts, control flow graph, cyclomatic complexity, basis paths, and connection matrix. Moreover, black-box testing has also been conducted. However, since the CLNP code is currently under the optimization phase and the priority is currently pointed towards the implementation of AF_ATN raw socket for the ATN TP4/CLNP Networking Suite, the only feasible methodology to be conducted is white-box. To conclude, we have successfully determined and conducted white-box and black-box software testing for CLNP functions.

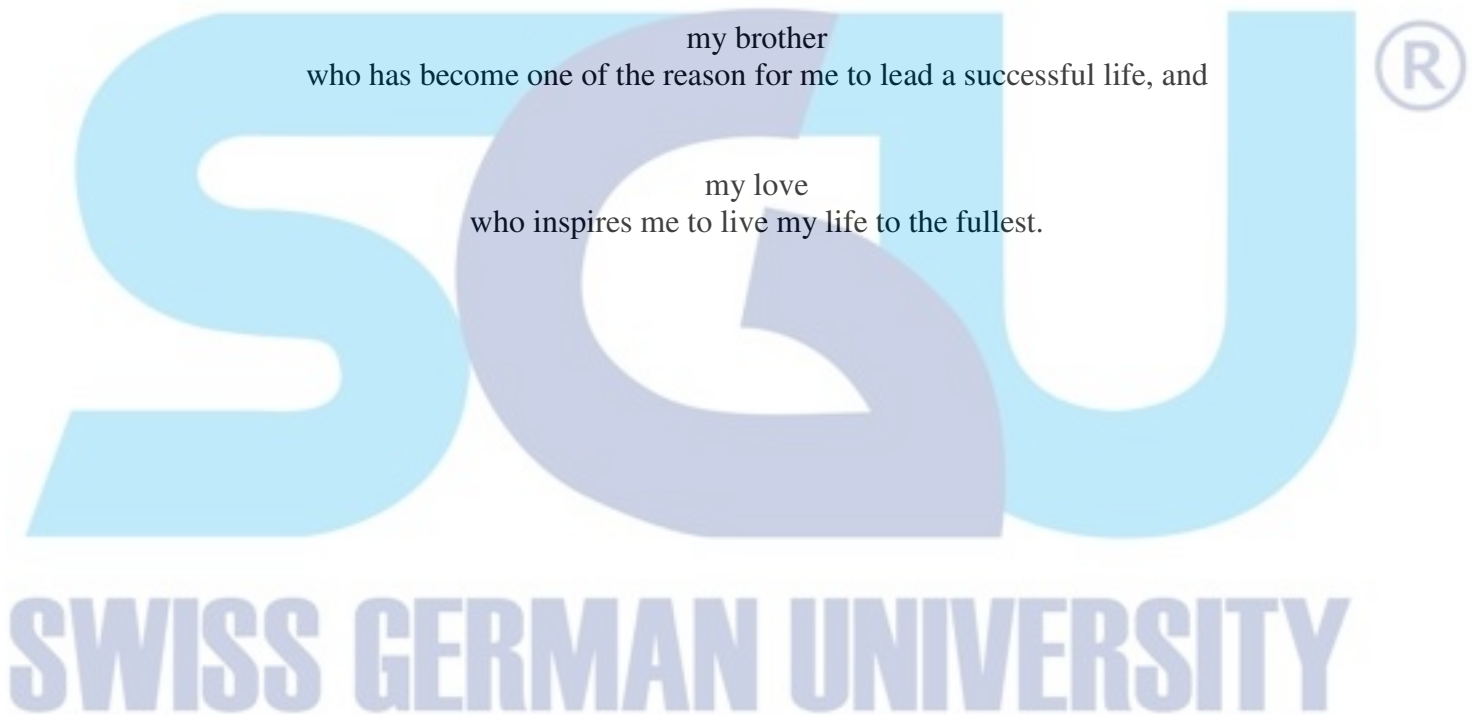
DEDICATION

This thesis is dedicated to

my parents
who bestow me never-ending support,

my brother
who has become one of the reason for me to lead a successful life, and

my love
who inspires me to live my life to the fullest.



ACKNOWLEDGMENTS

Praise to HIM who bestow health and guidance upon the author during the writing process of this thesis.

Warmth gratitude would be sent to Dr. Husni Fahmi, as the advisor, the motivator, and the chief scientist who inspires the author in the process of completing this thesis. Furthermore, a chance to do my thesis in this interesting topic of software project has been given by Agency for the Assessment and Application of Technology (BPPT). Author would also like to thank Mrs. Haret Faidah, Mr. Tri Sampurno, and Mr. Tahar for their help and support. Also another gratitude for Mr Oerip Santoso and other ATN team from ITB that author can not mention one by one

A great thank to Mr. Charles Lim who is giving his excellent support and utmost care as a co-advisor of this thesis. Furthermore, a sincere thank would be sent to Mr. Ford Lumban Gaol for his earnest and meaningful guidance in the process of writing this thesis. Moreover, it has been wonderful to work together with CLNP ES-IS team whose groups are Cipindo Tanjung (ES-IS Developer), Novianto Dharma (ES-IS Tester), and Tadeus Prastowo (CLNP Developer).

Special thanks to Tadeus Prastowo who gave his best encouragement and answers when the author faces problems.

TABLE OF CONTENTS

STATEMENT BY THE AUTHOR.....	2
ABSTRACT.....	3
DEDICATION.....	4
ACKNOWLEDGMENTS.....	5
CHAPTER 1 – INTRODUCTION.....	10
1.1 Background.....	10
1.2 Objectives.....	11
1.3 Related Work.....	11
1.4 Proposed Work.....	11
1.5 Result.....	11
1.6 Thesis Organization.....	12
CHAPTER 2 – LITERATURE REVIEW.....	13
2.1 Connectionless Network Protocol (CLNP).....	13
2.2 Verification and Validation.....	14
2.3 Software Testing.....	14
2.3.1 White-Box Testing.....	14
2.3.1.1 Flow Chart.....	15
2.3.1.2 Control Flow Graph (CFG).....	15
2.3.1.3 Cyclomatic Complexity (CC).....	17
2.3.1.4 Basis Path.....	20
2.3.1.5 Connection Matrix.....	20
2.3.2 Black-Box Testing.....	21
2.4 Software Metrics.....	21
2.4.1 Measurement of Software Complexity.....	21
2.4.2 Relationship between Software Complexity and Software Testing.....	22
2.4.3 Interpretation of Metrics.....	22
2.4.4 Software Metrics of C.....	22
CHAPTER 3 – METHODOLOGY.....	25
3.1 Proposed Method.....	25
3.2 Proposed White-box Software Testing.....	25
3.2.1 Flowchart.....	25
3.2.1.1 Improved Decision Block.....	30
3.2.2 Control Flow Graph (CFG).....	30
3.2.3 Cyclomatic Complexity (CC).....	34
3.2.4 Basis Path.....	36
3.2.4.1 Derivation of Possible Independent Path(s) From Basis Paths.....	36
3.2.5 Connection Matrix.....	37
3.3 Proposed Software Metrics.....	38
3.4 Overview of White-box Testing.....	40
CHAPTER 4 – RESULT & DISCUSSION.....	42
4.1 White-Box Software Testing of CLNP Functions.....	42
4.1.1 CLNP Checksum.....	42
4.1.1.1 CLNP Adjust Checksum.....	42

4.1.1.2	CLNP Check Checksum	46
4.1.1.3	CLNP Generate Checksum	50
4.1.2	CLNP Error.....	54
4.1.2.1	CLNP Discard.....	54
4.1.2.2	CLNP Emit Error	56
4.1.3	CLNP Fragment.....	58
4.1.3.1	CLNP Comp Fragment	58
4.1.3.2	CLNP Defragment	62
	From the connection matrix on Figure 4.7, we can see that row two, row four, row seven and row nine add four to complexity. Therefore, the cyclomatic complexity is four plus one, equal to eight.....	65
	CLNP Find.....	66
4.1.3.3	CLNP Fragment Destroy	69
4.1.3.4	CLNP Fragment Expires.....	72
4.1.3.5	CLNP Insert Frag.....	75
4.1.3.6	CLNP New Packet	79
4.1.3.7	Compare Address.....	83
4.1.3.8	Concatenate.....	86
4.1.4	CLNP Input.....	87
4.1.4.1	CLNP Address Check	87
4.1.4.2	CLNP Local Deliver	87
4.1.4.3	CLNP Local Deliver Finish	89
4.1.4.4	CLNP Receive	91
4.1.4.5	CLNP Receive Finish	96
4.1.4.6	Is Our Datagram.....	102
4.1.4.7	Option Part Handler	103
4.2	Software Metrics of CLNP.....	106
4.2.1	Software Metrics of CLNP Checksum	106
4.2.2	Software Metrics of CLNP Error.....	108
4.2.3	Software Metrics of CLNP Fragment.....	109
4.2.4	Software Metrics of CLNP Input.....	111
	CHAPTER 5 – CONCLUSION AND RECOMMENDATION	113
5.1	Conclusion.....	113
5.2	Recommendation for Future Work	113
	GLOSSARY	115
	REFERENCES	116
	APPENDIX A – CLNP Source Code	118
A.1	Source Code of <i>clnp_csum.c</i>	118
A.2	Source Code of <i>clnp_err.c</i>	121
A.3	Source Code of <i>clnp_fragment.c</i>	125
A.4	Source Code of <i>clnp_input.c</i>	135
	CURRICULUM VITAE.....	144