Peer to Peer Search Protocol

Network Partition Aware Decentralized Search

Abhimanyu Babbar

Master of Science Thesis

Software Engineering of Distributed Systems
School of Information and Communication Technology
KTH Royal Institute of Technology
Stockholm, Sweden

12 June 2015

Examiner: Professor Seif Haridi

Abstract

Your abstract here.

Sammanfattning

IETF xxxx Arbetsgruppen har definierat

Acknowledgements

I would like to acknowldge my adviser's help in getting access to the necessary packet traffic at a commercial operator (who should be thanked but must remain unnamed).

Contents

List of Figures

List of Tables

List of listings

List of Acronyms and Abbreviations

This document requires readers to be familiar with terms and concepts described in RFC 1235 [**?**]. For clarity we summarize some of these terms and give a short description of them before presenting them in next sections.

IPv4	Internet Protocol version 4 (RFC 791 [**?**])
IPv6	Internet Protocol version 6 (RFC 2460 [**?**])

Chapter 1

Introduction

It was conjectured in [**?**] that multicasting could provide gains by . . . .
See also [**?**], the paper [**?**], and the book [**?**].

1.1 Problem description

Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page

1.2 Problem context

Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page

1.3 Structure of this thesis

Chapter **??** describes the problem and its context. Chapter **??** provides the background necessary to understand the problem and the specific knowledge that the reader will need to understand the rest of this thesis. Following this Chapter **??** describes the goals, metrics, and solution proposed in this thesis project. The solution is analyzed and evalued in Chapter **??**. Finally, Chapter **??** offers some conclusions and suggests future work.

Chapter 2

Background

2.1 The Internet Protocol Suite

The Internet protocol suite was developed in order to ....

2.2 Hardware Abstraction Layer

Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. The details of the link layer encapsulation will be shown in Section **??** Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass.

2.3 Link layer Encapsulation

Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass.

Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. See Figure ?? which used the bytefield LaTeX package.

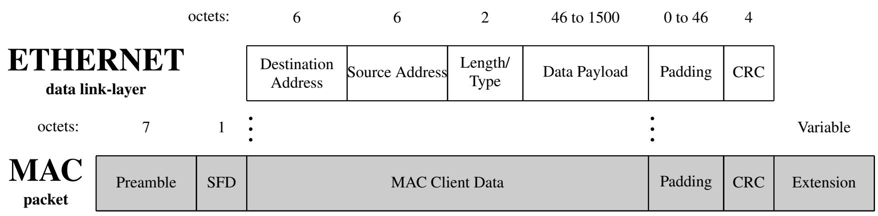

Figure 2.1: Ethernet data link layer protocol encapsulated into a IEEE 802.3 MAC packet

Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. See Figure ?? was drawn using OpenOffice and output as a PDF file.

Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page

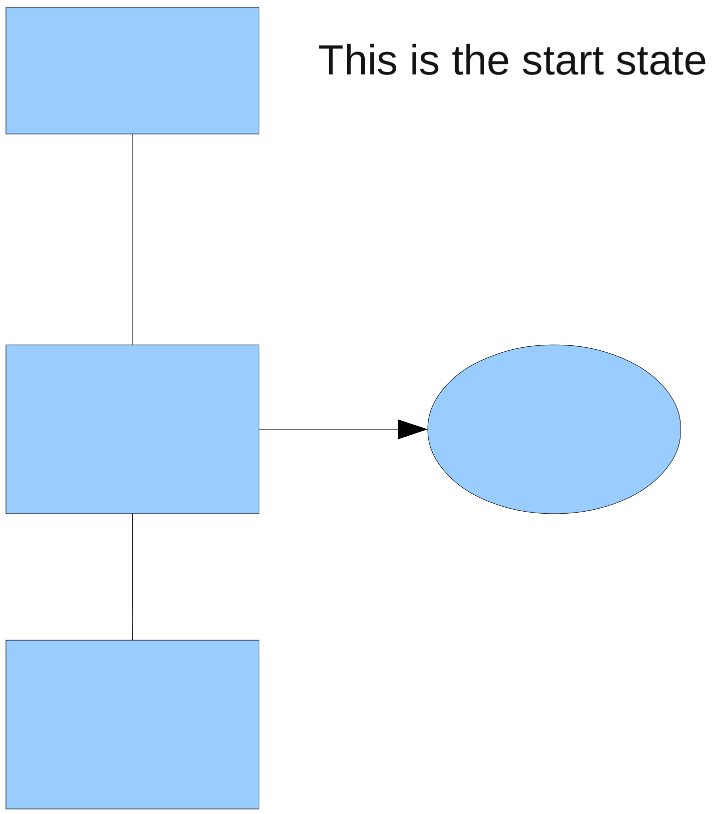

Figure 2.2: This state diagram illustrates nothing in particular.

Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass.

Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page

The data link layer will receive a packet from the IP layer. The layout of an IPv4 packet is shown in Figure **??**. This should be contrasted with the IPv6 header shown in **??**.

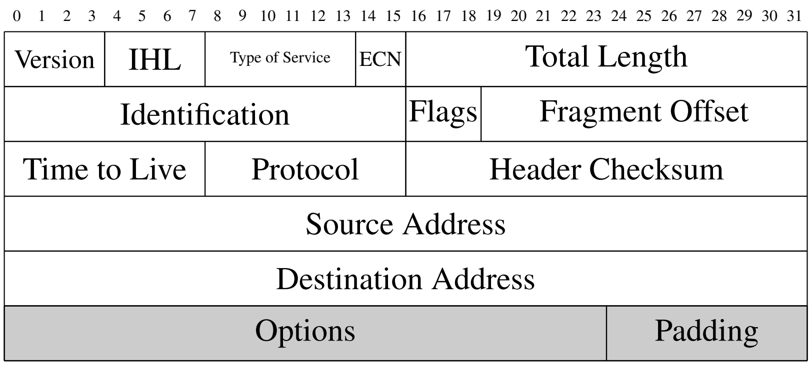

Figure 2.3: IPv4 datagram header. Light grey coloured fields are optional.

Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page Page filling text

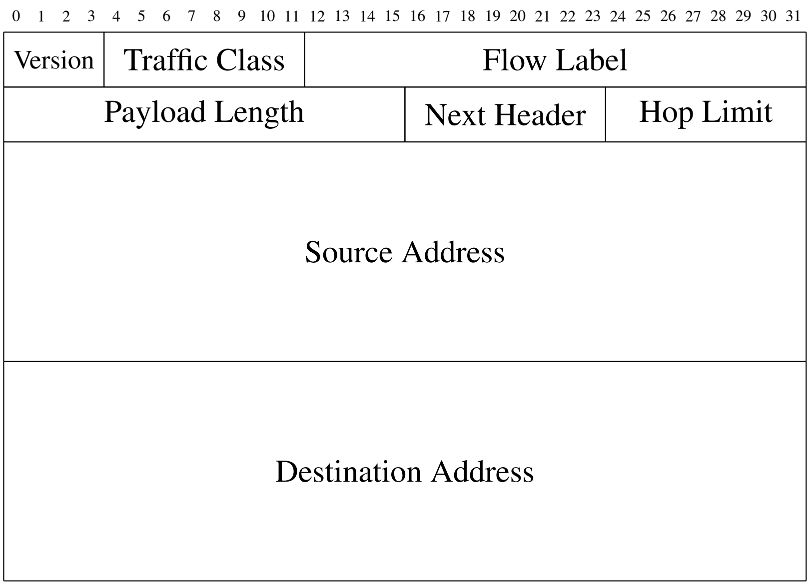

Figure 2.4: IPv6 datagram header

mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page

The code below shows .... While Listing **??** contains an example of a floating listing.

```
int main() {
  printf("hello, world");
  return 0;
}
```

Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page

```
int main() {
  int scalar = 2;
  int factor = 3;
  int result = scalar * factor

  printf("hello, world with %d", result);

  return result;
}
```

Listing 1: Example of a floating listing.

Figure **??** shows two alternative ways of wrting the same functionality in xxxxxx.

Page filling text mass. Page filling text mass. Page filling text mass. Page

```
state : {GREEN, RED}

void semaphore() {
 state = GREEN  // set initial state
 light(GREEN)
 timer_set(GREEN_TIMER)

 while (1) {
  switch(state) {
  case (GREEN):
   if (timer_expired(GREEN_TIMER)) {
    state = RED
    light(RED)
    timer_set(RED_TIMER)
   }
  case (RED):
   if (timer_expired(RED_TIMER) ||
     pedestrian_button_pressed()) {
    state = GREEN
    light(GREEN)
    timer_set(GREEN_TIMER)
   }
  }
 }
}
```

State machine implemented using a traditional loop-switch mechanism.

```
pt_semaphore:
PT_BEGIN
 while (1) {
  light(GREEN)
  timer_set(GREEN_TIMER)
  PT_WAIT_UNTIL(timer_expired(GREEN_TIMER))

  light(RED)
  timer_set(RED_TIMER)
  PT_WAIT_UNTIL(timer_expired(RED_TIMER) ||
    pedestrian_button_pressed())
 }
PT_END
```

State machine implemented using the protothreads abstraction mechanism.

Figure 2.5: Some sample code for example 1.

filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. See Table **??**.

Page filling text mass. Page filling text mass.

2.4 Related work

There are three earlier research projects that have done work related to this thesis topic. The first considers the problem from the point of view of throughput as measured in packets per second, the second in terms of throughput in bits per second, and the third in terms of the number of messages that are sent per second. Each of these methods will be described in the following subsections.

Table 2.1: Some IEEE 802.15.4 physical layers, sorted by release date

Physical layer (MHz)	**Frequency Band (MHz)**	**Modulation**	**Bit rate (kb/s)**
868/915	868 – 868.6	BPSK	20
	902 – 928		40
868/915	868 – 868.6	ASK	250
	902 – 928		250
2,450 (CSS)	2,400 – 2,483.5	DQPSK	1,000
			250
780	779 – 787	O-QPSK	250

2.4.1 Packets per second

Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass.

2.4.2 Bits per second

Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass.

2.4.3 Messages per second

Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass.

Chapter 3

Method

Chapter 4

Analysis

Figure **??** shows and example of the performance as measured in your experiments. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page

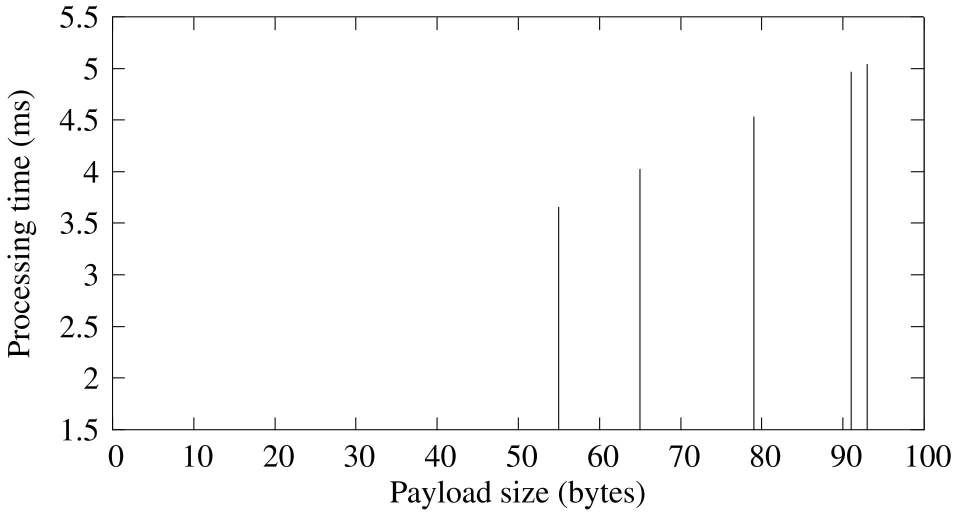

Figure 4.1: Processing time vs. payload length

Given these measurements, we can calculate our processing bit rate as the inverse of the time it takes to process an additional byte divided by 8 bits per byte:

$$bitrate = \frac{1}{\frac{time_{byte}}{8}} = 20.03 \quad kb/s$$

Page filling text mass. Page filling text mass.

Table **??** shows part of a table convert from an Excel spreadsheet to a LaTeX table using Calc2LaTex.

Page filling text mass. Page filling text mass.

Table 4.1: Main board

PART	TYPE	Digikey part	Order No	Qty	Value	Description
B1,B2	BRIDGE RECTIFIER DF01S	DF01STR-ND	1470959	2		
C1,C3,C5,C7,C9, C18,C19,C20, C21,C22,C23, C24,C27,C28	CAPACITOR 0603 (1608 metric)	399-1095-1-ND	9406140	14	100nF	
C11	CAPACITOR 0603 (1608 metric)	445-3454-2-ND	1710317	1	470nF	
C12,C13	CAPACITOR 0603 (1608 metric)	445-1270-2-ND	1740592	2	12pF	
C14	CAPACITOR 0603 (1608 metric)	445-1289-1-ND	1740626	1	470pF	
C16,C17,C32,C33	CAPACITOR 0603 (1608 metric)	445-1272-2-ND	1740595	4	18pF	
C2,C4,C6,C8, C10,C15	CAPACITOR 1206 (3216 metric)	493-2351-1-ND	1190107	6	10uF	POL
D1	DIODE SMAJ58A (DO214A)	SMAJ58ALFTR-ND	1899460	1		
D2,D3,D5	DIODE 11DQ09-ND (DO41-10)	11DQ09-ND	3694069	3		
D4	DIODE ZENER 60V (SOD123)	MMSZ5264BT1GOSTR-ND	1895056	1		60V
JP1,JP2	CONNECTOR 1x3 MALE		1022249	2		
JP3	CONNECTOR 2X4 MALE		1022233	1		
JP4	CONNECTOR 2X7 MALE		1319205	1		
L1	FERRITE BEAD (1806)	240-2541-1-ND		1		
L2	INDUCTOR 0805 (metric 2012)	587-2068-1-ND	1457862	1	47uH	160mA
L3	INDUCTOR PULSE ENGINEERING PE53120	553-1588-5-ND	1209561	1	1000uH	
L4	INDUCTOR 0805 (metric 2012)	587-2046-1-ND	1457861	1	22uH	
LED1	LED, SMD 1206, RED	160-1457-1-ND	1465997	1		
LED2,LED3	LED, SMD 1206, GREEN	160-1456-1-ND	1466000	2		
LED4	LED, SMD 1206, YELLOW	160-1458-1-ND	1465998	1		
Q1	CRYSTAL 32,768 kHz (TC26H)	300-8303-ND	1457084	1		
Q2	CRYSTAL 32MHz (HC49/US)	300-8518-ND	1078935	1		
Q3	CRYSTAL 25MHz (HC49/US)	300-8513-ND	2057969	1		
R1,R13,R15	RESISTOR 0603	RMCF0603FT47K0CT-ND	1469811	3	47k	
R10,R11	RESISTOR 0603	P330HCT-ND	1469803	2	330	
R12,R14	RESISTOR 0603	RMCF0603FT10K0CT-ND	1469748	3	10k	
R2,R3,R8	RESISTOR 0603	P470HCT-ND	1469815	3	470	
RBIAS	RESISTOR 0603	P2.32KHTR-ND	1170823	1	2.32k	
RCLASS	RESISTOR 0805	P953CTR-ND	1653044	1	953	
RDET	RESISTOR 0603	P24.9KHTR-ND	1469785	1	24.9k	
RLIM	RESISTOR 0603	P442KHTR-ND	1171050	1	442k	
RPULLUP	RESISTOR 0603	P24KGTR-ND	1469784	1	24k	
S1, S2	PUSH BUTTONS DTSM6	679-2383-2-ND		2		
T1	RJ45 JACK 7499211123	732-1862-ND		1		
U1	MSP430F5437a	MSP430F5437AIPN-ND		1		
U2	ENC28J60	ENC28J60-I/ML-ND	1564400	1		
U3	TPS2375	296-17061-ND	8456860	1		
X1	CONNECTOR DC INPUT		1243245	1		

Chapter 5

Conclusions

This chapter explains the conclusions obtained throughout the design, development, and evaluation described in this thesis and proposes a number of improvements, extensions, or complements that may be of interest in order to continue this work. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass.

5.1 Conclusion

In this section we will state the conclusions and insights gained as result of this thesis project.

5.1.1 Goals

The project met five of the ten initial goals. With the third goal not addressed at all and goals 6-10 only partially met. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass.

5.1.2 Insights and suggestions for further work

One of the most important insights to come from this work is the need to understand the interaction between the IP layer and the link layer. Unforunately I did not have time to examine this in detail in this thesis. However, the data shows that paying attendtion to this APR cache flushing patterns could yield improved performance. Future works should start by measuring the relationship between the cache validity and number of packets which are generated at or above the IP layer, but not transmitted on the network. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass.

5.2 Future work

Due to the breadth of the problem, only some of the initial goals have been met. In these section we will focus on some of the remaining issues that should be addressed in future work. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass.

5.2.1 What has been left undone?

The prototype does not address the third requirment, i.e., a yearly unavailability of less than 3 minutes, this remains an open problem. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass.

5.2.1.1 Cost analysis

The current prototype works, but the performance from a cost perspective makes this an impractical solution. Future work must reduce the cost of this solution, to do so a cost analysis needs to first be done. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass.

5.2.1.2 Security

A future research effort is needed to address the security holes that results from using a self-signed certificate. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass.

5.2.2 Next obvious things to be done

In particular, the author of this thesis wishes to point out xxxxxx remains as a problem to be solved. Solving this problem is the next thing that should be done. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass. Page filling text mass.

5.3 Required Reflections

Bibliography

[1] J. Ioannidis and G. Q. M. Jr., "Coherent file distribution protocol," *Internet Request for Comments*, vol. RFC 1235 (Experimental), Jun. 1991. [Online]. Available: http://www.rfc-editor.org/rfc/rfc1235.txt

[2] J. Postel, "Internet protocol," *Internet Request for Comments*, vol. RFC 791 (Standard), Sep. 1981, updated by RFC 1349. [Online]. Available: http://www.rfc-editor.org/rfc/rfc791.txt

[3] S. Deering and R. Hinden, "Internet protocol, version 6 (IPv6) specification," *Internet Request for Comments*, vol. RFC 2460 (Draft Standard), Dec. 1998, updated by RFCs 5095, 5722, 5871. [Online]. Available: http://www.rfc-editor.org/rfc/rfc2460.txt

[4] S. Long, "Database synchronization between devices: A new synchronization protocol for sqlite databases," Master's thesis, KTH, School of Information and Communication (ICT), Stockholm, May 2011, TRITA-ICT-EX-2011:88. [Online]. Available: http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-46712

[5] Anand Kannan, Gerald Q. Maguire Jr., Ayush Sharma, Volker Fusenig, and Peter Schoo, "N-ary tree based key distribution in a network as a service provisioning model," in *ICACCI '12 Proceedings of the International Conference on Advances in Computing, Communications and Informatics*. ACM Press, 2012. doi: 10.1145/2345396.2345550. ISBN 9781450311960 p. 952. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2345396.2345550

[6] Brent S. Baxter, Lewis E. Hitchner, and Gerald Q. Maguire Jr., *A standard format for digital image exchange*. New York N.Y.: Published for the American Association of Physicists in Medicine by the American Institute of Physics, 1982. ISBN 9780883184080

Appendix A

Insensible Approximation