UNIVERSITEIT
STELLENBOSCH
UNIVERSITY

Improving Automatic Position Reporting System (APRS) Throughput and Reliability

by

Hendrik Petrus van Tonder

Thesis presented at the University of Stellenbosch in partial fulfilment of the requirements for the degree of

Masters of Science in Electrical and Electronical Engineering

Department of Electrical and Electronical Engineering
University of Stellenbosch
Private Bag X1, 7602 Matieland, South Africa

Study leader: Prof S. Mostert

December 2004

Declaration

I, the undersigned, hereby declare that the work contained in this thesis is my own original work and that I have not previously in its entirety or in part submitted it at any university for a degree.

Signature: ....

H.P. van Tonder

Date: ....

Abstract

The Automatic Position Reporting System (APRS) is a well-established packet communication protocol that offers users a graphical position display system and a peer-to-peer textual message service. APRS is used in temporary and mobile networks where rapid deployment of infrastructure is required and limited *a priori* knowledge of the network topology is available. The APRS protocol can be used for emergencies and public service applications.

ARPS, functioning as an access network, was originally designed to require low complexity and support high flexibility of a network. These design directives have limited APRS's performance by resulting in low throughput and poor reliability. In order for APRS to be used in time-critical applications, these limitations would need to be improved.

The thesis considers the limitations of ARPS by proposing an improved protocol stack with a substitution of the media access control (MAC) layer. The new protocol is modelled in order to develop a largely platform-independent implementation, which could be efficiently retargeted for different platforms. Lastly, a protocol performance evaluation is done in order to determine the resulting improvements on APRS and the overall viability of the proposal.

Opsomming

Die Outomatiese Possisie Raporterings Stelsel (ARPS) is 'n gevestige pakkie kommunikasie stelsel en bied gebruikers 'n grafiese possisie vertoning stelsel en 'n gebruiker-tot-gebruiker teks boodskapdiens. APRS word gebruik in tydelike and mobiele netwerke waar vinnige ontplooing van infrastruktuur vereis word en beperkte *a priori* inligting van die network topologie beskikbaar is. Die APRS protokol kan gebruik word ten tyde van noodgevalle en vir toepassings in publieke dienste.

APRS, wat funksioneer as 'n toegangsnetwerk, is oorspronklik ontwerp om lae kompleksiteit te vereis en hoë buigsaamheid van 'n netwerk te ondersteun. Hierdie ontwerpsvereistes het veroorsaak dat APRS se werkverrigting beperk word deur 'n lae data deurvoer en betroubaarheid. Ten einde APRS se gebruik in tydkritiese toepassings te bevorder, sal hierdie beperkinge verbeter moet word.

Hierdie tesis bied 'n voorgestelde verbetering op die beperkinge van APRS deur 'n vervanging van die media toegangsbeheer vlak van die protokol stapel. Die nuwe protokol word dan gemodelleer ten einde 'n relatiewe platform onafhanklike implementering te ontwerp wat doeltreffend vir ander platforms aangepas kan word. Laastens word 'n protokol werkverrigting evaluasie gedoen om die verbetering op APRS te bepaal en die algehele lewensvatbaarheid van die voorstel vas te stel.

Acknowledgements

I would like to express my sincere gratitude to the following people and organisations who have contributed to making this work possible:

- My supervisor Sias Mostert for his guidance and inspiration to always see the "big" picture,
- The ESL personnel for the organized and sufficient research environment that they provided,
- Mr F. Retief for his support with Linux and the C language,
- Miss L. Scheepers for her encouragement and wisdom,
- My family and friends who supported and encouraged me.

Dedications

The thesis is dedicated to my Lord and Saviour, Jesus Christ,
who is the way, the truth and the life,
in whom only is abundant life !

Contents

List of Acronyms and Abbreviations

ACK	Positive Acknowledgement
AP	Access Point
APRS	Automatic Position Reporting System
APT	Access Point Terminal
BN	Base Network
BS	Base Station
CN	Cell Network
CRC	Cyclic Redundancy Check
CS	Control Station
CSMA	Carrier Sense Multiple Access
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CT	Channel Throughput
CTS	Clear-to-Send
DAMA	Demand Assigned Multiple Access
DASAP	Data-Link/Application Service Access Point
DL	Data-Link
DPSAP	Data-Link/Physical-Link Service Access Point
DT	Data Throughput
EFSM	Extended Finite State Machine
EI	Environment Interface
FCS	Frame Sequence Check
FDM	Frequency Division Multiplexing
FDMA	Frequency Division Multiple Access
FS	Field Station
FSM	Finite State Machine
GCC	General Computer Corporation
GPS	Global Positioning System
HTS	Hidden Terminal Syndrome

LEO	Low Earth Orbit
MAC	Media Access Control
MACA	Multiple Access with Collision Avoidance
MHT	Multiple Host Terminal
NAK	Negative Acknowledgement
OSI	Open Systems Interconnect
PL	Physical-Link
PLE	Physical-Link Emulator
PLS	Physical-Link Server
RF	Radio Frequency
RTS	Request-to-Send
S-ALOHA	Slotted-ALOHA
S-APRS	SRMA-Automatic Position Reporting System
SAP	Service Access Point
SDL	System Description Language
SRMA	Split-Channel Reservation Multiple Access
TAPR	Tucson Amateur Packet Radio
TDM	Time Division Multiplexing
TDMA	Time Division Multiple Access
TNC	Terminal Node Controller
TOR	Transmission Overhead Ratio
UI	Unnumbered Information
UML	Unified Modeling Language

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

The Automatic Position Reporting System (APRS) is a well-established packet communications protocol mainly used by packet-radio amateurs. APRS offers a graphical application user-interface that displays the positions of hosts on regional to worldwide maps using the Global Positioning System (GPS). The user-interface also provides a textual peer-to-peer messaging service to all the APRS hosts. Figure 1.1 shows a screenshot of an APRS application displaying a regional map with hosts being encircled.

APRS supports any host equipped with a two-way radio system, including amateur radio, marine band and cellular phones. The hosts disseminate their data packets throughout the APRS network using a type of flooding routing algorithm. The routing avoids the complexity of a connected network and makes the protocol adaptable to variation in network topology. Different APRS networks can also be interconnected via the Internet backbone using hosts that function as gateways.

APRS is mostly useful for temporary portable and mobile operations where it is often not feasible to coordinate a multi-host network in advance. Examples are emergency situations and public service applications where the rapid deployment of a communication infrastructure could be required. In the examples, APRS functions as a tool for monitoring the real-time position of units and provides the peer-to-peer communication necessary to coordinate activities.

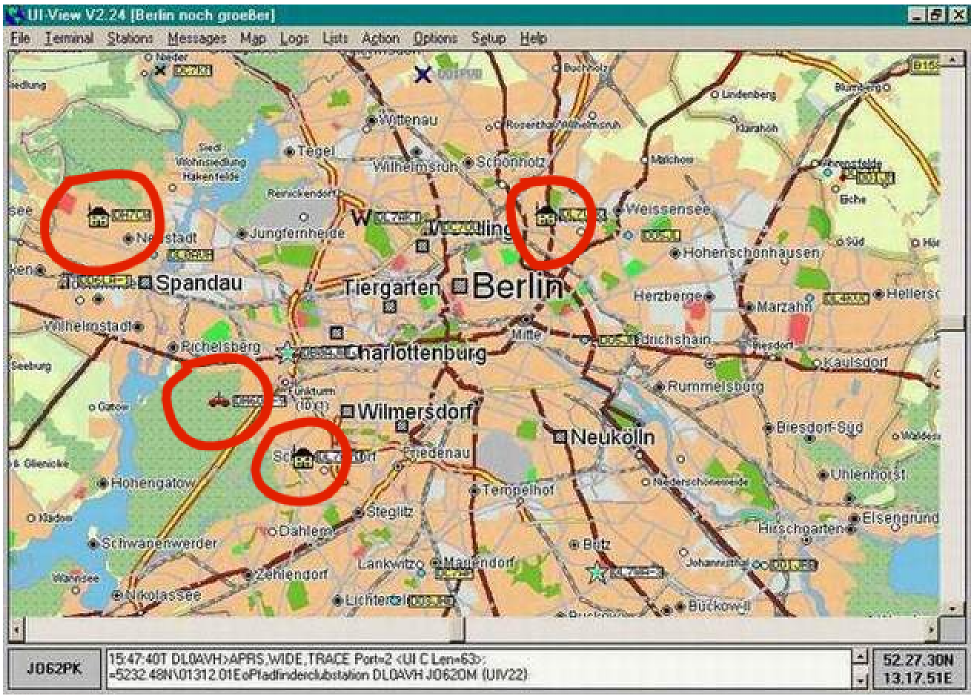


Figure 1.1: A map screenshot of an APRS application

1.2 Major Challenge

APRS was originally designed to address mainly two aspects of networks, namely the flexibility and complexity. The flexibility of the network refers to its ability to effectively support varying network population. It includes minimizing the routing reconfiguration of the network and the amount of *a priori* knowledge required of the network topology. The complexity of the network refers to the network architecture that includes the host hardware and software components and the measure of network intelligence required for individual hosts.

In the cause of meeting the requirements, APRS's protocol-design limits its efficiency with regard to two network characteristics:

1. Channel throughput and
2. Reliability.

APRS has a low maximum channel throughput with a resulting increase in network delay. The low measure of throughput introduces an additional time that a packet needs to wait in order to be successfully transmitted. APRS provides no indication or guarantee of packet delivery, resulting in decreased reliability. In order to improve the key performance indicators, the APRS protocol aspects governing the characteristics will need to be addressed.

APRS is classified as an access network where multiple-access is required to a single transmission channel. The multiple-access is governed by the *media access control (MAC)* implemented in the protocol stack. The choice of MAC technique is the main design challenge in access network protocols.

The first limited performance indicator, the measure of channel throughput, is determined by how efficiently the protocol stack implements media access control. The MAC technique determines the amount of packet collisions, which in return limit the maximum effective throughput of the channel.

The second limited performance indicator, the reliability of APRS, refers to the guarantees provided on packet-delivery, which is also determined by the MAC layer protocol. In APRS the MAC layer includes redundant transmission of a packet using a decay algorithm. It means that the packet is transmitted a few times without any indication of successful delivery. There are also no acknowledgements in response to deliveries.

Tucson Amateur Packet Radio (TAPR), which is an educational research and development organization, manages the development of the APRS implementation. Their research addresses the maintenance of APRS with regard to routing issues that presently result in network performance degradation. Research seems to indicate that there is no ongoing development of APRS other than that of TAPR.

The literature review has indicated the existence of a variety of MAC techniques that address different subsets of network characteristics. The characteristics include scalability, throughput, latency, bandwidth utilization and reliability. Therefore, in order to improve the two performance indicators of APRS, a combination of MAC technique principles will need to be considered.

1.3 Statement of Purpose

The aim of the thesis is to study the APRS protocol stack implementation in order to quantify its efficiency with regard to its maximum effective channel throughput and reliability. The two performance indicators, as described above, will be improved by the substitution of the MAC layer protocol.

In order to accomplish the aim, the following research objectives are defined:

1. Substitute the MAC layer protocol with minimal changes to the APRS specification

and network architecture,

2. Develop a protocol model of the new protocol stack in order to create a platform independent implementation and from which the implementation overhead can be effectively measured and optimized,

3. Evaluate the protocol performance characteristics and determine the improvement on the APRS performance indicators.

Finally, having attended to the objectives, the feasibility of the overall proposed protocol will be determined.

1.4 Research Significance

The limitations on the performance indicators compromise APRS's acceptance in a wider spectrum of applications that require higher level of guarantees and reliability. Typical applications are high-risk emergencies such as wildfires threatening residential areas. The applications require reliable communication of unit positions and low-delay peer-to-peer communication is of vital importance. The research will also improve APRS's set of guarantees that will increase its potential for acceptance in commercial use.

1.5 Work Methodology

The literature review has indicated that MAC-protocol performance is mainly evaluated with theoretical analysis [1; 2; 3; 4]. The methods followed in translating the different protocol characteristics into equations are well established.

The evaluation of only a small subset of the characteristics cannot sufficiently be done with an analysis. A good example of such a characteristic is the dynamics of routing algorithms contained in protocols. The characteristics require the method of simulation or experimentation to effectively approximate their behaviour.

The throughput of the proposed protocol can be sufficiently evaluated with an analysis and is the main methodology used in the research. The second method, experimentation, is followed by developing a software implementation of the protocol. The implementation is used to make implementation-specific measurements.

The implementation created from the protocol model is developed using a Unified Modeling Language (UML) description. UML provides a good platform to create models using object-orientated concepts. The platform enables software developers to efficiently produce code for the implementation. The more detailed description of the protocol is done with the System Descriptive Language (SDL). It consists of extended finite state machines that model the communication processes.

1.6 Thesis Structure

Here is an outline of the remainder of the document:

- **Chapter 2: Access Networks and Techniques**
  An introduction to access networks is given that includes a comparison of media access control techniques. APRS is classified as a type of access network and its performance is evaluated. A substitute for its MAC layer is proposed, which is evaluated in remainder of the document.

- **Chapter 3: Network Topology and Protocol Integration**
  A specification is given of the research-specific APRS network topology. The APRS standard is introduced and the integration with the new MAC layer protocol is described.

- **Chapter 4: Protocol Modeling and Emulation**
  The protocol stack is modelled using UML and the software implementation is developed and evaluated.

- **Chapter 5: Network Performance Analysis**
  The analytical performance evaluation is done of the substitute MAC layer in the new APRS protocol stack.

- **Chapter 6: Conclusions and Recommendations**
  Conclusions are drawn from the results with regard to the initial objectives of the research. Further recommendations and areas for research are suggested.

- **Appendices**
  Consist of a number of UML and SDL specification diagrams that are part of the S-APRS protocol model and the implementation code.

Chapter 2

Access Networks and Techniques

The chapter introduces the concept of access networks and the important aspects that need to be considered in order to address the limitations of APRS. An overview of MAC techniques is given, which describes the set of access network aspects that are addressed by each. The APRS access network characteristics that need to be improved are then identified and the appropriate MAC protocol substitute is chosen.

2.1 Access Networks

Access networks represent the set of networks that include a single access point that multiple hosts use to access the backbone of a network. The random simultaneous access to the access point required by hosts leads to contention on the single shared channel. The multiple-access results in a period during which a channel is unavailable to a host. The result is a limit on the amount of data that can be transmitted by each user during a transmission cycle. The multiple-access requirement and the techniques that are used to govern its efficiency are the main areas of ongoing development in access networks. As stated in the introduction, the area is generally referred to as media access control (MAC).

2.1.1 Media Access Control(MAC)

Media access control is implemented in the MAC sub-layers that form part of the data-link layer of the Open Systems Interconnect (OSI) [5] protocol stack model. Table 2.1 is a conceptual representation of the model.

The MAC layer serves two main functions as described in [6]. Firstly, it performs data

Table 2.1
THE OSI PROTOCOL STACK MODEL

Layer	Function
7	Application
6	Presentation
5	Session
4	Transport
3	Network
2	Data Link (MAC)
1	Physical Link

encapsulation, which includes framing, source and destination addressing and lastly, error detection. Secondly, it performs media access management. Media access management is the most challenging part of the MAC layer and also the subject mainly considered in the thesis. The management's first priority is to prevent simultaneous transmission of hosts, known as collision avoidance. If a collision occurs, the MAC layer's second priority is to do contention resolution. Contention resolution tries to correct the collision by either retransmitting the packet or rescheduling its transmission.

The dynamics of MAC and their combined implementation produce a set of performance measures. The measures and the principles that underlie them are introduced in the following section.

2.1.2 MAC Performance Measures

There are mainly two performance measures that need to be defined:

1. Throughput and
2. Latency.

The first measure requires a distinction to be made between terms that are frequently the cause of confusion in telecommunication literature. The terms are *bandwidth* and *throughput*.

In the thesis, the bandwidth of the channel refers to the maximum amount of bits per second that can be transmitted on the channel modulated at 1 bit/Hz·s. Important to note is that in practice, the transmitted amount is not necessarily the amount of data that is effectively transmitted and received on the channel. The throughput represents the actual measured performance of the channel, which is generally less than the bandwidth.

In the thesis, the term *channel throughput (CT)* is used to represent the actual throughput of the channel. The maximum effective CT is then its limited value describing its optimal performance.

The relation between the terms are defined in [7] as

$$Throughput \ = \ \frac{TransferSize}{TransferTime} \tag{2.1.1}$$

with

$$TransferTime \ = Propagation\ Delay \ + \ Transmission\ Delay \tag{2.1.2}$$

The first term in equation (2.1.2) is the transmission delay, which is

$$Transmission\ Delay \ = \frac{Transfer\ Size}{Baud\ Rate} \tag{2.1.3}$$

The propagation delay is determined by the link medium properties used in the access network. It is defined as

$$Propagation\ Delay \ = \frac{Distance\ to\ transmit}{Medium\ Propagation\ Speed} \tag{2.1.4}$$

The medium propagation speed is the fraction of the free space (vacuum) propagation speed of c $= 3{\times}10^8\ m/s$.

The second measure, latency, has already been partly introduced by the section. It constitutes three delays and is defined in [7] as

$$Latency \ = Transmission + Propagation + Queuing \tag{2.1.5}$$

The last term, the queuing delay, is the result of packets that need to be stored by the individual nodes along the routing path before they are forwarded.

The measures that have been introduced all depend on the characteristics of the network traffic. The important measures are considered next in order to understand their effect on the MAC performance measures.

2.1.3 Network Traffic

The network traffic is characterized by the following measures:

- Bandwidth

Figure 2.1: A low duty cycle packet input rate

- Packet length (TransferSize)
- Medium propagation speed
- Packet Duty Cycle

The relation between the bandwidth and packet length has already been defined. The medium propagation speed is defined as the ratio of the speed of light (c) in a vacuum. The speed determines the propagation delay for the packet and the maximum distance between two communicating nodes.

The final and important measure to consider when choosing the traffic model is the packet duty cycle. The duty cycle refers to the total rate at which packets are generated accumulatively on the network. The duty cycle decreases as the packet sources become more graphically distributed. Figure 2.1 shows a low duty cycle rate. The choice of the optimal MAC technique will depend upon the APRS network duty cycle.

2.2 MAC Techniques

The set of MAC techniques are categorized according to the network topology, mobility and most important, the nature of the data traffic. They are divided in three main categories:

1. Fixed assignment
2. Random Access
3. Centrally Controlled Assignment

The section vriefly introduces all three categories.

2.2.1 Fixed Assignment

Fixed assignment divides the channel into fixed segments in either the time or frequency domain. The result in the frequency domain is one or more simultaneous, continuous frequency channels each having a separate portion of the frequency spectrum. The frequency domain technique is known as Frequency Division Multiple Access (FDMA).

Its time-domain counterpart, Time Division Multiple Access (TDMA), produces separate continuous time channels that sequentially give each user access to the total bandwidth of the channel.

Fixed assignment is well suited for access networks that have a known and fixed number of users with a user input rate corresponding to a high duty cycle. A high duty cycle refers to a low ratio of the peak to average data rate. It implies that for the majority of the time, most of channel's capacity is utilized and results in efficient bandwidth utilization.

As hosts become more graphically distributed or their mobility in the network increases, the traffic become more random and the duty cycle decreases. Decreasing the duty cycle results in fixed assignment schemes becoming increasingly wasteful with regard to channel bandwidth. The reason is that the fixed channels idle for larger portions of the protocol cycle, which also decreases the channel throughput. The next category of MAC techniques is designed to address the bandwidth inefficiency.

2.2.2 Random Access

Random access techniques are more efficient with bursty sources than their fixed assignment counterparts. The techniques are discussed in the section in the order of increasing efficiency.

ALOHA

The most basic of random access techniques is ALOHA. Users on a single channel transmit packets randomly with no coordination between users. The technique is very flexible in that users are added and removed with the minimal reconfiguration of the network.

The throughput of ALOHA increases as the duty cycle of data traffic decreases. The main drawback of the technique is its low maximum channel throughput of 18%, as determined by [4; 8].

Slotted-ALOHA (S-ALOHA)

Slotted-ALOHA adds a slight improvement to ALOHA which doubles the throughput. The channel is divided into time-slots similar to TDMA. The difference is that each user randomly chooses a *slot* in which to transmit a packet. The result is that S-ALOHA wastes

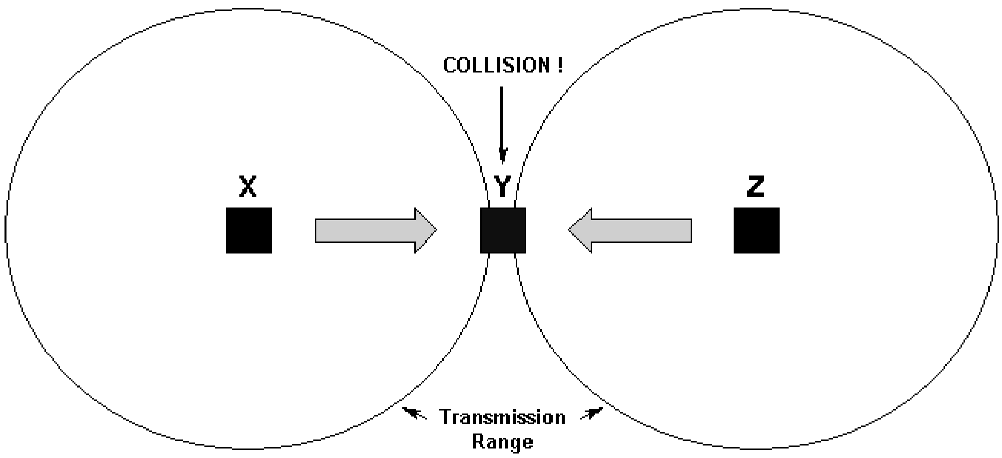


Figure 2.2: Hidden Terminal Syndrome (HTS) illustration

less bandwidth than TDMA for bursty sources and doubles the maximum throughput of ALOHA.

The probability of a collision is the probability of two users choosing the same random time slot. With only the added complexity of host synchronization, the maximum channel throughput of S-ALOHA is 38%.

Carrier Sense Multiple Access / with Collision Avoidance (CSMA and CSMA/CA)

Carrier Sense Multiple Access [1] has a further improvement on throughput. Each host senses the channel to determine if it is idle in order to transmit a packet. If the channel is busy, the host waits for a random period of time and then retries. CSMA produces a maximum channel throughput of up to 80% [9; 10; 11; 1] depending upon the ratio of the propagation delay to packet transmission time.

An essential assumption of CSMA is that each host is in the transmission range of every other host on the network. Hosts that do not comply with the criterion lead to the Hidden Terminal Syndrome (HTS) [2]. Hidden-terminals are prevalent in wireless networks where the transmission range of hosts is limited by power considerations. A simplified example of the HTS problem is shown in Figure 2.2.

Suppose neither of the three stations X,Y and Z are transmitting. Station X senses the channel to be idle and starts its transmission to Y. Almost at the same instance, Z senses the channel, and because it is out of range of X, Z incorrectly assumes the channel to Y is idle. Z then transmits and at Y a collision of the X and Z transmissions occurs. Y is identified as the hidden terminal. As mentioned, CSMA assumes that all the transmitters on the network are in range of each other, and therefore suffers from HTS.

An extension to CSMA, CSMA with collision avoidance (CSMA/CA), addresses the moment in multiple-access that has the highest probability of a collision, the moment when the channel is released. In CSMA/CA, after the host has sensed the medium idle, it backs off for a random time before transmitting. The random back off decreases the number of collisions and has an increased throughput compared to CSMA. CSMA/CA also suffers from HTS and require the hardware to have channel-sensing ability.

Multiple Access with Collision Avoidance (MACA)

Multiple Access with Collision Avoidance [12] introduces the idea of a Request-to-Send (RTS) and a Clear-to-Send (CTS) packet dialogue. A host wanting to send data first sends a RTS packet to the destination. Every host that overhears the RTS suspends its own transmission for the time needed by the destination to respond with the CTS. The time that it would take to transmit the data packet is included in the RTS. The destination then in turn replies with the CTS packet that inhibits other hosts of transmitting on the channel for the transmit time specified in the RTS, which is included in the CTS.

The RTS and CTS sequence enable MACA to reduce the existence of the HTS problem. The example in §2.2.2 is used again to illustrate the solution. By having the destination Y transmit a CTS in response to the RTS from X, the station Z is inhibited form transmitting for the time included in the CTS. With MACA, the Hidden Transmission Syndrome can still occur. The reason is because the sum of the transmission delay from X to Y and the transmitter delay of Y gives an aggregate delay during which time Z can also send an RTS that would then collide at Y. Therefore MACA reduces HTS greatly, but does not completely remove the problem.

In conclusion, MACA reduces the overhead caused by collisions and is an improvement on CSMA/CA throughput as long as the RTS packets are significantly smaller than the data packets [12]. It also minimizes the occurrence of HTS, the Hidden Terminal Syndrome.

2.2.3 Centrally controlled assignment

In centrally controlled assignment the focus shifts from a distributed user protocol to a centrally controlled one. A central station performs the bandwidth assignment for hosts and as a whole provides a more coordinated network. Three protocols are considered that address different scenarios.

Polling

Polling is the first scenario where the central station initiates the protocol cycle. It transmits a polling packet to each host sequentially, which gives each host the opportunity to transmit a fixed amount of data. Therefore, the channel is divided using time-division-multiplexing. As the traffic pattern becomes more random and bursty, the efficiency of the technique decreases because of the increasing waste of channel bandwidth.

Demand Assigned Multiple Access (DAMA) and Split-Channel Reservation Multiple Access (SRMA)

The second scenario is where each user transmits a request to the central station with the amount of intended data to be sent. Successfully received requests are then scheduled by the central station and serviced sequentially during the protocol cycle. The request channel can be accessed using either random access or fixed assignment techniques. Demand Assigned Multiple Access (DAMA) is the term that describes the set of protocols that employs the technique.

Split-Channel Reservation Multiple Access (SRMA) [3; 13] is a well-defined type of demand-assigned protocol that is centrally controlled. The central station initiates the protocol cycle as done by Polling, but the request channel is contended for as defined by DAMA. The successful requests are then scheduled by giving each host access to the total bandwidth of the channel.

SRMA protocol combines the strengths of both fixed assignment and random access techniques. Firstly, the protocol with its random access in the request channel maximizes channel efficiency for bursty traffic. Secondly, it increases the channel throughput of the access network by giving each successful user access to the total bandwidth of the channel at the expense of some added delay.

With SRMA, all the hosts are assumed to be in range of the central station. The central station initiates the transmission cycle and governs the sequence of host transmissions. Therefore, no hosts are allowed to transmit out of turn, which prohibits a potential hidden-terminal to cause a transmission collision other than that during the random access.

2.3 APRS as Access Network

Having considered the available MAC techniques and the different aspects of access networks that each address, the section discusses APRS's network characteristics and classifies it in order to determine the most suitable MAC candidate for improving APRS.

2.3.1 APRS Network Characteristics

The subsection gives an overview of the APRS network characteristics.

Network Flexibility

APRS's main advantage is its network flexibility and the accompanying characteristics. Hosts can easily be added to the network without having any *a priori* knowledge of the network topology. It is the result of the decentralized routing responsibility amongst the APRS hosts, which keeps the required knowledge of the network dynamics by each individual host to a minimum.

Repeaters and the Hidden Terminal Syndrome

Each host has the potential to function as a repeater of packets by using MAC-layer forwarding. The MAC layer marks the packet for repeating and each consecutive user's MAC layer retransmits the packet. The retransmission causes a flooding of the network with packets, which requires every host to be able to detect previously repeated packets and prevent their repetitive retransmission.

In addition, each host is set up according to the range of its transmission footprint [14]. The first set of transmission ranges includes mobile stations and home stations that have the smaller transmission range. When the hosts need to send data further than its local network, they make use of the next level of hosts, called digipeaters. Digipeaters are dedicated repeaters that can reach all the hosts in the local network.

If the general digipeaters do not provide the necessary coverage, WIDE-digipeaters can be used. The repeaters with their high transmission power interconnect neighbouring local APRS networks that are inaccessible to local digipeaters.

The Terminal Node Controller (TNC) of each APRS host uses the radio transceiver to sense the channel before transmitting. The presence of repeaters in APRS enables the network to address the Hidden Terminal Syndrome (HTS). As seen in §2.2.2, HTS is prevalent in especially wireless networks where not all the hosts are in range of each other. The hidden-terminal requires another in-range host to give an indication of the busy channel. The ARPS digipeaters, with their larger transmission range, repeat the successful transmission of a host, which is overheard by the out-of-range host. The out-of-range host sensing the repeated transmission then backs off and retries transmission only after the random waiting period.

Offered Services

The network supports both unicasting and multicasting modes, which is used for offering different services. Any set of hosts can periodically be requested for their accumulated data or routing information. The data can include information such as position coordinates and sensor data that enable the network to collectively monitor the area in which they are deployed. Unicasting is used for peer-to-peer communication that supports a textual message exchange service.

2.3.2 Classification of APRS Network

Having considered the APRS characteristics, it is concluded that the network consists of potentially large networks with graphically distributed hosts. APRS is characterized by high mobility as with Ad-Hoc networks, which could change both the network population and topology. The mobility adds to the random and unpredictable nature of the packet traffic.

In networks that exhibit high mobility and have a random traffic nature, the hosts are classified [3] as bursty sources of traffic. As seen in the previous sections, a network consisting of bursty hosts is best implemented using random access based MAC layer protocols.

2.4 Improving APRS

The section classifies the discussed characteristics with regard to their relevance to the two APRS limited performance indicators. The grouping will aid the identification of the appropriate MAC technique substitute for APRS.

2.4.1 Channel Throughput (CT)

As shown earlier, the channel throughput is limited by the MAC layer protocol and its efficiency in utilizing the bandwidth. The utilization in turn depends upon the network topology and characteristics as discussed earlier.

In APRS, the most basic of the MAC techniques, ALOHA, is used with its low maximum CT of 18%. ALOHA is chosen in order to keep the complexity of the hosts to a minimum and to support the flexibility associated with the APRS network.

The first objective is to find a suitable MAC protocol for APRS that will increase the throughput and then to determine to what extend the APRS requirements can still be met. The requirements include the numbers of hosts that can be supported, the traffic input rate of hosts on the network and most importantly, how it addresses the HTS problem.

2.4.2 Reliability

The other key performance indicator that is closely related to the throughput is the reliability of the protocol, which includes the guarantees on packet-delivery and the indication of collisions.

In APRS, the data is encapsulated in a *frame*, which is just the term used for *packet* in the AX.25 specification. In the remainder of the document, the term "packet" will be used to refer to packets in general and "frame" to refer to the APRS specific packet structure. APRS uses the AX.25 Unnumbered Information (UI) Frame implementation, which includes a frame sequence check (FCS) field that is a cyclic redundancy check (CRC) computed by the Data Link Layer. The CRC enables the receiver to determine the presence of transmission errors.

The MAC layer does not provide acknowledgements of received frames and therefore no automatic retransmission of frames could be implemented. The only reliability provided is that of the redundant transmission of frames, which is governed by a decay algorithm. A new frame is transmitted immediately and then the host waits for 20 seconds before it retransmits the frame. After every transmission the waiting-duration is doubled. After six transmissions, the 20-minute mark is reached and then the wait duration is changed to 10 minutes times the amount of digipeaters in the specified (UNPROTO) path. Therefore, the channel traffic governed according to the distance it is intended to travel.

The objective is to increase the reliability by adding positive acknowledgements, which provides a platform for implementing MAC Layer retransmission of collided packets.

2.5 MAC Protocol Substitution

Having discussed the APRS network aspects, the objective is now to choose the most appropriate MAC technique that will improve the ARPS limitations. The choice is mainly governed by the trade-off between improving APRS and keeping the modifications to the present network to a minimum.

2.5.1 Comparison of protocols

The aspects surrounding MAC protocols are now compared in order to determine the appropriate choice for the given APRS network specification. The specification includes the assumption of a bandwidth of 1200 bps and a local APRS network of approximately 50 bursty hosts.

ALOHA has the lowest CT of all the random access techniques. A good suggestion would be to use Slotted-ALOHA (S-ALOHA) and increase the CT by 100%. The result is already a satisfactory improvement, but the protocol does not increase the reliability of packet delivery. Adding a positive acknowledgement (ACK) mechanism to ALOHA the per-packet acknowledgements would double the channel traffic. The increased traffic would greatly reduce the CT performance according to [4].

An even better throughput performance is obtained by using CSMA or CSMA/CA, which gives a average CT of up to 60% when evaluated with the APRS network parameters. The throughput amounts to an increase of 450% relative to the throughput of ALOHA. The main problem is that both of the protocols suffer from the Hidden Terminal Syndrome and therefore would not support the present APRS network topology.

MACA has a comparable channel throughput with that of CSMA, providing a potential increase on ARPS throughput by a factor 4. MACA also solves the HTS problem to a large extent, being the first of the MAC protocols addressing HTS so far. Hidden-terminal support is a prerequisite to accurately analyze the performance of the network and support the APRS network topology and functionality. MACA can also support positive acknowledgements in order to provide increased reliability in the APRS network.

Lastly, there is polling and SRMA that are part of the centrally controlled assignment techniques. The random nature and low duty cycle of the APRS channel traffic suggests that polling would result in a substantial bandwidth waste. The large number of potential graphically distributed hosts results in polling producing large delays per packet [3].

For SRMA, using the channel throughput analysis done by [13], the estimated through-

Table 2.2
Comparison of MAC techniques

Protocol	CT (%)	Delay	Reliability	HTS Support
ALOHA	18	Average	Average	N/A
S-ALOHA	36	Average	Average	N/A
CSMA	60	Average	Good	No
MACA	60	Good	Good	Yes
Polling	100	Bad	Excellent	N/A
SRMA	65	Good	Good	Yes

put of SRMA evaluated for 50 hosts at 1200bps baud is expected to be around 65%. The amount of throughput is an increase of 360% to that of ALOHA. Positive acknowledgements are inherently part of SRMA's protocol-cycle design and provide the desired foundation for reliable transmission. SRMA can also reduce the HTS problem making use of its central station that require all hosts to be in its transmission range and governs the transmission sequence.

This concludes the comparison of the MAC candidate protocols. Table 2.2 gives a comparison summary of the most important aspects.

2.5.2 Chosen MAC Substitution

Considering the above comparison of MAC layer protocols, only MACA and SRMA has the ability to address all three aspects considered important with regard to the APRS network. Firstly, both protocols increase the channel throughput performance of ALOHA substantially. Secondly, the protocols have a form of reliability with positive acknowledgements, which is more than APRS providing no indication of packet delivery. Lastly, both MACA and SRMA reduce the prevalence of the Hidden Terminal Syndrome inherently addressed by APRS.

Further consideration suggested the comparison between the only major difference between MACA and SRMA:

- MACA is a sender-initiated protocol and
- SRMA is a receiver-initiated protocol.

In sender-initiated approaches the sender informs the receiver of its intended transmission. If the receiver accepts the request, the receiver then uses positive-acknowledgements (ACK-based) to indicate the arrival of every successful packet.

In receiver-initiated approaches, the receiver requests the sender for certain data. The sender then transmits the requested data in packets, where the receiver responds with negative-acknowledgements (NAK-based) only for packets that did not arrive.

The author in [15] provides a quantitive analysis of the superiority of receiver-initiated approaches over sender-initiated approaches. A throughput performance comparison is done for large-scale networks with up to a thousand hosts participating in a multicast group. The analysis considered moving the burden of providing reliable transfer from the sender (ACK-based) to the receiver by making use of negative acknowledgements (NAK-based). The analysis indicated that receiver-initiated approaches (NAK-based) outperform the sender-initiated approaches (ACK-based) on various multicast configurations.

The network architecture assumed in the analysis resembles that of APRS, which also consist of a potentially large-scale network and high network host-population. From the analysis it is concluded that the receiver-initiated protocol SRMA, when implemented with NAKs, would outperform the ACK-based MACA for the purpose of improving the relevant APRS limitations.

The effect of the difference between ACK-based and NAK-based protocols on the throughput performance is not part of the scope of the thesis. The superior performance of SRMA if it would be implemented with NAKs is sufficient to conclude that SRMA should be chosen above MACA for the MAC substitution in APRS. Previous analyses of SRMA [3; 13] considered ACK-based SRMA, which is also assumed for the purpose of the thesis.

2.6 Conclusion

The chapter has introduced the concept of access networks and the important aspects that needed to be considered in order to address the limitations of APRS. An overview of MAC techniques was given, which described the set of access-network aspects that are addressed by each. The APRS access network characteristics that need to be improved were identified and the most appropriate MAC technique substitute with regard to the objectives was chosen.

It is concluded that the choice of SRMA as MAC substitute will provide the necessary improvements on the throughput and reliability of the present APRS. The next chapter continues by defining the proposed S-APRS network topology and describing the integration of SRMA with the APRS protocol stack.

Chapter 3

Network Topology and Protocol Integration

The chapter introduces the proposed S-APRS network topology that is considered for the performance evaluation in the thesis. The integration of SRMA with the APRS protocol stack is done and the protocol upload and download cycles are described.

3.1 Proposed S-APRS Network

The intended performance evaluation of the S-ARPS protocol requires that certain assumptions be made with regard to the specific network topology to consider. The network consists of four different types of hosts that are distinguished according to their functionality and position in the communication path. They are, in sequence:

1. Control Station (CS)
2. Low Earth Orbit (LEO) satellite
3. Base Station (BS)
4. Field Station (FS)

The control station, at the one end of the path, sends messages or requests for data intended for the field stations at the other end of the path. The messages are forwarded via the LEO satellite to a set of base stations on the ground that broadcasts the messages to all the field stations in its local network. The field stations can then respond to the control station message, if it is required, by sending their response via the same return path. See figure Figure 3.1 for a representation of the network.

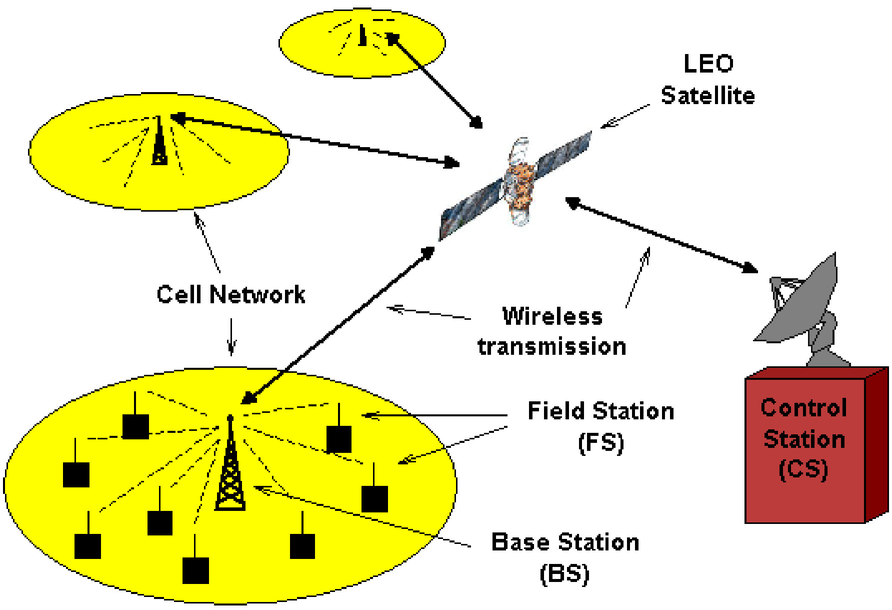


Figure 3.1: Proposed APRS network with SRMA

The network scalability is evident from the proposed topology. There are two distinct components, which are a number of hosts in a local network and a central station functioning as the access point. The network is divided into two levels, the Cell Network (CN) and the Base Network (BN).

3.1.1 Cell Network

The Cell Network consists of a large set of field stations that share a single transmission channel provided by the base station. The base station functions as the access point of the cell network to the rest of the APRS network.

The main function of a field station is to accumulate data from its sensors and support a text-based messaging service, provided it is equipped with an appropriate user interface. On request from the base station, it broadcasts its data and position coordinates as frames on the local cell network. The frames are stored by the base station for future requests by the LEO satellite or retransmitted to enable all the local field stations to receive the data.

Routing information can be disseminated throughout the entire cell network. Each field station uses the path information included in frames to update their routing information and dynamically supports variation in the network population. The cell network supports a bandwidth for 1200 bps as used in the present APRS.

Table 3.1
THE APRS PROTOCOL STACK

Layer	Function	Implementation
7	Application	APRS
4	Transport	N/A
3	Network	N/A
2	Data Link (MAC)	ALOHA
1	Physical Link	RF Link

3.1.2 Base Network

The second level, the base network, is a scaled version of the cell network. The set of base stations of all the cell networks each contends for access on the single transmission channel provided by the LEO satellite.

The satellite now functions as the access point that governs the access to the control station for all the base stations in its footprint. The satellite can also interconnect neighbouring CNs by repeating frames that it receives to all in its broadcasting footprint.

LEO satellites presently provide channel bandwidth up to 1 Mbps. The performance evaluation will consider the effect of bandwidth on the protocol performance. It will give an indication of the potential of the viability of increasing the base network channel bandwidth above the current APRS rate of 1200 bps.

3.2 APRS Protocol Standard

The protocol stack of APRS needs to be considered in order to conceptualize the integration process of the SRMA MAC layer. A simplified Open Systems Interconnect (OSI) model is used that only includes the physical, data-link, network, transport and application layer to describe the layered functionality.

3.2.1 The Protocol Stack

Table 3.1 represents the simplified OSI model of the APRS protocol stack. The first layer, the physical layer, controls the interface to the physical radio transmitter and receiver. It hides the characteristics of different radios from the higher layers.

Next is the data-link layer that consists of the MAC sub-layer that presently implements ALOHA and a simplified version of the amateur radio AX.25 protocol. It uses the AX.25

	Flag	Destination Address	Source Address	Digipeater Addresses 0-8	Control Field (UI)	Protocol ID	INFORMATION FIELD	FCS	Flag
Bytes:	1	7	7	0-56	1	1	1-256	2	1

Figure 3.2: AX.25 UI frame format

Unnumbered Information (UI) frames that presently do not implement acknowledgements of packet delivery. With ALOHA at MAC layer, there is no collision avoidance or MAC layer retransmission implemented.

With the lack of guarantees present at the data-link layer, it would be expected of the protocol stack to have either a network or a transport layer that provides the reliability. Both the layers are not implemented as part of the standard. The only reliability is provided with the redundant transmission of data by the data-link layer by either using the decay algorithm or fixed rate.

The decay algorithm transmits a new packet when generated and then retransmits it k seconds later. Each time the amount k is doubled until a limit is reached, and then continued at that rate. The net cycle time is the time within which a user will have heard, at least once, all the hosts that are in range.

The fixed rate method transmits every new frame and then retransmit it k seconds later. The cycle is repeated at k seconds time intervals for a limited amount of times and then stopped.

The last layer is the application layer of the protocol stack that provides the graphical user interface. The application includes the maps displaying the position of stations and the interface to send textual messages.

3.2.2 The APRS UI Frame Structure

The AX.25 UI frames-format used in APRS is shown in 3.2. The majority of the frame structure is standard. The digipeater addresses field contains the digipeater path for the frame to a maximum of eight digipeaters. The control field is fixed and indicates the UI frame type. The Protocol ID field is set to indicate that there is no layer-3 protocol implementation. The information field contains the actual data to be sent with a limit of up to 256 bytes. The Frame Check Sequence (FCS) field is a cyclic redundancy check done on the frame to enable hosts to identify transmission errors. The total maximum of number of bytes is 332 per frame.

3.3 SRMA Protocol Description

The SRMA protocol cycle is now considered in detail in order to identify the important aspects with regard to the integration with APRS. It will enable us to determine how smooth the transition will be and to what extend the APRS functionality will be retained.

3.3.1 General Implementation Aspects

As indicated by its name, the single channel is split up into two separate channels. The two channels are called the *request* channel and the *message* channel. The request channel uses S-ALOHA random access to enable multiple hosts to contend for the single channel. The successful requests are then serviced by allowing each host to sequentially use the collision free message channel to transmit the data.

The channels could be divided using either frequency division multiplexing (FDM) or time division multiplexing (TDM). The choice between the two depends mainly upon the complexity of the transceiver hardware and the protocol delay requirements. With FDM the transceiver hardware is more complex in order to provide the two frequency channels than its single frequency TDM counterpart. On the other hand, the protocol delay incurred by FDM is less, because both channels operate simultaneously, whereas TDM has to sequentially service each channel.

3.3.2 Research-Specific Implementation

The thesis considers the application specific implementation of SRMA in [13]. The study was done on a single LEO satellite functioning as the access point, which broadcasts to multiple hosts on the earth. TDM was utilized for the channel division mainly for the prospect of minimizing transceiver complexity. The minimal complexity contributes to the requirement for the minimum modifications to the APRS network architecture. Figure 3.3 shows the SRMA protocol cycle implemented with TDM.

SRMA has two separate protocol cycles, namely an *upload* and *download* cycle. The upload cycle refers to the flow of data from the hosts to the access point. For the proposed network topology in the thesis, the data flow is either from field stations to base station access points or the base stations to the satellite access point. The download cycle describes the flow of data packets in the reverse path.

In [13] both cycles implement multiple-access, but the thesis only considers the upload cycle for multiple-access. The reason is that with the download cycle, multiple-access

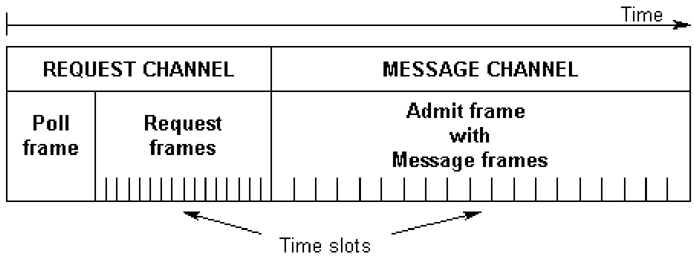


Figure 3.3: Time division multiplexed SRMA protocol cycle

is unnecessarily wasteful with regard to its throughput efficiency. Wireless transmission is inherently broadcast with all the hosts in the footprint receiving the data frames and accepting or discarding them based on their destination address. Therefore to require the hosts to first make a successful request in order to receive the data to be downloaded results in a decreased channel throughput.

3.4 SRMA-APRS Integration

The SRMA-APRS (S-APRS) integrated protocol is formally specified. The S-APRS upload and download cycles are described and the frame structure defined.

3.4.1 S-APRS Protocol Cycle

The upload cycle is the only cycle that will be considered for multiple-access in the performance evaluation. The download cycle is described only for the sake of a complete protocol description.

Upload Cycle

The upload cycle starts with the first timeslot during which the access point broadcasts the *poll* frame to all the hosts in its footprint. The frame contains the type of data that is requested by the satellite, which includes accumulated data, routing information or position coordinates. Every host that receives the frame, transmit a *request* frame to the satellite if they are required to respond. The transmission of request frames and the resulting multiple-access of the channel is governed by S-ALOHA. After a fixed duration, the satellite ends the request channel by receiving no more requests and scheduling the received requests.

The message channel starts by the satellite transmitting an *admit* frame destined for the host of the first request. With the channel being contention free, the host responds by transmitting a limited amount of data frames specified in the admit frame utilizing the total channel bandwidth. After the satellite receives the last data frame, it transmits an *acknowledge* frame, which includes the sequence numbers of all the received frames. If by any reason, all the data frames were not received, the sending host will during the next cycle retransmit them. The process continues until the request queue is finished and then the cycle restarts.

Download Cycle

The satellite initiates the download by transmitting a *data-leader* frame, which contains the amount of frames to be downloaded. A stream of *data* frames succeeds the data-leader frame until the last data is transmitted in the *data-end* frame. At the end the hosts can be queried for acknowledgements of the frames received since the previous query. The request for acknowledgements can be done as part of an upload cycle.

S-ARPS MAC overhead

All the frames that are sent by the S-APRS MAC layer that does not include the actual data to be sent are classified as the MAC transmission overhead. In the upload cycle, the poll, request, admit and acknowledge frames are therefore classified as part of the MAC transmission overhead. The download cycle only has the data-leader frame that is classified as overhead.

3.4.2 Integrated Frame Structure

In order to minimize the change required by the existing APRS implementation, the AX.25 UI frame structure is used as the basic frame structure. By making a redefinition of a single field, the *S-APRS frame* is created.

Research was done regarding the importance of the UI frame fields. It was necessary to determine which modifications to the frame would have the least impact on the present APRS network. Recent commentary at the Tucson Amateur Packet Radio (TAPR) website suggested the limiting of digipeaters in the path to avoid unnecessary congestion of the network. A maximum of 3 digipeater addresses was recommended, which produces excess bytes in the field. The digipeater design limitation is also incorporated in the most recent AX.25 protocol specification [16].

	Flag	Destination Address	Source Address	Digipeater Addresses 0-3	SMAC Type	SMAC Data	Control Field (UI)	Protocol ID	INFORMATION FIELD	FCS	Flag
Bytes:	1	7	7	0-21	1	6	1	1	1-256	2	1

Figure 3.4: The integrated UI frame structure

The recommendation led to the redefinition of the "digipeater addresses" field. Three digipeater addresses is provided for amounting to 21 bytes. The S-APRS MAC control information requires 7 bytes. The remaining 28 bytes space is discarded. The result is a total of 48 bytes overhead per S-ARPS overhead frame, bringing the maximum frame length to 304 (48 + 256) bytes.

The S-APRS frames that include data, i.e. *data* and *data-end*, utilizes the information field for data that has a maximum of 256 bytes as specified by AX.25 UI frames. See Figure 3.4 for the new frame structure.

3.4.3 Conclusion

The chapter introduced the research-specific S-APRS network topology. The SRMA protocol was described and main aspects of integration was identified. The integration was done that produced the S-APRS protocol specification that uses a modified APRS frame structure.

It is concluded that the integration process satisfies the integration objective, which was to successfully substitute the MAC layer with the minimal required changes to the existing APRS specification. Now that the protocol specification is done, the protocol model and an implementation will be developed in the next chapter.

Chapter 4

Protocol Modeling and Emulation

The chapter contains the modeling and implementation of the proposed S-APRS protocol. It is done in order to achieve a largely platform-independent implementation, so that the software can be retargeted for any platform with the minimum of changes. It also results in the implementation overhead being quickly and effectively measured and optimized.

4.1 Communication Protocol Modeling

The section first introduces the concepts of traditional communication protocol modeling as specified by the Open Systems Interconnect (OSI). An overview of the Unified Modeling Language (UML) and its notation is given in order to understand the object-orientated approach of UML and its diagrams contained in the thesis. Lastly, introducing a graphical model that offers a more structured object-orientated model and the concepts required for code-generation, extends the traditional model. For the remainder of the document, the term *protocol* is used to refer to communication protocol.

4.1.1 Traditional Protocol Modeling

The traditional way of modeling protocols is specified by the Open Systems Interconnect (OSI) model [5] of a layered communication protocol stack. The specification includes the differentiation of services provided by each layer and the format of inter-layer communication. It offers a basic description of the external behaviour of a protocol that is necessary to conceptualize the functional dynamics of the protocol. An overview of the model is now given in order to introduce the first building block of the modeling process.

There are two types of communication between layers i.e. *layer-to-layer* and *peer-to-peer*.

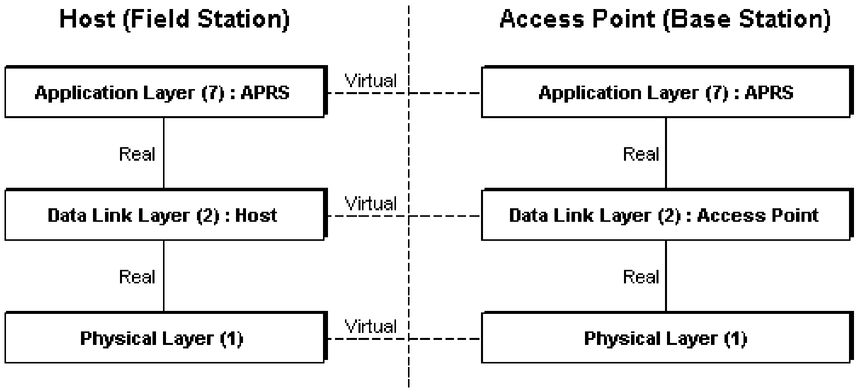


Figure 4.1: Conceptual model of APRS protocol stack

Layer-to-layer is a *real* message exchange between adjacent layers in the same host and peer-to-peer is a *virtual* exchange between layers of the same service in two different hosts. Figure 4.1 is a representation of the S-APRS proposed protocol stack where the peers are represented by the hosts on the ground and the access point. In order to have a basic understanding of the traditional way of protocol modeling, three fundamental concepts are introduced, namely *services*, *service primitives* and *service interfaces*.

Services

In the context of protocols, services refer to the set of functions that a *service provider* offers a *service user*. The services are the functions provided by each layer as specified by the OSI model where a layer can function as both a service producer and user. A lower layer provider always offers a service to a higher layer user. Access to the required service is gained through *service access points* (SAP).

Service Primitives

The service primitives are the messages that are communicated between two layers across the SAP. For the purpose of our network, the category of an *unconfirmed service* [5] is implemented.

It uses two primitives, namely *request* and *indication*. The service user uses the request primitive to communicate with the service provider and it in turn uses the indication. See Figure 4.2 for a diagrammatic illustration of the services and service primitives.

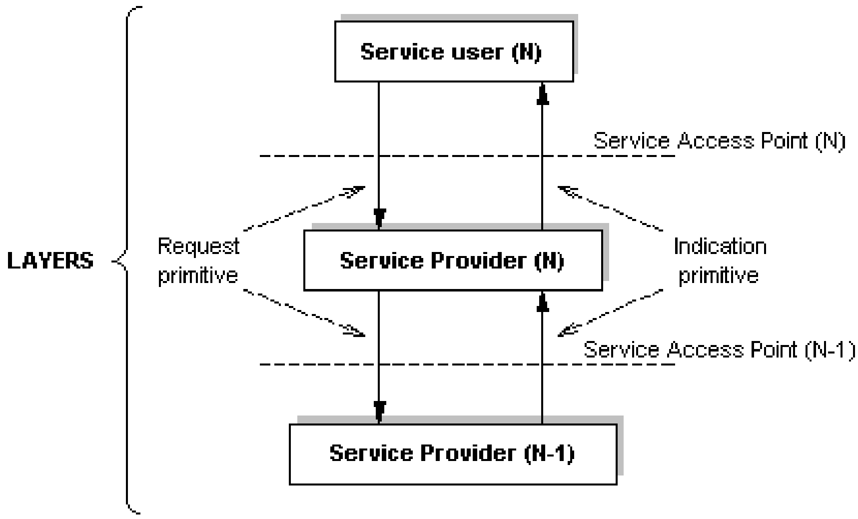


Figure 4.2: Components of communication protocol modeling

Service Interfaces

In order for the service layers to communicate with the primitives, a mutual interface is constructed. The interface specifies the type of messages that cross the SAP and the format of the information to be exchanged. The format describes the encapsulated packet created by adding a header to the data received from higher layers as the packet descends through the protocol stack. The headers are removed in reverse order, as the packet ascends the peer protocol stack, which results in the peer protocol layer receiving the information in the appropriate format.

4.1.2 Unified Modeling Language (UML)

Overview

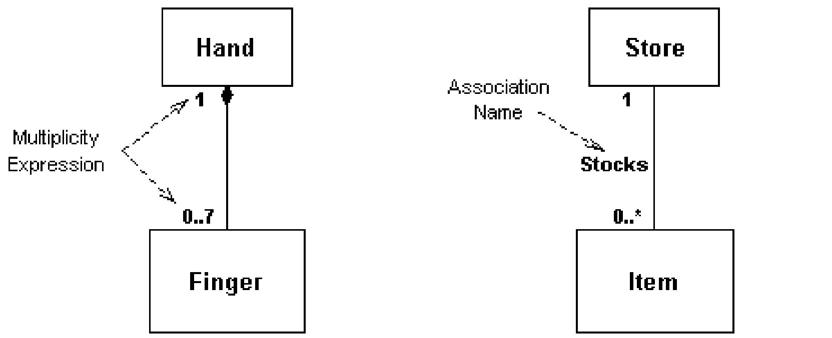
UML [17] is a widely used standard notation for modeling systems using object-orientated concepts. Its industry acceptance mainly stems from its flexibility in that it does not specify a fixed development process, but it leaves the opportunity for proprietary development of process specifications.

In the following sections, fundamental UML notation is used to describe the protocol model. The basic notation of *class diagrams* is briefly introduced, which implement the concepts of *association*, *aggregation*, and *multiplicity expressions*.

Notation Description

The class diagram contains classes defined according to object-orientated concepts and are represented by rectangular blocks. Each class can be connected to another using either association or aggregation. Association describes a relationship between concepts that indicate some meaningful and interesting connection. Aggregation describes whole-part relationships where the one class forms a part of the *composition* class. Multiplicity expressions are added to define how many instances of a type can be associated with one other type at a particular moment. See the illustration in Figure 4.3.

The multiplicity expressions "1", "0..7" and "0..*" in the figure respectively means one, zero-to-seven and zero-to-many instances of a class. The aggregation connection has a diamond-end at the class that represents the "whole" in the relationship.



(a) Aggregation (b) Association

Figure 4.3: UML Class Diagrams

4.1.3 Extended Graphical Model

Traditional protocol modeling is not sufficient for producing a graphical model from which an efficient code implementation can be created. The reason is that traditional model only describes the functionality and external behaviour of the layers. The model is unable to support intra-layer modularity and does not include the implementation concepts needed for efficient code-generation. The concepts include handling queues, variable storage and intra-layer communication.

The extended graphical model [18; 19] to be used for S-APRS is now introduced as defined by the author in [20]. The extended model consists of three main classes, namely a *System*, an *Entity* and a *Message*. Figure 4.4 illustrates the relationship between the mains classes in a class diagram.

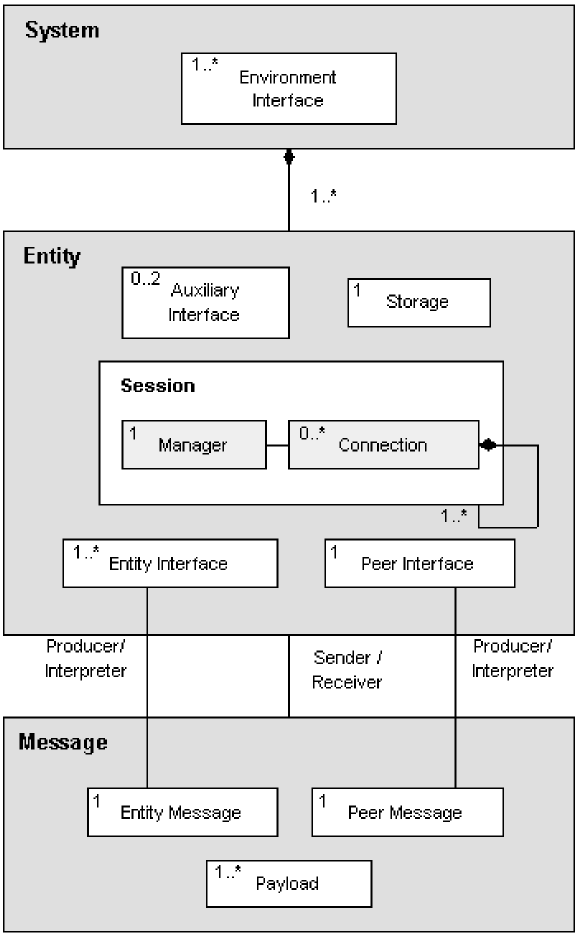


Figure 4.4: Conceptual UML classes for protocol system

System

The system class contains the Environment Interface (EI) and a number of entities. The EI represents the service access point mentioned in §4.1.1 and the entities generally represent the protocol layers of one protocol system. The EI specifies the interface to both the application layer and the physical layer.

Entity

The entities, that form part of the system, are the layers of the protocol stack. The layers each implement a service that generates service primitives corresponding to the layer functionality. There are several subclasses to consider in the entity. Firstly, the Auxiliary Interface specifies the communication between entities in the same layer, mainly used for management purposes. The Storage component keeps the internal state of the entity in memory as the cycle progresses. The Entity Interface handles the exchange of messages between entities in the same system, but in different layers, whereas the Peer Interface communicates with entities of the same layer in different systems. Lastly, the Session class handles the setup of multiple connections between peers.

Message

The message class contains two message specifications and the payload that accompanies either of them. The Entity Message is used for layer-to-layer communication and the Peer Message for peer-to-peer communication as described in §4.1.2. The payload accompanying each message consists of the data frame that is being manipulated.

4.2 S-APRS Protocol Modeling

In the section, the development process is described that is followed in creating the model and implementation of the S-APRS protocol. The process consists of the creation of the following components in chronological order:

1. Class Diagrams,
2. Interaction Diagrams,
3. Statechart Diagrams.

The set of diagrams provides a model of the protocol that is adequately specified for efficiently generating code for the protocol implementation. An overview of the each set of diagrams is now considered in order to get a basic understanding of the development process. Understanding the development process helps to produce a rapid implementation for a different platform.

4.2.1 Class Diagrams

The first step is to identify the classes in the object-orientated model and develop the class diagrams with UML according to the model defined in §4.1.3. Only the classes that are relevant to the protocol requirements are implemented.

S-APRS Host and Access Point Requirements

In the S-APRS network, there are two types of peers, namely the general host and the access point. Both the host and access point require a unique S-APRS layer implementation with regard to their different functionality, with the rest of the protocol stack implementing the original APRS.

The development of a mutual protocol stack implementation is suggested that can operate as either type of peer. When the system is started up, the user can then configure the implementation to either run as a general host of the access point. The advantage is that the protocol will be contained in a single software implementation, which is easily reconfigurable.

Another important distinction is made with regard to the protocol cycle. The upload and download cycles are separated in order to achieve modularity in the protocol model design. The modularity will speed up the implementation development process.

The upload cycle refers to the flow of data from the multiple hosts to the single access point. The download cycle is implemented as a broadcast channel where data flows from the access point to the multiple hosts in both the cell network and the base network.

Classes and Functionality

The classes of the protocol model are now modified and described according to the above-mentioned requirements of S-APRS. Refer to Figure 4.5 for the representation of the class diagram.

The System class contains the Environment Interfaces that are defined for both the application and hardware SAP. The Data-Link and Application SAP (DASAP) class specifies the interface to the S-APRS Application, which contains the related service primitives. The Data-Link and Physical Layer SAP (DPSAP) class interfaces with the physical layer with its own set of primitives.

There are three Entities implemented as the layers of the protocol stack. The Application Layer provides the user interface. The Data-Link (DL) Layer contains the S-APRS MAC implementation, which in turn interfaces to the Physical-Link (PL) Layer.

The data-link layer implementations are separately done for the general host and access point for reasons already discussed. The classes include the relevant service primitives and peer-to-peer interfaces used for both layer-to-layer and peer-to-peer communication.

The Message class specifies all the message formats and the payloads that constitute the

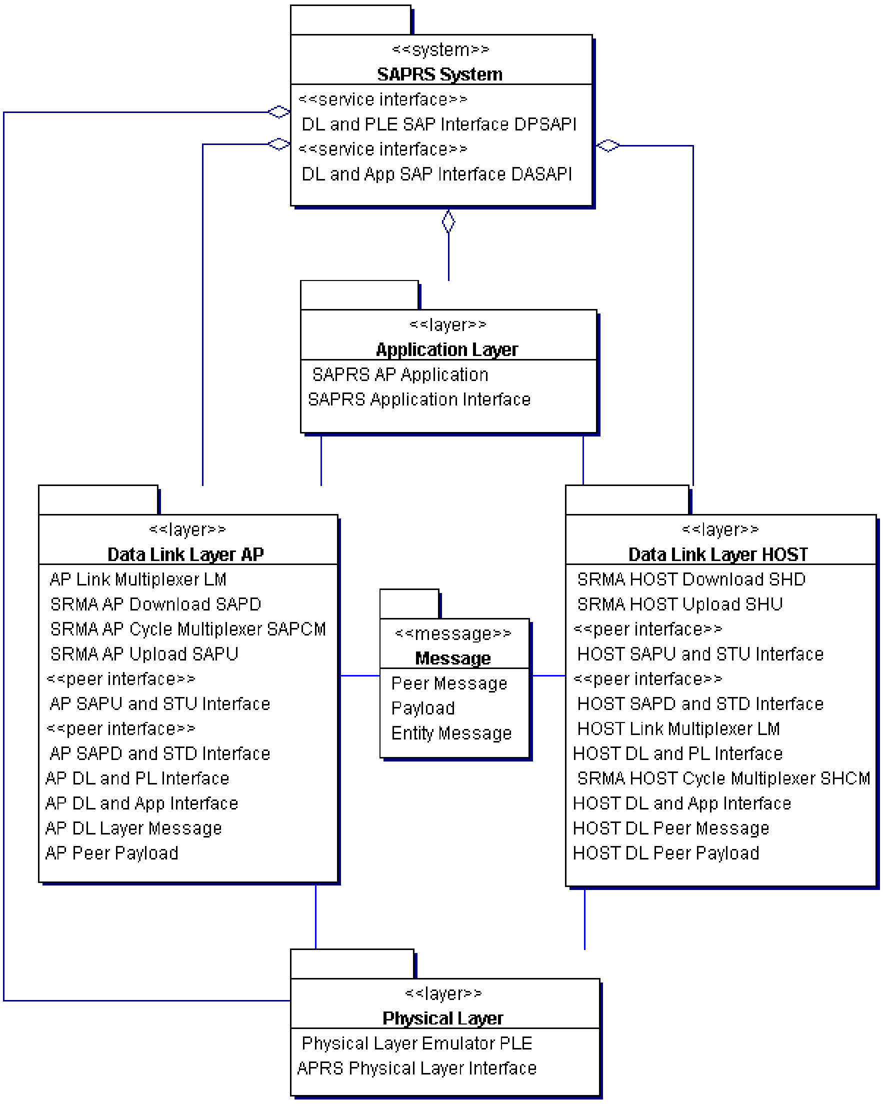


Figure 4.5: UML main class diagram of S-APRS

APRS UI frame. The peer messages are integrated with the UI frame as described in §3.4.2.

4.2.2 Sequence and Collaboration Diagrams

The next step is to model the interaction among the classes. UML specifies two types of interaction diagrams i.e. Sequence Diagrams and Collaboration Diagrams. Sequence diagrams focus on the time sequence of messages between classes. Collaboration diagrams describe more the connections between objects of classes and the messages using the connections.

Only the sequence diagrams are included in the thesis and chosen as the basis for the interaction description. The sequences of class interactions follow the protocol cycle sequence and are therefore necessary in order to derive the logical flow of the software implementation. The sequence diagrams are included in appendix A.

The conceptual representation of message connections provided by collaboration diagrams is of less significance for protocol modelling. If required by the developer, UML tools can be used to easily convert the sequence diagrams into collaboration diagrams. Appendix C contains a CD with the complete UML html pages generated by the UML tool. In the thesis, the "Together" software package was used as the UML tool.

4.2.3 Statechart Diagrams

Statechart diagrams contain the finite state machines (FSM), which is the final step in the model development process. It describes the processes, actions and activities of the protocol system. The developer translates the state machine into the implementation code.

The state machine is represented by a number of states in which the system resides until the time of a transition to another state. The transition is initiated by an *event*, which depending upon the actual state, determines the next state.

With regard to describing communication protocols, extended finite state machines (EFSM) offer a few additions to FSMs in three respects:

- They can maintain *internal variables* (sequence counters, flags).
- *Timers* are supported that at expiration act as triggers to states.
- *Internal queues* are supported that schedule the servicing of some signal at a later stage.

The additions offered by EFSMs are implemented by using the System Description Language (SDL). SDL supports a detailed description of events, transitions and other internal behaviour necessary for a protocol description. The SDL diagrams of the S-APRS protocol stack are included in appendix B.

4.2.4 Summary

In the section, the process of developing a model for the S-APRS protocol has been presented. It has laid both the foundation for the object-orientated modeling of the system

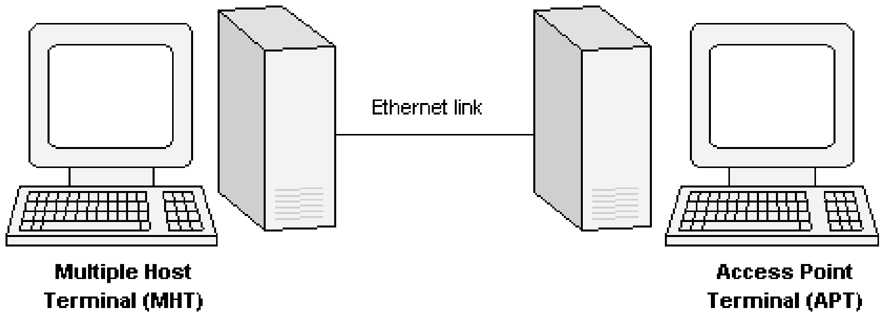


Figure 4.6: Ethernet Emulation Network

and the extended description of the system in order to produce the largely platform-independent protocol implementation.

4.3 Protocol Implementation

The section describes the software protocol implementation created using the developed protocol model and considers the resulting implementation challenges.

4.3.1 Implementation Aspects

During the time of implementation, no radio hardware was available that could provide the wireless physical link. Therefore, the decision was made to implement the protocol using a physical layer emulator.

The physical-link emulator is a software implementation that emulates the wireless link behaviour while using a different physical link. The physical link consists of a 100baseT (100 Mbps) Ethernet connection between two Pentium processors that run the S-APRS software implementation.

In order to measure the protocol overhead delay introduced by S-APRS, only two general hosts and an access point are needed to produce the multiple-access that would result in a true representation of the protocol dynamics. The general hosts implementations run on the Multiple Host Terminal (MHT) and connects to the Access Point Terminal (APT) via the Ethernet link. See Figure 4.6.

The remainder of the section describes the behaviour of the wireless link that needs to be implemented by the physical layer emulator. The behaviour includes link properties such as the transmission rate, propagation delay, the emulation of packet collisions and host synchronization on the network.

Transmission and Propagation Delay

The transmission delay is a function of the bandwidth of the transceiver. As mentioned previously, 1200 bps baud rate is used in the S-APRS cell network. The viability of increasing the bandwidth for the base network with the LEO satellite access point will be considered.

The transmission delay has to be determined for both S-ARPS data and overhead frames. Using the definition of transmission delay introduced in §2.1.2, the transmission delay of an overhead frame of 48 bytes, T_{TO}, is determined from equation (2.1.3) as

$$T_{TO} = (48 \times 8)/1200 = 0.320\ s. \tag{4.3.1}$$

The transmission delay of a data and data-end frame with a maximum of 304 bytes, T_{Data}, is determined as

$$T_{Data} = (304 \times 8)/1200 = 2.027\ s. \tag{4.3.2}$$

To determine propagation delay, the maximum transmission distance between hosts needs to be estimated for both the cell and base network. For the cell network, it is assumed that the most powerful transmitter, which resides in the access point, determines the distance.

An average 450 MHz band amateur radio transmitter with a rated power of 100 Watts and an antenna with a height of 6 meters is assumed for the cell network. The parameters produce [21] an estimated maximum transmitter distance of 10 km. Substituting the values in equation (2.1.4) and using the free space propagation speed (c $= 3\times10^8\ m/s$), gives the maximum propagation delay for the cell network, $T_{PD\ CN}$, as

$$T_{PD\ CN} = (10 \times 1000)/(3 \times 10^8) = 33.333 \times 10^{-6}\ s. \tag{4.3.3}$$

For the base network the LEO satellite functions as the access point. An average distance for a LEO satellite of 800 km from the earth's surface is assumed with a minimum bandwidth of 1200 bps. The base network propagation delay, $T_{PD\ BN}$, is therefore calculated as

$$T_{PD\ BN} = (800 \times 1000)/(3 \times 10^8) = 2.667 \times 10^{-3}\ s. \tag{4.3.4}$$

Considering the estimated propagation and transmission delay values for both networks, it is clear that the propagation delays are very small compared to the transmission delays. Therefore, only for the physical layer emulator implementation, the effect of propagation is disregarded. Only the transmission delay is left, which is incorporated in the emulator implementation and is essential to produce the correct wireless link emulation.

Collision Emulation

The main design challenge in the emulation is producing a correct emulation of the multiple-access collisions. A collision occurs when any two hosts on a multiple-accessed single wireless channel transmit simultaneously. The potential of a collision implies that for a given time from the start of a transmission, the channel needs to be tested for a collision. The collision duration time, which is denoted as T_{Col}, is the sum of both the transmission delay, T_{TD}, and the propagation delay, T_{PD}.

The two delays need to be compared to that of an actual S-APRS network. The CN with its low baud rate and closely distributed hosts has a negligible T_{PD} and therefore T_{TD} dominates its T_{Col}. On the other hand, the BN with a much higher baud rate produces a small T_{TD} and a larger T_{PD}, which in turn is caused by the greater transmission distance to its LEO satellite access point.

The collision duration time for each of the three scenarios is now determined. The Ethernet link is a 2-meter twisted-pair, which has a medium propagation speed of 0.59 of the speed of light (c = 3×10^8 m/s). It uses 1500 byte frames with a 100 Mbps transmission speed, which produces a collision duration time of

$$\begin{aligned} T_{ColEthernet} &= \frac{1500 \times 8}{1 \times 10^9} + \frac{2}{0.59 \times 3 \times 10^8} \\ &= 12 \times 10^{-6} \quad s. \end{aligned} \tag{4.3.5}$$

For the CN, with a maximum frame length of 304 bytes, 1200 bps signal rate and a distance of 10 km, equation (2.1.3) and equation (2.1.4) is substituted and the collision time determined as

$$\begin{aligned} T_{ColCN} &= \frac{304 \times 8}{1200} + \frac{10000}{3 \times 10^8} \\ &= 2.027 \quad s. \end{aligned} \tag{4.3.6}$$

where TColCN is the collision duration time for the CN.

For the BN, with a frame length of 304 bytes, 1200 bps signal rate and an average distance of 800 km for the LEO satellite from the earth, results in a base network collision time of

$$\begin{aligned} T_{ColBN} &= \frac{304 \times 8}{1200} + \frac{800\ 000}{3 \times 10^8} \\ &= 25.333 \times 10^{-3} \quad s. \end{aligned} \tag{4.3.7}$$

Considering the values, it is evident that the channel baud rate has the dominant effect on the total collision time. It can also be seen that the Ethernet collision time is much less than the required time by the cell and base networks. It shows that the emulation needs to add a delay to the Ethernet collision time during transmission in order to emulate that of the cell and base network links.

The main parameter that would influence the accuracy of the collision emulation is the resolution and accuracy of the timer used to introduce the additional delay. Standard operating system timers provide a resolution of 10 ms, which is more than sufficient when considering the values of collision times calculated above.

Synchronization

The importance of timer resolution, apart from the collision emulation, is mainly driven by the need for synchronization amongst the hosts. It is vital for the functioning of S-ALOHA request channel in SRMA with its separate continuous time slots. The higher the resolution of the timer, the smaller the margin for error on the start and finishing times of the slots will be.

On the other hand, the higher resolution requires a higher frequency of timer interrupts. The higher frequency results in more wasted time because of the increased interrupt-overhead. The wasted time in turn produces an increase in the rate at which the timer looses time, resulting in decreased synchronisation of the overall system. The margin of error and the system synchronization requires a trade-off between the resolution of the timers and the frequency at which network resynchronization is done.

4.3.2 The Coded Implementation

The last step in the implementation process is the coding of the protocol. The implementation is coded in the C language for mainly two reasons. Firstly, present APRS implementations are done in C, which offers a good code framework for the integration process. Secondly, the C language is largely processor independent and is currently supported by various distributions of both the Linux and Windows operating systems. The implementation code is included on the CD in appendix C.

The S-APRS implementation is done on the open-source Linux 4.2.22 kernel using the General Computer Corporation (GCC) compiler. The platform was chosen mainly because of its open source status. The status enables the software developer to access low-level kernel libraries where protocol stack implementations reside. The accompanying licence also allows the developer to create proprietary implementations by modifying existing source code and contributing the work to the open-source community.

The implementation consists of three processes running simultaneously:

- The Physical-Link Server (PLS)
- The Physical-Link Emulator (PLE) and Data-Link (DL) Layer
- The S-APRS application

S-APRS Application

The application process for the access point is the initiator of both the upload and download cycles. See Figure 4.7 for a diagrammatic representation of the application process program flow.

It provides the user interface that enables the user to enter the data to be transmitted and to view received data. The data is passed to and from the data-link layer process through the Data-Link/Application Service Access Point (DASAP) interface §4.2.1. The DASAP specification provides a standard interface for further S-APRS application based development in order to offer additional services to the user. The application process for the host has the same functionality except for the initiation of upload and download cycles.

Physical Link Emulator (PLE) and Data-Link (DL) Layer

The data-link layer receives the cycle initiation request through the DASAP. The corresponding S-APRS overhead frames are then created and passed on to the physical link emulator process through the DPSAP interface.

The PLE receives the frames and transmits them at the emulated channel baud rate. It also services the server queue, which contains the frames received by the physical-link server process. The frames are checked to see if there has been a collision. The difference between the arrival times of consecutive frames are required to be more than the transmission delay of a single frame for it to be successfully received.

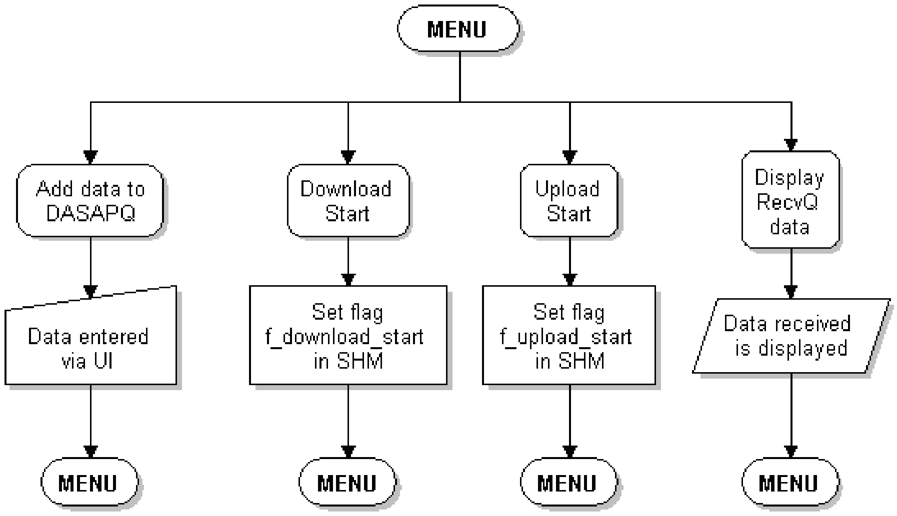


Figure 4.7: S-APRS Application process

The successful frame is passed on to the DL layer where the frame destination address is confirmed. From thereon the S-APRS MAC state machine handles the frame and passes received data onto the S-APRS application process via the DASAP.

As described in §4.2.1, a different implementation is done for both the host and access point. The distinction is made between the upload and download S-APRS states, which generate the different peer-to-peer messages. The only mutual functionality of the MAC state machine is the destination address validation. See Figure 4.8 and Figure 4.9 for the flowcharts of all three process implementations.

Physical Layer Server (PLS)

The server stores the received frames in the server queue and indicates their reception to the next process via the DPSAP. It also adds the arrival time to each frame to enable the PLE to determine channel collisions. The PLS process flow for the access point is added to the PLE & DL layer diagrams in Figure 4.8, which is the same used by the host.

Implementation Overhead

The implementation overhead measurement is discussed in the protocol performance-analysis in the next chapter. The S-APRS protocol cycle durations will be described and used to determine the durations that correspond to the total implementation overhead of a single cycle.

4.4 Conclusion

In the chapter the modeling and implementation of the S-APRS protocol was done. The protocol model resulted in the development of a largely platform-independent implementation, which can be retargeted for another platform with minimal changes. The implementation overhead will be evaluated as part of the protocol performance analysis in the next chapter.

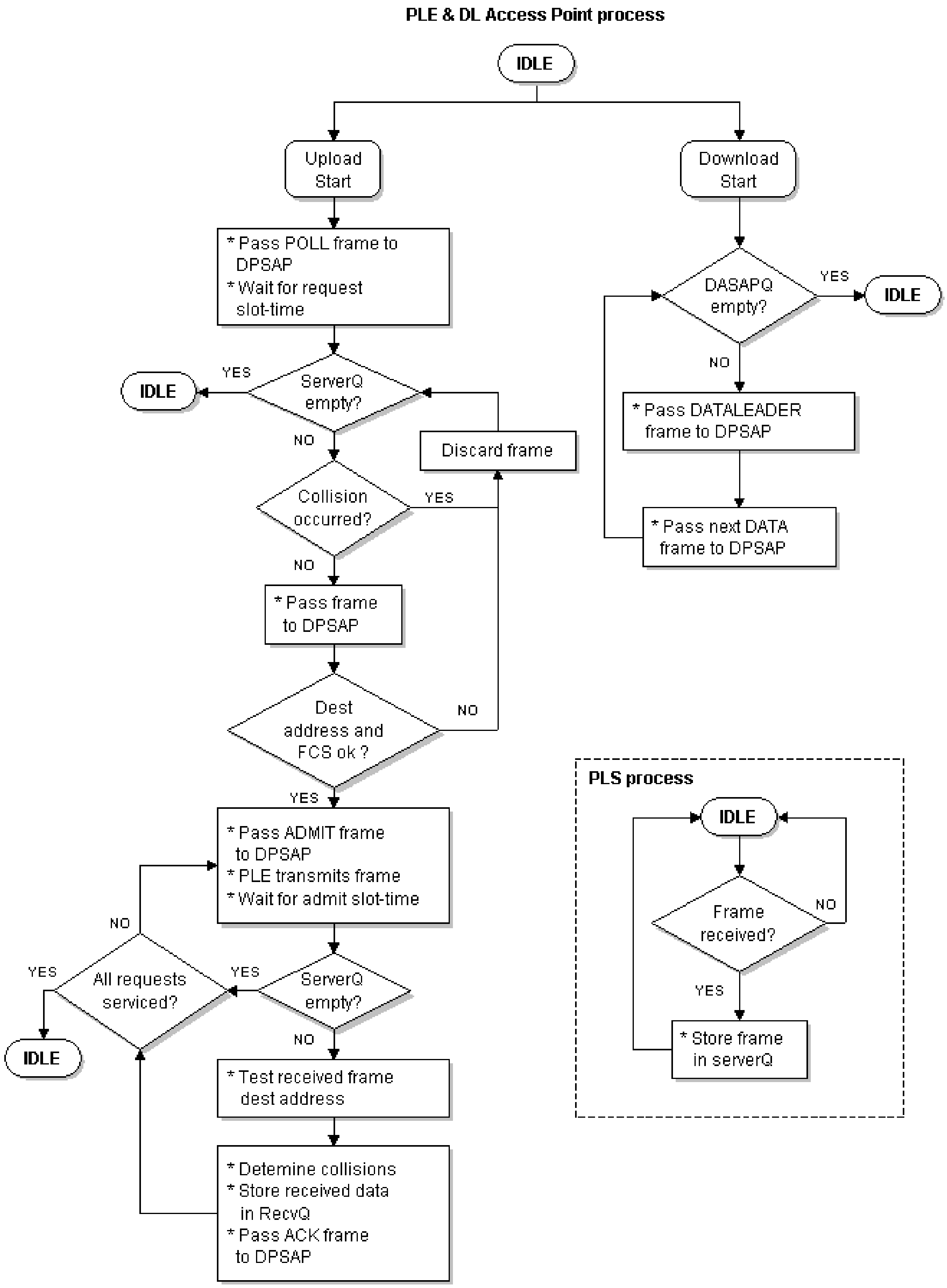


Figure 4.8: S-APRS PLE & DL Access Point and PLS process

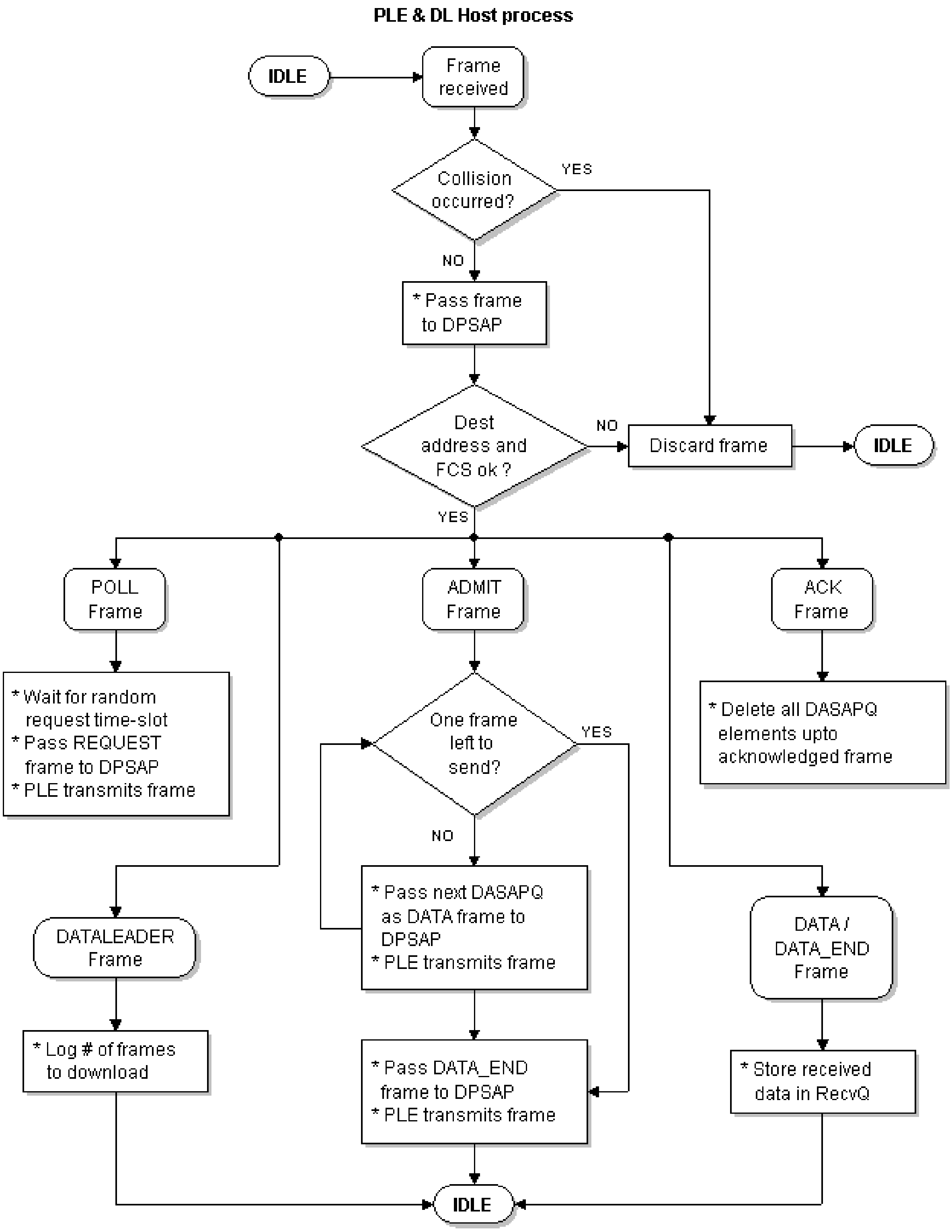


Figure 4.9: S-APRS PLE & DL Host process

Chapter 5

Protocol Performance Analysis

The previous chapters have defined and developed all the concepts and protocol characteristics necessary to do a performance evaluation of the S-APRS protocol. In the chapter, the network parameters are quantified and used to evaluate the protocol throughput performance.

The main objective for the thesis is to improve the maximum channel throughput of the APRS network. The evaluation is approached systematically by evaluating the throughput by varying key protocol parameters. The resulting effects are used to suggest further optimization of the protocol throughput.

5.1 Network Parameter Specification

In order to draw a comparison between the S-APRS and APRS network performance, a common base of network specifications need to be established. In Table 5.1, the common base parameters and their assumed values for the S-APRS network is specified.

The first of the characteristics, the bandwidth, is taken as 1200 bps for both the cell network (CN) and base network (BN). Increasing the baud rate for the BN and its effect on the throughput will be investigated later in the chapter. The second parameter, the packet length, was considered in §3.4.2. The medium propagation speed for the wireless S-APRS network is approximated by the speed of light for a vacuum (c), which is 3×10^8 m/s. The number of requests slots is assumed equal to the number of hosts contending for the channel. Setting the two numbers equal is known to produce the maximum throughput for S-ALOHA [13]. R_S(K,M), the number of successful requests received in the single protocol cycle, is a function of the number of request slots K and the number of hosts M contending for the slots. Its maximum value is equal to the maximum S-ALOHA

Table 5.1
S-APRS NETWORK PARAMETERS

Parameter	Symbol	S-APRS value	Unit
Bandwidth	W_{CN}	1200	bps
Frame Data Length	B_{FDL}	256	Bytes
Frame Overhead Length	B_{FOL}	48	Bytes
Maximum Total Frame Length	B_{TFL}	304	Bytes
Medium Propagation Speed (vacuum)	c	3×10^8	m/s
Number of Hosts	M	50	N/A
Request Slots	K	K=M	N/A
Number of successful Requests	R_S(K,M)	$DT_{SA}\times M$	requests/cycle
Number of data frames per cycle	D	1	N/A
Frame input rate per host	λ_{Host}	4.93×10^{-3}	frames/second
Total frame input rate	λ_{Tot}	$M\times\lambda_{Host}$	frames/second
Max throughput of S-ALOHA	DT_{SA}	36%	N/A

throughput times the number of hosts M. D is the number of data frames that a host is allowed to transmit per cycle. Increasing D has the effect of increasing the throughput at the cost of increased delay, as shown in [3]. The value of D is chosen for evaluation purposes at its minimum value of 1 frame.

The final parameter, that has not been discussed, is the number of hosts that is assumed to participate in the multiple access of the channel in both CN and BN. The concepts governing the number of hosts are introduced in the next section.

5.1.1 Traffic Distribution Model for APRS

In order to model the APRS network and determine its throughput, is it necessary to assume a traffic distribution model that best resemble the actual traffic. It requires a consideration of the traffic characteristics of APRS, which in turn is determined by its application.

In the thesis, public service applications are under consideration where the traffic consist of periodical updates on unit positions and infrequent textual messages exchange. The traffic therefore has a low duty cycle of packet transmission and exhibits a random nature. The model that is most commonly used [3; 4] to model traffic that exhibit a random nature and low duty cycle, is the Poisson distribution function. The Poisson model is assumed for this thesis and is used to describe the inter-arrival times of frames on the network with a mean of λ frames/second.

In [3] it is shown that random access techniques, in general, have the advantage over fixed assignment ones in that with decreasing duty cycle, it can support many more hosts for the same packet delay. For a number of hosts varying between 10 and a 100, the maximum packet rate for which random access performs better than fixed assignment is approximately $\lambda = 1$ [3].

5.1.2 Maximum Channel Throughput of ALOHA

As in [4; 8], the low duty cycle and random nature traffic is modeled by assuming the starting times of the frames to form a Poisson point process with mean λ_{Host} frames/second per host. With M hosts transmitting frames that each last for τ seconds, the normalized channel traffic G is defined as

$$G = \lambda_{Host}.M.\tau \tag{5.1.1}$$

Next, $\lambda' < \lambda$ is defined to be the rate at which frames that do not overlap are correctly received. The normalized channel throughput of ALOHA, S_A, is then defined as

$$S_A = \lambda'_{Host}.M.\tau \tag{5.1.2}$$

Now assuming that the channel traffic is Poisson, the probability that two frames will not overlap is determined [4] as $e^{-2\lambda.M.\tau}$, which gives

$$S_A = G.e^{-2G} \tag{5.1.3}$$

The equation is plotted in Figure 5.1. From the figure is can be seen that the maximum channel throughput of 18% is reached at a channel traffic G = 0.5. Any value higher or lower results in a decrease in throughput.

5.1.3 Maximum Number of Hosts in APRS

The maximum number of local hosts that can reliably be supported by the current APRS network is governed by the ALOHA limit. The limit is determined by the normalized channel traffic G of 0.5 that produces the maximum channel throughput as mentioned above. The value of G, the baud rate of 1200 bps and a frame length of 304 bytes are substituted into equation (5.1.1) and gives an aggregate frame rate of

$$\lambda_{Tot} = \frac{G}{\tau} = \frac{0.5}{\frac{8\times304}{1200}} = 0.247 \quad [frames/second] \tag{5.1.4}$$

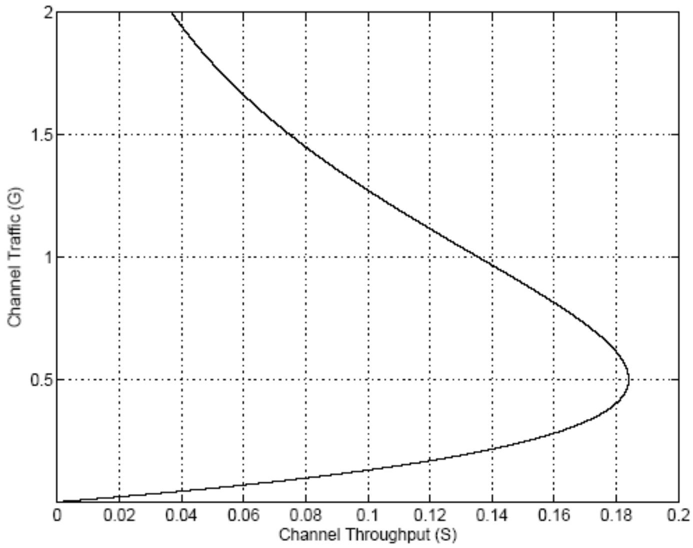


Figure 5.1: ALOHA channel traffic versus channel throughput

The value of λ_{Tot}, produces 444 frames on an APRS network in a net cycle time of 30 min. The author in [14] rounded the answer to $\lambda = 0.2$, which produces a maximum of 360 frames. The author then divided the amount of frames among a typical set of hosts in a local APRS network according to their individual frame input rates. The local APRS network host limit is the sum of the hosts, which add up to approximately M = 50. The value of M is assumed for both CN and BN and is added to the common-value base for comparison.

For the thesis, the hosts are assumed equal sources of frame input rates in order to simplify the analysis. Each host therefore produces frames at the same frame input rate of

$$\lambda_{Host} = \lambda_{Tot} \; / \; M \; = \; 0.247 \; / \; 50 \; = \; 4.93 \times 10^{-3} \;\; [frames/second] \qquad (5.1.5)$$

The value of the frame input rate corresponds to the classification of a bursty source according to [3], which is anything from a rate of 10^{-1} and lower and satisfies our requirement for the Poisson model. It is also assumed that each of the hosts are in the transmission range of the access point. The assumption is necessary to provide fixed traffic estimation in order to produce an accurate throughput analysis.

5.2 Protocol Cycle and Parameter Description

The section briefly introduces the protocol description as in [13] and the modifications made to it. Only the cycle that undergoes multiple-access is considered, which is the

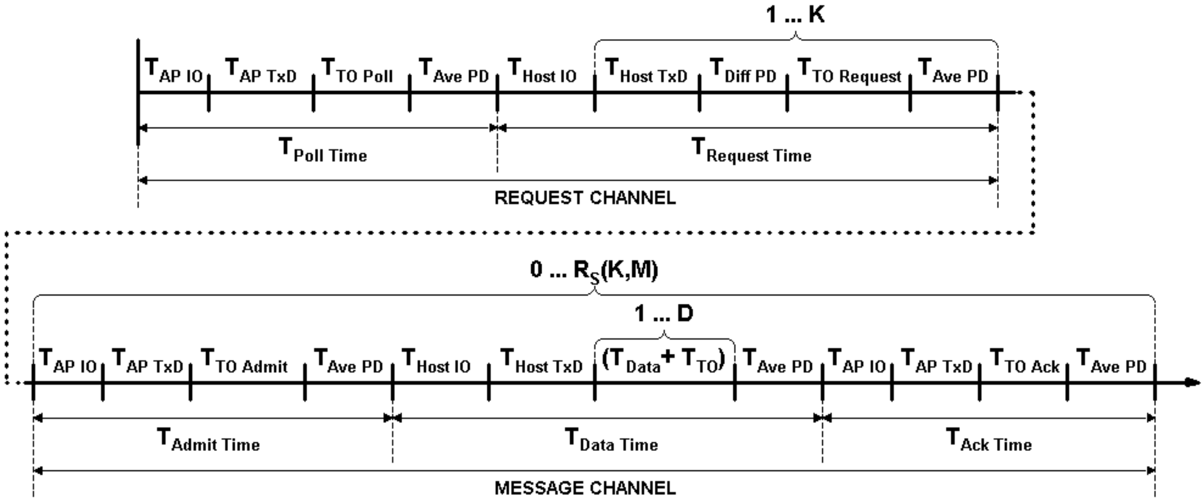


Figure 5.2: Protocol Cycle Diagram

upload cycle. Some protocol specific parameters are also defined that are necessary for the remainder of the chapter.

5.2.1 Protocol Cycle Sequence

As explained earlier, the upload cycle uses time division of the channel to create the consecutive request and message channels. The duration of the protocol cycle (T_{Cycle}) is the sum of the consecutive request channel and message channel time delays. The cycle consist of a number of sub-delays and is illustrated as a sequence of the delays in Figure 5.2. From the figure the total cycle delay, T_{Cycle} ,can be defined as

$$T_{Cycle} = T_{Request\ Cycle} + T_{Message\ Cycle} \tag{5.2.1}$$

where

$$\begin{aligned} T_{Request\ Cycle} &= T_{Poll\ Time} + T_{Request\ Time} \\ &= T_{AP\ IO} + T_{AP\ TxD} + T_{TO\ Poll} + T_{Ave\ PD} + T_{Host\ IO} \\ &\quad + K.(\ T_{Host\ TxD} + T_{Diff\ PD} + T_{TO\ Request} + T_{Ave\ PD}\ ) \end{aligned} \tag{5.2.2}$$

Table 5.2
PROTOCOL CYCLE DURATIONS

Parameter	Description
$T_{Host\ IO}$	Host Implementation Overhead Delay
$T_{AP\ IO}$	AP Implementation Overhead Delay
$T_{Host\ TxD}$	Host Transmitter Delay
$T_{AP\ TxD}$	AP Transmitter Delay
$T_{TO\ Poll}$	Poll Frame Transmission Delay
$T_{TO\ Request}$	Request Frame Transmission Delay
$T_{TO\ Admit}$	Admit Frame Transmission Delay
$T_{TO\ Ack}$	Acknowledge Frame Transmission Delay
T_{Data}	Data Frame Transmission Delay
T_{TO}	Transmission Overhead Delay
$T_{Ave\ PD}$	Average Propagation Delay
$T_{Diff\ PD}$	Difference between min and max Propagation Delay

and

$$\begin{aligned} T_{Message\ Cycle} &= R_S.(\ T_{Admit\ Time} + T_{Data\ Time} + T_{Ack\ Time}\ ) \\ &= R_S.[\ T_{AP\ IO} + T_{AP\ TxD} + T_{TO\ Admit} + T_{Ave\ PD} + T_{Host\ IO} \\ &\quad +\ T_{Host\ TxD} + D.(\ T_{Data} + T_{TO}\ ) + T_{Ave\ PD} + T_{AP\ IO} \\ &\quad + T_{AP\ TxD} + T_{TO\ Ack} + T_{Ave\ PD}\ ] \end{aligned} \tag{5.2.3}$$

The description of each delay is contained in Table 5.2.

5.2.2 Delay Definitions and Value Aspects

The definition and calculation of the protocol delay values are considered in the section. Further assumptions are made regarding the protocol parameters with the intent of developing an accurate protocol performance analysis.

Data Frame and Overhead Transmission Delay

For the S-APRS overhead frames, the individual frame control information is completely contained in the transmission overhead of a frame. It does not contain any of the possible 256 data bytes in the information field. It means that the transmission delay for each of the S-ARPS overhead frames is equal to general transmission overhead delay T_{TO}. Only the data-frame transmission delay T_{Data} is distinguished from the T_{TO}, which is separate from the data being transmitted. The delays are therefore defined as

$$T_{TO} = T_{TO\ Poll} = T_{TO\ Request} = T_{TO\ Admit} = T_{TO\ Ack} = B_{FOL}/W \quad (5.2.4)$$

and

$$T_{Data} = B_{FDL}/W \quad (5.2.5)$$

where W represents the channel specific bandwidth.

Propagation Delay

The value of the average propagation delay $T_{Ave\ PD}$ is determined by the minimum and maximum transmission distance from the host to the access point. It is defined as

$$T_{Ave\ PD} = \frac{T_{PDmin} + T_{PDmax}}{2} \quad (5.2.6)$$

The difference between the maximum and minimum propagation delays, $T_{Diff\ PD}$, is necessary to account for the loss in synchronization in the slotted-ALOHA request channel. The difference results in an additional time buffer that prohibits the overlapping of host requests. Therefore it is not included in the message channel, which does not undergo multiple-access. The difference is determined as

$$T_{Diff\ PD} = T_{PDmax} - T_{PDmin} \quad (5.2.7)$$

The values are now determined for both the CN and BN, which are defined as $T_{CNAvePD}$, $T_{BNAvePD}$, $T_{CNDiffPD}$ and $T_{BNDiffPD}$. For the CN, as discussed earlier, the values of 1200 bps baud rate, minimum and maximum propagation distance of 0-10 km and propagation speed of c $= 3\times10^8\ m/s$ are assumed. The values are substituted into equation (2.1.4) and equation (5.2.6) and gives

$$\begin{aligned} T_{CN\ AvePD} &= (T_{PDmin} + T_{PDmax})\ /\ 2 \\ &= \frac{0 + \frac{1\times10^3}{3\times10^8}}{2} \\ &= 1.666 \times 10^{-6}\ s. \end{aligned} \quad (5.2.8)$$

and

$$T_{CN\ DiffPD} = 3.333 \times 10^{-6}\ s. \quad (5.2.9)$$

Now for the BN, the same baud rate and propagation speed is assumed as with the CN. A minimum distance to the satellite access point of 800 km is assumed with the host directly under the satellite and a maximum of 3293 km [22] when on the horizon. Substituting the same equations, it gives

$$\begin{aligned} T_{BN\ AvePD} &= (T_{PDmin} + T_{PDmax})\ /\ 2 \\ &= \frac{\frac{800\times10^3}{3\times10^8} + \frac{3293\times10^3}{3\times10^8}}{2} \\ &= 6.822 \times 10^{-3}\ s. \end{aligned} \tag{5.2.10}$$

and

$$T_{BN\ DiffPD} = 8.31 \times 10^{-3}\ s. \tag{5.2.11}$$

Implementation Overhead Delay

The implementation overhead delay for both the host and access point implementations, $\mathrm{T}_{Host\ IO}$ and $\mathrm{T}_{AP\ IO}$, are measured on the coded protocol implementation done in chapter 4. The delays represent the sum of the delays in between the reception of a frame and the transmission of the response frame introduced by the implementation state machine.

The implementation overhead was measured and the average values for a single protocol cycle were determined. The value for the host, $T_{Host\ IO}$, and for the access point, $T_{AP\ IO}$, were determined as

$$T_{Host\ IO} = 4.884 \times 10^{-3}\ s \tag{5.2.12}$$

and

$$T_{AP\ IO} = 7.303 \times 10^{-3}\ s. \tag{5.2.13}$$

Transmitter Delay

The transmitter delays of both the host and access point, $\mathrm{T}_{HostTxD}$ and T_{APTxD}, are assumed equal and are referred to collectively as T_{TxD}. An average value is assumed of

$$T_{Host\ TxD} = T_{AP\ TxD} = T_{TxD} = 100 \times 10^{-3}\ s. \tag{5.2.14}$$

5.2.3 Summary

This concludes the protocol cycle delay description and the determination of the relevant analysis parameter values. The values are used in the following sections to do a systematic performance evaluation of the S-APRS protocol throughput.

5.3 Frame Specification Evaluation

The combination of the encapsulation of data and control information specifies a minimum amount of bytes needed for a valid frame. The amount of bytes that is transmitted per frame, which does not include actual data, is known as the *transmission overhead*. In the section the effect of the transmission overhead and frame length on the S-APRS throughput is evaluated.

5.3.1 Transmission Overhead and Frame Data Length

The transmission overhead of S-APRS frames was determined in §3.4.2 as 48 bytes. The length of the maximum amount of data per frame is 256 bytes. The relation between the two terms needs to be considered to determine how optimal the choice of values is.

The *transmission overhead ratio (TOR)* is defined as the ratio of the total frame length to the frame data length, which is

$$TOR = \frac{B_{FOL} + B_{FDL}}{B_{FDL}} \tag{5.3.1}$$

where the terms originate from Table 5.1. The graph of the transmission overhead ratio against the frame data length is shown in Figure 5.3.

The figure indicates that from a frame data length of approximately 150 bytes and onwards, the effect of the overhead per frame more or less settles and approaches 1 as the frame length approaches infinity. From 150 bytes onwards, the throughput is maximised with regard to the amount of transmission overhead.

The findings create the illusion that there is no bound to the length chosen for the data and increasing the value only contributes to the protocol throughput. It is not true because a governing limit on the frame length is introduced by the per frame transmission delay. The delay increases linearly with the increase of frame length. The delay requires a trade-off between the required effective throughput and an acceptable delay for the protocol.

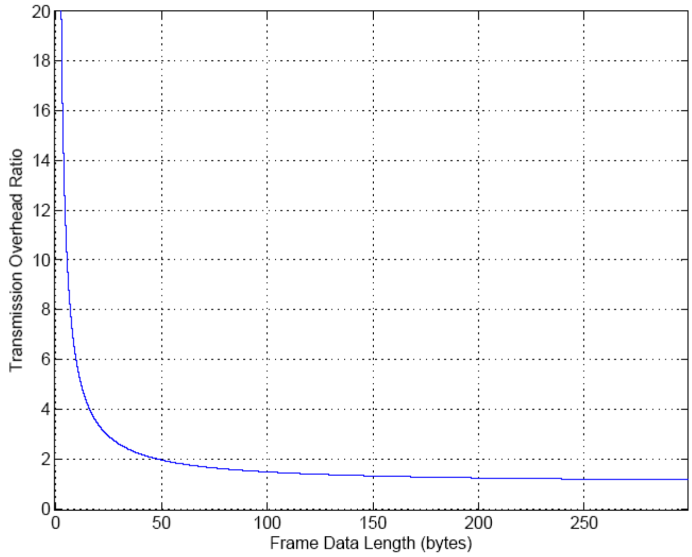


Figure 5.3: Transmission Overhead Ratio versus Frame Data Length

5.3.2 Conclusion for S-APRS frame

Considering the previous section findings, it is concluded that the maximum of 256 bytes frame data specified for the APRS UI frame is a satisfactory choice. The number of bytes is in the region of values that maximizes the throughput and small enough to produce an acceptable delay considering the low channel bandwidth of 1200 bps. In addition to the acceptable delay, it is also good practice to construct queues of data in multiples of byte sizes, which could result in more efficient processor computation.

5.4 S-APRS Throughput Definition

The section introduces the equation for the maximum effective channel throughput of the S-APRS protocol as defined by [13], which is from hereon referred to as the *data throughput*, denoted by DT_{S-APRS}. The maximum value of S-APRS throughput is considered because the maximum throughput of S-ALOHA, denoted by DT_{SA}, is assumed for the request channel. The derivation of the S-ALOHA throughput equation is also given to complete the definition of the S-APRS throughput equation.

5.4.1 S-ARPS Data Throughput

The data throughput of S-APRS [13] is defined as

$$DT_{S-APRS} = \frac{T_{Data} \times R_S(K, M) \times D}{T_{Cycle}} \tag{5.4.1}$$

The values of T_{Data}, D and T_{Cycle} have been discussed already in previous sections. The last term, R_S(K,M), the number of successful requests in a single protocol cycle, is defined as

$$R_S(K, M) = DT_{SA} \times M \tag{5.4.2}$$

The number of slots in the request channel, K, is assumed equal to the number of hosts M that contends for the channel. According to [13], the assumption results in the maximum effective channel throughput for S-ALOHA also referred to as its data throughput.

The data throughput equation is simplified by substituting the terms in equation (5.4.1) with equation (5.2.1),equation (5.2.2) and equation (5.2.3), which gives

$$DT_{S-APRS} = \frac{1}{\frac{T_{APIO}.(A) + T_{HostIO}.(B) + T_{TO}.(C) + (T_{TxD}+T_{AvePD}).(E) + T_{DiffPD}.K}{R_S(K,M) + T_{Data}.D} + 1} \tag{5.4.3}$$

where

$$A = 1 + 2.R_S(K, M) \tag{5.4.4}$$
$$B = 1 + R_S(K, M) \tag{5.4.5}$$
$$C = 1 + K + (D + 2).R_S(K, M) \tag{5.4.6}$$
$$E = 1 + K + 3.R_S(K, M) \tag{5.4.7}$$

The equation is used in the chapter to plot the throughput and the effect of varying different parameters has on its performance. The graphs assume the S-APRS values as contained in Table 5.1.

5.4.2 Definition of Slotted-ALOHA Throughput

As with ALOHA random access, it is needed to assume a suitable model to describe the channel traffic in the network to determine the S-ALOHA throughput. The Poisson model for a low duty cycle is again assumed and the mean traffic redefined to be G frames per *slot* [23]. G is defined in relation to λ_{Host}, the mean number of frames per second per host, as

$$G = \lambda_{Host}.T_{Rslot}.M \quad [frames/slot] \tag{5.4.8}$$

with

$$T_{Rslot} = B_{FOL} \;/\; W \quad [seconds] \tag{5.4.9}$$

where T_{Rslot} is the time of a single request slot and B_{FOL} the number of bits in a request frame. Assuming the traffic distribution to be Poisson, the maximum effective channel throughput, i.e. the data throughput, of S-ALOHA is then given by

$$DT_{SA} = G.e^{-G} \tag{5.4.10}$$

The equation shows that the data throughput is double that of ALOHA.

5.4.3 Summary

This concludes the definition of the throughput equation to be used for the data throughput performance analysis. The remainder of the chapter consist of the systematic throughput evaluation varying different protocol parameters and assuming the others fixed at the values defined by Table 5.1.

5.5 Frame Length Evaluation

An important measure to consider is the effect of the frame data length, with a fixed amount of overhead per frame, on the throughput. The effect of varying the frame data length is considered and the S-APRS frame performance determined.

5.5.1 Evaluation

In S-APRS, the overhead, B_{FOL}, is fixed and specified as 48 bytes. The effect of the frame data length is illustrated in Figure 5.4, which is a graph of DT_{S-APRS} versus the frame data length (B_{FDL}) for different numbers of hosts.

The graph shows that the rate at which the data throughput increases with increasing data length decreases monotonically. The throughput reaches an area of stabilization of between 90% and 95% from a frame data length of 10 kB onwards. It is also clear that the

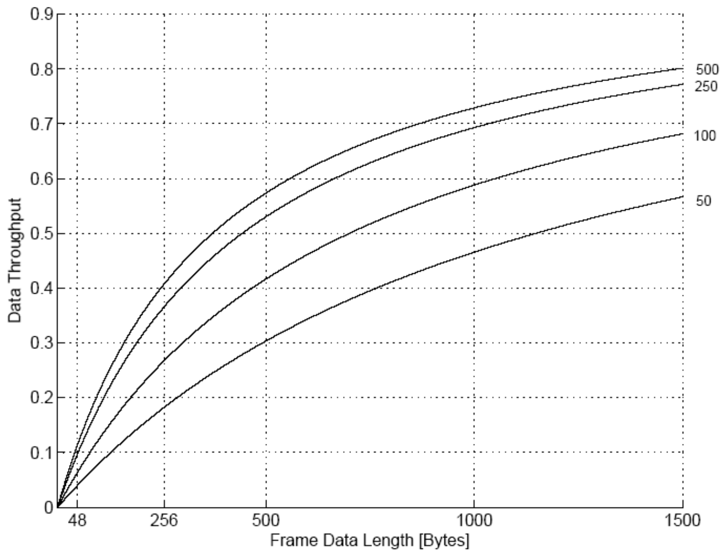


Figure 5.4: Data Throughput for S-APRS vs B_{FDL} for M = 50, 100, 250 and 500

effect of an increasing host population diminishes quickly. At approximately 500 hosts and onwards, the effect of the frame data length on the throughput becomes constant with the curves in the graph almost beginning to overlap.

With the S-APRS parameter values of 50 hosts and 256 data bytes, a data throughput of approximately 18% is achieved. Surprisingly, it equals the value of the ALOHA data throughput. The similarity and its governing parameters are considered in a later section.

5.5.2 Conclusion

From the evaluation done in the section, it is concluded that the frame data length has a noticeable effect on the S-APRS data throughput. The S-APRS specified maximum frame data length of 256 bytes results in no improvement on the present APRS data throughput. If the number of bytes cannot be increased, other protocol parameters and their effects would need to be considered in order to improve the APRS data throughput.

5.6 Traffic and Number of Hosts Evaluation

The previous section indicated the need for an extended study of the effects of other parameters on the S-APRS data throughput. It also assumed a fixed number of hosts M and a frame input rate per host of λ_{Host} with the values as indicated in Table 5.1. The values however are not fixed by the APRS specification, as is the case for frame overhead

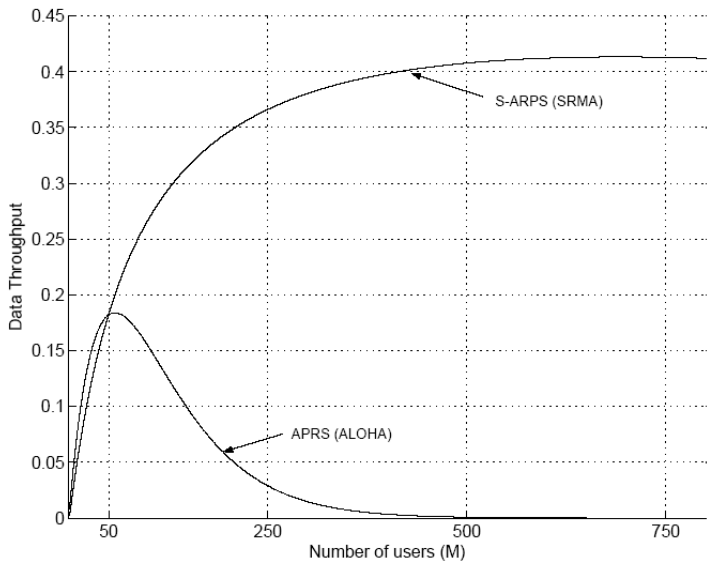


Figure 5.5: Data Throughput for S-APRS & APRS vs M

and data length. The effect of the number of hosts and the frame input rate per host are considered in the section.

5.6.1 Evaluation

Keeping the value of λ_{Host} fixed, the graph in Figure 5.5 is a comparison between the original APRS and the S-APRS protocol data-throughput for an increasing number of hosts. It shows that the throughput of APRS outperforms that of S-APRS up to a number of 50 hosts, at which stage APRS has its maximum data throughput. Higher than the number of 50 hosts, S-APRS dominates the throughput performance and reaches its maximum of 41% at approximately 700 hosts. After that the value then decreases until it again equals the APRS throughput of 18% at 2500 hosts.

It is necessary to understand the implications of increasing the number M hosts. The frame input rate per host λ_{Host} is fixed for each host and increasing M linearly increases the total input rate λ_{Tot}. It means that for the same 50 hosts in APRS, increasing the frame input rate per host by a factor of 14 would produce the same maximum throughput of 41%. It gives a per host rate for the 50 hosts of

$$\lambda_{Host} \;=\; 4.93 \times 10^{-3}.14 \;=\; 69.02 \times 10^{-3} \quad [frames/second] \tag{5.6.1}$$

The value is still below the boundary point of $\lambda = 1$, where above fixed assignment MAC techniques perform better than random access. Therefore, the use of S-ALOHA is still valid in the request channel.

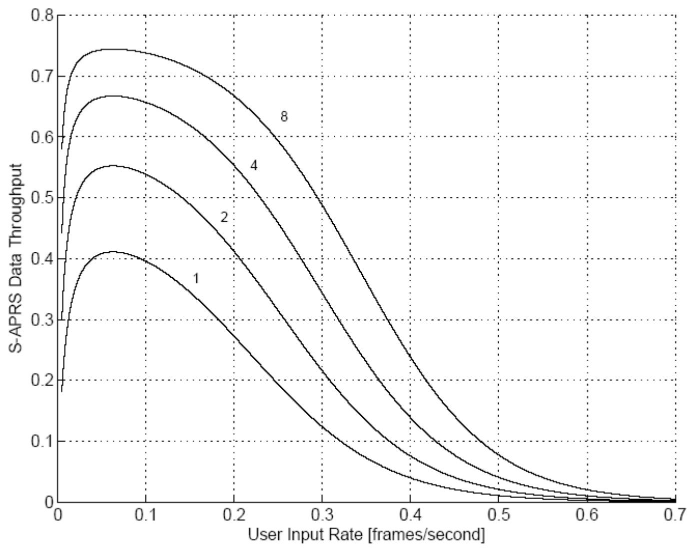


Figure 5.6: S-APRS Throughput vs User Input Rate for D = 1,2,4 and 8

5.6.2 Conclusion

The evaluation has shown that the S-APRS performs increasingly better than APRS when either the rate at which hosts generate frames or the number of hosts are increased above that of APRS. If it is required that the network population and traffic stay unchanged, there is still the need for a deciding parameter that would make S-APRS improve on APRS. The next section evaluates a SRMA specific parameter.

5.7 Message Slot Frames Evaluation

The section considers the effect on the data throughput of varying the parameter D, which is the amount of data frames sent per host per message slot. The parameter is a good candidate to consider varying because it has no relation to APRS and therefore its value is not specified.

5.7.1 Evaluation

Figure 5.6 shows the graph of the S-APRS data throughput against the frame input rate for values of D = 1,2,4 and 8. The graph indicates that with a small increase of D the data throughput is substantially increased.

Although it seems as if the parameter D could be increased unlimitedly in order to achieve the maximum possible data throughput, its value is governed by a trade-off. According to [13], increasing D also increases the average delay per user. Therefore, in order to obtain the optimal overall performance for the protocol, a trade-off is necessary between the throughput and delay performance.

For the sake of determining an estimated S-APRS throughput performance, an average trade-off value of D = 4 at a frame input rate as in APRS of $\lambda_{Host} = 4.93 \times 10^{-3}$ is assumed, which gives as S-APRS data throughput of 58%.

5.7.2 Conclusion

In the section it was shown that the parameter D has a substantial effect on the data throughput. The parameter's value is not specified by APRS and is therefore the most appropriate candidate parameter so far to consider varying in order to maximize the S-APRS throughput according to a set of user requirements. The delay requirement of the protocol limits the maximum value of D, which gives the user the freedom to make a compromise between the desired data throughput and delay. The next section considers the effect of bandwidth on the throughput.

5.8 Bandwidth Evaluation

The last parameter to consider is the effect of bandwidth on the throughput. It is done in order to determine the viability of increasing the bandwidth for the base network with the LEO satellite as access point.

5.8.1 Evaluation

The effect of the bandwidth is considered by plotting its effect on the data throughput for different values of D. The resulting graph is shown in Figure 5.7.

The graph indicates that there is an average decrease of 90% in throughput over a bandwidth range from 1000 to 10 000 bps. The significant decrease is mainly the result of the media access control in the request channel, which in SRMA is governed by S-ALOHA. The graph of the effect of an increase of bandwidth on S-ALOHA is shown in Figure 5.8.

The main reason for considering an increase in bandwidth would be to decrease the average frame delay per protocol cycle by the resulting increased transmission speed. As shown in Figure 5.7, there is the option of increasing D in order to counter the significant loss in throughput performance. The parameter D, on the other hand, again increases the delay

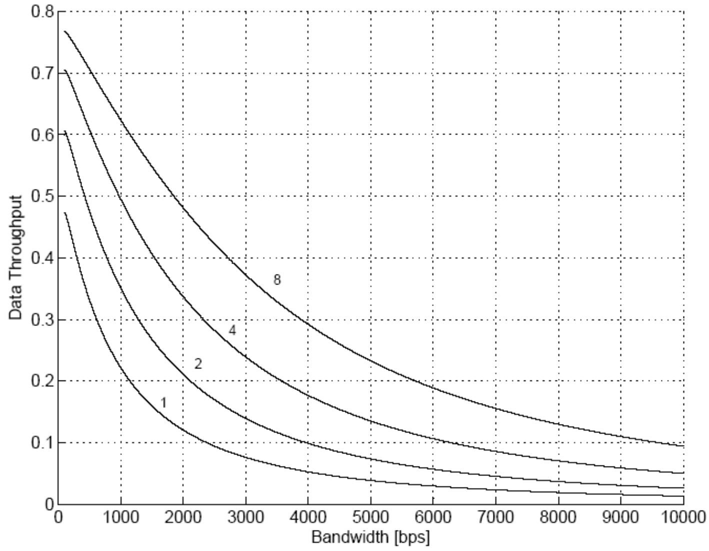


Figure 5.7: S-APRS Data Throughput vs Bandwidth for D = 1, 2, 4 and 8

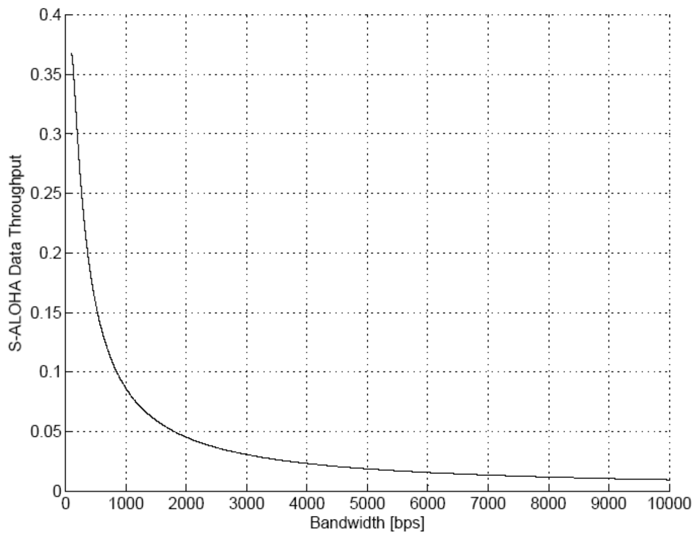


Figure 5.8: S-ALOHA Throughput vs Bandwidth

as described in the previous section. Therefore, the bandwidth could be used to assist in determining the trade-off between the throughput and delay requirements.

5.8.2 Conclusion

From the evaluation of the effect of increasing bandwidth on data throughput, it is concluded that the average available bandwidth of LEO satellites of 1 Mbps cannot be utilised efficiently by S-APRS. If the delay per user performance is the most important parameter in a network design, increasing the bandwidth will barely optimise the parameter.

5.9 Conclusion of Performance Evaluation

The main objective for the thesis is to improve the data throughput of the APRS network. In the chapter, a systematic evaluation was done of the effect of different protocol parameters on the data throughput.

The evaluation has shown that S-APRS can support either a greater host population or a substantial increase in traffic than APRS at a higher level of throughput. The number of data frames per message slot D was identified as the most appropriate parameter for optimising the data throughput performance. Using an average trade-off value of D=4, S-APRS produces a data throughput of 58%, which is a substantial increase to that of ALOHA's 18% in ARPS.

The evaluation of the effect of increasing bandwidth on throughput has indicated that it results in a substantial degradation of throughput. The bandwidth could be used together with the parameter D to establish a trade-off between throughput and delay.

Chapter 6

Conclusions

The chapter gives a summary of the work done in order to accomplish the research objectives. Conclusions are drawn with regard to the success achieved in meeting the objectives. Finally, recommendations are made regarding areas for future research.

The overall objective of the thesis is to improve the two performance indicators of APRS in order to enlarge its spectrum of applicability with regard to time-critical applications. The operation required the consideration of the media access control implemented by ARPS to determine the necessary modifications.

6.1 S-APRS Coherence with APRS Specification

The first objective was to substitute the MAC layer protocol with the minimal changes to the APRS specification and network architecture. The substitution with SRMA was done in chapter 3 and showed that the SRMA protocol supports the present network topology. It uses the APRS digipeaters to function as access points to the local host network. It also supports the flexibility of the APRS network, which includes the varying host population that accompanies mobility.

The second aspect of the integration was with minimizing the changes to the APRS network architecture, which includes the hardware and software of APRS hosts. SRMA has indicated that it utilizes the different APRS hosts that differ with regard to their transmission range. The digipeaters and wide-digipeaters that are equipped with different transmission ranges coincide with the host-to-access point topology. The similarity does not require any modification to the existing host hardware. The hardware generally consists of a radio transceiver that is connected to the terminal node controller.

The integrated protocol stack has indicated the need for minor modifications of the network architecture with regard to the software implementation. The "digipeater address" field of the AX.25 UI frame structure used in APRS was redefined. Recent APRS development by TAPR has indicated redundant byte-space in the field. The redundancy was utilized by subdividing the field for the use of SRMA MAC control information.

A further distinction is made between host and access point software. The difference is in the SRMA MAC-layer implementations and requires the user to configure the terminal for either of the two. Present APRS terminals require similar user intervention with configuring of its general host, digipeater of wide-digipeater status. Therefore, the SRMA substituted protocol resembles the APRS complexity with only the new implementation software modification required.

Considering the integration process and modifications described above, it is concluded that the proposed S-APRS protocol would be a viable operation. S-APRS also has good potential to be accepted by present APRS users because it does not require extra user expertise and retains the user interface.

6.2 Protocol Modelling and Implementation

The second objective was to develop a protocol model of the new protocol stack in order to create a largely platform-independent implementation from which the implementation overhead can be effectively measured and optimized. The objective was addressed in chapter 4 using UML and SDL standards, which provided the object-orientated conceptual design of the protocol.

The protocol modeling was done using traditional modeling as specified by the OSI. The specification was combined with some graphical model extensions based on object-orientated concepts. The combination was used to build the UML model which consisted mainly of class and sequence diagrams. The diagrams provided a good description of the protocol functionality.

Producing SDL diagrams, which use extended finite state machines, further extended the model. The diagrams provided a more detailed description of the protocol in order to further aid the code development process.

A complete protocol implementation was done in C on the Linux platform. The software was used for testing the protocol performance on an emulator of the wireless physical link of APRS. The emulator modeled the link characteristics and most importantly, the multiple-access frame collisions. The corresponding implementation specific measures was done and incorporated in the performance evaluation.

In conclusion, the protocol model developed created the basis for which to do rapid development of a platform-independent implementation. The measurements made on the protocol indicated that the software was efficiently implemented and that the protocol dynamics operated as estimated.

6.3 Protocol Performance Evaluation

The final objective was to evaluate the protocol performance and determine the improvement on the APRS data throughput. The existing APRS network parameters was used to create the basis for comparison between its own performance and that of the proposed S-APRS protocol. The traffic model was chosen according to the rate of traffic determined from the ALOHA limit imposed on a typical local APRS network.

The throughput evaluation was done by varying the different S-APRS protocol parameters. Varying the parameters resulted in the determination of the individual parameters' effects on the throughput. The common base comparison showed that S-APRS at worst would deliver the same throughput as APRS. The main reason was that the SRMA MAC layer performed worse than ALOHA at high traffic burstiness.

The number of hosts was varied, which in turn also varied the aggregate frame input rate. The evaluation showed that S-APRS, operating at its maximum channel throughput, could support approximately 14 times the amount of hosts in a local APRS network. The throughput was determined to be 41%, which is more than double that of APRS's 18%.

Then the effect of the only APRS-independant parameter, the data frames per message slot D, was considered in the S-ARPS data throughput. The evaluation showed that increasing D increased the throughput substantially. The effect of D increasing also the delay per user ratio, required a trade-off between throughput and delay. An average trade-off value of D=4 produced a S-APRS data throughput of 58%, an factor 3 increase of APRS throughput.

Finally, the effect of a variation in channel bandwidth was considered. Increasing bandwidth resulted in a significant decrease in S-APRS throughput. The decrease was mainly the result of the effect of slotted ALOHA in the request channel. It was concluded that the bandwidth could be used together with the parameter D in order to determine the optimal trade-off between throughput and delay.

The reliability was also increased with S-APRS making provision for positive acknowledgements, which is used to indicate successful transmission. The acknowledgements can be used to implement MAC layer retransmission. Together the acknowledgements and retransmissions provide a basis for increased network reliability.

6.4 Summary

In conclusion, the analysis has indicated that S-APRS's worst-case scenario throughput equals APRS's best throughput performance. With the choice of an average value for the parameter D, the S-APRS throughput can be increased by a factor of 3 to 58%. The basis for increased APRS reliability was created, which could be further developed to provide the required guarantees. It is also concluded that S-APRS would be well received by present APRS users by requiring no extra user expertise or modifications to the APRS user interface. Therefore, it is concluded that the thesis has provided sufficient proof to suggest that the S-APRS integration will be a viable operation.

6.5 Recommendations

The analysis of the effect of variable traffic on the S-APRS throughput suggests an area for future research. The analysis showed that S-APRS would perform well across a wide range of frame user input rates. It indicated that by increasing the number of data frames D in a single message slot, the throughput is also increased. In order to make the S-APRS protocol consistent for a range of applications, the protocol's response to varying traffic would need to be addressed.

It is recommended to study the possibility of an extended S-APRS protocol that stabilizes the throughput in response to traffic changes. The protocol would need to determine the traffic demand dynamically and vary the value D accordingly in order to keep the throughput constant. The result would be a more predictable system that optimizes its performance according to a set of user specifications.

Appendix A

UML Class and Sequence Diagrams

A.1 Class Diagrams

Chapter 4 introduced the main class diagram, which illustrated the packages of class diagrams that each represents a protocol layer. The class diagrams for the S-APRS data-link layer for both the host and access point implementation are shown in this appendix. The classes are the

- SRMA Access Point/Host Cycle Multiplexer (SAPCM/SHCM),
- SRMA Access Point Upload/Download-cycle (SAPU/SAPD),
- SRMA Host Upload/Download-cycle (SHU/SHD),
- Link Multiplexer (LM) and
- Physical Link Emulator (PLE)

Each of the classes is implemented as a state in the EFSM specified by the SDL diagrams. A brief description is now given of the function of each class.

SAPCM / SHCM

The first set of classes, the SAPCM and SHCM, are the cycle multiplexers that route data-link layer messages to either the upload or download classes depending upon the cycle in progress. It sends and receives messages with service primitives via the Data-Link/Application Service Access Point (DASAP) service interface to the application layer class.

SAPU / SHU

The second set, the SAPU and SHU, represent the upload cycle dynamics for the access point and the host implementation. It receives the message routed by the corresponding cycle multiplexer and determines the logical flow of the upload cycle. For the access point during the upload cycle, the received data are stored in the shared memory DASAP queue that enables the application layer to access the data.

SAPD / SHD

The third set, the SAPD and SHD, represent the download cycle dynamics for both implementations. Is also receives related messages from the corresponding cycle multiplexer and responds to it according to the logical flow of the cycle. For the host during the download cycle, the SHD receives the downloaded data and stores it in the shared memory DASAP queue for access by the S-APRS application layer.

LM

The Link Multiplexer has the same functionality for both the host and access point. It adds and checks the frame-check sequence (FCS), which is just a cyclic redundancy check (CRC), for incoming and outgoing frames. It also confirms the destination address of the frame corresponds to that of the host. The LM also uses the Data-Link/Physical-Link Service Access Point (DPSAP) interface to access the physical layer emulator (PLE).

PLE

The physical-link layer emulator sends and receives frames between the hosts. Received frames are tested for a collision and successful frames are passed to the LM through the DPSAP interface. The PLE also receives frames from the LM through the same interface and transmit these using the Ethernet physical link layer.

A.2 Sequence Diagrams

The appendix contains the UML sequence diagrams that describe the S-APRS upload and download cycles. These cycles are described for both the host and access point process implementation.

For the sequence diagrams, it is assumed that the DASAPQ already contains data frames added by the APRS UI application. The notation for sequence diagrams in UML Version 1.5 is not useful for denoting decision-making actions clearly. In order to avoid potential confusion or repetitive descriptions, the shortest sequence of events leading to a successful transmission is described.

The sequence diagrams notation consist of 3 objects:

1. The Class object,
2. the Process Bar and
3. the Message.

The class object is an instance of a class created in the main class diagram. The process bar indicates the progress of time as the cycle continues. The messages are the events initiated by objects that trigger other objects, which can include passing related information as arguments. An object can also send a message to itself, which is the initiation of an internal procedure.

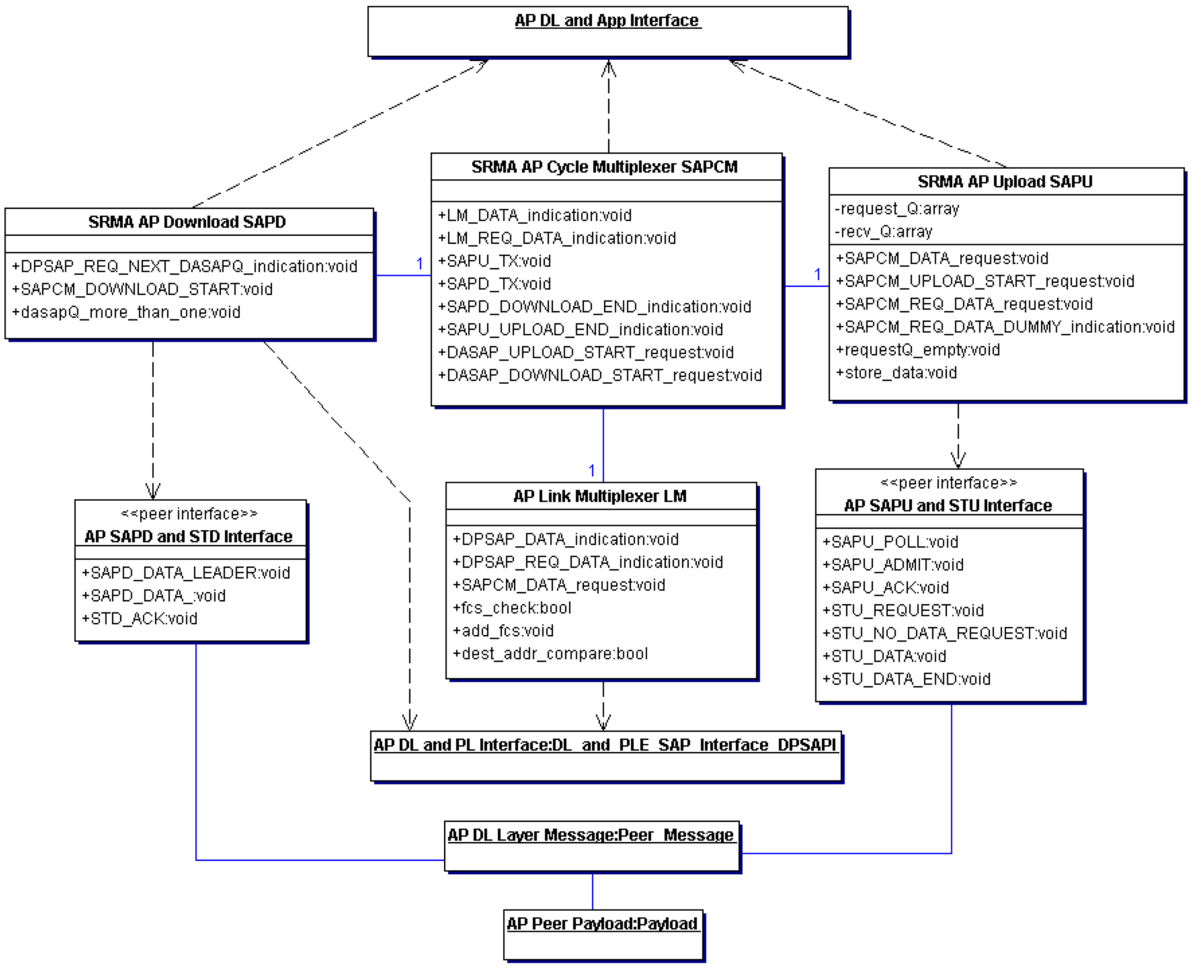


Figure A.1: Access Point Data Link Layer

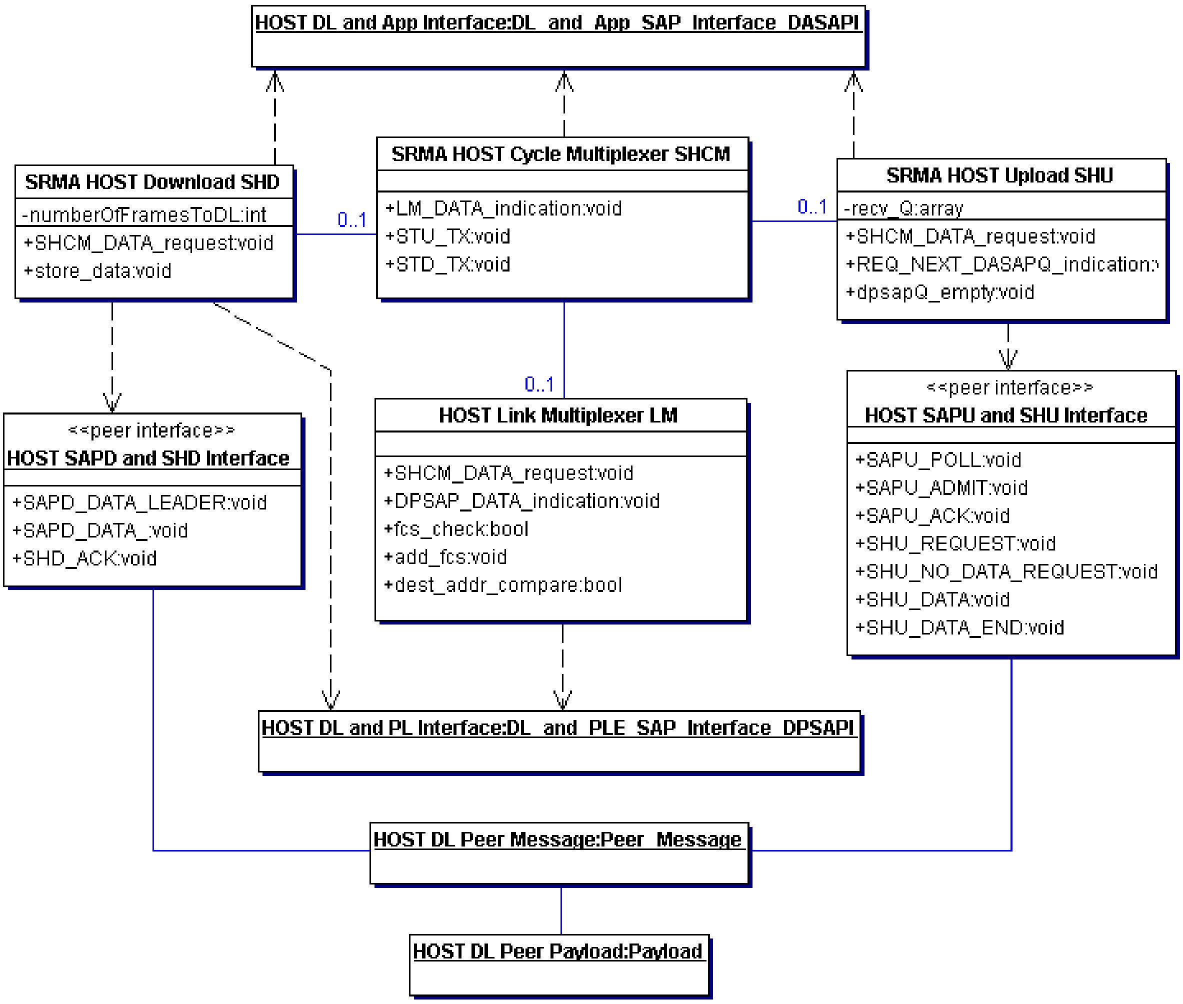


Figure A.2: Host Data Link Layer

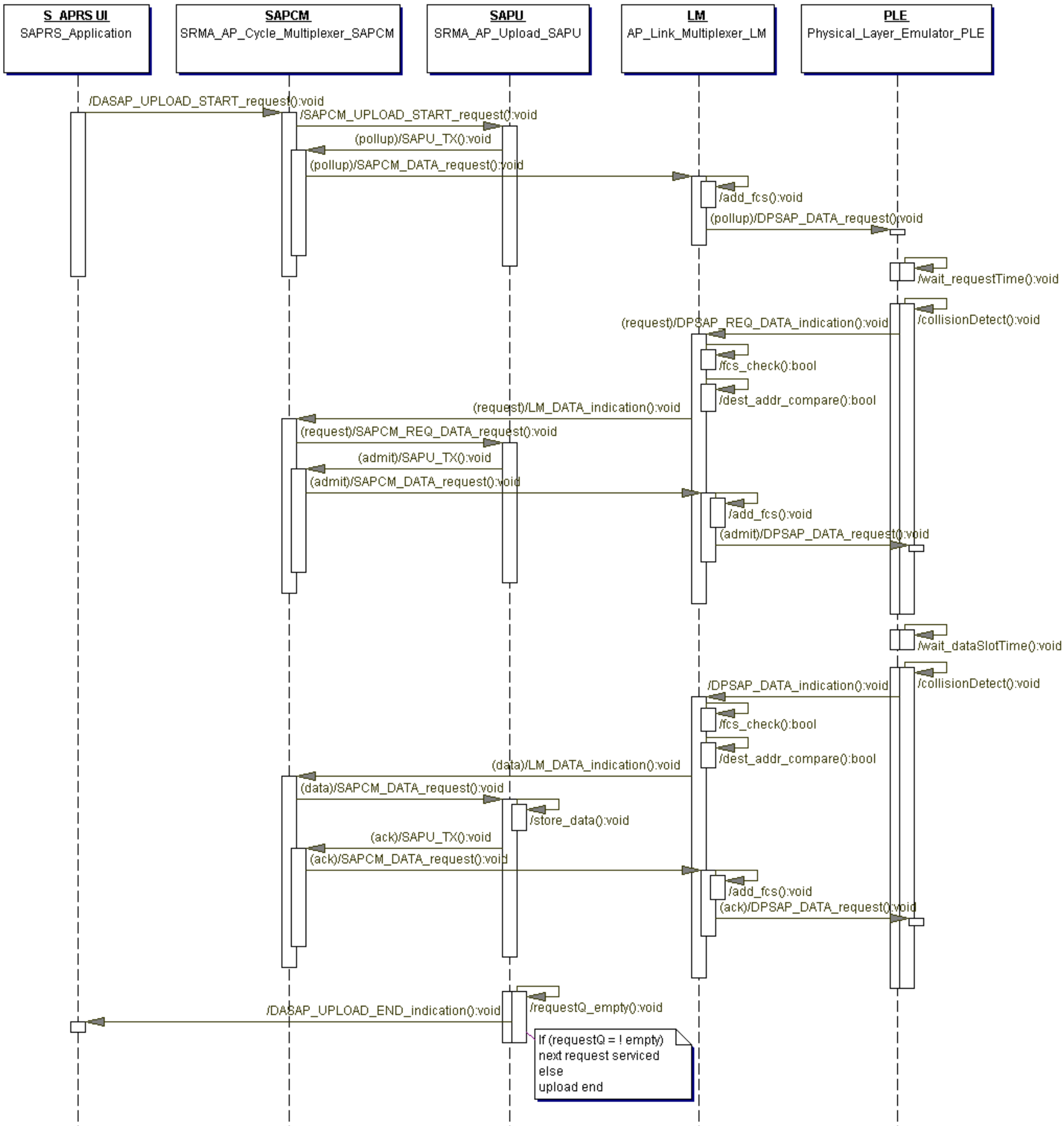


Figure A.3: AP Upload Cycle Sequence Diagram

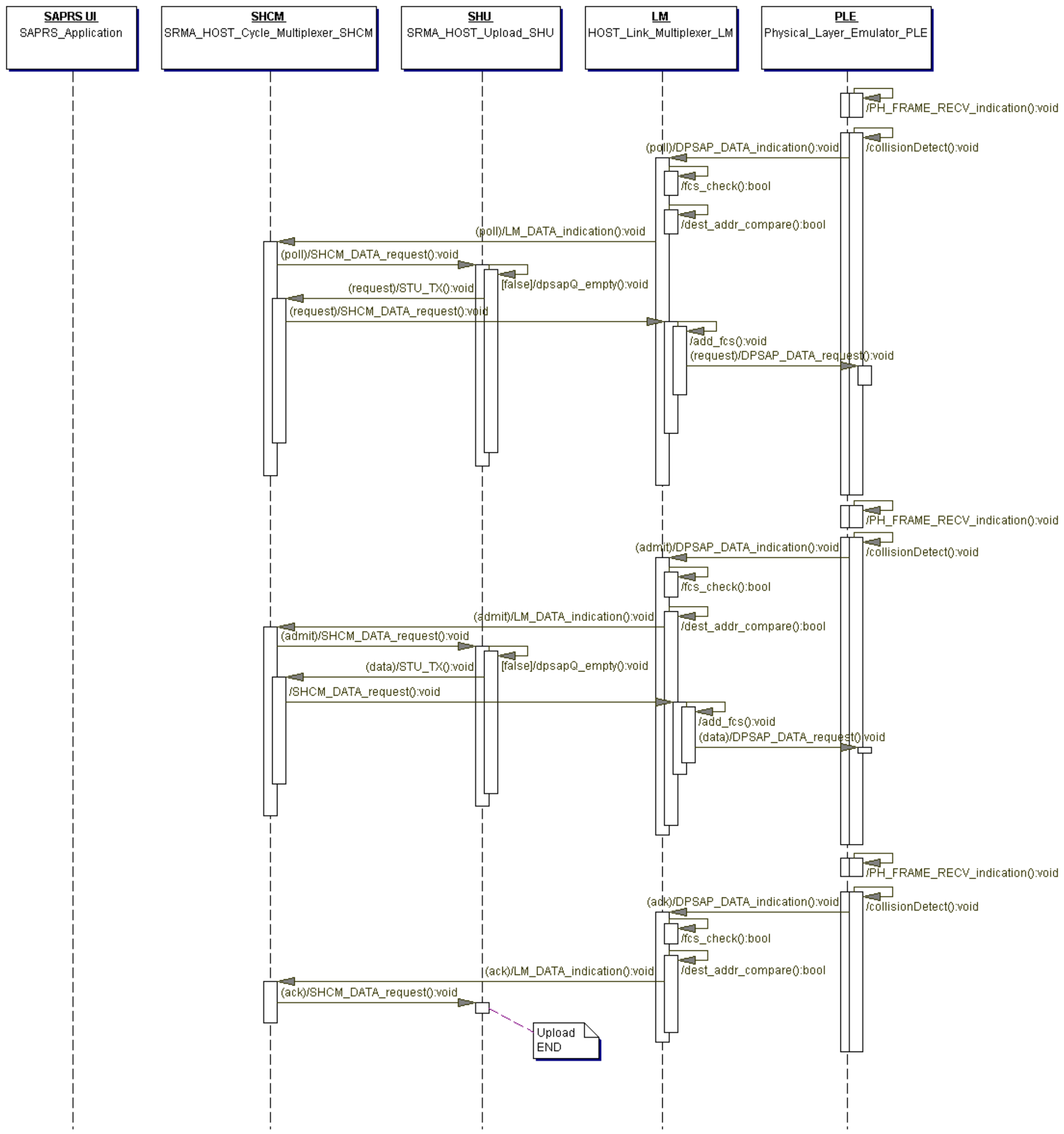


Figure A.4: Host Upload Cycle Sequence Diagram

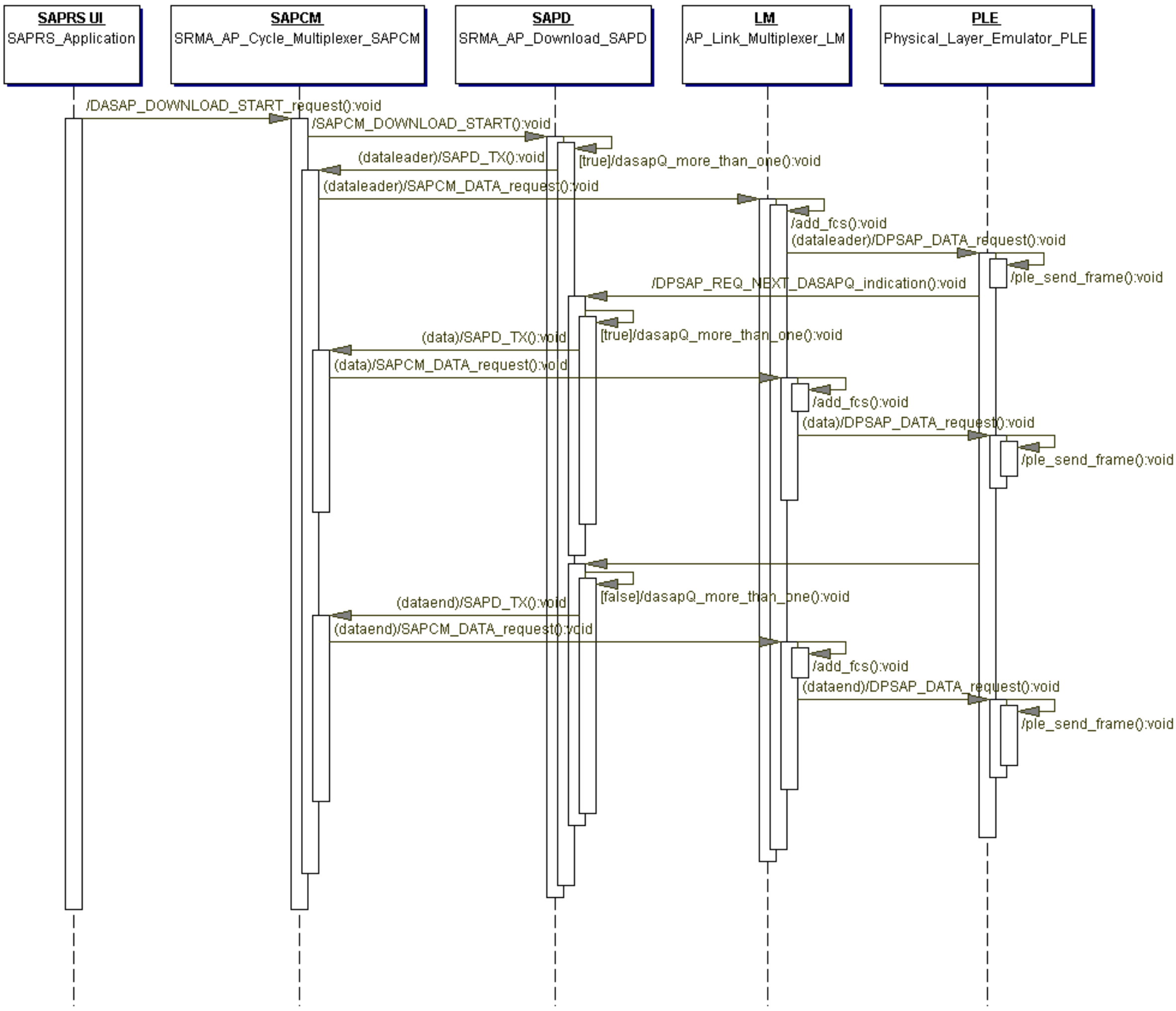


Figure A.5: AP Download Cycle Sequence Diagram

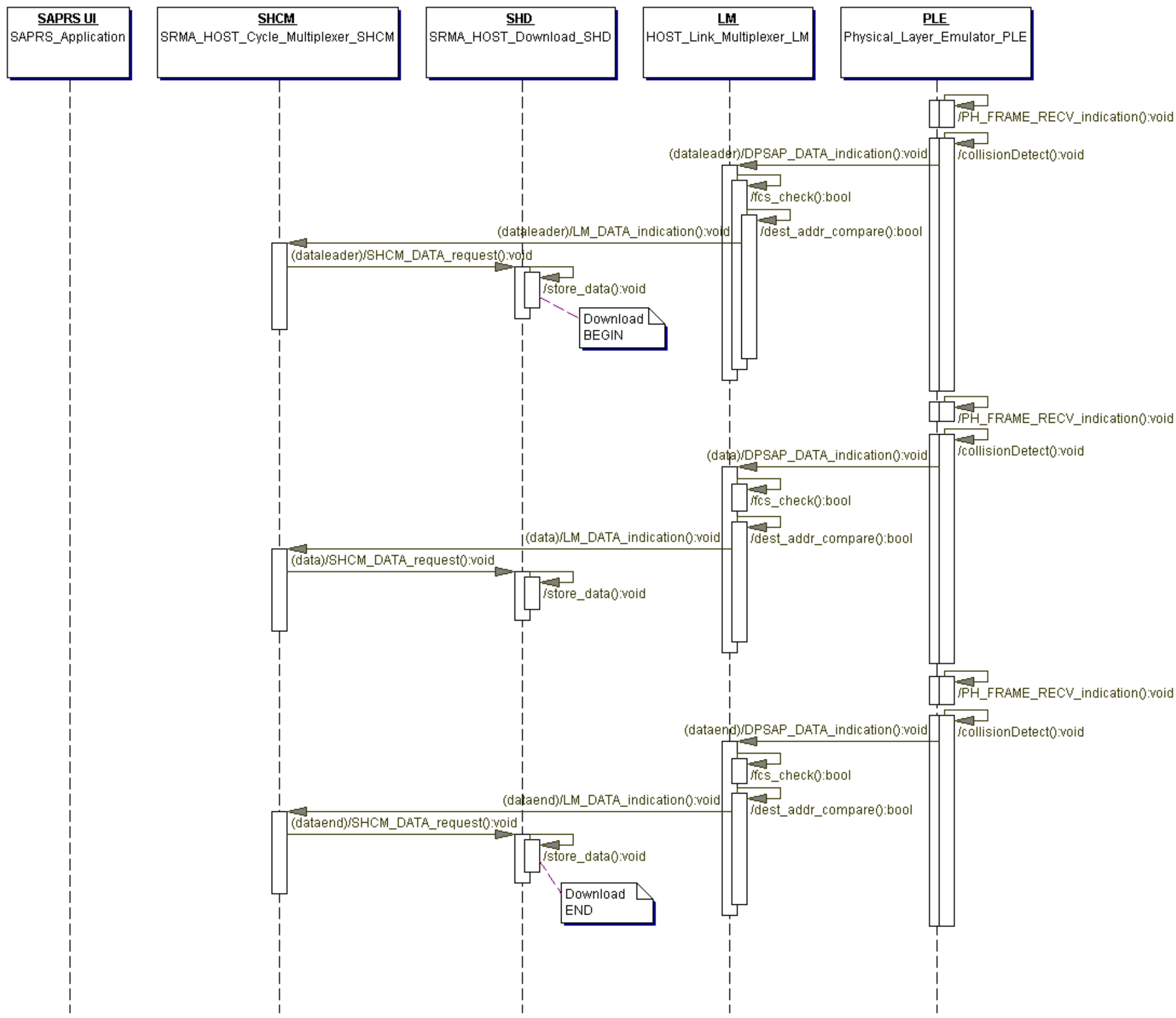


Figure A.6: Host Download Cycle Sequence Diagram

Appendix B

SDL Diagrams

B.1 Overview

The System Description Language (SDL) diagrams are used to provide a detailed description of the extended finite state machine (EFSM). A description of the SDL notation is given on the next page.

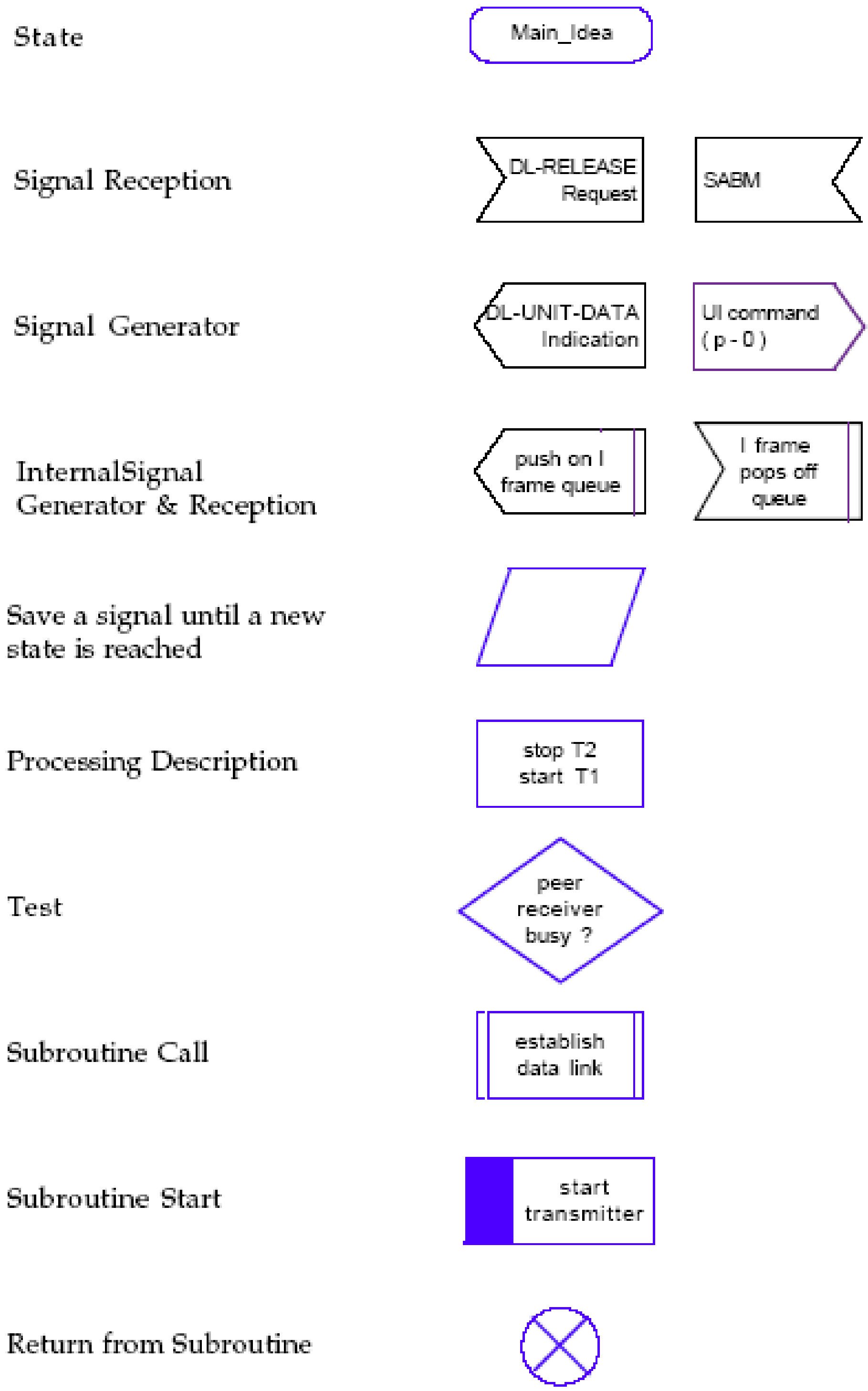


Figure B.1: SDL Notation Description

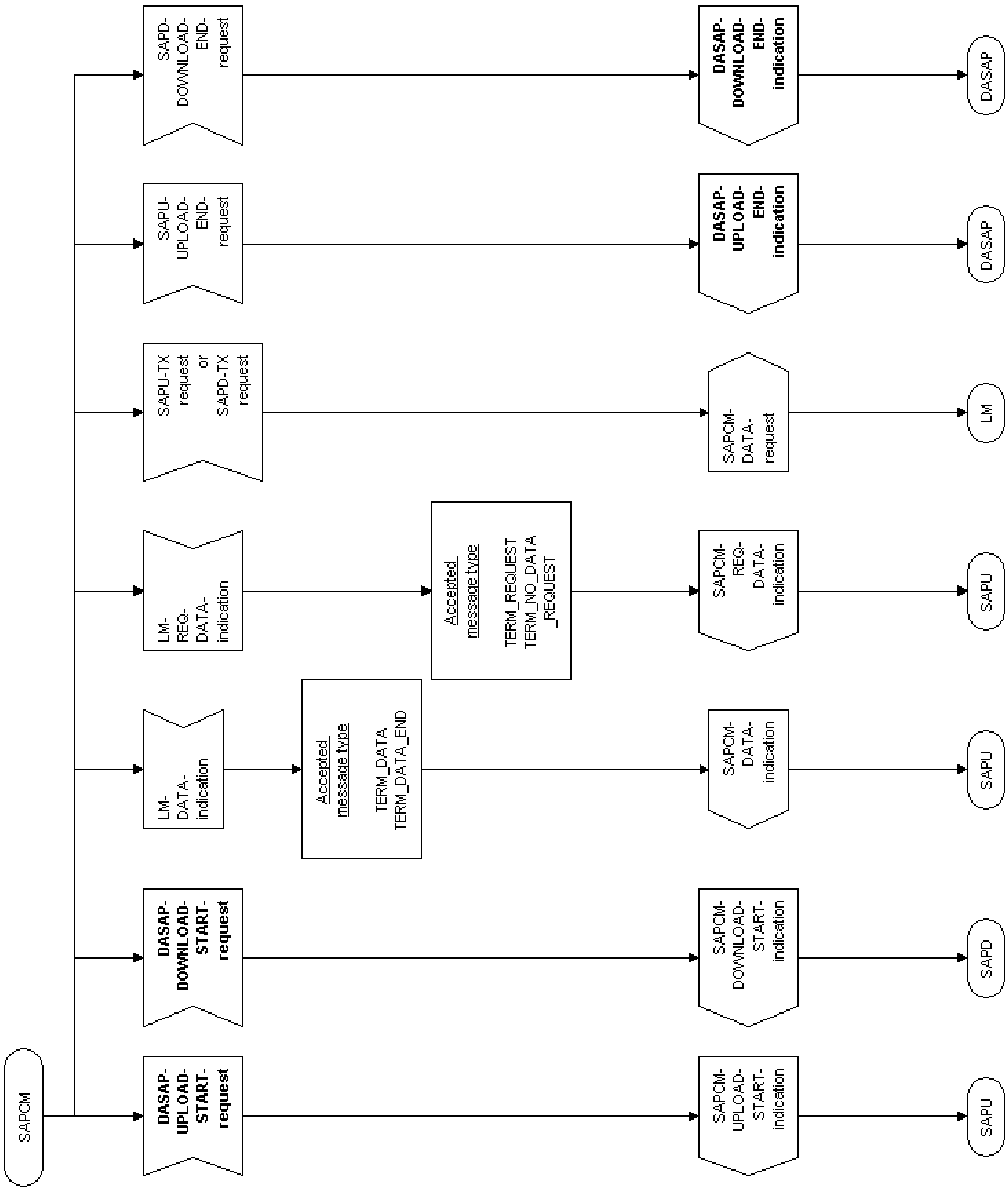


Figure B.2: SRMA Access Point Cycle Multiplexer

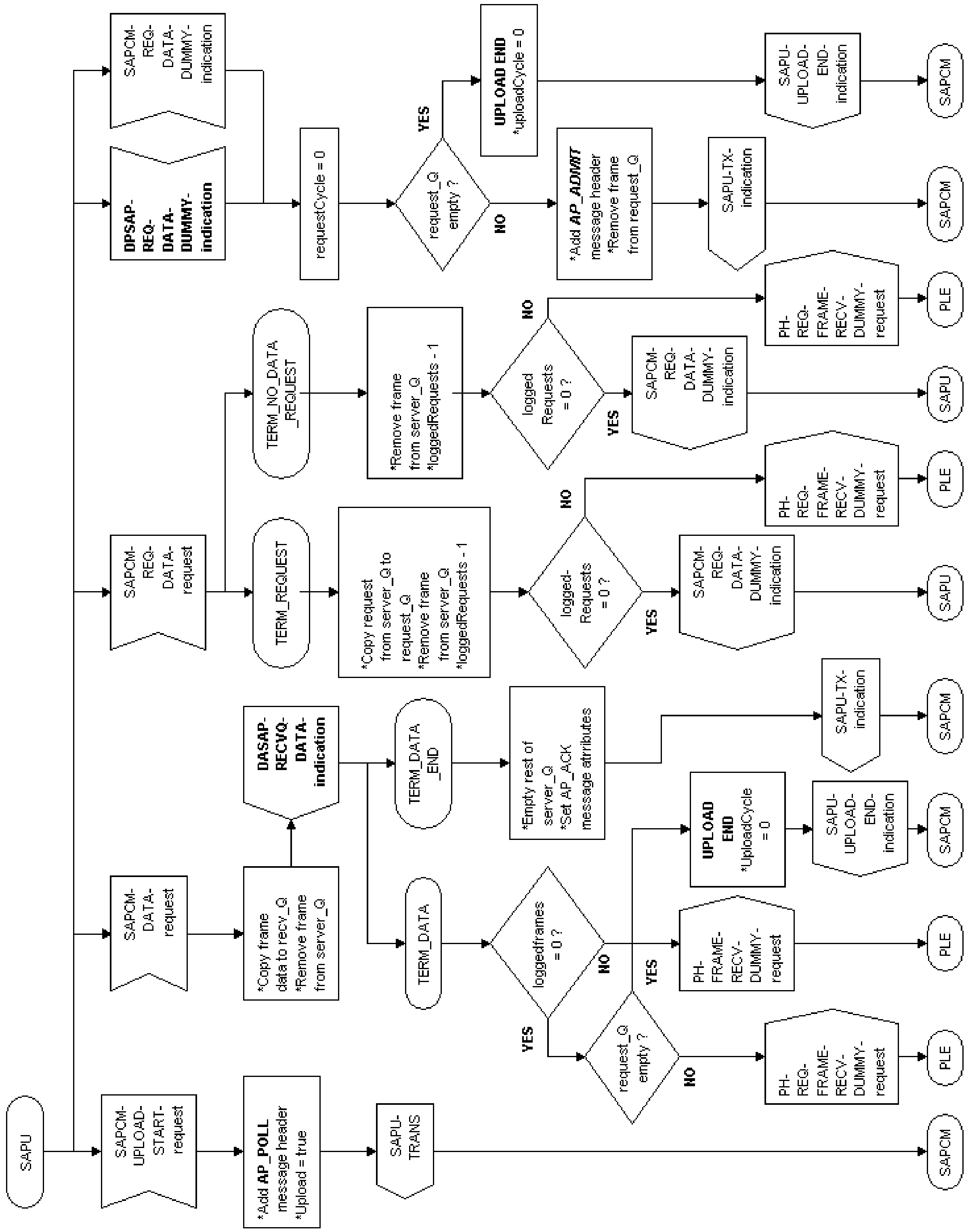


Figure B.3: SRMA Access Point Upload-cycle

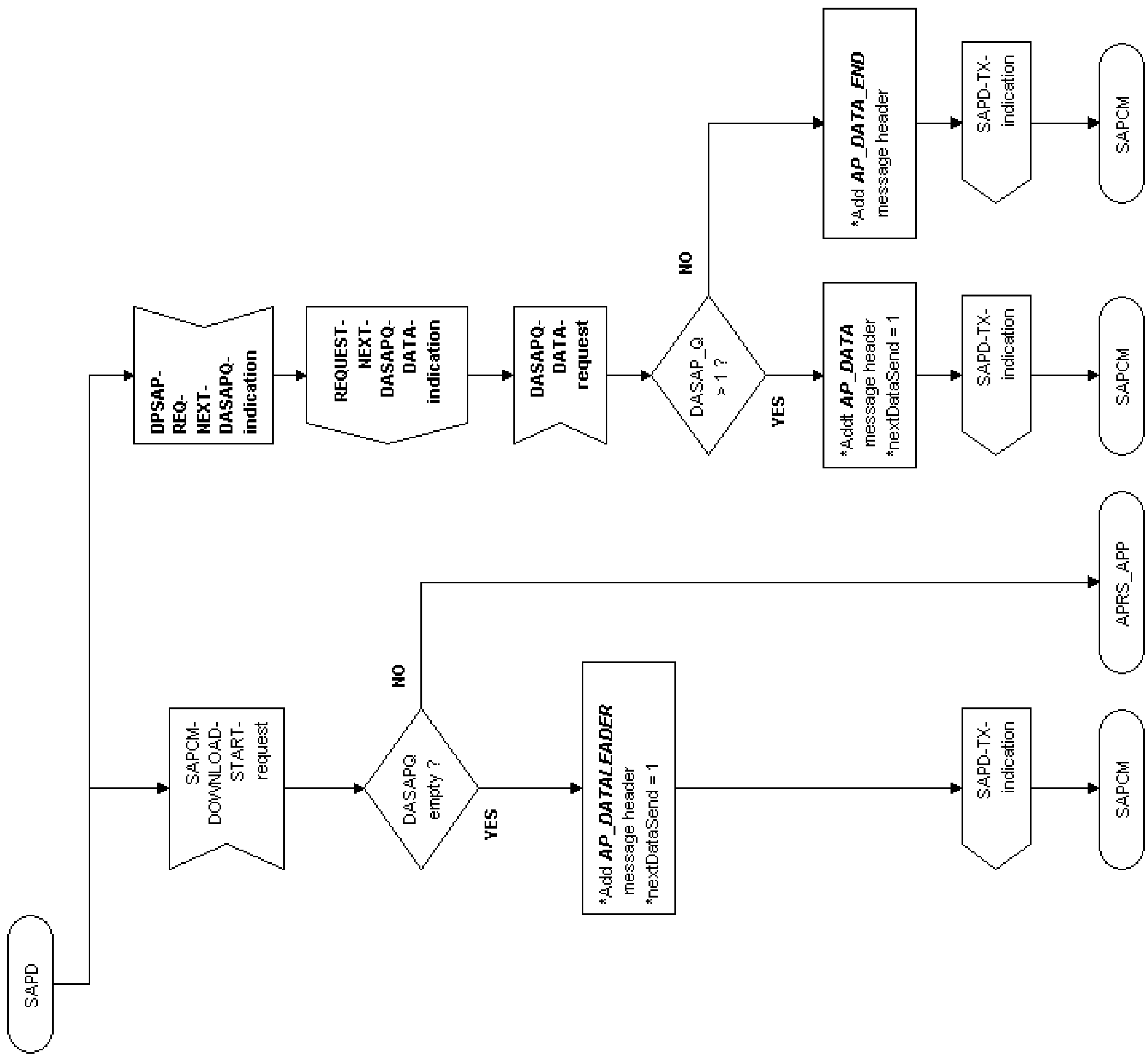


Figure B.4: SRMA Access Point Download-cycle

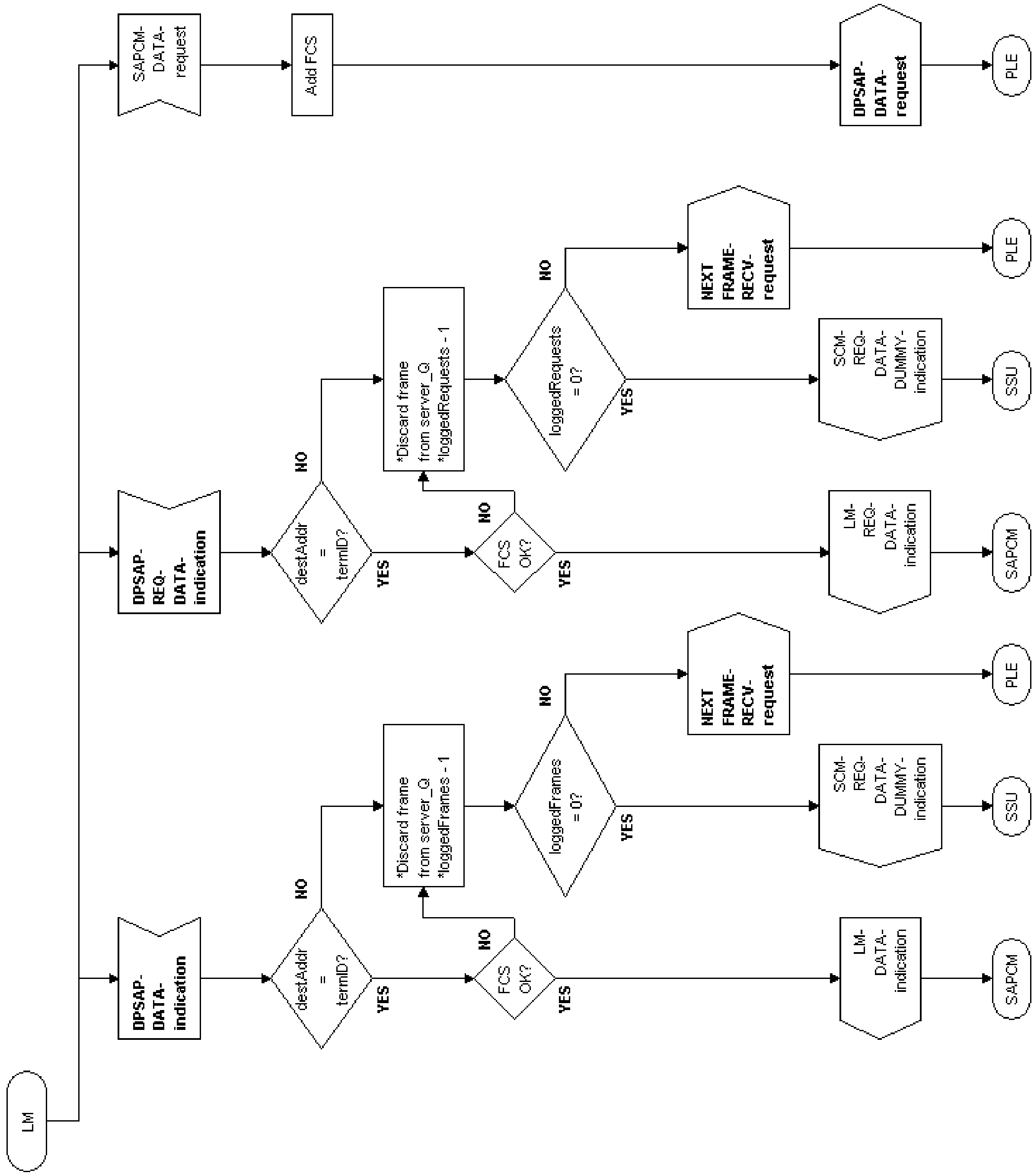


Figure B.5: Access Point Link Multiplexer

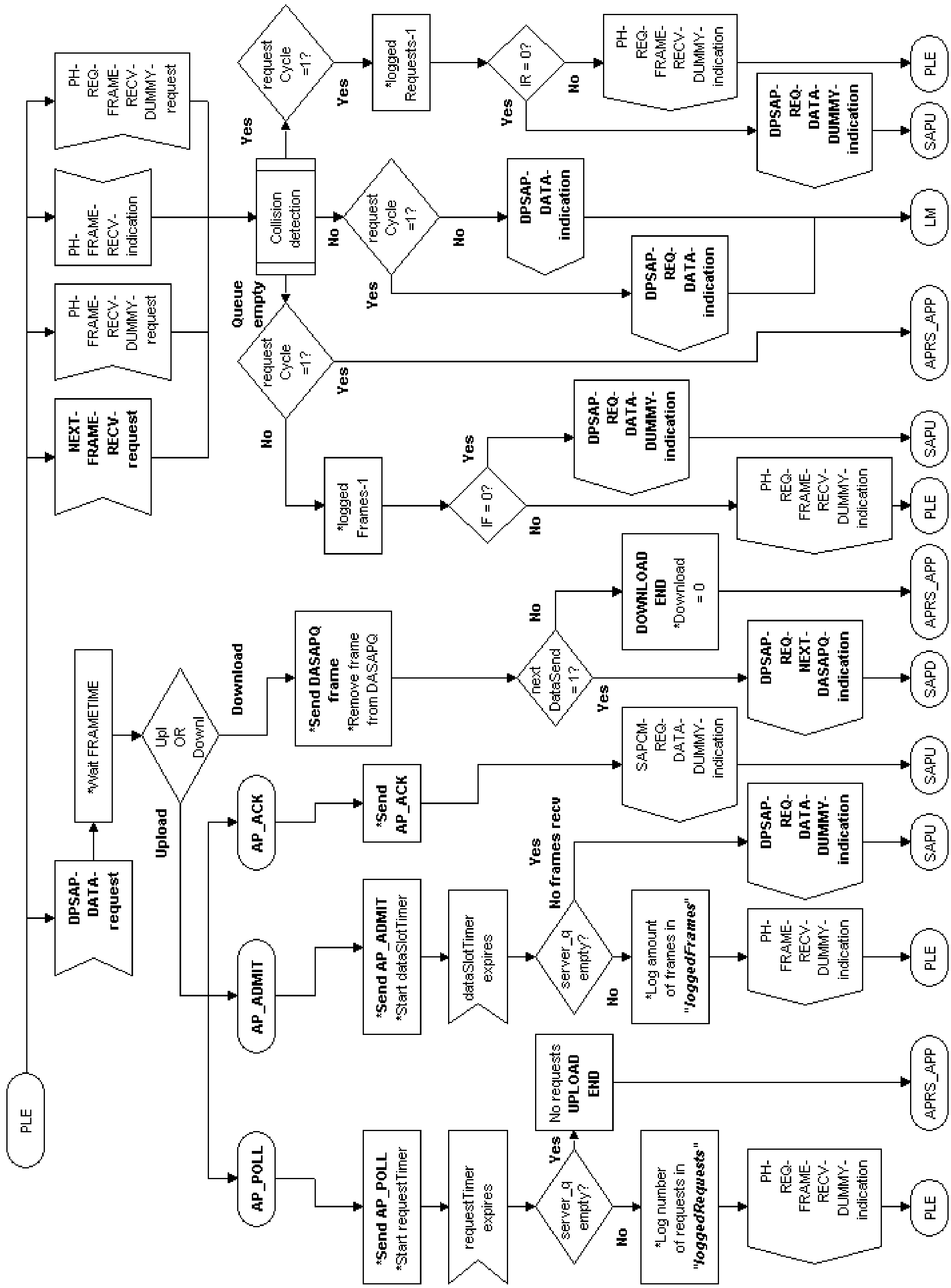


Figure B.6: Access Point Physical-link Layer Emulator

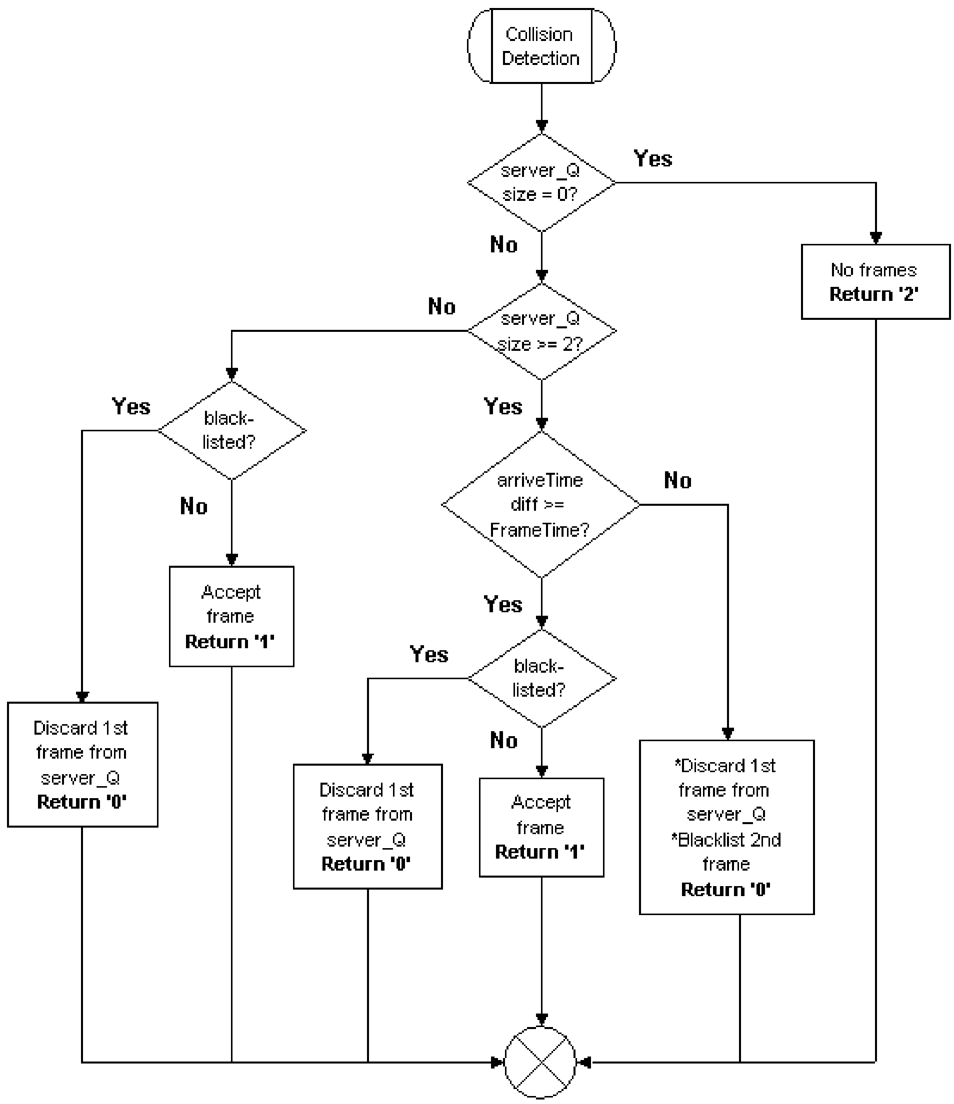


Figure B.7: Collision Detection Subroutine

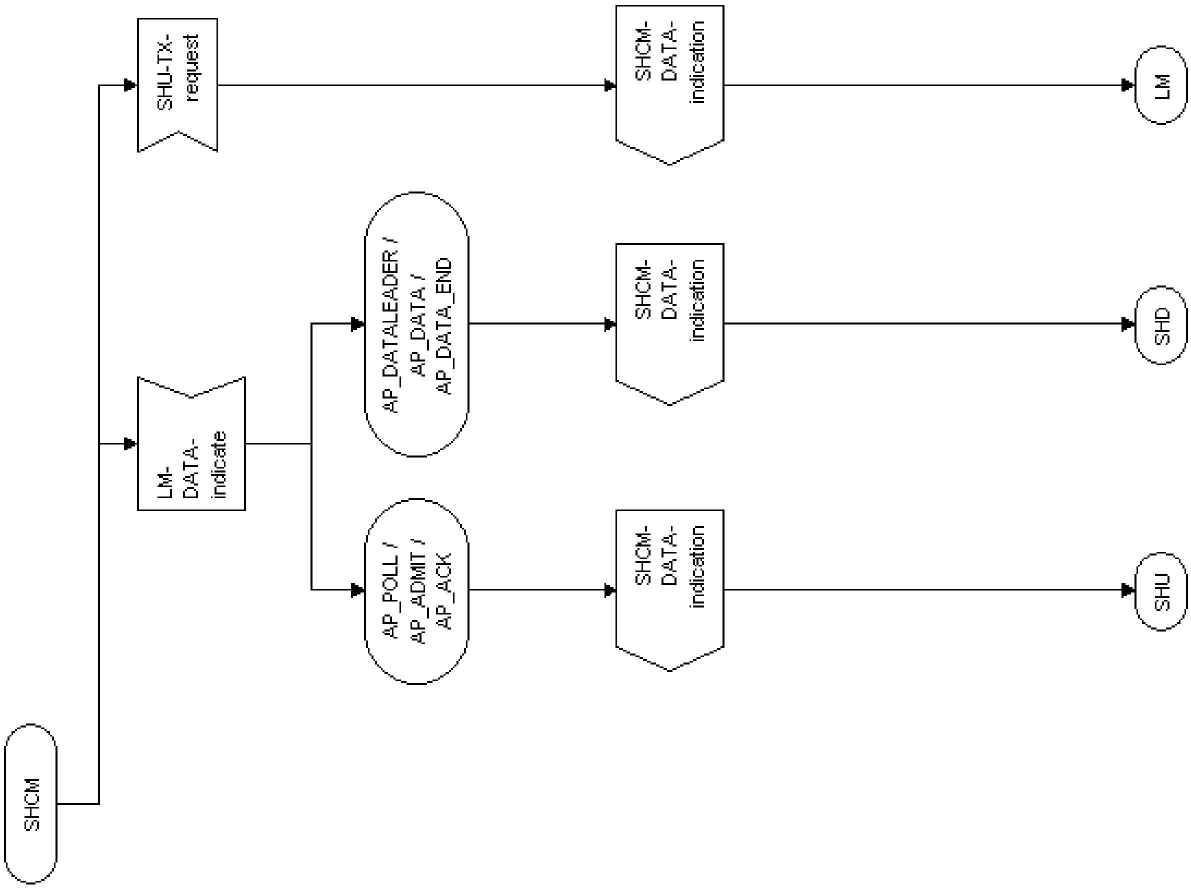


Figure B.8: SRMA Host Cycle Multiplexer

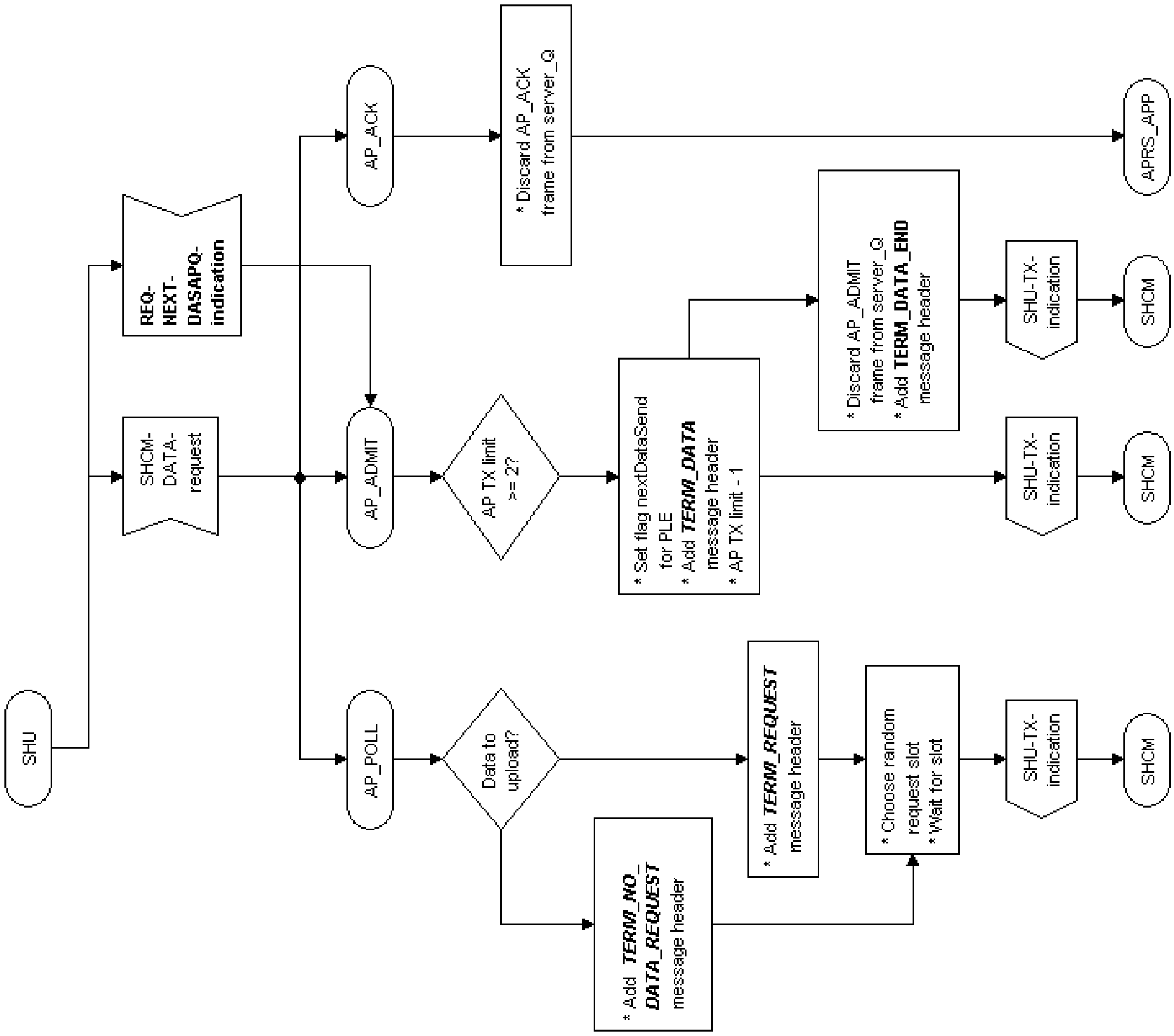


Figure B.9: SRMA Host Upload-cycle

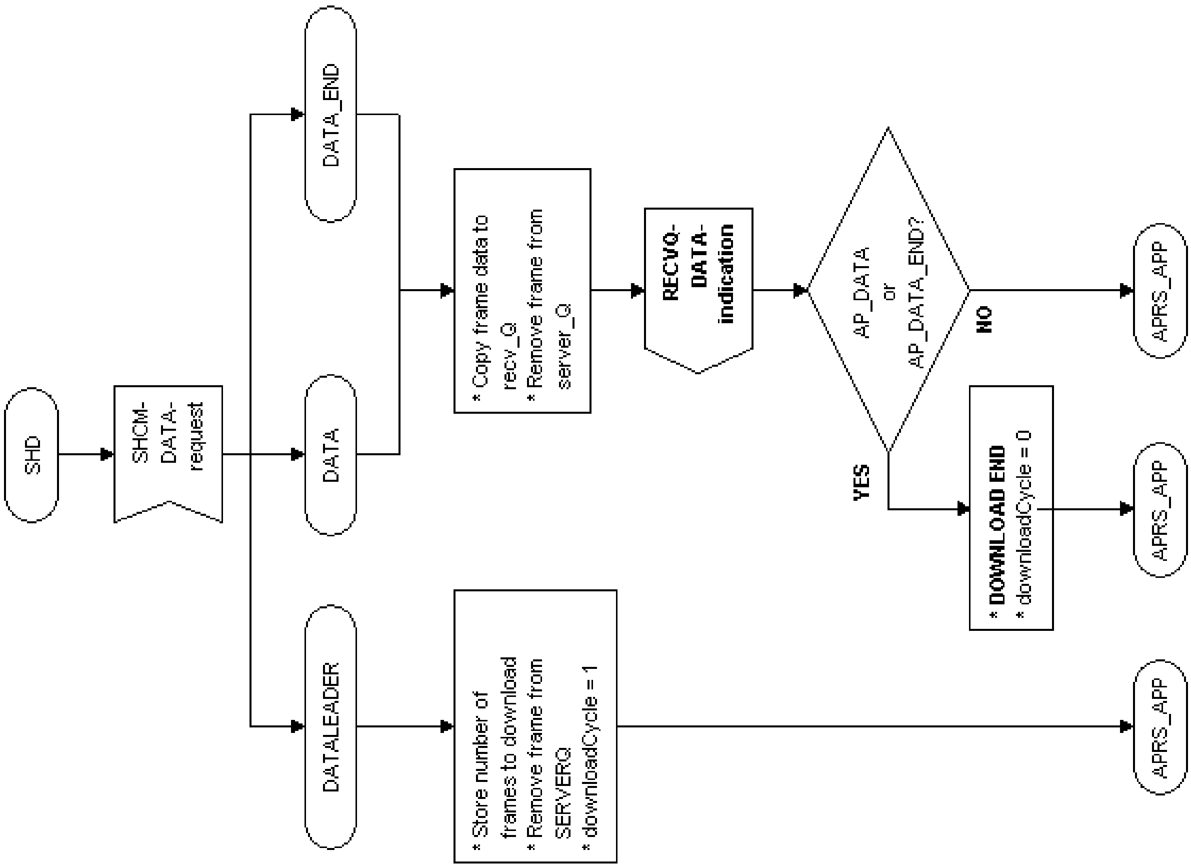


Figure B.10: SRMA Host Download-cycle

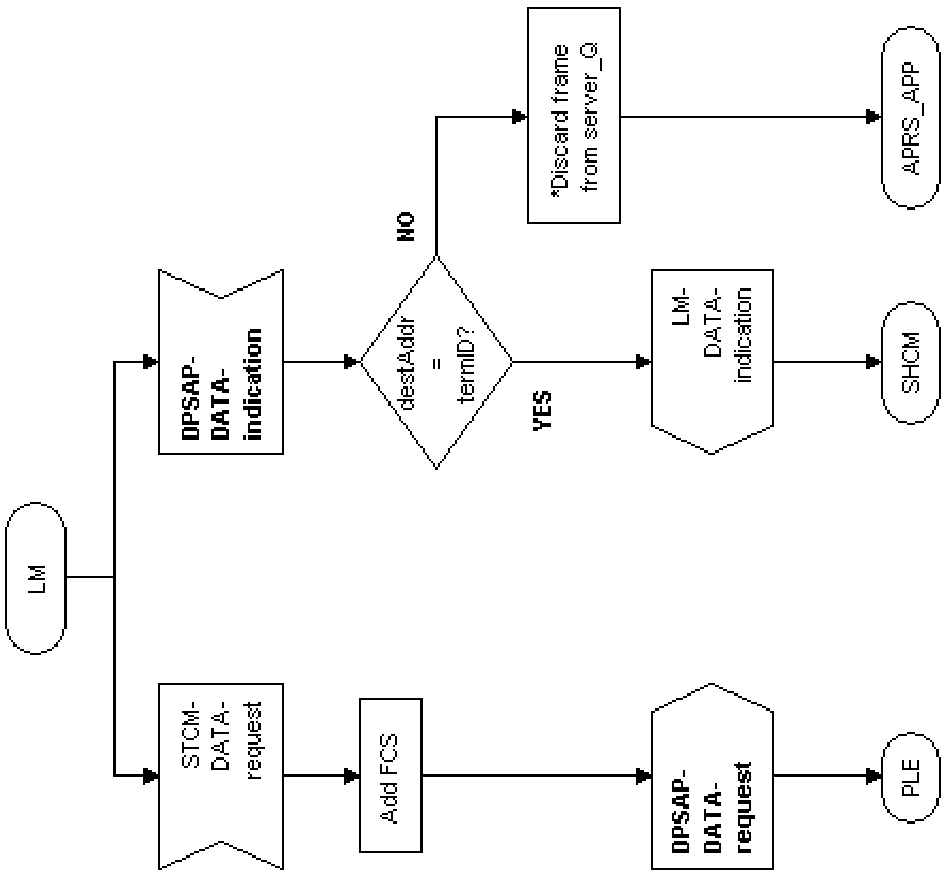


Figure B.11: Host Link Multiplexer

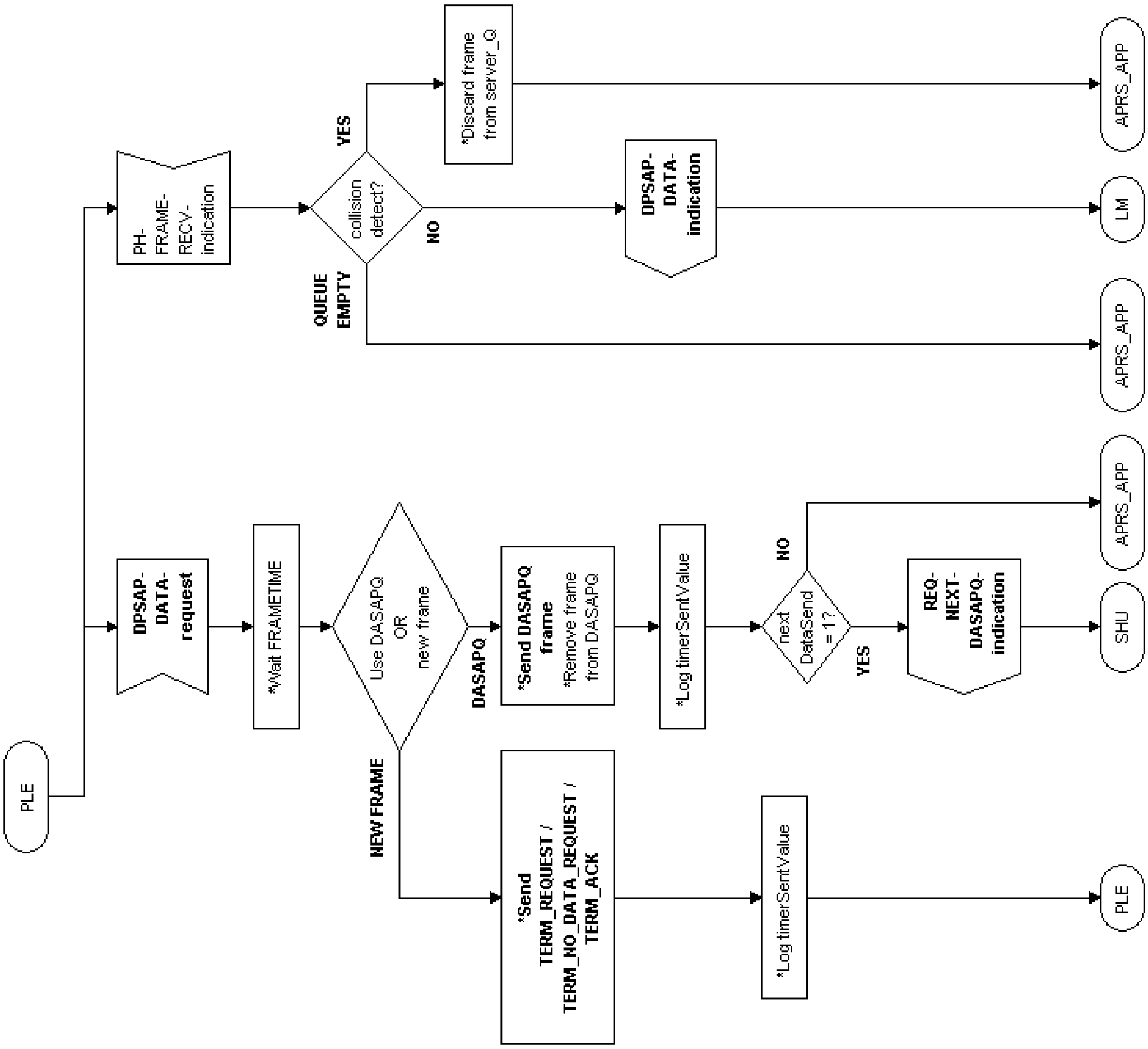


Figure B.12: Host Physical-link Layer Emulator

Appendix C

Implementation Code and UML HTML

C.1 Overview

The enclosed CD contains the implementation code for S-APRS and the UML html pages containing the complete UML S-APRS protocol model.

Bibliography

[1] Tobagi, F. and Kleinrock, L.: Packet switching in radio channels: Part I - carrier sense multiple access modes and their throughput delay characteristics. *IEEE Transactions on Communications*, vol. 24, pp. 1400–1416, December 1975.

[2] Tobagi, F. and Kleinrock, L.: Packet switching in radio channels: Part II - the hidden terminal problem in carrier sense multiple access and the busy tone solution. *IEEE Transactions on Communications*, vol. 24, pp. 1417–1433, December 1975.

[3] Tobagi, F. and Kleinrock, L.: Packet switching in radio channels: Part III - polling and dynamic split-channel reservation multiple access. *IEEE Transactions on Communications*, vol. 24, no. 8, pp. 832–845, August 1976.

[4] Abramson, N.: The throughput of packet broadcasting channels. *IEEE Transactions on Communications*, vol. 25, pp. 117–128, January 1997.

[5] ITU-T: *Information Technology - Open Systems Interconnection - Basic Reference Model: The Basic Model, Recommendation X.200.* ITU Telecommunication Standarization Sector, 1994. [Online]. Available: http://www.iso.org/iso/en/ISOOnline.openerpage.

[6] IEEE Std. 802.3-2002: *Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications.* The Institute of Electrical and Electronic Engineers (IEEE), March 2002. [Online]. Available: http://standards.ieee.org/reading/ieee/std/lanman/restricted/802.3-2002.pdf.

[7] Peterson, L.L. and Davie, B.S.: *Computer networks: A Systems Approach.* 2nd edn. Morgan Kaufmann Publishers, San Fransico, 2000.

[8] Abramson, N.: Packet switching with satellites. In: *National Computing Conference, AFIPS Conference Proceedings*, pp. 695–702. October 1973.

[9] Oliver, M. and Escudero, A.: Study of different CSMA/CA IEEE 802.11-based implementations. In: *EUNICE.* 1999. [Online]. Available: www.eunice-forum.org/eunice99/027.pdf.

[10] Jangeun Jun, P.P. and Sichitiu, M.: Theoretical maximum throughput of ieee 802.11 and its applications. In: *Second IEEE International Symposium on Network Computing and Applications.* April 2003. [Online]. Available: http://www4.ncsu.edu/j̃jun/index_files/full_tmt.pdf.

[11] Jangeun Jun, P.P. and Sichitiu, M.: Improving the aggregate throughput of access points in ieee 802.11 wireless lans. In: *28th Annual IEEE International Conference on Local Computer Networks*. October 2003. [Online]. Available: http://www.eecs.berkeley.edu/ẽrgen/docs/aggregate.pdf.

[12] Karn, P.: MACA - A new channel access method for packet radio. 1990. [Online]. Available: http://www.cal-net.org/aprs/APRS101.pdf.

[13] Grobler, F.: *Design and Implementation of a Store-and-Forward Communication System on the Sunsat Satellite*. Master's thesis, University of Stellenbosch, 1996.

[14] Bruninga, B.: Automatic packet reporting system digipeaters version 8.5.0. September 2003. [Online]. Available: http://web.usna.navy.mil/ bruninga/APRS-docs/DIGIS.TXT.

[15] D. Towsley, J.K. and Pingali, S.: A comparison of sender-initiated and receiver-initiated reliable multicast protocols. *IEEE Journal on Selected Areas in Communications*, vol. 15, April 1997.

[16] TAPR: *AX.25 Link Access Protcol for Amateur Packet Radio Version 2.2*. Tucson Amateur Packet Radio Corporation, November 1997. [Online]. Available: http://www.tapr.org/tapr/pdf/AX25.2.2.pdf.

[17] OMG: *Unified Modeling Language Specification Version 1.5*. Object Management Group, Inc, March 2003. [Online]. Available: http://www.omg.org/cgi-bin/apps/doc?formal/03-03-01.pdf.

[18] Pärssinen, J.: Classes of communication systems. 1999. [Online]. Available: http://www.vtt.fi/tte/staff/ttepap/ papers/Parssinen-ccs-Lre99.pdf.

[19] J. Pärssinen, N. von Knorring, J.H. and Turunen, M.: Uml for protocol engineering - extensions and experience. 2000.

[20] Pärssinen, J.: *Conceptual Modelling of Protocol Systems*. Master's thesis, Helsinki University of Technology, November 2003.

[21] Simpleton, B.: Amateur radio trader. [Online]. Available: http://www.artscipub.com/simpleton/simp.range.html.

[22] Peterson, L.L. and Davie, B.S. (eds.): *Space mission analysis and design*. 3rd edn. Microcosm Press and Kluwer Academic Publishers, San Fransico, 1999.

[23] Kleinrock, L. and Lam, S.: Packet switching in a slotted satellite channel. In: *National Computing Conference, AFIPS Conference Proceedings*, pp. 703–710. October 1973.