Implementation of WiFiRe and rural WiMAX

M.Tech. Project First Stage Report

Submitted in partial fulfillment of the requirements
for the degree of

Master of Technology

by

Janak Chandarana

Roll No: 05329R04

under the guidance of

Prof. Sridhar Iyer and Prof. Anirudha Sahoo

Department of Computer Science and Engineering
Indian Institute of Technology, Bombay
Mumbai

Acknowledgements

I would like to express my sincere gratitude toward my guide Prof. Sridhar Iyer and my co-guide Prof. Anirudha Sahoo for their teaching and constant support. I am extremely thankful to Dr. Ajay Bakre for his constant encouragement and invaluable guidance. I thank my colleagues Jeevan and Ranjith for their insightful suggestions. I also thank Intel Research and CEWiT, India for providing me platform to work on WiMAX and WiFiRe respectively.

Janak Chandarana
IIT Bombay

Abstract

Long range wireless for data and voice connectivity is being considered as viable and affordable solution for rural India since few years now. Numerous solutions were presented to bridge the digital divide; WiFiRe is one of them. Here, we discuss about implementation of WiFiRe MAC done so far and its effectiveness. We propose several improvement in design, development and strategy used to implement WiFiRe MAC. Plan for integration of 802.11b PHY is briefly described. We also present rural WiMAX, project at Intel Research where development of 802.16d based WiMAX devices is in progress. In rural WiMAX, we present 5116 WiMAX chip architecture, co-relation with 802.16 standard and few enhancements done by us. Some interesting open research problem which I came across in WiFiRe and WiMAX has been presented at the end.

Contents

Chapter 1

Introduction

India is a large country with about two hundred million homes. As we have seen, cellular market grew at right price point. Same might be the case for broadband connectivity. As described in [2], average user can not spend more than Rs. 300 per month for Internet. There is need for affordable and easily available devices using which broadband access can be spread to every village in India. WiFiRe[13] presents one such solution by using cheap 802.11b chipset but modifying MAC for longer range and better efficiency. Basic implementation of WiFiRe MAC is in progress and we have pointed few modification in development process. Current implementation runs on NICs and we intend to run WiFiRe MAC considering 802.11b hardware chipset by extending current MAC. Section 3 presents integration plan with 802.11b.

We also present rural WiMAX project at Intel Research and device used for the project. WiFiRe and rural WiMAX's MAC design is very similar to each other and we were able to test our code on working devices of WiMAX. We describe Intel's 5116 WiMAX chip's hardware architecture and how different modules are connected with each other. While working on this device, we have added basic scheduler, QoS and point-to-multipoint support in WiMAX chip. Section 4 gives insight in this project. Actual WiMAX devices which we used to test new MAC enhancement gave realistic view of networking device's internal. We came across few interesting problems while working for WiFiRe and rural WiMAX. Brief description of them is given in section 5.

Chapter 2

Literature Survey

WiMAX has been interesting topic in Indian research community. We are considering work done in WiMAX scheduling and QoS, WiMAX implementation and long range WiFi.

2.1 WiMAX scheduling, CAC and QoS

This section includes MTech projects done in WiMAX at IIT. They provide closer look at some QoS and scheduling designs for WiMAX. QoS guarantee is claimed and separate service queues has been proposed. This section also reminds that there is enough detailed work done in this area and it is good time to evaluate some of these concepts in reality.

An Efficient Call Admission Control for IEEE 802.16 Networks

This paper[3] presents CAC and QoS for WiMAX services. It considers different kind of services like UGS, rtPS, nrtPS and BE. Paper presents CAC architecture on BS and SS considering different QoS parameter. Bandwidth based CAC does not consider delay requirement and priority for real time traffic. This paper adds delay into CAC's consideration. Simulation is done using C on Linux platform. Simulation considers lots of input parameters like maxrate, minrate, total number of slots, nominal grant interval, jitter, polling interval etc. CAC considers all these input parameters and takes decision based on existing data it has. The main question here is that, how do client actually give this inputs to WiMAX CAC. 802.16 specify convergence sublayer which deals with different upper layer protocol. This CS layer will parse TCP packets and retrieve those service parameters. Parsing and storing them in database will take considerable amount of time. 802.16 devices will be having small processor and limited amount of memory to use. For VoIP kind fo traffic, it may happen that total time spent in processing this parameters may be more than total QoS benefit received by packet (in terms of ms). To prove this fact, we can have small performance based module which runs on actual hardware and takes few parameter into consideration. Then we can formulate equation like: for X parameters time taken is Y. For Z parameters, what will be the processing time. Paper presents some innovative ideas about different service queues, periodic grant generator and pseudo code to prove algorithm proposed.

An Efficient QoS Scheduling Architecture for IEEE 802.16 Wireless MANs

Paper[11] presents Efficient QoS scheduling architecture in 802.16 networks. It tires to provide delay and bandwidth guarantee, fairness maintenance and higher bandwidth utilization. Simulation is done in Qualnet and contains modules like GPSS, TDD frame, scheduling services, aggregate bandwidth requests etc. All nodes are using 802.16 MAC and 802.11b PHY layer. Main traffic is VoIP, FTP, telnet etc. This work was done just after introduction of 802.16 standards; it was one of the very initial works in WiMAX MAC scheduling implementation. Author used min-max fair allocation for uplink scheduling and WFQ for downlink scheduling. It introduces MAP generator, grant generator, data classifier, scheduler and traffic shaper.

Implementation and Evaluation of a MAC Scheduling Architecture for IEEE 802.16 Wireless MANs

In this paper[10], author used WFQ for both uplink and downlinks. He simulates his proposed algorithm using NS-2. Paper claims to provide delay bound scheduling for real-time traffic. Author consider GPC mode because he believes that connections are more important than number of SS. NS-2 simulation with modules for TDD frame structure, GPC mode and bandwidth allocation, Ranging request and type of services are described. NS-2 architecture is modified slightly to support WiMAX modules. Results describe effect of one type of flow to other flow and choice of correct bandwidth contention period.

2.2 Long range WiFi

802.11 based wireless is used for indoor use since last decade. It has been mainly used for home use but never considered as option for long range or broadband access. As 802.11 based devices became really cheap, researcher started trying to use 802.11 based RF chipset for long range communication. WiFi based networks are important for us because WiFiRe also uses 802.11b chipset for PHY layer.

Rethinking Wireless for the Developing World

Paper[9] describes technical and non-technical challenges associated with wireless deployment in developing regions. Paper looks at larger scenario where they think about long wireless links, affordable pricing model, intranet usages, traffic support etc. Authors also describe some working deployments in India, Ghana and San Francisco. WiFi-based Long Distance (WiLD) links provides many interesting challenges like routing, interference, multi-path etc. It also describes ACK timeouts, collisions, scheduling in Wireless links and various QoS mechanism. Overall, paper had compiled all major issues concerned with wireless deployment in developing regions.

Turning 802.11 Inside-Outs

This is probably first paper[12] to describe long range wireless in India. Paper describes digital divide, Indian telecom sector and cellular wireless technology in India. DGP[12] is having a network of eight nodes and eight point-to-point links. The longest point-to-point link spans over 38km. It claims to get voice (using VoIP) and cheap Internet to villages. DGP describes various technical challenges like PHY performance in outdoor channels, power efficiency, 802.11 MAC issues, timeouts, contentions, routing etc. Author presents real life deployment and problems involved in making all pieces working. Paper is highly cited in other such deployments.

2.3 Implementing WiMAX

The Design and Implementation of WiMAX Module for ns-2 Simulator

This paper[7] describes various modules of WiMAX MAC and simulates them with NS-2. They define clear difference among CS, CPS and CAC functionality. It presents very detailed architecture for 802.16 with different modules and how they interconnect to each other. All modules are designed in C++ and combined with NS-2. Simulation also gives very specific function definition, variables etc. It also gives ranging techniques, different response and situation which can happen in ranging etc. It also covers modulation being used and PHY parameters. Paper was very helpful as reference in writing code. Performance evaluation is done using realistic parameters and presents overall idea of WiMAX MAC.

Chapter 3

WiFiRe

WiFiRe stands for WiFi Rural extension. It uses licence free 2.4 GHz spectrum and cheap 802.11b RF chipset as PHY layer. It replaces 802.11b MAC mechanisms (DCF/PCF), with long range MAC (like 802.16[1]), keeping 802.11b PHY same. WiFiRe is a star topology - a Base Station (BS) at the fiber Point of Presence (PoP) and Subscriber Stations (ST) in the villages nearby with 6 directional and sectorised antennas at the System. It follows TDM frame structure which is similar to WiMAX and GSM. WiFiRe is promoted by CEWiT, India and project is spread across IIT Bombay, IIT Madras and IISc Bangalore.

3.1 First phase of Implementation

WiFiRe's MAC software stack has been partially implemented at IIT Bombay [8]. This work has been with the assumption of programmable 802.11b chipset running WiFiRe MAC on top of it. Implementation has taken care for establishing connection, basic packet flow, MAC header construction etc. Implementation was initially using C sockets (on layer 7). Later on, MAC was running as programs on layer-2 and talking with NIC directly. This was done using PCAP libraries in C language. Later on, for simplicity purposes, programs were moved back to layer-7 sockets but injecting on NIC directly this time. This (layer-7 implementation) gave opportunity to integrate other programs with PCAP modules. Layer-2 code was not flexible as it was sending MAC packets from BS to SS where program does not have much control on NIC's output. The latest C sockets works like peer to peer application and very easy to integrate with WiFiRe's MAC code. Simulation's behavior and packet structure remains same in layer-2 and layer-7's code. It can be verified using tcpdump or ethereal filters on respective NIC.

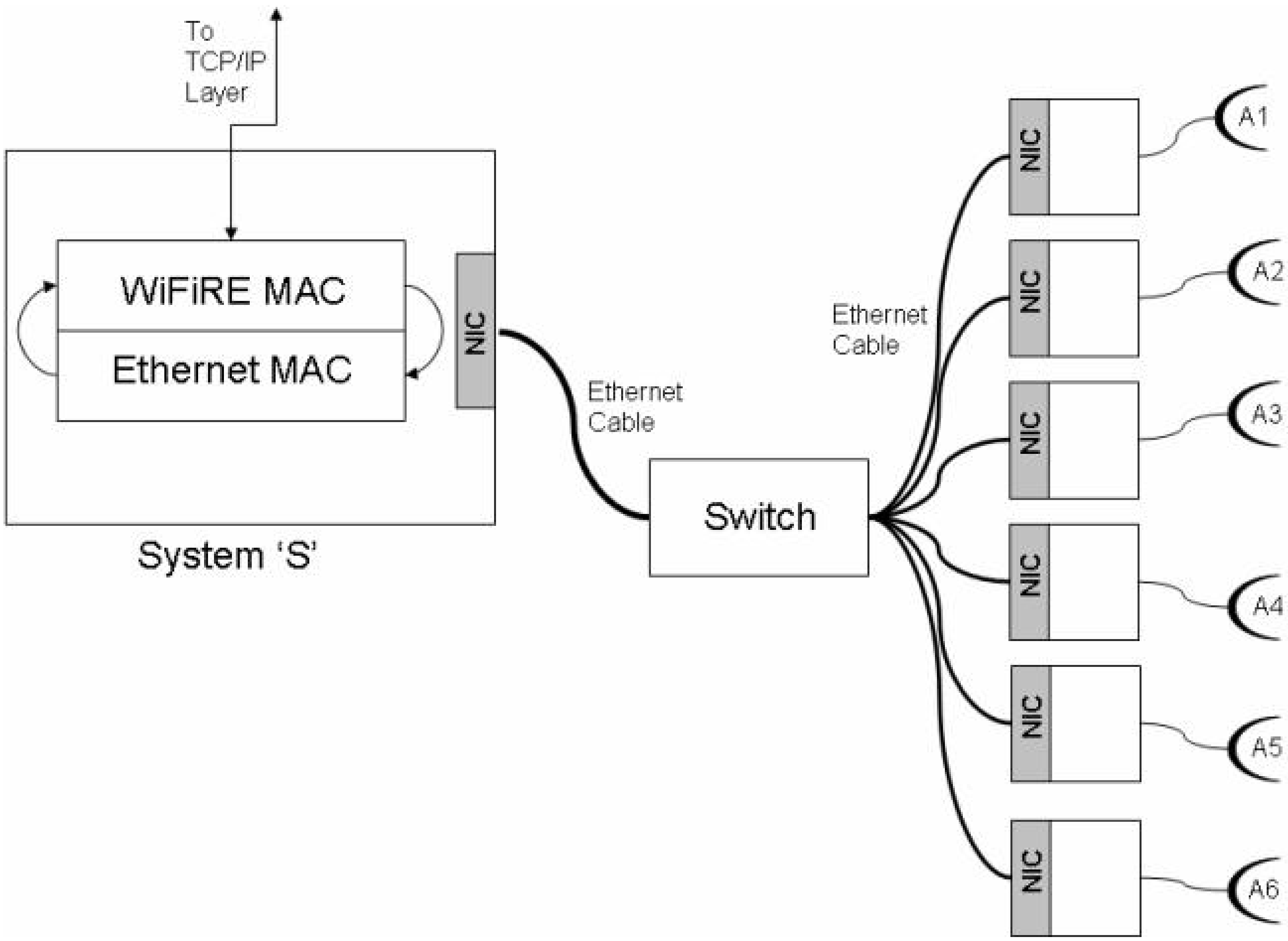

Figure 3.1: WiFiRe MAC and 6 BSs

Along with this, basic scheduler, classifier and MAP generator were implemented for WiFiRe uplink and downlink which will take care for incoming and outgoing packets at BS and ST. This work was dealing with various kinds of layer-2 protocol details and took considerable amount of time to come-up with proper design. Status of scheduler and MAP generator is still not clear. Scheduler and classifier are invoked for every single packet. This means that, all packets will go through same cycle even though they may be sharing common properties. This module is more concerned about maintaining CID tables, exchange of data among different data structures and understanding ULP's (upper layer protocol) requirement.

Implementation of MAC is using gcc in Linux where program is running in user space. Linux terminals with 2-3 NICs are working as BS and STs. This whole scenario gives us emulation of actual protocol. Many network parameters (static IP, ARP cache, DNS, proxy etc.) were kept fixed to keep design as simple as possible. In most of the cases, BS and STs were made layer-2 device where any packet going to layer-3 or above will be ignored. For the first phase of implementation, it is good approach but final goal as a product is the small chip running WiFiRe MAC inside.

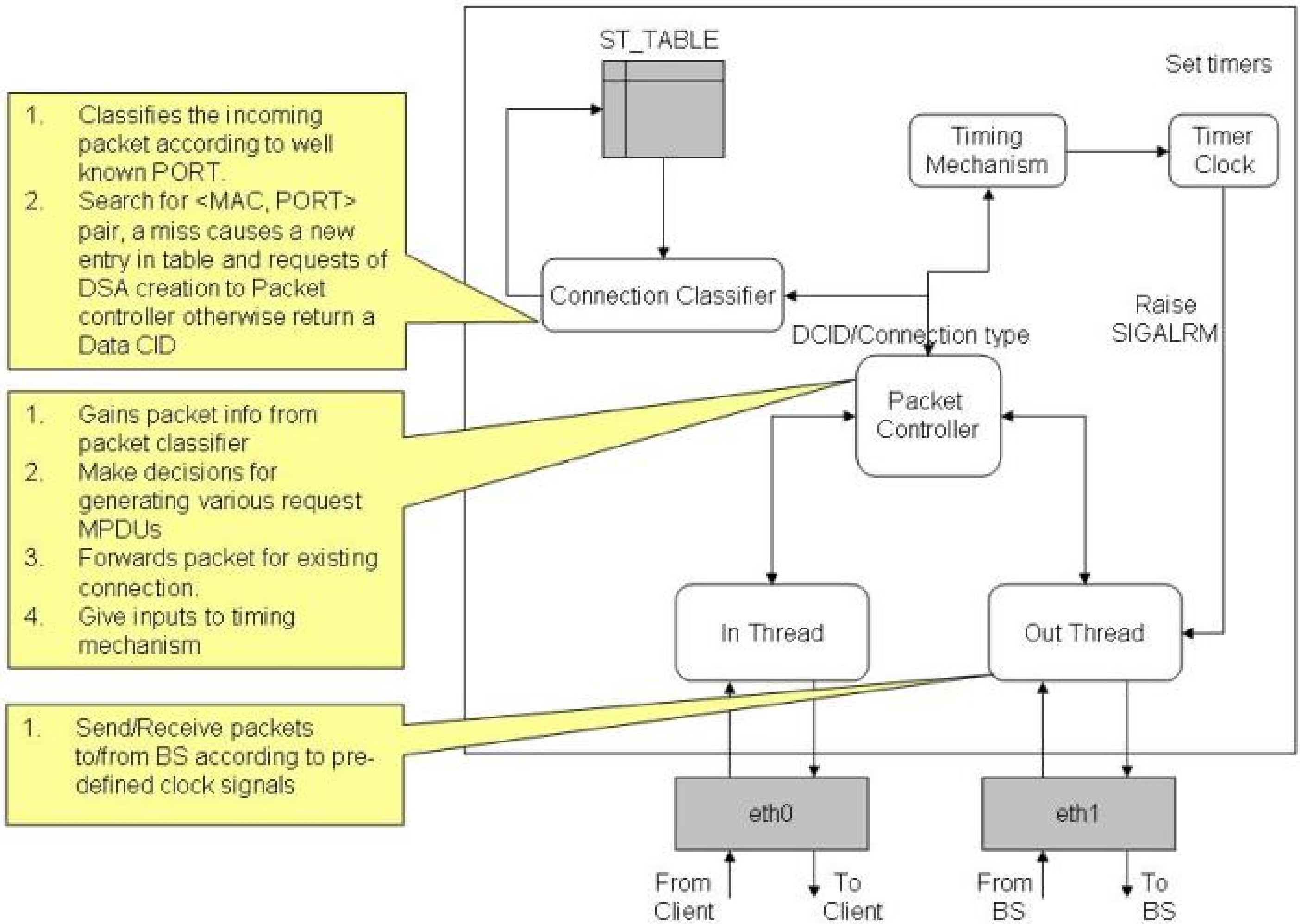

Figure 3.2: classifier on SS with threads

Four kinds of QoS classes are considered as per 802.16's specification: UGS, rtPS, nrtPS and BE. There is respective traffic pattern to match but none of them has been actually implemented. Requesting bandwidth for new connection has DSA support but its implementation is not tested. Traffic generators for real time data haven't been considered as QoS support is still far away. VoIP traffic is one of the main targets for WiFiRe's services. WiFiRe draft has given enough thought and precedence to VoIP traffic. Many design issues were taken keeping VoIP traffic in mind. VoIP's timeout and bandwidth requirement can be satisfied using current WiFiRe MAC.

BS connection with ST is having DIX based (similar to 802.3) Ethernet. This configuration depends on NIC drivers being used, connecting media and firmware's support for flexible frame structure. This link is treated as wireless link of WiFiRe. As our simulation runs on Ethernet and with PC connected directly, it can't get total flexibility on frame structure, size, CRC etc. This link restricts size of individual packets and mpdus as well. As of now, both BS and ST are kept next to each other which eliminate problems of propagation delays, ranging, synchronization and other such wireless specific issues. We also assume that some of those wireless specific issues will be taken care by underlying hardware in actual implementation. As of now, they are ignored for obvious reasons.

BS and STs handle packets based on two threads: OUT-THREAD and IN-THREAD (See figure 3.2). Initially, these mechanism lead to recursive Tx and Rx calls. When program captures packets on BS's NIC-1 which is connected to outside world, packet get processed and given to NIC-2. When NIC-2 tries to send packet to ST, it will be captured again and this recursive cycle goes on. By applying 'smart' filters on PCAP, we can remove this recursive property. To keep design simple, there are no shared variables or semaphores used for co-ordination among different functions. No hardware based interrupts and triggers are supported because there is no predefined hardware property as of now.

There are many clients connected to ST using layer-2 switch. ST is able to recognize and treat clients differently. CID is given on per client per TCP connection basis. For example, if client have one VoIP and one FTP connection, there will be 2 CIDs for that client. If client have 2 VoIP and 2 FTP connections running, it will get 4 CID in connection database. CS layer in WiMAX specification takes care for different connections. Similarly, WiFiRe also keeps a pool of CIDs in CS layer. Current CS layer can be interfaced with 802.3 based media only.

3.2 Approaches and assumptions taken

WiFiRe presents classical networking project where many of theoretical concept came into real application and gave wide range of design options. It also represents typical software engineering problem where design, execution and review cycle comes in picture. WiFiRe draft being initial design of protocol, implementation part considered various strategies for same problem. WiFiRe gives opportunity to see how protocols are built with their development life cycle. Lot of assumptions and dependencies came as implementation is spread across various groups.

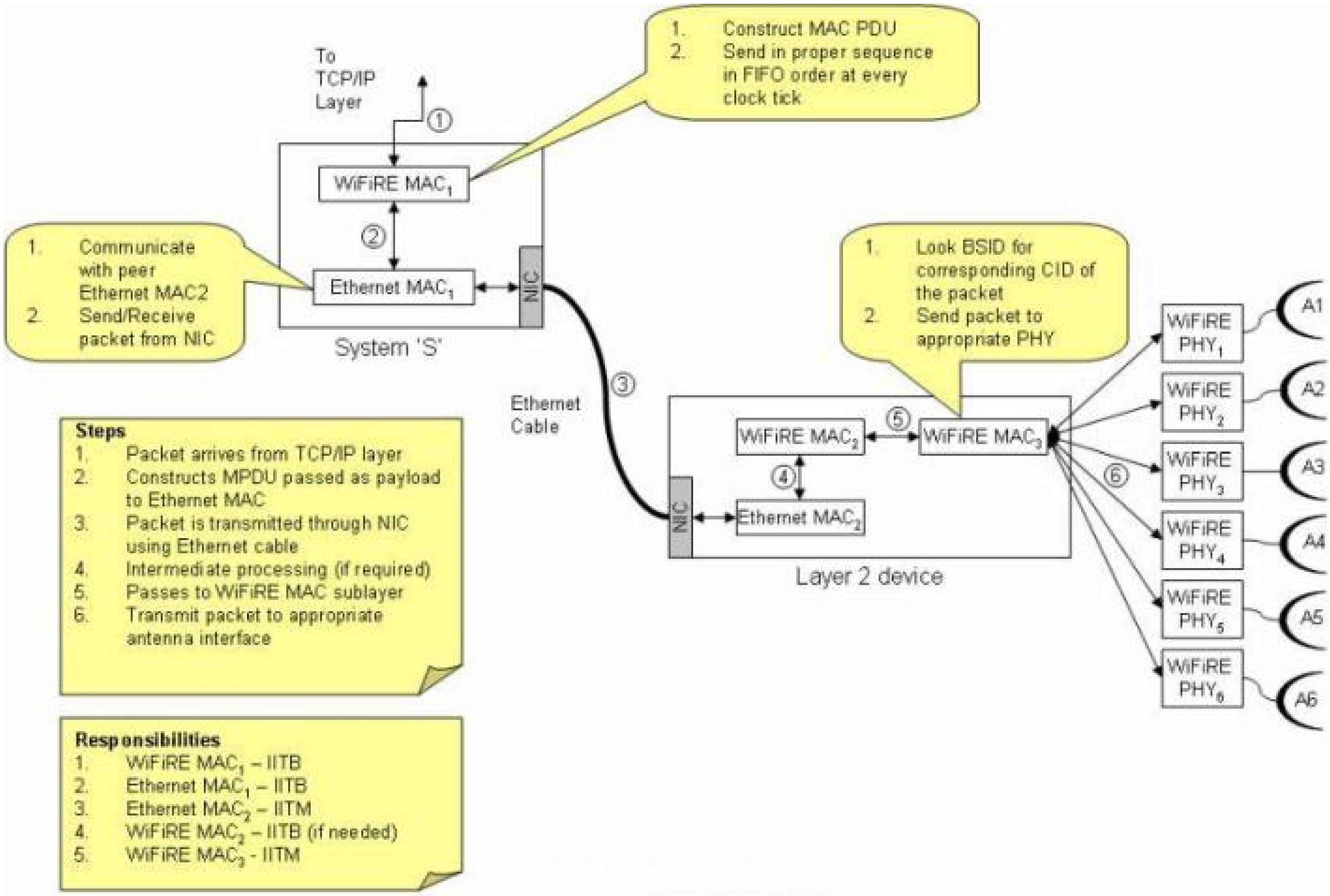

Figure 3.3: MAC and dependency on PHY

WiFiRe's implementation is based on PCAP libraries. PCAP relies on NIC and kernel's configuration heavily. Most of these properties can not be modified to make it WiFiRe compatible. Linux kernel is assuming 802.3 based networks and sets network parameters (like frame length, firmware property etc.) accordingly. Current BS-ST link is also Ethernet and WiFiRe MAC has to be written keeping that (802.3) in mind. The reasons are: separation of hardware from MAC (modularity), good design assumptions (dependency) about chip in WiFiRe draft and unavailability of 802.11b chip. Because of this approach, we concentrated heavily on "802.3 compatible WiFiRe MAC". In reality, it should not matter which physical layer will be used because once 802.11b chip is available, all 802.3 based assumption are not required. Also, current MAC runs on PC (200MB+ RAM and 1GHz+ processing power) which doesn't give realistic idea about actual hardware. In actual implementation, it is very likely that there will be very small flash memory and embedded processor running with RTOS. The alternative approach to that can be use of network processor/chip. There are several chips available like ADMtek AN983B and Intel IXP465 network processor. Many RTOS (like VxWorks) gives network stacks which can be accessed and reprogrammed based on protocol requirement.

WiFiRe's MAC processes packets on different machine as and when required. Let's take an example: Client-1 is connected to ST-1 with layer-2 switch. ST-1 is connected to BS. Now, Client-1 starts VoIP communication and send packet to ST. Here, ST takes that packet from 'normal 802.3' interface. ST will read MAC header of packet, decides its CID and based on UL-MAP, transmit it to next switch which is connected to directional antenna. Antenna (or memory) will store it temporarily and transmit packet when hardware trigger is generated for ST-1. Here, we are having two copies of each packet: at ST and at antenna. Figure 3.4 describes this problem.

Main Disadvantages are: redundancy, delay in processing, one extra hop-count, antenna's inability to access ST's memory etc. Also, it doesn't give any flexibility to change setting at the time of transmission because we are not sharing management function with antenna's end (like MAP generator). The solution to the problem is to use common single memory pool used by ST's MAC and antenna. This process is tricky on BS part because implementing locks on single shared memory among 6 antennas and their respective controller would be really complex.

WiFiRe's MAC uses cross layering at several places. Few Examples: To decide new connection, it tries to check new TCP SYN message. For VoIP, SIP and RTP packets are treated as separate connections. For ARP, there are dummy replies. For QoS, it checks TCP/IP PDU type and decides service flow. WiFiRe also assumes web proxy and DNS server running. Some of these issues were just discussed in design and few were implemented as well. This all leads to not just violation of classical layering approach but addition of great complexity to system. Argument against classical layering approach is that, now a day, almost all vendors are violating this and there is no question about data privacy.

WiFiRe classifier and CAC (which decides QoS type) recognize real time and best effort flows. Now, when ever a new packet comes to BS or ST, classifier must be aware of TCP/UDP structures, addressing parameters, delay requirement etc. Packet's header will go through different filters and several parameters will be captured from it. These parameters will serve as inputs to classifier and CAC. This complex CAC and classifier[3] will take significant amount of time to give output and then packet will be processed further. This processing delay is not desirable for VoIP packet which has very stringent latency requirements. All cross layer assumptions are

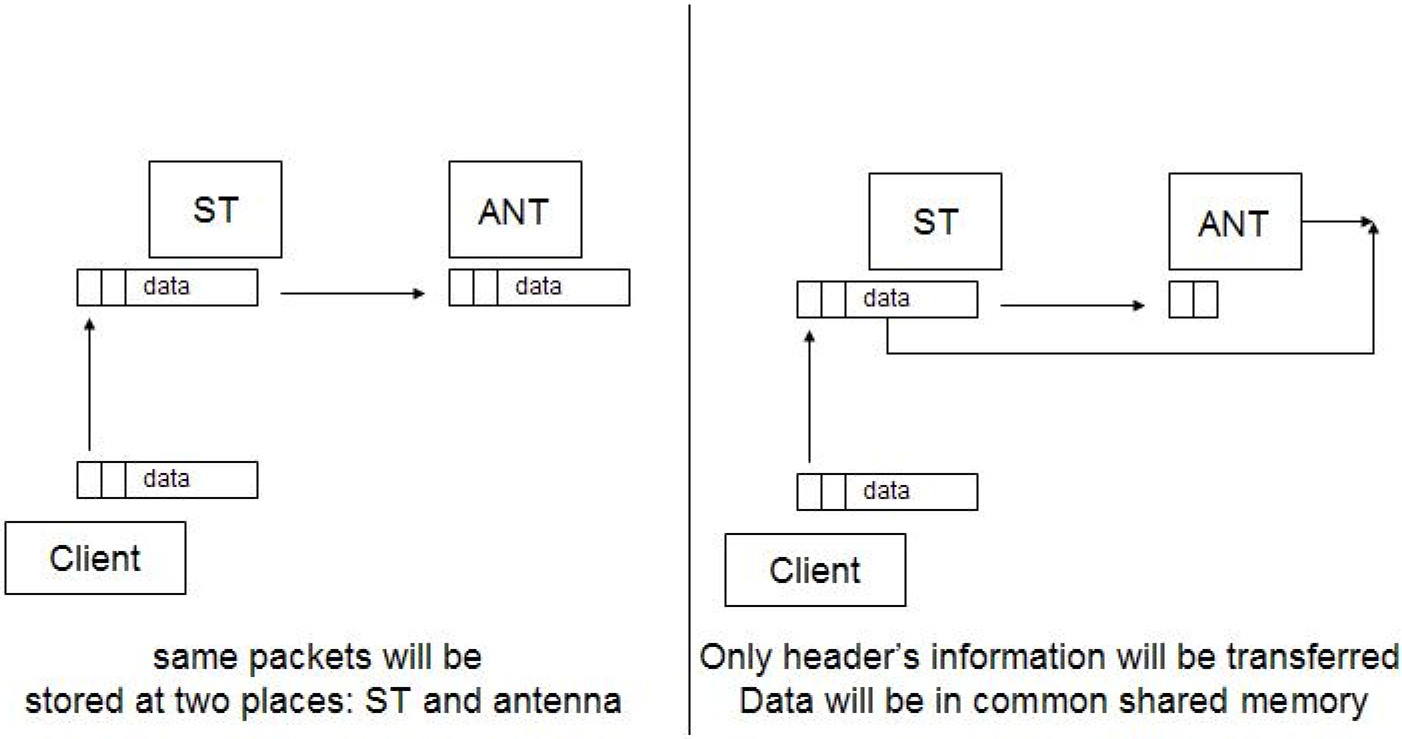

Figure 3.4: data stored in shared memory

invalid if packet is encrypted. Further details are given in Section 5.2.

WiFiRe design is very much similar to WiMAX which has several advantages. Each design and implementation decision can be compared with WiMAX's current status. Mistakes done by WiMAX will not be repeated in case of WiFiRe (like complex and expensive SS). Both have similar layering approach. Modularity and abstraction in those layers is also same. WiMAX is closely 'talking' with hardware while WiFiRe will keep modular approach and 'minimum talk' with hardware. This design decision was very helpful to keep software MAC in developing state and not worrying about hardware.

There are few more debatable things like fixed slot size, PHY overhead, header compressions, VoIP codec, total capacity of system etc. But, first iteration of implementation was supposed to be simple and basic working demo. It had gone in right direction. We will try to accommodate above critics in next phase of implementation.

3.3 Integrating MAC with 802.11b

WiMAX PHY is using OFDM modulation (or similar higher modulation technique) which enables higher bandwidth and long distance communication. It also assumes MIMO antennas in some cases. This PHY will be taking instructions from MAC about what to transmit and receive. BS and SS usually will have same PHY but BS will be having complex MAC running on top of PHY. In case of WiMAX, all the modules (MAC, RF, antenna, modem, and memory) are in same 'box'. Everything is closely coupled with each other and they exchange configuration parameters with each other. There is sophisticated way to communicate among these modules. For example, memory has driver and interface through which we can store/retrieve data to/from buffer. In disadvantages, WiMAX MAC has to be PHY aware. This means that, MAC has to implement functionality which takes configuration parameters from PHY. MAC behavior also changes based on hardware changes. Hardware based triggers has to be supported.

In current WiFiRe implementation plan, WiFiRe MAC and PHY are separate from each other (figure 3.1). They are connected to each other using RJ45 cable(s). Current hardware keeps 6 different MAC address for 6 BSs. WiFiRe frames will be sent to them with 'dst' field as that particular BS's MAC address. Frame structure and slot sizes are fixed. It helps PHY to understand its data easily. There will be centralized module which will instruct all 6 BSs to coordinate with each other and send/receive data as and when required.

Meta frame will be sent to BS and SS from MAC. This MAC might be running in PC or small embedded device. It will be also having fixed entry for all 6 BS's PHY capabilities and IP/layer-2 address. This Meta frame will be parsed by PHY device, divided according to their destination and then each BS will transmit it using its direction antenna. Meta frame expectation from this hardware gives lot of flexibility to MAC. It can be designed independently on different platform and PHY parameters (delay in synchronization, hardware triggers, memory access time etc.) can be ignored.

First phase of implementation was not taking care of any functions related to hardware. Further implementation will definitely need hardware's input which will be used to configure MAC accordingly. MAC need to be aware of frame structure, total bytes which can be sent, PHY overhead, coding scheme being used, STC burst enabled or not, transmission error if any. It should also give interface where we can specify which PDUs are to be received at what time. MAC may need maximum PDU size supported.

3.4 Implementation ahead and future plan

WiFiRe's implementation will be taking hardware into consideration now. It might include network processors, flash memory and programming with hardware requirements. Basic design remains same because hardware will still expect same Meta frame from MAC. If MAC is sending frame in correct format and if MAC is parsing received frame correctly, WiFiRe MAC will be in total working condition. Small term goal and incremental development will help WiFiRe to get targets in neatly fashion. Following are the modules which are supposed to be implemented (in order of priority)

- CID and connection table
- MAP generator
- MAP parsers
- Tx thread and Tx functions for NIC and other network devices
- Rx thread and Rx functions for NIC and other network devices
- Meta frame generator and parser
- Classifier for incoming and outgoing packets
- Scheduler at packet and flow level
- QoS support for real time services
- Ranging for SS
- Contention slots

Following are the functions which are not to be implemented by WiFiRe MAC. There can be additional modules that will provide these functions and WiFiRe may directly use them.

- ULP parsers
- VoIP header compression
- Combining SIP and RTP in single CID.

We also believe that current BS and SS architecture are such that, BS and SS have almost same capabilities. They run almost same MAC with similar RF capacity. Current WiMAX trends suggest that BS are made very complex (and costly) while SS are made as simple as possible (and cheap). SS are made like dumb layer-2 device. It should have low power requirement, negligible intelligence, small size and affordable cost sharing model.

Chapter 4

Rural WiMAX

4.1 Problem definition

802.16d standard is a fixed broadband wireless access mainly intended for BS to SS communication. Newly introduced 802.16e standard (which supports mobility and many other features) has been dominant for a while now. WiMAX forum (www.wimaxforum.org) is one of the main supporters of 802.16e. This new standard will allow individual users to directly connect with Base Station and will speed up the process of adaptation. 802.16e expects users to have WiMAX card in their laptops and other devices. As this is newly introduced protocol, current cost of such card is very high.

If we observe Indian rural setting, it is very unlikely that users will be using those devices for personal use. As described in [6], cost sharing and society running internet access looks much viable option. We expect one or two kiosk every village which might be running community network inside village. For this kiosk, 802.16d based wireless broadband access makes more affordable and feasible option. Currently all major WiMAX devices developed are 802.16e compatible and supports 802.16d as backward capability. Cost of such devices is still high in US and European market. We believe that development of 802.16d based devices will drive BWA adoption in India much faster. GSM, CDMA and other mobile operator may not appose 802.16d because it is not in direct competition with mobile market. License from government to operate and correct business model support for its commercial deployment will allow operators to introduce WiMAX in India.

4.2 Intel 5116 WiMAX chip - architecture

Intel's 5116 WiMAX chip [4] is 802.16-2004 compliant mainly intended for development of cost-effective CPE. It can be combined with third party RFIC and ODMs. It includes reference drivers for radio, Ethernet, TDM devices and modem APIs. It supports TDD and FDD mode with BPSK, QPSK and QAM modulations. PHY can give SNR and RSSI as feedback to upper layer. It has Dual-core ARM engine for PHY, MAC and protocol processing. It supports AES and DES security standards. It has 802.3 Ethernet MAC embedded with MII interface. It has TDM application support with RJ11 interface. It also has SDRAM and flash memory for storage.

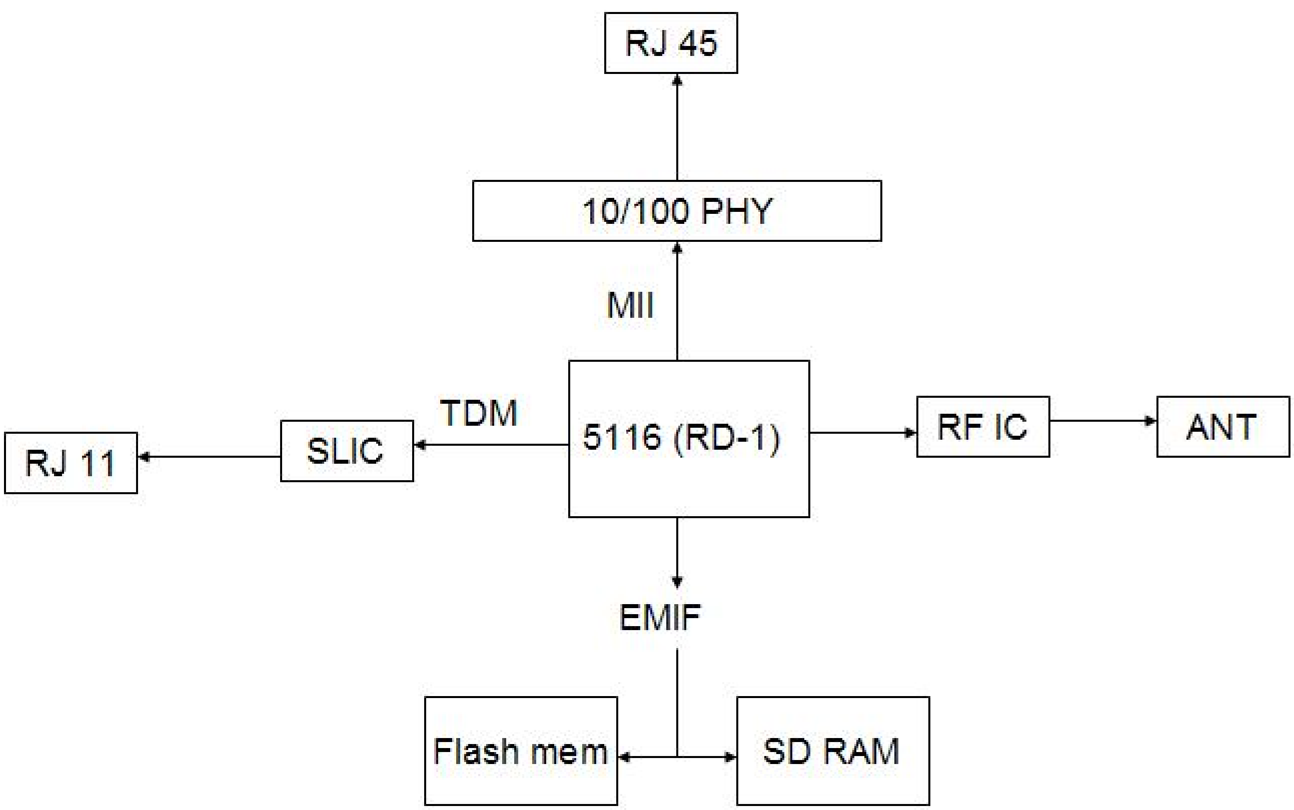

Figure 4.1: 5116 Architecture

5116's MAC is designed from 802.16d MAC recommendation. All the important functions for 'basic working prototype' are present in current code. It also follows layering approach described by 802.16 groups. Just like 802.16d, it has convergence sublayer (CS), common part sub layer (CPS) and lower layer MAC. CS deals with different CIDs and connection database. It also interacts with Ethernet driver to send and receive packets. Client connected with BS or SS by RJ45 connector will send and receive packet to Ethernet and driver.

CPS contains functions for MAP construction and parsing, Rx and Tx thread and LLM interaction. LLM actually talks with modem to transmit and receive data. There are few other drivers for Ethernet and memory buffers.

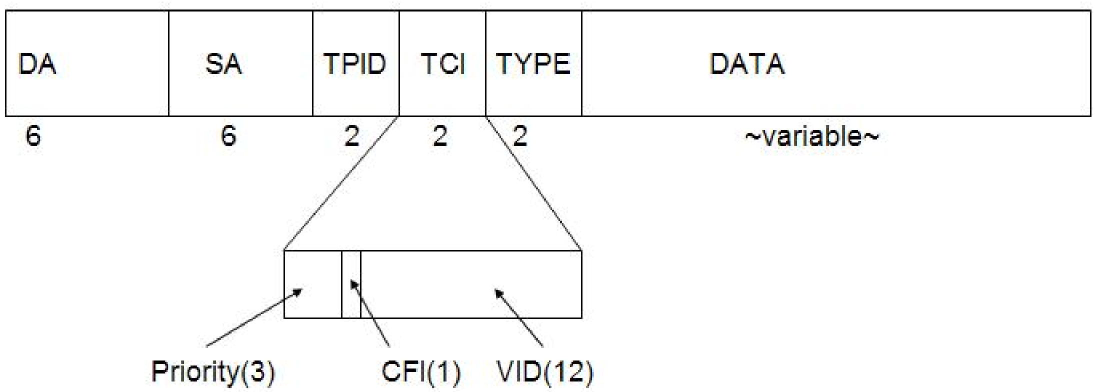

Figure 4.2: 802.1Q VLAN packet format

4.3 Enhancing 5116 MAC

5116 has point to point support and provides extension for PMP mode as well. 802.16d does not define scheduler, CID database and classifier design. These design decision are left for vendor-specific implementation. Currently, there is no QoS support in 5116 chip. We do not have licence for WiMAX frequency (even for testing) because of government regulations. 5116 chip is not tested for point-to-multipoint and may give surprising results. We have added basic Scheduler and Classifier in WiMAX MAC. We assume layer-2 frame and no cross layering policy here. The main reasons for that are data privacy, service provider's regulations, pricing models and simpler design. Classifier and QoS is described in figure 4.3 and 4.4.

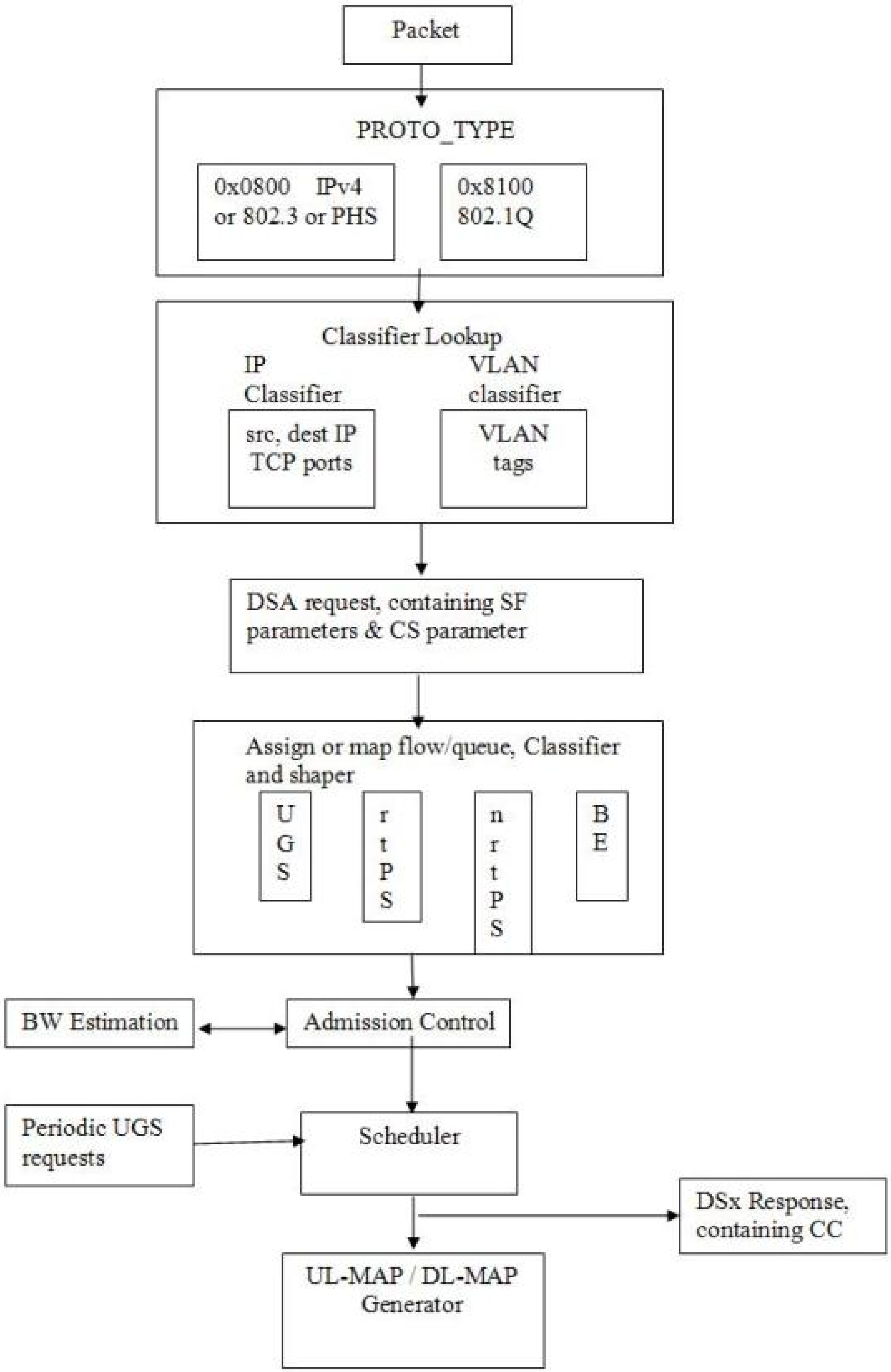

Figure 4.3: QoS using 802.1Q

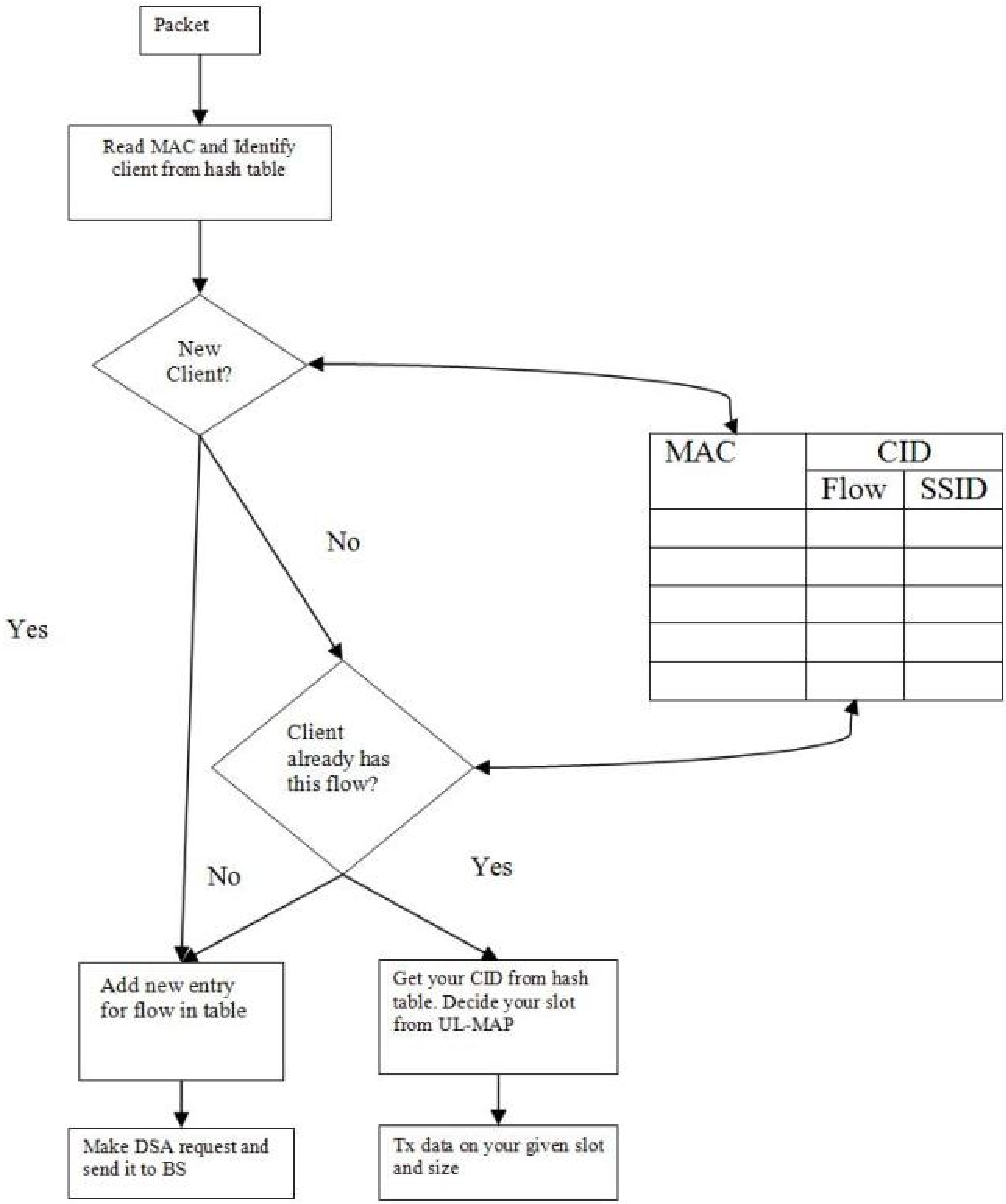

Figure 4.4: classifier on SS

Chapter 5

Interesting research problems in WiMAX

While working for WiFiRe and rural WiMAX, We came across some problems which are not described in 802.16 recommendation. Although, there is active work going on in this direction. Some of these are open research problem and can be explored if time permits.

5.1 Generic Packet Convergence Sublayer (GPCS)

It is difficult for the industry to accept a set of 802.16 convergence sublayers that all devices must implement to be called 'WiMAX compliant'. For example, if Ethernet CS, why should a phone have to implement Ethernet frame formats? Why device need to participate in IP address assignment, IP mobility, and tunneling, etc if its 802.16d ST? And if a vendor implements a proprietary upper layer protocol, how can its 802.16 layer are tested to be compliant? GPCS [5] suggests that a generic packet convergence sub layer can help 802.16 CS by simplifying a 'compatibility' that is independent of the upper layer protocol.

The 802.16 convergence sublayers do not define the capability for multiple upper layer protocols to transport the SDUs on a single 802.16 connection. A connection ID (CID) is a valuable resource in both base stations and subscriber stations. An 802.16 system should not be forced to open a different CID for each upper layer protocol if packets for upper layer protocols have the same QoS requirements (802.16 scheduling service types). The generic convergence sublayer provides a simple way to transport multiple protocols over a single 802.16 connection. In the current 802.16 specification, address-based classification rules define how data packets of different users are mapped to different CIDs, so that the differentiated QoS and/or security provisions can be provided. The address-based classification rules require the current 802.16 convergence

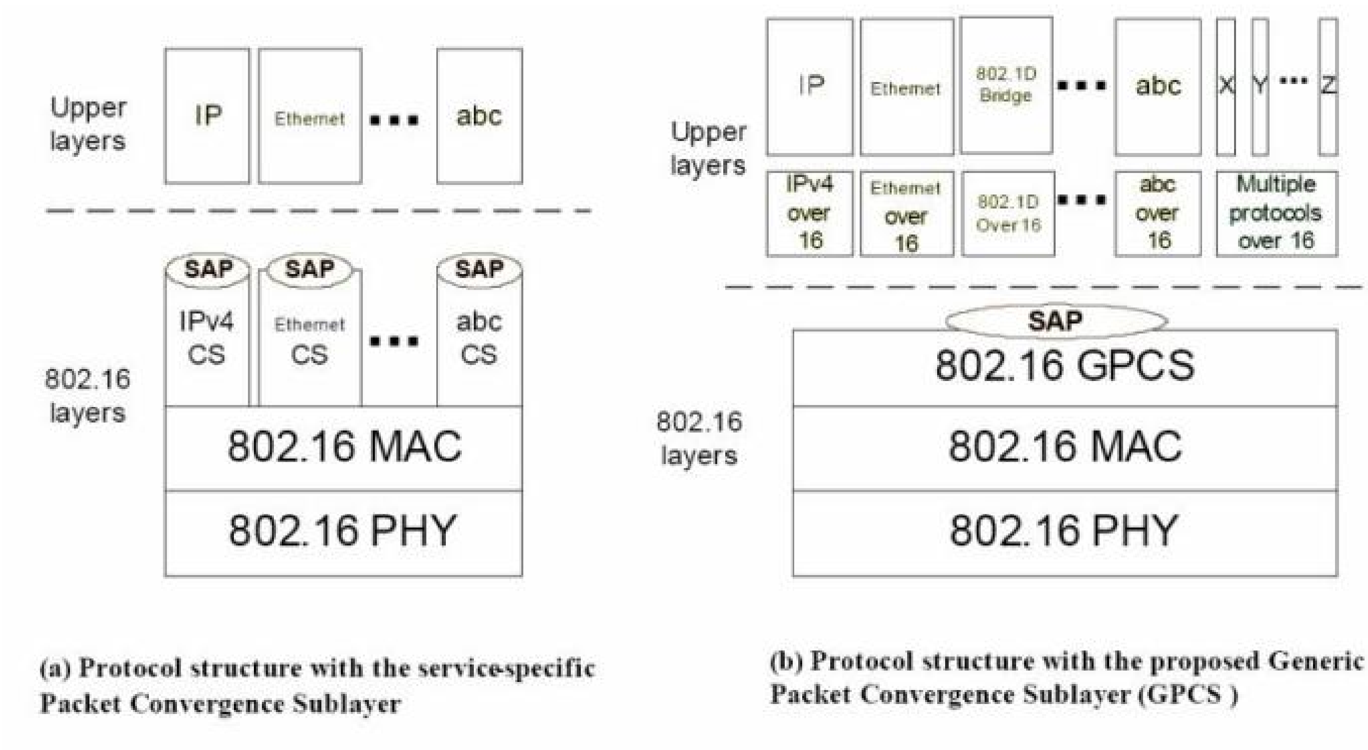

Figure 5.1: different CS and GPCS

sublayer to maintain the mapping information between upper-protocol-defined addresses (e.g., IP or Ethernet) and CIDs. This ultimately forces 802.16 CS to implement some upper layer functions. For example, the IPv4 CS at BS maintains a mapping state for IP addresses to CIDs. Whenever there is a change in IP addresses, the mapping state needs to be updated. In non-802.16 systems and protocol stacks, upper layer address assignment and mapping to link layer entities is typically part of a routing function at the network layer and not at MAC.

GPCS is equally important in case of WiFiRe as well. WiFiRe considers 802.3 based network for external connectivity. [6] describes intra-village connectivity and how it can be incorporated with WiFiRe. WiFiRe SS can be connected with 802.11 based access point. In that case, we may need to have CS which 'knows' 802.11. For voice connectivity, if we have RJ11 based network connecting to SS with TDM then we need this CS as well. 802.16e network has much wider range of clients. Beside normal SS, we can have laptop, mobile handset and other small devices which may use WiMAX. This all devices are having different usage and requirements (like mobility, security, QoS etc.). These devices may not want to implement CSs which are not required.

5.2 Performance Measurement in WiMAX

There is large possibility to have good performance evaluation based simulations in WiMAX. Here, we have listed some of them with their possible outputs.

WiMAX provide four different kinds of services: UGS, rtPS, nrtPS and BE. As we have described above in literature survey section, most of them recommend having different queue for different priority traffic. At the time of Tx, those queues will be appended one after another and then transmitted as a single PDU. This means that when ever we get a packet from Ethernet, it will be processed by CS and identified according to its service requirement. Then CPS will send them in different queue instead of FIFO. We argue that this process requires much higher processing time and may not provide significant QoS improvement. As we have seen so far, WiMAX frame is usually 10 ms. Considering 2:1 ratio for DL to UL, we get approximately 6 ms for DL. Now, putting UGS queue ahead of rtPS will give it advantage of 1-2 ms. Also note that, UGS is periodic request and doesn't get affected by re-arrangement of packets in priority sequence. To prove these results, one can setup small test bed and have fixed traffic of VoIP, FTP and HTTP.

WiMAX and WiFiRe currently support GPC based connections. WiMAX also prescribes GPSS mode for simpler design. There can be hybrid model which gives connection based on service flow. It means that each SS will have 3-4 different connection and there would be no more differences within that flow. It would be interesting to see that if we provide GPC based connection, what would be the table lookup time in CID database, classifier, header generation etc. It will be interesting to see that what the advantages of GPC mode against GPSF mode are.

There are few more interesting questions on traditional WiMAX settings like software-hardware coupling, complex scheduler, pending queues, VoIP header compression etc. If explored with realistic setting and environments, we may get surprising results.

5.3 Open Base Station architecture

An open base-station architecture[14] allows manufacturers to focus their research and development efforts on their core competencies. They also can buy selected base-station modules from each other and other specialist module vendors. These aspects will result in faster development of innovative, cost-effective base stations. They also will result in earlier and lower-cost introductions of new technologies and services to network operators. It supports different access technologies such as GSM/EDGE, CDMA2000, WCDMA and IEEE802.16/WiMAX. It has common operational and management interface. OBSAI is common forum joined by all major telecom companies. OBSAI allows multiple concurrent operation of air interface as well. It has different plane for control and user data. This problem is interesting because WiFiRe's one of the major cost is tower itself. If we can use existing tower (at least in urban area) of GSM or CDMA, spreading WiFiRe/WiMAX would be much easier. Here, we are not just sharing physical space but most of the common blocks are shared if we have more than one air interfaces. The OBSAI organization consists of more than 130 component, module and base station vendors. OBSAI specifications allow module vendors to manufacture modules that are capable of operating in any OBSAI-compatible BTS, thereby reducing substantially the development effort and costs involved in the introduction of a new range of BTS products. One more similarity with WiFiRe is that, OBSAI recommends RF components are to be kept remote from other modules. This means that RF module will take care for all Tx and Rx for GSM, WiMAX etc. and remaining software system will take care for operational and management functions.

Chapter 6

Conclusion

WiFiRe design is very much similar to other WiMAX deployment which we came across. Modularity given to MAC was great benefit as it helped development of WiFiRe MAC in absence of 802.11 PHY devices. WiFiRe's basic working prototype is the most important task ahead. We would like to use experience gained working for rural WiMAX in Integration of 802.11b chipset with WiFiRe MAC. We also feel that if we ignore client's ULP, we can make our system much simpler and efficient. For this purpose, additional modules can be developed to support WiFiRe MAC but they should not be written as part of MAC itself. Integration with 802.11b chipset requires frame construction in MAC with predefined structure which can be easily parsed. Integration with chipset will provide interesting wireless specific problems which we may not have expected.

Bibliography

[1] IEEE Std 802.16-2004. IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Fixed BroadbandWireless Access Systems. 2004.

[2] Ashok Jhunjhunwala Bhaskar Ramamurthi, Anand Kannan. Interim 3.5G broadband wireless system for India: Framework, requirement, performance needs. *CEWiT*, 2005.

[3] Sarat Chandra. An Efficient Call Admission Control for IEEE 802.16 Networks. *MTech Technical report, IIT Bombay*, 2006.

[4] Intel Corporation. Intel PRO/Wireless 5116 Broadband Interface. *http://www.intel.com/network/connectivity/products/wireless/307327.pdf*, 2007.

[5] IEEE 802.16 Broadband Wireless AccessWorking Group. GPCS for Supporting Multiple Protocols over 802.16 Air Interface. 2006.

[6] J Chandarana, Sravana K, S Perur, R Rangarajan, S Sahasrabuddhe and S Iyer . VoIP-based Intra-village Teleconnectivity: An Architecture and Case Study. *WiSARD, Comsware*, 2007.

[7] J Chen, CC Wang, F Chee-Da Tsai, CW Chang and SS Liu. The Design and Implementation of WiMAX Module for ns-2 Simulator. *ACM International Conference Proceeding Series; Vol. 202*, 2006.

[8] Sameer Kurkure. Design and Implementation of WiFiRE MAC Layer Protocol. *MTech Technical report, IIT Bombay*, 2007.

[9] Lakshminarayan Subramanian, Sonesh Surana and Eric Brewer . Rethinking Wireless for the Developing World. *Hot Topics in Networks*, 2006.

[10] Abhishek Maheshwari. Implementation and Evaluation of a MAC Scheduling Architecture for IEEE 802.16 WirelessMANs. *MTech Technical report, IIT Kanpur*, 2006.

[11] Supriya Maheshwari. An Efficient QoS Scheduling Architecture for IEEE 802.16Wireless MANs. *MTech Technical report, IIT Bombay*, 2005.

[12] D Sanghi P Bhagwat, B Raman. Turning 802.11 Inside-Outs. *ACM SIGCOMM Computer Communication Review*, 2004.

[13] Sridhar Iyer, Krishna Paul, Anurag Kumar, Bhaskar Ramamurthy. WiFiRe: Medium Access Control (MAC) and Physical Layer (PHY) Specifications. *CEWiT, India*, 2006.

[14] OBSAI technical working group. Open base-station architecture initiatives: BTS system refrence, Version 2.0 . 2006.