

## 摘要

宽带无线接入是未来通信网络极其重要的接入方式，尤其是基于 IEEE 802.16 协议标准的无线城域网（WMAN），由于其带宽高、覆盖范围广、提供 QoS 保证，已成为最重要的宽带无线接入技术之一，并是未来可重配置终端必不可少的配置技术。本论文的目的就是研究如何在 DSP 平台上设计和开发 802.16 标准的 MAC 层协议。

论文总结笔者参与江苏省高科技项目“下一代泛在网络端到端重配置关键技术”的研究心得，深入研究和探讨在 ARM、DSP 和 FPGA 的协作环境下，IEEE 802.16 MAC 层协议软件的设计和实现技术。

论文首先介绍了无线城域网的发展，并对其进行技术分析，引出了 IEEE802.16 标准及 WiMAX 技术。然后系统地分析了 IEEE802.16 标准的媒质访问控制（MAC）层总体功能，并对其中的关键技术作了重点阐述。接着给出了采用 TMS320C5410 DSP 实现的 IEEE802.16 MAC 层协议栈总体设计方案，设计了 MAC 层与其上下层的接口模块。最后研究 MAC 层协议的 DSP 实现技术，给出了 MAC 层协议的模块结构设计、各功能模块详细的 SDL 图以及模块的 C 语言软件设计。

论文提出的 IEEE 802.16 MAC 层协议栈 DSP 实现方案有良好的可扩展性，为项目后续实现终端 WiFi 和 WiMAX 协议重配置奠定了基础。

关键词：无线城域网 IEEE 802.16 媒体接入控制 数字信号处理

## ABSTRACT

Broadband wireless access is an extremely important future communication network access technology, in which IEEE 802.16 based WMAN is of the most importance owing to its high bandwidth, wide coverage and QoS. It is also an indispensable reconfiguration technology for the future reconfigurable terminals. This *thesis* is aimed at the study of the design and development of 802.16 MAC protocol on DSP platform.

Based on the author's research experience in the Jiangsu Provincial high-tech project "Research on the key technologies of end-to-end reconfiguration for next generation ubiquitous communication networks", the *thesis* makes an in-depth study on software design and implementation technology of IEEE802.16 MAC protocol under an ARM, DSP and FPGA collaborative environment.

Following an introduction of the development of WMAN and its technical analysis, IEEE 802.16 standard and WiMAX technology is briefed. Systematic analysis is then given to 802.16 MAC layer functionality with detailed discussion of its key technologies. Furthermore an overall design scheme is proposed of IEEE 802.16 MAC layer protocol using TMS320C5410 DSP. The interface modules between MAC layer and its upper layer and lower layer are designed. Finally a study is conducted of the DSP implementation technology of MAC protocol, including modular strcture design of the protocol, detailed SDL design of each module and their software design in C language.

The proposed DSP implemention scheme of IEEE 802.16 MAC layer protocol is of good scalability and has layed solid foundation for the follow-up project to realize WiFi and WiMAX reconfiguration in the terminals.

**Keywords:** Wireless Metropolitan Area Network(WMAN), IEEE802.16,

Media Access Control(MAC), Digital Signal Processing(DSP)

## 南京邮电大学学位论文原创性声明

本人声明所呈交的学位论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得南京邮电大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

研究生签名: 王成法 日期: 2009.4.12

## 南京邮电大学学位论文使用授权声明

南京邮电大学、中国科学技术信息研究所、国家图书馆有权保留本人所送交学位论文的复印件和电子文档，可以采用影印、缩印或其它复制手段保存论文。本文电子文档的内容和纸质论文的内容相一致。除在保密期内的保密论文外，允许论文被查阅和借阅，可以公布（包括刊登）论文的全部或部分内容。论文的公布（包括刊登）授权南京邮电大学研究生部办理。

研究生签名: 王成法 导师签名: 席之海 日期: 2009.4.12

# 第一章 引言

## 1.1 无线城域网概述

城域网是连接广域网和局域网的桥梁，它的特点表现在业务类型多样化以及业务流量流向的不确定。随着网络技术的不断发展，不断地向宽带IP化演进，网络上的业务也由早期的以话音业务为主逐步转向以数据业务、多媒体业务为主流。目前，网络建设也将朝着无线连接、无缝覆盖的方向努力。现在以蜂窝移动通信为代表的无线广域网和以802.11系列为代表的无线局域网已经在市场上取得了极大的成功，这也是对连接广域网和局域网的城域网的巨大挑战，因此宽带无线城域网的发展是很关键的问题。

无线城域网是指城域网向无线化的方向发展，它主要解决最后一公里的接入，也是实现无缝覆盖的网络的必然需求。

## 1.2 IEEE 802.16 标准产生的背景及发展进程

### 1.2.1 IEEE802.16 标准产生的背景

本地多点分配业务(LMDS)是一种固定宽带无线接入技术，它利用微波传输提供点到多点的通信，能够进行双向数据传输，并且带宽容量大。

LMDS可以支持包括话音业务、数据业务、图像业务以及多媒体业务在内的多种业务类型，像会议电视、视频点播、远程教育等，业务种类十分丰富，而且可以达到很高的用户接入速率，提供高质量的服务。除此以外，LMDS建设起来比较容易，成本也比较低，十分灵活而且见效很快，很有发展前景。

随着LMDS宽带无线接入系统的快速发展，有很多公司也已经推出了LMDS的产品，但是国际上对LMDS没有统一的标准，这样很容易对它的继续发展形成一定的障碍，因此标准化的工作显得非常重要。有很多标准化的组织开展了标准化的工作，发展较快的就是IEEE。IEEE 802.16工作组就是专门致力于宽带无线接入技术的标准化工作。

### 1.2.2 IEEE802.16 标准的发展进程

IEEE 802.16是IEEE制定的各种标准中定位于无线城域网的接入系统标准。IEEE 802.16的主要任务是，开发工作于2-66GHz频带的无线接入系统空中接口物理(PHY)层和媒体接入控制(MAC)层规范，同时还包括与空中接口协议相关的一致性测试以及不同无线

接入系统之间的共存规范。一般所说的 IEEE802.16 标准通常是指空中接口标准。

根据是否支持移动性，可以将 IEEE 802.16 系列标准分为两类，一类是固定宽带无线接入空中接口标准，主要包括 802.16-2001, 802.16a, 802.16d；另一类是移动宽带无线接入空中接口标准，主要是指 802.16e。此外，IEEE 802.16 系列标准还有 IEEE 802.16c, IEEE 802.16f 和 IEEE 802.16g 三个标准。

IEEE802.16 工作组成立于 1999 年，该工作组专门致力于固定宽带无线接入技术的标准化工作。

IEEE 802.16-2001 标准[1]由 IEEE802.16 工作组于 2002 年 4 月正式发布，它是 IEEE 802.16 协议的第一个正式版本，定义了无线城域网的空中接口规范，属于固定宽带无线接入空中接口标准。IEEE 802.16-2001 工作的物理频段是 10-66GHz，在这个频段范围内用户站和基站之间是视距传输。

IEEE 802.16c 标准[2]由 IEEE 802.16 工作组于 2002 年 12 月发布，它对 IEEE802.16-2001 标准进行了增补和改正。IEEE802.16c 的工作频段适用范围是 10-66GHz。

IEEE 802.16a 标准 [3] 由 IEEE802.16 工作组于 2003 年 1 月发布，它是对 IEEE802.16-2001 标准的修改和扩展，仍然属于固定宽带无线接入空中接口标准。IEEE802.16a 标准支持的工作频段为 2-11G Hz，在这个频段，系统可以运行于非视距传输环境。该频段与高频段相比有很多优势，在很大程度上降低了安装用户站的要求，并且节约了成本，能提供更广的覆盖范围。

IEEE 802.16d 标准[4]由 IEEE802.16 工作组于 2004 年 6 月发布，它属于固定宽带无线接入空中接口标准，在 IEEE 802.16-2001 标准和 IEEE 802.16a 标准的基础上进行了整合和修正，是固定宽带无线接入规范中最成熟的标准。IEEE802.16d 标准支持的频段是 2-66 GHz，详细制定了这个频段的物理层和 MAC 层规范。

随着技术和市场的发展，也对 802.16 系列标准提出了移动性的要求，IEEE802.16e 标准增加了对移动性的支持。

IEEE 802.16e 标准[5]已于 2005 年 12 月获得批准，它属于移动宽带无线接入空中接口标准。IEEE 802.16e 工作的物理频段是 2-6 GHz，在 IEEE 802.16d 的基础上提出了具有移动特性的系统结构。

此外，还有两个标准为 IEEE802.16f[6]和 IEEE802.16g[7]，前者用于定义 IEEE 802.16 系统 MAC 层和物理层的管理信息库(MIB)以及相关的管理流程，后者用于规定标准的 IEEE 802.16 系统管理流程和接口。

## 1.3 WiMAX 的技术特点及应用

### 1.3.1 WiMAX 与 IEEE 802.16 标准的关系

WiMAX[8][9][10]的全称为 Worldwide Interoperability for Microwave Access，可称为全球微波接入互操作系统。它由众多业界领先的通信公司及设备制造商共同组建。这个组织的目的就是对基于 IEEE 802.16 标准和 ETSI HiperMAN 标准的宽带无线接入产品进行一致性和互操作性认证。通过该组织认证的产品符合 802.16 标准的要求，具有一致性和互操作性，可用于宽带无线接入的应用。

由此可见，WiMAX 与 IEEE 802.16 之间有着非常紧密的关系，但同时它们在分工上又存在不同，具体来说就是，后者是标准的制定者，前者又是标准的推动者，前者将对产品是否符合后者的要求进行认证。目前，WiMAX 几乎已经成为 IEEE 802.16 标准的代名词。

### 1.3.2 WiMAX 的技术特点[11]

作为无线城域网的技术，WiMAX 有巨大的发展潜力及广泛的应用前景与其技术特点和优势密不可分。

WiMAX 的技术优势体现在很多方面。首先，它能提供高速率的宽带接入以及更远的覆盖距离。WiMAX 的基站可以提供最高每扇区 75Mbit/s 的接入速率，并且每个基站的覆盖范围可达 50 公里，较为典型的基站覆盖距离为 6-10 公里，这样只需建设少量的基站就可以实现全城范围的覆盖。

其次，在网络结构方面，可以支持点到多点(PMP) 结构和 Mesh 结构，前者在 IEEE 802.16-2001 标准中就提出了，后者是在 IEEE 802.16a 中提出的。这样在骨干网络覆盖不到的区域，可以采用 WiMAX 网状网覆盖，以灵活地扩展接入网的结构。

在物理层，可以根据实际情况灵活选择多种调制方式，包括单载波、OFDM 和 OFDMA，OFDM 技术能有效地降低多径衰落的影响。并且，随着传输信道状态发生变化，为了保证传输质量，载波调制方式以及一些物理层的参数都可以进行动态调整，十分灵活。

MAC 层的一个重要特点在于它是面向连接的，可以提供服务质量保证。不同的业务类型具有不同的服务质量参数，可以通过灵活的机制保障各种业务的服务质量。

### 1.3.3 WiMAX 的应用[12][13]

WiMAX 因其在网络拓扑结构上的特点，并且在技术上有很多优势，因此具有广泛的应用范围。

首先，网络可以采用 PMP 的拓扑结构，提供点到多点的宽带无线接入。无线网络相比有线网络更加方便灵活，相对还可以节约成本，比如可以将无线局域网热点接入高速因特网，还可以为一些企事业单位、住宅区、密集的商业区、网吧、写字楼等提供宽带接入。而且对于需要临时建网的情况，无线网络的优势就更明显了。

WiMAX 还可以采用 Mesh 的拓扑结构，这种无线网状网的结构可以非常灵活地建构网络，而且 WiMAX 的覆盖范围很广，尤其适用于农村地区等一些有线网络设施不健全的区域，构建无线网络比较方便，建网速度也相对较快。

此外，WiMAX 还可以用于基站之间的无线通信，主要是利用它的长程通信能力，还可以为企业提供虚拟专用网服务。

在 2007 年，WiMAX 已被接受为 3G 系统的无线标准之一。

#### 1.3.4 本人所做的工作

本文研究源于江苏省工业高科技研究项目“下一代泛在通信网络中端到端重配置关键技术研究”。通过对 IEEE 802.16 MAC 协议的系统性研究，给出了基于 ARM(ARM9)、DSP(TMS320C5410)和 FPGA 的协作环境下 MAC 层协议的实现方案，设计了 MAC 层软件模块，并使用 C 语言编写实现了 MAC 层协议软件，完成了联合调试，验证了相关模块及算法的正确性及有效性。

第一章首先回顾了无线城域网的发展，介绍了 IEEE 802.16 标准产生的背景及发展进程，由此引出 WiMAX 技术，并对 WiMAX 与 IEEE 802.16 标准之间的关系做了简要描述。第二章概述 IEEE802.16 标准，简要介绍了 IEEE802.16 协议栈参考模型以及物理层与 MAC 层协议。第三章对 IEEE802.16 MAC 层协议的各功能子层做了详细阐述，系统分析了公共服务功能子层的各种功能，并对其中的关键技术做了重点阐述。这些分析为第五章的程序设计打下了基础。第四章给出了基于 ARM(ARM9)、DSP(TMS320C5410)和 FPGA 硬件平台的 MAC 层协议实现方案，描述了终端的系统结构及工作流程，设计并实现了 MAC 层与其上下层的接口模块，并对接口模块进行了调试和分析。第五章给出了基于 DSP 的 IEEE 802.16MAC 协议设计和实现方案，首先介绍了 MAC 层协议软件架构，设计了 MAC 层协议软件调度器，然后对 MAC 层各软件模块进行了介绍及实现，最后对整个 MAC 层协议栈的实现进行了调试和测试。最后总结了本论文的工作情况，说明了本文存在的不足和后期需要进一步改进的地方。

## 第二章 IEEE802.16 协议概述

### 2.1 IEEE 802.16 协议栈参考模型

本文主要参考的是 IEEE802.16-2004 (即 802.16d) 标准。标准中给出了 IEEE802.16 协议栈的参考模型，如图 2-1 所示。

由协议模型图可以看出，IEEE 802.16 的空中接口由物理层(PHY)和媒体接入控制(MAC)层组成。MAC 层又分成了三个子层，分别是特定业务汇聚子层(Service-specific CS, Service-specific Convergence Sublayer)，MAC 公共部分子层(MAC CPS, MAC Common Part Sublayer)和安全子层(SS, Security Sublayer)。

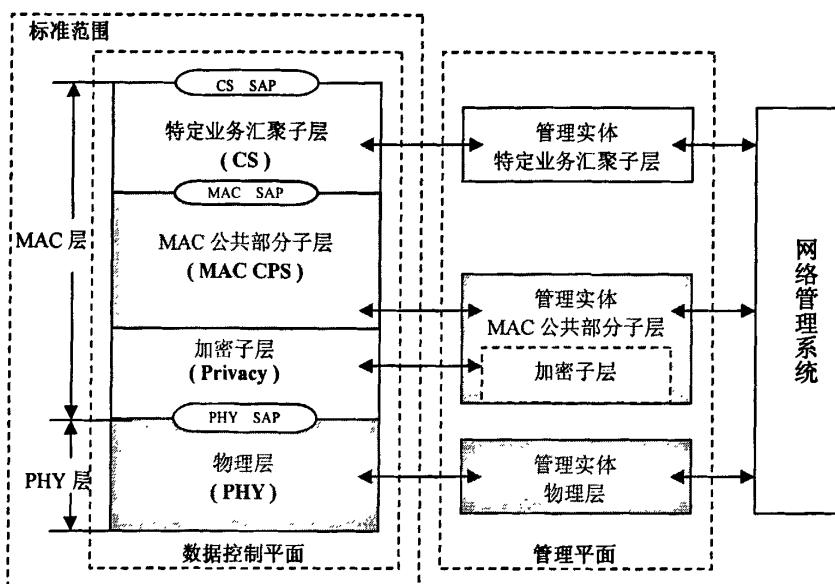


图 2-1 IEEE 802.16 协议栈参考模型

### 2.2 IEEE 802.16 物理层

物理层定义了多种规范，每一种都为了适用于特殊的频率范围和应用。

物理层使用 10-66GHz 和 2-11GHz 的频率范围，并可以支持时分双工(TDD)方式和频分双工(FDD)方式。时分双工是指上行传输和下行传输占用不同的时间段，频分双工是指上行传输和下行传输占用不同的频率。

在 10-66 GHz 这个频段，需要满足视距传输的需求，可以不用考虑多径效应。针对此频段 802.16 协议定义了一种物理层规范即无线城域网—单载波(WirelessMAN-SC)模式，即

采用单载波调制方式，可以采用 QPSK, 16QAM, 64QAM 调制技术。在 WirelessMAN-SC 模式下，下行信道采用时分复用(TDM)方式，上行信道采用时分多址接入(TDMA)与按需分配多址接入(DAMA)相结合的多址方式。

在 2-11G Hz 这个频段，满足非视距传播要求，针对此频段 802.16 协议定义了三种物理层规范，分别为无线城域网—增强单载波(WirelessMAN-SCa)模式、无线城域网—正交频分复用(WirelessMAN-OFDM)模式以及无线城域网—正交频分多址(WirelessMAN-OFDMA)模式。其中，WirelessMAN-SCa 模式也指单载波调制方式，WirelessMAN-OFDM 模式是采用 256 点变换 OFDM 技术，WirelessMAN-OFDMA 模式是采用 2048 点变换 OFDM 技术。在 WirelessMAN-SCa 和 WirelessMAN-OFDM 模式下，和 WirelessMAN-SC 模式类似，一般在下行信道采用时分复用(TDM)方式，上行信道采用时分多址接入(TDMA)与按需分配多址接入(DAMA)相结合的多址方式，而在 WirelessMAN-OFDMA 模式中，上行信道和下行信道都采用了正交频分多址(OFDMA)方式。

## 2.3 IEEE 802.16 MAC 层

前面已经提到，MAC 层包含三个子层，分别为特定业务汇聚子层、MAC 公共部分子层和安全子层。

特定业务汇聚子层位于 MAC 公共部分子层之上，接收高层来的协议数据单元(PDU)，对其分类并映射到特定的 MAC 连接，并通过 MAC SAP 将 MAC SDU 交给 MAC 公共部分子层。不同类型的汇聚子层可以为高层提供不同的接口。

MAC 公共部分子层主要提供 MAC 层的核心功能，包括系统接入、带宽分配、连接建立以及连接维护，它通过 MAC SAP 从各种类型的汇聚子层接收数据还可以提供 QoS 机制。

MAC 层还包含一个安全子层，用来提供鉴权、密钥交换和加密功能。

在后面将对 MAC 层协议进行详细的介绍。

## 第三章 IEEE 802.16 MAC 层

IEEE802.16 协议的 MAC 层由 CS (特定业务汇聚子层), MAC CPS (MAC 层公共部分子层) 和 Privacy (安全加密子层) 组成。3.1 节分别描述了 CS 子层为支持两种 (ATM CS 和 Packet CS) 汇聚业务而具备的功能。3.2 节研究了 MAC CPS 子层的协议封装、业务调度、支持的物理层特性。3.3 节简要介绍了作为可选项的加密子层。

### 3.1 CS 子层

CS 子层位于 MAC CPS 子层之上, 经由 MAC SAP(MAC 业务接入点), 接收 MAC CPS 的服务。它执行如下功能:

- 接收来自高层的 PDU。
- 对高层 PDU 分类并映射到以 16bits 的 CID 为标识的相应连接上。
- 发送 CS PDU 到和业务流相关的 MAC SAP, 以便传递到对等层 MAC SAP。
- 接收来自对等层 MAC SAP 的 CS PDU。

当前本协议仅提供两种 CS 规范: ATM CS 和 Packet CS。其它 CS 层规范可在未来加入。无论哪种 CS 都要执行“分类”(classification) 和可选的“净荷头抑制”(PHS, Payload Header Suppression) 功能。

#### 3.1.1 分类功能

分类: 是指外部网络 PDU 进入 CS 子层后(称为 CS SDU), 被映射到某个特定的连接, 在整个 MAC 对等层间的传递过程中使用, 在映射过程中将创建该连接的业务流特征和 QoS 限制。

分类器是一组匹配准则, 它逐个处理进入 IEEE802.16 协议层的每一个外部网络 PDU (CS SDU)。它由针对特定外部协议的某种匹配准则 (如 ATM 分类器以 VPI/VCI 为对象, Packet CS 分类器以目的地 IP 地址为对象)、某种分类优先级和一个对 CID 的引用法则组成。符合特定匹配准则的 CS SDU 将被传递到相应的 MAP SAP 上, 并使用由 CID 定义的“连接”在 MAC CPS 层中进一步传递。该连接的业务流特性将为该 SDU 提供 QoS。

多个分类器可以将 CS PDU 分类到同一个业务流上。分类器优先级用于指明优先使用哪个分类器对 CS PDU 分类。由于分类器使用的模式可能部分重叠, 因此有必要明确指定顺序。BS 和 SS 侧分别应用下行链路、上行链路分类器对收到的 CS SDU 分类。如果 CS

接收到无法识别和分类的 CS SDU 应当为其分配一个默认的 CID 或者简单丢弃。CS 子层的分类功能如图 3-1 和图 3-2 所示。

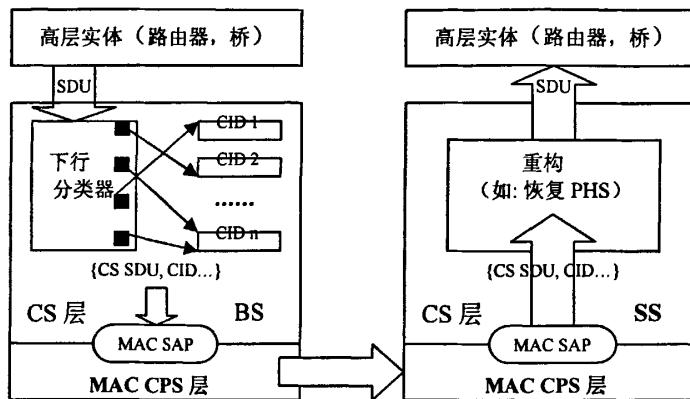


图 3-1 CS 层的分类和 CID 映射 (BS 到 SS)

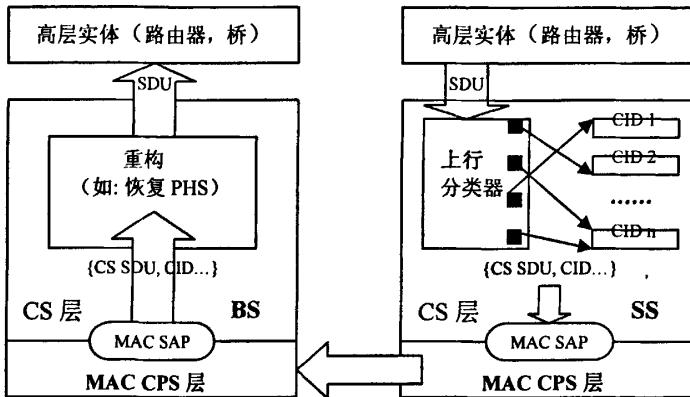


图 3-2 CS 层的分类和 CID 映射 (SS 到 BS)

### 3.1.2 净荷头抑制功能

**净荷头抑制:** 是指将进入 CS 层的 CS SDU 头部的部分冗余信息在发送端抑制，而在接收端对等实体中恢复的过程。抑制的原因是新形成的 CS PDU 中头部信息保留了原 SDU 头部必要信息，而部分冗余信息，如 ATM CS SDU 的 VPI/VCI 被映射成 CID，校验信息 HEC 被丢弃，在 MAC CPS 层中将使用新协议自身的校验位 HCS。这样通过抑制冗余信息来节省带宽和协议开销。PHS 一般是在对 CS SDU 进行分类后执行，并且是可选的。PHS 准则根据高层业务流（由 SFID 标识）的特点将来自高层业务分组的头部部分冗余字节抑制，并加以 PHSI 前缀。因此 PHS 准则由一对{SFID, PHSI}标识，在接收端 CS 层再应用 PHS 准则重构原 PDU 的头部。

### 3.1.3 ATM CS 子层

#### 3.1.3.1 ATM CS 业务定义

ATM CS 子层定义了对来自外部网络 ATM 层的 PDU 进行汇聚的协议规范。作为联系各种 ATM 业务和 MAC SAP 的逻辑接口，它对来自 ATM 层的 ATM 信元执行分类、净荷头抑制（可选的）和转发到适当 MAC SAP 等功能。由于 ATM 信元流是根据 ATM 标准产生的，因此在 ATM CS 子层中不需要 ATM CS 业务原语。ATM CS PDU 由 Header 和 Payload 组成。Payload 应当等于 ATM 信元的 payload 大小，Header 则应由来自外部 ATM 网络的 ATM 信元头按一定规则映射形成，最大不超过 5byte(不采用 PHS 时)。如图 3-3 所示。

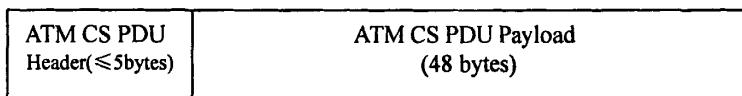


图 3-3 ATM CS PDU 格式

#### 3.1.3.2 ATM CS 的分类和净荷头抑制功能

外部网络 ATM 信元在 ATM CS 层中（也称为 ATM CS SDU）被分类、封装成 ATM CS PDU。一个 ATM 连接存在 VP 和 VC 交换 2 种模式，因此 ATM CS 子层使用 2 种匹配规则的分类器来区分收到的 ATM 信元。ATM CS 的分类器由某种 ATM 信元匹配准则（以 VPI,VCI 为对象）、分组优先级和一个对 CID 的引用法则组成。分类后的 ATM 信元将被映射到以 CID 为标识的 MAC 连接上。对于 VP 交换模式，ATM 信元的 VPI 域（NNI:12bit 或 UNI:8bit）被映射成 16bit 的 CID；对于 VC 交换模式，则 VPI 和 VCI 一起（NNI:28bit 或 UNI:24bit）被映射成 16bit 的 CID。由于在连接建立时将设置相应的 QoS 和业务参数类别，所以将 VPI/VCI 映射成 CID 就可保证 MAC CPS 子层能正确处理各种 ATM 业务。但 VC 模式下不能支持全部的 VPI/VCI 组合（16bit 的 CID 可寻址  $2^{16}$  个连接，VC 模式下在 NNI 侧和 UNI 侧的 VPI/VCI 分别可寻址  $2^{28}$  或  $2^{24}$  个 ATM 连接）。

表 3-1 给出了分类后采用 PHS 形成新的 ATM CS PDU 头部的 C 语言函数。可见，对于 VP 交换模式的 ATM 信元，其 PHS 法则中 VPI 被当成冗余成分抑制，因为 VPI 将被映射成 16bit 的 CID 用来寻址一个 MAC 连接，并随后在该连接上传递 MAC SDU 到 MAC CPS 层，故应在 ATM CS PDU Header 中被抑制(suppression)。对于 VC 交换模式的 ATM 信元，则 VPI 和 VCI 一起被映射成 16bit 的 CID 寻址，故在 ATM CS PDU Header 中均被抑制。

表 3-1 ATM CS PDU Header 生成函数

语法 (C 语言)	大小	注释
ATM_CS_PDU_Header()		ATM CS PDU Header 映射函数
If (no_PHS) {		如果不做净荷头抑制 (PHS),
ATM Header	40 bits	则直接继承全部 40bit 的 ATM 头。
}		
else if (VP_switched) {		否则, 若要求 PHS 且该 ATM 信元是属于 VP 交换的,
PTI	3bits	抽取 ATM 信元的 PTI 域
CLP	1bit	抽取 ATM 信元的 CLP 域
reserved	4bits	预留 4bits
VCI	16bits	抽取 ATM 信元的 VCI 域 (VPI 为冗余 bit)
}		
else (VC_Switched) {		否则, 若要求 PHS 且该 ATM 信元是属于 VC 交换的,
PTI	3bits	抽取 ATM 信元的 PTI 域
CLP	1bit	抽取 ATM 信元的 CLP 域
reserved	4bits	预留 4bits ( VPI 和 VCI 均为冗余 bit )
}		ATM 信元的 HEC 域在 2 种模式下都被抑制。
}		

另外 ATM 信元的 HEC 域在 2 种模式中都要抑制, 因为在 MAC CPS 层中形成的新 MAC PDU 有自己的头部校验域 (HCS, Header CheckSum)。图 3-4 和 3-5 给出了基于 VP 和 VC 交换的两种 ATM 信元被映射成的 2 种 ATM CS PDU 的具体格式。

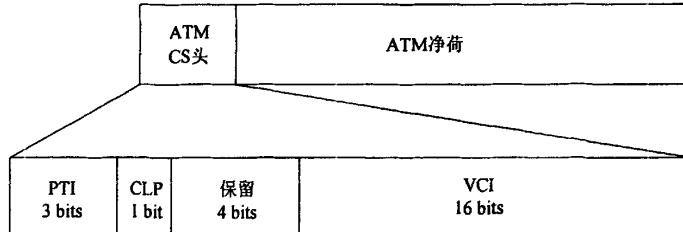


图 3-4 VP 交换模式的 ATM 连接下 ATM CS PDU 格式

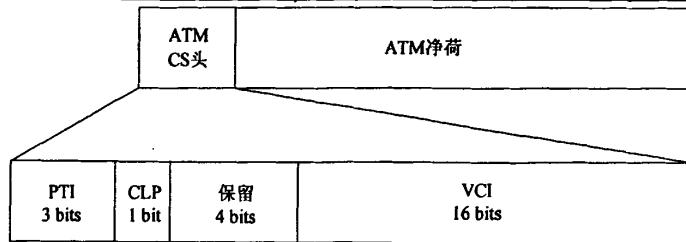


图 3-5 VC 交换模式的 ATM 连接下 ATM CS PDU 格式

### 3.1.4 Packet CS 子层

#### 3.1.4.1 Packet CS 的业务定义

Packet CS 支持的高层外部协议是诸如 IP、PPP、IEEE802.3(以太网)的基于分组的协议。它也执行和 ATM CS 类似的分类和 PHS 等功能。高层外部协议 PDU 应被封装在 Packet CS PDU 中。对于一些高层外部协议的 Packet PDU，其前面可能会加上一个 8bit 的净荷头抑制指示域 (PHSI, Payload Header Suppression Indication) 再形成 Packet CS PDU，另一些则直接将高层外部协议 PDU 映射成 Packet CS PDU。PHSI=0 表示对该高层 PDU 没有执行 PHS 操作，否则指示了某种抑制规则。Packet CS PDU 格式如图 3-6 所示。

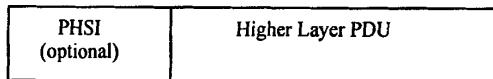


图 3-6 Packet CS PDU 格式

#### 3.1.4.2 Packet CS 的分类和净荷头抑制功能

Packet CS 的分类器由多个分类器组成。每个分类器包含一个优先级域来决定优先使用次序。如果分组和某个分类器的所有参数匹配则转发至相应的连接上，如果找不到匹配的分类器则选择默认的连接或丢弃分组。当 Packet CS 收到来自高层分组流时，将该分组和一组匹配器进行比较。一旦匹配则产生一个 SFID (标识该分组所属业务流)、一个 CID (标识分配的连接) 和 PHS 准则。

PHS 准则提供一组数据字节 (PHSF, PHSI, PHSM, PHSS 和 PHSV)。当 PHSV=1 时，发送方 CS 层把高层 PDU 头部字节和 PHSF 中将被抑制的字节 (由 PHSM 掩码指示) 做比较，若匹配则抑制 PHSF 中除 PHSM 屏蔽外的所有字节。最后，在高层 PDU 上加上 PHSI 前缀构成 CS PDU 并递交给 MAC SAP。收端 MAC CPS 去掉 MAC 头得到 CS PDU 和相应 CID，一起递交给 CS 层。接收方 CS 使用 CID 和 CS PDU 的 PHSI 来恢复 PHSF，并查询 PHSM 和 PHSS 字节，以重构原高层 PDU 的头部。完成后传递到 CS SAP。Packet CS 的具体 PHS 操作如图 3-7 所示。

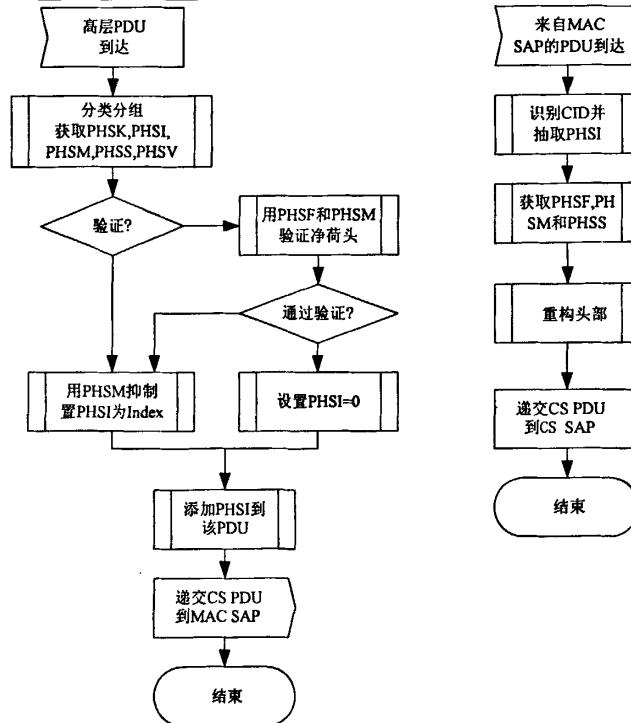


图 3-7 Packet CS 层中净荷头压缩 (PHS) 流程图

### 3.2 MAC CPS 子层

#### 3.2.1 MAC CPS 概述

MAC CPS (MAC Common Part Sublayer)即 MAC 公共部分子层。它位于 CS 子层下面，向 CS 提供服务，提供系统接入、带宽分配、连接建立和维护等核心 MAC 功能。

MAC 是面向连接的，MAC 层把来自高层的 PDU 映射到 SS 的不同业务流并与各种 QoS 等级关联，而所有业务流都是以 CID 标识的“连接”为背景。即 SS 在注册后立即将各种业务流和特定的连接关联（通常为一一映射），从而为“是谁请求的带宽”提供一个参考。并且，连接是随着业务流的变化而动态建立和拆除的。无论高层是面向连接的话音业务还是无连接的 IP 业务，在 MAC 层都将按业务流映射成动态的连接。连接一旦建立可能需要活动维护，具体依赖于连接携带的业务类型。例如，非信道化 T1 业务因每帧固定分配带宽而不需连接维护；信道化 T1 业务则由于压缩产生的动态带宽需求而需要某种维护；IP 业务则由于其突发性和较高的分段可能性，更需要一定量的维护。连接通常在一个业务合同改变时终结，转而动态建立符合新业务流 QoS 需求的连接。为实现动态业务管理，MAC 层通过使用连接的静态配置和连接的动态添加(DSA)、修改(DSC)、删除(DSD)等操作来实现，并且发起方可以是 BS 或 SS。

MAC 层定义了以下术语：连接(connection)，连接标识符(CID, Connection ID)，业务流(SF, Service Flow)，业务流识别符(SFID, Service Flow ID)。其中需要特别说明的是，连接标识符不同于设备的 MAC 物理地址，它是一个单向的 MAC 层地址，用来标识 BS 和 SS 的 MAC 层中对等实体间的连接。它映射到一个业务流的 SFID，而 SFID 又定义了该业务流的 QoS 参数组。因此，一个连接上的业务流实际上提供了上下行链路的 QoS 管理机制，并和带宽分配进程是紧密联系的。某个 SS 可以每连接为基础（隐含以业务流为标识）来请求上行带宽。带宽授予则由 BS 在某个调度时间间隔内集中或以每连接为基础授予该 SS。

### 3.2.2 MAC 业务原语

在分层协议中，层间信息用“原语”来定义，它描述各项不同的信息并触发行为的发生。由于存在几个子层，MAC 业务原语可分为 2 种：

- 向 CS 层提供 MAC 服务的原语，业务接入点是 MAC SAP。
- 向高层提供业务的原语，业务接入点是 CS SAP。

MAC SAP 业务接入点支持的业务原语分为连接类和数据类，连接类包含创建、改变和终结 3 种行为，每种行为有 request, indication, response, confirmation 四种原语，数据类仅有 request, indication 原语。在实现时，原语可定义为一个指向某个复杂的数据结构（数组、结构或链表等）的指针。图 3-8 表示了使用这些原语提供 CS—MAC 层间通信的流程。

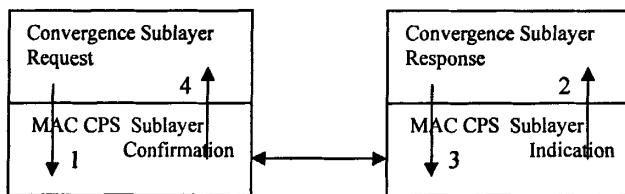


图 3-8 CS-MAC 层服务原语。

### 3.2.3 MAC 寻址和连接

#### 3.2.3.1 MAC 地址的作用

协议规定 SS 本身的唯一性由一个 48bit 的通用 MAC 地址标识。其作用是：

- 用于表示不同厂家和设备类型。
- 用于 SS 在初始测距过程中 (RNG-REQ, RNG-RSP) 为自己建立适当的连接。
- 用于在鉴权过程中 (PKM-REQ, PKM-RSP) BS 和 SS 相互认证。

#### 3.2.3.2 MAC 地址和 CID 的关系：

MAC 连接的唯一性有 16bit 的 CID 来标识。SS 在初始化时的 Initial Ranging 过程通

过 RNG-REQ 消息携带 MAC 地址 (SS MAC Address) 发给 BS, BS 根据不同的 MAC 地址给 SS 分配不同管理连接的 CID, 建立 SS MAC Address (48 bit) 与 CID (16bit) 的映射表(一个 SS 的 MAC 地址可以有多个属于自己的 CID)。然后通过 RNG-RSP 将分配的 CIDs 返回给 SS, SS 也建立一个映射表。以后 BS 和 SS 对等层间寻址则由建立的 MAC 连接的 CID 标识来寻址。

### 3.2.3.3 基本管理连接的建立

在初始化过程中(包含初始测距和注册过程), SS 必须先分配 3 类不同的管理连接(上行 3 类, 下行 3 类, 即至少 6 个不同的 CID), 这些连接的 CID 通过 BS 发出的 RNG-RSP 和 REG-RSP 响应消息来分配, 分别反映了 BS 和 SS 间 MAC 管理业务的 3 种不同 QoS。RNG-RSP 消息分配的是 Basic CID 和 Primary Management CID, REG-RSP 消息分配的是 Secondary Management CID, 其中

- Basic CID 用于传送短时、紧急的 MAC 管理消息。
- Primary Management CID 用于传送较长时延容忍的 MAC 管理消息。
- Secondary Management CID 用于传送时延容忍且基于 DHCP,TFTP 协议的管理消息。

除了以上三种管理连接 CID 外, 还有广播 CID、多播 CID 以及传递 CS 层 PDU(MSDU, MAC SDU) 的 Transport CID。划分范围如表 3-2 所示。

表 3-2 连接标识符 CID 划分范围

CID	Value	Description
Initial Ranging	0x0000	SS 在初始测距过程中使用。
Basic CID	0x0001~m	
Primary Management CIDs	m+1~2m	
Transport CIDs and Secondary Management CIDs	2m+1~0xFEFF	Transport CID 用于传递 User PDU, Secondary Management CID 用于传递 DHCP,TFTP 管理消息
Multicast Polling CIDs	0xFF00~0xFFFF	SS 可以包含在一个或多个多播组中, 以便通过轮询获取带宽, 这类连接没有关联的业务流。
Broadcast CID	0xFFFF	用作下行向所有 SS 的广播信息。

### 3.2.4 MAC PDU 格式与传送

#### 3.2.4.1 MAC PDU 格式

MAC PDU 是在 BS 和 SS 的 MAC 层之间传送的数据单元。MAC PDU 由固定长度的 MAC 报头、可变长度的净荷部分和可选的循环冗余校验部分构成。MAC 报头包括两种:通用报头(如图 3-9 所示)和带宽请求报头(如图 3-10 所示)。包含通用报头的 MAC PDU

净荷部分，或者携带 MAC 层管理消息，或者携带汇聚子层传递过来的数据。而带宽请求 MAC PDU 没有净荷部分，只包含带宽请求报头。表 3-3 和表 3-4 分别为通用报头和带宽请求报头中各域的说明。

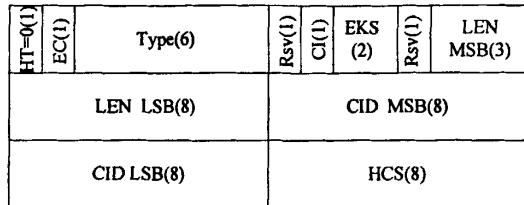


图 3-9 通用 MAC 报头

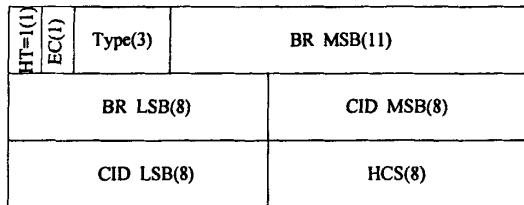


图 3-10 带宽请求报头

表 3-3 通用 MAC 报头中各域说明

域名	长度/比特	描述
CI	1	CRC 校验指示 1=该 PDU 中包含 CRC 校验信息 0=该 PDU 中不包含 CRC 校验信息
EC	1	加密控制 0=净荷没有被加密 1=净荷被加密
EKS	2	加密密钥序列
HCS	8	报头校验序列
HT	1	报头种类, 设为 0
LEN	11	该 MAC PDU 的字节长度, 包括报头和可能的 CRC 部分
Type	6	指示净荷部分所包含的子报头或特殊净荷类型

表 3-4 带宽请求报头中各域说明

域名	长度/比特	描述
BR	19	SS 在该 CID 上所请求的上行带宽的字节数
CID	16	连接标识
BC	1	总设置为 0
HCS	8	报头校验序列
HT	1	报头种类，总设置为 1
Type	3	指示带宽请求报头的种类

### 3.2.4.2 MAC 层管理消息

IEEE802.16 协议共定义了 50 种 MAC 层管理消息，主要的 MAC 层管理消息如表 3-5 所示。所有的 MAC 层管理消息都是以管理消息类型域开始，并可能包含其它的域。MAC 管理消息的格式如图 3-11 所示，MAC 管理消息只会在管理连接上进行传送。



图 3-11 MAC 管理消息格式

表 3-5 MAC 管理消息定义

类型	消息名称	消息描述	传送连接类型
0	UCD	上行信道描述	广播
1	DCD	下行信道描述	广播
2	DL-MAP	下行接入描述	广播
3	UL-MAP	上行接入描述	广播
4	RNG-REQ	测距请求	初始测距或基本
5	RNG-RSP	测距响应	初始测距或基本
6	REG-REQ	注册请求	主管理
7	REG-RSP	注册响应	主管理
11	DSA-REQ	动态服务流增加请求	主管理
12	DSA-RSP	动态服务流增加回复	主管理
13	DSA-ACK	动态服务流增加确认	主管理
14	DSC-REQ	动态服务流修改请求	主管理
15	DSC-RSP	动态服务流修改回复	主管理
16	DSC-ACK	动态服务流修改确认	主管理
17	DSD-REQ	动态服务流删除请求	主管理
18	DSD-RSP	动态服务流删除回复	主管理
26	SBC-REQ	基本能力协商请求	基本
27	SBC-RSP	基本能力协商回复	基本
30	DSX-RVD	等待服务流请求处理	主管理

### 3.2.4.3 MAC PDU 构造及传送

MAC 层支持的高层协议包括 ATM 协议和 IP 协议。来自汇聚子层的 MAC SDU 可能被分段或者打包，构建成为 MAC PDU。分段是将一个 MAC SDU 分成一个或多个 MAC PDU。分段使得系统能够根据连接服务流的 QoS 要求来有效地使用带宽资源。打包则是将多个 MAC SDU 封装成一个 MAC PDU。携带数据的 MAC PDU 在进行加密之后，还可能需要进行循环冗余校验计算，计算所得的 CRC 会添加到 MAC PDU 的尾部。CRC 所提供的保护包括 MAC 报头和 MAC PDU 的净荷部分。

多个 MAC PDU 可能经过相同的调制和编码方式连接成一个突发脉冲，如图 3-12 所示。MAC PDU 在物理层成帧之后，便送入无线信道中传送。在经过传送之后，MAC PDU 被还原成原来的 MAC SDU，以实现对等层间的透明传送。MAC PDU 的传送过程如图 3-13 所示。

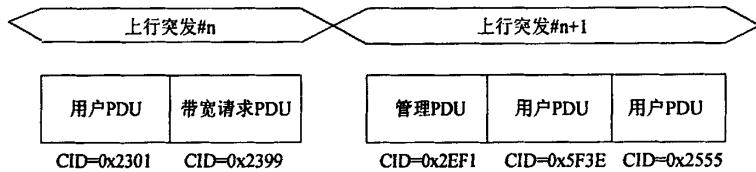


图 3-12 MAC PDU 的连接

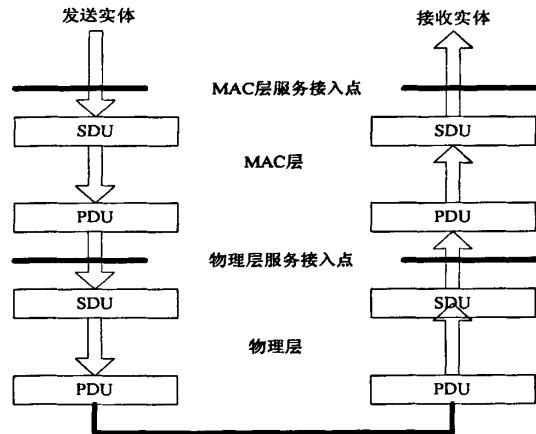


图 3-13 MAC PDU 的传送

### 3.2.5 上行链路调度服务

设计调度服务是为了提高轮询/授予进程的效率。通过指定一个调度服务及其相关的 QoS 参数，BS 就可以预期上行链路业务的吞吐量和时延需求并在适当的时候提供轮询/授予。上行链路调度按服务的对象可划分为 4 种基本调度服务：主动授予服务(UGS, Unsolicited

Grant Service)、实时轮询服务(rtPS, Real-Time Polling Service)、非实时轮询服务(nrtPS, Non-Real-Time Polling Service)和尽力传输服务(BE, Best Effort service)。

### 3.2.5.1 主动授予服务

该服务用来支持周期性、定长分组的CBR业务流(如T1/E1和无话音静默期抑制的VoIP业务)。BS实时、周期地向携带该业务的UGS连接提供固定带宽授予，以减少SS请求开销并保证满足业务流的实时需求。即该连接一旦接入成功后，应禁止其参与竞争请求机会(Request IE)，并且BS也不为其提供单播请求机会，除非授予管理子帧头(GMS, Grant Management Subheader)的PM比特置1时，才允许其为一个非UGS连接请求带宽轮询。因此，UGS连接只能使用BS主动提供的周期性带宽授予(Data Grant Burst Types IE)来发送数据。

此外，针对UGS连接，授予管理子帧头中的Slip Indicator(SI)比特可用来将与UGS业务流相关状态信息从SS传递到BS。SS一旦探测到本次业务流超出它的发送队列长度深度，就设置该比特为1，恢复到极限内则清除该SI标记(为0)。该标记允许BS通过发布额外的授予来提供长期条件补偿，如map消息丢失和时钟失步。BS分配的带宽不应超过Active QoS Parameter Set的Maximum Sustained Traffic Rate参数规定的带宽，除了当授予管理子帧头的SI域被置1时，BS可以为时钟速率失配补偿而额外授予1%的带宽。

### 3.2.5.2 实时轮询服务

用来支持周期性、变长分组的实时VBR业务，如MPEG视频业务流。和UGS不同，BS向携带该业务的rtPS连接提供实时、周期的单播请求机会。从而使该连接能周期地告知BS其变化的带宽需求(通过BRH方式)，BS也就能周期地为其分配可变的突发带宽(Data Grant Burst Types IE)供其发送变长分组。这种业务比UGS的请求开销大，但能使BS按需动态分配带宽以达到最佳数据传送效率。和UGS一样，该业务的实时性要求禁止rtPS连接使用竞争请求机会(Request IE)。即便是授予未决时，BS也可以发起该业务描述的单播请求机会。因此，rtPS连接只能使用单播请求机会和主动授予的Data Grant Burst Types IE作为上行发送机会。

### 3.2.5.3 非实时轮询服务

用来支持非周期、变长分组的非实时VBR业务流，如高带宽的FTP业务流，它需要变长的突发带宽。因此，要求BS有规律地向携带该业务的nrtPS连接提供单播请求机会，以保证即便在网络阻塞时，该连接也有机会发出带宽请求。典型地是，BS通常以秒为数量级的间隔，周期或非周期地轮询该nrtPS连接。该连接通过发送携带CID的带宽请求消息来响应轮询。由于非实时性，可允许该nrtPS连接使用竞争请求机会(Request IE)。因此，nrtPS连接既可使用单播请求机会、主动授予的Data Grant Burst Types IE，也可使用竞争请

求机会作为上行发送机会。

### 3.2.5.4 尽力传送业务

用来支持BE业务。其特点是不提供完整的可靠性，通常执行一些错误控制和有限重传机制。其稳定性由高层协议来保证。这种连接也可使用单播请求机会、主动授予的Data Grant Burst Types IE，但更多的是使用竞争请求机会来发送带宽请求。

## 3.2.6 MAC层对物理层的支持

MAC协议支持3种双工技术：Unframed FDD，Framed FDD，Framed TDD。选择不同的双工技术将影响系统特性和物理层参数。本MAC协议支持framed和unframed PHY规范，对于framed PHY，MAC层调准它的调度时间间隔和底层物理层成帧一致；对于unframed PHY，MAC层选择不同的调度时间间隔来优化系统性能。

### 3.2.6.1 未成帧频分双工（unframed FDD）

对于采用 Unframed FDD 双工模式的物理层，上、下行发送使用不固定的帧长。这种系统中，下行信道一直存在，所有 SS 也一直能够侦听到。因此在下行信道上，业务流以一种使用 TDM 的广播方式向下传送；上行信道则使用 TDMA 方式为各 SS 共享，在共享的地方有一个集中预定器控制上行带宽的分配。

### 3.2.6.2 成帧频分双工（framed FDD）

对于采用 framed FDD（也称 burst FDD）双工模式的物理层，上下行发送使用固定的帧长在不同频段同时收发，在下行信道上，业务流以突发方式发送。这种模式有助于利用不同的调制方式和同时使用全双工 SS 和可选的半双工 SS。如果使用半双工 SS，带宽控制器就不能在 SS 接收下行信道数据的同时给上行信道分配带宽。图 3-14 描述了基本的 Framed FDD 操作模式。上下行信道使用固定帧长可简化带宽分配算法。由图可见，全双工 SS 能连续侦听下行信道，而半双工 SS 仅当上行信道不发送数据时才能侦听下行信道。

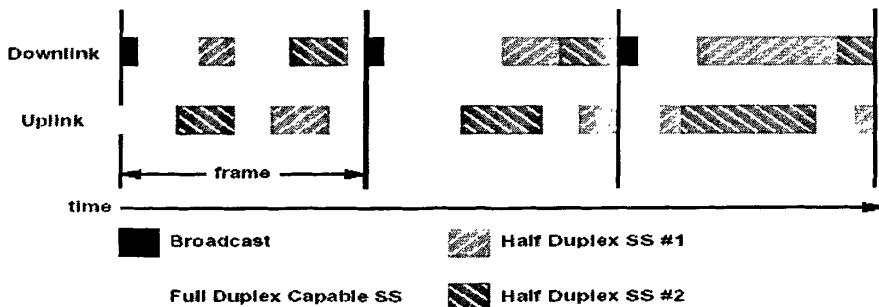


图3-14 Burst FDD带宽分配示意图

### 3.2.6.3 时分双工（TDD）

对于TDD模式，上下行链路通常共享相同频段但在不同时间发送数据。一个TDD帧也有一个固定帧长并包含一个上行子帧和下行子帧。一帧由整数倍的物理时隙(PS, Physical Slot)组成，这使得容易分割带宽。TDD帧长固定但其内部上下行子帧的长度可调，使分配给上下行的带宽可以根据需要变化。上下行子帧的分割由系统内部较高层的一个参数来控制。FDD的上下行子帧是同时发送，不存在先后。TDD的帧结构如图3-15所示。

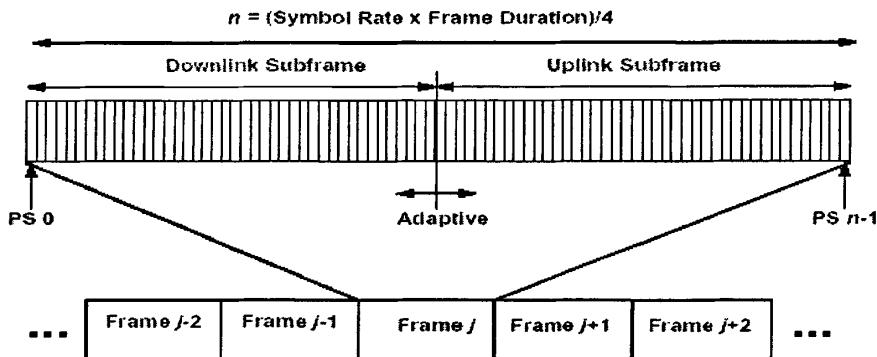


图3-15 TDD 帧结构

### 3.2.6.4 下行链路 MAP 消息(DL-MAP)

DL-MAP是BS下行发送的MAC层管理消息之一（在下行子帧的广播控制域中）。它定义了对下行链路中信息的访问方式，如定义了在下行链路上通过SS进行的时分复用burst和时分多址burst的起始时间。简言之，DL-MAP消息定义了按“burst”突发模式工作的物理层的下行链路时间间隔的用法。

### 3.2.6.5 上行链路 MAP 消息 (UL-MAP)

UL-MAP是由BS下行发送的，用作SS上行链路带宽分配的MAC层管理消息（在下行子帧的广播控制域中）。它规定以某个调度时间间隔SS接入信道（负责向SS分配上行接入信道）。它是以“微隙”为单位，多个“微隙”逻辑上构成了称为IE的上行时间间隔。总之，UL-MAP是根据相对于前一个IE起始的偏移值（由众多“微隙”组成）来规定上行链路的使用。

#### 3.2.6.5.1 上行链路定时 (Uplink timing)

SS上行发送定时是基于“上行时戳”这个参考时间的。该时戳由BS通过下行广播消息DL-MAP传给所有SS。它是个计数器，以16倍PS的速率递增。因此它有一个等于PS持续时间1/16的高分辨率。这使得SS能够以一个较小的时间偏移跟踪BS时钟。

对于 unframed PHY，SS 发送的上行定时是靠 BS 下行广播的 DL-MAP 消息中含有的“上行时戳”(称 BS 时戳)为参考来实现的。BS 向所有 SS 广播这个“上行时戳”值，各 SS 便用它来调整其内部 SS 时戳以便能追踪 BS 的定时。其调整方法是，SS 根据收到的 BS 时戳和收到的 RNG-RSP 消息中的 Timing Adjustment 参数计算出一个相对于 BS 时戳的偏移即 SS 时戳(分析：SS 时戳=BS 时戳+Timing Adjustment)。该偏移使上行 burst 在合适的时间到达 BS。当 BS 时戳和 SS 时戳到达最大值( $2^{29}-1$ )后复 0 并继续计数。

对于 framed PHY，SS 发送的上行定时不需要下行广播一个“上行时戳”计数器，而是以下行子帧起始为参考。具体地讲是以 UL-MAP 消息中的 Allocation Start Time 参数为参考点。并且 UL-MAP 消息的这个参考点可能位于当前帧或下一帧。因此，对 TDD 帧结构来说，UL-MAP 的“分配起始时间”可能位于当前帧或下一帧的上行子帧起始部分；对 FDD 来说，可能为当前上行帧的某个偏移(信号往返程时延+UL-MAP 消息的处理时间)，也可能位于下一上行帧的起始部分。

总之，无论对于 unframed PHY 还是 framed PHY，虽然上行定时概念不同，但 SS 都要基于 BS 发送的 RNG-RSP 消息中的 Timing Adjustment 参数来调整其各自的上行定时。(分析：二者虽然参考点不同，但都要加上 Timing Adjustment 这个上行发送定时调整量，才得到 SS 上行发送时间。该调整量补偿信号由 BS→SS→BS 的往返传播路径时延和 SS 处理接收 UL-MAP 消息时延决定)

### 3.2.6.5.2 上行链路微隙定义 (Uplink minislot definition)

UL-MAP 消息以“微隙”(minislot)为单位。“微隙”由若干个“物理时隙”组成，其大小在每个上行信道的UCD消息中给定。“微隙”和“物理时隙”的长度关系是：1 minislot = n个PS，(n=1~255)，物理时隙由4个Symbol组成。

### 3.2.6.5.3 上行链路间隔定义 (Uplink interval definition)

上行信道间隔，也即带宽分配由一系列IE定义，而IE由多个微隙组成。下面定义的所有IE应该为SS所支持，在创建一个UL-MAP消息时，SS可以使用这些IE中的任何一个。

Request IE —— 通过请求 IE，BS 指定一个上行时间段，让 SS 为发送的上行数据申请带宽。该 IE 的特性依赖于它使用的 CID 类型而变化。如果 BS 作出广播，则该 IE 表示邀请各 SS 发送竞争请求。如果是单播，则表示邀请某个特定的 SS 发出带宽请求。单播可被用作 QoS 调度方案的一部分，具体方案随厂家有所不同。在该 Request IE 时间段内发送的 PDU 应该使用带宽请求头的格式。

Initial Maintenance IE —— 初始维护IE，简称IM IE。通过它，BS指定一个上行时间段，让新的SS加入网络。

在一些UL-MAP消息中，应该提供一个长的时间段（等价于最大往返程传播时间+RNG-REQ消息的发送时间）以允许新的SS执行初始测距（initial ranging）。在该IE时间段发送的PDU应使用RNG-REQ这个MAC管理消息格式。

Station Maintenance IE —— 站维护IE，简称SM IE，通过它BS指定一个上行时间段，让SS执行一些例行的网络维护工作，如周期测距和功率调整。通常BS周期地为各SS分别预留一段时间，即SM IE以单播方式为执行该任务提供上行带宽。

Data Grant Burst Type IE —— 通过数据授予突发类型 IE，BS为某个SS提供一个发送一个或多个上行PDU的机会。发送这些IE要么是为了响应来自某个SS的请求，要么是由于某管理策略而向某个特定SS提供一定数量的带宽。总共定义了六种不同的Data Grant Burst Type。Data Grant Burst Type 1~6即为UL-MAP消息中UIUC= 4~9的IE，而请求IE、初始维护IE和站维护IE分别为UIUC=1, 2, 3的IE。在UL-MAP消息中定义了每种Data Grant Burst Type的具体描述。

Null IE —— 一个空 IE将终结IE列表中所有的实际分配，它是前一个Data Grant Burst Type IE的结束偏移，用来界定最后一个实际分配间隔的长度。（UIUC=10）

Empty IE —— 空 IE指示上行发送暂停。SS在Empty IE期间不应发送。（UIUC=11）

### 3.2.6.6 MAP 相关和同步

#### 3.2.6.6.1 成帧 PHY 系统的 MAP 消息关联

对于 framed PHY 系统，DL-MAP 消息中携带的信息属于当前下行帧（即 SS 收到的含有 DL-MAP 信息的帧）。UL-MAP 消息中携带的信息则可能属于当前上行帧或下一上行帧上的某个时间间隔，该时间间隔起始于 Allocation Start Time（以 SS 收到含有 UL-MAP 消息的那帧的起点为衡量），结束于最后一个分配的微隙之后。这种定义对 TDD 和 FDD 两种 framed 运作方式都适用。图 3-16 和图 3-17 是 framed TDD 情况下的物理层和 MAC 层的控制信息之间的时间关联示意图；图 3-18 和图 3-19 则是相应的 framed FDD 的情况。

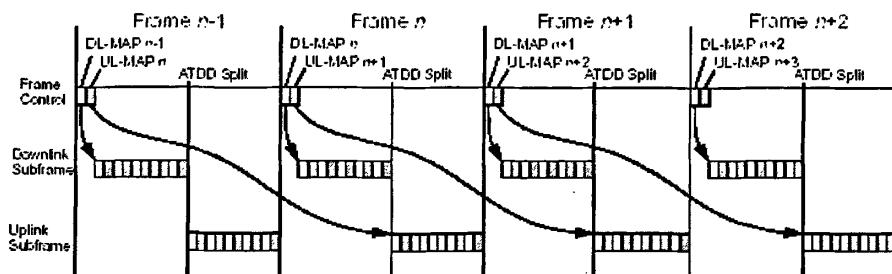


图3-16 PHY层和MAC层的控制信息的最大时间关联（TDD）

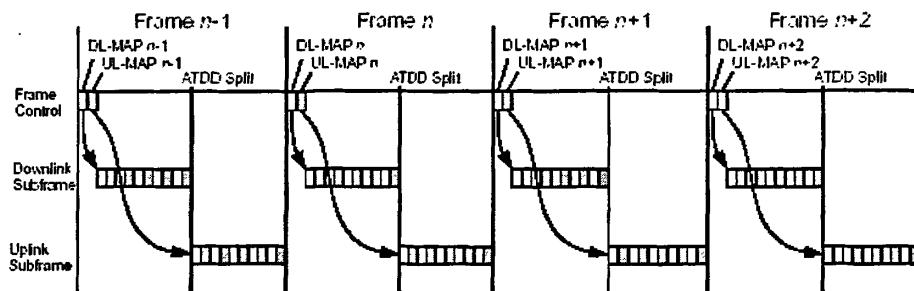


图3-17 PHY层和MAC层的控制信息的最小时间关联 (TDD)

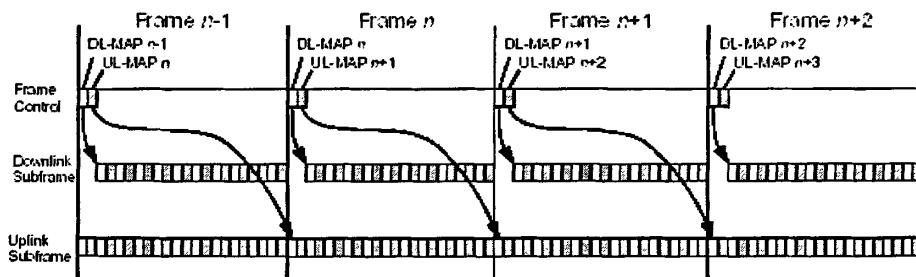


图3-18 PHY层和MAC层的控制信息的最大时间关联 (FDD)

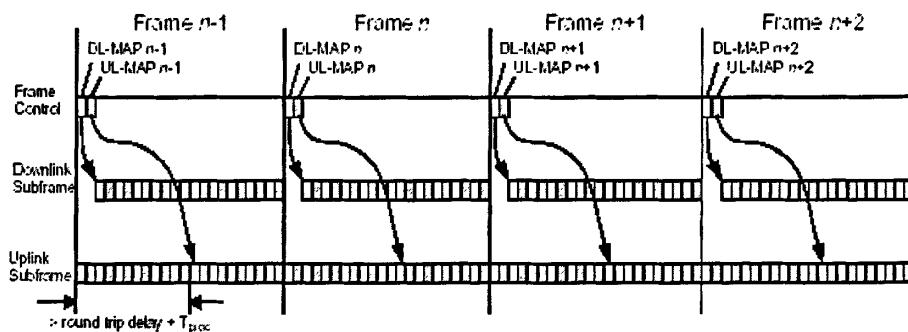


图3-19 PHY层和MAC层的最小时间关联 (FDD)

### 3.2.6.6.2 未成帧 PHY 系统的 MAP 消息关联

对于unframed PHY系统，DL-MAP消息只携带一个“上行时截”而并没有定义要发送什么信息。所有SS持续地从下行链路信号中搜索一切寻址给它们的消息。UL-MAP消息包含一个时截，它指示了该UL-MAP消息定义的第一个微隙的位置。

从UL-MAP结束( $t_0$ )到它定义的第一个上行间隔时间的起始( $t_1$ )这段时延( $t_1-t_0$ )应该大于最大往返程时延与SS处理UL-MAP所需时间之和。图3-20为未成帧UL-MAP信息的时间关联。

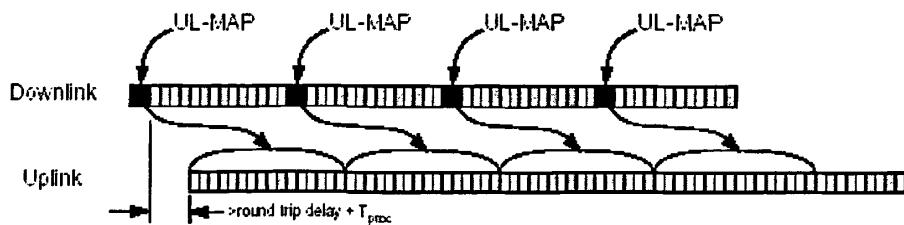


图3-20 UL-MAP信息的时间关联 (未成帧FDD)

### 3.2.7 服务质量

协议定义了“业务流 QoS 调度”，“动态业务建立”，“两阶段激活模型”几种 QoS 相关的概念。限于篇幅对于后 2 种将不作描述。协议规定的 QoS 需求包括以下几个方面：

- 对基于 SS 的 QoS 业务流和业务参数进行预配置和注册功能。
- 动态建立具有 QoS 的业务流和业务参数的信令功能。
- 利用上行业务流的 MAC 层调度和 QoS 业务参数。
- 利用下行业务流的 QoS 业务参数。
- 将业务流属性分组成具有不同名字的业务类别，以便上层实体和外部应用能以全局一致的方法用所需的 QoS 参数请求业务流。

提供 QoS 的主要机制是将进入整个 MAC 层的分组和一个由 SFID 标识的业务流相关联。BS 和 SS 根据协议定义的 QoS Parameter Set 来提供 QoS，它主要是定义空中接口上各种调度业务及其分组的发送优先顺序和时延处理。但这些特征需要和空中接口外的其它机制结合才能最佳地提供端到端的 QoS。

#### 3.2.7.1 业务流

业务流是一个 MAC 层传输业务，它是一个提供某种 QoS 描述的单向（上行或下行）分组流。业务流由一组诸如：时延、抖动、吞吐量等 QoS 参数表征。这些属性包含了 SS 如何请求上行链路微隙以及期望 BS 的上行调度器采取何种行为的详细细节。

##### 3.2.7.1.1 业务流属性

**Service Flow ID:** 是各种业务流的标识符。业务流可以处于非激活状态或激活（活动）状态。一个业务流至少有一个 SFID 和一个上下行方向指示 Direction，处于活动状态业务流还有一个 16bit 的 CID 与之关联。

**Connection ID:** 映射到一个业务流的连接标识符。仅仅当该连接有一个允许的业务流时才使用。

**ProvisionedQoSParameterSet (配置 QoS 参数组):** 由本协议范围以外的网络，如网管

系统提供的一组 QoS 参数。

**AdmittedQoSParameterSet** (允许 QoS 参数组): BS(可能 SS)用来描述某种预留资源的一组 QoS 参数组。主要资源如: 带宽 (“微隙”), 但也可包含用来激活该业务流的任何其它内存或基于时间的资源。

**ActiveQoSParameters** (活动 QoS 参数组): 真正提供给某个业务流的一组 QoS 参数。只有一个活动业务流才可以转发分组。

**Authorization Module** (授权模块): 在 BS 中允许或拒绝对一个业务流的 QoS 参数和分类器改变的功能模块。它定义了一个限制 AdmittedQoSParameterSet 和 ActiveQoSParameters 可能取值的“包络”。

QoS 参数组间的关系是: ActiveQoSParameters 是 AdmittedQoSParameterSet 的子集, 而后者又始终是一个包络的子集。在动态授权模型中, 包络由 Authorization Module 决定。在配置授权模型中, 包络由 ProvisionedQoSParameterSet 决定。

### 3.2.7.1.2 三种类型的业务流

由于存在 3 种 QoS 参数组, 也就对应存在 3 种业务流。

**Provisioned** (配置业务流): 这种类型业务流经由网管系统提供, 它的 AdmittedQoSParameterSet 和 ActiveQoSParameters 参数组是空的。

**Admitted** (允许业务流): 这种类型业务流由 BS 按照参数组 AdmittedQoSParameterSet 预留资源。但这些参数处于非活动状态, 因为它的 ActiveQoSParameters 参数组是空的。

**Active** (活动业务流): 这种类型业务流由 BS 按照 ActiveQoSParameters 参数组来处理资源。如: 为一个 UGS 业务流活动地发送包含主动授予的 MAP 消息 (Grant Management Subheader 中地 Poll-Me 比特置 1)。

### 3.2.7.2 对象模型

QoS 体系中的主要对象有 PDU, Service Flow, Connection, Service Class。如图 3-21 所示。每个对象有一系列属性。图中下划线强调的属性是唯一标识对象的属性, 方括号中的属性是可选属性。对象之间的对应关系由框间连线及其端点上的数字表示。例如: 一个业务流和 0~N 个 PDU 关联, 但一个 PDU 只和一个业务流关联。

CS 层的进程将外部数据递交到 MAC SAP 进而继续传递。递交到 MAC SAP 的信息包括了标识该连接的 CID, 这个连接上的业务流就映射到 CID。

Service Class 是一个由 ASCII 码的 Service Class Name 名字唯一标识的, 由 BS 所实现的可选对象。它定义了某种特定的 QoS 参数组。一个业务流具有的 QoS 参数组中就可能包含这个 Service Class Name, 作为 “宏” 来从该业务类别中选取 QoS 参数。业务流 QoS

参数组可以增加和覆盖由 Service Class 设定的 QoS 参数组。

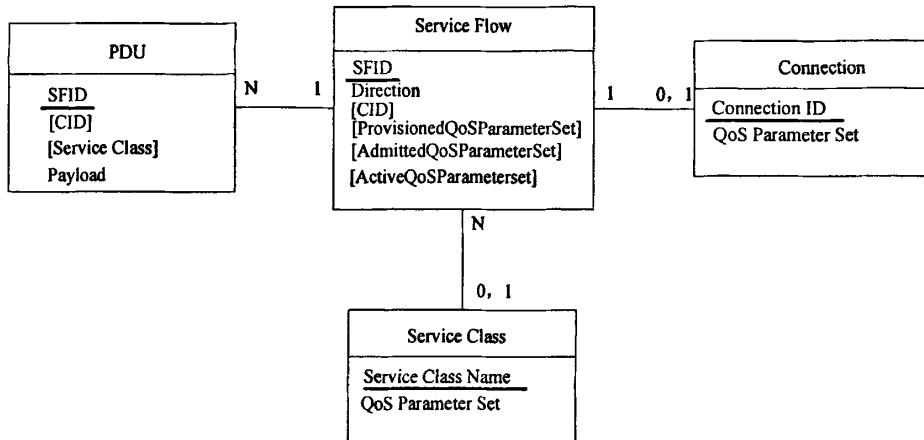


图 3-21 对象模型的操作理论

注：Service Class 只是某个特定 QoS 参数组的识别符。因此，使用业务类别是可选的。由一个 Service Class 标识的业务一旦建立，和明确指定相同 QoS 参数组的业务没什么区别。一个业务流可以用以下 3 种方法中任何一种来指定其 QoS 参数组。

- 明确地将所有业务参数包括进来。
- 通过指定一个 Service Class Name。
- 通过指定一个 Service Class Name 和修改参数一起来指定（动态改变）。

### 3.3 安全子层

Privacy 子层位于 MAC CPS 子层下方，其主要功能是对在 BS 和 SS 之间连接上传递的数据进行加密，并提供认证和密钥交换。该层协议采用一个授权模式的客户/服务器密钥管理协议。BS 作为服务器控制对作为客户端的 SS 的密钥分发。此协议中还允许添加采用基于 SS 认证的数字证书来增强保密机制。

Privacy 子层主要由以下两部分内容组成：

(一) 数据包的加密封装协议，即采用数据封装协议来对在固定宽带无线接入网上传输的分组数据进行加密。此协议定义了 (1) 一系列的认证和加密算法，例如：RSA,DES 算法等；(2) 将这些认证或加密算法运用到 MAC 层协议数据单元中净荷部分的规则。

(二) 密钥管理协议(PKM 协议)：提供 BS 与 SS 之间安全的密钥分配机制。此协议保证了 SS 与 BS 的密钥同步。PKM 协议是加密层里的核心内容，

SS 通过 PKM 协议从 BS 获得授权密钥(AK)和会话密钥(TEK)，并实现周期性的再认证和密钥更新。其实现分别是在 AK 有限状态机和 TEK 有限状态机的控制下完成的。

## 第四章 基于 DSP 的 MAC 层实现的系统设计

### 4.1 系统硬件结构

终端的系统结构如图 4-1 所示，包括 ARM 主控模块、DSP 实时数据处理模块和 FPGA 信号处理模块三个主要部分，还含有数据采集控制、主从系统及其外围器件、ARM 和 DSP 接口、ARM 和 FPGA 的接口、FPGA 和 DSP 的接口等部分。

ARM 主控模块负责系统监控和管理，协调系统的工作。ARM 装配有 Linux 操作系统，通过裁剪和修改的 Linux 内核实现了底层系统的应用程序接口以及 HPI(Host Port Interface) 底层驱动程序(ARM 部分)。通过 DSP 自带的 HPI 并行接口，ARM 可以直接访问 DSP 的存储空间，包括映射的外围设备。

DSP 实时数据处理模块提供完整的 HPI 驱动程序(DSP 部分)、IEEE 802.16 MAC 层控制协议以及相应数据处理控制算法程序，利用 HPI 并行接口与主机 ARM 进行通信，实现各种实时处理和控制。DSP 与 FPGA 之间通过数据总线、地址总线进行通信。

FPGA 信号处理模块主要负责接收来自 DSP 的数据和控制参数后，通过 MII(Media Independent Interface)接口控制收发器完成数据的发送，或者是缓冲收发器接收到的数据，以中断方式通知 DSP 进行数据的接收。根据 DSP 提供的控制参数，FPGA 进行信号提取，并完成信道编解码、CRC 附着、交织、调制解调等物理信道的基带数据运算。FPGA 和 ARM 之间的接口供 ARM 对 FPGA 的工作参数进行控制。

图 4-1 中所示的 ARM 芯片选择了三星电子公司生产的 S3C2410[14]。S3C2410 是基于 ARM920T 处理器核心、采用 0.18mm 制造工艺的微处理器，具有 16KB 指令和 16KB 数据 Cache、内存管理单元(MMU)、支持 TFT 的 LCD 控制器、NAND 闪存控制器、3 路 UART、4 路 DMA、4 路带 PWM 的 Timer 、I/O 口、RTC、8 路 10 位 ADC、Touch Screen 接口、IIC2BUS 接口、IIS2 BUS 接口、2 个 USB 主机、1 个 USB 设备、SD 和 MMC 接口和 2 路 SPI，S3C2410 处理器最高可运行在 268MHz。

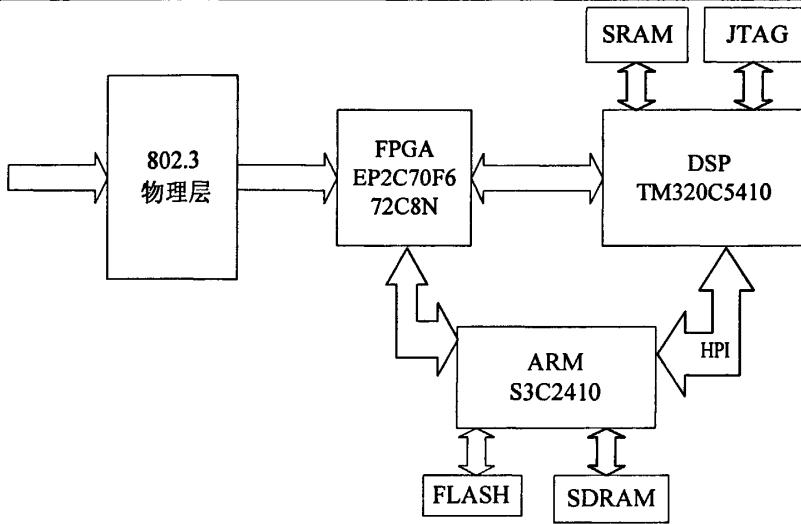


图 4-1 终端的系统结构

本系统 DSP 芯片选用了 TI 公司的定点型 DSP TMS320C5410[15]，其特点如下：操作速率可达 100Mips(百万条指令每秒)；具有先进的多总线结构(1 条程序总线、3 条数据总线和 4 条地址总线)；40bit 算术逻辑单元；可进行非流水线式单周期 MAC 运算；内置 64K RAM、16K ROM；带有软件可编程等待状态发生器、锁相环 PLL 时钟发生器；一个 8 位并行主机接口、3 个多通道缓冲串行口等。该芯片采用 3.3V 和 2.5V 双电源供电，其中 IO 采用 3.3V 电源供电，内核采用 2.5V 电源供电。

在 FPGA 的选择上，Cyclone II 器件[16]系列是成本最低的 FPGA 解决方案，提供高度的设计灵活性和高性能系统集成能力。Cyclone II 器件架构提供多达 68416 个逻辑单元(LE)；采用了 TSMC 领先的 90nm 低电介工艺技术；基于流行的 M4K 存储器块，提供多达 1.1 兆比特的嵌入式存储器，可以支持多种操作模式，包括 RAM、ROM、先进先出(FIFO)缓冲器以及单端口和双端口模式；提供最多 150 个  $18 \times 18$  比特乘法器，可用于实现通用 DSP 功能；提供高级外部存储器接口支持，数据速率最高可达 668 Mbps；支持串行总线和网络接口(如 PCI 和 PCI-X)，快速访问外部存储器件，同时还支持大量通讯协议，包括以太网协议和通用接口；支持最多达四个可编程锁相环和最多 16 个全局时钟线；Nios II 嵌入式处理器给低成本分立式微处理器提供了一个理想的替代方案；可以采用 Altera 的低成本串行配置器件进行配置，来提供最大 64 兆比特的 Flash 存储器。

图 4-1 中的 802.3 物理层即以太网收发器是供基带仿真调试所用（本项目未考虑射频实现），它采用 Intel 公司的 LXT972A[17]快速以太网物理层协议芯片，支持速率 10Mbps 和 100Mbps 的全双工通信，并提供三种工作模式：自动协商、并行检测、手动控制。LXT972A 通过 MII 接口与 FPGA 进行通信。

此方案的制定具有相当大的灵活性，对于软件或者硬件的开发都可按照需要进行设计和完善，而且在必要时可自行升级以符合环境需要。

## 4.2 MAC 层接口设计

### 4.2.1 接口要求

MAC 层与物理层，即 DSP 与 FPGA 之间以总线方式相连。此时，FPGA 被当成是 DSP 的外部存储器。在进行数据访问和控制时，接口部分共定义了四个 16 位的控制寄存器。FPGA 在接收到有效数据后以中断方式通知 DSP。DSP 可以直接访问 FPGA 的内部存储器空间，也可以响应 FPGA 产生的中断请求(外部中断引脚 INT0)。

MAC 层与 LLC 层间的通信，也就是 DSP 与 ARM 之间通信的核心是 DSP 器件本身带有的 HPI 并行接口。DSP 部分提供完整的 HPI 驱动程序(DSP 部分)，从 ARM 获取数据，然后交给通信协议及相应数据处理控制算法程序；ARM 部分提供 HPI 底层驱动程序，传递上层命令给 DSP。DSP 利用 HPI 并行接口与主机 ARM 进行数据交换。

### 4.2.2 MAC 层与物理层的接口

MAC 层与物理层之间通信接口(MAC\_PHY 接口)负责协调和规范 MAC 层和物理层的信息的交互的过程和方式。MAC\_PHY 接口主要分成两部分：DSP 部分负责向 FPGA 的发送缓冲写入数据和从 FPGA 的接收缓冲读取数据；FPGA 部分在发送时负责将发送缓冲中的数据发送到空中接口，在接收时负责缓存接收数据并通知 DSP。DSP 与 FPGA 的硬件总线连接设计如图 4-2 所示。

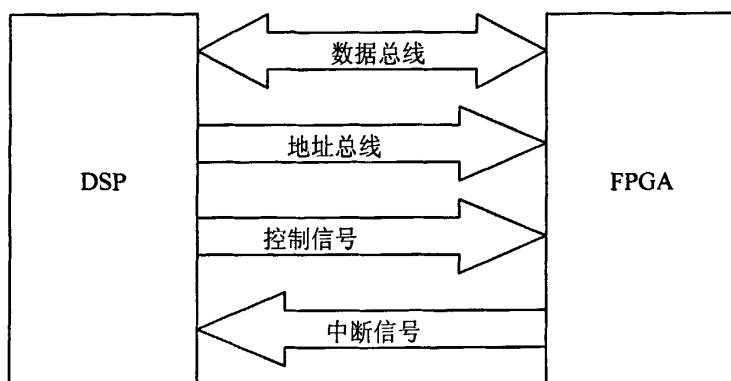


图 4-2 DSP 和 FPGA 的总线接口

为了实现 DSP 与 FPGA 之间的信息的交互，我们定义了 4 个 16 位的控制寄存器和 3 个宽度为 4bit，深度为 4096 的缓冲 RAM。控制寄存器主要用来存储控制信息，包括总体

控制寄存器 ACNTL、发送相关寄存器 TCNTL1 和 TCNTL2、接收相关寄存器 RCNTL。ACNTL 寄存器主要存储系统的状态信息和部分控制参数，如信道忙/闲状态、中断响应状态等；TCNTL1 和 TCNTL2 寄存器用来存储 DSP 发送数据时给 FPGA 的一些指示信息，如发送数据的长度、调制方式、发送速率等；RCNTL 存储 DSP 从 FPGA 接收数据时需要的一些参数信息，如接收缓冲 RAM 指示信息、接收数据长度等。

三个缓冲 RAM 分别为接收缓冲 RAM0 和 RAM1、发送缓冲 RAM2，接收缓冲 RAM 用来缓冲 FPGA 从信道中接收的有效数据，发送缓冲 RAM 被 FPGA 用来暂存 DSP 发送的数据。这三个缓冲 RAM 均映射到 DSP 的地址空间，DSP 可以通过映射地址直接读写它们。在接收时，DSP 根据 RCNTL 的第 14 比特位来获知应从哪个缓冲器接收数据（RAM0 或 RAM1）。

当终端处于发送状态时，在完成退避并且媒体空闲时间满足要求后，DSP 在写入数据到发送缓冲 RAM2 后，将控制信息写入寄存器 ACNTL、TCNTL1 及 TCNTL2，通知 FPGA 有数据需要发送。FPGA 在查询到有数据要发送后，调用编码模块对数据进行编码，最后通过收发器将数据发送至信道。

当终端处于接收状态时，FPGA 将从信道中接收的数据存储于接收缓冲 RAM0(1)，并以中断方式通知 DSP。DSP 在中断信号的中断处理程序中，根据 RCNTL 寄存器的指示到相应 RAM 映射地址读取数据。接着处理数据，处理过程包括辨别当前帧是否是广播帧或者目的地址是否是本站点、重复帧的过滤、去除 MSDU 头部、分段的重组、收到帧的应答等。如果收到的是不需重组或者是重组已完成且正确接收的数据帧或者管理帧，则通过 HPI 并行接口将数据发送至 ARM。

#### 4.2.3 MAC 层与上层的接口

MAC 层与上层的接口用来协调和规范 MAC 层和上层信息交互的过程和方式，该接口也分成两部分：DSP 部分完成从 ARM 获取数据，并交给通信协议及相应数据处理控制算法程序的工作；ARM 部分负责传递上层命令给 DSP 和从 DSP 接收 ARM 的数据。同时在该接口上完成 MAC 层可重配置参数的传递，由 ARM 通过 HPI 口控制 DSP 完成 MAC 层协议部分功能模块的参数重配置。

DSP 与 ARM 之间通信主要通过 HPI 并行接口，通过该接口 ARM 微控制器可以直接访问 DSP 的内部存储器空间或地址映射到存储器空间的外围设备，还可以响应 DSP 发过来的中断请求，对于 ARM 来说，DSP 系统芯片就相当于一个外部设备。具体的硬件连接设计如图 4-3 所示。

TMS320VC5410 芯片中 HPI 接口模块由 8 位数据线 HD[7:0]和 10 条控制线构成。10 条控制线中,两条地址线 HCNTRL[1:0]负责对 HPI 口的内部三个寄存器(控制寄存器 HPIC、地址寄存器 HPIA 和数据锁存器 HPID)进行寻址,从而访问 DSP 内存空间; HBIL 为字节区分信号线; HR/W 为读写控制; HCS 为 DSP 的片选; HRDY 为 DSP 就绪状态; HDS1、HDS2 用于和 HCS 生成内部读写控制逻辑; HINT 为中断请求; HAS 为主机地址使能,通常不使用,接高电平即可。

当上层有数据需要传输时，ARM 通过 HPI 接口将数据写入 HPI RAM，并置位 HPIC 中的 EINT0 位来中断 DSP。中断响应后，DSP 对接收的数据进行处理后交付到 FPGA 进行编码。当以太网收发器有数据到达时，FPGA 通知 DSP 对其进行处理。DSP 将处理过的数据写入 HPI RAM 中，然后通过将 HPIC 的 HINT 引脚置位低电平向主机 ARM 产生中断请求，告诉 ARM 有数据到达。

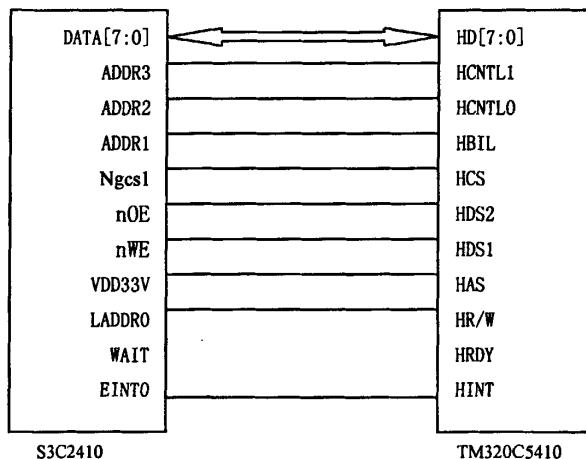


图 4-3 ARM 和 DSP 的总线接口

### 4.3 工作流程

基于图 4-1 所示的硬件平台，装配有 ARM 主控模块、DSP 实时数据处理模块和 FPGA 信号生成模块的终端，在这三个主要模块的协同工作下，实现了 IEEE 802.16 MAC 层协议功能。模块间协同工作过程如图 4-4 所示。

通过 HPI 并行接口，ARM 系统将来自上层的用户信息和控制参数传给 DSP，也可以接收由 DSP 上传的数据。DSP 从 HPI 接口接收到来自 ARM 的数据后，便将数据传递给 MAC 层控制协议程序，以便其对数据进行如下处理，如添加头部、根据消息长度决定是否分段等。接着通过数据总线、地址总线，将处理完的数据和一些控制参数交付给 FPGA。

一旦 FPGA 获知 DSP 有数据需要发送，便根据相应的控制参数对数据进行编码等工作，以使其适合在信道上传输。在完成编码等一系列运算后，便通过 MII 接口控制以太网收发器将数据发送到信道上。

在物理信道上捕获到有效接收信号后，以太网收发器便通知 FPGA 开始数据的接收。在一帧数据接收的开始和结束，FPGA 都会通过中断引脚发送中断信号给 DSP。根据相应的寄存器，DSP 在获知一帧数据接收结束后，就通过数据和地址总线读取 FPGA 的接收缓冲 RAM 中缓存的数据。接着 MAC 层控制协议程序对接收到的数据进行相应的处理，如去除头部、分段的重组、重复帧的过滤、应答当前帧等，并从接收的数据中获取一些有用的信息。数据处理结束后，DSP 通过 HPI 并行接口将处理完的数据，也就是上层数据交付到 ARM。

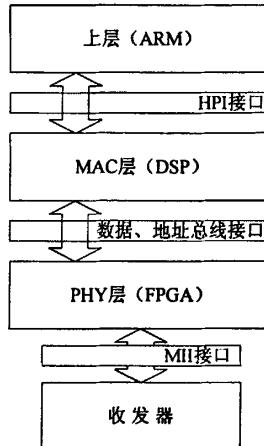


图 4-4 工作流程

## 第五章 IEEE 802.16MAC 层协议软件实现

IEEE802.16 空中接口协议可分为两层，分别为 MAC 层和物理层，本文的主要工作是在 DSP 的平台上设计和实现 MAC 层协议软件，其中 MAC 层协议参考标准为 IEEE802.16d。

在 IEEE802.16 MAC 层协议中，SS 和 BS 端的总体架构基本相同，只是在具体的细节方面有些区别，因此我们在编程过程中，为了考虑代码的可重用性，采用预处理的方式来区别它们。在整个软件设计及实现中，我们采用了模块化的程序结构，便于调试，而且也便于进一步的协议栈重配置功能的实现。

### 5.1 IEEE 802.16MAC 层协议软件架构

IEEE 802.16MAC 层软件采用了模块化的设计和实现方法，根据协议功能，将 MAC 层划分为七个模块，其总体架构如图 5-1 所示。每一个模块因 BS 和 SS 侧的不同而有所区别，在编程实现的过程中，使用预处理的方式来区分 BS 和 SS 侧的不同功能。

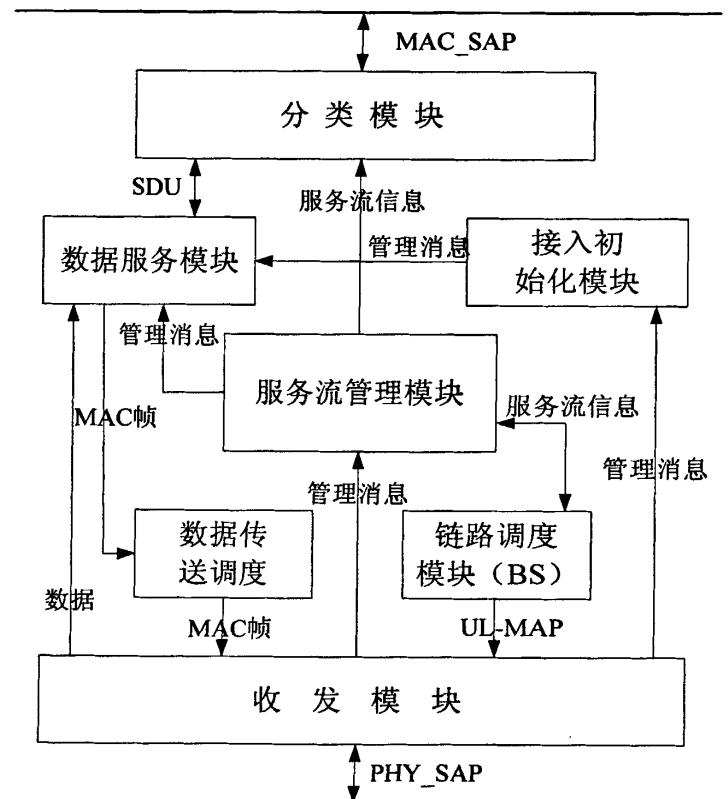


图 5-1 IEEE 802.16MAC 层协议软件总体架构

分类模块从上层接收 SDU，并根据该 SDU 的服务类型查找分类器，找到匹配的 CID，

如果找不到对应服务的 CID，那么就将该 SDU 分类到默认的 CID。在分类之后，将 SDU 交给数据服务模块。此外该模块还将数据服务模块接收到的业务数据交给上层。

数据服务模块接收来自分类模块的业务数据以及其它模块的管理消息，添加 MAC 帧头，形成 MPDU，并将 MPDU 插入到对应服务流的数据队列中。此外该模块还接收来自接收模块的 MPDU，去除帧头并交给分类模块。

服务流管理模块负责对服务流及 CID 的管理，包括服务流创建、服务流修改及服务流的删除。

接入初始化模块负责 SS 和 BS 的初始化工作，主要包括测距、能力协商和注册功能。

数据传输调度模块负责对数据发送队列中基于不同连接的 MPDU 进行调度，在基站侧该模块还将生成 DL-MAP 消息。

链路调度模块是 BS 特有的模块，该模块负责对上行链路进行调度，基于各种服务流特性，为各 SS 分配传输机会，根据调度结果，该模块还将生成 UL-MAP 消息。

收发模块负责数据的收发处理，为即将发送的数据添加 CRC，并通过 DSP 与 FPGA 的接口发送到物理层。在接收端，该模块对接收到的 MPDU 进行 CRC 校验，并根据 MPDU 的类型将该数据交给不同的模块。

模块之间通过定义不同的消息进行通信，消息可以携带参数。通过消息通信的方法使得 MAC 层协议软件更加模块化，更便于协议软件功能的扩展。

## 5.2 MAC 层协议软件调度器

MAC 层协议软件设计的总体思想是基于离散事件处理，而在基于 DSP 的平台上，由于缺乏操作系统的有力支持，无法有效进行多线程同时作业，因此在实现该复杂系统的时候，需要设计一个核心的软件调度器，由调度器负责处理系统中各模块产生的离散消息，对整个 MAC 层协议软件进行调度。

### 5.2.1 总体结构

在 MAC 层协议软件中，调度器负责对系统中各模块产生的消息进行调度，是整个软件的核心模块，其总体结构如图 5-2 所示。

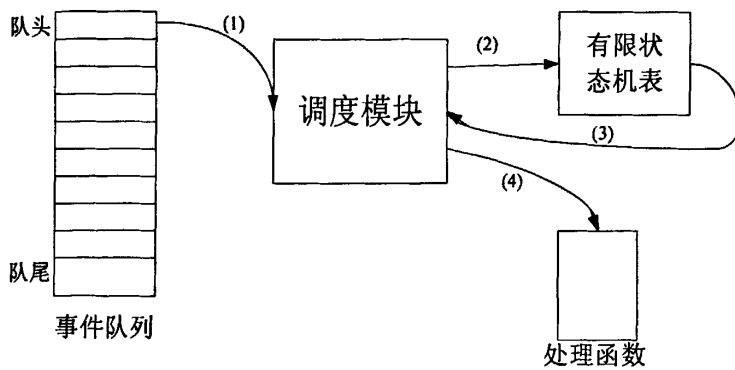


图 5-2 IEEE802.16 MAC 层协议软件调度器

如图所示，调度模块首先从事件队列中取出消息，然后根据消息参数查询有限状态机表，得到对应消息的处理函数，最后执行该处理函数。消息队列和有限状态机表的实现是调度模块实现的核心。

事件队列用于存储系统中各协议模块产生的离散消息，其结构如下所示：

```
typedef struct message_queue_node_s{
    message_id_t           messageid;
    unsigned short          priority;
    void                  *param;
    struct message_queue_node_s *nextmsg;
    system_module_t         tomodule;
    system_module_t         frommodule;
    bool                   msgstate;
}message_queue_node_t;
```

其中 `messageid` 成员标识了消息队列中的消息类型，包括管理消息以及用于各模块交互所需要的消息；`priority` 成员标识了消息的优先级；无类型指针 `param` 标识了该消息携带的参数，该参数可能是指向内存中待处理的一段数据，也可能是系统中其它协议模块所需要的控制信息；`nextmsg` 成员指示队列中的下一消息节点；`tomodule` 成员标识了该消息的目标模块；`frommodule` 成员则标识了该消息的产生模块；布尔类型的成员变量 `msgstate` 标识了消息的处理状态，如果为 `false`，则表示该消息还未得到处理，如果为 `true`，则表示消息已经处理过，调度器将删除队列中已经得到处理的消息节点。

消息队列用链表的结构实现，消息队列的头指针为全局变量 `g_msg_header`，调度器通过该头指针操作消息队列中的消息。

有限状态机表是二维的链表结构，表中存储了对应于各种消息的处理函数。在协议软

件设计过程中，根据协议功能将 MAC 层协议划分成若干个功能模块，这些功能模块的集合组成了一个完整的协议栈功能。其中每一个模块都是一个有限状态机，包括事件的触发、事件的处理以及状态的转换。所以整个协议又是若干个有限状态机的集合。在具体实现的时候，我们把所有的状态机用链表链接起来，形成一个二维表格供调度模块查询。有限状态机表的结构如图 5-3 所示。

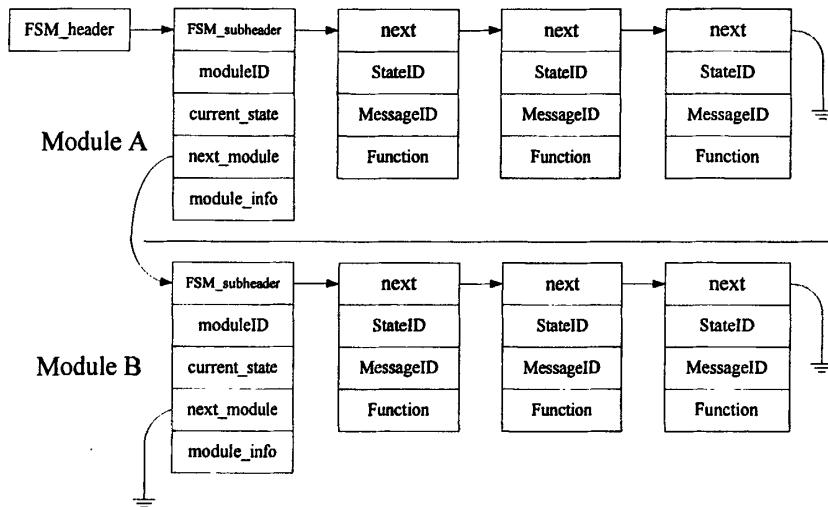


图 5-3 有限状态机表结构

其中，表中的每一行标识一个模块的完整状态机，如果有 N 个模块，那么有限状态机表就会有 N 行。表中每一行都有一个头节点，该头节点标识了该行所属模块以及该模块当前所处的状态。在头节点之后，是多个子节点，每一个子节点都是对应于该模块的一个小状态机。

子节点的数据结构如下所示：

```
typedef struct statemachine_s{
    message_id_t      whichmessage;
    module_state_t     whichstate;
    pfunc             function ;
    struct statemachine_s *next;
}statemachine_t;
```

其中，消息类型成员 whichmessage 标识了该状态机所处理的消息类型；状态类型成员 whichstate 标识了该状态机所处的状态；函数成员 function 标识了该协议模块在 whichstate 状态下接收到 whichmessage 消息时的处理函数；next 则标识了该模块的下一个状态机。

头节点的数据结构如下所示：

```
typedef struct fsm_header_s{
```

```

statemachine_t      *fsm_header;
system_module_t     whichmodule;
module_state_t      current_state;
void                *module_info;
struct fsm_header_s *next_module;

}fsm_header_t;

```

其中，`fsm_header` 成员指向了对应该模块的各个子节点；`whichmodule` 成员标识当前头节点所属的协议模块；`current_state` 成员标识了该模块当前所处的状态，调度程序根据该状态信息查找到对应的处理函数；无类型指针成员 `module_info` 中可存储模块信息，主要用于事务处理；`next_module` 成员指向下一模块的状态机。

整个有限状态机表由全局指针 `g_fsm_header` 标识，调度器通过该指针访问对应于各种消息的处理函数，访问函数主要有：

```

fsm_header_t* fsm_find_module(system_module_t module_id)
pfunc fsm_find_function(statemachine_t *header, module_state_t state,\n
                        message_id_t msg_id)

```

其中 `fsm_find_module` 函数以消息的目标模块为参数，通过全局指针 `g_fsm_header` 查找到对应模块的头结点。然后以头节点中存储的指向子节点的指针、模块的当前状态和消息类型作为参数，查找到对应的处理函数。

## 5.2.2 调度流程

调度器负责对整个协议软件进行调度，其调度流程如图 5-4 所示。

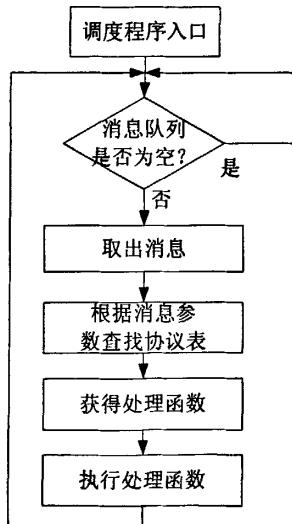


图 5-4 调度流程

如图所示，在进入调度程序后，调度器首先查询消息队列是否为空，如果为空则继续查询直至不为空，如果消息队列不为空，则首先根据消息优先级取出一个消息，然后根据消息的目标模块及目标模块的状态查找有限状态机表，得到对应该消息的处理函数，然后执行查找到的处理函数。在执行消息处理函数的过程中，又会产生新的消息并且插入到消息队列中。因此在协议运行过程中，调度器一直在循环做这样 4 件事情，分别为取消息、查找有限状态机表、获得处理函数以及执行处理函数。所以调度程序的结构大致如下所示：

```
while( 1 )  
{  
    get a message;  
    search in FSM;  
    get the function;  
    execute the function;  
}
```

## 5.3 接入初始化模块

接入初始化模块负责 SS 和 BS 的初始化工作，主要分为三个模块，分别为测距模块、协商模块和注册模块，不同的模块因 BS 侧和 SS 侧而有所区别。

### 5.3.1 测距模块

SS 在接收到 DL-MAP 和 UL-MAP 消息后，在 UL-MAP 消息中搜寻测距间隔，在竞争到发送机会后，SS 使用初始测距 CID 向 BS 发送 RNG-REQ 消息，如果 BS 成功接收到 RNG-REQ 消息，它将返回 RNG-RSP 消息。在 RNG-RSP 消息中包含分配给该 SS 基本及主管理 CID，还可能包含射频功率级别调整，偏移频率调整和时间偏移更正信息。

SS 侧测距模块的 SDL 图如图 5-5 所示。

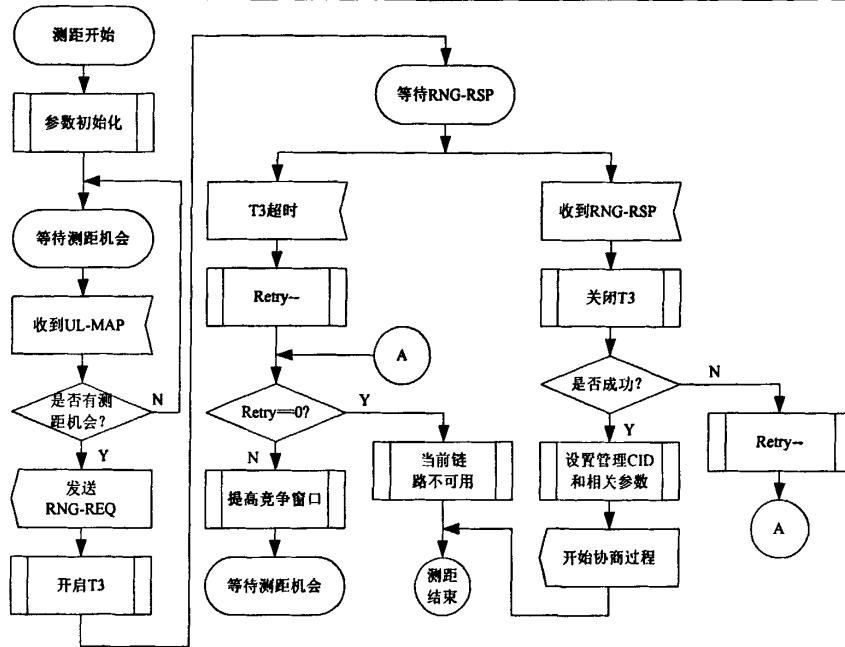


图 5-5 SS 侧测距模块 SDL 图

在 SS 初始化过程中，设置测距模块的状态为测距开始状态，在该状态下，首先对该模块的参数进行初始化，然后状态跳转到等待测距机会状态。在等待测距机会状态下，该模块不停地检查 UL-MAP 消息中是否有提供可用的测距机会，如果有则向 BS 发送 RNG-REQ 消息，然后开启定时器 T3 并将状态跳转到等待 RNG-RSP 状态。如果没有测距机会保持状态不变。

在等待 RNG-RSP 状态下，可能收到两种类型的消息，分别为 T3 超时以及收到 RNG-RSP。

如果是 T3 超时，那么说明 SS 发送的 RNG-REQ 消息出错，BS 无法处理，那么将测距次数 Retry 减一，如果此时 Retry 为 0，则表示当前链路不可用，测距过程结束。如果 Retry 不为 0，那么就提高测距竞争窗口，并重新将状态置为等待测距机会状态。

如果收到 RNG-RSP 消息，说明 SS 发送的 RNG-REQ 消息已经得到基站处理，此时关闭定时器 T3，查看 RNG-RSP 消息内容，此次测距是否成功，如果不成功，那么 Retry 减一，其它处理同 T3 超时。如果测距成功，那么 SS 可以从 RNG-RSP 消息中获得 BS 分配给它的基本 CID 和主管理 CID，并可能调整其它相关参数，如时间、功率等信息。最后向协商模块发送开始协商过程消息。

BS 侧测距模块的 SDL 图如图 5-6 所示。

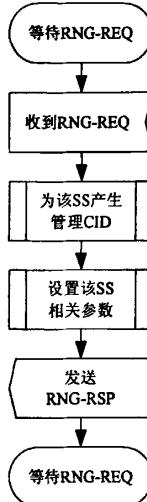


图 5-6 BS 侧测距模块 SDL 图

BS 侧测距模块的状态始终处于等待 RNG-REQ 状态，BS 在该状态下，为各 SS 提供测距功能。在收到 SS 的 RNG-REQ 消息后，BS 的服务流管理模块为该 SS 分配基本及主管理 CID，并将 CID 信息录入到全局变量 g\_ssinfo 中。然后该模块将设置 SS 站点的相关参数，如时间、功率等信息，最后向对应的站点发送 RNG-RSP 消息。

### 5.3.2 协商模块

在完成测距功能之后，SS 发送一个基本能力协商请求消息（SBC-REQ）给 BS，描述 SS 支持的调制级别、编码方案和编码速率，以及双工方式。BS 根据自己的能力接受或拒绝该 SS，并向 SS 返回 SBC-RSP 响应消息。

SS 侧协商模块的 SDL 图如图 5-7 所示。

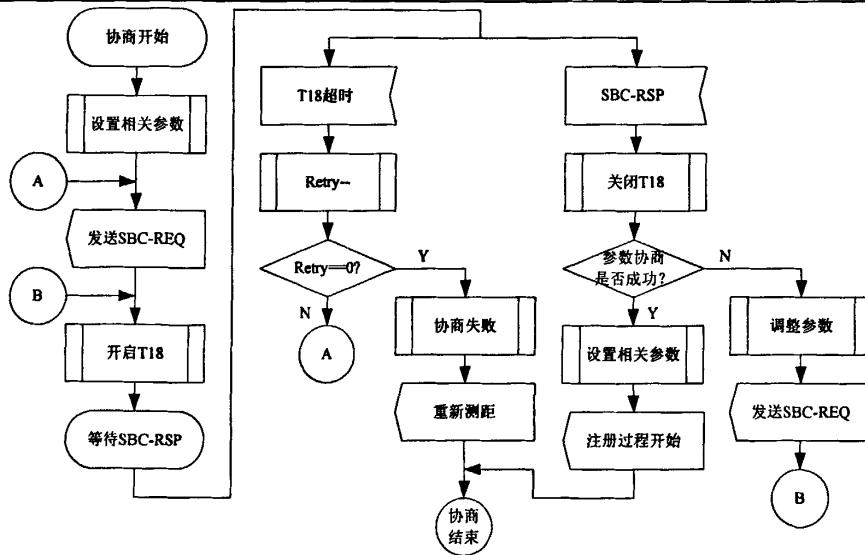


图 5-7 SS 侧协商模块 SDL 图

在 SS 成功测距后，测距模块发送开始协商过程消息给协商模块，此时协商模块开始协商过程。首先测距模块将根据 SS 能力设置相关参数，如编码方式，调制级别等等，然后产生 SBC-REQ 消息并发送出去，此时开启定时器 T18 并将状态跳转到等待 SBC-RSP 状态。

在等待 SBC-RSP 状态下，可能收到两类消息，分别为定时器 T18 超时和收到 SBC-RSP 消息。

如果收到定时器 T18 超时消息，说明 SS 发送的 SBC-REQ 消息在发送过程中出错，此时协商次数 Retry 参数减一，判断 Retry 是否为 0，如果不为 0，则重新发送 SBC-REQ 消息，如果为 0，那么协商过程失败，此时发送重新测距消息给测距模块，重新开始测距过程。

如果收到 SBC-RSP 消息，说明 BS 成功收到 SBC-REQ 消息，此时 SS 关闭定时器 T18，查询 SBC-RSP 消息，查看相关参数是否协商成功，如果不成功，则调整相关参数，重新开始协商过程。如果成功，则根据协商结果调整相关参数，此时协商过程成功结束，协商模块向注册模块发送注册过程开始消息。

BS 侧协商模块 SDL 图如图 5-8 所示。

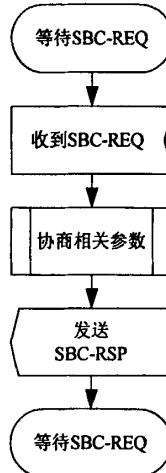


图 5-8 BS 侧协商模块 SDL 图

BS 侧协商模块的状态始终处于等待 SBC-REQ 状态, BS 在该状态下接收来自各个 SS 的 SBC-REQ 消息并进行能力协商, 在收到 SS 的 SBC-REQ 消息后, BS 根据自身能力接受或拒绝 SS 的能力请求, 并将协商好的能力存放到全局变量 g\_ssinfo 中, 最后产生相应的 SBC-RSP 消息发送给 SS。

### 5.3.3 注册模块

在 SS 完成协商之后, 协商模块向注册模块发送注册过程开始消息, 此时注册模块开始注册过程。SS 发送注册请求消息 REG-REQ 给 BS, BS 收到后, 发送响应 REG-RSP 消息。注册消息交互的内容包括所支持的 IP 协议版本、支持的分类选项及 CRC 等信息。

SS 侧注册模块的 SDL 图如图 5-9 所示。

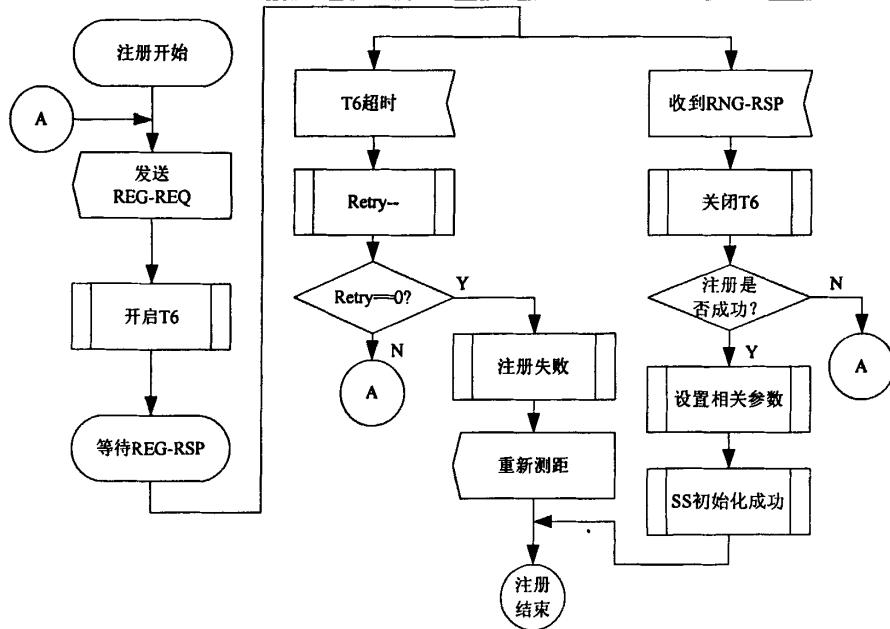


图 5-9 SS 侧注册模块的 SDL 图

在注册过程开始后，SS 首先向基站发送 REG-REQ 请求消息，然后开启定时器 T6 并将状态跳转到等待 REG-RSP 状态。

在 REG-RSP 状态下可能收到两种类型的消息，分别是定时器 T6 超时消息以及收到 RNG-RSP 消息。

如果收到定时器 T6 超时消息，说明发出的 REG-REQ 消息出现错误，那么注册次数 Retry 减一，判断 Retry 是否为 0，如果为 0，则注册失败，发送重新测距消息给测距模块，重新开始初始化接入过程，注册过程结束。

如果收到 REG-RSP 消息，则说明 SS 发出的 REG-REQ 消息得到 BS 的正确处理，此时关闭定时器 T6。查询响应消息，查看注册是否成功，如果不成功那么重新向 BS 发送 REG-REQ 消息，如果成功，那么设置相关参数，SS 初始化接入过程成功。

BS 侧注册模块的 SDL 图如图 5-10 所示。

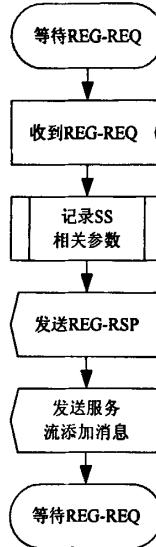


图 5-10 BS 侧注册模块的 SDL 图

BS 侧注册模块始终处于等待 REG-REQ 状态，接受来自各个 SS 的注册请求。如果收到来自某个 SS 的 REG-REQ 消息，那么注册模块首先查看 REG-REQ 消息内所携带的 SS 参数信息，并将该参数信息录入到全局变量 g\_ssinfo 中，然后发送 REG-RSP 消息给相应的 SS，最后向服务流管理模块发送服务流添加消息。

## 5.4 分类模块

分类模块的主要功能是将上层的业务数据与 MAC 层的业务流连接 CID 进行映射，并将映射后的业务数据交给数据服务模块。同时该模块还从数据服务模块接收数据并通过接口交给上层。

该模块维护了一个全局变量 g\_classifier，这个全局变量是一个链表结构的头指针，链表的每一个节点代表一个分类器。该链表结构如下所示：

```

typedef struct classifier_s{
    sf_t          sf_type;
    unsigned short cid;
    bool          default_or_not;
    struct classifier_s *next;
}classifier_t;

```

其中枚举类型变量 sf\_type 标识业务流类型，可能是 UGS、RTPS、NRTPS 或 BE；cid 标识一个业务流连接；default\_or\_not 标识该分类器是否是默认分类器。

每当收到上层的业务数据时，分类模块就会根据业务数据的服务流参数查询该链表，

找到对应该服务流的 CID 信息，如果找不到与服务流匹配的 CID，那么就将业务流映射到默认的 CID。

该链表结构由服务流管理模块负责维护，包括分类规则的修改、新建或删除。

## 5.5 数据服务模块

数据服务模块负责对 MAC 层向上下层传送的数据进行处理，包括添加、去除 MAC 帧头，以及数据的插入和分发工作。该模块处理的数据主要有 4 种类型，分别为来自分类模块的上层业务数据、来自管理模块相关的管理消息、来自数据传输调度模块的带宽请求消息以及来自接收模块的 MPDU。

数据服务模块的 SDL 图如图 5-11 所示。

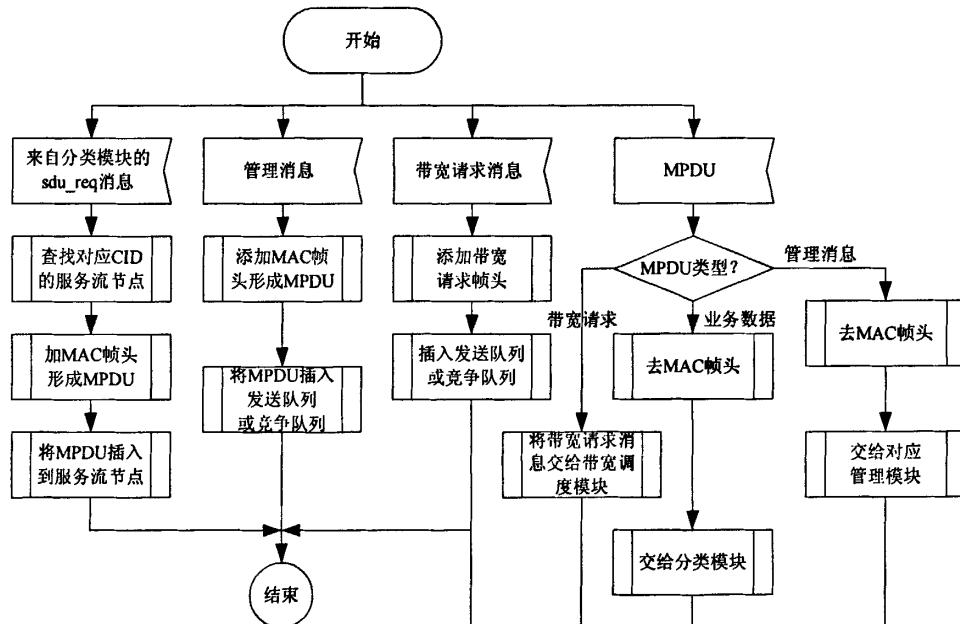


图 5-11 数据服务模块 SDL 图

对于来自分类模块的业务数据，首先根据业务分类得到的 CID 查询对应 CID 的服务流节点，然后为该业务数据添加 MAC 帧头，形成 MPDU，最后将该 MPDU 插入到查找到的服务流节点中去。

管理消息可能来自接入初始化和服务流管理等模块，在收到管理消息后，添加对应的 MAC 帧头，然后直接将 MPDU 插入到发送队列（如果是初始测距消息，则插入到竞争队列中去）。

带宽请求消息来自数据传输调度模块，只有 SS 才会产生这种消息。在收到这种消息后，添加对应的带宽请求帧头，如果是因 BS 轮询产生的带宽请求消息则直接插入发送队

列，否则插入到竞争队列。

MPDU 来自接收模块，在收到该消息后，首先查看消息类型，如果是业务数据，则去除帧头并交给分类模块；如果是管理消息，则去帧头并交给对应的管理模块；如果是 BS，则还可能接收到来自 SS 的带宽请求消息，那么直接将整个带宽请求消息交给带宽调度模块。

## 5.6 服务流管理模块

服务流管理是 IEEE802.16 MAC 层的核心模块之一，负责服务流的动态创建、修改和删除；负责维护分类模块的分类器，提供分类匹配规则的添加、修改和删除操作；同时该模块维护的服务流信息供数据传输调度模块和上行链路调度模块使用。

该模块维护了两个全局变量，分别为 `g_up_sfattr_header` 和 `g_down_sfattr_header`，其中 `g_up_sfattr_header` 是上行服务流链表的头指针，`g_down_sfattr_header` 是下行服务流链表的头指针，链表节点的数据结构为：

```
typedef struct sfattr_s{
    unsigned long      sfid;
    unsigned short     sf_state;
    unsigned short     bwr;
    unsigned short     sf_direction;
    unsigned short     cid;
    unsigned short     service_flow_scheduling_type;
    unsigned short     priority;
    ...
    unsigned short     pdu_size;
    wimax_pdu_t        *pdu;
    struct ssinfo_s   *whoami;
    struct sfattr_s   *global_sfattr_next;
}sfattr_t;
```

其中，`sfid` 是一个 32 位的无符号整数，标识一个服务流；`sf_state` 标识服务流的状态；`bwr` 标识该服务流上的带宽申请量；`sf_direction` 标识服务流的方向；`cid` 标识对应该服务流的连接信息；`service_flow_scheduling_type` 标识服务流类型，可能是 UGS、RTPS、NRTPS 或 BE；`priority` 标识服务流的优先级；`pdu_size` 标识了当前服务流连接上承载的数据量；`pdu` 标识了该服务流上承载的数据单元；`whoami` 标识了该服务流所属的 SS。

服务流管理模块分为服务流建立模块、服务流修改模块和服务流删除子模块。

### 5.6.1 服务流建立模块

在 SS 接入初始化过程结束后，SS 或 BS 可以选择发起服务流的建立过程，其中服务流建立的发起方称为本地，响应方称为远端，服务流修改和删除与此类似。服务流的建立通过三次握手的方式完成。

服务流建立模块本地端的 SDL 图如 5-12 所示。

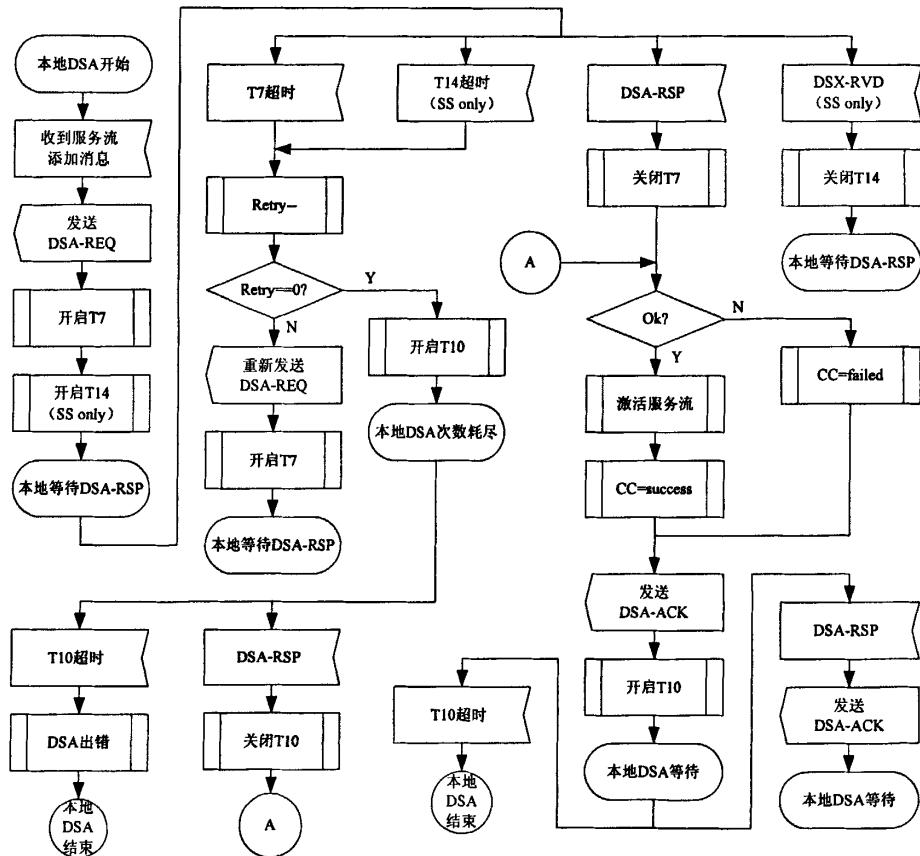


图 5-12 服务流建立模块本地端的 SDL 图

本地端在收到添加服务流指示后，查找对应 SS 的服务流表，生成 DSA-REQ 消息并发送出去，开启定时器 T7，如果本地是 SS，则还要开启定时器 T14，然后状态跳转到本地等待 DSA-RSP。

在本地等待 DSA-RSP 状态下，可能收到 4 类消息，分别为定时器 T7 超时、定时器 T14 超时、DSA-RSP 消息和 DSX-RVD 消息。

如果收到定时器 T7 超时消息，说明本地发出的 DSA-REQ 请求消息在发送过程中出错，此时 Retry 减 1，如果 Retry 等于 0，那么开启定时器 T10，状态跳转到本地 DSA 次数

耗尽状态；如果不为 0，则重新发送 DSA-REQ 消息，开启定时器 T7，状态跳转到本地等待 DSA-RSP 状态。

如果本地端是 SS，则还会收到 T14 超时消息，对该消息处理与定时器 T7 超时消息相同。

DSX -RVD 消息是一个 SS 侧特有的临时消息，说明远端 BS 正在处理 DSA-REQ 消息，此时关闭定时器 T14，状态跳转到本地等待 DSA-RSP 状态。

如果收到 DSA-RSP 消息，说明本地发出的 DSA-REQ 消息在远端得到了处理，此时首先关闭定时器 T7，查看响应消息，查询服务流建立是否成功，如果成功则使能服务流，设置确认信息为 success，如果不成功则设为 failed，然后发送 DSA-ACK 消息给远端。开启定时器 T10，跳转状态到本地 DSA 等待状态。

在本地 DSA 次数耗尽状态下可能收到两种消息，分别为 T10 超时和 DSA-RSP 消息。如果收到 T10 超时消息，则本次服务流建立失败，如果收到 DSA-RSP 消息，则关闭 T10，随后的消息处理同在本地等待 DSA-RSP 状态下收到 DSA-RSP 消息的处理。

本地 DSA 等待状态下可能收到两种消息，分别为 T10 超时和 DSA-RSP 消息，如果收到 T10 超时消息，则本次服务流建立成功结束。如果收到 DSA-RSP 消息，则向远端发送 DSA-ACK 消息，状态跳转到本地 DSA 等待状态。

服务流建立模块远端的 SDL 图如 5-13 所示。

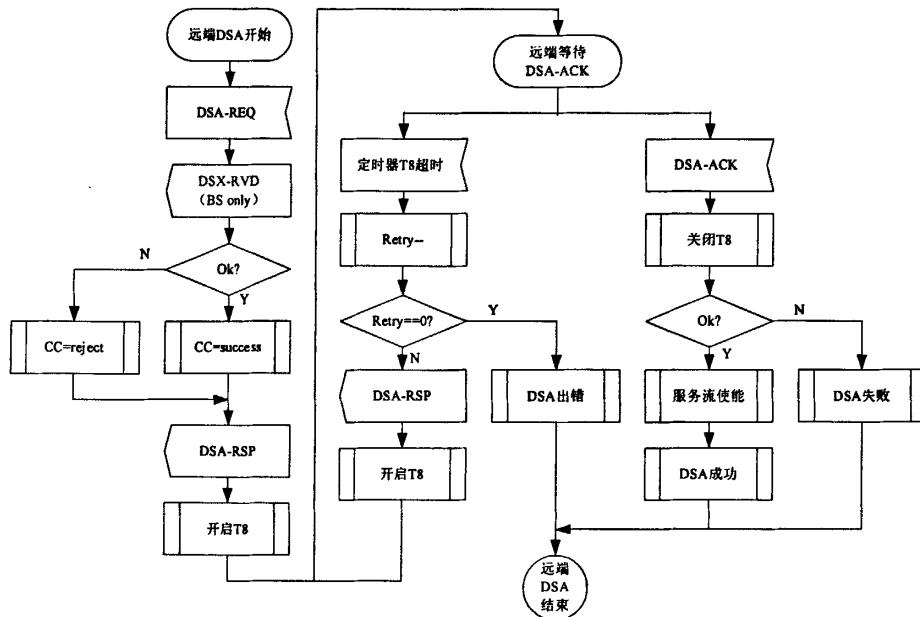


图 5-13 服务流建立模块远端的 SDL 图

远端服务流建立模块在收到 DSA-REQ 后，如果远端是 BS，则首先向本地端发送

DSX-RVD 消息，然后查看 DSA-REQ 消息，判断是否建立成功，如果成功则置确认状态为 success，如果不成功则置确认状态为 reject，生成 DSA-RSP 消息并发送给本地端，然后开启定时器 T8 并将状态跳转到远端等待 DSA-ACK 状态。

在远端等待 DSA-ACK 状态可能收到两类消息，分别为定时器 T8 超时和 DSA-ACK 消息。

如果收到定时器 T8 超时消息，则 Retry 减 1，判断是否为 0，如果为 0，则 DSA 出错，远端 DSA 过程结束，如果不为 0，则重新发送 DSA-RSP 消息，开启定时器 T8 并将状态跳转到远端等待 DSA-ACK 状态。

如果收到 DSA-ACK 消息，则首先关闭定时器 T8，判断建立是否成功，如果不成功则此次 DSA 过程失败并结束该过程，如果成功，则使能刚刚建立的服务流，服务流建立过程成功结束。

以上对服务流的建立过程作了详细的阐述，需要说明的是：服务流的建立会影响分类模块的分类匹配规则，因此如果 SS 建立了上、下行服务流，对应的 SS、BS 侧的分类匹配规则要作相应的更新。

### 5.6.2 服务流修改模块

服务流可以通过 DSC 消息来动态的修改。服务流的修改可以由 BS 发起，也可以由 SS 发起，其过程类似于服务流动态建立的三次握手过程。

服务流修改模块的本地端 SDL 图如图 5-14 所示。

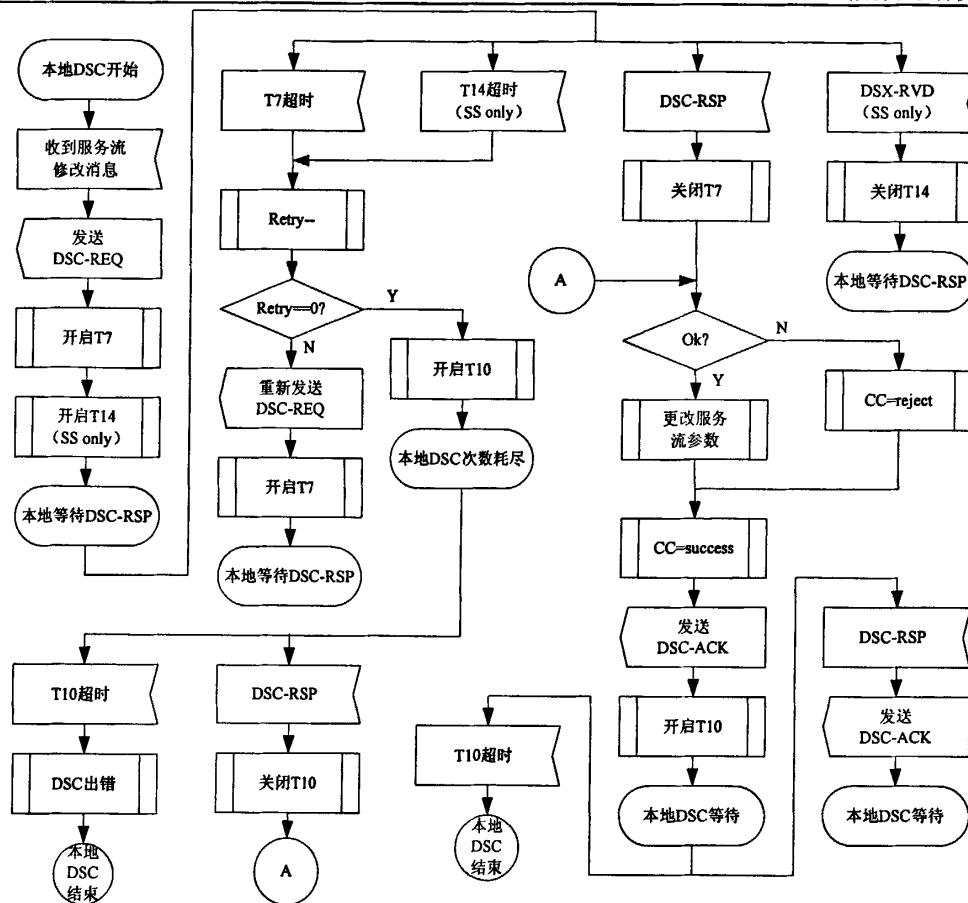


图 5-14 服务流修改模块的本地端 SDL 图

本地端在收到服务流修改指示后，生成 DSC-REQ 消息并发送出去，开启定时器 T7，如果本地是 SS，则还要开启定时器 T14，然后状态跳转到本地等待 DSC-RSP。

在本地等待 DSC-RSP 状态下，可能收到 4 类消息，分别为定时器 T7 超时、定时器 T14 超时、DSC-RSP 消息和 DSX-RVD 消息。

如果收到定时器 T7 超时消息，说明本地发出的 DSC-REQ 请求消息在发送过程中出错，此时 Retry 减 1，如果 Retry 等于 0，那么开启定时器 T10，状态跳转到本地 DSC 次数耗尽状态，如果不为 0，则重新发送 DSC-REQ 消息，开启定时器 T7，状态跳转到本地等待 DSC-RSP 状态。

如果本地端是 SS，则还会收到 T14 超时消息，对该消息处理与定时器 T7 超时消息相同。

DSX-RVD 消息是一个 SS 侧特有的临时消息,说明远端 BS 正在处理 DSC-REQ 消息,此时关闭定时器 T14,状态跳转到本地等待 DSC-RSP 状态。

如果收到 DSC-RSP 消息，说明本地发出的 DSC-REQ 消息在远端得到了处理，此时

首先关闭定时器 T7，查看响应消息，判断服务流参数的改变是否可以接受，如果可以则使能服务流，设置确认信息为 success，如果不可以则设为 reject，然后发送 DSC-ACK 消息给远端。开启定时器 T10，跳转状态到本地 DSC 等待状态。

在本地 DSC 次数耗尽状态下可能收到两种消息，分别为 T10 超时和 DSC-RSP 消息。如果收到 T10 超时消息，则本次服务流建立失败，如果收到 DSC-RSP 消息，则关闭 T10，随后的消息处理同在本地等待 DSC-RSP 状态下收到 DSC-RSP 消息的处理。

本地 DSC 等待状态下可能收到两种消息，分别为 T10 超时和 DSC-RSP 消息，如果收到 T10 超时消息，则本次服务流建立成功结束。如果收到 DSC-RSP 消息，则向远端发送 DSC-ACK 消息，状态跳转到本地 DSC 等待状态。

服务流修改模块远端的 SDL 图如 5-15 所示。

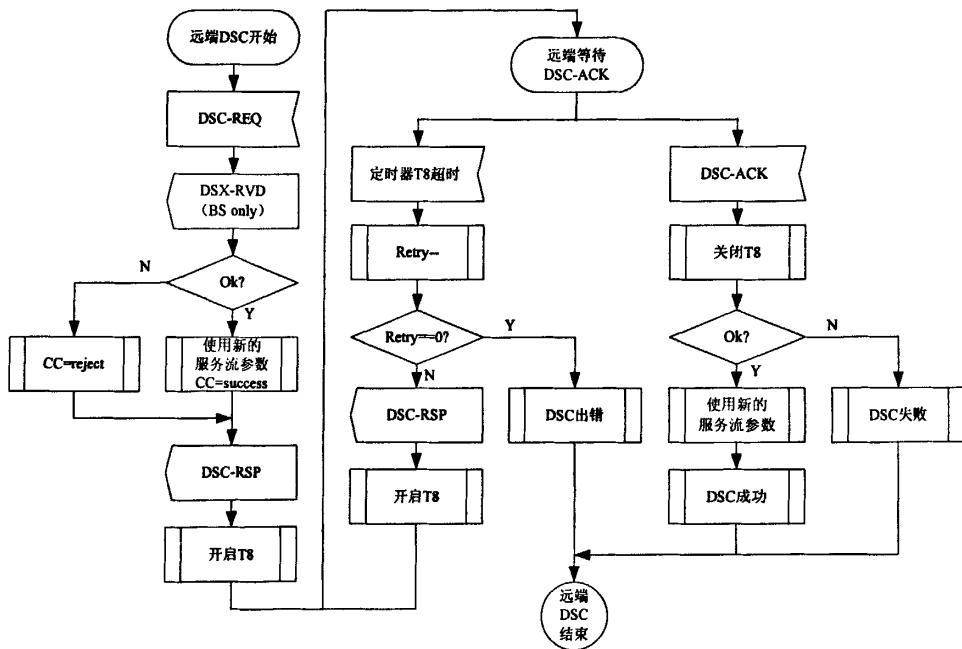


图 5-15 服务流修改模块远端的 SDL 图

远端服务流建立模块在收到 DSC-REQ 后，如果远端是 BS，则首先向本地端发送 DSX-RVD 消息，然后查看 DSC-REQ 消息，判断对服务流的修改是否可以接受，如果可以则置确认状态为 success，如果不可以则置确认状态为 reject，生成 DSC-RSP 消息并发送给本地端，然后开启定时器 T8 并将状态跳转到远端等待 DSC-ACK 状态。

在远端等待 DSC-ACK 状态可能收到两类消息，分别为定时器 T8 超时和 DSC-ACK 消息。

如果收到定时器 T8 超时消息，则 Retry 减 1，判断是否为 0，如果为 0，则 DSC 错误，远端 DSC 过程结束，如果不为 0，则重新发送 DSC-RSP 消息，开启定时器 T8 并将状态跳

转到远端等待 DSC-ACK 状态。

如果收到 DSC-ACK 消息，则首先关闭定时器 T8，判断修改是否成功，如果不成功则此次 DSC 过程失败并结束该过程，如果成功，则使能修改过的服务流，服务流修改过程成功结束。

### 5.6.3 服务流删除模块

服务流可以通过 DSD 消息来删除，当服务流删除时，所有的相关资源都会被释放。与服务流建立与修改一样，服务流的动态删除也可以由 SS 发起也可以由 BS 发起。

服务流删除模块的本地端 SDL 图如图 5-16 所示。

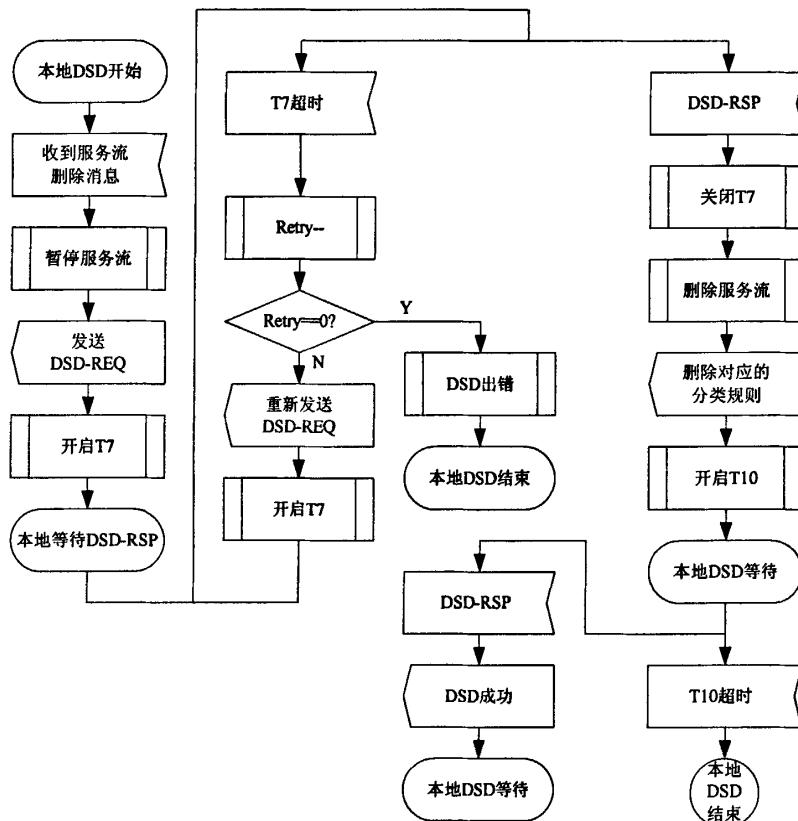


图 5-16 服务流删除模块的本地端 SDL 图

在收到服务流删除指示之后，本地端服务流模块首先暂停该服务流，生成 DSD-REQ 消息并发送给相应的远端，然后开启 T7 并将状态置为本地等待 DSD-RSP。

在本地等待 DSD-RSP 状态下，可能收到两种消息，分别为定时器 T7 超时和 DSD-RSP 消息。

如果收到定时器 T7 超时消息，说明本地发出的 DSD-REQ 请求消息在发送过程中出错，此时 Retry 减 1，如果 Retry 等于 0，那么 DSD 过程出错，本地 DSD 过程结束，如果

不为 0，则重新发送 DSD-REQ 消息，开启定时器 T7，状态跳转到本地等待 DSD-RSP 状态。

如果收到 DSD-RSP 消息，首先关闭定时器 T7，然后删除服务流，并且删除对应的分类规则，开启定时器 T10，状态跳转到本地 DSD 等待状态。

在本地 DSD 等待状态下，可能收到两种消息，分别为定时器 T10 超时消息和 DSD-RSP 消息。如果是定时器 T10 超时，那么本地 DSD 过程结束。如果收到 DSD-RSP 消息，则状态跳转到本地 DSD 等待状态。

服务流删除模块的远端 SDL 图如图 5-17 所示。

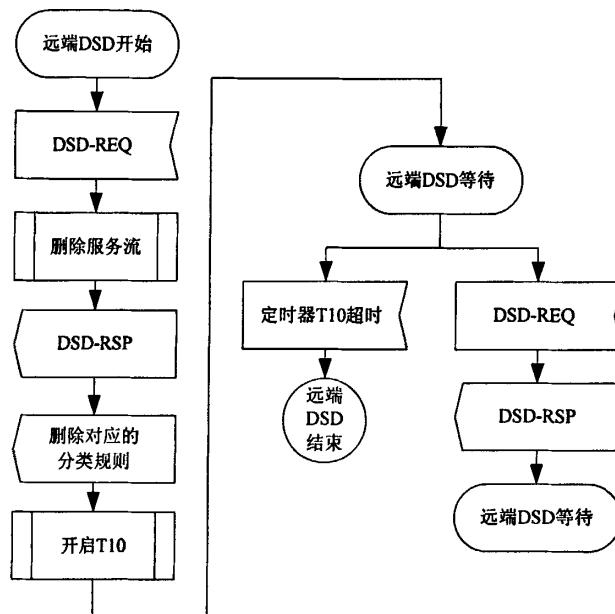


图 5-17 服务流删除模块的远端 SDL 图

服务流删除模块的远端在收到 DSD-REQ 消息后，首先删除对应的服务流，生成 DSD-RSP 消息并发送出去，然后删除对应该服务流的分类规则，开启定时器 T10 并将状态跳转到远端 DSD 等待状态。

在远端 DSD 等待状态下，可能会收到两种消息，分别为定时器 T10 超时和 DSD-REQ 消息。如果收到定时器 T10 超时消息，那么远端 DSD 过程结束，服务流成功删除。如果收到 DSD-REQ 消息，则重新发送 DSD-RSP 消息，并保持状态不变直至收到定时器 T10 超时消息。

## 5.7 数据传输调度模块

数据传输调度模块负责对不同服务流连接上承载的数据单元进行调度，保证不同优先级的业务得到合理的调度。在具体实现中，针对 4 种不同的业务流类型，对应建立了 4 个

数据先入先出队列 (FIFO)，优先级由高到低分别为 UGS 队列、RTPS 队列、NRTPS 队列和 BE 队列。数据传输调度模块根据数据队列的不同优先级，以及当前的可用带宽决定是否发送对应的协议数据单元。BS 侧和 SS 侧的数据传输调度模块基本相同，图 5-18 是数据传输调度模块的 SDL 图。

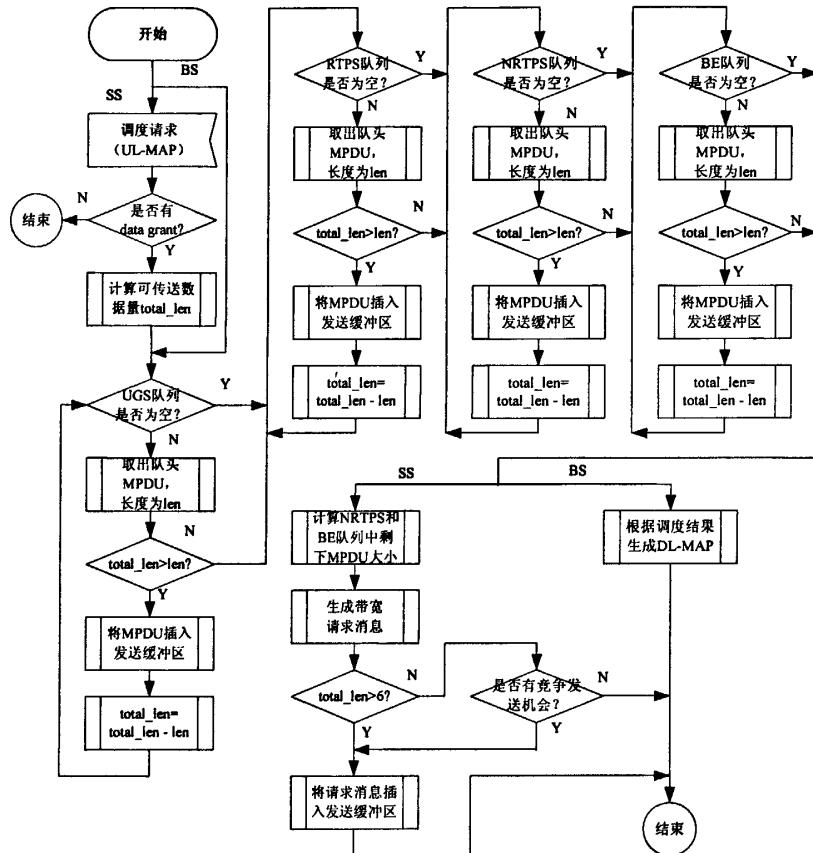


图 5-18 数据传输调度模块的 SDL 图

如图所示，调度开始后，如果是 SS，则首先检查 BS 发送的 UL-MAP，该消息中存储了所有 SS 的带宽分配情况，若 BS 为当前 SS 分配了 data grant，则计算对应的带宽大小 total\_len，如果没有分配 data grant，则当前调度结束；如果是 BS，则计算下行子帧中可传输的总数据量 total\_len。

在得到 total\_len 之后，根据队列优先级，首先检查 UGS 队列，如果为空则继续查看 RTPS 队列，不为空则取出队头的 MPDU，比较当前总的可用带宽和 MPDU 数据量的大小，如果可用带宽比 MPDU 大，则把该 MPDU 插入发送缓冲区并更新 total\_len，如果可用带宽比 MPDU 小，则开始处理 RTPS 队列。如此循环往复，直至 UGS 队列为空，或者 total\_len 比 MPDU 小。

如同处理 UGS 队列一样分别处理 RTPS、NRTPS 和 BE 队列，当处理完所有队列之后，

如果是 SS，则根据 NRTPS 或 BE 队列中队头 MPDU 的数据量，生成对应的带宽请求消息，此时判断 total\_len 的剩余大小是否比带宽请求消息的长度大，如果大则将带宽请求消息插入到发送缓冲区中，如果小，则查看当前是否有带宽请求竞争机会，如果有则将带宽请求消息发送到发送缓冲区中，并在竞争时刻到的时候把请求消息发出去；如果是 BS，则根据对 4 种队列的调度结果，生成对应的 DL-MAP 消息。

## 5.8 上行链路调度模块

### 5.8.1 模块概述

如前所述，在 IEEE802.16 MAC 层协议中，为了保证不同业务类型的服务质量，共定义了 4 种不同的业务流类型，优先级从高到低分别为 UGS、RTPS、NRTPS 和 BE。有了这些业务类型，就可以根据业务需要提供实时和非实时的不同传输速率的服务。

上行链路调度模块负责对各个 SS 的不同的服务流类型进行调度，基于 SS 所拥有的服务流类型，为各个 SS 分配传输带宽，最后根据分配结果，生成上行链路带宽映射消息（UL-MAP）。上行链路调度模块的总体结构如图 5-19 所示。

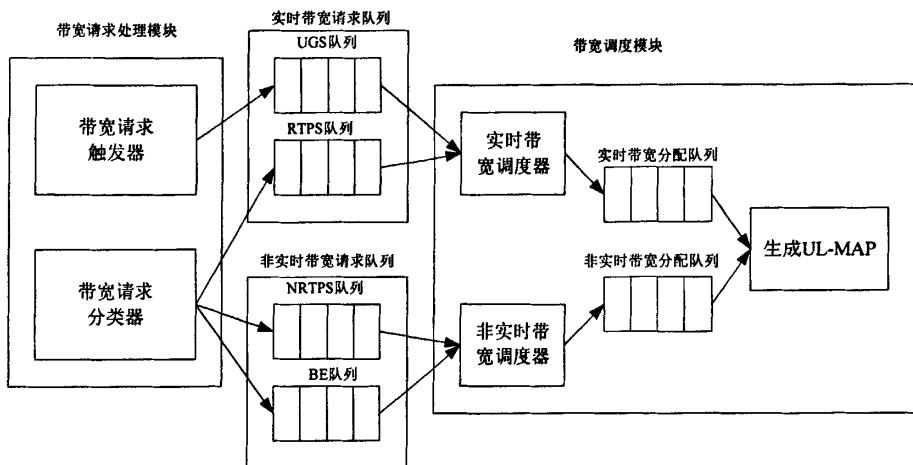


图 5-19 上行链路调度模块的总体结构

如图所示，上行链路调度模块总体结构分为 3 个部分，分别为带宽请求处理模块、带宽请求队列以及带宽调度模块。

其中带宽请求队列存储了各个 SS 的各个服务流的带宽请求大小，根据不同的业务流类型，带宽请求队列分为 4 种，分别为 UGS 队列、RTPS 队列、NRTPS 队列和 BE 队列，其中 UGS 和 RTPS 队列是实时带宽请求队列，NRTPS 和 BE 队列是非实时带宽请求队列。

带宽请求处理模块负责产生对应服务流的带宽请求，并且根据接收到的带宽请求消息

将带宽请求插入到相应的队列。该模块包括一个带宽请求触发器和一个带宽请求分类器。

带宽调度模块负责对 4 种带宽请求队列进行调度，分别形成实时带宽分配队列和非实时带宽分配队列，最后根据带宽分配结果，生成带宽分配映射消息 UL-MAP。

## 5.8.2 带宽请求处理模块

带宽请求处理模块包括一个带宽请求触发器和一个带宽请求分类器，带宽请求触发器根据服务流参数分别产生 UGS 带宽请求、RTPS 和 NRTPS 单播轮询带宽请求。带宽请求分类器接收来自除 UGS 之外的不同类型业务流对上行链路带宽的请求，带宽请求存入对应的带宽请求队列中接受调度。

UGS 业务主要用于实时的、周期性的、固定大小的数据包的业务流，像 VoIP 业务，这种业务的主要 QoS 参数为：数据传输速率、周期、时延和抖动。由于 UGS 业务是基站直接授予带宽，不需要通过发送带宽请求的方式来通知基站要求分配带宽。在实现过程中，根据 UGS 业务的 QoS 参数，计算 UGS 服务流带宽分配周期，以此周期作为 UGS 定时器定时周期，每当 UGS 定时器周期到，就将对应的带宽请求存入 UGS 带宽请求队列中，同时开始对该带宽请求的等待时间进行计时。

RTPS 和 NRTPS 单播轮询带宽请求是为了给变比特率的 RTPS 和 NRTPS 业务提供实时性的发送带宽请求的机会。根据 RTPS 和 NRTPS 服务流参数计算出对应的分配周期，以此周期作为定时器的定时周期，由于单播带宽请求所占用的带宽资源很少，所以每当 RTPS 和 NRTPS 定时周期到，就将对应的带宽请求存入优先级最高的 UGS 带宽请求队列。

带宽请求分类器接收来自除 UGS 之外不同业务流类型对上行链路带宽的请求，包括实时业务和非实时数据业务类型的带宽请求，然后将带宽请求转换成为与调度器匹配的带宽请求，同时将转换后的带宽请求存入对应的带宽请求队列中，对该带宽请求等待时间进行计时。

## 5.8.3 带宽调度模块

带宽调度模块对 4 种不同类型的带宽请求队列中的带宽请求计算优先级，并形成实时带宽分配队列和非实时带宽分配队列。最后根据优先级顺序为各带宽请求分配带宽，并形成 UL-MAP 消息。

带宽调度模块包括一个实时带宽调度器，一个非实时带宽调度器，一个实时带宽分配队列和一个非实时带宽调度队列，以及一个 UL-MAP 生成模块。

其中实时带宽调度器分别计算 UGS 和 RTPS 带宽请求队列中带宽请求的优先级，然

后按顺序将带宽请求放到实时带宽分配队列中。非实时带宽调度器分别计算 NRTPS 和 BE 带宽请求队列中带宽请求的优先级，并按优先级顺序将带宽请求放到非实时带宽分配队列中。

带宽调度流程如图 5-20 所示。

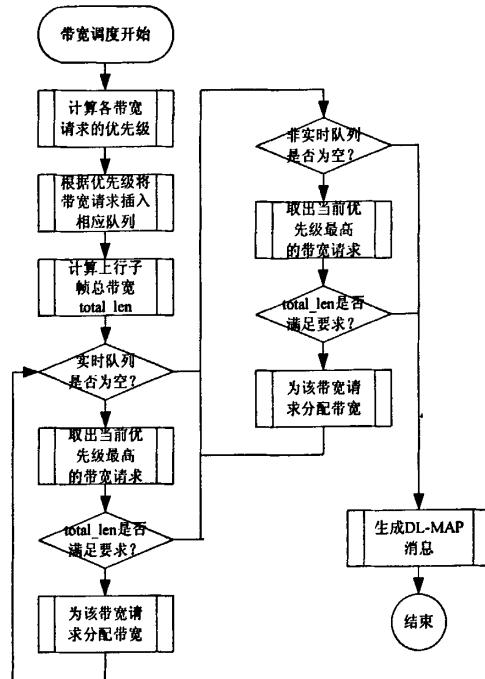


图 5-20 带宽调度流程

如图所示，在带宽调度开始后，实时调度器和非实时调度器分别计算各请求队列中带宽请求的优先级，并插入相应的带宽分配队列。然后计算出当前上行子帧的总带宽大小，之后首先遍历实时带宽分配队列，再遍历非实时带宽分配队列，为各带宽请求分配带宽，直至总带宽分配完毕，最后根据带宽分配的结果生成 UL-MAP 消息。

## 5.9 收发模块

### 5.9.1 发送模块

发送模块将 MPDU 从 MAC 层发送到物理层。在 SS 侧，当 SS 发送时隙到的时候，MAC 层将存储在发送缓冲区中的数据逐字节的向物理层发送，在发送过程中，计算 CRC 值并把 CRC 结果添加到 MPDU 的末尾。在 BS 侧，当下行子帧开始时，根据 DL-MAP 中下行服务流映射信息，将发送缓冲区的数据及 CRC 校验值发送到物理层。

## 5.9.2 接收模块

接收模块接收来自物理层的字节流，并做定界和数据校验。在 SS 侧，SS 首先接收并解析 DL-MAP 消息，在 DL-MAP 消息中存储了 BS 发送给各 SS 的下行数据信息以及对应的时隙，根据时隙信息 SS 接收到属于自己的 MPDU，然后进行定界并计算出每一个 MPDU 的 CRC 值，判断数据是否出错，最后根据 MPDU 的类型将数据分发给各个模块。

BS 在上行子帧中接收来自各个 SS 的数据，根据 UL-MAP 信息为每个 MPDU 进行定界，分别为各 MPDU 进行 CRC 校验。最后根据 MPDU 的类型将数据分发到各个模块。

## 5.10 MAC 层协议各模块联合调试

在各模块代码编写完成之后，首先进行单元测试，根据设计需求编写相应的单元测试案例，并进行相应的测试。在经过单元测试后，将多个模块联合起来进行系统集成测试。

根据项目实际情况，在测试平台方面，我们将各硬件目标板通过集线器相互联在一起，以此来代替无线通信方式。共有 3 台硬件目标板参与测试，其中一台作为基站，另两台作为站点，在测试过程中，我们在任一站点侧随机发送一组数据，在相应的目标站点侧都能够正确接收到。

虽然模拟的测试环境和实际工作环境差别较大，但是也能够初步验证各模块功能的正确性。接下来，我们将在此基础上，完成 IEEE802.11 和 IEEE802.16 MAC 层协议的重配置。

## 结束语

宽带无线接入是未来通信网络极其重要的接入方式，尤其是基于 IEEE802.16 的无线城域网，由于其带宽高，覆盖范围广，有 QoS 保证，已经成为最为重要的无线接入方式之一，并且在未来可重配置终端中也将扮演重要的角色。

本论文的目的就是深入研究 IEEE802.16 MAC 层协议，并在 ARM、DSP 和 FPGA 的协作环境下，完成 IEEE802.16 MAC 层协议软件的设计和实现，为项目后续实现 WiFi 与 WiMAX 协议栈重配置奠定了基础。

本文作者主要完成以下工作：

- 研究了 IEEE802.16 MAC 层协议。系统分析了 MAC 层汇聚子层和公共部分子层的协议功能。
- 和项目组成员协同，给出了基于 DSP 的 IEEE802.16 MAC 层实现的系统硬件设计方案。设计并编写了 MAC 层与其上下层的接口，通过测试，该设计方案满足设计需求。
- 实现了基于 DSP 的 IEEE802.16 MAC 层协议软件设计。根据协议功能划分了协议功能模块，设计了协议软件调度器，针对每个功能模块，给出了对应的设计方案并编码实现，测试结果说明了协议软件设计和实现的正确性。

本文存在的不足和需要进一步完成的工作：

- 本文实现的 MAC 层与上层的接口还比较简单，只能实现简单的数据传送，不能支持 TCP/IP 协议栈。
- 本文实现的 MAC 层协议功能并不完整，包括 ARQ 和安全子层在内的协议功能没有实现。
- 本文实现的数据传输调度和上行链路调度的算法还需要做进一步的改进。
- 本文实现的 MAC 层协议软件全部由 C 语言写成，执行效率不够高，涉及数据处理部分的协议功能可以改用汇编语言，可进一步提高软件运行速度。

## 致 谢

论文工作即将结束，借此机会，我衷心地感谢所有曾经给予我无私关怀和帮助的老师和同学们。

首先我要感谢我的导师糜正琨教授在这三年研究生阶段给我的巨大帮助和教诲，从生活上的关怀、学习上的教导，到学术科研工作中的指导，点点滴滴。糜老师严谨的治学态度、不懈的钻研精神、和蔼的指导风格深深地影响着我。这将成为使我终身受益的宝贵人生财富，并激励我今后踏实工作、努力学习不止。

感谢项目组的朱军师兄、黄志海、周恒辉、庄镒鹏、柯灿强、朱健以及其他同学。在项目过程中，他们提出了很好的意见和建议，开拓了我的思路，给了我很大的启发和帮助。和他们在一起的日子，我感到很愉快。

我要特别感谢多年来父母的支持和鼓励，没有他们不离不弃的支持，就没有我的今天。最后，还要感谢参加我的论文答辩的专家和老师，以及其他帮助过我的老师和同学。

## 参考文献

- [1] IEEE Std 802.16-2001 IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems
- [2] IEEE802.16c-2002 Standard for Local and metropolitan area networks - Part 16: Air Interface for Fixed Broadband Wireless Access Systems-Amendment 1: Detailed System Profiles for 10-66 GHz
- [3] IEEE Std 802.16aTM-2003(Amendment to IEEE Std 802.16TM-2001) IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems-Amendment 2: Medium Access Control Modifications and Additional Physical Layer Specifications for 2-11 GHz
- [4] IEEE Std 802.16TM-2004(Revision of IEEE Std 802.16-2001) IEEE Standard for Local and metropolitan area networks Part16:Air Interface for Fixed Broadband Wireless Access Systems
- [5] IEEE 802.16e -2005 Standard for Local and metropolitan area networks Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands and Corrigendum 1
- [6] IEEE 802.16f-2005 Standard for Local and metropolitan area networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems- Amendment 1: Management Information Base
- [7] IEEE802.16g-2007 Standards for Local and metropolitan area networks - Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems - Amendment 3: Management Plane Procedure and Services
- [8] 牛晓敏.宽带无线接入技术— WiMAX.电信技术, 2005 年 01 期:59-61
- [9] 汪坤, 李巍, 李莉莉.宽带无线接入— WiMAX.电信网技术, 2005 年 03 期: 59-62
- [10] 雷震洲.IEEE802.16 和 WiMAX.现代电信科技, 2004 年 07 期:2-7
- [11] 赵利,莫金旺,李晓记.WiMAX 宽带无线接入的特点及其应用模式.中国数据通信,2005 年 04 期:64-68
- [12] 陈卓, 余重秀, 徐大雄等.基于 WiMAX 技术的宽带 WMAN 及其应用.电信技术, 2005 年 01 期:62-64
- [13] 王茜, 王岩.无线城域网 WiMAX 技术及其应用.电信科学, 2004 年 08 期:27-30
- [14] Samsung Electronics Co., Ltd.“S3C2410X 32-Bit RISC Microprocessor User's Manual, Revision 1.2”,2003

- [15] 清源科技编著：《TMS320C54x DSP 应用程序设计教程》，机械工业出版社，2004年1月
- [16] EDA 先锋工作室，吴继华，王诚编著：《Altera FPGA/CPLD 设计》，人民邮电出版社，2007 年 6 月
- [17] Intel Corporation,“LXT972A 3.3V Dual-Speed Fast Ethernet Transceiver Datasheet”, January 2001
- [18] Kitti Wongthavarawat, Aura Ganz. Packet Scheduling for QoS Support in IEEE 802.16 Broadband Wireless Access Systems. International Journal of Communications Systems, 2003(16):81-96
- [19] Guo Songchu, Deng Wang, Shunliang Mei. A QoS Architecture for the MAC protocol of IEEE 802.16 BWA System[J]. IEEE 2002 International Conference,2002,1(29):435-439
- [20] 王洪熙，陈剑峰，焦文华. 一种用于 IEEE802.16 无线城域网 TDD 模式中的带宽调度方法. 电子与信息学报，2006, 28(5):789-794
- [21] HANNIKAINEN M, KNUUTILA J, HAMALAINEN T,et al. Using SDL for implementing a wireless medium access control protocol[A], 2000 Proc of Int Symp on Multimedia Software Engineering.Taipei,2000.229 -236
- [22] Govindan Nair, Joey Chou, Tomasz Madejski,etc.IEEE 802.16 Medium Access Control and Service Provisioning. Intel Technology Journal,Volume 8,Issue3,2004

## 攻读硕士学位期间发表的论文及科研成果

学术论文：

王成法，糜正琨，无线终端协议栈重配置技术研究，第 23 届南京地区研究生通信年会，  
2008

专利申请：

无线媒体接入控制层协议重配置方法，专利申请号：200810124156.5

软件著作权：

参数可重配置 802.16MAC 层软件 PaRMAC16(V1.0) ，登记号：2008SR38736

参加的项目：

下一代泛在通信网络端到端重配置关键技术研究，江苏省高科技计划项目高技术研究（工业部份）(BG2006039)