

[51] Int. Cl⁷

H04L 12/24

H04L 29/06



[12] 发明专利申请公开说明书

[21] 申请号 02808491.8

[43] 公开日 2004 年 6 月 9 日

[11] 公开号 CN 1504029A

[22] 申请日 2002.4.16 [21] 申请号 02808491.8

〔30〕 優先権

[32] 2001. 4.20 [33] US [31] 60/285,188

[32] 2001. 4.20 [33] US [31] 60/285,153

[86] 国际申请 PCT/US2002/012162 2002.4.16

[87] 国际公布 W002/087157 英 2002.10.31

「85」 进入国家阶段日期 2003.10.20

[71] 申请人 通用仪表公司

地址 美国宾夕法尼亚州

[72] 发明人 布拉尼斯拉夫·N·米德吉拉

杰夫·E·休伊特 蔡樱樱

[74] 专利代理机构 中原信达知识产权代理有限责
任公司

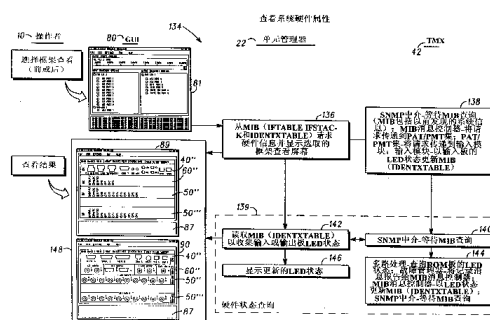
代理人 张天舒 谢丽娜

权利要求书 4 页 说明书 60 页 附图 12 页

[54] 发明名称 多路传输复用器的图形用户界面

[57] 摘要

本发明公开一种多功能的多路传输复用器以及多种相关的方法、系统、子系统、软件功能、图形用户界面和控制系统。 所公开的内容包括 GUI，其使得操作员能够轻易地实时监控并使用流过多路传输复用器的内容流。 所公开的界面包括多个屏幕，其能够向操作员提供系统硬件和软件的标识、结构、配置、带宽使用和/或状态信息。 所公开的功能还提供了通过简单操作(诸如图形目标的拖放操作)为现在和将来的路由事件进行的计算机辅助路由配置。 通过允许操作员同时配置单一内容流部件和这样的部件组，从而进一步简化了路由控制。



LSSN 1008-4274

1. 一种用于多路传输复用器中的图形用户界面，所述多路传输复用器具有多个输入端口和多个输出端口，其中在所述输入端口接收动态内容流，并将其发送到所述输出端口，所述内容流包括多个部件并具有包括码率、类型和名称在内的属性，所述图形用户界面包括：

目录结构查看屏幕，用于显示表示所述输入端口和所述输出端口的图形对象，并用于显示具有指示各自输入和输出内容流的层次结构的多个分支的输入和输出目录结构，所述目录结构查看屏幕指示了在所述多路传输复用器接收并通过所述多路传输复用器发送的所述内容流，并且实时显示在所述多路传输复用器接收的所述内容流中的变化。

2. 如权利要求 1 所述的图形用户界面，其中至少某些图形对象是图标，对图标进行选择将显示传输编辑器窗口，可以在该窗口中修改所述内容流属性。

3. 如权利要求 2 所述的图形用户界面，其中，在改变至少一个内容流属性之后，将会实时地重新计算并重新显示输出带宽使用。

4. 如权利要求 1 所述的图形用户界面，其中，所述目录结构查看屏幕进一步包括滚动记录消息窗口，用于实时地显示系统所产生的记录消息。

5. 如权利要求 4 所述的图形用户界面，其中，所述将要显示的记录消息将根据多个预设的过滤级别而被过滤。

6. 如权利要求 1 所述的图形用户界面，其中，所述目录结构进一步包括指示由所述输入端口接收并退出输出端口的内容流的类型的图形对象。

7. 一种用于多路传输复用器的图形用户界面，所述图形用户界面的框架具有多个硬件和软件部件，从而将由输入端口接收的输入内容流发送到预设的输出端口，所述图形用户界面包括：

5 框架查看屏幕，用于显示表示安装在所述多路传输复用器的框架中的所述硬件部件的物理结构的图形对象，所述图形对象指示了由所述硬件部件提供的输入和输出端口的类型。

8. 如权利要求 7 所述的图形用户界面，其中所述框架查看屏幕还实时显示所显示的硬件部件的操作状态。

9. 如权利要求 7 所述的图形用户界面，其中所述图形用户界面进一步包括软件属性屏幕，用于显示在所述多路传输复用器中使用的软件的属性，所述软件属性包括软件版本号。

10. 如权利要求 7 所述的图形用户界面，其中所述界面能够实时图形显示在所述硬件状态数据中检测到的变化。

11. 一种使用图形用户界面通过多路传输复用器来配置内容流的路由控制的方法，所述路由包括多个输入端口和多个输出端口，所述方法包括：

显示表示所述输入端口的图标；

显示表示所述输出端口的图标；

25 显示表示所述输入内容流的图标，所述流是由所述输入图标之一接收的，所述输入流图标在所述输入端口图标附近显示；

将所述输入流图标拖动到所述输出端口图标附近；

将所述输入内容流图标放置到所述输出端口图标附近；和

在进行完放置操作之后，显示表示退出所述输出端口的输出流的图标，所述图标实际上显示在放置所述输入流的地方。

30

12. 如权利要求 11 所述的方法，其中所述输出流图标表示目前退出所述输出端口的输出流。

5 13. 如权利要求 11 所述的方法，其中所述输出流图标表示将在未来的一个预设时刻开始退出所述输出端口的输出流。

14. 如权利要求 11 所述的方法，其中：
所述输入内容流包括多个具有属性的层次部件，
所述输入流图标表示所有的多个层次部件，和
10 所述输出流图标表示具有与所述输入流相同的多个层次部件的内容流。

15. 如权利要求 11 所述的方法，其进一步包括：
在所述输入流图标附近显示字母数字字符串；
15 拖动所述字母数字字符串和所述输入流图标；
放置所述字母数字字符串和所述输入流图标；和
在所述输出流图标附近显示字母数字字符串。

16. 一种用于多路传输复用器的图形用户界面，所述多路传输复用器具有多个输入端口和多个输出端口，其中在所述输入端口接收内容流，并将其路由到所述输出端口，所述内容流包括多个部件的层次并具有包括码率、类型和名称在内的属性，所述图形用户界面包括：

目录结构查看屏幕，用于显示表示所述输入端口和所述输出端口的图标，并用于显示具有指示输入和输出内容流的层次结构的多个分支的输入和输出目录结构；
25

所述图形用户界面允许通过将内容流部件从所述输入目录结构拖动并将所述内容流部件放置到所述输出目录结构，从而对路由控制进行配置，由此通过所述多路传输复用器将所述内容流自动路由到所希望的输出。

17. 如权利要求 16 所述的图形用户界面，其中，可以同时从所述输入目录结构拖动多个内容流部件并将其放置到输出目录结构中，由此通过所述多路传输复用器将多个内容流自动路由到所述输出。

- 5 18. 一种用于多路传输复用器的图形用户界面，所述多路传输复用器具有多个输入端口和多个输出端口，其中在所述输入端口接收动态内容流，并根据路由控制数据将其路由到所述输出端口，所述图形用户界面包括：

 多个目录结构查看屏幕，用于显示表示所述输入端口和所述输出
10 端口的图标，所述目录结构查看屏幕还显示具有指示各自输入和输出内容流的层次结构的多个分支的输入和输出目录结构，每个目录结构查看屏幕都以唯一的时间段示出了内容流路由；

 包括多个时间段的时间条，这些时间段定义在多对的段开始和结束时间之间；和

- 15 所述图形用户界面允许通过选择所述时间条的相应的时间段来连续地显示所述多个目录结构显示屏幕。

多路传输复用器的图形用户界面

5 本发明要求根据 35 U.S.C. 119(e)的、于 2001 年 4 月 20 日提交、
题为 “Broadband Bandwidth Management, Device Management and
Multi-Media Control System (宽带带宽管理, 设备管理与多媒体控制
系统)”、申请号为 60/285,188 以及于 2001 年 4 月 20 日提交、题为
10 “Data Insertion, Transport, Grooming Aliasing, Routing, and
Multiplexing of MPEG 2 Data Streams (MPEG 数据流的数据插入、传
输、修饰、混叠、路由和多路复用技术)”、申请号为 60/285,153 的
共同未决的美国临时申请权益; 在这里通过引用将这些临时专利结合
进来。

15 本申请与 2002 年 4 月 16 日提交 (基于 2001 年 9 月 13 日提交的
临时专利 60/322,063)、题为 “High Speed Serial Data Transport Between
Communications Hardware Modules (通信硬件模块间的高速串行数据
传输)” 的共同未决的美国专利申请___/___,___相关, 在这里通过引用
将其结合进来。

20

技术领域

 本发明涉及为通信硬件的使用提供图形用户界面的系统、处理、
方法、装置和相关软件。更具体地, 本发明涉及有助于通过多路传输
复用器监控和配置内容流路由的图形用户界面领域。因此, 本发明的
25 一般目标是提供具有以上特性的新颖的系统、方法、装置和软件。

背景技术

 汇聚视频、音频和数据的宽带媒体建立了一个不同标准和传统通
信技术的无序环境。需要在宽带硬件、控制系统和操作者之间运用该
30 系统中的物理和逻辑资源之间的关系, 并使这种关系在三者之间建立

联系。常规上，这应该通过使用与发送多媒体内容的任务相关联的每块通信硬件的专用控制计算机来完成。因此，现在的通信硬件人员（诸如电缆操作员和电视节目编排者）使用了大量的不同的设备，而其中每个设备都是由独立计算机上的独立控制系统来管理的。而这些常规通信系统通常也会需要使用很多技术的操作员界面来配置常规宽带设备。由于这些或者其它的原因，大多数这样的系统都难以使用、非直觉、晦涩、低效率，并且购买和操作将会是十分昂贵的。

传统宽带通信硬件的其它问题在于其不能够向操作者方便地提供关于系统硬件和软件的信息。这会在这种设备的工作中造成的低效率。例如，因为系统操作员必须物理地检查宽带通信硬件以确定所使用的系统和操作状态，所以对系统错误进行故障诊断在当前来说是困难而昂贵的处理过程。具体地说，对系统难点进行故障诊断可能需要操作员检查通信硬件的框架（rack）以确定是否所有的通信硬件都插入、接通、连接到内容流的源和/或以特殊方式进行操作。

除了与确定硬件配置和状态相关的难点之外，常规的宽带通信硬件还很难以确定现有内容流路由配置。这部分地是由于现有宽带通信硬件并没有足够完善到能够指示流入数据流的不同属性而造成的。此外，现有的常规宽带通信硬件的用户界面需要具有较高技术的人员才能为不同程序配置路由控制。与路由控制的常规方法相关联的难点使得很难有效地利用系统带宽。因此，这类常规系统的操作员通常不能充分利用可利用的带宽来进行内容流路由。假设传输带宽在本领域中仍旧是很珍贵的商品的话，那么其将是十分稀缺的。

与常规宽带通信硬件相关联的进一步的不足之处在于，这样的硬件必须手工地进行配置。因此，现有系统通常需要操作员熟练地使用晦涩和高深的计算机代码操作常规宽带通信硬件。随着信息/内容选项的持续增加，就会越来越难以手工配置宽带通信硬件以发送内容流。此外，这样的系统必须持续地工作并且每当要执行内容流路由的改变

时就适当地进行配置。

5 因此，本领域中需要能够通过多路传输复用器提供内容流路由的简单和实时显示的新方法、系统和装置。这样的方法和装置应该包括直观的图形用户界面，其能够向操作者提供关于相关系统硬件和软件的状态信息。这样的方法和装置还应该包括多功能（**feature-rich**）的图形用户界面，该图形用户界面能够使操作员通过多路传输复用器轻松地监控和操作内容流。

10 因此，在本领域中还需要能够允许对多流成分的内容流路由同时进行配置的新方法、系统和装置。这样的方法和装置应该足够灵活从而允许操作员同时对单个内容流成分以及这些成分的群组（例如程序、组件等）的路由控制进行配置。

15 本领域中进一步需要能够允许对宽带多路复用器中未来路由事件的路由控制进行配置的新方法、系统和装置。这样的方法和装置也应该允许操作员配置多路传输复用器，从而在未来时间点，即使操作员不存在，也可以自动地改变路由配置。这样的系统将允许通过在延长的时间段上对路由配置信息进行预先设定从而自动控制宽带媒体硬件，该系统还将能够自动进行配置更改。

20

本领域另外还需要能够显示直观的和协调有助记忆的标识符的多路传输复用器图形用户界面，所述标识符可用于显示在遍及图形用户界面的不同显示屏上的不同内容流。

25

因此，本领域进一步需要能够通过多路传输复用器同时提供内容流路由控制的手动与计算机辅助配置的新方法、系统和装置。这样的方法和装置还将优选地包括允许操作者通过简单地使用诸如图标的图形对象来配置路由控制的图形用户界面。

30

本领域还需要能够在宽带多路复用器中提供硬件识别/配置和/或状态数据的实时显示的新方法、系统和装置。这样的方法和装置应该可以使系统操作员从单独的图形用户界面识别通信硬件的物理结构和配置，并由此避免对通信硬件进行物理检查的必要性。这样的方法和装置还应该能够反映出当前通信硬件的状态，从而使操作员能够检验不同设备是否正在如所期望地进行工作。

发明内容

本发明通过提供能够通过多路传输复用器实时监控和使用内容流的方法、系统和装置，从而满足了上述需要并且克服了相关技术的上述和其它不足。本发明可以采取图形用户界面的形式，该图形用户界面能够显示系统硬件识别、结构、配置和/或状态信息。该界面还可以通过简单地使用图形对象（诸如在图标上进行的拖放操作）提供计算机辅助的路由配置。为了进一步简化操作，本发明能够显示直观的和协调有助记忆的标识符，该标识符可用于显示在遍及图形用户界面的不同显示屏上的不同内容流。可以通过允许操作员同时配置单个内容流成分以及这些成分的群组的路由控制从而进一步简化路由控制。更进一步地通过未来路由事件的配置提供了灵活性，由此使得能够在未来时刻自动执行配置更改。

20

本发明可以采取如下方法的形式，通过具有多个输入和输出端口的多路传输复用器，使用图形用户界面来配置内容流的路由控制。本发明的方法包括显示表示输入和输出端口的图标和表示由输入端口之一接收到的输入内容流的图标。输入流图标被拖放到（dragged and dropped）输出端口图标附近，使得可以显示出表示输出流正流出输出端口的图标。拖放的输出流图标可以表示目前正流出输出端口的输出流或者将要在未来的一个预设时刻开始流出输出端口的输出流。在另一种方法的实施例中，该方法可以显示字母数字字符串，该字符串可以随着输入流图标拖放，并由此显示出表示输出流的字母数字字符串。

30

当然，本发明的上述方法尤其适用于与本发明的上述装置一同使用。同样地，本发明的装置也适用于进行上述的本发明的方法。

5 根据下面对优选实施例的详细描述，根据权利要求以及根据附图，本发明的许多其它优点和功能对于本领域普通技术人员来说将会是显而易见的。

附图说明

10 下文中将结合附图描述本发明的优选实施例，其中相同的数字表示相同的单元，其中：

图 1a 示出了根据本发明的一个优选实施例的多路传输复用器的硬件结构；

15 图 1b 示出了图 1a 的多路传输复用器的固件主处理器结构的优选形式；

图 2 示出了图 1 的多路传输复用器的系统初始化和资源发现处理，该处理是根据本发明的一个优选实施例的；

图 3 示出了根据本发明的一个优选实施例的不同硬件属性查看性能；

20 图 4 示出了根据本发明的一个优选实施例的系统属性查看性能；

图 5 示出了根据本发明的一个优选实施例、使处理能够进行的不同输出端口；

图 6 示出了根据本发明的一个优选实施例的现有视频和/或音频流路由事件的详细指定；

25 图 7 示出了根据本发明的一个优选实施例的不同系统带宽利用查看性能；

图 8 示出了根据本发明的一个优选实施例的某种事件记录和查看性能以及处理过程；

30 图 9 示出了根据本发明的一个优选实施例的未来内容流路由事件的详细指定；

图 10 示出了根据本发明的一个优选实施例的不同 IP 数据封装和插入性能；

图 11 是更详细地说明图 10 的 IP 数据封装和插入性能的详细流程图。

5

具体实施方式

随后的详细说明书仅仅是提供优选的示范实施例，而并不是想限制本发明的范围、适用性或配置。相反，随后的对优选示范实施例的详细说明将向本领域技术人员提供一个能够实施本发明的优选的示范实施例的说明。应该可以理解，在不背离如所附权利要求所阐述的本发明的精神和范围的前提下，可以在各部件的功能结构上对本发明做出各种不同的修改。

如图 1a 和 1b 所示，TMX 20 包括多个硬件、固件和软件部件。图 1a 是根据本发明一个优选实施例的多路传输复用器（TMX）20 的硬件结构示意图。如图所示，例如，多路传输复用器可包括计算机 22'（具有单元管理器 22 和 GUI 80），其通过网络 24 通信连接到 TMX 框架 42。TMX 框架 42 优选地包括主处理器板 40'（优选地具有 VxWorks 操作系统）、输入处理器板 50'、输出处理器板 55'、多路复用器板 60' 和代码转换板 70'。这些部件优选地（但不一定非要）通过单一的 TMX 框架 42 相互通信连接。该优选的 TMX 框架硬件的基本物理模型如下：TMX 框架是一个装载板的器具，每个板又都是一个装载端口和处理器的器具，每个处理器都是装载固件的器具，每个端口则是装载协议体系（诸如 DS3、MPEG、TCP/IP 等）的器具。该框架以及每个板都具有一组状态和与之相关的状态变量。其包括：1) 管理状态，用于同步多路管理器的配置访问；2) 操作状态，用于指示 TMX（或其部件）是否处于完全启动和操作状态；和 3) 提示状态，用于通过 TMX 或其部件发信号通知不同的提示条件。如下面所详细描述，主处理器 40 控制 TMX 20 的不同的硬件和软件部件，并根据 SNMP 储存 MIB 表数据，以供 TMX 的其它的不同部件和单元管理器使用。

多路传输复用器 20 适合于很多不同的应用环境，包括：(1) 电缆头端；(2) 卫星上行链路；和(3) 陆地广播。根据修改的简单网络管理协议 (SNMP) 优选地进行单元管理器 22 和 TMX 框架 42 之间的通信，并且将要通过多路传输复用器 20 发送的内容流是优选地根据公知的 MPEG 标准之一的。更优选地，内容流是 MPEG2 数据流。尽管本发明的某些优选实施例使用根据公知 SNMP 标准的某些常规 MIB 表，这里所引用的 MIB 中的许多都包括新的数据结构。这些数据结构在计算机程序列表中被充分地定义，所述计算机程序列表将在本说明书的末尾给出。因此，根据这里所包含的说明，本领域普通技术人员将更加详尽地理解那些新数据结构的性质和功能。

前面提及的 2002 年 4 月 16 日提交、题为 “High Speed Serial Data Transport Between Communications Hardware Modules (通信硬件模块间的高速串行数据传输)” 的共同未决的美国专利申请____/____,____中提供了对 TMX 20 的硬件部件的更为详细的说明，在这里通过引用将其结合进来。因此，没有必要对这些硬件部件做进一步地详细讨论，通过参考这些结合进来的申请可以完全地理解这些部件。

需要利用系统中物理和逻辑资源之间的关系，并使该关系在 TMX 框架 42、单元管理器 22 和人类操作员 10 之间进行通信连接（例如如图 2）。这通常是通过对系统的不同硬件部件进行建模而获得的。TMX 建模单元通过使用标准 “MIB 2” 接口表 (ifTable) 被集成到 SNMP 管理信息库 (MIB) 中，从而将每一个建模单元表示为具有如下所指定的特定扩展的接口。这允许任意板和/或端口通过其表指针 (ifTable 中的 ifindex) 进行标识。

继续参看图 1a，单元管理器 22 优选地通过以太网连接到 TMX 框架 42。本领域技术人员可以认识到，另外也可以使用其它的网络技术。单元管理器 22 可以作为 java 应用程序通过使用浏览器从 TMX 42

上载到远程计算机，在初始建立阶段，该远程计算机优选地与多路传输复用器通过网络 24 通信连接。随后，单元管理器 22 可以在远程计算机上作为 java 程序运行。该远程计算机优选地是具有常规操作系统和浏览器的常规个人电脑，该系统允许在安装单元管理器 22 之后在
5 TMX 框架 42 上进行控制。优选地，将图形用户界面结合到单元管理器 22 中，以下将对其做详细地描述。优选地，在常规个人电脑的显示器（例如 LCD 屏幕或 CRT 显示器）上向操作员呈现 GUI。根据这里所包含的公开的内容，本领域普通技术人员将很容易想到用于掌管和操作图形用户界面和单元管理器 22 的多种不同的其它软件和硬件
10 组件。

图 1b 图示说明了 TMX 20 的不同的固件和软件部件 52-69，其如图 1b 所示地相互通信连接。这些组件包括 SNMP 中介 44、消息控制器 45 和故障管理器 59。TMX 20 进一步包括资源管理器 52、配置管理器 46、
15 PAT/PMT 集模块 54、PSIP 集模块 57、输入模块 50、IP 封装模块 66、时间表管理器 67 和大量的 DSP API。这些包括多路复用器处理 60、代码转换处理 62 和量化级处理 64。在本优选实施例的这些固件模块和某些硬件部件之间存在着一一对应的关系。可以在图 1a 中找到相应的硬件部件，其包括输入处理板 50'、多路复用器板 60'（具有多路复用处理器 60 和量化级处理器 61）、代码转换板 70'（具有 5 个代码
20 转换处理器或 3 个代码转换处理器、1 个多路复用处理器和 1 个 QLP 71'）。因此，当配置管理器在固件模块上进行操作时，也将影响相应的硬件模块。TMX 框架 42 内的不同部件之间的信息和命令流通常如在图 1a 和 1b 中通过使用箭头所指示的那样。更具体的说，来自单元管理器 22 的命令和信息流通过了 SNMP 中介 44，其将来自单元管理器 22 的 SNMP 协议命令转换成常规形式，以使它们能够被 TMX 20 的其它不同部件所理解。优选的常规通信协议是简单的协议，其中，
25 将大量指示性请求或命令随同相关的数据结构一同传递，用于根据相关的命令接收将要使用的数据。因此，SNMP 中介 44 通常用作单元管理器 22 和主处理器固件之间的中间代理。SNMP 中介 44 允许在
30

诸如修饰 (grooming)、拼接、数据插入等的固件功能上进行基于 SNMP 的管理, 并进行控制, 这是因为它向接口提供了不同的固件模块 (诸如输入处理任务 50、多路复用器处理 60、代码转换处理 62 和量化级处理 64), 这些不同的固件模块基本上提供了所需的功能。

5

配置管理器 46 通过 MIB 消息控制器 45 从 SNMP 中介 44 接收命令和信息并且确定怎样利用硬件和其它固件在板卡级执行那些命令。参看图 2 到图 9 以及说明书后续部分对于这些图的相应详细描述, 将获得对于 TMX 20 的不同其它部件的详细理解。

10

图 2 到图 11 图示说明了多路传输复用器 20 的九个主要的操作方面。这九个操作方面包括: (1) 初始化并发现系统资源 100; (2) 查看系统硬件属性 134; (3) 查看系统软件属性 156; (4) 开启输出端口 166; (5) 详细指定现有视频和/或音频路由事件; (6) 查看带宽利用 206; (7) 查看记录活动性 222; (8) 详细指定未来路由事件 238; 和(9) IP 数据封装和插入 260。下面将立即开始详细讨论本发明的这些方面。

15

参看图 2, 图示说明了图 1 的宽带多路复用器的系统初始化和资源发现的处理过程, 该过程是根据本发明的一个优选实施例的。如所示, 通过对 TMX 框架 42 上电 101 来初始化和发现本发明的系统开始, 由此资源管理器 52 将进行硬件和系统软件信息的发现 (102)。因此, TMX 框架 42 将在 104 执行大量的功能以识别安装在 TMX 框架 42 中的系统部件。MIB 消息控制器 45 也是在 102 给合适的 MIB (ifTable 和 ifStack) 组装信息, SNMP 中介 44 在 106 等待查询。完成这些任务之后, TMX 框架 42 将根据操作员通过单元管理器 22 提供给 TMX 框架 42 的驱动命令而准备执行不同的活动。

20

25

这里, 操作员 10 可以开启单元管理器 22, 作为对此操作的响应, 单元管理器将在 104 显示图形用户界面 80, 示出用于查看的空目录结构查看屏幕 81。空目录结构查看屏幕 81 包括输入目录结构窗口 82、

30

输出目录结构窗口 82'和记录消息窗口 87。单元管理器 22 在 108 自动地读取合适的 MIB 已发现目前安装在 TMX 20 中的硬件。这包括系统硬件属性数据, 诸如端口数据和/或物理结构。本发明的优选实施例支持几种类型的端口 (例如 ASI、DHEI、SMPTE 310、DS3)。不同端口参数的数据将由 ifentry MIB 表描述/定义。单元管理器 22 在 110 将合适的 DSP 代码下载到 IdentxTable MIB。TMX 框架 42 的 SNMP 中介 44 在 112 创建新 MIB 条目, 消息控制器 45 将该信息传递到配置管理器 46 以进行实现。这里, 单元管理器 22 在 114 请求 PAT 数据。这个请求将由 TMX 在 116 进行处理。单元管理器 22 在 120 解析 PAT, 以使合适的 PMT 能够被识别。这些是在 122 所请求的, 这个请求将由 TMX 在 124 进行处理。产生了请求信息之后, SNMP 中介 44 将在 126 进一步等待查询。这个数据随后被单元管理器 22 在 128 读取, 并且更新图形用户界面 80。具体地说, 请求的数据被用于在目录结构查看屏幕 81 上组装系统硬件图标 84 和 84'以及 (优选地有助记忆的) 硬件名称 83 和 83', 这些信息是通用 PSIP 集模块 57 从数据流本身中提取的。因此, 操作员 10 将看到系统硬件部件的可视显示。

从 TMX 框架 42 接收系统硬件属性数据之后, 单元管理器 22 将继续重现和显示可能在 130 产生的记录消息。这是在 132 通过故障管理器 59 和 SNMP 中介 44 的帮助完成的。因此, 一旦记录查询开始, 单元管理器 22 就会在 132 在图形用户界面 80 上显示端口和记录数据, 其中将更新目录结构查看屏幕以在记录消息窗口 87 中显示输入端口 85、输出端口 85'和记录消息 88。如所示, 输入和输出端口 85 和 85'优选地具有相关的有助记忆的字母数字标识符。这些端口同时也优选地是彩色的代码, 可以显示端口是否处于活动状态。再次观看到新完成的目录结构查看屏幕 81 之后, 操作员 10 可以启动不同的活动, 如下参照图 3 到 10 所述。例如, 这些活动可包括查看系统硬件属性 134、查看系统软件属性 156、开启输出端口 166、详细指定现有视频和/或音频路由事件 184、查看带宽利用 206、查看记录活动 222、详细指定未来路由事件 238 以及 IP 数据封装和插入事件 260。操作员也可进行

其它不同的相关活动，对于本领域普通技术人员来说，根据这里所包含的公开内容将很容易推出这一点。

5 现在看图 3，此图图示说明了根据本发明的一个优选实施例的不同系统硬件属性的查看处理过程 134。图 3 所示的硬件处理是由操作员 10 根据从目录结构查看屏幕 81 顶部的目录条目中选择框架查看屏幕而发起的。此操作可以通过选择“查看”屏幕顶部的目录条目并选择框架查看选项而完成。可用的硬件查看选项包括“前框架查看”和“后框架查看”以及“系统信息”。根据在图形用户界面上对框架查看选项之一的选择，单元管理器 22 在 TMX 框架的协助下从合适的 MIB 10 收集请求的硬件信息（136）。如 138 所示，这个 MIB 数据是由 TMX 框架 42 提供的，随后，单元管理器 22 将会在框架查看屏幕 89 和 90 之一上显示信息。

15 继续参看图 3，可以看到图形用户界面 80 使用了接收到的硬件和状态数据来显示系统硬件属性，具体地说就是框架查看屏幕 89 和 90，如操作员 10 所最初请求的一样。前框架查看屏 89 包括不同的图形对象，用于指示在 TMX 框架 42 内接收到的不同板卡的标识、物理结构、配置以及状态。在此说明性实例中，这些板卡包括 CPU 卡 40''、20 多路复用器卡 60''、第一输入处理器卡 50''和第二输入处理器卡 50''。尽管也可以在前框架查看屏 89 的记录消息窗口 87 内接收记录消息，但是在本说明性实例中不会产生记录消息。

25 操作员 10 也可以选择后框架查看屏 90 来替换前框架查看屏 89。在该说明性实例中，后框架查看屏 90 包括不同的图形对象，用于指示在 TMX 框架 42 内接收到的不同板卡的后部的标识、物理结构、配置以及状态，可以参考上述有关前框架查看的讨论。可以选择将记录消息显示在后框架查看屏 90 的记录消息窗口 87 中。本发明的这个方面允许操作员 10 轻易地选择并随后以上述讨论的方式查看系统硬件 30 属性。由于允许操作员 10 通过多路传输复用器 20 对难点进行故障检

修，而不必物理上访问通信硬件本身，所以本发明的这个功能特别有利。

5 在 139 示出了本发明的优选的连续硬件状态查询的功能。具体地说，通过重复执行框 140-146 的功能，以一定的间隔更新在框架查看屏幕中提供的 LED 状态信息。

10 现在参看图 4，本图说明了根据本发明的一个优选实施例的系统属性查看处理过程和性能 156。如其中所示，当 TMX 执行 158 所示的功能时，系统属性（诸如板类型、DSP 属性、软件版本等）的查看就可以开始其初始的系统发现处理。因此，该信息可很容易用于显示，SNMP 中介 44 在 160 等待这样的查询。操作员 10 在目录结构查看屏幕 81 的顶部选择了版本查看目录选项之后，单元管理器 22 将会在 162 收集请求的信息并将在系统属性屏幕 91 中将其显示出来。随后可以
15 如操作员所需地将数据显示出来。如图 4 和下面的表 1 所示，显示在屏幕 91 上的系统属性数据优选地包括板和运行在每个框架槽上的软件的如下数据域：

表 1
板名称
TMX 应用
JVM 版本
系统名称
IP 地址
框架 ID
板修订版本
FPGA 版本
VxWorks 08
CPU 版本
MAP 库版本
MUX 版本
QLP 版本
TPE 版本

在图 4 的说明性实施例中，TMX 框架 42 是一个在框架的每半个部分都具有五个板槽的中平面 TMX 框架。因此，这个说明性实例一共包括十个槽（每半平面各有五个槽）。在引用结合进来的发明申请中包括了对 TMX 框架 42 的结构和操作的详细描述，对于本领域技术人员来说，根据这里所包括的公开内容将很容易推出各种不同的变种装置。

如图 3 和 4 所示，本系统的优选实施例包括具有系统信息表的 GUI，通过这个 GUI 操作员可以访问有关系统的信息，诸如系统名称、系统说明、系统正常工作时间和系统位置。本发明的这个功能的操作方式通常类似于图 4 示出的并在上面刚刚描述过的查看软件版本功能。

图 5 图示说明了本脚本发明的一个优选实施例的不同输出端口启动性能。如所示，操作员 10 选择将要开启的特定端口之后，就可以发起输出端口启动。在选择了端口之后，单元管理器 22 将在 168 显示传输编辑器 92 的缺省值。操作员 10 随后可以显示缺省数据，如果需要还会编辑数据，诸如可以把状态从禁止变为启动。例如，操作员通常将启动传输流，命名该流，并为所选择的端口分配信息传输码率。在选择“OK”按钮之后，将关闭传输编辑器，单元管理器 22 收集来自编辑器的传输信息并将其放置在合适的 MIB 表中（参见 170）。TMX 框架也会使用该信息来执行启动请求，如 172 所示。根据不同端口类型的变化的信息要求，MIB 表可以是这两种类型之一：TMXiftable（用于大多数端口）或者 TMXgiexttable（用于 DS3 端口）。

此外，单元管理器 22 在 174 创建一个 PAT，该 PAT 由 TMX 输出，如 176 所示。最后，单元管理器将更新 GUI 的目录结构查看屏 81，如 178 所示。图形用户界面 80 通过改变目录结构查看屏 81 中的端口图标属性，指示了所需端口的成功开启。这优选地是通过改变端口

图标的颜色而实现的，当然，本领域普通技术人员将很容易想到以其它的方式（诸如改变形状、运动、位置、尺寸、声音等等）实现。因此，操作员 10 可以通过观看最新更新过的图形用户界面 80，可视地确认端口开启已经成功地完成。

5

图 6 图示说明了根据本发明的一个优选实施例的不同的系统辅助的视频和/或音频路由性能。如下面更为详细地描述，本发明使操作员 10 能够手动或半自动地定义并执行内容流路由。具体地说，本发明的优选实施例给操作员提供了一个字母接一个字母地手动输入路由数据的能力，或者拖放图形对象到目录结构查看屏 81 的不同位置、从目录结构查看屏 81 的不同位置拖放图形对象。单元管理器 22 与图形用户界面 80 合作执行由相应的拖放操作所指定的不同路由规范命令。这可以通过对应于需要定义路由命令的不同实际域的 MIB 表的自动组装而实现。图形用户界面 80 上的拖放操作帮助操作员 10 为系统定义视频、音频和/或 IP 数据路由事件。因此以这种方式定义路由规范就是半自动的。

10
15

图形用户界面上的拖放操作可用于进行不同的相关内容流路由功能。这些功能包括将与输入目录结构不同的层次拖到输出目录结构中。例如，操作员可以拖动 (1) 整个输入端口的内容流（可能包括多个程序，其中每个都可能包括多个部件）；(2) 输入端口到输出端口的完整程序；(3) 从输入端口到输出端口程序的完整程序；和(4) 从输入端口到输出端口的部件。根据这里所包含的公开内容，本领域普通技术人员将很容易地想到大量的其它拖放部件。但是，应该注意到，本说明书的这个部分具体地对目前出现的内容流路由进行寻址。然而，本发明也展望了将要在未来时间里自动执行的内容流路由的配置（例如参看图 9）。如下面更详细地描述，下面紧接着要描述的内容流处理过程（适用于执行现有路由命令）是与用于执行未来路由事件的路由操作协调一致的，并且构成了用于执行未来路由事件的路由操作的一部分。

20

25

30

主要参看图 6，操作员 10 可以通过选择表示一个或多个将要发送到所需位置（例如输出端口）的内容流的图形对象，详细指定一个或多个现有路由事件。内容流可以是简单的或者可以包含多个部件，
5 这些部件可以或不可以以一个或多种方式相互相关。例如，对象可以表示单一的部件内容流、共同组成一个程序的多个内容流或者在整个输入端口上共同组成数据流的多个内容流。在下面紧接着讨论的说明性实例中，操作员 10 将用于整个端口的内容流从输入目录结构拖动到输出目录结构中，并且继续编辑来自端口的一个程序中的视频和音频部件。
10

优选地，是通过对一个或多个图形对象从目录结构查看屏 81 的输入端口窗口 82 到输出端口窗口 82'进行拖放操作而完成根据本发明的协助路由的。这项操作具有获取与所选数据流的源相对应的配置数据的效果，如 186 所示。例如，拖放所需图形对象使得单元管理器 22
15 能够自动获取所需路由事件的相应的配置数据，诸如输入端口数量和位置、输出端口数量和位置、将要发送的内容流 PID 和将要发送的内容流的码率。此外，有关目标输出端口的信息（根据对象被放在哪里而决定）也将由单元管理器 22 获取，例如，该信息包括目标输出端口的
20 位置。该信息使单元管理器 22 能够在 186 创建缺省设置并自动进行 PID 混叠，从而使得在通过多路传输复用器 20 发送不同流时不存在任何数据流冲突。然后，单元管理器 22 显示出拖放编辑器 93 和 94，如 188 所示。操作员随后可以选择特定的部件进行编辑，并且在 192，单元管理器 22 接收这个选择并显示部件编辑器（用于视频流的
25 95 和用于音频流的 95'）的缺省信息，用于可能进行的编辑。如果部件编辑器 95 中所示的缺省数据对于操作员是可接受的话，就可以选择“OK”按钮以通知单元管理器可以进行后续的操作。具体地说，关闭部件编辑窗口将会使得单元管理器 22 收集来自 GUI 的信息并请求创建不同的 MIB 表条目，如 194 所示。TMX 在 196 根据更新的 MIB
30 执行路由事件，而在 198，由单元管理器 22 适当地更新 GUI。在操作

员看来，通过把图标从输入目录结构拖放到输出目录结构已经详细指定并简单进行了路由。事实上，通过如下所详述的系统的帮助已经详细指定了不同的路由参数。

5 如果操作员 10 希望修改缺省的和/或获取的数据的话，操作员 10 就具有为内容流包含的每个部件详细编辑信息的能力。在示出的实例中，操作员 10 选择了编辑器窗口 93 中所示的程序 1（一般地，操作员将选择某些类型的图形对象，诸如图标或其相关的文本），接着显示详细编辑器窗口 94，该窗口中示出了所选程序的组成部件。在图 6
10 的例子中，通过编辑选择了程序 1，其包括一个视频部件和一个音频部件。

图形用户界面 80 优选地具有通过使用不同屏幕上的不同图形对象来识别内容流的能力，所述不同图形对象包括图标、字母数字字符串、实际程序名等。内容流标识数据优选地是携带在媒体流内的，这样，不管呈现在操作员 10 面前的是哪个查看屏幕，内容流标识数据都可以在整个图形用户界面中始终被显示。这里要重申，图形用户界面 80 优选地呈现一致的内容流名称或符号，并且可以贯穿整个界面过程显示。

20 继续参看图 6，对窗口 94 的“OK”按钮的选择将关闭拖放窗口并打开对应于所选部件的部件编辑器窗口，如 192 所示。在这种情况下，选择将要编辑的部件之后将会显示部件编辑器窗口 95 和 95'中的一个，在这里操作员 10 还具有为任何所需程序的部件指定详细信息
25 （诸如码率、目标 PID 等）的能力。在这种情况下，视频编辑器窗口 95 和音频编辑器 95'都会显示出来，便于考虑和可能进行的编辑。这个功能使得用户能够更加容易地在正在发送的不同内容流中分配带宽，从而获得最大带宽利用率。

30 选择了窗口 95 或 95'之一的“OK”按钮之后，单元管理器 22 会

根据编辑的更改来改变 MIB 表数据，并通知 TMX 执行指定的路由配置。配置管理器 46 随后按顺序继续配置目标多路复用器和量化级处理器并启动输入处理器，如 196 所示。

5 当输出端口开启时，模块激活的顺序是本发明的一个重要方面。为了有效地执行路由事件，目标多路复用器、量化级处理器和输入处理器应该按照指定的顺序激活，从而将使系统不稳定的可能性降到最低。具体地说，配置管理器 46 使目标多路复用器收集所指定的 PID 并且将它们发送到目标输出。其次，配置管理器 46 必须向量化级处理器 64 提供将要发送的内容流所需的合适的码率和 PMT。再次，配置管理器 46 应该通知输入处理器将具有特定 PID 的所有内容流发送到多路复用器。通过进行 PID 混叠并且随后将相关的数据作为低电压差分信号发送到多路复用器，优选地完成这项操作。

15 如上面所提及的，使用不同于上述的顺序将模块激活，将会导致系统不稳定。例如，如果配置管理器想要首先启动输入处理器的话，多路复用器就可以开始接收内容流，这是所不期望的，并且这样的混乱将导致多路复用器失效。类似地，删除内容流（中止流到端口的发送）也应该以配置管理器 46 所指定的预设顺序进行。具体地说，应该将上述的顺序倒过来（对输入处理器去激活，对 QLP 去激活，最后对多路复用器去激活）。例如，如果首先停止多路复用器的话，多路复用器仍旧可能接收来自输入处理器的内容流，那么这种情况将会再次导致多路复用器失效。

25 现在参看图 7，本图说明了根据本发明的一个优选实施例的不同系统带宽利用查看性能。如图所示，操作员 10 通过从目录结构查看屏幕 81 的上部选择带宽管理器目录条目，发起本发明的查看带宽利用性能。这将使得单元管理器 22 在 208 能够显示带宽管理器屏幕，TMX 开始查询系统的带宽使用数据并开始等待对该数据的进一步查询，如 210-212 所示。如本发明书末尾所附计算机程序中更加完整的表示，

MIB 表启动对 MPEG 输入/输出带宽利用信息的监测。具体地说，TMXinputPIDtable 用作监控每个 PID 的输入速率。TMXoutputPIDtable 用作监控每个 PID 的输出速率。

5 具体地说，消息控制器 45 开始查询输入处理器和输出多路复用器的数据，该数据用于更新 MIB 表（从这两个源获取数据使得带宽查看窗口显示出输入带宽和输出带宽之间的比较），消息控制器还将该数据作为 SNMP 数据发送至单元管理器 22，如 214 所示。单元管理器 22 周期性地查询 TMX 的这个信息，并且在 216 在图形用户界面 80
10 上显示该数据。随后将在 214 继续查询新的带宽利用数据。这样，将会不断地更新所有开启端口的带宽利用数据，并且可以通过图形用户界面 80 实时地显示该数据。最好在操作员关闭带宽窗口 96 和 96'（诸如切换到框架或目录结构查看屏）时中止带宽数据查询。此时，将从 MIB 表中删除启动的内容流的 PID。

15 接收到带宽利用数据之后，图形用户界面 80 会显示带宽利用屏幕 96。这个屏幕优选地包括自动改变比例的 x 和 y 轴以及一个用于正发送的每个内容流的单独图形对象，每个对象优选地都是带宽条（所示实例中的条 97、97' 和 97''）。屏幕 96 中示出的每个带宽条都最好
20 包括下述的多个属性：输出带宽使用值 97a，输入带宽使用值 97b，最大输入带宽使用值 97c 和最小输入带宽使用值 97d。实际上，带宽利用上的变化是实时地自动显示在带宽利用屏幕 96 中的。

 带宽利用屏幕 96 可包括大量面向用户友好的功能，从而使图形
25 用户界面更加直观和有用。例如，可以向操作员 10 提供选择或取消选择在带宽利用屏幕 96 的右部上所示的图例显示的能力。同样地，操作员 10 最好具有选择或取消选择显示最小和最大带宽使用值的能力。此外，屏幕 96 最好具有为不同流显示相同有助记忆的标识符的能力，所述标识符可用于诸如目录结构查看屏幕的其它屏幕。需要重申的是，图形用户界面最好在系统中为每个内容流反映一致的标
30

标识符。当然，所过需要，也可以使用其它的标识符。这些标识符最好是通过内容流发送的，以便它们能够被删除并在不同的屏幕中显示。如上所述，这些标识符可以显示为彩色的图标和/或字母数字字符串等。

5

在查看带宽利用屏幕 96 之后，操作员 10 可以选择一个带宽条来动态地显示所选条上有关组成内容流的不同部件的更加详细的信息。例如，给定的程序可能包括一个视频和两个音频部件。选择带宽条将会使得详细带宽利用窗口 96'（以及有关这些部件的附加信息）出现在屏幕上。这种类型的选择将使单元管理器 22 在 216 产生查询，在 210/212 将由 TMX 做出响应。如窗口 96'中所示，此时所选取带宽条的程序名称、组 ID 和全部带宽将被获得并显示在屏幕上。在这个说明性实施例中，当带宽利用大约为 5 Mbit/s 时（比较图 7 的窗口 96 和 96'），选择程序 2 的带宽条。此外，详细窗口将所选程序分解为其组成的部件。在这个例子中，程序具有三个组成部分：IP 数据 1、视频数据 1 和音频数据 1。

屏幕 96'示出了该程序每个部件的详细信息。该信息最好包括选择详细带宽利用窗口时的组成部件的带宽最小值、带宽最大值和瞬时带宽使用。结合参考屏幕 96 和 96'可以看出，所显示的组成部件的带宽使用加在一起就是整个程序的带宽使用。此外，组成部件的最小值的和等于作为一个整体的程序的最小值。类似地，整个程序的最大值等于每个组成部件的最大值的和。最后，查看屏显示了与程序相关的分组标识符 PID。

25

由于该系统的这个方面可以实时地显示带宽，所以操作员将随着时间的变换而看到带宽的使用情况。不同时间点上的带宽使用情况的差异反映出了输入信号可以根据时间在整个系统的输入端上进行变化。例如，如果输入信号突然包括附加的部件，带宽查看屏幕就将实时地反映出这一变化。

30

图 8 图示说明了根据本发明的一个优选实施例的不同事件记录和查看性能 222。如图所示，系统具有过滤显示在图形用户界面上的记录消息的能力。根据本发明的查看记录信息最开始需要操作员选择合适的记录过滤级别，由此使系统处于四种模式之一。过滤级别是由单元管理器 22 记录的，通过过滤级别来规定显示在图形用户界面 80 的记录消息窗口 87 中的消息的数量和类型。可以从目录结构查看屏 81 顶部附近的“查看”下拉目录条目中选择所需的记录过滤级别，然后选择记录消息选项。优选地具有四种过滤级别：正常状态、紧急状态、故障状态和调试。在调试模式中，所有产生的记录消息都将显示出来。

启动之后，TMX 框架 42 状态查询任务开始查询系统由此产生用于组装 TMXLogPortTable 的记录消息，如 224 所示。SNMP 中介 44 随后等待对查询这些信息的响应，如 226 所示。这个记录消息可以由不同的固件模块和单元管理器 22 中的任意一个来产生，GUI 80 和 TMX 框架 42 协同连续地将根据前面所需记录级别的记录消息传递到图形用户界面，用于在滚动记录消息窗口中显示。此外，还将储存这些记录消息，用于将来的可能的恢复和分析。尽管在正常使用中可以过滤显示给操作员的记录消息，但是其实系统产生的所有消息都储存在单元管理器的主计算机上。最好在系统使用的每一天都产生一个单独的记录文件，并且操作员 10 具有在记录文件获得屏幕 98 中恢复和查看任意给定日期的记录消息的能力。

在操作员 10 选择记录文件目录之后，单元管理器 22 恢复、显示和储存记录文件，如 228 所示。通过选择目录结构查看屏幕 81 顶部附近的“查看”目录条目并选择合适的选项可以访问该屏幕。在从存档屏幕 98 中的记录文件列表中选择每天的记录文件之后，可以显示来自所选记录文件的单独的记录消息，用于在屏幕 98' 上查看，如 230 所示。检查储存的记录消息时，操作员还具有通过选择如上面所讨论的四种过滤级别之一来过滤信息的能力。

图 9 图示说明了根据本发明的一个优选实施例的不同未来内容流路由性能 238。通过操作员在目录查看屏幕上的操作初始驱动对未来事件的详细指定。具体地说，在系统初始化和发现之后，系统会初始地建立一个从当前时刻跨越到未来的预设时刻（例如 2 年）的路由事件。这在时间条 99 中示出。操作员 10 随后可以选择时间条 99，如目录查看屏幕 81 的右上部分所示。出现的弹出菜单允许操作员 10 修改显示的当前事件或创建新事件。在指定未来路由事件的例子中，操作员 10 将通过选择创建新事件选项并且指定新事件的开始和停止时间来创建一个新事件。此时，如 240 所示，单元管理器 22 将（缺省）创建另一个复制的事件。随后将该信息发送到图形用户界面 80，用于显示和进行可能的修改，如 241 所示。提供给操作者 10 的特定编辑器取决于将要创建什么类型的事件。在图 9 的典型实例中，提供了音频和视频编辑器 95 和 95'。也可以为未来路由事件指定 IP 数据流，本领域普通技术人员通过这里所包含的描述将很容易认识到这一点。一旦完成了未来事件的不同部件的所有不同细节，单元管理器 22 就会在 242 收集该信息，并在屏幕 81' 上显示该信息。如屏幕 81' 上所示，在图 9 的典型实例中定义了三个事件。在 244，单元管理器 22 请求将新条目添加到特定 MIB 中，TMX 框架 42 在 246 执行配置改变。单元管理器 22 还会在 252 更新 GUI。由此示出目录结构查看屏幕 81''，其基本上与屏幕 81' 相似，只是根据最新执行的配置显示路由目录结构。

最好不要向 TMX 框架 42 提供这个未来事件配置数据，直到新定义的未来事件开始的预设时刻之前一点。之后（例如大约在预设时刻前的 30 秒），将整个配置数据发送到 TMX 框架 42 用于执行。该路由事件数据与上面结合图 6 讨论的略有不同，因为它还包括预设时间数据，用于指示新路由配置何时发生。这样，操作员 10 可以配置系统在未来的预设时间点上自动改变配置路由控制，甚至不用操作员来配置。因此，系统通过为扩展的时间段预设路由配置信息并且启动这种

配置改变的自动执行，从而允许本发明自动控制宽带媒体硬件。

图 10 图示说明了根据本发明的一个优选实施例的不同 IP 数据封装和插入性能和处理过程 260。如下面更详细地描述，本发明使操作
5 员 10 能够手动或半自动地定义和执行 IP 数据封装。

具体地说，本发明的优选实施例给操作员提供了一个字母接一个字母地手动输入 IP 封装配置数据，或者将图形对象拖放到目录结构查看屏幕 81 的不同位置和从目录结构查看屏幕 81 的不同位置拖放图形
10 对象，从而自动输入 IP 封装配置数据的能力。单元管理器 22 协同图形用户界面 80 一起执行相应拖放操作所指定的不同的路由命令。这可以通过自动组装对应于定义路由命令所需的不同的域的 MIB 表来实现。在图形用户界面 80 上的拖放操作将协助操作员 10 定义系统的 IP 封装规范，其方式基本上类似于图 6 中所示和所述的视频和音频路由
15 事件的半自动定义方式。根据本说明书的描述，本领域普通技术人员将很容易推出怎样扩展这些概念从而实现拖放步骤以获得半自动的 IP 数据封装。下面将立即结合图 10 和 11 描述手动的或者一个字母接一个字母的 IP 数据封装技术。

主要参看图 10，操作员可以通过从目录结构查看屏幕 262 中选择表示所需位置（例如一个开启的输出端口）的图形对象指定一个或多个 IP 数据封装事件 260。操作员 10 随后可以选择特定的程序，在该程序中将插入封装好的 IP 数据。这使得单元管理器 22 在 264 能够
20 获取与目标输出端口及可以在其中驻留的程序有关的配置数据。在图 10 的典型实例中，选择程序 1 用于 IP 数据部件的插入。作为对操作员选择程序 1 的响应，单元管理器 22（在 266）显示程序编辑器 270 并且将缺省输出端口值发送到图形用户界面以显示。操作员 10 随后可以在 272 输入与程序相关的不同的值，在单元管理器 22 的帮助下所述程序中将插入 IP 数据部件。随后将显示通用的和详细的 IP 数据
25 部件编辑器 274，以便操作员 10 可以指定不同的其它参数。操作员 10
30

具有在 IP 数据部件编辑器中为每个包括内容流的部件编辑增/删/改变
详细信息的能力。具体地说，操作员 10 具有在通用和详细编辑器窗
口 274 中为所选程序的每个部件指定细节（诸如源和目标 IP 地址、码
率、目标 PID 等）的能力。这个功能使用户能够在正创建的不同 IP
5 数据流中更加容易地分配带宽，从而获得最大带宽使用率。以这种方式
可以最多同时为 128 个 IP 数据流指定封装和插入。

选择窗口 274 之一的“OK”按钮之后，单元管理器 22 将在 276
执行大量的功能。具体地说，单元管理器 22 从 GUI 收集编辑好的信息
10 并请求将不同的新条目的缺省的和/或编辑好的数据放入到特定的
MIB 表中（如 276 所示）。单元管理器 22 还将该信息提供给 TMX 42
用于执行，如图 10 的 278 和图 11 中所示。具体地说，在 278，SNMP
中介 44 创建新的 MIB 条目，消息控制器 45 将该信息传递到，配置管
理器 46，配置管理器 46 配置一个或多个多路复用器并通知 IP 封装模
15 块 66 开始收集 IP 数据。IP 封装模块 66 随后从指定的源 IP 地址接收
IP 数据，将每个 IP 数据封装为一个或多个 MPEG 包，由此构成 MPEG
数据流，并将其发送到目标多路复用器。目标多路复用器接收组装好
的 MPEG 数据包并合适地发送 MPEG 数据。在 280，单元管理器更新
图形用户界面 80，图形用户界面 80 在目录结构查看屏幕 289 上显示
20 更新后的信息。操作员 10 随后可以查看 IP 数据图标 290，该图标指
示了 IP 数据的封装和插入正在进行。

图 11 中详细说明了框 278 中进行 IP 封装操作的部分。如图所示，
执行完 IP 封装操作 282 之后，封装模块 66 将通知（运行在主处理器
25 上的操作系统的）IP 数据栈在 292 收集/接收并检查 IP 数据包。在 293，
模块 66 随后检验系统是否准备好处理 IP 包（例如是否已经合适地配
置好了目标多路复用器）。用于接收 IP 数据包的目的 IP 地址随后将在
294 测试有效性。具体地说，将检查目的 IP 地址以确定其是否是广播、
单点传送或多点传送 IP 地址。这最好是通过检验目的地址是否在多点
30 传送范围内以及该地址是否已经被指定用于数据收集/接收而完成。如

果 IP 地址显示出该 IP 包不是多点传送包的话，就会判断出该 IP 数据必然是广播或单点传送包。如果这样的话，数据包将以常规的方式通过操作系统（OS）栈传递，然后进行 296 的操作，在这里它将简单地等待接收下一个 IP 数据包。具体地说，优选的 OS（VxWorks）使用了标准的七层 OSI 适应（compliant）IP 栈，其将处理每个广播和/或单点传送包从而确定其类型及其应该处理它的应用程序。因此，例如被发现作为 ARP 请求的广播包将被发送至 ARP 任务进行处理。

相反地，如果源 IP 地址显示出该 IP 数据包是一个多点传送的 IP 包的话，该数据包就不能被简单地通过 OS 栈发送，这是因为，OS 将无法识别该数据包，除非它是该包所需的接收者，然而这是不可能情况。因此，如果 IP 地址显示出该数据包是一个多点传送的包并且如果该地址是单元管理器 22 指定作为与将要封装的 IP 数据相关的 128 个地址之一的话，该 IP 数据将会被转换成不同的形式，而且不需要作为 IP 数据包通过 IP 栈发送。为了完成这种情况，该处理首先进入到 297，在这里，如果需要，IP 数据包会被分段成更小的内容部件进行处理。过程随后传递到 298，在这里组装 MPEG 数据包并将数据包发送到合适的多路复用器。具体地说，将在 300 创建一个 4 字节的 MPEG 帧头，其包括该包的目标 PID。之后，在 302，从 IP 数据包中提取出 IP 地址，并使用该 IP 地址为第一个 MPEG 数据包创建一个 16 字节的 DSM-CC（数据存储媒体命令与控制）帧头。在最后的 MPEG 包中（例如后面的内容的最后一个字节）最好还包括一个常规的 4 字节的循环冗余码（CRC 或 CRC32）MPEG 后缀。由于系统可以支持 DVB 或 ATSC 数据格式的输数据，所以 DSM-CC 帧头还指示了输出数据是何种格式，由此说明这些格式间的差别。在 304，最多可以向正在创建的 MPEG 的 188 字节包中添加 168 字节的内容。如果这包括了所有要发送的内容，那么在内容的最后一个字节之后将附上 CRC。在 308，将判断是否还需要装填数据以完成 MPEG 包。如果需要的话，操作 282 将传递到 310，在这里将使用虚拟空数据填满 MPEG 包的剩余部分。该数据最好数值为 255（十六进制的 FF），并且一直重复直到构成一个完整

的 188 字节 MPEG 数据包。通过本优选实施例的这个系统，将把最大的一个 IP 数据包插入到单独的 MPEG 包中。如果不再需要装填（或者在已经装填了包之后），操作传递到 312，在这里将组装好的包发送到目标多路复用器并优选地存储在 FIFO 中用于组合附加的 MPEG 包（如果有的话）。操作还会传递到 314，在这里确定接收到的 IP 数据包是否已经完全封装。如果是的话，处理传递到 316，在这里多路复用器接收一个指示，其显示已经接收到了多少 MPEG 数据包，多路复用器还将接收到另一个指示，其显示这个/这些包应该被发送。操作 282 随后传递到 296，在这里 IP 封装模块等待下一个要被封装的 IP 数据包。

如果在 314 判断出 IP 数据并没有全部被封装的话，操作 282 就会传递到 318，在这里来自 IP 数据包的额外的内容将被组装进 MPEG 数据包并被发送到合适的多路复用器。具体地说，操作 282 从 314 传递到 320，在这里将会创建下一个 MPEG 数据包的 MPEG 帧头。随后在 322 向包中添加最多可达 184 字节的 IP 数据和 CRC，在 326 将做出判断是否需要装填数据以完成 MPEG 包。如果是的话，操作 282 将传递到 328，在这里将用虚拟空数据装填 MPEG 包的剩余部分。该数据最好数值为 255（十六进制的 FF），并且一直重复直到构成一个完整的 188 字节 MPEG 数据包。如果不再需要装填（或者在已经装填了包之后），操作 282 传递到 330，在这里将组装好的包发送到目标多路复用器并优选地存储在 FIFO 中用于组合以前和后续组装的 MPEG 包（如果有的话）。操作还会传递到 332，在这里确定接收到的 IP 数据包是否已经完全封装。如果否的话，将重复步骤 320 到 332 直到整个 IP 数据包都已经被完全封装，最后，操作传递到 316 和 296，如下所述。如果是的话，在 316，多路复用器会接收一个指示，其显示已经接收到了多少 MPEG 数据包，多路复用器还将接收到另一个指示，其显示这些包应该被发送。操作随后传递到 296，在这里 IP 封装模块等待下一个要被封装的多点传送的 IP 数据包。当操作员 10 为所述主体的输出端口指定不同的功能时，或者当用于制定时间的时间段终止的

时候，操作 282 就会终止。此时，IP 封装模块 66 将等待来自配置管理器 46 的进一步的指令。

下面的计算机程序列表解释了上面提及的 TMX-MIB 的定义：

5

```

TMX-MIB DEFINITIONS : : = BEGIN
    IMPORTS
        NetworkAddress, IpAddress, Gauge, TimeTicks
            FROM RFC1155-SMI
    10      RowStatus, DisplayString, DateAndTime, TEXTUAL-CONVENTION
            FROM SNMPv2-TC
        MODULE-COMPLIANCE, OBJECT-GROUP
            FROM SNMPv2-CONF
        ifIndex
    15      FROM RFC1213-MIB
        Counter32, Integer32, OBJECT-TYPE, MODULE-IDENTITY FROM
        SNMPv2-SMI
        bmr FROM NLS-BBNIDENT-MIB;

    20      tmxMIB MODULE-IDENTITY
        LAST-UPDATED      "0112202341Z"
        ORGANIZATION"Motorola BCS"
        CONTACT-INFO"John Sanders, Ext 3505 4A31"
        DESCRIPTION
    25      "TMX MIB Version 1. 1a"
        REVISION      "0008141355Z"
        DESCRIPTION
            ""
        : : = {bsr 1 }

    30      ActionTriggerType : : = TEXTUAL-CONVENTION
        STATUS current
        DESCRIPTION
            "Type of trigger mechanism for execution of
    35      an operation"
        SYNTAX BITS {
            operationCreation(0),
            activationTime(1),
            spliceImmediateFlag (2)
    40      )

        org OBJECT IDENTIFIER
            : : = (iso 3)
        dod OBJECT IDENTIFIER
    45      : : = {org 6 }
        internet OBJECT IDENTIFIER
            : : = {dod 1}
        private OBJECT IDENTIFIER
            : : = {internet 4}
    50      enterprises OBJECT IDENTIFIER
            : : = {private 1}
        tmxMIBObjectsOBJECT IDENTIFIER
            : : = {tmxMIB 1 }
        tmxMIBConformance OBJECT IDENTIFIER
    55      : : = {tmxMIB 2}
        tmxMediaInterfaces OBJECT IDENTIFIER

```

```

      : : = {tmxMIBObjects 1}
tmxMediaControl OBJECT IDENTIFIER
      : : = {tmxMIBObjects 2}
tmxMediaMonitor OBJECT IDENTIFIER
5      : : = {tmxMIBObjects 3}

-- TMX specific extensions to the MIB 2 Interfaces Table
tmxNetTable OBJECT-TYPE
10      SYNTAX SEQUENCE OF TmxNetEntry
      MAX-ACCESS not-accessible
      STATUS current
      DESCRIPTION
          "This is the network table associating TMX interfaces
15      (ifIndex)
          with other types of equipment."
      : : = {tmxMediaInterfaces 1}

tmxNetEntry OBJECT-TYPE
20      SYNTAX TmxNetEntry
      MAX-ACCESS not-accessible
      STATUS current
      DESCRIPTION "tmxNetTable entry"
      INDEX {ifIndex, tmxNetIndex}
25      : : = {tmxNetTable 1}

TmxNetEntry.. SEQUENCE {
      tmxNetIndex Integer32,
      tmxNetEqpType BITS,
30      tmxNetEqpName DisplayString,
      tmxNetEqpIpAddress IpAddress,
      tmxNetEqpOperationalState Integer32,
      tmxNetEqpAlarmStatus Integer32,
      tmxNetEqpHeartBeatMonitor IpAddress,
35      tmxNetEqpRedundancyGroup DisplayString,
      tmxNetEqpBackUp IpAddress,
      tmxNetEntryStatus RowStatus
    )

40
tmxNetIndex OBJECT-TYPE
      SYNTAX Integer32
      MAX-ACCESS read-create
      STATUS current
45      DESCRIPTION
          "Identifier index for this networked equipment."
      : : = {tmxNetEntry 1}

tmxNetEqpType OBJECT-TYPE
50      SYNTAX BITS {se1000 (0), se2000 (1), tmx (2),
      dataServer (3), adServer (4), computer (5)}
      MAX-ACCESS read-create
      STATUS current
      DESCRIPTION
55      "Type of equipment adjacent on ifIndex interface."
      : : = {tmxNetEntry 2}

tmxNetEqpName OBJECT-TYPE
      SYNTAX DisplayString
60      MAX-ACCESS read-create
      STATUS current
      DESCRIPTION

```

```

        "Unique user assigned name to the physical equipment."
        : : = {tmxNetEntry 3}

tmxNetEqpIpAddress OBJECT-TYPE
5      SYNTAX IpAddress
      MAX-ACCESS read-create
      STATUS current
      DESCRIPTION "IP Address of the adjacent equipment."
      : : = {tmxNetEntry 4}
10

tmxNetEqpOperationalState OBJECT-TYPE
      SYNTAX Integer32
      MAX-ACCESS read-create
      STATUS current
15     DESCRIPTION
        "OperationalState of this networked equipment."
        : : = {tmxNetEntry 5}

tmxNetEqpAlarmStatus OBJECT-TYPE
20     SYNTAX Integer32
      MAX-ACCESS read-create
      STATUS current
      DESCRIPTION
25     "AlarmStatus of this networked equipment."
        : : = {tmxNetEntry 6}

tmxNetEqpHeartBeatMonitor OBJECT-TYPE
      SYNTAX IpAddress
      MAX-ACCESS read-create
30     STATUS current
      DESCRIPTION
        "Heart beat monitor for this networked equipment."
        : : = {tmxNetEntry 7}

35     tmxNetEqpRedundancyGroup OBJECT-TYPE
      SYNTAX DisplayString
      MAX-ACCESS read-create
      STATUS current
      DESCRIPTION
40     "Unique name of the redundancy group of this networked
        equipment."
        : : = {tmxNetEntry 8}

tmxNetEqpBackUp OBJECT-TYPE
45     SYNTAX IpAddress
      MAX-ACCESS read-create
      STATUS current
      DESCRIPTION
50     "IP Address of the back up of this networked equipment."
        : : = {tmxNetEntry 9}

tmxNetEntryStatus OBJECT-TYPE
      SYNTAX RowStatus
      MAX-ACCESS read-create
55     STATUS current
      DESCRIPTION "Status of this entry."
        : : = {tmxNetEntry 10}

60     -- tmxMediaControl consists of tables used to configure the TMX media
      -- control functionality.

```

```

tmxOpNum OBJECT-TYPE
    SYNTAX INTEGER
    MAX-ACCESS read-create
    STATUS current
5    DESCRIPTION "The number of last Op created."
    ::= {tmxMediaControl 1}

tmxToaLock OBJECT-TYPE
    SYNTAX BITS {unlock (0), lock (1)}
10    MAX-ACCESS read-create
    STATUS current
    DESCRIPTION "The number of last Op created."
    ::= {tmxMediaControl 2}

15    tmxToaLockTime OBJECT-TYPE
        SYNTAX INTEGER
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
20        "The time (in second) to wait before starts executing the
        unlock operation"
        ::= {tmxMediaControl 3}

25    tmxDeleteAll OBJECT-TYPE
        SYNTAX BITS (delete (0), keep (1))
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
30        "The flag indicates the deletion of the entire media control
        database"
        ::= {tmxMediaControl 4}

tmxReserved OBJECT-TYPE
    SYNTAX INTEGER
35    MAX-ACCESS read-create
    STATUS current
    DESCRIPTION "The number of last Op created."
    ::= {tmxMediaControl 5}
*****

40    tmxIfTable OBJECT-TYPE
        SYNTAX SEQUENCE OF TmxIfEntry
        MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
45        "This is the TMX specific extension of ifTable."
        ::= {tmxMediaControl 6}

tmxIfEntry OBJECT-TYPE
    SYNTAX TmxIfEntry
50    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION "tmxIfTable entry"
    INDEX {ifIndex}
    ::= {tmxIfTable 1}

55    TmxIfEntry ::= SEQUENCE {
        tmxIfName DisplayString,
        tmxIfTSid Integer32,
        tmxIfType BITS,
60        tmxIfASIMode BITS,
        tmxIfStandard BITS,
        tmxIfTransportBitRate Integer32,

```

```

        tmxIfActionTrigger ActionType,
        tmxIfActionTime DateAndTime,
        tmxIfIQ BITS,
        tmxIfSttDestinationOffset Integer32,
5      tmxIfSttDsStatus BITS,
        tmxIfSttDsDay Integer32,
        tmxIfSttDsHour Integer32,
        tmxIfPacketLength BITS,
        tmxIfEntryStatus RowStatus
10    }
    *****
    tmxIfName OBJECT-TYPE
        SYNTAX DisplayString
        MAX-ACCESS read-create
15      STATUS current
        DESCRIPTION
        "User assigned name to the physical interface."
        : : = {tmxIfEntry 1}

20    tmxIfTSId OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
25      "Mpeg Transport Identifier associated with this physical
        interface, if any."
        : : = {tmxIfEntry 2}

    tmxIfType OBJECT-TYPE
30      SYNTAX BITS {asi (0), ds3-ansi (1), ds3-fsi (2),
        dhe1 (3), dhe1-high (4),
        smpte310 (5), ds3-rmi (6)}
        MAX-ACCESS read-create
        STATUS current
35      DESCRIPTION "tmxIfType."
        : : = {tmxIfEntry 3}

    tmxIfASIMode OBJECT-TYPE
        SYNTAX BITS {burst (0), byte (1)}
40      MAX-ACCESS read-create
        STATUS current
        DESCRIPTION "Output ASI mode, burst or byte."
        : : = {tmxIfEntry 4}

45    tmxIfStandard OBJECT-TYPE
        SYNTAX BITS {atsc (0), dvb (1), mpeg2 (2), none (3), dcii (4)}
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION "Output ASI standard."
50      : : = {taxIfEntry 5}

    tmxIfTransportBitRate OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS read-create
55      STATUS current
        DESCRIPTION "Transport bit rate in Mbps."
        : : = (tmxIfEntry 6)

    tmxIfActionTrigger OBJECT-TYPE
60      SYNTAX ActionType
        MAX-ACCESS read-create
        STATUS current

```

```

        DESCRIPTION
        "Trigger type that will initiate this operation."
        : : = {tmxIfEntry 7}

5      tmxIfActionTime OBJECT-TYPE
        SYNTAX DateAndTime
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
10     "If trigger type is a time of action this object contains
        the time."
        ::= {tmxIfEntry 8}

        tmxIfIQ OBJECT-TYPE
15     SYNTAX BITS {combined (0), i-out (1), q-out (2), iq-split (3)}
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
20     "Select IQ mux mode for this transport stream, combined, I,
        Q or I & Q."
        ::= {tmxIfEntry 9}

        tmxIfSttDestinationOffset OBJECT-TYPE
25     SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
        "Offset from UTC as used in the STT. This may not be the same
30     value
        as used by the TMX system and may vary between transport streams
        depending on
        destination"
        : : = {tmxIfEntry 10}

35     tmxIfSttDsStatus OBJECT-TYPE
        SYNTAX BITS {not-in-ds (0), in-ds (1)}
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION "Daylight savings status as used in the STT."
40     : : = {tmxIfEntry 11}

        tmxIfSttDsDay OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS read-create
45     STATUS current
        DESCRIPTION
        "Local day of month daylight savings status will change (1-
31) as
        used in the STT."
50     : : = {tmxIfEntry 12}

        tmxIfSttDsHour OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS read-create
55     STATUS current
        DESCRIPTION
        "Local hour of day daylight savings status will change (0-18)
as
        used in the STT."
60     : : = {tmxIfEntry 13}

        tmxIfPacketLength OBJECT-TYPE

```

```

        SYNTAX BITS {pkt188 (0), pkt204 (1)}
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION "Transport stream packet length."
5       : : = {tmxIfEntry 14}

        tmxIfEntryStatus OBJECT-TYPE
        SYNTAX RowStatus
        MAX-ACCESS read-create
10      STATUS current
        DESCRIPTION "Status of this entry."
        : : = {tmxIfEntry 15}

*****

15      ImxITGiExtTable OBJECT-TYPE
        SYNTAX SEQUENCE OF TmxIfGiExtEntry
        MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
20      "This is the extension table defining the DS3-GI TMX interface
        parameters."
        : : = {tmxMediaControl 7}

        tmxIfGiExtEntry OBJECT-TYPE
25      SYNTAX TmxIfGiExtEntry
        MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
        "tmxIfGiExtTable entry"
30      INDEX {ifIndex}
        : : = (tmxIfGiExtTable 1)

        TmxIfGiExtEntry : : = SEQUENCE (
        tmxIfGiExtCodeRate BITS,
35      tmxIfGiExtSymbolRate BITS,
        tmxIfGiExtModType BITS,
        tmxIfGiExtPowerLevel Integer32,
        tmxIfGiExtCarrierFrq Integer32,
        tmxIfGiExtCwMode BITS,
40      tmxIfGiExtMute BITS,
        tmxIfGiExtEntryStatus RowStatus
        )

*****

45      tmxIfGiExtCodeRate OBJECT-TYPE
        SYNTAX BITS {cr-5-11 (0), cr-1-2 (1), cr-3-5 (2),
        cr-2-3 (3), cr-3-4 (4), cr-4-5 (5),
        cr-5-6 (6), cr-7-8 (7)}
        MAX-ACCESS read-create
        STATUS current
50      DESCRIPTION
        "Code rate: 5/11, 1/2, 3/5, 2/3, 3/4, 4/5, 5/6,
        7/8"
        : : = {tmxIfGiExtEntry 1}

55      tmxIfGiExtSymbolRate OBJECT-TYPE
        SYNTAX BITS {sr-38-8 (0), sr-29-27 (1),
        sr-19-51 (2), sr-14-63 (3),
        sr-11-71 (4), sr-9-76 (5),
        sr-7-32 (6), sr-4-88 (7),
60      sr-3-25 (8), sr-2-44 (9),
        sr-1-83 (10)}
        MAX-ACCESS read-create

```

```

STATUS current
DESCRIPTION
"Symbol rate: 38.8 29.27,19.51,14.63,11.71,
5      9.76,7.32,4.88,3.25,2.44,1.83"
: : = {tmxIfGiExtEntry 2}

tmxIfGiExtModType OBJECT-TYPE
SYNTAX BITS {qpsk (0), bpsk (1), oqpsk (2)}
MAX-ACCESS read-create
10      STATUS current
DESCRIPTION"Modulation type."
: : = {tmxIfGiExtEntry 3}

tmxIfGiExtPowerLevel OBJECT-TYPE
15      SYNTAX Integer32
MAX-ACCESS read-create
STATUS current
DESCRIPTION
20      "Power level: 0-20 representing-5dBm to-15dBm in 0. 5dBm
steps."
: : = {tmxIfGiExtEntry 4}

tmxIfGiExtCarrierFrq OBJECT-TYPE
25      SYNTAX Integer32
MAX-ACCESS read-create
STATUS current
DESCRIPTION
30      "47000 kHz-93000 kHz for the 70 MHz band and
104000 khz-176000 kHz for the 140 MHz band.
The frequency must be in steps of 125 kHz."
: : = {tmxIfGiExtEntry 5}

tmxIfGiExtCwMode OBJECT-TYPE
35      SYNTAX BITS {modulated (0), unmodulated (1)}
MAX-ACCESS read-create
STATUS current
DESCRIPTION"IF carrier to be unmodulated."
: : = {tmxIfGiExtEntry 6}

40      tmxIfGiExtMute OBJECT-TYPE
SYNTAX BITS {notMuted (0), muted (1)}
MAX-ACCESS read-create
STATUS current
DESCRIPTION"IF carrier to be muted."
45      : : = {tmxIfGiExtEntry 7}

tmxIfGiExtEntryStatus OBJECT-TYPE
SYNTAX RowStatus
MAX-ACCESS read-create
50      STATUS current
DESCRIPTION"Status of this entry."
::= {tmxIfGiExtEntry 8}

*****
55      tmxStat ; GroupTable,, ; : : OBJECT-TYPE,
SYNTAX SEQUENCE OF TmxStatGroupEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
60      "The StatGroup Control table. Each row in this
table represents a create/delete Op command. All
Ops with the same Time of Action are to be executed

```

```

        in the order specified by the message index.
        Program elements within a StatGroup can be
        Constant Bit Rate (CBR) in which case their minBw=maxBw,
        Variable Bit Rate (VBR) in which case their minBw < maxBw,
        and opportunistic in which case they use up only spare Bw
        left unused after all CBR and VBR programs are multiplexed"
        : : = (tmxMediaControl 8}

5
tmxStatGroupEntry OBJECT-TYPE
10
    SYNTAX TmxStatGroupEntry
    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "StatGroupTable entry"
15
    INDEX {ifIndex, tmxStatGroupOpIndex}
    : : = {tmxStatGroupTable 1}

TmxStatGroupEntry : : = SEQUENCE {
20
    tmxStatGroupOpIndex Integer32,
    tmxStatGroupMsg BITS,
    tmxStatGroupId Integer32,
    tmxStatGroupSize Integer32,
    tmxStatGroupTranscBwPercent Integer32,
    tmxStatGroupMaxInputBw Integer32,
25
    tmxStatGroupMaxOutputBw Integer32,
    tmxStatGroupActionTrigger ActionTriggerType,
    tmxStatGroupActionTime DateAndTime,
    tmxStatGroupStatus RowStatus
    }
30
*****
tmxStatGroupOpIndex OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS not-accessible
    STATUS current
35
    DESCRIPTION
        "This is used for unique identification and sequencing of all
        operational configuration messages. The value is the value of
        tmxOpNum when entry was created and is set by the manager."
        : : = {tmxStatGroupEntry 1}
40

tmxStatGroupMsg OBJECT-TYPE
    SYNTAX BITS {create (0), delete (1)}
    MAX-ACCESS read-create
    STATUS current
45
    DESCRIPTION
        "Each table row constitutes a command message. The message
        type indicates whether the Command type is to either create
        or delete the item described in this row. Any create command
        is further restricted by the Action Trigger Type. Depending on
50
        the Action Trigger Type, the command will either be immediately
        executed, executed at a time specified by Time of Action, or
        executed when a SPliceImmediateFlag is raised."
        : : = (tmxStatGroupEntry 2)

55
tmxStatGroupId OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS read-create
    STATUS current
    DESCRIPTION
60
    "Group identifier in control multiplex."
    : : = {tmxStatGroupEntry 3}

```

```

tmxStatGroupSize OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS read-create
    STATUS current
5    DESCRIPTION "Number of members in group."
    ::= {tmxStatGroupEntry 4}

tmxStatGroupTranscBwPercent OBJECT-TYPE
    SYNTAX Integer32
10    MAX-ACCESS read-create
    STATUS current
    DESCRIPTION
        "Percentage of overall bandwidth to allocate to multiplex group,
        if transcoded."
15    ::= {tmxStatGroupEntry 5}

tmxStatGroupMaxInputBw OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS read-create
20    STATUS current
    DESCRIPTION
        "Define maximum bandwidth to allocate to this input stat group,
        if stat group is input."
25    ::= {tmxStatGroupEntry 6}

tmxStatGroupMaxOutputBw OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS read-create
30    STATUS current
    DESCRIPTION
        "Define maximum bandwidth to allocate to this output stat group,
        if stat group is input."
35    ::= {tmxStatGroupEntry 7}

tmxStatGroupActionTrigger OBJECT-TYPE
    SYNTAX ActionTriggerType
    MAX-ACCESS read-create
    STATUS current
40    DESCRIPTION
        "Trigger type that will initiate this operation."
        .. (tStatGroupEntry 8)

tmxStatGroupActionTime OBJECT-TYPE
    SYNTAX DateAndTime
45    MAX-ACCESS read-create
    STATUS current
    DESCRIPTION
        "If trigger type is a time of action this object contains the
        time."
50    ::= {tmxStatGroupEntry 9}

tmxStatGroupStatus OBJECT-TYPE
    SYNTAX RowStatus
    MAX-ACCESS read-create
55    STATUS current
    DESCRIPTION "Status of this row."
    ::= {tmxStatGroupEntry 10}

*****
60    tmxPvrgTable-OB^JECTx ThP
    SYNTAX SEQUENCE OF TmxProgEntry
    MAX-ACCESS not-accessible

```

```

STATUS current
DESCRIPTION
    "The MPEG Service/Program Control table. Each row in this
5      table represents a create/delete Op command. All
      Ops with the same Time of Action are to be executed
      in the order specified by the message index.
      This table is used to create programs in the output
      multiplex and to associate programs with possibly
10     external input sources. It is not used to route a
      complete program from input to output. All routing is done
      by the Program Component Table."
      ::= {tmxMediaControl 9}

tmxProgEntry OBJECT-TYPE
15     SYNTAX TmxProgEntry
      MAX-ACCESS not-accessible
      STATUS current
      DESCRIPTION
          "Entry into the Program Control table."
20     INDEX {ifIndex, tmxStatGroupOpIndex, tmxProgOpIndex}
      : : = {tmxProgTable 1}

TmxProgEntry : : = SEQUENCE {
25     tmxProgOpIndex Integer32,
      tmxProgMsg BITS,
      tmxProgGrpId Integer32,
      tmxProgEncIPAddr IpAddress,
      tmxProgEncPort Integer32,
      tmxProgTmxIPAddr IpAddress,
30     tmxProgTmxPort Integer32,
      tmxProgProgId Integer32,
      tmxProgProgName DisplayString,
      tmxProgPcrPID Integer32,
      tmxProgGIDS3IQ BITS,
35     tmxProgActionTrigger ActionTriggerType,
      tmxProgActionTime DateAndTime,
      tmxProgStatus RowStatus
      )
*****
40     tmxProgOpIndex OBJECT-TYPE
      SYNTAX Integer32
      MAX-ACCESS not-accessible
      STATUS current
      DESCRIPTION
45     "This is used for unique identification and sequencing of all
      operational configuration messages. The value is the value
      of
      tmxOpNum when entry was created and is set by the manager."
      : : = {tmxProgEntry 1}
50

tmxProgMsg OBJECT-TYPE
      SYNTAX BITS (create (0), delete (1))
      MAX-ACCESS read-create
      STATUS current
55     DESCRIPTION
          "Each table row constitutes a command message. The message
          type indicates whether the Command type is to either create
          or delete the item described in this row. Any create command
          is further restricted by the Action Trigger Type. Depending on
60     the Action Trigger Type, the command will either be immediately
          executed, executed at a time specified by Time of Action, or
          executed when a SPlliceImmediateFlag is raised."

```

```

        ::= {tmxProgEntry 2}

tmxProgGrpId OBJECT-TYPE
    SYNTAX Integer32
5      MAX-ACCESS read-create
    STATUS current
    DESCRIPTION
        "Statistical Group identifier in output multiplex,
10     if part of a statistical group."
    ::= {tmxProgEntry 3}

tmxProgEncIPAddr OBJECT-TYPE
    SYNTAX IpAddress
15     MAX-ACCESS read-create
    STATUS current
    DESCRIPTION
        "If any, IP address of the encoder for this service for
        distributed stat mux."
20     ::= {tmxProgEntry 4}

tmxProgEncPort OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS read-create
25     STATUS current
    DESCRIPTION
        "If any, Port number of the encoder for this service for
        distributed stat mux."
    ::= {tmxProgEntry 5}

30     tmxProgTmxIPAddr OBJECT-TYPE
        SYNTAX IpAddress
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
35         "If any, IP address of the TMX for this service for distributed
        stat mux."
        ::= {tmxProgEntry 6}

tmxProgTmxPort OBJECT-TYPE
40     SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
45         "If any, Port number of the TMX for this service for distributed
        stat mux."
        ::= {tmxProgEntry 7}

tmxProgProgId OBJECT-TYPE
    SYNTAX Integer32
50     MAX-ACCESS read-create
    STATUS current
    DESCRIPTION
        "Output MPEG program (i. e. service) number."
    ::= {tmxProgEntry 8}
55

tmxProgProgName OBJECT-TYPE
    SYNTAX DisplayString
    MAX-ACCESS read-create
60     STATUS current
    DESCRIPTION
        "Output program name assigned by the user."
    ::= {tmxProgEntry 9}

```

```

tmxProgPcrPID OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS read-create
5     STATUS current
    DESCRIPTION
        "PCR PID associated with the program."
        : : = {tmxProgEntry 10}

10    tmxProgGIDS3IQ OBJECT-TYPE
        SYNTAX BITS (splitI (0), splitQ (1), combined (2))
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
15         "Each table row constitutes a command message."
        : : = {tmxProgEntry 11}

    tmxProgActionTrigger OBJECT-TYPE
        SYNTAX ActionTriggerType
20         MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
            "Trigger type that will initiate this operation."
            : : = {tmxProgEntry 12}

25    tmxProgActionTime OBJECT-TYPE
        SYNTAX DateAndTime
        MAX-ACCESS read-create
        STATUS current
30         DESCRIPTION
            "If trigger type is a time of action this object contains the
            time."
            : : = {tmxProgEntry 13}

35    tmxProgStatus OBJECT-TYPE
        SYNTAX RowStatus
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
40         "Status of the row."
        : : = {tmxProgEntry 14}
*****
    tmxProgCompTable OBJECT-TYPE
        SYNTAX SEQUENCE OF TmxProgCompEntry
45         MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
            "The MPEG Service/Program Component Control table. Each row in
this
50         table represents a create/delete Op command. All
            Ops with the same Time of Action are to be executed
            in the order specified by the message index."
            : : = {tmxMediaControl 10}

55    tmxProgCompEntry OBJECT-TYPE
        SYNTAX TmxProgCompEntry
        MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION "Entry"
60         INDEX{ifIndex, tmxStatGroupOpIndex, tmxProgOpIndex,
            tmxProgCompOpIndex}
            : : = {tmxProgCompTable 1}

```

```

TmxProgCompEntry.. SEQUENCE {
    tmxProgCompOpIndex Integer32,
    tmxProgCompMsg BITS,
    5    tmxProgCompTpeId Integer32,
    tmxProgCompSrcId Integer32,
    tmxProgCompPID Integer32,
    tmxProgCompAliasPID Integer32,
    tmxProgCompStreamType BITS,
    10    tmxProgCompPriority Integer32,
    tmxProgCompTrcMode BITS,
    tmxProgCompEmbeddedPCR BITS,
    tmxProgCompActionTrigger ActionTriggerType,
    15    tmxProgCompActionTime DateAndTime,
    tmxProgCompMinBw Integer32,
    tmxProgCompMaxBw Integer32,
    tmxProgCompStatus RowStatus
}
*****
20    tmxProgCompOpIndex OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
    25    "This is used for unique identification and sequencing of all
        operational configuration messages. The value is the value of
        tmxOpNum when entry was created and is set by the manager."
        : : = {tmxProgCompEntry 1}

    30    tmxProgCompMsg OBJECT-TYPE
        SYNTAX BITS {create (0), delete (1)}
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
    35    "Each table row constitutes a command message. The message
        type indicates whether the Command type is to either create
        or delete the item described in this row. Any create command
        is further restricted by the Action Trigger Type. Depending on
        the Action Trigger Type, the command will either be immediately
    40    executed, executed at a time specified by Time of Action, or
        executed when a SPlliceImmediateFlag is raised."
        ::= {tmxProgCompEntry 2}

    tmxProgCompTpeId OBJECT-TYPE
    45    SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
        "Identifier of the TPE to process this component,
    50    if any."
        : : = {tmxProgCompEntry 3}

    tmxProgCompSrcId OBJECT-TYPE
    55    SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
        "Physical input source identifier (index)"
        : : = {tmxProgCompEntry 4}

    60    tmxProgCompPID OBJECT-TYPE
        SYNTAX Integer32

```

```

MAX-ACCESS read-create
STATUS current
DESCRIPTION
"MPEG Input PID"
5 ::= {tmxProgCompEntry 5}

tmxProgCompAliasPID OBJECT-TYPE
SYNTAX Integer32
MAX-ACCESS read-create
10 STATUS current
DESCRIPTION
"User assigned Component PID number in the output multiplex
to create or delete."
: : = {tmxProgCompEntry 6}

15 tmxProgCompStreamType OBJECT-TYPE
SYNTAX BITS {video (0), audio (1), data (2)}
MAX-ACCESS read-create
STATUS current
20 DESCRIPTION
"Stream content type of the MPEG component to be
created or deleted."
: : = {tmxProgCompEntry 7}

25 tmxProgCompPriority OBJECT-TYPE
SYNTAX Integer32
MAX-ACCESS read-create
STATUS current
DESCRIPTION
30 "This is a value between 1 (lowest) and 10 and indicates the
minimum video quality (quantization level) after which
opportunistic data can be added to the statistical
multiplex group. Alternatively, if data within the group
is guaranteed the this indicates the relative priority
35 of the video within the group."
::= {tmxProgCompEntry 8}

tmxProgCompTrcMode OBJECT-TYPE
SYNTAX BITS (transcode (0), passthru (1))
40 MAX-ACCESS read-create
STATUS current
DESCRIPTION
"Transcode or passthrough this component."
: : = {tmxProgCompEntry 9}

45 tmxProgCompEmbeddedPCR OBJECT-TYPE
SYNTAX BITS (enable (0), disable (1))
MAX-ACCESS read-create
STATUS current
50 DESCRIPTION "PCR mode."
: : = {tmxProgCompEntry 10}

tmxProgCompActionTrigger OBJECT-TYPE
SYNTAX ActionTriggerType
55 MAX-ACCESS read-create
STATUS current
DESCRIPTION
"Trigger type that will initiate this operation."
::= {tmxProgCompEntry 11}

60 tmxProgCompActionTime OBJECT-TYPE
SYNTAX DateAndTime

```

```

MAX-ACCESS read-create
STATUS current
DESCRIPTION
5  "If trigger type is a time of action this object contains the
   time."
   ::= {tmxProgCompEntry 12}

tmxProgCompMinBw OBJECT-TYPE
10  SYNTAX Integer32
   MAX-ACCESS read-create
   STATUS current
   DESCRIPTION
   "Minimum bandwidth for program component.
15  If minimum = maximum the bandwidth is fixed, i. e. not
   statistically multiplexed."
   ::= {tmxProgCompEntry 13}

tmxProgCompMaxBw OBJECT-TYPE
20  SYNTAX Integer32
   MAX-ACCESS read-create
   STATUS current
   DESCRIPTION
   "Maximum bandwidth for program component.
25  If minimum = maximum the bandwidth is fixed, i. e. not
   statistically multiplexed."
   ::= {tmxProgCompEntry 14}

tmxProgCompStatus OBJECT-TYPE
30  SYNTAX RowStatus
   MAX-ACCESS read-create
   STATUS current
   DESCRIPTION
   "Status of the row."
   ::= {tmxProgCompEntry 15}
35  *****

tmxIPDataTable OBJECT-TYPE
   SYNTAX SEQUENCE OF TmxIPDataEntry
   MAX-ACCESS not-accessible
40  STATUS current
   DESCRIPTION
   "The IP Data Component Control table. Each row in this
   table represents a create/delete Op command. All
   Ops with the same Time of Action are to be executed
45  in the order specified by the message index."
   ::= {tmxMediaControl 11}

tmxIPDataEntry OBJECT-TYPE
   SYNTAX TmxIPDataEntry
   MAX-ACCESS not-accessible
50  STATUS current
   DESCRIPTION
   "Entry"
   INDEX {ifIndex, tmxStatGroupOpIndex, tmxProgOpIndex,
55  tmxIPDataOpIndex}
   ::= (tmxIPDataTable 1)

TmxIPDataEntry ::= SEQUENCE {
   tmxIPDataOpIndex Integer32,
   tmxIPDataMsg BITS,
60  tmxIPDataPID Integer32,
   tmxIPDataDestAddr IpAddress,
   tmxIPDataFlowCtlType BITS,

```

```

tmxIPDataTmxFlowCtlServerAddr IpAddress,
tmxIPDataTmxFlowCtlServerPort Integer32,
tmxIPDataTmxFlowCtlAddr IpAddress,
tmxIPDataTmxFlowCtlPort Integer32,
5  tmxIPDataMuxModeBITS,
  tmxIPDataOutMode BITS,
  tmxIPDataAvgBitRate Integer32,
  tmxIPDataAvgTimePeriod Integer32,
  tmxIPDataMinBitRate Integer32,
10  tmxIPDataMaxBitRate Integer32,
  tmxIPDataActionTrigger ActionTriggerType,
  tmxIPDataActionTime DateAndTime,
  tmxIPDataSectionLength BITS,
  tmxIPDataStatus RowStatus
15  }
*****
tmxIPDataOpIndex OBJECT-TYPE
  SYNTAX Integer32
  MAX-ACCESS not-accessible
20  STATUS current
  DESCRIPTION
    "This is used for unique identification and sequencing of all
    operational configuration messages. The value is the value of
    tmxOpNum when entry was created and is set by the manager."
25  : : = {tmxIPDataEntry 1}

tmxIPDataMsg OBJECT-TYPE
  SYNTAX BITS {create (0), delete (1)}
  MAX-ACCESS read-create
30  STATUS current
  DESCRIPTION
    "Each table row constitutes a command message. The message
    type indicates whether the Command type is to either create
    or delete the item described in this row. Any create command
35  is further restricted by the Action Trigger Type. Depending on
    the Action Trigger Type, the command will either be immediately
    executed, executed at a time specified by Time of Action, or
    executed when a SPliceImmediateFlag is raised."
    : : = {tmxIPDataEntry 2}
40

tmxIPDataPID OBJECT-TYPE
  SYNTAX Integer32
  MAX-ACCESS read-create
  STATUS current
45  DESCRIPTION
    "MPEG PID number in output transport for this data stream.
    There may be multiple IP data route entries for the same PID!!!"
    : : = {tmxIPDataEntry 3}

50  tmxIPDataDestAddr OBJECT-TYPE
    SYNTAX IpAddress
    MAX-ACCESS read-create
    STATUS current
    DESCRIPTION
55  "IP address of destination host or network that this PID will
    route data for."
    : : = {tmxIPDataEntry 4}

tmxIPDataFlowCtlType OBJECT-TYPE
60  SYNTAX BITS (none (0), smpte-325m (1))
  MAX-ACCESS read-create
  STATUS current

```

```

DESCRIPTION
"The type of flow control used by the TMX to throttle the data
server"
::= {tmxIPDataEntry 5}

5
tmxIPDataTmxFlowCtlServerAddr OBJECT-TYPE
SYNTAX IPAddress
MAX-ACCESS read-create
STATUS current
10
DESCRIPTION
"Server IP Address for flow control of this data stream."
::= {tmxIPDataEntry 6}

tmxIPDataTmxFlowCtlServerPort OBJECT-TYPE
15
SYNTAX Integer32
MAX-ACCESS read-create
STATUS current
DESCRIPTION
20
data "Port number of the server's IP port for flow control of this
stream."
::= {tmxIPDataEntry 7}

tmxIPDataTmxFlowCtlAddr OBJECT-TYPE
25
SYNTAX IPAddress
MAX-ACCESS read-create
STATUS current
DESCRIPTION
30
"TMX IP Address for flow control of this data stream."
::= {tmxIPDataEntry 8}

tmxIPDataTmxFlowCtlPort OBJECT-TYPE
SYNTAX Integer32
MAX-ACCESS read-create
35
STATUS current
DESCRIPTION
"Port number of TMX's IP flow control port for this data stream."
::= {tmxIPDataEntry 9}

40
tmxIPDataMuxMode OBJECT-TYPE
SYNTAX BITS {guaranteed (0), opportunistic (1)}
MAX-ACCESS read-create
STATUS current
45
DESCRIPTION
"Data multiplexing mode. Either guaranteed, in which case either
sufficiently much multiplexing bandwidth is allocated or the
video quality will be sacrificed in a statistical multiplex;
or opportunistic in which case data multiplexing is dependent
on spare bandwidth after desired video quality is achieved"
50
::= (tmxIPDataEntry 10)

tmxIPDataOutMode OBJECT-TYPE
SYNTAX BITS {atsc (0), dvb (1), other (2)}
MAX-ACCESS read-create
55
STATUS current
DESCRIPTION
"IP data encapsulation."
::= (tmxIPDataEntry 11)

60
tmxIPDataAvgBitRate OBJECT-TYPE
SYNTAX Integer32
MAX-ACCESS read-create

```

```

STATUS current
DESCRIPTION
"Average data bit rate over tmxIPDataAvgTimePeriod, if any.
: : = {tmxIPDataEntry 12}

5      tmxIPDataAvgTimePeriod OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
10      DESCRIPTION
        "Averaging time period in Seconds, if any."
        : : = {tmxIPDataEntry 13}

        tmxIPDataMinBitRate OBJECT-TYPE
15      SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
20      "Minimum guaranteed or desired bit rate over
        tmxIPDataAvgTimePeriod,
        if any."
        : : = {tmxIPDataEntry 14}

        tmxIPDataMaxBitRate OBJECT-TYPE
25      SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
30      "Maximum guaranteed or desired bit rate over
        tmxIPDataAvgTimePeriod,
        if any."
        : : = {tmxIPDataEntry 15}

        tmxIPDataActionTrigger OBJECT-TYPE
35      SYNTAX ActionTriggerType
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
40      "Trigger type that will initiate this operation."
        : : = {tmxIPDataEntry 16}

        tmxIPDataActionTime OBJECT-TYPE
        SYNTAX DateAndTime
        MAX-ACCESS read-create
45      STATUS current
        DESCRIPTION
        "If trigger type is a time of action this object contains the
        time."
        : : = {tmxIPDataEntry 17}

50      tmxIPDataSectionLength OBJECT-TYPE
        SYNTAX BITS {len4096 (0), len1024 (1)}
        MAX-ACCESS read-create
        STATUS current
55      DESCRIPTION "Maximum length of DSM-CC sections."
        : : = {tmxIPDataEntry 18}

        tmxIPDataStatus OBJECT-TYPE
        SYNTAX RowStatus
60      MAX-ACCESS read-create
        STATUS current
        DESCRIPTION

```

```

        "Status of the row."
        : : = {tmxIPDataEntry 19}

*****
5      tmxAsyncDataTable OBJECT-TYPE
        SYNTAX SEQUENCE OF TmxAsyncDataEntry
        MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
10     "The RS-232/RS-422 Data Component Control table. Each row in
        this
        table represents a create/delete Op command. All
        Ops with the same Time of Action are to be executed
        in the order specified by the message index."
15     : : = {tmxMediaControl 12}

        tmxAsyncDataEntry OBJECT-TYPE
        SYNTAX TmxAsyncDataEntry
        MAX-ACCESS not-accessible
20     STATUS current
        DESCRIPTION
        "Entry"
        INDEX {ifIndex, tmxStatGroupOpIndex, tmxProgOpIndex,
        tmxAsyncDataOpIndex}
25     : : = {tmxAsyncDataTable 1}

        TmxAsyncDataEntry : : = SEQUENCE {
        tmxAsyncDataOpIndex Integer32,
        tmxAsyncDataMsg BITS,
30     tmxAsyncDataSrcId Integer32,
        tmxAsyncDataPID Integer32,
        tmxAsyncDataBaudRate Integer32,
        tmxAsyncDataParity Integer32,
        tmxAsyncDataBits Integer32,
35     tmxAsyncDataStopBits Integer32,
        tmxAsyncDataFlowControl BITS,
        tmxAsyncDataMuxMode BITS,
        tmxAsyncDataOutMode BITS,
        tmxAsyncDataAvgBitRate Integer32,
40     tmxAsyncDataAvgTimePeriod Integer32,
        tmxAsyncDataMinBitRate Integer32,
        tmxAsyncDataMaxBitRate Integer32,
        tmxAsyncDataActionTrigger ActionTriggerType,
        tmxAsyncDataActionTime DateAndTime,
45     tmxAsyncDataStatus RowStatus
        }

*****
        tmxAsyncDataOpIndex OBJECT-TYPE
        SYNTAX Integer32
50     MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
        "This is used for unique identification and sequencing of all
        operational configuration messages. The value is the value of
55     tmxOpNum when entry was created and is set by the manager."
        : : = {tmxAsyncDataEntry 1}

        tmxAsyncDataMsg OBJECT-TYPE
        SYNTAX BITS {create (0), delete (1)}
60     MAX-ACCESS read-create
        STATUS current
        DESCRIPTION

```

```

    "Each table row constitutes a command message. The message
    type indicates whether the Command type is to either create
    or delete the item described in this row. Any create command
    is further restricted by the Action Trigger Type. Depending on
5      the Action Trigger Type, the command will either be immediately
    executed, executed at a time specified by Time of Action, or
    executed when a SPliceImmediateFlag is raised."
    : : = {tmxAsyncDataEntry 2}

10      tmxAsyncDataSrcId OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
15      "Physical input identifier (ifIndex).".
        : : = {tmxAsyncDataEntry 3}

        tmxAsyncDataPID OBJECT-TYPE
        SYNTAX Integer32
20      MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
        "MPEG PID number in output multiplex for this data stream."
        : : = {tmxAsyncDataEntry 4}

25      tmxAsyncDataBaudRate OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
30      DESCRIPTION
        "RS-232 baud rate setting, if any."
        : : = {tmxAsyncDataEntry 5}

        tmxAsyncDataParity OBJECT-TYPE
35      SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
        "RS-232 parity, if any."
40      : : = {tmxAsyncDataEntry 6}

        tmxAsyncDataBits OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS read-create
45      STATUS current
        DESCRIPTION
        "RS-232 number of data bits, if any."
        : : = {tmxAsyncDataEntry 7}

50      tmxAsyncDataStopBits OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
55      "RS-232, number of stop bits, if any."
        : : = {tmxAsyncDataEntry 8}

        tmxAsyncDataFlowControl OBJECT-TYPE
        SYNTAX BITS {xonxoff (0), hw (1), none (2)}
60      MAX-ACCESS read-create
        STATUS current
        DESCRIPTION

```

```

"RS-232 flow control, if any."
: : = {tmxAsyncDataEntry 9}

tmxAsyncDataMuxMode OBJECT-TYPE
5   SYNTAX BITS {guaranteed (0), opportunistic (1)}
   MAX-ACCESS read-create
   STATUS current
   DESCRIPTION
10  "Data multiplexing mode. Either guaranteed, in which case either
   sufficiently much multiplexing bandwidth is allocated or the
   video quality will be sacrificed in a statistical multiplex;
   or opportunistic in which case data multiplexing is dependent
   on spare bandwidth after desired video quality is achieved"
   : : = {tmxAsyncDataEntry 10}

15  tmxAsyncDataOutMode OBJECT-TYPE
   SYNTAX BITS (atsc (0), dvb {1})
   MAX-ACCESS read-create
   STATUS current
20  DESCRIPTION
   "Data encapsulation on output."
   : : = (tmxAsyncDataEntry 11)

tmxAsyncDataAvgBitRate OBJECT-TYPE
25  SYNTAX Integer32
   MAX-ACCESS read-create
   STATUS current
   DESCRIPTION
30  "Average data bit rate over tmxAsyncDataAvgTimePeriod, if
   any."
   : : = {tmxAsyncDataEntry 12}

tmxAsyncDataAvgTimePeriod OBJECT-TYPE
35  SYNTAX Integer32
   MAX-ACCESS read-create
   STATUS current
   DESCRIPTION
40  "Averaging time period in Seconds, if any."
   : : = {tmxAsyncDataEntry 13}

tmxAsyncDataMinBitRate OBJECT-TYPE
   SYNTAX Integer32
   MAX-ACCESS read-create
   STATUS current
45  DESCRIPTION
   "Minimum guaranteed or desired bit rate over
   tmxAsyncDataAvgTimePeriod, if any."
   : : = {tmxAsyncDataEntry 14}

50  tmxAsyncDataMaxBitRate OBJECT-TYPE
   SYNTAX Integer32
   MAX-ACCESS read-create
   STATUS current
   DESCRIPTION
55  "Maximum guaranteed. or desired bit rate over
   tmxAsyncDataAvgTimePeriod, if any."
   : : = (tmxAsyncDataEntry 15)

tmxAsyncDataActionTrigger OBJECT-TYPE
60  SYNTAX ActionTriggerType
   MAX-ACCESS read-create
   STATUS current

```

```

        DESCRIPTION
        "Trigger type that will initiate this operation."
        : : = {tmxAsyncDataEntry 16}

5      tmxAsyncDataActionTime OBJECT-TYPE
        SYNTAX DateAndTime
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
10     "If trigger type is a time of action this object contains
        the time."
        : : = {tmxAsyncDataEntry 17}

        tmxAsyncDataStatus OBJECT-TYPE
15     SYNTAX RowStatus
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
        "Status of the row."
20     : : = {tmxAsyncDataEntry 18}

*****
        tmxCarouselTable OBJECT-TYPE
        SYNTAX SEQUENCE OF TmxCarouselEntry
25     MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
        "The Carousel Control table. Each row in this
30     table represents a create/delete Op command. All
        Ops with the same Time of Action are to be executed
        in the order specified by the message index."
        ::= (tmxMediaControl 13)

        tmxCarouselEntry OBJECT-TYPE
35     SYNTAX TmxCarouselEntry
        MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
        "Entry"
40     INDEX {ifIndex, tmxStatGroupOpIndex, tmxProgOpIndex,
        tmxCarouselOpIndex}
        : : = {tmxCarouselTable 1}

        TmxCarouselEntry : : = SEQUENCE {
45         tmxCarouselOpIndex Integer32,
        tmxCarouselMsg BITS,
        tmxCarouselPID Integer32,
        tmxCarouselOutStandard BITS,
        tmxCarouselRepetitionRate Integer32,
50         tmxCarouselLoadData OCTET STRING,
        tmxCarouselActionTrigger ActionTriggerType,
        tmxCarouselActionTime DateAndTime,
        tmxCarouselOnDemand BITS,
        tmxCarouselStatus RowStatus
55     }

*****
        tmxCarouselOpIndex OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS not-accessible
60     STATUS current
        DESCRIPTION
        "This is used for unique identification and sequencing of all

```

```

operational configuration messages. The value is the value of
tmxOpNum when entry was created and is set by the manager."
: : = {tmxCarouselEntry 1}

5      tmxCarouselMsg OBJECT-TYPE
        SYNTAX BITS (create (0), delete (1))
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
10     "Each table row constitutes a command message. The message
        type indicates whether the Command type is to either create
        or delete the item described in this row. Any create command
        is further restricted by the Action Trigger Type. Depending on
        the Action Trigger Type, the command will either be immediately
15     executed, executed at a time specified by Time of Action, or
        executed when a SPliceImmediateFlag is raised."
        ::= {tmxCarouselEntry 2}

        tmxCarouselPID OBJECT-TYPE
20     SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
        "MPEG PID number in output multiplex."
25     : : = {tmxCarouselEntry 3}

        tmxCarouselOutStandard OBJECT-TYPE
        SYNTAX BITS {atsc (0), dvb (1)}
        MAX-ACCESS read-create
30     STATUS current
        DESCRIPTION
        "Data encapsulation on output, if any."
        : : = {tmxCarouselEntry 4}

35     tmxCarouselRepetitionRate OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
40     "Repetition rate of the data in milliseconds."
        : : = {tmxCarouselEntry 5}

        tmxCarouselLoadData OBJECT-TYPE
        SYNTAX OCTET STRING
45     MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
        "File name on server or if LoadHost=NULL, this
        is the data itself that is to be put onto the
50     Carousel."
        ::= {tmxCarouselEntry 6}

        tmxCarouselActionTrigger OBJECT-TYPE
        SYNTAX ActionTriggerType
55     MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
        "Trigger type that will initiate this operation."
        : : = {tmxCarouselEntry 7}

60     tmxCarouselActionTime OBJECT-TYPE
        SYNTAX DateAndTime

```

```

MAX-ACCESS read-create
STATUS current
DESCRIPTION
5  "If trigger type is a time of action this object contains
   the time."
   ::= (tmxCarouselEntry 8)

tmxCarouselOnDemand OBJECT-TYPE
10  SYNTAX BITS (disabled (0), enabled (1))
   MAX-ACCESS read-create
   STATUS current
   DESCRIPTION "Switch for on-demand table generation"
   ::= {tmxCarouselEntry 9}

15  tmxCarouselStatus OBJECT-TYPE
   SYNTAX RowStatus
   MAX-ACCESS read-create
   STATUS current
   DESCRIPTION "Status of the row."
20  ::= {tmxCarouselEntry 10}

*****
tmxFifoTable OBJECT-TYPE
25  SYNTAX SEQUENCE OF TmxFifoEntry
   MAX-ACCESS not-accessible
   STATUS current
   DESCRIPTION
30  "The FIFO Control table. Each row in this
   table represents a create/delete Op command. All
   Ops with the same Time of Action are to be executed
   in the order specified by the message index."
   ::= (tmxMediaControl 14)

tmxFifoEntry OBJECT-TYPE
35  SYNTAX TmxFifoEntry
   MAX-ACCESS not-accessible
   STATUS current
   DESCRIPTION
40  "Entry"
   INDEX index, tmxStatGroupOpIndex, tmxProgOpIndex,
   tmxFifoOpIndex}
   ::= {tmxFifoTable 1}

TmxFifoEntry.. SEQUENCE {
45  tmxFifoOpIndex Integer32,
   tmxFifoMsg BITS,
   tmxFifoPID Integer32,
   tmxFifoOutStandard BITS,
   tmxFifoAvgBitRate Integer32,
50  tmxFifoAvgTimePeriod Integer32,
   tmxFifoMinBitRate Integer32,
   tmxFifoMaxBitRate Integer32,
   tmxFifoLoadTime Integer32,
55  tmxFifoLoadType BITS,
   tmxFifoLoadHost IpAddress,
   tmxFifoFileName DisplayString,
   tmxFifoLoadDataType BITS,
   tmxFifoActionTrigger ActionTriggerType,
   tmxFifoActionTime DateAndTime,
60  tmxFifoOnDemand BITS,
   tmxFifoStatus RowStatus
}
```

```

*****
tmxFifoOpIndex OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS not-accessible
5    STATUS current
    DESCRIPTION
        "This is used for unique identification and sequencing of all
        operational configuration messages. The value is the value of
        tmxOpNum when entry was created and is set by the manager."
10    ::= {tmxFifoEntry 1}

tmxFifoMsg OBJECT-TYPE
    SYNTAX BITS (create (0), delete (1))
    MAX-ACCESS read-create
15    STATUS current
    DESCRIPTION
        "Each table row constitutes a command message. The message
        type indicates whether the Command type is to either create
        or delete the item described in this row. Any create command
        is further restricted by the Action Trigger Type. Depending on
20    the Action Trigger Type, the command will either be immediately
        executed, executed at a time specified by Time of Action, or
        executed when a SPliceImmediateFlag is raised."
        ::= {tmxFifoEntry 2}
25

tmxFifoPID OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS read-create
    STATUS current
30    DESCRIPTION
        "MPEG PID number in output multiplex."
        ::= {tmxFifoEntry 3}

tmxFifoOutStandard OBJECT-TYPE
35    SYNTAX BITS {atsc (0), dvb (1)}
    MAX-ACCESS read-create
    STATUS current
    DESCRIPTION
        "Data encapsulation on output, if any."
40    ::= {tmxFifoEntry 4}

tmxFifoAvgBitRate OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS read-create
45    STATUS current
    DESCRIPTION
        "Average data bit rate over tmxFifoAvgTimePeriod, if any."
        ::= {tmxFifoEntry 5}

50    tmxFifoAvgTimePeriod OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
55        "Averaging time period in Seconds, if any."
        ::= {tmxFifoEntry 6}

tmxFifoMinBitRate OBJECT-TYPE
60    SYNTAX Integer32
    MAX-ACCESS read-create
    STATUS current
    DESCRIPTION

```

```

    "Minimum guaranteed or desired bit rate over
tmxFifoAvgTimePeriod, if any."
    ::= {tmxFifoEntry 7}

5      tmxFifoMaxBitRate OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
10      "Maximum guaranteed or desired bit rate over
tmxFifoAvgTimePeriod, if any."
        ::= {tmxFifoEntry 8}

tmxFifoLoadTime OBJECT-TYPE
15      SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
20      "Time at which to load the file from server, if any."
        ::= {tmxFifoEntry 9}

tmxFifoLoadType OBJECT-TYPE
        SYNTAX BITS {top (0), ftp (1), nfs (2), snmpdata (
25      3)}
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
        "FIFO load protocol."
30      ::= {tmxFifoEntry 10}

tmxFifoLoadHost OBJECT-TYPE
        SYNTAX IpAddress
        MAX-ACCESS read-create
        STATUS current
35      DESCRIPTION
        "IP address of server."
        ::= {tmxFifoEntry 11}

tmxFifoFileName OBJECT-TYPE
40      SYNTAX DisplayString
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
        "File name on server."
45      ::= {tmxFifoEntry 12}

tmxFifoLoadDataType OBJECT-TYPE
        SYNTAX BITS (data (0), dsmcc (1), prepacketized (2
50      ))
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
        "Format of data to be put onto the FIFO."
55      ::= {tmxFifoEntry 13}

tmxFifoActionTrigger OBJECT-TYPE
        SYNTAX ActionTriggerType
        MAX-ACCESS read-create
        STATUS current
60      DESCRIPTION
        "Trigger type that will initiate this operation."
        ::= {tmxFifoEntry 14}

```

```

tmxFifoActionTime OBJECT-TYPE
    SYNTAX DateAndTime
    MAX-ACCESS read-create
5    STATUS current
    DESCRIPTION
        "If trigger type is a time of action this object contains the
        time."
        : : = {tmxFifoEntry 15}
10
tmxFifoOnDemand OBJECT-TYPE
    SYNTAX BITS {disabled (0), enabled (1)}
    MAX-ACCESS read-create
    STATUS current
15    DESCRIPTION "Switch for on-demand table generation"
        : : = (tmxFifoEntry 16)

tmxFifoStatus OBJECT-TYPE
    SYNTAX RowStatus
20    MAX-ACCESS read-create
    STATUS current
    DESCRIPTION
        "Status of the row."
        : : = {tmxFifoEntry 17}
25 *****
tmxRrogIrisertTable OBJECT=TYPE
    SYNTAX SEQUENCE OF TmxProgInsertEntry
    MAX-ACCESS not-accessible
    STATUS current
30    DESCRIPTION
        "Table to control TMX's digital program insertion (Ads, NVOD)
        and splicing. Each row in this
        table represents a create/delete Op command. All
        Ops with the same Time of Action are to be executed
35    in the order specified by the message index."
        : : = {tmxMediaControl 15}

tmxProgInsertEntry OBJECT-TYPE
    SYNTAX TmxProgInsertEntry
40    MAX-ACCESS not-accessible
    STATUS current
    DESCRIPTION
        "Entry"
        INDEX {ifIndex, tmxStatGroupOpIndex, tmxProgOpIndex,
45    tmxProgInsertOpIndex}
        : : = {tmxProgInsertTable 1}

TmxProgInsertEntry ::= SEQUENCE {
50    tmxProgInsertOpIndex Integer32,
    tmxProgInsertMsg BITS,
    tmxProgInsertProgNameDisplayString,
    tmxProgInsertSplicerNameDisplayString,
    tmxProgInsertSplicerType BITS,
55    tmxProgInsertServerAddr IpAddress,
    tmxProgInsertServerPort Integer32,
    tmxProgInsertAdSrcId Integer32,
    tmxProgInsertTmxAddr IpAddress,
    tmxProgInsertTmxPort Integer32,
60    tmxProgInsertActionTrigger ActionTriggerType,
    tmxProgInsertActionTime DateAndTime,
    tmxProgInsertStatus RowStatus
}

```

```

*****
tmxProgInsertOpIndex OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS not-accessible
5    STATUS current
    DESCRIPTION
        "This is used for unique identification and sequencing of all
        operational configuration messages. The value is the value of
        tmxOpNum when entry was created and is set by the manager."
10    ::= {tmxProgInsertEntry 1}

tmxProgInsertMsg OBJECT-TYPE
    SYNTAX BITS {create (0), delete (1)}
    MAX-ACCESS read-create
15    STATUS current
    DESCRIPTION
        "Each table row constitutes a command message. The message
        type indicates whether the Command type is to either create
        or delete the item described in this row. Any create command
        is further restricted by the Action Trigger Type. Depending on
20    the Action Trigger Type, the command will either be immediately
        executed, executed at a time specified by Time of Action, or
        executed when a SPliceImmediateFlag is raised."
        : : = {tmxProgInsertEntry 2}
25

tmxProgInsertProgName OBJECT-TYPE
    SYNTAX DisplayString
    MAX-ACCESS read-create
    STATUS current
30    DESCRIPTION
        "User assigned program name of the program being spliced
        into."
        ::= {tmxProgInsertEntry 3}

35    tmxProgInsertSplicerName OBJECT-TYPE
        SYNTAX DisplayString
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
40        "Name of the splicer if more than one in TMX (as in DVS
        standard)."
        ::= {tmxProgInsertEntry 4}

        tmxProgInsertSplicerType OBJECT-TYPE
45        SYNTAX BITS {adstreaminsert (0), nvodstreaminsert (1),
        actvstreaminsert(2)}
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
50        "Type of splicer."
        : : = {tmxProgInsertEntry 5}

        tmxProgInsertServerAddr OBJECT-TYPE
        SYNTAX IpAddress
55        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
        "IP address of server for this stream insertion for
        DVS 380 API."
60        : : = {tmxProgInsertEntry 6}

        tmxProgInsertServerPort OBJECT-TYPE

```

```

        SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
5         "Server TCP port number for control of this stream insertion
          for DVS 380 API."
          ::= {tmxProgInsertEntry 7}

tmxProgInsertAdSrcId OBJECT-TYPE
10      SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
15         "Physical input identifier (index) of the ad/NVOD stream."
          ::= {tmxProgInsertEntry 8}

tmxProgInsertTmxAddr OBJECT-TYPE
        SYNTAX IpAddress
        MAX-ACCESS read-create
20      STATUS current
        DESCRIPTION
        "IP address of TMX for this stream insertion for DVS 380
        API."
        ::= {tmxProgInsertEntry 9}
25

tmxProgInsertTmxPort OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS read-create
        STATUS current
30      DESCRIPTION
        "TMX TCP port number for control of this stream insertion
        for DVS 380 API."
        ::= {tmxProgInsertEntry 10}

35      tmxProgInsertActionTrigger OBJECT-TYPE
        SYNTAX ActionTriggerType
        MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
40         "Trigger type that will initiate this operation."
          ::= {tmxProgInsertEntry 11}

tmxProgInsertActionTime OBJECT-TYPE
        SYNTAX DateAndTime
45      MAX-ACCESS read-create
        STATUS current
        DESCRIPTION
        "If trigger type is a time of action this object contains the
        time."
50      ::= {tmxProgInsertEntry 12}

tmxProgInsertStatus OBJECT-TYPE
        SYNTAX RowStatus
        MAX-ACCESS read-create
55      STATUS current
        DESCRIPTION
        "Status of the row."
        ::= {tmxProgInsertEntry 13}

60      -- tmxMediaMonitor tables enable monitoring of MPEG input/output
        -- information.
        -- They consist of:

```

```

--      tmxInputPIDTable for input rate monitoring per PID
--      tmxOutputPIDTable for output rate monitoring per PID
--      tmxPsiSiTable for MPEG Table monitoring per PID
*****
5      tmxInputPIDyTabl a OB-J-ECT-TYPE
      SYNTAX SEQUENCE OF TmxInputPIDEntry
      MAX-ACCESS not-accessible
      STATUS current
      DESCRIPTION
10     "Table with input PID statistics."
      : : = {tmxMediaMonitor 1}

      tmxInputPIDEntryOBJECT-TYPE
      SYNTAX TmxInputPIDEntry
15     MAX-ACCESS not-accessible
      STATUS current
      DESCRIPTION
      "Entry"
      INDEX {ifIndex, tmxInputPIDId}
20     : : = {tmxInputPIDTable 1}

      TmxInputPIDEntry : : = SEQUENCE {
      tmxInputPIDId Integer32,
      tmxInputPIDBitRate Integer32,
25     tmxInputPIDEntryStatus RowStatus
      }
*****
      tmxInputPIDId OBJECT-TYPE
      SYNTAX Integer32
30     MAX-ACCESS read-only
      STATUS current
      DESCRIPTION
      "Component PID with stream type."
      ::= {tmxInputPIDEntry 1}
35

      tmxInputPIDBitRate OBJECT-TYPE
      SYNTAX Integer32
      MAX-ACCESS read-only
      STATUS current
40     DESCRIPTION
      "Bit rate in bps."
      : : = {tmxInputPIDEntry 2}

      tmxInputPIDEntryStatus OBJECT-TYPE
45     SYNTAX RowStatus
      MAX-ACCESS read-create
      STATUS current
      DESCRIPTION
      "Status of this entry."
50     : : = {tmxInputPIDEntry 3}
*****
      tmxOutputPIDTable ~ OH. JECT-TY-L ? E
      SYNTAX SEQUENCE OF TmxOutputPIDEntry
      MAX-ACCESS not-accessible
55     STATUS current
      DESCRIPTION
      "Table with output PID statistics."
      ::= {tmxMediaMonitor 2}

      tmxOutputPIDEntry OBJECT-TYPE
60     SYNTAX TmxOutputPIDEntry
      MAX-ACCESS not-accessible

```

```

        STATUS current
        DESCRIPTION
        "Entry"
        INDEX index, tmxOutputPIDId}
5      : : = {tmxOutputPIDTable 1}

TmxOutputPIDEntry : : = SEQUENCE {
        tmxOutputPIDId Integer32,
        tmxOutputPIDBitRate Integer32,
10      tmxOutputPIDMinBitRate Integer32,
        tmxOutputPIDMaxBitRate Integer32,
        tmxOutputPIDFrameRate Integer32,
        tmxOutputPIDResolution Integer32,
        tmxOutputPIDBFrames Integer32,
15      tmxOutputPIDFilmMode Integer32,
        tmxOutputPIDEntryStatus RowStatus
    }
*****
    tmxOutputPIDId OBJECT-TYPE
20      SYNTAX Integer32
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION
        "Component PID with stream type."
25      : : = {tmxOutputPIDEntry 1}

    tmxOutputPIDBitRate OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS read-only
30      STATUS current
        DESCRIPTION
        "Bit rate in bps."
        : : = {tmxOutputPIDEntry 2}

    tmxOutputPIDMinBitRate OBJECT-TYPE
35      SYNTAX Integer32
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION
40      "Min Bit rate in bps."
        : : = {tmxOutputPIDEntry 3}

    tmxOutputPIDMaxBitRate OBJECT-TYPE
        SYNTAX Integer32
45      MAX-ACCESS read-only
        STATUS current
        DESCRIPTION
        "Max Bit rate in bps."
        : : = {tmxOutputPIDEntry 4}
50

    tmxOutputPIDFrameRate OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS read-only
        STATUS current
55      DESCRIPTION
        "Frame rate if the PID is a transcoded video PID."
        : : = {tmxOutputPIDEntry 5}

    tmxOutputPIDResolution OBJECT-TYPE
60      SYNTAX Integer32
        MAX-ACCESS read-only
        STATUS current

```

```

DESCRIPTION
"Number of Macroblocks per Second if PID is a transcoded
video PID."
: : = {tmxOutputPIDEntry 6}

5
tmxOutputPIDBframes OBJECT-TYPE
SYNTAX Integer32
MAX-ACCESS read-only
STATUS current
10 DESCRIPTION
"Average ratio of B frames to I and P frames if PID is a
transcoded video PID."
: : = (tmxOutputPIDEntry 7)

15
tmxOutputPIDFilmMode OBJECT-TYPE
SYNTAX Integer32
MAX-ACCESS read-only
STATUS current
20 DESCRIPTION
"Film mode status if PID is a transcoded video PID."
: : = (tmxOutputPIDEntry 8)

tmxOutputPIDEntryStatus OBJECT-TYPE
25 SYNTAX RowStatus
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"Status of this entry."
30 : : = {tmxOutputPIDEntry 9}

*****
tmxPsiSiTable OBJECT-TYPE
SYNTAX SEQUENCE OF TmxPsiSiEntry
35 MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"Table of all input transport streams."
::= {tmxMediaMonitor 3}

40
tmxPsiSiEntry OBJECT-TYPE
SYNTAX TmxPsiSiEntry
MAX-ACCESS not-accessible
STATUS current
45 DESCRIPTION
"Entry"
INDEX {ifIndex, tmxPsiSiPidNum, tmxPsiSiTableId,
tmxPsiSiCurrNext}
: : = {tmxPsiSiTable 1}

50
TmxPsiSiEntry : : = SEQUENCE {
tmxPsiSiPidNum Integer32,
tmxPsiSiTableId Integer32,
tmxPsiSiCurrNext Integer32,
55 tmxPsiSiTableInfo OCTET STRING,
tmxPsiSiVersion Integer32,
tmxPsiSiEntryStatus RowStatus
}
*****
60
tmxPsiSiPidNum OBJECT-TYPE
SYNTAX Integer32
MAX-ACCESS not-accessible
STATUS current

```

```

        DESCRIPTION
        "Transport Stream PID number"
        : : = {tmxPsiSiEntry 1}

5      tmxPsiSiTableId OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
10     "Transport Stream Table Id"
        : : = {tmxPsiSiEntry 2}

        tmxPsiSiCurrNext OBJECT-TYPE
        SYNTAX Integer32
15     MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
        "MPEG Table currnet next indicator."
        : : = {tmxPsiSiEntry 3}

20     tmxPsiSiTableInfo OBJECT-TYPE
        SYNTAX OCTET STRING
        MAX-ACCESS read-only
        STATUS current
25     DESCRIPTION
        "Transport Stream Table."
        : : = {tmxPsiSiEntry 4}

        tmxPsiSiVersion OBJECT-TYPE
        SYNTAX Integer32
30     MAX-ACCESS read-only
        STATUS current
        DESCRIPTION
        "Table version for this transport stream"
35     : : = {tmxPsiSiEntry 5}

        tmxPsiSiEntryStatus OBJECT-TYPE
        SYNTAX RowStatus
        MAX-ACCESS read-create
40     STATUS current
        DESCRIPTION
        "Status of this entry."
        : : = {tmxPsiSiEntry 6}

45     tmxMuxMemoryTable OBJECT-TYPE
        SYNTAX SEQUENCE OF TmxMuxMemoryEntry
        MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
50     "Table indicating the maximum contiguous block of memory
available
        for carousels."
        : : = (tmxMediaMonitor 4)

55     tmxMuxMemoryEntry OBJECT-TYPE
        SYNTAX TmxMuxMemoryEntry
        MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
60     "Entry"
        INDEX(if Index)
        : : = {tmxMuxMemoryTable 1}

```

```

TmxMuxMemoryEntry ::= SEQUENCE {
    tmxMuxMemorySize Integer32,
    tmxMuxMemoryEntryStatus RowStatus
5      }

tmxMuxMemorySize OBJECT-TYPE
    SYNTAX Integer32
    MAX-ACCESS read-only
10    STATUS current
    DESCRIPTION
        "The maximum byte size currently available."
        ::= {tmxMuxMemoryEntry 1}

tmxMuxMemoryEntryStatus OBJECT-TYPE
    SYNTAX RowStatus
    MAX-ACCESS read-create
    STATUS current
15    DESCRIPTION
        "Status of this entry."
        ::= {tmxMuxMemoryEntry 2}
20
END

```

虽然本发明是结合在目前看来最实用和优选的实施例而描述的，
 25 但是应该可以理解，本发明并不限于所公开的这些实施例，而是希望
 包含那些包括在所附权利要求的精神和范围内的不同的修改和等价的
 结构。至于以上的描述，例如，应该认识到，对于本领域技术人员来
 说，可以很容易地得到最优的实现、功能以及操作、组合和使用的方式，
 并且所附权利要求希望包括那些在附图中说明和在说明书中描述的
 30 所有的等价关系。因此，可以认为上述对本发明原理的描述是说明
 性的，而并非穷举。

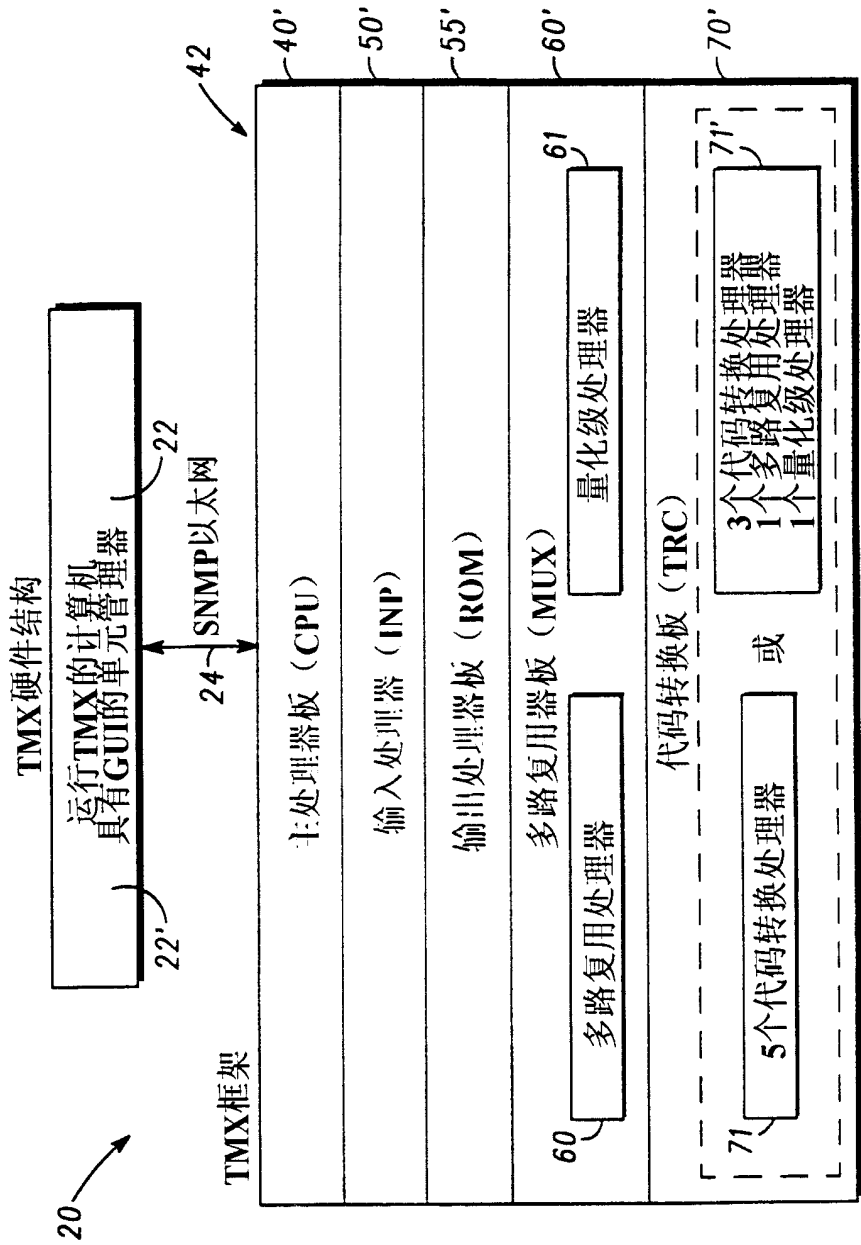


图1a

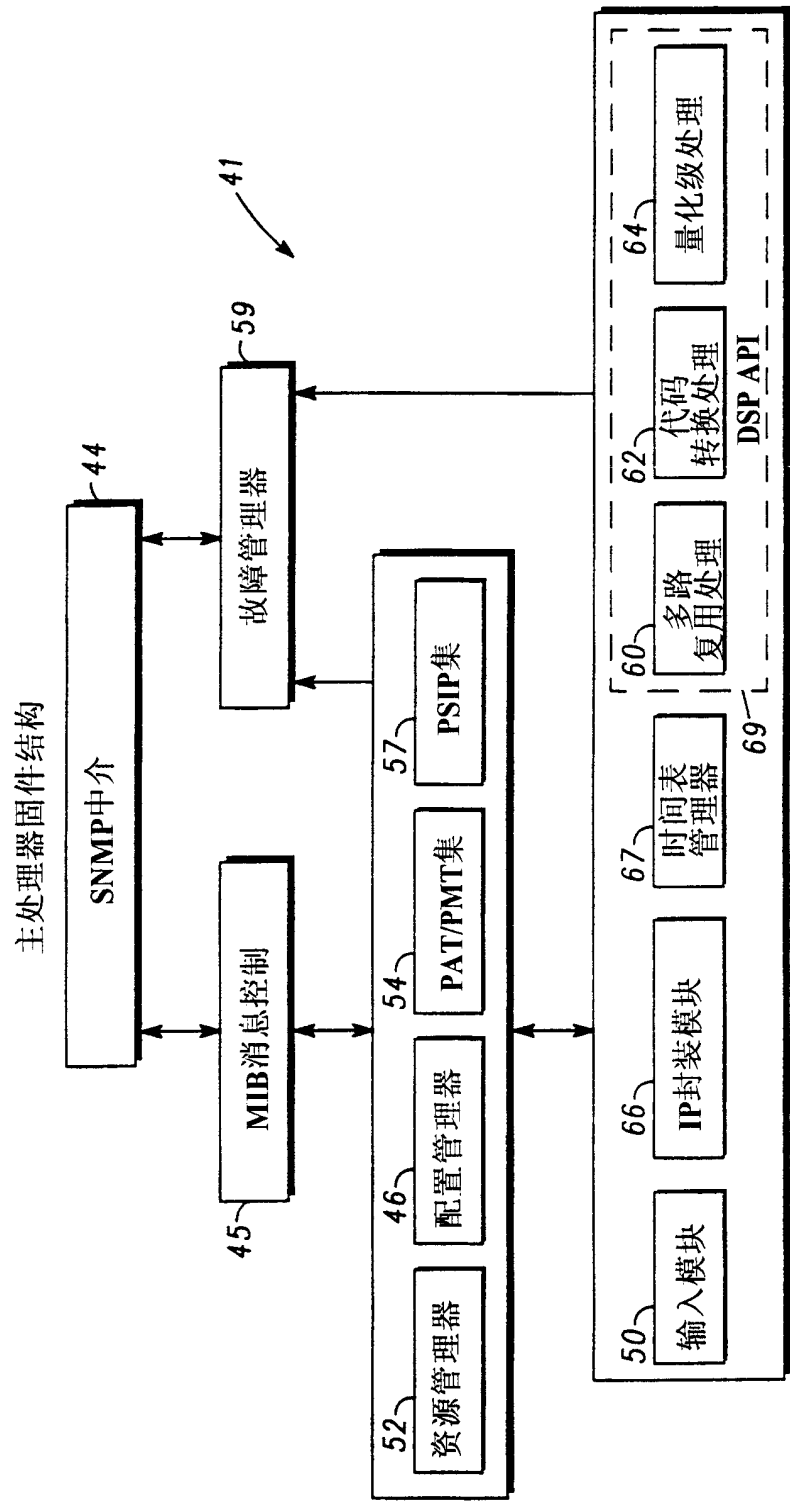
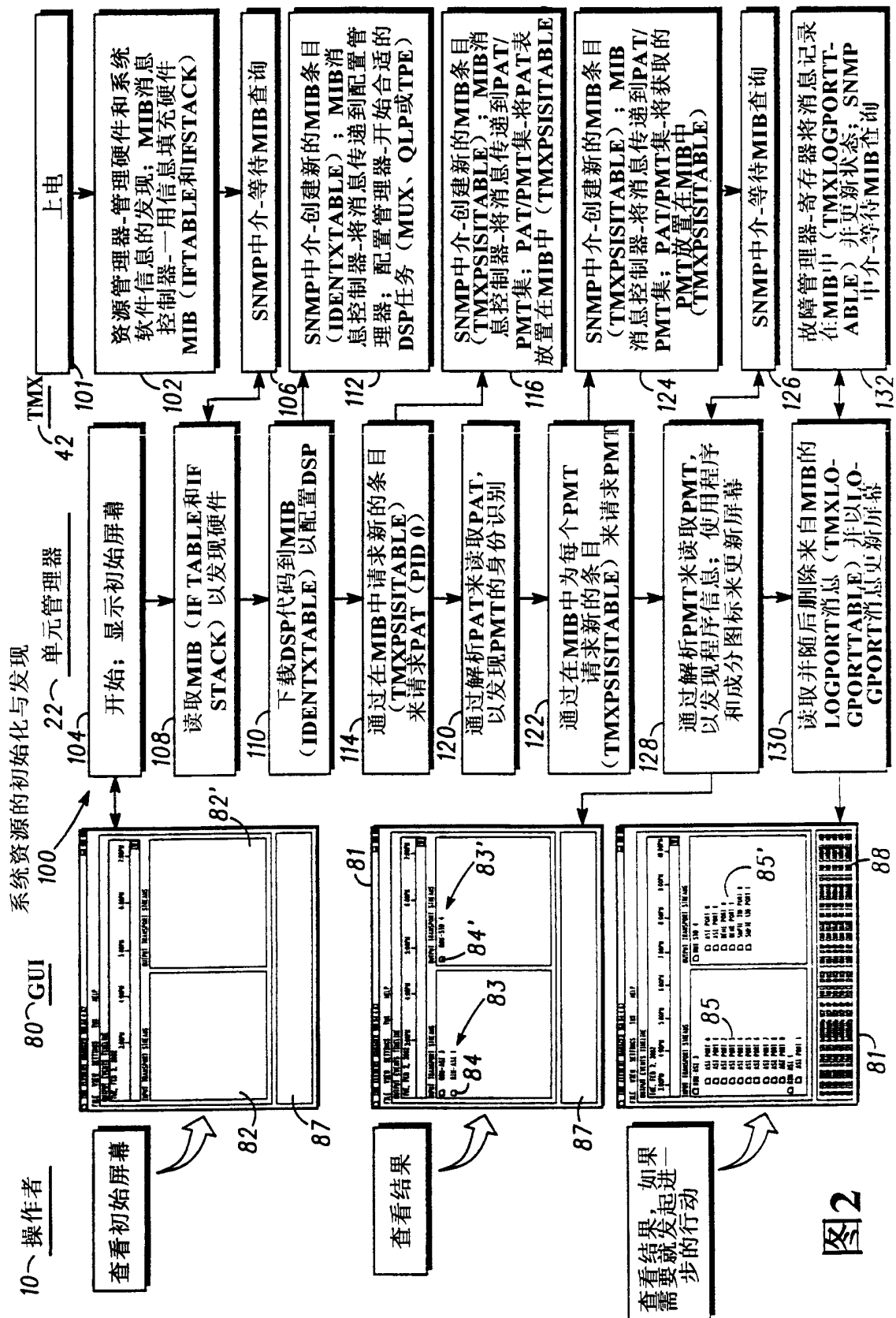
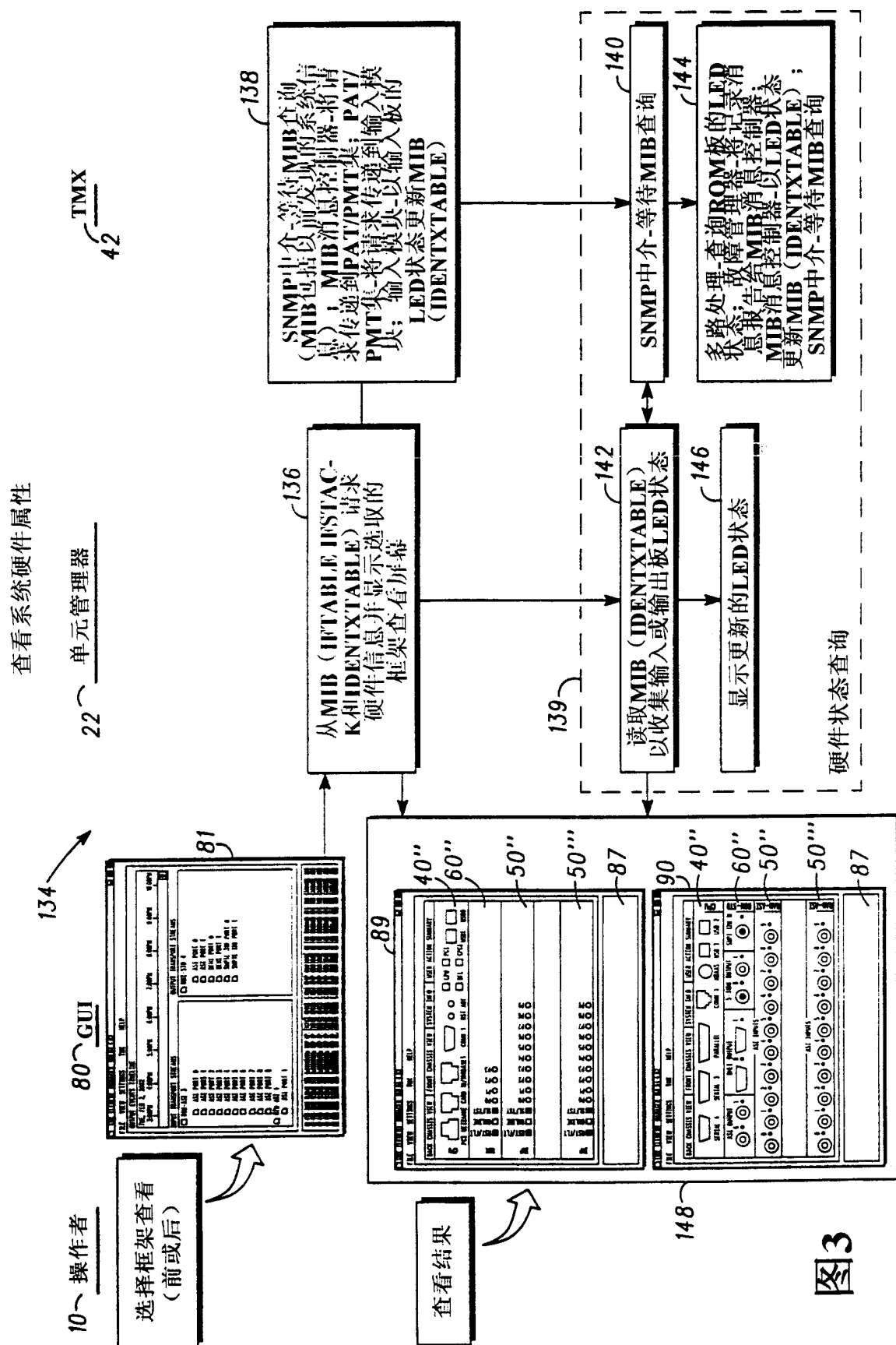
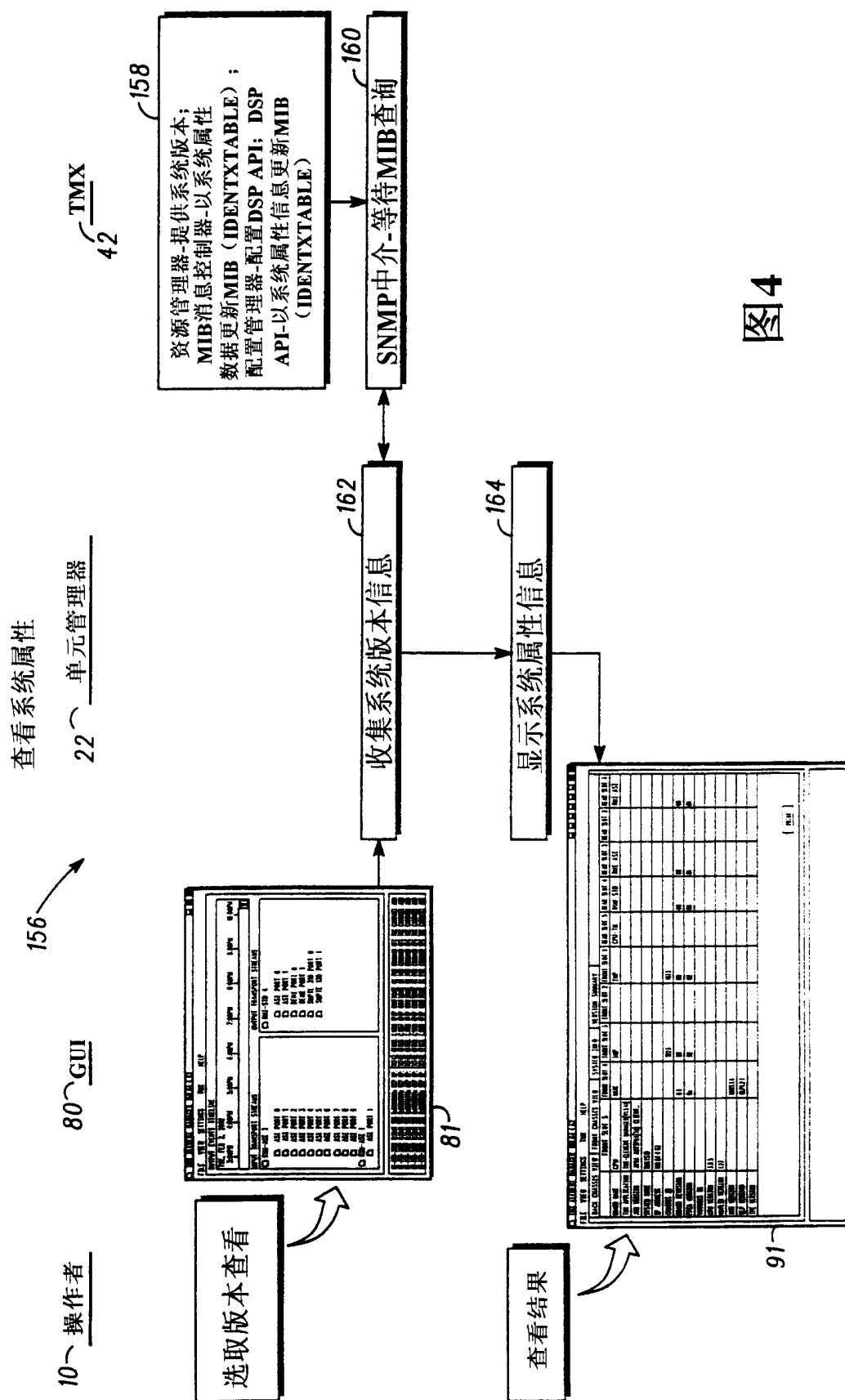
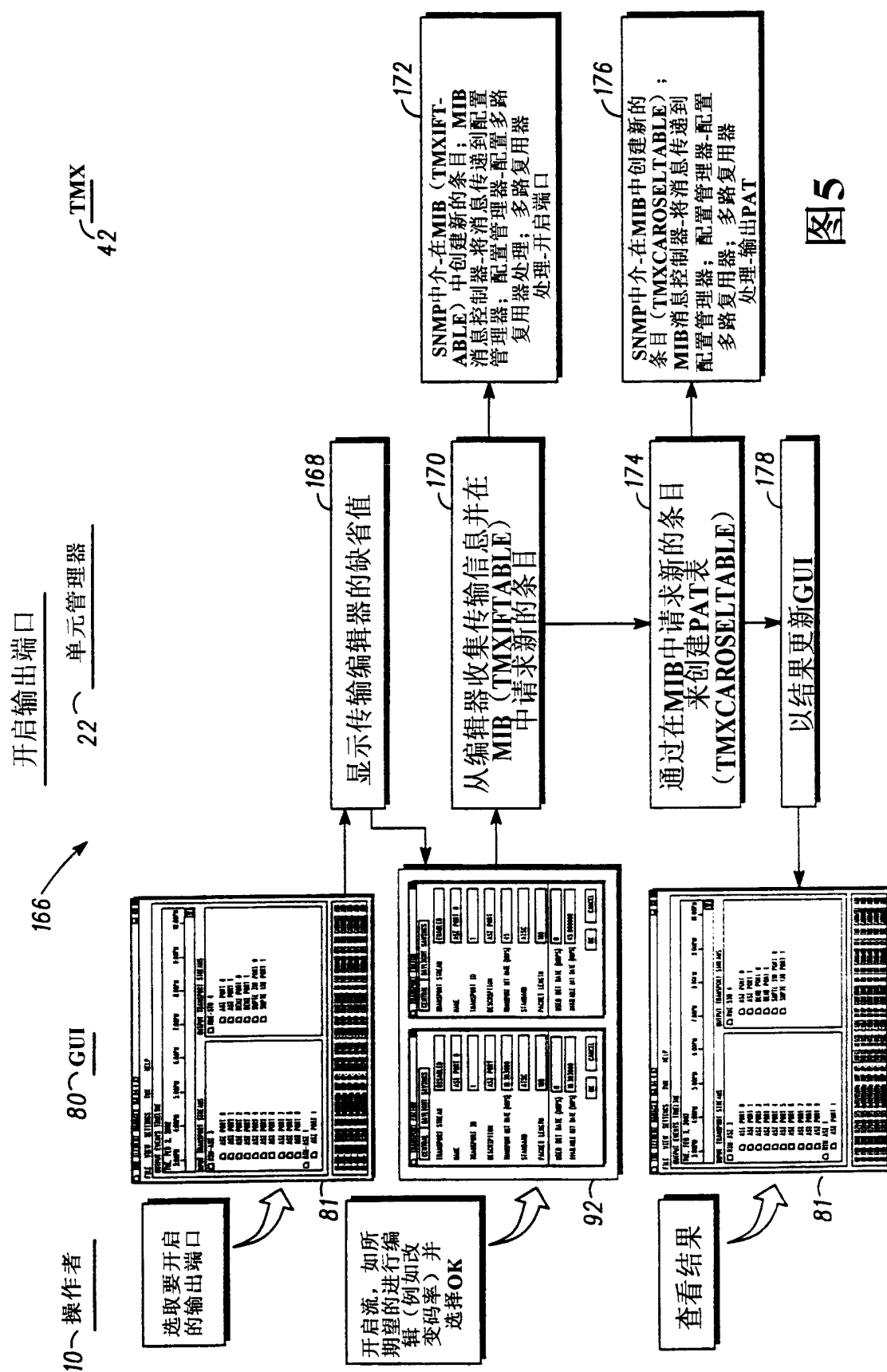


图1b









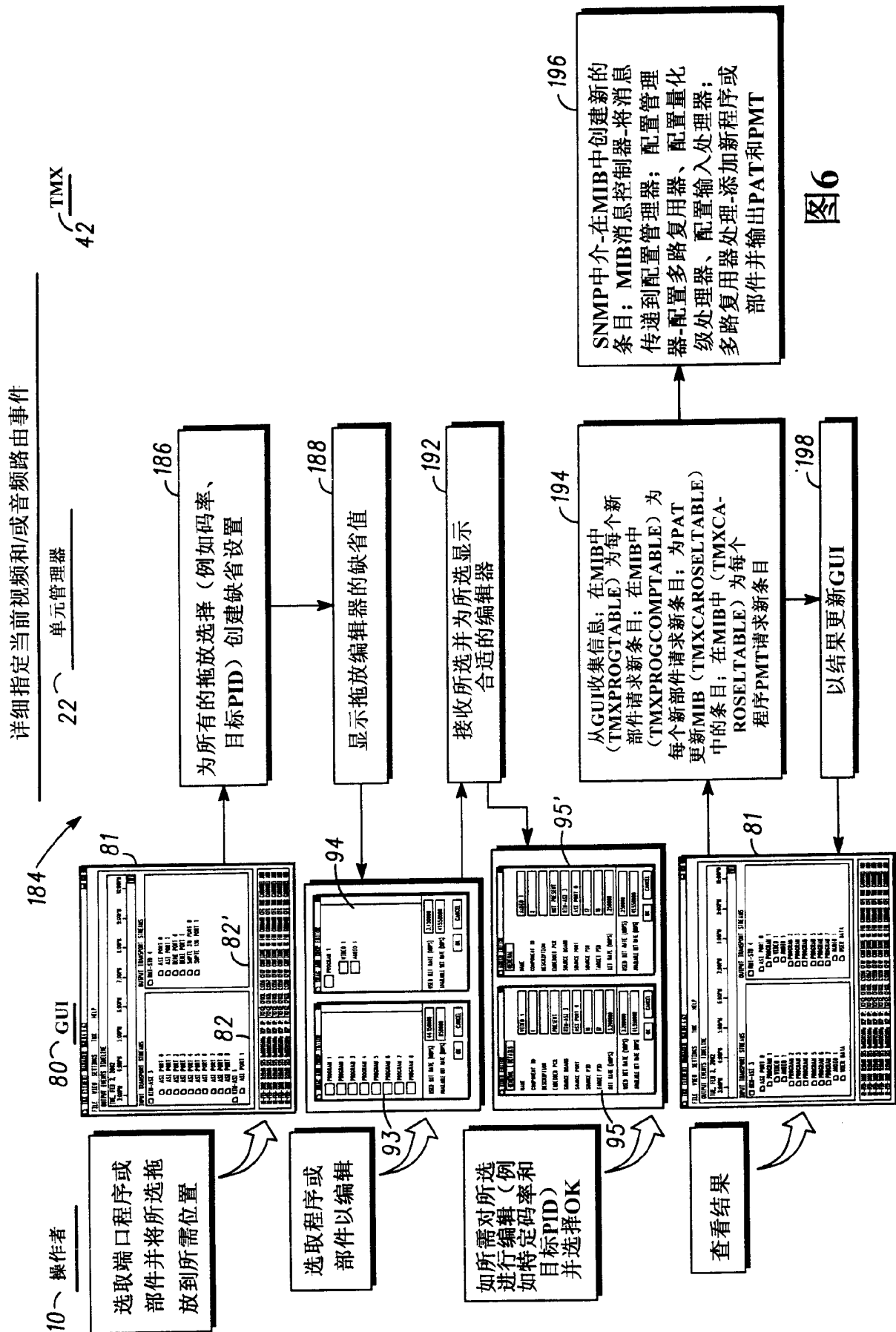
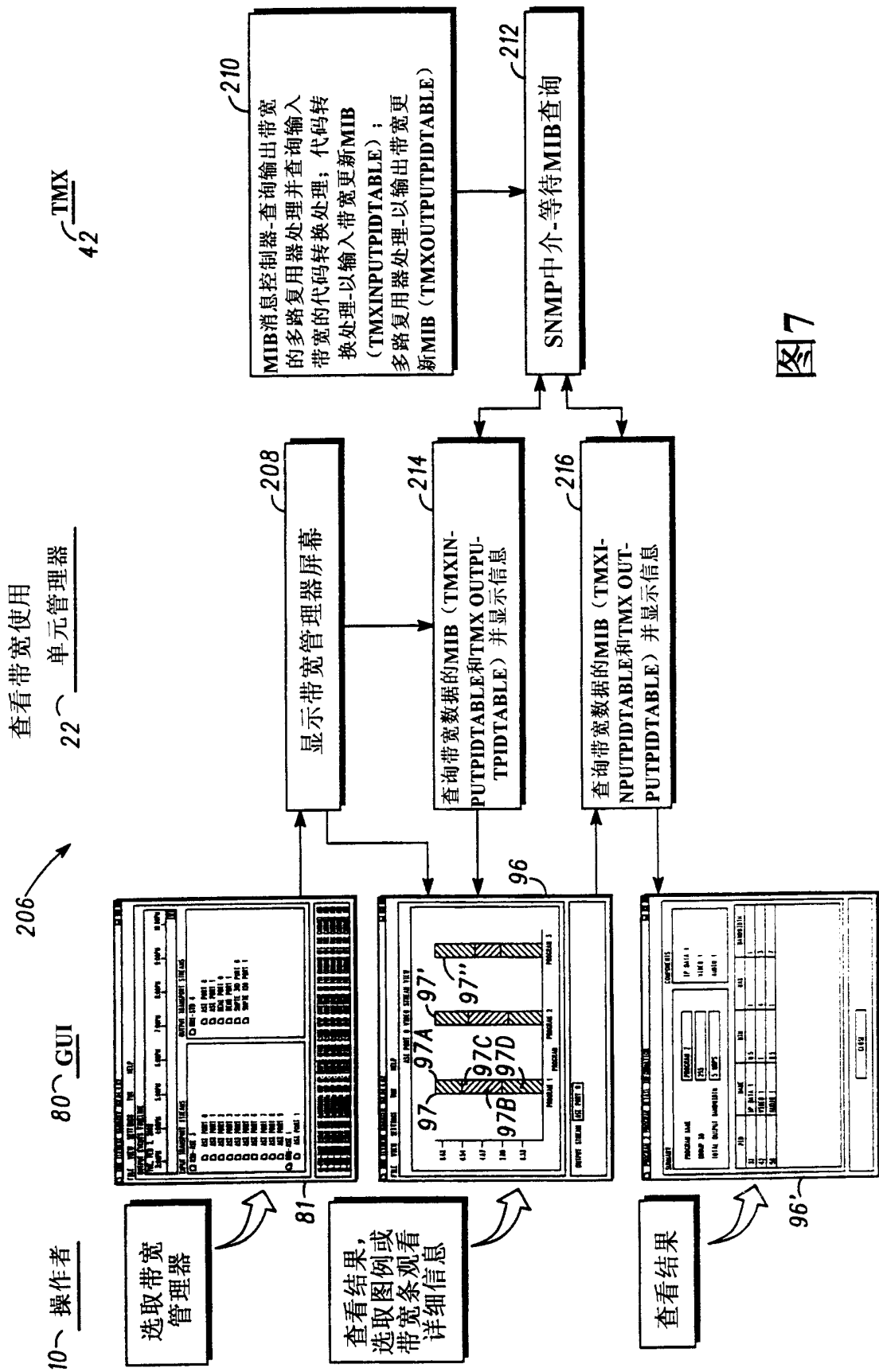


图6



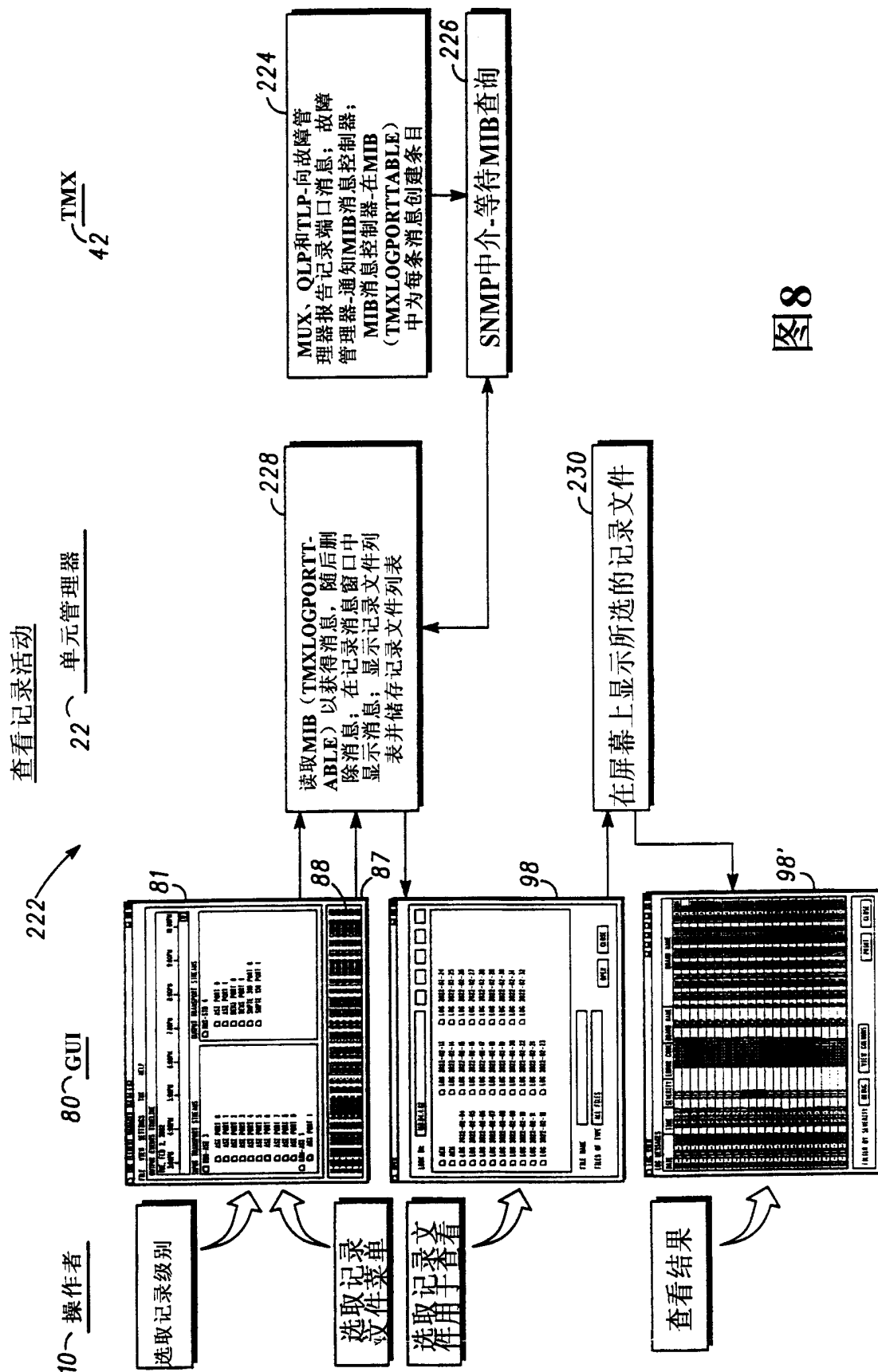


图8

