**WDM音频驱动程序简介**

内核流（KS）服务支持音频和其他类型的连续媒体的数据流的内核模式处理。从概念上讲，流在沿包含一些处理节点的数据路径流动时经历处理。一组相关节点被分组在一起以形成*KS过滤器*，该*过滤器*代表了流处理功能的独立块。通过将几个过滤器级联在一起以形成*过滤器图，*可以以模块化的方式构造更复杂的功能。

典型的音频适配器卡可能包含用于通过一组扬声器播放波形流，将来自麦克风的音频信号转换为波形流以及合成来自MIDI流的声音的音频设备。适配器驱动程序可以将每个音频设备包装在KS筛选器中，然后将其公开给操作系统。操作系统将过滤器连接到其他过滤器，以形成过滤器图，该过滤器图代表应用程序处理音频流。

KS滤波器通过其*引脚*连接在一起。音频过滤器上的引脚可以认为是音频插孔。当客户端需要将数据流路由到该过滤器中或从该过滤器路由出来时，客户端会实例化过滤器上的输入或输出引脚。在某些情况下，术语“ *引脚”*和“ *流”*可以互换使用。

上游滤波器的输出引脚连接到下游滤波器的输入引脚。来自输出引脚的数据流必须具有输入引脚可以接受的数据格式。通常需要数据缓冲来消除输出引脚产生数据而输入引脚消耗数据的速率的瞬时失配。

KS过滤器被实现为内核模式驱动程序对象，该对象封装了一些相关的流处理功能。该功能可以用软件或硬件来实现。在此模型中，音频适配器可以看作是硬件设备的集合，并且适配器驱动程序将这些设备中的每一个作为单独的过滤器公开给音频系统。

适配器驱动程序向音频系统公开*过滤器工厂*的集合。每个过滤器工厂都可以实例化特定类型的过滤器：

* 如果适配器包含一个或多个功能相似或相同的设备，则驱动程序会将这些设备的过滤器组合到同一过滤器工厂中。
* 如果适配器包含几种不同类型的设备，则这些设备将通过几种不同的过滤器工厂提供。

KS滤波器将一组*引脚工厂*公开给音频系统。每个引脚工厂都可以实例化特定类型的引脚。如果过滤器可以提供一个或多个功能相似或相同的引脚，则过滤器会将这些引脚归为同一引脚工厂。例如，执行音频混合的滤波器可能具有一个可以实例化单个输出引脚的引脚工厂和一个可以实例化多个输入引脚的第二引脚工厂。

KS服务基于Windows驱动程序模型构建。注意，术语“ *KS过滤器”*必须与术语“ *过滤器驱动程序”*区分开，后者是另一个WDM概念。筛选器驱动程序驻留在WDM驱动程序堆栈中，并且可以拦截和修改在堆栈中传播的I / O请求数据包（IRP）。上层和下层过滤器驱动程序分别位于功能驱动程序的上方和下方。在本节中，除非另有说明，否则术语*过滤器*是指KS过滤器而不是过滤器驱动器。有关筛选器驱动程序的更多信息，请参见[WDM驱动程序类型](https://docs.microsoft.com/windows-hardware/drivers/kernel/types-of-wdm-drivers)。

本节包含以下主题：

[WDM音频驱动程序的基本功能](https://docs.microsoft.com/en-us/windows-hardware/drivers/audio/basic-functions-of-a-wdm-audio-driver)

[供应商音频驱动程序选项](https://docs.microsoft.com/en-us/windows-hardware/drivers/audio/vendor-audio-driver-options)

[WDM音频术语](https://docs.microsoft.com/en-us/windows-hardware/drivers/audio/wdm-audio-terminology)

[样本音频驱动程序](https://docs.microsoft.com/en-us/windows-hardware/drivers/audio/sample-audio-drivers)

[KsStudio实用程序](https://docs.microsoft.com/en-us/windows-hardware/drivers/audio/ksstudio-utility)

有关WDM音频体系结构的新功能的更新和信息，请访问[音频技术](https://go.microsoft.com/fwlink/p/?linkid=8751)网站。

**WDM音频驱动程序的基本功能**

* 2017/04/20
* 2分钟阅读
  + [[https://github.com/DOMARS.png?size=32](https://github.com/MicrosoftDocs/windows-driver-docs/blob/staging/windows-driver-docs-pr/audio/basic-functions-of-a-wdm-audio-driver.md)](https://github.com/MicrosoftDocs/windows-driver-docs/blob/staging/windows-driver-docs-pr/audio/basic-functions-of-a-wdm-audio-driver.md" \o "1个贡献者)

Microsoft Windows驱动程序模型（WDM）音频驱动程序提供以下功能：

* 该驱动程序公开所有类型的输入和输出流，以及它可以支持的每种流类型的实例数。驱动程序以一组引脚工厂以及每个工厂可以实例化的引脚数量的形式提供此信息。例如，简单的音频设备可能会输入单个PCM音频流并输出单个PCM音频流。该设备的过滤器包含两个引脚工厂-一个用于输入流，一个用于输出流-每个引脚工厂仅支持一个引脚实例。如果适配器卡仅包含这些设备之一，则适配器驱动程序提供的筛选器工厂仅包含具有这些功能的筛选器的单个实例。
* 该驱动程序支持一个或多个属性集。例如，所有音频驱动程序应支持[KSPROPSETID\_Audio](https://docs.microsoft.com/windows-hardware/drivers/audio/kspropsetid-audio)，但是某些音频驱动程序也可能支持其他属性集。驱动程序的客户端使用属性请求来发现过滤器的功能并更改过滤器的可配置设置。
* 驱动程序可选地支持硬件时钟。此时钟应该是可读写的，以便流可以与相同或不同硬件上的其他流同步。有关更多信息，请参见[KSPROPSETID\_Clock](https://docs.microsoft.com/windows-hardware/drivers/stream/kspropsetid-clock)。
* 该驱动程序可选地支持其他媒体接口，例如[**KSINTERFACE\_STANDARD\_STREAMING**](https://docs.microsoft.com/windows-hardware/drivers/stream/ksinterface-standard-streaming)，**[KSINTERFACE\_MEDIA\_WAVE\_QUEUED](https://docs.microsoft.com/windows-hardware/drivers/stream/ksinterface-media-wave-queued)**或**[KSINTERFACE\_STANDARD\_LOOPED\_STREAMING](https://docs.microsoft.com/windows-hardware/drivers/stream/ksinterface-standard-looped-streaming)**。

**供应商音频驱动程序选项**

* 2017/04/20
* 2分钟阅读
  + [[https://github.com/DOMARS.png?size=32](https://github.com/MicrosoftDocs/windows-driver-docs/blob/staging/windows-driver-docs-pr/audio/vendor-audio-driver-options.md)](https://github.com/MicrosoftDocs/windows-driver-docs/blob/staging/windows-driver-docs-pr/audio/vendor-audio-driver-options.md" \o "1个贡献者)

为了利用对音频设备的内置系统支持，Microsoft建议供应商使用以下之一：

* ISA或PCI适配卡的端口类适配器驱动程序（请参阅[音频微型端口驱动程序](https://docs.microsoft.com/en-us/windows-hardware/drivers/audio/audio-miniport-drivers)）
* USB音频设备的USB音频类驱动程序（请参阅[USBAudio类系统驱动程序](https://docs.microsoft.com/en-us/windows-hardware/drivers/audio/kernel-mode-wdm-audio-components" \l "usbaudio_class_system_driver)）
* 针对IEEE 1394音频设备的自定义IEEE 1394设备驱动程序（请参阅[AVCAudio类系统驱动程序](https://docs.microsoft.com/en-us/windows-hardware/drivers/audio/kernel-mode-wdm-audio-components" \l "avcaudio_class_system_driver)）

但是，如果这些选项不够用，供应商可以实施以下其中一项：

* 专有的KS过滤器（请参阅[KS过滤器](https://docs.microsoft.com/windows-hardware/drivers/stream/ks-filters)）
* Microsoft不建议使用专有的KS筛选器，因为它们难以实现，并且对于大多数ISA，PCI和USB设备而言都是不必要的。
* 流类微型驱动程序（请参阅[流](https://docs.microsoft.com/windows-hardware/drivers/stream/streaming-minidrivers2)微型驱动[程序](https://docs.microsoft.com/windows-hardware/drivers/stream/streaming-minidrivers2)）
* Microsoft不建议使用专有的流类微型驱动程序，因为它很难实现，尽管它可能适合集成音频和视频的设备。

有关为音频设备提供驱动程序支持的可用选项的深入讨论，请参阅《[WDM音频驱动程序入门》](https://docs.microsoft.com/en-us/windows-hardware/drivers/audio/getting-started-with-wdm-audio-drivers)。

**WDM音频术语**

* 2017/04/20
* 7分钟阅读
  + [[https://github.com/DOMARS.png?size=32](https://github.com/MicrosoftDocs/windows-driver-docs/blob/staging/windows-driver-docs-pr/audio/wdm-audio-terminology.md)](https://github.com/MicrosoftDocs/windows-driver-docs/blob/staging/windows-driver-docs-pr/audio/wdm-audio-terminology.md" \o "1个贡献者)

本节描述了Microsoft Windows驱动程序模型（WDM）音频驱动程序体系结构和通用Windows分层驱动程序体系结构之间的术语差异。通用驱动程序体系结构以SCSI端口/微型端口驱动程序为例（请参阅[存储驱动程序体系结构](https://docs.microsoft.com/windows-hardware/drivers/storage/storage-driver-architecture)）。

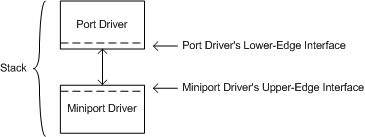
通用和WDM音频驱动程序体系结构定义的术语相似，但是它们确实有一些重要的区别，如下所述。

**微型端口驱动程序（通用）**

微型端口驱动程序（通用）是驻留在系统总线（例如，PCI或ISA）上的适配器的特定于硬件的驱动程序。该驱动程序具有单个入口点*[DriverEntry](https://docs.microsoft.com/windows-hardware/drivers/ddi/content/wdm/nc-wdm-driver_initialize)*，并向端口驱动程序注册功能表。此功能表充当微型端口驱动程序的上层接口。

微型端口驱动程序位于驱动程序堆栈中端口驱动程序的下方。也就是说，对微型端口驱动程序的所有调用都是从端口驱动程序发出的，所有从微型端口驱动程序发出的调用都是到端口驱动程序的下边缘接口的。

下图说明了在此上下文中使用的术语*stack*，*上边缘接口*和*下边缘接口*的含义。代表端口驱动程序的块堆叠在代表微型端口驱动程序的块的顶部。因此，微型端口驱动程序位于“堆栈”中端口驱动程序的下方。



端口和微型端口驱动程序通过彼此公开的软件接口进行通信。在上图中，这些接口与代表端口驱动器的模块的下边缘和代表微型端口驱动器的模块的上边缘相关联。此表示形式是术语“下边缘接口”和“上边缘接口”的来源。

**端口驱动程序（通用）**

端口驱动程序（通用）围绕微型端口驱动程序。

端口驱动程序：

* 实现WDM流过滤器。
* 提供与操作系统其余部分的通用接口。
* 处理来自系统的I / O请求，并将这些请求作为调用重新广播到微型端口驱动程序的功能表中。
* 为微型端口驱动程序提供支持功能的库（端口驱动程序的下边缘接口）。

端口驱动程序从微型端口驱动程序隐藏了操作系统的许多细节，而微型端口驱动程序从端口驱动程序隐藏了底层硬件的细节。端口驱动程序的实现可能会针对不同的操作系统版本进行更改，但是端口驱动程序与微型端口驱动程序的接口或多或少保持不变，从而使微型端口驱动程序在很大程度上与平台无关。

**迷你驱动程序（通用）**

微型驱动程序（通用）代表总线上的硬件组件。微型驱动程序使用总线驱动程序通过总线与物理设备进行通信，并将总线驱动程序和一个或多个类驱动程序绑定在一起。

*类驱动程序*帮助微型*驱动程序*将物理设备作为一种逻辑设备呈现给客户端。在WDM环境中，微型驱动程序通常从类驱动程序接收IRP形式的请求，并将IRP形式的请求发送到总线驱动程序。

微型驱动程序可能还必须与几个类驱动程序进行通信。绑定到多个类驱动程序的微型驱动程序的一个示例是IEEE 1394总线上的CD-ROM驱动器的微型驱动程序。它可能绑定到文件系统驱动程序，以便可以从文件系统访问该驱动器。但是，它也绑定到[Redbook系统驱动程序，](https://docs.microsoft.com/en-us/windows-hardware/drivers/audio/kernel-mode-wdm-audio-components#redbook_system_driver)以便可以从CD传输音频。

**总线驱动程序（通用）**

总线驱动程序（通用）使微型驱动程序可以访问物理总线。Microsoft Windows *硬件抽象层（HAL）*有时也称为*系统总线驱动程序，*因为它提供了对系统总线的访问。有关更多信息，请参见[总线驱动程序](https://docs.microsoft.com/windows-hardware/drivers/kernel/bus-drivers)。

**类驱动程序（通用）**

类驱动程序（泛型）实现了在类似设备类之间通用的行为。

类驱动程序：

* 消除了硬件特定驱动程序中的功能重复。
* 不是特定于总线的。
* 不了解资源问题（例如DMA和中断）。

**微型端口驱动程序（WDM音频）**

微型端口驱动程序（WDM音频）为驻留在系统总线上的音频适配器卡上的功能实现特定于功能的接口。微型端口驱动程序是适配器驱动程序的组件。操作系统不将其识别为驱动程序。在这一点上，音频微型端口驱动程序不同于通用微型端口驱动程序。

与通用微型端口驱动程序不同，音频微型端口驱动程序不实现*[DriverEntry](https://docs.microsoft.com/windows-hardware/drivers/ddi/content/wdm/nc-wdm-driver_initialize)*，未注册，也不完全依赖于其各自的端口驱动程序来提供支持。可以将解决多种功能的多个音频微型端口驱动程序链接到单个适配器驱动程序（并与单个设备对象关联）。

**适配器驱动程序（WDM音频）**

适配器驱动程序（WDM音频）充当与给定适配器关联的所有微型端口驱动程序的容器。该适配器驱动程序被操作系统识别为驱动程序，并且包含在其自己的.sys文件中。

音频适配器驱动程序由一组微型端口驱动程序和其他解决初始化问题的代码组成。例如，适配器驱动程序实现了*[DriverEntry](https://docs.microsoft.com/windows-hardware/drivers/ddi/content/wdm/nc-wdm-driver_initialize)*入口点。

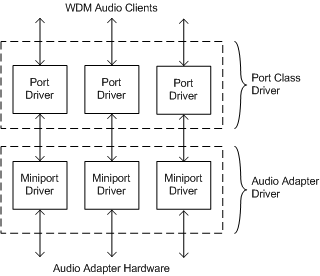
**端口驱动程序（WDM音频）**

端口驱动程序（WDM音频）代表微型端口驱动程序实现KS筛选器，并在端口类驱动程序的上下文中运行。端口驱动程序将微型端口驱动程序的特定于功能的代码作为KS筛选器公开给系统，并负责实现与适配器无关的功能。

与通用端口驱动程序不同，音频端口驱动程序共享设备对象，因此具有不同的实例化。音频端口驱动程序比通用端口驱动程序更类似于通用类驱动程序，因为它实现了一类设备所期望的行为（它与总线无关）。

**端口类驱动程序（WDM音频）**

端口类驱动程序（WDM音频）用作一组端口驱动程序的容器，每个端口驱动程序都支持不同类型的音频硬件功能。下图显示了音频端口类和适配器驱动程序之间的关系。



适配器驱动程序管理可能包含几种不同硬件功能的适配器卡。如上图所示，适配器驱动程序包含一个微型端口驱动程序，用于管理每种类型的硬件功能。同样，端口类驱动程序旨在为具有多种硬件功能的适配器卡提供支持。端口类驱动程序为其支持的每种定义良好的功能类型提供端口驱动程序。适配器驱动程序将其用于特定功能的微型端口驱动程序绑定到该功能类型的相应端口驱动程序。每个功能的端口驱动程序处理与使用该功能的WDM音频客户端的通信。微型端口驱动程序包含用于管理该功能的所有特定于硬件的代码。

端口类驱动程序（WDM音频）主要充当与单个设备对象关联的多个子设备的容器。总线驱动程序为其枚举的每个即插即用（PnP）节点创建一个*物理设备对象（PDO）*。

对于音频适配器，单个PnP节点通常包含多个音频功能。要将与节点关联的各种功能公开为不同的设备，通常需要编写适配器的总线驱动程序。总线驱动程序枚举硬件功能并创建相应的PDO。在这种情况下，一个或多个特定于功能的驱动程序需要绑定到PDO，并与总线驱动程序进行协商以访问适配器上的共享资源。

端口类驱动程序使用内核流驱动程序的功能来公开单个设备对象的各个方面，以便操作系统将设备识别为一组不同的子设备。

在设备名称后附加参考字符串以指定所需的子设备。内核流驱动程序根据此参考字符串调度创建IRP。创建文件对象后，内核流驱动程序将分派针对代表子设备的文件对象的IRP。另外，端口类驱动程序实现了用于包装子设备的基于COM的模型。

适配器驱动程序实例化端口驱动程序和微型端口驱动程序，并通过将指向微型端口驱动程序的指针作为端口驱动程序的初始化函数的参数传递，将它们绑定在一起（请参见[Subdevice Creation中](https://docs.microsoft.com/en-us/windows-hardware/drivers/audio/subdevice-creation)的代码示例）。生成的端口/微型驱动程序堆栈构成一个KS过滤器，该过滤器表示端口类驱动程序支持的子设备类型之一。

端口类驱动程序的**[PcRegisterSubdevice](https://docs.microsoft.com/windows-hardware/drivers/ddi/content/portcls/nf-portcls-pcregistersubdevice)**函数注册子设备，该子设备被系统的其余部分视为设备。端口驱动程序接收针对设备对象的创建IRP，但仅针对注册子设备的参考字符串所指定的那些IRP。端口驱动程序还接收针对与子设备关联的文件对象的IRP。端口驱动程序负责子设备作为KS筛选器的行为，并负责与微型端口驱动程序进行适当的通信。

有关设计多功能声卡驱动程序的更多信息，请参见[多功能音频设备](https://docs.microsoft.com/en-us/windows-hardware/drivers/audio/multifunction-audio-devices)。

**样本音频驱动程序**

* 2017/04/20
* 2分钟阅读
  + [[https://github.com/DOMARS.png?size=32](https://github.com/MicrosoftDocs/windows-driver-docs/blob/staging/windows-driver-docs-pr/audio/sample-audio-drivers.md)](https://github.com/MicrosoftDocs/windows-driver-docs/blob/staging/windows-driver-docs-pr/audio/sample-audio-drivers.md" \o "1个贡献者)

**SYSVAD音频样本**

**系统虚拟音频设备驱动程序样本（SYSVAD）**

SYSVAD驱动程序突出了WDM音频体系结构的许多重要功能。这些是带有源代码的有效实现，可以作为编写专有音频设备的自定义驱动程序的起点。

该*sysvad*解决方案文件包含以下项目。

* **TabletAudioSample**

该*TabletAudioSample*项目演示如何开发一个WDM音频驱动程序暴露出多种音频设备的支持。这些音频设备中的一些嵌入在系统中（扬声器，麦克风阵列），而其他一些则可插拔（耳机扬声器/麦克风，蓝牙耳机扬声器/麦克风）。该驱动程序将WaveRT和音频分载用于渲染设备。该驱动程序使用“虚拟音频设备”代替实际的基于硬件的适配器，并重点介绍了音频卸载WDM音频驱动程序体系结构的不同方面。有关Windows音频引擎的更多信息，请参见[硬件分载的音频处理（Windows驱动程序）](https://docs.microsoft.com/en-us/windows-hardware/drivers/audio/hardware-offloaded-audio-processing)。

* **PhoneAudioSample**

该*PhoneAudioSample*项目是非常相似的*TabletAudioSample*项目。它包括针对移动设备的优化。

* **通用端点**

该*EndpointsCommon*项目包含通用代码的平板电脑和手机两种。有关更多信息，请参见[通用Windows音频驱动程序](https://docs.microsoft.com/en-us/windows-hardware/drivers/audio/audio-universal-drivers)。

* **交换APO**

该*SwapAPO*项目演示如何开发音频处理对象。它包括示例代码，该代码演示了如何注册和注销音频处理对象，还显示了如何自定义“控制面板”属性页以反映处理对象中的可用功能。有关更多信息，请参见[Windows音频处理对象](https://docs.microsoft.com/en-us/windows-hardware/drivers/audio/windows-audio-processing-objects)。

* **KeywordDetectorAdapter**

该*KeywordDetectorAdapter*项目演示如何开发关键字检测适配器。有关更多信息，请参阅[语音激活](https://docs.microsoft.com/en-us/windows-hardware/drivers/audio/voice-activation)。

**从GitHub下载并提取Sysvad音频样本**

可以在[Windows Driver Samples GitHub上](https://github.com/Microsoft/Windows-driver-samples)获得SYSVAD音频示例。

您可以在此处浏览Sysvad音频示例：

<https://github.com/Microsoft/Windows-driver-samples/tree/master/audio/sysvad>

请按照以下步骤下载并打开SYSVAD示例。

一种。您可以使用GitHub工具来处理示例。您也可以在一个zip文件中下载通用驱动程序样本。

<https://github.com/Microsoft/Windows-driver-samples/archive/master.zip>

b。将master.zip文件下载到本地硬盘驱动器。

C。右键单击*Windows-driver-samples-master.zip*，然后选择**全部提取**。指定一个新文件夹，或浏览到将存储提取的文件的现有文件夹。例如，您可以指定*C：\ DriverSamples \*作为将文件提取到其中的新文件夹。

d。提取文件后，导航到以下子文件夹。

*C：\ DriverSamples \ Audio \ Sysvad*

**在Visual Studio中打开驱动程序解决方案**

在Microsoft Visual Studio中，单击“ **文件”** >“ **打开”** >“ **项目/解决方案...”**，然后导航到包含所提取文件的文件夹（例如*C：\ DriverSamples \ Audio \ Sysvad*）。双击*Sysvad*解决方案文件将其打开。

在Visual Studio中，找到“解决方案资源管理器”。（如果尚未打开，请从“ **视图”**菜单中选择“ **解决方案资源管理器** ” 。）在解决方案资源管理器中，您可以看到一个包含六个项目的解决方案。

**存档的音频样本**

这些音频样本支持Microsoft Windows驱动程序工具包（WDK）的早期版本。它们是可从[此处](https://go.microsoft.com/fwlink/p/?LinkId=618052)下载的zip文件的一部分。

* **Microsoft虚拟音频设备驱动程序样本（Msvad）**
* **AC97驱动程序（Ac97）**
* **DirectMusic UART驱动程序样本（Dmusuart）**
* **DirectMusic软件合成器样本（ddksynth）**
* **FM合成器（Fmsynth）**
* **音频适配器样品**

**音频处理编解码器样本**

* **Msfilter样本编解码器（MsFilter）**
* **Msgsm610示例编解码器（gsm610）**

有关详细信息，请参阅WDK中每个样本随附的自述文件。

有关WDK示例的信息，请参见[Windows驱动程序工具包示例包（Windows驱动程序）。](https://docs.microsoft.com/windows-hardware/drivers/samples/index)

**KsStudio实用程序**

* 2017/04/20
* 2分钟阅读
  + [[https://github.com/DOMARS.png?size=32](https://github.com/MicrosoftDocs/windows-driver-docs/blob/staging/windows-driver-docs-pr/audio/ksstudio-utility.md)](https://github.com/MicrosoftDocs/windows-driver-docs/blob/staging/windows-driver-docs-pr/audio/ksstudio-utility.md" \o "1个贡献者)

Microsoft Windows驱动程序工具包（WDK）中的软件工具包含KsStudio实用程序（Ksstudio.exe）。KsStudio是内核流工具，可用于在Windows Me / 98和Windows 2000及更高版本中构建，检查和测试WDM KS过滤器图。KsStudio最有用的功能之一是能够构建过滤器图形的图形表示，该图形显示过滤器之间的引脚到引脚连接以及过滤器的内部节点。尽管KsStudio主要设计用于音频滤波器图，但它可用于构建和浏览包含任何类型的WDM KS滤波器的图。

要安装KsStudio及其帮助文件KsStudio.chm，请执行以下操作：

* 运行WDK安装程序。
* 从组件列表中，单击“ **驱动程序开发工具包”**，单击**“驱动程序开发人员工具”**，然后单击“ **音频/视频流工具”**。

安装程序在\ tools \ avstream目录的x86，amd64和ia64子目录中安装特定于平台的KsStudio.exe版本。它将帮助文件KsStudio.chm安装在\ tools \ avstream目录中。要打开帮助文件，请双击KsStudio.chm。

Windows 2000和Windows Me驱动程序开发工具包（DDK）中包含KsStudio的较早版本，名为Grapher（Grapher.exe）。

有关KsStudio的更多信息，请参见WDK中的KsStudio实用程序帮助文件。