

## RE: AVStream INF files + Chkinf + Fastinst = Brain-ache!!

**Source:**

<http://www.tech-archive.net/Archive/Development/microsoft.public.development.device.drivers/2004-04/0355.html>

---

**From:** Richard Fricks (*richfr\_nospam\_at\_microsoft.com*)

**Date:** 04/05/04

Date: Mon, 05 Apr 2004 22:12:46 GMT

Hi John,

Here are some answer for you:

1. I tried using Walter Oney's FASTINST. Although the driver was installed, it wasn't installed correctly. KSSstudio revealed its presence but without registry-based information. Furthermore, the device did not show up in Graphedt. (TestCap from the Win2K DDK works fine).

*>I'm not sure what FASTINST does so I can't comment on it directly however, I can contrast the different between avssamp and testcap which should help clear this up. Both are virtual drivers – no hardware actually exists on the computer – but they each register themselves with a different Bus Driver and each bus driver has a different mechanism for enumerating the device and therefore the INF's are noticeably different. AVSSamp installs itself on the Software Bus and therefore follows the SWENUM INF specification. TestCap installs itself on the ROOT Bus and therefore follows that INF convention. TestCap is a traditional INF in this regard and does not have some of the "Special" entries that SWENUM requires and this is most likely why FASTINST do not work with it. See below comments for more details*

2. So I tried the Add Hardware wizard. It wouldn't show a list of devices when the avssamp.inf file was selected.

*>This method does not work with the software bus. It will work with ROOT, PCI, USB, and other "real" busses. Note that the Software bus is a "virtual" bus as the bus driver was written to enable the installation of drivers that had no other way of being enumerated. The AVSSamp could have been designed as a ROOT enumerated device and TestCap could have been designed as a SWENUM enumerated device as it is somewhat of a arbitrary decision on which method you want to use for virtual devices. I prefer the SWENUM method as installation is easier (once you know how it works :)).*

3. I ran Chkinf on the inf file (as supplied with the DDK) and it gave

microsoft.public.development.device.drivers: RE: AVStream INF files + Chkinf + Fastinst = Brain-ache!!

errors about not using layout.inf and layout1.inf (should only be used by Microsoft engineering).

>*ChkInf can be helpful at times and not helpful at other times. This is one of those other times.*

4. Did a Google 'groups' search and found a useful thread in which Walter Oney stated that the simple right-click/Install option should work.

Well, Walter was right (not unexpectedly – great book!) but it left me very confused about the whole inf file business.

>*Yes, SWENUM based INF's like being right clicked on and is the method that should be used.*

Is there a really good, digestible source of knowledge for the finer points of inf files? I don't really like to just copy/paste bits from existing ones unless I know what I am doing and why!

>*The key lines in the avssamp.inf are:*

```
[avssamp.RunOnce.AddReg]
HKLM,%RunOnce%,"avssamp.Reader.Capture",,"rundll32.exe
streamci,StreamingDeviceSetup
%avssamp.DeviceId%,%KSSTRING_Filter%,%KSCATEGORY_CAPTURE%,%17%\avssamp.inf,a
vssamp.Reader.Install"
HKLM,%RunOnce%,"avssamp.Reader.Capture",,"rundll32.exe
streamci,StreamingDeviceSetup
%avssamp.DeviceId%,%KSSTRING_Filter%,%KSCATEGORY_AUDIO%,%17%\avssamp.inf,avs
samp.Reader.Install"
```

```
[avssamp.Reader.Install]
AddReg=avssamp.Reader.AddReg
```

```
[avssamp.Reader.AddReg]
HKR,,CLSID,,%Proxy.CLSID%
HKR,,FriendlyName,,%avssamp.Reader.FriendlyName%
```

In the DDK docs do a search on "StreamingDeviceSetup" and you will find the documentation on it. What you will find is this is the method that must be used if you are installing a SWENUM enumerated driver. This entry will load the streamci.dll and call the entry point StreamDeviceSetup() in this DLL passing in the other RunOnce parameters to this function. In turn this function will process the [avssamp.Reader.AddReg] section which will load the DShow reg entry information so that it will be visible to DShow as well as register the device interfaces specified by the lines:

```
%avssamp.DeviceId%,%KSSTRING_Filter%,%KSCATEGORY_CAPTURE%
```

This device interface registration is what will allow the device to be visible to the system components that search for devices via the Device

RE: AVStream INF files + Chkinf + Fastinst = Brain-ache!!

microsoft.public.development.device.drivers: RE: AVStream INF files + Chkinf + Fastinst = Brain-ache!!

Categories listed here (the audio system requires all audio drivers register via KSCATEGORY\_AUDIO for example).

Inf's are deceptively complicated and I hope this helps clear things up at least a little.

Look me up if you are coming to WinHec2004 in May.

Thanks,  
Richard Fricks  
Audio DDK – Microsoft