

Re: Confused about Kernel Streaming / DShow clocks...

Source:

<http://www.tech-archive.net/Archive/Development/microsoft.public.development.device.drivers/2004-09/0901.html>

From: Geraint Davies (*geraintd_at__NO_THANK_YOU_REALLY_gdcl.co.uk*)

Date: 09/18/04

Date: Sat, 18 Sep 2004 11:49:12 +0100

It's been a few years, so I'm a bit rusty, but perhaps I can remember a few things that would help.

If you are a stream class minidriver, you implement the clock using the HW_STREAM_OBJECT entries that you mention, and you will query the clock using StreamClassQueryMasterClockSync. The stream class drivers implementation is layered on top of KS, so it will handle the KS_ stuff for you.

A few pointers from memory:

— you need to implement the event function as well, for KSEVENT_CLOCK_POSITION_MARK and KSEVENT_CLOCK_INTERVAL_MARK. A clock triggers these events as well as reporting the time.

— remember that the time on a KS clock is what a DirectShow clock would think of as Stream Time, not absolute reference time.

— you don't get an "indicate-clock" if you get an "open clock" srb, so when getting the time, you need to check if your clock is opened first and call it directly.

— the normal way to timestamp is to get the performance counter at interrupt time, then at DPC time you call StreamClassQueryMasterClockSync. This reports the clock time and the current performance counter time so you can extrapolate to what the stream time was at interrupt.

G

"Martin Harvey (work)" <m.harvey@nospam.snellwilcox.com> wrote in message news:%232F0aNMnEHA.3076@TK2MSFTNGP15.phx.gbl...

> *Hi folks,*

>

> *Just a quick post to see if someone can clarify my understanding of kernel streaming clocks. – I think I've got the basics, but there are a few loose*

> ends...
>
> I've got a multimedia card with separate audio and video capture drivers
> written under the stream class driver. At the moment, the captured audio &
> video has a habit of drifting out of sync, so I need to put clock support
> into the drivers. Additionally, the hardware has 27 MHz counter that can
> be
> queried to give me a very accurate clock source.
>
> As I currently understand it:
>
> a) I need to get both audio and video capture drivers to support being a
> master clock.
> b) I need to get the drivers to be able to timestamp samples based on the
> master clock for the graph (which might be themselves, or might be from
> somewhere else).
>
> Here's where I have a few questions...
>
> Q'n 1. It seems that there seem to be several ways of doing "a)" and "b)"
>
> – MSDN seems to indicate that I can do a) by including information in my
> HW_STREAM_OBJECT indicating that I can act as a master clock, by including
> appropriate clock support flags, and a callback routine (HW_CLOCK_OBJECT).
> – If I do that, then my clock support is going via the stream class
> driver.
> It then seems the the "similar" way of implementing b) is to use the
> "StreamClassQueryMasterClock" routine, using the clock handle passed to my
> driver via SRB_INDICATE_MASTER_CLOCK.
>
> Make sense so far?
>
> If so, then why is there also a whole load of KS Property stuff, which
> provide similar functionality? It seems that if I want to provide a master
> clock source via the KsProperty route, then I'll have to implement
> KSPROPSETID_Clock, and also KS_PROPERTY_STREAM_MASTERCLOCK.
>
> Do I have to do both of these, or is it an either/or proposition, with the
> stream class driver doing some behind the scenes work for me in the first
> case?
>
> Qn 2: If I have the two filters running, and both can provide a master
> clock, then the audio capture driver will provide the master clock of the
> graph – so I need to get the other driver to timestamp samples based on
> the
> audio driver. StreamClassQueryMasterClock would seem to do the job if I
> want
> to use stream.sys, however, it's asynchronous, and it doesn't seem
> sensible
> to try and perform such a query every time a get a media sample. Any
> suggestions on strategies for getting the drivers to query the master

microsoft.public.development.device.drivers: Re: Confused about Kernel Streaming / DShow clocks...

- > *clock*
- > *at intervals and then interpolate? How is this normally done?*
- >
- > *Qn3: If I was to use the property set approach, instead of*
- > *StreamClassQueryMasterClock, it seems that I need to get on driver to get*
- > *hold of the direct call interface (KSPROPERTY_CLOCK_FUNCTIONTABLE)*
- > *advertised by the other driver... how do I do that?*
- >
- > *thanks in advance,*
- >
- > *Martin Harvey.*
- >
- >