

Public Support - Support Request #3368

How does ADTF handles device timestamps ?

2018-07-17 16:25 - hidden

Status:	Closed	
Priority:	Normal	
Category:		
Customer:	AUDI	Product Issue Numbers:
Department:	EF	Affected Products: ADTF 2.14.2, ADTF Device Toolbox 2.7.0
Requester's Priority:	Low	Platform: Windows 7 64bit
Support Level:	2nd Level	Topic: DeviceTB::Flexray
Resolution:	Solved Issue	FAQ Links:

Description

Supportanfrage

Wird Sync2RefTime automatisch auf FlexRay-Streams (ADTF2) während

- Playback mittels Harddisk_Player
- Tools > DAT Exporter > MDF4 angewendet?

Trotz jitteriger MediaSample-Time im DAT-File scheinen im Playback/Export die Zeitstempel wesentlich "glatter" zu sein.

Lösung

Bsp: Flexray

Sample

> Ein Sample besteht aus mehreren Frames/PDUs

> Sample hat einen Zeitstempel (Transmit)

-> Bei jedem Frame/PDU/Signal gibt es somit einen leicht versetzten Zeitstempel anhand der Device Time, synchronisiert auf die ADTF Stream Time (=Media Sample)

CSV Export:

-> Exportiert jedes Sample und dessen Zeitstempel anhand der Media Description -> tFlexayData

MDF Export:

-> Exportiert **alle** Signale die in **einem** Sample hinterlegt sind -> anhand des fibex files

Die Normierung im Export hat nichts mit Sync2Ref zu tun.

Sync2Ref bedeutet, ich habe unterschiedliche Devices mit jeweils eigener Zeitgeber.

Um nun alle Daten im ADTF auf eine Zeitbasis zu harmonisieren, nutzt du Sync2Ref auf die StreamTime in ADTF (Laufzeit).

Ein Bsp. findest du in der Device Toolbox, dort macht das Demo Device.

Im Export hast du keine Stream Time, aber die PDUs/Frames/Signale haben nicht den gleichen Zeitstempel (den des Media Samples), denn sie kommen ja nicht zeitgleich.

Würdest du normal exportieren (CSV Exporter, Media Sample basiert), dann nimmst du den Zeitstempel des Media Samples (das müsste imho je nach Setup den ersten, letzten oder Mittelwert aller Signale eines Samples entsprechen).

Um nun die genauen Zeitstempel der Signale zu bekommen, werden sie gegen den Zeitstempel des Media Samples berechnet, wenn du so willst, dahingehend geglättet.

```
tResult cMDFFlexraySampleSink::Process(adtf::IMediaSample* pMediaSample, tTimeStamp nChunkTime)
{
    // [...]

    for (cFlexRayFrameVector::cIterator it = oFrameVector.GetFirst(); it.IsValid(); ++it)
    {
        const tFlexRayData* pData = *it;
        switch (pData->nTag)
```

```

    {
        case tFlexRayData::FR_TAG_CYCLESTART:
            break;

        case tFlexRayData::FR_TAG_FRAME:
            Process_Frame(pMediaSample->GetTime() + pData->nTimeStamp - nHWTime,
                pData->sFrame.nFlags, pData->sFrame.nChannel,
                pData->sFrame.nCycleCount, pData->sFrame.nSlot,
                pData->sFrame.nPayloadLength, pData->sFrame.aData);
            break;

        case tFlexRayData::FR_TAG_PDU:
            Process_PDU(pMediaSample->GetTime() + pData->nTimeStamp - nHWTime,
                pData->sPDU.nPDUID, pData->sPDU.nChannel, 0, 0,
                pData->sPDU.nPayloadLength, pData->sPDU.aData);
            break;

        default:
            break;
    }
}

// [...]
}

```

Das hat aber nichts mit dem Sync2Ref unterschiedlicher Zeitgeber zu tun, denn diese Signale sind schon gesyncnt.

Wenn dein Sample Aufbau so wäre, dass nur ein Signal anliegt, dann könntest du dir diesen Schritt sparen.
Demnach hilft dir da der EB Service auch nicht weiter, da du keine Laufzeit hast.

Um mit den EB FR Sample Typ MDF zu exportieren, müsstest du eine Exporter Sample Sink schreiben, die genau das gleiche macht wie die der Device TB, nur eben mit dem EB Datentyp umgehen kann.
Alternativ könntest du diesen Datentyp einfach nur Mappen (gibt es da nicht sogar einen Converter Filter ?), dann kannst du den MDF-Exporter weiter nutzen.

History

#1 - 2018-07-18 09:23 - hidden

- Topic set to DeviceTB::Flexray

#2 - 2018-07-18 13:33 - hidden

- Status changed from New to In Progress

Hallo Patrick,

Sync2RefTime hat nichts mit dem Harddisk Player oder dem DAT Exporter zu tun.
Sync2RefTime soll unterschiedliche Zeitquellen oder Eingangszeitstempel auf die Stream-Time der ReferenceClock anpassen bzw. abgleichen.
Das ist hier im ADTF 2 SDK beschrieben: [Sync2RefTime](#)
Oder auch hier nochmal bei den [ADTF Timings](#) beschrieben.

Beantwortet das deine Frage?

#3 - 2018-07-18 13:34 - hidden

- Status changed from In Progress to Customer Feedback Required

#4 - 2018-07-18 13:58 - hidden

Bspw. die aufgezeichneten Media-Sample-Timestamps eines FlexRay-Signals aus einem DAT-File:

```

0.0209
0.0186
0.0200
0.0199
0.0214

```

...und aus dem MDF4-Export:

```

0.0199

```

0.0200
0.0200
0.0200
0.0200

Beim MDF4-Export sehen die Werte "entjittert" aus. Passiert da etwas in der Richtung Sync2RefTime, wenn auch ohne die Funktion Sync2RefTime selbst?

#5 - 2018-07-18 15:56 - hidden

- Status changed from Customer Feedback Required to In Progress

Hallo Florian,
kannst Du dir das bitte mal ansehen?

#6 - 2018-07-18 16:17 - hidden

- Status changed from In Progress to Customer Feedback Required

Hallo Patrick,

welche Werte vergleichst du hierbei und v.a. in welchen Tooling ?
Während Playback, Showdatfileinfo, etc. ?

Bedenke dass im DAT File beim Sample der Zeitstempel des Samples steht.
Hinter einem Sample befindet sich ein Vector mit mehreren Frames/PDUs, die alle einen Zeitstempel haben (minimaler Drift, Hardwarezeit).

Beim MDF Export z.B. wird dieser mit berücksichtigt, im Showdatfileinfo siehst du ihn nicht (hier nur Media Sample), im Trace View müssten die Werte in etwa identisch sein.

#7 - 2018-07-19 10:40 - hidden

Hallo Florian,

ich vergleiche die ADTF-Exports (Tools > DAT Exporter) CSV mit MDF4:

MDF4-Export	CSV-Export
9617.47666	9617.477125
9617.49656	9617.498072
9617.51655	9617.516660

Ich gehe davon aus, daß ich beim CSV-Export die Media-Sample-Timestamps original wie im DAT-File gespeichert erhalte.

Beim MDF4-Export sind die Zeitstempel leicht anders, vielleicht max. 1 bis 2 ms. Wenn man die Zeitdeltas berechnet, sieht man, daß die MDF4-Zeitstempel weniger Jitter enthalten als die Media-Sample-Timestamps. Die Zeitdeltas sind bei diesem Signal fast stets genau 20 ms entsprechend dem Soll-Takt.

Werden beim MDF4-Export (leicht) angepaßte/korrigierte Zeitstempel auf Basis der Media-Sample-Zeit und unter Zuhilfenahme der Frame-Hardware-Zeitstempel berechnet? Kann man das so sagen?

#8 - 2018-07-19 11:15 - hidden

Hallo Patrick,

das darfst du nicht miteinander vergleichen.

Bsp: Flexray

Sample
> Ein Sample besteht aus mehreren Frames/PDUs
> Sample hat einen Zeitstempel (Transmit)
> Bei jedem Frame/PDU/Signal gibt es somit einen leicht versetzten Zeitstempel anhand der Device Time, synchronisiert auf die ADTF Stream Time (=Media Sample)

CSV Export:
> Exportiert jedes Sample und dessen Zeitstempel anhand der Media Description -> tFlexayData

MDF Export:
> Exportiert alle Signale die in einem Sample hinterlegt sind -> anhand des fibex files

#9 - 2018-07-19 16:54 - hidden

Bei jedem Frame/PDU/Signal gibt es somit einen leicht versetzten Zeitstempel anhand der Device Time, synchronisiert auf die ADTF Stream Time (=Media Sample)

Also schon eine Art "Sync2RefTime"?

Würde man den DAT Exporter erweitern um MDF4-Export "Elektrobit FlexRay" (MajorType 125932, SubType 1), was müßte man tun, um das gleiche Verhalten dorthin zu übertragen? Die Struktur und die Device-Zeitstempel sind dort anders. Es scheint aber einen ADTF-Service "EBClockSync" zu geben:

The EB ClockSync Service is an ADTF service allowing plugins to adjust ADTF media timestamps of the captured EB media sample data regarding the hardware time. Service is capable to use FlexRay or UTC hardware timestamps present in EB media samples.

Service helps to correct or reduce effect of following errors in ADTF sample media timestamps:

- Compensation of transmission offsets between video/bus card and ADTF
- Jitter introduced by os scheduling during transmission
- Clock drift between hardware timestamps source and ADTF PC clock

#10 - 2018-07-19 17:44 - hidden

Hallo Patrick,

nochmal, die Normierung im Export hat nichts mit Sync2Ref zu tun.

Sync2Ref bedeutet, ich habe unterschiedliche Devices mit jeweils eigener Zeitgeber.

Um nun alle Daten im ADTF auf eine Zeitbasis zu harmonisieren, nutzt du Sync2Ref auf die StreamTime in ADTF (Laufzeit).

Ein Bsp. findest du in der Device Toolbox, dort macht das Demo Device.

Im Export hast du keine Stream Time, aber die PDUs/Frames/Signale haben nicht den gleichen Zeitstempel (den des Media Samples), denn sie kommen ja nicht zeitgleich.

Würdest du normal exportieren (CSV Exporter, Media Sample basiert), dann nimmst du den Zeitstempel des Media Samples (das müsste imho je nach Setup den ersten, letzten oder Mittelwert aller Signale eines Samples entsprechen).

Um nun die genauen Zeitstempel der Signale zu bekommen, werden sie gegen den Zeitstempel des Media Samples berechnet, wenn du so willst, dahingehend geglättet.

```
tResult cMDFFlexraySampleSink::Process(adtf::IMediaSample* pMediaSample, tTimeStamp nChunkTime)
{
    // [...]

    for (cFlexRayFrameVector::cIterator it = oFrameVector.GetFirst(); it.IsValid(); ++it)
    {
        const tFlexRayData* pData = *it;
        switch (pData->nTag)
        {
            case tFlexRayData::FR_TAG_CYCLESTART:
                break;

            case tFlexRayData::FR_TAG_FRAME:
                Process_Frame(pMediaSample->GetTime() + pData->nTimeStamp - nHWTime,
                    pData->sFrame.nFlags, pData->sFrame.nChannel,
                    pData->sFrame.nCycleCount, pData->sFrame.nSlot,
                    pData->sFrame.nPayloadLength, pData->sFrame.aData);
                break;

            case tFlexRayData::FR_TAG_PDU:
                Process_PDU(pMediaSample->GetTime() + pData->nTimeStamp - nHWTime,
                    pData->sPDU.nPDUID, pData->sPDU.nChannel, 0, 0,
                    pData->sPDU.nPayloadLength, pData->sPDU.aData);
                break;

            default:
                break;
        }
    }

    // [...]
}
```

Das hat aber nichts mit dem Sync2Ref unterschiedlicher Zeitgeber zu tun, denn diese Signale sind schon gesynct.

Wenn dein Sample Aufbau so wäre, dass nur ein Signal anliegt, dann könntest du dir diesen Schritt sparen.

Demnach hilft dir da der EB Service auch nicht weiter, da du keine Laufzeit hast.

Um mit den EB FR Sample Typ MDF zu exportieren, müsstest du eine Exporter Sample Sink schreiben, die genau das gleiche macht wie die der

Device TB, nur eben mit dem EB Datentyp umgehen kann.

Alternativ könntest du diesen Datentyp einfach nur Mappen (gibt es da nicht sogar einen Converter Filter ?), dann kannst du den MDF-Exporter weiter nutzen.

#11 - 2018-07-20 08:23 - hidden

Hallo Florian, vielen Dank für die ausführliche Erläuterung. Jetzt ist mir klar geworden, was beim Export genau passiert.

gibt es da nicht sogar einen Converter Filter ?

Ja, der Filter macht aus dem EB-FR-Stream einen ADTF-Standard-FR-Stream. Allerdings gibt es Wünsche, diesen Umweg zu umgehen.

#12 - 2018-07-25 09:11 - hidden

Hallo Patrick,

Ja, der Filter macht aus dem EB-FR-Stream einen ADTF-Standard-FR-Stream. Allerdings gibt es Wünsche, diesen Umweg zu umgehen.

Ich denke es ist aufwändiger ein komplett neues Datexporter Plugin (MDF Sink für EB FR Type) zu schreiben, also die Daten vorher zu konvertieren, da es das schon gibt.

Ggf. hat EB ja auch einen Exporter ?

Gut, das sprengt den Supportrahmen von uns...

Deine Fragen sollten aber damit geklärt sein ?

#13 - 2018-07-25 10:30 - hidden

Ja, ich kann die Zeitstempel-Korrektur jetzt nachvollziehen.

Nur zu deinem Satz bezgl. Media-Sample-Zeit hätte ich noch eine Frage:

das müsste imho je nach Setup den ersten, letzten oder Mittelwert aller Signale eines Samples entsprechen

Meinst Du "timestamp" aus CCC bzw. FCC? Dort kann man \$FIRST\$, \$LAST\$, \$MEAN\$ auswählen. Der Media-Zeitstempel scheint bei mir den Transmit-Zeitpunkt des Media-Sampels zu meinen, so daß ich anhand der HW-Zeitstempel für jeden Einzel-Frame des FR-Vektors zeitlich zurückgehen muß. Kann ich etwa beim Aufzeichnen von Streams aus dem "Vector_FlexRay_Device" irgendwas umkonfigurieren?

#14 - 2018-07-26 09:38 - hidden

Hallo Patrick,

Meinst Du "timestamp" aus CCC bzw. FCC? Dort kann man \$FIRST\$, \$LAST\$, \$MEAN\$ auswählen.

Genau das meinte ich damit.

Der Media-Zeitstempel scheint bei mir den Transmit-Zeitpunkt des Media-Sampels zu meinen, so daß ich anhand der HW-Zeitstempel für jeden Einzel-Frame des FR-Vektors zeitlich zurückgehen muß. Kann ich etwa beim Aufzeichnen von Streams aus dem "Vector_FlexRay_Device" irgendwas umkonfigurieren?

Nein, beim Device findet ein Sync2Ref statt, die HW Zeit wird sich gemerkt (bei den Signalen) und beim Transmit wird das gesamte Media Sample mit der aktuellen Stream Time gestempelt.

#15 - 2018-08-01 08:43 - hidden

- Description updated

- Status changed from Customer Feedback Required to To Be Closed

- Private changed from Yes to No

- Resolution set to Solved Issue

- Project changed from 11 to Public Support

- Subject changed from Verständnisfrage Zeitstempel to How does ADTF handles device timestamps ?

#16 - 2018-10-09 13:44 - hidden

- Status changed from To Be Closed to Closed