

dSPACE



答Bosch数采、回放系统问 —— 2023.07.17会议备忘录

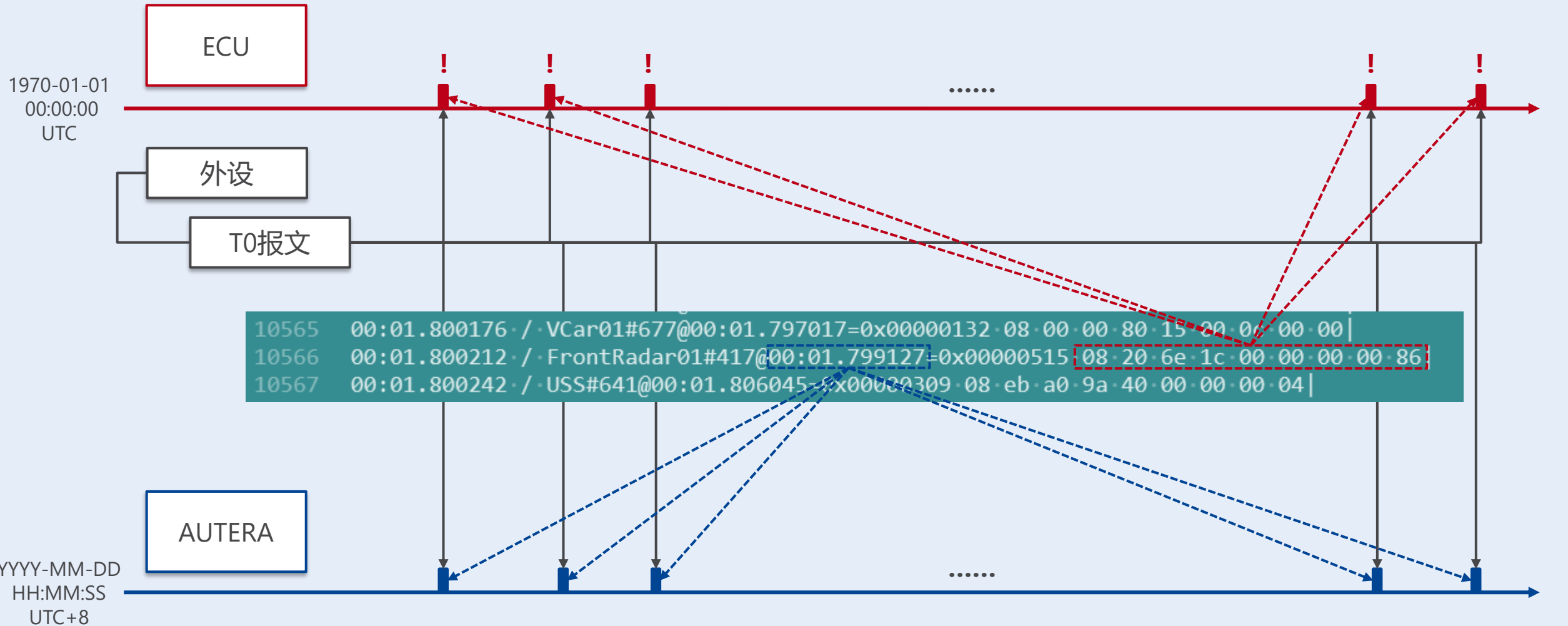
刘翔、廖浪滔, 2023/7/24, 2023/7/26

当前方案下的数采系统与ECU时域关系

两个独立时域

- AUTERA基于本体时钟，对接收之所有数据进行时间戳标记
 - 起点时间：数采程序启动时刻
 - AUTERA经由CAN通道采集所有ECU发送消息，包括：FRM – 0x515 报文（以下简称“T0”报文）
- ECU基于本体时钟
 - 起点时间：ECU上电后为Epoch时间（1970-01-01 00:00:00 UTC）
 - ECU向各毫米波雷达经由CAN消息授时，包括：FRM – 0x515报文
- AUTERA经由对标T0报文，获悉本地时间对应之ECU时间
 - T0报文的内容：ECU时间
 - T0报文的时间戳：AUTERA收到报文时的本地（东八区）时间

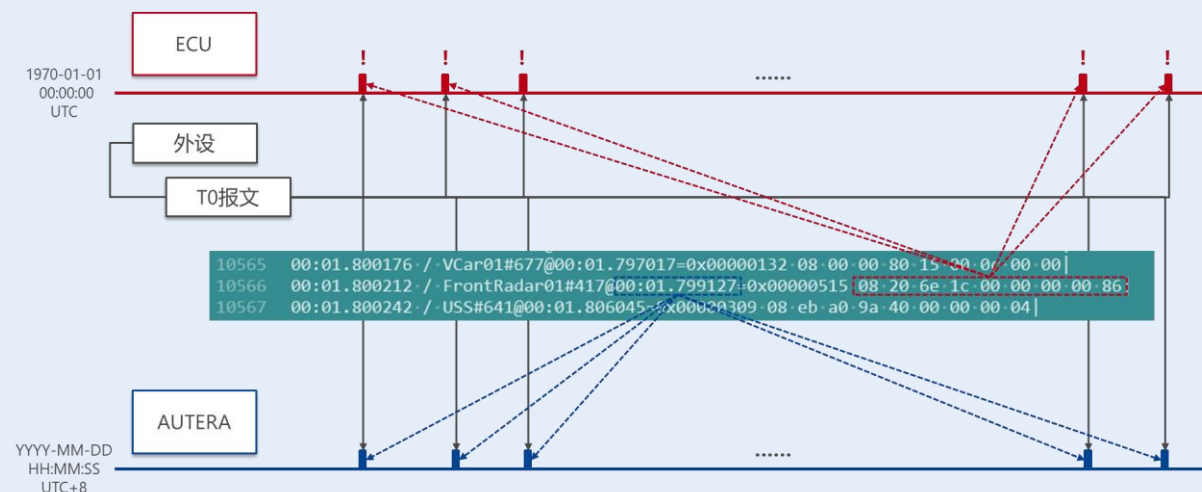
当前方案下的数采系统与ECU时域关系简化示意图



关于T0报文的注意事项

接收到T0报文之时刻 != T0报文内容时间（ECU时间）

- 取决于ECU如何发送T0报文（ECU何时更新T0内容），T0报文出现于总线时间并不严格等价于ECU当时时刻
- T0内容指征之ECU时间 < T0报文被接收之时间
- 当T0报文即时更新，且立时发送，并在总线仲裁延迟时间足够小时，AUTERA标记之T0报文时间戳可用于对齐ECU时间

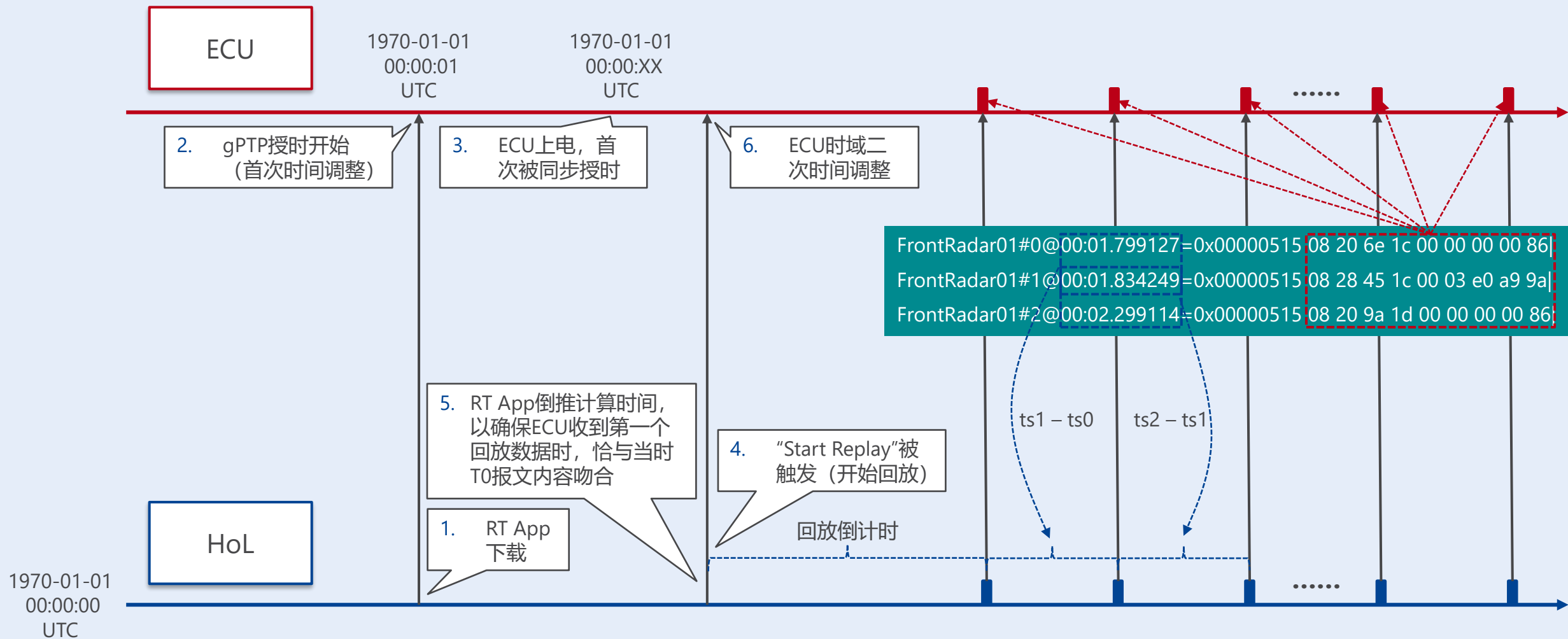


当前方案下的回放系统与ECU时域关系

HoL系统提供两个时域

- HoL本地时域
 - 实时系统的参考时域
 - 时钟频率源: **RTPC RTC**
 - 起点时间: RTPC上电后的Epoch时间 (1970-01-01 00:00:00 UTC)
- ECU时域
 - 基于HoL为主节点的gPTP授时时域
 - 时钟频率源: **RTPC RTC**
 - 起点时间:
 - 当ECU上电, 但尚未开始回放数据时: 可调Epoch时间 (1970-01-01 00:00:01 UTC)
 - 开始回放数据: 回放数据首个有效数据对应之采集时刻 (ECU当时时间) —— T0倒推时间

当前方案下的回放系统与ECU时域关系简化示意图



回放系统的时间信息处理换算示例 [1]

AUTERA REC (原始数据)

Launched at 14:53:12.722 (20/06/2023)
UTC+08:00 - 06:53:12.722 (20/06/2023) UTC

Offset (sec) : 0

[Data]

01:30.814683 / Sensor01 #0@**01:30.814639**
01:31.576782 / Sensor02 #0@**01:31.576762**
01:31.576829 / Bus01 #0@**01:31.576775**
01:31.585162 / Bus02 #0@**01:31.584320**
01:31.589797 / Sensor01 #1@**01:31.588776**
01:31.589909 / Sensor02 #1@**01:31.589435**
01:31.590812 / Bus01 #1@**01:31.590621**
01:31.591963 / Bus02 #1@**01:31.591859**
01:31.592396 / Sensor01 #2@**01:31.592076**
01:31.592599 / Sensor02 #2@**01:31.592184**
01:31.595812 / Bus01 #2@**01:31.595621**
01:31.599701 / Bus02 #2@**01:31.598859**

.....

回放系统时间戳起点归零

[Data]

01:30.814683 / Sensor01 #0@**00:00.000000**
01:31.576782 / Sensor02 #0@**00:00.762123**
01:31.576829 / Bus01 #0@**00:00.762136**
01:31.585162 / Bus02 #0@**00:00.769681**
01:31.589797 / Sensor01 #1@**00:00.774137**
01:31.589909 / Sensor02 #1@**00:00.774796**
01:31.590812 / Bus01 #1@**00:00.775982**
01:31.591963 / Bus02 #1@**00:00.777220**
01:31.592396 / Sensor01 #2@**00:00.777437**
01:31.592599 / Sensor02 #2@**00:00.777545**
01:31.595812 / Bus01 #2@**00:00.780982**
01:31.599701 / Bus02 #2@**00:00.784220**

.....

回放系统时间戳偏置对齐HoL本地时间

Start Replay触发时HoL本地时间 = **00:27.398721**
回放开始倒计时 (Countdown) = **00:10.000000**

[Data]

01:30.814683 / Sensor01 #0@**00:37.398721**
01:31.576782 / Sensor02 #0@**00:38.160844**
01:31.576829 / Bus01 #0@**00:38.160857**
01:31.585162 / Bus02 #0@**00:38.168402**
01:31.589797 / Sensor01 #1@**00:38.172858**
01:31.589909 / Sensor02 #1@**00:38.173517**
01:31.590812 / Bus01 #1@**00:38.174703**
01:31.591963 / Bus02 #1@**00:38.175941**
01:31.592396 / Sensor01 #2@**00:38.176158**
01:31.592599 / Sensor02 #2@**00:38.176266**
01:31.595812 / Bus01 #2@**00:38.179703**
01:31.599701 / Bus02 #2@**00:38.182941**

.....

回放系统的时间信息处理换算示例 [2]

如Hands-On培训时解释过的“时间戳校正”过程

AUTERA REC (原始数据)

Launched at 14:53:12.722 (20/06/2023)
UTC+08:00 - 06:53:12.722 (20/06/2023) UTC
Offset (sec) : 0

[Data]

01:30.814683 / Sensor01	#0@01:30.814639
01:31.576782 / Sensor02	#0@01:31.576762
01:31.576829 / Bus01	#0@01:31.576775
01:31.585162 / Bus02	#0@01:31.584320
01:31.589797 / Sensor01	#1@01:31.588776
01:31.589909 / Sensor02	#1@01:31.589435
01:31.590812 / Bus01	#1@01:31.590621
01:31.591963 / Bus02	#1@01:31.591859
01:31.592396 / Sensor01	#2@01:31.592076
01:31.592599 / Sensor02	#2@01:31.592184
01:31.595812 / Bus01	#2@01:31.595621
01:31.599701 / Bus02	#2@01:31.598859
.....	

回放系统时间戳起点归零

[Data]

01:30.814683 / Sensor01	#0@00:00.000000
01:31.576782 / Sensor02	#0@00:00.762123
01:31.576829 / Bus01	#0@00:00.762136
01:31.585162 / Bus02	#0@00:00.769681
01:31.589797 / Sensor01	#1@00:00.774137
01:31.589909 / Sensor02	#1@00:00.774796
01:31.590812 / Bus01	#1@00:00.775982
01:31.591963 / Bus02	#1@00:00.777220
01:31.592396 / Sensor01	#2@00:00.777437
01:31.592599 / Sensor02	#2@00:00.777545
01:31.595812 / Bus01	#2@00:00.780982
01:31.599701 / Bus02	#2@00:00.784220
.....	

回放系统时间戳偏置对齐HoL本地时间

Start Replay触发时HoL本地时间 = 00:27.398721
回放开始倒计时 (Countdown) = 00:10.000000

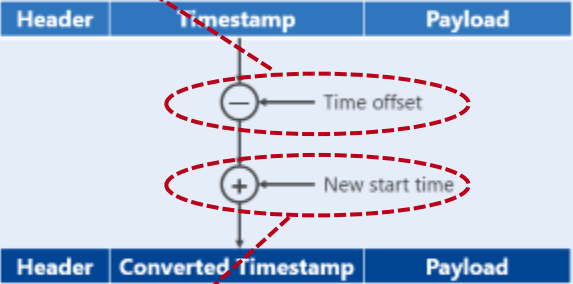
[Data]

01:30.814683 / Sensor01	#0@00:37.398721
01:31.576782 / Sensor02	#0@00:38.160844
01:31.576829 / Bus01	#0@00:38.160857
01:31.585162 / Bus02	#0@00:38.168402
01:31.589797 / Sensor01	#1@00:38.172858
01:31.589909 / Sensor02	#1@00:38.173517
01:31.590812 / Bus01	#1@00:38.174703
01:31.591963 / Bus02	#1@00:38.175941
01:31.592396 / Sensor01	#2@00:38.176158
01:31.592599 / Sensor02	#2@00:38.176266
01:31.595812 / Bus01	#2@00:38.179703
01:31.599701 / Bus02	#2@00:38.182941
.....	

Converting Timestamp for Replay

Replay Data with the Correct Timing

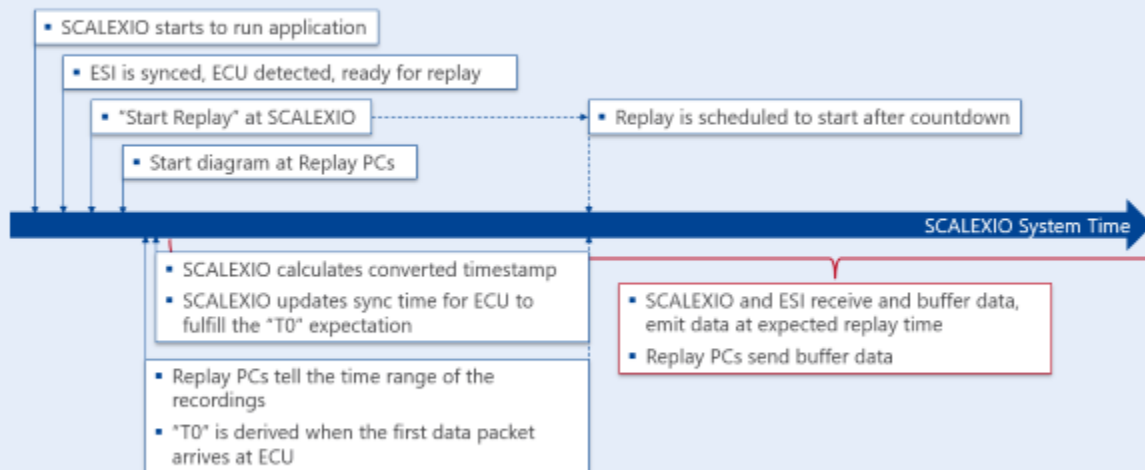
- Original timestamp is tagged when the data is recorded, thus reflects the timing order / interval of each data packet
- Replay system starts based on the SCALEXIO system time, which differs from the recording's
- Time offset is introduced to align the time bases
 - The timestamp of the first data packet will be regarded as the "offset"
- Replayed data starts from time "0"
- A "New start time" is specified to align the "0" time to SCALEXIO system time
- SCALEXIO, as the master and ESI units are all synced with gTP



回放系统的时间信息处理换算示例 [3]

以及何为“倒计时”与其必要性

Brief Overview of Replay Process [2]



Notes for the Replay Process

- User's "to-do" (from the perspective of regular operation)
 - Define **which recording** should be used, **where** to access
 - When to **start replay** at SCLX (and Replay PC (RTMaps))
 - Replay PC / RTMaps **must** be launched within countdown!
 - Replay PC / RTMaps **cannot** be launched before SCLX starts
 - Stop and unload** SCLX as reset for the next replay process
- Why Countdown?**
 - Leave enough time for Replay PC to buffer data at SCLX and ESI so that when replay starts, data is already ready to be emitted
 - To be explicit when the first data packet should be emitted from either SCLX or ESI
 - Leave buffer time for Bosch's in-house tooling to get ready to capture ECU outputs during the replay



回放系统的时间信息处理换算示例 [4]

回放系统时间戳偏置对齐HoL本地时间

Start Replay触发时HoL本地时间 = **00:27.398721**

回放开始倒计时 (Countdown) = **00:10.000000**

[Data]

01:30.814683 / Sensor01 #0@**00:37.398721**

01:31.576782 / Sensor02 #0@**00:38.160844**

01:31.576829 / Bus01 #0@**00:38.160857**

01:31.585162 / Bus02 #0@**00:38.168402**

01:31.589797 / Sensor01 #1@**00:38.172858**

01:31.589909 / Sensor02 #1@**00:38.173517**

01:31.590812 / Bus01 #1@**00:38.174703**

01:31.591963 / Bus02 #1@**00:38.175941**

01:31.592396 / Sensor01 #2@**00:38.176158**

01:31.592599 / Sensor02 #2@**00:38.176266**

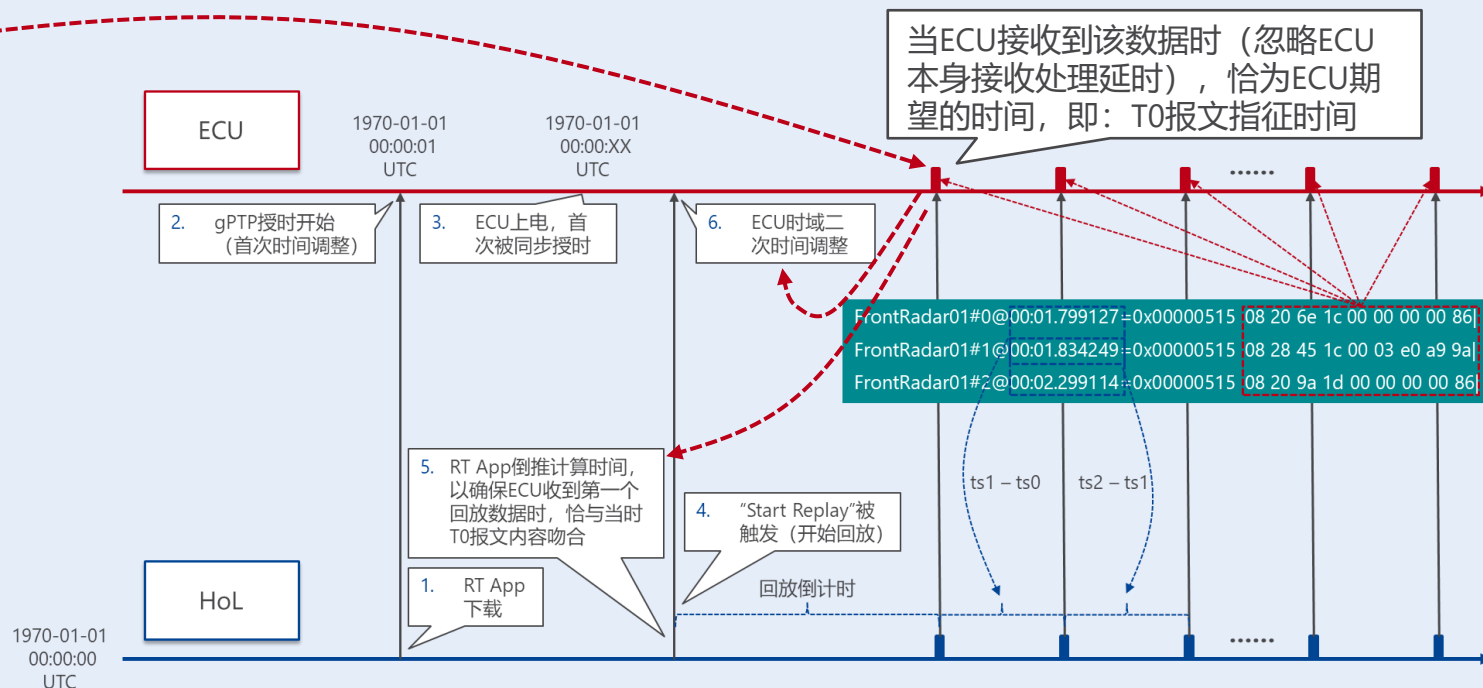
01:31.595812 / Bus01 #2@**00:38.179703**

01:31.599701 / Bus02 #2@**00:38.182941**

.....

重要:

参考系为HoL本地时域, 即: 当HoL本地时钟抵达该时刻时, 相应数据需要被发送



回放系统的时间信息处理换算示例 [5]

一个数学问题:

■ 已知:

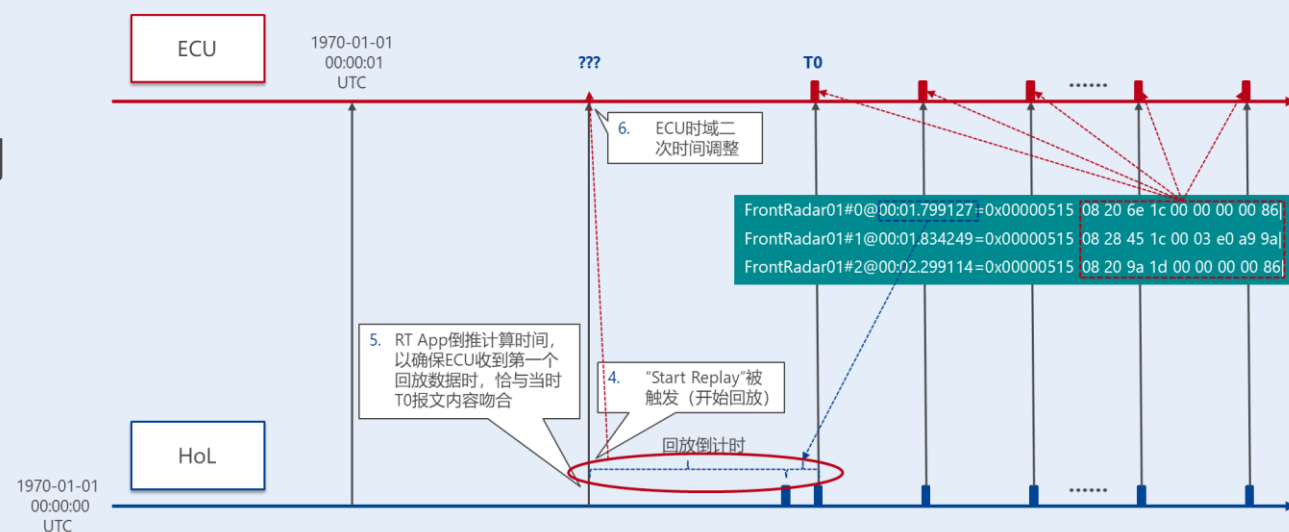
- 在回放数据集的ts_t0时刻将发送FRM-0x515报文
 - 报文内容指征当时ECU应当为T0时刻
- 在HoL被触发Start Replay时, HoL对ECU进行授时调节, 将ECU时间调整为ts_sync
- 当HoL被触发Start Replay后, 过Countdown时间开始回放第一个数据

■ 求:

- ts_sync为何?

■ 解:

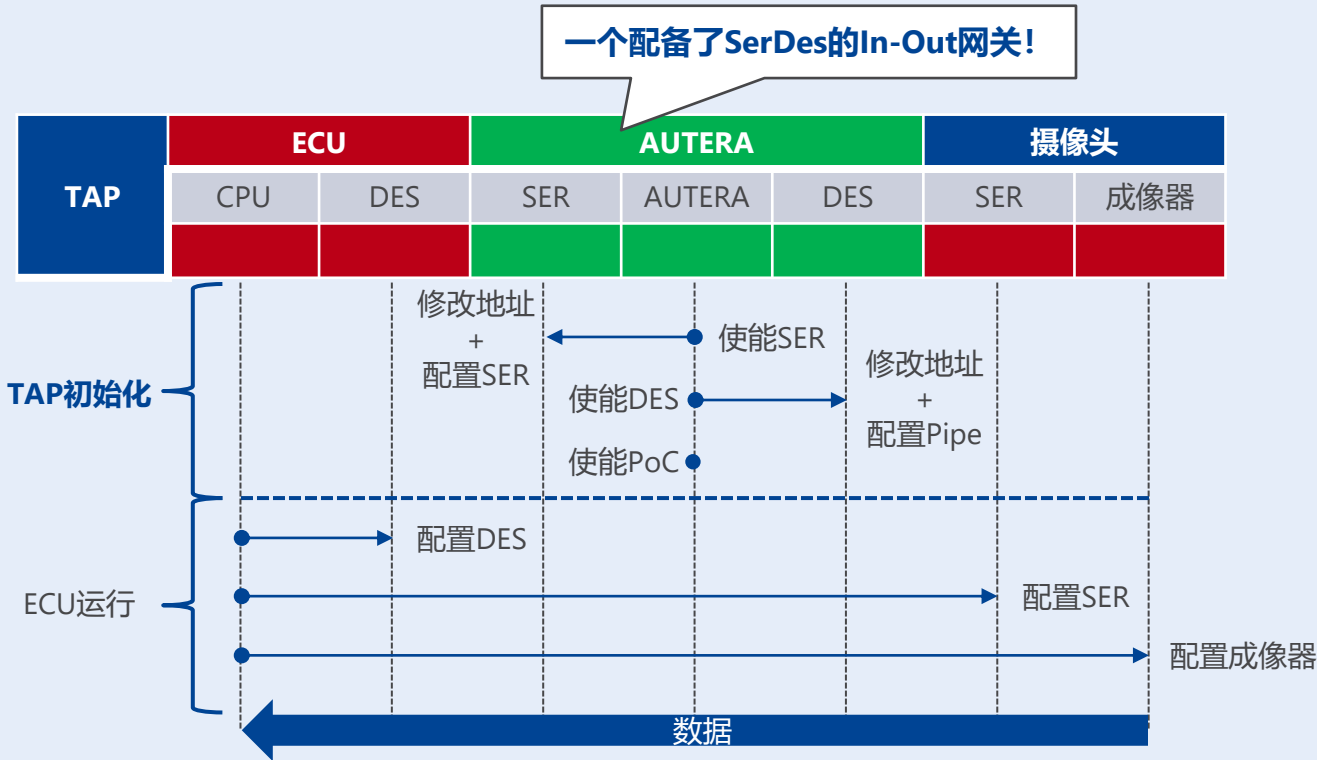
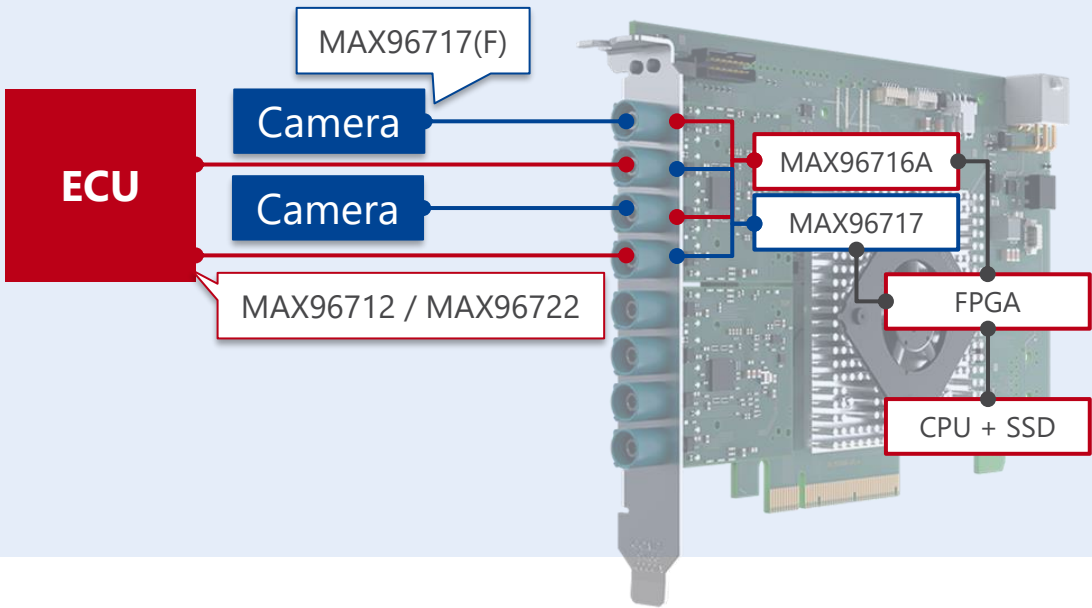
- $ts_sync = T0 - (ts_t0 - ts0 + Countdown)$



摄像头Tap采集

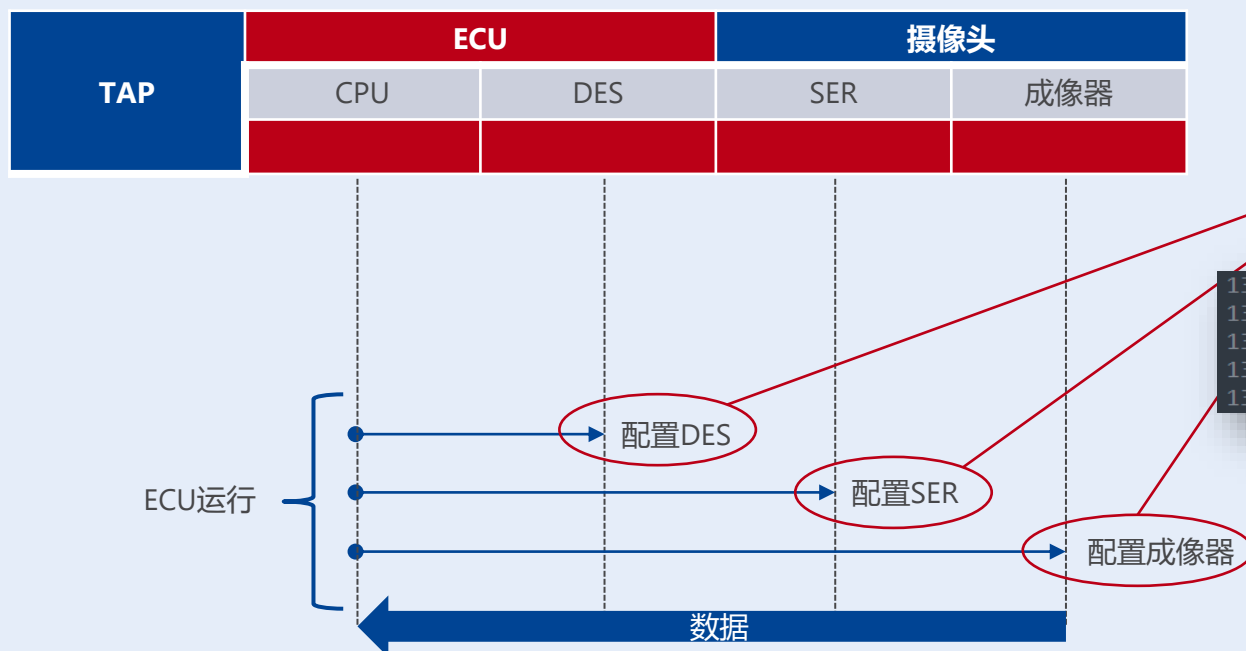
AUTERA Grabber —— 视频链路“网关”

- MIPI CSI2像素数据转发
- I2C转发
- 无数据篡改!!!
 - 33ms内对百万级数据量进行顺序读写操作，有那么容易???



ESI摄像头接口模拟 [1]

摄像头启动过程原理



数据回放应用中，ESI相对更关切ECU的“读”操作

- ECU读取了哪些SER与成像器的寄存器？尤其是成像器侧
- ECU底软对这些寄存器的“校验”逻辑如何？即，期望值？
- ESI需要满足“ECU”的期待

```
135 DevBlkCDIICPgmWriteUint8: Device: 0x29, Address: 0x0018, Data: 0x02
136 DevBlkCDIICPgmReadUint16: Device: 0x44, Address: 0x300A, Data: 0x58, 0x08, 0x41
137 DevBlkCDIICPgmReadUint16: Device: 0x44, Address: 0x302A, Data: 0x02, 0x00
138 OX08B40: Revision 2 detected!
139 OX08B40: Driver Version OX08B_607.001!
```

摘录自Bosch提供的I2C Log:

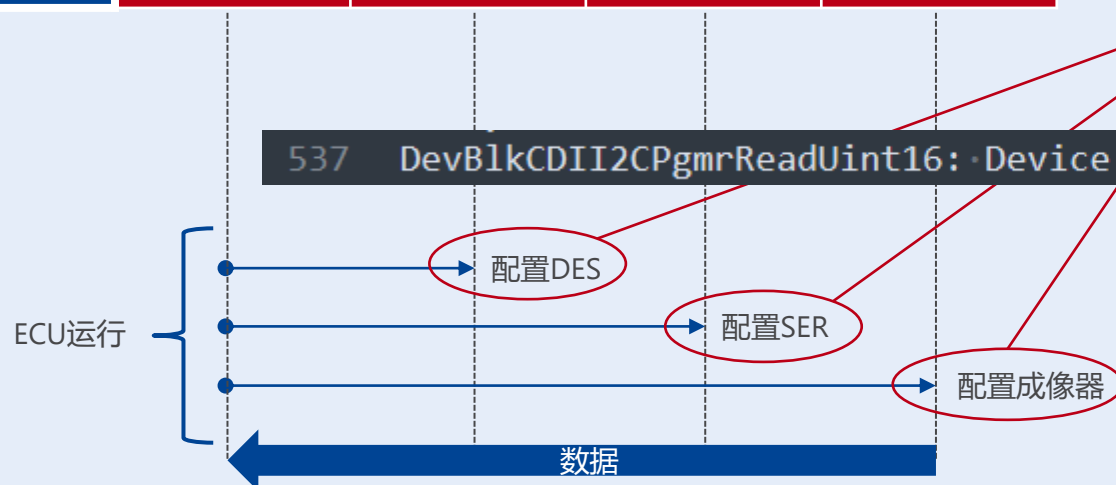
ECU通过检查OX08B40的**0x300A ... 0x300C**以及**0x302A ... 0x302B**判断所接摄像头的**成像器型号**，如138 - 139行ECU底软的反馈输出与期望（**0x58、0x08、0x41**以及**0x02、0x00**）吻合，因此得以进行下一步配置工作。

此步骤发生于对成像器配置伊始，是完成摄像头配置的必要前置条件。

还有很多类似的读以及判定操作，**详询底软开发人员，或基于I2C分析。**

ESI摄像头接口模拟 [2]

摄像头启动过程原理



一些“写”操作也同样重要

- ECU与Camera的SerDes连接要求如何？
—— SerDes接口物理联通的必要信息
- ECU要求Camera的分辨率以及时序参数如何？
—— ESI侧MIPI CSI2控制器的时序配置以及输入数据预处理
- ECU何时开始接收数据？
—— ESI侧何时可以打开数据链路进行输出？
-

537 DevBlkCDII2CPgmrReadUint16: Device: 0x44, Address: 0x3808, Data: 0x0F, 0x00, 0x08, 0x70

摘录自Bosch提供的I2C Log:

ECU通过配置OX08B40的**0x3808 ... 0x380B**设置摄像头的实际输出画面尺寸。

此类设置需与ESI实际模拟输出结果相符合。

摄像头适配工程服务 [1]

I2C Log、 I2C Log、 I2C Log!

- 如售前技术方案中已经明确，以及前述原理解释的，匹配服务需要了解ECU如何完成对摄像头的配置过程
 - 同一款摄像头，根据配置方式不同，可以表现出不同的行为以及输出
- 对于GMSL2：
 - Link Mode为3Gbps? 6Gbps?
 - Tunneling Mode是否启用?
 - 使用哪些GPIO?
 - TX? RX?
 - ID配置?
 - 电气特性配置?
 - 是否用作FSYNC?

特殊需求技术响应 [1]

需用户反馈

并联产品级控制器进行原始数据记录 [3]

- 定制服务需要用户配合!

重要性	配合事项	备注
必须	<ul style="list-style-type: none">已经可以正常工作的产品级控制器摄像头原件用于判断、调试控制器正常接收摄像头数据的必要诊断工具	<ul style="list-style-type: none">诊断工具如：开放额外视频输出接口以供查看控制器接收到的摄像头画面
需要	<ul style="list-style-type: none">控制器中的解串器引脚可被dSPACE直接接触测量或串行器以及成像器的I²C初始化参考序列	<ul style="list-style-type: none">可能意味着控制器可以被打开以暴露PCB引脚需协调底层开发同事或摄像头厂家提供相关代码片段提供参考序列存在一定程度的准确性风险
可选	<ul style="list-style-type: none">若控制器输出同步帧触发信号以控制所有摄像头同步拍摄，则需明确使用的串行器以及解串器各自的GPIO引脚	

- 上述准备工作需在项目启动时齐备并交予dSPACE

摄像头适配工程服务 [2]

I2C Log、 I2C Log、 I2C Log!

- 各节点的I2C设备地址？
 - 反复问及的：Tap模式下，ECU为何需要下电进行重新初始化？以及ECU停流后，为何需要重新初始化Tap？—— 均由于设备地址变化所引发
- 对于成像器
 - 分辨率、时序参数等
 - 特定寄存器的期望值
 - EEPROM
 -

特殊需求技术响应 [1]

需用户反馈

并联产品级控制器进行原始数据记录 [3]

- 定制服务需要用户配合！

重要性	配合事项	备注
必须	<ul style="list-style-type: none">已经可以正常工作的产品级控制器摄像头原件用于判断、调试控制器正常接收摄像头数据的必要诊断工具	<ul style="list-style-type: none">诊断工具如：开放额外视频输出接口以供查看控制器接收到的摄像头画面
需要	<ul style="list-style-type: none">控制器中的解串器引脚可被dSPACE直接接触测量或串行器以及成像器的I²C初始化参考序列	<ul style="list-style-type: none">可能意味着控制器可以被打开以暴露PCB引脚需协调底层开发同事或摄像头厂家提供相关代码片段提供参考序列存在一定程度的准确性风险
可选	<ul style="list-style-type: none">若控制器输出同步帧触发信号以控制所有摄像头同步拍摄，则需明确使用的串行器以及解串器各自的GPIO引脚	

- 上述准备工作需在项目启动时齐备并交予dSPACE

摄像头适配工程服务 [3]

一般工程服务过程

1. 获取固件以及刷新手册并对ECU进行更新
2. 直接尝试ECU更新后的版本是否能够与旧的ESI固件直接运行
3. 若不成功
 1. 使用ESI获取新的I2C log, 针对此项目, 同时也获取新的ECU版本和摄像头的log
 2. 联系相关底软开发, 获悉相关报错告警的逻辑背景与相关机制
4. 根据上述反馈进行分析
 1. 找出与旧版本相比不同点
 2. 查看ESI的log, 找到ESI和ECU的配置在哪一个阶段停止
5. 开始进行应用层适配工作
 1. 尝试借鉴摄像头的I2C log去修改ESI软件层, 使ESI和ECU的配置得以完成
 2. 使用客户所提供给的command通过ECU端查看失败原因
 3. 重复以上两步骤, 查看问题是否在应用层已解决
6. 若需要, 则开始底层适配工作
 1. 增加GPIO特定的功能, 如FSYNC或trigger等
 2. 修改传输速率, 如分辨率, 帧率等引起的改变
 3. 修改校验算法

复用不同EMB数据结构的视频数据进行回放 [1]

EMB数据结构差异

- 一般情况下，每个成像器芯片只有一种EMB数据排布结构，可见于芯片Datasheet
- 对于当前项目中使用的OV系芯片，可根据ECU侧的配置，改变EMB输出数据结构的排布方式
 - 同样见于其它用户项目
 - 2种排布方式：
 - NV605: EMB标志字节 + 寄存器值
 - NV607: EMB标志字节 + 寄存器地址 + 寄存器值
 - 对应不同的排布方式，EMB部分的CRC校验不同：
 - NV605: 4MCRC + 4CRC
 - NV607: 4CRC
- ECU侧的底层驱动按照事先明确的排布对EMB数据进行解构使用，同时亦对EMB数据内容进行校验检查
 - 不符合期待的EMB数据结构势必引发相关故障诊断错误，诸如：此前经历的Bad input stream
看上去是在提示数据流错误，实际是由于EMB中部分数据字节缺失所致（由于先前的ECU底软配置失误引发）

复用不同EMB数据结构的视频数据进行回放 [2]

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text	NV605+OVT602
00000120	0D	DA	00	06	0D	DA	09	99	0D	DA	00	05	0D	DA	00	01	.Û...Û.Û.Û.Û.	
00000130	0D	DA	02	2C	0D	DA	00	09	0D	DA	00	04	0D	DA	00	03	.Û...Û.Û.Û.Û.	
00000140	0D	DA	02	20	0D	DA	05	5D	0D	DA	00	00	0D	DA	0F	FF	.Û...Û.Û.Û.Û.	
00000150	0D	DA	0F	FF	0D	DA	00	00	0D	DA	00	00	0D	DA	02	23	.Û.Û.Û.Û.Û.Û.Û.	
00000160	0D	DA	08	8E	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	.Û.Û.Û.Û.Û.Û.Û.	
00000170	0D	DA	05	50	0D	DA	00	00	0D	DA	00	00	0D	DA	00	06	.Û.P.Û.Û.Û.Û.	
00000180	0D	DA	00	06	0D	DA	00	07	0D	DA	00	08	0D	DA	00	09	.Û...Û.Û.Û.Û.	
00000190	0D	DA	00	0A	0D	DA	00	0B	0D	DA	00	0C	0D	DA	00	0D	.Û...Û.Û.Û.Û.	
000001A0	0D	DA	00	0E	0D	DA	00	0F	0D	DA	01	10	0D	DA	01	11	.Û...Û.Û.Û.Û.	
000001B0	0D	DA	01	12	0D	DA	01	13	0D	DA	01	14	0D	DA	01	15	.Û...Û.Û.Û.Û.	
000001C0	0D	DA	01	16	0D	DA	01	17	0D	DA	00	00	0D	DA	00	00	.Û...Û.Û.Û.Û.	
000001D0	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	.Û...Û.Û.Û.Û.	
000001E0	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	.Û...Û.Û.Û.Û.	
000001F0	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	.Û...Û.Û.Û.Û.	
00000200	0D	DA	00	00	0D	DA	00	00	0D	DA	03	3F	0D	DA	00	00	.Û...Û.Û.Û.Û.	
00000210	0D	DA	00	00	0D	DA	02	2E	0D	DA	00	00	0D	DA	00	00	.Û...Û.Û.Û.Û.	
00000220	0D	DA	04	4E	0D	DA	00	00	0D	DA	00	00	0D	DA	08	86	.Û.N.Û.Û.Û.Û.	
00000230	0D	DA	00	00	0D	DA	00	00	0D	DA	0E	E6	0D	DA	00	00	.Û...Û.Û.Û.Û.	
00000240	0D	DA	00	01	0D	DA	08	8A	0D	DA	00	00	0D	DA	00	02	.Û...Û.Û.Û.Û.	
00000250	0D	DA	0A	A4	0D	DA	00	00	0D	DA	00	00	0D	DA	01	19	.Û.Û.Û.Û.Û.Û.	
00000260	0D	DA	00	00	0D	DA	00	00	0D	DA	02	2A	0D	DA	00	00	.Û...Û.Û.Û.Û.	
00000270	0D	DA	00	00	0D	DA	04	48	0D	DA	00	00	0D	DA	00	00	.Û...Û.H.Û.Û.	
00000280	0D	DA	07	7D	0D	DA	00	00	0D	DA	00	00	0D	DA	0D	D5	.Û.Û.Û.Û.Û.Û.	
00000290	0D	DA	00	00	0D	DA	00	01	0D	DA	06	6D	0D	DA	00	00	.Û...Û.Û.Û.Û.	
000002A0	0D	DA	00	00	0D	DA	03	30	0D	DA	00	00	0D	DA	00	00	.Û...Û.Û.Û.Û.	
000002B0	0D	DA	05	52	0D	DA	00	00	0D	DA	00	00	0D	DA	08	8C	.Û.R.Û.Û.Û.Û.	
000002C0	0D	DA	00	00	0D	DA	00	00	0D	DA	0F	F1	0D	DA	00	00	.Û...Û.Û.Û.Û.	
000002D0	0D	DA	00	01	0D	DA	09	9D	0D	DA	00	00	0D	DA	00	02	.Û...Û.Û.Û.Û.	
000002E0	0D	DA	0C	C4	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	.Û.Û.Û.Û.Û.Û.	
000002F0	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	.Û...Û.Û.Û.Û.	
00000300	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	.Û...Û.Û.Û.Û.	
00000310	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	.Û...Û.Û.Û.Û.	
00000320	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	.Û...Û.Û.Û.Û.	
00000330	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	.Û...Û.Û.Û.Û.	
00000340	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	.Û...Û.Û.Û.Û.	
00000350	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	.Û...Û.Û.Û.Û.	
00000360	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	.Û...Û.Û.Û.Û.	
00000370	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	.Û...Û.Û.Û.Û.	
00000380	0D	DA	00	00	0D	DA	00	00	0D	DA	00	00	0D	DA	05	5C	.Û...Û.Û.Û.Û.	
00000390	0D	DA	07	7D	0D	DA	06	6C	0D	DA	01	19	0D	DA	01	14	.Û.Û.Û.Û.Û.Û.	
000003A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000003B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000003C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000003D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000003E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000003F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000400	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text	NV607+OVT606
000000C0	0D	DA	03	35	01	18	00	01	0D	DA	03	35	01	19	00	00	.Û.S....Û.S....	
000000D0	0D	DA	03	35	01	1A	00	00	0D	DA	03	35	09	9E	00	02	.Û.S....Û.S.Û.	
000000E0	0D	DA	03	35	09	9F	00	00	0D	DA	03	35	09	98	00	01	.Û.S.Y...Û.S.~.	
000000F0	0D	DA	03	35	09	99	00	00	0D	DA	03	35	09	9A	00	00	.Û.S.Û...Û.S.Û.	
00000100	0D	DA	03	35	0D	DE	00	02	0D	DA	03	35	0D	DF	00	00	.Û.S.P...Û.S.Û.	
00000110	0D	DA	03	35	0D	D8	00	01	0D	DA	03	35	0D	D9	00	00	.Û.S.Û...Û.S.Û.	
00000120	0D	DA	03	35	0D	DA	00	00	0D	DA	03	35	05	5E	00	08	.Û.S.Û...Û.S.Û.	
00000130	0D	DA	03	35	05	5F	00	08	0D	DA	03	35	05	58	00	01	.Û.S...Û.S.X.	
00000140	0D	DA	03	35	05	59	00	00	0D	DA	03	35	05	5A	00	00	.Û.S.Y...Û.S.Z.	
00000150	0D	DA	05	50	0C	C0	00	0B	0D	DA	05	50	0C	C1	0E	EC	.Û.P.Û...Û.P.Û.	
00000160	0D	DA	05	50	0C	C2	00	04	0D	DA	05	50	0C	C3	00	00	.Û.P.Û...Û.P.Û.	
00000170	0D	DA	05	50	0C	C4	00	04	0D	DA	05	50	0C	C5	00	00	.Û.P.Û...Û.P.Û.	
00000180	0D	DA	05	50	0C	C6	00	07	0D	DA	05	50	0C	C7	00	08	.Û.P.Û...Û.P.Û.	
00000190	0D	DA	05	50	0C	C8	00	0B	0D	DA	05	50	0C	C9	0E	EC	.Û.P.Û...Û.P.Û.	
000001A0	0D	DA	05	50	0C	CA	00	04	0D	DA	05	50	0C	CB	00	00	.Û.P.Û...Û.P.Û.	
000001B0	0D	DA	05	50	0C	CC	00	04	0D	DA	05	50	0C	CD	00	00	.Û.P.Û...Û.P.Û.	
000001C0	0D	DA	05	50	0C	CE	00	07	0D	DA	05	50	0C	CF	00	08	.Û.P.Û...Û.P.Û.	
000001D0	0D	DA	05	50	0D	D8	00	0B	0D	DA	05	50	0D	D9	0E	EC	.Û.P.Û...Û.P.Û.	
000001E0	0D	DA	05	50	0D	DA	00	04	0D	DA	05	50	0D	DB	00	00	.Û.P.Û...Û.P.Û.	
000001F0	0D	DA	05	50	0D	DC	00	04	0D	DA	05	50	0D	DD	00	00	.Û.P.Û...Û.P.Û.	
00000200	0D	DA	05	50	0D	DE	00	07	0D	DA	05	50	0D	DF	00	08	.Û.P.Û...Û.P.Û.	
00000210	0D	DA	05	50	0D	00	00	0D	DA	05	50	0D	D1	0B	BE	00	.Û.P.Û...Û.P.Û.	
00000220	0D	DA	05	50	0D	D2	00	04	0D	DA	05	50	0D	D3	00	00	.Û.P.Û...Û.P.Û.	
00000230	0D	DA	05	50	0D	D4	00	04	0D	DA	05	50	0D	D5	00	00	.Û.P.Û...Û.P.Û.	
00000240	0D	DA	05	50	0D	D6	00	07	0D	DA	05	50	0D	D7	00	00	.Û.P.Û...Û.P.Û.	
00000250	0D	DA	04	4D	02	2A	03	36	0D	DA	04	4D	02	2B	0B	B6	.Û.M.*.6.Û.M.+.	
00000260	0D	DA	04	4F	00	06	00	00	0D	DA	04	4F	00	07	0C	00	.Û.O....Û.O....	
00000270	0D	DA	04	4F	00	08	01	14	0D	DA	04	4F	00	09	00	00	.Û.O....Û.O....	
00000280	0D	DA	04	4F	00	0A	0C	C0	0D	DA	04	4F	00	0B	01	14	.Û.O....Û.O....	
00000290	0D	DA	04	4F	00	0C	00	01	0D	DA	04	4F	00	0D	03	36	.Û.O....Û.O....	
000002A0	0D	DA	04	4F	00	0E	0B	B6	0D	DA	04	4F	00	0F	06	60	.Û.O....Û.O....	
000002B0	0D	DA	05	50	01	19	00	00	0D	DA	05	50	01	1A	0F	FF	.Û.P....Û.P....	
000002C0	0D	DA	05	50	01	1B	0F	FF	0D	DA	05	50	01	1D	00	00	.Û.P....Û.P....	
000002D0	0D	DA	05	50	01	1E	02	25	0D	DA	05	50	01	1F	02	24	.Û.P....Û.P....	
000002E0	0D	DA	05	50	02	25	00	00	0D	DA	05	50	02	26	02	25	.Û.P....Û.P....	
000002F0	0D	DA	05	50	02	27	02	24	0D	DA	05	50	02	21	00	00	.Û.P....Û.P....	
00000300	0D	DA	05	50	02	22	00	00	0D	DA	05	50	02	23	06	67	.Û.P....Û.P....	
00000310	0D	DA	03	30	00													

复用不同EMB数据结构的视频数据进行回放 [3]

重构EMB数据结构

- 实现构想（未经验证，仅为粗略思路）
 - 1) 正常读取原始数据的完整内容
 - 2) 分离EMB行
 - 3) 解构EMB各寄存器对象以及值
 - 4) 重新填充解构后的EMB数据对象至新布局
 - 5) 填充/删减缺失/多余的EMB数据对象
 - 6) 重新计算CRC
- 除步骤1) 以外，其余步骤均面向特定芯片具有高度定制性，不具有普适性，因此属于不可复用的开发工程。
- 相比闭环仿真，1) ... 3)是额外多余的步骤，且本身读取、分离、解析属于相对耗时的操作，难以在实时系统上轻易实现，也从未有过成功实践
- 属于篡改数据的范畴，不是常规数据回放系统解决方案设计的应用场景（尊重原始数据，高度复现），因此软硬件组成均没有相关应对预留方案

Important Information!

© 2023, dSPACE GmbH

All rights reserved. Written permission is required for reproduction of all or parts of this publication.

The source must be stated in any such reproduction.

This publication and the contents hereof are subject to change without notice.

Benchmark results are based on a specific application. Results are generally not transferable to other applications.

Brand names or product names are trademarks or registered trademarks of their respective companies or organizations.

dSPACE