

# 动脑学院

做一家受人尊敬的企业，做一位受人尊敬的老师

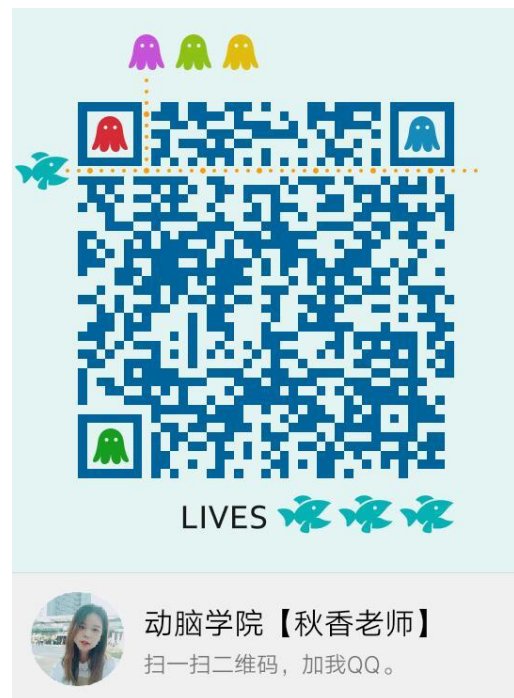


本课讲师：Darren老师

# 芯片原厂工程师 教你音视频同步分析

课程主讲Darren老师：2167195649

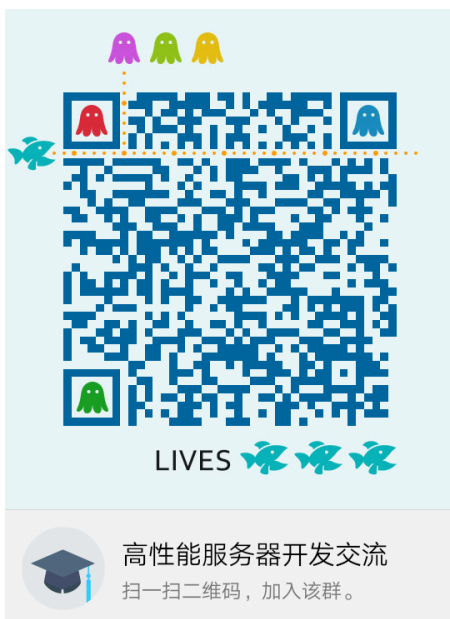
往期视频 秋香老师：1915869449



往期视频可以扫码加我们秋香美女老师

C/C++ linux服务器开发 / 后台服务架构【动脑学院】  
公开课地址：<https://ke.qq.com/course/131973>

更多课程通知可以扫码加群





## 讲师介绍

**动脑学院 -Darren老师 (廖庆富)**

**高级资深工程师**

- ◆ 音视频开发，嵌入式系统开发，图像处理以及服务器开发等方面有丰富的研发与实战经验。
- ◆ 曾供职于珠海扬智，深圳联发科等知名半导体公司。
- ◆ QQ: 2167195649





# 课程提纲

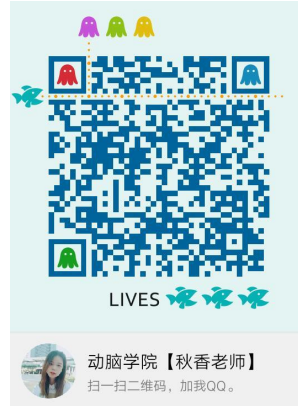
音视频录制和播放原理

同步概念讲解

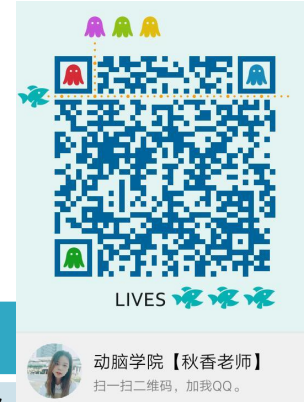
音视频采集时间戳记录方法

音视频播放同步要考虑的问题

ffplay的同步机制分析



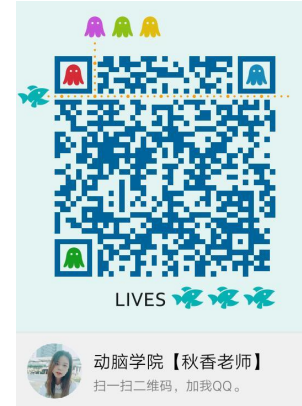
# 开源项目



项目	网址	备注
<b>ffmpeg</b>	<a href="http://ffmpeg.org">http://ffmpeg.org</a>	Windows/Linux/IOS/ Android
<b>ijkplayer</b> 手机端	<a href="https://github.com/Bilibili/ijkplayer">https://github.com/Bilibili/ijkplayer</a>	IOS/Android
<u>QMPlay2</u> PC端	<a href="https://github.com/zaps166/QMPlay2">https://github.com/zaps166/QMPlay2</a>	Windows/Linux
<u>ZLMediaKit</u> 跨平台的	<a href="https://github.com/xiongziliang/ZLMediaKit">https://github.com/xiongziliang/ZLMediaKit</a>	Windows/Linux
<u>EasyDarwin</u>	<a href="https://github.com/LinChengChun/EasyDarwin">https://github.com/LinChengChun/EasyDarwin</a>	Windows/Linux
<b>SRS</b>	<a href="https://github.com/ossrs/srs">https://github.com/ossrs/srs</a>	Linux
nginx-rtmp-module	<a href="https://github.com/arut/nginx-rtmp-module">https://github.com/arut/nginx-rtmp-module</a>	Linux



# 音视频中的时间戳



**PTS (Presentation Time Stamp)**：即显示时间戳，这个时间戳用来告诉**播放器**该在什么时候显示这一帧的数据。

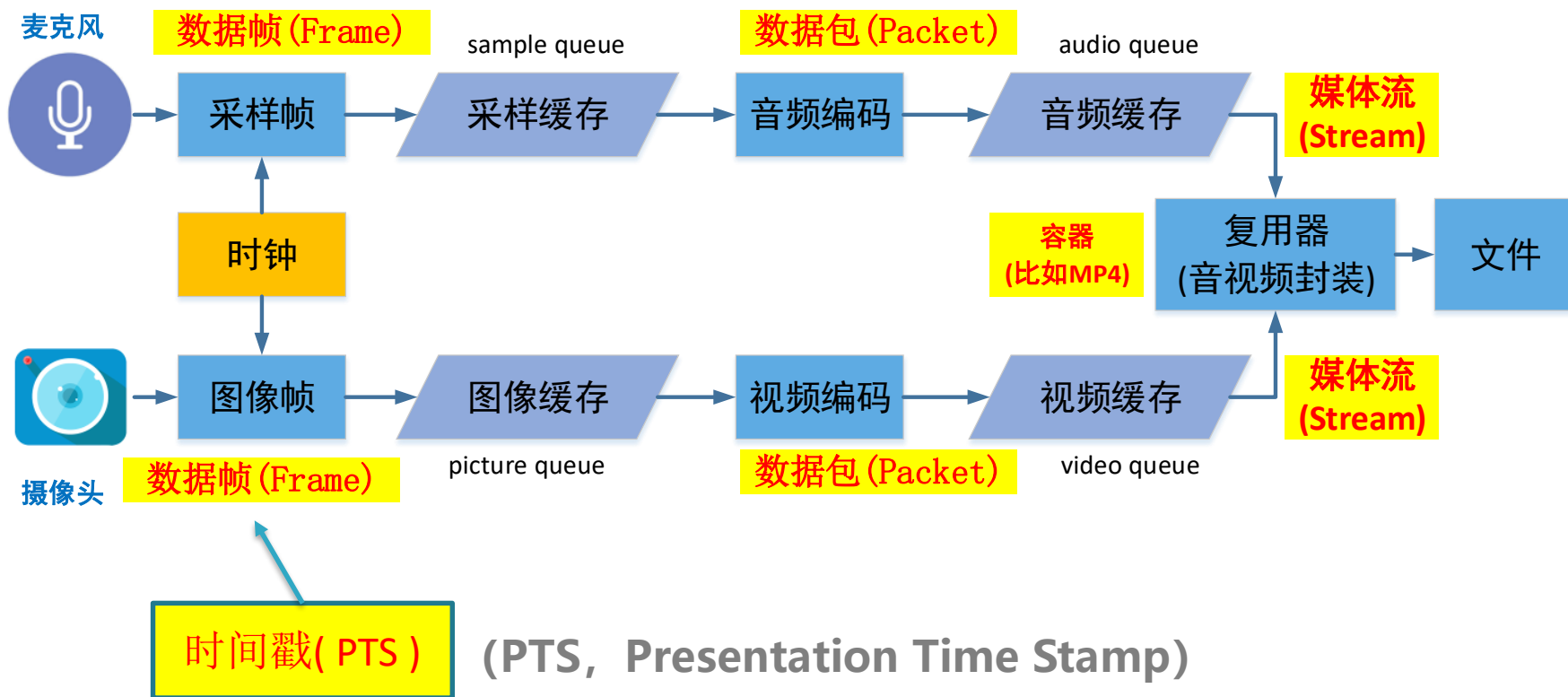


问题是时间戳 (PTS) 是从哪里来的呢？



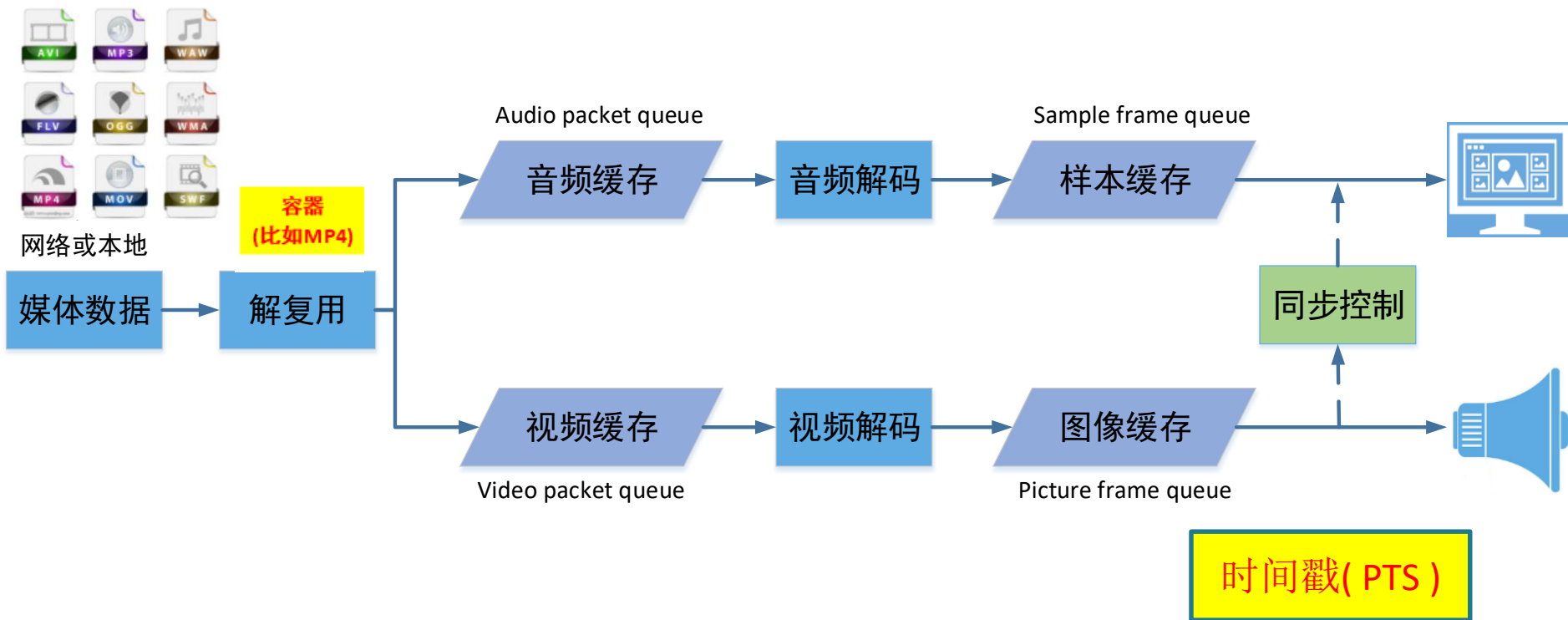
# 音视频录制框架

在音视频同步之前我们先以微信拍摄小视频讲解时间戳是怎么产生的。





# 音视频播放框架







# 什么是音视频同步?

- **音视频同步的实质**

- 画面和声音是同步的，比如看到**嘴唇在动**，那应该就会**听到声音**；看到开枪，那就应该听到枪声。

- **音视频同步标准**

- 音视频同步的标准其实是一个非常主观的东西
- 比如：【-100~+100ms】
  - **喇叭[-60, +30]ms**
  - **蓝牙音箱**[-160, +60]ms(基本没法做到**[-60, +30]**，**蓝牙协议本身延迟所致**)
  - 功放设备[-100, +40]ms
- **负值代表音频落后于视频**，正值代表**音频领先于视频**。



- **音视频同步认证**

- 杜比音效认证
- 美国**Netflix**认证

**NETFLIX**



# 什么是音视频同步？

- **如何测试音视频同步情况**

- ① 当然你可以直接播放一段相声，然后目测**声音和嘴型**是否对的上。
- ② 通过播放特定的测试片源（图像是黑白交替，声音间隔滴一声），并检测声音和屏幕亮度的变化，评判声音是落后于视频，还是领先于视频。**用示波器测试**，或者用Sync-One 设备。

# 音视频同步原理

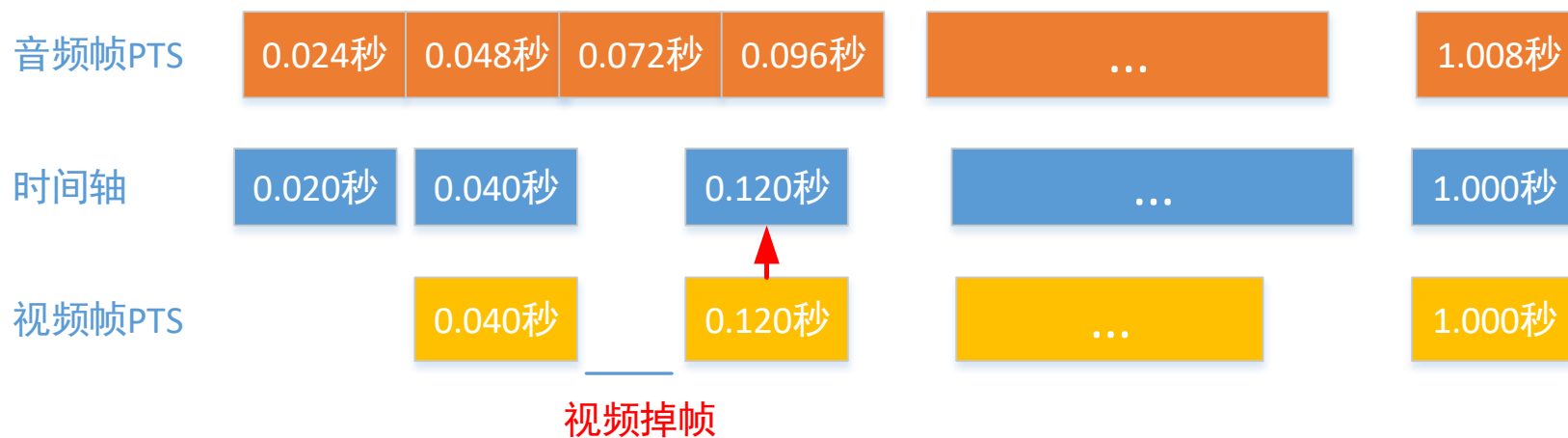
## 音视频同步有三种方式

- 以视频为基准，同步音频到视频
  - 音频慢了则加快播放速度 或 丢掉部分音频帧
  - 音频快了则放慢播放速度
- **以音频为基准**，同步视频到音频
  - 视频慢了则加快播放或**丢掉部分视频帧**
  - 视频快了则延迟播放，继续渲染上一帧
- **以外部时钟为基准**，同步音频和视频到外部时钟
  - 前两者的综合，根据外部时钟改变播放速度





## 音视频同步原理-视频为基准

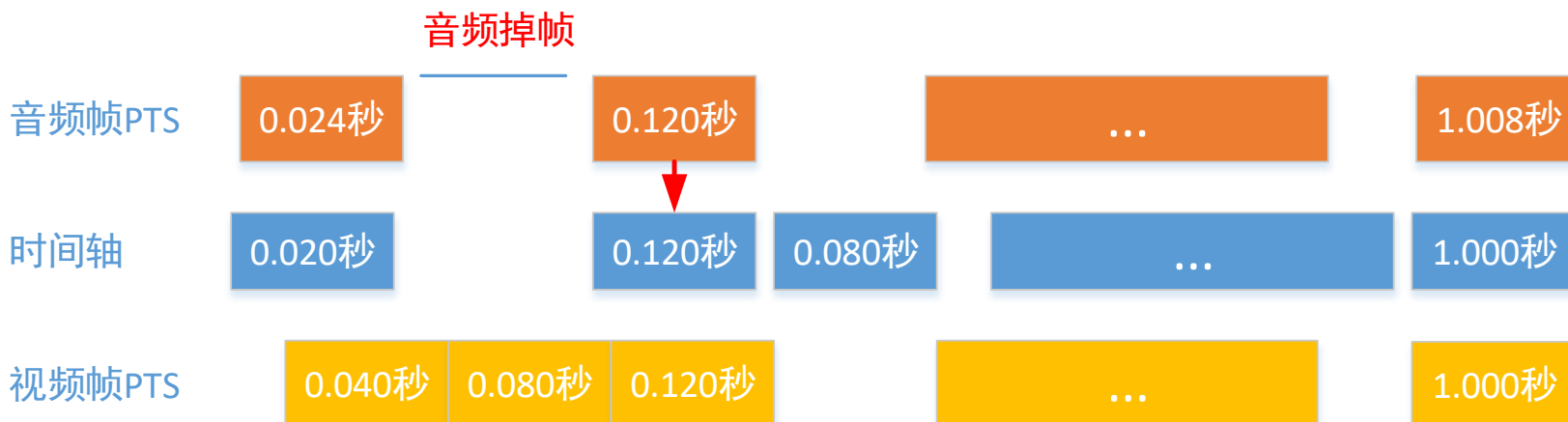


以**视频为同步基准**时，如果视频出现了掉帧，此时音频应该怎么做？可以参考的方式：

- 音频也相应丢帧（听觉->突然有断音）
- 音频加快播放速度（听觉->突然声音加快播放）



## 音视频同步原理-音频为基准



以**音频为同步基准**时，如果音频出现了掉帧，此时视频应该怎么做？可以参考的方式：

- 视频也相应丢帧（视觉->画面跳帧）
- 视频加快播放速度（视觉->画面快速播放）

## 音视频同步原理-外部时钟为基准

以**外部时钟为同步基准**时，如果音视频出现了丢帧，此时音视频应该怎么做？可以考虑以下方式：

- 当丢帧较多时，**重新初始化外部时钟**（pts和时钟进行比对，超过一定的阈值重设外部时钟，比如超过2秒）

TS流，码流本身带一个外部时间戳

同一个时间点：

比如**PCR = 300ms**

这个时候audio pts = 100ms

video pts = 200ms

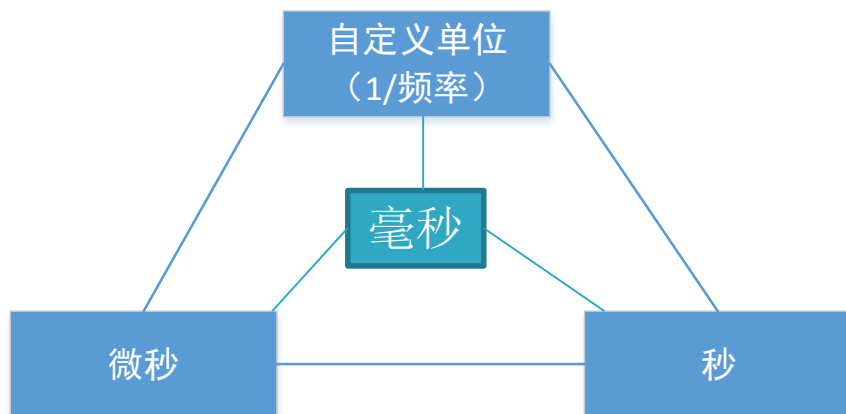
$PCR = PCR - 300ms$

另外**audio pts 和 video pts 都+300ms**

# 音视频时间换算的问题

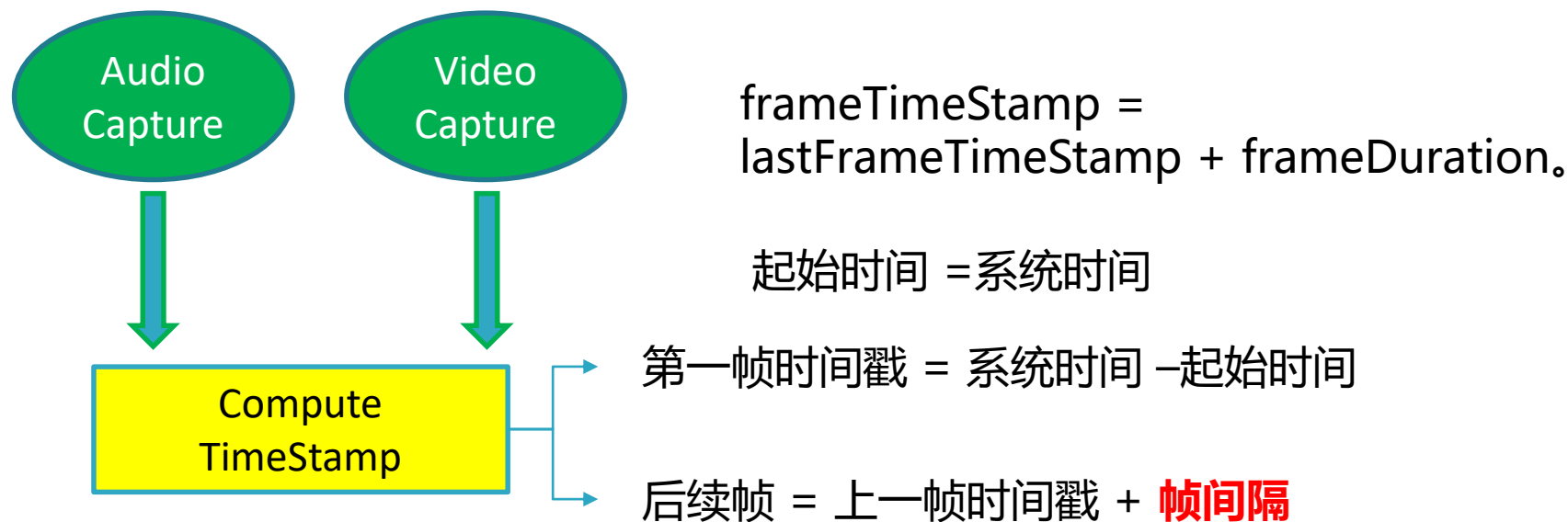
- 标准时间 秒 (seconds)
- 标准时间 毫秒 (millisecond)
- 标准时间 微秒 (microsecond)
- 自定义时间单位 (a/b 秒)
  - 以音频AAC音频帧举例, 如果pts以1/采样率为单位, 比如44.1khz, 则时间单位是1/44100, 因此PTS表示:  
第一帧  $PTS_1 = 0$   
第二帧  $PTS_2 = 1024$   
第三帧  $PTS_3 = 2048$   
播放的时候要将PTS换算成秒的单位, 则  $PTS_1 = 0 * 1/44100$ ,  
 $PTS_2 = 1024 * 1/44100$ ,  $PTS_3 = 2048 * 1/44100 = 0.046439$

PTS 的单位1/采样率  
PTS以采样点数累积为记录



自定义单位的pts不是统一的,  
我们在做音视频同步或计算时间  
长度的时候一定要注意其单位。  
避免在对比的时候使用了不同的  
时间单位

## 音视频采集时间戳记录-帧间隔模式



**优点：**能输出frame duration稳定的音视频时间戳。

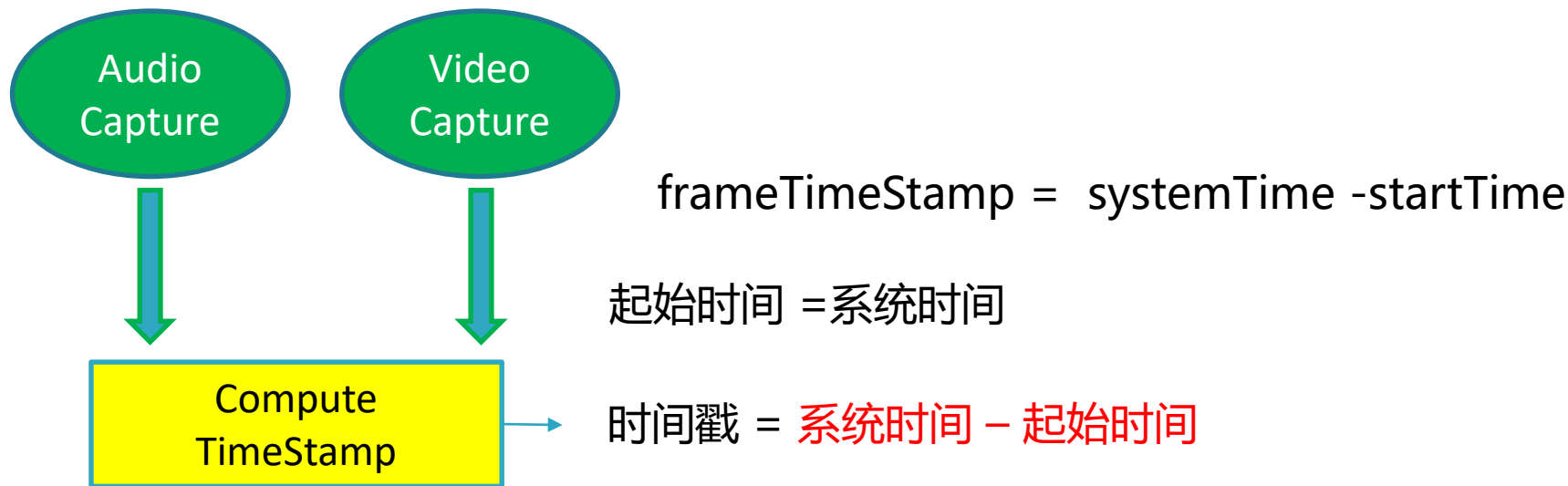
**风险：(1)**当系统负载比较重时有可能初学采集间隔不稳定的情况，比如预计1秒采集25帧图像，但实际采集了20帧，而音视频的时间戳是通过帧间隔累计的，这样计算出来的时钟就不正确了。

**(2)**帧间隔涉及到无限小数时，比如预计30帧，通常按帧间隔33毫秒处理，但实际是33.3333333毫秒。累积3333帧（约111秒）就出现1秒的误差。





## 音视频采集时间戳记录-直接系统时间模式

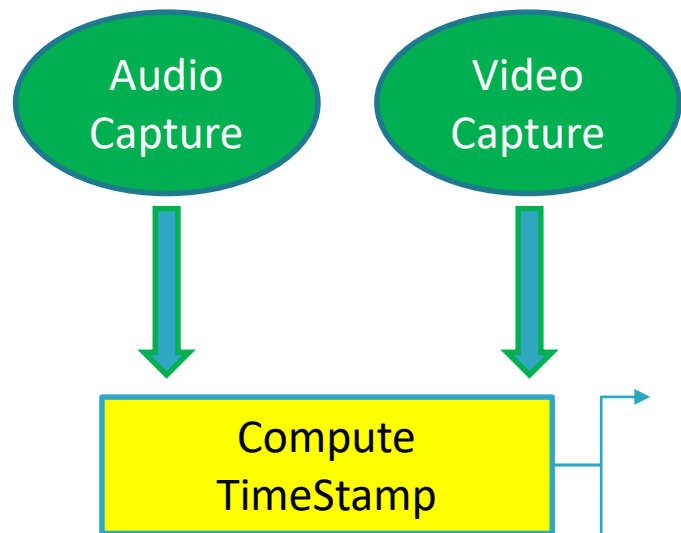


**优点：**能够实时纠正时间戳，只要系统正常运转，就能立即恢复正确的时间戳。

**风险：**帧间隔不均匀，能否正常播放依赖于终端。

比如，假如音频一帧间隔为24毫秒，被采集的回调时间可能为20毫秒，28毫秒，27毫秒，21毫秒。

## 音视频采集时间戳记录-帧间隔+直接系统时间模式



$\text{frameTimeStamp} = \text{lastFrameTimeStamp} + \text{frameDuration}.$

起始时间 = 系统时间

第一帧时间戳 = 系统时间 - 起始时间

后续帧 = 上一帧时间戳 + 帧间隔

$T = \text{当前系统时间} - \text{起始时间}$

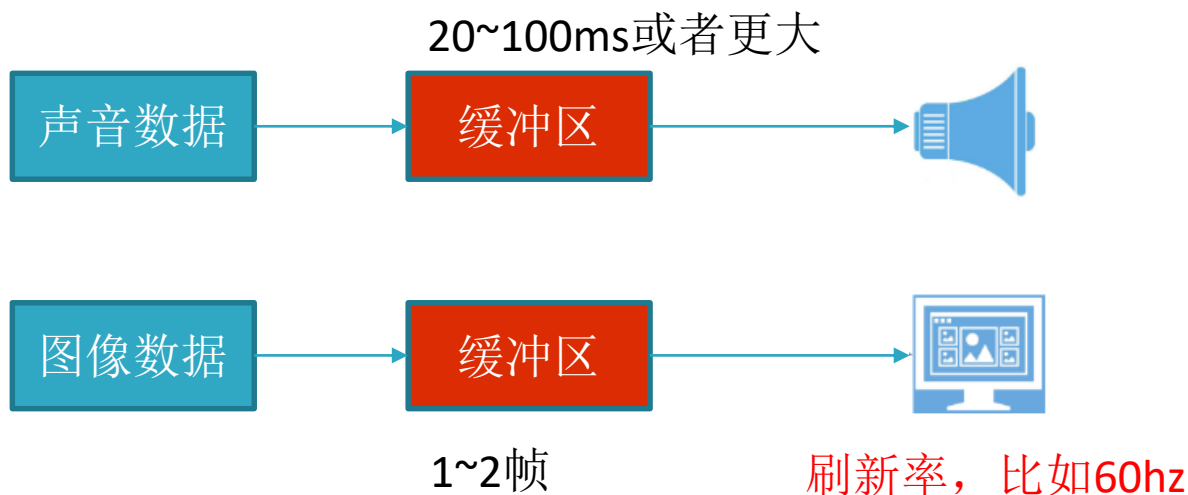
If  $(|\text{后续帧} - T| \geq \text{帧间隔}/2)$  后续帧 =  $T$

**优点：** 大部分音视频帧的frame duration都稳定。

**风险：** 纠正时间戳时可能会造成画面或声音卡顿的感觉。

# 音视频播放考虑的时间同步问题

1. 音视频同步模式的选择
  1. 音视频同时存在，优先选择以音频为基准
  2. 只有音频，优先选择以音频为基准
  3. 只有视频的时候（不要按帧率，可以按相邻帧间隔）
2. 声音输出延迟（和帧缓冲区有关）
3. 单独只有音频的情况
4. 单独只有视频的情况
5. 显示输出延迟（和帧缓冲区以及显示器刷新帧率有关）





# 音视频同步分析-专业打log

假定sys为系统时间简写

**发送端：** 音视频帧:帧序号-音视频时间戳:pts相邻帧间隔:pts\_duration;  
相邻帧系统时间间隔:sys\_duration

比如：

[send]audio:1-pts:20ms-ptsd:24ms; sysd=23ms

- (1) ptsd波动大，采集的帧间隔就不够稳定
- (2) 如果pts稳定，但是sysd波动大，则说明在发送数据的时候速率不稳定。
- (3) sysd和ptsd的值应该较为一致。

**接收端：** 音视频帧:帧序号-音视频时间戳:pts相邻帧间隔:pts\_duration;  
相邻帧系统时间间隔:sys\_duration

比如：

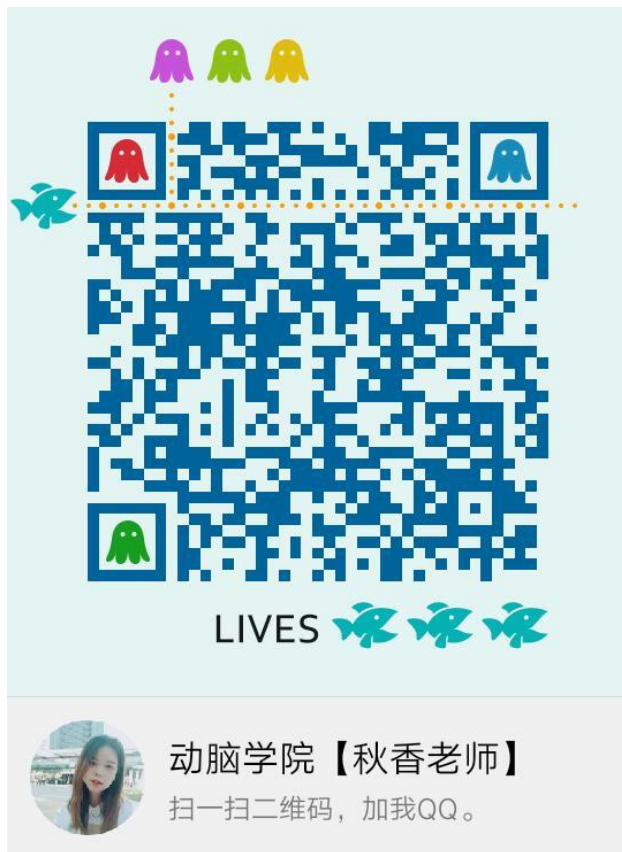
[recv]audio:1-pts:20ms-ptsd:24ms; sysd=23ms

- (1) ptsd波动大，采集的帧间隔就不够稳定
- (2) 如果pts稳定，但是sysd波动大，则说明在发送数据的时候速率不稳定。



# ffmpeg音视频同步分析

- 可以**扫码**加秋香美女老师要之前讲的课程：《07-FFMPEG之音视频同步原理分析》





# 站在前人的肩膀上



不管你年纪多大，你的时间总是宝贵的，IT行业从来没有真正的底层做起！  
培训的目的是为了在更少的时间内学习更多有用的知识！



# 非常感谢您的观看!

客服咨询 安然老师: 1463 124 473

