

# Tina3.0 项目

TPlayer 接口说明文档

# 文档履历

版本号	日期	制/修订人	制/修订记录
V1.0	2018/04/10		初始版本

Yllwinberte

# 目 录

1. 引言.....	5
1.1. 编写目的.....	5
1.2. 适用范围.....	5
1.3. 相关人员.....	5
2. TPlayer 状态图及状态说明.....	6
2.1. TPlayer 状态图.....	6
2.2. TPlayer 每个状态简要说明.....	7
2.2.1. Idle 状态.....	7
2.2.2. Initialized 状态.....	7
2.2.3. Preparing 状态.....	7
2.2.4. Prepared 状态.....	7
2.2.5. Started 状态.....	7
2.2.6. Paused 状态.....	7
2.2.7. Stopped 状态.....	7
2.2.8. PlaybackCompleted 状态.....	7
2.2.9. Error 状态.....	7
2.2.10. End 状态.....	7
3. 接口函数说明.....	8
3.1. TPlayerCreate.....	8
3.2. TPlayerDestroy.....	8
3.3. TPlayerSetDebugFlag.....	8
3.4. TPlayerSetNotifyCallback.....	8
3.5. TPlayerSetDataSource.....	8
3.6. TPlayerPrepare.....	9
3.7. TPlayerPrepareAsync.....	9
3.8. TPlayerStart.....	9
3.9. TPlayerPause.....	9
3.10. TPlayerStop.....	9
3.11. TPlayerReset.....	9
3.12. TPlayerSeekTo.....	9
3.13. TPlayerIsPlaying.....	10
3.14. TPlayerGetCurrentPosition.....	10
3.15. TPlayerGetDuration.....	10
3.16. TPlayerGetMediaInfo.....	10
3.17. TPlayerSetLooping.....	10
3.18. TPlayerSetScaleDownRatio.....	10
3.19. TPlayerSetRotate.....	11
3.20. TPlayerSetSpeed.....	11
3.21. TPlayerSetVolume.....	11
3.22. TPlayerGetVolume.....	11
3.23. TPlayerSetAudioMute.....	11
3.24. TPlayerSetExternalSubUrl.....	11
3.25. TPlayerGetSubDelay.....	12
3.26. TPlayerSetSubDelay.....	12
3.27. TPlayerGetSubCharset.....	12
3.28. TPlayerSetSubCharset.....	12
3.29. TPlayerSwitchAudio.....	12
3.30. TPlayerSwitchSubtitle.....	12
3.31. TPlayerSetSubtitleDisplay.....	13
3.32. TPlayerSetVideoDisplay.....	13
3.33. TPlayerSetDisplayRect.....	13
3.34. TPlayerSetSrcRect.....	13
3.35. TPlayerSetBrightness.....	13
3.36. TPlayerSetContrast.....	13
3.37. TPlayerSetHue.....	14
3.38. TPlayerSetSaturation.....	14
3.39. TPlayerSetEnhanceDefault.....	14

4. 播放器开发流程简单介绍..... 15  
5. Declaration..... 16

Allwinner

# 1. 引言

## 1.1. 编写目的

此文档说明在 tina3.0 平台，如何使用 TPlayer 的接口来开发播放器应用程序，方便播放器开发人员快速正确地开发。

## 1.2. 适用范围

本文档目前只适用于 tina3.0 平台的的播放器开发。目前支持的芯片平台:r16/r40/r18/r6/f35

## 1.3. 相关人员

Tina3.0 平台，播放器开发人员。

Allwinner

## 2. TPlayer 状态图及状态说明

### 2.1. TPlayer 状态图

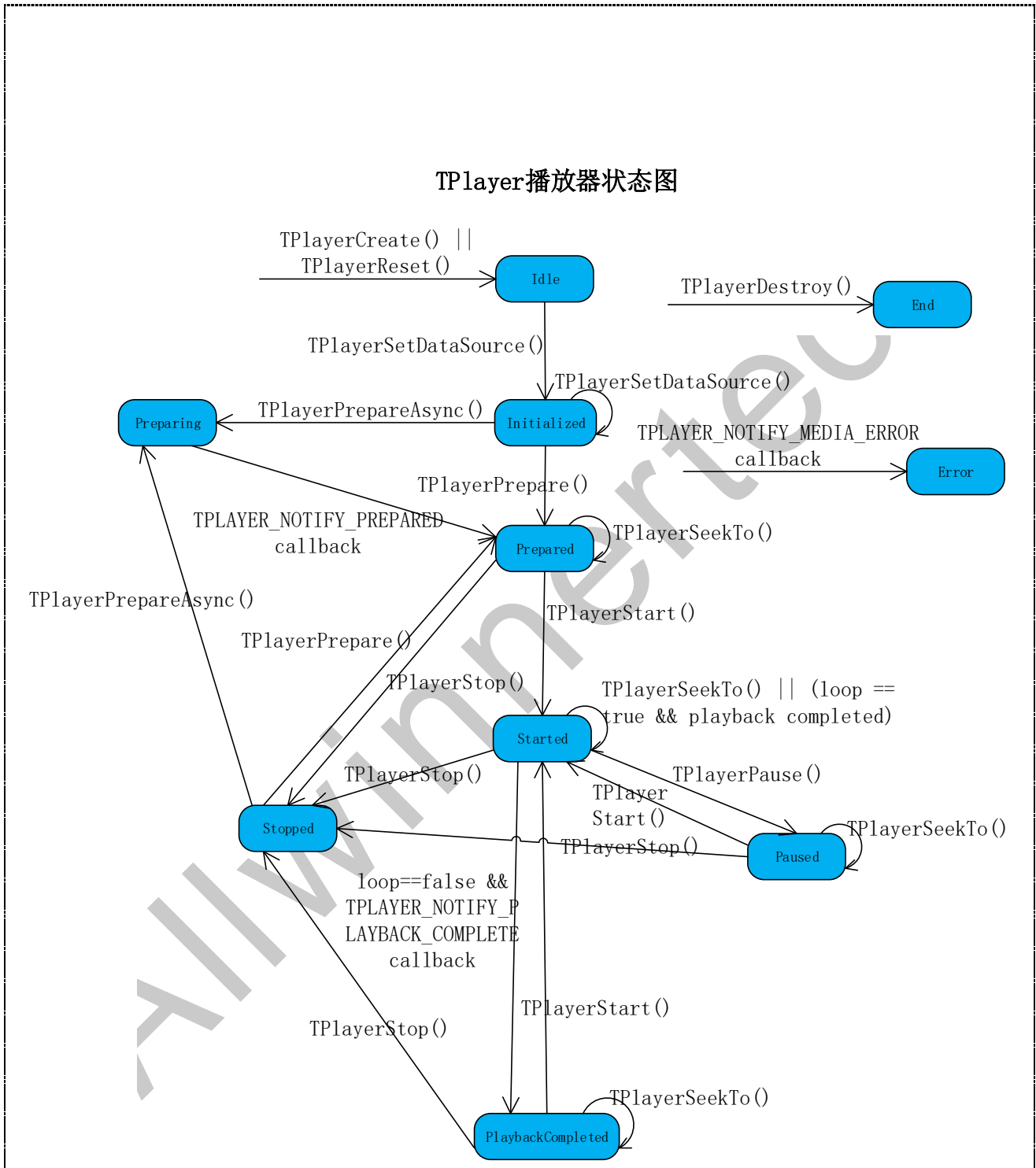


图 2-1 TPlayer 播放状态图

这张状态转换图清晰地描述了 TPlayer 的各个状态，也列举了主要的方法的调用时序，每种方法只能 在一些特定的状态下使用，否则会出错。

另外，只有在 Prepared、Started、Paused、PlaybackCompleted 这四种状态下可以进行 TPlayerSeekTo() 操作，并且 TPlayerSeekTo() 之后，状态不变。

## 2.2. TPlayer 每个状态简要说明

### 2.2.1. Idle 状态

Idle 状态：当调用 TPlayerCreate() 创建一个 TPlayer 或者调用了其 TPlayerReset() 方法时，TPlayer 处于 idle 状态。

### 2.2.2. Initialized 状态

这个状态比较简单，调用 TPlayerSetDataSource() 方法就进入 Initialized 状态，表示此时要播放的文件已经设置好了。

### 2.2.3. Preparing 状态

这个状态比较好理解，主要是和 TPlayerPrepareAsync() 配合，如果异步准备完成，会通过这个 TPLAYER\_NOTIFY\_PREPARED 消息回调上来，进而进入 Prepared 状态

### 2.2.4. Prepared 状态

初始化完成之后还需要通过调用 TPlayerPrepare() 或 TPlayerPrepareAsync() 方法，这两个方法一个是同步的一个是异步的，只有进入 Prepared 状态，才表明 TPlayer 到目前为止都没有错误，可以进行文件播放。

### 2.2.5. Started 状态

TPlayer 一旦准备好，就可以调用 TPlayerStart() 方法，这样 TPlayer 就处于 Started 状态，这表明 TPlayer 正在播放文件过程中。可以使用 TPlayerIsPlaying() 测试 TPlayer 是否处于 Started 状态。如果播放完毕，而又设置了循环播放，则 TPlayer 仍然会处于 Started 状态。

### 2.2.6. Paused 状态

Started 状态下可以调用 TPlayerPause() 方法暂停 TPlayer，从而进入 Paused 状态，TPlayer 暂停后再次调用 TPlayerStart() 则可以继续 TPlayer 的播放，转到 Started 状态。

### 2.2.7. Stopped 状态

Started 或者 Paused 状态下均可调用 TPlayerStop() 停止 TPlayer，而处于 Stop 状态的 TPlayer 要想重新播放，需要通过 TPlayerPrepareAsync() 和 TPlayerPrepare() 回到先前的 Prepared 状态重新开始才可以

### 2.2.8. PlaybackCompleted 状态

文件正常播放完毕，而又没有设置循环播放的话就进入该状态，并且会通过 TPLAYER\_NOTIFY\_PLAYBACK\_COMPLETE 这个消息回调给应用。此时可以调用 TPlayerStart() 方法重新从头播放文件，也可以 TPlayerStop() 停止 TPlayer，或者也可以 TPlayerSeekTo() 来重新定位播放位置。

### 2.2.9. Error 状态

由于某种原因 TPlayer 出现了错误，就会进入该状态，并且会通过 TPLAYER\_NOTIFY\_MEDIA\_ERROR 这个消息回调给应用。如果 TPlayer 进入了 Error 状态，可以通过调用 TPlayerReset() 来恢复，使得 TPlayer 重新返回到 Idle 状态。

### 2.2.10. End 状态

通过 TPlayerDestroy() 的方法可以进入 End 状态，只要 TPlayer 不再被使用，就应当尽快将其 destroy 掉。

## 3. 接口函数说明

### 3.1. TPlayerCreate

函数原型	TPlayer* TPlayerCreate(TplayerType type);
功能	创建一个 TPlayer
参数	type: 底层实际对接的播放器类型, 有 cedarx 和 gstreamer 这两种, 目前只支持 cedarx
返回值	成功返回 TPlayer 的指针, 失败返回 NULL
调用说明	无

### 3.2. TPlayerDestroy

函数原型	void TPlayerDestroy(TPlayer* p);
功能	销毁一个 TPlayer
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针
返回值	无
调用说明	无

### 3.3. TPlayerSetDebugFlag

函数原型	int TPlayerSetDebugFlag(TPlayer* p,bool debugFlag);
功能	设置是否打开调试信息
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 debugFlag:打开调试信息标志,true 为打开,false 为关闭
返回值	成功返回 0, 失败返回-1
调用说明	无

### 3.4. TPlayerSetNotifyCallback

函数原型	int TPlayerSetNotifyCallback(TPlayer* p, TPlayerNotifyCallback notifier, void* pUserData);
功能	设置 TPlayer 的消息回调函数
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 notifier: 回调消息处理函数指针, 需要由应用实现 pUserData: 回调消息处理对象
返回值	成功返回 0, 失败返回-1
调用说明	创建完 TPlayer 播放器之后, 就要调用该函数设置回调消息处理函数。

### 3.5. TPlayerSetDataSource

函数原型	int TPlayerSetDataSource(TPlayer* p, const char* pUrl,const CdxKeyedVectorT* pHeaders);
功能	设置播放文件的 url, 可以是本地文件也可以是网络源
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 pUrl:需要播放的文件的 url pHeaders:http 的一些头部信息
返回值	成功返回 0, 失败返回-1
调用说明	无



### 3.6. TPlayerPrepare

函数原型	int TPlayerPrepare(TPlayer* p);
功能	解析文件头部信息，获取元数据
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针
返回值	成功返回 0，失败返回-1
调用说明	该函数是阻塞函数，调用完返回之后就进入了 Prepared 状态，此时可调 TPlayerStart()函数进行播放

### 3.7. TPlayerPrepareAsync

函数原型	int TPlayerPrepareAsync(TinaPlayer* p);
功能	解析文件头部信息，获取元数据
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针
返回值	成功返回 0，失败返回-1
调用说明	该函数是非阻塞函数，需要等到 TPLAYER_NOTIFY_PREPARED 消息回调之后才能调 TPlayerStart()函数进行播放，而且 TPlayerStart()函数不能在回调函数中调用

### 3.8. TPlayerStart

函数原型	int TinaPlayerStart(TPlayer* p);
功能	开始播放
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针
返回值	成功返回 0，失败返回-1
调用说明	无

### 3.9. TPlayerPause

函数原型	int TPlayerPause(TPlayer* p);
功能	暂停播放
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针
返回值	成功返回 0，失败返回-1
调用说明	无

### 3.10. TPlayerStop

函数原型	int TPlayerStop(TPlayer* p);
功能	停止播放
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针
返回值	成功返回 0，失败返回-1
调用说明	无

### 3.11. TPlayerReset

函数原型	int TPlayerReset(TPlayer* p);
功能	重置播放器
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针
返回值	成功返回 0，失败返回-1
调用说明	在任何状态下都可以调用该函数，每次播放不同的音频之前，都需要调用该函数重置播放器，另外，一般收到 TPLAYER_NOTIFY_MEDIA_ERROR 这个消息的时候，也需要通过调用该函数来重置播放器。

### 3.12. TPlayerSeekTo

函数原型	int TPlayerSeekTo(TPlayer* p, int nSeekTimeMs);
------	---

功能	跳播
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 nSeekTimeMs: 跳播的位置, 单位是 ms
返回值	成功返回 0, 失败返回-1
调用说明	无

### 3.13. TPlayerIsPlaying

函数原型	bool TPlayerIsPlaying(TPlayer* p);
功能	是否正在播放
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针
返回值	返回 true 表示正在播放, 返回 false 表示没在播放
调用说明	无

### 3.14. TPlayerGetCurrentPosition

函数原型	int TPlayerGetCurrentPosition(TPlayer* p, int* msec);
功能	获取当前播放的位置
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 msec: 存放当前播放的位置值, 单位: ms
返回值	成功返回 0, 失败返回-1
调用说明	无

### 3.15. TPlayerGetDuration

函数原型	int TPlayerGetDuration(TPlayer* p, int* msec);
功能	获取播放的文件总时长
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 msec: 存储文件总时长, 单位: ms
返回值	成功返回 0, 失败返回-1
调用说明	需要在 prepared 状态之后才可以调用该函数

### 3.16. TPlayerGetMediaInfo

函数原型	int TPlayerGetMediaInfo(TPlayer* p, MediaInfo* mediaInfo);
功能	获取媒体信息
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 mediaInfo:存储媒体信息的指针
返回值	成功返回 0, 失败返回-1。如果失败, 则 mediaInfo 指针为 NULL
调用说明	需要在 prepared 状态之后才可以调用该函数

### 3.17. TPlayerSetLooping

函数原型	int TPlayerSetLooping(TPlayer* p, bool bLoop);
功能	设置循环播放模式
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 bLoop:true 表示单曲循环, false 表示不会单曲循环
返回值	成功返回 0, 失败返回-1
调用说明	无

### 3.18. TPlayerSetScaleDownRatio

函数原型	int TPlayerSetScaleDownRatio(TPlayer* p, TplayerVideoScaleDownType nHorizonScaleDown, TplayerVideoScaleDownType nVerticalScaleDown);
功能	设置视频的水平方向的缩放比例和垂直方向的缩放比例
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针

	nHorizonRatio:水平方向的缩放比例 nVerticalRatio:垂直方向的缩放比例
返回值	成功返回 0, 失败返回-1
调用说明	这个函数需要在 prepare 之前调用

### 3.19. TPlayerSetRotate

函数原型	int TPlayerSetRotate(TPlayer* p, TplayerVideoRotateType rotateDegree);
功能	设置视频旋转的角度
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 rotateDegree: 视频旋转的角度
返回值	成功返回 0, 失败返回-1
调用说明	这个函数需要在 prepare 之前调用, 注: 这个接口没实现

### 3.20. TPlayerSetSpeed

函数原型	int TPlayerSetSpeed(TPlayer* p, TplayerPlaySpeedType nSpeed);
功能	设置快进快退的速度
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 nSpeed:快进快退的速度
返回值	成功返回 0, 失败返回-1
调用说明	无

### 3.21. TPlayerSetVolume

函数原型	int TPlayerSetVolume(TPlayer* p,int volume);
功能	设置播放器的音量大小
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 volume: 需要设置的音量大小值
返回值	成功返回 0, 失败返回-1
调用说明	无

### 3.22. TPlayerGetVolume

函数原型	int TPlayerGetVolume(TPlayer* p,int* volume);
功能	获取播放器当前音量的大小
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 volume: 存放获取的音量的大小值
返回值	成功返回 0, 失败返回-1
调用说明	无

### 3.23. TPlayerSetAudioMute

函数原型	int TPlayerSetAudioMute(TPlayer* p,bool mute);
功能	静音或不静音
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 mute: 是否静音, true 为静音, false 为不静音
返回值	成功返回 0, 失败返回-1
调用说明	无

### 3.24. TPlayerSetExternalSubUrl

函数原型	int TPlayerSetExternalSubUrl(TPlayer* p, const char* filePath);
功能	设置外挂字幕的路径
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针

	filePath: 外挂字幕的路径
返回值	成功返回 0, 失败返回-1
调用说明	无

### 3.25. TPlayerGetSubDelay

函数原型	int TPlayerGetSubDelay(TPlayer* p);
功能	获取字幕提前或延时的时间
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针
返回值	成功返回字幕提前或延时的时间, 单位: ms, 失败返回-1
调用说明	无

### 3.26. TPlayerSetSubDelay

函数原型	int TPlayerSetSubDelay(TPlayer* p, int nTimeMs);
功能	设置字幕提前或延时的时间
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 nTimeMs: 字幕延时或提前的时间
返回值	成功返回 0, 失败返回-1
调用说明	无

### 3.27. TPlayerGetSubCharset

函数原型	int TPlayerGetSubCharset(TPlayer* p, char *charset);
功能	获取字幕的字符编码格式
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 charset: 获取字幕的编码格式
返回值	成功返回 0, 失败返回-1
调用说明	无

### 3.28. TPlayerSetSubCharset

函数原型	int TPlayerSetSubCharset(TPlayer* p, const char* strFormat);
功能	设置字幕的字符编码格式
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 strFormat: 字幕的编码格式
返回值	成功返回 0, 失败返回-1
调用说明	无

### 3.29. TPlayerSwitchAudio

函数原型	int TPlayerSwitchAudio(TPlayer* p, int nStreamIndex);
功能	切换音轨
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 nStreamIndex: 音轨的 index
返回值	成功返回 0, 失败返回-1
调用说明	无

### 3.30. TPlayerSwitchSubtitle

函数原型	int TPlayerSwitchSubtitle(TPlayer* p, int nStreamIndex);
功能	切换字幕
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 nStreamIndex: 字幕流的 index
返回值	成功返回 0, 失败返回-1

调用说明	无
------	---

### 3.31. TPlayerSetSubtitleDisplay

函数原型	void TPlayerSetSubtitleDisplay(TPlayer* p, bool onoff);
功能	设置打开或关闭字幕
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 onoff: 打开或关闭字幕的标志位
返回值	无
调用说明	无

### 3.32. TPlayerSetVideoDisplay

函数原型	void TPlayerSetVideoDisplay(TPlayer* p, bool onoff);
功能	设置是否显示视频
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 onoff: 是否显示视频的标志位
返回值	无
调用说明	无

### 3.33. TPlayerSetDisplayRect

函数原型	void TPlayerSetDisplayRect(TPlayer* p,int x, int y, unsigned int width, unsigned int height);
功能	设置显示的区域
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 x: 显示区域起始点 x 坐标的值 y: 显示区域起始点 y 坐标的值 width:显示区域的宽 height:显示区域的高
返回值	无
调用说明	无

### 3.34. TPlayerSetSrcRect

函数原型	void TPlayerSetSrcRect(TPlayer* p,int x, int y, unsigned int width, unsigned int height);
功能	设置原图像 crop 的坐标和大小
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 x: 源图像 crop 起始点 x 坐标的值 y: 源图像 crop 起始点 y 坐标的值 width:源图像 crop 的宽 height:源图像 crop 的高
返回值	无
调用说明	无

### 3.35. TPlayerSetBrightness

函数原型	void TPlayerSetBrightness(TPlayer* p,unsigned int grade);
功能	设置亮度
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 grade: 亮度的值, 范围是:0-100, 默认值是:50
返回值	无
调用说明	无

### 3.36. TPlayerSetContrast

函数原型	void TPlayerSetContrast(TPlayer* p,unsigned int grade);
------	---

功能	设置对比度
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 grade: 对比度的值, 范围是:0-100, 默认值是:50
返回值	无
调用说明	无,注: 这个接口没实现

### 3.37. TPlayerSetHue

函数原型	void TPlayerSetHue(TPlayer* p,unsigned int grade);
功能	设置色调
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 grade: 色调的值, 范围是:0-100, 默认值是:50
返回值	无
调用说明	无,注: 这个接口没实现

### 3.38. TPlayerSetSaturation

函数原型	void TPlayerSetSaturation(TPlayer* p,unsigned int grade);
功能	设置饱和度
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针 grade: 饱和度的值, 范围是:0-100, 默认值是:50
返回值	无
调用说明	无,注: 这个接口没实现

### 3.39. TPlayerSetEnhanceDefault

函数原型	void TPlayerSetEnhanceDefault(TPlayer* p);
功能	使能丽色系统, 并用默认设置的值
参数	p:通过 TPlayerCreate 函数创建的 TPlayer 指针
返回值	无
调用说明	无

## 4. 播放器开发流程简单介绍

- (1) TPlayerCreate() //创建一个播放器
- (2) TPlayerSetNotifyCallback() //设置消息回调函数
- (3) TPlayerSetDataSource() //设置 url
- (4) TPlayerPrepare()或 TPlayerPrepareAsync()  
//解析头部信息, 获取元数据, 并根据元数据的信息初始化对应的解码器
- (5) TPlayerStart() //播放  
(注:如果是用 TPlayerPrepareAsync()函数, 则需要等到 TPLAYER\_NOTIFY\_PREPARED 消息回调之后才可以调用 TPlayerStart()函数进行播放)
- (6) 如果需要跳播, 则可以调用 TPlayerSeekTo()函数
- (7) 如果需要暂停, 则调用 TPlayerPause()函数进行暂停
- (8) 如果需要停止, 则可以调用 TPlayerStop()或 TPlayerReset()函数进行停止(注: 建议用 TPlayerReset()函数进行停止, 因为任何状态下都可以调用 TPlayerReset()函数)
- (9) 如果需要播放下一个或其他的, 则可以先调用 TPlayerReset()函数使播放器进入 idle 状态, 然后再重复(3)(4)(5)的步骤
- (10) 详细的播放器开发 demo 可以参考以下这个路径的内容:  
package/allwinner/tina\_multimedia\_demo/tplayerdemo

## 5. Declaration

This document is the original work and copyrighted property of Allwinner Technology (“Allwinner”). Reproduction in whole or in part must obtain the written approval of Allwinner and give clear acknowledgment to the copyright owner.

The information furnished by Allwinner is believed to be accurate and reliable. Allwinner reserves the right to make changes in circuit design and/or specifications at any time without notice. Allwinner does not assume any responsibility and liability for its use. Nor for any infringements of patents or other rights of the third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Allwinner. This datasheet neither states nor implies warranty of any kind, including fitness for any particular application.

Allwinner