文档变更记录

版本	变更说明	日期	作者	审核
1.0	创建文档	2023年7月19日	唐威	

class TAFirmwareUpgradeManager

描述：升级管理类

method addListener

```
fun addListener(l: OTAFirmwareUpgradeProcessStateListener)
```

描述：增加升级状态上报监听。

输入参数：

l：监听对象，参考： OTAFirmwareUpgradeProcessStateListener

method addListener

```
fun removeListener(l: OTAFirmwareUpgradeProcessStateListener)
```

描述：移除升级状态上报监听。

输入参数：

l：监听对象，参考： OTAFirmwareUpgradeProcessStateListener

method startFirmwareUpgrade

```
suspend fun startFirmwareUpgrade(fileModels: Array<OTAUpgradeModel>)
```

描述：开始升级。

输入参数：

fileModels: 升级对象列表 参考： OTAUpgradeModel

注：必须捕获异常，fileModels至少要有一个，且天空端的升级对象放在前面

interface OTAFirmwareUpgradeProcessStateListener

描述：升级状态监听回调

method onOTAFirmwareUpgradeProcessState

```
fun onOTAFirmwareUpgradeProcessState(model: OTAUpgradeModel, status:
OTAFirmwareUpgradeProcessState)
```

描述：升级监听回调。

输入参数：

model: 升级对象 参考： OTAUpgradeModel

status: 升级状态 参考： OTAFirmwareUpgradeProcessState

enum class UpgradeClientTypeEnum

```
enum class UpgradeClientTypeEnum(var value: Int) {
    /**
     *客户端是GND
     */
    CLIENT_TYPE_GND(0),

    /**
     *客户端是SKY
     */
    CLIENT_TYPE_SKY(1);

}
```

描述：升级类型

enum class OTAFirmwareUpgradeProcessState

```
enum class OTAFirmwareUpgradeProcessState (val code: Int, des: String) {
    NONE(0,"未开始"),
    CONDITION_CHECK(1,"升级条件检查"),
    CONDITION_CHECK_COMPLETED(2,"升级条件检查完成"),
    MODE_CHANGE(3,"传输模式切换"),
    MODE_CHANGE_COMPLETED(4,"传输模式切换完成"),
    START_UPLOAD(5,"升级文件开始上传"),
    UPLOADING(6,"升级文件上传中"),
    UPLOAD_COMPLETED(7,"文件上传完成"),
    FILE_VERIFY(8,"开始文件校验"),
    FILE_VERIFY_COMPLETED(9,"文件校验完成"),
    START_UPGRADE(10,"开始升级"),
    UPGRADING(11,"升级中"),
    UPGRADE_COMPLETED(12,"升级完成"),
    OPERATION_FAILED(13, "升级失败");

    // 上传/升级进度
    var progress: Int? = null

    // 操作失败的错误属性
    var errorMessage: OTAUpgradeErrorMessage? = null
}
```

描述：升级状态枚举

属性： errorMessage：失败后的错误信息 参考: OTAUpgradeErrorMessage

class OTAUpgradeModel

描述： 升级对象

property clientType

描述： 升级类型

类型： UpgradeClientTypeEnum 参考: UpgradeClientTypeEnum

property file

描述： 升级文件

类型： File

class OTAUpgradeErrorMessage

描述： 升级错误信息

property errorCode

描述： 错误码

类型： Int

property message

描述： 错误信息

类型： String

property upgradeError

描述： 升级错误

类型： UpgradeErrorTypeEnum

注：当错误不能用upgradeError展示出的时候，会体现在errorCode中

enum class UpgradeErrorTypeEnum

```
enum class UpgradeErrorTypeEnum(var value: Int) {
    /**
     * 未知错误
     */
    CODE_UNKNOWN(-100),

    /**
     * 升级超时
     */
    UPGRADE_TIME_OUT  (-101),

    /**
     * 升级上传文件失败
     */
    UPGRADE_UPLOAD_ERROR  (-102),

    /**
     * 升级下载文件失败
     */
    UPGRADE_DOWNLAOD_ERROR  (-103),


    /**
     * 重试次数达到
     */
    UPGRADE_REPEAT_ERROR  (-104),

    /**
     * "解析失败"
     */
    PROTO_RESPONSE_PARSE_ERROR(-1),

    /**
     *   "返回数据为空"
     */
    PROTO_RESPONSE_EMPTY_DATA(-2),

    /**
     *   "响应超时"
     */
    PROTO_RESPONSE_TIMEOUT(-3),

    /**
     * "响应码错误"
     */
    RESPONSE_CODE_ERROR(-4),

    /**
     * "设备未连接"
     */
    DEVICE_DISCONNECTED(-5),

    /**
     * "重复请求"
     */
    REPEAT_REQUEST(-6),
    /**
     * no error
     */
    UPGRADE_ERR_NONE(0),

    /**
     * in flight
     */
    UPGRADE_ERR_FLY  (1),

    /**
     * in upgrading
     */
    UPGRADE_ERR_BUSY (2),

    /**
     * MD5 error
     */
    UPGRADE_ERR_MD5   (3),

    /**
     * file not exist
     */
    UPGRADE_ERR_EXIST (4),

    /**
     * low battery
     */
    UPGRADE_ERR_LOBAT (5),

    /**
     * model not match
     */
    UPGRADE_ERR_MODEL (6),

    /**
     * no enough space to upgrade
     */
    UPGRADE_ERR_SPACE (7),

    /**
     * error header of pkg, like crc error
     */
    UPGRADE_ERR_HEAD (8),

    /**
     * error signature, untrusted firmware
     */
    UPGRADE_ERR_SIGN (9),

    /**
     * error firmware size, not match header
     */
    UPGRADE_ERR_SIZE (10),

    /**
     * package error
     */
    UPGRADE_ERR_PACKAGE (11),

    /**
     * error major version, not match firmware
     */
    UPGRADE_ERR_VERSION (12),

    /**
     * lower than the current version, need force flag
     */
    UPGRADE_ERR_LVERSION (13),

    /**
     * refuse upgrade request, errDesc for more details
     */
    UPGRADE_ERR_REFUSE (14),

    /**
     * special error of every pkg, use errDesc for details
     */
    UPGRADE_ERR_SPECIAL (15);
}
```

描述： 升级错误枚举