# BELON\_BK7231\_AYLA\_LOCAL\_OTA协议简介

* 目的：

1：基于bk7231接入ayla云，做成产品后，考虑到后期如果有些产品出现程序问题。但是量产的产品有没有办法拆壳子，所以我们设计出可以通过局域网可以给产品升级协议和工具。

2：考虑到后期我们可能需要做本地一些服务控制，所以拟草出一套基于json格式的局域网控制。

* 协议简介：

1:可以通过Rtconfig.h中的宏：\_BELON\_LOCAL\_OTA\_

来开启和关闭本地ota功能。

2：协议基于TCP连接，我们bk7231开启了TCP Server 服务，端口号：1986，最多同时支持2个client设备连接。Server listen 最多同时支持5个。

3：协议中定义的宏。

#define SOCK\_SERVER\_PORT 1986

//tag len

#define LAN\_JSON\_TAG\_IP\_LEN 32

#define LAN\_JSON\_TAG\_DID\_LEN 32

#define LAN\_JSON\_TAG\_VER\_LEN 16

#define LAN\_JSON\_TAG\_DEVNAME\_LEN 64

#define LAN\_JSON\_TAG\_HOST\_LEN 256

#define LAN\_JSON\_TAG\_CDATA\_LEN 256

#define LAN\_JSON\_TAG\_HCONFIG\_LEN 16

#define LAN\_JSON\_TAG\_CONFIG\_LEN 16

#define LAN\_JSON\_CMD\_MAX\_LEN 256

#define LAN\_JSON\_STR\_START "BELON"

#define LAN\_JSON\_STR\_END "\r\n"

//json cmd

#define LAN\_JSON\_TAG\_HCMD "Hcmd"

#define LAN\_JSON\_TAG\_SEQ "SEQ"

#define LAN\_JSON\_TAG\_IP "IP"

#define LAN\_JSON\_TAG\_PORT "Port"

#define LAN\_JSON\_TAG\_DID "DID"

#define LAN\_JSON\_TAG\_RID "RID"

#define LAN\_JSON\_TAG\_HCONFIG "HConfig"

#define LAN\_JSON\_TAG\_KID "KID"

#define LAN\_JSON\_TAG\_DURATION "Duration"

#define LAN\_JSON\_TAG\_VER "VER"

#define LAN\_JSON\_TAG\_DEVNAME "DevName"

#define LAN\_JSON\_TAG\_DEVTYPE "DevType"

#define LAN\_JSON\_TAG\_HSTATE "HState"

#define LAN\_JSON\_TAG\_HCAUSE "HCause"

#define LAN\_JSON\_TAG\_HOST "Hostname"

#define LAN\_JSON\_TAG\_HSTATE "HState"

#define LAN\_JSON\_TAG\_CONFIG "Config"

#define LAN\_JSON\_TAG\_AUTHIND "AuthInd"

#define LAN\_JSON\_TAG\_RANGE "Range"

#define LAN\_JSON\_TAG\_CAUSE "Cause"

#define LAN\_JSON\_TAG\_CDATA "CData"

#define LAN\_JSON\_TAG\_RESULT "Err"

#define LAN\_JSON\_TAG\_UPDATEIND "UpdateInd"

#define LAN\_JSON\_TAG\_PROTOCOL "ProtocolVer"

#define LAN\_JSON\_TAG\_DEVVER "DevVer"

#define LAN\_JSON\_TAG\_DEVID "DevId"

#define LAN\_JSON\_TAG\_DEBUG "DebugInfo"

//used for HiLink 2.0

#define LAN\_JSON\_TAG\_RESP "RSP"

#define LAN\_JSON\_TAG\_PLVER "PLVer"

#define LAN\_JSON\_TAG\_DEVLIST "DevList"

#define LAN\_JSON\_TAG\_CONNPORT "ConnPort"

#define LAN\_JSON\_TAG\_BARCODE "Barcode"

#define LAN\_JSON\_TAG\_DSTATE "DState"

#define LAN\_JSON\_TAG\_PDVER "PDVer"

#define LAN\_JSON\_TAG\_HTYPE "HType"

#define LAN\_JSON\_TAG\_EDATA "ExtData"

#define LAN\_JSON\_TAG\_CINFO "CInfo"

#define LAN\_JSON\_TAG\_PDATA "DevPData"

#define LAN\_JSON\_PLVER\_STR "2.0"

typedef enum \_Jsoncmd\_err\_type{

JSON\_SUCCESS =0x00,

JSON\_AUTH\_FAIL,

JSON\_INVALID\_DATA,

JSON\_INVALID\_CMD,

JSON\_INVALID\_MD5,

JSON\_NO\_RESPONSE,

JSON\_ERR\_UNKNOWN,

}Jsoncmd\_err\_type;

typedef enum \_Jsoncmd\_enum\_type{

CMD\_BCWIFIMSG = 1,

CMD\_BCPHONEMSG = 2,

CMD\_RESULT = 3,

CMD\_CONFIG = 4,

CMD\_DEVICE = 5,

CMD\_DEVSTATUS = 6,

CMD\_DEVREQ = 7,

CMD\_DEVDATA = 8,

CMD\_RESET = 9,

CMD\_UPDATE = 10,

CMD\_SERVER = 11,

CMD\_DEVINFO = 12,

}Jsoncmd\_enum\_type;

* 举例：

如果需要本地发起ota升级指令：

发送数据格式：(数据结尾必须以回车换行结束 /r/n)

BELON{"Hcmd”:10,”UpdateInd”:”http://url”}

其中的url为需要ota文件的url地址。

