

ECx00U&EGx00U 系列

QuecOpen MMS API 参考手册

LTE Standard 模块系列

版本：1.0

日期：2021-08-31

状态：受控文件



上海移远通信技术股份有限公司（以下简称“移远通信”）始终以为客户提供最及时、最全面的服务为宗旨。如需任何帮助，请随时联系我司上海总部，联系方式如下：

上海移远通信技术股份有限公司
上海市闵行区田林路 1016 号科技绿洲 3 期（B 区）5 号楼 邮编：200233
电话：+86 21 5108 6236 邮箱：info@quectel.com

或联系我司当地办事处，详情请登录：<http://www.quectel.com/cn/support/sales.htm>。

如需技术支持或反馈我司技术文档中的问题，请随时登陆网址：
<http://www.quectel.com/cn/support/technical.htm> 或发送邮件至：support@quectel.com。

前言

移远通信提供该文档内容以支持客户的产品设计。客户须按照文档中提供的规范、参数来设计产品。同时，您理解并同意，移远通信提供的参考设计仅作为示例。您同意在设计您目标产品时使用您独立的分析、评估和判断。在使用本文档所指导的任何硬软件或服务之前，请仔细阅读本声明。您在此承认并同意，尽管移远通信采取了商业范围内的合理努力来提供尽可能好的体验，但本文档和其所涉及服务是在“可用”基础上提供给您。移远通信可在未事先通知的情况下，自行决定随时增加、修改或重述本文档。

使用和披露限制

许可协议

除非移远通信特别授权，否则我司所提供硬软件、材料和文档的接收方须对接收的内容保密，不得将其用于除本项目的实施与开展以外的任何其他目的。

版权声明

移远通信产品和本协议项下的第三方产品可能包含受移远通信或第三方材料、硬软件和文档版权保护的相关资料。除非事先得到书面同意，否则您不得获取、使用、向第三方披露我司所提供的文档和信息，或对此类受版权保护的资料进行复制、转载、抄袭、出版、展示、翻译、分发、合并、修改，或创造其衍生作品。移远通信或第三方对受版权保护的资料拥有专有权，不授予或转让任何专利、版权、商标或服务商标权的许可。为避免歧义，除了正常的非独家、免版税的产品使用许可，任何形式的购买都不可被视为授予许可。对于任何违反保密义务、未经授权使用或以其他非法形式恶意使用所述文档和信息的违法侵权行为，移远通信有权追究法律责任。

商标

除另行规定，本文档中的任何内容均不授予在广告、宣传或其他方面使用移远通信或第三方的任何商标、商号及名称，或其缩略语，或其仿冒品的权利。

第三方权利

您理解本文档可能涉及一个或多个属于第三方的硬软件和文档（“第三方材料”）。您对此类第三方材料的使用应受本文档的所有限制和义务约束。

移远通信针对第三方材料不做任何明示或暗示的保证或陈述，包括但不限于任何暗示或法定的适销性或特定用途的适用性、平静受益权、系统集成、信息准确性以及与许可技术或被许可人使用许可技术相关的不侵犯任何第三方知识产权的保证。本协议中的任何内容都不构成移远通信对任何移远通信产品或任何其他硬软件、设备、工具、信息或产品的开发、增强、修改、分销、营销、销售、提供销售或以其他方式维持生产的陈述或保证。此外，移远通信免除因交易过程、使用或贸易而产生的任何和所有保证。

免责声明

- 1) 移远通信不承担任何因未能遵守有关操作或设计规范而造成损害的责任。
- 2) 移远通信不承担因本文档中的任何因不准确、遗漏、或使用本文档中的信息而产生的任何责任。
- 3) 移远通信尽力确保开发中功能的完整性、准确性、及时性，但不排除上述功能错误或遗漏的可能。除非另有协议规定，否则移远通信对开发中功能的使用不做任何暗示或法定的保证。在适用法律允许的最大范围内，移远通信不对任何因使用开发中功能而遭受的损害承担责任，无论此类损害是否可以预见。
- 4) 移远通信对第三方网站及第三方资源的信息、内容、广告、商业报价、产品、服务和材料的可访问性、安全性、准确性、可用性、合法性和完整性不承担任何法律责任。

版权所有 ©上海移远通信技术股份有限公司 2021，保留一切权利。

Copyright © Quectel Wireless Solutions Co., Ltd. 2021.

文档历史

修订记录

版本	日期	作者	变更表述
-	2021-08-09	Herry GENG	文档创建
1.0	2020-08-31	Herry GENG	受控版本

目录

文档历史	3
目录	4
表格索引	5
1 引言	6
1.1. 适用模块	6
2 MMS 相关接口调用流程	7
3 MMS 相关 API	9
3.1. 头文件	9
3.2. 参考示例	9
3.3. 函数概览	9
3.4. API 详解	10
3.4.1. ql_mms_client_new	10
3.4.1.1. mms_result_code_e	10
3.4.1.2. mms_send_result_cb	11
3.4.2. ql_mms_client_send_msg	12
3.4.2.1. mms_option_t	12
3.4.2.2. mms_charset_type	13
3.4.2.3. mms_send_param_t	14
3.4.2.4. mms_valid_type	14
3.4.2.5. mms_priority_type	15
3.4.2.6. mms_visible_type	16
3.4.2.7. mms_class_type	16
3.4.2.8. mms_info_t	17
3.4.2.9. mms_subject_t	17
3.4.2.10. mms_addr_t	18
3.4.2.11. mms_attachment_t	18
3.4.3. ql_mms_client_release	19
4 附录 参考文档及术语缩写	20

表格索引

表 1: 适用模块 6

表 2: 函数概览 9

表 3: 参考文档 20

表 4: 术语缩写 20

1 引言

移远通信 LTE Standard ECx00U 系列和 EGx00U 系列模块支持 QuecOpen®方案；QuecOpen®是基于 RTOS 的嵌入式开发平台，可简化 IoT 应用的软件设计和开发过程。有关 QuecOpen®的详细信息，请参考文档 [1]。

本文档主要介绍在 QuecOpen®方案下，ECx00U 系列和 EGx00U 系列模块 MMS 相关接口调用流程和 API 函数详解。

1.1. 适用模块

表 1：适用模块

模块系列	模块
ECx00U	EC200U 系列
	EC600U 系列
EGx00U	EG500U-CN
	EG700U-CN

2 MMS 相关接口调用流程

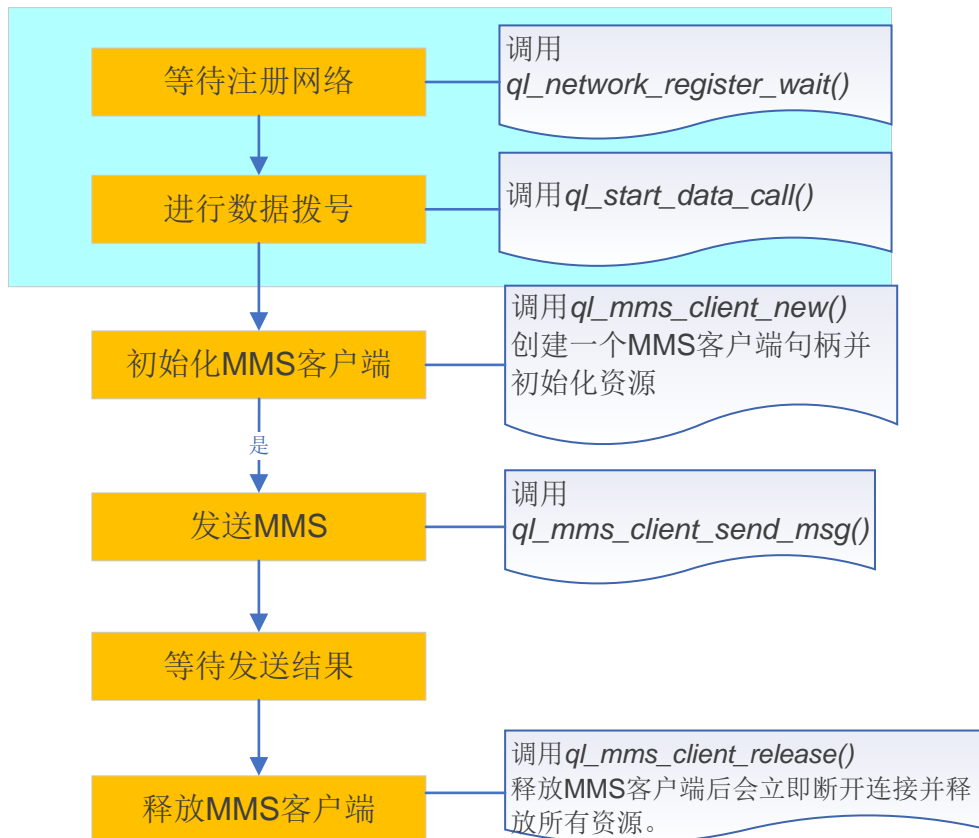


图 1: MMS 接口调用流程

MMS 服务基于指定的数据通道进行数据收发，因此在发送 MMS 之前，首先需要注网并拨号以建立数据通道（如上图中的蓝底部分）。若在其他任务中已经建立了数据通道，可跳过注网拨号步骤，直接调用 `ql_mms_client_new()` 创建 MMS 客户端句柄。数据拨号相关函数可参考文档 [2]。

注网拨号成功且已通过 `ql_mms_client_new()` 创建了一个 MMS 客户端句柄后，就可以调用 `ql_mms_client_send_msg()` 发送 MMS，该函数会立即返回，相关事件发生时调用相应的回调函数。

备注

1. ECx00U 系列和 EGx00U 系列 QuecOpen 模块仅支持 MMS 1.3。
2. 同一个 MMS 客户端同一时间只能执行一次发送彩信的请求，当前请求处理完成后，方可进行下一次请求。若需要同一时间执行多次发送彩信的请求，需要创建多个 MMS 客户端。

3 MMS 相关 API

3.1. 头文件

MMS 的头文件为 `ql_mms_client.h`，位于 `\components\ql-kernel\inc` 目录下。

3.2. 参考示例

MMS 客户端示例代码文件位于 SDK 包的 `\components\ql-application\mms\mms_demo.c` 路径下。

3.3. 函数概览

表 2：函数概览

函数	说明
<code>ql_mms_client_new()</code>	创建一个新的 MMS 客户端句柄并初始化 MMS 客户端资源
<code>ql_mms_client_send_msg()</code>	发送 MMS 消息
<code>ql_mms_client_release()</code>	释放 MMS 客户端句柄及其相关资源

3.4. API 详解

3.4.1. ql_mms_client_new

该函数用于创建一个新的 MMS 客户端句柄并初始化 MMS 客户端资源。

● 函数原型

```
int ql_mms_client_new(ql_mms_client *client_hdl, mms_send_result_cb result_cb, void *arg)
```

● 参数

client_hdl:

[Out] MMS 客户端句柄。

result_cb:

[In] 告知应用程序 MMS 发送结果的回调函数，详情请参考第 3.4.1.2 章。

arg:

[In] *result_cb* 的回调参数。

● 返回值

详情请参考第 3.4.1.1 章。

3.4.1.1. mms_result_code_e

函数调用后返回结果枚举定义如下：

```
typedef enum{
    QL_MMS_OK                      = 0,
    QL_MMS_INVLIAD_PARAM          = -1,
    QL_MMS_WOULDBLOCK             = -2,
    QL_MMS_OUT_OF_MEM             = -3,
    QL_MMS_HTTP_REQ_FAIL          = -4,
    QL_MMS_OPT_BUSY               = -5,
    QL_MMS_HTTP_RESP_FAIL         = -6,
    QL_MMS_SEND_FAIL              = -7,
    QL_MMS_HTTP_TIMEOUT           = -8,
    QL_MMS_DNS_FAIL               = -9,
}mms_result_code_e;
```

- 参数

参数	描述
<code>QL_MMS_OK</code>	函数执行成功
<code>QL_MMS_INVLIAD_PARAM</code>	无效参数
<code>QL_MMS_WOULDBLOCK</code>	操作被阻塞
<code>QL_MMS_OUT_OF_MEM</code>	内存不足
<code>QL_MMS_HTTP_REQ_FAIL</code>	发送 HTTP 请求失败
<code>QL_MMS_OPT_BUSY</code>	当前 MMS 客户端正在发送彩信
<code>QL_MMS_HTTP_RESP_FAIL</code>	HTTP 响应失败
<code>QL_MMS_SEND_FAIL</code>	彩信发送失败
<code>QL_MMS_HTTP_TIMEOUT</code>	HTTP 请求超时
<code>QL_MMS_DNS_FAIL</code>	域名解析失败

3.4.1.2. mms_send_result_cb

通过该函数告知应用程序 MMS 的发送结果。

- 函数原型

```
typedef void(*mms_send_result_cb)(ql_mms_client client_hdl, int mms_result, int http_code,
char *mms_err_msg, void *arg)
```

- 参数

client_hdl:

[In] MMS 客户端句柄。由 `ql_mms_client_new()` 获取。

mms_result:

[In] MMS 发送结果，详情请参考第 3.4.1.1 章。

http_code:

[In] HTTP 响应码，详情请参考文档 [3]。

mms_err_msg:

[In] 当 *mms_result* 为 `QL_MMS_SEND_FAIL` 时，这个字符串会给出具体失败的原因。

arg:

[In] 回调参数，由 *ql_mms_client_new()* 设置。

- 返回值

无

3.4.2. ql_mms_client_send_msg

该函数用于发送 MMS 消息。

- 函数原型

```
int ql_mms_client_send_msg(ql_mms_client *client_hdl, struct mms_option_t *send_option,
struct mms_info_t *mms_info)
```

- 参数

client_hdl:

[In] MMS 客户端句柄。由 *ql_mms_client_new()* 获取。

send_option:

[In] 发送配置，详情请参考第 3.4.2.1 章。

mms_info:

[In] MMS 消息的内容，包括主题、收件人、附件等，详情请参考第 3.4.2.8 章。。

- 返回值

详情请参考第 3.4.1.1 章。

3.4.2.1. mms_option_t

MMS 发送配置的定义如下：

```
struct mms_option_t{
    int                context_id;
    int                sim_id;
    char               *mmsc_url;
    char               *proxy_addr;
    uint16_t           proxy_port;
    int                charset;
    uint8_t            supportfield;
    int                wait_timeout;
```

```
struct mms_send_param_t    send_param;
};
```

● 参数

类型	参数	描述
int	<i>context_id</i>	数据通道号
int	<i>sim_id</i>	SIM 卡 ID
char *	<i>mmsc_url</i>	MMSC 的地址
char *	<i>proxy_addr</i>	MMS 的代理地址
uint16_t	<i>proxy_port</i>	MMS 的代理端口
int	<i>charset</i>	MMS 编码方式，详情请参考第 3.4.2.2 章。
uint8_t	<i>supportfield</i>	是否需要在 HTTP 报文表头中携带主叫号码信息
int	<i>wait_timeout</i>	发送的最大超时时间
struct mms_send_param_t	<i>send_param</i>	发送参数，详情请参考第 3.4.2.3 章。

3.4.2.2. mms_charset_type

MMS 编码方式枚举定义如下：

```
typedef enum{
    MMS_CHARSET_ASCII_VALUE        = 0x03,
    MMS_CHARSET_UTF8_VALUE         = 0x6A,
    MMS_CHARSET_UCS2_VALUE         = 0x03E8,
    MMS_CHARSET_BIG5_VALUE         = 0x07EA,
    MMS_CHARSET_GBK_VALUE          = 0x71,
}mms_charset_type;
```

● 参数

参数	描述
<i>MMS_CHARSET_ASCII_VALUE</i>	ASCII 编码方式
<i>MMS_CHARSET_UTF8_VALUE</i>	UTF8 编码方式
<i>MMS_CHARSET_UCS2_VALUE</i>	UCS2 编码方式

<i>MMS_CHARSET_BIG5_VALUE</i>	BIG5 编码方式
<i>MMS_CHARSET_GBK_VALUE</i>	GBK 编码方式

3.4.2.3. mms_send_param_t

MMS 发送参数定义如下：

```
struct mms_send_param_t{
    int        validity_period;
    int        priority;
    int        delivery_report;
    int        read_report;
    int        visible;
    int        class;
};
```

● 参数

类型	参数	描述
int	<i>validity_period</i>	MMS 有效期属性，详情请参考第 3.4.2.4 章
int	<i>priority</i>	MMS 优先级属性，详情请参考第 3.4.2.5 章
int	<i>delivery_report</i>	0 表示不需要递交报告；1 表示需要递交报告
int	<i>read_report</i>	0 表示不需要读取报告，1 表示需要读取报告
int	<i>visible</i>	MMS 可见性属性，详情请参考第 3.4.2.6 章
int	<i>class</i>	MMS 类型属性，详情请参考第 3.4.2.7 章

3.4.2.4. mms_valid_type

MMS 有效期属性枚举定义如下：

```
typedef enum {
    MMS_PARAM_VALID_1HOUR=0,
    MMS_PARAM_VALID_12HOURS,
    MMS_PARAM_VALID_24HOURS,
    MMS_PARAM_VALID_2DAYS,
    MMS_PARAM_VALID_1WEEK,
    MMS_PARAM_VALID_MAXIMUM,
```

```
MMS_PARAM_VALID_DEFAULT
}mms_valid_type;
```

● 参数

参数	描述
<i>MMS_PARAM_VALID_1HOUR</i>	有效期 1 小时
<i>MMS_PARAM_VALID_12HOURS</i>	有效期 12 小时
<i>MMS_PARAM_VALID_24HOURS</i>	有效期 24 小时
<i>MMS_PARAM_VALID_2DAYS</i>	有效期 2 天
<i>MMS_PARAM_VALID_1WEEK</i>	有效期 1 周
<i>MMS_PARAM_VALID_MAXIMUM</i>	有效期 0x7FFFFFFF 秒
<i>MMS_PARAM_VALID_DEFAULT</i>	不设置有效期，由网络决定

3.4.2.5. mms_priority_type

MMS 优先级属性枚举定义如下：

```
typedef enum {
    MMS_PARAM_PRIORITY_LOWEST=0,
    MMS_PARAM_PRIORITY_NORMAL,
    MMS_PARAM_PRIORITY_HIGHEST,
    MMS_PARAM_PRIORITY_DEFAULT
}mms_priority_type;
```

● 参数

参数	描述
<i>MMS_PARAM_PRIORITY_LOWEST</i>	最低优先级
<i>MMS_PARAM_PRIORITY_NORMAL</i>	正常优先级
<i>MMS_PARAM_PRIORITY_HIGHEST</i>	最高优先级
<i>MMS_PARAM_PRIORITY_DEFAULT</i>	默认优先级，由网络决定

3.4.2.6. mms_visible_type

MMS 可见性属性枚举定义如下：

```
typedef enum {
    MMS_PARAM_SENDERADDR_HIDE=0,
    MMS_PARAM_SENDERADDR_SHOW,
    MMS_PARAM_SENDERADDR_DEFAULT
}mms_visible_type;
```

● 参数

参数	描述
<i>MMS_PARAM_SENDERADDR_HIDE</i>	隐藏
<i>MMS_PARAM_SENDERADDR_SHOW</i>	可见
<i>MMS_PARAM_SENDERADDR_DEFAULT</i>	默认，由网络决定

3.4.2.7. mms_class_type

MMS 类型属性枚举定义如下：

```
typedef enum{
    MMS_PARAM_CLASS_PERSONAL=0,
    MMS_PARAM_CLASS_ADVERTISEMENT,
    MMS_PARAM_CLASS_INFORMATIONAL,
    MMS_PARAM_CLASS_AUTO,
    MMS_PARAM_CLASS_DEFAULT
}mms_class_type;
```

● 参数

参数	描述
<i>MMS_PARAM_CLASS_PERSONAL</i>	个人类
<i>MMS_PARAM_CLASS_ADVERTISEMENT</i>	广告类
<i>MMS_PARAM_CLASS_INFORMATIONAL</i>	信息类
<i>MMS_PARAM_CLASS_AUTO</i>	自动
<i>MMS_PARAM_CLASS_DEFAULT</i>	默认，由网络决定

3.4.2.8. mms_info_t

MMS 相关信息的定义如下：

```
struct mms_info_t{
    struct mms_subject_t    *subject;
    struct mms_addr_t      *to_addr;
    struct mms_addr_t      *cc_addr;
    struct mms_addr_t      *bcc_addr;
    struct mms_attachment_t *attachments;
};
```

● 参数

类型	参数	描述
struct mms_subject_t *	<i>subject</i>	主题，详情请参考第 3.4.2.9 章
struct mms_addr_t *	<i>to_addr</i>	收件人地址，详情请参考第 3.4.2.10 章
struct mms_addr_t *	<i>cc_addr</i>	抄送人地址，详情请参考第 3.4.2.10 章
struct mms_addr_t *	<i>bcc_addr</i>	密件抄送人地址，详情请参考第 3.4.2.10 章
struct mms_attachment_t *	<i>attachments</i>	附件信息，详情请参考第 3.4.2.11 章

3.4.2.9. mms_subject_t

MMS 主题的定义如下：

```
struct mms_subject_t{
    char    *subject;
    int     length;
};
```

● 参数

类型	参数	描述
char *	<i>subject</i>	主题内容
int	<i>length</i>	主题内容长度

3.4.2.10.mms_addr_t

MMS 收件人、抄送人或密件抄送人的地址定义如下：

```
struct mms_addr_t{
    struct mms_addr_t *next;
    int                addr_len;
    char               *address;
};
```

● 参数

类型	参数	描述
struct mms_addr_t *	<i>next</i>	下一个节点
int	<i>addr_len</i>	号码或地址长度
char *	<i>address</i>	号码或地址

3.4.2.11.mms_attachment_t

MMS 附件的定义如下：

```
struct mms_attachment_t{
    struct mms_attachment_t *next;
    char                     *file_path;
};
```

● 参数

类型	参数	描述
struct mms_attachment_t *	<i>next</i>	下一个节点
char *	<i>file_path</i>	附件所在的文件路径

3.4.3. ql_mms_client_release

该函数用于释放 MMS 客户端句柄及其相关资源。

- 函数原型

```
int ql_mms_client_release(ql_mms_client *client_hdl)
```

- 参数

client_hdl:

[In] MMS 客户端句柄。由 *ql_mms_client_new()* 获取。

- 返回值

详情请参考第 3.4.1.1 章。

4 附录 参考文档及术语缩写

表 3：参考文档

文档名称
[1] Quectel_ECx00U&EGx00U 系列_QuecOpen_CSDK_快速开发指导
[2] Quectel_ECx00U&EGx00U 系列_QuecOpen_数据拨号_API_参考手册
[3] Quectel_ECx00U&EGx00U 系列_QuecOpen_HTTP_API_参考手册

表 4：术语缩写

缩写	英文全称	中文全称
API	Application Programming Interface	应用程序编程接口
ASCII	American Standard Code for Information Interchange	美国信息交换标准码
DNS	Domain Name Server	域名系统（服务）协议
GBK	Chinese Internal Code Extension Specification	汉字内码扩展规范
HTTP	Hypertext Transfer Protocol	超文本传输协议
ID	Identifier	标识符
IoT	Internet of Things	物联网
LTE	Long-Term Evolution	长期演进
MMS	Multimedia Messaging Service	多媒体信息服务
MMSC	Multimedia Messaging Service Center	多媒体信息服务中心
RTOS	Real-Time Operating System	实时操作系统
SDK	Software Development Kit	软件开发工具包

SIM	Subscriber Identity Module	用户身份识别模块
UCS	Universal Character Set	通用字符集
URL	Uniform Resource Locator	统一资源定位符
UTF	Unicode Transformation Format	Unicode 转换格式