

# ECx00U&EGx00U 系列

# QuecOpen 摄像头 API

# 参考手册

**LTE Standard 模块系列**

版本：1.0

日期：2021-09-10

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](mailto: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](mailto: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失承担责任，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
-	2020-12-01	Chaos HUANG	文档创建
1.0	2021-09-10	Chaos HUANG	受控版本

## 目录

文档历史 .....	3
目录 .....	4
表格索引 .....	5
图片索引 .....	6
<b>1 引言 .....</b>	<b>7</b>
1.1. 适用模块 .....	7
<b>2 摄像头 API 介绍 .....</b>	<b>8</b>
2.1. 头文件 .....	8
2.2. 函数概览 .....	8
2.3. 摄像头相关 API .....	9
2.3.1. ql_CamInit .....	9
2.3.1.1. ql_errcode_camera_e .....	9
2.3.2. ql_CamPowerOn .....	10
2.3.3. ql_CamClose .....	11
2.3.4. ql_CamGetSensorInfo .....	11
2.3.4.1. ql_cam_drv_s .....	11
2.3.4.2. ql_cam_status_s .....	12
2.3.5. ql_CamPreview .....	13
2.3.6. ql_CamStopPreview .....	13
2.3.7. ql_CamCaptureImage .....	13
2.3.8. ql_camPrint .....	14
2.3.9. ql_CamSetBufopt .....	14
2.3.9.1. ql_buf_config_e .....	14
<b>3 摄像头开发示例及调试 .....</b>	<b>16</b>
3.1. 开发示例 .....	16
3.2. 相关调试 .....	17
<b>4 附录 参考文档及术语缩写 .....</b>	<b>19</b>

## 表格索引

表 1: 适用模块 .....	7
表 2: 函数概览 .....	8
表 3: 参考文档 .....	19
表 4: 术语缩写 .....	19

## 图片索引

图 1: ql_camera_app_init() .....	16
图 2: 摄像头 API 示例注释 .....	17
图 3: COM 设备图 .....	17
图 4: 摄像头 Log 1.....	18
图 5: 摄像头 Log 2.....	18

# 1 引言

移远通信 LTE Standard ECx00U 系列和 EGx00U 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 模块摄像头相关 API、开发示例及功能调试。

## 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 摄像头 API 介绍

### 2.1. 头文件

摄像头 API 头文件为 `ql_api_camera.h`，位于 SDK 包 `components\ql-kernel\inc` 目录下。若无特别说明，本文档所述头文件均在该目录下。

### 2.2. 函数概览

表 2：函数概览

函数	说明
<code>ql_CamInit()</code>	初始化摄像头
<code>ql_CamPowerOn()</code>	打开摄像头
<code>ql_CamClose()</code>	关闭摄像头
<code>ql_CamGetSensorInfo()</code>	获取摄像头配置信息
<code>ql_CamPreview()</code>	打开摄像头预览功能
<code>ql_CamStopPreview()</code>	关闭摄像头预览功能
<code>ql_CamCaptureImage()</code>	启动一次摄像头拍照功能
<code>ql_camPrint()</code>	将图像输出到 LCD
<code>ql_CamSetBufopt()</code>	设置摄像头的 buffer 数量

## 2.3. 摄像头相关 API

### 2.3.1. ql\_CamInit

该函数用于初始化摄像头。

- 函数原型

```
ql_errcode_camera_e ql_CamInit(uint16_t width, uint16_t height)
```

- 参数

*width:*

[In] 摄像头图像宽度。

*height:*

[In] 摄像头图像高度。

- 返回值

详见第 2.3.1.1 章。

### 备注

当前模块最高支持 30 万像素摄像头，图像最大尺寸可为 640 × 480。

#### 2.3.1.1. ql\_errcode\_camera\_e

错误码枚举定义如下：

```
typedef enum
{
    QL_CAMERA_SUCCESS = QL_SUCCESS,

    QL_CAMERA_INIT_ERR                = 1|QL_CAMERA_ERRCODE_BASE,
    QL_CAMERA_POWER_ON_ERR,
    QL_CAMERA_CLOSE_ERR,
    QL_CAMERA_PREVIEW_ERR,
    QL_CAMERA_STOP_PREVIEW_ERR,
    QL_CAMERA_CAPTURE_ERR,
    QL_CAMERA_GET_INFO_ERR,
    QL_CAMERA_PRINT_ERR,
    QL_CAMERA_QBUF_ERR,
```

```
QL_CAMERA_SET_BUF_ERR,  
}ql_errcode_camera_e
```

● 参数

参数	描述
QL_CAMERA_SUCCESS	函数执行成功
QL_CAMERA_INIT_ERR	摄像头初始化失败
QL_CAMERA_POWER_ON_ERR	摄像头打开失败
QL_CAMERA_CLOSE_ERR	摄像头关闭失败
QL_CAMERA_PREVIEW_ERR	摄像头预览失败
QL_CAMERA_STOP_PREVIEW_ERR	摄像头停止预览失败
QL_CAMERA_CAPTURE_ERR	摄像头摄影失败
QL_CAMERA_GET_INFO_ERR	摄像头信息获取失败
QL_CAMERA_PRINT_ERR	摄像头播放失败
QL_CAMERA_QBUF_ERR	摄像头传递 buffer 失败
QL_CAMERA_SET_BUF_ERR	摄像头设置 buffer 失败

### 2.3.2. ql\_CamPowerOn

该函数用于打开摄像头。

● 函数原型

```
ql_errcode_camera_e ql_CamPowerOn(void)
```

● 参数

无

● 返回值

详见第2.3.1.1章。

### 2.3.3. ql\_CamClose

该函数用于关闭摄像头。

- 函数原型

```
ql_errcode_camera_e ql_CamClose(void)
```

- 参数

无

- 返回值

详见第2.3.1.1章。

### 2.3.4. ql\_CamGetSensorInfo

该函数用于获取摄像头配置信息。

- 函数原型

```
ql_errcode_camera_e ql_CamGetSensorInfo(ql_cam_drv_s *pCamDevice)
```

- 参数

*pCamDevice*:

[Out] 摄像头配置信息。详见第2.3.4.1章。

- 返回值

详见第2.3.1.1章。

#### 2.3.4.1. ql\_cam\_drv\_s

摄像头配置信息结构体定义如下：

```
typedef struct
{
    char *pNamestr;
    uint32_t img_width;
    uint32_t img_height;
    uint32_t nPixels;
    ql_cam_status_s Status;
```

```
} ql_cam_drv_s
```

● 参数

类型	参数	描述
char *	<i>pNamestr</i>	摄像头设备名称
uint32_t	<i>img_width</i>	图像宽度
uint32_t	<i>img_height</i>	图像高度
uint32_t	<i>nPixels</i>	图像像素。最大像素可为 614400 字节。
<i>ql_cam_status_s</i>	<i>Status</i>	摄像头设备状态。详见第 2.3.4.2 章。

#### 2.3.4.2. ql\_cam\_status\_s

摄像头设备状态定义如下：

```
typedef struct
{
    bool Cam_Init_status;
    bool Cam_Power_status;
    bool Cam_Preview_status;
} ql_cam_status_s
```

● 参数

类型	参数	描述
bool	<i>Cam_Init_status</i>	摄像头初始化状态
bool	<i>Cam_Power_status</i>	摄像头上电状态
bool	<i>Cam_Preview_status</i>	摄像头预览状态

### 2.3.5. ql\_CamPreview

该函数用于打开摄像头预览功能。

- 函数原型

```
ql_errcode_camera_e ql_CamPreview(void)
```

- 参数

无

- 返回值

详见第2.3.1.1章。

### 2.3.6. ql\_CamStopPreview

该函数用于关闭摄像头预览功能。

- 函数原型

```
ql_errcode_camera_e ql_CamStopPreview(void)
```

- 参数

无

- 返回值

详见第2.3.1.1章。

### 2.3.7. ql\_CamCaptureImage

该函数用于启动一次摄像头拍照功能。

- 函数原型

```
ql_errcode_camera_e ql_CamCaptureImage(uint16_t **pFrameBuf)
```

- 参数

*pFrameBuf*:

[Out] 指向保存所得照片数据的指针。

- 返回值

详见第2.3.1.1章。

### 2.3.8. ql\_camPrint

该函数用于将图像输出到 LCD。

- 函数原型

```
ql_errcode_camera_e ql_camPrint(uint16_t *imageBuf)
```

- 参数

*imageBuf*:

[In] 被显示的图像。需要显示图像预览时，该参数值输入 *NULL*。

- 返回值

详见第2.3.1.1章。

### 2.3.9. ql\_CamSetBufopt

该函数用于设置摄像头的 buffer 数量。

- 函数原型

```
ql_errcode_camera_e ql_CamSetBufopt(ql_buf_config_e buf_cfg)
```

- 参数

*buf\_cfg*:

[In] 摄像头 buffer 数量。详见第2.3.9.1章。

- 返回值

详见第2.3.1.1章。

#### 2.3.9.1. ql\_buf\_config\_e

摄像头 buffer 数量枚举定义如下：

```
typedef enum
{
```

```

    single_buf,
    double_buf,
}ql_buf_config_e
    
```

● 参数

参数	描述
<i>single_buf</i>	单 buffer
<i>double_buf</i>	双 buffer



## 3 摄像头开发示例及调试

### 3.1. 开发示例

ECx00U 系列和 EGx00U 模块 QuecOpen SDK 代码中提供了摄像头 API 的示例文件，即 `camera_demo.c`，路径为 `components\ql-application\camera`。该文件中主要包含了如何初始化、打开和关闭摄像头等内容。

该示例运行时将首先初始化摄像头、打开摄像头及其预览功能；在 20 秒预览后进行 10 次拍照，如已连接 LCD，图像将依次在 LCD 上显示；拍照结束后将停止预览并关闭摄像头。

该示例入口函数为 `ql_camera_app_init()`，如下图所示。

```
void ql_camera_app_init(void)
{
    QICAMERAStatus err = QL_OSI_SUCCESS;
    ql_task_t camera_task = NULL;

    err = ql_rtos_task_create(&camera_task, QL_CAMERA_TASK_STACK_SIZE, QL_CAMERA_TASK_PRIO, "camera DEMO", ql_camera_demo_thread, NULL, QL_CAMERA_TASK_EVENT_CNT);
    if (err != QL_OSI_SUCCESS)
    {
        QL_CAMERADEMO_LOG("CAMERA demo task created failed");
    }
}
```

图 1: `ql_camera_app_init()`

如图 2 所示，上述示例已在 `ql_init_demo_thread` 线程中默认不启动，需要测试时请取消注释。注意，因摄像头需要的部分功能引脚和 `ql_ledcfg_app_init()` 中使用的引脚重复，从而产生冲突使摄像头无法正常工作，因此在使用该示例时需注释 `ql_ledcfg_app_init()`。此外，因 I2C API 示例文件和 decoder 示例文件运行时均需使用摄像头，在运行该示例时不可同时打开 `ql_i2c_demo_init()` 和 `ql_decoder_app_init()`，否则三个示例文件将在摄像头的使用上产生冲突，甚至导致模块死机。

```

20:
21: static void ql_init_demo_thread(void *param)
22: {
23:     QL_INIT_LOG("init demo thread enter, param 0x%x", param);
24:     #if 0
25:         ql_gpio_app_init();
26:         ql_gpioint_app_init();
27:     #endif
28:
29:     #ifdef QL_APP_FEATURE_CAMERA
30:         //ql_camera_app_init();
31:     #endif
32:
33:     #ifdef QL_APP_FEATURE_LEDCFG
34:         ql_ledcfg_app_init();
35:     #endif
36:
37:     #ifdef QL_APP_FEATURE_AUDIO
38:         ql_audio_app_init();
39:     #endif
40:     #ifdef QL_APP_FEATURE_LCD
41:         //ql_lcd_app_init();
42:     #endif
43:     ql_nw_app_init();
44:     //ql_datacall_app_init();

```

图 2: 摄像头 API 示例注释

## 3.2. 相关调试

ECx00U 系列和 EGx00U QuecOpen 模块可通过移远通信 LTE OPEN EVB 进行摄像头功能调试。

将编译版本烧录到模块中，使用 USB 线连接 LTE OPEN EVB 的 USB 端口和 PC。使用 coolwatcher\_debughost.exe 的 Trace Tool 抓取 log 后，可通过 USB 的 AP Log 端口（如下图所示）查看该示例的调试信息。

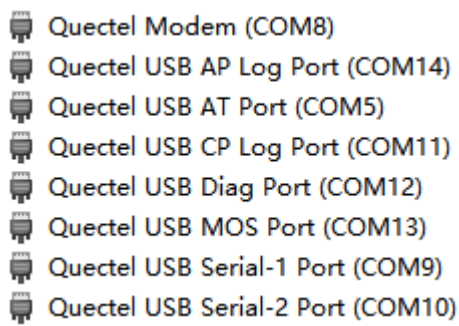


图 3: COM 设备图

模块开机后自动启动 `ql_camera_app_init()`，AP Log 端口输出的调试信息如图 4、图 5 所示：

```
[ql_CAMERADEMO][ql_camera_demo_thread, 99] Camera demo entry
cam: quec_CamInit
cam: _drvCamLoad
cam: _drvCamGetCfgCount count=2
power_id = CAMD, step = 12500, min_uv = 1400000, max_uv = 2187500
power_id = CAMA, step = 12500, min_uv = 1612500, max_uv = 3200000
cam get id 0x23,0x2a
CamGc032aGetSensorInfo 80c0ba14
[ql_api_camera][ql_CamGetSensorInfo, 175] Get cam sensor Info
cam: drvCamGetSensorInfo
[ql_api_camera][ql_CamInit, 220] Cam init success
cam: drvCamPowerOn
camGc032aOpen
power_id = CAMD, step = 12500, min_uv = 1400000, max_uv = 2187500
power_id = CAMA, step = 12500, min_uv = 1612500, max_uv = 3200000
[ql_api_camera][ql_CamPowerOn, 81] Cam power on success
[ql_CAMERADEMO][ql_camera_demo_thread, 113] testID[0] = 0x0, testID[1] = 0x0
[ql_CAMERADEMO][ql_camera_demo_thread, 117] testID[0] = 0x23, testID[1] = 0x2a
cam: drvCamStartPreview
camGc032aPrevStart gc032aInfo.preview_page=0
[ql_api_camera][ql_CamPreview, 124] Cam preview success
[ql_api_camera][ql_camPrint, 238] Cam lcd width = 240 height = 320
```

图 4: 摄像头 Log 1

```
[ql_api_camera][ql_CamCaptureImage, 165] Cam capture success
cam: drvCamCaptureImage
[ql_api_camera][ql_CamCaptureImage, 165] Cam capture success
cam: drvCamStopPreview
[ql_api_camera][ql_CamStopPreview, 148] Cam stop preview success
cam:drvCamClose,osiSemaphoreDelete
```

图 5: 摄像头 Log 2

## 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

表 4: 术语缩写

缩写	英文全称	中文全称
AP	Application Processor	应用处理器
API	Application Programming Interface	应用程序接口
APP	Application	应用程序
COM	Communication	通信
EVB	Evaluation Board	评估板
GPIO	General-Purpose Input/Output	通用型输入/输出
IoT	Internet of Things	物联网
LCD	Liquid Crystal Display	液晶显示屏
RTOS	Real-Time Operating System	实时操作系统
SDK	Software Development Kit	软件开发工具包
SPI	Serial Peripheral Interface	串行外设接口
USB	Universal Serial Bus	通用串行总线