

Optimus SDK

#### 为用户ICU模块与计算机之间建立一个灵活便捷的蓝牙数据传输通道。

Optimus SDK 是一个基于Optimus ICU模块的软件开发套件，它在 FPGA 与计算机之间建立了一个灵活便捷的蓝牙数据传输通道，并为用户提供了一套简单易用的软件编程接口，使用户在开发蓝牙数据传输系统时更简单、高效，从而在很大程度上缩短用户的产品开发周期。

#### 版权声明

版权所有©2021-2024 上海脑虎科技有限公司，保留所有权利。

未经本公司明确书面许可的情况下，任何单位或个人不得对本文档的部分或全部进行摘抄、复制，并不得以任何形式进行传播。

本文档所涉及的其它公司、组织或个人的产品、商标、专利，除非特别声明，归各自所有人所有。

#### 修订记录

|  |  |
| --- | --- |
| **修订日期** | **修订内容** |
| 2024/07/19 | 最初版本 |

[目录 4](#_Toc172292482)

[Optimus SDK 概述 5](#_Toc172292483)

[Optimus SDK 使用简介 6](#_Toc172292484)

[G3Controller 6](#_Toc172292485)

[API (应用程序编程接口) 7](#_Toc172292486)

[API 参考手册 8](#_Toc172292487)

[加载库 8](#_Toc172292488)

[G3\_ECU类 8](#_Toc172292489)

[G3Controller类 8](#_Toc172292490)

# Optimus SDK 概述

Optimus SDK 是基于 Optimus ICU模块的软件开发套件，旨在使用户在开发蓝牙数据传输系统时更简单、高效、省时、省成本。它在 FPGA 与计算机之间建立了一个灵活便捷的蓝牙数据传输通道，并为用户提供了一套简单易用的软件编程接口。用户使用 Optimus SDK 提供的 API可以实现上位机与 Optimus ICU 模块之间的通信。Optimus SDK 有以下几类功能：

1. 设备发现
2. FPGA 配置
3. 使用 wire，trigger，pipe 进行 PC 与 FPGA 之间的通信
4. 抽象出了一个基于蓝牙接口的通用开发平台

# Optimus SDK 使用简介

Optimus SDK 的主要作用是在 Optimus ICU 模块与 PC 之间建立一个便利且有效的数据通道，用户可以很容易地将现有的或新的 FPGA 设计与其建立连接。更重要的是，Optimus SDK 使用户不需要考虑物理接口（蓝牙）部分的具体实现，这在很大程度上减少了用户在产品开发过程中的风险和时间。

蓝牙接口的所有操作都是由上位机发起的。FPGA 端生成上传信号或数据后需要上位机主动查询来接收。

## G3Controller

在 Optimus SDK中，G3Controller可以是 Wire、Trigger 或 Pipe，并且数据传输方向可以是 In 或 Out。通过定义，数据传输方向是以 FPGA 为视角，所有向用户设计输入数据的 G3Controller都是 In 方向，而所有由用户设计输出数据的 G3Controller都是 Out 方向。在一个设计中，所有的 G3Controller都是使用 NxG3Controller动态库来实例化。

下表是对这四种G3Controller类型的概述，后文有对它们的详细介绍。

|  |  |  |
| --- | --- | --- |
| G3Controller | 同步/异步 | 描述 |
| Wire In | 同步 | PC 向FPGA输入一个信号状态。  （例如：虚拟按钮或开关） |
| Wire Out | 同步 | FPGA 向 PC 输出一个信号状态。  （例如：虚拟 LED 或数码管的显示） |
| Trigger In | 同步 | PC 向FPGA 输入一个可以和特定时钟同相的单脉冲  信号。（例如：用于启动状态机的按钮） |
| Trigger Out | 同步 | FPGA 通知 PC 有一个特定的事件发生。  （例如：状态机发出 Done 信号后 PC 上弹出窗口告知用户可以开始数据传输） |
| Pipe In | 同步 | PC 向FPGA 输入批量数据。  （例如：存储器的数据下载，流数据） |
| Pipe Out | 同步 | FPGA向 PC 输出批量数据。（例如：存储器的数据  上传，读取计算结果） |
| Block Pipe In | 同步 | 同 Pipe In，增加对数据块流控信号。 |
| Block Pipe Out | 同步 | 同 Pipe Out，增加对数据块流控信号。 |

# API (应用程序编程接口)

Optimus API 包含了能通过蓝牙接口实现通信的函数，这些函数经过特别设计，仅适用Optimus ICU模块，并且用于与模块中的 FPGA 建立连接。Optimus API 还提供了一些函数，用于与 Optimus ICU模块（Wire，Trigger，Pipe）直接建立连接。这样的抽象化，虽然牺牲了硬件接口的一些灵活性，却为 Optimus 软件带来了很大的灵活性和便利性，从而可以在很大程度上减少用户的产品开发周期（使用户软件与 FPGA 建立连接的开发时间以及相关知识的学习时间）。Optimus API 以动态链接库（DLL）形式提供，用户可以在 Windows 平台使用，并且支持C 和 C++。

### API 参考手册

本用户手册提供的 API 文档给出了关于如何使用 Optimus API 的总体概述。Optimus API 以动态链接库形式提供，您需要将它包含在您的应用程序中。该动态链接库的接口是 C++，提供的是 C++封装，这让它看起来好像是您的应用程序中的一个 C++类。目前，该库包含了一个控制类G3Controller，以及一个设备类G3\_ECU您需要在您的代码中实例化。如下表所示。

|  |  |
| --- | --- |
| 类 | 描述 |
| G3Controller | 用于查找设备、配置设备以及与设备通信的基本类。 |
| G3\_ECU | 用于进行蓝牙配置和连接。 |

### 加载库

Optimus API 是一个动态链接库，它必须在您的应用程序运行时被加载。静态加载以 Microsoft Visual Studio 为例，将 include和 lib 这两个文件夹拷贝至工程目录下（.vcxproj 所在目录），然后在工程中配置即可。

### G3\_ECU类

它是 Optimus API 的设备主体，用于蓝牙设备连接在一个典型应用中，您的软件需要执行以下步骤：

1. 配置蓝牙UUID。
2. 配置蓝牙连接名称
3. 创建一个 G3\_ECU类。
4. 打开蓝牙设备。

### G3Controller类

它是 Optimus API 的主体，可分为 3 组函数：设备交互、设备配置和 FPGA 通信。在一个典型应用中，您的软件需要执行以下步骤：

1. 创建一个 G3\_ECU类。
2. 创建一个 G3Controller类。

具体函数功能描述如下表所示

|  |  |
| --- | --- |
| 函数 | 描述 |
| getPID | 用于获取设备PID。 |
| getDeviceId | 用于获取设备ID。 |
| getVID | 用于获取设备VID。 |
| getBoardId | 用于获取设备BoardId。 |
| open | 打开蓝牙设备。 |
| isSynthetic | 是否模拟数据默认返回false。 |
| isPlayback | 是否回放数据默认返回false。 |
| uploadFPGABitfile | 加载FPGA程序，暂时无用。 |
| resetBoard | 重置设备 |
| isControllerAvailable | 控制器是否可用默认为true |
| samplingControl | 采样控制，可直接控制FPGA采样引脚电平实现采集开始与停止 |
| Run | 开始采集指令设置为1 |
| setWireInValue | 开始采集指令设置为0 |
| activeTriggerIn | 开始采集指令设置为0与 setWireInValue连用 |
| startRecord | 自动配置开始采集流程 |
| stopRecord | 自动配置停止采集流程 |
| isRunning | 检查是否在采集 |
| flush | 清空FPGA缓存 |
| resetFpga | 重置FPGA程序 |
| readDataBlock | 读取单次采集数据 |
| readDataBlocks | 按数量读取缓存队列中数据 |
| readDataBlocksRaw | 按数量读取缓存中数据 |
| setContinuousRunMode | 设置是否连续运行 |
| setMaxTimeStep | 设置最大时间步长 |
| setCableDelay | 设置延时 |
| setDspSettle | 设置dsp模式 |
| setSampleRate | 设置采样率 |
| selectAuxCommandLength | 设置配置寄存器指令长度 |
| selectAuxCommandBank | 设置配置虚拟寄存器位置 |
| uploadCommandList | 更新配置寄存器指令 |