

**QuePython**

**QuecPython多通道IV测试系统软件设计文档**

**LTE系列**

版本：QuecPython多通道IV测试系统软件设计文档\_V1.0

日期：2022-10-26

 状态：临时文件

www.quectel.com

|  |
| --- |
|  |
|  |

上海移远通信技术股份有限公司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 51086236 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://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，可随时登陆如下网址：

<http://quectel.com/cn/support/technical.htm>

或发送邮件至：[support@quectel.com](mailto:support@quectel.com)

**前言**

上海移远通信技术股份有限公司提供该文档内容用以支持其客户的产品设计。客户须按照文档中提供的规范、参数来设计其产品。由于客户操作不当而造成的人身伤害或财产损失，本公司不承担任何责任。在未声明前，上海移远通信技术股份有限公司有权对该文档进行更新。

**版权申明**

本文档版权属于上海移远通信技术股份有限公司，任何人未经我司允许而复制转载该文档将承担法律责任。

版权所有 ©上海移远通信技术股份有限公司 2019，保留一切权利。

***Copyright ©*** ***Quectel Wireless Solutions Co., Ltd. 2019.***

# 文档历史

**修订记录**

|  |  |  |  |
| --- | --- | --- | --- |
| **版本** | **日期** | **作者** | **变更表述** |
| 1.0 | 2022-10-26 | Pawn | 初始版本 |

# 目录

[文档历史 4](#_Toc128486729)

[目录 5](#_Toc128486730)

[1 前言 6](#_Toc128486731)

[2 系统框架 7](#_Toc128486732)

[2.1. 硬件系统框架 7](#_Toc128486735)

[2.2. 软件系统框架 7](#_Toc128486736)

[2.2.1. 系统框图 7](#_Toc128486737)

[2.2.2. 上位机 7](#_Toc128486738)

[2.2.2.1. 框架简介 7](#_Toc128486739)

[2.2.2.2. 层次图 8](#_Toc128486740)

[2.2.2.3. 目录结构 9](#_Toc128486741)

[2.2.3. 测试模组 9](#_Toc128486742)

[2.2.3.1. 框架简介 10](#_Toc128486743)

[2.2.3.2. 层次图 11](#_Toc128486744)

[2.2.3.3. 目录结构 11](#_Toc128486745)

[3 关键组件 12](#_Toc128486746)

[3.1. 模块通信组件 12](#_Toc128486747)

[3.1.1. 协议帧定义 12](#_Toc128486748)

[3.1.1.1. SYNC帧 12](#_Toc128486749)

[3.1.1.2. ACK帧 12](#_Toc128486750)

[3.1.1.3. RST帧 13](#_Toc128486751)

[3.1.1.4. VI曲线测试 13](#_Toc128486752)

[3.1.1.5. 阻抗测试 13](#_Toc128486753)

[3.1.1.6. GPIO驱动能力测试 13](#_Toc128486754)

[3.2. 绘图组件 14](#_Toc128486755)

[3.3. 文件读写组件 14](#_Toc128486756)

[3.4. 用户鉴权&认证 14](#_Toc128486757)

[4 系统初始化流程 15](#_Toc128486758)

[5 业务流程 16](#_Toc128486759)

[5.1. IV曲线测试一(接电流源板) 16](#_Toc128486760)

[5.2. IV曲线测试二(接曲线测试仪) 16](#_Toc128486761)

[5.3. 阻抗测试 17](#_Toc128486762)

[5.4. GPIO串电测试 18](#_Toc128486763)

[5.5. GPIO驱动能力测试 19](#_Toc128486764)

[6 案例演示 21](#_Toc128486765)

# 

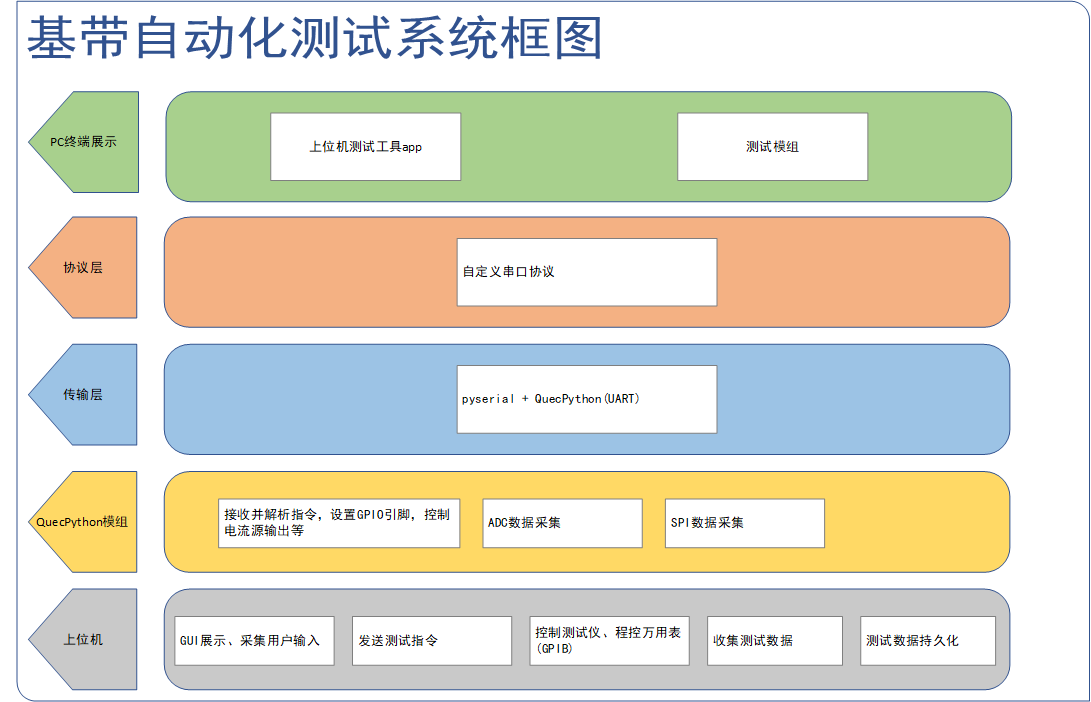
1. 前言

此文档用于描述《QuecPython多通道IV测试系统》 软件设计框架，旨在实现基带多通道IV测试自动化。主要包括系统的软/硬件框架、关键组件、业务流程、案例演示等各方面，帮助读者深入理解该方案/软件/系统的设计原理。

1. 系统框架
3. 1. 硬件系统框架

待补充。

* 1. 软件系统框架
     1. 系统框图



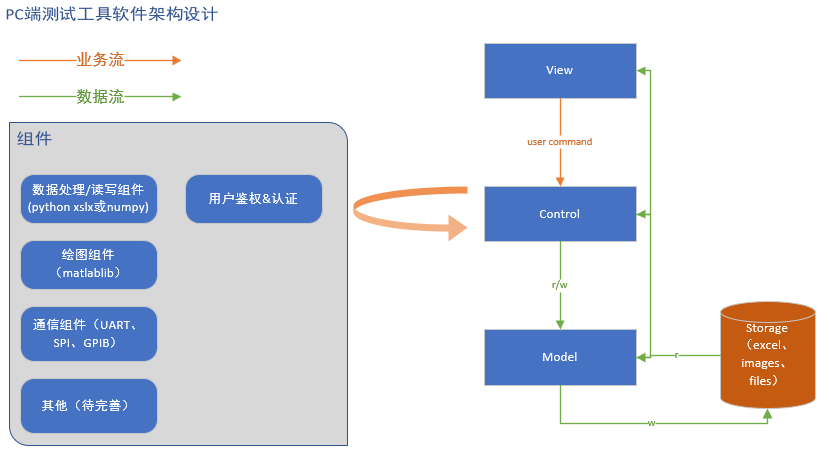
* + 1. 上位机
       1. 框架简介

采用C/S模型，PC端运行client服务进程，模组端运行server服务进程。

基于python的上位机测试工具，负责与模块通讯并下发测试指令，接收模块上传的测试数据并处理，根据测试类型输出对应的图表或测试结果文件。

设计思路：整体采用MVC模型设计。其中：

* + - Model：数据模型层，抽象各种Excel格式，用于存储交互。
    - View：视图层（即GUI图形层），提供用户操作界面（采用wxPython实现）。
    - Control：控制层（业务流程控制），接收视图层用户输入，与模组端交互发送测试指令和接收测试结果，与模型层交互读写数据（文件持久化存储）。



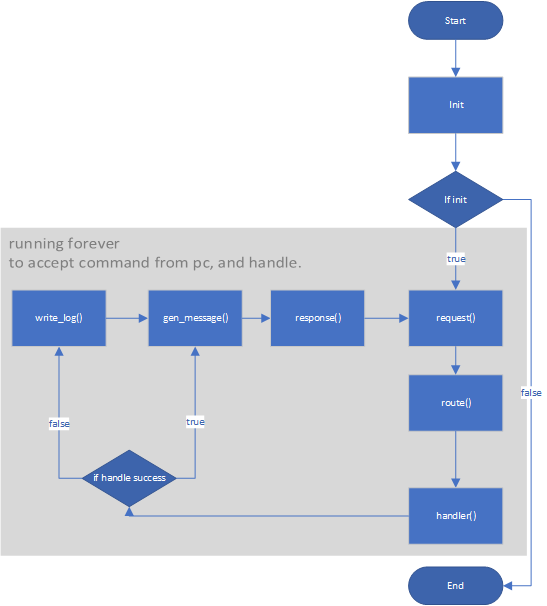
* + - 1. 层次图
      2. 目录结构

QuecTestGUI(工程主目录)

* main.py（主脚本）
* setting.py（应用配置）
* all.log（日志文件）
* QuecTest.ico（窗口图标）
* secret.db（用户数据加密文件）
* docs（文档）
* utils（工具）
  + crc.py（crc校验）
  + ……
* ui（窗口部件包）
  + login.py（登录窗口）
  + mainWin.py（主窗口）
  + vi.py（VI曲线测试窗口）
  + impedance.py（阻抗测试窗口）
  + ability.py（GPIO驱动能力测试窗口）
* logic（业务逻辑控制组件包）
  + boards.py（板级控制）
  + user.py（用户控制）
  + ……
* db（本地持久化组件包）
  + dbio.py（文件io组件）
  + ……
* comm（通信包）
  + protocol.py（协议帧）
  + serial.py（串口通信）
  + device.py（程控万用表、VI曲线测试仪）
  + ……
    1. 测试模组

基于QuecPython的测试系统(模块暂定EC600N)，主要功能为接收上位机的指令，切换测试通道，读取测试引脚的电压(GPIO \* 8 & ADC \* 2)上传至上位机。

* + - 1. 框架简介



如上基本流程图示，其中主要步骤：

1. request：解析指令及其参数。
2. route：根据指令，分发至具体业务处理函数。
3. handler：业务处理函数。
4. response：构建响应（包含指令执行情况，测试状态等）。
   * + 1. 层次图
       2. 目录结构

QuecTestModule(工程主目录)

* main.py（主脚本）
* setting.py（应用配置）
* service.py（主服务循环，请求&响应）
* route.py（路由，注册处理函数&分发处理函数）
* handles.py（业务句柄）
* comm.py（与上位机通信）
* protocol.py（协议帧）
* device.py（设备通信，自制电流板等）

1. 关键组件
   1. 模块通信组件

实现上位机与测试模组互联通信，用于发送测试指令以及接收测试结果。采用自定义协议格式。

* + 1. 协议帧定义

固定协议头标志：buf[0:1] = [0x61,0x74]

数据长度(buf[2])：数据长度N bytes，一个字节表示，包含CRC8校验1字节。

控制指令(buf[4])，如下：

1. SYNC(0x01)
2. ACK(0x02)
3. RST(0x03)
4. VI曲线测试 0x04
5. 阻抗测试 0x05
6. GPIO驱动能力测试 0x06

数据(buf[5:])。

CRC校验(buf[-1:-2])：尾部1字节。

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 4 | 5 | 6 | N |
| 0x61 | 0x74 | <length> | <command> | <param1> | …… | <CRC> |

公共参数（管脚测试表参数）如下(具体传递哪些参数，根据实际需求确定)：

board：板号1~2

channel：通道号0~223

pin\_no：引脚号

pin\_name：引脚名

is\_gpio：是否是gpio

gpio\_no：平台gpio号

* + - 1. SYNC帧

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 4 | 5 | 6 |
| 0x61 | 0x74 | 0x04 | SYNC | <port\_id> | <CRC> |

* + - 1. ACK帧

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 4 | 5 | 6 |
| 0x61 | 0x74 | 0x04 | ACK | <port\_id> | <CRC> |

* + - 1. RST帧

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 4 | 5 | 6 |
| 0x61 | 0x74 | 0x04 | RST | <port\_id> | <CRC> |

* + - 1. VI曲线测试

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 4 | 5 | …… | N |
| 0x61 | 0x74 | <length> | 0x04 | param1 | …… | <CRC> |

* + - 1. 阻抗测试

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 4 | 5 | …… | N |
| 0x61 | 0x74 | <length> | 0x05 | param1 | …… | <CRC> |

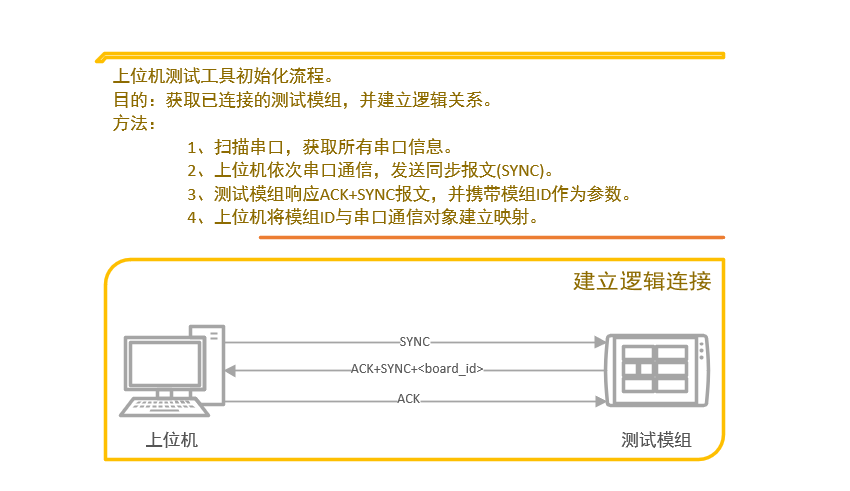
* + - 1. GPIO驱动能力测试

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 4 | 5 | …… | N |
| 0x61 | 0x74 | <length> | 0x06 | param1 | …… | <CRC> |

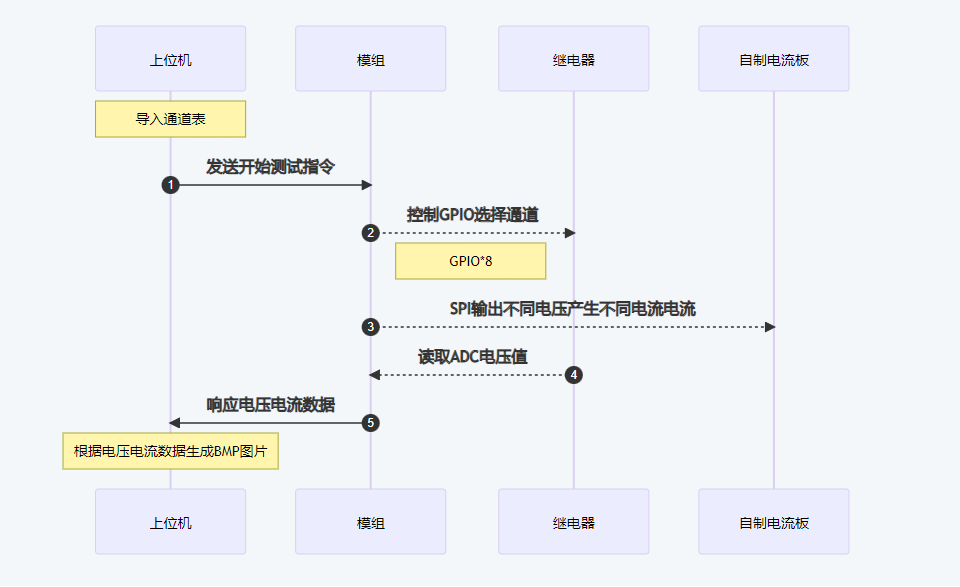
* 1. 绘图组件
  2. 文件读写组件
  3. 用户鉴权&认证

1. 系统初始化流程

<流程图，并在图中或图外辅以必要的文字说明>

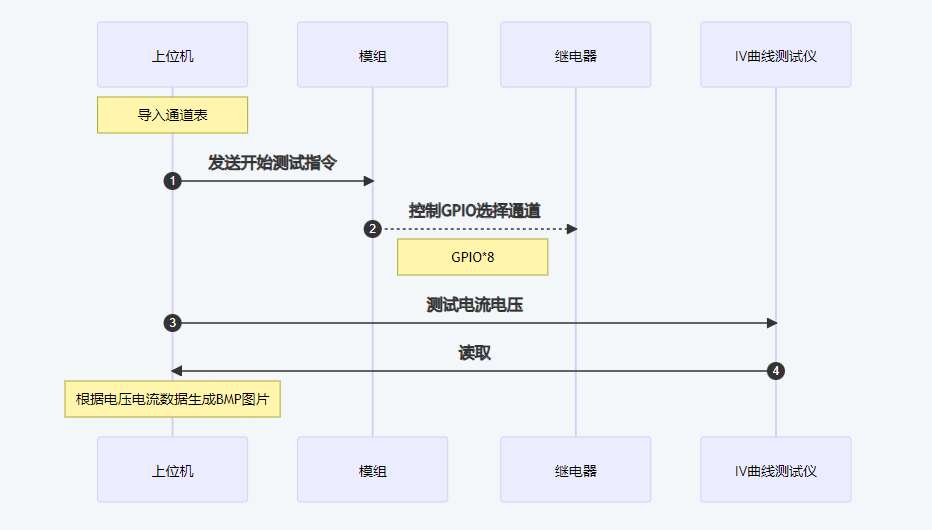


1. 业务流程
   1. IV曲线测试一(接电流源板)
2. 导入通道关系表。
3. 点击Start启动测试。
4. 控制EC600N TE-A 选择通道（从1依次开始）。
5. 控制EC600N TE-A控制SPI DAC输出不同电压产生不同电流。
6. 读取EC600N TE-A ADC值。
7. 根据电流电压数据生成BMP文件，一个通道一份，文件命名：样品编号\_待测模块/芯片型号\_PIN NO\_PIN NAME测试fail的图片放入一个文件夹，测试pass的图片放入另一个文件夹，标准待定，测试fail的管脚同时列表统计。



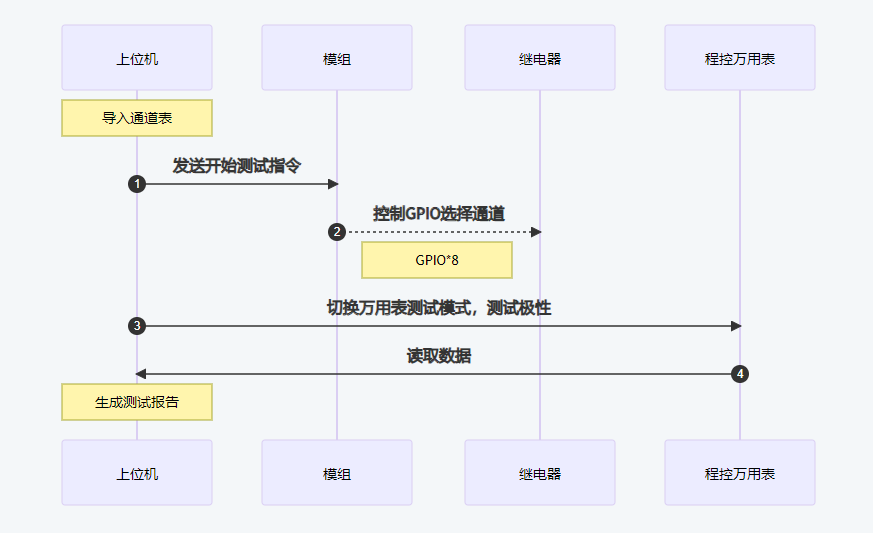
* 1. IV曲线测试二(接曲线测试仪)

1. 导入通道关系表。
2. 点击Start启动测试。
3. 控制EC600N TE-A 选择通道（从1依次开始）。
4. 控制IV曲线测试仪，生成IV曲线BMP文件，一个通道一份，文件命名：样品编号\_待测模块/芯片型号\_PIN NO\_PIN NAME。
5. 测试fail的图片放入一个文件夹，测试pass的图片放入另一个文件夹，标准待定，测试fail的管脚同时列表统计。



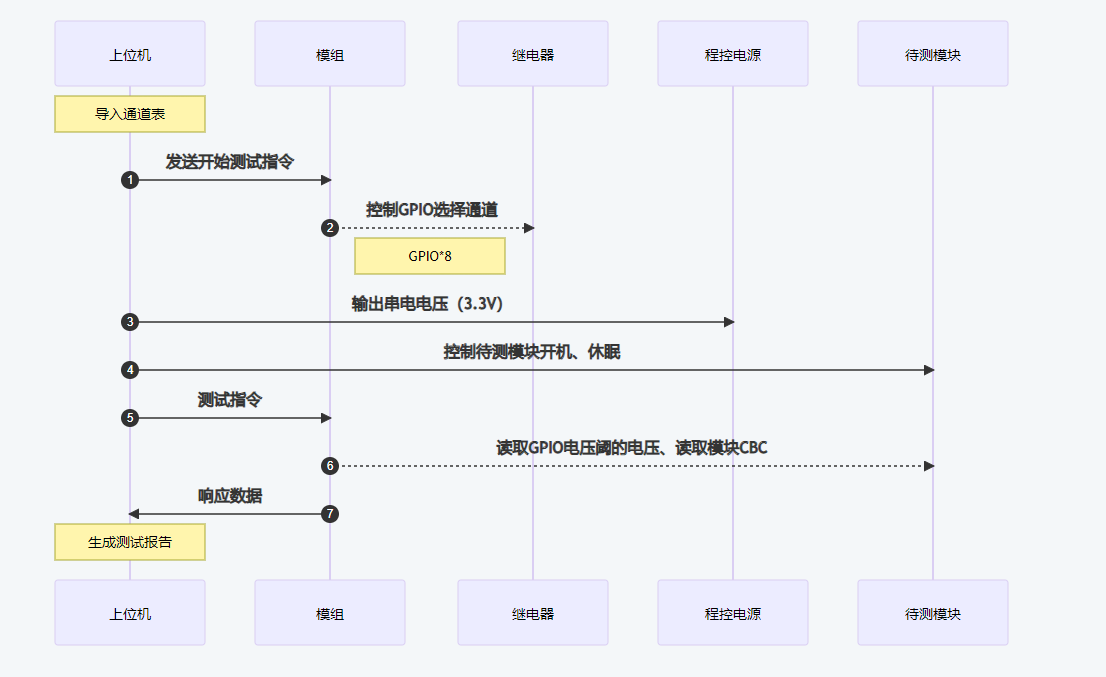
* 1. 阻抗测试

1. 导入通道关系表。
2. 点击Start启动测试。
3. 控制EC600N TE-A 选择通道（从1依次开始）。
4. 切换万用表测试模式，测试极性，读取数据生成测试报告，给出测试结论，测试标准由后台配置。



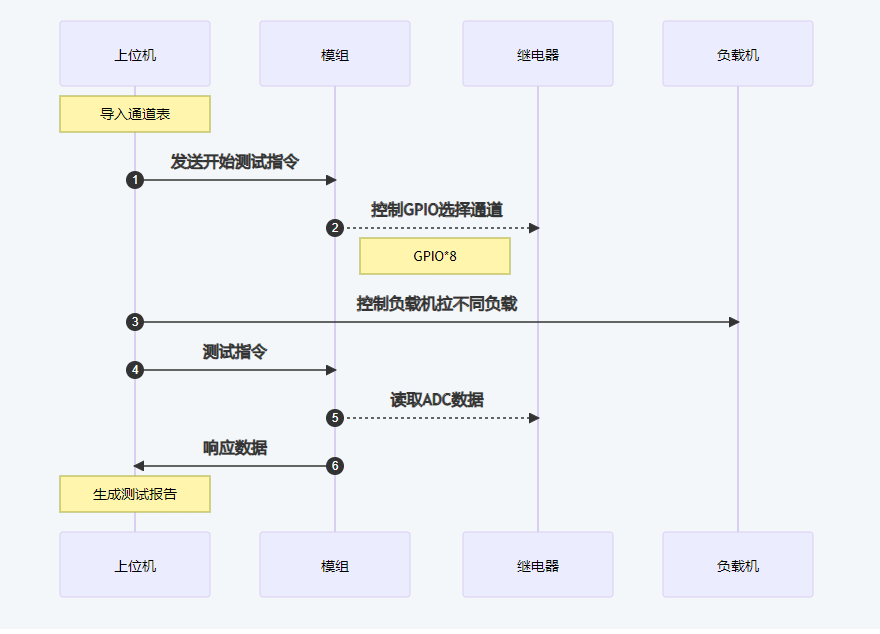
* 1. GPIO串电测试

1. 导入表格
2. 点击Start启动测试
3. 控制EC600N TE-A 选择GPIO通道
4. 控制程控电源输出串电电压，一般为3.3V
5. 控制模块开机，并确认是否开机成功，是否能正常进休眠
6. 控制EC600N TE-A读取GPIO电压阈的电压，读取模块CBC
7. 生成测试报告，给出测试结论，测试标准由后台配置



* 1. GPIO驱动能力测试

1. 导入表格
2. 点击Start启动测试
3. 控制EC600N TE-A 选择通道（从1依次开始）
4. 控制负载机拉不同负载
5. 读取EC600N TE-A ADC数据
6. 生成测试报告



1. 案例演示

<演示如何使用前面的组件和流程，进行业务代码整合，形成一个可用的demo>