

《八年级第五单元 第 25 课-智能书包项目》学生学习手册

学校名称：_____ 班级：_____ 姓名：_____

模块一：课程导入

真实情境

你是否遇到过这样的烦恼：老师在群里发了作业通知，但爸爸妈妈忘记告诉你；或者，你用本子记下了作业，却不小心弄丢了本子。结果，第二天回到学校，发现自己是唯一一个没完成作业的“小迷糊”。这种感觉一定糟透了！

传统的通知方式，就像“断了线的风筝”，老师发出信息后，无法确定每个同学是否真的收到了。如果有一种方法，能让作业通知像“必达的魔法信件”一样，直接发送到你的书包里，并且你收到后还能给老师一个“已阅”的回执，那该多好啊！

核心问题

我们能否利用物联网技术，打造一个“智能书包”，实现作业的精准下发和状态的实时反馈，彻底告别“忘记作业”的烦恼？

学习目标

通过本课程的学习，你将能够：

1. 理解物联网在家校沟通领域的应用价值，体验“数据闭环”的魅力。
2. 掌握 MQTT 协议的核心概念：发布（Publish）与订阅（Subscribe）。
3. 学会使用 JSON 格式来组织和解析复杂的数据。
4. 掌握回调函数（Callback）这一重要的异步编程思想。
5. 设计并实现一个具备信息接收、解析、交互和反馈功能的物联网应用。

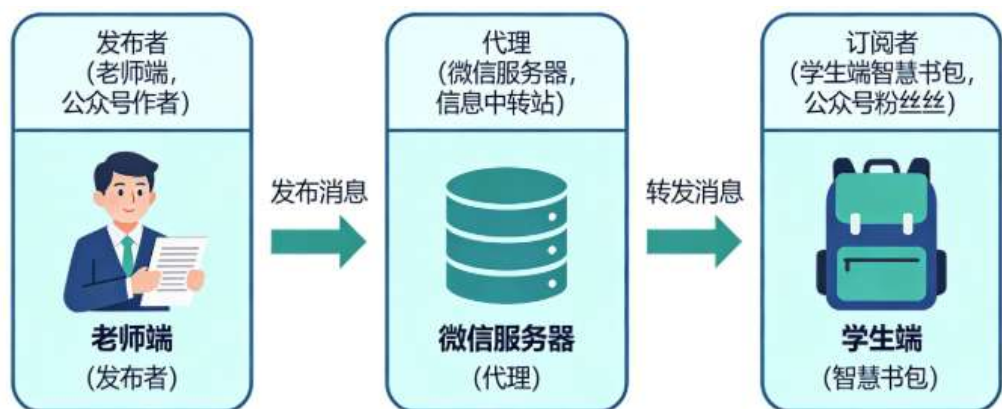
模块二：概念学习

1. 核心协议：MQTT

想一想：你订阅过微信公众号吗？当你订阅了一个公众号，它每次发布新文章，你都会自动收到推送。你并不需要每隔几分钟就去刷新一次公众号的主页。

MQTT (Message Queuing Telemetry Transport) 协议的工作方式与此非常相似，它就是物联网世界的“微信公众号系统”。

- **发布者 (Publisher):** 就像公众号的作者，负责发布信息。在我们的项目中，就是老师端。
- **订阅者 (Subscriber):** 就像公众号的粉丝，负责接收信息。在我们的项目中，就是学生端的“智能书包”。
- **代理 (Broker):** 就像微信的服务器，一个信息中转站。老师把作业信息发给它，它负责精准地推送给所有订阅了的学生。
- **主题 (Topic):** 就像公众号的名称。老师向特定的主题（如 class8-1/homework）发布作业，学生也必须订阅完全相同的“主题”，才能收到信息。这是一个“接头暗号”，确保信息不会发错班级。



主题: clas8-1/homework (公众号名称作为接头音号)

图 1: MQTT 工作原理示意图

2. 数据格式: JSON

想一想: 如果老师要一次性发布语文、数学、英语三科的作业，用一条消息发送，该怎么组织呢？

如果写成 完成语文练习册，做完数学卷子，背诵英语单词，我们的程序很难准确地把它们分开。万一语文作业里本身就包含了逗号，程序可能就会理解错误。

我们需要一种更清晰、更结构化的“清单”格式，**JSON (JavaScript Object Notation)** 就是为此而生的。它用大括号 `{}` 包裹所有内容，并使用“键: 值”对的形式来描述信息。

```
{
  "语文": "完成课本第 10 页练习",
  "数学": "完成练习册第 15-20 题",
  "英语": "背诵 Unit 3 单词"
}
```

这种格式一目了然，程序可以非常轻松地通过“键”（比如“数学”）来准确地获取对应的“值”（“完成练习册第 15-20 题”）。它就像一个贴着清晰标签的快递包裹，让你能快速找到想要的物品。

3. 编程思想：回调函数 (Callback)

想一想：你在家专心写作业时，如果门铃响了，你会怎么做？你肯定会停下手中的笔，跑去开门。你并不是每隔 10 秒就跑去门口看一下是否有人按门铃。

回调函数 (Callback) 就是程序世界的“门铃响应机制”。

- **传统方式（轮询）：**程序不断地主动去问服务器“有新消息吗？有新消息吗？”，这非常浪费资源，而且可能不及时。
- **回调方式：**程序告诉服务器：“我的‘门铃’（回调函数）在这里，如果有新消息（事件）来了，你就‘按门铃’（调用这个函数），我会自动去处理。”

设置好回调函数后，我们的主程序就可以安心地去做其他事情（比如检测按键），而不需要一直关心是否来消息了。当消息真的到达时，系统会自动“中断”一下，去执行我们预先定义好的回调函数，处理完消息后再回来继续做原来的事。这是一种更高效、更聪明的异步编程方式。

模块三：逻辑分析

1. 系统功能拆解

想一想：平时老师在学校是如何布置作业的呢？类比过来，我们的项目需要实现的基本功能模块包含哪些内容呢？和同学讨论讨论，一起设计吧！

系统部分	具体功能	实现方式
输入 (Input)	1. 接收老师发布的作业	1. 通过 Wi-Fi 和 MQTT 订阅 指定 Topic
	2. 接收学生的操作指令	2. 检测开发板上的 A/B/P/I 等按键是否被按下
处理 (Process)	1. 使用 回调函数 处理收到的 MQTT 消息	
	2. 使用 JSON 解析 技术，将消息字符串分解为结构化数据	
输出 (Output)	3. 根据按键，从解析后的数据中 提取 对应学科的作业内容	
	4. 构建一个包含“学生姓名+已确认”的 JSON 消息	
	5. 通过 MQTT 发布 确认消息到另一个 Topic	
	1. 在 OLED 屏幕 上显示收到的作业内容	
	2. 使用 LED 灯 或 蜂鸣器 提示有新作业到达	
	3. 老师在云平台或自己的客户端上看到学生的 确认回执	

2. 逻辑可视化：数据流图与状态图

数据是如何流动的？

画一画：请根据下面的描述，将数据流动的路径连接起来。

老师端 → 发布 JSON 作业 → MQTT 服务器 → 学生端 → 解析 JSON → 显示在屏幕上 → 学生按键确认 → 发布确认消息 → MQTT 服务器 → 老师端

你的数据流图：

智能书包有几种“状态”？

我们的智能书包在不同时间点会处于不同的工作状态。我们可以用状态图来清晰地描述这些状态以及它们之间是如何切换的。

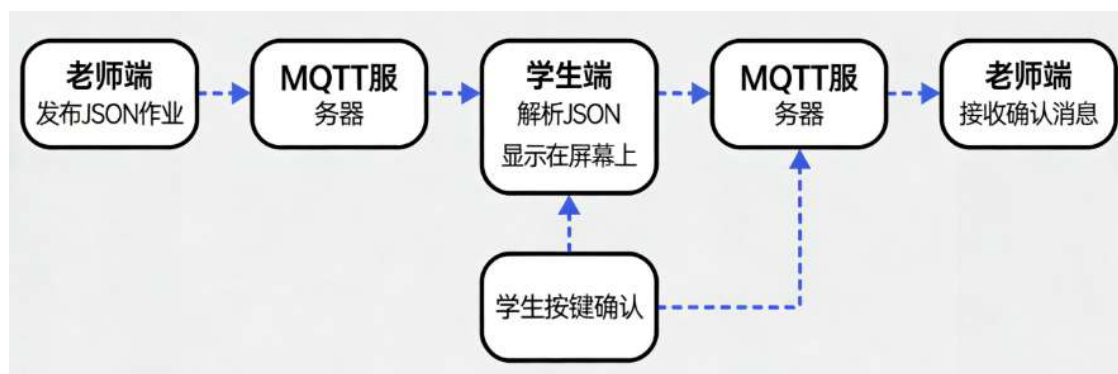


图2：智能书包工作状态图

看图说话：

1. 智能书包的初始状态是什么？
2. 当什么事件发生时，它会从“等待作业”状态切换到“收到新作业”状态？
3. 在“收到新作业”状态下，按“A1键”和按“B1键”会分别发生什么？
4. 如何才能让它切换到“已确认”状态？

模块四：算法设计

让我们用伪代码来梳理“智能书包”从接收到反馈的完整逻辑。

读一读，理一理：下面的伪代码描述了整个流程，请重点理解回调函数是如何工作的。

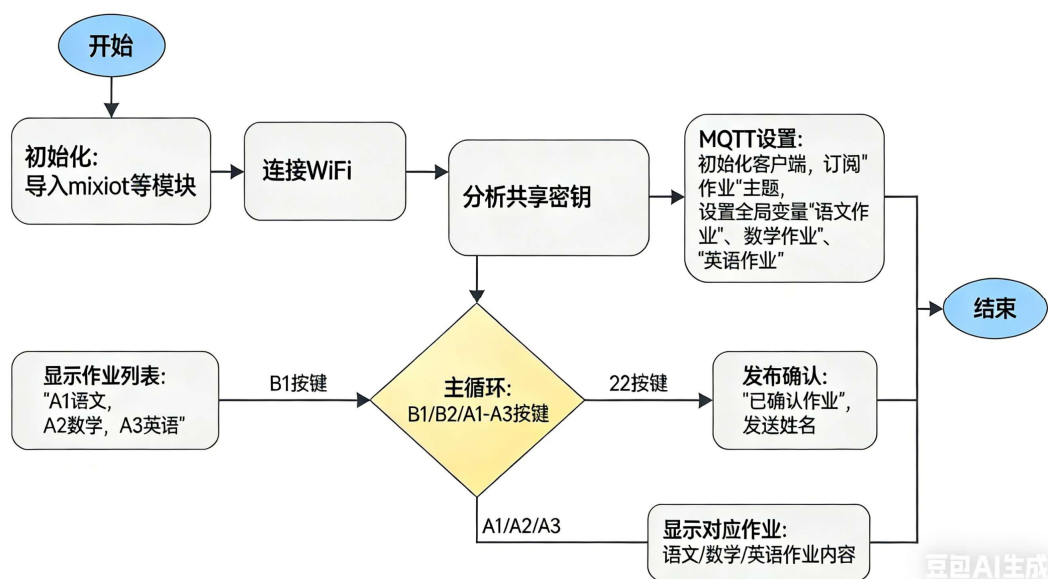


图3：代码实现流程图

模块五：代码实现

我们将通过四个核心任务，一步步构建出功能强大的“智能书包”。

任务 1：连接网络与云平台

目标：让你的开发板成功连接到 Wi-Fi 和 MQTT 服务器。



动手试一试：

1. 将你的 Wi-Fi 和 MQTT 账户信息填入代码中。

2. 修改并运行代码，使得 OLED 屏幕上依次显示“WiFi Connected!”和“MQTT Connected!”。
3. 如果任何一步连接失败，请仔细检查你填写的配置信息是否正确。

任务 2：订阅主题与回调函数

目标：订阅作业发布的 Topic，并设置一个回调函数，当收到消息时，能自动打印出来。

动手试一试：

1. 在任务 1 代码的基础上，增加回调函数的代码。
2. 运行程序，等待教师发布消息。
3. 观察你的开发板：OLED 屏幕是否显示了信息？蓝色 LED 灯是否亮起？电脑端是否打印出了你发送的消息？

任务 3：解析 JSON 并存储

目标：修改回调函数，使其能够解析收到的 JSON 字符串，并将内容存入字典中。



动手试一试：

1. 修改回调函数。
2. 运行程序。
3. 用 MQTT 工具发布一条 **JSON 格式** 的消息，例如：{"语文": "Page 10", "数学": "Page 20"}。
4. 观察电脑端的输出，是否成功打印出解析后的字典 {'语文': 'Page 10', '数学': 'Page 20'}？屏幕上是否显示了语文作业的内容？

任务 4：按键交互与信息反馈

目标：完善主循环，实现按键查询不同学科作业，并能发送确认回执。



动手试一试：

1. 将你的主循环替换成以上代码。
2. 运行完整程序，并从电脑端发送作业 JSON。
3. 收到作业后，尝试按 A1 键和到 A3 键，是否能切换显示不同科目的作业？
4. 按下 B2 键，是否有灯光提示信息发送过？同时，在你的 MQTT 客户端工具中，订阅 class8-1/feedback 主题，你是否能收到来自开发板的确认消息？

模块六：调试指导

物联网项目涉及硬件、网络、云平台等多个环节，调试起来会更有挑战性。学会系统性的调试方法，能让你事半功倍。

技巧 1：“分段测试”与“最小化原则”

一个复杂的物联网应用，千万不要想着一次性写完所有代码再运行。一定要遵循“分段测试”的原则。

1. **先测网络：**只写 Wi-Fi 连接的代码，确保能连上路由器。
2. **再测云平台：**只写 MQTT 连接的代码，确保能连上。

3. **然后测订阅：**写一个最简单的回调函数（只打印消息），确保能收到来自云平台的消息。
4. **接着测解析：**在回调函数里加入 JSON 解析，确保能正确处理数据格式。
5. **最后测交互：**加入按键和发布的逻辑，完成整个闭环。

如果在某一步出错了，就停下来只解决这一个问题。这种“最小化问题”的思路是最高效的调试方法。

技巧 2：利用好电脑端的“眼睛”——打印模块

在物联网项目中，打印模块是你最重要的调试工具，它能帮你看到那些在硬件上看不见的信息流。

- 打印收到的原始消息，可以帮你判断收到的消息格式是否正确。
- 打印解析后的字典，可以帮你确认 JSON 解析是否成功。
- 打印程序状态，可以让你清楚地知道程序执行到了哪一步。

技巧 3：使用专业的 MQTT 客户端工具

调试 MQTT 相关功能时，你不能只盯着开发板。你需要一个“上帝视角”来观察云平台上的消息流动，可以代开物联网平台查看数据流。

时间	数学	语文	英语
2026-01-30 14:52:14	练习册A	作文	背单词
2026-01-30 14:43:01	练习册A	作文	背单词
2026-01-30 14:41:56	练习册A	作文	背单词
2026-01-30 14:40:38	练习册A	作文	背单词

图 3：物联网平台数据展示图

你可以用它来做什么？

1. **模拟老师端：**手动向作业 Topic 发布各种格式的 JSON 消息，测试你的开发板在不同情况下的反应。
2. **模拟其他同学：**查看某个 Topic 下当前是否能够正常接收确认反馈。

3. **监听所有消息：**订阅，可以看到该服务器上所有公开的消息流动，帮你理解整个系统是如何工作的。

4. **检查回执：**订阅反馈 Topic（如 class8-1/feedback），验证你的开发板是否成功地将确认消息发了出来。

技巧 4：常见问题排查清单

错误现象	可能原因	解决方法
Wi-Fi 连接失败	1. ID 或密码错误 2. Wi-Fi 信号太弱或路由器非 2.4GHz 频段	1. 仔细检查大小写和特殊字符 2. 确保开发板靠近路由器，并使用 2.4GHz 的 Wi-Fi
MQTT 连接失败	1. 服务器地址、端口、用户名或密码错误 2. MQTT_CLIENT_ID 与其他设备冲突	1. 逐一核对 MQTT 配置信息 2. 尝试换一个更独特的客户端 ID，比如加上你的学号
收不到 MQTT 消息	1. 订阅的 Topic 与发布的 Topic 名称不一致 2. 回调函数没有被正确设置 3. 主循环中没有使用调用模块	1. 检查 Topic 的拼写，包括斜杠 2. 确保设置了回调 3. 确保主循环中有持续检查消息的机制
JSON 解析错误	1. 未进行反序列化 2. 存储数据的变量没有使用全局变量	1. 进行反序列化 2. 使用全局变量

模块七：拓展挑战

“智能书包”的潜力远不止于此。想要将它改造成一个连接学校和家庭的智能终端，我们可以为它赋予更多强大的功能。

挑战 1：更丰富的交互体验

- **多页面显示：**如果作业内容很长，一个屏幕显示不下怎么办？能否通过按键实现“翻页”功能？

- **历史作业查询：**能否将最近三次的作业都存起来，通过特定按键组合（如 A+B 键同时按下）来回看历史作业？
- **语音播报：**集成一个语音合成模块，当收到新作业时，能自动将作业内容读出来。

挑战 2：更智能的提醒系统

- **截止日期提醒：**在发布的 JSON 中增加一个“截止日期”字段。当离截止日期还剩 24 小时时，让书包的 LED 灯开始闪烁提醒。
- **个性化提醒：**不同的科目用不同颜色的 LED 灯来提示。例如，语文是蓝色，数学是红色，英语是绿色。
- **“未确认”自动重发：**老师端可以设计一个程序，如果在发布作业 1 小时后，仍未收到某个同学的确认回执，就自动向该同学再次发送提醒消息。

挑战 3：家校互动的更多可能

- **发布校园通知：**除了作业，老师还可以通过它发布放假通知、家长会通知等。
- **学生请假系统：**学生可以通过书包上的按键选择“事假”或“病假”，将请假申请发送给老师的客户端。
- **课堂匿名投票器：**老师提出问题，学生通过 A/B/C 键进行选择，并将投票结果匿名发送到云平台，老师可以实时看到全班的投票统计。

模块八：学习反思

1. 我的知识清单

请用自己的话，简要描述你对以下概念的理解：

- **MQTT (发布/订阅模型)：** _____
- **JSON (数据格式)：** _____
- **回调函数 (异步编程)：** _____
- **数据闭环：** _____

2. 我的“拦路虎”日志

我遇到的问题	我是如何解决的（或仍未解决）	我从中学到了什么
--------	----------------	----------

3. 我的“产品经理”初体验

在这个项目中，你不仅是一个程序员，更是一个“产品经理”。你需要思考用户的需求（老师和学生），设计产品的功能和流程。

请思考：如果让你来设计下一代的“智能书包”，你还会为它增加哪些功能来解决老师或同学的痛点？

从使用者到创造者，你已经迈出了最重要的一步。继续探索，用科技让学习变得更美好！