

Editorial Board

编委名单

ISSN: 3065-1220

<https://www.hanspub.org/journal/etis>

主编

何立民教授 北京航空航天大学

Editor-in-Chief

Prof. Limin He Beihang University

副主编

何小庆秘书长 嵌入式系统联谊会

Associate Editors

Allan He Secretary General of the Embedded Systems Association

吴薇特聘教授 杭州电子科技大学

Distinguished Prof. Wei Wu Hangzhou Dianzi University

名誉编委

王田苗教授 北京航空航天大学
严义教授 PLCopen China主席/杭州电子科技大学
邵贝贝教授 清华大学工程物理系

Honorary Chief Editor

Prof. Tianmiao Wang Beihang University
Prof. Yi Yan Hangzhou Dianzi University
Prof. Beibei Shao Department of Engineering Physics, Tsinghua University

编委会

马忠梅副教授 北京理工大学计算机学院
王朋朋系统工程 恩智浦(中国)管理有限公司
高级总监
牛建伟教授 北京航空航天大学
陈渝长特聘副教授 清华大学计算机系
张永进总经理 深圳拓普微科技开发公司

沈建华副教授 华东师范大学计算机学院
周立功创始人/ 广州致远电子股份公司
董事长
桑楠教授 电子科技大学信息与软件工程学院

袁涛副教授 清华大学自动化系
常晓明教授 太原理工大学
韩德强高级工程师 北京工业大学计算机学院
魏洪兴教授 北航机械工程及自动化学院
林金龙教授 北京大学软件与微电子学院
刘洪涛研发副总裁 华清远见教育科技集团
/研发中心总经理

Editorial Board

Prof. Zhongmei Ma Beijing Institute of Technology
Lucy Wang Senior Engineering Director of NXP China Management Ltd.

Prof. Jianwei Niu Beihang University
Prof. Yu Chen Tsinghua University
Yongjin Zhang General Manager of Shenzhen Topway Technology Ltd.

Prof. Jianhua Shen East China Normal University
Ligong Zhou Founder of Zhiyuan Electronics Ltd.

Prof. Nan Sang University of Electronic Science and Technology of China

Prof. Tao Yuan Tsinghua University
Prof. Xiaoming Chang Taiyuan University of Technology
Prof. Deqiang Han Beijing University of Technology
Prof. Hongxin Wei Beihang University
Prof. Jinlong Lin Peking University
Hongtao Liu Vice President of R&D of HQYJ Education Technology Group

TABLE OF CONTENTS

目 录

基于 Hi3863 的多模星闪网关系统设计 Design of a Multi-Mode SparkLink Gateway System Based on Hi3863	
安皓楠, 白浩洁	285
星闪技术: 一种确定性短距无线通信方案 NearLink: A Deterministic Short-Range Wireless Communication Solution	
张东阳, 张亚超, 侯兴晨	296
基于三层心理学模型的人格智能系统 Personality Intelligence System Based on a Three-Layer Psychological Model	
万学靖, 刘伟, 曹婷, 冯吭雨, 解运洲, 任付标, 过弋, 宋广奎	305
在 Zephyr RTOS 基础上构建兼容 AUTOSAR CP 的智能车控操作系统 Building AUTOSAR CP Compatible Vehicle Operating System Based on Zephyr RTOS	
付宝军, 任慰	319
基于提示工程的软件测试用例生成方法研究 Research on Software Test Case Generation Method Based on Prompt Engineering	
王媛, 韩启, 张絮, 杨丰辉	330

期刊信息

期刊中文名称:《嵌入式技术与智能系统》

期刊英文名称: **Embedded Technology and Intelligent Systems**

期刊缩写: **ETIS**

出刊周期: 双月刊

语 种: 中文

出版机构: 汉斯出版社(Hans Publishers, <https://www.hanspub.org/>)

编辑单位:《嵌入式技术与智能系统》编辑部

主 编: 何立民, 北京航空航天大学教授

网 址: <https://www.hanspub.org/journal/etis>

订阅信息

订阅邮箱: sub@hanspub.org

订阅价格: 180 美元每年

广告服务

联系邮箱: adv@hanspub.org

版权所有: 汉斯出版社(Hans Publishers)

Copyright©2025 Hans Publishers, Inc.

版权声明

文章版权和重复使用权说明

本期刊版权由汉斯出版社所有。

本期刊文章已获得知识共享署名国际组织(Creative Commons Attribution International License)的认证许可。

<https://creativecommons.org/licenses/by/4.0/>

单篇文章版权说明

文章版权由文章作者与汉斯出版社所有。

单篇文章重复使用权说明

注: 著作权者准许任选 CC BY 或 CC BY-NC 作为文章的重复使用权, 请慎重考虑。

权责声明

期刊所刊载的评论、意见、观点等均出自文章作者个人立场, 不代表本出版社的观点或看法。对于文章任何部分及文内引用材料给任何个人、机构、及其财产所带来的任何损失及伤害, 本出版社均不承担任何责任。我们郑重声明, 本出版社的出版业务, 不构成对任何产品商业性能的保证, 也不表示本社业已承认本社出版物中所述内容适用于某特定用途。如有疑问, 请寻找专业人士协助。

基于Hi3863的多模星闪网关系统设计

安皓楠, 白浩浩

北京华清远见科技发展有限公司, 研发中心, 北京

收稿日期: 2025年11月12日; 录用日期: 2026年1月16日; 发布日期: 2026年1月29日

摘要

随着物联网技术发展, 智能家居、工业互联等场景对无线通信提出低延迟、高可靠、多设备连接和低功耗的要求。星闪(SLE)作为我国自主研发的短距无线技术, 为物联网碎片化连接提供了解决方案。本文基于海思Hi3863芯片设计多模星闪网关系统, 实现星闪与Wi-Fi异构网络的互联, 构建边缘数据枢纽。本文首先分析了多模网关的需求, 提出了以Hi3863为主控、集成了星闪与Wi-Fi通信模组的硬件架构。系统采用分层软件设计, 底层基于LiteOS操作系统抽象层进行任务调度与资源管理; 中间层分别实现了星闪协议栈的设备发现、连接管理、数据传输功能, 以及基于LWIP的Wi-Fi TCP/IP通信功能; 应用层核心设计了一个智能数据转发引擎, 通过解析数据包中的网络与设备标识, 实现星闪网络与Wi-Fi网络间数据的无缝转换与可靠转发。测试结果表明, 本系统能够稳定维护多个星闪设备与Wi-Fi客户端的并发连接, 成功实现了跨异构网络的数据透传, 有效验证了基于Hi3863构建低成本、高性能多模星闪网关系统的可行性。

关键词

Hi3863, 星闪, 多模网关, 物联网, 异构网络

Design of a Multi-Mode SparkLink Gateway System Based on Hi3863

Haonan An, Haojie Bai

Research and Development Center, Beijing Huaqingyuanjian Technology Development Co., Ltd., Beijing

Received: November 12, 2025; accepted: January 16, 2026; published: January 29, 2026

Abstract

With the advancement of IoT technology, applications such as smart homes and industrial interconnection require wireless communications to meet demands for low latency, high reliability, multi-device connectivity, and low power consumption. As a short-range wireless technology independently developed in China, SparkLink provides an effective solution to fragmented connectivity in IoT. This

paper designs a multi-mode SparkLink gateway system based on HiSilicon's Hi3863 chip, enabling interconnection between SparkLink and Wi-Fi heterogeneous networks to establish an edge data hub. The study begins by analyzing the requirements of multi-mode gateways and proposes a hardware architecture with Hi3863 as the main controller, integrating both SparkLink and Wi-Fi communication modules. The system adopts a layered software design: the underlying layer utilizes the LiteOS operating system abstraction layer for task scheduling and resource management; the middle layer implements SparkLink protocol stack functions including device discovery, connection management, and data transmission, as well as Wi-Fi TCP/IP communication based on LWIP; the core application layer features an intelligent data forwarding engine that enables seamless and reliable data transmission between SparkLink and Wi-Fi networks by parsing network and device identifiers in data packets. Test results demonstrate that the system can stably maintain concurrent connections with multiple SparkLink devices and Wi-Fi clients, successfully achieving cross-heterogeneous-network data transparent transmission. This effectively validates the feasibility of constructing a low-cost, high-performance multi-mode SparkLink gateway system based on Hi3863.

Keywords

Hi3863, SparkLink, Multi-Mode Gateway, Internet of Things (IoT), Heterogeneous Network

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

随着物联网技术在智能家居、工业互联等领域的深化应用,场景对无线通信技术提出了低延迟、高可靠、多设备连接与低功耗的更高要求[1]。星闪作为我国自主研发的新一代短距无线通信技术,通过融合创新,在关键性能指标上超越了传统蓝牙与 Wi-Fi,为破解物联网碎片化连接难题提供了优质的底层技术方案。然而,一项新技术的规模化部署有赖于成熟的生态系统,其中,能够桥接新兴网络与现有网络的网关设备至关重要。

在此背景下,本研究基于海思 Hi3863 芯片设计多模星闪网关系统。Hi3863 芯片集成了星闪与 Wi-Fi 通信能力,具备高性能处理核心与丰富外设接口,为构建紧凑型、低成本的边缘数据枢纽提供了理想的硬件平台。本研究的核心意义在于,通过设计并实现一个星闪与 Wi-Fi 异构网络融合的网关,旨在打通两种网络的通信壁垒,构建高效的边缘数据枢纽。这能够满足智能家居场景下对海量传感器数据本地汇聚与响应的需求,对推动星闪技术的产业应用与构建自主可控的物联网体系具有重要的实践价值。

本研究的主要内容包括基于 Hi3863 的星闪-Wi-Fi 双模网关的软硬件一体化设计与实现,其核心创新点体现在异构网络数据转发机制上。

本研究首先完成了以 Hi3863 为主控的硬件平台设计,并采用了分层的软件架构。底层基于操作系统抽象层进行任务与资源调度;中间层并行实现了完整的星闪协议栈(涵盖设备发现、连接管理与数据传输)与基于 LWIP 的 Wi-Fi TCP/IP 通信功能;应用层则集成了智能数据转发引擎等核心逻辑。

本研究提出并实现了一套高效的异构网络数据转发机制。该机制首先定义了网关内部的统一数据模型,实现了星闪与 Wi-Fi 两种异构网络协议间的无缝转换。进而,设计了一种基于标识解析的智能路由算法:数据转发引擎通过解析数据包中预置的网络类型标识与设备类型标识,动态查询设备连接状态表,从而精准地将数据路由至目标星闪连接或 Wi-Fi Socket。该机制有效解决了异构网络间的通信壁垒,确保了数据在双模网络间可靠、低延迟的透传,为多模网关的核心功能实现了关键技术支撑。

2. 系统架构设计

2.1. 网络架构

本系统旨在设计并实现一个以海思 Hi3863 为核心的多模无线网关。该网关作为异构网络融合的关键，主要承担两大功能：一是实现星闪 SLE 与 Wi-Fi 两种异构无线网络间的协议转换与数据转发；二是作为边缘节点，通过串口与上层主控 MCU 进行数据交互，实现更复杂的业务逻辑或接入更广阔的网络。系统采用分层与模块化的设计思想，确保了功能的清晰性、稳定性与可扩展性。

本系统采用以 Hi3863 网关节点为中心的星型网络拓扑，如图 1 所示。该拓扑结构简单、可靠，易于管理，非常适合物联网中设备与网关的通信模型。

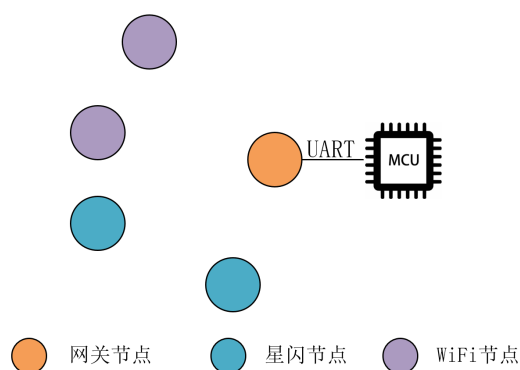


Figure 1. Network architecture

图 1. 网络架构

Hi3863 多模网关节点是整个网络的唯一核心。

星闪 SLE 节点为各类基于星闪技术的传感器(如温湿度、门磁)或执行器(如智能开关)。它们直接与 Hi3863 网关建立星闪 SLE 连接，形成一个独立的星闪 SLE 子网。

Wi-Fi 节点为各类基于 Wi-Fi 技术的传感器或执行器。它们直接与 Hi3863 网关建立的 Wi-Fi Socket 连接进行通信，形成一个 Wi-Fi 子网。

Hi3863 网关通过 UART 串口与一个功能更强大的上位主控 MCU 相连。此设计将复杂的多模无线通信功能与高级应用处理解耦，网关专注于高效、低延迟的数据汇聚与转发，而主控 MCU 则负责数据处理、决策、显示或通过无线网络/有线网络等方式上传至云端。

该星型拓扑实现了所有通信都必须经由网关中心节点，为集中式的设备管理、数据路由和网络监控提供了天然的便利。

2.2. 硬件构成

本系统的硬件围绕 Hi3863 主控芯片构建。Hi3863 系列是 2.4GHz Wi-Fi 6 星闪多模 IoT SoC 芯片[2]，采用高度集成的设计理念，实现了星闪与 Wi-Fi 双模通信的硬件基础。其集成 IEEE 802.11 b/g/n/ax 基带和 RF 电路，包括功率放大器 PA、低噪声放大器 LNA、RF balun、天线开关以及电源管理模块等；支持 20 MHz 频宽，提供最大 114.7 Mbps 物理层速率，支持更大的发射功率和更远的覆盖距离；支持星闪 SLE 1 MHz/2 MHz/4 MHz 频宽、SLE1.0 协议、支持 SLE 网关功能，最大空口速率 12 Mbps。

该芯片集成了高性能的 32 位 RISC-V 架构处理器核心，运行频率最高可达 260 MHz，内置 606 KB SRAM 和 300 KB ROM，为复杂的协议栈运行和数据转发处理提供了充足的计算资源和存储空间。

该芯片包含丰富的外设接口，包含 SPI、I2C、I2S、UART、GPIO 等，方便与传感器和其他芯片进

行交互。

在实际的部署过程中, 网关节点放置在华清远见的网关底板上。该底板包含无线模块接口和一个 STM32F103, 如图 2 所示。



Figure 2. Gateway hardware
图 2. 网关硬件

星闪节点和 Wi-Fi 节点放置在节点底板上。该节点底板可插接多种通信模组和多种传感器。

2.3. 软件架构

本系统的软件架构采用分层设计理念, 构建了一个模块化、可扩展的软件体系。如图 3 所示, 整个软件栈自底向上分为驱动层、操作系统抽象层、协议栈层和应用层, 各层之间通过清晰的接口进行通信, 确保了系统的稳定性和可维护性。

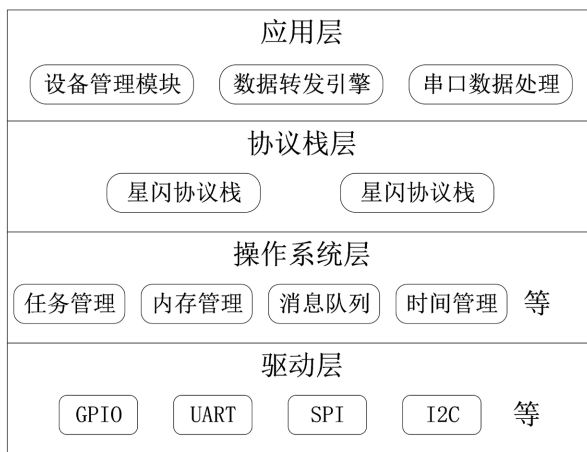


Figure 3. Network architecture
图 3. 软件架构

驱动层直接与硬件交互, 为上层提供统一的硬件操作接口。该层包含了多个关键外设的接口, 如 GPIO、UART、SPI、I2C 等, 可获取传感器数据、控制执行器。

操作系统层内部采用 OpenHarmony 架构下的 LiteOS 操作系统, 并将其封装成 OSAL 接口。该层通过封装实时操作系统的核心功能, 为上层应用提供了与具体操作系统内核无关的统一编程接口, 有效增强了软件的可移植性与可维护性。

协议栈层是双模通信能力的实现核心, 并行运行着星闪与 Wi-Fi 两套独立的协议栈。星闪协议栈完整实现了星闪技术的设备发现、连接建立与数据收发功能[3]。Wi-Fi 协议栈基于轻量级 IP 协议栈 LWIP [4], 实现了完整的 IP、TCP、UDP 等网络协议, 使网关能够接入本地局域网。网关作为 TCP 服务器, 创建监

听 Socket 并接受 Wi-Fi 客户端的连接请求。为每个连接的客户端维护一个 Socket 描述符, 并通过标准的 Socket API (如 send、recv) 进行数据收发。

应用层是系统业务逻辑的核心, 它整合下层服务, 实现了网关的智能数据路由与设备管理功能。设备管理模块负责维护两个核心数据结构——星闪设备连接表(sle_conn_device)和 Wi-Fi 设备连接表(wifi_conn_device)。数据转发引擎作为网关的智能中枢, 通过解析数据包中的网络与设备标识, 动态查询连接表并调用相应协议栈 API, 实现了星闪与 Wi-Fi 异构网络间的无缝数据透传。串口命令处理模块定义并实现了一套与上位主控 MCU 的通信协议。该模块解析来自串口的控制指令, 并将其转换为对网关内部模块的操作; 同时, 也将网关的状态信息(如连接设备数、网络异常等)封装成数据帧, 上报给上位 MCU。

3. 软件系统设计与实现

3.1. Wi-Fi AP 初始化及数据接收任务

Wi-Fi AP 初始化及数据接收任务负责启动设备的无线热点功能并建立 TCP 服务器。该任务首先等待 Wi-Fi 硬件初始化完成, 然后配置并启用 SoftAP 模式, 设置热点的 SSID、密码、IP 地址等网络参数, 同时创建 TCP Socket 进行端口绑定和监听。在任务运行期间, 它会创建专门的 TCP 连接接受任务来处理客户端连接请求, 并持续轮询所有已连接的 Wi-Fi 客户端, 接收它们发送的数据并存入缓冲区, 为后续的数据转发处理提供支持。

主要流程:

```
while (wifi_is_wifi_initied() == 0) // 等待Wi-Fi初始化完成
example_softap_function() // 启动SoftAP和TCP服务器
创建 TCP_accept_Task 任务
while (1) {
    for(所有Wi-Fi客户端) {
        wifi_device_recv() // 轮询接收数据
        osal_mdelay(100)
    }
}
```

3.2. TCP 连接接受任务

TCP_accept_Task 作为 TCP 连接接受任务, 在 Wi-Fi AP 网络环境中承担着客户端连接管理的核心职责。该任务持续监听服务器端口, 当检测到新的连接请求时, 通过系统调用接受连接并为每个客户端动态分配设备管理结构体, 记录其套接字描述符和网络地址信息。同时, 该任务维护着全局连接计数和设备状态跟踪, 确保系统能够同时处理多个客户端的并发访问, 为后续的数据通信建立稳定的网络链路基础。

主要流程:

```
while (1) {
    dev_id = FindFreeDeviceIndex() // 查找空闲设备槽位
    if (dev_id != -1) {
        new_fd = accept(sock_fd, ...) // 接受新连接
        if (new_fd != -1) {
            初始化 wifi_conn_device[dev_id] // 记录连接信息
            g_wifi_conn_num++ // 连接计数增加
        }
    }
    osDelay(100) // 延时等待
}
```

3.3. 星闪设备管理任务

星闪设备管理任务在异构网络架构中负责实现星闪协议的设备发现、连接建立与数据路由功能。该任务在系统启动后首先完成硬件初始化与协议栈配置，随后进入持续监听状态，通过消息队列机制接收来自串口的外设数据。在任务核心处理逻辑中，它解析数据包中的网络标识与设备类型信息，动态匹配已连接的星闪终端设备，并基于 SSAPC 协议将数据准确转发至目标设备。同时，该任务与设备扫描模块协同工作，维护着星闪设备的连接状态表，确保数据传输的可靠性与实时性。

主要流程：

```

osal_msleep(5000)           // 等待Wi-Fi初始化完成
app_uart_init_config()      // 初始化UART
bsp_init()                  // 初始化硬件(LCD, GPIO等)
sle_client_init()           // 初始化星闪协议栈

while (1) {
    osal_msg_queue_read_copy() // 从消息队列读取数据
    sle_wifi_send_data(msg_data) // 转发数据到SLE/Wi-Fi设备
}
    
```

sle_client_init()函数是星闪通信系统的核心初始化模块，主要完成星闪协议栈的完整启动与回调机制注册。该函数内部涉及的接口如图 4 所示。

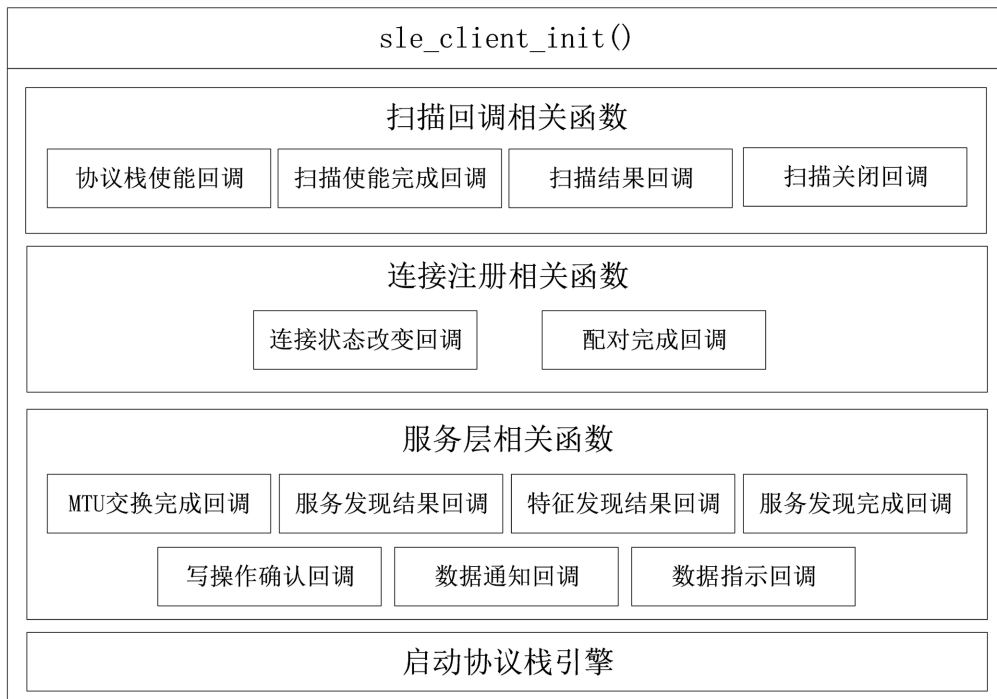


Figure 4. SLE protocol stack related callback functions
图 4. SLE 协议栈相关回调函数

该函数首先注册设备扫描相关的回调函数，包括协议栈使能通知、扫描启停状态反馈及设备发现结果处理，为后续的设备发现提供事件响应基础。接着建立连接管理回调，实时监控设备连接状态变化与配对完成事件，动态维护设备连接表。最后配置服务层通信回调，实现 MTU 协商、服务发现、特征值读

写及数据通知等高级通信功能的异步处理。在完成所有回调注册后, 函数调用 `enable_sle()` 正式启动星闪协议栈, 并将网络状态指示灯设置为就绪状态, 标志着星闪通信系统已全面进入可操作状态。

数据通知回调函数作为星闪协议栈的数据接收通知回调, 在系统中负责处理来自星闪设备的异步数据上报。当远程设备通过通知机制发送数据时, 该函数将接收到的数据安全拷贝至专用的星闪串口缓冲区, 并更新缓冲区的写入位置索引。最后, 它会调整缓冲区管理指针并为后续数据包预留存储空间, 从而确保连续数据流的可靠接收与处理, 为上层应用提供了完整的数据帧。

3.4. 数据发送处理任务

数据发送处理任务在通信系统中承担着数据汇聚与转发的关键职能。该任务以固定周期轮询检查星闪与 Wi-Fi 两个网络通道的接收缓冲区, 当检测到有效数据包时, 按照预定义的通信协议进行格式解析与校验, 并通过串口接口将整合后的数据发送至外部处理单元。其设计采用双缓冲区管理机制, 能够并行处理多路数据流, 确保来自不同网络协议的传感器数据能够实时、有序地传输, 为上层应用提供统一的数据接入服务。

主要流程:

```
osal_msleep(5000)           // 等待Wi-Fi初始化完成
while (1) {
    DataHandler(&Usart_SLE_Control, 0) // 处理SLE接收数据
    DataHandler(&Usart_WIFI_Control, 0) // 处理Wi-Fi接收数据
    osal_mdelay(50)           // 50ms周期处理
}
```

3.5. 串口数据接收回调函数

由于网关需要通过 UART 接收来自外部处理单元, 因此需要使用串口接收回调函数接收来自上位机的数据。其负责将接收到的原始数据安全地封装并传递到消息队列中。

3.6. 数据缓存机制

为高效管理来自星闪 SLE 与 Wi-Fi 异构网络的多路异步数据流, 并应对数据接收与处理速度不匹配的问题, 本系统设计并实现了一套基于环形指针队列的双缓冲区数据缓存机制。该机制确保了在高并发数据流入时的系统稳定性, 并提供了高效、可靠的数据存取路径。

3.6.1. 缓存结构设计

系统为星闪 SLE 和 Wi-Fi 分别独立定义了完全相同的缓存管理结构, 实现了数据隔离与并行处理。核心数据结构包括:

数据缓冲队列: 一个静态分配的字节数组, 作为存储原始数据的物理内存区域。

指针缓冲队列: 一个 `USART_Control` 类型的结构体, 用于管理缓冲区的读写状态。

```
typedef struct
{
    uint32_t RxBufferCounter; // 指示下一个字节应存入数据缓冲队列的哪个位置
    LocationPtr RxLocationPtr[LocationPtrNum]; // 管理指针缓冲队列的结构体数组
    LocationPtr *RxDataInPtr; // In指针: 标记指针缓冲队列插入数据的位置
    LocationPtr *RxDataOutPtr; // Out指针: 标记指针缓冲队列提取数据的位置
    LocationPtr *RxDataEndPtr; // End指针: 标记指针缓冲队列的末尾, 防止数组越界
} USART_Control;
```

其中 LocationPtr 类型如下所示。

```
typedef struct
{
    uint8_t *StartPtr; // 帧起始指针
    uint8_t *EndPtr;   // 帧结束指针
} LocationPtr;
```

该缓存结构如图 5 所示。

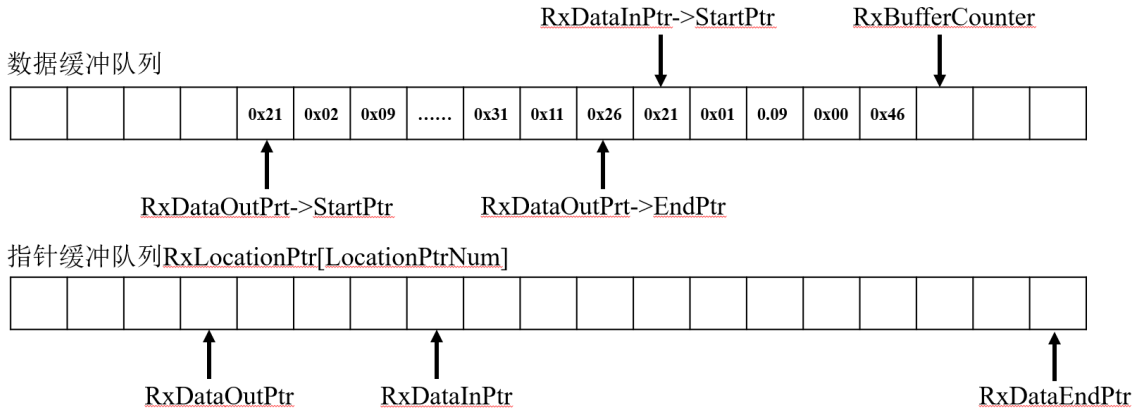


Figure 5. Buffer structure
图 5. 缓存结构

在系统初始化阶段, 对上述结构体进行初始化, 建立缓冲区与指针队列之间的映射关系, 并将读写指针均指向数据缓冲队列起始位置。

3.6.2. 数据接收与缓存流程

当串口空闲中断触发, 表明一个完整数据包接收完毕时, 系统会调用相应的接收完成回调函数。

封包入队: 回调函数首先将当前数据包在指针缓冲队列的结束地址写入 RxDataInPtr 的 EndPtr 指针所指向的位置, 从而完成一个数据包的封包操作。然后让 RxDataInPtr 指针往后移动一个 LocationPtr 数据结构长度。

移动写入指针: 随后, RxDataInPtr 的 StartPtr 指针移动到之前数据包结束位置 EndPtr 指针的后面, 指向下一个空闲数据元素。

队列循环处理:

情况一: 若移动后的 RxDataInPtr 已到达队列末尾, 则将其重置指向队列头部, 实现环形队列的循环使用。

情况二: 函数检查物理缓冲区的剩余空间。若剩余空间足以容纳下一个最大数据包, 则设置下一个数据包的起始地址为当前写入偏移; 若剩余空间不足, 则将写入偏移复位至缓冲区头部, 从头开始循环利用缓冲区, 防止数据覆盖紊乱。

3.6.3. 数据处理与转发流程

数据处理任务 DataHandler 被周期性地调度执行, 其负责检查并处理缓存中的数据包。

数据包检查: 任务通过比较 RxDataInPtr 和 RxDataOutPtr 判断是否有未处理的数据包。通过辅助函数 get_untreated_datapacket 可精确计算未处理数据包的数量。

拥塞处理模式: 当未处理数据包数量超过设定的阈值时, 系统进入拥塞处理模式。在此模式下, 任务将快速连续地处理队列中的大量积压数据, 直至清空队列, 以尽快恢复系统正常响应。

常规处理模式: 在正常负载下, 任务每次执行仅处理一个数据包。它首先验证数据包的协议头标识, 若为有效数据, 则通过相应的串口发送出去; 若为无效数据, 则直接丢弃。

移动读取指针: 无论数据有效与否, 在处理完一个数据包后, RxDataOutPrt 指针都会向后移动一位。同样, 当指针到达队列末尾时, 会重置到队列头部。

该双缓冲区机制将数据的高速接收与相对低速的处理解耦开来, 通过异步方式避免了数据丢失, 确保了在多模网关复杂通信环境下数据转发的完整性与可靠性。

此流程实现了数据包在逻辑队列中的有序存入, 并通过物理缓冲区的循环使用, 最大限度地利用了有限的内存资源。

3.7. 协议格式

为确保异构网络(星闪与 Wi-Fi)中数据帧的统一处理与可靠传输, 本系统设计了一套精简且高效的应用层通信协议。该协议定义了数据帧的统一格式, 使得网关能够正确解析来自不同网络节点的数据, 并根据协议中的字段进行路由和决策。节点与网关之间数据传输的格式如表 1 所示。

Table 1. Protocol format

表 1. 协议格式

标识	数据长度	数据偏移	数据类型	网络类型	设备地址	传感器类型	数据域	CRC8 校验和
1byte	1byte	1byte	1byte	1byte	2byte	1byte	...	1~255byte	1byte

各字段的详细定义与功能如下:

标识(Header): 固定为 0x21, 用于标识一个数据帧的开始。

数据长度(Data Length): 指示数据域的实际字节数。例如, 当数据域包含 1 字节数据时, 该字段值为 0x01。

数据偏移(Data Offset): 定义了从数据包起始位置(包头)到数据域开始位置的字节偏移量。该设计为协议头部保留了可扩展性, 未来若需增加新的固定字段, 只需调整偏移量即可保持兼容。

数据类型(Data Type): 用于区分数据包的功能属性。例如, 0x00 代表承载传感器读数的普通数据包, 而 0x01 代表用于设备配置、命令下发的控制命令包。

网络类型(Network Type): 标识数据包来源或目标节点的网络类型。该字段是网关实现双模数据路由的核心依据, 网关根据此值决定将数据通过星闪(SLE)或 Wi-Fi 网络转发。

设备地址(Device Address): 一个 2 字节的唯一标识符, 用于区分同一网络内的不同终端设备, 遵循传感器类型表中的预定义地址, 传输顺序为高位在前, 低位在后(Big-Endian)。

传感器类型(Sensor Type): 参照传感器类型表, 标识设备所搭载的传感器种类(如温湿度、光照强度等), 便于网关对数据进行分类处理与应用。

数据域(Data Field): 承载有效载荷的核心部分, 长度可变(1~255 字节), 内容可为传感器采集的实时数据或控制命令的具体参数。

CRC8 校验和(CRC8 Checksum): 用于保障数据传输的完整性。校验范围覆盖从包头至数据域结尾的所有字节, 采用生成多项式 $x^8 + x^2 + x + 1$ (CRC-8/MAXIM)进行计算。接收方通过验证校验和来判断数据包在传输过程中是否出错。

4. 实验部署

为验证本系统的功能与性能, 我们搭建了完整的实验环境, 对多模网关的通信能力、数据转发效率及系统稳定性进行了测试。

实验环境包括:

硬件平台: Hi3863 网关、星闪传感器节点(温湿度、光照)、Wi-Fi 传感器节点、STM32F103 上位主控板。

软件环境: LiteOS 操作系统、星闪协议栈 V1.0、LWIP 2.1.2。

网络拓扑: 采用星型结构, 网关同时接入 8 个星闪设备与 4 个 Wi-Fi 客户端, 主控 MCU 通过 UART 与网关交互。

通过实验部署验证了以下内容:

双模接入能力: 在 30 米室内距离内, 网关成功同时建立星闪与 Wi-Fi 连接, 设备识别率 100%, 星闪的连接建立时间小于 2 秒。

数据透传测试: 在 30 米室内距离内, 星闪传感器数据经网关转发至外部处理单元, 反向控制指令亦可准确送达星闪执行器, 双向传输成功率 100%。

协议解析与路由: 网关正确解析自定义协议中的网络类型与设备地址字段, 实现跨网络数据路由, 未出现路由错误或数据丢失。

实验结果表明, 本系统在软硬件协同设计下, 实现了星闪与 Wi-Fi 异构网络的高效融合。网关具备良好的实时性、可靠性与扩展性, 可满足智能家居等场景中对多协议接入与边缘数据汇聚的需求。基于 Hi3863 的解决方案在成本与性能之间取得了良好平衡, 具备规模化部署的潜力。

5. 未来工作

尽管本研究成功设计并实现了一个基于 Hi3863 的多模星闪网关原型系统, 验证了星闪与 Wi-Fi 异构网络融合的基本可行性, 但无论从系统本身的功能完善还是从技术发展的前沿趋势来看, 仍有诸多值得深入研究和改进的方向。未来的工作将主要围绕以下几个方面展开:

(1) 协议栈增强与云平台深度融合

当前系统实现了边缘侧的数据透传, 未来将重点增强其与云平台的交互能力。计划移植轻量级的 JSON 解析库(如 cJSON), 对传输数据进行标准化封装, 实现 MQTT/HTTPs 等物联网标准协议与云端服务的无缝对接。这将使网关不仅能进行本地数据路由, 还能作为真正的边缘代理, 实现云端指令的下发与设备状态的远程监控, 构建“端-边-云”一体化的完整解决方案[5]。

(2) 低功耗优化策略研究

低功耗是许多物联网场景的核心诉求。未来工作将深入研究 Hi3863 芯片及星闪协议栈的低功耗特性。通过软件层面优化任务调度策略, 引入动态电源管理, 在无数据传输时使系统进入休眠模式; 在硬件层面, 可设计专门的电源管理电路。同时, 也将评估在低功耗 Wi-Fi 模式下运行 MQTT 等长连接协议的可行性与性能表现, 以期大幅提升网关及终端节点在电池供电下的续航能力。

参考文献

- [1] Vanani, K., Patoliya, J. and Patel, H. (2016) A Survey: Embedded World around MQTT Protocol for IoT Application. *International Journal for Scientific Research & Development*, 4, 26-29.
- [2] 海思官网. Hi3863V100 产品简介[EB/OL]. <https://www.hisilicon.com/cn/products/connectivity/short-range-IoT/wifi-nearlink-ble/Hi3863V100>, 2025-11-01.

- [3] Gao, M., Wan, L., Shen, R., *et al.* (2023) SparkLink: A Short-Range Wireless Communication Protocol with Ultra-Low Latency and Ultra-High Reliability. *The Innovation*, 4, Article 100386. <https://doi.org/10.1016/j.xinn.2023.100386>
- [4] Dunkels, A. (2001) Design and Implementation of the LWIP TCP/IP Stack. Swedish Institute of Computer Science.
- [5] 柴天佑, 程思宇, 李平, 贾瑶, 郑锐. 端边云协同的复杂工业过程运行控制智能系统[J]. 控制与决策, 2023, 38(8): 2051-2062.

星闪技术：一种确定性短距无线通信方案

张东阳*, 张亚超, 侯兴晨

沈阳理工大学装备工程学院, 辽宁 沈阳

收稿日期: 2025年11月17日; 录用日期: 2026年1月16日; 发布日期: 2026年1月29日

摘要

为解决传统短距无线技术在智能汽车、工业控制等场景下面临的高时延与低可靠性问题,即“确定性鸿沟”,本文对新一代星闪(NearLink)无线通信技术进行了系统性分析。文章首先剖析了星闪为实现确定性通信所采用的核心技术,包括其集中式调度机制、SLB (SparkLink Basic)/SLE (SparkLink Low Energy)双模接入架构以及集成了Polar码和HARQ (Hybrid Automatic Repeat reQuest)机制的物理层设计。分析结果表明,星闪通过上述设计,在关键性能上取得了显著提升,其空口时延低于20 μ s,传输可靠性高于99.999%,峰值速率可达920 Mbps。目前,星闪技术已在消费电子、汽车电子、工业控制等领域得到初步商业应用,相关产业链与生态系统正在构建中。本文认为,星闪技术为需要精密同步和高可靠性交互的应用场景提供了有效的解决方案,标志着短距无线通信正从“尽力而为”的模式向着提供可预测服务的方向发展,具备较好的应用前景。

关键词

星闪, 短距无线通信, 确定性通信, 低时延

NearLink: A Deterministic Short-Range Wireless Communication Solution

Dongyang Zhang*, Yachao Zhang, Xingchen Hou

School of Equipment Engineering, Shenyang Ligong University, Shenyang Liaoning

Received: November 17, 2025; accepted: January 16, 2026; published: January 29, 2026

Abstract

To address the high latency and low reliability issues faced by traditional short-range wireless technologies in scenarios such as smart cars and industrial control, namely the “determinism gap”, this

*通讯作者。

article systematically analyzes the new generation of NearLink wireless communication technology. The article first analyzes the core technologies used by NearLink to achieve deterministic communication, including its centralized scheduling mechanism, SLB (SparkLink Basic)/SLE (SparkLink Low Energy) dual-mode access architecture, and a physical layer design that integrates Polar codes and HARQ (Hybrid Automatic Repeat reQuest) mechanisms. The analysis results show that through the above design, NearLink has achieved significant improvements in key performance, with an air interface latency of less than 20 μ s, a transmission reliability of more than 99.999%, and a peak rate of up to 920 Mbps. Currently, NearLink technology has been initially commercially applied in consumer electronics, automotive electronics, industrial control and other fields, and the relevant industry chain and ecosystem are being built. This article believes that NearLink technology provides an effective solution for application scenarios that require precise synchronization and high-reliability interaction, indicating that short-range wireless communications are evolving from a “best effort” model to providing predictable services, and have good application prospects.

Keywords

NearLink, Short-Range Wireless Communication, Deterministic Communication, Low Latency

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

随着物联网、车联网及工业互联网的发展,对无线通信的要求已从提供基础连接,转变为保障高效、稳定与确定性的数据交互。当前,主流的短距无线技术,如 Wi-Fi 与蓝牙,普遍采用基于竞争的信道接入机制[1]。这种“尽力而为”的传输模式在高密度、高实时的应用场景下,难以提供可预测的时延与可靠性保障,形成了“无线确定性鸿沟”[2]。尤其在要求微秒级同步的工业控制或车载系统中,现有技术的毫秒级时延已成为其应用的主要制约因素。

为应对此挑战,新一代短距无线通信技术——星闪(NearLink)技术应运而生。该技术通过对底层通信架构进行系统性重构,旨在从根本上解决无线传输的随机性问题。星闪的核心目标是实现低空口时延与高传输可靠性,推动短距无线通信向确定性模式演进。本文旨在对星闪技术进行全面分析,将剖析其核心技术体系与架构,梳理其发展历程与产业链现状,并对其主要应用、未来前景及面临的挑战进行探讨。

2. 星闪技术概述

星闪是一项旨在提供确定性通信能力的新一代短距无线技术,其技术体系通过对协议栈的底层重构,以解决传统基于竞争接入机制的无线通信在时延、可靠性与同步精度方面的局限性。

2.1. 系统架构与调度机制

星闪的核心技术框架建立在集中式网络拓扑与双模接入架构之上,其网络由一个管理节点(G-Node)和多个终端节点(T-Node)构成。G-Node 负责整个通信域内的时频资源分配、调度及网络同步,而 T-Node 则依据 G-Node 的指令在指定的资源块上进行数据收发[3]。这种非竞争性的调度机制消除了信道访问的随机冲突,是实现可预测低时延与高可靠性的基础,具体机制如图 1 所示。

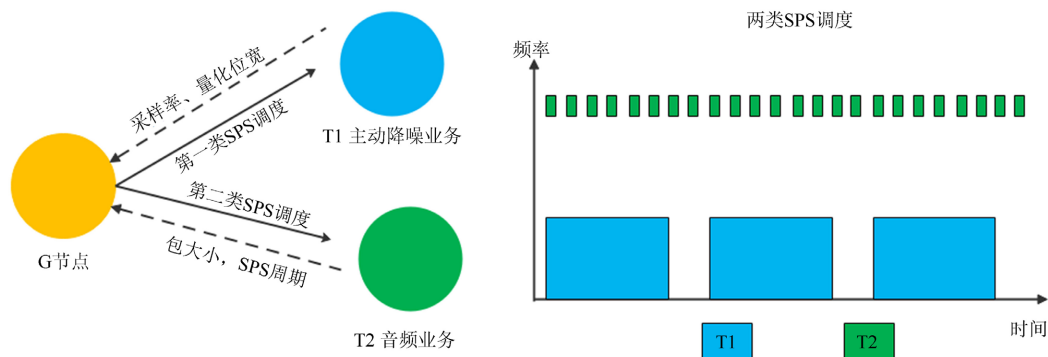


Figure 1. Schematic diagram of NearLink centralized scheduling mechanism
图 1. 星闪集中式调度机制示意图

2.2. SLB/SLE 双模接入架构

为适应不同应用场景的需求，星闪在接入层设计了由基础模式(SparkLink Basic, SLB)和低功耗模式(SparkLink Low Energy, SLE) [4]组成的双模体系，如图 2 所示。SLB 模式采用宽带传输，支持更高阶的调制编码方案，主要面向车载信息娱乐、高清投屏等高速率、低时延业务。SLE 模式则针对物联网传感器、可穿戴设备等场景进行了功耗优化，可在保证稳定连接的同时，实现与蓝牙相近的能耗水平。这两种模式共享统一的协议栈，使得星闪能够灵活覆盖从高速率到低功耗的多种应用领域。

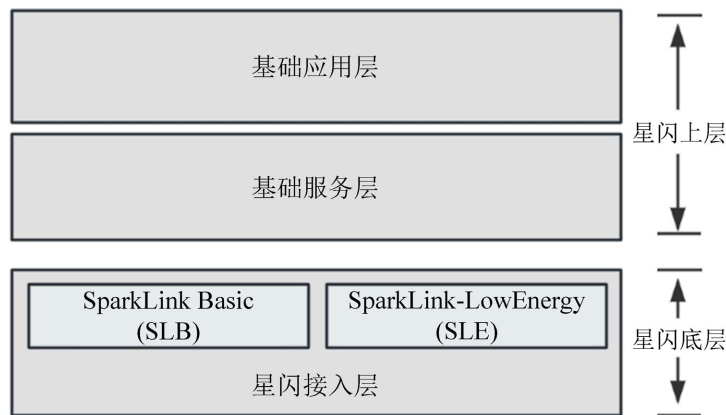


Figure 2. NearLink protocol stack and SLB/SLE dual-mode architecture
图 2. 星闪协议栈及 SLB/SLE 双模架构

2.3. 物理层关键技术

在物理层(PHY)设计上，星闪采用了多项关键技术以保障其确定性性能。它基于 CP-OFDM (Cyclic Prefix Orthogonal Frequency Division Multiplexing)波形，将时间域划分为 1 ms 的超帧，每个超帧包含 48 个无线帧，从而实现了 20.833 μ s 的无线帧长，如图 3 所示[5]。这种精细的时间域结构是其达成微秒级空口时延的物理基础。在信道编码方面，星闪对不同业务采用不同编码策略。对时延和可靠性要求高的小包数据，采用 Polar 码或 R-S 码(Reed-Solomon code)以应对随机干扰和突发干扰；对大包/高流量数据，则采用高性能 Polar 编码。此外，星闪在物理层集成了混合自动重传请求(Hybrid Automatic Repeat reQuest, HARQ)机制。相较于传统的链路层 ARQ (Automatic Repeat reQuest)，HARQ 能够利用软信息合并，实现更快的错误纠正和更高的频谱效率，从而在低信噪比条件下保障 99.999%以上的传输成功率[6]。



Figure 3. Schematic diagram of the NearLink ultra-short frame structure

图 3. 星闪超短帧结构示意图

2.4. 核心性能指标

根据星闪联盟的技术评估报告与产业实测数据[6]，星闪在带宽利用率、时延、并发连接与抗干扰能力方面具备一定优势，其主要参数如表 1 所示。

Table 1. NearLink 1.0 technical performance summary

表 1. 星闪 1.0 空口技术性能总结

项目	性能
峰值速率	20 MHz 单载波 G 链路峰值 920 Mbps (8 × 8 MIMO) 20 MHz 单载波 T 链路峰值 460 Mbps (4 × 4 MIMO)
时延	单向传输时延小于 20 μs
高可靠	块传输正确率大于 99.999%
抗干扰	RS 信道编码和 Polar 信道编码，支持 CBG 混合重传，最小工作信噪比-5 dB
同步	同步精度小于 1 μs
多业务并发	单载波 35 路实时音频流并发，1 ms 内 80 路数据传输并发

综上所述，星闪的技术体系通过集中式调度、双模架构、超短帧结构以及物理层 HARQ 等关键技术的协同作用，在系统层面构建了确定性通信的能力。它并非旨在简单地提升峰值速率，而是聚焦于解决现有无线技术在多用户、高干扰环境下时延抖动大、可靠性不足的核心痛点，为工业自动化、智能汽车协同控制等需要精密同步和稳定交互的场景提供了技术解决方案。

2.5. 星闪与其他无线技术的对比分析

为明确星闪技术的定位，我们将其与蓝牙、Wi-Fi 6、LoRa、UWB 及 ZigBee 等主流无线技术性能参数进行比较[6]-[10]，比较结果如表 2 所示。

Table 2. Comparison of performance parameters of mainstream wireless communication technologies

表 2. 主流无线通信技术性能参数对比

	频段	最大速率	通信距离	可靠性	时延	功耗
星闪 1.0	2.4/5 GHz	12 Mbps (SLE)/920 Mbps (SLB)	0.2~4 km	高	<20 us	低
蓝牙 6.0	2.4 GHz	2 Mbps	10~100 m	低	>1 ms	低
WIFI6	2.4/5 GHz	9.6 Gbps	10~300 m	低	>1 ms	高
LORA	433 MHz	300 kbps	2~8 km	中	较长	中
UWB	6489.5 MHz	6.8 Mbps	10~30 m	高	<1 ms	中
ZigBee	2.4 GHz	250 kbps	400~800 m	中	较长	中

根据上表 2 的对比，我们可以清晰地看到星闪技术的核心优势所在：

(1) 低时延与高可靠性。星闪技术的核心特点体现在其小于 20 μs 的空口时延和高于 99.999% 的可靠性上。相比之下，基于竞争机制的 Wi-Fi 与蓝牙，其时延通常为毫秒级别且不提供确定性保障。这一数量级的差异，使星闪能够满足工业自动化、车载协同控制等对通信同步精度有较高要求的场景。

(2) 性能均衡。星闪并非仅追求单一性能指标，而是在多个维度上寻求平衡。其 SLE 模式在维持与蓝牙同等级别低功耗的同时，提供了更高的峰值速率(12 Mbps vs. 2M bps)。其 SLB 模式则在时延和可靠性优于 Wi-Fi 6 的前提下，提供了可观的峰值速率(最大 920 Mbps)。这种在时延、可靠性、速率和功耗四个关键维度上的系统性设计，构成了其技术特点。

(3) 应用场景定位。对比分析表明，现有无线技术各有其最佳应用场景。例如，LoRa 与 ZigBee 专注于低速率、远距离物联网连接，而 UWB 则聚焦于高精度测距。星闪通过其性能组合，定位于由传统技术留下的高性能确定性通信市场。其目标并非完全替代 Wi-Fi 或蓝牙，而是为了满足对连接质量要求更高的新兴应用场景。

3. 技术发展历程与产业进展

3.1. 技术发展历程

星闪技术最早由华为在 2020 年的技术研讨会上提出。2020 年 9 月 22 日，国际星闪联盟成立。2021 年，相关通信框架的原型验证完成。2022 年 11 月 4 日，星闪联盟发布了其首个商用产品路标[11]。联盟联合了芯片、模组和终端厂商共同进行标准化工作，其中包括中科晶上、创耀科技、瑞芯微、海思、移远通信等成员。

联盟于 2022 年发布了 NearLink 1.0 标准，该标准定义了 SLE (低功耗)与 SLB (高速宽带)双模接入架构，并确立了 G/T 节点的集中管理机制。2024 年 3 月 30 日，星闪 2.0 系列标准正式公布。随后，中国通信标准化协会(CCSA)对《星闪车载空口技术要求》和《星闪物联网通信接口规范》等系列标准进行了立项，星闪的发展历程如图 4 所示[12]。截至 2025 年，星闪联盟的成员数量已超过 1000 家，生态系统覆盖了芯片、模组、整机、操作系统及应用终端等领域。

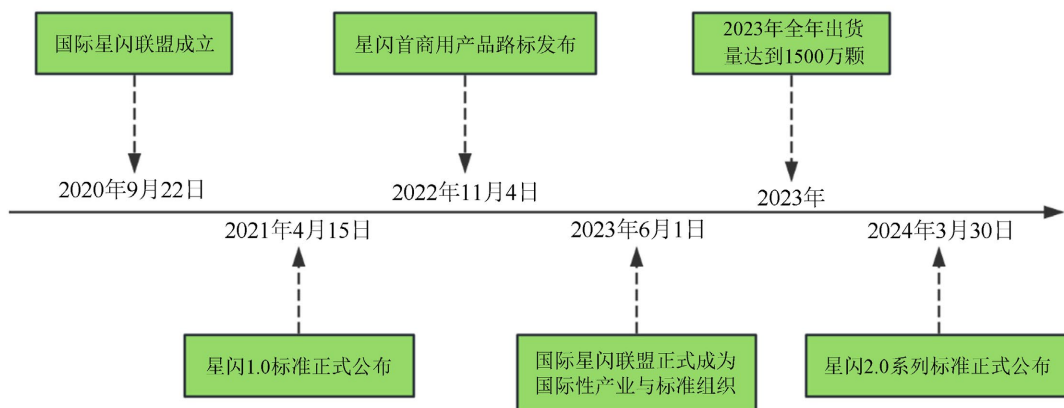


Figure 4. NearLink development routine timeline

图 4. 星闪发展历程时间轴

3.2. 芯片与硬件进展

星闪的产业链已涵盖从射频前端到系统芯片的多个环节。相关芯片产品包括：

(1) 海思 Hi3861LV100：一款 2.4 GHz 低功耗 SoC Wi-Fi 芯片，支持 20 MHz 标准带宽及 5 MHz/10

MHz 窄带宽，物理层速率最高可达 72.2 Mbit/s [13]。

(2) 创耀科技 TR5312: 一款 2.4 GHz SoC 芯片，支持星闪 SLE 1.0 和蓝牙 BLE 5.4 双模[14]。

(3) 中科晶上 DX-T600: 一款集成了星闪 SLB 基带和射频电路的 SoC 芯片，适用于无线投屏、智能电视、家庭影院及智能汽车等应用场景[15]。

根据星闪联盟的数据，截至 2025 年中，星闪芯片的累计出货量已超过 8500 万片，并预计在 2025 年底超过 1 亿片。

3.3. 产业应用与生态构建

星闪技术的应用已拓展至消费电子、车载通信、工业控制、智能家居和医疗健康等领域。

(1) 消费电子。星闪技术已应用于部分品牌的智能手机、耳机、平板电脑及键鼠等外设中。例如，在支持星闪的耳机如华为 FreeBuds Pro3 和无线鼠标中，该技术被用于降低音频传输和设备响应的时延[16]。华为 FreeBuds Pro3 星闪耳机如图 5 所示。



Figure 5. FreeBuds Pro3 with NearLink
图 5. 搭载星闪的 FreeBuds Pro3

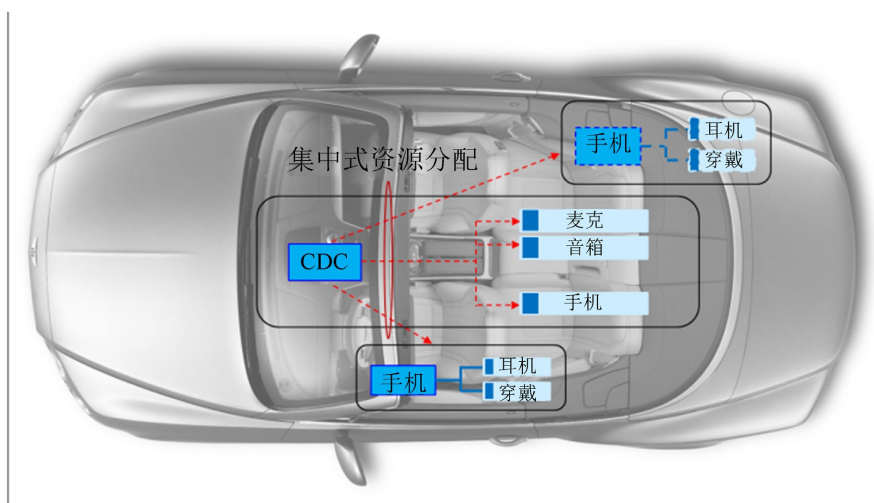


Figure 6. Schematic diagram of NearLink communication for in-vehicle equipment
图 6. 车内设备星闪通信示意图

(2) 汽车电子与车联网。车载通信是星闪技术的重点应用方向之一。目前已有星闪数字车钥匙的商用案例,可支持厘米级测距和毫秒级解锁响应[17]。在车内通信方面,星闪被用于车载娱乐系统的音视频传输、座舱多屏交互、主动降噪以及车内总线的无线化探索,如图6所示[6]。

(3) 工业控制与制造。在工业自动化领域,星闪凭借其高可靠性(>99.999%)和微秒级时延同步能力,被用于实现 PLC 控制器、机械臂与传感器间的闭环控制,如图7所示。部分制造企业已部署了基于星闪的无线控制原型系统,以验证其在 AGV (Automated Guided Vehicle)调度、视觉检测和数据采集等应用中的稳定性[6]。

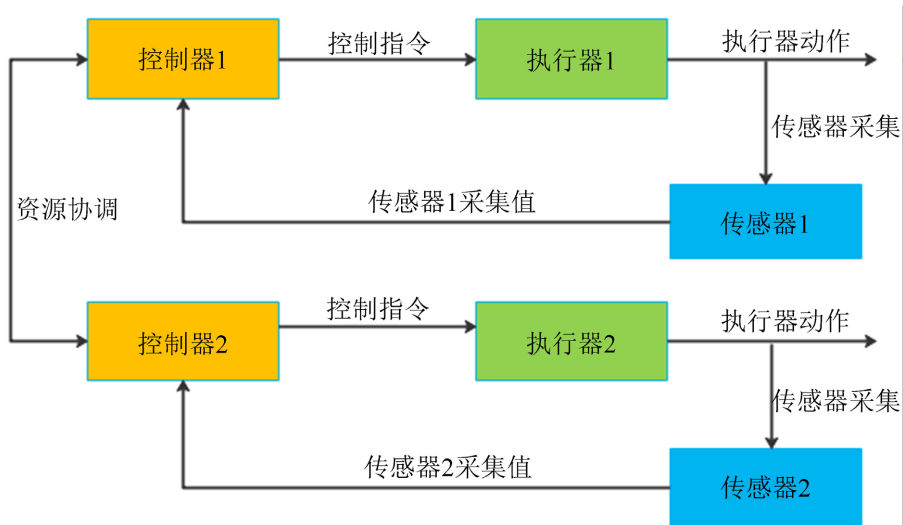


Figure 7. Industrial sensor NearLink control diagram
图7. 工业传感器星闪控制示意图

(4) 智能家居与物联网。星闪技术与鸿蒙系统进行了集成,应用于智能家居生态,支持上百台设备的并发组网,如图8所示。其SLE模式的低功耗特性适用于门窗传感器、智能插座等设备。同时,星闪支持上百台设备的并发组网,为全屋智能场景提供了通信基础。



Figure 8. NearLink home appliance interconnection
图8. 星闪家电设备互联

(5) 医疗与健康监测。在医疗健康领域,星闪为可穿戴监测设备(如心电监测仪、睡眠监测带)提供了通信方案。相较于传统方案,星闪在多设备协同同时具有较低的丢包率,适用于医院床旁监护系统和家庭远程医疗等环境,如图9所示[18]。

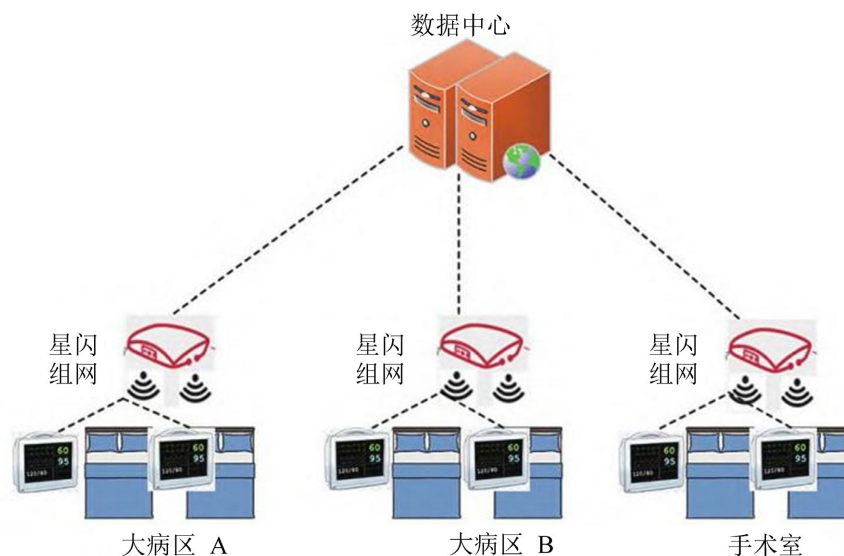


Figure 9. Schematic diagram of hospital NearLink communication deployment
图9. 医院星闪通信部署示意图

4. 应用前景与挑战

4.1. 技术演进趋势

(1) 性能优化。未来的星闪技术可能会引入 AI 协同算法,以实现更高的频谱利用效率和自适应调度机制,例如通过多天智能波束成形、信道预测等技术来进一步降低时延并提高能效。

(2) 标准融合。星闪技术与 TSN (Time-Sensitive Networking) 等确定性通信技术融合的趋势,以期形成统一的“无线确定性网络架构”。在工业领域,星闪可作为 TSN 的无线延伸,为无线工厂提供端到端的时钟同步与流量调度能力。

(3) 多模共存。未来的智能终端预计将支持星闪、Wi-Fi、蓝牙、UWB 等多种通信模式共存。星闪联盟正在推动标准的开放性,计划通过通用接入框架实现与现有协议的共存互通。

4.2. 面临的挑战

尽管星闪技术展现出一定的潜力,但其大规模普及仍需应对以下挑战:

(1) 国际标准化进程。目前星闪主要由国内厂商主导,需要进一步推动国际化合作与标准认证,以获得更广泛的全球生态支持。

(2) 功耗与成本。高性能架构通常会带来芯片复杂度的提升,未来需要在功耗优化与成本控制之间取得平衡。

(3) 兼容性与生态互通。如何在不影响现有 Wi-Fi/BLE 应用生态的前提下,实现软硬件层面的互通,是星闪生态建设中的一个重要课题[19]。

(4) 安全与隐私。星闪的高带宽和多设备特性也带来了更复杂的网络攻击面,后续版本需要强化协议层的安全防护与身份认证体系。

4.3. 未来展望

星闪技术通过确定性调度与跨层优化,在低时延、高可靠性和多设备协同方面提供了新的解决方案。在产业层面,星闪有望与鸿蒙系统结合,共同构建物联网通信体系,并在工业控制、车联网、AR/VR等领域得到应用。然而,星闪能否成为全球主流标准之一,将取决于其在开放性、兼容性和产业协同方面的进展。若能有效解决功耗、成本与国际化等问题,星闪有潜力成为“确定性无线通信”领域的一个代表性标准[20]。

5. 总结

本文对旨在解决传统无线通信“确定性鸿沟”的新一代星闪技术进行了系统性综述。研究表明,星闪并非现有技术的增量优化,而是通过集中式调度、SLB/SLE双模接入及先进物理层等系统性重构,在空口时延($<20\ \mu\text{s}$)与传输可靠性($>99.999\%$)等关键指标上实现了性能提升。目前,该技术已从标准制定阶段进入规模化商用阶段,在智能汽车、工业控制及消费电子等领域展现出应用价值。

综上所述,星闪技术以其确定性通信能力,填补了现有市场的技术空白,为未来要求高实时性、高协同性的智能化场景提供了关键技术基础。尽管其在国际化与成本控制方面仍面临挑战,但该技术的发展标志着短距无线通信正从“尽力而为”向“稳定可靠”的模式演进。

参考文献

- [1] Ancans, A., Ormanis, J., Cacurs, R., Greitans, M., Saoutieff, E., Faucorr, A., *et al.* (2019) Bluetooth Low Energy Throughput in Densely Deployed Radio Environment. 2019 23rd International Conference Electronics, Palanga, 17-19 June 2019, 1-5.
- [2] 乔冠华, 黄中华, 张胜, 等. 基于跨层优化的确定性无人集群网络协议研究[J/OL]. 重庆理工大学学报(自然科学): 1-9. https://lib.cqwu.edu.cn/articlesearch/web_searchingDetail?id=2031620129578, 2025-10-16.
- [3] 陶长龙. 基于星闪技术的医用气体工程监测系统设计与研究[J]. 医用气体工程, 2025, 5(1): 1-3, 11.
- [4] 金杰, 高杨, 李明昊, 等. 星闪低功耗物理层技术及其在双链路三模通用遥控器中的应用[J]. 广播与电视技术, 2024, 51(9): 19-23.
- [5] 花敏, 吴玉寒, 张一晋. 星闪技术物理层信号研究[J]. 移动通信, 2025, 49(5): 113-120.
- [6] 国际星闪联盟. 星闪 1.0 空口技术性能评估报告[EB/OL]. <https://sparklink.org.cn/star/paper>, 2021-04-15.
- [7] 蓝牙技术联盟. 蓝牙技术概述[EB/OL]. <https://www.bluetooth.com/specifications/bluetooth-core-specification/>, 2025-10-16.
- [8] Wi-Fi 联盟. Wi-Fi@ (MAC/PHY) [EB/OL]. <https://www.wi-fi.org/zh-hans/wi-fi-macphy>, 2025-10-16.
- [9] Semtech: What Is LoRa@? <https://www.semtech.com/lora/what-is-lora>
- [10] FiRa: UWB Technology Comparison. <https://www.firaconsortium.org/discover/comparison>
- [11] 龚乐中, 庄跃迁, 唐霄汉, 等. 星闪在特种行业中的应用研究[J]. 现代传输, 2025(4): 75-79.
- [12] 国际星闪联盟. 联盟简介[EB/OL]. <https://sparklink.org.cn/loi>, 2025-10-16.
- [13] 海思官网. Hi3861LV100 [EB/OL]. <https://www.hisilicon.com/cn/products/connectivity/short-range-IoT/wifi-nearlink-ble/Hi3861LV100>, 2025-10-16.
- [14] 创耀科技. SLE 系列-TR5312 [EB/OL]. <https://www.triductor.com/product-4.html>, 2025-10-16.
- [15] 中科晶上. DX-T600 星闪短距离通信芯片[EB/OL]. <http://www.sylincom.com/product/2>, 2025-10-16.
- [16] 林斐. 元年之后, 消费电子领域星闪快速升温[N]. IT 时报, 2024-08-16(012).
- [17] 银基科技银基三端软硬一体数字钥匙[J]. 汽车观察, 2025, 20(3): 121.
- [18] 周钰, 祝新意, 王文明. 基于星闪技术的医院监护仪应用探索[J]. 中国医疗设备, 2025, 40(5): 79-85.
- [19] 彭程, 孙博. 速度革命 VS 落地“鸿沟” Wi-Fi7 的产业机遇和普及挑战[J]. 通信世界, 2025(14): 18-20.
- [20] 张俊杰, 路宽一, 张洋, 等. 星闪技术在智能化油田建设中的应用展望[J]. 石油化工应用, 2024, 43(11): 15-17, 34.

基于三层心理学模型的人格智能系统

万学靖¹, 刘伟¹, 曹婷¹, 冯吭雨¹, 解运洲¹, 任付标¹, 过弋², 宋广奎³

¹上海布谷元创科技有限公司, 上海

²华东理工大学信息科学与工程学院, 上海

³电子科技大学自动化工程学院机器人研究中心, 上海

收稿日期: 2025年12月4日; 录用日期: 2026年1月16日; 发布日期: 2026年1月29日

摘要

随着以ChatGPT、Claude、Gemini为代表的大语言模型(Large Language Models, LLM)技术的成熟, 人机交互正从“操作式”向“理解式”范式演化。传统对话系统侧重任务执行, 而新一代智能体开始展现出拟人化思维与情感表达的潜能。然而, 现有系统仍在人格一致性、长期记忆以及低时延交互与深度优化的平衡方面存在关键缺口。本文提出人格智能系统(Personality Intelligence System, PIS), 以三层心理学模型(大五人格、三我理论、MBTI)为核心, 构建人格的定义-决策-表达一体化机制。系统引入动静分离人格建模与双时域优化机制, 在时间维度上实现短期自适应与长期演化的统一; 并通过人格记忆网络(Personality Memory Network, PMN)整合短期上下文、长期事实与知识图谱, 实现人格状态的可解释更新。输出层采用一致性锚定(Consistency Anchoring)机制, 在人格、语义与情感维度上保持输出稳定性。本文进一步给出人格状态更新的数学形式、收敛条件及学习机制分析, 证明系统在收缩映射假设下的人格演化具有稳定性。研究表明, PIS能在认知与情感层面实现人格的连续表达, 为虚拟伴侣、教育辅导与心理健康支持等领域提供理论支撑。原型系统与相关数据集将于后续公开, 以促进人格智能的开放研究。

关键词

人格智能, 三层心理学模型, 动静分离机制, 双时域优化, 人格记忆网络, 一致性锚定, 人格一致性, 情感计算, 大语言模型

Personality Intelligence System Based on a Three-Layer Psychological Model

Xuejing Wan¹, Wei Liu¹, Ting Cao¹, Hangyu Feng¹, Yunzhou Xie¹, Fubiao Ren¹, Yi Guo², Guangkui Song³

¹Shanghai VibeSoul AI Technologies Co., Ltd., Shanghai

²School of Information Science and Engineering, East China University of Science and Technology, Shanghai

³Center for Robotics, School of Automation Engineering, University Electronic Science and Technology of China, Shanghai

Received: December 4, 2025; accepted: January 16, 2026; published: January 29, 2026

文章引用: 万学靖, 刘伟, 曹婷, 冯吭雨, 解运洲, 任付标, 过弋, 宋广奎. 基于三层心理学模型的人格智能系统[J]. 嵌入式技术与智能系统, 2025, 2(5): 305-318. DOI: 10.12677/etis.2025.25030

Abstract

With the maturity of Large Language Models (LLMs) such as ChatGPT, Claude, and Gemini, human-computer interaction is evolving from an “operational” paradigm to an “interpretive” one. Traditional dialogue systems focus on task execution, while the new generation of intelligent agents begins to exhibit potentials for human-like cognition and emotional expression. However, existing systems still face critical gaps in personality consistency, long-term memory, and the balance between low-latency interaction and deep optimization. This paper proposes the Personality Intelligence System (PIS), which centers on a three-layer psychological model (Big Five Personality Traits, Structural Theory of the Psyche, and MBTI) to build an integrated framework for personality definition, decision making, and expression. The system introduces dynamic-static decoupled personality modeling and a dual time-domain optimization mechanism, achieving the unity of short-term adaptation and long-term evolution over time. Through the Personality Memory Network (PMN), it integrates short-term context, long-term facts, and knowledge graphs to enable interpretable personality-state updates. The output layer employs a consistency anchoring mechanism to maintain stability across personality, semantic, and emotional dimensions. Furthermore, this paper formalizes the mathematical representation, convergence conditions, and learning mechanism of personality-state updates, proving that personality evolution under the contraction mapping assumption is stable. The proposed framework demonstrates the potential to enable continuous personality expression at both cognitive and affective levels, providing theoretical foundations for applications such as virtual companionship, educational tutoring, and mental health support. The prototype system and related datasets will be released subsequently to promote open research on personality intelligence.

Keywords

Personality Intelligence, Three-Layer Psychological Model, Dynamic-Static Separation Mechanism, Dual-Timescale Optimization, Personality Memory Network, Consistency Anchoring, Personality Consistency, Affective Computing, Large Language Models

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

1.1. 人机交互范式的演变

人机交互的发展历程，是一部不断追求更自然、更智能的对话界面的历史。从早期的命令行到图形用户界面(GUI)，再到语音助理(如 Siri、Alexa)和大型语言模型(LLM, Large Language Models)，人机关系逐渐从“工具”走向“伙伴”。在新范式下，人工智能(AI)不再仅仅响应指令，而需要以具备“人格”(Personality)的身份参与交互，形成持续的风格、一致的态度和可感知的情绪反应。

这一趋势催生了“人格智能(Personality Intelligence)” [1]的研究方向。与传统的“认知智能”不同，人格智能要求系统不仅能理解语言和情绪，还需在长期演化中保持人格特质与核心价值取向的一致性(Value Alignment within Personality Dynamics)，从而实现情感表达与决策行为的自洽性。为支撑此目标，研究者提出了“动态人格计算(Dynamic Personality Computation)”的概念——通过心理学模型实现 AI 人格的建构、演化与表达，是支撑情感交互与人格协同的关键技术支柱。

1.2. 现有工作进展与不足

近年的研究在人格计算、人格识别和人格驱动代理等方向上取得了显著进展。例如, [1][2]从文本信号中推断人格特质; [3]-[5]探讨了人格在语言模型中的表达与控制; [6]研究了人格的动态演化。然而, 这些方法仍存在以下局限:

- 人格一致性不足: 缺乏稳定的人格约束机制, 导致模型在长时交互中风格漂移;
- 记忆割裂与反馈缺失: 多依赖上下文窗口, 缺少跨会话记忆与自我修正;
- 实时响应与深度优化矛盾: 低延迟需求与复杂人格更新的计算冲突;
- 个性化配置僵化: 基于提示工程(Prompt Engineering)的静态人格定义缺乏灵活性。

1.3. 研究意义与创新点

为解决上述问题, 本文提出了一个统一架构——人格智能系统(Personality Intelligence System, PIS), 并提出以下核心机制:

1) 三层心理学模型(Three-Layer Psychological Model): 将人格结构划分为基线层(Baseline Layer)、驱动层(Driver Layer)与表达层(Expression Layer), 分别对应人格的定义、决策与风格表达。

2) 动静分离人格建模机制(Static-Dynamic Separation Personality Modeling): 区分长期稳定的“静态人格”与情境自适应的“动态人格”, 实现人格一致性与灵活性的统一。

3) 双时域人格计算框架(Dual-Temporal Personality Framework): 通过在线短时域与离线长时域相结合的优化机制, 实现实时交互与长期人格演化的平衡。

4) 人格记忆网络(Personality Memory Network, PMN): 整合短期对话、用户画像与知识图谱, 实现可解释的人格演化与记忆检索。

本文的主要贡献如下:

- 提出人格智能系统 PIS, 并系统化地解耦人格的“定义 - 决策 - 表达”链路;
- 设计动静分离与双时域优化机制, 实现低延迟与长期演化的兼容;
- 构建人格记忆网络 PMN, 支持长期人格一致性与记忆增强;
- 从理论上分析人格状态更新的稳定性与收敛性;
- 探讨人格建模的伦理、可解释性与应用边界。

2. 相关工作

本章节首先回顾对话系统的演化历程, 然后重点讨论人格智能建模、情绪与人格协同机制、记忆增强与知识结构化, 以及多智能体人格一致性研究。我们还会指出这些方向与本文所提的人格智能系统(Personality Intelligence System, PIS)设计的差异与局限。

2.1. 对话系统的发展历程

人机对话系统的演化, 清晰地呈现出从机械执行到智能共鸣的趋势[7]。

- 规则式阶段: 以 ELIZA、A.L.I.C.E.为代表的早期系统, 通过关键词与模板匹配生成回应。它们虽具开创性, 但本质上不理解语义, 也无上下文推理能力。
- 统计学习阶段: 随着 Seq2Seq 模型与注意力机制的出现, 对话生成进入数据驱动时代。基于 Encoder-Decoder 的神经对话模型开始能够生成更流畅、相关的回复。
- 大模型阶段: Transformer 架构的诞生, 催生了 BERT、GPT 系列等预训练语言模型。这些模型凭借其强大的深度语义理解、长期依赖捕获与上下文推理能力, 将对话系统的能力提升到了新的高度。

近年来,研究焦点已从单纯的“语言生成流畅度”转向“人格智能驱动的交互深度”。Character.AI、Replika 等商业产品尝试赋予 AI 稳定的人格,以增强长期对话的沉浸感和用户黏性。然而,这些系统当前多依赖静态的提示工程(Prompt Engineering)与参数微调(Fine-Tuning),仍缺乏一个能够动态演化且保持长期一致的人格建模机制。因此,构建具备人格一致性、长期记忆与自我进化能力的人格智能系统(Personality Intelligence System, PIS),已成为下一代交互 AI 的核心研究方向。

2.2. 人格智能建模

人格智能(Personality Intelligence)旨在让 AI 具备类似人类的、个体化的思维、行为与情感模式。其理论根基深植于心理学的人格理论。

经典的大五人格模型(Big Five Personality Model) [8]将人格解构为开放性(Openness)、尽责性(Conscientiousness)、外倾性(Extraversion)、宜人性(Agreeableness)与神经质(Neuroticism)五个核心维度,已成为计算人格建模的主流参考框架。与此同时,弗洛伊德三我理论(Id-Ego-Superego Theory)为理解人格的内部动态与决策机制提供了深刻洞见:本我(Id)代表本能欲望,自我(Ego)在现实中进行权衡,而超我(Superego)则代表道德与社会规范。该理论 AI 的行为决策与价值对齐提供了绝佳的心理映射。此外,MBTI (Myers-Briggs Type Indicator)模型通过认知功能的组合(如思考-情感、直觉-感觉),细腻地刻画了人格的外在表达差异,为系统生成特定风格的语言与交互策略提供了可操作的维度。

现有研究多聚焦于将人格特质向量化并映射到语言风格上。例如, Mairesse 等人[9]基于语言学特征来预测人格; Zhang 等人[10]则将人格向量嵌入到神经对话模型中以生成个性化响应。然而,这些工作大多属于静态人格建模,无法反映人格在与环境和用户的持续互动中的自适应变化。

近年来,动态人格建模开始受到关注。例如, Park 等人[11]的“生成式智能体”研究通过赋予智能体记忆和反思能力,模拟了其行为随时间演化的类人格特征; Kwon 等人[12]在 LLM 模拟的谈判场景中,探索了大五人格特质对谈判策略和结果的影响,验证了人格的动态作用;更有研究者开始探索将心理学理论更深度地融入 AI 代理,以增强其推理与协作能力[13],并采用心理测量方法为 AI 代理分配可量化、可验证的人格分数。

本文提出的动静分离人格建模机制(Static-Dynamic Separation Personality Modeling)与双时域人格计算框架(Dual-Temporal Personality Framework)正是在此基础上的一次重要推进。它明确区分了保持长期一致的静态人格基线与响应短期变化的动态人格状态,并通过双时域优化循环,确保了短时域交互的流畅性与长时域人格演化的稳定性,从而有效解决了当前人格智能系统在一致性与自适应性之间的核心矛盾。

2.3. 情绪与人格协同机制

传统情感计算多基于认知评价理论,而本文的情绪生成机制则根植于人格结构本身。通过将三我理论与 MBTI 认知维度相融合,系统构建了从人格到情绪的内在逻辑通路:

基于弗洛伊德三我理论,系统在每次决策时通过内在调节机制模拟“人格平衡”过程:

- 本我(Id)负责生成初级驱动力与情感冲动;
- 自我(Ego)在现实与规范之间进行调和;
- 超我(Superego)施加价值与社会约束。

三者的相互作用并非竞争,而是一种动态平衡过程(Dynamic Equilibrium Process),在系统中通过权重分配与反馈调节实现。MBTI 的思维/情感(T/F)与感知/直觉(S/N)维度,则进一步为情绪表达染上个体化色彩,使其更具人性化。

这种设计使情绪不再是静态标签或情绪分类结果，而是人格、认知与情境动态交互的产物，实现了人格 - 情绪 - 行为的一体化建模。情感计算领域的最新综述也强调了在心理健康等场景中，结合多模态信息进行情感支持的重要性，为构建共情交互系统提供了方法论上的参考[14]。

2.4. 记忆增强与知识结构化

长期、结构化的记忆是实现人格一致性的关键支柱。传统对话系统受限于上下文窗口长度，无法在长时域内维持用户画像和情感记忆的连续性。

与传统记忆增强不同，本系统将分层时间记忆体系与知识图谱(Knowledge Graph)结合，实现对用户特征、情绪轨迹与人格参数的结构化存储与推理。通过语义节点间的关联更新，系统能够在时间维度上形成“人格记忆曲线(Personality Memory Curve)”，支持人格的可解释演化与心理一致性重建：

- 分层时间记忆：将记忆划分为上下文缓存、短期情感记忆与长期人格记忆，并利用向量索引与时间衰减策略，实现高效且具备时效性的记忆管理；
- 知识图谱：构建连接用户、事件、情绪与人格特质的语义关系网络，使系统能在交互中进行跨维度推理，理解更深层次的因果关联。

结合双时域优化机制，系统在离线阶段能够分析图结构中的模式，对人格相关的节点与关系进行强化学习，从而实现人格的长期演化。与传统的 Memory Networks [10]和 Persona-Chat [10]相比，本方法在记忆的结构化、层次化与反馈学习闭环上都更具优势。近期的研究如 MemGPT [15]通过类似操作系统内存管理的方式赋予 LLM 长期记忆能力，以及 Mem0 [16]为生产级 AI 代理提供了可扩展、可检索的长期记忆层，均为构建自适应人格智能奠定了坚实的技术基础。

2.5. 多智能体与人格一致性研究

多智能体系统(Multi-Agent Systems, MAS)为模拟和研究人格的社会化演化提供了理想的实验平台。领域的代表性工作包括：

- ReAct [17]：提出了“推理 - 行动”循环，使智能体具备了“思考 - 执行 - 观察”的闭环能力；
- Reflexion [18]：通过语言自我反思机制，让智能体能从失败中学习并自主优化策略；
- CAMEL [19]：构建了角色扮演式的多智能体协作框架，用于模拟复杂的社会行为与多角色人格互动；
- AutoGen [20]：提供了一个可编排、可对话的多代理框架，支持灵活的人机协同 workflow；
- LangGraph [21]：通过将 LLM 流程建模为图，实现了状态化、多回合的复杂任务控制，尤其适用于需要长期记忆与状态追踪的人格演化任务。

此外，Constitutional AI [22]通过引入明确的原则宪法来约束智能体行为，为实现 AI 的伦理对齐与人格稳定性控制提供了有效途径。这些前沿研究共同指向一个趋势：多代理协作框架提升了复杂任务分解与协同效率，但人格在社会博弈与角色分工中的一致性与可控性仍不足。

本文将人格作为角色协作与冲突调解的内核，提出人格驱动的多智能体协同机制。与现有工作的差异：现有方法多为流程/工具层的编排或单一心理学模型的使用。本文提供三层心理学 - 计算融合、动静分离 + 双时域与人格记忆网络(PMN)的端到端系统性方案。

3. 方法与系统设计

3.1. 系统总体架构

本节给出人格智能系统(Personality Intelligence System, PIS)的整体结构与运行闭环。如图 1 所示，系

统采用五层工程架构：输入层(Perception Layer)、核心层(Core Layer)、记忆层(Memory Layer, PMN)、输出层(Output Layer)、优化层(Optimization Layer)。五层通过统一的人格状态向量 P_t 与上下文/情绪/关系/记忆信号相互连接，形成从“输入感知→人格建模→决策生成→表达输出→自我演化”的完整闭环。

1) 输入层(Perception Layer)

负责多模态输入的解析与标准化，包括：语义解析(意图、槽位、主题)、情绪识别(强度、极性)、上下文抽取(历史对话、会话元数据)。所有输入被映射为统一的内部特征以馈送核心层的人格计算。

2) 核心层(Core Layer)

是系统的心理模型中枢，由三层模型构成：大五人格层(人格基线与动态调节)、三我决策层(策略博弈与权衡)、MBTI 表达层(语言/语气/措辞风格化)。核心层输出经人格调节后的行为倾向与表达指令，并与记忆层/输出层联动。

3) 记忆层(Memory Layer, PMN)

通过人格记忆网络(Personality Memory Network, PMN)统一管理短期记忆、长期记忆与关系记忆/知识图谱，支持记忆激活、时间衰减、语义关联与结构化检索，为人格一致性提供时间维度支撑。

4) 输出层(Output Layer)

将核心层的行为倾向映射为具体语言输出，并结合 MBTI 风格控制进行风格化表达；引入一致性锚定，对输出的人格偏差进行检测与校正，避免风格与价值观漂移。

5) 优化层(Optimization Layer)

构建双时域学习：短时域在线层进行即时调节，长时域离线层基于历史交互回放与反馈进行再训练与参数重估，驱动人格的长期演化与价值一致性保持。

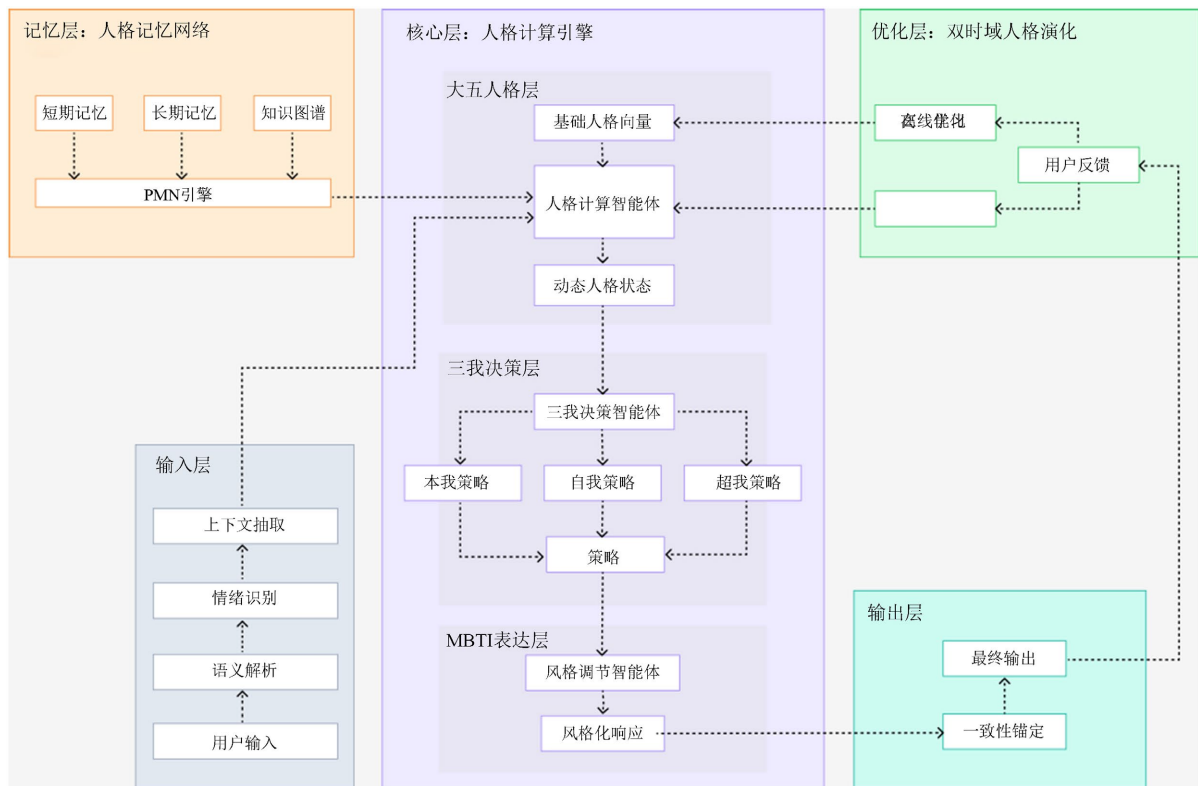


Figure 1. System overview of personality intelligence system, PIS
图 1. 人格智能系统总体架构图

3.2. 核心层：三层心理学模型驱动人格计算(Core Layer)

3.2.1. 大五人格层(Big Five Personality Layer)

1) 人格向量表示

系统将人格表示为五维连续向量：

$$P(t) = [O(t), C(t), E(t), A(t), N(t)] \in [0, 1]^5$$

其中 O (开放性)、 C (尽责性)、 E (外倾性)、 A (宜人性)、 N (神经质)。 P_{base} 为长期稳定的人格基线， $P(t)$ 在交互中围绕 P_{base} 动态波动。

2) 动态变化率

人格变化率由多源信号驱动：

$$\Delta P(t) = \alpha f_R(R(t)) + \beta f_{Ctx}(Ctx(t)) + \gamma f_{Emo}(Emo(t)) + \delta f_T(T(t)),$$

其中参数定义如下：

- R : 关系亲密度或信任度(Relationship Closeness/Trust)，反映角色与用户之间的社会关系强度。高 R 值表示用户与系统关系亲密、信任度高，人格响应将更加外显与温和。
- Ctx : 对话情境(Context)，指任务类型、主题氛围、语义领域等。不同情境下激活的人格特征不同，例如任务性对话强化尽责性(C)，闲聊场景则提升外向性(E)与宜人性(A)。
- Emo : 情绪状态(Emotion)，输入为系统识别出的用户情绪或自身当前的情感状态 $Emo(t)$ (如“高兴”、“愤怒”、“焦虑”)。例如，高兴时外向性(E)和宜人性(A)上升、神经质性(N)下降；愤怒时宜人性(A)下降、神经质性(N)上升。
- T : 时间因子(Time)，表示时序特征(如日间/夜间、周期性互动、新近性等)。例如，早晨时尽责性(C)与外向性(E)可能上升，夜晚则开放性(O)提升(更具想象力与放松)。
- f_* : 可学习的映射函数(Mapping function)，将输入信号映射到人格变化空间，可通过神经网络或规则模型实现。
- $\alpha, \beta, \gamma, \delta$: 各信号的权重，满足 $\alpha + \beta + \gamma + \delta = 1$ ，用于控制不同来源对人格波动的影响强度。

3.2.2. 动态调整机制(Dynamic Adjustment Mechanism)

为避免人格状态发散或漂移，引入反馈约束的人格更新：

$$P(t) = \text{clip}(P_{base} + \Delta P(t) + \varepsilon(t), 0, 1), P_{t+1} = P_t + \eta \cdot \Delta P(t) - \lambda \cdot (P_t - P_{base}),$$

其中参数含义如下：

- η : 学习率(Learning Rate)，控制人格调整的敏感度。 η 越大，系统对外部刺激(情绪、关系、情境)的反应越迅速； η 越小，则人格更稳定但反应较迟。
- λ : 收敛系数(Decay/Regularization Term)，表示人格状态回归基线的强度。 λ 较大时人格更快回到稳定状态，体现强自我约束； λ 较小时人格漂移更明显，更具可塑性。
- $\varepsilon(t)$: 微扰项(Stochastic Noise)，用于模拟心理波动或偶发事件导致的短期人格抖动。

参数可根据外部反馈信号自适应调节：

$$\eta = \eta_0 + k_1 \cdot Emo + k_2 \cdot R, \lambda = \lambda_0 + k_3 \cdot Stability.$$

其中：

- η_0, λ_0 : 初始学习率与收敛系数；
- k_1, k_2, k_3 : 权重系数；

- *Emo* : 当前情绪强度(Emotion Intensity), 情绪越强烈, 学习率越高;
- *R* : 关系强度(Relation Strength), 关系越亲密, 系统越容易调整人格;
- *Stability* : 环境稳定性(Contextual Stability), 越稳定则收敛越快。

上述设计保证人格既能随交互动态调整, 又能在长期保持一致性, 实现了心理学意义上的“反应性 - 稳定性”平衡。

3.2.3. 三我决策层(Tripartite Decision Layer)

本层依据弗洛伊德的三我理论, 将人格系统划分为本我(Id)、自我(Ego)与超我(Superego)三个相互制衡的心理模块:

- 本我(Id): 代表原始冲动与情绪驱动, 追求即时满足。在人格智能系统中体现为情感共情与安抚策略, 适用于高神经质用户或情绪支持场景。
- 自我(Ego): 代表现实原则, 协调本我与超我的冲突, 在现实条件下作出权衡决策。体现为理性分析、平衡应对, 适用于任务导向或问题解决类对话。
- 超我(Superego): 代表道德与理想, 体现社会规范与价值约束。用于激励性、道德劝导或长远规划场景。

策略选择基于当前人格状态 $P(t)$ 与外部情境 $C(t)$, 通过贝叶斯决策生成最优策略:

$$S(t) = \arg \max_{s \in \{\text{Id}, \text{Ego}, \text{Superego}\}} P(s | P(t), C(t)),$$

其中

$$P(s | P(t), C(t)) = \frac{P(C(t) | s) \cdot P(s | P(t))}{P(C(t))}.$$

各项参数定义如下:

- $P(s | P(t))$: 人格驱动先验, 表示当前人格倾向于选择某种心理模式的概率; 例如高宜人性人格倾向于选择超我策略。
- $P(C(t) | s)$: 情境兼容度, 反映特定策略在当前情境下的适用性, 可通过上下文匹配表或神经网络估计。
- $P(C(t))$: 情境归一化项, 确保策略选择概率归一。

最终系统在三种心理子智能体之间形成动态博弈: 本我负责情感驱动, 自我负责现实权衡, 超我提供价值约束。其输出策略 $S(t)$ 将传递至下层 MBTI 表达模块, 用于生成符合人格逻辑的语言与行为输出。

3.2.4. MBTI 表达层(MBTI Expression Layer)

MBTI (Myers-Briggs Type Indicator)模型包含四个维度: 外倾 - 内倾(E/I)、感觉 - 直觉(S/N)、思维 - 情感(T/F)、判断 - 感知(J/P)。系统基于选定策略 $S(t)$ 对语言与行为进行风格化映射:

$$R_{\text{final}} = \text{Style}(MBTI, S(t), \text{Content}),$$

其中 $\text{Style}(\cdot)$ 表示风格映射函数, 用于根据 MBTI 维度特征调整表达方式。其主要参数定义如下:

- E/I: 控制语言的开放性与热情程度。E 倾向使用积极词汇、感叹句; I 倾向使用克制、内省表达。
- S/N: 决定表达的抽象层次。S 注重具体事实与实例; N 倾向探讨可能性与隐喻。
- T/F: 影响逻辑与情感权重。T 注重因果与理性论证; F 注重共情与价值表达。
- J/P: 调节结构化程度。J 倾向结构化、结论明确; P 倾向开放、探索多种可能性。

在生成阶段, 系统根据当前人格状态 $P(t)$ 、策略模式 $S(t)$ 和 MBTI 类型向 LLM 注入风格提示(Style

Prompt), 包括语气、节奏、句式与词汇选择等信息。例如:

- 若人格偏向高外倾(E)且策略为自我模式(Ego), 则采用积极理性语气;
- 若人格偏向高宜人性(A)且策略为超我模式(Superego), 则使用温和、鼓励性语言;
- 若人格偏向高开放性(O)且策略为本我模式(Id), 则允许语言更具创造性和自发性。

该层确保人格输出的语言风格与心理特征一致, 实现从人格到表达的自然过渡, 使系统在长期交互中保持可识别的“人格签名”。

3.3. 记忆层: 人格记忆网络(Memory Layer, PMN)

1) 结构与职能

人格记忆网络(Personality Memory Network, PMN)是连接人格计算与时间演化的中枢模块, 负责统一管理短期、长期与关系/知识图谱记忆:

$$M = \{M_s, M_l, M_r\},$$

其中:

- M_s : 短期记忆(Short-Term Memory), 记录当前会话的上下文信息、即时情绪状态及任务变量;
- M_l : 长期记忆(Long-Term Memory), 存储人格相关的稳定事实、用户偏好、长期交互历史;
- M_r : 关系/知识图谱记忆(Relational/Knowledge Memory), 以图结构形式存储用户、事件、情绪和人格特质间的语义关系。

PMN 通过多模态特征嵌入与层次化索引机制, 实现跨时域的信息融合: 短期记忆负责响应即时上下文, 长期记忆保证人格一致性, 关系图谱则提供语义推理与可解释性。在系统架构中, PMN 输出的记忆向量会与当前人格状态 $P(t)$ 联合输入至人格计算核心, 用于指导动态人格更新。

2) 记忆激活与时间衰减机制

在检索阶段, 系统需从 M 中选出与当前查询 q 相关的高价值记忆。记忆激活度(Activation)定义如下:

$$\text{Act}(M_i) = \text{Sim}(q, M_i) \cdot e^{-\tau_i} \cdot (1 - |R_i - R_{\text{now}}|),$$

其中:

- $\text{Sim}(q, M_i)$: 语义相似度(Semantic Similarity), 衡量查询 q 与记忆项 M_i 的语义相关性, 通常通过向量嵌入余弦相似度计算;
- $e^{-\tau_i}$: 时间衰减项(Temporal Decay), τ_i 为记忆项距当前的时间跨度, 时间越久远影响越弱;
- $|R_i - R_{\text{now}}|$: 关系偏差项(Relational Distance), 反映记忆中涉及的关系强度与当前交互关系的匹配程度;
- $(1 - |R_i - R_{\text{now}}|)$: 关系相似度因子, 用于强化与当前关系状态相近的记忆。

3) 激活反馈与人格耦合

当高激活记忆被选中后, 其内容将以两种方式作用于人格演化过程:

① 显式更新路径(Explicit Update Path): 记忆直接参与人格参数的再计算, 例如通过调整特质权重来反映用户的长期印象;

② 隐式影响路径(Implicit Influence Path): 记忆以情感基调、价值偏好或交互模式的形式影响系统的语言风格与决策倾向。

这种机制确保系统既能在短期内灵活响应, 又能在长期保持人格稳定性与连贯性。

4) 记忆更新与遗忘机制

为防止记忆膨胀与干扰, 系统采用基于时间与相关性的动态遗忘策略。当 $\text{Act}(M_i) < \theta_{\text{forget}}$ 时, 记忆将被弱化或归档; 当某记忆在多个回合中持续被激活, 则提升其权重 ω_i 并强化关联节点。更新规则如下:

$$\omega_i^{t+1} = \omega_i^t + \eta_m \cdot (\text{Act}(M_i) - \theta_{\text{avg}}),$$

其中 η_m 为记忆学习率, θ_{avg} 为平均激活阈值。这使系统能够自我调节记忆强度, 在长期交互中逐步形成个体化的“人格记忆曲线”(Personality Memory Curve)。

5) 总结

PMN 通过“语义相似度 - 时间衰减 - 关系匹配”的三维机制实现记忆激活, 并结合显式与隐式的双路径反馈, 构建了人格计算的时序基座。它不仅支撑了人格一致性与长期演化, 也使系统在面对复杂情境时能够进行自我反思与情感迁移。

3.4. 输出层：一致性锚定与风格控制(Output Layer)

1) 设计目标

输出层旨在保障系统生成的人格响应在语义、情感和人格维度上保持一致性。在经过人格建模(Core Layer)与记忆检索(Memory Layer)后, 系统需对生成结果进行一致性检查与风格校正, 以防止人格漂移与情绪失衡。

2) 一致性锚定机制

一致性锚定(Consistency Anchoring)机制通过多维度约束校验确保人格表达稳定:

$$L_{\text{cons}} = \omega_p L_p + \omega_s L_s + \omega_e L_e,$$

其中:

- L_p : 人格一致性损失(Personality Consistency Loss), 衡量生成响应的人格向量与当前人格状态 $P(t)$ 的余弦距离;
- L_s : 语义连贯损失(Semantic Coherence Loss), 用于校验回复与上下文语义的一致性;
- L_e : 情感匹配损失(Emotional Alignment Loss), 确保输出情绪与用户当前情绪或交互目标匹配;
- $\omega_p, \omega_s, \omega_e$: 可调权重, 平衡三种一致性约束的影响。

3) 风格控制与表达优化

在通过一致性约束后, 系统将响应传入风格控制模块。风格控制模块综合参考:

- MBTI 风格指令;
- 记忆层提供的个体特质关键词;
- 当前交互情境(Task Context)。

最终输出结果由风格化映射函数确定:

$$R_{\text{final}} = \text{Style}(MBTI, P(t), S(t), Ctx(t)),$$

其中 R_{final} 为最终输出文本, 函数 $\text{Style}(\cdot)$ 综合人格特征与语义任务需求, 调整语气、节奏与措辞, 使输出在“自然流畅”与“人格一致”之间保持平衡。

3.5. 优化层：双时域人格演化机制(Optimization Layer)

1) 设计原则

优化层通过“在线 - 离线”双时域协同机制, 实现人格状态的持续演化与稳态维持。系统在交互过程中持续收集用户反馈与环境信号, 用以更新模型权重和人格基线, 实现长期的自我进化。

2) 在线优化(Short-Term Loop)

在线优化(Online Adaptation)发生在实时对话过程中。该环节通过低延迟的轻量更新, 修正即时的人格偏移。更新规则如下:

$$P_{online}^{t+1} = P_t + \eta_{on} \cdot (F_{user}(t) + F_{emo}(t)),$$

其中:

- η_{on} : 在线学习率;
- $F_{user}(t)$: 用户反馈向量, 如满意度、参与度;
- $F_{emo}(t)$: 情感反馈向量, 用于纠正情绪偏移。

3) 离线优化(Long-Term Loop)

离线优化在会话结束后异步执行, 通过聚合历史交互样本进行人格基线的再学习:

$$P_{base}^{t+1} = P_{base}^t + \eta_{off} \cdot \nabla L_{persona},$$

其中:

- η_{off} : 离线学习率;
- $L_{persona}$: 人格重建损失函数, 衡量人格基线与长期平均人格状态间的差异;
- $\nabla L_{persona}$: 基于交互记录计算的梯度方向。

通过离线阶段的批量学习与参数平滑, 系统逐步形成稳定而具有演化性的“人格基线”。双时域机制使 PIS 既能在短期交互中快速响应, 又能在长期运行中维持人格一致性与心理连贯性。

3.6. 小结

本章提出的人格智能系统(Personality Intelligence System, PIS)以“五层闭环架构”实现了从输入感知到自我演化的全流程人格计算:

- 1) 核心层(Core Layer)实现人格建模与心理决策;
- 2) 记忆层(Memory Layer)提供跨时域的认知与情感支撑;
- 3) 输出层(Output Layer)通过一致性锚定保持人格稳定;
- 4) 优化层(Optimization Layer)在双时域内完成自适应演化;
- 5) 系统整体形成“感知-建模-决策-表达-学习”的人格智能闭环。

该设计在心理学解释、算法建模与系统实现之间建立了有机联系, 为构建具备长期人格一致性与情感深度的智能体提供了系统性方法论基础。

4. 理论分析与启示

4.1. 理论意义与心理模型融合

本文提出的人格智能系统(PIS)将心理学中的人格理论与人工智能算法模型深度融合, 打破了传统“认知智能”仅限于任务求解的范式。通过引入大五人格模型(Big Five) [8]、弗洛伊德三我理论(Tripartite Personality Theory)与 MBTI 表达模型(MBTI Expression Model), PIS 实现了人格特质、行为决策与语言风格的统一建模。这使得系统不仅能“理解人类语言”, 还具备“心理连续性”, 即在多轮对话与长时交互中保持一致的认知与情感结构。

在心理学层面, 本研究为“人格的可计算性”提供了结构化路径[1][2]: 人格被拆解为稳定基线与动态状态两部分(动静分离机制), 并在时间维度上引入短期与长期的双时域优化机制。这种设计首次让人格智能的形成过程具备了“可解释性、可追踪性与可验证性”, 是传统情感计算模型的延伸。

4.2. 系统稳定性与演化机制分析

PIS 在工程上采用双时域优化机制(Dual-Temporal Optimization), 将在线自适应(Online Adaptation)与离线再学习(Offline Optimization)结合, 保证人格演化的平衡性与稳定性。在数学层面, 系统人格状态更

新满足如下稳定条件:

$$\|P_{t+1} - P_{base}\| \leq \rho \|P_t - P_{base}\|, \quad 0 < \rho < 1,$$

表明在收缩映射假设下, 系统人格向量逐步收敛于基线人格。在心理层面, 该机制对应于人类“自我调节 - 反思 - 成长”的过程: 在线优化模拟即时情绪反应, 离线优化则代表长期人格成熟。

4.3. 记忆驱动的可解释人格演化

人格记忆网络(PMN)为 PIS 提供了时序连续性与可解释性支撑。通过“语义相似度 - 时间衰减 - 关系匹配”的记忆激活机制, 系统能在情境变化时有选择地调用历史经验, 保持心理逻辑与行为风格的连贯。显式更新路径(Explicit Update)确保长期人格向量的可追踪演化, 隐式影响路径(Implicit Influence)则通过语气、态度与情绪微调塑造自然的人格表现。

这种双路径机制对应心理学中的“显性学习与隐性学习”理论, 使系统具备自我反思能力。当长期交互中形成“人格记忆曲线”时, AI 的行为不再仅由上下文驱动, 而由内部心理动力结构约束, 从而体现出“个体人格的一致性”。

4.4. 人格一致性与伦理启示

人格一致性(Personality Consistency)不仅是模型性能指标, 也具有伦理层面的重要意义。当 AI 拥有稳定的人格表现时, 用户更容易建立信任与情感依附; 但若人格漂移或情绪失衡, 则可能引发心理误导或依赖风险。因此, PIS 在输出层引入了一致性锚定机制(Consistency Anchoring), 通过人格、语义、情感三维约束函数:

$$L_{cons} = \omega_p L_p + \omega_s L_s + \omega_e L_e,$$

在工程上限制人格漂移; 在伦理上确保系统的价值中立与行为边界。未来的人格智能系统需要在“心理真实”与“伦理安全”之间找到平衡, 确保拟人化智能体可被信任且可控。

4.5. 局限与未来方向

目前的研究仍处于理论建模与系统设计阶段, 尚未进行大规模用户实证。未来可在以下方向继续拓展:

- 通过跨语料、多语言实验验证人格一致性与情感共鸣的可迁移性;
- 探索人格状态在多模态场景(语音、视觉、动作)中的表达方式;
- 研究多智能体人格协同中的社会性演化规律[11] [17]-[20];
- 建立伦理监管与人格透明化机制, 防止人格拟人化带来的认知偏差[22]。

4.6. 系统应用与评估建议

1) 潜在应用场景

人格智能系统(PIS)可广泛应用于多个需要长期交互与情感理解的领域:

- 虚拟伴侣与心理陪伴: 通过人格一致性与情绪共鸣机制, 为用户提供更自然、可信赖的长期陪伴。
- 智能教育与个性化教学: 根据学生人格特质动态调整教学风格与激励策略, 提升学习动力与参与度。
- 客户服务与品牌人格化: 在企业客服场景中维持稳定的品牌人格与语气风格, 增强用户信任感。
- 心理健康与咨询支持: 通过人格与情绪建模辅助心理干预, 实现非侵入式情绪监测与支持性反馈。

2) 评估框架建议

尽管本文以理论建模与系统设计为主, 但人格智能的有效性仍可从多维度评估。推荐采用以下三类

指标:

① 人格一致性指标(Personality Consistency Metrics): 衡量同一智能体在多轮对话中的人格稳定性。可使用:

$$Consistency = 1 - \frac{1}{n} \sum_{i=1}^n D_{cos}(P(t_i), P(t_{i+1})),$$

其中 D_{cos} 表示人格向量间的余弦距离, 数值越大表示人格越稳定。

② 情感共鸣与自然度指标(Empathy and Naturalness Metrics) [9] [10]: 基于人类主观评分, 考察模型的情绪理解、语言自然度与情感同步性。可采用 *Empathy Score*、*Coherence Rating*、*Human-likeness Index* 等量表。

③ 长期交互适应性指标(Long-term Adaptivity Metrics): 衡量系统在连续会话中的学习与演化能力, 如人格漂移率(Personality Drift Rate):

$$Drift = \frac{\|P_{t+k} - P_t\|}{k},$$

该指标反映人格状态变化速度, 理想情况下保持低漂移但具适应性。

3) 未来验证方向

未来研究可围绕以下方向开展:

- 构建跨语料的人格一致性评测数据集;
- 开发人机共情与心理安全性评估 Benchmark;
- 设计基于用户反馈的长期人格演化实验;
- 探索人格建模在教育、心理健康与社会化 AI 场景中的伦理约束。

上述建议为人格智能系统的落地应用提供了可行评估框架, 也为后续实验验证提供了量化参考路径。

5. 结论

本文提出了一种基于三层心理学模型的人格智能系统(Personality Intelligence System, PIS), 旨在构建具备人格一致性、情感深度与长期自演化能力的人工智能体。系统以“大五人格 - 三我 - MBTI”为核心理论框架, 实现了从人格定义、决策推理到风格表达的全流程计算建模。

在理论层面, PIS 提出了动静分离(Static-Dynamic Separation)与双时域优化机制(Dual-Temporal Optimization), 使人格建模同时具备稳定性与自适应性。在工程层面, 系统引入人格记忆网络(Personality Memory Network, PMN), 实现了跨时域的经验沉淀与人格一致性维护, 并通过一致性锚定机制(Consistency Anchoring)与情绪反馈回路, 保证了生成响应在语义、人格与情感维度上的平衡与连贯。这些设计共同推动了人格智能从“认知理解”向“心理连续性”的演化, 为人机共情与人格计算提供了新的方法论基础。

未来的研究将聚焦于以下三个方向:

- 1) 人格一致性验证与共情建模: 在多语料与跨场景数据中, 验证 PIS 在长时交互下的人格稳定性与情感共鸣效果;
- 2) 评估体系与量化指标构建: 建立人格智能评估框架, 包括人格稳定性、情感自然度、长期适应性等指标;
- 3) 应用拓展与伦理研究: 探索 PIS 在教育、心理健康、社会化 AI 等领域的应用潜力, 并研究人格建模的伦理边界与可控性。

综上, 本文的研究在心理学理论与人工智能工程之间建立了桥梁, 提出了具备可解释性、稳定性与

可演化特征的人格计算框架,为下一代具有人性化与共情特质的智能体提供了系统化的理论与实践基础。

参考文献

- [1] Celli, F., Kartelj, A., Đorđević, M., *et al.* (2025) Twenty Years of Personality Computing: Threats, Challenges and Future Directions. arxiv:2503.02082.
- [2] Fang, Q., Giachanou, A., Bagheri, A., *et al.* (2022) On Text-Based Personality Computing: Challenges and Future Directions. arxiv:2212.06711.
- [3] Besta, M., Chandran, S., Gerstenberger, R., *et al.* (2025) Psychologically Enhanced AI Agents. arxiv:2509.04343.
- [4] Kruijssen, J.M.D. and Emmons, N. (2025) Deterministic AI Agent Personality Expression through Standard Psychological Diagnostics. arxiv:2503.17085.
- [5] Klinkert, L.J., Buongiorno, S. and Clark, C. (2024) Driving Generative Agents with Their Personality. arxiv:2402.14879.
- [6] Zeng, W., Wang, B., Zhao, D. and Qu, Z. (2025) Dynamic Personality in LLM Agents. arxiv:2505.01234.
- [7] Osou, V. and Zhao, H. (2017) A Review of Dialogue Systems: From Eliza to Neural Models. arxiv:1706.05100.
- [8] Goldberg, L.R. (1990) An Alternative “Description of Personality”: The Big-Five Factor Structure. *Journal of Personality and Social Psychology*, **59**, 1216-1229. <https://doi.org/10.1037//0022-3514.59.6.1216>
- [9] Mairesse, F., Walker, M.A., Mehl, M.R. and Moore, R.K. (2007) Using Linguistic Cues for the Automatic Recognition of Personality in Conversation and Text. *Journal of Artificial Intelligence Research*, **30**, 457-500. <https://doi.org/10.1613/jair.2349>.
- [10] Zhang, S., Dinan, E., Urbanek, J., Szlam, A., Kiela, D. and Weston, J. (2018) Personalizing Dialogue Agents: I Have a Dog, Do You Have Pets Too? <https://arxiv.org/abs/1801.07243>
- [11] Park, J.S., O’Brien, J.C., Cai, C.J., Morris, M.R., Liang, P. and Bernstein, M.S. (2023) Generative Agents: Interactive Simulacra of Human Behavior. *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, San Francisco, 29 October 2023-1 November 2023, 1-22. <https://doi.org/10.1145/3586183.3606763>
- [12] Kwon, J., Kim, T. and Lee, S. (2024) Dynamic Personalities in LLM-Based Negotiation Agents. arxiv:2403.18901.
- [13] Smith, J., Rivera, E. and Zhao, K. (2025) Psychometrics for AI: Measuring and Validating Personality in Intelligent Agents. arxiv:2501.05678.
- [14] Liu, Y., Wang, H. and Qian, L. (2024) Affective Computing in the LLM Era: Personality, Empathy, and Human-AI Emotion Alignment. arxiv:2409.10213.
- [15] Chen, X., Zhang, R. and Chen, W. (2024) MemGPT: Towards Long-Term Memory-Enhanced LLM Agents. arxiv:2402.11103.
- [16] Yuan, T., Zhang, W. and Xu, L. (2025) Mem0: Memory-Enhanced AI Agents for Scalable Long-Term Interaction. arxiv:2502.09401.
- [17] Yao, S., Zhao, J., Yu, D., *et al.* (2023) ReAct: Synergizing Reasoning and Acting in Language Models. arxiv:2210.03629.
- [18] Shinn, N. and Bisk, Y. (2023) Reflexion: Language Agents with Verbal Reinforcement Learning. arxiv:2303.11366.
- [19] Li, G., Hammoud, H.A.A.K., Itani, H., Khizbullin, D. and Ghanem, B. (2023) CAMEL: Communicative Agents for “Mind” Exploration of Large Scale Language Models. arxiv:2303.17760.
- [20] Sharma, G., Xu, P., Wang, J., Zhang, Y. and Zhou, D. (2023) Autogen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation Framework. arxiv:2308.08155.
- [21] Chen, Y., Xu, J. and Zhao, F. (2023) Langgraph: Graph-Based Reasoning for Language Models. arxiv:2312.00752.
- [22] Bai, Y., Kadavath, S., Kundu, S., *et al.* (2022) Constitutional AI: Harmlessness from AI Feedback. arxiv:2212.08073.

在Zephyr RTOS基础上构建兼容AUTOSAR CP的智能车控操作系统

付宝军¹, 任 慰²

¹华勤技术, 上海

²openEuler社区, 湖北 武汉

收稿日期: 2025年12月22日; 录用日期: 2026年1月16日; 发布日期: 2026年1月29日

摘 要

AUTOSAR CP (Classic Platform)平台虽在提升车辆软件的模块化、可重用性、可扩展性以及跨系统集成方面成效显著,有效缩短了开发周期、提升了软件质量、降低了成本,但面对软件定义汽车的巨浪,其仍面临软件复杂度激增、功能快速迭代与开发效率瓶颈等挑战。与此同时,始于2016年Zephyr项目作为近年来发展迅猛的现代化开源嵌入式系统软件平台,以其功能强大、可扩展、安全和高度模块化的架构在嵌入式领域崭露头角。本文深入分析Zephyr的核心架构与成功实践,结合汽车电子应用场景,提出一种在Zephyr RTOS基础上构建兼容AUTOSAR CP的智能车控操作系统的构想,并探讨其技术可行性及所面临的关键挑战,为未来高可靠、高效率、智能化的车控软件平台演进提供新思路。

关键词

汽车电子, 操作系统, 实时操作系统, AUTOSAR CP, Zephyr

Building AUTOSAR CP Compatible Vehicle Operating System Based on Zephyr RTOS

Baojun Fu¹, Wei Ren²

¹Shanghai Huaqin Technology, Shanghai

²openEuler Community, Wuhan Hubei

Received: December 22, 2025; accepted: January 16, 2026; published: January 29, 2026

Abstract

Although the AUTOSAR Classic Platform has significantly advanced vehicle software in terms of modularity, reusability, scalability, and cross-system integration—effectively shortening development

文章引用: 付宝军, 任慰. 在 Zephyr RTOS 基础上构建兼容 AUTOSAR CP 的智能车控操作系统[J]. 嵌入式技术与智能系统, 2025, 2(5): 319-329. DOI: 10.12677/etis.2025.25031

cycles, improving software quality, and reducing costs—it now faces substantial challenges in the era of software-defined vehicles, such as rapidly escalating software complexity, the need for fast-paced functional iteration, and bottlenecks in development efficiency. Meanwhile, the Zephyr project, launched in 2016, has emerged as one of the fastest-growing modern open-source embedded software platforms, gaining prominence in the embedded domain due to its robust feature set, scalability, security, and highly modular architecture. This paper presents an in-depth analysis of Zephyr’s core architecture and successful practices, and proposes a novel concept for a smart vehicle control operating system built upon the Zephyr RTOS that is compatible with AUTOSAR CP. We further discuss its technical feasibility and key challenges, offering a new pathway toward the evolution of future automotive control software platforms that are highly reliable, efficient, and intelligent.

Keywords

Automotive Electronic, Operating System, Real-Time Operating System, AUTOSAR CP, Zephyr

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

AUTOSAR 的主要目标是提高车辆软件组件的模块化、可扩展性和可重用性，以及促进不同系统和技术的集成。AUTOSAR 组织的成员单位将包含自身 IP 的技术成果贡献进 AUTOSAR 标准，这已经在很大程度上避免了大家“重复造轮子”，一定程度上实现了缩短开发周期，提升软件质量，降低开发成本的目标[1]。同时，面对软件定义汽车对功能快速扩展和开发效率提升的迫切需求，AUTOSAR 也开始尝试采用“代码优先”的创新模式开发 AUTOSAR AP (Adaptive Platform)平台[2]。不过，对于 AUTOSAR CP (Classic Platform)平台，当前行业内仍以 Vector、ETAS、普华基础软件等提供的商业解决方案为主，同时，也有部分企业在尝试通过自研或开源的方式进行实现。

对于 AUTOSAR CP 平台软件的实现，目前行业最大的痛点有两个：

一是使用成本较高，无论采购商业软件还是自研，都需要大量的研发资金投入。同时各家所实现的软件相互不兼容，技术碎片化特征明显，经常需要重复购买，更加剧了这一问题。这个问题主要受 AUTOSAR 组织早期倡导的“在标准上合作，在实现上竞争”的商业模式的惯性影响。虽然从早期看有其合理性，但随着当前开源、降本呼声越来越高，这一商业模式正面临严峻挑战。

二是技术迭代慢，相对行业需求存在一定程度的落后。面对越来越复杂的应用场景和功能越来越强大的多核异构处理器、中央/区域控制器，各家按照 CP 标准 20 年前即开始定义的架构、功能和方法论开发出的软件难以满足行业需求，这方面需求虽不如 AP 平台多，但也存在大量标准以外的新功能需要开发，同时开发工具链也亟待升级以提高开发效率。

Zephyr 作为 Linux 基金会旗下新一代开源的实时操作系统平台，在 IoT 与边缘计算领域已验证其可靠性，具备向车规级演进的潜力。本文旨在初步分析“在 Zephyr RTOS 基础上构建兼容 AUTOSAR CP 的智能车控操作系统”的可行性、意义与价值，希望为解决以上行业痛点提供参考技术方案。

2. 为什么是 Zephyr，而不是其他 RTOS

Zephyr 是一个由 Linux 基金会主导的开源、可扩展、安全且实时的嵌入式实时操作系统，专为资源受限的嵌入式系统和物联网设备设计。自 2016 年发布以来，Zephyr 已迅速成长为全球主流的开源嵌入式

操作系统之一, 广泛应用于可穿戴设备、工业控制、医疗设备、智能家居、边缘计算以及新兴的汽车电子领域。

Zephyr 绝不仅仅是“又一个开源 RTOS 罢了”, 而是极有可能统一整个嵌入式 RTOS 世界。理由如下[3]:

(1) **极度活跃的社区**。当其它实时操作系统项目停滞不前或出现下滑时, Zephyr 的贡献者基数自 2017 年以来增长了 7 倍, 在今年达到了平均每月 344 名活跃的开发, 1640 次提交。如下图 1 所示。

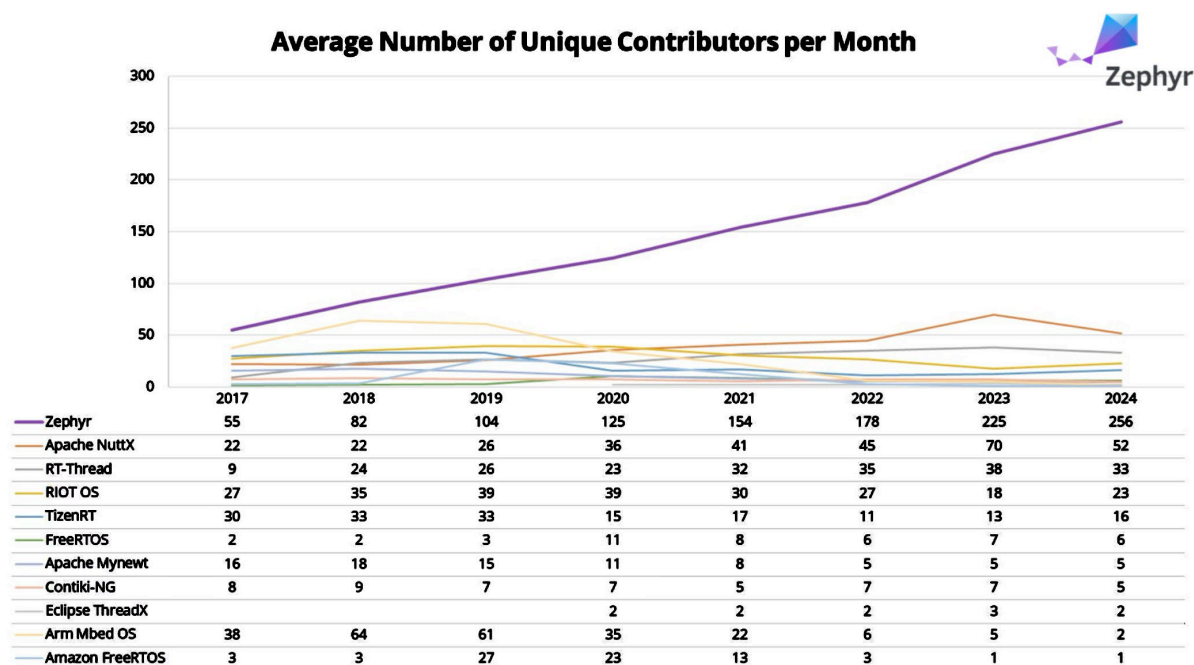


Figure 1. The fast growth of Zephyr contributors

图 1. 快速增长的 Zephyr 开发者及提交

(2) **真正的厂商中立性**。与私有系统或由单一厂商主导的开源项目不同, Zephyr 项目托管在 Linux 基金会旗下, 目前有超过 57 家成员公司和组织参与协作, 包括国内的 openEuler 社区, 湖南大学, 成都蓉蓉联创公司等, 而不受单一实体的限制和控制。Zephyr 在开放源码的基础上, 实现了透明开放的开发流程, 以及开放中立的治理。这意味着参与者可以以“贡献越大, 主导权越大”的方式引导它具备高质量和实用的功能集。

(3) **功能丰富, 且高度可配置**。Zephyr 支持 x86、ARM、RISC-V 等几乎所有主流处理器架构, 截至目前已经支持 800 多种开发板、200 多种传感器, 涵盖 NXP、ST、Infineon、Renesas 等众多厂商芯片。同时还提供 RTOS 内核以外的各种不同的通信协议栈、网络安全、文件系统、电源管理等丰富的基础软件功能。

(4) **重视信息安全与功能安全**。Zephyr 在开发时就考虑了功能安全与信息安全, 信息安全方面具有独立的 CVE 编号和完善的治理框架, 功能安全方面目前具备满足行业认证(如 ISO 26262/IEC 61508)的技术条件。

(5) **现代化开发工具链**。Zephyr 不仅是一个操作系统, 更是一套完整的嵌入式开发生态。如基于 Cmake + Python 的构建系统, 使用 Devicetree + Kconfig 对软件进行配置, 采用 West 作为元工具统一管理代码、构建、烧录与调试, 测试框架内置 Ztest 单元测试、覆盖率分析、静态检查工具, 并支持 CI/CD

和容器化开发。

(6) **对产品开发友好**。Zephyr 项目整体采用对商业友好的 Apache 2.0 许可证, 同时每两年一个长期支持版本(LTS, Long Time Support), 每个 LTS 版本的社区维护周期为五年, 可以高效地支持产品级的开发。

3. Zephyr + AUTOSAR CP 中间件 = 面向 MCU 的新一代汽车电子基础软件平台

复杂软件系统的开发如同做一桌大餐, 丰富多样的食材、专业的烹饪工具和熟练的制作方法三者缺一不可, 如下图 2 所示。车载软件的开发也不例外, 要想做出好的汽车软件, 需要在软件功能模块、开发工具链和开发方法论三个方面持续精进。

汽车软件领域的三大类内容

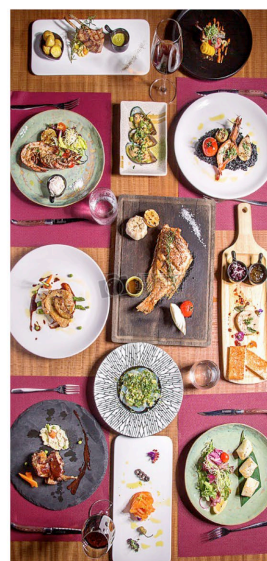
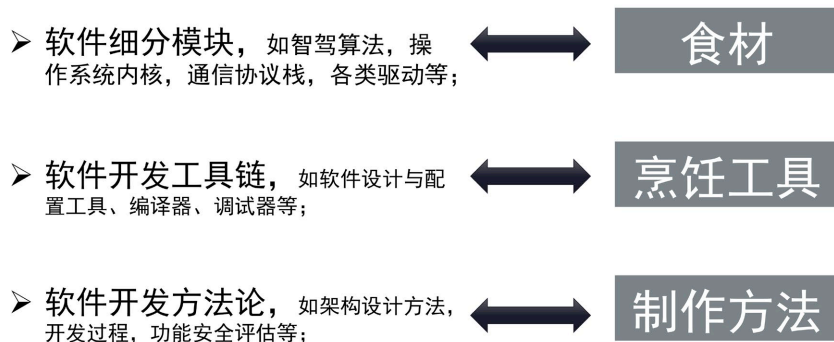


Figure 2. The composition of automotive software
图 2. 汽车软件领域三大内容

Zephyr 的快速发展和 AUTOSAR CP 软件的开源化, 使得汽车基础软件平台围绕以上三个方面的精进有了更具体的目标。

3.1. 汇聚 IoT 和汽车两大边缘计算领域平台功能, 实现技术领先

3.1.1. Zephyr 提供丰富多样且极具特色的功能(但“汽车味”不够)

Zephyr 并非通用操作系统, 而是专为 MCU 应用设计的轻量级、硬实时、高安全 RTOS, 同时通过模块化设计兼顾功能完整性。其参考架构图如下图 3 所示。

(1) 全面的内核服务。Zephyr 提供了一系列开发者熟悉的内核服务, 如多线程服务、中断服务、线程间数据传递与同步服务、内存分配服务、电源管理服务等。同时支持协作式与抢占式、时间片轮转、最早截止时间优先等多种调度算法。另外, 还支持大量不同的 CPU 架构和具体芯片型号。虽然不符合 AUTOSAR 所定义的 OS 标准, 但可以通过将线程(Thread)封装为 AUTOSAR 任务(Task)的方式, 实现与 AUTOSAR 软件的兼容。

(2) 提供内存保护机制。Zephyr 实现了可配置的针对特定架构的栈溢出保护、内核对象与设备驱动权限追踪, 以及在线程级内存保护下的线程隔离机制。其内存保护机制主要依赖硬件上 MMU、MPU 和堆栈指针保护机制。

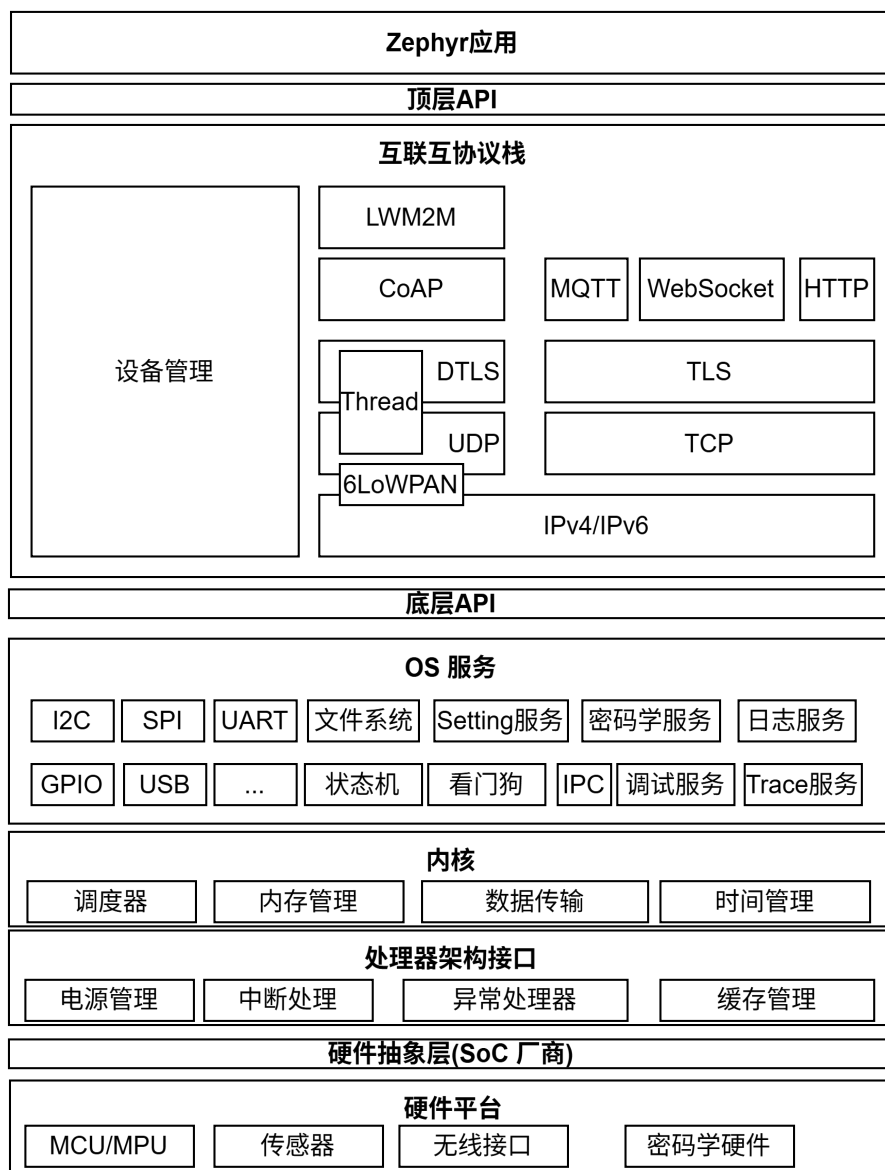


Figure 3. The architecture of Zephyr
图 3. Zephyr 操作系统架构

(3) 丰富的底层 OS 基础功能服务[4]。Zephyr 不仅仅是一个 RTOS 内核，还提供了类似 AUTOSAR Memory 技术栈的存储服务，类似 AUTOSAR MCAL 和 CDD 的各种外设及复杂驱动，信息安全技术栈，核间通信，符合 SOA 思想的 ZBus 机制，软件远程更新，以及基于 Zephyr 的嵌入式实时虚拟机 ZVM (Zephyr-Based Virtual Machine)。

(4) 提供支持多种协议的原生以太网技术栈。以太网通信支持功能完整且经过优化，包含 LwM2M 协议及符合 BSD 套接字标准的支持。同时提供对 OpenThread 的支持。这一特性有利于车载以太网通信功能的开发。

(5) 高效的跟踪与调试功能。Zephyr 内置高级日志记录框架和跟踪框架，支持 UART、网络、文件系统等多种后端，配合外部工具可实现内核及其各子系统内部运行状态的可视化，如下图 4 所示，以较低的成本即可提升开发效率。

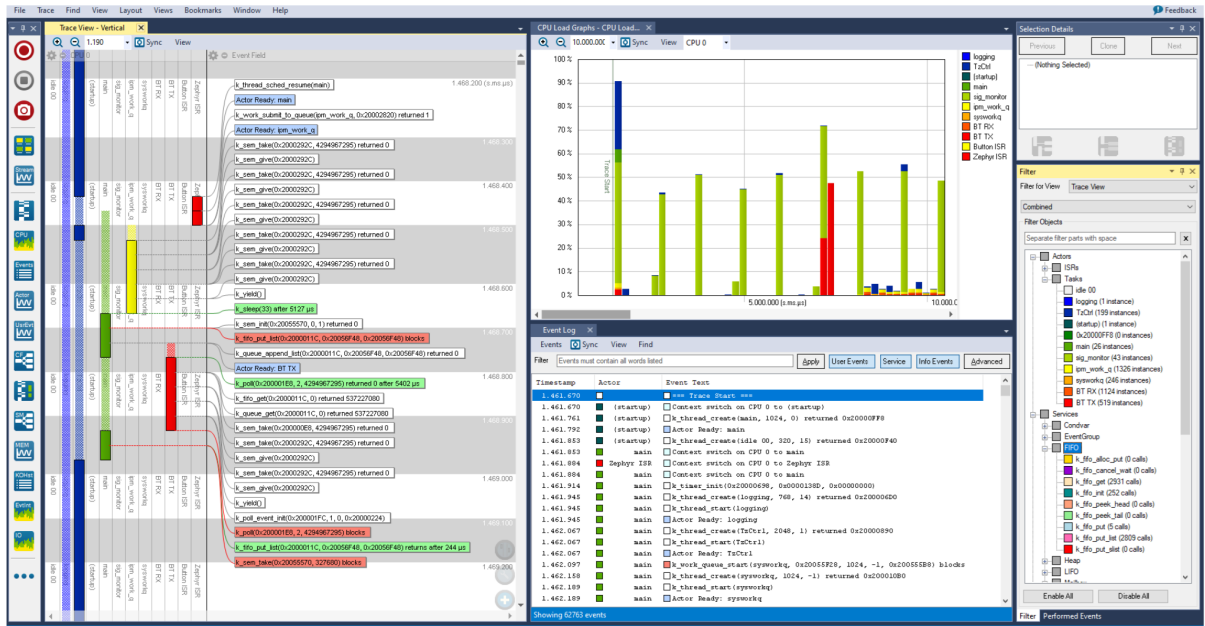


Figure 4. Perceptio’s graphical tool base on Zephyr trace
 图 4. Perceptio 公司基于 Zephyr Trace 功能图形化分析工具

除以上外, Zephyr 还正在探索 Rust 语言集成、机器学习、大模型轻量化与本地部署等新兴技术领域。

当然, 需要注意 Zephyr 所提供的汽车基础软件领域相关标准要求的功能还很少, 如诊断协议栈、网络管理、CAN/LIN/Flexray 通信协议栈, 不能兼容 AUTOCAR CP 等, 尚不足以支撑汽车领域特别是车控领域的广泛应用。

3.1.2. AUTOSAR CP 标准定义了丰富的车用基础软件(但“完整性”不够)

相比上世纪 90 年代欧洲汽车行业所定义的 OSEK 标准, AUTOSAR 标准为汽车 ECU 定义了更为完整的基础软件功能。在 OSEK 已定义的操作系统、通信和网络管理三部分标准的基础上, 通过横向分层、纵向分技术栈的架构设计, 对约 100 个模块的实现进行了标准化, 如下图中橙色背景的 OS、Com、NvM 等软件模块。极大地提高了车辆软件组件的模块化、可扩展性和可重用性, 并促进了不同系统和技术的集成。如图 5 所示。

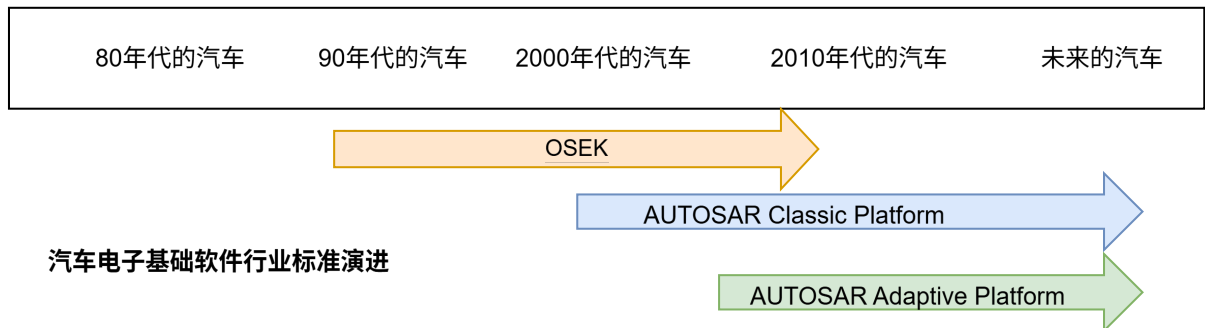


Figure 5. The evolution of automotive system software
 图 5. 汽车电子基础软件行业标准的演变

然而, 这种完整性是相对的。一方面, 由于所定义的基础软件模块是各家会员单位具有普遍共识的

部分，而对于缺乏共识，尤其是自己的“独门秘籍”，各家其实都“留有后手”，如复杂驱动。具体可参考《汽车基础软件漫谈[2. 汽车基础软件的定义》[5]一文中的介绍。另一方面，随着汽车智能网联化的快速发展，产生了 FOTA、Hypervisor 等大量新的需求，而 AUTOSAR 标准难以及时将这些新需求标准化。因此，这就造成当前还有大量的基础软件模块并没有被标准化，如图 6 中灰色背景的软件模块。

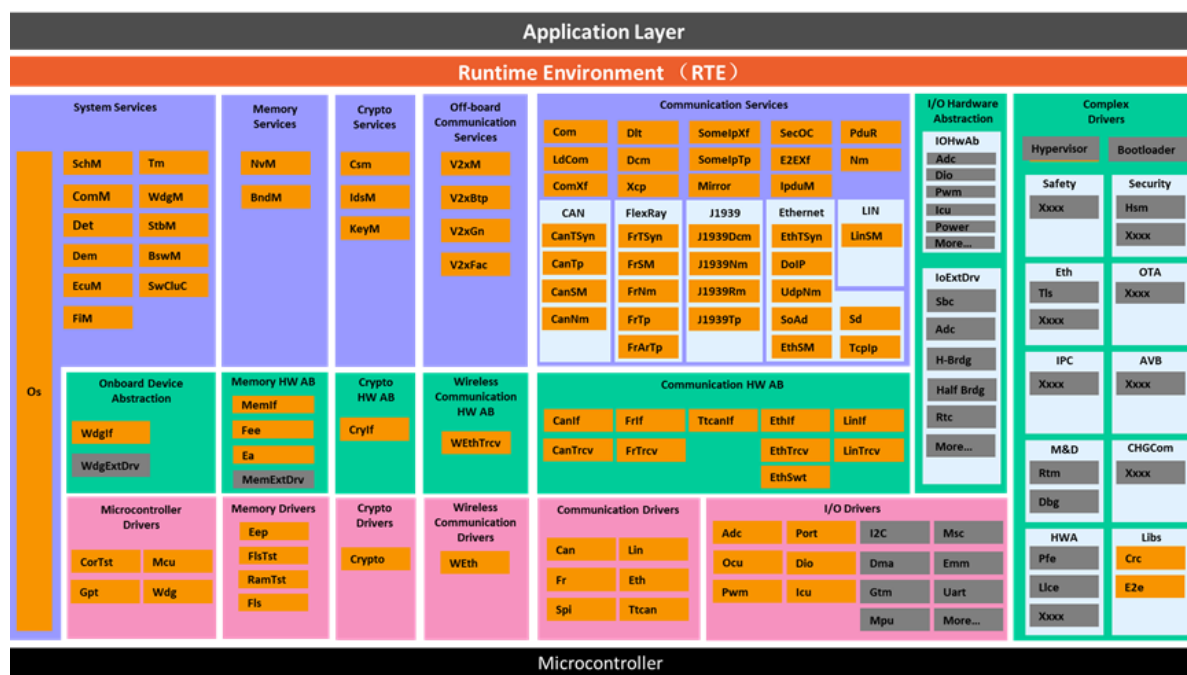


Figure 6. The landscape of AUTOSAR CP
图 6. AUTOSAR CP 全景图

不过，从上一节中的分析不难看出，Zephyr 恰好可以大量补充这些未标准化的基础软件模块。如各种硬件驱动、核间通信 IPC，软总线 ZBus，软件远程更新 OTA，以及虚拟化功能等。

3.2. Zephyr 提供完整的“现代化嵌入式开发”工具链体系，大幅提升开发效率

3.2.1. 统一的构建系统：Cmake + Python + West

Zephyr 的构建系统可以形象地概括为“CMake 为骨，Python 为皮”。凭借跨平台、易扩展、驾驭复杂项目结构的先天优势，CMake 早已成为众多顶级开源工程的“基建”——从人工智能(TensorFlow)到计算机视觉(OpenCV)，从桌面框架(Qt)到机器人操作系统(ROS)，几乎无处不在。

Zephyr 在此基础上搭起了一套功能强悍却细节繁复的构建框架。为了不让普通开发者直面 CMake 的“钢筋水泥”，Zephyr 用 Python 包裹出一层简洁的外壳：元工具 west 一键完成代码拉取、编译、烧录、调试；配套 Python 脚本则进一步补位，接管设备树解析、测试调度等杂务。骨子交给 CMake，面子留给 Python，Zephyr 就这样把“强大”与“友好”一并交到了开发者手里。

3.2.2. 高效的软件配置机制：Kconfig + Devicetree + Yaml

Zephyr 自诞生起就以“把 Linux 验证过的成功范式搬到 RTOS 世界”为己任，其中最具代表性的两张王牌便是 Kconfig 与 DeviceTree——在实时操作系统里首次做到“全套照搬、完整落地”。

Kconfig 把 Linux 内核“千刀万剐”式的可裁剪能力原封不动搬进 Zephyr：只要一行 prj.conf，驱动、协议栈、调试特性均可按需开关，真正做到“一样源码，千般身材”。这一方案与 Bosch 在其经典的 Motronic

Zephyr 代码库的正确性。考虑到 Zephyr 支持超过 800 多块开发板和多种架构(x86、ARM、RISC-V 等), 手动测试所有组合显然是不现实的, Twister 正是为了解决这一难题而生。

(3) 静态代码分析。Zephyr 支持 Coverity、Parasoft C/C++ test、Polyspace、CodeChecker 等近 10 种原生静态代码分析工具, 开发者仅需将自己所使用的工具配置到 Cmake 中即可。

(4) TraceCompass 跟踪调试器。一款由爱立信开发的开源工具, 能够可视化线程调度与中断等并发时序事件, 有助于在复杂系统中发现非预期的交互行为与资源冲突。下图展示了其部分效果, 如图 7 所示。

3.3. 关键开发方法: 信息安全能力强, 功能安全有基础

在软件定义汽车和智能网联时代, 功能安全是底线, 信息安全是防线。二者均强调系统性、全生命周期、基于风险的开发流程, 共同构成智能汽车可信运行的“双轮驱动”, 是车企技术竞争力、合规准入和用户信任的核心保障。那 Zephyr 在这两个方面有哪些特点, 是否能满足汽车行业的高要求呢?

3.3.1. 信息安全能力

Zephyr 的信息安全旨在保护设备在联网环境下的数据机密性、完整性和可用性, 防止恶意攻击。具有以下核心安全特性与机制。

(1) 安全的启动链与代码完整性

- **MCUboot:** Zephyr 官方推荐并深度集成 MCUboot 作为安全启动引导程序。它通过数字签名验证固件镜像的完整性和真实性, 防止运行被篡改的代码。
- **硬件信任根:** 支持利用芯片内的安全硬件(如 TrustZone、TPM、HSM)来安全存储密钥和进行加密操作。

(2) 加密套件与安全协议

- **丰富的加密库:** 内置了稳健的加密算法库, 包括 AES (加密)、SHA (哈希)、RSA/ECC (非对称加密) 等, 并支持硬件加速。
- **完整的网络安全协议栈:** 原生支持 TLS/DTLS (用于安全通信), 并集成了流行的库如 Mbed TLS。

(3) 安全的更新机制

- **通过 MCUboot 实现安全、可靠的固件空中升级。**支持 A/B 分区, 确保更新失败后能回滚到旧版本, 保障设备可用性。

(4) 内核级安全功能

- **内存保护单元支持:** 当硬件支持 MPU 或者 MMU 时, Zephyr 可以配置内存区域的访问权限(只读、只执行等), 防止用户态应用程序破坏内核或其他应用的内存。
- **栈溢出保护:** 内置栈保护机制(如栈边界检查), 可以有效缓解栈溢出攻击。
- **线程隔离:** 支持将线程限制在其专属的内存域中, 限制其可访问的资源, 实现权限分离。

(5) 信息安全审计与漏洞管理

- Zephyr 项目有一个活跃的信息安全委员会, 负责接收、评估和披露安全漏洞。
- 拥有明确的漏洞报告流程, 并定期发布安全公告。
- 代码库会接受持续的安全审查和静态代码分析(例如, 使用 Coverity Scan)。

总之, 信息安全是 Zephyr 的“核心优势”, 其已经具备了工业级应用所需的安全能力。车载软件在很大程度上可以复用, 而无需重新开发。

3.3.2. 功能安全能力

功能安全关注的是系统在发生故障或操作失误时, 能避免引发人身伤害、设备损坏或环境危害等风

险。Zephyr 的目标是满足国际功能安全标准。其核心特性与进展如下：

(1) 功能安全认证准备

- Zephyr 项目已通过 IEC 61508 (通用功能安全标准) SIL 3 的概念批准(Concept Approval)。正积极推进 IEC 61508 SIL3 的正式认证，同时准备 ISO 26262 ASIL-D 认证，以满足汽车电子等高安全等级应用需求。
- 项目正在为此进行系统性准备，包括建立完善的开发流程、文档和验证体系。

(2) 高可靠性内核设计

- 确定性行为：作为 RTOS，Zephyr 具有确定性的线程调度和中断响应时间，这对于安全关键型应用至关重要。
- 错误处理：内核包含多种运行时错误检测机制，如断言检查、空指针解引用检测等，能及时报告系统异常。

(3) 测试与验证

- 强大的测试框架：如前所述的 Twister 工具，用于进行大规模、自动化的回归测试，确保代码更改不会引入新的缺陷。
- 高测试覆盖率：项目对内核和关键模块有极高的代码覆盖率要求，并通过测试用例库和模拟器(如 QEMU)持续执行。
- 静态代码分析：强制使用静态分析工具，在代码合并前发现潜在缺陷。

(4) 详细的文档与可追溯性

- 功能安全认证要求严格的流程。Zephyr 项目正在完善其设计文档、需求规格说明和测试计划，确保开发过程的可追溯性。

总之，功能安全是 Zephyr 的“战略方向”，它正在系统性地为未来的国际安全认证做准备，虽然目前还不是一个现成的已认证操作系统。

4. 总结

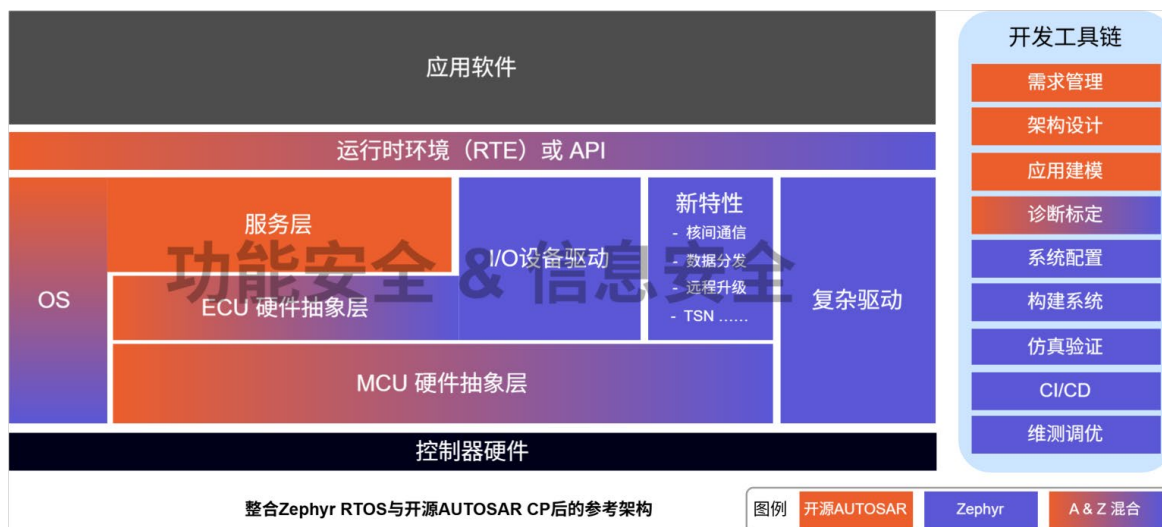


Figure 8. The reference software platform combined Zephyr and AUTOSAR CP

图 8. 整合 Zephyr RTOS 和 AUTOSAR CP 中间件的车控操作系统平台

伴随着普华开源小满、理想星环 OS 等开源汽车操作系统的出现，以 Zephyr 项目为基础，在其丰富

多样的基础功能、高效现代化的工具链体系和快速提升的安全性的基础上, 将开源汽车操作系统中相关软件模块与其整合在一起, 如图 8 所示, 从而打造一个开放、领先、安全、高效的开源智能车控操作系统的可能性正逐渐增大。尤其是当考虑到 Zephyr 正在以每月近 2000 次的提交速度高速增长时, 更让我们对未来充满期待。

参考文献

- [1] AUTOSAR 组织. AUTOSAR 和开源引发热议, 管控车用 OS 复杂度、实现降本增效走哪条路, 看这一篇就够了[EB/OL]. <https://mp.weixin.qq.com/s/0CfGiESYrU0bu6bJHp8mg>, 2025-04-09.
- [2] AUTOSAR 组织. AUTOSAR CAPI: 构建统一软件定义汽车(SDV)生态的基础[EB/OL]. <https://mp.weixin.qq.com/s/UhisFLGvpiB3 VRrdREuWA>, 2025-09-09.
- [3] 嵌入式系统专家之声. Zephyr 实时操作系统将主导嵌入式世界的五个理由[EB/OL]. <https://mp.weixin.qq.com/s/1NpxKHHtUK8izlnqYoX9oA>, 2025-05-10.
- [4] HalfCoder. Zephyr 2025 开发者大会演讲文档概览[EB/OL]. <https://mp.weixin.qq.com/s/9IcJKeSXi7XOnkwgz33cGg>, 2025-08-31.
- [5] 汽车软件开发 BBS. 汽车基础软件漫谈[2]. 汽车基础软件的定义[EB/OL]. <https://mp.weixin.qq.com/s/d5Gitj3ugiozIIHF0J3byw>, 2022-06-01.

基于提示工程的软件测试用例生成方法研究

王媛, 韩启, 张絮, 杨丰辉

航空工业第一飞机设计研究院, 陕西 西安

收稿日期: 2025年12月24日; 录用日期: 2026年1月16日; 发布日期: 2026年1月29日

摘要

软件测试用例的质量直接关系被测软件的质量。在当前大语言模型飞速发展的阶段, 人们运用提示工程等关键技术来生成软件黑盒测试用例。然而, 普遍存在生成的测试用例质量参差不齐, 尤其异常场景测试用例无法完全契合实际测试需求的问题。因此, 本文针对座舱显示控制软件进行了结构化提示词设计方法的研究。通过专家打分法以及结合软件代码缺陷检测率, 对比不同结构化提示词因子组合下的测试用例生成效果, 从而得出最佳结构化提示词, 有效提升了软件测试用例的生成质量与效率。

关键词

大语言模型, 提示工程, 黑盒测试, 座舱显示控制软件, 测试用例生成

Research on Software Test Case Generation Method Based on Prompt Engineering

Yuan Wang, Qi Han, Xu Zhang, Fenghui Yang

AVIC The First Aircraft Institute, Xi'an Shaanxi

Received: December 24, 2025; accepted: January 16, 2026; published: January 29, 2026

Abstract

The quality of software test cases directly impacts the quality of the software under test. In the current era of rapid development of large language models, prompt engineering and other key technologies are being applied to generate black-box test cases. However, a common issue is the inconsistent quality of generated test cases, particularly in abnormal scenarios, which often fail to fully meet actual testing requirements. To address this, this study focuses on the research of structured prompt design methods for cockpit display control software. By comparing the effectiveness of different structured prompt factor combinations in test case generation through expert scoring and software code defect detection rates, the optimal structured prompt is derived, effectively improving the quality and efficiency of software test case generation.

Keywords

Large Language Model, Prompt Engineering, Black-Box Testing, Cockpit Display Control Software, Test Case Generation

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

软件测试用例编写是测试工作的核心环节，测试用例的质量直接影响软件缺陷的检测率，对保障软件质量起着关键作用。软件测试技术可分为白盒测试技术和黑盒测试技术。白盒测试基于源代码结构，随着人工智能的发展，相关测试用例生成方法的研究不断涌现且成效显著，例如基于遗传算法、蚁群算法以及粒子群优化算法等生成高覆盖率的测试用例[1]。

而在黑盒测试方面，通常采用 UML 等基于模型的测试用例生成技术。对于需求繁杂且软件规模较大的座舱显示控制软件而言，通过人工绘制 UML 图来生成测试用例的方法，与传统手工编写用例相比，其工作效率并未显著提升[2]。因此，大部分座舱显示控制软件仍然采用手工编写用例的方式。然而，随着软件高频迭代升级的需求增加，测试用例的生成效率变得低下；而且测试用例的设计严重依赖于测试人员的经验积累，导致生成测试用例的质量参差不齐。

近年来，随着大语言模型(Large Language Model, LLM)的快速发展，其在文本理解、文本生成等方面表现突出，在测试领域也表现出极大的潜能。如何运用模型微调、提示工程(Prompt Engineering)和检索增强生成等 LLM 关键技术，使大语言模型更好地应用于测试领域，也是当前研究的热点方向。目前，在实际工程应用中，人们尝试编写提示词以开展基于 LLM 的黑盒测试用例生成工作。然而，大语言模型的性能极易受提示词的格式、精准度、约束等因素影响。因此，本文针对座舱显示控制软件进行结构化提示词设计方法研究，以提升 LLM 生成黑盒测试用例的效率和质量。

2. 提示工程技术研究现状

提示工程因具有无需对模型参数进行调整，便能增强大语言模型理解与推理能力的优势，广泛应用于多个业务领域。其核心在于设计离散或连续的提示模板，将任务目标转化为自然语言指令序列并输入大语言模型，有效缓解了大语言模型存在的幻觉、知识迁移等问题。

主流的提示工程技术包括无训练提示、逐步思考提示、自动提示和检索增强提示等。零样本提示和少样本提示作为无训练提示的核心技术，通过设定角色、明确任务、限制风格等优化手段，为特定领域的复杂任务提供高效简便的解决方案。思维链提示、自一致性提示和思维树提示方法，通过将复杂问题拆解为若干子问题，引导大语言模型逐步思考得到最终答案，提高了大语言模型的回答准确率。自动提示工程中的元提示利用大语言模型生成提示词，而无梯度指令提示搜索方法在轮次迭代中通过评分函数获取最优提示。检索增强提示方法通过在外挂知识库中检索相关信息，丰富输入大语言模型的上下文信息，在复杂多变的任务中同样表现出色[3]。

由于黑盒测试用例的构成元素较为复杂，包括输入数据、预期结果和操作步骤等，难以对其进行量化评估。以及结合黑盒测试用例生成的任务特点，本文基于零样本、少样本、思维链和元提示等，进行基于提示工程的座舱显示控制软件测试用例生成方法研究。

3. 本文研究内容

3.1. 研究方法

提示工程中的提示词即输入大语言模型的指令，简单无结构的指令如“请根据输入需求，帮我生成测试用例”，往往无法得到预期输出，甚至在连续两次输入同样的指令时会得到完全不同的输出。因此，设计好的提示词，应具备结构合理、指令明确、输出可控等特性。

通用的提示词模板，如“角色 - 任务 - 行动”等，在特定任务中的表现并非最佳。例如“你是一位经验丰富的测试用例设计专家，请依据输入需求，使用异常场景测试方法……生成测试用例”，实际模型输出的异常场景测试用例中，将参数显示的状态值当作连续数值进行异常场景设计，与实际测试需求不符。因此，设计提示词时需结合被测软件特点，对大模型回答进行约束或补充更详细的测试规则等。

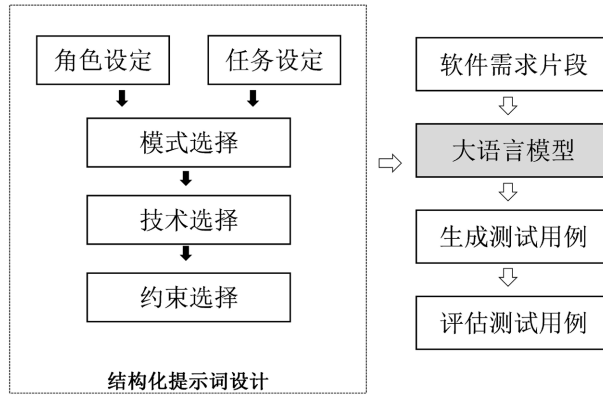


Figure 1. Research process for test case generation methods
图 1. 测试用例生成方法研究流程

座舱显示控制软件常见于汽车领域和航空领域，软件主要模块包括数据交互模块、画面逻辑管理模块[4][5]。数据交互模块，用于完成外围设备数据与软件图形界面之间的数据传输，以及用户与软件图形界面之间的数据传输，确保软件的实时响应功能。画面逻辑管理模块，用于完成图形界面内部逻辑控制，实现画面配置切换时图形界面在不同显示分区上的显示。座舱显示控制软件本质为 GUI 软件，由各类控件组成图形界面显示元素，控件的交互状态可分为状态显示和状态下发。

Table 1. Structured prompt design
表 1. 结构化提示词设计

元素	因子/关键词
角色	r: 你是一位拥有 30 年经验的软件测试专家
任务	g: 为用户给定的功能需求，生成测试用例集
技术	s1: 零样本提示 s2: 少样本提示 s3: 思维链提示 s4: 元提示
约束	cg: 运用功能分解法、等价类划分法、场景法、错误推测法和状态迁移法等方法设计测试用例
方法边界	cs1: 人机交互界面测试补充规则

续表

	cs2: 接口测试补充规则
	cs3: 边界值测试补充规则
	cs4: 异常场景测试补充规则
	cs5: 功能测试补充规则
质量约束	cq: 测试用例应满足 100%需求覆盖率, 应步骤清晰可执行, 预期结果明确, 功能划分独立、不冗余
格式约束	cf: 用例标识、名称、说明、输入数据、操作步骤、预期结果等

基于座舱显示控制软件的上述组成特点, 在结构化提示词中设计以下约束内容, 如表 1 所示。

(1) 人机界面测试补充规则: a. 外观测试, 测试元素的字体/形状、大小、颜色、; b. 布局测试, 测试元素的布局、位置、对齐; c. 画面配置测试, 测试画面配置切换时元素的外观、布局显示。

(2) 接口测试时补充规则: a. 显示数据流测试: 元素数据来源设备的各数据状态显示测试; b. 控制数据流测试: 用户操作元素为不同状态时的状态下发测试。

(3) 边界值测试时补充规则: a. 合法值测试, 包括最大值、最小值、最大值 - 步长、最小值 + 步长等; b. 非法值测试, 包括最大值 + 步长、最小值 - 步长、非法数据类型、非法小数位数、非法负值、以及空值等。

(4) 异常场景测试时补充规则: a. 数据异常测试, 测试数据的状态为故障、无效、未定义时的显示; b. 外围设备下电测试, 测试数据来源设备为下电状态时、异常下电并重启后的显示。

(5) 功能测试时补充规则: a. 测试计时功能时, 测试用例应至少包括以下场景: 1) 画面显示计时直至计时结束; 2) 计时过程中切换其他类型画面, 计时结束后再切回原画面显示; 3) 计时过程中切换其他类型画面, 计时结束前切回原画面显示; 4) 计时过程中画面切换至其他位置直至计时结束; 5) 计时过程中画面切换至其他位置, 计时结束前切回原位置显示。b. 测试叠加功能时, 测试用例应至少包括以下场景: 1) 画面叠加元素显示时取消叠加; 2) 画面叠加元素显示时, 切换为其他类型画面后再切回原画面; 3) 画面叠加元素显示时切换至其他位置显示。c. 其他特殊需求测试补充规则。

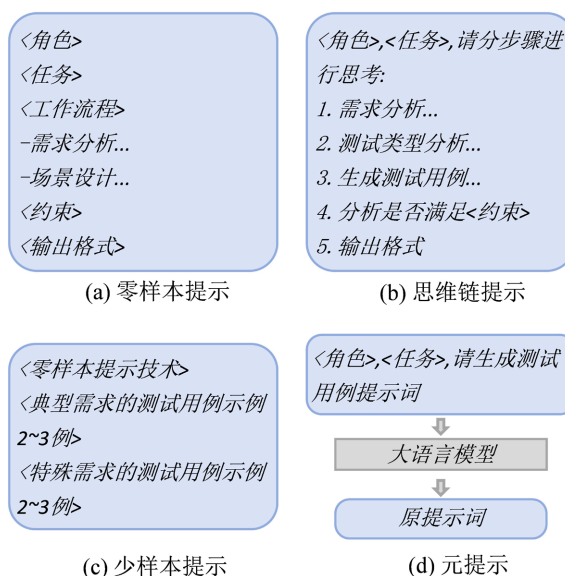


Figure 2. Explanation of structured prompting techniques
图 2. 结构化提示词技术说明

结合上述内容，本文在结构化提示词中添加以下要素：角色、任务、提示工程技术、以及针对座舱显示控制软件的任务边界、方法边界、质量约束和格式约束等。本文研究过程如图 1 所示，依据各要素选择不同的因子进行结构化提示词构建，然后对大语言模型生成的测试用例进行评估，最终得到最佳结构化提示词。构建的结构化提示词中使用的关键提示工程技术如图 2 所示。

3.2. 评估方法

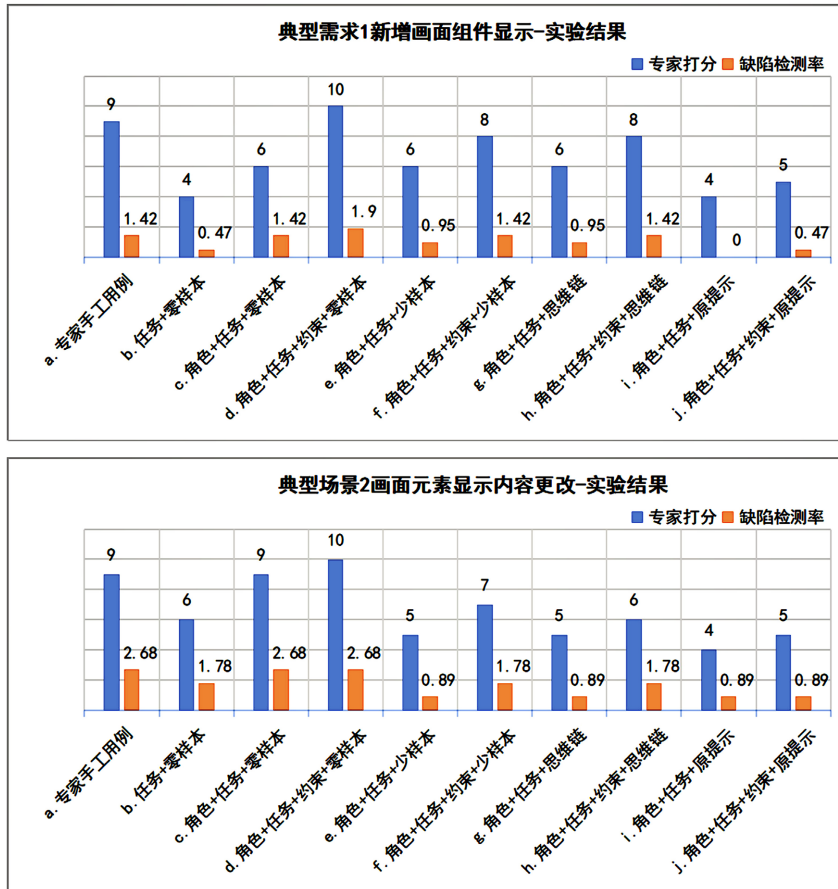
依据军用软件测试指南[6]，高质量的测试用例应满足 100%需求覆盖率，且具备完整性，即包含用例名称、用例标识、用例说明、前置条件、测试输入、预期结果和操作步骤等要素。

邀请 5 位在座舱显示控制软件测试领域经验丰富的测试专家，结合上述内容，将测试用例的评价标准划分为以下维度：a. 需求覆盖率。b. 用例完整性。c. 用例准确性。d. 异常场景覆盖率，对测试用例进行打分评价。同时，执行大语言模型生成的测试用例，对比实测结果与预期结果，结合代码更改规模得出 e. 软件代码缺陷检测率，计算公式为：软件代码缺陷检测率 = 测试问题个数/千行代码 × %。

3.3. 实验分析与讨论

以某军用座舱显示控制软件为被测对象，在其需求规格说明文档中抽取 3 类典型需求：新增画面组件显示、画面元素显示内容更改、画面元素显示逻辑更改；以及抽取 2 类特殊需求：画面新增计时显示、画面新增叠加显示。对每类需求整理 6 条需求，共 30 条需求。

由于 DeepSeek 是国内领先的大语言模型之一，其中 R1 系列专注于复杂任务的推理工作。因此，在本地部署 DeepSeek R1 32B 模型进行本文相关研究工作。



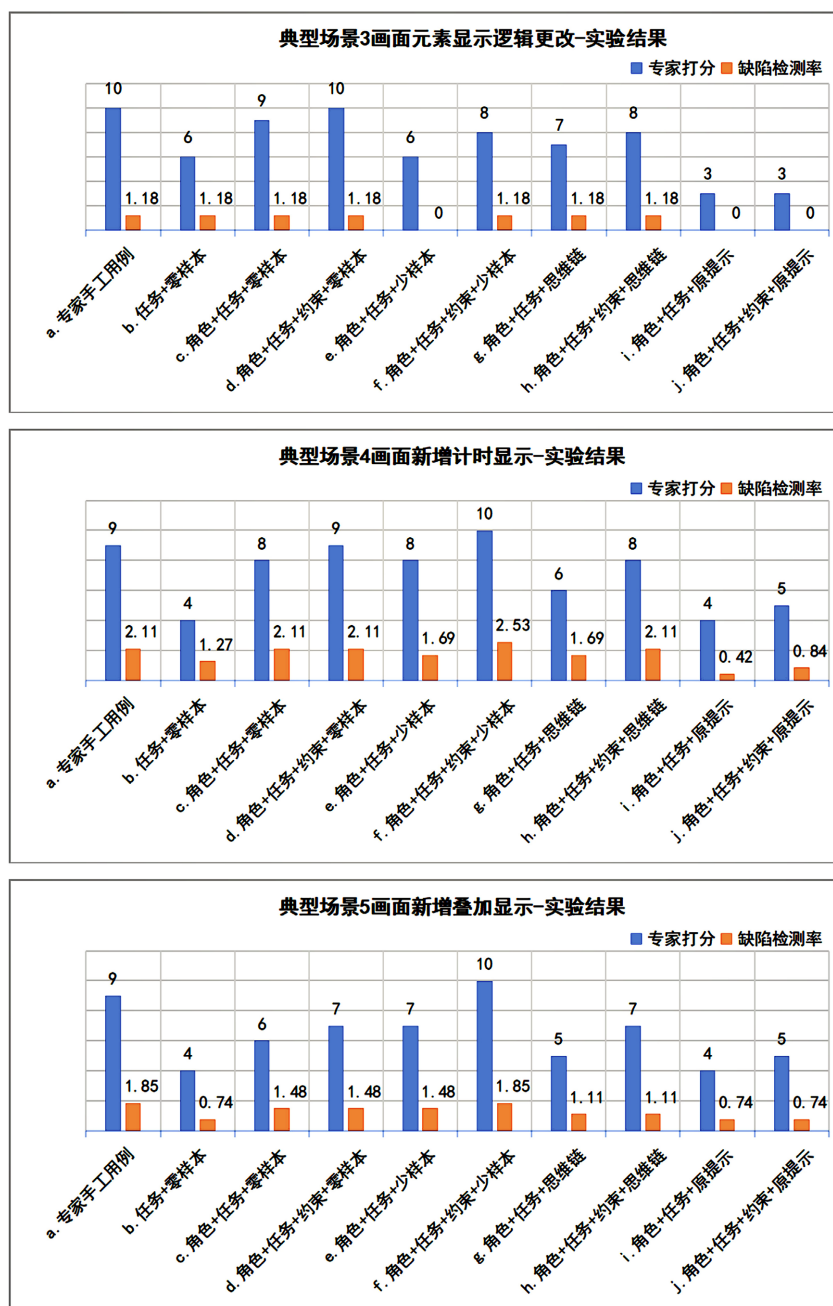


Figure 3. Comparison of test case generation results
图 3. 测试用例生成结果对比

将 30 条软件需求和结构化提示词组合因子作为大语言模型的输入，生成测试用例，专家对其进行打分评价。并且依据 30 条需求对应的软件代码更改规模 9153 行，以及执行用例得到的问题数，根据公式计算软件代码缺陷检测率，实验结果如图 3 所示，左列数据表示专家平均打分，右列数据表示软件代码缺陷检测率。

分析上述实验结果，在典型需求的测试用例生成中，“d. 角色 + 任务 + 约束 + 零样本”结构化提示词表现最佳，专家评分和软件缺陷检测率均为最高，在测试执行中发现了更多的软件问题。而在特殊需求的测试用例生成中“f. 角色 + 任务 + 约束 + 少样本”结构化提示词表现最佳。

专家手工编写用例的在所有需求场景下的评分均略低于最佳结构化提示词生成用例的评分。实验发现即使是测试专家在手工编写用例时,也会不可避免地出现笔误和格式问题,导致在用例执行时需要用例进行再次修正,增加了人力成本和时间成本。

零样本技术在典型需求的测试用例生成中表现优秀,结合约束因子能覆盖正常、边界测试场景以及绝大部分的异常场景。少样本技术因提示词中的示例具有强针对性,在特殊需求的测试用例生成中表现良好。思维链提示技术是针对复杂逻辑推理提出的方法,因此在测试用例生成领域的表现并未明显优于其他方法,在典型需求和特殊需求中表现较为均衡。而原提示技术由于提示词由大语言模型生成,其测试类型明显少于零样本等方法中设置的测试类型,存在冗余且低关联需求的用例,因而在所有需求场景中表现较差。

约束因子是结构化提示词中的关键因子。当结构化提示词中未添加约束时,所生成用例的评分和软件缺陷检测率,均明显低于有约束的结构化提示词生成效果。甚至在表现不佳的原提示技术中,添加约束时也能明显提高测试用例的生成质量。因此,设计符合软件特点的约束因子,是提高大语言模型在黑盒测试用例生成质量的关键手段。

4. 总结与展望

本文结合座舱显示控制软件的功能特点,设计该软件的结构化提示词,并在提示词中添加约束等因子。探究如何结合结构化提示词中的元素和因子信息以提高生成测试用例的覆盖率、准确率和合格率,最终得到座舱显示控制软件的最优结构化提示词。

人工对比和评估各结构化提示词的测试用例生成效果仍需耗费一定的人力,以及通过人工执行用例来评估软件缺陷检测率的方式也效率低下。在后续工作中,可与自动化测试方法相结合,自动评估生成测试用例的质量,利用反馈机制等,动态选择及生成结构化提示词,得到被测软件的最佳结构化提示词。

参考文献

- [1] 王延永,黄松. 测试用例自动生成技术综述[J]. 电子技术与软件工程, 2021(18): 51-53.
- [2] 杨波,吴际等. 一种软件测试需求建模及测试用例生成方法[J]. 计算机学报, 2014, 37(3): 522-538.
- [3] 黄峻,林飞等. 生成式 AI 的大模型提示工程: 方法、现状与展望[J]. 智能科学与技术学报, 2014, 6(2): 115-133.
- [4] 刘硕,林荣超. 综合座舱显示控制系统的设计与实现[J]. 现代电子技术, 2010, 33(15): 160-162.
- [5] 王华荣,宁伟. 综合显示控制系统的设计与实现[J]. 科技资讯, 2011(11): 36.
- [6] 中国人民解放军总装备部司令部. 军用软件测试指南: TE-BTCG-00302021 [S]. 北京: 总装备部军标出版发行部, 2021.



Call for Papers

Embedded Technology and Intelligent Systems

嵌入式技术与智能系统

国际中文期刊征文启事

<https://www.hanspub.org/journal/etis>

ISSN: 3065-1220

《嵌入式技术与智能系统》是一本开放获取、关注集成传统嵌入式技术与新兴智能系统的前沿研究最新进展的国际中文期刊，期刊特别注重软件算法、芯片设计与硬件实施的协同进展，以及理论研究与工程实践的紧密结合，面向学术界学者、产业界专家与工程师、学生及技术爱好者，关注中国领先产业集群的广阔发展潜力。本期刊强调发表原创性、创新性及具有实用价值的研究成果。该期刊由汉斯出版社出版，全球发行，现诚邀相关领域的学者投稿。

主编

何立民，北京航空航天大学教授

副主编

何小庆，嵌入式系统联谊会秘书长

吴薇，杭州电子科技大学特聘教授

投稿领域：

人工智能技术-边缘计算-端侧智能和大模型嵌入式应用
GPT-行业GPT以及GPT在嵌入式及智能系统研发中的应用
信息物理融合系统(CPS)-物联网技术-感知计算和无线传感网-泛在电力物联网-智能电表-储能技术-智能输变电
嵌入式系统结构-嵌入式操作系统与中间件-Linux、安卓和开源鸿蒙应用
实时操作系统-虚拟化和容器技术-混合关键系统
嵌入式软件形式化建模-软件测试和仿真-功能安全技术
嵌入式软件云原生技术-CI/CD和DevOpt-微服务
软硬件协同设计-开源指令集和开源芯片-RISC-V产业生态
嵌入式SoC技术--MCU 创新与生态-FPGA/DSP技术和应用
AI芯片和算法-存储技术-GPU技术-视觉芯片及嵌入式显控应用
CAN和工业总线技术-时间敏感系统-电机控制-PLC和工业PC
无线通信技术-WiFi/蓝牙/Mesh/蜂窝/5G网络-物联网安全-低功耗设计
嵌入式系统课程改革-物联网和AI教学研究-职业教育-企业人才培养
嵌入式智能系统应用（智能家居、可穿戴设备、机器人、医疗电子、汽车电子和航空航天等）

征文要求及注意事项：

1. 稿件务求主题新颖、论点明确、论据可靠、数字准确、文字精炼、逻辑严谨、文字通顺，具有科学性、先进性和实用性；
2. 稿件必须为中文，且须加有英文标题、作者信息、摘要、关键词和规范的参考文献列表；
3. 稿件请采用WORD排版，包括所有的文字、表格、图表、附注及参考文献；
4. 从稿件成功投递之日起，在2个月内请勿重复投递至其他刊物。本刊不发表已公开发表过的论文。文章严禁抄袭，否则后果自负；
5. 本刊采用同行评审的方式，审稿周期一般为5~14日。

欲了解更多信息请登录 <https://www.hanspub.org/journal/etis>

联系邮箱：etis@hanspub.org

