

# 汽车电子体系的发展-汽车控制器

## 一、汽车电子微控制器体系说明

早期的 4、8、16 位的微控制器(单片机)并没有消费级、工业级、汽车级等区别。

微控制器按照芯片内核的架构，可划分为不同的体系，这主要是跟软件的生态环境(含编译器)相关。常见的汽车 ECU 如下图所示



### 1. Infineon 体系

英飞凌体系早期生产 8/ 8051、16 位 C166 架构/内核的单片机，Tasking/Keil uv2 对 8051 及 XC2000 (C166 架构) 提供了支持，英飞凌早期的 DAVE 软件相对业界也比较先进，可惜英飞凌对软件生态环境的重视程度不够，在 keil 软件被 ARM 收购之后，英飞凌陷入了焦虑之中，向 ARM 生态靠拢，购买了 ARM Cortex-M3、Cortex-M4 内核，但由于研发的挫折以及内部原因，仅部分 Cortex-M3 内核的产品获得了汽车级认证，XMC4000 系列仍然为工业级。

英飞凌主要汽车级产品仍然为高端的 AURIX™ (based on Tricore)，收购 Cypress 意在整合其 ARM 生态系统的汽车级微控制器技术。英飞凌体系微控制器背着历史的包袱，带着整合 Cypress (赛普拉斯) 的巨大问题，在软件生态环境上，英飞凌没有自己的编译器。在当前通用中低端汽车级微控制器市场，英飞凌产品在短期内难以形成有效的竞争力 (不包括汽车直流无刷电机控制产品方向)。

#### ➤ 编译器支持

英飞凌不提供编译器支持，高版本的商业编译器 Tasking 提供了 AURIX™ 系列的完整支持。

#### ➤ MATLAB Simulink 支持

英飞凌不提供任何软件服务，第三方公司 Simulink 支持及相关工具链，非常昂贵。

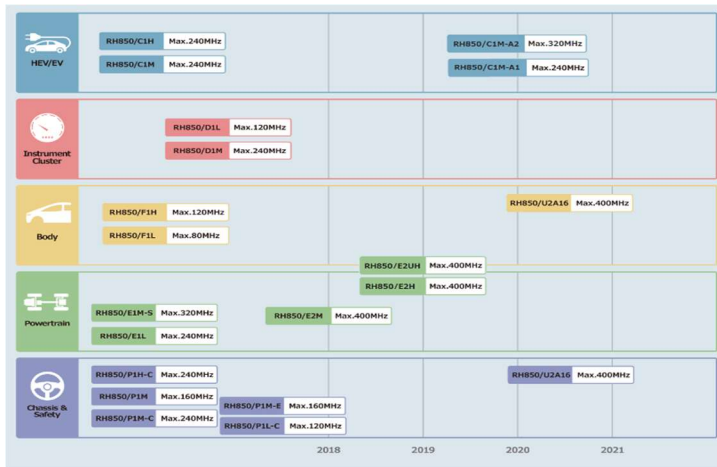
#### ➤ AUTOSAR 支持

英飞凌不提供 AUTOSAR 包，第三方公司提供 AUTOSAR 包和相关技术支持。

AURIX™ 系列是目前客车厂整车控制器，大型动力总成电机控制器等常用芯片。

## 2. Renesas 体系

RH850 Family 是瑞萨的微汽车级控制器，其发展路线图 Road Map 如下：



### ➤ 编译器支持

瑞萨 RH850 Family 是自主设计的架构，非 ARM 生态，使用的编译器为 Green Hills 。该编译器非常昂贵，一般人接触不到。

### ➤ MATLAB Simulink 支持

MATLAB Simulink 的商业编译器，目前仅有 IAR、Green Hills、Tasking 等，MDK-ARM 仍然不在支持范围内。

### ➤ AUTOSAR 支持

瑞萨不提供任何 AUTOSAR 支持包，第三方公司 Mentor 提供了瑞萨芯片 AUTOSAR 包的支持。

## 3. Power Architecture® 体系

在 2012 年，意法半导体和飞思卡尔联合发布了 Power Architecture 的汽车级微控制器，目前 ST 的 SPC 系列仍然在持续供货，飞思卡尔已被 NXP 收购，NXP 为其芯片提供了免费的 S32 Design Studio for Power Architecture® 开发环境。

### ➤ 编译器支持

NXP, S32 Design Studio for Power Architecture®

### ➤ MATLAB Simulink 支持

NXP, Simulink 库支持，工具箱插件。

### ➤ AUTOSAR 支持

NXP, 免费 AUTOSAR 包，免费 AUTOSAR 配置工具软件。

ST/NXP Power Architecture® 汽车级微控制器产品，仍在批量出货。新设计不推荐使用该体系的芯片。

## 4. Cortex-R4/R52 体系

TI (德州仪器)、Cypress (赛普拉斯) 汽车级微控制器芯片体系，由于政治原因，以及技术支持太少等缘故，该体系在汽车电子 ECU 中的应用不多，为非主流方案。

目前 Cypress (赛普拉斯) 已被英飞凌收购，ST/NXP/Infineon 的 Cortex-R52 产品可以重视，

TI 的 Cortex-R52 产品，由于政治原因，基本上可以忽略。

## 5. Cortex-M0//M3/M4/M7 体系

中低端汽车电子 ECU 主流芯片架构，目前国内外普遍采用的芯片架构，典型代表为 NXP 的 S32K 系列。

英飞凌也有一部分电机控制专用的汽车级 Cortex-M3 芯片。

## 6. 即将淘汰的产品体系

英飞凌 8/16 位汽车级微控制器 XC2000 等，NXP 的 S12 系列汽车级微控制器等产品，目前价格甚至高于低端 32 位汽车级微控制器，是即将被淘汰的产品 (目前仍有应用)。

## 二、汽车级微控制器开发环境及编译器说明

NXP S32 Platform 芯片开发环境简介:

- S32 Design Studio for ARM
  - 版本有 1.3、2.0、2018r1、2.2
  - 芯片专家系统生成初始化代码
  - 支持 JLINK 仿真器
  - 支持 32 位 ARM Cortex-M 内核汽车级微控制器
- S32 Design Studio for S32 Platform
  - 当前版本为 3.3
  - 芯片初始化代码配置工具，支持导入 AUTOSAR 包为自定义 SDK
  - 支持 JLINK 仿真器和高档仿真器
  - 支持 32bit/64bit 多种汽车级微控制器 (Cortex-A53 是 64 位 ARM 微控制器)
- IAR EW for Arm 8.40.2
  - 商业编译器，手写代码
- Green Hills
  - 商业编译器，手写代码

我们当前课程主要是针对 S32K144/S32K146 系列的芯片，其它开发环境暂不介绍。

## 三、汽车及工业微控制器的教程所涉及的内容

无论是汽车级，还是工业级的微控制器，基础功能之一是单片机的通信，这也是 Board Support Package 要解决的问题之一。视频入门教程、开发板例程涉及的内容如下：

- 基础通信功能 (Board Support Package)
  - UART 通信 (已有成熟例程)
  - SPI 通信 (已有成熟例程)
  - I2C 通信 (已有成熟例程)
  - USB 通信 (工业级微控制器有相关资源和例程)
  - CAN 通信 (已有成熟例程)
- 应用层和驱动层中间件 (FATFS、UDS 等)
- 实时操作系统 RTOS (FreeRTOS/UCOS-II 等)

## 四、主要研究课题 (研究方向)

研究课题：汽车级微控制器基础通信软件开发 (已实现)

芯片类型：S32K144/S32K146

硬件接口：UART/SPI/I2C/CAN

外设资源：GPS 模块 (非标配) / CAT1-4G 模块 (非标配) / PCF8563/W25Q128FV/TJA1043

开发板：计划研发中，支持学会软件开发的入门技能，产品级代码可嵌入到产品中。

研究课题：基于 UDS 的 CAN-boot loader 软件开发 (发送端，已实现)

芯片类型：STM32L433CC/STM32G474CE

硬件接口：USB/CAN/CANFD

开发板：正在研发中，支持 U 盘模式 MATLAB Simulink 一键下载，CAN-UDS 一键下载等功能。

研究课题：基于 UDS 的 CAN-boot loader 软件开发 (接收端)

芯片类型：S32K144/S32K146

硬件接口：CAN/CANFD

产品方案：计划开发中，支持企业实现快速产品开发，支持 CAN Boot loader 功能及加密功能。

## 五、开发板备注

出于硬件成本或软件例程等限制，NXP 提供的开发板，无法提供入门的基本知识，很多例程也跑不通。开发板需要尽可能多接入一些基本外设，多提供一些视频教程，才能让工程师有一个基本的实验硬件，按照步骤，进行通信软件的功能验证。

如果没有具体产品或项目，一个纯粹为学习研发的评估板或开发板，对于基础通信软件，实时系统的应用，也不可能涉及很深入，所以不可能让一个初学者，根据此类开发板了解基础通信软件及应用层软件，只能简单为学习而学习，学完就忘。

由于我本人长期做汽车电子产品和项目，对于一些常用外设芯片做了研究，希望通过视频教程，让更多的人掌握基本通信功能软件，不再因为资源及门槛的限制，无法获取到必要的基础软件开发技术，同时也作为自己创业的一个尝试，这就是我推出汽车电子开发板的原因。

Date: 2020/08/15

Author: [wjandsq@163.com](mailto:wjandsq@163.com)

Copyright : <http://mcu.so>

Telephone: 189-79813931