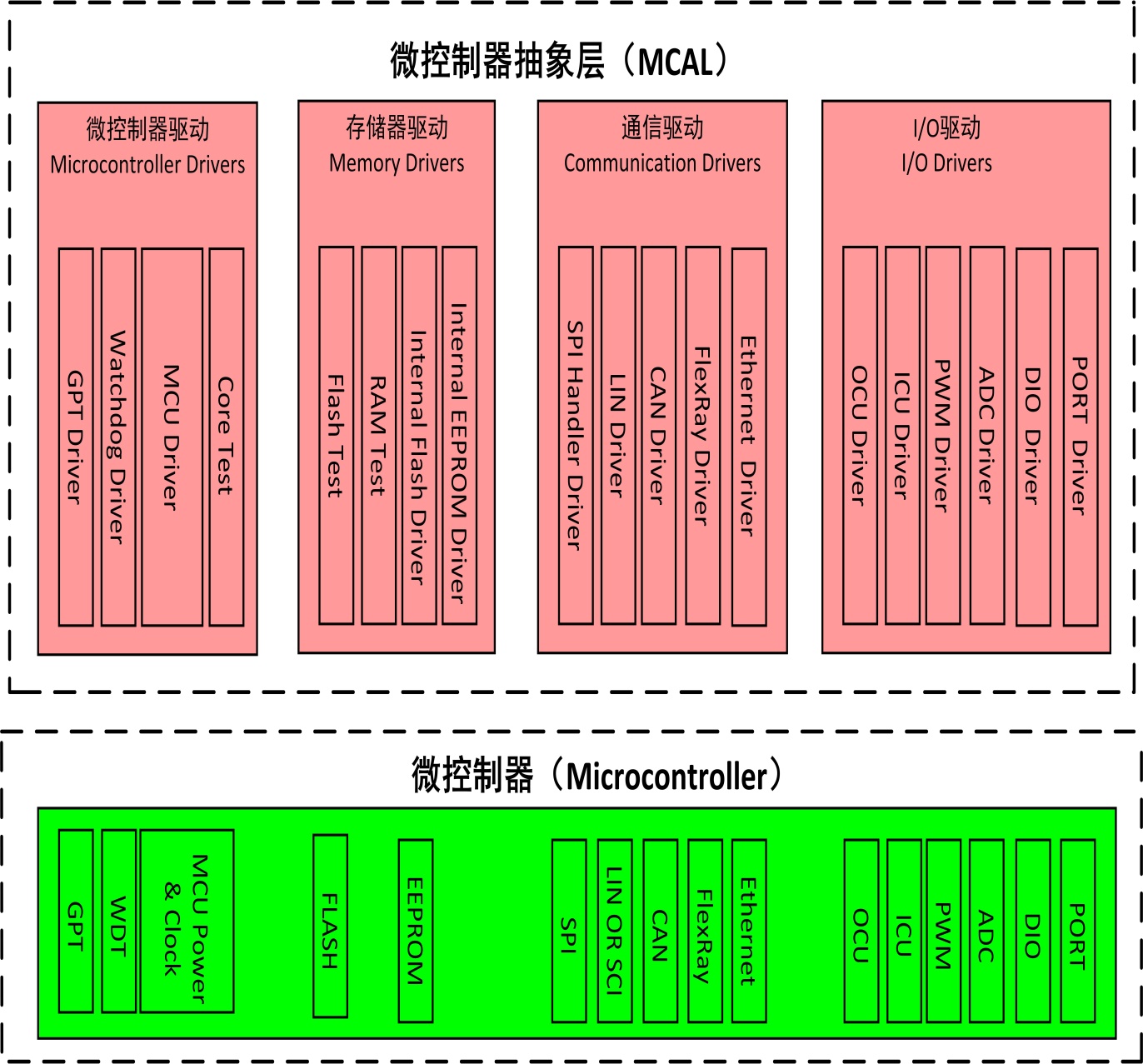
**图解AUTOSAR（五）——微控制器抽象层（MCAL）**

微控制器抽象层位于AUTOSAR分层模型中BSW最底层，包含内部驱动，可以直接访问微控制器和片内外设。更进一步地，MCAL又可以分为微控制器驱动、存储器驱动、通信驱动和I/O驱动四个部分，各部分又由具体的与微控制器硬件相对应的驱动模块组成。如下图所示。



**1、微控制器驱动**

**微控制器驱动由通用定时器驱动（General Purpose Driver,GPT Driver）、看门狗驱动（Watchdog Driver,WDG Driver）、微控制器单元驱动（Microcontroller Unit Driver,MCU Driver）和内核测试（Core Test）四个部分组成。**

**（1）GPT Driver**

在AUTOSAR中有两类定时器，操作系统定时器和硬件定时器。该模块使用通用定时器单元的硬件定时器通道，为操作系统或者其他基础软件模块提供计时功能。**GPT驱动的作用是：**

**启动和停止硬件定时器；**

**得到定时器数值；**

**控制时间触发的中断；**

**控制时间触发的中断唤醒。**

**（2）WDG Driver**

WDG Driver的功能主要是初始化和触发看门狗。WDG Driver有内部WDG Driver和外部WDG Driver。内部WDG Driver控制MCU的内部看门狗定时器，提供触发功能和模式选择服务；外部WDG Driver控制外部硬件看门狗，与内部WDG Driver一样，提供触发功能和模式选择服务。

**（3）MCU Driver**

MCU Driver位于MCAL层，可以直接访问微控制器硬件，它的主要功能是初始化、休眠、复位微控制器以及提供其他MCAL软件模块所需的与微控制器相关的特殊功能。MCU Driver还能够使能并设置MCU时钟，例如CPU时钟、外围器件时钟、预分频器等参数。

**（4）Core Test**

Core Test（内核测试）模块包含周期性测试和启动测试。内核测试模块可以对CPU所有寄存器进行测试，提供中断控制和异常检测。该模块还对算术逻辑单元、存储保护单元和缓存控制器等进行检测。

**2、存储器驱动**

**存储器驱动由内部EEPROM驱动、内部Flash驱动、RAM测试和Flash测试四部分组成。**

**（1）内部EEPROM驱动**

内部EEPROM驱动提供初始化服务，以及对内部EEPROM的读写、写、擦除等操作。该驱动模块一次只能接受一个任务。

**（2）内部Flash驱动**

内部Flash驱动提供内部Flash初始化服务，以及对内部Flash的读、写、擦除等操作。该驱动还可以将Flash访问代码下载到RAM中，如果需要的话，也可以执行写、擦除操作。

**（3）RAM测试**

RAM测试模块通过软件对RAM存储进行测试。该模块包含后台测试和前台测试。其中，后台测试是异步服务，前台测试是同步服务。

**（4）Flash测试**

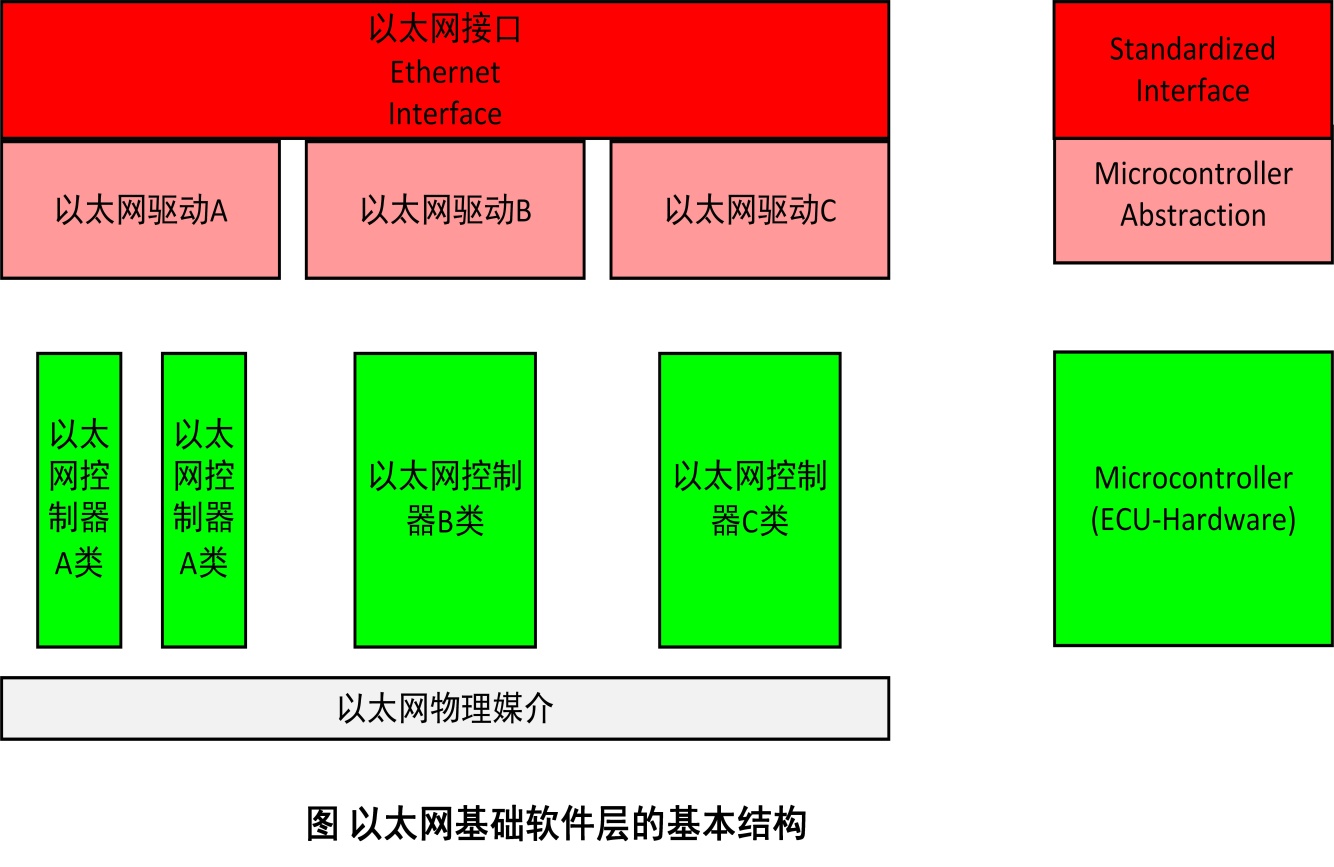
flash测试模块提供算法来测试诸如数据/程序闪存、程序SRAM等非易失性存储器，这些存储器可以是集成在微控制器内部的，也可以是外部映射到微控制器的存储器。

**3、通信驱动**

**通信驱动由以太网（Ethernet）驱动、FlexRay驱动、CAN驱动、LIN驱动和SPI驱动五部分组成。**

**（1）Ethernet驱动**

Ethernet驱动模块为以太网提供统一的接口。该驱动模块对所使用的以太网控制器的硬件特性进行了抽象。以太网接口模块使用以太网驱动层访问某些控制器。以太网驱动层由若干个以太网驱动模块组成。如下图所示。

****

**（2）FlexRay驱动**

FlexRay驱动用来抽象不同的FlexRay通信控制器及其硬件相关的特性。通信控制器的FlexRay协议强制特性经过封装后只能通过统一的API进行访问。API提供了映射到基于实际通信控制器的硬件访问序列的抽象功能操作。因此，使用FlexRay驱动可以保证FlexRay接口独立于硬件。对内部或外部FlexRay通信控制器的驱动来说，**需要进行下列处理：**

**FlexRay控制器的初始化；**

**配置数据处理单元；**

**控制指令向通信控制器的传递；**

**从协议引擎到控制器主接口状态数据的规定；**

**通信控制器和主处理机之间信息数据的传输。**

**（3）CAN驱动**

**CAN驱动针对的是微控制器内部的CAN控制器，它可以实现以下功能：**

**对CAN控制器进行初始化；**

**发送和接收报文；**

**对报文的数据和功能进行通知（对接收报文的指示、对发送报文的确认）；**

**溢出和错误处理；**

**唤醒检测。**

此外，CAN驱动还具有以下特性：单个或多个CAN通道、CAN驱动的多重实例化、对接收报文的中断/轮询模式。

CAN驱动是MCAL的一部分，可以执行硬件访问、向上层提供独立于硬件的API，而仅有的能够访问CAN驱动的上层是CAN接口（CAN Interface）。CAN驱动也可以为数据传输的初始化和通知接收事件的回调函数提供服务，该服务也是独立于硬件的。除此之外，CAN驱动也可以控制从属于同一个CAN硬件单元的CAN控制器的行为和状态。

**（4）LIN驱动**

LIN驱动使用标准的通用异步收发器（Universal Asynchronous Receiver Transmitter,UART）或者串行通信接口（Serial Communication Interface,SCI）进行通信。

**该模块可以完成下列任务：**

**LIN硬件的初始化；**

**调度表的处理；**

**LIN报文的发送（通过标志位和函数接口确认）；**

**LIN报文的接收（通过标志位和函数接口指示）；**

**睡眠和唤醒；**

**协议差错的处理；**

**报文的超时监测。**

LIN驱动也是MCAL的一部分，可以执行硬件访问、向上层提供独立于硬件的API。仅有的能够访问LIN驱动的上层是LIN接口（LIN Interface）。一个LIN驱动可以支持多个通道，但是这些通道要属于同一个LIN硬件单元。

**（5）SPI驱动**

SPI驱动模块是微控制器内部同步通信串行接口的驱动。SPI驱动为SPI总线上不同的设备（如EEPROM/Watchdog等）提供读写访问服务。一个SPI设备可以被所使用的SPI硬件和相关的片选信号识别。该模块可以在主、从或者主-从模式下运行。

**配置SPI驱动应遵循以下步骤：**

选择SPI驱动的功能级别，配置可选择的功能特性；

根据数据用途来定义SPI通道，它们可以是SPI驱动的内部缓冲器，或者是由用户提供的外部缓冲器；

根据硬件属性来定义SPI任务，它们会包含一系列使用这些属性的通道；

定义任务序列，以优先级排序的方式来传递数据。

**4、I/O驱动**

**I/O驱动由PORT驱动、DIO驱动、ADC驱动、PWM驱动、ICU驱动、OCU驱动六部分组成。**

**（1）PORT驱动**

PORT驱动初始化就是对微控制器的整个PORT模块进行初始化配置。

很多端口和管脚被分配有多种不同的功能，即可以进行引脚功能复用，比如通用I/O、模数转换、脉宽调制等功能。因此，对PORT必须有一个整体的配置和初始化，对各管脚的具体配置和使用取决于微控制器和ECU的引脚功能分配。PORT初始化数据应当尽可能高效地写到每个端口。DIO驱动中所用到的端口的配置和初始化都是在PORT驱动模块中完成的。因此，在使用DIO功能之前，应先进行PORT的初始化。

**（2）DIO驱动**

DIO驱动对微控制器硬件管脚的访问进行了抽象，除此之外，还可以对管脚进行分组。该模块通过DIO通道、DIO端口以及DIO通道组来读写数据，而且这类操作是同步的。

**（3）ADC驱动**

ADC驱动对微控制器内部模数转换单元进行初始化和控制。它可以提供启动和停止模数转换的服务，分别用来开启和禁用模数转换的触发源。

**（4）PWM驱动**

PWM驱动为微控制器PWM模块提供初始化和控制服务，可生成周期和占空比都可变的脉冲。

**（5）ICU驱动**

ICU驱动控制的是微控制器的输入捕获单元（Input Capture Unit），有两种模式：正常模式和休眠模式。

**ICU驱动可以提供一下服务：**

**信号边沿检测及通知；**

**中断唤醒；**

**周期性信号时间的测量；**

**边沿时间戳捕获；**

**边沿/脉冲计数**

**（6）OCU驱动**

OCU驱动的作用是对微控制器内部的输出比较单元（Output Compare Unit）进行初始化和控制。当计数器的值到达某个阈值时，OCU模块会自动开始比较并执行相应的操作。

**OCU驱动还可以为下列功能提供服务：**

**启动或停止输出通道；**

**设定某个阈值；**

**启用或禁用某个通道的通知函数；**

**获取计数器数值；**