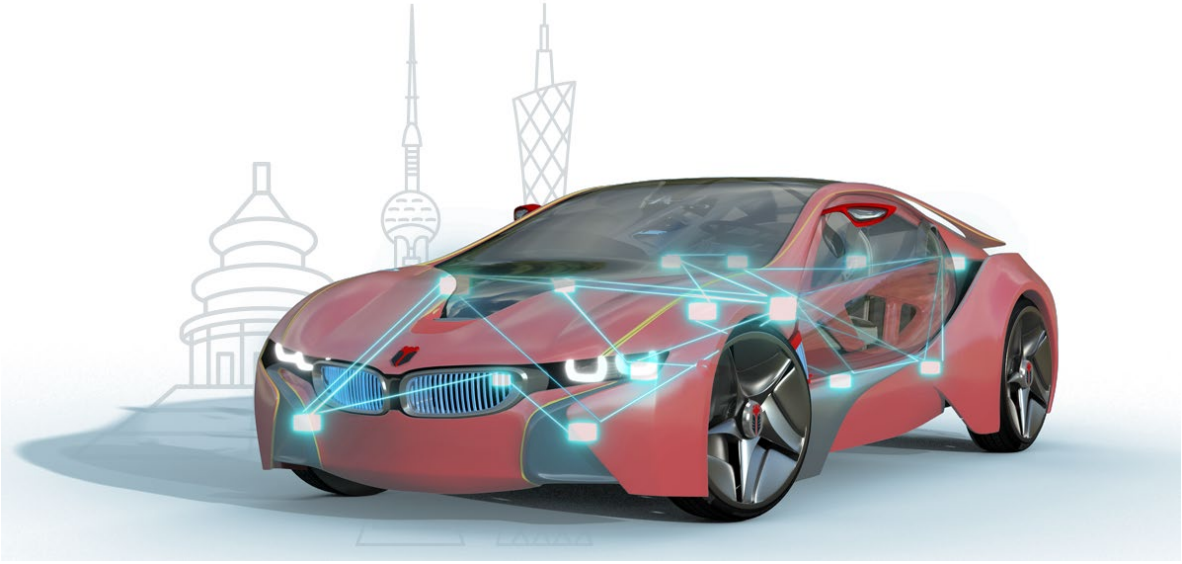
知从木牛基础软件 WDGM 在功能安全中的应用

Application of ZC.MuNiu WdgM in Functional Safety

知从木牛基础软件平台功能安全库

ZC.MuNiu Basic Software Platform SafetyLibrary



知从木牛基础软件 WDGM 在功能安全中的应用

ZC MUNIU BASIC SOFTWARE WDGM IN FUNCTIONAL SAFETY

知从木牛基础软件平台功能安全库

ZC.MuNiu Basic Software Platform SafetyLibrary

1 看门狗介绍 INTRODUCTION TO THE WATCHDOG

Watchdog，中文名为看门狗，就是为了实现在设备无人值守的情况下系统出现异常时能够自动完成系统复位操作保证整个功能的持续使用。

Watchdog, the Chinese name for the watchdog, is to realize the unattended equipment in the case of system anomalies can automatically complete the system reset operation to ensure that the entire function of the continued use.

为什么看门狗是必须的？原因是运行在硬件世界中的软件会受到各种外界因素的影响，比如：汽车上使用的诸多零部件，鉴于汽车环境的恶劣，各类 ECU 中的软件均有可能遭受如外部电磁干扰，高温等环境因素的影响，从而导致程序“跑飞”或者“死机”现象，此时如果有看门狗的存在，便可以主动触发系统复位机制保证能够再次正常使用。

Why is the watchdog necessary? The reason is that the software running in the hardware world will be affected by a variety of external factors, such as: many parts and components used in the car, in view of the harsh environment of the car, all types of ECU software may be subjected to such as external electromagnetic interference, high temperature and other environmental factors, which leads to the program “fly” or “dead” phenomenon, at this time, if the existence of the watchdog can actively trigger the system reset mechanism to ensure that it can be used normally again, rather than just watch being watched. “crash” phenomenon, at this time if the existence of the watchdog, you can take the initiative to trigger the system reset mechanism to ensure that it can be used normally again.

Translated with DeepL.com (free version)

看门狗在使用时软件必须在规定的时间间隔内向其发送特定信号，这个行为被形象地称为"喂狗"，以免 watchdog 记时超时引发系统重启。硬件看门狗一般是一块专门的芯片，它通过监控电路的电压和电流等指标来判断系统是否正常运行。在下图是 NXP 的一款 SBC 芯片，该芯片就提供了看门狗功能，在使用时需要 MCU 通过 SPI 进行看门狗初始化及喂狗操作:

Watchdog in use when the software must be in the specified time interval to send a specific signal, this behavior is figuratively known as “feed the dog”, so as to avoid watchdog timeout triggered by the system reboot. Hardware watchdog is generally a specialized chip, which monitors the circuit voltage and current and other indicators to determine whether the system is operating normally. In the following figure is a NXP SBC chip, the chip provides the watchdog function, when using the MCU through the SPI watchdog initialization and feed the dog operation:

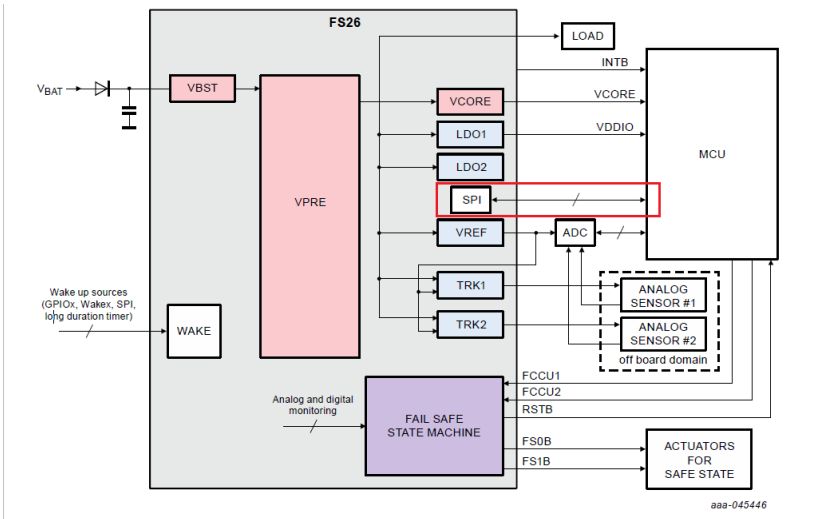


Figure 4. Example of application diagram (with VBST as a front-end regulator)

FIGURE1 EXTERNAL WDG MODULE

2 看门狗协议栈介绍 INTRODUCTION TO THE WATCHDOG PROTOCOL STACK

在汽车电子系统中，功能安全（Functional Safety）的核心目标是通过系统化的设计和安全机制，防止因电子电气系统故障导致的人身伤害或财产损失。WdgM（Watchdog Manager） 作为 AUTOSAR 标准中的关键安全模块，其设计严格遵循 ISO 26262 标准，并通过多层次的监控、容错和恢复机制来满足不同 ASIL（Automotive Safety Integrity Level）等级的要求。

In automotive electronic systems, the core objective of Functional Safety is to prevent personal injury or property damage due to electrical and electronic system failures through systematic design and safety mechanisms. WdgM (Watchdog Manager), as a key safety module in the AUTOSAR standard, is designed in strict compliance with the ISO 26262 standard and

meets the requirements of different ASIL (Automotive Safety Integrity Level) through multi-layered monitoring, fault tolerance and recovery mechanisms. The WdgM (Watchdog Manager), as a key safety module in the AUTOSAR standard, is designed in strict compliance with ISO 26262 and meets the requirements of different ASIL (Automotive Safety Integrity Level) through multi-level monitoring, fault tolerance and recovery mechanisms.

WdgM 里面主要部分是 Supervised Entity(SE) 和 CheckPoint(CP) 。

The main parts of WdgM are the Supervised Entity (SE) and CheckPoint (CP).

- SE ： 每一个 SE 可以有 Alive 、 deadline 、 Logical 三种形式的监督方式【 Logical Supervision 分为两种，同一个 SE 内部的程序流监督、不同 SE 内部的程序流监督】。一个 SE 可以是一个算法、一个函数、一个任务。

  SE: Each SE can have three forms of supervision: Alive, deadline, and Logical [Logical Supervision is divided into two types, program flow supervision within the same SE and program flow supervision within different SEs]. A SE can be an algorithm, a function, a task.

- Check Point ： 用于作为区分不同的监控方式，可以属于下面的一种或多种监控方式的类型

  Check Point : Used as a way to differentiate between different monitoring methods and can belong to one or more of the following types of monitoring methods.

2.1 周期监控 Alive Supervision

程序流监控：外部看门狗实现 WdgM 模块的 Alive Supervision 监控机制，用于监控周期性的 Task；WdgM 模块通过计算程序运行时检查点出现的个数，然后与期望值(配置信息)进行比较，如果超出容差范围，即视为违反程序流程, WdgM 模块将会记录为检查点故障。

Program Flow Monitoring: The external watchdog implements the WdgM module's Alive Supervision monitoring mechanism to monitor periodic Tasks; the WdgM module calculates the number of checkpoints that occur while the program is running and compares it with the expected value (configuration information), if it is out of tolerance, it is considered as a violation of the program flow, and will be recorded as a Checkpoint Fault by the WdgM module. The WdgM module records this as a checkpoint fault.

AEB 模块 Task 的 Alive Supervision 举例：

Example of Alive Supervision for the AEB module Task:

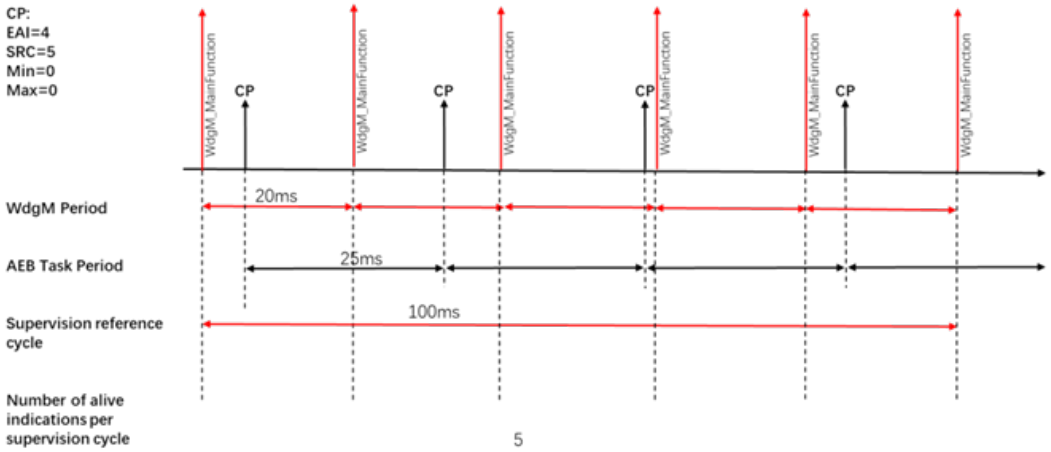


FIGURE2 ALIVE SUPERVISION

WdgM 的调用周期为 20ms，AEB 模块 Task 的周期为 25ms，如图 6.21 所示, 配置监控周期为 100ms，即：在 5 个 WdgM_MainFunction 区间，CheckPoint 到达数量为 3-5 个为正常。 如果发生的次数不在 3-5 之间 ,WdgM 模块将认为检测到 AEB 模块 Task 的周期异常，同时通过图 6.20 所示的喂狗数据流传输不喂狗请求给 PMIC 模块，PMIC 模块停止喂狗, 等待看门狗喂狗超时（超时时间 64ms*3）后，触发复位。

The call cycle of WdgM is 20ms and the cycle of AEB module Task is 25ms, as shown in Figure 6.21, the monitoring cycle is configured to be 100ms, i.e., the number of CheckPoint arrivals in the interval of 5 WdgM_MainFunction is 3-5 as normal. If the number of occurrences is not between 3-5, the WdgM module will consider that the cycle of AEB module Task is detected abnormally, and at the same time, it will transmit the request of not feeding the dog to the PMIC module through the dog feeder data stream shown in Fig. 6.20, and the PMIC module will stop feeding the dog, wait for the timeout of watchdog dog feeder (timeout time of 64ms*3), and then trigger the reset.

2.2 截止时间监控 Deadline Supervision

Deadline Supervision 需要关注某段程序运行的时间，过长或过短都说明程序执行异常。抽象为监控两个 Check Point 之间运行的时间 。 具体算法如下： 在 WdgM 中配置 Deadline Supervision 的起始 Check Point、结束 Check Point、最小时间门限和最大时间门限 。 在运行到起始 Check Point 时启动 Deadline Supervision， 获取系统时间， 在运行到结束 Check Point 时， 获取系统时间， 计算运行时间是否配置合理范围内。

注意： 两个 CP 属于相同的 SE，两个 CP 不能相同 。

Deadline Supervision is concerned with how long a program is running, whether it is too long or too short indicates that the program is executing abnormally. The abstraction is to monitor

the running time between two Check Points. The algorithm is as follows: Configure the Start Check Point, End Check Point, Minimum Time Threshold and Maximum Time Threshold of Deadline Supervision in WdgM. Start Deadline Supervision at the Start Check Point, get the system time, and at the End Check Point, get the system time, and calculate whether the runtime is within the reasonable range.

Note: Two CPs belong to the same SE and two CPs cannot be the same.

2.3 逻辑监控 Logical Supervision

Logical Supervision 主要用于监控应用程序的运行顺序是否正确，包括各个 SE 本地的运行路径的检查，和 SE 之间的全局路径检查。

Logical Supervision is mainly used to monitor whether the running order of the application is correct or not, including the checking of the local running paths of each SE, and the checking of the global paths between SEs.

比如程序按照： 1 -> 2 -> 3 -> 4 的顺序运行。程序在运行至 1 时，检查是否为第一个应该运行到的点，运行至 2 时则检查前一个到达的 check point 点是否为 1 。依次检查相邻两个点之间的实际运行顺序与配置之间的运行顺序是否匹配。当程序以 1 -> 2 -> 4 的顺序运行时，在运行至 4 处会检查到程序出错，更改 Logical Supervison 的 SE 状态。

Let's say the program runs in the order: 1 -> 2 -> 3 -> 4. When the program reaches 1, it checks to see if it is the first point it should run to, and when it reaches 2, it checks to see if the previous check point reached is 1. In turn, the program checks to see if the actual running order between the two neighboring points matches the running order between the configurations. When the program is run in the order of 1 -> 2 -> 4, a program error is checked at run 4, and the SE status of the Logical Supervison is changed.

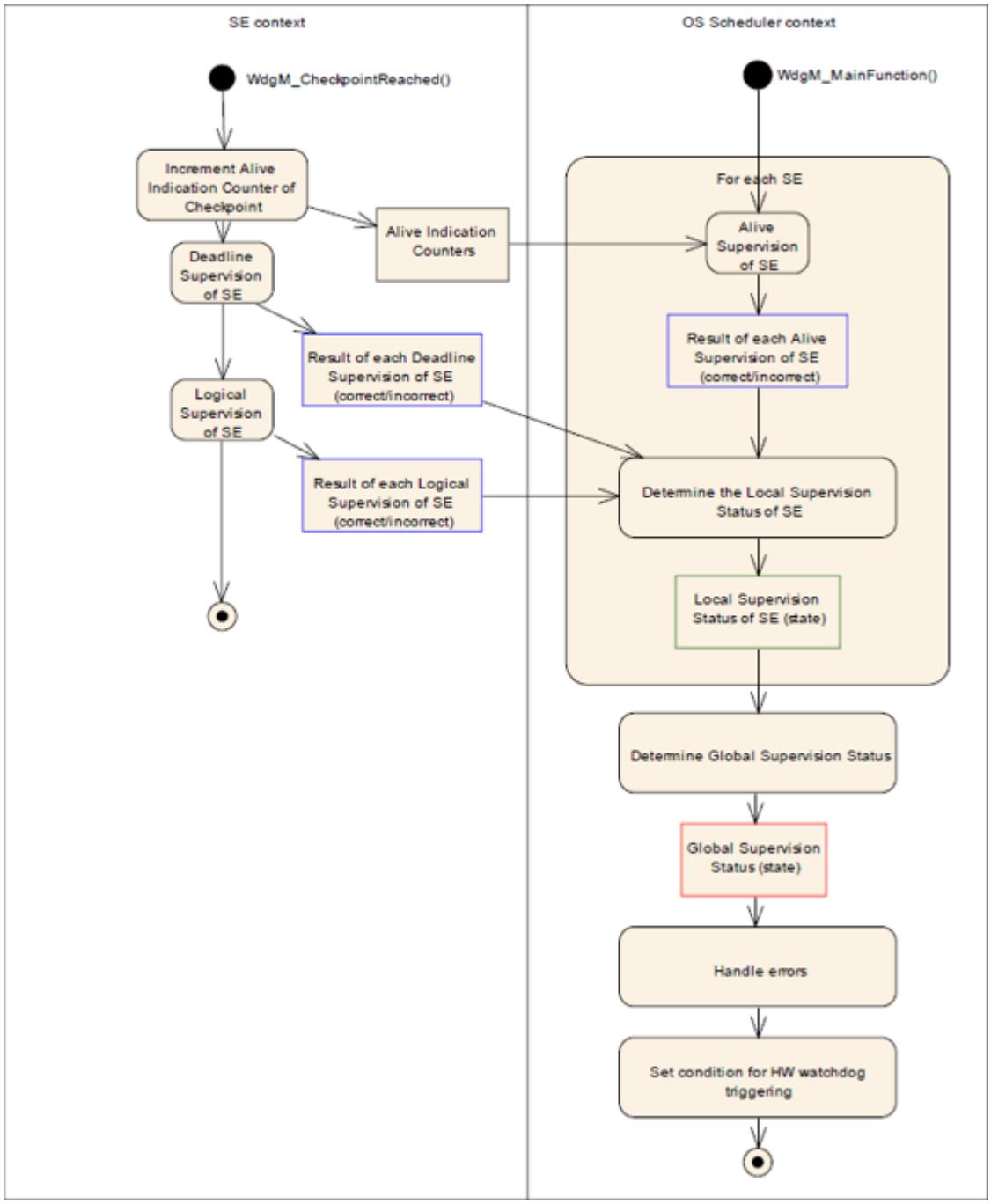


FIGURE 3: OVERVIEW OF WATCHDOG MANAGER SUPERVISION

3 知从木牛介绍 Introduction to ZC MuNiu

知从木牛配置工具基于最新 ARTOP 架构，支持最新 AUTOSAR R21-11 标准所提供的基础平台上，根据 AUTOSAR 开发方法中定义的 ECU 配置步骤，实现了从配置、验证到代码生成的 ECU 配置全流程的功能。主要优势可以总结为以下几个方面：配置、验证和代码生成全流程功能的实现，完整的实现了 AUTOSAR 开发方法中 ECU 配置阶段的开发要求。

下面将简单介绍一下使用 ZC MuNiu 进行 WdgM 模块配置的过程：

ZC MuNiu configuration tool is based on the latest ARTOP architecture, supports the latest AUTOSAR R21-11 standard provided by the basic platform, according to the ECU configuration steps defined in the AUTOSAR development methodology, realizes the ECU configuration from the configuration, validation to the code generation of the whole process of the function. The main advantages can be summarized in the following aspects: the realization of the whole process of configuration, verification and code generation has completely realized the development requirements of the ECU configuration stage in the AUTOSAR development methodology.

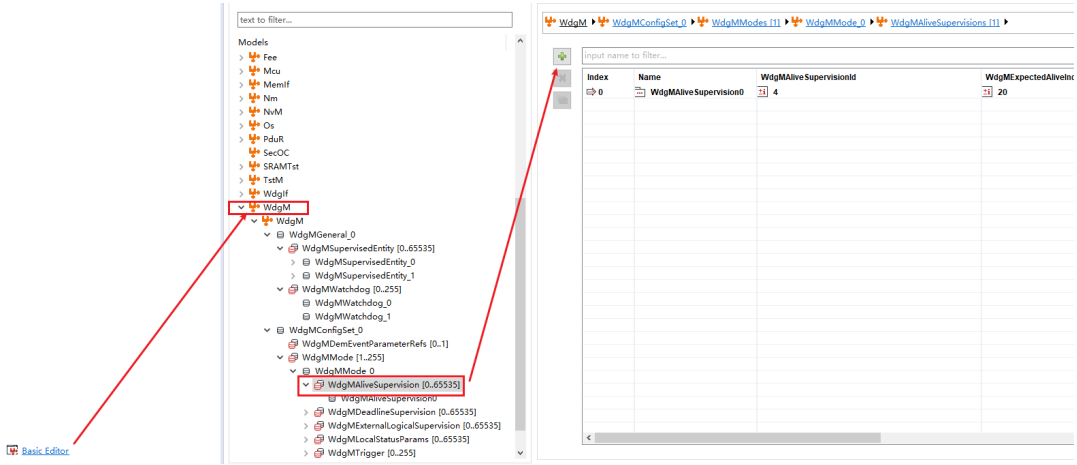
The following will briefly describe the process of WdgM module configuration using ZC MuNiu:

3.1 WdgM 配置简介 Introduction to WdgM Configuration

3.1.1 添加 SE 监控实例 ADDING SE MONITORING INSTANCES

通过在 WdgMMode 界面可以添加 Alive 监控相关的监控实例。

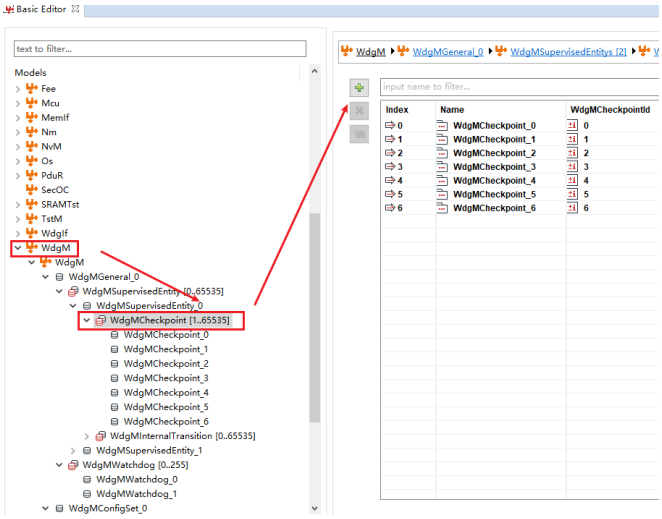
You can add monitoring instances related to Alive monitoring through the WdgMMode interface.



3.1.2 配置 SE 打卡点 CONFIGURING SE PUNCH POINTS

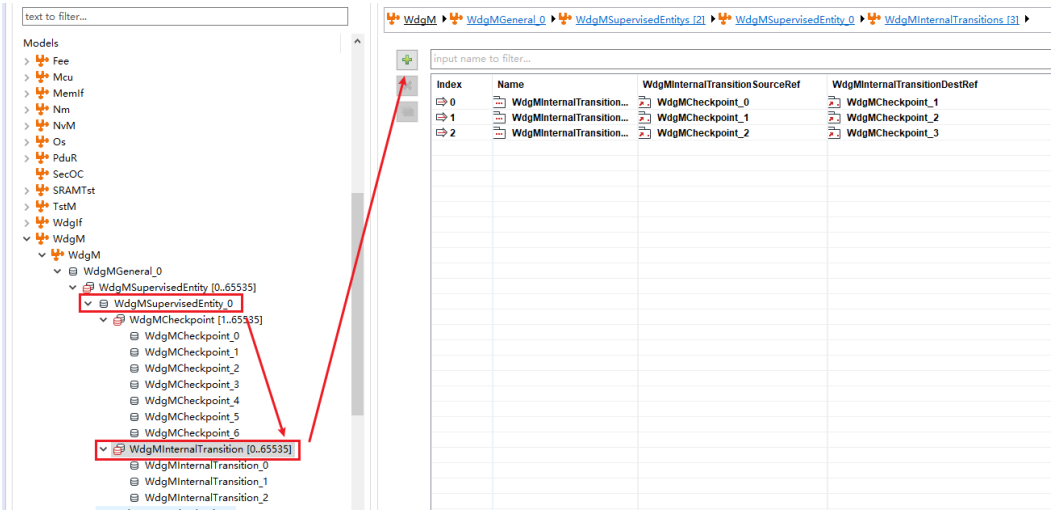
在对应的监控实例配置页面下可以添加多个 SE 打卡点。

Multiple SE punch points can be added under the corresponding monitoring instance configuration page.



另外，在 WdgMInternalTransition 页面中可以设置打卡点的执行顺序，以此来实现逻辑监控功能。

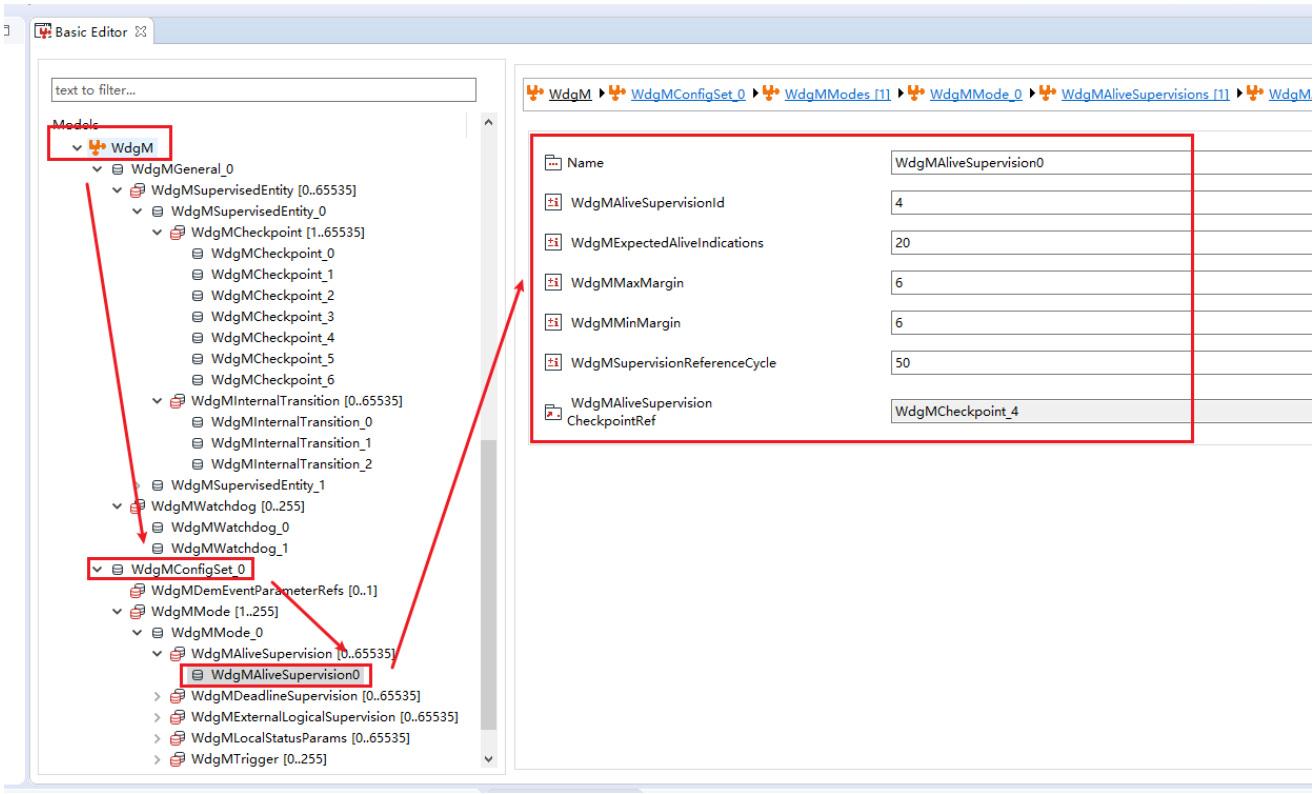
In addition, you can set the execution order of punch points in the WdgMInternalTransition page as a way to implement the logic monitoring function.



3.1.3 添加监控实例的状态参数配置 ADDING STATE PARAMETER CONFIGURATION FOR MONITORING INSTANCES

在 WdgMAliveSupervision0 页面下，可以配置 Alive 监控需要的相关参数，例如监控周期、可容忍的打卡次数限制、监控目标等等。

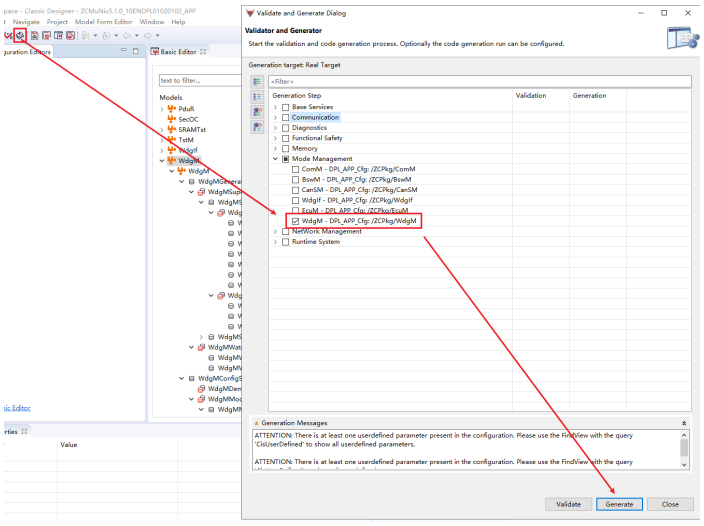
Under the WdgMAliveSupervision0 page, you can configure the relevant parameters needed for Alive monitoring, such as the monitoring period, the limit of the number of punches that can be tolerated, the monitoring target, and so on.



3.1.4 生成配置代码 GENERATING CONFIGURATION CODE

在配置完所有功能后即可生成 WdgM 相关的配置代码。

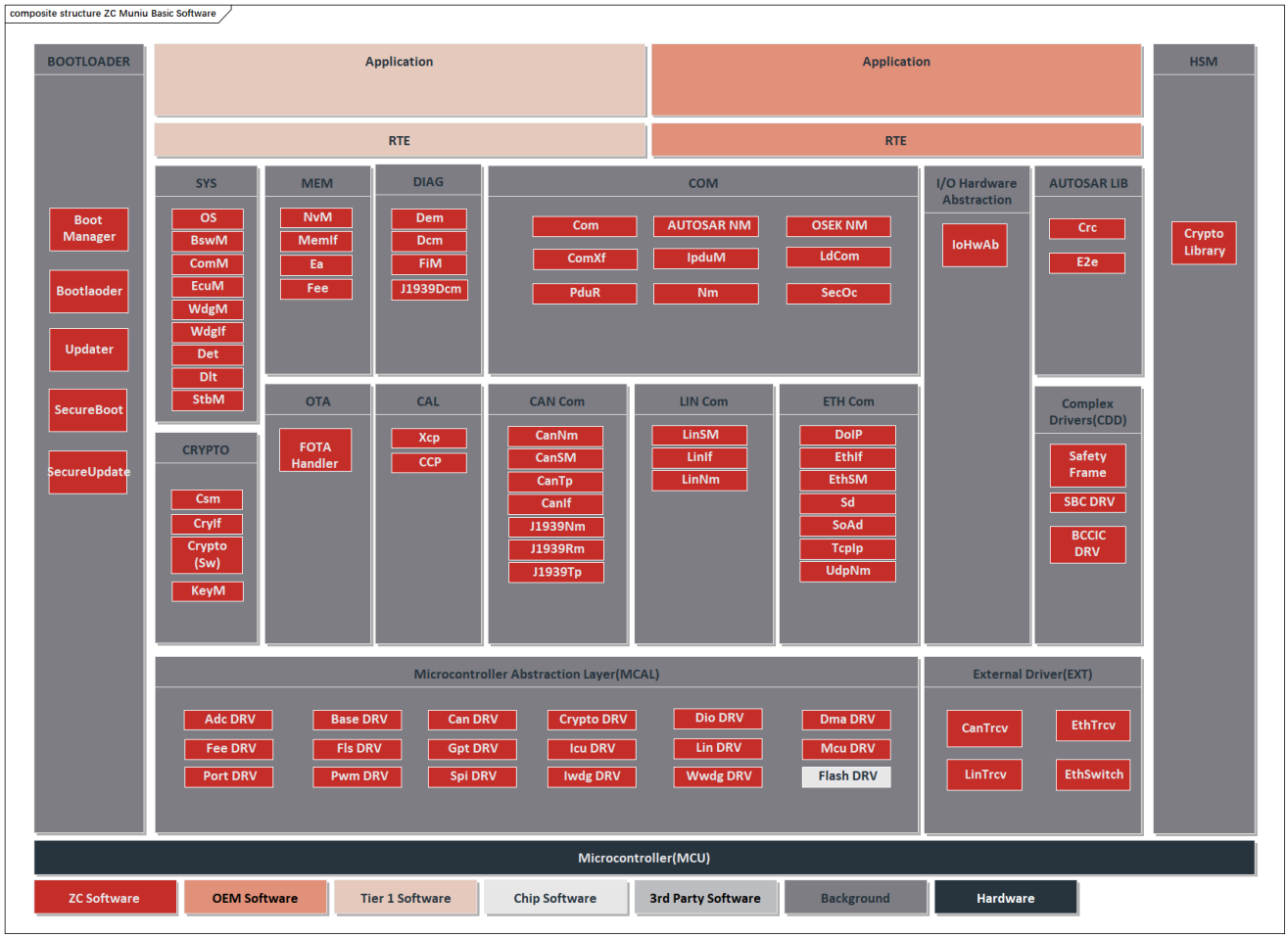
WdgM related configuration code can be generated after all functions are configured.



3.2 应用领域 Areas of application

知从木牛配置工具，给 ECU 控制器软件开发提供友好的人机界面。可以支持标准的 AUTOSAR 基础软件代码模块的配置，以及复杂驱动的配置界面开发。目前主要应用于如下场景：

The ZC MuNiu configuration tool provides a user-friendly HMI for ECU controller software development. It supports the configuration of standard AUTOSAR basic software code modules as well as the development of configuration interfaces for complex drivers. Currently, it is mainly used in the following scenarios:



- 知从木牛基础软件平台标准 AUTOSAR 模块配置
- 知从木牛基础软件平台复杂驱动模块配置
  - SAFETY FRAME
  - CRYPTO LIBRARY
  - BCCIC
  - SBC
- 同芯片企业合作，提供 MCU MCAL 的配置工具
- ZC MuNiu Basic Software Platform Standard AUTOSAR Module Configuration
- ZC MuNiu Basic Software Platform Complex Driver Module Configuration
  - SAFETY FRAME
  - CRYPTO LIBRARY
  - BCCIC
  - SBC
- Collaboration with chip companies to provide configuration tools for MCU MCALs