



文档名称: Matlab 生成与导入 Arxml 文件操作说明

编 制:	日期:	年	月	日
审 核:	日期:	年	月	日
批 准:	日期:	年	月	日

东风汽车公司

版本历史

序号	版本	修订记录	编辑	日期
1	V1.0	初版	周龙	2020.04.08
2				
3				
4				
5				
6				
7				
8				

Matlab 生成与导入 Arxml 文件操作说明

范围

本文档说明通过 matlab 生成 SWC 的 Arxml 描述文件，以及在 matlab 环境下导入 Arxml 文件生成 SWC 接口模型的操作过程。

引用文件

下列文件的条款通过在本标准的应用而成为本标准的条款。凡是注日期的引用文件，其随后所有的修改单（不包括勘误的内容）或修订版均不适用于本标准。然而，鼓励根据本标准达成协议的各方研究是否可使用这些文件的最新版本。凡是不注明日期的应用文件，其最新版本适用于本标准。

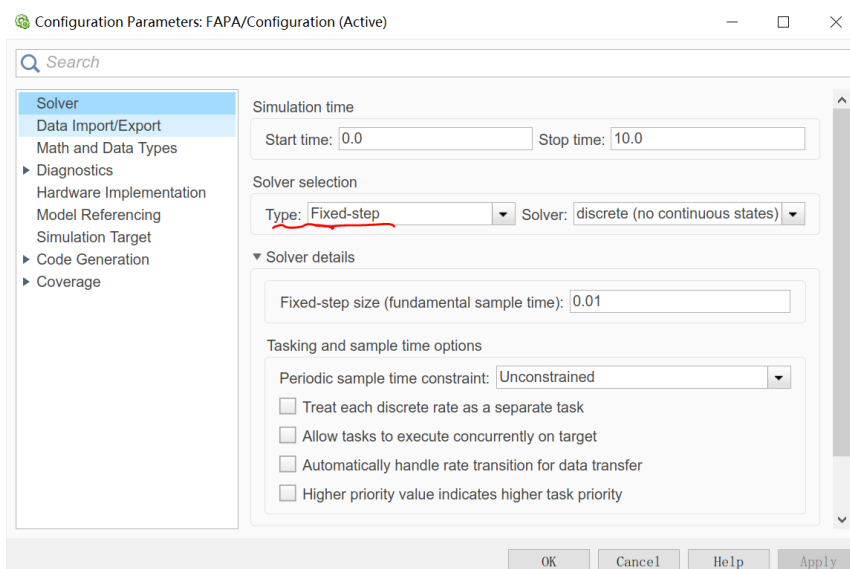
操作说明

1、 环境配置

软件工具：Matlab R2018b+Embedded Coder+Autosar Support Package

1.1. 设置求解器(Solver)

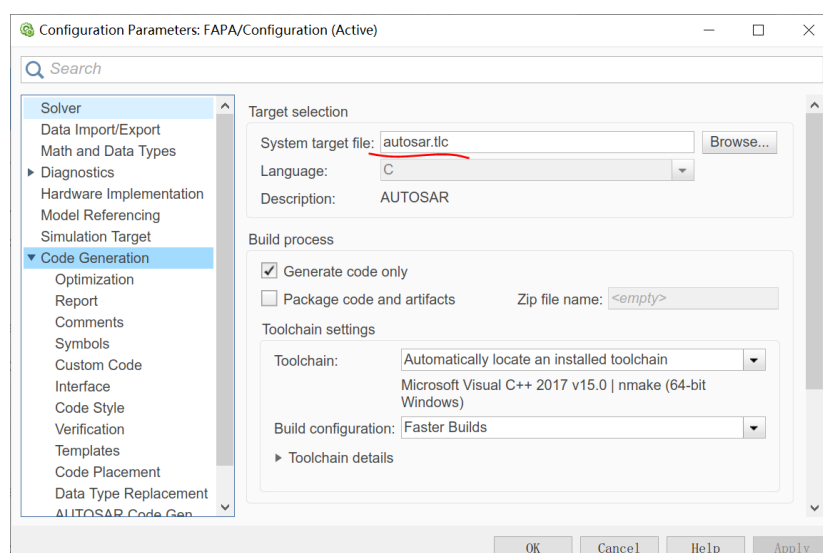
在 simulink 软件菜单栏中 Simulink-Module Configuration Paramenters 打开设置界面。
步长模式设置为定步长(Fixed-step)。



1.2. 设置目标文件及属性

菜单栏 Code-C/C++ Code-Code Generation Options。

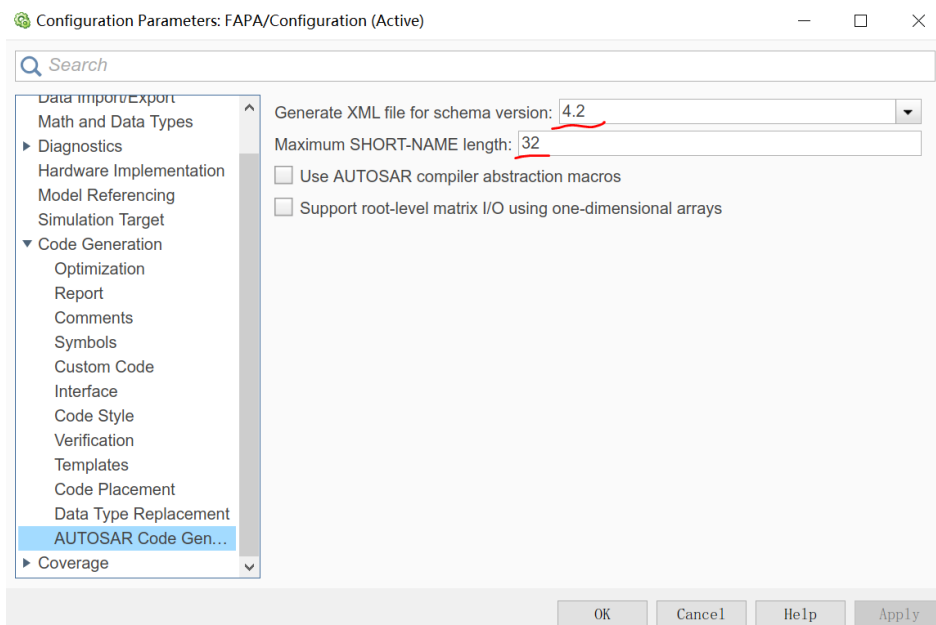
在 Code Generation 目录下将 System target file 选择为 autosar.tlc。



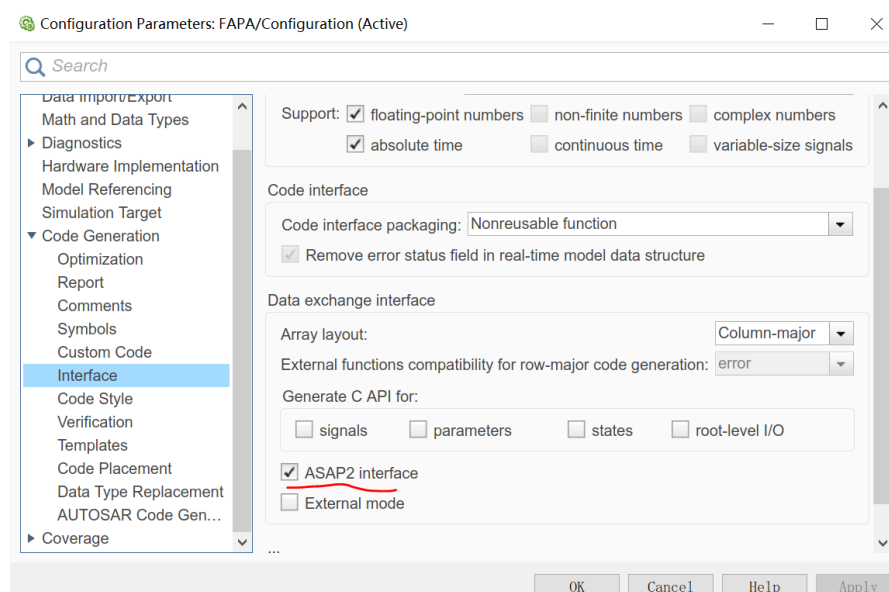
在 AUTOSAR Code Generation 配置选项中设置 autosar 版本为 4.2。

设置命名的最大长度为 32 个字符。

Use AUTOSAR compiler abstraction macros 选项是开启或者关闭 AUTOSAR 规则中所定义的一些宏，如 FUNC 等。

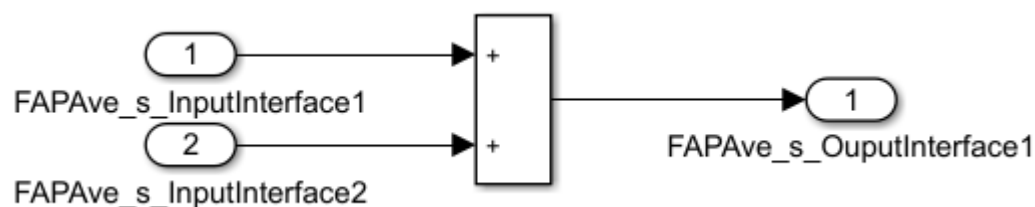


勾选 ASAP2 interface 后可生产 A2L 文件。



2、simulink 建模

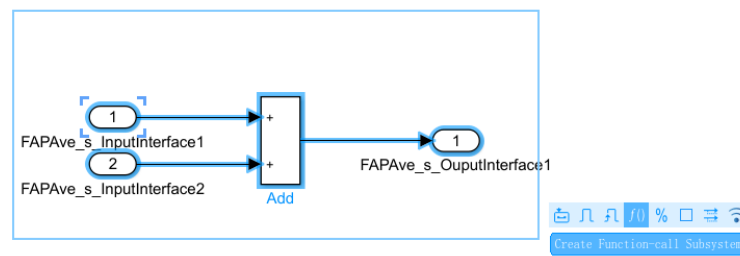
如图所示搭建了一个名为 FAPA 的模型（注意信号名和数据类型）。



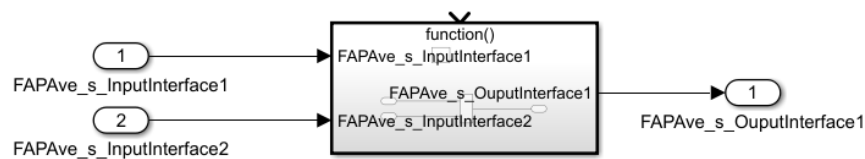
3、模型配置

3.1. 创建 Function-call Subsystem

Ctrl+A 选中模型，单击选取框右下角“...”，选择创建 Function-call Subsystem。

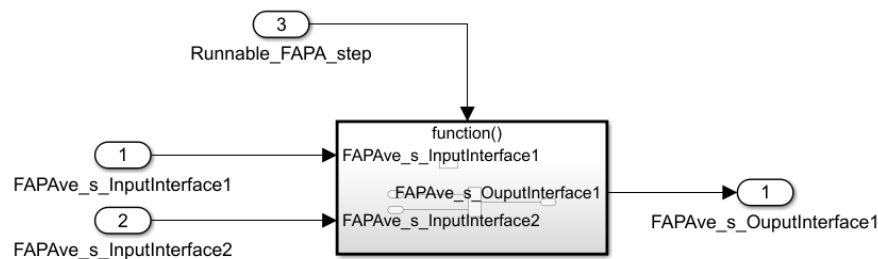


创建成功后模型如下图所示。



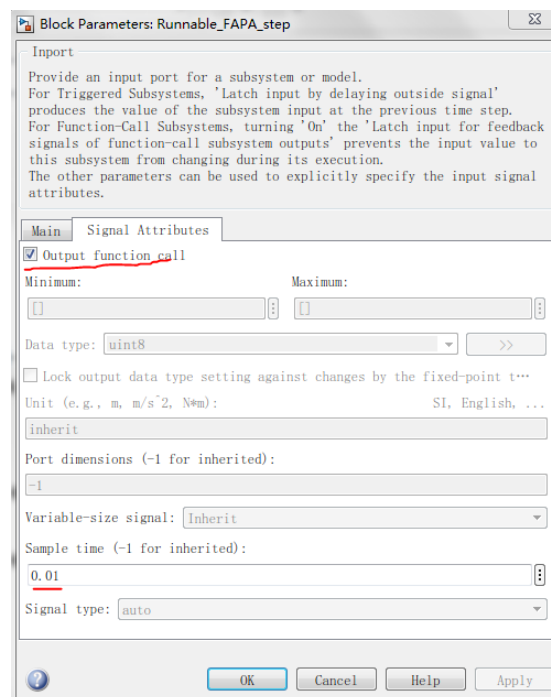
3.2. 配置 SWC 的运行周期

增加触发源，并命名为 **Runnable_(SW-Uint)_step**。根据后文来看这里S应该大写？

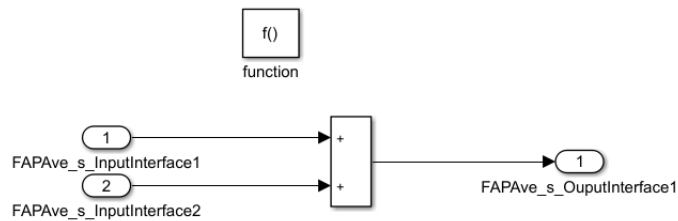


双击触发源信号进入参数配置界面，在 Sample time 中设置 SWC 的运行周期（单位为 s，此 demo 中 0.01 则表示 SWC 的运行周期为 10ms）。

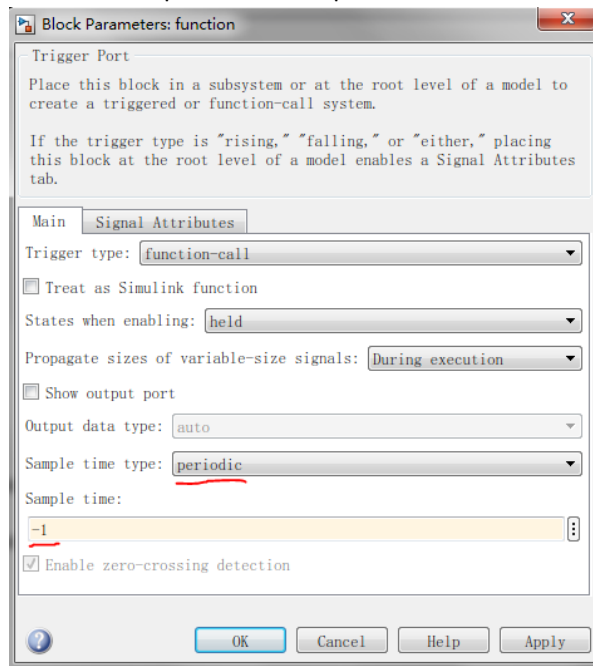
勾选“Output function call”。



双击打开 function-call subsystem，配置 f()。

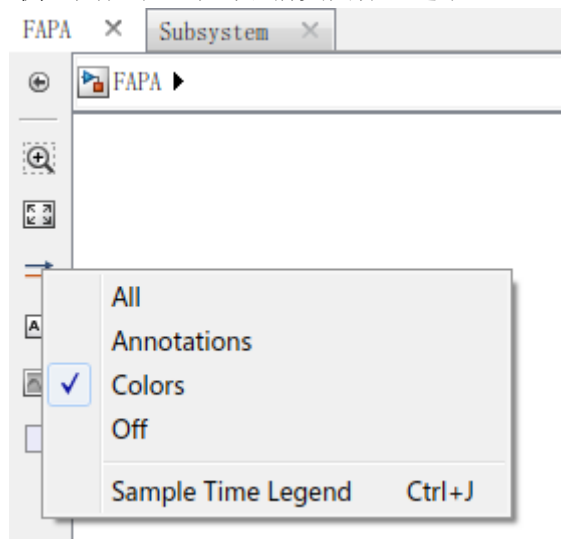


Sample time type 中选择为周期模式 periodic，Sample time 填-1。

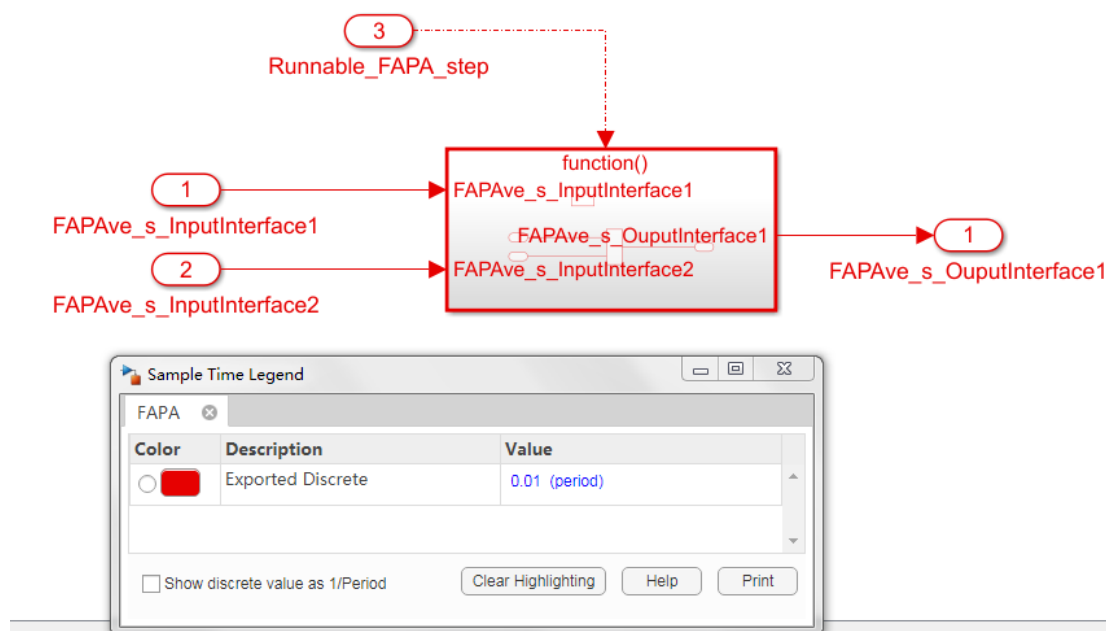


3.3. 验证是否配置成功

在模型最外层工作界面左侧工具栏中，单击双箭头图标，选中“Colors”。



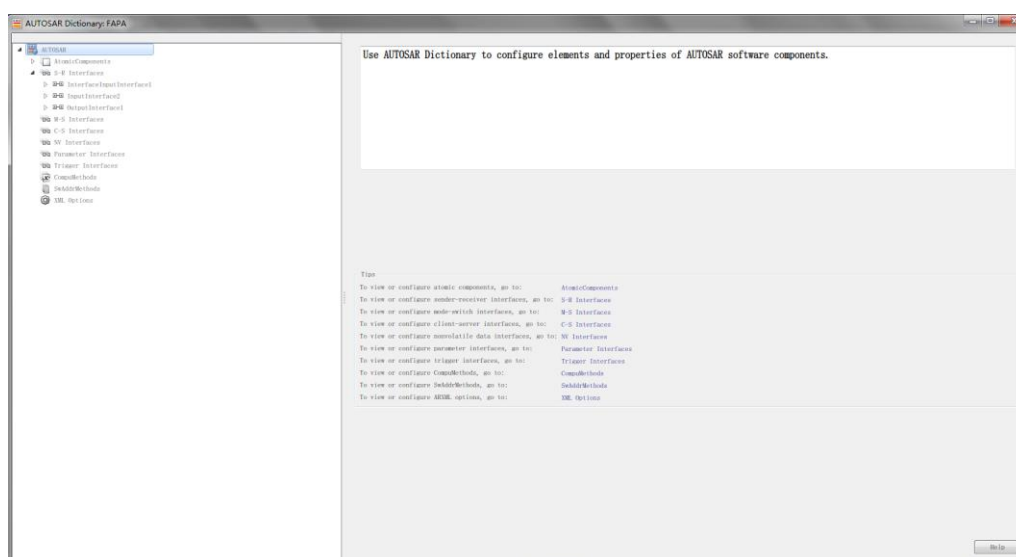
若模型颜色相同则成功配置成周期运行。



4、 Autosar 属性配置

此操作目的为在 AUTOSAR Propertise 界面完成软件组件元素的定义与相关配置。

菜单栏 **Code-C/C++ Code-Configure AUTOSAR Dictionary**。进入 AUTOSAR Dictionary 界面。



4.1. 添加 interface 及其元素 通过插件生成后，这部分直接配置完成，不需要再多做操作

目前模型中使用的均为 S-R interface，在左侧列表选中 S-R Interface，在右侧界面中增加 interface，并命名。（interface 名和信号名保持一致）

Name	IsService
FAPave_s_InputInterface1	false
FAPave_s_InputInterface2	false
FAPave_s_OuputInterface1	false

Interface 添加成功后，在左侧展开 S-R Interface，分别选中 DataElements，为 Interface 添加元素（元素名和信号名保持一致）。

AUTOSAR

AtomicComponents

FAPA

ReceiverPorts

SenderPorts

SenderReceiverPorts

ModeReceiverPorts

ModeSenderPorts

ClientPorts

ServerPorts

NvReceiverPorts

NvSenderPorts

NvSenderReceiverPorts

ParameterReceiverPorts

TriggerReceiverPorts

Runnables

IRV

Parameters

S-R Interfaces

FAPave_s_InputInterface2

DataElements

FAPave_s_InputInterface1

DataElements

FAPave_s_OuputInterface1

DataElements

?

+

-

Name	SwCalibrationAccess	DisplayFormat	SwAddrMethod
FAPave_s_InputInterface2	ReadOnly		<None>

4.2. 添加 port 并引用端口接口 4.2 部分已通过程序配置完成，不需要多做配置

ReceiverPorts 为信号输入端口，此 demo 中有 2 个输入信号，因此创建 2 个 ReceiverPorts，Ports 的命名和信号名保持一致，然后为其选中对应的 Interface。

AUTOSAR

AtomicComponents

FAPA

ReceiverPorts

SenderPorts

SenderReceiverPorts

ModeReceiverPorts

ModeSenderPorts

ClientPorts

ServerPorts

NvReceiverPorts

NvSenderPorts

NvSenderReceiverPorts

ParameterReceiverPorts

TriggerReceiverPorts

Runnables

IRV

Parameters

S-R Interfaces

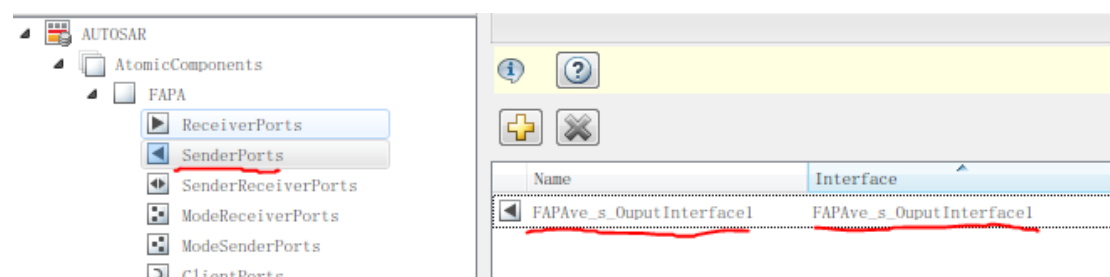
?

+

-

Name	Interface
FAPave_s_InputInterface1	FAPave_s_InputInterface1
FAPave_s_InputInterface2	FAPave_s_InputInterface2

SenderPorts 为信号输出端口，此 demo 中有 1 个输出信号，因此创建 1 个 SenderPorts，Ports 的命名和信号名保持一致，然后为其选中对应的 Interface。



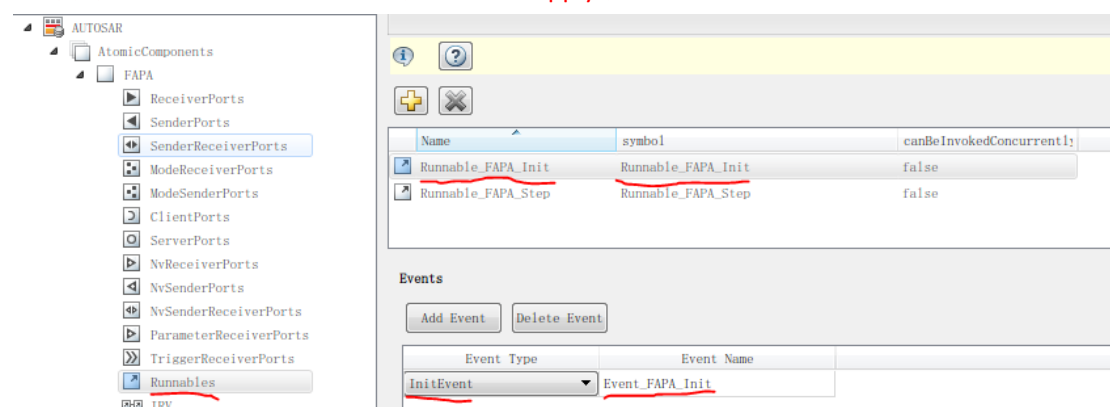
4.3. 添加 Runnable 及 Event

需创建两个 Runnable:

Runnable_(SW-Unit)_Init 为 SWC 初始化 runnable, 注意 runnable 的 Name 和 symbol 命名。

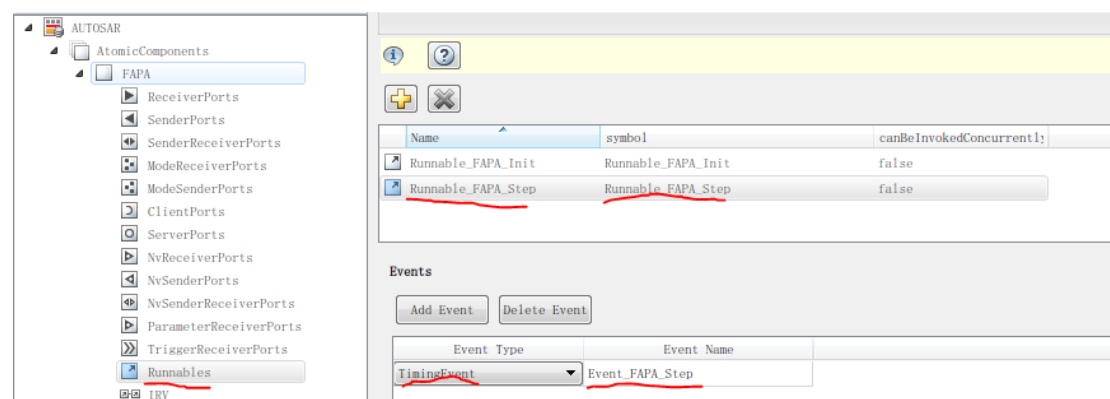
选择事件模式为 InitEvent 并命名为 Event_(SW-Unit)_Init。

(每次完成一个 Runnable 配置都需要点击 apply 按钮)



Runnable_(SW-Unit)_Step 为 SWC 周期 runnable, 注意 runnable 的 Name 和 symbol 命名。选择事件模式为 TimeEvent 并命名为 Event_(SW-Unit)_Step。

(每次完成一个 Runnable 配置都需要点击 apply 按钮)



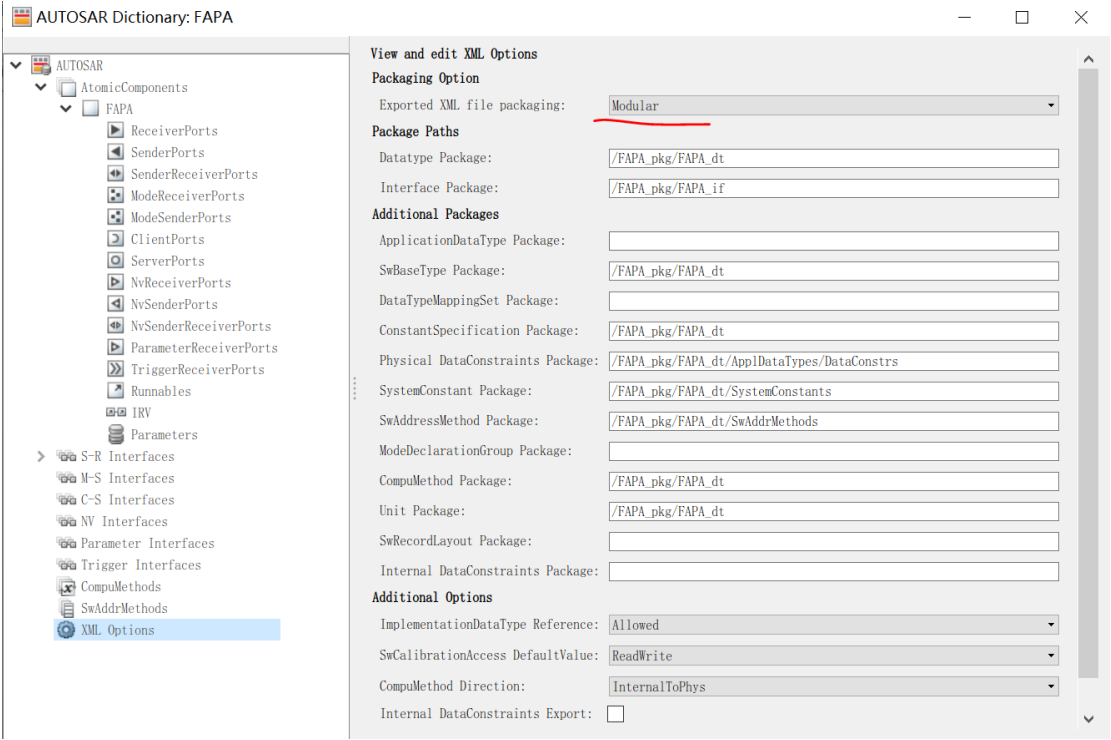
4.4. ARXML 文件选项配置

在 XML Options 配置界面中进行如下配置:

选择 Single 则生成一个 arxml 文件 (.arxml)。

选择 Modular 则生成 4 个 arxml 文件 (_dt.arxml/_if.arxml/_imp.arxml/_swc.arxml), 目前均选取 Modular。

各 package 的存放路径需和图片中设置保持一致 (注意用实际 SW-Unit 名替换 FAPA)。



5、 Simulink-AUTOSAR Mapping 配置

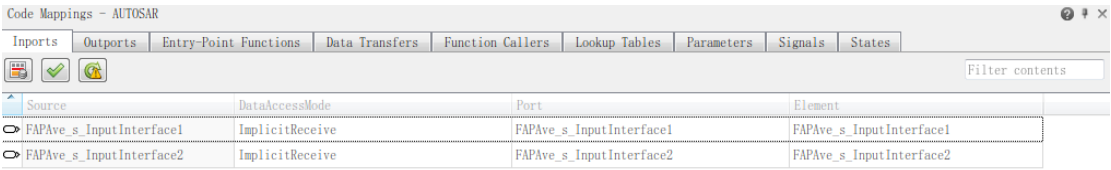
完成 AUTOSAR Dictionary 配置后再切换到 Simulink-AUTOSAR Mapping 界面完成 AUTOSAR 软件组件元素与 Simulink 元素的映射。

菜单栏 Code-C/C++ Code-Configure Model in Code Perspective 可打开 Code Mapping 配置栏。在每次完成 AUTOSAR Dictionary 更改后，都需要在 Code Mapping 配置栏中点击刷写按钮。三个按钮依次为：Dictionary、验证、刷新。



5.1. Inports 映射

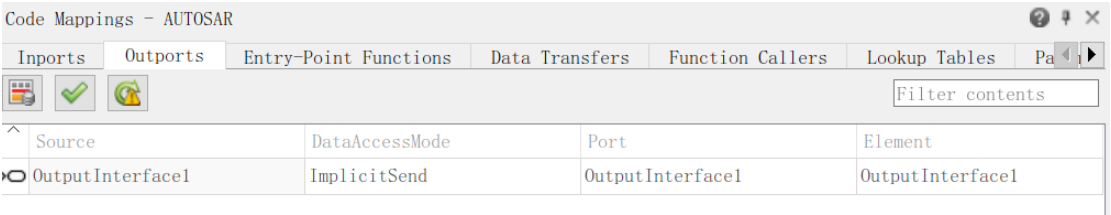
在 Inports 选项界面为输入信号配置对应的 Port 和 Element。



5.2. Outports 映射

插件工具已帮助自动配置完成，不需要另外再操作

在 Outports 选项界面为输出信号配置对应的 Port 和 Element。



5.3. Runnable 映射

Code Mappings - AUTOSAR	
Imports	Outputs
Entry-Point Functions	Data Transfers
Function Callers	Lookup Tables
Parameters	Signals
States	
Source	Runnable
Exported Function:Runnable_FAPA_step	Runnable_FAPA_Step
Initialize Function	Runnable_FAPA_Init

6、 Build

在 Simulink 主菜单中点击 Code-C/C++ Code-Build Model 选项即可生成 arxml 文件及.C/.H 文件。

Generated Code

[+] **Model files (4)**

[+] **Utility files (1)**

[-] **Interface files**

[FAPA.a2l](#)

[FAPA_component.arxml](#)

[FAPA_datatype.arxml](#)

[FAPA_implementation.arxml](#)

[FAPA_interface.arxml](#)

[+] **RTE files (6)**

7、 MATLAB 中导入 arxml 文件

按照 L2 的流程，由架构设计工具生成 Arxml 文件，算法工程师在 simulink 中导入 Arxml 文件，创建 SWC，然后将算法模型拷贝到 SWC 中，然后生成.C/.H 文件。
目前的流程是通过 simulink 直接生成 Arxml 文件，因此可以直接在 function-call 中进行模型设计，然后生成 Arxml 和.C/.H 文件。

7.1. 在 MATLAB 界面中打开 arxml 文件夹

当前文件夹

名称

html

stub

asap2.lis

buildInfo.mat

codedescriptor.dmr

codeInfo.mat

defines.txt

FAPA.a2l

FAPA.bat

FAPA.c

FAPA.h

FAPA.mk

FAPA.rsp

FAPA_canonical_dependency.list

FAPA_comp.rsp

FAPA_component.arxml

FAPA_datatype.arxml

FAPA_dwork_dependency.list

FAPA_implementation.arxml

FAPA_interface.arxml

FAPA_private.h

FAPA_ref.rsp

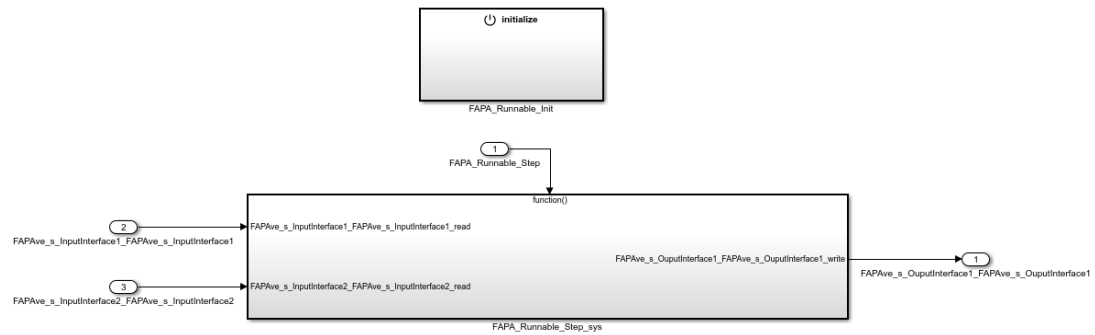
FAPA_types.h

详细信息

7.2. 输入如下指令：

```
>>arc=arxml.importer({'FAPA_component.arxml','FAPA_datatype.arxml','FAPA_implementation.arxml','FAPA_interface.arxml'})
>>createComponentAsModel(arc,'/FAPA_pkg/FAPA_swc/FAPA','ModelPeriodicRunnablesAs','FunctionCallSubsystem')
```

7.3. 导入后的模型如图所示



(接口命名为: Port 名_DataElement 名)