Document Title	TPS_ECUConfiguration: Complete Change Documentation 4.3.0 - 4.3.1
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	695

Document Status	Final
Part of AUTOSAR Standard	Classic Platform
Part of Standard Release	4.3.1

Table of Contents

1 TPS_ECUConfiguration

1.1 Specification Item TPS_ECUC_06074

Trace References:

none

Content:

Due to the hierarchical structure of the EcucParameterValues or the existence of post-build variants, it is possible that the same Referrable.shortName (in independent EcucContainer Value structures) is the base for multiple EcucParameterDef.symbolicNameValue definitions. If the respective value is equal in all occurrences of the Referrable.shortName according to TPS_ECUC_02108, the generation of the #define shall only be done once. If the respective value is different in any of the occurrences of the Referrable.shortName according to TPS_ECUC_02108, the configuration is invalid.

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #77429: Name pattern for symbolic names (TPS_ECUC_02108) lacks consideration of post-build variants

  Problem description:

  TPS_ECUC_02108 states the rule for creation of # defines for parameters with the symbolicNameValue set to TRUE. Thus rule however does not consider the fact that different post-build variants (configuration sets) may be present and thus leads to re-definitions of the respective # define.

  In order to resolve that the name pattern shall be extended to include the shortName of the respective pre-defined variant similar to TPS_ECUC_08011.

  This would yield the following name pattern:

  Mip/apiServicePrefixConf_<EcucParamConfContainerDef.shortName>_<EcucContainerValue.sh

  In case of a configuration without post-build variants, the optional suffix _<PredefinedVariant.shortName> shall be omitted.

  Agreed solution:

  Extend TPS_ECUC_06074 :
  Due to the hierarchical structure of the EcucParameterValues [or the existence of post-build variants], it is possible that the same shortName is the base for multiple symbolicNameValue definitions.....

Update Example 2.31 with variation points instead of multiple configuration containers.
Change Figure 2.31 description to: SymbolicNameValues and the generation of # defines: valid and invalid configurations [due to the existence of post-build variations]

Create a new example based on comment 8 and comment 9 :
SymbolicNameValues and the generation of # defines: valid and invalid configurations [due to the hierarchical structure of the EcucParameterValues]
–Last change on issue 77429 comment 13–

BW-C-Level:

Application	Specification	Bus
1	3	1

1.2 Specification Item TPS_ECUC_06087

Trace References:

RS_ECUC_00050

Content:

The special float values INF and -INF are allowed to be specified as EcucFloatParamDef.defaultValue of EcucFloatParamDef

RfCs affecting this spec item between releases 4.3.0 and 4.3.1:

- RfC #75110: EcuC; INF allowed for defaultValue?

  Problem description:

  Name: Robert Sakretz
  Phone:
  Role: WP-M

  Description/Motivation:

  I had an issue with using INF as defaultValue for a EcucFloatParamDef.
  Is this allowed?

  If yes we should check MMT to handle it properly.
  If no we should provide a constraint.

Agreed solution:

TPS_EcuConfiguration:

Introduce a new specification item:
[TPS_ECUC_XXXX1] INF and -INF allowed as defaultValue in EcucFloatParamDef
INF and -INF are allowed to be specified as defaultValue in EcucFloatParamDef
(RS_ECUC_00050)

MMT:
Allow INF and -INF as defaultValues in EcucFloatParamDef
–Last change on issue 75110 comment 3–

BW-C-Level:

Application	Specification	Bus
1	1	1