



EULYNX Initiative



Europe's Rail Joint Undertaking

Requirements specification for subsystem Point

Contents

1	Introduction	1
1.1	Release information	1
1.2	Impressum	1
1.3	Purpose	2
1.4	Applicable standards and regulations	2
1.5	Applicable documents	2
1.6	Terms and abbreviations	2
1.7	Variability management	2
1.8	Definition of object types	2
1.9	Modelling	2
2	Conditions of use	3
2.1	Functional Packages	3
3	Functional requirements specification	3
3.1	Subsystem - Point - General Infos and Assumptions	3
3.2	Subsystem - Point - Logical Viewpoint	3
3.2.1	Subsystem - Point - Logical Context	3
3.3	Subsystem - Point - Functional Viewpoint	4
3.3.1	Definition of time values	4
3.3.2	Subsystem - Point - Functional Context	4
3.3.3	Subsystem - Point - Functional Partitioning	70
3.3.4	Subsystem - Point - Functional Architecture	71
3.3.5	Subsystem - Point - Functional Entities	74
3.4	Subsystem - Point - Interfaces	95
3.4.1	SCI-P (Subsystem - Electronic Interlocking)	95
3.4.1.1	SCI-P - Logical Viewpoint	95
3.4.1.1.1	SCI-P - Logical Context	95
3.4.1.2	SCI-P - Information Flows	96
3.4.1.3	SCI-P - Functional Viewpoint	97
3.4.1.3.1	SCI-P - Functional Partitioning	97
3.4.1.3.2	SCI-P - Functional Architecture	98
3.4.1.3.3	SCI-P - Functional Entities	99
3.4.2	SMI-P (Subsystem - Maintenance and Data Management)	108
3.4.3	SDI-P (Subsystem - Maintenance and Data Management)	108
3.4.4	SSI-P (Subsystem - Security Services Platform)	108
3.4.5	P4 (Basic Data Identifier)	108
3.4.6	P1 (Maintainer)	108
3.4.7	P3 (Point machine)	109
3.4.7.1	Interpretation tables of 4-wire patterns	110
4	RAMSS requirements	112
5	Technical Requirements	112
5.1	Specific technical interface requirements	112
5.1.1	Interface to the Point of Service Signalling (PoS-Signalling)	112
5.1.2	Interface to the point machine	112
5.2	Time behaviour	112
5.2.1	Response times	112
5.3	Configuration and engineering data	113
5.3.1	Specific data	113
5.3.2	Value configuration	113

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.1	Head	1 Introduction		
Eu.P.2	Head	1.1 Release information		
Eu.P.3	Info	[Eu.Doc.36] Requirements specification for subsystem Point CENELEC Phase: 4 Version: 4.4 (0.A) Approval date: 29.05.2024		
Eu.P.3032	Info	Version history		
Eu.P.6591	Info	version number: 4.0 (0.A) date: 16.05.2022 author: Andreas Staudte model version: 18 Generic interface and subsystem requirements version: 4.0 (0.A) Generic interface and subsystem requirements for SCI version: 1.0 (0.A) review: CCB changes: EUP-383, EUP-384, EUP-385, EUP-386, EUP-387, EUP-391, EUP-393		
Eu.P.6701	Info	version number: 4.1 (0.A) date: 17.04.2023 author: Philipp Wolber, Filip Giering model version: 21 Generic interface and subsystem requirements version: 4.0 (1.A) Generic interface and subsystem requirements for SCI version: 1.0 (1.A) review: changes: EUP-396, EUP-397, EUP-389, EUP-405, EUP-406, EUP-409, EUP-412, EUP-413, EUP-415, EUP-416, EUP-418, EUP-421, EUP-429, EUP-431, EUP-433, EUP-434, EUP-436, EUP-437, EUP-441, EUP-442, EUP-443, EUP-444, EUP-445, EUP-446, EUP-448, EUP-449, EUP-451, EUP-452		
Eu.P.7097	Info	version number: 4.2 (0.A) date: 11.05.2023 author: Philipp Wolber, Filip Giering, Dominik Smajgl model version: 22 Generic interface and subsystem requirements version: 4.0 (1.A) Generic interface and subsystem requirements for SCI version: 1.0 (1.A) review: cluster changes: EUP-453, EUP-454, EUP-455, EUP-456, EUP-458, EUP-460, EUP-461, EUP-462, EUP-463, EUP-469, EUP-470, EUP-472, EUP-475, EUP-476, EUP-481		
Eu.P.7153	Info	version number: 4.2 (1.A) date: 01.06.2023 author: Philipp Wolber, Filip Giering, Dominik Smajgl model version: 22 Generic interface and subsystem requirements version: 4.0 (1.A) Generic interface and subsystem requirements for SCI version: 1.0 (1.A) review: cluster changes: EUP-457, EUP-459, EUP-467, EUP-468, EUP-471, EUP-473, EUP-478, EUP-479, EUP-482, EUP-483, EUP-488, EUP-489, EUP-491, EUP-493		
Eu.P.7198	Info	version number: 4.3 (0.A) date: 28.06.2023 author: Philipp Wolber, Filip Giering model version: 22 Generic interface and subsystem requirements version: 4.0 (3.A) Generic interface and subsystem requirements for SCI version: 1.0 (3.A) review: TACS Mirror Group changes: EUP-502, EUP-503, EUP-504, EUP-505, EUP-506, EUP-507, EUP-509, EUP-510, EUP-513		
Eu.P.7241	Info	version number: 4.3 (1.A) date: 15.12.2023 author: Philipp Wolber, Filip Giering model version: 25 Generic interface and subsystem requirements version: 4.0 (4.A) Generic interface and subsystem requirements for SCI version: 1.0 (4.A) review: M&T changes: EUP-417, EUP-419, EUP-485, EUP-500, EUP-514, EUP-515, EUP-516, EUP-518, EUP-519, EUP-529, EUP-530, EUP-532, EUP-535, EUP-536, EUP-537, EUP-538, EUP-539, EUP-540, EUP-541, EUP-542, EUP-543, EUP-544		
Eu.P.7369	Info	version number: 4.3 (2.A) date: 21.03.2024 author: Philipp Wolber, Filip Giering model version: 26 Generic interface and subsystem requirements version: 4.0 (4.A) Generic interface and subsystem requirements for SCI version: 1.0 (4.A) review: cluster changes: EUP-486, EUP-496, EUP-499, EUP-501, EUP-531, EUP-533, EUP-545, EUP-546, EUP-547, EUP-548, EUP-550, EUP-551, EUP-552		
Eu.P.7454	Info	version number: 4.4 (0.A) date: 18.06.2024 author: Philipp Wolber, Filip Giering model version: 26 Generic interface and subsystem requirements version: 4.0 (6.A) Generic interface and subsystem requirements for SCI version: 1.1 (0.A) review: TACS Mirror Group changes: EUP-554, EUP-555, EUP-556, EUP-557, EUP-558, EUP-559, EUP-571, EUP-573		
Eu.P.7	Head	1.2 Impressum		

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.8	Info	<p>Publishers: Europe's Rail Joint Undertaking https://rail-research.europa.eu</p> <p>EULYNX Initiative https://eulynx.eu/</p>		
Eu.P.9	Info	<p>Responsible for this document: EU-Rail System Pillar Trackside Assets Control and Supervision domain</p>		
Eu.P.3038	Info	<p>This document is drafted by and belongs to EU Rail.</p> <p>EU Rail encourages the distribution and re-use of this document, the technical specifications and the information it contains. EU Rail holds several intellectual property rights, such as copyright and trade mark rights, which need to be considered when this document is used.</p> <p>EU Rail authorizes you to re-publish, re-use, copy and store this document without changing it, provided that you indicate its source and include the following mention [EU Rail trade mark, title of the document, year of publication, version of document].</p> <p>EU Rail makes no representation or warranty as to the accuracy or completeness of the information contained within these documents. EU Rail shall have no liability to any party as a result of the use of the information contained herein. EU Rail will have no liability whatsoever for any indirect or consequential loss or damage, and any such liability is expressly excluded.</p> <p>You may study, research, implement, adapt, improve and otherwise use the information, the content and the models in this document for your own purposes. If you decide to publish or disclose any adapted, modified or improved version of this document, any amended implementation or derivative work, then you must indicate that you have modified this document, with a reference to the document name and the terms of use of this document. You may not use EU Rail's trade marks or name in any way that may state or suggest, directly or indirectly, that EU Rail is the author of your adaptations. EU Rail cannot be held responsible for your product, even if you have used this document and its content. It is your responsibility to verify the quality, completeness and the accuracy of the information you use, for your own purposes.</p>		
Eu.P.10	Head	1.3 Purpose		
Eu.P.11	Info	The purpose of the document is the specification of requirements for the Subsystem - Point.		
Eu.P.12	Info	This document describes functional, non-functional and technical requirements for the Subsystem - Point and functional requirements for interface SCI-P.		
Eu.P.13	Info	<p>This document is intended for the following users:</p> <ul style="list-style-type: none">• safety authorities• infrastructure managers• safety assessors• signalling system suppliers• validators		
Eu.P.14	Info	This document is the basis for the implementation by the supplier and for approval by the infrastructure manager.		
Eu.P.7144	Info	This document is applicable for both the EU-Rail System Pillar target architecture and the EULYNX architecture. The document is delivered as a single specification fitting both the System Pillar documentation sets and the EULYNX documentation sets. EU-Rail System Pillar is the technical authority for this document.		
Eu.P.15	Head	1.4 Applicable standards and regulations		
Eu.P.314	Info	The applicable standards and regulations used in EULYNX are listed in the EULYNX Reference Document List [Eu.Doc.12].		
Eu.P.35	Head	1.5 Applicable documents		
Eu.P.36	Info	The current versions of documents used as input or related to this document are listed in the EULYNX Documentation Plan [Eu.Doc.11]. The relationships between the documents are displayed in the Appendix A1 Documentation plan and structure [Eu.Doc.11_A1].		
Eu.P.51	Head	1.6 Terms and abbreviations		
Eu.P.52	Info	The terms and abbreviations are listed in the EULYNX Glossary [Eu.Doc.9].		
Eu.P.1350	Head	1.7 Variability management		
Eu.P.1351	Info	This document describes harmonised requirements. Variability management is not applicable.		
Eu.P.3024	Head	1.8 Definition of object types		
Eu.P.3025	Info	The following definition for object types is applied in this document:		
Eu.P.3026	Info	<ul style="list-style-type: none">• "Req" - This denotes a mandatory requirement.		
Eu.P.7246	Info	<ul style="list-style-type: none">• "Def" - This denotes referenceable model elements that are used in the model-based creation of requirements		
Eu.P.3027	Info	<ul style="list-style-type: none">• "Info" - This denotes additional information to help understand the specification. These objects do not specify any additional requirements.		
Eu.P.3028	Info	<ul style="list-style-type: none">• "Head" - This denotes chapter headings.		
Eu.P.53	Head	1.9 Modelling		
Eu.P.54	Info	The section "Functional requirements specification" follows a model based systems engineering process using Systems Modelling Language (SysML) and defines the functional system requirements for the Subsystem - Point operational in stimulus-response form. Furthermore the information objects (stimuli and responses) exchanged over the interfaces of the Subsystem - Point are defined.		
Eu.P.55	Info	The diagrams presented in this document are modelled in SysML [SysML].		
Eu.P.3050	Info	The rules for the interpretation of the model based parts of specification are defined in [Eu.Doc.29].		
Eu.P.3051	Info	In chapter 3 Functional requirements specification the functional system requirements, defined in the form of a SysML model in the PTC Integrity Modeler are depicted as a surrogate of this model in the form of DOORS-objects.		
Eu.P.3052	Info	A requirement thereby consists of the respective SysML model element, for instance a SysML diagram, and if necessary an additional extension of the requirement.		
Eu.P.3053	Info	In the column "Requirement Part 1" the particular SysML model element is depicted and in the column "Requirement Part 2" the corresponding extension of the definition is given. The stated object type normally applies both to "Requirement Part 1" and to "Requirement Part 2".		
Eu.P.3054	Info	There are requirements with type "Req" given, where the column "Requirement Part 2" or a part of it is provided with the heading "Information". In this case, the defined type only applies to the column "Requirement Part 1" and the part of "Requirement Part 2", which is not labelled as "Information".		

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7247	Info	State machines or several state machines linked together in a Functional Architecture define the totality of all functional requirements of an SUS or an SIUS in a coherent and consistent manner. State diagrams of a corresponding state machine are marked with the object type "Req". For the later design and implementation, it is not the description language SysML that is binding, but the domain-specific meaning expressed by it. The specified behaviour can be converted into a vendor specific language but must retain the domain specific meaning describing the functional requirements. The specific model elements are additionally specified and defined by object type "Def" to allow for traceability to supplier designs or test cases. The compliance of products to the specifications must be demonstrated by testing against EULYNX test cases, which are derived from the functionality specified by the models.		
Eu.P.57	Head	2 Conditions of use		
Eu.P.4952	Req	All references to [Eu.Doc.20] refer to version 4.0 (6.A) of that document.		
Eu.P.6374	Req	All references to [Eu.Doc.119] refer to version 1.1 (0.A) of that document.		
Eu.P.6375	Info	References to [Eu.Doc.120] do not refer to a concrete version of that document. The applicable version shall be defined by national specifications. Note: In future phases of the System Pillar, national specifications will be replaced by harmonised specifications.		
Eu.P.58	Info	The specifications defined in this document shall follow the requirements of the EULYNX System Architecture Specification [Eu.Doc.16].		
Eu.P.6159	Head	2.1 Functional Packages		
Eu.P.6160	Info	The specifications in this document are divided into functional packages. There are two types of packages related to the product capabilities.		
Eu.P.6161	Info	'Basic packages': One or more packages, at least one of them must be implemented. It is allowed to combine and implement more than one 'basic package' in a product.		
Eu.P.6162	Info	'Optional package': One or more packages that can be optionally implemented in addition to one or more basic packages.		
Eu.P.6163	Info	The specifications of the Subsystem – Point are divided into the following functional packages:		
Eu.P.6164	Info	Subsystem Point for a single point machine 4 wire (basic package) [Basic 4-wire single P].		
Eu.P.7154	Info	Subsystem Point for a single point machine non 4 wire (basic package) [Basic non-4-wire single P].		
Eu.P.6165	Info	Subsystem Point for multiple point machines 4 wire (basic package) [Basic 4-wire multiple P].		
Eu.P.7155	Info	Subsystem Point for multiple point machines non 4 wire (basic package) [Basic non-4-wire multiple P].		
Eu.P.6166	Info	Ability to Move functionality (optional package) [Option Able to move]. Note: This optional functional package can be combined with any of the 4 basic functional packages.		
Eu.P.7156	Info	Redrive functionality (optional package) [Option Redrive]. Note: This optional functional package can be combined only with the basic packages [Basic non-4-wire single P] and [Basic non-4-wire multiple P]		
Eu.P.7157	Info	Common Drive functionality (optional package) [Option Common Drive]. Note: This optional functional package can be combined only with the basic package [Basic non-4-wire multiple P].		
Eu.P.884	Head	3 Functional requirements specification		
Eu.P.6039	Head	3.1 Subsystem - Point - General Infos and Assumptions		
Eu.P.6292	Info	The defined model elements represent the Subsystem - Point in a general way. This refers to: <ul style="list-style-type: none">The functional architectures shown in the internal block diagramsThe defined number of Point machines in the state diagramsThe timing behaviour related to individual Point Machines.		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6158	Head	3.2 Subsystem - Point - Logical Viewpoint		
Eu.P.6282	Head	3.2.1 Subsystem - Point - Logical Context		

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.950	Def	<div><div>[Package] Subsystem - Point - Logical Context [Logical Viewpoint - Subsystem Definition]</div><div><div>bdd [Package] Subsystem - Point - Logical Context [Logical Viewpoint - Subsystem Definition]</div><div><div><div><div>«logical structural entity» Subsystem - Electronic Interlocking</div><div>1</div><div>SCI-P</div><div>1</div><div>SCI-P</div></div><div><div>«logical structural entity» Subsystem - Security Services Platform</div><div>1</div><div>SSI-P</div><div>1</div><div>SSI-P</div></div><div><div><div><div>«logical structural entity» Subsystem - Maintenance and Data Management</div><div>1</div><div>SMI-P</div><div>1</div><div>SMI-P</div></div><div><div>1</div><div>SDI-P</div><div>1</div><div>SDI-P</div></div></div><div><div>«environmental structural entity» Basic Data Identifier</div><div>1</div><div>P4</div><div>1</div><div>P4</div></div><div><div><div>«environmental structural entity» Maintainer</div><div>1</div><div>P1</div><div>1</div><div>P1</div></div><div><div>«logical structural entity» Subsystem - Point</div><div>1</div><div>P2</div><div>1</div><div>P2</div></div><div><div>«environmental structural entity» Point machine</div><div>1..*</div><div>P3</div><div>1</div><div>P3</div></div><div><div>«environmental structural entity» Power Supply</div><div>1</div><div>P2</div><div>1</div><div>P2</div></div></div></div></div></div></div></div>		Basic 4-wire single P Basic 4-wire multiple P Basic non-4-wire single P Basic non-4-wire multiple P
Eu.P.7479	Req	The Subsystem - Point shall provide a logical interface SCI-P to exactly one Subsystem - Electronic Interlocking.		
Eu.P.7482	Req	The Subsystem - Point shall provide a logical interface SSI-P to exactly one Subsystem - Security Services Platform.		
Eu.P.7480	Req	The Subsystem - Point shall provide a logical interface SMI-P to exactly one Subsystem - Maintenance and Data Management.		
Eu.P.7481	Req	The Subsystem - Point shall provide a logical interface SDI-P to exactly one Subsystem - Maintenance and Data Management.		
Eu.P.7483	Req	The Subsystem - Point shall provide a logical interface P4 to exactly one Basic Data identifier.		
Eu.P.7484	Req	The Subsystem - Point shall provide a logical interface P1 to exactly one Maintainer.		
Eu.P.7486	Req	The Subsystem - Point shall provide a logical interface P3 to at least one Point machine.		
Eu.P.7485	Req	The Subsystem - Point shall provide a logical interface P2 to exactly one Power Supply.		
Eu.P.5780	Head	3.3 Subsystem - Point - Functional Viewpoint		
Eu.P.2286	Head	3.3.1 Definition of time values		
Eu.P.3068	Info	The generic time values for SCI are specified in [Eu.Doc.119].		Basic 4-wire single P Basic 4-wire multiple P Basic non-4-wire single P Basic non-4-wire multiple P
Eu.P.6376	Info	The generic time values for SMI are specified in [Eu.Doc.120].		Basic 4-wire single P Basic 4-wire multiple P Basic non-4-wire single P Basic non-4-wire multiple P
Eu.P.2439	Def	Con_tmax_Point_Operation	Con_tmax_Point_Operation defines the max. time period the Point has to arrive to an End position, starting with the command Moving to the Point machine. After that time period the command Moving to the Point machine is set to Stop moving.	Basic 4-wire single P Basic 4-wire multiple P Basic non-4-wire single P Basic non-4-wire multiple P
Eu.P.5782	Head	3.3.2 Subsystem - Point - Functional Context		

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.1467	Info	<div><div>[Package] Subsystem - Point - Functional Context [Functional Viewpoint - Subsystem Definition - Initialisation]</div><div>uc [Package] Subsystem - Point - Functional Context [Functional Viewpoint - Subsystem Definition - Initialisation]</div><div><p>The diagram shows the functional context of the Subsystem - Point. It is enclosed in a dashed box. Inside, there are two external components: 'Subsystem - Electronic Interlocking' and 'Subsystem - Maintenance and Data Management', both represented by 3D cubes. The 'Subsystem - Point' box contains several Use Cases (UCs) represented by green ovals: 'SCI-XX EfeS IFUC1.1: Establish PDI connection', 'SCI-XX EfeS IFUC1.2: Close PDI connection', 'SMI-XX IFUC 1.1: Establish SMI connection', 'SMI-XX IFUC 1.2: Synchronous loading and activation of data', 'SMI-XX IFUC 1.3: Asynchronous preloading of data', 'SMI-XX IFUC 1.4: Reset EfeS', 'SMI-XX IFUC 1.5: Initiate maintenance', 'P_UC1.3: Report status', and 'P_UC1.4: Establish initial state of outputs'. A 'Point machine' (represented by a 3D cube) is connected to 'P_UC1.4'. Solid lines connect 'Subsystem - Electronic Interlocking' to 'SCI-XX EfeS IFUC1.1' and 'SCI-XX EfeS IFUC1.2', and 'Subsystem - Maintenance and Data Management' to 'SMI-XX IFUC 1.1', 'SMI-XX IFUC 1.2', 'SMI-XX IFUC 1.3', and 'SMI-XX IFUC 1.5'. A dashed line with the label «include» connects 'SCI-XX EfeS IFUC1.1' to 'P_UC1.3'.</p></div></div>		Basic 4-wire single P Basic 4-wire multiple P Basic non-4-wire single P Basic non-4-wire multiple P
Eu.P.3070	Info	The generic UseCases SCI-XX EfeS IFUC1.1: Establish PDI connection and SCI-XX EfeS IFUC1.2: Close PDI connection are specified in [Eu.Doc.119]. The generic UseCases SMI-XX IFUC 1.1: Establish SMI connection, SMI-XX IFUC 1.2: Synchronous loading and activation of data, SMI-XX IFUC 1.3: Asynchronous preloading of data, SMI-XX IFUC 1.4: Reset EfeS and SMI-XX IFUC 1.5: Initiate maintenance are specified in [Eu.Doc.120].		Basic 4-wire single P Basic 4-wire multiple P Basic non-4-wire single P Basic non-4-wire multiple P
Eu.P.1465	Info	P_UC1.3: Report status	The Subsystem-UseCase P_UC1.3: Report status defines a scenario about the transmission of status data of Subsystem - Point to Subsystem - Electronic Interlocking, while Process Data Interface protocol connection is establishing.	Basic 4-wire single P Basic 4-wire multiple P Basic non-4-wire single P Basic non-4-wire multiple P
Eu.P.5788	Info	<div><div>Main Success Scenario: Report status [P SD 1.3.1]</div><div>P_UC1.3: Report status</div><div><div>Main Success Scenario: Report status [P SD 1.3.1]</div><div>alt [The Subsystem - Point is configured with one Point machine]</div><div>1.a1 The Subsystem - Electronic Interlocking receives the status of the Subsystem - Point</div><div>else alt [The Subsystem - Point is configured with multiple Point machines]</div><div>1.b1 The Subsystem - Electronic Interlocking receives the status of the Subsystem - Point</div><div>end alt</div></div><div><p>The sequence diagram shows the interaction for the 'P_UC1.3: Report status' use case. It involves three lifelines: 'Subsystem - Electronic Interlocking' (actor), 'Point machine' (actor), and ':Subsystem - Point' (boundary). The diagram is an 'alt' block with two scenarios. The first scenario, 'Main Success Scenario: Report status with single point machine [P SD 1.3.1.1]', shows a message from 'Subsystem - Electronic Interlocking' to 'Point machine' (labeled 'ref') and then from 'Point machine' to ':Subsystem - Point'. The second scenario, 'Main Success Scenario: Report status with multiple point machines [P SD 1.3.1.2]', shows a message from 'Subsystem - Electronic Interlocking' to ':Subsystem - Point' (labeled 'ref').</p></div></div>	This SD is part of [SCI-XX EfeS IF SD 1.1.1] in [Eu.Doc.119].	Basic 4-wire single P Basic 4-wire multiple P Basic non-4-wire single P Basic non-4-wire multiple P

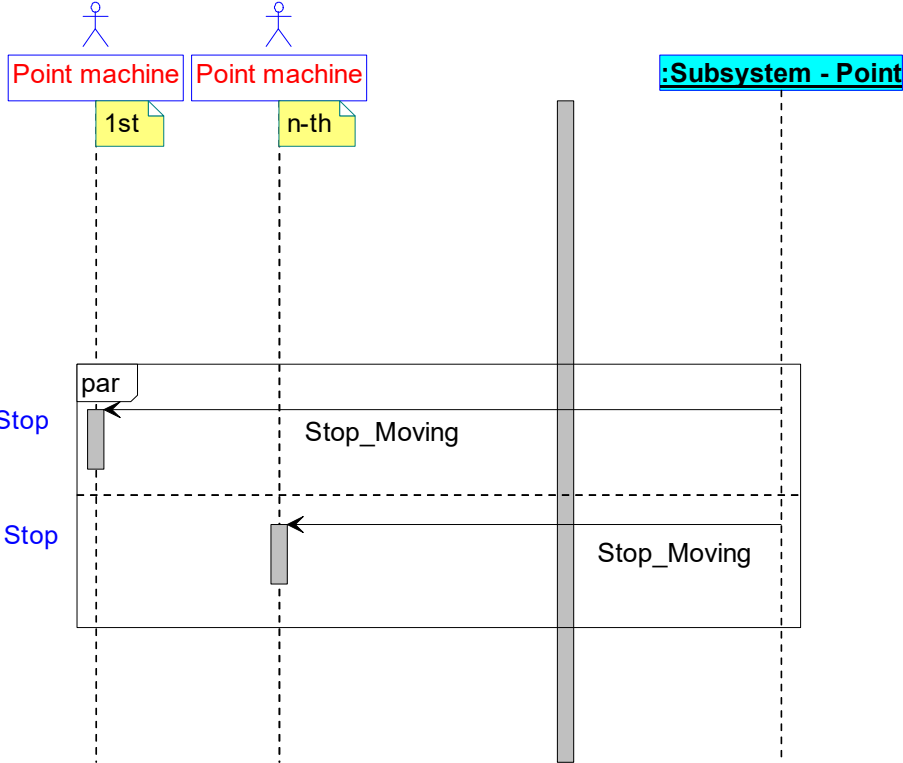
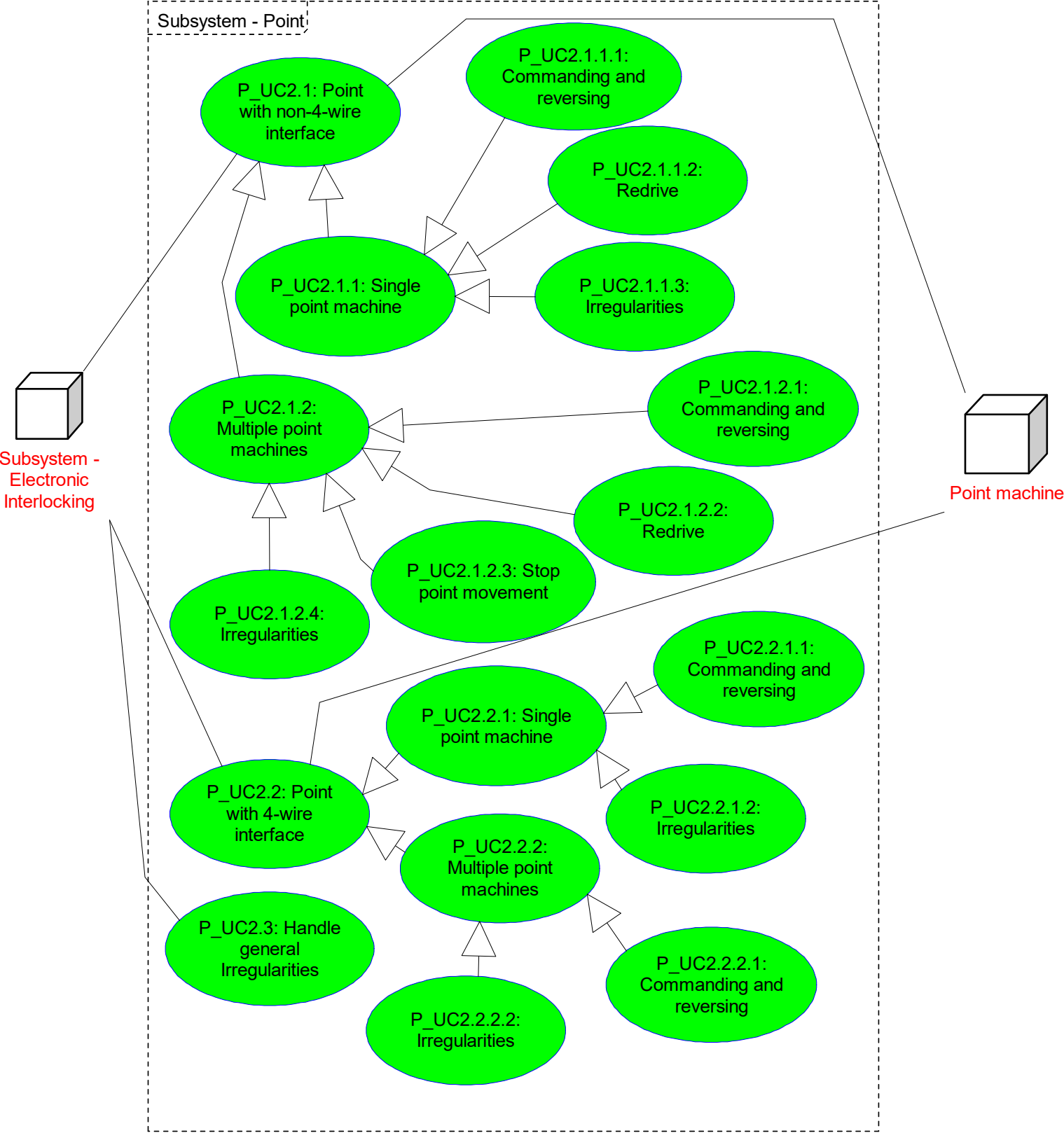
ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.1110	Info	<div><p>Main Success Scenario: Report status with single point machine [P SD 1.3.1.1]</p><p>P UC1.3: Report status</p><p>Main Success Scenario: Report status with single point machine [P SD 1.3.1.1]</p><p>par</p><div><p>1.a1 The Subsystem - Electronic Interlocking receives the point position status of the Subsystem - Point</p></div><p>also par</p><div><p>1.b1 The Subsystem - Electronic Interlocking receives the ability to move status of the Subsystem - Point</p></div><p>end par</p></div> <div><p>The diagram shows three lifelines: Subsystem - Electronic Interlocking, Point machine, and :Subsystem - Point. A 'par' block contains two 'ref' blocks. The first 'ref' block points to 'Main Success Scenario: Report point position status with single point machine [P SD 1.3.1.1.1]'. The second 'ref' block points to 'Main Success Scenario: Report ability to move point status with single point machine [P SD 1.3.1.1.2]'.</p></div>	This SD is part of [P SD 1.3.1].	Basic 4-wire single P Basic non-4-wire single P
Eu.P.5787	Info	<div><p>Main Success Scenario: Report point position status with single point machine [P SD 1.3.1.1.1]</p><p>P UC1.3: Report status</p><p>Main Success Scenario: Report point position status with single point machine [P SD 1.3.1.1.1]</p><p>Interaction 1.3.1.1.1.A</p><p>alt [The Subsystem - Point is configured with a non-4-wire interface to the point machine OR the Subsystem - Point is configured with a 4-wire interface to the point machine AND the last commanded position is End position "Y"]</p><div><p>alt [The Point is in an End position "Y"]</p><div><p>1.a1.a1 - The Subsystem - Point receives from the Point machine the Information that the Point is in an End position "Y".</p></div><p>else alt [The Point is in No end position]</p><div><p>1.a1.b1 - The Subsystem - Point receives from the Point machine the Information that the Point is in No end position.</p></div><p>else alt [The Point is in a Unintended position]</p><div><p>1.a1.c1 - The Subsystem - Point receives from the Point machine the Information that the Point is in a Unintended position.</p></div></div><p>end alt</p><p>else alt [The Subsystem - Point is configured with a 4-wire interface to the Point machine AND the last commanded position is not available]</p><div><p>1.b1 - The Subsystem - Point receives from the Point machine the Information that the Point is in No end position</p></div><p>end alt</p><p>Interaction 1.3.1.1.1.B</p><p>2. The Subsystem - Point reports to the Subsystem - Electronic Interlocking the Point position.</p></div> <div><p>The diagram shows three lifelines: Subsystem - Electronic Interlocking, Point machine, and :Subsystem - Point. An 'alt' block contains two 'alt' blocks. The first 'alt' block has three branches: 1.a1.a1 (Subsystem - Point receives Information that the Point is in an End position "Y"), 1.a1.b1 (Subsystem - Point receives Information that the Point is in No end position), and 1.a1.c1 (Subsystem - Point receives Information that the Point is in a Unintended position). The second 'alt' block has one branch: 1.b1 (Subsystem - Point receives Information that the Point is in No end position). Below the 'alt' block, a message 'Msg_Point_Position(current position)' is sent from :Subsystem - Point to Subsystem - Electronic Interlocking.</p></div>	<p>If a state change happens while establishing the PDI connection and the status report Msg_Point_Position has already been sent, a new status report Msg_Point_Position has to be sent to Subsystem - Electronic Interlocking immediately after completion of the establishment of the connection.</p> <p>This SD is part of [P SD 1.3.1.1].</p>	Basic 4-wire single P Basic non-4-wire single P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.5784	Info	<div><p>Main Success Scenario: Report ability to move point status with single point machine [P SD 1.3.1.1.2]</p><p>P UC1.3: Report status</p><p>Main Success Scenario: Report ability to move point status with single point machine [P SD 1.3.1.1.2]</p><p>Interaction 1.3.1.1.2.A</p><p>alt [The Point is Able to move point]</p><div><p>par</p><div><p>1.a1.a1 - The Subsystem - Point receives from the Point machine the Information that the Point is Able to move point.</p></div><div><p>also par</p><div><p>1.a1.a2 The Subsystem - Point receives from the internal trigger the Information that the Point is Able to move point.</p></div></div><p>end par</p><p>else alt [The Point is Unable to move point]</p><div><p>alt [The Point machine is Unable to move point.]</p><div><p>1.b1.a1 - The Subsystem - Point receives from the Point machine the Information that the Point is Unable to move point.</p></div><p>else alt [The internal trigger indicates Unable to move point]</p><div><p>1.b1.b1 The Subsystem - Point receives from the internal trigger the Information that the Point is Unable to move point.</p></div></div><p>end alt</p><p>end alt</p><p>Interaction 1.3.1.1.2 B</p><p>2. The Subsystem - Point reports to the Subsystem - Electronic Interlocking the Ability to move point.</p></div><div><pre>sequenceDiagram participant S as Subsystem - Electronic Interlocking participant P as Point machine participant SP as :Subsystem - Point alt [The Point is Able to move point] par SP->>P: Information_Ability_To_Move_Point P->>SP: Information_Ability_To_Move_Point_Available_TRUE and SP->>SP: Information_Ability_To_Move_Point_Available_TRUE end else [The Point is Unable to move point] alt [The Point machine is Unable to move point.] SP->>P: Information_Ability_To_Move_Point P->>SP: Information_Ability_To_Move_Point_Available_FALSE else [The internal trigger indicates Unable to move point] SP->>SP: Information_Ability_To_Move_Point_Available_FALSE end end SP->>S: Msg_Ability_To_Move_Point(current status)</pre></div></div>	<p>Only applicable if the package [Option Able to move] is used in combination with [Basic 4-wire single P] or [Basic non-4-wire single P].</p> <p>This SD is part of [P SD 1.3.1.1].</p>	<p>Option Able to move</p>
Eu.P.5789	Info	<div><p>Main Success Scenario: Report status with multiple point machines [P SD 1.3.1.2]</p><p>P UC1.3: Report status</p><p>Main Success Scenario: Report status with multiple point machines [P SD 1.3.1.2]</p><p>par</p><div><p>1.a1.a1 The Subsystem - Point receives the point position status from the Point machines.</p></div><div><p>also par</p><div><p>1.a1.a2 The Subsystem - Point receives the degraded point position status from the Point machines.</p></div></div><p>end par</p><p>1.a2 - The Subsystem - Point reports to the Subsystem - Electronic Interlocking the Point position.</p><p>also par</p><div><p>1.b1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking the Ability to move point.</p></div><p>end par</p></div> <div><pre>sequenceDiagram participant S as Subsystem - Electronic Interlocking participant P1 as Point machine 1st participant Pn as Point machine n-th participant SP as :Subsystem - Point par ref Main Success Scenario: Report point position status with multiple point machines [P SD 1.3.1.2.1] ref Main Success Scenario: Report degraded point position status with multiple point machines [P SD 1.3.1.2.2] end SP->>S: Msg_Point_Position(current position) ref Main Success Scenario: Report ability to move point status with multiple point machines [P SD 1.3.1.2.3]</pre></div>	<p>If a state change happens while establishing the PDI connection and the status report Msg_Point_Position has already been sent, a new status report Msg_Point_Position has to be sent to Subsystem - Electronic Interlocking immediately after completion of the establishment of the connection.</p> <p>This SD is part of [P SD 1.3.1].</p>	<p>Basic 4-wire multiple P</p> <p>Basic non-4-wire multiple P</p>

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.5786	Info	<div><p>Main Success Scenario: Report point position status with multiple point machines [P SD 1.3.1.2.1]</p><p>P UC1.3: Report status</p><p>Main Success Scenario: Report point position status with multiple point machines [P SD 1.3.1.2.1]</p><p>Interaction 1.3.1.2.1.A</p><p>alt [The Subsystem - Point is configured with a non-4-wire interface to the Point machine OR the Subsystem - Point is configured with a 4-wire interface to the Point machine AND the last commanded position is End position 'Y']</p><p>alt [The Point is in an End position 'Y']</p><p>par</p><p>1.a1.a1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the Point is in an End position 'Y'.</p><p>also par</p><p>1.a1.a1.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the Point is in an End position 'Y'.</p><p>end par</p><p>else alt [The Point is in a Unintended position]</p><p>alt [The 1st Point machine is in a Unintended position.]</p><p>1.a1.b1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the Point is in a Unintended position.</p><p>else alt [The n-th Point machine is in a Unintended position.]</p><p>1.a1.b1.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the Point is in a Unintended position.</p><p>end alt</p><p>end alt</p><p>alt [The 1st Point machine is in a No end position.]</p><p>1.a2.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the Point is in a No end position.</p><p>else alt [The n-th Point machine is in a No end position.]</p><p>1.a2.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the Point is in a No end position.</p><p>end alt</p><p>end alt</p><p>else alt [The Subsystem - Point is configured with a 4-wire interface to the Point machine AND the last commanded position is not available]</p><p>1.b1 - The Point is in No end position.</p><p>end alt</p></div> <pre>sequenceDiagram participant S as Subsystem - Electronic Interlocking participant P1 as Point machine 1st participant Pn as Point machine n-th participant SP as :Subsystem - Point alt [The Subsystem - Point is configured with a non-4-wire interface to the Point machine OR the Subsystem - Point is configured with a 4-wire interface to the Point machine AND the last commanded position is End position 'Y'] alt [The Point is in an End position 'Y'] par P1->>SP: Information_End_Position_Detected and Pn->>SP: Information_End_Position_Detected end else alt [The Point is in a Unintended position] alt [The 1st Point machine is in a Unintended position.] P1->>SP: Information_Unintended_Position else alt [The n-th Point machine is in a Unintended position.] Pn->>SP: Information_Unintended_Position end end end alt [The 1st Point machine is in a No end position.] P1->>SP: Information_No_End_Position else alt [The n-th Point machine is in a No end position.] Pn->>SP: Information_No_End_Position end end alt alt [The Subsystem - Point is configured with a 4-wire interface to the Point machine AND the last commanded position is not available] P1->>SP: Information_No_End_Position end alt</pre>	This SD is part of [P SD 1.3.1.2].	Basic 4-wire multiple P Basic non-4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.5785	Info	<div><div>Main Success Scenario: Report degraded point position status with multiple point machines [P SD 1.3.1.2.2]</div><div>P UC1.3: Report status</div><div><div>Main Success Scenario: Report degraded point position status with multiple point machines [P SD 1.3.1.2.2]</div><div>Interaction 1.3.1.2.2.A</div><div><div>opt [The Subsystem - Point is configured with one or more non-crucial Point machines]</div><div><div>opt [The Subsystem - Point is configured with a non-4-wire interface to the Point machine OR the Subsystem - Point is configured with a 4-wire interface to the Point machine AND the last commanded position is End position "X". The Point is in a Degraded point position "X"]</div><div><div>par</div><div><div>1.a1.a1.a1 - The Subsystem - Point receives from the 1st crucial Point machine the Information that the 1st crucial Point machine is in an End position "X".</div></div><div><div>also par</div><div><div>opt [The Subsystem - Point is configured with more than one crucial Point machines]</div><div><div>1.a1.a1.b1.a1 - The Subsystem - Point receives from the i-th crucial Point machine the Information that the 1st crucial Point machine is in an End position "X".</div></div><div>end opt</div></div><div><div>also par</div><div><div>alt</div><div><div>alt</div><div><div>1.a1.a1.c1.a1.a1 - The Subsystem - Point receives from the 1st non-crucial Point machine the Information that the 1st non-crucial Point machine is in a Unintended position.</div></div><div><div>else alt</div><div><div>1.a1.a1.c1.a1.b1 - The Subsystem - Point receives from the 1st non-crucial Point machine the Information that the 1st non-crucial Point machine is in a No end position.</div></div><div>end alt</div></div><div><div>else alt [The Subsystem - Point is configured with more than one non-crucial Point machine]</div><div><div>alt</div><div><div>1.a1.a1.c1.b1.a1 - The Subsystem - Point receives from the k-th non-crucial Point machine the Information that the k-th non-crucial Point machine is in a Unintended position.</div></div><div><div>else alt</div><div><div>1.a1.a1.c1.b1.b1 - The Subsystem - Point receives from the k-th non-crucial Point machine the Information that the k-th non-crucial Point machine is in a No end position.</div></div><div>end alt</div></div><div>end alt</div></div><div><div>also par</div><div><div>1.a1.a1.d1 - The 1st non-crucial Point machine is NOT in an End position "Y".</div></div><div><div>also par [The Subsystem - Point is configured with a k-th non-crucial Point machine]</div><div><div>1.a1.a1.e1 - The k-th non-crucial Point machine is NOT in an End position "Y".</div></div><div>end par</div></div><div>end opt</div><div>end opt</div></div></div></div></div></div></div></div></div></div></div></div>	This SD is part of [P SD 1.3.1.2].	Basic 4-wire multiple P Basic non-4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.5783	Info	<div><p>Main Success Scenario: Report ability to move point status with multiple point machines [P SD 1.3.1.2.3]</p><p>P UC1.3: Report status</p><p>Main Success Scenario: Report ability to move point status with multiple point machines [P SD 1.3.1.2.3]</p><p>Interaction 1.3.1.2.3.A</p><p>alt [The Point is Able to move point]</p><p>par</p><p>1.a1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the 1st Point machine is Able to move point.</p><p>also par</p><p>opt [The n-th Point machine is configured with full functionality]</p><p>1.a1.b1.a1 - The Subsystem - Point receives from the n-th Point machine the Information that the n-th Point machine Able to move point.</p><p>end opt</p><p>also par</p><p>1.a1.c1 - The Subsystem - Point internal trigger Ability to move point available indicates availability of Ability to move point.</p><p>end par</p><p>else alt [The Point is Unable to move point]</p><p>alt [The 1st Point machine is Unable to move point.]</p><p>1.b1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the 1st Point machine Unable to move point.</p><p>else alt [The n-th Point machine is configured with full functionality and is Unable to move point.]</p><p>1.b1.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the n-th Point machine Unable to move point.</p><p>else alt [The internal trigger indicates Unable to move point]</p><p>1.b1.c1 - The Subsystem - Point internal trigger Ability to move point available indicates non-availability of Ability to move point.</p><p>end alt</p><p>end alt</p><p>Interaction 1.3.1.2.3.B</p><p>2. - The Subsystem - Point reports to the Subsystem - Electronic Interlocking the Ability to move point.</p></div> <pre>sequenceDiagram participant SEI as Subsystem - Electronic Interlocking participant PM1 as Point machine 1st participant PMn as Point machine n-th participant SP as :Subsystem - Point alt [The Point is Able to move point] par opt [The n-th Point machine is configured with full functionality] PMn->>SP: Information_Ability_To_Move_Point SP->>SP: Information_Ability_To_Move_Point_Available_TRUE end end else [The Point is Unable to move point] alt [The 1st Point machine is Unable to move point.] PM1->>SP: Information_Ability_To_Move_Point else [The n-th Point machine is configured with full functionality and is Unable to move point.] PMn->>SP: Information_Ability_To_Move_Point else [The internal trigger indicates Unable to move point] SP->>SP: Information_Ability_To_Move_Point_Available_FALSE end end SP->>SEI: Msg_Ability_To_Move_Point(current staus)</pre>	<p>Only applicable if the package [Option Able to move] is used in combination with [Basic 4-wire multiple P] or [Basic non-4-wire multiple P].</p> <p>This SD is part of [P SD 1.3.1.2].</p>	<p>Option Able to move</p>
Eu.P.1109	Info	<p>P_UC1.4: Establish initial state of outputs</p>	<p>The Subsystem-UseCase P_UC1.4: Establish initial state of outputs defines the main success scenario for establishing the initial state of outputs of the Subsystem - Point.</p>	<p>Basic 4-wire single P Basic 4-wire multiple P Basic non-4-wire single P Basic non-4-wire multiple P</p>
Eu.P.1466	Info	<div><p>Main Success Scenario: Set Initial State of Outputs with single point machine [P SD 1.4.1]</p><p>P UC1.4: Establish initial state of outputs</p><p>Main Success Scenario: Set Initial State of Outputs with single point machine [P SD 1.4.1]</p><p>Precondition:</p><p>The Subsystem - Point is in the state BOOTING.</p><p>Interaction 1.4.1.A:</p><p>1. - The Subsystem - Point enters the state INITIALISING.</p><p>2. The Subsystem - Point sends the Command to the Point machine to Stop moving the Point machine.</p><p>Postconditions:</p><p>The Subsystem - Point is in the state INITIALISING.</p><p>The Initial State Of Outputs of the Subsystem - Point has been set.</p></div> <pre>sequenceDiagram participant PM as Point machine participant SP as :Subsystem - Point SP->>PM: Stop_Moving</pre>	<p>Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.</p>	<p>Basic 4-wire single P Basic non-4-wire single P</p>

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.5790	Info	<div><p>Main Success Scenario: Set Initial State of Outputs with multiple point machines [P SD 1.4.2]</p><p>P_UC1.4: Establish initial state of outputs</p><p>Main Success Scenario: Set Initial State of Outputs with multiple point machines [P SD 1.4.2]</p><p>Precondition:</p><p>The Subsystem - Point is in the state BOOTING.</p><p>Interaction 1.4.2.A:</p><p>1. - The Subsystem - Point enters the state INITIALISING.</p><p>par</p><p>2.a1 The Subsystem - Point sends the Command to the 1st Point machine to Stop moving the Point machine.</p><p>also par</p><p>2.b1 The Subsystem - Point sends the Command to the n-th Point machine to Stop moving the Point machine.</p><p>end par</p><p>Postconditions:</p><p>The Subsystem - Point is in the state INITIALISING.</p><p>The Initial State Of Outputs of the Subsystem - Point has been set.</p></div> <div></div>	Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.	Basic 4-wire multiple P Basic non-4-wire multiple P
Eu.P.6706	Info	<div><p>[Package] Subsystem - Point - Functional Context [Functional Viewpoint - Subsystem Definition - Operation]</p><p>uc [Package] Subsystem - Point - Functional Context [Functional Viewpoint - Subsystem Definition - Operation]</p></div> <div></div>		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P Option Redrive

ID	Type	Requirement Part 1		Requirement Part 2	Func. Pkg.
Eu.P.6732	Info	P_UC2.1: Point with non-4-wire interface		The Subsystem-UseCase "P_UC2.1: Point with non-4-wire interface" defines the behaviour of the Subsystem - Point which works with a non-4-wire interface. The behaviour will be defined in the following UseCases: P_UC2.1.1: Single point machine P_UC2.1.2: Multiple point machines	Basic non-4-wire single P Basic non-4-wire multiple P
Eu.P.6720	Info	P_UC2.1.1: Single point machine		The Subsystem-UseCase "P_UC2.1.1: Single point machine" defines the behaviour of the Subsystem - Point which works with a single point machine via non-4-wire interface. The behaviour will be defined in the following UseCases: P_UC2.1.1.1: Commanding and reversing P_UC2.1.1.2: Redrive P_UC2.1.1.3: Irregularities	Basic non-4-wire single P
Eu.P.6707	Info	P_UC2.1.1.1: Commanding and reversing		The Subsystem-UseCase "P_UC2.1.1.1: Commanding and reversing" defines the behaviour of commanding and reversing a single point machine via non-4-wire interface.	Basic non-4-wire single P
Eu.P.6714	Info	<p>Main Success Scenario: Moving of the Point with a single point machine Non 4W [P SD 2.1.1.1.1]</p> <p><u>P_UC2.1.1.1: Commanding and reversing</u></p> <div></div> <p>Main Success Scenario: Moving of the Point with a single point machine Non 4W [P SD 2.1.1.1.1]</p> <p>Precondition: The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y", or - No end position, or - an Unintended position.</p> <p>Interaction 2.1.1.1.1.A: 1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X". 2. The Subsystem - Point sends the Command to the Point machine to move the Point machine to an End position "X". At this moment the Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p> <p>Interaction 2.1.1.1.1.B: opt [The Subsystem - Point was previously in an End position or a Unintended position] 3.a1 - The Subsystem - Point receives from the Point machine the Information that the Point machine is in No end position. 3.b1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position. end opt</p> <p>Interaction 2.1.1.1.1.C: 4. - The Subsystem - Point receives from the Point machine the Information that the Point machine is in an End position "X". 5. The Subsystem - Point sends the Command to the Point machine to stop moving the Point machine. The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation. 6. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in an End position "X".</p> <p>Postcondition: The Subsystem - Point is in an End position "X".</p>		Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.	Basic non-4-wire single P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6711	Info	<div><p>Alternative Scenario: Reversing Point directly after the position has been commanded Non 4W [P SD 2.1.1.1.3]</p><p>P UC2.1.1.1: Commanding and reversing</p><p>Alternative Scenario: Reversing Point directly after the position has been commanded Non 4W [P SD 2.1.1.1.3]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y", or - an Unintended position.</p><p>Interaction 2.1.1.1.3.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X". 2. The Subsystem - Point sends the Command to the Point machine to move the Point to an End position "X". At this moment the Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>Interaction 2.1.1.1.3.B:</p><p>par</p><p>3.a1 The Subsystem - Point receives from the Point machine the Information that the Point machine is in No end position.</p><p>also par</p><p>3.b1 - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to the opposite End position "Y".</p><p>end par</p><p>4. The Subsystem - Point sends the Command to the Point machine to move the Point to an End position "Y". At this moment the Subsystem - Point restarts to monitor the time period Con_tmax_Point_Operation. 5. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</p><p>Interaction 2.1.1.1.3.C:</p><p>6. - The Subsystem - Point receives from the Point machine the Information that the Point machine is in an End position "Y". The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation. 7. The Subsystem - Point sends the Command to the Point machine to stop moving the Point machine. 8. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in an End position "Y".</p><p>Postcondition:</p><p>The Subsystem - Point is in an End position "Y".</p></div> <pre>sequenceDiagram participant IE as Subsystem - Electronic Interlocking participant PM as Point machine participant SP as Subsystem - Point Note over SP: Alternative Scenario: Reversing Point directly after the position has been commanded Non 4W [P SD 2.1.1.1.3] Note over SP: Precondition: Note over SP: The Subsystem - Point is in the state OPERATIONAL. Note over SP: The Subsystem - Point is configured with a non-4-wire interface to the Point machine. Note over SP: The Subsystem - Point is in: Note over SP: - an End position "Y", or Note over SP: - an Unintended position. Note over SP: Interaction 2.1.1.1.3.A: 1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X". 2. The Subsystem - Point sends the Command to the Point machine to move the Point to an End position "X". At this moment the Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation. Note over SP: Interaction 2.1.1.1.3.B: par 3.a1 The Subsystem - Point receives from the Point machine the Information that the Point machine is in No end position. and 3.b1 - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to the opposite End position "Y". end Note over SP: end par 4. The Subsystem - Point sends the Command to the Point machine to move the Point to an End position "Y". At this moment the Subsystem - Point restarts to monitor the time period Con_tmax_Point_Operation. 5. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position. Note over SP: Interaction 2.1.1.1.3.C: 6. - The Subsystem - Point receives from the Point machine the Information that the Point machine is in an End position "Y". The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation. 7. The Subsystem - Point sends the Command to the Point machine to stop moving the Point machine. 8. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in an End position "Y". Note over SP: Postcondition: Note over SP: The Subsystem - Point is in an End position "Y". IE->>SP: Cd_Move_Point(End Position X) activate SP SP->>PM: Cd_Move_Point(End Position X) activate PM PM-->>SP: Information_No_End_Position deactivate PM Note over SP: {<= Con_tmax_Point_Operation} Note over SP: Moving Note over SP: {<= Con_tmax_Point_Operation} Note over SP: Moving SP->>IE: Msg_Point_Position(No End Position) deactivate SP Note over SP: {<= Con_tmax_Point_Operation} Note over SP: Moving Note over SP: {<= Con_tmax_Point_Operation} Note over SP: Moving SP->>PM: Cd_Move_Point(End Position Y) activate PM PM->>SP: Information_End_Position_Detected deactivate PM Note over SP: Stop_Moving SP->>IE: Msg_Point_Position(End Position Y) deactivate SP</pre>	Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.	Basic non-4-wire single P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6709	Info	<div><p>Alternative Scenario: Moving of the Point with repeated command of moving #2 Non 4W [P SD 2.1.1.1.5]</p><p>P UC2.1.1.1: Commanding and reversing</p><p>Alternative Scenario: Moving of the Point with repeated command of moving #2 Non 4W [P SD 2.1.1.1.5]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y" or - an Unintended position.</p><p>Interaction 2.1.1.1.5.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X".</p><p>2. The Subsystem - Point sends the Command to the Point machine to move the Point to an End position "X". At this moment the Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>Interaction 2.1.1.1.5.B:</p><p>par</p><p>3.a1 - The Subsystem - Point receives from the Point machine the Information that the Point is in No end position.</p><p>also par</p><p>3.b1 - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X".</p><p>end par</p><p>4. The Subsystem - Point ignores the command from the Subsystem - Electronic Interlocking.</p><p>5. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</p><p>Interaction 2.1.1.1.5.C:</p><p>6. - The Subsystem - Point receives from the Point machine the Information that the Point is in an End position "X".</p><p>7. The Subsystem - Point sends the Command to the Point machine to stop moving the Point. The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation.</p><p>8. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in an End position "X".</p><p>Postcondition:</p><p>The Subsystem - Point is in an End position "X".</p></div>	Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.	Basic non-4-wire single P
Eu.P.6712	Info	<div><p>Alternative Scenario: Reversing Point directly without position change Non 4W [P SD 2.1.1.1.6]</p><p>P UC2.1.1.1: Commanding and reversing</p><p>Alternative Scenario: Reversing Point directly without position change Non 4W [P SD 2.1.1.1.6]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is in an End position "Y".</p><p>Interaction 2.1.1.1.6.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X".</p><p>2. The Subsystem - Point sends the Command to the Point machine to move the Point to an End position "X". At this moment the Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>Interaction 2.1.1.1.6.B:</p><p>3. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "Y". The Subsystem - Point is still in an End position "Y".</p><p>4. The Subsystem - Point sends the Command to the Point machine to stop moving the Point. The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation.</p><p>Postcondition:</p><p>The Subsystem - Point is in an End position "Y".</p></div>	Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.	Basic non-4-wire single P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6715	Info	P_UC2.1.1.2: Redrive	The Subsystem-UseCase "P_UC2.1.1.2: Redrive" defines the behaviour of redriving a single point machine via non-4-wire interface. Only applicable if the package [Option Redrive] is used in combination with [Basic non-4-wire single P].	Option Redrive
Eu.P.6717	Info	<p>Main Success Scenario: Redrive Point after lost end position Non 4W [P SD 2.1.1.2.1]</p> <p>P_UC2.1.1.2: Redrive</p> <p>Main Success Scenario: Redrive Point after lost end position Non 4W [P SD 2.1.1.2.1]</p> <p>Precondition:</p> <p>The Subsystem - Point is in the state INITIALISING or OPERATIONAL. The Subsystem - Point is in an End position "X". The last Cd_Move_Point Command received was also for End position "X". The last Cd_Move_Point Command received occurred since exiting the state BOOTING. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is configured as Redrive point.</p> <p>Interaction 2.1.1.2.1.A:</p> <p>1. - The Subsystem - Point receives from the Point machine the Information that the Point machine is in No end position.</p> <p>opt [The Subsystem - Point is in the state OPERATIONAL.]</p> <p>2.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</p> <p>end opt</p> <p>3. The Subsystem - Point sends the Command to the Point machine to move the Point back to an End position "X". At this moment the Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p> <p>Interaction 2.1.1.2.1.B:</p> <p>4. - The Subsystem - Point receives from the Point machine the Information that the Point machine is in an End position "X".</p> <p>5. The Subsystem - Point sends the Command to the Point machine to stop moving the Point machine. The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation.</p> <p>opt [The Subsystem - Point is in the state OPERATIONAL.]</p> <p>6.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in an End position "X".</p> <p>end opt</p> <p>Postcondition:</p> <p>The Subsystem - Point is in an End position "X".</p> <pre>sequenceDiagram participant SEI as Subsystem - Electronic Interlocking participant PM as Point machine participant SP as :Subsystem - Point Note over SEI, PM, SP: Main Success Scenario: Redrive Point after lost end position Non 4W [P SD 2.1.1.2.1] Note over SEI, PM, SP: P_UC2.1.1.2: Redrive Note over SEI, PM, SP: Main Success Scenario: Redrive Point after lost end position Non 4W [P SD 2.1.1.2.1] Note over SEI, PM, SP: Precondition: Note over SEI, PM, SP: The Subsystem - Point is in the state INITIALISING or OPERATIONAL. Note over SEI, PM, SP: The Subsystem - Point is in an End position "X". Note over SEI, PM, SP: The last Cd_Move_Point Command received was also for End position "X". Note over SEI, PM, SP: The last Cd_Move_Point Command received occurred since exiting the state BOOTING. Note over SEI, PM, SP: The Subsystem - Point is configured with a non-4-wire interface to the Point machine. Note over SEI, PM, SP: The Subsystem - Point is configured as Redrive point. Note over SEI, PM, SP: Interaction 2.1.1.2.1.A: Note over SEI, PM, SP: 1. - The Subsystem - Point receives from the Point machine the Information that the Point machine is in No end position. Note over SEI, PM, SP: opt [The Subsystem - Point is in the state OPERATIONAL.] Note over SEI, PM, SP: 2.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position. Note over SEI, PM, SP: end opt Note over SEI, PM, SP: 3. The Subsystem - Point sends the Command to the Point machine to move the Point back to an End position "X". At this moment the Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation. Note over SEI, PM, SP: Interaction 2.1.1.2.1.B: Note over SEI, PM, SP: 4. - The Subsystem - Point receives from the Point machine the Information that the Point machine is in an End position "X". Note over SEI, PM, SP: 5. The Subsystem - Point sends the Command to the Point machine to stop moving the Point machine. The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation. Note over SEI, PM, SP: opt [The Subsystem - Point is in the state OPERATIONAL.] Note over SEI, PM, SP: 6.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in an End position "X". Note over SEI, PM, SP: end opt Note over SEI, PM, SP: Postcondition: Note over SEI, PM, SP: The Subsystem - Point is in an End position "X". PM->>SP: Information_No_End_Position activate SP opt [The Subsystem - Point is in the state OPERATIONAL.] SP->>SEI: Msg_Point_Position(no end position) end deactivate SP SP->>PM: Moving activate PM Note over PM: {<= Con_tmax_Point_Operation} PM->>SP: Information_End_Position_Detected deactivate PM SP->>PM: Stop_Moving activate PM deactivate PM opt [The Subsystem - Point is in the state OPERATIONAL.] SP->>SEI: Msg_Point_Position(Endposition X) end deactivate SP</pre>	Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE. Only applicable if the package [Option Redrive] is used in combination with [Basic non-4-wire single P].	Option Redrive
Eu.P.6718	Info	P_UC2.1.1.3: Irregularities	The Subsystem-UseCase "P_UC2.1.1.3: Irregularities" defines the behaviour of the Subsystem - Point which works with a single point machine via non-4-wire interface, when an irregularity occurs.	Basic non-4-wire single P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.1284	Info	<div><p>Alternative Scenario: Handle and report Point operation timeout with position change with single point machine [P SD 2.1.1.3.1]</p><p>P UC2.1.1.3: Irregularities</p><p>Alternative Scenario: Handle and report Point operation timeout with position change with single point machine [P SD 2.1.1.3.1]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y", or - No end position, or - a Unintended position.</p><p>Interaction 2.1.1.3.1.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X".</p><p>2. The Subsystem - Point sends the Command to the Point machine to move the Point to an End position "X". At this moment the Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>Interaction 2.1.1.3.1.B:</p><p>opt [The Subsystem - Point was previously in an End position or a Unintended position]</p><p>3.a1 - The Subsystem - Point receives from the Point machine the Information that the Point is in No end position.</p><p>3.a2 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</p><p>end opt</p><p>Interaction 2.1.1.3.1.C:</p><p>4. - The Subsystem - Point detects that the time period Con_tmax_Point_Operation has exceeded and then sends the Command to the Point machine to stop moving the Point machine.</p><p>5. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that a Failed Movement has occurred.</p><p>Postcondition:</p><p>The Subsystem - Point is in No end position.</p></div> <pre>sequenceDiagram participant I as Subsystem - Electronic Interlocking participant P as Point machine participant S as :Subsystem - Point I->>S: Cd_Move_Point(End Position X) activate S S->>P: Moving activate P P->>S: after {Con_tmax_Point_Operation} deactivate P opt S->>I: Information_No_End_Position activate I I->>S: Msg_Point_Position(No End Position) deactivate I end S->>P: Stop_Moving deactivate S P->>I: Msg_Movement_Failed deactivate P deactivate I</pre>	Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.	Basic non-4-wire single P
Eu.P.6719	Info	<div><p>Alternative Scenario: Handle and report failed movement with single point machine without position change [P SD 2.1.1.3.2]</p><p>P UC2.1.1.3: Irregularities</p><p>Alternative Scenario: Handle and report failed movement with single point machine without position change [P SD 2.1.1.3.2]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y", - No end position, or - a Unintended position.</p><p>Interaction 2.1.1.3.2.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X".</p><p>2. The Subsystem - Point sends the Command to the Point machine to move the Point to an End position "X". At this moment the Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>Interaction 2.1.1.3.2.B:</p><p>3. - A failure occurred during the movement resulting in a Failed Movement. The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation.</p><p>4. The Subsystem - Point sends the Command to the Point machine to stop moving the Point machine.</p><p>5. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that a Failed Movement has occurred.</p><p>Postcondition:</p><p>---</p></div> <pre>sequenceDiagram participant I as Subsystem - Electronic Interlocking participant P as Point machine participant S as :Subsystem - Point I->>S: Cd_Move_Point(End Position X) activate S S->>P: Moving activate P P->>S: {<= Con_tmax_Point_Operation} deactivate P S->>I: Movement_Failed deactivate S S->>P: Stop_Moving deactivate S P->>I: Msg_Movement_Failed deactivate P deactivate I</pre>	Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.	Basic non-4-wire single P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.1474	Info	<div><p>Alternative Scenario: Handle and report No end position with single point machine [P SD 2.1.1.3.3]</p><p>P UC2.1.1.3: Irregularities</p><p>Alternative Scenario: Handle and report No end position with single point machine [P SD 2.1.1.3.3]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL.</p><p>The Subsystem - Point is configured with a non-4-wire interface to the Point machine.</p><p>The Subsystem - Point is in:</p><ul style="list-style-type: none">- an End position "Y", or- a Unintended position<p>Interaction 2.1.1.3.3.A:</p><ol style="list-style-type: none">- The Subsystem - Point receives from the Point machine the Information that the Point machine is in No end position.- The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.<p>Postcondition:</p><p>The Subsystem - Point is in No end position.</p></div> <div><pre>sequenceDiagram participant S as Subsystem - Electronic Interlocking participant P as Point machine participant SP as :Subsystem - Point P->>SP: Information_No_End_Position SP-->>S: Msg_Point_Position(No End Position)</pre></div>		Basic non-4-wire single P
Eu.P.1273	Info	<div><p>Alternative Scenario: Handle and report Unintended position with single point machine [P SD 2.1.1.3.4]</p><p>P UC2.1.1.3: Irregularities</p><p>Alternative Scenario: Handle and report Unintended position with single point machine [P SD 2.1.1.3.4]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL.</p><p>The Subsystem - Point is configured with a non-4-wire interface to the Point machine.</p><p>The Subsystem - Point is in:</p><ul style="list-style-type: none">- an End position "Y", or- No end position.<p>Interaction 2.1.1.3.4.A:</p><ol style="list-style-type: none">- The Subsystem - Point receives from the Point machine the Information that the Point machine is in a Unintended position.- The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in a Unintended position.<p>Postcondition:</p><p>The Subsystem - Point is in a Unintended position.</p></div> <div><pre>sequenceDiagram participant S as Subsystem - Electronic Interlocking participant P as Point machine participant SP as :Subsystem - Point P->>SP: Information_Unintended_Position SP-->>S: Msg_Point_Position(Unintended Position)</pre></div>		Basic non-4-wire single P
Eu.P.3207	Info	<div><p>Alternative Scenario: Handle and report End Position with single point machine [P SD 2.1.1.3.5]</p><p>P UC2.1.1.3: Irregularities</p><p>Alternative Scenario: Handle and report End Position with single point machine [P SD 2.1.1.3.5]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL.</p><p>The Subsystem - Point is configured with a non-4-wire interface to the Point machine.</p><p>The Subsystem - Point is in:</p><ul style="list-style-type: none">- No end position, or- a Unintended position.<p>Interaction 2.1.1.3.5.A:</p><ol style="list-style-type: none">- The Subsystem - Point receives from the Point machine the Information that the Point machine is in an End position.- The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in an End position "Y".<p>Postcondition:</p><p>The Subsystem - Point is in an End position "Y".</p></div> <div><pre>sequenceDiagram participant S as Subsystem - Electronic Interlocking participant P as Point machine participant SP as :Subsystem - Point P->>SP: Information_End_Position_Detected SP-->>S: Msg_Point_Position(End Position Y)</pre></div>		Basic non-4-wire single P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.4761	Info	<div><p>Alternative Scenario: Handle and report End Position out of the other End Position with single point machine [P SD 2.1.1.3.6]</p><p>P UC2.1.1.3: Irregularities</p><p>Alternative Scenario: Handle and report End Position out of the other End Position with single point machine [P SD 2.1.1.3.6]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is in an End position "Y".</p><p>Interaction 2.1.1.3.6.A:</p><p>1. - The Subsystem - Point receives from the Point machine the Information that the Point machine is in an End position "X".</p><p>2. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in an End position "X".</p><p>Postcondition:</p><p>The Subsystem - Point is in End position "X".</p></div> <pre>sequenceDiagram participant SIE as Subsystem - Electronic Interlocking participant PM as Point machine participant SP as :Subsystem - Point PM->>SP: Information_No_End_Position activate SP SP->>SIE: Msg_Point_Position(End Position X) deactivate SP</pre>		Basic non-4-wire single P
Eu.P.5373	Info	<div><p>Alternative Scenario: Handle and report loss of ability to move point with single point machine [P SD 2.1.1.3.7]</p><p>P UC2.1.1.3: Irregularities</p><p>Alternative Scenario: Handle and report loss of ability to move point with single point machine [P SD 2.1.1.3.7]</p><p>Precondition:</p><p>The Subsystem - Point is in the state INITIALISING or OPERATIONAL. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is configured to observe the Ability to move point. The Subsystem - Point is in: - Ability to move point</p><p>Interaction 2.1.1.3.7.A:</p><p>alt [The Point machine is Unable to move point]</p><p>1.a1 - The Subsystem - Point detects from the switch states of the Interface P3 to the Point machine that the Point has no Ability to move point.</p><p>else alt [The internal trigger indicates Unable to move point]</p><p>1.b1 - The Subsystem - Point internal trigger indicates that the Subsystem - Point is Unable to move point</p><p>end alt</p><p>opt [The Subsystem - Point is in the state OPERATIONAL]</p><p>2.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that it is Unable to move point.</p><p>end opt</p><p>Interaction 2.1.1.3.7.B:</p><p>opt [The point machine is Moving]</p><p>3.a1 The Subsystem - Point sends the Command to the Point machine to Stop moving the Point machine.</p><p>end opt</p><p>Postcondition:</p><p>The Subsystem - Point is in: - Unable to move point</p></div> <pre>sequenceDiagram participant SIE as Subsystem - Electronic Interlocking participant PM as Point machine participant SP as :Subsystem - Point alt [The Point machine is Unable to move point] SP->>SP: Information_Ability_To_Move_Point_Available_FALSE SP->>SP: Information_Ability_To_Move_Point else [The internal trigger indicates Unable to move point] SP->>SP: Information_Ability_To_Move_Point_Available_FALSE end opt [The Subsystem - Point is in the state OPERATIONAL] SP->>SIE: Msg_Ability_To_Move_Point(Unable) end opt [The point machine is Moving] SP->>PM: Stop_Moving end</pre>	Only applicable if the package [Option Able to move] is used in combination with [Basic non-4-wire multiple P].	Option Able to move

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.5797	Info	<div><p>Alternative Scenario: Handle and report restoring of Ability to move point with single point machine [P SD 2.1.1.3.8]</p><p>P UC2.1.1.3: Irregularities</p><p>Alternative Scenario: Handle and report restoring of Ability to move point with single point machine [P SD 2.1.1.3.8]</p><p>Precondition:</p><p>The Subsystem - Point is in the state INITIALISING or OPERATIONAL.</p><p>The Subsystem - Point is configured with a non-4-wire interface to the Point machine.</p><p>The Subsystem - Point is configured to observe the Ability to move point.</p><p>The Subsystem - Point is in:</p><ul style="list-style-type: none">- Unable to move point<p>Interaction 2.1.1.3.8.A:</p><p>par</p><p>1.a1 - The Subsystem - Point detects from the switch states of the Interface P3 to the Point machine that the Point machine has regained Ability to move point.</p><p>also par</p><p>1.b1 - The Subsystem - Point internal trigger indicates that the Subsystem - Point is Able to move point.</p><p>end par</p><p>opt [The Subsystem - Point is in state OPERATIONAL]</p><p>2.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that it is Able to move point.</p><p>end opt</p><p>Postcondition:</p><p>The Subsystem - Point is in:</p><ul style="list-style-type: none">- Able to move point</div> <pre>sequenceDiagram participant SEI as Subsystem - Electronic Interlocking participant PM as Point machine participant SP as :Subsystem - Point par SEI->>SP: Information_Ability_To_Move_Point SP->>PM: Information_Ability_To_Move_Point_Available_TRUE and end opt [The Subsystem - Point is in state OPERATIONAL] SEI->>SP: Msg_Ability_To_Move_Point(Able) end</pre>	Only applicable if the package [Option Able to move] is used in combination with [Basic non-4-wire single P].	Option Able to move

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6716	Info	<div><div><div>Alternative Scenario: Handle Point operation timeout during Redrive Non 4W [P SD 2.1.1.3.9]</div><div>P UC2.1.1.3: Irregularities</div><div><div><div>Subsystem - Electronic Interlocking</div><div>Point machine</div><div>:Subsystem - Point</div></div><div><div>Alternative Scenario: Handle Point operation timeout during Redrive Non 4W [P SD 2.1.1.3.9]</div><div>Precondition: The Subsystem - Point is in the state INITIALISING or OPERATIONAL. The Subsystem - Point is in an End position "X" The last Cd_Move_Point Command received was also for End position "x" The last Cd_Move_Point Command received occurred since exiting the state BOOTING. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is configured as Redrive point.</div><div>Interaction 2.1.1.3.9.A: <div>1. - The Subsystem - Point receives from the Point machine the Information that the Point machine is in No end position.</div><div>opt [The Subsystem - Point is in the state OPERATIONAL.]<div><div>2.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</div></div></div><div>end opt</div><div><div>3. The Subsystem - Point sends the Command to the Point machine to move the Point back to an End position "X". At this moment the Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</div><div>Interaction 2.1.1.3.9.B: <div>4. The Subsystem - Point detects that the time period Con_tmax_Point_Operation has exceeded and then sends the Command to the Point machine to stop moving the Point machine.</div></div><div>opt [The Subsystem - Point is in the state OPERATIONAL.]<div><div>5.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking to indicate that a Failed Movement has occurred.</div></div></div><div>end opt</div><div>Postcondition: The Subsystem - Point is in No end position.</div></div></div></div></div></div></div>	<div><div>Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE. Only applicable if the package [Option Redrive] is used in combination with [Basic non-4-wire single P].</div></div>	<div><div>Option Redrive</div></div>

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7370	Info	<div><div>Alternative Scenario: Redrive Point trigger unable to move [P SD 2.1.1.3.10]</div><div>P UC2.1.1.3: Irregularities</div><div><div>Alternative Scenario: Redrive Point trigger unable to move [P SD 2.1.1.3.10]</div><div>Precondition: The Subsystem - Point is in the state INITIALISING or OPERATIONAL. The Subsystem - Point is in an End position "X". The last Cd_Move_Point Command received was also for End position "X". The last Cd_Move_Point Command received occurred since exiting the state BOOTING. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is configured as Redrive point. The Subsystem - Point is Unable to move point.</div><div>Interaction 2.1.1.3.10.A: 1. - The Subsystem - Point receives from the Point machine the Information that the Point machine is in No end position. opt [The Subsystem - Point is in the state OPERATIONAL.] 2.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position. end opt</div><div>Postcondition: The Subsystem - Point is in No end position.</div></div><div><pre>sequenceDiagram actor SIE as Subsystem - Electronic Interlocking actor PM as Point machine participant SP as :Subsystem - Point PM->>SP: Information_No_End_Position activate SP SP->>SIE: Msg_Point_Position(no end position) deactivate SP opt [The Subsystem - Point is in the state OPERATIONAL.] SP->>SIE: Msg_Point_Position(no end position) end</pre></div></div>	Only applicable if the package [Option Redrive] is used in combination with [Basic non-4-wire single P].	Option Redrive
Eu.P.6731	Info	P_UC2.1.2: Multiple point machines	The Subsystem-UseCase "P_UC2.1.2: Multiple point machines" defines the behaviour of the Subsystem - Point which works with multiple point machines via non-4-wire interface. The behaviour will be defined in the following UseCases: P_UC2.1.2.1: Commanding and reversing P_UC2.1.2.2: Redrive P_UC2.1.2.4: Irregularities	Basic non-4-wire multiple P
Eu.P.6721	Info	P_UC2.1.2.1: Commanding and reversing	The Subsystem-UseCase "P_UC2.1.2.1: Commanding and reversing" defines the behaviour of commanding and reversing a multiple point machine via non-4-wire interface.	Basic non-4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6722	Info	<div><p>Main Success Scenario: Moving of the Point with multiple Point machines Non 4W [P SD 2.1.2.1.1]</p><p>P UC2.1.2.1: Commanding and reversing</p><p>Main Success Scenario: Moving of the Point with multiple Point machines Non 4W [P SD 2.1.2.1.1]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y", or - No end position, or - a Unintended position.</p><p>Interaction 2.1.2.1.1.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X".</p><p>par</p><p>2.a1 The Subsystem - Point sends the Command to the 1st Point machine to move the Point to an End position "X".</p><p>also par</p><p>opt [The n-th Point machine has drive capability]</p><p>2.b1.a1 The Subsystem - Point sends the Command to the n-th Point machine to move the Point to an End position "X".</p><p>end opt</p><p>also par</p><p>2.c1 The Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>end par</p><p>Interaction 2.1.2.1.1.B:</p><p>alt [The 1st Point machine was previously in an End position or a Unintended position and the n-th is not in a Unintended position]</p><p>3.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the Point is in No end position.</p><p>3.a2 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</p><p>else alt [The n-th Point machine was previously in an End position or a Unintended position and the 1st Point machine is not in a Unintended position]</p><p>3.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the Point is in No end position.</p><p>3.b2 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</p><p>else alt [The 1st Point machine was previously in a Unintended position and the n-th Point machine was in a Unintended position]</p><p>par</p><p>3.c1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the Point is in No end position.</p><p>also par</p><p>3.c1.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the Point is in No end position.</p><p>end par</p><p>3.d1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</p><p>end alt</p><p>Interaction 2.1.2.1.1.C:</p><p>alt [The Subsystem - Point is configured for individual drive.]</p><p>4.a1 The Subsystem - Point stops moving of the Point for individual drive.</p><p>else alt [The Subsystem - Point is configured for common drive.]</p><p>4.b1 The Subsystem - Point stops moving of the Point for common drive.</p><p>end alt</p><p>5. The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation.</p><p>6. When Information_End_Position_Detected has been received from all Point machines, the Subsystem - Point sends a Message to the Subsystem - Electronic Interlocking indicating that the Point is in an End position "X".</p><p>Postcondition:</p><p>The Subsystem - Point is in an End position "X".</p></div>	<p>Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.</p> <p>The timing related to driving individual (1 to n-th) Point Machines for a single Point is supplier specific and is specified only in a general way. The timing related to individual Point Machines is not specified by EULYNX as part of the application layer, this shall be handled by the physical implementation.</p>	Basic non-4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6723	Info	P_UC2.1.2.2: Redrive	The Subsystem-UseCase "P_UC2.1.2.2: Redrive" defines the behaviour of redriving a multiple point machine via non-4-wire interface. Only applicable if the package [Option Redrive] is used in combination with [Basic non-4-wire multiple P].	Option Redrive

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7098	Info	<div><p>Main Success Scenario: Redrive Point after lost end position [P SD 2.1.2.2.1]</p><p>P UC2.1.2.2: Redrive</p><p>Main Success Scenario: Redrive Point after lost end position [P SD 2.1.2.2.1]</p><p>Precondition:</p><p>The Subsystem - Point is in the state INITIALISING or OPERATIONAL. The Subsystem - Point is in an End position "X". The last Cd_Move_Point Command received was also for End position "X". The last Cd_Move_Point Command received occurred since exiting the state BOOTING. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is configured as Redrive point.</p><p>Interaction 2.1.2.2.1.A:</p><p>alt [The 1st Point machine was previously in an End position]</p><div><p>1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the Point is in No end position.</p><p>opt [The Subsystem - Point is in the state OPERATIONAL.]</p><div><p>1.a2.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</p></div><p>end opt</p></div><p>else alt [The n-th Point machine was previously in an End position]</p><div><p>1.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the Point is in No end position.</p><p>opt [The Subsystem - Point is in the state OPERATIONAL.]</p><div><p>1.b2.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</p></div><p>end opt</p></div><p>end alt</p><p>Interaction 2.1.2.2.1.B:</p><p>alt [The Subsystem - Point is configured for common drive.]</p><div><p>par</p><div><p>2.a1.a1 - The Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p></div><p>also par</p><div><p>2.a1.b1 - The Subsystem - Point sends the Command to the 1st Point machine to move the Point to an End position "X".</p></div><p>also par</p><div><p>2.a1.c1 - The Subsystem - Point sends the Command to the n-th Point machine to move the Point to an End position "X".</p></div><p>end par</p></div><p>else alt [The Subsystem - Point is configured for individual drive.]</p><div><p>alt [The 1st Point machine is in No end position]</p><div><p>2.b1.a1 - The Subsystem - Point sends the Command to the 1st Point machine to move the Point to an End position "X".</p></div><p>else alt [The n-th Point machine is in No end position]</p><div><p>2.b1.b1 - The Subsystem - Point sends the Command to the n-th Point machine to move the Point to an End position "X".</p></div><p>end alt</p></div><p>end alt</p><p>Interaction 2.1.2.2.1.C:</p><p>alt [The Subsystem - Point is configured for individual drive]</p><div><p>3.a1 - The Subsystem - Point stops moving of the Point for individual drive.</p></div><p>else alt [The Subsystem - Point is configured for common drive]</p><div><p>3.b1 - The Subsystem - Point stops moving of the Point for common drive.</p></div><p>end alt</p><p>4. The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation.</p><p>opt [The Subsystem - Point is in the state OPERATIONAL.]</p><div><p>4.a1 When Information_End_Position_Detected has been received from all Point machines, the Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in an End position "X".</p></div><p>end opt</p><p>Postcondition:</p><p>The Subsystem - Point is in an End position "X".</p></div> <div></div>	<p>Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE. Only applicable if the package [Option Redrive] is used in combination with [Basic non-4-wire multiple P].</p>	<p>Option Redrive</p>

ID	Type	Requirement Part 1		Requirement Part 2	Func. Pkg.										
Eu.P.6726	Info	P_UC2.1.2.3: Stop point movement			Basic non-4-wire multiple P										
Eu.P.6728	Info	<div>Alternative Scenario: Stop moving of the Point for individual drive [P SD 2.1.2.3.1]</div> <div>P UC2.1.2.3: Stop point movement</div> <div>Alternative Scenario: Stop moving of the Point for individual drive [P SD 2.1.2.3.1]</div> <div>par</div> <div><div>1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the 1st Point machine is in an End position "X".</div><div>1.a2 The Subsystem - Point sends a Command to the 1st Point machine to stop moving the Point.</div></div> <div>also par</div> <div><div>1.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the n-th Point machine is in an End position "X".</div><div>1.b2 The Subsystem - Point sends a Command to the n-th Point machine to stop moving the Point.</div></div> <div>end par</div> <div></div> <div><div>Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE. The timing related to driving individual (1 to n-th) Point Machines for a single Point is supplier specific and is specified only in a general way. The timing related to individual Point Machines is not specified by EULYNX as part of the application layer, this shall be handled by the physical implementation. This SD is part of [P SD 2.1.2.2.1].</div></div> <div>Basic non-4-wire multiple P</div>			Eu.P.6727	Info	<div>Alternative Scenario: Stop moving of the Point for common drive [P SD 2.1.2.3.2]</div> <div>P UC2.1.2.3: Stop point movement</div> <div>Alternative Scenario: Stop moving of the Point for common drive [P SD 2.1.2.3.2]</div> <div>par</div> <div><div>1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the 1st Point machine is in an End position "X".</div><div>1.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the n-th Point machine is in an End position "X".</div></div> <div>end par</div> <div>par</div> <div><div>2.a1 The Subsystem - Point sends a Command to the 1st Point machine to stop moving the Point.</div><div>2.b1 The Subsystem - Point sends a Command to the n-th Point machine to stop moving the Point.</div></div> <div>end par</div> <div></div> <div><div>Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE. The timing related to driving individual (1 to n-th) Point Machines for a single Point is supplier specific and is specified only in a general way. The timing related to individual Point Machines is not specified by EULYNX as part of the application layer, this shall be handled by the physical implementation. This SD is part of [P SD 2.1.2.2.1]. Only applicable if the package [Option Common Drive] is used in combination with [Basic non-4-wire multiple P].</div></div> <div>Option Common Drive</div>			Eu.P.6729	Info	P_UC2.1.2.4: Irregularities			Basic non-4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.5805	Info	<div><div><div>Alternative Scenario: Handle and report No end position with multiple Point machines [P SD 2.1.2.4.1]</div><div><div><div>P UC2.1.2.4: Irregularities</div><div><div><div>Alternative Scenario: Handle and report No end position with multiple Point machines [P SD 2.1.2.4.1]</div><div>Precondition:<div>The Subsystem - Point is in the state OPERATIONAL.</div><div>The Subsystem - Point is configured with a non-4-wire interface to the Point machine.</div><div>The Subsystem - Point is in:<div>- an End position "Y", or</div><div>- a Unintended position</div></div></div></div><div>Interaction 2.1.2.4.1.A:</div><div>par<div>alt [Any Point machine is in an End position]<div>alt [The 1st Point machine is in No end position]<div>1.a1.a1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the 1st Point machine is in No end position.</div><div>else alt [The n-th Point machine is in No end position]<div>1.a1.a1.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the n-th Point machine is in No end position.</div></div></div><div>end alt</div><div>else alt [The Subsystem - Point is not configured with a 4-wire interface to the Point machine]<div>alt [The 1st Point machine is in an End position "Y".]<div>opt [The n-th Point machine is in an End position "X".]<div>1.a1.b1.a1.a1 - The Subsystem - Point receives from the n-th Point machine the Information that the n-th Point machine is in an End position "X".</div><div>end opt</div><div>else alt [The n-th Point machine is in an End position "Y"]<div>opt [The 1st Point machine is in an End position "X".]<div>1.a1.b1.b1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the 1st Point machine is in an End position "X".</div><div>end opt</div></div></div><div>end alt</div></div></div><div>also par<div>1.b1 - The 1st Point machine is NOT in a Unintended position.</div></div><div>also par<div>1.c1 - The n-th Point machine is NOT in a Unintended position.</div></div><div>end par</div><div>Interaction 2.1.2.4.1.B:</div><div>2. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</div><div>Postcondition:<div>The Subsystem - Point is in No end position.</div></div></div></div></div><div><div><div>Subsystem - Electronic Interlocking</div><div>Point machine</div><div>Point machine</div><div>:Subsystem - Point</div></div><div><div>1st</div><div>n-th</div></div><div><div>Information_No_End_Position</div><div>Information_No_End_Position</div><div>Information_End_Position_Detected</div><div>Information_End_Position_Detected</div><div>Msg_Point_Position(No End Position)</div></div></div></div></div></div></div></div>		Basic non-4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.5808	Info	<div><p>Alternative Scenario: Handle and report Unintended position with multiple Point machines [P SD 2.1.2.4.2]</p><p>P UC2.1.2.4: Irregularities</p><p>Alternative Scenario: Handle and report Unintended position with multiple Point machines [P SD 2.1.2.4.2]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y", or - No end position.</p><p>Interaction 2.1.2.4.2.A:</p><p>alt [The 1st Point machine is in a End position "Y" or No end position.] 1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the Point machine is in a Unintended position. else alt [The n-th Point machine is in a End position "Y" or No end position.] 1.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the Point machine is in a Unintended position. end alt</p><p>Interaction 2.1.2.4.2.B:</p><p>2. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in a Unintended position.</p><p>Postcondition:</p><p>The Subsystem - Point is in a Unintended position.</p></div> <pre>sequenceDiagram actor Interlocking as Subsystem - Electronic Interlocking actor PM1 as Point machine 1 st actor PMn as Point machine n-th participant SP as :Subsystem - Point alt [The 1st Point machine is in a End position "Y" or No end position.] PM1->>SP: Information_Unintended_Position else alt [The n-th Point machine is in a End position "Y" or No end position.] PMn->>SP: Information_Unintended_Position end alt SP->>Interlocking: Msg_Point_Position(Unintended Position)</pre>		Basic non-4-wire multiple P
Eu.P.5803	Info	<div><p>Alternative Scenario: Handle and report End Position with multiple Point machines [P SD 2.1.2.4.3]</p><p>P UC2.1.2.4: Irregularities</p><p>Alternative Scenario: Handle and report End Position with multiple Point machines [P SD 2.1.2.4.3]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is in: - a No end position, or - an Unintended position.</p><p>Interaction 2.1.2.4.3.A:</p><p>alt [The 1st Point machine is an No end position or a Unintended position and the n-th Point machine is in an End position "Y".] 1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the Point machine is in an End position "Y". else alt [The n-th Point machine is an No end position or a Unintended position and the 1st Point machine is in an End position "Y".] 1.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the Point machine is in an End position "Y". else alt [The n-th Point machine and the 1st Point machine are in an No end position or a Unintended position.] par 1.c1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the Point machine is in an End position "Y". also par 1.c1.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the Point machine is in an End position "Y". end par end alt</p><p>Interaction 2.1.2.4.3.B:</p><p>2. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in an End position "Y".</p><p>Postcondition:</p><p>The Subsystem - Point is in an End position "Y".</p></div> <pre>sequenceDiagram actor Interlocking as Subsystem - Electronic Interlocking actor PM1 as Point machine 1 st actor PMn as Point machine n-th participant SP as :Subsystem - Point alt [The 1st Point machine is an No end position or a Unintended position and the n-th Point machine is in an End position "Y".] PM1->>SP: Information_End_Position_Detected else alt [The n-th Point machine is an No end position or a Unintended position and the 1st Point machine is in an End position "Y".] PMn->>SP: Information_End_Position_Detected else alt [The n-th Point machine and the 1st Point machine are in an No end position or a Unintended position.] par PM1->>SP: Information_End_Position_Detected and PMn->>SP: Information_End_Position_Detected end end alt SP->>Interlocking: Msg_Point_Position(End Position Y)</pre>		Basic non-4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.5802	Info	<p>Alternative Scenario: Handle and report opposite End Position with multiple Point machines via non 4-wire interface [P SD 2.1.2.4.4]</p> <p>P UC2.1.2.4: Irregularities</p> <p>Alternative Scenario: Handle and report opposite End Position with multiple Point machines via non 4-wire interface [P SD 2.1.2.4.4]</p> <p>Precondition:</p> <p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is in an End position "Y".</p> <p>Interaction 2.1.2.4.4.A:</p> <pre> sequenceDiagram participant S as Subsystem - Electronic Interlocking participant P1 as Point machine 1 st participant Pn as Point machine n-th participant SP as Subsystem - Point par P1->>SP: Information_End_Position_Detected Pn->>SP: Information_End_Position_Detected and S->>SP: end SP->>S: Msg_Point_Position(End Position X) </pre> <p>2. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in an End position "X".</p> <p>Postcondition:</p> <p>The Subsystem - Point is in End position "X".</p>	Whether this SD can occur, depends on the national implementation of P3.	Basic non-4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.5804	Info	<div><p>Alternative Scenario: Handle and report loss of ability to move point with multiple Point machines [P SD 2.1.2.4.5]</p><p>P UC2.1.2.4: Irregularities</p><p>Alternative Scenario: Handle and report loss of ability to move point with multiple Point machines [P SD 2.1.2.4.5]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is configured to observe the Ability to move point. The Subsystem - Point is Able to move point.</p><p>Interaction 2.1.2.4.5.A:</p><p>alt [The 1st Point machine is Unable to move point]</p><p> 1.a1 - The Subsystem - Point detects from the switch states of the Interface P3 to the 1st Point machine that the 1st Point machine has no Ability to move point.</p><p>else alt [The n-th Point machine is configured with full functionality]</p><p> opt [The n-th Point machine is Unable to move point]</p><p> 1.b1.a1 - The Subsystem - Point detects from the switch states of the Interface P3 to the n-th Point machine that the n-th Point machine has no Ability to move point.</p><p> end opt</p><p>else alt [internal trigger indicates non-ability to move point]</p><p> 1.c1 - The Subsystem - Point internal trigger indicates that the Point is Unable to move point.</p><p>end alt</p><p>Interaction 2.1.2.4.5.B:</p><p>2. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that it is Unable to move point.</p><p>Interaction 2.1.2.4.5.C:</p><p>opt [The1st point machine is in Moving]</p><p> 3.a1 - The Subsystem - Point sends the Command to the 1st Point machine to Stop moving the 1st Point machine.</p><p>end opt</p><p>opt [The n-th Point machine is configured with full functionality and is in Moving]</p><p> 4.a1 - The Subsystem - Point sends the Command to the n-th Point machine to Stop moving the n-th Point machine.</p><p>end opt</p><p>Postcondition:</p><p>The Subsystem - Point is Unable to move point.</p></div> <div></div>	Only applicable if the package [Option Able to move] is used in combination with [Basic non-4-wire multiple P].	Option Able to move

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.5374	Info	<div><p>Alternative Scenario: Handle and report restoring of Ability to move point with multiple Point machines [P SD 2.1.2.4.6]</p><p>P UC2.1.2.4: Irregularities</p><p>Alternative Scenario: Handle and report restoring of Ability to move point with multiple Point machines [P SD 2.1.2.4.6]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is configured to observe the Ability to move point. The Subsystem - Point is Unable to move point.</p><p>Interaction 2.1.2.4.6.A:</p><p>par [The 1st Point machine is Able to move point]</p><p>1.a1 - The Subsystem - Point detects from the switch states of the Interface P3 to the 1st Point machine that the 1st Point machine has regained Ability to move point.</p><p>also par [The n-th Point machine is Able to move point and is configured with full functionality]</p><p>1.b1 - The Subsystem - Point detects from the switch states of the Interface P3 to the n-th Point machine that the n-th Point machine has regained Ability to move point.</p><p>also par [internal trigger indicates ability to move point]</p><p>1.c1 - The Subsystem - Point internal trigger indicates that the Point is Able to move point.</p><p>end par</p><p>Interaction 2.1.2.4.6.B:</p><p>2. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that it is Able to move point.</p><p>Postcondition:</p><p>The Subsystem - Point is Able to move point.</p></div> <pre>sequenceDiagram participant IE as Subsystem - Electronic Interlocking participant PM1 as Point machine 1 st participant PMn as Point machine n-th participant SP as :Subsystem - Point par SP->>PM1: Information_Ability_To_Move_Point SP->>PMn: Information_Ability_To_Move_Point SP->>SP: Information_Ability_To_Move_Point_Available_TRUE and end SP->>IE: Msg_Ability_To_Move_Point(Able)</pre>	Only applicable if the package [Option Able to move] is used in combination with [Basic non-4-wire multiple P].	Option Able to move

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7372	Info	<div><p>Alternative Scenario: Handle and report Degraded Point Position with multiple Point machines after End Position case 1 [P SD 2.1.2.4.7]</p><p>P UC2.1.2.4: Irregularities</p><p>Alternative Scenario: Handle and report Degraded Point Position with multiple Point machines after End Position case 1 [P SD 2.1.2.4.7]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL.</p><p>The Subsystem - Point is configured with a non-4-wire interface to the Point machine.</p><p>Subsystem - Point in an End position "X".</p><p>Interaction 2.1.2.4.7.A</p><p>alt</p><p>1.a1 - The Subsystem - Point receives from the 1st non-crucial Point machine the Information that the Point machine is in No end position.</p><p>else alt</p><p>1.b1 - The Subsystem - Point receives from the k-th non-crucial Point machine the Information that the Point machine is in No end position.</p><p>end alt</p><p>Interaction 2.1.2.4.7.B</p><p>2. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in a Degraded point position "X"</p><p>Postcondition:</p><p>The Subsystem - Point is in a Degraded point position "X"</p></div> <pre>sequenceDiagram participant SIE as Subsystem - Electronic Interlocking participant P1 as 1st crucial Point Machine participant Pi as i-th crucial Point Machine participant Pn1 as 1st non-crucial Point Machine participant Pnk as k-th non-crucial Point Machine participant SP as :Subsystem - Point alt Pn1->>SP: Information_No_End_Position or Pnk->>SP: Information_No_End_Position end SP->>SIE: Msg_Point_Position(No End Position, Degraded Position X)</pre>		Basic non-4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7373	Info	<div><p>Alternative Scenario: Handle and report Degraded Point Position with multiple Point machines after End Position case 2 [P SD 2.1.2.4.8]</p><p>P UC2.1.2.4: Irregularities</p><p>Alternative Scenario: Handle and report Degraded Point Position with multiple Point machines after End Position case 2 [P SD 2.1.2.4.8]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is in a Non degraded position. The Subsystem - Point is in an End position "X".</p><p>Interaction 2.1.2.4.8.A</p><p>alt</p><p>1.a1 - The Subsystem - Point receives from the non-crucial 2nd Point machine the Information that the non-crucial 2nd Point machine is in a Unintended position.</p><p>else alt</p><p>1.b1 - The Subsystem - Point receives from the non-crucial n-th Point machine the Information that the non-crucial n-th Point machine is in a Unintended position.</p><p>end alt</p><p>Interaction 2.1.2.4.8.B</p><p>2. - The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in a Degraded point position "X"</p><p>Postcondition:</p><p>The Subsystem - Point is in a Degraded point position "X"</p></div>		Basic non-4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7374	Info	<div><p>Alternative Scenario: Handle and report non-degraded Point Position with multiple Point machines [P SD 2.1.2.4.9]</p><p>P UC2.1.2.4: Irregularities</p><p>Alternative Scenario: Handle and report non-degraded Point Position with multiple Point machines [P SD 2.1.2.4.9]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is in a Degraded point position "X". One or more non-crucial Point machine are in No end position.</p><p>Interaction 2.1.2.4.9.A</p><p>par</p><p>1.a1 - The Subsystem - Point receives from the 1st non-crucial Point machine the Information that the Point machine is in End position "X". 1.b1 - The Subsystem - Point receives from the k-th non-crucial Point machine the Information that the Point machine is in End position "X".</p><p>end par</p><p>Interaction 2.1.2.4.9.B</p><p>2. - The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is not in a Degraded point position "X"</p><p>Postcondition:</p><p>The Subsystem - Point is in an End position "X". The Subsystem - Point is in a Non degraded position.</p></div> <pre>sequenceDiagram participant SIE as Subsystem - Electronic Interlocking participant P1 as 1st crucial Point Machine participant Pi as i-th crucial Point Machine participant Pn1 as 1st non-crucial Point Machine participant Pnk as k-th non-crucial Point Machine participant SP as Subsystem - Point par Pn1->>SP: Information_End_Position_Detected Pnk->>SP: Information_End_Position_Detected end SP->>SIE: Msg_Point_Position(End Position X, Non Degraded Position)</pre>		Basic non-4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7371	Info	<div><p>Alternative Scenario: Handle and report Degraded Point Position with multiple Point machines after crucial End Position [P SD 2.1.2.4.10]</p><p>P UC2.1.2.4: Irregularities</p><p>Alternative Scenario: Handle and report Degraded Point Position with multiple Point machines after crucial End Position [P SD 2.1.2.4.10]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL.</p><p>The Subsystem - Point is configured with a non-4-wire interface to the Point machine.</p><p>The Subsystem - Point is in a Non degraded position.</p><p>The Subsystem - Point is in No end position.</p><p>One or more non-crucial Point machine are in No end position.</p><p>Interaction 2.1.2.4.10.A</p><p>par</p><p>1.a1 - The Subsystem - Point receives from the crucial 1st Point machine the Information that the Point machine is in an End position "X".</p><p>1.b2 - The Subsystem - Point receives from the n-th crucial Point machine the Information that the Point machine is in an End position "X".</p><p>end par</p><p>Interaction 2.1.2.4.10.B</p><p>2. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in a Degraded point position "X"</p><p>Postcondition:</p><p>The Subsystem - Point is in a Degraded point position "X"</p></div> <pre>sequenceDiagram participant SIE as Subsystem - Electronic Interlocking participant P1 as 1st crucial Point Machine participant Pi as i-th crucial Point Machine participant Pnc1 as 1st non-crucial Point Machine participant Pncn as k-th non-crucial Point Machine participant SP as :Subsystem - Point par SP->>P1: Information_End_Position_Detected SP->>Pi: Information_End_Position_Detected end SP->>SIE: Msg_Point_Position(No End Position, Degraded Position X)</pre>		Basic non-4-wire multiple P
Eu.P.7377	Info	<div><p>Alternative Scenario: Handle and report non-degraded Point Position with multiple Point machines after crutial Unintended Position [P SD 2.1.2.4.11]</p><p>P UC2.1.2.4: Irregularities</p><p>Alternative Scenario: Handle and report non-degraded Point Position with multiple Point machines after crutial Unintended Position [P SD 2.1.2.4.11]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL.</p><p>The Subsystem - Point is in a Degraded point position "X"</p><p>One or more non-crucial Point machine are in No end position or a Unintended position.</p><p>The Subsystem - Point is configured with a 4-wire interface to the Point machine AND the last commanded position is End position "X".</p><p>Interaction 2.1.2.4.11.A</p><p>alt</p><p>1.a1 - The Subsystem - Point receives the Information that the 1st crucial Point machine is in an detected Unintended position.</p><p>else alt</p><p>1.b1 - The Subsystem - Point receives the Information that the n-th crucial Point machine is in an detected Unintended position.</p><p>end alt</p><p>Interaction 2.1.2.4.11.B</p><p>2. - The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is not in a Degraded point position "X"</p><p>Postcondition:</p><p>The Subsystem - Point is in a Non degraded position.</p></div> <pre>sequenceDiagram participant SIE as Subsystem - Electronic Interlocking participant P1 as 1st crucial Point Machine participant Pi as i-th crucial Point Machine participant Pnc1 as 1st non-crucial Point Machine participant Pncn as k-th non-crucial Point Machine participant SP as :Subsystem - Point alt SP->>P1: Information_Unintended_Position else alt SP->>Pi: Information_Unintended_Position end SP->>SIE: Msg_Point_Position(Unintended position, Non Degraded Position)</pre>		Basic non-4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7375	Info	<div><p>Alternative Scenario: Handle and report non-degraded Point Position with multiple Point machines after crucial No End Position [P SD 2.1.2.4.12]</p><p>P UC2.1.2.4: Irregularities</p><p>Alternative Scenario: Handle and report non-degraded Point Position with multiple Point machines after crucial No End Position [P SD 2.1.2.4.12]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is in a Degraded point position "X". One or more non-crucial Point machine are in No end position.</p><p>Interaction 2.1.2.4.12.A</p><p>alt</p><div><div>1.b1 - The Subsystem - Point receives from the 1st crucial Point machine the Information that the Point machine is in No end position.</div><div>else alt</div><div><div>1.c1 - The Subsystem - Point receives from the n-th crucial Point machine the Information that the Point machine is in No end position</div><div>end alt</div></div><p>Interaction 2.1.2.4.12.B</p><p>2. - The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is not in a Degraded point position "X".</p><p>Postcondition:</p><p>The Subsystem - Point is in a Non degraded position.</p></div><pre>sequenceDiagram participant SIE as Subsystem - Electronic Interlocking participant PM1 as 1st crucial Point Machine participant PMi as i-th crucial Point Machine participant PM1nc as 1st non-crucial Point Machine participant PMknc as k-th non-crucial Point Machine participant SP as :Subsystem - Point alt PM1->>SP: Information_No_End_Position or PMi->>SP: Information_No_End_Position end SP-->>SIE: Msg_Point_Position(No End Position, Non Degraded Position)</pre></div>		Basic non-4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7376	Info	<div><p>Alternative Scenario: Handle and report non-degraded Point Position with multiple Point machines after crucial Opposit Position [P SD 2.1.2.4.13]</p><p>P UC2.1.2.4: Irregularities</p><p>Alternative Scenario: Handle and report non-degraded Point Position with multiple Point machines after crucial Opposit Position [P SD 2.1.2.4.13]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is in a Degraded point position "X". One or more non-crucial Point machine are in No end position.</p><p>Interaction 2.1.2.4.13.A</p><p>alt</p><div><div>1.a1 - The Subsystem - Point receives from the 1st crucial Point machine the Information that the Point machine is in End position "Y".</div><div>else alt</div><div><div>1.b1 - The Subsystem - Point receives from the n-th crucial Point machine the Information that the Point machine is in End position "Y".</div><div>else alt</div><div><div>1.c1 - The Subsystem - Point receives from the 1st 1st non-crucial Point machine the Information that the Point machine is in End position "Y".</div><div>else alt</div><div><div>1.d1 - The Subsystem - Point receives from the n-th non/crucial Point machine the Information that the Point machine is in End position "Y".</div><div>end alt</div></div><p>Interaction 2.1.2.4.13.B</p><p>2. - The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is not in a Degraded point position "X"</p><p>Postcondition:</p><p>The Subsystem - Point is in a Non degraded position.</p></div><div></div></div></div></div>		Basic non-4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.1475	Info	<div><p>Alternative Scenario: Handle and report Point operation timeout with multiple Point machines with position change [P SD 2.1.2.4.14]</p><p>P UC2.1.2.4: Irregularities</p><p>Alternative Scenario: Handle and report Point operation timeout with multiple Point machines with position change [P SD 2.1.2.4.14]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y", - No end position, or - a Unintended position.</p><p>Interaction 2.1.2.4.14.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X".</p><p>par</p><p> 2.a1 The Subsystem - Point sends the Command to the 1st Point machine to move the 1st Point machine to an End position "X".</p><p>also par</p><p> 2.b1 The Subsystem - Point sends the Command to the n-th Point machine to move the n-th Point machine to an End position "X".</p><p>also par</p><p> 2.c1 The Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>end par</p><p>Interaction 2.1.2.4.14.B:</p><p>alt [The 1st Point machine was previously in an End position or a Unintended position]</p><p> 3.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the 1st Point machine is in No end position.</p><p> 3.a2 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</p><p>else alt [The n-th Point machine was previously in an End position or a Unintended position]</p><p> 3.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the n-th Point machine is in No end position.</p><p> 3.b2 The Subsystem - Point reports to the Subsystem - Electronic Interlocking indicating that the Point is in No end position.</p><p>end alt</p><p>Interaction 2.1.2.4.14.C:</p><p>4. - The Subsystem - Point detects that the time period Con_tmax_Point_Operation has exceeded.</p><p>par</p><p> 5.a1 The Subsystem - Point sends the Command to the 1st Point machine to stop moving the 1st Point machine.</p><p>also par</p><p> 5.b1 The Subsystem - Point sends the Command to the n-th Point machine to stop moving the n-th Point machine.</p><p>end par</p><p>6. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that a Failed Movement has occurred.</p><p>Postcondition:</p><p>The Subsystem - Point is in No end position.</p></div>	<p>Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.</p> <p>The timing related to driving individual (1 to n-th) Point Machines for a single Point is supplier specific and is specified only in a general way. The timing related to individual Point Machines is not specified by EULYNX as part of the application layer, this shall be handled by the physical implementation.</p>	Basic non-4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6730	Info	<div><p>Alternative Scenario: Handle and report failed movement with multiple Point machines without position change [P SD 2.1.2.4.15]</p><p>P UC2.1.2.4: Irregularities</p><p>Alternative Scenario: Handle and report failed movement with multiple Point machines without position change [P SD 2.1.2.4.15]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y", - No end position, or - a Unintended position.</p><p>Interaction 2.1.2.4.15.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X".</p><p>par</p><p> 2.a1 The Subsystem - Point sends the Command to the 1st Point machine to move the 1st Point machine to an End position "X".</p><p>also par</p><p> 2.b1 The Subsystem - Point sends the Command to the n-th Point machine to move the n-th Point machine to an End position "X".</p><p>also par</p><p> 2.c1 The Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>end par</p><p>Interaction 2.1.2.4.15.B:</p><p>3. - A failure occurred during the movement resulting in a Failed Movement.</p><p>par</p><p> 4.a1 - The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation.</p><p>also par</p><p> 4.b1 - The Subsystem - Point sends the Command to the 1st Point machine to stop moving the 1st Point machine.</p><p>also par</p><p> 4.c1 The Subsystem - Point sends the Command to the n-th Point machine to stop moving the n-th Point machine.</p><p>end par</p><p>5. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that a Failed Movement has occurred.</p><p>Postcondition:</p><p>---</p></div> <pre>sequenceDiagram participant IE as Subsystem - Electronic Interlocking participant PM1 as Point machine 1st participant PMn as Point machine n-th participant SP as :Subsystem - Point IE->>SP: Cd_Move_Point(End Position X) activate SP par SP->>PM1: Moving activate PM1 and SP->>PMn: Moving activate PMn end PM1-->>SP: deactivate PM1 PMn-->>SP: deactivate PMn SP->>SP: {<= Con_tmax_Point_Operation} SP->>SP: Movement_Failed SP->>IE: Msg_Movement_Failed deactivate SP</pre>	<p>Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.</p> <p>The timing related to driving individual (1 to n-th) Point Machines for a single Point is supplier specific and is specified only in a general way. The timing related to individual Point Machines is not specified by EULYNX as part of the application layer, this shall be handled by the physical implementation.</p>	Basic non-4-wire multiple P

Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.
The timing related to driving individual (1 to n-th) Point Machines for a single Point is supplier specific and is specified only in a general way. The timing related to individual Point Machines is not specified by EULYNX as part of the application layer, this shall be handled by the physical implementation.

Basic non-4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7378	Info	<div><div><div>Alternative Scenario: Redrive Point trigger unable to move [P SD 2.1.2.4.17]</div><div>P UC2.1.2.4: Irregularities</div><div><div>Alternative Scenario: Redrive Point trigger unable to move [P SD 2.1.2.4.17]</div><div>Precondition:</div><div>The Subsystem - Point is in the state INITIALISING or OPERATIONAL. The Subsystem - Point is in an End position "X". The last Cd_Move_Point Command received was also for End position "X". The last Cd_Move_Point Command received occurred since exiting the state BOOTING. The Subsystem - Point is configured with a non-4-wire interface to the Point machine. The Subsystem - Point is configured as Redrive point. The Subsystem - Point is Unable to move point.</div><div>Interaction 2.1.2.2.17.A:</div><div>alt [The 1st Point machine was previously in an End position]</div><div><div><div>1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the Point is in No end position.</div><div>opt [The Subsystem - Point is in the state OPERATIONAL.]</div><div><div>1.a2.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</div><div>end opt</div></div></div><div><div><div>else alt [The n-th Point machine was previously in an End position]</div><div><div>1.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the Point is in No end position.</div><div>opt [The Subsystem - Point is in the state OPERATIONAL.]</div><div><div>1.b2.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</div><div>end opt</div></div></div><div>end alt</div></div><div>Postcondition:</div><div>The Subsystem - Point is in No end position.</div></div></div></div><div></div></div></div>	Only applicable if the package [Option Redrive] is used in combination with [Basic non-4-wire multiple P].	Option Redrive
Eu.P.6770	Info	P_UC2.2: Point with 4-wire interface	The Subsystem-UseCase "P_UC2.2: Point with 4-wire interface" defines the behaviour of the Subsystem - Point which works with a 4-wire interface. The behaviour will be defined in the following UseCases: P_UC2.2.1: Single point machine P_UC2.2.2: Multiple point machines	Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6752	Info	P_UC2.2.1: Single point machine	The Subsystem-UseCase "P_UC2.2.1: Single point machine" defines the behaviour of the Subsystem - Point which works with a single point machine via 4-wire interface. The behaviour will be defined in the following UseCases: P_UC2.2.1.1: Commanding and reversing P_UC2.2.1.2: Irregularities	Basic 4-wire single P
Eu.P.6733	Info	P_UC2.2.1.1: Commanding and reversing	The Subsystem-UseCase "P_UC2.2.1.1: Commanding and reversing" defines the behaviour of commanding and reversing a single point machine via 4-wire interface.	Basic 4-wire single P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6737	Info	<div><p>Alternative Scenario: Reversing Point 4W after End position reached [P SD 2.2.1.1.3]</p><p>P UC2.2.1.1: Commanding and reversing</p><p>Alternative Scenario: Reversing Point 4W after End position reached [P SD 2.2.1.1.3]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a 4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y", or - No end position, or - an Unintended position.</p><p>Interaction 2.2.1.1.3.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X".</p><p>Interaction 2.2.1.1.3.B:</p><p>opt [The Subsystem - Point was previously in an End position or a Unintended position]</p><p>2.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that Point is in No end position.</p><p>end opt</p><p>3. The Subsystem - Point sends the Command to the Point machine to move the Point to an End position "X". At this moment the Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>4. - The Subsystem - Point receives from the Point machine the Information that the Point machine has reached End position "X" as unsafe information.</p><p>Interaction 2.2.1.1.3.C:</p><p>5. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "Y".</p><p>6. - The Subsystem - Point sends the Command to the Point machine to move the Point to an End position "Y". At this moment the Subsystem - Point restarts to monitor the time period Con_tmax_Point_Operation.</p><p>7. - The Subsystem - Point receives from the Point machine the Information that the Point machine has reached End position "Y" as unsafe information.</p><p>8. The Subsystem - Point sends the Command to the Point machine to stop moving the Point machine.</p><p>9. - The Subsystem - Point receives from the Point machine the Information that the Point machine is in a detected End position "Y". The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation.</p><p>10. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in an End position "Y".</p><p>Postcondition:</p><p>The Subsystem - Point is in an End position "Y".</p></div> <pre>sequenceDiagram participant IE as Subsystem - Electronic Interlocking participant PM as Point machine participant SP as :Subsystem - Poir IE->>SP: Cd_Move_Point(End Position X) activate SP opt SP->>IE: Msg_Point_Position(No End Position) deactivate SP end SP->>PM: Moving activate PM PM->>SP: Information_End_Position_Reached deactivate PM SP->>SP: {<= Con_tmax_Point_Operation} SP->>PM: Cd_Move_Point(End Position Y) activate PM PM->>SP: Moving deactivate PM SP->>PM: Information_End_Position_Reached deactivate SP PM->>SP: {<= Con_tmax_Point_Operation} SP->>PM: Stop_Moving deactivate SP PM->>SP: Information_End_Position_Detected deactivate PM SP->>IE: Msg_Point_Position(End Position Y) deactivate SP</pre>	Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.	Basic 4-wire single P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6738	Info	<div><p>Alternative Scenario: Reversing Point 4W before End position is detected[P SD 2.2.1.1.4]</p><p>P UC2.2.1.1: Commanding and reversing</p><p>Alternative Scenario: Reversing Point 4W before End position is detected[P SD 2.2.1.1.4]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a 4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y", or - No end position, or - an Unintended position.</p><p>Interaction 2.2.1.1.4.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X".</p><p>Interaction 2.2.1.1.4.B:</p><p>opt [The Subsystem - Point was previously in an End position or a Unintended position]</p><p>2.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that Point is in No end position.</p><p>end opt</p><p>3. The Subsystem - Point sends the Command to the Point machine to move the Point to an End position "X". At this moment the Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>4. - The Subsystem - Point receives from the Point machine the Information that the Point machine has reached End position "X" as unsafe information.</p><p>5. The Subsystem - Point sends the Command to the Point machine to stop moving the Point machine.</p><p>Interaction 2.2.1.1.4.C:</p><p>6. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "Y".</p><p>7. - The Subsystem - Point sends the Command to the Point machine to move the Point to an End position "Y". At this moment the Subsystem - Point restarts to monitor the time period Con_tmax_Point_Operation.</p><p>8. - The Subsystem - Point receives from the Point machine the Information that the Point machine has reached End position "Y" as unsafe information.</p><p>9. The Subsystem - Point sends the Command to the Point machine to stop moving the Point machine.</p><p>10. - The Subsystem - Point receives from the Point machine the Information that the Point machine is in a detected End position "Y". The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation.</p><p>11. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in an End position "Y".</p><p>Postcondition:</p><p>The Subsystem - Point is in an End position "Y".</p></div> <pre>sequenceDiagram participant IE as Subsystem - Electronic Interlocking participant PM as Point machine participant SP as :Subsystem - Poir IE->>SP: Cd_Move_Point(End Position X) activate SP opt [The Subsystem - Point was previously in an End position or a Unintended position] SP->>IE: Msg_Point_Position(No End Position) deactivate SP end SP->>PM: Moving activate PM PM->>SP: Information_End_Position_Reached deactivate PM SP->>PM: Stop_Moving activate PM PM-->>SP: {<= Con_tmax_Point_Operation} deactivate PM IE->>SP: Cd_Move_Point(End Position Y) activate SP SP->>PM: Moving activate PM PM->>SP: Information_End_Position_Reached deactivate PM SP->>PM: Stop_Moving activate PM PM-->>SP: {<= Con_tmax_Point_Operation} deactivate PM PM->>SP: Information_End_Position_Detected deactivate PM SP->>IE: Msg_Point_Position(End Position Y) deactivate SP</pre>	Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.	Basic 4-wire single P

Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.

Basic 4-wire single P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6734	Info	<div><div>Alternative Scenario: Moving of the Point with repeated command of moving 4W case 1[P SD 2.2.1.1.5]</div><div><p>P UC2.2.1.1: Commanding and reversing</p><p>Alternative Scenario: Moving of the Point with repeated command of moving 4W case 1[P SD 2.2.1.1.5]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a 4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y", or - No end position, or - an Unintended position.</p><p>Interaction 2.2.1.1.5.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X".</p><p>Interaction 2.2.1.1.5.B:</p><p>opt [The Subsystem - Point was previously in an End position or a Unintended position]</p><p>2.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</p><p>end opt</p><p>3. The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X".</p><p>4. The Subsystem - Point ignores the command from the Subsystem - Electronic Interlocking.</p><p>5. The Subsystem - Point sends the Command to the Point machine to move the Point machine to an End position "X". At this moment the Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>Interaction 2.2.1.1.5.C:</p><p>6. - The Subsystem - Point receives from the Point machine the Information that the Point machine has reached End position "X" as unsafe information.</p><p>7. The Subsystem - Point sends the Command to the Point machine to stop moving the Point machine.</p><p>8. The Subsystem - Point receives from the Point machine the Information that the Point machine is in a detected End position "X". The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation.</p><p>9. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in an End position "X".</p><p>Postcondition:</p><p>The Subsystem - Point is in an End position "X".</p></div></div> <pre>sequenceDiagram participant IE as Subsystem - Electronic Interlocking participant PM as Point machine participant SP as Subsystem - Point IE->>SP: Cd_Move_Point(End Position X) activate SP opt [The Subsystem - Point was previously in an End position or a Unintended position] SP->>IE: Msg_Point_Position(No End Position) end IE->>SP: Cd_Move_Point(End Position X) activate SP SP->>PM: Moving activate PM PM->>SP: Information_End_Position_Reached deactivate PM SP->>PM: {<= Con_tmax_Point_Operation} PM->>SP: Stop_Moving deactivate PM SP->>PM: Information_End_Position_Detected deactivate PM SP->>IE: Msg_Point_Position(End Position X) deactivate SP</pre>	Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.	Basic 4-wire single P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6735	Info	<div><p>Alternative Scenario: Moving of the Point with repeated command of moving 4W case 2[P SD 2.2.1.1.6]</p><p>P UC2.2.1.1: Commanding and reversing</p><p>Alternative Scenario: Moving of the Point with repeated command of moving 4W case 2[P SD 2.2.1.1.6]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL.</p><p>The Subsystem - Point is configured with a 4-wire interface to the Point machine.</p><p>The Subsystem - Point is in:</p><ul style="list-style-type: none">- an End position "Y", or- No end position, or- an Unintended position.<p>Interaction 2.2.1.1.6.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X".</p><p>Interaction 2.2.1.1.6.B:</p><p>opt [The Subsystem - Point was previously in an End position or a Unintended position]</p><p>2.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</p><p>end opt</p><p>3. The Subsystem - Point sends the Command to the Point machine to move the Point machine to an End position "X". At this moment the Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>4. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X".</p><p>5. The Subsystem - Point ignores the command from the Subsystem - Electronic Interlocking.</p><p>Interaction 2.2.1.1.6.C:</p><p>6. - The Subsystem - Point receives from the Point machine the Information that the Point machine has reached End position "Y" as unsafe information.</p><p>7. The Subsystem - Point sends the Command to the Point machine to stop moving the Point machine.</p><p>8. The Subsystem - Point receives from the Point machine the Information that the Point machine is in a detected End position "X". The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation.</p><p>9. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in an End position "X".</p><p>Postcondition:</p><p>The Subsystem - Point is in an End position "X".</p></div> <pre>sequenceDiagram participant IE as Subsystem - Electronic Interlocking participant PM as Point machine participant SP as :Subsystem - Point IE->>SP: Cd_Move_Point(End Position X) activate SP opt [The Subsystem - Point was previously in an End position or a Unintended position] SP->>IE: Msg_Point_Position(No End Position) end SP->>PM: Moving activate PM IE->>SP: Cd_Move_Point(End Position X) deactivate PM PM->>SP: Information_End_Position_Reached activate SP SP->>PM: {<= Con_tmax_Point_Operation} deactivate SP SP->>PM: Stop_Moving deactivate SP PM->>SP: Information_End_Position_Detected activate SP SP->>IE: Msg_Point_Position(End Position X) deactivate SP</pre>	Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.	Basic 4-wire single P
Eu.P.6741	Info	P_UC2.2.1.2: Irregularities	The Subsystem-UseCase "P_UC2.2.1.2: Irregularities" defines the behaviour of the Subsystem - Point which works with a single point machine via 4-wire interface, when an irregularity occurs.	Basic 4-wire single P
Eu.P.6746	Info	<div><p>Alternative Scenario: Handle and report No end position with single point machine [P SD 2.2.1.2.1]</p><p>P UC2.2.1.2: Irregularities</p><p>Alternative Scenario: Handle and report No end position with single point machine [P SD 2.2.1.2.1]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL.</p><p>The Subsystem - Point is configured with a 4-wire interface to the Point machine.</p><p>The Subsystem - Point is in:</p><ul style="list-style-type: none">- an End position "Y", or- a Unintended position<p>Interaction 2.2.1.2.1.A:</p><p>1. - The Subsystem - Point receives from the Point machine the Information that the Point machine is in No end position.</p><p>2. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</p><p>Postcondition:</p><p>The Subsystem - Point is in No end position.</p></div> <pre>sequenceDiagram participant IE as Subsystem - Electronic Interlocking participant PM as Point machine participant SP as :Subsystem - Point PM->>SP: Information_No_End_Position activate SP SP->>IE: Msg_Point_Position(No End Position) deactivate SP</pre>		Basic 4-wire single P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6751	Info	<div><p>Alternative Scenario: Handle and report Unintended position with single point machine [P SD 2.2.1.2.2]</p><p>P UC2.2.1.2: Irregularities</p><p>Alternative Scenario: Handle and report Unintended position with single point machine [P SD 2.2.1.2.2]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL.</p><p>The Subsystem - Point is configured with a 4-wire interface to the Point machine.</p><p>The Subsystem - Point is in:</p><ul style="list-style-type: none">- an End position "Y", or- No end position.<p>Interaction 2.2.1.2.2.A:</p><ol style="list-style-type: none">- The Subsystem - Point receives from the Point machine the Information that the Point machine is in a Unintended position.The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in a Unintended position.<p>Postcondition:</p><p>The Subsystem - Point is in a Unintended position.</p></div> <div><pre>sequenceDiagram participant S as Subsystem - Electronic Interlocking participant P as Point machine participant SP as :Subsystem - Point P->>SP: Information_Unintended_Position SP-->>S: Msg_Point_Position(Unintended Position)</pre></div>		Basic 4-wire single P
Eu.P.6742	Info	<div><p>Alternative Scenario: Handle and report commanded End Position with single point machine via 4-wire interface [P SD 2.2.1.2.3]</p><p>P UC2.2.1.2: Irregularities</p><p>Alternative Scenario: Handle and report commanded End Position with single point machine via 4-wire interface [P SD 2.2.1.2.3]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL.</p><p>The Subsystem - Point is configured with a 4-wire interface to the Point machine AND the last commanded position is End position "Y".</p><p>The Subsystem - Point is in:</p><ul style="list-style-type: none">- No end position, or- a Unintended position.<p>Interaction 2.2.1.2.3.A:</p><ol style="list-style-type: none">- The Subsystem - Point receives from the Point machine the Information that the Point machine is in an End position "Y".The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in an End position "Y".<p>Postcondition:</p><p>The Subsystem - Point is in an End position "Y".</p></div> <div><pre>sequenceDiagram participant S as Subsystem - Electronic Interlocking participant P as Point machine participant SP as :Subsystem - Point P->>SP: Information_End_Position_Detected SP-->>S: Msg_Point_Position(End Position Y)</pre></div>		Basic 4-wire single P
Eu.P.6747	Info	<div><p>Alternative Scenario: Handle and report opposite End Position with single point machine via 4-wire interface [P SD 2.2.1.2.4]</p><p>P UC2.2.1.2: Irregularities</p><p>Alternative Scenario: Handle and report opposite End Position with single point machine via 4-wire interface [P SD 2.2.1.2.4]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL.</p><p>The Subsystem - Point is configured with a 4-wire interface to the Point machine AND the last commanded position is End position "Y".</p><p>The Subsystem - Point is in:</p><ul style="list-style-type: none">- an End position "Y", or- No end position.<p>Interaction 2.2.1.2.4.A:</p><ol style="list-style-type: none">- The Subsystem - Point receives from the Point machine the pattern that represents the opposite End position "X", which is interpreted as Information that the Point machine is in an Unintended position.The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in a Unintended position.<p>Postcondition:</p><p>The Subsystem - Point is in a Unintended position.</p></div> <div><pre>sequenceDiagram participant S as Subsystem - Electronic Interlocking participant P as Point machine participant SP as :Subsystem - Point P->>SP: Information_Unintended_Position SP-->>S: Msg_Point_Position(Unintended Position)</pre></div>		Basic 4-wire single P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6745	Info	<div><p>Alternative Scenario: Handle and report loss of ability to move point with single point machine [P SD 2.2.1.2.5]</p><p>P UC2.2.1.2: Irregularities</p><p>Alternative Scenario: Handle and report loss of ability to move point with single point machine [P SD 2.2.1.2.5]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a 4-wire interface to the Point machine. The Subsystem - Point is configured to observe the Ability to move point. The Subsystem - Point is in: - Able to move point</p><p>Interaction 2.2.1.2.5.A:</p><p>1 - The Subsystem - Point internal trigger indicates that the Subsystem - Point is Unable to move point</p><p>2. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that it is Unable to move point.</p><p>Interaction 2.2.1.2.5.B:</p><p>opt [The point machine is in Moving]</p><p> 3.a1 - The Subsystem - Point sends the Command to the Point machine to Stop moving the Point machine.</p><p>end opt</p><p>Postcondition:</p><p>The Subsystem - Point is in: - Unable to move point</p></div> <pre>sequenceDiagram participant SIE as Subsystem - Electronic Interlocking participant PM as Point machine participant SP as :Subsystem - Point Note over SP: self-call SP->>SIE: Information_Ability_To_Move_Point_Available_FALSE SP->>SIE: Msg_Ability_To_Move_Point(Unable) opt [The point machine is in Moving] SP->>PM: Stop_Moving end</pre>	Only applicable if the package [Option Able to move] is used in combination with [Basic 4-wire single P].	Option Able to move
Eu.P.6750	Info	<div><p>Alternative Scenario: Handle and report restoring of Ability to move point with single point machine [P SD 2.2.1.2.6]</p><p>P UC2.2.1.2: Irregularities</p><p>Alternative Scenario: Handle and report restoring of Ability to move point with single point machine [P SD 2.2.1.2.6]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a 4-wire interface to the Point machine. The Subsystem - Point is configured to observe the Ability to move point. The Subsystem - Point is in: - Unable to move point</p><p>Interaction 2.2.1.2.6.A:</p><p>1. - The Subsystem - Point internal trigger indicates that the Subsystem - Point is Able to move point.</p><p>2. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that it is Able to move point.</p><p>Postcondition:</p><p>The Subsystem - Point is in: - Able to move point</p></div> <pre>sequenceDiagram participant SIE as Subsystem - Electronic Interlocking participant PM as Point machine participant SP as :Subsystem - Point Note over SP: self-call SP->>SIE: Information_Ability_To_Move_Point_Available_TRUE SP->>PM: Msg_Ability_To_Move_Point(Able)</pre>	Only applicable if the package [Option Able to move] is used in combination with [Basic 4-wire single P].	Option Able to move

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6749	Info	<div><p>Alternative Scenario: Handle and report Point operation timeout with position change 4W case 2 [P SD 2.2.1.2.8]</p><p>P UC2.2.1.2: Irregularities</p><p>Alternative Scenario: Handle and report Point operation timeout with position change 4W case 2 [P SD 2.2.1.2.8]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a 4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y", or - No end position, or - an Unintended position.</p><p>Interaction 2.2.1.2.8.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X".</p><p>opt [The Subsystem - Point was previously in an End position or a Unintended position]</p><p>2.a1 The Subsystem - Point sends the Information to the Subsystem - Electronic Interlocking that the Point is in No end position.</p><p>end opt</p><p>3. The Subsystem - Point sends the Command to the Point machine to move the Point to an End position "X". At this moment the Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>Interaction 2.2.1.2.8.B:</p><p>4. The Subsystem - Point detects that the time period Con_tmax_Point_Operation has exceeded and then sends the Command to the Point machine to stop moving the Point machine.</p><p>5. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that a Failed Movement has occurred.</p><p>6. - The Subsystem - Point receives from the Point machine the Information that the Point machine is in No end position.</p><p>Postcondition:</p><p>The Subsystem - Point is in a No end position.</p></div> <div></div>	Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.	Basic 4-wire single P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6743	Info	<div><p>Alternative Scenario: Handle and report failed movement without position change 4W case 1 [P SD 2.2.1.2.9]</p><p>P UC2.2.1.2: Irregularities</p><p>Alternative Scenario: Handle and report failed movement without position change 4W case 1 [P SD 2.2.1.2.9]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a 4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y" or - a Unintended position.</p><p>Interaction 2.2.1.2.9.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X".</p><p>opt [The Subsystem - Point was previously in an End position or a Unintended position]</p><p>2.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking indicating that the Point is in No end position.</p><p>end opt</p><p>3. The Subsystem - Point sends the Command to the Point machine to move the Point to an End position "X". At this moment the Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>Interaction 2.2.1.2.9.B:</p><p>4. - A failure occurred during the movement resulting in a Failed Movement. The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation.</p><p>5. The Subsystem - Point sends the Command to the Point machine to stop moving the Point.</p><p>6. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that a Failed Movement has occurred.</p><p>alt [The Subsystem - Point was previously in an End position]</p><p>7.a1 The Subsystem - Point receives from the Point machine the pattern that represents the End position "Y", which is interpreted as Information that the Point machine is in an Unintended position.</p><p>else alt [The Subsystem - Point was previously in an Unintended position]</p><p>7.b1 The Subsystem - Point receives from the Point machine the Information that the Point machine is in a Unintended position.</p><p>end alt</p><p>8. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in Unintended position.</p><p>Postcondition:</p><p>The Subsystem - Point is in a Unintended position.</p></div> <pre>sequenceDiagram participant IE as Subsystem - Electronic Interlocking participant PM as Point machine participant SP as :Subsystem - Point IE->>SP: Cd_Move_Point(End Position X) activate SP opt [The Subsystem - Point was previously in an End position or a Unintended position] SP->>IE: Msg_Point_Position(No End Position) deactivate SP end SP->>PM: Moving activate PM PM-->>SP: {<= Con_tmax_Point_Operation} deactivate PM SP->>SP: Movement_Failed SP->>IE: Stop_Moving deactivate SP SP->>IE: Msg_Movement_Failed deactivate SP alt [The Subsystem - Point was previously in an End position] SP->>SP: Information_Unintended_Position deactivate SP else alt [The Subsystem - Point was previously in an Unintended position] SP->>SP: Information_Unintended_Position deactivate SP end SP->>IE: Msg_Point_Position(Unintended Position) deactivate SP</pre>	Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.	Basic 4-wire single P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6754	Info	<div><p>Main Success Scenario: Moving of the Point with multiple Point machines 4W [P SD 2.2.2.1.1]</p><p>P UC2.2.2.1: Commanding and reversing</p><p>Main Success Scenario: Moving of the Point with multiple Point machines 4W [P SD 2.2.2.1.1]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a 4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y", - No end position, or a Unintended position. The Subsystem - Point is configured with a 4-wire interface to the Point machine</p><p>Interaction 2.2.2.1.1.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X". opt [The Subsystem - Point was previously in an End position or a Unintended position]</p><p>2.a1 The Subsystem - Point reports the Subsystem - Electronic Interlocking that the Point is in No end position.</p><p>end opt</p><p>par</p><p>3.a1 - The Subsystem - Point sends the Command to the 1st Point machine to move the Point to an End position "X".</p><p>also par</p><p>opt [The n-th Point machine has drive capability]</p><p>3.b1.a1 - The Subsystem - Point sends the Command to the n-th Point machine to move the Point to an End position "X".</p><p>end opt</p><p>also par</p><p>3.c1 - The Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>end par</p><p>Interaction 2.2.2.1.1.B:</p><p>par</p><p>4.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the 1st Point machine has reached End position "X" as unsafe information.</p><p>4.a2 The Subsystem - Point sends the Command to the 1st Point machine to stop moving the Point.</p><p>4.a3 - The Subsystem - Point receives the Information that the 1st Point machine is in an detected End position "X".</p><p>also par</p><p>opt [The n-th Point machine has drive capability]</p><p>4.b1.a1 - The Subsystem - Point receives from the n-th Point machine the Information that the n-th Point machine has reached End position "X" as unsafe information.</p><p>4.b1.a2 The Subsystem - Point sends the Command to the n-th Point machine to stop moving the Point.</p><p>end opt</p><p>4.b2 - The Subsystem - Point receives the Information that the n-th Point machine is in an detected End position "X".</p><p>end par</p><p>5. The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation.</p><p>6. - The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in an End position "X".</p><p>Postcondition:</p><p>The Subsystem - Point is in an End position "X".</p></div> <pre>sequenceDiagram participant IE as Subsystem - Electronic Interlocking participant PM1 as Point machine 1st participant PMn as Point machine n-th participant SP as :Subsystem - Point IE->>SP: Cd_Move_Point(End Position X) activate SP opt SP->>IE: Msg_Point_Position(No End Position) end deactivate SP par SP->>PM1: Moving activate PM1 and opt SP->>PMn: Moving activate PMn end end par SP-->>SP: {<= Con_tmax_Point_Operation} and par PM1->>SP: Information_End_Position_Reached deactivate PM1 and SP->>PM1: Stop_Moving end and PMn->>SP: Information_End_Position_Reached deactivate PMn and SP->>PMn: Stop_Moving end end deactivate SP SP->>IE: Msg_Point_Position(End Position X) deactivate SP</pre>	<p>Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE. The timing related to driving individual (1 to n-th) Point Machines for a single Point is supplier specific and is specified only in a general way. The timing related to individual Point Machines is not specified by EULYNX as part of the application layer, this shall be handled by the physical implementation.</p>	Basic 4-wire multiple P
Eu.P.6755	Info	<p>P_UC2.2.2.2: Irregularities</p>	<p>The Subsystem-UseCase "P_UC2.2.2.2: Irregularities" defines the behaviour of the Subsystem - Point which works with a multiple point machine via 4-wire interface,</p>	Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
			when an irregularity occurs. Note: For implementations of the Subsystem - Point with more than 2 Point Machine interfaces, there may be more combinations of inputs possible than shown in the SDs in this section.	
Eu.P.6762	Info	<div><p>Alternative Scenario: Handle and report No end position with multiple Point machines [P SD 2.2.2.2.1]</p><p><u>P UC2.2.2.2: Irregularities</u></p><p>Alternative Scenario: Handle and report No end position with multiple Point machines [P SD 2.2.2.2.1]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a 4-wire interface to the Point machine AND the last commanded position is End position "Y". The Subsystem - Point is in: - an End position "Y", or- a Unintended position.</p><p>Interaction 2.2.2.2.1.A:</p><p>par</p><p> alt [The 1st Point machine is in No end position]</p><p> 1.a1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the 1st Point machine is in No end position.</p><p> else alt [The n-th Point machine is in No end position]</p><p> 1.a1.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the n-th Point machine is in No end position.</p><p> end alt</p><p>also par</p><p> 1.b1 - The 1st Point machine is NOT in a Unintended position.</p><p>also par</p><p> 1.c1 - The n-th Point machine is NOT in a Unintended position.</p><p>end par</p><p>Interaction 2.2.2.2.1.B:</p><p>2. - The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</p><p>Postcondition:</p><p>The Subsystem - Point is in No end position.</p></div> <pre>sequenceDiagram participant IE as Subsystem - Electronic Interlocking participant PM1 as Point machine 1st participant PMn as Point machine n-th participant SP as :Subsystem - Point par alt [The 1st Point machine is in No end position] PM1->>SP: Information_No_End_Position else alt [The n-th Point machine is in No end position] PMn->>SP: Information_No_End_Position end alt and 1.b1 - The 1st Point machine is NOT in a Unintended position. and 1.c1 - The n-th Point machine is NOT in a Unintended position. end SP->>IE: Msg_Point_Position(No End Position)</pre>		Basic 4-wire multiple P
Eu.P.6768	Info	<div><p>Alternative Scenario: Handle and report Unintended position with multiple Point machines [P SD 2.2.2.2.2]</p><p><u>P UC2.2.2.2: Irregularities</u></p><p>Alternative Scenario: Handle and report Unintended position with multiple Point machines [P SD 2.2.2.2.2]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a 4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y", or - No end position.</p><p>Interaction 2.2.2.2.2.A:</p><p>alt [The 1st Point machine is in an Unintended position.]</p><p> 1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the Point machine is in a Unintended position.</p><p>else alt [The n-th Point machine is in a Unintended position.]</p><p> 1.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the Point machine is in a Unintended position.</p><p>end alt</p><p>Interaction 2.2.2.2.2.B:</p><p>2. - The Subsystem - Point reports to the Subsystem - Electronic Interlocking indicating that the Point is in a Unintended position.</p><p>Postcondition:</p><p>The Subsystem - Point is in a Unintended position.</p></div> <pre>sequenceDiagram participant IE as Subsystem - Electronic Interlocking participant PM1 as Point machine 1 st participant PMn as Point machine n-th participant SP as :Subsystem - Point alt [The 1st Point machine is in an Unintended position.] PM1->>SP: Information_Unintended_Position else alt [The n-th Point machine is in a Unintended position.] PMn->>SP: Information_Unintended_Position end alt SP->>IE: Msg_Point_Position(Unintended Position)</pre>		Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6758	Info	<div><p>Alternative Scenario: Handle and report End Position with multiple Point machines [P SD 2.2.2.2.3]</p><p>P UC2.2.2.2: Irregularities</p><p>Alternative Scenario: Handle and report End Position with multiple Point machines [P SD 2.2.2.2.3]</p><p>Precondition:</p><p>he Subsystem - Point is in the state OPERATIONAL.</p><p>The Subsystem - Point is configured with a 4-wire interface to the Point machine AND the last commanded position is End position "Y".</p><p>The Subsystem - Point is in:</p><ul style="list-style-type: none">- an No end position, or- a Unintended position.<p>Interaction 2.2.2.2.3.A:</p><p>alt [The 1st Point machine is not in an End position "Y" and the n-th Point machine is in an End position "Y".]</p><p>1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the Point machine is in an End position "Y".</p><p>else alt [The n-th Point machine is not in an End position "Y" and the 1st Point machine is in an End position "Y".]</p><p>1.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the Point machine is in an End position "Y".</p><p>else alt [The n-th Point machine is not in an End position "Y" and the 1st Point machine is not in an End position "Y".]</p><p>par</p><p>1.c1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the Point machine is in an End position "Y".</p><p>also par</p><p>1.c1.b1 - The Subsystem - Point receives from the n-th Point machine the Information that the Point machine is in an End position "Y".</p><p>end par</p><p>end alt</p><p>Interaction 2.2.2.2.3.B:</p><p>2. - The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in an End position "Y".</p><p>Postcondition:</p><p>The Subsystem - Point is in an End position "Y".</p></div> <pre>sequenceDiagram participant SIE as Subsystem - Electronic Interlocking participant PM1 as Point machine 1 st participant PMn as Point machine n-th participant SP as :Subsystem - Point alt [The 1st Point machine is not in an End position "Y" and the n-th Point machine is in an End position "Y".] PM1->>SP: Information_End_Position_Detected else alt [The n-th Point machine is not in an End position "Y" and the 1st Point machine is in an End position "Y".] PMn->>SP: Information_End_Position_Detected else alt [The n-th Point machine is not in an End position "Y" and the 1st Point machine is not in an End position "Y".] par PM1->>SP: Information_End_Position_Detected PMn->>SP: Information_End_Position_Detected end end SP->>SIE: Msg_Point_Position(End Position Y)</pre>		Basic 4-wire multiple P
Eu.P.6764	Info	<div><p>Alternative Scenario: Handle and report opposite End Position with multiple Point machines via 4-wire interface [P SD 2.2.2.2.4]</p><p>P UC2.2.2.2: Irregularities</p><p>Alternative Scenario: Handle and report opposite End Position with multiple Point machines via 4-wire interface [P SD 2.2.2.2.4]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL.</p><p>The Subsystem - Point is configured with a 4-wire interface to the Point machine AND the last commanded position is End position "Y".</p><p>The Subsystem - Point is in:</p><ul style="list-style-type: none">- an End position "Y", or- No end position.<p>Interaction 2.2.2.2.4.A:</p><p>alt [The 1st Point machine is in End position "X".]</p><p>1.a1 - The Subsystem - Point receives from the 1st Point machine the pattern that represents the opposite End position "X", which is interpreted as Information that the Point machine is in an Unintended position.</p><p>else alt [The n-th Point machine is in End position "X".]</p><p>1.b1 - The Subsystem - Point receives from the n-th Point machine the pattern that represents the opposite End position "X", which is interpreted as Information that the Point machine is in an Unintended position.</p><p>end alt</p><p>Interaction 2.2.2.2.4.B:</p><p>2. - The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in a Unintended position.</p><p>Postcondition:</p><p>The Subsystem - Point is in a Unintended position.</p></div> <pre>sequenceDiagram participant SIE as Subsystem - Electronic Interlocking participant PM1 as Point machine 1 st participant PMn as Point machine n-th participant SP as :Subsystem - Point alt [The 1st Point machine is in End position "X".] PM1->>SP: Information_Unintended_Position else alt [The n-th Point machine is in End position "X".] PMn->>SP: Information_Unintended_Position end SP->>SIE: Msg_Point_Position(Unintended Position)</pre>	Note: The Subsystem - Point receives from the Point machine the pattern that represents the opposite end position "X", which is interpreted as Information that the Point machine is in an Unintended position.	Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6761	Info	<div><p>Alternative Scenario: Handle and report loss of ability to move point with multiple Point machines [P SD 2.2.2.2.5]</p><p>P UC2.2.2.2: Irregularities</p><p>Alternative Scenario: Handle and report loss of ability to move point with multiple Point machines [P SD 2.2.2.2.5]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL.</p><p>The Subsystem - Point is configured with a 4-wire interface to the Point machine.</p><p>The Subsystem - Point is configured to observe the Ability to move point.</p><p>The Subsystem - Point is Able to move point.</p><p>Interaction 2.2.2.2.5.A:</p><p>1. - The Subsystem - Point internal trigger indicates that the Point is Unable to move point.</p><p>2. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that it is Unable to move point.</p><p>Interaction 2.2.2.2.5.B:</p><p>par</p><p>opt [The 1st point machine is in Moving]</p><p>3.a1.a1 - The Subsystem - Point sends the Command to the 1st Point machine to Stop moving the 1st Point machine.</p><p>end opt</p><p>also par</p><p>opt [The n-th Point machine is in Moving and has drive capability]</p><p>3.b1.a1 - The Subsystem - Point sends the Command to the n-th Point machine to Stop moving the n-th Point machine.</p><p>end opt</p><p>end par</p><p>Postcondition:</p><p>The Subsystem - Point is Unable to move point.</p></div> <pre>sequenceDiagram participant SEI as Subsystem - Electronic Interlocking participant PM1 as Point machine 1st participant PMn as Point machine n-th participant SP as :Subsystem - Point Note over SP: Information_Ability_To_Move_Point_Available_FALSE SP->>SEI: Msg_Ability_To_Move_Point(Unable) activate SEI deactivate SEI par opt [The 1st point machine is in Moving] SEI->>PM1: Stop_Moving activate PM1 deactivate PM1 and opt [The n-th Point machine is in Moving and has drive capability] SEI->>PMn: Stop_Moving activate PMn deactivate PMn end end end</pre>	Only applicable if the package [Option Able to move] is used in combination with [Basic 4-wire multiple P].	Option Able to move
Eu.P.6767	Info	<div><p>Alternative Scenario: Handle and report restoring of Ability to move point with multiple Point machines [P SD 2.2.2.2.6]</p><p>P UC2.2.2.2: Irregularities</p><p>Alternative Scenario: Handle and report restoring of Ability to move point with multiple Point machines [P SD 2.2.2.2.6]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL.</p><p>The Subsystem - Point is configured with a 4-wire interface to the Point machine.</p><p>The Subsystem - Point is configured to observe the Ability to move point.</p><p>The Subsystem - Point is Unable to move point.</p><p>Interaction 2.2.2.2.6.A:</p><p>1. - The Subsystem - Point internal trigger indicates that the Point is Able to move point.</p><p>2. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that it is Able to move point.</p><p>Postcondition:</p><p>The Subsystem - Point is Able to move point.</p></div> <pre>sequenceDiagram participant SEI as Subsystem - Electronic Interlocking participant PM1 as Point machine 1 st participant PMn as Point machine n-th participant SP as :Subsystem - Point Note over SP: Information_Ability_To_Move_Point_Available_TRUE SP->>SEI: Msg_Ability_To_Move_Point(Able) activate SEI deactivate SEI</pre>	Only applicable if the package [Option Able to move] is used in combination with [Basic 4-wire multiple P].	Option Able to move

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7380	Info	<div><p>Alternative Scenario: Handle and report Degraded Point Position with multiple Point machines after End Position case 1 [P SD 2.2.2.2.7]</p><p>P UC2.2.2.2: Irregularities</p><p>Alternative Scenario: Handle and report Degraded Point Position with multiple Point machines after End Position case 1 [P SD 2.2.2.2.7]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is in a Non degraded position. The Subsystem - Point is in an End position "X".</p><p>Interaction 2.2.2.2.7.A</p><p>alt</p><p>1.a1 - The Subsystem - Point receives from the non-crucial 2nd Point machine the Information that the non-crucial 2nd Point machine is in No end position.</p><p>else alt</p><p>1.b1 - The Subsystem - Point receives from the non-crucial n-th Point machine the Information that the non-crucial n-th Point machine is in No end position.</p><p>end alt</p><p>Interaction 2.2.2.2.7.B</p><p>2. - The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in a Degraded point position "X"</p><p>Postcondition:</p><p>The Subsystem - Point is in a Degraded point position "X"</p></div> <pre>sequenceDiagram participant IE as Subsystem - Electronic Interlocking participant P1 as 1st crucial Point Machine participant Pi as i-th crucial Point Machine participant P2 as 1st non-crucial Point Machine participant Pk as k-th non-crucial Point Machine participant SP as :Subsystem - Point alt alt SP-->>P2: Information_No_End_Position else alt SP-->>Pk: Information_No_End_Position end end SP->>IE: Msg_Point_Position(No End Position, Degraded Position X)</pre>		Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7381	Info	<div><p>Alternative Scenario: Handle and report Degraded Point Position with multiple Point machines after End Position case 2 [P SD 2.2.2.2.8]</p><p>P UC2.2.2.2: Irregularities</p><p>Alternative Scenario: Handle and report Degraded Point Position with multiple Point machines after End Position case 2 [P SD 2.2.2.2.8]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is in a Non degraded position. The Subsystem - Point is in an End position "X".</p><p>Interaction 2.2.2.2.8.A</p><p>alt</p><p>1.a1 - The Subsystem - Point receives from the non-crucial 2nd Point machine the Information that the non-crucial 2nd Point machine is in a Unintended position.</p><p>else alt</p><p>1.b1 - The Subsystem - Point receives from the non-crucial n-th Point machine the Information that the non-crucial n-th Point machine is in a Unintended position.</p><p>end alt</p><p>Interaction 2.2.2.2.8.B</p><p>2. - The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in a Degraded point position "X"</p><p>Postcondition:</p><p>The Subsystem - Point is in a Degraded point position "X"</p></div>		Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7382	Info	<div><p>Alternative Scenario: Handle and report non-degraded Point Position with multiple Point machines [P SD 2.2.2.2.9]</p><p>P UC2.2.2.2: Irregularities</p><p>Alternative Scenario: Handle and report non-degraded Point Position with multiple Point machines [P SD 2.2.2.2.9]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a 4-wire interface to the Point machine AND the last commanded position is End position "X". The Subsystem - Point is in a Degraded point position "X" One or more non-crucial Point machine are in No end position or a Unintended position.</p><p>Interaction 2.2.2.2.9.A</p><p>par</p><p>1.a1 - The Subsystem - Point receives the Information that the 1st non-crucial Point machine is in an detected End position "X".</p><p>1.b1 - The Subsystem - Point receives the Information that the k-th non-crucial Point machine is in an detected End position "X".</p><p>end par</p><p>Interaction 2.2.2.2.9.B</p><p>2. - The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is not in a Degraded point position "X"</p><p>Postcondition:</p><p>The Subsystem - Point is in a Non degraded position.</p></div> <pre>sequenceDiagram participant SEI as Subsystem - Electronic Interlocking participant PM1 as 1st crucial Point Machine participant PMi as i-th crucial Point Machine participant PM1nc as 1st non-crucial Point Machine participant PMknc as k-th non-crucial Point Machine participant SP as :Subsystem - Point par PM1nc->>SP: Information_End_Position_Detected PMknc->>SP: Information_End_Position_Detected and SEI->>SP: Msg_Point_Position(End Position X, Non Degraded Position) end</pre>		Basic 4-wire multiple P
Eu.P.7379	Info	<div><p>Alternative Scenario: Handle and report Degraded Point Position with multiple Point machines after crucial End Positon - case 1 [P SD 2.2.2.2.10]</p><p>P UC2.2.2.2: Irregularities</p><p>Alternative Scenario: Handle and report Degraded Point Position with multiple Point machines after crucial End Positon - case 1 [P SD 2.2.2.2.10]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a 4-wire interface to the Point machine AND the last commanded position is End position "X". The Subsystem - Point is in a Non degraded position. One or more crucial Point machine are in No end position or a Unintended position. One or more non-crucial Point machine are in No end position.</p><p>Interaction 2.2.2.2.10.A</p><p>par</p><p>1.a1 - The Subsystem - Point receives the Information that the 1st crucial Point machine is in an detected End position "X".</p><p>also par</p><p>1.b1 - The Subsystem - Point receives the Information that the n-th crucial Point machine is in an detected End position "X".</p><p>end par</p><p>Interaction 2.2.2.2.10.B</p><p>2. - The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in a Degraded point position "X".</p><p>Postcondition:</p><p>The Subsystem - Point is in a Degraded point position "X"</p></div> <pre>sequenceDiagram participant SEI as Subsystem - Electronic Interlocking participant PM1 as 1st crucial Point Machine participant PMi as i-th crucial Point Machine participant PM1nc as 1st non-crucial Point Machine participant PMknc as k-th non-crucial Point Machine participant SP as :Subsystem - Point par PM1->>SP: Information_End_Position_Detected PMi->>SP: Information_End_Position_Detected and SEI->>SP: Msg_Point_Position(No End Position, Degraded Position X) end</pre>		Basic 4-wire multiple P

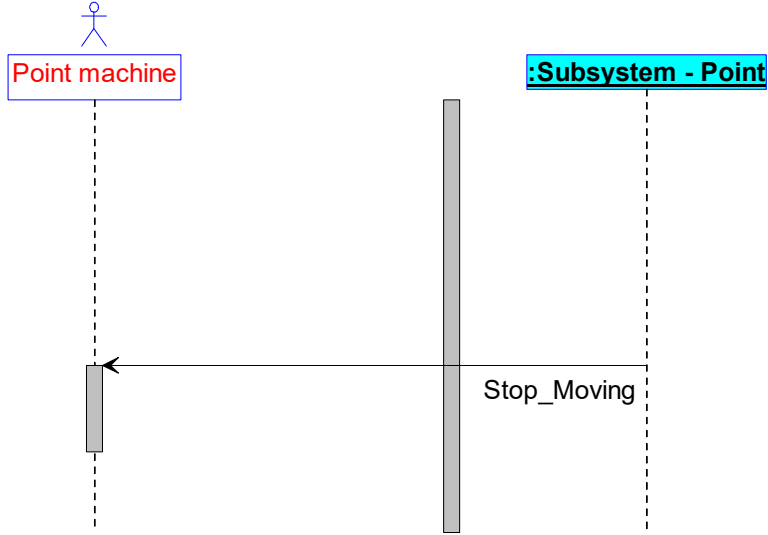
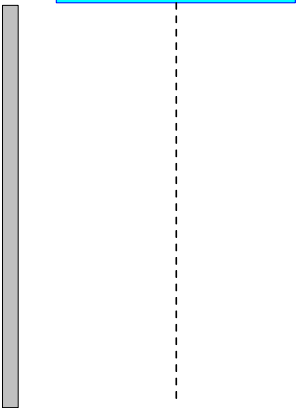
ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7383	Info	<div><p>Alternative Scenario: Handle and report non-degraded Point Position with multiple Point machines after crucial No End Position [P SD 2.2.2.2.13]</p><p>P UC2.2.2.2: Irregularities</p><p>Alternative Scenario: Handle and report non-degraded Point Position with multiple Point machines after crucial No End Position [P SD 2.2.2.2.13]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is in a Degraded point position "X" One or more non-crucial Point machine are in No end position or a Unintended position. The Subsystem - Point is configured with a 4-wire interface to the Point machine AND the last commanded position is End position "X".</p><p>Interaction 2.2.2.2.13.A</p><p>alt</p><p>1.a1 - The Subsystem - Point receives the Information that the crucial n-th Point machine is in an detected No end position.</p><p>else alt</p><p>1.b1 - The Subsystem - Point receives the Information that the crucial n-th Point machine is in an detected No end position.</p><p>end alt</p><p>2. - The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is not in a Degraded point position "X"</p><p>Postcondition:</p><p>The Subsystem - Point is in a Non degraded position.</p></div>		Basic 4-wire multiple P

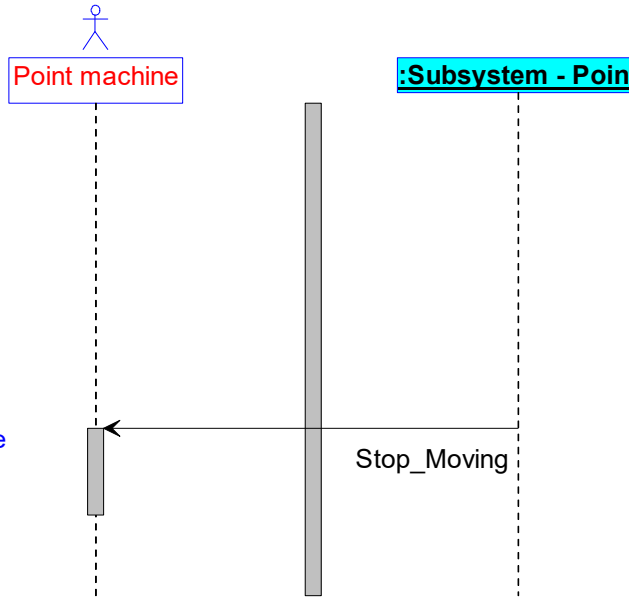
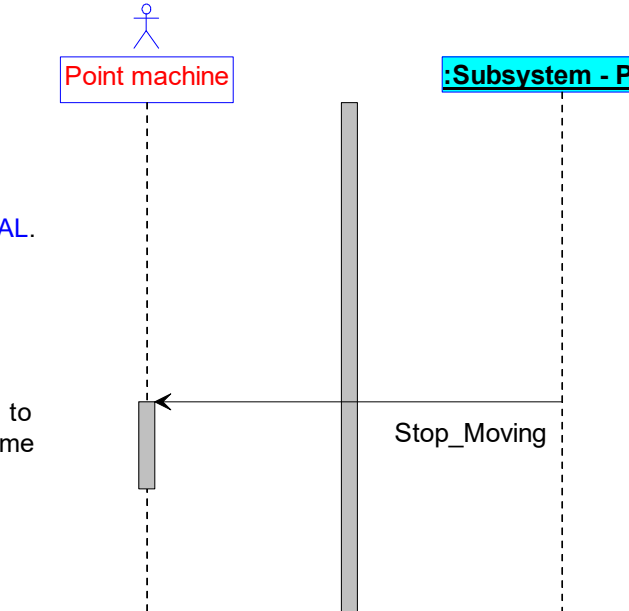
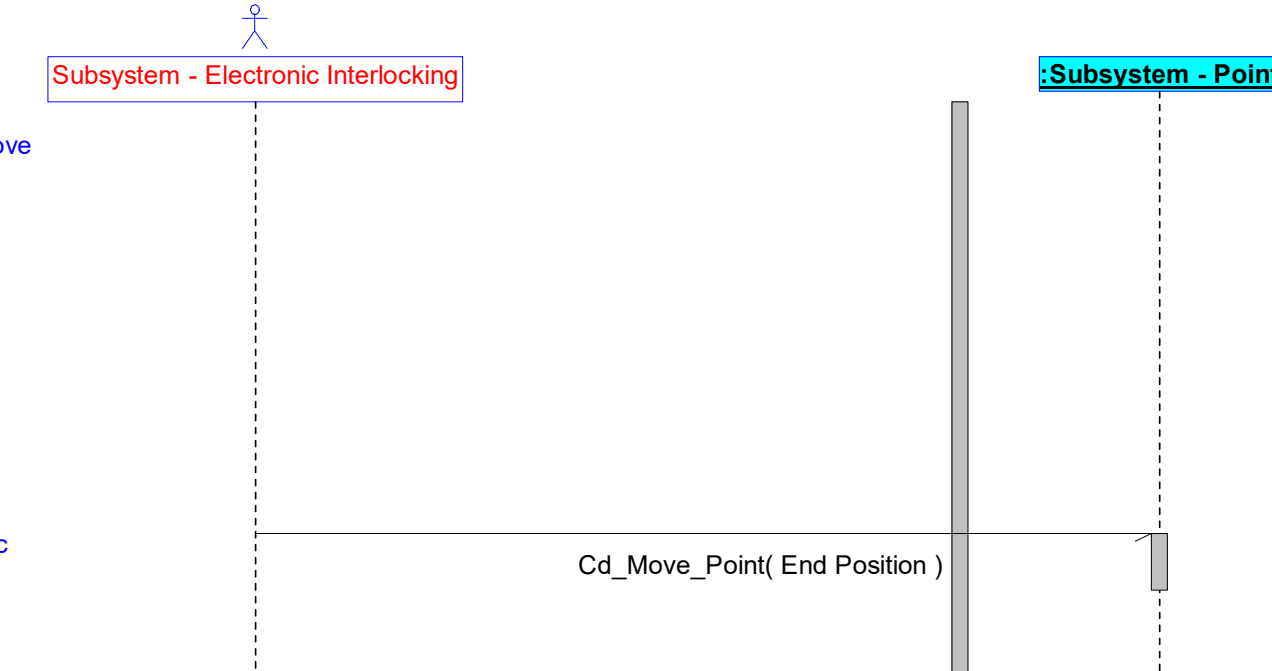
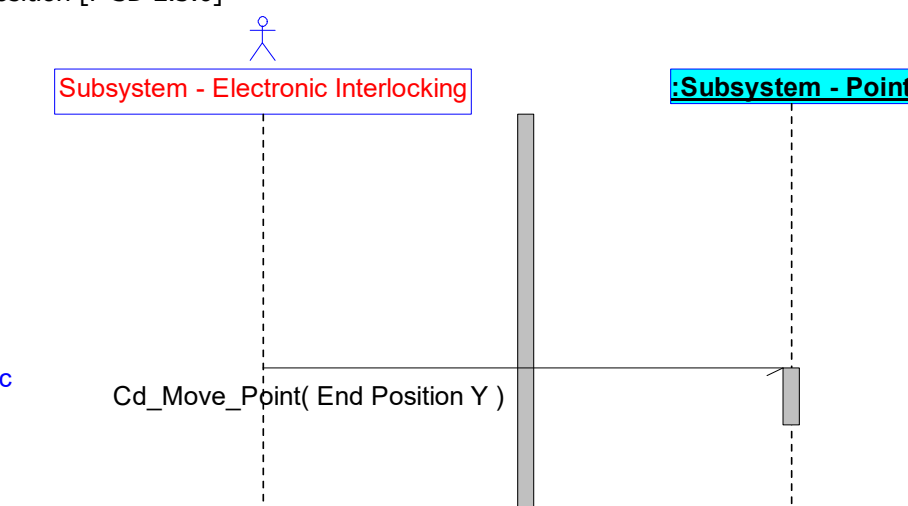
ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6765	Info	<div><p>Alternative Scenario: Handle and report Point operation timeout with multiple Point machines with position change case 1 [P SD 2.2.2.2.14]</p><p>P UC2.2.2.2: Irregularities</p><p>Alternative Scenario: Handle and report Point operation timeout with multiple Point machines with position change case 1 [P SD 2.2.2.2.14]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a 4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y", - No end position, or - a Unintended position.</p><p>Interaction 2.2.2.2.14.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X". opt [The Subsystem - Point was previously in an End position or a Unintended position]</p><p>2.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</p><p>end opt</p><p>par</p><p>3.a1 The Subsystem - Point sends the Command to the 1st Point machine to move the 1st Point machine to an End position "X".</p><p>also par</p><p>opt [The n-th Point machine has drive capability]</p><p>3.b1.a1 The Subsystem - Point sends the Command to the n-th Point machine to move the n-th Point machine to an End position "X".</p><p>end opt</p><p>also par</p><p>3.c1 The Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>end par</p><p>Interaction 2.2.2.2.14.B:</p><p>4. - The Subsystem - Point detects that the time period Con_tmax_Point_Operation has exceeded.</p><p>par</p><p>5.a1 The Subsystem - Point sends the Command to the 1st Point machine to stop moving the 1st Point machine.</p><p>also par</p><p>opt [The n-th Point machine has drive capability]</p><p>5.b1.a1 The Subsystem - Point sends the Command to the n-th Point machine to stop moving the n-th Point machine.</p><p>end opt</p><p>also par</p><p>5.c1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that a Failed Movement has occurred.</p><p>end par</p><p>par</p><p>6.a1 - The Subsystem - Point receives from the Point machine the Information that the Point machine is in a Unintended position.</p><p>also par</p><p>6.b1 - The Subsystem - Point receives from the Point machine the Information that the Point machine is in a Unintended position.</p><p>end par</p><p>7. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in Unintended position.</p><p>Postcondition:</p><p>The Subsystem - Point is in Unintended position.</p></div>	<p>Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.</p> <p>The timing related to driving individual (1 to n-th) Point Machines for a single Point is supplier specific and is specified only in a general way. The timing related to individual Point Machines is not specified by EULYNX as part of the application layer, this shall be handled by the physical implementation.</p>	Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6759	Info	<div><p>Alternative Scenario: Handle and report failed movement with multiple Point machines without position change case 1 [P SD 2.2.2.2.15]</p><p>P UC2.2.2.2: Irregularities</p><p>Alternative Scenario: Handle and report failed movement with multiple Point machines without position change case 1 [P SD 2.2.2.2.15]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a 4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y", or - an Unintended position.</p><p>Interaction 2.2.2.2.15.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X".</p><p>opt [The Subsystem - Point was previously in an End position or a Unintended position]</p><p> 2.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</p><p>end opt</p><p>par</p><p> 3.a1 The Subsystem - Point sends the Command to the 1st Point machine to move the 1st Point machine to an End position "X".</p><p>also par</p><p> opt [The n-th Point machine has drive capability]</p><p> 3.b1.a1 The Subsystem - Point sends the Command to the n-th Point machine to move the n-th Point machine to an End position "X".</p><p> end opt</p><p>also par</p><p> 3.c1 The Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>end par</p><p>Interaction 2.2.2.2.15.B:</p><p>4. - A failure occurred during the movement resulting in a Failed Movement. The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation.</p><p>par</p><p> 5.a1 The Subsystem - Point sends the Command to the 1st Point machine to stop moving the 1st Point machine.</p><p>also par</p><p> opt [The n-th Point machine has drive capability]</p><p> 5.b1.a1 The Subsystem - Point sends the Command to the n-th Point machine to stop moving the n-th Point machine.</p><p> end opt</p><p>also par</p><p> 5.c1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that a Failed Movement has occurred.</p><p>end par</p><p>opt [The Subsystem - Point was in an End position "Y".]</p><p> par</p><p> 6.a1.a1 - The Subsystem - Point receives from the 1st Point machine the pattern that represents the opposite End position "Y", which is interpreted as Information that the Point machine is in an Unintended position.</p><p> also par</p><p> 6.b1.a1 - The Subsystem - Point receives from the n-th Point machine the pattern that represents the opposite End position "Y", which is interpreted as Information that the Point machine is in an Unintended position.</p><p> end par</p><p>end opt</p><p>opt [The Subsystem - Point was in a Unintended position.]</p><p> par</p><p> 7.a1.a1 - The Subsystem - Point receives from the 1st Point machine the Information that the 1st Point machine is in a Unintended position.</p><p> also par</p><p> 7.b1.a1 - The Subsystem - Point receives from the n-th Point machine the Information that the n-th Point machine is in a Unintended position.</p><p> end par</p><p>end opt</p><p>8. The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in Unintended position.</p><p>Postcondition:</p><p>The Subsystem - Point is in Unintended position.</p></div>	<div><p>Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.</p><p>The timing related to driving individual (1 to n-th) Point Machines for a single Point is supplier specific and is specified only in a general way. The timing related to individual Point Machines is not specified by EULYNX as part of the application layer, this shall be handled by the physical implementation.</p></div>	Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6766	Info	<div><p>Alternative Scenario: Handle and report Point operation timeout with multiple Point machines without position change case 2 [P SD 2.2.2.2.16]</p><p>P UC2.2.2.2: Irregularities</p><p>Alternative Scenario: Handle and report Point operation timeout with multiple Point machines without position change case 2 [P SD 2.2.2.2.16]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a 4-wire interface to the Point machine. The Subsystem - Point is in: - an End position "Y", or - No end position, or - an Unintended position.</p><p>Interaction 2.2.2.2.16.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X". opt [The Subsystem - Point was previously in an End position or a Unintended position]</p><p>2.a1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that the Point is in No end position.</p><p>end opt</p><p>par</p><p>3.a1 The Subsystem - Point sends the Command to the 1st Point machine to move the 1st Point machine to an End position "X".</p><p>also par</p><p>opt [The n-th Point machine has drive capability]</p><p>3.b1.a1 The Subsystem - Point sends the Command to the n-th Point machine to move the n-th Point machine to an End position "X".</p><p>end opt</p><p>also par</p><p>3.c1 At this moment the Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>end par</p><p>Interaction 2.2.2.2.16.B:</p><p>4. - The Subsystem - Point detects that the time period Con_tmax_Point_Operation has exceeded.</p><p>par</p><p>5.a1 The Subsystem - Point sends the Command to the 1st Point machine to stop moving the 1st Point machine.</p><p>5.a2 The Subsystem - Point receives from the 1st Point machine the Information that the 1st Point machine is in No end position.</p><p>also par</p><p>opt [The n-th Point machine has drive capability]</p><p>5.b1.a1 The Subsystem - Point sends the Command to the n-th Point machine to stop moving the n-th Point machine.</p><p>end opt</p><p>5.b2 The Subsystem - Point receives from the n-th Point machine the Information that the n-th Point machine is in No end position.</p><p>also par</p><p>5.c1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that a Failed Movement has occurred.</p><p>end par</p><p>Postcondition:</p><p>The Subsystem - Point is in No end position.</p></div> <pre>sequenceDiagram actor SIE as Subsystem - Electronic Interlocking participant PM1 as Point machine 1st participant PMn as Point machine n-th participant SP as Subsystem - Point SIE->>SP: Cd_Move_Point(End Position X) activate SP opt SP->>SIE: Msg_Point_Position(No End Position) end par SP->>PM1: Moving activate PM1 and opt SP->>PMn: Moving activate PMn end end PM1-->>SP: deactivate PM1 PMn-->>SP: deactivate PMn SP->>SIE: Stop_Moving activate SP SP->>SP: after {Con_tmax_Point_Operation} SP->>PM1: Stop_Moving activate PM1 SP->>PMn: Stop_Moving activate PMn PM1-->>SP: Information_No_End_Position deactivate PM1 PMn-->>SP: Information_No_End_Position deactivate PMn SP->>SIE: Msg_Movement_Failed deactivate SP</pre>	<p>Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.</p> <p>The timing related to driving individual (1 to n-th) Point Machines for a single Point is supplier specific and is specified only in a general way. The timing related to individual Point Machines is not specified by EULYNX as part of the application layer, this shall be handled by the physical implementation.</p>	Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6760	Info	<div><p>Alternative Scenario: Handle and report failed movement with multiple Point machines without position change case 2 [P SD 2.2.2.2.17]</p><p>P UC2.2.2.2: Irregularities</p><p>Alternative Scenario: Handle and report failed movement with multiple Point machines without position change case 2 [P SD 2.2.2.2.17]</p><p>Precondition:</p><p>The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured with a 4-wire interface to the Point machine. The Subsystem - Point is in No end position.</p><p>Interaction 2.2.2.2.17.A:</p><p>1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "X".</p><p>par</p><p> 2.a1 The Subsystem - Point sends the Command to the 1st Point machine to move the 1st Point machine to an End position "X".</p><p>also par</p><p> opt [The n-th Point machine has drive capability]</p><p> 2.b1.a1 The Subsystem - Point sends the Command to the n-th Point machine to move the n-th Point machine to an End position "X".</p><p> end opt</p><p>also par</p><p> 2.c1 The Subsystem - Point starts to monitor the time period Con_tmax_Point_Operation.</p><p>end par</p><p>Interaction 2.2.2.2.17.B:</p><p>3. The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation.</p><p>par</p><p> 4.a1 The Subsystem - Point sends the Command to the 1st Point machine to stop moving the 1st Point machine.</p><p> 4.a2 The Subsystem - Point receives from the 1st Point machine the Information that the 1st Point machine is in No end position.</p><p>also par</p><p> opt [The n-th Point machine has drive capability]</p><p> 4.b1.a1 The Subsystem - Point sends the Command to the n-th Point machine to stop moving the n-th Point machine.</p><p> end opt</p><p> 4.b2 - The Subsystem - Point receives from the n-th Point machine the Information that the n-th Point machine is in No end position.</p><p>also par</p><p> 4.c1 The Subsystem - Point reports to the Subsystem - Electronic Interlocking that a Failed Movement has occurred.</p><p>end par</p><p>Postcondition:</p><p>The Subsystem - Point is in No end position.</p></div> <pre>sequenceDiagram participant IE as Subsystem - Electronic Interlocking participant PM1 as Point machine 1st participant PMn as Point machine n-th participant SP as :Subsystem - Point SP->>IE: Cd_Move_Point(End Position X) par SP->>PM1: Command PM1->>SP: Moving and opt [The n-th Point machine has drive capability] SP->>PMn: Command PMn->>SP: Moving end and Note over SP: {<= Con_tmax_Point_Operation} end SP->>IE: Movement_Failed par SP->>PM1: Stop_Moving PM1->>SP: Information_No_End_Position and opt [The n-th Point machine has drive capability] SP->>PMn: Stop_Moving PMn->>SP: Information_No_End_Position end and SP->>IE: Msg_Movement_Failed end</pre>	<p>Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE. The timing related to driving individual (1 to n-th) Point Machines for a single Point is supplier specific and is specified only in a general way. The timing related to individual Point Machines is not specified by EULYNX as part of the application layer, this shall be handled by the physical implementation.</p>	Basic 4-wire multiple P
Eu.P.1251	Info	P_UC2.3: Handle general Irregularities	The Subsystem-UseCase P_UC2.3: Handle general Irregularities defines the behaviour of the Subsystem - Point when an irregularity occurs.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.1252	Info	<div><p>Alternative Scenario: Perform fallback operation [P SD 2.3.1]</p><p>P UC2.3: Handle general Irregularities</p><p>Alternative Scenario: Perform fallback operation [P SD 2.3.1]</p><p>Precondition:</p><p>The <i>Subsystem - Point</i> monitors the Timevalue “Con_tmax_Point_Operation”.</p><p>Interaction 2.3.1.A:</p><p>1. - The <i>Subsystem - Point</i> enters the state <i>FALLBACK_MODE</i>.</p><p>2. The <i>Subsystem - Point</i> sends the <i>Command</i> to the <i>Point machine</i> to stop moving the <i>Point</i>. The <i>Subsystem - Point</i> stops to monitor the time period <i>Con_tmax_Point_Operation</i>.</p><p>Postcondition:</p><p>The <i>Subsystem - Point</i> is in the state <i>FALLBACK_MODE</i>.</p></div> 	Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.1299	Info	<div><p>Alternative Scenario: Handling of interrupted PDI connection [P SD 2.3.2]</p><p>P UC2.3: Handle general Irregularities</p><p>Alternative Scenario: Handling of interrupted PDI connection [P SD 2.3.2]</p><p>Precondition:</p><p>The <i>Subsystem - Point</i> is in the state <i>OPERATIONAL</i>.</p><p>Interaction 2.3.2.A:</p><p>1. - The <i>PDI</i> connection has been terminated.</p><p>Postcondition:</p><p>The <i>Subsystem - Point</i> is in the state <i>INITIALISING</i>. The <i>Process Data Interface protocol</i> connection is terminated.</p></div> 	<p>The following functionality remains available within the state INITIALISING after the termination of the PDI connection:</p> <ul style="list-style-type: none">• Redrive Point operation (for example as in Main Success Scenario: Redrive Point after lost end position Non 4W [P SD 2.1.1.2.1], Alternative Scenario: Handle Point operation timeout during Redrive Non 4W [P SD 2.1.1.3.9], Main Success Scenario: Redrive Point after lost end position [P SD 2.1.2.2.1], Alternative Scenario: Handle and report non-degraded Point Position with multiple Point machines after crutial Unintended Position [P SD 2.1.2.4.11])• Stop ongoing movement when Ability to move point is lost (for example as in Alternative Scenario: Handle and report loss of ability to move point with single point machine [P SD 2.1.1.3.7], Alternative Scenario: Handle and report non-degraded Point Position with multiple Point machines [P SD 2.1.2.4.9], Alternative Scenario: Handle and report Degraded Point Position with multiple Point machines after crucial End Position [P SD 2.1.2.4.10], Alternative Scenario: Handle and report loss of ability to move point with single point machine [P SD 2.2.1.2.5], Alternative Scenario: Handle and report loss of ability to move point with multiple Point machines [P SD 2.2.2.5])• Restoration of Ability to move point (for example as in Alternative Scenario: Handle and report restoring of Ability to move point with single point machine [P SD 2.1.1.3.8], Alternative Scenario: Handle and report restoring of Ability to move point with single point machine [P SD 2.2.1.2.6])• All functionality related to the maintainer interface P1 (for example as in P_UC3.1: Display status of Subsystem - Point locally) <p>Note: If the PDI Connection is terminated during ongoing point movement, the movement is continued as normal.</p>	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.3093	Info	<div>Alternative Scenario: Supply voltage of the Subsystem has gone outside of the required range for operation [P SD 2.3.3]</div> <div>P UC2.3: Handle general Irregularities</div> <div>Alternative Scenario: Supply voltage of the Subsystem has gone outside of the required range for operation [P SD 2.3.3]</div> <div>Precondition: The Subsystem - Point monitors the Timevalue "Con_tmax_Point_Operation".</div> <div>Interaction 2.3.3.A: 1. - The Subsystem - Point enters the state NO_OPERATING_VOLTAGE. 2. The Subsystem - Point sends the Command to the Point machine to stop moving the Point. The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation. Postcondition: The Subsystem - Point is in the state NO_OPERATING_VOLTAGE.</div> 	Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.3104	Info	<div>Alternative Scenario: Reset occurs [P SD 2.3.4]</div> <div>P UC2.3: Handle general Irregularities</div> <div>Alternative Scenario: Reset occurs [P SD 2.3.4]</div> <div>Precondition: The Subsystem - Point is in the state INITIALISING or OPERATIONAL. The Subsystem - Point monitors the Timevalue "Con_tmax_Point_Operation".</div> <div>Interaction 2.3.4.A: 1. - A reset has occurred. 2. The Subsystem - Point sends the Command to the Point machine to stop moving the Point. The Subsystem - Point stops to monitor the time period Con_tmax_Point_Operation. 3. The Subsystem - Point enters the state BOOTING. Postcondition: The Subsystem - Point is in the state BOOTING.</div> 	Stop_Moving is functionally realised by setting the Moving commands for left and right to FALSE.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5371	Info	<div>Alternative Scenario: Handle received move command if ability to move point is lost [P SD 2.3.5]</div> <div>P UC2.3: Handle general Irregularities</div> <div>Alternative Scenario: Handle received move command if ability to move point is lost [P SD 2.3.5]</div> <div>Precondition: The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is configured to observe the Ability to move point. The Subsystem - Point is Unable to move point. The Subsystem - Point is in: - an End position, or - No end position, or - an Unintended position.</div> <div>Interaction 2.3.5.A: 1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position. Postcondition: ---</div> 		Option Able to move
Eu.P.1469	Info	<div>Alternative Scenario: Commanding of the Point to the current End position [P SD 2.3.6]</div> <div>P UC2.3: Handle general Irregularities</div> <div>Alternative Scenario: Commanding of the Point to the current End position [P SD 2.3.6]</div> <div>Precondition: The Subsystem - Point is in the state OPERATIONAL. The Subsystem - Point is in an End position "Y".</div> <div>Interaction 2.3.6.A: 1. - The Subsystem - Point receives from the Subsystem - Electronic Interlocking the Command to move the Point to an End position "Y". Postcondition: ---</div> 		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.1127	Info	<div><div>[Package] Subsystem - Point - Functional Context [Functional Viewpoint - Subsystem Definition - Maintenance]</div><div>uc [Package] Subsystem - Point - Functional Context [Functional Viewpoint - Subsystem Definition - Maintenance]</div><div><div>Subsystem - Point</div><div><div>P_UC3.1: Display status of Subsystem - Point locally</div><div>P_UC3.2: Collect and provide event driven diagnostic data</div><div>P_UC3.3: Collect and provide preventive diagnostic data</div><div>P_UC3.4: Updating specific software</div></div><div><div>Maintainer</div><div>Subsystem - Maintenance and Data Management</div></div></div></div>		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.1124	Info	P_UC3.1: Display status of Subsystem - Point locally	Information: The Subsystem-UseCase P_UC3.1: Display status of Subsystem - Point locally defines the local display of the EULYNX field element Subsystem. See ID Eu.P.890.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.1125	Info	P_UC3.2: Collect and provide event driven diagnostic data	Information: The Subsystem-UseCase P_UC3.2: Collect and provide event driven diagnostic data defines the event driven collection and provision of diagnostic data in case of irregularities. See ID Eu.P.925.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.1126	Info	P_UC3.3: Collect and provide preventive diagnostic data	Information: The Subsystem-UseCase P_UC3.3: Collect and provide preventive diagnostic data defines the continuous collection and provision of diagnostic data for preventive maintenance. See ID Eu.P.925.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.1468	Info	P_UC3.4: Updating specific software	Information: The Subsystem-UseCase P_UC3.4: Updating specific software defines the process of updating the specific software between Subsystem - Maintenance and Data Management and the Subsystem.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6272	Head	3.3.3 Subsystem - Point - Functional Partitioning		

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.3297	Def	<div><div>[Package] Subsystem - Point - Functional Partitioning [Functional Viewpoint - Subsystem Requirements]</div><div><div>bdd [Package] Subsystem - Point - Functional Partitioning [Functional Viewpoint - Subsystem Requirements]</div><div><div><div>Subsystem - Point - Functional Architecture</div><div><div>SCI-P - Functional Viewpoint</div><div><div>«functional entity» F_SCI_P_Report</div><div>1</div></div><div><div>«functional entity» F_SCI_P_Receive</div><div>1</div></div></div><div><div>Generic requirements for subsystems</div><div><div>«functional entity» F_SCI_EfeS_Sec</div><div>1</div></div><div><div>«functional entity» F_EST_EfeS</div><div>1</div></div></div><div><div>Subsystem - Point - Functional Entities</div><div><div>«functional entity» F_Control_Non4W_PM</div><div>1..*</div></div><div><div>«functional entity» F_Observe_Overall_Point_Position</div><div>1</div></div><div><div>«functional entity» F_Observe_Degraded_Point_Position</div><div>1</div></div><div><div>«functional entity» F_Observe_Ability_To_Move</div><div>1</div></div><div><div>«functional entity» F_Control_Point</div><div>1</div></div><div><div>«functional entity» F_Observe_Movement_Failed</div><div>1</div></div><div><div>«functional entity» F_Control_And_Observe_4W_PM</div><div>1..*</div></div></div></div></div></div><div><div></div></div></div>		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5781	Head	3.3.4 Subsystem - Point - Functional Architecture		
Eu.P.939	Info	Subsystem - Point	The Subsystem - Point integrates the moveable elements, that may be moved to a different position by a request from the Subsystem - Electronic Interlocking.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.942	Def	P3	The functional control interface to Point machines for the information flow through the interface, which is defined by the FlowSpecification "Point_machine".	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6378	Def	P1	The functional Maintenance/Operation/Display interface to the Maintainer. The InformationFlow through the interface is defined by "Maintainer".	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6379	Def	P4	The functional System Data interface to the Basic Data identifier. The InformationFlow through the interface is defined by "Basic_Data_Identifier".	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6380	Def	SCI-P	The functional Process Data interface to the Subsystem - Electronic Interlocking (SCI: Standard Communication Interface). The InformationFlow through the interface is further defined in SCI_P_Subsystem_EIL.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6381	Def	SDI-P	The functional Diagnostic interface to the Subsystem - Maintenance and Data Management for the InformationFlow through the interface, which is defined by "Subsystem_MDM_D".	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6382	Def	SMI-P	The functional Maintenance Interface to the Subsystem - Maintenance and Data Management for the InformationFlow through the interface, which is defined by "Subsystem_MDM_M".	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6383	Def	SSI-P	The Security Service Interface to the Subsystem - Security Services Platform. The InformationFlow through the interface is further defined in SSI-P (Subsystem - Security Services Platform).	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5809	Head	3.3.5 Subsystem - Point - Functional Entities		
Eu.P.6948	Info	F_Control_Point		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6949	Req	<div>[Block] F_Control_Point [Functional Viewpoint - Subsystem Requirements - Functional Entity]</div> <div><div><div><div>«functional entity» F_Control_Point</div><div>values</div><div>«BlockProperty» Mem_Last_Required_Point_Position : String</div></div><div><div>d10in_Required_Point_Position : String</div><div>d12out_Required_PM_Position : String</div></div><div><div>d11in_Observed_Ability_To_Move : String</div><div>d13in_Observed_Movement_Failed : Boolean</div><div>d14in_Observed_Point_Position : String</div><div>D18in_Con_Use_Redrive : Boolean</div><div>d51in_EST_EfeS_State : String</div></div></div></div>		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6950	Def	d10in_Required_Point_Position		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6951	Def	d11in_Observed_Ability_To_Move		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6952	Def	d13in_Observed_Movement_Failed		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6953	Def	d14in_Observed_Point_Position	The port d14in_Observed_Point_Position provides the Point machine position to the Subsystem - Point. The port d14in_Observed_Point_Position refines the InformationFlow Information_No_End_Position, Information_End_Position_Reached, Information_End_Position_Detected and Information_Unintended_Position.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6954	Def	D18in_Con_Use_Redrive	The port D18in_Con_Use_Redrive provides a configuration value to the Point for Redrive point functionality. true: Point has Redrive point functionality false: Point has no Redrive point functionality	Option Redrive
Eu.P.6956	Def	d12out_Required_PM_Position		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6957	Def	d51in_EST_EfeS_State		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6958	Info	F_Control_Point - Behaviour		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6959	Req	<div>Functional Viewpoint - Subsystem Requirements - Functional Entity STD 1</div> <div>stm [State Machine] F_Control_Point - Behaviour [Functional Viewpoint - Subsystem Requirements - Functional Entity STD 1]</div> <pre>stateDiagram-v2 [*] --> Initial0 Initial0 --> OPERATING : when(d51in_EST_EfeS_State = "NO_OPERATING_VOLTAGE" OR d51in_EST_EfeS_State = "BOOTING" OR d51in_EST_EfeS_State = "FALLBACK_MODE") / state OPERATING { Entry / d12out_Required_PM_Position := "UNCOMMANDED"; Mem_Last_Required_Point_Position := "UNCOMMANDED"; } OPERATING --> WAITING : when(d51in_EST_EfeS_State = "INITIALISING") / state WAITING { Initial1 } WAITING --> STOPPED : when(d11in_Observed_Ability_To_Move = "UNABLE_TO_MOVE") / state STOPPED { Entry / d12out_Required_PM_Position := "UNCOMMANDED"; } STOPPED --> WAITING : when(d14in_Observed_Point_Position <> Mem_Last_Required_Point_Position) [Mem_Last_Required_Point_Position = "LEFT" AND D18in_Con_Use_Redrive AND d11in_Observed_Ability_To_Move = "ABLE_TO_MOVE"] / STOPPED --> STOPPED : when(d11in_Observed_Ability_To_Move = "UNABLE_TO_MOVE") / STOPPED --> MOVING_LEFT : when(d13in_Observed_Movement_Failed) / STOPPED --> MOVING_RIGHT : when(d13in_Observed_Movement_Failed) / MOVING_LEFT --> STOPPED : when(d14in_Observed_Point_Position = Mem_Last_Required_Point_Position) / MOVING_LEFT --> STOPPED : when(d10in_Required_Point_Position = "UNCOMMANDED") / MOVING_LEFT --> STOPPED : when(d14in_Observed_Point_Position = d10in_Required_Point_Position) / Mem_Last_Required_Point_Position := d14in_Observed_Point_Position; MOVING_RIGHT --> STOPPED : when(d14in_Observed_Point_Position = Mem_Last_Required_Point_Position) / MOVING_RIGHT --> STOPPED : when(d10in_Required_Point_Position = "UNCOMMANDED") / MOVING_RIGHT --> STOPPED : when(d14in_Observed_Point_Position = d10in_Required_Point_Position) / Mem_Last_Required_Point_Position := d14in_Observed_Point_Position; MOVING_LEFT --> MOVING_LEFT : when(d10in_Required_Point_Position = "LEFT") [d10in_Required_Point_Position <> d14in_Observed_Point_Position AND d11in_Observed_Ability_To_Move = "ABLE_TO_MOVE"] / Mem_Last_Required_Point_Position := "LEFT"; MOVING_RIGHT --> MOVING_RIGHT : when(d10in_Required_Point_Position = "RIGHT") [d10in_Required_Point_Position <> d14in_Observed_Point_Position AND d11in_Observed_Ability_To_Move = "ABLE_TO_MOVE"] / Mem_Last_Required_Point_Position := "RIGHT"; MOVING_LEFT --> STOPPED : when(d10in_Required_Point_Position = "RIGHT" AND (d10in_Required_Point_Position <> d14in_Observed_Point_Position)) [d11in_Observed_Ability_To_Move = "ABLE_TO_MOVE"] / MOVING_RIGHT --> STOPPED : when(d10in_Required_Point_Position = "LEFT" AND (d10in_Required_Point_Position <> d14in_Observed_Point_Position)) [d11in_Observed_Ability_To_Move = "ABLE_TO_MOVE"] /</pre>	This state machine diagram describes the requirements for the following functionalities: - observes and forward the required PM position	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6960	Def	Initial0		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6961	Def	/{Initial0 - OPERATING}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6962	Def	OPERATING		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6963	Def	Initial1		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6964	Def	/{Initial1 - WAITING}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6965	Def	MOVING_LEFT		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6966	Def	entry/d12out_Required_PM_Position := "LEFT";{State-internal in MOVING_LEFT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6967	Def	when(d10in_Required_Point_Position = "RIGHT" AND (d10in_Required_Point_Position <> d14in_Observed_Point_Position))[d11in_Observed_Ability_To_Move = "ABLE_TO_MOVE"]/{MOVING_LEFT - MOVING_RIGHT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6968	Def	when(d10in_Required_Point_Position = "UNCOMMANDED")/{MOVING_LEFT - STOPPED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6969	Def	when(d11in_Observed_Ability_To_Move = "UNABLE_TO_MOVE")/{MOVING_LEFT - STOPPED}		Option Able to move
Eu.P.6970	Def	when(d13in_Observed_Movement_Failed)/{MOVING_LEFT - STOPPED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6972	Def	when(d14in_Observed_Point_Position = d10in_Required_Point_Position)/Mem_Last_Required_Point_Position := d14in_Observed_Point_Position;{MOVING_LEFT - STOPPED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6973	Def	when(d14in_Observed_Point_Position = Mem_Last_Required_Point_Position)/{MOVING_LEFT - STOPPED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6974	Def	MOVING_RIGHT		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6975	Def	entry/d12out_Required_PM_Position := "RIGHT";{State-internal in MOVING_RIGHT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6976	Def	when(d10in_Required_Point_Position = "LEFT" AND (d10in_Required_Point_Position <> d14in_Observed_Point_Position))[d11in_Observed_Ability_To_Move = "ABLE_TO_MOVE"]/{MOVING_RIGHT - MOVING_LEFT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6977	Def	when(d10in_Required_Point_Position = "UNCOMMANDED")/{MOVING_RIGHT - STOPPED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6978	Def	when(d11in_Observed_Ability_To_Move = "UNABLE_TO_MOVE")/{MOVING_RIGHT - STOPPED}		Option Able to move
Eu.P.6979	Def	when(d13in_Observed_Movement_Failed)/{MOVING_RIGHT - STOPPED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6981	Def	when(d14in_Observed_Point_Position = d10in_Required_Point_Position)/Mem_Last_Required_Point_Position := d14in_Observed_Point_Position;{MOVING_RIGHT - STOPPED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6982	Def	when(d14in_Observed_Point_Position = Mem_Last_Required_Point_Position)/{MOVING_RIGHT - STOPPED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6983	Def	STOPPED		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6984	Def	entry/d12out_Required_PM_Position := "UNCOMMANDED";{State-internal in STOPPED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6985	Def	when(d10in_Required_Point_Position = "LEFT")[d10in_Required_Point_Position <> d14in_Observed_Point_Position AND d11in_Observed_Ability_To_Move = "ABLE_TO_MOVE"]/Mem_Last_Required_Point_Position := "LEFT";{STOPPED - MOVING_LEFT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6986	Def	when(d10in_Required_Point_Position = "RIGHT")[d10in_Required_Point_Position <> d14in_Observed_Point_Position AND d11in_Observed_Ability_To_Move = "ABLE_TO_MOVE"]/Mem_Last_Required_Point_Position := "RIGHT";{STOPPED - MOVING_RIGHT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6980	Def	when(d14in_Observed_Point_Position <> Mem_Last_Required_Point_Position)[Mem_Last_Required_Point_Position = "RIGHT" AND D18in_Con_Use_Redrive AND d11in_Observed_Ability_To_Move = "ABLE_TO_MOVE"]/ {STOPPED - MOVING_RIGHT}		Option Redrive
Eu.P.6971	Def	when(d14in_Observed_Point_Position <> Mem_Last_Required_Point_Position)[Mem_Last_Required_Point_Position = "LEFT" AND D18in_Con_Use_Redrive AND d11in_Observed_Ability_To_Move = "ABLE_TO_MOVE"]/ {STOPPED - MOVING_LEFT}		Option Redrive
Eu.P.6987	Def	entry/d12out_Required_PM_Position := "UNCOMMANDED"; Mem_Last_Required_Point_Position := "UNCOMMANDED";{State-internal in OPERATING}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6988	Def	WAITING		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6989	Def	when(d51in_EST_EfeS_State = "INITIALISING")/{WAITING - STOPPED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6990	Def	when(d51in_EST_EfeS_State = "NO_OPERATING_VOLTAGE" OR d51in_EST_EfeS_State = "BOOTING" OR d51in_EST_EfeS_State = "FALLBACK_MODE")/{OPERATING - OPERATING}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4548	Info	F_Observe_Overall_Point_Position		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.4549	Req	<div>[Block] F_Observe_Overall_Point_Position [Functional Viewpoint - Subsystem Requirements - Functional Entity]</div> <div><div>ibdd [Block] F_Observe_Overall_Point_Position [Functional Viewpoint - Subsystem Requirements - Functional Entity]</div><div><div><div>«functional entity»</div><div>F_Observe_Overall_Point_Position</div><div>Operation</div><div>«Operation» cOp1_Mem_PM_Position ()</div><div>values</div><div>«BlockProperty» Mem_PM1_Crucial_Position : String</div><div>«BlockProperty» Mem_PM1_Non_Crucial_Position : String</div><div>«BlockProperty» Mem_PMi_Crucial_Position : String</div><div>«BlockProperty» Mem_PMk_Non_Crucial_Position : String</div></div><div><div><div><div>→</div>D10in_PM1_Crucial_Position : String</div><div>d14out_Observed_Point_Position : String</div><div><div>→</div></div></div><div><div><div>→</div>D10in_PMi_Crucial_Position : String</div><div><div>→</div>D10in_PM1_Non_Crucial_Position : String</div><div><div>→</div>D10in_PMk_Non_Crucial_Position : String</div><div><div>→</div>D26in_Con_PMi_Crucial_Activation : Boolean</div><div><div>→</div>D27in_Con_PM1_Non_Crucial_Activation : Boolean</div><div><div>→</div>D28in_Con_PMk_Non_Crucial_Activation : Boolean</div><div><div>→</div>D29in_Con_Use_Unintended_Position : Boolean</div><div><div>→</div>d51in_EST_EfeS_State : String</div></div></div></div></div>		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6309	Def	<div>/* cOp1_Mem_PM_Position */</div> <div>if D26in_Con_PMi_Crucial_Activation then</div> <div> Mem_PMi_Crucial_Position := D10in_PMi_Crucial_Position;</div> <div>end if</div> <div>if D27in_Con_PM1_Non_Crucial_Activation then</div> <div> Mem_PM1_Non_Crucial_Position := D10in_PM1_Non_Crucial_Position;</div> <div>end if</div> <div>if D28in_Con_PMk_Non_Crucial_Activation then</div> <div> Mem_PMk_Non_Crucial_Position := D10in_PMk_Non_Crucial_Position;</div> <div>end if</div>	cOp1_Mem_PM_Position	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4566	Def	d51in_EST_EfeS_State		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4567	Def	D10in_PM1_Crucial_Position	The port D10in_PM1_Crucial_Position represents the Point position that was observed at the interface of the first crucial Point machine. The Point position is derived from the InformationFlow Information_No_End_Position, Information_End_Position_Reached, Information_End_Position_Detected and Information_Unintended_Position.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4568	Def	D10in_PMi_Crucial_Position	The port D10in_PMi_Crucial_Position represents the Point position that was observed at the interface of the n-th crucial Point machine. The Point position is derived from the InformationFlow Information_No_End_Position and Information_Unintended_Position.	Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.4586	Def	d14out_Observed_Point_Position		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4979	Def	D26in_Con_PMi_Crucial_Activation	The port D26in_Con_PMi_Crucial_Activation provides configuration values for the Point machine PM1n crucial and representing a n-th crucial Point machine in configurations with more than one Point machine. true: Point machine PM1n Crucial is activated false: Point machine PM1n Crucial is not activated	Basic non-4-wire multiple P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6033	Def	D10in_PM1_Non_Crucial_Position	The port D10in_PM1_Non_Crucial_Position represents the Point position that was observed at the interface of the first non-crucial Point machine. The Point position is derived from the InformationFlowInformation_No_End_Position , and Information_Unintended_Position.	Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6034	Def	D10in_PMk_Non_Crucial_Position	The port D10in_PMk_Non_Crucial_Position represents the Point position that was observed at the interface of the n-th non-crucial Point machine. The Point position is derived from the InformationFlowInformation_No_End_Position and Information_Unintended_Position.	Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6035	Def	D27in_Con_PM1_Non_Crucial_Activation	<p>The port D27in_Con_PM1_Non_Crucial_Activation provides configuration values for the Point machine PM2 non crucial, representing the 1st non crucial Point machine in configurations with more than one Point machine.</p> <p>true: Point machine PM2 Non Crucial is activated false: Point machine PM2 Non Crucial is not activated</p>	Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6036	Def	D28in_Con_PMk_Non_Crucial_Activation	<p>The port D28in_Con_PMk_Non_Crucial_Activation provides configuration values for the Point machine PM2n non crucial, representing a n-th non crucial Point machine in configurations with more than one Point machine.</p> <p>true: Point machine PM2n Non Crucial is activated false: Point machine PM2n Non Crucial is not activated</p>	Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6037	Def	D29in_Con_Use_Unintended_Position	<p>The port D29in_Con_Use_Unintended_Position provides a configuration value to the Subsystem - Point enabling interpretation of Unintended position.</p> <p>true: Point is configured for interpretation of Unintended position false: Point is not configured for interpretation of Unintended position</p>	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4587	Info	F_Observe_Overall_Point_Position - Behaviour		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.4588	Req	<div>Functional Viewpoint - Subsystem Requirements - Functional Entity STD 2</div> <div>stm [State Machine] F_Observe_Overall_Point_Position - Behaviour [Functional Viewpoint - Subsystem Requirements - Functional Entity STD 2]</div> <div><pre>stateDiagram-v2 [*] --> Initial0 Initial0 --> OBSERVE_OVERALL_POINT_POSITION state OBSERVE_OVERALL_POINT_POSITION { state NO_END_POSITION { entry / Mem_PM1_Crucial_Position := D10in_PM1_Crucial_Position; cOp1_Mem_PM_Position (); } state LEFT { entry / Mem_PM1_Crucial_Position := D10in_PM1_Crucial_Position; cOp1_Mem_PM_Position (); } state UNINTENDED_POSITION { entry / Mem_PM1_Crucial_Position := D10in_PM1_Crucial_Position; cOp1_Mem_PM_Position (); } state RIGHT { entry / Mem_PM1_Crucial_Position := D10in_PM1_Crucial_Position; cOp1_Mem_PM_Position (); } NO_END_POSITION --> Junction : [else] / d14out_Observed_Point_Position := "NO_END_POSITION"; Junction --> NO_END_POSITION : [(D10in_PM1_Crucial_Position = "UNINTENDED_POSITION" OR D10in_PMi_Crucial_Position = "UNINTENDED_POSITION" OR D10in_PM1_Non_Crucial_Position = "UNINTENDED_POSITION" OR D10in_PMc_Non_Crucial_Position = "UNINTENDED_POSITION") AND D29in_Con_Use_Unintended_Position] / d14out_Observed_Point_Position := "UNINTENDED_POSITION"; Junction --> LEFT : [D10in_PM1_Crucial_Position = "LEFT" AND (D10in_PMi_Crucial_Position = "LEFT" OR NOT D26in_Con_PMi_Crucial_Activation) AND (D10in_PM1_Non_Crucial_Position = "LEFT" OR NOT D27in_Con_PM1_Non_Crucial_Activation) AND (D10in_PMc_Non_Crucial_Position = "LEFT" OR NOT D28in_Con_PMc_Non_Crucial_Activation)] / d14out_Observed_Point_Position := "LEFT"; Junction --> UNINTENDED_POSITION : [D10in_PM1_Crucial_Position = "RIGHT" AND (D10in_PMi_Crucial_Position = "RIGHT" OR NOT D26in_Con_PMi_Crucial_Activation) AND (D10in_PM1_Non_Crucial_Position = "RIGHT" OR NOT D27in_Con_PM1_Non_Crucial_Activation) AND (D10in_PMc_Non_Crucial_Position = "RIGHT" OR NOT D28in_Con_PMc_Non_Crucial_Activation)] / d14out_Observed_Point_Position := "RIGHT"; Junction --> RIGHT : [D10in_PM1_Crucial_Position = "RIGHT" AND (D10in_PMi_Crucial_Position = "RIGHT" OR NOT D26in_Con_PMi_Crucial_Activation) AND (D10in_PM1_Non_Crucial_Position = "RIGHT" OR NOT D27in_Con_PM1_Non_Crucial_Activation) AND (D10in_PMc_Non_Crucial_Position = "RIGHT" OR NOT D28in_Con_PMc_Non_Crucial_Activation)] / d14out_Observed_Point_Position := "RIGHT"; NO_END_POSITION --> NO_END_POSITION : when(D10in_PM1_Crucial_Position <> Mem_PM1_Crucial_Position) / LEFT --> LEFT : when(D10in_PM1_Non_Crucial_Position <> Mem_PM1_Non_Crucial_Position) / UNINTENDED_POSITION --> UNINTENDED_POSITION : when(D10in_PMc_Non_Crucial_Position <> Mem_PMc_Non_Crucial_Position) / RIGHT --> RIGHT : when(D10in_PMi_Crucial_Position <> Mem_PMi_Crucial_Position) / } state Initial1 { [*] } Initial1 --> Junction</pre></div>	<div>This state machine diagram describes the requirements for the following functionalities:</div> <div>- observes and forward the overall point position state</div>	<div>Basic non-4-wire single P</div> <div>Basic non-4-wire multiple P</div> <div>Basic 4-wire single P</div> <div>Basic 4-wire multiple P</div>
Eu.P.6241	Def	Initial0		<div>Basic non-4-wire single P</div> <div>Basic non-4-wire multiple P</div> <div>Basic 4-wire single P</div> <div>Basic 4-wire multiple P</div>
Eu.P.6242	Def	/Mem_PM1_Crucial_Position := "UNDEFINED"; Mem_PMi_Crucial_Position := "UNDEFINED"; Mem_PM1_Non_Crucial_Position := "UNDEFINED"; Mem_PMc_Non_Crucial_Position := "UNDEFINED";{Initial0 - OBSERVE_OVERALL_POINT_POSITION}		<div>Basic non-4-wire single P</div> <div>Basic non-4-wire multiple P</div> <div>Basic 4-wire single P</div> <div>Basic 4-wire multiple P</div>
Eu.P.6310	Def	OBSERVE_OVERALL_POINT_POSITION		<div>Basic non-4-wire single P</div> <div>Basic non-4-wire multiple P</div> <div>Basic 4-wire single P</div> <div>Basic 4-wire multiple P</div>
Eu.P.6311	Def	Initial1		<div>Basic non-4-wire single P</div> <div>Basic non-4-wire multiple P</div> <div>Basic 4-wire single P</div> <div>Basic 4-wire multiple P</div>
Eu.P.6312	Def	/ {Initial1 - Junction}		<div>Basic non-4-wire single P</div> <div>Basic non-4-wire multiple P</div> <div>Basic 4-wire single P</div> <div>Basic 4-wire multiple P</div>
Eu.P.6313	Def	Junction		<div>Basic non-4-wire single P</div> <div>Basic non-4-wire multiple P</div> <div>Basic 4-wire single P</div> <div>Basic 4-wire multiple P</div>

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6314	Def	[D10in_PM1_Crucial_Position = "LEFT" AND (D10in_PMi_Crucial_Position = "LEFT" OR NOT D26in_Con_PMi_Crucial_Activation) AND (D10in_PM1_Non_Crucial_Position = "LEFT" OR NOT D27in_Con_PM1_Non_Crucial_Activation) AND (D10in_PMk_Non_Crucial_Position = "LEFT" OR NOT D28in_Con_PMk_Non_Crucial_Activation)]/d14out_Observed_Point_Position := "LEFT";{Junction - LEFT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6315	Def	[else]/ d14out_Observed_Point_Position := "NO_END_POSITION";{Junction - NO_END_POSITION}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6316	Def	[D10in_PM1_Crucial_Position = "RIGHT" AND (D10in_PMi_Crucial_Position = "RIGHT" OR NOT D26in_Con_PMi_Crucial_Activation) AND (D10in_PM1_Non_Crucial_Position = "RIGHT" OR NOT D27in_Con_PM1_Non_Crucial_Activation) AND (D10in_PMk_Non_Crucial_Position = "RIGHT" OR NOT D28in_Con_PMk_Non_Crucial_Activation)]/d14out_Observed_Point_Position := "RIGHT";{Junction - RIGHT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6317	Def	[(D10in_PM1_Crucial_Position = "UNINTENDED_POSITION" OR D10in_PMi_Crucial_Position = "UNINTENDED_POSITION" OR D10in_PM1_Non_Crucial_Position = "UNINTENDED_POSITION" OR D10in_PMk_Non_Crucial_Position = "UNINTENDED_POSITION") AND D29in_Con_Use_Unintended_Position]/d14out_Observed_Point_Position := "UNINTENDED_POSITION";{Junction - UNINTENDED_POSITION}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6318	Def	LEFT		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6408	Def	entry/ Mem_PM1_Crucial_Position := D10in_PM1_Crucial_Position; cOp1_Mem_PM_Position();{State-internal in LEFT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6319	Def	NO_END_POSITION		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6409	Def	entry/ Mem_PM1_Crucial_Position := D10in_PM1_Crucial_Position; cOp1_Mem_PM_Position();{State-internal in NO_END_POSITION}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6320	Def	RIGHT		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6410	Def	entry/ Mem_PM1_Crucial_Position := D10in_PM1_Crucial_Position; cOp1_Mem_PM_Position();{State-internal in RIGHT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6321	Def	UNINTENDED_POSITION		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6411	Def	entry/ Mem_PM1_Crucial_Position := D10in_PM1_Crucial_Position; cOp1_Mem_PM_Position();{State-internal in UNINTENDED_POSITION}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6322	Def	when(D10in_PM1_Crucial_Position <> Mem_PM1_Crucial_Position)/{OBSERVE_OVERALL_POINT_POSITION - OBSERVE_OVERALL_POINT_POSITION}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6323	Def	when(D10in_PMi_Crucial_Position <> Mem_PMi_Crucial_Position)/{OBSERVE_OVERALL_POINT_POSITION - OBSERVE_OVERALL_POINT_POSITION}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6324	Def	when(D10in_PM1_Non_Crucial_Position <> Mem_PM1_Non_Crucial_Position)/{OBSERVE_OVERALL_POINT_POSITION - OBSERVE_OVERALL_POINT_POSITION}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6325	Def	when(D10in_PMk_Non_Crucial_Position <> Mem_PMk_Non_Crucial_Position)/{OBSERVE_OVERALL_POINT_POSITION - OBSERVE_OVERALL_POINT_POSITION}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6326	Def	when(d51in_EST_EfeS_State = "NO_OPERATING_VOLTAGE" OR d51in_EST_EfeS_State = "BOOTING" OR d51in_EST_EfeS_State = "FALLBACK_MODE")/{OBSERVE_OVERALL_POINT_POSITION - OBSERVE_OVERALL_POINT_POSITION}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6002	Info	F_Observe_Degraded_Point_Position		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6003	Req	<div>[Block] F_Observe_Degraded_Point_Position [Functional Viewpoint - Subsystem Requirements - Functional Entity]</div> <div><div>ibd [Block] F_Observe_Degraded_Point_Position [Functional Viewpoint - Subsystem Requirements - Functional Entity]</div><div><div>«functional entity» F_Observe_Degraded_Point_Position</div><div><div><div>d14in_Observed_Point_Position : String</div><div><div>D10in_PM1_Crucial_Position : String</div><div>d17out_Observed_Degraded_Point_Position : String</div></div><div><div>D10in_PMi_Crucial_Position : String</div><div><div>D10in_PM1_Non_Crucial_Position : String</div><div>D10in_PMk_Non_Crucial_Position : String</div></div><div><div>D26in_Con_PMi_Crucial_Activation : Boolean</div><div>D27in_Con_PM1_Non_Crucial_Activation : Boolean</div><div>D28in_Con_PMk_Non_Crucial_Activation : Boolean</div></div><div><div>d51in_EST_EfeS_State : String</div></div></div></div></div></div></div>		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6004	Def	D10in_PM1_Crucial_Position	The port D10in_PM1_Crucial_Position represents the Point position that was observed at the interface of the first crucial Point machine. The Point position is derived from the InformationFlow Information_No_End_Position, Information_End_Position_Reached, Information_End_Position_Detected and Information_Unintended_Position.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6005	Def	D10in_PMi_Crucial_Position	The port D10in_PMi_Crucial_Position represents the Point position that was observed at the interface of the n-th crucial Point machine. The Point position is derived from the InformationFlow Information_No_End_Position and Information_Unintended_Position.	Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6006	Def	D10in_PM1_Non_Crucial_Position	The port D10in_PM1_Non_Crucial_Position represents the Point position that was observed at the interface of the first non-crucial Point machine. The Point position is derived from the InformationFlow Information_No_End_Position and Information_Unintended_Position.	Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6007	Def	D10in_PMk_Non_Crucial_Position	The port D10in_PMk_Non_Crucial_Position represents the Point position that was observed at the interface of the n-th non-crucial Point machine. The Point position is derived from the InformationFlow Information_No_End_Position and Information_Unintended_Position.	Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6008	Def	d17out_Observed_Degraded_Point_Position		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6009	Def	d51in_EST_EfeS_State		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6010	Def	D26in_Con_PMi_Crucial_Activation	The port D26in_Con_PMi_Crucial_Activation provides configuration values for the Point machine PM1n crucial and representing a n-th crucial Point machine in configurations with more than one Point machine. true: Point machine PM1n Crucial is activated false: Point machine PM1n Crucial is not activated	Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6011	Def	D27in_Con_PM1_Non_Crucial_Activation	The port D27in_Con_PM1_Non_Crucial_Activation provides configuration values for the Point machine PM2 non crucial, representing the 1st non crucial Point machine in configurations with more than one Point machine. true: Point machine PM2 Crucial is activated false: Point machine PM2 Crucial is not activated	Basic non-4-wire multiple P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6012	Def	D28in_Con_PMc_Non_Crucial_Activation	<div>The port D28in_Con_PMc_Non_Crucial_Activation provides configuration values for the Point machine PM2n non crucial, representing a n-th non crucial Point machine in configurations with more than one Point machine.</div> <div>true: Point machine PM2n Non Crucial is activated false: Point machine PM2n Non Crucial is not activated</div>	Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6013	Def	d14in_Observed_Point_Position		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6014	Info	F_Observe_Degraded_Point_Position - Behaviour		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6015	Req	<div>Functional Viewpoint - Subsystem Requirements - Functional Entity STD 3</div> <div>stm [State Machine] F_Observe_Degraded_Point_Position - Behaviour [Functional Viewpoint - Subsystem Requirements - Functional Entity STD 3]</div> <div></div>	<div>This state machine diagram describes the requirements for the following functionalities:</div> <div>- observes and forward the degraded point position state</div>	Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6230	Def	Initial0		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6231	Def	{Initial0 - OBSERVE_DEGRADED_POINT_POSITION}		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6232	Def	OBSERVE_DEGRADED_POINT_POSITION		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6016	Def	DEGRADED_LEFT		Basic non-4-wire multiple P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6017	Def	entry/d17out_Observed_Degraded_Point_Position := "DEGRADED_LEFT";{State-internal in DEGRADED_LEFT}		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6018	Def	when((d14in_Observed_Point_Position = "LEFT" OR d14in_Observed_Point_Position = "RIGHT") OR ((D10in_PMi_Crucial_Position <> D10in_PMi_Crucial_Position) AND D26in_Con_PMi_Crucial_Activation) OR ((D10in_PMi_Non_Crucial_Position <> D10in_PMi_Crucial_Position AND D10in_PMi_Non_Crucial_Position <> "UNINTENDED_POSITION" AND D10in_PMi_Non_Crucial_Position <> "NO_END_POSITION")AND D27in_Con_PMi_Non_Crucial_Activation) OR ((D10in_PMk_Non_Crucial_Position <> D10in_PMi_Crucial_Position AND D10in_PMk_Non_Crucial_Position <> "UNINTENDED_POSITION" AND D10in_PMk_Non_Crucial_Position <> "NO_END_POSITION")AND D28in_Con_PMk_Non_Crucial_Activation))){DEGRADED_LEFT - NOT_DEGRADED}		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6019	Def	when(D10in_PMi_Crucial_Position = "RIGHT" AND (D10in_PMi_Crucial_Position = "RIGHT" OR NOT D26in_Con_PMi_Crucial_Activation) AND ((D10in_PMk_Non_Crucial_Position = "NO_END_POSITION" OR D10in_PMk_Non_Crucial_Position = "UNINTENDED_POSITION") OR (D10in_PMi_Non_Crucial_Position = "NO_END_POSITION" OR D10in_PMi_Non_Crucial_Position = "UNINTENDED_POSITION"))){DEGRADED_LEFT - DEGRADED_RIGHT}		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6020	Def	DEGRADED_RIGHT		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6021	Def	entry/d17out_Observed_Degraded_Point_Position := "DEGRADED_RIGHT";{State-internal in DEGRADED_RIGHT}		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6022	Def	when(((d14in_Observed_Point_Position = "LEFT" OR d14in_Observed_Point_Position = "RIGHT") OR ((D10in_PMi_Crucial_Position <> D10in_PMi_Crucial_Position) AND D26in_Con_PMi_Crucial_Activation) OR ((D10in_PMi_Non_Crucial_Position <> D10in_PMi_Crucial_Position AND D10in_PMi_Non_Crucial_Position <> "UNINTENDED_POSITION" AND D10in_PMi_Non_Crucial_Position <> "NO_END_POSITION")AND D27in_Con_PMi_Non_Crucial_Activation) OR ((D10in_PMk_Non_Crucial_Position <> D10in_PMi_Crucial_Position AND D10in_PMk_Non_Crucial_Position <> "UNINTENDED_POSITION" AND D10in_PMk_Non_Crucial_Position <> "NO_END_POSITION")AND D28in_Con_PMk_Non_Crucial_Activation))){DEGRADED_RIGHT - NOT_DEGRADED}		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6023	Def	when(D10in_PMi_Crucial_Position = "LEFT" AND (D10in_PMi_Crucial_Position = "LEFT" OR NOT D26in_Con_PMi_Crucial_Activation) AND ((D10in_PMk_Non_Crucial_Position = "NO_END_POSITION" OR D10in_PMk_Non_Crucial_Position = "UNINTENDED_POSITION") OR (D10in_PMi_Non_Crucial_Position = "NO_END_POSITION" OR D10in_PMi_Non_Crucial_Position = "UNINTENDED_POSITION"))){DEGRADED_RIGHT - DEGRADED_LEFT}		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6024	Def	Initial1		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6025	Def	/ {Initial1 - WAITING_FOR_INITIALISATION}		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6233	Def	Junction1		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6234	Def	[D10in_PMi_Crucial_Position = "LEFT" AND (D10in_PMi_Crucial_Position = "LEFT" OR NOT D26in_Con_PMi_Crucial_Activation) AND ((D10in_PMk_Non_Crucial_Position = "NO_END_POSITION" OR D10in_PMk_Non_Crucial_Position = "UNINTENDED_POSITION") OR (D10in_PMi_Non_Crucial_Position = "NO_END_POSITION" OR D10in_PMi_Non_Crucial_Position = "UNINTENDED_POSITION"))]/ {Junction1 - DEGRADED_LEFT}		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6235	Def	[D10in_PMi_Crucial_Position = "RIGHT" AND (D10in_PMi_Crucial_Position = "RIGHT" OR NOT D26in_Con_PMi_Crucial_Activation) AND ((D10in_PMk_Non_Crucial_Position = "NO_END_POSITION" OR D10in_PMk_Non_Crucial_Position = "UNINTENDED_POSITION") OR (D10in_PMi_Non_Crucial_Position = "NO_END_POSITION" OR D10in_PMi_Non_Crucial_Position = "UNINTENDED_POSITION"))]/ {Junction1 - DEGRADED_RIGHT}		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6236	Def	[else]/ {Junction1 - NOT_DEGRADED}		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6237	Def	[NOT D27in_Con_PMi_Non_Crucial_Activation AND NOT D28in_Con_PMk_Non_Crucial_Activation]/ {Junction1 - DEGRADED_DENIED}		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6026	Def	NOT_DEGRADED		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6027	Def	entry/d17out_Observed_Degraded_Point_Position := "NOT_DEGRADED";{State-internal in NOT_DEGRADED}		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6028	Def	when(D10in_PMi_Crucial_Position = "LEFT" AND (D10in_PMi_Crucial_Position = "LEFT" OR NOT D26in_Con_PMi_Crucial_Activation) AND (((D10in_PMk_Non_Crucial_Position = "NO_END_POSITION" OR D10in_PMk_Non_Crucial_Position = "UNINTENDED_POSITION") AND D10in_PMi_Non_Crucial_Position <> "RIGHT")) OR (((D10in_PMi_Non_Crucial_Position = "NO_END_POSITION" OR D10in_PMi_Non_Crucial_Position = "UNINTENDED_POSITION") AND D10in_PMk_Non_Crucial_Position <> "RIGHT"))){NOT_DEGRADED - DEGRADED_LEFT}		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6029	Def	when(D10in_PMi_Crucial_Position = "RIGHT" AND (D10in_PMi_Crucial_Position = "RIGHT" OR NOT D26in_Con_PMi_Crucial_Activation) AND (((D10in_PMk_Non_Crucial_Position = "NO_END_POSITION" OR D10in_PMk_Non_Crucial_Position = "UNINTENDED_POSITION") AND D10in_PMi_Non_Crucial_Position <> "LEFT")) OR (((D10in_PMi_Non_Crucial_Position = "NO_END_POSITION" OR D10in_PMi_Non_Crucial_Position = "UNINTENDED_POSITION") AND D10in_PMk_Non_Crucial_Position <> "LEFT"))){NOT_DEGRADED - DEGRADED_RIGHT}		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6238	Def	WAITING_FOR_INITIALISATION		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6239	Def	when(d51in_EST_EfeS_State = "INITIALISING")/d17out_Observed_Degraded_Point_Position := "UNDEFINED";{WAITING_FOR_INITIALISATION - Junction1}		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6240	Def	when(d51in_EST_EfeS_State = "NO_OPERATING_VOLTAGE" OR d51in_EST_EfeS_State = "BOOTING" OR d51in_EST_EfeS_State = "FALLBACK_MODE")/{OBSERVE_DEGRADED_POINT_POSITION - OBSERVE_DEGRADED_POINT_POSITION}		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6406	Def	DEGRADED_DENIED		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6407	Def	entry/d17out_Observed_Degraded_Point_Position := "DEGRADED_DENIED";{State-internal in DEGRADED_DENIED}		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6993	Info	F_Observe_Movement_Failed		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6994	Req	<div>[Block] F_Observe_Movement_Failed [Functional Viewpoint - Subsystem Requirements - Functional Entity]</div> <div>ibd [Block] F_Observe_Movement_Failed [Functional Viewpoint - Subsystem Requirements - Functional Entity]</div> <div><div>«functional entity» F_Observe_Movement_Failed</div><div><div>d12in_Required_PM_Position : String</div><div>d13out_Observed_Movement_Failed : Boolean</div><div>D20in_Con_tmax_PM_Operation : Integer</div><div>t21in_Movement_Failed : PulsedIn</div></div></div>		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6995	Def	d12in_Required_PM_Position		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6996	Def	d13out_Observed_Movement_Failed		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6998	Def	D20in_Con_tmax_PM_Operation	The port D20in_Con_tmax_PM_Operation refines the time value Con_tmax_Point_Operation.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7015	Def	t21in_Movement_Failed	The port t21in_Movement_Failed represents all internal occurrences of conditions to the Subsystem - Point	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6999	Info	F_Observe_Movement_Failed - Behaviour		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7000	Req	<div>Functional Viewpoint - Subsystem Requirements - Functional Entity STD 4</div> <div>stm [State Machine] F_Observe_Movement_Failed - Behaviour [Functional Viewpoint - Subsystem Requirements - Functional Entity STD 4]</div> <div><div>Initial0</div><div><div>OBSERVE_MOVEMENT_FAILURE</div><div><div>Initial1</div><div><div>IDLE</div><div>Entry/d13out_Observed_Movement_Failed := FALSE;</div><div>when(d12in_Required_PM_Position = "LEFT" OR d12in_Required_PM_Position = "RIGHT")/ when(d12in_Required_PM_Position = "UNCOMMANDED")/</div></div><div><div>OBSERVING_MOVEMENT_FAILURE</div><div>after(D20in_Con_tmax_PM_Operation)/ d13out_Observed_Movement_Failed := TRUE; when(t21in_Movement_Failed)/ d13out_Observed_Movement_Failed := TRUE;</div><div>when(d12in_Required_PM_Position = "LEFT")/ when(d12in_Required_PM_Position = "RIGHT")/</div></div></div></div></div>	This state machine diagram describes the requirements for the following functionalities: - monitors the Con_tmax_PM_Operation and forwards an overrun as a Movent Failure - observes and forward a Movement failure	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7001	Def	Initial0		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7002	Def	/{Initial0 - OBSERVE_MOVEMENT_FAILURE}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7003	Def	OBSERVE_MOVEMENT_FAILURE		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7004	Def	IDLE		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7005	Def	entry/d13out_Observed_Movement_Failed := FALSE;{State-internal in IDLE}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7006	Def	when(d12in_Required_PM_Position = "LEFT" OR d12in_Required_PM_Position = "RIGHT")/{IDLE - OBSERVING_MOVEMENT_FAILURE}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7007	Def	Initial1		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7008	Def	/{Initial1 - IDLE}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7009	Def	OBSERVING_MOVEMENT_FAILURE		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7010	Def	after(D20in_Con_tmax_PM_Operation)/ d13out_Observed_Movement_Failed := TRUE;{State-internal in OBSERVING_MOVEMENT_FAILURE}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7011	Def	when(t21in_Movement_Failed)/ d13out_Observed_Movement_Failed := TRUE;{State-internal in OBSERVING_MOVEMENT_FAILURE}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7012	Def	when(d12in_Required_PM_Position = "LEFT")/{OBSERVING_MOVEMENT_FAILURE - OBSERVING_MOVEMENT_FAILURE}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7013	Def	when(d12in_Required_PM_Position = "RIGHT")/{OBSERVING_MOVEMENT_FAILURE - OBSERVING_MOVEMENT_FAILURE}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7014	Def	when(d12in_Required_PM_Position = "UNCOMMANDED")/{OBSERVING_MOVEMENT_FAILURE - IDLE}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5980	Info	F_Observe_Ability_To_Move		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5981	Req	<div><div>[Block] F_Observe_Ability_To_Move [Functional Viewpoint - Subsystem Requirements - Functional Entity]</div><div><div>ibdd [Block] F_Observe_Ability_To_Move [Functional Viewpoint - Subsystem Requirements - Functional Entity]</div><div><div>«functional entity» F_Observe_Ability_To_Move</div><div><div><div><div>→ D15in_Con_Observe_Ability_To_Move : Boolean</div><div>→ D26in_Con_PMi_Crucial_Activation : Boolean</div><div>→ D27in_Con_PM1_Non_Crucial_Activation : Boolean</div><div>→ D28in_Con_PMk_Non_Crucial_Activation : Boolean</div><div>→ D19in_Ability_To_Move_PM1_Crucial : String</div><div>→ D19in_Ability_To_Move_PM1n_Crucial : String</div><div>→ D19in_Ability_To_Move_PM2n_Non_Crucial : String</div><div>→ D19in_Ability_To_Move_PM2_Non_Crucial : String</div><div><div>→ D33in_Internal_Trigger_Ability_To_Move_Point_Available : Boolean</div><div>→ d51in_EST_EfeS_State : String</div></div><div>d11out_Observed_Ability_To_Move : String →</div></div></div></div></div></div></div>		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.5982	Def	D19in_Ability_To_Move_PM1_Crucial	The port D19in_Ability_To_Move_PM1_Crucial refines the InformationFlow Information_Ability_To_Move_Point from the first Point machine to Subsystem - Point. Note - Drive voltage levels monitored shall be defined by national specifications. In future phases of the System Pillar, national specifications will be replaced by harmonised specifications.	Option Able to move
Eu.P.5983	Def	D19in_Ability_To_Move_PM1n_Crucial	The port D19in_Ability_To_Move_PM1n_Crucial refines the InformationFlow Information_Ability_To_Move_Point from the n-th Point machine to Subsystem - Point. Note - Drive voltage levels monitored shall be defined by national specifications. In future phases of the System Pillar, national specifications will be replaced by harmonised specifications. Only applicable if the package [Option Able to move] is used in combination with a [Basic non-4-wire multiple P] or [Basic 4-wire multiple P].	Option Able to move
Eu.P.5984	Def	D19in_Ability_To_Move_PM2_Non_Crucial	The port D19in_Ability_To_Move_PM2_Non_Crucial refines the InformationFlow Information_Ability_To_Move_Point from the n-th Point machine to Subsystem - Point. Note - Drive voltage levels monitored shall be defined by national specifications. In future phases of the System Pillar, national specifications will be replaced by harmonised specifications. Only applicable if the package [Option Able to move] is used in combination with a [Basic non-4-wire multiple P] or [Basic 4-wire multiple P].	Option Able to move
Eu.P.5985	Def	D19in_Ability_To_Move_PM2n_Non_Crucial	The port D19in_Ability_To_Move_PM2n_Non_Crucial refines the InformationFlow Information_Ability_To_Move_Point from the n-th Point machine to Subsystem - Point. Note - Drive voltage levels monitored shall be defined by national specifications. In future phases of the System Pillar, national specifications will be replaced by harmonised specifications. Only applicable if the package [Option Able to move] is used in combination with a [Basic non-4-wire multiple P] or [Basic 4-wire multiple P].	Option Able to move
Eu.P.5986	Def	D33in_Internal_Trigger_Ability_To_Move_Point_Available	The port D33in_Internal_Trigger_Ability_To_Move_Point_Available represents all internal occurrences of conditions to the Subsystem - Point, leading to inability to move.	Option Able to move
Eu.P.5987	Def	d11out_Observed_Ability_To_Move		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5988	Def	d51in_EST_EfeS_State		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6224	Def	D26in_Con_PMi_Crucial_Activation	The port D26in_Con_PMi_Crucial_Activation provides configuration values for the Point machine PM1n crucial and representing a n-th crucial Point machine in configurations with more than one Point machine. true: Point machine PM1n Crucial is activated false: Point machine PM1n Crucial is not activated	Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6225	Def	D27in_Con_PM1_Non_Crucial_Activation	The port D27in_Con_PM1_Non_Crucial_Activation provides configuration values for the Point machine PM2 non crucial, representing the 1st non crucial Point machine in configurations with more than one Point machine.	Basic non-4-wire multiple P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
			true: Point machine PM2 Non Crucial is activated false: Point machine PM2 Non Crucial is not activated	
Eu.P.6226	Def	D28in_Con_PMk_Non_Crucial_Activation	The port D28in_Con_PMk_Non_Crucial_Activation provides configuration values for the Point machine PM2n non crucial, representing a n-th non crucial Point machine in configurations with more than one Point machine. true: Point machine PM2n Non Crucial is activated false: Point machine PM2n Non Crucial is not activated	Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6991	Def	D15in_Con_Observe_Ability_To_Move		Option Able to move
Eu.P.5992	Info	F_Observe_Ability_To_Move - Behaviour		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5993	Req	<div>Functional Viewpoint - Subsystem Requirements - Functional Entity STD 5</div> <div>strn [State Machine] F_Observe_Ability_To_Move - Behaviour [Functional Viewpoint - Subsystem Requirements - Functional Entity STD 5]</div>	This state machine diagram describes the requirements for the following functionalities: - observes and forward the Ability to move state	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6227	Def	Initial0		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6228	Def	/{Initial0 - WAITING_FOR_INITIALISING}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6229	Def	WAITING_FOR_INITIALISING		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6405	Def	when(d51in_EST_EfeS_State = \"INITIALISING\")/{WAITING_FOR_INITIALISING - Junction0}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6396	Def	ABLE_TO_MOVE		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6397	Def	entry/d11out_Observed_Ability_To_Move := "ABLE_TO_MOVE";{State-internal in ABLE_TO_MOVE}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6398	Def	when(D19in_Ability_To_Move_PM1_Crucial = "UNABLE" OR D19in_Ability_To_Move_PM1n_Crucial = "UNABLE" OR D19in_Ability_To_Move_PM2_Non_Crucial = "UNABLE" OR D19in_Ability_To_Move_PM2n_Non_Crucial = "UNABLE" OR NOT D33in_Internal_Trigger_Ability_To_Move_Point_Available)[D15in_Con_Observe_Ability_To_Move]/{ABLE_TO_MOVE - UNABLE_TO_MOVE}		Option Able to move
Eu.P.6399	Def	Junction0		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6400	Def	[D19in_Ability_To_Move_PM1_Crucial = "ABLE" AND (D19in_Ability_To_Move_PM1n_Crucial = "ABLE" OR D19in_Ability_To_Move_PM1n_Crucial = "NOT_USED" OR NOT D26in_Con_PMi_Crucial_Activation) AND (D19in_Ability_To_Move_PM2n_Non_Crucial = "ABLE" OR D19in_Ability_To_Move_PM2n_Non_Crucial = "NOT_USED" OR NOT D28in_Con_PMk_Non_Crucial_Activation) AND (D19in_Ability_To_Move_PM2_Non_Crucial = "ABLE" OR D19in_Ability_To_Move_PM2_Non_Crucial = "NOT_USED" OR NOT D27in_Con_PM1_Non_Crucial_Activation) AND D33in_Internal_Trigger_Ability_To_Move_Point_Available AND D15in_Con_Observe_Ability_To_Move]/{Junction0 - ABLE_TO_MOVE}		Option Able to move
Eu.P.6401	Def	[(D19in_Ability_To_Move_PM1_Crucial = "UNABLE" OR D19in_Ability_To_Move_PM1n_Crucial = "UNABLE" OR D19in_Ability_To_Move_PM2_Non_Crucial = "UNABLE" OR D19in_Ability_To_Move_PM2n_Non_Crucial = "UNABLE" OR NOT D33in_Internal_Trigger_Ability_To_Move_Point_Available) AND D15in_Con_Observe_Ability_To_Move]/{Junction0 - UNABLE_TO_MOVE}		Option Able to move
Eu.P.6992	Def	[NOT D15in_Con_Observe_Ability_To_Move]/{Junction0 - ABLE_TO_MOVE}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6402	Def	UNABLE_TO_MOVE		Option Able to move
Eu.P.6403	Def	entry/d11out_Observed_Ability_To_Move := "UNABLE_TO_MOVE";{State-internal in UNABLE_TO_MOVE}		Option Able to move
Eu.P.6404	Def	when(D19in_Ability_To_Move_PM1_Crucial = "ABLE" AND (D19in_Ability_To_Move_PM1n_Crucial = "ABLE" OR D19in_Ability_To_Move_PM1n_Crucial = "NOT_USED" OR NOT D26in_Con_PMi_Crucial_Activation) AND (D19in_Ability_To_Move_PM2n_Non_Crucial = "ABLE" OR D19in_Ability_To_Move_PM2n_Non_Crucial = "NOT_USED" OR NOT D28in_Con_PMk_Non_Crucial_Activation) AND (D19in_Ability_To_Move_PM2_Non_Crucial = "ABLE" OR D19in_Ability_To_Move_PM2_Non_Crucial = "NOT_USED" OR NOT D27in_Con_PM1_Non_Crucial_Activation) AND D33in_Internal_Trigger_Ability_To_Move_Point_Available)/{UNABLE_TO_MOVE - ABLE_TO_MOVE}		Option Able to move
Eu.P.5925	Info	F_Control_Non4W_PM		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5926	Req	<div><div>[Block] F_Control_Non4W_PM [Functional Viewpoint - Subsystem Requirements - Functional Entity]</div><div><div>ibd [Block] F_Control_Non4W_PM [Functional Viewpoint - Subsystem Requirements - Functional Entity]</div><div><div>«functional entity» F_Control_Non4W_PM</div><div><div>d12in_Required_PM_Position : String</div><div>d34out_Drive_Stop : Boolean</div><div>D35in_PM_Position : String</div><div>D38out_Move_Right : Boolean</div><div>D36in_Con_Drive_Capability : Boolean</div><div>D39out_Move_Left : Boolean</div><div>D37in_Con_Common_Drive : Boolean</div><div>D40in_Con_Active : Boolean</div><div>d51in_EST_EfeS_State : String</div></div></div></div></div>		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5927	Def	D35in_PM_Position	The port D35in_PM_Position provides the Point machine position to the Subsystem - Point. The port D35in_PM_Position refines the InformationFlow Information_No_End_Position and Information_Unintended_Position.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5928	Def	D39out_Move_Left	The port D39out_Move_Left refines the Informationflow Moving and contains the information if driving left was started. The following values are valid: "True" (driving left), "False" (not driving left)	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5929	Def	D38out_Move_Right	The port D38out_Move_Right provides the command Moving right to the point machine and refines the InformationFlow Moving and Stop moving. The following values are valid: "True" (driving right), "False" (not driving right, Stop moving).	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.5930	Def	d12in_Required_PM_Position		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5932	Def	d34out_Drive_Stop		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5934	Def	d51in_EST_EfeS_State		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5936	Def	D37in_Con_Common_Drive	The port D37in_Con_Common_Drive provides a configuration value to the Point for Redrive point functionality. true: Point has Common Drive functionality false: Point has Individual Drive functionality	Option Common Drive
Eu.P.6222	Def	D40in_Con_Active	The port D40in_Con_Active provides configuration values for the Point machine. The Point machine in single Point machine configurations and the first crucial Point machine in multiple Point machine configurations is always activated. true: Point machine is activated false: Point machine is not activated	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6223	Def	D36in_Con_Drive_Capability	The port D36in_Con_Drive_Capability provides a configuration value for the Point machine telling the Subsystem - Point whether the Point machine is configured with full drive capability or as Point detector only. true: full drive capability false: Point detector only	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5938	Info	F_Control_Non4W_PM - Behaviour		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.5939	Req	<div>Functional Viewpoint - Subsystem Requirements - Functional Entity STD 6</div> <div>stm [State Machine] F_Control_Non4W_PM - Behaviour [Functional Viewpoint - Subsystem Requirements - Functional Entity STD 6]</div> <div><pre>stateDiagram-v2 [*] --> Initial0 state Initial0 { when(d51in_EST_EfeS_State = "NO_OPERATING_VOLTAGE" OR d51in_EST_EfeS_State = "BOOTING" OR d51in_EST_EfeS_State = "FALLBACK_MODE") / OPERATING Entry/D39out_Move_Left := FALSE; D38out_Move_Right := FALSE; d34out_Drive_Stop := TRUE; } state Initial1 { [*] --> WAITING WAITING --> STOPPED : when(d51in_EST_EfeS_State = "INITIALISING") / } state STOPPED { Entry/D39out_Move_Left := FALSE; D38out_Move_Right := FALSE; d34out_Drive_Stop := TRUE; when(d12in_Required_PM_Position = D35in_PM_Position) [NOT D37in_Con_Common_Drive] / when(d12in_Required_PM_Position = "UNCOMMANDED") / } state MOVING_LEFT { Entry/D39out_Move_Left := TRUE; d34out_Drive_Stop := FALSE; when(d12in_Required_PM_Position = "LEFT" AND ((d12in_Required_PM_Position <> D35in_PM_Position) OR (d12in_Required_PM_Position = D35in_PM_Position AND D37in_Con_Common_Drive))) [D40in_Con_Active AND D36in_Con_Drive_Capability] / when(d12in_Required_PM_Position = "RIGHT") [(d12in_Required_PM_Position <> D35in_PM_Position) OR (d12in_Required_PM_Position = D35in_PM_Position AND D37in_Con_Common_Drive)] /D39out_Move_Left := FALSE; when(d12in_Required_PM_Position = "LEFT") [(d12in_Required_PM_Position <> D35in_PM_Position) OR (d12in_Required_PM_Position = D35in_PM_Position AND D37in_Con_Common_Drive)] /D38out_Move_Right := FALSE; } state MOVING_RIGHT { Entry/D38out_Move_Right := TRUE; d34out_Drive_Stop := FALSE; when(d12in_Required_PM_Position = "RIGHT" AND ((d12in_Required_PM_Position <> D35in_PM_Position) OR (d12in_Required_PM_Position = D35in_PM_Position AND D37in_Con_Common_Drive))) [D40in_Con_Active AND D36in_Con_Drive_Capability] / } STOPPED --> MOVING_LEFT : when(d12in_Required_PM_Position = D35in_PM_Position) [NOT D37in_Con_Common_Drive] / STOPPED --> MOVING_RIGHT : when(d12in_Required_PM_Position = "UNCOMMANDED") / MOVING_LEFT --> STOPPED : when(d12in_Required_PM_Position = "LEFT" AND ((d12in_Required_PM_Position <> D35in_PM_Position) OR (d12in_Required_PM_Position = D35in_PM_Position AND D37in_Con_Common_Drive))) [D40in_Con_Active AND D36in_Con_Drive_Capability] / MOVING_RIGHT --> STOPPED : when(d12in_Required_PM_Position = "RIGHT" AND ((d12in_Required_PM_Position <> D35in_PM_Position) OR (d12in_Required_PM_Position = D35in_PM_Position AND D37in_Con_Common_Drive))) [D40in_Con_Active AND D36in_Con_Drive_Capability] /</pre></div>	<p>This state machine diagram describes the requirements for the following functionalities:</p> <ul style="list-style-type: none">- observes and forward the Moving state for an non 4W PM- observes and forward the Drive stop state for an non 4W PM	<p>Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P</p>
Eu.P.5940	Def	Initial0		<p>Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P</p>
Eu.P.5941	Def	/ {Initial0 - OPERATING}		<p>Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P</p>
Eu.P.5942	Def	OPERATING		<p>Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P</p>
Eu.P.5947	Def	Initial1		<p>Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P</p>
Eu.P.5948	Def	/ {Initial1 - WAITING}		<p>Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P</p>
Eu.P.5955	Def	MOVING_LEFT		<p>Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P</p>
Eu.P.5957	Def	entry/D39out_Move_Left := TRUE; d34out_Drive_Stop := FALSE; {State-internal in MOVING_LEFT}		<p>Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P</p>

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.5960	Def	when(d12in_Required_PM_Position = "RIGHT")[(d12in_Required_PM_Position <> D35in_PM_Position) OR (d12in_Required_PM_Position = D35in_PM_Position AND D37in_Con_Common_Drive)]/D39out_Move_Left := FALSE;{MOVING_LEFT - MOVING_RIGHT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5961	Def	when(d12in_Required_PM_Position = "UNCOMMANDED")/{MOVING_LEFT - STOPPED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6301	Def	when(d12in_Required_PM_Position = D35in_PM_Position)[NOT D37in_Con_Common_Drive]/{MOVING_LEFT - STOPPED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5965	Def	MOVING_RIGHT		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5967	Def	entry/D38out_Move_Right := TRUE; d34out_Drive_Stop := FALSE;{State-internal in MOVING_RIGHT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5970	Def	when(d12in_Required_PM_Position = "LEFT")[(d12in_Required_PM_Position <> D35in_PM_Position) OR (d12in_Required_PM_Position = D35in_PM_Position AND D37in_Con_Common_Drive)]/D38out_Move_Right := FALSE;{MOVING_RIGHT - MOVING_LEFT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5971	Def	when(d12in_Required_PM_Position = "UNCOMMANDED")/{MOVING_RIGHT - STOPPED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6303	Def	when(d12in_Required_PM_Position = D35in_PM_Position)[NOT D37in_Con_Common_Drive]/{MOVING_RIGHT - STOPPED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5975	Def	STOPPED		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5976	Def	when(d12in_Required_PM_Position = "LEFT" AND ((d12in_Required_PM_Position <> D35in_PM_Position) OR (d12in_Required_PM_Position = D35in_PM_Position AND D37in_Con_Common_Drive)))[D40in_Con_Active AND D36in_Con_Drive_Capability]/{STOPPED - MOVING_LEFT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5977	Def	when(d12in_Required_PM_Position = "RIGHT" AND ((d12in_Required_PM_Position <> D35in_PM_Position) OR (d12in_Required_PM_Position = D35in_PM_Position AND D37in_Con_Common_Drive)))[D40in_Con_Active AND D36in_Con_Drive_Capability]/{STOPPED - MOVING_RIGHT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6304	Def	entry/D39out_Move_Left := FALSE; D38out_Move_Right := FALSE; d34out_Drive_Stop := TRUE;{State-internal in STOPPED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6392	Def	entry/D39out_Move_Left := FALSE; D38out_Move_Right := FALSE; d34out_Drive_Stop := TRUE;{State-internal in OPERATING}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6393	Def	WAITING		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6394	Def	when(d51in_EST_EfeS_State = "INITIALISING")/{WAITING - STOPPED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6395	Def	when(d51in_EST_EfeS_State = "NO_OPERATING_VOLTAGE" OR d51in_EST_EfeS_State = "BOOTING" OR d51in_EST_EfeS_State = "FALLBACK_MODE")/{OPERATING - OPERATING}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5810	Info	F_Control_And_Observe_4W_PM		Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.5811	Req	<div>[Block] F_Control_And_Observe_4W_PM [Functional Viewpoint - Subsystem Requirements - Functional Entity]</div> <div><div>ibd [Block] F_Control_And_Observe_4W_PM [Functional Viewpoint - Subsystem Requirements - Functional Entity]</div><div><div>«functional entity»</div><div>F_Control_And_Observe_4W_PM</div><div><div><div><div>D20in_Con_Drive_Capability : Boolean</div><div>d10out_PM_Position : String</div><div><div></div></div></div><div><div><div>d12in_Required_PM_Position : String</div><div>d19out_Ability_To_Move_PM : String</div><div><div></div></div></div><div><div><div>D26in_Internal_Trigger_Ability_To_Move_PM_Available : Boolean</div><div>D23out_Drive_Voltage_Left : Boolean</div><div><div></div></div></div><div><div><div>D27in_4W_PM_Position : String</div><div>D24out_Drive_Voltage_Right : Boolean</div><div><div></div></div></div><div><div><div>D45in_Con_Active : Boolean</div><div>D25out_Detection_Voltage : Boolean</div><div><div></div></div></div><div><div><div>d51in_EST_EfeS_State : String</div><div></div><div><div></div></div></div></div></div></div></div></div></div></div></div></div>		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5812	Def	D20in_Con_Drive_Capability	<div>The port D20in_Con_Drive_Capability provides configuration values for the Point machine.</div> <div>true: Point machine has drive capability false: Point machine has no drive capability</div>	Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5813	Def	d19out_Ability_To_Move_PM		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5814	Def	d10out_PM_Position		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5817	Def	D27in_4W_PM_Position	<div>The port D27in_4W_PM_Position refines the Informationflow Information_No_End_Position, Information_End_Position_Reached, Information_End_Position_Detected and Information_Unintended_Position and is a representative for the position of an undefined number of Point machines, in case they are implemented with a 4-wire interface.</div> <div>Note: The same 4-wire input pattern doesn't always represent the same Informationflow. The interpretation of all patterns is fully defined in Eu.P.6797.</div>	Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5818	Def	D26in_Internal_Trigger_Ability_To_Move_PM_Available	<div>The port D26in_Internal_Trigger_Ability_To_Move_PM_Available represents all internal occurrences of conditions concerning the Point machine, leading to inability to move.</div> <div>true: Point machine is able to move false: Point machine is unable to move</div>	Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5819	Def	D25out_Detection_Voltage	<div>The port D25out_Detection_Voltage refines the Informationflow Detection_Voltage and contains the information if Detection voltage is provided to the Point machine via the 4-wire interface.</div> <div>The following values are valid: "True" (detection), "False" (no detection)</div>	Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5820	Def	D24out_Drive_Voltage_Right	<div>The port D24out_Drive_Voltage_Right refines the Informationflow Moving and contains the information if Drive voltage for driving right is provided to the Point machine via the 4-wire interface.</div> <div>The following values are valid: "True" (driving right), "False" (not driving right)</div>	Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5821	Def	D23out_Drive_Voltage_Left	<div>The port D23out_Drive_Voltage_Left refines the Informationflow Moving and contains the information if Drive voltage for driving left is provided to the Point machine via the 4-wire interface.</div> <div>The following values are valid: "True" (driving left), "False" (not driving left)</div>	Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6209	Def	d12in_Required_PM_Position		Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6211	Def	D45in_Con_Active	The port D45in_Con_Active provides configuration values for the Point machine. true: Point machine is activated false: Point machine is not activated	Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6212	Def	d51in_EST_EfeS_State		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5822	Info	F_Control_And_Observe_4W_PM - Behaviour		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5823	Req	Functional Viewpoint - Subsystem Requirements - Functional Entity STD 7 <div>stm [State Machine] F_Control_And_Observe_4W_PM - Behaviour [Functional Viewpoint - Subsystem Requirements - Functional Entity STD 7]</div> <pre>stateDiagram-v2 [*] --> WAITING_FOR_INITIALISING : Initial0 WAITING_FOR_INITIALISING --> OPERATING : when(d51in_EST_EfeS_State = "BOOTING") [D45in_Con_Active] / OPERATING --> OPERATING : when(D26in_Internal_Trigger_Ability_To_Move_PM_Available) /d19out_Ability_To_Move_PM := "ABLE"; OPERATING --> OPERATING : when(NOT D26in_Internal_Trigger_Ability_To_Move_PM_Available) /d19out_Ability_To_Move_PM := "UNABLE"; OPERATING --> DETECTION : Initial1 DETECTION --> DETECTION : when(d12in_Required_PM_Position = "UNCOMMANDED") / DETECTION --> DETECTION : when(D27in_4W_PM_Position = "LEFT_DETECTED") /d10out_PM_Position := "LEFT"; DETECTION --> DETECTION : when(D27in_4W_PM_Position = "RIGHT_DETECTED") /d10out_PM_Position := "RIGHT"; DETECTION --> DETECTION : when(D27in_4W_PM_Position = "NO_END_POSITION") /d10out_PM_Position := "NO_END_POSITION"; DETECTION --> DETECTION : when(D27in_4W_PM_Position = "UNINTENDED_POSITION") /d10out_PM_Position := "UNINTENDED_POSITION"; DETECTION --> MOVING_LEFT_PM : when(d12in_Required_PM_Position = "LEFT") [D20in_Con_Drive_Capability AND D45in_Con_Active] / DETECTION --> MOVING_RIGHT_PM : when(d12in_Required_PM_Position = "RIGHT") [D20in_Con_Drive_Capability AND D45in_Con_Active] / MOVING_LEFT_PM --> MOVING_RIGHT_PM : when(d12in_Required_PM_Position = "RIGHT") [D20in_Con_Drive_Capability AND D45in_Con_Active] / MOVING_RIGHT_PM --> MOVING_LEFT_PM : when(d12in_Required_PM_Position = "LEFT") [D20in_Con_Drive_Capability AND D45in_Con_Active] / MOVING_LEFT_PM --> DETECTION : when(d12in_Required_PM_Position = "LEFT_REACHED") / MOVING_RIGHT_PM --> DETECTION : when(d12in_Required_PM_Position = "RIGHT_REACHED") /</pre>	This state machine diagram describes the requirements for the following functionalities: - observes and forward the Moving state for an 4W PM - observes and forward the Detection state for an 4W PM	Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5826	Def	OPERATING		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6221	Def	when(d51in_EST_EfeS_State = "NO_OPERATING_VOLTAGE" OR d51in_EST_EfeS_State = "FALLBACK_MODE")/D24out_Drive_Voltage_Right := FALSE; D23out_Drive_Voltage_Left := FALSE;{OPERATING - WAITING_FOR_INITIALISING}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7078	Def	when(d51in_EST_EfeS_State = "BOOTING")/D24out_Drive_Voltage_Right := FALSE; D23out_Drive_Voltage_Left := FALSE;{OPERATING - OPERATING}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5915	Def	DETECTION		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5918	Def	entry/D25out_Detection_Voltage := TRUE; D24out_Drive_Voltage_Right := FALSE; D23out_Drive_Voltage_Left := FALSE;{State-internal in DETECTION}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6935	Def	when(D27in_4W_PM_Position = "LEFT_DETECTED")/d10out_PM_Position := "LEFT";{State-internal in DETECTION}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6936	Def	when(D27in_4W_PM_Position = "NO_END_POSITION")/d10out_PM_Position := "NO_END_POSITION";{State-internal in DETECTION}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6937	Def	when(D27in_4W_PM_Position = "RIGHT_DETECTED")/d10out_PM_Position := "RIGHT";{State-internal in DETECTION}		Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6938	Def	when(D27in_4W_PM_Position = "UNINTENDED_POSITION")/d10out_PM_Position := "UNINTENDED_POSITION";{State-internal in DETECTION}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6939	Def	when(d12in_Required_PM_Position = "LEFT")[D20in_Con_Drive_Capability AND D45in_Con_Active]/{DETECTION - MOVING_LEFT_PM}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6940	Def	when(d12in_Required_PM_Position = "RIGHT")[D20in_Con_Drive_Capability AND D45in_Con_Active]/{DETECTION - MOVING_RIGHT_PM}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6941	Def	Initial1		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6942	Def	/ {Initial1 - DETECTION}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5888	Def	MOVING_LEFT_PM		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5890	Def	entry/D25out_Detection_Voltage := FALSE; D24out_Drive_Voltage_Right := FALSE; D23out_Drive_Voltage_Left := TRUE; d10out_PM_Position := "NO_END_POSITION";{State-internal in MOVING_LEFT_PM}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5877	Def	when(d12in_Required_PM_Position = "RIGHT")[D20in_Con_Drive_Capability AND D45in_Con_Active]/{MOVING_LEFT_PM - MOVING_RIGHT_PM}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6943	Def	when(d12in_Required_PM_Position = "LEFT_REACHED")/{MOVING_LEFT_PM - DETECTION}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6944	Def	when(d12in_Required_PM_Position = "UNCOMMANDED")/{MOVING_LEFT_PM - DETECTION}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5905	Def	MOVING_RIGHT_PM		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5907	Def	entry/D25out_Detection_Voltage := FALSE; D24out_Drive_Voltage_Right := TRUE; D23out_Drive_Voltage_Left := FALSE; d10out_PM_Position := "NO_END_POSITION";{State-internal in MOVING_RIGHT_PM}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5894	Def	when(d12in_Required_PM_Position = "LEFT")[D20in_Con_Drive_Capability AND D45in_Con_Active]/{MOVING_RIGHT_PM - MOVING_LEFT_PM}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6945	Def	when(d12in_Required_PM_Position = "RIGHT_REACHED")/{MOVING_RIGHT_PM - DETECTION}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6946	Def	when(d12in_Required_PM_Position = "UNCOMMANDED")/{MOVING_RIGHT_PM - DETECTION}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7134	Def	when(D26in_Internal_Trigger_Ability_To_Move_PM_Available)/d19out_Ability_To_Move_PM := "ABLE";{State-internal in OPERATING}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7135	Def	when(NOT D26in_Internal_Trigger_Ability_To_Move_PM_Available)/d19out_Ability_To_Move_PM := "UNABLE";{State-internal in OPERATING}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5909	Def	Initial0		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5910	Def	/ {Initial0 - WAITING_FOR_INITIALISING}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5911	Def	WAITING_FOR_INITIALISING		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6220	Def	when(d51in_EST_EfeS_State = "BOOTING")[D45in_Con_Active]/{WAITING_FOR_INITIALISING - Junction}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6487	Def	Junction		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6488	Def	[D20in_Con_Drive_Capability AND D26in_Internal_Trigger_Ability_To_Move_PM_Available]/d19out_Ability_To_Move_PM := "ABLE";{Junction - OPERATING}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6489	Def	[NOT D20in_Con_Drive_Capability]/d19out_Ability_To_Move_PM := "NOT_USED";{Junction - OPERATING}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6490	Def	[D20in_Con_Drive_Capability AND NOT D26in_Internal_Trigger_Ability_To_Move_PM_Available]/d19out_Ability_To_Move_PM := "UNABLE";{Junction - OPERATING}		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.889	Head	3.4 Subsystem - Point - Interfaces		
Eu.P.913	Head	3.4.1 SCI-P (Subsystem - Electronic Interlocking)		
Eu.P.6154	Head	3.4.1.1 SCI-P - Logical Viewpoint		
Eu.P.6281	Head	3.4.1.1.1 SCI-P - Logical Context		

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.3287	Def	<div><div>[Package] SCI-P - Logical Context [Logical Viewpoint - Interface Definition]</div><div><div>bdd [Package] SCI-P - Logical Context [Logical Viewpoint - Interface Definition]</div><div><div><div><div>Subsystem - Electronic Interlocking</div><div>«logical structural entity» Subsystem - Electronic Interlocking</div><div>1 SCI-P</div></div><div><div>«logical structural entity» SCI-P</div><div>SCI-P</div><div><div>Subsystem - Point - Functional Architecture</div><div>«logical structural entity» Subsystem - Point</div><div>1 SCI-P</div></div></div></div></div></div></div>		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6041	Head	3.4.1.2 SCI-P - Information Flows		
Eu.P.3064	Info	The generic commands and messages through the SCI_P_Subsystem_EIL are specified in [Eu.Doc.119].		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6042	Def	<div><div>[Package] SCI-P - Information Flows [Interface Requirements - Direction of Information Objects]</div><div><div>bdd [Package] SCI-P - Information Flows [Interface Requirements - Direction of Information Objects]</div><div><div><div><div>«information flow» SCI_P_Subsystem_EIL</div><div>proxyPorts «ProxyPort» P1inout : SCI_GEN «ProxyPort» P3out : SCI_P_1 «ProxyPort» P9in : SCI_P_2</div></div><div><div>«information flow» SCI_P_Subsystem_P</div><div>proxyPorts «ProxyPort» P1inout : SCI_GEN «ProxyPort» P3in : SCI_P_1 «ProxyPort» P9out : SCI_P_2</div></div></div><div><div><div>«information flow» SCI_P_1</div><div>prov «signal» Cd_Move_Point</div></div><div><div>«information flow» SCI_P_2</div><div>reqd «signal» Msg_Point_Position reqd «signal» Msg_Ability_To_Move_Point «signal» Msg_Movement_Failed</div></div></div><div><div>«information flow» SCI_GEN</div><div>prov «signal» : Cd_PDI_Version_Check reqd «signal» : Msg_PDI_Version_Check prov «signal» : Cd_Close_PDI prov «signal» : Cd_Initialisation_Request reqd «signal» : Msg_Start_Initialisation reqd «signal» : Msg_Initialisation_Completed prov «signal» : Cd_Release_PDI_for_Maintenance reqd «signal» : Msg_PDI_Available reqd «signal» : Msg_PDI_Not_Available reqd «signal» : Msg_Reset_PDI</div></div></div></div></div>		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6044	Def	<div><div>[Package] SCI-P - Information Flows [Interface Requirements - Information Objects]</div><div><div>bdd [Package] SCI-P - Information Flows [Interface Requirements - Information Objects]</div><div><div><div><div>«information object» signal Cd_Move_Point</div><div>CommandedPointPositionState : PointPositionControlableState</div></div><div><div>«information object» signal Msg_Point_Position</div><div>ReportedPointPositionState : PointPositionState ReportedDegradedPointPosition : PointPositionDegradedState</div></div><div><div>«information object» signal Msg_Movement_Failed</div></div><div><div>«information object» signal Msg_Ability_To_Move_Point</div><div>ReportedAbilityToMoveState : AbilityToMoveState</div></div></div><div><div><div>«valueType (enumeration)» PointPositionControlableState</div><div>Left Right</div></div><div><div>«valueType (enumeration)» PointPositionState</div><div>Left_ Right_ NoEndPosition_ UnintendedPosition...</div></div><div><div>«valueType (enumeration)» PointPositionDegradedState</div><div>DegradedLeft_ DegradedRight_ NotDegraded_ NotApplicable...</div></div><div><div>«valueType (enumeration)» AbilityToMoveState</div><div>AbleToMove_ UnableToMove...</div></div></div></div></div><div><div>CommandedPointPositionState</div><div>ReportedPointPositionState</div><div>ReportedDegradedPointPosition</div><div>ReportedAbilityToMoveState</div></div></div>		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6183	Def	Cd_Move_Point	Command (Cd) from Subsystem - Electronic Interlocking to Subsystem - Point to move the Point into the commanded position.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6185	Def	Msg_Ability_To_Move_Point	Message (Msg) from Subsystem - Point to Subsystem - Electronic Interlocking about the current Ability to move point.	Option Able to move
Eu.P.6187	Def	Msg_Point_Position	Message (Msg) from Subsystem - Point to Subsystem - Electronic Interlocking about the current Point position.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6190	Def	Msg_Movement_Failed	<div>Message (Msg) from Subsystem - Point to Subsystem - Electronic Interlocking that the ongoing moving failed. The Subsystem - Point has abandoned the moving for one of the following reasons: - The time period Con_tmax_Point_Operation has expired. - The point movement has failed.</div> <div>Note: The conditions defining a failed moment can be supplier and IM specific. They may include, for example, a failure of the star point buildup or a situation in which only non-driven point machines have not yet reached the commanded end position.</div>	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6045	Head	3.4.1.3 SCI-P - Functional Viewpoint		
Eu.P.6280	Head	3.4.1.3.1 SCI-P - Functional Partitioning		

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6155	Def	<div><div>[Package] SCI-P - Functional Partitioning [Functional Viewpoint - Interface Requirements]</div><div><div>bdd [Package] SCI-P - Functional Partitioning [Functional Viewpoint - Interface Requirements]</div><div><div><div><div><div>«logical structural entity» SCI-P</div><div>Subsystem - Electronic Interlocking</div><div><div>«logical structural entity» Subsystem - Electronic Interlocking</div><div>1</div><div>1</div><div>1</div></div><div>Subsystem - Point</div><div><div>«logical structural entity» Subsystem - Point</div><div>1</div><div>1</div><div>1</div></div></div><div>SCI-P</div><div>SCI-P</div></div><div><div>Generic requirements for subsystems</div><div><div>«functional entity» S_SCI_EfeS_Prim</div><div>1</div><div>«functional entity» F_SCI_EfeS_Sec</div><div>1</div></div><div><div>SCI-P - Functional Viewpoint</div><div><div>«functional entity» S_SCI_P_Command</div><div>1</div><div>«functional entity» F_SCI_P_Report</div><div>1</div></div><div><div>«functional entity» S_SCI_P_Receive</div><div>1</div><div>«functional entity» F_SCI_P_Receive</div><div>1</div></div></div></div></div></div></div></div>		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6244	Head	3.4.1.3.2 SCI-P - Functional Architecture		
Eu.P.3279	Info	SCI-P		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.3286	Def	<div><div>[Block] SCI-P - [Functional Viewpoint - Interface Requirements - Functional Architecture]</div><div><div>ibd [Block] SCI-P - [Functional Viewpoint - Interface Requirements - Functional Architecture]</div><div><div><div>«logical structural entity» SCI-P</div><div><div><div><div><div>«participant» {end = SCI-P} «logical structural entity» InLink : Subsystem - Electronic Interlocking</div><div><div>«functional entity» : S_SCI_EfeS_Prim P1inout : ~SCI_GEN</div><div>«functional entity» : S_SCI_P_Receive P9in : ~SCI_P_2</div><div>«functional entity» : S_SCI_P_Command P3out : ~SCI_P_1</div></div><div><div>«proxy» «logical port» SCI_P : SCI_P_Subsystem_EIL</div><div>«equal»</div><div>«proxy» «logical port» SCI-P : SCI_P_Subsystem_P</div><div>«equal»</div><div>«proxy» «logical port» SCI-P : SCI_P_Subsystem_P</div><div>«equal»</div><div>«proxy» «logical port» SCI-P : SCI_P_Subsystem_P</div><div>«equal»</div><div>«proxy» «logical port» SCI-P : SCI_P_Subsystem_P</div></div><div><div>«functional entity» : F_SCI_EfeS_Sec P1inout : SCI_GEN : F_SCI_EfeS_Sec p3inout : ~F_SCI_Specific d50out_PDI_Connection_State : String</div><div>«functional entity» : F_SCI_P_Report p3inout : F_SCI_Specific d50in_PDI_Connection_State : String</div><div>«functional entity» : F_SCI_P_Receive P3in : SCI_P_1 d50in_PDI_Connection_State : String</div></div></div></div></div></div></div></div></div></div>		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6245	Head	3.4.1.3.3 SCI-P - Functional Entities		
Eu.P.6412	Info	F_SCI_P_Receive		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6413	Req	<div>[Block] F_SCI_P_Receive [Functional Viewpoint - Interface Requirements - Functional Entity] <div>ibdd [Block] F_SCI_P_Receive [Functional Viewpoint - Interface Requirements - Functional Entity]<div><div>«functional entity» F_SCI_P_Receive</div><div><div>P3in : SCI_P_1</div><div>d10out_Required_Point_Position : String</div><div>d11in_Observed_Ability_To_Move : String</div><div>d12in_Required_PM_Position : String</div><div>d50in_PDI_Connection_State : String</div></div></div></div></div>		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6415	Def	d10out_Required_Point_Position	The port d10out_Required_Point_Positiont contains the Point Target Position. The following values are valid: "LEFT", "RIGHT".	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6429	Def	d50in_PDI_Connection_State		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6430	Def	d12in_Required_PM_Position		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7018	Def	d11in_Observed_Ability_To_Move		Option Able to move
Eu.P.6470	Def	P3in	The port P3in exchanges information objects according to SCI_P_1.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6431	Info	F_SCI_P_Receive - Behaviour		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6432	Req	<div>Functional Viewpoint - Interface Requirements - Functional Entity STD 1 stm [State Machine] F_SCI_P_Receive - Behaviour [Functional Viewpoint - Interface Requirements - Functional Entity STD 1]<div><div>Initial0</div><div>when(d50in_PDI_Connection_State = "NOT_READY_FOR_PDI_NO_SCP" OR d50in_PDI_Connection_State = "NOT_READY_FOR_PDI" OR d50in_PDI_Connection_State = "SUSPENDED") /</div><div>RECEIVING_COMMANDS</div><div>Entry/d10out_Required_Point_Position := "UNCOMMANDED"; Cd_Move_Point[CommandedPointPositionState = Left AND d50in_PDI_Connection_State = "ESTABLISHED"]/d10out_Required_Point_Position := "LEFT"; Cd_Move_Point[CommandedPointPositionState = Right AND d50in_PDI_Connection_State = "ESTABLISHED"]/d10out_Required_Point_Position := "RIGHT";</div><div>when(d12in_Required_PM_Position = "UNCOMMANDED") /</div><div>when(d11in_Observed_Ability_To_Move = "ABLE_TO_MOVE") /</div></div></div>	This state machine diagram describes the requirements for the following functionalities: - receives the commanded point position state and forwards it to the internal logic of the Subsystem - Point	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6433	Def	Initial0		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6434	Def	{/Initial0 - RECEIVING_COMMANDS}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6435	Def	RECEIVING_COMMANDS		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6469	Def	when(d50in_PDI_Connection_State = "NOT_READY_FOR_PDI_NO_SCP" OR d50in_PDI_Connection_State = "NOT_READY_FOR_PDI" OR d50in_PDI_Connection_State = "SUSPENDED")/{RECEIVING_COMMANDS - RECEIVING_COMMANDS}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7019	Def	Cd_Move_Point[CommandedPointPositionState = Left AND d50in_PDI_Connection_State = "ESTABLISHED"]/d10out_Required_Point_Position := "LEFT";{State-internal in RECEIVING_COMMANDS}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7020	Def	Cd_Move_Point[CommandedPointPositionState = Right AND d50in_PDI_Connection_State = "ESTABLISHED"]/d10out_Required_Point_Position := "RIGHT";{State-internal in RECEIVING_COMMANDS}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7021	Def	entry/d10out_Required_Point_Position := "UNCOMMANDED";{State-internal in RECEIVING_COMMANDS}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7022	Def	when(d11in_Observed_Ability_To_Move = "ABLE_TO_MOVE")/{RECEIVING_COMMANDS - RECEIVING_COMMANDS}		Option Able to move
Eu.P.7023	Def	when(d12in_Required_PM_Position = "UNCOMMANDED")/{RECEIVING_COMMANDS - RECEIVING_COMMANDS}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6522	Info	F_SCI_P_Report		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6523	Req	<div><div>[Block] F_SCI_P_Report [Functional Viewpoint - Interface Requirements - Functional Entity]</div><div><div>ibd [Block] F_SCI_P_Report [Functional Viewpoint - Interface Requirements - Functional Entity]</div><div><div><div>«functional entity» F_SCI_P_Report</div><div>Operation «Operation» cOp1_Point_Position () : PointPositionState «Operation» cOp2_Degraded_Point_Position () : PointPositionDegradedState «Operation» cOp3_Ability_To_Move () : AbilityToMoveState</div><div>values «BlockProperty» Mem_Last_Reported_Degraded_Point_Position : String «BlockProperty» Mem_Last_Reported_Point_Position : String</div><div><div>d11in_Observed_Ability_To_Move : Stringp3inout : F_SCI_Specific</div><div>d13in_Observed_Movement_Failed : BooleanP9out : SCI_P_2</div><div>d14in_Observed_Point_Position : String</div><div>D15in_Con_Observe_Ability_To_Move : Boolean</div><div>d17in_Observed_Degraded_Point_Position : String</div><div>d50in_PDI_Connection_State : String</div></div></div></div></div></div>		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6524	Def	<pre>/* cOp1_Point_Position */ if (d14in_Observed_Point_Position = "LEFT") then return PointPositionState.Left; elseif (d14in_Observed_Point_Position = "RIGHT") then return PointPositionState.Right; elseif (d14in_Observed_Point_Position = "NO_END_POSITION") then return PointPositionState.NoEndPosition; elseif (d14in_Observed_Point_Position = "UNINTENDED_POSITION") then return PointPositionState.UnintendedPosition; end if</pre>	cOp1_Point_Position	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6525	Def	<pre>/* cOp2_Degraded_Point_Position */ if (d17in_Observed_Degraded_Point_Position = "DEGRADED_LEFT") then return PointPositionDegradedState.DegradedLeft; elseif (d17in_Observed_Degraded_Point_Position = "DEGRADED_RIGHT") then return PointPositionDegradedState.DegradedRight;</pre>	cOp2_Degraded_Point_Position	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
		elseif (d17in_Observed_Degraded_Point_Position = "DEGRADED_DENIED") then return PointPositionDegradedState.NotApplicable; else return PointPositionDegradedState.NotDegraded; end if		
Eu.P.6526	Def	/* cOp3_Ability_To_Move */ if (d11in_Observed_Ability_To_Move = "ABLE_TO_MOVE") then return AbilityToMoveState.AbleToMove; elseif (d11in_Observed_Ability_To_Move = "UNABLE_TO_MOVE") then return AbilityToMoveState.UnableToMove; end if	cOp3_Ability_To_Move	Option Able to move
Eu.P.6527	Def	d17in_Observed_Degraded_Point_Position		Basic non-4-wire multiple P Basic 4-wire multiple P
Eu.P.6528	Def	d50in_PDI_Connection_State		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6529	Def	d14in_Observed_Point_Position		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6530	Def	d11in_Observed_Ability_To_Move		Option Able to move
Eu.P.6556	Def	P9out	The port P9out exchanges information objects according to SCI_P_2.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6557	Def	p3inout		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7024	Def	d13in_Observed_Movement_Failed		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7025	Def	D15in_Con_Observe_Ability_To_Move		Option Able to move
Eu.P.6531	Info	F_SCI_P_Report - Behaviour		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6532	Req	<div>Functional Viewpoint - Interface Requirements - Functional Entity STD 2</div> <div>stm [State Machine] F_SCI_P_Report - Behaviour [Functional Viewpoint - Interface Requirements - Functional Entity STD 2]</div> <div><div><div>Initial0</div><div>INTERFACE_CONNECTION_NOT_ESTABLISHED</div><div>Start_Status_Report/send Msg_Point_Position (cOp1_Point_Position,cOp2_Degraded_Point_Position) to P9out; Mem_Last_Reported_Point_Position := d14in_Observed_Point_Position; Mem_Last_Reported_Degraded_Point_Position := d17in_Observed_Degraded_Point_Position;</div><div>[D15in_Con_Observe_Ability_To_Move]/send Msg_Ability_To_Move_Point (cOp3_Ability_To_Move) to P9out;</div><div>[NOT D15in_Con_Observe_Ability_To_Move]</div><div>REPORTING_OVERALL_POINT_STATES</div><div>Entry/send Status_Report_Completed to p3inout;</div><div>REPORTING_ABILITY_TO_MOVE</div><div><div>Initial1</div><div>MSG_ABILITY_TO_MOVE</div><div>when(d11in_Observed_Ability_To_Move = "UNABLE_TO_MOVE") [d50in_PDI_Connection_State = "ESTABLISHED" AND D15in_Con_Observe_Ability_To_Move] / send Msg_Ability_To_Move_Point (UnableToMove) to P9out; when(d11in_Observed_Ability_To_Move = "ABLE_TO_MOVE") [d50in_PDI_Connection_State = "ESTABLISHED" AND D15in_Con_Observe_Ability_To_Move] / send Msg_Ability_To_Move_Point (AbleToMove) to P9out;</div></div><div>REPORTING_MSG_POINT_POSITION</div><div><div>WAITING</div><div>Initial2</div><div>when((d14in_Observed_Point_Position <> Mem_Last_Reported_Point_Position) OR (d17in_Observed_Degraded_Point_Position <> Mem_Last_Reported_Degraded_Point_Position)) [d50in_PDI_Connection_State = "ESTABLISHED"] /</div><div>MSG_OVERALL_POINT_POSITION</div><div>Entry/send Msg_Point_Position (cOp1_Point_Position,cOp2_Degraded_Point_Position) to P9out; Mem_Last_Reported_Point_Position := d14in_Observed_Point_Position; Mem_Last_Reported_Degraded_Point_Position := d17in_Observed_Degraded_Point_Position;</div><div>when((d14in_Observed_Point_Position <> Mem_Last_Reported_Point_Position)) [d50in_PDI_Connection_State = "ESTABLISHED"] /</div><div>when((d17in_Observed_Degraded_Point_Position <> Mem_Last_Reported_Degraded_Point_Position)) [d50in_PDI_Connection_State = "ESTABLISHED"] /</div></div><div>REPORTING_MOVEMENT_FAILED</div><div><div>Initial3</div><div>MSG_MOVEMENT_FAILED</div><div>when(d13in_Observed_Movement_Failed) [d50in_PDI_Connection_State = "ESTABLISHED"] / send Msg_Movement_Failed to P9out;</div></div></div></div>	<div>This state machine diagram describes the requirements for the following functionalities:</div> <div>- receives the observed ability to move status from internal logic and reports this to the Subsystem - Electronic Interlocking</div> <div>- receives the observed Point Position status from internal logic and reports this to the Subsystem - Electronic Interlocking</div> <div>- receives the observed Movement Failed status from internal logic and reports this to the Subsystem - Electronic Interlocking</div>	<div>Basic non-4-wire single P</div> <div>Basic non-4-wire multiple P</div> <div>Basic 4-wire single P</div> <div>Basic 4-wire multiple P</div>
Eu.P.6533	Def	Initial0		<div>Basic non-4-wire single P</div> <div>Basic non-4-wire multiple P</div> <div>Basic 4-wire single P</div> <div>Basic 4-wire multiple P</div>
Eu.P.6534	Def	/ {Initial0 - INTERFACE_CONNECTION_NOT_ESTABLISHED}		<div>Basic non-4-wire single P</div> <div>Basic non-4-wire multiple P</div> <div>Basic 4-wire single P</div> <div>Basic 4-wire multiple P</div>
Eu.P.6535	Def	INTERFACE_CONNECTION_NOT_ESTABLISHED		<div>Basic non-4-wire single P</div> <div>Basic non-4-wire multiple P</div> <div>Basic 4-wire single P</div> <div>Basic 4-wire multiple P</div>
Eu.P.6536	Def	Start_Status_Report/send Msg_Point_Position (cOp1_Point_Position,cOp2_Degraded_Point_Position) to P9out; Mem_Last_Reported_Point_Position := d14in_Observed_Point_Position; Mem_Last_Reported_Degraded_Point_Position := d17in_Observed_Degraded_Point_Position; {INTERFACE_CONNECTION_NOT_ESTABLISHED - Junction}		<div>Basic non-4-wire single P</div> <div>Basic non-4-wire multiple P</div> <div>Basic 4-wire single P</div> <div>Basic 4-wire multiple P</div>
Eu.P.6537	Def	REPORTING_OVERALL_POINT_STATES		<div>Basic non-4-wire single P</div> <div>Basic non-4-wire multiple P</div> <div>Basic 4-wire single P</div> <div>Basic 4-wire multiple P</div>
Eu.P.6538	Def	REPORTING_ABILITY_TO_MOVE		Option Able to move

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.6539	Def	Initial1		Option Able to move
Eu.P.6540	Def	/{Initial1 - MSG_ABILITY_TO_MOVE}		Option Able to move
Eu.P.6541	Def	MSG_ABILITY_TO_MOVE		Option Able to move
Eu.P.6542	Def	when(d11in_Observed_Ability_To_Move = "ABLE_TO_MOVE")[d50in_PDI_Connection_State = "ESTABLISHED" AND D15in_Con_Observe_Ability_To_Move]/ send Msg_Ability_To_Move_Point (AbleToMove) to P9out;{State-internal in MSG_ABILITY_TO_MOVE}		Option Able to move
Eu.P.6543	Def	when(d11in_Observed_Ability_To_Move = "UNABLE_TO_MOVE")[d50in_PDI_Connection_State = "ESTABLISHED" AND D15in_Con_Observe_Ability_To_Move]/ send Msg_Ability_To_Move_Point (UnableToMove) to P9out;{State-internal in MSG_ABILITY_TO_MOVE}		Option Able to move
Eu.P.6544	Def	REPORTING_MSG_POINT_POSITION		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6545	Def	Initial2		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6546	Def	/{Initial2 - WAITING}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6547	Def	MSG_OVERALL_POINT_POSITION		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6548	Def	entry/send Msg_Point_Position (cOp1_Point_Position,cOp2_Degraded_Point_Position) to P9out; Mem_Last_Reported_Point_Position := d14in_Observed_Point_Position; Mem_Last_Reported_Degraded_Point_Position := d17in_Observed_Degraded_Point_Position;{State-internal in MSG_OVERALL_POINT_POSITION}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6549	Def	when((d17in_Observed_Degraded_Point_Position <> Mem_Last_Reported_Degraded_Point_Position))[d50in_PDI_Connection_State = "ESTABLISHED"]/{MSG_OVERALL_POINT_POSITION - MSG_OVERALL_POINT_POSITION}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6550	Def	when((d14in_Observed_Point_Position <> Mem_Last_Reported_Point_Position))[d50in_PDI_Connection_State = "ESTABLISHED"]/{MSG_OVERALL_POINT_POSITION - MSG_OVERALL_POINT_POSITION}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6551	Def	WAITING		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6553	Def	when((d14in_Observed_Point_Position <> Mem_Last_Reported_Point_Position) OR (d17in_Observed_Degraded_Point_Position <> Mem_Last_Reported_Degraded_Point_Position))[d50in_PDI_Connection_State = "ESTABLISHED"]/{WAITING - MSG_OVERALL_POINT_POSITION}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6554	Def	entry/send Status_Report_Completed to p3inout;{State-internal in REPORTING_OVERALL_POINT_STATES}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6555	Def	when(d50in_PDI_Connection_State = "READY_FOR_PDI_NO_SCP" OR d50in_PDI_Connection_State = "READY_FOR_PDI" OR d50in_PDI_Connection_State = "SUSPENDED")/{REPORTING_OVERALL_POINT_STATES - INTERFACE_CONNECTION_NOT_ESTABLISHED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7029	Def	REPORTING_MOVEMENT_FAILED		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7030	Def	Initial3		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7031	Def	/{Initial3 - MSG_MOVEMENT_FAILED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7032	Def	MSG_MOVEMENT_FAILED		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7033	Def	when(d13in_Observed_Movement_Failed)[d50in_PDI_Connection_State = "ESTABLISHED"]/send Msg_Movement_Failed to P9out;{State-internal in MSG_MOVEMENT_FAILED}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7026	Def	Junction		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7027	Def	[NOT D15in_Con_Observe_Ability_To_Move]/{Junction - REPORTING_OVERALL_POINT_STATES}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7028	Def	[D15in_Con_Observe_Ability_To_Move]/send Msg_Ability_To_Move_Point (cOp3_Ability_To_Move) to P9out;{Junction - REPORTING_OVERALL_POINT_STATES}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4704	Info	S_SCI_P_Command		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4711	Req	<div><div>[Block]S_SCI_P_Command_SR [Functional Viewpoint - Interface Requirements - Functional Entity]</div><div><div>ibd [Block]S_SCI_P_Command_SR [Functional Viewpoint - Interface Requirements - Functional Entity]</div><div><div><div>«functional entity» S_SCI_P_Command</div><div><div><div>d2in_Move_Point : String</div><div>t1in_Move_Point : PulsedIn</div><div>d50in_PDI_Connection_State : String</div></div><div>P3out : ~SCI_P_1</div></div></div></div></div></div>		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4707	Def	d2in_Move_Point		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4706	Def	d50in_PDI_Connection_State		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6153	Def	t1in_Move_Point		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6477	Def	P3out	The port P3out exchanges information objects according to SCI_P_1.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4712	Info	S_SCI_P_Command - Behaviour		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4729	Req	<div>Functional Viewpoint - Interface Requirements - Functional Entity STD 3</div> <div><div>stm [State Machine] S_SCI_P_Command - Behaviour [Functional Viewpoint - Interface Requirements - Functional Entity STD 3]</div><div><div><div>Initial0</div><div>SENDING_COMMANDS</div><div>when(t1in_Move_Point) [d2in_Move_Point = "LEFT" AND d50in_PDI_Connection_State = "ESTABLISHED"]/send Cd_Move_Point (Left) to P3out; when(t1in_Move_Point) [d2in_Move_Point = "RIGHT" AND d50in_PDI_Connection_State = "ESTABLISHED"]/send Cd_Move_Point (Right) to P3out;</div></div></div></div>	<div>This state machine diagram describes the requirements for the following functionalities:</div> <div>- receives the Moving Command to be set from internal logic and commands this to the Subsystem - Point</div>	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4713	Def	Initial0		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

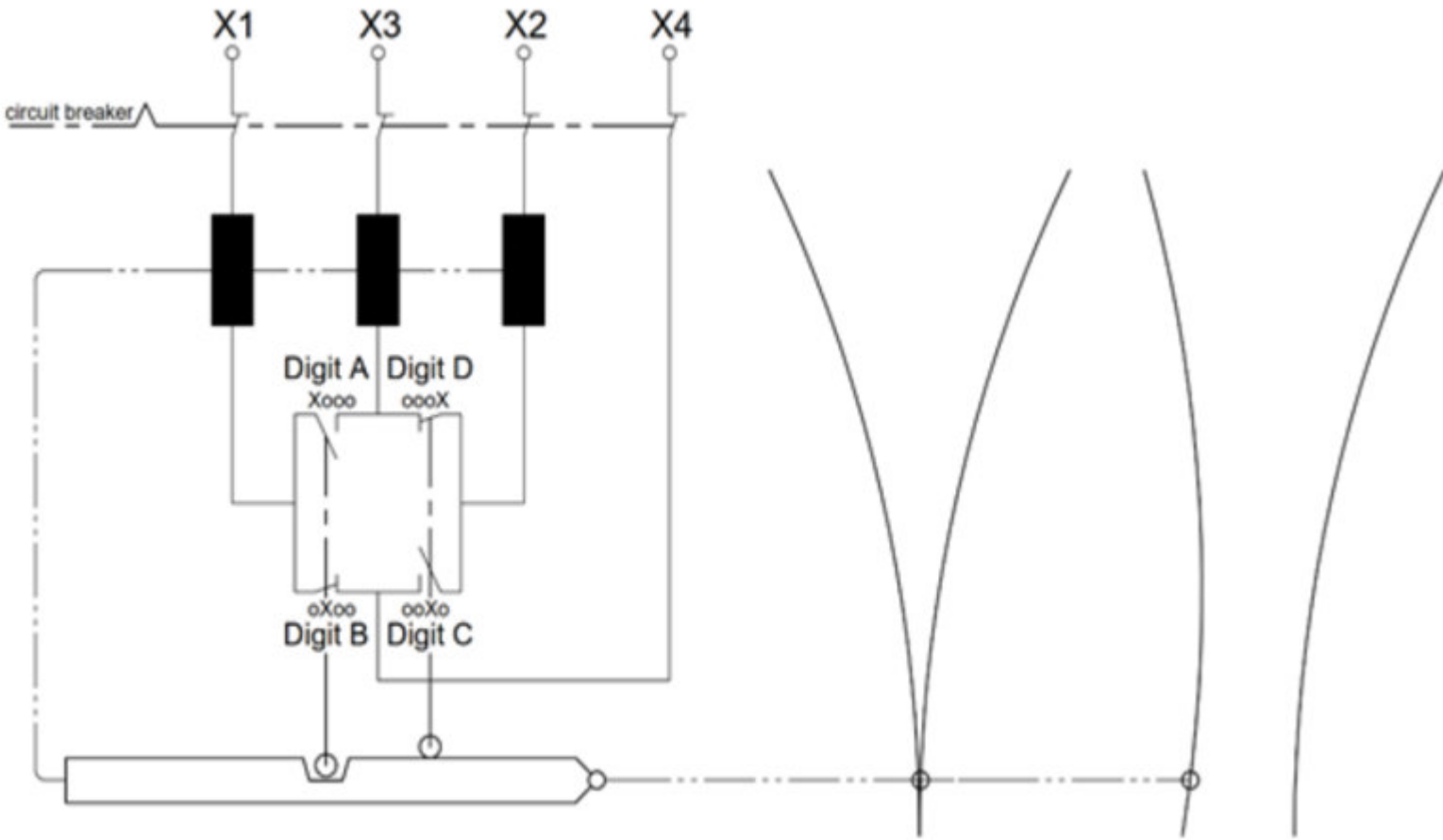
ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.4714	Def	{Initial0 - SENDING_COMMANDS}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6131	Def	SENDING_COMMANDS		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6132	Def	when(t1in_Move_Point)[d2in_Move_Point = "RIGHT" AND d50in_PDI_Connection_State = "ESTABLISHED"]/send Cd_Move_Point (Right)to P3out;{State-internal in SENDING_COMMANDS}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6133	Def	when(t1in_Move_Point)[d2in_Move_Point = "LEFT" AND d50in_PDI_Connection_State = "ESTABLISHED"]/send Cd_Move_Point (Left) to P3out;{State-internal in SENDING_COMMANDS}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7034	Info	S_SCI_P_Receive		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7035	Req	<div>[Block]S_SCI_P_Receive_SR [Functional Viewpoint - Interface Requirements - Functional Entity] <div><div>ibdd [Block]S_SCI_P_Receive_SR [Functional Viewpoint - Interface Requirements - Functional Entity]</div><div><div>«functional entity» S_SCI_P_Receive</div><div><div>P9in : ~SCI_P_2</div><div>t4out_Point_Position : PulsedOut</div><div>d5out_Point_Position : String</div><div>t6out_Movement_Failed : PulsedOut</div><div>t7out_Ability_To_Move : PulsedOut</div><div>d8out_Ability_To_Move : String</div></div></div></div></div>		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7037	Def	d5out_Point_Position		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7038	Def	d8out_Ability_To_Move		Option Able to move
Eu.P.7039	Def	P9in	The port P9in exchanges information objects according to SCI_P_2.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7072	Def	t4out_Point_Position		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7073	Def	t6out_Movement_Failed		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7074	Def	t7out_Ability_To_Move		Option Able to move
Eu.P.7040	Info	S_SCI_P_Receive - Behaviour		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7041	Req	<div>Functional Viewpoint - Interface Requirements - Functional Entity STD 4</div> <div>stm [State Machine] S_SCI_P_Receive - Behaviour [Functional Viewpoint - Interface Requirements - Functional Entity STD 4]</div> <div><div><div>Initial0</div><div>OPERATING</div><div>POINT_POSITION_OBSERVING</div><div><div>Initial1</div><div>RECEIVE_OVERALL_POINT_POSITION_AND_DEGRADED_POINT_POSITION_REPORT</div><div>Msg_Point_Position[ReportedPointPositionState = Left AND ReportedDegradedPointPosition = NotApplicable]/d5out_Point_Position := "LEFT, NOT_APPLICABLE"; t4out_Point_Position := TRUE; Msg_Point_Position[ReportedPointPositionState = Left AND ReportedDegradedPointPosition = NotDegraded]/d5out_Point_Position := "LEFT, NOT_DEGRADED"; t4out_Point_Position := TRUE; Msg_Point_Position[ReportedPointPositionState = Right AND ReportedDegradedPointPosition = NotApplicable]/d5out_Point_Position := "RIGHT, NOT_APPLICABLE"; t4out_Point_Position := TRUE; Msg_Point_Position[ReportedPointPositionState = Right AND ReportedDegradedPointPosition = NotDegraded]/d5out_Point_Position := "RIGHT, NOT_DEGRADED"; t4out_Point_Position := TRUE; Msg_Point_Position[ReportedPointPositionState = NoEndPosition AND ReportedDegradedPointPosition = DegradedLeft]/d5out_Point_Position := "NO_END_POSITION, DEGRADED_LEFT"; t4out_Point_Position := TRUE; Msg_Point_Position[ReportedPointPositionState = NoEndPosition AND ReportedDegradedPointPosition = DegradedRight]/d5out_Point_Position := "NO_END_POSITION, DEGRADED_RIGHT"; t4out_Point_Position := TRUE; Msg_Point_Position[ReportedPointPositionState = NoEndPosition AND ReportedDegradedPointPosition = NotDegraded]/d5out_Point_Position := "NO_END_POSITION, NOT_DEGRADED"; t4out_Point_Position := TRUE; Msg_Point_Position[ReportedPointPositionState = NoEndPosition AND ReportedDegradedPointPosition = NotApplicable]/d5out_Point_Position := "NO_END_POSITION, NOT_APPLICABLE"; t4out_Point_Position := TRUE; Msg_Point_Position[ReportedPointPositionState = UnintendedPosition AND ReportedDegradedPointPosition = DegradedLeft]/d5out_Point_Position := "UNINTENDED_POSITION, DEGRADED_LEFT"; t4out_Point_Position := TRUE; Msg_Point_Position[ReportedPointPositionState = UnintendedPosition AND ReportedDegradedPointPosition = DegradedRight]/d5out_Point_Position := "UNINTENDED_POSITION, DEGRADED_RIGHT"; t4out_Point_Position := TRUE; Msg_Point_Position[ReportedPointPositionState = UnintendedPosition AND ReportedDegradedPointPosition = NotDegraded]/d5out_Point_Position := "UNINTENDED_POSITION, NOT_DEGRADED"; t4out_Point_Position := TRUE; Msg_Point_Position[ReportedPointPositionState = UnintendedPosition AND ReportedDegradedPointPosition = NotApplicable]/d5out_Point_Position := "UNINTENDED_POSITION, NOT_APPLICABLE"; t4out_Point_Position := TRUE;</div></div></div><div><div>ABILITY_TO_MOVE_OBSERVING</div><div><div>Initial2</div><div>RECEIVE_ABILITY_TO_MOVE_REPORT</div><div>Msg_Ability_To_Move_Point[ReportedAbilityToMoveState = AbleToMove]/d8out_Ability_To_Move := "ABLE_TO_MOVE"; t7out_Ability_To_Move := TRUE; Msg_Ability_To_Move_Point[ReportedAbilityToMoveState = UnableToMove]/d8out_Ability_To_Move := "UNABLE_TO_MOVE"; t7out_Ability_To_Move := TRUE;</div></div></div><div><div>MOVEMENT_FAILED_OBSERVING</div><div><div>Initial3</div><div>RECEIVE_MOVEMENT_FAILED_REPORT</div><div>Msg_Movement_Failed/t6out_Movement_Failed := TRUE;</div></div></div></div> <div>This state machine diagram describes the requirements for the following functionalities:</div> <div>- receives the reported point position status and forwards it to the internal logic - receives the reported ability to move status and forwards it to the internal logic - receives the reported movement failed report and forwards it to the internal logic</div> <div>Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P</div>		
Eu.P.7042	Def	Initial0		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7043	Def	/ {Initial0 - OPERATING}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7044	Def	OPERATING		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7045	Def	ABILITY_TO_MOVE_OBSERVING		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7046	Def	Initial2		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7047	Def	/{Initial2 - RECEIVE_ABILITY_TO_MOVE_REPORT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7048	Def	RECEIVE_ABILITY_TO_MOVE_REPORT		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7049	Def	Msg_Ability_To_Move_Point[ReportedAbilityToMoveState = UnableToMove]/d8out_Ability_To_Move := "UNABLE_TO_MOVE"; t7out_Ability_To_Move := TRUE;{State-internal in RECEIVE_ABILITY_TO_MOVE_REPORT}		Option Able to move
Eu.P.7050	Def	Msg_Ability_To_Move_Point[ReportedAbilityToMoveState = AbleToMove]/d8out_Ability_To_Move := "ABLE_TO_MOVE"; t7out_Ability_To_Move := TRUE;{State-internal in RECEIVE_ABILITY_TO_MOVE_REPORT}		Option Able to move
Eu.P.7051	Def	MOVEMENT_FAILED_OBSERVING		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7052	Def	Initial3		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7053	Def	/{Initial3 - RECEIVE_MOVEMENT_FAILED_REPORT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7054	Def	RECEIVE_MOVEMENT_FAILED_REPORT		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7055	Def	Msg_Movement_Failed/t6out_Movement_Failed := TRUE;{State-internal in RECEIVE_MOVEMENT_FAILED_REPORT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7056	Def	POINT_POSITION_OBSERVING		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7057	Def	Initial1		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7058	Def	/{Initial1 - RECEIVE_OVERALL_POINT_POSITION_AND_DEGRADED_POINT_POSITION_REPORT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7059	Def	RECEIVE_OVERALL_POINT_POSITION_AND_DEGRADED_POINT_POSITION_REPORT		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7060	Def	Msg_Point_Position[ReportedPointPositionState = Left AND ReportedDegradedPointPosition = NotApplicable]/d5out_Point_Position := "LEFT, NOT_APPLICABLE"; t4out_Point_Position := TRUE;{State-internal in RECEIVE_OVERALL_POINT_POSITION_AND_DEGRADED_POINT_POSITION_REPORT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7061	Def	Msg_Point_Position[ReportedPointPositionState = UnintendedPosition AND ReportedDegradedPointPosition = DegradedRight]/d5out_Point_Position := "UNINTENDED_POSITION, DEGRADED_RIGHT"; t4out_Point_Position := TRUE;{State-internal in RECEIVE_OVERALL_POINT_POSITION_AND_DEGRADED_POINT_POSITION_REPORT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7062	Def	Msg_Point_Position[ReportedPointPositionState = UnintendedPosition AND ReportedDegradedPointPosition = NotDegraded]/d5out_Point_Position := "UNINTENDED_POSITION, NOT_DEGRADED"; t4out_Point_Position := TRUE;{State-internal in RECEIVE_OVERALL_POINT_POSITION_AND_DEGRADED_POINT_POSITION_REPORT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7063	Def	Msg_Point_Position[ReportedPointPositionState = UnintendedPosition AND ReportedDegradedPointPosition = NotApplicable]/d5out_Point_Position := "UNINTENDED_POSITION, NOT_APPLICABLE"; t4out_Point_Position := TRUE;{State-internal in RECEIVE_OVERALL_POINT_POSITION_AND_DEGRADED_POINT_POSITION_REPORT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7064	Def	Msg_Point_Position[ReportedPointPositionState = Left AND ReportedDegradedPointPosition = NotDegraded]/d5out_Point_Position := "LEFT, NOT_DEGRADED"; t4out_Point_Position := TRUE;{State-internal in RECEIVE_OVERALL_POINT_POSITION_AND_DEGRADED_POINT_POSITION_REPORT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7065	Def	Msg_Point_Position[ReportedPointPositionState = Right AND ReportedDegradedPointPosition = NotApplicable]/d5out_Point_Position := "RIGHT, NOT_APPLICABLE"; t4out_Point_Position := TRUE;{State-internal in RECEIVE_OVERALL_POINT_POSITION_AND_DEGRADED_POINT_POSITION_REPORT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7066	Def	Msg_Point_Position[ReportedPointPositionState = Right AND ReportedDegradedPointPosition = NotDegraded]/d5out_Point_Position := "RIGHT, NOT_DEGRADED"; t4out_Point_Position := TRUE;{State-internal in RECEIVE_OVERALL_POINT_POSITION_AND_DEGRADED_POINT_POSITION_REPORT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7067	Def	Msg_Point_Position[ReportedPointPositionState = NoEndPosition AND ReportedDegradedPointPosition = DegradedLeft]/d5out_Point_Position := "NO_END_POSITION, DEGRADED_LEFT"; t4out_Point_Position := TRUE;{State-internal in RECEIVE_OVERALL_POINT_POSITION_AND_DEGRADED_POINT_POSITION_REPORT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7068	Def	Msg_Point_Position[ReportedPointPositionState = NoEndPosition AND ReportedDegradedPointPosition = DegradedRight]/d5out_Point_Position := "NO_END_POSITION, DEGRADED_RIGHT"; t4out_Point_Position := TRUE;{State-internal in RECEIVE_OVERALL_POINT_POSITION_AND_DEGRADED_POINT_POSITION_REPORT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7069	Def	Msg_Point_Position[ReportedPointPositionState = NoEndPosition AND ReportedDegradedPointPosition = NotDegraded]/d5out_Point_Position := "NO_END_POSITION, NOT_DEGRADED"; t4out_Point_Position := TRUE;{State-internal in RECEIVE_OVERALL_POINT_POSITION_AND_DEGRADED_POINT_POSITION_REPORT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7070	Def	Msg_Point_Position[ReportedPointPositionState = NoEndPosition AND ReportedDegradedPointPosition = NotApplicable]/d5out_Point_Position := "NO_END_POSITION, NOT_APPLICABLE"; t4out_Point_Position := TRUE;{State-internal in RECEIVE_OVERALL_POINT_POSITION_AND_DEGRADED_POINT_POSITION_REPORT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7071	Def	Msg_Point_Position[ReportedPointPositionState = UnintendedPosition AND ReportedDegradedPointPosition = DegradedLeft]/d5out_Point_Position := "UNINTENDED_POSITION, DEGRADED_LEFT"; t4out_Point_Position := TRUE;{State-internal in RECEIVE_OVERALL_POINT_POSITION_AND_DEGRADED_POINT_POSITION_REPORT}		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.929	Head	3.4.2 SMI-P (Subsystem - Maintenance and Data Management)		
Eu.P.3066	Info	The generic FlowSpecification and the related FlowProperties through SMI-P are specified in [Eu.Doc.120].		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.925	Head	3.4.3 SDI-P (Subsystem - Maintenance and Data Management)		
Eu.P.3065	Info	The generic data points through the SDI-P are specified in [Eu.Doc.94]. The specific data points through the SDI-P are specified in [Eu.Doc.80].		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6156	Head	3.4.4 SSI-P (Subsystem - Security Services Platform)		
Eu.P.6157	Info	The generic content through SSI-P is specified in [Eu.Doc.117]. Note: In future phases, the EULYNX security specifications will be replaced by harmonised specifications published by the EU-Rail System Pillar Cyber Security domain.		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.910	Head	3.4.5 P4 (Basic Data Identifier)		
Eu.P.3063	Info	The generic FlowSpecification and the related FlowProperties through P4 are specified in [Eu.Doc.20].		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.890	Head	3.4.6 P1 (Maintainer)		
Eu.P.3173	Info	The generic FlowProperties through P1 are specified in [Eu.Doc.20].		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6206	Info	The defined FlowProperties through P1 are mandatory only when the physical interfaces related to the specific maintainer information are available on the Subsystem – Point. Example: The FlowProperty Point_Unintended_Position is only mandatory when the Subsystem – Point is equipped with a point machine interface that can detect unintended position.		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.891	Info	Maintainer	Definition of the InformationFlow (by FlowSpecification) for Maintenance/Operation/Display P1 (Maintainer).	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.896	Def	Point_Moving	Displays the moving of the point at the local status display.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.1377	Def	End_Position_R	Displays the status of the detection of point end position on the right hand.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.894	Def	End_Position_L	Displays the status of the detection of point end position on the left hand.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.3037	Def	Point_Unintended_Position	Displays the unintended position of the point at the local status display (in an unintended position or not).	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4949	Def	P3_Moving_N	Displays the state for each respective physical Point Machine Output (N > 0). (Moving, Stopped).	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4950	Def	P3_Status_N	Displays the state for each respective physical Point Machine Input (N > 0). (End_Position, No_End_Position, Unintended_Position).	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.902	Head	3.4.7 P3 (Point machine)		
Eu.P.903	Info	Point_machine	Definition of the InformationFlow (by FlowSpecification) for the Control Interfaces P3 (Point machine). Note: The behaviour of the interfaces P3 is described generically. The Subsystem - Point needs to be able to write and to read the generic information objects of the statuses from the Point machine.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.905	Def	Information_No_End_Position	Information object from Point machine to Subsystem - Point that the Point machine has No end position. Note: This is expressed as a 4-wire pattern in case the Point machine is implemented with a 4-wire interface. The interpretation of the 4-wire patterns depends on the driving/detection state and the last commanded position. The interpretation tables can be found in Eu.P.6797.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.906	Def	Information_Unintended_Position	Information object from Point machine to Subsystem - Point that the Point machine has an Unintended position position. Note 1: This is expressed as a 4-wire pattern in case the Point machine is implemented with a 4-wire interface. The interpretation of the 4-wire patterns depends on the driving/detection state and the last commanded position. The interpretation tables can be found in Eu.P.6797. Note 2: This information may not be available in case the Point machine is implemented with a non-4-wire interface.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.907	Def	Moving	Information object from Subsystem - Point to Point machine to move the Point machine to an end position (left hand position or right hand position).	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.909	Def	Stop_Moving	Information object from Subsystem - Point to Point machine to stop Moving the Point machine.	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.5364	Def	Information_Ability_To_Move_Point	Information object from Point machine to Subsystem - Point that the Point machine has Drive voltage (able) or has no Drive voltage (unable). Note: The reason for reporting ability to move depends on national specification and also Drive voltage levels monitored shall be defined by national specifications. In future phases of the System Pillar, national specifications will be replaced by harmonised specifications.	Option Able to move
Eu.P.6274	Def	Detection_Voltage	Information object from Subsystem - Point to Point machine to switch the Point machine to Detection Mode (on or off). Note: This information is only available in case the Point machine is implemented with a 4-wire interface.	Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.7016	Def	Information_End_Position_Detected	Information object from Point machine to Subsystem - Point that the Point machine has reliably reached an End position (left hand position or right hand position). Note 1: This is expressed as a 4-wire pattern, in case the Point machine is implemented with a 4-wire interface. LEFT: 1010 RIGHT: 0101	Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7017	Def	Information_End_Position_Reached	Information object from Point machine to Subsystem - Point that the Point machine has reached an End position (left hand position or right hand position), but the position has not yet been reliably detected. Note 1: This information is only available in case the Point machine is implemented with a 4-wire interface. This is expressed as a 4-wire pattern. LEFT: 1010 RIGHT: 0101	Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6797	Head	3.4.7.1 Interpretation tables of 4-wire patterns		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6801	Info	For the 4-wire implementation, the input information is represented as 4-wire patterns.		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6806	Info	The 4-wire pattern consists of four digits, each being in a state of „1" or „0". The state of „1" represents a closed contact in the 4-wire circuit while „0" represents an open contact in the 4-wire circuit.		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6805	Info	There are four contact pairs, where each pair is represented by a specific digit in the 4-wire pattern (ABCD): Contact 1+3 -> Digit A Contact 1+4 -> Digit B Contact 2+4 -> Digit C Contact 2+3 -> Digit D		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6807	Info	<p>The figure shows a schematic representation of the 4-wire circuit</p> 		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6804	Req	The interpretation of the 4-wire patterns is state-dependent. There are 5 interpretations states. The interpretation state is determined by two factors. - Whether the Subsystem Point has been commanded to move the point or the movement has stopped (moving to left, moving to right, no movement/detection) - The last commanded position of the point machine (left, right, not commanded)		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6803	Req	The implementation of the Subsystem Point and/or the Point machine shall be such that the last commanded position is always retained, including in case of loss of communication and in case of rebooting. Note: The only case in which the state 'not commanded' occurs is after the initial start-up.		Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6703	Req	Interpretation of 4-wire patterns without command to move the point and last commanded position "Right"		Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1		Requirement Part 2	Func. Pkg.																																		
		<table><tr><th>4-wire pattern</th><th>Interpretation of information on interface P3</th></tr><tr><td>0000</td><td>Information_No_End_Position</td></tr><tr><td>0001</td><td>Information_No_End_Position</td></tr><tr><td>0010</td><td>Supplier specific</td></tr><tr><td>0011</td><td>Supplier specific</td></tr><tr><td>0100</td><td>Information_No_End_Position</td></tr><tr><td>0101</td><td>Information_End_Position_Detected (Right)</td></tr><tr><td>0110</td><td>Supplier specific</td></tr><tr><td>0111</td><td>Supplier specific (not Information_No_End_Position)</td></tr><tr><td>1000</td><td>Supplier specific</td></tr><tr><td>1001</td><td>Information_Unintended_Position</td></tr><tr><td>1010</td><td>Information_Unintended_Position</td></tr><tr><td>1011</td><td>Supplier specific (not Information_No_End_Position)</td></tr><tr><td>1100</td><td>Supplier specific</td></tr><tr><td>1101</td><td>Supplier specific (not Information_No_End_Position)</td></tr><tr><td>1110</td><td>Supplier specific (not Information_No_End_Position)</td></tr><tr><td>1111</td><td>Supplier specific (not Information_No_End_Position)</td></tr></table>		4-wire pattern	Interpretation of information on interface P3	0000	Information_No_End_Position	0001	Information_No_End_Position	0010	Supplier specific	0011	Supplier specific	0100	Information_No_End_Position	0101	Information_End_Position_Detected (Right)	0110	Supplier specific	0111	Supplier specific (not Information_No_End_Position)	1000	Supplier specific	1001	Information_Unintended_Position	1010	Information_Unintended_Position	1011	Supplier specific (not Information_No_End_Position)	1100	Supplier specific	1101	Supplier specific (not Information_No_End_Position)	1110	Supplier specific (not Information_No_End_Position)	1111	Supplier specific (not Information_No_End_Position)		
4-wire pattern	Interpretation of information on interface P3																																						
0000	Information_No_End_Position																																						
0001	Information_No_End_Position																																						
0010	Supplier specific																																						
0011	Supplier specific																																						
0100	Information_No_End_Position																																						
0101	Information_End_Position_Detected (Right)																																						
0110	Supplier specific																																						
0111	Supplier specific (not Information_No_End_Position)																																						
1000	Supplier specific																																						
1001	Information_Unintended_Position																																						
1010	Information_Unintended_Position																																						
1011	Supplier specific (not Information_No_End_Position)																																						
1100	Supplier specific																																						
1101	Supplier specific (not Information_No_End_Position)																																						
1110	Supplier specific (not Information_No_End_Position)																																						
1111	Supplier specific (not Information_No_End_Position)																																						
Eu.P.6704	Req	Interpretation of 4-wire patterns without command to move the point and last commanded position "Left" <table><tr><th>4-wire pattern</th><th>Interpretation of information on interface P3</th></tr><tr><td>0000</td><td>Information_No_End_Position</td></tr><tr><td>0001</td><td>Supplier specific</td></tr><tr><td>0010</td><td>Information_No_End_Position</td></tr><tr><td>0011</td><td>Supplier specific</td></tr><tr><td>0100</td><td>Supplier specific</td></tr><tr><td>0101</td><td>Information_Unintended_Position</td></tr><tr><td>0110</td><td>Supplier specific</td></tr><tr><td>0111</td><td>Supplier specific (not Information_No_End_Position)</td></tr><tr><td>1000</td><td>Information_No_End_Position</td></tr><tr><td>1001</td><td>Information_Unintended_Position</td></tr><tr><td>1010</td><td>Information_End_Position_Detected (Left)</td></tr><tr><td>1011</td><td>Supplier specific (not Information_No_End_Position)</td></tr><tr><td>1100</td><td>Supplier specific</td></tr><tr><td>1101</td><td>Supplier specific (not Information_No_End_Position)</td></tr><tr><td>1110</td><td>Supplier specific (not Information_No_End_Position)</td></tr><tr><td>1111</td><td>Supplier specific (not Information_No_End_Position)</td></tr></table>		4-wire pattern	Interpretation of information on interface P3	0000	Information_No_End_Position	0001	Supplier specific	0010	Information_No_End_Position	0011	Supplier specific	0100	Supplier specific	0101	Information_Unintended_Position	0110	Supplier specific	0111	Supplier specific (not Information_No_End_Position)	1000	Information_No_End_Position	1001	Information_Unintended_Position	1010	Information_End_Position_Detected (Left)	1011	Supplier specific (not Information_No_End_Position)	1100	Supplier specific	1101	Supplier specific (not Information_No_End_Position)	1110	Supplier specific (not Information_No_End_Position)	1111	Supplier specific (not Information_No_End_Position)		Basic 4-wire single P Basic 4-wire multiple P
4-wire pattern	Interpretation of information on interface P3																																						
0000	Information_No_End_Position																																						
0001	Supplier specific																																						
0010	Information_No_End_Position																																						
0011	Supplier specific																																						
0100	Supplier specific																																						
0101	Information_Unintended_Position																																						
0110	Supplier specific																																						
0111	Supplier specific (not Information_No_End_Position)																																						
1000	Information_No_End_Position																																						
1001	Information_Unintended_Position																																						
1010	Information_End_Position_Detected (Left)																																						
1011	Supplier specific (not Information_No_End_Position)																																						
1100	Supplier specific																																						
1101	Supplier specific (not Information_No_End_Position)																																						
1110	Supplier specific (not Information_No_End_Position)																																						
1111	Supplier specific (not Information_No_End_Position)																																						
Eu.P.6798	Req	Interpretation of 4-wire patterns without command to move the point and no last commanded position <table><tr><th>4-wire pattern</th><th>Interpretation of information on interface P3</th></tr><tr><td>0000</td><td>Information_No_End_Position</td></tr><tr><td>0001</td><td>Information_No_End_Position</td></tr><tr><td>0010</td><td>Information_No_End_Position</td></tr><tr><td>0011</td><td>Supplier specific</td></tr><tr><td>0100</td><td>Information_No_End_Position</td></tr><tr><td>0101</td><td>Information_End_Position_Detected (Right)</td></tr><tr><td>0110</td><td>Supplier specific</td></tr><tr><td>0111</td><td>Supplier specific</td></tr><tr><td>1000</td><td>Information_No_End_Position</td></tr><tr><td>1001</td><td>Information_Unintended_Position</td></tr><tr><td>1010</td><td>Information_End_Position_Detected (Left)</td></tr><tr><td>1011</td><td>Supplier specific</td></tr><tr><td>1100</td><td>Supplier specific</td></tr><tr><td>1101</td><td>Supplier specific</td></tr><tr><td>1110</td><td>Supplier specific</td></tr><tr><td>1111</td><td>Supplier specific</td></tr></table>		4-wire pattern	Interpretation of information on interface P3	0000	Information_No_End_Position	0001	Information_No_End_Position	0010	Information_No_End_Position	0011	Supplier specific	0100	Information_No_End_Position	0101	Information_End_Position_Detected (Right)	0110	Supplier specific	0111	Supplier specific	1000	Information_No_End_Position	1001	Information_Unintended_Position	1010	Information_End_Position_Detected (Left)	1011	Supplier specific	1100	Supplier specific	1101	Supplier specific	1110	Supplier specific	1111	Supplier specific		Basic 4-wire single P Basic 4-wire multiple P
4-wire pattern	Interpretation of information on interface P3																																						
0000	Information_No_End_Position																																						
0001	Information_No_End_Position																																						
0010	Information_No_End_Position																																						
0011	Supplier specific																																						
0100	Information_No_End_Position																																						
0101	Information_End_Position_Detected (Right)																																						
0110	Supplier specific																																						
0111	Supplier specific																																						
1000	Information_No_End_Position																																						
1001	Information_Unintended_Position																																						
1010	Information_End_Position_Detected (Left)																																						
1011	Supplier specific																																						
1100	Supplier specific																																						
1101	Supplier specific																																						
1110	Supplier specific																																						
1111	Supplier specific																																						
Eu.P.6799	Req	Interpretation of 4-wire patterns when commanded to move the point to "Right", until the Subsystem Point has stopped all point machines with drive capability. <table><tr><th>4-wire pattern</th><th>Interpretation of information on interface P3</th></tr><tr><td>0101</td><td>Information_End_Position_Reached (Right)</td></tr><tr><td>All other</td><td>Information_No_End_Position</td></tr></table>		4-wire pattern	Interpretation of information on interface P3	0101	Information_End_Position_Reached (Right)	All other	Information_No_End_Position		Basic 4-wire single P Basic 4-wire multiple P																												
4-wire pattern	Interpretation of information on interface P3																																						
0101	Information_End_Position_Reached (Right)																																						
All other	Information_No_End_Position																																						
Eu.P.6800	Req	Interpretation of 4-wire patterns when commanded to move the point to "Left", until the Subsystem Point has stopped all point machines with drive capability. <table><tr><th>4-wire pattern</th><th>Interpretation of information on interface P3</th></tr><tr><td>1010</td><td>Information_End_Position_Reached (Left)</td></tr><tr><td>All other</td><td>Information_No_End_Position</td></tr></table>		4-wire pattern	Interpretation of information on interface P3	1010	Information_End_Position_Reached (Left)	All other	Information_No_End_Position		Basic 4-wire single P Basic 4-wire multiple P																												
4-wire pattern	Interpretation of information on interface P3																																						
1010	Information_End_Position_Reached (Left)																																						
All other	Information_No_End_Position																																						
Eu.P.6802	Req	For certain 4-wire patterns, a supplier specific interpretation is allowed. When one of these patterns is detected on the 4-wire interface to the Point Machine, the Subsystem Point may react to it in three ways: - interpret the abstract state 'Information_No_End_Position'. - interpret the abstract state 'Information_Unintended_Position'			Basic 4-wire single P Basic 4-wire multiple P																																		

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
		<p>- detect a fatal error (T5in_SIL_Not_Fulfilled in generic EfeS state machine)</p> <p>The supplier may choose which of the 3 system reactions is most appropriate. The reaction may also be time dependent. For the 4-wire patterns 0111, 1011, 1101, 1110 and 1111, it is not allowed to have a persistent interpretation as the abstract state 'Information_No_End_Position' if the commanded position is 'Right' or 'Left'.</p>		
Eu.P.233	Head	4 RAMSS requirements		
Eu.P.2987	Info	The requirements for reliability, availability, maintainability, safety and security are specified in [Eu.Doc.20]		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.3244	Head	5 Technical Requirements		
Eu.P.3245	Info	The generic technical requirements are specified in [Eu.Doc.20]		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.3246	Head	5.1 Specific technical interface requirements		
Eu.P.3247	Head	5.1.1 Interface to the Point of Service Signalling (PoS-Signalling)		
Eu.P.3248	Req	Via the technical interface PoS-Signalling , the data of the functional interface "SCI-P" shall be exchanged with the Subsystem - Electronic Interlocking as specified in [Eu.Doc.92].		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.3249	Req	Via the technical interface PoS-Signalling , the data of the functional interface "SMI-P" shall be exchanged with the Subsystem - Maintenance and Data Management as specified in [Eu.Doc.76].		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.3250	Req	Via the technical interface PoS-Signalling , the data of the functional interface "SDI-P" shall be exchanged with the Subsystem - Maintenance and Data Management as specified in [Eu.Doc.77].		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6207	Req	Via the technical interface PoS-Signalling , the data of the functional interface "SSI-P" shall be exchanged with the Subsystem - Security Services Platform as specified in [Eu.Doc.117]. Note: In future phases, the EULYNX security specifications will be replaced by harmonised specifications published by the EU-Rail System Pillar Cyber Security domain.		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.3251	Head	5.1.2 Interface to the point machine		
Eu.P.6271	Req	The interface to the point machine can be implemented as one of 2 variants: - non-4-wire - 4-wire		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.3252	Req	The technical requirements for both implementation variants shall be defined by national specifications. Note: In future phases of the System Pillar, national specifications will be replaced by harmonised specifications.		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4946	Info	Power property assumptions <ul style="list-style-type: none">• The P3 interface may be implemented with separate power supplies for driving and detection.• The power supply for detection is provided by the Subsystem - Point.• A loss of the power supply for detection will report as "Information_No_End_Position" on P3.• When a crank handle is inserted to the Point Machine, typically its power supply for driving is disconnected.• If the point machine monitors its power supply for driving, a loss of the power supply for driving will report as "Information_Ability_To_Move_Point" on P3. (Note: A point machine may not be implemented with the function to monitor its power supply for driving).		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4941	Info	P3 is defined as a functional interface, physical properties are not currently defined. This specification is based upon the following assumptions on the properties of the P3 interface.		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4942	Req	On a Point machine with detection functionality only, the information objects Moving and Stop_Moving are not available on P3.		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4943	Req	The timing behaviour related to multiple Point Machines for a single Point is supplier specific and is not specified by EULYNX as part of the application layer, this shall be handled by the physical implementation.		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.3253	Head	5.2 Time behaviour		
Eu.P.3254	Req	The time values defined in the chapter Functional requirements specification (Eu.P.2286) shall be configured for the operation of the Subsystem - Point.		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.3262	Head	5.2.1 Response times		
Eu.P.3263	Req	The Subsystem - Point shall send the corresponding message telegram to the Subsystem - Electronic Interlocking within 250 ms after successful change of state, according to specific use cases.		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P

ID	Type	Requirement Part 1	Requirement Part 2	Func. Pkg.
Eu.P.3264	Req	The Subsystem - Point shall start the reversal operation within 500 ms after receiving a command telegram.		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.3265	Req	The Subsystem - Point shall start the redrive operation within 500 ms after detecting No end position.		Option Redrive
Eu.P.3255	Head	5.3 Configuration and engineering data		
Eu.P.3256	Head	5.3.1 Specific data		
Eu.P.3257	Req	The engineering and configuration data for the Subsystem - Point shall include as a minimum the following information:		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.3258	Req	<ul style="list-style-type: none">The applicable time values defined in chapter Definition of time values (Eu.P.2286).		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4947	Req	<ul style="list-style-type: none">The number of implemented Point Machines		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4948	Req	<ul style="list-style-type: none">For each configured Point Machine, whether it is implemented with full functionality or as point detector only. The 1st PM must be configured with drive capability.		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6293	Req	<ul style="list-style-type: none">Whether the Subsystem - Point is configured to perform redrive		Option Redrive
Eu.P.6294	Req	<ul style="list-style-type: none">Whether the Subsystem - Point is configured to detect and report unintended position		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6295	Req	<ul style="list-style-type: none">For each configured Point Machine, whether it is crucial or non-crucial		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.6636	Req	<ul style="list-style-type: none">The implementation variant of the interface to the point machine (non-4-wire/4-wire)		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7108	Req	<ul style="list-style-type: none">Whether the Subsystem - Point is configured to detect and report Ability to Move		Option Able to move
Eu.P.7109	Req	<ul style="list-style-type: none">Whether the Subsystem - Point is configured to stop driving Point Machines individually or according to Common Drive.		Option Common Drive
Eu.P.3259	Info	Two different data sections can be loaded which are the safety-relevant data and the non safety-relevant data. The following definitions apply to the assignment of the sections:		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.3260	Req	<ul style="list-style-type: none">configuration data, such as the IP addresses of the Subsystem - Electronic Interlocking , the value of the diagnostic data points with attribute type 'configuration', is not safety-relevant. This data shall be used to calculate the CSNS.		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.3261	Req	<ul style="list-style-type: none">The remaining configuration data is currently categorised as safety-relevant. This data shall be used to calculate the CSS.		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.4546	Req	<ul style="list-style-type: none">The engineering data is safety-relevant. This data shall be used to calculate the CSS.		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P
Eu.P.7244	Head	5.3.2 Value configuration		
Eu.P.7245	Req	Con_tmax_Point_Operation The time value shall be configured in accordance with: Configurable resolution: steps of 100 ms. Configurable range: from 100 ms up to 30 s Con_tmax_Point_Operation is defined in Eu.P.2439.		Basic non-4-wire single P Basic non-4-wire multiple P Basic 4-wire single P Basic 4-wire multiple P