

EM58 PL HS58 PL HM58 PL



ETHERNET 
POWERLINK

This is a phase-out product, it is still in production, but not recommended for new applications. We recommend using the new EXM58-EX058 encoder.

Suitable for the following models:

- EM58, EM58S PL
- EMC58, EMC59, EMC60 PL
- HS58, HS58S PL
- HSC58, HSC59, HSC60 PL
- HM58, HM58S PL
- HMC58, HMC59, HMC60 PL

General Contents

Safety summary	21
Identification	23
Mounting instructions	24
Electrical connections	29
Quick reference	36
POWERLINK interface	61
Default parameters list	104

This publication was produced by Lika Electronic s.r.l. 2023. All rights reserved. Tutti i diritti riservati. Alle Rechte vorbehalten. Todos los derechos reservados. Tous droits réservés.

This document and information contained herein are the property of Lika Electronic s.r.l. and shall not be reproduced in whole or in part without prior written approval of Lika Electronic s.r.l. Translation, reproduction and total or partial modification (photostat copies, film and microfilm included and any other means) are forbidden without written authorisation of Lika Electronic s.r.l.

The information herein is subject to change without notice and should not be construed as a commitment by Lika Electronic s.r.l. Lika Electronic s.r.l. reserves the right to make all modifications at any moments and without forewarning.

This manual is periodically reviewed and revised. As required we suggest checking if a new or updated edition of this document is available at Lika Electronic s.r.l.'s website. Lika Electronic s.r.l. assumes no responsibility for any errors or omissions in this document. Critical evaluation of this manual by the user is welcomed. Your comments assist us in preparation of future documentation, in order to make it as clear and complete as possible. Please send an e-mail to the following address info@lika.it for submitting your comments, suggestions and criticisms.

The logo for Lika Electronic s.r.l. consists of the word "lika" in a bold, lowercase, sans-serif font. The letter "i" has a dot above it. The logo is positioned in the bottom right corner of the page.

General contents

User's guide.....	1
General contents.....	3
Subject index.....	7
Typographic and iconographic conventions.....	10
Preliminary information.....	11
Glossary of POWERLINK terms.....	12
List of abbreviations.....	18
References.....	20
1 Safety summary.....	21
1.1 Safety.....	21
1.2 Electrical safety.....	21
1.3 Mechanical safety.....	22
2 Identification.....	23
3 Mounting instructions.....	24
3.1 Solid shaft encoders.....	24
3.1.1. Customary installation.....	24
3.1.2 Installation using fixing clamps (code LKM-386).....	25
3.1.3 Installation using a mounting bell (code PF4256).....	25
3.2 Hollow shaft encoders.....	26
3.2.1 EMC58, HSC58, HMC58.....	26
3.2.2 EMC59, HSC59, HMC59.....	27
3.2.3 EMC60, HSC60, HMC60.....	28
4 Electrical connections.....	29
4.1 Connection cap.....	29
4.2 PWR Power supply connector (Figure 2).....	30
4.3 P1 Port 1 and P2 Port 2 connectors (Figure 2).....	30
4.4 Network configuration: cables, hubs, switches - Recommendations.....	31
4.5 Ground connection.....	31
4.6 Connection of the shield.....	31
4.7 MAC address and IP address.....	32
4.8 POWERLINK Node ID: DIP A.....	32
4.9 Diagnostic LEDs (Figure 2).....	34
5 Quick reference.....	36
5.1 Before starting.....	36
5.1.1 Network identity.....	36
5.1.2 Network and communication settings.....	36
5.2 Configuring the encoder with Automation Studio V. 4.1 from B&R.....	37
5.3 MAC address.....	37
5.4 Encoder installation under Automation Studio environment.....	38
5.4.1 Description of the XDD file.....	38
5.4.2 Installing the XDD file.....	39
5.4.3 Setting the device node address in the project.....	43
5.4.4 Configuring the CN device.....	44
5.4.5. Downloading the parameters to the encoder.....	47
5.4.6. Download of parameter values at PLC start.....	48
5.4.7 Monitoring the position and speed values.....	49

5.4.8	Setting the cycle time.....	51
5.4.9	Preset setting program.....	52
5.4.10	Parameters storing program.....	55
5.4.11	Default parameters restoring program.....	56
5.4.12	Entering the System Diagnostics Manager (SDM).....	57
5.4.13	Logger Monitor.....	60
6	POWERLINK interface.....	61
6.1	Overview.....	61
6.2	Physical layer.....	62
6.3	Reference model.....	62
6.4	CANopen over Ethernet.....	63
6.5	POWERLINK nodes.....	63
6.5.1	POWERLINK Managing Node (MN).....	63
6.5.2	POWERLINK Controlled Node (CN).....	63
6.6	POWERLINK Basic Frame.....	63
6.7	Message types.....	64
6.7.1	Start of Cycle (SoC).....	64
6.7.2	PollRequest (PReq).....	64
6.7.3	PollResponse (PRes).....	64
6.7.4	Start of Asynchronous (SoA).....	65
6.7.5	Asynchronous Send (Asnd).....	65
6.8	POWERLINK Cycle.....	65
6.8.1	Isochronous Period.....	65
6.8.2	Asynchronous Period.....	65
6.8.3	Idle Period.....	66
6.9	CN Node NMT States.....	66
6.9.1	NMT_CS_NOT_ACTIVE.....	67
6.9.2	NMT_CS_PRE_OPERATIONAL_1.....	67
6.9.3	NMT_CS_PRE_OPERATIONAL_2.....	67
6.9.4	NMT_CS_READY_TO_OPERATE.....	68
6.9.5	NMT_CS_OPERATIONAL.....	68
6.9.6	NMT_CS_STOPPED.....	68
6.9.7	NMT_CS_BASIC_ETHERNET.....	69
6.10	XDD file.....	69
6.11	Communication objects.....	70
6.11.1	NMT Network Management.....	70
6.11.2	PDO objects.....	71
6.11.3	PDO Mapping.....	71
6.11.4	SDO objects.....	72
6.12	Object Dictionary.....	72
6.12.1	Communication Profile Area objects (DS 301).....	74
	1000 NMT_DeviceType_U32.....	74
	1001 ERR_ErrorRegister_U8.....	74
	1006 NMT_CycleLen_U32.....	74
	1008 NMT_ManufactDevName_VS.....	74
	1009 NMT_ManufactHwVers_VS.....	75
	100A NMT_ManufactSwVers_VS.....	75
	1018-00 NMT_IdentityObject_REC.....	75
	1018-01 NMT_IdentityObject_REC.VendorID_U32.....	75
	1018-02 NMT_IdentityObject_REC.ProductCode_U32.....	75
	1018-03 NMT_IdentityObject_REC.RevisionNo_U32.....	75

1018-04 NMT_IdentityObject_REC.SerialNo_U32.....	76
1020-00 CFM_VerifyConfiguration_REC.....	76
1020-01 CFM_VerifyConfiguration_REC.ConfDate_U32.....	76
1020-02 CFM_VerifyConfiguration_REC.ConfTime_U32.....	76
1030-00 NMT_InterfaceGroup_0h_REC.....	76
1030-01 NMT_InterfaceGroup_0h_REC.InterfaceIndex_U16.....	76
1030-02 NMT_InterfaceGroup_0h_REC.InterfaceDescription_VSTR.....	77
1030-03 NMT_InterfaceGroup_0h_REC.InterfaceType_U8.....	77
1030-04 NMT_InterfaceGroup_0h_REC.InterfaceMtu_U16.....	77
1030-05 NMT_InterfaceGroup_0h_REC.InterfacePhysAddress_OSTR.....	77
1030-06 NMT_InterfaceGroup_0h_REC.InterfaceName_VSTR.....	77
1030-07 NMT_InterfaceGroup_0h_REC.InterfaceOperState_U8.....	77
1030-08 NMT_InterfaceGroup_0h_REC.InterfaceAdminState_U8.....	78
1030-09 NMT_InterfaceGroup_0h_REC.Valid_BOOL.....	78
1300 SDO_SequLayerTimeout_U32.....	78
1800-00 PDO_TxCommParam_00h_REC.....	78
1800-01 PDO_TxCommParam_00h_REC.NodeID_U8.....	78
1800-02 PDO_TxCommParam_00h_REC.MappingVersion_U8.....	78
1A00-00 PDO_TxMappParam_00h_AU64.....	79
1A00-01 PDO_TxMappParam_00h_AU64.ObjectMapping.....	79
1A00-02 PDO_TxMappParam_00h_AU64.ObjectMapping.....	79
1C0B-00 DLL_CNLossSoC_REC.....	80
1C0B-01 DLL_CNLossSoC_REC.CumulativeCnt_U32.....	80
1C0B-02 DLL_CNLossSoC_REC.ThresholdCnt_U32.....	80
1C0B-03 DLL_CNLossSoC_REC.Threshold_U32.....	80
1C0FB-00 DLL_CNCRCErrror_REC.....	80
1C0FB-01 DLL_CNCRCErrror_REC.CumulativeCnt_U32.....	81
1C0FB-02 DLL_CNCRCErrror_REC.ThresholdCnt_U32.....	81
1C0FB-03 DLL_CNCRCErrror_REC.Threshold_U32.....	81
1C14 DLL_CNLossOfSocTolerance_U32.....	81
1F81-00 NMT_NodeAssignment_AU32.....	81
1F81-01 NMT_NodeAssignment_AU32.NodeAssignment.....	82
1F82 NMT_FeatureFlags_U32.....	83
1F83 NMT_EPLVersion_U8.....	84
1F8C NMT_CurrNMTState_U8.....	85
1F8D-00 NMT_PResPayloadLimitList_AU16.....	85
1F8D-01 NMT_PResPayloadLimitList_AU16.PResPayloadLimit.....	86
1F93-00 NMT_EPLNodeID_REC.....	86
1F93-01 NMT_EPLNodeID_REC.NodeID_U8.....	86
1F93-02 NMT_EPLNodeID_REC.NodeIDByHW_BOOL.....	86
1F98-00 NMT_CycleTiming_REC.....	86
1F98-01 NMT_CycleTiming_REC.IsochrTxMaxPayload_U16.....	86
1F98-02 NMT_CycleTiming_REC.IsochrRxMaxPayload_U16.....	87
1F98-03 NMT_CycleTiming_REC.PresMaxLatency_U32.....	87
1F98-04 NMT_CycleTiming_REC.PreqActPayloadLimit_U16.....	87
1F98-05 NMT_CycleTiming_REC.PresActPayloadLimit_U16.....	87
1F98-06 NMT_CycleTiming_REC.AsndMaxLatency_U32.....	88
1F98-07 NMT_CycleTiming_REC.MultiplCycleCnt_U8.....	88
1F98-08 NMT_CycleTiming_REC.AsyncMTU_U16.....	88

1F99 NMT_CNBasicEthernetTimeout_U32.....	88
1F9B-00 NMT_MultiplCycleAssign_AU8.....	89
1F9B-01 NMT_MultiplCycleAssign_AU8.CycleNo.....	89
1F9E NMT_ResetCmd_U8.....	89
6.12.2 Manufacturer Specific Profile Area objects.....	90
2002 Speed format.....	90
6.12.3 Standardised Device Profile Area objects (DS 406).....	91
6000 Operating parameters.....	91
Code sequence.....	91
Scaling function.....	91
Restore default parameters.....	92
Store parameters.....	92
6001 Measuring units per revolution.....	93
6002 Total measuring range in measuring units.....	94
6003 Preset value.....	96
6004 Position value.....	97
6030 Speed value.....	97
6500 Operating status.....	98
Code sequence.....	98
Scaling function.....	98
6501 Singleturn resolution.....	98
6502 Number of revolutions.....	99
6503 Alarms.....	99
Machine data not valid.....	99
Setting data not valid.....	100
Flash memory error.....	100
6504 Supported alarms.....	100
6506 Supported warnings.....	100
6507 Profile and software version.....	100
6508 Operating time.....	101
6509 Offset value.....	101
650B Serial number.....	101
6.13 SDO abort codes.....	102
7 Default parameters list.....	104
7.1 Communication Profile Area objects.....	104
7.2 Manufacturer Specific Profile Area objects.....	106
7.3 Standardized Device Profile Area objects.....	107

Subject index

1	
1000	NMT_DeviceType_U32.....74
1001	ERR_ErrorRegister_U8.....74
1006	NMT_CycleLen_U32.....74
1008	NMT_ManufactDevName_VS.....74
1009	NMT_ManufactHwVers_VS.....75
100A	NMT_ManufactSwVers_VS.....75
1018-00	NMT_IdentityObject_REC.....75
1018-01	NMT_IdentityObject_REC.VendorID_U32.....75
1018-02	NMT_IdentityObject_REC.ProductCode_U32. 75
1018-03	NMT_IdentityObject_REC.RevisionNo_U32.....75
1018-04	NMT_IdentityObject_REC.SerialNo_U3276
1020-00	CFM_VerifyConfiguration_REC.....76
1020-01	CFM_VerifyConfiguration_REC.ConfDate_U3276
1020-02	CFM_VerifyConfiguration_REC.ConfTime_U3276
1030-00	NMT_InterfaceGroup_0h_REC.....76
1030-01	NMT_InterfaceGroup_0h_REC.InterfaceIndex_ U16.....76
1030-02	NMT_InterfaceGroup_0h_REC.InterfaceDescrip tion_VSTR.....77
1030-03	NMT_InterfaceGroup_0h_REC.InterfaceType_U 8.....77
1030-04	NMT_InterfaceGroup_0h_REC.InterfaceMtu_U 16.....77
1030-05	NMT_InterfaceGroup_0h_REC.InterfacePhysAd dress_OSTR.....77
1030-06	NMT_InterfaceGroup_0h_REC.InterfaceName_ VSTR.....77
1030-07	NMT_InterfaceGroup_0h_REC.InterfaceOperSt ate_U8.....77
1030-08	NMT_InterfaceGroup_0h_REC.InterfaceAdmin State_U8.....78
1030-09	NMT_InterfaceGroup_0h_REC.Valid_BOOL.....78
1300	SDO_SequLayerTimeout_U32.....78
1800-00	PDO_TxCommParam_00h_REC.....78
1800-01	PDO_TxCommParam_00h_REC.NodeID_U8...78
1800-02	PDO_TxCommParam_00h_REC.MappingVersio n_U8.....78
1A00-00	PDO_TxMappParam_00h_AU64.....79
1A00-01	PDO_TxMappParam_00h_AU64.ObjectMappin g.....79
1A00-02	PDO_TxMappParam_00h_AU64.ObjectMappin g.....79
1COB-00	DLL_CNLossSoC_REC.....80
1COB-01	DLL_CNLossSoC_REC.CumulativeCnt_U32.....80
1COB-02	DLL_CNLossSoC_REC.ThresholdCnt_U3280
1COB-03	DLL_CNLossSoC_REC.Threshold_U32..80
1COFB-00	DLL_CNCRCErrror_REC.....80
1COFB-01	DLL_CNCRCErrror_REC.CumulativeCnt_U32...81
1COFB-02	DLL_CNCRCErrror_REC.ThresholdCnt_U32.....81
1COFB-03	DLL_CNCRCErrror_REC.Threshold_U3281
1C14	DLL_CNLossOfSocTolerance_U32.....81
1F81-00	NMT_NodeAssignment_AU32.....81
1F81-01	NMT_NodeAssignment_AU32.NodeAssignment82
1F82	NMT_FeatureFlags_U32.....83
1F83	NMT_EPLVersion_U8.....84
1F8C	NMT_CurrNMTState_U8.....85
1F8D-00	NMT_PResPayloadLimitList_AU16.....85
1F8D-01	NMT_PResPayloadLimitList_AU16.PResPayload Limit.....86
1F93-00	NMT_EPLNodeID_REC.....86
1F93-01	NMT_EPLNodeID_REC.NodeID_U8.....86

1F93-02	
NMT_EPLNodeID_REC.NodeIDByHW_BOOL...	86
1F98-00	NMT_CycleTiming_REC.....86
1F98-01	
NMT_CycleTiming_REC.IsochrTxMaxPayload_U	
16.....	86
1F98-02	
NMT_CycleTiming_REC.IsochrRxMaxPayload_U	
16.....	87
1F98-03	
NMT_CycleTiming_REC.PresMaxLatency_U32	
.....	87
1F98-04	
NMT_CycleTiming_REC.PreqActPayloadLimit_U	
16.....	87
1F98-05	
NMT_CycleTiming_REC.PresActPayloadLimit_U	
16.....	87
1F98-06	
NMT_CycleTiming_REC.AsndMaxLatency_U32	
.....	88
1F98-07	
NMT_CycleTiming_REC.MultiplCycleCnt_U8..	88
1F98-08	NMT_CycleTiming_REC.AsyncMTU_U16
.....	88
1F99	NMT_CNBasicEthernetTimeout_U32.....88
1F9B-00	NMT_MultiplCycleAssign_AU8.....89
1F9B-01	NMT_MultiplCycleAssign_AU8.CycleNo
.....	89
1F9E	NMT_ResetCmd_U8.....89
2	
2002	Speed format.....90
6	
6000	Operating parameters.....91
6001	Measuring units per revolution.....93
6002	Total measuring range in measuring units
.....	94
6003	Preset value.....96
6004	Position value.....97
6030	Speed value.....97
6500	Operating status.....98
6501	Singleturn resolution.....98
6502	Number of revolutions.....99
6503	Alarms.....99
6504	Supported alarms.....100
6506	Supported warnings.....100
6507	Profile and software version.....100
6508	Operating time.....101
6509	Offset value.....101
650B	Serial number.....101

A

Alarms.....	99
C	
CFM_VerifyConfiguration_REC.....	76
CFM_VerifyConfiguration_REC.ConfDate_U32..	76
CFM_VerifyConfiguration_REC.ConfTime_U32..	76
Code sequence.....	91, 98
D	
DLL_CNCRCErrror_REC.....	80
DLL_CNCRCErrror_REC.CumulativeCnt_U32.....	81
DLL_CNCRCErrror_REC.Threshold_U32.....	81
DLL_CNCRCErrror_REC.ThresholdCnt_U32.....	81
DLL_CNLossOfSocTolerance_U32.....	81
DLL_CNLossSoC_REC.....	80
DLL_CNLossSoC_REC.CumulativeCnt_U32.....	80
DLL_CNLossSoC_REC.Threshold_U32.....	80
DLL_CNLossSoC_REC.ThresholdCnt_U32.....	80
E	
ERR_ErrorRegister_U8.....	74
F	
Flash memory error.....	100
M	
Machine data not valid.....	99
Measuring units per revolution.....	93
N	
NMT_CNBasicEthernetTimeout_U32.....	88
NMT_CS_BASIC_ETHERNET.....	69
NMT_CS_NOT_ACTIVE.....	67
NMT_CS_OPERATIONAL.....	68
NMT_CS_PRE_OPERATIONAL_1.....	67
NMT_CS_PRE_OPERATIONAL_2.....	67
NMT_CS_READY_TO_OPERATE.....	68
NMT_CS_STOPPED.....	68
NMT_CurrNMTState_U8.....	85
NMT_CycleLen_U32.....	74
NMT_CycleTiming_REC.....	86
NMT_CycleTiming_REC.AsndMaxLatency_U32..	88
NMT_CycleTiming_REC.AsyncMTU_U16.....	88
NMT_CycleTiming_REC.IsochrRxMaxPayload_U1	
6.....	87
NMT_CycleTiming_REC.IsochrTxMaxPayload_U16	
.....	86
NMT_CycleTiming_REC.MultiplCycleCnt_U8.....	88
NMT_CycleTiming_REC.PreqActPayloadLimit_U1	
6.....	87
NMT_CycleTiming_REC.PresActPayloadLimit_U16	
.....	87
NMT_CycleTiming_REC.PresMaxLatency_U32...	87
NMT_DeviceType_U32.....	74
NMT_EPLNodeID_REC.....	86
NMT_EPLNodeID_REC.NodeID_U8.....	86
NMT_EPLNodeID_REC.NodeIDByHW_BOOL.....	86

NMT_EPLVersion_U8.....	84
NMT_FeatureFlags_U32.....	83
NMT_IdentityObject_REC.....	75
NMT_IdentityObject_REC.ProductCode_U32.....	75
NMT_IdentityObject_REC.RevisionNo_U32.....	75
NMT_IdentityObject_REC.SerialNo_U32.....	76
NMT_IdentityObject_REC.VendorID_U32.....	75
NMT_InterfaceGroup_0h_REC.....	76
NMT_InterfaceGroup_0h_REC.InterfaceAdminSta te_U8.....	78
NMT_InterfaceGroup_0h_REC.InterfaceDescripti on_VSTR.....	77
NMT_InterfaceGroup_0h_REC.InterfaceIndex_U1 6.....	76
NMT_InterfaceGroup_0h_REC.InterfaceMtu_U16	77
NMT_InterfaceGroup_0h_REC.InterfaceName_VS TR.....	77
NMT_InterfaceGroup_0h_REC.InterfaceOperStat e_U8.....	77
NMT_InterfaceGroup_0h_REC.InterfacePhysAddr ess_OSTR.....	77
NMT_InterfaceGroup_0h_REC.InterfaceType_U8	77
NMT_InterfaceGroup_0h_REC.Valid_BOOL.....	78
NMT_ManufactDevName_VS.....	74
NMT_ManufactHwVers_VS.....	75
NMT_ManufactSwVers_VS.....	75
NMT_MultiplCycleAssign_AU8.....	89
NMT_MultiplCycleAssign_AU8.CycleNo.....	89
NMT_NodeAssignment_AU32.....	81
NMT_NodeAssignment_AU32.NodeAssignment	82
NMT_PResPayloadLimitList_AU16.....	85




NMT_PResPayloadLimitList_AU16.PResPayloadLi mit.....	86
NMT_ResetCmd_U8.....	89
Number of revolutions.....	99
O	
Offset value.....	101
Operating parameters.....	91
Operating status.....	98
Operating time.....	101
P	
PDO_TxCommParam_00h_REC.....	78
PDO_TxCommParam_00h_REC.MappingVersion_ U8.....	78
PDO_TxCommParam_00h_REC.NodeID_U8.....	78
PDO_TxMappParam_00h_AU64.....	79
PDO_TxMappParam_00h_AU64.ObjectMapping	79
Position value.....	97
Preset value.....	96
Profile and software version.....	100
R	
Restore default parameters.....	92
S	
Scaling function.....	91, 98
SDO_SequLayerTimeout_U32.....	78
Serial number.....	101
Setting data not valid.....	100
Singleturn resolution.....	98
Speed format.....	90
Speed value.....	97
Store parameters.....	92
Supported alarms.....	100
Supported warnings.....	100
T	
Total measuring range in measuring units.....	94

Typographic and iconographic conventions

In this guide, to make it easier to understand and read the text the following typographic and iconographic conventions are used:

- parameters and objects both of the device and the interface are coloured in **GREEN**;
- alarms are coloured in **RED**;
- states are coloured in **FUCSIA**.

When scrolling through the text some icons can be found on the side of the page: they are expressly designed to highlight the parts of the text which are of great interest and significance for the user. Sometimes they are used to warn against dangers or potential sources of danger arising from the use of the device. You are advised to follow strictly the instructions given in this guide in order to guarantee the safety of the user and ensure the performance of the device. In this guide the following symbols are used:

	This icon, followed by the word WARNING , is meant to highlight the parts of the text where information of great significance for the user can be found: user must pay the greatest attention to them! Instructions must be followed strictly in order to guarantee the safety of the user and a correct use of the device. Failure to heed a warning or comply with instructions could lead to personal injury and/or damage to the unit or other equipment.
	This icon, followed by the word NOTE , is meant to highlight the parts of the text where important notes needful for a correct and reliable use of the device can be found. User must pay attention to them! Failure to comply with instructions could cause the equipment to be set wrongly: hence a faulty and improper working of the device could be the consequence.
	This icon is meant to highlight the parts of the text where suggestions useful for making it easier to set the device and optimize performance and reliability can be found. Sometimes this symbol is followed by the word EXAMPLE when instructions for setting parameters are accompanied by examples to clarify the explanation.

Preliminary information

This guide is designed to provide the most complete information the operator needs to correctly and safely install and operate the following encoders **equipped with POWERLINK interface**:

EMxxx-13-14-PL2-...	(multiturn encoder 13 + 14 bits)
HSxxx-18-00-PL2-...	(singleturn encoder 18 bits)
HMxxx-16-14-PL2-...	(multiturn encoder 16 + 14 bits)

For technical specifications please refer to the product datasheet.

To make it easier to read the text, this guide can be divided into some main sections.

In the first section (from chapter 1 to chapter 4) general information concerning the safety, the mechanical installation and the electrical connection.

In the second section (chapter 5) information on how to install and configure the encoder under the Automation Studio development environment as well as tips for setting up and running properly and efficiently the unit are provided.

In the third section (chapters 6 and 7) both general and specific information is given on the POWERLINK interface. In this section the interface features and the parameters implemented in the unit are fully described.

Glossary of POWERLINK terms

POWERLINK, like many other networking systems, has a set of unique terminology. Table below contains a few of the technical terms used in this guide to describe the POWERLINK interface. They are listed in alphabetical order.

Ageing	Ageing is a common mechanism to maintain (cache) tables. Entries which are not used or refreshed are removed after a specified time.
Application Process	The Application Process is the task on the Application Layer.
Async-only CN	An Async-only CN is operated in a way, that it is not accessed cyclically in the isochronous slot by the MN. It is polled during the asynchronous period by a StatusRequest message.
Asynchronous Data	Data in a POWERLINK network which is not time critical. Within the POWERLINK cycle there is a specific period reserved for Asynchronous Data which is shared by all nodes. Each node connected to the network can send asynchronous data by requesting it to the Managing Node. The Managing Node keeps a list of all asynchronous data requests and will subsequently grant the network access to one node after the other. Refer also to page 65.
Asynchronous Period	The Asynchronous Period is the second part of the POWERLINK cycle, starting with a Start of Asynchronous (SoA) frame. Refer to page 65.
Asynchronous Scheduling	The MN's asynchronous scheduler decides when a requested asynchronous data transfer will happen.
Basic Ethernet Mode	Basic Ethernet Mode provides the Legacy Ethernet communication. Refer also to page 69.
Bus	A bus is a communication medium connecting several nodes. Data can be transferred via serial or parallel circuits, that is, via electrical conductors or fiber optic.
CANopen	CANopen is a network technology optimized for the usage in industrial control environments, in machine internal networks and in embedded systems (any control unit deeply "embedded" in a device with electronics). The lower-layer implementation of CANopen is based upon CAN (Controller Area Network).
Continuous	Continuous is a POWERLINK communication class where isochronous communication takes place every cycle (the opposite to multiplexed).
Controlled Node (CN)	Node in a POWERLINK network without the ability to manage the SCNM mechanism. Refer to page 63.

Cycle State Machine	The Cycle State Machine controls the POWERLINK cycle on the Data Link Layer and is itself controlled by the NMT state machine defining the current operating mode.
Cycle Time	The time between two consecutive Start of Cyclic (SoC) frames – i.e. repeating – process. The Cycle Time includes the time for data transmission and some idle time before the beginning of the next cycle. Refer also to page 51.
Destination NAT (D-NAT)	D-NAT (Destination- Network Address Translation) changes the destination address of the IP / ICMP packet.
Determinism	Determinism means that a system responds in a predictable (deterministic) manner.
Deterministic Communication	It describes a communication process whose timing behaviour can be predicted exactly. I.e. the time when a message reaches the recipient is predictable. Refer to page 65.
Device Configuration File	The configuration parameters of a specific device are stored in the Device Configuration File (XDC).
Device Description File	All device dependent information is stored in the Device Description File (XDD) of each device. Refer to page 69.
Domain	In the context of CANopen: A Domain is a data object of arbitrary type and length which can be transferred over a POWERLINK network. In the context of internet protocols: A Domain is a part of the internet name space which is supported by the Domain Name System (DNS).
Encoder Profile	POWERLINK integrates with CANopen Profiles "DS301 CANopen Application Layer and Communication Profile" and "DS406 Device Profile for Encoders" for device interoperability. The "DS406 Device Profile for Encoders" is intended to define a standard application interface for encoders. The profile is a supplement to the "DS301 CANopen Application Layer and Communication Profile", so it is mandatory to read the DS301 profile before implementing the encoder profile. POWERLINK encoders from Lika Electronic comply with the "EPSG Draft Standard 301 Ethernet POWERLINK Communication Profile Specification Version 1.2.0". See also "Profile". Refer to page 63.
Ethernet POWERLINK (EPL)	An extension to Legacy Ethernet on layer 2, to exchange data under hard real-time constraints. It was developed for deterministic data exchange, short cycle time and isochronous operation in industrial automation.
IdentRequest	IdentRequests are POWERLINK frames sent by the MN in order to identify active CNs waiting to be included into the network.
IdentResponse	The IdentResponse is a special form of an ASnd frame in response to an IdentRequest.
Idle Period	The Idle Period is time interval remaining between the completed asynchronous period and the beginning of the next cycle. Refer also to page 66.

IEEE 1588	This standard defines a protocol enabling synchronisation of clocks in distributed networked devices (e.g. connected via Ethernet).
IP address	The IP address is the name of the unit in a network using the Internet protocol. Refer to page 32.
Isochronous	Pertains to processes that require timing coordination to be successful. Isochronous data transfer ensures that data flows continuously and at a steady rate in close timing with the ability of connected devices.
Isochronous Data	Data in a POWERLINK network which is to be transmitted every cycle (or every nth cycle in case of multiplexed isochronous data). Refer also to page 65.
Isochronous Period	The Isochronous Period of a POWERLINK cycle offers deterministic operation, i.e. it is reserved for the exchange of (continuous or multiplexed) isochronous data. Refer to page 65.
Legacy Ethernet	Ethernet as standardised in IEEE 802.3 (non-deterministic operation in non-time-critical environments).
MAC address	The MAC address is an identifier unique worldwide consisting of two parts: the first 3 bytes are the manufacturer ID and are provided by IEE standard authority; the last three bytes represent a consecutive number of the manufacturer. Refer to page 32.
Managing Node (MN)	A node capable to manage the SCNM mechanism in a POWERLINK network. Refer to page 63.
Media Access Control (MAC)	One of the sub-layers of the Data Link Layer in the POWERLINK reference model that controls who gets access to the medium to send a message.
Multiplexed	Multiplexed is a POWERLINK communication class where cyclic communication takes place in such a way that m nodes are served in s cycles (the opposite to continuous).
Multiplexed CN	A node which is allowed to send isochronous data every n th cycle.
Multiplexed Timeslot	A slot destined to carry multiplexed isochronous data, i.e. the timeslot is shared among multiple nodes.
NetTime	The MN's clock time is distributed to all CNs within the SoC frame.
Network Management (NMT)	Network Management functions and services in the POWERLINK model. It performs initialisation, configuration and error handling in a POWERLINK network. Refer to page 70.
NMT State Machine	The state machine controlling the overall operating mode and status of a POWERLINK node. Refer to page 70.
Object Directory	The repository of all data objects accessible over POWERLINK communications.

PollRequest	A PollRequest is a frame, which is used in the isochronous part of the cyclic communication. The MN request with this frame the CN to send its data. Refer to page 64.
PollResponse	A PollResponse is a frame, which is used in the isochronous part of the cyclic communication. The CN responses with this frame to a PollRequest frame from an MN. Refer to page 64.
POWERLINK Command Layer	The POWERLINK Command Layer defines commands to access parameters of the object dictionary. This layer is on top of the Sequence Layer and distinguishes between an expedited and a segmented transfer.
POWERLINK Cycle	Data exchange within a POWERLINK network is structured in fix intervals, called cycles. The cycle is subdivided into the isochronous and the asynchronous period and is organized by the MN. Refer to page 65.
POWERLINK Mode	The POWERLINK Mode includes all NMT states in which POWERLINK cycles are run. Refer to page 66.
POWERLINK Node ID	Each POWERLINK node (MN, CN and Router) is addressed by an 8 bit POWERLINK Node ID on the POWERLINK layer. This ID has only local significance (i.e. it is unique within a POWERLINK segment). Refer to page 32.
Precision Time Protocol (PTP)	IEEE 1588, Standard for a Precision Clock Synchronisation Protocol for Networked Measurement and Control Systems.
Process Data Object (PDO)	Object for isochronous data exchange between POWERLINK nodes.
Profile	Profiles define application-specific functionality to ensure the openness of POWERLINK is utilized consistently. Profiles can cover simple devices such as encoders by defining how signals are used and how they are physically connected. However, profiles are increasingly covered more complex systems or requirements. Profiles guarantee quicker system design and they support faster device interchange, promoting competition amongst vendors, increased choice for users and full interoperability. POWERLINK encoders from Lika Electronic comply with the "EPSG Draft Standard 301 Ethernet POWERLINK Communication Profile Specification Version 1.2.0". See also "Encoder profile". See on page 61.
Real-time	Real-time means that a system processes external events within a defined time. If the reaction of a system is predictable, one speaks of a deterministic system. The general requirements for real-time are therefore: deterministic response and defined response time. Refer to page 65.
Reserved	Reserved bits shall be set 0 by the sender. The receiver shall not interpret such bits. It is not allowed to use reserved bits. Their use is reserved for further development or by extensions of this specification.
Router Type 1	A Type 1 POWERLINK Router is a coupling element in a

	network that allows IP communication between a POWERLINK segment and any other datalink layer protocol carrying IP e.g. legacy Ethernet, POWERLINK etc. It is usually a separate network element acting as Controlled Node within the POWERLINK segment.
Router Type 2	A Type 2 POWERLINK Router is a router between a POWERLINK segment and a CANopen network.
Sequence Layer	The POWERLINK Sequence Layer provides the service of a reliable bidirectional connection that guarantees that no messages are lost or duplicated and that all messages arrive in the correct order.
Service Data Object (SDO)	Peer to peer communication with access to the object dictionary of a device. Refer to page 72.
Slot Communication Network Management (SCNM)	In a POWERLINK network, the managing node allocates data transfer time for data from each node in a cyclic manner within a guaranteed cycle time. Within each cycle there are slots for Isochronous Data, as well as for Asynchronous Data for ad hoc communication. The SCNM mechanism ensures that there are no collisions during physical network access of any of the networked nodes thus providing deterministic communication via Legacy Ethernet. Refer to page 61.
Source NAT (S-NAT)	S-NAT (Source - Network Address Translation) changes the source address of the IP / ICMP packet.
StatusRequest	A StatusRequest frame is a special SoA frame used to poll the status of a node.
StatusResponse	A StatusResponse frame is transmitted by a CN upon assignment of the asynchronous slot via the StatusRequest in the SoA frame.
TCP/IP	<p>The Ethernet system is designed solely to carry data. It is comparable to a highway as a system for transporting goods and passengers. The data is actually transported by protocols. This is comparable to cars and commercial vehicles transporting passengers and goods on the highway.</p> <p>Tasks handled by the basic Transmission Control Protocol (TCP) and Internet Protocol (IP) (abbreviated to TCP/IP):</p> <ol style="list-style-type: none"> 1. The sender splits the data into a sequence of packets. 2. The packets are transported over the Ethernet to the correct recipient. 3. The recipient reassembles the data packets in the correct order. 4. Faulty packets are sent again until the recipient acknowledges that they have been transferred successfully.
Topology	<p>Network structure. Commonly used structures:</p> <ul style="list-style-type: none"> • Line topology; • Ring topology; • Star topology;

	<ul style="list-style-type: none">• Tree topology. Refer to page 61.
Transmission rate	Data transfer rate (in bps). Refer to page 62.

List of abbreviations

Table below contains a list of abbreviations (in alphabetical order) which may be used in this guide to describe the POWERLINK interface.

ACL	Access Control List
ARP	Address Resolution Protocol
ASnd	Asynchronous Send (POWERLINK frame type)
CAN	Controller Area Network
CiA	CAN in Automation
CN	POWERLINK Controlled Node
DCF	Device Configuration File
EA	Exception Acknowledge (flag in POWERLINK frame)
EIA	Electronic Industries Association
EMC	Electro Magnetic Compatibility
EN	Exception New (flag in POWERLINK frame)
EPL	Ethernet POWERLINK
EPSG	Ethernet POWERLINK Standardisation Group
ICMP	Internet Control Message Protocol
ID	Identifier
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronic Engineers
IP	Internet Protocol
MAC	Media Access Control
MIB	Management Information Base
MN	POWERLINK Managing Node
MS	Multiplexed Slot (flag in POWERLINK frame)
MSS	Maximum Segment Size
MTU	Maximum Transmission Unit
NAT	Network Address Translation
NIL	Not in List (Basic Data Type)
NMT	Network Management
PDO	Process Data Object
PR	Priority (bit field in POWERLINK frame)
PReq	PollRequest (POWERLINK frame type)

PRes	PollResponse (POWERLINK frame type)
PS	Prescaled Slot (flag in POWERLINK frame)
PTP	Precision Time Protocol
RD	Ready (flag in POWERLINK frame)
RFC	Requests for Comments
RPDO	Receive Process Data Object
RS	Request to Send (flag in POWERLINK frame)
SCNM	Slot Communication Network Management
SDO	Service Data Object
SNMP	Simple Network Management Protocol
SoA	Start of Asynchronous (POWERLINK frame type)
SoC	Start of Cyclic (POWERLINK frame type)
TCP	Transmission Control Protocol
TIA	Telecommunications Industry Association
TPDO	Transmit Process Data Object
UDP	User Datagram Protocol
VPN	Virtual Private Network
XDC	XML device configuration file
XDD	XML device description file

References

- [1] EPSG Draft Standard 301, Ethernet POWERLINK, Communication Profile Specification, Version 1.2.0
- [2] EPSG Draft Standard 311, Ethernet POWERLINK, XML Device Description, Version 1.0.0
- [3] CiA Draft Standard Proposal 301, Application layer and communication profile, Version 4.2.0
- [4] CiA Draft Standard 406, Device profile for encoders, Version 4.0.1
- [5] EPSG Draft Standard 302-A (EPSG DS 302-A), Ethernet POWERLINK, Part A: High Availability, Version 1.1.0
- [6] EPSG Draft Standard 302-B (EPSG DS 302-B), Ethernet POWERLINK, Part B: Multiple-ASnd, Version 1.0.0
- [7] EPSG Draft Standard 302-C (EPSG DS 302-C), Ethernet POWERLINK, Part C: PollResponse Chaining, Version 1.0.0
- [8] EPSG Draft Standard 302-D (EPSG WDP 302-D), Ethernet POWERLINK, Part D: Multiple PReq/Pres, Version 1.0.0
- [9] EPSG Draft Standard 302-E (EPSG WDP 302-E), Ethernet POWERLINK, Part E: Dynamic Node Allocation, Version 1.0.0
- [10] IEC 61918 Industrial communication networks – Installation of communication networks in industrial premises
- [11] IEC 61784-5-13 Industrial communication networks – Profiles – Part 5-13: Installation of fieldbuses – Installation profiles for CPF 13

1 Safety summary



1.1 Safety

- Always adhere to the professional safety and accident prevention regulations applicable to your country during device installation and operation;
- installation and maintenance operations have to be carried out by qualified personnel only, with power supply disconnected and stationary mechanical parts;
- device must be used only for the purpose appropriate to its design: use for purposes other than those for which it has been designed could result in serious personal and/or the environment damage;
- high current, voltage and moving mechanical parts can cause serious or fatal injury;
- warning! Do not use in explosive or flammable areas;
- failure to comply with these precautions or with specific warnings elsewhere in this manual violates safety standards of design, manufacture, and intended use of the equipment;
- Lika Electronic assumes no liability for the customer's failure to comply with these requirements.



1.2 Electrical safety

- Turn off the power supply before connecting the device;
- connect according to the explanation in the "4 -Electrical connections" section on page 29;
- in compliance with the 2014/30/EU norm on electromagnetic compatibility, the following precautions must be taken:
 - before handling and installing, discharge electrical charge from your body and tools which may come in touch with the device;
 - power supply must be stabilized without noise, install EMC filters on device power supply if needed;
 - always use shielded cables (twisted pair cables whenever possible);
 - avoid cables runs longer than necessary;
 - avoid running the signal cable near high voltage power cables;
 - mount the device as far as possible from any capacitive or inductive noise source, shield the device from noise source if needed;
 - to guarantee a correct working of the device, avoid using strong magnets on or near by the unit;
 - minimize noise by connecting the shield and/or the connector housing and/or the frame to ground. Make sure that ground is not affected by noise. The connection point to ground can be situated both on the device side and on user's side. The best solution to minimize the interference must be carried out by the user. Provide the ground connection as close as possible to the encoder. We suggest using the ground point provided in the cap, use one TCEI M3 x 6 cylindrical head screw with two tooth lock washers.





1.3 Mechanical safety

- Install the device following strictly the information in the "3 -Mounting instructions" section on page 24;
- mechanical installation has to be carried out with stationary mechanical parts;
- do not disassemble the encoder;
- do not tool the encoder or its shaft;
- delicate electronic equipment: handle with care; do not subject the device and the shaft to knocks or shocks;
- respect the environmental characteristics declared by manufacturer
- unit with solid shaft: in order to guarantee maximum reliability over time of mechanical parts, we recommend a flexible coupling to be installed to connect the encoder and user's shaft; make sure the misalignment tolerances of the flexible coupling are respected;
- unit with hollow shaft: the encoder can be mounted directly on a shaft whose diameter has to respect the technical characteristics specified in the purchase order and clamped by means of the collar and, when requested, the anti-rotation pin.

2 Identification

Device can be identified through the **order code**, the **serial number** and the **MAC address** printed on the label applied to its body. Information is listed in the delivery document too. Please always quote the order code, the serial number and the MAC address when reaching Lika Electronic for purchasing spare parts or needing assistance. For any information on the technical characteristics of the product refer to the technical catalogue.



Warning: encoders having order code ending with "/Sxxx" may have mechanical and electrical characteristics different from standard and be supplied with additional documentation for special connections (Technical info).

3 Mounting instructions



WARNING

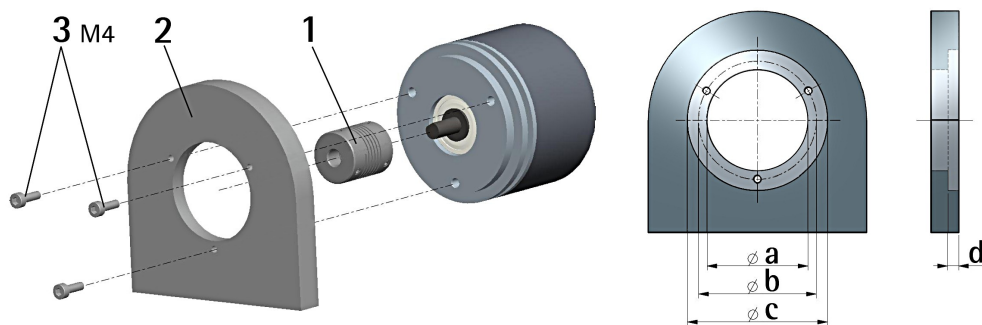
Installation and maintenance operations have to be carried out by qualified personnel only, with power supply disconnected and mechanical parts absolutely in stop.

For any information on the mechanical data and the electrical characteristics of the encoder please refer to the technical catalogue.

3.1 Solid shaft encoders

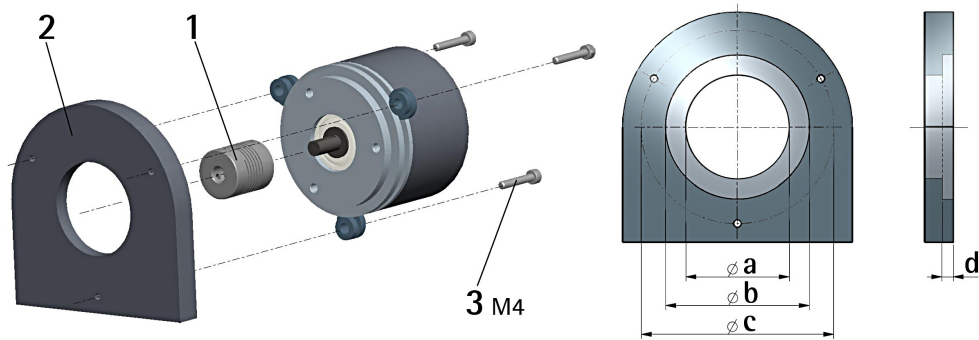
- Mount the flexible coupling **1** on the encoder shaft;
- fix the encoder to the flange **2** (or to the mounting bell) by means of screws **3**;
- secure the flange **2** to the support (or the mounting bell to the motor);
- mount the flexible coupling **1** on the motor shaft;
- make sure the misalignment tolerances of the flexible coupling **1** are respected.

3.1.1. Customary installation



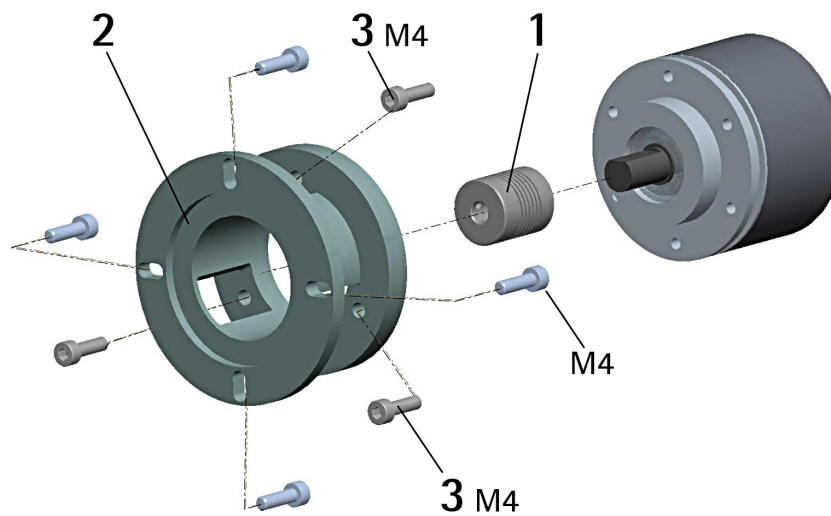
	a [mm]	b [mm]	c [mm]	d [mm]
EM58, HS58, HM58	-	42	50 F7	4
EM58S, HS58S, HM58S	36 H7	48	-	-

3.1.2 Installation using fixing clamps (code LKM-386)



	a [mm]	b [mm]	c [mm]	d [mm]
EM58, HS58, HM58	-	50 F7	67	4
EM58S, HS58S, HM58S	36 H7	-	67	-

3.1.3 Installation using a mounting bell (code PF4256)



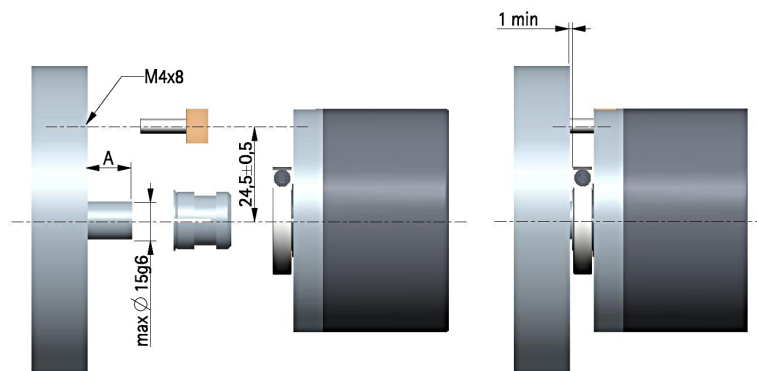
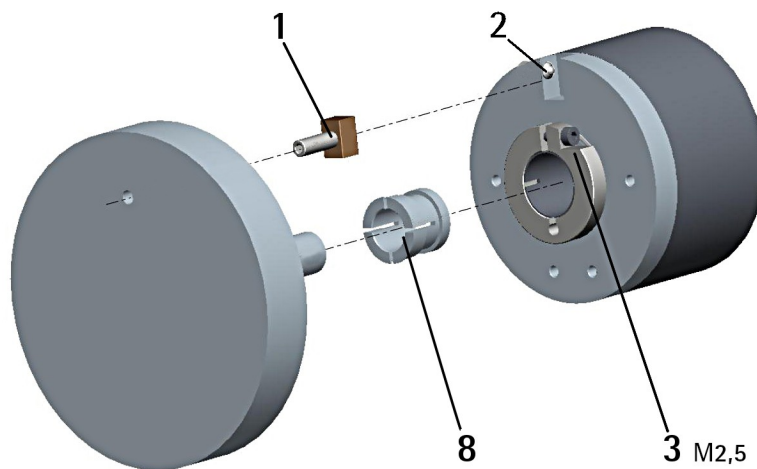
NOTE

In order to guarantee reliability over time of the encoder mechanical parts, we recommend a flexible coupling to be installed between the encoder and the motor shaft. Make sure the misalignment tolerances of the flexible coupling are respected.

3.2 Hollow shaft encoders

3.2.1 EMC58, HSC58, HMC58

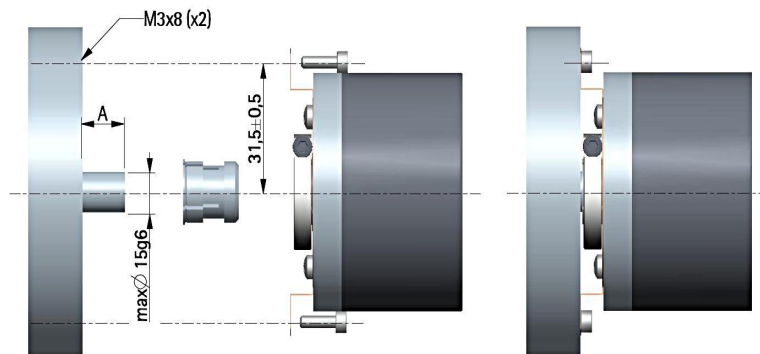
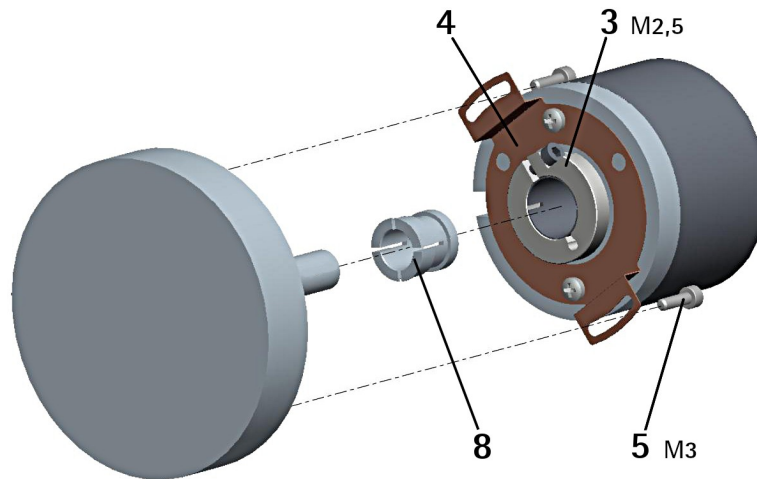
- Fasten the anti-rotation pin **1** to the rear of the motor (secure it using a locknut);
- mount the encoder on the motor shaft using the reducing sleeve **8** (if supplied). Avoid forcing the encoder shaft;
- insert the anti-rotation pin **1** into the slot on the flange of the encoder; this secures it in place by grub screw **2**, preset at Lika;
- fix the collar **3** to the encoder shaft (apply some threadlocker to the screw **3**).



A = min. 8 mm, max. 18 mm

3.2.2 EMC59, HSC59, HMC59

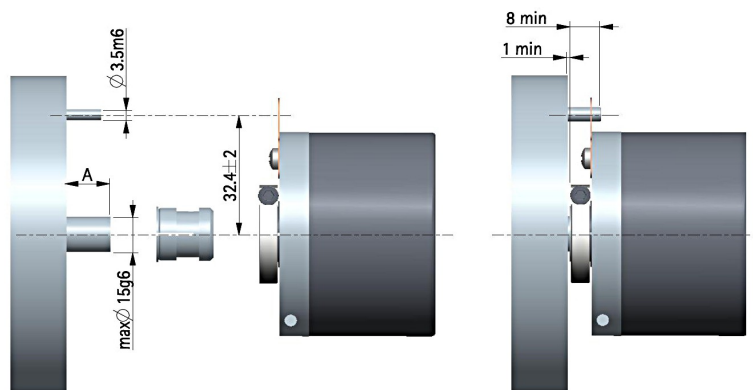
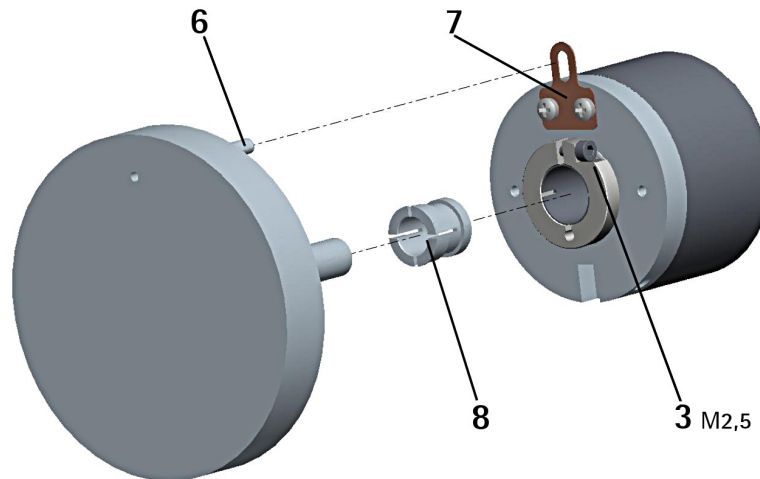
- Mount the encoder on the motor shaft using the reducing sleeve **8** (if supplied). Avoid forcing the encoder shaft;
- fasten the fixing plate **4** to the rear of the motor using two M3 x 8 cylindrical head screws **5**;
- fix the collar **3** to the encoder shaft (apply some threadlocker to the screw **3**).



A = min. 8 mm, max. 18 mm

3.2.3 EMC60, HSC60, HMC60

- Fix the tempered pin **6** to the rear of the motor;
- mount the encoder on the motor shaft using the reducing sleeve **8** (if supplied). Avoid forcing the encoder shaft;
- make sure the anti-rotation pin **6** is inserted properly into the fixing plate **7**;
- fix the collar **3** to the encoder shaft (apply some threadlocker to the screw **3**).



A = min. 8 mm, max. 18 mm



NOTE

You are strongly advised not to carry out any mechanical operations (drilling, milling, etc.) on the encoder shaft. This could cause serious damages to the internal parts and an immediate warranty loss. Please contact our technical personnel for the complete availability of "custom made" shafts.

4 Electrical connections



WARNING

Power supply must be turned off before performing any electrical connection!

4.1 Connection cap



WARNING

Do not remove or mount the connection cap with power supply switched ON. Damage may be caused to internal components.

The DIP switch meant to set the encoder node ID is located inside the connection cap. Thus you must remove the connection cap to access it.



NOTE

Be careful not to damage the internal components when you perform this operation.

To remove the connection cap loosen the two screws **1**. Please be careful with the internal connector.

Always replace the connection cap at the end of the operation. Take care in re-connecting the internal connector. Tighten the screws **1** using a tightening torque of approx. 2.5 Nm.



WARNING

You are required to check that the encoder body and the connection cap are at the same potential before replacing the connection cap!

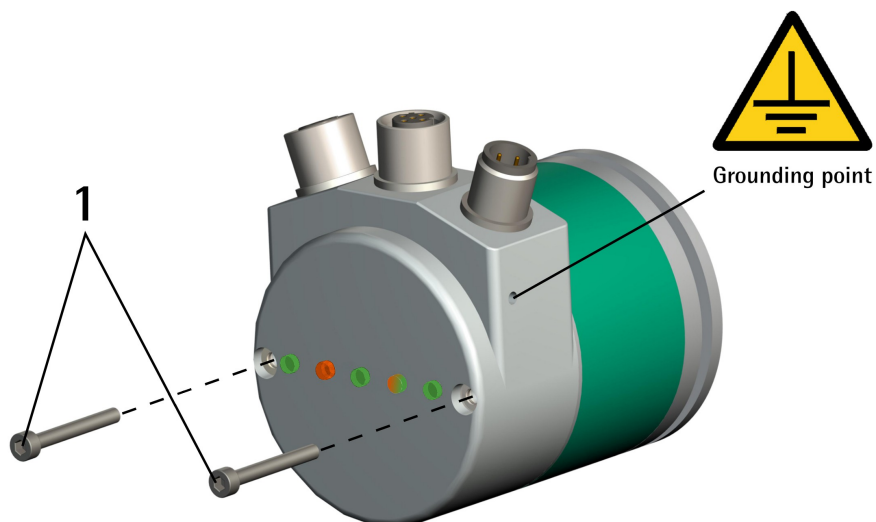


Figure 1 - Removing the connection cap

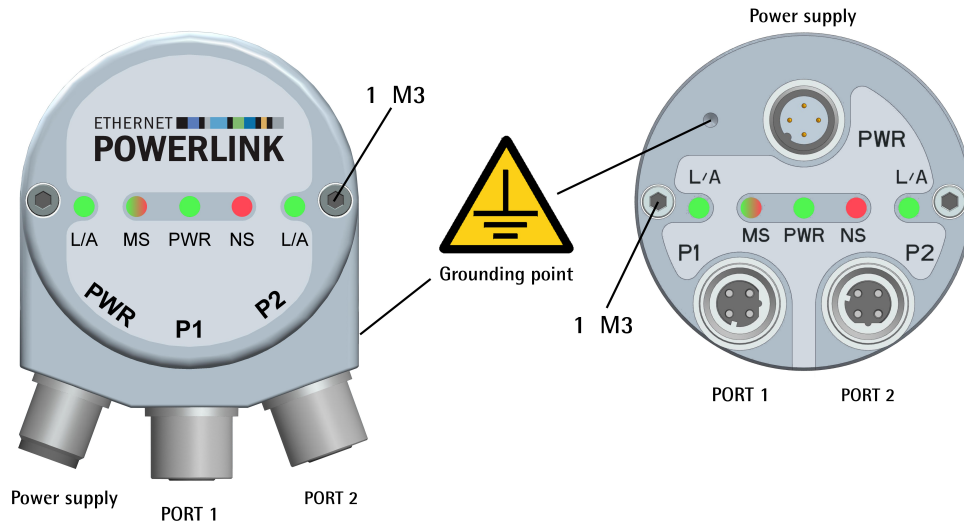
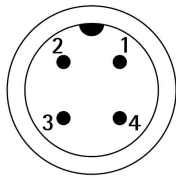


Figure 2 - Connectors and diagnostic LEDs

4.2 PWR Power supply connector (Figure 2)

M12 4-pin male connector with A coding is used for power supply.

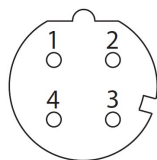


Description	Pin
+10Vdc +30Vdc	1
n.c.	2
0Vdc	3
n.c.	4

n.c. = not connected

4.3 P1 Port 1 and P2 Port 2 connectors (Figure 2)

Two M12 4-pin female connectors with D coding are used for Ethernet connection through port 1 and port 2.



Description	Pin
Tx Data +	1
Rx Data +	2
Tx Data -	3
Rx Data -	4

The Ethernet interface supports 100 Mbit/s, half duplex operation.

P1 PORT 1 and P2 PORT 2 M12 connectors have pin-out in compliance with the POWERLINK standard. Therefore you can use standard POWERLINK cables commercially available.

P1 PORT 1 and P2 PORT 2 connectors are interchangeable.

4.4 Network configuration: cables, hubs, switches - Recommendations

Cables and connectors comply with the POWERLINK specifications. Cables are CAT-5 shielded cables.

Standard POWERLINK cables commercially available can be used.

For complete information please refer to IEC 61918, IEC 61784-5-13 and IEC 61076-2-101.

To increase noise immunity only S/FTP or SF/FTP cables must be used (CAT-5).

The maximum cable length (100 meters) predefined by Ethernet 100Base-TX must be compulsorily fulfilled.

EPL recommends the use of hubs to fit POWERLINK jitter requirements.

Switches may be used to build a POWERLINK network.

It has to be considered that any POWERLINK network constructed with anything but Class 2 Repeater Devices does not conform to the POWERLINK standard.

Regarding wiring and EMC measures, the IEC 61918 and IEC 61784-5-13 must be considered.

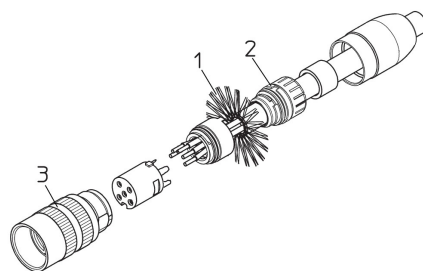
For a complete list of the available cordsets and connection kits please refer to the product datasheet ("Accessories" list).

4.5 Ground connection

To minimize noise connect properly the shield and/or the connector housing and/or the frame to ground. Connect properly the cable shield to ground on user's side. Lika's EC- pre-assembled cables are fitted with shield connection to the connector ring nut in order to allow grounding through the body of the device. Lika's E- connectors have a plastic gland, thus grounding is not possible. If metal connectors are used, connect the cable shield properly as recommended by the manufacturer. Anyway make sure that ground is not affected by noise. It is recommended to provide the ground connection as close as possible to the device. We suggest using the ground point provided in the cap (see Figure 2, use 1 TCEI M3 x 6 cylindrical head screw with 2 tooth lock washers).

4.6 Connection of the shield

Disentangle and shorten the shielding **1** and then bend it over the part **2**; finally place the ring nut **3** of the connector. Be sure that the shielding **1** is in tight contact with the ring nut **3**.



4.7 MAC address and IP address

The unit can be identified in the network through the **MAC address** and the **IP address**.

The MAC address has to be intended as a permanent and globally unique identifier assigned to the unit for communication on the physical layer; while the IP address is the name of the unit in a network using the Internet protocol. MAC address is 6-byte long and cannot be modified. It consists of two parts, numbers are expressed in hexadecimal notation: the first three bytes are used to identify the manufacturer (OUI, namely Organizationally Unique Identifier), while the last three bytes are the specific identifier of the unit. The MAC address can be found on the label applied to the encoder.

The IP address must be assigned by the user to each interface of the unit to be connected in the network, while the subnet mask is always 255.255.255.0 as in a class C net.

For additional information on the MAC address refer to the "5.3 MAC address" section on page 37.

For additional information on the IP address refer to the "4.8 POWERLINK Node ID: DIP A" section on page 32.

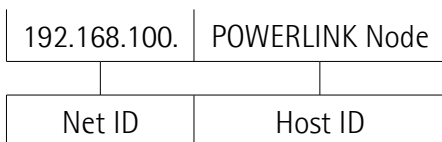
4.8 POWERLINK Node ID: DIP A



WARNING

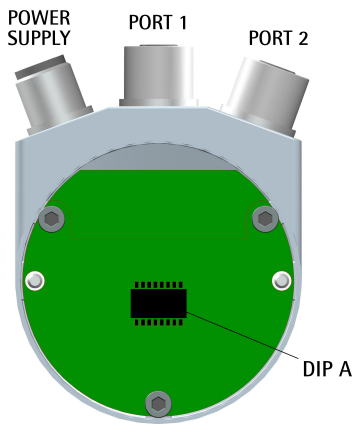
Power supply must be turned off before performing this operation!
The POWERLINK Node ID cannot be set via software.

The private class C Net ID **192.168.100.0** is used in the POWERLINK network. A class C network provides 254 (1-254) IP addresses which match the number of valid POWERLINK Node IDs. As they are CNs, Lika encoders accept values from 1 to 239, in compliance with the POWERLINK specifications. Address 0 is invalid, addresses from 240 to 255 are reserved to MNs or special functions. The POWERLINK MN (Active MN) is addressed to 240. The Host ID of the private class C Net ID 192.168.100.0 is identical to the POWERLINK Node ID. Hence the last byte of the IP address (Host ID) has the same value as the POWERLINK Node ID. The following figure illustrates the construction of the IP address.



The following table summarises the default IP parameters.

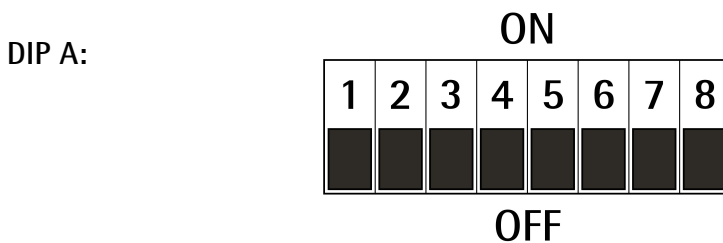
IP Parameter	IP address
IP address	192.168.100.<POWERLINK Node ID>
Subnet mask	255.255.255.0
Default Gateway	192.168.100.254 (it may be modified)



The POWERLINK node ID is set via hardware using the DIP A DIP switch located inside the enclosure. To access the DIP A DIP switch please refer to the "4.1 Connection cap" section on page 29.

Allowed node addresses range between 1 and 239. Address 0 is invalid. The default value is 1.

If you set an invalid address or any value greater than 239 the address will be forced to the default value.



Set the POWERLINK node ID in binary value: ON = 1, OFF = 0

bit	1	2	3	4	5	6	7	8
	LSB							MSB
	2^0	2^1	2^2	2^3	2^4	2^5	2^6	2^7

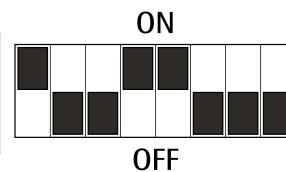


EXAMPLE

Set the node ID = 25:

$25_{10} = 0001\ 1001_2$ (binary value)

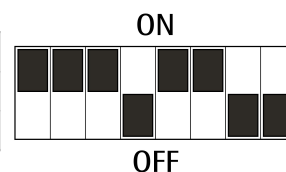
bit	1	2	3	4	5	6	7	8
	2^0	2^1	2^2	2^3	2^4	2^5	2^6	2^7
	ON	OFF	OFF	ON	ON	OFF	OFF	OFF



Set the node ID = 55:

$55_{10} = 0011\ 0111_2$ (binary value)

bit	1	2	3	4	5	6	7	8
	2^0	2^1	2^2	2^3	2^4	2^5	2^6	2^7
	ON	ON	ON	OFF	ON	ON	OFF	OFF



4.9 Diagnostic LEDs (Figure 2)

Five LEDs located in the cap of the encoder (see the Figure 2) are meant to show visually the operating or fault status of the encoder and the POWERLINK interface. The meaning of each LED is explained in the following tables.

LED	Description
L/A Link/Activity LED for port 1 P1 (green)	It shows the state and the activity of the physical link (port 1 P1).
BLINKING	Activity on port 1 P1.
ON	Port 1 P1 link active, no activity.

LED	Description
MS Module Status LED (green / red)	It shows the state of the POWERLINK device.
OFF	The device is switched OFF; the device is initializing; the device is not active.
FLICKERING green (50ms ON, 50ms OFF)	The network communication is in Basic Ethernet Mode , see NMT_CS_BASIC_ETHERNET , no POWERLINK traffic, see on page 69.
SINGLE FLASH green	The device is in Pre-Operational 1 state, see NMT_CS_PRE_OPERATIONAL_1 , see on page 67, asynchronous communication and no PDO exchange.
DOUBLE FLASH green	The device is in Pre-Operational 2 state, see NMT_CS_PRE_OPERATIONAL_2 , see on page 67, asynchronous and synchronous communication and no PDO exchange.
TRIPLE FLASH green	The device is in Ready to Operate state, see NMT_CS_READY_TO_OPERATE , see on page 68, there is no PDO exchange.
ON green	The device is in Operational state, see NMT_CS_OPERATIONAL , see on page 68, there is PDO exchange.
BLINCKING green (200ms ON, 200ms OFF)	The device is in Stopped state (for example, due to a controlled shutdown), see NMT_CS_STOPPED , see on page 68, there is no PDO exchange.

ON red	If the NS Network State Error LED is also lit, a fatal error has occurred.
---------------	--

LED	Description
PWR Power LED (green)	It shows the power supply state.
OFF	The encoder power supply is switched OFF.
ON	The encoder power supply is switched ON.

LED	Description
NS Network State Error LED (red)	It shows the current state of the network.
OFF	No error is currently active.
ON	<ul style="list-style-type: none"> • If the MS Status LED is green: a non fatal error has occurred. • If the MS Status LED is red: a fatal error has occurred.

LED	Description
L/A Link/Activity LED for port 2 P2 (green)	It shows the state and the activity of the physical link (port 2 P2).
BLINKING	Activity on port 2 P2.
ON	Port 2 P2 link active, no activity.

5 Quick reference

5.1 Before starting

Lika encoders are **CN devices** (Controlled Nodes e.g. Slaves: nodes without the ability to manage the SCNM mechanism, see on page 63) and comply with the "EPG Draft Standard 301 Ethernet POWERLINK Communication Profile Specification Version 1.2.0". as well as with the CANopen Profiles "DS301 CANopen Application Layer and Communication Profile" and "DS406 Device Profile for Encoders" according to the POWERLINK specifications. They require a **MN device** installed in the POWERLINK network (Managing Node, e.g. a Master: a node capable to manage the SCNM mechanism, see on page 63).

5.1.1 Network identity

Lika POWERLINK encoders use the following identity settings:

Identity Name: **Vendor ID**

Object: **1018-01 NMT_IdentityObject_REC.VendorID_U32**

Setting: **0000 0012Eh**

Identity Name: **Device Type**

Object: **1000 NMT_DeviceType_U32**

Setting: **0001 0196h** **HS58 series**
 0002 0196h **EM58, HM58 series**

Identity Name: **Product Code**

Object: **1018-02 NMT_IdentityObject_REC.ProductCode_U32**

Setting: **1000h** **HM58 series**
 1001h **EM58 series**
 1002h **HS58 series**

Identity Name: **Manufacturer Device Name**

Object: **1008 NMT_ManufactDevName_VS**

Setting: **EMxxx13/16384-PL-xx** **EM58 series**
 HSxxx18-PL-xx **HS58 series**
 HMxxx16/16384-PL-xx **HM58 series**

5.1.2 Network and communication settings

The **MAC address** of the device is always reported in the label applied to the encoder enclosure. See on page 32.

The **POWERLINK Node ID** is set via hardware using the DIP A DIP switch located inside the encoder enclosure. The default value is 1. See on page 32.

5.2 Configuring the encoder with Automation Studio V. 4.1 from B&R

In this manual some screenshots are shown to explain how to install and configure the encoder in a supervisor. In the specific example the development environment is Automation Studio V. 4.1.4.402 from B&R; it is used in combination with the X20CP1584 PLC from B&R. Therefore, the information on the installation of the XDD file, the assignment of the IP address and the device name, the configuration of the encoder in the network, topology, diagnostics, etc. will always refer to the aforementioned development tools. If you need to install the encoder using a different configuration tool, please read and follow carefully the instructions given in the documentation provided by the manufacturer.



Lika Electronic POWERLINK encoder documentation is complete with an **example project** supplied free of charge. This program is designed to make your own project planning, programming, communication and diagnostics with the Automation Studio V. 4.1 development environment user-friendly and reliable. For instance it allows to execute the following functions: setting the Preset value and execute it; saving the parameters; download the default parameters (refer to page 52 and following). You can find it in the **SW_Hx58x_PL_Example.zip** compressed file.

5.3 MAC address

The MAC address is an identifier unique worldwide.

The MAC-ID consists of two parts: the first three bytes are the manufacturer ID and are provided by IEE standard authority; the last three bytes represent a consecutive number of the manufacturer.



NOTE

The MAC address is always printed on the encoder label for commissioning purposes.

The MAC address has the following structure:

Bit value 47 ... 24			Bit value 23 ... 0		
10	B9	FE	X	X	X
Company code (OUI)			Consecutive number		

The MAC address can be read next to the **1030-05 NMT_InterfaceGroup_0h_REC.InterfacePhysAddress_OSTR** object.

5.4 Encoder installation under Automation Studio environment

5.4.1 Description of the XDD file

The functionality of a POWERLINK device is always described in a XDD file (XML Device Description file). The Device Description File provides information about the device basic communication and functional properties. It must be installed in the MN device.

The file name is primarily built up as follows:

OxVendorID_ProductName.xdd

e.g. 0000012E_Lika_EM13_16384

POWERLINK encoders from Lika Electronic are supplied with their own XDD file. Specific XDD files are provided to each encoder series, please refer to the order code: EM58 series; HS58 series; and HM58 series.

They are:

- **0000012E_Lika_EM13_16384_EthernetPOWERLINK.xdd**: it is intended for installation of **EM58 series encoders** ("0000012E_Lika" shows the Vendor ID -expressed in hexadecimal notation- and name; "EM" is the encoder series; "13" is the physical singleturn resolution of the encoder expressed in bits; "16384" is the number of physical revolutions of the multiturn versions);
- **0000012E_Lika_HS18_EthernetPOWERLINK.xdd**: it is intended for installation of **HS58 series encoders**;
- **0000012E_Lika_HM16_16384_EthernetPOWERLINK.xdd**: it is intended for installation of **HM58 series encoders**.

XDD files are paired with the **Lika.bmp** picture file available inside the file folder.

Follow the path **www.lika.biz > ROTARY ENCODERS > ABSOLUTE ENCODERS** to download the XDD files from Lika's corporate web site.

5.4.2 Installing the XDD file

In the menu bar of the main page, press **Tools** and then the **Import Fieldbus Device ...** command.

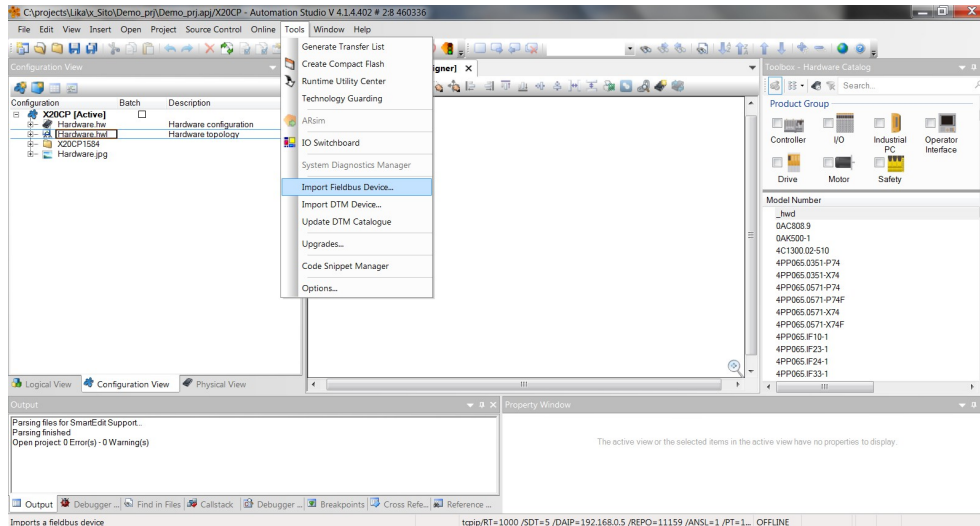


Figure 3 - Installing the XDD file

The **Open** dialog box will appear. Browse through the folders and search for the directory where the XDD file is located. Please make sure that the lika.bmp bitmap file representing the encoder is located in the same folder as the XDD file. Select the file corresponding to the encoder to be installed (please check the order code) and press the **Open** button to install it.

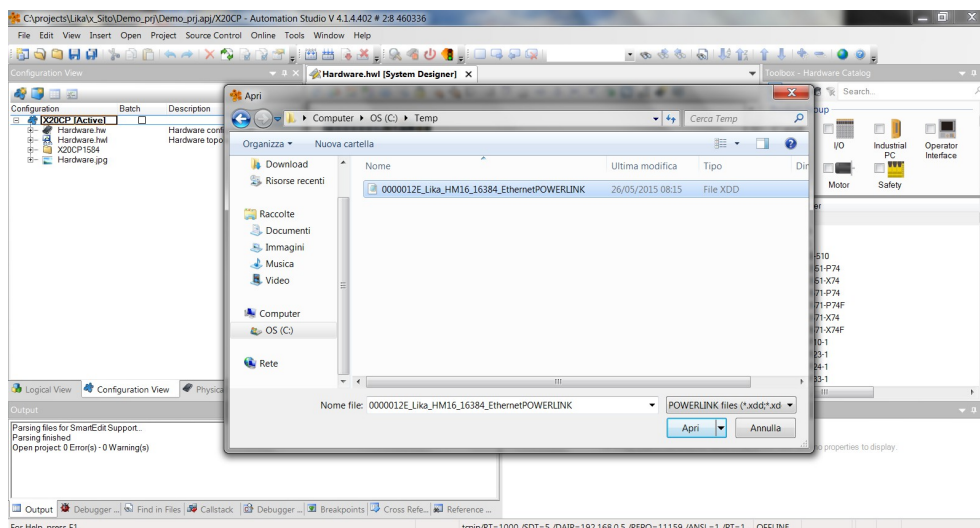


Figure 4 - Selecting the XDD file

As soon as the operation is carried out, a confirmation message will appear in the **Output** window.

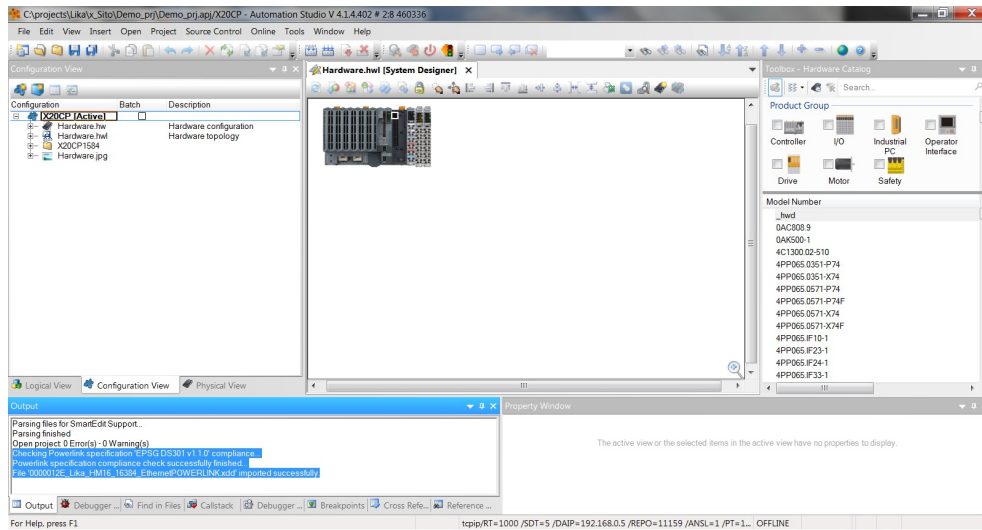


Figure 5 - XDD file installation

The installed device will be listed in the **Toolbox – Hardware Catalog** window (at the top right corner in the snapshot above, see Figure 5).



NOTE

To display the **Hardware.hwl [System Designer]** window in the main page, double click the **Hardware.hwl** item in the **Configuration View** window.

To display the **Toolbox – Hardware Catalog** window in the main page, right click in the **Hardware.hwl [System Designer]** window and press the **Add Hardware Module...** command.

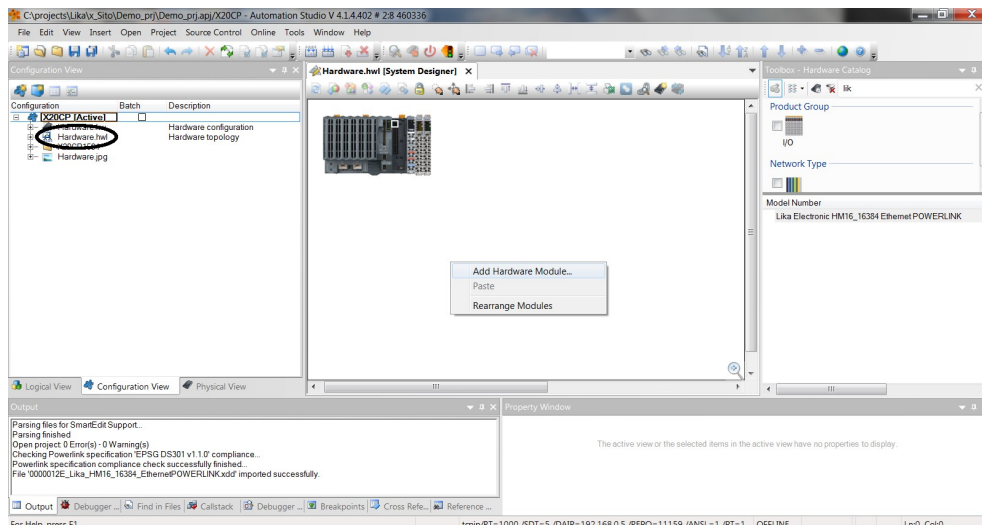


Figure 6 - Adding an hardware module

Now you need to link the installed encoder (that is: the module) to the PLC.
There are two ways.

1. In the **Toolbox – Hardware Catalog** window, enter the name of the installed device (or just "Lika") in the **Search** box; all Lika modules that have been installed will be listed in the **Model Number** window.
2. Or check the **Powerlink** option in the **Network Type** section of the **Toolbox – Hardware Catalog** window and then scroll through the list of the installed Powerlink devices in the **Model Number** window.

Select the required Lika CN device in the **Model Number** window (for instance, the "Lika Electronic HM16_16384 Ethernet POWERLINK" module); a preview will appear in the **Properties** window.

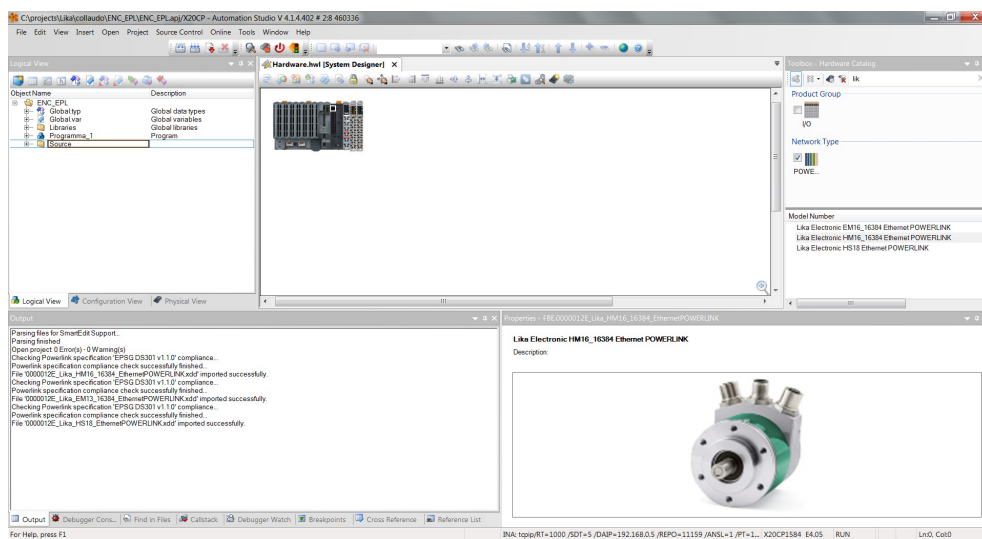


Figure 7 - Adding a module

Drag the module to the **Hardware hwl [System Designer]** window and drop it to the desired position.
 The Lika device icon will appear in the **Hardware hwl [System Designer]** window.

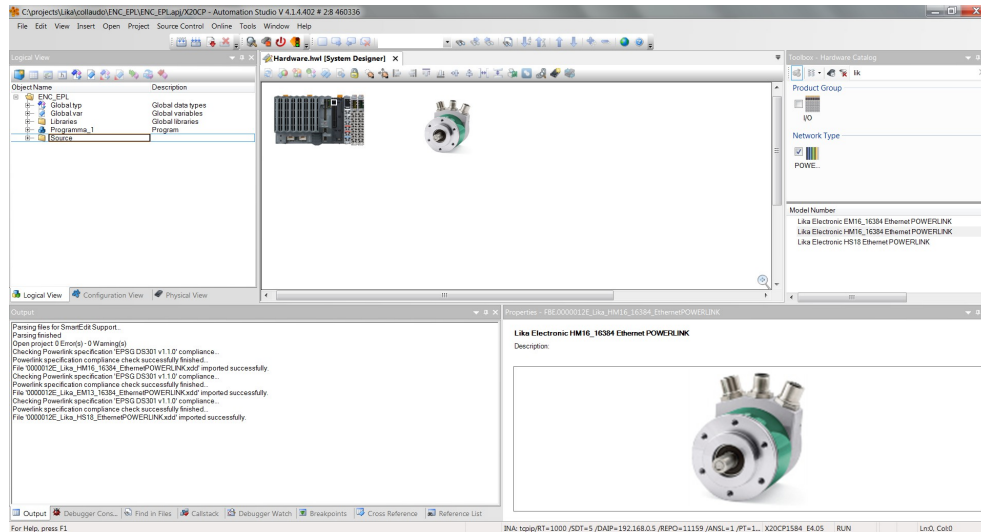


Figure 8 - Installed module

To link the MN device to the CN device, move the cursor over the MN icon; a small circle will appear on the icon; drag it (a blue line will appear) and drop it onto the small circle that is shown when you move the cursor over the CN icon, as shown in the snapshot below, see Figure 9.

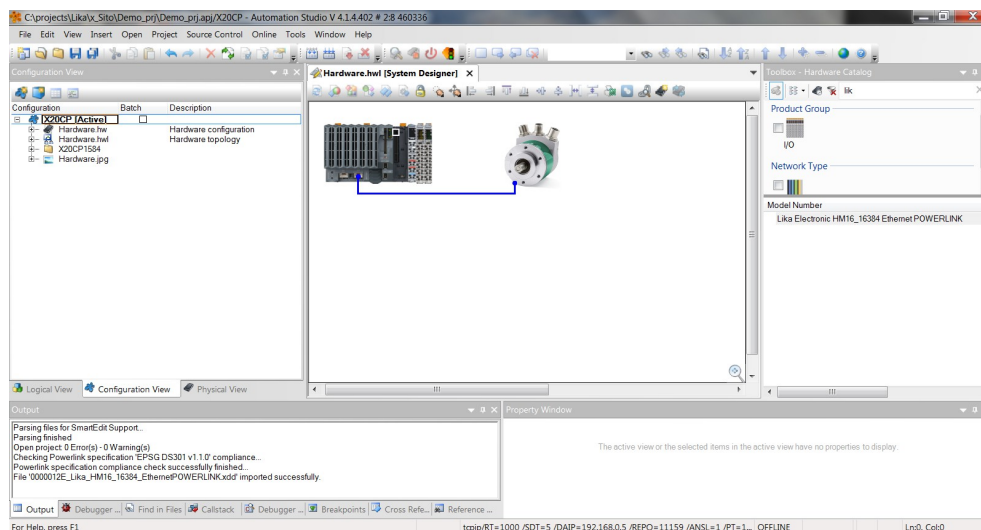


Figure 9 - Adding the module to the network

5.4.3 Setting the device node address in the project

You are required to enter in the project the node address that has been set physically in the installed CN device. For information on the POWERLINK Node ID setting refer to the "4.8 POWERLINK Node ID: DIP A" section on page 32. Enter the **Physical View** window by pressing the **Physical View** tab.

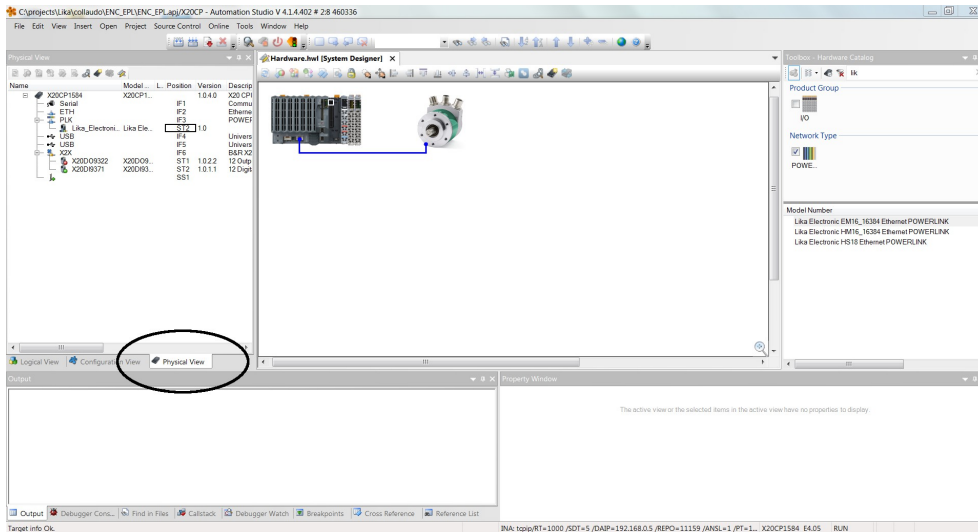


Figure 10 - Physical View window

Scroll through the tree of the installed devices, extend the PLK (Powerlink) group and select the module (for instance: **Lika_Electronic_HM16...**). Right-click the **Lika_Electronic_HM16...** module in the list and then press the **Change Node Number** command.

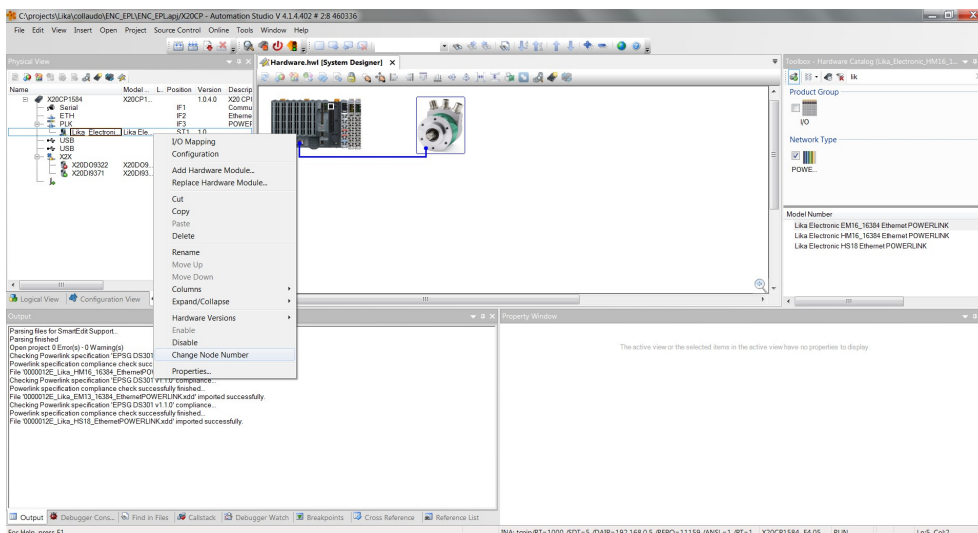


Figure 11 - Setting the node address

Double click the field in the **Position** column and set the required address value in the box.

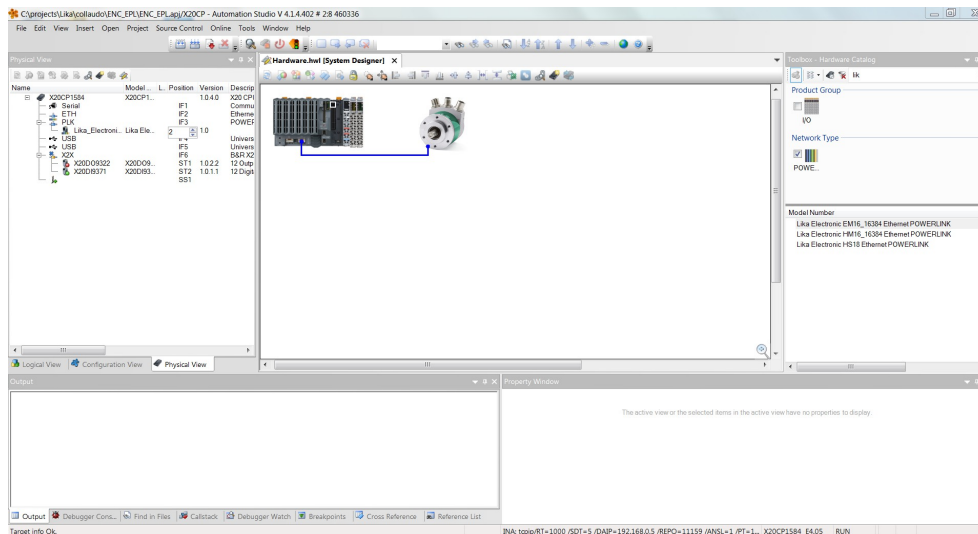


Figure 12 – Setting the node address

5.4.4 Configuring the CN device

The Configuration objects are grouped in the **Object Dictionary** (refer also on page 72).

The Object Dictionary is the most important part of a device profile. It is essentially a grouping of objects accessible via the network in an ordered, pre-defined fashion. Each object within the dictionary is addressed using a 16-bit index.

The Object Dictionary can contain a maximum of 65,536 entries.

The user-related objects are grouped in three main areas: the Communication Profile Area, the Manufacturer Specific Profile Area and the Standardised Device Profile Area. The objects are all described in the XDD file.

The **Communication Profile Area** at indexes from 1000h to 1FFFh contains the communication specific parameters for the POWERLINK network. These entries are common to all devices. The Communication Profile Area objects comply with the "CiA Draft Standard Proposal 301 CANopen Application layer and communication profile". Refer to the "6.12.1 Communication Profile Area objects (DS 301)" section on page 74.

The **Manufacturer Specific Profile Area** at indexes from 2000h to 5FFFh is free to add manufacturer-specific functionality. Refer to the "6.12.2 Manufacturer Specific Profile Area objects" section on page 90.

The **Standardised Device Profile Area** at indexes from 6000h to 9FFFh contains all data objects common to a class of devices that can be read or

written via the network. The device profiles may use entries from 6000h to 9FFFh to describe the device parameters and the device functionality. The Standardised Device Profile Area objects comply with the "CiA Draft Standard 406 CANopen Device profile for encoders". Refer to the "6.12.3 Standardised Device Profile Area objects (DS 406)" section on page 91.

Right click the Lika encoder icon and press the **Configuration** button to enter the encoder **Configuration** window.

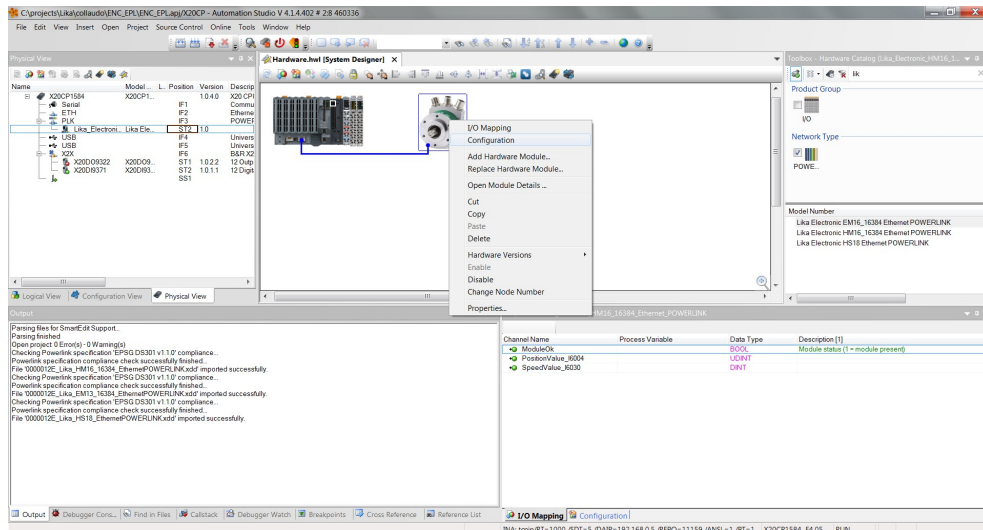


Figure 13 - Entering the Configuration window

In the **Configuration** window all the implemented objects are listed.

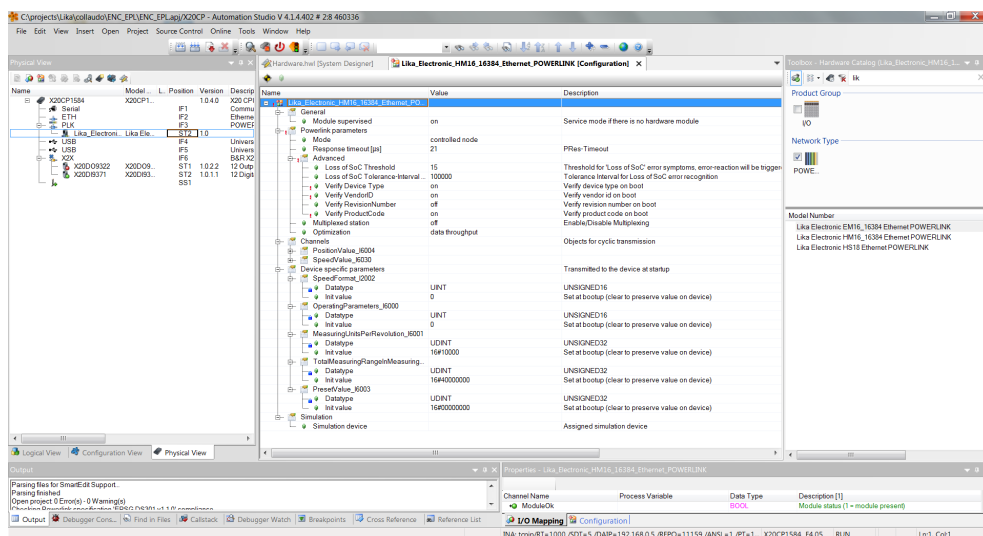


Figure 14 - Configuration window

Under the **Powerlink Parameters** group the Communication Profile Area objects are found.

Under the **Channels** group the **Position** and **Speed** values can be found.

Under the **Device Specific Parameters** group the Manufacturer Specific Profile Area objects and the Standardised Device Profile Area objects are found.

In the **Powerlink Parameters** group we suggest setting the following items to ON:

- **Verify Device Type:** the Device Type compliance is checked on boot;
- **Verify VendorID:** the VendorID compliance is checked on boot;
- **Verify ProductCode:** the Product Code compliance is checked on boot.

This is useful to avoid mismatch errors. See the Figure 15.

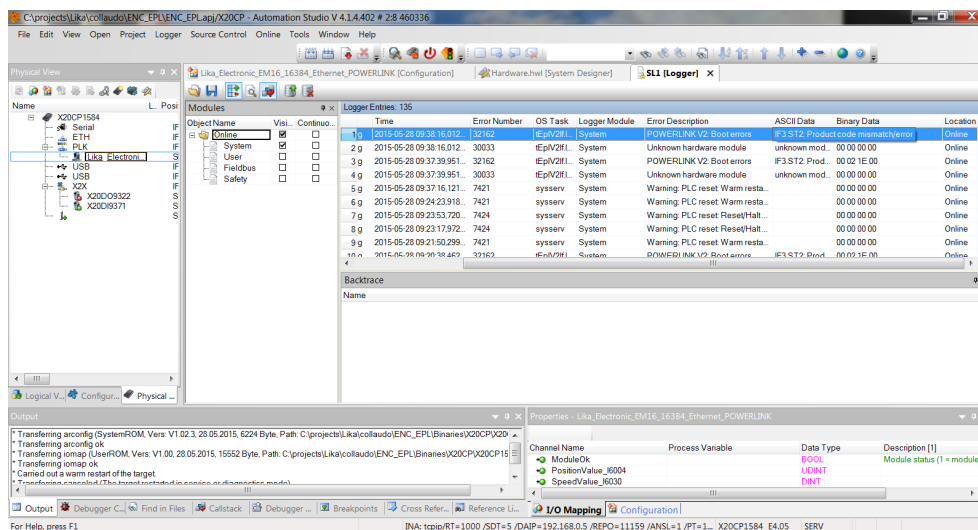


Figure 15 – Mismatch error

5.4.5. Downloading the parameters to the encoder

After having set the parameters in the **Configuration** window, you must download the new values to the encoder to make them effective.

To do this you must press the **Build** button in the Toolbar.

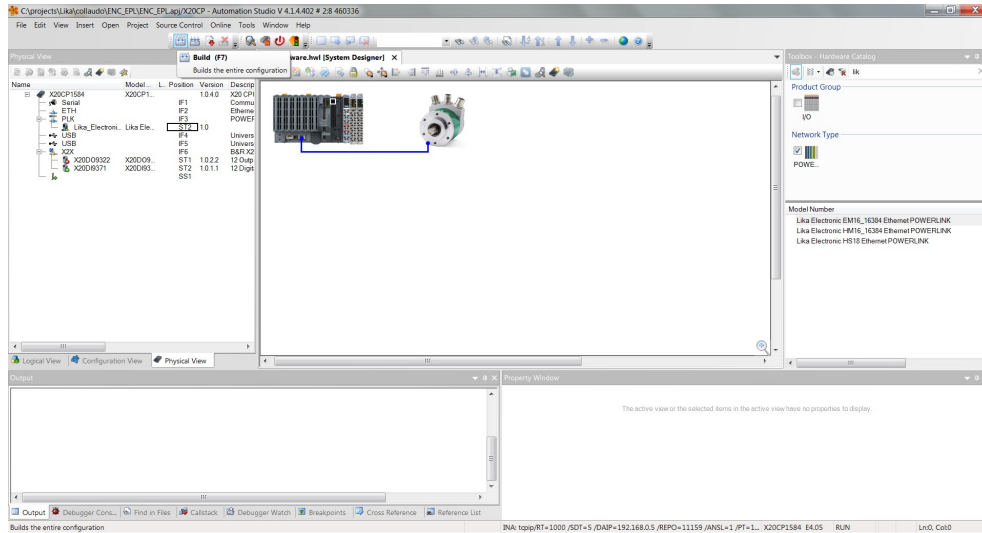


Figure 16 - Downloading the parameter values

As soon as the download is complete, the **Transfer Project** message appears on the display.

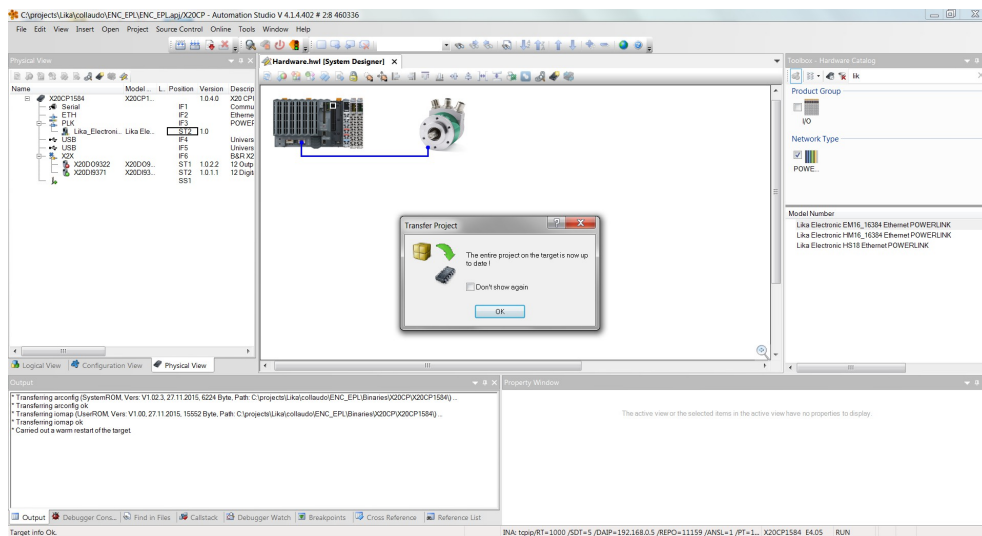


Figure 17 - Download completed

5.4.6. Download of parameter values at PLC start



NOTE

Please consider attentively this PLC behaviour.

During the initialisation process at the PLC start, B&R PLC downloads to the encoder the parameter values that have been set next to the entries in the **Device Specific Parameters** group.

All values that are set in the **Init value** line of each parameter are sent to the control node and overwrite the values currently saved.

To avoid transferring a value and overwriting the parameter stored in the encoder memory, the **Init value** line under the entry in the **Device Specific Parameters** group has to be kept blank.

In the example shown in Figure 18, the **6003 Preset value** next to the **Init value** line is blank, thus it is not transferred to the encoder during the initialisation process at the PLC start. The preset stored in the encoder memory is preserved.

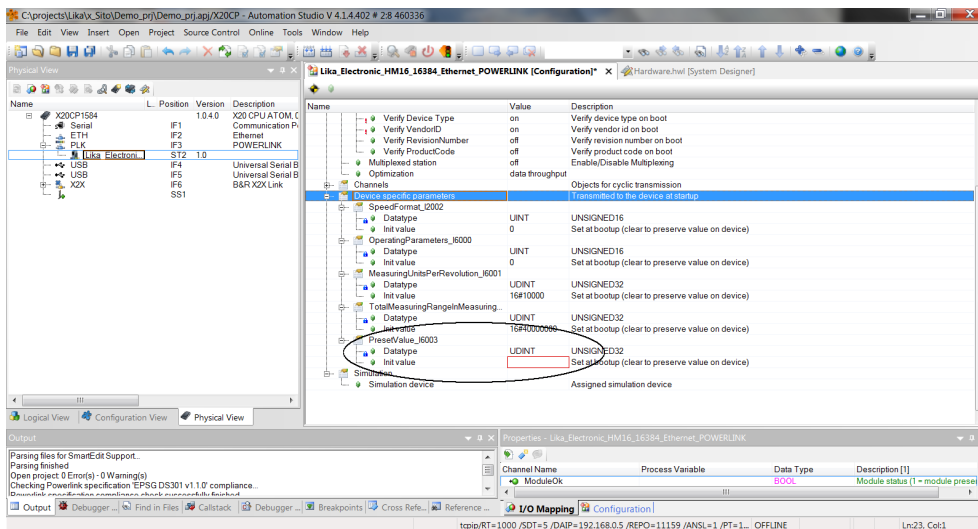


Figure 18 – Parameter values download at PLC start

5.4.7 Monitoring the position and speed values

To monitor the position and speed values of the encoder, you must enter the **I/O Mapping** window first. Right click the Lika encoder icon and press the **I/O Mapping** button to enter the **I/O Mapping** window.

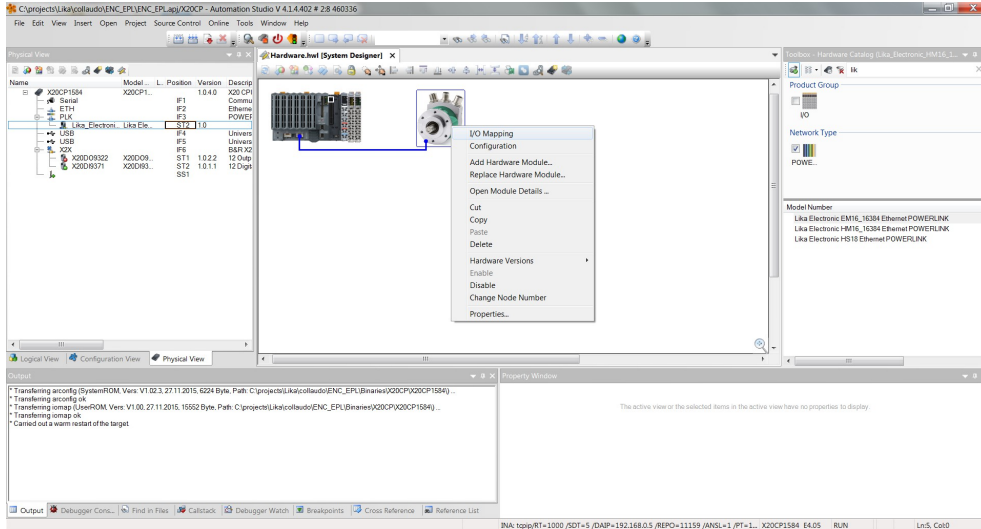


Figure 19 - Entering the I/O Mapping window

You can display the **I/O Mapping** window also by scrolling through the tree of the installed devices in the **Physical View** window, extending the **PLK** (Powerlink) group and selecting the Lika module (for instance: **Lika_Electronic_HM16...**). Right-click the **Lika_Electronic_HM16...** module in the list and then press the **I/O Mapping** command.

Now press the **Monitor** button in the Toolbar to activate the Monitor mode and start monitoring the system.

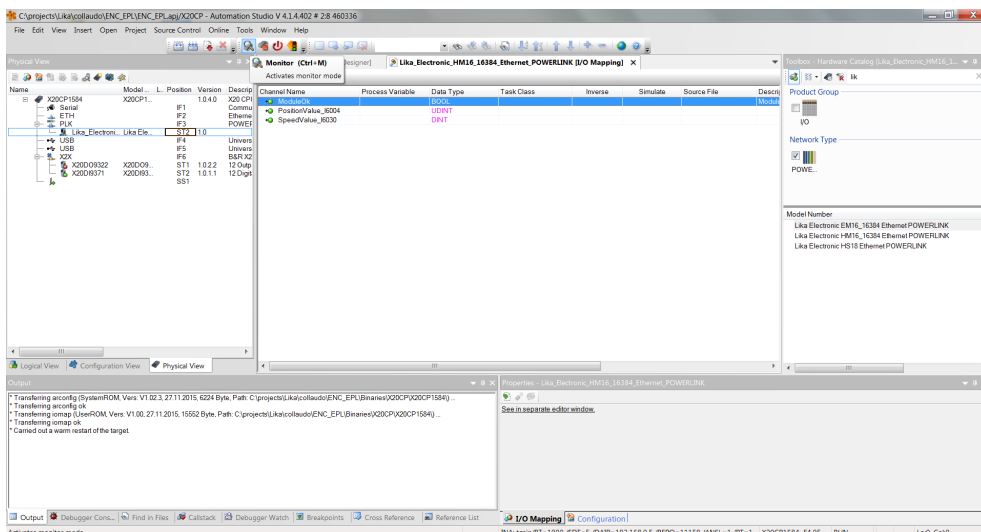


Figure 20 - Monitoring the device

As soon as the system is in Monitor mode, the background of the windows becomes grey.

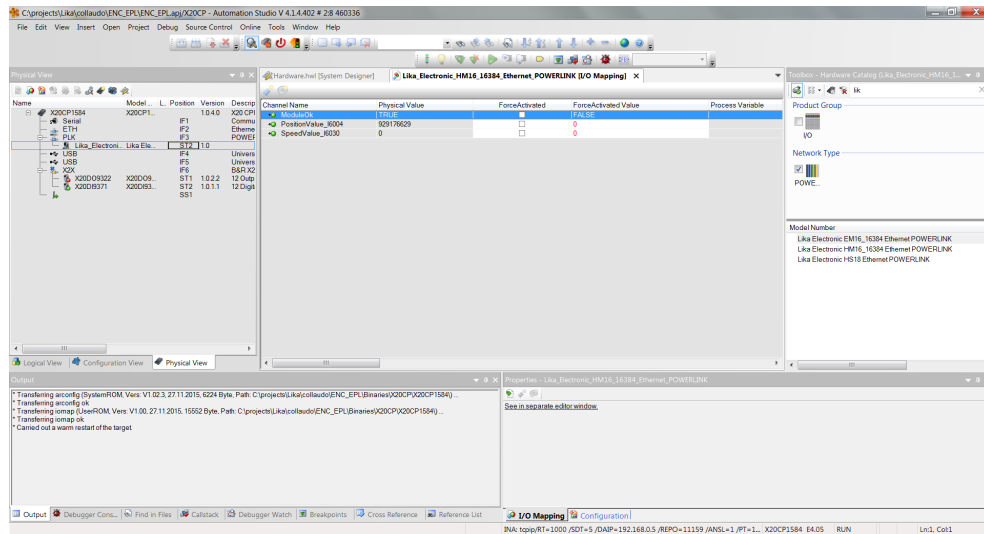
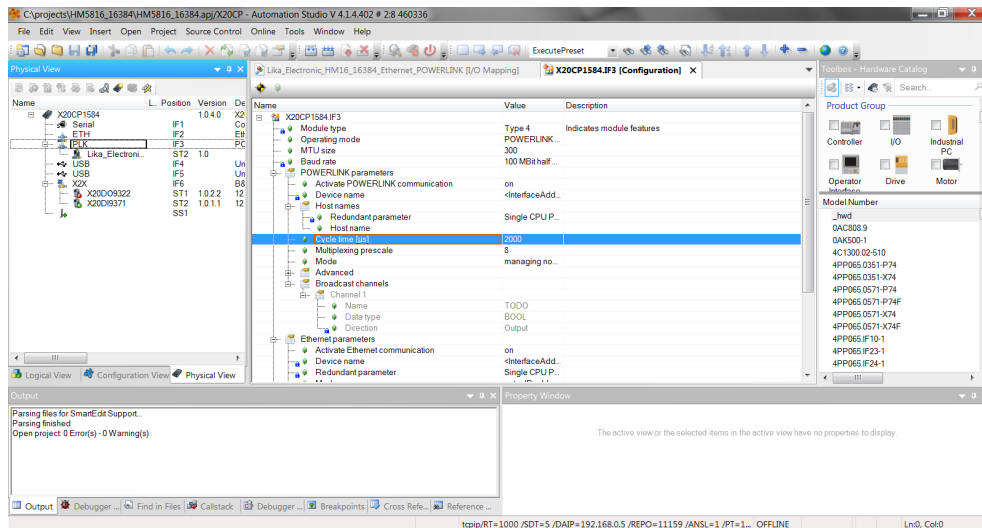


Figure 21 - Monitoring the device

The **6004 Position value** and the **6030 Speed value** objects are displayed in the **I/O Mapping** window and can be monitored.

5.4.8 Setting the cycle time

In a POWERLINK network, the managing node allocates data transfer time for data from each node in a cyclic manner within a guaranteed cycle time. The Cycle Time is the time between two consecutive Start of Cyclic (SoC) frames – i.e. repeating – process. The Cycle Time includes the time for data transmission and some idle time before the beginning of the next cycle. See the [1006 NMT_CycleLen_U32](#) object on page 74.



5.4.9 Preset setting program

As previously stated (see the "5.4.6. Download of parameter values at PLC start" section on page 48), during the initialisation process at the PLC start, B&R PLC downloads to the encoder the parameter values that have been set next to the entries in the **Device Specific Parameters** group.

If any value is set in the **Init value** line of the **PresetValue_I6003** parameter, it is sent to the control node.

To execute the preset the value must be transmitted to the **6003 Preset value** object in the asynchronous phase via SDO when the encoder is in operational state (**NMT_CS_OPERATIONAL**) and then execute the **Store parameters** function (see the **Store parameters** function on page 92).

You can set and execute the preset also by using the Preset sample task provided by Lika Electronic (AsEPL Sample Task using the AsEPL library available in Automation Studio).

To open the Preset sample task you must enter the **Logical View** window by pressing the **Logical View** tab.

Scroll through the tree of the available EPLs, extend the **Source** directory and select the **Preset** AsEPL Sample Task. Right-click the **Preset** AsEPL Sample Task in the list, press the **Open** command first and then the **Watch** command.

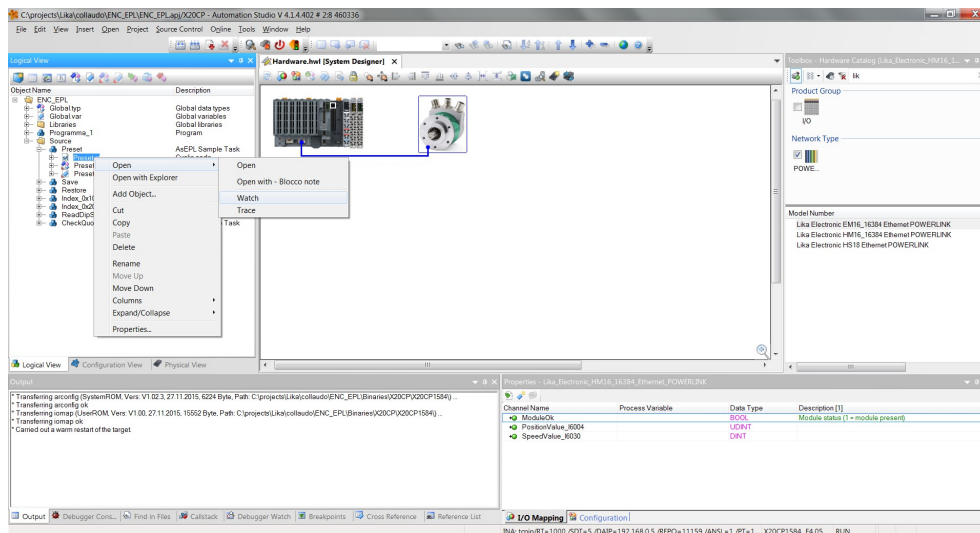


Figure 22 - Opening the Preset AsEPL Sample Task

The **Preset.pvm [Watch]** window will be displayed.
 Activate the Monitor mode by pressing the **Monitor** button in the Toolbar.

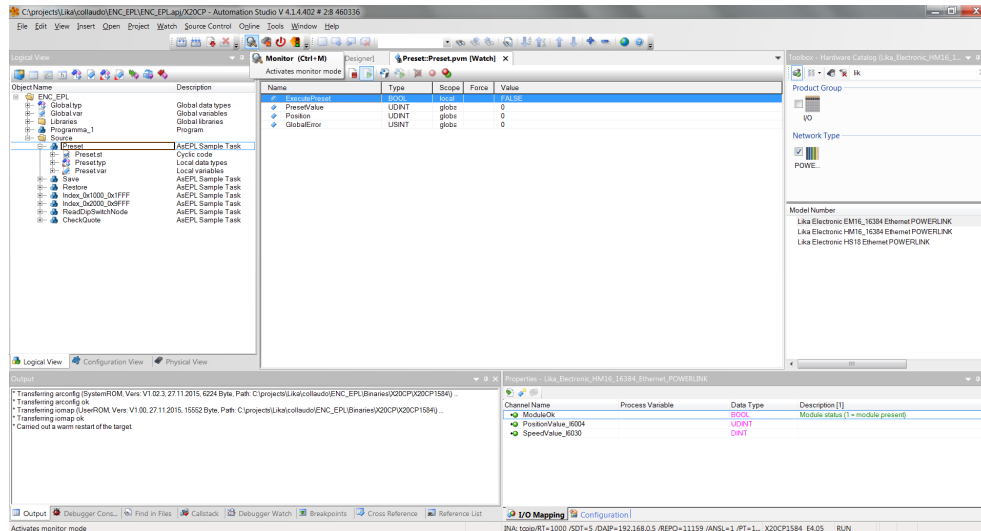


Figure 23 - Preset.pvm [Watch] window

The background of the windows becomes grey.

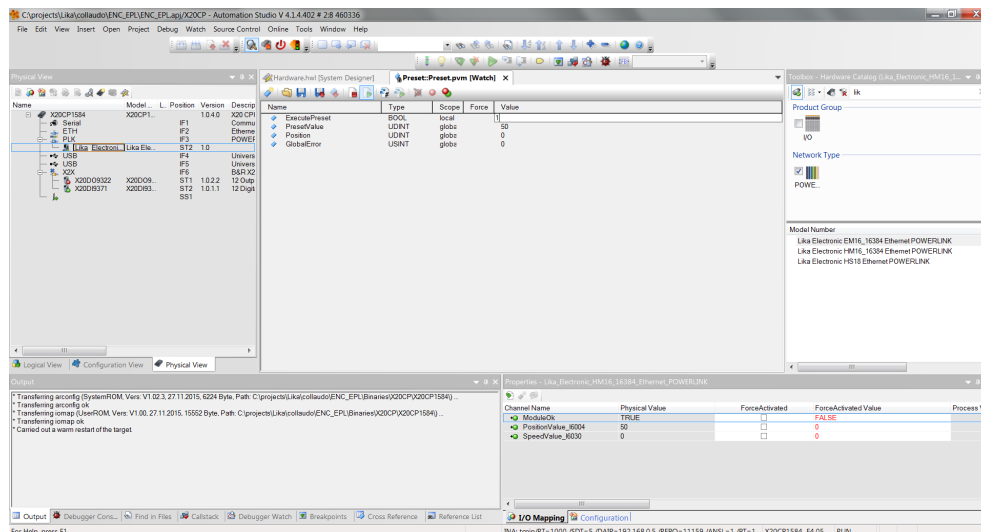


Figure 24 - Activating the Monitor mode

Enter the preset value next to the **PresetValue** entry ("50" in the sample shown in the snapshot, see Figure 24). Set the value "1" (TRUE) next to the **ExecutePreset** entry (see Figure 24) and then set it again to "0" (FALSE).

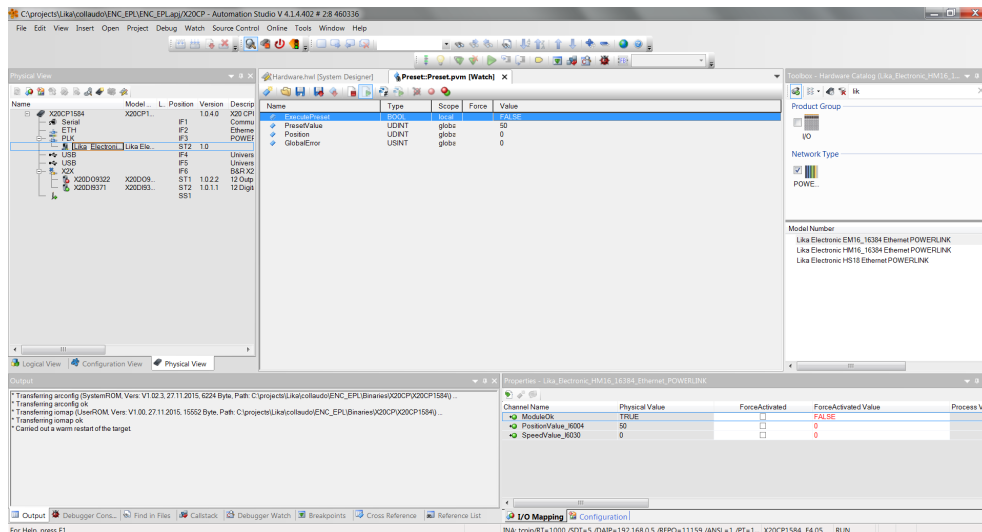


Figure 25 – Preset activated

You can enter the **I/O Mapping** window (see the "5.4.7 Monitoring the position and speed values" section on page 49) and check that the **PositionValue_I6004** entry is "50".

After executing the preset command, you are required to save the parameters in order to store on memory the calculated offset. If you do not save the new offset value, at next power up the system will calculate the encoder position using the offset value previously saved. To save the parameters refer to the following "5.4.10 Parameters storing program" section on page 55.

5.4.10 Parameters storing program

To store the parameters you must set the bit 15 **Store parameters** in the **6000 Operating parameters** object in the asynchronous phase via SDO when the encoder is in operational state (**NMT_CS_OPERATIONAL**).

You can execute the parameters storing function also by using the Save sample task provided by Lika Electronic (AsEPL Sample Task using the AsEPL library available in Automation Studio).

To open the Save sample task you must enter the **Logical View** window by pressing the **Logical View** tab.

Scroll through the tree of the available EPLs, extend the **Source** directory and select the **Save** AsEPL Sample Task. Right-click the **Save** AsEPL Sample Task in the list, press the **Open** command first and then the **Watch** command.

The **Save.pvm [Watch]** window will be displayed.

Activate the Monitor mode by pressing the **Monitor** button in the Toolbar.

The background of the windows becomes grey.

Set the value "1" (TRUE) next to the **bit_save** entry; set the value "1" (TRUE) next to the **write_bit_save** entry; it will be automatically set back to "0" (FALSE); now set the value "0" (FALSE) next to the **bit_save** entry; set the value "1" (TRUE) next to the **write_bit_save** entry; it will be automatically set back to "0" (FALSE).

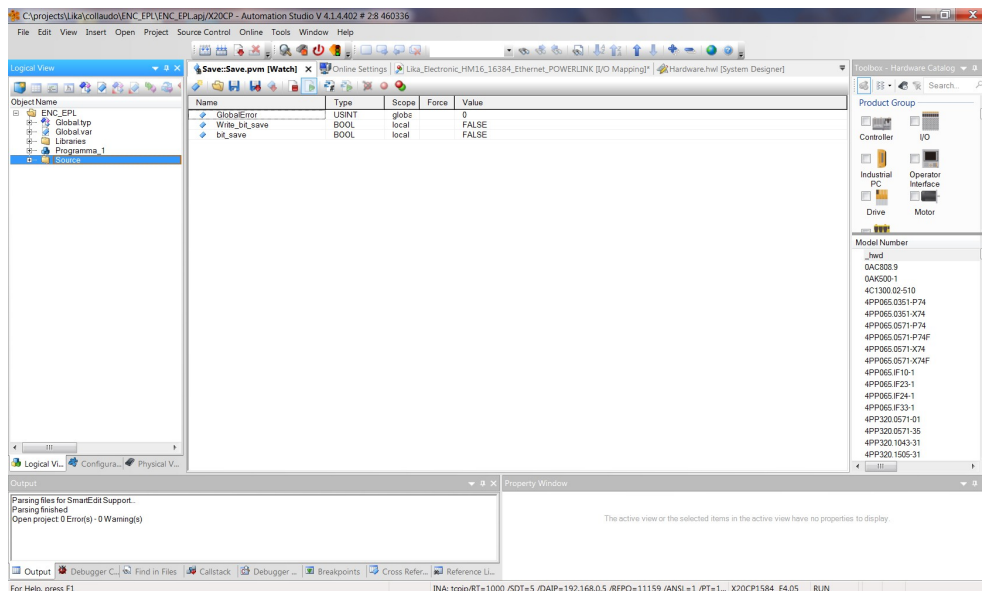


Figure 26 – Save AsEPL Sample Task

5.4.11 Default parameters restoring program

Default values are provided to each parameter of the device and are preset at the factory by Lika Electronic engineers. The first time you install the encoder, it will operate using the default values. They allow the operator to run the CN device for standard and safe operation. They are plainly not optimized for specific application yet they provide maximum performance for most systems. To suit the specific application requirements it may be advisable and even necessary to enter new parameters instead of the factory default settings.

There could be exceptional circumstances where it would be necessary for you to restore the default values of the settable parameters. When this is the case, you must set the bit 14 **Restore default parameters** in the **6000 Operating parameters** object in the asynchronous phase via SDO when the encoder is in operational state (**NMT_CS_OPERATIONAL**) and then execute the **Store parameters** function (see the **Store parameters** function on page 92).

You can execute the default parameters restoring function also by using the Restore sample task provided by Lika Electronic (AsEPL Sample Task using the AsEPL library available in Automation Studio).

To open the Restore sample task you must enter the **Logical View** window by pressing the **Logical View** tab.

Scroll through the tree of the available EPLs, extend the **Source** directory and select the **Restore** AsEPL Sample Task. Right-click the **Restore** AsEPL Sample Task in the list, press the **Open** command first and then the **Watch** command.

The **Restore.pvm [Watch]** window will be displayed.

Activate the Monitor mode by pressing the **Monitor** button in the Toolbar.

The background of the windows becomes grey.

Set the value "1" (TRUE) next to the **bit_restore** entry; set the value "1" (TRUE) next to the **write_bit_restore** entry; it will be automatically set back to "0" (FALSE); now set the value "0" (FALSE) next to the **bit_restore** entry; set the value "1" (TRUE) next to the **write_bit_restore** entry; it will be automatically set back to "0" (FALSE).



NOTE

When you restore the default values, please always consider that:

- the encoder parameters will be restored to the default values;
- the encoder offset will be reset.



WARNING

The execution of this command causes all the values which have been set previously next to each parameter to be overwritten!

**NOTE**

The complete list of machine data and relevant default parameters preset by Lika Electronic engineers is available on page 104.

5.4.12 Entering the System Diagnostics Manager (SDM)

Automation Studio provides the user with a wide variety of diagnostics tools for commissioning applications and searching for errors:

- System Diagnostics Manager (SDM);
- Status bar;
- Logger Monitor of the software and hardware configuration, see on page 60;
- ...

These tools range from simple monitoring of operating states (variable and program status, I/O, etc.), forcing I/O channels, and tracing variable states over time to profiling the entire runtime behaviour, debugging operations for programs and libraries, and simulating and commissioning axes.

The **System Diagnostics Manager (SDM)** is a diagnostic tool used to diagnose the system by means of a standard web browser from any location (Intranet or Internet).

It only requires a web browser and a TCP/IP connection to the controller.

It can be used for:

- hardware analysis for detection of configuration or hardware problems on the target system;
- analysis of system configuration and runtime parameters (e.g. configured IP address, etc.);
- software analysis (software modules and versions on the target system);
- display and upload of error logbook for target system.

The user profits not just from being able to access information about system hardware and software from anywhere in the world, but also from ready-made diagnostic applets that can be easily integrated directly into applications.

Since all service functions necessary for a machine or system are already integrated in the System Diagnostics Manager, it can be launched from any PC without any installation whatsoever. The only requirement is a conventional web browser.

To enter the System Diagnostics Manager, press **Tools** in the menu bar of the main page and then the **System Diagnostics Manager** command.

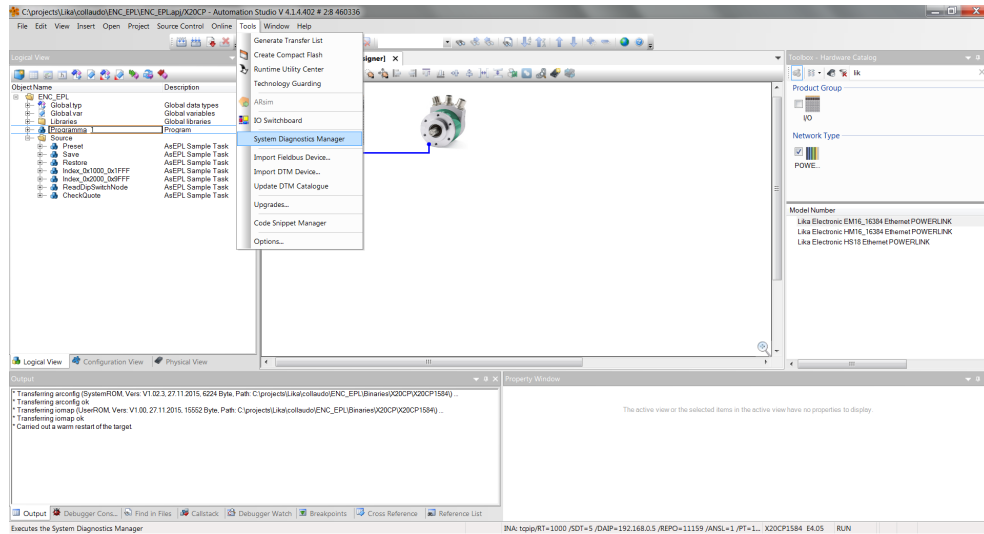


Figure 27 - Entering the System Diagnostics Manager

The first page of B&R System Diagnostics Manager will appear in your predefined web browser.

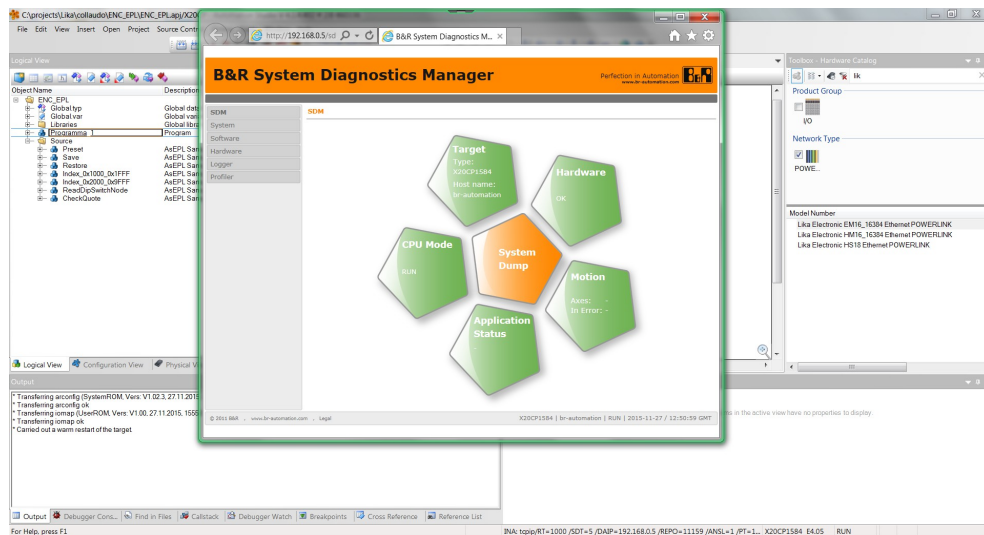


Figure 28 - System Diagnostics Manager

Press either the buttons in the left navigation bar or the graphic figures to enter the specific pages.

Press the **Hardware** button in the left navigation bar to collect some diagnostic information on the Lika module.

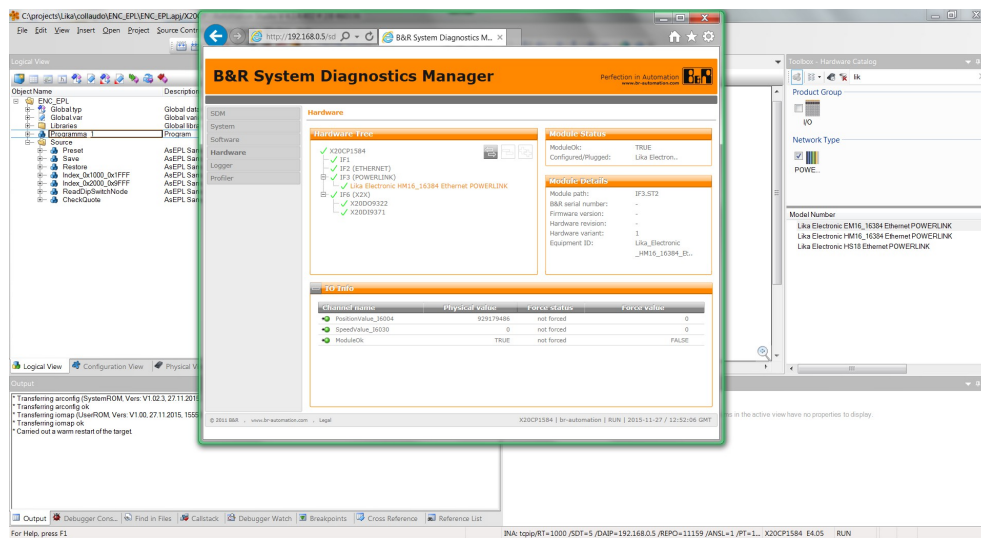


Figure 29 - Hardware diagnostic page

5.4.13 Logger Monitor

Among the diagnostics tools provided by Automation Studio is the Logger Monitor.

It is used to record system information, the system messages are automatically entered in the list in the **SL1 [Logger]** window.

To enter the Logger Monitor window, press **Open** in the menu bar of the main page and then the **Logger** command.

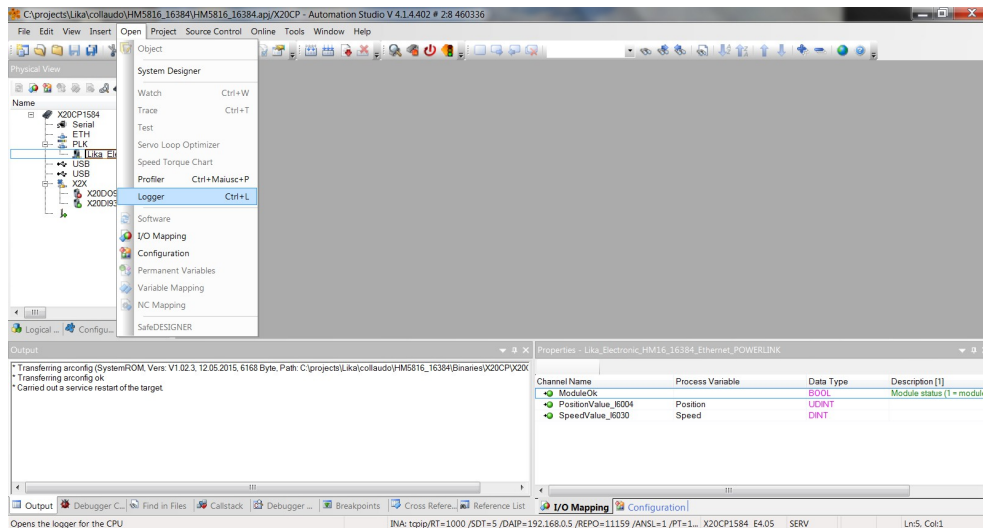


Figure 30 - Entering the Logger Monitor

The **SL1 [Logger]** window will be displayed.

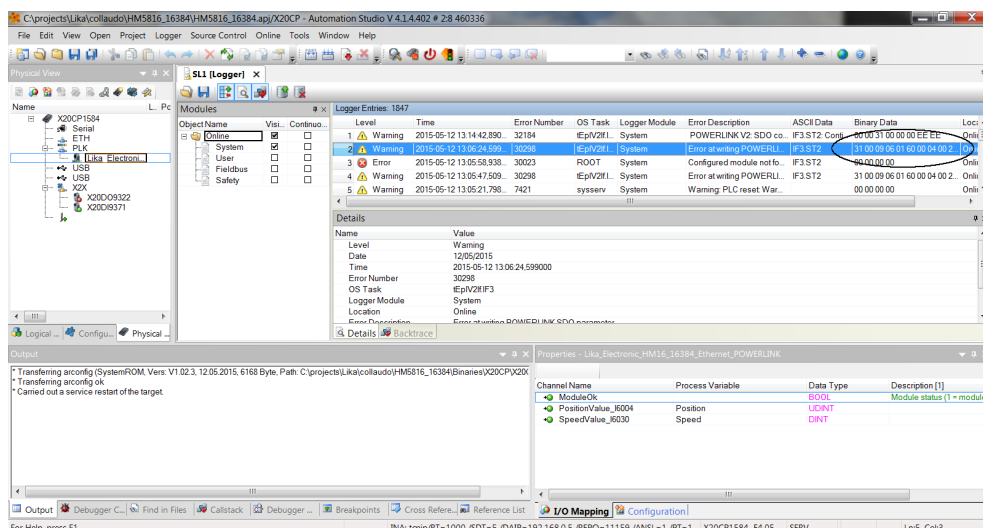


Figure 31 - Logger entries

In the highlighted logger entry the SDO abort code is shown in the Binary Data column (6090031h abort code: "Value of parameter written too high", for the complete list of the SDO abort codes see on page 102).

6 POWERLINK interface

6.1 Overview

Ethernet POWERLINK (EPL) is a communication profile for Real-Time Ethernet (RTE). It extends Ethernet according to the IEEE 802.3 standard with mechanisms to transfer data with predictable timing and precise synchronisation. The communication profile meets timing demands typical for high-performance automation and motion applications. It does not change basic principles of the Fast Ethernet Standard IEEE 802.3 but extends it towards RTE. Thus it is possible to leverage and continue to use any standard Ethernet silicon, infrastructure component or test and measurement equipment like a network analyser.

EPL was originally designed by B&R GmbH and the first version was released in 2001. Later on, the Ethernet POWERLINK Standardization Group (EPSG) was in charge of continuing its management and published the first EPSG Draft version of EPL as an open standard in 2003 fostering for a free distribution of its specifications as well as for the open-source distribution of the protocol source code. The current version of this communication standard can actually be downloaded from the EPSG web site.

EPL is able to provide the real-time capabilities required by critical processes, control tasks and management functions typical of the industrial scenario. As a matter of fact, which is compliant with the Industrial Automation Open Network Alliance (IAONA) real-time class 4 (highest performance) recommendations, it is able to cope with communication cycles in the order of hundreds of microseconds, ensuring, at the same time, jitters below 1 μ s.

Also, the EPL application layer is based on the popular and settled CANopen standard (practitioners often refer to EPL as "CANopen over Ethernet", see below on page 63). This feature ensures, at high layer of the protocol stack, compatibility with several other industrial communication devices.

POWERLINK provides mechanisms to achieve the following aims:

1. transmit time-critical data in precise isochronous cycles. Data exchange is based on a publish/subscribe relationship. Isochronous data communication can be used for exchanging position data of motion applications of the automation industry;
2. synchronise networked nodes with high accuracy;
3. transmit less time-critical data asynchronously on request. Asynchronous data communication can be used to transfer IP-based protocols like TCP or UDP and higher layer protocols such as HTTP, FTP, etc.

POWERLINK manages the network traffic in a way that there are dedicated time-slots for isochronous and asynchronous data. It takes care that always only one networked device gains access to the network media. Thus transmission of isochronous and asynchronous data will never interfere and precise communication timing is guaranteed. The mechanism is called Slot

Communication Network Management (SCNM). SCNM is managed and supervised by exactly one particular networked device – the Managing Node (MN, e.g. the Master) – which includes the MN functionality. All other nodes (up to 240) are called Controlled Nodes (CN, e.g. the Slave) and are deployed in various topologies (networks may have a star, tree, daisy chain or ring structure, or any combination of these topologies). Find further information in the "6.5 POWERLINK nodes" section on page 63.

6.2 Physical layer

POWERLINK is a protocol residing on top of the standard IEEE 802.3 MAC layer. The physical layer is 100Base-X (copper and fiber, see IEEE 802.3). Half-Duplex transmission mode is used.

POWERLINK uses Ethernet as it is, without any modifications. Hence any advancement in Ethernet Technology can be exploited (e.g. Gigabit Ethernet).

To increase noise immunity only S/FTP or SF/FTP cables must be used (CAT-5). The maximum cable length (100 meters) predefined by Ethernet 100Base-TX must be compulsorily fulfilled.

EPL recommends the use of hubs to fit POWERLINK jitter requirements.

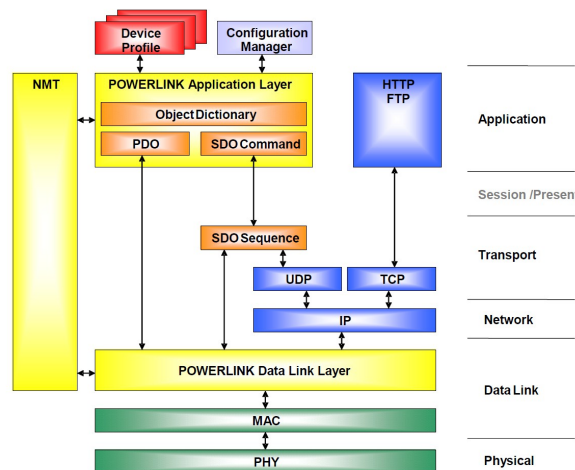
Switches may be used to build a POWERLINK network.

It has to be considered that any POWERLINK network constructed with anything but Class 2 Repeater Devices does not conform to the POWERLINK standard.

Refer also to the "Electrical connections" section on page 29.

6.3 Reference model

POWERLINK-based networks use the following reference model.



6.4 CANopen over Ethernet

The Ethernet POWERLINK Standardisation Group (EPSG) is working closely with the CiA (CAN in Automation) organisation to integrate CANopen with POWERLINK. CANopen standards define widely deployed communication profiles, device profiles and application profiles. These profiles are in use millions of times all over the world. Integration of POWERLINK with CANopen combines profiles, high performance data exchange and open transparent communication with TCP/UDP/IP protocols.

The POWERLINK communication and device profiles implemented in the encoder are based on CANopen communication profile DS301 and device profile DS406.

6.5 POWERLINK nodes

The node managing the permission to send messages to the Ethernet is called the POWERLINK Managing Node (MN).

All other nodes transmit only within communication slots assigned by the MN. They are thus called Controlled Nodes (CN).

6.5.1 POWERLINK Managing Node (MN)

Only the MN is allowed to send messages independently – i.e. not as a response to a received message. Controlled Nodes are only allowed to send when requested to by the MN.

The Controlled Nodes are accessed cyclically by the MN. Unicast data are sent from the MN to each configured CN (frame: PReq), which then publishes its data via multicast to all other nodes (frame: PRes).

All available nodes in the network are configured on the MN.

Only one active MN is permitted in a POWERLINK network.

6.5.2 POWERLINK Controlled Node CN)

CNs are passive bus nodes. They only send when requested by the MN.

The ability of a node to perform CN functions is indicated by the device description entry **D_DLL_FeatureCN_BOOL**.

Lika encoders are CN devices and comply with the "EPSG Draft Standard 301 Ethernet POWERLINK Communication Profile Specification Version 1.2.0". as well as with the CANopen Profiles "DS301 CANopen Application Layer and Communication Profile" and "DS406 Device Profile for Encoders" according to the POWERLINK specifications.

6.6 POWERLINK Basic Frame

The POWERLINK Basic Frame format contains 5 fields:

- Reserved (1 bit)
- MessageType (7 bits)
- Destination node address (1 byte)
- Source node address (1 byte)

- Data (n bytes)

The POWERLINK Basic Frame format is encapsulated by the Ethernet II wrapper consisting of 14 bytes of leading Ethernet header (Destination and Source MAC addresses, EtherType) and 4 bytes of terminating CRC32 checksum.

Allowed frame sizes are ranging between 64 bytes to 1518 bytes, thus the POWERLINK frame can be between 46 bytes and 1500 bytes (see the [1030-04 NMT_InterfaceGroup_0h_REC.InterfaceMtu_U16](#) object on page 77).

Byte offset	Bit offset								Entry defined by
	7	6	5	4	3	2	1	0	
0 ... 5	Destination MAC Address								Ethernet Type II
6 ... 11	Source MAC Address								
12 ... 13	EtherType (0x88AB)								
14	res	MessageType							Ethernet POWERLINK
15	Destination (8 bit node ID)								
16	Source (8 bit node ID)								
17 ... n	Data								
n+1 ... n+4	CRC32								Ethernet Type II

POWERLINK is identified via the EtherType 88ABh.

6.7 Message types

6.7.1 Start of Cycle (SoC)

Start of Cycle (SoC, ID = 01h) is a broadcast frame sent by the MN to begin the POWERLINK cycle (see the "6.8 POWERLINK Cycle" section on page 65). To maintain a fixed cycle time, the SoC frame is issued on a precise periodic basis, keeping jitter on it as low as possible; this also serves to the purpose of providing time synchronisation for all the devices.

6.7.2 PollRequest (PReq)

After the SoC frame has been issued the Isochronous Period is entered (see the "6.8.1 Isochronous Period" section on page 65). In this key part of the cycle, the MN polls each CN by means of the PollRequest (PReq, ID = 03h) frame which is sent only to the selected CN and carry output data for it.

6.7.3 PollResponse (PRes)

Following a PollRequest (PReq) the accessed CN responds to the query by issuing a PollResponse (PRes, ID = 04h) frame which is instead a multicast frame carrying input data, made available to all nodes in the network.

6.7.4 Start of Asynchronous (SoA)

Once the Isochronous Period is concluded, the MN sends a broadcast frame called Start of Asynchronous (SoA, ID = 05h) which informs all CNs about the start of the Asynchronous Period (see the "6.8.2 Asynchronous Period" section on page 65). This second phase ensures the transmission of only one asynchronous message by a selected node. The SoA frame is also used to inform which node has been selected for the acyclic communication.

6.7.5 Asynchronous Send (Asnd)

The Asynchronous Send (Asnd, ID = 06h) frame transports asynchronous data via the POWERLINK/ASnd protocol (e.g. NMT commands).

6.8 POWERLINK Cycle

Data exchange within a POWERLINK network is structured in fix intervals, called POWERLINK cycles. The cycle is subdivided into the Isochronous Period, the Asynchronous Period and the Idle Period and is managed by the MN.

6.8.1 Isochronous Period

The Isochronous Period of a POWERLINK cycle offers deterministic operation, i.e. it is reserved for the exchange of (continuous or multiplexed) isochronous data. Isochronous data exchange between nodes occurs cyclically. It is repeated in a fixed interval, called the POWERLINK cycle. The POWERLINK cycle is controlled by the MN.

At the beginning of a POWERLINK cycle, the MN sends a SoC (Start of Cycle, see on page 64) frame to all nodes via Ethernet multicast. The send and receive time of this frame is the basis for the common timing of all the nodes.

Only the SoC frame is generated on a periodic basis. The generation of all other frames is event controlled (with additional time monitoring per node).

The MN starts the isochronous data exchange after the SoC frame has been sent. A PReq frame (see on page 64) is sent to every configured and active node. The accessed node respond by a PRes frame (see on page 64).

6.8.2 Asynchronous Period

The Asynchronous Period is the second part of the POWERLINK cycle, starting with a Start of Asynchronous (SoA) frame (see on page 65).

In the asynchronous phase of the cycle, access to the POWERLINK network may be granted to one CN or to the MN for the transfer of a single asynchronous message only.

There are two types of asynchronous frames available:

- the POWERLINK ASnd frame uses the POWERLINK addressing scheme and is sent via unicast or broadcast to any other node (see on page 65);
- a Legacy Ethernet message can be sent.

If no asynchronous message transmission request is pending at the MN scheduling queues, the MN issues a SoA without assignment of the right to send to any node. No ASnd frame follows to the SoA frame in this case.

The MN starts the asynchronous phase with the SoA. The SoA is used to identify CNs, request status information of a CN, to poll async-only CNs and to grant the asynchronous transmit right to one CN.

The SoA frame is the first frame in the asynchronous phase and is a signal to all CNs that all isochronous data have been exchanged during the isochronous phase.

The asynchronous phase is calculated from the start of SoA to the end of the asynchronous response. If no asynchronous response is allowed for any node, the asynchronous phase is terminated by the end of SoA.

6.8.3 Idle Period

After both the SoA and the acyclic frames have been transmitted, the Idle Period is entered. All the nodes wait for the new cycle to start, i.e. to receive a new SoC frame from the MN.

6.9 CN Node NMT States

After the Initialization NMT State Machine (it is common to both MN and CNs), the POWERLINK devices enter specific MN and CN states.

Here follows the list of the available CN states.

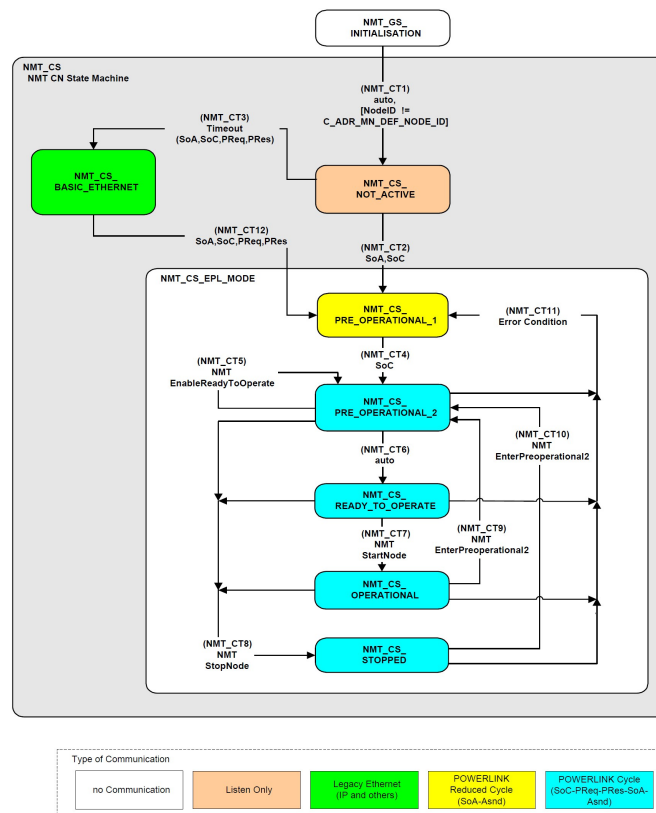


Figure 32 - CN NMT State Machine

6.9.1 NMT_CS_NOT_ACTIVE

NMT_CS_NOT_ACTIVE is a non-permanent state which allows a starting node to recognize the current network state. The MS (Module Status) LED is off. The CN observes network traffic. The node is not authorised to send frames autonomously. There is no Legacy Ethernet frame transmission allowed at **NMT_CS_NOT_ACTIVE**. The node is able to recognize **NMTReset** commands sent via Asnd.

The transition from **NMT_CS_NOT_ACTIVE** to **NMT_CS_PRE_OPERATIONAL_1** is triggered by a SoA or SoC frame being received.

The transition from **NMT_CS_NOT_ACTIVE** to **NMT_CS_BASIC_ETHERNET** is triggered by timeout for SoC, PReq, PRes and SoA frames.

6.9.2 NMT_CS_PRE_OPERATIONAL_1

In the state **NMT_CS_PRE_OPERATIONAL_1**, the CN sends a frame only if the MN has authorised it to do so by a SoA AsyncInvite command. The MS (Module Status) LED gives single green flashes.

In **NMT_CS_PRE_OPERATIONAL_1** the node is identified by the MN via **IdentRequest**. If required the CN downloads its configuration data from a configuration server. Both processes may be completely or partially shifted to **NMT_CS_PRE_OPERATIONAL_2**, if the MN is not in **NMT_CS_PRE_OPERATIONAL_1** or leaves **NMT_CS_PRE_OPERATIONAL_1** before the CN has completed its configuration.

The transition from **NMT_CS_PRE_OPERATIONAL_1** to the following state is triggered by a SoC frame being received.

There is no PDO communication in **NMT_CS_PRE_OPERATIONAL_1**.

6.9.3 NMT_CS_PRE_OPERATIONAL_2

In the state **NMT_CS_PRE_OPERATIONAL_2**, the CN waits for the configuration to be completed. The MS (Module Status) LED gives double green flashes.

The node is queried by the MN via PReq. The received PDO data may be invalid. The PDO data received from the MN via PReq and from other CNs and the MN via PRes is ignored by the CN.

CNs respond to **AsyncInvite** commands via SoA. If not invited by the MN, there is no Ethernet frame transmission allowed at the **NMT_CS_PRE_OPERATIONAL_2** state.

Precondition for the transition from **NMT_CS_PRE_OPERATIONAL_2** to **NMT_CS_READY_TO_OPERATE** is the reception of an **NMTEnableReadyToOperate** command. The transition is triggered if the application is ready for operation.

The transition from **NMT_CS_PRE_OPERATIONAL_2** to **NMT_CS_PRE_OPERATIONAL_1** is triggered by an error recognition.

The transition from **NMT_CS_PRE_OPERATIONAL_2** to **NMT_CS_STOPPED** is triggered by reception of NMT state command **NMTStopNode**.

6.9.4 NMT_CS_READY_TO_OPERATE

With the state **NMT_CS_READY_TO_OPERATE**, the CN signals its readiness to operation to the MN. The MS (Module Status) LED gives triple green flashes.

The node may participate in cyclic frame exchange. Nodes respond via PRes when queried via PReq by the MN.

CNs respond to **AsyncInvite** commands via SoA. If not invited by the MN, there is no Ethernet frame transmission allowed at the **NMT_CS_READY_TO_OPERATE** state.

The transition from **NMT_CS_READY_TO_OPERATE** to **NMT_CS_OPERATIONAL** is triggered by the reception of NMT state command **NMTStartNode**.

The transition from **NMT_CS_READY_TO_OPERATE** to **NMT_CS_PRE_OPERATIONAL_1** is triggered by an error recognition.

The transition from **NMT_CS_READY_TO_OPERATE** to **NMT_CS_STOPPED** is triggered by reception of NMT state command **NMTStopNode**.

6.9.5 NMT_CS_OPERATIONAL

NMT_CS_OPERATIONAL is the normal operating state of a CN. The MS (Module Status) LED is solidly lit green.

The CN may participate in cyclic frame exchange. The CN responds via PRes when queried via PReq by the MN.

CNs respond to **AsyncInvite** commands via SoA. If not invited by the MN, there is no Ethernet frame transmission allowed at the **NMT_CS_OPERATIONAL** state.

The PDO data received from the MN via PReq and from other CNs and the MN via PRes shall be interpreted if selected by the CN application.

The transition from **NMT_CS_OPERATIONAL** to **NMT_CS_PRE_OPERATIONAL_2** is triggered by the reception of NMT state command **NMTEnterPreOperational2**.

The transition from **NMT_CS_OPERATIONAL** to **NMT_CS_PRE_OPERATIONAL_1** is triggered by an error recognition.

The transition from **NMT_CS_OPERATIONAL** to **NMT_CS_STOPPED** is triggered by reception of NMT state command **NMTStopNode**.

6.9.6 NMT_CS_STOPPED

In the **NMT_CS_STOPPED** state, the node is largely passive. The MS (Module Status) LED blinks green (200 ms ON, 200 ms OFF).

NMT_CS_STOPPED is used for controlled shutdown of a selected CN while the system is still running.

The node does not participate in cyclic frame exchange, but still observes SoA frames.

It is not be queried by the MN via PReq.

The node does not respond via PRes when queried by the MN via PReq.

The node responds to **AsyncInvite** commands via SoA. If not invited by the MN, there is no Ethernet frame transmission allowed at the **NMT_CS_STOPPED** state.

The transition from **NMT_CS_STOPPED** to **NMT_CS_PRE_OPERATIONAL_2** is triggered by the reception of NMT state command **NMTEnterPreOperational2**. The transition from **NMT_CS_STOPPED** to **NMT_CS_PRE_OPERATIONAL_1** is triggered by an error recognition.

6.9.7 NMT_CS_BASIC_ETHERNET

In the **NMT_CS_BASIC_ETHERNET** state the node is allowed to perform Legacy Ethernet communication according to IEEE 802.3. There is no POWERLINK specific network traffic control. The MS (Module Status) LED flickers green (50 ms ON, 50 ms OFF).

The node is allowed to transmit autonomously.

Any Legacy Ethernet protocol can be applied.

Asnd frames can be transmitted by a CN in state **NMT_CS_BASIC_ETHERNET**.

To avoid disturbance of POWERLINK network traffic when the node is in **NMT_CS_BASIC_ETHERNET**, the node recognizes SoC, PReq, PRes and SoA frames. On the reception of such a frame, the CN immediately stalls any autonomous frame transmission and change over to **NMT_CS_PRE_OPERATIONAL_1**.

6.10 XDD file

The functionality of a POWERLINK device is always described in a XDD file (XML Device Description file). The Device Description File provides information about the device basic communication and functional properties. It must be installed in the MN device.

The file name is primarily built up as follows:

0xVendorID_ProductName.xdd

e.g. 0000012E_Lika_EM13_16384

POWERLINK encoders from Lika Electronic are supplied with their own XDD file. Specific XDD files are provided to each encoder series, please refer to the order code: EM58 series; HS58 series; and HM58 series.

They are:

- **0000012E_Lika_EM13_16384_EthernetPOWERLINK.xdd**: it is intended for installation of **EM58 series encoders** ("0000012E_Lika" shows the Vendor ID -expressed in hexadecimal notation- and name; "EM" is the encoder series; "13" is the physical singleturn resolution of the encoder expressed in bits; "16384" is the number of physical revolutions of the multiturn versions);
- **0000012E_Lika_HS18_EthernetPOWERLINK.xdd**: it is intended for installation of **HS58 series encoders**;
- **0000012E_Lika_HM16_16384_EthernetPOWERLINK.xdd**: it is intended for installation of **HM58 series encoders**.

XDD files are paired with the **Lika.bmp** picture file available inside the file folder.

Follow the path **www.lika.biz > ROTARY ENCODERS > ABSOLUTE ENCODERS** to download the XDD files from Lika's corporate web site.

6.11 Communication objects

POWERLINK uses the same device description files as CANopen, the same Object Dictionaries and the same communication mechanisms, such as Network Management (NMT), Process Data Objects (PDO) and Service Data Objects (SDO).

Three main kinds of communication messages are used in a POWERLINK network:

- **Network Management NMT** protocol: NMT protocols are used to issue state machine change commands (i.e. to start and stop the devices), detect remote device boot-ups and error conditions.
- **Process Data Objects PDO** protocol: used to process real time data (transmission of process data in real time).
- **Service Data Objects SDO** protocol: used to set and read values from the Object Dictionary of a remote device in the asynchronous phase.

6.11.1 NMT Network Management

POWERLINK Network Management (NMT) is node-oriented and follows a Master/Slave relationship. The function of the NMT Master is carried out by the MN.

- **NMT State Command Services.** The MN uses NMT State Command Services to control the CN state machine(s), see the "6.9 CN Node NMT States" section on page 66.
- **NMT Managing Command Services.** The MN uses NMT Managing Command Services to access NMT data items of the CN(s) in a fast coordinated way.
- **NMT Response Services.** NMT Response Services indicate the current NMT state of a CN to the MN.
- **NMT Info Services.** NMT Info Services are used to transmit NMT information from the MN to a CN.
- **NMT Guard Services.** NMT Guard Services are used by the MN and CNs to detect failures in a POWERLINK network.

A CN may request NMT command and info services to be issued by the MN. NMT services are defined in the Communication Profile Area of the Object Dictionary, refer to the "6.12 Object Dictionary" section on page 72.

6.11.2 PDO objects

The real-time data transfer is performed by means of Process Data Objects (PDO).

PDO communication in POWERLINK is always performed isochronously by PReq and/or PRes frames. The PRes frames are sent as broadcasts following the producer/consumer scheme. The PReq frames with unicast addresses comply with the Master/Slave relationship.

The transmission type of PDO is continuous. There is no "on event" or "on change" transmission type provided.

6.11.3 PDO Mapping

The PDO Mapping determines the payload of a PDO frame in a POWERLINK network. PDO payload is transported via the frame types PollRequest and PollResponse. Due to the IEEE802.3 standard, any Ethernet frame can have a size between 64 bytes and 1518 bytes. Subtracting the Ethernet- and the POWERLINK-header possible payload sizes of 36 bytes up to 1490 bytes are resulting that can be used for PDO data.

The PDO Mapping is derived from the communication relations between the different nodes in the network. Based on these relations, the payload of the Transmit PDO frames (TPDOs) and the Receive PDO frames (RPDOs) is defined. The PDO mapping can be unchangeable ("static mapping") or configurable ("dynamic mapping").

Mapping tables

The PDO Mapping itself is configured in the form of mapping tables in the Object Dictionary on each node in the network. It consists of PDO Communication Parameters and PDO Mapping Parameters.

The following regions of the Object Dictionary are assigned for the PDO Mapping:

Region	Name	Description
1400h - 14FFh	PDO_RxCommParam_XXh_REC	Communication parameters for RPDOs
1600h - 16FFh	PDO_RxMappParam_XXh_AU64	Object mapping for RPDOs
1800h - 18FFh	PDO_TxCommParam_XXh_REC see on page 78 and following	Communication parameters for TPDOs
1A00h - 1AFFh	PDO_TxMappParam_XXh_AU64 see on page 79	Object mapping for TPDOs

Relationship of Communication Parameters and Mapping Parameters

A single PDO is always described by a pair of one communication parameter object and one mapping parameter object. The pairing is based on the lower byte of the object index.

Examples:

Direction	Communication parameter	Mapping parameter
RPDO	1400h	1600h
	1401h	1601h

TPDO	1800h	1A00h
	1801h	1A01h

6.11.4 SDO objects

To access the entries of the Object Dictionary of a device via Ethernet POWERLINK a set of command services is specified.

An SDO client establishes a connection to an SDO server and issues a specific command (read from or write to an object). This connection is unicast, allows access to all objects of a remote node and is not deterministic (i.e. there is no guarantee on how long the response takes after sending a request). SDO communication takes place in the asynchronous phase and can be embedded in a POWERLINK Asynchronous Send (ASnd) frame or in a UDP/IP packet.

SDO communication attends to the Client / Server model.

6.12 Object Dictionary

The most important part of a device profile is the Object Dictionary. The Object Dictionary is essentially a grouping of objects accessible via the network in an ordered, pre-defined fashion. Each object within the dictionary is addressed using a 16-bit index.

The Object Dictionary can contain a maximum of 65,536 entries.

The user-related objects are grouped in three main areas: the Communication Profile Area, the Manufacturer Specific Profile Area and the Standardised Device Profile Area. The objects are all described in the XDD file.

The **Communication Profile Area** at indexes from 1000h to 1FFFh contains the communication specific parameters for the POWERLINK network. These entries are common to all devices. NMT services, PDO objects and SDO objects are described in this section. The Communication Profile Area objects comply with the "CiA Draft Standard Proposal 301 CANopen Application layer and communication profile". Refer to the "6.12.1 Communication Profile Area objects (DS 301)" section on page 74.

The **Manufacturer Specific Profile Area** at indexes from 2000h to 5FFFh is free to add manufacturer-specific functionality. Refer to the "6.12.2 Manufacturer Specific Profile Area objects" section on page 90.

The **Standardised Device Profile Area** at indexes from 6000h to 9FFFh contains all data objects common to a class of devices that can be read or written via the network. The device profiles may use entries from 6000h to 9FFFh to describe the device parameters and the device functionality. The Standardised Device Profile Area objects comply with the "CiA Draft Standard 406 CANopen Device profile for encoders". Refer to the "6.12.3 Standardised Device Profile Area objects (DS 406)" section on page 91.

In the following pages the objects implemented are listed and described as follows:

Index-subIndex Object name

[data types, attribute]

- Index and subindex are expressed in hexadecimal notation.
- Attribute:
 - ro = read only access
 - rw = read and write access
 - const = constant

6.12.1 Communication Profile Area objects (DS 301)

1000 NMT_DeviceType_U32

[Unsigned32, const]

It contains information about the device type. The object describes the type of device and its functionality.

Default = 0001 0196h = singleturn encoder, HS58 series

0002 0196h = multiturn encoder, EM58 and HM58 series

1001 ERR_ErrorRegister_U8

[Unsigned8, ro]

This object provides error information. The POWERLINK device maps internal errors into this object. The structure of the error register is as follow:

Bit	Meaning
0	Generic error
1	Current
2	Voltage
3	Temperature
4	Communication error
5	Device profile specific
6	Reserved (always 0)
7	Manufacturer specific

If a generic error occurs, the bit 0 will be set to "1".

If a specific error occurs, the corresponding bit (except 6) will be set to "1".

Default = 00h

1006 NMT_CycleLen_U32

[Unsigned32, rw]

This object defines the communication cycle time interval expressed in μ s. This period defines the SYNC interval. The object should be set by the system configuration.

Default = 1000000 (min. = 200, max. = 2147483)

1008 NMT_ManufactDevName_VS

[String64, const]

It contains the manufacturer device name.

Default = EMxxx13/16384-PL-xx = EM58 series

HSxxx18-PL-xx = HS58 series

HMxxx16/16384-PL-xx = HM58 series

1009 NMT_ManufactHwVers_VS

[String64, const]

It shows the manufacturer hardware version description.

Default = device dependent

100A NMT_ManufactSwVers_VS

[String64, const]

It shows the manufacturer software version description.

Default = device dependent

1018-00 NMT_IdentityObject_REC

[Unsigned8, const]

The following objects contain general information about the device. This sub-Index contains the number of entries.

Default = 4

1018-01 NMT_IdentityObject_REC.VendorID_U32

[Unsigned32, const]

It provides the manufacturer-specific vendor ID. The POWERLINK vendor ID is equal to the CANopen vendor ID.

Default = 0000 012Eh

1018-02 NMT_IdentityObject_REC.ProductCode_U32

[Unsigned32, const]

The manufacturer-specific product code identifies a specific device version.

Default = 0000 1000h = HM58 series

0000 1001h = EM58 series

0000 1002h = HS58 series

1018-03 NMT_IdentityObject_REC.RevisionNo_U32

[Unsigned32, const]

The manufacturer-specific revision number consists of a major revision number and a minor revision number. The major revision number identifies a specific device behaviour. The minor revision number identifies different versions with the same device behaviour.

Default = 0000 0001h

31	16	15	0
Major revision number		Minor revision number	
MSB	LSB

1018-04 NMT_IdentityObject_REC.SerialNo_U32

[Unsigned32, const]

It provides the Serial Number of the device.

Default = FFFF FFFFh (=not used)

1020-00 CFM_VerifyConfiguration_REC

[Unsigned8, const]

The following objects hold device local configuration date and time. This sub-Index contains the number of entries.

Default = 2

1020-01 CFM_VerifyConfiguration_REC.ConfDate_U32

[Unsigned32, rw]

It holds the local configuration date. It contains the number of days since January 1, 1984.

Default = 0 (min. = 0, max. = FFFF FFFFh)

1020-02 CFM_VerifyConfiguration_REC.ConfTime_U32

[Unsigned32, rw]

It holds the local configuration time. It contains the number of milliseconds (ms) since midnight.

Default = 0 (min. = 0, max. = FFFF FFFFh)

1030-00 NMT_InterfaceGroup_0h_REC

[Unsigned8, const]

The following objects are used to configure and retrieve parameters of the network interfaces (physical or virtual) via SDO. This sub-Index contains the number of entries.

Default = 9

1030-01 NMT_InterfaceGroup_0h_REC.InterfaceIndex_U16

[Unsigned16, ro]

Interface index of the physical interface. This number is the index number subtracted by 102Fh. The POWERLINK node that adds an interface generates the respective value.

Default = 1 (min. = 1, max. = 10)

1030-02 NMT_InterfaceGroup_0h_REC.InterfaceDescription_VSTR

[String, const]

This string provides information about the name of the manufacturer, the product name and the version of the hardware interface.

Default = LIKA ROTACOD EMxxx13/16384-PL-xx 1.0 = EM58 series
LIKA ROTACOD HSxxx18-PL-xx 1.0 = HS58 series
LIKA ROTACOD HMxxx16/16384-PL-xx 1.0 = HM58 series

1030-03 NMT_InterfaceGroup_0h_REC.InterfaceType_U8

[Unsigned8, const]

The type of interface, distinguished according to the physical/link protocol(s) immediately 'below' the network layer in the protocol stack.

Default = 6 (ethernet-csmacd)

1030-04 NMT_InterfaceGroup_0h_REC.InterfaceMtu_U16

[Unsigned16, const]

It contains the size of the largest datagram that can be sent/received on the interface, specified in bytes.

Default = 1500 (min. = 0, max. = 65535)

1030-05 NMT_InterfaceGroup_0h_REC.InterfacePhysAddress_OSTR

[Octet_String, const]

This object contains the MAC address of the Ethernet device assigned at production.

1030-06 NMT_InterfaceGroup_0h_REC.InterfaceName_VSTR

[String, ro]

The user reference name for the interface.

Default = Interface 1

1030-07 NMT_InterfaceGroup_0h_REC.InterfaceOperState_U8

[Unsigned8, ro]

It shows the current operational state of the interface.

It can be: 0 = Down
 1 = Up

Default = 1 (min. = 0, max. = 1)

1030-08 NMT_InterfaceGroup_0h_REC.InterfaceAdminState_U8

[Unsigned8, rw]

It shows the current administration state of the interface.

It can be: 0 = Down

 1 = Up

Default = 1 (min. = 0, max. = 1)

1030-09 NMT_InterfaceGroup_0h_REC.Valid_BOOL

[Boolean, rw]

It specifies whether the data of this object is valid. If the value is TRUE (1) the data of this object is valid. If the value is FALSE (0) the data of this object is invalid.

Default = TRUE (1) (min. = FALSE (0), max. = TRUE (1))

1300 SDO_SequLayerTimeout_U32

[Unsigned32, rw]

The object provides a timeout value in milliseconds (ms) for the connection abort recognition of the SDO sequence layer.

The connection is detected as broken if the opposite node is shut down or disconnected from the network. The connection is considered broken when no acknowledgement is received within the timeout set in this object.

Default = 15000 (min. = 100, max. = FFFF FFFFh)

1800-00 PDO_TxCommParam_00h_REC

[Unsigned8, const]

These indices describe the communication attributes of the TPDO channels. Mapping version and address information are provided. As a CN has only one TPDO channel, only the first index **PDO_TxCommParam_XXh_REC** is implemented on a CN. This sub-Index contains the number of entries.

Default = 2

1800-01 PDO_TxCommParam_00h_REC.NodeID_U8

[Unsigned8, ro]

It contains the Node ID of the PDO target. It is 0 (not used) for CNs.

Default = 0

1800-02 PDO_TxCommParam_00h_REC.MappingVersion_U8

[Unsigned8, ro]

Compatibility of TPDO channels and corresponding RPDO channels may be ensured by PDO mapping version handling. PDO Mapping can be variable (configurable) or static (unchangeable). Variable mapping may be dynamically modified by the application, even under operation. Static mapping is pre-defined and may not be modified in any way. The version info is transmitted by the Master or producer with every PDO transporting PReq and PRes frame. A

PDO mapping version value of 0 indicates that there is no mapping version available.

Default = 0

1A00-00 PDO_TxMappParam_00h_AU64

[Unsigned8, ro]

These indices describe the mapping of the objects contained in TPDO payload to the Object Dictionary entries. As a CN has only one TPDO channel, only the first index **PDO_TxMappParam_XXh_AU64** is implemented on a CN. This sub-Index contains the number of active entries.

Default = 2

1A00-01 PDO_TxMappParam_00h_AU64.ObjectMapping

[Unsigned64, ro]

It describes the mapping of the **6004 Position value** object to the according TPDO. The entries are interpreted according to the following table (all values in hex):

Length (bit)	Offset (bit)	res	Sub-Index	Index
LLLL	0000	rr	ss	IIII
e.g. 0020	0000	00	00	6004

Map 20h = 32bit starting at offset 0 to **6004 Position value** object

Default = 0020 0000 0000 6004h

1A00-02 PDO_TxMappParam_00h_AU64.ObjectMapping

[Unsigned64, ro]

It describes the mapping of the **6030 Speed value** object to the according TPDO. The entries are interpreted according to the following table (all values in hex):

Length (bit)	Offset (bit)	res	Sub-Index	Index
LLLL	0000	rr	ss	IIII
e.g. 0020	0020	00	00	6030

Map 20h = 32bit starting at offset 20h = 32bit to **6030 Speed value** object

Default = 0020 0020 0000 6030h

1COB-00 DLL_CNLossSoC_REC

[Unsigned8, const]

The following objects are used to monitor "Loss of SoC" (Start of Cycle frame) error symptoms detected by a CN. The record consists of a cumulative counter and a threshold counter data object and its threshold data object. This sub-Index contains the number of entries.

Default = 3

1COB-01 DLL_CNLossSoC_REC.CumulativeCnt_U32

[Unsigned32, rw]

The cumulative counter is incremented by 1 every time a "Loss of SoC" error symptom occurs. Its value monitors all "Loss of SoC" error symptoms that are detected by the CN.

Default = 0

**NOTE**

If the unit is reset, this attribute is set to its default value.

1COB-02 DLL_CNLossSoC_REC.ThresholdCnt_U32

[Unsigned32, ro]

The threshold counter shall be incremented by 8 every time a "Loss of SoC" error symptom occurs and decremented by 1 at every cycle without reoccurrence of the error. Its value monitors the quality of network in relation to the "Loss of SoC" error occurrence.

Default = 0

1COB-03 DLL_CNLossSoC_REC.Threshold_U32

[Unsigned32, rw]

Every time **1COB-02 DLL_CNLossSoC_REC.ThresholdCnt_U32** reaches the threshold, a defined action proceeds and **1COB-02 DLL_CNLossSoC_REC.ThresholdCnt_U32** is reset to 0.

Threshold Counting can be deactivated by setting **1COB-03 DLL_CNLossSoC_REC.Threshold_U32** to 0. If Threshold Counting is deactivated, no error reaction will occur.

Default = 15 (min. = 0, max. = FFFF FFFFh)

1COFB-00 DLL_CNCRCErrror_REC

[Unsigned8, const]

The following objects are used to monitor CRC (Cyclic Redundancy Check) errors. The record consists of a cumulative counter and a threshold counter data object and its threshold data object. This sub-Index contains the number of entries.

Default = 3

1C0FB-01 DLL_CNCRCErrror_REC.CumulativeCnt_U32

[Unsigned32, rw]

The cumulative counter is incremented by 1 every time a CRC error occurs. Its value monitors all CRC errors that are detected by the CN.

Default = 0



NOTE

If the unit is reset, this attribute is set to its default value.

1C0FB-02 DLL_CNCRCErrror_REC.ThresholdCnt_U32

[Unsigned32, ro]

The threshold counter is incremented by 8 every time a CRC error occurs on the CN and decremented by 1 at every cycle without reoccurrence of the error. Its value monitors the quality of network in relation to the CRC error occurrence.

Default = 0

1C0FB-03 DLL_CNCRCErrror_REC.Threshold_U32

[Unsigned32, rw]

Every time **1C0FB-02 DLL_CNCRCErrror_REC.ThresholdCnt_U32** reaches the threshold, a defined action proceeds and **1C0FB-02 DLL_CNCRCErrror_REC.ThresholdCnt_U32** is reset to 0.

Threshold Counting can be deactivated by setting **1C0FB-03 DLL_CNCRCErrror_REC.Threshold_U32** to 0. If Threshold Counting is deactivated, no error reaction will occur.

Default = 15 (min. = 0, max. = FFFF FFFFh)

1C14 DLL_CNLossOfSocTolerance_U32

[Unsigned32, rw]

This object provides a tolerance interval expressed in ns to be applied by CNs for "Loss of SoC" error recognition.

Default = 100000 (min. = 0, max. = 2147483000)

1F81-00 NMT_NodeAssignment_AU32

[Unsigned8, rw]

This object assigns nodes to the NMT Master (MN). Each sub-Index in the array corresponds to the node with the Node ID equal to the sub-Index. This sub-Index contains the number of entries.

Default = 254 (min. = 1, max. = 254)

1F81-01 NMT_NodeAssignment_AU32.NodeAssignment

[Unsigned32, rw]

Bit field, its meaning is according to the following table:

Byte	Bit	Value	Description
0	0	0	Node with this ID does not exist, bits 1 to 30 are not used.
		1	Node with this ID exists.
	1	0	Node with this ID is not a CN, Bits 2 .. 7, 9, 13 .. 30 are not used.
		1	Node with this ID is a CN. After configuration (with Configuration Manager) the Node will be set to state NMT_CS_OPERATIONAL .
	2	0	On detection of a booting CN MN informs the application but does NOT automatically configure and start the node.
		1	On detection of a booting CN MN informs the application and continues the process "START_CN".
	3	0	Optional CN.
		1	Mandatory CN.
	4	0	The CN node can be reset by the NMTSwReset , NMTResetNode , NMTResetCommunication or NMTResetConfiguration commands independent of its state. Hence no checking of its state needs to be performed prior to NMT Reset Communication.
		1	MN must not send any of the reset commands listed above to this node if it notices that the CN is in NMT_CS_OPERATIONAL state.
	5	0	Application software version verification for this node is not required.
		1	Application software version verification for this node is required.
	6	0	Automatic application software update (download) is not allowed.
		1	Automatic application software update (download) is allowed.
7	-	Reserved	
1	8	0	Isochronously accessed node.
		1	AsyncOnly node.
	9	0	Continuously accessed CN.
		1	Multiplexed CN.
	10	0	Device is not a Router Type 1.
		1	Device is a Router Type 1.
	11	0	Device is not a Router Type 2.
		1	Device is a Router Type 2.
	12	0	MN does not transmit PRes.
		1	MN transmits PRes.
	13	-	Reserved.
14	-	Reserved.	
15	-	Reserved.	

2	16 ... 23	-	Reserved.
	24 ... 30	-	Reserved.
3	31	0	Bit 0 ... 30 not valid.
		1	Bit 0 ... 30 valid.

Default = 0

1F82 NMT_FeatureFlags_U32

[Unsigned32, const]

Feature Flags indicate communication profile specific properties of the device given by its design. The object is set up by the device firmware during system initialisation. Its meaning is according to the following table (grey background shows the set values, see the default binary value):

Byte	Bit	Name	TRUE (1)	FALSE (0)
0	0	Isochronous	Device can be isochronously accessed via PReq, it can be operated as isochronous CN. D_NMT_Isochronous_BOOL	Device does not support isochronous access via PReq, it may be exclusively used as async-only CN.
	1	SDO by UDP/IP	Device supports SDO communication via UDP/IP frames. D_SDO_SupportUdplp_BOOL	Device does not support
	2	SDO by Asnd	Device supports SDO communication via POWERLINK Asnd frames. D_SDO_SupportASnd_BOOL	Device does not support
	3	SDO by PDO	Device supports SDO communication via container embedded in PDO communication. D_SDO_SupportPDO_BOOL	Device does not support
	4	NMT Info Services	Device supports NMT Info Services.	Device does not support
	5	Extended NMT State Commands	Device supports extended NMT State Commands. D_NMT_ExtNmtCmds_BOOL	Device does not support
	6	Dynamic PDO Mapping	Device supports dynamic PDO mapping. D_PDO_DynamicMapping_BOOL	Device does not support
	7	NMT Services by UDP/IP	Device supports NMT Services by UDP/IP. D_NMT_ServiceUdplp_BOOL	Device does not support

1	8	Configuration Manager	Device supports configuration manager functions. D_CFM_ConfigManager_BOOL	Device does not support
	9	Multiplexed Access	Device supports multiplexed isochronous access. D_DLL_CNFeatureMultiplex_BOOL	Device does not support
	10	Node ID setup by SW	Device supports Node ID setup via SW. D_NMT_NodeIDBySW_BOOL	Device does not support
	11	MN Basic Ethernet Mode	MN Device supports Basic Ethernet Mode. D_NMT_MNBasicEthernet_BOOL	MN Device does not support
	12	Routing Type 1 Support	Device supports Routing Type 1 functions. D_RT1_RT1Support_BOOL	Device does not support
	13	Routing Type 1 Support	Device supports Routing Type 2 functions. D_RT2_RT2Support_BOOL	Device does not support
	14	SDO Read/Write All by Index	Device supports SDO commands Read and Write All by Index. D_SDO_CmdReadAllByIndex_BOOL D_SDO_CmdWriteAllByIndex_BOOL	Device does not support
	15	SDO Read/Write Multiple Parameter by Index	Device supports SDO commands Read and Write Multiple Parameter by Index. D_SDO_CmdReadMultParam_BOOL D_SDO_CmdWriteMultParam_BOOL	Device does not support
2	... 23	-	Reserved.	
3	24 ... 31	-	Reserved.	

Default = 0004 8205h = 0100 1000 0010 0000 0101₂

1F83 NMT_EPLVersion_U8

[Unsigned8, const]

The index holds the POWERLINK communication profile version that is implemented by the device, according to the following table:

High Nibble	Low Nibble
POWERLINK Main Version	POWERLINK Sub Version

Default = 20h

1F8C NMT_CurrNMTState_U8

[Unsigned8, ro]

The index holds the node's current NMT state, according to the following table. The CN Node specific states are described in the "6.9 CN Node NMT States" section on page 66.

	NMT states	Value	
MN and CN States	NMT_GS_OFF	0000 0000	
	NMT_GS_POWERED	xxxx 1xxx	Super state
	NMT_GS_INITIALISATION	xxxx 1001	Super state
	NMT_GS_INITIALISING	0001 1001	
	NMT_GS_RESET_APPLICATION	0010 1001	
	NMT_GS_RESET_COMMUNICATION	0011 1001	
	NMT_GS_RESET_CONFIGURATION	0111 1001	
	NMT_GS_COMMUNICATING	xxxx 11xx	Super state
CN States	NMT_CS_NOT_ACTIVE	0001 1100	
	NMT_CS_EPL_MODE	xxxx 1101	Super state
	NMT_CS_PRE_OPERATIONAL_1	0001 1101	
	NMT_CS_PRE_OPERATIONAL_2	0101 1101	
	NMT_CS_READY_TO_OPERATE	0110 1101	
	NMT_CS_OPERATIONAL	1111 1101	
	NMT_CS_STOPPED	0100 1101	
	NMT_CS_BASIC_ETHERNET	0001 1110	



EXAMPLE

1F8C NMT_CurrNMTState_U8 = 253₁₀ = 1111 1101₂ = NMT_CS_OPERATIONAL state

1F8D-00 NMT_PResPayloadLimitList_AU16

[Unsigned8, rw]

The following object(s) hold(s) a list of the expected PRes payload data slot size in bytes for each configured node that is isochronously accessed, e.g. via PReq / PRes frame exchange. The payload data slot size is a measure for the configured size of the PRes frame. The data slot can be filled by PDO data up to this limit. This sub-Index contains the number of entries.

Default = 1 (min. = 1, max. = 254)

1F8D-01 NMT_PResPayloadLimitList_AU16.PResPayloadLimit

[Unsigned16, rw]

Each sub-Index in the array corresponds to the node with the Node ID equal to the sub-Index. The sub-Index value is valid only if there is an isochronous node assigned to the Node ID by index **NMT_NodeAssignment_AU32**[sub-Index] bits 0 and 8 (see page 81).

Sub-Index **C_ADR_MN_DEF_NODE_ID** indicates the payload size of the PRes frame issued by the MN.

Values should be equal on all nodes of the segment.

Default = 36 (C_DLL_ISOCHR_MAX_PAYL)

1F93-00 NMT_EPLNodeID_REC

[Unsigned8, const]

The following objects store the devices' POWERLINK Node ID. This sub-Index contains the number of entries.

Default = 2

1F93-01 NMT_EPLNodeID_REC.NodeID_U8

[Unsigned8, ro]

This sub-Index holds the device's current POWERLINK Node ID. It is provided by hardware settings (DIP switch), refer to the "4.8 POWERLINK Node ID: DIP A" section on page 32. By default the POWERLINK Node ID is set to "1".

Default = 1 (min. = 1, max. = 254)

1F93-02 NMT_EPLNodeID_REC.NodeIDByHW_BOOL

[Boolean, ro]

This sub-Index displays the Node ID setup mode of the device. It is set up during system initialisation.

- TRUE (1) = the device's POWERLINK Node ID is set up exclusively by HW.
- FALSE (0) = the device's POWERLINK Node ID is set up exclusively by SW.

Default = TRUE (1) (min. = 0 (FALSE), max. = 1 (TRUE))

1F98-00 NMT_CycleTiming_REC

[Unsigned8, const]

The following objects provides node specific timing parameters, that influence the POWERLINK cycle timing. This sub-Index contains the number of entries.

Default = 8

1F98-01 NMT_CycleTiming_REC.IsochrTxMaxPayload_U16

[Unsigned16, const]

It provides the device specific upper limit for payload data size expressed in bytes of isochronous messages to be transmitted by the device.

On all nodes, the sub-Index limits the size of the PRes frame issued by the node (sub-Index **1F98-05 NMT_CycleTiming_REC.PresActPayloadLimit_U16**, refer

below). Additionally on the MN, the size of transmitted PReq messages (object [NMT_MNPreqPayloadLimitList_AU16](#)) is affected.

The limit is set up by the device firmware during system initialisation.

Default = 1490 (min. = 36, max. = 1490)

[1F98-02 NMT_CycleTiming_REC.IsochrRxMaxPayload_U16](#)

[Unsigned16, const]

It provides the device specific upper limit for payload data size expressed in bytes of isochronous messages to be received by the device.

On all nodes, the sub-Index limits the size of the PRes frames received by the node (object [1F8D-00 NMT_PResPayloadLimitList_AU16](#)). Additionally on the CN, the size of the received PReq message (sub-Index [1F98-04 NMT_CycleTiming_REC.PreqActPayloadLimit_U16](#), refer below) is affected.

The limit is set up by the device firmware during system initialisation.

Default = 1490 (min. = 36, max. = 1490)

[1F98-03 NMT_CycleTiming_REC.PresMaxLatency_U32](#)

[Unsigned32, const]

It provides the maximum time expressed in nanoseconds (ns), that is required by the CN to respond to PReq.

The value is set up by the device firmware during system initialisation.

Default = 1000 (min. = 0, max. = 4294967295)

[1F98-04 NMT_CycleTiming_REC.PreqActPayloadLimit_U16](#)

[Unsigned16, rw]

It provides the configured PReq payload data slot size expressed in bytes expected by the CN. The payload data slot size plus headers gives the size of the PReq frame. The data slot can be filled by PDO data up to this limit.

Default = 36 (min. = 36, max. = 1490)



NOTE

This results in a fixed frame size regardless of the size of PDO data used.

[1F98-05 NMT_CycleTiming_REC.PresActPayloadLimit_U16](#)

[Unsigned16, rw]

It provides the configured PRes payload data slot size expressed in bytes sent by the CN. The payload data slot size plus headers gives the size of the PRes frame. The data slot can be filled by PDO data up to this limit.

Default = 36 (min. = 36, max. = 1490)



NOTE

This results in a fixed frame size regardless of the size of PDO data used.

1F98-06 NMT_CycleTiming_REC.AsndMaxLatency_U32

[Unsigned32, const]

It provides the maximum time in nanoseconds (ns), that is required by the CN to respond to SoA.

The value is set up by the device firmware during system initialisation.

Default = 1000 (min. = 0, max. = 4294967295)

1F98-07 NMT_CycleTiming_REC.MultiplCycleCnt_U8

[Unsigned8, rw]

This sub-Index describes the length of the multiplexed cycle in multiples of the POWERLINK cycle.

The value is upper limited by the MN's device description entry **D_NMT_MNMultiplCycMax_U8**. It is equal in all nodes of the segment.

If **1F98-07 NMT_CycleTiming_REC.MultiplCycleCnt_U8** is zero (0), there is no support of multiplexed cycle on the network.

Default = 0 (min. = 0, max. = 255)

1F98-08 NMT_CycleTiming_REC.AsyncMTU_U16

[Unsigned16, rw]

This sub-Index describes the maximum asynchronous frame size expressed in bytes. The value applies to Asnd frames as well as to UDP/IP and other legacy Ethernet type frames. For this reason the value describes the length of the complete Ethernet frame minus 14 bytes Ethernet header and 4 bytes checksum.

It is upper limited by the **1030-04 NMT_InterfaceGroup_0h_REC.InterfaceMtu_U16** values of all devices in the segment. This limit is 18 bytes less than the minimum **InterfaceMTU_U16** value provided by any node in the segment. **AsyncMTU_U16** can grow up to **C_DLL_MAX_ASYNC_MTU**.

AsyncMTU_U16 is equal in all nodes of the segment.

This sub-Index is valid in all NMT states.

Default = 300 (min. = 300, max. = 1500)

1F99 NMT_CNBasicEthernetTimeout_U32

[Unsigned32, rw]

It provides the time expressed in microseconds (μ s) to be applied before changing from **NMT_CS_NOT_ACTIVE** state to **NMT_CS_BASIC_ETHERNET** state.

Please note that MN and CN startup timing has to be well balanced. System power up sequence has to be considered.

Value "0" means that the state never changes to **NMT_CS_BASIC_ETHERNET**.

If other than "0", the value shall be greater than **1006 NMT_CycleLen_U32**.

To avoid erroneous change over to **NMT_CS_BASIC_ETHERNET** state at system startup, **1F99 NMT_CNBasicEthernetTimeout_U32** must be greater than the **NMT_BootTime_REC.MNWaitNotAct_U32** of the MN.

Default = 5000000 (min. = 0, max. = 4294967295)

1F9B-00 NMT_MultiplCycleAssign_AU8

[Unsigned8, rw]

These objects assign the nodes to the particular POWERLINK cycles of the multiplexed cycle period defined by **1F98-07 NMT_CycleTiming_REC.MultiplCycleCnt_U8**. The value has to be equal in all nodes of the segment. This sub-Index contains the number of entries.

Default = 254 (min. = 1, max. = 254)

1F9B-01 NMT_MultiplCycleAssign_AU8.CycleNo

[Unsigned8, rw]

Each sub-Index in the array corresponds to the node with the Node ID equal to the sub-Index. The sub-Index value is valid only if there is a multiplexed node assigned to the Node ID by index **NMT_NodeAssignment_AU32**[sub-Index] bits 0, 1 and 9.

This object defines the POWERLINK cycle index in the multiplexed cycle, when the respective node are accessed. If the value is "0", the node is accessed continuously.

Default = 0 (min. = 0, max. = 255)

1F9E NMT_ResetCmd_U8

[Unsigned8, rw]

This is used to initiate the reset of a node.

Setting **1F9E NMT_ResetCmd_U8** to the NMT Command ID **NMTResetNode**, **NMTResetConfiguration**, **NMTResetCommunication** or **NMTSwReset** will trigger the node internal generation of a respective NMT command to itself.

1F9E NMT_ResetCmd_U8 is automatically reset to **NMTInvalidService** by the node when the reset has been completed.

On read access, **1F9E NMT_ResetCmd_U8** always shows **NMTInvalidService**.

If applied in **NMT_CS_EPL_MODE** state, resets by **1F9E NMT_ResetCmd_U8** may violate the NMT rules and stimulate DLL and NMT Guarding errors.

Default = 255 (min. = 0, max. = 255)

6.12.2 Manufacturer Specific Profile Area objects

2002 Speed format

[Unsigned16, rw]

This attribute defines the engineering unit for the velocity value (see the [6030 Speed value](#) object on page 97).

0 = steps/s: number of steps per second;

1 = rpm: revolutions per minute.

Default = 0 (min. = 0, max. = 255)

6.12.3 Standardised Device Profile Area objects (DS 406)

6000 Operating parameters

[Unsigned16, rw]

Bit	Function	bit = 0	bit = 1
0	Code sequence	CW (clockwise)	CCW (counter clockwise)
1	not used		
2	Scaling function	disabled	enabled
3 ... 13	not used		
14	Restore default parameters		
15	Store parameters		

Default = 0000h

Code sequence

It defines whether the position value output by the encoder increases (count up information) when the encoder shaft rotates clockwise (0 = CW) or counter-clockwise (1 = CCW). If bit 0 **Code sequence** = 0, the absolute position value **increases** when the encoder shaft rotates **clockwise**; on the contrary, if bit 0 **Code sequence** = 1, the absolute position value **increases** when the encoder shaft rotates **counter-clockwise**. CW and CCW rotations are viewed from the shaft end.



WARNING

Changing this value causes also the position calculated by the controller to be necessarily affected. Therefore it is mandatory to execute a new preset and save the parameters after setting this parameter.



NOTE

To know whether the **Code sequence** is currently set to CW or CCW, you can read the bit 0 **Code sequence** of the **6500 Operating status** object, see on page 98.

Scaling function

If this function is disabled (bit 2 **Scaling function** = 0), the device uses the physical resolution to arrange the absolute position value (see the **6501 Singleturn resolution** and **6502 Number of revolutions** objects); if this function is enabled (bit 2 **Scaling function** = 1), the device uses the custom resolution set next to the **6001 Measuring units per revolution** and **6002 Total measuring range in measuring units** objects in compliance with the following relation:

Transmitted position =

6001 Measuring units per revolution

6501 Singleturn resolution

* real position ≤ **6002 Total measuring range in measuring units**



NOTE

To know whether the **Scaling function** is currently enabled, you can read the bit 2 **Scaling function** of the **6500 Operating status** object, see on page 98.



WARNING

Every time you enable the scaling function and/or change the scaling values (see the **6001 Measuring units per revolution** and **6002 Total measuring range in measuring units** objects) then you are required to set a new preset value (see the **6003 Preset value** object) and finally save the new parameters (see the **Store parameters** function).

Restore default parameters

This function allows the operator to restore all parameters to default values (default values are set at the factory by Lika Electronic engineers to allow the operator to run the device for standard operation in a safe mode).

Default parameters are restored at each rising edge of the bit; in other words, the default parameters uploading operation is performed each time this bit is switched from logic level low ("0") to logic level high ("1"). The complete list of machine data and relevant default parameters preset by Lika Electronic engineers is available on page 104.



WARNING

The execution of this command causes all parameters which have been set previously to be overwritten!

Store parameters

This function allows to save all parameters on non-volatile memory.

Data is saved on non-volatile memory at each rising edge of the bit; in other words, data save is performed each time this bit is switched from logic level low ("0") to logic level high ("1").



NOTE

Always execute the **Store parameters** function to save the new parameter values.

If the power is turned off, the parameters that are not saved will be lost.

6001 Measuring units per revolution

[Unsigned32, rw]



WARNING

This object is active only if the bit 2 **Scaling function** in the **6000 Operating parameters** object is set to "1"; otherwise it is ignored and the system uses the physical values (**6501 Singleturn resolution** and **6502 Number of revolutions**) to calculate the position information.

This object sets the custom number of measuring steps per revolution that are output for the absolute singleturn position value.

If you enter an out-of-range value, the number of steps per revolution is forced to the physical singleturn resolution.

To avoid counting errors, please check that:

$$\frac{\text{6501 Singleturn resolution}}{\text{6001 Measuring units per revolution}} = \text{integer value.}$$

Allowed values must be less than or equal to the physical singleturn resolution (**6501 Singleturn resolution**).

Default = 8192 (min. = 1, max. = 8192)	for EM58 series
262144 (min. = 1, max. = 262144)	for HS58 series
65536 (min. = 1, max. = 65536)	for HM58 series



WARNING

When you set a new value next to the **6001 Measuring units per revolution** object, please always check also the **6002 Total measuring range in measuring units** object value and be sure that the resulting number of revolutions complies with the physical number of revolutions of the device (see the **6502 Number of revolutions** object).

Let's suppose that the HM58-16-14-PL2-... encoder is programmed as follows:

6001 Measuring units per revolution: 8,192

6002 Total measuring range in measuring units = 33,554,432 = 8,192 (cpr) * 4,096 (rev.)

Let's set a new singleturn resolution, for instance: **6001 Measuring units per revolution** = 360.

If we do not change the **6002 Total measuring range in measuring units** value at the same time, we will get the following result:

$$\text{Number of revolutions} = \frac{33,554,432 \text{ (6002 Total measuring range in measuring units)}}{360 \text{ (6001 Measuring units per revolution)}} = 93,206.755...$$

As you can see, the encoder is required to carry out more than 93,000 revolutions, this cannot be as the hardware number of revolutions is, as stated, 16,384 (see the **6502 Number of revolutions** object). When this happens, the encoder falls into an error signalling the faulty condition through the diagnostic LEDs (see on page 34).



WARNING

Every time you change the value in this object then you are required to set a new preset value (see the **6003 Preset value** object) and finally save the new parameters (see the **Store parameters** function).

6002 Total measuring range in measuring units

[Unsigned32, rw]



WARNING

This object is active only if the bit 2 **Scaling function** in the **6000 Operating parameters** object is set to "1"; otherwise it is ignored and the system uses the physical values (**6501 Singleturn resolution** and **6502 Number of revolutions**) to calculate the position information.

This object sets a custom number of distinguishable steps over the total measuring range. The total resolution of the encoder results from the product of **6001 Measuring units per revolution** by the required **Number of revolutions**. Allowed values are less than or equal to **6001 Measuring units per revolution * 6502 Number of revolutions**.

Default = 134217728 (min. = 1, max. = 134217728)	for EM58 series
262144 (min. = 1, max. = 262144)	for HS58 series
1073741824 (min. = 1, max. = 1073741824)	for HM58 series



WARNING

When you set a new value next to the **6002 Total measuring range in measuring units** object, please always check also the **6001 Measuring units per revolution** object value and be sure that the resulting number of revolutions complies with the **Hardware number of revolutions** of the device.

Let's suppose that the HM58-16-14-PL2-... encoder is programmed as follows:

6001 Measuring units per revolution: 8,192

6002 Total measuring range in measuring units = 33,554,432 = 8,192 (cpr) * 4,096 (rev.)

Let's set a new total resolution, for instance: **6002 Total measuring range in measuring units** = 360.

As the **6002 Total measuring range in measuring units** must be greater than or equal to the **6001 Measuring units per revolution**, the above setting is not allowed. When this happens, the encoder falls into an error signalling the faulty condition through the diagnostic LEDs (see on page 34).

**WARNING**

Every time you change the value in this object then you are required to set a new preset value (see the **6003 Preset value** object) and finally save the new parameters (see the **Store parameters** function).

**EXAMPLE**

We install the HM58-16-14-PL2-... multiturn encoder.

Its physical resolution is as follows (see the order code):

- Hardware counts per revolution: **6501 Singleturn resolution** = 65,536 (2^{16})
- Hardware number of revolutions: **6502 Number of revolutions** = 16,384 (2^{14})
- Total hardware resolution: **6501 Singleturn resolution** * **6502 Number of revolutions** = 1,073,741,824 (2^{30})

In the specific installation 2,048 counts/rev. * 1,024 turns are required:

- Enable the scaling function: **6000 Operating parameters**, bit 2 **Scaling function** = "1"
- Counts per revolution: **6001 Measuring units per revolution** = 2,048 (0000 0800h)
- Total resolution: **6002 Total measuring range in measuring units** = 2048 * 1024 = 2,097,152 (0020 0000h)

**NOTE**

We suggest setting values which are power of 2 (2^n : 2, 4, ..., 2048, 4096, 8192,...) to be set in the **6001 Measuring units per revolution** and **6002 Total measuring range in measuring units** objects to avoid counting errors.

**WARNING**

If **6001 Measuring units per revolution** and/or **6002 Total measuring range in measuring units** values change, the **6003 Preset value** must be updated in accordance with the new resolution. A new preset operation is also required.

6003 Preset value

[Unsigned32, rw]

This object allows to set the encoder position to a Preset value. The Preset function is meant to assign a desired value to a physical position of the encoder shaft. The chosen position will get the value set next to this object and all the previous and the following positions will get a value according to it. This function is useful, for example, when the zero position of the encoder and the zero position of the axis need to match. The preset value will be set for the position of the encoder in the moment when the preset value is transmitted. We suggest setting the preset value when the encoder is in stop.

To execute the preset you must transmit the value to the **6003 Preset value** object in the asynchronous phase via SDO when the encoder is in operational state (**NMT_CS_OPERATIONAL**) and then execute the **Store parameters** function (see the **Store parameters** function on page 92).

Default = 0 (min. = 0, max. = 134217727 *)	for EM58 series
0 (min. = 0, max. = 262143 *)	for HS58 series
0 (min. = 0, max. = 1073741823 *)	for HM58 series

* See the NOTE below.



EXAMPLE

Let's take a look at the following example to better understand the preset function and the meaning and use of the related objects: **6003 Preset value** and **6509 Offset value**.

The encoder position which is transmitted results from the following calculation:

Transmitted value = read position (it does not matter whether the position is physical or scaled) + **6003 Preset value - 6509 Offset value**.

If you never set the **6003 Preset value** and you never performed the preset setting, then the transmitted value and the read position are necessarily the same as **6003 Preset value = 0** and **6509 Offset value = 0**.

When you set the **6003 Preset value** and then execute the preset setting, the system saves the current encoder position in the **6509 Offset value** object. It follows that the transmitted value and the **6003 Preset value** are the same as **read position - 6509 Offset value = 0**; in other words, the value set next to the **6003 Preset value** object is paired with the current position of the encoder as you wish.

For example, let's assume that the value "50" is set next to the **6003 Preset value** object and you execute the preset setting when the encoder position is "1000". In other words, you want to receive the value "50" when the encoder reaches the position "1000".

We will obtain the following information sequence:

Transmitted value = read position (= "1000") + **6003 Preset value** (= "50") - **6509 Offset value** (= "1000") = 50.

The following transmitted value will be:

Transmitted value = **read position** (= "1001") + **6003 Preset value** (= "50") - **6509 Offset value** (= "1000") = 51.

And so on.



NOTE

- If the scaling function is disabled (the bit 2 **Scaling function** in the **6000 Operating parameters** object = 0), then the **6003 Preset value** must be less than or equal to the "Total hardware resolution" - 1, i.e. (**6501 Singleturn resolution** * **6502 Number of revolutions**) - 1.
- If the scaling function is enabled (the bit 2 **Scaling function** in the **6000 Operating parameters** object = 1), then the **6003 Preset value** must be lower than or equal to the **6002 Total measuring range in measuring units** - 1.



WARNING

Check the value in the **6003 Preset value** object and perform the preset operation every time you set a new **Code sequence** or change the scaled values (**6001 Measuring units per revolution** and/or **6002 Total measuring range in measuring units**).

6004 Position value

[Unsigned32, ro]

This object contains the position value, the transmitted value is calculated considering the scaling values (see on page 91).

6030 Speed value

[Integer32, ro]

This attribute shows the current output speed value detected by the position encoder and calculated every 100 ms.

The value can be expressed in either steps per second or revolutions per minute according to the setting next the **2002 Speed format** object on page 90.

6500 Operating status

[Unsigned16, ro]

Bit	Function	bit = 0	bit = 1
0	Code sequence	CW Clockwise	CCW counter clockwise
1	not used		
2	Scaling function	Disabled	Enabled
3 ... 15	not used		

Code sequence

It shows whether the code sequence is set to clockwise (CW) or counter-clockwise (CCW). If the bit is "0" the output encoder position value has been set to increase (count up information) when the encoder rotates clockwise; if the bit is "1" the output encoder position value has been set to increase when the encoder rotates counter-clockwise. To set the code sequence to either CW or CCW you must set the bit 0 **Code sequence** of the **6000 Operating parameters** object to 0 / 1. For any further information on setting and using the counting direction refer to the **6000 Operating parameters** object on page 91.

Scaling function

It shows whether the scaling function is disabled or enabled. If the value is "0" the scaling function is disabled (i.e. the system uses the resolution physical values -**6501 Singleturn resolution** and **6502 Number of revolutions**- to calculate the position information); if the value is "1" the scaling function is enabled (i.e. the system uses the custom values -**6001 Measuring units per revolution** and **6002 Total measuring range in measuring units**- to calculate the position information). To disable / enable the scaling function you must set the bit 2 **Scaling function** of the **6000 Operating parameters** object to 0 / 1. For any further information on setting and using the scaling function refer to the **6000 Operating parameters** object on page 91.

6501 Singleturn resolution

[Unsigned32, ro]



WARNING

This object is active only if the bit 2 **Scaling function** in the **6000 Operating parameters** object is set to "0"; otherwise it is ignored and the system uses the custom values (**6001 Measuring units per revolution** and **6002 Total measuring range in measuring units**) to calculate the position information.

This object is intended to show the number of physical distinguishable steps each turn provided by the hardware (physical singleturn resolution).
 If you want to set a custom resolution see the **6001 Measuring units per revolution** object.

Default = 8192 for EM58 series
 262144 for HS58 series
 65536 for HM58 series

6502 Number of revolutions

[Unsigned16, ro]



WARNING

This object is active only if the bit 2 **Scaling function** in the **6000 Operating parameters** object is set to "≠0"; otherwise it is ignored and the system uses the custom values (**6001 Measuring units per revolution** and **6002 Total measuring range in measuring units**) to calculate the position information.

This object is intended to show the number of physical distinguishable turns provided by the hardware (number of physical revolutions).

The **Total hardware resolution** results from **6501 Singleturn resolution** * **6502 Number of revolutions**.

If you want to set a custom number of turns see the **6001 Measuring units per revolution** and **6002 Total measuring range in measuring units** objects.

Default = 16384 for EM58 series
 1 for HS58 series
 16384 for HM58 series

6503 Alarms

[Unsigned16, ro]

Bit	Function	bit = 0	bit = 1
0 ... 11	not used		
12	Machine data not valid	Alarm not active	Alarm active
13	Setting data not valid	Alarm not active	Alarm active
14	Flash memory error	Alarm not active	Alarm active
15	not used		

This object provides further alarm messages. An alarm will be set if a malfunction in the encoder could lead to incorrect position value. If an alarm occurs, the according bit is set to logical high (1) until the alarm is cleared and the encoder is able to provide an accurate position value.

The available alarm messages are listed here afterwards.

Machine data not valid

One or more parameters are not valid, set proper values to restore normal work condition.

Setting data not valid

This alarm message is currently disabled in this firmware version.

Flash memory error

Flash memory internal error, it cannot be restored (bad checksum error, etc.).

6504 Supported alarms

[Unsigned16, ro]

Bit	Function	bit = 0	bit = 1
0 ... 11	not used		
12	Machine data not valid	Alarm not supported	Alarm supported
13	Setting data not valid	Alarm not supported	Alarm supported
14	Flash memory error	Alarm not supported	Alarm supported
15	not used		

This object contains the information on the alarms supported by the encoder. The available alarm messages are listed in the **6503 Alarms** object.

Default = 7000h (= 0111 0000 0000 0000 = alarms at bits 12, 13, and 14 of the previous **6503 Alarms** object are supported).

6506 Supported warnings

[Unsigned16, ro]

This object contains the information on the warnings supported by the encoder. No warnings are supported in this encoder.

Default = 0

6507 Profile and software version

[Unsigned32, ro]

This object contains the implemented encoder device profile version and the manufacturer specific software version.

Profile version for encoders = 3.1

Software version = device dependent

Upper profile version	Lower profile version	Upper software version	Lower software version
MSB		LSB	

Default = 0301 xxxxh

6508 Operating time

[Unsigned32, ro]

This object contains the information on the operating time. The operating time monitor stores the operating time for the encoder expressed in operating hours. The operating time is stored in the encoder non-volatile memory as long as the encoder is power supplied.

This object is currently not used in this encoder.

Default = FFFF FFFFh

6509 Offset value

[Integer32, ro]

As soon as you activate the preset, the current position value of the encoder is saved in this object. The offset value is then used in the preset function in order to calculate the encoder position value to be transmitted. To zero set the value in this object you must upload the factory default values (see the **Restore default parameters** function on page 92).

For any further information on the preset function and the meaning and use of the related objects refer to page 96.

Default = 0000 0000h

650B Serial number

[Unsigned32, ro]

This object contains the encoder serial number.

This object is currently not used in this encoder.

Default = FFFF FFFFh



NOTE

Always execute the **Store parameters** function to save the new parameter values.

If the power is turned off, the parameters that are not saved will be lost.

6.13 SDO abort codes

Here follows the list and meaning of the SDO abort codes indicated by POWERLINK but not necessarily supported by the manufacturer. They comply with the CANopen SDO abort codes (for CANopen SDO abort codes refer to the "SDO abort transfer protocol" section in the "CiA Draft Standard 301" document). For complete information on the POWERLINK implemented abort codes please refer to the "SDO Abort Codes" section in the "EPSP DSP 301 V1.2.0" document.

Abort code	Description
0503 0000h	Reserved
0504 0000h	SDO protocol timed out
0504 0001h	Client/server Command ID not valid or unknown
0504 0002h	Invalid block size
0504 0003h	Invalid sequence number
0504 0004h	Reserved
0504 0005h	Out of memory
0601 0000h	Unsupported access to an object
0601 0001h	Attempt to read a write-only object
0601 0002h	Attempt to write a read-only object
0602 0000h	Object does not exist in the object dictionary
0604 0041h	Object cannot be mapped to the PDO
0604 0042h E_PDO_MAP_ OVERRUN	The number and length of the objects to be mapped would exceed PDO length
0604 0043h	General parameter incompatibility
0604 0044h E_NMT_INVALID_ _HEARTBEAT	Invalid heartbeat declaration
0604 0047h	General internal incompatibility in the device
0606 0000h	Access failed due to an hardware error
0607 0010h	Data type does not match, length of service parameter does not match
0607 0012h	Data type does not match, length of service parameter too high
0607 0013h	Data type does not match, length of service parameter too low
0609 0011h	Sub-Index does not exist
0609 0030h	Value range of parameter exceeded (only for write access)
0609 0031h	Value of parameter written too high
0609 0032h	Value of parameter written too low
0609 0036h	Maximum value is less than minimum value
0800 0000h	General error
0800 0020h	Data cannot be transferred or stored to the application
0800 0021h	Data cannot be transferred or stored to the application because of local

	control
0800 0022h	Data cannot be transferred or stored to the application because of the present device state
0800 0023h	Object dictionary dynamic generation fails or no object dictionary is present (e.g. object dictionary is generated from file and generation fails because of a file error)
0800 0024h E_CFM_DATA_ SET_EMPTY	EDS, DCF or Concise DCF Data set empty

The abort codes not listed here are reserved.

7 Default parameters list

Default values are expressed in hexadecimal notation, unless otherwise indicated.

7.1 Communication Profile Area objects

Parameters list	Default values		
1000 NMT_DeviceType_U32	0001 0196 * = singleturn encoder, HS58 series 0002 0196 * = multiturn encoder, EM/HM58 series		
1001 ERR_ErrorRegister_U8	00		
1006 NMT_CycleLen_U32	000F 4240		
1008 NMT_ManufactDevName_VS	EMxxx13/16384-PL-xx * = EM58 series HSxxx18-PL-xx * = HS58 series HMxxx16/16384-PL-xx * = HM58 series		
1009 NMT_ManufactHwVers_VS	Device dependent		
100A NMT_ManufactSwVers_VS	Device dependent		
1018-01 NMT_IdentityObject_REC.VendorID_U32	0000 012E		
1018-02 NMT_IdentityObject_REC.ProductCode_U32	0000 1000 = HM58 series 0000 1001 = EM58 series 0000 1002 = HS58 series		
1018-03 NMT_IdentityObject_REC.RevisionNo_U32	0000 0001		
1018-04 NMT_IdentityObject_REC.SerialNo_U32	FFFF FFFF		
1020-01 CFM_VerifyConfiguration_REC.ConfDate_U32	0000 0000		
1020-02 CFM_VerifyConfiguration_REC.ConfTime_U32	0000 0000		
1030-01 NMT_InterfaceGroup_0h_REC.InterfaceIndex_U16	0001		
1030-02 NMT_InterfaceGroup_0h_REC.InterfaceDescription_VSTR	LIKA ROTACOD EMxxx13/16384-PL-xx 1.0 * = EM58 series LIKA ROTACOD HSxxx18-PL-xx 1.0 * = HS58 series LIKA ROTACOD HMxxx16/16384-PL-xx 1.0 * = HM58 series		
1030-03 NMT_InterfaceGroup_0h_REC.InterfaceType_U8	06		
1030-04 NMT_InterfaceGroup_0h_REC.Interface	05DC		

1030-05 NMT_InterfaceGroup_0h_REC.InterfacePhysAddress_OSTR	not available		
1030-06 NMT_InterfaceGroup_0h_REC.InterfaceName_VSTR	Interface 1 *		
1030-07 NMT_InterfaceGroup_0h_REC.InterfaceOperState_U8	01		
1030-08 NMT_InterfaceGroup_0h_REC.InterfaceAdminState_U8	01		
1030-09 NMT_InterfaceGroup_0h_REC.Valid_BOOL	1 ** = TRUE		
1300 SDO_SequLayerTimeout_U32	0000 3A98		
1800-01 PDO_TxCommParam_00h_REC.NodeID_U8	00		
1800-02 PDO_TxCommParam_00h_REC.MappingVersion_U8	00		
1A00-01 PDO_TxMappParam_00h_AU64.ObjectMapping	0020 0000 0000 6004		
1A00-02 PDO_TxMappParam_00h_AU64.ObjectMapping	0020 0020 0000 6030		
1C0B-01 DLL_CNLossSoC_REC.CumulativeCnt_U32	0000 0000		
1C0B-02 DLL_CNLossSoC_REC.ThresholdCnt_U32	0000 0000		
1C0B-03 DLL_CNLossSoC_REC.Threshold_U32	0000 000F		
1C0FB-01 DLL_CNCRCErrror_REC.CumulativeCnt_U32	0000 0000		
1C0FB-02 DLL_CNCRCErrror_REC.ThresholdCnt_U32	0000 0000		
1C0FB-03 DLL_CNCRCErrror_REC.Threshold_U32	0000 000F		
1C14 DLL_CNLossOfSocTolerance_U32	0001 86A0		
1F81-01 NMT_NodeAssignment_AU32.NodeAssignment	0000 0000		
1F82 NMT_FeatureFlags_U32	0004 8205		
1F83 NMT_EPLVersion_U8	20		
1F8D-01 NMT_PResPayloadLimitList_AU16.PResPayloadLimit	0024		
1F93-01	01		

1F93-02 NMT_EPLNodeID_REC.NodeIDByHW_BOOL	1 ** = TRUE		
1F98-01 NMT_CycleTiming_REC.IsochrTxMaxPayload_U16	05D2		
1F98-02 NMT_CycleTiming_REC.IsochrRxMaxPayload_U16	05D2		
1F98-03 NMT_CycleTiming_REC.PresMaxLatency_U32	0000 03E8		
1F98-04 NMT_CycleTiming_REC.PreqActPayloadLimit_U16	0024		
1F98-05 NMT_CycleTiming_REC.PresActPayloadLimit_U16	0024		
1F98-06 NMT_CycleTiming_REC.AsndMaxLatency_U32	0000 03E8		
1F98-07 NMT_CycleTiming_REC.MultiplCycleCnt_U8	00		
1F98-08 NMT_CycleTiming_REC.AsyncMTU_U16	012C		
1F99 NMT_CNBasicEthernetTimeout_U32	004C 4B40		
1F9B-01 NMT_MultiplCycleAssign_AU8.CycleNo	00		
1F9E NMT_ResetCmd_U8	FF		

* String value

** Boolean value

7.2 Manufacturer Specific Profile Area objects

Parameters list	Default values		
2002 Speed format	0000 = steps/s		

7.3 Standardized Device Profile Area objects

Parameters list	Default values		
6000 Operating parameters	0000		
6001 Measuring units per revolution	0000 2000 = EM58 series 0004 0000 = HS58 series 0001 0000 = HM58 series		
6002 Total measuring range in measuring units	0800 0000 = EM58 series 0004 0000 = HS58 series 4000 0000 = HM58 series		
6003 Preset value	0000 0000		
6500 Operating status	0000		
6501 Singleturn resolution	0000 2000 = EM58 series 0004 0000 = HS58 series 0001 0000 = HM58 series		
6502 Number of revolutions	0000 4000 = EM58 series 0000 0001 = HS58 series 0000 4000 = HM58 series		
6504 Supported alarms	7000		
6506 Supported warnings	0000		
6507 Profile and software version	0301 device dependent		
6508 Operating time	FFFF FFFF		
650B Serial number	FFFF FFFF		

Document release	Release date	Description	HW	SW	XDD file version
1.0	16.12.2015	First issue	1.0	1.0	1.0
1.1	03.03.2022	Axial connectors outputs added, general review	1.0	2.0	1.0, 2.0
1.2	18.09.2023	New order codes, phase-out product information	1.0	2.0	1.0, 2.0



This device is to be supplied by a Class 2 Circuit or Low-Voltage Limited Energy or Energy Source not exceeding 30 Vdc. Refer to the order code for supply voltage rate.

Ce dispositif doit être alimenté par un circuit de Classe 2 ou à très basse tension ou bien en appliquant une tension maxi de 30Vcc. Voir le code de commande pour la tension d'alimentation.



Dispose separately

lika

Lika Electronic

Via S. Lorenzo, 25 • 36010 Carrè (VI) • Italy

Tel. +39 0445 806600

Fax +39 0445 806699



info@lika.biz • www.lika.biz