Pacdrive 3 - Schneider Electric

MACHINE-EXPERT-KONFIGURATION

Beispiel für die Konfiguration für den Linearmotor L050P1215-GH.

Bitte berücksichtigen Sie, wenn der Motor anders ist, die richtigen Daten mit dem Datenblatt-Engine: https://www.ni-lab.online/new_websmart/read_datasheet.php

```
st_UserMotorData_01.etMotorType := MTP.ET_MotorType.LinearPMSM;
st_UserMotorData_01.sMotorname := 'NiLAB GmbH';//Armonic Drive
st_UserMotorData_01.sMotorSerialNumber := '123456';
st_UserMotorData_01.sMotorArticleNumber := '789012';
st_UserMotorData_01.stMotorDataPMSM.uiEncoderType := mtp.ET_EncoderType.SincosHiperfaceLinear;
st_UserMotorData_01.stMotorDataPMSM.uiNominalSpeed := 3998; // mm/s
st_UserMotorData_01.stMotorDataPMSM.rNominalVoltage := 220.0;
st_UserMotorData_01.stMotorDataPMSM.rNominalCurrent := 2.1; // A
st_UserMotorData_01.stMotorDataPMSM.rPeakCurrent := 8.0; // A , OPTIONAL, std: Nom*1.5
st_UserMotorData_01.stMotorDataPMSM.rContStallCurrent := 2.1; // A
st_UserMotorData_01.stMotorDataPMSM.rConstStallTorque := 110; // N, OPTIONAL
st_UserMotorData_01.stMotorDataPMSM.rPeakTorque := 880; // Nm, OPTIONAL
st_UserMotorData_01.stMotorDataPMSM.rPhaseResistance := 12; // Ohm , OPTIONAL
st_UserMotorData_01.stMotorDataPMSM.rQuadraturePhaseInductance := 18000; // uH , OPTIONAL
st_UserMotorData_01.stMotorDataPMSM.rDirectPhaseInductance := 16200; // uH
st_UserMotorData_01.stMotorDataPMSM.uiRotatingFieldDirection := 0; // OPTIONAL, std: 0
st_UserMotorData_01.stMotorDataPMSM.rEMK_Constant := 18;

st_UserMotorData_01.stMotorDataPMSM.uiTempSensorType := 1;
st_UserMotorData_01.stMotorDataPMSM.uiTempSensorResistanceOvertemp := 4000; // Ohm, OPTIONAL (verw bei SensorType = 1), std: 4000
st_UserMotorData_01.stMotorDataPMSM.uiMaxSpeed := 4500; // mm/s
st_UserMotorData_01.stMotorDataPMSM.udiMotorInertia := 1500;
st_UserMotorData_01.stMotorDataPMSM.uiBrake := 0;

// For Linear Motors

st_UserMotorData_01.stMotorDataPMSM.rPeriodLength := 2000.0;
st_UserMotorData_01.stMotorDataPMSM.rPolePairPitch := 32000.0;
```

Innerhalb des Antriebsabschnitts setzen Sie MotorIdentification auf Motor **ohne Typenschild/2** und MotorTemperatureMonitoring auf **Sensor / 2** (eine Brücke an den Temperaturanschlüssen am Leistungsport muss angeordnet werden).

Die Motordaten müssen im Antriebsspeicher gespeichert werden, mit diesen Anweisungen (ASSE_UNO ist der Name des Lexium LXM62 Linear Drive)

```
IF (xFile = TRUE) THEN
xFileCreation:= MTP.FC_MotorDataFileCreate(i_sFilename:= 'ide0:MDF/UserMotorData.mdf',i_stUserMotorData:= st_UserMotorData_01, q_sMsg => sMsg_t); END_IF;

FB_MotorDataDelete_inst(i_xEnable:=xEnable_Delete, i_xExecute:=xExecute_Delete, i_ifDrive := ASSE_UNO, i_etStorageLocation := mtp.ET_StorageLocation.Drive, i_xForce := TRUE, i_xDeleteAll:=TRUE, q_sMsg => Delete_sMsg_t);

FB_MotorDataWrite_inst(i_xEnable:=xEnable_Write, i_xExecute:=xExecute_Write,  i_ifDrive := ASSE_UNO, i_sFilename:='ide0:MDF/UserMotorData.mdf', i_etStorageLocation := mtp.ET_StorageLocation.Drive, q_sMsg => Write_sMsg_t);
```

Um die Motordaten zu speichern, schalten Sie die Sercos-Phase auf 2. Erstellen Sie die mdf-Datei, setzen Sie die xFile-Variable auf TRUE, dann löschen Sie die Daten im Antrieb, indem Sie xEnable_Delete und xExecute_Delete auf TRUE setzen. Schließlich schreiben Sie die Motordaten in den Antrieb, indem Sie xExecute_write und xEnable_Write auf TRUE setzen. Nach der erfolgreichen Prozedur schalten Sie die Stromversorgung des Encoders und des Antriebs aus. Wenn der Antrieb neu gestartet wird, setzen Sie die SERCOS-Phase auf 4. Der Servoregler muss ohne Fehler sein und den Encoder-Positionswert ohne Fehler anzeigen.

Jetzt wählen Sie das PSM_Lexium62-Stromversorgungsfenster, stellen Sie die Stromleitungen (400V) zum Antrieb bereit und setzen Sie PowerSupplyCheckSet auf 1. Gehen Sie zurück zum Antriebsabschnitt und berechnen Sie den Kommutierungswinkel, indem Sie MotorCommutationMode auf **mit Bewegung / 0** setzen und MotorCommutationControl auf **erkennen und schreiben / 1** setzen.

STECKVERBINDER-PINOUT

WARNUNG: ENTFERNEN SIE NICHT DIE 24 VDC VOM ANTRIEB, OHNE DIE 24 VDC AUF DEN ENCODER-STROMLEITUNGEN ZU ENTFERNEN.

Für die Verwendung mit dem LXM62D-Servoregler verwenden Sie bitte das Adapterkabel VW3E2092R005 (Feedback-Adapter SH3/MH3-Servomotor L=0,5 m) in Kombination mit unserem DE20600780-5M (5 Meter Revenc-Encoderkabel für LXM62-Servoregler) mit DSUB9-Stecker männlich.

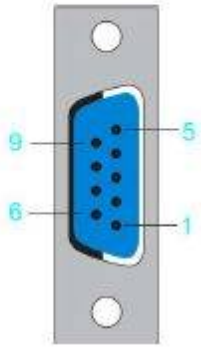
Signal	DE20600780 D-SUB 9 Pin	VW3E2092R005 D-SUB 9 Pin	Feedback-Steckverbinder am LXM62D-Servo (RJ45 + Stromleitungen)
SIN-	6	1	6
SIN+	3	2	3
COS-	2	3	2
COS+	1	4	1
DATA-	5	6	5
DATA+	4	7	4
GND	7	9	B
NC	-	-	A (Encoder-Stromversorgung ist nicht verbunden, der Encoder wird extern mit 24VDC versorgt)

Beachten Sie, dass bei Verwendung von Machine Expert die Feed-Konstante des Encoders auf die verwendete Encoder-Konfiguration gesetzt werden muss.
Zum Beispiel, mit einem Ausgangs-Sinusperiode von 2 mm muss die Feed-Konstante in Mechanik auf 1000 gesetzt werden.

D-Sub 9 Pin Buchsensteckverbinder-Pinout VW3E2092R005

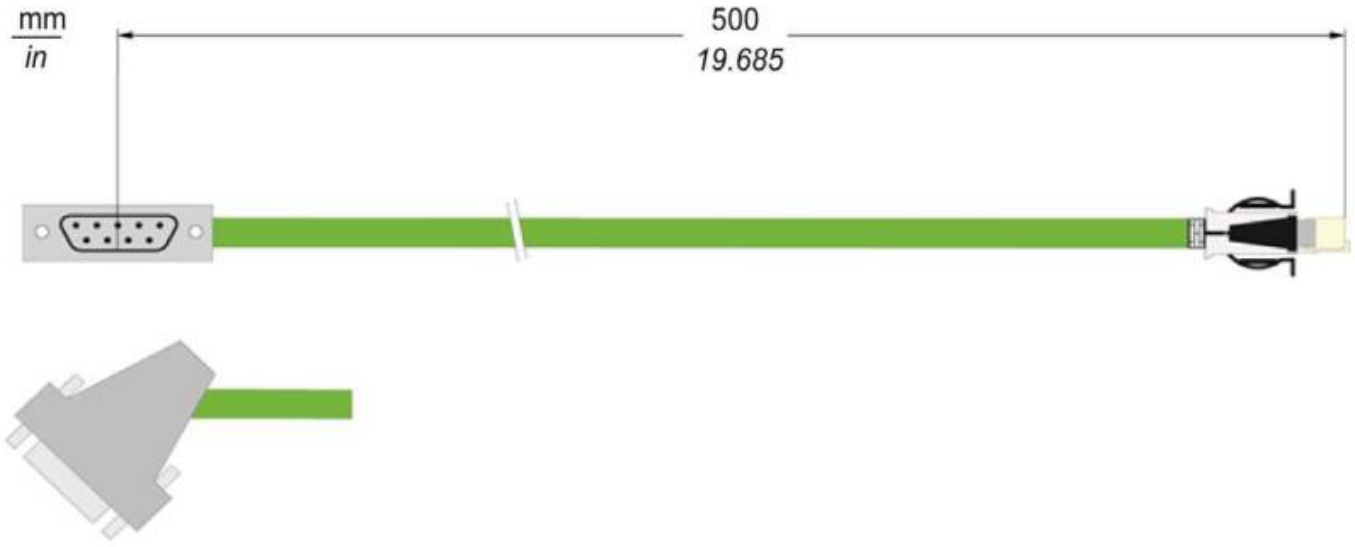
D-Sub 9-Pin Female Connector - 5 V Encoder Adapter Output

The D-Sub 9-pin female connector is connected to the D-Sub 9-pin male connector of the encoder cable (user furnished).



Electrical connection D-Sub 9-pin female connector

Pin	Designation	Description	Range
1	SIN	Positive sine signal	1 V_{pp} ±0.1 V
2	Ref_Sin	Negative sine signal	Offset 2.5 ±0.3 V
3	COS	Positive cosine signal	1 V_{pp} ±0.1 V
4	Ref_Cos	Negative cosine signal	Offset 2.5 ±0.3 V
5	RS485+	Positive RS-485 signal	–
6	P5V	5 V encoder supply voltage	5 V ±1% / I_{out_max}=250 mA
7	P10V	10 V encoder supply voltage	10 V ±5% / I_{out_max}=125 mA
8	RS485-	Negative RS-485 signal	–
9	GND	Encoder return	0 V



From:
https://www.nilab.at/dokuwiki/ - **NiLAB GmbH Knowledgebase**

Permanent link:
https://www.nilab.at/dokuwiki/doku.php?id=de:reverso_encoder:pd3_servo

Last update: **2026/09/12 14:02**