

JVL Profinet Library Manual

Siemens S7-1200 & S7-1500 PLC Function Block Library for
the control of JVL Motors

V1.2 04/2022

Contents

1	Note	3
2	Configuration	4
2.1	Requirements.....	4
2.2	Import Library	4
2.3	Device Configuration.....	6
2.4	MacTalk Configuration.....	11
3	Common Function Blocks (for MIS & MAC).....	14
3.1	JVL_ReadWriteReg	14
4	MIS Function Blocks	16
4.1	JVL_MIS_Comms.....	16
4.2	JVL_MIS_Disable	17
4.3	JVL_MIS_Halt	19
4.4	JVL_MIS_Home	21
4.5	JVL_MIS_Jog.....	24
4.6	JVL_MIS_Position	26
4.7	JVL_MIS_Reset	29
4.8	JVL_MIS_Velocity	30
5	MAC Function Blocks	33
5.1	JVL_MAC_Comms	33
5.2	JVL_MAC_Disable.....	34
5.3	JVL_MAC_Halt.....	36
5.4	JVL_MAC_Home.....	38
5.5	JVL_MAC_Jog	42
5.6	JVL_MAC_Position	44
5.7	JVL_MAC_Reset	47
5.8	JVL_MAC_Velocity	48

1 Note

Please note that this library has been tested with MacTalk V1.90.21 and JVL Profinet firmware V3.40.

2 Configuration

2.1 Requirements

TIA Portal V14 or higher.

TIA Portal project containing an S7-1200 or S7-1500 PLC.

MacTalk project for the JVL motor configuration.

2.2 Import Library

To import the JVL Profinet Function Block library into TIA Portal, right-click in the Global Libraries section of the Library sidebar in TIA Portal, and select 'Retrieve Library...'. Select the library archive (.zal14 file) then choose the destination to save the extracted library files.

When opened with later versions of TIA Portal the library will be updated to the required version automatically.

2.2.1 Add Library Components

2.2.1.1 User data Types (UDTs)

The UDTs are required for the function blocks to be compiled. In the Master Copies folder within the library, drag the 'JVL UDT' group into the PLC's 'PLC data types' folder.

2.2.1.2 Common Function Blocks

The 'JVL Common' function blocks group are blocks that can be used, or are required for, both MIS and MAC devices. This group includes:

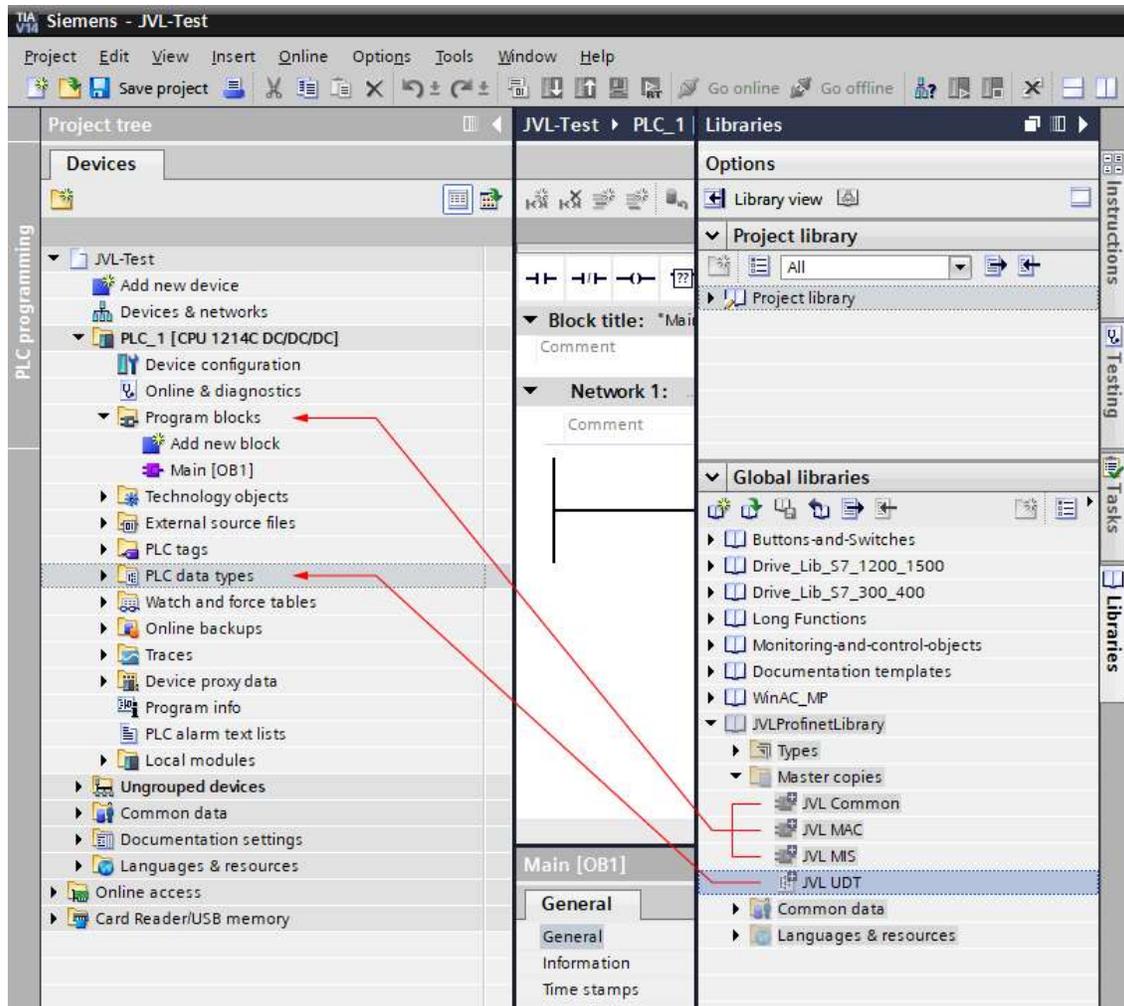
JVL_ReadWriteReg - Function block for reading or writing motor registers.

The 'JVL Common' group should be dragged into the PLC's 'Program blocks' folder.

2.2.1.3 Device Specific Function Blocks

If using a MIS device, drag the 'JVL MIS' group into the PLC's 'Program blocks' folder.

If using a MAC device, drag the 'JVL MAC' group into the PLC's 'Program blocks' folder.



2.3 Device Configuration

2.3.1 Import GSD (General Station Description) File

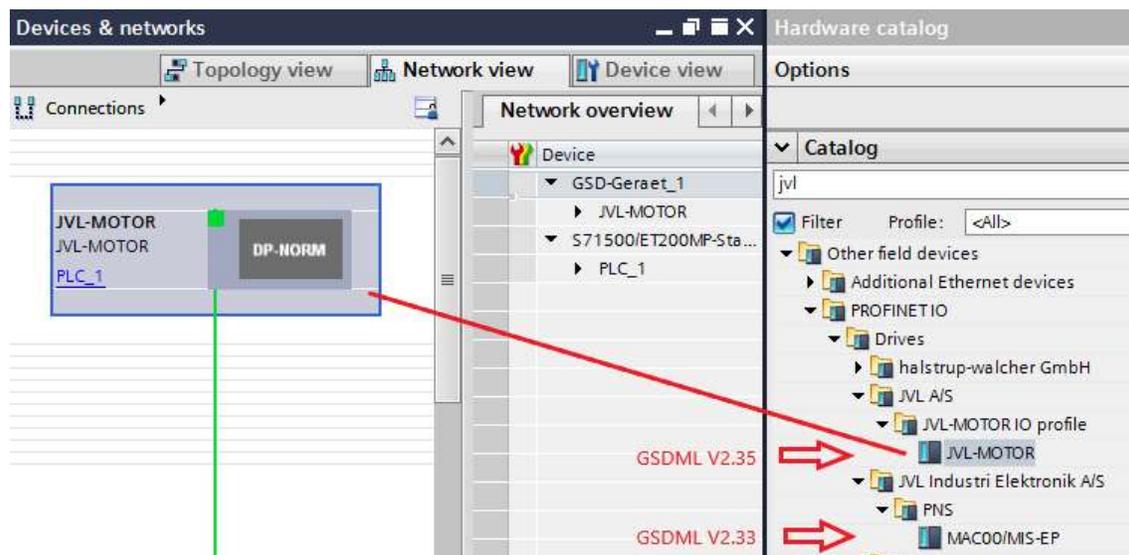
For each JVL device to be configured, a GSD file must be included in the TIA Portal project. The GSD file can be downloaded from the JVL website [GSD section](#). When downloading from the JVL website, please select the 8 register Profinet GSD file.

Profinet configuration:

Profinet - GSD file - 8 regs. PZD (2020-12-10 - From FW 3.40)

[Download](#)

Firstly, the GSD must be installed. Using the Options > Manage general station description files (GSD) menu option. Select the GSD file for installation into TIA Portal. Once the GSD is installed, it can be added to the project from the Hardware Catalog sidebar. It can be found under Other field devices > PROFINET IO > Drives > JVL A/S > JVL-MOTOR IO profile folder. For each JVL drive/axis, drag a JVL_MOTOR GSD into the Devices & Networks window in the TIA Portal project.

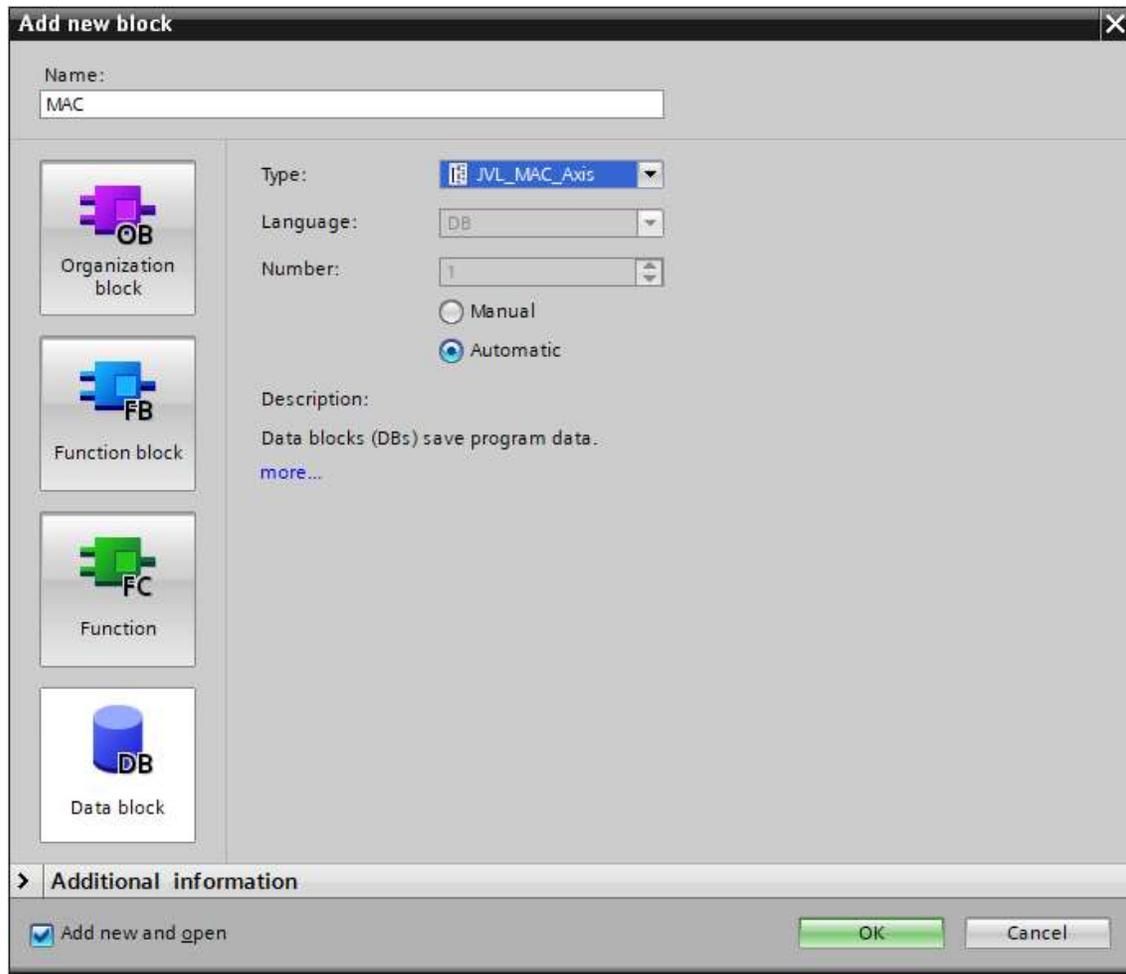


Remember to set the IP and Profinet name of each of the JVL motors and connect them to the PLC's subnet.

2.3.2 Data Blocks

For each JVL device/axis, a data block must be created of the correct type. A data block can be created using the 'Add new block' command, located under the 'Program blocks' PLC folder. For MAC devices, create a data block of type JVL_MAC_Axis.

For MIS devices, create a data block of type JVL_MIS_Axis.

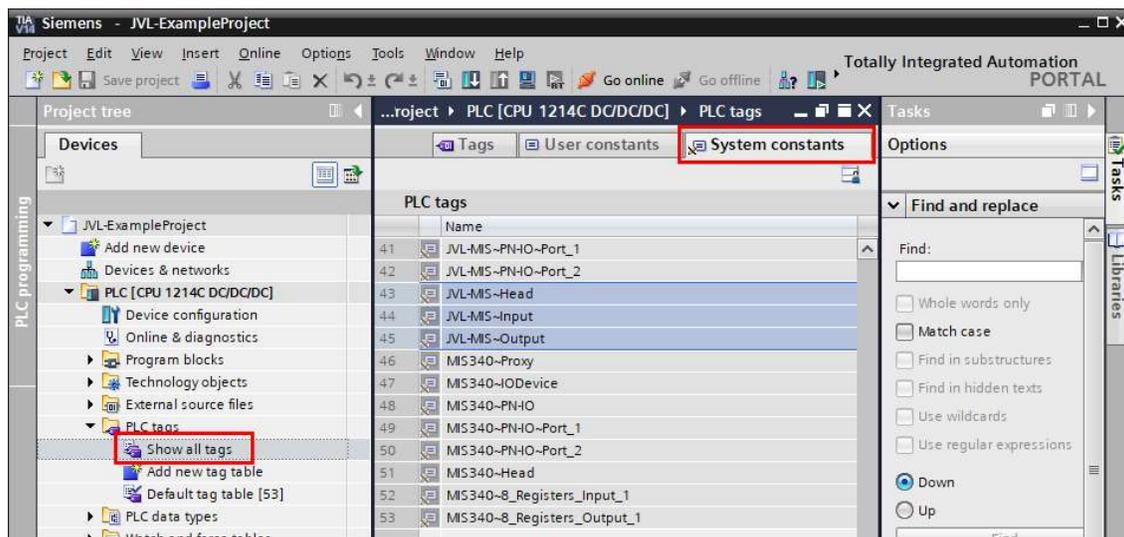


2.3.2.1 Data block Configuration

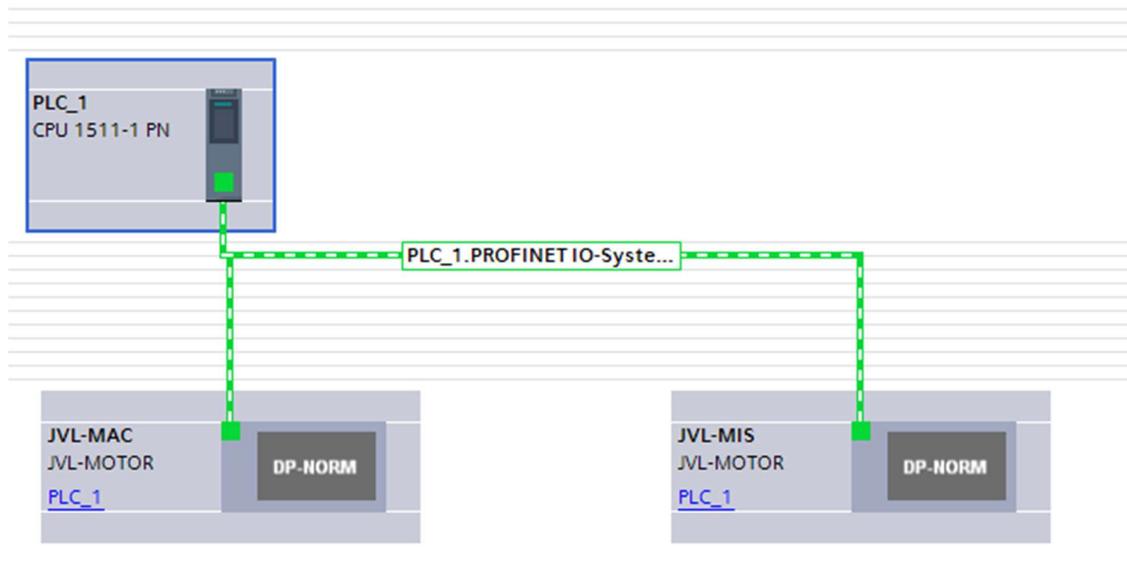
Each of the axis data blocks must be configured, in order to link the data block to its corresponding JVL motor. The Device > Config section within the data block contains parameters which must be set. For the MAC you also have to configure the Motortype. With the Motortype you set the acceleration and velocity scaling factors for the MAC.

Name	Description
DeviceId (MAC and MIS)	Used to define the main address of the unit, used to perform register reads and writes.
TelegramInputAddress (MAC and MIS)	Used to define the address of the JVL input telegram.
TelegramOutputAddress (MAC and MIS)	Used to define the address of the JVL output telegram.
MotorType (only MAC)	0 = MAC50-141F, MAC400/402, MAC1500-4500; 1 = MAC800-1200; 2 = MAC50-141A/K (old)
A_SOLL_CountsPerSample2 (only MAC)	Acceleration scaling value. (For MAC motors only)
V_SOLL_CountsPerSample (only MAC)	Velocity scaling value. (For MAC motors only)

The device ID and telegram input and output address values can be found in the PLC Tags > Show All Tags > System Constants section.



For each JVL motor in the project, you will find a HEAD, INPUT and OUTPUT tags, for example:

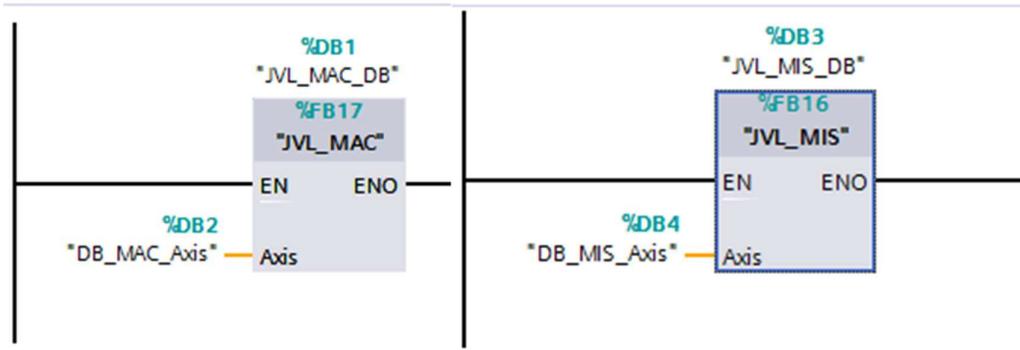


Add these 3 tags to the data block Device > Config section.

DB_MS_Axis				DB_MAC_Axis			
	Name	Datentyp	Startwert		Name	Datentyp	Startwert
1	Static			1	Static		
2	Device	Struct		2	Device	Struct	
3	Config	Struct		3	Config	Struct	
4	DeviceId	HW_IO	*JVL-MS-Head*	4	DeviceId	HW_IO	*JVL-MAC-Head*
5	TelegramInputAddress	HW_IO	*JVL-MS-8_Registers_Eingang_1*	5	TelegramInputAddress	HW_IO	*JVL-MAC-8_Registers_Input_1*
6	TelegramOutputAddress	HW_IO	*JVL-MS-8_Registers_Ausgang_1*	6	TelegramOutputAddress	HW_IO	*JVL-MAC-8_Registers_Output_1*
7	CommsOK	Bool	false	7	MotorType	Int	0
8	CommandId	DWord	16#0	8	V_SOLL_CountsPerSample	LReal	0.0
9	Telegram	Struct		9	A_SOLL_CountsPerSample2	LReal	0.0
				10	CommsOK	Bool	false
				11	CommandId	DWord	16#0
				12	Telegram	Struct	

For the MAC-Motors set also the MotorType (0 = MAC50-141F, MAC400/402, MAC1500-4500; 1 = MAC800-1200; 2 = MAC50-141A/K (old)).

These data blocks will be used as the Axis input on all the motion function blocks in this library. For example:



When you call these function blocks in Main[OB1] then you can control these motors by the instance Datablocks JVL_MAC_DB and JVL_MIS_DB.

JVL_MAC_DB			JVL_MIS_DB		
	Name	Datentyp		Name	Datentyp
1	Input		1	Input	
2	Output		2	Output	
3	InOut		3	InOut	
4	Axis	"JVL_MAC_Axis"	4	Axis	"JVL_MIS_Axis"
5	Static		5	Static	
6	▶ Disable	"JVL_MAC_Disable"	6	▶ Disable	"JVL_MIS_Disable"
7	▶ Halt	"JVL_MAC_Halt"	7	▶ Halt	"JVL_MIS_Halt"
8	▶ Velocity	"JVL_MAC_Velocity"	8	▶ Velocity	"JVL_MIS_Velocity"
9	Input		9	Position	"JVL_MIS_Position"
10	Execute	Bool	10	Input	
11	ContinuousUp...	Bool	11	Execute	Bool
12	Velocity	LReal	12	ContinuousUp...	Bool
13	Acceleration	LReal	13	Position	DInt
14	Direction	Bool	14	Velocity	LReal
15	▶ Output		15	Acceleration	LReal
16	▶ InOut		16	Deceleration	LReal
17	▶ Static		17	AbsolutePositi...	Bool
18	▶ Position	"JVL_MAC_Position"	18	▶ Output	
19	▶ Jog	"JVL_MAC_Jog"	19	▶ InOut	
20	▶ Home	"JVL_MAC_Home"	20	▶ Static	
21	▶ Reset	"JVL_MAC_Reset"	21	▶ Jog	"JVL_MIS_Jog"
22	▶ Reg	"JVL_ReadWriteReg"	22	▶ Home	"JVL_MIS_Home"
23	ActuaTorquePercentage	LReal	23	▶ Reset	"JVL_MIS_Reset"
24	ActualVelocityRPM	LReal	24	▶ Reg	"JVL_ReadWriteReg"
25	ActualPosition	DInt	25	ActualPosition	DInt

2.4 MacTalk Configuration

In the MacTalk project for your JVL device, a specific telegram must be set in order to communicate with the Profinet library function blocks. The contents of this telegram depend on the type of JVL drive to be used. Use the cyclic data setup section of the MAC00-EP Profinet tab in MacTalk to configure the telegram.

Once configured, please remember to click the Apply and Save button in order to save the telegram configuration in the drive.

2.4.1 For MAC50-MAC141 A/K old Devices (older smaller MACs)

Main	Registers	Advanced	Tests	Scope	MAC00-EP -ProfiNet	Homing
Setup						
Ethernet settings			Cyclic data setup			
IP address	192.168.0.0	192.168. 25.201	Read Word1	2 - Operating mode		
Subnet mask	255.255.255.0	255.255.255. 0	Read Word2	10 - Actual position		
Default gateway	192.168.0.158	192.168. 25.201	Read Word3	12 - Actual velocity		
<input type="checkbox"/> Use DHCP to obtain IP address			Read Word4	121 - Actual torque		
ProfiNet error handling			Read Word5	35 - Error status		
<input checked="" type="radio"/> Motor set "Passive mode"			Read Word6	168 - Homing done		
<input type="radio"/> Motor set velocity = 0			Read Word7	0 - No Selection		
Protocol settings			Read Word8	0 - No Selection		
Enter the Station Name (Max 240 characters):			Write Word1	2 - Operating mode		
<input type="text" value="jvl-mac"/>			Write Word2	3 - Requested position		
<input type="checkbox"/> Powerup with blank 'Name of station'			Write Word3	5 - Velocity		
<input type="checkbox"/> Alt. Byte Order			Write Word4	6 - Acceleration		
			Write Word5	983040 - General command		
			Write Word6	0 - No Selection		
			Write Word7	0 - No Selection		
			Write Word8	0 - No Selection		
			Set module factory defaults		Apply and save	

2.4.2 For MAC50F-MAC141F (F-Type) and MAC400 - MAC4500 Motors

Main	I/O Setup	Registers	Advanced	Tests	Scope	ePLC™	MAC00-EP -ProfiNet	Homing	Units (Disabled)	
Setup										
Ethernet settings					Cyclic data setup (32bit)					
IP address	255.255.255.255	192.168. 25.201		Read Word1	2 - Operating mode					
Subnet mask	255.255.255.255	255.255.255. 0		Read Word2	10 - Actual position					
Default gateway	255.255.255.255	192.168. 25.201		Read Word3	11 - Actual Velocity 16bit					
<input type="checkbox"/> Use DHCP to obtain IP address				Read Word4	169 - Actual torque					
ProfiNet error handling					Read Word5	35 - Error status				
<input checked="" type="radio"/> Motor set "Passive mode"					Read Word6	42 - Home mode				
<input type="radio"/> Motor set velocity = 0					Read Word7	0 - No Selection				
Protocol settings					Read Word8	0 - No Selection				
Sercos Address			0	Write Word1	2 - Operating mode					
Poll division			0	Write Word2	3 - Requested position					
ModbusTCP timeout			0	Write Word3	5 - Velocity					
<input type="checkbox"/> Powerup with blank 'Name of station'					Write Word4	6 - Acceleration				
<input type="checkbox"/> Alt. Byte Order					Write Word5	983040 - General command				
<input type="checkbox"/> Use I/O in ePLC™					Write Word6	0 - No Selection				
					Write Word7	0 - No Selection				
					Write Word8	0 - No Selection				
					Set module factory defaults		Apply and save			

2.4.3 For MIS Devices

Main	I/O Setup	Registers	DMX512 Setup	Advanced	Scope	ePLC™	ProfiNet	Homing	Units (Disabled)
------	-----------	-----------	--------------	----------	-------	-------	----------	--------	------------------

Setup

Ethernet settings

IP address: 255.255.255.255 | 192.168. 25.200

Subnet mask: 255.255.255.255 | 255.255.255. 0

Default gateway: 255.255.255.255 | 192.168. 25.200

Use DHCP to obtain IP address

ProfiNet error handling

Motor set "Passive mode"

Motor set velocity = 0

Protocol settings

Sercos Address:

Pol division:

ModbusTCP timeout:

Powerup with blank 'Name of station'

Alt. Byte Order

Use I/O in ePLC™

Cyclic data setup (32bit)

Read Word1	2 - Operating Mode	▼
Read Word2	10 - Projected Position	▼
Read Word3	12 - Actual Velocity	▼
Read Word4	25 - Status Bits	▼
Read Word5	36 - Warnings	▼
Read Word6	35 - Errors	▼
Read Word7	0 - No Selection	▼
Read Word8	0 - No Selection	▼
Write Word1	2 - Operating Mode	▼
Write Word2	3 - Requested Position	▼
Write Word3	5 - Max Velocity	▼
Write Word4	6 - Acceleration	▼
Write Word5	174 - Deceleration	▼
Write Word6	983040 - General command	▼
Write Word7	0 - No Selection	▼
Write Word8	0 - No Selection	▼

3 Common Function Blocks (for MIS & MAC)

3.1 JVL_ReadWriteReg

3.1.1 Description

Read or write a motor register.

This function block performs acyclic read/write of motor registers. Acyclic operations take multiple PLC cycles to complete. Care must be taken to ensure that only one acyclic operation is active on the Profinet network at any one time. For systems with multiple calls of the JVL_ReadWriteReg and other software functions that perform acyclic read/write access it is highly recommended that you implement a resource manager to prevent conflicts.

A resource manager is provided in the Siemens library LAcycCom available for download:

<https://support.industry.siemens.com/cs/gb/en/view/109479553>

3.1.2 Inputs

Input	Data Type
Execute	Bool
Register	Int
WriteValue	DWord
WriteRegister	Bool
DeviceID	HW_IO

3.1.2.1 Execute

A rising edge is required to start the read/write command. If the input is given a rising edge whilst a command is currently running, then the current command is overridden/aborted. The input is not required to remain TRUE for the duration of the command. Setting the value of Execute to FALSE during the command will not stop the command.

3.1.2.2 Register

The register number to be read or written to. A full list of register numbers is available in the [Ethernet Industrial modules manual](#) appendix 12.2/12.3.

3.1.2.3 WriteValue

The new value of the register to be written.

3.1.2.4 WriteRegister

Choose between reading or writing a register.

Value	Description
TRUE	Write Register
FALSE	Read Register

3.1.2.5 DeviceID

The DeviceID input should be linked to a data block of type Axis.Device.Config.DeviceID. This data block provides axis references to the unit.

Use the axis data block device section for this input.
For example: MIS_AXIS.Device.Config.DeviceID

3.1.3 Outputs

Output	Data Type
Busy	Bool
Done	Bool
Error	Bool
ReadValue	DWord

3.1.3.1 Error

Error is TRUE when an error occurs during the reading or writing of a register value.

3.1.3.2 Busy

Busy is TRUE whenever the command is in the process of reading or writing to a register and the Error output is FALSE.

3.1.3.3 Done

Done is TRUE when the read/write command has completed successfully.

3.1.3.4 ReadValue

The value of the register after reading.

4 MIS Function Blocks

4.1 JVL_MIS_Comms

4.1.1 Description

Performs cyclic Profinet communication. The cyclic telegram data is transferred between PLC input/output area and the axis data block.

This function is required for all other motion control function blocks to operate. A single instance of the JVL_MIS_Comms function must be called once per PLC cycle for every JVL motor in the system.

4.1.2 InOut

InOut	Data Type
Axis	JVL_MIS_AXIS

4.1.2.1 Axis

The axis InOut should be linked to a data block of type JVL_MIS_Axis. This data block provides axis references to the unit and to its telegram slots.

4.1.3 Return Value

The function returns a TRUE value if the communication is operating without fault.

In the event of a fault (Ret_Val = FALSE) the return values of DPWR_DAT and DPRD_DAT inside this function must be evaluated to determine the cause.

4.2 JVL_MIS_Disable

4.2.1 Description

Place the drive in passive mode, removing all torque from the motor.

If the motor is running then it stops with the configured deceleration of the command that is in progress.

Caution, this can result in a long stopping distance if the configured deceleration is low. It is recommended that JVL_MIS_Halt is used with suitable deceleration before then executing JVL_MIS_Disable.

4.2.2 Inputs/InOut

Input	Data Type
Execute	Bool
Axis	JVL_MIS_AXIS

4.2.2.1 Execute

A rising edge is required to start the motion command. If the input is given a rising edge whilst a motion command is currently running, then the current motion command is overridden/aborted. The input is not required to remain TRUE for the duration of the motion command.

4.2.2.2 Axis

The axis input should be linked to a data block of type JVL_MIS_Axis. This data block provides axis references to the unit and to its telegram slots.

4.2.3 Outputs

Output	Data Type
Busy	Bool
Done	Bool
Command Aborted	Bool
Error	Bool

4.2.3.1 Busy

Busy is TRUE whenever the motion command is running and the Error output is FALSE.

4.2.3.2 Done

Done is TRUE when the motor has been successfully disabled.

4.2.3.3 Command Aborted

The command was interrupted by the execution of another command.

4.2.3.4 Error

Error is TRUE if there is an error in the drive while the motion command is running, given by the Error signal in the Error word (register 35 Error bit 0).

4.3 JVL_MIS_Halt

4.3.1 Description

Brings the motor to a standstill with defined deceleration.

4.3.2 Inputs/InOut

Input	Data Type	Units
Execute	Bool	
Deceleration	LReal	RPM/s
Axis	JVL_MIS_AXIS	

4.3.2.1 Execute

A rising edge is required to start the motion command.

If the input is given a rising edge whilst a motion command is currently running, then the current motion command is overridden/aborted.

The input is not required to remain TRUE for the duration of the motion command.

Setting the value of Execute to FALSE during the motion command will not stop the motion command.

4.3.2.2 Deceleration

The target deceleration in RPM/s.

4.3.2.3 Axis

The axis input should be linked to a data block of type JVL_MIS_Axis. This data block provides axis references to the unit and to its telegram slots.

4.3.3 Outputs

Output	Data Type
Busy	Bool
Done	Bool
Command Aborted	Bool
Error	Bool

4.3.3.1 Busy

Busy is TRUE whenever the motion command is running (both when the motor is accelerating and when 'At Velocity') and the Error output is FALSE.

4.3.3.2 Done

Done is TRUE when the motor has come to a standstill.

4.3.3.3 Command Aborted

The command was interrupted by the execution of another command.

4.3.3.4 Error

Error is TRUE if there is an error in the drive while the motion command is running, given by the Error signal in the Error word (register 35 Error bit 0).

4.4 JVL_MIS_Home

4.4.1 Description

Homes or references the motor position.

4.4.2 Inputs

Input	Data Type	Units
Execute	Bool	
Mode	Int	
Position	Dint	Counts
Velocity	LReal	RPM
VelocityReduced	LReal	RPM
Acceleration	LReal	RPM/s
Deceleration	LReal	RPM/s
TorqueLimit	Real	%
Offset	Dint	Counts

4.4.2.1 Execute

A rising edge is required to start the motion command.

If the input is given a rising edge whilst a motion command is currently running, then the current motion command is overridden/aborted.

The input is not required to remain TRUE for the duration of the motion command.

Setting the value of Execute to FALSE during the motion command will not stop the motion command.

4.4.2.2 Mode

Set the homing method.

Value	Name
0	Set Home Position
1	Home to Sensor
2	Home to Endstop

0 - Set Home Position

The current motor actual position is set to the value of the Position input. The motor does not move. If the motor is running then it is stopped.

1 - Home to Sensor

The motor runs until it reaches the home sensor and then stops. The position of the motor is set relative to the point at which the home sensor was triggered. The exact homing operation is configured in MacTalk.

2 - Home to Endstop

The motor runs until the torque limit is reached. The position of the motor is set relative to the point at which the torque limit was reached.

4.4.2.3 Position

The new axis position, after referencing, in encoder counts. Only effective in homing mode 0.

4.4.2.4 Velocity

The target velocity for the homing move in RPM.

Only effective in homing modes 1 and 2.

4.4.2.5 VelocityReduced

The target velocity for the slow travel while the homing sensor is triggered in RPM. The operation of VelocityReduced depends on the homing configuration in MacTalk.

Only effective in homing mode 1.

4.4.2.6 Acceleration

The target acceleration in RPM/s.

Only effective in homing modes 1 and 2.

4.4.2.7 Deceleration

The target deceleration in RPM/s. If set to 0, the value in Acceleration is used instead.

Only effective in homing modes 1 and 2.

4.4.2.8 TorqueLimit

The torque threshold for the motor to stop and home position to be set in %.

Valid range 0 – 100%. Only effective in homing mode 2.

4.4.2.9 Offset

Relative distance for offset move following home to sensor or home to torque limit in counts. The operation of Offset depends on the homing configuration in MacTalk.

Only effective in homing modes 1 and 2.

4.4.2.10 Axis

The axis input should be linked to a data block of type JVL_MIS_Axis. This data block provides axis references to the unit and to its telegram slots.

4.4.3 Outputs

Output	Data Type
Busy	Bool
Done	Bool
Command Aborted	Bool
Error	Bool

4.4.3.1 Busy

Busy is TRUE whenever the motion command is running and the Error output is FALSE.

4.4.3.2 Done

Done is TRUE when the homing sequence has completed and the axis has been successfully referenced.

4.4.3.3 Command Aborted

The command was interrupted by the execution of another command.

4.4.3.4 Error

Error is TRUE if there is an error in the drive while the motion command is running, given by the Error signal in the Error word (register 35 Error bit 0).

4.5 JVL_MIS_Jog

4.5.1 Description

Run the motor at the specified velocity. The axis is stopped when the control inputs JogForwards and JogBackwards are reset.

4.5.2 Inputs/InOut

Input	Data Type	Units
JogForwards	Bool	
JogBackwards	Bool	
Velocity	LReal	RPM
Acceleration	LReal	RPM/s
Deceleration	LReal	RPM/s
Axis	JVL_MIS_AXIS	

4.5.2.1 JogForwards

When TRUE, the axis runs in the positive direction at the speed set in the Velocity input.

4.5.2.2 JogBackwards

When TRUE, the axis runs in the negative direction at the speed set in the Velocity input.

Note: If JogForwards and JogBackwards are both set TRUE, the axis will come to a standstill.

4.5.2.3 Velocity

The target velocity in RPM. Giving a negative value will NOT reverse the motor direction.

4.5.2.4 Acceleration

The target acceleration in RPM/s.

4.5.2.5 Deceleration

The target deceleration in RPM/s. If set to 0, the value in Acceleration is used instead.

4.5.2.6 Axis

The axis input should be linked to a data block of type JVL_MIS_Axis. This data block provides axis references to the unit and to its telegram slots.

4.5.3 Outputs

Output	Data Type
Busy	Bool
At Velocity	Bool
Command Aborted	Bool
Error	Bool

4.5.3.1 Busy

Busy is TRUE whenever the motion command is running (both when the motor is accelerating/decelerating and when 'At Velocity') and the Error output is FALSE.

4.5.3.2 At Velocity

At Velocity is TRUE when the drive has reached the motion command target velocity and the Error output is FALSE.

4.5.3.3 Command Aborted

The command was interrupted by the execution of another command or an error on the motor (e.g. position limit reached).

4.5.3.4 Error

Error is TRUE if there is an error in the drive while the motion command is running, given by the Error signal in the Error word (register 35 Error bit 0).

4.6 JVL_MIS_Position

4.6.1 Description

Move the axis to the specified position, or by the specified distance.

4.6.2 Inputs/InOut

Input	Data Type	Units
Execute	Bool	
ContinuousUpdate	Bool	
Position	LReal	Counts
Velocity	LReal	RPM
Acceleration	LReal	RPM/s
Deceleration	LReal	RPM/s
AbsolutePositioning	Bool	
Axis	JVL_MIS_AXIS	

4.6.2.1 Execute

A rising edge is required to start the motion command.

If the input is given a rising edge whilst a motion command is currently running, then the current motion command is overridden/aborted.

The input is not required to remain TRUE for the duration of the motion command.

Setting the value of Execute to FALSE during the motion command will not stop the motion command.

4.6.2.2 ContinuousUpdate

If ContinuousUpdate is TRUE when the Execute input is set then the Position and Velocity inputs can be changed while the motion command is active and the new values are applied immediately (with the defined acceleration and deceleration).

A change to the ContinuousUpdate input is not effective when the command is running, Execute must be re-triggered to change ContinuousUpdate.

4.6.2.3 Position

The axis target position, or distance to move, in encoder counts. See AbsolutePositioning input and ContinuousUpdate input.

4.6.2.4 Velocity

The target velocity for the positional move in RPM. See ContinuousUpdate input.

A negative velocity does not affect the direction of movement.

4.6.2.5 Acceleration

The target acceleration in RPM/s.

Acceleration is not affected by the ContinuousUpdate input, the value when execute was triggered is used. Execute must be re-triggered to change the acceleration.

4.6.2.6 Deceleration

The target deceleration in RPM/s. If set to 0, the value in Acceleration is used instead.

Deceleration is not affected by the ContinuousUpdate input, the value when execute was triggered is used. Execute must be re-triggered to change the deceleration.

4.6.2.7 AbsolutePositioning

Choose between absolute or relative positioning.

Value	Description
TRUE	Absolute Positioning Selected
FALSE	Relative Positioning Selected

4.6.2.8 Axis

The axis input should be linked to a data block of type JVL_MIS_Axis. This data block provides axis references to the unit and to its telegram slots.

4.6.3 Outputs

Output	Data Type
Busy	Bool
Done	Bool
Command Aborted	Bool
Error	Bool

4.6.3.1 Busy

Busy is TRUE whenever the motion command is running (both when the motor is accelerating and when 'At Velocity') and the Error output is FALSE.

4.6.3.2 Done

Done is TRUE when the positioning command has completed successfully.

4.6.3.3 Command Aborted

The command was interrupted by the execution of another command or an error on the motor (e.g. position limit reached).

4.6.3.4 Error

Error is TRUE if there is an error in the drive while the motion command is running, given by the Error signal in the Error word (register 35 Error bit 0).

4.7 JVL_MIS_Reset

4.7.1 Description

Resets any errors or faults in the drive.

4.7.2 Inputs/InOut

Input	Data Type
Execute	Bool
Axis	JVL_MIS_AXIS

4.7.2.1 Execute

A rising edge is required to start the command.

If the input is given a rising edge whilst a command is currently running, then the current command is overridden/aborted.

The input is not required to remain TRUE for the duration of the command.

Setting the value of Execute to FALSE during the command will not stop the command.

4.7.2.2 Axis

The axis input should be linked to a data block of type JVL_MIS_Axis. This data block provides axis references to the unit and to its telegram slots.

4.7.3 Outputs

Output	Data Type
Busy	Bool
Done	Bool
Command Aborted	Bool

4.7.3.1 Busy

Busy is TRUE whenever the command is in progress.

4.7.3.2 Done

Done is TRUE when the reset command has completed successfully.

4.7.3.3 Command Aborted

The command was interrupted by the execution of another command.

4.8 JVL_MIS_Velocity

4.8.1 Description

Run the motor at the specified velocity.

4.8.2 Inputs/InOut

Input	Data Type	Units
Execute	Bool	
ContinuousUpdate	Bool	
Velocity	LReal	RPM
Acceleration	LReal	RPM/s
Deceleration	LReal	RPM/s
Direction	Bool	
Axis	JVL_MIS_AXIS	

4.8.2.1 Execute

A rising edge is required to start the motion command.

If the input is given a rising edge whilst a motion command is currently running, then the current motion command is overridden/aborted.

The input is not required to remain TRUE for the duration of the motion command.

Setting the value of Execute to FALSE during the motion command will not stop the motion command.

4.8.2.2 ContinuousUpdate

If ContinuousUpdate is TRUE when the Execute input is set then the Velocity input can be changed while the motion command is active and the new value is applied immediately (with the defined acceleration and deceleration).

A change to the ContinuousUpdate input is not effective when the command is running, Execute must be re-triggered to change ContinuousUpdate.

4.8.2.3 Velocity

The target velocity in RPM.

Giving a negative value will NOT reverse the motor direction. The Direction input must be used to change the direction of motion.

4.8.2.4 Acceleration

The target acceleration in RPM/s.

Acceleration is not affected by the ContinuousUpdate input, the value when execute was triggered is used. Execute must be re-triggered to change the acceleration.

4.8.2.5 Deceleration

The target deceleration in RPM/s. If set to 0, the value in Acceleration is used instead.

Deceleration is not affected by the ContinuousUpdate input, the value when execute was triggered is used. Execute must be re-triggered to change the deceleration.

4.8.2.6 Direction

Set the direction of travel.

Value	Description
TRUE	Positive motor direction (clockwise)
FALSE	Negative motor direction (counter-clockwise)

4.8.2.7 Axis

The axis input should be linked to a data block of type JVL_MIS_Axis. This data block provides axis references to the unit and to its telegram slots.

4.8.3 Outputs

Output	Data Type
Busy	Bool
At Velocity	Bool
Command Aborted	Bool
Error	Bool

4.8.3.1 Busy

Busy is TRUE whenever the motion command is running (both when the motor is accelerating and when 'At Velocity') and the Error output is FALSE.

4.8.3.2 At Velocity

At Velocity is TRUE when the drive has reached the motion command target velocity and the Error output is FALSE.

4.8.3.3 Command Aborted

The command was interrupted by the execution of another command or an error on the motor (e.g. position limit reached).

4.8.3.4 Error

Error is TRUE if there is an error in the drive while the motion command is running, given by the Error signal in the Error word (register 35 Error bit 0).

5 MAC Function Blocks

5.1 JVL_MAC_Comms

5.1.1 Description

Performs cyclic Profinet communication. The cyclic telegram data is transferred between PLC input/output area and the axis data block.

This function is required for all other motion control function blocks to operate. A single instance of the JVL_MAC_Comms function must be called once per PLC cycle for every JVL motor in the system.

5.1.2 InOut

Input	Data Type
Axis	JVL_MAC_AXIS

5.1.2.1 Axis

The axis input should be linked to a data block of type JVL_MAC_Axis. This data block provides axis references to the unit and to its telegram slots.

5.1.3 Return Value

The function returns a TRUE value if the communication is operating without fault.

In the event of a fault (Ret_Val = FALSE) the return values of DPWR_DAT and DPRD_DAT inside this function must be evaluated to determine the cause.

5.2 JVL_MAC_Disable

5.2.1 Description

Place the drive in passive mode, removing all torque from the motor.

If the motor is running then it stops with the configured deceleration of the command that is in progress.

Caution, this can result in a long stopping distance if the configured deceleration is low. It is recommended that JVL_MAC_Halt is used with suitable deceleration before then executing JVL_MAC_Disable.

5.2.2 Inputs/InOut

Input	Data Type
Execute	Bool
Axis	JVL_MAC_AXIS

5.2.2.1 Execute

A rising edge is required to start the motion command. If the input is given a rising edge whilst a motion command is currently running, then the current motion command is overridden/aborted. The input is not required to remain TRUE for the duration of the motion command.

5.2.2.2 Axis

The axis input should be linked to a data block of type JVL_MAC_Axis. This data block provides axis references to the unit and to its telegram slots.

5.2.3 Outputs

Output	Data Type
Busy	Bool
Done	Bool
Command Aborted	Bool
Error	Bool

5.2.3.1 Busy

Busy is TRUE whenever the motion command is running and the Error output is FALSE.

5.2.3.2 Done

Done is TRUE when the motor has entered passive mode.

Note, if moving when the disable command was executed, Done = TRUE does not indicate that the motor has stopped. It is recommended that JVL_MAC_Halt is used with suitable deceleration before then executing JVL_MAC_Disable.

5.2.3.3 Command Aborted

The command was interrupted by the execution of another command.

5.2.3.4 Error

Error is TRUE if there is an error in the drive while the motion command is running, given by the ANY_ERR signal in the drive status word (DRIVE_DB.Telegram.Input.StatusWord, bit 24 of drive register 35 ERR_STAT).

For more information on the error that occurred, see the value of DRIVE_DB.Telegram.Input.StatusWord (drive register 35 ERR_STAT).

5.3 JVL_MAC_Halt

5.3.1 Description

Brings the motor to a standstill with defined deceleration.

5.3.2 Inputs/InOut

Input	Data Type	Units
Execute	Bool	
Deceleration	LReal	RPM/s
Axis	JVL_MAC_AXIS	

5.3.2.1 Execute

A rising edge is required to start the motion command.

If the input is given a rising edge whilst a motion command is currently running, then the current motion command is overridden/aborted.

The input is not required to remain TRUE for the duration of the motion command.

Setting the value of Execute to FALSE during the motion command will not stop the motion command.

5.3.2.2 Deceleration

The target deceleration in RPM/s.

Note: Due to the resolution of the acceleration/deceleration setpoint in the telegram (register 6 A_SOLL), a value of **at least** 150 RPM/s should be given, or the motion command will not execute.

5.3.2.3 Axis

The axis input should be linked to a data block of type JVL_MAC_Axis. This data block provides axis references to the unit and to its telegram slots.

5.3.3 Outputs

Output	Data Type
Busy	Bool
Done	Bool
Command Aborted	Bool
Error	Bool

5.3.3.1 Busy

Busy is TRUE whenever the motion command is running (both when the motor is accelerating and when 'At Velocity') and the Error output is FALSE.

5.3.3.2 Done

Done is TRUE when the motor has come to a standstill.

5.3.3.3 Command Aborted

The command was interrupted by the execution of another command.

5.3.3.4 Error

Error is TRUE if there is an error in the drive while the motion command is running, given by the ANY_ERR signal in the drive status word (DRIVE_DB.Telegram.Input.StatusWord, bit 24 of drive register 35 ERR_STAT).

For more information on the error that occurred, see the value of DRIVE_DB.Telegram.Input.StatusWord (drive register 35 ERR_STAT).

5.4 JVL_MAC_Home

5.4.1 Description

Homes or references the motor position.

The correct function block must be used for the type of MAC motor.

5.4.2 Inputs/InOut

Input	Data Type	Units
Execute	Bool	
Mode	Int	
Position	Dint	Counts
Velocity	LReal	RPM
Acceleration	LReal	RPM/s
TorqueLimit	Real	%
Offset	Dint	Counts
Axis	JVL_MAC_AXIS	

5.4.2.1 Execute

A rising edge is required to start the motion command.

If the input is given a rising edge whilst a motion command is currently running, then the current motion command is overridden/aborted.

The input is not required to remain TRUE for the duration of the motion command.

Setting the value of Execute to FALSE during the motion command will not stop the motion command.

5.4.2.2 Mode

Set the homing method.

Value	Name
0	Set Home Position
1	Home to Sensor
2	Home to Endstop
3	Set Absolute Encoder

0 - Set Home Position

The current motor actual position is set to the value of the Position input. The motor does not move. If the motor is running then it is stopped.

1 - Home to Sensor

The motor runs until it reaches the home sensor and then stops. The position of the motor is set relative to the point at which the home sensor was triggered. The exact homing operation is configured in MacTalk.

Note: Please make sure to use Input AIN on the MAC device as the homing sensor. See MAC operating instructions for details of input wiring.

2 - Home to Endstop

The motor runs until the torque limit is reached. The position of the motor is set relative to the point at which the torque limit was reached.

3 - Set Absolute Encoder

The current motor Absolute Encoder position is set to the value of the Position input. The motor does not move. If the motor is running then it is stopped. Only available for motors with Absolute Encoder. The range is $\pm 2^{24}$ (different to actual position: $\pm 2^{31}$)

5.4.2.3 Position

The new axis position, after referencing, in encoder counts. Only effective in homing mode 0 and 3.

5.4.2.4 Velocity

The target velocity for the homing move in RPM.

Only effective in homing modes 1 and 2.

5.4.2.5 Acceleration

The target acceleration and deceleration in RPM/s.

Only effective in homing modes 1 and 2.

Note: Due to the resolution of the acceleration/deceleration setpoint in the telegram (register 6 A_SOLL), a value of **at least** 150 RPM/s should be given, or the motion command will not execute.

5.4.2.6 TorqueLimit

The torque threshold for the motor to stop and home position to be set in %.

Valid range 0 – 100%. Only effective in homing mode 2.

5.4.2.7 Offset

Relative distance for offset move following home to sensor or home to torque limit in counts. The operation of Offset depends on the homing configuration in MacTalk.

Only effective in homing modes 1 and 2.

5.4.2.8 Axis

The axis input should be linked to a data block of type JVL_MAC_Axis. This data block provides axis references to the unit and to its telegram slots.

5.4.3 Outputs

Output	Data Type
Busy	Bool
Done	Bool
Command Aborted	Bool
Error	Bool

5.4.3.1 Busy

Busy is TRUE whenever the motion command is running and the Error output is FALSE.

5.4.3.2 Done

Done is TRUE when the homing sequence has completed and the axis has been successfully referenced.

5.4.3.3 Command Aborted

The command was interrupted by the execution of another command.

5.4.3.4 Error

Error is TRUE if there is an error in the drive while the motion command is running, given by the ANY_ERR signal in the drive status word (DRIVE_DB.Telegram.Input.StatusWord, bit 24 of drive register 35 ERR_STAT).

For more information on the error that occurred, see the value of DRIVE_DB.Telegram.Input.StatusWord (drive register 35 ERR_STAT).

5.5 JVL_MAC_Jog

5.5.1 Description

Run the motor at the specified velocity. The axis is stopped when the control inputs JogForwards and JogBackwards are reset.

5.5.2 Inputs/InOut

Input	Data Type	Units
JogForwards	Bool	
JogBackwards	Bool	
Velocity	LReal	RPM
Acceleration	LReal	RPM/s
Axis	JVL_MAC_AXIS	

5.5.2.1 JogForwards

When TRUE, the axis runs in the positive direction at the speed set in the Velocity input.

5.5.2.2 JogBackwards

When TRUE, the axis runs in the negative direction at the speed set in the Velocity input.

Note: If JogForwards and JogBackwards are both set TRUE, the axis will come to a standstill.

5.5.2.3 Velocity

The target velocity in RPM. Giving a negative value will NOT reverse the motor direction.

5.5.2.4 Acceleration

The target acceleration and deceleration in RPM/s.

Note: Due to the resolution of the acceleration/deceleration setpoint in the telegram (register 6 A_SOLL), a value of **at least** 150 RPM/s should be given, or the motion command will not execute.

5.5.2.5 Axis

The axis input should be linked to a data block of type JVL_MAC_Axis. This data block provides axis references to the unit and to its telegram slots.

5.5.3 Outputs

Output	Data Type
Busy	Bool
AtVelocity	Bool
Command Aborted	Bool
Error	Bool

5.5.3.1 Busy

Busy is TRUE whenever the motion command is running (both when the motor is accelerating/decelerating and when 'AtVelocity') and the Error output is FALSE.

5.5.3.2 AtVelocity

At Velocity is TRUE when the drive has reached the motion command target velocity and the Error output is FALSE.

5.5.3.3 Command Aborted

The command was interrupted by the execution of another command or an error on the motor (e.g. position limit reached).

5.5.3.4 Error

Error is TRUE if there is an error in the drive while the motion command is running, given by the ANY_ERR signal in the drive status word (DRIVE_DB.Telegram.Input.StatusWord, bit 24 of drive register 35 ERR_STAT).

For more information on the error that occurred, see the value of DRIVE_DB.Telegram.Input.StatusWord (drive register 35 ERR_STAT).

5.6 JVL_MAC_Position

5.6.1 Description

Move the axis to the specified position, or by the specified distance.

5.6.2 Inputs/InOut

Input	Data Type	Units
Execute	Bool	
ContinuousUpdate	Bool	
Position	Dint	Counts
Velocity	LReal	RPM
Acceleration	LReal	RPM/s
AbsolutePositioning	Bool	
Axis	JVL_MAC_AXIS	

5.6.2.1 Execute

A rising edge is required to start the motion command.

If the input is given a rising edge whilst a motion command is currently running, then the current motion command is overridden/aborted.

The input is not required to remain TRUE for the duration of the motion command.

Setting the value of Execute to FALSE during the motion command will not stop the motion command.

5.6.2.2 ContinuousUpdate

If ContinuousUpdate is TRUE when the Execute input is set then the Position and Velocity inputs can be changed while the motion command is active and the new values are applied immediately (with the defined acceleration and deceleration).

A change to the ContinuousUpdate input is not effective when the command is running, Execute must be re-triggered to change ContinuousUpdate.

5.6.2.3 Position

The axis target position, or distance to move, in encoder counts. See AbsolutePositioning input and ContinuousUpdate input.

5.6.2.4 Velocity

The target velocity for the positional move in RPM. See ContinuousUpdate input.

A negative velocity does not affect the direction of movement.

5.6.2.5 Acceleration

The target acceleration and deceleration in RPM/s.

Note: Due to the resolution of the acceleration/deceleration setpoint in the telegram (register 6 A_SOLL), a value of **at least** 150 RPM/s should be given, or the motion command will not execute.

5.6.2.6 AbsolutePositioning

Choose between absolute or relative positioning.

Value	Description
TRUE	Absolute Positioning Selected
FALSE	Relative Positioning Selected

5.6.2.7 Axis

The axis input should be linked to a data block of type JVL_MAC_Axis. This data block provides axis references to the unit and to its telegram slots.

5.6.3 Outputs

Output	Data Type
Busy	Bool
Done	Bool
Command Aborted	Bool
Error	Bool

5.6.3.1 Busy

Busy is TRUE whenever the motion command is running (both when the motor is accelerating and when 'At Velocity') and the Error output is FALSE.

5.6.3.2 Done

Done is TRUE when the positioning command has completed successfully.

5.6.3.3 Command Aborted

The command was interrupted by the execution of another command or an error on the motor (e.g. position limit reached).

5.6.3.4 Error

Error is TRUE if there is an error in the drive while the motion command is running, given by the ANY_ERR signal in the drive status word (DRIVE_DB.Telegram.Input.StatusWord, bit 24 of drive register 35 ERR_STAT).

For more information on the error that occurred, see the value of DRIVE_DB.Telegram.Input.StatusWord (drive register 35 ERR_STAT).

5.7 JVL_MAC_Reset

5.7.1 Description

Resets any errors or faults in the drive.

5.7.2 Inputs/InOut

Input	Data Type
Execute	Bool
Axis	JVL_MAC_AXIS

5.7.2.1 Execute

A rising edge is required to start the command.

If the input is given a rising edge whilst a command is currently running, then the current command is overridden/aborted.

The input is not required to remain TRUE for the duration of the command.

Setting the value of Execute to FALSE during the command will not stop the command.

5.7.2.2 Axis

The axis input should be linked to a data block of type JVL_MAC_Axis. This data block provides axis references to the unit and to its telegram slots.

5.7.3 Outputs

Output	Data Type
Busy	Bool
Done	Bool
Command Aborted	Bool

5.7.3.1 Busy

Busy is TRUE whenever the command is in progress.

5.7.3.2 Done

Done is TRUE when the reset command has completed successfully.

5.7.3.3 Command Aborted

The command was interrupted by the execution of another command.

5.8 JVL_MAC_Velocity

5.8.1 Description

Run the motor at the specified velocity.

5.8.2 Inputs/InOut

Input	Data Type	Units
Execute	Bool	
ContinuousUpdate	Bool	
Velocity	LReal	RPM
Acceleration	LReal	RPM/s
Direction	Bool	
Axis	JVL_MAC_AXIS	

5.8.2.1 Execute

A rising edge is required to start the motion command.

If the input is given a rising edge whilst a motion command is currently running, then the current motion command is overridden/aborted.

The input is not required to remain TRUE for the duration of the motion command.

Setting the value of Execute to FALSE during the motion command will not stop the motion command.

5.8.2.2 ContinuousUpdate

If ContinuousUpdate is TRUE when the Execute input is set then the Velocity input can be changed while the motion command is active and the new value is applied immediately (with the defined acceleration and deceleration).

A change to the ContinuousUpdate input is not effective when the command is running, Execute must be re-triggered to change ContinuousUpdate.

5.8.2.3 Velocity

The target velocity in RPM.

Giving a negative value will NOT reverse the motor direction. The Direction input must be used to change the direction of motion.

5.8.2.4 Acceleration

The target acceleration and deceleration in RPM/s.

Note: Due to the resolution of the acceleration/deceleration setpoint in the telegram (register 6 A_SOLL), a value of **at least** 150 RPM/s should be given, or the motion command will not execute.

5.8.2.5 Direction

Set the direction of travel.

Value	Description
TRUE	Positive motor direction (clockwise)
FALSE	Negative motor direction (counter-clockwise)

5.8.2.6 Axis

The axis input should be linked to a data block of type JVL_MAC_Axis. This data block provides axis references to the unit and to its telegram slots.

5.8.3 Outputs

Output	Data Type
Busy	Bool
AtVelocity	Bool
Command Aborted	Bool
Error	Bool

5.8.3.1 Busy

Busy is TRUE whenever the motion command is running (both when the motor is accelerating and when 'At Velocity') and the Error output is FALSE.

5.8.3.2 AtVelocity

At Velocity is TRUE when the drive has reached the motion command target velocity and the Error output is FALSE.

5.8.3.3 Command Aborted

The command was interrupted by the execution of another command or an error on the motor (e.g. position limit reached).

5.8.3.4 Error

Error is TRUE if there is an error in the drive while the motion command is running, given by the ANY_ERR signal in the drive status word (DRIVE_DB.Telegram.Input.StatusWord, bit 24 of drive register 35 ERR_STAT).

For more information on the error that occurred, see the value of DRIVE_DB.Telegram.Input.StatusWord (drive register 35 ERR_STAT).

5.8.4 Example Function Diagram

