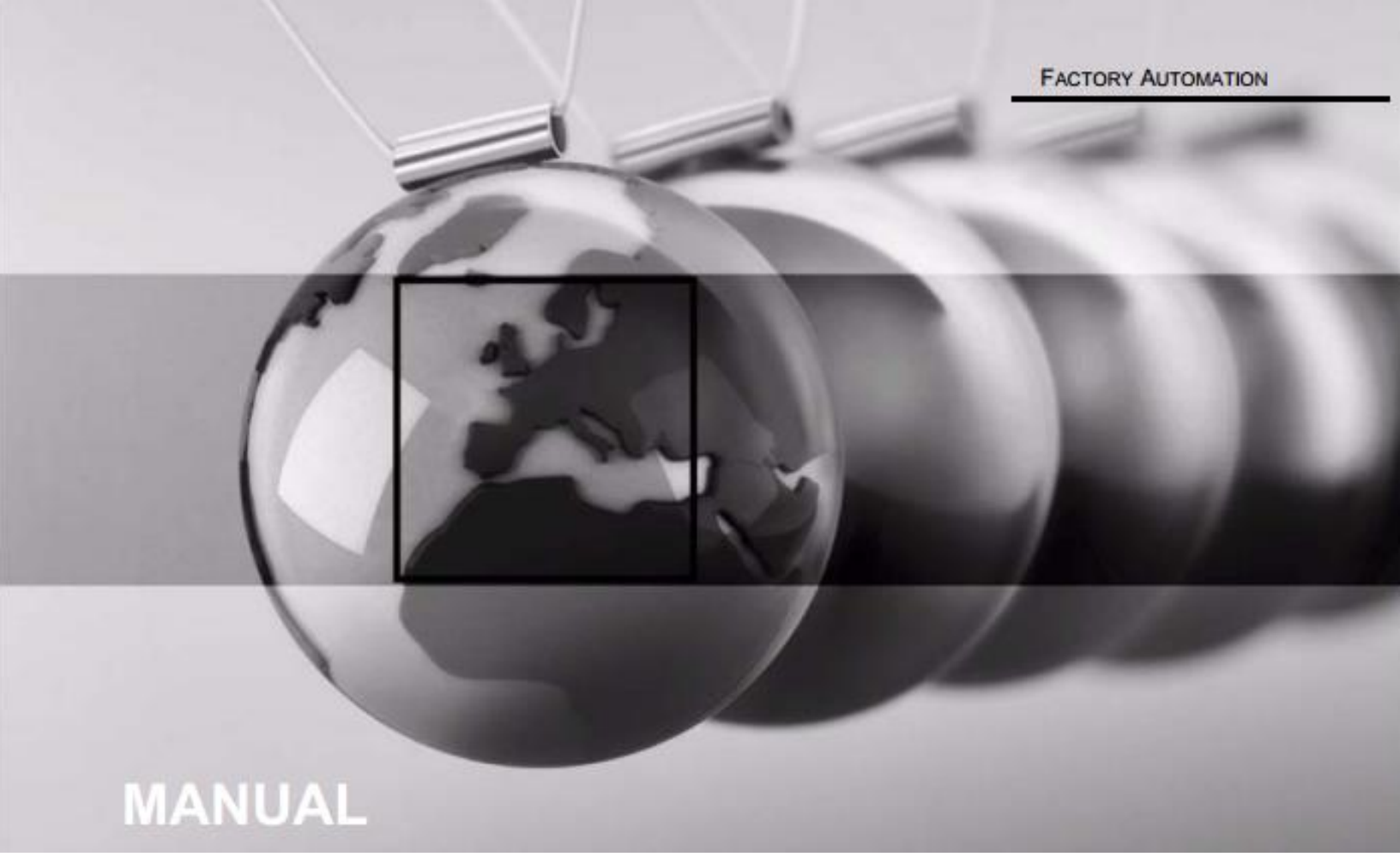
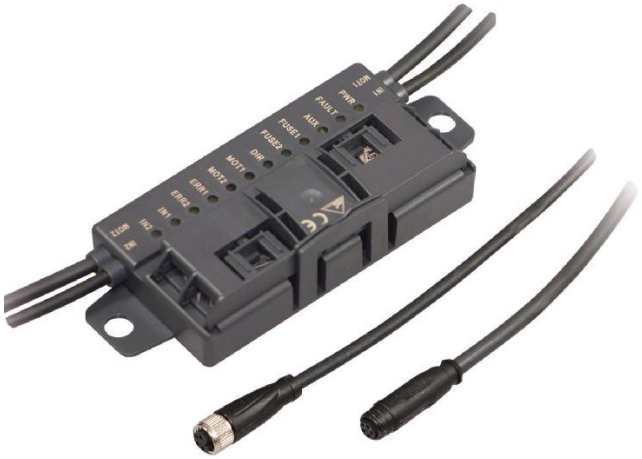


Add-On Instruction Description for G20 Modules



Contents

Introduction

This add-on instruction is used to facilitate motor ramp, speed, and diagnostic retrieval setting. Use this add-on instruction to set the speed or ramp of a single AS-Interface module. The address and network are defined by the add-on instruction parameters.

Software Compatibility

The add-on is compatible with Studio 5000 Version 24 or higher.

Hardware Compatibility

The add-on must be used with two network EtherNet/IP AS-Interface gateways starting with VBG-ENX-Kxx-Dxx-... The ramp and speed can only be set on special motor modules VBA-4ExA-G20-ZEL/MxL-Px. Setting the ramp or speed on non-motor modules can lead to undesirable results.

VBA-4E3A-G20-ZEL/M1L-P2	MDR controller module for Interroll EC310, Rulmeca BL3, and Itoh Denki PM500XK
VBA-4E3A-G20-ZEL/M1L-P6	MDR controller module for Interroll EC310, Rulmeca, BL3, and Itoh Denki PM500XK
VBA-4E4A-G20-ZEJ/M3L-P9	MDR controller module for Interroll EC310, Rulmeca BL3, and Itoh Denki PM500XK
VBA-4E4A-G20-ZEJ/M3L-P10	MDR controller module for Itoh Denki PM500XE, PM500XP

Connection Compatibility

Set the IO connection path to "I/O: C1/2 A/B Slaves + CI" on dual network AS-Interface gateways. The length of this assembly is 51 INTs in and 51 INTs out. The size must be INT and has to be changed manually. See Figure 1. Other assemblies can also be used, but the CI (command interface) is mandatory. In cases where a different assembly is used, the data must first be copied into the array variables to mimic this assembly. See Figure 2.

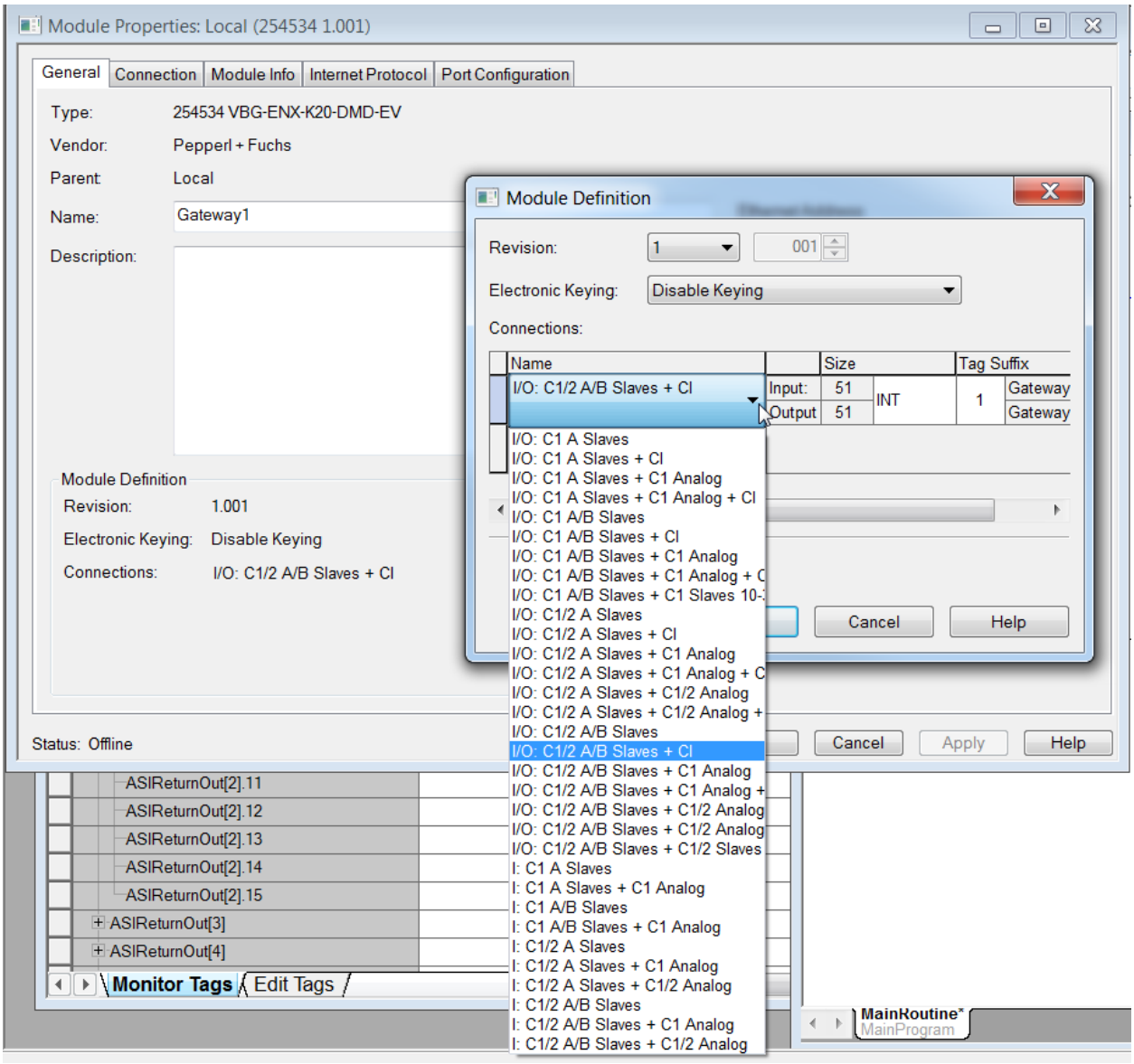


Figure 1: Required connection parameters

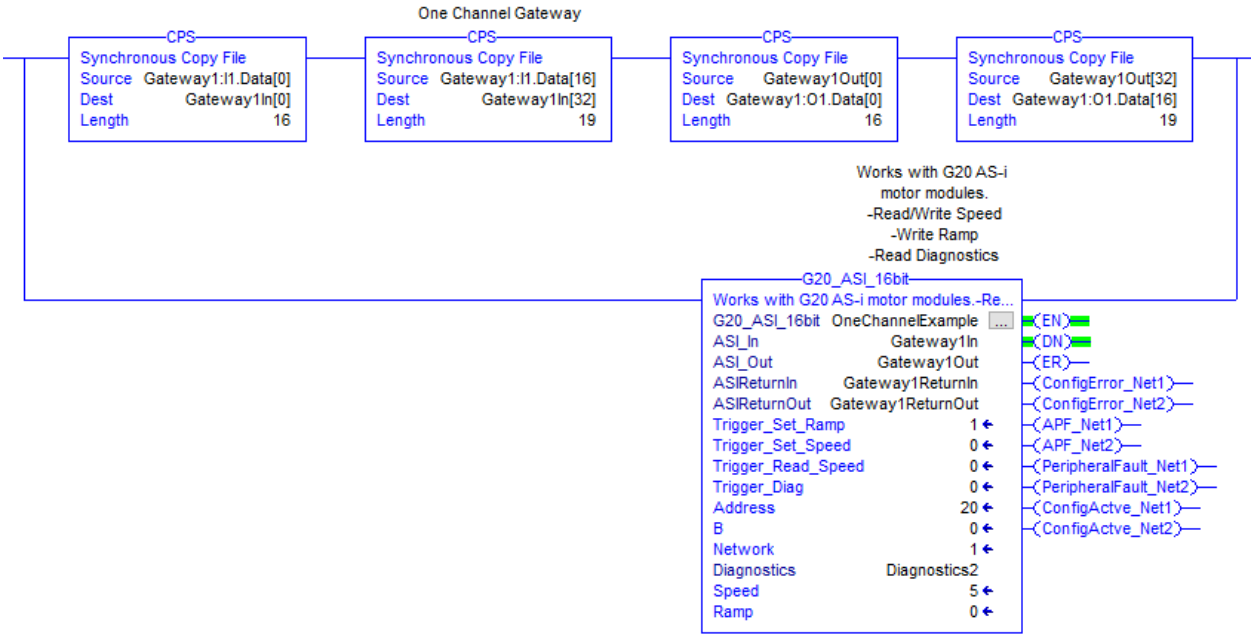


Figure 2: Add-on used with one channel gateway

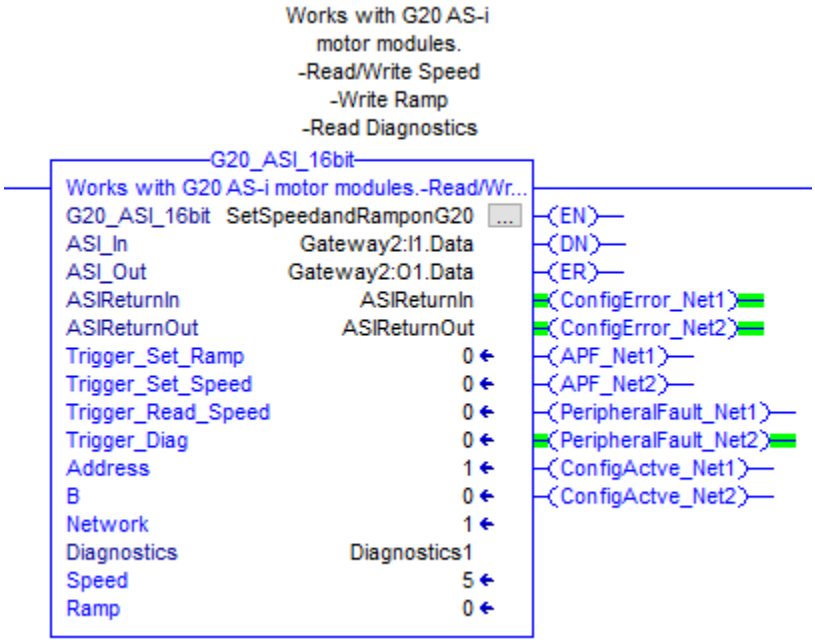


Figure 3: AS-Interface add-on

Array Index	Data (INT)
0	Input data Network 1
…	
15	
16	Input data Network 2
…	
31	
32	CI (Command Interface)
…	
51	

Array Index	Data (INT)
0	Output data Network 1
…	
15	
16	Output data Network 2
…	
31	
32	CI (Command Interface)
…	
51	

Figure 4: Required AS-I_In and AS-I _Out mapping

IO Mapping

Inputs are automatically copied into ASIReturnIn. Use these inputs instead of the inputs directly from ASI_In; however, ASI_In can be used if desired.

Outputs can only be used from the ASIReturnOut variable, and all outputs must be referenced to the ASIReturnOut variable. If Gateway2:O1 is used, these outputs are overwritten by ASIReturnOut automatically. See Figure 5.

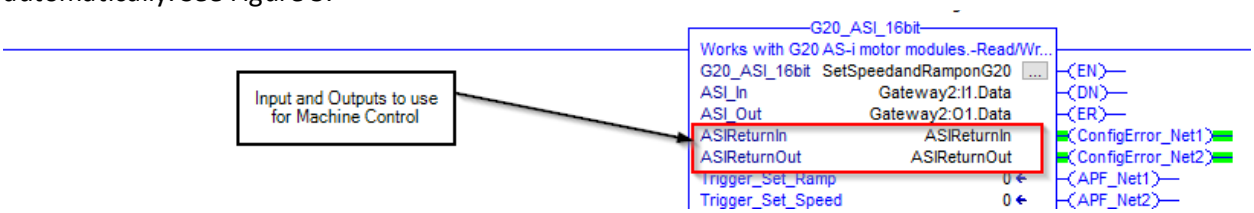


Figure 5: Inputs and outputs to use for PLC control

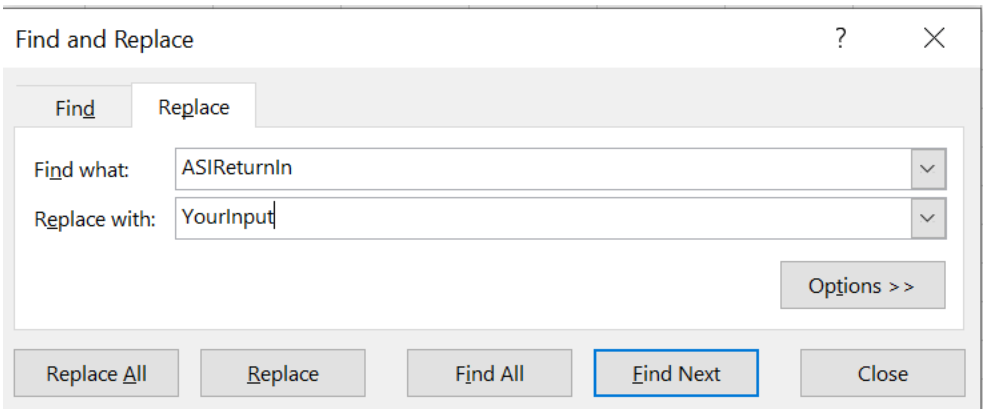
The IO is mapped into the two separate input and output files, ASIReturnIn and ASIReturnOut, as a large block of data. Four inputs and four outputs are mapped for each node, even if the node is not being used. Check Figure 6 for input and output location mapping.

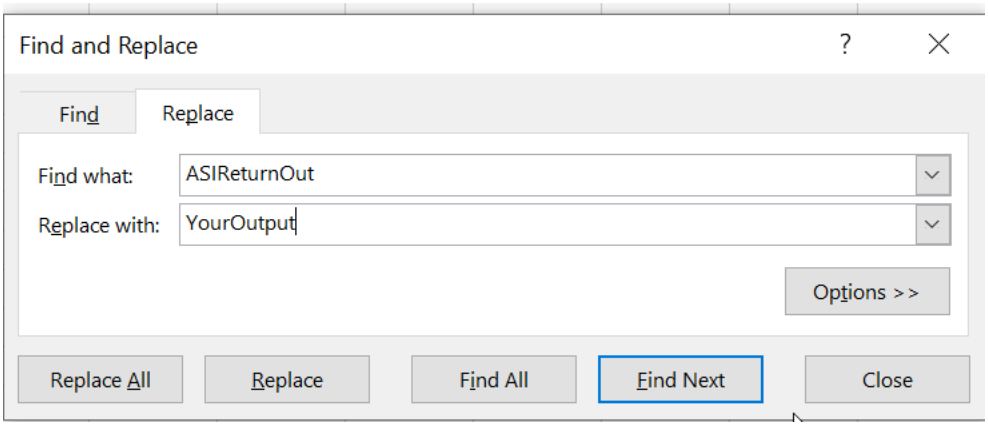
	Bits																
Word	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0	Net 1 Adr 2				Net 1 Adr 3				Flags				Net 1 Adr 1				Network 1, 1-31A
1	Net 1 Adr 6				Net 1 Adr 7				Net 1 Adr 4				Net 1 Adr 5				
2	Net 1 Adr 10				Net 1 Adr 11				Net 1 Adr 8				Net 1 Adr 9				
3	Net 1 Adr 14				Net 1 Adr 15				Net 1 Adr 12				Net 1 Adr 13				
4	Net 1 Adr 18				Net 1 Adr 19				Net 1 Adr 16				Net 1 Adr 17				
5	Net 1 Adr 22				Net 1 Adr 23				Net 1 Adr 20				Net 1 Adr 21				
6	Net 1 Adr 26				Net 1 Adr 27				Net 1 Adr 24				Net 1 Adr 25				
7	Net 1 Adr 30				Net 1 Adr 31				Net 1 Adr 28				Net 1 Adr 29				
8	Net 1 Adr 2b				Net 1 Adr 3b								Net 1 Adr 1b				Network 1, 1-31B
9	Net 1 Adr 6b				Net 1 Adr 7b				Net 1 Adr 4b				Net 1 Adr 5b				
10	Net 1 Adr 10b				Net 1 Adr 11b				Net 1 Adr 8b				Net 1 Adr 9b				
11	Net 1 Adr 14b				Net 1 Adr 15b				Net 1 Adr 12b				Net 1 Adr 13b				
12	Net 1 Adr 18b				Net 1 Adr 19b				Net 1 Adr 16b				Net 1 Adr 17b				
13	Net 1 Adr 22b				Net 1 Adr 23b				Net 1 Adr 20b				Net 1 Adr 21b				
14	Net 1 Adr 26b				Net 1 Adr 27b				Net 1 Adr 24b				Net 1 Adr 25b				
15	Net 1 Adr 30b				Net 1 Adr 31b				Net 1 Adr 28b				Net 1 Adr 29b				
16	Net 2 Adr 2				Net 2 Adr 3				Flags				Net 2 Adr 1				Network 2, 1-31A
17	Net 2 Adr 6				Net 2 Adr 7				Net 2 Adr 4				Net 2 Adr 5				
18	Net 2 Adr 10				Net 2 Adr 11				Net 2 Adr 8				Net 2 Adr 9				
19	Net 2 Adr 14				Net 2 Adr 15				Net 2 Adr 12				Net 2 Adr 13				
20	Net 2 Adr 18				Net 2 Adr 19				Net 2 Adr 16				Net 2 Adr 17				
21	Net 2 Adr 22				Net 2 Adr 23				Net 2 Adr 20				Net 2 Adr 21				
22	Net 2 Adr 26				Net 2 Adr 27				Net 2 Adr 24				Net 2 Adr 25				
23	Net 2 Adr 30				Net 2 Adr 31				Net 2 Adr 28				Net 2 Adr 29				
24	Net 2 Adr 2b				Net 2 Adr 3b								Net 2 Adr 1b				Network 2, 1-31B
25	Net 2 Adr 6b				Net 2 Adr 7b				Net 2 Adr 4b				Net 2 Adr 5b				
26	Net 2 Adr 10b				Net 2 Adr 11b				Net 2 Adr 8b				Net 2 Adr 9b				
27	Net 2 Adr 14b				Net 2 Adr 15b				Net 2 Adr 12b				Net 2 Adr 13b				
28	Net 2 Adr 18b				Net 2 Adr 19b				Net 2 Adr 16b				Net 2 Adr 17b				
29	Net 2 Adr 22b				Net 2 Adr 23b				Net 2 Adr 20b				Net 2 Adr 21b				
30	Net 2 Adr 26b				Net 2 Adr 27b				Net 2 Adr 24b				Net 2 Adr 25b				
31	Net 2 Adr 30b				Net 2 Adr 31b				Net 2 Adr 28b				Net 2 Adr 29b				

Figure 6: Input and output table layout

Description Files

Two description files have been created. One describes the inputs and the other describes the outputs. By default, the names are ASIReturnIn and ASIReturnOut. If your variable names are different, simply open the CSV files for each and do a "find" and "replace." See Figure 5. Also edit the Excel file with the name of your ASIReturnOut variable. See Figure 6. After the Excel files have been corrected, import them into your PLC. See Figure 7. The tags will end up with the descriptions of all inputs and outputs on AS-Interface. Depending on if the variables are in controller tags or local tags, you may also have to change the scope. See Figures 8 and 9.





6	0.3						
7	TYPE	SCOPE	NAME	DESCRIPTI	DATATYPE	SPECIFIER	ATTRIBUTES
8	COMMEN	MainProgram	ASIReturnIn	Inputs for A Nodes S		ASIReturnIn[0]	
9	COMMEN	MainProgram	ASIReturnIn	Node 1(A)$NInput D		ASIReturnIn[0].0	
10	COMMEN	MainProgram	ASIReturnIn	Node 1(A)$NInput D		ASIReturnIn[0].1	
11	COMMEN	MainProgram	ASIReturnIn	Node 1(A)$NInput D		ASIReturnIn[0].2	
12	COMMEN	MainProgram	ASIReturnIn	Node 1(A)$NInput D		ASIReturnIn[0].3	
13	COMMEN	MainProgram	ASIReturnIn	F0$N0=Okay, 1=Conf		ASIReturnIn[0].4	
14	COMMEN	MainProgram	ASIReturnIn	F1$N0=AS-I power O		ASIReturnIn[0].5	

Figure 7: Replacing ASIReturnOut and ASIReturnIn with variable names in description files. Changing the scope for controller tags or local variables

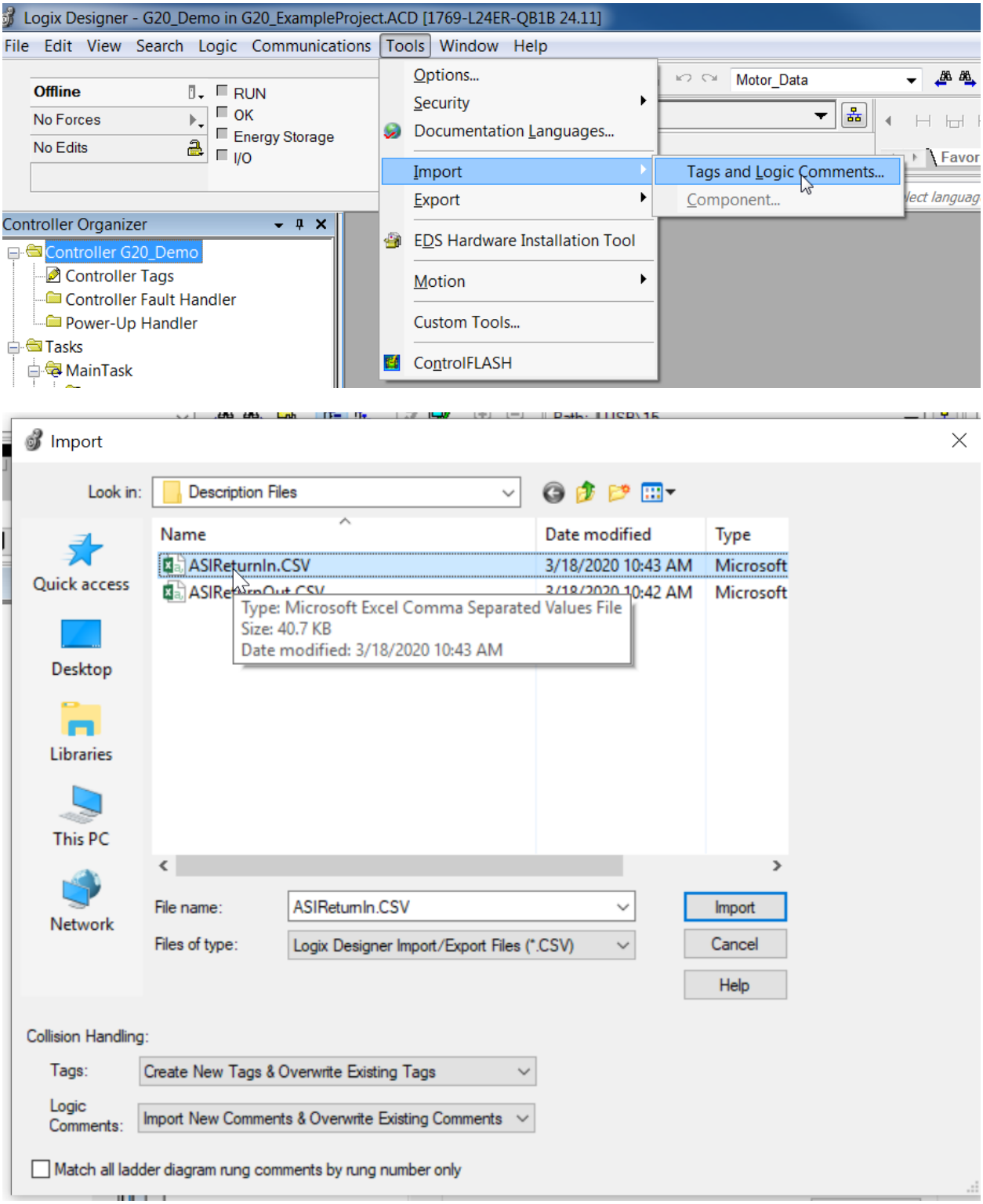


Figure 8: Import description files into PLC

Controller Tags - G20_Demo(controller)

Scope: G20_Demo Show: All Tags Enter Name Filter...

Name	Value	Style	Data Type	Description
Gateway1:I1.Data	{...}	Decimal	INT[51]	
Gateway1:I1.Data[0]	13059	Decimal	INT	Inputs for A Nodes Start
Gateway1:I1.Data[0].0	1	Decimal	BOOL	Node 1(A) Input D0
Gateway1:I1.Data[0].1	1	Decimal	BOOL	Node 1(A) Input D1
Gateway1:I1.Data[0].2	0	Decimal	BOOL	Node 1(A) Input D2
Gateway1:I1.Data[0].3	0	Decimal	BOOL	Node 1(A) Input D3
Gateway1:I1.Data[0].4	0	Decimal	BOOL	F0 0=Okay, 1=Config Error
Gateway1:I1.Data[0].5	0	Decimal	BOOL	F1 0=AS-I power Okay, 1=AS-i Power failure
Gateway1:I1.Data[0].6	0	Decimal	BOOL	F2 0=Okay, 1=Peripheral fault
Gateway1:I1.Data[0].7	0	Decimal	BOOL	F3 0=Protected mode active, 1=Configuration mo...
Gateway1:I1.Data[0].8	1	Decimal	BOOL	Node 3(A) Input D0
Gateway1:I1.Data[0].9	1	Decimal	BOOL	Node 3(A) Input D1
Gateway1:I1.Data[0].10	0	Decimal	BOOL	Node 3(A) Input D2
Gateway1:I1.Data[0].11	0	Decimal	BOOL	Node 3(A) Input D3
Gateway1:I1.Data[0].12	1	Decimal	BOOL	Node 2(A) Input D0
Gateway1:I1.Data[0].13	1	Decimal	BOOL	Node 2(A) Input D1
Gateway1:I1.Data[0].14	0	Decimal	BOOL	Node 2(A) Input D2
Gateway1:I1.Data[0].15	0	Decimal	BOOL	Node 2(A) Input D3
Gateway1:I1.Data[1]	13107	Decimal	INT	
Gateway1:I1.Data[2]	13107	Decimal	INT	
Gateway1:I1.Data[3]	13107	Decimal	INT	

Controller Tags - G20_Demo(controller)

Scope: G20_Demo Show: All Tags Enter Name Filter...

Name	Value	Style	Data Type	Description	Constan
ASIReturnOut	{...}	Decimal	INT[32]		
ASIReturnOut[0]	0	Decimal	INT	Outputs for A Nodes Start	
ASIReturnOut[0].0	0	Decimal	BOOL	Node 1(A) Output D0	
ASIReturnOut[0].1	0	Decimal	BOOL	Node 1(A) Output D1	
ASIReturnOut[0].2	0	Decimal	BOOL	Node 1(A) Output D2	
ASIReturnOut[0].3	0	Decimal	BOOL	Node 1(A) Output D3	
ASIReturnOut[0].4	0	Decimal	BOOL	F0 0=On-line and 1=Offline	
ASIReturnOut[0].5	0	Decimal	BOOL	F1 Set if you want to take Network Off-line on confi...	
ASIReturnOut[0].6	0	Decimal	BOOL	Rising edge of this bit will take the AS-i master int...	
ASIReturnOut[0].7	0	Decimal	BOOL	Rising edge of this bit will take the network into pr...	
ASIReturnOut[0].8	0	Decimal	BOOL	Node 3(A) Output D0	
ASIReturnOut[0].9	0	Decimal	BOOL	Node 3(A) Output D1	
ASIReturnOut[0].10	0	Decimal	BOOL	Node 3(A) Output D2	
ASIReturnOut[0].11	0	Decimal	BOOL	Node 3(A) Output D3	
ASIReturnOut[0].12	0	Decimal	BOOL	Node 2(A) Output D0	
ASIReturnOut[0].13	0	Decimal	BOOL	Node 2(A) Output D1	
ASIReturnOut[0].14	0	Decimal	BOOL	Node 2(A) Output D2	
ASIReturnOut[0].15	0	Decimal	BOOL	Node 2(A) Output D3	
ASIReturnOut[1]	0	Decimal	INT		
ASIReturnOut[2]	0	Decimal	INT		
ASIReturnOut[3]	0	Decimal	INT		
ASIReturnOut[4]	0	Decimal	INT		
ASIReturnOut[5]	0	Decimal	INT		
ASIReturnOut[6]	0	Decimal	INT		

Figure 9: Inputs and outputs described after tag import

Ramp and Speed Values

There are eight possible ramp setting values numbered 0 – 7 and eight speed settings numbered 0 – 7. The default ramp is zero and the default speed is seven. Change as required by the application.

Ramp	Ramp time (Stop→Vmax, Vmax→Stop)
0	No slope (default setting)
1	50 ms
2	100 ms
3	200 ms
4	300 ms
5	500 ms
6	1000 ms
7	1500 ms

Figure 10: Possible ramp settings

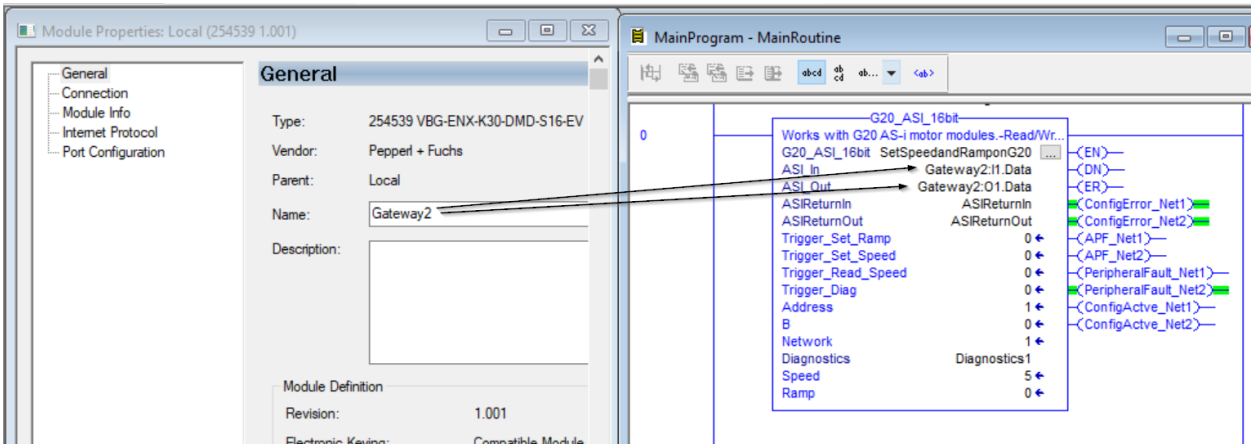
Speed	Speed control signal P2/P6
off	< 1.5 V
0	3.96 V (3.92 ... 4.00 V)
1	4.78 V (4.73 ... 4.83 V)
2	5.61 V (5.55 ... 5.67 V)
3	6.44 V (6.38 ... 6.50 V)
4	8.50 V (8.42 ... 8.59 V)
5	9.63 V (9.53 ... 9.73 V)
6	10.00 V (9.90 ... 10.10V)
7	**7.26 V (7.19 ... 7.33 V)**

Speed	Speed control signal P9/P10
off	< 1 V
0	4.7 V fast, 0.7 V slow, The two motors spin opposite directions
1	5.7 V fast, 1.7 V slow, The two motors spin opposite directions
2	7.7 V fast, 2.7 V slow, The two motors spin opposite directions
3	9.7 V fast, 3.7 V slow, The two motors spin opposite directions
4	4.7 V fast, 0.7 V slow, The two motors spin in same direction
5	5.7 V fast, 1.7 V slow, The two motors spin in same direction
6	7.7 V fast, 2.7 V slow, The two motors spin in same direction
7	**9.7 V fast, 3.7 V slow, The two motors spin in same direction**

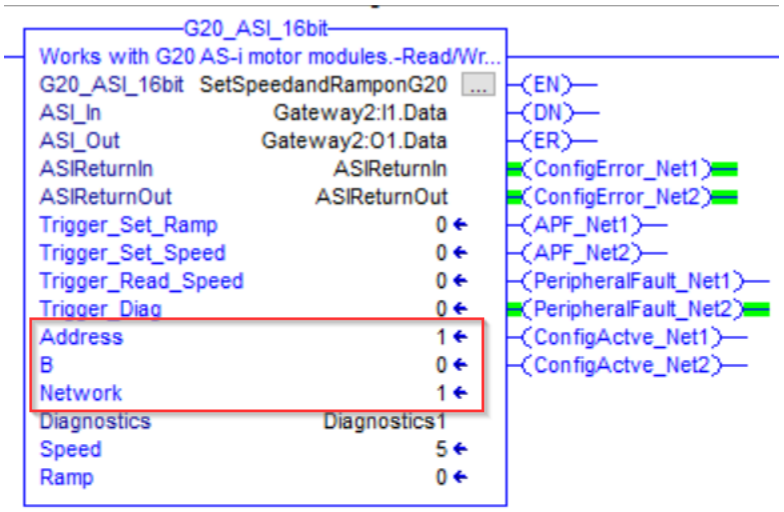
Figure 11: Speed settings depending on the motor module type

Configuring the Add-on

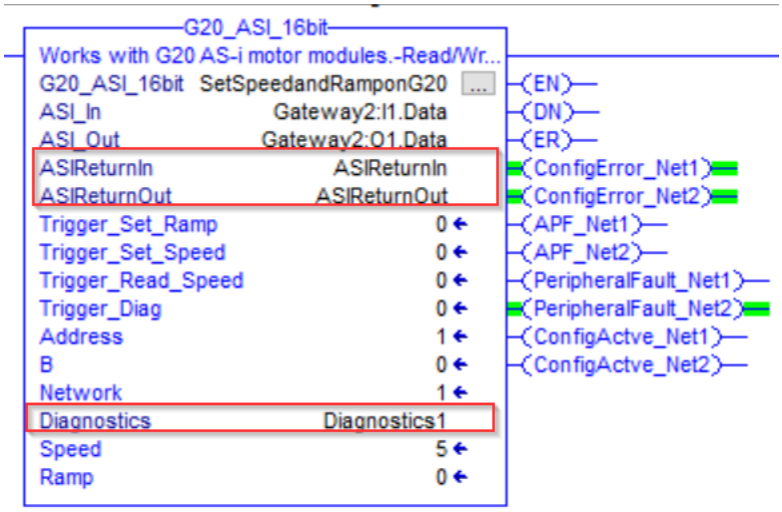
Set ASI_In and ASI_Out to the input and output data from the AS-Interface gateway.



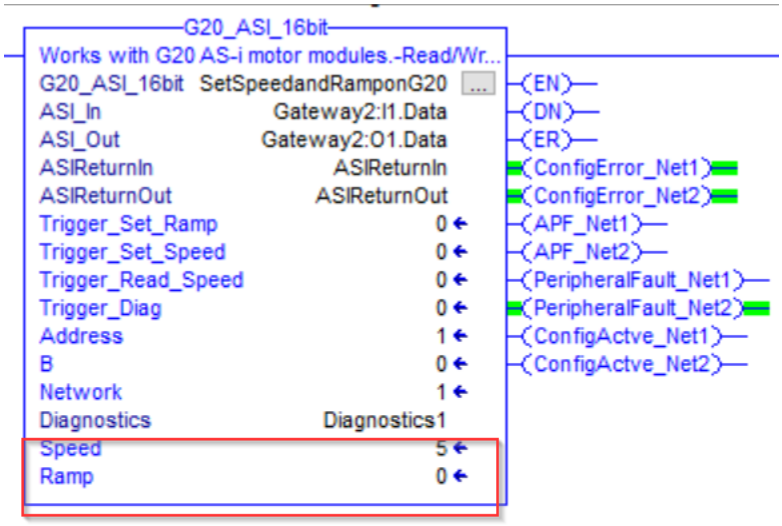
Set the AS-Interface network and AS-Interface address for which the ramp or speed will be configured.



Setup the ASIReturnOut and ASIReturnIn variables. These will be used for all inputs on your machine and out and the diagnostics variable that will be used to store diagnostic data.



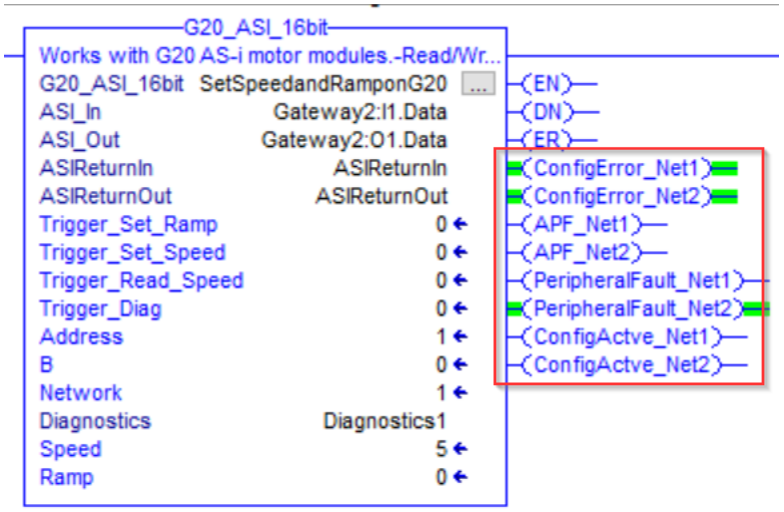
Input the proper speed and ramp values that will be set on the motor control module.



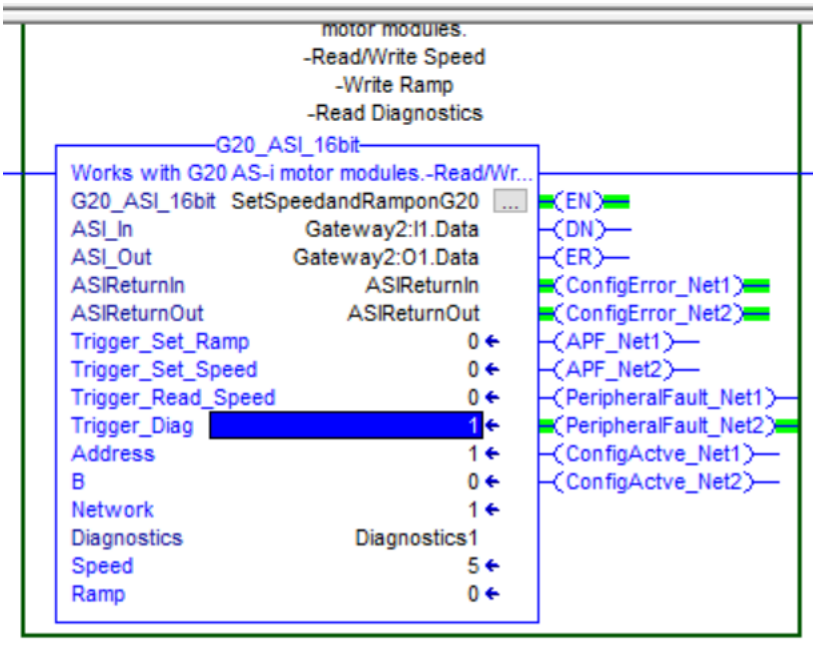
Reading Diagnostic Data

Diagnostic data must only be retrieved when one of the diagnostic bits is set. If a network is not being used at all, one of the bits may stay on all the time. Only one trigger bit can be used at a time.

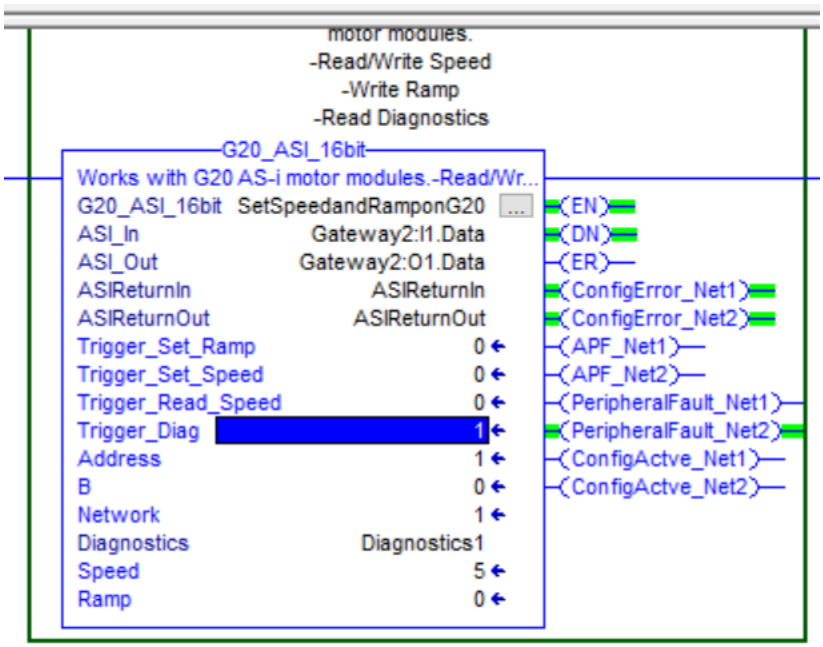
Network = 1 Read diagnostics for network 1
Network = 2 Read diagnostics for network 2



Wait for the diagnostic bit to turn on and lower all the trigger bits. Make sure that the EN, DN, and ER bits are all low. Raise the Trigger_Diag bit. The EN bit will turn on and tell you the instruction is being executed.



If successful, the DN bit will turn on and the ER bit will be off.



The add-on is only valid if the connection is not faulted. Check the ConnectionFaulted variable associated with the AS-Interface gateway and make sure it is zero.

Gateway2:I1	{...}	{...}			_0039:254539_41...
Gateway2:I1.ConnectionFaulted	0			Decimal	BOOL
Gateway2:I1.Data	{...}	{...}		Decimal	INT[51]
Gateway2:O1	{...}	{...}			_0039:254539_6E...

Figure 11: ConnectionFaulted variable shows if the AS-Interface gateway is running.

The advanced diagnostic data is available in the variable called "Diagnostics." Diagnostics[0] is for network 1 and Diagnostics[1] for network 2. See Figure 7. Here is a detailed description of each set of diagnostics.

LAS_A(DINT) – List of Activated Slaves: This is one bit per slave. If the node is exchanging data, the bit is on. For slaves 1A to 31A or 1-31.

LAS_B(DINT) – List of Activated Slaves: This is one bit per slave. If the node is exchanging data, the bit is on. For slaves 1B to 31B.

LDS_A(DINT) – List of Detected Slaves: This is one bit per slave. If the node is detected, the bit is on. For slaves 1A to 31A or 1-31.

LDS_B(DINT) – List of Detected Slaves: This is one bit per slave. If the node is detected, the bit is on. For slaves 1B to 31B.

LPS_A(DINT) – List of Projected Slaves: This is one bit per slave. If on, the node should be in the configuration. For slaves 1A to 31A or 1-31.

LPS_B(DINT) – List of Projected Slaves: This is one bit per slave. If on, the node should be in the configuration. For slaves 1B to 31B.

Delta_A(DINT) – List of slaves with a configuration error: This is one bit per slave. If on, the slave has an error. For slaves 1A to 31A or 1-31.

Delts_B(DINT) – List of slaves with a configuration error: This is one bit per slave. If on, the slave has an error. For slaves 1B-31B.

LCS_A(DINT) – List of corrupted slaves: This is one bit per slave. If a node had a prior configuration error, it shows up as a bit here. It is cleared as soon as it is read. Bit 0 will be on for a past power failure. For slaves 1A to 31A or 1-31. If you want to save/use this information, move the data to another register when not zero, otherwise it will be overwritten.

LCS_B(DINT) – List of corrupted slaves: This is one bit per slave. If a node had a prior configuration error, it shows up as a bit here. It is cleared as soon as it is read. Bit 0 will be on for a past power failure. For slaves 1B to 31B. If you want to save/use this information, move the data to another register when not zero, otherwise it will be overwritten.

DUP_A(DINT) – List of duplicate addresses: This is one bit per slave. If two nodes have the same address, it is displayed here. For slaves 1A to 31A or 1-31.

DUP_B(DINT) – List of duplicate addresses: This is one bit per slave. If two nodes have the same address, it is displayed here. For slaves 1B-31B.

Fault_Detector_Historic(SINT) – Shows additional advanced diagnostic bits. Historic means it happened in the past and may not be currently affecting the system.

Bit 0 – Not used

Bit 1 – Not used

Bit 2 – Pfa, failure of 24 V on safety monitor (only on gateway with built-in safety monitor)

Bit 3 – PFr, failure of redundant 24 V (only on single masters with redundant supply)

Bit 4 – ES, earth fault, ground fault

Bit 5 – Overvoltage

Bit 6 – Noise

Bit 7 – Duplicate address

Fault_Detector_Now(SINT) – Shows additional advanced diagnostic bits. Affects the current system.

Bit 0 – Reserved

Bit 1 – Reserved

Bit 2 – Pfa, failure of 24 V on safety monitor (only on gateway with built-in safety monitor)

Bit 3 – PFr, failure of redundant 24 V (only on single masters with redundant supply)

Bit 4 – ES, earth fault, ground fault

Bit 5 – Overvoltage

Bit 6 – Noise

Bit 7 – Duplicate address

Peripherals_OK(BOOL) – On if there is no peripheral fault.

ADR_0_Exists – A node with address 0 exists on the network.

Auto_Adr_Possible(BOOL) – Shows that the conditions are right such that automatic addressing would work if required, i.e., protected mode, auto addressing enabled.

Auto_Adr_Active(BOOl) – Automatic addressing is running, i.e., only one slave is missing (more than one slave can be missing if they have different configurations).

Config_Mode_Active(BOOL) – Configuration mode is running, PRJ mode LED is ON.

Normal_Op_Active(BOOL) – Protected mode is running.

ASi_Power_Fail(BOOL) – On if voltage is too low or current limit has been exceeded.

Offline_Ready(BOOL) – On if network is offline.

Config_OK(BOOL) – On if there is no configuration error on the network.

Auto_Adr_Enabled(BOOL) – Automatic addressing is enabled (has not been disabled by the user).

Offline(BOOL) – On if network is offline.

Data_Exchange_Active(BOOL) – Communication with slaves is running.

Error_Counters_A(SINT[32]) – Counts up when a retry on AS-Interface has occurred at that specific address. The array index is the module address. It is cleared every time the data is read. When the data is not zero, you can move it or store the data if required. For nodes 1A-31A or 1-31.

Error_Counters_B(SINT[32]) – Counts up when a retry on AS-Interface has occurred at that specific address. The array index is the module address. It is cleared every time the data is read. When the data is not zero, you can move it or store the data if required. For nodes 1B-31B.

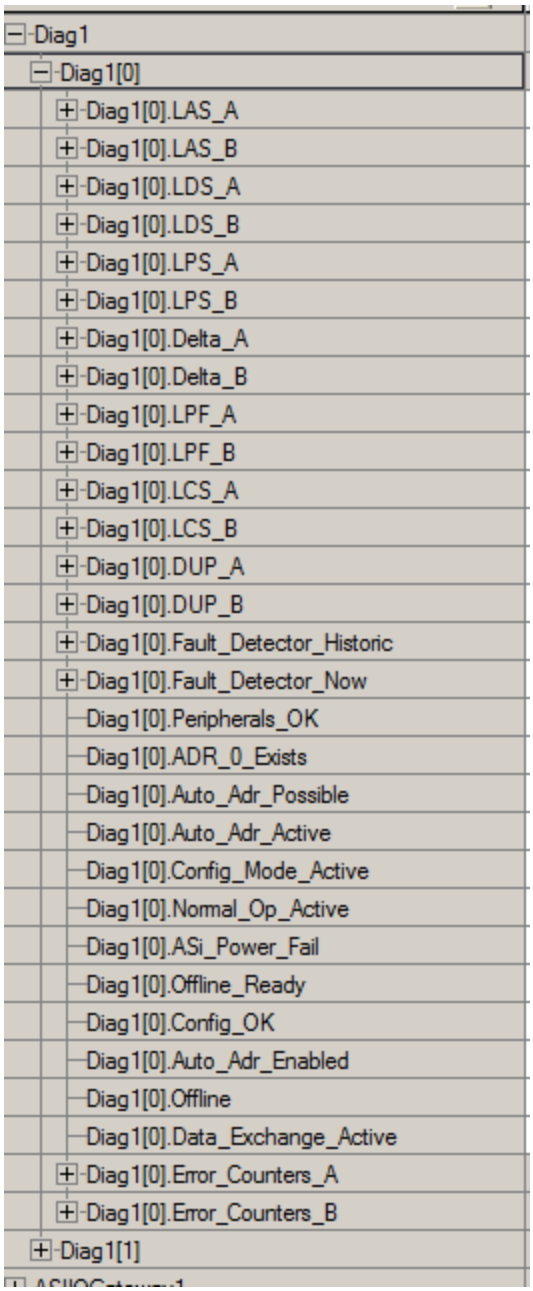
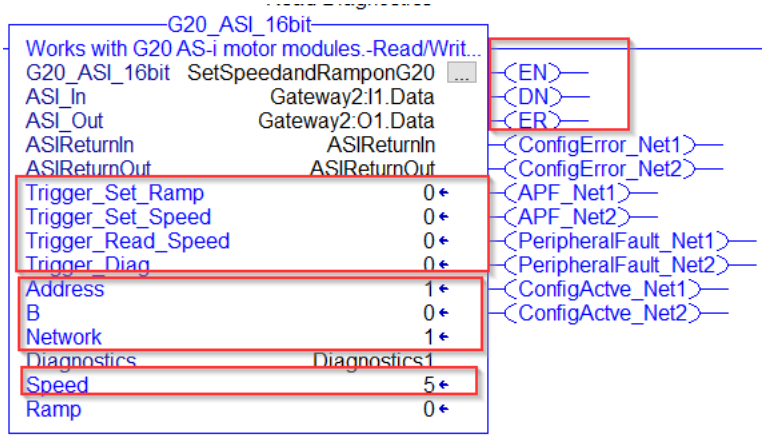


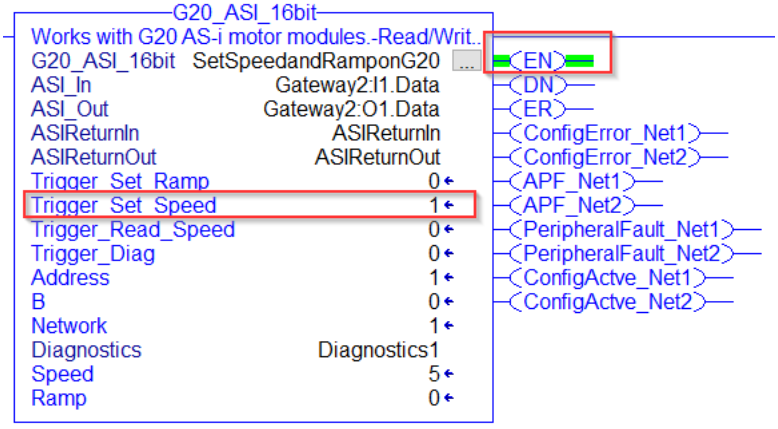
Figure 7: Diagnostics data per network

Setting the Speed

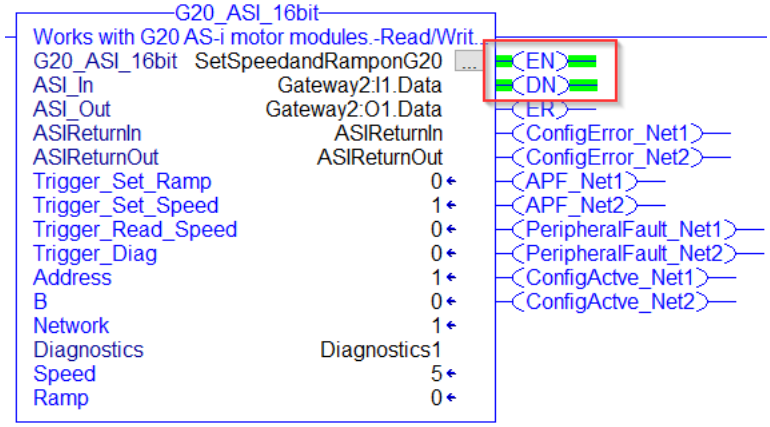
The speed can be set on the fly. If the speed changes while the motor is running, the motor will immediately jump to the new speed setting. Lower all trigger bits and set the address and network—the speed will be set to on. Make sure the output bits EN, DN, and ER are all off.



Set the Trigger_Set_Speed bit, and the EN bit will let you know the command is being sent.



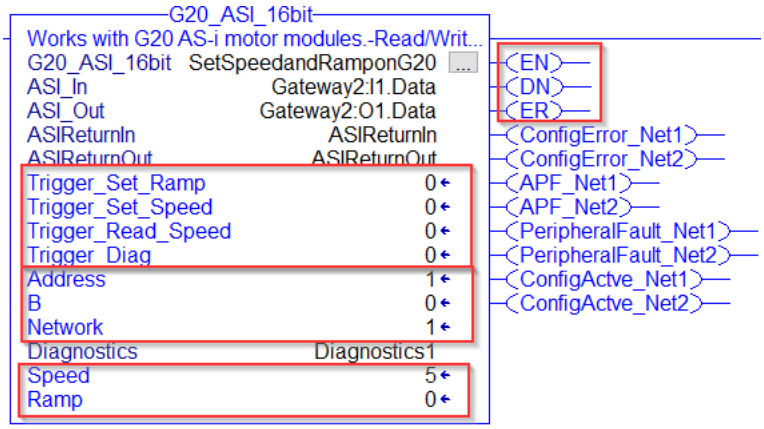
When the DN bit is on and the ER bit is off, the command has been completed successfully.



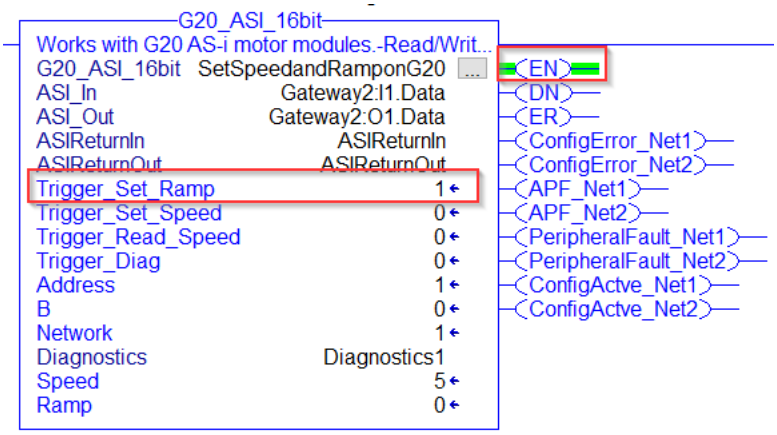
Setting the Ramp

Setting the ramp stops the motor. Only run the command if the motor outputs are off. The speed is also reset at the end of the ramp-setting procedure. Make sure the ramp is set to the correct value.

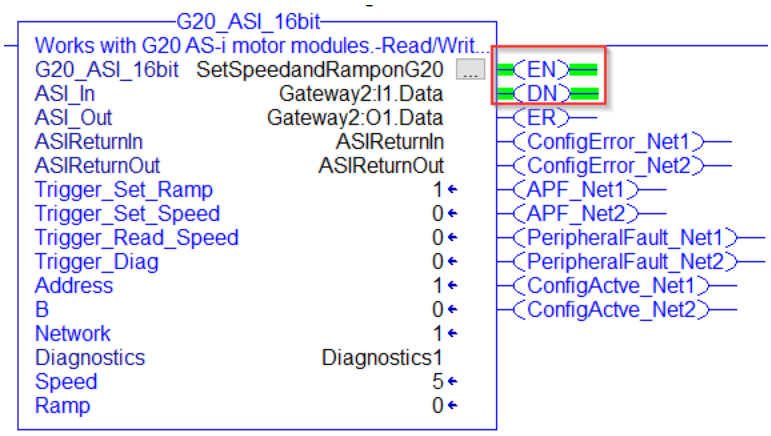
Set all the trigger bits to off and verify that the EN, DN, and ER bits are all off. Set the address and network for the ramp. Set the ramp to a value.



Set the Trigger_Set_Ramp on and the EN bit will let you know when the command starts to execute.

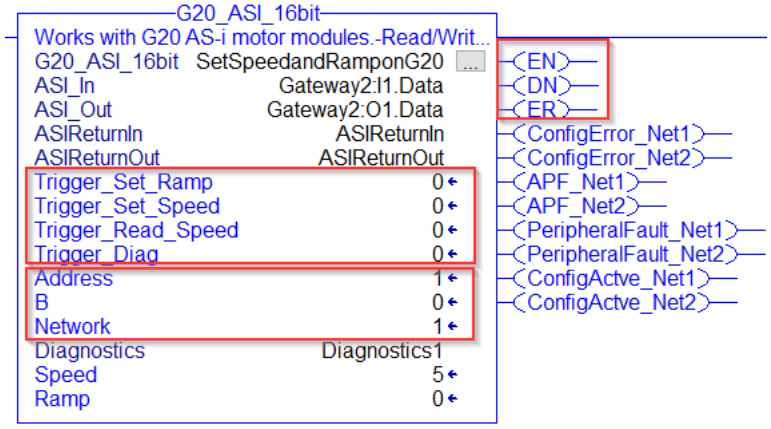


When the DN bit comes on and the ER bit remains off, the ramp has been set successfully.

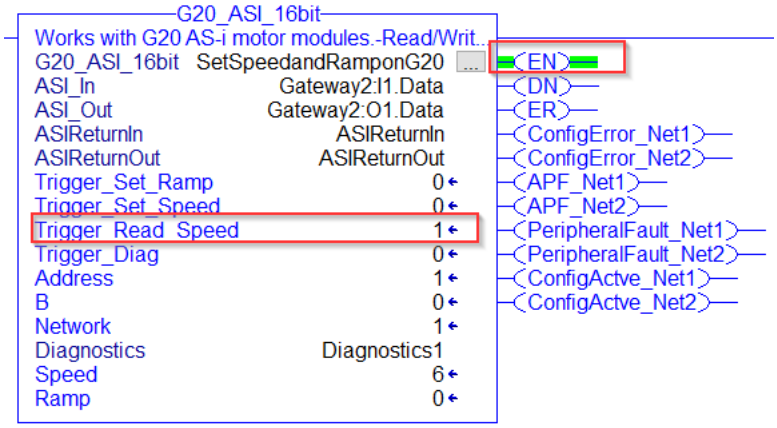


Reading the Speed

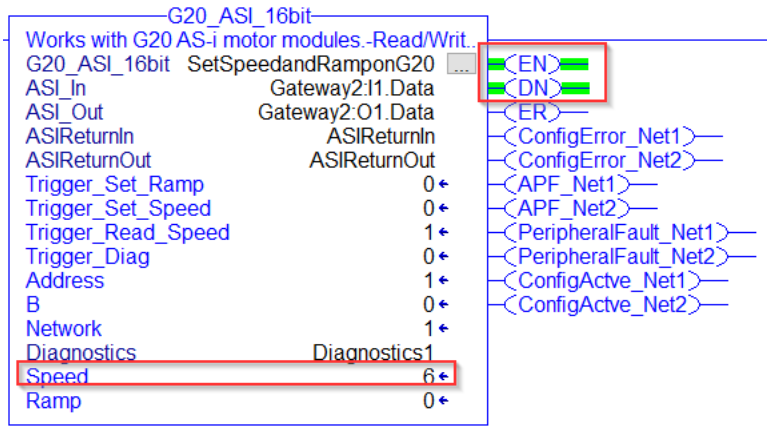
The speed can be read at any time. Make sure all trigger bits are low and the EN, DN, and ER bits are off. The add-on will read the speed of one module. Enter the AS-Interface network and address from which you will read the speed.



Raise the Trigger_Read_Speed bit and the EN bit will turn on to tell you the command is enabled.



Once the DN turns on, the speed is automatically loaded into the “Speed” parameter in the add-on.



EDS File

Download the EDS file from our website. Each gateway has its own EDS file, so make sure to install and use the correct one. Once the EDS files are downloaded, install them using the EDS Hardware Installation Tool. See Figure 8. Two EDS files are associated with each product. One is for optional use with DLR. Only use this one if DLR will be used because the minimum RPI rates are higher.

VBG-ENX-K20-DMD-EV – Gateway for two networks

VBG-ENX-K30-DMD-S16-EV – Gateway for two networks with built-in safety monitor

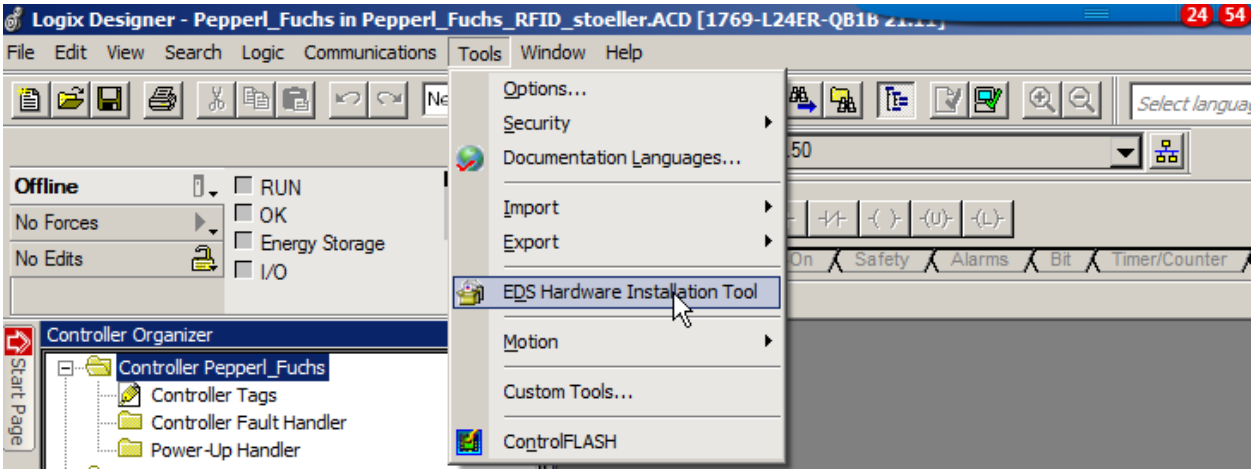


Figure 8: Install EDS file