

To read faults you can either use the AKD_Get_Parameter AOI method or the MSG instruction. The method is up to the programmer.

Our sample project for AKD1G shows the AOI method for reading the current faults (link is shared below).

The Instances for Ethernet IP for DRV.FAULT1 through 10 are shown in the Web Help (478-487) [DRV.FAULT1 to DRV.FAULT10 \(kollmorgen.com\)](http://kollmorgen.com)

Fieldbus	Parameter	Instance	Data Size	Data Type
EtherNet/IP	DRV.FAULT1	478	2 Byte	Integer
	DRV.FAULT2	479		
	DRV.FAULT3	480		
	DRV.FAULT4	481		
	DRV.FAULT5	482		
	DRV.FAULT6	483		
	DRV.FAULT7	484		
	DRV.FAULT8	485		
	DRV.FAULT9	486		
	DRV.FAULT10	487		

The EIP Status Word 1 (Byte 0 of the response assembly) has a General Fault bit to indicate faulted/no fault present.

6.2.3.2 Status Word 1

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Enable State	Reserved	Homed	Current Direction	General Fault	In Position	Block in Execution	In Motion

Enable State: This bit reflects the enable state of the amplifier.

Homed: This bit is set when the drive has been successfully homed.

Current Direction: This bit reflects the actual direction of motion.

General Fault: This bit indicates whether or not a fault has occurred.

In Position: This bit indicates whether or not the motor is on the last targeted position (1=On Target).

Block in Execution: When set, indicates the amplifier is running a motion task.

Executing Block # (Byte 1 in Response Assembly): Indicates the index of the currently executing Motion Task when the Block in Execution bit is set.

In Motion: This bit indicates whether a trajectory is in progress (1) or has completed (0).

This bit is set immediately when motion begins and remains set for the entire motion.

Moving to the Sample Project (see the zip file https://www.kollmorgen.com/sites/default/files/public_downloads/EtherNetIP_AIO_v6.0.zip) and look for the AKD_Gen1 Sample Project.

Name	Type	Compressed size	Password p...	Size	Ratio	Date modified
AOIs v6_0	File folder					1/28/2022 1:02 PM
Data Types v6_0	File folder					1/28/2022 1:02 PM
Sample_Project_AKD_Gen1_V6_0.ACD	ACD File	1,962 KB	No	2,211 KB	12%	1/28/2022 1:50 PM
Sample_Project_AKD2G_V6_0.ACD	ACD File	2,027 KB	No	2,281 KB	12%	11/22/2021 8:38 AM

When the axis name is declared (i.e. AXIS_ONE in the Sample project) in the AKD_Drive AOI's axis internal field, a lot of status and diagnostics are already defined in the structure that is created in the controller tags.

In the Status Word there is a General Fault bit that indicates if a fault is present on the axis (drive).

Drive Communication

AKD_Drive	
AKD_Drive	AKD1G ...
Axis_Input	akd1g:1
Axis_Output	akd1g:0
Axis_Internal	AXIS_ONE

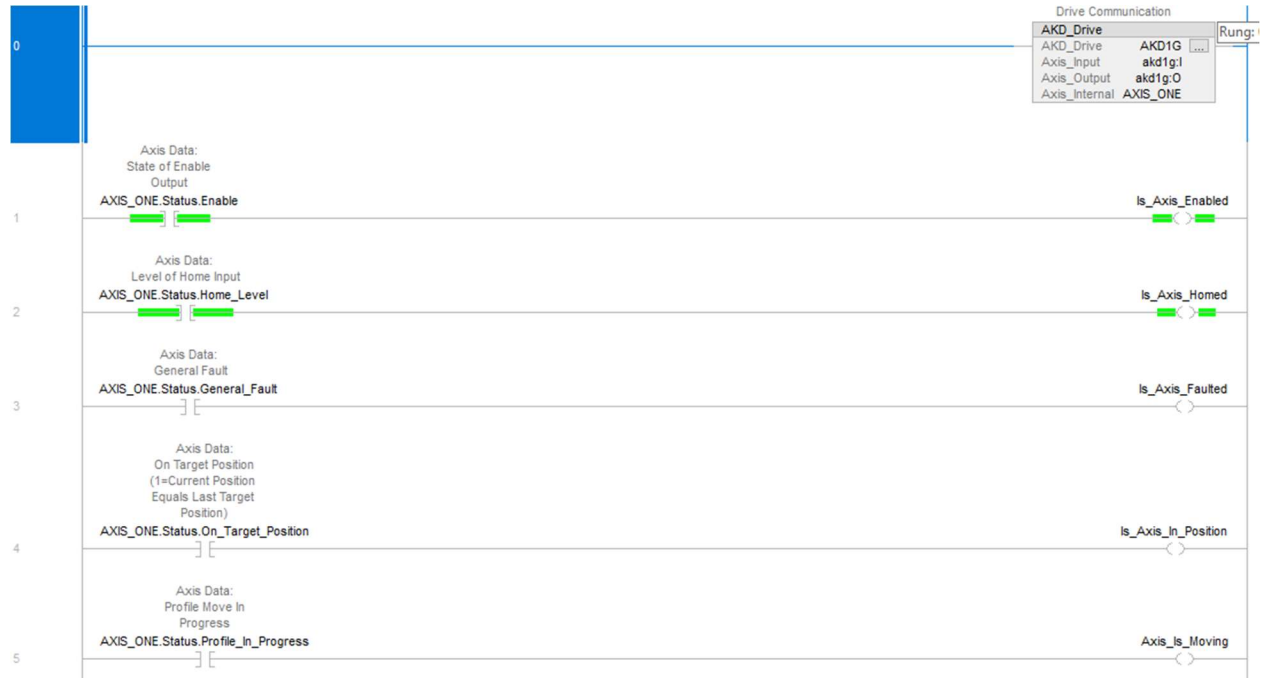
Controller Organizer

Controller Tags - Test_AKD_Gen1(controller)

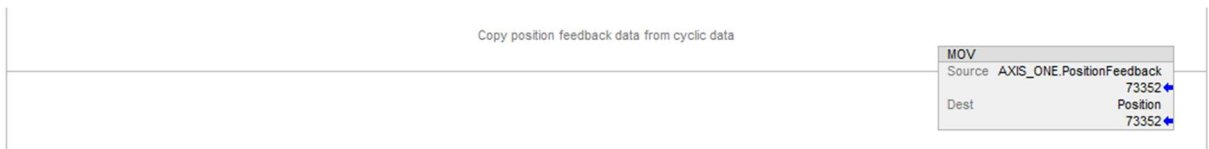
Name	Value	Force Mask	Style	Data Type
AXIS_ONE.Status		{...}	{...}	AKD_Status
AXIS_ONE.Status.Profile_In_Progress	0		Decimal	BOOL
AXIS_ONE.Status.Block_In_Execution	0		Decimal	BOOL
AXIS_ONE.Status.On_Target_Position	0		Decimal	BOOL
AXIS_ONE.Status.General_Fault	0		Decimal	BOOL
AXIS_ONE.Status.Current_Direction	0		Decimal	BOOL
AXIS_ONE.Status.Home_Level	1		Decimal	BOOL
AXIS_ONE.Status.Reg_Level	0		Decimal	BOOL
AXIS_ONE.Status.Enable	1		Decimal	BOOL
AXIS_ONE.Status.Fault_Input_Fault	0		Decimal	BOOL
AXIS_ONE.Status.Fwd_Limit	0		Decimal	BOOL
AXIS_ONE.Status.Rev_Limit	0		Decimal	BOOL
AXIS_ONE.Status.Positive_Limit	0		Decimal	BOOL
AXIS_ONE.Status.Negative_Limit	0		Decimal	BOOL
AXIS_ONE.Status.FE_Fault	0		Decimal	BOOL
AXIS_ONE.Status.Block_Fault	0		Decimal	BOOL
AXIS_ONE.Status.Load_Complete	0		Decimal	BOOL
AXIS_ONE.Input	{...}	{...}		AKD_Data

In the Main Routine of our sample project statuses are demonstrated in the ladder for monitoring etc. such as Is_Axis_Homed, Is_Axis_Faulted, etc.

You can monitor when the axis (drive) is faulted using this method and then read the faults any time the general fault bit becomes true (=1).

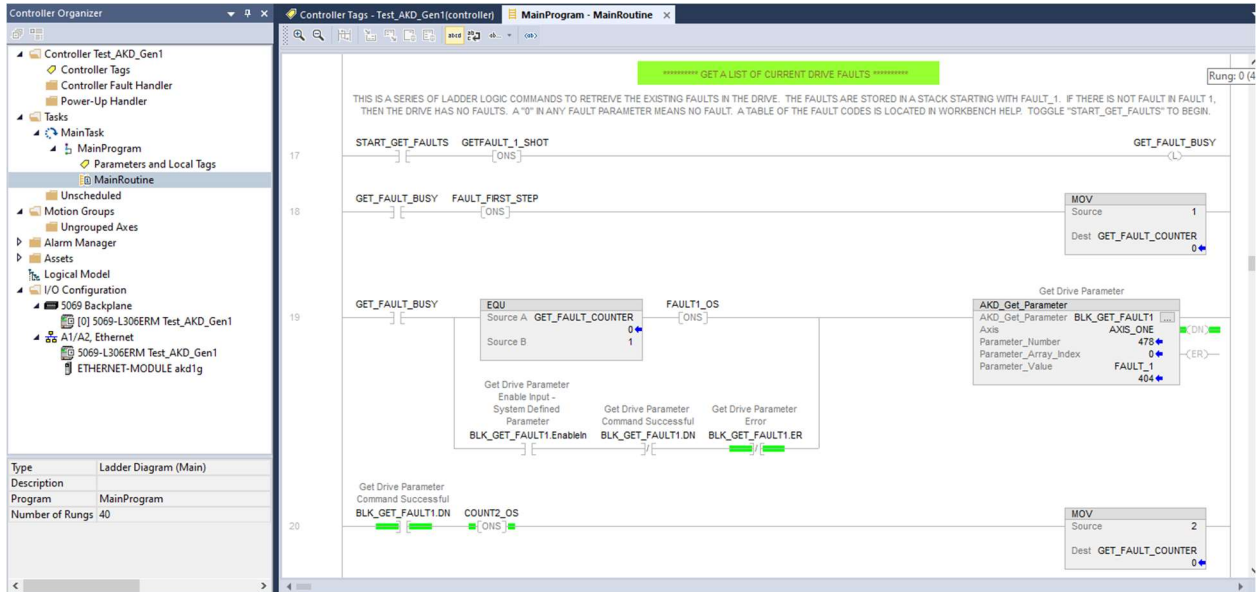


We also show the position feedback in the ladder.



At the bottom of the ladder we show a method using the AKD_Get_Parameter AOI to read the 10 fault parameters via their EIP instances shown above. You could also do this with the MSG instruction instead.

The method reads each fault/instance in succession using a counter as shown in the sample project.



The following manual has a section with examples of explicit messaging and the MSG instruction if you elect to go that route.

https://www.kollmorgen.com/sites/default/files/public_downloads/AKD_AKD2G_EIP_Studio_5000.pdf