

| | | | | | | | | | |
|--------------------------------------|----------------|----------------------|----------|---------------|--|----------------|----------|---|--|
| Totally Integrated Automation Portal | | | | | | | | | |
| AKD_PN_TG1_v01 [FB1] | | | | | | | | | |
| AKD_PN_TG1_v01 Properties | | | | | | | | | |
| General | | | | | | | | | |
| Name | AKD_PN_TG1_v01 | Number | 1 | Type | FB | Language | SCL | | |
| Numbering | automatic | | | | | | | | |
| Information | | | | | | | | | |
| Title | AKD_PN_TG1_v01 | Author | jcoleman | Comment | This function block supports Profinet communication with the AKD drive using Telegram 1. It is written in TIA V13. | Family | TG1 | | |
| Version | 1.0 | User-defined ID | | | | | | | |
| | | | | | | | | | |
| Name | | Data type | Offset | Default value | Accessible from HMI | Visible in HMI | Setpoint | Comment | |
| ▼ Input | | | | | | | | | |
| iID | | HW_IO | 0.0 | 16#0 | False | False | False | Hardware identifier | |
| iEStop | | Bool | 2.0 | false | False | False | False | immediate quick stop and disable | |
| iReset | | Bool | 2.1 | false | False | False | False | reset drive faults | |
| iSW_Enable | | Bool | 2.2 | false | False | False | False | enable/disable | |
| iStop | | Bool | 2.3 | false | False | False | False | stop all movement | |
| iFreezeVcmd | | Bool | 2.4 | false | False | False | False | pause active motion task | |
| iSpeedCmd | | Int | 4.0 | 0 | False | False | False | | |
| iStart_Move | | Bool | 6.0 | false | False | False | False | start motion task | |
| ▼ Output | | | | | | | | | |
| oStatus | | DWord | 8.0 | 16#0 | False | False | False | ZSW1 | |
| oEnabled | | Bool | 12.0 | false | False | False | False | Drive power on | |
| oError | | Bool | 12.1 | false | False | False | False | fault in function block or commu- nication | |
| oAtSpeed | | Bool | 12.2 | false | False | False | False | | |
| oFault | | Bool | 12.3 | false | False | False | False | drive fault | |
| oWarning | | Bool | 12.4 | false | False | False | False | drive warning | |
| InOut | | | | | | | | | |
| ▼ Static | | | | | | | | | |
| ▼ TG1 | | "UDT_AKD_TG1" | 14.0 | | False | False | False | | |
| ▼ Send | | Struct | 0.0 | | False | False | False | | |
| STW1 | | Int | 0.0 | 0 | False | False | False | Control word 1 | |
| NSOLL_A | | Int | 2.0 | 0 | False | False | False | Velocity command (VL.CMD) | |
| ▼ Receive | | Struct | 4.0 | | False | False | False | | |
| ZSW1 | | Int | 0.0 | 0 | False | False | False | Status word 1 | |
| NIST_A | | Int | 2.0 | 0 | False | False | False | Actual Velocity (VL.FB) | |
| wPNReadStatus | | Word | 22.0 | 16#0 | False | False | False | | |
| bPNReadError | | Bool | 24.0 | false | False | False | False | | |
| wPNWriteStatus | | Word | 26.0 | 16#0 | False | False | False | | |
| bPNWriteError | | Bool | 28.0 | false | False | False | False | | |
| bResetDone | | Bool | 28.1 | false | False | False | False | | |
| bHomeStarted | | Bool | 28.2 | false | False | False | False | | |
| bStartMoveDone | | Bool | 28.3 | false | False | False | False | | |
| bEstop | | Bool | 28.4 | false | False | False | False | | |
| wStartReset | | Int | 30.0 | 0 | False | False | False | | |
| wStartMove | | Int | 32.0 | 0 | False | False | False | | |
| wSDOStatus | | Word | 34.0 | 16#0 | False | False | False | | |
| ▼ bySDODdataArray | | Array[0..63] of Byte | 36.0 | | False | False | False | | |
| bySDODdataArray[0] | | Byte | 0.0 | 16#0 | False | False | False | | |
| bySDODdataArray[1] | | Byte | 1.0 | 16#0 | False | False | False | | |
| bySDODdataArray[2] | | Byte | 2.0 | 16#0 | False | False | False | | |
| bySDODdataArray[3] | | Byte | 3.0 | 16#0 | False | False | False | | |
| bySDODdataArray[4] | | Byte | 4.0 | 16#0 | False | False | False | | |
| bySDODdataArray[5] | | Byte | 5.0 | 16#0 | False | False | False | | |
| bySDODdataArray[6] | | Byte | 6.0 | 16#0 | False | False | False | | |
| bySDODdataArray[7] | | Byte | 7.0 | 16#0 | False | False | False | | |
| bySDODdataArray[8] | | Byte | 8.0 | 16#0 | False | False | False | | |
| bySDODdataArray[9] | | Byte | 9.0 | 16#0 | False | False | False | | |
| bySDODdataArray[10] | | Byte | 10.0 | 16#0 | False | False | False | | |
| bySDODdataArray[11] | | Byte | 11.0 | 16#0 | False | False | False | | |
| bySDODdataArray[12] | | Byte | 12.0 | 16#0 | False | False | False | | |
| bySDODdataArray[13] | | Byte | 13.0 | 16#0 | False | False | False | | |
| bySDODdataArray[14] | | Byte | 14.0 | 16#0 | False | False | False | | |
| bySDODdataArray[15] | | Byte | 15.0 | 16#0 | False | False | False | | |
| bySDODdataArray[16] | | Byte | 16.0 | 16#0 | False | False | False | | |
| bySDODdataArray[17] | | Byte | 17.0 | 16#0 | False | False | False | | |
| bySDODdataArray[18] | | Byte | 18.0 | 16#0 | False | False | False | | |
| bySDODdataArray[19] | | Byte | 19.0 | 16#0 | False | False | False | | |
| bySDODdataArray[20] | | Byte | 20.0 | 16#0 | False | False | False | | |
| bySDODdataArray[21] | | Byte | 21.0 | 16#0 | False | False | False | | |
| bySDODdataArray[22] | | Byte | 22.0 | 16#0 | False | False | False | | |
| bySDODdataArray[23] | | Byte | 23.0 | 16#0 | False | False | False | | |
| | | | | | | | | | |

| | | | | | | | | |
|--------------------------------------|--|-----------|--------|---------------|---------------------|----------------|----------|---|
| Totally Integrated Automation Portal | | | | | | | | |
| Name | | Data type | Offset | Default value | Accessible from HMI | Visible in HMI | Setpoint | Comment |
| bySDODataArray[24] | | Byte | 24.0 | 16#0 | False | False | False | |
| bySDODataArray[25] | | Byte | 25.0 | 16#0 | False | False | False | |
| bySDODataArray[26] | | Byte | 26.0 | 16#0 | False | False | False | |
| bySDODataArray[27] | | Byte | 27.0 | 16#0 | False | False | False | |
| bySDODataArray[28] | | Byte | 28.0 | 16#0 | False | False | False | |
| bySDODataArray[29] | | Byte | 29.0 | 16#0 | False | False | False | |
| bySDODataArray[30] | | Byte | 30.0 | 16#0 | False | False | False | |
| bySDODataArray[31] | | Byte | 31.0 | 16#0 | False | False | False | |
| bySDODataArray[32] | | Byte | 32.0 | 16#0 | False | False | False | |
| bySDODataArray[33] | | Byte | 33.0 | 16#0 | False | False | False | |
| bySDODataArray[34] | | Byte | 34.0 | 16#0 | False | False | False | |
| bySDODataArray[35] | | Byte | 35.0 | 16#0 | False | False | False | |
| bySDODataArray[36] | | Byte | 36.0 | 16#0 | False | False | False | |
| bySDODataArray[37] | | Byte | 37.0 | 16#0 | False | False | False | |
| bySDODataArray[38] | | Byte | 38.0 | 16#0 | False | False | False | |
| bySDODataArray[39] | | Byte | 39.0 | 16#0 | False | False | False | |
| bySDODataArray[40] | | Byte | 40.0 | 16#0 | False | False | False | |
| bySDODataArray[41] | | Byte | 41.0 | 16#0 | False | False | False | |
| bySDODataArray[42] | | Byte | 42.0 | 16#0 | False | False | False | |
| bySDODataArray[43] | | Byte | 43.0 | 16#0 | False | False | False | |
| bySDODataArray[44] | | Byte | 44.0 | 16#0 | False | False | False | |
| bySDODataArray[45] | | Byte | 45.0 | 16#0 | False | False | False | |
| bySDODataArray[46] | | Byte | 46.0 | 16#0 | False | False | False | |
| bySDODataArray[47] | | Byte | 47.0 | 16#0 | False | False | False | |
| bySDODataArray[48] | | Byte | 48.0 | 16#0 | False | False | False | |
| bySDODataArray[49] | | Byte | 49.0 | 16#0 | False | False | False | |
| bySDODataArray[50] | | Byte | 50.0 | 16#0 | False | False | False | |
| bySDODataArray[51] | | Byte | 51.0 | 16#0 | False | False | False | |
| bySDODataArray[52] | | Byte | 52.0 | 16#0 | False | False | False | |
| bySDODataArray[53] | | Byte | 53.0 | 16#0 | False | False | False | |
| bySDODataArray[54] | | Byte | 54.0 | 16#0 | False | False | False | |
| bySDODataArray[55] | | Byte | 55.0 | 16#0 | False | False | False | |
| bySDODataArray[56] | | Byte | 56.0 | 16#0 | False | False | False | |
| bySDODataArray[57] | | Byte | 57.0 | 16#0 | False | False | False | |
| bySDODataArray[58] | | Byte | 58.0 | 16#0 | False | False | False | |
| bySDODataArray[59] | | Byte | 59.0 | 16#0 | False | False | False | |
| bySDODataArray[60] | | Byte | 60.0 | 16#0 | False | False | False | |
| bySDODataArray[61] | | Byte | 61.0 | 16#0 | False | False | False | |
| bySDODataArray[62] | | Byte | 62.0 | 16#0 | False | False | False | |
| bySDODataArray[63] | | Byte | 63.0 | 16#0 | False | False | False | |
| ▼ SDOWrite | | WRREC | | | False | False | False | |
| ▼ Input | | | | | | | | |
| REQ | | Bool | | FALSE | False | False | False | REQ = 1: Transfer data record |
| ID | | HW_IO | | 16#0 | False | False | False | HW-Id of the DP slave/PROFINET IO component |
| INDEX | | DInt | | 0 | False | False | False | Data record number |
| LEN | | UInt | | 0 | False | False | False | Maximum length in bytes of the data |
| ▼ Output | | | | | | | | |
| DONE | | Bool | | FALSE | False | False | False | Function performed |
| BUSY | | Bool | | FALSE | False | False | False | Function busy |
| ERROR | | Bool | | FALSE | False | False | False | Error flag |
| STATUS | | DWord | | DW#16#0 | False | False | False | Function result/error message |
| ▼ InOut | | | | | | | | |
| RECORD | | Variant | | | False | False | False | Data record |
| Static | | | | | | | | |
| ▼ SDORead | | RDREC | | | False | False | False | |
| ▼ Input | | | | | | | | |
| REQ | | Bool | | FALSE | False | False | False | REQ = 1: Transfer data record |
| ID | | HW_IO | | 16#0 | False | False | False | HW-Id of the DP slave/PROFINET IO component |
| INDEX | | DInt | | 0 | False | False | False | Data record number |
| MLEN | | UInt | | 0 | False | False | False | Maximum length in bytes of the data |
| ▼ Output | | | | | | | | |
| VALID | | Bool | | FALSE | False | False | False | Function performed |
| BUSY | | Bool | | FALSE | False | False | False | Function busy |
| ERROR | | Bool | | FALSE | False | False | False | Error flag |
| STATUS | | DWord | | DW#16#0 | False | False | False | Function result/error message |
| LEN | | UInt | | 0 | False | False | False | Length of the fetched data record |
| ▼ InOut | | | | | | | | |
| RECORD | | Variant | | | False | False | False | Target area for the fetched data record |
| Static | | | | | | | | |
| bSDOReq | | Bool | 100.0 | false | False | False | False | |
| bSDOWriteBusy | | Bool | 100.1 | false | False | False | False | |
| bSDOWriteDone | | Bool | 100.2 | false | False | False | False | |
| | | | | | | | | |

| | | |
|--------------------------------------|---|------------------------------------|
| Totally Integrated Automation Portal | | |
| 0061 | IF #iEstop THEN | |
| 0062 | #TG1.Send.STW1.%X2 := FALSE; | |
| 0063 | IF #TG1.Receive.ZSW1.%X13 THEN | |
| 0064 | #bEstop := TRUE; | |
| 0065 | END_IF; | |
| 0066 | END_IF; | |
| 0067 | | |
| 0068 | // Do a reset | |
| 0069 | IF #iReset AND NOT #bResetStarted THEN | |
| 0070 | #wSDOStatus := 16; | |
| 0071 | #TG1.Send.STW1.%X7 := TRUE; | |
| 0072 | #bEstop := FALSE; | |
| 0073 | #bResetStarted := TRUE; | |
| 0074 | #wResetCounter := 0; | |
| 0075 | END_IF; | |
| 0076 | IF #bResetStarted THEN | |
| 0077 | #wResetCounter := #wResetCounter + 1; | |
| 0078 | END_IF; | |
| 0079 | IF #wResetCounter > 3 THEN | |
| 0080 | #TG1.Send.STW1.%X7 := FALSE; | |
| 0081 | #bResetStarted := FALSE; | |
| 0082 | #wResetCounter := 0; | |
| 0083 | END_IF; | |
| 0084 | | |
| 0085 | // Enable the drive | |
| 0086 | IF #iSW_Enable AND NOT #bEstop AND NOT (#bSWEnabled AND #TG1.Receive.ZSW1.%X6 OR #TG1.Receive.ZSW1.%X3) | |
| | THEN | |
| 0087 | IF #TG1.Receive.ZSW1.%X6 THEN | // Switch on inhibited |
| 0088 | #TG1.Send.STW1.%X1 := TRUE; | |
| 0089 | #TG1.Send.STW1.%X2 := TRUE; | |
| 0090 | ELSE | |
| 0091 | IF #TG1.Receive.ZSW1.%X0 THEN | // Ready for switching on |
| 0092 | #TG1.Send.STW1.%X0 := TRUE; | |
| 0093 | IF #TG1.Receive.ZSW1.%X1 THEN | // Switched on |
| 0094 | #TG1.Send.STW1.%X3 := TRUE; | |
| 0095 | IF #TG1.Receive.ZSW1.%X2 THEN | // Operation |
| 0096 | #bSWEnabled := TRUE; | |
| 0097 | | |
| 0098 | // Do a Stop | |
| 0099 | IF #iStop THEN | |
| 0100 | #TG1.Send.STW1.%X4 := FALSE; | // Stop active motion task |
| 0101 | #TG1.Send.STW1.%X5 := FALSE; | // Stop active motion task |
| 0102 | #TG1.Send.STW1.%X8 := FALSE; | // Stop jog 1 |
| 0103 | #TG1.Send.STW1.%X9 := FALSE; | // Stop jog 2 |
| 0104 | ELSE | |
| 0105 | // Jogging1 | |
| 0106 | IF #iStart_Jog1 AND NOT #iStart_Move THEN | //do Jog1 |
| 0107 | #TG1.Send.STW1.%X4 := FALSE; | // disable ramp generator |
| 0108 | #TG1.Send.STW1.%X5 := TRUE; | // unfreeze ramp generator |
| 0109 | #TG1.Send.STW1.%X6 := FALSE; | |
| 0110 | #TG1.Send.STW1.%X9 := FALSE; | //not Jog2 |
| 0111 | #TG1.Send.STW1.%X8 := TRUE; | |
| 0112 | // Jogging2 | |
| 0113 | ELSIF #iStart_Jog2 AND NOT #iStart_Move THEN | //or do Jog2 |
| 0114 | #TG1.Send.STW1.%X4 := FALSE; | // disable ramp generator |
| 0115 | #TG1.Send.STW1.%X5 := TRUE; | // unfreeze ramp generator |
| 0116 | #TG1.Send.STW1.%X6 := FALSE; | |
| 0117 | #TG1.Send.STW1.%X8 := FALSE; | //not Jog1 |
| 0118 | #TG1.Send.STW1.%X9 := TRUE; | |
| 0119 | | |
| 0120 | ELSE | //otherwise turn off both jog bits |
| 0121 | #TG1.Send.STW1.%X4 := TRUE; | // enable ramp generator |
| 0122 | #TG1.Send.STW1.%X5 := TRUE; | // unfreeze ramp generator |
| 0123 | #TG1.Send.STW1.%X8 := FALSE; | |
| 0124 | #TG1.Send.STW1.%X9 := FALSE; | |
| 0125 | | |
| 0126 | // Interrupt a move | |
| 0127 | IF #iFreezeVcmd THEN | |
| 0128 | #TG1.Send.STW1.%X5 := FALSE; | // freeze ramp generator |
| 0129 | ELSE | |
| 0130 | #TG1.Send.STW1.%X5 := TRUE; | // unfreeze ramp generator |
| 0131 | | |
| 0132 | // Move | |
| 0133 | #TG1.Send.NSOLL_A := #iSpeedCmd; | // speed setpoint |
| 0134 | IF #iStart_Move AND NOT #iStart_Jog1 AND NOT #iStart_Jog2 THEN | |
| 0135 | #TG1.Send.STW1.%X4 := TRUE; | // enable ramp generator |
| 0136 | #TG1.Send.STW1.%X5 := TRUE; | // unfreeze ramp generator |
| 0137 | #TG1.Send.STW1.%X6 := TRUE; | // start the motion |
| 0138 | ELSE | |
| 0139 | #TG1.Send.STW1.%X6 := FALSE; | |
| 0140 | END_IF; | |
| 0141 | END_IF; | |
| 0142 | END_IF; | |
| 0143 | END_IF; | |
| 0144 | END_IF; | |
| 0145 | END_IF; | |
| 0146 | END_IF; | |
| 0147 | END_IF; | |
| | | |

| | | |
|--------------------------------------|---|--|
| Totally Integrated Automation Portal | | |
| 0148 | | |
| 0149 | // If the axis gets disabled all the corresponding motion siganl have to be reset | |
| 0150 | ELSE | |
| 0151 | #bSWEnabled := FALSE; | |
| 0152 | | |
| 0153 | #TGl.Send.STW1.%X0 := FALSE; | |
| 0154 | #TGl.Send.STW1.%X1 := FALSE; | |
| 0155 | #TGl.Send.STW1.%X2 := FALSE; | |
| 0156 | #TGl.Send.STW1.%X3 := FALSE; | |
| 0157 | #TGl.Send.STW1.%X4 := FALSE; | |
| 0158 | #TGl.Send.STW1.%X5 := FALSE; | |
| 0159 | #TGl.Send.STW1.%X6 := FALSE; | |
| 0160 | #TGl.Send.STW1.%X8 := FALSE; | |
| 0161 | #TGl.Send.STW1.%X9 := FALSE; | |
| 0162 | END_IF; | |
| 0163 | END_IF; | |
| 0164 | ELSE | |
| 0165 | #bPNWriteError := TRUE; | |
| 0166 | END_IF; | |
| 0167 | | |
| 0168 | // Read drive faults and warning on occurens | |
| 0169 | IF #oFault THEN | |
| 0170 | // If alarm is not read yet: start read | |
| 0171 | IF NOT #wSDOStatus.%X6 AND NOT #bSDOStatusReadActive THEN | |
| 0172 | #wSDOStatus := 64; | |
| 0173 | #wSDOAddress := 16#09AD; // Read PNU 2477 (DRV.FAULT1) | |
| 0174 | #bSDOStatusReadActive := TRUE; | |
| 0175 | END_IF; | |
| 0176 | | |
| 0177 | // If warning is not read yet: start read | |
| 0178 | ELSIF #oWarning THEN | |
| 0179 | IF NOT #wSDOStatus.%X5 AND NOT #bSDOStatusReadActive THEN | |
| 0180 | #wSDOStatus := 32; | |
| 0181 | #wSDOAddress := 16#0AE7; // Read PNU2791 (DRV.WARNING1) | |
| 0182 | #bSDOStatusReadActive := TRUE; | |
| 0183 | END_IF; | |
| 0184 | ELSE | |
| 0185 | IF NOT #bSDOStatusReadActive THEN | |
| 0186 | #wSDOStatus := 16; | |
| 0187 | END_IF; | |
| 0188 | END_IF; | |
| 0189 | | |
| 0190 | // start the reading signal | |
| 0191 | IF #wSDOStatus >= 32 THEN | |
| 0192 | // Always clean out old values first | |
| 0193 | IF #bSDOWriteBusy THEN | |
| 0194 | #bSDOReq := FALSE; | |
| 0195 | ELSE | |
| 0196 | FILL_BLK(IN := 0, | |
| 0197 | COUNT := 64, | |
| 0198 | OUT => #bySDODataArray[0]); | |
| 0199 | #bSDOReq := TRUE; | |
| 0200 | #bSDOWriteDone := FALSE; | |
| 0201 | END_IF; | |
| 0202 | // Start the read cylce to the corrspoding PNU | |
| 0203 | IF #wSDOStatus = 32 OR #wSDOStatus = 64 THEN | |
| 0204 | #bySDODataArray[0] := #wSDOStatus.%B0; | |
| 0205 | #bySDODataArray[1] := 16#01; | |
| 0206 | #bySDODataArray[2] := 16#00; | |
| 0207 | #bySDODataArray[3] := 16#01; | |
| 0208 | | |
| 0209 | #bySDODataArray[4] := 16#10; | |
| 0210 | #bySDODataArray[5] := 16#01; | |
| 0211 | #bySDODataArray[6] := #wSDOAddress.%B1; | |
| 0212 | #bySDODataArray[7] := #wSDOAddress.%B0; | |
| 0213 | #bySDODataArray[8] := 16#00; | |
| 0214 | #bySDODataArray[9] := 16#00; | |
| 0215 | | |
| 0216 | #SDOWrite(REQ := #bSDOReq, | |
| 0217 | ID := #iID, | |
| 0218 | INDEX := 47, | |
| 0219 | LEN := 14, | |
| 0220 | BUSY => #bSDOWriteBusy, | |
| 0221 | DONE => #bSDOWriteDone, | |
| 0222 | ERROR => #bSDOWriteError, | |
| 0223 | RECORD := #bySDODataArray); | |
| 0224 | IF #bSDOWriteDone THEN | |
| 0225 | #wSDOStatus := #wSDOStatus + 1; | |
| 0226 | END_IF; | |
| 0227 | END_IF; | |
| 0228 | | |
| 0229 | // read the value back | |
| 0230 | IF #wSDOStatus = 33 OR #wSDOStatus = 65 THEN | |
| 0231 | #SDORead(REQ := #bSDOReq, | |
| 0232 | ID := #iID, | |
| 0233 | INDEX := 47, | |
| 0234 | MLen := 14, | |
| 0235 | BUSY => #bSDOReadBusy, | |
| | | |

Totally Integrated Automation Portal

```
0236             VALID => #bSDOReadDone,
0237             ERROR => #bSDOReadError,
0238             RECORD := #bySDODataArray);
0239 IF #bSDOReadDone THEN
0240     IF #bySDODataArray[4] = 66 THEN // if the read signal is a Word use Byte 6&7
0241         #wSDOReadValue.%B0 := #bySDODataArray[7];
0242         #wSDOReadValue.%B1 := #bySDODataArray[6];
0243     ELSE // if the read signal is a DWord use Byte 8&9
0244         #wSDOReadValue.%B0 := #bySDODataArray[9];
0245         #wSDOReadValue.%B1 := #bySDODataArray[8];
0246     END_IF;
0247     IF #wSDOReadValue = 0 THEN // If the read value is zero repeat the read cycle
0248         #wSDOStatus := #wSDOStatus - 1;
0249     ELSE
0250         #wSDOStatus := #wSDOStatus + 1;
0251         #bSDOStatusReadActive := FALSE;
0252     END_IF;
0253 END_IF;
0254 END_IF;
0255 END_IF;
0256 END_IF;
0257
0258 // check for rising edge for the input parameters
0259 IF NOT #iReset AND #bResetStarted THEN
0260     #bResetStarted := FALSE;
0261 END_IF;
0262
0263 // fault detection/handling within the bloc
0264 IF #bPNReadError OR #bPNWriteError OR #bSDOReadError OR #bSDOWriteError THEN
0265     #oError := TRUE;
0266     IF #bPNReadError THEN
0267         #oStatus := #oStatus;
0268     END_IF;
0269     IF #bPNWriteError THEN
0270         #oStatus := #oStatus;
0271     END_IF;
0272 ELSIF #wSDOStatus > 15 THEN
0273     #oError := FALSE;
0274 END_IF;
0275
0276
```

| Symbol | Address | Type | Comment |
|-----------------------|---------|-----------|--|
| #bEstop | | Bool | |
| #bInitialize | | Bool | |
| #bMoveStarted | | Bool | |
| #bPNReadError | | Bool | |
| #bPNWriteError | | Bool | |
| #bResetStarted | | Bool | |
| #bSDOReadBusy | | Bool | |
| #bSDOReadDone | | Bool | |
| #bSDOReadError | | Bool | |
| #bSDOReq | | Bool | |
| #bSDOStatusReadActive | | Bool | |
| #bSDOWriteBusy | | Bool | |
| #bSDOWriteDone | | Bool | |
| #bSDOWriteError | | Bool | |
| #bSWEnabled | | Bool | |
| #bySDODataArray | | Array | |
| #iEstop | | Bool | immediate quick stop and disable |
| #iFreezeVcmd | | Bool | pause active motion task |
| #iID | | HW_IO | Hardware identifier |
| #iReset | | Bool | reset drive faults |
| #iSpeedCmd | | Int | |
| #iStart_Jog1 | | Bool | |
| #iStart_Jog2 | | Bool | start jog |
| #iStart_Move | | Bool | start motion task |
| #iStop | | Bool | stop all movement |
| #iSW_Enable | | Bool | enable/disable |
| #oAtSpeed | | Bool | |
| #oEnabled | | Bool | Drive power on |
| #oError | | Bool | fault in function block or communication |
| #oFault | | Bool | drive fault |
| #oStatus | | DWord | ZSW1 |
| #oStatus.%W0 | | Word | ZSW1 |
| #oStatus.%W1 | | Word | ZSW1 |
| #oWarning | | Bool | drive warning |
| #SDORead | | Multi_SFB | |
| #SDOWrite | | Multi_SFB | |
| #TG1.Receive | | Struct | |
| #TG1.Receive.ZSW1 | | Int | Status word 1 |
| #TG1.Receive.ZSW1.%X0 | | Bool | Status word 1 |
| #TG1.Receive.ZSW1.%X1 | | Bool | Status word 1 |
| #TG1.Receive.ZSW1.%X2 | | Bool | Status word 1 |
| #TG1.Receive.ZSW1.%X3 | | Bool | Status word 1 |
| #TG1.Receive.ZSW1.%X6 | | Bool | Status word 1 |
| #TG1.Receive.ZSW1.%X7 | | Bool | Status word 1 |

