Samkoon®

Samkoon Technology

PRO, MAX series bus-type PLC

Register Specification

Version 1.4

2024.2

www.samkoon-automation.com

Copyright Notice

Documentation version record

Version number	**Revision date**
Version 1.0	June 3, 2022
Version 1.1	March 17, 2023 Add GM register, EC register
Version 1.2	May 23, 2023 Add axle group velocity and acceleration mapping registers
Version 1.3	June 27, 2023 Add software-triggered power-off hold registers, shaft group speed-limiting radius Communication module status monitor register
Version 1.4	February 3, 2024 Add tool magazine register, G5x register, current G5x, G92 register

The PRO and MAX series bus-type PLC registers are divided into two categories: dedicated registers and general-purpose registers. The dedicated registers are mainly used for control and feedback of special functions. All registers are mapped to the corresponding area of Modbus. The upper computer can access all registers through communication, and the local PLC program can also access all registers. At the same time, the G-code module can also access the GHWord register. When programming, be careful to avoid register read/write conflicts. The registers and descriptions supported by the PRO and MAX series motion control platforms are as follows:

Register name	Size	Modbus start address	Explanation
0x(read&write bit)			
SMBit	1024bit	0x0	Added general-purpose bit registers
YBit	2048bit	0x1024	
Mbit	8192bit	0x3072	
CBit	256bit	0x11264	
TBit	256bit	0x11520	
SBit	1024bit	0x11776	
1x(read bit)			
XBit	2048bit	1x0	
4x(read&write word)			
GHWord	1736word	4x0	G-code interaction area #5700 ~ #6567
DMWord	10000word	4x1736	Added general-purpose data registers
DWord	8192word	4x11736	
VWord	8word	4x19928	
ZWord	8word	4x19936	
TVWord	256word	4x19944	
CVWord	312word	4x20200	
FXWord	2048word	4x20512	(System reserved)
FYWord	2048word	4x22560	(System reserved)
OPWord	12800word	4x24608	MC control axis dedicated registers
ECWord	592word	4x37408	(System retention)
GMWord	20000word	4x38000	General-purpose data registers can also correspond to the G-code interaction area #7000 to #16999
3x(read only word)			
EPWord	1600word	3x0	EtherCAT process data feedback area
EMWord	256word	3x1600	EtherCAT status feedback zone

Note: The red parts in the picture are system-specific registers

The meanings and uses of the special registers in the red section will be explained in sequence below:

1.Mbit word (8192*1bit) M special function bit register:

1.1 Single-axis MC control configuration and status information		
Bit register	**Read-write Cap (R/W)**	**Parameter description**
M_8150	R	Connect during the first scan cycle
M_8151	R	Connect when the program is running
M_8152	R	Connect when the program stops
M_8158	R	10ms clock pulse (1-high, 0-low)
M_8159	R	100ms clock pulse (1-high, 0-low)
M_8160	R	1sec clock pulse (1-high, 0-low)
M_8161	R	1-minute clock pulse (1-high, 0-low)
M_8162	R/W	Non-volatile retention area data is all cleared (1-clear, 0-none)
M_8163	R/W	Power-fail retention area data is all cleared (1-clear, 0-none)
M_8164	R/W	All output prohibited (1-disble, 0-enable)
M_8165	R	Bit move direction specified (1-left, 0-right)
M_8166	R	Shift instruction overflow bit (1-error, 0-ok)
M_8167	R	Shift instruction 0 flag (1-error, 0-OK)
M_8168	R	BCD code conversion error flag (1-error, 0-ok)
M_8169	R	Arithmetic Operation Overflow Flag (1-error, 0-ok)
M_8170	R	Arithmetic Negative Result Flag (1-error, 0-ok)
M_8171	R	Arithmetic Zero Result Flag (1-error, 0-OK)
M_8172	R	Arithmetic Divide-by-Zero Flag (1-error, 0-OK)
M_8173	R	Offset address out of bounds
M_8174	R	error status (1-error, 0-ok)
M_8175	R	Running status (1-run, 0-stop)
M_8179	R/W	Software-Triggered Data Saving on Power Loss, Auto-Clear
M_8180+n	R/W	COMn TX Complete Flag
M_8183+n	R/W	COMn RX Complete Flag

2. GHWord (1736*16bit) G code interaction area:

GHWord registers are the registers for data interaction between G code and the PLC ladder diagram, which can be read and written. The corresponding address areas are as shown in the following table:

3.1 GHWord G-code Register mapping area			
Name	Registers	Data type	Description
G-code mapping Register #5700	***GHWord[0]***	***float(32bit)***	**Mapping numeric parameter #5700**
G-code mapping register #5701	***GHWord[2]***	***float(32bit)***	**Mapping numeric parameter #5701**
G-code mapping register #5702	***GHWord[4]***	***float(32bit)***	**Mapping numeric parameter #5702**
G-code mapping Register #5703	***GHWord[6]***	***float(32bit)***	**Mapping numeric parameter #5703**
G-code mapping register #5704	***GHWord[8]***	***float(32bit)***	**Mapping numeric parameter #5704**
……	……	……	……
G-code mapping register #6567	***GHWord[1734]***	***float(32bit)***	**Mapping numeric parameter #6567**

2 OPWord word (12800* 16-bit) MC axis control area:

OPWord registers are mainly used for registers related to MC axis control, including configuration information for single-axis control, status information during operation, online debugging information, multi-axis interpolation status information, communication configuration

information, etc. The specific register area configuration is as follows:

2.1 Single-axis MC control configuration and status information					
Serial number	**class type**	**Parameter names**	**Read-Write Cap R/W**	**Register address (AxiOrd as the axis number)**	**Parameter description**
1	**FLOAT**	**Current target location**	**R**	**OPWord0+60*AxiOrd+0**	The value of this parameter specifies the current moving target position of the axis, and this value is updated each time motion trajectory planning is successfully performed
2	**FLOAT**	**Positive soft limit position**	**R/W**	**OPWord0+60*AxiOrd+2**	The value of this parameter is the positive soft limit value. If the command position is greater than this value, the command position will be automatically changed to the positive soft limit position.
3	**FLOAT**	**Negative soft limit position**	**R/W**	**OPWord0+60*AxiOrd+4**	This parameter value is a negative soft limit value. If the current position is less than this value, the instruction position will automatically changed to a positive soft limit position.
4	**BOOL**	**Positive soft limit enable switch**	**R/W**	**OPWord0+60*AxiOrd+6**	This value is a positive soft limit enable switch: “0” is off; The rest are on.
5	**BOOL**	**Negative soft limit enable switch**	**R/W**	**OPWord0+60*AxiOrd+7**	This value is a negative soft limit enable switch: “0” is off; The rest are on.
6	**BOOL**	**Position lag monitoring enabled**	**R/W**	**OPWord0+60*AxiOrd+8**	This value is the position lag monitoring enable switch: “0” is off; The rest are on.
7	**BOOL**	**Configured axis mode**	**R/W**	OPWord0+60*AxiOrd+9	When this value is 0, it indicates that the axis is in linear mode; when the value is 1, it indicates that the axis is in rotary mode.
8	**FLOAT**	**Maximum allowable position lag value**	**R/W**	**OPWord0+60*AxiOrd+10**	This parameter value is the maximum allowable position lag value. If the interpolation between the currently feedback position and the real-time required position is greater than this parameter

					value, an error will be reported.
9	**FLOAT**	**The maximum speed value allowed by the system**	**R**	**OPWord0+60*AxiOrd+12**	This parameter value represents the maximum operating speed allowed for trajectory planning of the current axis, and it is the upper limit of the "speed" item entered by the user in the command. If the input value is greater than this parameter value, the effect is equivalent to inputting this parameter value.
10	**FLOAT**	**The maximum speed of the axle Jog allowed by the system**	**R/W**	**OPWord0+60*AxiOrd+14**	This parameter value is the maximum JOG speed allowed on the current axis during jog movement.
11	**FLOAT**	**Current speed value**	**R**	**OPWord0+60*AxiOrd+16**	This parameter is the velocity value of the current axis, and the remote axis is determined by the number of feedback pulses.
12	**FLOAT**	**Current target velocity value**	**R**	**OPWord0+60*AxiOrd+18**	This parameter is the target velocity value of the current axis, determined by the result of curve planning.
13	**FLOAT**	**The maximum acceleration value allowed by the system**	**R**	**OPWord0+60*AxiOrd+20**	This parameter value represents the maximum allowable acceleration during trajectory planning for the current axis. It serves as the upper limit for the "acceleration" setting in user commands. If a user-entered value exceeds this parameter, the system will default to using this predefined maximum value instead.
14	**FLOAT**	**Axis Limit Deceleration**	**R/W**	**OPWord0+60*AxiOrd+22**	This parameter value represents the deceleration of the motion limit signal for the axis in this movement.
15	**FLOAT**	**The maximum deceleration value allowed**	**R**	**OPWord0+60*AxiOrd+24**	This parameter value represents the maximum allowable running acceleration for trajectory planning on the current axis, it

		by the system			serves as the upper limit of the "deceleration" item entered by the user in the instruction. If the input value is greater than that, the effect is the same as entering that value
16	**FLOAT**	**Shaft failure deceleration**	**R/W**	**OPWord0+60*AxiOrd+26**	The value of this parameter indicates the deceleration of faults that occur during the operation of the motion axis;
17	**FLOAT**	**The maximum acceleration value allowed by the system**	**R**	**OPWord0+60*AxiOrd+28**	This parameter value is the maximum running acceleration allowed for trajectory planning on the current axis.
18	**WORD**	**Gear ratio molecule**	**R/W**	**OPWord0+60*AxiOrd+30**	This parameter value indicates the value of the gear ratio molecule of the gear shifting device
19	**WORD**	**Gear score parent**	**R/W**	**OPWord0+60*AxiOrd+31**	The value of this parameter indicates the value of the gear ratio of the gear shifting device
20	**FLOAT**	**Current actual location**	**R**	**OPWord0+60*AxiOrd+32**	This parameter is the velocity value of the current axis, and the remote axis is determined by the number of feedback pulses
21	**FLOAT**	**Current target acceleration value**	**R**	**OPWord0+60*AxiOrd+34**	This parameter is the target acceleration value of the current axis, determined by the result of curve planning
22	**FLOAT**	**The current actual acceleration value**	**R**	**OPWord0+60*AxiOrd+36**	This parameter is the velocity value of the current axis, and the remote axis is determined by the number of feedback pulses
23	**FLOAT**	**The amount of movement of the worktable in one full rotation**	**R**	**OPWord0+60*AxiOrd+38**	The value of this parameter indicates the amount of movement of the workbench for one full rotation
24	**DWORD**	**The number of pulses the motor/encoder rotates one full circle**	**R**	**OPWord0+60*AxiOrd+40**	This parameter value indicates the number of pulses the motor/encoder takes to complete one rotation

25	**INT**	**Current axis state machine state**	**R**	**OPWord0+60*AxiOrd+42**	The state machine status of the current axis is defined as follows: ErrorStop=0x01 Disabled=0x02 StandStill=0x04 Stopping=0x08 Homing=0x10 ContMotion=0x20 SyncMotion=0x40 DiscMotion=0x80
26	**INT**	**A misalignment of the current motion axis**	**R**	**OPWord0+60*AxiOrd+43**	Internal error reporting for plc-open
27	**INT**	**Servo error of the current motion axis**	**R**	**OPWord0+60*AxiOrd+44**	Only for remote axes, servo error
28	**WORD**	**Control bit configuration register**	**R/W**	**OPWord0+60*AxiOrd+45**	**Bit-0**: The initial movement direction for home return in the axis home return settings; **Bit-1**: The edge signal direction for home stop in the axis home return settings; **Bit-2**: The movement direction before home stop in the axis home return settings; **Bit-3**: The inverse enable for unit conversion settings in the axis basic configuration; **Bit-4**: The enable for using speed change devices in the axis basic configuration.
29	**FLOAT**	**Axis Homing Speed**	**R/W**	**OPWord0+60*AxiOrd+46**	This parameter is the origin return velocity value of the current axis
30	**FLOAT**	**Axis Homing Creep Speed**	**R/W**	**OPWord0+60*AxiOrd+48**	This parameter is the value of the origin crawl speed of the current axis
31	**FLOAT**	**Axis Homing Acceleration**	**R/W**	**OPWord0+60*AxiOrd+50**	This parameter is the origin return acceleration value of the current axis
32	**DWORD**	**The number of pulses sent by the current axis command**	**R**	**OPWord0+60*AxiOrd+56**	This parameter is the instruction pulse count of the axis

33	**DWORD**	**The number of current axis feedback pulses**	**R**	**OPWord0+60*AxiOrd+58**	This parameter is the feedback pulse count of the axis

Additional note:

1. Register Address Allocation for Axis Control:

- Each axis corresponds to a contiguous block of register addresses for the interaction functions described above. As shown in the "Register Address" column of the table, each axis number corresponds to **60 OPWord registers**.

2. Write-Only Register Constraints:

- Registers with write permissions can only be modified successfully when the specified axis is in a stopped state. Therefore, users must transition the state machine to the **Disabled** or **StandStill** state before writing.

3. Direct Register Manipulation Guidelines:

- Users may directly read/write registers by specifying their addresses, but must adhere to the "Parameter Type" to avoid writing invalid values that could cause operational issues.
- Ensure parameter validity. For example, when modifying soft limits from `100~500` to `-200~0`, set the negative limit to `-200` first, then the positive limit to `0`. Writing the positive limit first would create an invalid state (negative limit > positive limit), causing the write to fail.
- Valid parameter writes persist after power cycling.

4. Parameter Access via Function Calls:

- Users may also read/write **FLOAT-type registers** using `MC_READLPARAMETER` and `MC_WRITEPARAMETER`, specifying the "Parameter Number" instead of manual register addresses.
- Similarly, **BOOL-type registers** are accessed via `MC_READBOOLPARAMETER` and `MC_WRITEBOOLPARAMTER`.
- **Note**: Even when using these functions, users must ensure writes occur during valid axis states and parameters are logically valid.

The following table shows the registers for the 'Online Debug' page, allowing users to debug without the host computer. All the functions on the Online Debug page can be achieved by reading or writing to the following registers.

2.2 Online Debugging Registers				
Parameter types	**Parameter name**	**Read -Writ**	**Register address (AxiOrd as the axis number)**	**Parameter description**

		e Cap R/W		
INT	**Control registers**	**R/W**	**OPWord3840+AxiOrd*40+0**	This register is a bit-composed word. When users read or write to this register, Please pay attention to the meanings of each bit as follows. The function can be analogous to the instructions after the '-' : Online debug enable (0x0001) Motion axis enable (0x0002)-MC_POWER Motion axis reset (0x0004)-MC_RESET Motion axis Stop (0x0008)-MC_STOP Set position (0x0010)-MC_SETPOSITION Enable Return to zero (0x0020)-M C_HOME Forward jog switch (0x0040)-MC_JOG Negative jog switch (0x0080)-MC_JOG Motion start switch (0x0100) Motion Stop switch (0x0200)-MC_HALT
INT	**Control Register 2**	**R/W**	**OPWord3840+AxiOrd*40+1**	Reserve
FLOAT	**Set location**	**R/W**	**OPWord3840+AxiOrd*40+2**	The effect is equivalent to the input value of MC_SETPOSITION
FLOAT	**Origin position**	**R/W**	**OPWord3840+AxiOrd*40+4**	The effect is equivalent to the input value of MC_HOME
FLOAT	**Forward point speed**	**R/W**	**OPWord3840+AxiOrd*40+6**	When the user performs forward jogging through the control register, the movement is carried out at the speed specified in this register value.
FLOAT	**Negative point movement velocity**	**R/W**	**OPWord3840+AxiOrd*40+8**	When the user performs reverse jogging via the control register, the axis moves at the speed specified by this register value.
INT	**Type of movement**	**R/W**	**OPWord3840+AxiOrd*40+10**	When the user enables motion start through the control register, the motion mode is determined by this register value: 0: Relative positioning 1: Absolute positioning Any other value: Error
INT	**Curve type**	**R/W**	**OPWord3840+AxiOrd*40+11**	When the user enables motion start through the control register, the motion profile type is determined by this register value: - `2`: S-curve profile - Any other value: Trapezoidal (T-curve) profile
FLOAT	**Input position**	**R/W**	**OPWord3840+AxiOrd*40+12**	When the user enables motion startup through the control register, performing with this

				register as the input position
FLOAT	**Input speed**	**R/W**	**OPWord3840+AxiOrd*40+14**	When the user enables motion startup through the control register, performing at the input speed of that register
FLOAT	**Input acceleration**	**R/W**	**OPWord3840+AxiOrd*40+16**	When the user enables motion startup through the control register, performing with the register as the input acceleration
FLOAT	**Input deceleration**	**R/W**	**OPWord3840+AxiOrd*40+18**	When the user enables motion to start through the control register, performing deceleration with that register as input
INT	**Output status**	**R**	**OPWord3840+AxiOrd*40+21**	This register is a bit-composed word, and the meaning of each bit is as follows: - Software positive limit flag (0x0020) - Software negative limit flag (0x0010) - Home signal (0x0008) - Negative hard limit flag (0x0004) - Positive hard limit flag (0x0002) - Axis in motion flag (0x0001)
INT	**Communication status**	**R**	**OPWord3840+AxiOrd*40+22**	Keep
\	\	\	**OPWord3840+AxiOrd*40+23 ~40**	Keep
Additional note: 1 The "Online Debug Enable" in the control register is like a master switch, and the online debug function can only be used when the flag bit is set, while all MC instructions in the ladder diagram will be inactivated; 2. Other flag bits in the control register will be reset once they are done processed; For forward and negative point switches, the user needs to manually hold the position to continue, and if the user stops setting, the point will stop 0.5 seconds after the last setting. 3 The motion start in the control register needs to be combined with "motion type", "curve type", "input position", "input speed", "input acceleration" and "input deceleration" to determine the motion trajectory;				

The following table shows the area of the axle group interpolation status register:

2.3 Interpolate the shaft group status register area				
Parameter types	**Parameter name**	**Read-Write Cap(R/W)**	**Register address (GroupOrd for group number, AxiOrd for axis number)**	**Parameter description**

WORD	**Current Axis Status of the Current Axis Group**	**R**	**OP10000+GroupOrd*90+ AxiOrd*10+0**	The state machine states of the current axis are defined as follows: ErrorStop=0x01 Disabled=0x02 StandStill=0x04 Stopping=0x08 Homing=0x10 ContMotion=0x20 SyncMotion=0x40 DiscMotion=0x80
FLOAT	**Current Set Position of the Current Axis in the Axis Group**	**R**	**OP10000+GroupOrd*90+ AxiOrd*10+2**	When this parameter serves as the target position value for the current axis, it is determined by the execution of interpolation commands.
FLOAT	**Actual Position of the Current Axis in the Current Axis Group**	**R**	**OP10000+GroupOrd*90+ AxiOrd*10+4**	This parameter is the actual position value of the current axis
FLOAT	**Set Speed of the Current Axis in the Current Axis Group**	**R**	**OP10000+GroupOrd*90+ AxiOrd*10+6**	When this parameter is the target speed value of the current axis, it is determined by the result of curve planning.
FLOAT	**Current Set Position of the Current Axis in the Axis Group**	**R**	**OP10000+GroupOrd*90+ AxiOrd*10+8**	This parameter is the actual speed value of the current axis.
WORD	**The running status of the current axis group**	**R**	**OP11000+GroupOrd*10+ 0**	When this parameter represents the operating status of the current axis group: - “1” indicates an alarm or abnormal status. - “0” indicates normal operating status.
WORD	**Cache Segment Count for Interpolation Motion of the Current Axis Group**	**R**	**OP11000+GroupOrd*10+ 1**	When this parameter refers to the cache segment count for interpolation motion of the current axis group, it is used to indicate the number of cache segments that have been successfully added to the queue.
WORD	**Interpolation Motion Operation Flag of the Current Axis Group**	**R**	**OP11000+GroupOrd*10+ 2**	When this parameter serves as the operation flag for interpolation motion of the current axis group: - “1” indicates completion. - “0” indicates ongoing operation.

FLOAT	**Set the speed of the current axis group**	**R**	**OP11000+GroupOrd*10+4**	This parameter is the set synthetic speed of the current axis group.
FLOAT	**The actual speed of the current axis group**	**R**	**OP11000+GroupOrd*10+6**	This parameter is the actual synthesis speed of the current axis group
FLOAT	**The remaining supply of the current axis group**	**R**	**OP11000+GroupOrd*10+8**	This parameter is the remaining feed amount of the current instruction segment of the current axis group.
FLOAT	**Maximum speed of the current axle group**	**R/W**	**OP11100+GroupOrd*4+0**	This parameter is the maximum speed set for the current axle group
FLOAT	**Maximum acceleration of the current axis group**	**R/W**	**OP11100+GroupOrd*4+2**	This parameter is the maximum acceleration set for the current axis group
FLOAT	**Arc Speed Limit Radius of the Current Axis Group**	**R/W**	**OP11200+GroupOrd*2+0**	This parameter indicates the arc speed-limiting radius set for the axle group
Additional note: 1. The interpolation motion of the axis group can cache up to 2000 motion positions, please attention to Buffer Full Control. 2. The MC instruction block can be added to the interpolation instruction cache segment by changing the target position in the rising edge or enabled state.				

The following table shows the communication configuration parameters:

2.4 PLC_OPEN serial/network port debug registers				
Parameter types	**Parameter name**	**Read-Write Cap (R/W)**	**Register address Com serial port number (two serial**	Parameter Description

			ports 0 and 1)	
WORD	**Serial Enable**	**R/W**	**OPWord9000+Com*10+0**	Serial port Enable
WORD	**Baud rate**	**R/W**	**OPWord9000+Com*10+1**	The parameter setting for the current serial port baud rate.
WORD	**Parity Check**	**R/W**	**OPWord9000+Com*10+2**	Current Serial Port Parity Setting
WORD	**Data bit**	**R/W**	**OPWord9000+Com*10+3**	Current Serial Port Data Bit Parameter Setting
WORD	**Stop position**	**R/W**	**OPWord9000+Com*10+4**	Current Serial Port Stop Bit Parameter Setting
WORD	**Transmission mode**	**R/W**	**OPWord9000+Com*10+5**	Current Serial Port Transmission Mode Parameter Setting
WORD	**Transmission Protocol**	**R/W**	**OPWord9000+Com*10+6**	Current serial transport protocol parameter Settings
WORD	**Timeout**	**R/W**	**OPWord9000+Com*10+7**	Timeout parameter Settings during current serial port transmission
WORD	**Retransmission Count**	**R/W**	**OPWord9000+Com*10+8**	Set the parameter for the number of retransmissions during the current serial port transmission
WORD	**Slave site address**	**R/W**	**OPWord9000+Com*10+9**	Current serial slave address parameter Settings
WORD	**IPaddressA**	**R/W**	**OPWord9020+0**	Network port IP address A
WORD	**IPaddressB**	**R/W**	**OPWord9020+1**	Network port IP address B
WORD	**IPaddressC**	**R/W**	**OPWord9020+2**	Network port IP address C
WORD	**IPaddressD**	**R/W**	**OPWord9020+3**	Network port IP address D
WORD	**IPmaskA**	**R/W**	**OPWord9020+4**	Network port subnet mask A
WORD	**IPmaskB**	**R/W**	**OPWord9020+5**	Network port subnet mask B
WORD	**IPmaskC**	**R/W**	**OPWord9020+6**	Network port subnet mask C
WORD	**IPmaskD**	**R/W**	**OPWord9020+7**	Network port subnet mask D
WORD	**Retransmission Count**	**R/W**	**OPWord9020+8**	Setting parameters for the number of retransmissions during network port transmission
WORD	**Timeout**	**R/W**	**OPWord9020+9**	Timeout parameter Settings during network port transmission

The following table shows the tool magazine configuration parameters:

2.5 Tool magazine read/write Register				
Parameter types	**Parameter name**	**Read-Write Cap (R/W)**	**Register address**	Parameter description

WORD	**Read enable**	**R/W**	**OPWord9030+0**	Tool Magazine Number Reading Enable with Automatic Reset after Completion
WORD	**Axis group number**	**R/W**	**OPWord9030+1**	Axis Group Number for Reading
WORD	**Tool magazine Number**	**R/W**	**OPWord9030+2**	Tool magazine Number for number
WORD	**Return - Tool number**	**R/W**	**OPWord9030+3**	Return - Tool number
FLOAT	**Return - X-axis offset**	**R/W**	**OPWord9030+4**	Return - X-axis offset
FLOAT	**Return - Y-axis offset**	**R/W**	**OPWord9030+6**	Return - Y-axis offset
FLOAT	**Return - Z-axis offset**	**R/W**	**OPWord9030+8**	Return - Z-axis offset
FLOAT	**Return - A-axis offset**	**R/W**	**OPWord9030+10**	Return - A-axis offset
FLOAT	**Return - B-axis offset**	**R/W**	**OPWord9030+12**	Return - B-axis offset
FLOAT	**Return - C-axis offset**	**R/W**	**OPWord9030+14**	Return - C-axis offset
FLOAT	**Return - U-axis offset**	**R/W**	**OPWord9030+16**	Return - U-axis offset
FLOAT	**Return - V-axis offset**	**R/W**	**OPWord9030+18**	Return - V-axis offset
FLOAT	**Return - W-axis offset**	**R/W**	**OPWord9030+20**	Return - W-axis offset
FLOAT	**Return - Tool diameter**	**R/W**	**OPWord9030+22**	Return - Tool diameter
FLOAT	**Return - Tool rake Angle**	**R/W**	**OPWord9030+24**	Return - Tool rake Angle
FLOAT	**Return - Tool relief Angle**	**R/W**	**OPWord9030+26**	Return - Tool relief Angle
WORD	**Return - Tool rotation direction**	**R/W**	**OPWord9030+28**	Return - Tool rotation direction
WORD	**Return - the finish bit**	**R/W**	**OPWord9030+29**	Read Completion Flag
WORD	**Write enable**	**R/W**	**OPWord9060+0**	Tool Magazine Number Writing Enable with Automatic Reset after Completion
WORD	**Axis group number**	**R/W**	**OPWord9060+1**	Axis group number

WORD	**Tool magazine Number**	**R/W**	**OPWord9060+2**	Tool Magazine Number
WORD	**Tool number**	**R/W**	**OPWord9060+3**	Tool number
FLOAT	**X-axis offset**	**R/W**	**OPWord9060+4**	X-axis offset
FLOAT	**Y-axis offset**	**R/W**	**OPWord9060+6**	Y-axis offset
FLOAT	**Z-axis offset**	**R/W**	**OPWord9060+8**	Z-axis offset
FLOAT	**A-axis offset**	**R/W**	**OPWord9060+10**	A-axis offset
FLOAT	**B-axis offset**	**R/W**	**OPWord9060+12**	B-axis offset
FLOAT	**C-axis offset**	**R/W**	**OPWord9060+14**	C-axis offset
FLOAT	**U-axis offset**	**R/W**	**OPWord9060+16**	U-axis offset
FLOAT	**V-axis offset**	**R/W**	**OPWord9060+18**	V-axis offset
FLOAT	**W-axis offset**	**R/W**	**OPWord9060+20**	W-axis offset
FLOAT	**Tool diameter**	**R/W**	**OPWord9060+22**	Tool diameter
FLOAT	**Tool rake Angle**	**R/W**	**OPWord9060+24**	Tool rake Angle
FLOAT	**Tool relief Angle**	**R/W**	**OPWord9060+26**	Tool rear corner
WORD	**Tool rotation direction**	**R/W**	**OPWord9060+28**	Tool rotation direction
WORD	**Return - the finish bit**	**R/W**	**OPWord9060+29**	Write Completion Flag

The following table shows the configuration parameters of the G54 coordinate system:

2.6 G54 coordinate system configuration registers				
Parameter types	**Parameter name**	**Read-Write Cap (R/W)**	**Register address**	Parameter description

WORD	**Write enable**	**R/W**	**OPWord8000+0**	Offset Value Writing Enable with Automatic Reset after Completion
WORD	**Axis group number**	**R/W**	**OPWord8000+1**	Write the axis group number of the operation
WORD	**Coordinate system offset index**	**R/W**	**OPWord8000+2**	Values 1-9 correspond to G54-G59.3
WORD	**Write Success Return Value**	**R/W**	**OPWord8000+3**	Return 1 after successful writing
FLOAT	**Write input - X-axis offset**	**R/W**	**OPWord8000+4**	Write input - X-axis offset
FLOAT	**Write input - Y-axis offset**	**R/W**	**OPWord8000+6**	Write input - Y-axis offset
FLOAT	**Write input - Z-axis offset**	**R/W**	**OPWord8000+8**	Write input - Z-axis offset
FLOAT	**Write input - A-axis offset**	**R/W**	**OPWord8000+10**	Write input - A-axis offset
FLOAT	**Write input - B-axis offset**	**R/W**	**OPWord8000+12**	Write input - B-axis offset
FLOAT	**Write input - C-axis offset**	**R/W**	**OPWord8000+14**	Write input - C-axis offset
FLOAT	**Write input - U-axis offset**	**R/W**	**OPWord8000+16**	Write input - U-axis offset
FLOAT	**Write input - V-axis offset**	**R/W**	**OPWord8000+18**	Write input - V-axis offset
FLOAT	**Write input - W-axis offset**	**R/W**	**OPWord8000+20**	Write input - W-axis offset
WORD	**Manually select the G54 coordinate system**	**R/W**	**OPWord8000+30**	G54 coordinate system manually select 1-9
WORD	**Axis group number**	**R/W**	**OPWord8000+31**	

The following table shows the current valid coordinates of G5x and G92 during the current operation:

2.7 Current active registers of G5x and G92

Parameter type	Parameter name	Read-Write Cap (R/W)	Register address	Parameter description
WORD	**G5x enable index**	**R/W**	**OPWord8100+GroupOrd*40+0**	Current G5x index
FLOAT	**G5x_x-axis offset**	**R/W**	**OPWord8100+GroupOrd*40+2**	G5x X-axis offset
FLOAT	**G5x_y-axis offset**	**R/W**	**OPWord8100+GroupOrd*40+4**	G5x Y-axis offset
FLOAT	**G5x_z-axis offset**	**R/W**	**OPWord8100+GroupOrd*40+6**	G5x Z-axis offset
FLOAT	**G5x_A axis offset**	**R/W**	**OPWord8100+GroupOrd*40+8**	G5x A-axis offset
FLOAT	**G5x_B axis offset**	**R/W**	**OPWord8100+GroupOrd*40+10**	G5x B-axis offset
FLOAT	**G5x_C axis offset**	**R/W**	**OPWord8100+GroupOrd*40+12**	G5x C-axis offset
FLOAT	**G5x_U axis offset**	**R/W**	**OPWord8100+GroupOrd*40+14**	G5x U-axis offset
FLOAT	**G5x_V axis offset**	**R/W**	**OPWord8100+GroupOrd*40+16**	G5x V-axis offset
FLOAT	**G5x_W axis offset**	**R/W**	**OPWord8100+GroupOrd*40+18**	G5x W axis offset
FLOAT	**G92_x-axis offset**	**R/W**	**OPWord8100+GroupOrd*40+20**	G92 X-axis offset
FLOAT	**G92_y-axis offset**	**R/W**	**OPWord8100+GroupOrd*40+22**	G92 Y-axis offset
FLOAT	**G92_z-axis offset**	**R/W**	**OPWord8100+GroupOrd*40+24**	G92 Z-axis offset
FLOAT	**G92_A axis offset**	**R/W**	**OPWord8100+GroupOrd*40+26**	G92 A axis offset
FLOAT	**G92_B axis offset**	**R/W**	**OPWord8100+GroupOrd*40+28**	G92 B-axis offset
FLOAT	**G92_c-axis offset**	**R/W**	**OPWord8100+GroupOrd*40+30**	G92 C-axis offset
FLOAT	**G92_U axis offset**	**R/W**	**OPWord8100+GroupOrd*40+32**	G92 U-axis offset
FLOAT	**G92_v-axis offset**	**R/W**	**OPWord8100+GroupOrd*40+34**	G92 V-axis offset
FLOAT	**G92_W axis offset**	**R/W**	**OPWord8100+GroupOrd*40+36**	G92 W axis offset

3.EPWord (1600* 16-bit) EtherCAT Process Data Feedback Area:

Map the data directly to this memory area in sequence according to the PDO configuration, arranged in byte order

4. EMWord (256*16bit) EtherCAT bus, extension module state feedback area:

4.1EMWord EtherCAT bus status feedback area			
Name	Corresponding address	Data type	Description
EtherCAT master station link status	***EMWord[0]*** ***3x1600***	***ushort(16bit)***	**Registers indicate EtherCAT link status:** **0x00: EtherCAT device unconnected** **0x01: EtherCAT device linked**
Frame loss count	***EMWord[1]*** ***3x1601***	***ushort(16bit)***	**Registers represent the EtherCAT frame loss count**
Frame error count	***EMWord[2]*** ***3x1602***	***ushort(16bit)***	**Registers represent the EtherCAT frame error count**
Master station state machine	***EMWord[3]*** ***3x1603***	***ushort(16bit)***	**Registers represent the master station state machine:** **0x01: Init** **0x02: Pre-OP** **0x04: Safe-OP** **0x08: OP**
Number of slave stations	***EMWord[4]*** ***3x1604***	***ushort(16bit)***	**Register indicating the configured number of slaves stations**
Slave 0 status	***EMWord[5]*** ***3x1605***	***ushort(16bit)***	**Register indicating the state machine of Slave 0:** **0x01: Init** **0x02: Pre-OP** **0x04: Safe-OP** **0x08: OP**
State from Station 1	***EMWord[6]*** ***3x1606***	***ushort(16bit)***	**Register indicating the state machine of Slave 1:** **0x01: Init** **0x02: Pre-OP** **0x04: Safe-OP** **0x08: OP**
……	……	……	……
PDO0 byte offset in process data	***EMWord[100]*** ***3x1700***	***ushort(16bit)***	**PDO0 byte offset**
PDO0 bit offset in process data	***EMWord[101]*** ***3x1701***	***ushort(16bit)***	**PDO0 bit offset, valid when PDO0 is of bit attribute**
PDO1 byte offset in process data	***EMWord[102]***	***ushort(16bit)***	**PDO1 byte offset**

	3x1702		
PDO1 bit offset in process data	***EMWord[103]*** *3x1703*	***ushort(16bit)***	**PDO1 bit offset, valid when PDO1 is of bit attribute**
……	……	……	……

4.2EMWord extension module State feedback area			
Name	Corresponding address	Data type	Description
Quantity	***EMWord[370]*** *3x1970*	***ushort(16bit)***	**Registers indicate the actual number of extension module connections**
Communication successful	***EMWord[371]*** *3x1971*	***ushort(16bit)***	**Registers indicate the communication success status which represents the configuration information is consistent with the actual connection information**
Connection error	***EMWord[372]*** *3x1972*	***ushort(16bit)***	**Registers indicate communication connection errors**
Host connection status	***EMWord[373]*** *3x1973*	***ushort(16bit)***	**Registers indicate the connection status of the host**
Communication status	***EMWord[374]*** *3x1974*	***ushort(16bit)***	**Registers represent communication status, including scan, configuration, normal communication processes**
CRC error count	***EMWord[375]*** *3x1975*	***ushort(16bit)***	**The register represents the count of communication CRC errors**
WC error frame count	***EMWord[376]*** *3x1976*	***ushort(16bit)***	**Registers represent WC frame error counts**
Frame loss count	***EMWord[377]*** *3x1977*	***ushort(16bit)***	**Registers represent the number of WC frame loss**
DCM Cycle difference	***EMWord[378]*** *3x1978*	***ushort(16bit)***	**Registers represent the DCM single cycle difference**
Cumulative deviation of DCM	***EMWord[379]*** *3x1979*	***ushort(16bit)***	**Register representation of DCM accumulated clock deviation**
DCM adjusts value	***EMWord[380]*** *3x1980*	***ushort(16bit)***	**Registers represent DCM adjustment values**
Year of the underlying version	***EMWord[381]*** *3x1981*	***ushort(16bit)***	**Registers represent the year of firmware**
Month of the underlying version	***EMWord[382]***	***ushort(16bit)***	**Registers represent the firmware**

	3x1982		**month**
Underlying version date	*EMWord[383]* *3x1983*	***ushort(16bit)***	**Registers represent firmware dates**
Actual connected extension module type 1	*EMWord[384]* *3x1984*	***ushort(16bit)***	**Registers indicate the type of submodule 1:** **0x40: PRO-E-16X** **0x44: PRO-E-16T** **0x48: PRO-E-8X8T** **0x4C: PRO-E-16X16T** **0x80: PRO-E-4AI** **0x84: PRO-E-4AO**
Actual connected extension module type 2	*EMWord[385]* *3x1985*	***ushort(16bit)***	**Registers indicate the type of submodule 2:** **0x40: PRO-E-16X** **0x44: PRO-E-16T** **0x48: PRO-E-8X8T** **0x4C: PRO-E-16X16T** **0x80: PRO-E-4AI** **0x84: PRO-E-4AO**
……	……	……	……