

SMC Command	SMC Command Description	IEC-61131 / PLCopen equivalent	Note
#AUTO	Start Program upon power up	Built in, no need to reproduce. IEC Pro application can execute a Warm or Cold start SYSTEM task upon power up to initialize data.	
#CMDERR	Software interrupt to handle command errors in program	This is handled a variety of different ways. PLCopen FBs give ErrorID, MC_ReadAxisError, Y_ReadAlarms.	
#COMINT	Software interrupt when a character was received via serial port (dumb terminal)	No equivalent, use CYCLIC driver updates of Modbus or Ethernet/IP	
#ININT	Input Interrupt	No equivalent EVENT handling supported. Use a fast CYCLIC task and monitor an input for change of state.	SMC3010 would require about 2 mSec to react to input change
#LIMSWI	Software interrupt when a limit switch is tripped	Built in via PLCopen ErrorID and auto created Global variables. (POT, NOT)	
#MCTIME	Software interrupt if motor did not get into position within certain time of profiler completing motion	No equivalent, use TON function to determine if motion has not been completed within a specified timeframe. Use the P_SET global variable in conjunction with the PLCopen FB Done output as shown in the help file for MC_MoveAbsolute.	
#POSERR	software interrupt when excessive position error	Closest equivalent function is to set amplifier Pn 520 to the maximum permissible following error, which will cause A.DO alarm when the value is exceeded.	
&	and	SAME	
()	parenthesis	SAME	
*	multiply	SAME	
/	divide	SAME	
@ABS	absolute value function	SAME	
@ACOS	arc cosine function	SAME	
@AN	read analog input function	Use automatically created global variable	
@ASIN	arc sine function	SAME	
@ATAN	arc tangent function	SAME	
@COM	two's complement function		
@COS	cosine function	SAME	
@FRAC	Return only the fractional proportional of a value	Make your own function or use Yaskawa Toolbox 'REM'	
@IN	Read digital input	Use Ladder Contact or variable	
@INT	Truncate a value to nearest integer value	Use DataType conversion functions	
@OUT	Read status of output value	Use ladder coil or variable	
@RND	Round a number to the nearest integer value	Use DataType conversion functions	
@SIN		SAME	
@SQR		SAME	
@TAN		SAME	
	OR	SAME	
+		SAME	
<R<S	reset back to factory defaults	Use web server to delete all user files	
<R<V	obtain firmware version number	Use web server	
AB	Abort Motion	Disable servo by dropping MC_Power.Enable	
AC	Acceleration	PLCopen motion function blocks have a acceleration input	
AD	Trippoint that waits until the motor has gone past a specified distance from the beginning of a move.	Use MC_ReadParameter and R_TRIG to determine event. There is no parameter that indicates the motion travelled relative to the start of a move. Use math logic to determine distance.	

SMC Command	SMC Command Description	IEC-61131 / PLCopen equivalent	Note
AF	Set Analog feedback mode	No firmware level equivalent - use PID Control from Yaskawa Toolbox and Y_DirectControl from YMotion firmware library	
AI	Trippoint that waits for an input to change state	Inherent with ladder contacts and R_TRIG or F_TRIG functions	
AL	Arm Latch to capture an encoder position	Use PLCopen function called MC_TouchProbe	
AM	Trippoint that waits until the motion profile is complete	Inherent with PLCopen motion function blocks "Done" output.	
AO	Set analog output	Use automatically created global variable	
AP	Trippoint that waits until the motor crosses an absolute position	Use MC_ReadParameter and R_TRIG to determine event	
AR	Trippoint that waits for the motor to move a relative distance from the start of the move	There is no direct equivalent. Use MC_ReadParameter to read actual or commanded position and do your own math to determine when the appropriate distance has been achieved.	
AS	Trippoint that waits until the motor is at the specified speed	If using MC_MoveVelocity, use the AtVelocity output, otherwise use MC_ReadParameter to obtain the commanded velocity and compare to the function blocks Velocity input.	
AT	Trippoint that waits for time	Use TON function, or use global variable PLC_SYS_TICK_CNT that counts ticks since power up, or use Real Time clock functions from Yaskawa Toolbox	
BG	Begin Motion	Most PLCopen function blocks start motion upon the rising edge of the Execute input	
BN	Burn parameters into flash memory	Use Hardware Configuration to save XML files to the controller. This is not the same as BN, which mainly saved motion parameters, the most equivalent of which would be the PLCopen function block input values in many cases.	
BP	Burn program into flash memory	Set the checkbox to download the bootproject when downloading the application program.	
BV	Burn Variables to flash memory	Set the checkbox for retain variables. The data will be stored in battery backed memory	
CB	Clear output bit	Use Ladder Coil or set the BOOL variables associated with a physical output to FALSE.	
CD	Contour Data (Not applicable in SMC3010, meant for multiple axes, similar to a time based virtual master for camming)	Use time based virtual master and a cam profile	
CE	Configure Encoder (Quadrature, Pulse & Direction)	Use the settings in the Hardwar Configuration	
CF	Configure default port to send messages	No equivalent	
CH	Connect to Handle	No Equivalent. In version 2.1 coming in September 2011, there will be an additional firmware library to support TCP socket connections to replicate the functionality.	
CM	Contour Mode	No Equivalent. Use the camming feature with a virtual master.	
CN	Configure limit, home, and latch switches	Use Servopack Pn settings	
CS	Clear Sequence (for interpolation moves)	No equivalent, interpolation not applicable on MP2600iec	
DA	De allocate variable or array	Just delete from variable worksheet	Existed on SMC because it was an interpreted language, not compiled
DC	Deceleration	PLCopen function blocks have a Deceleration input	
DE	Read Dual / External Encoder position	Use PLCopen function block MC_ReadParameter	

SMC Command	SMC Command Description	IEC-61131 / PLCopen equivalent	Note
DL	Download Program	Use Resource Dialog box, or Patch POU	
DM	Define Array	Use DataType worksheet	
DP	Define Position	Use PLCopen function block MC_SetPosition	
DT	Delta Time (for contour mode)	Create a cam profile with a virtual master using time as master units.	
DV	Enable Dual Velocity Loops (Dual encoder PID mode)	No built in equivalent, use PID Control FB from Yaskawa Toolbox, Y_DirectControl from Motion and create control loop in application code	SMC could update this at 250uSec, MP2600iec can only do it at 1000uSec
EA	Specify Cam Master	Use PLCopen function block Y_CamIn	
EB	Enable Cam Mode	Use PLCopen function block Y_CamIn	
EC	Returns the current index into the cam table	Support provided through Cam Toolbox functions	
ED	Edit Mode	No equivalent	Existed on SMC because it was an interpreted language, not compiled
EG	Engage Cam Slave to Master	Use PLCopen function block Y_CamIn	
ELSE	For Logical statements	SAME	
EM	Cam cycle for camming	Automatically determined by firmware by looking at last master and slave position in the cam table	
EN	End a subroutine or Program	Built in part of IEC61131 programming (POU, FB layout)	Existed in SMC as part of the interpreted scripting language
ENDIF	For Logical statements	END_IF	
EO	Set communication Echo state	No equivalent	No need on Ethernet.
EP	Set camming interval (resolution)	No equivalent, but not needed because camming data files can contain irregular segment length from point to point.	In SMC, you could not specify the master data that correlated to the slave, it was required to be at regular intervals.
EQ	Disengage Camming	Use PLCopen function block Y_CamOut	
ER	Set maximum following error limit	Set Pn520 in servopack	Set in servopack reference units. Must convert from User Units to Reference units manually.
ET	Define cam data points	Use a Y_MS_CAM_STRUCT DataType, Use the Cam Toolbox (CamGenerator) function block	
FA	Set Acceleration Feedforward value	No equivalent	
FE	Find the Edge of the Home input	No Equivalent because MC_Step is not implemented. Closest equivalent is MC_StepLimitSwitch, otherwise, use a Jog function until the home input is detected.	
FI	Find the C channel on the motor	MC_StepRefPulse	
FL	Set / Read the Forward Software limit	Use MC_Read / Write parameter 1201	
FV	Set / Read the Velocity Feed forward value	No equivalent, possibly use the friction compensation Pn in the ServoPack	
GA	Set the master for gearing	Use PLCopen function block MC_GearIn	
GR	Set the gear ratio	Use PLCopen function block MC_GearIn	
HM	Home to the home input and C channel	No equivalent because MC_Step not implemented. Closest equivalent is to use the Home_LS_Pulse function block in the PLCopen Toolbox	
HX	Halt Execution	No Equivalent	Main focus of IEC-61131 implementation is CYCLIC tasks which are never stopped by the application itself
IA	Set / Read the controller IP address	No Application code equivalent, however the IP address can be set / read in MotionWorks IEC, Hardware Configuration, and the Web Server	
IF	IF (For logical statements)	IF	

SMC Command	SMC Command Description	IEC-61131 / PLCopen equivalent	Note
IH	Set / Read Internet Handle, for configuring slave devices	Use the Hardware Configuration	SMC had no concept of Hardware Configuration as a separate entity, all configuration and programming was done through command interface.
II	Configure Input Interrupt	No Equivalent	Future versions of MotionWorks IEC will include support for EVENT tasks, however due to the execution differences between the SMC and the IEC platform, it is possible to react to an input at scan rates as fast as 1 mSec, which is basically as fast as the SMC interrupt would react.
IL	PID Integrator Limit	No Equivalent. Use ServoPack tuning parameters. Pn101 is the closest.	Use Servopack Pn 102
IN	Get input string from serial port	No Equivalent	
IP	Increment Position	Use PLCopen function MC_MoveSuperImposed.	
IT	Set / Read Integration Time Constant (S Curve)	Use MC_Read / Write Parameter 1300 & 1301	
JG	Set the Jog speed	Use PLCopen function MC_MoveVelocity. Use the Jog function block provided in the PLCopen Toolbox.	
JP	Jump to a program location	Although the IEC language supports jumping to labels, it is not recommended as other methods are available which make programming logic easier to follow.	
JS	Jump to Subroutine	IEC61131 does not support subroutines, but it supports the creation of functions and function blocks, which provide the same functionality and more.	
KD	Set / Read the Derivative gain of the PID loop	No Equivalent	Position loop tuning exists in the Servopack.
KI	Set / Read the Integral gain of the PID loop	No Equivalent	Position loop tuning exists in the Servopack.
KP	Set / Read the Proportional gain of the PID loop	No Equivalent	Position loop tuning exists in the Servopack.
LA	List Arrays defined in the controller	Read the DataType worksheets	Existed in SMC as part of the interpreted scripting language
LE	Linear Interpolation Segment End	No Equivalent	
_LF	Read the status of the forward limit switch	Use the Global variables that are auto created by the Hardware Configuration for each servo axis.	
LI	Linear Interpolation		
LL	List Labels in the program	No Equivalent, refer to the Project Tree Window	Existed in SMC as part of the interpreted scripting language
LM	Define axes to be used in Linear Interpolation mode	No Equivalent, look at using the PathGenerator and MovePath from the Gantry Toolbox.	Use the PathGenerator and MovePath functions in the gantry Toolbox.
LO	LockOut access to controller application program	In MotionWorks IEC Pro, use the "Enter Password" feature on the File menu.	
_LR	Read the status of the reverse limit switch	Use the Global variables that are auto created by the Hardware Configuration for each servo axis.	
LS	List Program	No Equivalent. Completely different programming methodology.	
LV	List Variables	No Equivalent. Refer to the Variables grids (Global and local POU)	
LZ	Set / Read configuration for whether leading zeros are displayed	No Equivalent	SMC had no concept of Hardware Configuration as a separate entity, all configuration and programming was done through command interface.

SMC Command	SMC Command Description	IEC-61131 / PLCopen equivalent	Note
MB	Write a Modbus command	Modbus functions are synchronized with any CYCLIC task by setup in the Hardware Configuration	SMC only send out or read Modbus data when the command executed, which was only as often as the script application could get back to it. No guarantee on interval
MC	Trip point that indicates when motion is complete. (Encoder cross the final commanded position.)	Use PLCopen Done output in conjunction with the P_CON Global variable auto created for each Servopack.	
MF	Trippoint that indicates when the actual motor encoder moves past a specific position in the forward direction	Use MC_ReadActualPosition in conjunction with an R_TRIG to flag the event	
MG	Send a message out to one of the serial or Ethernet ports	no equivalent	
MO	Turn the motor off (disable bus power)	Use PLCopen function MC_Power	
MR	Trippoint that indicates when the actual motor encoder moves past a specific position in the reverse direction	Use MC_ReadActualPosition in conjunction with an R_TRIG to flag the event	
MT	Set / Read Motor Type	No Equivalent	SMC had no concept of Hardware Configuration as a separate entity, all configuration and programming was done through command interface.
NA	Set / Read the number of configured axes	No Equivalent	The was for the SMC3010s distributed control mode
NB	Set / Read the Notch Filter bandwidth	Sigma 5 has support for this	
NF	Set / Read Notch Filter	Sigma 5 has support for this	
NO	No Operation	Use IEC 61131 Comment	
NZ	Set / Read Notch Zero	Sigma 5 has support for this	
OB	Set an output bit when a condition is true	Use either ladder logic or ST Boolean logic	
OC	Set High Speed output compare function	Use Parameters ...	
OE	Off-On-Error (Disable the servo when there is excessive following error)	If Pn520 is set appropriately, the ServoPack will automatically shut down and provide the A.D0 alarm.	
OF	Offset	No Equivalent.	This was intended as an offset to counter against gravity. If the OF command was used to set a specific voltage for open loop torque with the PID gains set to zero, then consider using the Y_DirectControl function block in the YMotion firmware library.
OP	Set Output Bank	Use a global variable as a WORD type rather than a BOOL type and set accordingly	
PA	Position Absolute	Use PLCopen function MC_MoveAbsolute	
PF	Position Format	Use the Debug Dialog box to change the on screen display format	
PL	Pole	Use the ServoPack notch filter Pns	
PR	Position Relative	Use PLCopen function block MC_MoveRelative	
QD	Array Download	Use the Resource dialog box to send a file to an MP2300iec controller, or use the hardware Configuration's Online menu to send a file. Or use an HTTP post function from PC software to send a data file to the controller.	
QR	Data Record	Use the Logic Analyzer, or Sigma Win Data Trace	
QU	Array Upload	Use the Project Archive function in the Webserver to obtain any data files written by the	
QW	Sets the update rate for data between multiple SMC3010 controllers (slave to Master)	No Equivalent. Could use EIP or Modbus to transfer data between controllers, or use an MP2300Siec to put multiple axes on Mechatrolink. Network Variables feature for controller to controller communications coming in 2012.	
QZ	Notch Filter Zero	Use the Sigma-5 / Sigma Win tuning features	
RA	Record Array	Use the Logic Analyzer, or Sigma Win Data Trace	

SMC Command	SMC Command Description	IEC-61131 / PLCopen equivalent	Note
RC	Record Array	Use the Logic Analyzer, or Sigma Win Data Trace	
RD	Record Array	Use the Logic Analyzer, or Sigma Win Data Trace	
RE	Return from Error Subroutine	No Equivalent, because no concept of automatic error handling routines, but the IEC61131 supports the RETURN statement to exit a POU before completing. Not really an equivalent feature	
RI	Return from Interrupt	No Equivalent	
RL	Report the Latch position	Use PLCopen function MC_TouchProbe.RecordedPosition output, or use MC_ReadParameter to obtain either the cyclic or noncyclical latch value, parameters 1131	
RP	Report the commanded position	Use MC_ReadParameter 1010, 1015, or 1016	
RS	Reset the controller (reboot)	Use Web server Reboot feature after login in. Use the Reboot feature in the Hardware Configuration	
SA	Send a TCP string message to another device	Not currently supported. Basic TCP socket connection functions scheduled for the 2.1 release in late September 2011	
SB	Set an output bit On	Use ladder logic coil or set the global variables associated with the hardware to TRUE	
SC	Read the stop code status flags	Use PLCopen function block MC_ReadStatus which as similar status flags	
SH	Servo Enable	Use PLCopen function block MC_Power	
SP	Set / Read motor commanded speed	Use PLCopen motion function blocks Velocity input, and MC_ReadParameter 1011 to read	
SR			
ST	Stop motor	Use PLCopen function block MC_Stop	
TB	Tell Status Byte	No equivalent	No need for the exact same kind of status reported here
TC	Tell Error Code of last program fault	Use PLCopen FB ErrorID, MC_ReadAxisError, and Y_ReadAlarm. Also use Web server Alarm History	
TD	Tell the position of the Dual Encoder	Use MC_ReadParameter to read 1000, or 1005, or 1006	
TE	Tell the following Error	Use MC_ReadParameter to read 1130	
TI	Tell the status of the Digital inputs	Use a Contact or Global variable name associated with the input driver of the appropriate hardware	
TIME	Reports the number of milliseconds since the power was turned on.	Use system variable PLC_SYS_TICK_CNT or Real Time Clock functions	
TL	Set / Read the torque limit	Use MC_Read/Write Parameter for values ??	
TM	Set / Read the servo position loop update rate	Use the hardware Configuration to set / read the Mechatrolink cycle or dual pot ram update on MP2600iec.	This is the update for position target, but the ServoPack will update the position tuning loop at 250uSec
TN	Set Tangent axis used with interpolation	No Equivalent	
TP	Report the actual position of the motor	Use MC_ReadActualPosition	
TR	Trace the script program	Use the debug mode, Set a breakpoint for single step mode.	
TS	Tell Status Switches	Use MC_ReadStatus, Global variables, and Done output of MC_TouchProbe to see the status of the items listed in the TS command	
TT	Tell the torque of the motor	Use PLCopen function block MC_ReadActualTorque	
TV	Tell the velocity of the motor	Use PLCopen function block MC_ReadActualVelocity	

SMC Command	SMC Command Description	IEC-61131 / PLCopen equivalent	Note
TW	Timeout Wait for encoder to be in position at the end of a move	No equivalent.	Use the P_CON input from the servopack I/O driver in conjunction with the Done output from a PLCopen function block. Also incorporate a TON function. If the TON is complete before the P_CON, then the encoder did not get into position before the timeout occurred.
UL	Upload Program from controller	Use the Resource dialog box. Program source code must have been previously stored on the controller.	
VF	Variable Format	Use the Debug Dialog box to set the display type for on-screen values.	
WC	Wait for Contour to finish	No equivalent	
WT	Trippoint to wait for a specified time	TON and R_TRIG functions	
XQ	Execute a task	No equivalent	Only similarity is the EVENT task function block in the ProConOS firmware library, which allows executing an EVENT task, but its usefulness is very limited because PLCopen function blocks are designed to operate in a CYCLIC task environment.
ZA			
ZB			
ZR			
ZS	Zero Subroutine stack	No equivalent	Should not have to replicate this in IEC61131