Class 5 Command Comparison

Class 5 Motion Command Quick Reference

		Absolute Position	Relative Position	Velocity	Accel and Decel Together	Accel	Decel	Following Error	DE/Dt Derivitive Error Limit	Over Speed Limit
Report	Actual	RPA	RPRA	RVA	RAA	RAA		REA	RDEA	
Report	End Target	RPT	RPRT	RVT	RAT	RAT	RDT	REL	RDEL	RVL
Report	Commanded	RPC	RPRC	RVC	RAC	RAC				
Assign	End Target	PT=	PRT=	VT=	ADT=	AT=	DT=			
Assign	Command	N/A	N/A	N/A	N/A	N/A	N/A	EL=	DEL=	VL=

NOTE: For the complete list of Class 5 (and later) commands, see the *SmartMotor Developer's Guide*.

CLASS 4 PLS2		CLASS 5		
Mode/Motion Commands		**Mode/Motion Commands**		
F=2	'Invert Commutation	MINV(1)	'Invert motor commutation	
Not assigning F=2 will keep default Direction		MINV(0)	Default Motor Commutation Direction	
MP	'Mode Position (Default)	MP	'Mode Position (Default)	
V=x	'Set Target Velocity	VT=x	'Set Target Velocity	
A=x	'Set Target Accel./Decel.	ADT=x	'Set Target Accel./Decel.	
		AT=x	'Set Target Accel only	DT follows AT until DT is assigned
		DT=x	'Set Target Decel only	
P=x	'Set Target Absolute Position	PT=x	'Set Target Absolute Position	
D=x	'Set Target Relative Position	PRT=x	'Set Target Relative Position	
X	Decel to stop at rate of "A"	X	'Decel to stop at rate of "AT"	Note: If AT not assigned , Fast stop at "S" rate
S	'Stop at fast fixed rate	S	'Stop at fast fixed rate	
MV	'Mode Velocity	MV	'Mode Velocity	
Reporting Motion and Position Values		**Reporting Motion & Position Values**		**8000**
O=0	'Set present position to zero	O=0	'Set present position to zero	
		OSH(x)	'Shift present position by "x" counts, may be used on the fly	
RP	'Report Actual Position	RPA	'Report Actual Position	
x=P Rx	'Report Target Position	RPT	'Report Target Position	
		RPC	'Report Commanded Position	
RV	'Report Commanded Velocity	RVC	'Report Commanded Velocity	
x=V Rx	'Report Target Velocity	RVT	'Report Target Velocity	
RA	Report Target Acceleration	RAT	'Report Target Acceleration	
E=x	'Set Max Following Error Limit	EL=x	'Set Max Following Error Limit	
RE	'Report Error Limit	REL	'Report Error Limit	
		DEL=x	Set Derivative Error Limit	
		RDEL	Report Derivative Error Limit	
		RDEA	Report Actual Derivative Error	

Class 5 Command Comparison

CLASS 4 PLS2		CLASS 5		
Open Loop Mode		**Open Loop Mode**		
MT	'Mode Torque (No G required)	MT	'Mode Torque (G will re-enter)	
T=x	'Assign Torque Value, 0 to +/-1023	T=x	'Assign Torque Value, 0 to +/-32765 Note: Torque Mode requires a "G" command in Class 5	
Slave/Following Modes:		**Slave/Following Modes:**		
Step and Direction Input Modes:		**Step and Direction Input Modes:**		
MS0	'Set Ports A and B as Step/Dir	MS0	'Set Ports A and B as Step/Dir	
	'and zero out CTR (ext. counter)		'and zero out CTR(1) (ext. counter)	
'Note: Also zero's out CTR register		'Note: Also zero's out CTR register		
MS	Mode Step at 1:1 (No G required)	No Equivalent in Class 5		
MFMUL=x	Set Incoming pulse multiplier	MFMUL=x	Set Incoming pulse multiplier	RMFMUL Report MFMUL
MFDIV=x	Set Incoming Pulse Divisor	MFDIV=x	Set Incoming Pulse Divisor	RMFDIV Report MFDIV
MSR	'Calculate Mode Step Ratio	MSR	'Calculate Mode Step Ratio	
G	'MSR takes effect, Drive Activates	G	'MSR takes effect, Drive Activates	
RCTR	Report External Counter Value	RCTR(1)	Report External Counter Value	
Encoder Following Mode		**Encoder Following Mode**		
MF0	'Set Ports A and B as Encoder Input	MF0	'Set Ports A and B as Encoder Input	
'Note: Also zero's out CTR register		'Note: Also zero's out CTR register		
MF1	Mode Follow 1:4 (No G required)			
MF1	Mode Follow 1:2 (No G required)			
MF4	Mode Follow 1:1 (No G required)			
MFMUL=x	Set Incoming pulse multiplier	MFMUL=x	Set Incoming pulse multiplier	
MFDIV=x	Set Incoming Pulse Divisor	MFDIV=x	Set Incoming Pulse Divisor	
		MFA(x)	accel over x master distance. Default is zero (off)	
		MFD(x)	decel over x master distance. Default is zero (off)	
		MFX	Slow down based on MFD.	
		MFSLEW(x)	Stay at slew for “x” distance, then decel.	
MFR	'Calculate Mode Follow Ratio	MFR	'Calculate Mode Follow Ratio	
G	'MFR Takes effect, Drive Activates	G	'MFR Takes effect, Drive Activates	
RCTR	Report External Counter Value	RCTR(1)	Report External Counter Value	

Class 5 Command Comparison

CLASS 4 PLS2		CLASS 5	
Program Flow		**Program Flow**	
RUN	'Start Program Execution	RUN	'Start Program Execution
Note: No changes to any values in Flash		Note: No changes to any values in Flash	
RUN?	'Wait at this point for RUN command	RUN?	'Wait at this point for RUN command
GOTO#	'GOTO C#	GOTO(x)	'GOTO C# where x can be any number 0-999
GOSUB#	Call subroutine #	GOSUB(x)	Call subroutine "x"
IF {condition is true}		IF {condition is true}	
ELSEIF {condition is true}		ELSEIF {condition is true}	
ELSE {if no other condition above is true}		ELSE {if no other condition above is true}	
ENDIF	End if IF code structure	ENDIF	End if IF code structure
Note: only two operands allowed		Class 5 allows complex comparisons:	
Example:		**Example:**	
IF a<b	If a is less than b	IF (a<b)&(c<d)	If a is less than b and c is less than d
		IF ABS(EA)>x	If the absolute value of position error is greater than x
WHILE {condition is true}		WHILE {condition is true}	
LOOP	Execute any code between WHILE and LOOP when true	LOOP	Execute any code between WHILE and LOOP when true
Example:		**Example:**	
WHILE a<b	While a is less than b	WHILE a<b	While a is less the b careful not to add this too
BREAK	break out of WHILE loop	BREAK	break out of WHILE loop
Example:		**Example:**	
WHILE a>b IF c>d BREAK ENDIF LOOP		WHILE a>b IF c>d BREAK ENDIF LOOP	

Class 5 Command Comparison

CLASS 4 PLS2		CLASS 5	
Math Functions		**Math Functions**	
+	Addition	**+**	Addition
-	Subtraction	**-**	Subtraction
*****	Multiplication	*****	Multiplication
/	Division	**/**	Division
		%	Modulo (returns remainder of division)
		ABS(*x*)	Absolute Value
		^	Raise to Power
		Integer Only:	
		SQRT(*x*)	Square Root
		ABS(*x*)	Absolute Value
		Floating Point:	
		FSQRT(*xf*)	Square Root
		FABS(*xf*)	Absolute Root
		SIN(*xfd*)	Sine
		COS(*xfd*)	Cosine
		TAN(*xfd*)	Tangent
		ACOS(*xfd*)	Arc Cosine
		ASIN(*xfd*)	Arc Sine
		ATAN(*xfd*)	Arc Tangent
		See notes in Variables Section for more information	

NOTE:

x - is reduced/truncated to a whole integer
xf - is promoted to a float
xfd - is a value in degrees always promoted to float

CLASS 4 PLS2		CLASS 5	
Comparison Operators		**Comparison Operators (may be used as Logical Assignment)**	
>	Greater Than	>	Greater Than
<	Less Than	<	Less Than
==	Equal To	==	Equal To
!=	Not Equal To	!=	Not Equal To
>	Greater Than	>	Greater Than
>=	Greater than or Equal To	>=	Greater than or Equal To
<=	Less then or Equal To	<=	Less then or Equal To
|	Logical OR	|	Logical OR
&	Logical AND	&	Logical AND
		!|	Exclusive OR

Class 5 Command Comparison

CLASS 4 PLS2		CLASS 5	
Variables	**32 bit signed integer**	**Variables**	**32 bit signed integer**
a thru z		a thru z	
aa thru zz		aa thru zz	
aaa thru zzz		aaa thru zzz	
Array Access from aa to zzz		Array Variables (Dedicated)	
ab[0] thru ab[200]	8 Bit Byte array access	ab[0] thru ab[200]	8 Bit Byte array access
aw[0] thru aw[100]	16 Bit Word array access	aw[0] thru aw[100]	16 Bit Word array access
al[0] thru al[50]	32 Bit Long array access	al[0] thru al[50]	32 Bit Long array access
Note: All array variables overlap aa thru zzz in Class 4 motors!!!			
		Floating Point Array Variables:	
		af[0] thru af[7]	
		The floating point array variables meet IEEE-754 specification. So they are a true floating point where the location of the decimal point can vary with the exponent from very tiny i.e. approximately 1x10^-300 to very large approximately 1x10^+300. Class 5 can do basic operations at 64-bit precision: +, -, *, /. BUT, the trig functions are only being calculated with 32-bit precision	

CLASS 4 PLS2		CLASS 5	
All I/O commands refer to local I/O unless otherwise noted			
Configuring Ports as general Inputs:		**Configuring Ports as general Inputs:**	
UAI	'set User port A as Input	EIGN(0)	'set User port 0 as Input
UBI	'set User port B as Input	EIGN(1)	'set User port 1 as Input
UCI	'set User port C as Input	EIGN(2)	'set User port 2 as Input
UDI	'set User port D as Input	EIGN(3)	'set User port 3 as Input
UEI	'set User port E as Input	EIGN(4)	'set User port 4 as Input
UFI	'set User port F as Input	EIGN(5)	'set User port 5 as Input
UGI	'set User port G as Input	EIGN(6)	'set User port 6 as Input
Configuring Ports as general Outputs:		**Configuring and Setting Ports as general Outputs:**	
UAO	'set User port A as Output	OUT(0)=	'set/reset Port 0 Output state
UBO	'set User port B as Output	OUT(1)=	'set/reset Port 1 Output state
UCO	'set User port C as Output	OUT(2)=	'set/reset Port 2 Output state
UDO	'set User port D as Output	OUT(3)=	'set/reset Port 3 Output state
UEO	'set User port E as Output	OUT(4)=	'set/reset Port 4 Output state
UFO	'set User port F as Output	OUT(5)=	'set/reset Port 5 Output state
UGO	'set User port G as Output	OUT(6)=	'set/reset Port 6 Output state

CLASS 4 PLS2		CLASS 5	
Resetting Output Logic States:		**Configuring and Resetting Output Logic States:**	
UA=0	'set User port A to 0VDC	OR(0)	'set User port 0 to 0 VDC
UB=0	'set User port B to 0VDC	OR(1)	'set User port 1 to 0 VDC
UC=0	'set User port C to 0VDC	OR(2)	'set User port 2 to 0 VDC
UD=0	'set User port D to 0VDC	OR(3)	'set User port 3 to 0 VDC
UE=0	'set User port E to 0VDC	OR(4)	'set User port 4 to 0 VDC
UF=0	'set User port F to 0VDC	OR(5)	'set User port 5 to 0 VDC
UG=0	'set User port G to 0VDC	OR(6)	'set User port 6 to 0 VDC
Setting Output Logic States:		**Configuring and Setting Output Logic States:**	
UA=1	'set User port A to 5VDC	OS(0)	'set User port 0 to 5 VDC
UB=1	'set User port B to 5VDC	OS(1)	'set User port 1 to 5 VDC
UC=1	'set User port C to 5VDC	OS(2)	'set User port 2 to 5 VDC
UD=1	'set User port D to 5VDC	OS(3)	'set User port 3 to 5 VDC
UE=1	'set User port E to 5VDC	OS(4)	'set User port 4 to 5 VDC
UF=1	'set User port F to 5VDC	OS(5)	'set User port 5 to 5 VDC
UG=1	'set User port G to 5VDC	OS(6)	'set User port 6 to 5 VDC

Class 5 Command Comparison

CLASS 4 PLS2		CLASS 5	
Set ports A and B as counter inputs (just counting):		**Set ports A and B as counter inputs (just counting):**	
MS0	'(A) Step and (B) Direction Input	MS0	'(A) Step and (B) Direction Input
MF0	'(A) and (B) as Enc. Input	MF0	'(A) and (B) as Enc. Input
CLASS 4 PLS2		**CLASS 5**	
Limit Switch Control (Ports C and D) Default State		**Limit Switch Control (Ports C and D) Default State**	
UCP	'Set Port C as Pos. Travel Limit	EILP	'Set Port C as Pos. Travel Limit
UDM	'Set Port D as Neg. Travel Limit	EILN	'Set Port D as Neg. Travel Limit
CLASS 4 PLS2		**CLASS 5**	
Go (G) command assignment		**Go (G) command assignment**	
UG	'Set Port G to "G" command	EISM(6)	'Set Port G to "G" command
CLASS 4 PLS2		**CLASS 5**	
Brake Control		**Brake Control**	
BRKI	'Default, operates internal brake	'EOBK(0-6 are valid),	
BRKC	'Redirect control to Port C	EOBK(2)	'Redirect control to Port C
BRKG	'Redirect control to Port D	EOBK(6)	'Redirect control to Port D

CLASS 4 PLS2		CLASS 5	
Report I/O Signal Levels		**Report I/O Signal Levels**	
Digital returns 0 or 1 for 0 or 5 VDC		Digital returns 0 or 1 for 0 or 24 VDC	
RUA	'Report Port A Digital	RIN(0)	'Report Port 0 Digital
RUB	'Report Port B Digital	RIN(1)	'Report Port 1 Digital
RUC	'Report Port C Digital	RIN(2)	'Report Port 2 Digital
RUD	'Report Port D Digital	RIN(3)	'Report Port 3 Digital
RUE	'Report Port E Digital	RIN(4)	'Report Port 4 Digital
RUF	'Report Port F Digital	RIN(5)	'Report Port 5 Digital
RUG	'Report Port G Digital	RIN(6)	'Report Port 6 Digital
Group Reporting of all Local I/O		Group Reporting of all Local I/O	
RU	'Report all 7 I/O (7 bit)	RIN(B,0)	'Report all 10 I/O (10 bit)
CLASS 4 PLS2		**CLASS 5**	
Analog Input Commands		**Analog Input Commands**	
Analog returns 0-1023 for 0 to 5 VDC		Analog returns 0-32736 for 0 to 24 VDC	
RUAA	'Report Port A Analog	RINA(V1,0)	'Report Port 0 Analog
RUBA	'Report Port B Analog	RINA(V1,1)	'Report Port 1 Analog
RUCA	'Report Port C Analog	RINA(V1,2)	'Report Port 2 Analog
RUDA	'Report Port D Analog	RINA(V1,3)	'Report Port 3 Analog
RUEA	'Report Port E Analog	RINA(V1,4)	'Report Port 4 Analog
RUFA	'Report Port F Analog	RINA(V1,5)	'Report Port 5 Analog
RUGA	'Report Port G Analog	RINA(V1,6)	'Report Port 6 Analog
CLASS 4 PLS2		**CLASS 5**	
Assign analog value to a variable		**Assign analog scaled voltage to a variable**	
x=UAA	'Assign Port A analog to "x"	x=INA(V1,0)	'Assign Port 0 analog to "x"
x=UBA	'Assign Port B analog to "x"	x=INA(V1,1)	'Assign Port 1 analog to "x"
x=UCA	'Assign Port C analog to "x"	x=INA(V1,2)	'Assign Port 3 analog to "x"
x=UDA	'Assign Port D analog to "x"	x=INA(V1,3)	'Assign Port 4 analog to "x"
x=UEA	'Assign Port E analog to "x"	x=INA(V1,4)	'Assign Port 5 analog to "x"
x=UFA	'Assign Port F analog to "x"	x=INA(V1,5)	'Assign Port F analog to "x"
x=UGA	' Assign Port G analog to "x"	x=INA(V1,6))	'Assign Port 6 analog to "x"