I Revision 0 0 0	Туре	Command	Dephy	Design and a	
0			Reply	Purpose	Comment
	Date / Time Date / Time	:SCMM/DD/YY# :GC#	0 or 1 MM/DD/YY#	Set date Get date	
	Date / Time	:SLHH:MM:SS#	0 or 1	Set time (Local)	
0	Date / Time	:Ga#	HH:MM:SS#	Get time (Local, 12hr format)	
0	Date / Time	:GL#	HH:MM:SS#	Get time (Local, 24hr format)	
0	Date / Time	:SSHH:MM:SS#	0 or 1	Set time (Sidereal)	
0	Date / Time	:GS#	HH:MM:SS#	Get time (Sidereal)	
0	Site/Location	:SgsHH#	0 or 1	Set UTC Offset(for curent site)	The UTC Offset value is the number of hours to add to your Local Time (Standard Time) to get Universal Time.
0	Site/Location	:GG#	sHH#	Get UTC Offset(for curent site)	
0 0	Site/Location	:StsDD*MM#	0 or 1	Set Latitude (for curent site)	
1 0	Site/Location	:Gt#	sDD*MM#	Get Latitude (for curent site)	
2 0 3 0	Site/Location Site/Location	:SgDDD*MM# :Gg#	0 or 1 DDD*MM#	Set Longitude (for curent site) Get Longitude (for curent site)	
4 0	Site/Location	:Smsss#	0 or 1	Set site 0 name	
5 0	Site/Location	:Snsss#	0 or 1	Set site 1 name	
6 0	Site/Location	:Sosss#	0 or 1	Set site 2 name	
7 0	Site/Location	:Spsss#	0 or 1	Set site 3 name	
B 0	Site/Location	:GM#	sss#	Get site 0 name	
9 0	Site/Location	:GN#	SSS#	Get site 1 name	
0 L 0	Site/Location Site/Location	:GO# :GP#	SSS#	Get site 2 name Get site 3 name	
2 0	Site/Location	:Wn#	sss# [none]	Select site n (0-3)	
3 0	Slewing/Movement	:SrHH:MM:SS#	0 or 1 *	Set target RA	* = Defaults to high precision mode, in low
					precision mode "HH:MM.M", "sDD*MM", or "DDD*MM" are used as appropriate.
4 0	Slewing/Movement	:Gr#	HH:MM:SS#*	Get target RA	
5 0	Slewing/Movement	:SdsDD:MM:SS#	0 or 1 *	Set target Dec Get target Dec	
6 0 7 0	Slewing/Movement Slewing/Movement	:Gd# :SzDDD:MM:SS#	sDD*MM'SS#* 0 or 1 *	Get target Dec Set target Azm	
3 0	Slewing/Movement	:SasDD:MM:SS#	0 or 1 *	Set target Alt	
9 0	Slewing/Movement	:GR#	HH:MM:SS#*	Get telescope RA	
0	Slewing/Movement	:GD#	sDD*MM'SS#*	Get telescope Dec	
L O	Slewing/Movement	:GZ#	DDD*MM'SS#*	Get telescope Azm	
2 0	Slewing/Movement	:GA#	sDD*MM'SS#*	Get telescope Alt	
3 0	Limits	:ShsDD#	0 or 1	Set horizon limit	The horizon limit sets how far below (or above) the horizon the telescope will point for a goto: Valid range (in degrees) is +30 to -30.
4 0	Limits	:GhsDD#	sDD#	Get horizon limit	
5 0	Limits	:SoDD#	0 or 1	Set overhead limit	The overhead limit helps keep the telescope tut from hitting the tripod etc. during a goto: Valid range (in degrees) is 60 to 90. The overhead limit helps keep the telescope tut from hitting the tripod etc. during a goto
6 0	Limits	:GoDD#	sDD#	Get overhead limit	
7 0	Movements	:MS#	e *2	Move telescope (to current Equ target)	*2 = Error codes for the MS and MA commands are as follows: e=0 (no error), e=1 (below horizon), e=2 (no object), e=4 (position unreachable), e=5 (not aligned), E=6 (outside limits)
в 0	Movements	:MA#	e *2	Move telescope (to current Hor target)	
9 0	Movements	:Q#	[none]	Stop telescope	
0 0	Movements	:Me#	[none]	Move telescope east (at current rate)	
L 0	Movements	:Mw#	[none]	Move telescope west (at current rate)	
2 0	Movements	:Mn#	[none]	Move telescope north (at current rate)	
3 0	Movements	:Ms#	[none]	Move telescope south (at current rate)	
0 5 0	Movements Movements	:Qe# :Qw#	[none]	Stop moving east Stop moving west	
5 0 5 0	Movements	:Qn#	[none] [none]	Stop moving north	
7 0	Movements	:Qs#	[none]	Stop moving south	
3 0	Rates	:Mgdnnnn#	[none]	(from 20 to 16399mS)	Pulse guide (at current rate): D=n,s,e,w,
9 0	Rates	:RG#	[none]	Set rate to Guide	Nnnn=time in mS (from 20 to 16399mS)
0 0	Rates	:RC#	[none]	Set rate to Centering	
L 0	Rates	:RM#	[none]	Set rate to Move	
2 0	Rates	:RS#	[none]	Set rate to Slew	
3 0	Rates	:Rn#	[none]	Set rate to n (0-9)*3	*3 = Slew rates are as follows : All values are in multipules of the sidereal rate: R0=0.25X, R1=0.5X, R2(RG)=1X, R3=2X, R4(RC)=4X, R5=8X(RM), R5=60X (for the -Dev-Alpha branch of OnStep: R0=0.25X, R1=0.5X, R1=0.5X, R3=2X, R4=4X, R3=2X, R4=4X, R5(RC)=8X, R6(RM)=24X, R7=48X, R6(RM)=24X, R7=48X, R6(RM)=24X, R7=48X, R6(RM)=24X, R7=48X, R6(RS)=112 MaxRate, R9=MaxRate)
4 0	Pier Pier	:D# :Gm#	\0x7F# N#, E#or W#	Get distance bars (indicates slew) Pier side	R6(RM)=24X, R7=48X, R8(RS)=1/2 MaxRate,

				C	ommands	
56	0	Tracking rate	:STdd.dddd#	0 or 1	Set sidereal rate RA	Tracking rate adjustment is as follows: The sidereal rate is default and is always selected on power-up. The T+ and T- commands can adjust any of the rates; however only the sidereal rate, if selected, remembers the adjusted rate through a power cycle. Refraction rate tracking adjusts the RA rate dynamically to best compensate for refraction in a given region of the sky; again this works for any of the rates. This setting isn't remembered between power cycles, but the OnStep firmware can be compiled so that this setting defaults to enabled.
57 58	0	Tracking rate Tracking rate	:GT# :TQ#	dd.ddddd# [none]	Get sidereal rate RA Track sidereal rate RA (default)	
	0	Tracking rate	:TR#	[none]	Track sidereal rate reset	
	0	Tracking rate	:T+#	[none]	Track rate increase 0.02Hz	
61 62	0	Tracking rate Tracking rate	:T-# :TS#	[none]	Track rate decrease 0.02Hz Track solar rate RA	
63	0	Tracking rate	:TL#	[none]	Track lunar rate RA	
	0 0	Tracking rate Tracking rate	:TK# :Te#	[none] 0 or 1	Track king rate RA Tracking enable	
66	0	Tracking rate	:Td#	0 or 1	Tracking disable	
67 68	0	Tracking rate Tracking rate	:Tr# :Tn#	0 or 1 0 or 1	Refraction rate tracking No refraction rate tracking	
69	0	Sync	:CS#	[none]	Sync. with current target RA/Dec	Note: Sync's that are not allowed fail silently. This can happen due to slews, parking, or exceeded limits.
70	0	Sync	:CM#	N/A#	Sync. with current target RA/Dec	
	0	Library Library	:Lonn# :LB#	0 or 1 [none]	Select catalog no. Move Back in catalog	
73	0	Library	:LN#	[none]	Move to Next in catalog	
74 75	0 0	Library Library	:LCnnnn# :L\$#	[none] 1	Move to catalog item no. Move to catalog name rec.	
75 76	0	Library	:Ll#	name,type#	Get catalog item id.	
77	0	Library	:LR#	name,type,RA,Dec#	Read catalog item info. (also moves	
78	0	Library	:LWssss,ttt#	0 or 1	forward) Write catalog item info.	ssss=name, ttt=type code: UNK,
						OC, GC, PN, DN, SG, EG, IG, KNT, SNR, GAL, CN, STR, PLA, CMT,
79						AST
	0	Library Library	:LD# :LL#	[none] [none]	Clear current record Clear current cataLog	AST
80	0	Library Library	:LL# :L!#	[none] [none]	Clear current cataLog Clear all catalogs	AST The LI#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Sd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$. A special search can then be done with the :L\$#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface speed limitations.
80 81 82	0	Library	:LL#	[none]	Clear current cataLog	The LI#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Gd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L\$#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface
80 81 82 83 84	0 0 0 0 0 0	Library Library Anti-backlash Anti-backlash Anti-backlash	:LL# :LI# \$BRnnn# :\$BDnnn# :\$QZ+#	[none] [none] 0 or 1 0 or 1 [none]	Clear current cataLog Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (Alt) backlash amount (in ArcSec) Turn PEC on	The LI#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Gd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L\$#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface
80 81 82 83 84 85	0 0 0 0	Library Library Anti-backlash Anti-backlash	:LL# :LI# :SBRnnn# :SBRnnn#	[none] [none] 0 or 1 0 or 1	Clear current cataLog Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (Alt) backlash amount (in ArcSec)	The LI#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Gd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L\$#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface
80 81 82 83 84 85 86 87	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Library Library Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash	:LL# :LI# :SBRnnn# :SBDnnn# :SQZ+# :SQZ=# :SQZZ# :SQZZ# :SQZZ#	[none] [none] 0 or 1 0 or 1 [none] [none] [none] [none] [none]	Clear current cataLog Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (Alt) backlash amount (in ArcSec) Turn PEC on Turn PEC off Clear PEC data Start recording PEC	The LI#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Gd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L\$#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface
80 81 83 83 84 85 86 87 88 89	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Library Library Library Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash	:LL# :LI# :LI# :SBRnnn# :SBDnn# :SQZ+# :SQZ+# :SQZ# :SQZ!# :SQZ!# :SQZ!# :SQZ!#	[none] [none] 0 or 1 0 or 1 [none] [none] [none] [none] [none] [none] [none] [s#	Clear current cataLog Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (Alt) backlash amount (in ArcSec) Turn PEC on Turn PEC off Clear PEC data/ Start recording PEC Save PEC data/settings to EEPROM Get PEC status returns: I-Ignore PEC, P- Playing PEC, p-Getting ready to record PEC	The LI#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Gd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L\$#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface
80 81 81 83 83 84 85 86 87 88 88 89 90 91	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Library Library Library Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash	:LL# :LL# :LI# :SBRnnn# :SBDnn# :SQZ+# :SQZ+# :SQZ# :SQZ# :SQZ!# :SQZ!# :SQZ!# :SQZ!# :SQZ?#	[none] [none] 0 or 1 0 or 1 0 or 1 [none] [none] [none] [none] [none] S# \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$	Clear current cataLog Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (All) backlash amount (in ArcSec) Turn PEC on Turn PEC on Turn PEC on Turn PEC off Clear PEC data/settings to EEPROM Get PEC status returns: I-Ignore PEC, P- Playing PEC, p-Getting ready to play PEC, R-Record PEC, r-Getting ready to record PEC Readout PEC data at current index (while playing/recording), also returns index	The LI#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Gd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L\$#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface
80 81 81 83 83 84 85 88 88 89 90 91 92	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Library Library Library Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash Anti-backlash	:LL# :LL# :LI# :SBRnnn# :SBDnn# :SQDnn# :SQZ# :SQZ# :SQZ# :SQZ# :SQZ# :SQZ# :SQZ# :SQZ# :SQZ# :SQZ!# :SQZ?#	[none] [none] 0 or 1 0 or 1 [none] [none] [none] [none] s# skdd#	Clear current cataLog Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (Alt) backlash amount (in ArcSec) Turn PEC off Clear PEC data Start recording PEC Save PEC data/settings to EEPROM Get PEC data/settings to EEPROM Get PEC data/settings to EEPROM Get PEC data/settings to EEPROM PEC, P-Getting ready to play PEC, R-Record PEC, r-Getting ready to record PEC Readout PEC data Readout PEC data Readout PEC data	The LI#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Gd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L\$#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface
80 81 81 83 83 84 85 86 87 88 89 90 91 92 93 94	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Library Library Library Anti-backlash	:LL# :LL# :LI# :LI# :SBRnnn# :SBDnn# :SQZ+# :SQZ	[none] [none] [one] [none] [none] [none] [none] [none] [none] S# sddd# sddd# sddd# o o 1 0 o 1 0 o 1 0 o 1	Clear current cataLog Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (All) backlash amount (in ArcSec) Turn PEC on Turn PEC on Turn PEC on Turn PEC on Clear PEC data Start recording PEC Save PEC data/settings to EEPROM Get PEC status returns: I-Ignore PEC, P- Playing PEC, p-Getting ready to play PEC, R-Record PEC, r-Getting ready to record PEC Readout PEC data Readout PEC data at current index (while playing/recording), also returns index Write PEC data Align, accept*4	The LI#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Gd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L\$#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface
80 81 81 83 84 85 86 88 89 91 92 93 94 95	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Library Library Library Anti-backlash	:LL# :LL# :LL# :LI# :SBRnnn# :SBDnn# :SQZ# :SQZ# :SQZ# :SQZ# :SQZ# :SQZ# :SQZ# :SQZ# :SQZ# :SQZ# :SQZ!# :SQZ!# :SQZ?# :SQZ!# :SQZ?# :SQZ!# :SQZ?# :SQZ!# :SQZ?# :SQZ!# :SQZ?# :SQZ!# :SQZ?# :SQZ!# :SQZ?# :SQZ?# :SQZ!# :SQZ?# :SQZ!# :SQZ?# :SQZ! :SQZ! :SQZ!# :SQZ!# :SQZ!# :SQZ!# :SQZ! :SQZ! :SQZ! :SQZ! :SQZ! :SQZ! :SQZ! :SQZ! :SQZ! :SQZ!# :SQZ!# :SQZ!	[none] [none] 0 or 1 0 or 1 0 or 1 [none] [none] [none] [none] [s# \$ddd,ddd# 0 or 1 0 or 1	Clear current cataLog Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (Alt) backlash amount (in ArcSec) Turn PEC off Clear PEC data Start recording PEC Save PEC dataSettings to EEPROM Get PEC dataSettings to EEPROM Get PEC dataSettings to EEPROM Get PEC dataSettings to tecord PEC R-Record PEC, r-Getting ready to play PEC, R-Record PEC, r-Getting ready to play PEC, R-Record PEC data Readout PEC data at current index (while playing/recording), also returns index Write PEC data Align, write model to EEPROM Align, accept*4 Set park position	The LI#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Gd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L\$#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface
80 81 81 82 83 84 85 86 87 88 89 91 92 93 94 95 96 97	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Library Library Library Anti-backlash	:LL# :LL# :LI# :LI# :SBRnnn# :SBDnn# :SQZ+# :SQZ	[none] [none] [none] 0 or 1 0 or 1 0 or 1 [none] [none] [none] [none] [none] [none] [none] S# Sddd# Sddd# Sddd# O or 1 0 or 1 [none] [][][][][][][][][][][][][][][][][][][Clear current cataLog Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (All) backlash amount (in ArcSec) Turn PEC on Turn PEC on Turn PEC on Turn PEC off Clear PEC data Start recording PEC Save PEC data/settings to EEPROM Get PEC status returns: I-gnore PEC, P- Playing PEC, p-Getting ready to play PEC, R-Record PEC, r-Getting ready to play PEC, R-Record PEC, r-Getting ready to record PEC Readout PEC data at current index (while playing/recording), also returns index Write PEC data Align, accept ^{r4} Set park position Move to park position Restore parked telescope to operation	The LI#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Gd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L\$#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface
80 81 81 82 82 84 85 88 87 88 89 90 91 92 93 94 95 96 97 98	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Library Library Library Anti-backlash	:LL# :LL# :LL# :LI# :SBRnnn# :SBDnn# :SBDnn# :SQZ# :SQ	inone] [none] [none] [none] 0 or 1 [none] [none] [none] [none] [none] [none] [none] [none] [none] [none] [none] [none] [none] \$ddd,ddd# 0 or 1 0 or 1 0 or 1 0 or 1 0 or 1 0 or 1 [none]	Clear current cataLog Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (Alt) backlash amount (in ArcSec) Turn PEC off Clear PEC data Start recording PEC Save PEC data/settings to EEPROM Get PEC data/settings to EEPROM Get PEC data/settings to EEPROM Get PEC data/setting ready to play PEC, P- Playing PEC, p-Getting ready to play PEC, R-Record PEC, r-Getting ready to record PEC Readout PEC data at current index (while playing/recording), also returns index Write PEC data Align, accept 4 Set park position Move to park position Restore parked telescope to operation Set home (CWD)	The LI#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Gd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L\$#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface
80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Library Library Library Anti-backlash Anti-b	:LL# :LL# :LL# :LI# :SBDnnn# :SBDnnn# :SQZ+# :SQZ# :SQ	inone] [none] 0 or 1 0 or 1 [none] 0 or 1	Clear current cataLog Clear all catalogs Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (All) backlash amount (in ArcSec) Turn PEC on Turn PEC on Turn PEC off Clear PEC data Start recording PEC Save PEC data/settings to EEPROM Get PEC status returns: I-Ignore PEC, P- Playing PEC, p-Getting ready to play PEC, R-Record PEC, r-Getting ready to play PEC, R-Record PEC, r-Getting ready to record PEC Readout PEC data at current index (while playing/recording), also returns index Write PEC data Align, write model to EEPROM Align, accept*4 Set park position Restore parked telescope to operation Set home (CWD) Move to home (CWD)	The LI#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Gd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L\$#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface
80 81 81 82 83 84 85 86 87 88 89 91 92 93 94 95 96 97 98 99 100 1001	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Library Library Library Anti-backlash Anti-b	:LL# :LL# :LL# :LH# :SBDnn# :SBDnn# :SBDnn# :SQZ	inone] [none] [none] [none] 0 or 1 [none] 0 or 1 [none] [none] [none] 0 or 1 0 or 1 0 or 1 0 or 1 0 or 1 [none] [none] [none] [none] [none] 0 or 1 0 or 1	Clear current cataLog Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (Alt) backlash amount (in ArcSec) Turn PEC off Clear PEC data Start recording PEC Save PEC data/settings to EEPROM Get PEC data/settings to EEPROM Get PEC data/settings to EEPROM PEC Character (Marching) Get PEC data/setting ready to play PEC, R-Record PEC, r-Getting ready to play PEC, R-Record PEC, r-Getting ready to record PEC Readout PEC data Align, write model to EEPROM Align, accept 4 Set park position Move to park position Restore parked telescope to operation Set home (CWD) Move to home (CWD) Focuser1 Active?	The LI#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Gd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L\$#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface
80 81 82 83 84 85 88 87 90 91 92 93 94 95 96 97 98 99 100 101 102 103	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Library Library Library Anti-backlash Anti-b	:LL# :LL# :LL# :LI# :SBRnnn# :SBDnnn# :SQDnn# :SQZ# :S	inone] [none] 0 or 1 0 or 1 0 or 1 [none] 0 or 1	Clear current cataLog Clear all catalogs Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (All) backlash amount (in ArcSec) Turn PEC on Turn PEC on Turn PEC off Clear PEC data Start recording PEC Save PEC data/settings to EEPROM Get PEC status returns: I-Ignore PEC, P- Playing PEC, p-Getting ready to play PEC, R-Record PEC, r-Getting ready to play PEC, R-Record PEC, r-Getting ready to record PEC Readout PEC data at current index (while playing/recording), also returns index Write PEC data Align, write model to EEPROM Align, accept*4 Set park position Restore parked telescope to operation Set home (CWD) Move to home (CWD) Focuser1 Active? Select primary focuser n = 1 or 2 Get primary focuser	The LI#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Gd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L\$#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface
80 81 81 82 82 83 84 85 87 88 86 87 88 89 91 92 93 99 93 99 93 99 93 99 93 99 100 101 103 104	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Library Library Library Anti-backlash Anti-b	:LL# :LL# :LL# :LI# :LI# :SBRnnn# :SBDnn# :SBDnn# :SQZ	inone] [none] [none] 0 or 1 0 or 1 [none]	Clear current cataLog Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (Alt) backlash amount (in ArcSec) Turn PEC off Clear PEC data Start recording PEC Save PEC data/settings to EEPROM Get PEC data/settings to EEPROM Get PEC data/settings to EEPROM Get PEC data/settings to EEPROM PEC Readout PEC data at current index (while playing/recording), also returns index Write PEC data Align, accept ⁴ Set park position Move to park position Restore parked telescope to operation Set home (CWD) Focuser1 Active? Focuser2 Active? Select primary focuser n = 1 or 2 Get status M#= moving, S#= stopped	The Ll#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Sd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L5#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface speed limitations.
80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 1001 1021 103 1041 105	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Library Library Library Anti-backlash Anti-b	:LL# :LL# :LL# :LI# :LI# :SBRnnn# :SBDnn# :SQD+# :SQZ#	inone] [none] 0 or 1 0 or 1 [none]	Clear current cataLog Clear all catalogs Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (All) backlash amount (in ArcSec) Turn PEC off Clear PEC data Start recording PEC Save PEC data/settings to EEPROM Get PEC status returns: I-Ignore PEC, P- Playing PEC, p-Getting ready to play PEC, R-Record PEC, r-Getting ready to play PEC, R-Record PEC, r-Getting ready to record PEC Readout PEC data at current index (while Playing/recording), also returns index Write PEC data Align, accept*4 Set park position Move to park position Restore parked telescope to operation Set home (CWD) Move to home (CWD) Focuser1 Active? Focuser2 Active? Select primary focuser n = 1 or 2 Get primary focuser Get status M#= moving, S#= stopped Get mode 0 = absolute 1 = pseudo absolute	The Ll#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Sd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L5#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface speed limitations.
80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 99 99 99 99 99 100 101 104 105 104 104 104	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Library Library Library Anti-backlash Anti-b	:LL# :LL# :LL# :LH# :LH# :SBRnnn# :SQT# :SQZ	inone] [none] [none] 0 or 1 [none] [nor 1 [n# <td>Clear current cataLog Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (Alt) backlash amount (in ArcSec) Turn PEC on Turn PEC off Clear PEC data/Settings to EEPROM Clear PEC status returns: I-Ignore PEC, P- Playing PEC, p-Getting ready to play PEC, R-Record PEC data/Setting ready to play PEC, Readout PEC data current index (while playing/recording), also returns index Write PEC data Readout PEC data Set park position Move to park position Restore parked telescope to operation Set home (CWD) Focuser1 Active? Select primary focuser n = 1 or 2 Get status M#= moving, S#= stopped Get mode 0 = absolute 1 = pseudo absolute Get full in position (in microns or steps)</td> <td>The Ll#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Sd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L5#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface speed limitations.</td>	Clear current cataLog Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (Alt) backlash amount (in ArcSec) Turn PEC on Turn PEC off Clear PEC data/Settings to EEPROM Clear PEC status returns: I-Ignore PEC, P- Playing PEC, p-Getting ready to play PEC, R-Record PEC data/Setting ready to play PEC, Readout PEC data current index (while playing/recording), also returns index Write PEC data Readout PEC data Set park position Move to park position Restore parked telescope to operation Set home (CWD) Focuser1 Active? Select primary focuser n = 1 or 2 Get status M#= moving, S#= stopped Get mode 0 = absolute 1 = pseudo absolute Get full in position (in microns or steps)	The Ll#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Sd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L5#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface speed limitations.
80 81 82 83 84 85 86 88 87 88 88 89 90 91 92 93 99 99 99 99 99 99 99 99 90 91 100 100 1	0 0 0 0 0 0 0 0 0 0 0 0 0 0	Library Library Library Anti-backlash Anti-b	:LL# :LL# :LL# :LI# :LI# :SBDnn# :SBDnn# :SBDnn# :SQZ#	inone] [none] 0 or 1 0 or 1 none] [none] [nor1] [nor1]<	Clear current cataLog Clear all catalogs Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (All) backlash amount (in ArcSec) Turn PEC on Turn PEC off Clear PEC data Start recording PEC Save PEC data/settings to EEPROM Get PEC status returns: I-Ignore PEC, P- Playing PEC, p-Getting ready to play PEC, R-Record PEC, r-Getting ready to play PEC, R-Record PEC, c-Getting ready to record PEC Readout PEC data at current index (while playing/recording), also returns index Write PEC data Align, accept*4 Set park position Move to park position Restore parked telescope to operation Set home (CWD) Move to home (CWD) Focuser1 Active? Focuser2 Active? Select primary focuser n = 1 or 2 Get primary focuser Get status M#= moving, S#= stopped Get mode 0 = absolute 1 = pseudo absolute Get full in position (in microns or steps) Get max position (in microns or steps)	The Ll#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Sd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L5#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface speed limitations.
80 81 82 83 84 85 86 87 90 91 92 93 94 95 96 97 99 93 99 100 101 102 103 104 105 106 107	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Library Library Library Anti-backlash Anti-b	:LL# :LL# :LL# :LI# :SBRnnn# :SBDnn# :SBDnn# :SQZ+# :SQZ+# :SQZ+# :SQZ+# :SQZ# :S	inone] [none] [none] 0 or 1 0 or 1 0 or 1 [none] 0 or 1	Clear current cataLog Clear all catalogs Set RA (Azm) backlash amount (in ArcSec) Set Dec (All) backlash amount (in ArcSec) Turn PEC on Turn PEC on Turn PEC on Turn PEC on Clear PEC data Start recording PEC Save PEC data/settings to EEPROM Get PEC status returns: I-Ignore PEC, P- Playing PEC, p-Getting ready to play PEC, R-Record PEC, r-Getting ready to play PEC, R-Record PEC, r-Getting ready to record PEC Readout PEC data at current index (while playing/recording), also returns index Write PEC data Align, accept*4 Set park position Move to home (CWD) Focuser1 Active? Focuser2 Active? Select primary focuser n = 1 of 2 Get minary focuser Get status M#= moving, S#= stopped Get mode 0 = absolute 1 = pseudo absolute Get full in position (in microns or steps)	The Ll#and LW#commands also set/get target coordinates (as with :Gr#, :Sr#, :Gd#, :Sd#) Library record storage is in EEPROM. A catalog name record is like any other except the name must start with a '\$'. A special search can then be done with the :L.5#command to move to that record. It's up to the user to not waste EEPROM with more than one name record per catalog. When the default PEC table size of 824 bytes is used, the first 1024 bytes are devoted to settings. The remaining EEPROM is used for catalog records. Each record is 16 bytes. It's often best to divide up large Libraries into several smaller catalogs due to serial interface speed limitations.

No. Position Position Position Position Position 11 Position Position Position Position Position 12 Position Position Position Position Position 13 Position Position Position Position Position 14 Position Position Position Position Position 15 Position Position Position Position Position Position 15 Position Position Position Position Position Position 16 Position Position Position Position Position Position 17 Position Position Position Position Position Position 17 Position Position Position Position Pos					cc	ommands	
13 14 <	112	0	Focus	:FBn#	0 or 1		
14 14 15 16 </td <td>113</td> <td>0</td> <td>Focus</td> <td>:FC#</td> <td>n.n#</td> <td>Get focuser temperature compensation</td> <td></td>	113	0	Focus	:FC#	n.n#	Get focuser temperature compensation	
Image: Margin and the second of th	114	0	Focus	:FCsn.n#	0 or 1		
10 10 10 10 10 10 10 10 10 1						coefficient in um per deg. C (+ moves out	
10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10	115	0	Focus	:Fc#	0 or 1	Get focuser temperature compensation	
10 10 10 10 10 10 10 10 10 10 11 10 <	116	0	Focus	:Fcn#	0 or 1	Enable/disable focuser temperature	
10 10 10 10 10 10 10 10 10 10 <	117	0	Focus	:FD#	n#	Get focuser temperature compensation	
Image: Process of the second	118	0	Focus	:FDn#	0 or 1		
No. No. No. No. No. No. No. No. No. </td <td></td> <td>0</td> <td></td> <td></td> <td></td> <td>deadband amount (in microns or steps)</td> <td></td>		0				deadband amount (in microns or steps)	
Dia Resid Price P	120	0	Focus	:FPn#	0 or 1	Set focuser DC Motor Power Level (in %)	
13 14 15.00 15.00 15.00 15.00 15.00 15.00 15.00 15.00 15.00 17 17.00 <td></td> <td></td> <td></td> <td>-</td> <td><u> </u></td> <td></td> <td></td>				-	<u> </u>		
No. No. <td></td> <td></td> <td></td> <td></td> <td>[none]</td> <td>Set focuser for slow motion (0.01mm/s)</td> <td></td>					[none]	Set focuser for slow motion (0.01mm/s)	
125 Social Fuel Introd Nove finder at located dispersion 17 Resca Aless Aless Aless Aless 17 Resca Aless Aless Aless Aless 17 Resca Filteria Biologic Set focater higher dispersion hashing (mail and aless) 17 Resca Filteria Biologic Set focater higher dispersion hashing (mail and all aless) 17 Resca Filteria Resca Filteria Set focater higher dispersion hashing (mail and all aless) 17 Resca Filteria Resca Resca Resca Resca 17 Resca Filteria Resca Resca Resca Resca 17 Resca Resca Resca Resca Resca	124	0	Focus	.=1#	[none]	for 0.01mm/sec, n=3 for 0.1mm/sec, n=4 for	
177 Noals P62 9 P63 9 P63 9 P63 P63 P64	125	0	Focus		[none]		
Image: Constraint of the second sec							
No. Packa Piska Cription Memory design of the pick of a fination of a finatio a finatio a fination of a finatio a fination of a finatio a fin		0				steps)	
No. Control Control <thcontrol< th=""> <thcontrol< th=""> <thcontr< td=""><td></td><td>Č .</td><td></td><td></td><td></td><td>microns or steps)</td><td></td></thcontr<></thcontrol<></thcontrol<>		Č .				microns or steps)	
30 50 500 520 1000 Set focus position as som 131 0 Particle Accessing Particle Accessing Particle Accessing 133 0 Redick/Accessing Particle Accessing Particle Accessing 134 0 Redick/Accessing Particle Accessing Particle Accessing 135 0 Redick/Accessing Particle Accessing Particle Accessing 136 0 Redick/Accessing Particle Accessing Particle Accessing 137 0 Redick/Accessing Particle Accessing Particle Accessing 138 0 Macc CVDH Particle Accessing Particle Accessing 139 0 Macc CVDH Particle Accessing Particle Accessing 140 0 CVDH Particle Accessing Particle Accessing 141 CVDH Particle Accessing Particle Accessing 142 C CVDH Partin Accessin	129	0	Focus	:FSn#	0 or 1	Set focuser target position (in microns or steps)	
join Point Prote Note Proteom Propagation and Minimed 130 0 Reference of the Section of						Set focuser position as zero	
134 134 <th134< th=""> 134 <th134< th=""></th134<></th134<>	132	0	Focus	:Fh#	[none]	Set focuser target position at half-travel	
358 58 89 9 of 1 Set bad rate: 1-56 77, 6-31 dc, 4-30 dc, 4-							
Image: Part of the second state of the seco						Set baud rate: 1=56.7K, 2=38.4K, 3=28.8K,	
37 50 Mage CVUP MED DYF Cell Immore date Inclusion 128 0 Made CVVP 31.06 and Cell Immore industry 128 0 Made CVVP 31.06 and Cell Immore industry 128 0 Made CVVP 31.06 and Cell Immore industry 128 0 Made CVVP 31.06 and Cell Immore industry 128 0 Not develop, hy Made At Prone position, hy Made Not develop, hy Made 128 0 Not develop, hy Made Cell Immore industry Cell Immore industry 129 0 Seath Seath Return Inf Cell Immore industry 129 0 Seath Seath Return Inf Cell Immore industry 129 0 Seath Seath Seath Seath Seath 129 0 Seath Seath Seath Seath 129 0 <t< td=""><td></td><td>-</td><td></td><td></td><td></td><td>8=2400, 9=1200 :SBn#Reply: 0 or 1</td><td></td></t<>		-				8=2400, 9=1200 :SBn#Reply: 0 or 1	
130 0 Mac CVMac CVMa							
Het Mac OVP On Slight State State State Net Schwing, Property and Property and							
Image: Second	140	0	Misc	:GVP#	On-Step#	Get firmware name	
Image: Second	141	0	Misc	:GU#	sss#	Get Status	H At Home position,
Image: Constraint of the image is a stability of the image is a							
Image: Constraint of the							
13 0 9880" Return: nit Get RAArm Antibackiesh value in ac-seconds 145 0 380n" Set Dec/H backkish in ac-seconds 145 0 380n" Set Dec/H backkish in ac-seconds 146 0 Arm Align fields Set RAA/arm backkish value in ac-seconds 147 0 Arm Align fields Set RAA/arm backkish value in ac-seconds 147 0 Arm Align fields Set RAA/arm backkish value in ac-seconds 148 0 Arm Align fields Set Taka/arm backkish value in ac-seconds 149 0 Arm Align fields Set Taka/arm backkish value in ac-seconds 149 0 C Align fields Set Taka/arm backkish value in ac-seconds 150 0 C C Align fields Align fields Set Taka/arm backkish value in ac-seconds 151 0 C C C Align fields Set Taka/arm backkish value in ac-seconds 152 0 C C C Set Taka/arm backkish value in ac-seconds Set Taka/arm backkish value in ac-seconds 153 0 C	1.40	0		10/ PD#		Deturnu off	RPEC Recorded
145 0 Set RAArm backabh ar s-seconds 147 0 Anii Return modil Align status 147 0 Anii Align (09) stars Shat Telescope Manual Alignment Sequence 148 0 Anii Align (09) stars Shat Telescope Manual Alignment Sequence 149 0 Anii Anii Anii 149 0 Anii Aniii Aniiii Manual Ma				:%BR#			
146 0 A/W Return mode Align n (0 - 9) stars Align n (0 - 9) stars 148 0 Anu/ Align n (0 - 9) stars Sort Telescope Manual Alignment Sequence 148 0 Anu/ Align n (0 - 9) stars Sort Telescope Manual Alignment Sequence 159 0 Align Wite to EEPROM Align Wite to EEPROM 150 0 Align Wite to EEPROM Decresse Return Brighments and the numer 158 0 Align Wite to EEPROM Decresse Return Brighments and the numer 159 0 Decresse Return Structure							
148 0 Align Write to EEPROM 150 0 Align Write to EEPROM 151 0 Bit of the second	146	0		:A?#			Align status
150 0 Align Write to EEPROM 151 0 O O O O O O O O O O O O O O O O O Styrchorize the telescope with the current fight accessor and decination coordinates Styrchorize the telescope with the current fight accessor and decination coordinates Styrchorize the telescope with the current fight accessor and decination coordinates 155 0 O O O Styrchorize the telescope with the current fight accessor and decination coordinates 156 0 ECG# Return: "0x7##" if the mount is moving, other terms at more and tracking turned of for this command to work. 158 0 ESPEASH# Note to caser out (away from objective) Sector and tacking turned of for this command to work. 158 0 ESPEASH# Note to caser out (away from objective) Sector and tacking turned of for this command to work. 159 0 Image: Sector accessor access				:An#		Align n (0 9) stars	Start Telescope Manual Alignment Sequence
15.1 0 Image Image <thimage< th=""> <thimage< th=""> Image<</thimage<></thimage<>							
153 0 Synchonize the telescope with the current right the telescope withe right the telescope with	151	0					Decrease Reticule Brightness
154 0 Br Return: "UX 7H" if the mount is moving, other Synchonize the letescope with the current right assessment and decination coordinates 155 0 ECC str Ech string [G on DebugSer. 157 0 Ech string [G on DebugSer. Wyee flash-OnStep must be at home and marking turned of for this command to work. 158 0 ESP8266 device flash mode OnStep must be at home and marking turned of for this command to work. 159 0 ESP8266 device flash mode OnStep must be at home and marking turned of for this command to work. 159 0 ESP8266 device flash mode OnStep must be at home and marking turned of for this command to work. 159 0 ESP8266 device flash mode OnStep must be at home and marking turned of for this command to work. 160 0 More focuser more rate, where n = 1 tor firset. 2 161 0 More focuser in (twerd divictive) 162 0 Set focuser more rate, where n = 1 tor firset. 2 163 0 Active? 164 0 Active? 165 0 Set focuser takabah amount (in steps or microns) 166 0 Set focuser transfer advecadvecadvecadvecadvecadvecadvecadvec							
Image: control of the sector of the secto	154	0				Return: "N/A#" on success, "En#" on failure v	
166 0 ECd# Echo string [c] on DebugSer. 157 0 ENVRESET# Wipe flash.OnStep must be at home and tracking turned off for this command to work. 158 0 ESPE266 device flash mode OnStep must be at home and tracking turned off for this command to work. 159 0 ESPE266 device flash mode OnStep must be at home and tracking turned off for this command to work. 159 0 Move focuser out (away from objective) 160 Set focuser move rate, where n = 1 for finest. 2 for 0.01mm/second, 3 for 0.1mm/second, 4 for 1mm/second 161 0 Move focuser in (toward objective) 162 0 Active? 163 0 Active? 164 0 Active? 165 0 Set focuser backlash amount (in steps or minicons) 166 1 Set focuser backlash amount (in steps or minicons) 168 0 Set focuser tacklash amount (in steps or minicons) 169 0 Set focuser tacklash amount (in steps or minicons) 169 0 Set focuser temperature compensation of the device of the mode of the device of the mode of the device of the dev		0		10#		Deturn "Ov76" if the mount is moving other	ascension and declination coordinates
Image:	156	0					Echo string [c] on DebugSer.
Image: Seperity of the second of the seco	157	0		:ENVRESET#			
Image: Constraint of the second of the se	158	0					
160 0 Set focuser more rate, where n = 1 for finest, 2 for 0.0 mm/second, 3 for 0.1mm/second, 4 for 1mm/second 161 0 Move focuser in (toward objective) 162 0 Setect primary focuser where [n] = 1 or 2 163 0 Active? 164 0 Active? 165 0 Active? 166 0 Cet primary focuser 167 0 Set focuser backlash amount (in steps or microns) 168 0 Set focuser backlash amount (in steps or microns) 167 0 Set focuser backlash amount (in steps or microns) 168 0 Set focuser temperature compensation where [n] = 0 or 1 169 0 Set focuser temperature compensation or where [n] = 0 or 1 170 0 Set focuser temperature compensation enable 171 0 Set focuser temperature compensation enable 173 0 Set focuser temperature compensation differential 174 0 Return: n# Get focuser temperature compensation deadbale 174 0 Set focuser temperature compensation filterential 175 0 Set focuser temperature compensation f	150	0		:ESPFLASH#			work.
Image: Constraint of the second se							Set focuser move rate, where n = 1 for finest, 2
161 0 Move focuser in (toward objective) 162 0 Select primary focuser where [n] = 1 or 2 163 0 Active? 164 0 Active? 165 0 Cet primary focuser 166 0 Set focuser backlash amount (in steps or microns) 166 0 Set focuser backlash amount (in steps or microns) 167 0 Return: n# 168 0 Set focuser backlash amount (in steps or microns) 168 0 Set focuser tacklash amount (in steps or microns) 168 0 Set focuser tacklash amount (in steps or microns) 169 0 Set focuser temperature compensation coefficient in un per deg. C (+ moves out as temperature compensation coefficient in un per deg. C (+ moves out as temperature fails) 170 0 Set focuser temperature compensation enable status 171 0 Set focuser temperature compensation deadband amount (in steps or microns) 172 0 Set focuser temperature compensation deadband amount (in steps or microns) 173 0 Set focuser temperature compensation deadband amount (in steps or microns) 174 0 Return: n# Get focuser							1mm/second
163 0 Active? 164 0 Active? 165 0 Get primary focuser 166 0 Set focuser backlash amount (in steps or microns) 167 0 Return: n# 168 0 Set focuser backlash amount (in steps or microns) 168 0 Set focuser backlash amount (in steps or microns) 168 0 Set focuser backlash amount (in steps or microns) 169 0 Set focuser temperature compensation there [in] = 0 or 1 170 0 Set focuser temperature compensation coefficient in um per deg. C (+ moves out as temperature compensation coefficient in um per deg. C (+ moves out as temperature compensation coefficient is to user temperature compensation coefficient is to user temperature compensation deadband amount (in steps or microns) 171 0 Set focuser temperature compensation deadband amount (in steps or microns) 172 0 Set focuser temperature compensation deadband amount (in steps or microns) 173 0 Return: n# Get focuser temperature compensation deadband amount (in steps or microns) 174 0 Set focuser temperature compensation deadband amount (in steps or microns) 174 0 Set focuser temperature admenut (in steps or microns)							
165 0 Get primary focuser 166 0 Set focuser backlash amount (in steps or microns) 167 0 Return: n# 168 0 Enable/disable focuser temperature compensation microns) 168 0 Set focuser temperature compensation microns) 169 0 Set focuser temperature compensation microns) 170 0 Set focuser temperature compensation coefficient in um per deg. C (+ moves out as temperature lalls) 171 0 Return: n.m# Get focuser temperature compensation coefficient degree temperature compensation coefficient compensation model status 172 0 Return: n.m# Get focuser temperature compensation deadband amount (in steps or microns) 173 0 Return: n.m# Get focuser temperature compensation deadband amount (in steps or microns) 173 0 Return: n## Get focuser temperature compensation deadband amount (in steps or microns) 174 0 Set focuser temperature compensation (fin steps or microns) 176 0 Return: n## Get focuser temperature filternital 174 0 Return: n## Get focuser temperature filternital 175 0 Return: n##	163	0					Active?
166 0 Set focuser backlash amount (in steps or microns) 167 0 Return: n# Get focuser backlash amount (in steps or microns) 168 0 Enable/disable focuser temperature compensation more [n] = 0 or 1 169 0 Set focuser temperature compensation coefficient in un per deg. C (+ moves out as temperature compensation coefficient in un per deg. C (+ moves out as temperature compensation coefficient in un per deg. C (+ moves out as temperature talls) 170 0 Get focuser temperature compensation coefficient is un per deg. C (+ moves out as temperature compensation coefficient in un per deg. C (+ moves out as temperature talls) 171 0 Get focuser temperature compensation coefficient is un per deg. C (+ moves out as temperature compensation coefficient is un per deg. C (+ moves out as temperature compensation coefficient is un per deg. C (+ moves out as temperature compensation coefficient is un per deg. C (+ moves out as temperature compensation coefficient is un per deg. C (+ moves out as temperature compensation coefficient is un per deg. C (+ moves out as temperature compensation deadband amount (in steps or microns) 172 0 Set focuser temperature compensation deadband amount (in steps or microns) 173 0 Return: n# Get focuser temperature differential 174 0 Return: n# Get focuser temperature of the coust of the position (in microns or steps) 176 Set focuser of tast motion (Get primary focuser
167 0 Get focuser backlash amount (in steps or microns) 168 0 Enable/disable focuser temperature compensation where [n] = 0 or 1 169 0 Set focuser temperature compensation ocefficient in um per deg. C (+ moves out as temperature tails) 170 0 Set focuser temperature compensation ocefficient in um per deg. C (+ moves out as temperature tails) 171 0 Get focuser temperature compensation ocefficient 171 0 Set focuser temperature compensation ocefficient 172 0 Get focuser temperature compensation odelfactor destatus 173 0 Set focuser temperature compensation deadband amount (in steps or microns) 174 0 Return: n# Get focuser temperature compensation deadband amount (in steps or microns) 174 0 Return: n# Get focuser temperature differential 175 0 Set focuser temperature differential Set focuser temperature differential 176 Set focuser temperature differential Set focuser temperature differential 175 Set focuser temperature differential Set focuser temperature differential 177 0 Set focuser temperature differential 178 0 Set focuser temp							Set focuser backlash amount (in steps or
168 0 Enable/disable focuser temperature compensation where [n] = 0 or 1 169 0 Set focuser temperature compensation coefficient in un per deg. C (+ moves out as temperature fails) 170 0 Return: n.# 171 0 Get focuser temperature compensation coefficient in un per deg. C (+ moves out as temperature fails) 171 0 Return: n.# 172 0 Get focuser temperature compensation deadband amount (in steps or microns) 173 0 Get focuser temperature compensation deadband amount (in steps or microns) 173 0 Return: n# Get focuser temperature differential 175 0 Return: n## Get focuser temperature differential 175 Get focuser temperature differential Set focuser temperature differential 176 Return: n# Get focuser temperature differential 177 0 Set focuser temperature differential 177 0 Set focuser temperature differential 177 Get focuser temperature differential Set focuser temperature differential 178 Get focuser temperature differential Set focuser temperature differential 178 Get focuser temperature differential <td>167</td> <td>0</td> <td></td> <td></td> <td></td> <td>Peturn: n#</td> <td>Get focuser backlash amount (in steps or</td>	167	0				Peturn: n#	Get focuser backlash amount (in steps or
169 0 Set focuser temperature compensation coefficient in um per deg. C (+ moves out as temperature lails) 170 0 Get focuser temperature compensation coefficient in um per deg. C (+ moves out as temperature compensation coefficient in um per deg. C (+ moves out as temperature compensation coefficient in um per deg. C (+ moves out as temperature compensation coefficient 171 0 Get focuser temperature compensation coefficient 172 0 Get focuser temperature compensation deadband amount (in steps or microns) 173 0 Set focuser temperature compensation deadband amount (in steps or microns) 174 0 Return: n# Get focuser temperature compensation deadband amount (in steps or microns) 174 0 Return: n# Get focuser temperature differential 175 0 Set focuser temperature differential Set focuser temperature differential 175 0 Set focuser temperature differential Set focuser temperature differential 176 0 Set focuser to fast motion (in microns or steps) Set focuser trapet position at half-travel 178 0 Set focuser target position at half-travel Set focuser target position at half-travel 178 0 Set focuser target position (in microns or steps) Set focuser DC Motor Power Level (i	168	0				Netulii. II#	Enable/disable focuser temperature
Image: set of the set of	169	0					
170 0 Get focuser temperature compensation coefficient 171 0 Get focuser temperature compensation enable status 172 0 Set focuser temperature compensation deadband amount (in steps or microns) deadband amount (in steps or microns) 173 0 Return: n# Get focuser temperature compensation deadband amount (in steps or microns) 173 0 Return: n# Get focuser temperature compensation deadband amount (in steps or microns) 174 0 Return: n# Get focuser temperature compensation deadband amount (in steps or microns) 174 0 Return: n# Get focuser temperature differential 175 0 Set focuser temperature compensation (1nmicrons or steps) 176 Get focuser temperature differential 177 0 Set focuser temperature differential 178 Get focuser tor for fast motion (1nmicrons or steps) 178 Set focuser torget position at half-travel 178 Set focuser temperature differential 179 Return: n# Get full in position (in microns or steps) 180 Return: n# Get max position (in microns or steps) 181 Get Set focuser Level (in %)							coefficient in um per deg. C (+ moves out as
171 0 Get focuser temperature compensation enable status 172 0 Set focuser temperature compensation deadband amount (in steps or microns) 173 0 Return: n# Get focuser temperature compensation deadband amount (in steps or microns) 173 0 Return: n# Get focuser temperature compensation deadband amount (in steps or microns) 174 0 Return: n#temperature in deg. C Get focuser temperature differential 175 0 Set focuser temperature differential Set focuser temperature differential 176 0 Return: n# Get focuser temperature find (in microns or steps) 176 0 Set focuser target position at half-travel 177 0 Set focuser target position at half-travel 178 0 Set focuser target position at half-travel 178 0 Return: n# Get full in position (in microns or steps) 180 0 Return: n# Get focuser tory over tevel (in %) 181 0 O Set focuser DC Motor Power Level (in %)	170	0				Peturn: n n#	Get focuser temperature compensation
172 0 Set focuser temperature compensation deadband amount (in steps or microns) 173 0 Get focuser temperature compensation deadband amount (in steps or microns) 174 0 Return: n# Get focuser temperature compensation deadband amount (in steps or microns) 174 0 Return: n# Get focuser temperature compensation deadband amount (in steps or microns) 176 0 Return: n#temperature in deg. C Get focuser temperature differential 176 0 Set focuser or temperature differential 177 0 Set focuser temperature differential 178 Extern: sn# Get focuser transposition (in microns or steps) 178 Return: n# Set focuser traget position at half-travel 179 0 Return: n# Get full in position (in microns or steps) 180 0 Return: n# Get focuser DC Motor Power Level (in %) 181 0 Set focuser DC Motor Power Level (in %)	171	0					Get focuser temperature compensation enable
Image: constraint of the second se	172	0					
Image: constraint of the second se							deadband amount (in steps or microns)
175 0 Set focuser for fast motion (1mm/s) 176 0 Return: sn# Get focuser current position (in microns or steps) 177 0 Set focuser position as half-travel 178 0 Set focuser position at half-travel 179 0 Return: n# Set focuser target position at half-travel 179 0 Return: n# Get full in position (in microns or steps) 180 0 Return: n# Get max position (in microns or steps) 181 0 Set focuser DC Motor Power Level (in %)		-					deadband amount (in steps or microns)
Return: sn# Return: sn# 177 0 Set focuser position as half-travel 178 0 Set focuser target position at half-travel 179 0 Return: n# Get full in position (in microns or steps) 180 0 Return: n# Get max position (in microns or steps) 181 0 Set focuser DC Motor Power Level (in %)	175	0				Return: n#temperature in deg. C	Set focuser for fast motion (1mm/s)
177 0 Set focuser position as half-travel 178 0 Set focuser target position at half-travel 179 0 Return: n# 180 0 Return: n# 181 0 Set focuser DC Motor Power Level (in %)	176	0				Return: sn#	Get focuser current position (in microns or steps)
IP3 0 Return: n# Get full in position (in microns or steps) 180 0 Return: n# Get max position (in microns or steps) 181 0 Set focuser DC Motor Power Level (in %)							
180 0 Return: n# Get max position (in microns or steps) 181 0 Set focuser DC Motor Power Level (in %)	179	0					
						Return: n#	
				:Fp#			

		commands	
183 0	:FP#	Return: n#	Get focuser DC Motor Power Level (in %)
184 0			Stop the focuser
185 0			Set focuser target position relative (in microns or steps)
186 0	:FSn#		Set focuser target position (in microns or steps)
187 0		Dature M#/far maying) or C#/far atonned)	Set focuser for slow motion (0.01mm/s) Get status
188 0 189 0		Return: M#(for moving) or S#(for stopped) Return: n#temperature in deg. C	Get status Get focuser temperature
190 0		Return: n.n#	Get focuser microns per step
191 0			Set focuser position as zero
192 0 193 0	:GB#	Return: sDD*MM#or sDD*MM'SS#(based or	Get Telescope Altitude Get Fastest Recommended Baud rate
194 0	:Gc#	Return: 24#	Get the current local time format
195 0		Return: sDD*MM#or sDD*MM:SS#(based or	
196 0 197 0	:GDH#	Return: sDD*MM#or sDD*MM:SS#(based or Return: sDD*MM:SS.SSSS#(high precision)	
198 0	:GdH#	Return: sDD*MM:SS.SSS#(high precision)	Get Currently Selected Target Declination
199 0	:GE#	Return: CC#	Get last command error numeric code
200 0		Return: sHH#	Get UTC offset time, the number of decimal hours to add to local time to convert to UTC
201 0		Return: sDDD*MM#	Get Current Site Longitude, east is negative
202 0	:GgH#	Return: sDD*MM:SS.SSS#(high precision)	Get current site Longitude
203 0	:Gh#	Return: sDD*#	Get Horizon Limit, the minimum elevation of the mount relative to the horizon
204 0	:GLH#	Return: HH:MM:SS.SSSS#(high precision)	Get Local Time in 24 hour format
205 0			Get site 1 name
206 0 207 0		Return: E#, W#, N#(none/parked)	Gets the meridian pier-side Get site 2 name
207 0			Get site 3 name
209 0	:Go#	Return: DD*#	Get Overhead Limit
210 0 211 0		Return: s Return: HH:MM.T#or HH:MM:SS#(based on	Get site 4 name Get Telescope RA
211 0	:Gr#	Return: HH:MM.T#or HH:MM:SS#(based on Return: HH:MM.T#or HH:MM:SS (based on	
213 0	:GRH#	Return: HH:MM:SS.SSSS#	Get Telescope RA High Precision
214 0	:GrH#	Return: HH:MM:SS.SSSS#(high precision)	Get Telescope RA
215 0		Return: HH:MM:SS#	Get the Sidereal Time as sexagesimal value in 24 hour format
216 0			Get the Sidereal Time as sexagesimal value in
217 0	:GSa#	Return HH:MM:SS.ss# Return: n.n#(OnStep returns more decimal p	24 hour format, with high precision Get tracking rate, 0.0 unless TrackingSidereal
218 0			Get current site Latitude, positive for North
		Return: sDD*MM#	latitudes
219 0	:GtH#	Return: sDD*MM:SS.SSS#(high precision)	Get current site Latitude, positive for North latitudes
220 0		Return: s#	Get telescope Status
221 0	:Gu#	Return: s#	Get bit packed telescope status
222 0 223 0	:GVM#	Return: MTH DD YYYY# Return: s#(where s is a string up to 16 chars	Get Telescope Firmware Date General Message
223 0		Return: M.mp#	Get Telescope Firmware Number
225 0		Return: s#	Get Telescope Product Name
226 0 227 0	:GW#	Return: HH:MM:SS# Return: [mount][tracking][alignment]#	Get Telescope Firmware Time Get alignment status
227 0	:GXII#	Return: [mount][tracking][alignment]#	Get OnStep value where II is the numeric index
229 0		Return: DDD*MM#or DDD*MM'SS#(based o	r Get telescope azimuth
230 0			Moves telescope to the home position
231 0			Reset telescope at the home position. This position is required for a cold Start.
232 0			Goto the Park Position
233 0 234 0			Set the park position
234 0			Restore parked telescope to operation Clear library (all catalogs)
236 0	:L?#	Return: n#	Get library free records (all catalogs)
237 0			Move to catalog name record
238 0			Find previous object and set it as the current target object
239 0			Set current target object to catalog object number
240 0	:LCn#		Clear current record
240 0		Return: s#(string containing the current target	
242 0			Clear current catalog
243 0			Find next deep sky target object subject to the current constraints.
244 0	:Lon#		Select Library catalog by catalog number n
245 0			Write catalog object information including current
	:LWs#	command returns an empty string "#") a ne	target RA,Dec to next available empty record If at the end of the object list (:LI
246 0		Return: 09, see :MS#	Goto the target Alt and Az
247 0 248 0			Move Telescope East at current guide rate
248 0	:Mgdn#		Pulse guide command where n is the guide time in milliseconds
249 0			Pulse guide command where n is the guide time
250 0	:MGdn#		in milliseconds Move Telescope North at current guide rate
251 0			Goto current RA/Dec but East of the Pier (within
	:MN#	Return: 09, see	meridian limit overlap for GEM mounts)
252 0	:Mp#		Move Telescope for sPiral search at current quide rate
253 0	:MP#	Return: 09, see :MS#	Goto the Current Position for Polar Align
			Move Telescope South at current guide rate
		Return: s#	Goto the Target Object
254 0 255 0 256 0			Move Telescope West at current guide rate
255 0 256 0 257 0			
255 0 256 0 257 0 258 0			Halt all slews, stops goto
255 0 256 0 257 0 258 0 259 0			Halt east Slews
255 0 256 0 257 0 258 0			
255 0 256 0 257 0 258 0 259 0 260 0 261 0 262 0			Halt east Slews Halt southward Slews Halt north Slews Halt westward Slews
255 0 256 0 257 0 258 0 259 0 260 0 261 0 262 0 263 0	:r-#		Halt east Slews Halt southward Slews Halt north Slews Halt westward Slews Disable derotator
255 0 256 0 257 0 258 0 259 0 260 0 261 0 262 0			Hait east Slews Hait southward Slews Hait north Slews Hait westward Slews Disable derotator Move increment where n = 1 for 1 degrees, 2 for
255 0 256 0 257 0 258 0 259 0 260 0 261 0 262 0 263 0 264 0 265 0	r-# :rh# :rh#		Halt east Slews Halt southward Slews Halt north Slews Disable derotator Move increment where n = 1 for 1 degrees, 2 for 2 degrees, 3 for 5 degrees, 4 for 10 degrees Enable derotator
255 0 256 0 257 0 258 0 259 0 260 0 261 0 262 0 263 0 264 0 265 0	:rn# :r+#		Halt east Slews Halt southward Slews Halt north Slews Halt westward Slews Disable derotator Move increment where n = 1 for 1 degrees, 2 for 2 degrees, 3 for 5 degrees, 4 for 10 degrees
255 0 256 0 257 0 258 0 259 0 260 0 261 0 262 0 263 0 264 0	:m#		Halt east Slews Halt southward Slews Halt north Slews Disable derotator Move increment where n = 1 for 1 degrees, 2 for 2 degrees, 3 for 5 degrees, 4 for 10 degrees Enable derotator

		CO	mmands	
268 0				Set Axis1 Guide rate to n.n degrees per sidereal
	:RAn.n#			second
269 0	:rbn#			
270 0	:rb#		Return: n#	Get rotator backlash amount in steps
271 0	:rC#			Set slew rate to Centering rate 8X
272 0 273 0	.1C#			Moves rotator to the home position Set continuous move mode (for next move
213 0	:rc#			command)
274 0	:rD#		Return: n.n#	Get rotator degrees per step
275 0				Set Axis2 Guide rate to n.n degrees per sidereal
	:REn.n#			second
276 0	:RF#			Set slew rate to Fast Rate 48X
277 0	:rF#			Reset rotator at the home position
278 0	0.11		B	Set slew rate to Guiding Rate 1X
279 0 280 0	:rG# :rI#		Return: sDDD*MM# Return: n#	Get rotator current position in degrees Get mIn position (in degrees)
280 0	.11#		Return. n#	Set slew rate to Find Rate 20X
282 0	:rM#		Return: n#	Get Max position (in degrees)
283 0				Set slew rate to n, where $n = 09$
284 0	:rP#			Move rotator to the parallactic angle
285 0	:rQ#			Stops movement (except derotator)
286 0	:rR#			Reverse derotator direction
287 0	:rSsDDD*MM'SS#			
288 0				Set slew rate to Half Max (VF)?X (1/2 of
				maxRate)
289 0	:rT#		Return: M#(for moving) or S#(for stopped)	Get status
290 0	:SasDD*MM#			Cat David Data where a is an ACOU divit (1, 0)
291 0				Set Baud Rate where n is an ASCII digit (19) with the following interpertation
292 0				
293 0	:SdsDD*MM:SS.SSS#			
294 0	:SdsDD*MM:SS#			
295 0	:SdsDD*MM#			
296 0	:Sg(s)DDD*MM:SS.SSS#			
297 0	:Sg(s)DDD*MM:SS#			
298 0	:Sg(s)DDD*MM#			
299 0	:SGsHH:MM#(where MM	is 30 or 45)#		
300 0	:SGsHH#			
301 0	:ShsDD#			
302 0	:SLHH:MM:SS.SSS# :SLHH:MM:SS#			
303 0 304 0	:SMs#			
304 0	:SNs#			
306 0	:SoDD#			
307 0	:SOS#			
308 0	:SPs#			
309 0	:SrHH:MM:SS.SSSS##			
310 0	:SrHH:MM:SS##			
311 0	:SrHH:MM.T##			
312 0				
313 0	:STH.H#			Set Tracking Rate in Hz where 60.0 is solar rate
314 0	:StsDD*MM:SS.SSS##			
315 0	:StsDD*MM:SS##			
316 0	:StsDD*MM##			
317 0	0/// //			Set OnStep value where II is the numeric index
318 0	:SXII,n# :SXA1,R#			and n is the value to set (possibly floating point) to revert an axis to defaults
319 0	 :SXAC,n#			to switch between compile and run-time settings
320 0	:SXAn:#			Set axis settings :SXA1,#
321 0	:SzDDD*MM#			Set unis settings .Svi (1,
322 0				Master sidereal clock slower by 0.02 Hertz
				(stored in EEPROM)
323 0				Master sidereal clock faster by 0.02 Hertz
				(stored in EEPROM)
324 0	:T1#			Track single axis (disable Dec tracking on Eq
325 0	:T1# :T2#			mounts) Track dual axis
326 0				Tracking disable
327 0				Tracking enable
328 0				Track rate king
329 0				Track rate lunar
330 0				Track refraction disable
331 0	:To#			OnTrack enable
332 0				Track rate sidereal
333 0				Master sidereal clock reset (to calculated
1 1				sidereal rate, stored in EEPROM) Track refraction enable
224 0				TTACK TETTACTION ENABLE
334 0 335 0				Track rate solar
335 0				Track rate solar
335 0 336 0	:VH#		Return: n#	Toggle between low/hi precision positions
335 0 336 0 337 0	:VH#		Return: n#	
335 0 336 0	:VH# :VRn#		Return: n# Return: sn#	Toggle between low/hi precision positions PEC index sense position in seconds Read PEC table entry rate adjustment (in steps +/-) for worm segment n (in seconds)
335 0 336 0 337 0	:VRn#		Return: sn#	Toggle between low/hi precision positions PEC index sense position in seconds Read PEC table entry rate adjustment (in steps +/-) for worm segment n (in seconds) Read out RA PEC ten byte frame in hex format
335 0 336 0 337 0 338 0 339 0				Toggle between low/hi precision positions PEC index sense position in seconds Read PEC table entry rate adjustment (in steps +/-) for worm segment n (in seconds) Read out RA PEC ten byte frame in hex format starting at worm segment n (in seconds)
335 0 336 0 337 0 338 0	:VRn#		Return: sn#	Toggle between low/hi precision positions PEC index sense position in seconds Read PEC table entry rate adjustment (in steps +/) for worm segment n (in seconds) Read out RA PEC ten byte frame in hex format starting at worm segment n (in seconds) Read PEC table entry rate adjustment (in steps
335 0 336 0 337 0 338 0 339 0	:VRn#		Return: sn# Return: x0x1x2x3x4x5x6x7x8x9#(hex one b)	Toggle between low/hi precision positions PEC index sense position in seconds Read PEC table entry rate adjustment (in steps +/-) for worm segment n (in seconds) Read out RA PEC ten byte frame in hex format starting at worm segment n (in seconds) Read PEC table entry rate adjustment (in steps +/-) for currently playing segment and its rate
335 0 336 0 337 0 338 0 339 0 340 0	:VRn#		Return: sn#	Toggle between low/hi precision positions PEC index sense position in seconds Read PEC table entry rate adjustment (in steps +/-) for worm segment n (in seconds) Read out RA PEC ten byte frame in hex format starting at worm segment n (in seconds) Read PEC table entry rate adjustment (in steps +/-) for currently playing segment and its rate adjustment (in steps +/-)
335 0 336 0 337 0 338 0 339 0	:VRn#		Return: sn# Return: x0x1x2x3x4x5x6x7x8x9#(hex one by Return: sn,n#	Toggle between low/hi precision positions PEC index sense position in seconds Read PEC table entry rate adjustment (in steps +/) for worm segment n (in seconds) Read out RA PEC ten byte frame in hex format starting at worm segment n (in seconds) Read PEC table entry rate adjustment (in steps +/) for currently playing segment and its rate adjustment (in steps +/-) PEC number of steps per second of worm
335 0 336 0 337 0 338 0 339 0 340 0 341 0	:VRn# :Vrn#		Return: sn# Return: x0x1x2x3x4x5x6x7x8x9#(hex one b)	Toggle between low/hi precision positions PEC index sense position in seconds Read PEC table entry rate adjustment (in steps +/-) for worm segment n (in seconds) Read out RA PEC ten byte frame in hex format starting at worm segment n (in seconds) Read PEC table entry rate adjustment (in steps +/-) for currently playing segment and its rate adjustment (in steps +/-) PEC number of steps per second of worm rotation
335 0 336 0 337 0 338 0 339 0 340 0	:VRn# :Vrn#		Return: sn# Return: x0x1x2x3x4x5x6x7x8x9#(hex one b) Return: sn,n# Return: n.n#	Toggle between low/hi precision positions PEC index sense position in seconds Read PEC table entry rate adjustment (in steps +/) for worm segment n (in seconds) Read out RA PEC ten byte frame in hex format starting at worm segment n (in seconds) Read PEC table entry rate adjustment (in steps +/) for currently playing segment and its rate adjustment (in steps +/-) PEC number of steps per second of worm
335 0 336 0 337 0 338 0 339 0 340 0 341 0 342 0	:VRn# :Vrn# :VS# :VS#		Return: sn# Return: x0x1x2x3x4x5x6x7x8x9#(hex one by Return: sn,n# Return: n.n# Return: n.m#	Toggle between low/hi precision positions PEC index sense position in seconds Read PEC table entry rate adjustment (in steps +/-) for worm segment n (in seconds) Read out RA PEC ten byte frame in hex format starting at worm segment n (in seconds) Read PEC table entry rate adjustment (in steps +/-) for currently playing segment and its rate adjustment (in steps +/-) PEC number of steps per second of worm rotation PEC number of steps per worm rotation
335 0 336 0 337 0 338 0 339 0 340 0 341 0 342 0 343 0 344 0 3445 0	:VRn# :Vm# :VS# :VW# :VW#		Return: sn# Return: x0x1x2x3x4x5x6x7x8x9#(hex one by Return: sn,n# Return: n.n# Return: n.m#	Toggle between low/hi precision positions PEC index sense position in seconds Read PEC table entry rate adjustment (in steps +/-) for worm segment n (in seconds) Read out RA PEC ten byte frame in hex format starting at worm segment n (in seconds) Read PEC table entry rate adjustment (in steps +/-) for currently playing segment and its rate adjustment (in steps +/-) PEC number of steps per second of worm rotation PEC number of steps per worm rotation Queries current site Sets current site to n, where n = 03 Move PEC Table back by one second
335 0 336 0 337 0 338 0 339 0 340 0 341 0 342 0 343 0 344 0	:VRn# :Vm# :VS# :VV# :WV# :W7# :W7#		Return: sn# Return: x0x1x2x3x4x5x6x7x8x9#(hex one by Return: sn,n# Return: n.n# Return: n.m#	Toggle between low/hi precision positions PEC index sense position in seconds Read PEC table entry rate adjustment (in steps +/-) for worm segment n (in seconds) Read out RA PEC ten byte frame in hex format starting at worm segment n (in seconds) Read PEC table entry rate adjustment (in steps +/-) for current by playing segment and its rate adjustment (in steps +/-) PEC number of steps per second of worm rotation PEC number of steps per worm rotation Queries current site Sets current site to n, where n = 03 Move PEC Table back by one second Write PEC table entry for worm seconent In (in
335 0 336 0 337 0 338 0 339 0 340 0 341 0 342 0 343 0 344 0 3445 0	:VRn# :Vm# :VS# :VV# :WV# :W7# :W7#		Return: sn# Return: x0x1x2x3x4x5x6x7x8x9#(hex one by Return: sn,n# Return: n.n# Return: n.m#	Toggle between low/hi precision positions PEC index sense position in seconds Read PEC table entry rate adjustment (in steps +/-) for worm segment n (in seconds) Read out RA PEC ten byte frame in hex format starting at worm segment n (in seconds) Read PEC table entry rate adjustment (in steps +/-) for currently playing segment and its rate adjustment (in steps +/-) PEC number of steps per second of worm rotation PEC number of steps per worm rotation Queries current site Sets current site to n, where n = 03 Move PEC Table back by one second

commands