

Ioline Plotter Language Syntax

Ioline equipment includes a custom parser for syntax commands allowing customization and control. These commands are listed and explained below. Please read the reference sections for help with using Ioline Syntax.

Contents

Page

Command List

Plotter Set Up and Verification Commands	4
AB Abort Registration	4
AE Auto Loop Enable	4
AL Set Auto Loop Distance	4
AS Set Speedy Corner Angle	4
CL Set Cut Out Cut and Space Length	4
CO Cut Out Mode Enable	4
CS Calibrate Countour Cutter Sensor	4
DC Send Registration Points	4
DD Output Current Settings	4
DR Dump Receive Buffer	6
EB Manual Control of Pen/Knife Status	6
ES Enable Plotter Frame Sensor	6
FB Flush Buffers	6
FE Set Final Pen Force	6
FG Set Frame Gap	6
FS Frames To Skip Sensor Mark Procedure	6
HO HP-GL Origin Select	6
KA Set Knife Steering Angle	7
KO Set Knife Offset Distance	7
KV Set Knife Overcut Distance	7
MC Reset MicroCalibration Factors	7
MS Manual Move Speed Select	8
OK Output Knife Status	8
OS Output Curve Smoothing Status	8
OT Output Date and Time of Compile	8
PL Set Pouncing Cut and Space Distances	8
PN Pouncing Mode Enable	8
PO Start/Stop Machine	8
PT Page Type Select	9
PZ Page Size Select	9
RA Registration Adjustment	9
RB Registration Begin	9
RE Roll Feed Toggle	9
RO Registration Output	10
SA Set Acceleration	11
SD Set Pen Up and Pen Down Delays	11

SF	Set Force	11
SM	Curve Smoothing Enable	11
SO	Display Sensor Calibration Values	11
SR	Set Plot Rotation	11
SS	Set Scaling Factors for X and Y	11
SU	Set Pen Up Max Speed	11
SV	Set Pen Down Speed	11
TD	Dump Tangential Cutting Variables	11
TE	Tangential Cutting Enable	11
TF	Set Tangential Cutting Force	11
TL	Set Length of Tangential Move	11
UD	Set Current Settings To Power-up Defaults	11
Plotter Identification Commands		12
IC	Output Plotter ID Code	12
RV	Output Part Number and Revision of Device	12
VR	Output Internal Plotter ROM Version Number	12
Plotter Testing Commands		13
AT	Automatic Serial Test	13
MF	MegaBuffer Test	13
TC	Calibration Plot	13
TM	Burn-in Test Plot	13
TP	HP-GL Test Plot	13
Plotting Commands		14
DP	Put Pen Down	14
FR	Frame Advance	14
GT	Go To Position	14
HM	Move To Home Position	14
LF	Signcutter Loop Feed	14
LL	Set Lower Left	14
UP	Put Pen Up	15
UR	Set Upper Right	15
XY	Output Current Position	15
Plotting Commands No Longer Supported		17
Reference		20
	Abbreviations	20
	Using Ioline Syntax	20
	Using a Terminal Program to Communicate with an Ioline Plotter	21

Command List

Plotter Set Up and Verification Commands

- AB** Abort registration of a device.
- AEx** Auto loop enable, for the Signcutter machines. AE0 disables the auto loop feature. AE1 enables the auto loop feature. If auto loop is enabled, the plotter will automatically provide a service loop during plotting. The service loop size is set by the **AL** command.
- Example: **! AE0 @ .** Disables auto loop.
- ALx** Set auto loop distance in mils, for the Signcutter machines. The auto loop distance only applies if the auto loop feature is enabled. The plotter will automatically provide a service loop of *n* length during plotting. Subsequent loop feeds will occur when an *x* move exceeds the current loop.
- Example: **! AL40000 @ .** Sets loop size to 40 inches (40000 mils).
- ASx** Set speedy corner angle, where *x* is the angle in degrees. The speedy corner angle can not be set to a negative number.
- Example: **! AS3 @ .** Sets the Speedy Corner angle to 3 degrees.
- CLx,y** Set cut out cut length and cut out space length in mils, for the Signcutter. The cut out cut, and space length apply only if the cut out feature is enabled which is done with the **CO** command. The plotter will automatically cut for distance *x* and then leave a blank of distance *y* as if follows the path of the cut file during plotting.
- Example: **! CL40,600 @ .** Sets the cut length to 40 mils and the space length to 600 mils.
- COx** Cut out Enable for the Signcutter. CO1 enables the cut out feature. The cut out cut and space lengths are set by the **CL** command.
- Example: **! CO0 @ .** Disables cut outs.
- CS** Calibrate sensor for use with the Countour Cutter. Calibration is done internally, and thus, requires no parameters to complete.
- DC** Send registration points.
- DD** Data dump of device. Outputs the current settings in ASCII format. A linefeed character is output between each number, no spaces or tabs are added, and the output numeric format is integer unless otherwise specified. **DD** causes the following data to be output, in this order:

*For a Signcutter, **DD** causes the following data to be output, in this order:*

Setting:	Values:	
HPGL origin.	0 = Lower Left, 1= Center	<lf>
Custom X Frame Size	0 to 838000 mills	<lf>
Custom Y Frame Size	0 to 30000, 40000, or 52000 mills (depending on model)	<lf>
X scaling factor, integer percent	0 to 999 percent (separate from language scaling).	<lf>
Y scaling factor, integer percent	0 to 999 percent (separate from language scaling).	<lf>
Knife-up delay	in mSec.	<lf>
Knife-down delay	in mSec.	<lf>
Acceleration	in 10ths of Gs.	<lf>
Knife minimum force	in grams.	<lf>
Knife maximum force	in grams.	<lf>
Knife tip offset	in mills.	<lf>
Knife overcut distance	in mills.	<lf>
Knife lift angle	in degrees.	<lf>
User X calibration factor,	floating-point format.	<lf>
User Y calibration factor,	floating-point format.	<lf>
Factory X calibration factor,	floating-point format.	<lf>
Factory Y calibration factor,	floating-point format.	<lf>
Auto Loop control	0 = disabled, 1 = enabled	<lf>
Auto Loop size	in mills	<lf>
Pounce control	0 = disabled, 1 = enabled	<lf>
Cut Out control	0 = disabled, 1 = enabled	<lf>
Cut Out cut length	in mills	<lf>
Cut Out space length	in mills	<lf>
Pounce cut length	in mills	<lf>
Pounce space length	in mills	<lf>

Maximum plot width in mills <lf>

End of string, *an underscore* _.

DR Dump receive buffer. Outputs the previously sent commands in ASCII format.

EB_x Pen/knife status and control. EB displays the current pen/knife status as well as allows for manual user control:

Example: **!EB @** . Outputs current status.

Example: **!EB1 @** . Always knife.

Example: **!EB2 @** . Always pen.

ES_x Enable frame sensor, for the *Pen Plotters only*. ES0 disables the sensor. ES1 enables the sensor. If the frame sensor is enabled, then the frame gap distance is ignored. When the sensor is disabled, the frame gap is added to the frame advance to produce a gap or an overlap.

Example: **!ES0 @** . Disables the sensor.

FB Flush buffers by initializing them. The beeper will sound twice when flushing the buffers is complete.

FG_x Set frame gap distance in mils for the Pen Plotters. The frame gap is added to the frame advance to produce a gap or an overlap. A positive number gives a gap and a negative number gives an overlap. The range is -1000 to 1000 mils. For the Summit and Plotters, the frame gap is enabled only when the sensor is disabled. For the Studio 40", the frame gap is enabled only when the roll feed is enabled.

Example: **!FG-1000 @** . Gives a 1000 mil (1 inch) overlap.

FS_x Frames to skip sensor mark procedure, where x is the number of frames.

Example: **!FS10 @** . Skips 10 frames.

HO_x HP-GL Origin select only affects HP-GL. The 'origin' is the (0, 0) coordinate. **HO0** selects the origin at the center of the page, as used (for example) in the HP7585 and HP7596 Plotters. **HO1** selects the origin at the lower left corner of the page, as used, for example, in the HP7475 plotter. This command affects the ID code output for the IOLS and HP-GL **OI** instructions and the HP-GL **ESC.A** command. When lower-left origin is selected, the ID code is "7545". When center-origin is selected, the present ID code is "7596". This allows software to auto-configure for the different styles of HP Plotters. The **HO** command does not affect DM-PL.

Example: **!HO0 @** . Selects 7596 emulation mode.

KA_x Set the knife (steering) angle in degrees for the Signcutter. Ignored by other models. This is the angle threshold for knife steering. Turns larger than this angle get a steering-arc performed; turns smaller than this do not get a steering-arc. This angle is limited to the range of 0 to 180 degrees.

Example: **! KA30 @** . Sets a 30 degree steering angle threshold.

KO_x Offset distance in mils for the Signcutter, ignored by other models. This tells the cutter the distance between the center of knife rotation and the knife tip. This must be set according to the offset distance built into the blade.

Example: **! KO17 @** . Sets 17 mils blade offset.

KV_x Overcut distance in mils for the Signcutter. Ignored by other models. The last knife-down cut is extended by this distance, in order to make the last cut of a figure cross the first cut.

Example: **! KV10 @** . Sets 10 mils of overcut.

MC Reset user *X* & *Y* MicroCalibration factors to 1.000, 1.000.

MC_{x,y} Multiply user *X* & *Y* MicroCalibration factors by *x* and *y*. These values are normally set to 1.00000; they are changed when a user wants to calibrate the plotter. For example, if a 20 inch plot is sent to the plotter and the result measures 19.9 inches in the *X*-axis, the user can adjust for this. The *X* and *Y* factors are equal to the quotient of the **intended value/measured value**. In the above example the *X*-factor would be 20.0/19.9=1.0050. (**!MC1.0050,1.000@**) See also the TC command.

Successive uses of the **MC_{x,y}** command multiply the old MicroCalibration factors by the new MicroCalibration factors, and use the results as the new factors. So, sending **! MC1.1,1.1 @ ! MC1.1,1.0 @** will set MicroCalibration factors of 1.21, 1.1. This arrangement allows fine adjustment of existing MicroCalibration factors. If you want to directly set the values of the MicroCalibration factors, send **! MC @** first to reset the factors to 1.0, 1.0 and then send **! MC_{x,y} @** to set the new factors.

The allowed range of the *X* and *Y* values is 0.20000 to 5.00000, or 1/5 scaling to 5/1 scaling.

MC sets User MicroCal for the currently selected user **only** in the Signature 5000.

We recommend using about 5 decimal places, which will provide 0.001" resolution for a 100.000" plot. Any more decimals would be wasted. The standard values for user-microcal are 1.00000, 1.00000.

Important: This command does not affect factory MicroCalibration. Do not tamper with factory MicroCalibration; you can seriously interfere with your plotter's operation.

Example: **! MC2.00000,2.00000 @** . Multiplies the current X and Y calibration factors by 2.

MS_x Manual-move speed select. Sets speed in tenths of centimeters / second for the **MM_{x,4}** command.

Example: **! MS45 MM1,4 <pause> @** . Moves the carriage at 4.5 cps in the +Y direction, until the @ character is sent.

OK Output knife status for the Signcutters.
0 = Knife not installed
1 = Knife installed

Example: **! OK @** . Displays current knife status as a 1 or 0.

OS Output curve smoothing status for the Signcutter. See **SM** command for setting the curve smoothing option.
0 = Curve smoothing not enabled.
1 = Curve smoothing enabled.

Example: **! OS @** . Displays curve smoothing status as a 1 or 0.

OT Output the date and time of that the firmware was compiled.

Example: **! OT @** . Displays the last time the firmware was compiled.

PL_{x,y} Pouncing length and space setting command for the Signcutter. The pouncing cut, and space length apply only if the pouncing feature is enabled which is done with the **PN** command. The plotter will automatically cut for distance *x* and then leave a blank of distance *y* as it follows the path of the cut file during pouncing.

Example: **! PL50,200 @** . Sets the pounce length to 50 mils and the spaces to 200 mils.

PN_x Pouncing enable for the Signcutter. PN1 enables the pouncing feature. The pouncing cut and space lengths are set by the **PL** command.

Example: **! PN0 @** . Disables cut outs.

PO Emulates a press of the Start/Stop button, which will take the machine from Online to Offline. The user must manually press the Start/Stop button to bring the device back Online.

PT_x Page type *x* select. Part of the sequence for selecting a new page size. **Use this only in the specified format and sequence.** See the **UR** command.

Chooses which set of standard paper sizes are used for standard sizes, or indicates that a custom size is to be selected.

0 = ENGR (Engineering) sizes
1 = ARCH (Architectural) sizes
2 = ISO (International) sizes
3 = CUSTOM size

PT0 to **PT2** are setups for the **PZ** command; **PT3** is a setup for **LL** and **UR**. This command **MUST** precede **PZ** or **LL** and **UR**. See **PZ** or **LL**, **UR** for examples.

Example: **! PT0 @ .** Selects the Architectural sizes as the page type.

PZ_x Select a standard page size. **Must** be preceded by **PT0** , **PT1** , or **PT2** , to select the set of standard sizes we are choosing from:

PZ0 selects A-size.	PZ5 selects A4-size.
PZ1 selects B-size.	PZ6 selects A3-size.
PZ2 selects C-size.	PZ7 selects A2-size.
PZ3 selects D-size.	PZ8 selects A1-size.
PZ4 selects E-size.	PZ9 selects A0-size.

Example: **! PT0 PZ2 @ .** Selects an Engineering C-size sheet.

RA Registration adjustment properties. Returns the current registration adjustment properties.

RA_{x,y} Registration adjustment of a device. Adjusts the current registration properties by setting them to X and Y.

Example: **! RA1,1 @ .** Adjusts the registration parameters by 1 and 1.

RB_{x,y} Begin registration of a device. Parameters set the current registration point.

Example: **! RB1,1 @ .** Adjusts the registration parameters to 1 and 1.

RE_x Roll feed toggle.

For the Studio Models: **RE0** = Cut Sheet, **RE1** = Roll Feed.

When roll feed is enabled, **RF1**, the plotter does a service loop at the end of each frame advance command. The size of the service loop is equal to the X page size.

For the Signature 5000: **RE0** = Cut Sheet, **RE1** = Roll Feed On and do an immediate service loop to initialize the supply roll, **RE2** = Roll Feed On without immediate service loop. When roll feed is enabled, **RE1** or **RE2**, the plotter does a service loop at the end of each frame advance command. The size of the service loop is equal to the loop size parameter.

Example: **! RE1 @ .** Turns on Roll Feed.

RO Registration output of a device. Outputs the current registration parameters.

SA_x Set acceleration to *x*, in tenths of gravities.

Example: **!SA30 @** . Selects 3.0G acceleration.

SD_{x,y} Set pen up/down delays in milliseconds to x, y. Standard values are 0 and 6 milliseconds for pen-plotting, and 50 and 50 msec for knife cutting (which takes a long time to penetrate / lift clear of the thick vinyl or sandblast material).

Example: **!SD0,6 @** . Sets the standard delays for pen-plotting.

Example: **!SD50,50 @** . Sets the standard delays for knife-cutting.

SF_x Set force in grams. The **SF_x** form of this command sets the force of the Signature 5000. This command is **not** overridden by the Speed Override setting.

SF_{x,y} For all other machines, set the low/high force settings of the keypad force knob. The x parameter sets the low force setting and the y parameter sets the high force setting. The high and low settings define the range of the keypad force knob.

Example: **!SF24 @** . Sets 24 grams force for machines without force knobs.

Example: **!SF100,200 @** . Sets the range of the force knob from 100 to 200 grams.

The Studio 7 plotter ignores this command. The Studio 7 sets its knife force by adding weights to the carriage mechanism.

SM_x Set curve smoothing option for the Signcutter. SM1 enables curve smoothing and SM0 disables curve smoothing. See **OS** command for checking current curve smoothing status.

Example: **!SM1 @** . Enables curve smoothing.

SO Display sensor calibration values. Output is displayed as w,x,y,z where:

w = X Sensor Calibration Forward
x = X Sensor Calibration Backwards
y = Y Sensor Calibration Forward
z = Y Sensor Calibration Backwards

Example: **!SO @** . Displays sensor calibration values.

SR_x Set plot rotation by x, where:

x = 0 Sets Plot Rotation By 0 Degrees
x = 1 Sets Plot Rotation By 90 Degrees
x = 2 Sets Plot Rotation By 180 Degrees
x = 3 Sets Plot Rotation By 270 Degrees

Example: **!SR1 @** . Sets the plot rotation to 90 degrees.

- SS_{x,y}** Set scale value (percent) for *X* and *Y* axes to *x* and *y*. Default scaling is 100 for 100%, full-size. Range is currently 0 to 1000%, for zero-size to 10X. *X* and *Y* may be scaled independently.
- Example: **!SS25,25 @ .** Sets up 25% scaling, for testing a large plot on small (inexpensive) paper.
- SU_x** Set pen up maximum speed in cm/sec. The range limited between 1 and the constant hard-coded for the specific machine for units in cm/sec.
- Example: **!SU1 @ .** Sets pen up maximum speed to 1 cm/sec.
- SV_{x,y}** Set the minimum and maximum pen down knob speed in cm/sec, where *x* is the minimum and *y* is the maximum speed.
- Example: **!SV1,2 @ .** Sets minimum pen down knob speed to 1cm/sec and the maximum pen down knob speed to 2 cm/sec.
- TD** Displace the current values of the tangential cutting variables. The output is *x,y,z* where:
- x* = Tangential Enabled (0 for No, 1 for Yes)
y = Tangential Force
z = Tangential Length
- TE_x** Enable or disable tangential cutting. *x* = 1 enables tangential cutting, while *x* = 0 disables it.
- Example: **!TE1 @ .** Enables tangential cutting.
- TF_x** Set force for tangential cutting, where *x* is the force in grams. The tangential cutting force must be between 1 and 100 grams.
- TL_x** Set length of move in tangential cutting, where *x* is the length in mils. The length of move in tangential cutting must be between 10 and 200 mils.
- Example: **!TL20 @ .** Sets the length of tangential cutting to 20 mils.
- UD** For all machines, save the current parameters as the new power-up defaults. The plotter normally holds settings only until power is switched off, allowing resetting of most settings by turning the plotter off and on again. This way, for example, a scaled & rotated plot can in all cases be followed by a normally scaled plot if you turn the plotter off to clear it.

Plotter Identification Commands

IC Output plotter's ID code. Example response codes are:

MACHINE	Product ID
SC98	32
600Ae	33
28Ae	34
300 (Obsolete)	36
200	39
600AeX	40
28AeX	41
350 HF	42
100 (Obsolete)	46
300	47
I/S ColdFire 500g	48
I/S ColdFire 400g	49
I/S Contour	50
Studio A	53
100	54

Example: **!IC @** . Responds with 50<cr> for the I/S Contour.

RV Return the manufacturing part number and revision of the device.

Example: **!RV @** . Displays the PN and REV of device.

VR Transmits the Engineering (Internal) ROM version & revision numbers, in the following format: the 1- or 2-digit version number, a decimal point, the 2-digit revision number, and a terminating line-feed character.

Example: **!VR @** . Gets a response of 2.10<cr> for a device with an Internal firmware of 2.10.

Plotter Testing Commands

AT x Automatic Serial Test. Returns x , where x is a valid ASCII character value (33-126) which corresponds to basic letters, numbers, and other symbols.

Example: **!AT44 @ .** Returns the integer value 44.

MF x Example: **!MF0 @ .** Output total Mega Buffer RAM sensed.
Mega Buffer Function command. Tests for Mega Buffer. NOT normally used during normal operation. Currently only the Mega Buffer RAM sensed test is allowed.

Example: **!MF0 @ .** Output total Mega Buffer RAM sensed.

TC Plot a Calibration plot, used to set up MicroCal factors for MC command. See the user manual.

TC x For the Studios, Classics, Super 88s, Summits, and Plotters, plot the Calibration Plot in English units ($x = 0$) or Metric units ($x = 1$).

TM Perform a manufacturing burn in test.

Example: **!TM @ .**

TP Perform a Test Plot. Plots out the standard HPGL test-plot.

Example: **!TP @ .**

Plotting Commands

DP Down pen. Puts the pen down. Note that the pen will always be lifted when you exit IOLS; the standard IOLS usage of **! DP @! GT100,1000 @! UP @** cannot be used to draw a line, so send **! DP GT100,1000 UP @** instead. Be careful; this command can interfere with plots.

Example: **! DP GT100,1000 UP @ .**

FR Frame advance. Performs a frame advance using the current page size settings.
FRx Frame advance using specified *x* distance in mils.

Example: **! FR @ .** Frame advance page length
! FR46000 @ . Frame advance 46 inches

GT_{x,y} Go to position (*x*, *y*). Moves the pen to the specified *X*, *Y* position. *X* and *Y* are specified in mils, 1000 mils per inch, 1mil = 0.001". This move can be done with the pen either up or down; pen down draws a line. Note that exiting IOLS lifts the pen, so you must do **! DP GT1000,1000 UP @** instead of **! DP @ ! GT1000,1000 @ ! UP @** to actually draw a line.

Example: **! GT15000,20000 @ .** Moves the pen to X=15.000 inches, Y=20.000 inches.

HM Move to Home position. Lifts the pen, moves to the current lower-left position, leaves the pen up.

Example: **! HM @ .**

LF Loop feed, for the Signcutter, using the current page size at 13 cm/s.
LF_{x,y} Loop feed, using the specified *x* distance in mils at the specified *y* speed in centimeters per second. A loop feed slowly pulls a length of new media off the feed roll and returns to the current position. This creates a service loop for plotting a new plot. A service loop prevents jerking the new media off the feed roll at high speeds.

The Summit ignores this command.

! LF20000,10@ . Moves out 20,000 mils (20 inches) and back at 10 cm/s

LL Lower left at current position.
LL_{x,y} Lower left at (*x*, *y*) position. The **LL** form is the same as pressing the "LowLeft" key on the menu; it sets the lower-left corner of the media to the current pen position. Often followed by **UR** command; if so, it **must** be preceded by the **PT3** command. See **UR** command for full documentation of this sequence.

Example: **! LL @ .**

The **LL_{x,y}** form is the same as a **GT_{x,y}** followed by **LL**. It moves to the specified **X,Y** position (relative to the previous lower-left corner position), and then sets the lower-left corner of the media to that position. The **X** and **Y** distances may be positive, zero, or negative.

Example: **! LL-1000,0 @ .** Moves the lower-left corner NEGATIVE one inch in the *x* direction.

Example: **! LL1000,-400 @ .** Moves the lower-left corner POSITIVE one inch in the *x* direction, and NEGATIVE 0.400 inches in the *y* direction.

UP Put pen up. Will put the pen up, and leave it there.

Example: **! UP @ .** Puts the pen up, and leaves it there.

UR Set upper right corner of media to the current pen position.
UR_{x,y} Set upper right corner of media to (*x,y*) relative to the previously selected (using the **LL** command) Lower-Left point. Used as the final part of the **PT3 LL(*x,y*) UR_{x,y}** sequence.

This sequence assumes that the current pen-position is set to the new lower-left corner of the media. This also assumes you have set **AV** the way you want it, on or off.

First, select a Custom paper-size with **PT3**.

Next, send the **LL** command to set lower-left to the current point.

Finally, use the **UR_{x,y}** command to set the upper-right corner of the media to (*x,y*) relative to the **LL** point.

Example: **! PT3 LL UR5000,10000 @ .** Selects a Custom paper-size, with lower-left at the current position, 5.000" wide, 10.000" high.

XY Output the actual *X,Y* pen position, as measured by the encoders.

X and *Y* are in mils relative to the lower-left corner of the media. The *X,Y* values output are **not** affected by keypad scaling and rotation; they are affected **only** by pen motion, and user and factory MicroCal.

The output format is:

X distance from LL in mils, in signed integer form.

Comma.

Y distance from LL in mils, in signed integer form.

Carriage return (0D hex).

The plotter may transmit IMMEDIATELY after receiving the **XY** command, so be prepared to receive the data. See the **XD** command; you can add minimum delays to the plotter serial output if desired. Also note that if the plotter buffers

are full, there could be a very long delay before the response to an output command comes.

Example: At $X = 1.2345$ " from **LL** (rounds to 1235 mils), $Y = 6.7893$ " from **LL** (rounds to 6789 mils), **!XY @** causes the plotter to transmit 1235,6789

Plotting Commands No Longer Supported

FL Force DM/PL language for the Signcutters. Disables language auto-switch to HP-GL. Once **FL** is sent it will remain in effect until the plotter is powered down. This setting is lost after power down.

Example: **!FL @ .**

MM Manual move mode exit.

MM_{x,y} Manual move mode, direction *x*, speed *y*. This mode allows keypad manual-moves to be controlled from a remote system.

Direction (*x*) values, moving counter-clockwise

0 = Stopped, Equivalent of All Keypad-Keys Released

1 = +*Y*, Equivalent of '4'-Key on Keypad

2 = +*X*, +*Y*, Equivalent of '7'-Key on Keypad

3 = +*X*, Equivalent of '8'-Key on Keypad

4 = +*X*, -*Y*, Equivalent of '9'-Key on Keypad

5 = -*Y*, Equivalent of '6'-Key on Keypad

6 = -*X*, -*Y*, Equivalent of '3'-Key on Keypad

7 = -*X*, Equivalent of '2'-Key on Keypad

8 = -*X*, +*Y*, Equivalent of '1'-Key on Keypad

Invalid = Stopped, Equivalent of All Keypad-Keys Released

Note that the key-numbers apply only to the Signature 5000.

Speed (*y*) values:

0 = Stopped, Equivalent of All Keypad-Keys Released

1 = Normal Speed, Equivalent of Just Direction-Key Held Down

2 = Slow Speed, Equivalent of Holding 'Enter'-Key Down While Pressing a Direction-Key

3 = Fast Speed, Equivalent of Holding '0'-Key Down While Pressing a Direction-Key

4 = Custom Speed Set By the **Ms_x** Command

Invalid = Stopped, Equivalent of All Keypad-Keys Released

This is Real Time: once you enter a non-stopped manual move, the plotter **will keep moving** until it is stopped. If the *X* axis is moving, it will eventually throw your paper on the floor; if the *Y* axis is moving, it will try to move through an endplate. It will not bother the plotter to try to move beyond an endplate, since the motor power is limited to a safe value for manual moves. However, the carriage or the *Y* axis drive train can be damaged by repeated high-speed impacts against an endplate, so take some care when you are using the high-speed setting.

The **MM** command does **not** follow the usual conventions; an **MM** command followed by **any character** except white space or another **MM** command will cause the plotter to stop the manual move and return to plotting mode. This was done to avoid runaway manual moves.

The following conditions cause a manual move to take effect:

Receipt of a complete command. This means that the plotter can tell that it either has the **MM** command that is **not** going to be followed by any numbers, or the **MM d,s** command and it has received the full second number.

For the **MM** form, this could be **!MM @** ; the @ character is non-numeric and non-white space, and tells us we have the **MM** form of this command.

For the **MM x,y** form, this could be **!MM3,1** ; the space following the '1' tells us the second number is complete, without a non-white space character like @ to kick us out of manual-move mode.

Normally for manual-moves you do the following:

Send a **!** to enter IOLS.

As movement-keys are pressed and released, send the appropriate **MM x,y** commands, each followed by a space. The length of time that this **MM** command is in effect will determine how far the pen is moved. Sending the same command over and over will not cause any problems, unless you cause delays in response by filling up serial buffers.

When a movement-key has been released, send an @ to bring us back to the normal IOLS form.

Example: **!MM3,1 <pause> MM0,0 <pause> MM4,1 <pause> MM4,1 <pause> MM7,2 <pause> @ .**

This moves normal speed in +X direction (3), stops, moves in +X, +Y direction (4), continues in same direction (4), moves slowly in the -X direction (7), then stops and exits both manual-move mode and IOLS. The "<pause>" indicates a time delay; the actual character string "<pause>" is NOT sent to the plotter.

The following conditions will stop a manual move:

Sending the **MM** form.

Sending the usual **!MM3,1 @** form; the @ kicks us out of the manual-move mode.

Pressing the "Abort" or "Offline" keys on the plotter keypad.

Note: pressing "Offline" and then "Online" on the plotter keypad during IOLS manual moves may cause strange results. We assume that users will use only one keypad at a time.

Either the 'direction' or 'speed' parameters set to 0.

Sending some other IOLS command. You can send more **MM** commands after the other IOLS command to resume manual-moves.

- MV** This command is no longer supported. Using this command generates no errors, but performs no functions.
- OI** Output plotter ID string. Used with the IOLS **HO** command to set and verify the HP-GL plotter model emulated (queried with the HP-GL **OI** command); some software packages want to think that they are hooked up to a particular model of HP plotter, and these commands let you keep that software happy.
- OV** This command is no longer supported. Using this command generates no errors, but performs no functions.
- PA** Page advance to new lower left using current page size, effectively the same as the **FR** command.
- PA_x** Page advance to new lower left, *x* distance in mils. Effectively the same as **FR_x** command.
- PS** Pen select 0 (put away current pen).
PS_x Pen select *x*.
- Example: **! PS2 @ .** Selects pen 2.
- Example: **! PS @** or **! PS0 @ .** Both put the pen away.
- The **PS** command applies only to the Signature 5000 plotter*
- PV** This command is no longer supported. Using this command generates no errors, but performs no functions.
- RF** This command is no longer supported. Using this command generates no errors, but performs no functions.
- RG** This command is no longer supported. Using this command generates no errors, but performs no functions.
- SL_x** Select plotter language *x*, for the Signature 5000. 0 = HP-GL, 1 = DM/PL.
- Example: **! SL0 @ .** Selects HP-GL, and forces an interpreter reset.
- TS** This command is no longer supported. Using this command generates no errors, but performs no functions.
- XD_{x,y}** Set transmission delays for plotter serial output (responses to output commands).

'*x*' is the turnaround delay. This is the added delay in mSec between processing of an output command, and the transmission of the first character. Minimum 0, maximum 10000 mSec (10 seconds).

'y' is the inter-character delay. This is the added delay in mSec between the transmitted characters. Minimum 0, maximum 10000 mSec.

Example: **!XD50,10 @ .** Sets a 50 mSec (0.050 second) minimum delay before the first character is sent, and 10 mSec (0.010 second) minimum delay between characters after that.

ZD This command is no longer supported. Using this command generates no errors, but performs no functions.

ZF This command is no longer supported. Using this command generates no errors, but performs no functions.

ZL This command is no longer supported. Using this command generates no errors, but performs no functions.

ZP This command is no longer supported. Using this command generates no errors, but performs no functions.

ZR This command is no longer supported. Using this command generates no errors, but performs no functions.

ZV_{u,v,w,y,z} This command is no longer supported. Using this command generates no errors, but performs no functions.

Reference Section

Abbreviations

IOLS - Ioline Language Syntax

HPGL - Hewlett Packard Graphics Language

DMPL - Digital Microprocessor Plotting Language

Using Ioline Syntax

Ioline syntax commands work standalone or embedded in a plot file with other syntax like HPGL or DMPL. If sent standalone, make sure to read the section, *Terminal Communication w/ Ioline Plotters..* Please note that some Ioline syntax commands have parameters and/or return values to the host PC.

All Ioline syntax commands are two CAPITAL letters.

Accepted: **!RB@**

Causes syntax error: **!rb@**

All Ioline syntax commands must have a preceding exclamation point (!) and an 'at' symbol (@) terminator. For example,

Accepted: **!RB@**

Causes syntax error: **RB**

Strings of multiple commands within the (!) and (@) are accepted but must have a space between them. For example,

Accepted: **!IC RB@**

Causes syntax error: **!ICRB@**

Commands with parameters must NOT have spaces between them. For example,

Accepted: **!RB10000,5000@**

Causes syntax error: **!RB 10000,5000@**

Using a Terminal Program to Communicate with an Ioline Plotter

Introduction

This guide demonstrates how to send instructions and files to Ioline Plotters with a terminal program like Hyper Terminal (included with Microsoft Windows). Using a terminal program is not necessary if using the Ioline Control Center, available on the web at <http://www.ioline.com/> or by mail from Ioline customer service. A complete list of Ioline syntax commands is shown above.

Prepare to communicate

1. Attach the plotter to a computer communication port.
2. Turn on both the computer and plotter.
3. After the plotter finishes the initialization sequence the keypad LED will turn red.
4. Press **Start Point** or **Set Origin** so that the LED changes from red to green.

Setup the Terminal Program

1. Start the terminal program (Hyper Terminal in Windows, Zterm in Mac, etc.).
2. Set up the terminal for communication through the computer port chosen above.
3. If using serial communication, ensure the properties are set to match the following:

COM Port:	Set to match PC port
Baud rate:	9600
Data Bits:	8
Parity:	none
Stop Bits:	1
Flow Control:	Xon/Xoff

4. If the program has a 'Local Echo' setting, enable this to view what is typed into the terminal and sent to the plotter.
5. Some programs (like HyperTerminal) may require that the communication settings are saved and the program is restarted before changes take effect. Follow this step if there is trouble communicating with the plotter.

Communicating

1. After setting the communication properties, check to see if the connection is active by typing !IC@. An ID code should appear on the screen (see IC command for a list of device ID codes). The keypad LED remains green when the plotter receives the command without an error. See the *Troubleshooting* section below if the keypad LED flashes alternating red and green.
2. If the terminal program did not automatically connect to the plotter, look for a command to start the connection and try the above test again.
3. When communication is successful, the plotter will accept any Ioline syntax command or a plot file written in HPGL 7475 or DMPL. To use HPGL 7596, switch modes as described in the Ioline Syntax Guide under the HO command.

Troubleshooting

1. If nothing happens when trying to communicate, try the following:
 - Check that local echo is enabled in the terminal program.
 - Make sure that all characters are typed in UPPERCASE.

- Ensure that the syntax command always starts with an ! and ends with @.
- More than one syntax command can go between the ! and @.

Example:

!LL0,0 UR46000,49000 RF1@

- Check that the communication settings are correct.
- A single space is allowed between each command but not between a command and associated vector endpoints or parameters. Example:

Good:

!RF1@
!IC AE0@
!LL0,0@

Bad:

!RF 1@
!IC AE0@
!LL 0,0@

- Vector endpoints, in the Ioline syntax, are in mils (thousandths of an inch) and are in the format X,Y. Example, to set a Lower Left point at 1 inch in X and 2 inches in Y, type:

!LL1000,2000@

2. If the keypad LED is flashing alternating green and red, then the last syntax was not understood. This is a *Syntax Error*. To clear the syntax error, press the **Start/Stop** key until the LED turns green then try the command again using the above suggestions as a guideline.