

Downloading coordinates from a PC program into the controller.

The controller needs to have a complete route before RUN so the DSP can compute the necessary profiles in advance.

First off create a route (usually Cartesian) and reserve more than enough lines for the coordinates you will be downloading.

The code below allows you to download the values from a list delimited by spaces or returns.

It envisages sending the coordinates 5 values at a time:

X Y Z PITCH ROLL

Each value is separated by a space. After 5 values you can send another space or a return (but not both). If you want some other delimiter change ASPACE to the hex value of your delimiter.

To use the function send RX then return (0D) then your lists of values. Obviously the total number of values should be divisible by 5. Finally send ETX (03, end of text).

Note that characters sent or typed are not reflected back. You can test as follows, but there will be nothing on the screen.

enter RX

then

1(space)2(space)3(space)4(space)5(return)

11(space)22(space)33(space)44(space)55(return)

then enter control-C (etx) - should show OK

- or any other numbers you choose – note a 5 figure number will cause an overflow.

then enter

L. (L-dot)

This lists the route as follows

LINE	X	Y	Z	PITCH	ROLL
------	---	---	---	-------	------

1	0.1	0.2	0.3	0.4	0.5
---	-----	-----	-----	-----	-----

2	1.1	2.2	3.3	4.4	5.5
---	-----	-----	-----	-----	-----

Note that your numbers are multiples of 0.1mm or 0.1 degrees.

Sample code follows. Use Acrobat select tool to copy the text.

Help sheet 17



```
HEX
USER FLAG1
: GETVAL
0 FLAG1 !
0
BEGIN
  KEY DUP 2F > IF
    SWAP 0A * OVER OF AND +
    SWAP ( BACK
  THEN
DUP ASPACE = IF 1 FLAG1 ! THEN ( DELIMITER
DUP OD = IF 2 FLAG1 ! THEN
03 = IF 3 FLAG1 ! THEN ( END OF TEXT
FLAG1 @ UNTIL
;
: GETLINE ( meaning a line in a route, i.e. one position.
LINE# !
5 0 DO
  GETVAL
  FLAG1 @ 3 = IF ( ETX
    LEAVE
  ELSE
    LINE# @ LINE I 2* + E!
    0 LINE# @ LINE 0A + E! ( 6th value
    2 LINE# @ LINE 0C + E!
    0 LINE# @ LINE 0E + E!
  THEN
LOOP
;
: RX
TEM ( or whatever your route is called
0 MOVES E!
BEGIN
  MOVES E@ 1+ DUP
  GETLINE
  FLAG1 @ ETX < IF MOVES E! ELSE DROP THEN
FLAG1 @ ETX = UNTIL
;
```