

LSS Information Pack



# Principles of Survey and Feature Coding



Creative Software  
for a  
Changing World

## Contents

- 1 LSS Feature Coding
  - a. Linking
  - b. Closing links
  - c. Multiple coding / changing features
  - d. General coding rules
- 2 Rectangular Offsets
- 3 Rectangular Indents
- 4 Perpendicular Offsets
- 5 Intersection
- 6 String Offsets
- 7 Angular Offsets
- 8 Station Control
- 9 Line of Sight Eccentric Offsets
- 10 Up / Down Eccentric Offsets
- 11 Previous Point Eccentric Offsets
- 12 Steps (Stairs)
- 13 Building Elevations

## 1 LSS feature coding

---

### Linking

An observation can have multiple link feature codes each of which will link back to the last observation using the same feature.

A link can be started or ended, even when the feature has previously been used, by the combined use of the full stop (“.”) character and/or string numbers, as described below.

A full stop prior to the link feature indicates starting a new link i.e. not to join back, and a full stop following the feature will stop any outgoing connection.

1	2	3	4	5	6	7
X	X	X	X	X	X	X
K	K	.K	K.	K	K	K

The full stop can be used by itself where the feature is taken from the previous observation.

1	2	3	4	5	6	7
X	X	X	X	X	X	X
K	K	.K	.	K	K	K

It is also possible to start and end link features using string numbers after the code. This will only link back to a specific feature of the same number. A string number can be any positive value.

1	2	3	4	5	6	7
X	X	X	X	X	X	X
K1	K1	K2	K2	K3	K3	K3

Further, string numbers and dots can be used in combination.

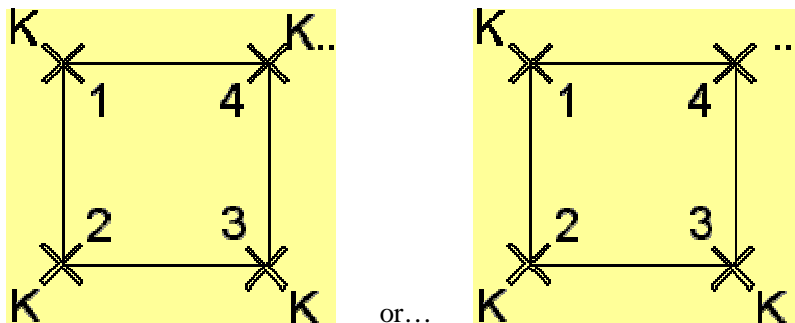
1	2	3	4	5	6	7
X	X	X	X	X	X	X
K12	K12	.K12	.	K12	K12	K12.

## LSS feature coding

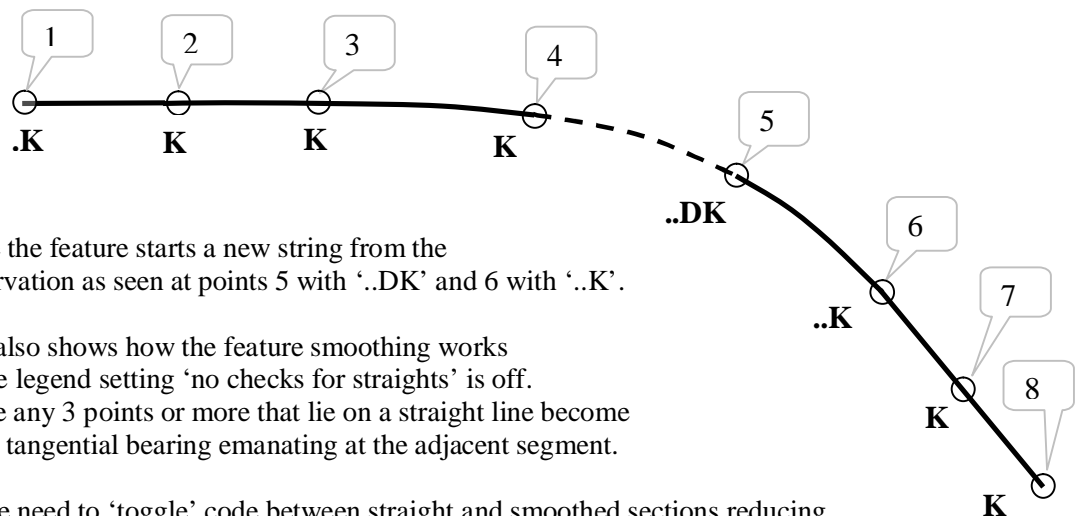
### Closing links

The use of ‘.’ or ‘@’ following the link feature code will ‘close’ the feature string i.e. link back to the first observation of the string. Choose the character(s) most appropriate to the logger in use.

A double dot can be used by itself to link back to the previous observation using the feature at that point and also to close the string. This close operation will operate on a string of any geometric shape.



### Smoothing links



The ‘.’ before the feature starts a new string from the previous observation as seen at points 5 with ‘..DK’ and 6 with ‘..K’.

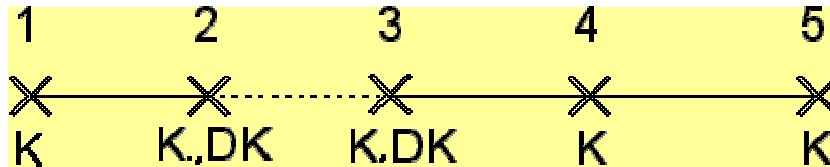
This diagram also shows how the feature smoothing works when using the legend setting ‘no checks for straights’ is off. This will make any 3 points or more that lie on a straight line become straight with a tangential bearing emanating at the adjacent segment.

This avoids the need to ‘toggle’ code between straight and smoothed sections reducing the coding input as well as the number of observations required. The smoothing will even be applied across different adjacent feature codes that are also set to be smoothed.

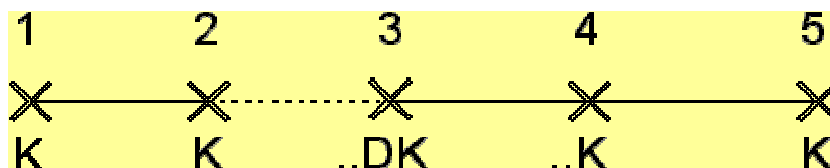
In the example the points 1 to 3 are straight so the smoothing starts at 3 to 4 using the bearing from 2 to 3. Similarly, 6 – 8 are straight with the smoothing from 5 tying into the bearing 7 to 8. The smoothing between 3 and 6 happens because both features K and DK are set to be smoothed.

## Multiple coding / changing features

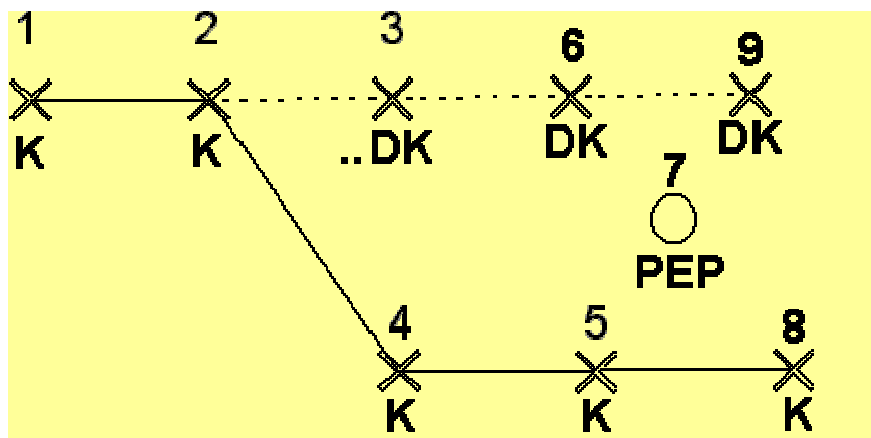
The diagrams below show two methods of inserting a feature link into an ongoing link feature.



As it can be seen the two methods produce the same results but the '..' code reduces the number of features needing to be specified.



The following multi-coding example shows how link features should be surveyed sequentially along the line of the feature as the points forming the feature are connected in the order they are observed and not on any relative location basis.



## General coding Rules

Spot levels are the default state of an observation i.e. without a point or a link code.

An unlimited number of links can emanate from a single point. Also point and link features can exist at the same observation e.g. a tree can lie along a fence line, 'F6, PT8'.

Only one reserved code e.g. '-REC' is allowed per observation and must also be the first code if multiple feature codes are present (with the exception of '-SO').

The divide '/' symbol can be used instead of a comma as the separator between feature codes.

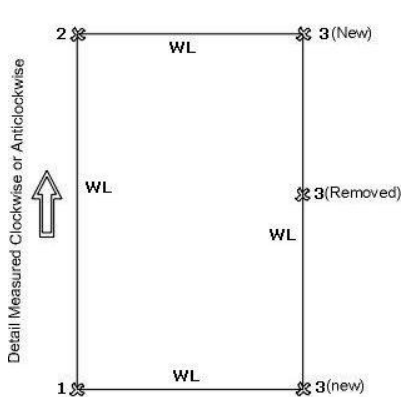
## 2 Rectangular Offsets

### -REC

This will create true rectangles from three observed points. Two points along one side (preferably the longest) of the rectangle are observed, the third point is then observed with the code '-REC'. The width of the rectangle is based on the perpendicular distance from the third point to the line joining the first two points. LSS can then generate two new points at the same direction and perpendicular distance as the third point to complete the other two corners of the rectangle.

The third observed point is REMOVED (which can later be REINSTATED if required).

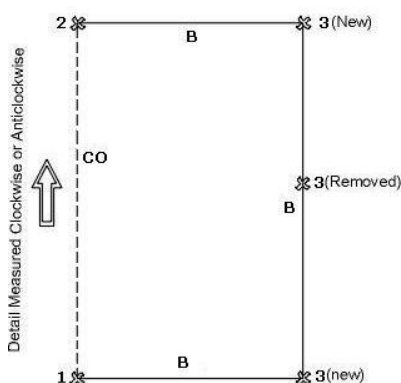
The two generated points have their levels calculated assuming a plane passes through the initial three points and their point numbers are the same as the removed third observation.



A link feature can be applied to the rectangle by being on the first two observations.

Example LSS load file feature coding;

```
5 → WL  
5 → WL  
5 → -REC
```



If the code following the -REC is different to the code for the first two observations the former will be applied to the created three sides and the latter to the link between points one and two.

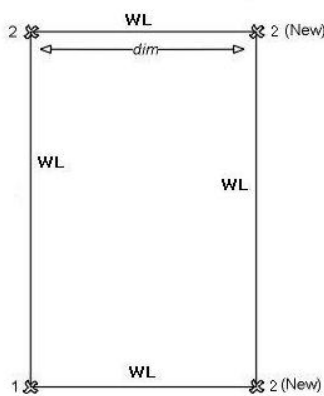
Example LSS load file feature coding;

```
5 → CO  
5 → CO  
5 → -REC, B
```

## -REC(dim)

Two points along one side (preferably the longest) of the rectangle are observed with the second observation given the code '-RECdim' where *dim* is the offset value in metres (positive to the right). LSS will automatically generate two new points, at the required offset, which are the other two corners of the rectangle.

The newly generated observations are given the same level of the adjacent observed points and the same point number as the second point.

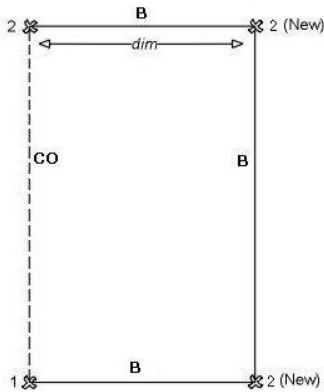


A link feature can be applied to the rectangle by the code being given to the first observation.

Example LSS load file feature coding;

5 → WL

5 → -REC1.2



If a different code follows the -RECdim to that on the first observation then it will be applied to the created three sides with the first code being applied between points one and two.

Example LSS load file feature coding;

5 → CO

5 → -REC1.2, B

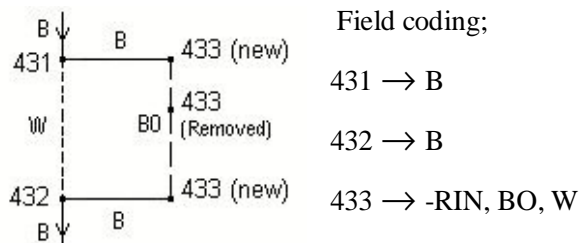
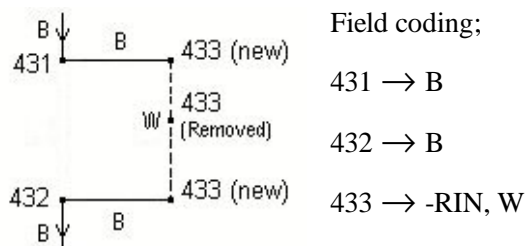
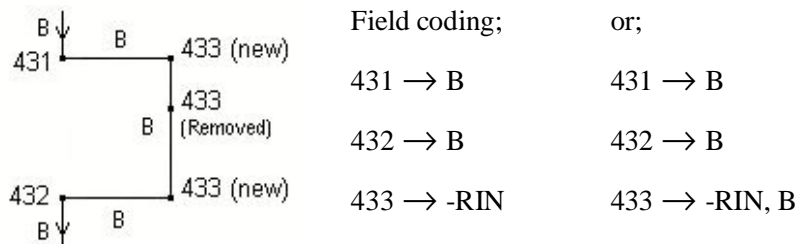
### 3 Rectangular Indents

#### -RIN

This code is used for the calculation of rectangular indentations along a feature line e.g. a building. The first two points are the baseline and the third point can be taken anywhere along the indented (rear) face between them with the code '-RIN'. Two new points will be calculated as per '-REC'.

The link feature used on the first two observations is applied to the two sides of the indentation. The code that follows -RIN in the second field is applied to the rear face, if this field is blank then the feature used will be the same as that of the two sides. The third field specifies the feature for the front face, which will be left open if no code is specified.

If the same string code is observed on a previous or subsequent point, to this rectangular feature, the link will continue in sequence as per the diagram. Hence this option will be formed within an existing string.

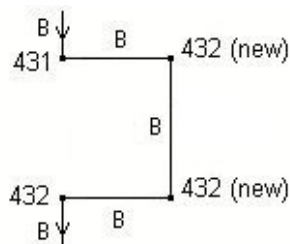


## -RIN(dim)

This code is used for the calculation of rectangular indentations along a feature line e.g. a building. The two surveyed points will be the baseline and the dimension to the indented (rear) face will be given on the second point (positive to the right) using the code `-RINdim`. Two new points will be calculated as per '`-RECdim`'.

The link feature used on the first observation is applied to the two sides of the indentation. The code that follows `-RIN` in the second field is applied to the rear face, if this field is blank then the feature used will be the same as that of the two sides. The third field specifies the feature for the front face, which will be left open if no code is specified.

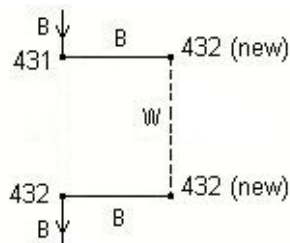
If the same string code is observed on a previous or subsequent point, to this rectangular feature, the link will continue in sequence as per the diagram. Hence this option will be formed within an existing string.



Field coding;                      or;

431 → B                              431 → B

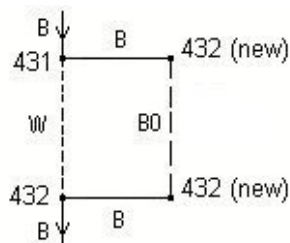
432 → -RIN-1.2                    432 → -RIN-1.2, B



Field coding;

431 → B

432 → -RIN-1.2, W



Field coding;

431 → B

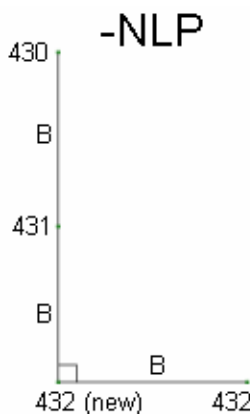
432 → -RIN-1.2, BO, W

## 4 Perpendicular Offsets

### -NLP

This will create an extra point as a perpendicular between three observed points. Two points are observed as the baseline followed by the third point with the code '-NLP'. LSS can then generate a new point such that the line from the third point to it forms a perpendicular to the extended baseline from point one to point two.

The newly generated point has its level calculated assuming the extended baseline is linear and its point number becomes the same as the third observation.



The link feature used on the first two observations is applied on to the created point and then to the third point. If the same string code is used elsewhere then the link will continue in sequence, hence this option will allow the creation of new points within an existing string.

Example LSS load file data;

```
5, 430, 137.10.10, 87.31.45, 48.8780, 1.600, B
```

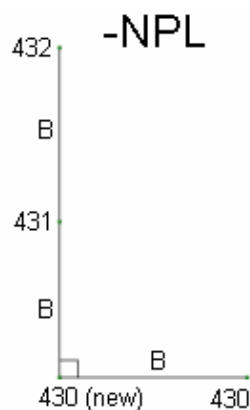
```
5, 431, 125.31.35, 87.18.10, 42.3440, 1.600, B
```

```
5, 432, 123.50.20, 87.04.10, 36.5400, 1.600, -NLP
```

### -NPL

This will create an extra point as a perpendicular between three observed points. A single point is observed followed by two points forming a baseline with the third point having the code '-NPL'. LSS can then generate a new point such that the line from the first point to the new point forms a perpendicular to the extended baseline from point three to point two.

The newly generated point has its level calculated assuming the extended baseline is linear and its point number becomes the same as the first observation.



The link feature used on the first two observations has the created point inserted between them and then continued to the third point. If the same string code is used elsewhere then the link will continue in sequence, hence this option will allow the creation of new points within an existing string.

Example LSS load file data;

```
5, 430, 123.50.20, 87.04.10, 36.5400, 1.600, B
```

```
5, 431, 125.31.35, 87.18.10, 42.3440, 1.600, B
```

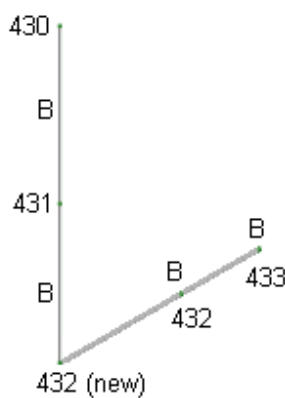
```
5, 432, 137.10.10, 87.31.45, 48.8780, 1.600, -NPL
```

## 5 Intersection

### -INT

This will create a point at the intersection of two lines. The first and second points are observed as the initial baseline and the third and fourth points as the second baseline, the fourth point being given the code '-INT'. LSS can then generate a new point where the two baselines intersect. The intersection must extend on both sides as per the diagram and not intersect internal to either line.

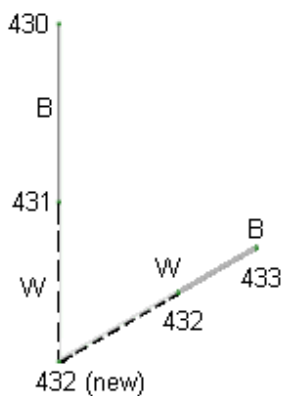
The newly generated point has its level calculated assuming the line between point two, the new point and point three is linear and its point number becomes the same as the third observation.



The link feature used on the first three observations is applied to the created point which is inserted between points two and three, and then to the fourth point. If the same string code is used elsewhere then the link will continue in sequence, hence this option will allow the creation of new points within an existing string.

Example LSS load file data;

```
5, 430, 131.11.52, 87.19.12, 45.060, 1.600, B
5, 431, 125.31.35, 87.18.10, 42.3440, 1.600, B
5, 432, 122.20.39, 87.13.04, 38.485, 1.600, B
5, 433, 124.55.22, 86.58.22, 35.277, 1.600, -INT
```



If there is a different link feature following the '-INT' to that of the first three points then the former will be applied to the two links formed when the created point is inserted between points two and three and the latter is applied from point one to point two and from point three to point four. If the link on the first three points is used elsewhere, to this intersected feature, then it will continue in sequence, hence this option will allow the creation of new points within an existing string.

Example LSS load file data;

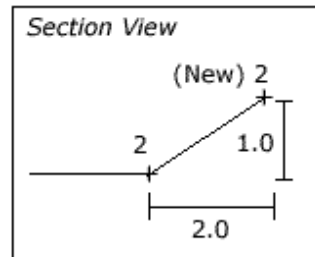
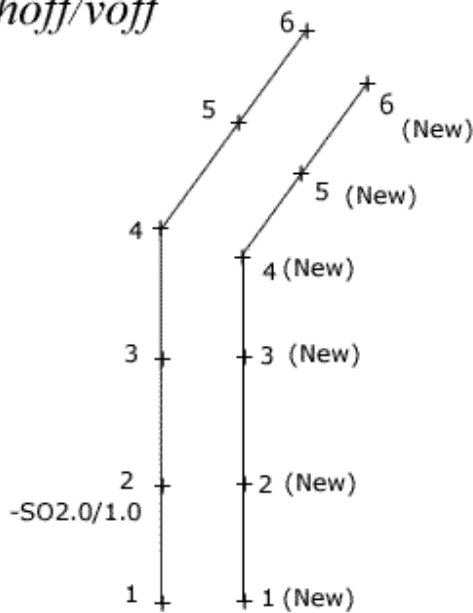
```
5, 430, 131.11.52, 87.19.12, 45.060, 1.600, B
5, 431, 125.31.35, 87.18.10, 42.3440, 1.600, B
5, 432, 122.20.39, 87.13.04, 38.485, 1.600, B
5, 433, 124.55.22, 86.58.22, 35.277, 1.600, -INT, W
```

## 6 String Offsets

**-SO & -SOM** - generate offset string(s) from the complete reference link feature. With '-SO' offsetting to one side only and '-SOM' to both sides.

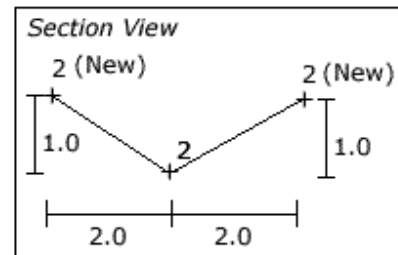
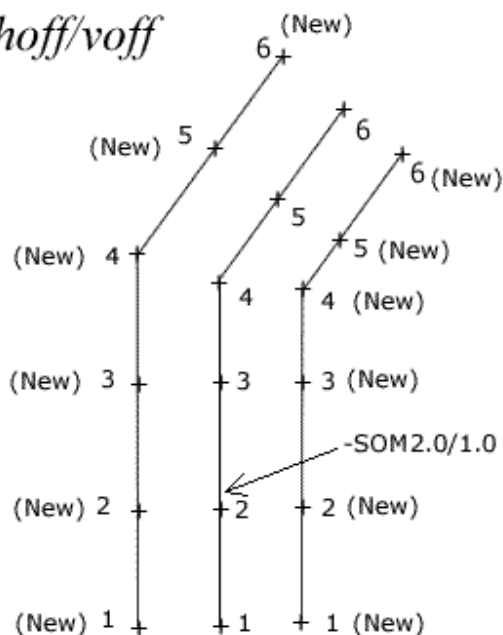
**1. Dimensioned version** – where the parameters are the horizontal and vertical offsets and the offset feature.

**-SO** *hoff/voff*

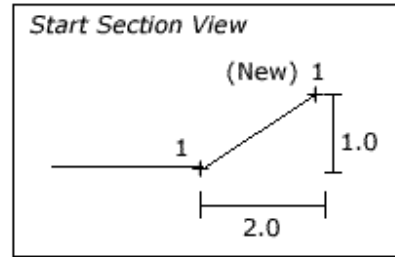
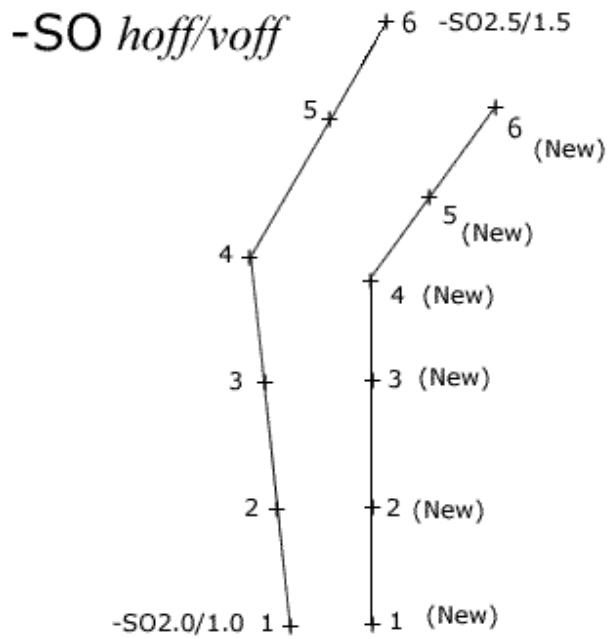


Here the single dimensioned '-SO' on point number 2 will give a parallel string 2m to the right and 1m above the reference line (see section view).

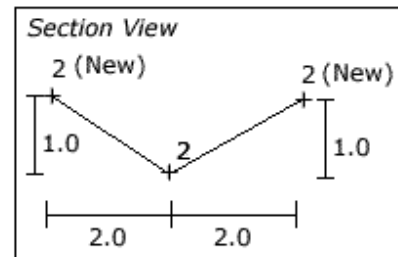
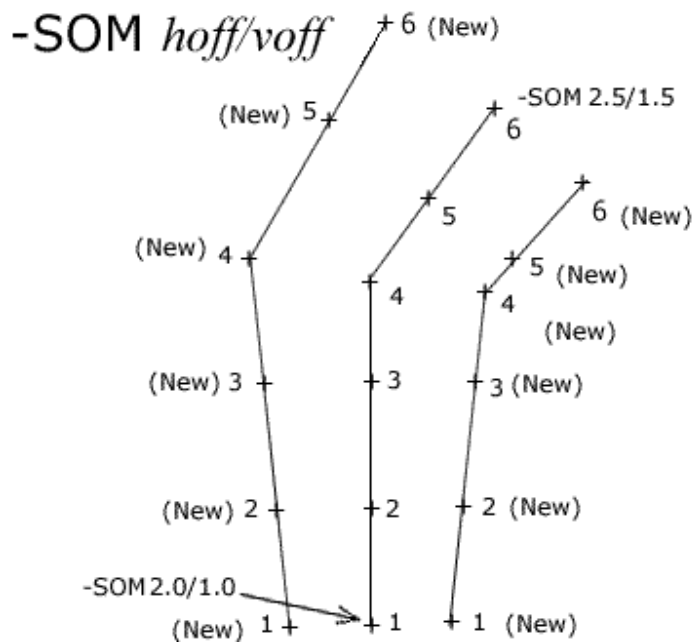
**-SOM** *hoff/voff*



Here the single dimensioned '-SOM' on point number 2 will give two parallel strings 2m to either side and 1m above the reference line (see section view). Negative values for the horizontal are treated as positive.

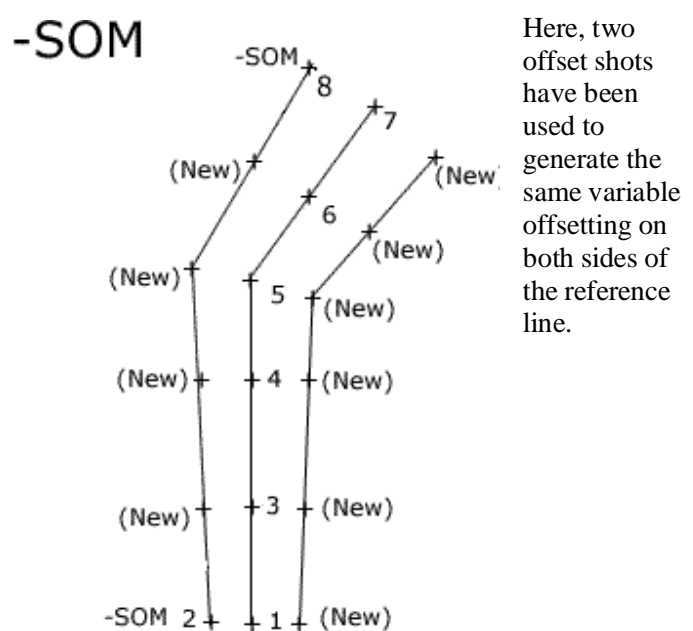
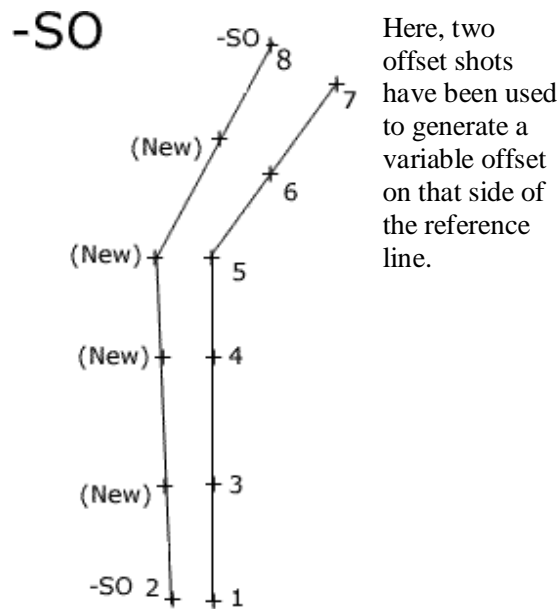
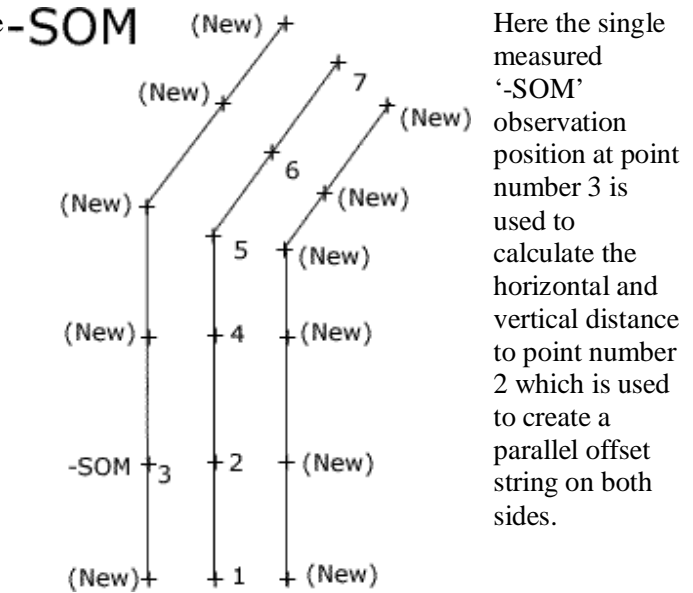
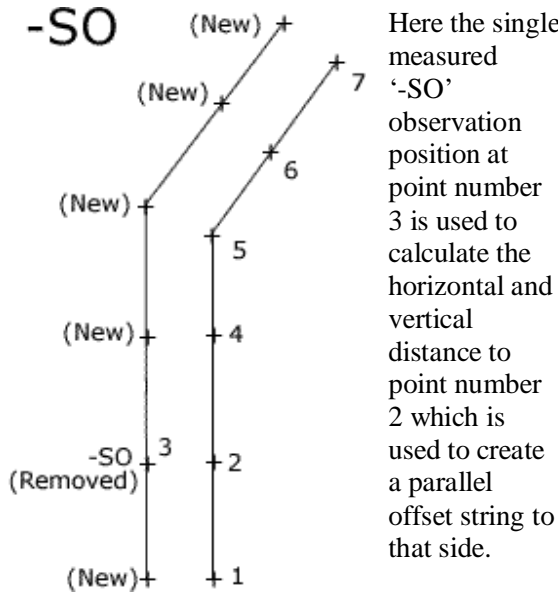


Here, two dimensioned '-SO' codes have been used to generate a variable offset string.



Here, two dimensioned '-SOM' codes have been used to generate a variable offset string on either side of the reference line.

**2. Measured version** - allows the user to survey a sequential observation on the other string adjacent to the observation just taken on the reference string. LSS calculates the straight line distance horizontally and vertically between the two points and applies these values as the string offsets i.e. not the perpendicular distance to the reference string. The shot itself will be removed during processing.



### General notes:

- A single use of the string offset will create a parallel string feature and can be made at any point along the reference string as the offsets apply backwards and forwards along the string to either end or until another string offset occurs.
- If more than one string offset is applied then the points that lie between those where the offsets were taken have their horizontal and vertical offsets calculated such that the values vary in proportion to the distance traveled along the string.
- The stringing of LSS link features, and therefore the string offset, will extend across setups by default. To prevent this the [CLEAR](#) command must be specified in the load file.
- Corners are mitred i.e. one point created on the offset string for each point on the reference string. The internal corners are calculated to the bisector and external by the intersection of the offset lines. The vertical offset is calculated to the corner points not to the adjacent lines.
- Neither the dimensioned and measured or '-SO' and '-SOM' can be mixed on the same reference string.
- For '-SOM' the offset features on both sides will always be the same.
- The link feature for the reference string cannot change.

### Coding notes

The reference line feature does not need to be specified with the offset coding as the feature link of the previous point is assumed e.g. '**-SO-1.0/0.1, F**' except for the start of a string (see below).

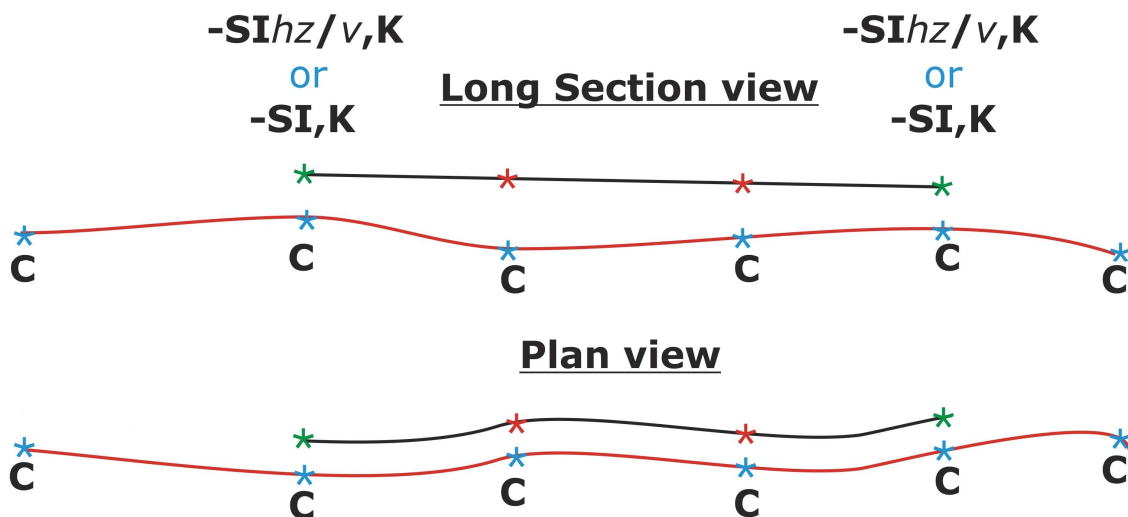
- Dimensioned version;
  - The general coding form is; *(-SO or -SOM)hoff / voff, offset feature.*
  - The horizontal offset is mandatory and a negative value offsets to the left.
  - If the optional vertical offset is omitted or 0m then the offset is horizontal.
  - If the optional offset feature is not given LSS will use the reference line's feature code for the offset string.
  - For '-SO' the offsets are permitted to both sides but are treated independently i.e. an offset string will not cross the reference string.
- Measured version;
  - The offset feature can be specified by e.g. '**-SO, F**'
- String coding
  - Start or stop an offset string - the offset string is generated along the entire length of the reference line (including when closed) unless the full stop '.' is used to stop/start the offset string e.g. '**-SO[ dims], .F**' to start or, '**-SO[ dims], F.**' to stop.
  - Offset coding on the first point of a reference string needs to include the reference string feature e.g. '**.W, -SO-1.0/0.1, F**'
  - Offset used with join to previous has to be specified as per the start of a string where the use of the double full stop '..' before the reference string feature will start the string, and hence the offset feature, from the previous point e.g. '**..W, -SO -1.0 / 0.1, F**' (any variable offsets will commence from this shot).

## String Interpolations

**-SI & -SIM** - generate offset string(s) from the complete reference link feature as per '-SO' with the important difference of the levels being calculated for the intermediary points, between the coded offset points, on the level plane.

### Purpose

This allows the creation of offset features e.g. slabs and top of kerbs where the level gradient is based purely between the levels at the offset points and not the reference string.



### Differences to '-SO'

- The offsets are only calculated between the offset coded points i.e. there is no extrapolation to the end of the string if offset coding is not present.
- The exception to the above is the condition of a single offset code where the complete reference link feature is offset onto a datum plane as per the value from the offset coded point.

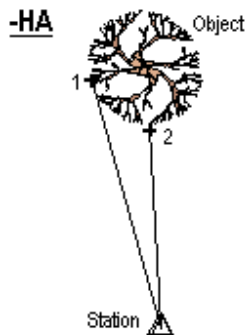
## 7 Angular Offsets

### -HA

This is used to create an 'offset' observation e.g. where the centre of an object is not accessible. A pair of observations is made, the first being taken at the correct distance i.e. next to the object. The second observation is made to the centre of the object i.e. along the required line of sight, and is given the code '-HA'.

The first observation is automatically REMOVED but can later be REINSTATED if required.

The second point becomes an AMENDED survey observation with all the data taken from the first point except the horizontal angle. Its ORIGINAL position being the survey observation data.



Example LSS load file data:

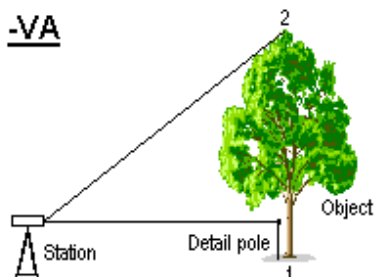
```
5, 1, 1621344, 901020, 783.777, 2.412, PT18 << Removed  
5, 2, 1624526, 881224, 803.650, 2.412, -HA << Original
```

Combined observation data;

```
5, 2, 1624526, 901020, 783.777, 2.412, PT18 << Amended
```

### -VA

This is used to create an 'offset' observation in the vertical plane to an inaccessible point from an accessible one. Two observations are made, the first point is observed, stored and Loaded into the DTM as normal. The second point is given the code '-VA' and observed to vertically above or below the first point. The vertical angle from the second observation is used, assuming a target height of 0.0m, to determine the vertical difference in height between the first and second points. LSS then creates a block of text saying "Observed ht. from G.L. xx.xxx" which is centred over the first observation (where xx.xxx is the value of vertical difference).



The second observation itself is ignored and as such does not require a recorded distance etc.

Example LSS load file data:

```
5, 1, 1621344, 901020, 783.777, 2.412, PT18  
5, 2, 1621344, 881224, 803.650, 2.412, -VA
```

## 8 Station Control

---

### -ST radial data

The following examples show how -ST is used to create a control observation to a new or existing station

Example LSS load file data:

0, Survey: EXAMPLE SURVEY	
1, DMS, VASD	
2, 5, 2000.000, 5000.000, 50.000	
2, 5A, 2000.000, 6643.850, 52.411	
2, 5B, 1982.348, 4076.700, 53.185	
	Station coordinates
3, 5, , , 1.32	
1.32	
	Set-up at station 5, instrument height
4, 0, 0000000, 895315, 1643.850, 2.412, 5A	
5, 1, 1951618, 901110, 923.318, 2.412, -ST5B	
5, 2, 1765212, 901151, 831.142, 2.241, -ST5C	
	Control observations

The above example shows the use of '5, ..., -ST' and the '4,' records as control observations. They have identical meanings, but the '-ST' code can be used during ordinary detailing. A control observation is not loaded into the DTM as per detail, but used to create new stations or check against existing stations during the Load process. As station 5B already exists, LSS will report a table of differences during Input/Load. However, as station 5C does not exist, then LSS will be able to create a new station.

### -ST coordinate data

0, Survey: EXAMPLE SURVEY

21, 1, 441635.791, 634207.627, 1.142, -ST9  
21, 2, 441845.602, 634416.151, 1.442, -ST10

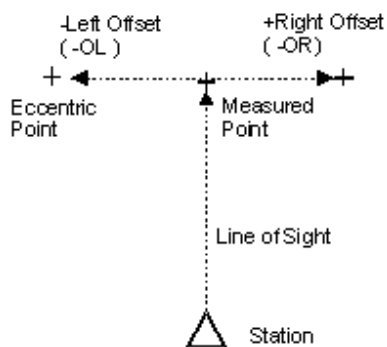
In the above example, LSS treats the '21, ..., -ST' record as a '2,' station record. The only difference being that if the station already exists Load will report the usual table of differences between observed and stored values but the station coordinate concerned will not be updated as per the '2,' record. This enables the '21, ..., -ST' codes to be used by GPS instruments as the equivalent to a control observation in conventional radial observations.

## 9 Line of Sight Eccentric Offsets

The Line of Sight offsets are based on a baseline between the set-on station and the current measured point i.e. the line of sight and can be introduced by using the following codes as features during detailing. If a feature code is also required then multiple coding is needed where the offset code must be first e.g. '-OL1.3, H2' to offset a point 1.3 left whilst on a hedge string.

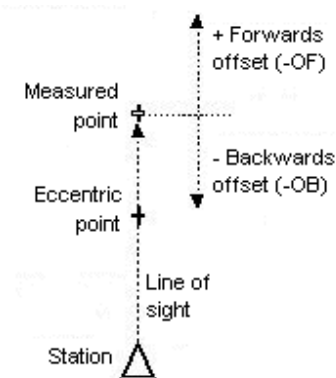
The calculated offset position is made a new point and the observed point is REMOVED and can later be REINSTATED if required.

The level for the new point is calculated on the same plane as the observation height i.e. the slope along the line of sight is ignored.



### Offset left/right

- OLn.nn - offset left of the line of sight n.nn metres
- ORn.nn - offset right of the line of sight n.nn metres
- TLn - offset left of the line of sight. Where n = 1x current detail pole height
- TRn - offset right of the line of sight. Where n = 1x current detail pole height



### Offset forwards/backwards

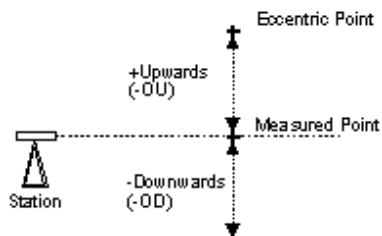
- OFn.nn - forward /extend the line of sight n.nn metres
- OBn.nn - back / shorten the line of sight n.nn metres
- TFn - forward /extend the line of sight. Where n = 1x current detail pole height
- TBn - back / shorten the line of sight Where n = 1x current detail pole height

## 10 Up / Down Eccentric Offsets

---

The Up / Down offsets are based on a height change at the surveyed point where the target height is amended by the required vertical displacement (positive change in target height for a negative change in height).

This option has the advantage of applying a height change to one point only i.e. you do not need to remember to change the target height back to its original value because the coded target height remains the same.



### Offset up/down

-OU $n.n$  - end point up  $n.nn$  metres

-OD $n.nn$  - end point down  $n.nn$  metres

-TU $n$  - end point up. Where  $n = 1 \times$  current detail pole height

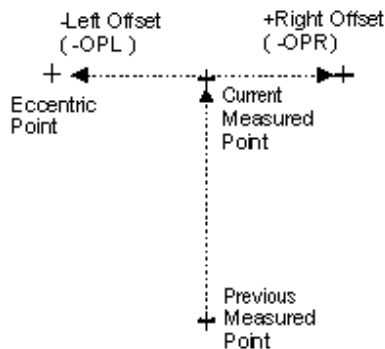
-TD $n$  - end point down. Where  $n = 1 \times$  current detail pole height

## 11 Previous Point Eccentric Offsets

The Previous Points offsets are based on a baseline between the last measured point (Previous Point) and the current measured point and can be introduced by using the following codes as features during detailing. If a feature code is also required then multiple coding is needed where the offset code must be first e.g. '-OPL1.3, H2' to offset a point 1.3 left whilst on a hedge string.

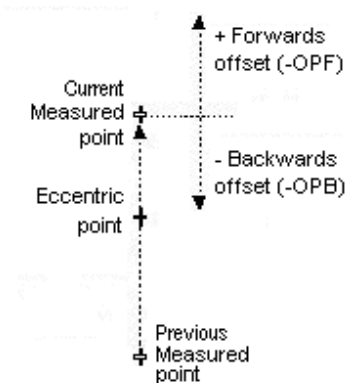
The calculated offset position is made a new observation with the same point number as the original observation which is maintained. The link feature used on the measured observation is applied to the calculated point. If the same string code is observed on a subsequent point the link will continue in sequence. Hence this option will allow the creation of the new point within an existing string

The level for the new points are calculated on the same plane as the observation height i.e. the slope along the line of sight is ignored.



### Offset Point left/right

- OPLn.nn - offset left of the baseline n.nn metres
- OPRn.nn - offset right of the baseline n.nn metres
- TPLn - offset left of the baseline. Where n = 1x current detail pole height
- TPRn - offset right of the baseline. Where n = 1x current detail pole height



### Offset Point forwards/backwards

- OPFn.nn - forward /extend the baseline n.nn metres
- OPBn.nn - back / shorten the baseline n.nn metres
- TPFn - forward /extend the baseline. Where n = 1x current detail pole height
- TPBn - back / shorten the baseline Where n = 1x current detail pole height

## 12 Steps

These options enable different shapes of steps or stairways to be surveyed.

The points may be surveyed in any order, excepting the following rules:

2 or 3 points must exist consecutively to describe at least one of the steps.

The “n” in all cases refers to the number of step risers e.g. ‘-RSn’.

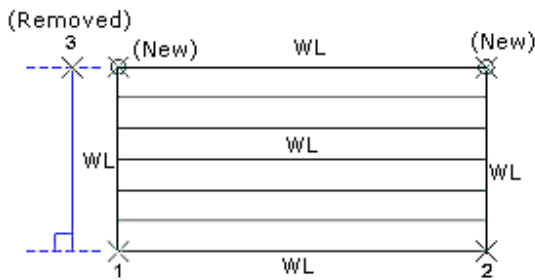
The level difference between top and bottom steps must exceed the level difference between points on the same step.

All generated points will assume the same point number as the detail shot that was assigned the reserved code.

A link feature can be applied to the steps by being on the initial observations or by being added to the last observation containing the steps special code (e.g. -RS). If a different code exists for the first two observations to that following the steps special code then the former is applied to the two sides of the steps and the latter to the created steps themselves.

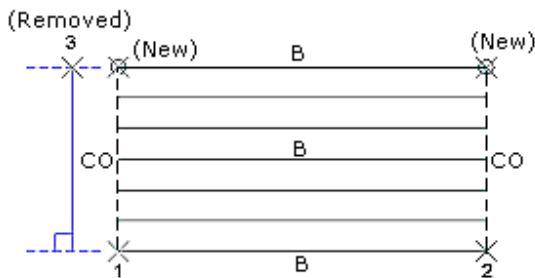
### -RSn

The coding for this is similar to -REC i.e. 3 points with the third point being REMOVED (which can later be REINSTATED if required)



Example LSS load file data;

```
5, 1, 572335, 861720, 8.230,1.850, WL
5, 2, 324320, 810810, 3.450,1.850, WL
5, 3, 93120, 863040, 8.340,1.850, -RS7 ,or...
5, 1, 572335, 861720, 8.230,1.850
5, 2, 324320, 810810, 3.450,1.850
5, 3, 93120, 863040, 8.340,1.850, -RS7, WL
```

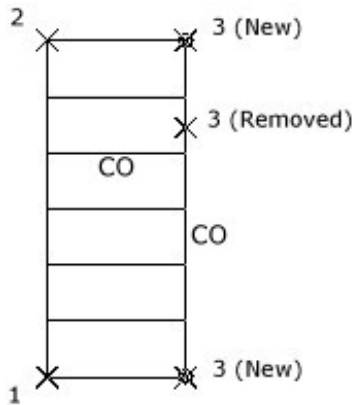


Example LSS load file data;

```
5, 1, 572335, 861720, 8.230,1.850, CO
5, 2, 324320, 810810, 3.450,1.850, CO
5, 3, 93120, 863040, 8.340,1.850, -RS7, B
```

## Steps

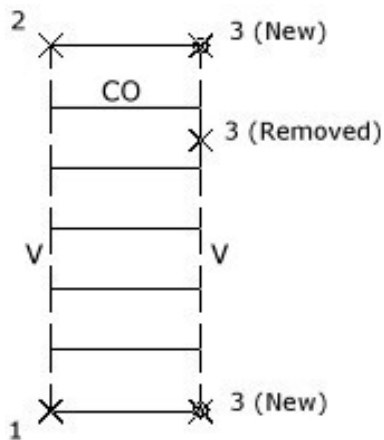
### -PSn



3 points will need to be provided to calculate a rectangular stairway - the 3rd shot having the code - PSnn (where nn= number risers). The two points are the highest and lowest levels, in either order, representing the top step and the bottom riser and must be one of the sides of the steps. The remaining point will only be used to determine the width of the steps and so must be on the opposite side to the other two points.

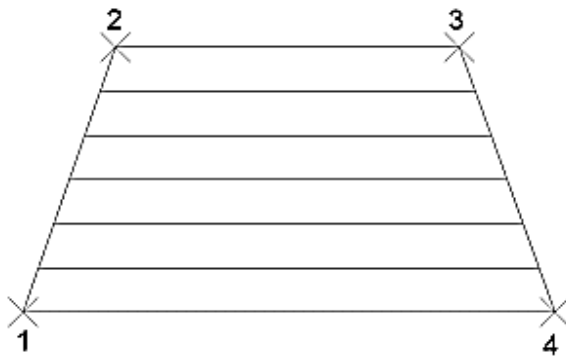
Example LSS load file data;

```
5, 355, 1 54 30, 91 37 25, 520.753, 1.500, CO
5, 357, 3 35 56, 91 39 01, 516.241, 1.500, CO
5, 359, 2 54 45, 91 40 04, 509.325, 1.500, -PS7
```

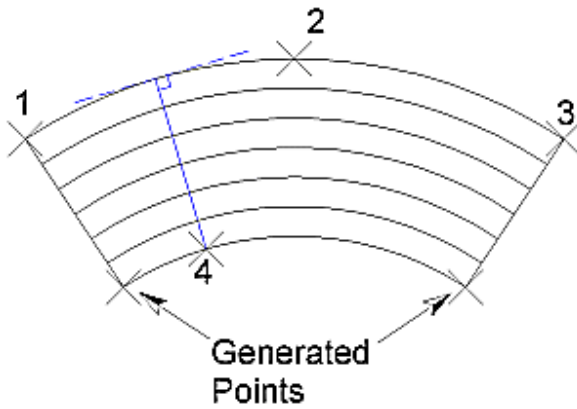


Example LSS load file data;

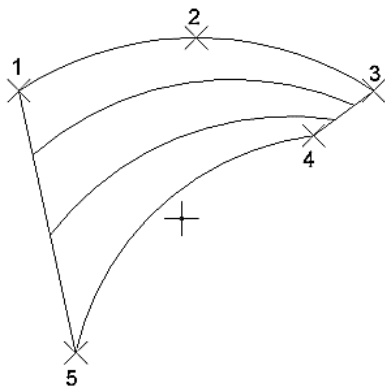
```
5, 355, 1 54 30, 91 37 25, 520.753, 1.500, CO
5, 357, 3 35 56, 91 39 01, 516.241, 1.500, CO
5, 359, 2 54 45, 91 40 04, 509.325, 1.500, -PS7, V
```



**-QSn (4 points)**



**-ASn (4 points)**



**-A5Sn (5 points)**

**-CSn (4 points) as -ASn but full circle**

## 13 Building Elevations

---

### -EL

This reserved code allows surveyed elevation data to be included within an LSS load file. The data is not retained by LSS - it is simply projected on to a user defined baseline (or series of baselines) and output to another file. This new file can be a DXF file or another LSS load file (for a different "survey") and will contain the transformed co-ordinates - the X will be the distance along the baseline(s), the Y as the level and the Z as either the perpendicular distance to the baseline or a repeat of the level. The output may (optionally) include DTM data from the current survey.

Each setup can contain topographical data followed by elevation data. To recognise that elevation data exists a shot is required to have a label of "-EL". This shot will not be used if there is no other label field following the "-EL". Once LSS has identified this label no more DTM data may be surveyed until the user supplies a new setup (code 3), a new station (code 2) or indeed a parameter record (code 1). When LSS recognises codes 1 to 3 then, by default, DTM data is understood to follow.

Labelled shots within the elevation data may use :

- all the 'Special codes', for example the ".", ".." or "@",
- only the 'Reserved codes' "-REC" and "-RECdim",
- a new code "-RW" will, with it's previous shot, create an orthogonal shape - thus, to describe a window only 2 shots forming a diagonal is required.

The Load file is read and any topographical data is added to the current survey's DTM. LSS will then request the type and name of the output file and then request details of each baseline. Each elevation face has a "corridor" or envelope that limits the data to be considered for projecting onto the baseline. The corridor is a user defined distance to the left and right of the baseline as well as possibly extending the base line - this allows, say, 2 corners of a building to give the direction of the baseline whilst the extensions in the four directions permit the required data to be included, for example the ridge line, porches, building extensions, etc.

Elevation co-ordinates can be further adjusted by specifying a start chainage and vertical offset. These two settings allow multiple elevation projections to be output to the same file without overdrawing each other. LSS will adjust the horizontal offset at the end of each projection.

#### **Note**

When using the Reserved codes be sure they only exist in the first specified label field. It would also be preferable that the initial shot starts with a "." to indicate start of feature.

Ensure that the field separator "," is used when specifying a label after the "-EL"

During an Input Load session, if there is any elevation data in the load file, LSS will automatically invoke the elevation extraction facility. The options in this facility are:

### **Create File**

The first window for elevation extraction requests the name and type of the file to be created.

- LSS Load file - This option converts the elevation data to LSS load input format. When selecting "Next", LSS will generate the next available number for the specified survey. This LSS load input file cannot refer to the current survey.
- DXF - This option converts the elevation data to AutoCAD DXF format. Having given the name of the file to hold the exported DXF data, the following must be specified:
  - i. whether the units are to be millimetres on the plan or metres on the ground.
  - ii. whether levels are to be output for each point (ie. 3-D).
  - iii. the AutoCAD Version number.
  - iv. whether levels text are to be output to the feature's layer.

Link features are written as CONTINUOUS linetypes to a layer whose name is the LSS legend's description. Each layer assumes the LSS display colour. LDUM strings will be processed but each link output as two spot levels.

Point features are written as blocks to a layer whose name is the LSS legend's description. Each layer assumes the LSS display colour.

Determination as to which features will have their levels text output will be whether the Configure / Legend / Point or Link or Spot Level feature has it's "Levels text displayed" field set. The text size and it's number of decimal places will be obtained from the DFLT style. The offset and text direction will be equivalent to sheet / sheet. The value of the levels text is the surveyed level.

Examples of layer names; Spot Levels are written to a layer called "Spot\_Levels", and General Text is written to a layer called "General\_Text".

### **Option**

The option "Include DTM" will include the profile of the DTM.

### **Third dimension**

- Perpendicular distance to baseline - This option displays the observation levels as planimetric distances (Right +ve, Left -ve) away from the selected baseline.
- Surveyed level - This option will display the observation levels as their surveyed / calculated values.

### **Baseline**

This option specifies the baseline for the elevation data.

- Start - Enter or Locate the start position of the baseline.
- End - Enter or Locate the end position of the baseline.
- Horizontal offset - Specifies a constant that will be added to each calculated point (i.e the X ordinate). When a new elevation is requested the width of the previous corridor will be added to the offset value. Thus multiple elevations may be spaced horizontally within a survey.
- Datum - Specifies a constant that will be added to each calculated point (i.e the Y ordinate). Thus multiple elevations may be spaced vertically within a survey.

### Corridor

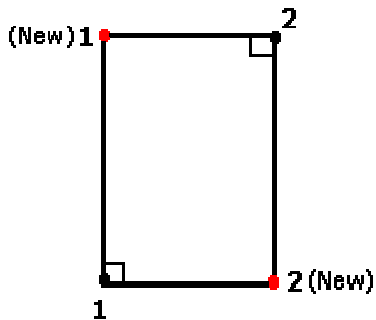
This option enables the extent of the elevation data corridor to be specified.

- Width to left of baseline - Enter or locate a distance away to the left of the reference line to include elevation data.
  - Width to right of baseline - Enter or locate a distance away to the right of the reference line to include elevation data.
  - Extend baseline at start - Enter or locate a distance away from the start the reference line to include elevation data.
  - Extend baseline at end - Enter or locate a distance away from the start the reference line to include elevation data.
  - Include intersections with corridor - This option will determine whether data along a link that is intersected by the corridor is to be included as part of the elevation.
- 

At the end of this command, LSS will prompt to

- Input another Elevation - This option will prompt to specify a new baseline and corridor.
  - Save and Load - This option will save the newly created load file and load the data into the relevant survey.
  - Save - This option will save the newly created load file
  - Delete - This option will delete the specified load file.
- 

### -RW



This reserved code will create an orthogonal shape, by generating two new points assuming the sides lie parallel to the grid.

A link feature can be applied to the rectangle by the code being given to the first observation or by following the -RW on the second observation.

## Elevations

---

Example LSS load file data:

```
1, METRES
1, DMS, VASD

3, STN1,,, 1.600

4, 1, 2580823, 0970411, 17.028, 0.000, STN2
5, 2, 0734628, 0882344, 9.247, 0.000, -EL
5, 2, 0734628, 0882344, 9.247, 0.000, .B
5, 3, 1261037, 0882346, 6.521, 0.000, B
5, 4, 0674419, 0962859, 8.405, 0.000, .VT
5, 5, 1260807, 1002615, 5.196, 0.000, VT
5, 6, 2425707, 1154418, 3.911, 0.000, VT
5, 7, 3011804, 1073733, 2.739, 0.000, PT10
5, 8, 0334759, 1001206, 5.021, 0.000, PTR5
5, 9, 1211903, 0853254, 6.502, 0.000, .W
.
.
5, 37, 1021045, 0642239, 7.479, 0.000, .W
5, 38, 1021048, 0570206, 8.092, 0.000, W
5, 42, 0860324, 0671810, 8.348, 0.000, -RW
5, 43, 0840006, 0641754, 8.795, 0.000, .W
.
.
5, 48, 0754329, 0784207, 9.078, 0.000, W..
5, 49, 0712523, 0641119, 10.264, 0.000, .BC
```

Within the same set-up all data below the -EL code will be included in the elevation calculations which are made to the vertical plane at the chosen baseline position. These points are placed in a nominated load file ready for loading into a new elevation survey.