PRAGOPROJEKT

# ROADPAC'14

**PROGRAM RP31**

## Design of vertical alignment

## *User Guide*

**Release 25.2.2014**

# 1. Introduction

Program DESIGN OF VERTICAL ALIGNMENT is a part of the RoadPAC program package. This program is to be used as one of the first programs for calculation of road vertical alignment specified by tangents in the complex design process of a road. The calculation of the vertical alignment can be performed after calculation of the horizontal alignment, but before the calculation of cross section arrangement or corridor cross sections in any case.

**In recent years users prefer interactive vertical alignment design style, using program VIANIV. See special manual.**

The vertical alignment is given by fixed vertical polyline that is determined by fixed points. These points define either IP's (intersection points, vertexes) or tangents of the tangent polyline. Each tangent can be specified either by pairs of points or by one point and slope. The curves are formed by quadratic parabolas and can be specified:

- by radius of osculation curve (by parameter) or
- by length of tangent or
- by length of the straight segment within two curves or
- by point to be passed through.

The program allows to construct contraflexure curves formed by quadratic parabolas when for each parabola are defined IPs of tangent polygon.

## 1.1 Function of the Program

1) Calculation of the vertical alignment according to specified parameters.
2) Listing of resulting data with main vertical alignment points and writing those data into the .SNI file, if not fatal error occurs.
3) Computing levels and grades in detailed points given by table of chainage. The table of chainage can be read also from the file .SSS and can be saved to the file .SSS again. The new file will contain added chainage according the tab CHAINAGE and chainage of calculated vertical alignment main points levels of the road.
4) Protocol about calculation and results are written into text file .L31.

## 1.2 Processed Data Files

**Input files:**

.V31 - input data
.SNI - vertical alignment file
.SSS - chainage file

**Output files:**

.L31 - output listing
.SNI - vertical alignment file
.SSS - chainage file

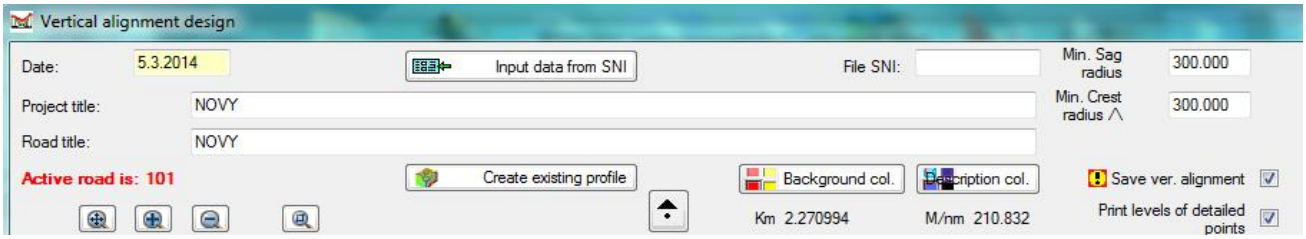
# 2. Input Data

The input data are provided by filling and editing of tables or to click on controls placed on forms appearing on the computer display. Display operation is described in the user guide. See chapter "Introduction". Forms usually incorporate common control part, graphical part if it is useful and input data part. A Picture box, if is placed on form, serve to display current results.

Input data for the program VERTICAL ALIGNMENT are prepared in three data blocks. Some tables can be omitted (depending on type of task).

## 2.1 Control Data Block

Choose Input data in previous main menu to view form for entering of data as shown on picture:




Individual items meaning is following:

**Date** is date of preparing of input data.

**Project Title** and **Road Title**

This arbitrary texts will be inserted into the header of output listings.

**File Name Vertical Alignment and File Name Chainage,** does not have to be specified. In case that those files are used in accordance with demands state below and file name is not specified the program uses the standard name of 'active road' from main menu. The file name specified in proper text box is preferred to name 'active road' from main menu.

Note: In other text are files named 'road '+ 'type'. Name 'road' is either name keyed in table or name 'road' specified main menu.

**Save Vertical Alignment to file:**

- ☐ Vertical alignment will be not saved to file .SNI
- ☑ Vertical alignment will be saved to file .SNI. If this file exists it will be rewritten by newly computed vertical alignment. The name of the file is 'road' .SNI.

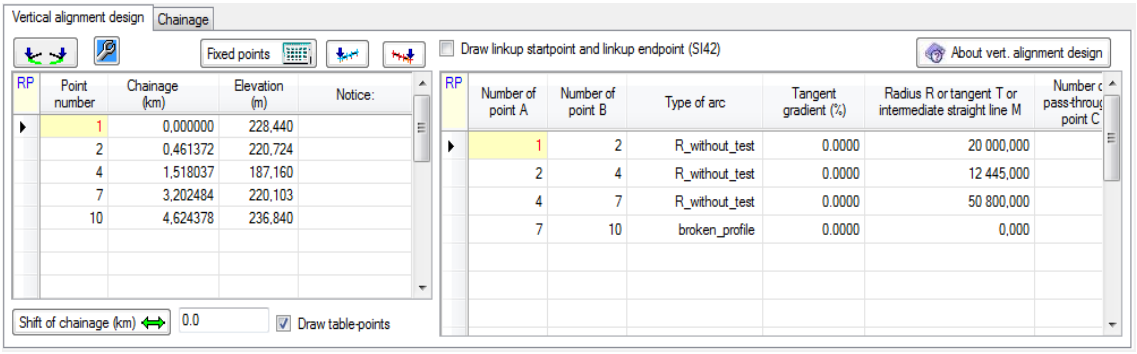
**Print levels of detailed points:**

- ☐ Levels in detailed points will be not calculated and printed.
- ☑ Levels in detailed points of chainage will be calculated and printed.

Two values are specified as MINIMAL CURVE RADIUS. The first one refers to **minimal sag** radius, second one refers to **crest minimal IP** radius. The radius of osculatory circle of quadratic parabola is always specified in meters. Missing of both data causes that testing on minimal radiuses is not performed.

## *2.2 Vertical alignment design.*

Vertical alignment specification is performed in two tables that are placed on tab Vertical alignment design. First (left) table defines IP points and in the second table there are entered Parameters of vertical Alignment.




- In the first mentioned table they are specified support Points of the vertical alignment calculation. One point defined by its number, chainage in km and level is to be specified on one row.
- Point number must be from interval 1 - 999.
- Two points cannot have the same number. Point numbering is not necessary to be in ascending order.

In second table there is specified the vertical alignment parameters, fixed tangent polyline respectively. One tangent with parameters of curve behind tangent is filled in one row. The tangents **must** be sorted in ascendant order (according the ascending chainage).

**Data: Point A, Point B and gradient** are relate with tangent. The tangent can be defined:

- By pair of points A, B. These two points can define either two vertexes of polyline or two support points defining one tangent. Gradient will be not be specified.
- By one point C and gradient (%). The support point must be specified in table.

**Data: Parameter Type, Point C** define parabolic curve behind the tangent. In case of the last element of the vertical alignment there is no curvature behind the last tangent, type is zero and three last data need not to be filled in. In following text there are shown individual possibilities of curve specification.

There are used following symbols:

R radius of osculatory circle in meters

T tangent length in meters

M1 length of straight segment of vertical alignment before curve in meters

M2 length of straight segment of vertical alignment behind curve in meters

C Name (number) of the passed through support points. The curve is defined by passage through this point.

| **curve type** | **ıg** | **parameter** |
|---|---|---|
| 0 | zero radius in vertical alignment, road end or break | - |
| 1 | curve given by radius with test on min. R | R |
| 2 | curve given by radius without test on min. R | R |
| 3 | curve given by tangent length with test on min. R | T |
| 4 | curve given by tangent length without test on min. R | T |
| 5 | curve given by passed through point with test on min. R | C |
| 6 | curve given by passed through point without test on min. R | C |
| 7 | curve given by length of intermediate straight segment in front of curve with test on min. R | M1 |
| 8 | curve given by length of intermediate straight segment behind curve without test on min. R | M2 |
| 9 | zero radius/jump in vertical alignment | - |

Notes:

1) After code 8 must follow one type of fix specified curves. (Type 1 till 6, 9, 0)

2) If the test on minimal radius is not successful the program uses specified minimal radius. In case that data will lead to overlapping of curves, the tangent will be shortened to the end of the previous curve.

## *2.3 Chainage*

This part allows define chainage list in two tables placed on tab Chainage. Chainage “In regular step” is defined in the first table. “Individual chainages” are defined in the second one.

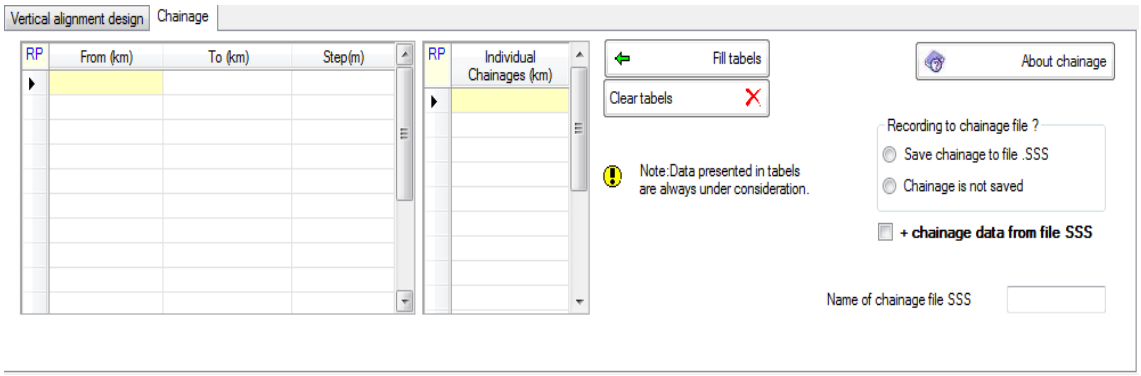
**Read chainage from file:**

☐ The table of chainage will be not read from file .SSS. If necessary the chainage will be read from the tables in tab Chainage.

☑ Table chainage will be read from file .SSS.

**Save chainage to file:**

- ☐ Table of chainage will be not saved.
- ☑ Table of chainage will be written to file .SSS (after calculation of main vertical alignment level points chainage is added to the table). The file name is 'road' .SSS.




The number of rows in the "In Regular step" (left) table is not limited. On one line is defined one segment with regular step. The first data is beginning of segment "**From (km)"**. The second data is end of segment "**To (km)"**. The third value is **step** in metres. New list of chainage will be created in given interval according given step. Segment end will be used only if the chainage is a multiple of the step.

The table "Individual chainages" can content any number of rows. On one row is specified **a** chainage in kilometres. The program elaborates a new table of chainage which is a combination of both above mentioned tables. The duplicates of chainage are eliminated. Work with chainage will be performed as follows: First the file CHAINAGE is processed and then tables of chainage will be read and after sorting, the duplicated chainages will be eliminated. After calculation of vertical alignment the chainage of main vertical alignment points will be added and the table will be saved into file chainage .SSS again. 8000 chainage points is the maximal number.

Note: After clicking on button <Read Data from SNI file> all tables in this form will be automatically filled in by the data from already existing vertical alignment file .SNI. If so is exists.

## *3. DESCRIPTION OF OUTPUT LISTING*

The output listing is generated in the course of the computation in the 'road'.L31 file. Its printing can be controlled from the main menu.

Printing listing contents following information:

1) Protocol about used files 'road '.SNI and 'road' .SSS
2) Protocol about specified and calculated elements of vertical alignment includes following information on each IP:

- IP number
- IP chainage in kilometres
- IP level in metres
- Curve type
- Radius of osculatory circle in metres
- Tangent length in metres
- Deflection of parabolic curve in metres
- Gradient of tangent behind the IP in %
- Distance between IPs in metres
- Length of straight segment between curves in metres

3) Protocol about detailed points of the vertical alignment (written longitudinal profile) contents following information about every point:
   - Chainage of detailed point in kilometres
   - Symbol of detailed point (**)

- symbol of main vertical alignment point (BA,CA,IP,EA), where
  BA is beginning of camber,
  CA is the IP of camber (gradient is 0%)
  IP point with chainage of IP
  EA is end camber
- Level of detailed point in metres above sea level
- Gradient in detailed point in %

## *4. Warning messages*

The program differentiates between fatal errors, which usually cause program termination with unusable results and formal errors that are handled by alternate solution. Fatal errors are marked by *** in reports and formal errors are marked by **. The following table contains list of warning messages and comments to alternate solution:

| **Text of error report** | **Alternative Solution** |
|---|---|
| *** Leading line is missing * 31 | |
| *** Inaccessible type of leading line: nnn | |
| *** Incorrect code function number x = nn | |
| ** First line ignored | |
| *** Between leading data read no marked line * | |
| *** Premature data end | |
| ** line 999 is missing | |
| ** Incorrect line type nnn ignored | |
| ** Formal error, line ignored: (line copy) | |

| | |
|---|---|
| ** More than nnn detailed points, ignored from km n.nnnnnn | |
| *** Incorrect line type nnn ignored | |
| ** More than nnn given points, ignored | |
| ** More than nnn given tangents, ignored | |
| ** Negative value R (T) is taken as positive | |
| *** Same number of two points nnn, refused | |
| ** Point number less than 1, ignored | |
| *** Required point nnn was not specified | |
| *** Determining points of tangent coincide | |
| *** Tangent IP unreal | |
| *** Tangent IP before last IP | |
| ** Code of last IP is not 0 | |
| *** IP no. nn: change of grade equal 0 | |
| *** IP no. nn: through point on tangent | |
| *** IP no. nn: through point between tangents | |
| *** IP no. nn: intermediate straight on the left too big | |
| *** IP no. nn: curve after type 8 is not fixed | |
| ** IP no. nn: overlap with previous curve, replaced by inflection | |
| *** IP no. nn: tangent overlaps next IP | |
| *** IP no. nn: intermediate straight on the right too big | |
| ** End of road moved to the end of last tangent | |
| ** Input corrected - min. radius was not kept at IP | |
| *** Formal error in line: (duplication) | |
| *** While merging chainage and main points of v. nnn alignment nnn main points of v. alignment exceeded | |
| ** While merging chainage and main points of ver. nnnn alignment nnn points are exceeded | |