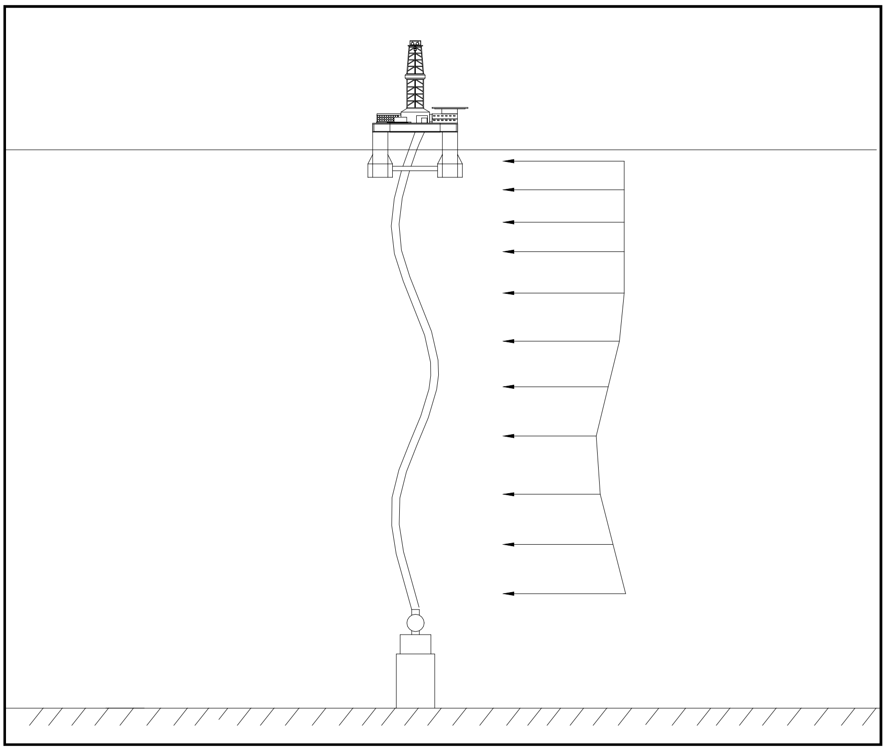
Vortex-Induced Vibration Analysis

Marine Riser Software



User's Manual

DOS Version 8.4

Preface and Copyright

This program package includes VIVA version 8.4 and VIVARRAY version 8.3, which have been re-structured to be an open-access code with the option to use a standard database or a user-specified hydrodynamic database. A three-dimensional incoming current can be specified. The present manual replaces all previous versions (versions 8 and earlier). There are two preparatory programs: *risprep-v8.f* and *ris6-v8.f* for VIVA, which handles a single riser; and two preparatory programs *risprep-vv81.f* and *ris6-vv81.f* for VIVARRAY, which handles two interfering risers, the front (F) and rear (R) risers. There are two programs: *viva-v84.f* for a single riser and *vivarray-vv83.f* for two risers.

The programs described herein have been tested using experimental data and other numerical results. Modifications, however, may be deemed necessary in the future, as experimental and numerical verification continues, which will be communicated in writing to all users. Any problems should be communicated in writing to:

Prof. Michael Triantafyllou
M.I.T. Room 5-226
77 Massachusetts Avenue
Cambridge, Massachusetts 02139
Email: mistetri@triant.mit.edu

or through:

Genesis Engineering, LLC
2550 Gray Falls, Suite 350
Houston, TX 77077
Email: genesis@genesisengineers.com

August 2015

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

VIVA and VIVARRAY are software analysis tools created to study vortex-induced vibrations (VIV) in marine riser applications. The programs are based on an extensive database of VIV experimental results conducted by Professor Michael Triantafyllou at Massachusetts Institute of Technology (MIT) on smooth cylinders, straked cylinders and riser-like sections (modeling choke & kill lines). The software allows three-dimensional analyses of both rigid drilling risers and SCR or lazy-wave risers in current profiles specified by the user. VIVA is for a single riser and VIVARRAY is for two interfering risers, one designated as the front (F) riser and another as the rear (R) riser, subjected to an oncoming variable current.

This manual covers the file structure and analysis procedure in Section 2, coordinate systems and conventions in Section 3, required input in Section 4, and program output in Section 5. Several example problems are described in Section 6.

Updates in Version 8.4 include:

- A new multi-frequency algorithm is implemented that makes VIVA better suited for sheared current profiles as measured in the field, especially the Gulf of Mexico (GoM) current profiles;

- A new fatigue subroutine has been prepared that allows the user to specify single or two-slope fatigue curves, which can be different for each segment of the riser (not applicable to VIVARRAY yet);

- The user can choose between modal damping (damping increases with mode number), or constant damping (all modes have the same damping) for a single riser or two interfering risers. The choice is made in the file *conditions.in* or *conditionsF/R.in* (no need for the separate file *damping_type.in*;

- The multi-mode option is standard, so no choice is needed in *conditions.in* or *conditionsF/R.in*;

- The output file *summ.out* for single riser and *summF/R.out* for two risers has been streamlined to provide both single and multi-frequency data;

- A change that is important only for those using other software to prepare the input for VIVA, there is no need for VIVA to read the file *rispre.in* or *rispreF/R,in*, the data must be appended at the end of *dyn-n.dat* or *dyn-nF/R.dat*;

- For externally specified modes, the curvature natural modes are also needed, in addition to the displacement modes;

- Use a new high Reynolds number database (change in file *no_files.in*);

- Predict high harmonic stresses and calculate fatigue life based on first and third harmonic (need standard input files *free.in* and *forced.in*, which are provided as part of VIVA and VIVARRAY); and

- VIVA can be configured to calculate inline vortex-induced vibrations. It must be noted that it will not calculate the combined inline and cross-flow responses, only the inline responses and the fatigue damage. The database used is *basic_bare-inline* instead of *basic_bare.db*, an example is included in the installation package.

2. FILE STRUCTURE & ANALYSIS PROCEDURE

The software package includes VIVA version 8.4 and VIVARRAY version 8.3, which are set up in a modular format. The programs must be run sequentially in order to create the necessary intermediate files for input to the next module. The run order for VIVA is *risprep-v8.exe*, *ris6-v8.exe*, and finally *viva-v84.exe*, and the run order for VIVARRAY is *risprep-vv81.exe*, *ris6-vv81.exe*, and finally *vivarray-vv83.exe*. Two batch files, *RunVIVA.bat* and *RunVIVARRAY.bat*, have been created to aid users with these procedures. After all user-input files for a specific project have been created, the user should type **RunVIVA** for a single riser or **RunVIVARRAY** for two risers at the DOS prompt to begin the analysis, or double click the icon of *RunVIVA.bat* or *RunVIVARRAY.bat* in *Windows Explorer*. Table 2.1 provides a schematic chart of the required input files and the running sequence for a complete single riser VIV analysis. The required input files and the running sequence for a two riser VIV analysis are similar and are described in

Table 2.2.

The user must define the problem with input files describing the riser configuration, current profile, fatigue curves, boundary conditions, etc. These files are then used in conjunction with the database files from the experimental test results to complete a VIV analysis run. Output is then generated in files as listed in Table 2.1 and Table 2.2.

Table 2.1 VIVA File Structure and Analysis Procedure

Input Files Prepared by User	Input Files from Previous Step	Standard Database	Programs *RUNVIVA.bat*	Output Files
rispre.in			*risprep-v8.exe*	*risout.dat* *risout.out*
risdyn-n.in *vivo-n.in*[1]	*risout.dat*		*ris6-v8.exe*	*dyn-n.dat* *dynout.out*
no_files.in *conditions.in* *risfat.in* *freq.in*[1] *mode_us.in*[1] *curv_us.in*[1] *visc-def.in*[1] *hydro-data*[2]	*dyn-n.dat*	*basic_bare.db* *out1.db* *out2.db* *out3.db* *out4.db* *out5.db* *out6.db* *out7.db* *basic_bare-hr.db* *out_s.db* *free.in*[3] *force.in* [3] *basic_bare-inline*[4] *basic_bare-sche*[5] *basic_bare-07*[6] *basic_bare-14*[6] *out_hr.in*[7]	*viva-v84.exe*	*bend.out* *bend_mm.out* *cdrag.out* *cdrag_mm.out* *clv_dom.out* *fat.out* (*fat1.out*) *fat_h3.out* (*fat1_h3.out*) *fat-mono.out* *fat-mono_h3.out* *fat-multi.out* *fat-multi_h3.out* *freq.out* *out.out* *out_mm.out* *strain.out* *strain_mm.out* *summ.out* (*summ1.out*) *summ_mm.out* (*summ1_mm.out*) *summary.out*

Table 2.2 VIVARRAY File Structure and Analysis Procedure

Input Files Prepared by User	Input Files from Previous Step	Standard Database	Programs *RUNVIVA.bat*	Output Files
rispreF / *R.in*			*risprep-vv81.exe*	*risoutF* / *R.dat* *risoutF* / *R.out*
risdyn-nF / *R.in* *vivo-nF* / *R.in*[1]	*risoutF* / *R.dat*		*ris6-vv81.exe*	*dyn-nF* / *R.dat* *dynoutF* / *R.out*
no_files.in *conditionsF* / *R.in* *risfatF* / *R.in* *freqF* / *R.in*[1] *mode_usF* / *R.in*[1] *curv_usF* / *R.in*[1] *visc-defF* / *R.in*[1] *hydro-data*[2]	*dyn-nF* / *R.dat*	*basic_bare.db* *out1.db* *out2.db* *out3.db* *out4.db* *out5.db* *out6.db* *out7.db* *basic_bare-hr.db* *out_s.db* *free.in*[3] *force.in* [3]	*vivarray-vv83.exe*	*bendF* / *R.out* *bend_mmF* / *R.out* *cdragF* / *R.out* *cdrag_mmF* / *R.out* *clv_dom F / R.out* *fatF* / *R.out* (*fat1F* / *R.out*) *fat_h3F* / *R.out* (*fat1_h3F* / *R.out*) *fat-monoF* / *R.out* *fat-mono_h3F* / *R.out* *fat-multiF* / *R.out* *fat-multi_h3F* / *R.out* *freqF* / *R.out* *outF* / *R.out* *out_mmF* / *R.out* *strainF* / *R.out* *strain_mmF* / *R.out* *summ.outF* / *R* (*summ1F* / *R.out*) *summ_mmF* / *R.out* (*summ1_mmF* / *R.out*) *summaryF* / *R.out*

Notes:

1) Optional input files.

2) *hydro-data* refers to the entire hydrodynamic database, which is user specified, except for a single file, *basic_bare.db*, which is the fallback database (bare cylinder data). The number of files and name are all specified in the input file *no_files.in*.

3) Additional database files for providing high-harmonic data.

4) New database file for in-line VIV responses.

5) New database file for high Reynold's number cases in the GoM sheared current profiles.

6) New database files for the GoM sheared current profiles with subcritical Reynold's number. The file *basic_bare-07* is less conservative than *basic_bare-14*.

7) New database file for high Reynold's number cases.

Additional Notes on the Program Use:

1. Number of Points for Calculations: Note that up to 6001 equidistant points may be used for numerical calculations. Make sure that at least three points are included in each segment, particularly in cases where small segments are used, like when representing a flexible joint. It is best to reduce the number of segments, if needed, to ensure adequate resolution. Consult with M. Triantafyllou for further advice.

2. Maximum Number of Point for Calculations: If familiar with FORTRAN language, you can increase the maximum number of points up to 6001.

3. Running Time: When VIVA calculates natural frequencies and modes, the execution time depends on the third power of the number of points used. For example, doubling the number of points means 8-times longer running time. When the natural frequencies are input externally (calculated by a different software, such as a finite element code), then the time of execution is almost linearly varying with the number of points. Hence, if you run the program for several velocity profiles but with the same top tension and the other material properties, it is best to run it once in the beginning with internally calculated frequencies and modes, and then store the results to use them as input for the remaining cases, as externally input modes. This will reduce the running time greatly, especially when a large number of points is needed.

4. Basic Hydrodynamic Database to Use: The program always requires a standard database for a smooth cylinder (for other riser sections consult the manual, but note that the smooth cylinder database must always be present and specified in *no_files.in*). The file *basic_bare.db* is the standard file for subcritical Reynolds numbers, accounting for in-line motion effects. It is a conservative approach, always assuming the worst case scenario for the effect of in-line motion. It has been extensively tested against uniform and linearly sheared profiles in subcritical Reynolds number. The file *basic_bare-07* has been tested with GoM sheared profiles and is less conservative than *basic_bare-14* for subcritical Reynold numbers. The *basic_bare-hr.db* is a conservative high Reynolds number file. The file *basic_bare-sche* has been tested for high Reynolds number cases in the GoM sheared profiles.

5. Riser-Soil Interaction: You can model the riser-soil interaction by specifying a number of distributed linear springs and linear dashpots.

3. COORDINATE SYSTEMS AND CONVENTIONS

3.1. Coordinate Systems

3.1.1. Global coordinate system

The global orthogonal system (*OXYZ*) (Figure 3.1) has the origin, *O*, fixed on the bottom end of the riser or the front riser in the two riser case. Usually this is the touchdown point, but in some occasions when springs and dampers are used to model the riser/soil interaction, it can be an arbitrary point of the riser lying on the floor.

The *X* axis is pointing vertically upwards. The *Y* and *Z* axes can be arbitrary in principle. For example, the *Y* axis can be chosen as the one pointing "downstream". This has absolute meaning only when the current is unidirectional; then the *Y* component is nonzero, while the *Z* component is zero.

3.1.2. Local coordinate system

When the riser is inclined, we specify a local coordinate system (*oxyz*) at each point of the riser. The direction *x* is tangential to the local riser configuration (when the riser is vertical *x* coincides with *X*). The *y* and *z* coordinates are defined according to the case considered. When the riser is vertical *y* coincides with *Y* and *z* with *Z*. Figure 3.1 depicts a special case with the riser in the *XOY* plane. Figure 3.2 shows a more general case, where the *y* axis is defined to parallel the *XOY* plane

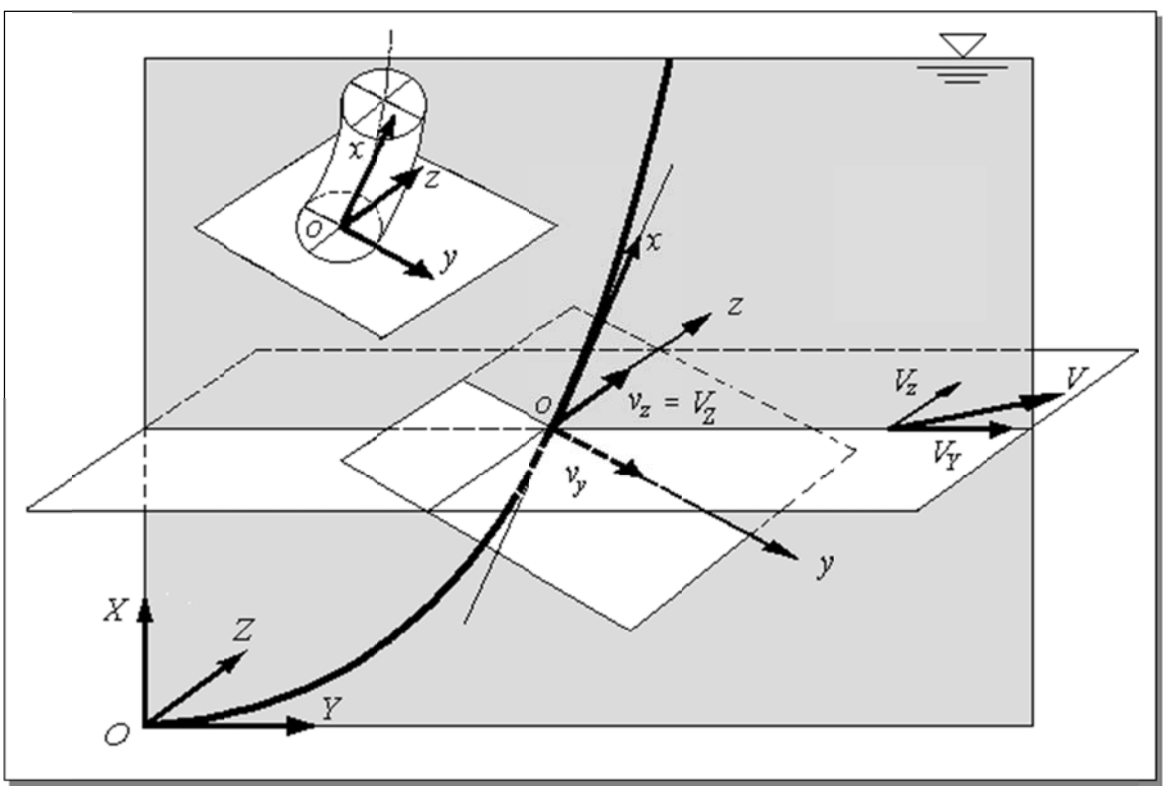


Figure 3.1 Coordinate System

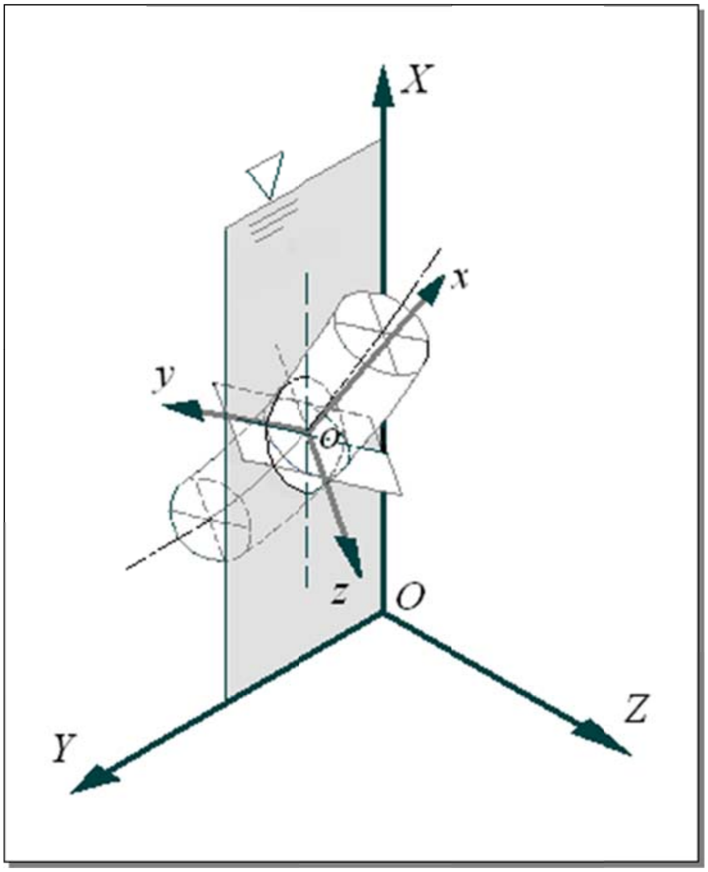


Figure 3.2 Local Coordinate System

3.2. Current Decomposition

The program requires two current components to be specified at each point. We distinguish three cases of increasing complexity:

3.2.1. Vertical, straight-rigid riser

For this case, the local system (*xyz*) coincides with the global system (*XYZ*), hence specify the current along two perpendicular horizontal directions, one along the *Y* axis and the other along the *Z* axis. It is advisable to consider the *y* direction as the "downstream" direction and the *z* direction as the "transverse" direction, although in principle it does not make any difference.

The simplest case is for a unidirectional current. Then you specify the nonzero current velocity along the *y* axis, and set the *z* component equal to zero.

3.2.2. Curved riser, static configuration co-planar (two-dimensional)

When the riser is curved but the static configuration is contained within a plane, then one must specify the velocity components in a plane perpendicular to the local riser axis.

There are still three cases to distinguish:

1. The static configuration is contained within the *xoy* plane. Then the *z* component of the velocity remains the same as the *Z* component of the current, while the *y* component of the current is equal to the *Y* component times the cosine of the inclination angle (zero inclination angles means vertical riser). See Figure 3.1.

2. The static configuration is contained within the *xoz* plane. Then the *y* component of the velocity remains the same as the *Y* component of the current, while the *z* component is equal to the *Z* component times the cosine of the inclination angle (zero inclination angles means vertical riser).

3. The static configuration is not contained in the *xoy* plane nor the *xoz* plane. Rotate the coordinate system so you fall under case (1) or (2) above.

In summary, we consider the riser curvature to lie entirely within either the *XOY* plane or the *XOZ* plane.

3.2.3. Curved riser, static configuration three-dimensional

The program is not strictly valid for this case, because the decomposition of the riser response into two perpendicular planes, even within the linear regime, is not valid. One may argue that the coupling terms are small and proceed with providing the velocity components along perpendicular directions.

Now we define at each point of the riser a plane perpendicular to the local riser centerline configuration (plane *yoz* as shown in Figure 3.2). The *x* axis is tangent to the riser configuration and two Euler angles are needed for its precise definition. To avoid having to deal with internal calculations with Euler angles, we ask the user to specify the current velocity components along the local *y* and *z* directions. The *z* direction is defined as the intersection of the *XOZ* plane and the plane *yoz* normal to the riser. The *y* direction is defined as perpendicular to the axis *z*, contained within the normal plane *ABC*. The system must be right-handed, so the choice of *y* is determined by the consideration that if the riser is vertical *y* must coincide with *Y*.

4. DATA PREPARATION AND INPUT FILES

4.1. *rispre.in*, *rispreF.in* and *rispreR.in*

Rispre.in is the input file to *risprep-v8.exe*, and *rispreF.in* and *rispreR.in* are the input files to *risprep-vv81.exe* for the front and rear riser, respectively. The riser space-out, mass properties, segment characteristics, and type, rigid riser or SCR riser, are specified in these files according to Table 4.1. An example is included below:

Table 4.1 Format of *rispre.in*, *rispreF.in* and *rispreR.in*

Line	Name	Description	Remarks
1	FILENAME	Output File Name - User Specified	Enclosed in ' ', 6 characters long, e.g. 'RISOUT'
2	NSEGM	Number of Segments in Riser	Maximum 1,200
Repeat 3-13 for each riser segment (I = 1 to NSEGM), starting from the bottom end of the riser towards the top end			
3	RLEN(I)	Length of Segment [m]	
4	RMAS(I)[1]	Mass per Length [kg/m],	Include riser pipe and content, auxiliary line pipe and content, and buoyancy module, etc.
5	AMAS(I)[1]	Added Mass per Length [kg/m]	≈ cross-sectional area × density of water
6	WEIGH(I)[1]	Weight in Water per Length [N/m]	Include riser pipe and content, auxiliary line pipe and content, and buoyancy module, etc.
7	DI(I)	Drag Diameter [m]	Normally the buoyancy module diameter, or the outer diameter of bare joints
8	DIP(I), THICK(I)	Pipe Diameter [m], Wall Thickness [m]	Both on the same line separated by a space/tab
9	E(I)	Bending Stiffness, *EI* [N-m^2]	Effective bending stiffness that may include contribution from auxiliary lines
10	NTYPE[2]	Riser Surface Type	7: fitted with strakes 8: fitted with fairings 1: otherwise
11	HTD(I)[3]	Strake Height to Diameter Ratio	Only when NTYPE=7, otherwise omit this line. ICHAR(I) = 1
12	THIC(I), CHORD(I)	Fairing Thickness [m], Fairing Chord Length [m]	Only when NTYPE=8, otherwise omit this line. Input on the same line. ICHAR(I) = 1
13	ICHAR(I)[4]	Riser Cross Section Type	Using Standard VIVA Database 1: smooth cylinder 2: Vetco riser 0 deg 3: Vetco riser 30 deg 4: Vetco riser 60 deg 5: Vetco riser 90 deg 6: Vetco riser 120 deg 7: Vetco riser 150 deg (see Figure 4.1) 8: staggered buoyant-bare (Default 1:3 ratio, with buoyancy module diameter on Line 7) 9. high-Re data from VIVA JIP 10 straked riser (3 strakes with h/D = 0.2 and P/D = 17). NTYPE = 1
14	ICHOICE[5]	Riser Type	1: Rigid riser 2: SCR or lazy-wave riser (with additional input in *vivo-n*; or *vivo-nF* and *vivo-nR* files)

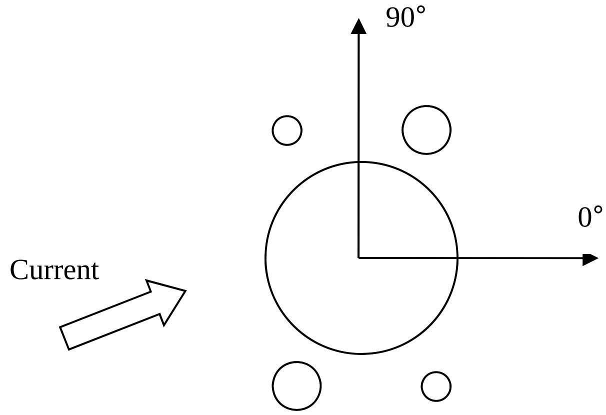


Figure 4.1 VETCO Riser Section

Notes:

1) Metric units are used in this program. The following are two riser joint examples to explain how to prepare and convert joint data from English units.

   Given

   ρ_s = steel density = 489.02 lb_m/ft^3
   ρ_b = density of buoyancy module = 51.17 lb_m/ft^3
   ρ_m = mud density = 15 ppg = 1.8 s.g. = 112.3 lb_m/ft^3
   ρ_w = density of sea water = 64 lb_m/ft^3

   75' × 21"OD × 5/8"thk Bare Joint

   Mass = $\frac{\pi}{4}$ (212 – 19.752) / 144 × 489.02 = 135.86 lbm/ft = 202.6 kg/m

   Displacement = $\frac{\pi}{4}$ (212 – 19.752) / 144 × 64 = 17.8 lbm/ft = 26.5 kg/m

   Wet Weight = 202.6 – 26.5 = -176 kg/m = -1725 N/m

   50.5"OD Buoyancy Module

   Mass = $\frac{\pi}{4}$ (50.52 – 212) / 144 × 51.17 = 588.7 lbm/ft = 877.6 kg/m

   Displacement = $\frac{\pi}{4}$ (50.52 – 212) / 144 × 64 = 736.3 lbm/ft = 1098 kg/m

   Wet Weight = 877.6 – 1098 = -220 kg/m = -2157 N/m

   15 ppg Mud

   Mass = $\frac{\pi}{4}$ 19.752 / 144 × 112.3 = 239 lbm/ft = 356.3 kg/m

   Displacement = $\frac{\pi}{4}$ 19.752 / 144 x 64 = 136.2 lbm/ft = 203 kg/m

   Wet Weight = 356.3 – 203 = 153.3 kg/m = 1502 N/m

   Added Mass of the Bare Joint

$$\text{Added Mass} = \frac{\pi}{4}\ 212 / 144 \times 64 = 154 \text{ lbm/ft} = 230 \text{ kg/m}$$

Added Mass of the Buoyant Joint

$$\text{Added Mass} = \frac{\pi}{4}\ 50.5^2 / 144 \times 64 = 890 \text{ lb}_m\text{/ft} = 1327 \text{ kg/m}$$

Input Data for the Buoyant Joint with 15-ppg Mud
RMAS = 202.6 + 877.6 + 356.3 = 1436.4 kg/m
AMAS = 1327 kg/m
WEIGH = 1725 – 2157 + 1502 = 1070 N/m

Input Data for the Bare Joint with 15-ppg Mud
RMAS = 202.6 + 356.3 = 558.9 kg/m
AMAS = 230 kg/m
WEIGHT = 1725 + 1502 = 3227 N/m

2) NTYP=1 except when "typical strakes" or "typical fairings" are to be used – these use data from the literature, which are not very detailed – to use detailed strake data see below

   NTYP=7 then typical strakes are used -- in this case add a line below the line of NTYP providing the height/diameter ratio, also specify ICHAR=1

   NTYP=8 then typical fairings are used -- in this case add a line below the line of NTYP providing the thickness/diameter ratio and chord/diameter ratio, also specify ICHAR=1

3) Vortex-Suppressing Strakes:

   Basic parameters for strakes are their pitch, *P*, and height, *h*. The height is measured perpendicular to the surface of the riser.

   Data used in the program for standard strake configuration (NTYPE = 7) are valid for pitch to diameter ratios, *P/D*, larger than 4, up to a value of 20, preferably in the range of 5 to 8; and height to diameter ratios, *h/D*, larger than 0.05, preferably in the range of 0.06 to 0.15. The user is allowed to specify the value of the height to diameter ratio *h/D* or HTD in the program. A previous restriction of the program that no longer applies is that if more than one segment has strakes, they must all have the same height to diameter ratio.

   If the strake configuration is close to the tested one (3 strakes with *h*/*D* = 0.2 and *P*/*D* = 17), I_CHAR may be input as 10 to allow the program to use the hydrodynamic data from experiments. In this case, NTYPE must be entered as 1 with no HTD input.

4) ICHAR=1 if a standard bare cylinder is to be used -- always supply an external file *bare_basic.db*

   ICHAR=2, 3, 4, 5, 6, 7, 8 if VETCO risers (bare with 4 kill and choke lines) are to be used, 2 for an angle of attack of 0 degrees, 3 for 30 degrees, 4 for 60 degrees, 5 for 90 degrees, 6 for 120 degrees, and 7 for 150 degrees.

   ICHAR=8 for staggered buoyancy-bare pipe.

   ICHAR=9 for high Reynolds number data from the database "*basic_bare-hr.db*".

ICHAR=10 for straked riser with pitch/diameter=17, and height/diameter=25%.

5) ICHOICE=1 for straight riser

CHOICE=2 if the riser is curved or has an unusual static shape. The user must then give the static solution externally in a file called *vivo-n.in*, *vivo-nF.in* or *vivo-nR.in*

4.2. *risdyn-n.in*, *risdyn-nF.in* and *risdyn-nR.in*

The VIVA dynamics program requires *risdyn-n.in* as input to *ris6-v8.exe*. VIVARRAY requires *risdyn-nF.in* for the front riser and *risdyn-nR.in* for the rear riser as input to *ris6-vv81.exe*. The files specify the current profile, riser top tension, and various numerical calculation parameters. Additionally, the files specify the top horizontal positions (*Y*, *Z*) in global coordinates for the front and rear risers in the VIVARRAY analyses. Table 4.2 provides the data input format.

Table 4.2 Format of *risdyn-n.in*, *risdyn-nF.in* and *risdyn-nR.in*

Line	Name	Description	Remarks
1	FILENAME1	Output File Name	Enclosed in ' ', 6 characters long, e.g. '*RISOUT*'
2	FILENAME2	Output File Name	Enclosed in ' ', 6 characters long, e.g. '*DYNOUT*'
3	TENUP	Riser Top Tension [N]	0 for free-standing riser
4	NP	Number of Points along riser	Points number in analysis. Minimum 200, maximum 6,001.
5	ITMAX	Maximum Iterations	Minimum 30
6	NV	Number of Current Data	Maximum 200
For each point from I = 1 to NV, repeat line 7 to specify the current profile from the top of the riser to the mudline (depth measured positive down from WL, i.e. riser portion above the water line has negative depth, see note 1 for example):			
7	XV(I), VY(I), VZ(I)[1]	Depth [m], Velocity *Y*-Component [m/s], Velocity Z-Component [m/s]	Current velocity can be positive or negative. For Definition of coordinate system, see Figure 3.1.
8	RHO	Water Density [kg/m^3]	Typically 1025 kg/m^3
9	ZETA[2]	Damping Ratio	Generally 0.01 to 0.07, see comment below
10	NO_SPRI	Number of Intermediate Lateral Supports	
For each lateral support from I = 1 to NO_SPRI, repeat line 11 from the uppermost support to the bottom.			
11	X_SPRI(I), C_SPRI(I), B_SPRI(I)	Distance along riser [m]. Spring constant [N/m]. Damper constant [N-sec/m].	The distance is measured along the riser from the top end.
12	Y, Z[3]	Distance of the top of the riser in the horizontal plane from the origin *O* fixed on the bottom end of the riser	*Y* and *Z* in global coordinates, only for VIVARRAY

Notes:

1) Riser Segment above Water:

For the section of the riser above the waterline, a 0 knot current velocity must be inputted at a point **just** above the waterline and at the top of the riser. For example, a 110 m long straight-vertical riser in 100 m of water depth and a constant 1.0 m/s current profile would be entered as follows:

Depth [m]	Velocity *Y* [m/s]	Velocity *Z* [m/s]
-10.00	0.0	0.0
-0.01	0.0	0.0
0.00	1.0	0.0
100.00	1.0	0.0

2) ZETA, Structural Damping:

The user must specify the structural damping as a fraction of the critical damping of the system. Typically the damping is in the range of 0.01 to 0.07 (1% to 7% critical). Trends are as follows:

- For a uniform flow, this parameter may play a very significant role in the drag amplification and must be evaluated accurately.
- For shear flows, the role of critical damping is significantly less important.

An estimate of the importance of the structural damping can be found from the parameter Q:

$$Q = 2\pi^3 \zeta (St)^2 \left(1 + \frac{\rho_c}{\rho_w} \right) \qquad [4.1]$$

where

St - Strouhal Number, take 0.17
ρ_c - density of pipe material (usually steel)
ρ_w - density of water

Q is the ratio of the structural over the hydrodynamic damping, hence some general trends for the parameter are:

- If Q ~ 1.0 (or higher), ζ is very important
- If Q << 1.0, ζ does not have a large effect
- The higher the shear flows, structural damping is less important

3) *Y* and *Z* (only for VIVARRAY)

This specifies the position (*Y*, *Z*) at the top of the riser in the horizontal plane from the global origin *O* (usually *Y* is in the direction of the surface current and *Z* traverse to it). *Y* shows how far downstream the top of the riser is from the origin and *Z* how far traversed at the top. These quantities are very important for the interaction studies. This input is mandatory for the vertical risers where *vivo-n* files are not required. In the case of curved risers, the *vivo-n* files are used to specify the relevant position along the water depth, the *Y* and *Z* coordinates here will be superseded.

4.3. *vivo-n.in, vivo-nF.in* and *vivo-nR.in* (required only for SCR / Lazy-Wave Riser option)

The SCR or lazy wave riser static configuration must be supplied by the user to *ris6-v8.exe* or *ris6-vv81.exe* when ICHOICE is inputted as 2 in file *rispre.in*, *rispreF.in*, or *rispreR.in*. For catenary risers, the portion of the riser on the ocean floor from touchdown point to the well is truncated in general, although this portion may be modeled with a series intermediate lateral supports. The file *vivo-n.in* or *vivo-nF.in* and *vivo-nR.in* should be created according to Table 4.3. A graphic depiction is given in Figure 3.1 and Figure 3.2.

Table 4.3 Format of *vivo-n.in*, *vivo-nF.in* and *vivo-nR.in* (for SCR Riser Analysis)

Line	Name	Description	Remarks
1	NPP, DEPTH[1]	Number of Points, Water Depth [m]	The *point* here is referred to the input points in this file, and can be different with NP in *RISDYN-N.in*
For each point from I = 1 to NPP, repeat the following data from the top of the riser and going down. See Figure 3.1.			
I+1	SX1(I)	Distance along the Riser to point I [m]	0 means point I is at the bottom end and the top point has SX1(I) equal to the total modeled length of the riser.
	TEX(I)	Axial Static Tension [N]	
	DTEX(I)[2]	Change in Static Tension [N/m]	Negative when riser wet weight > 0, can be estimated with Equations 4.2 to 4.4
	XXX(I)[3]	Vertical Distance [m]	Measured from the bottom end, and always positive.
	YYX(I)[4]	Horizontal Distance along *Y* [m]	Measured in global coordinate
	ZZX(I)[4]	Horizontal Distance along *Z* [m]	Measured in global coordinate
	VELEY(I)[5]	Current Velocity along *y* axis [m/s]	Normal current component
	VELEZ(I)[5]	Current velocity along *z* axis [m/s]	Normal current component
	URIST(I)[6]	Riser Velocity in *XY* plane [m/s]	Dummy parameter, not used in program
	VRIST(I)[6]	Riser Velocity in *XZ* plane [m/s]	Dummy parameter, not used in program

Notes:

1) For catenary risers, the portion of the riser lying on the floor shall be truncated in general. With the introduction of intermediate lateral support, the grounded portion may be modeled as lateral supports.

2) Numerical estimate for DTEX can be calculated with Equations 4.2 to 4.4 as follows:

For *i* = 2, 3, … (NPP-1),
$$\frac{dT_i}{ds} \approx \frac{T_{i+1} - T_{i-1}}{2\Delta s_i} \qquad [4.2]$$

For *i* = 1,
$$\frac{dT_1}{ds} \approx \frac{T_2 - T_1}{\Delta s_1} \qquad [4.3]$$

For *i* = NPP,
$$\frac{dT_{NPP}}{ds} \approx \frac{T_{NPP} - T_{NPP-1}}{\Delta s_{NPP-1}} \qquad [4.4]$$

3) The vertical coordinate at every riser point, I, in the global coordinate system.

4) Although they are not used in VIVA, they are essential, however, to VIVARRAY. They are the horizontal coordinates at every riser point, I, in the global coordinate system.

5) When the riser contains large curvatures, one must specify the velocity components on a plane perpendicular to the local axis of the riser. In previous VIVA versions this was done internally to the program, but in a three-dimensional shape this is impossible, since there are two Euler angles needed. Hence the user must make sure to input the two components of velocity in a plane perpendicular to the local riser configuration.

6) URIST(I) and VRIST(I) are riser self-velocities and are omitted in the current version.

4.4. *no_files.in*

no_files.in lists the hydrodynamic databases to be used in VIVA or VIVARRAY. Table 4.4 provides the data input format.

Table 4.4 Format of *no_files.in*

Line	Name	Description
1	N_OFL[1]	Number of hydrodynamic files including file *basic_bare.db*
		Repeat line 2 for every hydrodynamic file from I = 1 to N_OFL
2	NM(I)[2]	Name of the file
	NA(I)	Number of non-dimensional frequencies to be specified in this file
	NB(I)	Number of C_d values to be given for each frequency as function of A/D

Notes:

1) The number of files is 10 for the provided *no_files.in*, which may be changed based on the available database files.

2) The sequence is important because it must correspond with the ICHAR value given in the *rispre.in*, *rispreF.in* or *rispreR.in*. For example, ICHAR=4 means to use the fourth file in the *no_file.in* below *out3.db*. The list below gives the "standard hydrodynamic database":
ICHAR=1 for *basic_bare.db* is for a bare cylinder
ICHAR=2 for *out1.db* is for a VETCO-section at 0 deg
ICHAR=3 for *out2.db* is for a VETCO-section at 30 deg
ICHAR=4 for *out3.db* is for a VETCO-section at 60 deg
ICHAR=5 for *out4.db* is for a VETCO-section at 90 deg
ICHAR=6 for *out5.db* is for a VETCO-section at 120 deg
ICHAR=7 for *out6.db* is for a VETCO-section at 150 deg
ICHAR=8 for *out7.db* is for a staggered buoyancy-bare configuration
ICHAR=9 for *basic_bare-hr.db* is for high-RE data
ICHAR=10 for *out_s.db* is for a straked cylinder with strakes P/D=17, h/D=-0.25

Notes: If the standard database files are used, no-files.in should be input as:

10

basic_bare.db 36 0
out1.db 14 5
out2.db 14 5
out3.db 14 5
out4.db 14 5
out5.db 14 5
out6.db 14 5
out7.db 11 5
basic_bare-hr.db 36 5
out_s.db 15 4

The optional databases, as mentioned in Section 2, should be input as:

basic_bare-sche 36 0
basic_bare-07 36 0
basic_bare-14 36 0
out_hr.in 78 61

4.5. *conditions.in*, *conditionsF.in* and *conditionsR.in*

These files specify the control variables and boundary conditions at the top and bottom of the riser for both rigid and SCR risers. There are up to seven lines of input as seen in Table 4.5.

Table 4.5 Format of *conditions.in*, *conditionsF.in* and *conditionsR.in*

Line	Name	Description	Remarks
1	DEF_CON	Control Variable	0: by internal calculation 1: import modal frequencies only 2: import modal frequencies, modal shapes and curvatures[1]
2	RNU[2]	Kinematic Viscosity [m^2/s]	Normally 1.114 x 10^{-6} m^2/s for sea water 0: input through *visc-def.in*
3	I_HIRE[3], NO_HRF	Control Variable	- I_HIRE = 1, NO_HRF = 1: use the old approach: re-scaling the low-Reynolds number data (default) - I_HIRE = 2: use the external high Reynolds number database, NO_HRF specifies the sequential file number in *no_files.in* corresponding to the high Reynolds database, or 9 for *out_hr.db*
4	IBC(1)[4], IBC(2)[4]	Upper Boundary Condition, Lower Boundary Condition	0: pinned connection 1: fixed end 2: end connected to an elastic rotational spring 3: free end
4a[5]	CUPK	Upper Rotational Spring, [1/rad-m]	See Equation 4.5, only needed when IBC(1) = 2
4b[5]	CLOK	Lower Rotational Spring, [1/rad-m]	See Equation 4.5, only needed when IBC(2) = 2
5	G_M[6]	Damping Type	0: constant damping 1: modal damping

Notes:

1) If DEF_CON = 2, both the displacement and curvature natural modes are required.

2) If RNU = 0, a default value of 1.114 x 10^{-6} m^2 /s would be used by the program. Alternatively, the user may specify the viscosities by *VISC-DEF.in* file with RNU input of zero.

3) When I_HIRE = 1, the built-in high Reynolds number database will be used. When I_HIRE = 2, the file number of *basic_bare-hr.db*, or a user-specified high Reynolds number database must be specified as in *no_files.in*.

4) A riser cannot have both upper and lower ends free.

5) Only required when the corresponding boundary condition has been specified as 2, where an end is connected to an elastic rotational spring.

6) The choice is between (a) modal damping, where if one specifies for example, 3% damping ratio, it means that each mode separately has the same modal damping ratio; or (b) constant damping where only the first mode damping is calculated and the same value of damping (not damping ratio) is used for all other modes (hence higher order modes have increasingly smaller damping ratio).

When a rotational spring (flex-joint) is chosen as the boundary condition for either top or bottom of the riser, the stiffness of the upper or lower rotational spring must be defined, respectively. The boundary condition for a riser beam at the end of the flex-joint is governed by Equation 4.5.

$$K\frac{dy}{ds} + EI\frac{d^2y}{ds^2} = 0 \qquad \text{or} \qquad C\frac{dy}{ds} + \frac{d^2y}{ds^2} = 0 \qquad [4.5]$$

where

K - rotational-spring constant for the flex-joint [N-m/rad]

E I - bending stiffness of the riser beam [N-m^2]

y - lateral deflection of the riser [m]

S - distance along the riser axis [m]

C - stiffness input to VIVA for the flex-joint [1/m-rad], $C = \frac{K}{EI}$

Notes: If the user input for K is in the unit of [N-m/deg], then

$$C = \frac{180\,[\text{deg}]}{\pi\,[\text{rad}]}\ \frac{K\,[\text{Nm/deg}]}{EI\,[\text{Nm}^2]} = 57.296\frac{K}{EI} \quad [1/rad\text{-}m]$$

If the user input for K is in the unit of [lb_f-ft/deg] and EI is in the unit of [lb_f-ft^2], then

$$C = \frac{180\,[\text{deg}]}{\pi\,[\text{rad}]}\ \frac{K\,[\text{lb}_\text{f}\,\text{ft/deg}]}{EI\,[\text{lb}_\text{f}\,\text{ft}^2]\times 0.3048\,[\text{m/ft}]} = 187.978\frac{K}{EI} \quad [1/rad\text{-}m]$$

4.6. *risfat.in, risfatF.in* and *risfatR.in*

This file defines the constants A, B and stress concentration factor to be used in fatigue calculation. Results of VIV dynamic analysis are used to calculate fatigue life along the riser length for a given stress concentration factor and S-N curve. Table 4.6 provides the data input format.

The fatigue constants, A and B, of some often used fatigue curves from API RP-2A and HSE are given in Table 4.7 for IFFAT =1 or $N = (A / S_{amplitude})^B$.

Table 4.6 Format of *risfat.in*, *risfatF.in* and *risfatR.in*

Line	Name	Description	Remarks
1	IFFAT	Type of S-N Curves	1: $N = (A / S_{amplitude})^B$ 2: $N = (A / S_{range})^B$ 3: $N = A / S_{amplitude}^B$ 4: $N = A / S_{range}^B$ 5: $N = (A\_1 / S_{amplitude})^{B\_1}$; $N = (A\_2 / S_{amplitude})^{B\_2}$, If $S_{amplitude} < S\_W$ 6: $N = (A\_1_i / S_{amplitude})^{B\_1i}$; $N = (A\_2_i / S_{amplitude})^{B\_2i}$, If $S_{amplitude} < S\_W_i$ N number of cycles to fatigue failure S VIV stress amplitude or range in N/m² or Pa i the ith segement Options 5 and 6 are only available for VIVA
2	A, B	Fatigue Constants	As described in S-N curve types
2a	A_1, B_1, A_2, B_1, S_W, SCF	Fatigue Constants for a double slope fatigue curve	For IFFAT = 5 only
2b	A_1_i, B_1_i, A_2_i, B_1_i, S_W_i, SCF_i	Fatigue Constants for a double slope fatigue curve of the *i*th segment	For IFFAT = 6 only, repeat line 2b for *i* = 1 to NSEGM, from bottom to top.
3	SCF	Stress Concentration Factor	For IFFAT = 1 through 4 only

Table 4.7 Fatigue Constants for IFFAT = 1

Class	A (Pa)	B
API X	1.3729E+09	4.38
API X'	1.9115E+09	3.74
HSE B	2.8279E+09	4
HSE C	3.9197E+09	3.5
HSE D	5.7408E+09	3
HSE E	5.0773E+09	3
HSE F	4.2885E+09	3
HSE F2	3.7639E+09	3
HSE G	3.1307E+09	3
HSE W	2.7058E+09	3
HSE T	5.6533E+09	3

It is assumed that each VIV mode is not in the steady state, but participates as a random process. Hence, gamma function is applied in fatigue life calculations as in the Equation 4.6 for IFFAT = 1.

$$N = (\sqrt{2}A / S_{amplitude})^B / \Gamma(1 + B/2) \quad [4.6]$$

4.7. *freq.in*, *freqF.in* and *freqR.in*

These files are optional if the user wishes to have them as an external input to specify the natural frequencies of the risers rather than letting it be calculated by VIVA or VIVARRAY. If DEF_CON = 1 or DEF_CON = 2 in the *conditions.in*, *conditionsF.in* or conditions*R.in*, then the natural frequencies of the riser should be provided sequentially in *freq.in*, *freqF.in* or *freqR.in*. Table 4.8 provides the data input format.

Table 4.8 Format of *freq.in or freqF.in or freqR.in*

Line	Name	Description	Remarks
For each mode I, repeat line 1			
1	I	Sequence Number	Maximum NP/7[1]
	FREQ(I)	Modal Frequency [rad/sec]	

Notes:

1) The maximum number of frequencies, *M*, that the program will consider is up to *np_r*/7 (one-seventh of the number of points). The minimum number is 1, but one should consider that the program predictions may be inaccurate if too few modes are specified. The program automatically finds the number *M* from the file, so there is no need to specify this number.

4.8. *modes_us.in, modes_usF.in* and *modes_usR.in*

These are also optional files. They allow the user to specify the natural modes in addition to the natural frequencies. If DEF_CON = 2 in *conditions.in*, *conditionsF.in* or *conditionsR.in*, then in addition to the natural frequencies, the natural displacement modes of the riser should be defined in the file *modes_us.in*, *modes_usF.in* or *modes_usR.in*. Table 4.9 provides the data input format.

Table 4.9 Format of *modes_us.in*, *modes_usF.in* and *modes_usR.in*

Line	Name	Description	Remarks
For each mode I, specified in *FREQ.in*, repeat line 1 for every nodes from J = 1 to NP_R starting from the top end			
1	X(J)	Axial Distance of Node J [m]	Measured from the top end
	Y_R(I,J)	Real Part of Displacement or Curvature at Node J [m]	
	Y_I (I,J)	Imaginary Part of Displacement or Curvature at Node J [m]	Equals 0 if the modes are standing modes

4.9. *curv_us.in, curv_usF.in* and *curv_usR.in*

These are also optional files. When DEF_CON = 2 in *conditionsF.in* or *conditionsR.in*, then specify, in addition to the natural frequencies and the natural displacement modes, the natural curvature modes of each riser in the files *curv_us.in*, *curv_usF.in* or *curv_usR.in*. Table 4.10 provides the data input format.

Table 4.10 Format of *curvs_us.in*, *curvs_usF.in* and *curvs_usR.in*

Line	Name	Description	Remarks
For each mode I, specified in *freq.in*, repeat line 1 for every nodes from J = 1 to NP_R starting from the top end			
1	X(J)	Axial Distance of Node J [m]	Measured from the top end
	Y_R(I,J)	Real Part of Displacement or Curvature at Node J [m]	
	Y_I (I,J)	Imaginary Part of Displacement or Curvature at Node J [m]	Equals 0 if the modes are standing modes

4.10. *visc-def.in, visc-defF.in* and *visc-defR.in*

If you specified a zero value in *conditions.in*, *conditionsF.in* or *conditionsR.in* for the viscosity, then you must specify the value of the viscosity (in m^2/s) in *visc-def.in*, *visc-defF.in* or *visc-defR.in*, point-by-point for the riser. Table 4.11 provides the data input format.

Table 4.11 Format of *visc-def.in*, *visc-defF.in* and *visc-defR.in*

Line	Name	Description	Remarks
Repeat line 1 for every nodes from I = 1 to NP_R starting from the top end			
1	VISC(I)	Viscosity at Node I [m^2/s]	Normally 1.114×10^{-6} m^2/s for sea water

4.11. User-Specified Database Files for *viva-v84.exe* and *vivarray-vv83.exe*

Users may specify their own hydrodynamic databases by replacing or appending to the existing ones for the VIV calculations.

Table 4.12 provides the format of these databases.

Table 4.12 Format of Hydrodynamic Database File

Line	Name	Description
Repeat the following lines (1 to [NB(I) + 1]) for every frequency, J = 1 to NA(I) (see Table 4.4)		
1	Frequency (J)	Non-dimensional frequency
	Lift (J)	Lift coefficient in phase with velocity at $A/D = 0$
	Added-mass (J)	The corresponding added mass
	First slope (J)	The first slope of the lift curve
	Second slope (J)	The second slope of the lift curve
	A/D (J)	The value of A/D where the slope changes
Repeat Line 2 for every C_d value, K = 1 to NB(I) (see Table 4.4)		
2	AMD(J,K)	The value of A/D
	CD(J,K)	The corresponding drag coefficient

5. PROGRAM OUTPUT FILES

Every VIVA or VIVARRAY module will generate a set of output files. Some serve as input files for the consequent module, some can be used for checking the correctness of input, and the rest are the analysis results. Files of the last two categories are detailed in the following sections.

5.1. *risprep-v8.exe* or *risprep-vv81.exe*

risout.out (or with the name inputted in Line 1 of *rispre.in* or *rispreR.in* and *rispreR.in*)

The user can use this file to check input data in *rispre.in* or *rispreF.in* and *rispreR.in*.

5.2. *ris6-v8.exe* or *ris6-vv81.exe*

dynout.out (or with the name inputted in Line 2 of *risdyn-n.in* or *risdyn-nF.in* and *risdyn-nR.in*)

The user can use this file to check input data in *risdyn-n.in* or *risdyn-nF.in* and *risdyn-nR.in* and, if required, *vivo-n.in* or *vivo-nF.in* and *vivo-nR.in*.

5.3. *viva-v84.exe and vivarray-vv83.exe*

- *bend.out*, *bendF.out* and *bendR.out*

  These files contain the bending moment and stress response along the riser length of each excited mode. It is appropriate for plotting and provides three columns: (1) *S* coordinate, measured along the riser length from the top end (m); (2) amplitude of VIV bending moment in (N-m); and (3) amplitude of VIV bending stress in (N/m^2) or (Pa).

- *bend_mm.out*, *bend_mmF.out* and *bend_mmR.out*

  These files contain the multi-frequency bending moment and stress response. It is appropriate for plotting and provides three columns: (1) *S* coordinate, measured along the riser length from the top end (m); (2) equivalent amplitude of VIV bending moment in (N-m); and (3) equivalent amplitude of bending stress in (N/m^2) or (Pa).

- *cdrag.out*, *cdragF.out* and *cdragR.out*

  These files contain the drag coefficient along the risers of each excited mode. It is appropriate for plotting and provides two columns: (1) *S* coordinate, measured along the riser length from the top end (m); and (2) drag coefficient at that point.

- *cdrag_mm.out*, *cdrag_mmF.out* and *cdrag_mmR.out*

These files contain the drag coefficient for multi-frequency response. It is appropriate for plotting and provides two columns: (1) *S* coordinate, measured along the riser length from the top end (m); and (2) drag coefficient at that point.

- *clv_dom.out* (VIVARRAY only)

  This file summarizes the lift coefficients of the rear riser in the current wake of the front riser. It is appropriate for plotting and provides two columns: (1) *S* coordinate, measured along the riser length from the top end (m); and (2) lift coefficient at that point.

- *fat.out*, *fatF.out* and *fatR.out*

  These files summarize the fatigue life of the risers for all the excited individual modes as well as the multi-frequency mode.

- *fat1.out; fat1F.out & fat1R.out*

  These files contain the same information as files *fat.out, fatF.out* or *fatR.out*, but without notations.

- *fat_h3.out*, *fat_h3F.out* and *fat_h3R.out*

  These files contain the minimum single-frequency and multi-frequency fatigue life (in years) including the effect of high harmonics. This file is relatively short in length as it calculates the minimum fatigue life along the entire length for both single and multi-frequency response.

- *fat1_h3.out*, *fat1_h3F.out* and *fat1_h3R.out*

  These files contain the same information as file *fat_3h.out*, *fat_3hF.out* or *fat_3hR.out* but without notations.

- *fat-mono.out*, *fat-monoF.out* and *fat-monoR.out*

  These files provide sequentially for each excited mode the fatigue life (in year) calculated at each point along the length of the riser. It contains the following items:

  a) The first line provides the number of excited modes (*INDEX*) and the number of points (*NP_R*) at which the fatigue life is provided.

  b) For each excited mode, it provides two columns of numbers: (1) *S* coordinate, measured along the riser length from the top end (m); and (2) corresponding fatigue life at the point.

- *fat-mono_h3.out*, *fat-mono_h3F.out* and *fat-mono_h3R.out*

  These files provide sequentially for each excited mode the fatigue life (in year) calculated at each point along the length of the riser including the effect of high harmonics. It contains the following items:

  a) The first line provides the number of excited modes (*INDEX*) and the number of points (*NP_R*) at which the fatigue life of both single and multi-frequency is provided.

b) For each excited mode, it provides two columns of numbers: (1) *S* coordinate, measured along the riser length from the top end (m); and (2) corresponding fatigue life at the point.

- *fat-multi.out*, *fat-multiF.out* and *fat-multiR.out*

  These files contain the fatigue life (in year) for the multi-frequency responses, calculated at each point along the length of the riser. The file contains two columns: (1) *S* coordinate, measured along the riser length from the top end (m); and (2) corresponding fatigue life.

- *fat-multi_h3.out*, *fat-multi_h3F.out* and *fat-multi_h3R.out*

  These files contain the fatigue life (in year) for the multi-frequency responses, calculated at each point along the length of the riser, including the effect of high harmonics. The file contains two columns: (1) *S* coordinate, measured along the riser's length from the top end (m); and (2) corresponding fatigue life.

- *out.out*, *outF.out* and *outR.out*

  These files contain the motion responses of each excited mode. It is appropriate for plotting and provides two columns: (1) *S* coordinate, measured along the riser length from the top end; and (2) VIV amplitude at that point. All these data are in meters (m).

- *out_mm.out*, *out_mmF.out* and *out_mmR.out*

  These files contain the multi-frequency motion responses. It is appropriate for plotting and provides two columns: (1) *S* coordinate, measured along the riser length from the top end; and (2) equivalent VIV amplitude at that point. All these data are in meters (m).

- *freq.out*, *freqF.out* and *freqR.out*

  These files contain the modal frequencies of the risers in the absence of the external current.

- *strain.out*, *strainF.out* and *strainR.out*

  These files contain the bending strain along the riser length of each excited mode. It is appropriate for plotting and provides two columns: (1) *S* coordinate, measured along the riser length from the top end (m); (2) amplitude of VIV bending strain.

- *strain_mm.out*, *strain_mmF.out* and *strain_mmR.out*

  These files contain the multi-frequency bending strain along the riser length. It is appropriate for plotting and provides two columns: (1) S coordinate, measured along the riser length from the top end (m); (2) equivalent amplitude of bending strain.

- *summ.out*, *summF.out* and *summR.out*

  This file contains a brief summary of all the modes that were excited by VIV, including their frequency and amplitude.

- *summ1.out*, *summ1F.out* and *summ1R.out*

These files contain the same information as *summ.out*, *summF.out* and *summR.out* in a tabulated format. There are 4 columns: (1) mode number; (2) initial frequencies in calm water; (3) coupled frequencies; and (4) maximum VIV amplitude for that mode.

- *summ_mm.out*, *summ_mmF.out* and *summ_mmR.out*

  These files contain a brief summary of the multi-frequency VIV response, including the dominant and subdominant, if present, modes, their frequency and amplitude, and the maximum RMS response and its location.

- *summ1_mm.out*, *summ1_mmF.out* and *summ1_mmR.out*

  These files contain the same information as file *summ_mm.out*, *summ_mmF.out* and *summ_mmR.out*, but without notations.

- *summary.out*, *summaryF.out* and *summaryR.out*

  These files contain a summary of the modal responses, which includes (1) the mode number, (2) the frequency of response in Hz, (3) the amplitude in m, (4) the first stress harmonic amplitude in N/m^2, (5) the third stress harmonic amplitude in N/m^2, and (6) the probability assigned to that mode.

6. EXAMPLE PROBLEMS

To aid the user in becoming familiar with the software, several example problems are outlined in this section that cover the various riser options available during analysis, such as rigid risers in connected and disconnected modes, steel catenary risers, and free-standing risers. The standard database files are assumed for the examples hence *no_files.in* should be input as in Section 4.4.

To run examples of single riser, copy the files from the desired example directory to the directory with VIVA programs installed and type **RunVIVA** at the DOS prompt.

To run examples of a two-riser-array, copy the files from the desired example directory to the directory with VIVARRAY programs installed and type **RunVIVARRAY** at the DOS prompt.

6.1. Single Riser Example 1

This example is a 10,000 ft water depth rigid riser in GOMEX 3.5 knots loop current (Figure 6.1). The input files for VIVA are as follows:

- *rispre.in*

```
'RISOUT'
 9
 617.22
 983.1533
 245.697
 6579.024
 0.5524501
 0.5524501     0.028575
 3.235463E+08
 1
 2
 297.18
 1887.917
 1571.11
 2451.479
 1.397
 0.5524501     0.028575
 3.235463E+08
 1
 1
 297.18
 1887.917
 1571.11
 2451.479
 1.397
 0.5524501     0.028575
 3.235463E+08
 1
 1
 297.18
 1887.917
 1571.11
 2451.479
 1.397
 0.5524501     0.028575
 3.235463E+08
```

```
1
1
320.04
1887.917
1571.11
2451.479
1.397
0.5524501     0.028575
3.235463E+08
1
1
297.18
1868.492
1542.674 2540.016
1.3843
0.5524501     0.028575
3.235463E+08
1
1
868.6801
1830.216
1486.582
2714.753
1.3589
0.5524501     0.028575
3.235463E+08
1
1
22.86
983.1533
245.697
6579.024
0.5524501
0.5524501     0.028575
3.235463E+08
1
```

```
2                                  0.5524501     0.028575
15.24                              3.235463E+08
983.1434                           1
245.697                            2
6579.122                           1
0.5524501
```

- *risdyn-n.in*

```
'RISOUT'                           152.4         1.028889      0
'DYNOUT'                           304.8         0.5144444     0
 1.067573E+07                      457.2         0.2572222     0
 600                               3352.8        0.1028889     0
 30                                1025
 6                                 0.01
 0             1.800555      0     0
 91.44         1.749111      0
```

- *conditions.in*

```
0
1.114E-06
2             9
0             0
1
```

- *risfat.in*

```
1
1.3729E+09    4.38
1.5
```

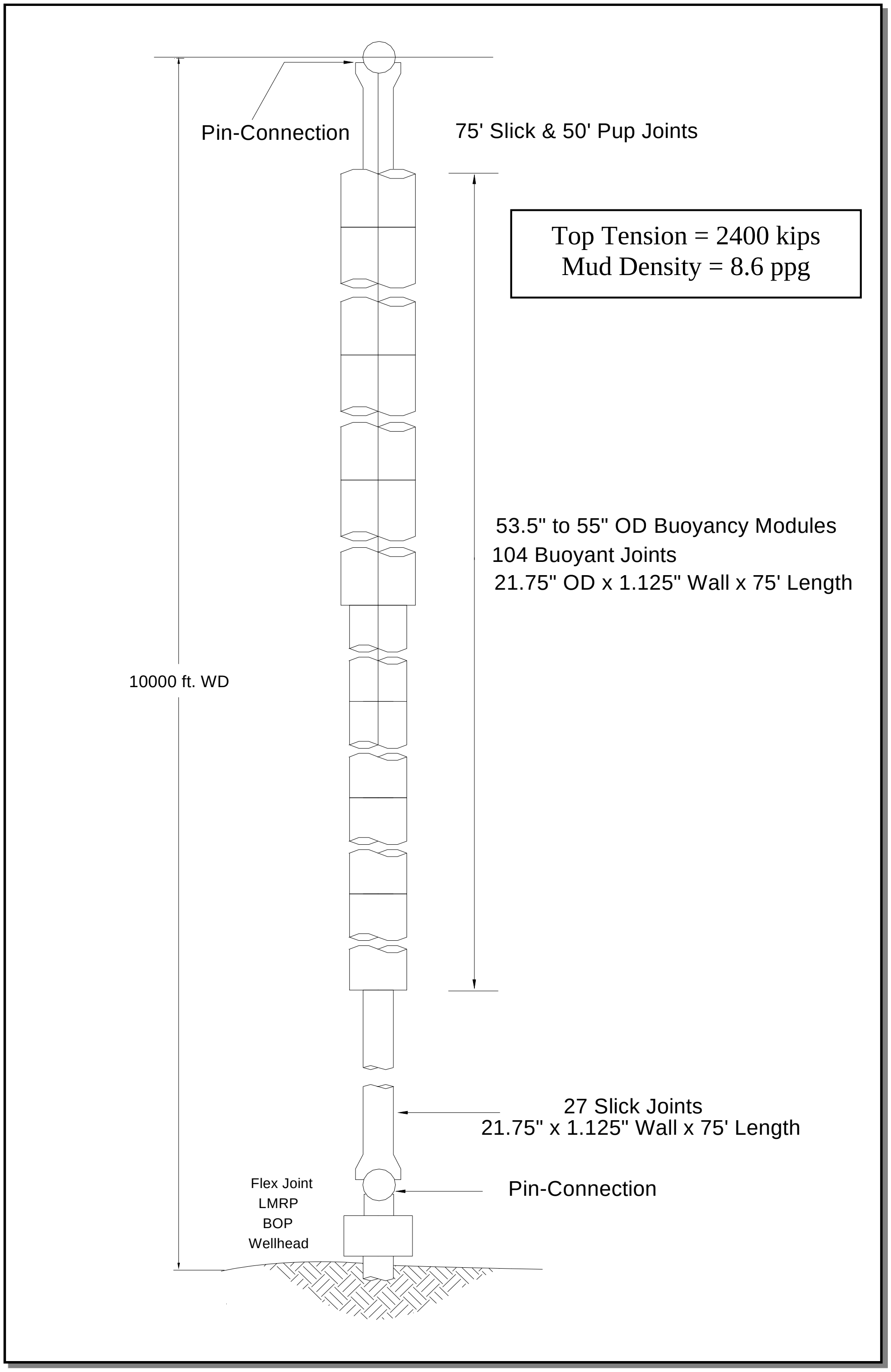


Figure 6.1 Example Problem 1:

10000 ft Riser in GOMEX 3.5 kts Loop Current

Pinned Upper and Lower Boundary Conditions

6.2. Single Riser Example 2

This example is the above example in disconnected, or hang-off, condition (Figure 6.2). The input files for VIVA are as follows:

- *rispre.in*

```
'RISOUT'
 10
 6.4008
 23583.99
 673.1103
 199460.4
 0.9144
 0.9144        0.209549993276596
 2.428339E+09
 1
 2
 617.22
 983.1533
 245.697
 6579.024
 0.5524501
 0.5524501     0.028575
 3.235463E+08
 1
 1
 297.18
 1887.917
 1571.11
 2451.479
 1.397
 0.5524501     0.028575
 3.235463E+08
 1
 1
 297.18
 1887.917
 1571.11
 2451.479
 1.397
 0.5524501     0.028575
 3.235463E+08
 1
 1
 297.18
 1887.917
 1571.11
 2451.479
 1.397
 0.5524501     0.028575
 3.235463E+08
 1
 1
 320.04
 1887.917
 1571.11
 2451.479
 1.397
 0.5524501     0.028575
 3.235463E+08
 1
 1
 297.18
 1868.492
 1542.674
 2540.016
 1.3843
 0.5524501     0.028575
 3.235463E+08
 1
 1
 868.6801
 1830.216
 1486.582
 2714.753
 1.3589
 0.5524501     0.028575
 3.235463E+08
 1
 1
 22.86
 983.1533
 245.697
 6579.024
 0.5524501
 0.5524501     0.028575
 3.235463E+08
 1
 2
 15.24
 983.1434
 245.697
 6579.122
 0.5524501
 0.5524501     0.028575
 3.235463E+08
 1
 2
 1
```

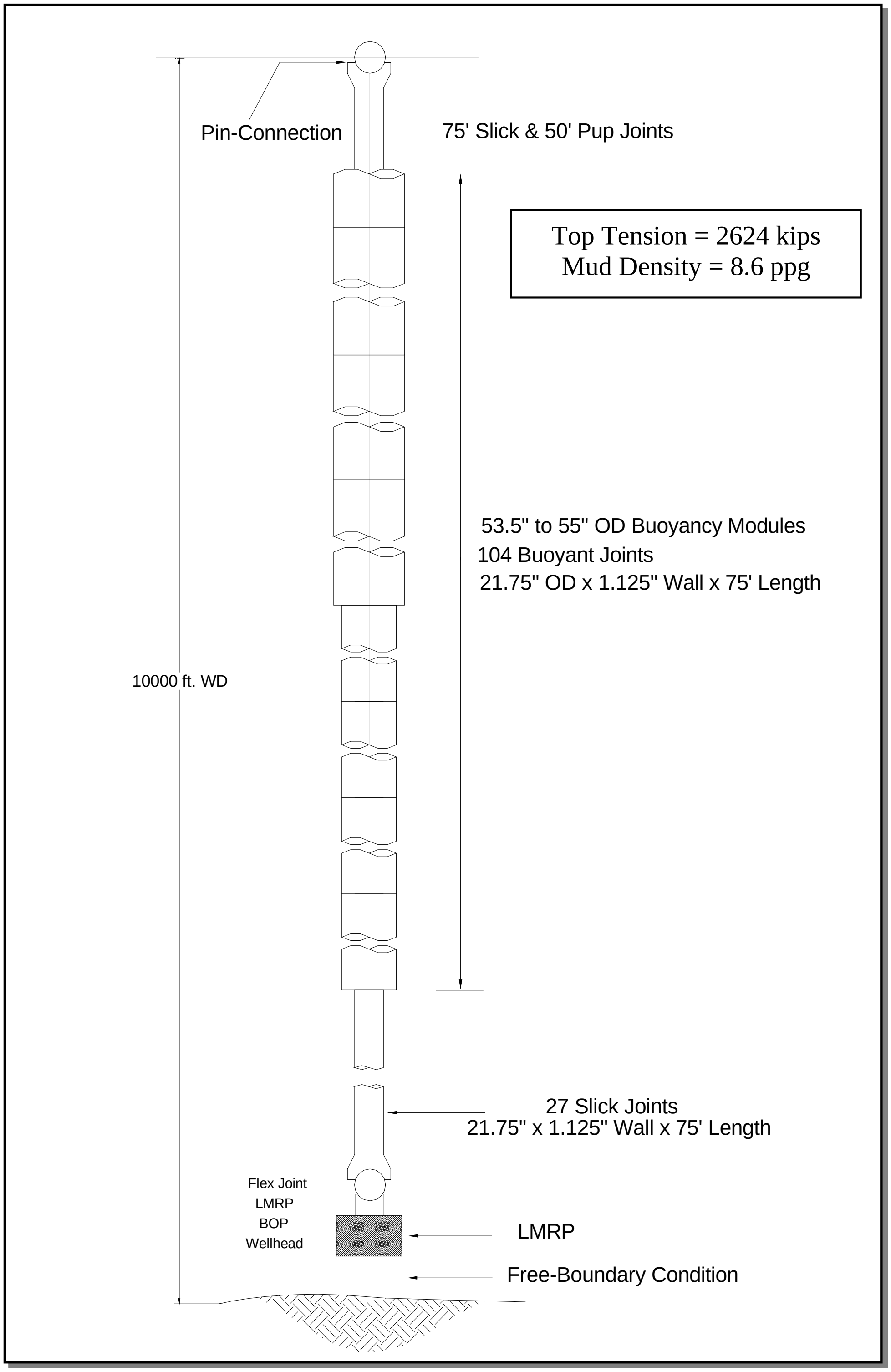


Figure 6.2 Example Problem 2:

10000 ft Riser in GOMEX 3.5 kts Loop Current

Pinned Upper and Free Lower Boundary Conditions

- *risdyn-n.in*

```
'RISOUT'
'DYNOUT'
 1.039463E+07
 600
 30
 6
 0              1.800555      0
 91.44          1.749111      0
 152.4          1.028889      0
 304.8          0.5144444     0
 457.2          0.2572222     0
 3352.8         0.1028889     0
 1025
 0.01
 0
```

- *conditions.in*

```
 0
 1.114E-06
 2              9
 0              3
 1
```

- *risfat.in*

```
 1
 1.3729E+09     4.38
 1.5
```

6.3. Single Riser Example 3

This example is a 3000 ft riser in WOS 1.0 m/s current (Figure 6.3). The input files for VIVA are as follows:

- *rispre.in*

```
'RISOUT'
 5
 76.2
 720.5848
 229.0445
 4239.197
 0.5334001
 0.5334001     0.015875
 1.729392E+08
 1
 5
 304.8
 1219.775
 960.3248
1961.964
 1.0922
 0.5334001     0.015875
 1.729392E+08
 1
 1
 441.96
 1219.775
 960.3248
 1961.964
 1.0922
 0.5334001     0.015875
 1.729392E+08
 1
 1
 45.72
 1314.422
 1098.998
 1529.984
 1.1684
 0.5334001     0.015875
 1.729392E+08
 1
 1
 76.2
 874.4907
 229.0445
 5493.689
 0.5334001
 0.5334001     0.015875
 1.729392E+08
 1
 5
 1
```

- *risdyn-n.in*

```
'RISOUT'
'DYNOUT'
 4448221
 600
 30
 8
-30.48         0              0
-0.01          0              0
 0             0.9980222      0
 100.584       0.9414333      0
 198.12        0.8179666      0
 396.24        0.5813222      0
 609.6         0.463          0
 914.4         0.463          0
 1025
 0.01
 0
```

- *conditions.in*

```
0
1.114E-06
2             9
0             2
1.796762E-02
1
```

- *risfat.in*

```
1
1.3729E+09    4.38
1.5
```

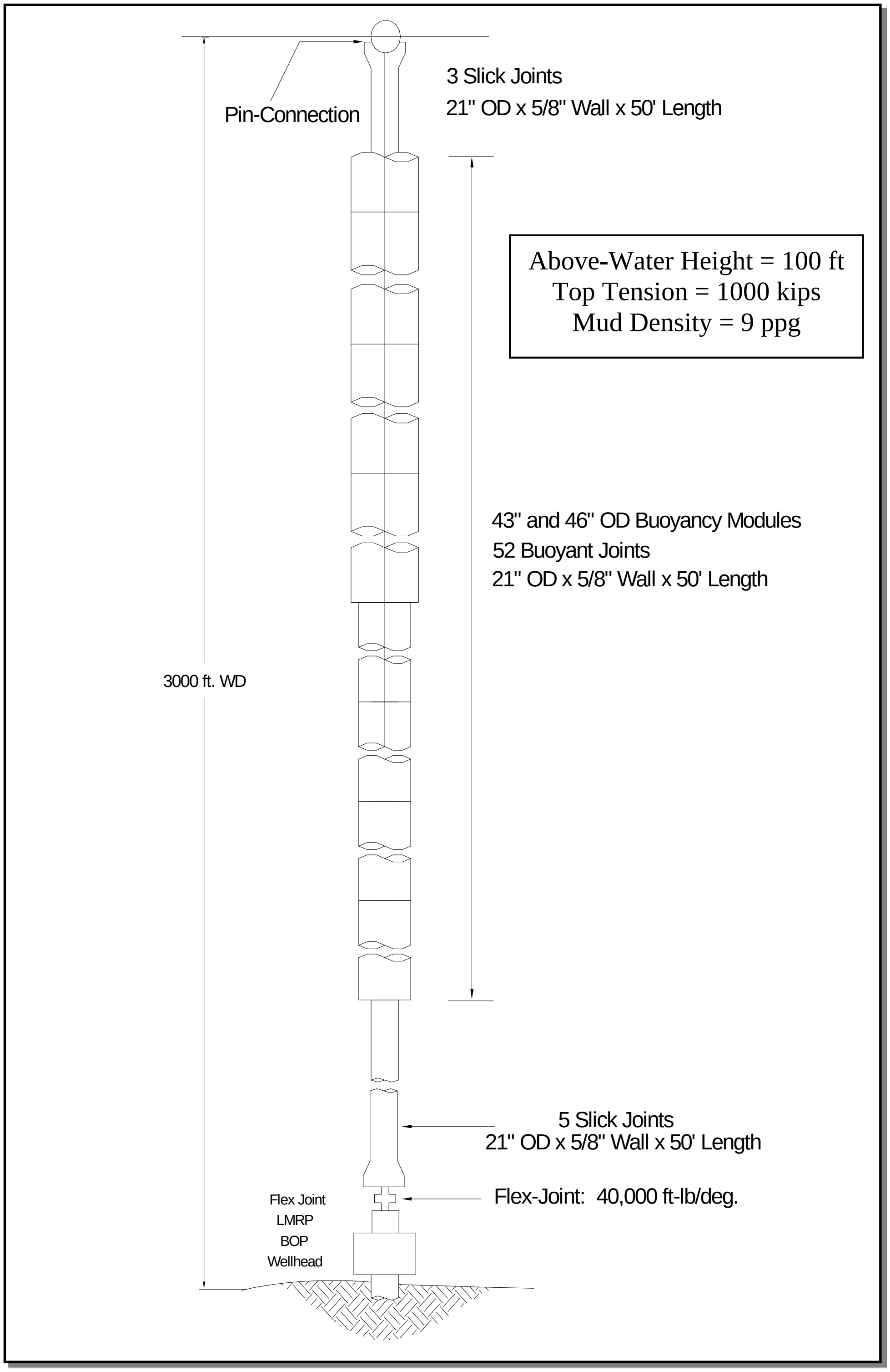


Figure 6.3 Example Problem 3:

3000 ft Riser in WOS 1.0 m/s Current

Pinned Upper and Flex-Joint Lower Boundary Conditions

6.4. Single Riser Example 4

This example is a 2790 ft water depth steel catenary riser in GOMEX 3.0 knots eddy current (Figure 6.4). The input files for VIVA are as follows:

- *rispre.in*

```
'RISOUT'
 1
 1036.32
 140.4619
 98.19438
 543.5759
 0.34925
 0.3048        0.0174625
 3.264837E+07
 1
 1
 2
```

- *risdyn-n.in*

```
'RISOUT'
'DYNOUT'
 560475.9
 400
 30
 9
 0             1.543333      0
 38.1          1.440444      0
 76.2          1.183222      0
 114.3         0.8745555     0
 152.4         0.6173333     0
 228.6         0.463         0
 304.8         0.2572222     0
 457.2         0.2057778     0
 457.2031      0             0
 1025
 0.01
 0
```

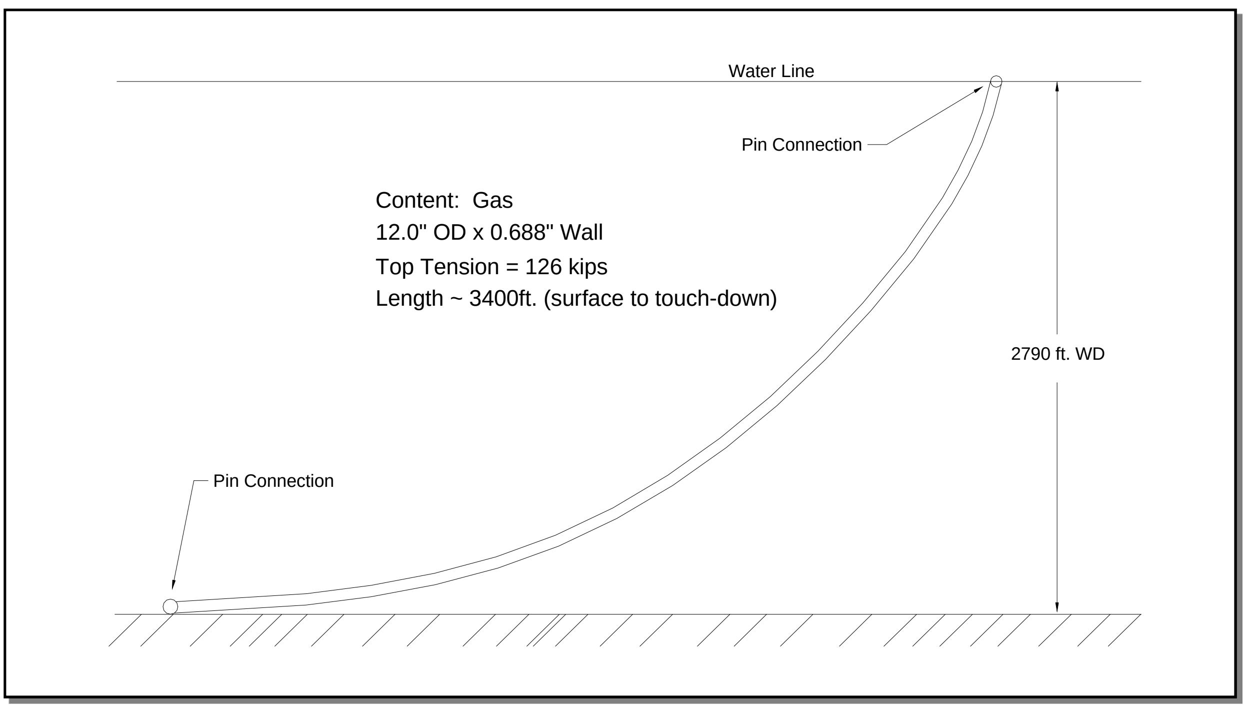


Figure 6.4 Example Problem 4:

SCR Riser in 2790 ft Deep Water, GOMEX 3.0 kts Current

Pinned Upper and Lower Boundary Conditions

- *vivo-n.in*

```
 69           850.392
 1035.46       560475.9     -542.6597     850.392       0             0
1.514513      0             0             0
 1020.05       552113.3     -531.1013     835.2618      0             0
1.473911      0             0             0
 1004.639     544106.5     -519.4116     820.1528      0             0
1.432986      0             0             0
 989.2192     536099.7     -516.8723     805.0408      0             0
1.363361      0             0             0
 973.8299     528181.8     -515.9528     789.9624      0             0
1.26285       0             0             0
 958.4406     520219.5     -515.9528     774.8931      0             0
1.162425      0             0             0
 943.0512     512301.7     -517.5436     759.842       0             0
1.043339      0             0             0
 927.671       504294.9     -520.7396     744.8032      0             0
0.9235044     0             0             0
 912.3          496288.1     -516.5658     729.7827      0             0
0.8123015     0             0             0
 896.9288      488414.8     -513.72       714.7834      0             0
0.7128755     0             0             0
 881.5609      480496.9     -515.3253     699.7934      0             0
0.613607      0             0             0
 866.199       472579.1     -514.1724     684.8216      0             0
0.5752402     0             0             0
 850.8492      464705.7     -510.0277     669.8834      0             0
0.5452821     0             0             0
 835.4995      456921.3     -511.852      654.9512      0             0
0.515338      0             0             0
 820.1711      449003.5     -519.2365     640.0526      0             0
0.4855103     0             0             0
 804.8305      440996.7     -519.1927     625.1631      0             0
0.4557326     0             0             0
 789.4991      433078.9     -515.1939     610.301       0             0
0.4185099     0             0             0
 774.1798      425205.5     -515.5588     595.4634      0             0
0.3792276     0             0             0
 758.8697      417287.7     -509.8964     580.6532      0             0
0.3400881     0             0             0
 743.5596      409592.3     -508.4516     565.8613      0             0
0.3010743     0             0             0
 728.2495      401718.9     -518.7695     551.1028      0             0
0.2622205     0             0             0
 712.9485      393712.1     -513.3113     536.3627      0             0
0.2445061     0             0             0
 697.6597      386016.7     -507.7073     521.6622      0             0
0.2393365     0             0             0
 682.371       378187.8     -510.772      506.983       0             0
0.2341698     0             0             0
 667.0914      370403.4     -509.619      492.3435      0             0
0.2290001     0             0             0
 651.8209      362619       -508.4662     477.7314      0             0
0.2238481     0             0             0
 636.5596      354879.1     -505.5474     463.1619      0             0
0.2186832     0             0             0
 621.2891      347183.7     -507.0651     448.623       0             0
0.2135019     0             0             0
 606.0308      339399.3     -510.2758     434.1206      0             0
0.2083551     0             0             0
 590.7786      331614.9     -504.5988     419.673       0             0
0.2031889     0             0             0
```

575.5295	324008.5	-498.9655	405.2621	0	0
0.1980223	0	0	0		
560.2894	316402	-499.2574	390.8908	0	0
0	0	0	0		
545.0586	308795.6	-502.3805	376.5835	0	0
0	0	0	0		
529.8308	301100.1	-502.5264	362.3219	0	0
0	0	0	0		
514.6091	293493.7	-499.6077	348.1212	0	0
0	0	0	0		
499.3813	285887.2	-495.5652	333.9724	0	0
0	0	0	0		
484.1809	278414.2	-490.0195	319.9029	0	0
0	0	0	0		
468.9714	270985.7	-490.0195	305.8912	0	0
0	0	0	0		
453.771	263512.7	-493.2447	291.9619	0	0
0	0	0	0		
438.5798	255995.2	-487.5385	278.1208	0	0
0	0	0	0		
423.3885	248700.1	-480.2708	264.3622	0	0
0	0	0	0		
408.2003	241405	-477.5271	250.6919	0	0
0	0	0	0		
393.0213	234198.9	-474.8856	237.1313	0	0
0	0	0	0		
377.8514	226992.8	-474.725	223.6927	0	0
0	0	0	0		
362.6907	219800	-471.3539	210.373	0	0
0	0	0	0		
347.5299	212700.6	-464.9034	197.1934	0	0
0	0	0	0		
332.36	205699.1	-458.4821	184.151	0	0
0	0	0	0		
317.2115	198799.9	-448.923	171.2823	0	0
0	0	0	0		
302.069	192100.9	-445.6832	158.6027	0	0
0	0	0	0		
286.9204	185299.6	-442.6331	146.1121	0	0
0	0	0	0		
271.7902	178698.4	-429.5277	133.8529	0	0
0	0	0	0		
256.6599	172301.9	-419.6768	121.8408	0	0
0	0	0	0		
241.5296	165998.7	-413.1826	110.1029	0	0
0	0	0	0		
226.4085	159802.4	-403.3463	98.67291	0	0
0	0	0	0		
211.2904	153801.7	-387.0303	87.59338	0	0
0	0	0	0		
196.1693	148099.1	-370.51	76.89191	0	0
0	0	0	0		
181.0603	142601.1	-357.5214	66.63239	0	0
0	0	0	0		
165.9606	137298.8	-341.1325	56.85132	0	0
0	0	0	0		
150.8608	132299	-317.8552	47.61279	0	0
0	0	0	0		
135.761	127699.5	-294.826	38.98083	0	0
0	0	0	0		
120.6703	123398.1	-271.6217	31.02258	0	0
0	0	0	0		
105.5797	119501.5	-245.1046	23.80188	0	0
0	0	0	0		

```
 90.48903      116000.7     -212.1078      17.41321     0            0
0             0             0             0
 75.40143      113100.5     -179.038       11.90247      0           0
0             0             0             0
 60.32906      110600.6     -149.281       7.373108      0           0
0             0             0             0
 45.24146      108598.9     -105.9955      3.861816     0            0
0             0             0             0
 30.15996      107402.3     -59.58691      1.441711     0            0
0             0             0             0
 15.0815       106801.8     -26.54631      0.1432495    0            0
0             0             0             0
0              106601.6    -13.26586      0                   0      0
0             0             0             0
```

- *conditions.in*

```
 0
 1.114E-06
 2             9
 0             0
 1
```

- *risfat.in*

```
 1
 1.3729E+09    4.38
 1.5
```

6.5. Single Riser Example 5

This example is a 5000 ft water depth free-standing riser in Brazil current (Figure 6.5). The input files for VIVA are as follows:

- *rispre.in*

```
'RISOUT'
 5
 297.18
 890.4704
 245.697
 5898.948
 0.5524501
 0.5524501     0.028575
 3.235463E+08
 1
 5
 617.22
 1794.639
 1571.11
 1771.403
 1.397
 0.5524501     0.028575
 3.235463E+08
 1
 1
 297.18
 1730.013
 1571.11
 1137.639
 1.397
 0.5524501     0.028575
 3.235463E+08
 1
 1
 228.6
 1794.639
 1571.11
 1771.403
 1.397
 0.5524501     0.028575
 3.235463E+08
 1
 1
 6.096
 88402.84
 119664
-600241.2
 12.192
 0.5524501     0.028575
 3.235463E+08
 1
 1
 1
```

- *risdyn-n.in*

```
'RISOUT'
'DYNOUT'
 0
 600
 30
 7
 76.2          1.006135      0
 152.4         0.4372778     0
 243.84        0.3189555     0
 304.8         0.3189555     0
 457.2         0.4475666     0
 609.6         0.4012666     0
 1524          0.4012666     0
 1025
 0.01
 0
```

- *conditions.in*

```
 0
 1.114E-06
 2             9
 3             0
 1
```

- *risfat.in*

```
 1
 1.3729E+09    4.38
 1.5
```

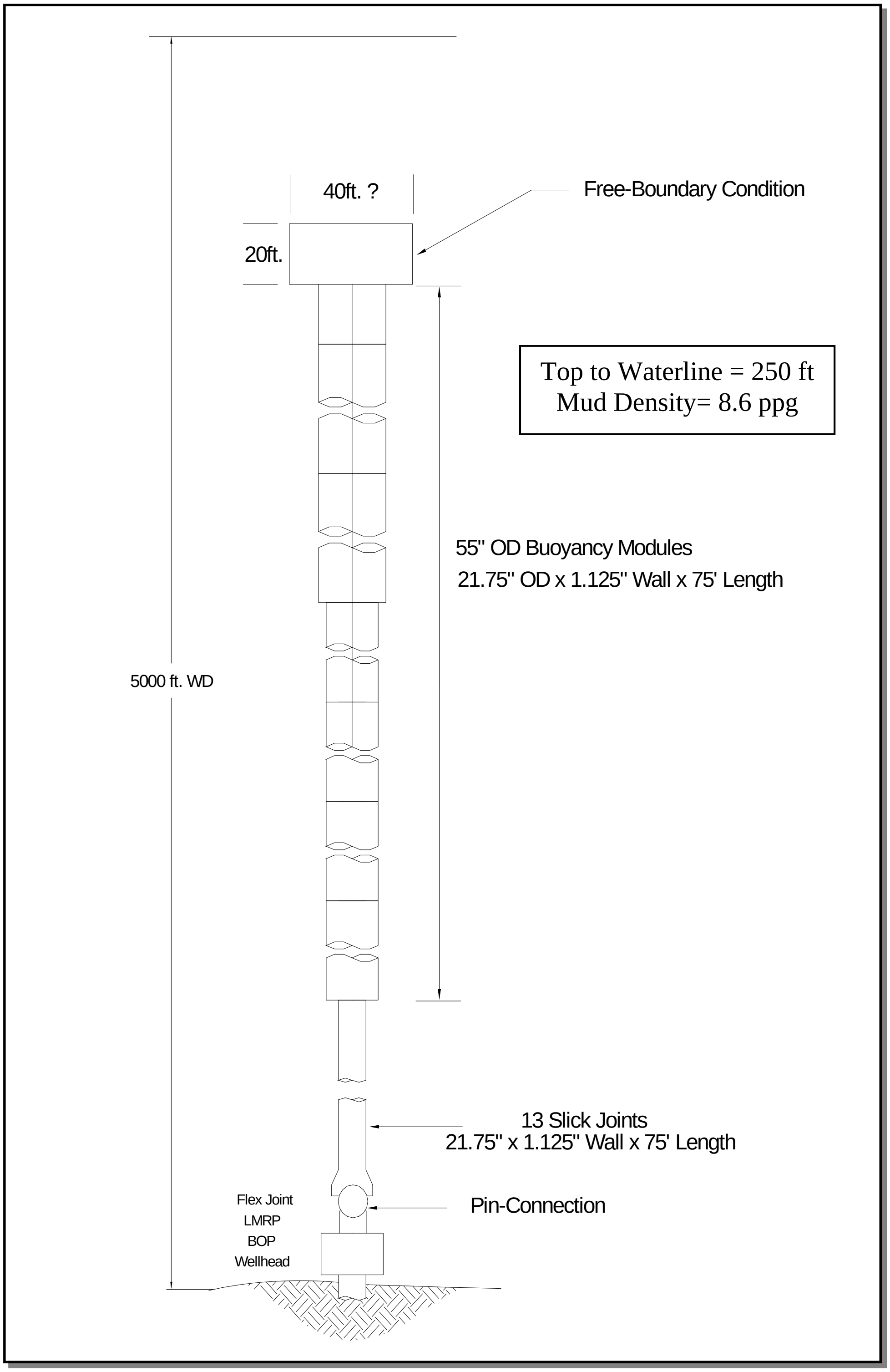


Figure 6.5 Example Problem 5:

4750 ft Free-Standing Riser in 5000 ft WD, Brazil Current Profile

Free Upper and Pinned Lower Boundary Condition

6.6. Two-Riser Example 4

All input files except *risdyn-nF.in*, *risdyn-nR.in*, *vivo-nF.in* and *vivo-nR.in* are the same as the "**Single Riser Example 4**", since both front and rear risers are identical to the riser in "**Single Riser Example 1**".

The two risers are side-by side with a center to center distance of 2 ft. The corresponding files *risdyn-nF.in*, *risdyn-nR.in*, *vivo-nF.in* and *vivo-nR.in* are listed below.

- *risdyn-nF.in*

```
'RISOUT'
'DYNOUT'
 560475.9
 400
 30
 9
 0              0     1.543333
 38.1           0     1.440444
 76.2           0     1.183222
 114.3          0     0.8745555
152.4          0     0.6173333
228.6          0     0.463
304.8          0     0.2572222
457.2          0     0.2057778
457.2031       0     0
1025
0.01
0
466.7799       0
```

- *risdyn-nR.in*

```
'RISOUT'
'DYNOUT'
 560475.9
 400
 30
 9
0              0     1.543333
 38.1           0     1.440444
 76.2           0     1.183222
 114.3          0     0.8745555
152.4          0     0.6173333
228.6          0     0.463
304.8          0     0.2572222
457.2          0     0.2057778
457.2031       0     0
1025
0.01
0
466.7799       0.6096
```

- *vivo-nF.in*

```
 69           850.392
 1035.46        560475.9      -542.6597       850.392         466.7799       0
0             1.543333      0              0
 1020.05        552113.3      -531.1013       835.2618        463.8157       0
0             1.502474      0              0
 1004.639      544106.5      -519.4116       820.1528        460.825        0
0             1.461672      0              0
 989.2192      536099.7      -516.8723       805.0408        457.7849       0
0             1.39149       0              0
 973.8299      528181.8      -515.9528       789.9624        454.7064       0
0             1.289692      0              0
 958.4406      520219.5      -515.9528       774.8931        451.5829       0
0             1.187955      0              0
 943.0512      512301.7      -517.5436       759.842         448.4096       0
0             1.066966      0              0
 927.671        504294.9      -520.7396       744.8032        445.1909       0
0             0.945129      0              0
 912.3           496288.1      -516.5658       729.7827        441.9216       0
0             0.8319595     0              0
 896.9288      488414.8      -513.72         714.7834        438.5999       0
0             0.7306963     0              0
 881.5609      480496.9      -515.3253       699.7934        435.2267       0
0             0.6294947     0              0
 866.199        472579.1      -514.1724       684.8216        431.7971       0
0             0.5906583     0              0
```

850.8492	464705.7	-510.0277	669.8834	428.3129	0	0	0.5604028	0	0
835.4995	456921.3	-511.852	654.9512	424.7711	0	0	0.5301597	0	0
820.1711	449003.5	-519.2365	640.0526	421.172	0	0	0.4999844	0	0
804.8305	440996.7	-519.1927	625.1631	417.5077	0	0	0.4698277	0	0
789.4991	433078.9	-515.1939	610.301	413.7803	0	0	0.4319687	0	0
774.1798	425205.5	-515.5588	595.4634	409.9862	0	0	0.3918996	0	0
758.8697	417287.7	-509.8964	580.6532	406.1244	0	0	0.3519047	0	0
743.5596	409592.3	-508.4516	565.8613	402.1903	0	0	0.3119592	0	0
728.2495	401718.9	-518.7695	551.1028	398.1813	0	0	0.2721041	0	0
712.9485	393712.1	-513.3113	536.3627	394.0948	0	0	0.2541067	0	0
697.6597	386016.7	-507.7073	521.6622	389.9319	0	0	0.2491444	0	0
682.371	378187.8	-510.772	506.983	385.6845	0	0	0.2441893	0	0
667.0914	370403.4	-509.619	492.3435	381.3527	0	0	0.2392475	0	0
651.8209	362619	-508.4662	477.7314	376.9312	0	0	0.234315	0	0
636.5596	354879.1	-505.5474	463.1619	372.4208	0	0	0.2293969	0	0
621.2891	347183.7	-507.0651	448.623	367.8086	0	0	0.2244891	0	0
606.0308	339399.3	-510.2758	434.1206	363.0936	0	0	0.2195937	0	0
590.7786	331614.9	-504.5988	419.673	358.2766	0	0	0.2147168	0	0
575.5295	324008.5	-498.9655	405.2621	353.3473	0	0	0.2098522	0	0
560.2894	316402	-499.2574	390.8908	348.3026	0	0	0	0	0
545.0586	308795.6	-502.3805	376.5835	343.1408	0	0	0	0	0
529.8308	301100.1	-502.5264	362.3219	337.8528	0	0	0	0	0
514.6091	293493.7	-499.6077	348.1212	332.4374	0	0	0	0	0
499.3813	285887.2	-495.5652	333.9724	326.8763	0	0	0	0	0
484.1809	278414.2	-490.0195	319.9029	321.1797	0	0	0	0	0
468.9714	270985.7	-490.0195	305.8912	315.3275	0	0	0	0	0
453.771	263512.7	-493.2447	291.9619	309.3199	0	0	0	0	0
438.5798	255995.2	-487.5385	278.1208	303.1483	0	0	0	0	0
423.3885	248700.1	-480.2708	264.3622	296.7979	0	0	0	0	0
408.2003	241405	-477.5271	250.6919	290.264	0	0	0	0	0
393.0213	234198.9	-474.8856	237.1313	283.5363	0	0	0	0	0
377.8514	226992.8	-474.725	223.6927	276.6068	0	0	0	0	0

```
 362.6907       219800        -471.3539      210.373        269.4636      0
0              0              0              0
 347.5299       212700.6      -464.9034      197.1934       262.0881      0
0              0              0              0
 332.36          205699.1      -458.4821      184.151        254.4665      0
0              0              0              0
 317.2115       198799.9      -448.923       171.2823       246.5988      0
0              0              0              0
 302.069         192100.9     -445.6832      158.6027       238.4627      0
0              0              0              0
 286.9204       185299.6      -442.6331      146.1121       230.0356      0
0              0              0              0
 271.7902       178698.4      -429.5277      133.8529       221.3162      0
0              0              0              0
 256.6599       172301.9      -419.6768      121.8408       212.2803      0
0              0              0              0
 241.5296       165998.7      -413.1826      110.1029       202.9071      0
0              0              0              0
 226.4085       159802.4      -403.3463      98.67291       193.1834      0
0              0              0              0
 211.2904       153801.7      -387.0303      87.59338       183.0909      0
0              0              0              0
 196.1693       148099.1      -370.51        76.89191       172.6078      0
0              0              0              0
 181.0603       142601.1      -357.5214      66.63239       161.7265      0
0              0              0              0
 165.9606       137298.8      -341.1325      56.85132       150.4315      0
0              0              0              0
 150.8608       132299        -317.8552      47.61279       138.7034      0
0              0              0              0
 135.761        127699.5      -294.826       38.98083       126.5356      0
0              0              0              0
 120.6703       123398.1      -271.6217      31.02258       113.9328      0
0              0              0              0
 105.5797       119501.5      -245.1046      23.80188       100.8917      0
0              0              0              0
 90.48903       116000.7      -212.1078      17.41321       87.42795      0
0              0              0              0
 75.40143       113100.5      -179.038       11.90247       73.56665      0
0              0              0              0
 60.32906       110600.6      -149.281       7.373108       59.35617      0
0              0              0              0
 45.24146       108598.9      -105.9955      3.861816       44.8166       0
0              0              0              0
 30.15996       107402.3      -59.58691      1.441711       30.03055      0
0              0              0              0
 15.0815         106801.8      -26.54631      0.1432495      15.06772     0
0              0              0              0
 0                   106601.6     -13.26586      0                            0
0              0              0              0              0
```

- *vivo-nR.in*

```
69             850.392
 1035.46        560475.9     -542.6597      850.392       466.7799        0.6096
0              1.543333      0              0
 1020.05        552113.3     -531.1013      835.2618      463.8157        0.6096
0              1.502474      0              0
 1004.639       544106.5     -519.4116      820.1528      460.825         0.6096
0              1.461672      0              0
 989.2192       536099.7     -516.8723      805.0408      457.7849        0.6096
0              1.39149       0              0
 973.8299       528181.8     -515.9528      789.9624      454.7064        0.6096
0              1.289692      0              0
```

958.4406	520219.5	-515.9528	774.8931	451.5829	0.6096	0	1.187955	0	0
943.0512	512301.7	-517.5436	759.842	448.4096	0.6096	0	1.066966	0	0
927.671	504294.9	-520.7396	744.8032	445.1909	0.6096	0	0.945129	0	0
912.3	496288.1	-516.5658	729.7827	441.9216	0.6096	0	0.8319595	0	0
896.9288	488414.8	-513.72	714.7834	438.5999	0.6096	0	0.7306963	0	0
881.5609	480496.9	-515.3253	699.7934	435.2267	0.6096	0	0.6294947	0	0
866.199	472579.1	-514.1724	684.8216	431.7971	0.6096	0	0.5906583	0	0
850.8492	464705.7	-510.0277	669.8834	428.3129	0.6096	0	0.5604028	0	0
835.4995	456921.3	-511.852	654.9512	424.7711	0.6096	0	0.5301597	0	0
820.1711	449003.5	-519.2365	640.0526	421.172	0.6096	0	0.4999844	0	0
804.8305	440996.7	-519.1927	625.1631	417.5077	0.6096	0	0.4698277	0	0
789.4991	433078.9	-515.1939	610.301	413.7803	0.6096	0	0.4319687	0	0
774.1798	425205.5	-515.5588	595.4634	409.9862	0.6096	0	0.3918996	0	0
758.8697	417287.7	-509.8964	580.6532	406.1244	0.6096	0	0.3519047	0	0
743.5596	409592.3	-508.4516	565.8613	402.1903	0.6096	0	0.3119592	0	0
728.2495	401718.9	-518.7695	551.1028	398.1813	0.6096	0	0.2721041	0	0
712.9485	393712.1	-513.3113	536.3627	394.0948	0.6096	0	0.2541067	0	0
697.6597	386016.7	-507.7073	521.6622	389.9319	0.6096	0	0.2491444	0	0
682.371	378187.8	-510.772	506.983	385.6845	0.6096	0	0.2441893	0	0
667.0914	370403.4	-509.619	492.3435	381.3527	0.6096	0	0.2392475	0	0
651.8209	362619	-508.4662	477.7314	376.9312	0.6096	0	0.234315	0	0
636.5596	354879.1	-505.5474	463.1619	372.4208	0.6096	0	0.2293969	0	0
621.2891	347183.7	-507.0651	448.623	367.8086	0.6096	0	0.2244891	0	0
606.0308	339399.3	-510.2758	434.1206	363.0936	0.6096	0	0.2195937	0	0
590.7786	331614.9	-504.5988	419.673	358.2766	0.6096	0	0.2147168	0	0
575.5295	324008.5	-498.9655	405.2621	353.3473	0.6096	0	0.2098522	0	0
560.2894	316402	-499.2574	390.8908	348.3026	0.6096	0	0	0	0
545.0586	308795.6	-502.3805	376.5835	343.1408	0.6096	0	0	0	0
529.8308	301100.1	-502.5264	362.3219	337.8528	0.6096	0	0	0	0
514.6091	293493.7	-499.6077	348.1212	332.4374	0.6096	0	0	0	0
499.3813	285887.2	-495.5652	333.9724	326.8763	0.6096	0	0	0	0
484.1809	278414.2	-490.0195	319.9029	321.1797	0.6096	0	0	0	0

468.9714	270985.7	-490.0195	305.8912	315.3275	0.6096
0	0	0	0		
453.771	263512.7	-493.2447	291.9619	309.3199	0.6096
0	0	0	0		
438.5798	255995.2	-487.5385	278.1208	303.1483	0.6096
0	0	0	0		
423.3885	248700.1	-480.2708	264.3622	296.7979	0.6096
0	0	0	0		
408.2003	241405	-477.5271	250.6919	290.264	0.6096
0	0	0	0		
393.0213	234198.9	-474.8856	237.1313	283.5363	0.6096
0	0	0	0		
377.8514	226992.8	-474.725	223.6927	276.6068	0.6096
0	0	0	0		
362.6907	219800	-471.3539	210.373	269.4636	0.6096
0	0	0	0		
347.5299	212700.6	-464.9034	197.1934	262.0881	0.6096
0	0	0	0		
332.36	205699.1	-458.4821	184.151	254.4665	0.6096
0	0	0	0		
317.2115	198799.9	-448.923	171.2823	246.5988	0.6096
0	0	0	0		
302.069	192100.9	-445.6832	158.6027	238.4627	0.6096
0	0	0	0		
286.9204	185299.6	-442.6331	146.1121	230.0356	0.6096
0	0	0	0		
271.7902	178698.4	-429.5277	133.8529	221.3162	0.6096
0	0	0	0		
256.6599	172301.9	-419.6768	121.8408	212.2803	0.6096
0	0	0	0		
241.5296	165998.7	-413.1826	110.1029	202.9071	0.6096
0	0	0	0		
226.4085	159802.4	-403.3463	98.67291	193.1834	0.6096
0	0	0	0		
211.2904	153801.7	-387.0303	87.59338	183.0909	0.6096
0	0	0	0		
196.1693	148099.1	-370.51	76.89191	172.6078	0.6096
0	0	0	0		
181.0603	142601.1	-357.5214	66.63239	161.7265	0.6096
0	0	0	0		
165.9606	137298.8	-341.1325	56.85132	150.4315	0.6096
0	0	0	0		
150.8608	132299	-317.8552	47.61279	138.7034	0.6096
0	0	0	0		
135.761	127699.5	-294.826	38.98083	126.5356	0.6096
0	0	0	0		
120.6703	123398.1	-271.6217	31.02258	113.9328	0.6096
0	0	0	0		
105.5797	119501.5	-245.1046	23.80188	100.8917	0.6096
0	0	0	0		
90.48903	116000.7	-212.1078	17.41321	87.42795	0.6096
0	0	0	0		
75.40143	113100.5	-179.038	11.90247	73.56665	0.6096
0	0	0	0		
60.32906	110600.6	-149.281	7.373108	59.35617	0.6096
0	0	0	0		
45.24146	108598.9	-105.9955	3.861816	44.8166	0.6096
0	0	0	0		
30.15996	107402.3	-59.58691	1.441711	30.03055	0.6096
0	0	0	0		
15.0815	106801.8	-26.54631	0.1432495	15.06772	0.6096
0	0	0	0		
0	106601.6	-13.26586	0		0
0.6096	0	0	0	0	

6.7. Single Riser Inline VIV Example 1

This example is based on the single riser example in Section 6.1. The input files for VIVA are as follows:

- *rispre.in*

```
'RISOUT'
 9
 617.22
 983.1533
 245.697
 6579.024
 0.5524501
 0.5524501     0.028575
 3.235463E+08
 1
 1
 297.18
 1887.917
 1571.11
 2451.479
 1.397
 0.5524501     0.028575
 3.235463E+08
 1
 1
 297.18
 1887.917
 1571.11
 2451.479
 1.397
 0.5524501     0.028575
 3.235463E+08
 1
 1
 297.18
 1887.917
 1571.11
 2451.479
 1.397
 0.5524501     0.028575
 3.235463E+08
 1
 1
 320.04
 1887.917
 1571.11
 2451.479
```

```
1.397
0.5524501     0.028575
3.235463E+08
1
1
297.18
1868.492
1542.674 2540.016
1.3843
0.5524501     0.028575
3.235463E+08
1
1
868.6801
1830.216
1486.582
2714.753
1.3589
0.5524501     0.028575
3.235463E+08
1
1
22.86
983.1533
245.697
6579.024
0.5524501
0.5524501     0.028575
3.235463E+08
1
2
15.24
983.1434
245.697
6579.122
0.5524501
0.5524501     0.028575
3.235463E+08
1
1
1
```

- *risdyn-n.in*

```
'RISOUT'
'DYNOUT'
 1.067573E+07
 600
 30
 6
 0             1.800555      0
 91.44         1.749111      0
```

```
152.4         1.028889      0
304.8         0.5144444     0
457.2         0.2572222     0
3352.8        0.1028889     0
1025
0.01
0
```

- *conditions.in*

```
0
1.114E-06
1             1
0             0
1
```

- *risfat.in*

```
1
1.3729E+09    4.38
1.5
```

- *no_files.in*

```
2                             ! number of hydrodynamic files
basic_bare-inline  41  0      ! BASIC HYDRO DATA for smooth cylinder, in-line
out_s.db  15  4               ! file with strakes P/D=17, h/D=-0.25
```