Structural Analysis Software

Reference Manual

FEM48

Version 4.4

COPYRIGHT AND DISCLAIMER OF WARRANTY

CONTENTS

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Scope

The FEM48 library is a so-called Finite Elements Method application for the Hewlett Packard G series calculators[1]. It is designed to be used for the structural analysis of 2D frames, trusses and beams.
FEM48 is THE PREMIER STRUCTURAL ANALYSIS SOFTWARE for the Hewlett Packard G series calculators. It's ease of use, flexibility, power and capabilities combined in one package are not seen in any other program available[2], either commercially or as freeware.

Main features

- extremely easy to use due to excellent interface
- supports frames, trusses and beams
- beam analysis (numerical and/or graphical)
- data can be entered in three (!) ways:
  - using a structure wizard for "standard structures"
  - using input prompts for structure data
  - assemble data arrays and store them from the stack
- text file (string) output of input and calculation results
- (un)deformed structure plots with optional displaying of supports, node numbers, members numbers
- moving loads generator
- file management
- configurable matrix viewer/editor (choose your own matrix editor i.e. MATRIX or EDITB with Metakernel)
- configurable stringviewer (i.e. VV, the author's VIEW library or EDITB with Metakernel)
- superfast Cholesky matrix solve routine (assembly language)
- links to an external section database (SED48 v1.2 or higher)
- completely programmable (write your own batch files)
- etc etc

1.2 Method

FEM48 uses the displacement method for primary analysis. All other calculated values are derived from the nodal displacements.

1.3 Description

FEM48 follows a modular approach. The available modules integrate seamlessly into the FEM48 interface. Below a short description of each module is given. See the sections describing all behaviour of the modules (4, 5, 6, 7, 8) for more details

1.3.1 FEM48 module

This is the main module, which should always be installed. It provides the means of entering and editing data, calculation of the structure, plot routines and file management. The calculation only provides Finite Element Analysis, thus only data at the nodes will be calculated. For beam analysis you will need the QUERY module (see below).

1.3.2 QUERY module

This module is optional and provides beam analysis facilities. The member to be analysed can be selected after which numerical and graphical output like axial force, shear force, bending moment, rotation and axial and transverse displacements can be obtained.

[1] Please note that the FEM48 library is not designed for the HP48S(X) calculators. It has been ported to the HP49 but this manual only describes FEM48. HP49 specific operation is not covered in this manual.
[2] For calculators that is.

1.3.3 WIZRD module

This module is optional and provides wizards for entering geometry and properties of "standard" structures like beams, bay frames and lattices. A link to SED48, a Structural Engineering Database by the author of FEM48 is provided.

1.3.4 PRINT module

This module is optional and provides commands for easy viewing and printing[3] of the input and output of FEM48.

1.3.5 MOVLD module

This module is optional and provides a command for generating moving loads along an already defined (multi-spanned) beam.

1.3.6 Module version

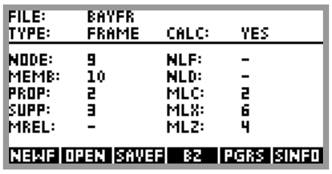
Each module is available in two versions, a "standard" version and a "compressed" version[4]. The compressed versions are unique in that they provide exactly the same features as the standard versions at a smaller size! This because subroutines of these modules have been compressed. They decompress during runtime (which slows operations down a little bit). The standard and compressed modules can be mixed and matched. This allows you to select standard versions for much used modules and compressed versions for not so much used modules.

1.4 Screenshots

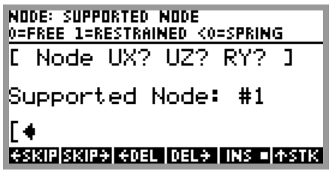
Below some screenshots of FEM48 in action are given to get a quick impression of the program.

File management

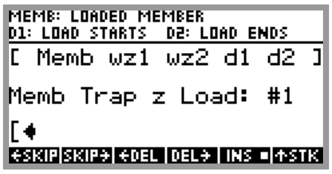
Structure information



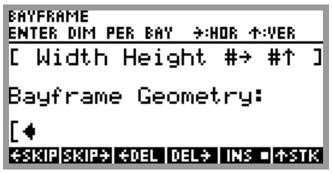
Supports entry prompt



Trapezoid loads entry prompt

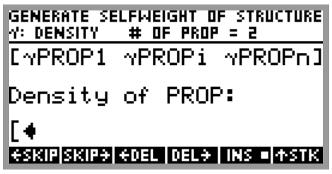


Bayframe geometry wizard

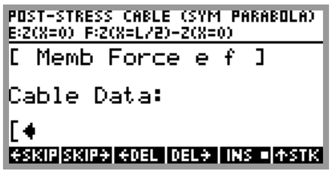


Section properties wizard

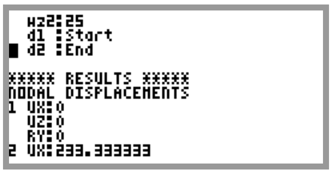
Selfweight generator wizard



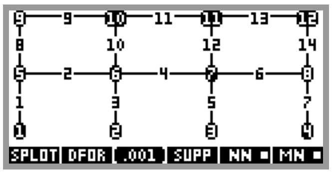
Post-stressing cable wizard



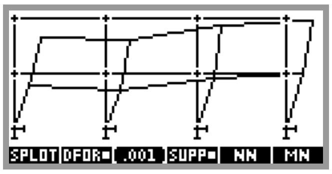
String output of calculation



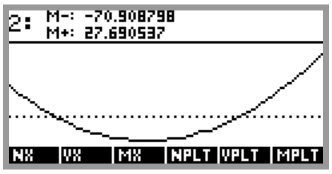
Structure plot



Deformed structure plot



Bending moment plot



[3] If you own an HP printer. Note that it is also easy when using FEM48 on an emulator, this enables you to copy/paste the strings to a text editor on your PC.

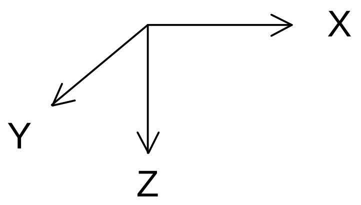
[4] FEM49 is not modular. All modules have been assembled into one library.

2. COORDINATES AND UNITS

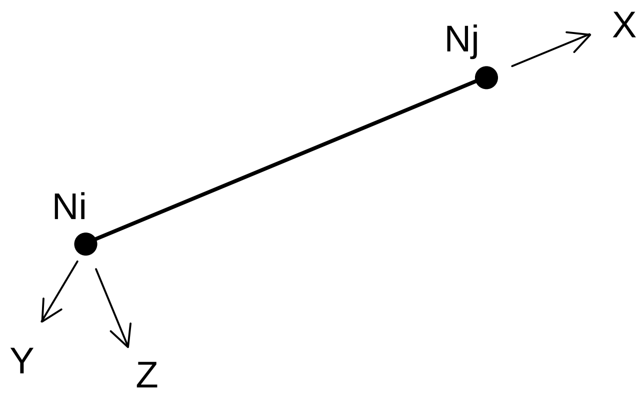
2.1 Coordinate systems

FEM48 uses a right hand rule X-Y-Z coordinate system where the positive X-direction is to the right and the positive Z-direction is downwards. Note that there are actually two coordinate systems. The global coordinate system and the local coordinate system.
The global coordinate system is used to enter the nodal coordinates. It is also used to enter nodal loads and to calculate nodal displacements.
The local coordinate system of a member is defined by the location of the start (Ni) and end (Nj) node of the member. It is used to enter member loads and to calculate member end forces.
Please note that positive loads, reactions and displacements also follow these coordinate systems.

Global coordinate system:



Local member coordinate system:



2.2 Units

Since the HP48 does not support units within arrays you have to be careful about the implied units you use. They must be consistent for meaningful results. So, if you use meter as the unit for length and kilo Newton as the unit for force you have to use square meter as the unit for area and kilo Newton meter as the unit for moment and so on.

3. QUICK START GUIDE

3.1 Data entry

Generally speaking, there are three ways to enter data with the FEM48 library:

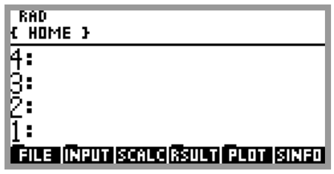
1. Use the inputline data prompts.
2. Assemble your own data matrices and store them with the →… commands.
3. Use the geometry, property and load wizards of the WIZRD module.

Below only method 1 is handled, have a look at the Command reference of the modules for more information about the other two data entry modes.

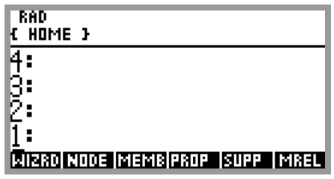
3.2 Example of structure entry

Note: A general understanding of the HP48 calculator is assumed. Only the FEM48 module is discussed here. Please refer to the Command reference of the modules (4.3, 5.4, 6.3, 7.3) for more details.

- Execute the FEM command (4.3.1)
  Now you have access to all commands with an easy to use interface.

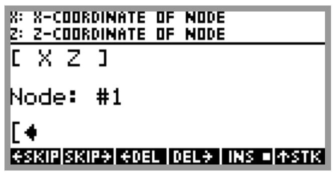


- Go to the INPUT menu
  Here all structure and load data can be entered and/or edited. Pressing a data menukey unshifted starts an inputline prompt for data, pressing it left-shifted stores data from the stack and pressing it right-shifted recalls the data to the stack (depending on the state of the AV? (4.3.54)*).*

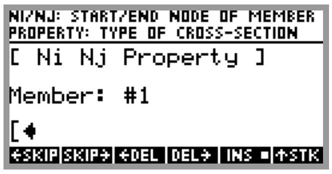


About the inputline prompts described next: The [ENTER] key confirms data line and proceeds to next data line, the [ON] key signals the end of input and 1) saves (overwrites) the data type in the current structure or 2) cancels input <u>if no data line has been [ENTER]ed yet</u>. See the Command reference of each module (4.3, 5.4, 6.3, 7.3) for related commands with which you can edit data or enter data directly from the stack.

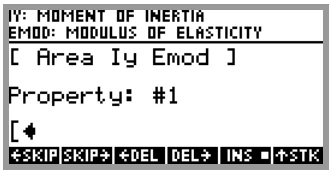
- Enter nodes with the NODE command (4.3.13).
  Note the coordinate system: X → and Z ↓.



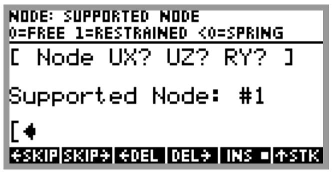
- Enter members with the MEMB command (4.3.19).
  Each member has a reference to a PROP, see below.



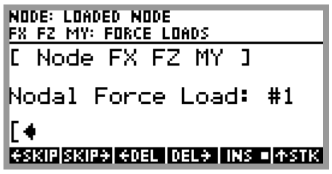
- Enter member properties with the PROP command (4.3.16).



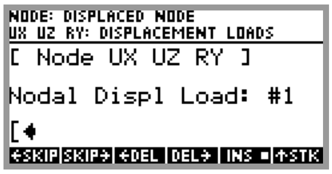
- Enter supports with the SUPP command (4.3.22).
  Springs can be entered too, just enter the spring value as a negative number.



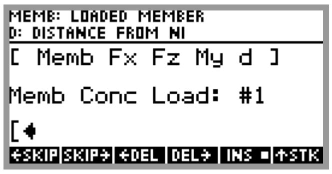
- Enter either one or more loads with the following commands:
  - NLF (4.3.28)
    Nodal force load (global system).



  - NLD (4.3.31)
    Nodal displacement load (global system).

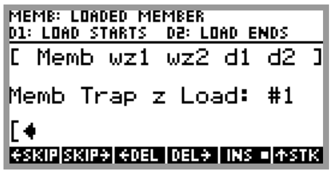


  - MLC (4.3.34)
    Concentrated member load (local system).

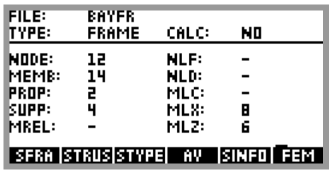


  - MLX (4.3.37)
    Trapezoid axial member load (local system). For uniform loads wx1 = wx2.

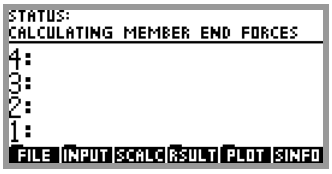
- MLZ (4.3.40)
  Trapezoid transverse member load (local system). For uniform loads wz1 = wz2.



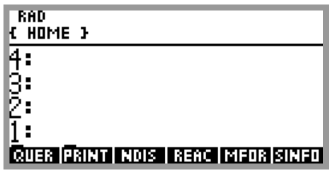
- Set the desired structure type with the SFRAM (4.3.44) or STRUS (4.3.45) command (3rd menurow).
  The default state of a new structure is FRAME.
- Check what you have entered with the SINFO (4.3.7) command



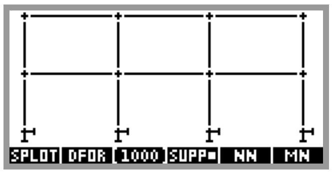
- Go back to the FEM menu and perform the calculation with the SCALC (4.3.47) command.



- Go to the RSULT menu.



- Look at the calculation results with one or more of the following commands:
  - NDIS (4.3.49)
    Nodal displacements (global system)
  - REAC (4.3.50)
    Support reactions (global system)
  - MFOR (4.3.51)
    Member end forces (local system)
- Go back to the FEM menu, go to the PLOT menu and use the SPLOT (4.3.48) command to plot the structure.
  Also possible before calculation as a geometry check.



4. FEM48 MODULE

4.1 Description

The FEM48 module is the main module of the FEM48 library. This module has to be installed[5] in order to use other modules, since the other modules also use subroutines from the FEM48 module.
The FEM48 module provides the basic tools for structure analysis like file management, entering and editing of data, calculation of the structure and structure plot routines.

4.2 Menu structure

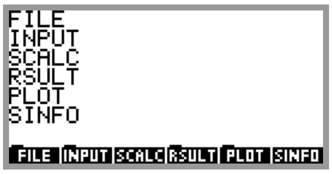
4.2.1 Library menu

After displaying the library menu (RS 2) and pressing the FEM48 library menukey you will see the FEM48 library menu. All commands of the FEM48 module are programmable and can be accessed from the library menu. However, this is not the best possible interface for easy use of FEM48. Therefore the FEM menu has been made, accessible through the FEM command. See 4.2.2 for details.

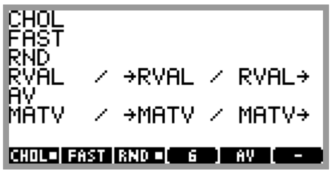
4.2.2 FEM menu

The FEM menu, accessible through the FEM command provides a very easy to use interface for the FEM48 library. All commands of the FEM48 module[6] and the QUERY, WIZRD and PRINT modules are accessible through this menu. Below the pages of the FEM menu and it's submenus will be shown as screenshots.
Note that you can see all commands that are available in the order: unshifted, left-shifted and right-shifted. Toggle menu keys can also be used in the left and right shifted plane, the toggle command and its argument are then echoed in the command line for easy programmable use.

FEM menu row 1



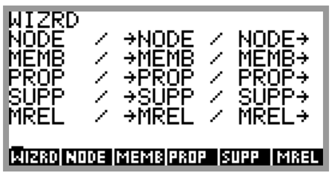
FEM menu row 2



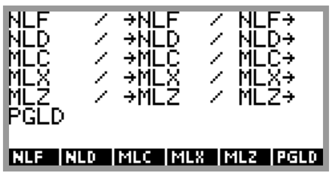
FILE menu row 1

FILE menu row 2

INPUT menu row 1



INPUT menu row 2

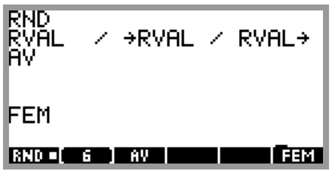


INPUT menu row 3

5 The commands of the other modules are protected and can not be used without the FEM48 module present or when this has a different version number.
6 With the exception of the CKFEM and ERASEFEM commands.

RSULT menu row 1

RSULT menu row 2



PLOT menu row 1

PLOT menu row 2

4.3 Command reference

4.3.1 FEM

Displays the FEM menu.

Level 1	→	Level 1
	→	

Remarks:

Suggested user key binding: 73.3 (RS 5)

4.3.2 CKFEM

Displays installed FEM48 modules and their version number.

Level 1	→	Level 1
	→	

Remarks:

All modules must be of the same version number as the main FEM48 module or they will be disabled (including programmable use). Only accessible from the FEM48 library menu.

4.3.3 ERASEFEM

Erases all hidden FEM48 files (FEM_cfg, FEM_in, FEM_ck, FEM_out) from the hidden directory[7] after confirmation.

Level 1	→	Level 1
	→	

Remarks:

You will lose all structure data that has not been saved as a file. Only accessible from the FEM48 library menu.

4.3.4 NEWFEM

Starts a new "empty" structure.

Level 1	→	Level 1
	→	

Remarks:

If the current structure is not "empty" you will be prompted to save the current structure before it is replaced by an "empty" one.

4.3.5 OPENFEM

Opens a previously saved structure. You can select the file to open with a browser interface alike the HP48 choose box engine. The OPENFEM browser is much faster though.

Level 1	→	Level 1
	→	

Remarks:

If the current structure is not "empty" you will be prompted to save the current structure before it is replaced by the opened one. OPENFEM looks for files with the FEM extension (i.e. Truss.FEM) in the current directory.

4.3.6 SAVEFEM

Saves the current structure as a file.

[7] A nullnamed subdirectory of HOME.

Level 1	→	Level 1
	→	

Remarks:

The FEM extension is automatically added by the program. The structure is saved in the current directory as a Library Data object. Illegal names (i.e. Truss Big) are allowed. Memory can be saved if you have the BZ compressor present on your HP48. Saved files will be BZ compressed first if the BZ option (with BZ? (4.3.70) command) is toggled on. An additional saved file size reduction can be accomplishes by purging the calculation results with the PGRS (4.3.12) command before saving the file. Note that then you will have to calculate the structure again after re-opening the file.

4.3.7 SINFO

Displays information about the current structure.

Level 1	→	Level 1
	→	

Remarks:

Also displays erroneous or undefined (but necessary) input.

4.3.8 →FEM

Stores a FEM file as the current structure.

Level 1	→	Level 1	
Library Data	→		overwrite current structure
0	→		erase current structure

Remarks:

Intended for programmable use. Overwrites or erases current structure. Checks for validity of the Library Data.

4.3.9 FEM→

Recalls the current structure as a Library Data object.

Level 1	→	Level 1
	→	Library Data

Remarks:

Intended for programmable use.

4.3.10 →SNAM

Stores a name string as the name of the current structure.

Level 1	→	Level 1
"string"	→	

Remarks:

Intended for programmable use.

4.3.11 SNAM→

Recall the name string of the current structure.

Level 1	→	Level 1
	→	"string"

Remarks:

Intended for programmable use.

4.3.12 PGRS

Purges the calculation results of the current structure.

Level 1	→	Level 1
	→	

Remarks:

Can be used to save memory when files are saved, see SAVEFEM (4.3.6) command. This command works without confirmation. However, the command asks for confirmation when executed from the FEM (4.3.1) menu. The programmable version of the command is left-shifted available in the FEM menu.

4.3.13 NODE

Starts inputline prompt for nodal X and Z coordinates.

Level 1	→	Level 1
	→	

Remarks:

The [ENTER] key confirms data line and proceeds to next data line, the [ON] key signals the end of input and 1) saves (overwrites) the data type in the current structure or 2) cancels input if no data line has been [ENTER]ed yet.

4.3.14 →NODE

Stores the nodal X and Z coordinates matrix.

Level 1	→	Level 1	
[[matrix]] $_{n \times 2}$	→		overwrite
0	→		erase

Remarks:

Column 1 contains the X coordinates and column 2 contains the Z coordinates, see also 2.1. Row 1 equals node 1, row 2 equals node 2 etc.

4.3.15 NODE→

Recalls the nodal X and Z coordinates matrix.

Level 1	→	Level 1	
	→	[[matrix]] $_{n \times 2}$	defined and AV? (4.3.54) off
	→	unknown	defined and AV? on
	→	0	undefined

Remarks:

See 4.3.14.

4.3.16 PROP

Starts inputline prompt for member properties.

Level 1	→	Level 1
	→	

Remarks:

The [ENTER] key confirms data line and proceeds to next data line, the [ON] key signals the end of input and 1) saves (overwrites) the data type in the current structure or 2) cancels input if no data line has been [ENTER]ed yet.

4.3.17 →PROP

Stores the member properties matrix.

Level 1	→	Level 1	
[[matrix]] $_{n \times 3}$	→		overwrite
0	→		erase

Remarks:

Column 1 contains the cross-sectional Area, column 2 contains the moment of inertia Iy and column 3 contains the modulus of elasticity E. Row 1 equals property 1, row 2 equals property 2 etc.

4.3.18 PROP→

Recalls member properties matrix.

Level 1	→	Level 1	
	→	[[matrix]] $_{n \times 3}$	defined and AV? (4.3.54) off
	→	unknown	defined and AV? on
	→	0	undefined

Remarks:

See 4.3.17.

4.3.19 MEMB

Starts inputline prompt for member incidences and the member property reference number.

Level 1	→	Level 1
	→	

Remarks:

The [ENTER] key confirms data line and proceeds to next data line, the [ON] key signals the end of input and 1) saves (overwrites) the data type in the current structure or 2) cancels input <u>if no data line has been [ENTER]ed yet</u>.

4.3.20 →MEMB

Stores the member matrix.

Level 1	→	Level 1	
[[matrix]] $_{n \times 3}$	→		overwrite
0	→		erase

Remarks:

Column 1 contains the start node Ni, column 2 contains the end node Nj and column 3 contains the property reference. Row 1 equals member 1, row 2 equals member 2 etc.

4.3.21 MEMB→

Recalls the member matrix.

Level 1	→	Level 1	
	→	[[matrix]] $_{n \times 3}$	defined and AV? (4.3.54) off
	→	unknown	defined and AV? on
	→	0	undefined

Remarks:

See 4.3.20.

4.3.22 SUPP

Starts inputline prompt for the supported nodes and their restraints.

Level 1	→	Level 1
	→	

Remarks:

The [ENTER] key confirms data line and proceeds to next data line, the [ON] key signals the end of input and 1) saves (overwrites) the data type in the current structure or 2) cancels input <u>if no data line has been [ENTER]ed yet</u>.
A value of 0 for a degree of freedom means that movement is free, a value of 1 means that movement is restrained.
Spring supports can be assigned by entering a negative number. The absolute value of the number is the stiffness of the spring. Rotational springs are entered in "moment" per radian.

4.3.23 →SUPP

Stores the supports matrix.

Level 1	→	Level 1	
[[matrix]] $_{n \times 4}$	→		overwrite
0	→		erase

Remarks:

Column 1 contains the restrained node Ni, column 2 contains the UX restraint, column 3 contains the UZ restraint and column 4 contains the RY restraint. Row 1 equals support 1, row 2 equals support 2 etc. See also 4.3.22.

4.3.24 SUPP→

Recalls the support matrix.

Level 1	→	Level 1	
	→	[[matrix]] $_{n \times 4}$	defined and AV? (4.3.54) off
	→	unknown	defined and AV? on
	→	0	undefined

Remarks:

See 4.3.22 and 4.3.23.

4.3.25 MREL

Starts inputline prompt for member end releases data.

Level 1	→	Level 1
	→	

Remarks:

The [ENTER] key confirms data line and proceeds to next data line, the [ON] key signals the end of input and 1) saves (overwrites) the data type in the current structure or 2) cancels input <u>if no data line has been [ENTER]ed yet</u>.
As opposed to early versions of FEM48, members with end releases can now be loaded with any type of load. However, please note that the QUERY module is not able to calculate transverse displacements and rotation of members that have a release at the start node Ni. Therefore, if a member needs only one release, it is preferred to orientate the member (choice of Ni and Nj) so that the release is at the end node Nj.
Be aware that a node must be attached to at least one member for rotation, otherwise the structure cannot be solved (the node could then rotate unhindered, this could be solved by giving it a restraint against rotation[8] but it is preferred to leave it attached to a member).

4.3.26 →MREL

[8] Inside information: This is actually the way the TRUSS option works. Truss structures use a different local stiffness matrix (only axial stiffness) so rotation of nodes is initially not restrained. This is internally solved by giving each node a restraint against rotation. When viewing the nodal displacements of a TRUSS structure you will notice that the nodes have not rotated.

Stores the member end releases matrix.

Level 1	→	Level 1	
[[matrix]] $_{n \times 3}$	→		overwrite
0	→		erase

Remarks:

Column 1 contains the member with end release(s), column 2 contains the state of the start release at Ni (0=no release, equals fixed to node Ni and 1=release, equals pinned to node Ni) and column 3 contains the state of the end release at Nj (0=no release, equals fixed to node Nj and 1=release, equals pinned to node Nj). If the member has releases at both the beginning and the end it is turned into a truss element. Row 1 equals member with releases 1, row 2 equals member with releases 2 etc. See also 4.3.25.

4.3.27 MREL→

Recalls the member end releases matrix.

Level 1	→	Level 1	
	→	[[matrix]] $_{n \times 3}$	defined and AV? (4.3.54) off
	→	unknown	defined and AV? on
	→	0	undefined

Remarks:

See 4.3.25 and 4.3.26.

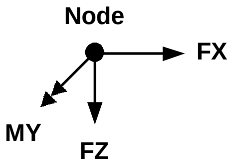
4.3.28 NLF

Starts inputline prompt for nodal force loads.

Level 1	→	Level 1
	→	

Remarks:

The [ENTER] key confirms data line and proceeds to next data line, the [ON] key signals the end of input and 1) saves (overwrites) the data type in the current structure or 2) cancels input <u>if no data line has been [ENTER]ed yet</u>.
Nodal force loads are entered in the global coordinates system. Multiple occurrences of a node are allowed.
Below the positive directions are drawn:



4.3.29 →NLF

Stores the nodal force loads matrix.

Level 1	→	Level 1	
[[matrix]] $_{n \times 4}$	→		overwrite
0	→		erase

Remarks:

Column 1 contains the loaded node, column 2 contains the FX load, column 3 contains the FZ load and column 4 contains the MY load. Row 1 equals nodal force load 1, row 2 equals nodal force load 2 etc. See also 4.3.28.

4.3.30 NLF→

Recalls the nodal force loads matrix.

Level 1	→	Level 1	
	→	[[matrix]] $_{n \times 4}$	defined and AV? (4.3.54) off
	→	unknown	defined and AV? on
	→	0	undefined

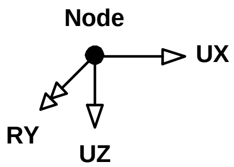
Remarks:
See 4.3.28 and 4.3.29.

4.3.31 NLD

Starts inputline prompt for nodal displacement loads, where rotation is entered in radians.

Level 1	→	Level 1
	→	

Remarks:
The [ENTER] key confirms data line and proceeds to next data line, the [ON] key signals the end of input and 1) saves (overwrites) the data type in the current structure or 2) cancels input if no data line has been [ENTER]ed yet.
Nodal displacement loads are entered in the global coordinates system. Multiple occurrences of a node are allowed, but only if the loaded degree of freedom is different for each occurrence of that node. NLD overrules defined supports. So, to "sink" a support, just load it with a NLD. Unlike with FEM48 version 4.2, there is no need to "unrestrain" the degree of freedom that is loaded with the NLD.
Below the positive directions are drawn:



4.3.32 →NLD

Stores the nodal displacement loads matrix.

Level 1	→	Level 1	
[[matrix]] $_{n \times 4}$	→		overwrite
0	→		erase

Remarks:
Column 1 contains the loaded node, column 2 contains the UX load, column 3 contains the UZ load and column 4 contains the RY load (in radians). Row 1 equals nodal displacement load 1, row 2 equals nodal displacement load 2 etc. See also 4.3.31.

4.3.33 NLD→

Recalls the nodal displacement loads matrix.

Level 1	→	Level 1	
	→	[[matrix]] $_{n \times 4}$	defined and AV? (4.3.54) off
	→	unknown	defined and AV? on
	→	0	undefined

Remarks:
See 4.3.31 and 4.3.32.

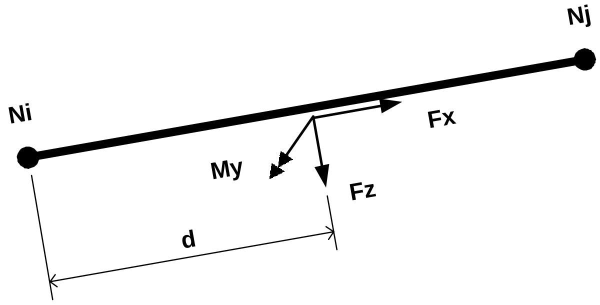
4.3.34 MLC

Starts inputline prompt for concentrated member loads (local axes).

Level 1	→	Level 1
	→	

Remarks:

The [ENTER] key confirms data line and proceeds to next data line, the [ON] key signals the end of input and 1) saves (overwrites) the data type in the current structure or 2) cancels input if no data line has been [ENTER]ed yet.
Member loads are entered in the local coordinates system. Multiple occurrences of a member are allowed.
Below the positive directions are drawn:



4.3.35 →MLC

Stores the concentrated member loads matrix (local axes).

Level 1	→	Level 1	
[[matrix]] $_{n\,x\,5}$	→		overwrite
0	→		erase

Remarks:

Column 1 contains the loaded member, column 2 contains the Fx load, column 3 contains the Fz load, column 4 contains the My load and column 5 contains the distance of the load(s) from the start node Ni. Row 1 equals concentrated member load 1, row 2 equals concentrated member load 2 etc. See also 4.3.34.

4.3.36 MLC→

Recalls the concentrated member loads matrix (local axes).

Level 1	→	Level 1	
	→	[[matrix]] $_{n\,x\,5}$	defined and AV? (4.3.54) off
	→	unknown	defined and AV? on
	→	0	undefined

Remarks:

See 4.3.34 and 4.3.35.

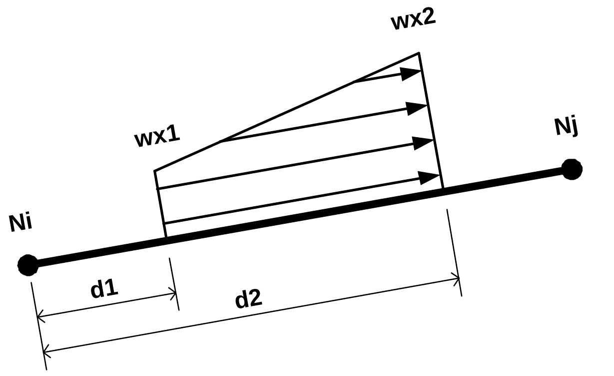
4.3.37 MLX

Starts inputline prompt for trapezoidal distributed axial (local x direction) member loads.

Level 1	→	Level 1
	→	

Remarks:

The [ENTER] key confirms data line and proceeds to next data line, the [ON] key signals the end of input and 1) saves (overwrites) the data type in the current structure or 2) cancels input if no data line has been [ENTER]ed yet.
Member loads are entered in the local coordinates system. Multiple occurrences of a member are allowed.
Below the positive directions are drawn:



4.3.38 →MLX

Stores the trapezoidal distributed axial (local x direction) member loads matrix.

Level 1	→	Level 1	
[[matrix]] $_{n \times 5}$	→		overwrite
0	→		erase

Remarks:

Column 1 contains the loaded member, column 2 contains the start load value wx1, column 3 contains the end load value wx2, column 4 contains the distance of the start of the load d1 from the start node Ni and column 5 contains distance of the end of the load d2 from the start node Ni. Note that when the load ends at the end node Nj you can also enter the value 0 for d2. This to avoid calculating the length of inclined members. Row 1 equals trapezoidal distributed axial member load 1, row 2 equals trapezoidal distributed axial member load 2 etc. See also 4.3.37.

4.3.39 MLX→

Recalls the trapezoidal distributed axial (local x direction) member loads matrix.

Level 1	→	Level 1	
	→	[[matrix]] $_{n \times 5}$	defined and AV? (4.3.54) off
	→	unknown	defined and AV? on
	→	0	undefined

Remarks:

See 4.3.37 and 4.3.38.

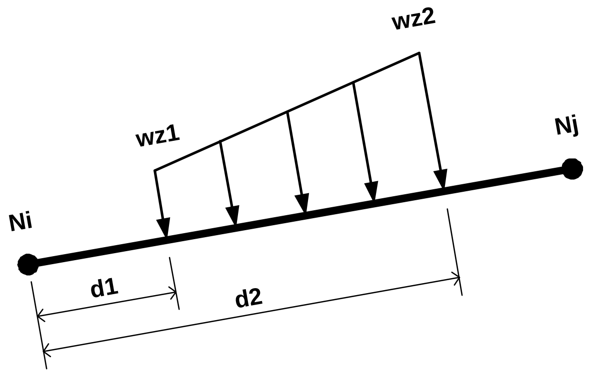
4.3.40 MLZ

Starts inputline prompt for trapezoidal distributed transverse (local z direction) member loads.

Level 1	→	Level 1
	→	

Remarks:

The [ENTER] key confirms data line and proceeds to next data line, the [ON] key signals the end of input and 1) saves (overwrites) the data type in the current structure or 2) cancels input <u>if no data line has been [ENTER]ed yet</u>.
Member loads are entered in the local coordinates system. Multiple occurrences of a member are allowed.
Below the positive directions are drawn:



4.3.41 →MLZ

Stores the trapezoidal distributed transverse (local z direction) member loads matrix.

Level 1	→	Level 1	
[[matrix]] $_{n \times 5}$	→		overwrite
0	→		erase

Remarks:

Column 1 contains the loaded member, column 2 contains the start load value wz1, column 3 contains the end load value wz2, column 4 contains the distance of the start of the load d1 from the start node Ni and column 5 contains distance of the end of the load d2 from the start node Ni. Note that when the load ends at the end node Nj you can also enter the value 0 for d2. This to avoid calculating the length of inclined members. Row 1 equals trapezoidal distributed transverse member load 1, row 2 equals trapezoidal distributed transverse member load 2 etc. See also 4.3.40.

4.3.42 MLZ→

Recalls the trapezoidal distributed transverse (local z direction) member loads matrix.

Level 1	→	Level 1	
	→	[[matrix]] $_{n \times 5}$	defined and AV? (4.3.54) off
	→	unknown	defined and AV? on
	→	0	undefined

Remarks:

See 4.3.40 and 4.3.41.

4.3.43 PGLD

Purges all loads on the structure.

Level 1	→	Level 1
	→	

Remarks:

This command works <u>without</u> confirmation. However, the command asks for confirmation when executed from the FEM (4.3.1) menu. The programmable version of the command is left-shifted available in the FEM menu.

4.3.44 SFRAM

Sets the structure type to FRAME.

Level 1	→	Level 1
	→	

Remarks:

This is also the default state of a new structure.

4.3.45 STRUS

Sets the structure type to TRUSS.

Level 1	→	Level 1
	→	

Remarks:

The complete structure is turned into a truss. Have a look at the MREL command (4.3.25) on how to turn just a few members into truss elements.

4.3.46 STYPE

Recalls the current structure type

Level 1	→	Level 1	
	→	"FRAME"	structure is a frame
	→	"TRUSS"	structure is a truss

Remarks:

Intended for programmable use.

4.3.47 SCALC

Calculates the current structure.

Level 1	→	Level 1
	→	

Remarks:

Will execute +/- 10% faster if the FAST? (4.3.53) option is enabled. When CHOL? (4.3.52) option is enabled SCALC uses a Cholesky solver instead of the internal HP solver. The Cholesky solver is optimised for speed and memory usage. Exits with SINFO (4.3.7) if input contains errors or when not enough input is present.

4.3.48 SPLOT

Plots the current structure.

Level 1	→	Level 1	
	→		→STK? (4.3.69) off
	→	grob	→STK? on

Remarks:

The resulting plot depends upon the state of the plot options: DFOR? (4.3.62), MAGN (4.3.63), SUPP? (4.3.66), NN? (4.3.67) and MN? (4.3.68).

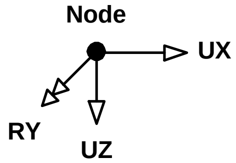
4.3.49 NDIS

Recalls the nodal displacements matrix.

Level 1	→	Level 1	
	→	[[matrix]] $_{n \times 4}$	AV? (4.3.54) off
	→	unknown	AV? on

Remarks:

Column 1 contains the node number, column 2 contains the displacement in global x direction, column 3 contains the displacement in global z direction and column 4 contains the rotation around the global y axis (in radians).
Below the positive directions are drawn:



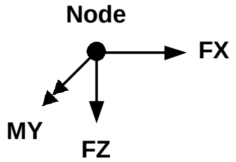
4.3.50 REAC

Recalls the support reactions matrix.

Level 1	→	Level 1	
	→	[[matrix]] n x 4	AV? (4.3.54) off
	→	unknown	AV? on

Remarks:

Column 1 contains the node number, column 2 contains the reaction force FX in global x direction, column 3 contains the reaction force FZ in global z direction and column 4 contains the reaction moment MY around the global y axis.
Below the positive directions are drawn:



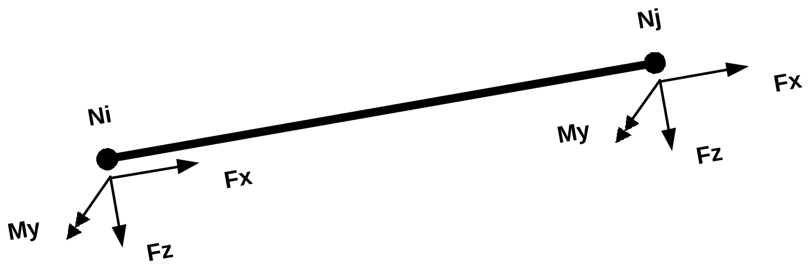
4.3.51 MFOR

Recalls the member end forces matrix.

Level 1	→	Level 1	
	→	[[matrix]] n x 5	AV? (4.3.54) off
	→	unknown	AV? on

Remarks:

Column 1 contains the member number, column 2 contains the node number where the end forces are situated, column 3 contains the member end force FX in local x direction, column 4 contains the member end force FX in local z direction and column 5 contains the member end moment MY around the local y axis.
Below the positive directions are drawn:



4.3.52 CHOL?

Toggles Cholesky solver.

Level 1	→	Level 1	
1	→		enabled
0	→		disabled (uses HP solver)

Remarks:
Usage of the Cholesky solver is recommended because it is much faster and more memory efficient than the internal HP solve routine. The Cholesky solver is written in assembly language.

4.3.53 FAST?

Toggles fast mode.

Level 1	→	Level 1	
1	→		enabled
0	→		disabled

Remarks:
Enabling fast mode turns of the display during calculations which reduces calculation time with +/- 10%.

4.3.54 AV?

Toggles auto-viewing.

Level 1	→	Level 1	
1	→		enabled
0	→		disabled

Remarks:
Matrices are viewed with the viewer defined with the MATV command (4.3.55), strings (PRINT module) with the viewer defined with the STRV command (7.3.9).

4.3.55 MATV

Starts inputline prompt for an external matrixwriter/viewer.

Level 1	→	Level 1
	→	

Remarks:
External matrixwriter/viewer, can also be a text editor.
Command should be a library command, a variable name is not allowed.
When no external matrixwriter/viewer is defined the internal HP48 matrixwriter is used.
Also see 4.3.54.

4.3.56 →MATV

Stores the external matrixwriter/viewer command as a string.

Level 1	→	Level 1
"string"	→	

Remarks:
See 4.3.55.

4.3.57 MATV→

Recalls the external matrixwriter/viewer command as a string.

Level 1	→	Level 1
	→	"string"

Remarks:
See 4.3.55.

4.3.58 RND?

Toggles auto-rounding of calculation results.

Level 1	→	Level 1	
1	→		enabled
0	→		disabled

Remarks:
The results are saved in full precision. The rounding takes place when they are recalled. Results are rounded with the value stored with the RVAL command, see 4.3.59.

4.3.59 RVAL

Starts inputline prompt for the rounding value.

Level 1	→	Level 1
	→	

Remarks:
The rounding follows the conventions of the HP RND command:

$0 \le n \le 11$ round to n decimal places
$-11 \le n \le -1$ round to n significant digits
$n = 12$ round to current display format

4.3.60 →RVAL

Stores the rounding value.

Level 1	→	Level 1
integer	→	

Remarks:
See 4.3.59.

4.3.61 RVAL→

Recalls the rounding value.

Level 1	→	Level 1
	→	integer

Remarks:
See 4.3.59.

4.3.62 DFOR?

Toggles deformation plot.

Level 1	→	Level 1	
1	→		enabled
0	→		disabled

Remarks:
When toggled on the deformed structure is plotted on top of the undeformed structure[9]. The deformation scale can be set with the MAGN command (4.3.63). Only nodal displacements are taken into account. Member deformations are not plotted, use the QUERY module (5) for this.

[9] The structure will be auto-calculated if necessary.

4.3.63 MAGN

Starts inputline prompt for the deformation magnification.

Level 1	→	Level 1
	→	

Remarks:

A value of 1000 yields the plotting of 1 millimetre as 1 meter. The deformation magnification must be greater than zero.

4.3.64 →MAGN

Stores the deformation magnification.

Level 1	→	Level 1
real number	→	

Remarks:

See 4.3.63.

4.3.65 MAGN→

Recalls the deformation magnification.

Level 1	→	Level 1
	→	real number

Remarks:

See 4.3.63.

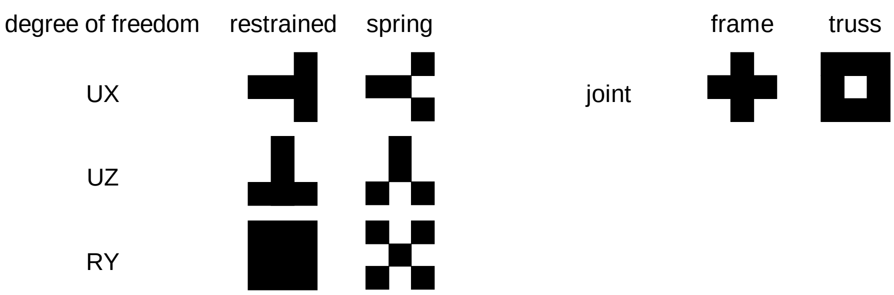
4.3.66 SUPP?

Toggles the plotting of supports

Level 1	→	Level 1	
1	→		enabled
0	→		disabled

Remarks:

Supports can not be plotted at the same time as the node numbers, see NN? command (4.3.67).



4.3.67 NN?

Toggles the plotting of node numbers.

Level 1	→	Level 1	
1	→		enabled
0	→		disabled

Remarks:
Node numbers can not be plotted at the same time as the supports, see SUPP? command (4.3.66).

4.3.68 MN?

Toggles the plotting of member numbers.

Level 1	→	Level 1	
1	→		enabled
0	→		disabled

Remarks:
None

4.3.69 →STK?

Toggles the export of plots to the stack.

Level 1	→	Level 1	
1	→		enabled
0	→		disabled

Remarks:
None

4.3.70 BZ?

Toggles BZ compression of files while saving them.

Level 1	→	Level 1	
1	→		enabled
0	→		disabled

Remarks:
See 4.3.6. The BZ compressor should be available as a library command. A program in a directory is not supported.

4.3.71 FBROW

Gives the user access to the custom browser used by FEM48 (much faster than the HP48 choose engine).

Level 4	Level 3	Level 2	Level 1	→	Level 2	Level 1	
"string1"	"string2"	{ choices }	init_pos	→	choice	choice_pos	ENTER key
"string1"	"string2"	{ choices }	init_pos	→		0	ON key

Remarks:
For use by user's own UserRPL programs. Is fastest when used with a choice list comprising of strings, since HP48 internal string conversion routine can be very slow for large objects (related to stack speed!). Only available from the FEM48 library menu, since unrelated to FEM48 workings.

4.3.72 ABOUTFEM

Recalls the version string.

Level 1	→	Level 1
	→	"string"

Remarks:
None

5. QUERY MODULE

5.1 Description

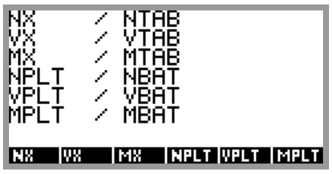
This module is optional and provides beam analysis facilities. The member to be analysed can be selected after which numerical and graphical output like axial force, shear force, bending moment, rotation and axial and transverse displacements can be obtained.

5.2 Menu structure

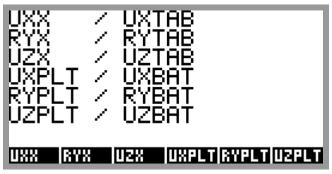
Below the pages of the QUERY menu will be shown as screenshots. Note that you can see all commands that are available in the order: unshifted, left-shifted and right-shifted. Toggle menu keys can also be used in the left and right shifted plane, the toggle command and it's argument are then echoed in the command line for easy programmable use.

QUERY menu row 1

QUERY menu row 2

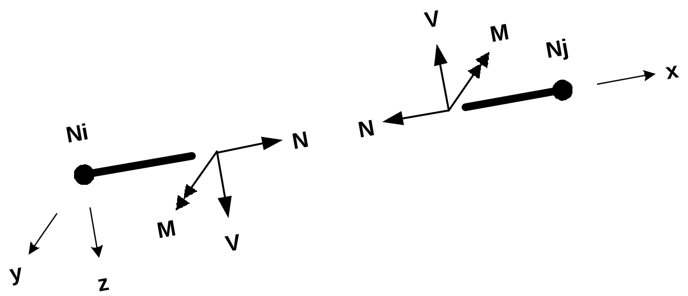


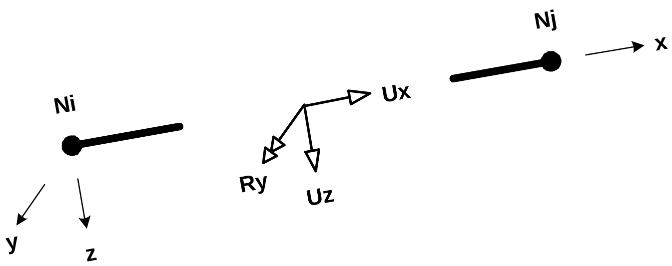
QUERY menu row 3



QUERY menu row 4

5.3 Sign of internal forces and displacements





5.4 Command reference

5.4.1 ABOUTFEMQUERY

Recalls the version string.

Level 1	→	Level 1
	→	"string"

Remarks:

None

5.4.2 QUERYFEM

Displays the QUERY menu.

Level 1	→	Level 1
	→	

Remarks:

Normally accessed through the use of the FEM menu (4.3.1).

5.4.3 MINFO

Displays information about the currently selected member.

Level 1	→	Level 1
	→	

Remarks:

See 5.4.4.

5.4.4 QMEM

Starts inputline prompt for the member to be queried.

Level 1	→	Level 1
	→	

Remarks:

None.

5.4.5 →QMEM

Stores the member to be queried.

Level 1	→	Level 1
integer	→	

Remarks:

None.

5.4.6 QMEM→

Recalls the member to be queried.

Level 1	→	Level 1
	→	integer

Remarks:

None.

5.4.7 KEYP

Starts inputline prompt for the number of keypoints along the queried member.

Level 1	→	Level 1
	→	

Remarks:

Member query data will only be calculated at the keypoints. Number of keypoints allowed: 2 ≤ keypoints ≤ 131. Keypoints are evenly distributed over the member.

5.4.8 →KEYP

Stores the number of keypoints along the queried member.

Level 1	→	Level 1
integer	→	

Remarks:

See 5.4.7.

5.4.9 KEYP→

Recalls the number of keypoints along the queried member.

Level 1	→	Level 1
	→	integer

Remarks:

See 5.4.7.

5.4.10 NX

Calculates the axial force at the local x coordinate of the queried member.

Level 1	→	Level 1
real number	→	real number

Remarks:

See 5.4.4.

5.4.11 VX

Calculates the shear force at the local x coordinate of the queried member.

Level 1	→	Level 1
real number	→	real number

Remarks:

See 5.4.4.

5.4.12 MX

Calculates the bending moment at the local x coordinate of the queried member.

Level 1	→	Level 1
real number	→	real number

Remarks:

See 5.4.4.

5.4.13 NTAB

Creates a table of values for the axial force at the local x coordinate of the queried member at each keypoint.

Level 1	→	Level 1	
	→	[[matrix]] $_{n \times 2}$	AV? (4.3.54) off
	→	unknown	AV? on

Remarks:

Column 1 contains the local x coordinates and column 2 contains the axial force. See also 5.4.4 and 5.4.7.

5.4.14 VTAB

Creates a table of values for the shear force at the local x coordinate of the queried member at each keypoint.

Level 1	→	Level 1	
	→	[[matrix]] $_{n \times 2}$	AV? (4.3.54) off
	→	unknown	AV? on

Remarks:

Column 1 contains the local x coordinates and column 2 contains the shear force. See also 5.4.4 and 5.4.7.

5.4.15 MTAB

Creates a table of values for the bending moment at the local x coordinate of the queried member at each keypoint.

Level 1	→	Level 1	
	→	[[matrix]] $_{n \times 2}$	AV? (4.3.54) off
	→	unknown	AV? on

Remarks:

Column 1 contains the local x coordinates and column 2 contains the bending moment. See also 5.4.4 and 5.4.7.

5.4.16 UXX

Calculates the axial displacement at the local x coordinate of the queried member.

Level 1	→	Level 1
real number	→	real number

Remarks:

See 5.4.4.

5.4.17 RYX

Calculates the rotation in radians at the local x coordinate of the queried member.

Level 1	→	Level 1
real number	→	real number

Remarks:

Calculation errors if queried member has a release at the start node Ni. See 5.4.4.

5.4.18 UZX

Calculates the transverse displacement at the local x coordinate of the queried member.

Level 1	→	Level 1
real number	→	real number

Remarks:

Calculation errors if queried member has a release at the start node Ni. See 5.4.4.

5.4.19 UXTAB

Creates a table of values for the axial displacement at the local x coordinate of the queried member at each keypoint.

Level 1	→	Level 1	
	→	[[matrix]] $_{n\,x\,2}$	AV? (4.3.54) off
	→	unknown	AV? on

Remarks:
Column 1 contains the local x coordinates and column 2 contains the axial displacement. See also 5.4.4 and 5.4.7.

5.4.20 RYTAB

Creates a table of values for the rotation in radians at the local x coordinate of the queried member at each keypoint.

Level 1	→	Level 1	
	→	[[matrix]] $_{n\,x\,2}$	AV? (4.3.54) off
	→	unknown	AV? on

Remarks:
Column 1 contains the local x coordinates and column 2 contains the rotation. Calculation errors if queried member has a release at the start node Ni. See also 5.4.4 and 5.4.7.

5.4.21 UZTAB

Creates a table of values for the transverse displacement at the local x coordinate of the queried member at each keypoint.

Level 1	→	Level 1	
	→	[[matrix]] $_{n\,x\,2}$	AV? (4.3.54) off
	→	unknown	AV? on

Remarks:
Column 1 contains the local x coordinates and column 2 contains the transverse displacement. Calculation errors if queried member has a release at the start node Ni. See also 5.4.4 and 5.4.7.

5.4.22 NPLT

Plots the axial force of the queried member at each keypoint.

Level 1	→	Level 1	
	→		→STK? (4.3.69) off
	→	grob	→STK? on

Remarks:
The queried member, the minimum and the maximum value are displayed above the plot. See also 5.4.4 and 5.4.7.

5.4.23 VPLT

Plots the shear force of the queried member at each keypoint.

Level 1	→	Level 1	
	→		→STK? (4.3.69) off
	→	grob	→STK? on

Remarks:
The queried member, the minimum and the maximum value are displayed above the plot. See also 5.4.4 and 5.4.7.

5.4.24 MPLT

Plots the bending moment of the queried member at each keypoint.

Level 1	→	Level 1	
	→		→STK? (4.3.69) off
	→	grob	→STK? on

Remarks:
The queried member, the minimum and the maximum value are displayed above the plot. The positive direction (upwards or downwards) depends on the state of ↑M? (5.4.34). See also 5.4.4 and 5.4.7.

5.4.25 NBAT

Performs a batchplot of the axial force at each keypoint for all members.

Level 1	→	Level 1 to n	
	→		→STK? (4.3.69) off
	→	grob	→STK? on

Remarks:
The queried member, the minimum and the maximum value are displayed above each plot. The delay between the plots can be set with the DLAY command (5.4.36). See also 5.4.7.

5.4.26 VBAT

Performs a batchplot of the shear force at each keypoint for all members.

Level 1	→	Level 1 to n	
	→		→STK? (4.3.69) off
	→	grob	→STK? on

Remarks:
The queried member, the minimum and the maximum value are displayed above each plot. The delay between the plots can be set with the DLAY command (5.4.36). See also 5.4.7.

5.4.27 MBAT

Performs a batchplot of the bending moment at each keypoint for all members.

Level 1	→	Level 1 to n	
	→		→STK? (4.3.69) off
	→	grob	→STK? on

Remarks:
The queried member, the minimum and the maximum value are displayed above each plot. The delay between the plots can be set with the DLAY command (5.4.36). The positive direction (upwards or downwards) depends on the state of ↑M? (5.4.34). See also 5.4.7.

5.4.28 UXPLT

Plots the axial displacement of the queried member at each keypoint.

Level 1	→	Level 1	
	→		→STK? (4.3.69) off
	→	grob	→STK? on

Remarks:
The queried member, the minimum and the maximum value are displayed above the plot. See also 5.4.4 and 5.4.7.

5.4.29 RYPLT

Plots the rotation (in radians) of the queried member at each keypoint.

Level 1	→	Level 1	
	→		→STK? (4.3.69) off
	→	grob	→STK? on

Remarks:
The queried member, the minimum and the maximum value are displayed above the plot. Errors if queried member has a release at the start node Ni. See also 5.4.4 and 5.4.7.

5.4.30 UZPLT

Plots the transverse displacement of the queried member at each keypoint.

Level 1	→	Level 1	
	→		→STK? (4.3.69) off
	→	grob	→STK? on

Remarks:
The queried member, the minimum and the maximum value are displayed above the plot. Errors if queried member has a release at the start node Ni. See also 5.4.4 and 5.4.7.

5.4.31 UXBAT

Performs a batchplot of the axial displacement at each keypoint for all members.

Level 1	→	Level 1 to n	
	→		→STK? (4.3.69) off
	→	grob	→STK? on

Remarks:
The queried member, the minimum and the maximum value are displayed above each plot. The delay between the plots can be set with the DLAY command (5.4.36). See also 5.4.7.

5.4.32 RYBAT

Performs a batchplot of the rotation (in radians) at each keypoint for all members.

Level 1	→	Level 1 to n	
	→		→STK? (4.3.69) off
	→	grob	→STK? on

Remarks:
The queried member, the minimum and the maximum value are displayed above each plot. The delay between the plots can be set with the DLAY command (5.4.36). Errors if queried member has a release at the start node Ni. See also 5.4.7.

5.4.33 UZBAT

Performs a batchplot of the transverse displacement at each keypoint for all members.

Level 1	→	Level 1 to n	
	→		→STK? (4.3.69) off
	→	grob	→STK? on

Remarks:
The queried member, the minimum and the maximum value are displayed above each plot. The delay between the plots can be set with the DLAY command (5.4.36). Errors if queried member has a release at the start node Ni. See also 5.4.7.

5.4.34 ↑M?

Toggles positive direction of bending moment plot.

Level 1	→	Level 1	
1	→		positive upwards
0	→		positive downwards

Remarks:
None.

5.4.35 AXIS?

Toggles the plotting of the x axis.

Level 1	→	Level 1	
1	→		enabled
0	→		disabled

Remarks:
None.

5.4.36 DLAY

Starts inputline prompt for the delay between batchplots.

Level 1	→	Level 1
	→	

Remarks:
Delay is in seconds, a value of -1 signals to wait for a keypress.

5.4.37 →DLAY

Stores the delay between batchplots.

Level 1	→	Level 1
real number	→	

Remarks:
See 5.4.36.

5.4.38 DLAY→

Recalls the delay between batchplots.

Level 1	→	Level 1
	→	real number

Remarks:
See 5.4.36.

6. WIZRD MODULE

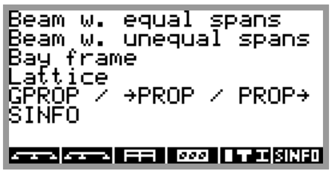
6.1 Description

This module is optional and provides wizards for entering geometry and properties of "standard" structures like beams, bay frames and lattices and for generating loads like selfweight and pre/post-stressing cables. A link to SED48, a Structural Engineering Database by the author of FEM48 is provided.

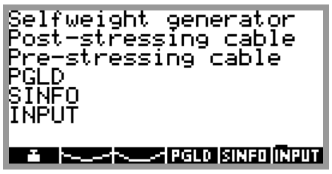
6.2 Menu structure

Below the pages of the WIZRD menu will be shown as screenshots. Note that you can see all commands that are available in the order: unshifted, left-shifted and right-shifted.

WIZRD menu row 1



WIZRD menu row 2



6.3 Command reference

6.3.1 ABOUTFEMWIZRD

Recalls the version string.

Level 1	→	Level 1
	→	"string"

Remarks:

None.

6.3.2 WIZRDFEM

Displays the WIZRD menu.

Level 1	→	Level 1
	→	

Remarks:

Normally accessed through the use of the FEM menu (4.3.1).

6.3.3 RBEAM

Starts inputline prompt for regular beam geometry.

Level 1	→	Level 1
	→	

Remarks:

The length per span and the number of spans are entered after which the nodes, members and supports are generated and stored (overwriting!) in the current structure. Structure type is set to FRAME. All other data is untouched.

All spans have the same length. All members have property reference 1. Each node is supported with a roll, with the exception of the first node, this has a pinned support.

6.3.4 IBEAM

Starts inputline prompt for irregular beam geometry.

Level 1	→	Level 1
	→	

Remarks:

The length per span is entered after which the nodes, members and supports are generated and stored (overwriting!) in the current structure. Structure type is set to FRAME. All other data is untouched.
All members have property reference 1. Each node is supported with a roll, with the exception of the first node, this has a pinned support.

6.3.5 BAYFR

Starts inputline prompt for bayframe geometry.

Level 1	→	Level 1
	→	

Remarks:

The width per bay, the height per bay, the number of horizontal bays and the number of vertical bays are entered after which the nodes, members and supports are generated and stored (overwriting!) in the current structure. Structure type is set to FRAME. All other data is untouched.
Horizontal members have property reference 1, vertical members have property reference 2. The nodes on the bottom row have a pinned support.

6.3.6 LATTI

Starts inputline prompt for lattice geometry.

Level 1	→	Level 1
	→	

Remarks:

The length per bay, the height and the number of bays are entered after which the nodes, members and supports are generated and stored (overwriting!) in the current structure. Structure type is set to TRUSS. All other data is untouched.
Horizontal members have property reference 1, vertical members have property reference 2 and diagonal members have property reference 3. The leftmost node on the bottom row has a pinned support, the rightmost node on the bottom row has a rolling support.

6.3.7 GPROP

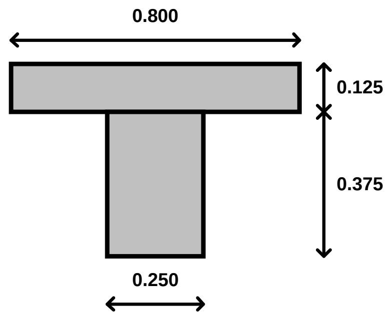
Starts properties generation browser.

Level 1	→	Level 1
	→	

Remarks:

A cross section type can be selected (Rectangle (multiple), Circle, Circular Ring, SED48 database[10], Numerical) and generated. The first cross section generated has property reference 1, the second one property reference 2, etc. The ON key quits the generation after which the properties are stored (overwriting!) in the current structure. All other data is untouched.

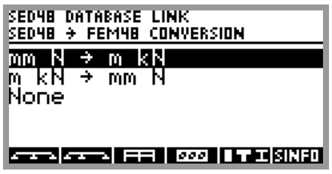
[10] Since FEM48 and SED48 both don't support units there is a unit conversion routine available for SED48 to FEM48.



The "Rectangle (multiple)" wizard can be used to generate properties of sections that are made of multiple rectangular sections, like T and I sections (and normal rectangles of course).

A T section (see picture) with a modulus of elasticity of 3E7 is entered like this: `[ 0.8 0.125 0.25 0.375 3E7 ]`

After the calculation the properties (h, A, e, I, E) are displayed on the screen and the program halts until a key is pressed[11]. When the [UpArrow] key is pressed the properties are also exported to the stack, for further calculations by the user.



6.3.8 SELFW

Starts inputline prompt for the generation of selfweight.

Level 1	→	Level 1
	→	

Remarks:

Selfweight can be generated for arbitrary inclined members. The density for each property reference has to be entered. Since each property has defined the cross sectional area the selfweight can be calculated (density * area). The selfweight is then stored in the current structure as MLX and/or MLZ loads. All other data is untouched. Note that the selfweight is <u>added</u> to the structure, no loads are replaced or overwritten. Note that it is thus possible to load a structure with it's selfweight more than once!

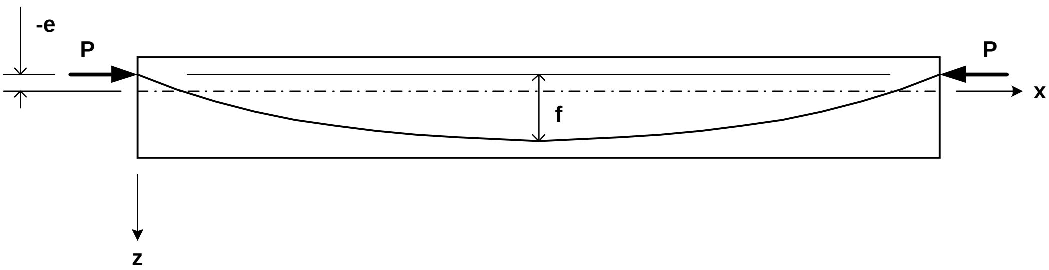
6.3.9 POCAB

Starts inputline prompt for the generation of forces introduced by a parabolic post-stressing cable.

Level 1	→	Level 1
	→	

Remarks:

The member, post-stressing force and the geometrical data of the cable must be entered. After this the loads introduced by the cable are calculated and <u>added</u> to the structure (no loads are replaced or overwritten). Note that by entering f=0 a straight cable can be generated.



[11] Only when a multiple rectangular section has been entered.

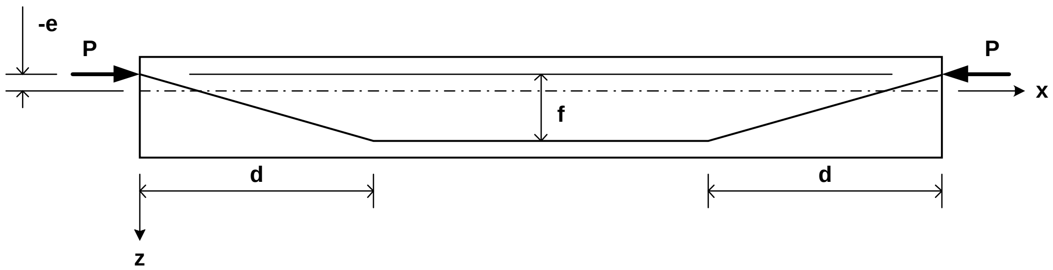
6.3.10 PRCAB

Starts inputline prompt for the generation of forces introduced by a polygon pre-stressing cable.

Level 1	→	Level 1
	→	

Remarks:

The member, pre-stressing force and the geometrical data of the cable must be entered. After this the loads introduced by the cable are calculated and added to the structure (no loads are replaced or overwritten). Note that by entering f=0 a straight cable can be generated.



7. PRINT MODULE

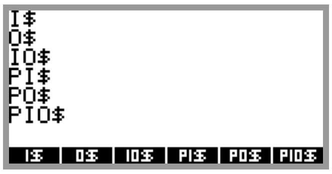
7.1 Description

This module is optional and provides commands for easy viewing and printing[12] of the input and output of FEM48.

7.2 Menu structure

Below the pages of the PRINT menu will be shown as screenshots. Note that you can see all commands that are available in the order: unshifted, left-shifted and right-shifted. Toggle menu keys can also be used in the left and right shifted plane, the toggle command and it's argument are then echoed in the command line for easy programmable use.

PRINT menu row 1



PRINT menu row 2

7.3 Command reference

7.3.1 ABOUTFEMPRINT

Recalls the version string.

Level 1	→	Level 1
	→	"string"

Remarks:

None

7.3.2 PRINTFEM

Displays the QUERY menu.

Level 1	→	Level 1
	→	

Remarks:

Normally accessed through the use of the FEM menu (4.3.1).

7.3.3 I$

Assembles a pretty print string of the input of the current structure.

Level 1	→	Level 1	
	→	"string"	AV? (4.3.54) off
	→	unknown	AV? on

Remarks:

None.

[12] If you own an HP printer. Note that it is also easy when using FEM48 on an emulator, this enables you to copy/paste the strings to a text editor on your PC.

7.3.4 O$

Assembles a pretty print string of the calculated results (output) of the current structure.

Level 1	→	Level 1	
	→	"string"	AV? (4.3.54) off
	→	unknown	AV? on

Remarks:

None.

7.3.5 IO$

Assembles a pretty print string of the input and the calculated results (output) of the current structure.

Level 1	→	Level 1	
	→	"string"	AV? (4.3.54) off
	→	unknown	AV? on

Remarks:

None.

7.3.6 PI$

Prints a pretty print string of the input of the current structure.

Level 1	→	Level 1
	→	

Remarks:

None.

7.3.7 PO$

Prints a pretty print string of the calculated results (output) of the current structure.

Level 1	→	Level 1
	→	

Remarks:

None.

7.3.8 PIO$

Prints a pretty print string of the input and the calculated results (output) of the current structure.

Level 1	→	Level 1
	→	

Remarks:

None.

7.3.9 STRV

Starts inputline prompt for an external stringviewer.

Level 1	→	Level 1
	→	

Remarks:

Can also be a text editor. Command should be a library command, a variable name is not allowed. When no external viewer is defined the internal HP48 editor is used. Also see 4.3.54.

7.3.10 →STRV

Stores the external stringviewer command as a string.

Level 1	→	**Level 1**
"string"	→	

Remarks:

See 7.3.9.

7.3.11 STRV→

Recalls the external stringviewer command as a string.

Level 1	→	**Level 1**
	→	"string"

Remarks:

See 7.3.9.

8. MOVLD MODULE

8.1 Description

This module is optional and provides a command for generating moving loads along an already defined beam[13]. Features are:

- define load system of an unlimited number of concentrated member loads and the interval + stepsize
- outputs the NDIS REAC MFOR minimums and maximums in the current directory
- when defined the user program MOVLD.USR is run after each step, so you can add your own postprocessing

8.2 Menu structure

The MOVLD module is available from the library menu as a separate library softkey (next to FEM48). There are only two commands available, MOVLD and ABOUTFEMMOVLD. These commands are not available[14] from the FEM menu.

8.3 Command reference

8.3.1 MOVLD

Calculates minimum and maximum values for the NDIS, REAC and MFOR output for a given load system and placecement interval and stepsize. A user program (MOVLD.USR) can be run after each iteration to perform any analysis desired. Load system loads are <u>added</u> to the already defined loads.

Level 2	Level 1	→	Level 1
[[matrix]] $_{n \times 4}$	[vector] $_{1 \times 3}$	→	

Remarks:

<u>Load system definition</u>

The load system is defined by a matrix:

```
[[ FX1 FZ1 MY1 x1 ]
 [ FXi FZi MYi xi ]
 [ FXn FZn MYn xn ]]
```

It is understood that $x_i < x_{i+1}$. `x1` is the first load of the load system and this x value is the reference point for the load system. The x values are local "load system" coordinates (but are added to the global coordinates for positioning!).

<u>Interval definition</u>

The interval and stepsize are defined by a vector: `[ Xstart Xend Xstep ]`
The X values are defined in the global coordinates system of the structure. For the first iteration the load system will be placed on the global X coordinate: `Xstart+x1`
This means that a non zero value of x for the first load(s) of the system will lead to a displacement of the load system from the global Xstart coordinate! Negative Xstart and Xend values are also allowed (this in order to let the load system move completely over a beam). Interval bounderies can fall beyond the beam at both ends if needed.

<u>Output</u>

After finishing, the minima (most negative) and maxima (most positive) of NDIS, REAC and MFOR are stored in the current directory as matrices (↑NDIS, ↓NDIS, ↑REAC, ↓REAC, ↑MFOR and ↓MFOR).
<u>UserRPL program</u>

[13] This can be a multi-spanned beam, with any kind of load and all kinds of supports.
[14] For FEM49: they are available from the FEM49 library menu and from the APPS choose box.

After each iteration, the program MOVLD.USR (when present in current or a parent directory) is run. This to enable the user to perform a secondary analysis for each step. For user's sake, the following "compiled local variables" are available:

←X : the current global X position of the load system
←M : the current member loaded by the first load of the load system
←x : the current local x position of the first load of the load system on the member ←M

Examples:

Display the local x value and the current MLC

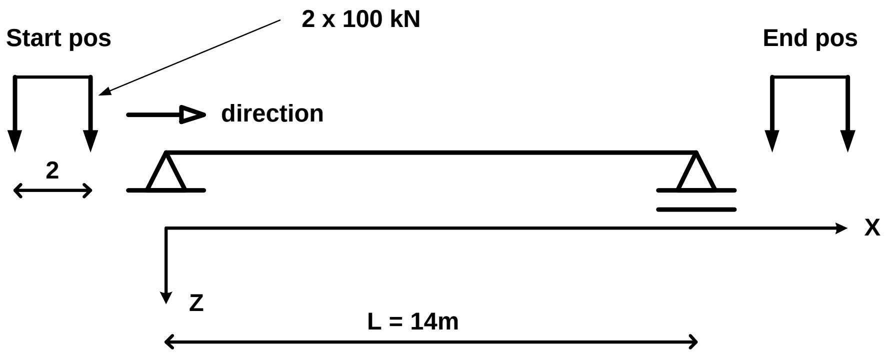
```
<<
  "Current local pos: "
  ←x + 4 DISP
  "Current MLC:" 5 DISP
  0 AV?
  MLC→ 6 DISP
>>
```

Influence of moving load on moment at a certain point (*using ΣDAT*)

```
<<
  ←X 1.5 MX         @ moment of member in QMEM at local x = 1.5 for current load pos
  2 →ARRY           @ create vector
  Σ+                @ sto in ΣDAT
>>
```

The MOVLD.USR program should not alter the stack (but is protected in case it does).

<u>Example of MOVLD use</u>



Load system:

```
[[ 0 100 0 0 ]
 [ 0 100 0 2 ]]
```

Interval and stepsize:

`[ -4 16 0.5 ]` (could also be `[ -2 14 0.5 ]`, then just on beam at first and last iteration)

8.3.2 ABOUTFEMMOVLD

Recalls the version string.

Level 1	→	Level 1
	→	"string"

Remarks:

None

9. APPENDIX

9.1 Tips and tricks

9.1.1 Hand calculations

When performing calculations by hand you do not take extension into account. So, you are actually saying: members do not deform axially. When checking such calculations with FEM48 you have to be aware that you need to make the axial stiffness of the structure members very large (make the cross sectional area very big), otherwise you will get answers that can not be compared with your hand calculation.

9.1.2 Inclined supports

FEM48 can not handle inclined supports by default. However, by using a trick you can! You replace the inclined support by a member that should be inclined in the same direction as the reaction force of the support. This member should be axially very stiff and have a release (4.3.25) at the node where the inclined support is wanted. At the other side of the member you define a pinned support. The axial force of the member is your reaction force! You may need to experiment a little bit before you get the axial stiffness and the length of the member ok.

9.1.3 Multiple structures

FEM48 does not check for multiple structures. So, you could take advantage of this and calculate two structures (unconnected by any member) at the same time.

9.2 Installation

The FEM48 library modules can be installed in any available port on the calculator. However, they work fastest from a non-covered port, that is port 0 or 1. To install the FEM48 library modules on your calculator do the following:

- Transfer the library module to your calculator and place the library module on the stack.
- Place the port number you wish to store the library module in on the stack and press the STO button.
- Warmstart the calculator (press ON and C simultaneously) or turn it off and on again.
- Purge the variable which still contains the library module.

When in doubt, refer to chapter 28 of the HP User's Guide.

9.3 Characteristics

Module	Library ID	"standard" size (bytes)	"standard" checksum (dec)	"compressed" size (bytes)	"compressed" checksum (dec)	optional?	miscellaneous
FEM48	1605	27111.5	36896	20416.0	2930	no	-
QUERY	1606	10030.5	48607	8285.5	44414	yes	nullnamed library
WIZRD	1607	8576.5	28526	7061.0	38865	yes	nullnamed library
PRINT	1608	4391.5	2078	2953.0	8624	yes	nullnamed library
MOVLD	1604	2305.5	18016	1931.5	33873	yes	-

System flags used: -20, -21, -22 (all cleared)
User flags used: none
Variables used: FEM_in, FEM_ck, FEM_out and FEM_cfg (all in hidden directory)
↑NDIS, ↓NDIS, ↑REAC, ↓REAC, ↑MFOR and ↓MFOR (by MOVLD module)
Language used: SystemRPL (99%) / Assembly (1%)

9.4 Memory usage

A good guesstimate is:

Using the Cholesky solver

$$memory = 1.1\left[15 + 8\cdot(3\cdot nodes)^2\right]$$

$$nodes = \frac{1}{3}\cdot\sqrt{\frac{\left(\frac{memory}{1.1} - 15\right)}{8}}$$

Using the internal HP solver

$$memory = 2.3\left[15 + 8\cdot(3\cdot nodes)^2\right]$$

$$nodes = \frac{1}{3}\cdot\sqrt{\frac{\left(\frac{memory}{2.3} - 15\right)}{8}}$$

9.5 Speed

FEM48 speed depends mostly upon the library version (standard or compressed) and the port it is stored (uncovered or covered). A good rule of thumb is:

- compressed versions execute 20% slower than the standard versions
- when stored in a covered port (2 or greater) the execution is about 13% slower than when stored in an uncovered port (0 or 1)

Measurements were done for the SCALC routine.

9.6 Limitations

Although FEM48 is very capable it does have its limitations:

- FEM48 should only be used for the calculation of structures which have relatively small displacements.
- It is understood that the members of a frame are about five times (or more) longer than they are in height. This because deformation due to shear force is not taken into account.

9.7 Credits

I would like to thank the following people:

- All users who gave support and asked questions (and thus motivated me to improve FEM48)
- Gjermund Skailand for allowing the use of his Cholesky matrix solver
- Edwin Córdoba for porting FEM48 to the HP49 (and doing this more than once!)
- Alex Tomanovich for beta-testing and giving suggestions for improving FEM48
- Alexander Reif for beta-testing and giving suggestions for improving FEM48
- Mika Heiskanen and Jan Brittenson for creating Jazz
- Sebastien Carlier and Christoph Gießelink for creating and improving EMU48
- Eric Rechlin for adding FEM48 to his wonderful site www.hpcalc.org

9.8 Author

FEM48 has been created by Caspar Lugtmeier.
Contact the author: c.lugtmeier@hccnet.nl
Visit the homepage: http://home.hccnet.nl/c.lugtmeier