FORTRAN CAD MANUAL

Version 26.02
Date 2026-2-15

Pere Hernàndez I Casellas

pere@phc-enginyeria.com

1. Introduction

FORTRAN CAD version 26.02 is an utility for creating DXF drawing files from a command sequence. All the drawing elements are programmed and parameterized.

Drawings created can be completed or transferred to other DXF drawings using other conventional CAD programs.

It can be used as a CAD graphic library for any programming language. The FortranCAD system is what I have been using in Fortran for many years. The program is very simple, and the results very useful. Anyone who works with programming or data files can learn the program easily.

2. How to use FORTRAN CAD

The program reads a data file that contains the information about the graphic entities to be drawn (lines, circles, polylines...). The result is a text file in DXF2000 format that can be edited with a CAD program.

The data file is read sequentially line by line. Line items are separated by one or more blank spaces or tabs.

The first element of the line is the command to be executed, followed by some parameters on the same line and in some cases more parameters on lines below.

Used parameters, variable type and interpretation.

x1 real x-coordinate point 1
y1 real y-coordinate point 1
y1 real z-coordinate point 1
x2 real x-coordinate point 2
y2 real y-coordinate point 2
z2 real y-coordinate point 2
....
z9 real z-coordinate point 9

rad real

a1,a2,a3,...,b1,b2,b3,... real parameters

layer_name character type up to 100 characters. Use between " " if layer name has bllanc espaces.

color integer type indicating the color according your ctb file (example 1=red, 2=yellow, 3=green...)

3. Commands

C

Any line beginning by letter c and then a blanc or tab espace (one single "c" or "C") are considered as code comment, and not interpreted

Example:

```
c This is a comment
```

LINE2D

Line between point 1 (x1,y1) and 2 (x2,y2), in layer "layer_name", and CAD color defined by integer.

Format:

```
line2d x1 y1 x2 y2 layer_name color
```

LINE3D

Line between point 1 and 2, in layer "layer_name", and CAD color defined by integer.

Format:

```
line3d x1 y1 z1 x2 y2 z2 layer_name color
```

Example:

LINE2DR

Line 2d relative increments x y defined by a1 b1

Format:

line2dr a1 b1 layer_name color

LINEP

Line from point (x1,y1), length l, road banking p, layer name, and CAD color
If iflag=1 line left to right

If iflag=-1 line right tot left

Format:

linep iflag x1 y1 l p layer_name color

POLY2D

2D polyline with n points, I layer layer_name, and CAD color. Then write coordinates (x,y) of the n points below.

Format:

poly2d n layer_name color
x1 y1
x2 y2
…….
xn yn

POLY3D

3D polyline with n points, I layer layer_name, and CAD color. Then write coordinates (x,y,z) of the n points below.

Format:

poly3d n layer_name color
x1 y1 z1
x2 y2 z2
…….
xn yn zn

EPOLY2D

Reads 2D polyline from an external file using rows like:
x1 y1
x2 y2
...

Format:

epoly2d filename.txt layer_name color

CHARTPOLY

Reads 2D polyline from an external file using rows like:
x1 y1
x2 y2
...
and prints longitudinal chart with axes elevation versus PK

Format:

chartpoly filename.txt layer_name color
title
x y
PKini PKfin steep
elevation_min elevation_max steep
vertical_scale text_size

EPOLY3D

Reads 3D polyline from an external file using rows like:
x1 y1 z1
x2 y2 z2
...

Format:

epoly3d filename.txt layer_name color

CIRCLE

Draws a circle centered in (x1,y1) radious r. layer "layer_name" and CAD color defined by integer.

Format:

circle x1 y1 r layer_name color

Example:

circle 1.0 2.0 0.55 "circles layer" 7

RECTANGLE

Draws rectangle

Format:
rectangle iflag x1 y1 x2 y2 layer_name color

ELLIPSE (not implemented yet)

CLOTOIDE

Format:

clotoide itype cw_or_ccw layer_name color
x0 y0 x1 y1
Ox Oy Fx Fy

Fixed clotoide type 1. Start point in (x0,y0). Tangent to (x0,y0) to (x1,y1) segment. Tangent to a circular alignement in point (Fx,Fy) and center (Ox,Oy),

Example:

clotoide 1 ccw alignement 1
0.0 0.0 1.0 0.0
5.0 6.0 2.0 6.2

ARC3P (not implemented yet)

TEXT

Example:

text x y htext atext layer_name color
"here the text"

x y are the initial coordinates, htext the height, atext clockwise angle in degrees

ITEXT

Example:

itext x y htext atext layer_name color
1317

x y are the initial coordinates, htext the height, atext clockwise angle in degrees
use only for writting integer numbers

IMAGE

Reference external Images in a separate file drawing_img.dxf

Format:

image x y z scale angle layer_name
pixels_x pixels_y
bright transparency (??? inspect DXF)
image_name

Example:

image 1.3 2.2 0.0 1.0 0.0 layer
1462 946
50.0 50.0
"PRIMERA.png"

BLOCK (not implemented yet)

XREF (not implemented yet)

Preconfigured objects

DITCH03

Draw triangular ditch.

Format:

ditch3 iflag x y h1 h1 v1 layer_name ncolor

iflag 1 = drw to the right; -1 = draw to the left
x y insertion point
h1 h2 horizontal parameters
v vertical depth

Example:

ditch03 -1 2.0 2.0 0.5 0.8 0.5 cuneta 6

DITCH04

Draw trapezoidal ditch.

Format:

ditch4 iflag x y h1 h2 h3 v layer_name ncolor

iflag 1 = draw to the right; -1 = draw to the left
x y insertion point
h1 h2 h3 horizontal parameters
v vertical depth

Example:
ditch04 1 2.0 4.0 0.5 0.5 0.8 0.8 cuneta 7

WALLT

Draw section of a "T" wall.

Format:

wallt iflag x y layer_name color
r s
h1 h2 h3
v1 v2

iflag 1 = draw to the right; -1 = draw to the left
x y insertion point
r=earth cover s=depth under crowning
h1=footing left h2=wall thickness h3=footing right

v1= wall height v2=footing thickness

Example:

wallt 1 0.0 0.0 murs 6
0.25 0.45
0.5 0.30 1.8
2.0 0.35

CURB_T3 (not implemented yet)

CURB_T2 (not implemented yet)

CURB_P1 (not implemented yet)

BNJ1

New Jersey type 1

Format:

bnj1 x y alpha xcase layer_name ncolor

Example:

BNJ1 1.0 1.0 0.0 c new 3

Modifiers

SETCURSOR

Set or force drawing cursor in point (x1,y1)

Format:

setcursor x1 y1

RELATIVE

This command does not take any parameters. Its meaning is that the insertion point of the next graphic entity is the end point of the previous graphic entity. If the graphic entity below has a specific insertion point, this is ignored (eg CIRCLE entity, LINE2D,...)

Format:

relative

SCALE (not implemented yet)

ROTATE2D (not implemented yet)

Rotates next graphical element (if elementt admits) in alpha degrees counterclockwise around its base point.

Format:

rotate2d alpha

ROTATE3D (not implemented yet)

MOVER

MOVETO

Moves graphical cursor to (x1,y1)

Format:
moveto x1 y1

TRANSLATE

Translate next object

Format:

moveto x1 y1

MIRROR

CIVIL

Axes

Logitudinal

Cross_sections

civil
Start a section with parameters for road design

Axe_name axe_number alignements_number

cv_nalign

cv_segment

cv_rad

cv_rad_flo

cv_euler

cv_param T/F

cv_pk 100 20

cv_pk_ini

cv_text_pk

cv_text_param

ST CONSTRUCTOR COMMANDS

VORAL

Road shoulder

Format:

voral iflag width banking layer_name ncolor

Example:

voral 1 1.2 2.0 road 8

CARRIL

Road lane

Format:

carril iflag width banking layer_name ncolor

Example:

carril 1 3.5 2.0 road 8

BERMA

Road verge

Format:

berma iflag width banking layer_name ncolor

Example:

berma 1 0.75 8.0 road 30

BANQUETA

Road XX

Format:

banqueta iflag width banking layer_name ncolor

Example:

banqueta 1 1.2 2.0 road 8

TERRAPLE

Road XX

Format:

terraple iflag width horiz vert layer_name ncolor

Example:

terraple 1 2.0 3 2 road 3

DESMUNT

Road XX

Format:

banqueta iflag width horiz vert layer_name ncolor

Example:

desmunt 1 2.0 2 3 road 3

Special commands

STL_SURF (not implemented yet)

Makes STL surfaces from two or more polylines

DXF_SURF (not implemented yet)

Makes DXF surfaces from two or more polylines

MOUNT_SECT (not implemented yet)

Mount 3D model from polylines