MACHINE INDEPENDENT FORTRAN 77

(in a Graphics Environment)

David Bradly
Bradly Associates Ltd, UK

Scope

Implementing a Graphics Library written in Fortran 77 to be machine independent raises many issues regarding the use of a so called Language Standard. Issues include those of the language itself, its relationship with the underlying machine architecture and operating system and the design of the compiler. This paper looks at these issues from the standpoint of successfully implementing our graphics library on over 30 machine and compiler combinations.

Introduction

GINO-F started its days as a assembler language library operating on a single computer system, but the decision over 25 years ago to rewrite it in Fortran paved the way to its long life and machine independence. The gradual conversion to Fortran 77 took place in the late 1980's and in order to maintain a single source code for as long as possible, the removal of Fortran 66 routines did not take place until 1990.

From the very early days, however, it was never possible to have exactly the same source on all implementations and so a separate 'SYSTEM' module was developed which contained those routines which varied according to the different machines GINO was implemented on. This module has obviously changed over the life of GINO-F to reflect the growing sophistication of the library in the ever changing graphics marketplace.

The areas to which this module has had to cater for are as follows:

1) Machine Architecture
2) File Handling
3) Input/Output
4) Character Encoding
5) Non-standard functions
6) Address plugging

The problems and advantages of different Fortran compilers in the development of a machine independent library such as GINO-F will also be discussed. Experience of FORTRAN 90 is currently being gained, but it is expected that most of the above issues will NOT be solved by its adoption in our work.

An extension of this problem is the requirement to access computer ports directly, so that the graphics information can be sent directly to the device during the operation of the program.

With the history of Fortran steeped in files consisting of formatted records, another omission is the ability to get a single character input (with no record terminator) from standard input. This is a fundamental requirement for low level graphics input from interactive devices.

Character Encoding

An ongoing problem on some computer architectures is the use of the EBCDIC character encoding for its input and output. Most machines use the ASCII character set and in fact all graphics devices (to my knowledge) also use this for communications. Apart from the problem of different translation tables being used when converting from ASCII to EBCDIC, there is also the problem of at which stage should the translation take place. The problem has been increased with the growth of networks with their own translation operations.

Non-Standard Functions

In spite of the standardization of many of the maths functions within Fortran, there are a number of areas where standard functions are still not provided. These are:

1) Current Date and Time
2) Access to system clock
3) User name enquiry
4) Bit handling (particularly shifting)
5) System environment names

Many of these must be required by thousands of applications using Fortran and yet no standard routines are available. The system environment name is a recent addition to the list as far as GINO-F is concerned but is a very powerful tool giving users the ability to set specific requirements. This facility has been available for many years within most operating systems and again many application programs must have need of it.

Address Plugging

A particular requirement of GINO-F is to be able store and change the address of a routine call by 'plugging' the address of one routine in place of another to alter the calling sequence at run-time. This facility is being more widely provided in Fortran with the LOC and REF functions but is by no means standard. Assembler language has often to be used to code this functionality.

Its uses in GINO-F are to provide a library with a wide selection of graphics device drivers, but with the linker only loading the ones actually required by the calling program, and the means to omit other large modules where their specific use is not required.

In both cases the initial calling sequence is coded so that a dummy routine is loaded by the linker. But when the user program makes reference to the required device driver or other major facility, this in turn makes reference to its associated code module through an EXTERNAL statement and so is brought in by the linker. When the routine is actually called, the address of the dummy routine is then replaced by the address of this major code module. The facility can not be provided through normal calling sequences because of the need to make the reference to the required code low down in the calling tree of GINO-F.

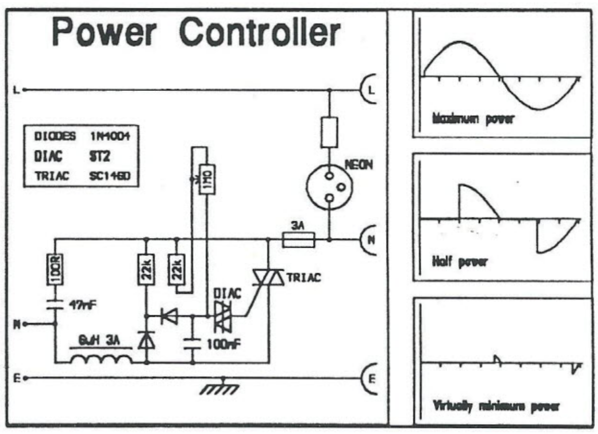
GINO

FORTRAN-90 Conference
Cologne, 16-17 November 1992
Kevin Bradly
Bradly Associates Ltd, England

GINO is a set of software routines written in FORTRAN which enables the operator to add graphics to their program quickly and easily, and on almost any computer. GINO was one of the first graphics systems to become available and through continual development, it remains a versatile, general-purpose system that can be used for a very wide range of purposes on a wide range of graphics systems.
Despite the range of hardware systems available, the number of truly device-independent portable software products remains remarkably small. The GINO family of products is the result of 25 years of effort and provides a set of graphics facilities suitable for the widest range of applications whilst retaining maximum portability and independence from the host computer and graphics peripheral.
During the design of GINO, it was appreciated that the range of applications was so varied, that the flexibility of a programming toolkit offered the only solution, hence GINO was designed as a library of ANSI FORTRAN subroutines. It provides all the basic facilities required by a systems programmer to write a graphics-based application of any complexity. The use of an extensive range of defaults makes the package simple enough to be used by a relatively inexperienced programmer who wishes to write a small graphics facility. The success of this design facility is illustrated by the range of host machines on which GINO has been implemented, the range of peripherals which can be driven and the varied environments in which it is used.
The three main GINO packages are as follows:

GINO-F

GINO-F is an extensive library of software routines for developing graphic data presentations and integrating them with existing applications. It is used in industries as varied as aerospace, defence, education, research, oil, gas and water and its applications include CAD/CAM, statistical analysis, architectural design and presentation graphics. Over 300 user-callable routines are available which can be incorporated into the users coding providing fast and efficient graphical interaction.

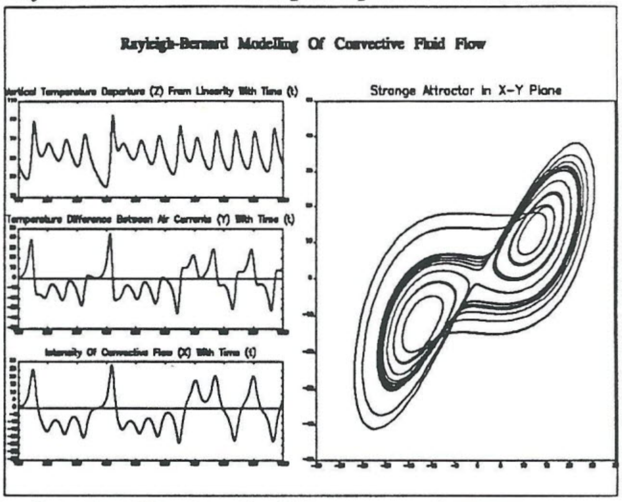


Features include: line, arc and curve drawing, masking and windowing, simple and complex polygon filling, interactive control, hardware/software segment structures, transformations, projections and viewing, character display with high quality polygonal-filled fonts, full colour control, metafile input and output, dialogue area control and run-time diagnostic facilities.

Over 200 output devices can be driven including colour, raster, thermal, ink-jet, laser and electrostatic printers, pen plotters and film recorders supporting all major graphics protocols including HPGL2, Regis, DXF and X-Windows. Metafile facilities include the storage and retrieval of CGM files allowing graphics integration with other applications such as desktop publishing.

GINOGRAF

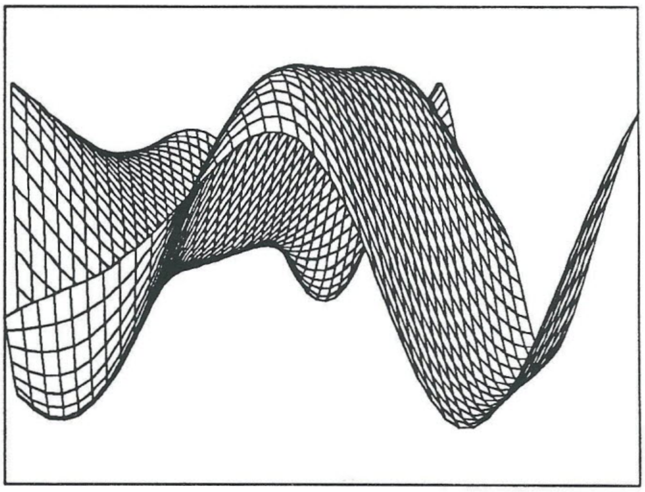
GINOGRAF is the complete software package for displaying numerical data in a range of graphical forms. The library of Fortran callable subroutines can be utilized to enhance users data display needs with the minimum of programming effort and unlike many other packages contains no limit to the number and styles of different output options available.



Bar charts, pie charts, X-Y graphs, histograms, and polar charts can all be generated from a single subroutine call. A whole host of composite routines are available which offer control of position, scaling and labelling of axes, line styles and symbol types, curve fitting characteristics, colour and solid fill capabilities, linear and logarithmic scales and titling.
GINOGRAF uses the low-level routines of the GINO-F package enabling the user to either fine-tune a particular style of output using additional administrative routines or creating a complex visualization plot by displaying the 2D data in 3D space using projection, transformation and viewing routines to create a final publication quality diagram. All facilities associated with GINO-F are available including access to the CGM metafile input/output and over 200 output devices.

GINOSURF

GINOSURF is an established multi-dimensional graphical display package providing a variety of surface representations by utilizing a library of Fortran callable subroutines.



Contour plots, isometric or perspective views and cross sectional diagrams can all be generated to perform accurate visual appraisal of your multi-dimensional data. This impressive and high performance surface mapping can all be created from a single subroutine call, providing a fully annotated, scaled and labelled picture. The multi-dimensional data can be supplied in gridded, random or functional form with interpolation control available to obtain the desired data configuration. The use of additional routines allow full colour control, scaling, direction of viewing, character and annotation control, area fill of specified data regions, axis control and full hidden surface facilities.
GINOSURF also uses the low-level routines of the GINO-F package, providing complete device and system independence. All facilities associated with GINO-F are available

including access to CGM and over 200 output devices supporting all major graphics protocols.

VERSION 4.0

An extensive and comprehensive development programme has culminated in the announcement of new releases of all three GINO products.
New features include:

New features of GINO-F V4.0

Viewport mapping providing the scaling and positioning of pictures to any area of the output device
Pixel rectangle read/write and transformation control
30 new routines for character and font control
Hardware font access
20 fixed/proportional software character fonts
1000+ symbols and dingbats
Text justification, fill style, underline, block structure, inter-line spacing, line-fit, curve following
Text exponents, indices, stroke weight
Device suspension and control allowing switching of graphics drawing
2D/3D polygon set definition and filling
Further transformation matrix control
Polygonal windowing and masking facilities
New event types including motion type events
New cursor types including rubber-banding
Error handling and tracer functionality extended
Additional software display file control including emulation of 'marking' and archiving
Picture segment transformations and new control

New features of GINOGRAF V4.0

Step charts
Area charts
Vector diagrams
Error bars using various bar types
Text charts: Numerical, text, auto-generated values, or graphical display
Improved annotation: Justification, truncation, angle, box types, location, skip values
Filling between data sets and axes
Fit a square wave to a set of data points
Hardware text switch
General purpose titling routine
Display a reference text string

New features of GINOSURF V4.0

Triangulation routines:
- Triangulation from random data
- Add a polygonal region to a triangulation
- Add a new point to a triangulation
- Draw contour maps from triangulated data
- Draw the boundary of the convex hull
- Draw the triangulation nodes and grid
- Compute rectangular grid from triangulation
- Interpolation method control
- Allow user specified first partial derivatives

Contours displayed on a surface
Contour slices displayed through a perspective projection
User supplied contour level specification
Fill style control of projected base frame
Back plane axes for projections, including terrain profiles

Background to GINO

GINO was originally developed at Cambridge University and then at the CADCentre also in Cambridge. Bradly Associates was formed in 1974 after its founder Peter Bradly left the CADCentre in Cambridge and set up as an agent for selling GINO. Originally Bradly Associates sold GINO on DEC equipment and then later on Control Data and Data General systems. Bradly Associates were also active in writing and marketing device drivers and other add-on products to GINO and these were made available to all users of GINO on a wide variety of equipment.
In 1988, Bradly Associates took over all responsibility for the total development and marketing of GINO from CADCentre and has now expanded employing a full development team of software engineers.
Its marketing activities include direct selling on all popular computer systems and the increasing use of dealers and distributors in the UK and worldwide for increased product distribution. Other areas of activity include be-spoke application programming and consultancy to end-users of GINO and other major graphics systems.