

19
2 Eje

Universidad Nacional Autónoma de México

FACULTAD DE INGENIERIA

SISTEMA GENERADOR DE GRAFICAS
Y REPORTES PARA ECONOMISTAS
GRAFOS

T E S I S

QUE PARA OBTENER EL TITULO DE:
INGENIERO EN COMPUTACION

P R E S E N T A N:

LUIS DANIEL RODRIGUEZ JIMENEZ
FAUSTINO SALAZAR MATA

Dir. Mstro. Hugo Raymundo Rangel Gutiérrez





Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

C O N T E N I D O

PAGINA

1.	INTRODUCCION	1
1.1	JUSTIFICACION	4
2.	MEDIO AMBIENTE	8
2.1	CONFIGURACION DEL EQUIPO	10
2.2	SISTEMA PROCESADOR DE INFORMACION ECONOMICA (SPIE)	13
2.3	SERIES DE TIEMPO	16
2.4	BANCOS DE DATOS	18
2.4.1	PARTES DE UN BANCO DE DATOS	19
2.5	RELACION GRAFOS - MALER	21
3.	ESTRUCTURA DEL PAQUETE	23
3.1	ETAPAS DEL PAQUETE	27
3.1.1	DEFINICION DE GRAFICAS	29
3.1.2	MODIFICACION DE GRAFICAS	30
3.1.3	EJECUCION DE GRAFICAS	31
3.2	LENGUAJE DE COMANDOS	32
3.3	ESTRUCTURAS DE DATOS PRINCIPALES	34
3.3.1	ARCHIVO "DESCRIPTOR" DE LA GRAFICA	35
3.3.2	DIRECTORIO FALTA DE INTEGRAR	36
3.3.3	ARCHIVOS DE MENSAJES	37

C O N T E N I D O

PÁGINA

4.	COMPONENTES DE UNA GRAFICA	40
4.1	ZONAS DE UNA SUBAREA GRAFICA	44
4.2	ELEMENTOS DE UNA SUBAREA GRAFICA	46
4.2.1	PARAMETROS	47
4.2.2	ENCABEZADOS	49
4.2.3	ACOTACIONES	54
4.2.4	PIES DE PAGINA	57
4.2.5	LETREROS POSICIONADOS	58
4.2.6	DETERMINACION DE DATOS	61
5.	DESCRIPCION DE PROGRAMAS	64
5.1	CARGADOR DE INSTRUCTIVO	65
5.2	CARGADOR DE MENSAJES DE ERRORES	67
5.3	DEFINICION DE GRAFICAS	68
5.4	EJECUCION DE GRAFICAS	136
5.5	MODIFICACION DE GRAFICAS	223
6.	EJEMPLOS	259
	BIBLIOGRAFIA	272
	APENDICES	

C O N T E N I D O

PAGINA

- A) MANUAL DE USUARIO GRAFOS
- B) LISTADO DE MENSAJES DE ERRORES
- C) LISTADO DE AYUDAS
- D) SUBROUTINAS DE VERSAPLOT

A N E X O S

LISTADO DE PROGRAMAS

I. INTRODUCCION:

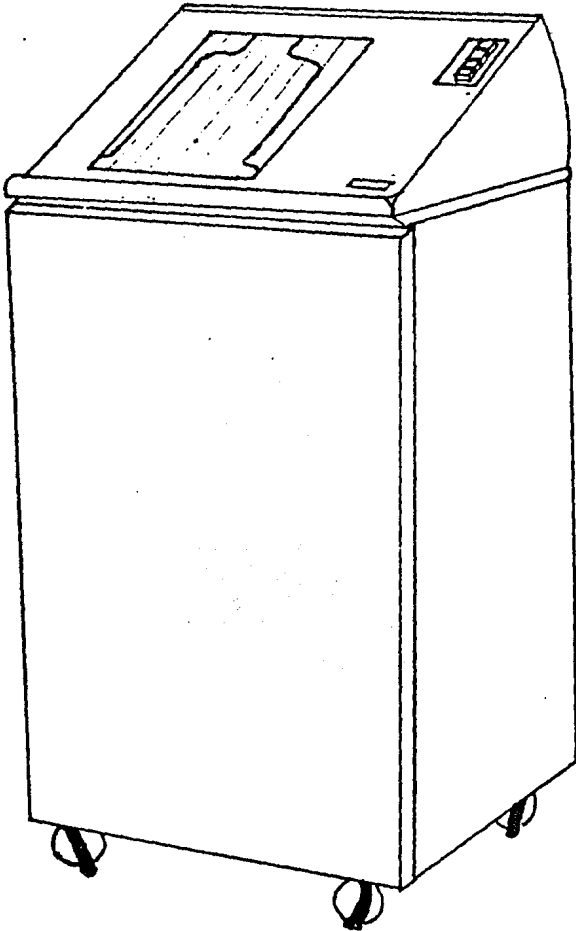
El sistema tiene como objetivo cumplir con los requerimientos de graficación de cualquier analista o investigador del área de Economía o Finanzas sin conocimientos de cómputo o programación. De tal forma que pueda definir, redefinir, modificar y ejecutar tantos procesos de graficación como sus necesidades lo requieran. Todo esto sin la necesidad de los servicios directos del especialista en cómputo, mas que para capacitación y asesoría.

Es evidente que por cumplir con las necesidades específicas de una instalación y de una cierta área; el sistema cuenta con algunas partes rígidas. Sin embargo, no sería difícil cambiarlo de instalación o aplicación en caso de que fuera requerido, ya que por un lado; los programas fueron implementados en FORTRAN, en la forma más standar posible, y por otro existen algunos elementos que pueden ser afines a diversas áreas de aplicación.

La principal causa por la que puede comportarse más rígido es porque el sistema se desarrolló para un graficador electrostático VERSATEC 1200-A.

FIGURA 1

**GRAFICADOR ELECTROSTATICO VERSATEC
MODELO 1200-A**



Existen varios elementos que unidos logran cumplir los objetivos del sistema GRAFOS dichos elementos son:

Lenguaje de Comandos similar al Lenguaje Cotidiano; - capacidad de modificar un proceso ya definido (Editor Propio), instructivo integrado; mezcla de gráficos y tipos de gráficas; etc. Estos elementos serán descritos a través de todo el texto.

En el capítulo 2 se explican todos los elementos específicos, ajenos al sistema que influyen en él directamente (Medio - Ambiente). En el capítulo 3 se explican las tres funciones principales del sistema y las características adicionales que lo hacen versátil y manejable; en el capítulo 4 se explican los conceptos específicos relativos a gráficas para presentación de informes: En el capítulo 5 se explica con cierto detalle, la implementación de cada programa con sus respectivas rutinas y en el capítulo 6 se muestran ejemplos elaborados con el sistema. Existen también cuatro apéndices para consulta en el caso de duda o deseo de profundizar; estos apéndices son : A) el manual del usuario, en el cual se explican todos los comandos de GRAFOS; B) el listado de ayudas (instructivo integrado) ; C) el listado de mensajes de error; D) la descripción de los programas ("Software") proporcionados por el proveedor para manejo del graficador. También pueden consultarse en un anexo adicional los listados de los programas.

1.1

JUSTIFICACION DEL SISTEMA

En el ámbito del análisis y la planeación económica y fi nanciera, es necesario contar con elementos que presenten resulta- dos de las investigaciones realizadas sobre dicha área. Así pues, puede verse que constantemente los grupos de analistas financieros- emiten sus resultados en forma de documentos o publicaciones que - normalmente constan de tres elementos básicos; textos, cuadros nu- méricos y gráficas.

Siendo la Secretaría de Hacienda uno de los organismos que mayor responsabilidad tienen en materia de Planeación Econó- mica y Financiera, es muy importante contar con productos eficien- tes que muestren claramente los resultados de análisis y estudios- estadísticos que influyen en la toma de decisiones que afectan a -- todo el país.

En la Coordinación de Informática se contaba ya con herra- mientas lo suficientemente capaces para manipular y analizar la in formación estadística que constantemente capta la Secretaría de Ha- cienda. Asimismo, por lo que respecta a la elaboración de reportes- numéricos, con leyendas o textos aclaratorios se tenían elementos bas- tante versátiles y eficientes. Sin embargo, por lo que respecta a gra-

ficación automatizada, se tenían elementos bastante pobres. Ya que sóloamente se contaba con graficación de "histogramas" y "polígonos" creados por los caracteres de una impresora de martillos, con la cual se obtenía una resolución muy limitada y a la vez la versatilidad con respecto a la información a graficar era mínima. (Véase figura 1.1.A).

Pese a las limitaciones existentes sobre dicho punto, la Dirección General de Planeación Hacendaria se veía precisada a presentar resultados de alta calidad y que como ya se mencionó mezclan tres elementos que se apoyan y estimulan entre sí; textos, gráficas, y números. Por lo tanto se creó un grupo de dibujantes que se encargaron de elaborar manualmente un buen número de gráficas que posteriormente combinaban con textos y reportes numéricos para obtener así los productos con la calidad de presentación requerida por los directivos.

Sin embargo, las representaciones gráficas sólo estaban limitadas para algunas personas de cierta jerarquía, ya que al elaborarse el trabajo manualmente el grupo de dibujantes se saturaba rápidamente con un sin fin de peticiones. En estas condiciones, el analista "standar" se veía limitado al presentar los resultados de sus investigaciones. Ya que, podía realizar reportes numéricos bastante versátiles en forma autónoma y automática; pero si -

...../

deseaba enriquecer o sustituir estos por representaciones gráficas solo tenía dos opciones; presentar las gráficas de baja calidad elaboradas con impresora o entrar en una larga lista de espera para que su trabajo fuera elaborado con buena calidad. Además, generalmente el analista desea graficar solo un período determinado de algún fenómeno y el programa de graficación, sólo podría graficar el fenómeno completo.

Por todas estas razones se vió la necesidad de adquirir un graficador que obviamente, sí podría proporcionar la calidad suficiente en el trazo de gráficas, las cuales ya serían equiparables con el trabajo de un dibujante.

Por otro lado, era necesario crear un sistema de graficación que permitiera al usuario del Centro de Cómputo realizar en forma autónoma todas las gráficas que pudiera necesitar. Además, el sistema no debería requerir de lenguajes de programación, ya que generalmente el usuario en este caso no es especialista en cómputo. Asimismo, requería de versatilidad tanto en los tipos de gráficas a realizar como en la periodicidad de los datos. El sistema debería también, ser capaz de mezclar varios elementos. Y por último debería aprovechar hasta donde fuera posible, las bases de programación ya existentes.

Así pues, se fueron realizando muestreos y sondeos sobre las necesidades más generales de graficación hasta que se llegó a un diseño preliminar y posteriormente se obtuvo finalmente el diseño definitivo, cuyas características se describirán a lo largo de esta tesis.

020.P.01

SISTEMA PROCESAL DE INVESTACION ECONOMICA
ESTADISTICA GAFICA DE DISTRIEION LABORAL DEL IMPRESO

FECHA 18/07/77
LUGAR CALI
PAIS COLOMBIA
PARQUE PARQUE

420.0
418.0
416.0
414.0
412.0
410.0
408.0
406.0
404.0
402.0
400.0
398.0
396.0
394.0
392.0
390.0
388.0
386.0
384.0
382.0
380.0
378.0
376.0
374.0
372.0
370.0
368.0
366.0
364.0
362.0
360.0
358.0
356.0
354.0
352.0
350.0
348.0
346.0
344.0
342.0
340.0
338.0
336.0
334.0
332.0
330.0
328.0
326.0
324.0
322.0
320.0
318.0
316.0
314.0
312.0
310.0
308.0
306.0
304.0
302.0
300.0
298.0
296.0
294.0
292.0
290.0
288.0
286.0
284.0
282.0
280.0
278.0
276.0
274.0
272.0
270.0
268.0
266.0
264.0
262.0
260.0
258.0
256.0
254.0
252.0
250.0
248.0
246.0
244.0
242.0
240.0
238.0
236.0
234.0
232.0
230.0
228.0
226.0
224.0
222.0
220.0
218.0
216.0
214.0
212.0
210.0
208.0
206.0
204.0
202.0
200.0
198.0
196.0
194.0
192.0
190.0
188.0
186.0
184.0
182.0
180.0
178.0
176.0
174.0
172.0
170.0
168.0
166.0
164.0
162.0
160.0
158.0
156.0
154.0
152.0
150.0
148.0
146.0
144.0
142.0
140.0
138.0
136.0
134.0
132.0
130.0
128.0
126.0
124.0
122.0
120.0
118.0
116.0
114.0
112.0
110.0
108.0
106.0
104.0
102.0
100.0
98.0
96.0
94.0
92.0
90.0
88.0
86.0
84.0
82.0
80.0
78.0
76.0
74.0
72.0
70.0
68.0
66.0
64.0
62.0
60.0
58.0
56.0
54.0
52.0
50.0
48.0
46.0
44.0
42.0
40.0
38.0
36.0
34.0
32.0
30.0
28.0
26.0
24.0
22.0
20.0
18.0
16.0
14.0
12.0
10.0
8.0
6.0
4.0
2.0
0.0

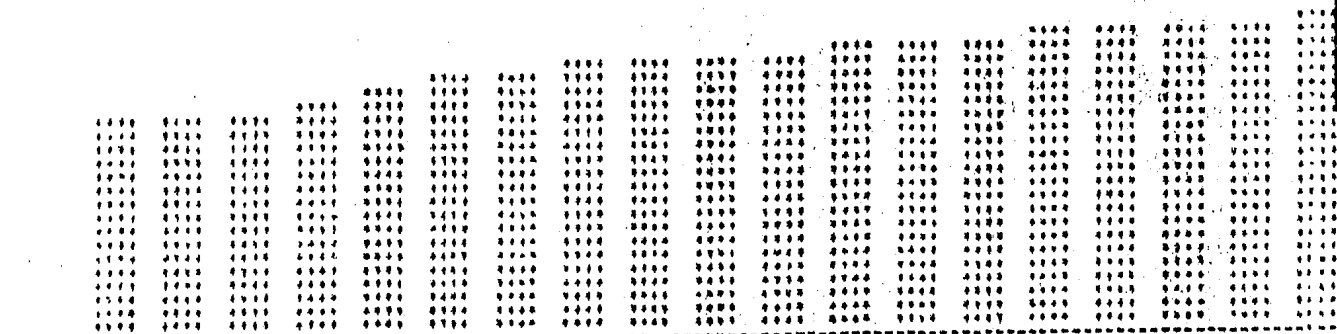


FIGURA 1.1.B

70c. 1 70c. 2 70c. 3 70c. 4 70c. 5 70c. 6 70c. 7 70c. 8 70c. 9 70c. 10 70c. 11 70c. 12 70c. 13 70c. 14

2. MEDIO AMBIENTE.

Una de las características más importante de un sistema, son sus cualidades de interacción con otros sistemas.

En el área de cómputo, este aspecto es fundamental, e incluso podría afirmarse que un programa es tanto más valioso - mientras mayor sea su explotación. Por supuesto este criterio puede ser considerado demasiado programático no obstante, dicha condición no debe ir en detrimento de la calidad del sistema.

En el caso de GRAFOS, se trata precisamente de un proyecto que satisfaga un conjunto de necesidades bien específicas y que sin embargo pueda tener la suficiente versatilidad como - para no agotar sus posibilidades en poco tiempo.

Por un lado, fue necesario analizar profundamente el medio ambiente en el cual sería aplicada la herramienta buscando así que fuera completamente útil y práctica. Por el otro, hubo necesidad de darle cierta versatilidad para no convertirlo - en un sistema de propósito tan específico que sólo cubriera las - necesidades temporalmente inmediatas del usuario.

En este capítulo, se describirá con cierto detalle, el sistema general en donde se desarrolla GRAFOS, mientras que-

en los demás capítulos se explica como se intentó dar a GRAFOS capacidad para satisfacer nuevas necesidades de presentación. Intentando así, que sea una herramienta de composición en vez de un pequeño paquete de graficación.

2.1. CONFIGURACION DEL EQUIPO

El ámbito de un sistema automatizado, está compuesto por dos partes principales: los recursos físicos o "Hardware" y por otro lado los programas y entidades abstractas o "Software". En este subtema se describirán los componentes físicos, y en los siguientes subtemas del mismo capítulo se tratarán las relativas al "Software".

El conjunto de elementos que componen el equipo son:

- 1.- Procesador Central de 16 bits
modelo PDP-11/40
- 2.- Memoria Central de
256 "Kwords" o 512 "Kbytes"
- 3.- Unidad de Disco Removible
Modelo RP04 con capacidad de
almacenamiento de 80 Mega-bytes
- 4.- Unidad de Disco Removible
Modelo RP06 con capacidad de
almacenamiento de 170 Mega-bytes
- 5.- Unidad de Cintas
Modelo TUI6 para formato
ANSI estándar.
- 6.- Impresora Mécanica LP34
con velocidad de impresión de
600 líneas por minuto.

- 7.- Impresora Graficadora Electroestática
Modelo Versatec 1200-A cuya
densidad es de 200 puntos por pulgada
tanto en "X" como en "Y"
- 8.- Consola de Operacion LPA-11
- 9.- Diez Terminales de Video VT50
- 10.- Cinco Terminales de Video VT100

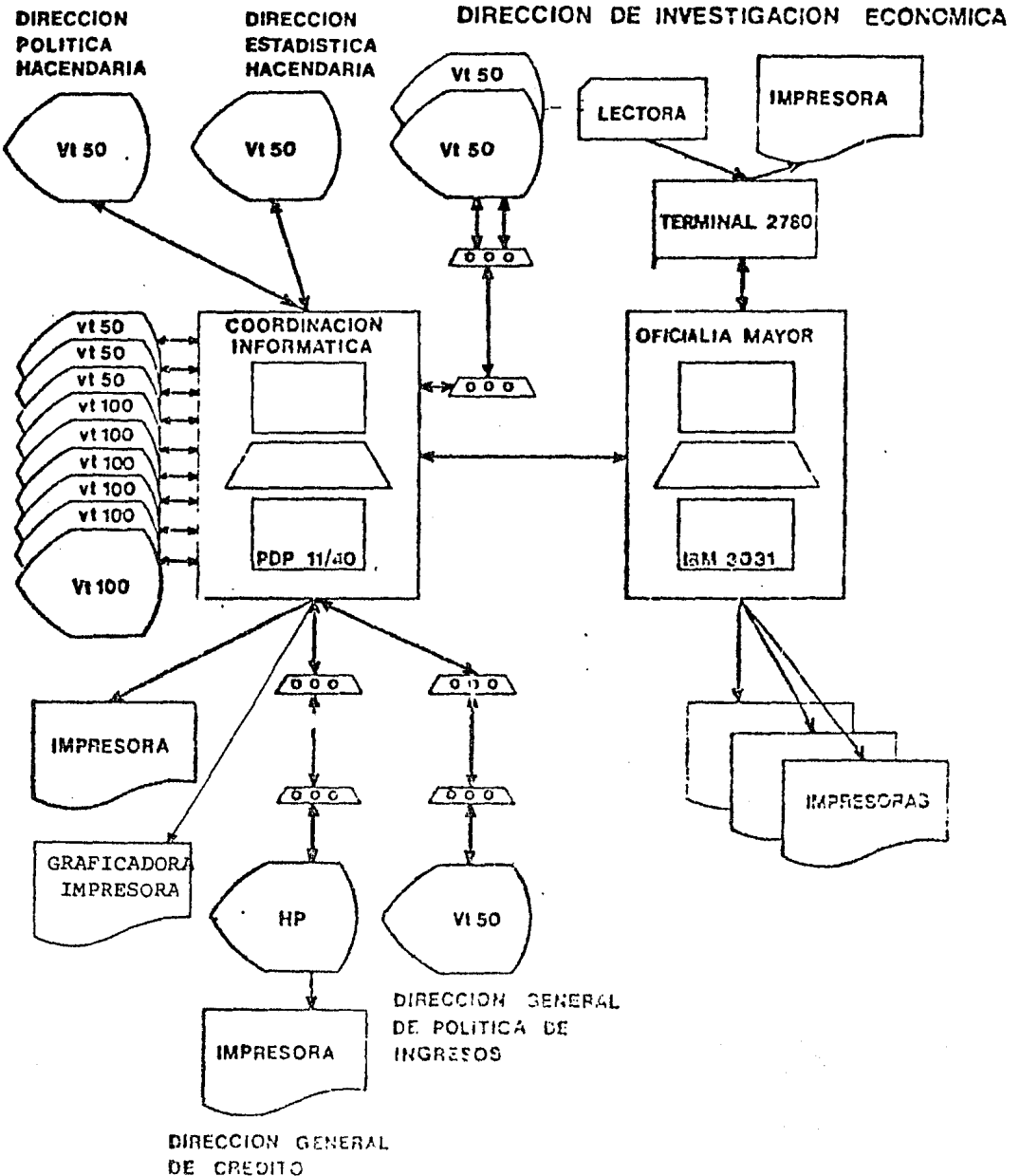
Cuatro de las terminales funcionan como remotas, mientras que las once restantes son locales.

Cabe mencionar que el Sistema Operativo del equipo es el RSX-11M Versión 3.2 cuya característica principal es que - permite la multiprogramación y facilita el desarrollo de programas interactivos.

Las dos características mencionadas, permiten trabajar a múltiples usuarios, aparentemente en forma simultánea y disponiendo de todos los recursos del sistema.

Esquemáticamente la configuración del equipo se muestra en la figura 2.1.A.

CONFIGURACION DE RED DE COMPUTO DE LA DIRECCION GENERAL DE PLANEACION HACENDARIA COORDINACION DE INFORMATICA



2.2. SISTEMA PROCESADOR DE INFORMACION ECONOMICA (SPIE)

Como ya se mencionó al principio del capítulo, GRAFOS no se da solo, por el contrario, surge como una necesidad de enriquecer las capacidades de presentación de un sistema, ya establecido; el SPIE.

SPIE puede definirse como un conjunto compuesto por programas; estructuras de datos, convenciones; información y recursos; interrelacionados entre sí de tal forma que permiten al investigador del área económico - financiera analizar los fenómenos de dicha área.

En términos estrictos, SPIE no puede ser considerado como una "Base de Datos" desde el punto de vista teórico, ya que no cuenta con las características principales de una B.D. convencional. Por mencionar algunas cabe indicar que la información no está centralizada; por otro lado, la redundancia de información no se controla con ningún mecanismo del sistema; por último, la relación entre distintas unidades de información no obedece a los esquemas típicos de una B.D.

En términos generales puede decirse que se cuenta con un "Sistema de Información" con una cierta estructura, que

si bien no es idónea teóricamente, tampoco debe considerársele incapaz de resolver en forma eficiente, necesidades de información y análisis.

Una representación gráfica del SPIE puede observarse en la figura 2.2.A.

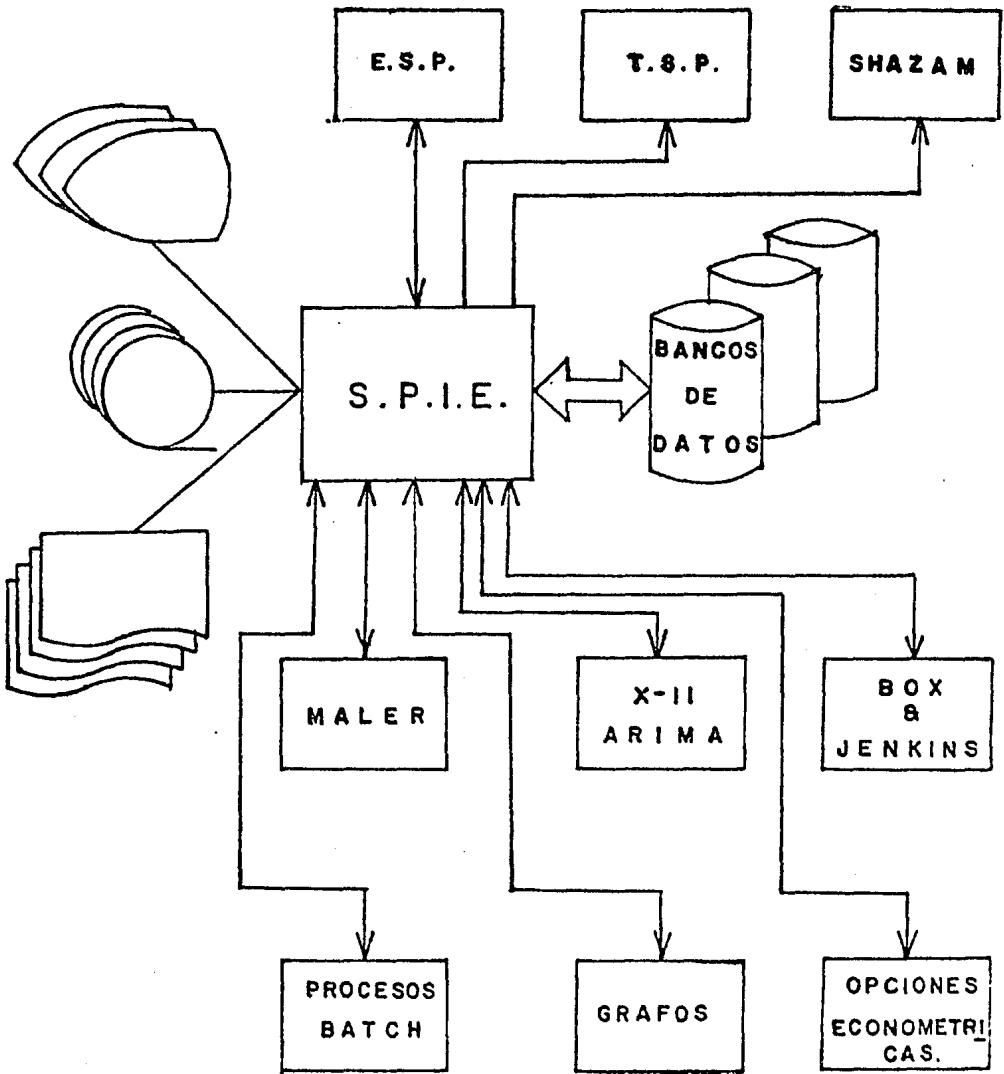
En dicha figura, puede apreciarse varios elementos, como son:

- A) Los Bancos de Datos, que constituyen la estructura básica de almacenamiento de la información. Y que posteriormente serán descritos más ampliamente.
- B) El módulo central que puede ser considerado como el "Manejador de la información, y que también establece las convenciones generales del sistema.
- C) Los módulos adicionales o complementarios se encargan de satisfacer en una forma más amplia, algunas de las funciones del sistema - (TSP, GENERA. MALER. BOX & JENKINS, GRAFOS, etc.)
- E) Por último se ven esquematizados, los recursos físicos del sistema.

En los próximos subtemas se detallarán algunos conceptos del Sistema Procesador de Información Económica, los cuales serán de utilidad para comprender posteriormente las especificaciones del subsistema GRAFOS.

FIGURA 2.2.A

SISTEMA S.P.I.E.



2.3. SERIES DE TIEMPO

En términos generales, las series de tiempo son empleadas para registrar estadísticamente un fenómeno determinado. La observación del fenómeno a través del tiempo se hace en intervalos regulares y en forma discreta, aún cuando al fenómeno pudiera darse en forma continúa.

Para el SPIE, el concepto básico es el mismo, sin embargo, existe un conjunto de elementos que siempre van asociados a la serie de tiempo. Así pues, los elementos que componen una serie de tiempo son:

- A) Observaciones: es el conjunto de valores numéricos que describen cuantitativamente el comportamiento del fenómeno a través del tiempo.
- B) Periodicidad: indica la duración de los intervalos regulares de tiempo que transcurren para registrar el fenómeno. En SPIE solo se consideran tres tipos de periodicidad: - anual, trimestral y mensual.
- C) Año y período de las observaciones: indican en que tiempo fueron registradas cada una de las observaciones.
- D) Nombre: Es una cadena de dieciseis caracteres - que normalmente indican el nombre común con que se conoce el fenómeno.

- E) **Unidades:** indica el patrón utilizado para medir el fenómeno en cuestión.
- F) **Fuente:** es una cadena de dieciséis caracteres que indican la forma en que fue obtenida la información.
- G) **Descripción:** es una cadena de cincuenta y seis caracteres que se usa para describir por medio del lenguaje cotidiano el tipo de fenómeno que se analiza.

Normalmente todos los componentes antes mencionados conforman una serie de tiempo, sin embargo, no todos son indispensables para el sistema. Los elementos que pueden estar -- vacíos o indeterminados sin que afecten la operación de SPIE -- son los mencionados en los incisos: D al G. Y los que sí -- son indispensables son los primeros tres (A, B, y C.).

Por último cabe mencionar que las series de tiempo son las componentes básicas con las cuales funciona el Sistema Procesador de Información Económica (SPIE). Y por consiguiente todos los subsistemas que lo conforman, incluyendo a GRAFOS.

2.4. BANCOS DE DATOS.

El siguiente concepto por determinar es el "Banco de Datos". El cual cuenta ya con más elementos concretos, en comparación con la serie de tiempo.

En primera instancia, puede definirse al "Banco de Datos" como un conjunto de "series de tiempo agrupadas, cuya periodicidad es la misma y que además dicho conjunto se identifica por un nombre o nemónico de seis caracteres alfa-numéricos

Como las series de un banco tienen la misma periodicidad, los bancos también se clasifican en; trimestrales, mensuales y anuales.

Para indicar la periodicidad de un banco se tiene la convención de que el nombre de éste contendrá en el sexto carácter un dígito que varíe de uno a tres. Y de acuerdo con esto se tendrá:

1. Banco Anual
2. Banco Trimestral
3. Banco Mensual

(Véase ejemplos en la siguiente tabla).

Tabla 2.4.1.		Nombres de Bancos	
Nombre del Banco		Tipo del Banco	
FLUJO	1	Anual	
GASTO	1	Anual	
IMPUE	3	Mensual	
INGRE	2	Trimestral	
FLUJO	3	Mensual	

Para identificar a cada una de las series que contiene un "Banco de Datos" se utiliza un número entero, el cual es usado como llave única. Así pues cada serie de tiempo tendrá asociado además un número de serie dentro del banco.

2.4.1. PARTES DE UN BANCO DE DATOS.

Cada Banco de Datos está compuesto por dos archivos - que son: el de datos y el de documentación. El archivo de Datos contiene los elementos indispensables para determinar una "serie de tiempo" y el archivo de documentación contiene los no indispensables. Esto de acuerdo con la clasificación hecha en el punto 2.3.

Existe una correspondencia biunívoca entre las series que contiene un "Banco de Datos"; los registros del archivo de datos -

y los registros del archivo de documentación. Así pues, por - - cada serie del banco existirá un registro en el archivo de datos y uno en el archivo de documentación.

Cabe mencionar además, que ambos archivos son de acceso directo y que el número de la serie coincide con el número de registro.

Al crear un banco, queda determinado el número de observaciones máximas que puede contener, así como también el número máximo de series que pueda almacenar.

Los dos parámetros antes mencionados, y el nombre del -- Banco en cuestión, se graban en un archivo adicional llamado: - - "Directorio de Bancos". El cual es útil para depuraciones y documentación.

En forma análoga con el punto 2.3., debe recalcar que el "Banco de Datos" es la estructura que relaciona a todos los módulos o subsistemas de SPIE, ya que el Banco de Datos puede considerarse como la implementación concreta del concepto de series de tiempo.

2.5. RELACION GRAFOS-MALER

Existe un subsistema de SPIE que está íntimamente ligado con GRAFOS, dicho subsistema es el Módulo Generador de Reportes (MALER).

Esto se debe inicialmente, a que ambas herramientas son aplicables para presentación de resultados. No es -- poco común que un informe financiero o económico presente - reporte numéricos y gráficas.

Debido a lo anterior, se tomarán ciertas consideraciones, que son:

- A) Es más cómodo para el usuario usar una misma -- sintaxis para resolver varios requerimientos. Por lo tanto la sintaxis del generador de reportes es - un subconjunto de la sintaxis empleada en GRAFOS. Así el usuario en vez de aprender una filosofía diferente, simplemente agregará nuevos elementos en su universo.

- B) Es muy probable que el usuario deseara mezclar - en una misma presentación uno o más reportes -- con gráficas que refuerzan el impacto de las cifras. Por otro lado, ya existía un generador de reportes, y no tenía caso que el nuevo sistema repitiera las - funciones de otro similar. En conclusión, se decidió que dentro de la sintaxis de GRAFOS, se permitira involucrar reportes ya elaborados con - - - MALER.

No obstante esta relación, ambos módulos pueden trabajar completamente independientes uno del otro. (Solo - conservando la relación necesaria entre los subsistemas de - SPIE a través de los " Bancos de Datos!")

Para terminar este punto es importante mencionar que la relación que guardan GRAFOS y MALER, enriquece las posibilidades y los alcances de ambos.

3. ESTRUCTURA DEL PAQUETE GRAFOS

El Generador de Gráficas GRAFOS está compuesto por un conjunto de Subrutinas acopladas entre sí, esta interacción se muestra en los Diagramas de Arbol de las figuras 3A, 3B, 3C. En las cuales cada nombre identifica un módulo objeto generado a partir de un Módulo Fuente, cabe hacer la aclaración que algunas de estas rutinas se auxilian de otras que proporciona el Sistema VERSAPLOT-07 y algunas propias del lenguaje FORTRAN, las cuales se encuentran en la biblioteca del Sistema Operativo, por lo cual no se representan en el Diagrama de Arbol.

En la implementación del Paquete GRAFOS, en la Computadora PDP 11/40, se utilizó un proceso llamado OVERLAY el cual es proporcionado por el constructor de tareas (TASK-BUILDER).

DIAGRAMA DE ARBOL DEL MODULO DE DEFINICION (DEF INE)

	ABRE DEFCAM	DEPURA DISTR I CONVER GRABA ERRDR HELP	
	GRABA VALF IL RENOM DEPURA	VALF IL	
		PARALF PARNUM	ERROR ERROR ESPAR LLENAR ERROR
		METRO	ERROR
		HELP	ESPAR ERROR
			NUMER
			TAB
DEF INE		SINTAX	ERROR NUMER NUMER NUMER
			MAS
			TAMANO PATRON LETRE
	AREAS		NUMER NUMER ERROR
			DEPURA TAMANC PATRON
		LETPOS	PSC ION LETRE ERROR
			ERROR
		DETDAT	ESPAR DEPURA ERROR
		GRABA	
	CATAL HELP ERROR COCL IR		

DIAGRAMA DE ARBOL DEL MODULO DE MODIFICACION (MODIF)

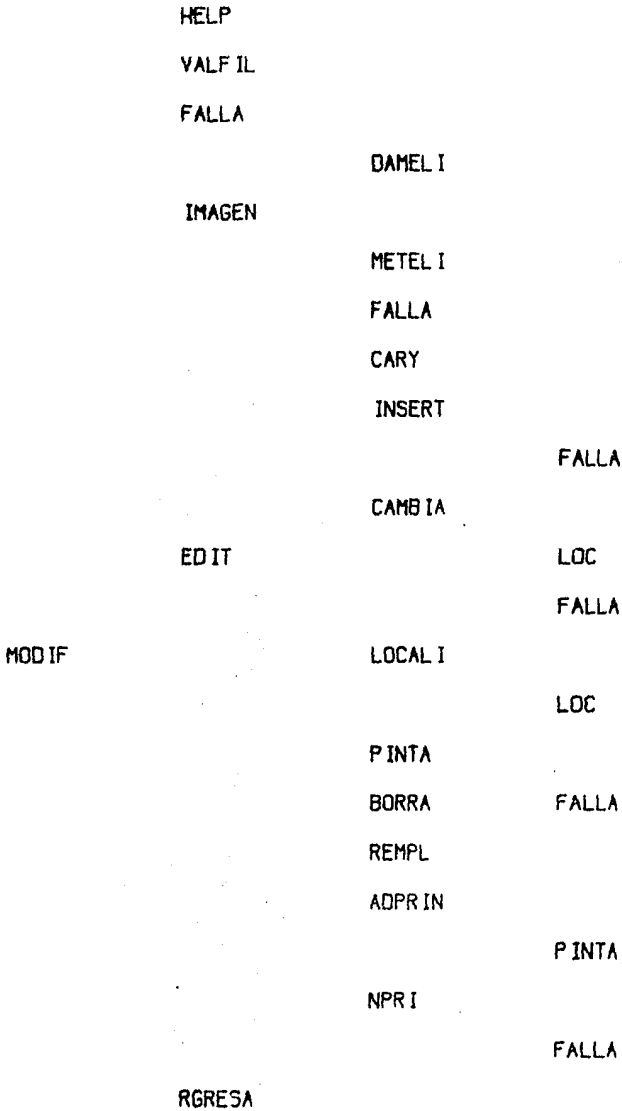


DIAGRAMA DE ARBOL DEL MÓDULO DE EJECUCION (EJECUB)

HELP			
VALF IL			
TESTCA	ALFANUM		
REPMLR	ERR		
	DAMEL I		
	DAME IN	DAMEL I	
		NOREAL	
P IENCA	SALEDC	RECPAR	DAMEL I
	RECPAR		
	IMPTEX		
	NOREAL		
	KEPTX		
	ERR		
	DAMEL I		
	RECPAR		
ACOTAC	IMPTEX		
	NOREAL		
	EJEPAT		
EJECU	DAMEL I		
	RECPAR	DAMEL I	
	TMNO		
	NOREAL		
	RECPAR	DAMEL I	
	PAT		
	NOREAL		
POSLET	RECPAR	DAMEL I	
	POS		
	NOREAL		
	LTR		
	DAMEL I		
	RECPAR	DAMEL I	
	VAC IA		
	NOREAL		
	CRETAB		
	PREPAR		
	ESCAL		
DOGRAF	EJET IM		
	EJEBS	ERR	
	POL I	SEGARR	
	HISTO		
	VERSA		
	ENVOL	ERR	
	IMPUL		
	CURVA		

3.1. ETAPAS DEL PAQUETE GRAFOS

En la generación de una gráfica el usuario utiliza un conjunto de instrucciones denominadas "Comandos", los cuales se almacenan en una estructura llamada "Descriptor". Una vez definido un descriptor se puede conservar en memoria secundaria (DISCO), para que el usuario pueda obtener una o -- varias gráficas ya definidas, sin tener necesidad de repetir -- los Comandos.

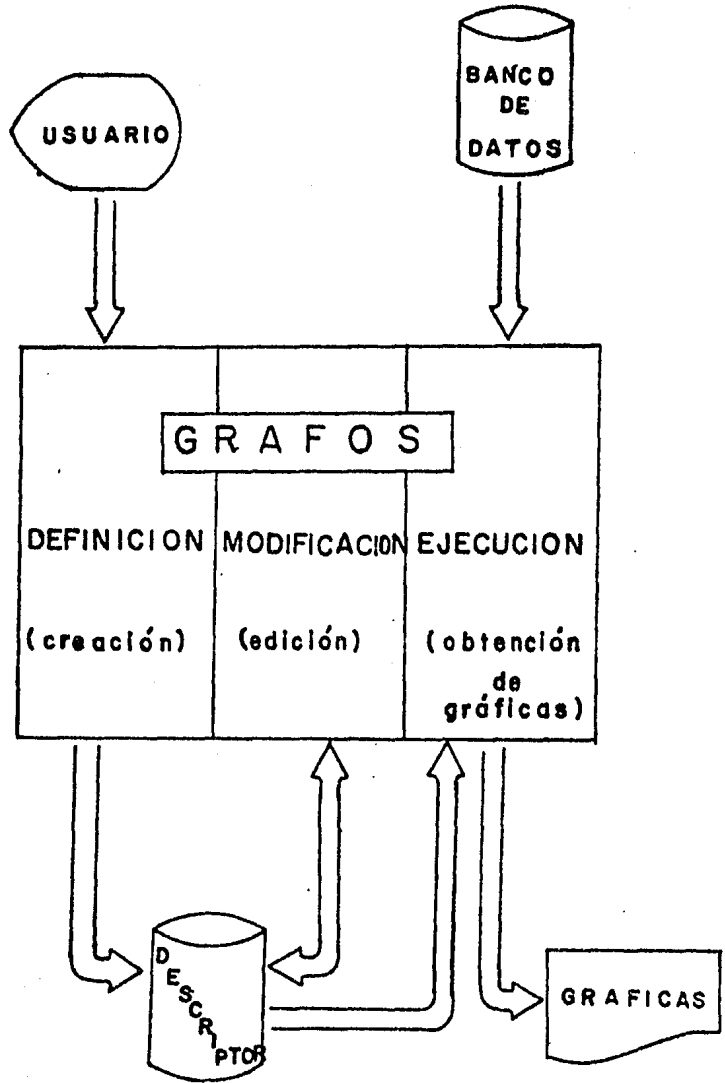
El Descriptor está contenido en un archivo del tipo- de acceso directo de longitud fija, en el cual su contenido real será variable, de acuerdo a la cantidad de comandos que el -- usuario proporcione.

El paquete GRAFOS está dividido en tres procesos principales que son:

DEFINICION DE GRAFICAS
MODIFICACION DE GRAFICAS
EJECUCION DE GRAFICAS

Esquemáticamente esto se muestra en la figura 3.1.A

Figura 3.1.A



3.1.1. DEFINICION DE GRAFICAS

En la definición de las gráficas el usuario proporciona paso a paso todos los comandos necesarios para elaborar su gráfica o gráficas.

El Paquete GRAFOS al momento de la DEFINICION-efectúa las siguientes funciones:

- Ubica al usuario indicando mediante mensajes el área que está definiendo.
- Explicación (opcional) de la sintaxis de los Comandos, así como de las características de cada área.
- Revisión de la sintaxis de cada comando proporcionado por el usuario, en base a una sintaxis definida en el manual de usuario, en el caso de existir error se envía un mensaje especificando el tipo de error, a continuación el usuario tendrá que repetir la línea errónea.
- Almacenamiento de todos los comandos en el Descriptor.
- Detectar en base a una bandera ubicada en el Descriptor si el chequeo de la sintaxis se efectúa al momento de que el usuario está tecleando los comandos o si éste chequeo de sintaxis se efectúa después del proceso de modificación para lo cual los posibles errores y su especificación se enviarán a un archivo de escritura para ser recuperado en forma de un listado.

3.1.2. MODIFICACION DE GRAFICAS

Esta parte del Paquete Generador de Gráficas -- (GRAFOS) es también llamado EDITOR y al proceso de efectuar la modificación se le llama EDITAR.

Al momento que un usuario modifica una o unas -- gráficas definidas se altera el DESCRIPTOR creado en el -- proceso de DEFINICIÓN, esto se logra con un conjunto de -- instrucciones denominadas DIRECTIVAS.

Las funciones del proceso de MODIFICACION - - - (EDICION) son:

- Con las directivas proporcionadas por el usuario modificar el archivo DESCRIPTOR
- Ubicar al usuario indicando mediante mensajes que parte o área está modificando.
- Explicación (opcional) de las directivas de -- EDITOR
- Invocar al proceso de DEFINICION para que -- efectúe el chequeo de la sintaxis de lo modificado.

3.1.3. EJECUCION DE GRAFICAS

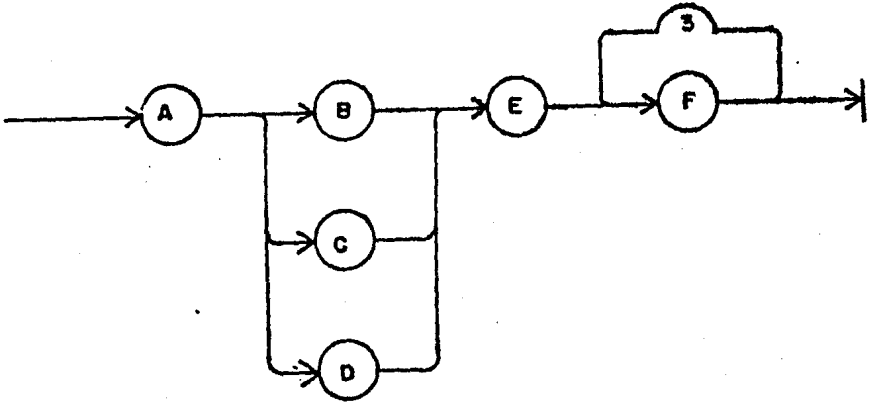
En este proceso se obtiene la gráfica o gráficas ésto se logra mediante la ejecución de cada comando almacenado en el archivo DESCRIPTOR, además se auxilia el proceso de ejecución de las rutinas VERSAPLOT-07 residentes en la biblioteca del sistema Operativo, así como de los bancos de datos en los cuales se encuentran los valores (en series de tiempo) a graficar.

3.2. LENGUAJE DE COMANDOS

En la definición de las gráficas no se utilizó un pro-ceso interactivo (pregunta respuesta) dado que las instrucciones requeridas son varias, haciendo fastidioso este método, - por lo cual, se empleo en el paquete GRAFOS, en la etapa - de DEFINICION, una agrupación de instrucciones que se denominan COMANDOS, permitiendo fluidez en la definición de - las gráficas. Como el objetivo es hacer un lenguaje versátil estos comandos permiten omitir o repetir algunas opciones. Así como agrupar varios comandos por línea, a su - vez es posible agrupar líneas en módulos, los cuales también cuentan con opciones de omitir algunos de ellos.

En GRAFOS la descripción de dichas agrupaciones se hace mediante DIAGRAMAS DE SINTAXIS, mejor conocidas como: DIAGRAMAS DE FERROCARRILERO, los cuales - tienen una entrada y una salida y el sentido de recorrido es unidireccional, a continuación se muestra un ejemplo de este diagrama.

Figura 3.2.A



En este ejemplo se observan las siguientes posibilidades de recorrido, pasar forzosamente por el elemento "A", - para llegar al elemento "E", se puede llegar a través del elemento "B" o "C" o "D", para llegar al elemento "F" es forzoso pasar por "E", ahora para salir de este diagrama se logra pasando por "F" de 1 a 4 veces

3.3. ESTRUCTURA DE DATOS PRINCIPALES

La eficiencia de un sistema en cuanto a su uso y re cur so s, se debe a dos factores, T i e m p o de R e s p u e s t a y M e m o r i a U t i l i z a d a.

Para el Tiempo de Respuesta, implica por un lado - tiempos cortos de uso de procesador y por otro lado una mejor respuesta al usuario, sobre todo en sistemas de tiempo com-- partido.

Para la Memoria Utilizada, implica menos recursos de memoria utilizados por el sistema. Por lo tanto, la eficiencia está en la utilización mínima de los recursos; procesador y memoria, en un tiempo de respuesta satisfactorio para el - usuario.

Es por lo anteriormente expuesto que se diseñó - - GRAFOS utilizando las estructuras de datos que a continua-- ción se describen.

3.3.1. ARCHIVO DESCRIPTOR DE LA GRAFICA

En este archivo se almacenan todos los comandos y directivas (sin errores) después del chequeo sintáctico, se crea un Archivo Descriptor cada vez que se define una nueva área, subáreas y gráficas permanece presente a lo largo de esta etapa, el objeto de crear un archivo por cada definición es el conservar, modificar y ejecutar (cuantas veces lo requiera) alguna área gráfica, el archivo descriptor tiene la siguiente estructura:

TIPO DE ACCESO:	Directo
LONGITUD DEL ARCHIVO:	300 Registros
LONGITUD DEL REGISTRO:	33 Palabras
UNIDAD ASIGNADA:	La No. 9

A continuación se muestra la Figura 3.3.A

FIGURA 3.3.A ESQUEMA DEL DESCRIPTOR PARA GRAFOS

Nreg.

1	TIPCAM: G, M, G
2	XMIN \emptyset , XMAY \emptyset , YMIN \emptyset , YMAX \emptyset , Y
3	TAM, NUMLIN, NOCHAR, XMIN, XMAX, YMIN, YMAX, X, Y, GRUESO, APUNT(8,2)
12	TAM, NUMLIN, NOCHAR, XMIN, XMAX, YMIN, X, Y, GRUESO, APUNT (8,2)
13	CAMPO \emptyset = 30, 40? CAMPO 1 =10, 20: LIN = 60? CAMPO 2 = 20, 20-40: LIN = 6.
14	0; REPOR?

3.3.2. ARCHIVO DIRECTORIO DE GRAFICAS

En este archivo se almacenan todos los nombres de los ARCHIVOS DESCRIPTORES definidos, el archivo directorio se mantiene siempre actualizado por la etapa de modificación, este archivo es ocupado en la etapa de ejecución para verificar que exista el ARCHIVO DESCRIPTOR que contiene los comandos y directivas para graficar.

El ARCHIVO DIRECTORIO se encuentra en una cuenta única para todos los usuarios, su estructura es la siguiente:

TIPO DE ACCESO: Secuencial

UNIDAD ASIGNADA: La No. 8

NOMBRE		NUMERO DE CUENTA			
1	8	9	11	12	14

3.3.3. ARCHIVOS DE MENSAJES

Se tienen dos ARCHIVOS DE MENSAJES ambos de tipo de acceso directo; uno de ellos es de mensajes de error que se cometan durante las etapas de DEFINICION Y MODIFICACION, esta Modificación y Definición se ejecutan sobre el archivo DESCRIPTOR; el otro ARCHIVO DE MENSAJES es el de Orientación al Usuario, en este archivo se en encuentran contenidos mensajes indicadores de la etapa en que está trabajando, así como las capacidades de cada etapa.

A continuación se detalla cada uno de estos archivos:

ARCHIVO DE MENSAJES DE ERROR (ERROR.DAT)

TIPO DE ACCESO:	Directo
LONGITUD DEL ARCHIVO	200 Registros
ANCHO DE REGISTRO:	60 Palabras
UNIDAD ASIGNADA:	La No. 1

El número de error estará asociado con el número de registro en el cual se encuentra almacenado.

ARCHIVO DE MENSAJES DE ORIENTACION (HELPGRAF. DAT)

TIPO DE ACCESO:	Directo
LONGITUD DEL ARCHIVO	604 Registros
ANCHO DEL REGISTRO:	35 Palabras
UNIDAD ASIGNADA:	La No. 2

Este archivo se encuentra organizado en la forma siguiente:

En el registro No. 1 están grabados el número de registro donde inicia cada mensaje correspondiente a la descripción general de cada módulo.

En el registro No. 2 están grabados el número de registro donde inicia cada mensaje correspondiente a la descripción de comandos. (etapa de definición).

En el registro No. 3 están grabados el número de registro donde inicia cada mensaje correspondiente a la descripción de directivas (etapa de modificación).

Los mensajes se graban a partir del registro No. 5, la forma de acceso a algún mensaje es bajo la ecuación:

$$\text{LIMITE SUPERIOR} = \text{APUNTADOR (STATUS)}$$

$$\text{LIMITE INFERIOR} = \text{APUNTADOR (STATUS + 1) - 1}$$

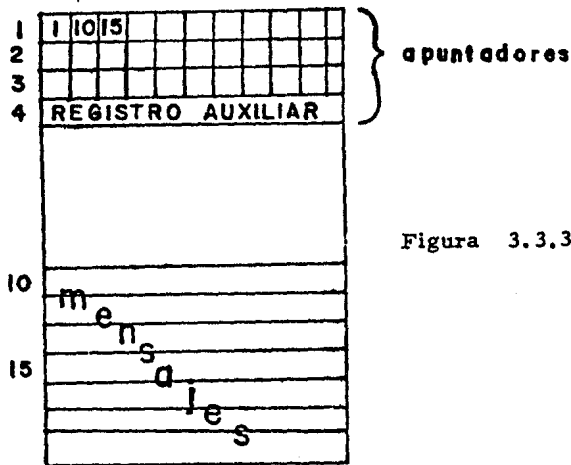


Figura 3.3.3. A

Ejemplo:

$$\text{LIMITE SUPERIOR} = \text{APUNTADOR (2)} = 10$$

$$\text{LIMITE INFERIOR} = \text{APUNTADOR (2+1) - 1} = 15 - 1 = 14$$

el mensaje número dos corresponde a la descripción general del 2do. módulo, está comprendido entre los registros del 10 al 14.

4. COMPONENTES DE UNA AREA GRAFICA

En ocasiones es necesario en una área tener más de una gráfica o bien tener una gráfica y un reporte numérico. En base a estos requerimientos es que el paquete -- GRAFOS fue diseñado con la capacidad de que el usuario pueda hacer la estructuración adecuada a sus necesidades, para esto, GRAFOS permite al usuario:

- DEFINIR HASTA UN METRO CUADRADO COMO AREA O CAMPO PRINCIPAL
- DEFINIR HASTA 10 SUBAREAS O SUBCAMPOS DENTRO DE SU AREA ---
PRINCIPAL.

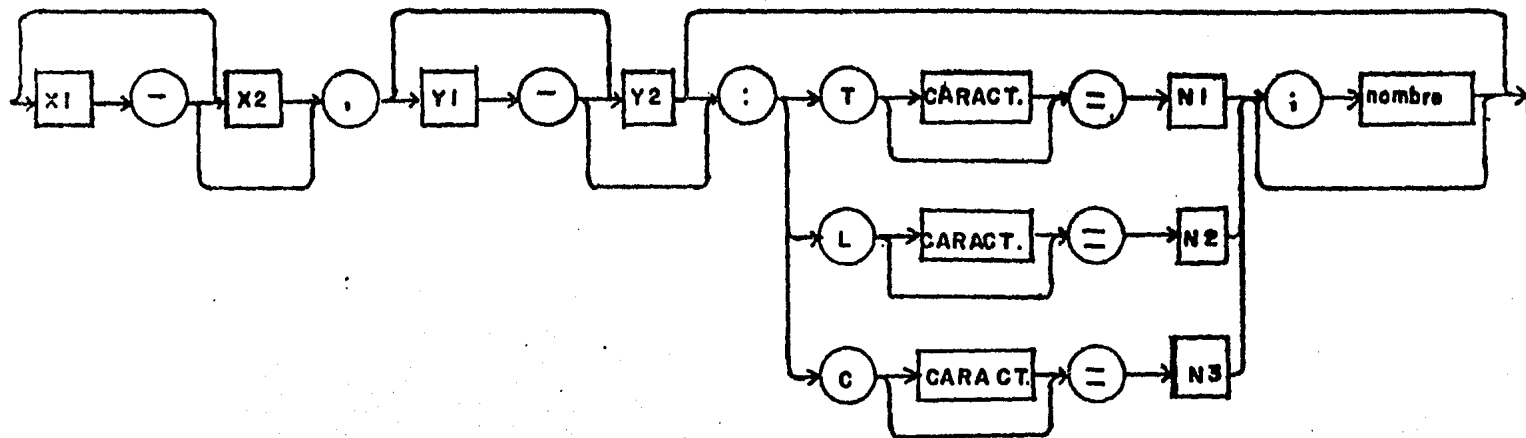
Cabe hacer la aclaración que es posible usar una subárea gráfica para imprimir sólo textos, lo cual en ocasiones es muy útil.

En cada subárea se tiene la posibilidad de definir alguna de las siguientes opciones:

- TAMAÑO DE CARACTER
- NUMERO DE LINEAS EN LA SUBAREA
- CARACTERES POR LINEA

al proporcionar alguna de estas opciones automáticamente se calculan las otras dos, esto permite al usuario, dependiendo de sus necesidades, ajustar en la subárea gráfica lo que para él tenga más relevancia, la forma de definir el área --- principal y sus subáreas se muestra en el siguiente diagrama de sintáxis:

DEFINICION DE CAMPOS



DONDE: X1 ABSISA MINIMA
X2 ABSISA MAXIMA
Y1 ORDENADA MINIMA
Y2 ORDENADA MAXIMA
N1 NUMERO ENTERO INDICA TAMAÑO DE CARACTER (POR CATALOGOS).
N2 NUMERO ENTERO INDICA NUMERO DE LINEAS QUE SE TENDRAN POR CAMPO.
N3 NUMERO DE CARACTERES QUE TENDRA CADA LINEA - DEL CAMPO.
NOMBRE CADENA DE CARACTERES ALFANUMERICOS QUE SE DEFINE EN EL CASO DE REPORTES NUMERICOS.

4.1 ZONAS DE UNA SUBAREA GRAFICA

Es importante en la presentación de cualquier fenómeno econométrico representado en forma gráfica, tener en el mismo cuerpo de la presentación textos que faciliten la interpretación de dicha gráfica, además debe contar con presentación aceptable, es por esto que GRAFOS cuenta dentro de cada subárea gráfica con las siguientes zonas.

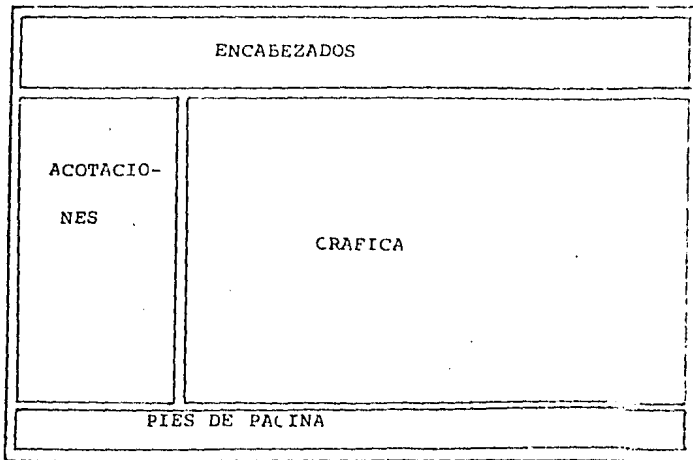
- ENCABEZADOS
- ACOTACIONES
- PIES DE PAGINA
- GRAFICA

Para las tres primeras zonas se tiene una relación directa con las especificaciones dadas al definir cada subárea gráfica, esto es en cuanto al número de líneas, tamaño de carácter y de caracteres por línea, la última zona es independiente.

Los Encabezados, Acotaciones y Pies de Página forman el marco para la gráfica (Figura 4.2.A), estas cuatro zonas facilitan la presentación de un fenómeno económico en forma objetiva y profesional.

Figura 4.2.A

AREAS DE UNA GRAFICA



4.2 ELEMENTOS DE UNA SUBAREA GRAFICA

Como se mencionó anteriormente una subárea gráfica está compuesta por el marco y la gráfica, para la definición de éstos se cuenta con los siguientes elementos:

- PARAMETROS (ALFABETICOS Y NUMERICOS)
- ENCABEZADOS
- ACOTACIONES
- PIES DE PAGINA
- LETREROS POSICIONADOS
- DETERMINACION DE DATOS

La estructura de cada elemento se detalla en las siguientes páginas.

4.2.1 PARAMETROS

Al efectuar el estudio sobre requerimientos en graficación se encontró que en ocasiones existen algunos - datos numéricos y cadenas alfanuméricas que se repiten con cierta frecuencia en el contexto que define a las gráficas, además se encontró que en ocasiones el usuario desea aprovechar las gráficas ya diseñadas y que sólo requiere modificar algunos datos numéricos o cadenas alfanuméricas. Es posible hacer estas modificaciones ya que como se recordará GRAFOS cuenta con una etapa de modificación, pero hacer estos cambios implica ir buscando en el descriptor y cambiar los datos o cadenas alfanuméricas que desea modificar, lo que es un tanto laborioso, por lo que se pensó en incluir en la descripción o generación de la gráfica o gráficas parámetros alfabéticos y numéricos y estos ocuparían una zona fija dentro del descriptor, estos parámetros ofrecen la posibilidad de ser definidos una sola vez e invocados las veces que sea necesario lo cual permite que las gráficas sean modificadas de manera rápida y con menor grado de dificultad.

Cada parámetro se identificará por un nombre o cadena de caracteres alfanumérica compuesta de uno a seis caracteres, el contenido del parámetro se encuentra separado por un símbolo de igual.

A continuación se muestran los diagramas de sintaxis de la definición de dichos parámetros.

Figura: 4.2.1.A

PARAMETRO ALFABETICO

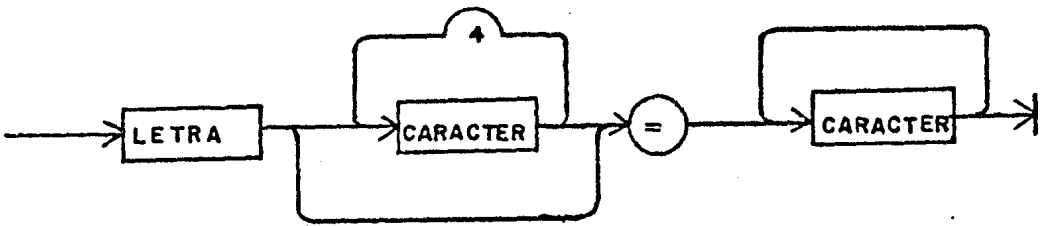
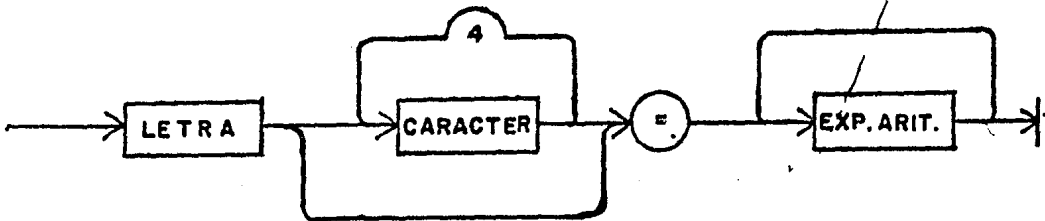


Figura: 4.2.1.B

PARAMETRO NUMERICO



4.2.2 ENCABEZADOS

Este elemento de la Subárea o Area Gráfica consta de cadenas de caracteres alfanuméricos asociados con comandos, estos letreros son colocados en la parte superior del área o subárea gráfica y no requiere de coordenadas de colocación ya que se inician en este lugar y su limitante es la parte inferior del área o subárea gráfica, los comandos aceptados en -- los encabezados son:

- TAMAÑO:** Indica el tamaño de caracter dado por catálogo así como el grueso con que se imprimirá el letrero asociado.
- TABULA:** Indica en qué posición se inicia el letrero.
- DERECHA:** El letrero asociado es justificado a la derecha de la línea.
- IZQUIERDA:** El letrero asociado es justificado a la izquierda de la línea.

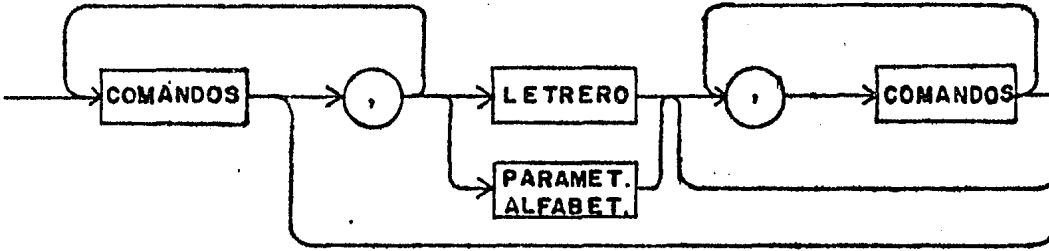
- CENTRADO:** El letrero asociado aparece centrado en la línea.
- BLANCO:** Indica los "n" renglones a dejar en blanco.
- SUBRAYA:** Subraya
- REPITE:** Repite "n" veces un caracter o conjunto de - caracteres.
- +** Indicador de que es continuación de la línea anterior.

Estos comandos al momento de ser definidos, se tiene la opción de teclear toda la cadena que los identifica o sólo el primer caracter.

Los diagramas de definición de un encabezado y de cada comando se muestra en los siguientes diagramas:

Figura: 4.2.2.A

ENCABEZADOS



TABULA



DÉRECHA



IZQUIERDA

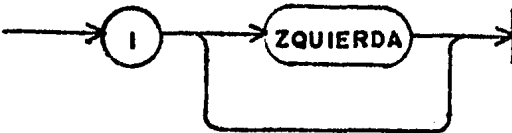
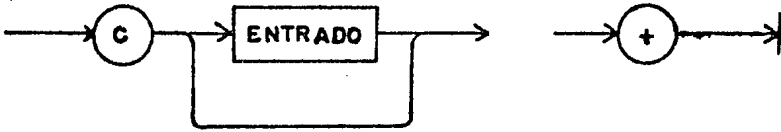


Figura: 4.2.2.B

CENTRADO

MAS



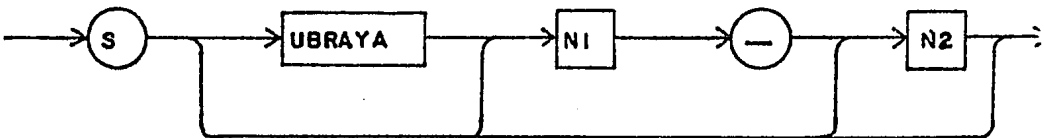
BLANCO



LETRERO



SUBRAYADO



REPITE

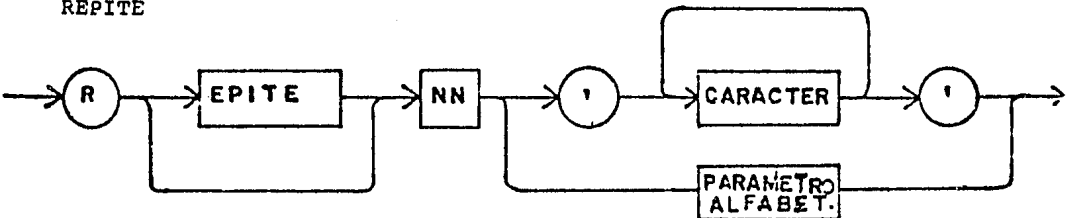
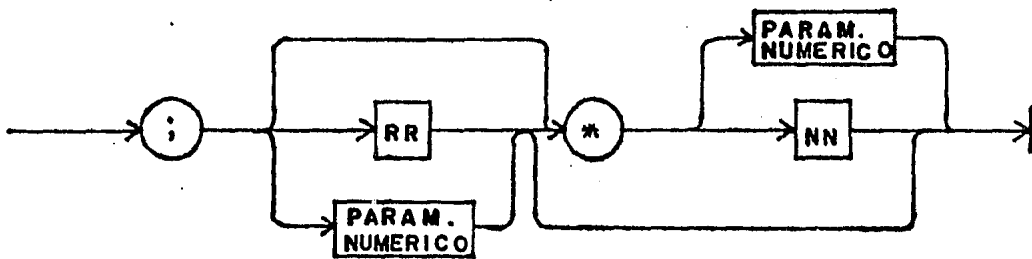


Figura: 4.2.2.C

TAMAÑO



4.2.3 ACOTACIONES

Las Acotaciones son de utilidad para identificar por medio de un letrero y un patrón gráfico a cada una de -- las gráficas. Este elemento de la subárea o área consta de cadenas alfanuméricas o letreros asociados con el comando de nominado Patrón y otro denominado Tamaño, este letrero y su patrón son colocados a continuación de la última línea definida como encabezado y está justificado a la izquierda de la subárea o área gráfica, no requiere de coordenadas de colocación, tiene como límites a la derecha el ancho máximo del -- área o subárea y hacia abajo la parte inferior de la misma.

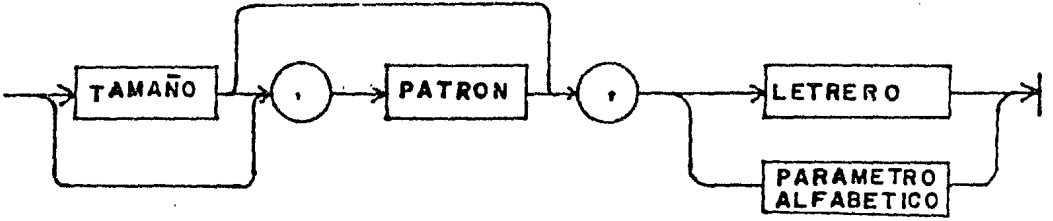
En detalle los comandos aceptados en acotaciones son:

TAMAÑO: Indica el tamaño de caracter que se empleará para su impresión (dada por catálogo), así como el grueso que se empleará.

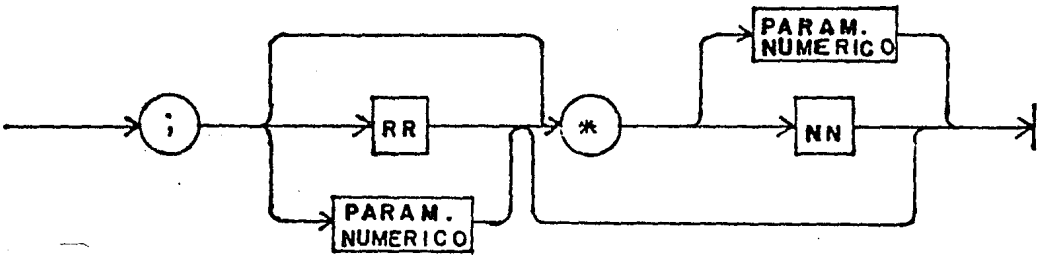
PATRON: Indica el tipo de gráfica que se asociará -
con un patrón gráfico, ya que se cuenta con
seis tipos de gráficas en los cuales cada -
uno de ellos cuenta hasta con ocho patrones
gráficos.

La definición de acotaciones así como sus comandos se mues-
tran enseguida.

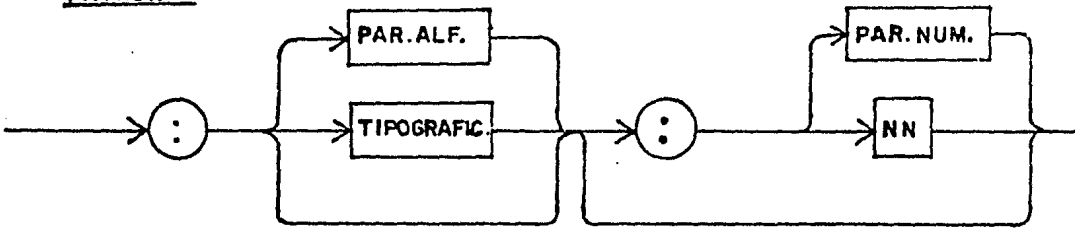
Figura: 4.2.3.A



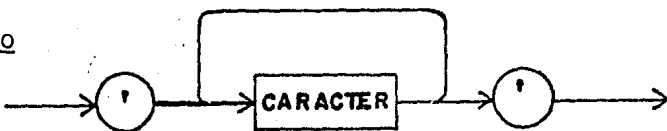
TAMAÑO



PATRON



LETRERO



4.2.4 PIES DE PAGINA

Este elemento del área o subárea gráfica consta de cadenas de caracteres alfanuméricos asociados con comandos, estos letreros son colocados tomando como referencia - el límite inferior del área o subárea gráfica no requiere - de coordenadas de colocación, tiene como limitante el espacio dejado entre la última línea de acotaciones y el límite inferior del área o subárea gráfica.

Los comandos aceptados en pies de página así como sus diagramas son los mismos que acepta encabezados.

Hasta aquí se ha descrito cada uno de los elementos que conforman el marco de la gráfica, el espacio en donde se coloca la gráfica es calculado en base a los espacios ocupados por cada elemento del marco.

4.2.5. LETREROS POSICIONADOS

Al hacer el estudio y análisis de los requerimientos más frecuentes en una gráfica se detecta la necesidad de contar con alguna flexibilidad en cuanto a letreros se refiere, esto es que no fueran tan rígidos, ya que en ocasiones es necesario imprimir un letrero con un ángulo de giro y una posición determinada, es por esto que se implementó en GRAFOS un elemento que hace posible este requerimiento y lo denominamos LETRERO POSICIONADO, es importante hacer notar que este letrero puede tener giros hasta de ---360°, esto es posible ya que en las rutinas de VERSAPLOT-07 se cuenta con una rutina que efectúa giros. LETRERO POSICIONADO cuenta además con tres comandos asociados que son:

TAMAÑO: Indica el tamaño de carácter que se empleará para la impresión del letrero, así como el grueso que se empleará.

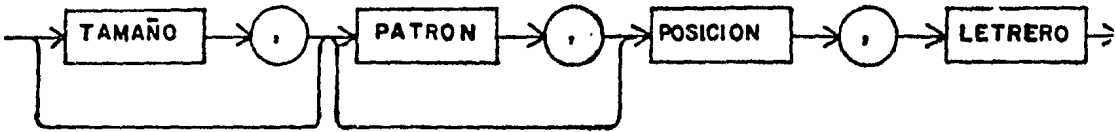
PATRON: Indica el tipo de gráficas que se asociará con un patrón gráfico, esto permite que en ocasiones el letrero posicionado sea ocupado como -- acotación.

POSICION: Indica las coordenadas, (dentro del área o -- subárea gráfica) en donde será colocado el letrero así como su ángulo de giro.

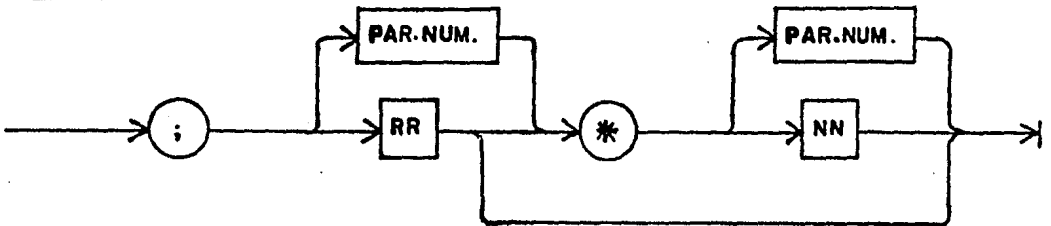
La definición de un letrero posicionado así como sus comandos asociados se muestran a continuación.

Figura: 4.2.5.A

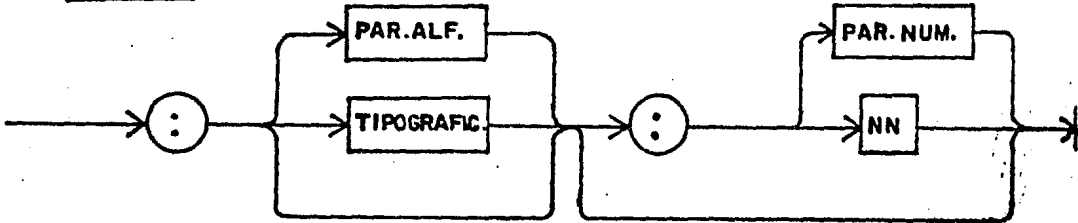
LETRERO POSICIONADO



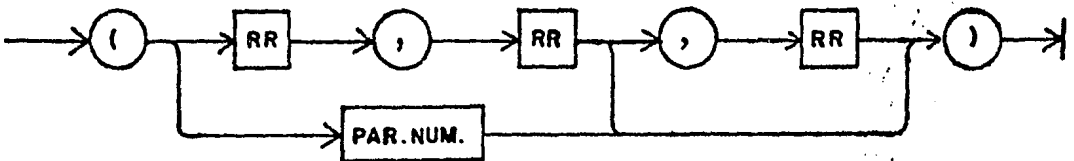
TAMAÑO



PATRON



POSICION



4.2.6 DETERMINACION DE DATOS

En este punto se tratará la parte medular de este trabajo, que es la definición de la gráfica o gráficas - ya que pueden ser hasta ocho diferentes tipos de gráficas - cuyos datos de elaboración pueden estar tomados de un sólo Banco de Datos o de ocho diferentes Bancos de Datos, con diferentes series de tiempo (Véase Bancos de Datos y Series de Tiempo en el capítulo uno), es necesario hacer notar que la gráfica que aquí se defina será colocada en el espacio - dejado por los elementos del marco, esto es, se ajustará esta gráfica al espacio disponible ya sea amplificando la gráfica o efectuando una reducción de esta.

En la determinación de datos se cuenta con los siguientes - elementos o datos:

BANCO: Nombre del banco donde se tomará la información.

NSI: Número de Serie Inicial a partir de la cual
 se tomará la información.

NSF: Número de Serie Final, hasta donde se tomará
 dicha información.

AI: Año Inicial a partir del cual se graficará.

PI: Período Inicial a partir del cual se grafica
 rá.

AF: Año Final hasta donde se graficará.

PF: Período Final hasta donde se graficará.

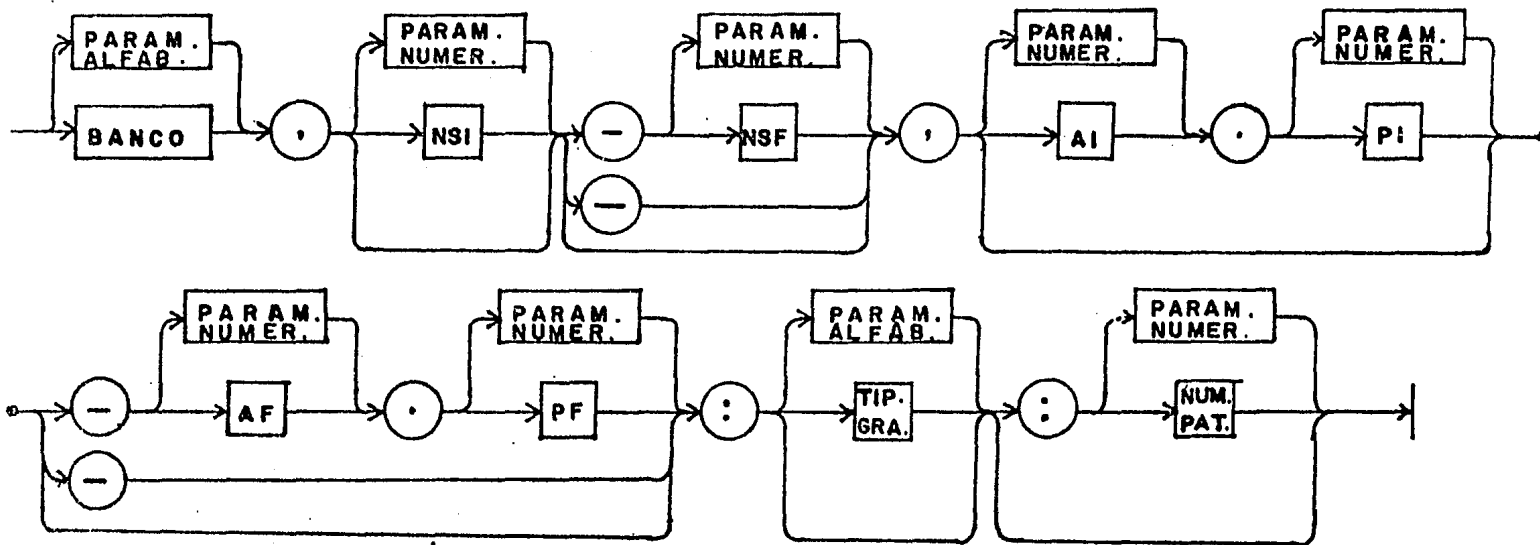
TIPO DE
GRAFICA: Tipo de Gráfica que se ocupará.

No. PATRON: Es el número de patrón gráfico que se emplea
 rá para el tipo de gráfica definida.

A continuación se muestra el diagrama de sintáxis de la de
terminación de datos.

Figura: 4.2.6.A

DETERMINACION DE DATOS



5. DESCRIPCION DE PROGRAMAS

En este capítulo se incluye la definición específica de los cinco Programas que comprenden el Sistema completo. Así como todas y cada una de las Rutinas o Subprogramas - que componen a cada una de ellas.

En el caso de los dos primeros Programas por ser muy simples, solo se tiene un solo Módulo por cada Programa. (Cargadores de Archivos). Sin embargo, los tres siguientes -- Programas sí cuentan con un número considerable de Subrutinas, por lo tanto, se recomienda consultar la Sección 3 en caso de alguna duda con respecto a la interrelación entre Módulos.

En todos los Módulos que aparecen a continuación existen los siguientes elementos constantes:

- NOMBRE DEL MODULO
- FUNCION QUE REALIZA
- ARCHIVOS QUE ACCESA
- MODULOS INVOCADOS
- MODULOS QUE LO INVOCAN

5.1 CARGADOR DE INSTRUCTIVO

PROGRAM LOADHE.FTN

FUNCION:

Transferir la información de un archivo secuencial a uno de tipo directo.

Inicialmente en el archivo secuencial están contenidas 3 áreas - (identificadas por dos arrobas). En la primera se tiene la información general de cada módulo que ocupa el usuario del paquete - GRAFOS.

En la segunda área se tiene la información con los comandos que se ocupan en la etapa de definición.

En la tercera área se tiene la información con las directivas -- que se ocupan en la etapa de edición.

El archivo de tipo secuencial se denomina como AUXHELGRF.DAT y - el de tipo directo como HELPGRA.DAT, en este último archivo en - los 3 primeros registros se graban los apuntadores de cada área y los límites de cada mensaje, ésto es, en el primer registro se guardan las direcciones de la información general de cada módulo, en el segundo registro se guardan las direcciones de los coman-- dos y en el tercer registro se guardan las direcciones de las di-- rectivas.

La información en sí queda grabada a partir del registro número 5 en adelante.

En este programa se puede considerar como entrada el archivo de acceso secuencial AUXHELGRF y como salida el archivo de acceso directo HELPGRAF.

OBSERVACIONES:

Este programa no pertenece al sistema, es un auxiliar.

ARCHIVOS:

AUXHELGRF.DAT (Acceso secuencial)

HELPGRAF.DAT (Acceso directo, 604 registros con 35 palabras -
cada uno)

5.2 CARGADOR DE MENSAJES DE ERRORES

PROGRAM LOAD.FTN

FUNCION:

Cargar o modificar en un archivo, los mensajes de posibles errores sintácticos que se cometan en la etapa de - DEFINICION, cada error está contenido en un registro de dos- vectores de 60 caracteres cada uno.

El número de error está asociado al número de registro en el cual se encuentra almacenado.

El archivo donde se almacenan los errores es de 200 registros con 60 palabras cada uno y es de tipo acceso directo.

OBSERVACIONES:

Este programa, es un auxiliar del Sistema.

ARCHIVOS:

ERROR.DAT (En la Unidad No. 1)

INVOCADA POR:

INVOCA A:

5.3 DEFINICION DE GRAFICAS

PROGRAMA PRINCIPAL: DEFINE

FUNCION:

Controla los procesos necesarios para "CREAR" una gráfica. Esto es; crear un archivo que contiene un conjunto de instrucciones ordenadas que posteriormente serán ejecutadas para obtener una gráfica. La creación de una gráfica se divide en:

"Definición de Campos" y Definición de Areas" para dichos campos. Antes de crear una gráfica, es necesario saber si dicha gráfica no fue creada con anterioridad, para lo cual se utiliza una subrutina que la verifica.

También en el programa principal se abre el archivo de trabajo que genéricamente es llamado archivo "DESCRIP--TOR".

Este programa se usa para definir una gráfica nueva, pero también se emplea después de modificar alguna. Por lo tanto, es necesario identificar cual de los dos casos se presenta.

Cuando se trata de modificar un "DESCRIPTOR" ya creado, existen algunas funciones adicionales que deben realizarse; estas son: obtener un listado del archivo y renombrar algún "DESCRIPTOR".

ARCHIVOS

*.GRF: DESCRIPTOR de la gráfica

TEMPGRAF.TMP: Imagen del DESCRIPTOR

TEMPO.TMP

INVOCA A:

ABRE

VALFIL

AREAS

CATAL

HELP

AYUDA

ERROR

COCLIR

RENOM

LISDES

INVOCADA POR:

La tarea MODIFI

OBSERVACIONES:

Existen algunas rutinas del Sistema Operativo que se emplean,
estas son:

CLREF

REQUES

SUBROUTINA: ABRE,FTN

FUNCION:

Asignar los archivos que estarán presentes durante toda la etapa de definición, básicamente son dos los archivos: el de Información de los tipos de errores que sintácticamente se cometan y el de Mensajes de ayuda u orientación para el usuario.

ARCHIVOS:

ERROR.DAT	UNIDAD No. 1
	ACCESO DIRECTO
	LONG.REG. 60 PALABRAS
	LONG.ARCH. 200 REGISTROS

HELPGRAF.DAT	UNIDAD No. 2
	ACCESO DIRECTO
	LONG.REG. 35 PALABRAS
	LONG.ARCH. 604 REGISTROS

INVOCADO POR:

DEFINE

INVOCA A:

Ninguna

SUBROUTINA DEFCAM.FTN

FUNCION:

Capturar una cadena de datos con los cuales:

- Se define el área de las ventanas para gráficas o reportes.
- Se define una de las tres siguientes llaves; tamaño de carácter asociado a la ventana, o el número de renglones asociado a la ventana, o el número de caracteres que se desee en cada línea.
- Se define un posible nombre de un reporte. Esta información es limpiada de posibles espacios blancos intermedios con la ayuda de DEPURA, se checa si al inicio de la cadena se solicitó ayuda (HELP), valida los datos de la ventana y los checa sintácticamente, se verifica que no exista traslape entre las ventanas que se definan, ésto se logra con el auxilio de la rutina - - DISTRI.
- Se checa sintácticamente y valida la llave que se proporcione (tamaño, líneas, caracteres), con cualquiera de ellas son calculadas dos restantes y se informa al usuario.
- Se checa la existencia de un nombre de reporte, en caso de - - existir se graba una M en el vector TIPCAM en la posición correspondiente a esa ventana, en caso de no existir nombre de - reporte se asume que se trata de una ventana para gráfica en -

cuyo caso se graba una G en el vector TIPCAM.

Una vez hecho el chequeo sintáctico y validación de la cadena capturada, ésta es grabada en un archivo de trabajo,

PARAMETROS:

- COMMON/CAMPO/TABCAM: (Real) (4,11) tabla con áreas de trabajo, no es ocupada aquí.
- COMMON/GRAB/A PUNT: (Entera) apuntador a las áreas de trabajo, no se ocupa aquí.
- TIPCAM: (Lógica) (11) vector que contiene el tipo de ventana - (gráfica o reporte) asociada (S).
- NR: (Entera) número de renglón para el área de trabajo, no es ocupada aquí.
- NCOL: (Entera) número de columna para el área de trabajo, no es ocupada aquí.
- IN: (Entera) número de unidad para lectura de datos. (E)
- KONTER: (Entera) contador de invocaciones a la rutina de error. (S)
- NOUT: (Entera) número de unidad donde se graban los mensajes - (en archivo o en pantalla) (E/S)

Los valores correspondientes para cada campo, se calculan en base a las siguientes relaciones:

$$\text{TAMAÑO} = \text{DISTY}/(\text{NUMLIN} * (1 + 5/7))$$

$$\text{NOCAR} = \text{DISTX}/\text{TAMAÑO}$$

$$\text{NOLIN} = \text{DISTY}/(\text{TAMAÑO} * (1 + 5/7))$$

Por ejemplo, si el usuario solicita 20 líneas en un campo de 5 X 6 cm., entonces, el tamaño y el número de caracteres por cada línea serán:

$$\text{TAMAÑO} = 5/(20 * (1 + 5/7))$$

$$\text{TAMAÑO} = 0.14 \text{ cm.}$$

$$\text{NOCAR} = 6/0.14$$

$$\text{NOCAR} = 42$$

OBSERVACIONES:

La variable IN además de indicar el número de unidad en que se va a leer, indica de si la ejecución de esta rutina es - efectuada antes o después de la etapa de modificación.

Cuando esta rutina es efectuada después de la etapa de modificación, los posibles mensajes de error que se tengan en la validación y chequeo sintáctico en las cadenas de información, son grabados en un archivo con la ayuda de la rutina - ERROR y no se envían a pantalla como en el caso de captura manual.

Todas las conversiones que son necesarias ejecutar sobre la cadena, de caracteres alfanuméricos a números reales se lo - gra mediante la rutina CONVER.

ARCHIVOS:

*.GRF (Archivo de Trabajo)

INVOCADA POR:

DEFINE

INVOCA A:

DEPURA, HELP, CONVER, ERROR, DISTRI,
GRAEA.

SUBROUTINA: DEPURA

FUNCION:

Se encarga de eliminar caracteres en blanco que no estén dentro de algún letrero. Esto significa, como su nombre lo indica, que la rutina depura cada línea tecleada por el usuario al momento de la definición.

PARAMETROS:

CADENA: Arreglo de caracteres que contiene la línea tecleada por el usuario (E/S).

ARCHIVOS:

Ninguno

INVOCA A:

Ninguna

INVOCADA POR:

SINTAX

DEFCAM

SUBROUTINA: DISTRI.FTN

FUNCION:

 Checar la correcta distribución de los campos o áreas definidas como marco para una gráfica o reporte. Esta rutina recibe en un parámetro las coordenadas X,Y, de cada uno de los elementos (campos o subcampos) que serán grabados en una tabla cuando estén correctamente definidos, en dicha tabla están grabadas las coordenadas de estos elementos, esta rutina es invocada cada vez que se define un campo o subcampo y checa que el nuevo elemento; no salga de los límites del campo o área principal, que no exista traslape con algún elemento (subcampo o subárea) de los ya existentes en la tabla, cuando ocurre alguno de los dos casos mencionados se enciende una bandera de error que ocupará la rutina que invocó a ésta y se enviará un mensaje notificando el tipo de error cometido al momento de la definición.

PARAMETROS:

COMMON/CAMPO/TABCAM Tabla que contiene las coordenadas del
(REAL) (4,11) área y subáreas (E/S).

VECAMP (REAL) 4 Arreglo que contiene las coordenadas del
campo a analizar (E).

MALO (LOGICAL) Bandera que indica a la rutina invocada
ra la existencia de un error de definición (S).

NC (ENTERA) Contiene el valor correspondiente al número
de campo o subcampo (E).

ARCHIVOS:

Ninguno

INVOCADO POR:

DEFCAM

INVOCA A:

Ninguna

SUBROUTINA: CONVER.FTN

FUNCION:

Convertir algunas partes de la cadena de información de datos alfanuméricos a numéricos (Reales de 3 dígitos enteros con un decimal).

- Esta rutina asigna un número para mensaje de error, si en el lugar que se esperan caracteres en un rango de cero a nueve existe otro caracter.

PARAMETROS:

CADENA: 72A1 cadena con información. (E).

J : (Entera) apuntador a una posición en la cadena (E/S).

RES1: (Real) contiene el valor numérico que resulta de la conversión (S).

IBAN1: (Entera) bandera indicadora de error -- IBAN=0 no existe error, IBAN=8 existe error (S).

NUMERR: (Entera) contiene el número de error a
enviar vía pantalla o archivo (S).

ARCHIVOS:

INVOCADA POR:

DEFCAM

INVOCADA A:

SUBROUTINA: GRABA

FUNCION:

Se encarga de grabar en el archivo descriptor las líneas tecleadas por el usuario, para esto, utiliza un arreglo en forma de BUFFER con 132 caracteres.

PARAMETROS:

NOCAMP: Es el número de campo que se realiza en ese momento (E).

NOZONA: Es el número de zona que se procesa en dicho momento. (E/S).

CADENA: Arreglo de caracteres que contiene la línea tecleada por el usuario (E).

**RENG
COL** Apuntador al renglón y columna disponible para grabar la información (E/S).

BUFFER: Arreglo de caracteres donde se almacenan las líneas tecladas, antes de grabarlos (E/S).

APUNT: Conjunto de apuntadores a las diferentes áreas de las cuales se compone una gráfica (E/S).

ARCHIVOS:

* . GRF

INVOCA A:

Ninguna

INVOCADA POR:

AREAS

DEFCAM

SUBROUTINA: ERROR.FTN

FUNCION:

Desplegar en pantalla (o almacenar en un archivo auxiliar) la cadena en que se comete el error (de tipo sintáctico), así como un apuntador a la posición de la línea en donde se localiza el error, en las siguientes dos líneas se envía el número de error, así como una breve descripción de éste.

PARAMETROS:

NC: (Entera) Número de campo o ventana (E).

NUMERR: (Entera) Número de error (E).

PSCION: (Entera) posición en el vector donde está el error (E).

CADENA: (A1) cadena que contiene la línea en donde se cometió el error (E).

KONTER: (Entera) contador de invocaciones a esta rutina (E/S).

NOUT: (Entera) Número de unidad donde se envía el mensaje de error (E).

OBSERVACIONES:

Básicamente esta rutina es ocupada por el módulo de DEFINICION, por lo tanto, cuando este módulo es ocupado después de MODIFICACION, los mensajes de error se enviarán a un archivo auxiliar.

ARCHIVOS:

ERRORS.LST (Acceso secuencial) archivo auxiliar.

INVOCADA POR: Las rutinas que hacen chequeo sintáctico.

INVOCA A:

SUBROUTINA: VALFIL.FTN

FUNCION:

 Checar que exista el descriptor de la gráfica en el directorio de graficación, el cual será único para todos los usuarios del Sistema GRAFOS, esta subrutina checa fundamentalmente tres cosas; el nombre que identifica un descriptor, la familia de cuentas a la que pertenece y el número de miembro, estos elementos son grabados al dar de alta un descriptor, esta rutina checa si los tres elementos mencionados se cumplen en caso afirmativo se enciende una bandera de lo contrario se apaga. La causa de tener en el archivo directorio los tres elementos se debe a que, al ser un directorio único para los usuarios del Sistema GRAFOS es muy factible - que se repita el nombre de un archivo descriptor de gráficas, al tener estos tres elementos la única restricción que existe es que un mismo usuario no puede tener dos archivos descriptors con el mismo nombre.

PARAMETROS:

DESCFIL (REAL *8) Contiene el nombre del archivo descriptor
de gráficas (E).

EXISTE (ENTERO) Bandera indicadora de existencia o inexis-
tencia de un archivo descriptor de gráfi-
cas (S).

ARCHIVOS:

DIRECT.GRF,3 Unidad No. 1
Acceso Secuencial

INVOCADO POR:

DEFINE

INVOCA A:

Ninguna

SUBROUTINA: RENOM

FUNCION:

Este subprograma se encarga de renombrar una gráfica ya definida. Este proceso se realiza, sólo después de haber ingresado al proceso de modificación.

Si el usuario no desea renombrar la gráfica, entonces la rutina finaliza de inmediato; en caso contrario, solicita el nuevo nombre para la gráfica. Si el -- nombre pertenece a un ARCHIVO DESCRIPTOR ya catalogado, el programa lo indica y pide que se confirme si el nombre es el correcto, ya que entonces un archivo ya definido puede perderse. Si el nombre no está catalogado entonces no se realiza la verificación y en ambos casos se procede a copiar del archivo temporal (creado al salir de la etapa de modificación) al archivo definitivo.

PARAMETROS:

NOMCAT: Es el nombre de la gráfica modificada, ya sea que haya sido renombrada o no.

Es una variable alfanumérica de tipo
REAL *8 (E/S).

ARCHIVOS:

* . GRF .- Archivo Descriptor definitivo
TEMPGRAF.TMP.- Archivo Temporal

INVOCA A : Ninguno

INVOCADA POR:

DEFINE.- (Programa Principal)

SUBROUTINA: TRIMS

FUNCIÓN:

Se encarga de controlar la definición de cada una de las áreas de un campo, para esto, se vale de varios módulos.

Existen módulos que verifican la sintaxis y la validez de las instrucciones tecleadas por el usuario. Y, posteriormente a la "validación" se usa un sólo módulo para grabar la información en el DESCRIPTOR correspondiente. Las áreas a definir son:

Parámetros Alfabéticos

Parámetros Numéricos

Encabezados

Acotaciones

Pies de Página

Letreros Posicionados

Determinación de Datos

PARAMETROS:

- NOCAMP:** Indica el número de campo a definir (E)
- RENG:** Es el número de registro donde se graba la última línea del campo (S).
- COL:** Es la posición dentro del registro donde se termina la última línea del campo (S).
- BUFFER:** Contenido de la última línea del campo (S).
- IN:** Número de unidad lógica para entrada de datos (E).
- KONTER:** Contador de errores (S).
- OUT:** Número de unidad lógica para envío de mensajes al usuario (E).
- COMMONS:**
- /GRAB/ APUNT:** Apuntadores al inicio de las distintas áreas (S).
- /ALFAS/TABALF:** Tabla de nombres de parámetros alfabéticos (E).
- /NUME/TABNUM:** Tabla de nombres de parámetros numéricos (E).

ARCHIVOS:

* . GRF O TEMPGRAF.GRF

IN

OUT

INVOCADA A :

PARALE

PARNUM

PRO

SINTAX

LETPOS

DETDAT

HELP

AYUDA

CRABA

INVOCADA POR:

DEFINE

OBSERVACIONES:

Ninguna

SUBROUTINA: PARALP

FUNCION:

Valida la sintaxis correspondiente a la definición de Parámetros Alfabéticos para una gráfica. Si existe alguna falla envía los mensajes correspondientes al usuario y la línea es desechada.

En cambio, si la expresión es correcta, el nombre del parámetro es asignado a la "Tabla de Parámetros Alfabéticos" que es usada posteriormente para verificar que sólo sean usados los parámetros ya definidos en la elaboración de las áreas subsecuentes.

Las reglas de sintaxis con que debe cumplirse son:

- UN PARAMETRO ESTA CONSTITUIDO POR: NOMBRE Y CONTENIDO.
- EL NOMBRE DEL PARAMETRO ES UNA CADENA DE UNO A OCHO CARACTERES ALFANUMERICOS.

EL PRIMER CARACTER DEL NOMBRE SERA ALFABETICO

- DESPUES DEL NOMBRE APARECERA EL SIMBOLO DE --
IGUAL (=) QUE SEPARARA EL CONTENIDO Y EL --
NOMBRE.

- EL CONTENIDO SERA UNA CADENA NO VACIA DE 1NO
A SETENTA CARACTERES ASCII.

PARAMETROS:

CADENA: Arreglo de caracteres que contiene la li
nea tecleada por el usuario 72 A1 (E).

BLANCO: Variable lógica usada como bandera para
indicar que la línea tecleada está en --
blanco. (S).

MAI: Variable lógica que es usada para indicar
que se detectó algún error en la valida--
ción (S).

KTABAL: Apuntador para inserción en la "Tabla de Parámetros Alfabéticos" (E).

KONTER: Contador de errores (E/S).

NOUT: Variable entera que contiene el número de unidad lógica usada como salida.

COMMON

/ALFAS/

TABALF: Arreglo de caracteres alfanuméricos que contiene la: "Tabla de Parámetros Alfabéticos". (E/S).

ARCHIVOS:

Ninguno

INVOCA A:

ERROR

INVOCADA POR:

AREAS

SUBROUTINA PARNUM

FUNCION:

Verifica que la definición de un Parámetro Numérico cumpla con las reglas de sintaxis establecidas.

Estas reglas se asemejan mucho a las que operan para Parámetros Alfabéticos, con la única diferencia que especifica: la cadena de contenido sólo puede estar formada por números y los símbolos:

- . PUNTO
- , COMA
- : DOS PUNTOS
- GUIÓN
- * ASTERISCO

Si el Parámetro es correcto se almacena en la "Tabla de Parámetros Numéricos" para posteriores validaciones.

PARAMETROS:

CADENA: Arreglo de caracteres que contiene la línea tecleada por el usuario (E).

BLANCO: Variable lógica que indica si la línea tecleada está en blanco (S).

MAL: Variable lógica que indica si la línea contiene errores (S).

KTABNU: Apuntador de la tabla de Parámetros Numéricos (E).

KONTER: Contador de errores (E/S).

NOU: Variable entera que contiene el número de unidad lógica para salida (E).

COMMON

/NUME/

TABNUM: Arreglo de caracteres que contiene la: "Tabla de Parámetros Numéricos" (E/S).

ARCHIVOS:

Ninguno

INVOCADA A:

DEFURIA

ERROR

INVOCADA POR:

AREAS

OBSERVACIONES:

Ninguna

SUBROUTINA: METRO

FUNCION:

Esta rutina se encarga de eliminar aquellos caracteres que resultan innecesarios para cada comando - teclado por el usuario. El único caracter necesario para identificar cada comando, es la letra inicial.

También, verifica que los comandos tecleados, sean válidos sin revisar propiamente su sintáxis.

PARAMETROS:

- A: Arreglo de caracteres que contiene la línea tecleada por el usuario (E/S).
- NOZONA: Variable entera que indica el número de zona que se está capturando. (E).
- MAL: Bandera que indica si se detectó algún error (S).
- ELANCO: Bandera que indica si la línea está en blanco (S).

KONTER: Contador de errores que a cometido el
usuario (E/S).

NOUT: Variable que indica el número de unidad
lógica usada como dispositivo de salida
(E).

ARCHIVOS:

Ninguno

INVOCA A:

LLENAR

ERROR

ESPAR

INVOCADA POR:

AREAS

SUBROUTINA: ESPAR

FUNCION:

Esta rutina es una función de tipo lógica que nos indica, en base a un conjunto de parámetros si la cadena de caracteres que se está procesando contiene el nombre de un parámetro. Ya sea alfabético o numérico.

PARAMETROS:

CADENA: Arreglo de caracteres que contiene la línea tecleada por el usuario - (E).

TABLA: Arreglo bidimensional que contiene los nombres de los parámetros, ya sean alfabéticos o numéricos (E).

INICIO: Apuntador que indica el inicio de la cadena que supuestamente contiene el nombre del parámetro (E).

FIN:

Apuntador que indica el número de
caracter donde terminó la búsqueda
del probable nombre (S).

ARCHIVOS:

Ninguno

INVOCADA A:

Ninguna

INVOCADA POR:

METRO

SINTAX

DET DAT

SUBROUTINA: LLENAR

FUNCION:

Se encarga de complementar el contenido de la línea teclada por el usuario después de que los comandos han sido compactados o "Tokenizados" por la subrutina correspondiente.

Esta subrutina solo se emplea en el caso de ocurrir algún error de sintáxis y por lo tanto, solo habrán sido compactados los comandos verificados antes de ocurrir el error.

La utilización de esta subrutina se hace necesaria debido a que la cadena con comandos compactados se reasigna a la misma línea que tecló el usuario.

PARAMETROS:

CADENA: Arreglo de caracteres alfanuméricos que contiene la línea teclada por el usuario (E/S).

I: Variable entera que es usada como apuntador de la posición a partir de la cual - debe copiarse la cadena (E).

J: Variable entera que es usada como apuntador de la última posición ya reasignada a la cadena (E).

ARCHIVOS:

Ninguno

INVOCA A:

Ninguna

INVOCADA POR:

METRO

OBSERVACIONES:

Ninguna

SUBROUTINA HELP .FTN

FUNCION:

Proporcionar al usuario información de orientación en forma general del módulo en que se encuentre (DEFINICION, EJECUCION, MODIFICACION) o de algún sub-módulo, cuando se solicite se proporcionará la explicación de las instrucciones comandos o directivas con que se cuenta en los diferentes módulos o sub-módulos.

PARAMETROS:

IPAR (Entera) apuntador al registro que contiene el inicio de la información solicitada. (E)

TEX (Logical) (70) cadena que contiene la información requerida, ésta será desplegada en pantalla.

ARCHIVOS

HELPGRAF .DAT

INVOCADA POR: DEFINE, DEFCAM, AREAS, PARALF, PARNUM, METRO, --
LETPOS, DETDAT, EJECU, MODIF, EDIT.

INVOCA A: Ninguna.

SUBROUTINA; SINTAX

FUNCION:

Se encarga de controlar la verificación sintáctica de cada línea tecleada por el usuario.

Para esto, realiza repetidas iteraciones comprobando que la sintáxis sea correcta e invocando a la rutina correspondiente en cada caso.

PARAMETROS:

CADENA: Arreglo de caracteres que contiene la línea tecleada por el usuario (E).

MAL: Bandera indicadora de errores (S).

BLANCO: Bandera que indica si el usuario ha tecleado una línea en blanco (S).

CHECAR: Bandera que indica si se debe verificar la existencia de un letrero como parte de la sintáxis (S).

NADA: Sin valor

KONTER: Contador de errores cometidos por el usuario. (E/S).

NOUT: Número de unidad lógica que es usada como salida (E).

ARCHIVOS:

Ninguno

INVOCA A:

ERROR

ESPAR

TAB

REPIT

BLANCO

SUBRA

MAS

TAMAÑO

PATRON

LETRE

INVOCADA POR:

AREAS

SUBROUTINA: TAB

FUNCION:

Se encarga de verificar la sintáxis del comando tabula, cuidando que existan números asociados a dicho comando.

PARAMETROS:

- CADENA: Arreglo de caracteres que contiene la línea del usuario (E).
- I: Apuntador del arreglo cadena (E/S).
- LTRERO: Bandera lógica que indica si existió algún letrero (E/S).
- MAL: Bandera que indica si existió error en cada línea (S).
- CHECAR: Bandera para verificar la existencia de letrero o equivalente (S).

BRINCO: Bandera que inhibe el checar el letrero (S).

KONTER: Contador de errores (E/S).

NOUT: Unidad l6gica para salida (E).

ARCHIVOS:

Ninguno

INVOCA A:

ERROR

INVOCADA POR:

SINTAX

SUBROUTINA:REPIT

FUNCION:

Verifica que la sintáxis del comando REPITE - sea correcta, para lo cual se debe encontrar un valor numérico y un letrero asociados.

PARAMETROS:

CADENA: Arreglo que contiene la línea teclada por el usuario (E).

I: Apuntador del arreglo CADENA (E/S).

LTRERO: Bandera para indicar que es letrero (S).

MAL: Bandera de error (S).

KONTER: Contador de errores (E/S).

NOUT: Unidad lógica para salida de mensajes (E).

ARCHIVOS:

Ninguno

INVOCADA:

NUMER

ERROR

INVOCADA POR:

SINTAX

SUBROUTINA: BLNKO

FUNCION:

Validar la estructura del comando BLANCO, el cual puede llevar asociado un número pero no es forzoso.

PARAMETROS:

CADENA: Línea tecleada por el usuario (E).
I: Apuntador al arreglo cadena (E/S).
LTRERO: Bandera de letrero (S).
MAL: Bandera de error (S).
KONTER: Contador de errores (E/S).
NOUT: Unidad de salida para mensajes (E).

INVOCA A:

NUMER

ERROR

INVOCADA POR:

SINTAX

ARCHIVOS:

Ninguno

SUBROUTINA: SUBRA

FUNCION:

Verificar el comando SUBRAYA que subraya las posiciones indicadas por los números asociados al comando, los cuales van separados por un guión.

También se permite el comando sin ningún número asociado.

PARAMETROS:

CADENA: Línea tecleada por el usuario (E).

I: Apuntador al arreglo CADENA (E/S).

LTRERO: Bandera que indica que no existió letrero (S).

MAL: Bandera de error (S).

KONTER: Contador de error (E/S).

NOUT: Unidad lógica para salida de mensajes (E).

ARCHIVOS:

Ninguno

INVOCA A:

NUMER

ERROR

INVOCADA POR:

SINTAX

SUBROUTINA: MAS

FUNCION:

Revisar que la sintáxis del comando más (+) esté correctamente anotada. Las reglas son, que aparezca como primer comando y que cuando aparece puede existir un letrero.

PARAMETROS

CADENA: Línea de comandos (E).

I: Apuntador a la línea (E/S).

LTRERO: Bandera de existencia de letrero.

MAL: Indica si existió error (S).

BRINCO: Bandera para no verificar letrero (S).

NADA: No usada

KONTER: Contador de errores (E/S).

NOUT: Unidad lógica para mensajes de salida (E).

ARCHIVOS:

Ninguno

INVOCA A:

ERROR

INVOCADA POR:

SINTAX

SUBROUTINA: TAMAÑO

FUNCION:

Verifica que la sintáxis del comando TAMAÑO sea correcta. El cual debe indicar el tamaño en centímetros y el grueso del carácter, ya sea en forma directa (con un número real y uno entero respectivamente) o por medio de Parámetros Numéricos.

El grueso máximo permitido es cinco.

PARAMETROS:

- CADENA: Arreglo de caracteres que contiene la línea con comandos tecleada por el usuario (E).
- LTRERO: Variable lógica que indica la existencia del comando letrero o su equivalente (S).
- MAL: Variable lógica que indica error de sintáxis (S).

KONTER: Contador de errores (E/S).

NOUT: Unidad lógica utilizada como salida (E).

ARCHIVOS:

Ninguno

INVOCA A:

ESPAR

NUMER

ERROR

INVOCADA POR:

SINTAX

OBSERVACIONES:

Ninguna

SUBROUTINA: PATRON

FUNCION:

Verificar sintácticamente el comando PATRON, el cual es usado para establecer alguna simbología gráfica -- asociado a un letrero.

El comando debe especificar el tipo de gráfica y el número de patrón a emplear. Ambos elementos pueden ser dados directamente o bien empleando Parámetros Alfabéticos y o Numéricos respectivamente.

En caso de ocurrir alguna incongruencia se le especifica al usuario.

PARAMETROS:

CADENA: Arreglo de caracteres que contiene la línea teclada por el usuario (E).

- I:** Apuntador a la posición donde ocurrió la identificación del comando (E/S).
- LETRERO:** Variable lógica que indica la ocurrencia del comando letrero, o su equivalente (S).
- MAL:** Variable lógica que indica error sintáctico (S).
- CHECAR:** Variable lógica que indica que debe verificarse como siguiente comando un letrero o su equivalente (E/S).
- BRINCO:** Variable lógica que indica a la subrutina de jerarquía superior que debe evitarse - la verificación de existencia de letrero (S).
- KONTER:** Contador de errores (E/S).
- NOUT:** Número de unidad lógica que es usada como salida (E).

ARCHIVOS:

Ninguno

INVOCA A:

ESPAR

ERROR

NUMER

INVOCADA POR:

SINTAX

OBSERVACIONES:

Ninguna

SUBROUTINA: LETRE

FUNCION:

Se encarga de validar la sintaxis del comando letrero, la cual es bastante sencilla, ya que las únicas posibilidades de error son:

- que no cheque el cierre de apóstrofes o bien:
- que el letrero esté vacío.

PARAMETROS:

CADENA: Arrégllo alfanumérico que contiene el letrero a validar (E).

I: Apuntador del arreglo CADENA, que indica la posición del carácter a procesar (E/S).

LTRERO: Variable lógica que indica que se ha detectado algún letrero (S).

MAL: Variable lógica que indica si existió algún error de validación (S).

KONTER: Contador de errores (E/S).

NOUT: Número de unidad lógica que se usa como salida (E).

ARCHIVOS:

Ninguno

INVOCA A:

Error

INVOCADA POR:

Syntax

SUBROUTINA: NUMER

FUNCION:

Es una función de tipo entera, definida por el usuario para convertir una cadena de dígitos, en un valor numérico.

PARAMETROS:

CADENA: Contiene el arreglo de caracteres donde se extrae el valor numérico (E).

INICIO: Es el apuntador de la variable cadena, el cual indica donde se iniciará la conversión numérica de los caracteres (E).

FIN: Es el apuntador del arreglo alfanumérico que indica la posición donde se localiza el primer carácter no numérico. O sea, que es en esa posición - donde finaliza el número (S).

INVOCADA POR:

TAB

REPIT

BLNKO

SUBRA

TAMNO

PATRON

OBSERVACIONES:

Si los Parámetros INICIO y FINAL son de igual valor, eso significa que no se localizó un número.

SUBROUTINA LETPOS.FTN

FUNCION:

Checar sintácticamente el flujo del tren de comandos para letreos posicionados, los comandos son; Tamaño, Patrón, Posición, - Letrero, para checar sintácticamente cada uno de ellos se invoca a rutinas especializadas (una por cada comando), también se lleva el control si después de cada comando se continua en otra línea el tren de comandos (pudiera darse el caso de que el tren de comandos no cupiera en una línea de pantalla).

PARAMETROS:

CADENA (Logical) (72) cadena que contiene la información (E)

BLANCO (Logical) (11) bandera indicadora de cadena vacia (S)

MAL (Logical) (1) bandera indicadora de existencia de error (S)

IFLAG2 (Entera) bandera indicadora de continuación de línea - -
(indica también que comando continua) (E/S)

KONTER (Entera) contador del No. de veces que se invoca a la --
rutina error (S)

NOUT (Entera) número de unidad a la que se enviarán los mensa--
jes de error (a pantalla o archivo auxiliar)

ARCHIVOS:

INVOCADA POR: AREAS

INVOCADA A: DEPURA, TAMAÑO, PATRON, PSCION, LETRE, ERROR.

SUBROUTINA PSCION.PTN

FUNCION:

Checar sintácticamente el comando posición, el cual debe contener las coordenadas que posicionan un letrero así como el ángulo de inclinación (opcional, puede estar definida o no, en caso de no estarlo se asume ángulo cero grados).

Puede existir un PARAMETRO NUMERICO, en el cual deben estar contenidos estos datos, en tal caso se checa que exista dicho parámetro, esto se logra con el auxilio de la función ESPAR.

PARAMETROS:

K (Entero) apuntador a la cadena de información (E/S)

CADENA (Logical) (72) cadena que contiene la información (E)

MAL (Logical) (1) bandera indicadora de existencia de error (S)

KONTER (Entera) contador de invocaciones a la rutina de error(S)

NOUT (Entera) número de unidad a la que se enviarán los mensajes de error (a pantalla o archivo auxiliar)

OBSERVACIONES:

El apuntador (K) se recibe en el delimitador inicial del comando '(' y es retornado en una posición adelante del delimitador final ')' o sea en la coma.

ARCHIVOS:

INVOCADA POR: LETPOS

INVOCA A: ERROR, y a la función ESPAR

SUBROUTINA: DETDAT,FTN

FUNCION:

Limpiar de posibles espacios en blanco a la cadena que contiene la información de la determinación de datos, esto se logra con la ayuda de la rutina DEPURAR.

Hacer el chequeo sintáctico del nombre del banco y su identificador, además verifica si este nombre e identificador está contenido en un Parámetro Alfanumérico, esto se logra con la ayuda de la rutina ESPAR. En forma similar se hace el chequeo para el número de serie inicial (NSI) y final (NSF), año y período inicial (AI,PI), así como año y período final (AF,PF), tipo de gráfica (TIPGRA) y número de patrón (No. PAT), en algunos casos se checa si es Parámetro Numérico

puede haber tantos tipos de gráficas y números de patrón como lo indique el rango entre número de serie inicial y final.

PARAMETROS:

CADENA (Logical) (72) cadena que contiene la información (tecleada por el usuario) (E)

BLANCO (Logical) (1) bandera indicadora de cadena vacía (S)

MAL (Logical) (1) bandera indicadora de existencia de error(S)

KONTER (Entera) contador de invocaciones a la rutina de error (S)

NOUT (Entera) Número de unidad a la que se enviarán los mensajes de error (a pantalla o a un archivo auxiliar) (S)

ARCHIVOS:

INVOCADA POR: AREAS

INVOCA A: DEPURA , ESPAR, ERROR

SUBROUTINA: CATAL

FUNCION:

Se encarga de catalogar en el directorio de gráficas, el nombre de una nueva gráfica.

Para catalogar una gráfica, además del nombre debe anotar el número de cuenta del usuario, ya que se usa un directorio único para toda la instalación.

PARAMETROS:

NOMBRE: Arreglo alfanumérico de tipo REAL *8, contiene el nombre de la gráfica a catalogar.

ARCHIVOS:

DIRECT.GRF

INVOCA A:

GETTSK (rutina del sistema)

IDATE (rutina del sistema)

ASSIGN (rutina del sistema)

FDBSET (rutina del sistema)

INVOCADA POR:

DEFINE: Programa Principal

OBSERVACIONES: Ninguna

SUBROUTINA: COCLIR .FTN

FUNCION:

Cerrar los archivos que fueron ocupados durante toda la etapa de definición, los archivos ocupados son dos: el de Información de los tipos de errores que sintácticamente se cometan y el de Mensajes de ayuda u orientación para el usuario.

ARCHIVOS:

ERROR.DAT	UNIDAD No. 1
	ACCESO DIRECTO
	LONG.REG. 60 PALABRAS
	LONG.ARCH. 200 REGISTROS
HELPGRAF.DAT	UNIDAD No. 2
	ACCESO DIRECTO
	LONG.REG. 35 PALABRAS
	LONG.ARCH. 604 REGISTROS

5.4 EJECUCION DE GRAFICAS

PROGRAM EJECU.FTN

FUNCION:

Solicitar el nombre del descriptor donde se encuentra la información capturada y procesada en el módulo de DEFINICION, con la ayuda de la rutina de validación de archivos VALFIL, se checa que el descriptor exista (en caso de no existir se solicita nuevamente), con el nombre del descriptor verificado se anexa a éste la extensión .GRF y se asigna una unidad (la No. 1) con el nombre del descriptor dado (*.GRF) - dicho descriptor consta de 300 registros con 66 palabras cada registro.

Con la unidad asignada se lee el primer registro en el cual existe un vector con 11 campos lógicos, y contiene una G si es gráfica o una M si es reporte Maler, para cada caso se invoca a la rutina encargada de la ejecución correspondiente.

Se consulta si desea prueba de campos en la cual se dibujan las ventanas definidas por el usuario. Para este caso es invocada la rutina TESTCA.

Para el caso de (M) reporte MALER, se leen los datos contenidos en el registro correspondiente al número - de campo y se invoca a la rutina que se encarga de esta ejecución REPMLR.

Esta rutina se encarga también de inicializar en modo graficación y de terminar dicho modo.

Para el caso (G) gráficas se leen los datos correspondientes, al número de campos en que está asignada la gráfica y se invoca a las rutinas encargadas de su ejecución en el orden siguiente:

ACTUALIZACION DE PARAMETROS	ACTPAR
ENCABEZADOS Y PIES DE PAGINA	PIENCA
EJECUCION DE ACOTACIONES	ACOTAC
EJECUCION DE GRAFICAS	DOGRAF
EJECUCION DE LETREROS POSICIONADOS	POSLET

Una vez ejecutada esta secuencia se retorna al - punto en que se checa nuevamente que es lo que se tiene una M o una G.

PARAMETROS.- COMMON/CARFIS/ contiene las características de - los campos (salida).

VECTOR VAL (REAL), contiene los valores con las coordenadas para la prueba de campos (salida).

OBSERVACIONES:

El COMMON CARFES, es ocupado por reportes MALER, como
por ejecución de Gráficas.

ARCHIVOS.- *.GRF nombre del descriptor.

Invocado por:

**Invoca s: VALFIL, TESCA, REPMLP, PIENCA, DOACOT, DOGRAF, POSLET
PLOTS, PLOT.**

SUBROUTINA HELP .FTN

FUNCION:

Proporcionar al usuario información de orientación en forma general del módulo en que se encuentre (DEFINICION, EJECUCION, MODIFICACION) o de algún sub-módulo, cuando se solicite se proporcionará la explicación de las instrucciones comandos o directivas con que se cuenta en los diferentes módulos o sub-módulos.

PARAMETROS:

IPAR (Entera) apuntador al registro que contiene el inicio de la información solicitada (E)
TEX (Logical) (70) cadena que contiene la información requerida, ésta será desplegada en pantalla.

ARCHIVOS

HELPGRAF .DAT

INVOCADA POR: DEFINE, DEFCAM, AREAS, PARALF, PARNUM, METRO, --
LETPOS, DETDAT, EJECU, MODIF, EDIT.

INVOCA A: Ninguna.

SUBROUTINA: VALFIL.FTN

FUNCION:

 Checar que exista el descriptor de la gráfica en el directorio de graficación, el cual será único para todos los usuarios del Sistema GRAFOS, esta subrutina checa fundamentalmente tres cosas; el nombre que identifica un descriptor, la familia de cuentas a la que pertenece y el número de miembro, estos elementos son grabados al dar de alta un descriptor, esta rutina checa si los tres elementos mencionados se cumplen, en caso afirmativo se enciende una bandera de lo contrario se apaga. La causa de tener en el archivo directorio los tres elementos se debe a que, al ser un directorio único para los usuarios del Sistema GRAFOS es muy factible que se repita el nombre de un archivo descriptor de gráficas, al tener estos tres elementos la única restricción que existe es que un mismo usuario no puede tener dos archivos descriptores con el mismo nombre.

PARAMETROS:

DESCFIL (REAL *8) Contiene el nombre del archivo descriptor
de gráficas (E).

EXISTE (ENTERO) Bandera indicadora de existencia o inexis-
tencia de un archivo descriptor de gráfi-
cas (S).

ARCHIVOS:

DIRECT.GRF Unidad No. 1
Acceso Secuencial

INVOCADO POR:

DEFINE

INVOCA A:

Ninguna

SUBROUTINA: TESTCA.FTN

FUNCION:

Mostrar en papel los campos declarados por el -
usuario esto es, sus gráficas pueden salir si así lo desea
enmarcados y con un pequeño letrero que identifica al campo
o área declarado por el usuario, esto es útil ya que da una
buena presentación a las gráficas elaboradas.

PARAMETROS:

VAL (REAL *4) (44) Arreglo donde van contenidas las coorde-
nadas máximas y mínimas de cada campo o
área (hasta 11 campos o áreas) (E).

ICAMP (ENTERO) Contiene el número de campos o áreas que
se enmarcarán. (E).

ARCHIVOS:

Ninguno

INVOCADO POR:

EJECU

INVOCA A:

ALFANU (RUTINAS VERSAPLOT: PLOT, SYMBOL).

SUBROUTINA: ALFANU

FUNCION:

Este módulo se encarga de agregar los caracteres ASCII que corresponden a un número entero de cero a diez. En los caracteres son colocados en los dos últimos elementos de una cadena de caracteres ASCII que contiene en los primeros seis elementos la (constante): "CAMPO".

De esta manera es posible que el usuario identifique el número de campo que se está elaborando.

PARAMETROS:

CADE1: Arreglo de caracteres que contendrá el letrero que identifica al número de campo (S).

NUMERO: Variable entera que indica el número de campo (E).

ARCHIVOS:

Ninguno

INVOCADA A:

Ninguna

INVOCADA POR:

TESTCA

OBSERVACIONES:

Ninguna

SUBROUTINA: REPMLR

FUNCION:

Este módulo tiene como función imprimir en un campo determinado de la gráfica un cuadro estadístico, previamente elaborado por medio del "GENERADOR DE REPORTES: MALEH".

Lo primero que debe realizarse es localizar el nombre del reporte para identificarlo y poder accederlo. Como dicha información está almacenada en el descriptor: de ba localizarse por medio de los apuntadores que existen para cada campo en los primeros registros del Descriptor.

Después de haber localizado el registro en el cual se encuentra almacenada la información relativa al campo, se realiza la búsqueda del delimitador punto y coma (;) ya que después de éste, se encuentra el nombre del reporte.

El reporte es impreso en el campo correspondiente, de acuerdo con las características establecidas al momento de la definición, las cuales son:

Tamaño de los Caracteres, Número de Líneas del Campo, Número de Caracteres en cada Línea, y Dimensiones Físicas del Campo.

Para realizar el dibujo de los caracteres alfanuméricos que componen el cuadro se utiliza una rutina del proveedor que es explicada en la sección de apéndices de la tesis.

PARAMETROS:

NOCAMP: Variable entera que indica el número de campo en el cual se colocará el reporte (E).

AREA COMMON:

CARFIS

TAM: Variable real que determina el tamaño de los caracteres (E).

NUMLIN: Variable entera que indica el número de líneas del campo (E).

NOCHAR: Variable entera que contiene el número de caracteres por línea (E).

XMIN:

XMAX: Absisas mínima y máxima del campo (E).

YMIN

YMAX: Ordenada mínima y máxima del campo (E).

ARCHIVOS:

/*.GRF/UNID=7/ACC.DIR./

Archivo Descriptor de la Gráfica.

/*.LST/UNID=9/ACC.SEC./

Reporte que se asigna al campo.

INVOCA A:

ERR

SYMBOL (Versaploth)

INVOCADA POR:

EJECU

OBSERVACIONES:

Para información sobre el Generador de Reportes MALER, consúltese la referencia número tres de la bibliografía.

SUBROUTINA FIENCA

FUNCION:

Se encarga de ejecutar los comandos tecleados por el usuario para el área de encabezados y para pies de página.

Se usa la misma subrutina para ambas áreas, debido a que los comandos permitidos son los mismos. Para la ejecución de -- los comandos se tienen los siguientes datos:

- a) LETREROS QUE SE IMPRIMEN DE INMEDIATO
- b) LETREROS QUE SE ALMACENAN Y POSTERIORMENTE SE IMPRIMEN
- c) COMANDOS QUE MODIFICAN A LOS LETREROS.

Cuando aparece un letrero o parámetro alfabético, este debe estar afectado por un tipo de justificación, - esto es: izquierda, derecha, centrado o tabula. Si la justificación es izquierda o tabula, el letrero se imprime de inmediato. Pero, si la justificación es derecha o centrado, - primero se almacena el letrero en un vector que colocará en forma correcta todos los letreros con la misma justificación

y en la misma línea. Y después, al cambiar de línea, todos los letreros son impresos. Es necesario también, conservar el tamaño de los caracteres para cada letrero almacenado.

Los comandos permitidos son:

IZQUIERDA

TABULA

DERECHA

CENTRADO

REPITE

BLANCO

SUBRAYA

LETRERO

TAMAÑO

Parámetro Alfabético

+ (continuación de línea).

Cuando un letrero o parámetro aparece sin ningún comando - asociado, este será centrado.

PARAMETROS:

NUMZON: Variable entera que indica el número de zona a ejecutar. 3 encabezados y 5 - pies de página.

YUSADA: Variable de tipo real que contiene la - cantidad de espacio usado por la rutina al elaborar el área correspondiente.

ARCHIVOS:

/*.GRF/UNID=7/ACC.SEC/

Este es el archivo descriptor de la gráfica, de donde se - van obteniendo los comandos a ejecutar. Este archivo ya - está abierto desde el programa principal.

INVOCADA POR: EJECU: programa principal que se encarga- de ejecutar una gráfica.

INVOCA A: DAMEIN:

DAMELI:

SALEDC:

.../

RECPAR:

IMPTEX:

KEPTX:

ERR:

NOREAL:

PLOT

NEWPEN

LLAMA A LAS RUTINAS:

Ninguna

ES LLAMADA POR
LAS RUTINAS:

PIENCA:

ACOTAC:

IMPTEX

EJEBOB:

ENVOL:

OBSERVACIONES:

Ninguna

SUBROUTINA: ERR

FUNCION:

Se encarga de extraer del archivo de mensajes de error, el texto correspondiente y lo despliega en la pantalla cuando ocurre algún error en la ejecución de una gráfica.

El mecanismo es bastante sencillo, ya que existe una relación biunívoca entre el número de error y el número de registro del archivo.

PARAMETROS:

NUMERR: Variable de tipo entera, que contiene el número de error que ha ocurrido.

ARCHIVOS:

/ERROR.DAT/UNID=6/ACC.DIR/

Este es el archivo que contiene los mensajes de error.

LLAMA A LAS RUTINAS:

Ninguna

ES LLAMADA POR
LAS RUTINAS:

PIENCA:

ACOTAC:

IMPTEX:

EJE OBS:

ENVOL:

OBSERVACIONES:

Ninguna

SUBROUTINA: DAMELI

FUNCION:

Se encarga de extraer una línea teclada por el usuario. Dichas líneas se encuentran almacenadas en el archivo descriptor de la gráfica.

Para extraer una línea de comandos se utiliza un arreglo que tiene la misma longitud que los registros del archivo (132 caracteres). En dicho arreglo se conservan una o más líneas que el usuario tecló y las cuales están separadas por un caracter de control: (BELL).

Mientras no se localice el caracter de control, la rutina asignará los caracteres que aparecen en el registro al arreglo: LINEA, y dicho arreglo, al terminar la rutina contendrá los comandos teclados por el usuario en una línea de setenta y dos caracteres máximo.

PARAMETROS:

- INICIO:** Variable entera que indica la posición dentro del registro, donde se inicia la línea.
- FINAL:** Variable entera que apunta a la posición dentro del registro donde se ha localizado el caracter de control y que indica el final de la línea.
- NOREG:** Indica el número de registro que se accesa en el momento de extraer las líneas.
- LINEA:** Arreglo de setenta y dos caracteres -- que contiene la línea extraída del archivo.
- BUFFER:** Arreglo de ciento treinta y dos caracteres que se usa para contener temporalmente el contenido de un registro.

ARCHIVOS:

/ *.GRF/UNID=7/ACC.DIR/

Es el archivo descriptor de la gráfica, el cual ya ha sido abierto en el programa principal.

LLAMA A LAS RUTINAS:

Ninguna

ES LLAMADA POR
LAS RUTINAS:

PIENCA: Se encarga de elaborar encabezados y -
pies de página.

ACOTAC: Elabora los encabezados de la gráfica.

POSLET: Realiza los letreros posicionados de
la gráfica.

DAMEIN: Obtiene la posición inicial donde se -
imprimirán los pies de página.

RECPAR: Se encarga de obtener el contenido de
un parámetro ya sea alfabético o numé-
rico.

VACIA: Crea las tablas de números de series y periodicidad, así como las características de los Bancos de Datos.

OBSERVACIONES:

Ninguna

SUBROUTINA: DAMEIN

FUNCION:

Esta subrutina se encarga de calcular la posición dentro del campo, a partir de la cual se inicia la impresión de los pies de página.

Para esto, debe revisar los cambios de línea, ya sea que se deban al comando BLANCO o bien a que se cambie físicamente de línea. Además, es necesario tomar en cuenta el comando tamaño que afecta directamente, el espacio asignado para cada renglón.

Por otro lado, no siempre el cambio de línea en la definición, implica un cambio de línea en la ejecución. Esto es cuando aparece el comando de continuación de línea (+).

PARAMETROS:

NOREG: Es una variable entera que indica el número de registro del archivo descriptor donde se inician los pies de página (E).

INICIO: Variable entera que indica la posición dentro del vector donde se inicia el área de pies de página (E).

YINI: Variable de tipo real que indica la posición en el eje de ordenadas a partir de donde se debe iniciar la impresión de pies de página (S).

TAMM: Variable real que indica el tamaño por "default" de los caracteres (E).

ARCHIVOS:

/ *.GRF/UNID=7/ACC.DIR/

Archivo descriptor, ya ha sido abierto en el programa principal.

LLAMA A LAS RUTINAS:

- DAMELI: Extrae líneas del archivo descriptor.
- NOREAL: Convierte a valor numérico una cadena de dígitos.
- RECPAR: Se encarga de obtener el contenido de - un parámetro alfabético o numérico.

ES LLAMADA POR LAS RUTINAS:

- PIENCA: Módulo que elabora encabezados y pies - de página de un campo.

OBSERVACIONES:

Ninguna

SUBROUTINA NOREAL.FTN

Función:

Convertir un fragmento de una cadena alfanumérica en un numérico real. Esta rutina tiene la característica de que si el dato está con un punto decimal, lo regresa en valor decimal - pero si no tiene dicho punto entonces lo retorna sin este punto.

PARAMETROS:

LINEA (logical) (72) vector donde está contenida la información (E).

POINT (entera) apuntador a la cadena (E/s).

VALNUM (real) resultado de la conversión (S)

OBSERVACIONES: El apuntador POINT se recibe en el primer caracter a convertir y es regresado en el delimitador como el valor regresado es de tipo real, cuando se tenga la seguridad de que se requiere de tipo entero basta con asignar a la salida de este. - Una variable entera. Se convierten hasta 4 dígitos enteros.

../..

ARCHIVOS:

INVOCADA POR: VACIA, TMNO, PAT, POS

INVOCADA:

SUBROUTINA RECPAR.FTN

Función:

Recuperar un parámetro alfanumérico o numérico, este parámetro es buscado en el descriptor con el auxilio de la rutina DAMELI, la cual va sacando el nombre de cada parámetro y su contenido, se checa el nombre del parámetro y si existe dicho nombre se vacía el contenido de este en un vector de salida --- (CONTE), para el caso de que exista o no exista el parámetro se enciende o apaga una bandera (SIES).

PARAMETROS:

LINEA: (logical) (72) cadena que contiene la información (E).

POINT: (entera) apuntador a la cadena LINEA, (E/S).

CONTE: (logical) (72) cadena que tiene el contenido del nombre del parámetro (S).

ALFNUM: (entera) indicador de tipo de parámetro alfabético (1), numérico (2) (E)

SIES: (entera) indicador de existencia del parámetro - alfabético o numérico, existe (TRUE) no existe (FALSE). (S).

../..

COMMON/APUNT/APUN: entera) (8,2) apuntador a las diferentes áreas de trabajo (E).

OBSERVACIONES: El apuntador POINT, se recibe apuntando al primer caracter del nombre del parámetro y se regresa - apuntando en el delimitador (=) del parametro.

ARCHIVOS:

INVOCADA POR: VACIA, NO, PAT, POS

INVOCADA A: DAMELI

SUBROUTINA: SALEDC

FUNCION:

Esta rutina se emplea para imprimir los textos o letreros que se justifican a la derecha o que aparecen centrados. Estos textos fueron previamente almacenados en un vector por la rutina KEPTX, así como los distintos tamaños, gruesos de línea y longitudes que afectan a cada uno.

Cada vez que se tiene un tamaño distinto, se invoca a la rutina SYMBOL y se anota el texto desde la primera posición del arreglo para imprimirlo, y se establece el tamaño y grueso de línea correspondientes.

PARAMETROS:

VECTOR: Arreglo que contiene los letreros a imprimir (E).

POINTR: Apuntador del arreglo VECTOR (E).

FINALS: Arreglo de enteros que indica la posición final de cada texto (E).

VECTAM: Arreglo que contiene los tamaños correspondientes a cada letrero (E).

GROSCR: Arreglo que contiene los gruesos de línea que corresponden a cada texto (E).

YACT: Indica la ordenada a partir de la cual se imprimirán los textos (E).

JUSTIF: Variable entera que indica si el letrero aparecerá centrado o justificado a la derecha (E).

ARCHIVOS:

Ninguno

INVOCA A:

NEWPEN (Versaploth)

SYMBOL (Versaploth)

INVOCADA POR:

PIENCA

OBSERVACIONES:

Ninguna

SUBROUTINA: IMPTEX

FUNCION:

La función de este subprograma es imprimir los letreros cuya justificación sea a la izquierda o con tabulación hacia alguna posición. Para esto, debe tomarse en cuenta el tamaño actual del caracter y diferenciar si el texto a imprimir es el contenido de un parámetro o parte de una línea tecleada por el usuario. Para realizar la impresión del letrero se usa un vector auxiliar, en el cual se copia desde la primera posición, el letrero a imprimir, y posteriormente, se invoca a la rutina SYMBOL para realizar la impresión.

PARAMETROS:

LINEA: Arreglo de caracteres que contiene el texto a imprimir (E).

TAMACT: Variable real que contiene el tamaño actual del caracter (E).

XACT,YACT: Variables reales que contienen las coordenadas de inicio del letrero (E/S, E).

DONDE: Apuntador del arreglo LINEA (E/S).

PAR: Variable lógica que indica si el texto es un parámetro o no. (E).

NN: Variable entera que indica el número de veces que debe imprimirse el letrero -- (E/S).

ARCHIVOS:

Ninguno

INVOCA A:

SYMBOL (Versaploth)

INVOCADA POR:

PIENCA

ACOTAC

OBSERVACIONES:

Ninguna

SUBROUTINA: KEPTEX

FUNCION:

Se encarga de almacenar los textos que deben justificarse a la derecha o los que deben centrarse.

Para esto, debe tomarse en cuenta no sólo el letrero, sino también el tamaño de éste. Los letreros son almacenados en un vector de 200 elementos, lo cual significa que una sola línea podrá tener como máximo 200 caracteres.

El tamaño que corresponde a cada letrero se almacena en otro vector de 10 elementos, así que en una sola línea se permiten diez cambios de tamaño como máximo. También se usa otro vector auxiliar que indica donde termina un letrero con el mismo tamaño.

Estos elementos, sirven para que posteriormente los letreros sean justificados e impresos adecuadamente, por la rutina SALEDC.

PARAMETROS:

- VECTOR: Es el arreglo que sirve para almacenar los letreros a justificar (E/S).
- LINEA: Es el arreglo que contiene el letrero que será almacenado (E).
- KONT: Variable entera que indica el número de letrero que se está almacenando (E).
- POINTR: Apuntador del arreglo VECTOR (E/S).
- VECTAM: Arreglo de variables reales que contiene los tamaños de los distintos letreros (S).
- FINALS: Arreglo de enteros que indica el final de cada letrero (S).
- DONDE: Apuntador del arreglo LINEA (E).
- PAR: Variable lógica que indica si el letrero es un parámetro alfabético (E).

GROSOR: Arreglo de enteros que indica el grueso de línea para los caracteres de cada letrero (S).

GROSLI: Variable real que indica el grueso del letrero que se almacenará (E).

NN: Variable entera que indica el número de veces que un letrero debe repetirse (E).

TAMACT: Tamaño de los caracteres para el letrero en cuestión (E).

ARCHIVOS:

Ninguno

INVOCA A:

Ninguna

INVOCADA POR:

PIENCA

OBSERVACIONES:

Ninguna

SUBROUTINA: ACOTAC

FUNCION:

La función de este módulo es elaborar el área de las acotaciones de la gráfica. Lo cual consiste en imprimir de inmediato aquellos letreros o parámetros alfabéticos que aparecen en dicha zona. Estos letreros y/o parámetros, se ven afectados por los siguientes comandos:

BLANCO

REPITE

SUBRAYA

TAMAÑO

PATRON

+ (continuación de línea)

Las acotaciones aparecerán al lado izquierdo - del área gráfica y serán justificados hacia el límite inferior del campo en cuestión (XMIN).

En esta rutina, a diferencia de los encabezados o pies de página, no es necesario almacenar los letreros en vectores, para su posterior impresión, ya que siempre se imprimen de inmediato por estar justificados a la izquierda.

PARAMETROS:

XUSADA: Es la longitud ocupada por el letrero más largo anotado en acotaciones, incluyendo el patrón gráfico, en caso de existir.

YACT: Es la posición a partir de la cual, se inicia la impresión de las acotaciones.

ARCHIVOS:

/ * .GRF/UNID=7/ACC.SEC./

Este es el archivo descriptor de la gráfica. El cual, ya fue abierto en el programa principal.

LLAMA A LAS RUTINAS:

- DAMELI: Se encarga de extraer del archivo descriptor, una línea de comandos.
- EJEPAT: Su función es dibujar un patrón gráfico - asociado con un letrero de acotaciones.
- ERR: Se encarga de desplegar los posibles errores al ejecutar los comandos.
- IMPTEX: Imprime un letrero o parámetro alfabético.
- NOREAL: Convierte a valor numérico una cadena de dígitos.
- RECPAR: Obtiene el contenido de un parámetro alfabético o numérico.
- PLOT: (VERSAPLOTH) mueve la "Pluma" para trazar líneas o posiciones en algún punto.
- NEWPEN: (VERSAPLOTH) aumenta el grosor de las línneas a trazar.

ES LLAMADA POR
LAS RUTINAS:

EJECU: Programa Principal

OBSERVACIONES:

SUBROUTINE: EJPAT.FTN

FUNCION:

Elaborar el identificador (patrón) que será colocado en la zona de acotaciones, este identificador es elaborado para los casos de identificar las gráficas: Histograma, Envolvente, Polígono, Versatec, Impulsos.

El identificador es colocado a continuación de un texto que el usuario define, esta rutina recibe el lugar donde se pondrá el patrón y su tamaño, para cada tipo de gráfica existe un conjunto de patrones que lo identifican, se recibe también cuál de estas es el que se pondrá las acotaciones.

PARAMETROS:

TIPGRA (ENTERO)	Contiene el tipo de gráfica que se identifica con un patrón (E).
NUMPAT (ENTERO)	Contiene el número de patrón que se ocupará (E).

X (REAL *4) Coordenada en X del patrón que se imprimirá (E).

Y (REAL *4) Coordenada en Y del patrón que se imprimirá (E).

TAMACT (REAL *4) Tamaño que tendrá el patrón (E).

ANG (REAL *4) Angulo al que se imprimirá el patrón (E).

ARCHIVOS:

Ninguno

INVOCADO POR:

ACOTAC

INVOCA A:

(RUTINAS VERSAPLOT: PLOT, TONE, SYMBOL).

SUBROUTINA: POSLET,FTN

FUNCION:

Efectuar la lectura de apuntadores para letreros posicionados, y con estos efectuar la lectura de los datos de letreros posicionados, en esta rutina se invoca a la rutina - que saca líneas (físicas del tamaño de la pantalla), con estas líneas leídas en base al delimitador que se encuentre, se invocará a la rutina de tamaño, Patrón Posición, Letrero, que son los comandos que se requieren para ejecutar un letrero posicionado este proceso se repite "N" veces hasta que no existan más datos en el espacio de letreros posicionados.

Al invocar a cada rutina de comandos éstas reciben el apuntador en el primer caracter del comando, y lo regresan una posición adelante del delimitador (,), puede darse el caso que una serie de comandos no termine al tomar la línea esto es detectado con (&), la rutina por si sola interpreta esto y sigue con el mismo comando, esto dicho en otras palabras en una línea normal deben estar los comandos Tamaño, - Patrón, Posición, Letrero, pero cuando no se llegue a terminar esta línea en una línea normal la pantalla ocupará un (&).

PARAMETROS:

COMMON/APUNT/APUN/ (entera) (0,2) apuntadores al --
área de letreros posicionados --
(entrada).

ARCHIVOS: La unidad N°. 1 *.GRF abierto en programa ejecu.

INVOCADA POR: EJECU

INVOCA A: DAMELI, TMNO, PAT, POS, LTR, NEWPEN, SYMBOL

SUBROUTINA TMNO.FTN

Función:

Obtener en dos variables los datos del comando tamaño, que son el tamaño del caracter y el grueso de línea al ser invocada esta rutina, el apuntador IAP, se recibe en primer caracter del comando (;) y es retornado una posición a delante del delimitador externo del comando (,), esta rutina checa si lo que se tiene de información es un parámetro numérico para lo cual se auxilia de la rutina RECPAR, sea el caso de parámetro numérico o no, se ocupa a la rutina NOREAL, que convierte la cadena en un valor numérico real y que fuera de ella es donde se asigna una variable entera o real según se desee, el default para grueso de línea es el N°. 1 asignado en Poslet.

PARAMETROS:

CDNA: contiene la información (lógica) (72) (E)

IAP : (entera) apuntador a la cadena (E/s)

TAMACT: (real) tamaño del caracter (S)

INP: (entera) grueso de línea (S) por default en ejecución tiene asignado el N°. 1

ARCHIVOS:

INVOCADO POR: POSLET

INVOCADO A: NOREAL, RECPAR

SUBROUTINA PAT.FTN

FUNCION:

Obtener dos variables (las dos de tipo entero) que contienen el tipo de gráfica y el número de patrón gráfico o ashurado (dados por catálogo).

El apuntador IAP, se recibe en el primer caracter del comando (:) y se retorna una posición adelante del delimitador externo (,). Esta rutina checa si los datos son Parámetros Alfabéticos o Numéricos, para lo cual, cuando es Parámetro se auxilia de la rutina RECPAR para el caso de tipo de gráfica, se checa también qué tipo de gráfica se trata, el default es el No. 1 del catálogo para el caso de Patrón o Ashurado. Sea Parámetro Numérico o no, se auxilia con NOREAL, el cual obtiene en valor numérico (real) el valor de default es el No. 1 del catálogo asignado desde POSLET.

PARAMETROS:

CDNA: (logical) (72) cadena que contiene la información (E).

IAP: (entera) apuntador a la cadena (E/S).

TIPGRA: (entera) tipo de gráfica asignada al siguiente orden: 1-P, 2-H, 3-V, 4-E, 5-I, 6-C (S).

NUMPAT: (entera) número de patrón gráfica o ashurado (S).

ARCHIVOS:

INVOCADA POR:

POSLET

INVOCA A:

NOREAL

RECPAR

SUBROUTINA POS.FTN

Función:

Obtener en 3 variables de tipo real los valores de --
los ejes coordenados y del ángulo.

En caso de ser parámetro numérico, deberán de ser los 3 datos
contenidos en esta, para lo cual se auxilia la rutina RECPAR.

Sea parámetro o no se auxilia de la rutina NOREAL.

PARAMETROS:

CDNA -(logica) (72) cadena que contiene la información
(E).

IAP - (entera) apuntador a la cadena (E/s)

X - (real) valor del eje X (S)

Y - (real) valor del eje Y (S)

ANG.- (real) ángulo de inclinación (S)

ARCHIVOS:

INVOCADA POR: POSLET

INVOCA A: NOREAL, RECPAR

SUBROUTINA: LTR,FTN

FUNCION:

Obtener el texto para ser impreso, así como un apuntador y un contador de caracteres (JA), en (JB) se tiene el contador para el caso de que se tenga un ampersan (&) en la línea, esta variable guarda el valor.

PARAMETROS:

CDNA: (logical) (72) cadena que contiene la información (E).

IAP: (entera) apuntador a la cadena (E/S)

JA: Contador de caracteres (entera) (S)

JB: (entera) contador de caracteres para -- cuando el letrero continúa en otra línea (S).

TEX: (logical) (72) texto a imprimir (S)

ARCHIVOS:

INVOCADA POR: POSLET

INVOCA A:

libre para la gráfica y además en --
posición de 1/14 del vértice infe--
rior derecho. (Todas son variables -
reales).

COMMON/GR2/ABX, ORDY, contiene la longitud de los ejes
X, y (en valor real) (salida)

COMMON/BAT/TBAN, TBD es un COMMON que crean entre la --
rutina VACIA y CRETAB y que ocupara
TOMDE en este lugar no se ocupa ---
(Nota son unas tablas.)

ARCHIVOS: Ninguno

INVOCADA POR: El módulo de ejecución EJECU

INVOCA A: VACIA, TOMDE

SUBROUTINA DOGRAF.FTN

Función:

Con los datos enviados por las rutinas de encabezados y Píes de Página y la de acotaciones (y2, y3; X2), se calcula la ventana disponible para graficación además se posiciona la pluma a 1/14 de la arista inferior derecha de dicha ventana.

Antes de estos calculos se invoca a la rutina VACIA -- que con ayuda de otra rutina CRETAB, crearán las tablas para -- las gráficas.

Después de calcular la ventana se invoca a la rutina -- que ejecuta los ejes coordenados y el tipo de gráfica que se requiera TOMDE.

PARAMETROS:

Y3 - (Real) - Límite inferior de encabezados (Entrada)
Y2 - (Real) Límite superior de Píes de Página (Entrada)
X2 - (Real) Límite izquierdo de acotaciones (Entrada)
COMMON/APUNT/APUN, de paso para otras rutinas no se --
ocupa en esta.

COMMON/CR1/XNMIN, XNMAX, YNMIN, YNMAX

Contiene las coordenadas de la ventana

../..

SUBROUTINA: VACIA.FTN

FUNCION:

Limpiar las tablas que serán usadas (TBAN, TBD, DAT) con un apuntador a la determinación de datos que se encuentra contenida en APUN (7,1) se lee el registro con los datos. En dicha línea pueden existir datos como son: NOMBRE DEL BANCO, NUMERO DE SERIE INICIAL Y FINAL, AÑO INICIAL Y FINAL, PERIODO INICIAL Y FINAL, TIPO DE GRAFICA Y PATRON.

Para tomar cada uno de los datos se invoca a la rutina DAMELI, que saca la cadena de la figura anterior.

- Para el caso en que se tengan Parámetros Alfabéticos o Numéricos se invoca a recuperador de Parámetros RECPAR.

- En caso de no ser parámetro, entonces se invoca a la rutina NOREAL que convierte la cadena alfanumérica en valores numéricos.
- Para el caso donde se acepte, no dar datos se asigna valor de cero a esas variables.
- Con las variables del nombre del banco, Número de serie inicial y final, año y período inicial y final, el tipo de gráfica y el número de patrón se invoca a la rutina que creará las tablas.

PARAMETROS:

COMMON/APUNT/APUN.- Apuntador a las diferentes áreas de trabajo (INTEGER * 2 (8,2) (Entrada).

BUFFER(LOGICAL * 1) (132).-Contiene toda la información que se ve en la figura de la página anterior.

Los siguientes parámetros son de salida:

BANCO (LOGICAL*1) (10) nombre del banco sin extensión
NSI (entera) número de serie inicial

.../..

NSF (entera) número de serie final

IAI (entera) año inicial

IPI (entera) período inicial

IAF (entera) año final

IPF (entera) período final

GR (logical*1) (8) tipo de gráfica

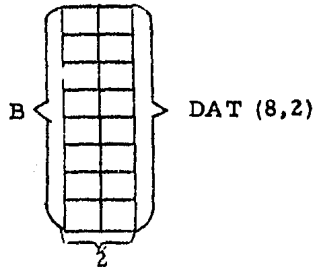
NP (logical *1)(8) patron gráfico o asurado

ARCHIVOS: *.GRF unidad 1 abierto con anterioridad en el principal

INVOCADA POR: DOGRAF

INVOCADA A: DAMELI, RECPAR, NOREAL, CRETAB

La tercera tabla contiene los atributos físicos para la asignación de los bancos declarados en la tabla primera.



Al recibir el nombre del banco se completa con la extensión .BDA, se decodifica el tipo de banco, se busca que el nombre del banco exista en el directorio de bancos y al localizarlo se crean las tablas ya mencionadas.

PARAMETROS: BANCO (LOGICAL *1) (10). - Contiene el nombre del banco sin extensión (entrada)

LOC (entera) contador de tabla de 1 a 8 (E/S)

NRA (entera) contador de registros ocupados (E/S)

NSI (entera) número de serie inicial (E)

LAI (entera) año inicial (E)

IPI (entera) período inicial (E)

LAF (entera) año final (E)

.../..

IPF (entera) período final (E)

GR (logical) tipo de gráfica (E)

NP (entera número de patrón ó asegurado)

KTIP (entera) tipo de banco (Si es = 9 tabla llena)

COMMON/BAT/TBAN, TED, DAT. - Las tablas ya elaboradas (S)

TBAN (8,11) alfanumérica, TBD (8,6) entera, DAT (8,2) entera.

ARCHIVOS: DIRECT. BDA (directorio de bancos)

INVOCADA POR: VACIA

INVOCA A:

SUBROUTINA TOMDE.FTN

Función:

De los valores almacenados en la tabla TBD

Localizar:

Año y período mínimo iniciales

Año y período máximo finales

Se genera vector de años comprendidos entre año mínimo y máximo, se ajusta al rango de período RANPE, de acuerdo al tipo de banco que se tenga en la tabla. Se invoca a las rutinas que crean los ejes coordenados, con -- el nombre del banco contenido en tabla TBAN , se asigna en la unidad 4 dicho banco de datos, se asignan los valores de las tablas TBD.

Con el tipo de gráfica contenida en TBAN posición 11 - se invoca a la rutina que ejecute un tipo de gráfica este método de invocaciones se repite hasta 8 veces.

PARAMETROS:

COMMON/BAT/TBAN, TBD, DAT (entrada)

TBAN (lógica) (8,11) TBD (entera) (8,6)

DAT (entera) (8,2)

../..

COMMON/GR1/XNMIN, XNMAX, YNMIN, YNMAX(entrada)

coordenados de la nueva ventana y
colocado a 1/4 del vertice inf. —
derecho.

COMMON/GR2/ABX.ORDY (entrada)

longitud de los ejes coordenados

COMMON/GR3/ NOSE, IAI, IPI, IAF, IPF, IAMIN, IPMIN, KTIP,
XS, YS

NOSE (entera) número de series del banco (salida)

IAF (entera) año inicial (salida)

IPI (entera) período inicial (salida)

IAF (entera) año final (salida)

IPF (entera) período final (salida)

IAMIN (entera) año inicial mínimo (salida)

IPMIN (entera) período inicial mínimo (salida)

XS, Ys (reales) origen de los ejes coordenados (salida)

ARCHIVOS: *.DAT

INVOCADA POR: DOGRAF

INVOCADA A: POLI, HISTO, VERSA, ENVOL, IMPUL, CURVA

PREPAR, ESCALS, EJETIM, EJEBOB

SUBROUTINA: PREPAR

FUNCION:

Se encarga de elaborar los dos arreglos de -
números reales que constituyen los pares coordenados para
trazar la gráfica de la serie.

Para elaborar estos, primero accesa el banco
de datos, y extrae la serie correspondiente y posteriormen
te, en base a la periodicidad especificada por el usuario
coloca los datos en un vector a partir de la primera posi-
ción, y agrega los parámetros requeridos por Versaploth.

Por otro lado, genera el vector de coordenadas
correspondiente al eje de los tiempos, también en base a -
la periodicidad especificada.

Y por último, también calcula el número de ob-
servaciones a graficar para esa serie en especial.

PARAMETROS:

- RANPE: Es el número de observaciones a graficar tomando en cuenta el rango del período de todas las series a graficar (E).
- FVAL: Es el valor numérico a graficar de todas las series (E).
- DVAL: Es el número de unidades no escaladas por unidad de medida real (E).
- VEC: Arreglo de números reales que contiene los valores de la serie a graficar en forma ordenada (S).
- TIM: Arreglo de números reales que contiene los valores correspondientes al eje -- del tiempo (S).
- LONG: Es el número de observaciones a graficar para esa serie (S).

COMMON /GR3/

NOSE: Es el número de serie a graficar (E).

IAI, IPI: Año y período iniciales de la serie a graficar (E).

IAF, IPF: Año y período finales de la serie a graficar (E).

IAMIN, IPMIN: Año y período mínimos de todas las series a graficar (E).

KTIP: Indica el tipo de banco que se utilizará (E).

XS, YS: No son usadas.

ARCHIVOS:

/ *. BDA/UNID = 4/ ACC.SEC/

Es el "Banco de Datos" de donde se extrae la información.
(El Banco ya fue abierto).

LLAMA A LAS RUTINAS:

Ninguna

ES LLAMADA POR
LAS RUTINAS:

TOMDE: Controla la graficación de todas las
series.

OBSERVACIONES:

Ninguna

SUBROUTINA: ESCALS

FUNCION:

Se encarga de calcular la escala necesaria para graficar el rango de valores numéricos de todas las series de tiempo en las dimensiones del eje de las abscisas. El escalamiento se hace con una dimensión supuesta, ya -- que además se calcula un factor de escalamiento adicional para toda la gráfica.

Para realizar este escalamiento es necesario leer todas las series a graficar y efectuar una búsqueda de los valores mínimo y máximo, pero sólo dentro del período especificado por el usuario, para aparecer en la -- gráfica.

En seguida, la relación de escalamiento será, la diferencia entre el mínimo y máximo contra la longitud supuesta del eje de las ordenadas. $DVAL = MAXIMO - MINIMO$.

PARAMETROS:

FUAL: Es el valor mínimo de los datos a graficar (S).

DVAL: Es la relación de escalamiento, expresado en unidades no escaladas por unidades físicas (S).

FACT: Es el factor de escalamiento adicional para toda la gráfica en general (E).

ARCHIVOS:

/ *. BDA/UNID = 4/ACC.DIR/

Son los "Bancos de Datos" donde se encuentran almacenadas las series de tiempo a graficar.

LLAMA A LAS RUTINAS:

SCALE: Se encarga de escalar un vector de números reales a una dimensión dada (RUTINA VERSAPLOTH).

ES LLAMADA POR
LAS RUTINAS:

TOMDE: Se encarga de controlar la graficación
de todas las series de tiempo.

OBSERVACIONES:

Ninguna

SUBROUTINE: EJETIM.FTN

FUNCION:

Elaborar el eje X a partir de un origen relativo en base al número de periodos que se tendrán sobre el eje, - se tiene un valor mínimo entre marcas que es de $\emptyset.68$ pulg-- das, es en base al tipo de banco que se marcan sobre dicho - eje X las marcas e identificadores se tiene tres tipos de -- bancos: Anual, Trimestral y Mensual para los cuales se ten-- drá sobre el eje las siguientes marcas e identificadores:

Anual; pequeñas marcas y abajo de estas un número que identi-- fica a cada año, Trimestral; pequeñas marcas y abajo de es-- tas una letra que identifica el Trimestre y a su vez abajo - de cada cuatro Trimestres se encuentra el año a que pertene-- ce, Mensual; es similar al Trimestral sólo que en este caso se tiene una letra que identifica cada mes y cada doce meses abajo del identificador se tiene el año, para crear el eje X se cuenta con toda la información necesaria en los parámetros

que recibe esta subrutina, cabe hacer la aclaración que el tamaño del eje X es tan grande como lo indique el tipo de banco y el número de periodos y que el hecho de que se ajuste a un subárea delimitada, está en función del factor de - escalamiento que es calculado por la rutina que invoca a esta. En esta rutina se hace uso de rutinas del paquete VER-SAPLOT.

PARAMETROS:

TIPBCO (ENTERO)	Contiene el tipo de banco que se va a ocupar en la gráfica (E).
NPERI (ENTERO)	Contiene el número de periodos que se van a marcar sobre el eje más uno de olgura (E).
IAI (ENTERO)	Año inicial a partir del cual se marcará el eje (E).

IPI (ENTERO) Periodo inicial a partir del cual se -
marca el eje (E).

AIO (ENTERO) (40) Arreglo que contiene los diferentes años
que se ocuparán en el eje (no incluye el
inicial) (capacidad de 40 años) (E).

ARCHIVOS:

Ninguno

INVOCADO POR:

TOMDE

INVOCA A:

(RUTINAS VERSAPLOT: PLOT, SYMBOL)

SUBROUTINA: EJE OBS

FUNCION:

Esta rutina se encarga de trazar el eje de referencia para las ordenadas de la gráfica, el cual corresponde a las observaciones de la serie de tiempo.

Para esto, debe localizar los valores numéricos que corresponden a cada pulgada y detectar cuántos dígitos necesita para representar el valor máximo, que debe aparecer acotado.

En base al número de dígitos y al espacio disponible para anotarlos, se calcula el tamaño de los caracteres.

Si los números que serán acotados, son mayores que uno, se utilizan dos decimales, y si son menores que -- uno, se usan cuatro decimales.

PARAMETROS:

MINIMO: Es una variable real que contiene el -
valor mínimo a graficar para todas las
series (E).

DVAL: Es una variable real que indica la rela-
ción que se obtiene entre el número de
unidades no escaladas por unidad real -
de medida. (E).

FACT: Variable de tipo real que indica el fac-
tor de escalamiento para toda la gráfi-
ca (E).

COMMON /GR2/

ABX, ORDY: Son los valores de las longitudes co -
rrespondientes al eje de las abscisas y
de las coordenadas.

ARCHIVOS:

Ninguno

LLAMA A LAS RUTINAS:

ERR: Se encarga de enviar los mensajes de error en caso de que se detecte alguno al momento de ejecución.

NUMBER: Imprime variables numéricas con un formato dado. (RUTINA VERSAPLOTH).

PLOT: Traza líneas o mueve la "Pluma" a --- cierta posición (RUTINA VERSAPLOTH).

ES LLAMADA POR LAS RUTINAS:

TOMDE: Se encarga de controlar la graficación de todas las series de tiempo.

OBSERVACIONES:

Ninguna

SUBROUTINA: POLI

FUNCION:

Se encarga de controlar la elaboración del Polígono Especial, llamado así para diferenciarlo del tipo de gráfica que proporciona el proveedor.

El Póligono Especial está formado únicamente por símbolos y no utiliza líneas rectas para unirlos entre sí.

Esta rutina envía los valores del fenómeno a graficar a otra rutina auxiliar que une de dos en dos los puntos correspondientes.

PARAMETROS:

VEC: Arreglo de números reales que contiene los valores del fenómeno a graficar (E).

TIM: Arreglo de números reales que contiene los valores de la variable independiente o tiempo (E).

LONG: Variable entera que indica el número de datos a graficar (E).

ISY: Número de símbolo que será empleado para la graficación. (E).

FACT: Factor de escalamiento aplicado en el -- campo (E).

ARCHIVOS:

Ninguno

INVOCA A:

SEGARR

INVOCADA POR:

TOMDE

OBSERVACIONES:

Es importante aclarar que en los vectores VEC y TIM, se tienen en seguida de los valores a graficar, los parámetros de escalamiento necesarios. (FVAL y DVAL).

SUBROUTINA: SEGARR

FUNCION:

Une dos puntos por medio de un conjunto de símbolos alineados. Para lograr esto, se calcula la distancia -- real entre los valores ya escalados.

Y se coloca los caracteres con un tamaño fijo uno tras otro a una distancia preestablecida. Sin embargo, como las dimensiones son afectadas también por un factor general de reducción, la distancia y el tamaño de los símbolos depende del -- tamaño del campo definido.

PARAMETROS:

VECl: Ordenada no escalada del punto final que será unido por los símbolos alineados -- (E).

TIMl: Absisa no escalada del punto final que -- será unido por los símbolos alineados (E).

IK: Índice que indica el número de elemento a procesar (E).

NSY: Número entero que identifica el símbolo que será empleado como elementos de la línea (E).

FACT: Factor de escalamiento del campo.

ARCHIVOS:

Ninguno

INVOCA A:

SYMBOL (Versaploth)

INVOCADA POR:

POLI

OBSERVACIONES:

Nótese que cuando se envía el primer punto a graficar, no se imprime realmente ningún símbolo.

La rutina conserva los valores de la invocación anterior, -- asignándose los valores a variables locales y sin embargo, el valor no se pierde debido a que esta rutina se encuentra en el mismo módulo objeto que la rutina POLI.

SUBROUTINA: HISTO.FTN

FUNCION:

Ejecutar la Gráfica de Histograma, esto se ejecuta en base al tipo de ashurado dado por el usuario o por default se asigna el primer ashurado del catálogo del Manual de Usuario, se calculan los incrementos en el tiempo y al FVAL, esto es para el vector de observaciones y tiempos.

PARAMETROS:

NASH: (entera) tipo de ashurado (entrada)
PER: (entera) periodicidad entre puntos dada por eje por (entrada).
DVAL: (real) DELTA VALIU proporcionado por eje y (entrada).

COMMON/GR2/,/GR3/

ARCHIVOS:

INVOCADA POR: TOMDE

INVOCA A: PLOT, TONE

SUBROUTINA IMPUL.FTN

Función:

Ejecutar la gráfica de impulsos esto se ejecuta en base al tipo de simbolo dado por el usuario, se calculan los incrementos para el vector de observaciones y tiempos.

PARAMETROS:

NSY - (entera) número de símbolo (E)

PER - (entera) periodicidad entre los puntos dada por eje X(E).

DVAL (entrada) DELTA, VALIU proporcionada por eje Y (E).

COMMON/GR2/,/GR3/

ARCHIVOS:

INVOCADA POR: TOMDE

INVOCA A: PLOT, SYMBOL,

SUBROUTINA: LISDES

FUNCION:

Se encarga de elaborar un listado que muestre la información contenida en el archivo Descriptor de la Gráfica. Para lograr esto, el "Descriptor" ya debe estar catalogado.

Al crear el listado, la rutina indica por medio de textos, cuál es la información correspondiente - - a cada área.

Debe recordarse que para acceder la información de cada área de los distintos campos, se utiliza un conjunto de apuntadores que se localiza al inicio de cada "Descriptor", abarcando desde el registro dos hasta el -- trece.

PARAMETROS:

NCMBRE: Variable alfanumérica que contiene el nombre del descriptor que será listado (E).

ARCHIVOS:

* . GRF Archivos Descriptores

INVOCA A:

Ninguno

INVOCADA POR:

Programa Principal DEFINE

5.5 MODIFICACION DE GRAFICAS

PROGRAM MODIF.FTN

FUNCION:

Controlar la etapa de modificación de un archivo descriptor de una gráfica, para lo cual como primer paso solicita el nombre de la gráfica checa que esté catalogada, -- checa también si lo que se tecló fue un HELP en cuyo caso - invocaré a la rutina que da información de orientación, una vez capturado el nombre de la gráfica a modificar hace uso - de rutinas tales como: la que crea un archivo de trabajo --- (EDIWORK.TMP) que es imagen del descriptor de la gráfica, la rutina que pone en modo editor al archivo descriptor, y la - rutina que efectúa la función inversa a la que crea el archivo de trabajo.

PARAMETROS:

Ninguno

ARCHIVOS:

Ninguno

INVOCADO POR:

Ninguna

INVOCA A:

HELP

VALFIL

FALLA

IMAGEN

EDIT

RGRESA

SUBROUTINA: HELP.FTN

FUNCION:

Proporcionar al usuario información de orientación en forma general del módulo en que se encuentre (DEFINICION, MODIFICACION, EJECUCION) o de algún sub-módulo, cuando se solicite se proporcionará la explicación de las instrucciones comandos o directivas con que se cuenta en los diferentes módulos o submódulos.

PARAMETROS:

IPAR (Entera) apuntador al registro que contiene el inicio de la información solicitada. (E).

TEX (Logical) (70) cadena que contiene la información requerida, ésta será desplegada en pantalla.

ARCHIVOS:

HELPGRAF.DAT

INVOCADA POR: MODIF

INVOCA A: Ninguna

SUBROUTINA: VALFIL,FTN

FUNCION:

 Checar que exista el descriptor de la gráfica en el directorio de graficación, el cual será único para todos los usuarios del Sistema GRAFOS, esta subrutina checa fundamentalmente tres cosas; el nombre que identifica un descriptor, la familia de cuentas a la que pertenece y el número de miembro, estos elementos son grabados al dar de alta un descriptor, esta rutina checa si los tres elementos mencionados se cumplen en caso afirmativo se enciende una bandera de lo contrario se apaga. La causa de tener en el archivo directorio los tres elementos se debe a que, al ser un directorio único para los usuarios del Sistema GRAFOS es muy factible que se repita el nombre de un archivo descriptor de gráficas, al tener estos tres elementos la única restricción que existe es que un mismo usuario no puede tener dos archivos descriptors con el mismo nombre.

PARAMETROS:

DESCFIL (REAL *8) Contiene el nombre del archivo des-
cripto de gráficas (E).

EXISTE (ENTERO) Bandera indicadora de existencia o
inexistencia de un archivo descrip-
tor de gráficas (S).

ARCHIVOS:

DIRECT.GRF,3 Unidad No. 1
 Acceso Secuencial

INVOCADO POR:

MODIF

INVOCA A:

Ninguna

SUBROUTINA: FALLA

FUNCION:

Envía a la pantalla mensajes de error al momento de ejecución de una gráfica. Los textos a desplegar los extrae del archivo de mensajes que ha sido creado precisamente para dicho fin. Para identificar cada error se utiliza un número entero que corresponde con el número de registro donde se ubica el texto a desplegar.

PARAMETROS:

NUMERR: Variable entera que indica el número de error a desplegar (E).

ARCHIVOS:

/ERROR.DAT/UNID=1/ACC.DIR/

Archivo que contiene los mensajes de error.

INVOCA A:

Ninguna

INVOCADA POR:

MODIF

EDIT

CAMBIA

BORRA

NPRI

SUBROUTINA: IMAGEN

FUNCION:

Se encarga de crear una copia del Descriptor de una Gráfica en una área de trabajo que permite realizar las funciones de edición en una forma más eficaz.

El área de trabajo para edición, se ajusta a una estructura de datos típica conocida como: "Lista Circular Doblemente Ligada".

Dicha estructura facilita las funciones de borrado, e inserción de nuevos elementos y también agiliza la búsqueda de registros vacíos.

Existe una pequeña variante, con respecto a una lista ligada tradicional. Esto es que, el campo correspondiente al contenido del elemento y las ligas del mismo, se encuentran separados físicamente.

Por un lado, el contenido de los elementos se aloja en un archivo de acceso directo, teniéndose un registro del archivo por cada elemento. Por otro lado, las ligas están almacenadas en una matriz bidimensional, la cual tiene tantos renglones como registros existen en el archivo. Y por último, existe una correspondencia biunívoca entre los renglones de la matriz y los registros del archivo.

PARAMETROS:

NOMDES: Es el nombre de descriptor de la gráfica (E).

LIGA: Matriz bidimensional que contiene las ligas hacia adelante y hacia atrás de la lista (S).

INVOCA A:

DAMELI

METELI

INVOCADA POR:

MODIFI

SUBROUTINA: METELI

FUNCION:

Se encarga de insertar un elemento en la lista doblemente ligada que funciona como área de trabajo.

Esto se hace conservando siempre la lista como circular, es decir, la última liga hacia adelante debe --- apuntar al inicio o cabeza de la lista y la cabeza de la - lista, en su liga hacia atrás siempre debe apuntar al último elemento de la lista.

PARAMETROS:

DONDE: Apuntador al registro que se está procesando (E).

LIBRE: Apuntador al registro libre en donde se insertará un nuevo elemento (E).

NOCAMP: Número de campo al que pertenece el registro (E).

NOAREA: Número de área a la cual pertenece el registro (E).

LINEA: Arreglo de caracteres que serán grabados como contenido del registro.

COMMON:

/LIGAS/ LIGA: Arreglo de apuntadores que conforman las ligas de la lista (S).

ARCHIVOS:

EDIWORK.TMP

INVOCA A:

Ninguna

INVOCADA POR:

IMAGEN

SUBROUTINA EDIT.FTN

FUNCION:

Simular un EDITOR con los siguientes comandos de edición.

- Inserta (I)
- Cambia (C)
- Localiza (L)
- Imprime (P)
- Atrás (A)
- Bottom (B)
- Borra (D)
- Reemplaza (R)
- Top (T)
- Imprimir una línea determinada (NP)

Comandos de salida del editor

- ED
- CD
- EX
- SP

Para los comandos de edición se cuenta con una rutina especializada para cada caso.

PARAMETROS:

COMMON/LIGAS/LIGA (ENTERA) (100,2) matriz que funciona como ligas del archivo de trabajo.

LIS (ENTERA) bandera indicadora, si la salida del editor fue por SPOOL (SP) (S).

ARCHIVOS:

INVOCADA POR: MODIF

INVOCA A: FALLA, CARY, INSERT, CAMBIA, LOCALI,
PINTA, NUMER, BORRA, REMPL, ADPRIN, NPRI.

SUBROUTINA CARY.FTN

FUNCION:

Al oprimir la tecla de RETURN se imprime la línea siguiente de la que se encuentre posicionado en ese momento, y se actualiza apuntador de líneas.

PARAMETROS:

LIN (ENTERA) apuntador de línea (E/S)
COMMON/LIGAS/LIGA (ENTERA) (400,2) matriz de ligas del archivo de trabajo.

OBSERVACIONES:

El apuntador LIN al entrar esta rutina apunta a la línea antes del RETURN al salir apunta a la línea que se imprimió. En caso de un RETURN en el cual la siguiente línea ya es diferente - - área o campo, se avisa al usuario y no se imprimé ni actualiza al apuntador.

ARCHIVOS: EDIWORK.TMP (Archivo de trabajo)

INVOCADA POR: EDIT

INVOCA A:

SUBROUTINA INSERT.FTN

FUNCION:

Ejecutar el comando de insercción, cuando la insercción se - -
quiera efectuar después de terminada una área o un campo se in
forma al usuario y se solicita defina donde quiera insertar.

PARAMETROS:

LIN (ENTERA) apuntador de línea.

VER (LOGICAL) (72) cadena con información. (E)

COMMON/LIGAS/LIGA (400,2) matriz de ligas del archivo de tra--
bajo.

ARCHIVOS: EDIWORK.TMP (Archivo de Trabajo)

INVOCADA POR: EDIT

INVOCA A:

SUBROUTINA: CAMBIA

FUNCION:

Se encarga de ejecutar el comando cambia del - proceso editor. Esto es, cambiar una cadena de caracteres que ya existe en el texto por una nueva cadena de caracte-- res.

Como puede verse, al menos una de las cadenas debe ser no vacía, ya que de otra forma, el comando no sería válido. Además, es posible asignarle un número, con lo cual el cambio se realiza repetidas veces.

Lo primero que debe hacer el comando, es localizar la cadena 1, con lo cual conoce la posición inicial de ésta, y a partir de esta posición copiará el nuevo texto y por último agrega el ya existente.

Si la línea tratará de exceder de setenta y dos caracteres, se realiza un truncamiento por la derecha.

PARAMETROS:

DONDE: Es el apuntador que indica cuál es la línea del texto, en la cual está posicionado el usuario (E).

COMAND: Es la línea que contiene el comando teclado por el usuario (E).

ARCHIVOS:

EDIWORK.TMP

*.GRF

ARCHIVOS:

EDIWORK.TMP

INVOCA A:

LOC

FALLA

INVOCADA POR:

EDIT

SUBROUTINA: LOC

FUNCION:

Se encarga de localizar a lo largo del archivo EDIWORK.TMP la existencia de una cadena de caracteres.

La búsqueda la realiza siempre de la línea actual en adelante.

En caso de no localizar la cadena enciende una bandera.

Cabe aclarar que esta rutina no despliega en la pantalla la línea que contiene el texto sino que, solamente realiza la búsqueda interna.

PARAMETROS:

DONDE: Indica en qué línea está posicionado el usuario (E/S).

EXITO: Variable lógica que indica si se localizó o no la cadena (S).

INICIO: Apunta al inicio de la cadena por localizar (E).

FINAL: Apunta al último caracter que debe localizarse (E).

FIN3: Indica cual es la posición donde se inicia la cadena localizada (S).

LINEA: Es la línea que contiene la cadena por ser localizada (E).

ARCHIVOS:

EDIWORK.TMP

INVOCA A:

Ninguna

INVOCADA POR:

LOCALI

CAMBIA

SUBROUTINA: LOCALI

FUNCION:

Controla la ejecución del comando: LOCALIZA.

Este subprograma no realiza la búsqueda en si del texto sino que, se vale de una rutina auxiliar.

En caso de localizar el texto imprime en la pantalla la línea que lo contenga. De lo contrario se llega al fin del archivo.

Si la cadena a localizar está vacía, se omite un mensaje de error, por medio de otra rutina auxiliar.

PARAMETROS:

DONDE: Apuntador a la línea que aparece en la pantalla en ese momento (E/S).

LINEA: Arreglo que contiene el comando tecleado por el usuario (E).

ARCHIVOS:

EDIWORK.TMP

Pantalla

INVOCADA A:

FALLA

LOC

INVOCADA POR:

EDIT

SUBROUTINA: PINTA

FUNCION:

Se encarga de imprimir en la pantalla las líneas del archivo de trabajo que le sean indicadas, ya sea hacia adelante o hacia atrás, de acuerdo con el signo del número de líneas a imprimir.

Cuando el número de líneas es positivo la rutina se vale de las ligas hacia adelante de la "LISTA". Si el número es negativo entonces se usan las ligas hacia atrás.

Si la impresión de líneas tratara de rebasar el final del archivo, se envía a la pantalla el indicador de dicha condición: [* EOF *]

Si se tratara de rebasar hacia atrás el inicio del archivo también se envía a la pantalla el indicador correspondiente: [* TOF *]

PARAMETROS:

DONDE: Apuntador a la línea que aparece en la pantalla (E/S).

LINEAS: Número de líneas a imprimir (E).

NP: Bandera que indica si todas las líneas se imprimen o sólo la última (E).

COMON /LIGAS/

LIGA: Arreglo bidimensional que funciona como ligas dobles del área de trabajo (E).

ARCHIVOS:

EDIWORK.TMP

Pantalla

INVOCADA A:

Ninguna

INVOCADA POR:

EDIT

NPRI

SUBROUTINA: BORRA

FUNCION:

Ejecuta el comando "DELETE", el cual elimina - un cierto número de líneas contando hacia adelante a partir de la línea que aparece en la pantalla.

Si el número fuera negativo, se genera un mensaje de error.

En caso de que el número de líneas por borrar exceda el límite inferior del área de trabajo aparecerá el identificador correspondiente [*** EOF ***] y en ese momento - termina la ejecución del comando.

Para realizar la eliminación de líneas, se --- efectúa un seguimiento de las ligas hacia adelante del --- área de trabajo. Y para indicar que una línea está vacía, basta con asignar valor de cero a las ligas de la misma, y no será necesario borrar la información del registro.

PARAMETROS:

DONDE: Apuntador a la línea que aparece en la pantalla en ese momento.

LINEA: Arreglo de caracteres que contiene el comando tecleado por el usuario.

AREAS COMUNES:

/LIGAS/LIGA: Arreglo bidimensional que contiene las ligas hacia adelante y hacia atrás del área de trabajo.

ARCHIVOS:

EDIWORK.TMP

Pantalla

INVOCA A:

FALLA

INVOCADA POR:

EDIT

SUBROUTINA: REMPL

FUNCION:

Se encarga de ejecutar el comando **REPLAZA**, el cual sustituye el texto de la línea que aparece en la pantalla, por el texto asociado al comando.

Para esto, simplemente se captura la línea teclada por el usuario, la cual contiene el nuevo texto y se graba en el registro apuntado por el programa en ese momento.

Aún en el caso de que el texto sea una línea con blancos o vacía, el registro no es borrado sino que será llenado con blancos.

PARAMETROS:

DONDE: Variable entera que apunta a la línea procesada en ese momento (E).

LINEA: Arreglo de caracteres que contiene el
comando teclado por el usuario (E).

ARCHIVOS:

EDIWORK.TMP

INVOCA A:

Ninguna

INVOCADA POR:

EDIT

SUBROUTINA: ADPRIN

FUNCION:

Se encarga de agregar al final de la línea el texto asociado al comando.

Si el primer caracter del texto es un blanco, este es ignorado.

Es importante aclarar que el final de la línea es considerado como el primer caracter distinto de blanco de derecha a izquierda.

PARAMETROS:

DONDE: Apunta a la línea que aparece en la --
pantalla en ese momento (E).

LINEA: Arreglo de caracteres que contiene el
comando tecleado por el usuario (E).

ARCHIVOS:

EDIWORK.TMP

INVOCA A:

Ninguna

INVOCADA POR:

EDIT

SUBROUTINA NPRI.FTN

FUNCION:

Ejecutar el comando por medio del cual apunte e imprima una --
línea, ya sea anterior o posterior a la que se muestre, ésto -
se logra al recibir (NP+ n).

PARAMETROS:

LIN (ENTERA) apuntador de línea (E)
VER (LOGICAL) (72), cadena con información (E)
COMMON/LIGAS/LIGA matriz de apuntadores al área de trabajo.

ARCHIVOS:

INVOCADA POR: EDIT

INVOCADA A: PINTA, FALLA

SUBROUTINA; RGRESA

FUNCION:

Se encarga de elaborar un archivo que simule - el "Diálogo" interactivo con la tarea DEFINE para elaborar un "DESCRIPTOR DE GRAFICA". Este archivo se crea a partir del área de trabajo utilizada en la modificación de un --- "DESCRIPTOR". Dicho de otra forma, por medio del archivo que simula la interacción con el usuario, se "RE-DEFINE" - el descriptor modificado.

Para esto, es necesario tomar en cuenta las convenciones empleadas en el módulo de definición, que resumidas son:

- PARA CAMBIAR DE AREA SE PROPORCIONA UNA LINEA VACIA.

- PARA CAMBIAR DE CAMPO TAMBIEN SE PROPORCIONA UNA LINEA VACIA.

- EL NUMERO MAXIMO DE PARAMETROS ALFABETICOS SON DIEZ.
- EL NUMERO MAXIMO DE PARAMETROS NUMERICOS SON DIEZ.

Por último, recuérdese que el área de trabajo usada en la modificación, corresponde a la estructura denominada "LISTA CIRCULAR DOBLEMENTE LIGADA".

PARAMETROS:

LIGAS: Arreglo de números enteros que son usados como apuntadores o ligas del área de trabajo (E) 300 X 2 I * 2.

ARCHIVOS:

/EDIWORK.TMP/UNID=4/ACC.DIR/

Area de trabajo que corresponde a una imagen del descriptor a modificar.

/TEMPO.TMP/UNID=6/ACC.SEC/

Archivo de salida que es usado para redefinir el descriptor de la gráfica.

INVOCA A:

Ninguna

INVOCADA POR:

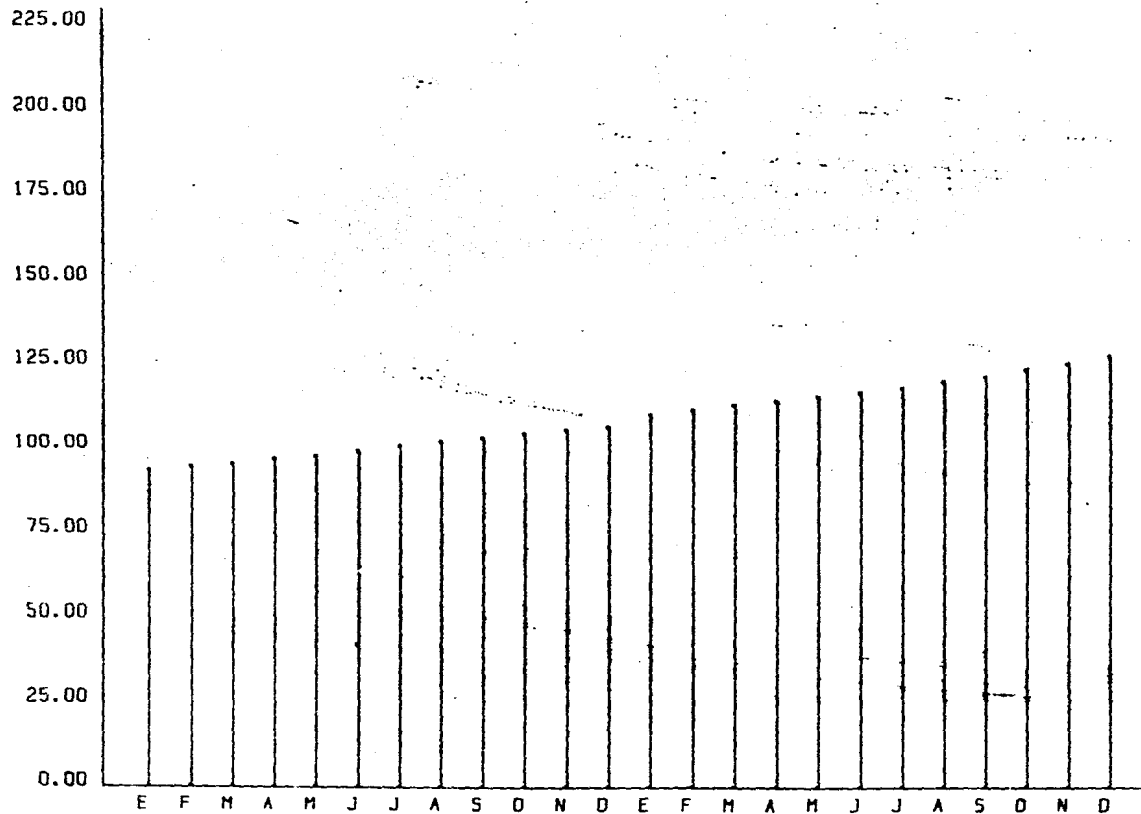
MGDIF

6. EJEMPLOS

En este capítulo se muestran los ejemplos del Paquete GRAFOS obtenidos a través del Graficador Electrostático VERSATEC 1200-A, en éstos se pueden apreciar los distintos tipos de Gráficas que pueden obtenerse. Posteriormente, aparecen dos ejemplos más completos en los que se puede observar el uso de distintas áreas o campos, así como el empleo de reportes Numéricos.

Todos los ejemplos van acompañados de un listado del archivo descriptor de cada Gráfica.

EJEMPLO PARA DEMOSTRACION DE GRAFICA DE IMPULSOS



TIPCAN: 6

DEFINICION DE CAMPOS:
V=20,0-20?0-20,0-20:11A=50??

NO. CAMPO: 1 APUNTAJES: 14 1 14 2 14 3 14 58 14 59 14 60 14 61 14 88

POR. ALF.

POR. NUM.

ENCABEZ.
1, EJEMPLO PARA DEMOSTRACION DE GRAFICA DE IMPULSOS??

ACOTACI.

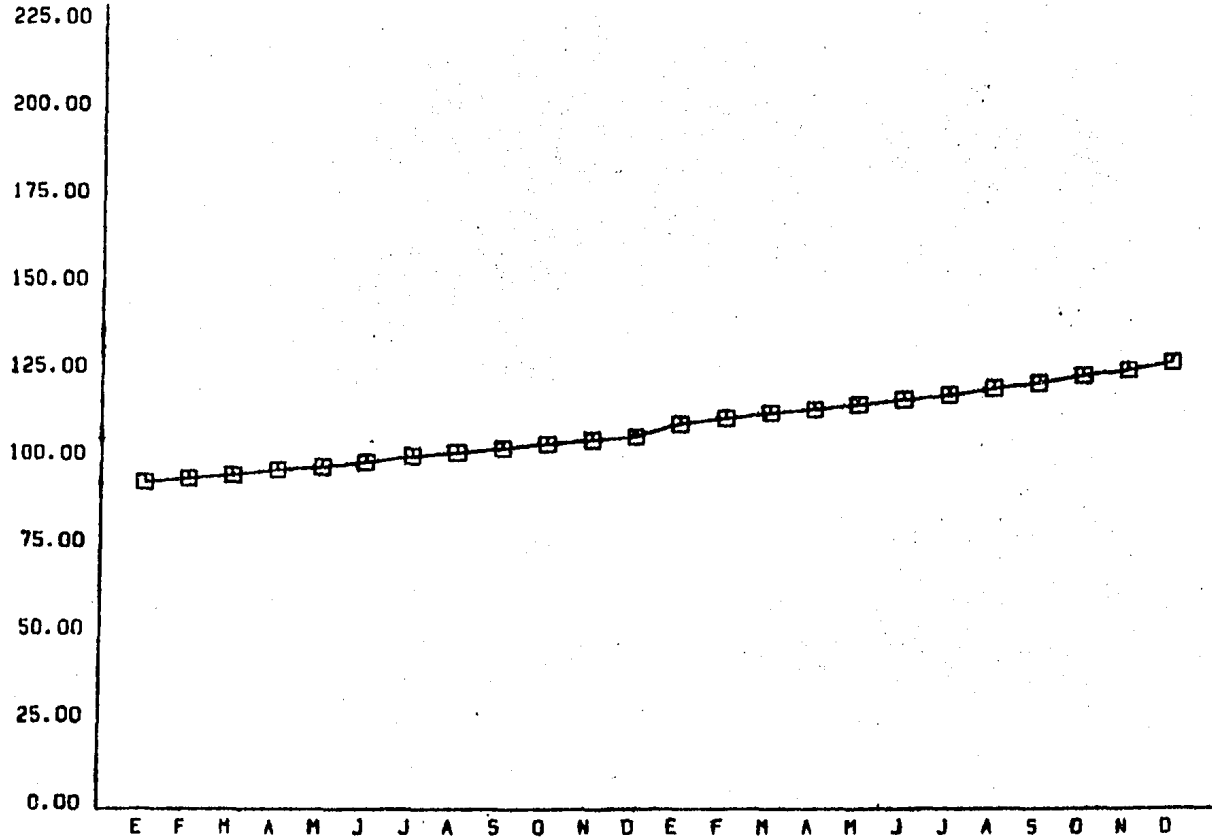
PIE PAG.

DEL. POS.

DET. DAT.

BANCO, 2, 78, 1-79, 12:1:85

EJEMPLO PARA DEMOSTRACION DE GRAFICA: POLIGONO VERSATEC



TIPO: G

DEFINICION DE CAMPOS:
C-20,0-200-20,C-20:LIN=50??

NO. CAMPO: 1 FUENTES: 14 1 14 2 14 13 14 65 14 66 14 67 14 68 14 93

PAR. ALF.

PAR. NUM.

ENCABEZ.
1, EJEMPLO PARA DEMOSTRACION DE GRAFICA POLIGONO VERSATEC??

ACOTACI.

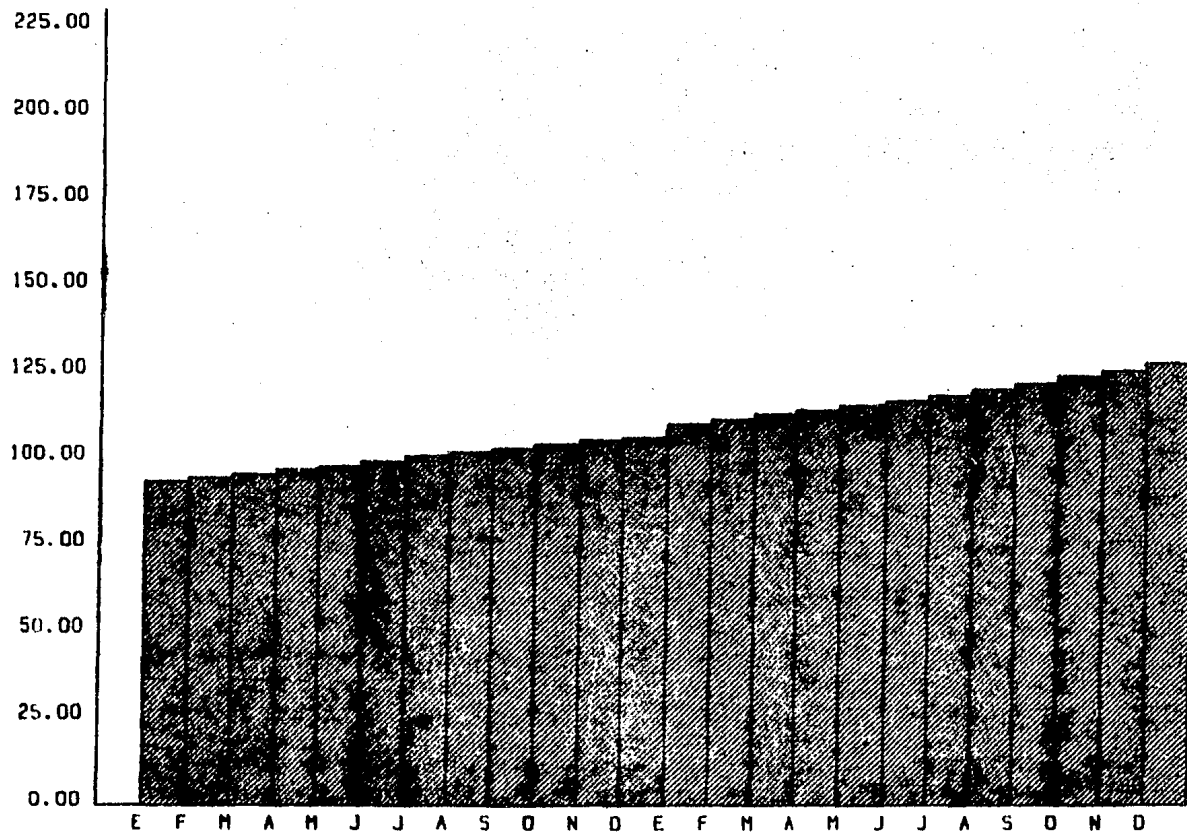
PIE PAG.

DET. POS.

DET. DAT.

EANCO3,2,76.1-79.12:V:1??

EJEMPLO PARA DEMOSTRACION DE : HISTOGRAMA



TIPCAM: 6

DEFINICION DE CAMPOS:
0-26,0-20? -20,0-20:11A=50??

NO. CAMPO: 1 SEPTIADRES: 14 1 14 2 14 3 14 51 14 52 14 53 14 54 14 79

PAR.ALF.
?

PAR.NUM.
?

ENCABEZ.
1, EJEMPLO PARA DEMOSTRACION DE : HISTOGRAMA??

ACOTACI.

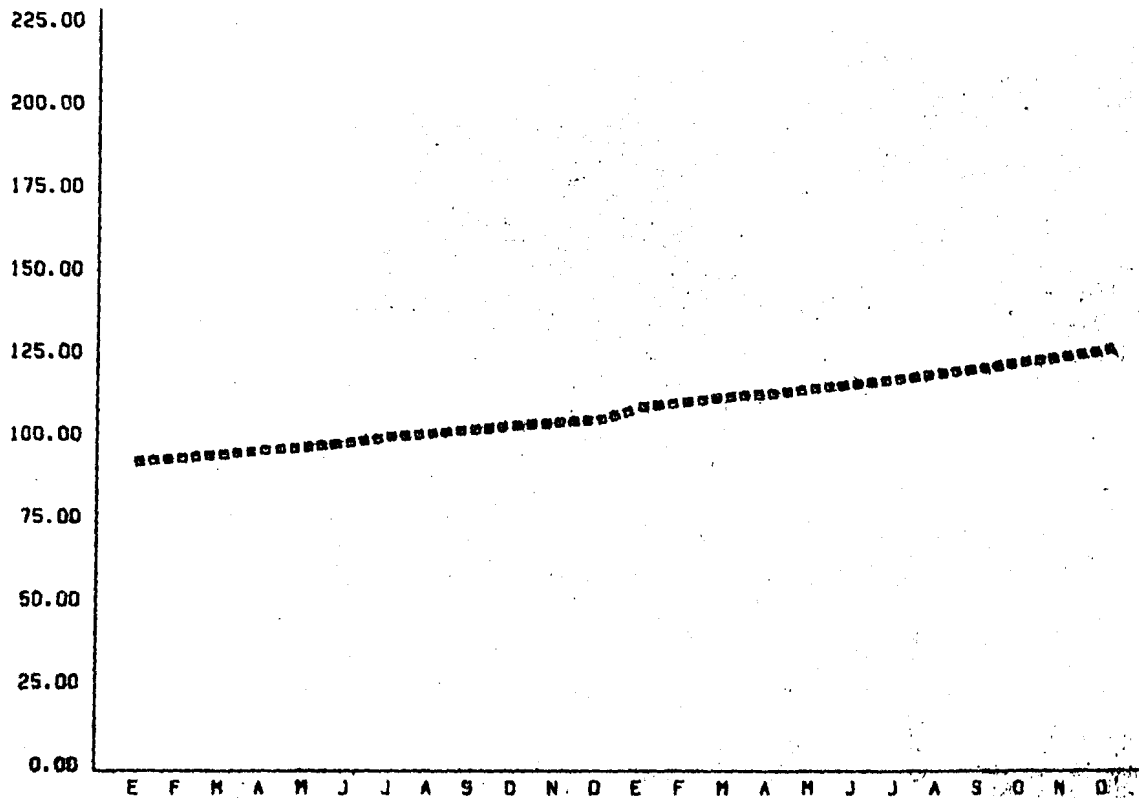
PIE PAG.

DET.POS.

DET.DAT.

BANCOS,2,78.1-79,12:H:1??

EJEMPLO PARA DEMOSTRACION DE POLIGONO



TIPCAM: G

DEFINICION DE CAMPOS:

U-26, U-2870-26, C-29:LIN=50??

NO. CAMPO: 1 APUÑTADRES: 14 1 14 2 14 3 14 47 14 48 14 49 14 50 14 75

PAR. ALF.

PAR. NUM.

ENCABEZ.

1, EJEMPLO PARA DEMOSTRACION DE POLIGONO??

ACOTACI.

PIE PAG.

LET. POS.

DLT. DAT.

BANCC3,2,78,1-79.12:8:1??

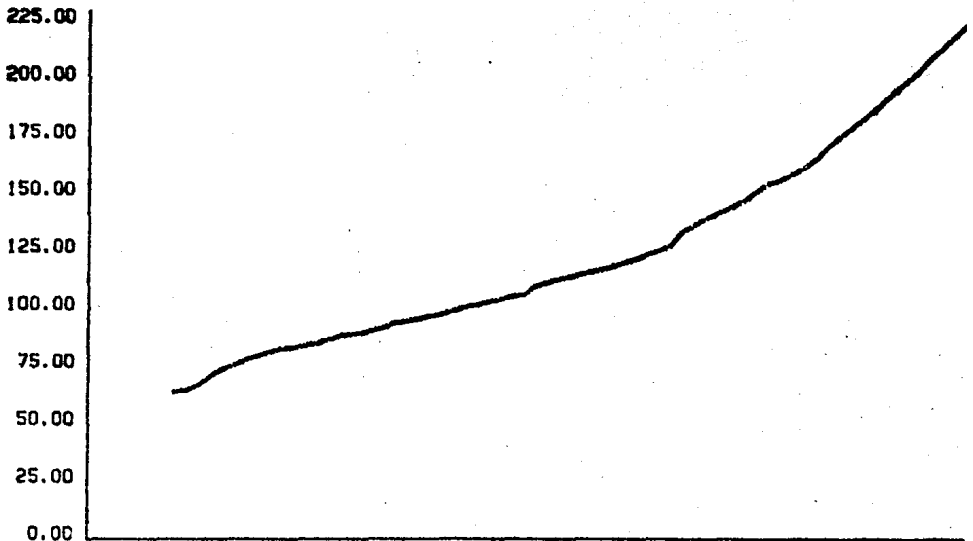
INDICE DE PRECIOS
AL CONSUMIDOR A NIVEL NACIONAL

SEMA 1961. I

SECTOR	1960 IN	1977	AGROS 1970	1979	1980	1981
ENERGIA	-	70,235	83,516	110,050	133,775	170,363
INDUSTRIAL	-	70,235	83,516	110,050	133,775	170,363
COMERCIO	-	82,368	84,893	113,146	136,881	178,388
SERVICIOS	-	82,368	84,893	113,146	136,881	178,388
MINAS	-	83,316	97,000	105,857	108,804	107,000
TRANSPORTE	-	84,308	85,250	118,337	137,380	178,388
ALIMENTOS	69,178	85,388	104,882	118,367	131,821	157,362
ALQUILER	69,178	85,388	104,882	118,367	131,821	157,362
SERVICIOS FINANCIEROS	69,178	85,388	104,882	118,367	131,821	157,362
DEPRECIACION	70,788	88,288	104,383	123,740	139,400	212,374
SERVICIOS DE COMUNICACION	73,598	88,237	103,385	125,338	151,334	217,338
OTROS SERVICIOS	75,829	91,488	104,389	127,584	158,428	222,388

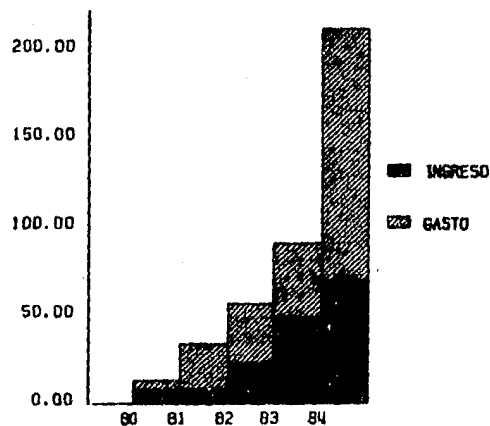
INFORMACION A JUNIO DE 1981 CON INFORMACION PARA 1976
ESTADISTICA DE MEXICO
CLASIFICACION DE INDICADORES NACIONALES U.S.E.

INDICE DE PRECIOS AL CONSUMIDOR
1976 - 1981



EFHAMJJASONDEFHAMJJASONDEFHAMJJASONDEFHAMJJASONDEFHAMJJASONDEFHAMJJASOND

TOTAL DE INGRESO-GASTO
AL MES DE ABRIL
(M.M.P.)



GRAFICA Y CUADROS PARA DEMOSTRACION

SITUACION ECONOMICA
AL MES DE ABRIL
(MILES DE MILLONES)

CONCEPTO	1980	1981	1982	1983	1984
CTA. CORRIENTE					
. INGRESOS	7.7	8.4	22.9	48.6	69.2
.GTOS. DE OP.	13.1	33.7	56.6	90.8	211.5
DEF O SUP DE OP.	-5.4	-25.3	-33.7	-42.2	-142.3
. TRANSFERENCIAS	7.7	15.0	20.0	46.8	122.6
. IMPUESTOS	-	-	-	-	-
DEF O SUP CTA C.	2.3	-10.3	-13.7	4.6	-19.7
CTA DE CAPITAL					
. INGRESOS	-	-	-	-	-
. TRANSF.	0.8	0.3	3.2	3.0	-
. GASTOS	-	-	-	-	-
AJENAS NETAS	-2.8	-1.2	-0.6	2.5	-3.4
DEF. O SUP TOT.	1.5	-10.0	-10.5	1.6	-19.7

GASTOS DE OPERACION
AL MES DE ABRIL
(MILES DE MILLONES DE PESOS)

CONCEPTO	1980	1981	1982	1983	1984
GASTOS DE OPER.	13.1	33.7	56.6	90.8	211.5
.SUELDOS Y SAL.	0.3	0.4	0.4	0.8	1.2
.ADQ. DE B. Y SERV.	9.6	25.2	48.8	63.8	158.3
.INTERESES	0.4	0.5	3.1	11.6	17.8
.OTROS	2.8	7.6	12.3	14.6	36.2

INGRESOS PROPIOS
AL MES DE ABRIL
(MILES DE MILLONES DE PESOS)

CONCEPTO	1980	1981	1982	1983	1984
ING. PROPIOS	7.7	8.4	22.9	48.6	69.2
.VTA. DE B. Y SERV.	7.6	7.9	22.2	46.8	65.4
- INTERNOS	7.6	7.9	22.2	46.8	65.4
- EXTERNOS	-	-	-	-	-
OTROS	0.1	0.5	0.7	2.6	3.8
ING. DE CAPITAL	-	-	-	-	-

B I B L I O G R A F I A

- DEFINICION DEL MODULO GENERADOR DE GRAFICAS: GRAFOS
(PRELIMINAR)
S.H.C.P. 1982

- DEFINICION DEL MODULO GENERADOR DE GRAFICAS: GRAFOS
S.H.C.P. 1982

- MODULO GENERADOR DE REPORTES; MALER
MANUAL DE USUARIO
S.H.C.P. 1981

- MODULO GENERADOR DE REPORTES; MALER
MANUAL TECNICO
S.H.C.P. 1982

- SISTEMA PROCESADOR DE INFORMACION ECONOMICA SPIE
MANUAL DE USUARIO
S.H.C.P. 1979

- VERSAPLOT
GRAPHICS PROGRAMING MANUAL
VERSATEC 1978

- VERSAPLOT
OPERATING MANUAL PDP-11 RSX-11M
VERSATEC 1980

- OPERATING MANUAL
C-PDP11/DMA/18BIT CONTROLLER (MODEL 121)
VERSATEC 1980

- BOLETIN MENSUAL DEL BANCO DE MEXICO
BANXICO 1981

- BOLETIN MENSUAL DE LA DIRECCION GENERAL DE PLANEACION HDARIA.
S.H.C.P. 1982

- ECONOMUNDO
REVISTA DE DISTRIBUCION MENSUAL
S.H.C.P. 1979

A P E N D I C E S

- A). MANUAL DE USUARIO; GRAFOS
- B). LISTADOS DE MENSAJES DE ERROR
- C). LISTADO DE AYUDAS
- D). SUBROUTINAS DE VERSAPLOT

APENDICE A

MANUAL DE USUARIO: GRAFOS

I N D I C E

PAGINA

1.	INTRODUCCION	A-1
2.	OBJETIVOS	A-2
3.	DESCRIPCION DEL SISTEMA GENERADOR DE GRAFICAS (GRAFOS)	A-3
3.1	DEFINICION	A-4
3.2	MODIFICACION	A-5
3.3	EJECUCION	A-6
4.	ZONAS DE UNA GRAFICA	A-11
5.	DEFINICION DE LAS ZONAS DEL AREA GRAFICA	A-12
5.1	COMANDOS DE DEFINICION	A-13
5.2	DEFINICION DE CAMPOS	A-20
5.3	PARAMETROS ALFABETICOS	A-22
5.4	PARAMETROS NUMERICOS	A-23
5.5	ENCABEZADOS	A-24
5.6	ACOTACIONES	A-25
5.7	PIE DE PAGINA	A-26

I N D I C E

PAGINA

5.8	LETRERO POSICIONADO	A-26
5.9	DETERMINACION DATOS	A-27
5.10	CATALOGO DE PATRONES	A-30
6.	MODIFICACION	A-31
6.1	COMANDOS DE MODIFICACION	A-32
7.	ACTUALIZACION DE PARAMETROS	A-38
8.	EJECUCION	A-39

1. INTRODUCCION:

La característica de los sistemas de cómputo comerciales que les permite éxito en el mercado, generalmente es la documentación -- que está a disposición del usuario, mediante la cual se podrá capacitar a otros usuarios en la operación del sistema, resolver sus problemas por error o falla y administrar, proteger y recuperar la información del sistema en forma autosuficiente y sin la dependencia de especialistas en cómputo.

Dichas características son requeridas por sistemas interactivos que están instalados en procesadores, cuya arquitectura y orientación de operación son para estar ubicados en el ambiente de trabajo del usuario y no en salas de cómputo especiales.

El manual de usuario que es utilizado para la capacitación de un nuevo usuario, permitirá en principio, el arraigo a corto plazo del sistema como herramienta de trabajo del usuario y permitirá un diálogo sencillo entre éste y los especialistas en cómputo para dar mantenimiento al sistema y para definir nuevas funciones que resultan de la evaluación misma de los organismos y de los sistemas.

1. OBJETIVOS:

Este documento describe el uso del Generador de Gráficas "GRAFOS", implantado en la Coordinación de Informática de la Secretaría de Hacienda y Crédito Público, como una herramienta más para análisis y presentación de información económica y estadística.

El Generador de Gráficas "GRAFOS" apoya a dos sistemas que existen y están en funcionamiento y que son:

- El Sistema Procesador de Información Económica (SPIE), el cual representa una infraestructura completa para el almacenamiento, manejo y tratamiento de información económica en forma de series de tiempo.
- El Generador de Reportes el cual permite la representación de la información almacenada por medio de S.P.I.E., ésta representación se efectúa en forma dinámica.
- Tanto el SPIE como el Generador de Reportes son sistemas interactivos que no requieren de personal especializado para su manejo, ni requieren del conocimiento de lenguajes de programación.

El Generador de Gráficas está creado para proporcionar independencia absoluta al usuario, ya que para su uso:

- Evita totalmente lenguajes de programación
- Permite la explotación ilimitada de información contenida en Bancos de Datos SPIE, así como de cuadros de MALER
- Permite que se definan, modifiquen y obtengan gráficas sin depender del personal de cómputo o dibujantes.

3. DESCRIPCION DEL SISTEMA GENERADOR DE GRAFICAS (GRAFOS)

El Sistema Generador de Gráficas GRAFOS, permite la explotación de la información almacenada por SPIE en Bancos de Datos, ésta información es representada por GRAFOS en forma de 4 diferentes tipos Poligono, Histograma, Impulsos y Versatec (caso particular del polígono), permite también la presentación de cuadros o reportes (Generados por MALER), da la flexibilidad de amplificar o reducir gráficas y/o reportes a placer, ésto es de gran utilidad ya que en una hoja se puede tener una gráfica con los datos representados a su vez en un cuadro o reporte.

Para generar o describir una gráfica son utilizadas una serie de instrucciones (Comandos) los cuales son almacenados en una estructura denominada Descriptor (alojado en un archivo), el cual es almacenado en memoria secundaria lo cual permite su conservación para futuras emisiones de la misma gráfica sin tener que repetir los comandos o instrucciones.

Es importante señalar que el Generador de Gráficas GRAFOS no modifica bajo ninguna circunstancia la información contenida de los Bancos de Datos de SPIE.

Un esquema simplificado de la forma como funciona GRAFOS se muestra en la siguiente figura.

3.1. DEFINICION:

La definición de gráficas, por parte del usuario, consiste en ir proporcionando uno a uno los comandos o instrucciones necesarios para obtener las gráficas que se deseen.

Por su parte GRAFOS efectúa las siguientes funciones:

- Ubicar al usuario, indicando la zona de la gráfica que está definiendo.
- Explicación (opcional) de la sintaxis de los comandos o instrucciones y las características de cada zona de las gráficas.
- Revisar que cada comando esté anotado correctamente de acuerdo a la sintaxis fijada. De lo contrario se despliega en pantalla un mensaje indicando el tipo de error cometido. Para lo cual el usuario puede repetir la línea errónea.
- Almacenar todos los comandos válidos en el DESCRIPTOR y catalogarlo.

MODIFICACION

Este módulo del Generador de Gráficas, permite al usuario - modificar sus gráficas, para lo cual es necesario modificar al DES CRIPTOR creado en la etapa de definición. Para lograr ésto, se cuenta con un conjunto de comandos de edición.

Las funciones de GRAFOS en el proceso de Modificación son:

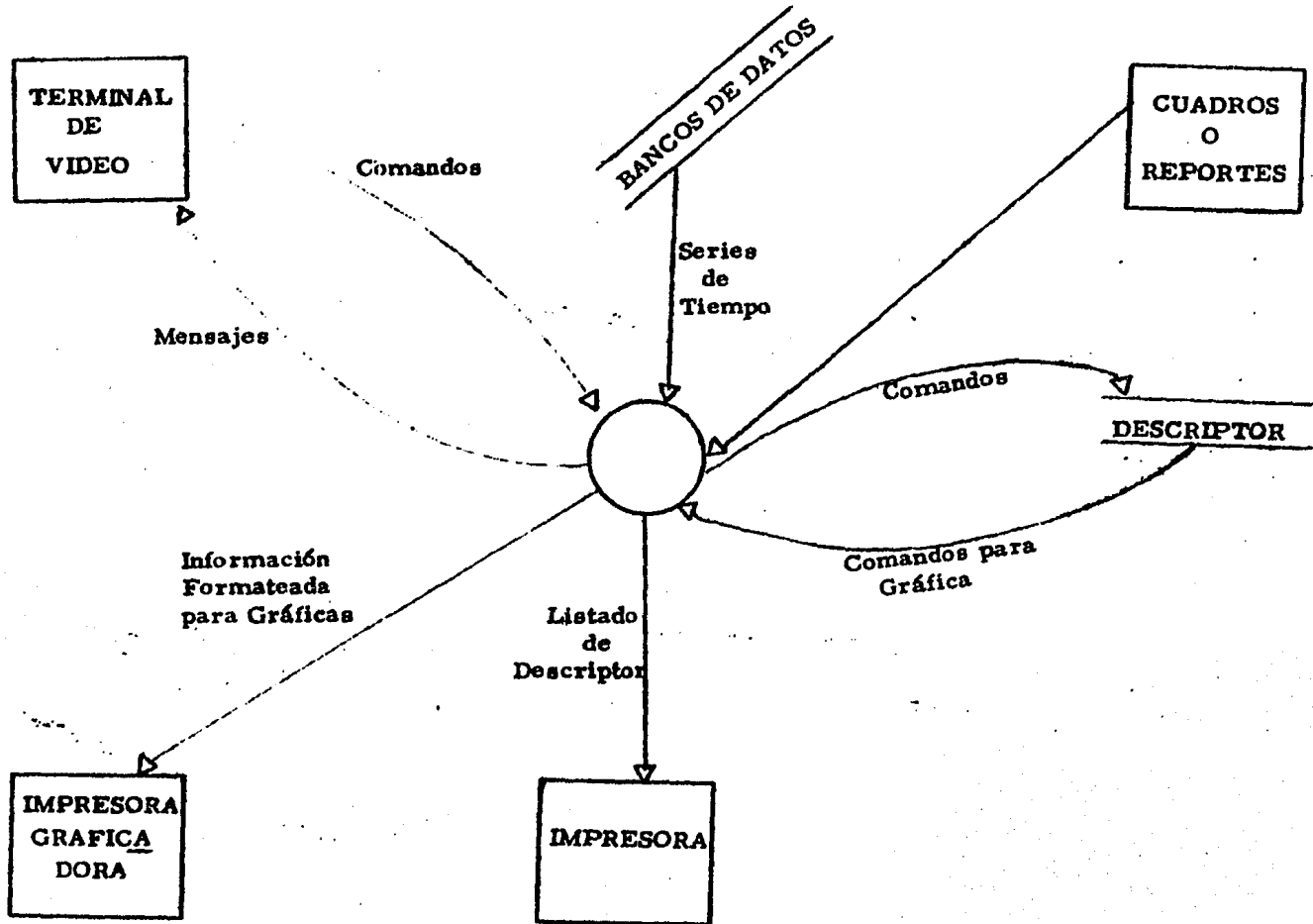
- Modificar el Descriptor (alojado en un archivo), de acuerdo a los comandos de edición que proporcione el usuario
- Ubicar al usuario, indicando las partes de las gráficas que - está modificando.
- Explicación (opcional) de los comandos de edición.
- Al finalizar el proceso de Modificación, se realiza el chequeo de la sintaxis de los comandos.

3.3. EJECUCION U OBTENCIÓN DE GRAFICAS

Este Módulo del Generador de Gráficas consiste en obtener en papel (del graficador electrostático VERSATEC) las gráficas, solicitadas por el usuario, para lo cual el sistema ejecuta cada uno de los comandos que el usuario proporcionó y que se encuentran almacenados en el Archivo Descriptor creado en el módulo de definición y/o modificado en el módulo de modificación.

En la figura de la siguiente página puede observarse las tres funciones del Generador de Gráficas GRAFOS.

CONFIGURACION DEL SISTEMA GENERADOR DE GRAFICAS GRAFOS



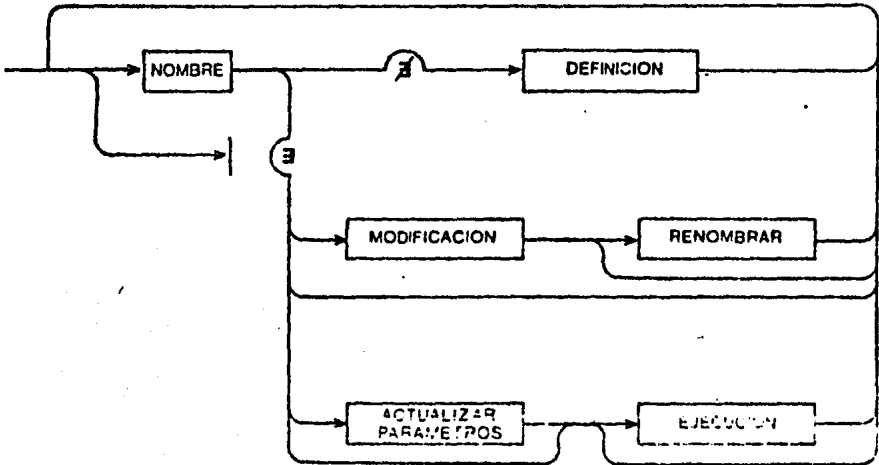
En el diagrama anterior se puede observar que el sistema verifica la existencia de un Descriptor que contiene los coman--dos para graficar (el nombre consta de 1 a 6 caracteres alfanumericos); si existe este nombre de descriptor se puede pasar a modifica--ción o ejecución en caso contrario se ingresa a definición.

GRAFOS esta dividido en tres procesos principales que son:

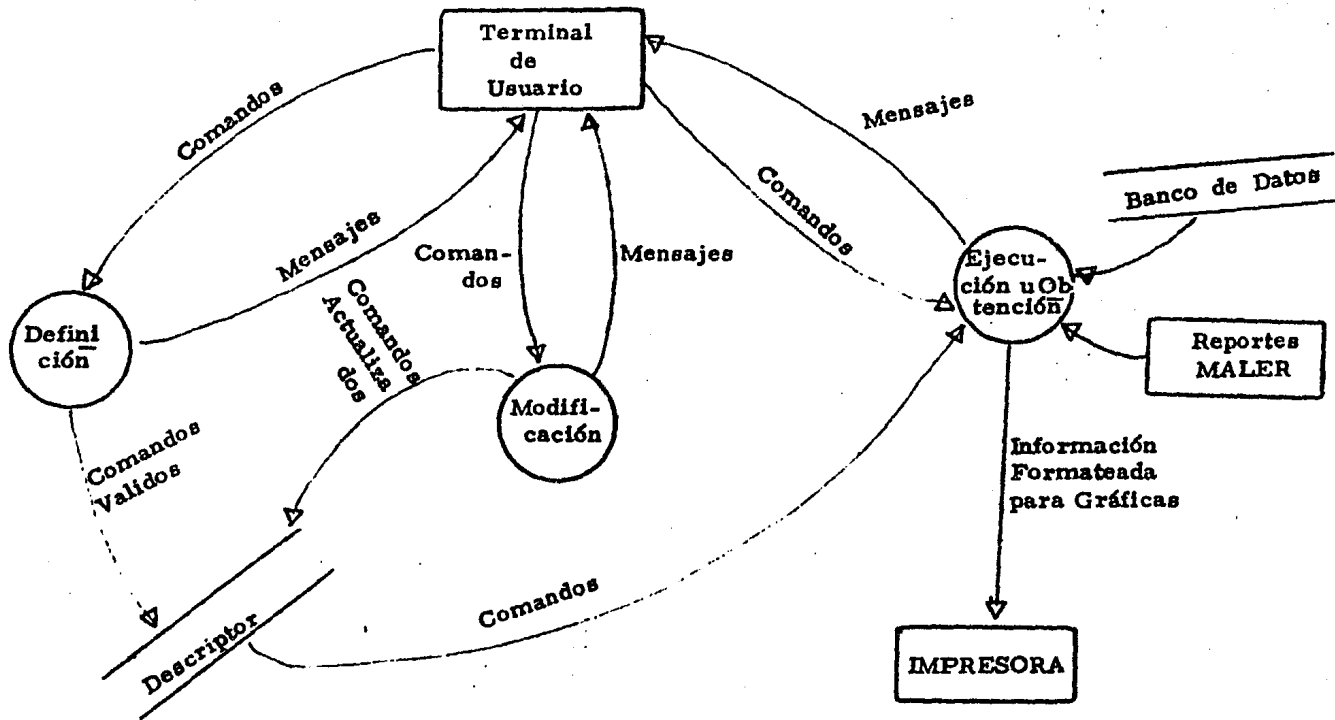
- DEFINICION DE GRAFICAS
- MODIFICACION DE GRAFICAS
- EJECUCION U OBTENCION DE GRAFICAS

Utilizando diagrama de sintaxis se muestran estos procesos:

OPCIONES DE GRAFOS



PROCESOS DEL GENERADOR DE GRAFICAS GRAFOS



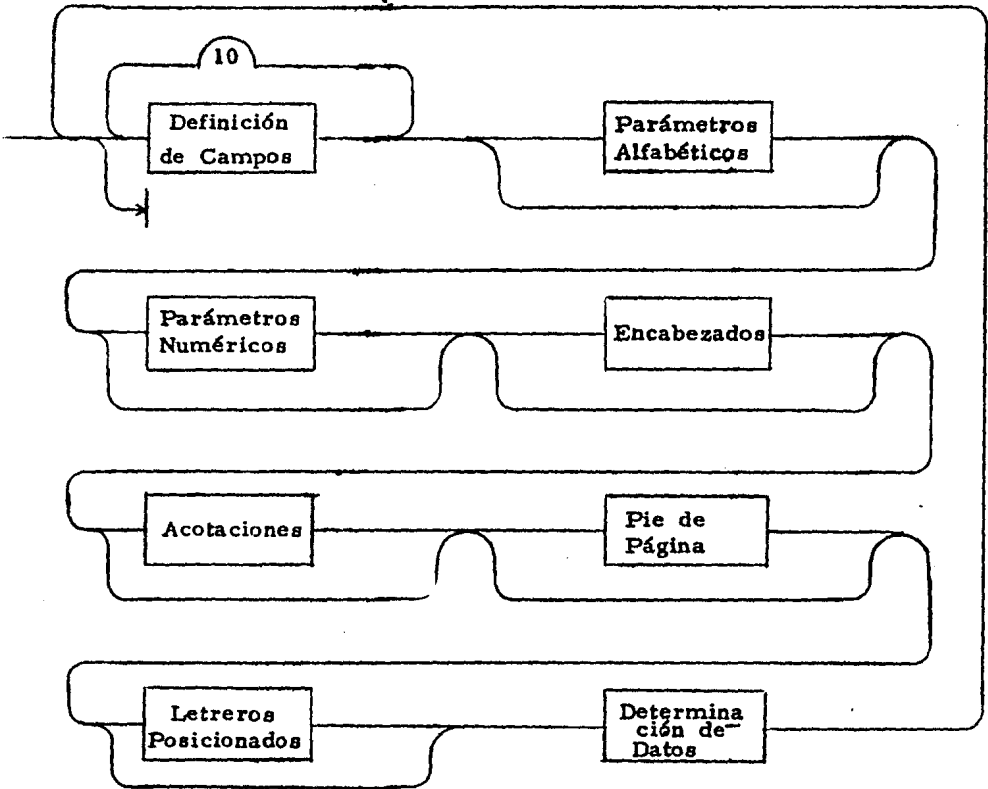
4.- ZONAS DE UNA GRAFICA

En general cada una de las áreas gráficas se ajustan a un patrón el cual puede ser cumplido total o parcialmente por el usuario.

La distribución de estas zonas por cada área gráfica se muestra a continuación:

ENCABEZADOS	
ACOTACIONES	GRAFICA
PIE DE PAGINA	

5. DEFINICION DE LAS ZONAS DEL AREA GRAFICA



La forma de definir el ó las áreas gráficas, así como sus zonas se muestra en el siguiente esquema.

5.1. COMANDOS DE DEFINICION

En este inciso se explican los comandos que se pueden utilizar en la definición de gráficas. Posteriormente se indicará en que zonas se pueden ocupar los diferentes comandos.

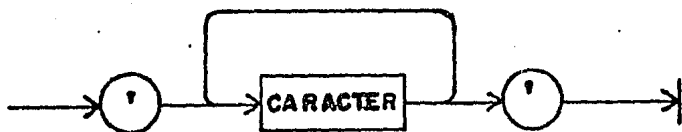
Los comandos están asociados a un nombre, el cual no es necesario escribirlo completo, ya que el sistema sólo reconoce el primer caracter.

Ejemplo: BLANCOS 3
 BLAN 3
 B 3

El efecto es el mismo.

A continuación se enuncian cada uno de los comandos

LETRERO



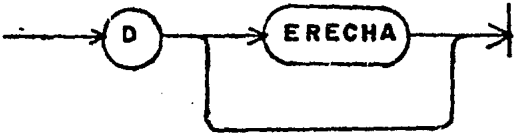
Conjunto de caracteres que pueden ser letras, dígitos, o cualquier símbolo de teclado a excepción del apostrofe, ya que éstos son los delimitadores del comando letrero.

BLANCO



Este comando tiene como función saltar tantas líneas - como lo indique el número entero (N) el no anotar ningún número - implica un salto de línea por default.

DÉRECHA



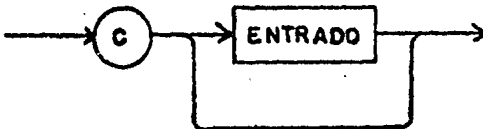
Este comando tiene efecto sobre los letreros que se coloquen enseguida de él, su función consiste en colocar al lado derecho de la hoja el letrero asociado.

IZQUIERDA



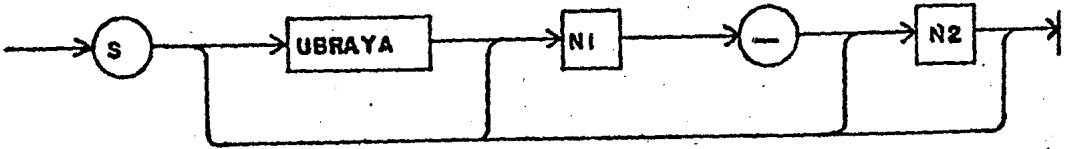
Este comando tiene efecto sobre los letreros que se coloquen enseguida de él, su función consiste en colocar al lado izquierdo de la hoja el letrero asociado.

CENTRADO



Este comando tiene efecto sobre letreros que se coloquen enseguida de él, su función es la de colocar en la parte central el letrero asociado.

SUBRAYADO

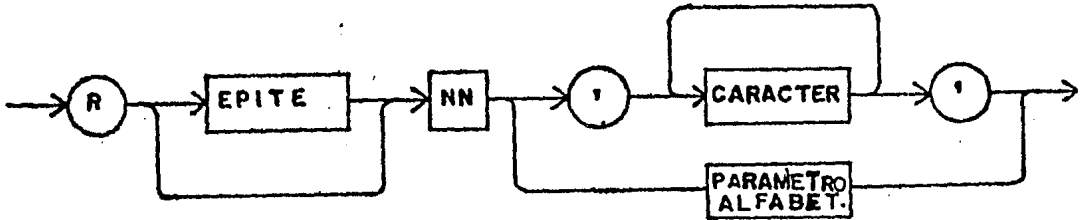


Este comando subraya desde la columna inicial (N1) hasta la columna final (N2) su función se aplica sobre un renglón, la omisión de los números implica el subrayado de toda la línea.



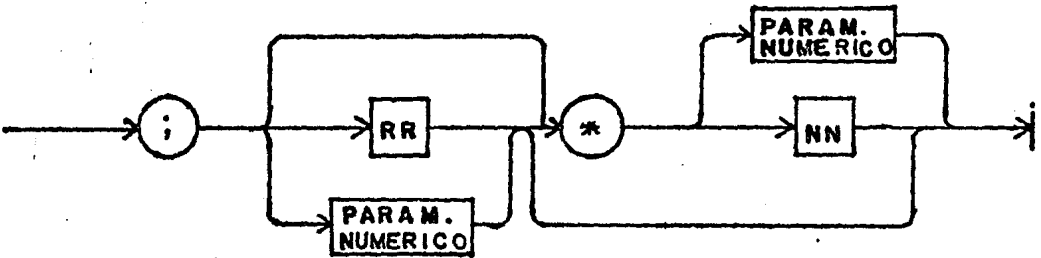
Este comando sólo se permite en la primer posición de la línea su función es indicar que la línea actual, es continuación de la anterior, éste comando es utilizado debido a las limitaciones que se tienen en el video.

REPITE



Este comando repite el letrero o parámetro alfabético tantas veces como lo indique el número entero (NN)

TAMAÑO

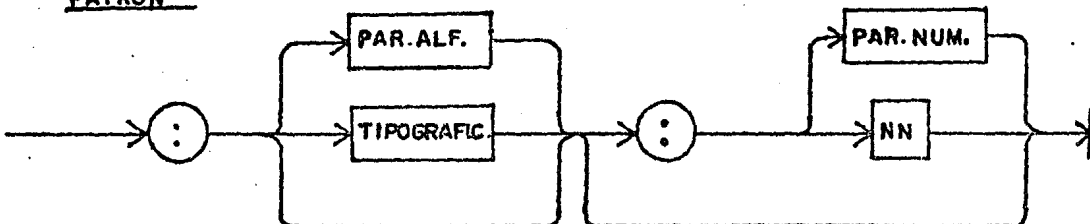


Este comando está compuesto de dos instrucciones - las cuales a su vez pueden ser parámetros numéricos:

RR: Número real que define el tamaño de un caracter

NN: Número entero que define el grueso de línea - para el caracter.

PATRON



Este Comando está compuesto por dos instrucciones, que

son:

TIPOGRAF: Caracter que indica el tipo de gráfica que se va emplear a saber

H - Histograma

P - Polígono

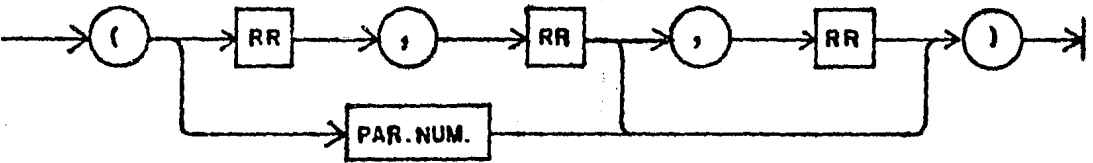
I - Impulso

V - Versatec

O bien un parámetro alfanumérico.

NN: Define el tipo de símbolo o ashurado según el tipo de gráfica que se trate vease tabla-

POSICION



Este Comando consta de 3 componentes que son:

RR1, RR2, 6 Parámetro numérico que definen las coor
de nadas para un letrero.

RR3 : Define el angulo de inclinación del letrero -
asociado, la ausencia de éste número implica
por default cero grados.

En este comando todos los números son de tipo real.

5.2. DEFINICION DE CAMPOS

La definición de campos se realiza como lo muestra la figura 5.2.1. donde:

X1, X2: Definen el eje X

Y1, Y2: Definen el eje Y

T: Tamaño del caracter asociado al Campo Gráfico

L: Líneas asociadas al campo gráfico

C: Caracteres que se desean por líneas.

N1,N2,N3: Números asociados a las opciones; tamaño, líneas o caracteres.

Nombre: Se define en caso de que en éste Campo se vaya a imprimir un cuadro o reporte, éste nombre será con el que se haya creado el reporte mediante MALER.

DEFINICION DE CAMPOS

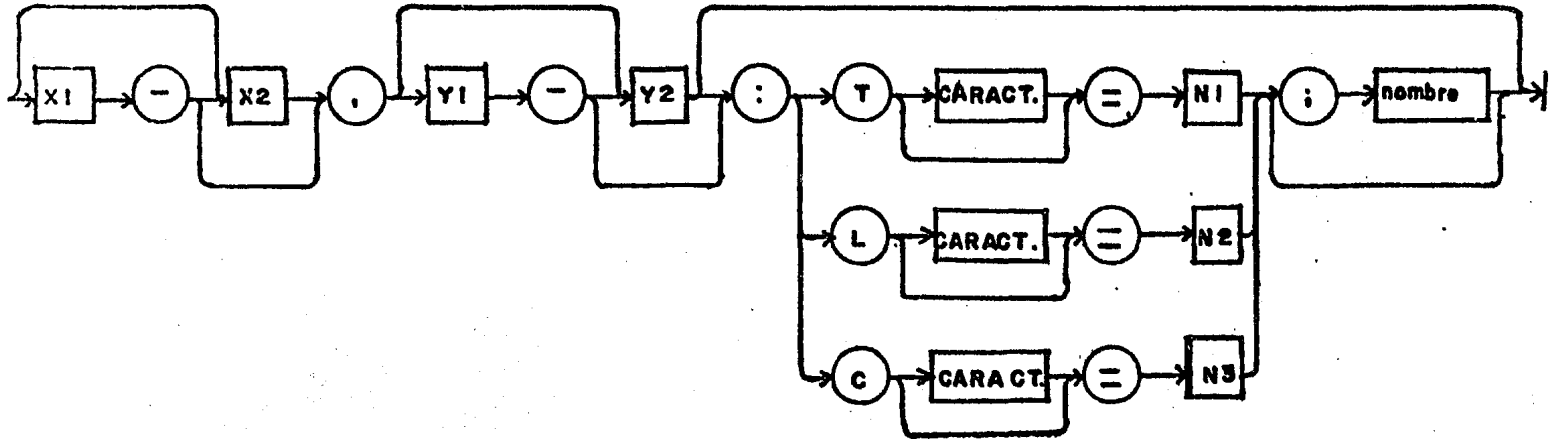
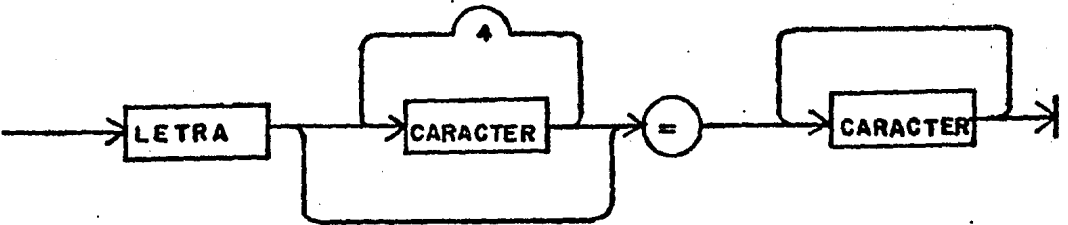


Figura 5.2.1.

5.3. PARÁMETROS ALFABETICOS

Los parámetros alfabéticos se definen como:

PARAMETRO ALFABETICO



Donde el parametro alfabético consta de 2 partes, un nombre y un letrero.

El nombre puede ser de 1 a 6 letras y/o dígitos donde el prime mero debe ser siempre una letra.

El letrero estará formado por una secuencia de caracteres cua lesquiera que éstos sean y su longitud dependerá del ancho de la pantalla.

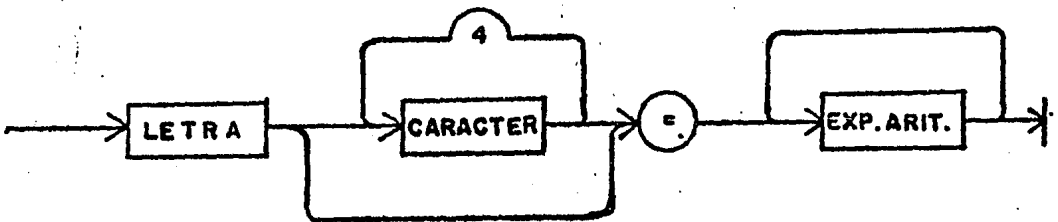
La función de los parámetros es la siguiente: Cuando aparezca el nombre del parámetro dentro de encabezados, acotaciones, letreros posicionados o pie de página se substituirá el nombre del parámetro por el letrero asociado.

NOTA: También es posible hacer la substitución en determinación de los datos en la parte que corresponde al nombre del banco.

5.4. PARAMETROS NUMERICOS

Los parámetros numéricos se definen como:

PARAMETRO NUMERICO



Donde el nombre sigue la misma estructura que la de parámetros alfabéticos.

El contenido consta de una expresión numérica la cual estará separada por:

- . punto
- , coma
- guión

Su función es substituir la expresión numérica en donde aparezca el nombre del parámetro numérico.

Estos parámetros son ocupados básicamente en la determinación de datos, en las partes en que existen expresiones numéricas.

5.5. ENCABEZADOS

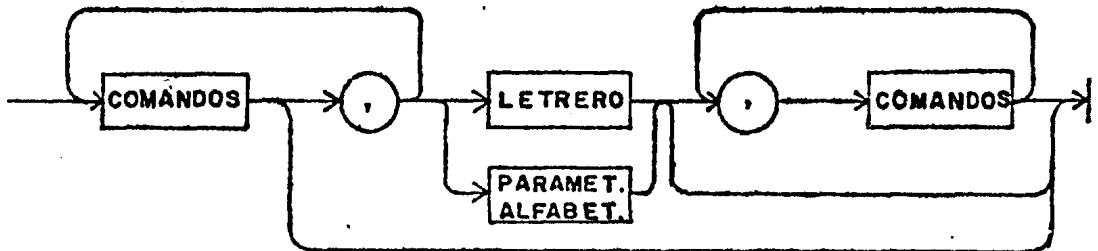
Los comandos validos para definir la zona de encabezados son:

- LETRERO
- TABULA
- BLANCO
- DERECHA
- IZQUIERDA
- CENTRADO
- SUBRAYADO
- REPITE
- PARAMETROS ALFABETICOS

Todos éstos se pueden combinar para definir el encabezado del área gráfica, las combinaciones deben ir separados por comas.

La estructura en diagrama de sintaxis es:

ENCABEZADOS



5.6. ACOTACIONES

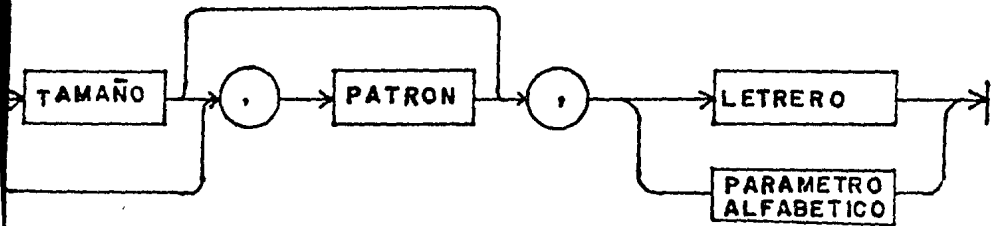
Los comandos validos para definir la zona de acotaciones son:

TAMAÑO
PATRON
LETRERO

Estos comandos deberán estar separados por comas.

La estructura en diagrama de sintaxis es:

ACOTACIONES



En esta zona se declaran los textos identificadores de las gráficas, así como un símbolo o ashurado asociado al texto.

5.7. PIE DE PAGINA

Los comandos validos para definir la zona de Pie de Página son exactamente los mismos que para Encabezados.

5.8 LETRERO POSICIONADO

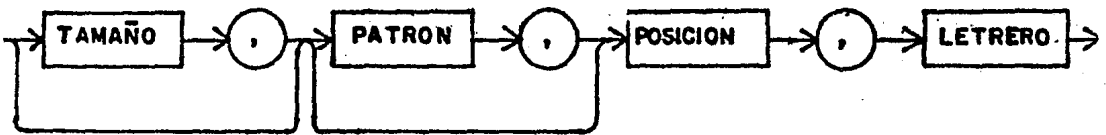
Los comandos validos para definir un letrero posicionado son:

TAMAÑO
PATRON
POSICION

Estos comandos deberán estar separados por comas.

La estructura en diagrama de sintaxis es:

LETRERO POSICIONADO



La función del letrero posicionado es el de colocar una o más líneas de texto, con algún símbolo o ashurado si lo desea, en cualquier parte del área definida para la gráfica, con el angulo de inclinación que desee, así como tamaño de caracter y grueso de éste.

5.9. DETERMINACION DE DATOS

La Determinación de datos es la parte del Sistema Graficador-
GRAFOS, en el cual se describen los datos a graficar y el tipo de gráfica
a emplear, así como el tipo de símbolo o ashurado que se ocupará.

El diagrama de sintaxis se muestra en la figura de la siguiente -
página, en donde:

BANCO.- Indica el nombre del banco que contiene los datos que serán -
graficados (dicho nombre puede ser un parámetro alfabético),
tomando en cuenta que el nombre constará de 5 caracteres al-
fanuméricos y el sexto será un número que identifique la orga-
nización como está almacenada la información.

El sexto caracter puede ser:

1. - Indica organización anual
2. - Indica organización trimestral
3. - Indica organización mensual

NSI.- Número de la serie inicial a partir de la cual se tomarán los
datos (por default se toma la primera serie que contenga el
banco)

NSF.- Número de la serie final hasta la cual se tomarán los datos
(por default se toma la última serie que contenga datos)

AI .- Año Inicial a partir del cual se tomarán los datos

PI.- Período inicial a partir del cual se tomarán los datos
(En estas dos instrucciones por default se toma el primer -
año y período que contenga la serie.

AF. - Año Final hasta el cual se tomarán los datos

AF. - Periodo Final hasta el cual se tomaran los datos
(En estas dos instrucciones por default se toma el último año
y perfodo que contenga la serie)

TIP.GRAF. - Se indica el tipo de gráfica que se empleará y que puede ser:

H - Histograma

P - Polígono

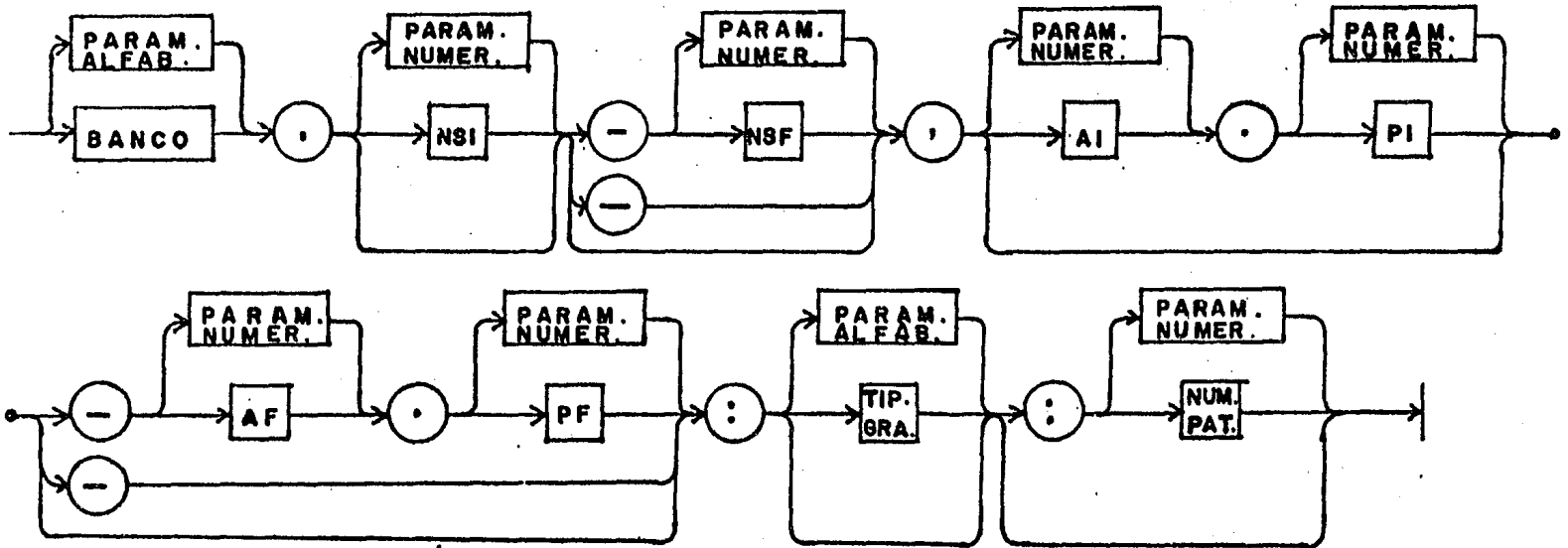
I - Impulso

V - Versatec.

por default se toma histograma









NUM.PAT. - Indica el tipo de símbolo o ashurado que se desea em
plear de acuerdo al tipo de gráfica de que se trate - -
véase tabla

DETERMINACION DE DATOS











CATALOGO DE PATRONES

Los patrones de sombreado se aplican tanto a HISTOGRAMAS como a GRAFICAS CIRCULARES y son los siguientes:

-  PATRON NO. 1
-  PATRON NO. 2
-  PATRON NO. 3
-  PATRON NO. 4
-  PATRON NO. 5
-  PATRON NO. 6
-  PATRON NO. 7
-  PATRON NO. 8

Los símbolos del catálogo se aplican tanto al polígono VERSATEC como al polígono estándar, y son los siguientes:

-  PATRON NO. 1
-  PATRON NO. 2
-  PATRON NO. 3
-  PATRON NO. 4
-  PATRON NO. 5
-  PATRON NO. 6
-  PATRON NO. 7
-  PATRON NO. 8

6. MODIFICACION

La etapa de Modificación del sistema GRAFOS, - luce como un editor de textos convencional. Esto significa que cada línea tecleada por el usuario al momento de la "Definición" de una gráfica podrá ser afectada por comandos de edición como por ejemplo: borrar líneas, cambiar cadenas de caracteres, o localizarlas, insertar nuevas líneas, etc.

En este caso particular se ha diseñado el programa de tal forma que se asemeje lo más posible al editor con que cuenta el computador PDP 11/40. Ya que de esta forma el usuario invertirá un mínimo esfuerzo para manejar ambos, conociendo uno de los dos.

Todos los comandos llevan un espacio en blanco para separar la letra o letras que lo identifican del resto del comando, y también deben contener al final de la línea

el caracter carrige Return o <CR> . Debido a que estas convenciones son generales, dichos caracteres no serán - indicados dentro de la sintáxis.

Por último, para indicar el inicio del archivo se usa la convención [* TOF *] (TOP OF FILE), y para - indicar el final se tiene: [* EOF *] (END OF FILE).

En seguida se muestra la sintáxis de todos los comandos y su significado.

6.1. Comandos de Modificación INSERTA



Este comando sirve para insertar nuevas líneas en el texto, a partir de la línea que aparece en la pantalla. Toda línea tecleada después del comando se tomará como un nuevo registro, hasta que el usuario proporcione una línea vacía.

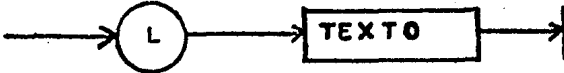
CAMBIA



Este comando cambia la cadena del texto 1 por la cadena - del texto 2, tantas veces como lo indique el número asociado al comando.

Por otro lado, al menos una de las cadenas debe ser "no vacía".

LOCALIZA



Este comando, localiza a través del archivo, el texto anotado a la derecha de éste.

La búsqueda se realiza hacia adelante y a partir de la línea que se observa en la pantalla en ese momento. Si la cadena no es localizada aparecerá la identificación de Final del Archivo.

PINTA



Este comando imprime en la pantalla, el número de líneas indicadas por el número asociado al comando, y contando hacia adelante. En caso de omitir el número sólo se imprimirá una línea.

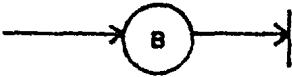
Si se tratara de rebasar el límite inferior del archivo - aparecerá la indicación correspondiente: * EOF * , y la impresión se suspenderá.

ATRAS



Este comando también imprime en la pantalla el número de líneas indicadas por el número asociado, sólo que el conteo es hacia atrás. En caso de tratar de rebasar el límite superior del archivo también se hará la indicación correspondiente: * TOF * .

BOTOM



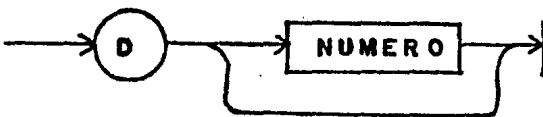
Al aplicarse este comando el usuario se posicionará en la última línea del archivo y la imprimirá en la pantalla.

TOP



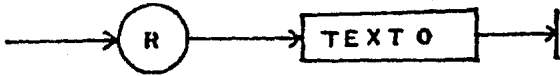
Este comando en cambio, coloca al usuario al inicio del área de trabajo, pero no aparecerá en la pantalla ninguna línea, ya que el inicio o "Cabeza de la Lista", siempre permanece vacía.

DELETE



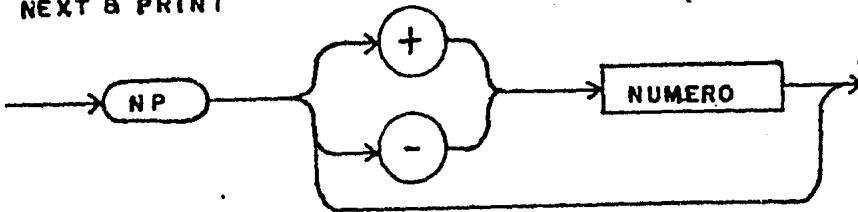
Al aplicar este comando serán eliminadas tantas líneas como lo indica el número asociado al comando. El cual puede omitirse y se tomará como uno.

REPLAZA



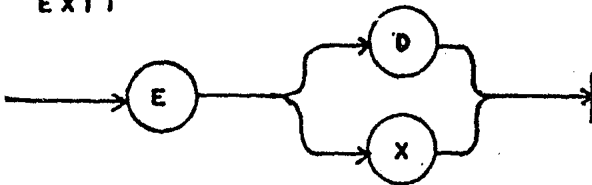
Este comando cambiará completamente el contenido de la línea actual por el texto asociado al comando.

NEXT & PRINT



La función de éste será imprimir la línea que se encuentra hacia adelante o hacia atrás, tantas posiciones como lo indica el número asociado.

EXIT



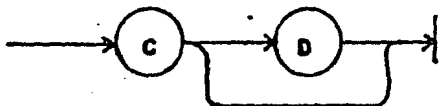
Al aplicarse este comando, el usuario finalizará la fase de modificación.

SPOOLER



Con este comando, también se finaliza la etapa de modificación.

CLOSE



Este último comando, además de finalizar la sesión de modificación, obtiene un listado del archivo modificado.

7. ACTUALIZACION DE PARAMETROS:

Recordando que pueden existir dos tipos de parámetros: los alfabéticos y los numéricos, los cuales están compuestos por el nombre y su contenido.

La actualización de parámetros permite la flexibilidad de cambiar algunos textos o datos de una gráfica en forma dinámica sin necesidad de ingresar a la etapa de modificación.

NOTA: En ésta etapa no es posible modificar el nombre de los parámetros sino solo su contenido, tampoco es posible borrar los parámetros definidos o aumentarlos, para los casos anteriormente mencionados se logra en la fase de modificación.

8. EJECUCION:

Para la ejecución de gráficas es necesario que se hayan definido, en cuyo caso existirá un archivo descriptor el cual contendrá todos los comandos válidos para la obtención de las gráficas.

Como al momento de la definición se verificó que los comandos fueran válidos y además en caso de que se hayan efectuado modificaciones también fueran verificados los comandos, la ejecución para la obtención de las gráficas será exitosa, ya que en este módulo se ejecutan uno a uno los comandos que conforman a las gráficas.

En este módulo existe además una opción en la cual el usuario indicará si las áreas que definió las quiere con un marco o no.

A P E N D I C E B

LISTADO DE MENSAJES DE ERRORES

1. DEFINIO COORDENADA (X) MINIMA O (Y) MINIMA FALTA UN GUIÓN.
2. DEFINIO COORDENADA (X) MAXIMA O (Y) MAXIMA FALTA DOS PUNTOS.
3. EL CAMPO DEFINIDO OCACIONA TRASPASE (POR OTROS).
4. A PARTIR DE ESTE LUGAR SE ACEPTA SOLO ALGUNA DE ESTAS LLAVES: TAMANO, LINEAS, CARACTERES.
5. DESPUES DE LA LLAVE DEBE SEGUIR UN '='.
6. EL NOMBRE DEFINIDO DEBE SER UNA CADENA ALFANUMERICA DE 6 CARACTERES MAXIMO EN EL RANGO A-Z Y 0-9.
7. CARACTER INVALIDO DEBE SER UN DIGITO ENTRE 0-9.
8. EL NUMERO MAXIMO PERMITIDO EN ESTA PARTE ES DE 999.9.
9. SI VA A DEFINIR: LINEAS, TAMANO O CARACTERES NO TIENE SENTIDO EL DATO CERO.
10. SOLO SE ACEPTA UN CAMPO DECIMAL.
11. GRAFICA PREVIAMENTE DEFINIDA, PUEDE MODIFICARLA O EJECUTARLA.
12. LOS PRIMEROS 5 PRIMEROS CARACTERES DEL NOMBRE DEL BANCO DE DATOS DEBEN SER ALFABETICOS EL SEXTO UN DIGITO.
13. EL SEXTO CARACTER DEL BANCO DEBE SER UN DIGITO (1- ANUAL, 2-TRIMESTRAL, 3-MENSUAL).
14. LA LINEA NO DEBE INICIAR CON DELIMITADOR COMA.
15. DEBE EXISTIR EL DELIMITADOR COMA EN ESTA POSICION.
16. SOLO SE ACEPTAN 4. DIGITOS COMO MAXIMO.
17. REVISE SU LINEA Y TECLEE NUEVAMENTE.
18. SERIE INICIAL O FINAL DEFINIDA NO ES PARAMETRO NUMERICO.
19. EN LA DEFINICION DE AÑO O PERIODO (INICIAL O FINAL) DEBE SEPARARSE CON COMA O GUIÓN.
20. EN ESTA POSICION DEBE EXISTIR DOS PUNTOS O BLANCO.
21. SOLO SE ACEPTAN DOS DIGITOS COMO MAXIMO.
22. DEBE SER DIGITO O IDENTIFICADOR (POR CATALOGO).
23. EL PRIMER CARACTER DEL PARAMETRO DEBE SER ALFABETICO.
24. NO SE ENCONTRA EL SIMBOLC IGUAL ENTRE EL NOMBRE DEL PARAMETRO Y LA CADENA.
25. NO ES PERMITIDO TENER UN PARAMETRO ASOCIADO A UNA CADENA VACIA.
26. CARACTER NO PERMITIDO EN LA CADENA PARA PARAMETROS NUMERICOS.
27. OPCION INVALIDA PARA ENCABEZADOS O PIE DE PAGINA.
28. OPCION INVALIDA EN ACOTACIONES.
29. NO CHECA EL CIERRE DE APOSTROFES PARA LETRERO.
30. OPCION INVALIDA.

30. EL COMANDO DEBE ESTAR ASOCIADO CON UN LETRERO O SU EQUIVALENTE (PARAMETRO ALFABETICO O REPITE).
31. LOS COMANDOS DEBEN IR SEPARADOS POR COMAS.
32. EN ESTA POSICION DEBE EXISTIR UN NUMERO COMPRENDIDO EN EL RANGO DE 1 A 132.
33. EL COMANDO REPITE DEBE ESTAR ASOCIADO CON UN LETRERO.
34. EL COMANDO LETRERO DEBE CONTENER UNA CADENA NO VACIA.
35. EL NUMERO ASOCIADO CON EL COMANDO BLANCO DEBE ESTAR EN UN RANGO DE 1 A 20.
36. EL COMANDO SUBRAYA DEBE ESTAR ASOCIADO CON UN GUION EN ESTA POSICION.
37. EN EL COMANDO SUBRAYA, N2 DEBE SER MAYOR QUE N1.
38. EN EL COMANDO SUBRAYA, N1 DEBE SER MAYOR QUE 1.
39. EN EL COMANDO SUBRAYA, N2 DEBE SER MENOR QUE 132
40. EL COMANDO "+" DEBE COLOCARSE EN LA PRIMERA POSICION DE LA LINEA.
41. EL GRUESO DEL CARACTER DEBE VARIAR ENTRE 1 Y 5 EN EL COMANDO TAMANO.
42. NOTACION NUMERICA INVALIDA.
43. TIPO DE GRAFICA NO CATALOGADA PARA EL COMANDO PATRON.
44. EL NUMERO DE PATRON VALIDO EN UN RANGO DE 1 A 10.
45. PARA DEFINIR NOMBRE DE UN REPORTE EN ESTA POSICION DEBE EXISTIR PUNTO Y COMA.
46. EN ESTA POSICION SOLO SE ACEPTA [:] PARA TAMANO, [:] PARA PATRON, [C] PARA POSICION.
47. EN ESTA POSICION SOLO SE ACEPTA [:] PARA DEFINIR PATRON [C] PARA DEFINIR POSICION.
48. EN ESTA POSICION ES OBLIGADO QUE DEFINA [C].
49. PARA LA DEFINICION DE UN LETRERO DEBE INICIAR CON [C] PARA EFECTUAR CAMBIO DE LINEA [C].
50. ESTA REVASANDO EL NUMERO MAXIMO PERMISIBLE 199.9 CMS.
51. EN LA DEFINICION DE ANGULO O COORDENADA: FALTA UNA COMA SI YA DEFINIO AMBOS FALTA [C].
52. PARA TERMINAR LA DEFINICION DE POSICION DEBE SER [C].
53. LA INICIACION DE POSICION DEBE SER CON [C].
54. CARACTER INVALIDO, SI LO QUE DESEA ES CONTINUAR EL LETRERO DEBE EXISTIR UNA COMA AMPERSAN [C].
55. EL NOMBRE DE LA GRAFICA TECLADA NO ESTA DEFINIDA.
56. ** COMANDO INVALIDO **
57. ** CADENA VACIA IMPOSIBLE LOCALIZARLA **
58. ** NUMERO INVALIDO **

59. ** SEGUNDO DELIMITADOR EN POSICION INCORRECTA **

60. ** NO SE LOCALIZO EL ULTIMO DELIMITADOR **

61. ** AL MENOS UNO DE LOS TEXTOS NO DEBE SER VACIO **

62. SE REBASAN LOS LIMITES DEL CAMPO AL SUBRAYAR.

63. LOS VALORES DEL EJE 'Y' REBASAN EL LIMITE PARA ACOTACION

64. EN EL TIPO DE GRAFICA ENVOLVENTE, SOLO SE ACEPTA
UN MAXIMO DE 58 OBSERVACIONES POR CADA SERIE.

A P E N D I C E C

LISTADO DE AYUDAS

••Modulo de control de procesos de graficacion, maneja las rutinas a que tienes acceso, tales como: DEFINICION, MODIFICACION, EJECUCION ACTUALIZACION DE PARAMETROS y CONECCION CON MALER.

• Modulo de DEFINICION, en este se proporcionan las instrucciones o Comandos necesarios para definir graficas y/o reportes.

• Modulo de EJECUCION, en este se controla la obtencion de graficas y/o reportes previamente definidas.

• Modulo de MODIFICACION, en este se permite cambiar las instrucciones o comandos que el usuario proporciona para definir graficas y/o reportes. Esto se realiza por medio de un editor de textos similar al del computador.

• Modulo de ACTUALIZACION DE PARAMETROS, este permite cambiar los parametros (partes variables dentro de graficas y/o reportes). Estos se pueden modificar cada vez que se ejecuta la grafica, pero solo es factible alterar su contenido, en caso de querer aumentar o disminuir el numero de parametros, o cambiar el nombre de estos debera hacerse con el modulo de MODIFICACION.

• Definicion de Campos: En esta parte se proporcionan las dimensiones y características físicas de los campos una grafica. En cada campo se puede asociar una grafica o un reporte (elaborado con MALER).

Cuando el programa despliega:

CAMPO NN=

El usuario debera usar la siguiente sintaxis:

XMIN - XMAX , YMIN - YMAX : llave=numero ; nombre

Donde:

llave: puede tener los valores: LINEAS, CARACTERES o TAMA&O.

numero: sera numero entero para LINEAS o CARACTERES, y numero real para TAMA&O.

nombre: es el nombre de un reporte generado con MALER.

• El modulo de PRUEBA DE CAMPOS permite obtener por graficadora electrostatica una muestra objetiva de la DEFINICION DE CAMPOS realizada por el usuario con anterioridad. En la muestra aparecen los limites de cada campo enmarcados con lineas y el numero de cada campo para poder identificarlos.

• La DEFINICION DE AREAS consiste en proporcionar los comandos o instrucciones adecuadas para definir una grafica. Cada campo definido se considera como una entidad y a cada uno se le asocian las siguientes areas:

ENCABEZADOS, ACOTACIONES, PIES DE PAGINA, LETREROS POSICIONADOS, DETERMINACION DE DATOS. Ademas cada uno puede tener PARAMETROS ALFABETICOS y PARAMETROS NUMERICOS.

Cada campo puede tener todas las areas mencionadas o solo algunas de ellas, segun convenga a las necesidades del usuario.

• 1er. vacio

• 2o. vacio

• 3er. vacio

• 4o. vacio

• 5o. vacio

• Modulo de MODIFICACION, en este se permite cambiar las instrucciones o comandos que el usuario proporciona para definir graficas y/o reportes. Esto se realiza por medio de un editor de textos similar al del computador.

••El comando LETRERO se define como una cadena de caracteres alfanumericos limitada por apostrofes al inicio y al final de esta.

Cuando un letreo aparece en ENCABEZADOS y PIES DE PAGINA, puede asociarse con los comandos: TAMA&O, IZQUIERDA, DERECHA, TABULA.

Cuando un letreo aparece en ACOTACIONES puede asociarse con los mismos comandos que en el caso anterior, y dos comandos mas que son: POSICION y PATRON.

- POSICION: requiere de tres variables separadas por comas, las dos primeras son las coordenadas que posicionan al letrero, la tercera es un angulo de inclinacion.

Las coordenadas son en centimetros y se toma como referencia el campo que se esta definiendo en ese momento.

La sintaxis es la siguiente: (posx,posy,ang)

Donde:

posx,posy y ang : deben ser numeros reales con punto decimal.

- El comando PATRON se aplica a un LETRERO o PARAMETRO ALFABETICO y su funcion consiste en asociar un patron grafico con el letrero o parametro en cuestion.

La sintaxis empleada es la siguiente.

:tipgraf:nl

Donde:

tipgraf: indica el tipo de grafica a emplear Poligono,Histogr. Impulsos,Versatec,Envolvente,Curva.Este valor puede darse con Par. Alf. o directamente con la primera letra del tipo de grafica.

nl: es un numero entero o un parametro numerico que indica el numero de patron a escoger dado por catalogo.

- El comando TAMANO se usa para modificar el tamaño del caracter para el campo que se esta definiendo en ese momento. Este tamaño se conserva hasta que aparezca otro comando TAMA&O o al cambiar de campo. La sintaxis usada es la siguiente:

:tamano*grueso

Donde:

tamano: es un numero real o un parametro que indica el tamaño de los caracteres en centimetros.

grueso: es un numero entero entre 1 y 5 que indica el grueso de los caracteres.

- El comando TABULA se debe asociar con un letrero o equivalente y su funcion consiste en iniciar la impresion de dicho letrero en la posicion indicada por el numero del comando. La sintaxis empleada es:

TABULAnl

Donde:

nl : es un numero entero mayor que 1 y menor que 132, el cual indica la posicion donde se inicia el letrero.(Nota: solo es necesaria la primera letra del comando)

- El comando DERECHA se usa para justificar a la derecha del campo en cuestion el letrero o equivalente que aparece en seguida de este. Los siguientes letreros tambien se justificaran a la derecha hasta que otro comando anule su efecto.

La sintaxis es la siguiente:

DERECHA

(Nota: solo es necesaria la primera letra del comando).

- El comando IZQUIERDA se usa para justificar a la izquierda del campo el letrero o equivalente que aparece en seguida. Los siguientes letreros tambien se justificaran a la izquierda hasta que otro comando anule su efecto.

La sintaxis es la siguiente:

IZQUIERDA

(Nota: solo es necesaria la primera letra del comando).

- El comando CENTRADO se usa para centrar un letrero o equivalente en la linea del campo en cuestion.Su efecto permanece hasta que otro comando lo anule.

La sintaxis es la siguiente:

CENTRADO

(Nota: solo es necesaria la primera letra del comando)

A P E N D I C E D

SUBROUTINAS DE VERSAPLOT

SUBROUTINA PLOTS.FTN

PLOTS (0,0,0)

FUNCION:

Es usada para inicializar en modo graficación al sistema VERSAPLOTH-07.

PARAMETROS:

Cuenta con tres variables del tipo entero las cuales deben asignarse con valor cero.

SUBROUTINA: PLOT,FTN

FUNCION:

Control de la pluma, pintar una línea, pintar una línea y hacer re-origen, hacer re-origen, posicionar la pluma en un punto y terminar el modo graficación.

PLOT (X,Y, \pm IPEN)

PARAMETROS:

X,Y (reales) Par de ejes coordenados que definen la posición de movimientos para la Pluma (E).

\pm IPEN (ENTERA)

IPEN = + 999 Fin del modo graficación (E).
IPEN = + 3 Mueve la pluma al punto (X,Y) (E).
IPEN = + 2 Pinta hasta el punto (X,Y) (E).
IPEN = - 2 Pinta hasta el punto y efectúa re-origen en (X,Y) (E).
IPEN = - 3 Efectúa re-origen en el punto (X,Y) (E)

SUBROUTINA: FACTOR

FUNCION:

Permite la ampliación o reducción de tamaño de lo definido en modo graficación.

FACTOR (FACT)

PARAMETROS:

FACT (Real) Variable que contiene el valor al que se va a reducir o ampliar.

SUBROUTINA: NEWPEN,FTN

FUNCION:

Es usada para seleccionar el ancho de línea -
que se desee en número de NIBS.

NEWPEN (INP)

PARAMETROS:

INP (Entera) Variable que contiene el ancho de lí-
nea en número de Nibs de 1 a 5, si --
INP > 5 se asume 5, si INP < 1 se asu-
me 1.

SUBROUTINA: SYMBOL,FTN

FUNCION:

Es usada para imprimir caracteres alfanuméricos o una cadena de estos en varios ángulos y tamaños.

SYMBOL (X,Y, HEIGHT, ITEXT, ANGLE, NC)

PARAMETROS:

X,Y (REALES) Coordenadas del punto a partir del --
cual se va a imprimir los alfanuméricos. (E).

HEIGHT (REAL) Variable con el valor del tamaño de --
los caracteres alfanuméricos a imprimir (E).

ITEXT (ALFANU-
MERICA)

Texto Alfanumérico a imprimir.

ANGLE (REAL)

Angulo en grados, con el cual será im-
preso el texto.

NC (ENTERA)

Indica el número de caracteres defini-
dos en ITEXT para ser impresos.

SUBROUTINA: SCALE

FUNCION:

Escalar un arreglo de datos entre un rango -
máximo y mínimo encontrado por esta misma función.

SCALE (ARRAY, AXLEN, NPTS, INC)

PARAMETROS:

ARRAY (REAL): Arreglo de datos para ser escalado (E)

AXLEN (REAL): Longitud del eje sobre el que se va a
escalar (E).

NPTS (ENTERA); Define el número de datos para escalar
definidos en ARRAY (E).

INC (ENTERA): Indica en que orden se tomarán los da-
tos del arreglo (E).

INC = 1 Se tomarán de 1 en 1.

INC = 2 Se tomarán de 2 en 2 etc.

OBSERVACIONES:

Esta rutina retrona en los elementos del arreglo ARRAY ubicados en NPTS + 1 y NPTS + 2, el mínimo valor escalado ---- (FVAL) y el incremento de escalamiento (DVAL).

- NPTS (ENTERA)** Contiene el número de valores de cada arreglo X.Y.
- INC (ENTERA)** Contiene el incremento con que se tomarán los valores de los arreglos X, Y - (de uno en uno, de dos en dos, etc.).
- LINTYP (ENTERA)** Define el tipo de gráfica a elaborar y la frecuencia con que se toman los valores a graficar. Por ejemplo, si **LINTYP** \pm 4 indica que cada cuatro valores se graficará un símbolo.
- LINTYP = 0** Indica que los puntos sólo serán unidos por líneas rectas y no se usarán símbolos.
- +LINTYP** Indica que se usarán líneas y símbolos en la gráfica.
- _LINTYP** Indica que sólo se usarán símbolos en la gráfica.
- INTEQ (ENTERA)** Define el símbolo centrado que se ocupará.

SUBROUTINA: AXIS.FTN

FUNCION:

Es usada para generar un eje con un ancho de línea que se defina, un texto alfanumérico centrado, y - marcas sobre el eje.

AXIS (X,Y, LABEL, +NCHAR, AXLEN, ANGLE, FVAL, DVAL)

PARAMETROS:

X,Y (REAL)	Define el punto donde se imprimirá el eje (E).
LABEL	Cadena de caracteres que será usada como texto asociado al eje.
+NCHAR (ALFANUMERICO)	Define el número de caracteres que tiene el texto (LABEL) si: (E)
+NCHAR	Indica que el texto y las marcas se imprimirán sobre el eje.
-NCHAR	Indica que el texto y las marcas se imprimirán abajo del eje.

AXLEN (REAL) Longitud que se destine para el eje (E)

ANGLE (REAL) Angulo en grados, con que se imprimirá el eje (E).

FVAL (REAL) Primer valor (mínimo) con el cual se - iniciará la impresión en las marcas (E).

DVAL (REAL) Incremento con que se varía el escala-- miento a partir de FVAL.

SUBROUTINA: TONE,FTN

FUNCION:

Se encarga de sombrear una área determinada de acuerdo con un patrón específico que el usuario especifica. Esta rutina tiene dos formas de invocarse:

FORMA 1 Establece el patrón de sombreado

CALL TONE (0.0, 0.0, IPAT, -NTP)

DONDE: IPAT es el arreglo que contiene el patrón de sombreado.

NTP es el número en valor absoluto de elementos válidos del arreglo IPAT (máximo 16).

FORMA 2

Realiza el sombreado de la superficie correspondiente.

CALL TONE (X, Y, NE, + NA)

X,Y Arreglos de números reales que contienen las coordenadas del área a sombrear.

NE Número de elementos de dichos arreglos.

NA Número de áreas a sombrear.