



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

Facultad de Ingeniería

**Desarrollo e Implementación de un Sistema de Diseño
Asistido por Computadora (CAD) para Graficación
de Perspectivas con el Método Red Modular**

T E S I S

Que para obtener el título de

INGENIERO EN COMPUTACION

P r e s e n t a :

Ricardo Javier Juárez del Moral

Director de Tesis: M. en I. Rafael B. Carmona Paredes

México, D. F.

1984



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 esta 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.

Indice

Sinopsis	1
Introducción	2
I. LA PERSPECTIVA Y RED MODULAR	7
I.1 Métodos tradicionales para el trazo de perspectivas.	8
I.2 Red Modular: Un método sistemático para trazo de perspectivas.	13
II. TRANSFORMACION RED MODULAR	
II.1 Obtención de la transformación.	21
II.2 Perspectivas de Cuerpos Geométricos.	24
III. CALCULO DE GIROS Y DESPLAZAMIENTOS EN RED MODULAR	
Introducción	29
III.1 Movimientos del Cuerpo Geométrico en Red Modular	30
III.1.1 Traslación.	31
III.1.2 Giratorio	36
III.2 Coordenadas Límite.	42

IV. DISEÑO DE UN SISTEMA CAD UTILIZANDO EL METODO RED MODULAR	49
Introducción	
IV.1 Organización del Sistema Red Modular.	52
IV.2 Descripción de Módulos:	
. Módulo maestro o de control.	60
. Módulo de entrada.	64
. Módulo de información y utilerías.	79
. Módulo de graficación:	
- Elementos de gráficas por computadora.	84
- Operación del módulo de graficación.	92
. Módulo de movimientos o ejecución	107
. Módulo de salida.	128
IV.3 Procedimiento en la elaboración del sistema	132
V. CONCLUSIONES.	136
VI. BIBLIOGRAFIA.	140

SINOPSIS

El presente trabajo trata de la manera en como se diseño un sistema de "diseño asistido por computadora (CAD)", basado en el método Red Modular que se desarrollo en el Centro de Investigaciones Arquitectónicas de la UNAM.

Por un lado se muestra a Red Modular como un método sistemático para la construcción de perspectivas y se compara con otros métodos de perspectiva como la cónica, caballera, etc. a fin de señalar ventajas y desventajas. Por otro lado se expone la metodología que se empleo en la elaboración de un sistema CAD enfocado al diseño arquitectónico.

El trabajo incluye 55 gráficas de las cuales 17 fueron hechas por la computadora.

El manual de operación y el listado del sistema no se incluyen debido a que no están en los objetivos de esta tesis.

INTRODUCCION

El uso de la computadora en las distintas actividades del hombre es un proceso irreversible e imposible de detener. La creciente demanda de agilizar el aparato productivo hace imperativa la necesidad de utilizar computadoras en todas las áreas del conocimiento humano, ya que se han convertido en poderosos, veloces y confiables instrumentos de trabajo que no solo procesan datos y ejecutan sofisticadas operaciones numéricas a gran velocidad, sino también, se usan como herramientas auxiliares en el diseño de máquinas-herramientas, en la construcción de carreteras, edificios, automóviles, casas habitación, circuitos electrónicos, etcétera, mediante el uso de gráficas planas o tridimensionales según las necesidades del diseño.

Retomando la frase del artista Leonardo Da Vinci "Una imagen dice más que mil palabras" de igual forma las gráficas computarizadas* representan por medio de imágenes, una gran cantidad de datos numéricos, que de presentarse en listados serían imposibles de analizar, por lo que la aceptación de la graficación por computadora se ha incrementado hasta ser la base de los sistemas de diseño asistido por computadora**.

El desarrollo de métodos que faciliten la realización de ideas mediante el diseño asistido por computadora es un proceso que no se detiene y deja sentir su penetración en el quehacer cotidiano del hombre. Como un ejemplo de este testimonio este trabajo de tesis es el desarrollo de un sistema de diseño asistido por computadora aplicado a la arquitectura basado en el método Red Modular.

El objetivo de la tesis es mostrar la metodología empleada en el desarrollo e implementación de un sistema computarizado, para asistir en el diseño arquitectónico, como referencia en el desarrollo de cualquier sistema de diseño asistido por computadora, ya que en la literatura consultada sobre graficación por computadora no existe estandarización ni metodología alguna en el diseño de este tipo de sistemas.

El sistema que presentamos surgió del proyecto de investigación Red Modular el cual es un método sistemático para la construcción de perspectivas de Cuerpos Geométricos. Fué desarrollado en el Centro de Investigaciones Arquitectónicas de la UNAM

* Tomado del inglés Computer Graphics y traducido como gráficas por computadora o gráficas computarizadas.

** Aceptación en español de Coputer Aided Design (CAD).

por el M. en A. Tomás García Salgado quien mediante el análisis de los diferentes tipos de perspectivas y la conceptualización geométrica del espacio, estableció una serie de postulados geométricos que le permitieron la construcción del modelo matemático 'Red Modular' en forma de una transformación $T: R^3 \rightarrow R^2$.

Debido a las características de Red Modular se plantearon los siguientes objetivos:

- . Demostrar que Red Modular es un método sistemático para trazar perspectivas tanto manualmente como en forma computarizada en un graficador.
- . Demostrar con ayuda de la computadora, que Red Modular es un método capaz de resolver cualquier caso de perspectiva, es decir, es de carácter universal.
- . Demostrar que Red Modular es un método que tiene aplicaciones en la investigación, en la docencia y en la práctica profesional del diseño arquitectónico.

Objetivos que se convirtieron en un proyecto de investigación que requirió de la interdisciplina Arquitectura-Ciencias de la Computación y de la creatividad e imaginación del computólogo para la realización del mismo.

El objetivo principal del proyecto consistió no solo en el desarrollo del software para realizar gráficas computarizadas de perspectivas arquitectónicas, sino también, en la elaboración de un programa suficientemente inteligente que controlara la ejecución de éstas siguiendo el método Red Modular para la construcción de perspectivas. El resultado es un sistema que opera a base de comandos para que el diseñador pueda, de manera sencilla a través del diálogo hombre-máquina, obtener un conjunto de distintas perspectivas de un mismo objeto proporcionándole así una herramienta de trabajo. Además con el desarrollo del sistema se al

canzaron los objetivos planteados al inicio de la investigación.

La computarización de Red Modular se realizó en distintas etapas con objetivos específicos e interrelacionados que permitieron un avance progresivo y retroalimentado del proyecto hasta su fase final. En la tesis estas etapas se presentan de la siguiente manera:

Se inicia con la descripción del método tradicional para la construcción de perspectivas en especial la cónica, y más adelante el de Red Modular con objeto de introducir los conceptos de Campo Visual, Cuerpo Geométrico, etc., mostrar la manera sistemática para construir perspectivas y las ventajas de Red Modular con respecto a otros métodos. Más adelante se explica la concepción del Espacio Modular (EM2) y se obtienen las ecuaciones de la Transformación Red Modular, movimientos de giro y traslación y coordenadas límite, y mediante ejemplos se muestran los algoritmos utilizados en la aplicación de estas ecuaciones en los capítulos I, II y III. Estos algoritmos se sumaron a los de graficación para llegar a un 'modelo de graficación' en el que se definió la estructura básica del sistema automatizado Red Modular y con él se dió inicio a la computarización del mismo.

En el capítulo IV se describe la manera en que se desarrolló el sistema mediante el uso de un metalenguaje (notación Backus Naur) y algoritmos escritos en pseudolenguaje para facilitar su comprensión.

El listado del sistema y el manual del usuario no se incluyen debido a que no entran en los objetivos de la tesis. En cambio se presentan los resultados del sistema como son los listados de datos y las gráficas que se muestran en este capítulo.

Los tiempos requeridos en la graficación y en la introducción de datos, y la manera de operar el sistema fueron parámetros que se compararon con los de otros paquetes similares actualmente en el mercado como son los de la compañía Hewlett Packard (HP), el

BDS (Building Design System) de Edimburgo Inglaterra, el Graphak II (Perspective Projection and Plotting Program) de la Universidad de Nueva York; con objeto de evaluar y medir la eficiencia y capacidad del sistema y para estimar su costo. Los resultados de esta comparación fueron alagadores ya que los rangos resultaron menores de un 25% a un 35% con respecto a los otros sistemas. Sin embargo no hay que perder de vista que estos sistemas trabajan en otras máquinas como son la HP, PDP11/45 y UNIVAC respectivamente.

Con este proyecto iniciado en abril de 1980 y concluido en febrero de 1982 queda plasmado el resultado de casi dos años de investigación en el desarrollo de sistemas de diseño asistido por computadora como muestra de creación tecnológica en Ciencias de la Computación y deja la puerta abierta a la creatividad en este tipo de sistemas.

I. 1 Métodos tradicionales para trazo de perspectivas.

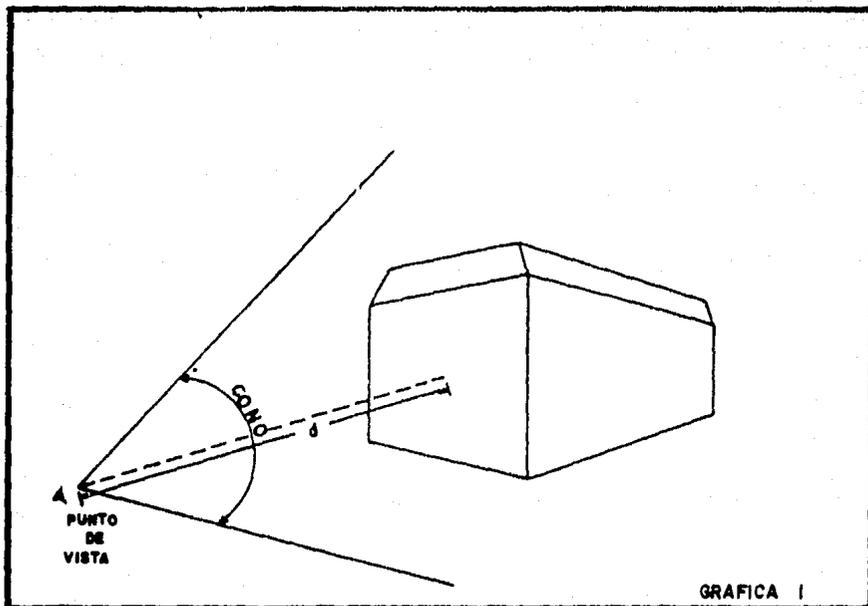
La Perspectiva

La perspectiva es la representación de los objetos en una superficie o plano (llamado cuadro) en la forma y disposición con que aparecen a la vista del observador desde un lugar determinado llamado punto de vista.

La técnica empleada para obtener cualquier tipo de perspectiva es la siguiente:

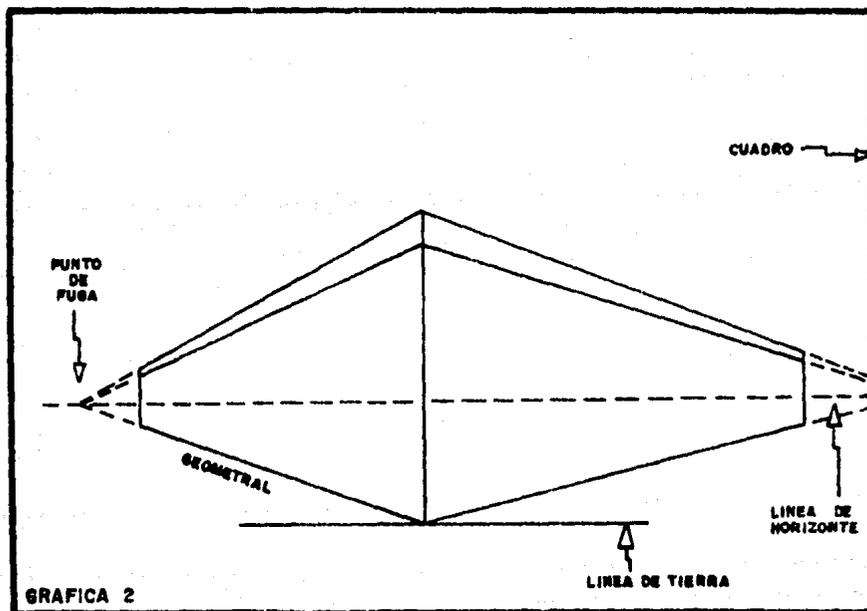
- . el objeto a representar se supone colocado sobre un plano horizontal llamado geometral.
- . el cuadro se supone vertical al geometral.
- . la intersección del cuadro con el plano horizontal que pasa a la altura de la vista del observador se llama línea de horizonte o de ojos. Esta línea es paralela al geometral.
- . el lugar de observación o punto de vista seleccionado da por resultado una perspectiva, de tal manera que al variar éste se tendrá una nueva perspectiva, por esta razón se dice que el punto de vista en una perspectiva es único.
- . el punto de fuga o elemento límite se localiza en la línea de horizonte, y es el punto hacia el cual se proyectan todas las rectas del objeto.
- . en una perspectiva se pueden tener hasta tres puntos de fuga, los cuales pueden hallarse fuera o dentro del cuadro dependiendo de la magnitud de éste así como de la posición del observador.

La ubicación del observador (punto de vista) permite clasificar las perspectivas: si el observador está localizado en el infinito se tendrá la perspectiva paralela (también llamada pro-



GRAFICA 1

PERSPECTIVA CONICA



GRAFICA 2

PERSPECTIVA CENTRAL A DOS PUNTOS DE FUGA

yección ortogonal por ser plana y carecer de volumen); si su ubicación es finita, es decir, el observador se aproxima más al objeto se tendrá la perspectiva cónica; si el punto de vista se halla en lo alto, esto es, el objeto se observa desde arriba, se tendrá la perspectiva aérea o a ojo de pájaro. Estas son algunas de las perspectivas más usadas.

En la perspectiva cónica a dos puntos de fuga del edificio mostrado en la gráfica 2 se tienen los siguientes elementos:

- . El geometral es el plano horizontal sobre el cual se halla el objeto (edificio) en este caso es el piso.
- . El cuadro es el plano perpendicular al geometral y está frente al observador. Limita el campo visual* del objeto, es decir, es la superficie donde el observador dibuja el objeto.
- . La línea de tierra está sobre el geometral y se localiza en la intersección del objeto con el geometral, en este caso coincide con la esquina inferior izquierda del edificio más próxima al observador, es decir, la arista mas larga por estar frente al observador.
- . Las aristas del objeto (edificio) en esta perspectiva quedan sobre las líneas que unen los vertices del objeto con los puntos de fuga. En este caso uno se halla dentro del cuadro y el otro fuera.

La perspectiva más usual es la cónica. Este tipo de perspectiva se caracteriza por dirigir las líneas visuales del objeto hacia el punto de vista como se muestra en la gráfica 1, por lo que algunos autores la definen como proyección central. Sus elementos principales son el cuadro y el punto de vista; la magnitud y la posición de estos elementos determinan si el o los puntos de fuga están dentro o fuera del cuadro como en el caso del edificio de la gráfica 2.

* El concepto de Campo Visual en materia de perspectiva es introducido por el método de Red Modular.

En la proyección central los elementos de referencia son un plano en lugar del cuadro y el punto de vista, la proyección de este último en el plano se llama punto principal, y la distancia entre ambos puntos se llama distancia principal y se indica con D.

Sobre el geometral se puede construir un sistema de ejes coordenados espaciales que permiten proyectar (sobre el plano), todo cuerpo localizado en él, entendiéndose por proyección la operación de construir todas las rectas que van desde el punto de vista a los puntos del objeto (nodos^{*}), los cuales se cortan con el plano dando por resultado la proyección deseada que es muy similar a la perspectiva cónica.

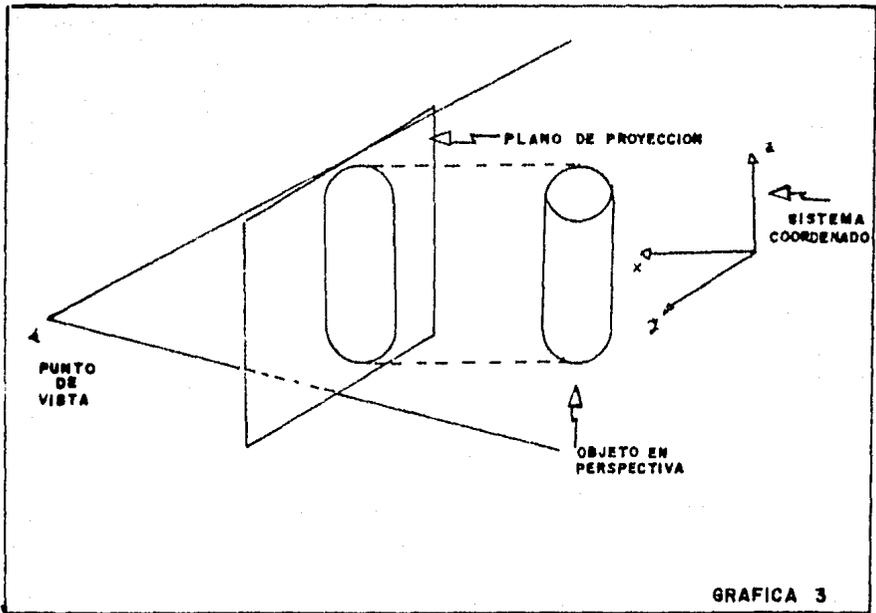
A continuación describimos el método para trazar perspectivas cónicas:

- . determinar el punto de vista deseado manteniéndolo fijo.
- . trazar el cuadro que servirá de marco de referencia sobre el cual se va a representar el objeto deseado.
- . trazar las líneas de tierra y horizonte del objeto con respecto al punto de vista.
- . fugar (proyectar) las rectas del objeto a su lugar correspondiente.

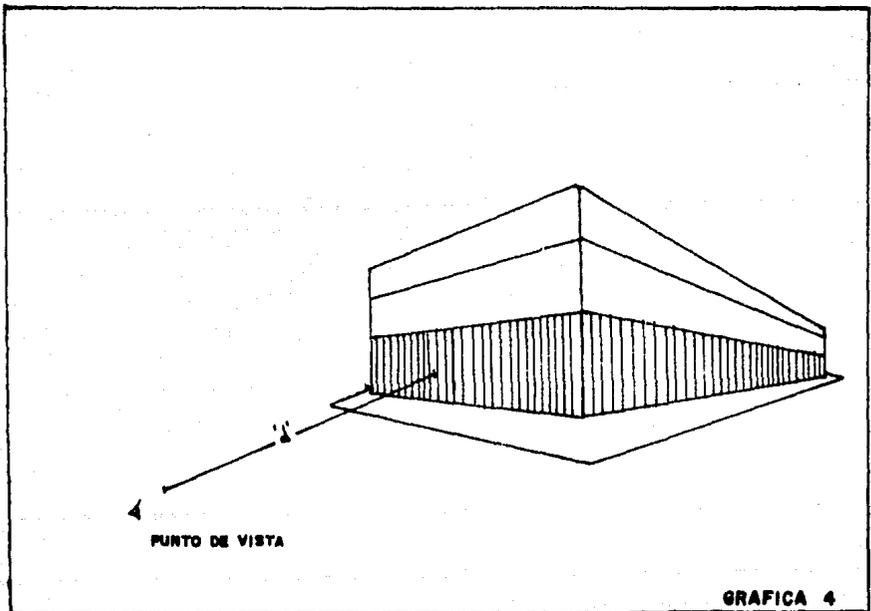
Aunque esta perspectiva es la más utilizada tiene las desventajas que se listan a continuación:

- . Para localizar los puntos de fuga se usan artificios geométricos complejos y hay errores en su localización cuando están bastante lejos del cuadro.

* Al tenerse un sistema espacial un punto está representado y por ende queda localizado, por su terna de coordenadas (X,Y,Z), que al unirse con todos los demás puntos del objeto se tiene una gráfica la cual está formada por nodos, por tanto un punto del objeto es un nodo.



PROYECCION CENTRAL



PERSPECTIVA CONICA

- . Se desconocen las dimensiones reales del objeto.
- . Por cada perspectiva distinta del mismo objeto se tiene que repetir el proces .
- . La posición real del punto de vista es desconocida.

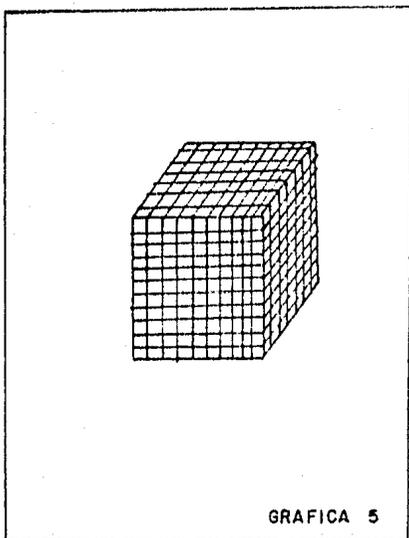
Estas desventajas no se presentan en el método Red Modular.

Comparando la perspectiva cónica de la gráfica 1 con la proyección central mostrada en la gráfica 3 observamos la similitud que existe entre ambas, sin embargo, utilizaremos la proyección central en lo sucesivo para referirnos a perspectivas, debido a que ésta permite usar ejes coordenados que serán de gran utilidad en la deducción del método Red Modular.

I.2 Red Modular; un método sistemático para trazo de perspectivas.

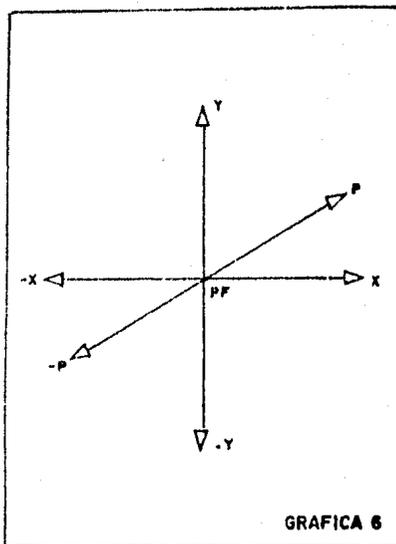
Del estudio de los distintos tipos de perspectiva (cónica, caballera, aérea, etc.) y la conceptualización de los elementos comunes en la construcción de éstas (punto de fuga, línea de tierra, línea de horizonte, etc.) se estableció un conjunto de postulados y definiciones que conforman la columna vertebral de Red Modular, es decir, Red Modular tiene una conceptualización del espacio y define los elementos de una manera formal facilitando la construcción de perspectivas. Los conceptos de Campo Visual, Estación de Observación y planos de simetría y proyectivos son introducidos por Red Modular.

Partiendo de la perspectiva cónica de la gráfica 4 en la que el observador se halla a una distancia 'd' finita del objeto desde la cual percibe solamente una parte de éste y de su medio ambiente, a lo que llamamos Campo Visual, y considerando que el mismo fenómeno se tiene para cualquier otro caso de perspectiva, Red Modular (RM) establece el siguiente postulado:



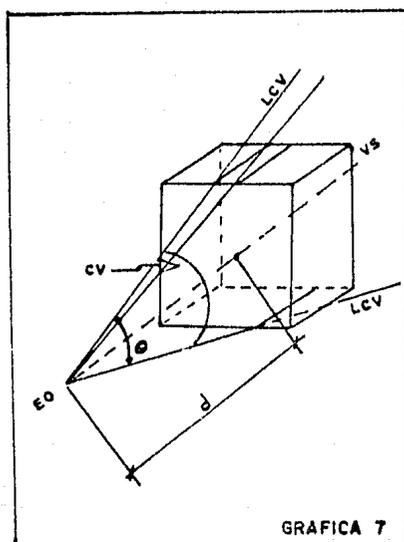
GRAFICA 5

CUBO DE 10 MODULOS POR LADO



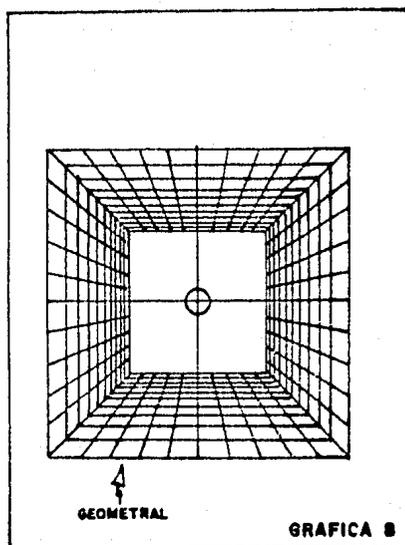
GRAFICA 6

SISTEMA COORDENADO (X / Y / P)



GRAFICA 7

CONSTRUCCION DE LA RED



GRAFICA 8

RED MODULAR

"todo fenómeno de perspectiva captado por el ojo humano ocurre en una porción limitada del espacio, que es el Campo Visual".

Para seguir una secuencia lógica se introduce aquí la definición de Cuerpo Geométrico y Estación de Observación:

- . Al objeto en observación se le llama Cuerpo Geométrico (CG). Este puede ser cualquier sólido limitado por su superficies planas, curvas o irregulares.
- . Al punto de vista se le llama Estación de Observación (EO) y se define como el lugar desde el cual se observa(n) el(los) cuerpo(s) geométrico(s).

Utilizando estos dos conceptos se define al Campo Visual (CV) como la capacidad que tiene el observador de percibir un Cuerpo Geométrico (CG) y su medio ambiente desde una Estación de Observación (EO) determinada, dicha capacidad o poder de resolución está en función del ángulo visual o Amplitud de Campo Visual (ACV) del observador.

Por el postulado anterior puede idealizarse al espacio como un espacio geométrico finito constituido por un cubo dividido en diez unidades de igual magnitud por lado llamadas Módulos, como el mostrado en la gráfica 5, en la que la base del cubo se congtruye sobre el geometral.

La construcción de este cubo permite utilizar un sistema coordenado tridimensional (X/Y/P), como el mostrado en la gráfica 6, el cual analizaremos con más detalle después de construir la Red.

CONSTRUCCION DE LA RED.

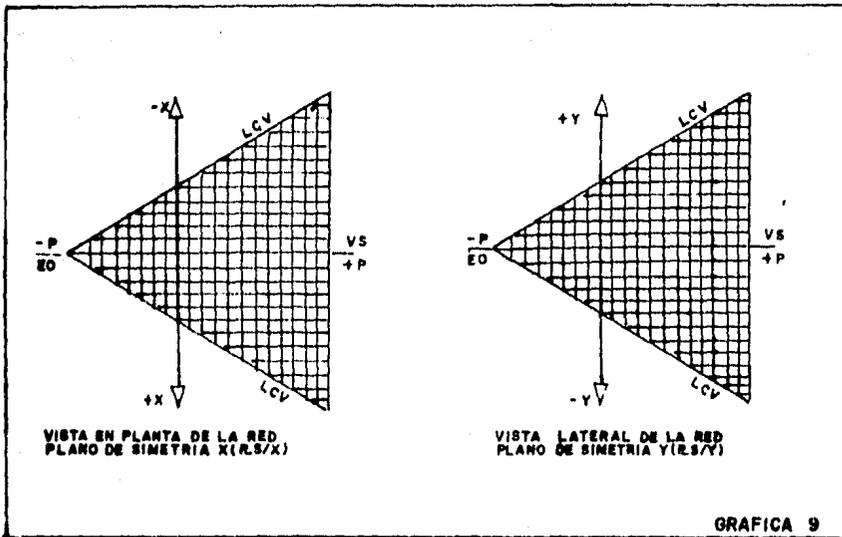
Situando la Estación de Observación (EO) a una distancia de 'd' módulos frente a alguna de las caras del cubo para que el observador dirija su vista al centro de dicha cara y considerando

una Amplitud de Campo Visual (ACV) de θ grados para que sus límites visuales incidan en los bordes del cubo, como se muestra en la gráfica 7, desarrollamos la Red en base a las siguientes definiciones:

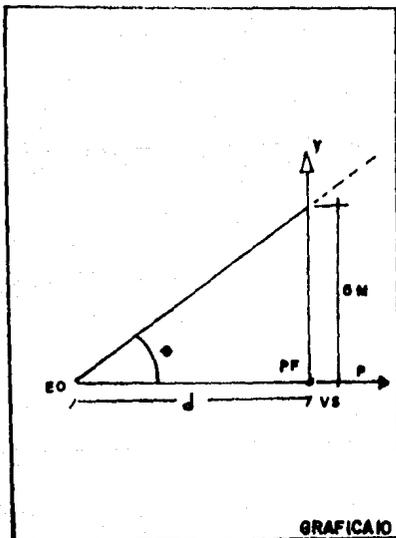
- . Visual de Simetría (VS): Es la línea que pasa por la Estación de Observación (EO) y el punto P del cubo que se proyecta al infinito; llamado así porque el Campo Visual (CV) se distribuye simétricamente, 5 módulos en ambos sentidos con respecto a esta línea.
- . Límite de Campo Visual (LCV): Son las líneas que limitan al Campo Visual (CV) que van de la Estación de Observación (EO) a los bordes de la cara frontal del cubo proyectadas hacia el infinito.
- . Distancia Focal (d): Es la distancia a la que se halla la Estación de Observación (EO). Su valor está en función de la Amplitud de Campo Visual (ACV).

Limitando el Campo Visual por el LCV y el Cubo obtenemos la Amplitud de Campo Visual (ACV) máxima del observador. Proyectando cada módulo del cubo a que se intersecten con el LCV se construye la Red, que por estar dividida en módulos se llama Red Modular (RM) (ver gráfica 8), misma que se representa por los dos siguientes planos:

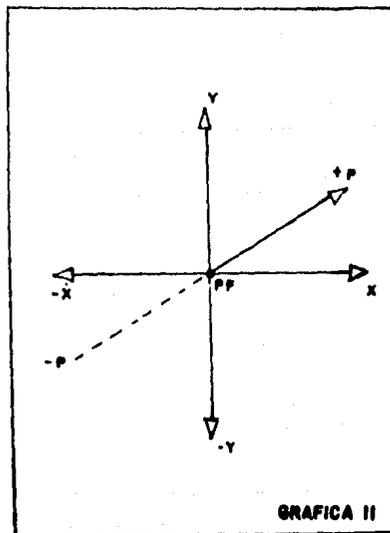
- . Plano de Simetría X (PLS/X): es la proyección ortogonal de la Red vista desde arriba, es decir, la proyección en planta del cubo que a su vez representa el Geometral (ver gráfica 9).



PLANOS DE SIMETRIA X y Y
QUE FORMAN LA RED MODULAR



CALCULO DE LA DISTANCIA FOCAL "d"



SISTEMA COORDENADO (X/Y/P) DE
RED MODULAR

Plano de Simetría Y (PLS/Y): es la proyección ortogonal de la Red vista lateralmente, es decir, representa la altura del Cuerpo Geométrico (ver gráfica 9).

A diferencia de los métodos convencionales para el trazo de perspectivas, Red Modular (RM) postula la existencia de un solo punto de fuga denotado por "PF" y localizado en la intersección de la Visual de Simetría (VS) y la cara frontal del cubo llamado Plano Proyectivo de Campo Visual (PLPCV), es decir, al centro de dicho plano el cual tiene la misma función del cuadro. Cualquier otro punto de fuga de la perspectiva es relativa al Cuerpo Geométrico (CG) y por lo tanto no hay que localizarlo, por esta razón llamamos a "PF" Punto de Fuga Unico de Red Modular. La ventaja de este punto es que siempre se localiza en el centro del PLPCV, que a su vez coincide con el origen del sistema coordenado (X/Y/P) de Red Modular (ver gráfica 6).

Una característica importante de Red Modular es que la distancia a la que se halla la Estación de Observación (EO) puede conocerse ya que está tanto en función de la Amplitud de Campo Visual (ACV) como de la magnitud del Plano Proyectivo de Campo Visual (PLPCV), por tanto determinaremos su valor; para ello usaremos cualquiera de los planos de simetría de RM (ver gráfica 9). Tomando el Plano de Simetría Y (PLS/Y) en la forma mostrada en la gráfica 10, obtenemos el valor de 'd', en el que θ representa la ACV y la magnitud del PLPCV es de 10 módulos, aplicando la ley de cosenos obtenemos la siguiente expresión:

$$d = 5 \operatorname{ang} \tan (\theta / 2) m^* \quad (1)$$

Aun cuando la Amplitud de Campo Visual (ACV) puede tomar dis

* En lo sucesivo denotamos módulos con la letra 'm'.

tintos valores angulares, debe determinarse y mantenerse constante una vez definida. En el modelo de Red Modular aquí propuesto usamos una ACV de 60 grados debido a que se aproxima bastante al ángulo visual del ojo humano. Sustituyendo este valor de θ en (1) obtenemos el valor de 'd' que debe mantenerse constante una vez determinado.

$$d = 5 \operatorname{Tg}^{-1}(30) = 8.66 \text{ m}$$

Red Modular (RM) es un espacio perfectamente simétrico, por lo que también se llama Espacio Modular (EM2)* en el que, para localizar un punto, es necesario conocer su posición, para ello se requiere de un sistema coordenado espacial, es decir, tridimensional (3D).

En la gráfica 11 se muestra el sistema coordenado tridimensional de RM que tiene las siguientes características:

- . El eje P coincide con la Visual de Simetría (VS) e indica la profundidad a la que se encuentra un punto i localizado en EM2, el sentido es positivo si se sigue la dirección de la vista, es decir, de la EO hacia el Punto de Fuga (PF) y negativo en cualquier otro caso. El sentido del eje P es inverso al de un sistema coordenado derecho ya que se ha girado 180 grados con respecto a éste resultando un sistema coordenado izquierdo.**
- . El eje X indica la posición con respecto al PF de un punto i localizado en el EM2; el punto se hallará a la derecha del PF para valores de X positivos y a la izquierda del PF en cualquier otro caso.

* El modelo de RM que estamos empleando es una segunda versión de la original y se indica con el número 2.

** Esto indica que a mayor profundidad el punto está mas alejado de la EO y viceversa.

- . Los ejes X y Y están contenidos en un mismo plano llamado Plano Proyectivo de Campo Visual (PLPCV).
- . El plano de simetría X (PLS/X) y el Plano de simetría Y (PLS/Y) quedan descritos por los planos X-P y Y-P respectivamente.

La construcción de este sistema coordenado tridimensional permite conocer la posición de un punto i cualquiera localizado en el EM2 mediante la terna de coordenadas (X/Y/P), llamadas coordenadas planimétricas ya que generalmente son leídas de planos arquitectónicos.

II. TRANSFORMACION "RED MODULAR"

II. 1 Obtención de la Transformación.

Al inicio de este trabajo se dijo que Red Modular es un método para trazar perspectivas, entendiendo a éstas como la forma de representar en un plano llamado cuadro a un Cuerpo Geométrico (CG) en la forma y disposición con que aparecen a la vista del observador desde una Estación de Observación (EO) determinada.

En Red Modular el cuadro es el Plano Proyectivo de Campo Visual (PLPCV) llamado así porque cualquier punto localizado en el Espacio Modular (EM2) que es proyectado sobre este plano queda representado por su correspondiente par coordenado. Una manera esquemática de representar esta proyección de puntos se muestra en la gráfica 12, donde el PLPCV también se le llama plano imagen y T es una función de transformación que permite pasar un punto del espacio tridimensional EM2 al espacio bidimensional PLPCV ($E3 \rightarrow E2$). De aquí que un punto localizado en el PLPCV quede definido por el par coordenado (X_0/Y_0) llamadas coordenadas proyectivas.

Paradeducir la transformación de Red Modular nos auxiliaremos de la proyección central de la gráfica 13 en la que se desea proyectar al punto Q de coordenadas planimétricas ($X/Y/P$) sobre el PLPCV. La transformación consiste en hallar sus correspondientes coordenadas proyectivas (X_0/Y_0). De la gráfica 13 tenemos las siguientes relaciones:

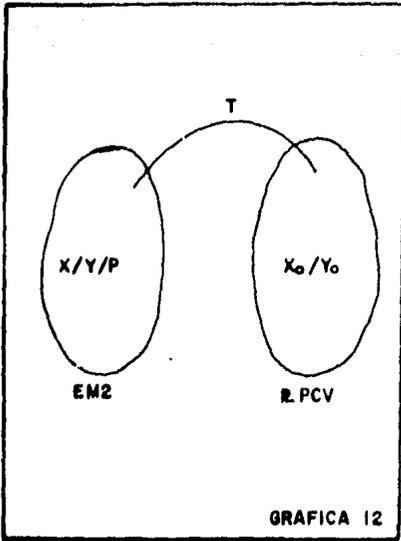
$$X_0 = \left[\frac{d + P_1}{d + P} \right] \cdot X$$

$$Y_0 = \left[\frac{d + P_1}{d + P} \right] \cdot Y$$

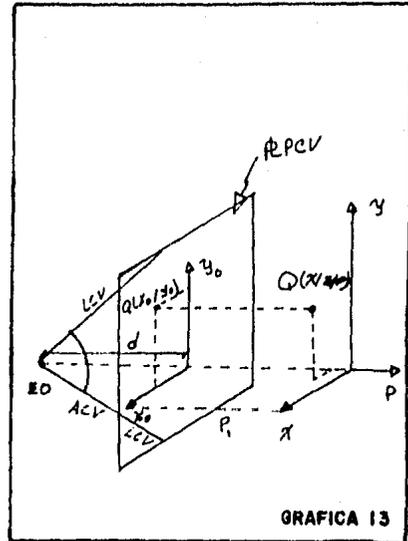
Donde P_1 es cero ya que el origen del sistema tridimensional de Red Modular coincide con el Punto de Fuga (PF) el cual se halla al centro del PLPCV, por lo tanto resultan las siguientes ecuaciones de transformación:

$$X_0 = \frac{d \cdot X}{d + P} \quad (2)$$

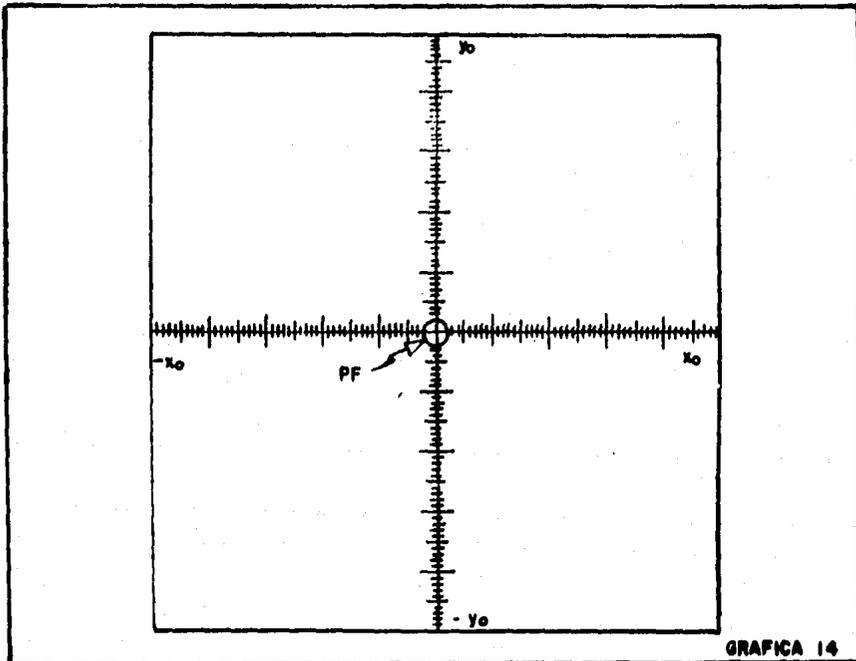
$$Y_0 = \frac{d \cdot Y}{d + P} \quad (3)$$



REPRESENTACION GRAFICA DE LA TRANSFORMACION DE RED MODULAR



PROYECCION CENTRAL DEL PUNTO Q



PLANO PROYECTIVO DE CAMPO VISUAL (R.PCV)

Nótese que si d es muy grande ($d \rightarrow \infty$), es decir, la Estación de Observación (EO) se halla en el infinito, se tiene la proyección ortogonal de ecuaciones:

$$X_o = X$$

$$Y_o = Y$$

La magnitud del PLPCV es de 10 módulos por lado por coincidir con la cara frontal del cubo en cuyo centro se localiza el punto de fuga "PF", único de RED MODULAR, mismo que coincide con el origen del sistema coordenado (X_o/Y_o) como se muestra en la gráfica 14. Este sistema coordenado está definido en el intervalo cerrado $[-5, 5]$ para ambos ejes, de tal manera que cualquier par coordenado (X_o/Y_o) con valores que salgan de este intervalo está fuera del Campo Visual del observador y por lo tanto no es posible verlo desde la Estación de Observación (EO). Por esta razón llamamos a este plano "Plano Proyectivo de Campo Visual" (PLPCV) ya que representa unicamente la proyección de puntos visibles al observador. Por lo tanto es condición necesaria que las ecuaciones 2 y 3 de cualquier punto i de coordenadas planimétricas ($X_i/Y_i/P_i$) estén dentro del intervalo cerrado $[5, -5]$ para que sus correspondientes coordenadas proyectivas (X_{oi}/Y_{oi}) cumplan las siguientes condiciones de inclusión:

$$-5 \leq X_{oi} \leq 5$$

$$-5 \leq Y_{oi} \leq 5$$

II.2 Perspectivas de cuerpos geométricos

Aun cuando hemos dado un tratamiento puntual al método de Red Modular, todo Cuerpo Geométrico localizado en el Espacio Modular (EM2) es susceptible de representarse en perspectiva, por tanto redefiniremos el concepto de Cuerpo Geométrico para tratar

lo de manera puntual y al mismo tiempo le daremos un enfoque a gráficas por computadora que nos será de gran utilidad.

Empleando teoría de gráficas* decimos que un Cuerpo Geométrico (CG) está formado por un conjunto de vértices unidos por aristas, es decir, es una gráfica de nodos conexos no disjuntos** que respetan un orden de recorrido. A su vez cada nodo o vértice es un punto de tres coordenadas planimétricas (X/Y/P), por lo que definimos a un CG de la siguiente manera:

$$CG = \left\{ (X_i/Y_i/P_i) / X_i, Y_i, P_i \in \mathbb{R} ; \forall i=1, \dots, n \right\}$$

Donde las coordenadas planimétricas de cada vértice i están definidas en los reales, y por estar localizadas en el EM2 son proyectadas en el Plano Proyectivo de Campo Visual (PLPCV) mediante la Transformación Red Modular que representamos con la notación:

$$T \left[(X_i/Y_i/P_i) \right] \rightarrow (X_{oi}/Y_{oi})$$

"T" representa la función de transformación de los n vértices y está dada por las ecuaciones 2 y 3 usadas en cada vértice del CG.

El método de Red Modular sigue una secuencia lógica de pasos para trazar perspectivas que se basan en los conceptos que hemos definido. Antes de listar estos pasos consideremos que tenemos una Red Modular trazada en algún material transparente por ejemplo plástico o mica, la cual es idéntica tanto al Plano de Simetría X (PLS/X) como al Plano de Simetría Y (PLS/Y) (ver gráfica 9) que usamos como pantalla, de tal manera que en lugar de dibujar la Red Modular sobre algún Cuerpo Geométrico basta poner la pantalla sobre éste para leer los datos correspondientes.

* Data Structures Techniques. Thomas A Standish Addison Wisley

** nodo conexo no disjunto se refiere a vértices unidos por aristas que por pertenecer a un mismo CG son no disjuntas.

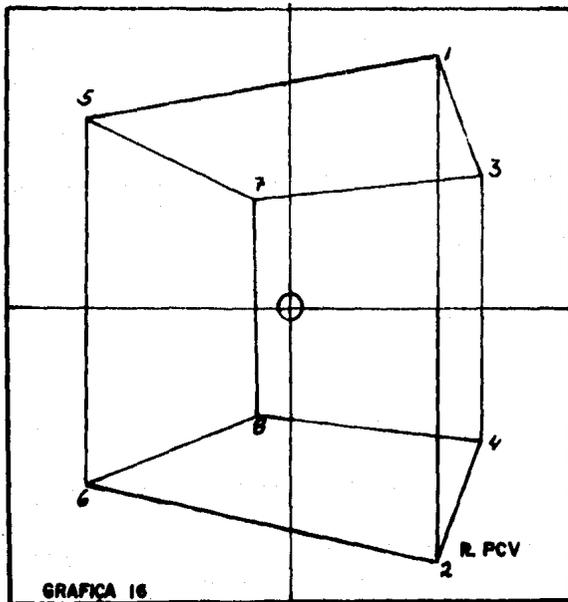
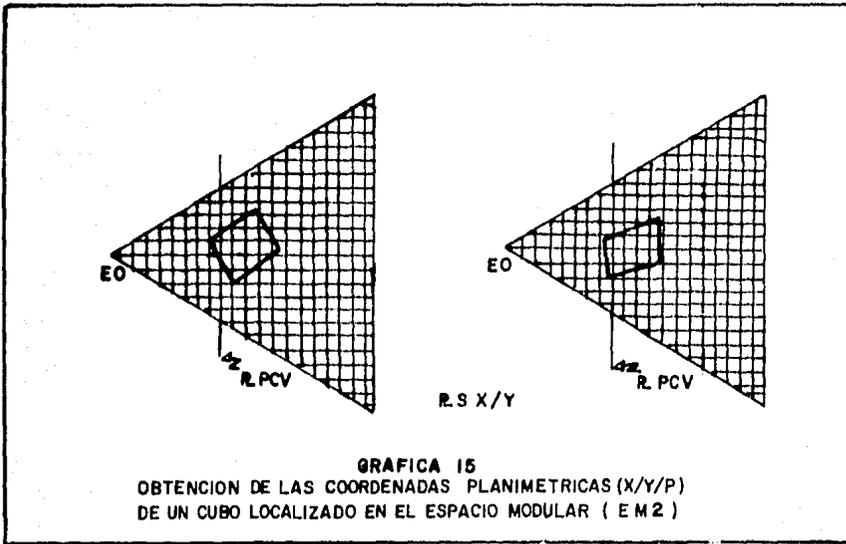
Procedimiento para trazar perspectivas según Red Modular:

- Paso 1. Seleccionar algún Cuerpo Geométrico (CG)
- Paso 2. Poner la pantalla sobre el CG de modo que quede dentro de ésta.
- Paso 3. Leer las coordenadas planimétricas (X/Y/P) de cada vértice del CG siguiendo un orden de recorrido.
- Paso 4. Aplicar la Transformación Red Modular a cada vértice leído en el paso 3 para obtener sus coordenadas proyectivas (Xo/Yo).
- Paso 5. Localizar cada par coordenado (Xo/Yo) en el Plano Proyectivo de Campo Visual (PLPCV).
- Paso 6. Unir los vértices de acuerdo al orden de recorrido establecido en el paso 3.
- Paso 8. Trazo perspectivo terminado.

Nótese que la perspectiva resultante está en función de la Estación de Observación (EO) seleccionada, es decir, el usuario tiene libertad de ubicar la EO en el lugar que considere pertinente.

Con objeto de dejar más claro el método de Red Modular obtendremos la perspectiva de un CUBO. En la gráfica 15 se muestran las pantallas de los planos de simetría X y Y de donde obtenemos la siguiente lista de datos.

PUNTO	COORDENADAS PLANIMETRICAS		
	X	Y	P
1	3	5	1.34
2	3	-5	1.34
3	7.5	5	10.24
4	7.5	-5	10.24
5	-6	5	5.71
6	-6	-5	5.71
7	-1.5	5	14.64
8	-1.5	-5	14.64



CUBO EN PERSPECTIVA

Tomando el punto 1 de coordenadas planimétricas (3/5/1.34) para aplicar la Transformación Red Modular, ecuaciones (2) y (3) tenemos:

$$X_0 = \frac{d(3)}{d + 1.34}$$

$$Y_0 = \frac{d(5)}{d + 1.34}$$

Como la Amplitud de Campo Visual (ACV) es de 60 grados el valor de "d" es 8.66 m (ecuación 1), sustituyendo este valor en las ecuaciones anteriores resulta el par de coordenadas proyectivas (2.6/4.33), Trabajando de la misma manera con los demás puntos obtenemos las coordenadas proyectivas del cubo, mismas que listamos a continuación.

PUNTO	COORDENADAS PROYECTIVAS	
	X ₀	Y ₀
1	2.6	4.33
2	2.6	-4.33
3	3.44	2.24
4	3.44	-2.24
5	-3.62	3.01
6	-3.62	-3.01
7	-0.56	1.86
8	-0.56	-1.86

Por último localizamos cada par coordenado de la tabla en el Plano Proyectivo de Campo Visual (PLPCV) y los unimos en forma correcta obteniéndose la perspectiva del CUBO como se muestra en la gráfica 16.

III. CALCULO DE GIROS Y DESPLAZAMIENTOS EN RED MODULAR

INTRODUCCION

Es de gran utilidad para el diseñador observar el resultado de su proyecto desde diferentes ángulos para corregir y mejorar detalles, lo que requiere su graficación en diferentes perspectivas. Esto en términos de Red Modular equivale a aplicar su procedimiento para cada Estación de Observación (EO) que se seleccione, sin embargo, aprovechando las características del Espacio Modular EM2 resulta equivalente mover el Cuerpo Geométrico (CG), proceso mas sencillo debido a que cualquier variación en la posición del CG localizado en el EM2 genera un nuevo conjunto de coordenadas planimétricas (X/Y/P) mismas que por tener

su correspondencia en el Plano Proyectivo de Campo Visual (PLPCV) representan a dicho cuerpo en diferentes perspectivas. Aún más a partir de los datos iniciales del CG pueden obtenerse tantas perspectivas de él como movimientos realice, conocidas como perspectivas dinámicas.

En el presente capítulo se presentan los movimientos que puede realizar un Cuerpo Geométrico localizado en el Espacio Modular EM2, basados en la cinemática del cuerpo rígido para el análisis de traslaciones y rotaciones de éste. Estos movimientos siempre se harán con respecto a ejes paralelos del Espacio Modular EM2.

Al final del capítulo se resuelve el problema de intersección de las aristas del Cuerpo Geométrico con los límites del campo visual.

III. 1 Movimientos del Cuerpo Geométrico en Red Modular.

Los movimientos que un Cuerpo Geométrico puede efectuar son dos:

- . Movimiento de Traslación
- . Movimiento de rotación

Para iniciar el estudio de estos movimientos, se considerará a un Cuerpo Geométrico (CG), localizado en Red Modular, como el conjunto de todos los puntos de coordenadas planimétricas (X/Y/P) que lo componen, pudiendo así representarse por la expresión:

$$CG = \left\{ (X_i/Y_i/P_i) / X_i, Y_i, P_i \in \mathbb{R}; i \in I \forall i=1..n \right\}$$

donde

X_i, Y_i, P_i : Son las coordenadas planimétricas del i -ésimo punto del CG y cuyo valor está contenido en el campo de los números reales.

- i : Es un número entero entre 1 y n que denota la i -ésima terna de coordenadas planimétricas.
- n : Es un número entero que indica el total de puntos que forman al CG.

En general la variación de la posición de un punto del Cuerpo Geométrico implicará la variación de la posición de los n puntos que lo componen.

III.2.1 MOVIMIENTO DE TRASLACION

El movimiento de traslación, también llamado de trayección, se presenta cuando cambia, al transcurrir el tiempo, el valor de las coordenadas planimétricas de los puntos que forman al Cuerpo Geométrico en la misma cantidad. Esta variación produce un nuevo conjunto de coordenadas proyectivas. La traslación realizada por el CG siempre se hará con respecto a ejes paralelos a los del sistema coordenado de Red Modular.

Al incremento o decremento en el valor de cualquier componente de un punto, se le denota con el símbolo delta (Δ) y los posibles casos de trayección se muestran a continuación.

- o Si se tiene una variación en la componente X denotada por ΔX , entonces la trayección será hacia la izquierda para valores de ΔX negativos, o hacia la derecha para valores ΔX positivos. En este caso sólo será variada la coordenada proyectiva X_0 , así reescribiendo la ecuación (2) de la transformada de Red Modular se tiene:

$$X_0 = \frac{d \cdot (X + \Delta X)}{P + d} \quad (12)$$

- o Si la variación es sobre la componente Y, denotada por ΔY , entonces la trayección será hacia arriba para valores de ΔY positivos, o hacia abajo para valores de ΔY negativos. Por lo que al aplicar la transformada de Red Modular sólo será variada la coordenada proyectiva Y_o , reescribiendo la ecuación (3) se tiene:

$$Y_o = \frac{d \cdot (Y + \Delta Y)}{P + d} \quad (13)$$

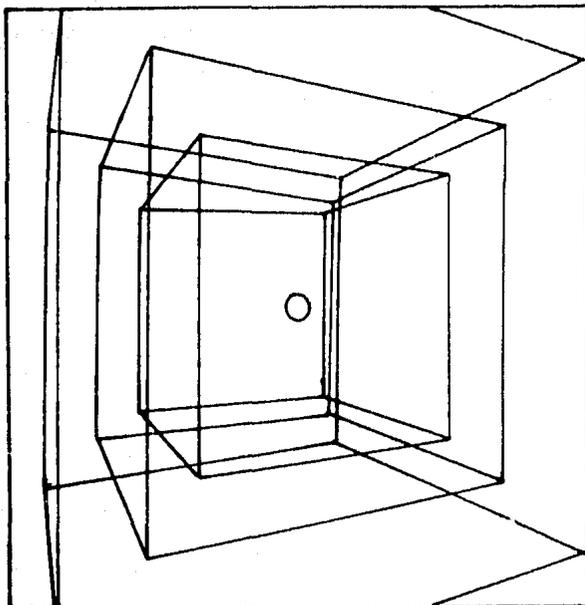
- o Si ahora la variación es en la componente P, denotado por ΔP , se tendrá un acercamiento para valores de ΔP negativos, o un alejamiento para valores de ΔP positivos. Ya que tanto las ecuaciones (1) y (2), de la transformación Red Modular, están en función de los valores de profundidad (P), éste par de ecuaciones será reescrito como:

$$X_o = \frac{d \cdot X}{(P + \Delta P) + d}$$

$$Y_o = \frac{d \cdot Y}{(P + \Delta P) + d}$$

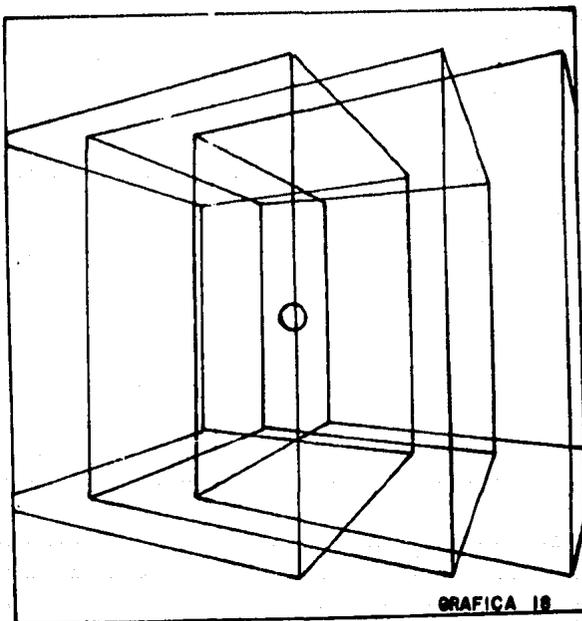
Para mostrar este tipo de movimiento se hará una serie de ejemplos.

Supóngase que se tiene el cubo de la gráfica 16, el cual se rá trayectado para $\Delta P=5m$ y para $\Delta P=3.98m$ cuyos valores coordenados se muestran a continuación:



CUBO DESPLAZADO CON
RESPECTO A SU POSICION
ORIGINAL:
 $\Delta P = 5m$ (ALEJAMIENTO)
 $\Delta P = -3.98$ (ACERCAMIENTO)

GRAFICA 17



CUBO DESPLAZADO CON
RESPECTO A SU POSICION
ORIGINAL:
 $\Delta X = 3$ (DERECHA)
 $\Delta X = -3$ (IZQUIERDA)

GRAFICA 18

PUNTO	X	Y	P	Xo	Yo	Xo	Yo	Xo	Yo
1	3	5	1.34	2.6	4.33	1.73	2.88	4.32	7.19
2	3	-5	1.34	2.6	-4.33	1.73	-2.88	4.32	-7.19
3	7.5	5	10.24	3.44	2.24	2.71	1.8	4.35	2.9
4	7.5	-5	10.24	3.44	-2.24	2.71	-1.8	4.35	-2.9
5	-6	5	13.5	-3.62	3.01	-2.68	2.23	-5	4.17
6	-6	-5	13.5	-3.62	-3.01	-2.68	-2.23	5	-4.17
7	-1.5	5	1.64	-0.56	1.86	-0.45	1.53	-0.67	2.29
8	-1.5	-5	1.64	0.50	-1.86	-0.45	-1.53	-0.67	-2.29

$$\Delta P = 5m \quad \Delta P = 3.98m$$

graficando estos valores proyectivos en el plano proyectivo de campo visual resulta la gráfica 17.

Tomando el mismo cubo y variando la coordenada X se tiene:

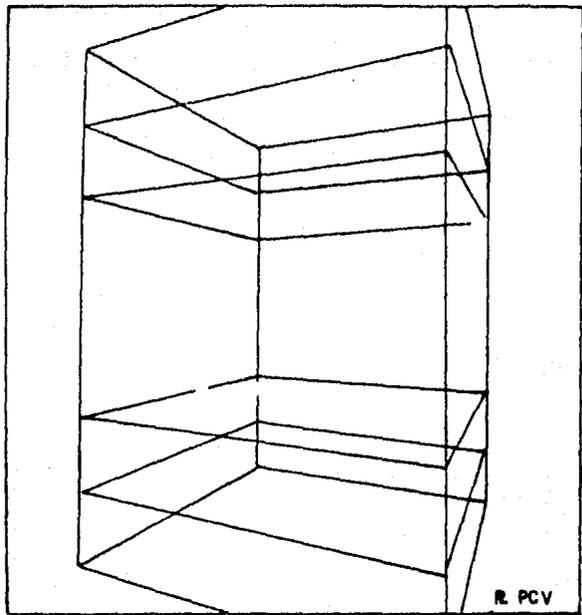
	$\Delta X=3$	$\Delta X=-3$
Yo	Xo	Xo
4.33	5.19	0
-4.33	5.19	0
2.29	4.79	2.05
-2.29	4.79	2.05
3.01	-5.42	-5.42
-3.01	-5.42	-5.42
1.86	0.55	-1.67
1.86	0.55	1.67

la gráfica 18 muestra el resultado perspectivo.

Haciendo el caso análogo al anterior, pero para la coordenada Y se tiene:

**CUBO DESPLAZADO CON
RESPECTO A SU POSICION
ORIGINAL:**

**$\Delta y = 2$ (ARRIBA)
 $\Delta y = -2$ (ABAJO)**



GRAFICA 19

Xo	Yo	$\Delta Y=2$	$\Delta Y=-2$
		Yo	Yo
2.6	4.33	6.06	2.59
2.6	-4.33	-2.59	-6.06
3.44	2.29	3.19	-11.37
3.44	-2.29	-1.37	-3.19
-3.62	3.01	4.21	1.8
-3.62	-3.01	-1.8	-4.21
-0.56	.86	2.6	1.11
-0.56	-1.86	-1.11	-2.6

la gráfica 19 muestra el resultado perspectivo

III.2.2 MOVIMIENTO GIRATORIO

Al ser Red Modular un sistema coordinado, llamado espacio modular (EM2), un punto localizado en EM2 podría girar en torno a cualquiera de sus tres ejes (X/Y/P).

Puede considerarse que el movimiento de giro siempre se realiza en torno a un eje^{*}, tomando como referencia un origen, que no necesariamente es el del sistema coordinado, por lo que será considerado como movimiento relativo similar al que se trata en Cinemática del Cuerpo Rígido.

Por lo que para cada giro habrá que determinar:

- o El origen relativo del giro
- o El ángulo que se desea girar

Debido a que este movimiento involucra un ángulo de giro, su tratamiento será más sencillo si se emplean coordenadas polares.

(*) El eje de giro se determina por un punto en el plano perpendicular a él, de esta manera se puede decir que se requiere un origen para ejecutar el giro.

Un punto definido por sus coordenadas planimétricas (X/Y/P) podrá girar en torno a los ejes como se indica en la gráfica 20 en donde Q es el origen de giro, sobre el cual ha sido trazado un sistema coordenado relativo al de Red Modular.

Giro sobre el eje Y.

De la gráfica 20.1 se obtienen los parametros.

$$\Delta x = X_p - X_q \quad : \text{ Amplitud en el eje X}$$

$$\Delta p = P_p - P_q \quad : \text{ Amplitud en el eje P.}$$

El valor del ángulo se obtiene por trigonometría como:

$$\theta_y = \text{ang} \tan \frac{\Delta p}{\Delta x}$$

es un radio vector, por lo que tendrá magnitud y sentido y está dado por las expresiones

$$\vec{r} = \frac{\Delta p}{\text{SEN} \theta_y}$$

$$\vec{r} = \frac{\Delta x}{\text{COS} \theta_y}$$

Para cada giro, habrá que proporcionar un incremento o decremento en el valor del ángulo θ_y donotado por $\Delta \theta_y$, y con este valor se obtienen las coordenadas planimétricas giradas, denotadas por (Xg/Yg/Pg) en forma cartesiana con las relaciones:

$$X_g = X_Q + \vec{r} \text{ COS} (\theta_y + \Delta \theta_y)$$

$$Y_g = \text{CONSTANTE}$$

$$P_g = P_Q + \vec{r} \text{ SEN} (\theta_y + \Delta \theta_y)$$

y con la transformación Red Modular se obtienen las coordenadas proyectivas:

$$X_o = \frac{X_g \cdot d}{P_g + d}$$

$$Y_o = \frac{Y_g \cdot d}{P_g + d}$$

Nótese que aún cuando Y_g se mantuvo constante, su valor proyectivo se vio afectado por el valor de la profundidad (P_g).

Giro sobre el eje P.

Con el procedimiento anterior sobre la gráfica 20.2, se obtiene:

$$A_y = Y_q - Y_p \quad : \text{ Amplitud en Y}$$

$$A_x = X_q - X_p \quad : \text{ Amplitud en X}$$

por lo tanto

$$\theta_p = \text{Ang} \tan \frac{A_y}{A_x}$$

y el radio vector queda determinado por:

$$r = \frac{A_y}{\text{SEN} \theta_p} = \frac{A_x}{\text{COS} \theta_p}$$

Considerando el valor de $\Delta \theta_p$, se obtienen las coordenadas planimétricas.

$$X_g = X_q + r \text{ COS } (\theta_p + \Delta \theta_p)$$

$$Y_g = Y_q + r \text{ SEN } (\theta_p + \Delta \theta_p)$$

$$P_g = \text{CONSTANTE}$$

y aplicando la transformación Red Modular se tiene:

$$X_o = \frac{X_g \cdot d}{P_g + d}$$

$$Y_o = \frac{Y_g \cdot d}{P_g + d}$$

Giro sobre el eje X

Haciendo el mismo razonamiento que en los casos anteriores se tiene (ver gráfica 20.3):

$$A_y = Y_q - Y_p \quad : \quad \text{Amplitud de Y}$$

$$A_p = P_q - P_p \quad : \quad \text{Amplitud de profundidad P}$$

con lo que el ángulo de posición es:

$$\theta_x = \text{ang tan } \frac{A_y}{A_p}$$

y el radio vector está dado por

$$\vec{r} = \frac{A_y}{\text{SEN}\theta_x}$$

6

$$\vec{r} = \frac{A_p}{\text{COS}\theta_x}$$

Las coordenadas planimétricas quedan como:

$$X_g = \text{CONSTANTE}$$

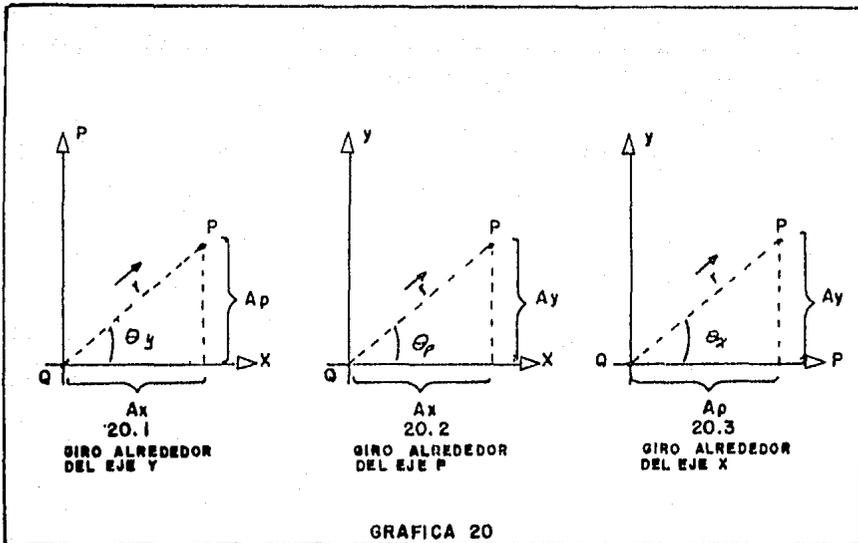
$$Y_g = Y_q + \vec{r} \text{ SEN } (\theta_x + \Delta\theta_x)$$

$$P_g = P_q + \vec{r} \text{ COS } (\theta_x + \Delta\theta_x)$$

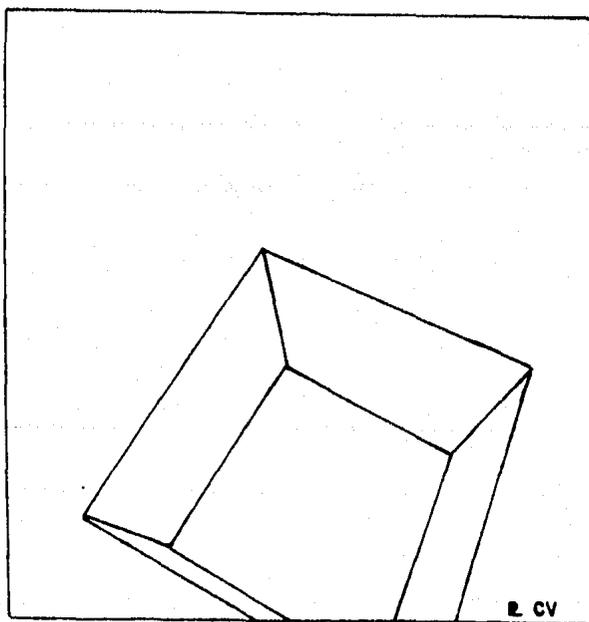
de donde las coordenadas proyectivas son:

$$X_o = \frac{X_g \cdot d}{P_g + d}$$

$$Y_o = \frac{Y_g \cdot d}{P_g + d}$$



GIROS DE UN PUNTO Q EN RED MODULAR



GRAFICA 21

GIRO DEL CUBO DE 60° SOBRE EL EJE X
 TOMANDO COMO ORIGEN (-1.5/0/14.64)

Nótese que al igual que en el giro sobre el eje Y la coordenada (X_0) se ve afectada por el valor de la profundidad (P_g).

Se ilustrará el movimiento giratorio a través de un ejemplo, más ejemplos de ellos serán presentados en el siguiente capítulo.

Supóngase que se tiene el cubo de la gráfica 16, cuyas coordenadas planimétricas se indican en la tabla de abajo y se desea girar sobre el eje X 60° . Tomando como origen el punto Q de coordenadas $(-1.5/0/14.64)$.

PUNTO	X	Y	P
1	3	5	1.34
2	3	-5	1.34
3	7.5	5	10.24
4	7.5	-5	10.24
5	-6	5	5.71
6	-6	-5	5.71
7	-1.5	5	14.64
8	-1.5	-5	14.64

Resolviendo para el punto 1 de coordenadas $(3/5/1.34)$ se tiene:

$$A_y = 5 - 0 = 5$$

$$A_p = 1.34 - 14.64 = 13.3$$

$$\theta_x = \text{tg}^{-1} \left[\frac{5}{-13.3} \right] = 20.63^\circ$$

$$\overline{r} = \frac{5}{\text{SEN}(-20.61)} = 14.20$$

Haciendo $\Delta\theta_x=60^\circ$, se obtiene las coordenadas planimétricas.

$$X_g=3m$$

$$Y_g=0-14.20 \text{ SEN } (39.39) =-9.01m$$

$$P_g=14.64-14.20 \text{ COS } (39.39)=3.63m$$

Aplicando la transformación Red Modular se tiene:

$$X_o = \frac{3 (8.66)}{3.63+8.66} = 2.11 \text{ m}$$

$$Y_o = \frac{-9 (8.66)}{3.63+8.66} = -6.34 \text{ m}$$

De igual forma se obtuvieron los valores de las otras coordenadas proyectivas, resultando el trazo perspectivo de la Gráfica 21.

III. 2. Coordenadas Límite

Para saber si un punto está dentro del campo visual, las coordenadas proyectivas (X_o/Y_o) tiene que cumplir las siguientes condiciones de inclusión:

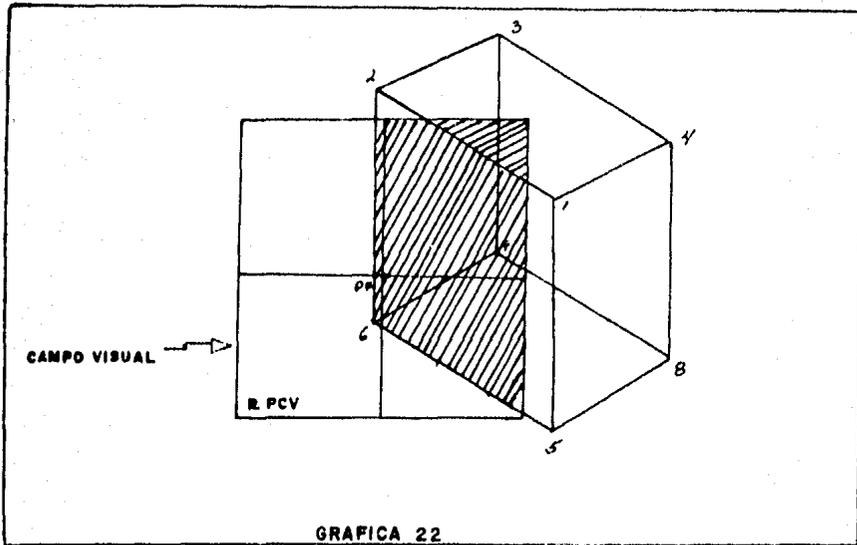
$$-5m \leq X_o \leq 5m$$

$$-5m \leq Y_o \leq 5m$$

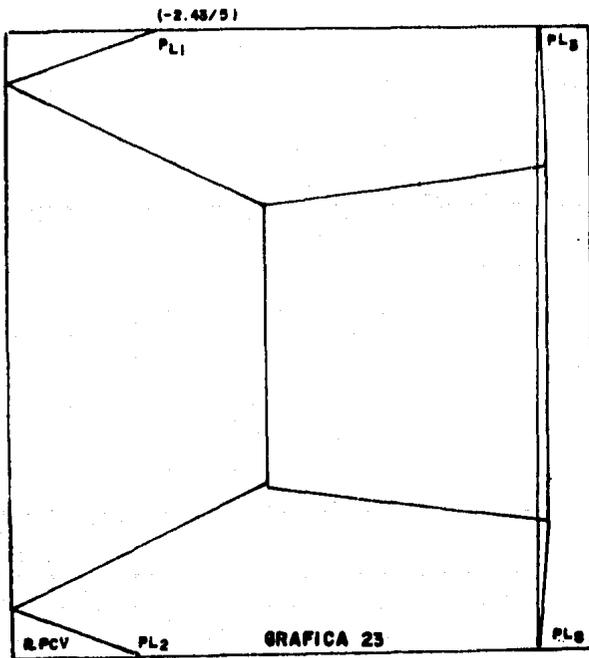
donde X_o y Y_o estan dadas por las ecuaciones (2) y (3)

$$X_o = \frac{d \cdot X}{P + d} \quad (2)$$

$$Y_o = \frac{d \cdot Y}{P + d} \quad (3)$$



**GASOS DE INTERSECCION DEL CUERPO GEOMETRICO
CON EL CAMPO VISUAL (CLIPPING)**



**CUBO PARCIALMENTE FUERA
DE CAMPO VISUAL**

De tal forma que si un punto i cualquiera de coordenadas proyectivas (X_{oi}/Y_{oi}) no cumple las condiciones de inclusión (1) y (2), se tendrá un punto fuera de campo visual y por estar unido a otros puntos del Cuerpo Geométrico, habrá que hallar sus coordenadas límite denotadas por (X_{oL}/Y_{oL}) .

Para la deducción de coordenadas límite obsérvese la gráfica 22 en la cual el cubo está parcialmente fuera del campo visual, teniéndose dos casos límite:

- Dos puntos de intersección del Cuerpo Geométrico con el campo visual, como el caso de la recta $\overline{12}$.
- Un solo punto de intersección del Cuerpo Geométrico con el campo visual, como es el caso de las rectas $\overline{65}$, $\overline{78}$, $\overline{26}$ y $\overline{37}$.

Las rectas $\overline{34}$, $\overline{48}$, $\overline{58}$ y $\overline{15}$ por estar totalmente fuera del campo visual no tienen coordenadas límite.

Para hallar las coordenadas límite se requiere:

- La ecuación de la línea que se intersecta con el campo visual que en general está dada por la ecuación:

$$Y = mX + b$$

la cual puede escribirse como:

$$Y_{o1} = mX_{o1} + b \quad (4)$$

donde X_{o1} y Y_{o1} son las coordenadas límite:

m : es la pendiente dada por:

$$m = \frac{Y_{o2} - Y_{o1}}{X_{o2} - X_{o1}} \quad (5)$$

b: es la ordenada al origen y está dado por:

$$b = Y_{o_1} - mX_{o_1} \quad (6)$$

Nota: Los índices 1 y 2 denotan los puntos 1 y 2, pero puede suceder entre cualquier par de puntos i y j del Cuerpo Geométrico.

Una vez determinada la ecuación de la línea hay que determinar el valor de la coordenadas límite, procediendo de la siguiente manera:

Prueba para la coordenada X_{oi} .

Si $X_{oi} > 5$ tomar el valor límite $X_{o_L} = 5$

Si $X_{oi} < -5$ tomar el valor límite $X_{o_L} = -5$.

sustituir X_{o_L} en la ecuación (4) y calcular Y_{o_L} . Si el valor resultante de Y_{o_L} no cumple la condición (2) implica que tanto X_{o_L} como Y_{o_L} están fuera del campo visual y no existe punto límite, de lo contrario se tendrá el punto de coordenadas proyectivas límite (X_{o_L}/Y_{o_L}) .

Ahora bien, en caso de que la coordenada X_{oi} cumpla con la condición (1) se hará:

Prueba para la coordenada Y_{oi} :

Si $Y_{oi} > 5$ tomar el valor límite $Y_{o_L} = 5$.

Si $Y_{oi} < -5$ tomar el valor límite $Y_{o_L} = -5$.

sustituir Y_{o_L} en la ecuación (4) y calcular X_{o_L} . Si el valor resultante de X_{o_L} no cumple la condición (1) implica que tanto X_{o_L} como Y_{o_L} están fuera del campo visual y por lo tanto no existe punto límite, en caso contrario se tendrá el punto límite de coordenadas proyectivas (X_{o_L}/Y_{o_L}) .

Como ejemplo supóngase que el cubo de la gráfica 16 realizó

un movimiento resultando las siguientes coordenadas proyectivas:

PUNTO	Xo	Yo
1	4.32	7.19
2	4.32	-7.19
3	4.35	2.29
4	4.35	-2.29
5	-5	4.17
6	-5	4.17
7	-0.67	2.29
8	-0.67	-2.29

Tomando los puntos 1 y 5, por estar fuera del Campo Visual se tiene:

$$P1 \quad (4.32/7.19)$$

$$P2 \quad (-5/4.17)$$

de la ecuación (5) se determina la pendiente:

$$m = \frac{4.17 - 7.19}{-5 - 4.32} = 0.324 \text{ m}$$

de la ecuación (6) se obtiene la ordenada al origen:

$$b = 7.19 - (0.324)(4.32) = 5.7903 \text{ m}$$

resultando la siguiente ecuación:

$$Yo_L = (0.324) Xo_L + 5.7903$$

Como siguiente paso se determinan las coordenadas límite:

. Para el punto 1.

$Xo_1 = 4.32$ cumple la condición (1) por lo tanto está dentro del campo visual.

$Y_{o_1} = 7.14 > 5$ está fuera de campo visual, por lo tanto hay que asignarle el valor límite

$$Y_{o_L} = 5.$$

Sustituyendo Y_{o_L} en la ecuación (4) se tiene:

$$X_{o_L} = \frac{5 - 5.7903}{0.324} = -2.43m$$

Como X_{o_L} cumple la condición (1) se tiene el punto límite de coordenadas proyectivas $(-2.43/5)$

Para el punto 5.

Como $X_{o_5} = -5$ y $Y_{o_5} = 4.17$ cumplen las condiciones (1) y (2) respectivamente están dentro del campo visual.

De igual manera se obtuvieron las coordenadas límite entre las rectas* de los puntos $\overline{13}$, $\overline{12}$, $\overline{42}$ y $\overline{62}$, y con ellos la gráfica 23.

RECTA	X_{o_L}	Y_{o_L}
$\overline{P1P5}$	-2.43	5
$\overline{P6P2}$	-2.43	-5
$\overline{P1P3}$	4.33	5
$\overline{P2P1}$	4.33	-5

La determinación de coordenadas límite o de intersección del Cuerpo Geométrico con el Plano Proyectivo de Campo Visual (PLPCV), son de gran utilidad ya que la gráfica se va a dibujar en una pantalla graficadora controlada por una computadora y no manualmente, de tal manera que el algoritmo debe ser capaz de resolver estos casos para el trazo correcto de la gráfica.

No se debe perder de vista que en esta primera parte del trabajo se intentó establecer un "modelo" de graficación basado

(*) Se usa la acepción de recta, aunque en realidad es un segmento de recta.

en Red Modular, para ello fué necesario, además de la Transformación Red Modular, desarrollar las ecuaciones y algoritmos de movimientos y coordenadas límite, ya que estos serán empleados en la implementación del sistema computarizado Red Modular que exponemos en el siguiente capítulo.

IV. DISEÑO DE UN SISTEMA CAD UTILIZANDO EL METODO RED MODULAR.

INTRODUCCION

Red Modular es un método para el trazo de perspectivas de Cuerpos Geométricos, que sirve como herramienta en el diseño arquitectónico* ya que se puede mover al cuerpo geométrico y observarlo desde diferentes ángulos, permitiendo así corregir y/o mejorar detalles de diseño.

(*) Aun cuando fue desarrollado para estos fines puede usarse en otras áreas como mecánica, ingeniería civil, etc.

Dentro de las características principales de este método se listan las siguientes:

- . La Red está dividida en módulos para trabajar a la escala deseada (metros, pies, pulgadas, centímetros, etc.)
- . La posición de la Estación de Observación siempre es conocida y se localiza a "d" módulos del Espacio Modular (EM2).
- . La Amplitud de Campo Visual es conocida (por convención 60 grados) por tanto es posible limitar el espacio físico de todo fenómeno de perspectiva que se presente en la Red.
- . Un punto puede localizarse en la Red mediante la terna de coordenadas planimétricas (X/Y/P).
- . Todo Cuerpo Geométrico localizado en la Red es susceptible de moverse, y obtener su resultado perspectivo a partir de la entrada inicial de datos (coordenadas planimétricas).
- . Los procedimientos tanto para el trazo perspectivo como para la ejecución de movimientos y cálculo de coordenadas límite se dan en forma de algoritmos, es decir, una secuencia finita de pasos lógicos que conducen a un fin determinado.

Además desde el punto de vista de gráficas por computadora: el método Red Modular tiene las siguientes ventajas:

- . Tiene un sistema coordinado de dos dimensiones llamado Plano Proyectivo de Campo Visual (PLPCV)
- . El Plano Proyectivo de Campo Visual (PLPCV) funciona como ventana.

- . Se tiene siempre el mismo factor de escala, dado por las condiciones de inclusión o valores límite representados por el intervalo cerrado $[-5,5]$.
- . Las coordenadas límite (X_{o_L}/Y_{o_L}) resuelven el caso de intersección con la ventana (clipping).
- . Al definir a un Cuerpo Geométrico como una gráfica de nodos conexos no disjuntos (vértices) que respetan un orden de recorrido, donde cada vértice se localiza en la Red por su correspondiente terna de coordenadas planimétricas $(X/Y/P)$, es posible representar a cualquier Cuerpo Geométrico localizado en la Red por el conjunto de nodos que lo describen.
- . Todo punto de coordenadas planimétricas $(X/Y/P)$ se transforma, en el Plano Proyectivo de Campo Visual (PLPCV), en el par de coordenadas proyectivas (X_o/Y_o) , mediante la transformación "Red Modular". Por tanto todo Cuerpo Geométrico localizado en el Espacio Modular (EM2), por estar formado por un conjunto de vértices, puede pasar al PLPCV teniendo su resultado perspectivo.
- . El límite de Campo Visual (LCV) determina la parte del Cuerpo Geométrico visible al observador y en caso de estar parcialmente fuera de él es posible conocer los puntos de intersección.

Todas estas ventajas del método Red Modular permitieron diseñar un "modelo de graficación"^{*} susceptible de implementarse

(*) Objetivo de la primera parte de la investigación.

en una computadora. A este "modelo" bastará sumarle el software necesario a fin de obtener un sistema de diseño asistido por computadora (CAD).**

Todo sistema de diseño asistido por computadora (CAD) y en general de información está definido como un par

$$P = (L, I)$$

donde L es un lenguaje e I es un intérprete para ese lenguaje.

En el sistema que desarrollamos para el método Red Modular, L es Algol-Fortran e I es el conjunto de comandos de control y primitivos*** de graficación. Los primeros permiten el diálogo hombre-máquina más próximo con la computadora, los segundos hacen factible la construcción de gráficas.

En el presente capítulo se presenta la organización y estructura del sistema Red Modular así como los algoritmos empleados para controlar cada proceso que éste realiza.

IV. 1 Organización del sistema Red Modular (RM)

El sistema RM está formado por un conjunto de rutinas agrupadas en bloques o módulos que ejecutan funciones determinadas para cumplir un fin específico, lo que permite:

- Tener control sobre cada proceso que el sistema realiza.
- Tener procesos completamente independientes, es decir, tareas disjuntas.
- Optimizar memoria, ya que no se tiene todo el sistema presente sino solo el módulo solicitado.

(**) Traducción literal de Computer-aided design (CAD)

(***) El concepto de primitivos se verá ampliamente al tratar graficación.

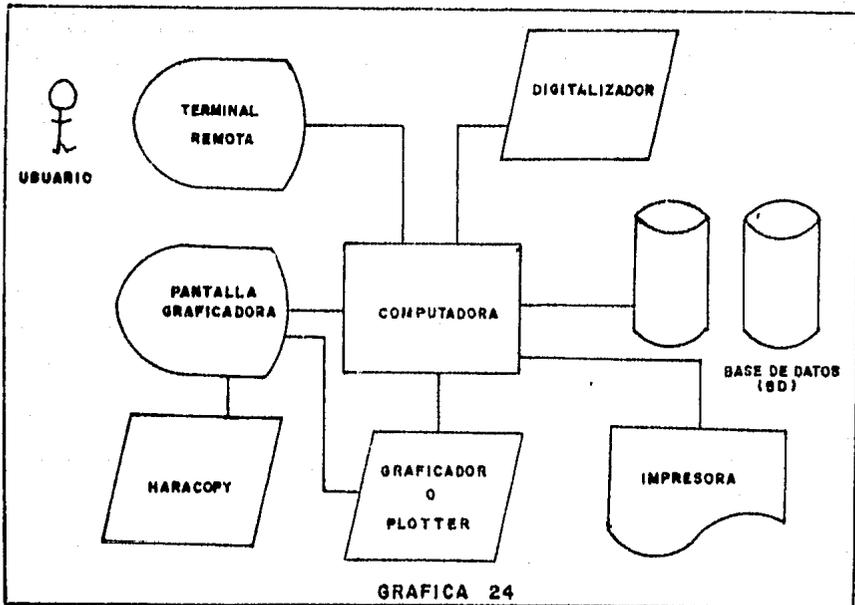
- . Programación estructurada.
- . Detectar fácilmente fallas o errores del sistema.
- . Aumentar nuevos módulos al sistema sin alterar la estructura interna del mismo.
- . Mayor comprensión de lo que se está haciendo.

Los módulos de que consta el sistema son:

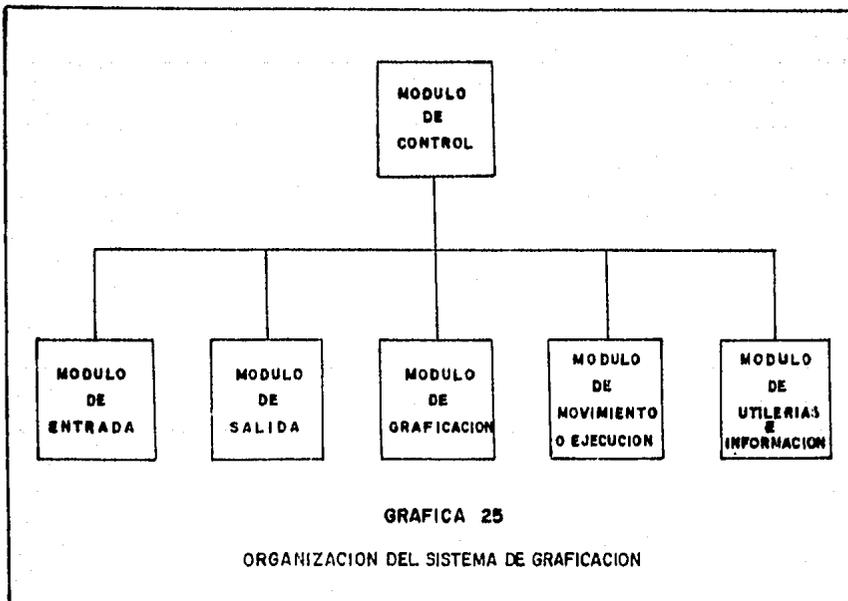
- . Módulo maestro o de control.
- . Módulo de entrada
- . Módulo de salida
- . Módulo de graficación
- . Módulo de movimiento o ejecución
- . Módulo de utilerías o información

Antes de describir con detalle cada módulo del sistema véase la gráfica 24 en la que se muestran las acciones que puede ejecutar el usuario a través de una terminal remota conectada a una computadora central (host) en donde radica el sistema: Estas acciones permiten:

- . Introducir datos de un Cuerpo Geométrico a la computadora ya sea por terminal, por digitalizador o por algún archivo en disco.
- . Obtener datos de algún Cuerpo Geométrico registrado en la Base de Datos del sistema o bien.
- . Registrar datos (alta) de algún Cuerpo Geométrico en la Base de Datos.
- . Imprimir el valor numérico tanto de las coordenadas proyectivas (X_0/Y_0) como planimétricas ($X/Y/P$) del Cuerpo Geométrico deseado.
- . Desplegar en pantalla en forma de tabla, los valores numéricos de las coordenadas planimétricas y proyectivas del Cuerpo Geométrico.
- . Graficar la perspectiva del Cuerpo Geométrico ya sea por pantalla graficadora, por graficador o por ambos.



ACCIONES QUE PERMITE EJECUTAR EL SISTEMA DE MANERA INTERACTIVA MEDIANTE UNA TERMINAL REMOTA



- . Mover tanto en traslación como en giro al Cuerpo Geométrico y graficar la perspectiva resultante.
- . Obtener la perspectiva de algún Cuerpo Geométrico a diferentes aberturas de Campo Visual.

Para llevar a cabo todas estas acciones, el sistema se encuentra organizado en la forma como se muestra en la gráfica 25. Donde el módulo maestro funciona como árbitro de decisiones que envía el control a otros módulos en función de la acción solicitada.

Como se observa en la gráfica 25 los módulos se encuentran interconectados entre si, por lo que pueden trabajar en forma independiente pasando por el módulo maestro o directamente.

Esta independencia e interdependencia de módulos, sin embargo, debe seguir una secuencia lógica para llegar al objetivo deseado en el menor número de pasos posibles.

Aun cuando el sistema posee los mecanismos de control que ejecutan o no una acción determinada (entrada, graficación, etc.) es conveniente respetar los niveles de jerarquía en cuanto a ejecución de los módulos para llegar al resultado deseado casi instantáneamente. Los niveles de jerarquía se muestran a continuación:

- nivel 1: Módulo maestro o de control
- nivel 2: Módulo de entrada
Módulo de utilerías o información
- nivel 3: Módulo de graficación
Módulo de salida
Módulo de ejecución de movimientos

Los módulos de nivel 2 dependen solo del nivel 1 mientras que los módulos de nivel 3 dependen de los niveles 1 y 2. Se tiene interconexión tanto entre los módulos de nivel 3 como en-

tre los módulos de niveles 2 y 3.

La razón de jerarquizar cada módulo del sistema en niveles de ejecución, se debe a la abstracción lógica del pensamiento humano para llegar a un fin determinado a través del proceso:

Idea \longrightarrow recurso \longrightarrow fin

es decir, partir de una idea original, ver que recursos se tienen sobre dicha idea para obtener el fin deseado. Obsérvese que si se altera la jerarquía de los elementos el resultado puede ser insatisfactorio.

En el sistema el módulo maestro es el portavoz del usuario, es él quien transmite la idea de éste para llegar a alguno o algunos de los siguientes fines:

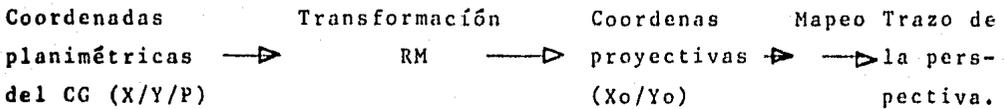
- . Graficar perspectivas
- . Almacenar información (datos)
- . Imprimir información (datos)
- . Desplegar información (datos)

de algún Cuerpo Geométrico, siguiendo la secuencia lógica:

Cuerpo Geométrico (CG) \longrightarrow Datos del CG \longrightarrow fin

y de esta manera obtener el resultado deseado.

El sistema por estar basado en el método Red Modular, sigue la misma filosofía que éste, es decir, se trazará la perspectiva de un Cuerpo Geométrico a partir de sus coordenadas planimétricas (X/Y/P) aplicando la transformación RM para obtener sus correspondientes coordenadas proyectivas (Xo/Yo) en el Plano Proyectivo de Campo Visual (PLPCV) para que unidos los puntos en forma correcta se tenga la perspectiva deseada. Resumiendo se tiene la siguiente secuencia lógica:



Este tipo de proposiciones lógicas que se siguen para llegar a un fin determinado son posibles debido a la organización modular o de bloques del sistema, que aun cuando actúan de manera independiente o conjunta utilizan el mismo medio de comunicación para ejecutar alguna acción. Esta forma de comunicación se explica a continuación:

Ya que el usuario desea obtener perspectivas de un Cuerpo Geométrico* (objeto) éste deberá tener asociado un nombre que lo identifique. El diseñador (usuario) generalmente da un nombre a cada uno de sus proyectos, por lo tanto le es familiar el uso de nombres, por otro lado el sistema lo utiliza para asociarlo a un grupo de datos.

Para entender cómo se controla esta asociatividad nombre-datos definiremos algunos conceptos para establecer un denominador común y usarlos en lo sucesivo.

Identificador: Es el nombre que el usuario asigna a un Cuerpo Geométrico por medio del cual el sistema reconoce un conjunto de datos. (generalmente es el mismo que el nombre del proyecto).

Dato: Es una terna de coordenadas planimétricas de un Cuerpo Geométrico localizado en el espacio modular (EM2).

Datos: Es el conjunto de coordenadas planimétricas que describen al Cuerpo Geométrico.

(*) Para ser consistentes con la terminología de Red Modular.

Texto: Es un conjunto de una o mas líneas de texto opcionales, cuyo objetivo es documentar el proyecto con información de los objetivos, lugar de ubicación, etc. que lo describen a grandes rasgos.

Programa: Es el conjunto de datos y textos registrados bajo un identificador que en forma algebraica se representa por la expresión:

Programa = Identificador + Datos + Texto

Por lo que diremos que el usuario trabaja con programas, aun que realmente son archivos de trabajo (en disco).

Hasta aquí dejamos abierta la definición de los conceptos an tes mencionados ya que serán analizados en detalle en la descripción de cada módulo. Esta descripción se hará conforme al nivel de jerarquía al que pertenece cada módulo empezando con los de ni vel 1 y terminando con los de nivel 3 para seguir el orden lógico establecido en el diseño del sistema.

IV. 2 DESCRIPCION DE MODULOS

MODULO MAESTRO O DE CONTROL

El módulo maestro controla todos y cada uno de los procesos que el usuario solicita. Dispone de un conjunto de comandos de control que permiten la comunicación usuario-recurso. Donde el término recursos denota cualquier proceso requerido por la computadora: entrada, salida, graficación, digitalización, etc.

Un comando de control es una palabra definida en la tabla de comandos del sistema que permite llevar a cabo una acción específica de acuerdo al comando enviado. Una vez detectada la petición solicitada por el comando, el control es pasado a los módulos de niveles dos o tres y atendida la petición en el módulo correspondiente, el control es enviado nuevamente al módulo maestro. Nótese que la acción de decidir a donde enviar el control, es decir que proceso activar, es la que determina su nivel de jerarquía y no de importancia.

El módulo maestro, además, tiene un monitor cuya función es mantener informado al usuario del estado de la petición solicitada, para ello se manejan tres tipos de mensajes:

Estado: Estos mensajes informan de la acción que se está haciendo como: lectura de datos, búsqueda de programas, explicación del sistema, etc.

Aviso: Este tipo de mensajes pone sobre aviso al usuario que se detectó alguna falla y que por lo tanto no se podrán realizar algunas acciones; graficador deshabilitado, programa sin datos, no existe programa; son algunos ejemplos de avisos.

Error: Este tipo de mensajes indican que se cometió un error en el proceso solicitado o en la sintaxis del comando empleado, se tendrá un error si los archivos no están

presentes; si se usan comandos no válidos; si se buscan o habilitan programas que no existan en la biblioteca del sistema, etc.

El monitor permite llevar la secuencia lógica de un proceso, ya que si se detecta alguna falla no solo se indica que ésta ocurrió sino también el porqué de la misma, de manera que con los mensajes enviados, el usuario puede ejecutar correctamente la acción deseada. El algoritmo que a continuación se muestra, indica el funcionamiento del módulo maestro.

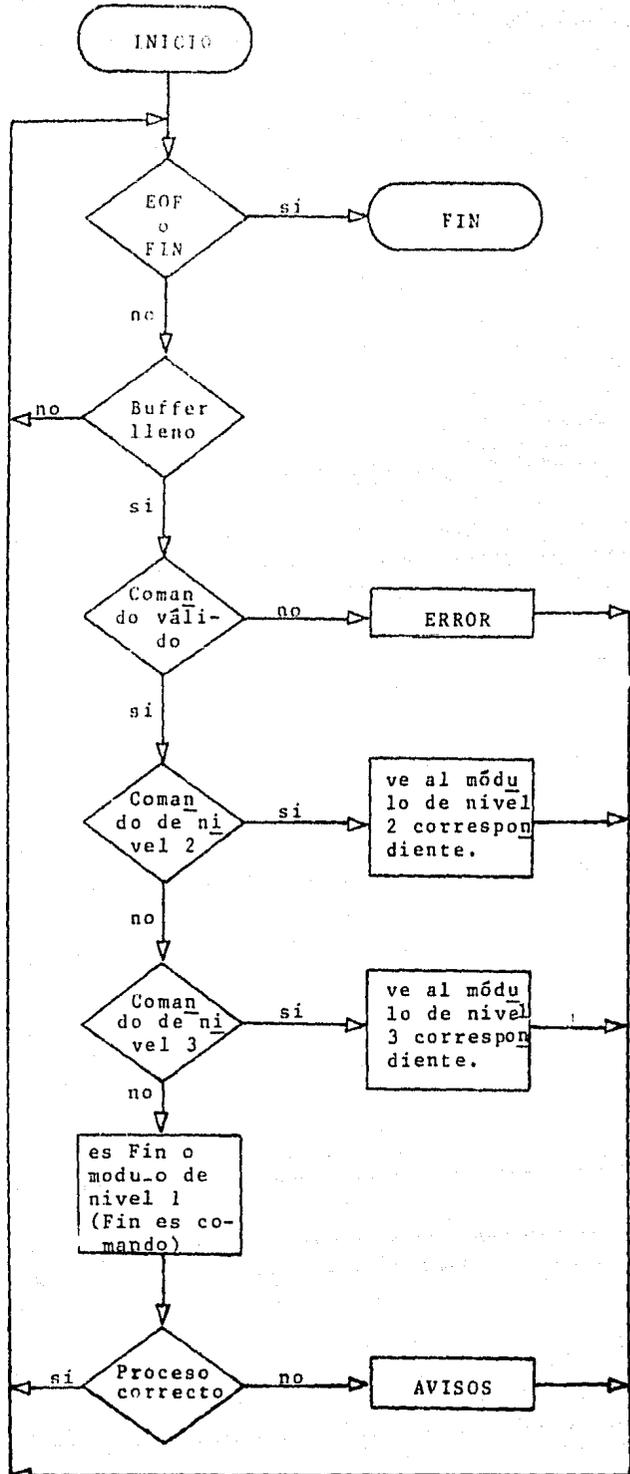
Procedure Control:

```

Begin
  Fin Falso;
  While Not eof and not Fin do
    begin
      if buffer lleno then
        if comando válido then
          if comando de nivel 2 then
            ve al módulo de nivel 2 correspondiente
          else
            if comando de nivel 3 then
              ve al módulo de nivel 3 correspondiente
            else
              if fin de proceso then
                if no hay error then
                  Fin ← cierto
                else
                  'error de tipo 1'
            else
              'comando erroneo'
          else
            % el buffer está vacío y no se hace nada
        end del While;
    End de Control;
  
```

El diagrama de flujo de la página 62 muestra el algoritmo del módulo de control en el que se observa que todo proceso se activa por un comando, que al ser identificado en la tabla de comandos del sistema, el módulo maestro transfiere el control al módulo correspondiente en donde se ejecuta la acción requerida. Todos los

Diagrama de flujo del módulo maestro



procesos son síncronos, es decir, se ejecutan uno después de otro nunca en paralelo. De aquí que para terminar un proceso e iniciar otro se regrese siempre al módulo maestro, a menos, que sea un proceso con módulos interconectados como se verá más adelante.

La manera de terminar una sesión de trabajo del sistema, es decir, desactivarlo, es por medio de comando, sin embargo si alguna falla se detecta al momento de lectura de algún archivo en la transmisión de datos, el sistema es desactivado automáticamente.

Debido a la organización del sistema el usuario puede solicitar una acción con una o más opciones, por ejemplo se puede requerir la edición de coordenadas de un programa ya sea por impresora o por pantalla, o más aún editar las coordenadas originales o las resultantes después de haber realizado movimientos o ambas. A estas opciones dentro de una acción (comando) la llamamos predicado teniéndose así dos tipos de comandos simples o sin predicado y compuestos o con predicado, los cuales deben cumplir con la sintaxis preestablecida por medio de diagramas de ferrocarrilero para cada comando, de lo contrario se produce un error en la estructura sintáctica de dicho comando.

Aun cuando la función del módulo maestro es recibir un buffer que contiene al comando y su predicado y revisar que dicho comando esté registrado en la Tabla de comandos del sistema, el análisis sintáctico del predicado se hace en el módulo correspondiente donde al detectarse alguna falla es enviada al monitor quien se encarga de hacer el diagnóstico correspondiente de aviso o error para que el usuario sea enterado de la falla.

MODULO DE ENTRADA

Para graficar un Cuerpo Geométrico en perspectiva, editar las coordenadas planimétricas y proyectivas del mismo o cualquier otra acción que involucre la presencia de datos se requiere pasar por el módulo de entrada que permite crear y/o habilitar programas de biblioteca o registrar uno nuevo en la misma.

Un programa en el sistema está definido por la expresión:

Programa = Identificador + Datos + Texto

Los elementos de la derecha de la expresión aparecen en orden de importancia de acuerdo a su función, la cual se explica a continuación.

IDENTIFICADOR

El identificador de programa es el elemento más importante debido a sus funciones. Por un lado es el nombre que el usuario asigna a un Cuerpo Geométrico, así si el programa se llama 'CUBO' se refiere a un cubo, etc. Por otro lado el sistema reconoce dicho programa con el mismo nombre lo que permite manejar un lenguaje común entre el usuario y la computadora. La importancia del identificador radica en lo que representa para el sistema.

Una vez que el usuario asigna el identificador a un programa se revisa que esté sintácticamente correcto y que no haya otro nombre igual en la biblioteca del sistema. Un identificador es sintácticamente correcto si y solo si cumple la siguiente expresión en la forma BNF:

```
IDENTIFICADOR ::= LETRA/DIAGONAL
LETRA          ::= A/B/C/.../Z
DIAGONAL      ::= /
```

Los siguientes nombres son ejemplos de identificadores correctos

M1/PROYECTO
 CUBO/VERSION/1
 EDIFICIO/MOVIL/DE/TRES/NIVELES

El objeto de que un identificador no esté repetido en la biblioteca del sistema es evitar cualquier colisión en la búsqueda de programas, ya que el usuario tiene la opción de registrar (dar de alta) cualquiera de sus programas en la biblioteca del sistema o bien usar cualquier otro programa registrado en la misma.

Si estas dos condiciones se cumplen, es decir, que el identificador del programa sea sintácticamente correcto y no exista en la biblioteca del sistema, se genera un vector de estados que funciona como el Program-Status-Word (PSW) que el sistema operativo utiliza para controlar procesos.

Este vector tiene una longitud de 16 palabras de 48 bits cada una*. Su función es informar al sistema del estado del programa presente. El contenido de este vector se muestra a continuación:

Palabra(s)	Bits	Contenido
[0] a [15]	Todos:	Se tienen 12 palabras o 72 caracteres para guardar el identificador del programa.
[12]	[1:1]	0 - indica que no hay programa 1 - indica que un programa fue creado.
	[1:1]	0 - indica que el programa no

(*) Por estar implementado en computadora Bourroughs.

Palabra(s)	Bits	Contenido
		está registrado en la biblioteca del sistema.
		1 - indica que el programa está registrado en la biblioteca del sistema.
	[2:1]	0 - indica que se cargaron los datos del programa.
	[3:1]	0 - indica que el programa no está documentado, i.e., no tiene texto.
		1 - indica que el programa está documentado.
	[4:1]	0 - si [2:1] = 1 entonces indica que el programa tiene datos y además no están registrados en la Base de Datos del sistema. De lo contrario no indica nada.
		1 - indica que el programa tiene datos, y están registrados en la Base de Datos del sistema.
	[5:1]	0 - si [3:1] = 1 entonces indica que el programa tiene texto, pero éste no ha sido registrado en la Base de datos del sistema. De lo contrario no hace nada.
		1 - indica que el programa está documentado y su texto está registrado en la Base de Da

Palabra(s)	Bits	Contenido
		tos del sistema.
[13]	[47:05]	día en que fue creado el programa
	[40:04]	mes en que fue creado el programa
	[36:08]	año en que fue creado el programa
	[27:05]	Último día que fue usado el programa
	[22:04]	Último mes que fue usado el programa
	[18:08]	último año que fue usado el programa
[14]	[47:24]	si[3:1] = 1 y [5:1] = 1 entonces apunta al primer registro donde comienza el texto. De lo contrario no vale nada.
	[23:24]	si [3:1] = 1 y [5:1] = 1 entonces apunta al último registro del texto.
[15]	[47:24]	si[2:1] = 1 y [4:1] = 1 entonces apunta al registro donde comienzan los datos.
	[23:24]	si [2:1] = 1 y [4:1] = 1 entonces indica cuantos datos tiene el programa, es decir, cuantos registros se van a leer a partir del primer registro de datos.

Como se puede observar, el vector de estados permite controlar cualquier operación que sobre un programa se haga, de ahí la

importancia del identificador.

Todo este conjunto de bits usados como banderas permiten un acceso más rápido en los archivos de la Base de Datos del sistema, además, este vector funciona como BUS de información entre los módulos, ya que estos revisan el estado del programa antes de tomar acción alguna.

Al crear o habilitar un programa se prende una bandera que le informa al sistema que se tiene un programa presente lo que indica que se tiene un vector de estados obtenido mediante el identificador, y de esta manera se podrá controlar cualquier proceso que se haga sobre el programa.

DATOS.

No obstante que los datos aparecen en segundo termino, después del identificador, su importancia es primordial ya que sin ellos no es posible graficar perspectivas de Cuerpo Geométrico, editar coordenadas planimétricas y proyectivas o ejecutar movimientos.

Para introducir datos de un programa estos deberán aparecer en forma secuencial y cumpliendo el siguiente formato:

(X/Y/P), marca

donde "()/ " son delimitadores y son opcionales, en su lugar puede usarse al menos un blanco ¶ que tiene la misma función. Las literales "X,Y,P" son números reales que representan el valor de la terna de coordenadas planimétricas del Cuerpo Geométrico, y marca es un número que toma los valores 1, 2 o 3 que funcionan como caracter de control en el módulo de graficación y cuyo significado se muestra a continuación:

marca = 1: indica que es el primer dato del programa, es decir donde se comienza la gráfica.

marca = 2: indica que hay un salto en dicho punto (dato), es decir, que la gráfica no es continua.

marca = 3: indica que es el ultimo dato del programa y fin de la gráfica.

marca = 0 o blanco

: indica que el dato pertenece a la misma gráfica, es decir, gráfica continúa.

Enviar los datos en la forma (X/Y/P) es familiar al usuario ya que es consistente con el método Red Modular. Cada vez que un dato es enviado al sistema, se produce lo siguiente:

1. Se le asigna un número en orden de aparición de 1 a n que corresponde al punto de que se trate.
2. Se aplica la transformación RM para obtener las coordenadas proyectivas (X_o/Y_o) de dicho punto.
3. Se registra el dato en dos archivos auxiliares con la misma imagen y que tienen la siguiente organización:

X	Y	P	M	X_o	Y_o	X_{oL}	Y_{oL}	X_{oLS}	Y_{oLS}	Punto
---	---	---	---	-------	-------	----------	----------	-----------	-----------	-------

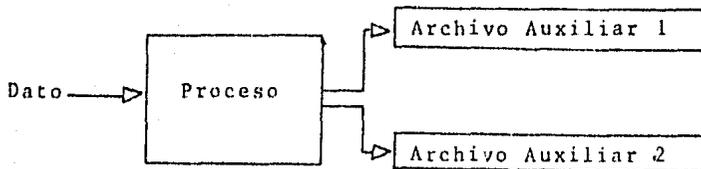
donde M = marca y Punto = número de punto.

longitud de registro = 11 palabras

X_{oL} , Y_{oL} , X_{oLS} , Y_{oLS} : Son las coordenadas límite para cuando se tienen datos que salen del plano Proyectivo de Campo Visual (PLPCV). Su valor aquí es cero, es los campos se usan en otros módulos.

Este procedimiento se puede representar con la siguiente gr̄a

fica:



Gráfica 26

De tal manera que al haber introducido los datos de un programa, el vector de estado indica que tiene datos contenidos en dos archivos auxiliares que se usan como áreas de trabajo (workspace). Ambos archivos contienen la misma información, sin embargo, su función no es la misma; el primer archivo es constante, siempre mantiene los datos originales; mientras que el segundo archivo es dinámico ya que es en él, donde se modifican los valores de las coordenadas planimétricas, proyectivas y límite.

Estas áreas de trabajo facilitan la comunicación entre módulos y la velocidad de respuesta entre procesos. Otra ventaja de trabajar en estos archivos es que nunca están presentes todos los datos en memoria principal lo que minimiza código, además de que el número de datos que pueden almacenarse en los archivos es suficientemente grande para evitar zonas de sobre flujo (overflow).

TEXTO.

Siempre que se trabaja con un proyecto en particular es conveniente documentarlo para tener una idea general de lo que se está haciendo. Aun cuando este elemento (texto) no es de primordial importancia en cuanto a la ejecución de una acción en el sistema, permite informar al usuario de lo que trata un determinado proyecto registrado bajo un identificador.

Un texto está formado por un conjunto de hasta 10 líneas que a diferencia de los datos son almacenados en una matriz que indica el número de líneas de texto que se recibieron y que corresponden al programa presente.

Hasta aquí se han visto los elementos que conforman un programa, así como la función de cada uno de ellos, lo que permite crear un programa, sin embargo, nada se ha mencionado sobre habilitación de programas en la Base de Datos del sistema (biblioteca), o del registro de estos en la misma. Para explicar estas dos acciones (habilitado y registro de programas), es conveniente describir la organización y esquema de la Base de Datos del sistema debido a que es en ésta donde se realizan las dos acciones.

BASE DE DATOS DEL SISTEMA RM.

Definimos una base de datos como un conjunto de uno o más archivos "relacionados entre sí" que contienen información similar (no idéntica) para evitar redundancias perjudiciales. Esto permite optimizar el tamaño y la organización de los archivos de la Base de Datos y disminuye la duplicidad de Información.

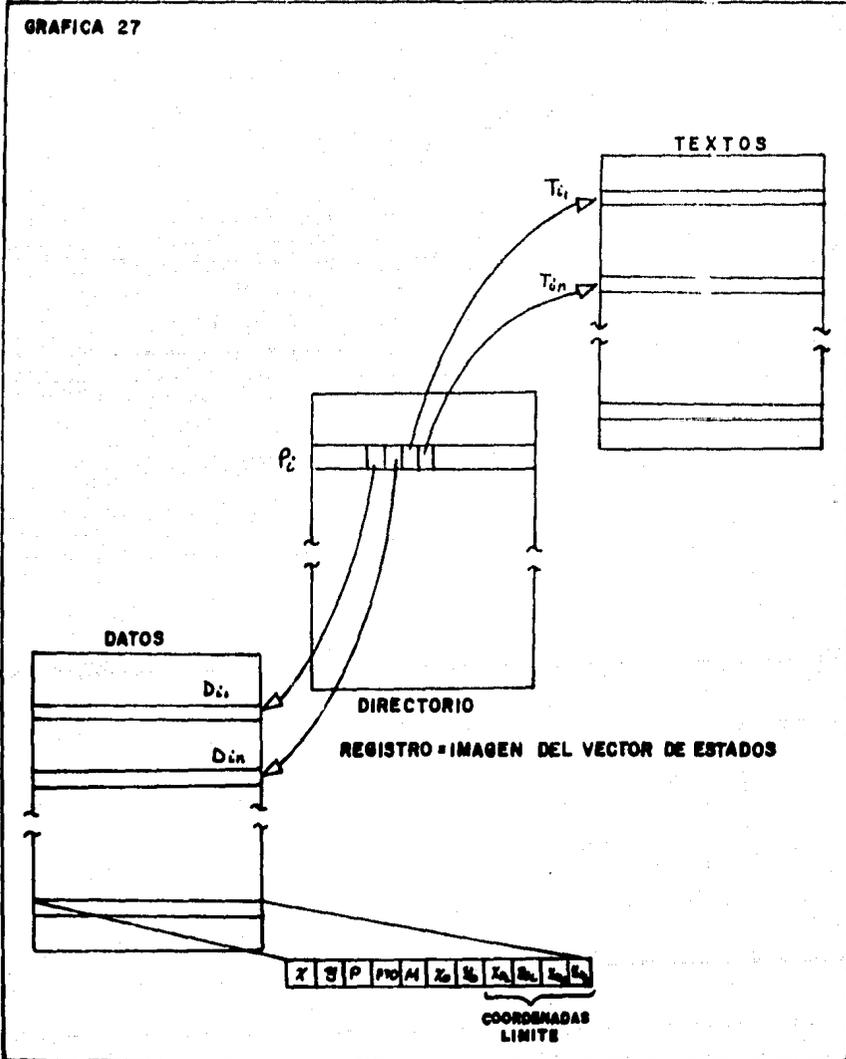
La Base de Datos (BD) del sistema consta de cuatro archivos que describimos a continuación.

Directorio: Este archivo tiene una longitud de registro de 16 palabras para ser consistente con el vector de esta dos. Su función es mantener un directorio de programas que registrados en la BD a la que llamamos biblioteca. Una vez que el identificador de Programa es registrado aquí, se revisa que los apuntadores a los demás archivos estén correctos.

Datos: Este archivo tiene una longitud de registro de once palabras, y cada registro está organizado de la misma forma que los archivos auxiliares.

Contiene los datos de todos los programas registrados

GRAFICA 27



ORGANIZACION DE LA BASE DE DATOS DEL SISTEMA

en el Directorio. Aun cuando todos los datos estan juntos nunca se tienen confusión ya que el vector de estados, registrado en el directorio, controla los apuntadores a este archivo.

Información: Este archivo tiene una longitud de registro de catorce palabras y contiene el texto de los programas registrados en el Directorio. El inicio y fin de texto de cada programa se controla por el vector de estados registrado en el Directorio.

Comandos: Este archivo tiene una longitud de registro de catorce palabras. Contiene la explicación de todos y cada uno de los comandos del sistema. Al final de la explicación de un comando se tiene un caracter de control denotado por "*" para controlar la lectura, del archivo.

Los archivos Directorio, Datos e Información están relacionados entre si de la siguiente manera:

En el Directorio se registran los vectores de entados de cada programa. Este vector contiene tanto el identificador del programa como los apuntadores a los archivos Datos e Información.* Para ilustrar esta conexión véase la gráfica 27 en la que se muestra la forma en que están organizados los archivos. En la gráfica P_i es el i -esimo programa del Directorio, $T_1 - T_n$ son los registros primero y último del texto del programa i y D_1^i apunta al registro donde inician los datos en el archivo Datos y D_n^i indica cuantos datos a partir de D_1^i contiene el programa i .

(*) El contenido del vector de estados se muestra en las páginas

Debido a esta organización decimos que la Base de Datos del sistema utiliza un esquema jerárquico representado de la siguiente manera:

Programa

- . Datos
- . Texto

La razón de usar este esquema se debe a que no se requiere de consulta de programas sino solo habilitado y alta de estos, lo que facilita el acceso y la velocidad para realizar estas dos acciones.

Las operaciones que se permiten en la Base de Datos son:

- a) Creación de la Base de Datos
- b) Alta de programas
- c) Actualización de programas.

La supresión o baja de programas no es permitida ya que generalmente es el usuario quien tiene la opción de remover (borrar) sus programas antes de darlos de alta (registrarlos) en la BD del sistema, por lo tanto será él quien decida que programa debe darse de alta.

Para que un programa se registre en la BD del sistema se deben cumplir las siguientes condiciones:

1. El programa debe tener cuando menos identificador.
2. Cualquiera de los elementos del programa (identificador, datos, texto) no estén registrados en la BD.

El objeto de tener una Base de Datos formada de programas es permitir al usuario habilitar cualquier programa existente en la BD, llamados programas de biblioteca, propios o de

otros usuarios^{*}, en sesiones subsecuentes. Esto evita duplicidad de esfuerzos, es decir, en una sesión se puede crear un programa e introducir los datos del mismo para darlo de alta en la BD, para que en sesiones posteriores (independientes del tiempo) se trabaje con el mismo programa sin pasar por el proceso de creación de programa. Nótese que una vez que el identificador está en la Directorio, los datos o texto o ambos pueden registrarse en sesiones posteriores (actualización).

Conociendo la función, contenido y organización del vector de estados, de los archivos auxiliares y de la Base de Datos del sistema, estamos en condiciones de ver los algoritmos de creación, habilitado y registro de programas que se presentan en pseudo-lenguaje.

Algoritmo para crear un programa.

Procedure Creaprograma;

```

Begin
  if (hay programa AND es de biblioteca) OR
    no hay programa then
    if identificador correcto then
      begin
        Se prende la bandera de programa a true.
        El número de programa es uno mayor que el último programa del Directorio (Prog ← Prog + 1).
        vector de estados ← clear;
        Se almacena el identificador de programa en el vector de estados y se prenden las banderas de "programa creado" a true.
        Aviso "número y nombre de programa".
      end
    else
      error "identificador inválido"
    else
      error "salvo o remuevo programa presente";
  End de Creaprograma;

```

(*) Hasta ahora no se ha requerido de mecanismos de seguridad para tener acceso solo a unos cuantos programas, por lo tanto estos no se han implementado.

Algoritmo para introducir datos.

```

Procedure Introducedatos;
begin
  if hay programa then
    if no tiene datos then
      if es por pantalla then Recibedatos (pantalla)
      else Recibedatos (archivo)
    else
      error "el programa ya tiene datos"
  else
    aviso "crear o habilitar programa";
End de Introducedatos;

```

Algoritmo para recibir datos.

```

Procedure Recibedatos (archivo);
file archivo;
begin
  Do
    if not eof then
      begin
        leedatos (archivo) (X/Y/P), marca
        aplica Transformación RM
        punto ← punto + 1;
        registra en archivo auxiliar 1 (X,Y,P, mar
          ca Xo, Yo, punto);
        registra en archivo auxiliar 2 (X,Y,P, mar
          ca Xo, Yo, punto);
      end
    until marca= 3 or eof;
end de Recibedatos;

```

Cabe mencionar que la rutina "leedatos" que se menciona arriba permite bastante flexibilidad en la introducción de datos, ya que pueden venir de la siguiente manera:

X, Y/P), marca

X /Y P, marca

(X Y P) marca

o de cualquier otra combinación ya que la rutina siempre interpreta los datos en la forma (X/Y/P) marca.

Algoritmo para dar de alta un programa en la BD.

Procedure Alta:

```

begin
  if hay programa then
    begin
      if hay datos AND no estan salvados* then
        begin
          Se toman los datos del archivo auxiliar
          1 y se copian idénticos al archivo Datos.
          Se actualizan los apuntadores correspondientes
          en el vector de estados y se prende bandera
          que indica datos registrados.
        end;
      if hay texto AND no está salvado then
        begin
          Actualiza bandera y apuntadores de texto
          en el vector de estados.
          registra el texto en el archivo Información.
        end;
      if bandera "programa registrado" apagada then
        programa registrado ← true;
      registra el vector de estados en el archivo Directorio.
    end
  else
    Aviso "no hay programa presente";
  End de Alta;

```

Nótese que el identificador (contenido en el vector de estados) se registra al último ya que los apuntadores a los archivos Datos e Información se actualizan en el vector de estados durante el proceso de Alta de Datos y Texto, de lo contrario, si se registrara al principio, se perdería información ya que los apuntadores no serían correctos.

(*) Salvados indica registrados en el Archivo Datos.

Algoritmo para habilitar programas.

```

Procedure Habilita;
begin
  if programa en biblioteca then
  begin
    if tiene datos then
    begin
      Lee los datos del programa del archivo
      Datos y las copias en los archivos au-
      xiliares 1 y 2.
      Aviso "datos presentes".
    end
    else
      aviso "el programa no tiene datos"
    end
  else
    error "el programa solicitado no existe";
  end de Habilita;

```

En el proceso de Habilitado de programas, los archivos auxiliares 1 y 2 son nuevamente usados, ya que como se recordará se utilizan como áreas de trabajo en otros módulos.

Para terminar la exposición de este módulo restaría ver la función del archivo COMANDOS, sin embargo ésta se analizará en el módulo de información, aquí solo diremos que está relacionado en forma directa con el sistema y disociado de los otros archivos de la Base de Datos del sistema.

Hemos visto, entonces, que el habilitado, la creación y el registro de programas son las tres funciones del módulo de entrada y se han descrito los algoritmos empleados para ejecutar dichas acciones.

MODULO DE INFORMACION Y UTILERIAS

Hay procesos en los que no se requiere la presencia de programas para ejecutarse; buscar un programa de biblioteca; saber que programa está activo o variar la Amplitud de Campo Visual son algunos ejemplos de éstos. El módulo de información y utilerías es el encargado de realizar estas tareas.

El no depender de la presencia de un programa lo convierte en un módulo de nivel 2 (jerárquicamente hablando) cuya función principal es brindarle al usuario una herramienta para una mejor operación del sistema con objeto de optimizar los resultados esperados por el usuario.

El sistema está habilitado para proporcionar la siguiente información al usuario:

- a) Buscar un determinado programa en la biblioteca del sistema*.
- b) Mostrar en pantalla todos los programas registrados en la biblioteca del sistema.
- c) Preguntar cual es el programa que está activo.
- d) Conocer el valor de la Amplitud de Campo Visual (ACV) y de la distancia focal (d) con que se trabaja.
- e) Explicar cada comando del sistema.

Las utilerías que se tienen son las siguientes:

- I. Variar el valor de la Amplitud de Campo Visual, y por tanto de la distancia focal (d).
- II. Habilitar la terminal graficadora y el graficador o plotter.
- III. Remover, si es que existe, el programa presente o activo.

(*) Se usa el término de biblioteca aunque se infiere que es la Base de Datos del sistema.

A continuación se describe como se ejecuta cada uno de estos procesos.

- a) Para la búsqueda de un programa se debe enviar el identificador o el número del programa deseado, en el primer caso la búsqueda es secuencial ya que los programas no son ordenados en orden alfabético; en el segundo caso la búsqueda se hace de manera directa, ya que el número de programa corresponde también al número de registro en el archivo Directorio. Si el programa solicitado existe en la biblioteca del sistema se indica el número e identificador del mismo, de lo contrario se informa que no existe.

- b) Para saber qué programas hay en biblioteca se ejecutan los siguientes pasos.

P1. Se lee del directorio, secuencialmente, cada identificador registrado en biblioteca, es decir, se habilita el vector de estados del programa en cuestión.

P2. Del vector de estados se despliega en pantalla el número e identificador de programa así como la fecha en que éste ingreso a la BD.

P3. Si la bandera de texto, en el vector de estados, está encendida entonces se lee del archivo Información las líneas del texto del programa mismas que se despliegan en pantalla a continuación de lo escrito en P2; es aquí donde tiene utilidad el texto ya que dice de que trata el programa desplegado. En caso de que el programa no esté documentado se despliega en pantalla el aviso que así lo indica.

El objeto de mostrar los programas de biblioteca o catálogo es que el usuario conozca lo que hay y si le interesa alguno tiene la posibilidad de habilitarlo.

Tanto en la búsqueda como en la muestra del catálogo de

programas no afectan el programa que esté activo en caso de haberlo.

- c) Es frecuente que durante una sesión de trabajo con el sistema se olvide que programa está presente o activo, para ello se implementó este proceso que permite conocer el número identificador del programa activo el cual se despliega en pantalla además de indicar la cantidad de datos del mismo indicando si estos están registrados en la Base de Datos del sistema. Toda la información que se despliega en pantalla se obtiene del vector de estados, y en caso de no haber programa activo se le informa al usuario.
- d) En ocasiones se obtienen perspectivas de Cuerpos Geométricos que no corresponden a los esperados, esto se debe a que la Amplitud de Campo Visual sufrió algún cambio: Aunque generalmente se trabaja con la ACV de omisión que es de 60 grados su valor puede cambiarse, por tal razón se implementó este proceso en el que se despliega tanto la ACV como la distancia focal (d) con la que se está trabajando.
- e) El sistema está habilitado para autoexplicarse a fin de que el usuario conozca como opera y los comandos que tiene así como la forma como debe manejarse cada uno de ellos. Es aquí donde se tiene la relación entre el sistema y el archivo Comandos.

En el proceso donde se ejecuta la explicación se tiene una tabla de direcciones al archivo Comandos y cada una corresponde al registro donde comienza la explicación de ese comando. A su vez el archivo Comandos tiene una marca de fin de comando denotada por "*". El sistema puede explicarse globalmente o por comando de acuerdo a como lo estime el usuario⁴. La forma de desactivar este proceso es mediante

(*) Para una exposición mas detallada de este proceso consúltese el comando explica en el manual del sistema.

otro comando.

El algoritmo que se emplea en este proceso se muestra a continuación:

Procedure Explica:

```

begin
  while not eof and not fin
  do
    if comando valido then
      if not fin then
        begin
          J ← comando;
          explicacomando (dirección [ j ]);
        end
      else
        fin ← true;
      else
        mensaje "dame comando";
      end de Explica;
    end
  end

```

Los procesos de utilería permiten ejecutar las siguientes acciones:

1. Variar la Amplitud de Campo Visual (ACV) y por ende la distancia focal (d), para ello se requiere de un comando cuyo predicado es un número real que indica los grados de la ACV deseada. Usando diagramas de ferrocarrilero lo representamos de la siguiente manera:



con el nuevo valor de ACV se calcula la distancia focal (d) con la expresión:

$$d = 5 \operatorname{angtan} (ACV) \text{ m}$$

Una vez variados la ACV y d se mantienen constantes hasta que sus valores sean cambiados nuevamente mediante este proceso.

II. La habilitación del graficador y la pantalla graficadora dependen de la decisión del usuario y del tipo de terminal con que se esté trabajando. Cuando se desea habilitar las salidas gráficas se pregunta el tipo de terminal en la que se está trabajando y se revisa una tabla en la que se definen las terminales gráficas válidas, si el tipo enviado está en la tabla entonces se habilita las salidas gráficas de lo contrario no se hace.

III. En caso de que se desee remover el programa activo o presente se producen las siguientes acciones:

- . El vector de estados es puesto en ceros, es decir, vacío o nulo.
- . Los archivos auxiliares 1 y 2 son destruidos
- . El texto es borrado
- . La bandera de programa activo es apagada.

En caso de no haber programa presente no se ejecutan las acciones anteriores.

Con la información que este módulo proporciona, el diálogo hombre-máquina se hace más próximo entre el usuario y la computadora ya que los procesos de este módulo no interfieren con los procesos que se ejecutan en otros módulos.

Con las utilerías disponibles se permite mayor versatilidad en el diseño como es el caso de la variación de la Amplitud de Campo Visual (ACV) que equivale a usar diferentes lentes (normal, gran angular, medio gran angular, etc.) en la toma de una misma fotografía, que en el sistema equivale a diferentes resultados perspectivos de un mismo Cuerpo Geométrico.

MODULO DE GRAFICACION

Haciendo un análisis retrospectivo hacia el objetivo principal para el cual fué diseñado el sistema encontramos que la representación gráfica de perspectivas por computadora sigue siendo la piedra angular de este trabajo, ya que son las gráficas la herramienta que el sistema proporciona al usuario para que él como diseñador, analice con detalle sus proyectos.

Aun cuando las pantallas graficadoras y los graficadores son dispositivos de salida. Se requiere de un software especial para controlar a estos dispositivos, tarea realizada por el módulo de graficación.

ELEMENTOS DE GRAFICAS POR COMPUTADORA (Computer-Graphics).

Las gráficas por computadora involucran la representación de objetos gráficos obtenidos de información no gráfica, es decir, de datos almacenados en un archivo (generalmente disco).

Las dos condiciones que deben cumplirse para la construcción de gráficas por computadora son:

- . Tener un conjunto de pares coordenados (x,y) llamados puntos.
- . Definir un sistema coordenado de referencia para localizar tales puntos.

En nuestro caso la primera condición se satisface si y solo si se tiene un programa activo con datos, esto es, con coordenadas proyectivas (X_o/Y_o) , la segunda condición se satisface ya que estos pares coordenados se localizan en el Plano Proyectivo de Campo Visual (PLPCV). Una tercera condición que se tiene que cumplir en el sistema es que las salidas gráficas estén habilitadas.

En los dispositivos de salidas gráficas es necesario establecer un marco de referencia para mejorar la representación de las gráficas, estas dos referencias son:

Ventana: Es cualquier dominio rectangular arbitrario, localizado en la pantalla graficadora, es decir, es un subdominio de la pantalla.

Puerto de vista

(viewport): Es el área rectangular de la pantalla en la que la ventana puede mapearse.

El puerto de vista está limitado por el tamaño físico de la pantalla de tal manera que cualquier punto que sobre pase esta área visible, no es mapeado. La ventana tiene por objeto definir el área de la pantalla donde se va a mapear un punto, esto permite el escalamiento de éste para que sea mapeado en la ventana.

En el sistema, la ventana está definida por el Plano Proyectivo de Campo Visual (PLPCV) el cual es un cuadrado de diez módulos de magnitud por lado y localizado en el centro de la pantalla.

Desde el punto de vista de gráficas podemos definir al Cuerpo Geométrico como un conjunto de nodos conexos no disjuntos, es decir, nodos o vértices unidos por aristas que pertenecen a un mismo cuerpo. A este conjunto de nodos y aristas se les llama primitivos gráficos. En graficación por computadora estos primitivos están representados por la operación de mover el cursor o pluma en un dispositivo de graficación y son:

mover el cursor de un punto a otro punto sin pintar la trayectoria del movimiento.

mover el cursor de un punto a otro pintando la trayectoria del movimiento mediante un vector brillante.

Como se observa estos dos primitivos de graficación por computadora equivalen a los primitivos gráficos del Cuerpo Geométrico (nodos y aristas).

Nótese que trazar un punto equivale a mover el cursor al mismo punto pintando la trayectoria del movimiento, es decir, una recta de longitud cero.

Se tiene una variante en el movimiento del cursor cuando las aristas del Cuerpo Geométrico son descritas por un conjunto finito de puntos (mas de dos), en este caso el cursor se desplaza por fracciones incrementales para unir los puntos de los aristas, obteniéndose gráficas con mas detalle y definición, es decir, con mayor resolución. Definimos resolución a la capacidad que tiene un dispositivo para imagenes de representar las gráficas que realiza lo mas cercano a la realidad, esta proximidad no solo depende del dispositivo empleado sino también del número de puntos de las aristas del Cuerpo Geométrico u objeto gráfico. Nótese que con este tipo de movimiento es factible graficar curvas.

El movimiento del cursor se puede ejecutar de dos maneras:

- a) Absoluta: El movimiento del cursor a cualquier punto de la pantalla se hace tomando como referencia a otro punto de la pantalla ya sea definido por el usuario o empleando el de omisión "HOME"*.
- b) Relativo: El movimiento del cursor a cualquier punto de la pantalla se hace tomando como referencia el punto anterior, el cual a su vez, sirve de referencia al siguiente punto y asi sucesivamente.

Los primitivos del módulo de graficación utilizan movimien-

(*) Home es la posición inicial del cursor, localizado en la esquina superior izquierda de la pantalla.

to absoluto y toman como referencia el centro de la pantalla, que a su vez coincide con el centro de la ventana, que por ser la misma que el Plano Proyectivo de Campo Visual (PLPCV) los movimientos del cursor se hacen con respecto al punto de fuga único "PF" de Red Modular.

Para mover el cursor ya sea de manera absoluta o relativa hacia cualquier punto de la pantalla se requieren sus coordenadas (x,y) de pantalla, las cuales se especifican con un conjunto de caracteres ASCII.

Una pantalla graficadora de desplegado o display de tubo de rayos catodicos (CRT) Tektronix modelos 4006, 4010 o 4012, y un graficador o plotter digital modelo 4662 son los dispositivos de graficación que éste módulo utiliza.

Este tipo de dispositivos trabajan en dos modos:

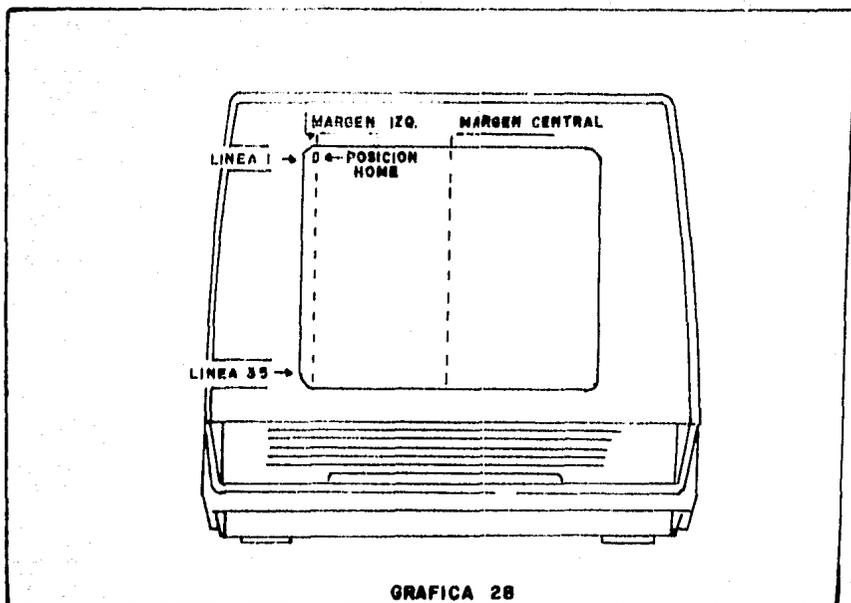
- . modo alfanumérico
- . modo gráfico

MODO ALFANUMERICO

En este modo la pantalla funciona como cualquier terminal remota con la diferencia de ser una pantalla de persistencia y no de refresco por lo que no se puede circular, es decir, mover las líneas escritas en ella, lo que hace necesario limpiarla totalmente cuando se llena para seguir enviando nuevas líneas.

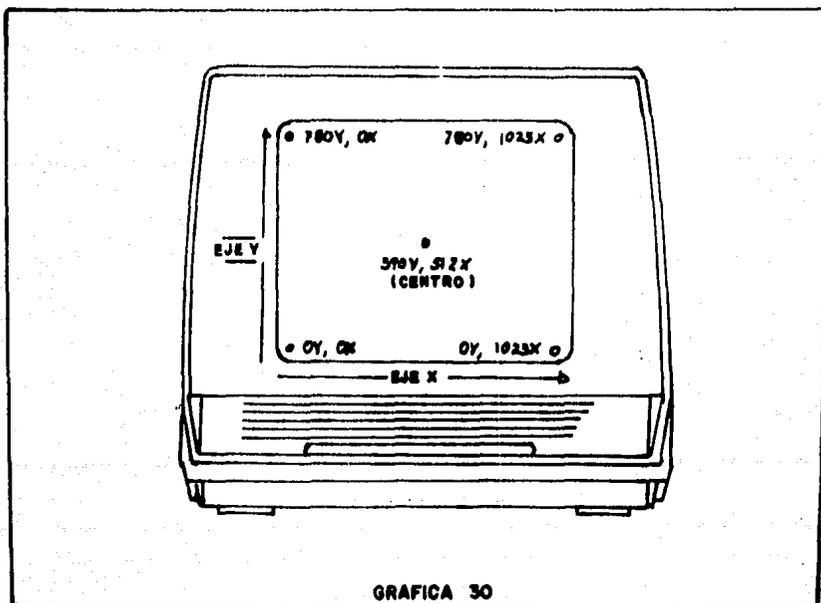
La pantalla esta dividida en dos columnas; la primera comienza en el extremo izquierdo de la pantalla (home), mientras que la segunda empieza en la mitad de ésta como se indica en la gráfica 28.

Las líneas se escriben siempre desde el inicio de cada margen sin exceder un total de 35 por columna. Para editar mas líneas se tiene que limpiar la pantalla mediante la tecla PAGE ya



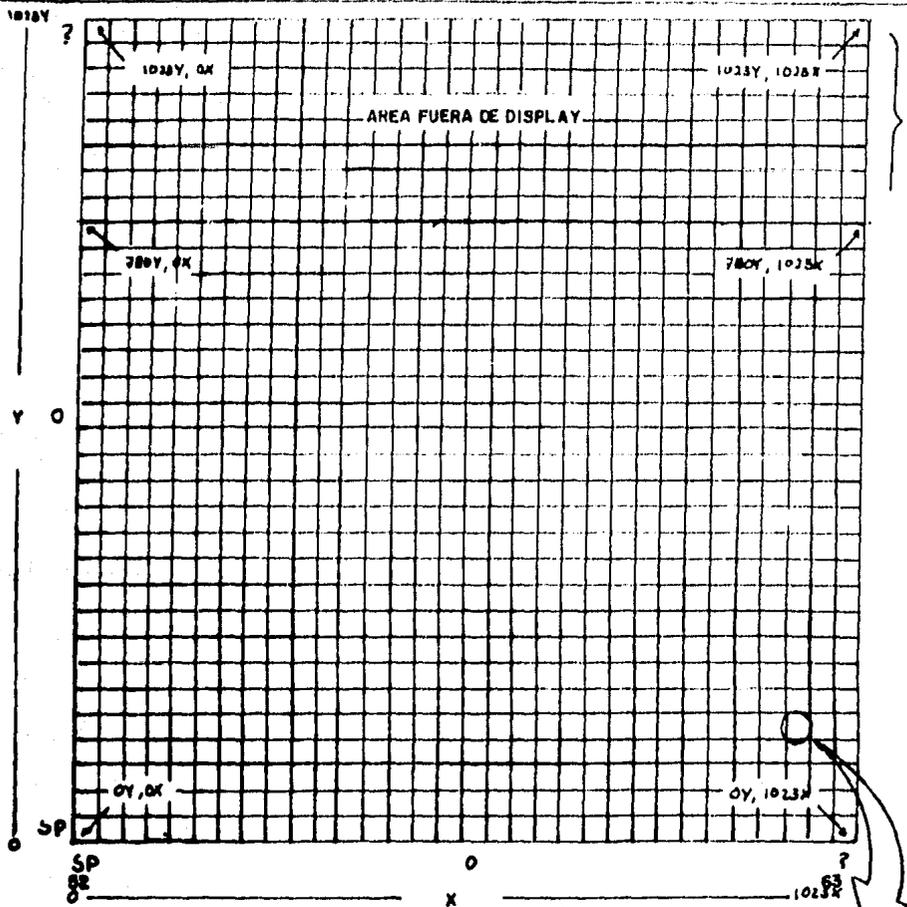
GRAFICA 28

DESCRIPCION DE LA PANTALLA GRAFICADORA



GRAFICA 30

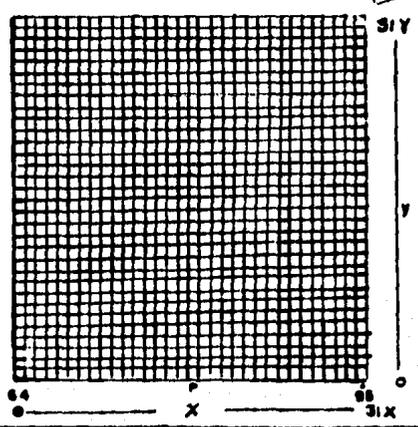
RANGO DE COORDENADAS PANTALLA QUE MANEJA LA TERMINAL



GRAFICA 29.a

REPRESENTACION DE COORDENADAS
PANTALLA EN LA TERMINAL GRAFICA

GRAFICA 29.b



que no es posible circular las líneas escritas en la pantalla.

En modo alfanumérico es conveniente trabajar solo en la primera columna ya que en la segunda, por estar en la mitad de la pantalla, se escribe sobre parte de las líneas de la primera columna.

MODC GRAFICO.

En este modo la pantalla funciona como graficadora y acepta solamente caracteres de control en código ASCII. Igual que en el modo alfanumérico, en el modo gráfico los caracteres pueden enviarse desde el teclado de la terminal cuando está en LOCAL o ECHO o desde una computadora central (Host) cuando está en línea.

En nuestro caso la pantalla trabaja en línea conectada a una computadora Burroughs de las series 6000 o 7000. Estas computadoras trabajan en código EBCDIC, pero para controlar la pantalla graficadora se requiere código ASCII, esta transformación de códigos la realiza el DCP* y por tanto no hay que hacer la transformación de códigos a través de programas. Los primitivos que permiten este tipo de pantallas son los mismos que todo dispositivo de graficación:

- . mover el cursor de un punto a otro de la pantalla sin pintar la trayectoria del movimiento.
- . mover el cursor de un punto a otro punto de la pantalla trazando la trayectoria mediante un vector brillante.

La pantalla está dividida en una malla de 1024 puntos horizontales (eje x) y 1024 puntos verticales (eje y) para localizar un punto en ella. Para dar coordenadas a los diferentes puntos

(*) DCP Data control Processor de computadoras Burroughs.

de la pantalla, estos se agrupan en dos matrices de 32 x 32 puntos* cada una como se muestra en la gráfica 29, de esta manera cada punto de la pantalla tendrá asociadas cuatro coordenadas. Dos coordenadas altas (X,Y) para determinar el elemento de la matriz grande que contiene al punto (gráfica 29a) y dos coordenadas bajas (x,y) para determinar el elemento de la matriz pequeña que representa al punto (gráfica 29b). La secuencia para mandar las coordenadas de un punto a la terminal es: Y,y,X,x. Al trabajar en línea se requiere que se envíen los caracteres ASCII correspondientes a un punto desde la computadora central (Host).

El puerto de vista o viewport, como se había mencionado, está limitado por el tamaño físico de la pantalla, se reduce el área visible o campo visual a 1024 puntos en el eje X y 780 puntos en el eje Y como se muestra en la gráfica 30. Entonces se tienen los siguientes rangos para representar un punto:

$$0 \leq X \leq 1023$$

$$0 \leq Y \leq 779$$

En los ejemplos se mostrará que esta es una resolución aceptable para graficar un Cuerpo Geométrico.

El módulo de graficación siempre hace funcionar a la pantalla en modo gráfico para efectuar el trazo perspectivo del Cuerpo Geométrico. En general, las coordenadas de un punto de una gráfica son números reales y para que queden dentro del puerto de vista de la pantalla es necesario:

- . usar factor de escala para hacer que cualquier punto con valores reales quede dentro del puerto de vista.
- . Hacer una transformación de este valor escalado de tipo entero a su correspondiente carácter ASCII, es decir, una transformación de coordenadas enteras a coordenadas pantalla.

(*) De lo contrario se necesitaría 1024 caracteres para el eje X y 1024 caracteres para el eje Y y no existe código que los contenga.

La ventaja de usar este par de transformaciones es que permiten trabajar en centímetros (cm) y no en coordenadas pantalla.

El puerto de vista mide 17.5 cm en el eje X por 12 cm en el eje Y y el número de puntos que hay en 1 cm es 48. Como en Red Modular se trabaja con módulos que toman cualquier valor de unidades se tomará un centímetro por módulo y entonces la ventana medirá diez cm por lado y su centro será el de la pantalla (coordenadas 390y, 512x).

Ya que la ventana es idéntica al Plano Proyectivo de Campo Visual (PLPCV), los valores que puede tomar un punto para que sea mapeado en la ventana, están determinados por las condiciones de inclusión de la Transformada RM, es decir, deben estar en el intervalo cerrado $[-5,5]$.

Conociendo los valores máximos y mínimos de la ventana y el tamaño en cm de la pantalla se calcula el factor de escala con la siguiente expresión:

$$\text{FACTOR} = \frac{12(48)}{X_{\text{max}} - X_{\text{min}}} = \frac{12(48)}{10} = 60$$

donde 48 es el número de puntos que hay en un cm. Se tomó solo el valor en cm del eje Y por ser mas pequeño ya que se utilizan esca las iguales en ambos ejes por trabajar con una ventana cuadrada.

Operación del módulo de graficación.

Las tres funciones básicas que el módulo de graficación realiza se presentan a continuación en el orden que se ejecutan:

1. Revisa que la pantalla graficadora o el plotter o ambos estén habilitados. Para ello se requiere que la bandera que indica tal estado esté prendida. Aun cuando la pantalla graficadora puede habilitarse desde el inicio de la sesión de

Trabajo con el sistema, ésta se puede deshabilitar mediante un comando del módulo de utilería, por tal razón es conveniente conocer el estado de la pantalla graficadora cada vez que trabaja este módulo para obtener un resultado satisfactorio y evitar fallas de transmisión.

2. Si la condición 1 se satisface, entonces se requiere que haya un programa activo con datos. Esta tarea consiste en revisar el vector de estados de dicho programa, que al tener la bandera de datos prendida implica que se tienen los archivos auxiliares 1 y 2 presentes. El archivo auxiliar 2 contiene los datos a graficar, por lo tanto si éste no existe no es posible graficar objeto alguno. En este proceso se anexa otro archivo auxiliar que contiene los mensajes a editar en la gráfica, el cual al igual que el archivo auxiliar 2, es cerrado y habilitado para iniciar el proceso de graficación.

3. Graficación de los datos.

Antes de iniciar la gráfica la pantalla se cambia de modo alfanumérico a modo gráfico enviando el carácter ASCII de control GS o 29 en decimal, y el cursor se posiciona al centro de la pantalla coordenadas ASCII (390y, 512x) o (60x, 147y) en EBCDIC. A partir de este momento la pantalla está lista para iniciar la gráfica.

Cada registro del archivo auxiliar 2 es leído en forma secuencial y puesto en un vector (arreglo) que tiene la misma imagen que los registros, como se indica en la gráfica 31. Para mapear cada par coordenado (X,Y) primero se aplica el factor de escala mediante las siguientes expresiones:

$$X = \text{RAUND} ((X - X_{\text{minima}}) * \text{FACTOR}) + X_{\text{inicial}}$$

$$Y = \text{RAUND} ((Y - Y_{\text{minima}}) * \text{FACTOR}) + Y_{\text{inicial}}$$

donde

- . Xminima y Yminima son los valores mínimos de X y Y (-5,-5) respectivamente.
- . FACTOR es el factor de escala calculado anteriormente cuyo valor es 60.
- . Xinicial y Yinicial indican la posición del origen para el movimiento absoluto, que en este caso es el centro de la pantalla

Una vez transformados los valores de X y Y se busca su correspondencia en coordenadas pantalla, es decir, en caracteres EBCDIC que al ser enviados a través del DCP de máquinas Burroughs se transforman en caracteres ASCII que permite mover el cursor al punto deseado. Para pasar de coordenadas enteras a coordenadas pantalla se sigue el siguiente algoritmo:

$$\left. \begin{array}{l} \text{XHIGH} = \text{DIV} (X, 32) + 32 \\ \text{YHIGH} = \text{DIV} (Y, 32) + 32 \end{array} \right\} \text{Coordenadas altas}$$

$$\left. \begin{array}{l} \text{XLOW} = \text{MOD} (X, 32) + 64 \\ \text{YLOW} = \text{MOD} (Y, 32) + 96 \end{array} \right\} \text{Coordenadas bajas}$$

donde 64 corresponde al caracter SP y 96 corresponde al caracter "-" o coordenadas pantalla (0,0). Aplicando este algoritmo se tiene la secuencia Y,y,X,x o coordenadas pantalla correspondientes. Este valor se almacena como un caracter, usando los valores XHIGH, YHIGH, XLOW, YLOW como son índices que apuntan a una tabla de equivalencias decimales de caracteres EBCDIC obteniéndose el carácter de control correspondiente, mismo que se almacena en una variable temporal para ser enviado posteriormente a la pantalla.

Para saber que primitivo seleccionar se revisa el valor de la marca:

Si marca = 1 o marca = 2

El cursor se mueve al punto indicado sin trazar la trayectoria del movimiento.

Si marca = blanco o marca = 0

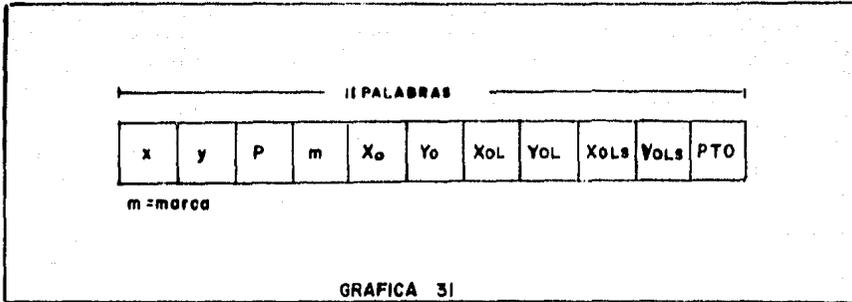
El cursor se mueve del punto donde estaba anteriormente al punto actual pintando la trayectoria con un vector brillante.

Si marca = 3

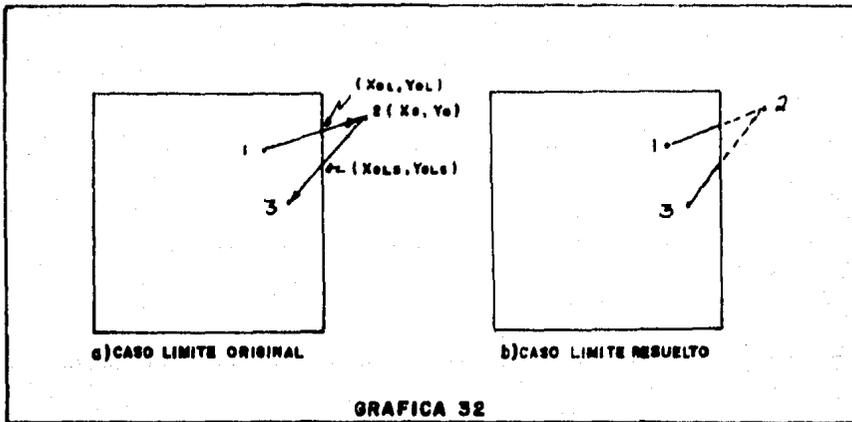
Tiene el mismo efecto que el caso anterior, solo que una vez efectuado el movimiento se termina el proceso de graficación.

El par coordenado (X, Y) a graficar del registro leído puede ser cualquiera de los siguientes (X_0/Y_0) , (X_{0L}/Y_{0L}) , (X_{0LS}/Y_{0LS}) dependiendo del caso de que se trate (ver gráfica 31).

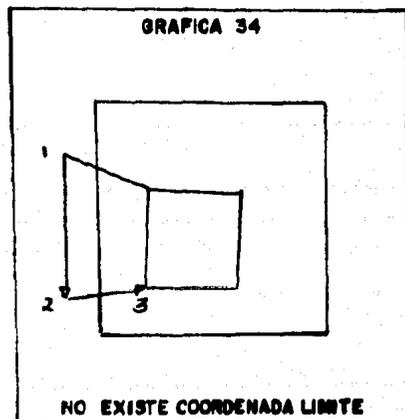
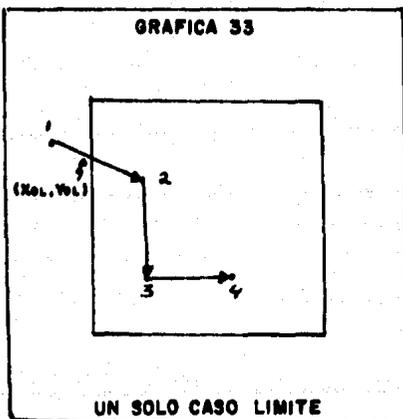
Puesto que la ventana está determinada por el Plano Proyectivo de Campo Visual (PLPCV) cualquier par de coordenadas proyectivas (X_0/Y_0) debe cumplir las condiciones de inclusión, es decir estar dentro de el intervalo cerrado $[-5, 5]$, de lo contrario están fuera del Campo Visual y no es posible graficarlas. Cuando estas condiciones no se cumplen entonces se usan los pares coordenados (X_{0L}/Y_{0L}) y/o (X_{0LS}/Y_{0LS}) que representan las coordenadas proyectivas límite para primera y segunda intersección. La forma en que se obtuvieron se vera en el módulo de movimientos. Los casos de intersección con el Campo Visual o clipping que se resuelven con estas coordenadas son cuando se tienen:



CONTENIDO DEL REGISTRO AUXILIAR DOS



SOLUCION DE CASOS LIMITE (CLIPPING)
PARA EL TRAZO CORRECTO DE LA GRAFICA



- a) Dos intersecciones
- b) Una intersección.

Los cuales se describen a continuación.

Caso a).

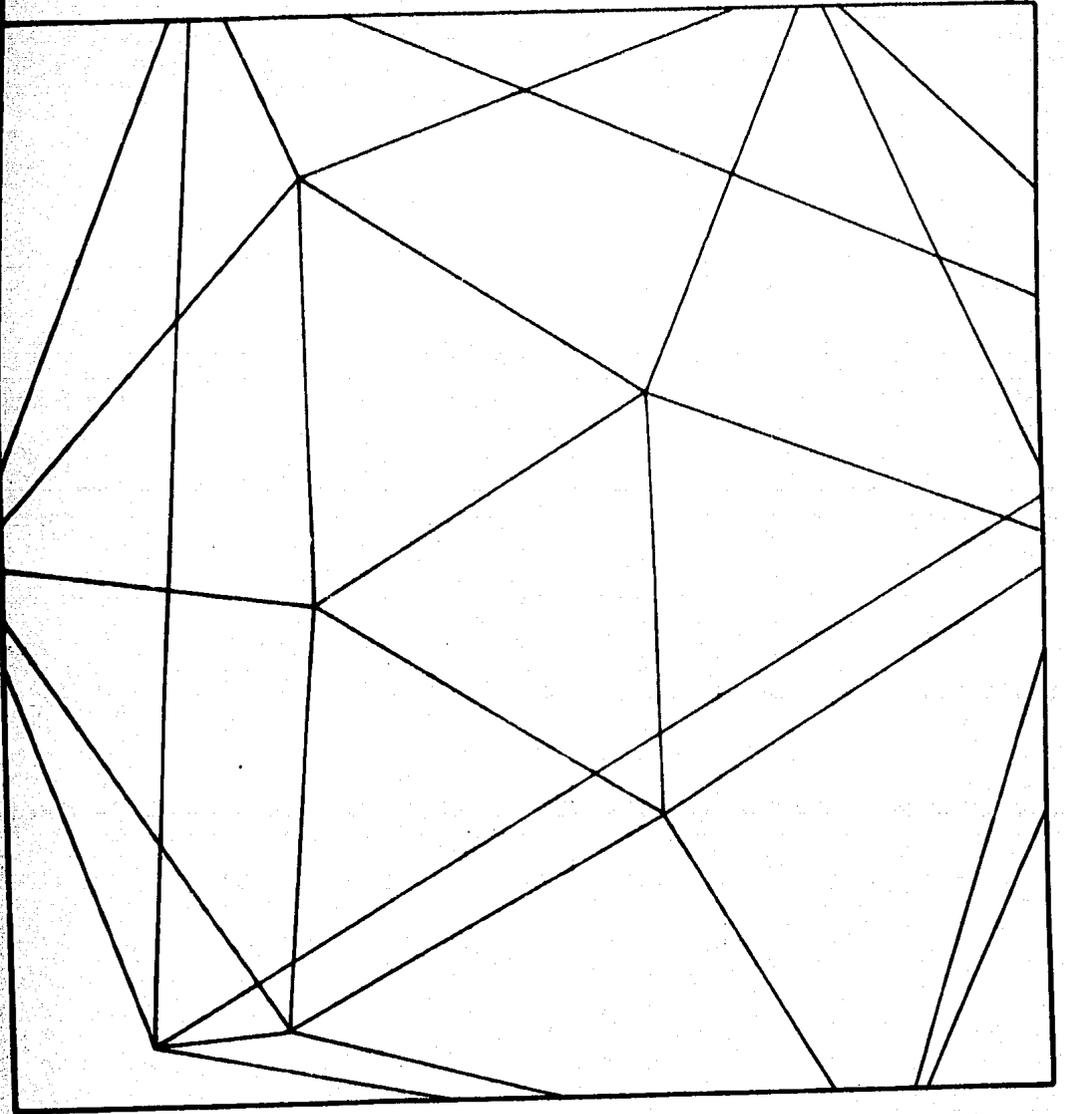
Obsérvese la gráfica 32. a donde se desea mover el cursor del punto 1 al punto 2 denotado por P1-P2. Ya que las coordenadas proyectivas de P2 no cumplen las condiciones de inclusión, se tiene un caso de clipping, por tanto se usan las coordenadas límite (Xo_L/Yo_L) o de primera intersección con la ventana (PLPCV), modificándose el movimiento de P1 a la intersección (Xo_L/Yo_L) como se indica en la gráfica 32.a.

Dado que el siguiente movimiento del cursor se hará hacia P3 (P2-P3), habrá que posicionar el cursor en la segunda intersección con el PLPCV, es decir, coordenadas (Xo_{LS}/Yo_{LS}) o de segunda intersección, el movimiento a este par coordenado se hará sin pintar la trayectoria, esto es, ejecución del primer primitivo. De esta manera al leerse el siguiente dato P3 que está dentro del campo visual se trazará la trayectoria P2-P3 a partir de la segunda intersección de P2. Resultando la gráfica 32.b.

Caso b).

A diferencia del caso anterior solo se tiene una intersección como se indica en la gráfica 33 y ésta ocurre en la trayectoria P1-P2. Aquí se dirige el cursor del punto (Xo_L/Yo_L) de P1 hacia el punto P2 coordenadas (Xo/Yo). Por haber solo una intersección el valor de (Xo_{LS}/Yo_{LS}) de P1 vale cero lo que

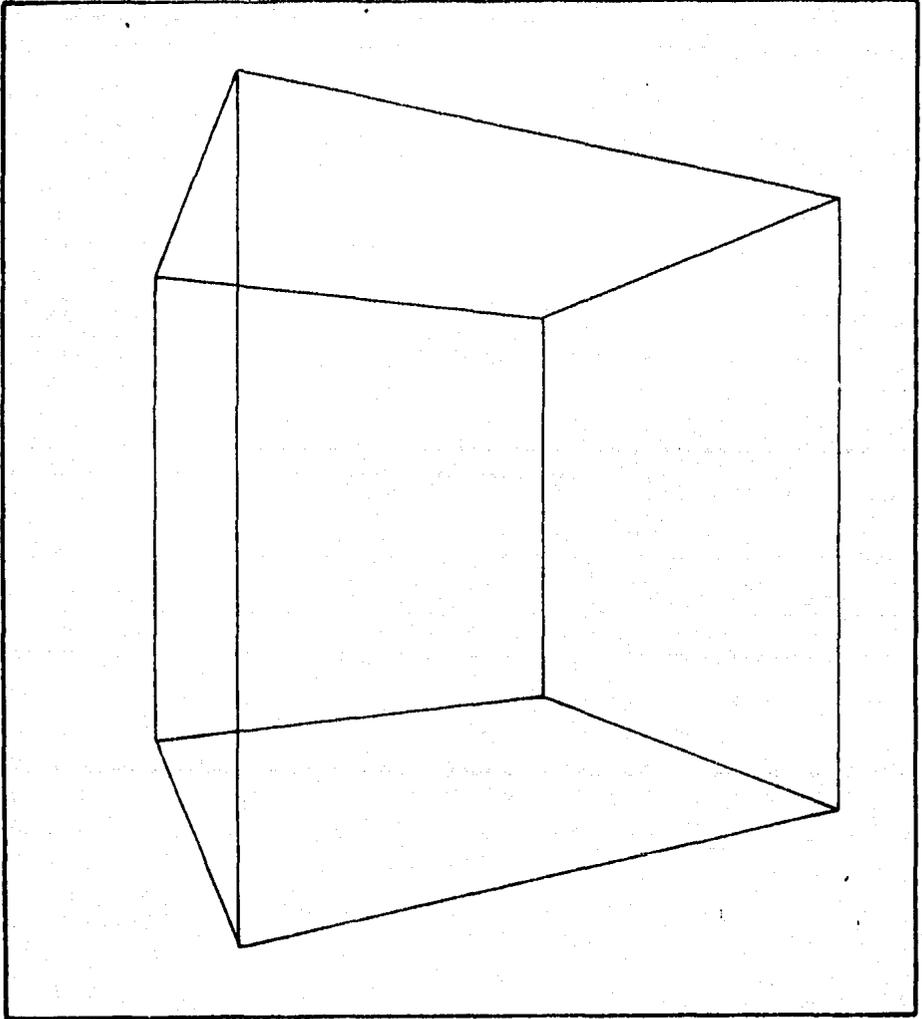
ICOSAEDRO



GRAFICA 34 b

CASO DE CLIPPING RESUELTO POR EL SISTEMA PARA UN ICOSAEDRO

IDENTIFICADOR



GRAFICA 35

EO: μ ; < MOVIMENTO > ; < COORDENADAS > ; < ACV >

indica que no existe una segunda intersección.

El que un punto esté fuera de campo visual no implica que tenga necesariamente dos o una intersección como es el caso de la gráfica 34 en donde P1 y P2 están fuera de la ventana, por lo tanto la trayectoria P1-P2 no tiene intersección con el PLPCV, por lo que en lugar de trazar la trayectoria el cursor es posicionado en la primera coordenada límite de P2 para la trayectoria P2-P3 indicado por X en la gráfica 34.

Si cada par coordenado (X,Y) una vez transformado en coordenadas pantalla, se transmite a la pantalla para que el cursor se mueva a dicho punto (trazando o no la trayectoria del movimiento), además de que los movimientos del cursor serían demasiado bruscos, se consumiría mayor tiempo de proceso por la cantidad de entrada y salida que se realiza. Para evitar este efecto se utiliza un arreglo de 256 palabras que funciona como buffer en el cual son almacenadas las coordenadas pantalla junto con su primitivo y una vez que el buffer está lleno o la gráfica está terminada, es decir, tiene menos de 256 caracteres, lo que suceda primero, se envía el buffer con los 256 caracteres a la pantalla. Esto además de disminuir el tiempo de graficación por realizar menos entrada y salida hace que los movimientos del cursor sean suaves. Una vez que finaliza la gráfica (marca =3), se procede a trazar la ventana y el punto de fuga único de Red Modular, es decir, un punto al centro de la pantalla.

Para completar la gráfica se editan los siguientes datos: identificador de programa; estación de observación (EO); Movimiento realizado; y amplitud de campo visual (ACV), con el formato que se muestra en la gráfica 35.

La gráfica resultante es la perspectiva del Cuerpo Geométrico vista desde una determinada estación de observación (EO) y con una amplitud de campo visual en particular.

Debido a que la terminal gráfica empleada es una pantalla de persistencia habrá que limpiarla cada vez que se trace una perspectiva del Cuerpo Geométrico de interés, esto hará que cada gráfica que se realice parecerá independiente de la anterior aun cuando sea el resultado de haber movido al Cuerpo Geométrico.

A continuación describimos los algoritmos empleados en el módulo de graficación.

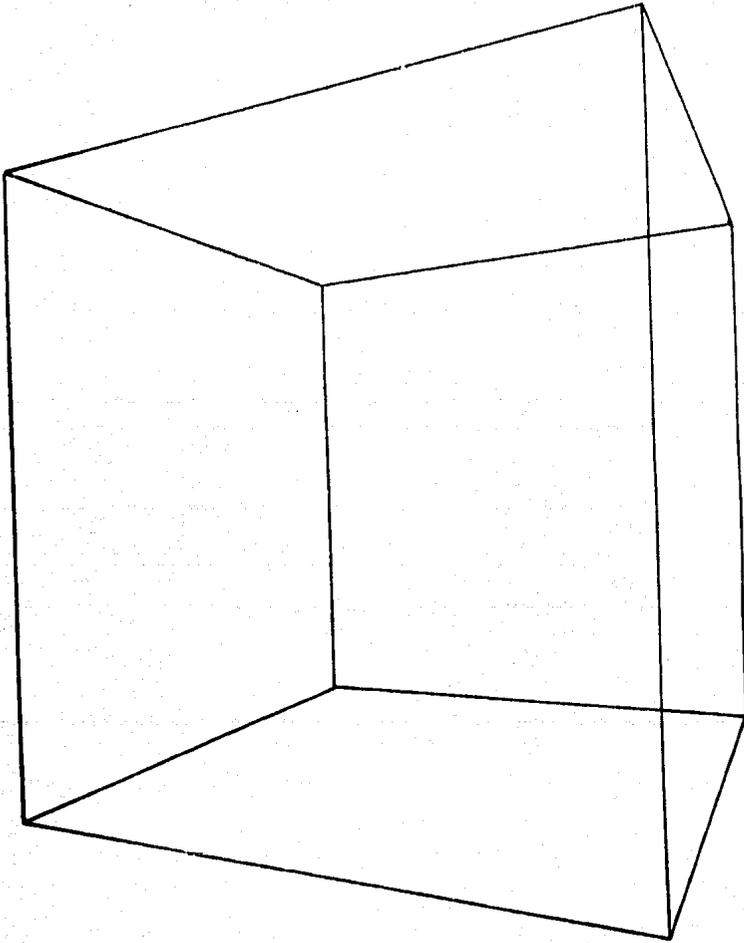
Algoritmo para trazar la gráfica.

```

Procedure Trazagrafica;
begin
  limpiapantalla;
  do
    begin
      lee (archivo) registro;
      if  $-5 \geq X_o, Y_o \leq 5$  then
        Grafica (Xo, Yo, marca)
      else
        if  $-5 \geq X_o_L, Y_o_L \leq 5$  then
          Grafica (XoL, YoL, marca)
        else
          if  $-5 \geq X_o_{LS}, Y_o_{LS} \leq 5$  then
            Grafica (XoLS, YoLS, marca)
          else
            grafica (Xo, Yo, 1)
        end
      until marca = 3;
    end de Trazagrafica;
  end

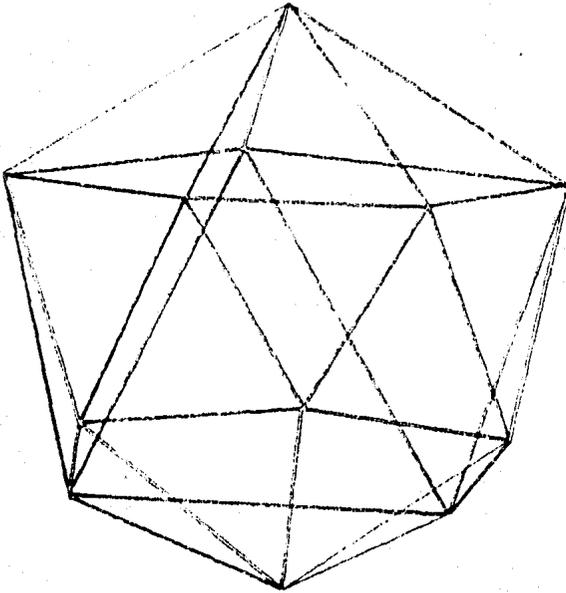
```

CUBO



GRAFICA 36
EO = 1 ; POSICION ORIGINAL

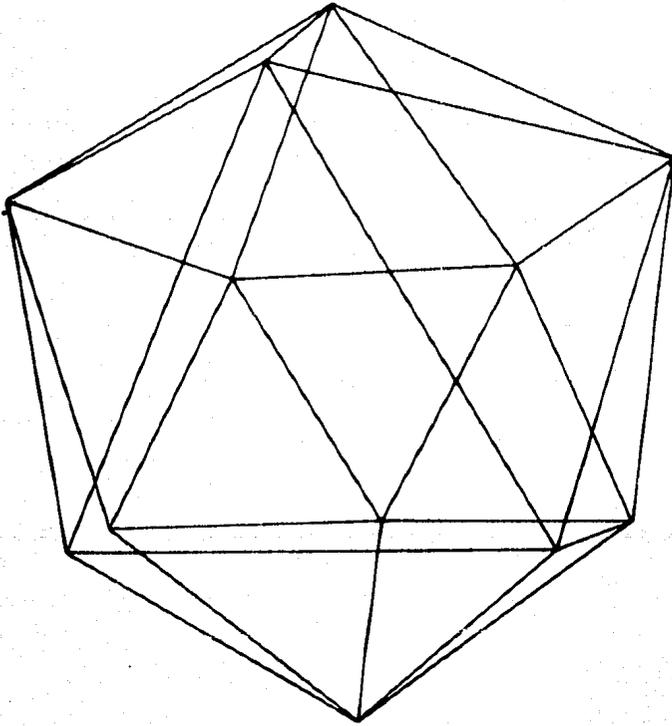
ICOSAEDRO



GRAFICA 37

EO = 14 ; GIRO PLS/Y

ICOSAEDRO



GRAFICA 38

EO = 8, TRAYECCION

Algoritmo para realizar la gráfica

```

Procedure Grafica (x,y, marca);
  Real x,y;
  Integer marca;
  begin
    Escala las coordenadas (x,y) a rangos pantalla;
    caracter ← coordenadas pantalla resultante de
    (x,y);
    if marca = 1 or marca = 2 then
      caracter ← caracter + primitivo 1
    else
      caracter ← caracter + primitivo 2;
    if buffer llene OR marca = 3 then
      ENVIA (buffer)
    else
      begin
        i ← i + 1;
        buffer [ i ] ← caracter;
      end;
  end de Grafica;

```

Ejemplos de los resultados obtenidos en este módulo se muestra en las gráficas 36, 37 y 38 que corresponden a las perspectivas de un cubo y un icosaedro respectivamente, este último se presenta con dos estaciones de observación (EO) distintas para mostrar diferentes perspectivas del mismo Cuerpo Geométrico.

La función del módulo de graficación juega un papel muy importante dentro del sistema ya que hace las veces del dibujante con la diferencia de que las perspectivas son trazadas en un tiempo de proceso mínimo. A continuación se presenta una tabla en la que se indica el tiempo que se requiere para trazar una gráfica en función del número de datos.

DATOS	Tiempo (seg) de Procesador	CG
12	0.5	CUBO
36	0.8	Icosaedro
100	1.0	Edificio

Tabla de tiempos para la
ejecución de gráficas.

Como se observa en la tabla para realizar 10 perspectivas del Cubo se requerirían 5 segundos u 8 segundos para el Icosaedro, etc. De donde podemos decir que el tiempo que se utiliza para trazar perspectivas es aceptable para fines de diseño que es el objetivo principal del sistema.

MODULO DE MOVIMIENTOS O EJECUCION.

Para el diseño óptimo de un proyecto es deseable apreciarlo desde diferentes ángulos con objeto de corregir detalles no previstos para afinar el resultado perseguido.

Estos diferentes ángulos del proyecto son proporcionados por las distintas perspectivas que de un mismo Cuerpo Geométrico pueden obtenerse por medio de movimientos que éste realice, en otras palabras el observador realiza una inspección del Cuerpo Geométrico desde diversas Estaciones de Observación (EO).

Mover el Cuerpo Geométrico y obtener la perspectiva resultante es la tarea que tiene a su cargo el módulo de movimientos o ejecución.

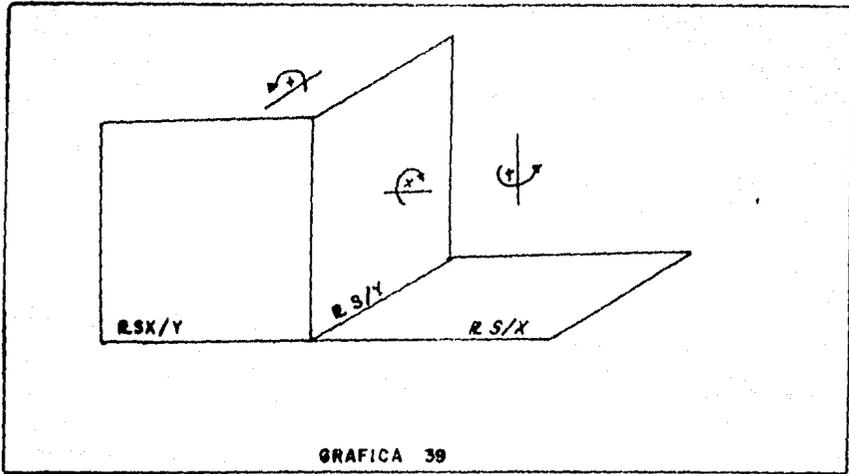
La filosofía bajo la cual está diseñado este módulo se apoya en los movimientos de traslación y giro, es decir, (perspectivas dinámicas) de Red Modular presentadas en el capítulo III del presente trabajo.

Los movimientos implementados son:

- . Movimiento de giro
- . Movimiento de traslación o trayección.

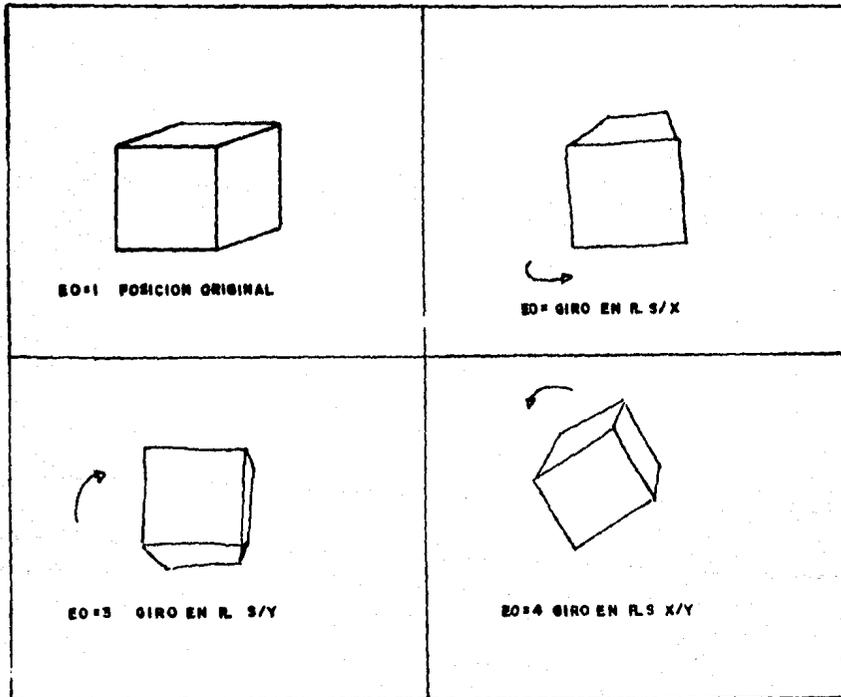
La condición que se debe cumplir para efectuar cualquiera de estos movimientos es que exista programa con datos, es decir, la presencia de un Cuerpo Geométrico (CG)

La ejecución de cualquier movimiento del CG se hace sobre el Plano de Simetría X/Y (PLS X/Y) o Espacio Modular (EM2), Lo que involucra el uso de coordenadas planimétricas. Esto se debe a que el Cuerpo Geométrico se mueve en el espacio y no en un plano.



GRAFICA 39

MOVIMIENTO GIRATORIO IMPLEMENTADO EN EL SISTEMA



GRAFICA 40

GIRO DE UN CUBO EN LOS TRES PLANOS DE RED MODULAR

Los movimientos de giro que un Cuerpo Geométrico (CG) puede realizar en el Espacio Modular (EM2), ocurren en cualquiera de los tres planos de Red Modular como se muestra en la gráfica 39. El sentido de recorrido que describe el CG al girarse en cualquiera de los planos se presenta a continuación:

• Giro en el Plano de Simetría X (PLS/X):

El Cuerpo Geométrico registrará un giro hacia la derecha del PLS/X para valores angulares positivos, y hacia la izquierda para valores angulares negativos (gráfica 40 EO=2).

• Giro en el Plano de Simetría Y (PLS/Y):

El Cuerpo Geométrico registraría un giro hacia atrás del PLS/Y para valores angulares positivos, y hacia adelante para valores angulares negativos (gráfica 40 EO=3).

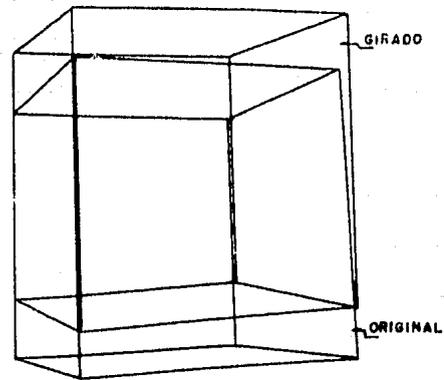
• Giro en el plano de Simetría X/Y (PLS X/Y):

El Cuerpo Geométrico registrará un giro hacia la izquierda del PLS X/Y para valores angulares positivos, y hacia la derecha para valores angulares negativos (gráfica 40 EO=4).

Para la ejecución de giros del CG en cualquiera de los tres planos es necesario establecer el origen sobre el cual se desea girar dicho cuerpo. Este origen corresponde a una terna de coordenadas planimétricas (X/Y/P), sin embargo, para brindarle mayor flexibilidad al diseñador al trabajar en este módulo, el origen puede omitirse, en este caso se tomará el centroide del Cuerpo Geométrico como origen.

Los comandos que se emplean para el movimiento giratorio consisten:

En la especificación del plano sobre el cual se desea girar al Cuerpo Geométrico; y un predicado formado para el centro de giro u origen y del ángulo de variación. Expresando estos comandos



GRAFICA 41

GIRO DE UN CUBO CON RESPECTO A SU POSICION ORIGINAL

en notación Backus Naur (BNF) tenemos:

GIRO :: = PLANO | PREDICADO
 PLANO :: = PLS/X | PLS/Y | PLS X/Y
 PREDICADO :: = ORIGEN Y ANGULO | ANGULO
 ORIGEN :: = COORDENADAS PLANIMETRICAS (X/Y/P) | NADA
 ANGULO :: = NUMERO REAL
 NADA :: = CENTROIDE DEL CUERPO GEOMETRICO.

Cada giro del Cuerpo Geométrico de K grados no se hace con respecto a la posición original de éste, sino se toma la posición última del mismo, como es el caso del giro realizado en la gráfica 41.

En el movimiento de traslación el Cuerpo Geométrico es desplazado hacia arriba o hacia abajo, de izquierda a derecha o hacia adelante o atrás, es decir acercarse o alejarse, del espacio modular EM2 que en perspectiva es el Plano Proyectivo de Campo Visual PLPCV.

Los comandos del movimiento de traslación se representan mediante la siguiente notación BNF:

TRASLACION :: = TRASLACION | PREDICADO
 TRASLACION :: = TRASLACION
 PREDICADO :: = TERNA (X/Y/P)
 X/Y/P :: = NUMEROS REALES

La ejecución de estos movimientos de giro y traslación se les llama funciones (en el sistema) y se representan con la siguiente notación BNF:

FUNCION :: = GIRO | TRASLACION

donde Giro y Traslación son las funciones definidas anteriormente.

EJECUCION DE FUNCIONES

El módulo de movimientos tiene un analizador sintáctico que sigue las notaciones Backus Nahur de cada movimiento, de tal manera que le permiten tomar las siguientes acciones:

1. Identificar la función solicitada, y en base a ello
2. Actualizar la memoria de cálculos
3. Registrar la función detectada en un área auxiliar de memoria, en la cual se lleva la historia de las funciones realizadas.
4. Desplegar en pantalla la función que se ejecuta en este momento (mediante el monitor).

EL PROCESO QUE SE REALIZA ES EL SIGUIENTE:

La identificación de una función consiste en detectar si es de giro o traslación, en caso de ser giro se determina sobre que plano se va a efectuar éste. Una vez obtenida la función se analiza el predicado de la misma de acuerdo a las notaciones Backus Nahur establecidas.

Un error ocurre en caso de que la función solicitada no esté definida en el sistema, o si el predicado de ésta sea sintácticamente incorrecto.

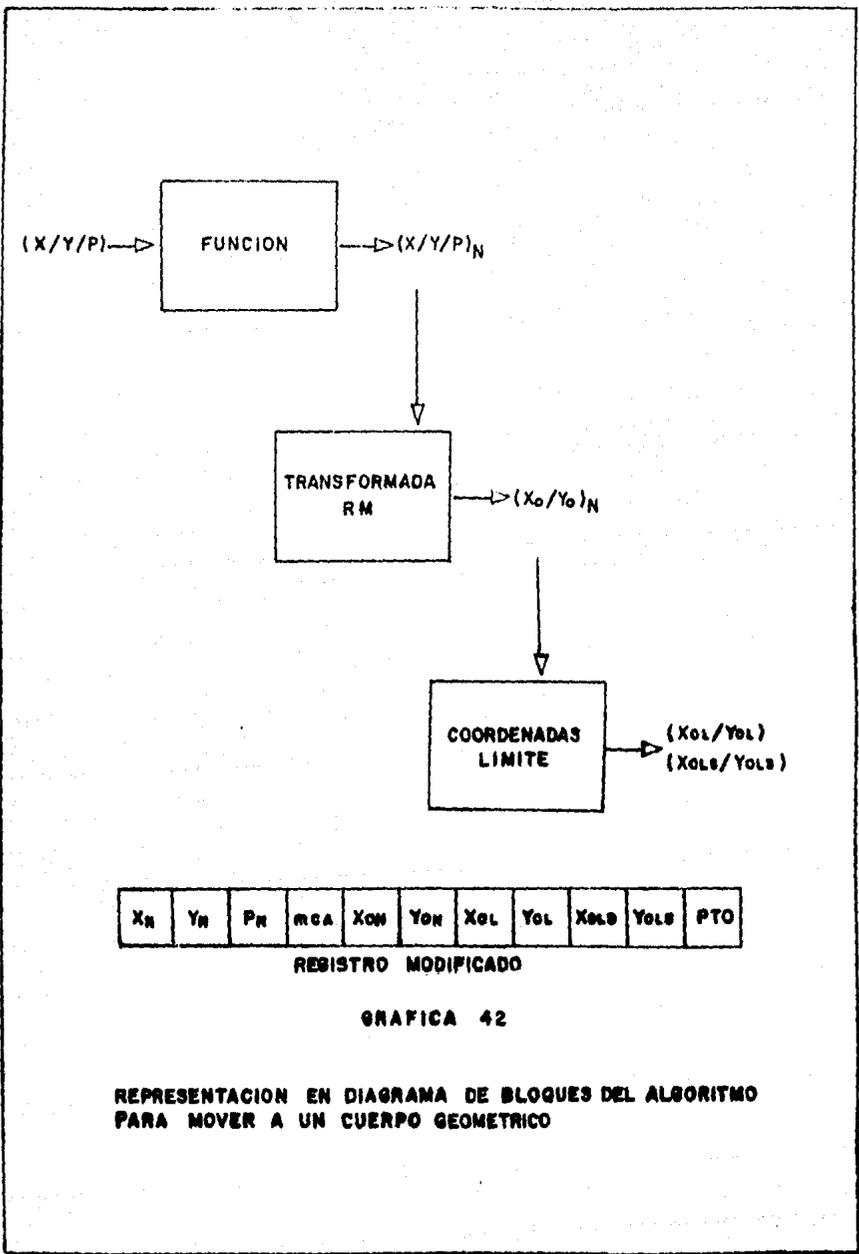
Cabe aclarar que: si se aplica una función sin error alguno, se puede enviar nuevamente la misma función omitiéndose el predicado correspondiente, en este caso se tomarán los datos registrados en la memoria de cálculos. Esta memoria consiste en un vector en el cual se almacenan:

- . el origen si la función es de giro
- . el ángulo si la función es de giro
- . los módulos de traslación (X/Y/P) si la función es de traslación.
- . incrementa la cuenta de:
 - Estación de Observación (EO)
 - Número de funciones de giro realizadas
 - Número de trayecciones efectuadas

Cada vez que el Cuerpo Geométrico es movido, se calcula el centroide en su nueva posición, el cual se toma como origen de giro si éste se omite en el predicado de la función. En este caso se actualiza la memoria de cálculos con el nuevo origen, y el ángulo de giro se mantiene constante a menos que se exprese en el predicado de la función. Este tipo de movimiento siempre será relativo a la última posición del Cuerpo Geométrico.

Para girar al Cuerpo Geométrico con un mismo origen y ángulo de giro en cualquiera de los planos de Red Modular (PLS/X, PLS/Y, PLS X/Y) se utiliza un comando del módulo de movimientos que mantiene constantes los últimos datos registrados en la memoria de cálculos y solo podrán actualizarse mediante otro comando que deshabilita esta opción.

El desplegado de la función en pantalla consiste en escribir en un arreglo usado como buffer, la estación de observación (EO), la función a ejecutar y su predicado y la amplitud de campo visual (ACV) empleada. Una vez lleno el buffer se despliega en pantalla y al mismo tiempo se registra en un archivo en disco para llevar una bitácora de movimientos del Cuerpo Geométrico, misma que se puede desplegar en pantalla en el momento que el usuario lo solicite. El objeto de utilizar un arreglo como buffer de edición es evitar el paso por el procesador de entrada y salida de la computadora consumiendo de esta manera, menor tiempo de procesador. La ventaja de tener una bitácora de movimientos es que se puede editar por impresora para conocer que



GRAFICA 42

REPRESENTACION EN DIAGRAMA DE BLOQUES DEL ALGORITMO PARA MOVER A UN CUERPO GEOMETRICO

función corresponde a la gráfica obtenida.

Para ejecutar el movimiento del Cuerpo Geométrico se aplica la función solicitada, para ello se tiene una rutina por movimiento, es decir, tres para giro y una para traslación. Los algoritmos que estas rutinas utilizan son los que emplea Red.Modular para movimientos del Cuerpo Geométrico (capítulo III).

Todas las rutinas del módulo de movimientos emplean el mismo procedimiento que se explica a continuación:

1. Se lee el registro i del archivo auxiliar 2 y se pone en un arreglo temporal que tiene la misma imagen del registro (ver gráfica 42).
2. Se aplica el algoritmo de movimiento correspondiente sobre la terna de coordenadas planimétricas y se obtiene una nueva terna $(X/Y/P)$.
3. Se aplica la Transformación RM a la terna resultante en 2 para conseguir sus coordenadas proyectivas (X_o/Y_o) .
4. Se escribe en el registro i del archivo auxiliar 2 el arreglo temporal con las coordenadas planimétricas y proyectivas modificadas. Nótese que los demás campos del registro i no se alteraron.
5. Si "marca" no es ni 1 ni 2 en el registro i se ejecuta 6, de lo contrario se ejecuta 7.
6. Se revisa que las coordenadas proyectivas (X_o/Y_o) cumplan las condiciones de inclusión, de no cumplirse estas se lee el registro $i-1$ del archivo auxiliar 2 y, mediante el algoritmo de caso límite, se obtienen las coordenadas límite de primera intersección (X_{oL}/Y_{oL}) del punto i . Después se lee el registro $i+1$ y, con el mismo algoritmo, se obtienen las coordenadas límite de segunda intersección (X_{oLS}/Y_{oLS}) .
7. Se escribe el registro i en el archivo auxiliar 2 con los valores límite en caso de haberlos.

8. Si la marca es igual a 3 (fin de datos) y el archivo aun tiene datos (para evitar un fin de archivo EOF) entonces se incrementa i ($i \leftarrow i + 1$) y se regresa a 1, en caso contrario se ejecuta 9.
9. Se cierra el archivo auxiliar 2 y se llama al módulo de graficación para trazar la perspectiva resultante.
10. Fin del procedimiento.

Como se observa en este módulo se resuelven los casos de intersección con la ventana mencionados en el módulo de graficación. A continuación se muestra el algoritmo empleado en este módulo.

```

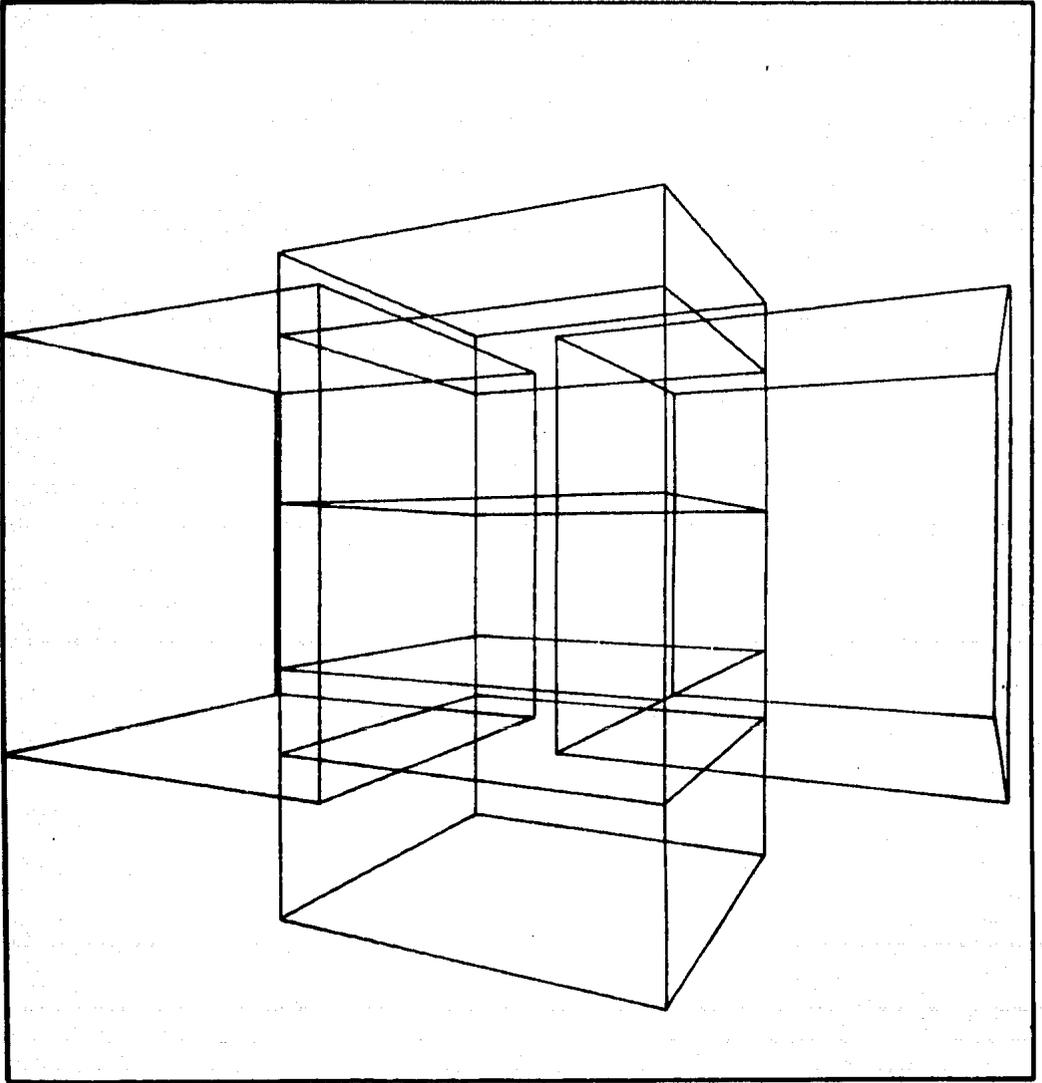
{ Procedure Movimientos:
  begin
    fin ← false:
    do
      begin
        leefunción:
        if función valida then
          if predicado correcto then
            begin
              mensaje "función X y predicado";
              write (archivo, "función X y predicado");
              ejecuta función X; % i. e. movimiento
              llama módulo de graficación;
            end
          else
            error "predicado erroneo"
        else
          if comando de control then
            fin ← true
          else
            error "función erronea"
        end
      until fin
    end de Movimientos;
  
```

Debido a las características de la pantalla graficadora es necesario limpiarla antes de trazar gráfica alguna, por lo que no es posible apreciar la evolución del movimiento de un objeto en este dispositivo. Sin embargo para poder observar la secuencia del movimiento de un Cuerpo Geométrico se puede conectar el graficador (plotter) en paralelo con la pantalla graficadora para superponer las gráficas en el papel usando uno o varios colores a fin de obtener el resultado deseado.

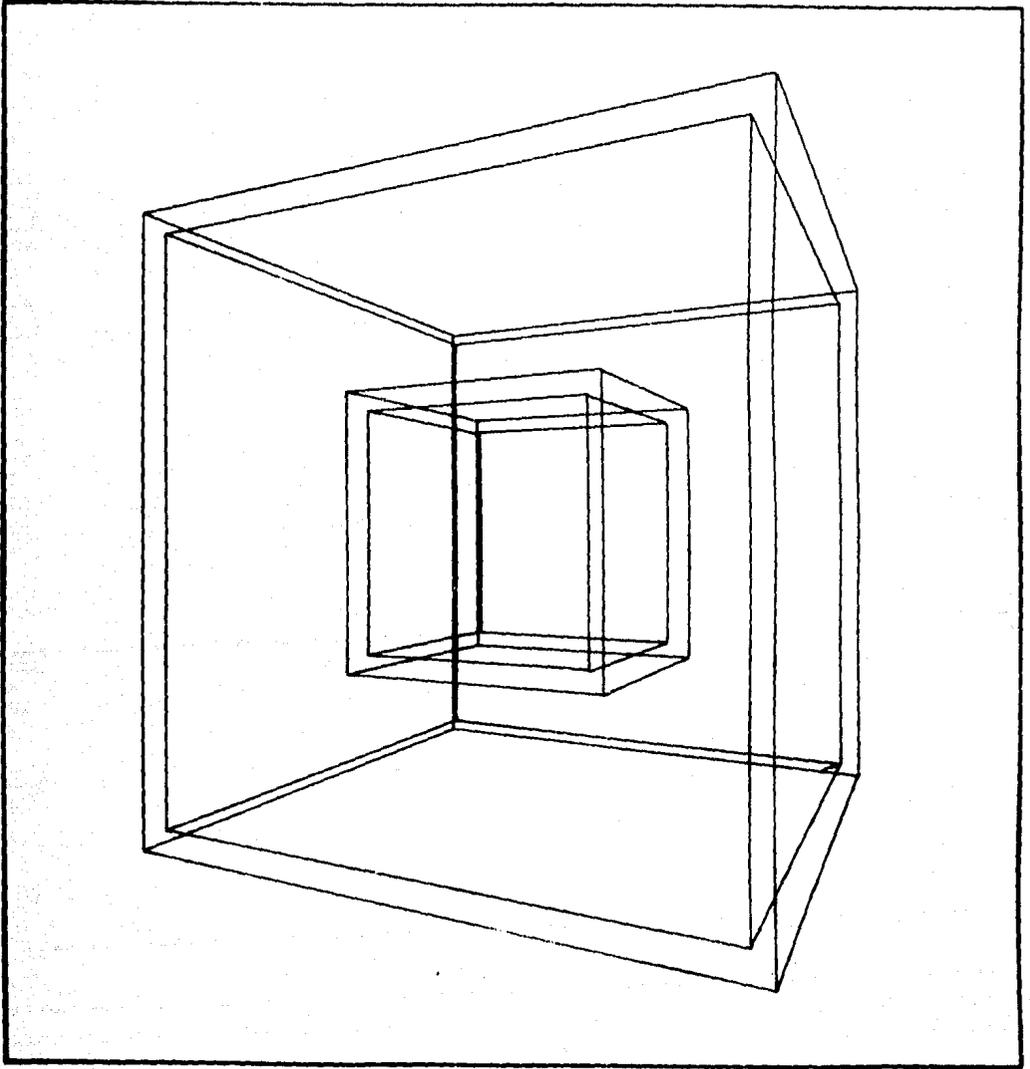
A continuación se muestran las gráficas de los movimientos realizados por un ICOSAEDRO Y UN CUBO. En todas las gráficas se utilizó una Amplitud de Campo Visual (ACV) de 60 grados con excepción de las del ICOSAEDRO con $EO=14$, $EO=16$ y $EO=19$ en las que se usaron ACV de 80, 45 y 63 grados respectivamente para una misma posición del ICOSAEDRO, obteniéndose diferentes resultados perspectivos equivalentes a haber usado distintos lentes para la toma de la fotografía de un mismo elemento.

En las gráficas del CUBO se pueden apreciar secuencias de movimientos en un mismo plano que permiten ver la evolución del movimiento, por esta razón no se incluye en estas gráficas la Estación de Observación (EO) ya que cada posición del CUBO corresponde a una EO distinta.

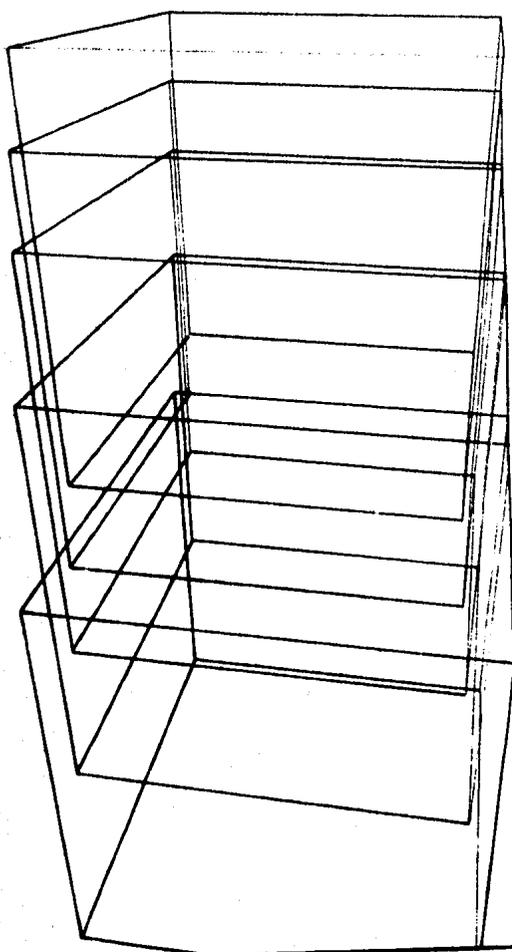
La gráfica final corresponde a un edificio para mostrar un caso de aplicación en el diseño arquitectónico.



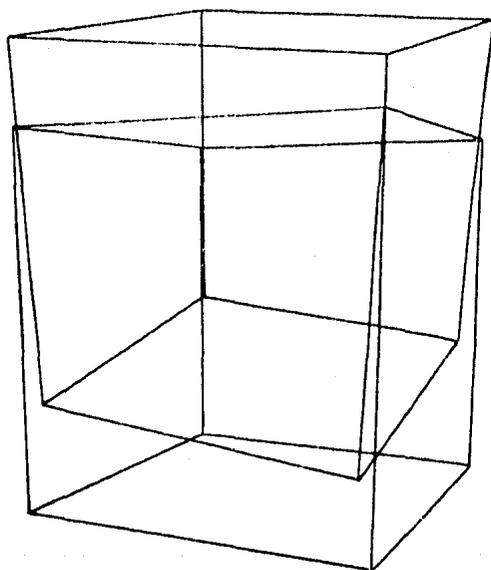
TRAYECCIONES DE UN CUBO



TRAYECCIONES EN ALEJAMIENTO DE UN CUBO

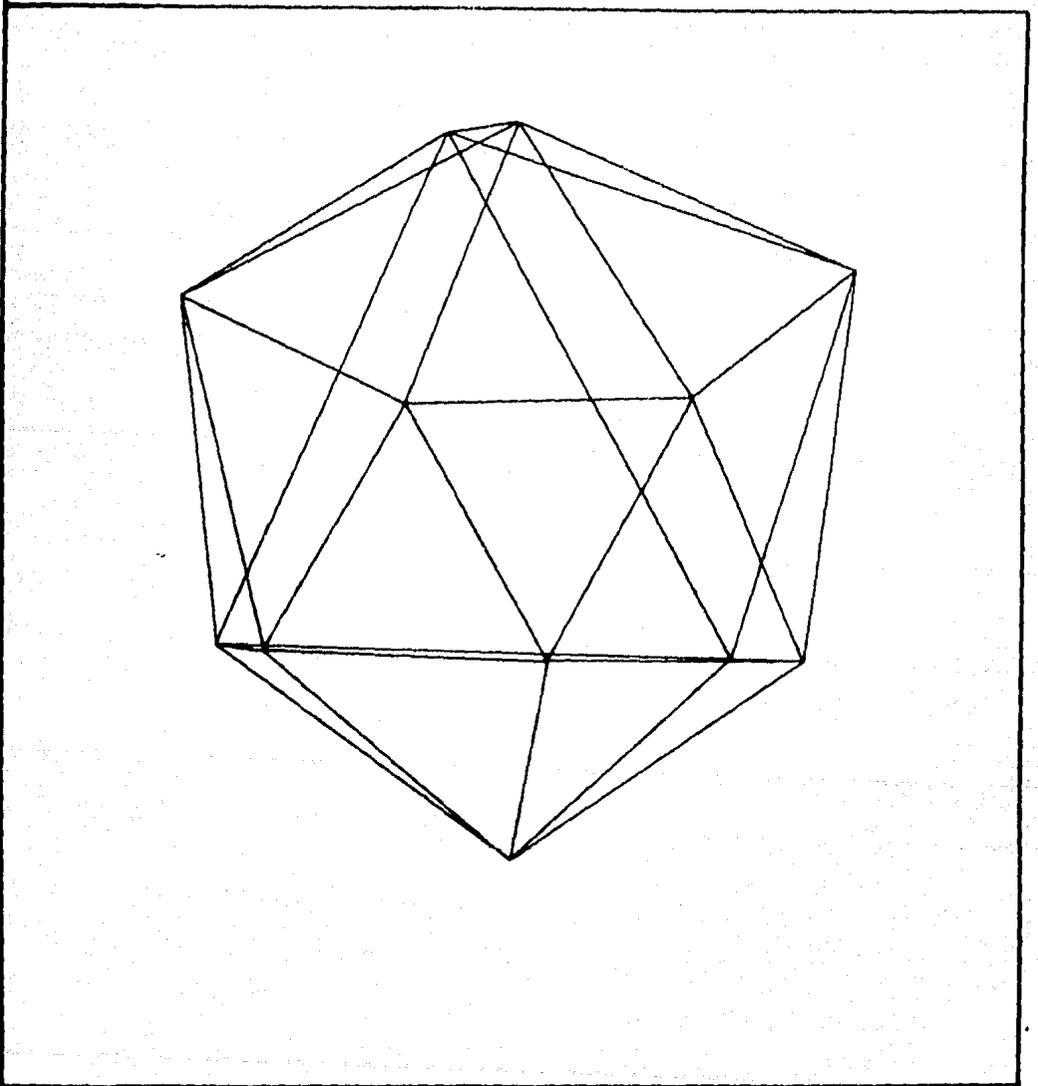


TRAYECCION DE UN CUBO

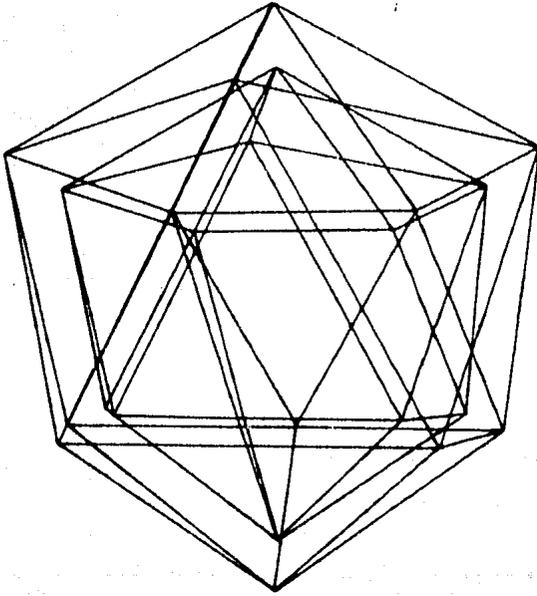


EO= 23 ; GIRO DE UN CUBO

ICOSAEDRO

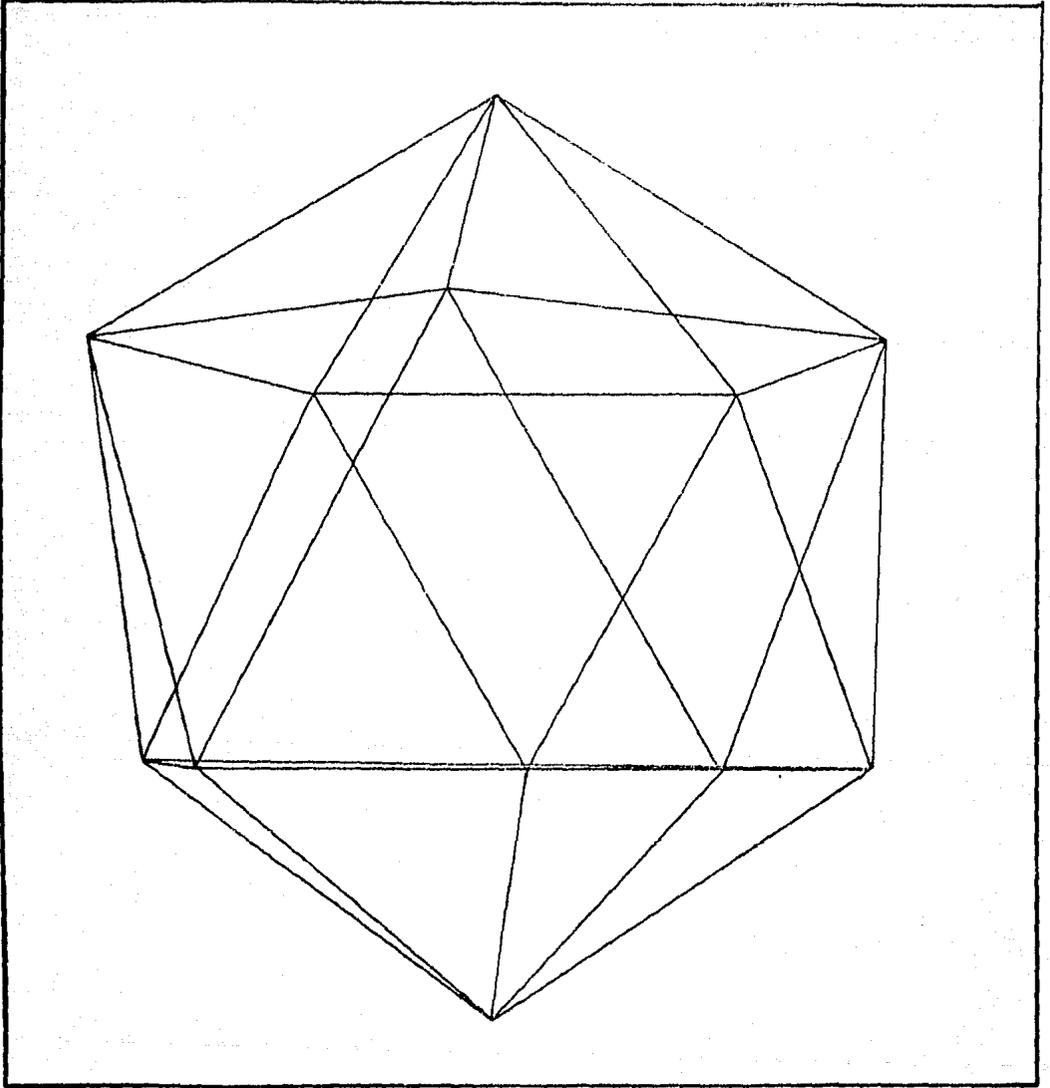


EO=5; TRAYECCION



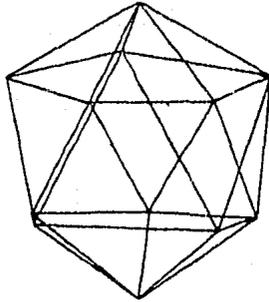
TRAYECCIONES DE UN ICOSAEDRO

ICOSAEDRO



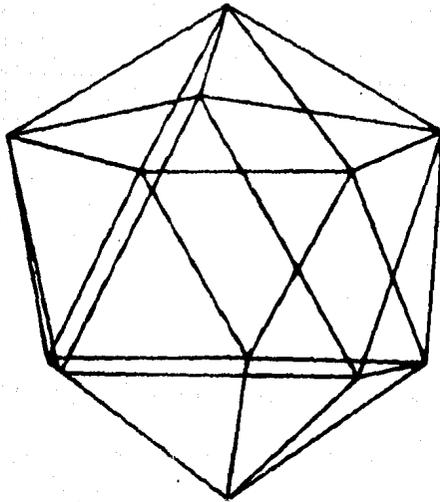
$E0=14$; $ACV=80$ GRADOS

ICOSAEDRO

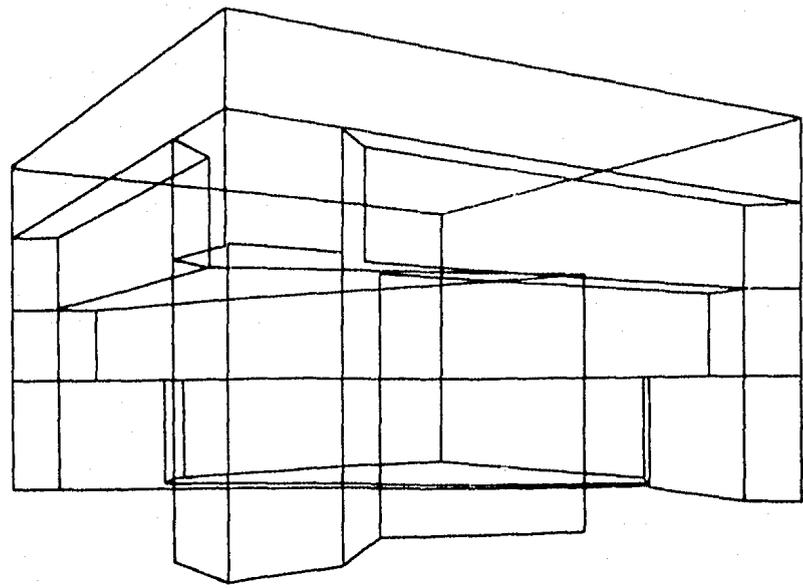


EO=16 ; ACV = 45 GRADOS

ICOSAEDRO



$E=19$; $ACV = 63$ GRADOS



PERSPECTIVA DE UN EDIFICIO RESUELTO POR EL SISTEMA

MODULO DE SALIDA

La presentación en perspectiva de un Cuerpo Geométrico permite una mejor concepción del desarrollo de un proyecto en la etapa del diseño como se mostró en los módulos de graficación y movimientos. Sin embargo, no debemos olvidar que la perspectiva de un Cuerpo Geométrico, y en general de cualquier gráfica se traza a partir de datos no gráficos contenidos en un archivo (generalmente en disco), que en Red Modular corresponden a las coordenadas proyectivas (X_o/Y_o) obtenidas mediante la Transformación RM. Este conjunto de pares ordenados representan un valor, en módulos, en el Plano Proyectivo de Campo Visual (PLPCV), en donde cada uno de estos pares ordenados (X_o/Y_o) es mapeado para trazar la gráfica resultado, es decir, la perspectiva del Cuerpo Geométrico.

Si bien una gran cantidad de datos presentados en forma de tabla no dicen nada, resulta interesante conocerlos para diferentes propósitos, razón por la cual fué creado el módulo de salida, cuya función es proporcionar los valores numéricos (en módulos) de las coordenadas planimétricas y proyectivas del programa activo, valores que pueden editarse tanto en pantalla como en impresora.

Estos valores coordenados se presentan a manera de tabla para permitirle al diseñador, por un lado trazar la gráfica manualmente con mayor precisión y calidad de línea; y por el otro conocer el valor real (en módulos), de las coordenadas proyectivas a fin de mover el Cuerpo Geométrico eficientemente en el módulo de movimientos.

Ya que este módulo trata de la edición de datos, obliga a la existencia de un programa con datos para que estos puedan ser impresos. En impresora se obtiene un listado en el que se anejan los valores coordenados límite en primera intersección como se muestra en la gráfica 43.

En cambio cuando la salida es por pantalla no se proporcionan las coordenadas límite por el número de columnas disponibles.

A diferencia de los módulos de graficación y movimientos, aquí se emplean ambos archivos auxiliares de datos, lo que permite editar no solo los datos originales del programa (contenidos en el archivo auxiliar 1), sino también los datos resultantes de algún movimiento (contenidos en el archivo auxiliar 2).

Como se observa en la gráfica 43 se proporciona además de los datos, información adicional como: nombre del programa; fecha de creación; tipo de datos, es decir, originales o los resultantes de un movimiento y en este caso se indica el tipo de movimiento efectuado; y la Amplitud de Campo Visual (ACV) empleada. Que corresponden a la gráfica 44.

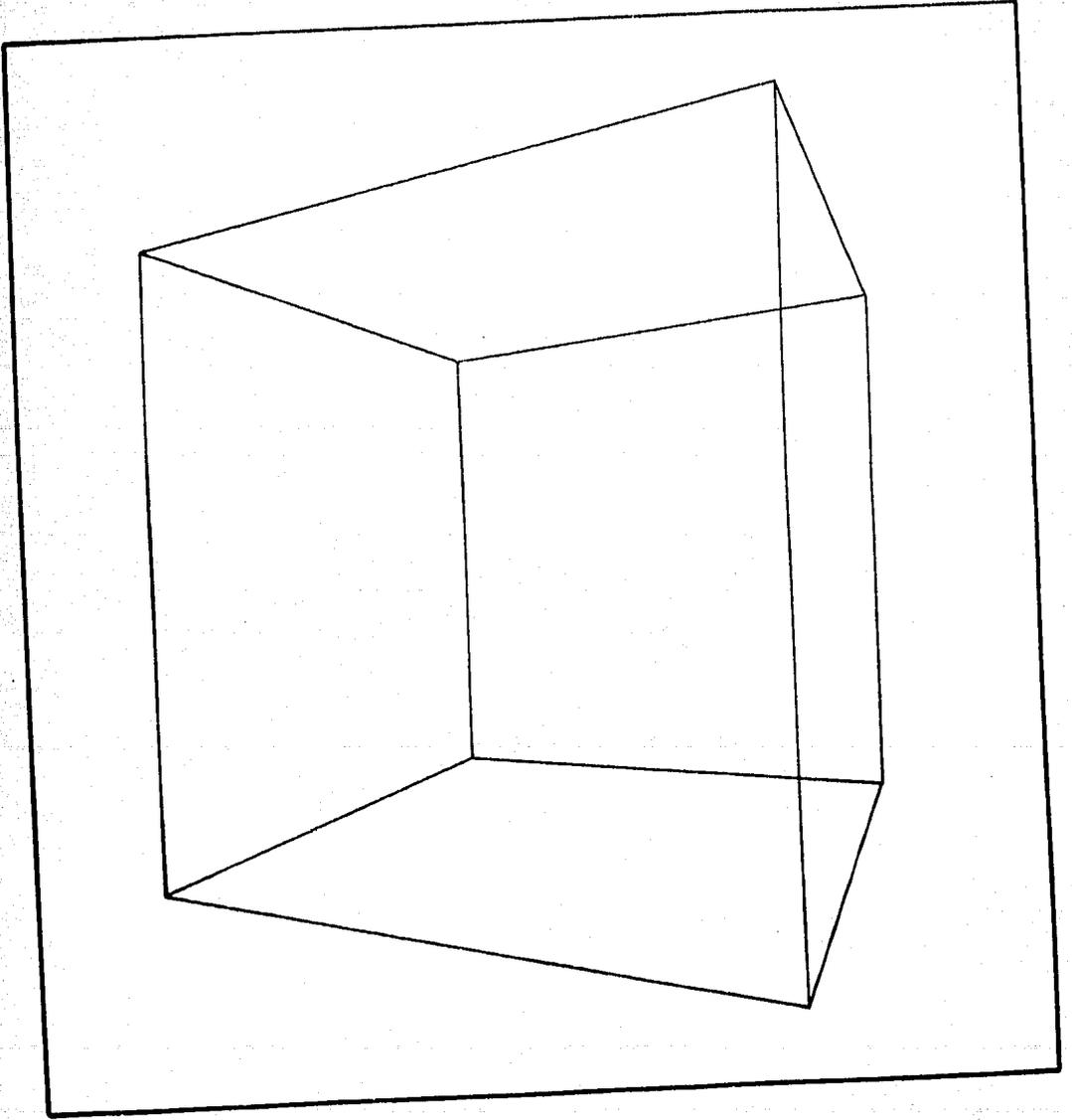
El comando usado en este módulo puede o no tener predicado, la siguiente notación Backus Naur (BNF) muestra su estructura sintáctica.

```

SALIDA      :: = SALIDA | SALIDA PREDICADO
SALIDA      :: = SALIDA
PREDICADO   :: = DATOS | DATOS DISPOSITIVO
DATOS       :: = ORIGINALES | MODIFICADOS | AMBOS
AMBOS       :: = ORIGINALES Y MODIFICADOS
DISPOSITIVO :: = PANTALLA | IMPRESORA | NADA
NADA        :: = IMPRESORA

```


CUBO



GRAFICA 44
EO=1 ; POSICION ORIGINAL

IV. 3 PROCEDIMIENTO EN LA ELABORACION DEL SISTEMA RM.

Como se ve a lo largo de este capítulo, la computarización del método Red Modular se realizó en diversas etapas que permitieron un avance progresivo, sistemático y retroalimentado en la puesta en marcha del sistema. A continuación se resumen estas etapas a manera de guía en la elaboración de sistemas CAD.

Etapas de Diseño

Consiste en definir los objetivos y características del sistema CAD para satisfacer las exigencias del proyecto de investigación y elaborar el 'modelo' que contiene el diseño del sistema. Este 'modelo' está constituido por un conjunto de módulos o bloques independientes que describen los procesos necesarios para el desarrollo del sistema como son: graficación de perspectivas, ejecución de movimientos del Cuerpo Geométrico, entrada y salida de datos, etc. Al mismo tiempo los módulos deben estar relacionados entre sí por medio de un módulo maestro que controla los procesos mediante un lenguaje de comandos que permiten que el usuario opere el sistema interactivamente, es decir, diálogo hombre-máquina.

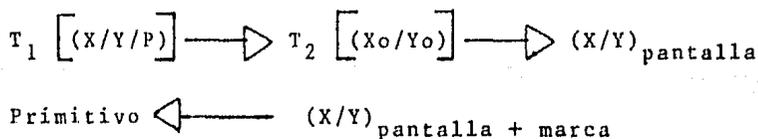
Un módulo realiza uno o más procesos del mismo tipo, por lo tanto puede tener sus propios comandos para ofrecer diversas opciones; como el módulo de movimientos.

El 'modelo' comenzó con el módulo de graficación para asegurar que era factible graficar perspectivas (capítulos I, II y III), de lo contrario no se tendría un sistema de graficación y no se podría continuar con el diseño del 'modelo'

Para el diseño de este módulo se consideró un 'modelo de graficación' que consta de los siguientes elementos:

- a) Par coordinado (X/Y), sistema de referencia X/Y y ventana. Dados por la transformación RM y el Plano Proyectivo de Campo Visual (PLPCV) respectivamente.
- b) Algoritmo de transformación de coordenadas reales a coordenadas pantalla para graficar un punto, lo que requiere de 'factor de escala' dado por las dimensiones de la pantalla, tamaño de la ventana, Plano Proyectivo de Campo Visual (PLPCV), y por las condiciones de inclusión o límite de Campo Visual (LCV) de Red Modular (ver módulo de graficación).
- c) Caracteres de control para efectuar los primitivos gráficos, para ello se definió la marca.
- d) Algoritmo para resolver intersección (clipping) del Cuerpo Geométrico con la ventana o Plano Proyectivo de Campo Visual (PLPCV) (capítulo III).
- e) Algoritmo de recorridos de nodos para trazar la gráfica.

Estos elementos se resumen de la siguiente manera:



donde

T_1 : Transformación RM.

T_2 : Transformación de coordenadas reales a coordenadas pantalla.

Con el 'modelo de graficación' terminado se diseñaron los demás módulos del sistema para completar el 'modelo', utilizándose el siguiente procedimiento:

- i) Definir la función y las opciones que debe tener cada módulo así como los comandos que acepta ese módulo mediante el uso de algún metalenguaje, por ejemplo notación Backus Nahur (BNF), diagramas de ferrocarrilero*, etc.
- ii) Elaborar el (los) algoritmo(s) que cumplan las notaciones descritas por el metalenguaje, ajustándose además, a las características del equipo a emplear (de lo contrario se tendrían algoritmos imposibles de implementar). El uso de un pseudolenguaje no solo facilita la comprensión del algoritmo sino también ayuda mucho a su programación.
- iii) Probar cada algoritmo para optimizarlo en tiempo de procesador, y programación (espacio en memoria) para incrementar su eficiencia.

Este procedimiento se hizo por cada módulo, consiguiéndose un avance progresivo.

Etapa de implementación

Consiste en la programación de los módulos del modelo en el equipo de cómputo seleccionado, en algún lenguaje de alto nivel permitido por dicho equipo.

Cada módulo debe programarse respetando las reglas de programación estructurada de documentación, sangría, flujo en una sola dirección, nombres de variables adecuados, etc., para entender lo que se está haciendo.

(*) El manual de operación contiene los diagramas de ferrocarrilero que describe en detalle cada módulo del sistema.

Una vez que un módulo es programado se prueba exhaustivamente para afinar detalles no previstos en el diseño, y es depurado hasta satisfacer los requerimientos solicitados.

Terminados todos los módulos se ensamblan para integrar el sistema en su totalidad, y se prueba la comunicación tanto entre módulos como entre el usuario y la computadora, y se hacen los ajustes necesarios al editor del sistema.

Etapa de Prueba

Consiste en probar el sistema para lo que fué creado, es decir, en este caso diseño arquitectónico asistido por computadora. Para ello se utilizaron Cuerpos Geométricos conocidos, CUBO e ICOSAEDRO, para afinar detalles y hacer los ajustes necesarios no previstos en el diseño e implementación del sistema. Después se probó con casos reales o con otros Cuerpos Geométricos de interés.

Para medir la eficiencia del sistema se consideraron los siguientes parámetros: Tiempos de procesador y de entrada y salida para graficar, realizar movimientos y acceso a la Base de Datos del sistema; velocidad de respuesta del editor para atender una petición; espacio en memoria requerido para operar el sistema. Y determinar, en base a los recursos utilizados, si el sistema satisface las necesidades para las que fué creado con un consumo mínimo de recursos de cómputo.

Para completar el sistema de diseño asistido por computadora debe anexarse el manual de operación en el que se explica su funcionamiento. En este trabajo no se publica el manual de operación del sistema porque no está en los objetivos de la tesis.

V. CONCLUSIONES

Las exigencias de la investigación arquitectónica en la búsqueda de métodos de diseño llevaron a la creación del método Red Modular. La computarización de RM satisface nuevos objetivos en la investigación de perspectivas en general y aún cuando el objetivo inicial era sólo graficar perspectivas esta computarización conforma un sistema de diseño asistido por computadora que brinda al diseñador algo más que un dibujante autómatas; es una herramienta de trabajo que actualmente se encuentra disponible en la computadora Burroughs B7800 del Programa Universitario de Computo (PUC) y puede utilizarse desde cualquier terminal remota con equipo de graficación.

El sistema computarizado Red Modular permitió mostrar que Red Modular es un método computarizable de características universales porque resuelve cualquier caso de perspectiva según se vió en el capítulo IV. Además debido a las características del sistema a la facilidad de operarlo interactivamente mediante un diálogo nombre-máquina y por la velocidad para graficar perspectivas de Cuerpos Geométricos resulta una herramienta importante en:

- . Docencia: Resolviendo problemas de perspectiva y/o de geometría descriptiva en forma interactiva, ya que el sistema permite mover Cuerpos Geométricos, ayudando al alumno a comprender de manera objetiva el espacio y abriendo su imaginación en el diseño.
- . Investigación: Analizando distintos casos de perspectiva o de algún diseño en particular, le permiten al investigador proponer alternativas, metodologías, etc., en su campo de trabajo.
- . Areas diversas: Perspectivas de: máquinas herramientas, automóviles, estructuras, etc. que permiten auxiliar al diseñador de diversas áreas como la ingeniería civil, ingeniería mecánica, diseño gráfico, etc. y en general en cualquier área que involucre diseño en tres dimensiones.

Por lo anterior el sistema computarizado RM satisface los objetivos para los que fue creado.

En la elaboración del sistema fue necesario diseñar un 'modelo' constituido por un conjunto de módulos o bloques que ejecutan procesos específicos y que se ajustan a las necesidades del diseño. La utilización de una estructura modular permitió un avance progresivo, sistemático y retroalimentado en el desarrollo del sistema ya que se realizó cada módulo de manera independiente. Además es posible anexar nuevos módulos usando la metodología descrita en el capítulo IV, consiguiendo un sistema dinámico que crece en función de las necesidades del diseñador (usuario).

El uso de diagramas de ferrocarrilero y de notaciones Backus Naur (BNF) fueron fundamentales en el diseño para definir la estructura y sintaxis de cada módulo del 'modelo'. La ventaja de usar metalenguajes es que permiten visualizar clara y objetivamente lo que se está haciendo y facilita la elaboración del algoritmo. Asimismo la utilización de pseudolenguajes para escribir cada algoritmo hizo más sencilla su implementación en un lenguaje estructurado (e.g. algol).

La eficiencia del sistema se midió tanto en la facilidad de operación, velocidad para atender una petición del usuario y opciones que ofrece, como en la cantidad de recursos de cómputo utilizados en su operación como son: tiempo de procesador para: graficar perspectivas, mover Cuerpos Geométricos y acceso a la Base de Datos; espacio ocupado en memoria; tiempo de entrada y salida; etc. obteniéndose resultados satisfactorios.

Debido a las características del equipo de graficación empleado por el sistema, no fué posible apreciar la secuencia del movimiento de un Cuerpo Geométrico. Para analizarlo se conectó el graficador (plotter) en paralelo con la pantalla graficadora y se superpusieron las graficas en el plotter, como puede apreciarse en el capítulo IV.

Ahora bien si se usara una pantalla graficadora de refresco o dinámica sería posible ver el efecto de movimiento. El sistema está provisto para operar en este tipo de dispositivos donde los movimientos se realizan a intervalos de tiempo, por lo tanto habrá solo que hacer los ajustes necesarios al módulo de movimientos para que el sistema funcione adecuadamente.

Empero, la investigación no se detiene aquí ya que se pretende que el sistema sea más versátil para ofrecer al diseñador más elementos en el desarrollo de sus proyectos tales como resolver líneas ocultas (hidden lines) para dar volumen a Cuerpos Geométricos borrando o punteando líneas y teoría de sombras para corregir ubi-

cación de casa habitación, edificios, etc., sabiendo en que partes del Cuerpo Geométrico se hacen incidir los rayos de sol a determinada hora del día. Los algoritmos para resolver líneas ocultas y teoría de sombras se elaboraron y se dejaron para una segunda versión del sistema en el que se contemplan nuevos objetivos.

Es evidente que el sistema computarizado Red Modular ofrece toda una gama de posibilidades en el diseño arquitectónico y hace surgir ideas para satisfacer nuevas necesidades como los de líneas ocultas y teoría de sombras antes citados. Ideas que crean nuevos objetivos que requieren ser implementados en la computadora para satisfacerlos. Una forma de implementar esta clase de ideas se expuso en el presente trabajo, que ofrece una metodología alternativa en el diseño, desarrollo e implementación de sistemas de diseño asistido por computadora.

VI BIBLIOGRAFIA

M. en A. Tomás García Salgado
 Perspectiva Modular Aplicada al
 Diseño Arquitectónico (RM1)
 Vol. I 2a. Edición (1982) Ed. UNAM.

M. en A. Tomás García Salgado
 Perspectiva Modular Aplicada al
 Diseño Arquitectónico (RM2)
 Vol. II 1a. Edición (1982) Ed. UNAM.

Rual V. Churchill 2a. Ed.
 Complex Variables and Applications
 International student edition -
 McGraw - Hill Kogakusha
 (mapping, transformation).

Enciclopedia de la Técnica y la Mecánica
 Ed. Nauta Barcelona, S.A.
 Vol. 6 pp 74-75 y 254-256
 Vol. 2 pp 80-84
 (Punto de fuga, Perspectiva, Proyecciones)

Velasco Reyes/Aguirre Rodríguez/
 Rivera Benitez
 Apuntes de Estadística Aplicada
 Facultad de Ingeniería/UNAM 1980
 Capítulos 1 y 2 (Regresión Lineal)

W. G. Mc Dean/E.W. Nelson
 Mecánica Técnica
 Serie Schaum - Mc Graw - Hill

Murray R. Spiegel
 Análisis Vectorial
 Serie Schaum - McGraw - Hill

Louis Leighthold 1a. Ed./Editorial Trillas
 El Cálculo con geometría Analítica
 (Sistemas coordenados, mapeo, análisis vectorial)

Standish, Thomas A,
Data structure Techniques.
Addison Wisley

Niklaus, Wirth,
Algorithms + Data structures = Programs
Printence-Hall.

Wiederhold, Gio,
Database Design
Mc Graw-Hill.

Wolfgang, D. Giloi,
Interactive computer graphics data structures algorithms
lenguages.
Printence Hall Inc.

Fis. Rafael Carmona Paredes,
Fundamentos de la graficación a través de computadoras
CSC/UNAM 1980.

Fis. Rafael Carmona Paredes,
Graficación Niveles I y II
CSC/UNAM 1980.

Tektronix user references

- Interactive digital plotter model 4662.
 - Interactive graphic display models 4006, 4012
- Tektronix committed to excellence.

Fis. Rafael Carmona Paredes,
Uso del equipo de graficación tektronix
- Terminal modelo 4006
- Graficador modelo 4662
CSCS/UNAM 1981.

J. W. Demie,
Perspective geometric and computer graphics: techniques
and application in computer-aided-design.
A Gunther et al (eds) International computing symposium 1973
North Holland publ. C. 1974.

Newman W. M. and Sproull,
Principles of interactive computer graphics
Mc Graw - Hill Book Company NY-NY. 1973.

Andrew D. Rabinowitz,
Graphak II: Perspective projection an plotting program.
New York University Brons, New York, Agust 1969.

American Telephon and Telegraph Company,
Perspective drowing of surfaces with hidden lines elimination,
N.Y. Graham Feb., 18, 1972.
The bell system technical journal, vol. 51 no. 4. april 1972.

Kubert, Bruce R
A computer method for perspective representation of curves and
surfaces.
Technical report no. Tr - 8200-2 Aerospace corporation (dic.
1980)

Per Brich Hansen,
Operating Systems
Printence - Hall

Donald Gregory
A primer algol vol. I.
Bourroughs co.

Donald Gregory,
A primer algol vol. II.
Bourrougs co.

Input Output Subsystem reference manual,
Burroughs, co.

Fotran reference manual,
Burroughs, co.