

UES
504
3866
996
ET-2

UNIVERSIDAD DE EL SALVADOR
FACULTAD DE INGENIERIA Y ARQUITECTURA
ESCUELA DE INGENIERIA ELECTRICA



TRABAJO DE GRADUACION

**"SISTEMA DE ADQUISICION Y TRATAMIENTO DIGITAL DE
IMAGENES METEOROLOGICAS"**

TRABAJO DE GRADUACION PRESENTADO POR:

CARLOS EUGENIO MARTINEZ CRUZ

15101324

15101324

PARA OPTAR AL TITULO DE:

INGENIERO ELECTRICISTA

ENERO, 1996



SAN SALVADOR,

EL SALVADOR,

CENTRO AMERICA.

UNIVERSIDAD DE EL SALVADOR

RECTOR :

DR. BENJAMÍN LÓPEZ GUILLÉN

SECRETARIO GENERAL:

LIC. ENNIO LUNA

FACULTAD DE INGENIERÍA Y ARQUITECTURA

DECANO:

ING. JOAQUIN ALBERTO VANEGAS AGUILAR

SECRETARIO:

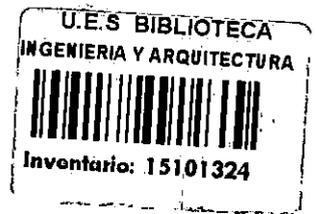
ING. JOSÉ RIGOBERTO MURILLO CAMPOS

ESCUELA DE INGENIERÍA ELÉCTRICA

ESCUELA DE INGENIERÍA ELÉCTRICA
FACULTAD DE INGENIERÍA
Y ARQUITECTURA
Universidad de El Salvador

DIRECTOR :

ING. SALVADOR DE JESÚS GERMAN



UNIVERSIDAD DE EL SALVADOR
FACULTAD DE INGENIERIA Y ARQUITECTURA
ESCUELA DE INGENIERIA ELECTRICA

Trabajo de Graduación previo a la opción al grado de:

INGENIERO ELECTRICISTA

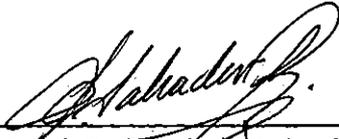
Titulo: " SISTEMA DE ADQUISICION Y TRATAMIENTO DIGITAL
DE IMAGENES METEOROLOGICAS "

Presentado por:

Carlos Eugenio Martínez Cruz

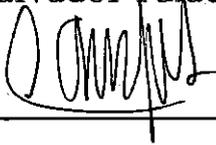
Trabajo de Graduación Aprobado por:

Coordinador y Asesor:


Ing. José Salvador Palacios



Asesor:


Lic. Leonardo Merlos Ventura.

ACTA DE CONSTANCIA DE NOTA Y DEFENSA FINAL

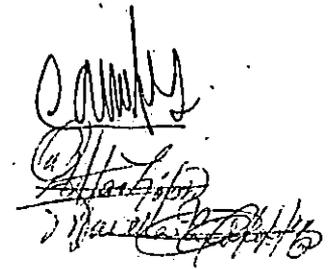
En esta fecha, 9 de Enero de 1995, en el local de Sala de Lectura de la Escuela de Ingeniería Eléctrica, a las 14:00 horas, con la presencia de las siguientes autoridades de la Escuela de Ingeniería Eléctrica de la Universidad de El Salvador:

- 1- Ing. Salvador de J. German
Director
- 2- Ing. Gerardo Marvin Jorge Hernández
Secretario



Y con el Honorable Jurado de evaluación integrado por las personas siguientes:

- 1- Lic. Leonardo Merlos Ventura
- 2- Ing. Ana Patricia Alfaro
- 3- Ing. Marta Lidia Merlos



Se efectuó la defensa final reglamentaria del Trabajo de Graduación:

"SISTEMA DE ADQUISICION Y TRATAMIENTO DIGITAL DE IMAGENES METEOROLOGICAS"

A cargo del Br.:

MARTÍNEZ CRUZ, CARLOS EUGENIO

Habiendo obtenido el presente trabajo una nota final, global de 10.0

(diez punto cero)

AGRADECIMIENTOS

A Dios por todo.

A mi padre, por sus consejos y por sembrar en mi un espíritu de lucha insistente.

A mi familia, por todo el apoyo que me han brindado.

A mis profesores, por compartir sus conocimientos.

A mis amigos, por estar en todo momento apoyandome.

A mis compañeros de Trabajo, por su colaboración.

A todos Gracias, por contribuir a este logro.

Carlos E.

DEDICATORIA

Dedico este trabajo, a la Memoria
de Tía Mercedes, quién con mucho sacrificio
supo orientarme por el camino que conduce
al éxito.

Carlos E.

LISTA DE TABLAS

Tablas	Página
3.1 Tamaño, dirección y número de páginas para los diferentes adaptadores de video	42
3.2 Módos de despliegue disponibles por los diferentes tipos de adaptadores de video	44
4.1 Elementos que forman la cabecera de los archivos PCX .	56
4.2 Número de colores a partir de los bits por pixel y el número de planos	58
4.3 Almacenamiento de los bytes con un sólo plano de color	59
4.4 Almacenamiento de los bytes con varios planos de color	60
4.5 Ejemplo de codificación según la técnica de los archivos PCX	60
4.6 Programa que despliega la información contenida en la cabecera de un archivo PCX	62
4.7 Lectura de la Cabecera de el archivo HG3.PCX	63
4.8 Estructura que contiene todos los elementos de la cabecera de un archivo PCX	64
4.9 Función que verifica la existencia de un archivo . . .	64
4.10 Función que lee la cabecera de un archivo PCX	65
4.11 Verifica la marca y versión de un archivo PCX	66
4.12 Función que despliega en pantalla el contenido de una cabecera PCX	67
4.13 Programa para leer un archivo PCX	68
4.14 Programa principal para leer y escribir archivos PCX .	69
5.1 Máscaras de convolución utilizadas para realizar filtros paso bajo a imágenes	76

5.2	Segmento de una imagen con baja frecuencia espacial .	77
5.3	Segmento de una imagen con alta frecuencia espacial .	77
5.4	Aplicación de filtros paso bajo (tabla 5.2)	78
5.5	Aplicación de filtros paso bajo (tabla 5.3)	78
5.6	Máscaras de convolución, para realizar filtrado a imágenes	79
5.7	Cortado y Pegado de Imágenes	80
5.8	Escalado de una imagen (zoom). Método de repetición	81

PREFACIO

Hasta la fecha no existían trabajos desarrollados en el área de Adquisición y Tratamiento Digital de Imágenes, porque hablar de adquisición de datos en nuestro país, hasta hace un par de años, era trabajo que solo podía realizarse por equipo comerciales provenientes del exterior, aún mas, el área de tratamiento Digital de señal empezó a desarrollarse en los E.U. en la década de los 60 y posteriormente como una especialización del Tratamiento Digital de Señales, surgió el Tratamiento Digital de imágenes; el desarrollo de estas áreas requerían de grandes centros de procesado de información lo que significaba que sólo podían ser desarrollados por grandes Universidades o centros de investigación con suficientes recursos económicos.

La reciente aparición de computadores como una herramienta asequible al gran público ha facilitado que en nuestro país se desarrollen los primeros trabajos en las áreas de adquisición y tratamiento digital de imágenes.

El procesado digital de imágenes es un área sumamente compleja, que involucra muchas disciplinas de la ingeniería y actualmente es un tema de interés en todos los departamentos de investigación de todo el mundo (exploración térmica del terreno, búsqueda de yacimientos, cuidado de cosechas, etc.)

El presente trabajo de graduación involucra dos áreas bien definidas, las cuales son: la adquisición de datos y el procesado digital de estos datos. Para realizar adquisición de datos es necesario comprender el funcionamiento de el periférico con el cual se realizara la comunicación periférico-computador, así como las diferentes técnicas empleada para realizar adquisición de datos.

Dentro de el tema procesado digital de imágenes se incluyen rutinas capaces de realizar cortado, pegado, ampliado, escalado y filtrado de imágenes. Dichas rutinas son de propósito general, es decir que dentro del trabajo se utilizan con imágenes meteorológicas pero esta colección de funciones puede ser utilizada en cualquier otra aplicación que requiera procesado de imágenes.

RESUMEN

El trabajo básicamente consiste en la construcción de una interfase y el desarrollo de un programa capaz de comunicarse con el equipo receptor de imágenes meteorológicas WSR513/515. El trabajo inicia con un estudio sobre operación y funcionamiento de el equipo receptor de imágenes, así como la teoría básica sobre satélites meteorológicos en general, la importancia de este tema se debe a que es necesario comprender el proceso que lleva la señal desde que es enviada por el satélite hasta que es convertida a imágenes en el computador.

El tema de Adquisición de datos merece mucha importancia en el desarrollo de este trabajo y es por eso que el tema se trata con cierto detalle, dando una clasificación general de las diferentes técnicas, y explicando la técnica por chequeo de estado, conocida como polling, que es la que se usa en este trabajo.

La necesidad de manipular imágenes, hace indispensable el conocer a fondo el hardware de el computador, es por eso que se incluye el tema de dispositivos entrada/salida y tecnologías gráficas, dando énfasis a los adaptadores de despliegue, y a la programación de las tarjetas gráficas EGA/VGA.

Cada imagen recibida es guardada en un archivo de tipo PCX, esto permite que cualquier programa comercial pueda visualizar dichas imágenes, la información completa de los archivos PCX, se incluye en este documento así como en los anexos B,C,y D.

Por ultimo se incluye el tema de procesado digital de imágenes, debido a que este tema es demasiado complejo y extenso, solo se aplican ciertos tópicos, dentro de los cuales destacan escalado, ampliado de zonas, mapeado de contornos, filtrado de imágenes, etc. Este trabajo incluye anexos en los cuales se muestran las rutinas que realizan la adquisición y el procesado digital de dichas imágenes

TABLA DE CONTENIDOS

LISTA DE TABLAS
LISTA DE FIGURAS
LISTA DE ACRONIMOS Y ABREVIACIONES

I. SATÉLITES METEOROLÓGICOS Y FUNCIONAMIENTO DE EL EQUIPO RECEPTOR DE IMÁGENES METEOROLÓGICAS WSR513/515

INTRODUCCIÓN	1
1.1 Satélites Meteorológicos	1
1.2 Selección de el satélite	2
1.3 Satélites Meteorológicos explorados por Estados Unidos	2
1.3.1 Situación de Satélites Geoestacionarios	3
1.4 Receptor de satélite WSR513	4
1.5 Receptor de satélite WSR515	5
1.6 Descripción funcional de el WSR513	6
1.6.1 Antena	7
1.6.2 Unidad electrónica.	8
1.6.2.1 Modulo Receptor.	8
1.6.2.2 Modulo Decodificador.	8
1.6.2.3 Unidad de Control y Memoria	9
1.7 Descripción de el modulo decodificador.	9
1.7.1 Recuperación de el reloj	9
1.7.2 Demodulación y digitalización.	10
1.7.3 Señales de sincronización	12
1.7.3.1 Señales de sincronización de satélites de órbita polar	12
1.7.3.2 Sincronización de satélites de órbita geoestacionaria	13
1.8 Transmisiones Satelitales APT y WEFAX	14
1.8.1 Transmisión Whether Facsímil	14
1.8.2 Historia del WEFAX	16
1.8.3 Productos wefax y fuente de datos.	16
1.8.4 Imágenes GOES VISSR	16
CONCLUSIONES DEL CAPITULO I	18
REFERENCIAS BIBLIOGRAFICAS	20

II. ADQUISICIÓN DE DATOS Y DISEÑO DE LA INTERFASE

INTRODUCCIÓN	22
2.1. Técnicas de transferencia de Entrada/Salida.	22
2.2. Coordinación de las transferencias de Entrada/Salida	23
2.3. Clasificación de las técnicas de transferencia	24
2.4. Transferencia por consulta de estado	25
2.4.1 Puerto Serie	26
2.4.2 Puerto paralelo	27
2.5. Tarjeta de entrada/salida UN-1072	30
2.6. Diseño de la interfase de adquisición de datos	33
CONCLUSIONES DEL CAPITULO II	35
REFERENCIAS BIBLIOGRAFICAS	36

III. DISPOSITIVOS DE ENTRADA/SALIDA Y TECNOLOGÍAS GRÁFICAS

INTRODUCCIÓN	38
3.1 Llevando una imagen a la pantalla	38
3.2 Modos de video	41
3.2.1 Resolución de video	41
3.2.2 El sistema de despliegue (Controladores de video)	42
3.3 Guardando y Desplegando datos en video.	42
3.4 Formas de desplegar video.	44
3.5 Bits y color	45
3.6 Gráficos.	46
3.7 Modelo de barrido vrs. modelos vectoriales	47
3.7.1 Mapa de bits	47
3.7.2 Gráficos basados en objetos	48
3.8 Formatos Gráficos	49
3.8.1 Formatos en mapas de bits.	50
3.8.1.1 TIFF	50
3.8.1.2 PCX	50
3.8.1.3 FAX	51
3.8.2 Formatos basados en objetos	51
3.8.2.1 PostScript	51
3.8.2.2 CGM	52
3.8.3 Otros formatos vectoriales	52
CONCLUSIONES DE EL CAPITULO III	53
REFERENCIAS BIBLIOGRAFICAS	54

IV. FORMATO DE ARCHIVO PARA IMÁGENES PCX

INTRODUCCIÓN	56
4.1. Estructura de Cabecera.	56
4.2. Detalle de cada uno de los elementos que forman la cabecera del archivo PCX	57
4.3. Estructura de los datos.	60
4.4. Técnicas de Codificación Run-length orientada a bytes	61

4.5	Técnicas para decodificar archivos PCX	
4.6	La utilidad PCX.H	64
4.7	Leyendo una Imagen PCX	68
4.8	Escribiendo un archivo PCX.	68
4.9	Leyendo y Escribiendo Imágenes Meteorológicas	69
CONCLUSIONES DEL CAPITULO IV		72
REFERENCIAS BIBLIOGRAFICAS		73

V. INTRODUCCIÓN AL PROCESADO DIGITAL DE IMÁGENES.

INTRODUCCION	75	
5.1 Frecuencia espacial, y filtrado de imágenes	75	
5.1.1 Frecuencia Espacial	75	
5.1.2 Filtrado.	77	
5.2 Aplicación de filtrado a imágenes.	77	
5.3 Frecuencia vrs. Filtrado espacial	77	
5.3.1 Filtros paso bajo	78	
5.3.2 Implementación de un filtro paso bajo	80	
5.3.3 Efectos de los filtros paso bajo en las imágenes Meteorológicas	82	
5.4 Operaciones con imágenes	87	
5.4.1 Cortado y Pegado	87	
5.4.2 Escalado de Imágenes (Zoom)	87	
5.5 Visualización simultanea de varias imágenes	89	
5.6 Animación de imágenes meteorológicas	91	
CONCLUSIONES DEL CAPITULO V		92
REFERENCIAS BIBLIOGRAFICAS		93

CONCLUSIONES GENERALES	94
RECOMENDACIONES	95

ANEXOS

ANEXO A	Guía de usuario de el programa de adquisición y tratamiento digital de imágenes meteorológicas (SAT) y animación de imágenes meteorológicas (ANIMATE)	97
ANEXO B	Listado de el archivo cabecera PCX.H	102
ANEXO C	Listado de el archivo RPCX.C	106
ANEXO D	Listado de el archivo WPCX.C	109

LISTA DE FIGURAS

Figura	Página
1.1 Mapa de cobertura global de satelites geoestacionarios	4
1.2 Panel frontal de el receptor de imágenes meteorológicas WSR513	5
1.3 Panel frontal de el receptor de imágenes meteorológicas WSR515	6
1.4 Equipo completo de recepción de imagnes meteorologicas WSR513/515	7
1.5 Detección usando contadores ascendente/descendente . .	11
1.6 Formato de una línea de video de una APT	13
1.7 Señales de sincronía A y B	14
1.8 Señales de protocolo, para satélites geoestacionarios	15
1.9 Flujo de datos GOES	17
1.10 Formato de una imagen WEFAX	18
2.1 Esquema que resume el concepto de interfase	23
2.2 Diferentes tipos de Entrada/Salida	26
2.3 Comparación de velocidades de puertos utilizados en la adquisición de datos	27
2.4 Conector paralelo de 25 pines	28
2.5 Representación en forma de registro del puerto paralelo	30
2.6 Tarjeta IDE I/O UN-1072	31
2.7 Diseño de la interfase de adquisición de datos	33
3.1 Diseño básico de un CRT	38
3.2 Principio de un CRT a color	38
3.3 Monitor de rastreo	39

4.1	Imagen meteorológica cuyo formato es PCX	71
5.1	Muestra de una imagen co baja frecuencia espacial . .	76
5.2	Imagen con alta frecuencia espacial	76
5.3	Imagen meteorológica (con ruido)del tipo infrarrojo, recibida el 12 de Diciembre de 1992	83
5.4	Resultado de aplicar filtro paso bajo mascara #6 a la figura 5.3	84
5.5	Resultado de aplicar filtro paso bajo mascara #9 a la figura 5.3	85
5.6	Resultado de aplicar filtro paso bajo mascara #32 a la figura 5.3	86
5.7	Ampliación de una sección de la figura 5.3. Técnica de repetición	88
5.8	Visualización simultánea de dos imágenes meteorológicas	89
5.9	Visualización simultánea de cuatro imágenes meteorológicas	90

LISTA DE ACRONIMOS Y ABREVIACIONES

AM	Amplitud Modulada
APT	Transmisión automática de Imagenes
ATS	Satélites de Aplicación Tecnológica
AVHRR	Radiómetros Avanzados de Muy Alta Resolución
CDA	Comandos y Adquisición de Datos
CGA	Color Graphics Adapter
CGMS	Coordinación sobre Satelites Meteorológicos Geoestacionarios
CRT	Tubo de rayos catódicos
dB	decibel
dBm	decibel referido a un milivatio
DACS	Subsistema de Control y Adquisición de Datos
DCS	Sistema de Colección de Datos
EGA	Enhanced Graphics Adapter
ESA	Agencia Espacial Europea
FET	Transmisor decodificador de facsímile
FM	Modulación en frecuencia
GOES	Satélite de Ambiente Operativo Geoestacionario
HGA	Hercules Graphics Adapter
Hz	Hertz
MCGA	Multicolor Graphics Array
MDA	Monochrome Display Adapter
MHz	Megahertz
NOAA	Administración Nacional de la Atmósfera y el Océano
VGA	Virtual Graphics Array
VISSR	Radiómetro Visible Infrarrojo de Barrido Rotación
WEFAX	Wether Facsímile

CAPITULO I

SATÉLITES METEOROLÓGICOS Y FUNCIONAMIENTO DE EL EQUIPO RECEPTOR DE IMAGENES METEOROLÓGICAS WSR513/515.

Satélites Meteorológicos y funcionamiento de el Equipo Receptor de Imágenes

INTRODUCCION

En este capítulo se pretende describir el funcionamiento de el receptor de imágenes meteorológicas WSR513/15 y la forma en que las señales son recibidas y procesadas por el equipo. El capítulo ofrece una breve explicación de los tipos de satélites meteorológicos mas importantes, explicando a la vez los formatos de las imágenes ya sean estas de satélites polares o geoestacionarios, que se utilizan en la comunicación satélite-estación y en detalle se explica el formato WEFAX, propio de los satélites geoestacionarios y desarrollado por los Estados Unidos.

Se describe en detalle el funcionamiento y capacidad de el equipo receptor y se analiza cada uno de los módulos que constituyen el equipo.

1.1 Satélites Meteorológicos

La función de un satélite meteorológico es observar libremente la tierra desde el espacio, y enviar imágenes a la tierra de como se ve ella misma. Hay dos tipos de satélites meteorológicos, clasificados según su órbita:

1. Polar. Los satélites de órbita polar viaja a una altura alrededor de los de los 850 Km. pasando cerca de los polos norte y sur en cada órbita.

2. Geoestacionario. La órbita de un satélite geoestacionario es circular y ecuatorial. Así pues para estos satélites la altitud es de 36,000 Km., y el período es exactamente de un día y por eso mantienen una posición relativamente constante a la tierra, por lo que son visibles permanentemente desde las estaciones que enlazan con ellos.

Una vez que el satélite esta en órbita, su utilidad se da cuando este es capaz de tomar imágenes de la tierra. A diferencia de una cámara fotográfica normal el satélite no toma "instantáneas" si no que realiza un "barrido" continuo de la tierra, este barrido se realiza de izquierda a derecha en línea, directamente desde abajo. El dispositivo de barrido es llamado un radiometro de barrido.

Satélites Meteorológicos y funcionamiento de el Equipo Receptor de Imágenes

La información colectada por el satélite es inútil a menos que pueda ser enviada a la tierra. El método mas simple es a través de las ondas de radio, esta señal es enviada a la tierra para ser usada por todas las estaciones terrenas incluyendo el equipo receptor de imágenes meteorológicas WSR513. Este proceso es continuo y la información es enviada línea a línea, este método de enviar imágenes es llamado Transmisión Automática de Imágenes o APT.

1.2 Selección de el satélite

Una de las principales preguntas que surgen es de cual satélite o satélites podemos recibir imágenes, esto depende de la localización de nuestro equipo receptor de imágenes. Un mapa de cobertura global se muestra en la figura 1.1, donde se muestra claramente la situación geográfica, cada semicírculo muestra el área dentro de la cual la imagen es recibida con un mínimo de distorsión (esta distorsión es debido a la curvatura de la tierra).

En algunas áreas es posible recibir transmisiones desde dos satélites. Cambiar de uno a otro es posible únicamente orientando la antena y si es necesario también se debe cambiar la polarización de la antena.

1.3 Satélites Meteorológicos explorados por los Estados Unidos

(a) Cuando funcionan dos satélites GOES, estos se ubican a 75 °W (GOES-Posición este) y a 135 °w (GOES- Posición oeste) con carácter permanente. Desde marzo de 1995, el GOES-I (llamado actualmente GOES-8) pasó a ser el nuevo GOES-East y el GOES-7 fue desplazado a la posición del GOES-West. En la figura 1.1 se observa la cobertura lograda desde estas posiciones nominales.

GOES-8 es el primero de la próxima generación de satélites y sus capacidades han mejorado mucho con respecto al GOES-7; por ejemplo, tiene mejores resoluciones en todos los modos. La resolución de infrarrojo (IR) de 4 Km permitirá un mejor seguimiento y detección de tormentas fuertes/inundaciones repentinas. Las sondas (resolución de 8 Km) funcionan permanentemente en siete canales suplementarios, capaces de

Satélites Meteorológicos y funcionamiento de el Equipo Receptor de Imágenes

efectuar 2500 sondeos en una hora. A diferencia del GOES-7, la obtención de imágenes y los sondeos se realiza simultáneamente. Gracias a su capacidad de barrido rápido puede obtener "pequeñas imágenes" de tormentas fuertes cada cinco minutos, si se requiere.

Todas las imágenes se transmiten en tiempo real a las estaciones del servicio satelital sobre el terreno del Servicio Meteorológico Estado Unidense, al departamento de análisis de datos satelitales del servicio nacional de información sobre el medio ambiente mediante datos satelitales (NESDIS), y a las oficinas de predicción del servicio meteorológico.

(b) Los Satélites de Órbita Polar de la NOAA (Administración Nacional del Océano y de la Atmósfera) proporcionan, dos veces diarias, imágenes mundiales de la banda visible y de IR de una resolución de 0,8 Km (0,5 millas) que se reciben, procesan y difunden centralmente.

1.3.1 Situación de Satélites Geoestacionarios

GOES-8:

Primer satélite de la serie GOES-NEXT, lanzado el 13 de abril de 1994. Operativo el 1 de marzo de 1995 a 75 °W, en la posición GOES-Este; es ahora el principal satélite operativo.

GOES-7:

Era el principal satélite operativo. El GOES-7 se ha pasado a 135 °W (posición GOES-Oeste) por lo que hay dos satélites operativos GOES en sus posiciones nominales por primera vez desde enero de 1989. El funcionamiento del GOES-7 se deteriora y se acerca al final de su vida útil.

Satélites Meteorológicos y funcionamiento de el Equipo Receptor de Imágenes

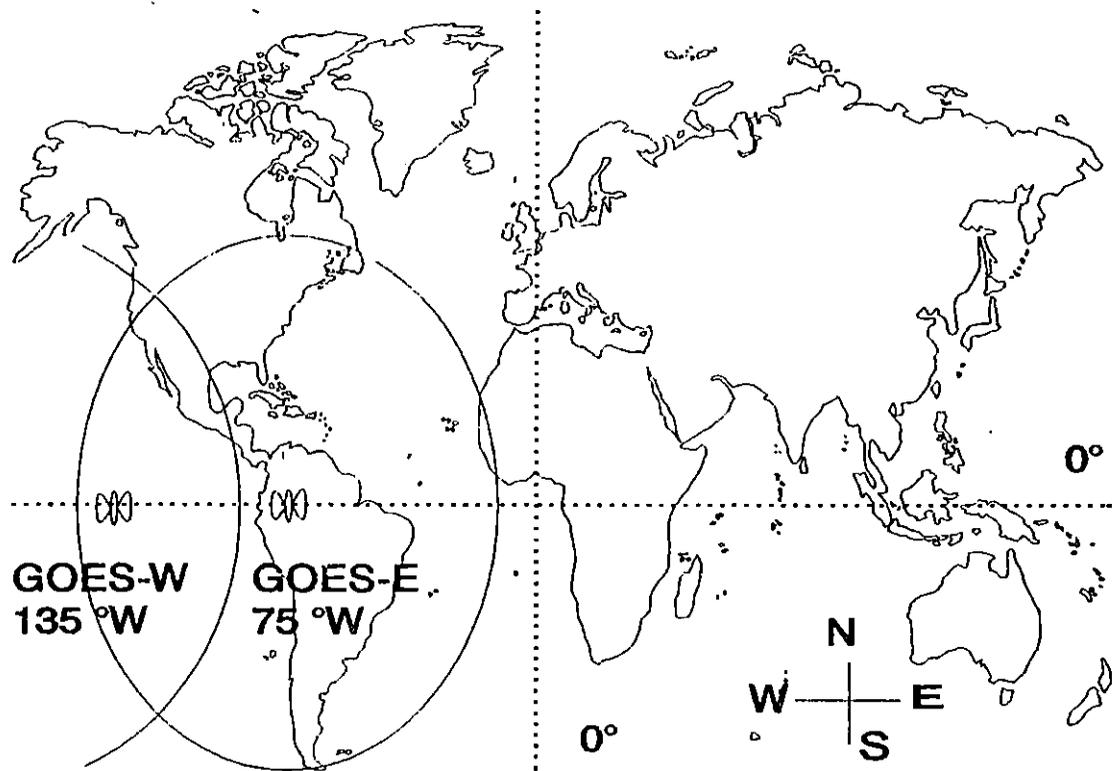


Figura 1.1 Mapa de Cobertura Global de Satélites Geostacionarios.

1.4 Receptor de satélite WSR513

El WSR513 está diseñado para recibir APT de la serie de satélites estadounidenses TIROS que es una serie de satélites meteorológicos de órbita polar operados por la Administración Nacional de la Atmósfera y Océanos (NOAA) y la serie de satélites METEOR operados por la URSS.

Con el equipo receptor de satélite WSR513 básico se pueden recibir señales de VHF cuyo formato sea APT, tales señales son generadas por los satélites de órbita polar.

Las imágenes se pueden recibir en el espectro visual como en el infrarrojo esto es según se haya seleccionado desde el panel frontal, también se le puede dar un mínima capacidad de edición a las imágenes recibidas, es decir se puede mejorar la resolución de la imagen seleccionando el número de líneas por segundo que se desean recibir y/o se puede cambiar el número de pixel por línea de la imagen, existe una escala de grises

Satélites Meteorológicos y funcionamiento de el Equipo Receptor de Imágenes

configurables manualmente, así como una escala de tonalidades de color que mejoran la presentación de la imagen, ya que en algunos caso se puede ocultar importante información tal es el caso de las imágenes infrarrojas. El panel frontal se muestra en la figura 1.2.

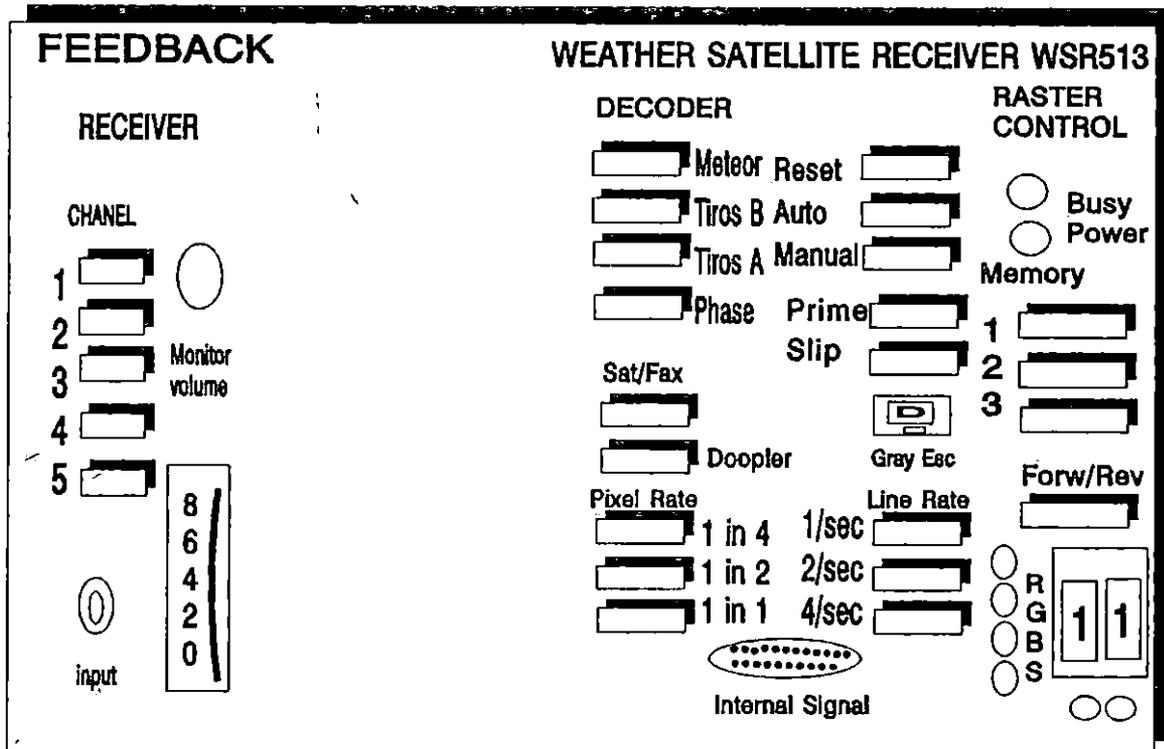


Figura 1.2 Panel Frontal de el Receptor de Imágenes Meteorológicas WSR513.

1.5 Receptor de satélite WSR515

El propósito de el WSR515 S-BAND es mejorar las funciones de el receptor básico de tal manera que pueda recibir señales de S-BAND (que es la frecuencia a la que transmiten los satélites geostacionarios) y convertirlas a señales VHF, manteniendo la capacidad de el WSR513 de recibir señales de VHF y cualquiera de los dos tipos de señales puede ser seleccionada por un interruptor que se encuentra sobre el WSR515, lo que en otras palabras significa que desde el equipo seleccionamos manualmente el satélite de el cual deseamos recibir las imágenes

Satélites Meteorológicos y funcionamiento de el Equipo Receptor de Imágenes

meteorológicas.

Las transmisiones en S-BAND recibidas por el WSR515 son señales analógicas descritas como facsímil meteorológicas o WEFAX. Esta difieren de APT(que es el formato para la recepción de satélites de órbita polar) principalmente en que WEFAX es transmitido en cuadros, mientras que la transmisión APT es continua.

El receptor de Imágenes Meteorológicas WSR513/515 es capaz de recibir imágenes de los satélites geoestacionarios Estadounidenses GOES, Europeos METEOSAT, Japones GMS, y Rusos GOMS. Es de tener claro que solo se es capaz de recibir señal de un solo satélite esto es debido a que cada uno de estos satélites tienen determinada su zona de cobertura.

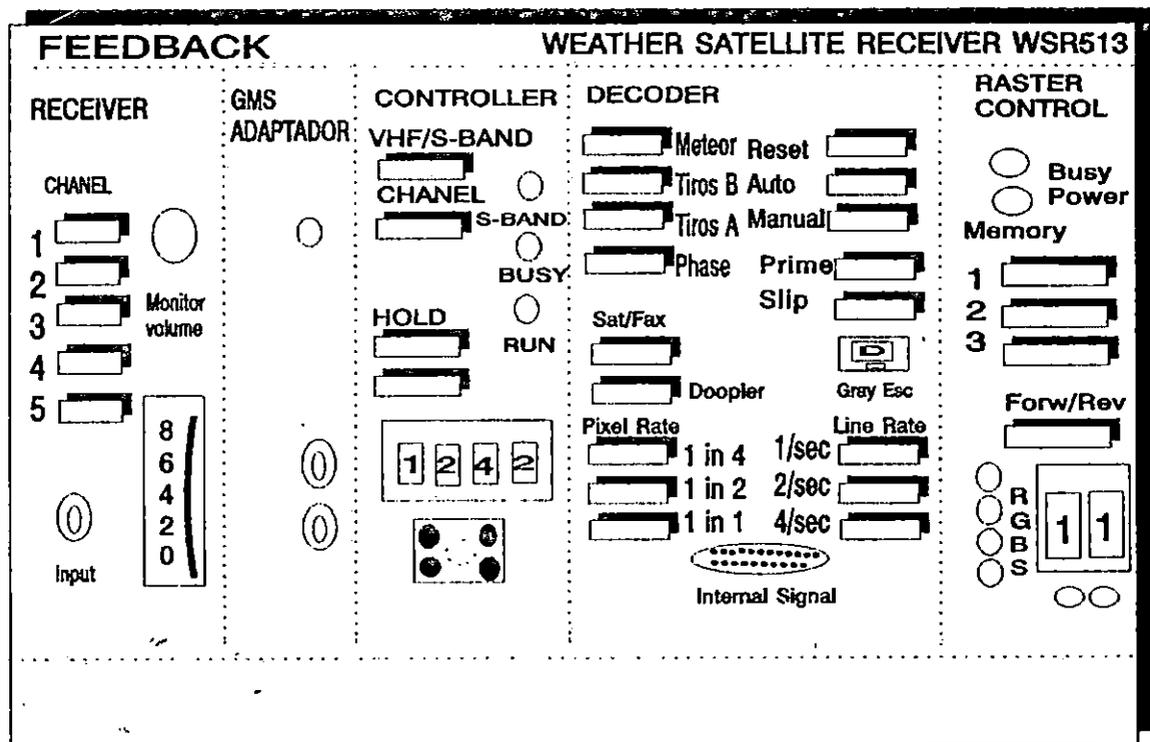


Figura 1.3 Panel Frontal de el Receptor de Imágenes Meteorológicas WSR515

Satélites Meteorológicos y funcionamiento de el Equipo Receptor de Imágenes

1.6 Descripción funcional de el WSR513

El receptor de satélite WSR513 se puede dividir en tres etapas claramente diferenciadas:

1. Una unidad de antena a ser montada en una azotea o sobre un edificio.
2. Una unidad electrónica la cual puede ser montada sobre un escritorio.
3. Un monitor a color para desplegar las imágenes.

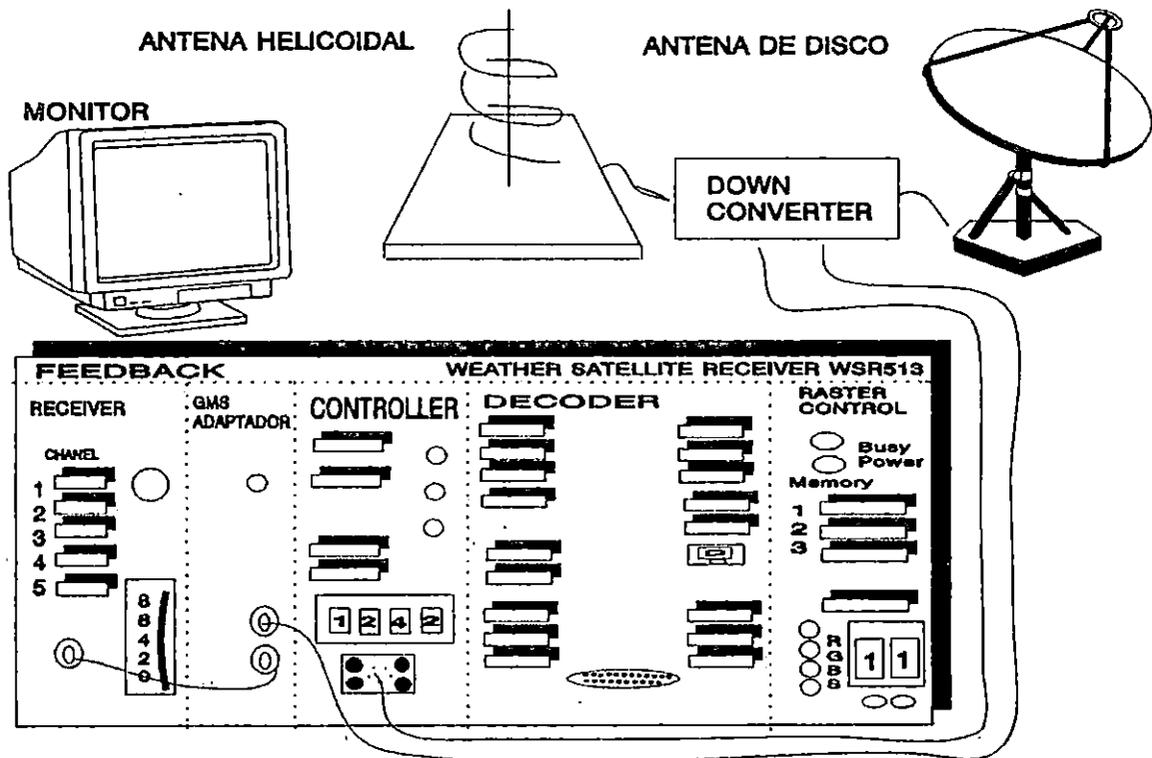


Figura 1.4 Equipo Completo de Recepción de Imágenes Meteorológicas WSR513/515

1.6.1 Antena.

Existe dos antenas, una parabólica de disco para S-BAND y una antena helicoidal para recepción de VHF, cada una de estas

Satélites Meteorológicos y funcionamiento de el Equipo Receptor de Imágenes

antenas tiene su propia amplificador de cabecera, cuya salida son conectados dentro de la unidad convertidora, esta unidad contiene un autoconvertidor el cual convierte las señales de S-BAND a señales de VHF :

la antena helicoidal esta montada sobre un plano de tierra. Esta antena tiene suficiente ganancia sobre un arco ancho de ciclo, lo que permite recibir una mejor calidad de la señal. El centro de la frecuencia para esta antena es 137.3 Mhz .

Por que el receptor de señal es bastante malo, un amplificador de bajo ruido es incluido en la antena, para amplificar la señal antes que sea degradada. Entre la antena y el amplificador hay un filtro para remover interferencia.

1.6.2 Unidad electrónica.

Esta unidad contiene muchas tarjetas de circuitos impresos, cada una de las cuales tiene su propio panel.

La potencia suministrada a las unidades se localiza en la parte posterior de el equipo. Hay muchos espacios vacíos en el equipo lo cual permite expandir las funciones de el equipo.

El cable desde la antena alimenta directamente el modulo receptor, a continuación se detalla la función de los módulos por los cuales esta formada la unidad Electrónica.

1.6.2.1 Modulo Receptor.

Este es un receptor de 5 canales de VHF el cual amplifica y demodula la señal de el satélite al seleccionar el canal a recibir a través de los botones 1 a 5 . Estos canales son:

1. METEOR Experimental.
2. METEOR 2.
3. TIROS descendente y ascendente.
4. TIROS ascendente.
5. METEOR-2.

1.6.2.2 Modulo Decodificador.

Esta unidad usa los bordes de sincronización de la señal para proveer pulsos de inicio de línea desde su propio reloj

Satélites Meteorológicos y funcionamiento de el Equipo Receptor de Imágenes

interno. El tipo de imagen y la resolución de la imagen final es controlada por interruptores que se encuentran en la parte frontal de el equipo. También la señal analógica es digitalizada y lista para ser usada por la unidad de control y memoria.

1.6.2.3 Unidad de Control y Memoria

Esta unidad convierte los datos de baja velocidad a mas alta velocidad para representar estos datos en un monitor normalizador de televisión. Los datos son recibidos línea por línea desde el fondo ó el final de el monitor. Todas las líneas existentes son desplazadas hacia arriba ó hacia abajo según se haya seleccionado.

Las líneas desplazadas fuera de la pantalla se pierden. La unidad de memoria y control tiene tres separadas memorias de tal manera que tres imágenes pueden ser guardadas. La memoria activa es seleccionada desde el frente de el panel de la unidad de control.

1.7 Descripción de el modulo decodificador.

Las señales que se reciben desde el satélite a través de una antena pasan por un preamplificador y luego esta señal pasa a la unidad de recepción, la cual convierte la señal de VHF a una señal modulada en amplitud cuya frecuencia es de 2400 Hz, esta señal pasa al modulo de decodificación.

La operación de el decodificador puede dividirse en cuatro secciones principales:

1. Recuperación de el reloj.
2. Demodulación y digitalización.
3. Decodificación de señal de sincronía.
4. Generación de la señal de inicio de línea.

1.7.1 Recuperación de el reloj.

El circuito que recupera el reloj usa un PLL para generar una señal de reloj de dos fases la cual es continua e independiente de la señal modulada en amplitud. La señal de reloj es disponible en una de las líneas que llegan a el monitor y

Satélites Meteorológicos y funcionamiento de el Equipo Receptor de Imágenes

también es disponible en el conector de salida.

1.7.2 Demodulación y digitalización.

El proceso de demodular y digitalizar es como sigue: la señal analógica modulada en amplitud es amplificada y luego pasa a un oscilador controlado por voltaje ó VCO, obteniendo a la salida una señal de frecuencia variable, esta señal es aplicada a un contador ascendente/descendente. El contador cuenta ascendentemente durante la primera mitad de el ciclo y descendentemente durante la segunda mitad de el ciclo (véase figura 1.5) al finalizar la primera mitad de el ciclo el contador tendrá un numero proporcional a el área 'A' el contador se comporta semejante a un integrador, durante la segunda mitad de el ciclo el contador cuenta descendentemente y sustrae un numero proporcional a el área 'B'. Por lo tanto al final de el primer ciclo el contador tendrá un valor proporcional a el área B. por lo tanto al final de el primer ciclo el contador tendrá un valor proporcional a el área A menos el área B, el cual a su vez es proporcional a el valor pico a pico de el ciclo completo, este valor corresponde a el valor de un elemento de imagen(pixel). La operación se repite para el segundo ciclo y para los siguientes.

Para prevenir la posibilidad de un subflujo(es decir un valor mas pequeño que el mínimo aceptado también conocido como 'undeflow') en presencia de ruido o señales de baja amplitud; el contador no se reinicializa a cero si no que se pone a un numero pequeño. Es necesario ser capaz de producir un total representando uno, dos o cuatro ciclos. Esto se logra permitiéndole a el contador continuar contando hacia arriba y hacia abajo sin llegar a ser reinicializado, este proceso se lleva acabo durante un numero apropiado de ciclos de tal manera que el valor final digital en el contador representa la amplitud promedio de el apropiado numero de ciclos.

El control de el numero de ondas moduladas en amplitud a promediar se hace a través de la tecla pixel nominal que se encuentra en el panel receptor .

Para prevenir errores todas las señales de control son sincronizadas con la salida de un oscilador controlado por voltaje VCO. El equipo genera una señal de control llamada DINSTR la cual es puesta en el estado alto activo por un pulso de captura en memoria(Latch) y es baja activa para el próximo pulso de reloj para indicar que un nuevo dato a sido completado. El

Satélites Meteorológicos y funcionamiento de el Equipo Receptor de Imágenes

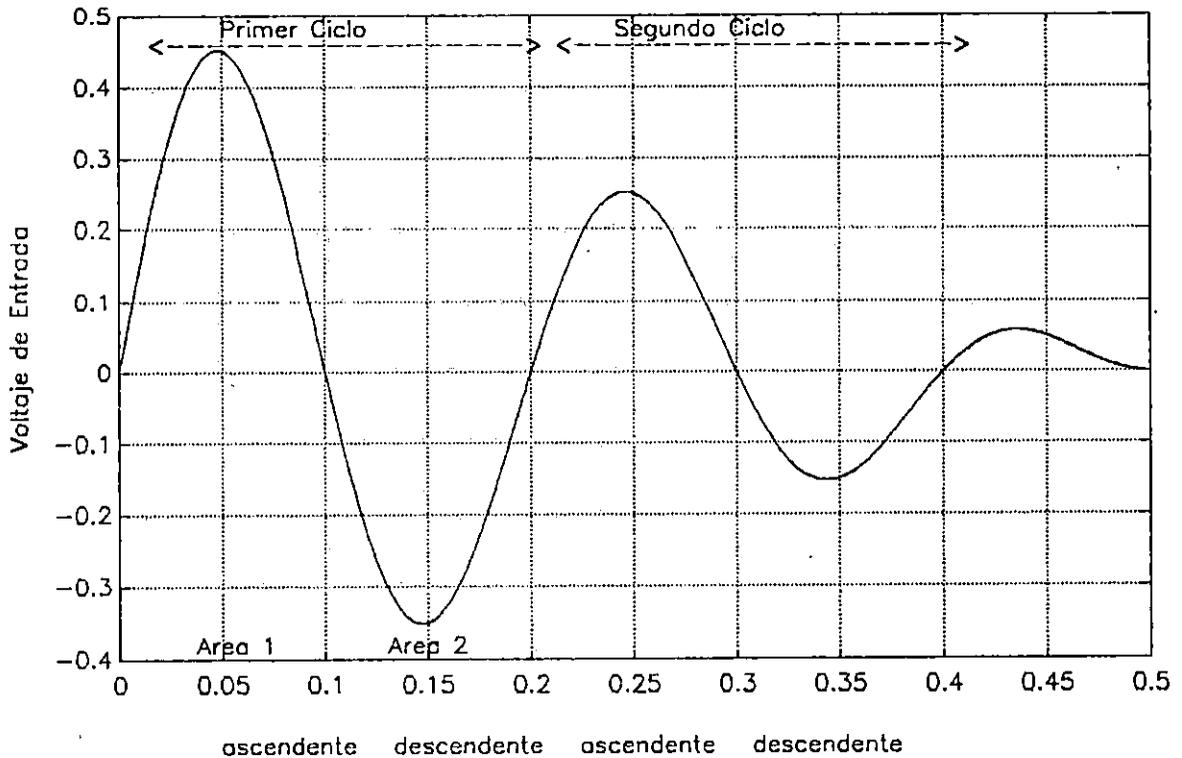


Figura 1. 5 Detección usando Contadores ascendente/descendente

borde activo es bajo, por lo tanto en ese instante toda la línea de datos deberían tener estado estable, la señal DINSTR se encuentra disponible en el conector de salida.

La digitalización de la señal es a 6 bit y es disponible en el conector de salida como los pines 6W0 a 6W5 después de ser convertidos a niveles TTL estos seis bit son pasados internamente a una EPROM, esta EPROM tiene a su salida solo 4 bits, lo que significa que comprimimos la información de 6 a 4 bit.

La memoria ROM tiene un programa capaz de cambiar el tono de la imagen acorde a una escala de grises configurables manualmente la escala particular a seleccionar es controlada por los cuatro bit mas significantes los cuales son provistos por un interruptor hexadecimal que se encuentra en el frente de el equipo receptor de imágenes WSR513.

Satélites Meteorológicos y funcionamiento de el Equipo Receptor de Imágenes

1.7.3 Señales de sincronización

Las imágenes son enviadas desde el satélite como una sucesión de líneas, cada una de las cuales es una secuencia de elementos de imágenes(pixel). Idealmente cada pixel debería ser guardado en una localización específica la cual controlara la calidad de la imagen como parte correspondiente de la imagen presentada.

Para evitar situaciones de Lecturas incorrectas de cada línea que forma la imagen, cada línea debe empezar en el momento correcto en relación a la transmisión recibida. La generación de inicio de línea marca con un pulso el comienzo de cada nueva línea, el número de líneas por segundo se puede seleccionar desde los interruptores que están en la parte frontal de el panel la velocidad mas rápida es de 4 líneas por segundo. Seleccionando otra velocidad la imagen puede deslizarse a 2 líneas por segundo y a 1 línea por segundo.

1.7.3.1 Señales de sincronización de satélites de órbita polar

Así para la serie de satélites TIROS hay dos tipos de señales de sincronía, cada una habilita el canal apropiado de video, ya que esta serie de satélites transmite dos tipos de imágenes: en el espectro visual y en el espectro infrarrojo.

Cada una de los tipos de imágenes posee su propia señales de sincronía, la forma de estas señales es como se muestra en la figura 1.6. Note que la señal de sincronía es 39 veces el período (T). El período T es el período de una palabra. Una línea completa esta formada por 2080 palabras de largo y distribuidas como se muestra :

Sincronía A	39
Espacio para datos/Marca de minutos	47
Datos de video para el canal A	909
Telemetría	45
Sincronía B	39
Espacio para datos/Marca de minutos	47
Datos de video canal B	909
Telemetría	45

Satélites Meteorológicos y funcionamiento de el Equipo Receptor de Imágenes

La sincronía A tiene su frecuencia fundamental de 1040 Hz mientras que la sincronía B tiene una frecuencia de 832 Hz.

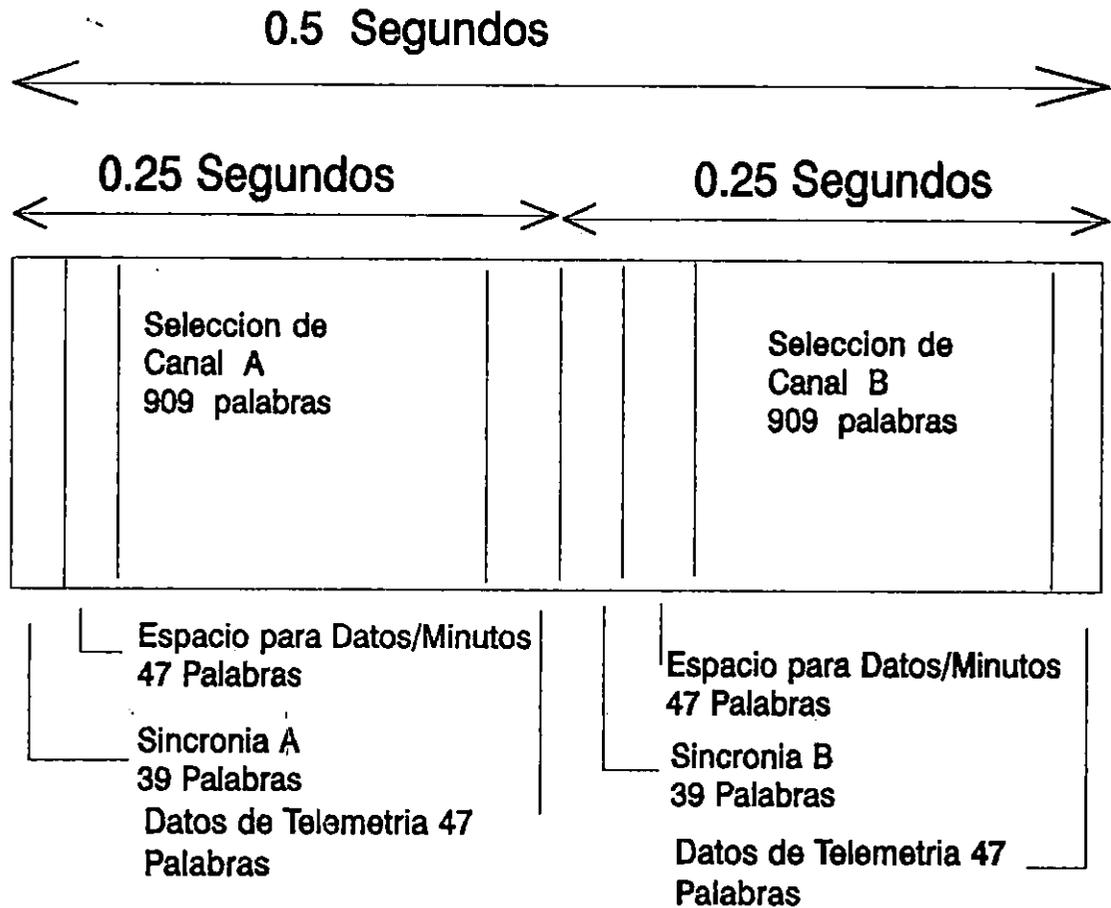


Figura 1.6 Formato de una línea de video de una APT

1.7.3.2 Sincronización de satélites de órbita geoestacionaria

Para la serie de satélites de órbita geoestacionaria GOES todos los controles asociados normalmente con la operación de el equipo receptor básico WSR513 son los mismos, excepto que en esta banda el grupo de botones de sincronía no tiene efecto, debido a que el formato de las imágenes es continuo.

Satélites Meteorológicos y funcionamiento de el Equipo Receptor de Imágenes

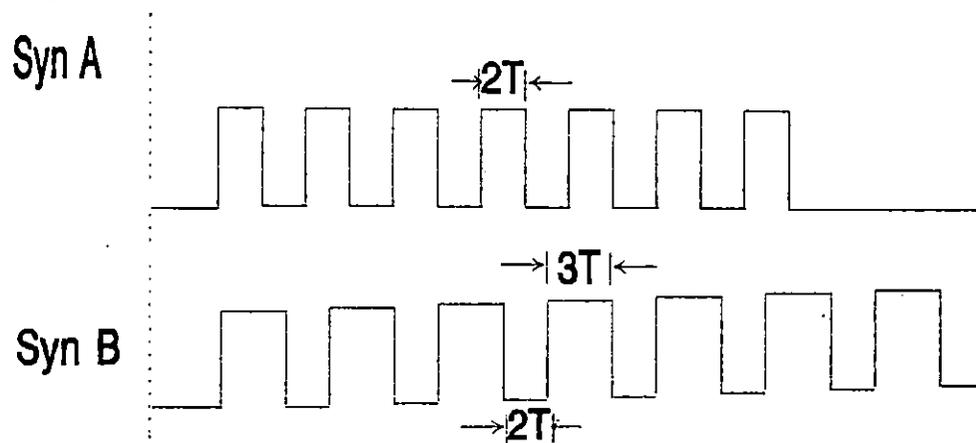


Figura 1.7 Señales de Sincronía A y B

- Notas:
- (1) $T=1/4160$
 - (2) Sincronía A precede el Canal A de datos
 - (3) Sincronía B precede el Canal B de datos

La recepción de imágenes es inicializada oprimiendo la tecla "operate/hold", la recepción da inicio después de un tono que indica el inicio de una imagen. La sincronización de la imagen toma lugar justo después de el inicio de el tono, los indicadores 'lock' y 'run' se activan casi simultáneamente durante la recepción de una imagen. El indicador 'busy' ubicado en el modulo de control se estará activando cada vez que una línea de la imagen de inicio, este efecto puede verse en el panel frontal de el equipo como un encendido/ apagado de el indicador 'busy'.

Los datos son colocados en tiempo real en el conector de salida, por un reloj de 2400 Hz; la señal que indica cuando existe un dato valido es el DINSTR (se encuentra en el pin 12c de el conector de salida), cuyo borde ascendente indica la estabilidad de la línea de datos y estos pueden ser transferidos.

El inicio de una nueva línea de la imagen es señalada por el borde descendente de un pulso angosto sobre el pin 7c esta señal se le llama LWB (inicio de línea a escritura), cuyo valores son 12 o 0 voltios, las señales DINSTR y LWB se muestra en la figura 1.8.

Satélites Meteorológicos y funcionamiento de el Equipo Receptor de Imágenes

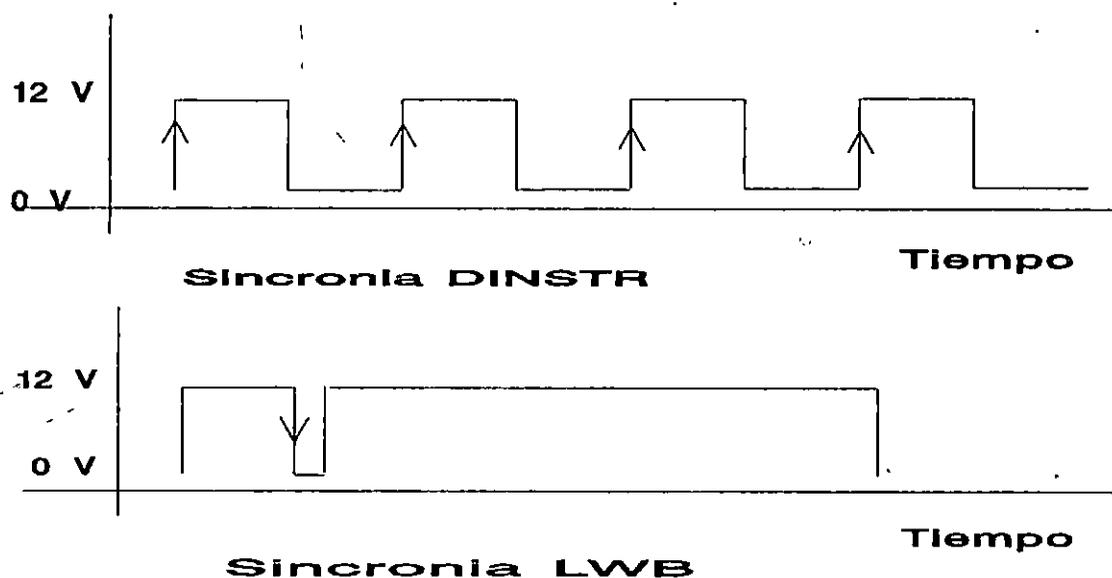


Figura 1.8 Señales de Protocolo, para satélites geoestacionarios

1.8 Transmisiones Satelitales APT y WEFAX

Las imágenes enviadas desde los Satélites Meteorológicos pueden ser de dos tipos WEFAX (Whether Facsímile) para satélites geoestacionarios, y APT (Transmisión Automática de Imágenes) para satélites de Órbita Polar. Cabe aclarar que en el presente trabajo solo trataremos el formato WEFAX, debido a que es el único tipo de imágenes que actualmente se pueden recibir en el laboratorio de comunicaciones eléctricas de la Escuela de Ingeniería Eléctrica.

1.8.1 Transmisión Whether Facsímil

Las transmisiones en la banda-S son recibidas por el equipo receptor de Imágenes Meteorológicas WSR515 esta señal es de tipo Analógico descrita como Wheter Facsímil o WEFAX, esta señal es diferente de una APT principalmente en que WEFAX es transmitida en tramas, mientras que la transmisión APT es continua.

El termino WEFAX involucra la retransmisión de imágenes satelitales de baja resolución de los Satélites geoestacionarios, Polares y otros datos meteorológicos a través del GOES.

Satélites Meteorológicos y funcionamiento de el Equipo Receptor de Imágenes

1.8.2 Historia del WEFAX

WEFAX fue introducido por primera vez como un experimento innovador en las comunicaciones sobre satélites tecnológicos de Aplicación ATS-1, mas tarde se incluyo sobre el ATS-3, y mejorado posteriormente para incorporarse sobre los SMS/GOES.

La transmisión de las imágenes WEFAX se realizo desde los satélites SMS/GOES, los cuales comenzaron a operar en 1976, inicio y se mantiene permanente sobre el enlace descendente Banda-S, a una frecuencia de 1691.0 Mhz. Dentro de los primeros días de SMS/GOES, se asumió que un alto porcentaje de usuarios de WEFAX ya tenían operando estaciones receptoras de APT dentro de la banda de los 135 a 137 Mhz, y estas mismas estaciones se utilizan para recibir señales WEFAX.

1.8.3 Productos wefax y fuente de datos.

La radiodifusión WEFAX se realiza de los siguientes productos primarios:

GOES VISSR imagery
Polar Orbiter Imagery
National Meteorological Center (NMC) charts
TBUS Bulletins
Operational Messages

1.8.4. Imágenes GOES VISSR

La mayoría de productos transmitidos sobre WEFAX son derivados desde los satélites operando en el modo de imagen VISSR. El VISSR provee las imágenes satelitales en ambos espectros visible e infrarrojo.

Los datos en bruto VISSR son transmitidos a la Estación de Comando y Adquisición de Datos (CDA), ubicada en Wallops Station, V.A., como se muestra en la Figura 1.9. Los datos son recibidos por uno de dos sistemas utilizando antenas parabólicas de 60 pies de diámetro. Previo a la retransmisión de los datos de regreso a el satélite, los datos son procesados en el CDA para incrementar su utilidad. El enlace ascendente desde Wallops utiliza modulación PSK a 2029.1 Mhz, como se muestra en la figura 1.9. La señal del enlace ascendente es recibida por el satélite con una calidad de -25 dB, esta señal es amplificada y trasladada a una frecuencia de 1687.1 Mhz, esta señal es recibida en

Satélites Meteorológicos y funcionamiento de el Equipo Receptor de Imágenes

suitland, Md., sobre uno de dos sistemas de recepción, cada uno compuesto de antenas de 24 pies de diámetro. La señal es enviada desde suitland, Md., a Wallop nuevamente a través de la línea telefónica, luego esta señal WEFAX es enviada desde Wallops ascendentemente a el satélite GOES a una frecuencia de 1691 Mhz, luego esta señal es retransmitida a los usuarios.

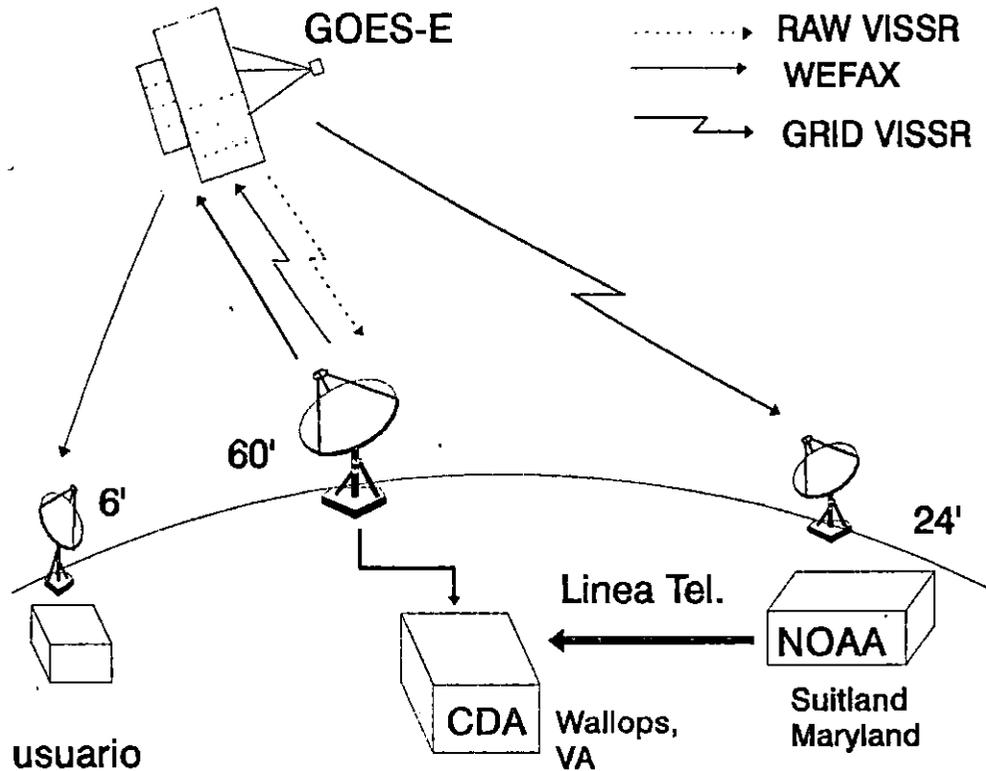


Figura 1.9 Flujo de datos GOES

El formato de la imagen WEFAX es tal que una transmisión de una imagen completa consiste de cuatro partes: El tono de inicio, la señal de fase, la porción de datos, y el tono final. Una transmisión completa consiste de 920 líneas, cada una de 250 milisegundos de duración.

El tono de inicio es de 5 segundos o 20 líneas de duración y consiste de la subportadora de 2400 Hz, que esta siendo modulada por una onda cuadrada de 300 Hz.

La señal de fase es de 20 segundos o 80 líneas de duración,

Satélites Meteorológicos y funcionamiento de el Equipo Receptor de Imágenes

la modulación de la subportadora es tal que durante los primeros 12.5 milisegundos de cada línea son completamente modulados. Este es el mínimo nivel de la subportadora (Equivalente a un nivel blanco), para el resto de la línea , 237.5 milisegundos la subportadora no se modula y el nivel de 2400 Hz, es maximo o Blanco.

Durante la porción de datos de la transmisión, existen 200 segundos de duración o 800 líneas, los primeros 12.5 milisegundos, son modulados por una señal rectangular tal que 2/3 sean negro, y 1/3 sea blanco.

El tono final es de 5 segundos de duración o 20 líneas y consiste de la subportadora de 2400 Hz, que esta siendo modulada por una onda cuadrada de 450 Hz. Existirán 30 segundos de pausa entre el final de una imagen y el inicio de la siguiente. Este formato se describe en la figura 1.10

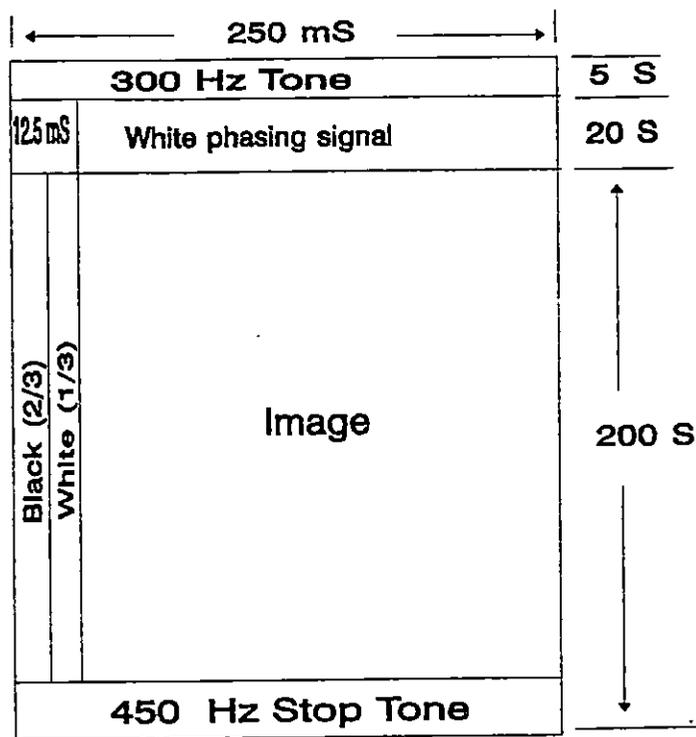


Figura 1.10 Formato de una Imagen WEFAX

CONCLUSIONES

Conocer el principio básico de funcionamiento del equipo receptor de imágenes, así como la teoría relacionada con los formatos de las imágenes satelitales, ha sido el tema principal de este capítulo, no dejando atrás información tan importante como lo son los datos de ubicación y cobertura de satélites meteorológicos geoestacionarios operados por los Estados Unidos, Europa, Japón y Rusia.

Cabe resaltar la observación de que información, acerca de posición y adelantos de los satélites geoestacionaria debe constantemente ser investigada, para lo cual es necesario tener una constante comunicación con el Departamento de Comercio de los Estados Unidos, así como con el departamento de Sinoptica de la división de Meteorología e Hidrología del Ministerio de Agricultura y Ganadería.

REFERENCIAS BIBLIOGRÁFICAS

Feedback Instrument Limited, Wheter Satellite Receiver WSR513 volumen 1, Operating Manual. Englad,1984.

Feedback Instrument Limited, Wheter Satellite Receiver WSR513 volumen 2, Decoder Acces Panel for student Experiments. Englad,1984.

Feedback Instrument Limited, Wheter Satellite Receiver WSR513 volumen 3, S-Band Controller WSR515. Englad,1984.

Martínez C. Carlos Eugenio. Sistema de Adquisición de Imágenes Meteorológicas, Proyecto de Ingenieria Electrica, Universidad de El Salvador 1994.

Richard M. Clark y EARl W. Feigel, The WEFAX User's Guide. U.S. Departament of commerce, Washington, D.C., Noviembre 1984.

CAPITULO II

ADQUISICIÓN DE DATOS Y DISEÑO DE LA INTERFACE

Adquisición de datos y diseño de la Interface

En el presente capítulo se detalla los criterios utilizados en la construcción de la interface, así como también se le da especial interés a la forma en que se realiza la adquisición de datos, esto es a través de el puerto paralelo.

También se explica el funcionamiento de la tarjeta controladora de disco UN-1072 cuyo puerto paralelo es capaz de funcionar como entrada/salida, todos estos detalles se explican para comprender de manera sencilla el funcionamiento de la interface hardware utilizada para realizar la adquisición de datos.

2.1. Técnicas de transferencia de Entrada/Salida.

Resulta obvia la necesidad de la existencia de una comunicación de el computador con el exterior, dado que es necesario gobernar la tarea que se ejecuta en el con los comandos y datos procedentes de el exterior, siendo necesario igualmente, obtener una serie de datos o acciones resultantes de la ejecución de los programas.

Se denomina periférico a todo elemento relacionado (conectado) con el computador no perteneciente al entorno CPU+Memoria. Esta definición incluye a los controladores de periféricos y parece, como así es efectivamente, crear una especie de frontera que en ocasiones permite realizar un tratamiento relativamente sencillo y racional de sistemas complejos.

La salida de un dato se reduciría, desde el punto de vista de la CPU, a depositar dicho dato en la frontera, de donde el dispositivo exterior se encarga de recogerlo y tratarlo acorde a la función que realice. Este tratamiento, por parte del periférico, ya no es de la incumbencia de la CPU, por lo que existe una clara separación funcional de áreas de forma similar se predecirá la entradas de datos, la CPU se limitara a recoger estos de la frontera, donde el periférico los habrá depositado.

Puede parecer, por lo expuesto hasta el momento, que la interface es únicamente circuitería (hardware). Al igual que en una computadora, hardware y software son inseparables, y ni uno ni otro tendrían sentido por sí solos, en un interface se distinguen también estas dos partes que colaboran o se

Adquisición de datos y diseño de la Interface

complementan. Existirá, por tanto, una circuitería que realizara un serie de funciones pero la parte inteligente y que controla el funcionamiento del interface acostumbra a ser un programa (ver figura 2.1)

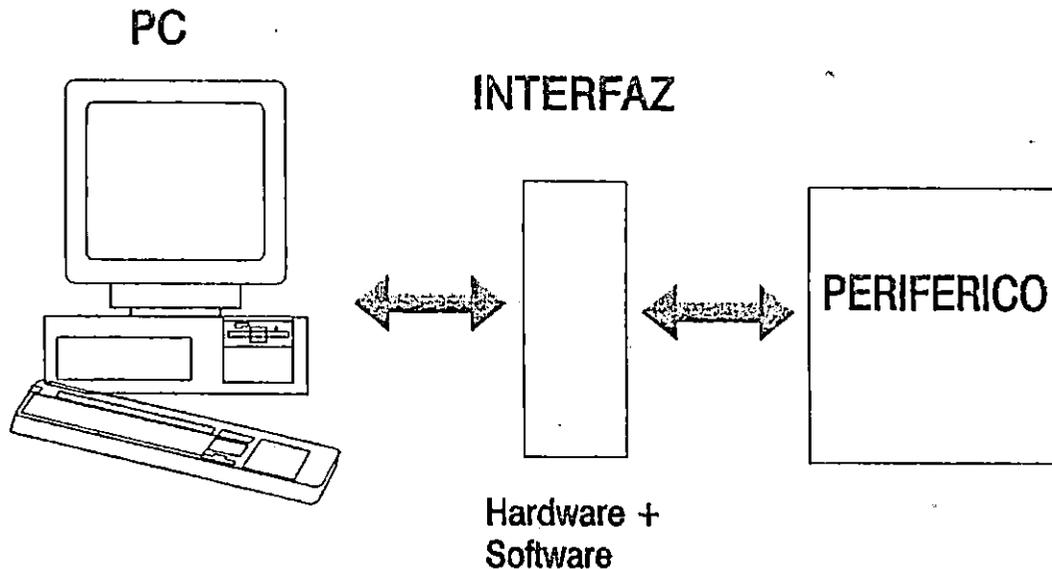


Figura 2.1 Esquema que resume el concepto de Interfase

2.2. Coordinación de las transferencias de Entrada/Salida

Existen dos tipos de información en las transferencias de entrada salida E/S:

a) Datos: Constituyen la entrada de información para procesar y la salida de resultados

b) Control: Como salida se utiliza esta información para el gobierno de los periféricos y como entrada para obtener información de el estado de los mismos.

El fenómeno de transferencia de E/S implica la existencia de una comunicación ente CPU + Memoria y periféricos y como en toda comunicación va a ser necesario establecer las normas que la van a regir o lo que es lo mismo definir un protocolo de comunicación este protocolo podrá ser muy complicado o muy sencillo, dependerá en general de la aplicación y deberá asegurar en todo momento que no se pierda dato alguno en la transferencia de información.

Adquisición de datos y diseño de la Interface

Un ejemplo de protocolo sencillo podría ser el existente en una aplicación en la que se emplee un conversor de A/D, entre este y la CPU, en el cual el conversor se limita a decir que ha convertido de analógico a digital un dato y que la CPU lo puede recoger si lo desea. A partir de este momento el conversor seguirá convirtiendo datos y si la CPU no ha leído el anterior, este será sustituido por el nuevo dato convertido, en este caso existe un monologo, el conversor avisa a la CPU que tiene un dato pero no espera que este lo recoja para convertir otro.

Este protocolo (que puede ser ni siquiera considerado como tal) tan simple, no es valido en todas las ocasiones, debiendo recurrir a otros mas complicados en forma dialogadamente, el diálogo es realizado en base a la información de control y su fin es precisamente la de lograr una comunicación de la información de datos.

La información de control puede viajar a través de líneas de los periféricos dedicadas únicamente a este fin o bien a través del bus de datos, procedente de los registros de control de los periféricos.

2.3. Clasificación de las técnicas de transferencia

Respecto a la forma de ejecución de la transferencia cabe distinguir entre tres alternativas diferentes:

a) Transferencia por programa

El intercambio de información entre computador y periférico se hace vía CPU, mediante un programa que se encarga de transferir esta información dentro de este tipo de transferencia se puede distinguir a demás dos modalidades de la forma de control de la comunicación entre CPU y periférico:

a.1) Transferencia sincróna : la transferencia se realiza por iniciativa del programa, sin consultar al periférico si esta o no dispuesto a aceptar o a entregar el dato que se le envía o que se le pide. Se supone por defecto que el periférico siempre estará listo. Este tipo de transferencia es la mas sencilla de resolver, tanto en lo referente al software como al hardware, pero su utilización esta siempre restringida a periféricos del tipo antes expuesto.

Adquisición de datos y diseño de la Interface

a.2) **Transferencia Asíncrona** : En este caso es preciso establecer un diálogo entre CPU y periférico a fin de controlar la ejecución de la transferencia en el instante y a la forma adecuadas, cabe considerar nuevamente dos alternativas en cuanto a la forma de control:

a.2.1) **Por consulta de estado**: Se pregunta al periférico sobre la disposición para aceptar la transferencia.

a.2.2) **Por interrupción**: Se delega en el periférico la facultad de avisar al CPU cuando esta dispuesto aceptar la transferencia.

b) Transferencia por Acceso Directo a Memoria (DMA):

La CPU ya no interviene en la transferencia, siendo esta realizada de forma directa entre memoria y periférico (o a la inversa) por otro periférico denominado controlador de DMA, aunque la CPU no intervenga propiamente en la transferencia se deberá encargar de programar al controlador de DMA.

c) Transferencia controlada por procesador de E/S especializado:

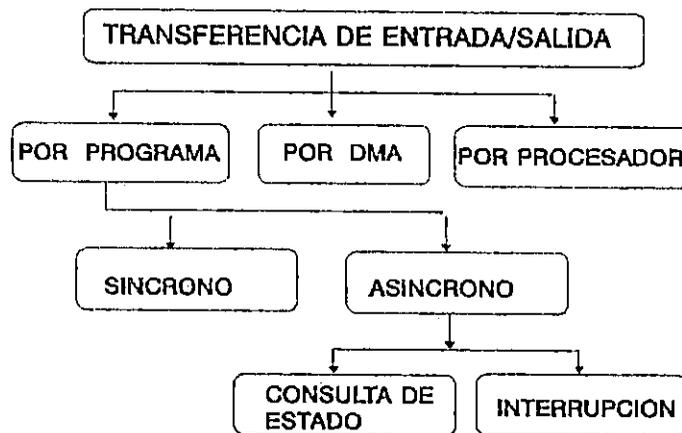
La complejidad y alto volumen de datos de ciertas transferencias de E/S condujeron al diseño de procesadores dedicados únicamente a tareas de E/S, estos procesadores al igual que la CPU ejecutan un programa (igualmente almacenado en la memoria central), siendo a través de este como se controla la transferencia de datos. Supone una mejora considerable a la eficiencia de los controladores de DMA. La figura 2.2 resume los diferentes tipos de E/S

2.4. Transferencia por consulta de estado

Como se ha visto en el apartado anterior, la transferencia por consulta de estado, están enmarcadas dentro de las transferencias por programa, en las que la CPU gestiona totalmente, a través de un programa, el intercambio de información con los periféricos.

En este tipo de transferencia la información de control (estado de el periférico) viaja hacia la CPU utilizando

Adquisición de datos y diseño de la Interface



DIFERENTES TIPOS DE ENTRADA SALIDA

Figura 2.2 Diferentes tipos de Entrada/Salida

únicamente el bus de datos, dentro de este tipo de transferencia tenemos la comunicación a través de el puerto serie y el puerto paralelo que es el tipo de transferencia que se utiliza en el presente trabajo para realizar la adquisición de datos.

2.4.1 Puerto Serie

El método mas común para realizar adquisición de datos desde instrumentos hacia el computador personal es a través de el puerto serie RS-232, virtualmente muchos PC tienen tal puerto y pueden soportar comunicación serie a velocidades al rededor de los 20 Kbytes/segundos véase figura 2.3 Aunque la velocidad sea lenta este método es muy adecuado para muchos requerimientos de adquisición de datos, es de notar que la operación de lectura es mucho mas lenta que la de escritura, otra dificultad además de la velocidad es la potencia requerida, si el puerto no ha sido diseñado para aplicaciones portátiles el puerto será físicamente inútil o incompatible con la fuente de potencia ya que en aplicaciones de adquisición de datos se debe alimentar generalmente las interfaces desde el puerto.

Adquisición de datos y diseño de la Interface

2.4.2 Puerto paralelo

El puerto paralelo, cuya aplicación principal es comunicarse con el impresor, puede ser usado como interface de adquisición de datos capaz de funcionar mejor que la interface serie RS-232.

Mientras que la velocidad de los datos en una interface serie es típicamente en el rango de 5 a 10 Kbytes, el puerto paralelo lleva acabo velocidades alrededor de los 50 a 150 Kbytes/seg que son mucho mas rápidas que la interface serie RS-232 y mas económicos si lo comparamos con algunas interfaces que existen comercialmente (por ejem. PCMCIA: Personal computer Memory Card International Association). Esta es una de las ventajas que hacen atractivo el usar el puerto paralelo como interface para adquisición de datos.

Por muchos años el puerto paralelo solo fue ocupado como salida a la impresora, pero hace pocos años atrás se desarrollo un proyecto sobre el puerto de la impresora llamado "interface normalizado bidireccional para computadores personales" este es normalizado por el P1284 IEEE comite. El comite se propuso mejorar la operación de los puertos paralelos lográndose con esto la construcción de los puertos capaces y avanzados (ECP) y los puertos paralelos avanzados (EPP).

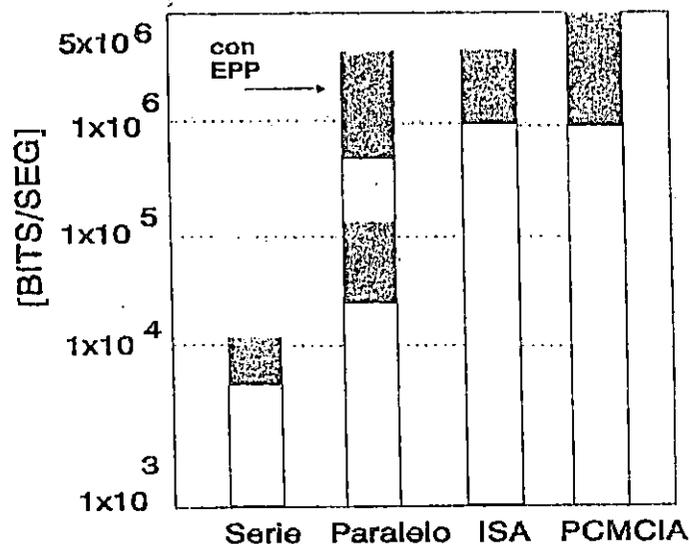


Figura 2.3 Comparación de velocidad de puertos utilizados en la adquisición de datos

Adquisición de datos y diseño de la Interface

ECP ha sido promovido por Microsoft, el EPP ha sido promovido por la corporación INTEL y es una generalización de el puerto paralelo. Es un puerto paralelo cuyas direcciones y datos están multiplexados dentro de un bus de propósito general de 8 bits.

Desafortunadamente computadores con ECP y EPP no representan una buena inversión en el mercado. Tales puertos pueden ser agregados a los computadores personales cambiando o reemplazando la tarjeta de el puerto paralelo.

Ya que muchos computadores poseen solamente un puerto paralelo este es diseñado de tal manera que las señales de control no permiten que se den dos situaciones simultáneas.

Cuando se usa el puerto paralelo como entrada la línea de salida es deshabilitada y el CPU puede leer señales de nivel TTL que se encuentren el bus de datos. Esta capacidad bidireccional varía entre computadores y debe ser individualmente determinada para cada computador individual.

El puerto paralelo de los computadores personales usa un conector de 25 pines, que contiene 17 señales activas y ocho que se conectan a tierra. Las 17 señales activas se dividen de la siguiente manera:

- * 8 son para datos
- * 5 son de status
- * 4 son líneas de control

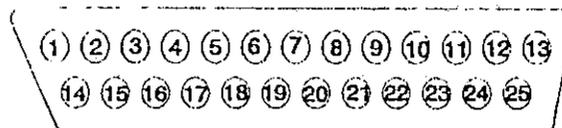


Figura 2.4 Conector Paralelo de 25 Pines

La asignación de pines par el conector de el puerto paralelo se muestra en la tabla siguiente:

Adquisición de datos y diseño de la Interface

Anatomía del conector para puerto paralelo de 25 pines

Pin	Descripción
1	-STROBE
2	Bit de datos 0
3	Bit de datos 1
4	Bit de datos 2
5	Bit de datos 3
6	Bit de datos 4
7	Bit de datos 5
8	Bit de datos 6
9	Bit de datos 7
10	-ACK
11	BUSY
12	PE
13	SLCT
14	-AUTO FEED XT
15	-ERROR
16	-INIT
17	-SLCT IN
18 A 25	Tierra

Las líneas de datos pueden ser utilizadas para leer o escribir y la operación que realizan se controla a través de el registro de control.

Las líneas de control comprenden los cuatro bits menos significantes de el puerto de control de 8 bits, los otros bits son usados por el puerto paralelo como control de interrupciones y algunas veces como puerto de datos. Las líneas de control pueden ser usadas como entradas, pero en general esto no es recomendable ya que algunos puertos paralelos usan integrados cuya salida son en forma de totem-pole y no lo mas común que son de colector abierto. Algunas de las líneas de control son invertidas mientras que otras no lo son (ver fig.2.5).

Las líneas de status comprenden los 5 bits mas significantes de el puerto de status de 8 bits, los otros 3 bits no se usan. Las líneas de status se usan únicamente para lectura y comprenden los 5 bits mas significantes de un bus de 8 bits, donde los tres bits menos significantes no se usan, las 5 líneas activas aceptan señales cuyos niveles de voltaje sean TTL, como sucede con el puerto de control algunas de las líneas están invertidas y se debe tener cuidado en esto (ver fig.2.5).

Adquisición de datos y diseño de la Interface

El bit 6 de este puerto de status es la señal de petición de interrupción IRQ, si este bit es habilitado (colocando un nivel lógico alto '1' en el bit 4 de el registro de control), en la transición de 0 a 1 de esta señal se interrumpirá el CPU. Desafortunadamente algunas computadoras usan la negación de 1 a 0 y por lo tanto complican el uso de el puerto paralelo ya que nuevamente hay que verificar bajo que transición se puede interrumpir la CPU.

Resumiendo podemos decir que para aplicaciones sencillas de adquisición de datos la capacidad de el puerto paralelo es la siguiente : las líneas de datos pueden ser usadas como entrada/salida, las líneas de control deberán de ser usadas como salida solamente, las líneas de status deberán ser usadas como entrada solamente.

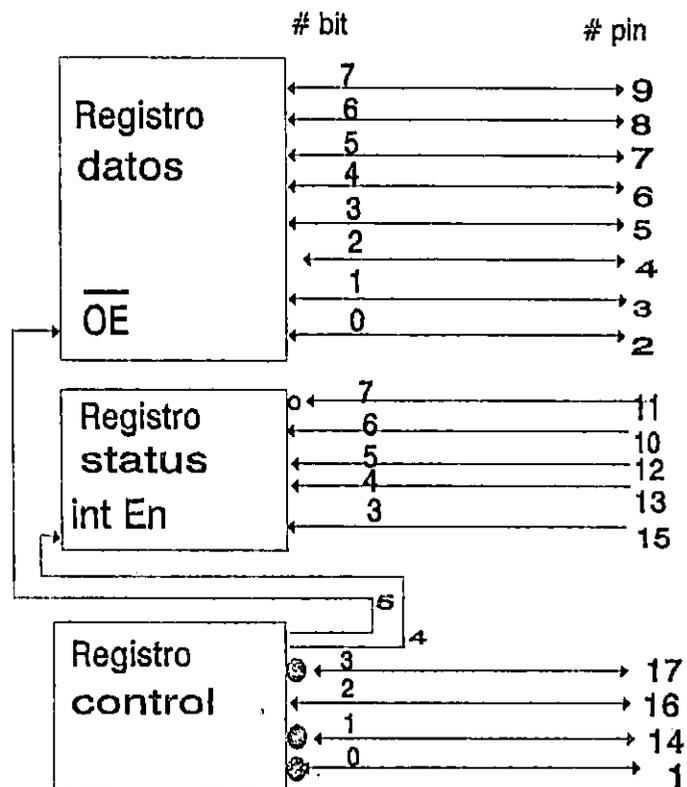


Figura 2.5 Representación en forma de Registro del puerto paralelo

Adquisición de datos y diseño de la Interface

2.5. Tarjeta de entrada/salida UN-1072

La tarjeta de entrada/salida UN-1072 ha sido diseñada para trabajar con los computadores PC/AT y compatibles. Esta tarjeta consta de dos puertos serie, un puerto paralelo, un puerto de juego, un puerto controlador de disco y un puerto controlador de disco duro.

La fig 2.6 muestra la distribución de cada uno de los puertos y la construcción física de la tarjeta.

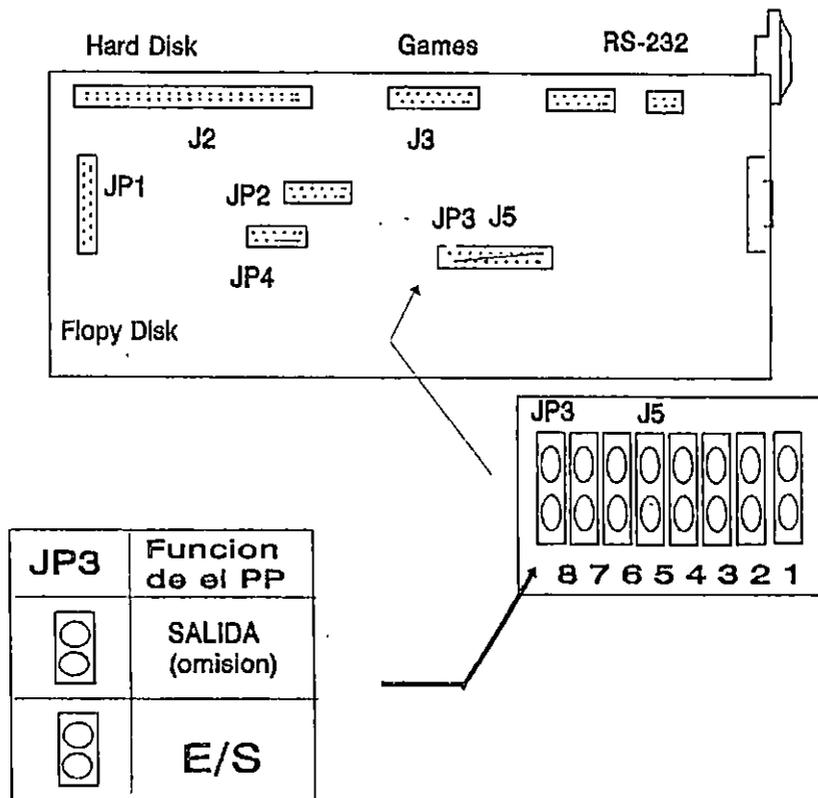


Figura 2.6 Tarjeta IDE I/O UN-1072

La parte de interés para el presente trabajo es el puerto paralelo, de las especificaciones de el fabricante de la tarjeta entrada/salida UN-1072, podemos configurar la tarjeta dejando

Adquisición de datos y diseño de la Interface

abierto el jumper numero 3 y la función de entrada/salida depende de el valor que se ponga en el registro de control.

De la figura 2.5 puede notarse que el registro de control tiene dentro de sus bits uno que le sirve para habilitar el uso de el printer si se desahabilita este bits, podemos efectuar operaciones de lectura de datos desde el puerto paralelo.

Ya que el bits de habilitación de printer corresponde al bit menos significativo de el un bus de 8 bits basta con poner un nivel lógico alto ('1') en el registro de control, ya que como se dijo anteriormente el registro de control tiene alguna de sus líneas negadas y de ahí que se hace evidente el que se haya seleccionado el uno lógico en el bit menos significativo.

También experimentalmente se determino que el bit 5 de el registro de control debe tener un nivel lógico alto, se dice que se obtuvo experimentalmente ya que según las especificaciones en los libros de texto este línea no se usa.

Esto equivale decir que para configurar la tarjeta de entrada/salida UN-1072 se necesitan básicamente dos cosas:

- * El jumper 3 debe estar abierto
- * El registro de control debe tener un valor de XX1XXXX1

2.6. Diseño de la interface de adquisición de datos

La adquisición de datos se lleva acabo a través de una interface que se encuentra entre el receptor de imágenes meteorológicas WSR513 y el PC/AT LEMON 386, la cual consiste básicamente de optoacopladores y buffers triestado.

Los optoacopladores sirven para aislar la señal que proviene de el receptor de señales meteorológicas WSR513 y el computador, de tal forma que si sucede algún fallo en cualquiera de los dos equipos se aislé únicamente la parte afectada.

Ya que el puerto paralelo esta diseñado para salida de datos es necesario configurarlo de tal forma que pueda ser usado como entrada de datos, según las especificaciones de la tarjeta UN-1072 esta configuración se lleva acabo abriendo el jumper 3 . Al efectuar esta operación el puerto paralelo puede ser usado como entrada/salida, la función que realice depende de el valor que pongamos en el registro de control.

Adquisición de datos y diseño de la Interface

Para evitar que por accidente o descuido se conecte el puerto paralelo al equipo receptor de imagen WSR513 sin antes haber configurado el puerto como entrada de datos, se utiliza un bufer triestado, el cual permanece deshabilitado hasta que el puerto paralelo es configurado para entrada de datos. Fácilmente puede lograrse esta operación si conectamos las líneas de habilitación de el bufer triestado ha el bit menos significativo de las líneas de control.

Así que con esta idea implementada en la interface se garantiza que ningún dato pueda estar entrando a el bus de datos de el puerto paralelo sin antes este no haya sido configurado para ser usado como entrada, de lo contrario se correría el riesgo de dañar la tarjeta UN-1072

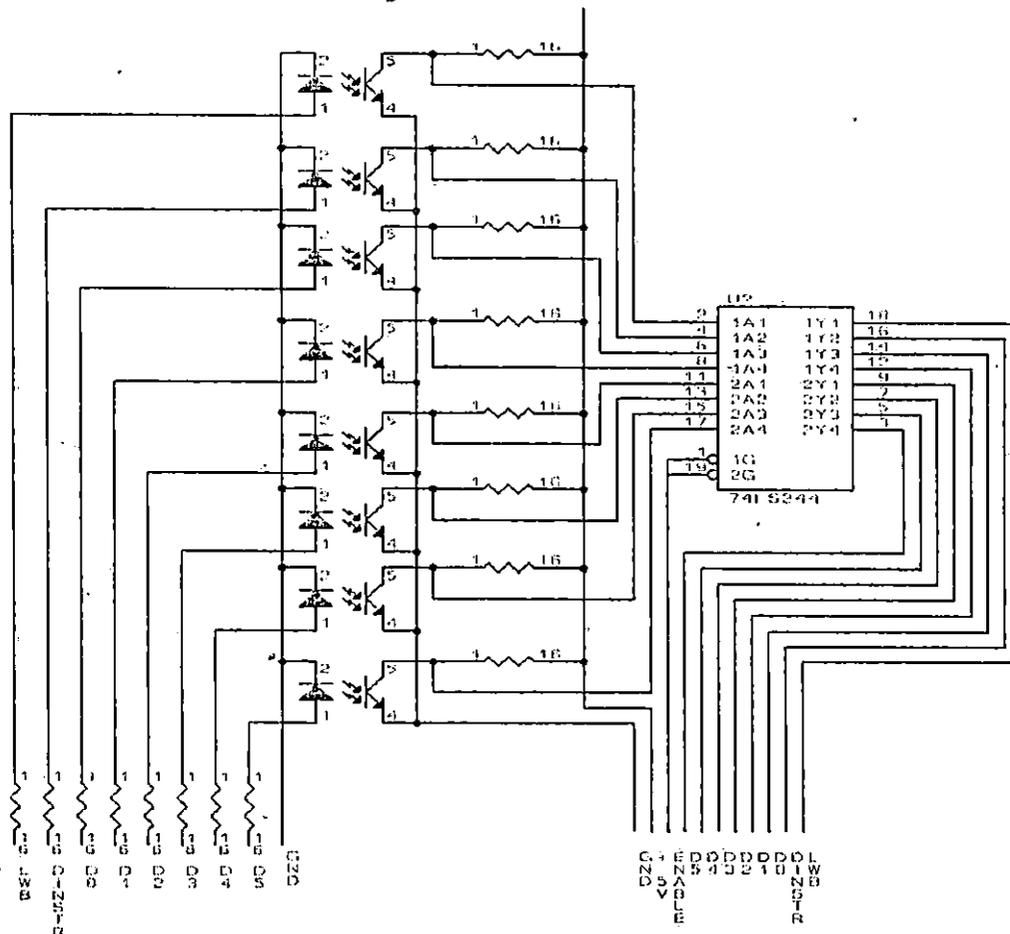


Figura 2.7. Diseño de la Interface de Adquisición de Datos

CONCLUSIONES

En el presente capítulo se ha tratado de explicar de una forma sencilla el concepto, y las diferentes técnicas empleadas para comunicar un computador con un periférico, lo que se conoce como adquisición de datos, basta decir que la técnica empleada en este trabajo se conoce como polling, es de tener presente la técnica de interrupción por Hardware, la cual será una de las más explotadas en un futuro próximo, esto es obvio, ya que si revisamos lo antes expuesto observamos que la IRQ 7 de el puerto paralelo responde a flancos, lo que garantiza más exactitud en nuestra adquisición de datos.

Cabe mencionar que paralelo a este trabajo se desarrollan otros, que tienen que ver con la construcción de tarjetas más robustas, potentes y versátiles en la tarea de adquirir datos, así que no sería extraño que este software cambiara un poco sus funciones de comunicación con el mundo exterior.

REFERENCIAS BIBLIOGRÁFICAS

Allen E. Tracht. Adapting laptops to data acquisition, IEEE Spectrum octubre de 1993.

Ruben de Diego Martínez. Apuntes de la Asignatura Fundamento de Ordenadores, Universidad Politecnica de Madrid 1993.

UN-1072 IDE I/O CARD USER'S MANUAL. 1994

CAPITULO III

DISPOSITIVOS DE ENTRADA/SALIDA Y TECNOLOGÍAS GRÁFICAS

Dispositivos Entrada/Salida y Tecnologías Gráficas

Introduccion

Con el objeto de entender de una forma amplia como las imágenes adquiridas desde el equipo receptor son presentadas en el monitor del computador, se explica el principio básico de los monitores cuya base es un tubo de rayos catódicos, a si como las diferentes técnicas conocidas par el manejo de archivos gráficos, este tema es de mucha importancia ya que nos da un amplio criterio para seleccionar un formato de almacenamiento de imágenes.

3.1 Llevando una imagen a la pantalla

La mayoría de los monitores de los computadores personales utilizan un tubo de rayo catódicos (CRT) para proyectar una imagen en la pantalla.

El CRT es una válvula electrónica al vacío con una pantalla plana en un extremo y un cañón de electrones en el otro. El cañón de electrones dispara un flujo de electrones a través de la válvula hacia la pantalla en el otro extremo. Esto no es tecnología reciente: El primer CRT fue construido en 1897.

Los rayos catódicos tiene que ser controlados de alguna forma, por lo que un CRT contiene un cilindro de afoque(que hace que los rayos se enfoquen en un pequeñísimo punto al llegar a la pantalla) y un sistema de desviación para poder mover los rayos vertical y horizontal, moviendo por lo tanto también el punto sobre la pantalla.

La intensidad del rayo de electrones también puede ser controlada para ajustar la brillantez del rayo al alcanzar la pantalla. La pantalla misma está recubierta de materiales especiales llamados fósforos, que brillan cuando el rayo los toca. Este brillo es el que usted ve cuando observa la pantalla del monitor de su computadora, o más bien, el brillo de miles de áreas de la pantalla al ser golpeadas por los rayos catódicos un ay otra vez, la figura 3.1 ilustra lo antes descrito.

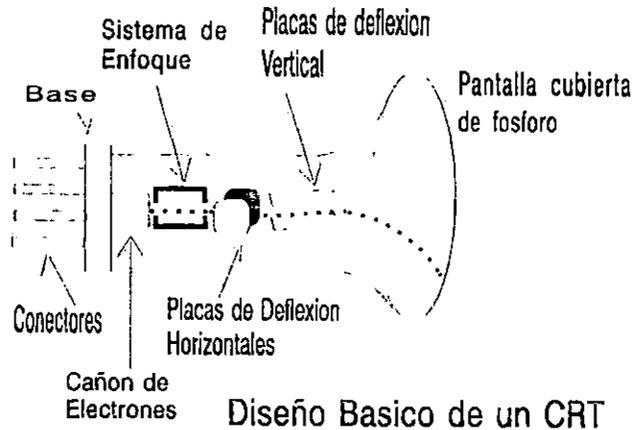


Figura 3.1. Diseño Básico de un CRT

Los monitores de color funcionan de una manera un poco diferente. Las pantallas de color contienen tríadas, o sea grupos de tres puntos de fósforo, uno rojo, uno verde y uno azul (cada punto contiene miles de partículas individuales de fósforo), Estos tres puntos están colocados en forma triangular.

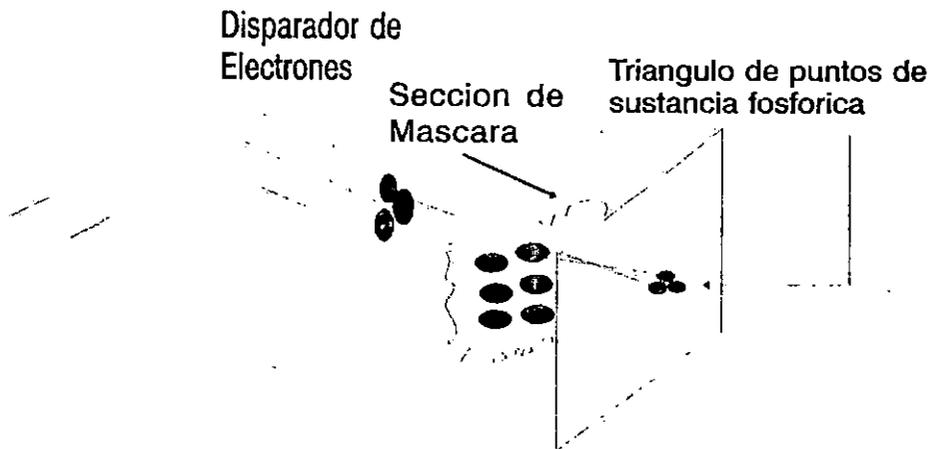


Figura 3.2. Principio de un CRT a Color

Dispositivos Entrada/Salida y Tecnologías Gráficas

El monitor no tiene uno, sino tres cañones de electrones, uno para cada punto de la triada. Por lo tanto, cada punto en la pantalla se forma en realidad por tres pequeños puntos, uno de cada color. Un punto de color es generado al golpear los tres colores primarios con electrones de diferentes intensidades, mezclando los colores de la misma manera que usted mezcla colores de acuarela; un rayo de alta intensidad en el verde añade mucho verde; un rayo de baja intensidad en el rojo, añade solamente un poco de rojo, y así sucesivamente, vease figura 3.2.

Los cañones de electrones se dirigen utilizando una máscara, que es una especie de malla con agujeros, uno para cada triada.

Cuando la pantalla es golpeada por el rayo de electrones, brilla por un fracción de segundo -quizás de 30 a 40 milisegundo- y después desaparece. Para mantener una imagen en video, el rayo de electrones debe golpear el mismo sitio una y otra vez, docenas de veces cada segundo, por lo que el rayo debe hacer un barrido a través de la pantalla, golpeando un punto después de otro en una línea recta, bajando a la siguiente línea y así sucesivamente.

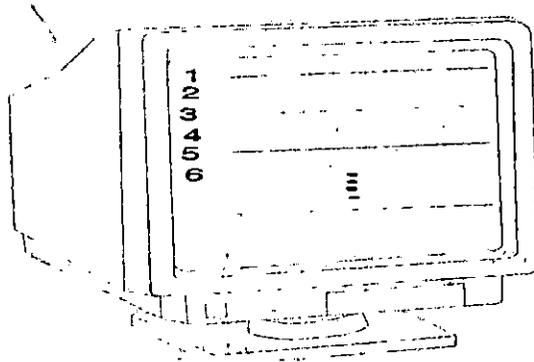


Figura 3.3. Monitor de rastreo. Primero todos los puntos de las líneas pares son desplegados, después las impares

El rayo barre así cientos de estas líneas horizontales para "pintar" una pantalla y luego vuelve a iniciar desde arriba, antes de que el fósforo pierda su brillo, la figura 3.3 ilustra claramente este principio.

Dispositivos Entrada/Salida y Tecnologías Gráficas

3.2 Modos de video

Un modo de video es la manera en la cual su computadora puede colocar imágenes en el monitor: si es que puede producir gráficos o sólo puede desplegar texto, el número de colores que pueden ser desplegados, y la resolución de la imagen (esto es, el número de pixels en la imagen.) La división fundamental entre modos es modo gráfico y modo de texto.

Aunque las imágenes pueden ser dibujadas en el modo de texto, su grado de sofisticación está muy limitado por el hecho de que el programador no puede llegar a los pixels individuales de la pantalla. El elemento mas pequeño con el que pueda trabajar el programador es el tamaño de un carácter, y lo único que se puede hacer es colocarlo en algún sitio como un carácter predefinido. En el modo gráfico, sin embargo, la imagen que se envía a la pantalla está dividida en millares de puntos y de pixels, y cada pixel puede ser ajustado de manera independiente de los demás.

Ahora, en vez de estar limitado a cerca de 2,000 áreas sobre la pantalla(en el modo de texto de 80 por 25, por ejemplo), el programador tiene quizás 64,000 puntos separados sobre la pantalla, cada uno de los cuales puede ser ajustado en forma independiente a los otros. De echo, algunas resoluciones de pantalla permiten a los programas trabajar con 300,000 y hasta con 800,000 pixels.

3.2.1 Resolución de video

Los modos gráficos tiene resoluciones diferentes, esto es, un número distinto de pixels de un lado a otro de la pantalla y hacia abajo de la misma. Esto se denomina a veces resolución dirigitible. Usted ha visto probablemente resoluciones descritas como 360x400, 640x480, y así sucesivamente. Los números se refieren a la cantidad de pixel a través de la pantalla y el número de hileras de pixels hacia abajo en la pantalla; por lo tanto, 640 por 480 significa que la pantalla tiene 640 pixels de ancho y 480 filas de pixels de arriba hasta abajo.

Incidentalmente, los pixels de una imagen en pantalla no están relacionados directamente con los puntos de fósforo en la superficie de la misma. El tamaño de los pixels depende del modo de video utilizado por el software y su controlador, y no por el tamaño físico del fósforo o triadas. Por lo tanto una pantalla

Dispositivos Entrada/Salida y Tecnologías Gráficas

puede ser capaz de manejar muchas resoluciones diferentes, desde la más baja de las resoluciones hasta las más altas.

3.2.2 El sistema de despliegue (Controladores de video)

El sistema de despliegue de muchos computadores, esta basado en la siguiente variedad de estándar de controladores de video que existen actualmente.

- * Monochrome Display Adapter (MDA)
- * Color Graphics Adapter (CGA)
- * Hercules Graphics Adapter (HGA)
- * Enhanced Graphics Adapter (EGA)
- * Multicolor Graphics Array (MCGA)
- * Virtual Graphics Array (VGA)

Todos los controladores excepto el Hercules Graphics Card son compatibles y soportados por IBM a través de los servicios estandares del BIOS y de DOS.

El sistema de video puede ser accesado de tres formas diferentes:

1. A través de las funciones de llamada del DOS. Este método es el mas compatible, pero es el mas lento.
2. A través de las funciones de llamada del BIOS. Este método razonablemente compatible de acceder el video es mas rápido que DOS. Muchos sistemas, pero no todos son compatibles con este método de acceder la pantalla de video.
3. Directamente al nivel de Hardware. Este método es incompatible por que el hardware difiere ampliamente entre muchos sistemas existentes. La ventaja principal de este método es su velocidad.

3.3 Guardando y Desplegando datos en video.

El sistema de despliegue de los computadores personales esta basado en el Integrado controlador de tubo de rayos catódicos (CRTC) 6845 de motorola. Los sistema EGA y VGA usan integrados

Dispositivos Entrada/Salida y Tecnologías Gráficas

al gusto basados en el 6845. Este integrado maneja importantes tareas de despliegue de tal manera que el programador no tenga que molestarse en realizarlas.

El diseño del sistema es conceptualmente simple. Una presentación en video de un PC es un mapa de memoria, dentro del cual cada cosa que aparece reflejada sobre la pantalla esta en la memoria del computador(Véase la figura). Un Buffer de memoria guarda la información que se presenta en la pantalla. La dirección de Inicio de la memoria buffer y el tamaño varia, dependiendo del tipo del adaptador de video en uso, El modo actual de despliegue, y la cantidad de memoria de despliegue.

Los adaptadores de video generalmente contienen desde 4K a 256K de memoria. Porque los datos necesarios a desplegar sobre la pantalla pueden ser mucho menores que esta cantidad de memoria, Algunos adaptadores de video pueden controlar mas que una pantalla de despliegue. Notese que se dice pantalla de despliegue y no monitor de despliegue. Pantallas de despliegue, o páginas, son la representación de memoria de lo que aparece sobre su pantalla. La tabla siguiente muestra la dirección de inicio de la memoria buffer, el largo del buffer, y el numero de páginas de despliegue para los diferentes tipos de despliegue.

Controlador	Modo	Dirección del buffer	Largo del buffer	Páginas de Despliegue
MDA	Text	B000h	4K	1
CGA	Text Graphics	B800H B800h	16k 16k	4/8 1
EGA	Mono Text Graphics CGA	B800h B000h A000h B800h	Varia Varia Varia Varia	Varia Varia Varia Varia
VGA	Mono Text Graphics	B000h B800h A000h	256K 256K 256K	8 8 1/2/4/8

Tabla 3.1., Tamaño, dirección y número de páginas para los diferentes adaptadores de video.

Dispositivos Entrada/Salida y Tecnologías Gráficas

El tamaño del buffer de la tarjeta EGA varía porque la tarjeta EGA puede tener 64k, 128k, o 256k de memoria.

3.4 Formas de desplegar video.

La interpretación de los adaptadores de video depende de el modo de video que se este utilizando, el cual controla la apariencia de la pantalla. la tabla siguiente detalla los modos de despliegue disponibles por los diferentes tipos de adaptadores de video.

Modo	Tipo	Color	Resoluc	CGA	EGA	VGA
00h	Text	16	40x25	x	x	x
01h	Text	16	40x25	x	x	x
02h	Text	16	80x25	x	x	x
03h	Text	16	80x25	x	x	x
04h	Graphics	4	320x200	x	x	x
05h	Graphics	4	320x200	x	x	x
06h	Graphics	2	640x200	x	x	x
07h	Text	Mono	80x25		x	x
08h	Graphics	16	160x200			
09h	Graphics	16	320x200			
0Ah	Graphics	4	640x200			
0Bh	---RESERVADO---					
0Ch	---RESERVADO---					
0Dh	Graphics	16	320x200		x	x
0Eh	Graphics	16	640x200		x	x

Dispositivos Entrada/Salida y Tecnologías Gráficas

0Fh	Graphics	Mono	640x350		x	x
10h	Graphics	16	640x350		x	x
11h	Graphics	2	640x480			x
12h	Graphics	16	640x480			x
13h	Graphics	256	320x200			x

Tabla 3.2. Modos de despliegue disponibles por los diferentes tipos de adaptadores de video.

El numero dentro de la columna resolucion representa fila y columna para modo texto, y pixels para modo gráfico.

3.5 Bits y color

El VGA Estándar tiene 16 colores. Cada pixel en la pantalla necesita de 4 bits para describirlo, y por lo tanto el VGA estándar se conoce como color de 4 bits. Cuatro bits (medio byte) pueden ser utilizados para crear 16 diferentes números :0000, 0001, 0010, 0011, y así sucesivamente. Cada uno de estos números se asigna a un color específico, por lo que cada pixel puede ser 1 de 16 diferentes colores. Por lo tanto, el color de 4 bits tiene una paleta de colores de 16 diferentes colores.

Para producir 256 colores, necesitamos asignar 8 bits (un byte) para cada pixel, porque un byte puede ser utilizado para crear 256 números distintos. Otra vez, cada número se asigna a un color específico. La descripción de un pixel con un numero en particular, hace que se despliegue dicho color automáticamente. Para dar el salto hasta 16,777,216 colores, necesitamos de 3 bytes (24 bits) de información por cada pixel en la pantalla. Así, con 24 bits se puede generar 16,777,216 números distintos, cada uno de ellos asignados a un color distinto, ocho bytes se asigna a cada uno de los tres canales de colores de video (rojo, verde, y azul), por lo que de cada uno de esos colores primarios, se pueden definir 256 tonos. Al mezclar esos tonos, alcanzamos 16,777,216 diferentes colores.

Dispositivos Entrada/Salida y Tecnologías Gráficas

3.6 Gráficos.

Las imágenes gráficas, al igual que los archivos de texto, también utilizan valores binarios para representar la información que contienen:

Sin embargo, la información en un archivo de imagen se interpreta de una forma enteramente distinta. Los bytes de una imagen representan grupos de números. Estos números describen los atributos de cada sección individual de la imagen. Llamamos a estas secciones elementos pictóricos o píxeles. Cada pixel es un solo punto sobre la página impresa o sobre la pantalla de su computadora. (Observe su monitor, o una fotografía de un periódico, desde muy cerca, para ver estos puntos.) Los valores numéricos en un archivo de la imagen codifican la intensidad de cada pixel en la imagen, determinando si un pixel es brillante o no. También se usan varios números para representar la intensidad de los componentes de color de cada pixel.

La cantidad de información (cuantos números) que se usa para describir el color y brillantez de cada pixel, se conoce como profundidad de la imagen.

Naturalmente, las imágenes son algo mas que conjuntos de números en código binario. Cada imagen es una representación abstracta de algo. Este algo, ya sea una fotografía o la representación gráfica de una función matemática, es la realidad subyacente a la información de la imagen. Esta realidad puede ser muestreada o puede ser calculada de alguna forma para reducirla a información computacional abstracta.

Utilizando el modelo apropiado de formación de imágenes - que es simplemente una manera normalizada de representar e interpretar información gráfica como un conjunto de números- la verdadera imagen puede ser reconstruida.

Como en cualquier reconstrucción, que debe ser, para ser verdadera, fiel al original, las imágenes gráficas por computadoras persiguen obviamente el mismo objetivo. Probablemente no podrá jamás almacenar suficiente información para conseguir una reconstrucción exactamente fiel a la imagen original,, pero si la imagen por computadora cumple con sus especificaciones par la reproducción dentro de cualesquiera limites de error que se haya fijado, entonces la tecnología seleccionada de formación de imágenes es adecuada.

3.7 Modelo de barrido vrs. modelos vectoriales

Existen dos formas para describir imágenes por medio de una computadora: descripciones por barrido y descripciones vectoriales. Se conocen más comúnmente como gráficos en mapas de bits y gráficos basados en objetos, respectivamente. Debemos apuntar que ambas alternativas generan mapas de bits y que estos mapas de bits son utilizados por su computadora para crear finalmente la imagen en la pantalla. Es solamente que los gráficos basados en objetos introducen en el proceso un nivel adicional de abstracción, como se verá más adelante.

Los dos modelos alternativos utilizan técnicas de programación, formatos de archivos y capacidades diferentes. Hablaremos más sobre estas diferentes conforme se avance el capítulo.

3.7.1 Mapa de bits

Un mapa de bits es un dibujo de su imagen literalmente en forma de mapa. Cada imagen se divide en un conjunto de puntos o pixels que pueden estar conectados o no, pintados de negro o blanco, o con sombreados intermedios (escala de grises), o coloreados con algún componente de color primario. La forma más común de los pixels es rectangular o cuadrada, de tal manera que un pixel se ajusta con el adyacente para formar una imagen continua. Conceptualmente pueden ser descritos como la intersección de líneas en una malla. Los programas que utilizan mapas de bits son programas de pintar como paintbrush IV, programa de edición de imágenes como PhotoStyler de Aldus.

El tamaño de un pixel determina la resolución del mapa de bits. Mientras más pequeños y, por extensión, mientras más densamente estén agrupados mayor será la resolución; sin embargo, se requerirá de más información para describir la imagen. El despliegue de video VGA de 640x480 requiere de 307,200 pixels, o se información de 38.4 KB para blanco y negro. Para 256 colores en la pantalla (ocho bits por píxel), esta descripción se multiplicaría por ocho. Compare estas cifras con los 3 KB más o menos que se requieren para almacenar una página de texto en un archivo de un procesador de palabras.

Los objetos en mapas de bits son dependientes de la resolución, esto es, la información que describe un mapa de bits tiene un tamaño intrínseco de pixel.

Dispositivos Entrada/Salida y Tecnologías Gráficas

¿ Por qué los mapas de bits son algunas veces llamados imágenes de barrido? Por que las imágenes que crean se forman de líneas horizontales formadas por píxels y dichas líneas se llaman líneas de barrido. De echo, es el mismo procedimiento que utiliza la pantalla de video o la televisión. En un monitor de rayos catódicos(CRT), el rayo de electrones es barrido a través de la superficie interior de la pantalla, encendiéndose cuando un pixel debe aparecer iluminado, y apagándose cuando debe aparecer obscuro. Un mapa de bits en la computadora es utilizado para controlar el rayo de electrones, determinando que pixels estarán iluminados y cuales oscuros y que colores tendrán todos.

Puede usted pensar en el rayo de electrones como "pintando" la imagen en el interior de su pantalla, siguiendo las instrucciones de su mapa de bits. Cada línea recorrida por el rayo de electrones es un barrido. Cuando un barrido ha sido pintado, el rayo vuelve al lado opuesto de la pantalla y empieza a recorrer la siguiente línea, o barrido, inmediatamente debajo del anterior. Los barridos se van encimando para formar una imagen completa.

3.7.2 Gráficos basados en objetos

Una imagen, que puede ser descrita con una ecuación matemática, utiliza gráficos basados en objetos. Subyacente a la descripción de estos objetos, esta la matemática vectorial. Un vector es simplemente un cantidad multidimensional. Por ejemplo, la posición de un píxel individual en un mapa de bits puede ser dada por un vector bidimensional, como x,y . Un número corresponde a la fila donde esta el píxel, y el otro a la columna: es decir, 320,240, es la posición del píxel central en un despliegue VGA.

Las imágenes basadas en objetos se construyen a partir de elementos geométricos, tales como líneas, círculos, polígonos y otras curvas. Las líneas tienen puntos en sus extremos, mismos que pueden ser descritos por un vector. Las líneas también pueden tener un ancho, que puede contener patrones de relleno. Estos también pueden ser definidos por vectores. (los rellenos se generan con un algoritmo matemático para la localización de puntos.) En forma similar, los círculos tienen un centro (origen), un radio, un relleno y una línea limítrofe que también puede tener anchura y relleno. Un solo vector con suficientes dimensiones puede describir todo lo anterior. El programa que crea el objeto determinará qué partes del vector corresponden a las diferentes partes de un círculo basado en objetos, o a

Dispositivos Entrada/Salida y Tecnologías Gráficas

cualquier otro gráfico de éstos.

Formas aún mas complejas pueden ser descritas por vectores. Algunas técnicas utilizan las llamadas curvas de bézier. Estas son curvas no uniformes, que tienen tanto puntos terminales como puntos de inflexión; estos últimos describen cómo se tuerce la curva. Todos los puntos de este tipo de curvas se especifican con vectores. Las imágenes pueden ser rellenadas de una manera mas compleja, utilizando medios tonos; estos son patrones de puntos en la pantalla, usado para simular grises tersos.

Considere un ejemplo sencillo. El círculo del que hablábamos hace poco puede ser descrito con un vector de seis dimensiones. El vector daría la posición del origen del círculo (dos números), su radio, el patrón de relleno, el ancho de la línea que lo limita, y el color de relleno de dicha línea. Todo lo anterior podría ocupar 12 bytes, si calculamos dos bytes por número en el vector. El círculo equivalente en mapa de bits ocuparía mucho más espacio para almacenarse. Suponiendo que necesitamos dos bytes para describir cada pixel y que el círculo de dos pulgadas de ancho en la pantalla VGA, !requeriríamos de más de 32000 bytes para describirlo! Claramente la descripción basada en objetos es más económica.

Los gráficos vectoriales también tienen la propiedad de ser independientes del dispositivo: la calidad de la imagen desplegada es tan buena como puede realizarla el dispositivo junto con su manejador, utilizando la descripción original. El mismo círculo basado en objetos, desplegado en píxels de 72 dpi en un monitor, puede ser construido en píxels de 300 dpi en una impresora láser.

3.8 Formatos Gráficos

Los formatos de archivo de imágenes utilizan diversos esquemas para describir imágenes. Además de estar basados ya sea en mapas de bits o en gráficos parados en objetos, muchos formatos de archivo también especifican los esquemas de compresión de los archivos.

Los formatos de archivos gráficos han proliferado de tal manera que se han convertido casi en un tópico de pesadilla. Los fabricantes trabajan arduamente para maximizar sus estructuras de archivo para una mayor ventaja competitiva, pero el resultado puede ser frustrante si usted necesita utilizar esos archivos en

Dispositivos Entrada/Salida y Tecnologías Gráficas

otro programa. Por ejemplo, TIFF es un estándar, pero existen cuatro versiones por lo menos distintas! A continuación se presenta información general para los formatos gráficos mas ampliamente usados.

3.8.1 Formatos en mapas de bits.

3.8.1.1 TIFF

El formato de mapas de bits Tagged Image File Format (TIFF) es el mas ampliamente utilizado. TIFF se encuentra frecuentemente en aplicaciones digitales de formación de imágenes, como los rastreadores (scanners). La extensión usada comúnmente es .TIF. Desarrollado por Aldus y Microsoft, la versión actual de TIFF es la versión 5.0. Esta especificación puede contener una escala de grises de 8 bits (256 tonos de grises), imágenes de color de hasta 24 bits, y naturalmente blanco y negro. Los archivos TIFF son normalmente bastante voluminoso: una imagen de 24 bits de 8.5 por 11 pulgadas puede alcanzar 6.2 MB.

Existen varias versiones de TIFF. Hay el TIFF monocromatico (blanco y negro), TIFF de escala de grises (soportando 256 niveles de grises, o sea 8 bits) y TIFF de color. Este ultimo soporta imágenes de 32 bits de color. Para hacer todo aun mas confuso, el estándar TIFF permite el uso de algoritmos de compresión exclusivos. Existe el TIFF sin comprimir, el tiff comprimido y el TIFF en PackBits.

TIFF tiene reputación de introducir cambios sutiles en la calidad y tono de la imagen y debe ser usado con cuidado y con precaución.

3.8.1.2 PCX

Este es el formato que esta en segundo lugar de utilización en el caso de formatos para imágenes en mapas de bits. Este formato fue creado por ZSoft para utilización de sus programas Paintbrush y Publisher Paintbrush para PC. Paint de Microsoft (que viene con Windows), que es una versión menos potente de esos programas, también puede usar archivos PCX. La mayoría de los rastreadores (scanner) pueden también crear en este formato, y asimismo, casi todos los programas de gráficos lo leen y lo pueden escribir. En su especificación original, PCX era un formato en blanco y negro, solamente útil para arte lineal. Una

Dispositivos Entrada/Salida y Tecnologías Gráficas

extensión de formato soporta hoy en día escala de grises de 8 bits y color. El formato PCX es famoso por el tamaño reducido de sus archivos de mapas de bits.

3.8.1.3 FAX

Puede sorprenderle el saber que los facsímiles transmitidos utilizan un formato de mapa de bits estándar llamado FAX (filename.FAX). El estándar actual de Grupo III creado por la CCITT es un mapa de bits en blanco y negro y tiene una resolución de 200x100 dpi. En alta resolución, se obtiene 200x200 dpi. El estándar FAX utiliza una rutina RLE de compresión (run length encoded), afin de reducir el tamaño de los archivos, así como reducir el tiempo de transmisión. Dados los tiempos requeridos para transmitir una página de fax, este exitoso.

Aunque eminentemente útil, FAX es realmente un método poco eficiente para transmitir texto. Partiendo de información compacta, basada en caracteres ASCII, la toma y al convierte en una mapa de bits. Un documento de una página podría tener quizá, solamente mil caracteres ASCII (aproximadamente 1 KB), pero tomara 200 KB para describirlo como un mapa de bits FAX. Peor aun, en comparación con otros estandares de mapas de bits, FAX es bastante tosco. CCITT ya esta trabajando en un standard de grupo IV. Entre otras cosas, incluirá soporte de escala de grises y una resolución mas alta. Pero debido al enorme universo de equipo FAX existente, que no será compatible con el nuevo estándar, tomara mucho tiempo antes de que el nuevo estándar de alta calidad se puede poner en uso.

3.8.2 Formatos basados en objetos

3.8.2.1 PostScript

Muchos programas le permiten generar como una opción un archivo de texto puramente PostScript. Los archivos PostScript pueden ser abiertos e impresos con cualquier programa que pueda leer texto y pueda trabajar con el lenguaje PostScript.

3.8.2.2 CGM

Dispositivos Entrada/Salida y Tecnologías Gráficas

Computer Graphics Metafile (filename.CGM), un metaarchivo es una secuencia de llamados gráficos a un conjunto de rutinas. El metaarchivo almacena la secuencia de llamadas en vez de la imagen de dibujo de un círculo. Esta forma de trabajar es aun mas compacta que la descripción de la ecuación que define el círculo. CGM tiene descripciones de archivo muy compactas y soporta un amplio rango de colores.

3.8.3 Otros formatos vectoriales

Existe otros dos formatos de archivo basado en objetos claramente definidos y que merecen ser mencionados. Son Data Exchange Format(filename.DXF) de AutoCAD y Graphic Language Format(HPGL) de Hewlett-Packard. Los archivos HPGL utilizan la extensión filename .PLT

CONCLUSIONES

Los datos gráficos son guardados, y manipulados para producir imágenes, también sabemos que existen diferentes aplicaciones que usan una gran variedad de datos en diferentes formas para satisfacer diferentes necesidades.

En este capítulo se dieron los detalles mas sobresalientes de los formatos gráficos mas importantes, se pretende que con esta información y dependiendo la aplicación a realizar se este en la capacidad de seleccionar el formato grafico mas adecuado a la aplicación deseada.

REFERENCIAS BIBLIOGRAFICAS

Brown Wayne C. and J. Shepherd Barry. Graphics File Formats,
Primera edición, Manning Publications Company 1995

Dewey R. Bruce. Computer Graphics for Engineers. Primera edición,
Harper & Row publishers, 1988

Norton Peter. Perifericos y Accesorios Para la IBM-PC, PC/2 y
compatibles. Primera edición, Prentice Hall 1992.

CAPITULO IV

FORMATO DE ARCHIVO PARA IMAGENES PCX

Formato de Archivo para Imágenes PCX.

Introducción

El procesado digital de imágenes comprende la manipulación de una imagen ya existente a una forma deseada. El primer paso es obtener dicha imagen. Esto pueda sonar muy obvio, pero no es tan simple como se oye. Se necesita un método simple para recuperar las imágenes y que además cumplan con los estándares de almacenamiento de datos. El formato PCX se ha convertido actualmente en un estándar soportado por múltiples programas de gestión de imágenes y figuras gráficas. Por ello es muy importante conocer los aspectos internos del formato PCX para ser capaz de reconocer e incluir en nuestro propios programas las imágenes PCX.

4.1 Estructura de Cabecera.

Los Archivo pcx comienzan con una cabecera de 128 (80h) bytes, que sigue la estructura mostrada en la siguiente figura.

Elemento de la Cabecera	Número de bytes	Descripción
00h(0)	1 byte	Marca PC Paintbrush 10(ah) = PC Paintbrush PCX
01h(1)	1 byte	Información sobre la versión 0 = versión 2.5 2 = versión 2.8 con paleta 3 = versión 2.8 sin paleta 5 = versión 3.0 con paleta
02h(2)	1 byte	Método de Codificación 1 = Codificación run-length
03h(3)	1 byte	Bits por pixel
04h(4)	2 bytes	Coordenada X esquina superior izquierda (suele ser 0) [x1]
06h(6)	2 bytes	Coordenada Y esquina superior izquierda (suele ser 0) [y1]

Formato de Archivo para Imágenes PCX.

08h(8)	2 bytes	Coordenada X esquina inferior derecha (ancho de la figura) [x2]
Ah(10)	2 bytes	Coordenada y esquina inferior derecha (altura de la figura) [y2]
0ch(12)	2 bytes	Resolución horizontal
0Eh(14)	2 bytes	Resolución vertical
10h(16)	48 bytes	Definición de la paleta de colores (vectores de 16 valores RGB)
40h(64)	1 bytes	(Reservado; normalmente cero)
41h(65)	1 byte	Numero de planos de colores en la imagen
42h(66)	2 bytes	Numero de bytes por línea de barrido
44h(68)	61 bytes	(reservados)
81h(129)	-----	Primer Carácter codificado de la imagen

Tabla 4.1 Elementos que forman la cabecera de los archivos PCX.

4.2 Detalle de cada uno de los elementos que forman la cabecera del archivo .PCX

El primer byte:

Es siempre el valor 10 (A hexadecimal) y sirve para identificar los ficheros .PCX

El segundo byte:

Indica la versión PC Paintbrush, pero también otro dato más importante. Cuando dicho byte posee el valor 5, el programa PC Paintbrush IV entiende que la paleta está activada (ON) y respeta los colores de la figura.

Formato de Archivo para Imágenes PCX.

Por el contrario, si posee el valor 3, se interpreta que la paleta está desactivada (OFF) y se ajustan los colores a la paleta actual. Pruebe a introducir el valor 3 y obtendrá resultados sorprendentes propios de un genio del impresionismo.

El tercer byte:

Posee el valor 1, indicando que el fichero está codificado según la técnica run-length orientada a byte que se explica más adelante.

El cuarto byte:

Offset 3, visualiza el número de bits por pixel o, lo que es lo mismo, el resultado de dividir entre ocho el número de pixels por bit.

A continuación se almacenan cuatro enteros (cada uno de dos bytes), que indican las coordenadas de la figura de la pantalla de PC Paintbrush. Los dados primeros son la fila y columna de la esquina superior izquierda (que suele ser 0,0 para los ficheros PCX y un valor distinto de cero para los ficheros PCC del Microsoft Pintbrush), y los dos siguientes enteros son la fila y columna de la esquina inferior derecha. Puesto que la esquina superior izquierda posee habitualmente el valor 0,0, las coordenadas de la esquina inferior derecha indican la altura y el ancho de la figura PCX. Recordase que la numeración de filas y líneas comienza en cero. Por tanto la altura de la figura PCX es igual a la coordenada Y de la esquina inferior derecha más 1.

En los bytes 12 (0C hex.) y 14 (E hexa.) se almacenan, dos enteros que indican, respectivamente, la resolución horizontal y vertical.

Luego aparecen 48 bytes con información relativa a la paleta de colores de la figura. Estos 8 bytes se interpretan de distinta forma, según la tarjeta gráfica utilizada.

En una tarjeta EGA los datos se almacenan como 16 triples. Cada triple es un valor de tres bytes que indica la cantidad de RGB (rojo, verde y azul). Los valores oscilan en el rango 0..255. En una tarjeta EGA hay cuatro niveles posibles de rgb para cada color. Por tanto, ya que $255/4=64$, un valor entre 0..63 define el primer nivel, un valor entre 64..127 define el segundo nivel,

etc.

En una tarjeta CGA la paleta se define con un método un poco más complejo. Sólo se utiliza el primer byte de cada triple de la siguiente forma. El primer byte del primer triple representa el color de fondo (background), que se halla tomando el valor del byte (sin signo) y dividiendo entre 16 para hallar un número entre 0 y 15. El primer byte del segundo triple representa el color de la figura (foreground). Paintbrush soporta 8 posibles paletas CGA, por tanto, el color de la figura se codifica como un número entre 0..255 con 8 rangos de números cada 32 (0..31 primera paleta, 32..63 segunda paleta, etc).

La información de la paleta almacenada en los 48 bytes no es significativa cuando se trata de figuras de 256 colores, apareciendo a partir del offset 20h la letra de la unidad en la cual se grabó el fichero y en el 24h la letra de la unidad en la cual se grabó el fichero y en el 24h la trayectoria absoluta (path) con la que se grabó el fichero.

Por último, en la cabecera quedan cuatro bytes más con información. El byte situado con el desplazamiento 40h está reservado. El byte siguiente indica el número de planos de colores. Conociendo el número de planos y el número de bits por pixel, se puede determinar el número de colores según la siguiente tabla:

Bit por pixel	Planos de colores	No de colores
1	1	Blanco y negro
1	2	4
1	3	8
1	4	16
2	1	4
2	1	256
4	1	16
8	1	256

Tabla 4.2 Número de colores a partir de los bits por pixel y el número de planos

Formato de Archivo para Imágenes PCX.

4.3 Estructura de los datos.

A partir del desplazamiento u offset 129 (81h) se almacenan los bytes que forman la imagen del fichero PCX. Es importante distinguir dos cosas:

- La forma como se almacenan los bytes en el fichero según su disposición en el buffer de video.
- El método de codificación empleado para almacenar los bytes.

El formato de almacenamiento de los bytes de la imagen depende del número de planos de color y del número de bytes por línea. Cuando hay un único plano, los bytes se almacenan secuencialmente según el número de línea de barrido de la imagen, como se puede ver en la siguiente imagen:

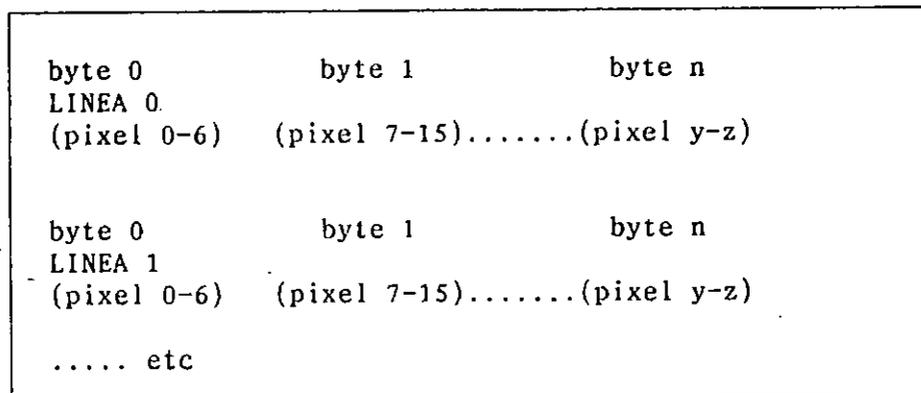


Tabla 4.3 Almacenamiento de los bytes con un sólo plano de color

Pero cuando hay más de un plano de color, cada línea de la figura se almacena en base a los planos de colores (ordenados normalmente en el formato RGBI:red, green, blue, intensity).

Por ejemplo: Una figura con cuatro planos de color y 6 bytes por línea se almacenaría tal y como muestra la siguiente figura, donde cada una de las letras R, G, B e I representan un byte.

Formato de Archivo para Imágenes PCX.

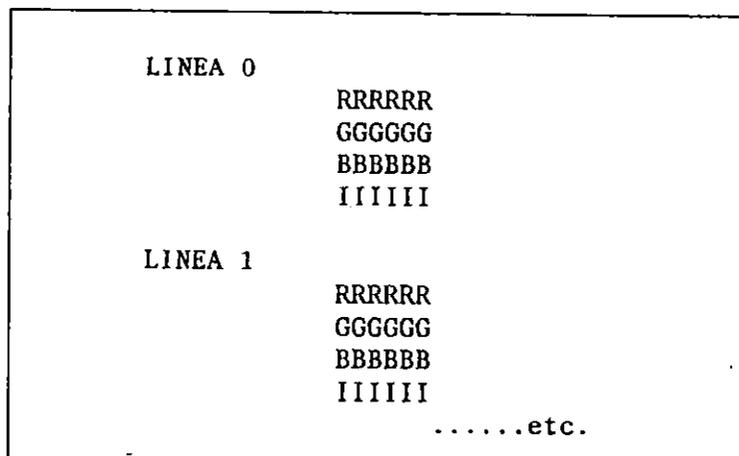


Tabla 4.4 Almacenamiento de los bytes con varios planos de color.

4.4. Técnicas de Codificación Run-length orientada a bytes

Otro asunto distinto es el método de codificación de los bytes. Los bytes no se graban en el Archivo tal y como aparecen en el buffer de video, sino codificados mediante una técnica run-length orientada a bytes.

La técnica de codificación es muy sencilla. Cuando hay más de un byte seguido con el mismo valor, se almacena únicamente dos bytes (contador y dato): el primer byte indica cuantas veces se repite el carácter a repetir. El contador comienza con /c0h; es decir c5h esta expresado que se repite 5 veces el byte situado a continuación, por ejemplo:

LINEA 0	
AA AA AA AA AA 55 55 AA AA AA	R
00 00 00 00 12 00 00 00 00 00	G
FF FF FF 43 43 43 43 43 43 43	B
AA AA FE AA AA AA AA AA AA AA	I
LINEA 0 CODIFICADA	
C5 AA C2 55 C3 AA	
C4 00 12 C5 00	
C3 FF C7 43	
C2 AA C1 FE C7 AA	

Tabla 4.5 Ejemplo de codificación según la técnica de los ficheros PCX.

Formato de Archivo para Imágenes PCX.

4.5 Técnicas para decodificar archivos PCX

El pseudocódigo para codificar una imagen PCX podría ser el siguiente:

Pseudocódigo para decodificar un archivo PCX

```
for cada byte x que se lee del fichero
  if los dos bits más significativos de x están a 1
    contador=los seis bits menos significativos de x
    dato =byte siguiente a x
  else
    contador=1
    dato=x
```

Se advierte que cuando hay un byte que se repite una sola vez, se puede añadir o no el contador C1h.

A continuación se presenta una función de lenguaje C para decodificar un fichero PCX:

```
#include <dos.h>

int decodifica (int *pdato, int *pcont, FILE *pfichero)
{
  int i;
  *pcont =1;
  if (EOF == (i=getc(pfichero)))
    return(EOF);
  if (0xC == (0xC0 & i))
  {
    *pcont=0x3F &i;
    if (EOF == (i=getc(pfichero)))
      return(EOF);
  }
  *pdato=i;
  return (0);
}
```

A continuación se presenta una función en lenguaje C para el almacenamiento en un buffer de memoria la imagen decodificada del fichero PCX.

Formato de Archivo para Imágenes PCX.

```
/* A continuación se muestra una rutina que utiliza la */
/* función decodifica() para almacenar en Buffer los */
/* bytes del fichero decodificados */

while( EOF != decodifica(&carácter, &cont, fichero))
{
for (i=0;i<cnt;i++)
*buffer++=carácter;
}
```

El mejor método para comprobar la información de los ficheros PCX, es diseñar alguna utilidad para dichos ficheros. Por ejemplo en la siguiente tabla aparece un pequeño programa en lenguaje C que visualiza los datos más importantes de la cabecera de un fichero PCX.

```
#include "pcx.h"

void main(int nparam, char *param[])
{
    if (nparam <= 1 )
        { printf("\nNecesito Un nombre de Archivo");
          exit(1);
        }
   strupr(param[1]);
    displaypcxinfo(param[1]);
    getch();
}
```

Tabla 4.6 Programa que despliega la información contenida en la cabecera de un archivo PCX.

Este programa incluye el archivo cabecera pcx.h dentro del cual están definidos todos los datos que presenta la cabecera de los archivos pcx, la forma de correr este programa es muy sencilla, una vez tengamos el archivo ejecutable basta con escribir el nombre del programa y luego el nombre del archivo PCX. El nombre del archivo del cual queremos ver la información, se pasa a través de la función main, en caso de no pasar el parámetro el programa da una salida al DOS.

Formato de Archivo para Imágenes PCX.

Por ejemplo si el programa anterior tiene el nombre de `verpcx.exe`, y nosotros queremos ver la información del archivo `hg3.pcx` (que es la presentación de Harvard Graphics), el cual se encuentra en el directorio `hg3`, basta con escribir:

```
verpcx c:\hg3\hg3.pcx
```

Este programa, cuya salida se parecerá a la posteriormente mostrada, emula la utilidad `PCXHDR` incluida en el programa `Paintbrush IV`.

File : \hg3\hg3.pcx, size 53676	
Version	: version 3.0 con paleta
Compresion	: RLE-Compressed
Upper Left Corner	: (0 , 0)
Lower Right Corner	: (639 , 349)
Resolution	: (640 , 350)
Bit per pixel	: 1
Bytes per line	: 80
Numb. Colour Planes	: 4
Paleta de Colores	
[0]-(0)(0)(0)	[1]-(167)(83)(83)
[2]-(255)(255)(179)	[3]-(255)(255)(255)
[4]-(255)(0)(255)	[5]-(255)(0)(255)
[6]-(91)(0)(7)	[7]-(171)(171)(171)
[8]-(255)(0)(255)	[9]-(255)(27)(255)
[10]-(255)(0)(255)	[11]-(0)(83)(83)
[12]-(255)(0)(255)	[13]-(255)(85)(255)
[14]-(255)(0)(255)	[15]-(255)(255)(255)

Tabla 4.7 Lectura de la Cabecera de el archivo HG3.PCX

El archivo cabecera `pcx.h` se explicara en detalle a continuación también se incluye en el anexo I.

4.6 La utilidad `pcx.h`

El archivo cabecera `PCX.H`, contiene cuatro funciones principales que son de gran ayuda para leer archivos `pcx.h`. La estructura principal en la cual estan definidos todos los

Formato de Archivo para Imágenes PCX.

elementos que forman la cabecera se muestran a continuación.

```
typedef struct
{
    unsigned char id,version,compr,bitstperpixel;
    unsigned int xmin,ymin,xmax,ymax;
    unsigned int horidpi,vertdpi;
    unsigned char colormap[16][3];
    unsigned char reserved;
    unsigned char ncolplanes;
    unsigned int bytesperline;
    unsigned int greyscale;
    unsigned char filler[58];
}PCX ;
```

Tabla 4.8 Estructura que contiene todos los elementos de la cabecera de un archivo PCX

Luego de definir la estructura de datos, se necesita saber si el archivo cuya cabecera deseamos conocer realmente existe, de no existir dicho archivo se obtendra una salida al D.O.S., la base de esta función es la estructura `ffblk` del `borlandc`.

```
int assignfname(char *dest,char *src )
{
    struct ffbk ffbk;
    char drive[MAXDRIVE];
    char dir[MAXDIR];
    char file[MAXFILE];
    char ext[MAXEXT];
    strcpy(dest,src);
    if (!(fnsplit(dest,drive,dir,file,ext) & EXTENSION))
        strcat(dest,".PCX");
    if (findfirst(dest,&ffbk,0))
    {
        pcxerror = "No existe tal archivo";
        return 1;
    }
    filesize = ffbk.ff_fsize;
    return 0;
}
```

Tabla 4.9 Funcion que verifica la existencia de un archivo.

Formato de Archivo para Imágenes PCX.

Una vez verificada la existencia del archivo se procede a leer los primeros 128 bytes, de este o sea la cabecera del archivo, la función `readpcxfile`, hace uso de la función `assignfname` para verificar primero la existencia del archivo. Si la operación de lectura de la cabecera se realizó correctamente la función retorna un 1 si no un 0.

```
int readpcxfile(char *fpcxname,PCX *pcx)
{ FILE *fptr;
  if (assignfname(filename,fpcxname))
    return 1;
  if ((fptr = fopen(filename,"rb")) == NULL)
  { pcxerror="No pude abrir el archivo";
    return 1;
  }
  if ( fread(pcx,1,sizeof(PCX),fptr) !=sizeof(PCX) )
  { pcxerror="Eof encontrado prematuramente";
    return 1;
  }
  if( verifypcxheader(pcx))
  { pcxerror="No es archive pcx";
    return 1;
  }
  fclose(fptr);
  return 0 ;
}
```

Tabla 4.10 Función que lee la cabecera de un archivo PCX.

El siguiente paso es determinar si el archivo es PCX, la función `verifypcxheader`, verifica los primeros dos elementos de la cabecera los cuales determinan la marca y la versión.

Formato de Archivo para Imágenes PCX.

```
int verifypcxheader(PCX *pcx)
{  if(pcx->id != '\n')
    {    pcxerror ="Header no es PCX";
      return 1;
    }
  if(pcx->version!=5 || pcx->compr > 1 )
  {  pcxerror ="Codificación desconocida";
    return 1;
  }
  if(pcx->ncolplanes > 4 )
  {    pcxerror ="Solo información de Planos";
    return 1;
  }
  if( pcx->greyscale != 2 ) pcx->greyscale =1;
  return 0;
}
```

Tabla 4.11 Función que verifica la marca y versión de un archivo PCX

Habiendo verificado que el archivo seleccionado es un archivo PCX, de versión conocida el siguiente paso es desplegar toda la información contenida en su cabecera. La función `displayinfo` se encarga de realizar dicha tarea.

Formato de Archivo para Imágenes PCX.

```
displaypcxinfo(char * fname)
{ PCX pcx;
  char *s,*t,i,*ver;
  if( readpcxfile(fname,&pcx) )return 1;
  switch ( pcx.compr )
  {   case 0: s="Un-Compressed";break;
      case 1: s="RLE-Compressed";break;
      default: s="Unknown compression method";
  }
  if( pcx.greyscale == 2) t = "Display as grey scales";
  else t = "Display as Colour rsp.";
  clrscr();
  printf("File : %s, Size: %ld",fname,filesize);
  printf("\n-----");
  switch(pcx.version)
  {   case 0: ver ="Versión 2.5";break;
      case 2: ver ="Versión 2.8 con paleta";break;
      case 3: ver ="Versión 2.8 sin paleta";break;
      case 5: ver ="Versión 3.0 con paleta";break;
      default: ver ="Desconocida";break;
  }
  printf("\nVersion          : %s",ver);
  printf("\nCompresion           : %s",s);
  printf("\nUpper Left corner: ( %3d , %3d )",pcx.xmin,pcx.ymin);
  printf("\nLower Right corner: ( %3d , %3d )",pcx.xmax,pcx.ymax);
  printf("\nResolution: Hz      : ( %3d , %3d
)",pcx.horidpi,pcx.vertdpi);
  printf("\nBits per Pixel      : %3d",pcx.bitsperpixel);
  printf("\nBytes per Line      : %3d",pcx.bytesperline);
  printf("\nNumb.Colour planes: %3d",pcx.ncolplanes);
  printf("\n%s",t);
  printf("\n\nPaleta de Colores:");
  for( i = 0; i< 16;i++)
  {   if( i%2 )
      printf("\t\t");
      else
      printf("\n");
      printf("[%2d]-(%3d)(%3d)(%3d)",i,pcx.colormap[i][0],pcx.colormap[i][1],pcx.colormap[i][2]);
  }
  return 0;
}
```

Tabla 4.12 Función que despliega en pantalla el contenido de una Cabecera PCX.

Formato de Archivo para Imágenes PCX.

4.7 Leyendo una Imagen PCX

Después de haber verificado la cabecera de un archivo PCX el paso mas importante es tener la capacidad de enviar a video dicha imagen, a continuación se presentaran una serie de funciones cuya capacidad es presentar en pantalla el fichero PCX, dichas funciones también aparecen en el anexo I, como un archivo completo. Es de notar que el programa incluye el archivo RPCX.C cuyo código fuente se encuentra en el anexo I.

```
/****** VIEWPCX.C ***** */
/*      RECÚPERA GRAFICOS EN FORMATO PCX      */
/****** */
#include "rpcx.c"

void main(int nparam, char *param[])
{ unsigned char pal[16][3];

  if (nparam <= 1 )
  {printf("\nNecesito Un nombre de Archivo");
  exit(1);
  }
  strupr(param[1]);
  viewpcxfile( param[1] );
  if (videomode != 3 )
  setvideomode(3);
}
```

Tabla 4.13 Programa para leer un archivo PCX

4.8 Escribiendo un archivo PCX.

Es necesario en muchas aplicaciones de ingeniería el poder darle un formato estandar a muchos archivos de imágenes, por lo que también se presentan en este capítulo funciones cuya capacidad es darle a las imágenes el formato PCX, el programa principal se muestra en la tabla 4.14

Formato de Archivo para Imágenes PCX.

```
/* ***** VIEWPCX.C ***** */
/*          ALMACENA GRAFICOS EN FORMATO PCX          */
/* ***** */
#include "rpcx.c"
#include "wpcx.c"

void main(int nparam, char *param[])
{ unsigned char pal[16][3];

  if (nparam <= 1 )
  { printf("\nNecesito Un nombre de Archivo");
    exit(1);
  }
 strupr(param[1]);
  evga_getrgbpalette(pal);
  viewpcxfile( param[1] );

  writepcxfile("PRUEBA.PCX");
  viewpcxfile("PRUEBA");

  evga_putrgbpalette(pal);
  if (videomode != 3 )
    setvideomode(3);
}
```

Tabla 4.14 Programa principal para leer y escribir archivos PCX

4.9 Leyendo y Escribiendo Imágenes Meteorológicas

A continuación se presenta la corrida de el programa de la tabla 4.14, pasando como parametro a la función main, una imagen meteorológica infrarroja correspondiente al sector noroeste de el meridiano occidental, recibida el 11 de Diciembre de 1995, dicha imagen cumple con el formato estandar PCX, y es por eso que el procesador de texto en el cual se escribio este trabajo no tiene problemas en presentar la imagen.

El programa primero muestra la información sobre la cabecera de el archivo como se muestra a continuación.

Formato de Archivo para Imágenes PCX.

File : \FILES\SAT\P11.pcx, size 172866

Version : version 3.0 con paleta
Compresion : RLE-Compressed
Upper Left Corner : (0 , 0)
Lower Right Corner : (639 , 679)
Resolution : (640 , 480)
Bit per pixel : 1
Bytes per line : 80
Numb. Colour Planes : 4

Paleta de Colores

[0]-(0)(0)(0)	[1]-(16)(16)(16)
[2]-(32)(32)(32)	[3]-(48)(48)(48)
[4]-(64)(64)(64)	[5]-(80)(80)(80)
[6]-(96)(96)(96)	[7]-(112)(112)(112)
[8]-(128)(128)(128)	[9]-(144)(144)(144)
[10]-(160)(160)(160)	[11]-(176)(176)(176)
[12]-(192)(192)(192)	[13]-(208)(208)(208)
[14]-(224)(224)(224)	[15]-(240)(240)(240)

Despues de pulsar cualquier tecla el programa muestra la imagen siguiente:

Formato de Archivo para Imágenes PCX



Figura 4.1 Imagen Meteorológica cuyo formato es PCX.

CONCLUSIONES

El formato PCX se ha convertido en el estándar para muchas aplicaciones, que involucran el procesamiento de imágenes, es de aclarar que existen otros formatos de archivos que aventajan al PCX en lo que a procesado de imágenes meteorológicas se refiere, y que este capítulo solamente habré las puertas para una nuevas areas de investigación las cuales involucran: formatos de archivos gráficos, compresión de imágenes, mapeado de imágenes, etc. Es de señalar también las virtudes del formato PCX, las cuales son sencillez y claridad de código.

Formato de Archivo para Imágenes PCX.

REFERENCIAS BIBLIOGRÁFICAS

Microsoft Product Support Hardware Group. PC Paintbrush & Frieze Graphics.

Papas Chris h. & Murray William . Borlandc C++ HandBook, Editorial McGraw-Hill 1991.

Schildt Herbert. Programación en Turbo C. Segunda edición, Editorial Mc Graw-Hill 1988.

Zsoft Corporation. Zsoft Technical Reference Manual

Zsoft Corporation. Paintbrush IV. Manual de usuario. Editorial anaya Multimedia

CAPITULO V
INTRODUCCION AL PROCESADO DIGITAL DE
IMAGENES

INTRODUCCION

El procesado digital de imágenes involucra la ejecución de operaciones de Entrada/Salida sobre disco y video, de imágenes previamente almacenadas y luego el resultado guardarlas en un archivo de salida. Este trabajo pretende hacer mas fácil esta tarea de manipulación de imágenes previamente grabadas.

La primer tarea en el procesado digital de imágenes es efectuar operaciones de Entrada/Salida en disco de las imágenes, para ello nos encontramos con la necesidad de tener un formato de archivo standard de almacenamiento de imágenes. El formato de archivo especifica como guardar la imagen y toda la información contenida en ella misma. El presente trabajo hace uso del formato PCX, aunque no se descarta la posibilidad que dentro de futuros trabajos se use otro tipo de formato de archivo como lo es el formato .TIF, en este trabajo se discute el formato PCX, en los anexos se muestra el código que lee y escribe archivos PCX.

La segunda tarea en el procesado digital de imágenes es la manipulación de imágenes.

La tarea final es guardar el resultado en disco o imprimir en papel la imagen ya procesada. Esto se logra creando rutinas capaces de guardar con los requerimientos, standard .PCX y .TIF.

5.1 Frecuencia espacial; y filtrado de imágenes

5.1.1 Frecuencia Espacial

Todas las imágenes y graficos contienen frecuencia espacial. Muchos de nosotros estamos familiarizados con algún tipo de frecuencia tal como la de 60 ciclos, de la red de baja tensión en nuestros hogares. El voltaje varia en el tiempo como una senoide, y la senoide completa un ciclo completo 60 veces por segundo. Una frecuencia de 60 Hz.

Las imágenes tienen una frecuencia espacial. El nivel de gris en la imagen varia en el espacio (no en el tiempo), por'ej. va arriba y abajo. La figura 5.1 muestra una imagen con baja frecuencia espacial. El nivel de gris es bajo en el borde izquierdo de la figura, permanece constante y luego cambia ha mas alto nivel de gris. El nivel de gris es bastante constante (Solamente un cambio en el espacio) de tal manera que podemos

5.1.2 Filtrado.

El filtrado es también un concepto común. Cuando nosotros ajustamos el bass y treble de un equipo de sonido, estamos filtrando la salida de ciertas frecuencias de audio y amplificando otras. Filtros paso-Alto pasan frecuencia altas y detienen frecuencias bajas. Filtros paso-bajo pasan frecuencias bajas y detienen frecuencias altas. De la misma manera nosotros podemos filtrar la frecuencia espacial en las imágenes. un filtro paso-Alto Amplificara o "pasara" cambios frecuentes, en niveles de gris y filtro paso bajo reducirá cambios frecuentes en niveles de gris.

Considere la naturaleza de un frecuente o repentino cambio en el nivel de gris, la figura 5.1 muestra una imagen con solamente un cambio en el nivel de gris. El cambio en el nivel de gris, es muy repentino (es un flanco). Un filtro paso alto pasara, amplificara,. Cambiara el borde, un filtro paso bajo tratara de remover el borde. En lugar de que bruscamente se cambia de un nivel de gris a otro, el filtro paso bajo producirá una gradual entonación entre los dos niveles. Los dos niveles de grises existirán pero la transición será mas lenta.

5.2 Aplicación de filtrado a imágenes.

El filtrado espacial de imágenes tiene muchas aplicaciones básicas dentro del procesado de imágenes. Dentro de las cuales destacan la remoción de ruido, suavizado. El ruido dentro de las imágenes usualmente aparece como nieve (Blanco o Negro) aleatoriamente esparcida sobre la imagen. Puntos (clavos) o muchos cambios bruscos, bordes angostos dentro de la imagen causan la nieve. Un filtro paso bajo suavizara y removerá frecuentes bordes bruscos.

5.3 Frecuencia vrs. Filtrado espacial

Considere Sonido y ruido variando en el dominio del tiempo. Una sinusoides pura completa un ciclo 1000 veces en un segundo luego es un tono de 1KHz. En el dominio de la frecuencia, esto es un solo valor. Para tener una salida filtrada, se multiplica

Procesado digital de Imágenes

decir que la figura tiene baja frecuencia espacial.

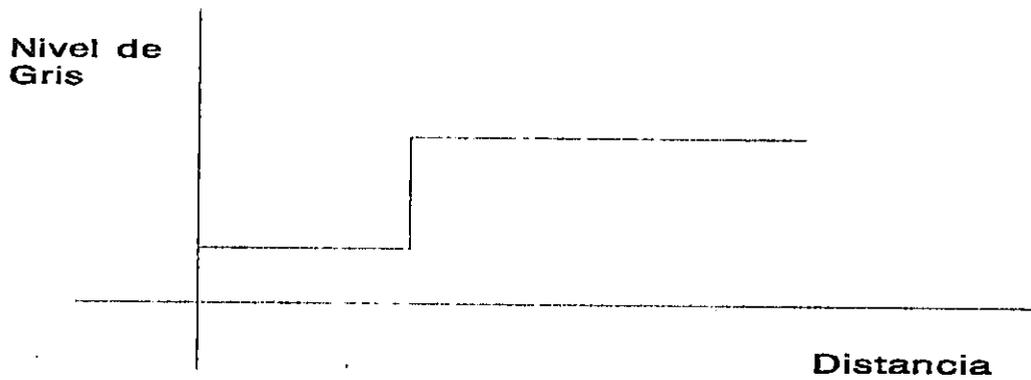


Figura 5.1. Muestra de una imagen con baja Frecuencia Espacial

La figura 5.2 Muestra una imagen con alta frecuencia espacial, El nivel de gris cambia muchas veces dentro del espacio de la imagen la velocidad o frecuencia de cambio dentro del espacio de la imagen es alto, de tal manera que la imagen tiene alta frecuencia espacial

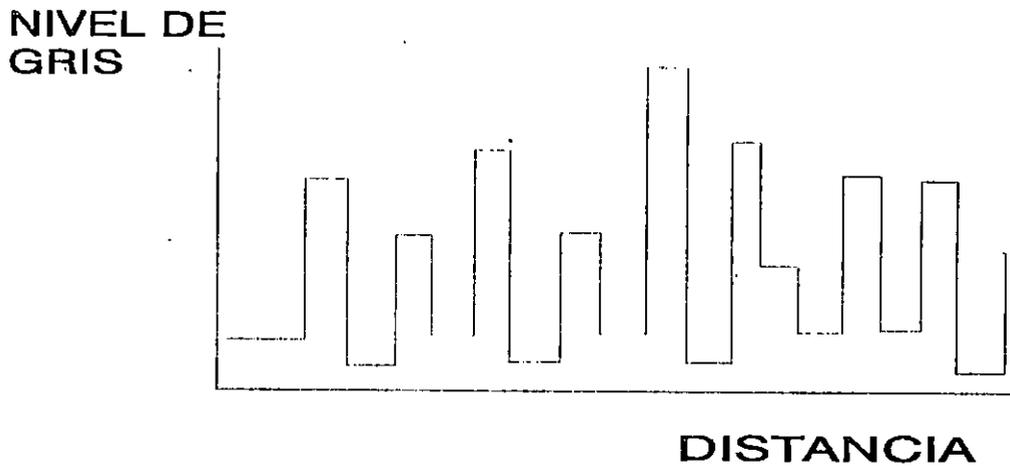


Figura 5.2. Imagen con Alta Frecuencia Espacial

por un filtro paso bajo que solo pasa frecuencias entre los 500 Hz.

Multiplicación en el dominio de la frecuencia es una tarea muy simple, de cualquier manera hay un problema. Se escucha el sonido dentro del dominio del tiempo, pero se debe transformar al dominio de la frecuencia antes de poder multiplicar. Esto se realiza utilizando la Transformada de Fourier, la cual requiere un cálculo substancial, y en algunos casos no es de mucho valor el esfuerzo.

Multiplicación en el dominio de la frecuencia corresponde a la convolución en el dominio del tiempo y el dominio espacial. Si nosotros usamos una pequeña máscara de convolución, tal como 3x3, luego la convolución de esta máscara sobre una imagen es mucho más fácil y más rápido que ejecutando la transformada de Fourier y multiplicación.

5.3.1 Filtros paso bajo

Los filtros paso bajo suavizan transiciones bruscas dentro de niveles de grises y remueven ruido. La tabla 5.1 muestran cuatro filtros paso bajo representados por su respectiva máscara de convolución. Convolucionando estos filtros con un área de nivel constante de gris de una imagen no cambiará nada. Note como la segunda máscara de convolución reemplaza el pixel del centro de la imagen de entrada con el promedio del nivel de gris de toda el área. Las otras tres máscaras tienen la misma forma general. Un valor alto en el centro y valores pequeños en las esquinas.

Máscaras de Convolución Para Filtros Paso Bajo			
	0 1 0		1 1 1
1/6*	1 2 1	1/10*	1 2 1
	0 1 0		1 1 1
	1 1 1		1 2 1
1/9*	1 1 1	1/16*	2 4 2
	1 1 1		1 2 1

Tabla 5.1 Máscaras de convolución utilizadas para realizar filtros paso bajo a Imágenes

Procesado digital de Imágenes

Las proximas cuatro tablas muestran ejemplos numéricos de como un filtro paso bajo afecta una imagen. La tabla 5.2 muestra un segmento de una imagen con baja frecuencia espacial. El segmento de la imagen cambia el nivel de gris una vez, pero con una transición brusca. La tabla 5.3 muestra un segmento de imagen con alta frecuencia espacial. Si se observa cuidadosamente el nivel de gris cambia por cada fila de la imagen, cada cambio es una transición brusca.

La tabla 5.4 muestra el resultado de convolucionar el primer filtro paso bajo de 3x3 de la tabla 5.1 con el segmento de imagen de la tabla dado en la tabla 5.2. El alto y Bajo Nivel de Gris permanece pero la transición difiere. El filtro paso bajo suaviza la transición desde una fila hasta la tercera fila de pixels.

La tabla 5.5 muestra el resultado de convolucionar el primer filtro paso bajo de 3x3 de la tabla 5.1 con el segmento de imagen dado en la tabla 5.3. La imagen resultante se muestra en la tabla 5.5. El segmento de imagen de la tabla 5.3 tiene transiciones en cada fila, el filtro paso bajo por lo tanto reduce la magnitud de las transiciones.

Dentro de una fotografía(podria ser una imagen meteorológica), el efecto de el filtro se vería como un oscurecimiento de la imagen comparada con la original

Segmento de Una Imagen con Baja Frecuencia Espacial						
150	150	150	150	150	150	150
150	150	150	150	150	150	150
150	150	150	150	150	150	150
150	150	150	150	150	150	150
150	150	150	150	150	150	150
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1

Tabla 5.2

Segmento de una Imagen con Alta Frecuencia Espacial						
150	150	150	150	150	150	150
1	1	1	1	1	1	1
150	150	150	150	150	150	150
1	1	1	1	1	1	1
150	150	150	150	150	150	150
1	1	1	1	1	1	1
150	150	150	150	150	150	150
1	1	1	1	1	1	1
150	150	150	150	150	150	150
1	1	1	1	1	1	1

Tabla 5.3

Procesado digital de Imágenes

Resultado de Aplicar, el filtro paso bajo(tabla 5.1)						
150	150	150	150	150	150	150
150	150	150	150	150	150	150
150	150	150	150	150	150	150
150	150	150	150	150	150	150
125	125	125	125	125	125	125
25	25	25	25	25	25	25
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1

Tabla 5.4

Resultado de Aplicar, el filtro paso bajo(tabla 5.1)						
100	100	100	100	100	100	100
50	50	50	50	50	50	50
100	100	100	100	100	100	100
50	50	50	50	50	50	50
100	100	100	100	100	100	100
50	50	50	50	50	50	50
100	100	100	100	100	100	100
50	50	50	50	50	50	50
100	100	100	100	100	100	100
50	50	50	50	50	50	50

Tabla 5.5

5.3.2 Implementacion de un filtro paso bajo

Para implementar el filtrado de imágenes se ha creado una cabecera, dentro de la cual están definidos cada una de las máscaras de convolucion, para los tipos de filtros paso bajo y paso alto.

El código fuente para los filtros se muestran a continuación. La primera seccion de código declara las cuatro máscaras que corresponde a su respectivos filtros paso bajo

Procesado digital de Imágenes

```
#define SHORT int
#define ROWS 430
#define COLS 615
#include <stdio.h>
#include <graphics.h>

SHORT lpf_filter_6[3][3] =
    { {0, 1, 0},
      {1, 2, 1},
      {0, 1, 0}};

SHORT lpf_filter_9[3][3] =
    { {1, 1, 1},
      {1, 1, 1},
      {1, 1, 1}};

SHORT lpf_filter_10[3][3] =
    { {1, 1, 1},
      {1, 2, 1},
      {1, 1, 1}};

SHORT lpf_filter_16[3][3] =
    { {1, 2, 1},
      {2, 4, 2},
      {1, 2, 1}};

SHORT lpf_filter_32[3][3] =
    { {1, 4, 1},
      {4, 12, 4},
      {1, 4, 1}};

void filtrado(SHORT filter[3][3],int d)
{ int max,i,j,a,b,sum;

max = 255;

    for(i=30; i<ROWS-1; i++){
        for(j=15; j<COLS-1; j++){
            sum = 0;
            for(a=-1; a<2; a++){
                for(b=-1; b<2; b++){
                    sum = sum + (getpixel(j+b,i+a)*filter[a+1][b+1]);
                }
            }
            sum = sum/d;
            if(sum < 0) sum = 0;
            if(sum > max) sum = max;
            putpixel(j,i,sum);
        } /* ends loop over j */
    } /* ends loop over i */
}
```

Tabla 5.6 Archivo que contiene las mascararas de convolución, que realizan filtrado de imágenes.

Procesado digital de Imágenes

5.3.3 Efectos de los filtros paso bajo en las imágenes Meteorológicas

A continuación se presenta una secuencia de imágenes meteorológicas del tipo infrarrojo correspondientes al cuadrante sureste, recibidas el día 12 de diciembre de 1995, la figura 5.3 muestra la imagen original, mientras que las figuras 5.4, 5.5, y 5.6 muestran el resultado de aplicar filtros paso bajo correspondiente a la tablas 5. 1, notese como el filtro remueve el ruido aleatorio.

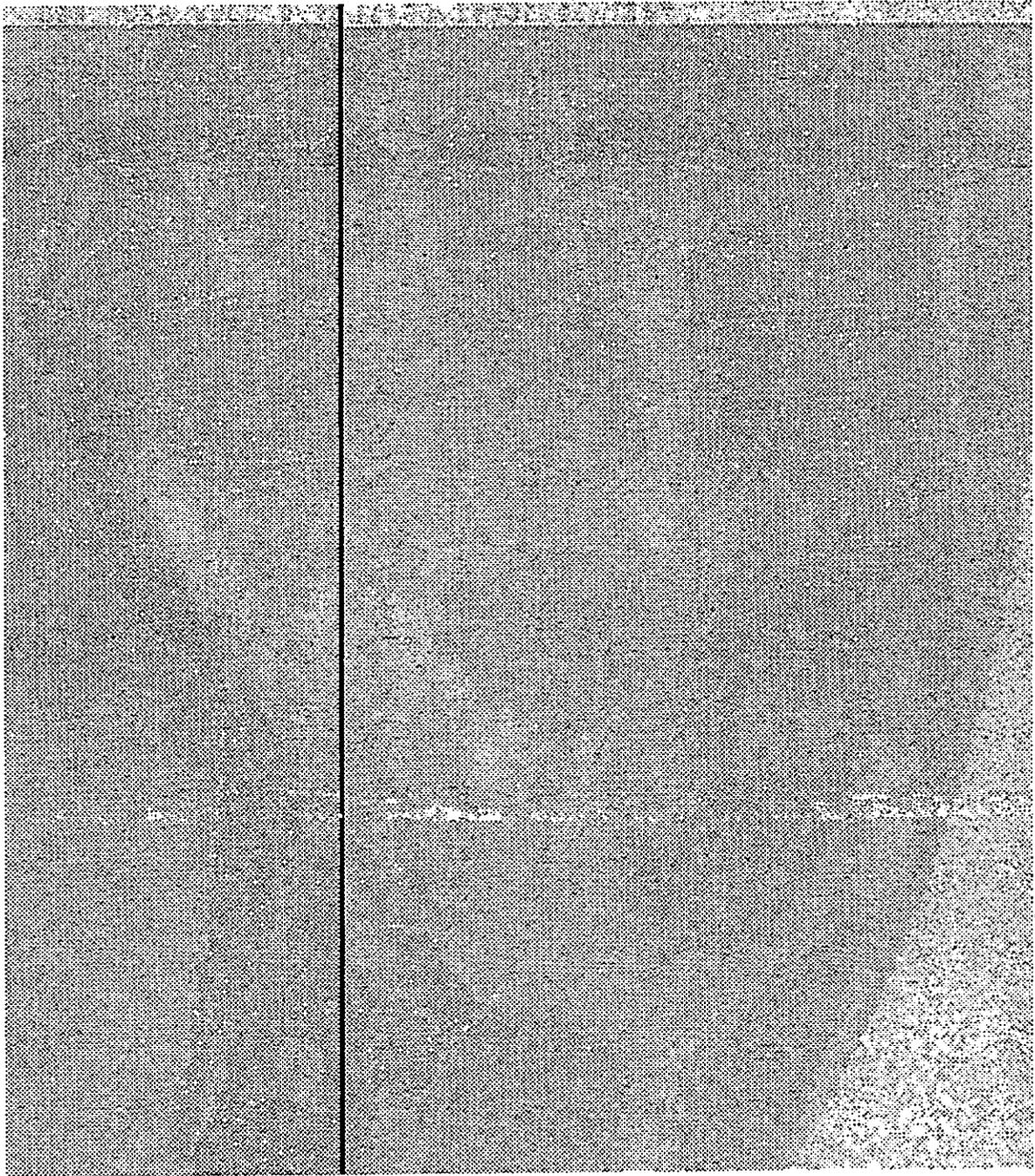


Figura 5.3 Imagen meteorológica (con ruido) del tipo infrarrojo, recibida el 12 de diciembre de 1995

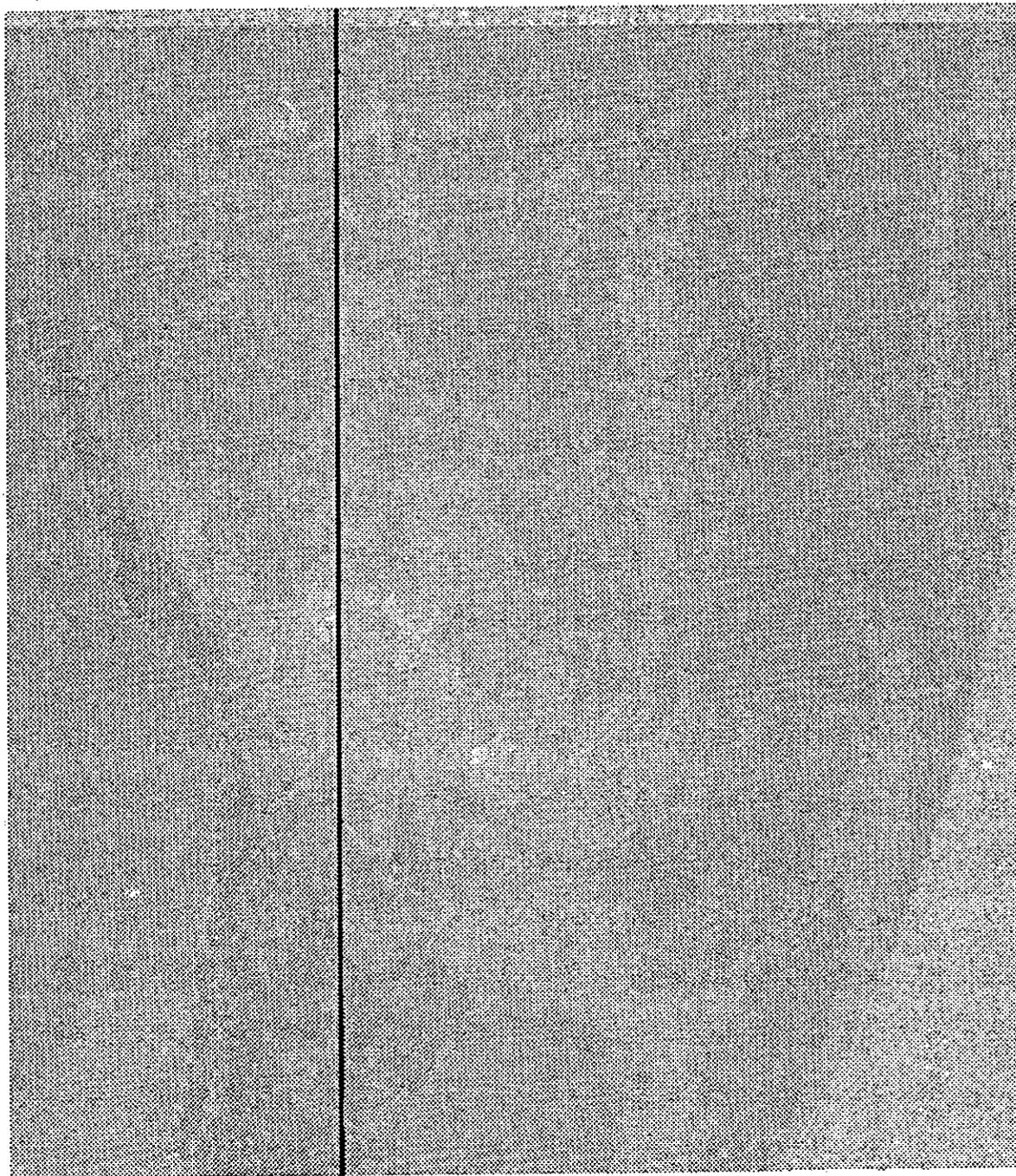


Figura 5.4 Resultado de aplicar filtro paso bajo mascara #6 a la figura 5.3



Figura 5.5 Resultado de aplicar filtro paso bajo máscara #9 a la figura 5.3

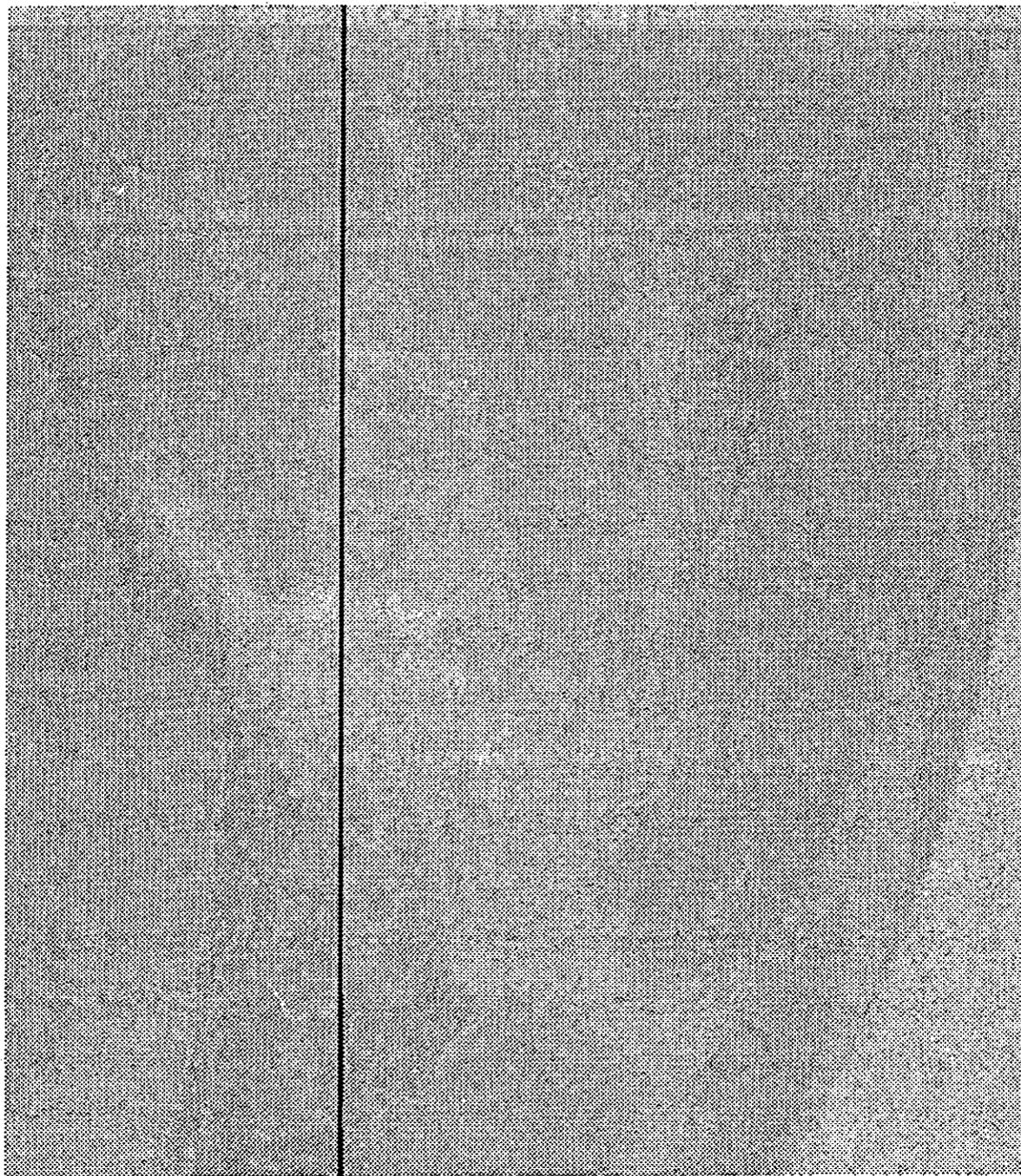


Figura 5.6 Resultado de aplicar filtro paso bajo máscara #32 a la figura 5.3

5.4 Operaciones con imágenes

Ya que el desarrollo de la presente Tesis de procesado de imágenes, esta orientada hacia el procesado de Imágenes Meteorológicas, Las operaciones con imágenes que se incluyen son: cortado, pegado de partes de imágenes, y Escalado de secciones de imágenes (Zoom).

5.4.1 Cortado y Pegado

Las funciones de cortado y pegado permiten cortar y pegar rectángulos de una imagen a otra. La tabla muestra un cortado y pegado; donde una sección de la imagen B es pegado sobre la Imagen A, esto se logra leyendo desde una imagen y pegando en la otra.

Cortado y Pegado. Una Sección de la Imagen B es pegada en la Imagen A											
Imagen A				Imagen B				Resultado			
1	2	3	4	0	1	0	1	1	2	3	4
5	6	7	8	0	1	0	1	5	1	0	1
9	10	11	12	0	1	0	1	9	1	0	1
13	14	15	16	0	1	0	1	13	1	0	1

Tabla 5.7 Cortado y Pegado de Imágenes

5.4.2 Escalado de Imágenes (Zoom)

Existen dos técnicas para realizar un engrandecimiento (zoom) de una sección de una imagen completa, estos métodos son el repetición por un factor y el método de interpolación. Estos métodos son ilustrados en la tabla 5.7 y la función que aplica este métodos se muestra en la tabla 5.8.

Ejemplo de Escalado de una Imagen(Zoom). Método de repetición									
Imagen de Entrada		Repetición 2x				Interpolación 2x			
1	4	1	1	4	4	1	2	4	4
6	9	1	1	4	4	3	5	6	6
		6	6	9	9	6	7	9	9
		6	6	9	9	6	7	9	9

Tabla 5.8

La función que realiza el escalado de imágenes se encuentra en el archivo SAT1.PCX (vease anexo), la figura 5.7 muestra una ampliación de un segmento de la figura 5.3, aplicando el metodo de repetición.

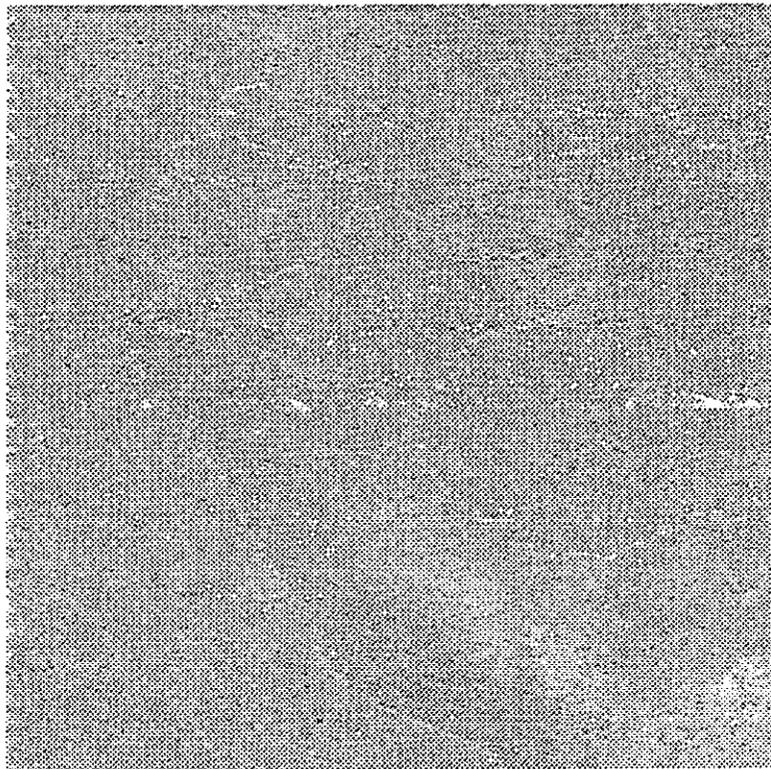


Figura 5.7 Ampliación de una sección de la figura 5.3. Tecnica de repetición

5.5 Visualización simultanea de varias imágenes

La visualización simultanea de imágenes es una operación sencilla sobre los archivos PCX, describiendo líneas y columnas podemos llegar a escribir en memoria de video 2, 4, 6 y mas imágenes, las cuales perderan resolución a medida se incrementa el numero de imágenes, las figuras 5.7 y 5.8 muestran algunas de las capacidades de el programa SAT (vease anexo), que es capaz de visualizar dos y cuatro imágenes simultaneamente.

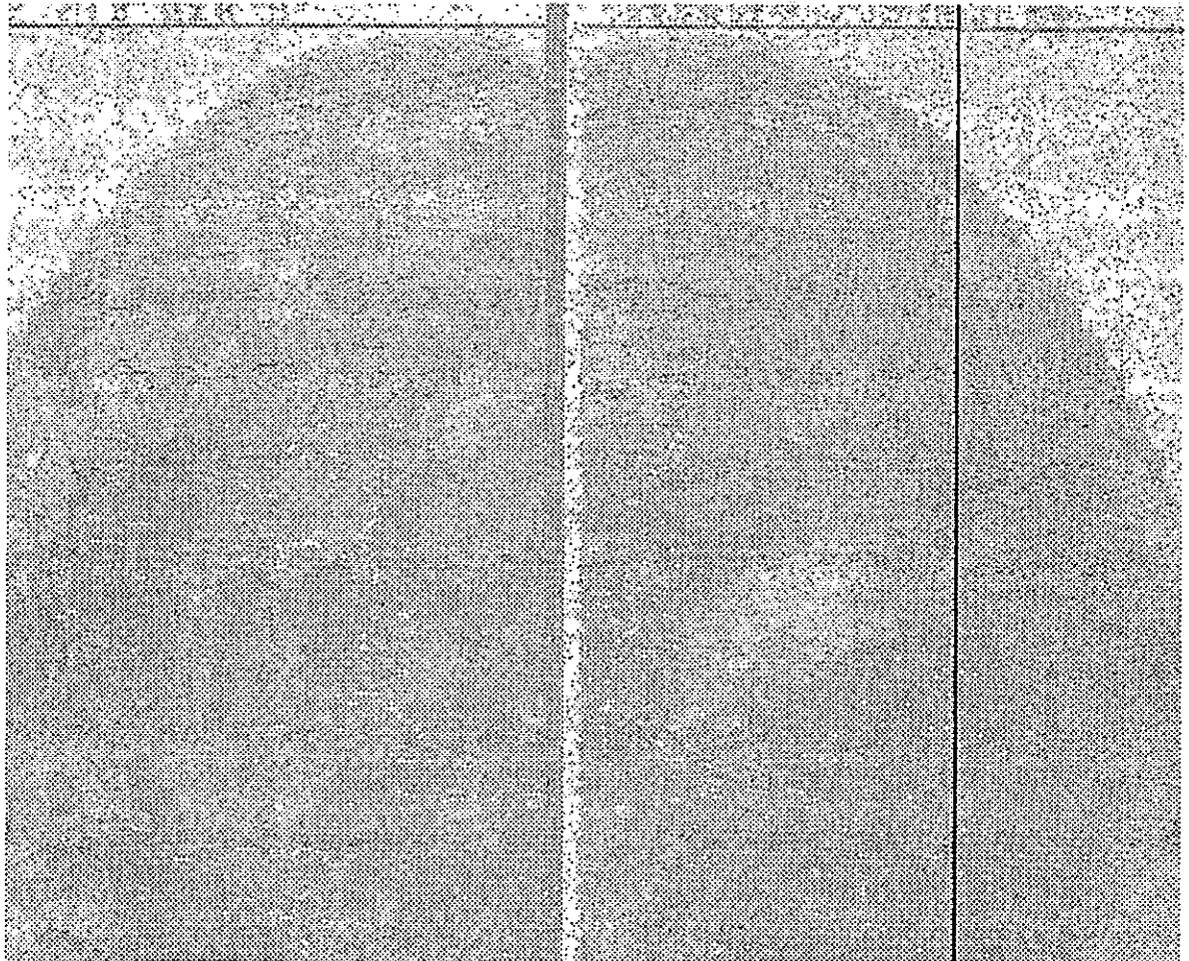


Figura 5.8 Visualización simultanea de dos imágenes meteorológicas

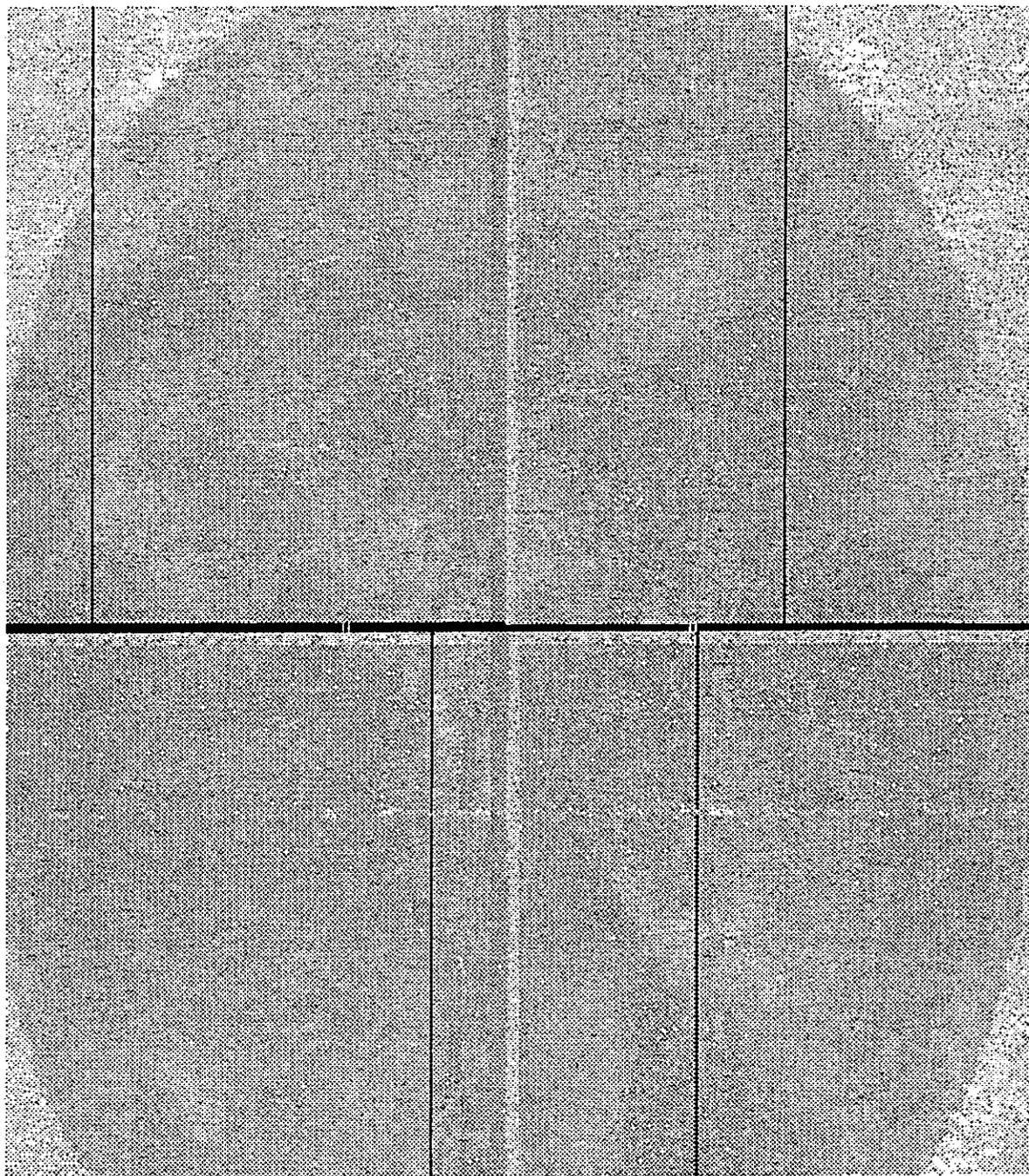


Figura 5.8 Visualización simultanea de cuatro imágenes meteorológicas.

5.6 Animación de imágenes meteorológicas

La animación de imágenes meteorológicas esta basada en escribir directamente a memoria de video, imágenes que corresponden a la misma posición geografica y tipo de imagen (sea esta en luz visible o infrarroja), en el anexo I, se encuentra la explicación de como usar el programa para realizar la animación de imágenes meteorológicas.

CONCLUSIONES

El procesado digital de imágenes es un área compleja, pero sus frutos son sorprendentes, la aplicación que tienen hacia las imágenes meteorológicas es muy importante ya que muchas de las imágenes no se reciben con la calidad deseada y un ruido está presente en ellas, luego de filtrarlas este ruido tiende a desvanecerse.

Las funciones de engrandecimiento le dan al programa, aun mas poder ya que aunque el engrandecimiento es en base a pixel, aun podemos ver con mayor claridad segmentos de imágenes que a simple vista no se podían visualizar.

REFERENCIAS BIBLIOGRÁFICAS

Brow C. Wayne and Shepherd Barry J. Graphics File Formats,
Manning Publications Co. 1995

Papas Chris h. & Murray William . Borlandc C++ HandBook,
Editorial McGraw-Hill 1991.

Phillips Dwayne. Image Processing in C., R & D Publications, Inc.
1994.

Schildt Herbert. Programación en Turbo C. Segunda edición,
Editorial Mc Graw-Hill 1988.

CONCLUSIONES GENERALES

1. A pesar de que el equipo receptor de imágenes meteorológicas es de tecnología bastante antigua, este permite ampliar sus capacidades de almacenamiento y tratamiento, y este trabajo exploto ese recurso.
2. Se experimento con diferentes técnicas de adquisición de datos, resultando mas efectiva la técnica de barrido de líneas de estado, es de aclarar que esta técnica no es la mas confiable pero que para esta aplicación en particular los resultados fueron satisfactorios.
3. El exagerado tamaño de las imágenes meteorológicas, hizo necesario crear rutinas que leen y escriben directamente a la memoria de video y los registros internos de la tarjeta controladora VGA, esto hace que el programa no sean tan versátil, necesitando obligatoriamente un adaptador de desplegué VGA.
4. El formato PCX se ha convertido en el estándar para muchas aplicaciones, que involucran el procesamiento de imágenes, es de aclarar que existen otros formatos de archivos que aventajan al PCX en lo que a procesado de imágenes meteorológicas se refiere, pero para esta aplicación en particular se obtuvieron muy buenos resultados.
5. Este trabajo abre las puertas a una nueva área de investigación, ya que es un tema de que esta dando mucho de que hablar en los diferentes departamentos de investigación de todo el mundo.

RECOMENDACIONES

1. Se obtuvieron excelentes resultados, en la adquisición y el procesado digital de imágenes, sin embargo no hay que sobreestimar estos resultados, lo que motiva a seguir investigando nuevas técnicas de adquisición de datos y procesado de los mismos.
2. Se debe seguir desarrollando el programa de adquisición y tratamiento digital de imágenes meteorológicas, seguido más de cerca por la división de meteorología e hidrología del ministerio de Agricultura y Ganadería.
3. Es importante para el desarrollo de el área de satélites meteorológicos, así como de el procesado digital de imágenes, el mantener comunicación con diferentes instituciones, dentro de las cuales destacan: la división de Meteorología e Hidrología del MAG, el departamento de comercio de los Estados Unidos, específicamente con la administración nacional de la atmósfera y los océanos.

ANEXO A

GUIA DE USUARIO DE EL PROGRAMA
DE ADQUISICION Y TRATAMIENTO
DIGITAL DE IMAGENES
METEOROLOGICAS

GUÍA DE USUARIO DE EL PROGRAMA DE ADQUISICIÓN Y TRATAMIENTO DIGITAL DE IMÁGENES METEOROLÓGICAS (SAT) Y ANIMACIÓN DE IMÁGENES METEOROLÓGICAS (ANIMATE)

A continuación se presenta una guía de usuario para la persona encargada de operar el equipo receptor de imágenes, y el programa software que amplía la capacidad de el equipo, el programa ha sido diseñado de tal manera que el usuario es orientado por un menú y un sub menú para cada comando (Menú desplegable tipo Pull-Down), a demás esta guía de usuario presenta la explicación de cada uno de los comando de el programa de adquisición de datos SAT (Sistema de Adquisición y Tratamiento Digital de Imágenes Meteorológicas).

El Entorno SAT

El propósito de este anexo es mostrar el entorno SAT, en otras palabras, conseguir que el usuario interactue con el programa.

Para instalar el programa se necesita crear un subdirectorio que se recomienda nombrar como SAT y copiar dentro de este los archivos SAT.EXE y el archivo EGAVGA.BGI de el compilador de borland C. Las características que debe tener el computador para que el programa funcione correctamente son: El monitor debe ser VGA a color 640x480, además la tarjeta para puerto paralelo debe de ser del tipo avanzado (ECP o EPP) ya que la adquisición de datos debe ser realiza a través de el puerto de la impresora, el computador debe tener como mínimo 1 Mbyte de memoria RAM ya que se necesita almacenar gran cantidad de información en tiempo real. Una vez que nuestro computador cumpla con estas especificaciones, puede realizarse la instalación sin ninguna preocupación.

Suponemos que se ha instalado apropiadamente y que el computador cumple los requerimientos hardware. Para entrar al entorno SAT, digitar SAT y pulsar la tecla de retorno, cuando el programa da inicio con una pantalla de presentación.

El programa espera a que se pulse cualquier tecla para continuar, al pulsar cualquier tecla aparece un menú tal como se muestra en la figura I.1, esta se le llama la pantalla principal del menú. El menú principal se usa para decirle al programa SAT que comando deseamos ejecutar, como por ejemplo leer imágenes, cargar imágenes, borrar imágenes, obtener una ayuda, etc. Cualquier comando que este dentro de el menú principal se selecciona moviendo la barra iluminada a cada comando, por

ejemplo si se quiere seleccionar el comando Ver debería moverse la barra iluminada hasta este comando y pulsar ENTER.

En la tabla numero 1 se resume cada uno de los comandos de el programa SAT.

ELEMENTO	DESCRIPCIÓN
ARCHIVO	Carga y Guarda Archivos, realiza adquisición de Imágenes, y sale al DOS.
EDITAR	Permite limpiar la pantalla de presentación de Imágenes, Cambiar la paleta de colores, etc.
FILTRADO	Realiza filtrado a las Imágenes previamente cargadas (Filtros paso Bajo y Paso Alto)
VER	Establece varias opciones para la visualización de Imágenes, Zoom in, Zoom Out, visualización simultanea de Varias Imágenes, etc.
AYUDA	Sistema de ayuda de el entorno SAT

Tabla N.1 Entorno SAT.



Figura I.1 Menú Principal de El programa SAT

ARCHIVO

Mover el cursor iluminado sobre el comando Archivo y presionar Enter. Este proceso activa el menú pull-down de Archivo. Para hacer una selección en un menú pull-down, se debe mover la barra iluminada a la opción y pulsar ENTER o pulsar simplemente ESC si deseamos abandonar el submenú. Veamos cada una de las opciones de Archivo.

La opción abrir espera el nombre de un archivo y entonces lo carga en la pantalla de presentación. La opción borrar permite eliminar o remover archivos previamente creados, por la opción leer, este proceso despliega un mensaje que pregunta cual archivo desea borrar y a la vez se muestra un directorio de los archivos existentes. Seleccionar Leer, este proceso espera el nombre de un archivo, después de esto el programa está chequeando continuamente los datos que se envían desde el equipo receptor de Imágenes meteorológicas los cuales son recibidos por el PC, estos datos son representados en pantalla y puede verse como la imagen va formándose línea a línea. Finalmente la opción SALIR, abandona el entorno SAT, y devuelve el control al DOS.

EDITAR

Mover la barra iluminada sobre la opción Editar y pulsar ENTER. La primera opción permite limpiar la pantalla de presentación. La segunda opción pretende resaltar zonas, de las imágenes infrarrojas, coloreándolas, para resaltar la información contenida en dichas imágenes, esta opción presenta una paleta de colores la cuales pueden cambiarse con las teclas cursoras, para desactivar el comando colores, se debe pulsar la tecla de Esc.

FILTRADO

La opción Filtrado permite acceder diferentes tipos de filtros paso bajo y paso alto, aplicados a Imágenes Meteorológicas, basta con mover el cursor iluminado y presionar ENTER al filtro que deseamos, aplicar a una imagen previamente cargada en la ventana de presentación.

VER

La opción Ver permite visualizar las imágenes meteorológicas de diferente manera. La primera opción es el comando Zoom Out, el cual permite visualizar, una Imagen completa(600 x 800), este comando permite realizar un desplazamiento de la imagen con las teclas de cursor (scroll), para desactivar este comando se debe pulsar la tecla de Escape. La segunda y tercera opción permite visualizar dos y cuatro imágenes simultáneamente respectivamente. El comando Zoom, permite realizar una ampliación de una imagen previamente cargada en la ventana de presentación.

AYUDA

Esta opción da una breve guía de como usar el programa, y es de mucha utilidad para recibir una orientación rápida.

PROGRAMA PARA ANIMACIÓN DE IMÁGENES METEOROLOGICAS (ANIMATE)

El programa para la animación de imágenes meteorológicas esta basado en el archivos cabecera creado para la lectura de archivos PCX, su principio esta basado en escribir muy rápidamente a memoria de video una secuencia de imágenes que correspondan a diferentes horas y ha diferentes días.

Para realizar la animación de una secuencia de Imágenes previamente almacenadas cuyos nombres sean p1, p2, p3, p4, etc., se hace lo siguiente:

```
animate p1 p2 p3 p4 p5 p6
```

El programa presenta la primera imagen y espera a que el usuario seleccione la parte de la imagen que se desea animar (recuerde que la resolución es de 600 x800 y no se puede visualizar toda la imagen), luego de seleccionar la imagen se debe presionar la tecla de escape y el programa estará escribiendo en memoria de video la secuencia de imágenes establecida, el programa realizara una salida al DOS hasta que se pulse cualquier tecla.

ANEXO B

**LISTADO DE EL ARCHIVO CABECERA
PCX.H**

```
#include <stdio.h>
#include <string.h>
#include <conio.h>
#include <dir.h>
```

```
typedef struct
{
    unsigned char id,version,compr,bitsperpixel;
    unsigned int xmin,ymin,xmax,ymax;
    unsigned int horidpi,vertdpi;
    unsigned char colormap[16][3];
    unsigned char reserved;
    unsigned char ncolplanes;
    unsigned int bytesperline;
    unsigned int greyscale;
    unsigned char filler[58];
}PCX ;
```

```
char *pcxerror;
```

```
long filesize;
char filename[80];
```

```
int assignfname(char *dest,char *src )
{
    struct ffblk ffblk;
    char drive[MAXDRIVE];
    char dir[MAXDIR];
    char file[MAXFILE];
    char ext[MAXEXT];
    strcpy(dest,src);
    if (!(fnsplit(dest,drive,dir,file,ext) & EXTENSION))
        strcat(dest, ".PCX");
    if (findfirst(dest,&ffblk,0))
    {
        pcxerror = "No existe tal archivo";
        return 1;
    }
    filesize = ffblk.ff_fsize;
    return 0;
}
```

```
int verifypcxheader(PCX *pcx)
{
    if (pcx->id != '\n')
    {
        pcxerror = "Header no es PCX";
        return 1;
    }
    if (pcx->version != 5 || pcx->compr > 1 )
    {
        pcxerror = "Codificación desconocida";
    }
}
```

```

        return 1;
    }
    if(pcx->ncolplanes > 4 )
    {
        pcxerror ="Solo información de Planos";
        return 1;
    }
    if( pcx->greyscale != 2 ) pcx->greyscale =1;
    return 0;
}

```

```

int readpcxfile(char *fpcxname,PCX *pcx)
{ FILE *fptr;
  if (assignfname(filename,fpcxname))
    return 1;
  if ((fptr = fopen(filename,"rb")) == NULL)
  { pcxerror ="No pude abrir el archivo";
    return 1;
  }
  if ( fread(pcx,1,sizeof(PCX),fptr) != sizeof(PCX) )
  { pcxerror="Eof encontrado prematuramente";
    return 1;
  }
  if( verifypcxheader(pcx))
  { pcxerror="No es archivo pcx";
    return 1;
  }
  fclose(fp);
  return 0 ;
}

```

```

displaypcxinfo(char * fname)
{ PCX pcx;
  char *s,*t,*i,*ver;
  if( readpcxfile(fname,&pcx) )
    return 1;
  switch ( pcx.compr )
  {
    case 0: s="Un-Compressed";break;
    case 1: s="RLE-Compressed";break;
    default: s="Unknown compression method";
  }
  if( pcx.greyscale == 2) t = "Display as grey scales";
  else t = "Display as Colour rsp.";
  clrscr();
  printf("File : %s, Size: %ld",fname,filesize);
  printf("\n-----");
  switch(pcx.version)
  {
    case 0: ver ="Versión 2.5";break;
    case 2: ver ="Versión 2.8 con paleta";break;
    case 3: ver ="Versión 2.8 sin paleta";break;
  }
}

```

```

        case 5: ver ="Versión 3.0 con paleta";break;
        default: ver ="Desconocida";break;
    }
    printf("\nVersion          : %s",ver);
    printf("\nCompresion           : %s",s);
    printf("\nUpper Left corner: ( %3d , %3d )",pcx.xmin,pcx.ymin);
    printf("\nLower Right corner: ( %3d , %3d )",pcx.xmax,pcx.ymax);
    printf("\nResolution: Hz       : ( %3d x %3d )",pcx.horidpi,pcx.vertdpi);
    printf("\nBits per Pixel       : %3d",pcx.bitsperpixel);
    printf("\nBytes per Line       : %3d",pcx.bytesperline);
    printf("\nNumb.Colour planes: %3d",pcx.ncolplanes);
    printf("\n%s",t);
    printf("\n\nPaleta de Colores:");
    for( i = 0; i< 16;i++)
    { if( i%2 )
        printf("\t\t");
        else
        printf("\n");
    }
    printf("[%2d]-(%3d)(%3d)(%3d)",i,pcx.colormap[i][0],pcx.colormap[i][1],pcx.colormap[i][2]);
}
return 0;
}

```

ANEXO C

LISTADO DE EL ARCHIVO RPCX.C

```

#include "pcx.h"

#define fbuffsize 1024*5
long filesize;
char filename[80];

int getlinepixel(FILE *fpcx, char bytesperline)
{ unsigned int c=0, rep=0, qued, i=0, tbytes=bytesperline*4;
jump1: c = getc(fpcx);
    if (feof(fpcx))
        {pcxerror = "\nEOF encontrado prematuramente"; return 1;}
    if ( c > 0xc0 )
        { rep = c & 0x3f; c = getc(fpcx);
          if (feof(fpcx))
              {pcxerror = "\nEOF encontrado prematuramente"; return 1;}
          if ( i+rep > tbytes)
              { qued = i+rep - tbytes;
                ungetc(c, fpcx); ungetc(0xc0, fpcx); rep -= qued;
              }
          _fmemset(uncpl+i, c, rep); i+=rep;
        }
    else
        uncpl[i++] = c;
    if( i < tbytes )
        goto jump1;
jump2:
    return 0;
}

int displaypcximáge(char *file)
{ int i,j;
  char *fbuff, far *direc;
  FILE *fpcx; PCX pcx;
  char bytesline;

  if ( readpcxfile(file, &pcx) || verifypcxheader(&pcx) )
      return 1;

  if( (fpcx = fopen(file, "rb")) == NULL)
      { pcxerror = "No puedo abrir archivo"; return 1;}

  if (!(fbuff = malloc(fbuffsize)) || setvbuf(fpcx, fbuff, _IOFBF, fbuffsize)
!= 0)
      { pcxerror = "failed to set up buffer for input file\n";
        return 1;
      }
  if( fseek(fpcx, sizeof(pcx), SEEK_SET))
      { pcxerror = "Eof Inesperado";

```

```

        free( fbuff );
        return 1;
    }
    bytesline = pcx.bytesperline;
    j =pcx.ymin;
    if ( pcx.vertdpi > 351 )
        setvideomode(0x12);
    else
        if (pcx.vertdpi > 201)
            setvideomode(0x10);
        else setvideomode(0xe );
    evga_putrgbpalette(pcx.colormap);
    jump1:
        if ( j > pcx.ymax) goto jump4;
        if (getlinepixel(fpcx,bytesline)) return 1;
        i=0;
        direc = videoff+j*videobpl;
        jump2:
            if(i ==4) goto jump3;
            outportb(0x3c4,2);
            outportb(0x3c5,1<<i);
            _fmemcpy(direc,uncpl+i*bytesline,bytesline);
            i++;
            goto jump2;
        jump3:
            j++;
        goto jump1;
    jump4:
        outportb(0x3c4,2);
        outportb(0x3c5,15);
        free(fbuff);
        fclose(fpcx);
        return 0;
    }

void viewpcxfile( char *file )
{ setvideomode(3);
  assignfname(filename,file);
  displaypcxinfo(filename);
  printf("\n\nPresione una tecla para continuar...(ESC not display)");
  if( getch() != 27)
  {   if (displaypcximage(filename) )
      { setvideomode(3);
        printf("\n %s",pcxerror);
        return;
      }
    getch();
  }
}

```

ANEXO D

LISTADO DE EL ARCHIVO WPCX.C

```

#include "evgalib.c"
#include "pcx.H"
#include <stdio.h>
#include <stdlib.h>
#include <mem.h>

```

```

unsigned int rlelen;

```

```

int count_rle_bytes(int bytesperline)

```

```

{ int i,pr;
  pr = 0;
  for(i =0;i< bytesperline;)
  { if ((compl[pr] & 0xc0) == 0xc0 )
    { i += compl[pr] & 0x3f;
      pr+=2;
      continue;
    }
    i++;
    pr++;
  }
  if( i != bytesperline)
  { setvideomode(3);
    printf("\nerrrrrrrrr");
    exit(1);
  }
  return pr;
}

```

```

void rle_line(int bytesperline)

```

```

{ int i,pw,count=1;
  memset(compl,0,80);
  if ( (vline[0] == 0xff) && (vline[1] == 0) )
    pw = 0;
  for( pw=i = 0;i< bytesperline*4;count = 1)
  { while( vline[i] == vline[i+count] && i+count < bytesperline*4)
    count ++;
    if ( count > 1 )
    { while ( count >= 0x3f )
      { compl[pw++] = 0xff;compl[pw++] = vline[i];count -=
0x3f;i+=0x3f;
      }
      if (count >0 )
      { compl[pw++]=0xc0 ; count; compl[pw++] =
vline[i];i+=count;
      }
      continue;
    }
    if ( vline[i] >= 0xc0 )
      compl[pw++] = 0xc1;
  }
}

```

```

        compl[pw++] = vline[i++];
    }
    rlelen = pw;
    unrle_line(bytesperline);
    if ( memcmp(vline,uncpl,bytesperline*4) )
    {
        setvideomode(3);
        printf("\nerrrrrorrrr");
        exit(1);
    }
}

void write_rle_line( FILE *fpcx )
{
    fwrite(compl,1,rlelen,fpcx);
}

void writepcxfile(char *name)
{
    PCX pcx;
    int i,j; unsigned char far* direccion; unsigned seepage;
    FILE *fpcx;
    seepage = 0xa000;

// load pcxinfo
    pcx.id = '\n';    pcx.version = 5;
    pcx.compr = 1;    pcx.bitsperpixel= 1;
    pcx.xmin=pcx.ymin = 0;pcx.xmax =videomaxx;pcx.ymax= videomaxy;
    pcx.horidpi = videomaxx+1;pcx.vertdpi=videomaxy+1;

    evga_getrgbpalette(pcx.colormap);
    pcx.ncolplanes = 4;
    pcx.bytesperline= pcx.horidpi/8;
    pcx.greyscale = 1;
    fpcx = fopen(name,"wb");
    if (fpcx == NULL )
    {
        setvideomode(3);
        printf("\nNo podemos crear archivo destino");
        exit(1);
    }
    fwrite(&pcx,1,sizeof(pcx),fpcx);
    for(j = 0;j <= videomaxy;j++)
    {
        direccion = (unsigned char far *)MK_FP(seepage
,j*pcx.bytesperline);
        memset(vline,0,80);
        for(i= 0; i<4;i++)
        {
            outputb(0x3ce,4);
            outputb(0x3cf,i);
        }
        _fmemcpy(vline+(int)(pcx.bytesperline*i),direccion,pcx.bytesperline);
    }
}

```

```
    }  
    outportb(0x3cf,0);  
    rle_line(pcx.bytesperline);  
    write_rle_line(fpcx);  
  }  
  fclose(fpcx);  
}
```