

## Tema 9: Dispositivos periféricos y controladores

### Objetivos:

- Entender los principios de funcionamiento de los discos magnéticos y analizar la estructura y funcionamiento del controlador para un caso sencillo (*floppy*).
- Entender los principios de grabación y lectura de discos ópticos así como los niveles de codificación de la información.
- Describir el comportamiento de un monitor CRT y su controlador asociado.
- Entender el funcionamiento de las pantallas de cristal líquido de celdas pasivas (LCD) y de celdas activas con transistores de película delgada (TFT).
- Estudiar el funcionamiento de algunas impresoras y otros periféricos de uso común.

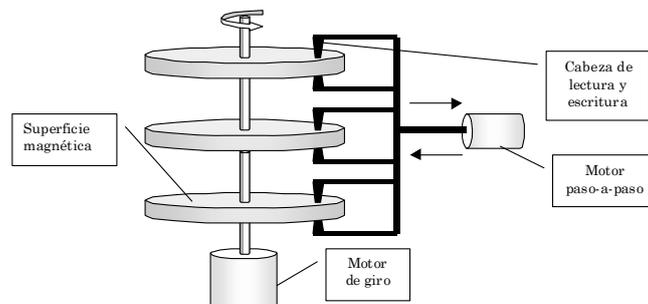
### Contenido:

1. Discos magnéticos
2. Discos ópticos
3. Monitores de tubo de rayos catódicos (CRT)
4. Pantallas de cristal líquido (LCD-TFT)
5. Monitores de color
6. Unidades de procesamiento gráfico (GPU)
7. Impresoras
8. Ratón

Estudiaremos en este apartado los principios de funcionamiento de los principales dispositivos periféricos que se conectan a un computador así como la estructura lógica de los correspondientes controladores.

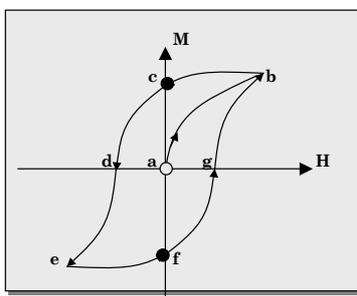
### 1. Discos magnéticos

Los discos magnéticos se conectan al computador como dispositivos periféricos, aunque muchos forman parte de la jerarquía de memoria del computador. Un dispositivo de discos magnéticos está constituido por una o más superficies circulares recubiertas por un material ferromagnético sobre el que se puede grabar información alterando el sentido de la magnetización de pequeñas áreas de su superficie. El conjunto de superficies gira por la acción de un motor. La información se escribe y lee a través de un conjunto de cabezas (una por superficie) que se mueve radialmente de forma solidaria por la acción de un motor *paso-a-paso*, tal como se muestra en la siguiente figura:

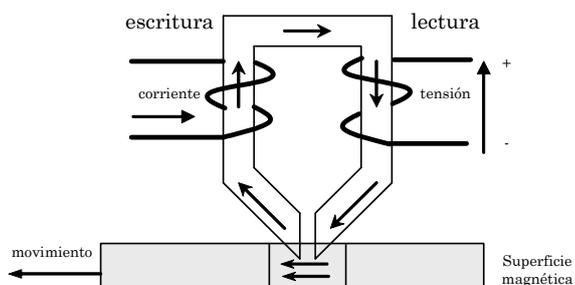


### 1.1. Grabación de información sobre superficies magnéticas

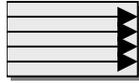
Los materiales *ferromagnéticos* se caracterizan porque se magnetizan de forma permanente cuando se les aplica un campo magnético externo. Normalmente el campo externo se genera eléctricamente. Esto no les ocurre a los materiales diamagnéticos, que son repelidos por los campos magnéticos y no se magnetizan, ni a los materiales paramagnéticos, que son atraídos por los campos magnéticos y tampoco se magnetizan. Un material ferromagnético se caracteriza por sus curvas de magnetización e histéresis. Supongamos que el material ferromagnético está originariamente desmagnetizado (punto *a* de la siguiente gráfica). Conforme se incrementa el campo magnético externo *H*, también se incrementa la magnetización inducida *M*, hasta que se alcanza el valor de saturación (punto *b*). La curva entre los puntos *a* y *b* se denomina *curva de magnetización*. Si ahora reducimos el campo externo, la magnetización inducida también se reduce, pero no sigue la curva original. El material retiene una cierta magnetización permanente denominada *magnetización remanente*, que persiste incluso cuando se anula el campo externo (punto *c*). Si reducimos aún más el campo externo (sentido opuesto) podemos llegar a anular la magnetización remanente (Punto *d*), y si continúa la reducción se crea una magnetización inducida de sentido opuesto, llegándose también en este caso a un valor de saturación (punto *e*). Si ahora vamos reduciendo el campo externo hasta anularlo, el material queda con una magnetización remanente de sentido opuesto a la anterior (punto *f*). Para anular esta magnetización remanente debemos aumentar el campo externo (con el sentido original) hasta alcanzar el punto *g*, y si seguimos aumentando llegamos de nuevo a la saturación (punto *b*), con lo que se completa el ciclo de histéresis. Los materiales ferromagnéticos tienen por tanto dos valores estables de magnetización, uno norte, correspondiente al punto *c* de la gráfica y otro sur, correspondiente al punto *f*. Estos dos estados de magnetización remanente que se mantienen en ausencia de campo externo son los que se utilizan para codificar la información binaria cuando se utilizan los materiales ferromagnéticos como soporte de memoria.



Los procesos de lectura y escritura sobre las superficies ferromagnéticas de un disco (habitualmente de óxido de hierro) se realizan a través de una cabeza de grabación compuesta por un núcleo de hierro en forma de herradura y dos bobinas conductoras, una para escritura y otra para lectura. La cabeza opera muy cerca de la superficie, de forma que casi vuela sobre ella, sobre el colchón de aire formado por su propio movimiento. Debido a esto, están cerrados herméticamente, porque cualquier partícula de polvo puede dañarlos.



**Escritura:** Se hace pasar por la bobina de escritura una corriente que crea un campo magnético en el núcleo de hierro que se cierra a través de la pequeña región de la superficie magnética que en ese momento está bajo la cabeza, dejando una magnetización remanente del mismo sentido que el campo de la bobina.



**Lectura:** Se mueve la superficie magnética previamente grabada por debajo de la cabeza. Cada variación de flujo producida por las pequeñas áreas magnetizadas inducirá una pequeña tensión en la bobina de lectura. Amplificando e interpretando adecuadamente estas tensiones podemos reconstruir en forma de señal eléctrica digital los valores grabados en la superficie del disco.

Los discos magnéticos actuales disponen de *AUTOPARK* o aparcamiento automático de las cabezas, que posiciona las cabezas en un lugar fuera del alcance de la superficie del disco al apagar el ordenador. Esto evita posibles daños en la superficie del disco cuando la unidad es sometida a vibraciones o golpes en un posible traslado.

## 1.2. Métodos de codificación sobre superficies magnéticas

Para grabar información en la superficie, se siguen una serie de códigos que transforman un patrón de bits en una secuencia de celdas con diferentes estados de magnetización.

### 1) Retorno a cero (RZ: «return to zero»).

El 1 lógico se graba magnetizando el centro de la celda. El 0 es ausencia de magnetización

### 2) Retorno a saturación (RS: «return to saturation»).

El 1 se representa por un estado de magnetización en el centro de la celda. El resto de la celda y el 0 se representa con el estado de magnetización opuesto.

### 3) Retorno a cero bipolar (BRZ: «bipolar return to zero»).

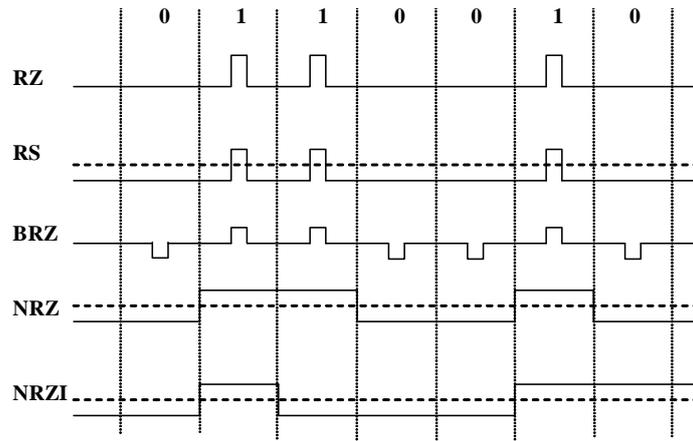
El 1 se representa por un estado de magnetización en el centro de la celda y el 0 por el estado de magnetización opuesto, también en el centro de la celda. Los extremos de la celda quedan sin magnetizar.

### 4) Sin retorno a cero (NRZ: «non-return zero»).

El 1 se representa por un estado de magnetización en toda la celda y el 0 por el estado contrario, también en toda la celda.

### 5) Sin retorno a cero inverso (NRZI: «non-return to zero invert»).

El 1 se representa por un cambio del sentido del flujo de magnetización en toda la celda respecto al estado de la celda anterior. El 0 se representa por ausencia de cambio.



**6) Codificación en fase (PE: «phase encoding»).**

El 1 se representa por un cambio y el 0 por el cambio contrario. Estos cambios se realizan en el centro de la celda. Para poderse detectar dos ceros o dos unos seguidos, es necesario efectuar también un cambio al comienzo del segundo.

**7) Modulación de frecuencia (FM: «frequency modulation»).**

Es la codificación más sencilla, consiste en la grabación de un cambio de flujo para cada uno, y el omitir el cambio de flujo para cada cero. Es decir, siempre se produce un cambio al comienzo de la celda, en el centro se producirá un nuevo cambio de magnetización si y sólo si se graba un 1. Este procedimiento se puede realizar con una electrónica de control relativamente simple, pero tiene el inconveniente de que cada bit de datos consume dos cambios de flujo, limitando mucho la capacidad del disco

**8) Coeficiente de pulso (PR: «pulse ratio»).**

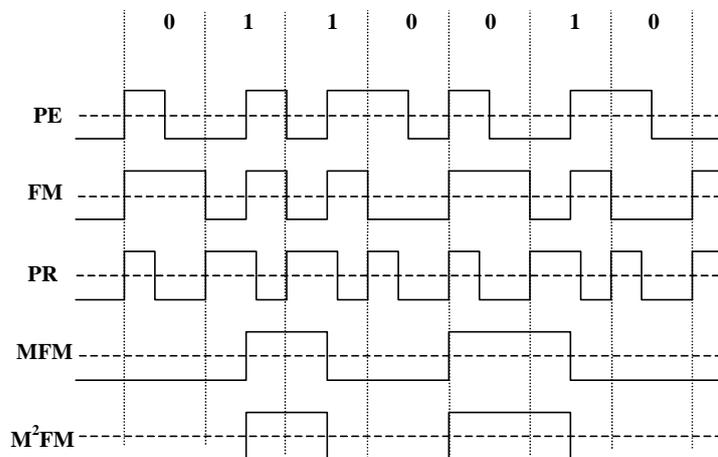
La celda tiene un primer intervalo de longitud RI en un estado de magnetización, y el segundo intervalo, de longitud R2, en el estado contrario. El 1 se representa por un pulso tal que  $R1/R2 < 1$ , y el 0 por un pulso tal que  $R1/R2 > 1$ .

**9) FM modificada (MFM: «modified FM»).**

Similar a FM excepto que el cambio de estado al inicio de la celda se efectúa sólo si el bit precedente y actual son ambos 0.

**10) Modulación de frecuencia modificada-modificada (MIFM: «modified-modified FM»).**

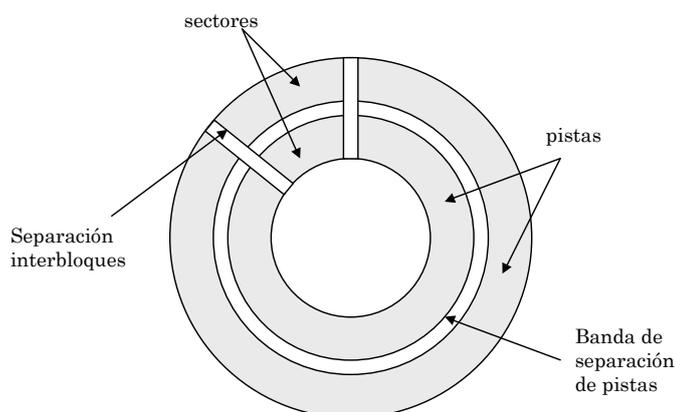
Igual que FM pero sólo se efectúa el cambio de estado al comienzo de la celda si la celda anterior no contiene ningún cambio (ni al principio ni en el centro), y la celda actual no es un 1.



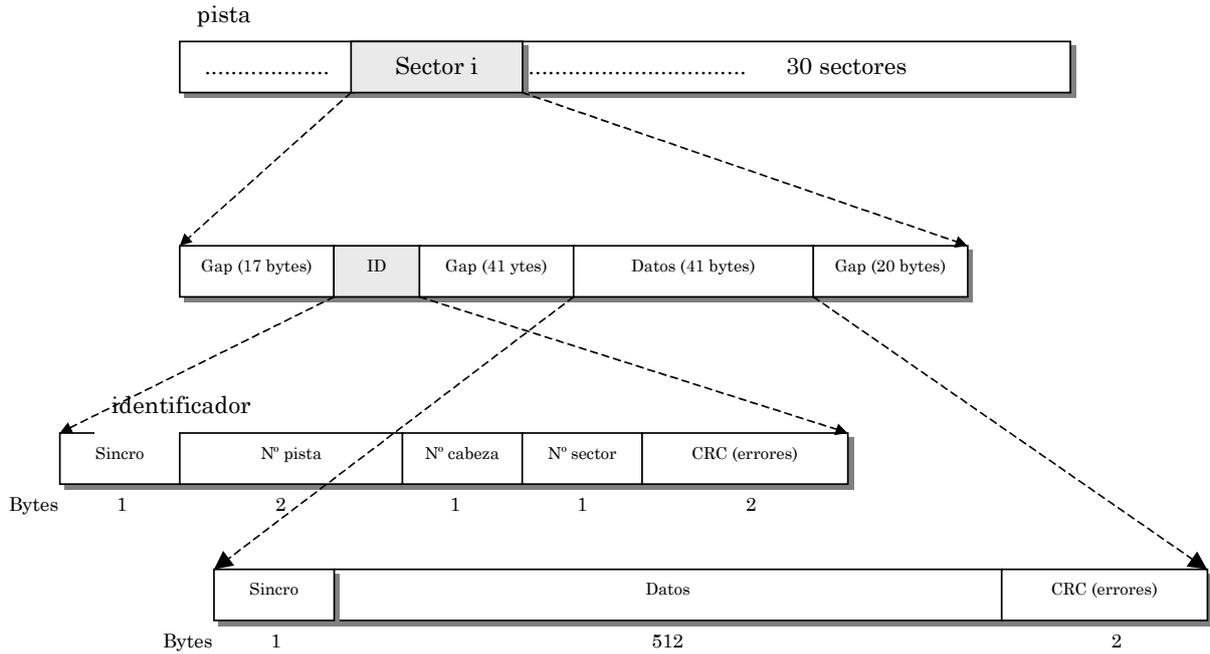
### 1.3. Formato de grabación

Los datos en un disco magnético se organizan en un conjunto de anillos concéntricos sobre el plato denominados *pistas*. Las pistas adyacentes están separadas por bandas vacías. Esto minimiza los errores debidos a falta de alineamiento de la cabeza o a interferencias del campo magnético. Para simplificar la electrónica, se suele almacenar el mismo número de bits en cada pista. Esto significa que la densidad en bits es mayor en las pistas más internas.

Los datos se transfieren en bloques y se almacenan en regiones del tamaño de un bloque, conocidas como sectores. Normalmente hay entre 10 y 100 sectores por pista, y estos pueden ser de longitud fija o variable. Los sectores adyacentes se separan con regiones vacías. El disco se graba con un formato que contiene algunos datos extra, usados sólo por el controlador del disco y no accesibles al usuario. En la siguiente figura se puede ver la disposición de los datos:



En la siguiente figura se muestra un ejemplo del formato de grabación de un disco. En este caso, cada pista contiene 30 sectores de longitud fija, de 600 bytes cada uno. Cada sector contiene 512 bytes de datos, junto con la información de control utilizada por el controlador del disco. El campo ID es un identificador único, o dirección, usado para localizar un sector particular. El byte SINCRO es un patrón de bits especial que delimita el comienzo del campo. El número de pista identifica cada pista. El número de cabeza identifica una cabeza, si el disco tiene varias superficies. El ID y los campos de datos contienen, cada uno, un código de detección de errores de redundancia cíclica (CRC) que estudiaremos en el apartado siguiente.



Existen procedimientos para aumentar la capacidad del disco colocando más sectores sobre las pistas exteriores que son más largas, manteniendo constante la velocidad de rotación. El sistema ZBR (Zone Bit Recording) es un ejemplo que requiere el uso de controladores más complejos.

Un parámetro a tener en cuenta en el funcionamiento de un disco es el tiempo medio entre fallos, *MTBF* (Mean Time Between Failures), que se mide en horas (15000, 20000, 30000). Cuanto mayor sea este parámetro mayor será la fiabilidad del disco.

#### 1.4. Comprobación de Redundancia Cíclica (CRC)

Los códigos de comprobación de redundancia cíclica (CRC) se utilizan fundamentalmente en dispositivos de tratamiento serie de la información. El emisor genera a partir de un mensaje de  $k$ -bits que se quiere transmitir una secuencia de  $n$ -bits, denominada secuencia de comprobación (FCS: *frame check sequence*), que concatena con el anterior para formar el mensaje realmente transmitido, con  $n+k$  bits. El FCS se elige de manera que el mensaje transmitido sea divisible por algún número predeterminado. El receptor dividirá el mensaje recibido por ese número, y si el resto de la división es cero significa que no ha habido errores en la transmisión.

##### 1.4.1. Aritmética módulo 2

La aritmética que se utiliza en las operaciones CRC es módulo 2, de acuerdo con las reglas de la teoría de campos algebraicos. Es decir, no existen acarreo para la suma ni préstamos para la resta. Tanto la suma como la resta resultan pues idénticas a una O-exclusiva:

Suma	Resta	Multiplicación	División
$\begin{array}{r} 1111101 \\ + 0101001 \\ \hline 1010100 \end{array}$	$\begin{array}{r} 1101101 \\ - 0101001 \\ \hline 1000100 \end{array}$	$\begin{array}{r} 1111101 \\ \times 11 \\ \hline 1111101 \end{array}$	$\begin{array}{r} 11 \leftarrow \text{Cociente} \\ \hline 1111101 \\ 10000111 \leftarrow \text{Dividendo} \\ \hline 1111101 \end{array}$

		<pre> 1111101 ----- 10000111 </pre>	<pre> ----- 1111101 1111101 ----- 0 ← Resto </pre>
			Divisor

**1.4.2. Codificación**

M = mensaje de  $k$  bits original que se desea transmitir

F =  $n$  bits de la secuencia de comprobación (FCS: *Frame Check Sequence*)

T = mensaje de  $(k + n)$  bits que se transmite, con  $n < k$

P = patrón de  $n + 1$  bits: divisor elegido

Es evidente que multiplicar M por  $2^n$  equivale a desplazar hacia la izquierda  $n$  bits, introduciendo ceros por la derecha. Por tanto la operación de concatenación de M con F para dar T podemos expresarla de la siguiente forma:

$$T = 2^n M + F$$

El objetivo es escoger F de tal manera que T sea divisible por P. Como:

$$\begin{array}{r} 2^n M \quad R \\ \text{-----} = Q + \text{--} \\ P \quad P \end{array}$$

Siendo Q el cociente y R el resto. Por tanto si tomamos  $F = R$ , T será divisible por P

$$T = 2^n M - R = 2^n M + R$$

Ya que la suma y resta coinciden en aritmética módulo 2.

Luego el mensaje total a transmitir estará compuesto por la concatenación del mensaje original M y el resto de la división de M, desplazado  $n$  unidades a la izquierda ( $2^n M$ ), por el patrón P.

**Ejemplo**

Emisor

Mensaje original a transmitir M = 1010001101 (10 bits)

Patrón P = 110101 (6 bits)

FCS R = resto de la división de  $2^n M$  entre P

$$2^n M = 2^5 M = 101000110100000$$

```

          1101010110 ← Q
          -----
P → 110101 101000110100000 ← 2^n M
          110101
          -----
          111011
          110101
          -----
          111010
          110101
          -----
          111110
          110101
          -----
          101100
          110101
          -----

```

```

110010
110101
-----

```

01110 ← R

El resto se suma a  $2^n M$  para formar T

$T = 2^n M + R = 101000110\ 01110$

Que es el mensaje o trama que se transmite

### Receptor

Si no hay errores el receptor recibe el mensaje transmitido por el emisor y calcula el resto de la división por P:

```

                1101010110 ← Q
                -----
P → 110101    101000110101110 ← 2^n M
                110101
                -----
                111011
                110101
                -----
                111010
                110101
                -----
                111110
                110101
                -----
                101111
                110101
                -----
                110101
                110101
                -----
                0 ← R

```

Como el resto es cero se sabe que no ha habido errores.

### 1.4.3. Representación de los errores

Si se produce un error en la transmisión de un bit del mensaje significa que si en origen valía 0 en destino vale 1, y viceversa. Por tanto la representación de cualquier error (simple o múltiple) podemos hacerla con una cadena de bits  $E$  de longitud igual al mensaje transmitido, con valor 1 en aquellas posiciones de bit en las que se haya producido un error. El mensaje recibido  $Tr$  tendrá pues la siguiente expresión en términos del mensaje transmitido  $T$  y el error producido  $E$ :

$$Tr = T \oplus E$$

Si  $T = 11000111$  y se ha producido un error en el tercer bit por la derecha, es decir,  $E = 00000100$ , entonces  $Tr = (11000111) \oplus (00000100) = 11000011$

Evidentemente el receptor fallará en la detección de un error si y solo si  $Tr$  es divisible por P, lo que equivale a decir que E sea divisible por P.

### 1.4.4. Interpretación polinómica de los códigos CRC

Las secuencias de bits podemos representarlos como polinomios de una variable anónima X con coeficientes 0 ó 1. Con ello facilitaremos los razonamientos de divisibilidad entre mensajes T y patrones de bits P. Los coeficientes corresponderán con los bits de la configuración binaria de la secuencia. Así, si la secuencia es  $M = 110011$ , entonces  $M(X) = X^5 + X^4 + X + 1$ . Las operaciones

aritméticas siguen realizándose en módulo 2. El polinomio correspondiente al mensaje que se transmite en código CRC vendrá dado por:

$$T(X) = X^n M(X) + R(X) \text{ donde } R(X) \text{ es el resto de la división de } X^n M(X) \text{ entre } P(X),$$

es decir:

$$X^n M(X) / P(X) = Q(X) + R(X) / P(X)$$

#### 1.4.5. Tipos de errores

La forma del polinomio  $P(X)$  determina el tipo de error que es capaz de detectar el sistema, es decir, simple, doble, de ráfagas, etc. Al recibir el mensaje  $Tr(X) = T(X) \oplus E(X)$  el receptor lo divide por  $P(X)$ . Como  $T(X)$  por construcción es divisible por  $P(X)$ , si  $E(X)$  también es divisible por  $P(X)$  nos quedamos sin saber si se ha producido un error de tipo  $E(X)$ , puesto que en este caso todo el mensaje recibido  $Tr(X)$  sería divisible por  $P(X)$ . Veamos pues qué condición debe cumplir  $P(X)$  para que sea posible detectar un error de tipo  $E(X)$

##### 1.4.5.1. Error simple

A un error de tipo simple (en un solo bit) le corresponde un polinomio  $E(X) = X^i$  donde  $i$  determina la posición del error. Si  $P(X)$  contiene el factor  $X+1$ ,  $E(X)$  nunca será divisible por  $P(X)$ , por lo que detectarán los errores simples.

##### 1.4.5.2. Errores dobles

En este caso  $E(X) = X^i + X^j$ , donde  $i > j$ . Esto también se puede escribir como  $E(X) = X^j (X^{i-j} + 1)$ . Si suponemos que  $P(X)$  no es divisible entre  $X$ , una condición suficiente para detectar todos los errores dobles es que  $P(X)$  no sea divisor exacto de  $(X^k + 1)$  para ningún  $k$  hasta el valor máximo de  $i - j$ , es decir, hasta la longitud máxima del mensaje. Se conocen polinomios sencillos de bajo grado que dan protección a mensajes largos. Por ejemplo,  $X^{15} + X^{14} + 1$  no dividirá exactamente a  $(X^k + 1)$  para ningún valor de  $k < 32.768$ .

##### 1.4.5.3. Errores en ráfaga:

Un error en ráfaga comienza y finaliza en 1, y entre ambos 1's hay 0's y 1's. Para un error en ráfaga de longitud  $k$ , el polinomio tendrá la forma:  $X^i(X^{k-1} + \dots + 1)$ , donde  $i$  determina la distancia a la que se encuentra la ráfaga desde el extremo derecho del mensaje recibido. Si  $P(X)$  contiene un término  $X^0$ , no tendrá a  $X^i$  como factor, por lo que, si el grado de  $(X^{k-1} + \dots + 1)$  es menor que el grado de  $P(X)$ , el residuo nunca puede ser cero. Si la longitud de la ráfaga es  $r+1$ , el residuo de la división entre  $P(X)$  será cero si y solo si la ráfaga es idéntica a  $P(X)$ . Por la definición de ráfaga, el primero y último bits deben ser 1, así que el que sean iguales o no depende de los  $r - 1$  bits intermedios. Si se consideran igualmente probables todas las combinaciones, la probabilidad de que se acepte como válido el mensaje recibido de de  $1/2^{r-1}$

#### 1.4.6. Polinomios generadores estándar

$$\text{CRC-12} = X^{12} + X^{11} + X^3 + X^2 + X^1 + X^0$$

$$\text{CRC-16} = X^{16} + X^{15} + X^2 + X^0$$

$$\text{CRC-CCITT} = X^{16} + X^{12} + X^5 + X^0$$

$$\text{CRC-22} = X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X^1 + X^0$$

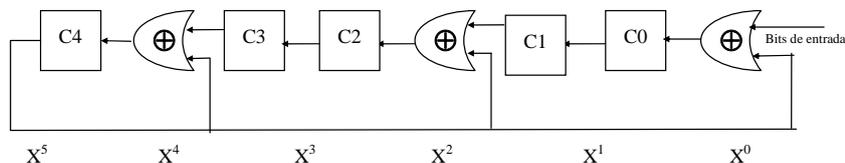
#### 1.4.7. Implementación hardware

La generación y detección CRC se implementa con un circuito divisor compuesto por puertas O-exclusiva y un registro de desplazamiento. El registro contiene tantos bits como el FCS, es decir,  $n$ , y  $n-1$  puertas O-exclusiva. Cada puerta O-exclusiva se ubica a la entrada de un biestable correspondiente a un monomio de coeficiente 1, salvo  $X^n$ .

Para el divisor del ejemplo, 110101, cuyo polinomio asociado es

$$X^5 + X^4 + X^2 + X^0$$

El divisor tendrá la siguiente forma:



Sobre este circuito podemos seguir el proceso de generación CRC del ejemplo anterior, para el mensaje M = 1010001101:

	C4	C3	C2	C1	C0	C4 ⊕ C3	C4 ⊕ C1	C4 ⊕ entrada	entrada	
Inicial	0	0	0	0	0	0	0	1	1	<div style="display: flex; align-items: center; justify-content: center;"> <div style="font-size: 3em; margin-right: 10px;">}</div> <div style="text-align: center;">Mensaje a enviar</div> </div> <div style="display: flex; align-items: center; justify-content: center;"> <div style="font-size: 3em; margin-right: 10px;">}</div> <div style="text-align: center;">cinco ceros</div> </div>
Paso 1	0	0	0	0	1	0	0	0	0	
Paso 2	0	0	0	1	0	0	1	1	1	
Paso 3	0	0	1	0	1	0	0	0	0	
Paso 4	0	1	0	1	0	1	1	0	0	
Paso 5	1	0	1	0	0	1	1	1	0	
Paso 6	1	1	1	0	1	0	1	0	1	
Paso 7	0	1	1	1	0	1	1	1	1	
Paso 8	1	1	1	0	1	0	1	1	0	
Paso 9	0	1	1	1	1	1	1	1	1	
Paso 10	1	1	1	1	1	0	0	1	0	
Paso 11	0	1	0	1	1	1	1	0	0	
Paso 12	1	0	1	1	0	1	0	1	0	
Paso 13	1	1	0	0	1	0	1	1	0	
Paso 14	0	0	1	1	1	0	1	0	0	
Paso 15	0	1	0	1	0	1	1	0	0	

### 1.5. Parámetros de rendimiento de un disco magnético

#### 1.5.1. Tiempo de búsqueda

El tiempo de búsqueda ( $T_s$ ) es el tiempo que tarda la cabeza en posicionarse en la pista. Se compone de dos partes, el tiempo de comienzo  $s$  o tiempo necesario para que la cabeza adquiera una cierta velocidad radial, y el tiempo necesario para atravesar las pistas que preceden a la que se accede, que no es una función lineal del número de pistas atravesadas. Sin embargo podemos aproximar  $T_s$  con la siguiente expresión:

$$T_s = m * n + s$$

donde:

$n$  = número de pistas atravesadas

$m$  = constante que depende del disco

En un disco barato de PC  $m$  puede tener un valor de  $0,3 ms$  y  $s$  de  $20 ms$ ; mientras que en un disco más caro  $m$  puede tener un valor de  $0,1 ms$  y  $s$  de  $3 ms$

#### 1.5.2. Retardo rotacional

El retardo rotacional ( $T_r$ ) es el tiempo que tarda el sector en alcanzar la cabeza. Si  $r$  es la velocidad angular del disco en revoluciones/segundo ( $rps$ ), la expresión del valor medio de  $T_r$  será la siguiente:

$$T_r = 1/2r$$

En discos duros que giran a  $3.600 rpm$ , una revolución tardará  $16,7 ms$ , y el valor medio del retardo rotacional será  $8,3 ms$ . En cambio los disquetes que giran entre  $100$  y  $600 rpm$  tienen un retardo rotacional medio que va de  $100$  a  $200 ms$ .

### 1.5.3. Tiempo de acceso

El tiempo de acceso ( $T_a$ ) es la suma del tiempo de búsqueda y el retardo rotacional

$$T_a = T_s + T_r = m * n + s + 1/2r$$

### 1.5.4. Tiempo de transferencia ( $T_t$ )

Es el tiempo que tardan en transferirse los datos a ó desde el disco una vez accedido el inicio de los mismos. Dependerá de la velocidad de rotación del disco:

$$T_t = b/rN$$

donde:

$b$  = número de bytes a transferir

$N$  = número de bytes de una pista

$r$  = velocidad de rotación en rps

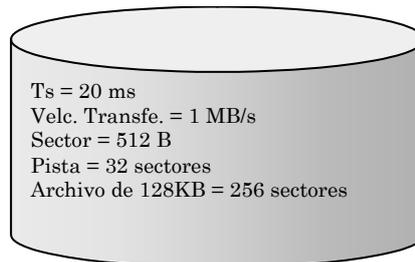
### 1.5.5. Tiempo de operación ( $T_o$ )

Es la suma de las componentes anteriores, es decir:

$$T_o = m * n + s + 1/2r + b/rN$$

En un computador que funciona bajo control de un SO, para realizar una operación de E/S, a estos tiempos habrá que añadir el *tiempo de espera por un canal*, si el disco no dispone del suyo propio, y el *tiempo de espera en la cola* hasta que el dispositivo esté disponible.

### Comparación de tiempos en acceso secuencial y aleatorio



#### 1) Acceso secuencial

8 pistas adyacentes (256 sectores / 32 sectores/pista = 8 pistas)

Acceso a la primera pista =  $T_s + T_r + T_t(1 \text{ pista})$

$T_s = 20$  ms

Bytes / pista =  $512 * 32 = 16.384$

Tiempo / pista =  $16.384 / 1.000.000 = 0,016384$  s.

$T_r = 0,016384 / 2 = 0,008192 = 8,192$  ms

$T_t(1 \text{ pista}) = 0,016384$  s. = 16,384 ms

Acceso a la primera pista =  $T_s + T_r + T_t(1 \text{ pista}) = 45$  ms.

Acceso a las restantes 7 pistas =  $7 * (T_r + T_t(1 \text{ pista})) = 7 * (8,192 + 16,384) = 7 * 25 = 220$  ms.

#### 2) Acceso aleatorio

Acceso a un sector =  $T_s + T_r + T_t(1 \text{ sector}) = 20 + 8,192 + 512/1.000.000 = 28,8$  ms.

Acceso a 256 sectores =  $256 * 28,8 = 7.372$  ms

Existen una serie de técnicas que permiten aminorar los accesos a disco así como acelerar las transferencias de datos entre el sistema y el dispositivo. Una de las más conocidas es la utilización de memorias intermedias, *buffers* o cachés.

**Buffer de Pista:** Es una memoria incluida en el controlador del disco, que almacena el contenido de una pista completa. Así cuando se hace una petición de lectura de una pista, esta se puede leer de una sola vez, enviando la información a la CPU.

**Cachés de Disco:** Pueden estar dentro del propio disco duro, en tarjetas especiales o bien a través de programas utilizando la memoria central. La gestión de esta memoria es completamente transparente y consiste en almacenar en ella los datos más pedidos por la CPU y retirar de ella aquellos no solicitados en un determinado tiempo.

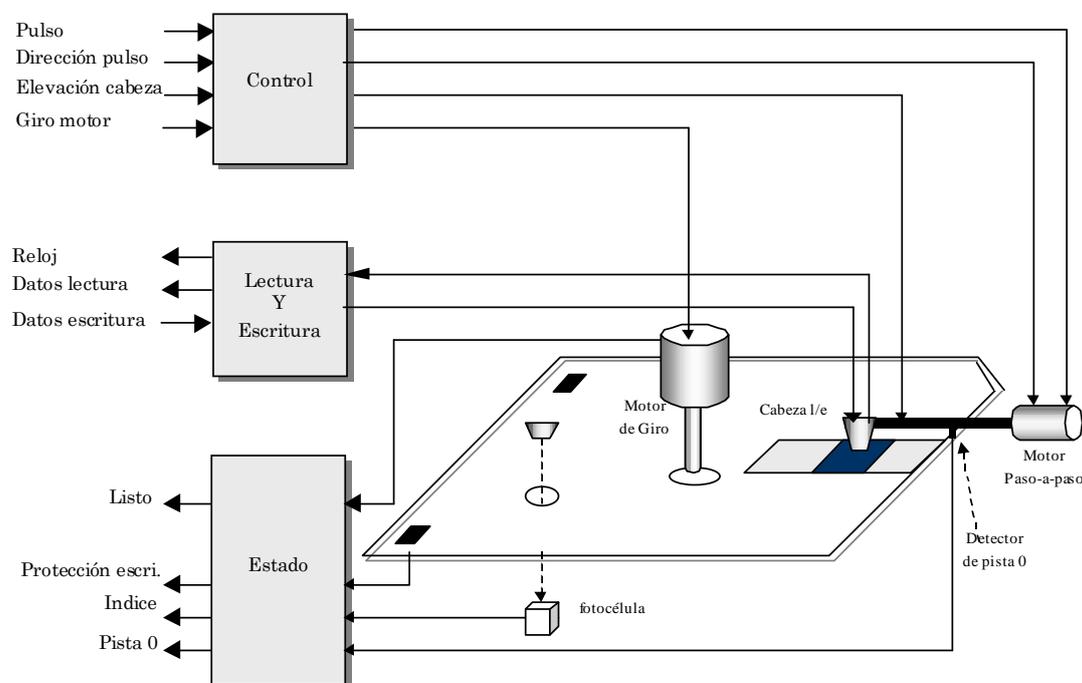
## 1.6. Dispositivo floppy

El dispositivo *floppy disk* fue introducido por IBM en 1967. La primera versión tenía un tamaño de 8 pulgadas y se redujo a 5.50 pulgadas en la segunda con 360 KB, que se utilizó en el primer IBM PC en 1981. Posteriormente apareció la versión de 3.25 pulgadas con 1.44 MB. El *floppy* utiliza una fina capa de material plástico recubierta de óxido de hierro (material ferromagnético)

### 1.6.1. Estructura de bloques del dispositivo

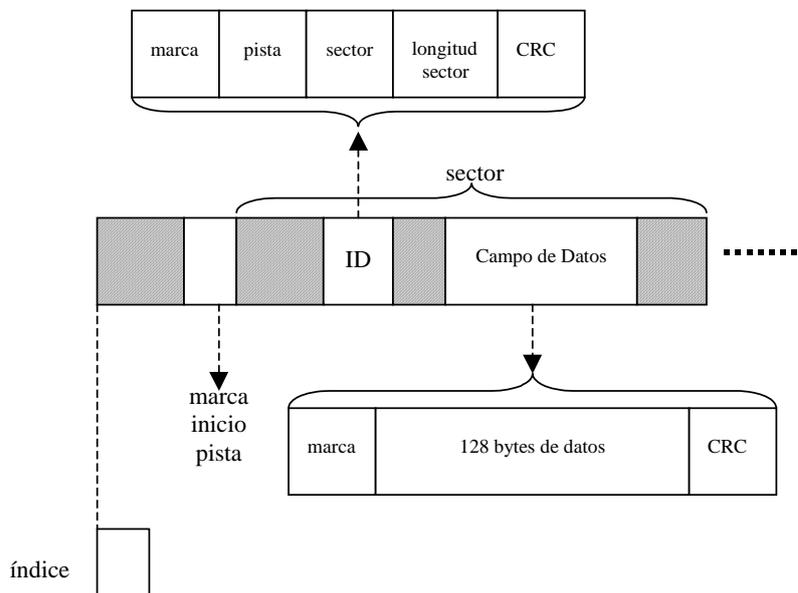
Como puede verse en el esquema de la siguiente figura, un dispositivo *floppy* dispone de los siguientes elementos:

- **Mecanismo de cabeza:** una sola cabeza para lectura y escritura, y otra para borrado inmediatamente antes de la escritura. La cabeza es accionada por un motor paso-a-paso que le permite posicionarla radialmente en una cualquiera de las pistas del *floppy*. También dispone de un mecanismo de elevación de la cabeza que se utiliza para evitar el rozamiento con la superficie cuando se mueve radialmente.
- **Motor de giro:** hace girar el *diskette* a una velocidad de 300 ó 369 revoluciones por minuto.
- **Detector de la pista cero:** que el dispositivo utiliza como referencia para desplazar la cabeza a la pista adecuada.
- **Detector de inicio de pista:** fotodetector que genera una señal cada vez que se inicia una nueva revolución de giro.



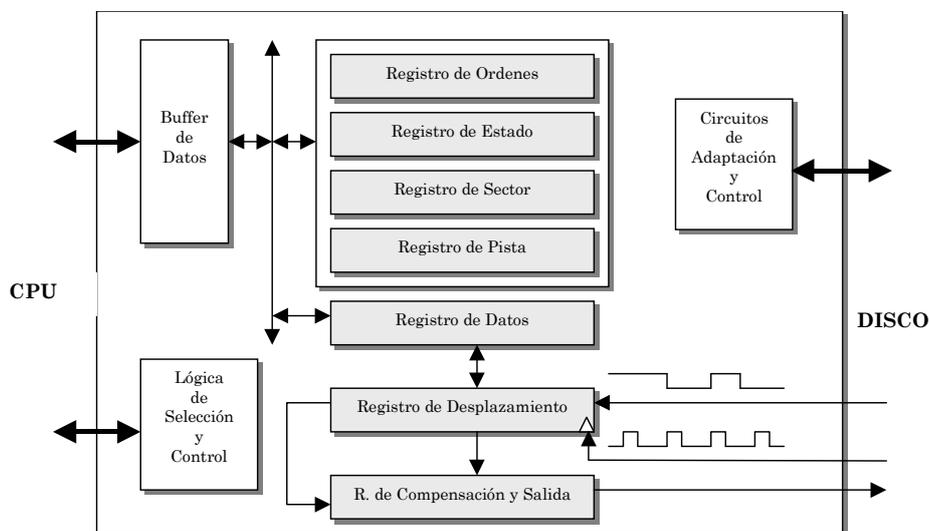
Todos los elementos del dispositivo son accionados y supervisados desde los circuitos de Control, Lectura/Escritura y Estado, a través del controlador del dispositivo.

### 1.6.2. Formato de datos



### 1.6.3. Estructura y funcionamiento del controlador de dispositivo

La estructura del controlador del *floppy disk* se representa en la siguiente figura:



Consta de los siguientes elementos:

**Registro de pista** contiene el número de pista sobre el que se encuentra la cabeza.

**Registro de estado** codifica la información de estado del dispositivo: disponible,....

**Registro de Ordenes** almacena la orden recibida por la unidad de E/S. Entre otras dispone de las siguientes:

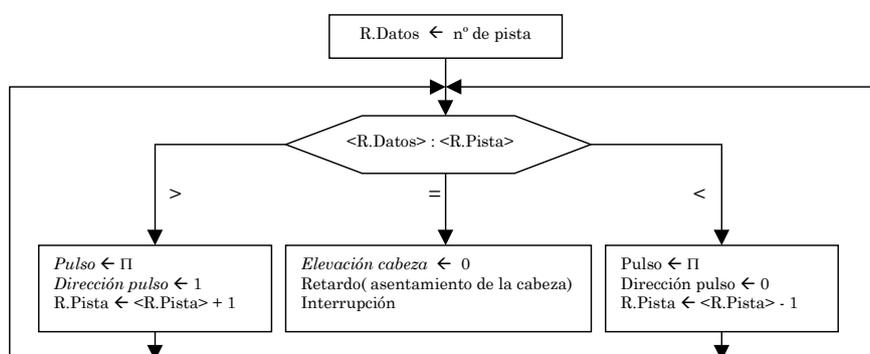
- Búsqueda de pista
- Búsqueda de sector
- Lectura de sector

❑ Escritura de sector

Analicemos las acciones que tienen lugar en el dispositivo durante la ejecución de la primera de estas órdenes.

Búsqueda de pista

El registro de pista se inicializa a 0 cuando se enciende el motor de giro y se activa el detector de pista 0. Una vez que el módulo de E/S ha puesto la orden de *Búsqueda de pista* sobre el *Registro de Ordenes*, carga en el *Registro de Datos* el número de la pista a buscar y comienza la operación. Se comparan los registros de *Datos* y *Pista*. Si sus contenidos coinciden, la cabeza se encuentra en la pista buscada, y la operación finaliza con una interrupción después de desactivar la señal de *Elevación de cabeza* y transcurrir un retardo de tiempo para el asentamiento de la cabeza. En caso contrario se envía un pulso al motor *paso-a-paso* de la cabeza con la dirección adecuada (dependiendo del resultado menor o mayor de la comparación) a fin de desplazarla a la pista contigua. Después se actualiza el *Registro de Pista* y se vuelve a comparar.

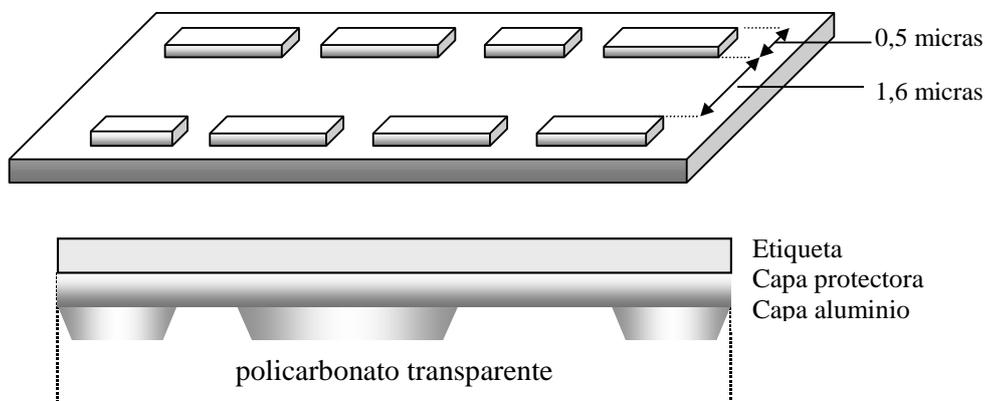


**2. Discos ópticos**

Se trata de dispositivos para el almacenamiento de información cuya lectura se realiza por medios ópticos. Existen diferentes tipos, siendo los más importantes el *CD-ROM* (*Compact Disc ROM*) y el *WORM* (*Write Once, Read Many times*).

**2.1. CD-ROM**

La información se almacena en forma de hoyos (*pits*) y salientes (*bumps*), grabados mecánicamente sobre un sustrato de aluminio brillante, y son leídos por medio de un láser que se refleja con diferente dirección en los hoyos que en los salientes.



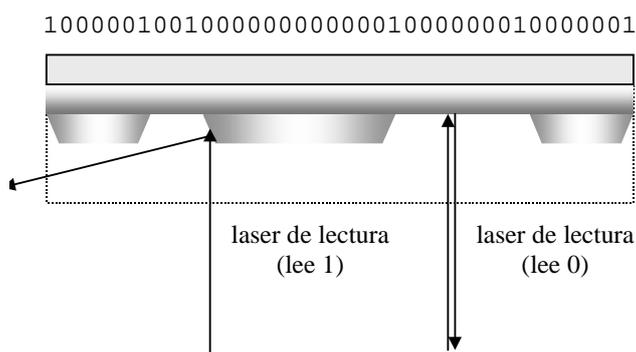
### 2.1.1. Proceso de escritura (grabación)

- La información se codifica usando un *código de canal* con el que cada byte (ASCII) queda representado por 14 bits y se graba en una cinta magnética
- La cinta es leída por un grabador de moldes para controlar un modulador óptico de un láser de potencia.
- El láser se enfoca sobre la superficie fotosensible que recubre el disco molde (que es de cristal).
- Mediante un proceso análogo al utilizado en la fabricación de CIs se revela el disco.
- Las zonas de la superficie fotosensible que han sido expuestas a la luz se convierten en hoyos, resultando un relieve exactamente igual al que tendrán los CD-ROM.
- Del molde original se obtienen varios discos denominados matrices, que son copias en negativo de los discos definitivos.
- Con las matrices se obtiene la capa de aluminio de CD-ROM.
- Finalmente se aísla del exterior, emparedándola entre la capa protectora superior y la base inferior de plástico

Con los CD-ROM se consiguen una densidad de grabación de 16.000 pistas/pulgada (t/i) muy superior a la de los discos magnéticos (los disquetes 96 t/i).

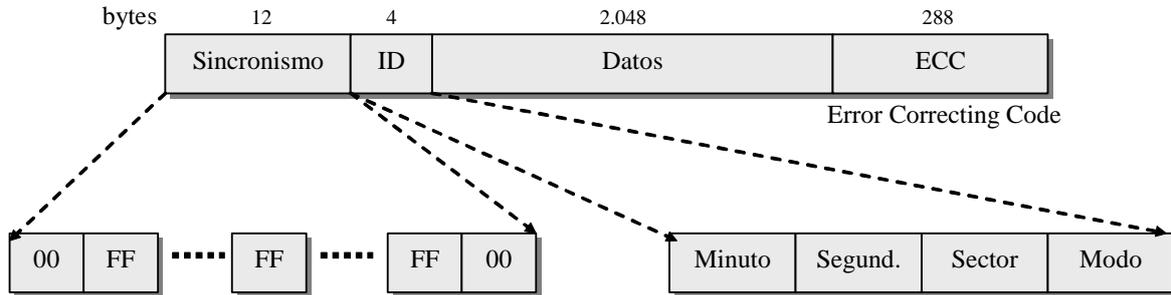
### 2.1.2. Proceso de lectura

- Se utiliza un haz de luz generado por un diodo láser de arseniuro de galio, que una lente enfoca sobre la superficie de la base de plástico, convirtiéndose en un punto de 1 micra.
- El enfoque no se realiza sobre la superficie externa de la base de plástico, sino sobre el aluminio, por lo que las partículas de polvo o rayados del exterior del disco apenas afectan a su lectura.
- La lectura se efectúa con un fotodetector que mide la cantidad de luz reflejada (genera una corriente proporcional a la radiación recibida).
- Cuando la luz incide sobre los bordes de los hoyos se desvía, de forma que apenas vuelve nada hacia el objetivo, obteniéndose así un "1".
- Cuando el haz incide sobre la superficie plana de hoyos y salientes se refleja casi sin desviación, captándose en el detector prácticamente la totalidad del haz, obteniéndose así un "0".



### 2.1.3. Codificación de la información

A nivel lógico la información se organiza en bloques (sectores) de 2.352 bytes, con el formato que aparece en la siguiente figura:

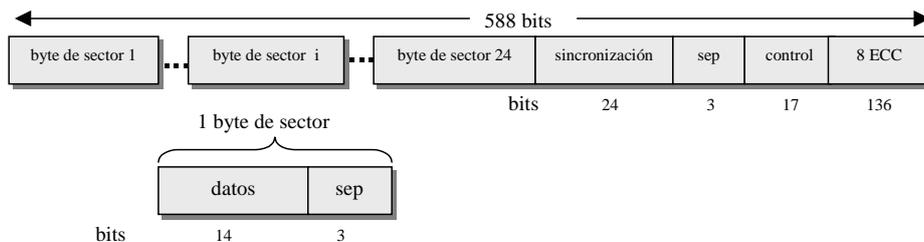


A nivel físico

Los 1's se representan por el cambio de hoyo a saliente y los 0's por las superficies de los hoyos y salientes. Cada 0,3 micras de hoyo o saliente a lo largo de la pista corresponde a un 0. La longitud mínima físicamente posible para hoyos y salientes es de 0,6 micras, lo que exige que 2 1's deben estar separados como mínimo por 2 0's. Por eso cada byte de datos se codifica con 14 bits de canal, según la siguiente tabla:

Codificación EFM (Eight to Fourteen Modulation)	
Byte	Código de canal
0000 0000	01 0010 0010 0000
0000 0001	10 0001 0000 0000
0000 0010	10 0100 0010 0000
0000 0011	10 0010 0010 0000
0000 0100	01 0001 0000 0000
0000 0101	00 0001 0001 0000
0000 0110	00 0100 0010 0000
0000 0111	00 1001 0000 0000
.....	.....

Los 2.352 bytes de cada sector se estructuran en tramas. Cada trama contiene 24 bytes de sector.



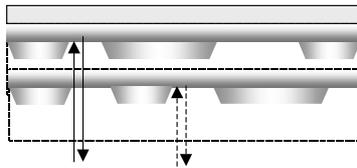
Un sector queda representado por 98 tramas (98\*24=2.352)

Bits de canal de un sector = 588 bits/trama \* 98 tramas/sector = 57.624 bits

Luego los 2.048 bytes = 16.388 bits de datos de usuario ocupan 2.352 bytes = 18.816 bits de datos lógicos y 57.624 bits físicos de canal.

**2.2. DVD (Digital Versatil Discs)**

Constan de varias capas de información sobre una misma cara del disco separadas por materiales que permiten el acceso selectivo a cada capa por medio de láseres con diferentes propiedades de reflexión.



### 2.3. WORM (Write Once, Read Many times)

Son unidades de discos ópticos similares a los lectores de CD-ROM, pero que contienen un láser de potencia, de forma que en la propia unidad se puede efectuar la grabación del disco.

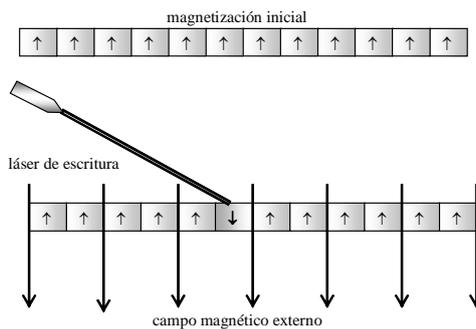
### 2.4. Discos magnetoópticos o WMRA (Write Many, Read Always)

#### 2.4.1. Proceso de escritura

Los discos vírgenes son previamente magnetizados, es decir, escritos todo a 0's, magnetización norte, o todo a 1's, magnetización sur.

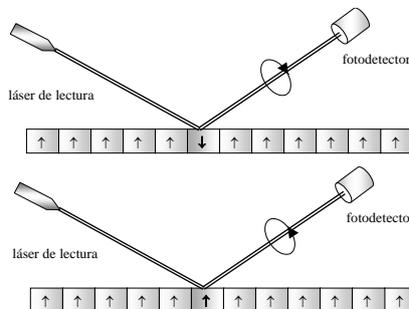
En la fase de escritura se realizan dos acciones:

1. Toda la superficie del disco se somete a un campo magnético uniforme, de sentido contrario al de magnetización inicial, y de intensidad no suficiente para alterar el estado de magnetización inicial de cada celda.
2. Simultáneamente se hace incidir un láser de unos 40 mW de potencia sobre las celdas en las que hay que escribir un valor distinto al inicial. De esta forma la celda se calienta cerca de su punto de Curie, con lo que cambia el sentido de magnetización inicial de la celda.



#### 2.4.2. Proceso de lectura

El proceso de lectura se funda en el efecto *Kerr*, que consiste en que el plano de polarización de un haz de luz gira en un sentido u otro, según el estado de magnetización de la superficie en la que se refleja. La lectura se realiza con un láser de unos 10 mW de potencia, que incide secuencialmente sobre las celdas del disco que se van a leer, captándose la luz reflejada en un fotodetector después de haber pasado por un analizador óptico.

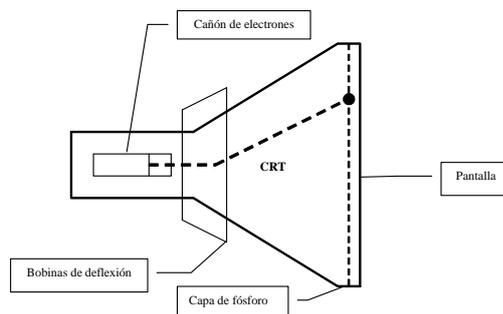


#### 2.4.3. Proceso de regrabación

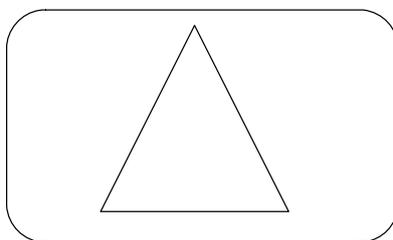
Para volver a escribir sobre un disco previamente escrito se procede en primer lugar al borrado de todas las celdas que contengan un sentido de magnetización distinto al inicial. Para ello se aplica un campo magnético externo de sentido igual al inicial y con la suficiente intensidad para invertir el sentido de magnetización de cualquier celda que no coincida con el del campo externo. La capacidad de este tipo de discos va de 0.5 a 1 GB y una velocidad de transferencia de unos 150 KB/s con tiempo de acceso de 300 ms.

### 3. Monitores de tubo de rayos catódicos (CRT)

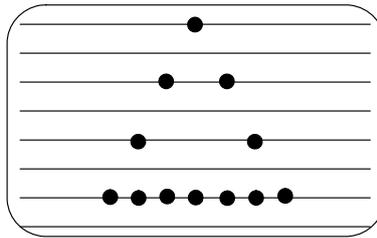
El tubo de rayos catódicos (CRT) ha ocupado un lugar preferente entre los dispositivos de salida de un computador, fundamentalmente porque comparte la tecnología básica con los tubos tradicionales de los televisores, beneficiándose de la economía de escala de las grandes series de producción. El elemento básico de un CRT es un tubo de vidrio en cuyo interior está hecho el vacío. En un extremo del tubo existe un cátodo que al calentarse emite un chorro de electrones (cañón de electrones) que puede ser controlado y enfocado por una serie de bobinas de deflexión situadas en el cuello del tubo. En el otro extremo el tubo de vidrio se ensancha y se recubre interiormente por una o varias capas de fósforo constituyendo la pantalla. Cuando el fósforo de la pantalla es impactado por el chorro de electrones, emite luz de un color que depende del tipo de fósforo. La posición del punto de emisión y la intensidad de la luz emitida son controladas por las bobinas del cuello del tubo. La persistencia de la pantalla es el tiempo que transcurre desde que se deja de emitir el haz hasta que la fosforescencia decae al 10% de su valor inicial. En la práctica este valor está comprendido entre 10 y 60 microsegundos. Por tanto, para que la imagen aparezca de forma estable y no fluctuante, la electrónica asociada a la pantalla (el controlador) deberá refrescar la imagen periódicamente a una frecuencia determinada por la persistencia.



Existen dos tipos de monitores basados en tubos de rayos catódicos, los *vectoriales* o *caligráficos* y los de *barrido* o *raster*. En un CRT de refresco *vectorial* la unidad de representación es la línea (*vector*). Generan la imagen haciendo que un haz de electrones trace sobre la superficie fosforescente de la pantalla los segmentos que forman la figura que se quiere visualizar.

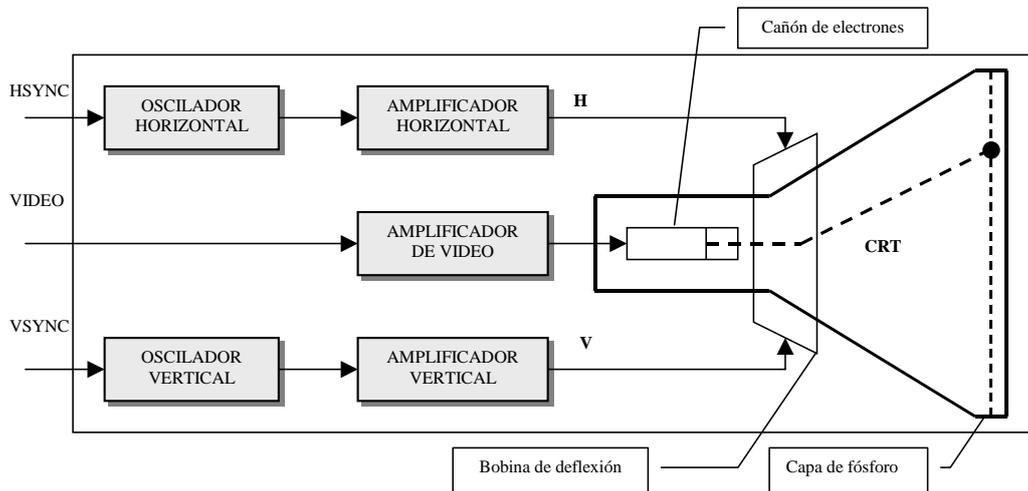


En los CRTs de barrido la unidad de representación es el punto (*pixel*). La pantalla se recorre (barre) en líneas horizontales del mismo número de puntos. Estos tubos componen la imagen modulando el haz de electrones en función de la intensidad de gris del correspondiente punto justo en el momento que el haz está situado en la posición adecuada sobre la pantalla.

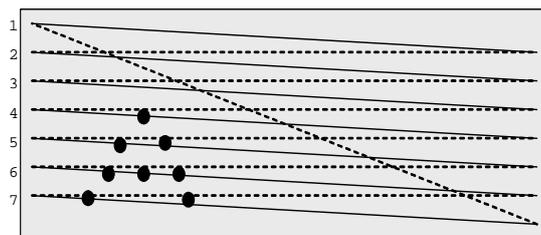


### 3.1. Principios de funcionamiento del tubo de barrido

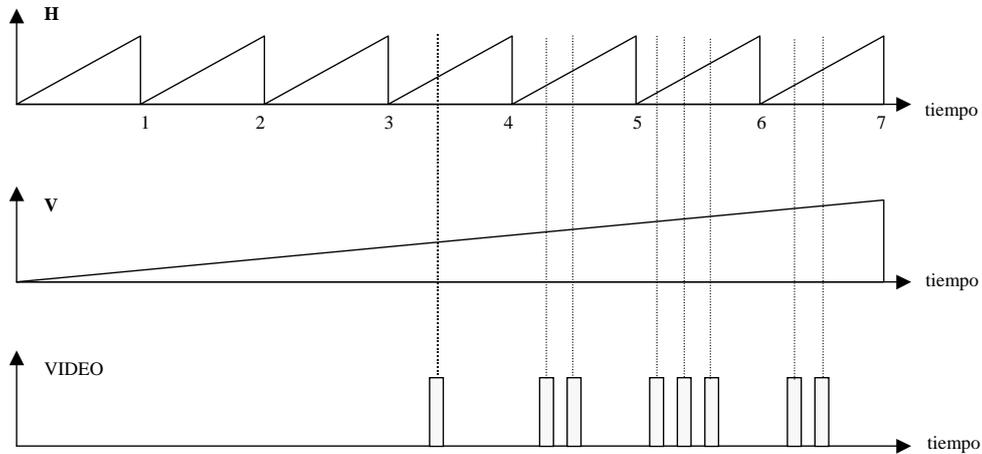
En la siguiente figura se han representado los componentes principales de un monitor monocromo (blanco y negro) de un tubo de rayos catódicos (CRT) que utiliza la técnica de barrido. Esta técnica es análoga a la utilizada por los aparatos de televisión. Las señales que actúan sobre la bobina de deflexión son generadas por dos osciladores, uno actuando sobre la componente horizontal y el otro sobre la vertical. Los osciladores son sincronizados por señales externas, *HSYNC* para la sincronización horizontal, y *VSYNC* para la vertical, después se amplifican antes de entrar en la bobina. La señal de *VIDEO* controla la intensidad del haz de electrones, y por tanto la iluminación de los puntos de la imagen sobre la pantalla.



En la siguiente figura hemos representado la forma en que se realiza el barrido de una pantalla monocromo. El haz de electrones comienza en la esquina superior izquierda y se mueve horizontalmente de izquierda a derecha. Cuando completa la primera línea horizontal vuelve al flanco izquierdo para realizar el barrido de la segunda línea, al tiempo que el haz sufre una pequeña deflexión vertical, en sentido descendente. De esta forma se genera un conjunto de líneas horizontales paralelas hasta alcanzar la esquina inferior derecha. En este punto se dice que se ha completado un barrido vertical o un marco. Después el haz de electrones vuelve a la esquina superior izquierda para repetir el proceso y así refrescar la pantalla. El refresco se hace necesario para que la imagen aparezca persistente en la retina del usuario.

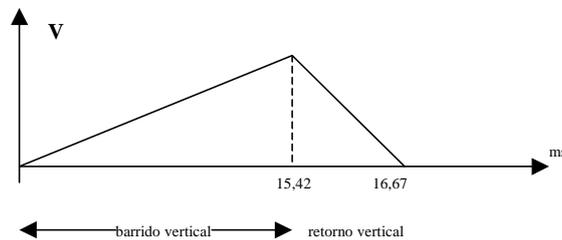


Para producir el barrido descrito los osciladores horizontal y vertical deberán generar unas señales en forma de diente de sierra, tal como las que se presentan en la figura siguiente:

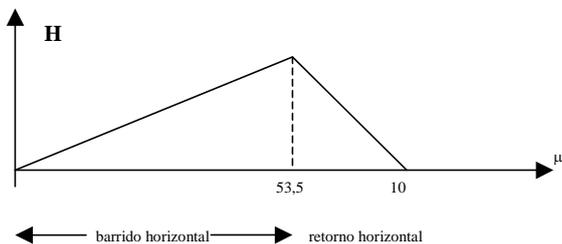


La señal de VIDEO controla la intensidad de iluminación del punto de barrido en cada instante, haciendo que se dibuje la imagen sobre la pantalla (la letra **A** mayúscula en la figura anterior). En este caso la intensidad la hemos considerado binaria, presencia o ausencia de luz (blanco o negro). En muchos CRTs la señal de VIDEO será un valor entero que permite graduar la intensidad de los pixels en una escala de grises.

Si el refresco de la pantalla se realiza a razón de 60 marcos/segundo, corresponde a un intervalo de tiempo de 16.67 ms, de los cuales sólo 15,42 ms se consumen estrictamente en el barrido vertical, los restantes 1,25 ms se emplean en el retorno vertical.



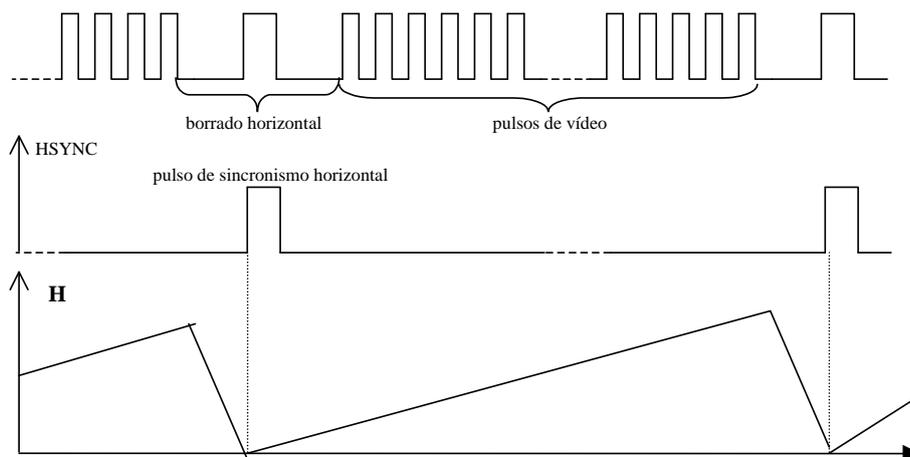
Lo mismo le ocurre al barrido horizontal. Unos valores típicos para el barrido y el retorno son los que se muestran en la siguiente figura:



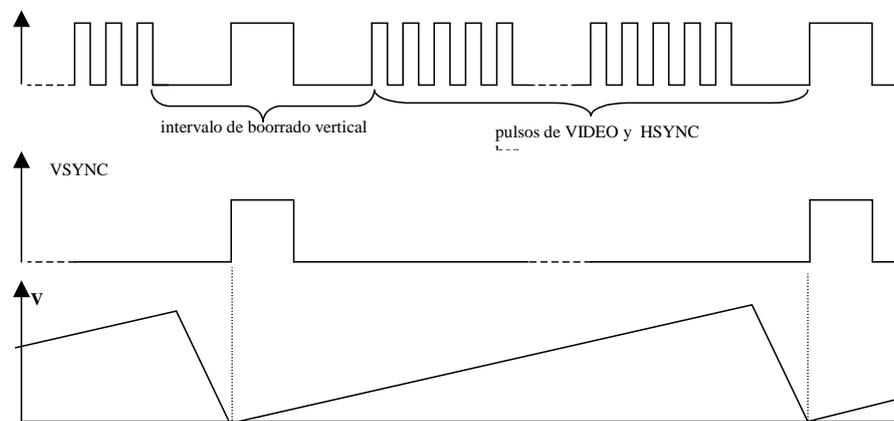
Con los valores anteriores, el número total de líneas de un marco será:  $1.542 \mu s / 63,5 \approx 242$  líneas.

### 3.1.1. Sincronismo

Las señales de barrido horizontal y vertical tienen evidentemente que sincronizarse entre sí y con la señal de VIDEO, que por simplicidad seguiremos considerando que es binaria. La sincronización del barrido horizontal y los pulsos de video se muestra en la siguiente figura:

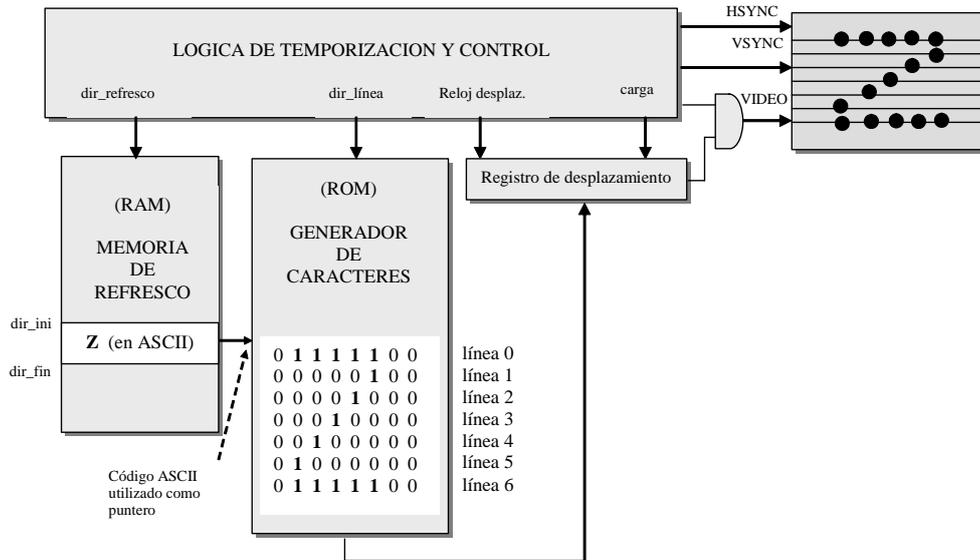


Algo equivalente ocurre para el sincronismo vertical:

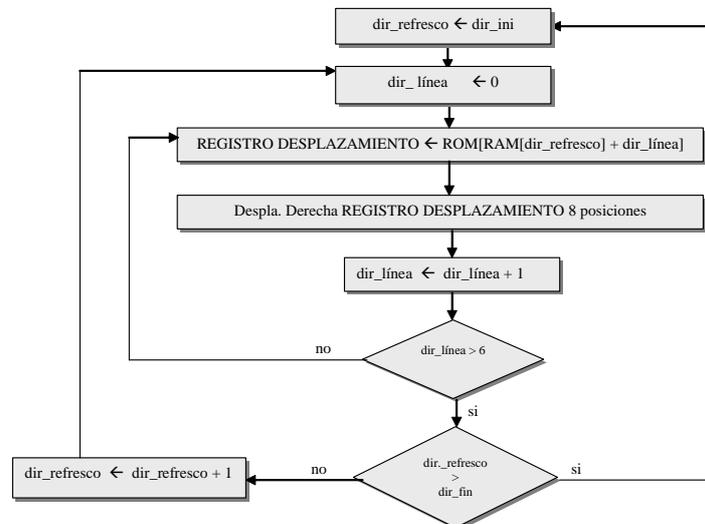


### 3.1.2. CRT de caracteres

Para entender el funcionamiento de la lógica de temporización y control de una pantalla basada en un CRT que sólo visualiza caracteres, consideraremos en primer lugar el caso aislado de generación de un carácter, la *Z*, que constituye el primer carácter de una línea almacenada entre las direcciones *dir\_ini* y *dir\_fin* en la RAM de refresco. La codificación en mapa de bits de todos los caracteres visualizables se halla en una memoria ROM, denominada Generador de Caracteres. En *dir\_ini* se encuentra en realidad la codificación ASCII de la *Z*, que se utilizará como puntero (dirección) al inicio de la zona de la ROM donde comienza la codificación en mapa de bits de la *Z*.

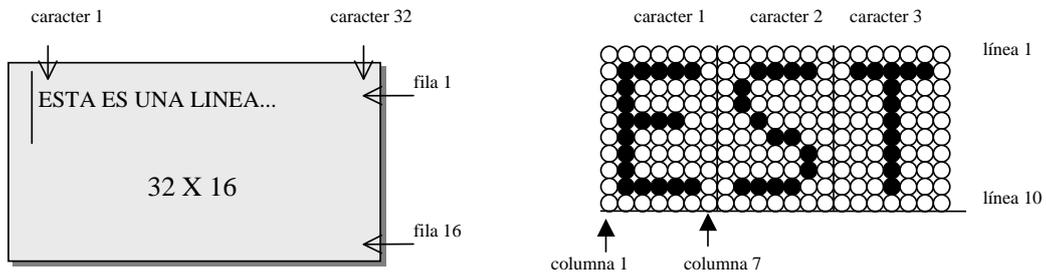


Para visualizar la Z tendremos que ir extrayendo de la ROM el valor binario de sus pixels, línea a línea, llevando cada línea al Registro de Desplazamiento que controla la señal de VIDEO, es decir, la presencia (1) o ausencia (0) de luz en la pantalla. Hemos supuesto que la resolución de cada carácter, incluyendo los puntos de separación es de  $8 \times 7 = 56 \text{ pixels}$ . El proceso completo de refresco de la línea contenida entre las direcciones *dir\_ini* y *dir\_fin* de la memoria de refresco lo hemos representado en el siguiente diagrama:

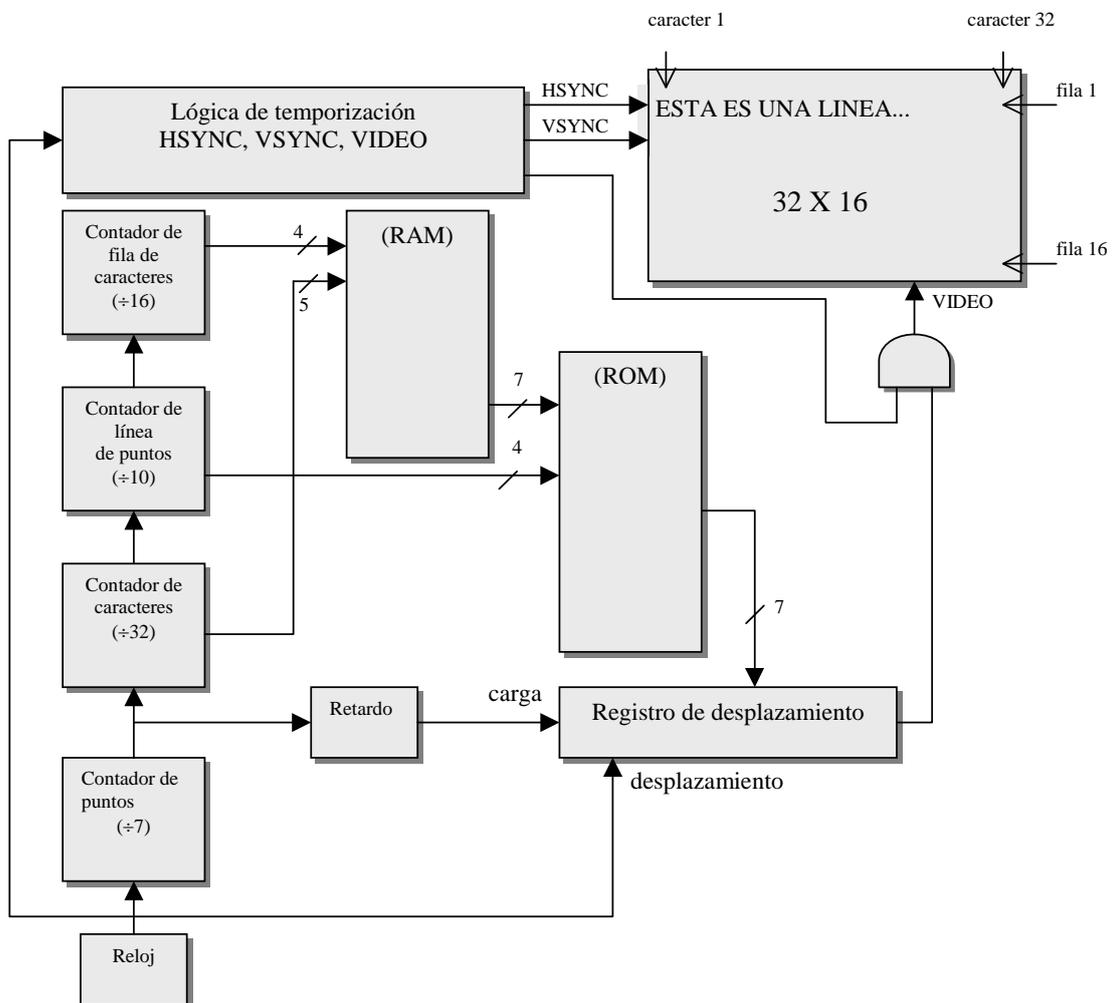


### 3.2. Controlador de un CRT de caracteres

La estructura del controlador completo de la pantalla de tubo de rayos catódicos de caracteres la componen un conjunto de contadores que actúan sobre la RAM de refresco, la ROM de generación de caracteres, el Registro de Desplazamiento que controla la señal de VIDEO y la Lógica de Temporización. Supondremos que la resolución de la pantalla es de 16 filas con 32 caracteres cada una, y cada carácter está definido por una matriz de  $7 \times 10$  puntos, tal como se muestra en la siguiente figura:



El controlador actúa en sincronización con un Reloj que genera un pulso por cada píxel a visualizar. El Reloj actúa sobre el Registro de Desplazamiento para marcar los instantes en que tiene que renovarse el punto de la línea en fase de visualización. También actúa de forma directa sobre el Contador de Puntos de cada línea de un carácter. En realidad este contador opera como divisor de la frecuencia del Reloj para cargar el Registro de Desplazamiento con las sucesivas líneas de todos los caracteres que se visualizan en la misma fila de la pantalla.



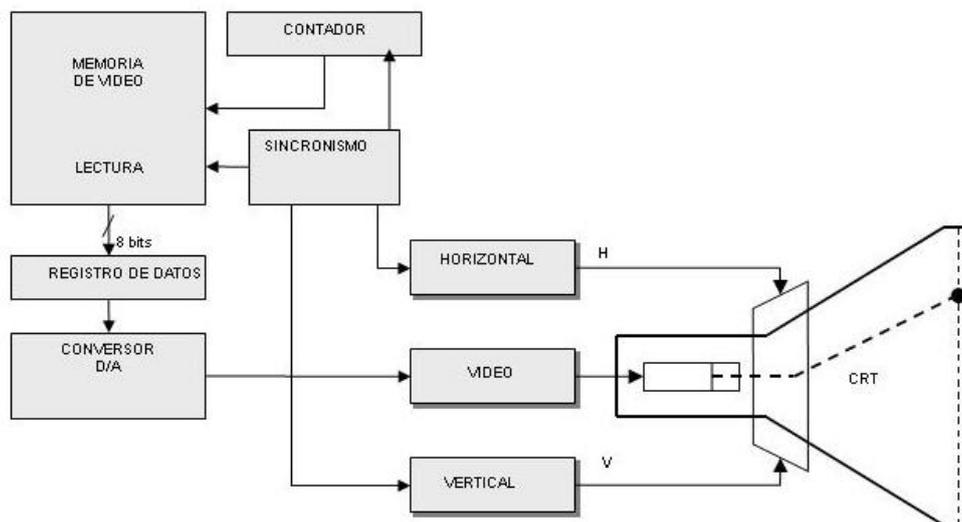
La salida del Contador de Puntos actúa sobre el Contador de Caracteres, que a su vez actúa sobre el Contador de Línea de Puntos, que finalmente actúa sobre el Contado de Fila de Caracteres. Este funcionamiento en cascada de los 4 contadores va dividiendo la frecuencia del

Reloj en los valores adecuados al recorrido de cada contador, que se corresponden con la resolución de la pantalla:

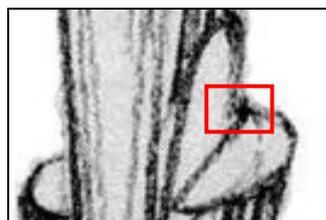
- Contador de Puntos → Puntos de cada fila de cada carácter (7)
- Contador de Caracteres → Número de caracteres de una fila de la pantalla (32)
- Contador de Línea de Puntos → Número de líneas de puntos de cada carácter (10)
- Contador de Fila de Caracteres → Número de filas de caracteres de la pantalla (16)

### 3.3. Controlador de un CRT de mapa de bits

Un CRT de mapa de bits permite visualizar en pantalla cualquier forma gráfica cuyos *pixels* hayan sido codificados previamente y almacenados en la memoria de vídeo del controlador. Esta memoria tendrá un número de posiciones (bytes) igual al número de *pixels* que componen la pantalla. Cada posición de la memoria de vídeo almacena el valor de gris (de 0 a 255) del correspondiente píxel. En la siguiente figura hemos representado un esquema general del controlador para un CRT de mapa de bits.



Un contador cíclico actúa como registro de direcciones de la memoria de vídeo en sincronismo con las señales de barrido horizontal y vertical. El byte leído de la memoria de vídeo pasa a un registro de datos cuya salida está conectada a un convertidor digital/analógico que transforma el byte en una señal analógica proporcional a su valor. Esta señal es la que modula la entrada de vídeo del CRT haciendo que el punto de la pantalla donde en ese momento se encuentre posicionado el haz de electrones por las señales de barrido horizontal y vertical, se ilumine con una intensidad proporcional al valor del byte. En la siguiente figura hemos representado el contenido de la memoria de vídeo correspondiente a una pequeña área rectangular de una figura representada en escala de grises. La intensidad de los pixels varía de 0 (negro) a 255 (blanco)

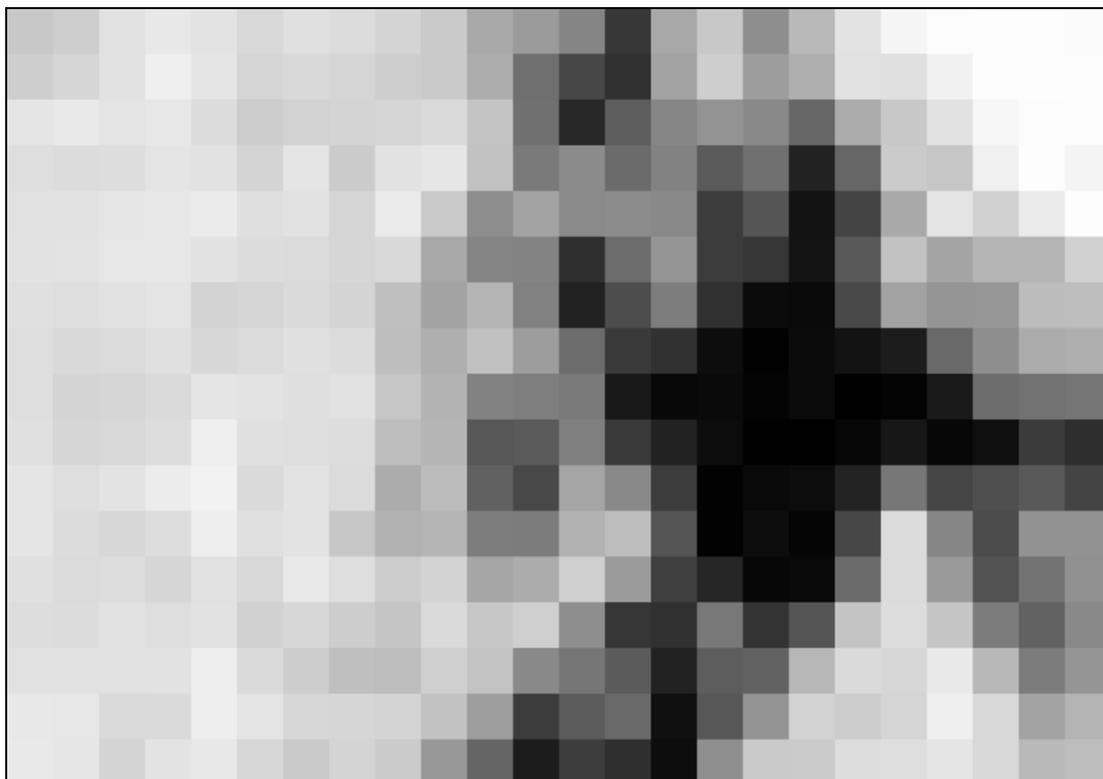


Memoria (buffer) de vídeo

200	206	225	232	227	216	224	220	212	202	169	155	133	55	168	200	142	185	227	245	252	253	253	251
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

206	214	226	239	229	213	217	213	206	202	173	111	71	48	162	206	157	175	225	224	241	253	253	252
229	232	228	231	220	204	210	212	213	218	196	112	39	93	134	147	137	102	171	200	226	247	252	252
222	220	221	228	226	212	228	203	226	230	198	122	139	107	130	90	112	34	102	203	198	240	253	244
225	226	230	231	235	223	226	213	234	201	141	163	138	140	137	60	85	19	68	169	228	209	234	253
226	227	231	231	226	220	219	214	217	168	132	132	47	108	149	60	54	20	89	194	164	181	180	209
224	222	225	228	211	213	218	212	191	163	181	129	33	77	125	48	11	9	73	162	149	150	188	189
222	217	219	224	215	219	224	220	190	175	192	157	108	58	49	13	2	11	19	28	105	142	171	174
222	212	213	218	229	227	223	225	198	179	130	126	122	25	8	10	4	11	1	3	26	109	114	117
224	214	216	221	239	224	223	221	189	181	87	90	128	57	34	12	1	0	7	23	7	15	59	46
229	222	227	236	242	218	227	219	172	188	97	72	165	137	60	2	9	12	35	119	69	79	89	68
229	220	215	221	238	224	227	219	177	181	125	124	178	189	84	2	13	5	71	219	124	76	146	146
224	219	221	214	225	216	233	222	204	204	211	166	172	207	63	38	7	9	107	219	154	83	114	144
221	220	226	222	226	209	215	222	197	218	199	207	142	54	48	120	51	84	196	221	197	124	98	137
225	225	225	226	237	218	205	191	189	207	195	137	118	91	34	93	98	183	218	215	233	184	123	149
232	231	218	218	238	228	215	214	212	194	158	60	92	105	16	87	148	210	204	213	239	216	163	180
232	230	212	227	232	215	201	212	203	152	101	28	61	47	13	143	202	203	221	222	229	216	185	189

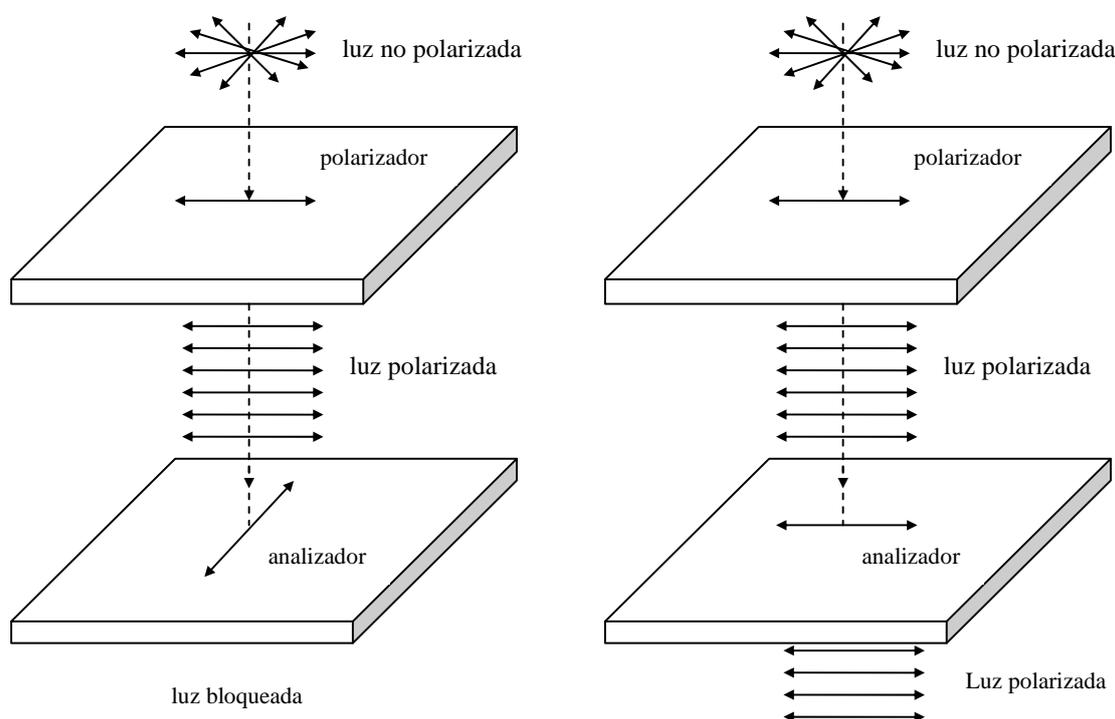
↓ Pantalla



#### 4. Pantallas de cristal líquido

El cristal líquido de tipo *twisted nematic*, inventado por Schald y Helfrich y mostrado por Fergason en 1971, representa la primera aplicación con éxito de los cristales líquidos. Se basa en la regulación del paso de luz polarizada mediante cristal líquido sometido a una tensión eléctrica.

Cuando la luz no polarizada pasa a través de un filtro polarizador, sólo se transmite un plano de polarización. Si se utilizan dos filtros polarizadores juntos, la transmisión de la luz dependerá de su orientación relativa. Por ejemplo, cuando los polarizadores se disponen de forma que sus planos de polarización son perpendiculares entre sí, la luz se bloquea. Pero cuando el segundo filtro, llamado analizador, es paralelo al primero, la luz que pasa por el primer filtro pasa también a través del segundo.

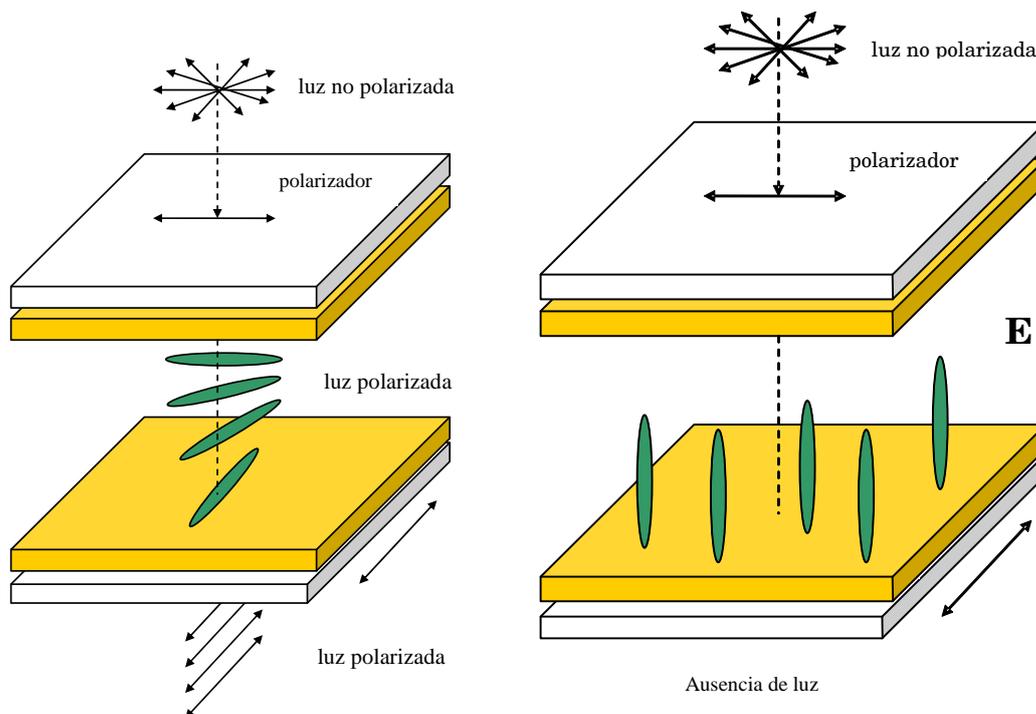


Una celda de cristal líquido *twisted nematic* está compuesta por:

- Dos superficies de vidrio transparente cubiertas de un material conductor también transparente (óxido de estaño) que actúa como electrodo
- Un regulador de precisión del *gap* entre las dos superficies de la celda.
- Dos polarizadores cruzados: el polarizador y el analizador
- Material de cristal líquido *nematic*

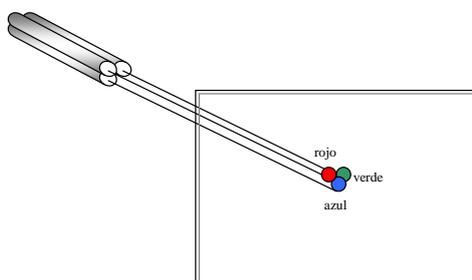
Las superficies de los electrodos transparentes en contacto con el cristal líquido se recubren de una fina capa de polímero que ha sido frotado en una dirección. Las moléculas de cristal líquido tienden a orientarse con sus ejes longitudinales paralelos a esta dirección. Las superficies de vidrio se disponen de manera que las moléculas adyacentes al electrodo superior se orienten formando un ángulo recto con el inferior. Cada polarizador está orientado con su plano de polarización paralelo a la dirección de frotación del electrodo adyacente (de esta forma el polarizador y el analizador están cruzados). En ausencia de un campo eléctrico el eje director del cristal líquido sufre un giro suave de  $90^\circ$  dentro de la celda. La luz no polarizada entra al primer filtro polarizador y sale polarizada en el mismo plano que la orientación local de las moléculas de cristal líquido. La disposición girada (*twisted*) de las moléculas de cristal líquido dentro de la celda

actúan como guía de las ondas luminosas y rotan el plano de polarización en un cuarto de giro ( $90^\circ$ ) de manera que la luz que alcanza el segundo polarizador (analizador) puede atravesarlo. En este estado la celda de cristal líquido es transparente. Cuando se aplica un voltaje a los electrodos, las moléculas de cristal líquido tienden a alinearse con el campo eléctrico resultante  $E$ , y de esta forma se pierde su propiedad de guía de las ondas luminosas, bloqueándose el paso de la luz polarizada. Cuando se desconecta el campo eléctrico, las moléculas tienden a adoptar de nuevo el estado girado (twistado) y la celda se convierte de nuevo en transparente.



## 5. Monitores de color

Los monitores de color tienen una organización similar a los de blanco y negro pero utilizan un cañón de electrones para cada uno de los tres colores básicos. Cada punto de la pantalla, *píxel*, está compuesto por tres pequeñas áreas que se iluminan de rojo, verde y azul cuando incide el correspondiente haz de electrones. La relación entre las tres intensidades de electrones produce la correspondiente intensidad de los tres colores básicos para formar el color real de la imagen.



### 5.1. Modelo aditivo del color (RGB)

Los colores rojo, verde y azul se denominan colores aditivos primarios porque todos los demás colores del espectro se pueden obtener a partir de ellos. Son los colores utilizados en el estudio de los monitores de ordenador, ya que un monitor produce los puntos de luz (*pixels*) a partir de tres

fuentes de luz, una roja, otra verde y otra azul (tres cañones de electrones en los CRT, o tres celdas básicas en las pantallas TFT). Todos los colores que se visualizan en el monitor son función de las cantidades de rojo, verde y azul que se suman en cada *pixel*. Por este motivo, uno de los modelos de color más utilizados en informática gráfica es el modelo RGB (Red, Green, Blue). Para representar un color en el sistema RGB se asigna un valor entre 0 y 255 (un byte) a cada uno de los componentes rojo, verde y azul que lo forman.

Los colores primarios o puros son los siguientes:  $RGB(255,0,0)$  = rojo claro,  $RGB(0,255,0)$  = verde claro y  $RGB(0,0,255)$  = azul claro.

Los colores secundarios son cian, magenta y amarillo, y resultan de la combinación de diferentes pares de los colores primarios en iguales intensidades:

$RGB(0,255,255)$  = cian claro = verde claro + azul claro

$RGB(255,0,255)$  = magenta claro = rojo claro + azul claro

$RGB(255,255,0)$  = amarillo claro = rojo claro + verde claro.

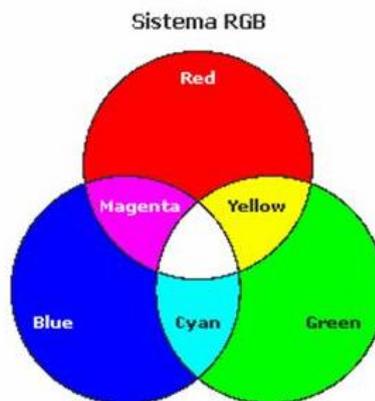
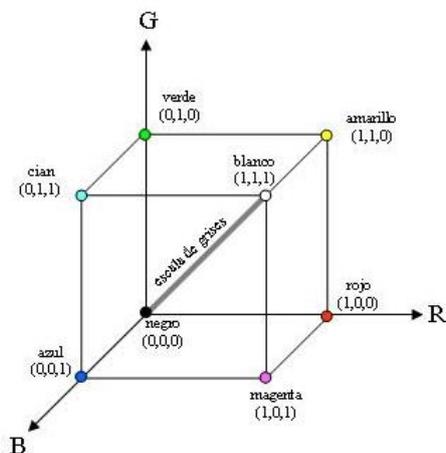
El negro es la ausencia de color (o la ausencia de luz) y su valor es  $RGB(0,0,0)$ .

El blanco es la presencia de todos los colores (en su intensidad máxima) y su valor es  $RGB(255,255,255)$ .

Todos los tonos de grises del negro al blanco se representan con tres valores iguales para los componentes rojo, verde y azul (no predomina ningún color), es decir que tienen la forma  $RGB(x, x, x)$ , por ejemplo  $RGB(192,192,192)$ .

El espacio de colores del modelo RGB puede representarse mediante un cubo unitario en el que los colores básicos han sido normalizados (de 0 a 1).

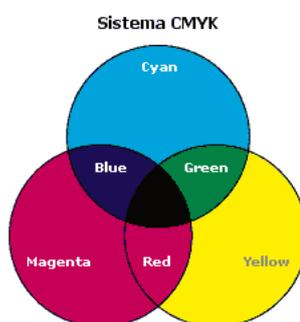
Cubo unitario del espacio de colores del modelo RGB



## 5.2. Modelo sustractivo del color (CMYK)

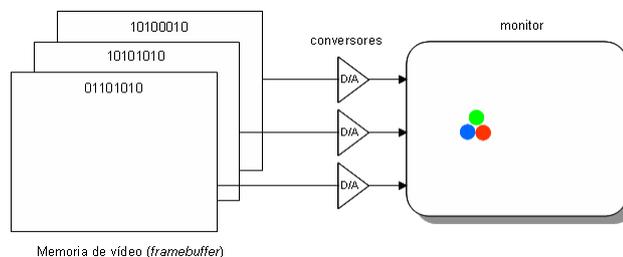
Por el contrario, los colores generados por filtrado de longitudes de onda cuando la luz incide sobre una superficie se denominan colores sustractivos. La superficie absorbe ciertas longitudes de onda y refleja otras. El color final va a depender de las longitudes de onda de la luz incidente que son reflejadas por los pigmentos de color de la superficie. Una superficie será de color azul cuando absorba todas las longitudes de onda de la luz incidente excepto las correspondientes al color azul, que son reflejadas. En cambio una superficie será blanca cuando refleje todas las longitudes de onda, y negra cuando las absorba todas. En este esquema sustractivo, los colores primarios son diferentes, concretamente el cian, el magenta y el amarillo. A partir de estos tres colores podemos

obtener casi todos los demás, salvo el blanco. La mezcla de pigmentos cian, magenta y amarillo no produce el color blanco sino un color gris. El negro tampoco es posible obtenerlo a partir de los primarios, siendo necesario incluirlo en el conjunto de colores sustractivos básicos. Por tanto, el modelo de color utilizado en pintura, imprenta y, en general, en todas aquellas composiciones en las que los colores se obtienen mediante la reflexión de la luz sobre mezclas de pigmentos es el modelo CMYK (Cyan, Magenta, Yellow, Black). En el modelo CMYK el color resulta de la superposición de tintas semitransparentes de los colores cian (azul claro), magenta (rosa intenso), amarillo y negro, y su notación se corresponde con el valor en tanto por ciento de cada uno de estos colores. De esta forma, un color cualquiera vendrá expresado en el sistema CMYK mediante la expresión (C,M,Y,K), en la que figuran los tantos por ciento que el color posee de los componentes básicos del sistema. Por ejemplo, (0,0,0,0) es blanco puro (el blanco del papel), mientras que (100,0,100,0) corresponde al color verde. El sistema CMYK se utiliza en las impresoras de inyección de tinta.



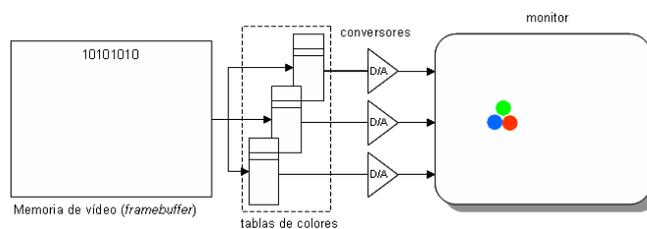
### 5.3. Memoria de vídeo de un monitor de color

Con independencia de la tecnología utilizada en la construcción de un monitor de color (CRT o LCD) deberá disponer de una memoria de vídeo (*framebuffer*) con capacidad para almacenar las tres componentes de color (canales) de cada uno de los *pixels* de la pantalla.



Cada píxel requerirá 24 bits, 1 byte para cada color. El número total de colores será pues de  $2^{24} = 16.777.216$ . Si la pantalla dispone de  $1024 \times 1024$  pixels, será necesario una memoria de vídeo de  $2^{10} \times 2^{10} \times 2^{24} = 2^8 \times 2^{36} = 64\text{GB}$ .

Una alternativa para reducir el tamaño de la memoria de vídeo consiste en codificar cada píxel con un menor número de bits y utilizar el valor binario del píxel como índice a una paleta de colores donde se encuentran las tres componentes RGB correspondientes al color codificado con el valor del índice.



## 6. Unidad de procesamiento gráfico (GPU)

### 6.1. Evolución de los controladores gráficos

La primera tarjeta gráfica para PCs fue desarrollada por IBM en 1981, la MDA (*Monochrome Display Adapter*), que trabajaba en modo texto y era capaz de representar en pantalla 25 líneas de 80 caracteres. Contaba con una memoria de vídeo de 4KB con capacidad para almacenar una sola página. Se utilizó con monitores monocromo de color verde. Después se sucedieron diversas tarjetas gráficas que resumimos en la siguiente tabla:

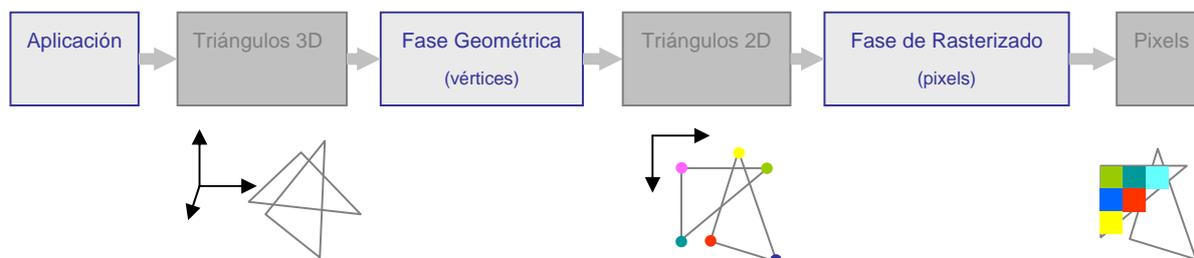
	Año	Modo Texto	Modo Gráficos	Colores	Memoria
MDA ( <i>Monochrome Display Adapter</i> )	1981	80*25	-	1	4 KB
CGA ( <i>Color Graphics Adapter</i> )	1981	80*25	640*200	4	16 KB
HGC ( <i>Hercules Graphics Card</i> )	1982	80*25	720*348	1	64 KB
EGA ( <i>Enhanced Graphics Adapter</i> )	1984	80*25	640*350	16	256 KB
VGA ( <i>Video Graphics Array</i> )	1987	720*400	640*480	256	256 KB
SVGA ( <i>Super Video Graphics Array</i> )	1989	80*25	1024*768	256	2 MB
XGA ( <i>Extended Graphics Array</i> )	1990	80*25	1024*768	65K	1 MB

La evolución de los controladores gráficos dio un giro importante en 1995 con la aparición de las primeras tarjetas 2D/3D, fabricadas por Matrox, Creative, S3 y ATI, entre otros. Dichas tarjetas cumplían el estándar SVGA, pero incorporaban funciones 3D. Se inició la era de las Unidades de Procesamiento Gráfico (GPU).

Las actuales GPUs han ido incorporando al hardware buena parte de las operaciones implicadas en el *renderizado* de imágenes 3D sobre pantallas 2D. Por ello es necesario un conocimiento mínimo de las complejas operaciones que tienen lugar en este proceso.

### 6.2. Etapas en el procesado gráfico (*pipeline* gráfico)

El procesado gráfico de una escena 3D que forma parte de una aplicación se puede organizar como un proceso segmentado (*pipeline*) compuesto de dos fases principales: la *fase geométrica* y la *fase de rasterizado*, tal como aparece en la siguiente figura.

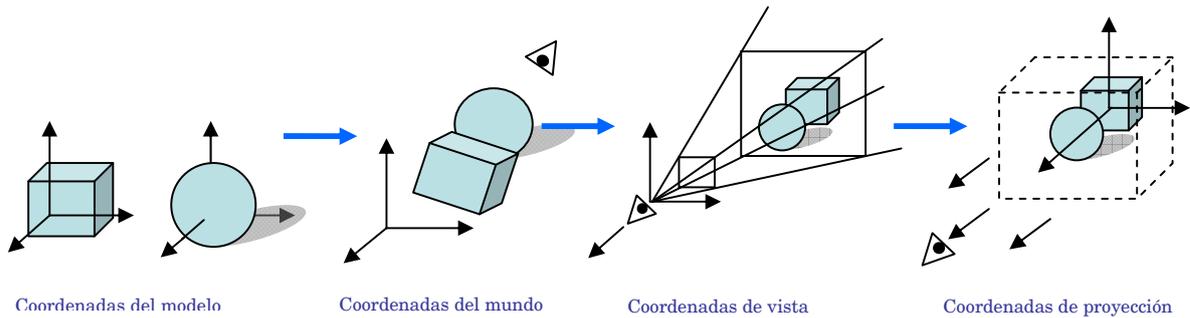


La *fase geométrica* opera sobre cada vértice de los triángulos que componen los objetos de la escena, transformando la posición 3D en una posición 2D de la pantalla y calculando sus atributos.

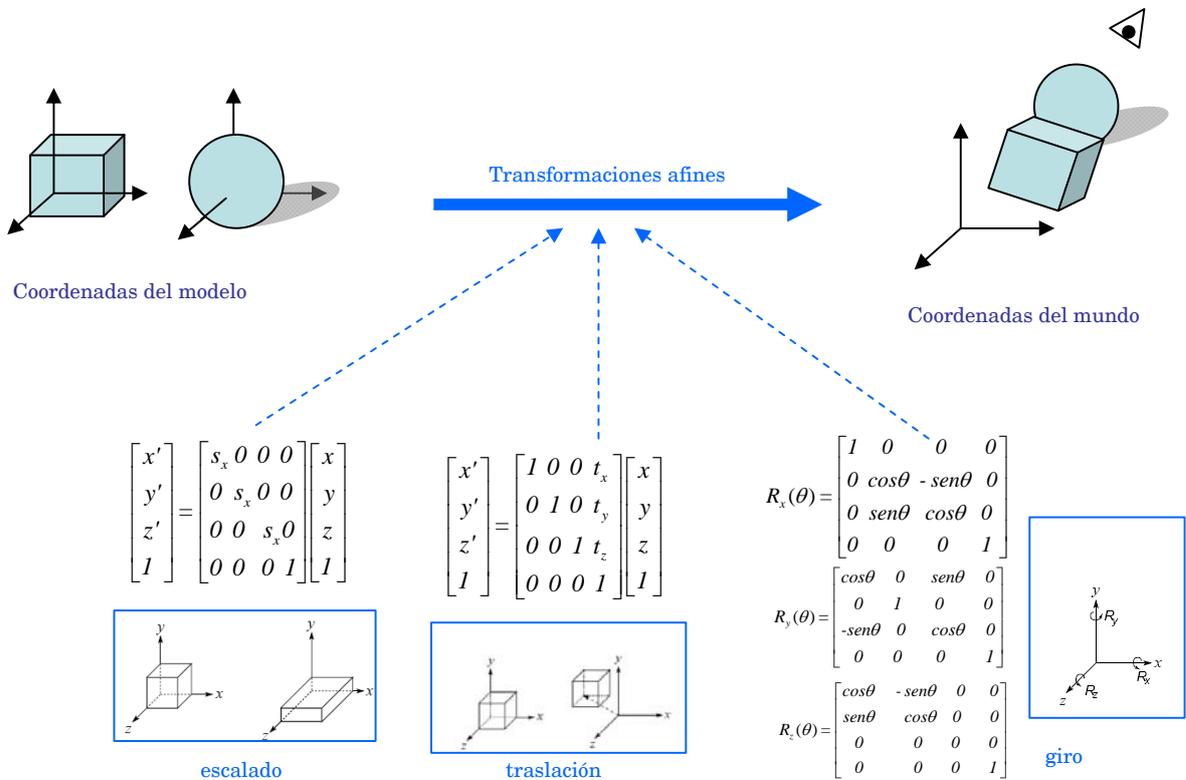
La fase de rasterizado opera sobre cada triángulo, rasterizándolo, interpolando los atributos de los vértices a lo largo del triángulo, y resolviendo la visibilidad.

### 6.2.1. Fase geométrica (vértices)

Durante la fase geométrica se realizan una serie de transformaciones matemáticas antes de llegar a calcular las coordenadas de pantalla. La primera transforma las coordenadas locales de los objetos de la escena, denominadas *coordenadas del modelo*, en coordenadas generales y únicas para toda la escena, denominadas *coordenadas del mundo*. Una vez fijado el punto de vista del observador de la escena (cámara) las coordenadas del mundo se transforman en *coordenadas de vista*, en las que solo aparecen los objetos de la escena vistos desde la cámara.



Estas transformaciones se realizan en coordenadas homogéneas y vienen determinadas por las correspondientes matrices de transformación. Por ejemplo, para obtener las coordenadas del mundo se aplican a cada uno de los vértices de la escena tres tipos de transformaciones afines: translación, giro y escalado, cuyas correspondientes matrices aparecen en la siguiente figura.



Dentro de la fase geométrica tienen lugar otras dos operaciones: el recorte (*clipping*) y la ocultación (*culling*). El **recorte** descarta parcial o totalmente los triángulos que quedan fuera del ángulo de visión definido para la imagen. La **ocultación** elimina los triángulos de los objetos que quedan ocultos tras su cara visible.

### 6.2.2. Fase de rasterizado (pixels)

Esta fase consta de una serie de operaciones realizadas sobre los pixels de la imagen, de las que las más importantes son las siguientes:

#### 6.2.2.1. Conversión por barrido (*scan conversion o rasterizer*)

Esta etapa se encarga de convertir las imágenes 3D en una matriz 2D de puntos donde se guarda información acerca del color de cada uno de los pixels. Para ello se utiliza el *frame buffer* o memoria de vídeo, un *array* cuyos elementos son ternas RGB que definen el color para cada uno de los puntos 2D asociados con la conversión de los polígonos 3D.

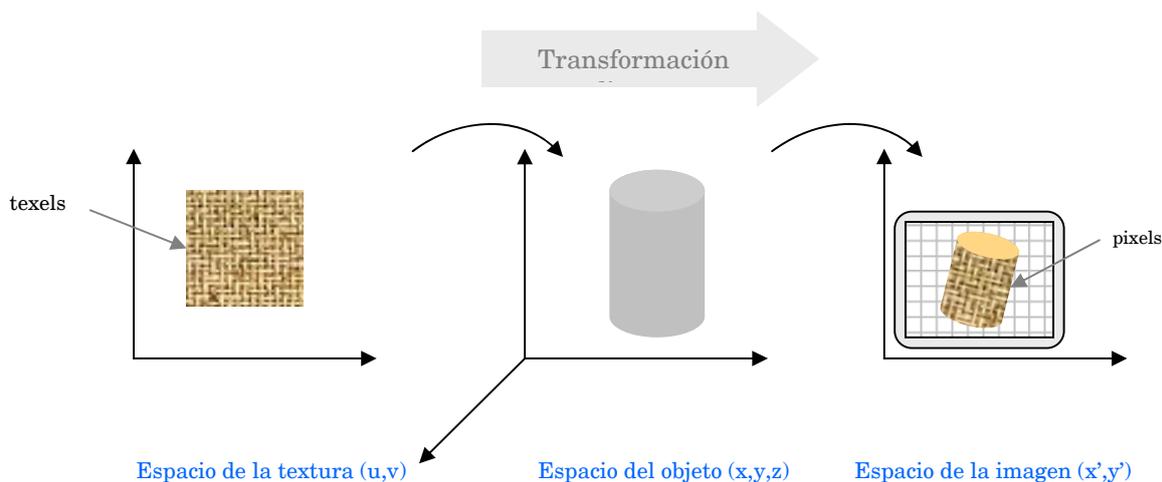
#### 6.2.2.2. Ocultación-z: algoritmo de z-buffer (z-culling)

El *z-buffer* es una memoria encargada de gestionar las coordenadas de profundidad de los *pixels* de las imágenes. Y el algoritmo de z-buffer es una solución al problema de decidir qué *pixels* de una escena renderizada son visibles y cuales ocultos

Cuando se renderiza un objeto la profundidad de cada píxel generado (cordenada *z*) se almacena en el *z-buffer*. Este buffer se organiza como un *array* de 2 dimensiones (*x-y*) con un elemento por cada *pixel* de la pantalla. Si algún otro objeto de la escena se tiene que renderizar en el mismo *pixel*, el algoritmo compara las dos profundidades y elige el más cercano al observador. La profundidad elegida es entonces salvada en el *z-buffer*, reemplazando a la antigua. El z-buffer permitirá, pues, reproducir correctamente la percepción de que los objetos cercanos ocultan a los más lejanos.

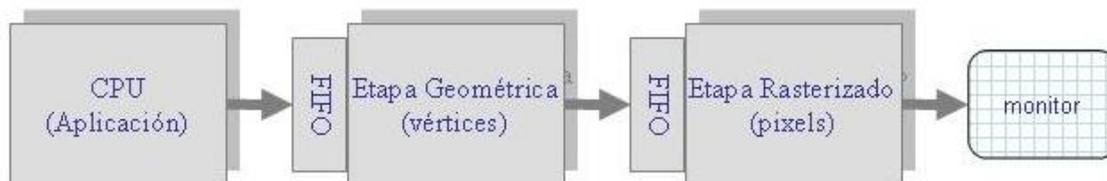
#### 6.2.2.3. Texturizado

Una textura es una imagen utilizada como superficie para cubrir los objetos tridimensionales de la escena gráfica. El texturizado de los objetos produce una apariencia más real de los mismos, confiriéndoles el aspecto de distintos materiales. Este proceso tiene un alto grado de complejidad puesto que el mapa de texturas debe transformarse en función del ángulo, tamaño y posición de los polígonos a los que recubre.

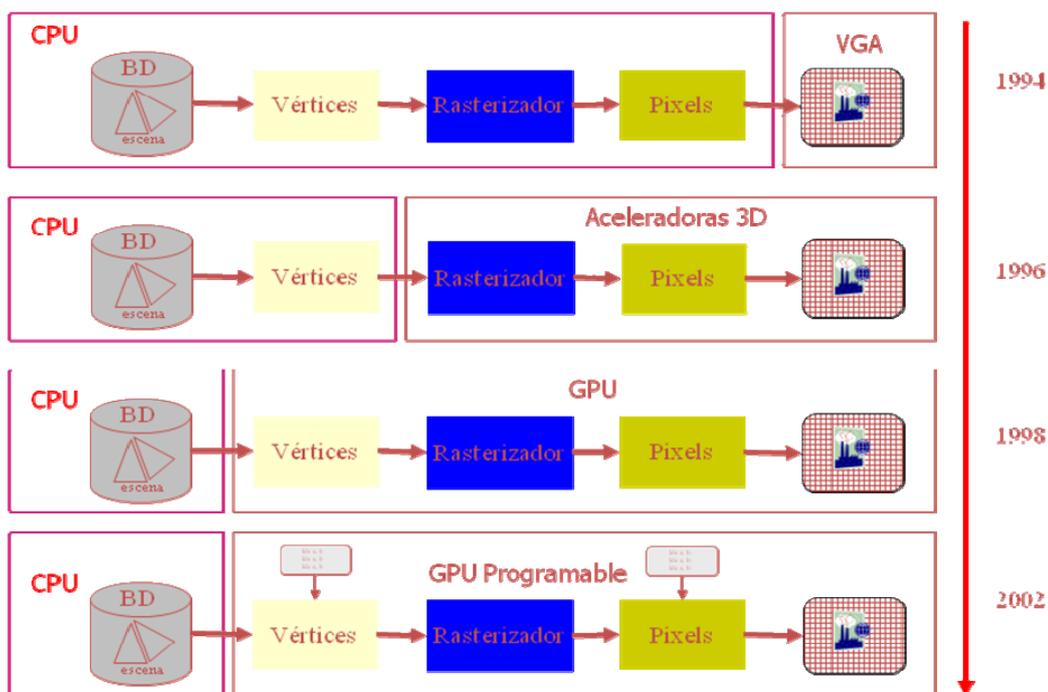


## 6.3. Procesadores gráficos

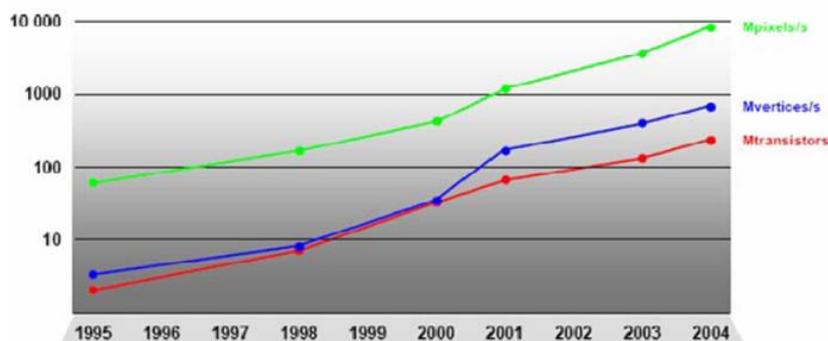
Los actuales procesadores gráficos explotan en mayor o menor grado el paralelismo implícito (temporal y espacial) en el proceso de renderizado de las imágenes 3D. El paralelismo temporal o segmentado surge de forma natural del propio proceso de renderizado, de manera que las fases geométrica y de rasterizado se realizan de forma segmentada sobre sendas etapas del procesador conectadas por las correspondientes memorias FIFOs.



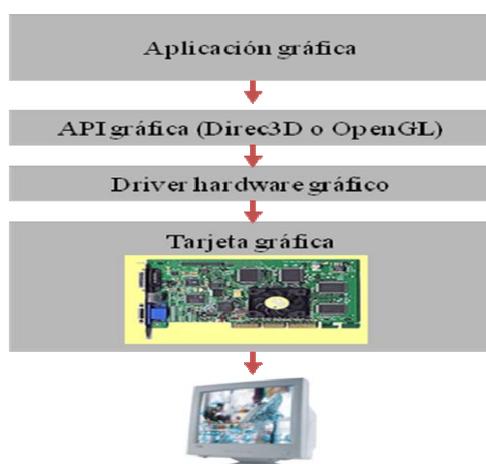
Desde el año 1994 se han sucedido una serie de generaciones de GPUs que han ido incorporando al hardware y paralelizando de forma progresiva las operaciones del renderizado gráfico, algunas de las cuales hemos descrito en el apartado anterior.



En la siguiente gráfica hemos representado la evolución de las GPUs en función de la velocidad de procesamiento de vértices y pixels ,así como el número de transistores que integran.



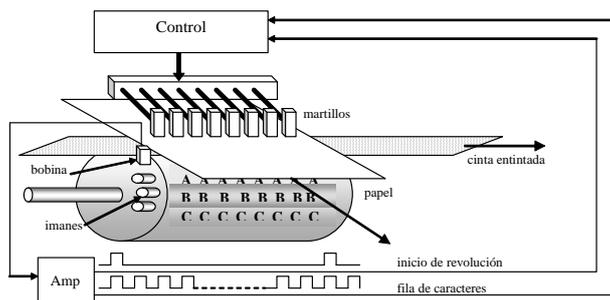
La GPU constituye el elemento principal (chip) de una tarjeta gráfica. La tarjeta es gestionada desde una API gráfica (normalmente Direct3D o OpenGL) a través de los drivers específicos de la tarjeta.



## 7. Impresoras

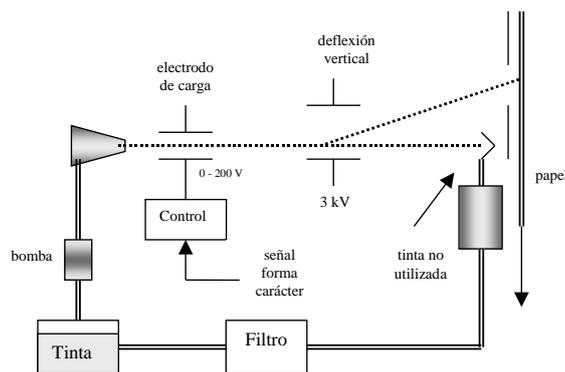
### 7.1. Impresora de martillo

La impresora de martillos se ha utilizado mucho en aplicaciones que operan en ambientes industriales, ya que su sencillez la hace muy robusta. Consta de un cilindro con el mismo carácter repetido a lo largo de toda una generatriz, y tantas generatrices como caracteres imprimibles. También dispone de una batería de martillos que se accionan eléctricamente, tantos como veces se repite el carácter en una generatriz. Entre los martillos y el cilindro se desliza una cinta entintada y el papel. Cada generatriz lleva asociada un pequeño imán en su extremo que genera un pulso cada vez que pasa debajo de una pequeña bobina. También existe un imán asociado a todo el cilindro que genera un pulso al inicio de una revolución. Con la información de los dos trenes de pulsos, el Control conoce en todo momento el carácter que se halla debajo de la batería de martillos. Sólo tiene que accionar los martillos correspondientes a las posiciones de la línea de escritura donde deba aparecer el carácter bajo los martillos. En general la escritura de una línea requerirá una vuelta completa del cilindro.



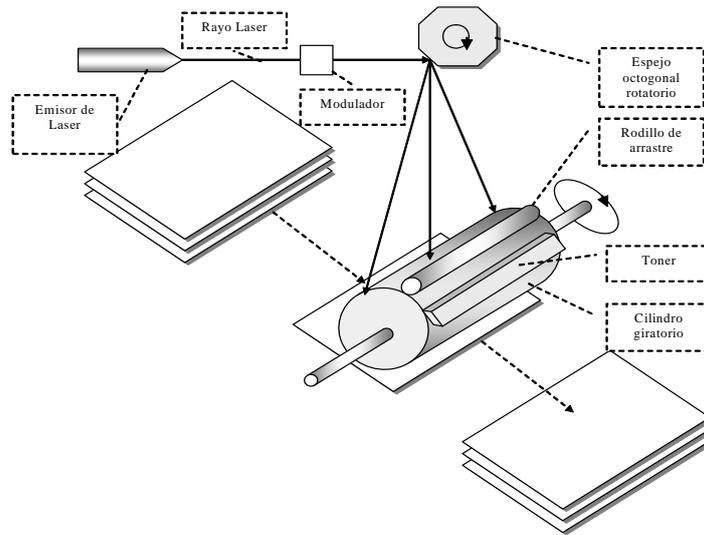
### 7.2. Impresora de chorro de tinta

La impresora de chorro de tinta utiliza un principio de funcionamiento análogo a las pantallas caligráficas de rayos catódicos. Controla la desviación eléctrica que experimenta un chorro de tinta ionizada para dibujar sobre el papel la forma deseada. En este caso la información de la forma a dibujar se traduce en una diferente ionización del chorro de tinta para que se desvíe de la forma adecuada cuando pasa por unos electrodos a potencial fijo en su camino hacia el papel. La ionización de la tinta se produce en un electrodo de carga que se controla con la señal del carácter a imprimir. La impresora dispone de un dispositivo para recoger la tinta no utilizada y devolverla al circuito de alimentación de la cabeza que emite el chorro. Las impresoras de inyección de tinta pueden producir texto e imágenes de alta calidad en blanco y negro o color.



### 7.3. Impresora láser

El principio de funcionamiento de la impresora láser se asemeja al de la pantalla de barrido de rayos catódicos. La imagen eléctrica a imprimir la forma un rayo láser sobre un cilindro recubierto de un material fotoconductor giratorio cuya superficie se corresponde con la de la página impresa. El láser opera generatriz tras generatriz. Cada generatriz del cilindro se corresponde con una línea de puntos de la página. La imagen eléctrica se transforma en imagen real por la aplicación de un polvo de carbón denominado *toner*. El cilindro impregnado de *toner* se transfiere a la página de papel con la ayuda del calor y la presión.



## 8. Ratón

