

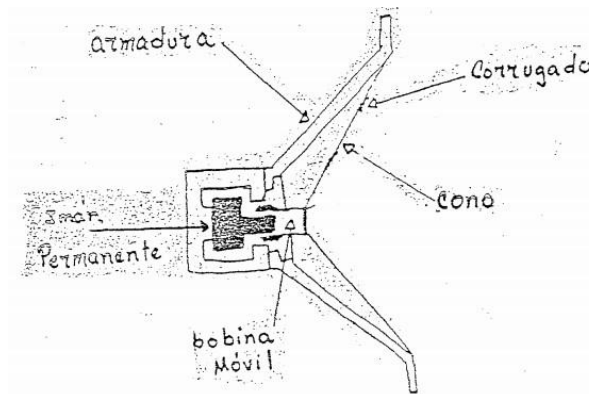
SONIDO

COMO SE PRODUCEN LOS SONIDOS EN LA BOCINA

La figura 1 ilustra la construcción típica de una bocina (el armazón es la pieza principal de soporte). El cono (o también conocido como diafragma) está hecho de un papel especial y los extremos están cimentados al armazón. Pegados al círculo interno del cono hay una forma cilíndrica que sostiene la bobina móvil.

Esta bobina penetra en un pequeño espacio donde se halla colocado permanentemente el imán. Hay una interacción entre el campo magnético del imán y el de la bobina móvil, que produce un movimiento hacia adentro y hacia afuera en el cono de la bocina, esto ocurre cuando diferentes cargas de energía llegan a la bobina. Hay que tener claro que si éstas son constantes no producen dicho movimiento, ni el sonido, ya que el cono no vibra y por lo tanto no mueve el viento que es el encargado de llevar las ondas sonoras hasta nuestros oídos.

El cono de papel sigue movimientos de la bobina produciendo el sonido. Este está corrugado para que tenga suficiente flexibilidad que le permita vibrar sin dificultad.



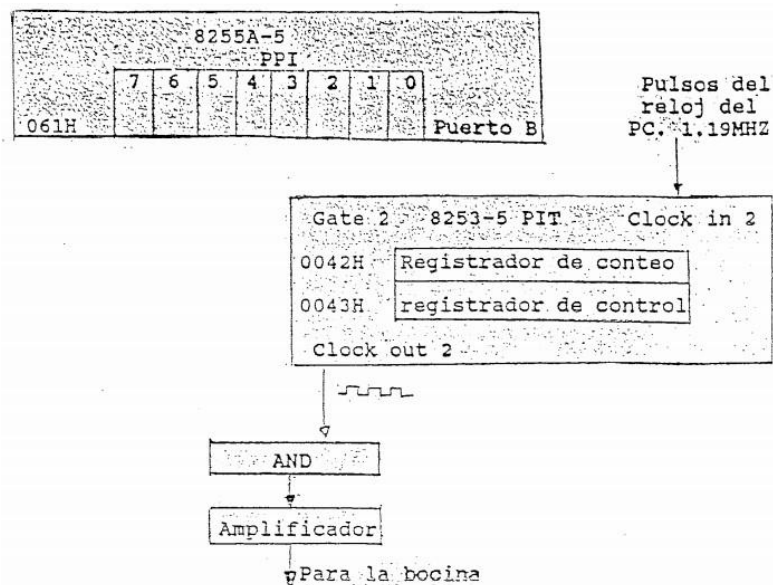
(Figura 1)

IMPORTANCIA DE LOS PULSOS DE RELOJ EN UNA PC

En primer lugar debemos tener claro que todas las computadoras traen un reloj incorporado, dado que el sistema necesita estar sincronizado; es decir, tener un ordenamiento en el tiempo de ejecución de cada una de las actividades desarrolladas por el computador. Entonces cualquier sistema basado en el microprocesador 8088 requiere una lógica adicional encargada de generar las señales (pulsos de reloj) de sincronización para todo el sistema. Los pulsos de reloj determinan la velocidad del funcionamiento del sistema.

INTERFACE DE LA BOCINA DE LA PC IBM

Dos chips programables de interface, uno el Programable Peripheral Interface (PPI) 8255A-5 y el Programable Interval Timer (PIT) 8253-5 son involucrados en la interface de la bocina como muestra la figura 2. El sonido básico (señal enviada al amplificador de la bocina), es una onda cuadrada producida por el PIT. El período de la onda cuadrada (que es el tono del sonido), depende de uno de los valores de los dieciséis bits almacenados en el registrador de conteo del PIT. Este registrador de conteo (los 16 bits) tiene la dirección de E/S 0042H.



(Figura 2)

Para el interface de la bocina almacenaremos el valor B6H en el registrador de control.

En general los bits 6 y 7 son usados para seleccionar cual canal está siendo programado, esto depende de los valores que en ellos se almacenan, el resto de los bits en este registrador definen como ese canal operará y como se comunicará con su registrador de conteo (que está en la dirección 0041H).

Dado que el registrador de conteo es de 16 bits de ancho y su puesto de acceso es solamente de 8 bits, se necesitan dos accesos de Entrada/Salida para leer o escribir el registrado de conteo, por eso es que en los bits 4 y 5 del registrador de control cargamos el valor de uno, que indica que primero se almacena el byte menos significativo (LSB) y luego el byte más significativo (MSB).

Esto hay que tenerlo en cuenta a la hora de escribir un programa; porque nosotros podemos leer el registrador de conteo usando dos instrucciones IN sucesivas, la primera retornará el byte menos significativo del registrador de conteo y la segunda retornará el byte más significativo de dicho registrador.

Similarmente, podemos escribir el registrador de conteo (en el orden LSB-MSB) con dos instrucciones OUT sucesivas.

COMO ENCENDER O APAGAR LA BOCINA DEL COMPUTADOR.

Cuando nosotros queremos que el computador produzca un sonido primero debemos encender o activar la bocina, para esto los bit 0 y 1 del puerto B (con dirección de Entrada/Salida 0061H) deben tener el valor de 1, y si deseamos desactivarla, el valor de 0.

PROGRAMACIÓN EN LENGUAJE ENSAMBLADOR PARA GENERAR UN SONIDO.

A continuación se explica lo que hace y cómo funciona el programa dado en el ejemplo 8.1; este programa produce un tono cuya graduación (o timbre) depende del valor cargado en el registrador DX y la duración del valor dado en el registrador BX.

Para probar el sonido del programa, guardaremos el valor de graduación 00300H en el registrador DX y el valor de duración 000FH en el registrador BX.

ONDAS RECTAS

- **Frecuencia:** Es una magnitud que mide el número de repeticiones por unidad de tiempo de cualquier fenómeno o suceso periódico.
- **Amplitud:** Distancia vertical entre la cresta y el punto medio
- **Longitud de onda:** Distancia que hay entre dos crestas consecutivas de dicho tamaño.
- **Período:** El periodo es el tiempo que tarda la onda en ir de un punto de máxima amplitud al siguiente.

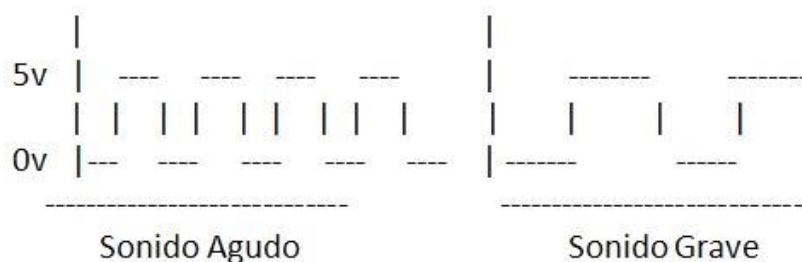
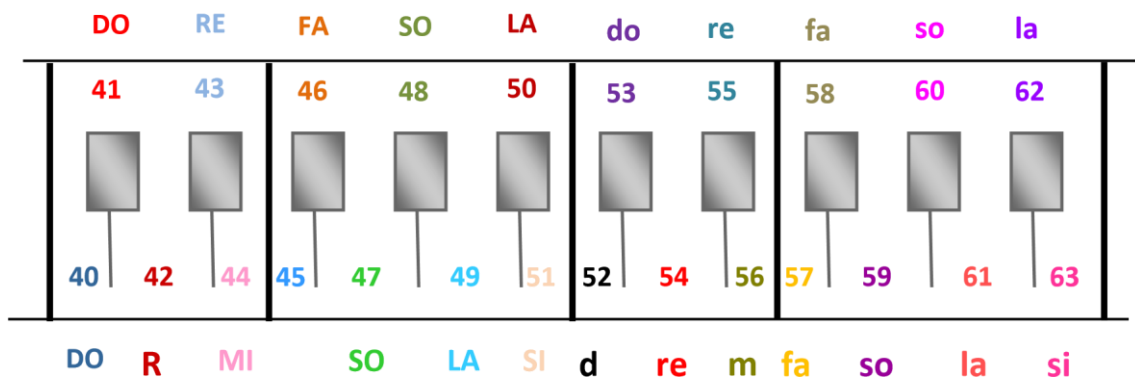


TABLA DE FRECUENCIAS POR OCTAVAS

	Octava 0	Octava 1	Octava 2	Octava 3	Octava 4	Octava 5	Octava 6	Octava 7
DO	16,35	32,70	65,41	130,81	261,63	523,25	1046,50	2093,00
DO#	17,32	34,65	69,30	138,59	277,18	554,37	1108,74	2217,46
RE	18,35	36,71	73,42	146,83	293,66	587,33	1174,66	2349,32
RE#	19,45	38,89	77,78	155,56	311,13	622,25	1244,51	2489,02
MI	20,60	41,20	82,41	164,81	329,63	659,26	1328,51	2637,02
FA	21,83	43,65	87,31	174,61	349,23	698,46	1396,91	2793,83
FA#	23,12	46,25	92,50	185,00	369,99	739,99	1479,98	2959,96
SOL	24,50	49,00	98,00	196,00	392,00	783,99	1567,98	3135,96
SOL#	25,96	51,91	103,83	207,66	415,30	830,61	1661,22	3322,44
LA	27,50	55,00	110,00	220,00	440,00	880,00	1760,00	3520,00
LA#	29,14	58,27	116,54	233,08	466,16	923,33	1864,66	3729,31
SI	30,87	61,74	123,47	246,94	493,88	987,77	1975,53	3951,07

TABLA DE FRECUENCIAS SIMPLES



EJEMPLO EN ENSAMBLADOR UTILIZANDO LA TEORIA ANTES VISTA

```

pila segment para stack db 256 dup(0)
pila ends
codigo segment para public
principal proc far assume cs:codigo
;prólogo del programa push ds mov ax,0
push ax
;empilar los 4 registradores
push ax push bx push cx push
dx
mov dx,0300h ;frecuencia mov
bx,000fh ;duración del sonido
mov al, 0b6h out 43h,al mov al,dl ;se guarda
el byte menos significativo out 42h,al ;en el
registrador de conteo del PIT mov al,dh ;se

```

```

guarda el byte menos significativo out 42h,al
;en el registrador de conteo del PIT
in al,61h ;Lee el puerto B del PPI en el caso
que
;reciba una señal externa
mov ah,al or al,03h ;en el caso que no haya valor
externo se
;coloca 3 que en binario es 11 out 61h,al
;con el valor 11 se envía a la
;dirección 61h y eso enciende la bocina
tn1:mov cx,1118h ;lazo ejecuta en tiempo de
1/16 ;segundos tn2:loop tn2 dec bx ;decrementa
valor de bx jne tn1 ;si bx <> 0 ir a tn1 mov
al,ah out 61h,al ;con esto se desactiva la
bocina
;desempila los registros
pop dx pop cx pop bx pop
ax ret principal endp
codigo ends end
principal

```