FUNDAMENTOS DE INGENIERIA DE SOFTWARE.

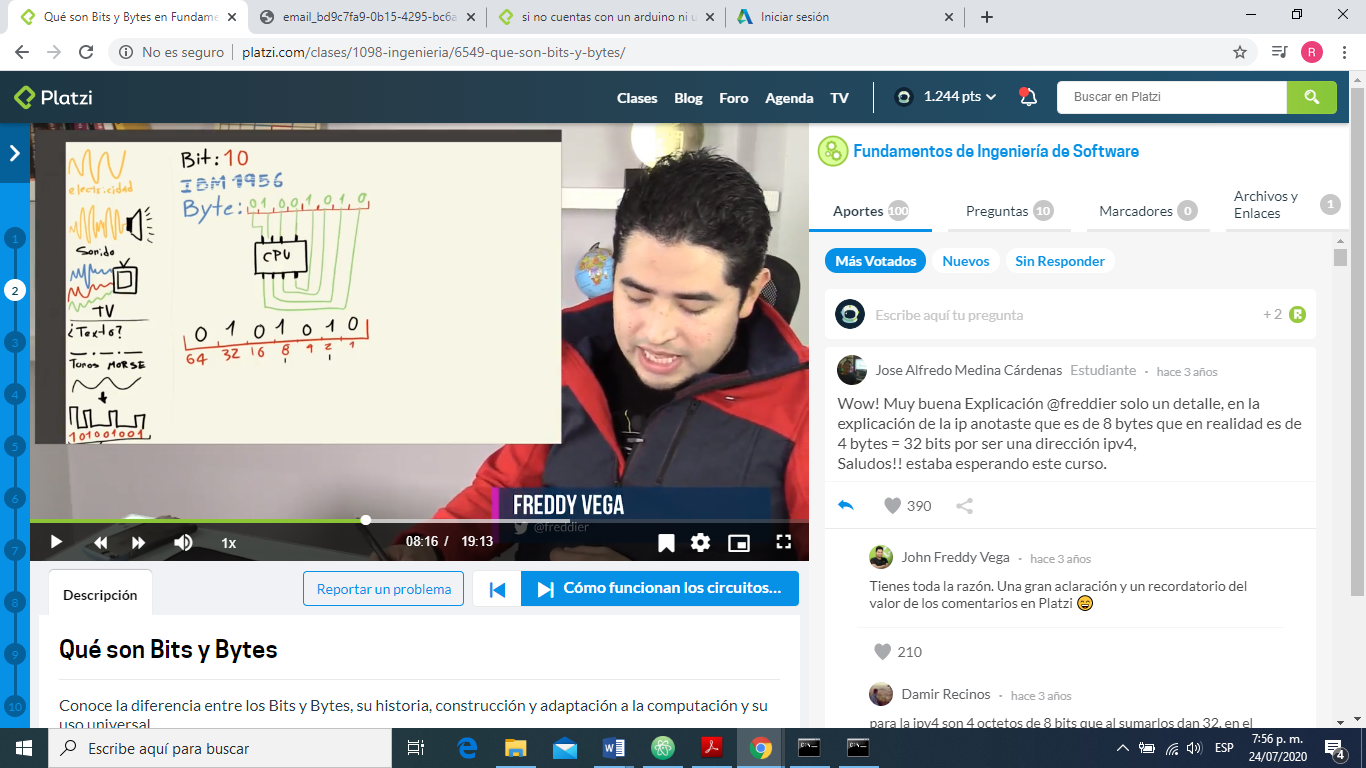
Onda digital es una señal eléctrica que tiene subidas y bajadas representadas por 0 y 1.

Bit = es una subida de onda o bajada de onda o 1 o un 0.

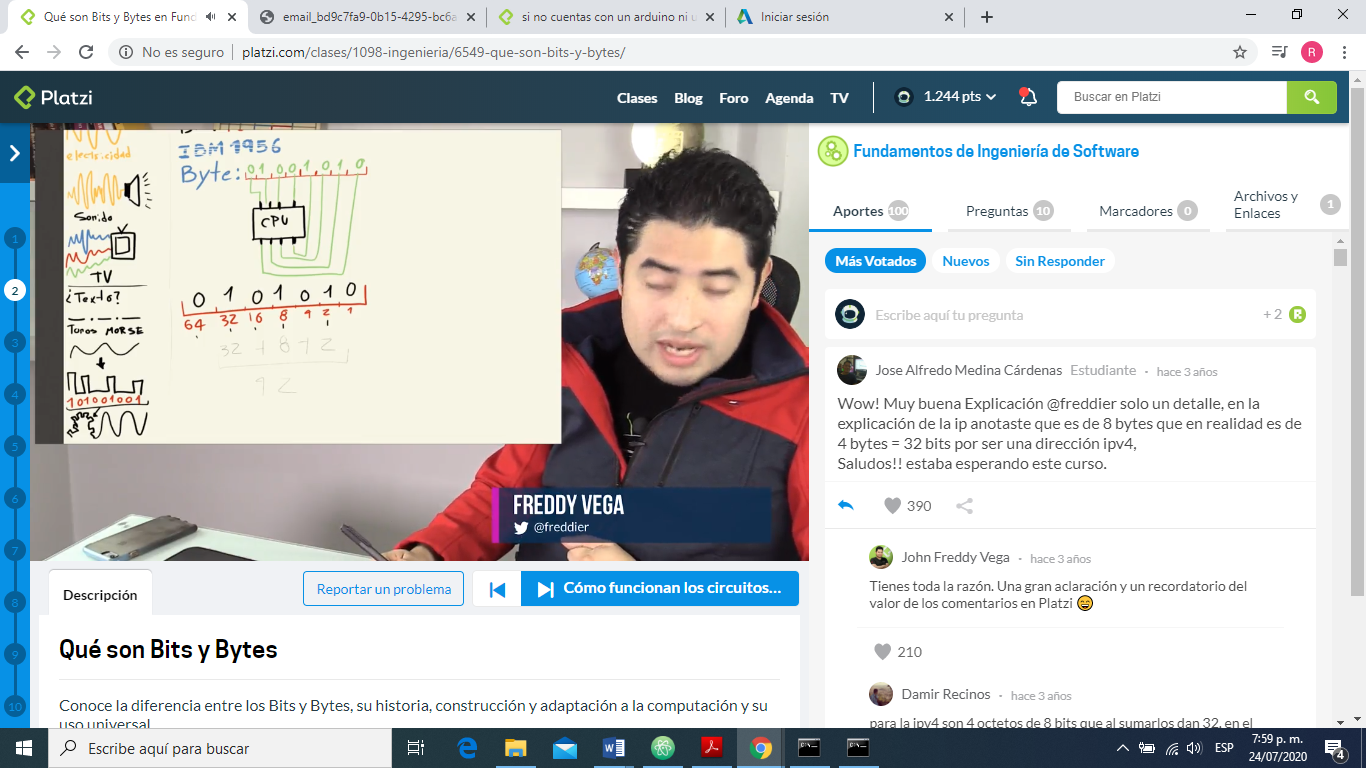
Byte= es una unidad de memoria para guardar un solo elemento. Como letras.

En los byte que contiene 8 espacios guardamos 1 y 0.

Cada espacio del byte tiene un valor que son



El valor es asignado por el numero 1. En el caso de arriba el numero asignado seria 42

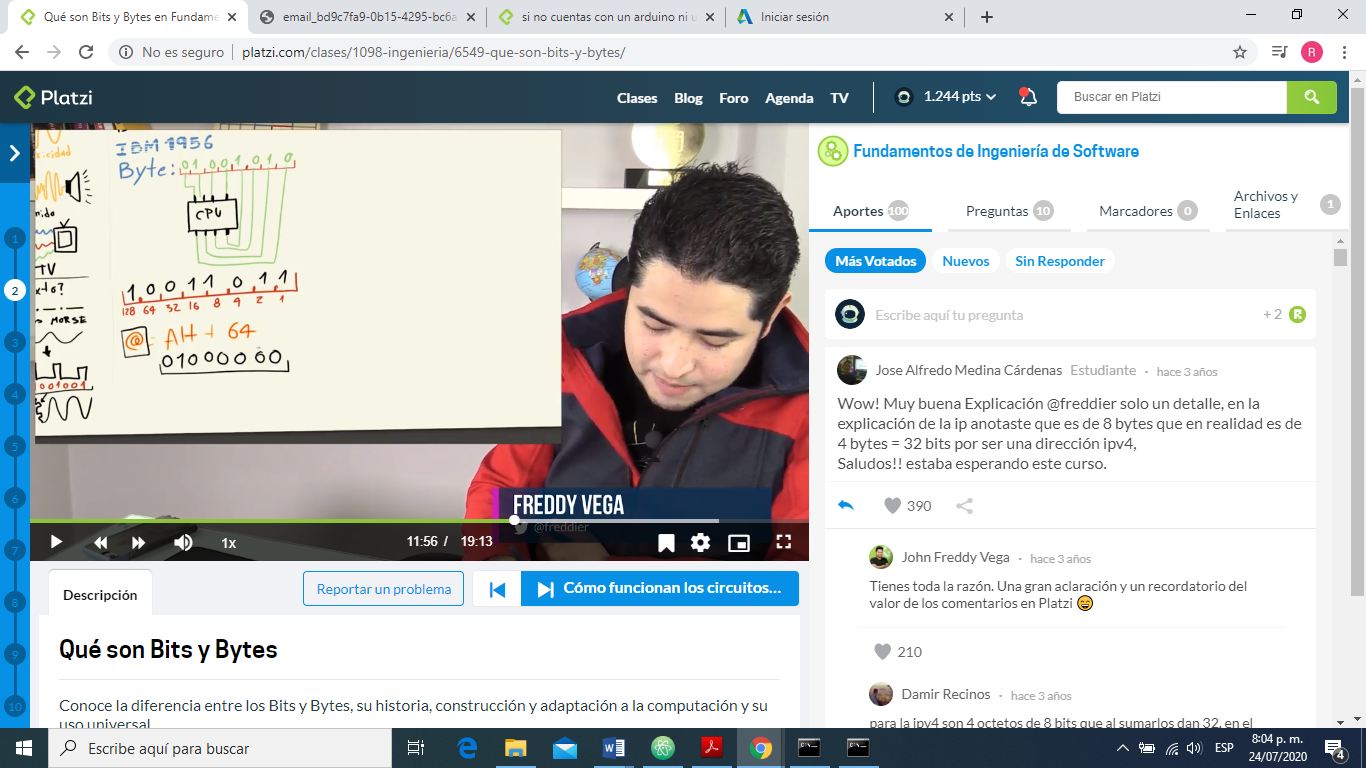


42 seria el resultado total del byte.

Notemos que cada posición se va multiplicando en realidad lo duplica. Es una base numérica. Hay diferentes tipos de base como la decimal o hexadecimal.

Para enviar texto lo que hicieron es asignar un byte a cada letra

Por ejemplo el @ tiene una representación en byte es:



Existe la tabla ASCII es una tabla de byte donde cada byte corresponde a una letra. Entonces internamente cuando se manda un string de 8 bits eso es un byte. Y el oredenador caundo recibe una señal de 8 bits hace una pausa y continua con otros 8 bits

También existe ASSEMBLER que son instrucciones especiales del procesador.

**PROCESADORES Y ARQUITECTURAS.**

Dentro de los computadores encontramos las CPU que tienen características como GHz que es la velocidad a la que procesan una instrucion. Core es cuantas instrucciones en parealelo pueden hacer. Core también refiere a cantidad de cpu.

La CPU por si sola no hace nada necesita quien le envie la información y eso la hace la BIOS es la que entiende todo lo que esta conectado es como un pequeño sistema operativo y da la señal de arranque a laCPU

CPU unidad de control de procesamiento.

Disco duro. Es el lugar donde se guarda información y es donde esta lo mas importante para arrancar un pc el sistema operativo.

RAM (random acces memori) es memoria de ultra alta velocidad, solo funciona cuando hay flujo eléctrico.

System on a chip.

MEMORIA RAM Y DISCO DUROS.

Disco duro guarda archivos de manera persistente (sin estar energizados)

Lo hace de manera secuencial (por pasos)

Los sistemas de archivos son convenciones internas de los sistemas operativos para poder acceder a los archivos almacenados.

En Linux existe ext3 o ext4

En Windows existía FAT16 o FAT32 (File Allocation Table), que fue reemplazado por NTFS (New Technology File System)

En Mac OSX el sistema de archivos se llamaba HFS (Hierarchical File System) pero ahora se llama AFS (Apple File System) en macOS Sierra.

Sistema operativo como anillos:

Primer anillo - Kernel: El Kernel lo podemos entender como la capa mas profunda de nuestro S.O. por lo tanto tiene acceso completo a archivos, drivers, programas, etc…Igual que cualquier otro proceso, se carga en la RAM como la cualidad de que es lo primero en cargar.

En esta capa también viven programas capaces de encriptar y desencriptar información, de tal forma que ninguna otra capa del S.O. tenga acceso a ellos.

Segundo anillo - Drivers: Como se ha dicho antes los drivers son código que se encargan de interpretar las señales de el hardware y establecer una comunicación con el software del PC. Estos primeros drivers pertenecen a piezas de hardware bastante importantes como la pantalla, el teclado, el mouse, etc…

Es importante indicar que entre el primer y el segundo anillo hay un indice de permisos donde estan almacenados qué permisos tiene cada app

Tercer anillo - Mas Drivers: Otra capa de drivers carga en un tercer “puesto” en la RAM. Dado que están más alejados del Kernel, tienen menos permisos y privilegios que los drivers del segundo anillo. Dado que mediante los drivers de este anillo, se comunican en su mayoría las Apps, es necesario que primero los drivers del tercer anillo pidan permisos a los del segundo anillo para luego así comunicarse con el hardware.

Cuarto anillo - Apps: Finalmente en la última capa del modelo de anillos del S.O. nos encontramos con las apps, que se cargan en la RAM para ejecutar procesos. Sin embargo a diferencia de los otros anillos no tienen ningún tipo de acceso directo al hardware del PC. Es importante tener en cuenta que así debería ser, pues de lo contrario cualquier Software escrito por terceros tendría la capacidad de acceder casi por completo al PC y a sus piezas de Hardware.