



CAMBIO BOOTLOGO

Tutorial cambio de Bootlogo by Grimor

Agradecimientos:

Antes de empezar quiero agradecer el trabajo de las personas que lo han hecho posible:

@Undef que ha dedicado muchas noches a la investigación de nuestra pantalla.

@Galagor (Félix) que ha desarrollado un cable usb fijo, que es mucho mejor que cualquier pinza 3D.

@Sebastian F. sin cuyos conocimientos del algoritmo y de la estructura de memoria MTK seguiríamos perdidos.

¿Que necesitamos?

Una Qicycle, un pc y un cable USB para la pantalla.

¿Vale cualquier pinza? sí, pero aquí vuelvo a recomendar el cable USB que Felix desarrolló para Qihack, por la seguridad de conexión que aporta.

Archivos Necesarios:

<https://mega.nz/folder/gZsXGBpS#l7zSgU0EjpYxrFHue-Pgvw>



QiHack Clamp

Pinza específica especialmente diseñada para los modelos de Qicycle sin bluetooth, de manera que puedas tener una conexión permanente entre el móvil y la bicicleta incluso circulando.

de igual modo, podrás conectarte al PC para poder realizar las tareas habituales que se hacen con la pinza tradicional



podrías traducir, desbloquear BT, flashear Custom Rom's (QiHack), cambio de logo, cambio de número de serie, restauración de kms totales recorridos, etc

Puede adquirirse aquí: <https://3dparts459136784.wordpress.com/>

Software:

Utilizaremos un editor de Gifs para adaptar la imagen a nuestra resolución y tamaño. Recomendando la herramienta online gratuita **EZ GIF OPTIMIZER** <https://ezgif.com/optimize>

Además vamos a usar un editor HEX, en nuestro caso **Hex Editor Neo**.

Necesitaremos la rom "**Grimor Logo**" como base pues ya lleva redireccionada la imagen al mayor espacio de memoria libre que dispone la rom.

Es una versión modificada de la 1.4 Spanish, por lo que estará en español.

Si alguien la quiere en otro idioma solo tiene que pedirla, es solo cuestión de cambiar unos bytes.

Recordemos que la imagen de arranque (bootlogo) original ocupa 1.4kb y ahora podremos poner imágenes de hasta 14kb (u opcionalmente 27kb , pero con un bug del salto de imagen arriba).

La imagen:

Usaremos una imagen gif de dimensiones 160x128 (ancho x alto) y un tamaño máximo de 14800 bytes (14kb).

¿Por qué gif? Por su reducido tamaño, posibilidad de transparencias y lo mejor de todo, permite animaciones!

Podemos descargar cualquier imagen o hacerla nosotros mismos con Photoshop o similar, pero siempre habrá que adaptarla a esas reglas de tamaño y peso.

Para ello la pasaremos por los diferentes filtros de <https://ezgif.com/optimize> hasta que cumpla esos requisitos.

Online GIF optimizer



File size: **12.33KIB**, width: 160px, height: 128px, frames: 8, type: gif [convert](#)

Optimization method (see description at bottom of the page):

Lossy GIF

Compression level:

35 (30 is very light compression, 200 is heavy)

Optimize GIF

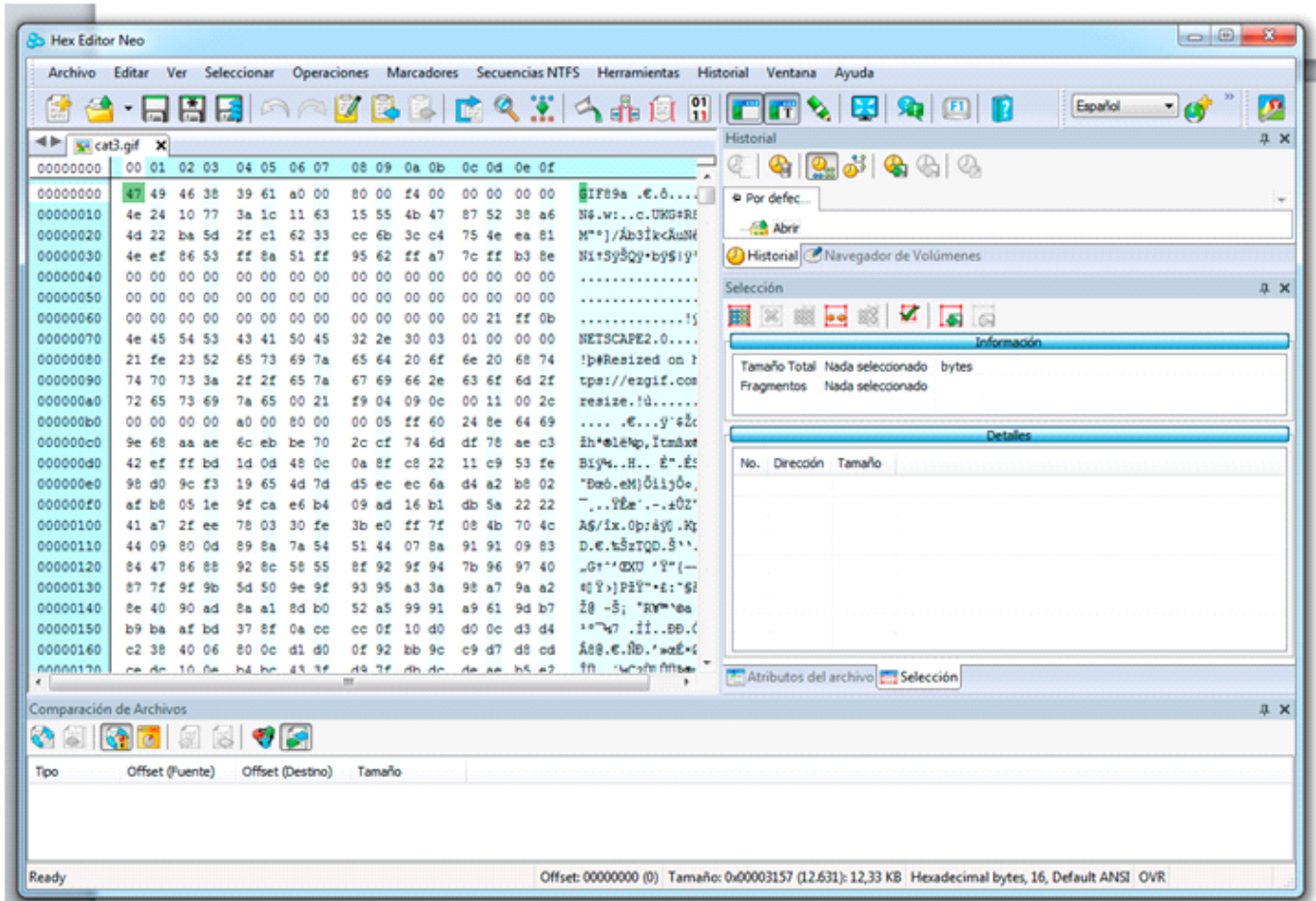
(loosy gif y color reduction consiguen achicar mucho la imagen, luego podéis probar a cortar "crop" a una relación de aspecto cercana a 4:3 y por ultimo escalar "resize" custom a 160x128.)

Si el resultado está por debajo de 14.8kb ya la tendríamos.

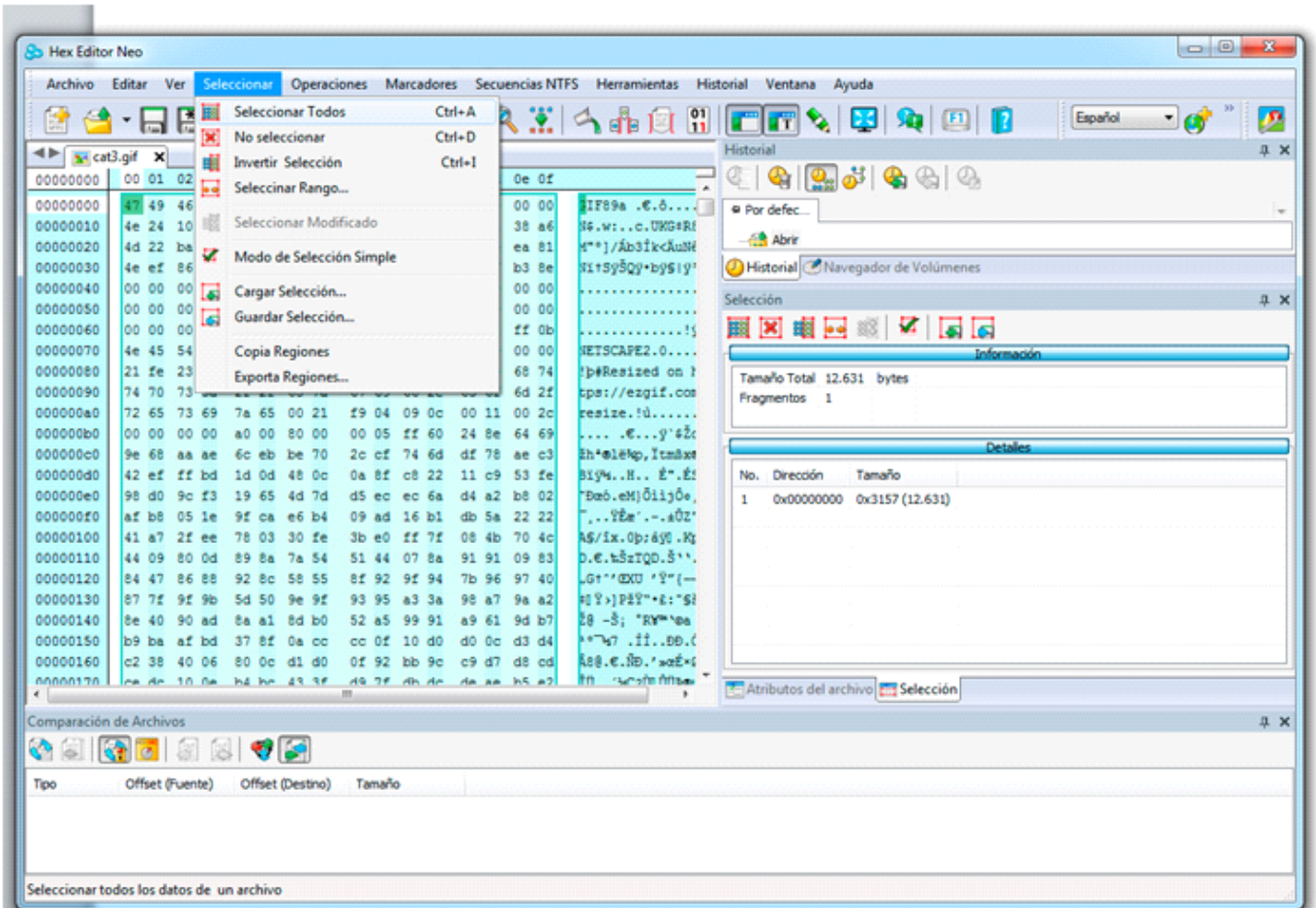
Insertarla en la ROM:

Abrimos **Hex Editor Neo** y cargamos nuestro gif.

Archivo/abrir o simplemente arrastrar la imagen y echarla al centro del editor. Obtendremos algo así:



Ahora pinchamos en Seleccionar/todo:



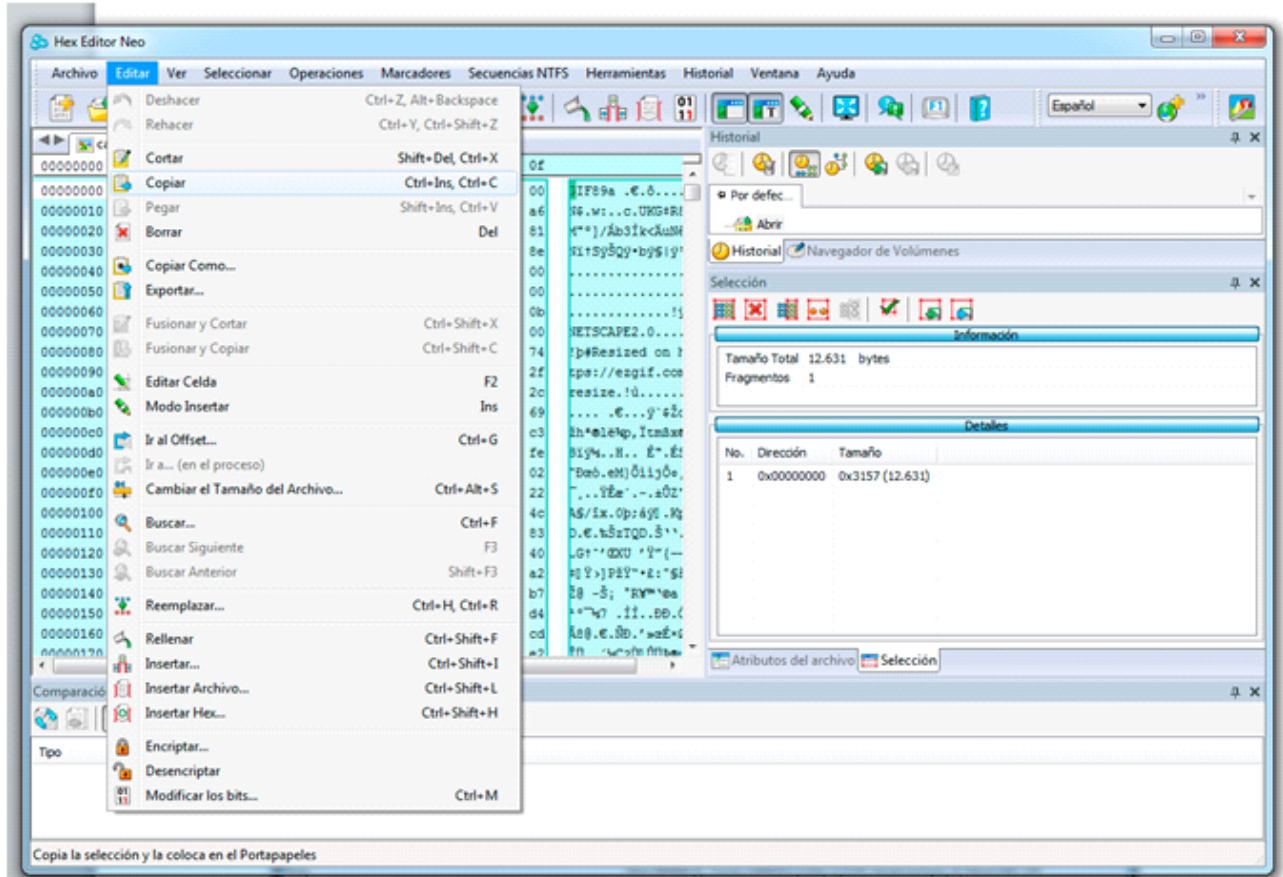
Ahora podemos ver en la pestaña información el peso total de la imagen (12631 bytes en sistema decimal y su equivalente 3157 en sistema hexadecimal).

Anotaremos el valor que nos de nuestra imagen en Hexadecimal, pero la usaremos invertida por pares (Little endian).

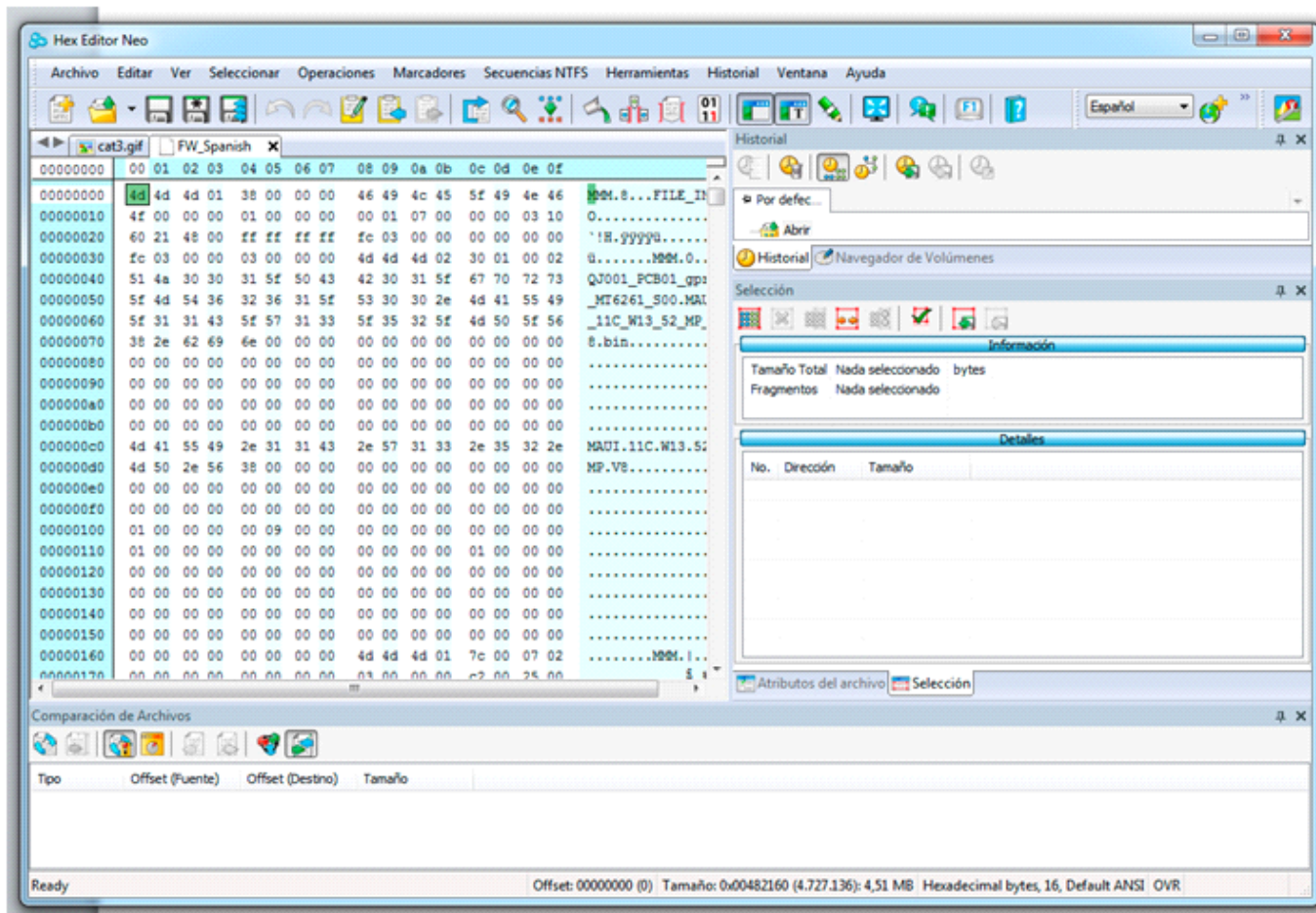
Si el tamaño es ABCD, anotaremos y la usaremos así: CDAB

En nuestro ejemplo 3157 lo vamos a insertar en la rom como 5731.

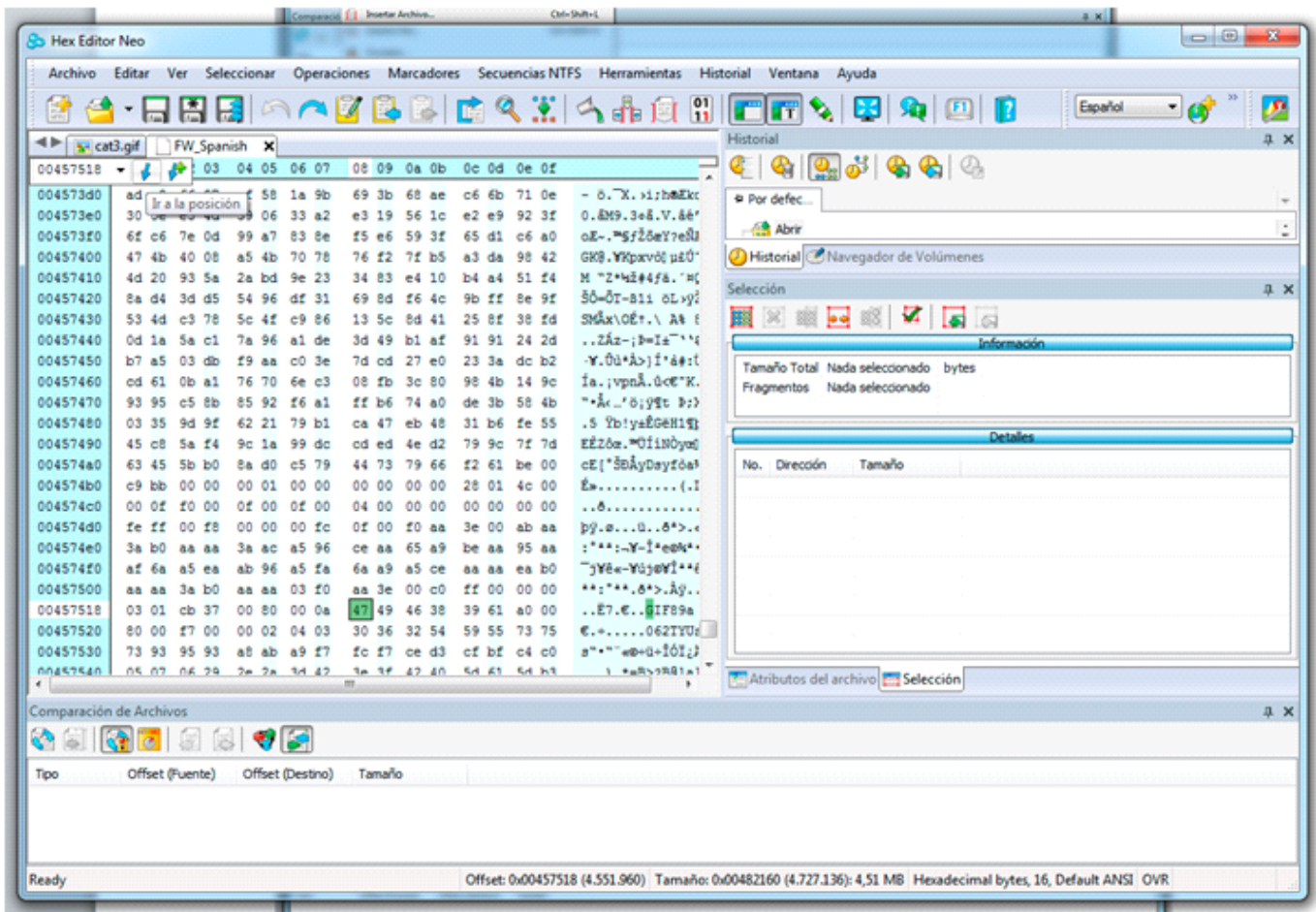
Por ultimo pincharemos en edición/copiar.



Ahora abrimos en Hex editor Neo el archivo de la rom de 4 megas (grimorlogo/ FW_Spanish) de igual manera que antes hicimos con el gif (archivo/abrir o arrastrando sobre la otra). No es necesario cerrar la anterior pues se abre como pestaña nueva en el editor:

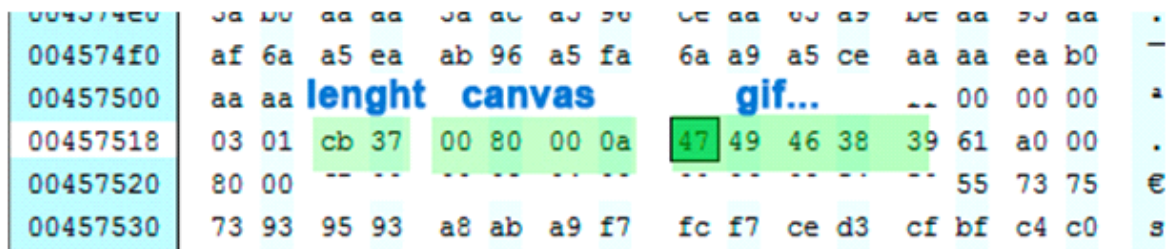


Ahora en la parte superior izquierda esta el buscador de Offset (dirección) donde escribiremos 457518 y daremos a la flechita azul para que nos lleve a ese punto:



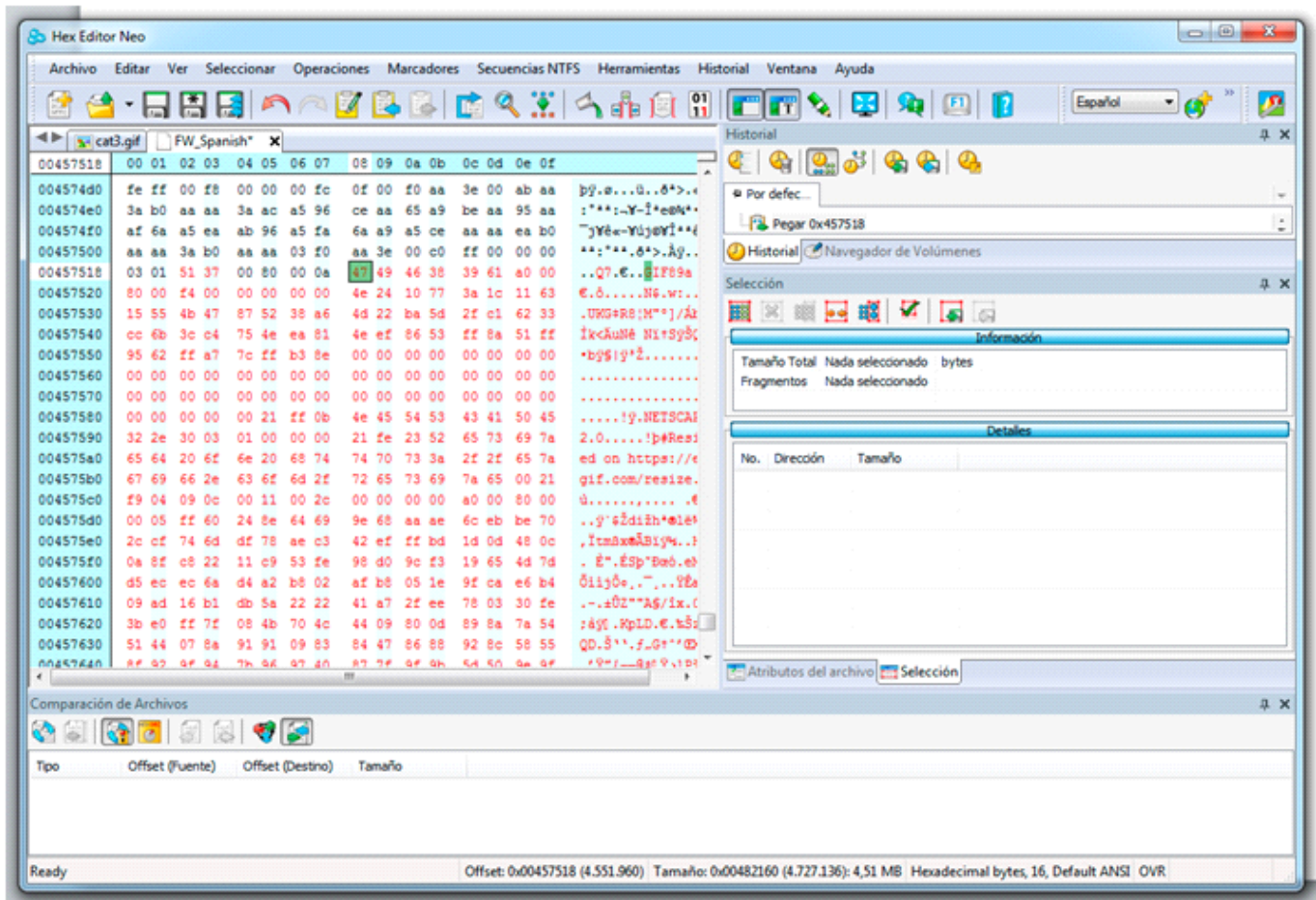
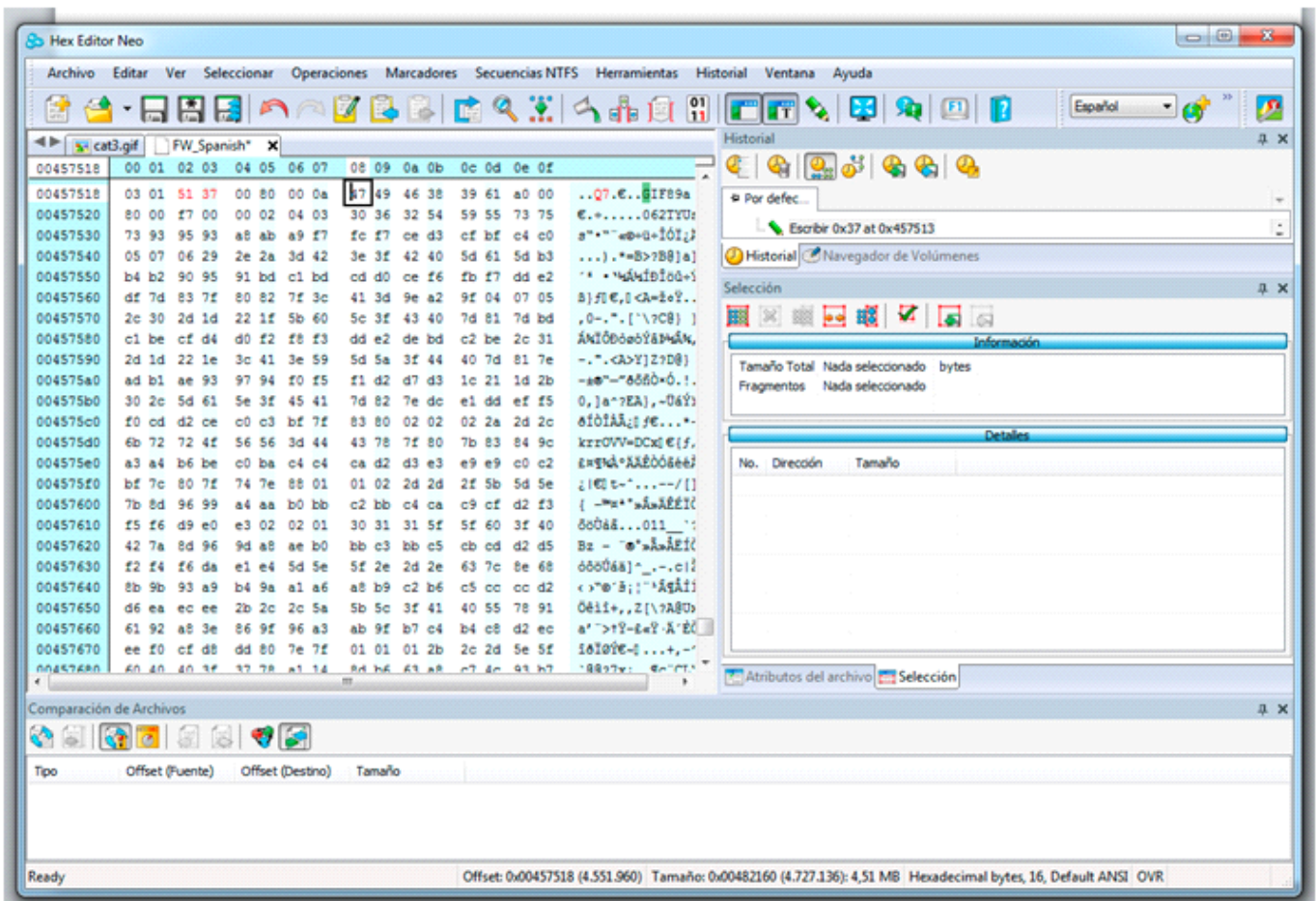
En ese punto pegaremos la nueva imagen que hemos copiado antes.

Pero antes, veamos la estructura que precede al gif:



Lenght es el peso de nuestra imagen, canvas la dimension del lienzo, y gif el inicio o cabecera de la imagen.

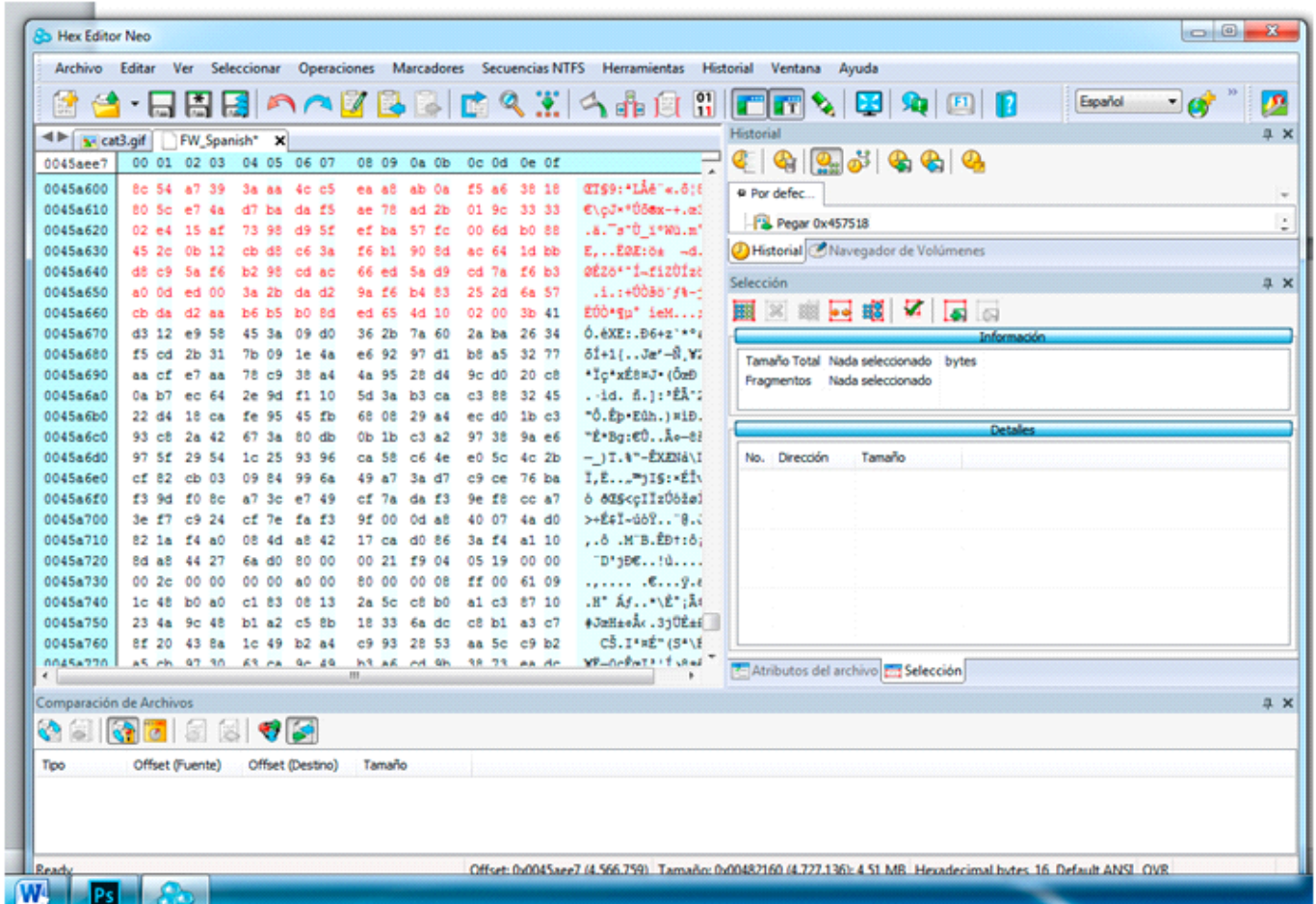
Así en el lugar del Lenght pondremos nuestra anotación hexadecimal invertida (en nuestro ejemplo se cambiaría ese cb37 por 5731), el canvas no habría que cambiarlo (si vuestra imagen realmente es 160x128) y con el cursor delante del 47, pegaremos nuestro gif nuevo que tenemos en el portapapeles:



En rojo se marcan las partes modificadas del código.

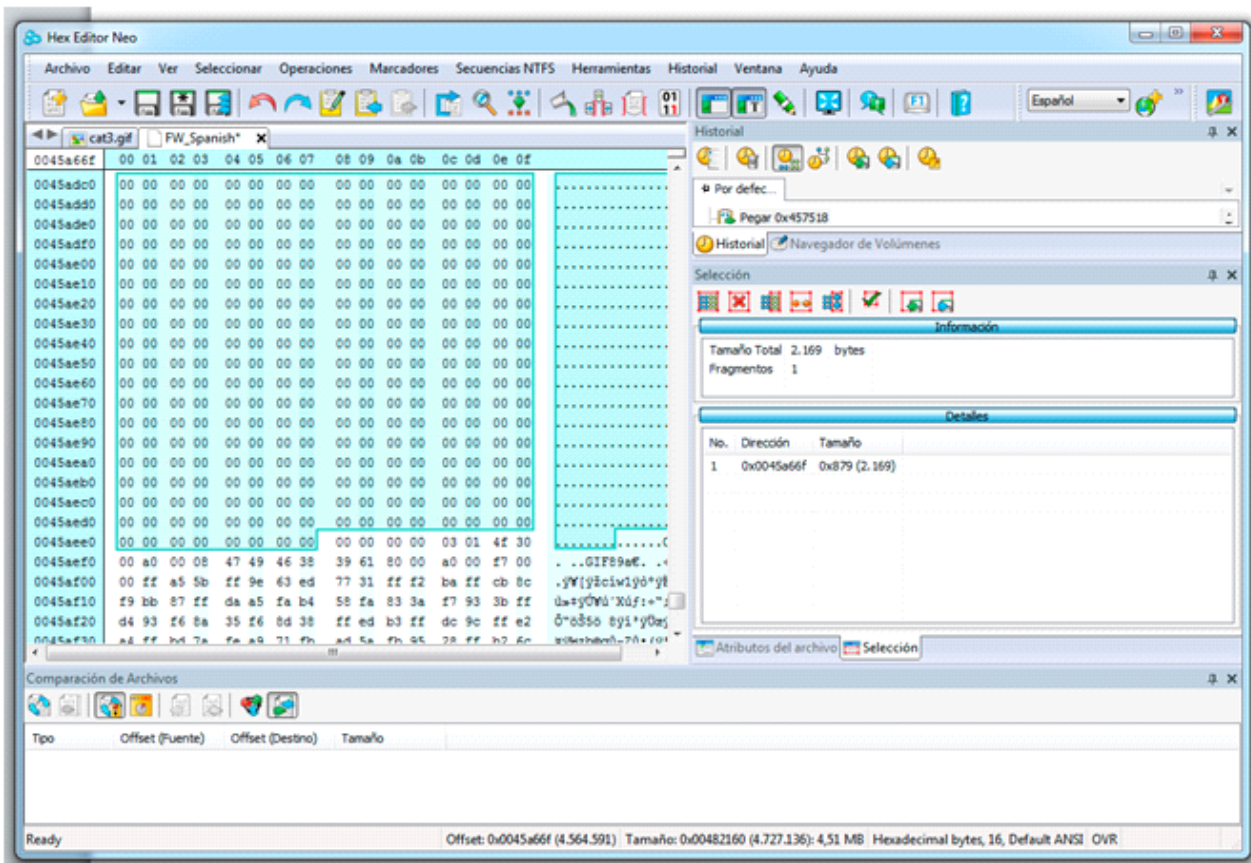
Por último, iríamos al final de lo rojo (final de la imagen nueva colocada) y completaremos con 00 hasta el offset 45aee7.

Para rellenar los ceros el espacio restante, podemos bien escribirlos con el teclado dejándolo pulsado, pero sin pasarnos, o mas fácil usando la utilidad de relleno que tiene el editor:

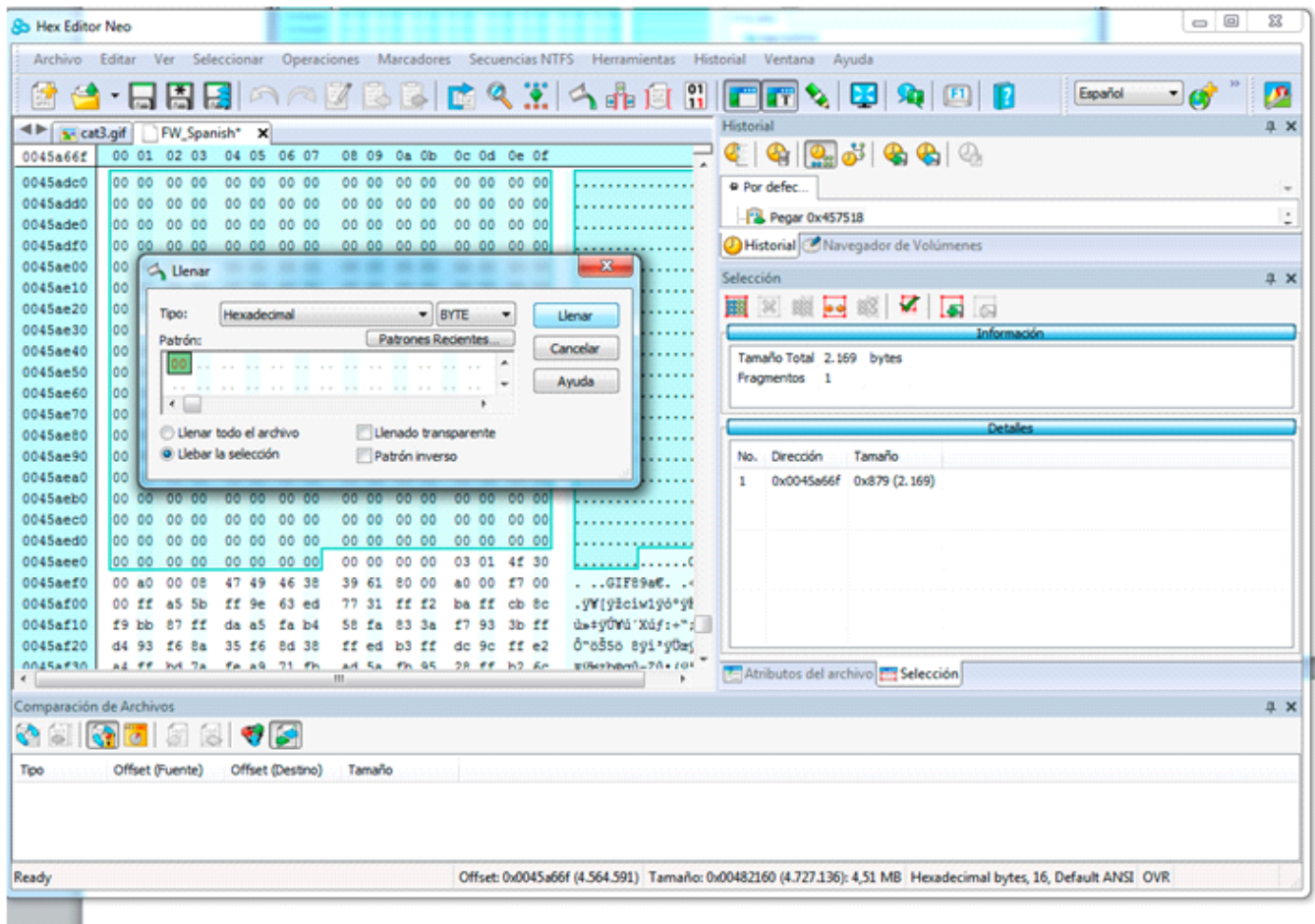


Todo lo negro debe ser 00 00 00 hasta el offset 45aee7

Seleccionamos todo el bloque a poner a 0:



Elegimos la herramienta del cubo de rellenado y escribimos el byte 00 así:



Con esto ya tendríamos la rom con nuestro bootlogo personalizado.

Terminaríamos con Archivo/ Guardar o Guardar como en función de si queremos o no sobre escribir la rom anterior.

Para flashear la rom modificada lo haríamos como una traducción, es decir con flashtool a partir de los archivos de la carpeta "Grimor Logo".

Notas:

- Opcionalmente podemos poner una imagen mayor, hasta el límite de 27179 bytes (27K); de igual manera, se completaría con "00" hasta la posición 45df42. El único inconveniente de esto último es que se rompen las tablas de memoria originales y aunque la rom funciona perfectamente, al cargar la imagen tiene un pequeño bug que le origina un salto arriba y a la derecha.
- Durante la secuencia de carga de la imagen normal, en el arranque, el primer fotograma del gif aparece congelado hasta q se carga toda la imagen entonces el resto se reproduce sin paradas. En el apagado es al contrario, el principio de la animación se visualiza al instante y se congela el último fotograma hasta que se apaga la pantalla. Esto puede ser tenido en cuenta a la hora de diseñar tus logos, para programar la duración de los fotogramas del gif.

VIDEO DE COMO HACERLO

El proceso del vídeo lleva la redirección del logo en la memoria que con la rom Grimor Logo no hace falta hacerla. Así que se empezaría en el 2'45" del vídeo tutorial.



Si le sirvió de utilidad este tutorial considere una pequeña donación al autor:
<https://www.paypal.me/grimorjander> Gracias!!

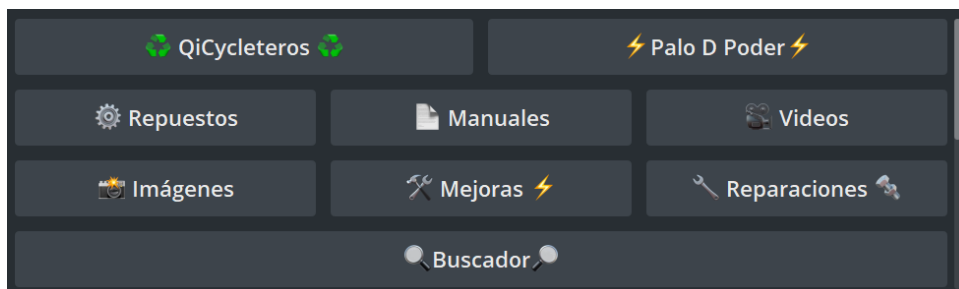


TELEGRAM

<https://t.me/Qicycleteros>

BOT QICYCLETEROS

@QiCycleteros_bot



En telegram tienes este Bot de ayuda.