

SERVICE MANUAL

**Asset-Tools-FT81x
v.3**

Asset-Tools-FT81X-v3

Herramienta para crear Asset con mas de 256KBytes de RAM Con las fuentes de este repositorio alcanza hasta los 2MBytes

Instrucciones de instalación

Leeremos la documentación desde este enlace

<http://excamera.com/sphinx/gameduino2/code.html> Crearemos una carpeta por ejemplo llamada gd3asset

Descargaremos las fuentes desde <https://pypi.python.org/pypi/gameduino2> o bien las de aquí ya preparadas y listas para compilar. las descomprimos (pulsando botón derecho, y extraer), lo copiaremos todo a la carpeta que hemos creado anteriormente

una vez descomprimido nos vamos a la carpeta /home/mi_usuario/gd3asset/gameduino2-0.1.8/gameduino2/ Editaremos el fichero llamado **prey.py** con gedit y buscaremos dos líneas exactamente la que pone;

if len(self.alldata) > 0x40000: y la cambiaremos por if len(self.alldata) > 0x200000: y lo guardaremos

Ahora toca instalar dependencias de python y compilar el binario.

sudo apt-get install python-setuptools python-imaging

nos situamos en la carpeta /home/mi_usuario/gd3asset/gameduino2-0.1.8/ Con el boton derecho del ratón elegimos “Abrir terminal aquí”

sudo python setup.py install

Lo normal es que salga esto;

```
running install
running build
running build_py
running build_scripts
running install_lib creating /usr/local/lib/python2.7/dist-packages/gameduino2
copying build/lib.linux-i686-2.7/gameduino2/registers.py -> /usr/local/lib/python2.7/dist-packages/gameduino2
copying build/lib.linux-i686-2.7/gameduino2/init.py -> /usr/local/lib/python2.7/dist-packages/gameduino2
copying build/lib.linux-i686-2.7/gameduino2/prep.py -> /usr/local/lib/python2.7/dist-packages/gameduino2
copying build/lib.linux-i686-2.7/gameduino2/psdparser.py -> /usr/local/lib/python2.7/dist-packages/gameduino2
copying build/lib.linux-i686-2.7/gameduino2/base.py -> /usr/local/lib/python2.7/dist-packages/gameduino2
copying build/lib.linux-i686-2.7/gameduino2/convert.py -> /usr/local/lib/python2.7/dist-packages/gameduino2
copying build/lib.linux-i686-2.7/gameduino2/remote.py -> /usr/local/lib/python2.7/dist-packages/gameduino2
copying build/lib.linux-i686-2.7/gameduino2/imbytes.py -> /usr/local/lib/python2.7/dist-packages/gameduino2
byte-compiling /usr/local/lib/python2.7/dist-packages/gameduino2/registers.py to registers.pyc
byte-compiling /usr/local/lib/python2.7/dist-packages/gameduino2/init.py to init.pyc
byte-compiling /usr/local/lib/python2.7/dist-packages/gameduino2/prep.py to prep.pyc
byte-compiling /usr/local/lib/python2.7/dist-packages/gameduino2/psdparser.py to psdparser.pyc
byte-compiling /usr/local/lib/python2.7/dist-packages/gameduino2/base.py to base.pyc
byte-compiling /usr/local/lib/python2.7/dist-packages/gameduino2/convert.py to convert.pyc
byte-compiling /usr/local/lib/python2.7/dist-packages/gameduino2/remote.py to remote.pyc
byte-compiling /usr/local/lib/python2.7/dist-packages/gameduino2/imbytes.py to imbytes.pyc
running install_scripts copying build/scripts-2.7/gd2asset -> /usr/local/bin
changing mode of /usr/local/bin/gd2asset to 775 running install_egg_info Writing /usr/local/lib/python2.7/dist-packages/gameduino2-0.1.8.egg-info
```

De ser correcto ya tenemos la herramienta preparada.