



OpenCore

Reference Manual (0.6.~~5~~.6)

[2021.01.06]

11.3 Tools and Applications

Standalone tools may help to debug firmware and hardware. Some of the known tools are listed below. While some tools can be launched from within OpenCore, see more details in the Tools subsection of the configuration, most should be run separately either directly or from **Shell**.

To boot into OpenShell or any other tool directly save **OpenShell.efi** under the name of **EFI\BOOT\BOOTX64.EFI** on a FAT32 partition. In general it is unimportant whether the partition scheme is GPT or MBR.

While the previous approach works both on Macs and other computers, an alternative Mac-only approach to bless the tool on an HFS+ or APFS volume:

```
sudo bless --verbose --file /Volumes/VOLNAME/DIR/OpenShell.efi \
--folder /Volumes/VOLNAME/DIR/ --setBoot
```

Listing 3: Blessing tool

Note 1: **/System/Library/CoreServices/BridgeVersion.bin** should be copied to **/Volumes/VOLNAME/DIR**.

Note 2: To be able to use **bless** disabling System Integrity Protection is necessary.

Note 3: To be able to boot Secure Boot might be disabled if present.

Some of the known tools are listed below (builtin tools are marked with *):

BootKicker*	Enter Apple BootPicker menu (exclusive for Macs with compatible GPUs).
ChipTune*	Test BeepGen protocol and generate audio signals of different style and length.
CleanNvram*	Reset NVRAM alternative bundled as a standalone tool.
GopStop*	Test GraphicsOutput protocol with a simple scenario.
HdaCodecDump*	Parse and dump High Definition Audio codec information (requires AudioDxe).
KeyTester*	Test keyboard input in SimpleText mode.
MemTest86	Memory testing utility.
OpenControl*	Unlock and lock back NVRAM protection for other tools to be able to get full NVRAM access when launching from OpenCore.
OpenShell*	OpenCore-configured UEFI Shell for compatibility with a broad range of firmware.
PavpProvision	Perform EPID provisioning (requires certificate data configuration).
ResetSystem*	Utility to perform system reset. Takes reset type as an argument: ColdReset , Firmware , Shutdown , WarmReset . Defaults to ColdReset .
RtcRw*	Utility to read and write RTC (CMOS) memory.
VerifyMsrE2*	Check CFG Lock (MSR 0xE2 write protection) consistency across all cores.

11.4 OpenCanopy

OpenCanopy is a graphical OpenCore user interface that runs in **External PickerMode** and relies on **OpenCorePkg OcBootManagementLib** similar to the builtin text interface.

OpenCanopy requires graphical resources located in **Resources** directory to run. Sample resources (fonts and images) can be found in **OcBinaryData** repository. Customised icons can be found over the internet (e.g. [here](#) or [there](#)).

OpenCanopy provides full support for **PickerAttributes** and offers a configurable builtin icon set. The default chosen icon set depends on the **DefaultBackgroundColor** variable value. For **Light Gray Old** icon set will be used, for other colours — the one without a prefix.

Predefined icons are put to **\EFI\OC\Resources\Image** directory. Full list of supported icons (in **.icns** format) is provided below. Missing optional icons will use the closest available icon. External entries will use **Ext**-prefixed icon if available (e.g. **OldExtHardDrive.icns**).

Note: In the following all dimensions are normative for the 1x scaling level and shall be scaled accordingly for other levels.

- **Cursor** — Mouse cursor (mandatory, [up to 144x144](#)).
- **Selected** — Selected item (mandatory, [144x144](#)).
- **Selector** — Selecting item (mandatory, [up to 144x40](#)).
- **Left** — Scrolling left (mandatory, [40x40](#)).
- **Right** — Scrolling right (mandatory, [40x40](#)).

- **HardDrive** — Generic OS (mandatory, [128x128](#)).
- [Background](#) — Centred background image.
- **Apple** — Apple OS ([128x128](#)).
- **AppleRecv** — Apple Recovery OS ([128x128](#)).
- **AppleTM** — Apple Time Machine ([128x128](#)).
- **Windows** — Windows ([128x128](#)).
- **Other** — Custom entry (see [Entries](#), [128x128](#)).
- **ResetNVRAM** — Reset NVRAM system action or tool ([128x128](#)).
- **Shell** — Entry with UEFI Shell name (for e.g. [OpenShell](#) ([128x128](#))).
- **Tool** — Any other tool ([128x128](#)).

Predefined labels are put to `\EFI\OC\Resources\Label` directory. Each label has `.1b1` or `.12x` suffix to represent the scaling level. Full list of labels is provided below. All labels are mandatory.

- **EFIBoot** — Generic OS.
- **Apple** — Apple OS.
- **AppleRecv** — Apple Recovery OS.
- **AppleTM** — Apple Time Machine.
- **Windows** — Windows.
- **Other** — Custom entry (see [Entries](#)).
- **ResetNVRAM** — Reset NVRAM system action or tool.
- **Shell** — Entry with UEFI Shell name (e.g. [OpenShell](#)).
- **Tool** — Any other tool.

Note: All labels must have a height of exactly 12 px. There is no limit for their width.

Label and icon generation can be performed with bundled utilities: `disklabel` and `icnspack`. ~~Please refer to sample data for the details about the dimensions.~~ Font is Helvetica 12 pt times scale factor.

Font format corresponds to AngelCode binary BMF. While there are many utilities to generate font files, currently it is recommended to use `dpFontBaker` to generate bitmap font (using `CoreText` produces best results) and `fonverter` to export it to binary format.

11.5 OpenRuntime

`OpenRuntime` is an OpenCore plugin implementing `OC_FIRMWARE_RUNTIME` protocol. This protocol implements multiple features required for OpenCore that are otherwise not possible to implement in OpenCore itself as they are needed to work in runtime, i.e. during operating system functioning. Feature highlights:

- NVRAM namespaces, allowing to isolate operating systems from accessing select variables (e.g. `RequestBootVarRouting` or `ProtectSecureBoot`).
- Read-only and write-only NVRAM variables, enhancing the security of OpenCore, Lilu, and Lilu plugins, such as `VirtualSMC`, which implements `AuthRestart` support.
- NVRAM isolation, allowing to protect all variables from being written from an untrusted operating system (e.g. `DisableVariableWrite`).
- UEFI Runtime Services memory protection management to workaround read-only mapping (e.g. `EnableWriteUnprotector`).

11.6 Properties

1. APFS

Type: plist dict

Failsafe: None

Description: Provide APFS support as configured in APFS Properties section below.

2. Audio

Type: plist dict

Failsafe: None

Description: Configure audio backend support described in Audio Properties section below.