vekatech

VK-PyFlasher Rescue tool

Content:

Introduction

This tool is designed to program various VK-RZ **S**ingle **B**oard **C**omputers at the factory. It can also serve as a **rescue** tool, by performing a factory reset, if an issue arises. Currently it supports the following boards:

- VK-RZ/x2L
- VK-RZ/G2LC
- VK-CMG2LC (with VK-CM-RZ/G2LC **SoM**)

Use Case

PyFlasher requires the presence of specific device files before it can operate. Depending on the SBC you need to rescue, download the relevant files (uboot & image) from the website.

Once you have the files you can use:

- `flash_boot.py` to write **U-boot** loader in the MMC or in the SPI Flash
- `flash_img.py` to write **Debian** OS or **Yocto** in the MMC

flash_boot.py

The operation of the script is quite straightforward. Use following arguments to specify what will be flashed and where it will be written:

- `--board=<device>` where **device** refers to one of the supported boards on which the flashing process will occur: **`vkrzv2l`** or **`vkrzg2l`** or **`vkrzg2lc`** or **`vkcmg2lc`**.

- `--serial_port=<port>` where **port** refers to the serial cable through which the flashing process will occur, such as: **`/dev/ttyUSB<n>`** or **`/dev/ttyACM<n>`** or **`/dev/ttySC<n>`**, and **n** specifies the port number.

- `--qspi` by default U-boot will be flashed to the MMC, but you can use this argument to change its destination to the SPI Flash. In production, the script is run twice to flash U-Boot to both: the MMC and SPI flash.

- `--debug` enables more verbose output.

Here is an example of flashing U-Boot to the VK-RZ/G2LC in production:

```
python3 flash_boot.py --board=vkrzg2lc --serial_port=/dev/ttyUSB0
python3 flash_boot.py --board=vkrzg2lc --serial_port=/dev/ttyUSB0 --qspi
```

Executing these commands will effectively perform a factory reset, restoring the VK-RZ/G2LC to its original production state, regardless of its current condition.

flash_img.py

The operation of the script is quite straightforward. Use following arguments to specify what will be flashed:

- `--board=<device>` where **device** refers to one of the supported boards on which the flashing process will occur: **`vkrzv2l`** or **`vkrzg2l`** or **`vkrzg2lc`** or **`vkcmg2lc`**.

- `--serial_port=<port>` where **port** refers to the serial cable through which the flashing process will occur, such as: **`/dev/ttyUSB<n>`** or **`/dev/ttyACM<n>`** or **`/dev/ttySC<n>`**, and **n** specifies the port number.

- `--image_rootfs=<imgfile>` where **imgfile** refers to the image that will be written to the MMC. i.e. **Debian** **`debian-bookworm-<device>.img`** or **Yocto** **`core-image-<bsp/weston/qt>-<device>.wic`**. In production, the script is used to flash Debian.

- `--image_path=<path>` by default, the script searches for the **imgfile** in the **`images/<device>`** directory. However, you can use this argument to specify a different **path**.

- `--udp` by default, the script loads the **imgfile** via USB. However, you can use this argument to switch the transfer medium to Ethernet if USB is unavailable on the machine running the script.

- `--static_ip=<ip>` by default if Ethernet medium is specified, the script attempts to assign a dynamic IP to the device. However, you can use this option to assign a static IP if a DHCP network infrastructure is unavailable.

- `--debug` enables more verbose output.

Here is an example of flashing Debian to the VK-RZ/G2LC in production:

```
python3 flash_img.py --board=vkrzg2lc --serial_port=/dev/ttyUSB0 --image_rootfs=debian-bookworm-vkrzg2lc.img
```

Executing this command will effectively perform a factory reset, restoring the VK-RZ/G2LC to its original production state, regardless of its current condition.

If you need to flash a custom image, rather than the factory default Debian or Yocto, note that U-Boot expects the image to be in a sparse format. To create a sparse image, install the package **`android-sdk-libsparse-utils`** and use the **`img2simg`** tool.

```
sudo apt-get install android-sdk-libsparse-utils
```

```
img2simg <path_to_custom_img>/<custom_img> …images/<device>/<custom_img>
```

```
python3 flash_img.py --board=<device> --serial_port=/dev/ttyUSB<n> --image_rootfs=<custom_img>
```

Revision overview list

Revision number	Description changes
0.1	Initial

Vekatech Ltd

63 Nestor Abadzhiev st.
4023 Plovdiv
Bulgaria https://vekatech.com
+359 (0) 32 262362
info@vekatech.com