

Method A

- 1.) Install Debian package for Rockchip U-Boot

```
sudo apt install u-boot-rockchip
```

- 2.) Flashing U-Boot to our Pine64 Rock64 SBC

```
lsblk
```

remember storage device name

```
sudo u-boot-install-rockchip /dev/mmcblkX
```

Method B

- 1.) Install Compiler for building U-Boot on our Pine64 Rock64 SBC

```
sudo apt install device-tree-compiler build-essential libssl-dev python3-dev bison  
sudo apt install flex swig gcc-aarch64-linux-gnu gcc-arm-none-eabi gcc make git
```

- 2.) Build U-Boot on our Pine64 Rock64 SBC

```
cd /home/youruser/
```

```
git clone https://github.com/ARM-software/arm-trusted-firmware
```

```
cd arm-trusted-firmware
```

```
git tag
```

remember last stable (lts-v2.12.1)

```
git checkout lts-v2.12.1
```

```
make PLAT=rk3328 bl31
```

```
cd ..
```

```
git clone git://git.denx.de/u-boot.git
```

```
cd u-boot
```

```
git tag
```

remember last stable (v2025.01)

```
git checkout v2025.01
```

```
ln -s /home/youruser/arm-trusted-firmware/build/rk3328/release/bl31/bl31.elf bl31.elf
```

```
make BL31=bl31.elf rock64-rk3328_defconfig
```

```
make -j2 BL31=bl31.elf all u-boot.itb
```

```
cd ..
```

```
cp -r /home/youruser/u-boot/idbloader.img /home/youruser/
```

```
cp -r /home/youruser/u-boot/u-boot.itb /home/youruser/
```

- 3.) Flashing U-Boot to our Pine64 Rock64 SBC

```
cd /home/youruser/
```

```
lsblk
```

remember storage device name

```
sudo dd if=idbloader.img of=/dev/mmcblkX seek=64 conv=notrunc
```

```
sudo dd if=u-boot.itb of=/dev/mmcblkX seek=16384 conv=notrunc
```

- 4.) Reboot the machine

```
sudo reboot
```