

1 Filesystem update via uSD

1.1 Create uSD

Create uSD from ISO. Be carefull to change /dev/sdb with your system uSD device path, minimum 8GB size required.

```
gunzip -c ska-low-smm_v0.4.0_20230516.img.tgz | sudo dd of=/dev/sdb status=progress
```

1.2 Boot from uSD

Insert uSD into uSD slot of SMB board and power-on the board

1.3 Web server start

No action required. Filesystem start with web_server service active for external control (e.g. with SKALAB).

The filesystem configure CPU ip address calculated from CPLD ip address read from EEPROM, decreasing by 6. (e.g. CPLD ip 10.0.10.70 follow to CPU ip address 10.0.10.64).

Below ip addresses MUST be reserved for board function:

RESERVED IPs	
10.0.10.64	CPU
10.0.10.65	reserved
10.0.10.66	reserved
10.0.10.67	reserved
10.0.10.68	reserved
10.0.10.69	reserved
10.0.10.70	CPLD
10.0.10.71	SLOT-1 TPM
10.0.10.72	SLOT-2 TPM
10.0.10.73	SLOT-3 TPM
10.0.10.74	SLOT-4 TPM
10.0.10.75	SLOT-5 TPM
10.0.10.76	SLOT-6 TPM
10.0.10.77	SLOT-7 TPM
10.0.10.78	SLOT-8 TPM
10.0.10.79	reserved

1.4 Connect to board via SSH

```
sshpass -p SkaUser ssh -o StrictHostKeyChecking=no mnguser@10.0.10.64
```

1.5 Gateway

Configure external HOST as gateway (no DHCP needed) and NTP server [optional required for 1.6 and 1.9]

1.6 BIOS tool update

Check for ska-low-smm-bios update if needed (needs internet access configured at 1.5)

```
(venv) mnguser@ska-low-smm:~/SubrackMngAPI$ pip install git+https://gitlab.com/sanitaseg/ska-low-smm-bios.git
```

1.7 BIOS update into board

Update BIOS if needed (read below)

1.8 Network configuration

Change ip address if needed (read below)

1.9 SubrackMngAPI update

Check for SubrackMngAPI update if needed (needs internet access configured at 1.5)

```
(venv) mnguser@ska-low-smm:~/SubrackMngAPI$ git pull
Already up to date.
```

1.10 Reboot

Shutdown and reboot to apply changes.

```
sudo poweroff
```

2 BIOS update into board

ska_low_smm_bios can be used to update a SMM board, you needs to specify bios version. Ip address is not required because it operate on localhost only.

```
$ python -m ska_low_smm_bios --bios v1.0.0
=====
PLEASE READ THE AGREEMENT CAREFULLY.
BY USING THIS SOFTWARE, YOU ACCEPT THE TERMS OF THE AGREEMENT.
You can read license by '--show-license' option
=====
```

BOARD INFO	
:-----	:-----
CPLD_ip_address	10.0.10.86
CPLD_netmask	255.255.255.0
CPLD_gateway	10.0.10.1
CPLD_ip_address_eep	10.0.10.86
CPLD_netmask_eep	255.255.255.0
CPLD_gateway_eep	10.0.10.1
CPLD_MAC	04:91:62:b2:28:20
CPU_ip_address	10.0.10.80
CPU_netmask	255.255.255.0
CPU_MAC	04:91:62:b2:6c:b8
SN	
PN	SKA_SMB
bios	v??.?.?
bios_cpld	0xbe7a1014_0x202106150954
bios_mcu	0xdb000102_0x2021040600125020
bios_uboot	2018.03-00002-g692c8e6e-dirty
bios_krn	4.14.98-0002-00003-gffba12ad9
BOARD_MODE	SUBRACK
LOCATION	65535:255:255
HARDWARE_REV	v1.2.4

BIOS	ACTUAL	REQUESTED	diff	
:-----	:-----	:-----	:-----	
rev	v??.?.?	v1.0.0	*	
cpld	0xbe7a1014_0x202106150954	0xbe7a1014_0x202106150954		
mcu	0xdb000102_0x2021040600125020	0xdb000102_0x2021040600125020		
uboot	2018.03-00002-g692c8e6e-dirty	2018.03-00005-gda75be7d	*	
krn	4.14.98-0002-00003-gffba12ad9	4.14.98-0002-00003-gffba12ad9		

3 Change network configuration

ska_low_smm_bios can be also used to change network configuration stored into non-volatile memory. The OS of SMM, at boot time, retrieve information from non-volatile memory to generate /etc/network/interfaces. OS also assume, for convenience, that a ntp server is available and try to exec a update time at boot.

```
$ python -m ska_low_smm_bios --change-ip 10.0.10.64 --change-netmask 255.255.0.0 --
change-gateway 10.0.10.254
```

```
=====
PLEASE READ THE AGREEMENT CAREFULLY.
BY USING THIS SOFTWARE, YOU ACCEPT THE TERMS OF THE AGREEMENT.
You can read license by '--show-license' option
=====
```

BOARD INFO	
:-----	:-----

CPLD_ip_address	10.0.10.86	
CPLD_netmask	255.255.255.0	
CPLD_gateway	10.0.10.1	
CPLD_ip_address_eep	10.0.10.86	
CPLD_netmask_eep	255.255.255.0	
CPLD_gateway_eep	10.0.10.1	
CPLD_MAC	04:91:62:b2:28:20	
CPU_ip_address	10.0.10.80	
CPU_netmask	255.255.255.0	
CPU_MAC	04:91:62:b2:6c:b8	
SN		
PN	SKA_SMB	
bios	v?..?.	
bios_cpld	0xbe7a1014_0x202106150954	
bios_mcu	0xdb000102_0x2021040600125020	
bios_uboot	2018.03-00002-g692c8e6e-dirty	
bios_krn	4.14.98-0002-00003-gffba12ad9	
BOARD_MODE	SUBRACK	
LOCATION	65535:255:255	
HARDWARE_REV	v1.2.4	

===== WARNING !!! =====

Error in network configuration may leads to unreachable board.

Below ip addresses MUST be reserved for board function:

RESERVED IPs		
:-----	:-----	
10.0.10.64	CPU	
10.0.10.65	reserved	
10.0.10.66	reserved	
10.0.10.67	reserved	
10.0.10.68	reserved	
10.0.10.69	reserved	
10.0.10.70	CPLD	
10.0.10.71	SLOT-1 TPM	
10.0.10.72	SLOT-2 TPM	
10.0.10.73	SLOT-3 TPM	
10.0.10.74	SLOT-4 TPM	
10.0.10.75	SLOT-5 TPM	
10.0.10.76	SLOT-6 TPM	
10.0.10.77	SLOT-7 TPM	
10.0.10.78	SLOT-8 TPM	
10.0.10.79	reserved	

	ACTUAL	NEW	
:-----	:-----	:-----	
CPU ip address	10.0.10.80	10.0.10.64	
CPLD ip address	10.0.10.86	10.0.10.70	

```
| netmask          | 255.255.255.0 | 255.255.0.0 |
| gateway          | 10.0.10.1      | 10.0.10.254 |
Do you want continue (y/N)
```

Here you can found network configuration applied

/etc/network/interfaces

```
# interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:
# WARNING!!! This file will be overwritten at boot by rc.local
source-directory /etc/network/interfaces.d
```

```
auto eth0
allow-hotplug eth0
iface eth0 inet static
    address 10.0.10.80
    netmask 255.255.255.0
```

/etc/resolv.conf

```
nameserver 8.8.8.8
nameserver 8.8.4.4
```

route

Kernel IP routing table

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
0.0.0.0	10.0.10.1	0.0.0.0	UG	0	0	0	eth0
10.0.10.0	0.0.0.0	255.255.255.0	U	0	0	0	eth0