

Linux fw_printenv fw_setenv to access U-Boot's environment variables

05/06/2020

Biyong Sun

fw_printenv/fw_setenv

U-Boot provide `fw_printenv/fw_setenv` for Linux to access U-Boot environment variables.

It is compiled in the U-Boot, but used in Linux.

Test Environment

HW: i.MX8QXP MEK

SW: L4.14.98_2.0.0_ga

Build fw_printenv

```
source /opt/fsl-imx-xwayland/4.14-sumo/environment-setup-aarch64-poky-linux
```

```
make envtools CC="$CC"
```

Now, you will find the `u-boot/tools/env/fw_printenv`

Note: the `fw_setenv` is applet of `fw_printenv` as `busybox` does.

fw_env.config

fw_env.config default directory is in the /etc/

/dev/mmcblk1 **0x400000** **0x2000**

uboot/include/configs/imx8qxp_mek.h

0x2000 : #define CONFIG_ENV_SIZE 0x2000

0x400000: #define CONFIG_ENV_OFFSET (64 * SZ_64K)

Add all the files in target rootfs

`fw_env.config` copy to `/etc/`

`fw_printenv` copy to `/bin`

Make symbol link `fw_setenv` from `fw_printenv`

```
ln -s fw_printenv fw_setenv
```

`fw_setenv` -> `fw_printenv`

Test fw_printenv/fw_setenv

The first boot, the environment is generated by uboot C code.

i.MX uboot detects the which board it is, then give proper u-boot environment variables. It is in the memory not in the storage such as emmc, till you use saveenv to do save.

If it is the first time to do the test, get into uboot run saveenv first.

Before run saveenv

```
3F FF80: 00 00 00 00 00 00 00 00 00 00 00 00 .....
3F FF00: 00 00 00 00 00 00 00 00 00 00 00 00 .....
3F FF80: 00 00 00 00 00 00 00 00 00 00 00 00 .....
3F FF00: 00 00 00 00 00 00 00 00 00 00 00 00 .....
3F FF80: 00 00 00 00 00 00 00 00 00 00 00 00 .....
40 0000: 00 00 00 00 00 00 00 00 00 00 00 00 .....
40 0010: 00 00 00 00 00 00 00 00 00 00 00 00 .....
40 0020: 00 00 00 00 00 00 00 00 00 00 00 00 .....
40 0030: 00 00 00 00 00 00 00 00 00 00 00 00 .....
40 0040: 00 00 00 00 00 00 00 00 00 00 00 00 .....
40 0050: 00 00 00 00 00 00 00 00 00 00 00 00 .....
```

After run saveenv

```
3F FFC0: 00 00 00 00 00 00 00 00 00 00 00 00 .....
3F FF00: 00 00 00 00 00 00 00 00 00 00 00 00 .....
3F FF80: 00 00 00 00 00 00 00 00 00 00 00 00 .....
3F FF00: 00 00 00 00 00 00 00 00 00 00 00 00 .....
3F FF80: 00 00 00 00 00 00 00 00 00 00 00 00 .....
40 0000: 02 B3 EA 16 58 59 5A 3D 31 00 61 75 74 68 5F 6F ...XYZ=1.auth_o
40 0010: 73 3D 61 75 74 68 5F 63 6E 74 72 20 24 7B 63 6E s=auth_cntr ${cn
40 0020: 74 72 5F 61 64 64 72 7D 00 62 61 75 64 72 61 74 tr_addr}.baudrat
40 0030: 65 3D 31 31 35 32 30 30 00 62 6F 61 72 64 5F 6E e=115200.board_n
40 0040: 61 6D 65 3D 4D 45 4B 00 62 6F 61 72 64 5F 72 65 ame=MEK.board_re
40 0050: 76 3D 69 4D 58 38 51 58 50 00 62 6F 6F 74 5F 66 v=iMX8QXP.boot_f
40 0060: 64 74 3D 74 72 79 00 62 6F 6F 74 5F 6F 73 3D 62 dt=try.boot_os=b
```

Test fw_printenv/fw_setenv(Cont.)

Get in to linux then test fw_printenv

```
root@imx8qxpmeek:/# fw_printenv
auth_os=auth_cntr ${cntr_addr}
baudrate=115200
board_name=MEK
board_rev=iMX8QXP
boot_fdt=try
boot_os=booti ${loadaddr} - ${fdt_addr};
bootcmd=mmc dev ${mmcdev}; if mmc rescan; then if run loadbootscript; then run bootscrip; else if
if run loadimage; then run mmcboot; else run netboot; fi; fi; fi; else booti ${loadaddr} - ${fdt_a
bootcmd_mfg=run mfgtool_args;if iminfo ${initrd_addr}; then if test ${tee} = yes; then bootm ${tee}
else echo "Run fastboot ..."; fastboot 0; fi;
bootdelay=3
bootscrip=echo Running bootscrip from mmc ...; source
cntr_addr=0x98000000
cntr_file=os_cntr_signed.bin
commit_atf=1cb68fa
commit_mkimage=dd023400
commit_scfw=f83a2bed
commit_secofw=92ef1143
console=ttyLPO
dom0fdt_file=fsl-imx8qxp-mek-dom0.dtb
earlycon=lpuart32,0x5a060000
emmc_dev=0
ethact=ethernet@5b040000
ethaddr=00:04:9f:05:8e:ea
ethprime=eth0
fastboot_dev=mmc1
fdt_addr=0x83000000
fdt_file=fsl-imx8qxp-mek-rpmc.dtb
```


Test fw_printenv/fw_setenv(Cont.)

The linux fw_setenv to Set u-boot environment variables then verify in uboot
fw_setenv fdt_file fsl-imx8qxp-mek.dtb (Linux Side)

The reboot board and stop at u-boot, using “printenv” to check if it is changed.

```
emmc_dev=0
ethact=ethernet@5b040000
ethaddr=00:04:9f:05:8e:ea
ethprime=eth0
fastboot_dev=mmc1
fdt_addr=0x83000000
fdt_file=fsl-imx8qxp-mek.dtb
fdt_high=0xffffffffffffffff
fdtcontroladdr=85660208
image=Image
initrd_addr=0x83100000
initrd_high=0xffffffffffffffff
ib_mmcboot=setenv fdt_file fsl-imx8qxp-mek.dtb
```

U-Boot

Test fw_printenv/fw_setenv(Cont.)

Test Batch mode. Every time using fw_setenv to set single variable will cause write to storage such as emmc. Batch mode provide a way to set several variables and store them to the storage just one write.

1. Create scprit file call uboot_vars
uboot_vars
ABC 1
XYZ 2
2. fw_setenv -s uboot_vars (Linux Side)
3. Reboot board, Stop at uboot, printenv to check the changes

```
xenlinux_bootargs=  
xenlinux_console=hvc0 earlycon=xen  
xenmmcbboot=setenv get_cmd "fatload  
xennetboot=setenv get_cmd dhcp;set  
ABC=1  
XYZ=2  
root@imx8qxpme:~#
```

U-Boot

Demo on i.MX6ULL EVK

Demo description

In this demo, the normal 1st rootfs will use `fw_setenv` to set the u-boot bootargs to let the system to the 2nd recovery rootfs after reboot.

Once system goes into the 2nd recovery rootfs, it just prints out some information as a demo show. Then the recovery rootfs will also call the `fw_setenv` to let the system mount the 1st normal rootfs after reboot.

Demo using the `CONFIG_ENV_OFFSET_REDUND` to store the 2nd u-boot environment as backup.

Demo image layout



Major changes in u-boot code

Add CONFIG_ENV_OFFSET_REDUND

u-boot/include/configs/mx6ullevk.h

```
296 #define CONFIG_ENV_SIZE          SZ_8K
297 #if defined(CONFIG_ENV_IS_IN_MMC)
298 #define CONFIG_ENV_OFFSET        (14 * SZ_64K)
299 #define CONFIG_ENV_OFFSET_REDUND (14 * SZ_64K + CONFIG_ENV_SIZE)
300 #elif defined(CONFIG_ENV_IS_IN_SPI_FLASH)
```

Major changes in 1st rootfs(normal run)

-rwxr-xr-x 1 root root 33996 May 6 2020 fw_printenv

lrwxrwxrwx 1 root root 11 May 6 2020 fw_setenv -> fw_printenv

/etc/fw_env.config (configuration for fw_printenv/fw_setenv)

/dev/mmcblk1 0xE0000 0x2000

/dev/mmcblk1 0xE2000 0x2000 #redundancy

/usr/sbin/upg (script)

fw_setenv mmcroot /dev/mmcblk1p3 rootwait rw

reboot

Major changes in 2nd rootfs(recovery)

```
-rwxr-xr-x 1 root root 33996 May 6 2020 fw_printenv  
lrwxrwxrwx 1 root root 11 May 6 2020 fw_setenv -> fw_printenv
```

/etc/fw_env.config (configuration for fw_printenv/fw_setenv)

```
/dev/mmcblk1 0xE0000 0x2000
```

```
/dev/mmcblk1 0xE2000 0x2000 #redundancy
```

/etc/init.d/S50upg (script)

```
case "$1" in  
start)  
    echo "Upgrade Start..."  
    echo "*****"  
    sleep 1  
    echo "*****"  
    sleep 1  
    echo "*****"  
    sleep 1  
    echo "*****"  
    sleep 1  
    echo "*****"  
    sleep 1  
    echo "*****"  
    echo " "  
    echo "Upgrade Done! Reboot..."  
    fw_setenv mmcroot /dev/mmcblk1p2 rootwait rw  
    reboot  
    ;;  
stop)  
    exit 1  
    ;;  
restart|reload)  
    exit 1  
    ;;  
*)  
esac  
exit $
```


Demo screenshots

```
EXT4-fs (mmcblk1p2): re-mounted. Opts: data=order
Starting syslogd: OK
Starting klogd: OK
Running sysctl: OK
Initializing random number generator: OK
Saving random seed: random: dd: uninitialized urandom read (51
OK
1. type upg to show the demo
Starti
the upg script set u-boot env
Welcome to fw_printenv/fw_setenv demo
root login: root
type u
It is in the 1st rootfs(normal run)
the script is at /usr/sbin/upg
fw_setenv mmcroot /dev/mmcblk1p3 rootwait rw
reboot

#
#
# upg
```

```
Running sysctl: OK
Initializing random number
Saving random seed: random
OK
Starting network: OK
Upgrade Start...
*****
*****
*****
*****
*****
*****
Upgrade Done! Reboot...
Stopping network: OK
```

3. run to 2nd rootfs(recovery)
the /etc/init.d/S50upg script print out demo upgrade
set u-boot env mmcroot=/dev/mmcblk1p2 rootwait rw
then reboot

```
#
# upg
# Stopping network: OK
Saving random seed: random: dd: uninitialized urandom read (51
OK
Stopping klogd: OK
Stopping syslogd: OK
mount: devtmpfs busy - remounted read-only...
EXT4-fs (mmcblk1p2): re-mounted. Opts: data=order
The system is going down for reboot now...
Sent SIGTERM to all processes
Sent SIGKILL to all processes
Requesting system reboot
imx-sdma 20ec000.sdma: external firmware not found, using ROM
ci_hdrc ci_hdrc.1: remove, state 4
usb usb1: USB disconnect, device number 1
ci_hdrc ci_hdrc.1: USB bus 1 deregistered
reboot: Restarting system
```

```
Freeing unused kernel memory: 1024K
EXT4-fs (mmcblk1p2): re-mounted. Opts: data=order
Starting syslogd: OK
Starting klogd: OK
Running sysctl: OK
Initializing random number generator: OK
Saving random seed: random: dd: uninitialized urandom read (51
OK
4. back to the 1st rootfs(normal run)
Starting network:
```

```
Welcome to fw_printenv/fw_setenv demo
root login: root
type upg to show the upgrade demo
the script is at /usr/sbin/upg
fw_setenv mmcroot /dev/mmcblk1p3 rootwait rw
reboot

#
```