

Embedded Linux Course

OTA Project

Presented by:
Ahmed Nader



Project Description

- Vexpress kit (QEMU) is connected to a TFTP server and interfaced with SD Card which contains:
 - Test.elf => The applications
 - Vexpress_blob => pre-built environment variables
- U-boot choose between either application from SD card or TFTP depend on last updated version and backup old version in SD card

pseudo code

```
load test.elf file from MMC to ram in address 0x73000000;  
save test.elf size downloaded from MMC;  
setup TFTP credentials  
load test.elf file from TFTP to ram in address 0x70000000;  
if (load test.elf file from TFTP to ram in address 0x70000000)  
{  
    save test.elf size downloaded from TFTP server;  
    if ( MMC_file_size == tftp_file_size ){  
        print "same Version"  
    }  
    else{  
        print "different Version"  
        remove old backup.elf file from SD card;  
        save new image ,downloaded from TFTP server, to MMC under test.elf;  
        save old image ,already exist in mmc, to MMC under backup.elf;  
    }  
}  
else{  
    load from 0x70000000  
}  
boot test.elf
```

Pre-built Environment variables

```
load test.elf file from MMC to ram in address 0x73000000;
save test.elf size downloaded from MMC;
setup TFTP credentials
load test.elf file from TFTP to ram in address 0x70000000;
if (load test.elf file from TFTP to ram in address 0x70000000)
{
    save test.elf size downloaded from TFTP server;
    if ( MMC_file_size == tftp_file_size ){
        print "same Version"
    }
    else{
        print "different Version"
        remove old backup.elf file from SD card;
        save new image ,downloaded from TFTP server, to MMC under test.elf;
        save old image ,already exist in mmc, to MMC under backup.elf;
    }
}
else{
    load from 0x70000000
}
boot test.elf
```

```
setenv bootcmd 'fatload mmc 0:1 ${mmcLoadAddr} test.elf ;
setenv mmcFileSize $filesize ;
dhcp ;
setenv serverip 192.168.50.188;
if tftp ${tftpLoadAddr} test.elf ;
then;
    setenv tftpFileSize $filesize ;
    if test "${tftpFileSize}" = "${mmcFileSize}";
    then ;
        echo "same version";
    else ;
        echo "New version";
        fatrm mmc 0:1 backup.elf ;
        fatwrite mmc 0:1 ${mmcLoadAddr} backup.elf ${mmcFileSize};
        fatwrite mmc 0:1 ${tftpLoadAddr} test.elf ${tftpFileSize};
    fi;
else;
    setenv tftpLoadAddr $mmcLoadAddr ;
fi;
bootelf ${tftpLoadAddr}'
```

Implementation Phase



Prerequisites to run OTA project

- I. Uboot => executable uboot for vExpress
- II. KS_SD_512M.img => Virtual SD card
- III. lab2.sh => shell script to vExpress on QEMU

1- save environment variables in MMC

```
embedded_system_ks@embedded-KS: ~/labs/lab8/lab2
pulseaudio: Reason: Invalid argument
pulseaudio: set_sink input_mute() failed
pulseaudio: Reason: Invalid argument

U-Boot 2020.01-rc2-00035-g3ff1ff3ff7-dirty (Feb 01 2020 - 12:51:10 +0200)

DRAM: 512 MiB
WARNING: Caches not enabled
Flash: 128 MiB
MMC: MMC: 0
Loading Environment from FAT... *** Warning - bad CRC, using default environment

In: serial
Out: serial
Err: serial
Net: smc911x-0
Hit any key to stop autoboot: 0
=> lets save our environmnt variables
Unknown command 'lets' - try 'help'
=> setenv mmcLoadAddr 0x73000000
=> setenv tftpLoadAddr 0x70000000
=> setenv bootcmd 'fatload mmc 0:1 ${mmcLoadAddr} test.elf ; setenv mmcFileSize $filesize ; dhcp ; setenv serverip 192
.168.1.10; if tftp ${tftpLoadAddr} test.elf ; then ; setenv tftpFileSize $filesize ; if test "${tftpFileSize}" = "${mm
cFileSize}"; then ; echo "same FW version"; else ; echo "there is a new FW "; fatrm mmc 0:1 backup.elf ; fatwrite mmc
0:1 ${mmcLoadAddr} backup.elf ${mmcFileSize} ; fatwrite mmc 0:1 ${tftpLoadAddr} test.elf ${tftpFileSize} ; fi ; fi ;
bootelf ${tftpLoadAddr}'
=> saveenv
Saving Environment to FAT... OK
=>
=>
```

2- Run App over U-boot

embedded_system_ks@embedded-KS: ~/labs/lab8/lab2

67816 bytes read in 186 ms (355.5 KiB/s)

smc911x: MAC 52:54:00:12:34:56

smc911x: detected LAN9118 controller

smc911x: phy initialized

smc911x: MAC 52:54:00:12:34:56

BOOTP broadcast 1

DHCP client bound to address 10.0.2.15 (5 ms)

*** Warning: no boot file name; using '0A00020F.img'

Using smc911x-0 device

TFTP from server 10.0.2.2; our IP address is 10.0.2.15

Filename '0A00020F.img'.

smc911x: MAC 52:54:00:12:34:56

TFTP error: trying to overwrite reserved memory...

smc911x: MAC 52:54:00:12:34:56

smc911x: MAC 52:54:00:12:34:56

smc911x: detected LAN9118 controller

smc911x: phy initialized

smc911x: MAC 52:54:00:12:34:56

Using smc911x-0 device

TFTP from server 192.168.1.10; our IP address is 10.0.2.15; sending through gateway 10.0.2.2

Filename 'test.elf'.

Load address: 0x70000000

Loading: #####

61.5 KiB/s

done

Bytes transferred = 67816 (108e8 hex)

smc911x: MAC 52:54:00:12:34:56

same FW version

Starting application at 0x48000000 ...

first run_

embedded_system_ks@embedded-KS: ~/labs/lab8/lab2

```
1
2 setenv mmioLoadAddr 0x70000000
3
4 setenv tftpLoadAddr 0x70000000
5
6 setenv bootcmd "tftp 10.0.2.2 0x70000000 test.elf; bootm 0x70000000"
7
8 ./lab2.sh /media/embedded_system_ks/Em
KS_50_512M.img
```


3- change app in TFTP server and run APP again

```
embedded_system_ks@embedded-KS: ~/labs/lab7/bare_metal
volatile unsigned char * const UART0DR = (unsigned char *) 0x10009000;

void print_uart0(const char * s){
    while(*s != '\0'){
        *UART0DR = *s;
        s++;
    }
}

void c_entry(void){
    print_uart0("after uploading new app in tftp server");
}

-- INSERT --
```

```
1
2 setenv macLoadAddr 0x73880000
3
4 setenv tftpLoadAddr 0x70000000
5
6 setenv bootcmd "fatload mmc 0:1 $tftpLoadAddr $filesize; test ${tftpLoadAddr} test.elf; test ${tftpLoadAddr} test.elf; echo "version"; else; echo "no version"; fatwrite mmc 0:1 $tftpLoadAddr $filesize; fatload mmc 0:1 $tftpLoadAddr $filesize; test ${tftpLoadAddr} test.elf; echo "version"; else; echo "no version";"
7
```

KS: SD 512M.img

17,39-46 All

4- saving old version & save and run new one

```
embedded_system_ks@embedded-KS: ~/labs/lab8/lab2
smc911x: phy initialized
smc911x: MAC 52:54:00:12:34:56
BOOTP broadcast 1
DHCP client bound to address 10.0.2.15 (31 ms)
*** Warning: no boot file name; using '0A00020F.img'
Using smc911x-0 device
TFTP from server 10.0.2.2; our IP address is 10.0.2.15
Filename '0A00020F.img'.
smc911x: MAC 52:54:00:12:34:56

TFTP error: trying to overwrite reserved memory.
smc911x: MAC 52:54:00:12:34:56
smc911x: MAC 52:54:00:12:34:56
smc911x: detected LAN9118 controller
smc911x: phy initialized
smc911x: MAC 52:54:00:12:34:56
Using smc911x-0 device
TFTP from server 192.168.1.10; our IP address is 10.0.2.15; sending through gateway 10.0.2.2
Filename 'test.elf'.
Load address: 0x70000000
Loading: #####
          55.7 KiB/s

done
Bytes transferred = 67848 (10908 hex)
smc911x: MAC 52:54:00:12:34:56
there is a new FW
backup.elf: doesn't exist
67816 bytes written
67848 bytes written
## Starting application at 0x48000000 ...
after uploading new app in tftp server
```

5- Run old APP (backup.elf)

```
embedded_system_ks@embedded-KS: ~/labs/lab8/lab2
Specify the 'raw' format explicitly to remove the restrictions.
pulseaudio: set_sink_input_volume() failed
pulseaudio: Reason: Invalid argument
pulseaudio: set_sink_input_mute() failed
pulseaudio: Reason: Invalid argument
arm-none-eabi-gcc -c -mcpu=arm926ej-s -I . -g test.c -o test.o
arm-none-eabi-ld -T test.ld test.o startup.o -o test.elf
arm-none-eabi-objcopy -O binary test.elf test.bin
echo "finished"
finished
U-Boot 2020.01-rc2-00035-g3ff1ff3ff7-dirty (Feb 01 2020 12:51:10 +0200)
DRAM: 512 MiB
WARNING: Caches not enabled
Flash: 128 MiB
MMC: MMC: 0
Loading Environment from FAT... OK
In: serial
Out: serial
Err: serial
Net: smc911x-0
Hit any key to stop autoboot: 0
=> fatls mmc 0:1
    67848  test.elf
   262144  test_uboot.env
    67816  backup.elf

3 file(s), 0 dir(s)

=> fatload mmc 0:1 0x70000000 backup.elf
67816 bytes read in 237 ms (279.3 KiB/s)
=> bootelf 0x70000000
## Starting application at 0x48000000 ...
first run
```


6- Run new APP (test.elf)

```
embedded_system_ks@embedded-KS: ~/labs/lab8/lab2
embedded_system_ks@embedded-KS:~/labs/lab8/lab2$ let try our new APP ^C
embedded_system_ks@embedded-KS:~/labs/lab8/lab2$ ./lab2.sh /media/embedded_system_ks/Embedded_KS_labs/u-boot/u-boot KS
_SD_512M.img
WARNING: Image format was not specified for 'KS_SD_512M.img' and probing guessed raw.
Automatically detecting the format is dangerous for raw images, write operations on block 0 will be restricted.
Specify the 'raw' format explicitly to remove the restrictions.
pulseaudio: set_sink_input_volume() failed
pulseaudio: Reason: Invalid argument
pulseaudio: set_sink_input_mute() failed
pulseaudio: Reason: Invalid argument
U-Boot 2020.01-rc2-00035-g3ff1ff3ff7-dirty (Feb 01 2020 - 12:51:10 +0200)

DRAM: 512 MiB
WARNING: Caches not enabled
Flash: 128 MiB
MMC: MMC: 0
Loading Environment from FAT... OK
In: serial
Out: serial
Err: serial
Net: smc911x-0
Hit any key to stop autoboot: 0
=> fatload mmc 0:1 0x70000000 test.elf
67848 bytes read in 228 ms (290 KiB/s)
=> bootelf 0x70000000
## Starting application at 0x48000000 ...
after uploading new app in tftp server qemu-system-arm: terminating on signal 2
embedded_system_ks@embedded-KS:~/labs/lab8/lab2$
```

7- Run again u-boot

```
embedded_system_ks@embedded-KS: ~/labs/lab8/lab2
67848 bytes read in 257 ms (257.8 KiB/s)
smc911x: MAC 52:54:00:12:34:56
smc911x: detected LAN9118 controller
smc911x: phy initialized
smc911x: MAC 52:54:00:12:34:56
BOOTP broadcast 1
DHCP client bound to address 10.0.2.15 (18 ms)
*** Warning: no boot file name; using '0A00020F.img'
Using smc911x-0 device
TFTP from server 10.0.2.2; our IP address is 10.0.2.15
Filename '0A00020F.img'.
smc911x: MAC 52:54:00:12:34:56

TFTP error: trying to overwrite reserved memory...
smc911x: MAC 52:54:00:12:34:56
smc911x: MAC 52:54:00:12:34:56
smc911x: detected LAN9118 controller
smc911x: phy initialized
smc911x: MAC 52:54:00:12:34:56
Using smc911x-0 device
TFTP from server 192.168.1.10; our IP address is 10.0.2.15; sending through gateway 10.0.2.2
Filename 'test.elf'.
Load address: 0x70000000
Loading: #####
54.7 KiB/s
done
Bytes transferred = 67848 (10908 hex)
smc911x: MAC 52:54:00:12:34:56
same FW version
## Starting application at 0x48000000 ...
after uploading new app in tftp server
```


Explanatory video



Embedded_System_KS [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Terminal

embedded_system_ks@embedded-KS: ~/labs/lab8/lab2

embedded_system_ks@embedded-KS:~/labs/lab8/lab2\$

```
1
2 setenv mmcLoadAddr 0x73880000
3
4 setenv tftpLoadAddr 0x70000000
5
6 setenv bootcmd 'tftpload mmc 0:1 ${mmcLoadAddr} test
  mmcFileSize ${filesize} ; dump ; setenv serverip 19
  ${tftpLoadAddr} test.elf ; then ; setenv tftpFile
  test "${tftpFileSize}" = "${mmcFileSize}"; then ;
  version"; else ; echo "there is a new FW "; fatrn
  [atwrite mmc 0:1 ${mmcLoadAddr} backup.elf ${mmc
  mmc 0:1 ${tftpLoadAddr} test.elf ${tftpFileSize}
  ${tftpLoadAddr}
7
8 /lab2 sh /media/embedded_system_ks/Embedded_KS_1
  KS_5D_512M.img
```

Plain Text Tab Width: 9

2:12 AM 2/6/2020



thank you!

