

## Noțiuni introductive

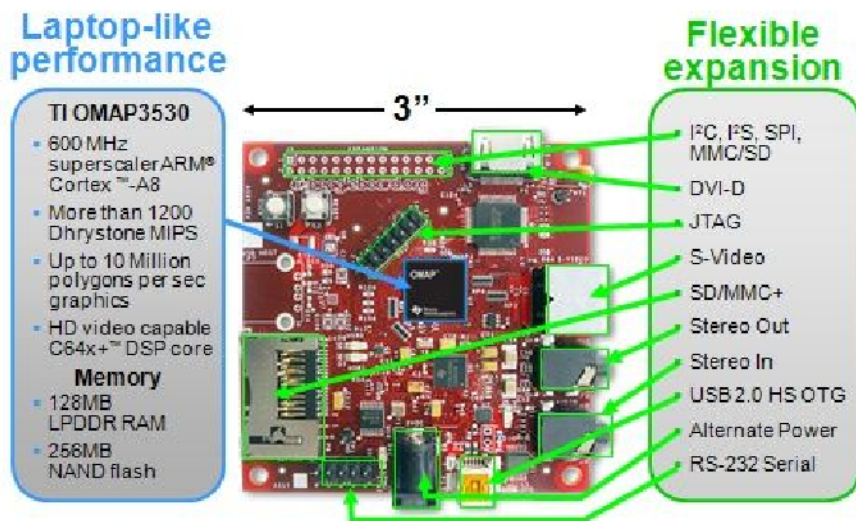
### Recapitulare elemente importante ale plăcii Beagleboard

Deoarece în acest laborator vom lucra din nou cu placa SBC Beagleboard, reamintim noțiunile de bază legate de periferice, și modul de conectare al plăcii.

**Reamintim observația din laboratorul precedent referitoare la grija de intarctiunea cu placa, și referitoare la prevenirea descărcării electrostatice. În cazul în care vă pișcați la atingerea unei suprafețe metalice în intrarea laboratorului chemați profesorul. Placa nu se atinge decât pe margini, și dacă trebuie contact direct pe altă regiune decât butoanele, se recomandă atingerea unei suprafețe metalice mari de pe placă (de exemplu portul SVideo).**






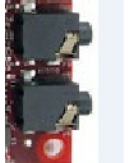

**Placa Beagleboard** a fost una din primele plăci de dezvoltare tip kit SBC low cost. Când a apărut pe piață costa ~USD150, într-o vreme când kituri similar erau în jur de USD1000+. Site-ul oficial este [beagleboard.org](http://beagleboard.org). A apărut în câteva variante de echipare hardware și revizii.

Cea mai populară a fost revizia C. Din cauza evoluției tehnologice, placa a fost înlocuită de Beagleboard-xM, Pandaboard, Beaglebone, Beaglebone BLACK. Este frecvent ca la căutări pe net să găsiți ceva referitor la unul din succesorii săi, nu legat de placa în sine.



În lucrările de la SMP vom interacționa cu următoarele periferice:

- Slot SD /card SD
- Port serial RS232, prin adaptor RS232-DB9
- Port USB Host – prin care conectăm periferice (tastatură, mouse, hub) la placă
- Alimentare 5VDC
- Port USB OTG – prin care accesăm placa ca gadget dinspre calculator
- Mufa audio Out / Audio In
- Mufa HDMI

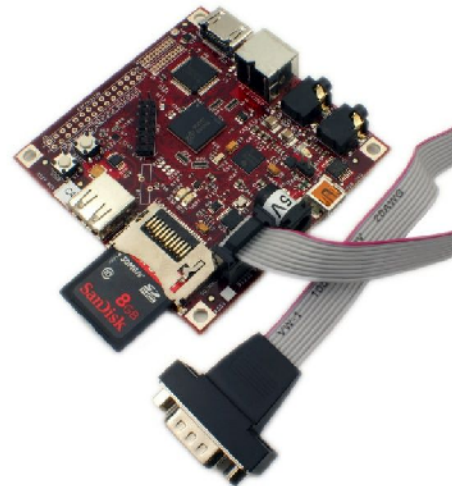
Slot SD	Port serial RS232, cablu adaptor RS232-DB9	Port USB Host	Alimentare 5VDC	Port USB OTG	Mufa Audio IN/Out	Mufa HDMI
						

Conectarea plăcii la PC

Card SD



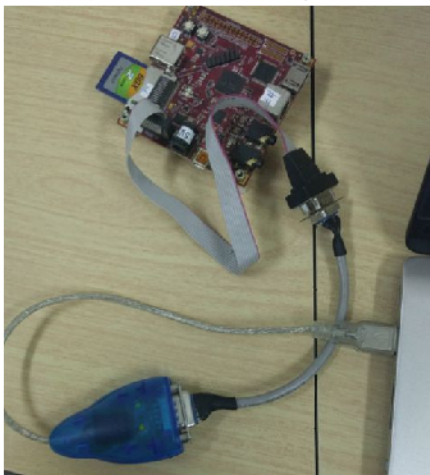
Modul de conectare al cardului SD și al  
cablului RS232-DB9.



Cablu null-modem si cablu USB-serial



Conexiunea serial – cablu serial adaptor, null-modem, si USB-serial înlanțuite



Legarea cablului HDMI-DVI



## Secvența de bootup.

Din laboratorul precedent ați aflat că toate sistemele cu microprocesoare au o procedură de inițializare în stagii.

Într-un prim stagiu de bootare (așa numitul BootROM, first boot loader, sau FBL), SOC-ul ARM decide de unde să încarce programul X-Loader. În general pe plăci Opțiunea poate fi hardcoded (prin legarea la masă sau alimentare a unor pini de pe SOC, sau poate fi

Opțiunile pot fi (conform site-ului oficial [http://processors.wiki.ti.com/index.php/Boot\\_Sequence](http://processors.wiki.ti.com/index.php/Boot_Sequence))

1. NAND (memoria de pe placă),
2. MMC/SD – de pe cardul inserat, citind fișierul MLO, din prima partiție de boot).
3. UART

Placa Beagleboard nu are inclus un port Ethernet, ca atare nu are suport pentru bootare prin TFTP.

O secvență de bootare pentru placa Beagleboard (processor cu arhitectura ARM, tip OMAP3530 de la Texas Instruments)

1. BootROM
2. Xloader (denumit MLO pe card, în prima partiției, boot).
3. Uboot (denumit u-boot.img sau u-boot.bin pe placă, în prima partiție, boot). Uboot va căuta un fișier boot.scr pe prima partiție, și în caz că îl găsește încearcă să autoexecute operațiile.
4. Boot kernel Linux (tipic fișier comprimat ulmage, uneori însoțit de un RAM file system primordial, destinat încărcării în RAM la bootare)

## Pornirea plăcii. Interacțiune cu uboot

Porniți Linux.

Porniți Terminal.

Legăți placa de dezvoltare prin suita de cabluri seriale la un port USB din PC.

Legați placa prin cablu HDMI la monitor.

Legați un hub Usb+Ethernet la placă în portul USB Host al plăcii.

Identificați portul `/dev/tty*` care corespunde adaptorului USB-serial pe care de abia l-ați inserat.

#### Modalitatea 1:

Scoateți cablul usb-serial din PC.

Tastați în terminal `ls /dev/tty` apoi apăsați tasta TAB.

Fiți atenți la finalul listei.

Inserați cablul USB-serial.

Fiți atenți la finalul listei.

#### Modalitatea 2:

Inserați cablul USB-serial.

Tastați

```
dmesg |tail
```

Ultimele mesaje ar trebui să comunice identificarea unui device nou căruia îi este asignat portul `ttyUSB0` (sau `ttyUSB1`, `2`,...etc). Tastați în terminal `ls /dev/tty` apoi apăsați tasta TAB.

Fiți atenți la finalul listei. Ar trebui să apară o intrare nouă tip `/dev/ttyUSB0`, care corespunde cablului USB-serial.

in funcție de câte astfel de dispozitive aveți instalate în calculator la momentul respectiv).

**Presupunem în continuarea acestui îndrumar că portul detectat va fi `/dev/ttyUSB0`. Dacă portul diferă, de exemplu `/dev/ttyUSB1`, ajustați corespunzător următorii pași.**

Porniți `screen`, utilitarul de tip console peste interfețe seriale. Ca parametrii comanda acceptă

- portul serial (`/dev/ttyUSB0`, în exemplul nostru, dar pe PC-ul vostru ar putea să difere).
- configurația portului serial care conține 3 elemente **115200n8**
  - viteza BAUD serial (viteza de comunicație conform protocolului RS232), 115200 baud
  - paritatea (dacă în comunicație se folosește bit de paritate, opțiuni pot fi n-none, o-odd, e-even). În cazul nostru folosim n de la none.
  - Numărul de biți ai cuvântului 7,8 uzual. Noi folosim 8.

Configurația 115200n8 se găsește în manualul plăcii (sau în documentația aferentă secvenței de bootare) și este obișnuit să se ofere împreună cu un produs la care utilizatorul final va accesa produsul prin serială.

```
screen /dev/ttyUSB0 115200n8
```

În această fereastră de terminal ar trebui să vedeți outputul de la placă. Dacă apare vreo problemă de conexiune între placă și PC, fereastra `screen` se va închide cu un mesaj tip

```
[screen has terminated]
```

**Înainte de alimentarea plăcii chemați profesorul pentru a verifica corectitudinea conexiunii.**

Alimentați placa.

Leगाți placa printr-un cablu usb-miniusb între un port USB de la PC și portul host de pe placă.

Observați secvența de bootare. Aceasta ar trebui să fie similară cu cea de mai jos, posibilele surse de diferență semnificative pentru noi fiind cele bolduite.

```
Texas Instruments X-Loader 1.4.2 (Feb 19 2009 - 12:01:24)
Reading boot sector
Loading u-boot.bin from mmc
```

```
U-Boot 2010.06 (Jan 14 2011 - 15:46:27)
```

```
OMAP3430/3530-GP ES3.0, CPU-OPP2 L3-165MHz
OMAP3 Beagle board + LPDDR/NAND
I2C: ready
DRAM: 256 MiB
NAND: 256 MiB
In: serial
Out: serial
Err: serial
Beagle Rev C1/C2/C3
Die ID #133a00030000000040323090b00901e
Hit any key to stop autoboot: 0
OMAP3 beagleboard.org #
```

În mod normal, cardul SD va fi detectat de BootROM, și acesta va boota XLoader și uboot de pe cardul MMC.

### **Tema 1. Interațiune cu screen.**

Implicit, screen nu vă va permite scroll înapoi în fereastră, așa cum o face terminal-ul de obicei.

Dacă doriți să vizualizați text care nu mai este în zona vizibilă a ecranului screen trebuie folosită o procedură specială.

Folosiți combinația Ctrl-A, apoi apăsați ESC. În acest mod de lucru, screen accepta scroll back, și nu mai puteți interacționa direct cu seriala.

Faceți scroll back pentru a vizualiza output mai vechi din serială.

Pentru a reveni la modul normal (în care comunicați bidirecțional cu placa), trebuie tastat Esc din nou.

**Tema 2. Cazul A1. Tastați în console `reset`. Notați outputul consolei seriale pentru regiunile marcate cu roșu mai sus. Coincid sau nu cu exemplul de mai sus?**

**Cazul A2:** Scoateți cardul SD din placă și tastați în command promptul UBoot comanda

reset

**Tema 3. Notați din nou outputul de la seriala. S-a schimbat ceva în regiunile marcate cu roșu?**

Verificarea originii programelor XLoader și UBoot rulate la secvența de inițializare.

**Tema 4. Notați semnificația fișierelor care apar în partiția boot (pe baza informațiilor din pagina 3).**

```
MLO .....  
u-boot.bin.....,  
boot.scr.....,  
ulmage.....
```

**Tema 5. Copierea MLO, u-boot.bin, boot.scr, ulmage de pe SDcard pe PC pentru o inspecție ulterioară.**

Se scoate cardul SD din placă și se inserează în PC.

În Linux Mint partițiile de pe card vor apărea automat pe desktop, și în ferestre de tip File explorer (nautilus sau nemo).

Copiați fișierul MLO, u-boot.bin, și boot.scr de pe prima partiție a SD-cardului în directorul /home/student/Downloads.

**Tema 6. Citirea și verificarea conținutului boot.scr**

Deschideți o altă consolă Terminal pe PC, decât cea care rulează deja screen. Schimbați directorul current din Terminal în Downloads.

```
cd /home/student/Downloads
```

Fișierul boot.scr conține în mod text setul de comenzi care ar trebui rulate (de către uboot) pentru a autoboota cu succes imaginea de pe cardul respectiv. Acesta conține și câteva caractere neprintabile, folosite pentru gestiunea operației de bootare de către uboot.

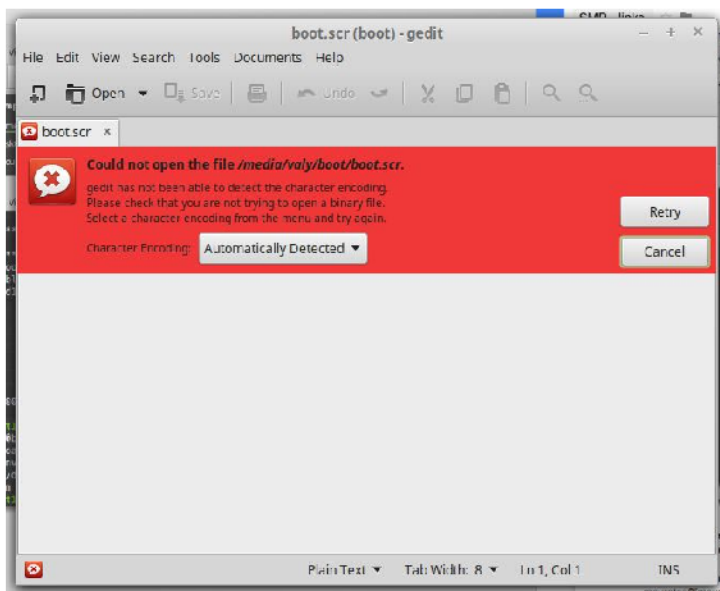
Datorită prezentei caracterelor neprintabile majoritatea editoarelor text vor refuza deschiderea acestui fișier.

Pentru a demonstra acest lucru încercați să deschideți cu gedit boot.scr

Rulați în terminal

```
gedit boot.scr
```

Comanda va eșua cu un mesaj tip Could not open the file. Gedit, și în general editoarele complexe nu sunt soluție pentru a vizualiza fișiere “binare”.



Studiați conținutul `boot.scr` cu comanda `cat` (care vă reamintesc citește conținutul unui fișier și îl afișează la linia de comandă).

```
cat boot.scr
```

Simplitatea de operare a comenzii `cat` (care citește orice fișier și afișează conținutul la consolă), este motivul pentru care aceasta funcționează.

Exemplu de listing pentru `boot.scr`:

```
'VpJ M4b m=Execute ulmage.binZif fatload mmc 0 80200000 ulmage
then
echo ***** Kernel: /dev/mmcblk0p1/ulmage *****
fi
echo ***** RootFS: /dev/mmcblk0p2 *****
setenv bootargs 'console=ttyS2,115200n8 androidboot.console=ttyS2 mem=256M
root=/dev/mmcblk0p2 rw rootfstype=ext3 rootdelay=1 init=/init ip=off mpurate=720
omap_vout.vid1_static_vrfb_alloc=y'
bootm 0x80200000
```

Inițializarea la nivelul uboot conține de obicei următoarele etape

- Montarea cardului SD pentru a îl face vizibil pentru uboot (comanda `mmc init`)
- Încărcarea de pe prima partiție a cardului (`mmc 0` sau `mmc 0:1`) a cărui sistem de fișiere este FAT, a unui fișier la o adresă de memorie fizică (data în valori hexazecimale) stabilită de utilizator.
  - Tipic (pentru beagleboard) fatload `mmc 0 80200000 ulmage`
- ajustarea parametrilor de environment uboot cu `setenv`. Tipic se introduce un argument `bootargs`, care reprezintă setul de parametri pe care uboot îi pasează Linuxului în faza de bootare

Se remarcă o parte din succesiunea de comenzi care trebuie introdusă de la tastatură este prezentă. Comanda care lipsește e chiar prima din secvență (`mmc init`).

### Tema 7. Studiați stringurile conținute în binarele MLO și uboot.bin

Dorim să vedem ce versiune sunt binarele de pe card, și să le comparăm cu binarele care au rulat pe placă în cazul A1 și A2.

Pe PC în directorul Downloads rulați următoarele comenzi.

```
strings MLO
strings u-boot.bin
```

Pentru a ușura identificarea secvenței text rulați comenzile rafinate

```
strings MLO | grep -i x-loader
strings u-boot.bin | grep -I u-boot\ 2
```

Notați pe hârtie următoarele

Versiune XLoader placă.....
Versiune Xloader citită cu strings.....
Versiune Uboot placă.....
Versiune Uboot citită cu strings.....

### Tema 8. Studiu comenzi uboot

R: Ejectați corect cardul de memorie SD: în felul următor: pe PC, pe desktop, faceți click dreapta pe iconița boot. Selectație Eject.

Inserați cardul SD înapoi în placă.

Se selectează fereastra Terminal, care conține screen-ul (consola serială către placa SBC).

Se tastează pentru montarea cardului SD astfel încât să devină vizibil pentru placă.

```
mmc init
```

Se studiază setul de comenzi disponibile în uboot.

În consola screen (deci serial de legătură cu placa) tastați

```
help
```

Notați ce spune help despre următoarele comenzi.

fatload.....
fatls.....
mmc init.....
bootm.....
setenv.....
printenv.....

Puteți tasta comenzi speciale pentru comenzile compuse

```
help usb
```



```
help i2c
```

Se vizualizeaza setul de fişiere de pe prima partiţie a SDcardului direct din uboot.

```
fatls mmc 0
```

Prin fatls efectuăm o comandă similar cu dir/ls, de la primul laborator, pe partiţie tip FAT, a sdcardului (mmc), pe partiţia 0 a SDCardului (parametrul 0).

### Tema 9. Bootaţi în android.

Finalizaţi bootarea în android.

Introduceţi comenzile necesare bootării în android.

#### Metoda 1 – cu copy paste din cat-ul boot.scr în fereastra screen

Scrieţi în consola screen

```
mmc init
```

Pe PC în terminalul care nu conţine screen, faceţi

```
cat boot.scr
```

Selectaţi cu mouseul secţiunea fatload mmc 0 80200000 uImage din outputul terminalul,

Selectaţi fereastra terminal care rulează screen. Apăsaţi butonul de mijloc al mouse-ului.

#### Metoda 2 – prin tastare direct în screen

În consola serială de pe placă (accesibilă în fereastra screen) tastaţi comenzile de mai jos (atenţie, comanda setenv se introduce pe o singură linie).

```
mmc init
```

```
fatload mmc 0 80200000 uImage
```

```
setenv bootargs
```

```
'console=ttyS2,115200n8
```

```
androidboot.console=ttyS2
```

```
mem=256M root=/dev/mmcblk0p2 rw
```

```
rootfstype=ext3 rootdelay=1
```

```
init=/init ip=off mpurate=720
```

```
omap_vout.vid1_static_vrfb_allo  
c=y'
```

```
bootm 0x80200000
```

Observaţi bootarea în android.

Pentru început remarcăm faptul că uboot încarcă programul aflat în RAM, la adresa 0x80200000 (valoare hexazecimală). Această regiune a fost scrisă anterior de comanda fatload mmc 0 80200000 ulmage, deci uboot încarcă programul ulmage.

```
OMAP3 beagleboard.org # bootm 0x80200000
```

```
## Booting kernel from Legacy Image at 80200000 ...
```

Uboot detectează tipul de imagine (linux) și atributele ei (ARM, uncompressed), și efectuează o verificare a corectitudinii binare (prin calculul sumei de control checksum)

```
Image Name: Linux-2.6.32
```

```
Image Type: ARM Linux Kernel Image (uncompressed)
```

```
Data Size: 2618368 Bytes = 2.5 MiB
```

```
Load Address: 80008000
```

```

Entry Point: 80008000
Verifying Checksum ... OK
Loading Kernel Image ... OK
OK

```

Se decomprimă imaginea de la 0x80200000, în regiunea de memorie 0x80008000. Fișierul uImage conține kernelul și informații de control într-o formă comprimată, de obicei Gzip, sau LZMA.

*Starting kernel ...*

```

Uncompressing Linux.....
..... done, booting the kernel.

```

Se încarcă apoi instrucțiunea de la entry pointul găsit anterior (0x80008000), și dacă procedura a fost corectă următoarele mesaje care apr țin de consola Linux.

Mai întâi apar veriunea de Linux (pe care ulterior o putem citi din consola Linux cu cat /proc/version).

```

Linux version 2.6.32 (a0393957@swubn01) (gcc version 4.4.0 (GCC) ) #4 Wed Jan 19
15:12:38 IST 2011

```

Urmează datele de identificare ale procesorului și mașinii Linux (o abstracție pentru placa Beagleboard).

```

CPU: ARMv7 Processor [411fc083] revision 3 (ARMv7), cr=10c53c7f
CPU: VIPT nonaliasing data cache, VIPT nonaliasing instruction cache
Machine: OMAP3 Beagle Board
Memory policy: ECC disabled, Data cache writeback
OMAP3430/3530 ES3.1 (l2cache iva sgx neon isp 720m)

```

Ceva inițializare de memorie.

```

SRAM: Mapped pa 0x40200000 to va 0xfe400000 size: 0x100000
Reserving 4194304 bytes SDRAM for VRAM
Built 1 zonelists in Zone order, mobility grouping on. Total pages: 65024

```

Remarcați comanda pasată de uboot (prin parametrul bootargs pe care l-ati tastat mai devreme).

```

Kernel command line: console=ttyS2,115200n8 androidboot.console=ttyS2 mem=256M
root=/dev/mmcblk0p2 rw rootfstype=ext3 rootdelay=1 init=/init ip=off mpurate=720
omap_vout.vid1_static_vrfb_alloc=y

```

Dacă totul este în regulă ultimile comenzi pe serial din consola Linux vor fi de tipul

```

# enabling adb
adb open

```

Puteți verifica buna bootare schimbând și canalul monitorului de pe VGA pe DVI.

## Flashing. Programarea plăcii SBC prin flashing-ul SDcard

De obicei softul pentru plăcile SBC, în specila partea de OS, vine în două forme

- imaginile prebuilt
  - tip imagine complete
  - tip imagini parțiale cu script de inscripționare al SD cardului
- surse care trebuie compilate local

În continuare studiem procesul de flashuire al SDcardului cu android pentru placa SBC. Pentru android se va folosi metoda cu inscripționare prin script, și pentru ubuntu metoda inscripționării imaginii complete.

### Flashing Android.

Se rebootează placa

Metoda 1. Apăsare buton reboot de pe placă

Metoda 2. Se tastează reboot în consola screen (dacă sunt în Linux, sau reset, dacă sunt în uboot)

Se scoate cardul SD din placă și se baga în PC.

În PC, căutați în directorul /home/student/Documents/smp/ fișierul beagleboard-rev-c4.tar.gz

Dezarhivați-l folosind interfața grafică a File Explorer-ului.

Mergeti în interiorul său, până ajungeți la nivelul cu mkmmc-android.sh

Deschideți o fereastră terminal în acest director, folosind click dreapta în file explorer, și din meniul contextual alegând open In terminal.

Descoperiți drive-ul asociat cu sdcardul. Chemați profesorul pentru a vă face un shell administrativ.

În shell-ul root, cu ajutorul profesorului efectuați procesul de flăshuire al SDcardului, cu imaginea Froyo.

Tastați apoi `sync`

Comanda `sync` se asigură că toată informația de copiat pe card a ajuns pe card. Linux folosește un mecanism de deferred copying, care are performante IO mai bune în general, dar care în lipsa `sync` poate duce la o copiere defectoasă pe SD:

Pentru a ejecta corect cardul SD, cereți opinia profesorului.

Inserați cardul SD în placă.

Introduceți comenzile de bootare în android din nou (fie cu copy paste din boot.scr-ul android pe care l-ați salvat anterior în /home/students/Downloads, fie tastând literă cu literă de la tastatură).

Observați bootarea cu succes.

## ADB

Asigurați-vă că placa este legată la PC și printr-un cablu miniusb.

Porniti adb ca administrator.

Vizualizați tipurile de comenzi pe care le puteți rula cu adb.

Tastați

```
adb
```

Vizualizați device-urile conectate.

Dacă sunteți bootați în android și în consola serială (screen) vedeți mesajele de enabling adb adb open, înseamnă că demonul de la nivelul plăcii este funcțional.

Pentru a vedea placa din PC tastați

```
adb devices
```

Notați outputul, Device-ul Beagleboard este identificat printr-un serial (probabil data de fabricație a imaginii, din 2011).

Comenzile cu care vom lucra astăzi sunt comenzile de sideload (adb install, adb uninstall), enable/dezabla program android.

## Sideload

Instalarea unui program android (tip apk) prin sideload se face cu comanda adb install urmată de pachetul apk dorit, într-o fereastră de terminal PC obișnuită.

Pentru instalarea unui program neapk se folosește comanda adb push.

Căutați în /home/student/Documents//smp pachetul apk pentru benchmarkul Nenamark. Apoi instalați-l.

```
adb install benchmarks/NenaMark1-v1.0.apk
[100%] /data/local/tmp/NenaMark1-v1.0.apk
  pkg: /data/local/tmp/NenaMark1-v1.0.apk
Success
```

Intrați pe interfața android și rulați benchmarkul o dată.

Reveniți în consola adb.

```
adb uninstall se.nena.nenamarkl
```

**Succes**

Încercați să rulați din nou benchmarkul din interfața android. Ce observați?

Rulați din interfața android aplicația clock. Ieșiți din ea. Dezactivați-o.

```
adb shell pm disable com.android.deskclock
```

**Package com.android.deskclock new state: disabled**

Încercați să intrați din nou în aplicația android. Ce observați?

Re-enațiați aplicația Clock.

```
adb shell pm enable com.android.deskclock
```

**Package com.android.deskclock new state: enabled**

Instalați programul busybox în /data/data/

```
adb push busybox /data/data
```

Încercați să îl rulați

```
adb shell busybox
```

## Bonus: Flashing Ubuntu.

Pentru Ubuntu avem o imagine complete.

Se rebootează placa

Metoda 1. Apăsare buton reboot de pe placă

Metoda 2. Se tastează reboot în consola screen (dacă sunt în Linux, sau reset, dacă sunt în uboot)

Se scoate cardul SD din placă și se baga în PC.

În PC, căutați în directorul /home/student/Documents/smp/ fișierul Ubuntu-12.04-desktop.img

Identificați drive-ul dispozitivului SD.

Chemați profesorul și utilizați dd pentru a flashui imaginea de Ubuntu pe sdcard.

Ejectați corect cardul.