

# Jedan pristup prilagođenju Android operativnog sistema digitalnom TV prijemu

Tomislav Maruna, Stanko Novaković, Svetozar Miućin, Milan Savić,  
Nikola Kuzmanović, Goran Radenović

**Sadržaj** — U radu su objašnjeni osnovni koraci prilagođavanja Android operativnog sistema za digitalni TV prijemnik. Cilj rada je studija izvodljivosti prenošenja Android platforme na specifični hardver koji se koristi u digitalnim TV prijemnicima visoke definicije.

**Ključne reči** — Android, Kernel, Linux, MIPS, SoC.

## I. UVOD

ANDROID je platforma za mobilne uređaje razvijena od strane kompanije Google. Realizovan je u obliku softverskog steka koji uključuje jezgro operativnog sistema, srednji (middleware) i aplikativni sloj programa. [1] Osnovni razlog velikog interesovanja za Android je dostupnost u izvornom kodu. Samim tim Android je pogodan za dalja poboljšanja i modifikacije u skladu sa potrebama pojedinaca širom sveta. Ceo stek je, iako prvobitno namenjen uređajima zasnovanim na ARM (Advanced RISC machine) arhitekturi, 2009. godine prilagođen i MIPS (Microprocessor without Interlocked Pipeline Stages) arhitekturi procesora što je otvorilo nove mogućnosti za korišćenje u čitavom novom spektru uređaja od televizora, VoIP telefona do digitalnih okvira za slike. Samu MIPS adaptaciju Androida je dalje neophodno prilagoditi određenoj ciljnoj hardverskoj platformi. Razlog je prisutnost različitog perifernog hardvera kao i samih procesora na različitim platformama. Kao početna tačka u procesu prilagođavanja je modifikacija Linux jezgra koja obuhvata dodavanje hardverski zavisnog koda specifičnog za datu platformu kao i svih neophodnih rukovalaca perifernim uređajima.

## II. OPIS HARDVERSKE PLATFORME

Ciljna platforma za prilagođenje Android-a u ovom radu je Micronas VCTH PlatinumD digitalni TV prijemnik.

U osnovi prijemnika se nalazi 32-bitni MIPS74 centralni procesor. SoC(System-on-a-chip) arhitektura povezuje centralni procesor i memoriju sa procesnim jedinicama pomoću sistema magistrala visoke propusnosti.

Centralni procesor kontroliše jedinice za obradu slike preko sistemske kontrolne magistrale, odvojene od magistrale koja povezuje sve procesne elemente sa memorijom. Sjedinjena arhitektura memorije omogućuje deljenje memorijski mapiranih podataka između procesora.

Većina specijalizovanih procesnih jedinica funkcioniše autonomno, što ostavlja centralnom procesoru dovoljno resursa za Linux/Android OS i korisničke aplikacije.

## III. OPIS ANDROID PLATFORME

Na najnižem nivou Android steka se nalazi Linux jezgro. Srednji nivo sadrži opšte Linux biblioteke za rad sa multimedijom, web-om, itd. kao i posebnu-redukovanu verziju standardne C biblioteke (libc) - Bionic, razvijene u cilju smanjenja zahtevnosti za resursima. U srednjem sloju takođe se nalazi osnova za funkcionalnost aplikacija - Dalvik virtualna mašina. Dalvik je derivat Java virtualne mašine, modifikovana da koristi što manje resursa i bude pogodna za izvršavanje na uređajima sa ograničenim resursima. Sprega za programiranje aplikacija za Android Operativni Sistem je zasnovana na standardnom Java API okruženju. Iz okruženja su uklonjene biblioteke za podršku štampanju, AWT (Abstract Window Toolkit) grafičko okruženje, Java Beans, i druge. Dodata je nova grafička sprega i zadržana Java podrška za rukovanje zvukom, slikom, mrežom i ostale biblioteke koje mogu naći primenu u mobilnim uređajima. Na samom vrhu se nalazi aplikativni sloj koji obuhvata Java aplikacije. Iako su prevedene u međukod sličan Java međukodu (Java bytecode), Android aplikacije nisu izvršive u okviru Java virtualne mašine.



Sl. 1. Dijagram Android softverskog steka.

## IV. RAZVOJNO OKRUŽENJE

Android stek za MIPS arhitekturu je dostupan u obliku GIT (fast version control system) repozitorijuma.[2] Preporučeni paket alata je mips-4.3 gcc toolchain i Java 1.5 verzija. Linux jezgro koje je sadržano u sklopu ove distribucije Android-a je modifikovano jezgro verzije 2.6.27. Okruženje za prevođenje izvornog koda je organizovano tako da koristi alat make i hijerarhiju konfiguracionih datoteka za sve delove steka. Make

program iz osnovnog direktorijuma Android izvornih kodova pokreće prevođenje celog sistema osim samog Linux jezgra.

Linux kod se prevodi pokretanjem programa make iz osnovnog direktorijuma samog jezgra. Ovo je jedini deo sistema koji zahteva ozbiljnije prilagođenje ciljnoj platformi, jer se ostatak sistema oslanja na jezgro i rukovaoce uređajima unutar njega. Pre samog prevođenja neophodno je definisati sve promenljive u okruženju (ARCH, CC ...), koje koristi make alat kako bi uspešno parametrizovao prevođenje. Za pronalaženje grešaka, prilikom prilagodavanja Linux jezgra, koristi se Loterbah EJTAG debager (Lauterbach EJTAG debugger). Na taj način se postiže lako i brzo učitavanje prevedenog Linux jezgra direktno u RAM memoriju kao i njegovo kontrolisano izvršavanje sa radne stanice.

#### V. PRILAGOĐENJE LINUX JEZGRA

U ovom slučaju, dostupna verzija Linux jezgra za ciljnu platformu je 2.6.25, te je hardverski zavisna kod kao i rukovaoce, koji su pisani za ovu verziju potrebno modifikovati tako da budu u skladu sa novijom verzijom (2.6.27) neophodnom za ispravan rad Android-a.

Prvi korak je dodavanje hardverski zavisnog koda koji implementira podršku za procesor (ako nije već podržan), rukovanje sistemskim pozivima, brojačima i funkcije za razne inicijalizacije specifične za ciljnu platformu. Posredstvom kernel konfiguracionog alata (make config, make menuconfig...) neophodno je za početak ukloniti iz procesa sve rukovaoce koji nisu neophodni za inicijalnu komunikaciju između platforme i Linux jezgra. Tako je logičan izbor za preostale module specifične za platformu rukovalac UART prolazom. Dalji razvoj uključuje postepeno dodavanje i prilagođavanje rukovalaca perifernim uređajima. Rukovaoce se dodaju jedan po jedan i za svaki se obavljaju neophodna prilagođenja novoj verziji jezgra.

Za pokretanje Linux jezgra na ciljnoj platformi, neophodno je razviti osnovni sistem datoteka (root filesystem). Ovo se postiže korišćenjem initramfs sistema u kome se, posredstvom konfiguracione datoteke, definišu putanje do svih direktorijuma, kao i datoteka unutar njih koji zajedno čine osnovni sistem datoteka. Osnovni sistem datoteka poseduje posebne datoteke bez kojih uspešno podizanje Linux jezgra na ciljnoj platformi ne bi bilo moguće. Tipičan primer posebne datoteke je init skripta. Za Android platformu koristi se osnovni sistem datoteka koji se prethodno generiše prevođenjem celog Android steka.

#### VI. PRILAGOĐENJE VIŠIH SLOJEVA ANDROID SOFTVERSKOG STEKA

Pošto je jezgro prilagođeno ciljnom hardveru, može se pristupiti postepenom dodavanju ostalih komponenti u osnovni (root) sistem datoteka.

Prvo se u sistem dodaju biblioteke za rukovanje multimedijom, web-om, bazama podataka, itd. Sa svakom dodatom bibliotekom neophodno je uveriti se u njeno ispravno funkcionisanje ukoliko je to moguće, pozivanjem probnih aplikacija. Hijerarhijski redosled biblioteka mora biti ispoštovan, tj. prvo se dodaju biblioteke koje ne zahtevaju druge biblioteke. Zatim se u sistem dodaje Dalvik virtualna mašina i obavlja njeno modifikovanje-prilagođavanje ciljnoj platformi, ukoliko je to neophodno.

Aplikaciono okruženje i konkretne probne aplikacije moraju poslužiti za proveru same virtualne mašine. Na ovom nivou, problemi sa zadovoljavanjem međuzavisnosti, ukoliko se jave, spadaju u gornje nivoe Android steka koji su vrlo dobro dokumentovani i ispitani. Sa pokretanjem nekih univerzalnih probnih aplikacija, specifičnih za konkretan hardver, proces prilagođenja sistema novoj platformi se može završiti.

#### VII. ZAKLJUČAK

Ovaj rad prikazuje jedan mogući pristup prilagođenju operativnog sistema Android ciljnoj platformi za razvoj televizijskog softvera zasnovanog na MIPS familiji procesora. Sam proces prilagođenja se većinom svodi na prilagođavanje Linux jezgra i rukovalaca uređajima.

Dalji razvoj bi mogao teći u pravcu osposobljavanja Android operativnog sistema za prikaz tokova televizijskog signala, kao i razvoja aplikativnog softvera specifičnog za televizijske uređaje.

#### LITERATURA

- [1] Wikipedia – Android (Operating System), [http://en.wikipedia.org/wiki/Android\\_%28operating\\_system%29](http://en.wikipedia.org/wiki/Android_%28operating_system%29)
- [2] Android on MIPS, <http://www.mipsandroid.com>
- [3] Linux porting guide, <http://www.embedded.com/story/OEG20010221S0092>
- [4] Daniel P. Bovet, Marco Cesati, "Understanding the Linux Kernel"
- [5] Jonathan Corbet, Alessandro Rubini, and Greg Kroah-Hartman, "Linux Device Drivers"

#### ABSTRACT

This paper analyzes one approach to porting the Android Operating System by Google Inc. to a MIPS based, HDMI capable television board.

#### A SOLUTION OF PORTING THE ANDROID OPERATING SYSTEM TO A TV PLATFORM

Tomislav Maruna, Stanko Novaković, Svetozar Miučin, Milan Savić, Nikola Kuzmanović, Goran Rađenović