

FreeBSD support for CubieBoard

Summary

Many open source boards have been developed out of which only some have the full support of the freeBSD kernel. This project aims to provide support for some peripherals which are currently not supported by the freeBSD kernel for CubieBoard which is based on AllWinner A10/A20 SoC

Current Support Status

Drivers

As per the information available on the freebsd-arm mailing lists, freebsd-arm wiki page (<https://www.freebsd.org/platforms/arm.html>) and the freeBSD CubieBoard page.

(<https://wiki.freebsd.org/FreeBSD/arm/Cubieboard>)

The following drivers are currently used by the kernel :-

- 1) GPIO driver
- 2) glue driver for USB EHCI
- 3) Ethernet media access control (EMAC) drivers.

Following drivers are written but not stable yet (rigorous testing remains) :-

- 1) SD Controller driver

All the other drivers have not been written yet.

Boot Process

Due, to the fact that the kernel doesn't have a working driver for SD controller the boot procedure for the kernel is rather clumsy if use a micro SD card to boot our OS. In that case, we need to use both a MicroSD card which

contains the boot loader and the kernel and a USB stick which contains the root file system.

The Project

The project aims to provide/test the remaining drivers for CubieBoard which include the following , the required timeline for the same will be mentioned below.

- 1) SATA controller
- 2) SD Controller
- 3) Sound Subsystem

Once the SD controller driver is written, it is possible to boot without the help of the USB stick also which will simplify the boot process.

Testing details

Testing of drivers is as important as creating new drivers. Therefore, I plan on using the below mentioned three methods to test CubieBoard's drivers.

- 1) Static analysis tools.
- 2) Actual running the driver on CubieBoard.
- 3) Using SymDrive
(<https://www.usenix.org/system/files/conference/osdi12/osdi12-final-4.pdf>)

Project Timeline and Deliverables

MidTerm

- SD controller driver testing/rewriting.
- Boot process correction.

Final Submission

- SATA controller driver
- Sound Subsystem driver

If time permits then , we can move on to other peripherals

Week No	Target
1-3	1) Testing of the current SD controller driver and making changes if necessary.
3-5	2) Using the SD controller driver made above, correct the boot process of freeBSD kernel
5-9	3) Writing of driver for SATA controller.
9-12	4) Writing of drivers for Sound subsystem

About Me

Full Name :- Pratik Singhal

Nick :- ps06756

JOS Kernel Repository :- <https://bitbucket.org/ps06756/jos-kernel>

BITS_OS Repository :- https://github.com/ps06756/BITS_OS

I am a computer science undergraduate student currently in 2nd year studying in BITS Pilani- Goa Campus, India .

I have interest in systems programming and have made a mini os for academic purpose (BITS_OS) and now completing the MIT's JOS kernel whose repository's link is given above.