BareMetal Design Doc

Introduction

This is the design specification for the BareMetal Operating System. BareMetal gets its name from being a very low level operating system and giving more control over things to the programmer. The Operating System does not aim to have POSIX compliance, instead it aims to create a completely original design for its kernel. BareMetal is meant not for the General Purpose user, but for advanced programmers who wish to test out their ideas under a stable OS while still having control over everything they need. Device drivers will be available, but will not be required to be used by software to access the hardware.

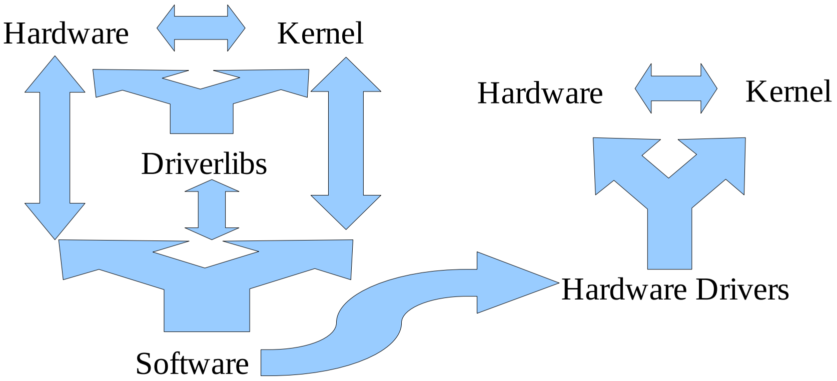
Goals

- A 64 bit OS that people can use

- An Os written completely in FASM assembly
- A stable OS that won't crash
- An OS that gives two options to the programmer low-level or high-level

Running Environment

The Operating System will be organized in a manner as such:



Most software will run in ring 0. The only protection for the software will be provided through the kernel. Software can use the kernels software interrupts, use the driverlib functions, or communicate directly with the hardware. This provides a number of options while maintaining a maximum of control over the computers functions. There will also be a ring 3 option available so that malicious software has less access when you are connected to the internet. The ring 3 software will only be able to use the software interrupts to communicate with the kernel and drivers, direct hardware control will not be allowed, nor will access to the driverlibs. There will be a driver interface for the ring 3 software to use the harware.