Nintendo Switch Operating System

Introduction

The Nintendo Switch is a video game console that you can take anywhere with you. The device itself has the form factor of a tablet, what makes it so fascinating is the components that it leverages to make it unique. The Switch comes packaged with a dock which connects to an HDMI display and is powered by a USB C cable, upon placing the tablet into the dock the Switch’s audio and video output are put on whatever display you are using.

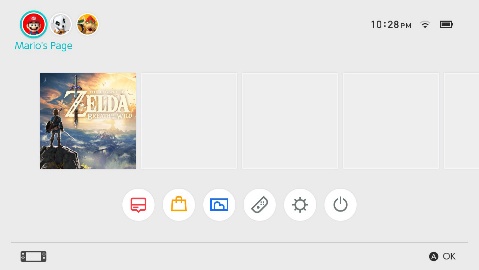
The Switch has two states, handheld mode and docked mode. When the switch is in handheld mode the frames per second are normally capped at 30, and docked mode being capped at 60. So the Operating System has to be ready to “switch” at any time. The switch has a main menu where you can run a specific program or game at a time, you can interrupt your game at any time to go back to the main menu and while the game is still in the background you can open a list of lightweight online applets or photos and settings. The switch also has a small notification system that tells you when your device is low on battery or when you have received an online invitation from someone.

The Nintendo Switch Operating System is given the codename “Horizon” and is closed source. However, through many exploits many users in the homebrew community have reverse engineered the code well enough where switch owners can apply their own custom firmware causing much of the homebrew userbase to be able to familiarize themselves with the system’s kernel and memory management. Having reverse engineered the architecture of the Nintendo switch allows for fan-made optimizations to their own games and programs, with many users adding cheats and mods we can skillfully benchmark the system to get a deeper look at what is going on.

Architecture

Horizon uses a microkernal

<https://news.ycombinator.com/item?id=17534593>



**User Space**

Application and services

System call interface (Horizon implements Supervisor Call Instruction for this)

Kernel

Inter-process communication (IPC)

Synchronization

Memory Management

More functions can be added or removed

CPU Scheduling

Device Manager

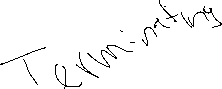
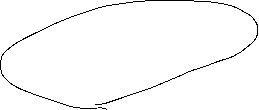
File System

**Kernel Space**



Process states

Suspend



Idle



Running



Terminating

Suspend

Schedule

I/O or events

IDLE

Terminated

Running

NEW

