Building a driver in Linux

J. Alberto Hernandez Espinoza / Dennis Kingston

Guadalajara,Jalisco, México

Abstract

Everyday we use many electronic devices like smart phones, laptops, computers, smart watches, tablets, etc. Everyone have an operating system that is essential the device. One popular operating system is Linux that have different distributions like Ubuntu, inside of every Linux we find the kernel where live all drivers. Driver development is important because they do the connection between the operating system and one device, nowadays many companies develop new devices and is important create all drivers for new devices.

1. Introduction

Linux is an operating system compatible with Unix, and is formed by the kernel and many software like libraries. Many kernels exist for various operating systems. Two main types of kernels exist - monolithic kernels and micro kernels. Monolithic systems have large kernel spaces. For instance, one difference is the placement of device drivers. Monolithic kernels contain drivers (modules) and place them in kernel space while micro kernels lack drivers.

2. Develop

To build a simple driver for Linux, we need to create a c file where have different methods to do the connection with the kernel. First we need to create the method init where the driver going to be registered in memory by a major number and minor number, other method is exit to unregister the driver in the kernel. The next steps are creating the methods open, release, read, and write. To this specific project in open method we read how many chars are introduced by console, in read the chars that were introduced are putting in kernel space from user space, in write method we read the specific char introduced and decide what action going to append.