We created a Device Driver for Linux systems using C code. We downloaded virtual box and the linux iso. After setting up the virtual box, a partition of the hard drive is created for Ubuntu. The iso is then booted into virtual box and linux is now available through virtual box

Open Virtual box and run Ubuntu

Pull up the terminal

Type in su root to log in to the administrator account

Enter password: Foxtrot605!

Type in vi new.c which pulls up linux’s IDE in the terminal which allows you to write, compile and run C code in linux

Type in :q which exits the IDE

Type in make – which compiles the file

Drivers are small programs that enable the kernel to communicate and handle hardware or protocols (rules and standards). Without a driver, the kernel does not know how to communicate with the hardware or handle protocols (the kernel actually hands the commands to the BIOS and the BIOS passes them on the the hardware). The Linux Kernel source code contains many drivers (in the form of source code) in the drivers folder. Each folder within the drivers folder will be explained. When configuring and compiling the kernel, it helps to understand the drivers. Otherwise, a user may add drivers to the kernel that they do not need or leave out important drivers. The driver source code usually includes a commented line that states the purpose of the driver. For example, the source code for the tc driver has a single commented line that says the driver is for TURBOchannel buses. Because of the documentation, users should be able to look at the first few commented lines of future drivers to learn their purpose.  
  
There are different terms that should be understood so that the information below is understandable. An I/O device is an Input/Output device. A modem and network card are examples of this; they send and receive data. A monitor is an output device - information only comes out. A keyboard, mouse, and joystick are input only - data goes into the system. Storage devices store data. Examples of these include SD cards, Hard-drives, CD-roms, memory cards, etc. The CPU (also called a processor) is the "brain" or "heart" of the computer. Without this single processing chip, the computer cannot function. A motherboard (mainboard) is a small board with printed circuits that connect to the components that are on the board. The board and the components are essential to the functionality of a computer. Many computer users say that the motherboard is the heart of the computer (the motherboard holds the CPU). The motherboard contains ports for peripherals. Peripherals include the input, output, and storage devices. A bus is the circuitry of the motherboard and connection to peripherals. Network devices deal with the connection of two or more computers. Ports are devices that users can plug another device or cable into. For example, users would plug a FireWire memory stick into a FireWire port; an Ethernet cable would go into an Ethernet port. Optical discs are read using lasers that read off of reflective surfaces that will either scatter or reflect the laser light. A common optical disk is the DVD. Many systems are 32-bit or 64-bit systems; this refers to the number of bits of registers, address buses, or data buses. For instance, on a 64-bit motherboard, the data buses (the silver lines going between components) have sixty-four lines running parallel to the same destination. Memory addresses are addresses to the memory in the form of bits (ones and zeros). So, a 32-bit memory address contains thirty-two ones and zeros that give the location of a spot on the memory.  
  
Many of the drivers are generic driver meaning that a generic keyboard driver will allow the kernel to handle nearly every keyboard. However, some drivers are specialized. Apple and Commodore, for instance, have made specialized hardware for their Apple computer and Amiga system, respectively. The Linux kernel contains drivers for many devices like Smartphones, Apples, Amiga systems, Sony's Playstation3, Android tablets, and many others.