

Writing Device Drivers, The Easy Way

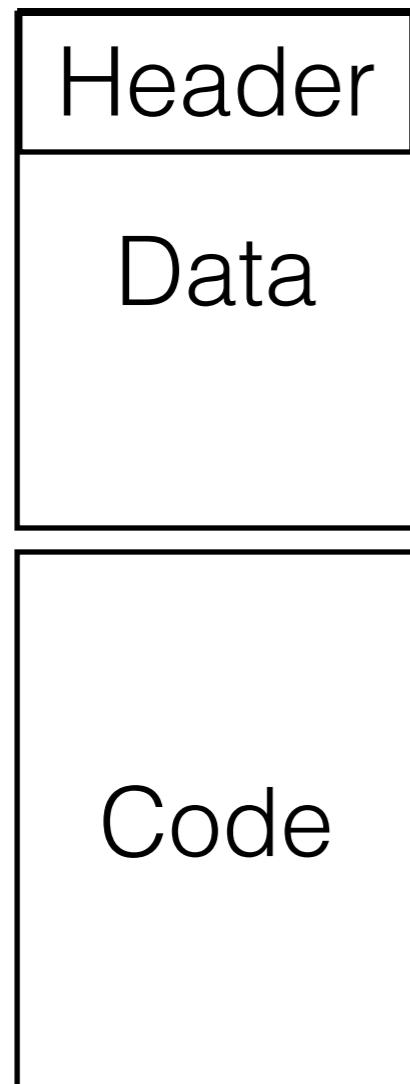
A presentation for Warpstock Europe 2015

By David Azarewicz

David Azarewicz is a founding member of Arca Noae, LLC



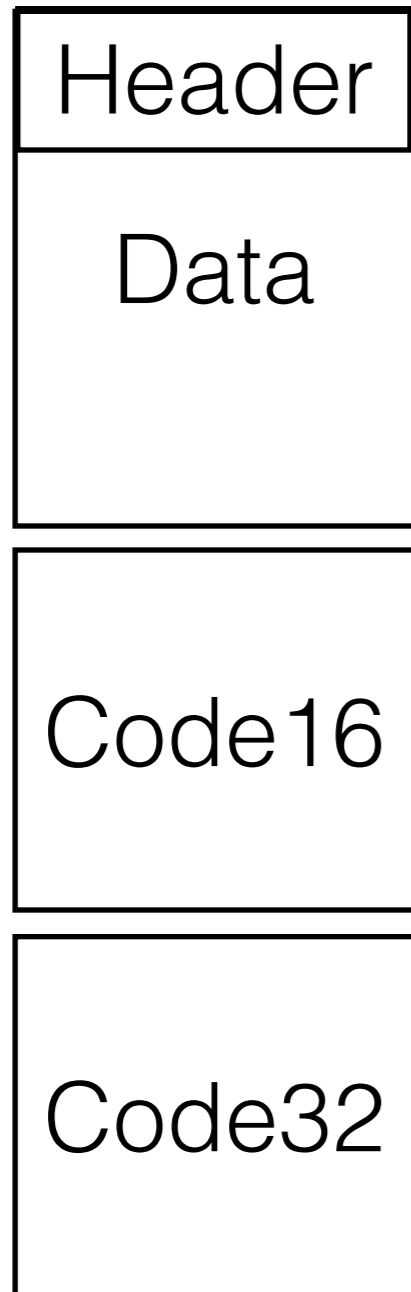
16 Bit Driver Layout



A 16 bit driver has a one code segment and one data segment.

The header is at offset zero in the data segment.

32 Bit Driver Layout



A 32 bit driver has one 16 bit code segment, one 32 bit code segment, and one combined 16/32 bit data segment.

The header is at offset zero in the data segment.

Strategy Handler

- The Strategy Handler is how the system communicates with your driver.
- When your driver is first loaded the kernel calls your driver's Strategy Handler with the INIT function.
- Later, after all the drivers are loaded, the kernel calls your driver's Strategy Handler with the INIT_COMPLETE function.
- Other Strategy Handler functions are OPEN, CLOSE, READ, WRITE, SAVE, RESTORE, SHUTDOWN, etc.

Driver Kits

Drv16

- Can build DEVICE and BASEDEV drivers.
- Very flexible and easy to use.
- No significant restrictions.

Drv32

- Can build DEVICE and BASEDEV drivers.
- Very flexible and easy to use.
- Lots of restrictions

Driver Kits Make It Easy

- Implements all of the memory layout, system interface, and compiler setup.
- Provides memory management functions and hardware interface functions.
- Essentially provides everything needed for a driver except for the stuff that is unique to your specific task.
- All you need to worry about is the code for your own hardware or function.

What Is In The Kit?

Kit Directory

Drv16.lib - A library containing the setup and many functions needed for a device driver.

Dev16lib.h - The header file for the Drv16.lib library.

ndis.h - A header file for NDIS definitions.

pci_regs.h - A header file with PCI register definitions.

Required Directory

Driver.c - A sample of the minimum required elements for building a device driver with the Drv16 kit.

BaseDev.c - A sample of the minimum required elements for building a basedev device driver.

makefile - The makefile for building the sample drivers.

Sample Directory

PciAc5.c - The sample device driver

ioctl.h - The header file for IOCTL communication between the driver and an application.

test.c - A test application for testing the driver's IOCTL interface.

makefile - The makefile for building the sample driver and test application.

Examples In The Kit

Required

- Example of the minimum functions required to make a loadable driver that does nothing.

PciAc5

- A real, working driver for the AC5 port I/O cards.

Live Demonstration

Other Examples

MultiMac drivers

- NVETH and E1000e use Drv16.
- R8169 and E1000B use Drv32

Questions?

David Azarewicz

www.88watts.net

david@88watts.net

David Azarewicz is a member of Arca Noae, LLC

