Steve Tranby  
Created: 9/29/2007  
Last-Modified: 9/29/2007

**README**

1. What is an operating system (OS)?
2. Why would I want to create my own OS?
3. What programming language should I use?
4. <http://www.nondot.org/sabre/os/files/Misc/os-faq/index.html>
5. Information
   1. Necessary to understand the intricacies of an operating system
   2. Compilers, Linkers, File Systems, Executable Formats
6. My Development System and Tools
   1. Virtual Machine
      1. Why?
      2. VMWare Server
      3. QEmu
      4. Bochs
      5. VirtualBox
   2. Assembler
      1. nasm
      2. masm
   3. C / C++ Compiler
      1. gcc and g++
      2. djgpp
   4. Linker
   5. SVN
7. Boot Loader
8. Kernel
9. Memory
   1. Virtual Memory
   2. Segmentation
   3. GDT / LDT
10. Interrupts and Exceptions
    1. Interrupt Service Routines (ISRs)
    2. Interrupt Requests (IRQs)
    3. Programmable Interrupt Controller (PIC)
11. Storage
    1. ATA
       1. PIO
          1. Registers
          2. Status Register
          3. Error Register
          4. Control Register
          5. Commands
          6. Setup
          7. Software Reset
       2. DMA
          1. Setup
       3. Identify Command
    2. ATAPI
    3. SATA
    4. Floppy
    5. USB Flash Drive
    6. USB Mass Storage
12. Managing Resources
    1. Manage Memory
       1. LRU
       2. Swapping
    2. Manage Processes
    3. Manage Threads
    4. Manage File System
13. Input / Output
    1. Console
    2. Keyboard
    3. Mouse
    4. Video
       1. VGA
       2. Vesa
14. Files and File Systems
15. Beyond or MISC
    1. PCI
    2. Plug and Play

**Parts of OS Development (biased toward x86)**

**Developing for x86**<http://en.wikipedia.org/wiki/X86_instruction_listings>

**Develop on Windows (use PE format)**<http://ksrenevasan.blogspot.com/2005/10/writing-multiboot-pe-kernels-using.html>

**Suggestions** <http://www.osdever.net/tutorials/mysuggestions.php>

**Tools**Assembler, Compiler,

**Step By Step Process I went through**

1. Setup development environment.
2. Setup testing Virtual Machine.
   1. I decided to create a virtual machine for my compiling and linking environment.
   2. Write the code on Windows XP using text editor (Notepad++).
   3. FTP over to my ubuntu Server development virtual machine
   4. Copy the final compiled binary onto a floppy image file.
   5. Start Virtual Machine used to test the OS binary from the same floppy image.
3. Initial Tool Chain Test
   1. Create a NASM assembly file we’ll call ‘test.s’
4. Boot Loader
   1. I chose to use a Boot Loader called GRUB instead of “rolling” my own. However there are some great tutorials on creating a boot loader, however most people suggest that a boot loader can be just as tough as building a simple kernel or operating system.
   2. I would say that a boot loader specific to your kernel and architecture is not as difficult as developing the OS kernel, but is still quite hard to do.
5. Use a Multiboot compatible Boot Loader
   1. I used GRUB, it seems to be the most powerful and is easy to use.
6. Create GRUB based floppy image
   1. I used a premade floppy image
7. Create Initial GRUB based kernel – version 0.1.0
   1. Write the basics for the kernel to display a single character of a specified color
8. Write/Copy Kernel Binary onto Floppy Image
   1. I created a simple shell script to do the following
   2. Mount the floppy image to the file system
   3. copy the kernel binary output (kernel.bin) to the floppy image in the folder boot/
   4. un mount the floppy image from the file system
9. Test GRUB based kernel (kernel 0.2)
   1. Load virtual machine with floppy image
   2. Watch it display the single character ‘T’ in white
10. Problems I had
11. Develop 2nd Kernel – version 0.2.0
    1. Output Test
    2. Implement basic GDT,IDT,IRS,IRQ
    3. Keyboard Support
    4. Screen Functions
12. Test Kernel
13. Problems
14. Develop 3rd Kernel – version 0.3.0
    1. Hard Drive Access (ATA – PIO)
15. Test Kernel
    1. VirtualBox
       1. Didn’t seem to like only having a primary drive, and not a slave drive.
16. Problems
17. …

**Information about the areas of OS development**

Boot Loader

Roll Your Own (RYO)

Utilize a Multi-Boot-Loader like GRUB, LILO, etc

Real Mode

Protected Mode

: <http://linuxgazette.net/issue82/raghu.html>

Crash course in pmode: <http://www.geocities.com/SiliconValley/2151/pmode.html>

Virtual Memory

Segmentation

Global Descriptor Table (GDT)

http://en.wikibooks.org/wiki/X86\_Assembly/Global\_Descriptor\_Table

Local Descriptor Table (LDT)

Interrupt Descriptor Table (IDT)

Memory Management

GRUB hands off the size of RAM in a multi-boot header

<http://www.osdev.org/osfaq2/index.php/How%20do%20I%20determine%20the%20amount%20of%20RAM%3F>

Start by writing a flat memory model using the simple GDT of one code segment (4GB) and one data segment (4GB).

<http://en.wikipedia.org/wiki/Memory_manager>

<http://www.osdever.net/tutorials/memory1.php>  
<http://www.osdever.net/tutorials/memory2.php>

# Storage

## ATA Device

ATA Identify Commands

<http://www.win.tue.nl/~aeb/linux/Large-Disk-10.html#identify>  
<https://www.bustrace.com/bustrace6/generate.htm#ata>

On a WDC WD80 ATA device, the Identify data is shown in busTRACE 6.0 as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| **Word(s)** | **Value** | **Bit(s)** | **Description** |
| 0 | 427Ah | 15 | ATA Device = Yes |
|  |  | 17 | Removable Media Device = No |
|  |  | 2 | Response Incomplete = No |
| 1 | 3FFFh | - | (Obsolete) # of Logical Cylinders = 3FFFh |
| 2 | 0000h | - | Specific Configuration = None |
| 3 | 0010h | - | (Obsolete) # of Logical Heads = 10h |
| 6 | 003Fh | - | (Obsolete) # of Sectors per Track = 3Fh |
| 10-19 |  | - | Serial Number = WD-WMA8E3265410 |
| 23-26 |  | - | Firmware Revision = 16.06V1 |
| 27-46 |  | - | Model Number = WDC WD800BB-75CAA0 |
| 47 | 8010h | 7-0 | Max # of sectors xfered per interrupt on R/W MULTIPLE cmds = 10h |
| 48 | 0000h | - | Trusted Computing feature set options = 0h |
| 53 | 0007h | 2 | Fields reported in word 88 valid = Yes |
|  |  | 1 | Fields reported in words 70:64 valid = Yes |
| 59 | 0110h | 8 | Multiple sector setting is valid = Yes |
| 60-61 |  | - | Total number of user addressable sectors = 9502F90h (74.51 Gbytes) |
| 63 | 0007h | - | \*\*\* MULTIWORD DMA SETTINGS \*\*\* |
|  |  | 10 | Multiword DMA Mode 2 selected = No |
|  |  | 9 | Multiword DMA Mode 1 selected = No |
|  |  | 8 | Multiword DMA Mode 0 selected = No |
|  |  | 2 | Multiword DMA Mode 2 and below supported = Yes |
|  |  | 1 | Multiword DMA Mode 1 and below supported = Yes |
|  |  | 0 | Multiword DMA Mode 0 supported = Yes |
| 64 | 0003h | 7-0 | PIO Modes Supported = 3h |
| 65 | 0078h | - | Minimum Multiword DMA xfer cycle time per word in nsecs = 120 |
| 66 | 0078h | - | Recommended Multiword DMA xfer cycle time in nsecs = 120 |
| 67 | 0078h | - | Minimum PIO xfer cycle time without flow control in nsecs = 120 |
| 68 | 0078h | - | Minimum PIO xfer cycle time with IORDY flow control in nsecs = 120 |
| 82 | 3469h | - | \*\*\* COMMAND SETS SUPPORTED \*\*\* |
|  |  | 14 | NOP command supported = No |
|  |  | 13 | READ BUFFER command supported = Yes |
|  |  | 12 | WRITE BUFFER command supported = Yes |
|  |  | 10 | Host Protected Area feature set supported = Yes |
|  |  | 9 | DEVICE RESET command supported = No |
|  |  | 8 | SERVICE interrupt supported = No |
|  |  | 7 | Release interrupt supported = No |
|  |  | 6 | Look-ahead supported = Yes |
|  |  | 5 | Write cache supported = Yes |
|  |  | 3 | Mandatory Power Management feature set supported = Yes |
|  |  | 2 | Removable Media feature set supported = No |
|  |  | 1 | Security Mode feature set supported = No |
|  |  | 0 | SMART feature set supported = Yes |
| 83h | 4B01h | - | \*\*\* COMMAND SETS SUPPORTED \*\*\* |
|  |  | 13 | FLUSH CACHE EXT command supported = No |
|  |  | 12 | Mandatory FLUSH CACHE command supported = No |
|  |  | 11 | Device Configuration Overlay feature set supported = Yes |
|  |  | 10 | 48-bit Address feature set supported = No |
|  |  | 9 | Automatic Acoustic Management feature set supported = Yes |
|  |  | 8 | SET MAX security extension supported = Yes |
|  |  | 6 | SET FEATURES subcommand required to spinup after power-up = No |
|  |  | 5 | Power-Up In Standby feature set supported = No |
|  |  | 4 | Removable Media Status Notification feature set supported = No |
|  |  | 3 | Advanced Power Management feature set supported = No |
|  |  | 2 | CFA feature set supported = No |
|  |  | 1 | READ/WRITE DMA QUEUED supported = No |
|  |  | 0 | DOWNLOAD MICROCODE command supported = Yes |
| 84 | 4000h | - | \*\*\* COMMAND SET/FEATURE SUPPORTED EXTENSION \*\*\* |
|  |  | 12 | Time-limited R/W feature set R/W continuous enabled = No |
|  |  | 11 | Time-limited R/W feature set supported = No |
|  |  | 10 | URG bit supported for WRITE STREAM DMA/PIO = No |
|  |  | 9 | URG bit supported for READ STREAM DMA/PIO = No |
|  |  | 8 | World wide name supported = No |
|  |  | 7 | WRITE DMA QUEUED FUA EXT command supported = No |
|  |  | 6 | WRITE DMA FUA EXT and WRITE MULTIPLE FUA EXT supported = No |
|  |  | 5 | General Purpose Logging feature set supported = No |
|  |  | 4 | Streaming feature set supported = No |
|  |  | 3 | Media Card Pass Through Command feature set supported = No |
|  |  | 2 | Media serial number supported = No |
|  |  | 1 | SMART self-test supported = No |
|  |  | 0 | SMART error logging supported = No |
| 85 | 3469h | - | \*\*\* COMMAND SET/FEATURE ENABLED \*\*\* |
|  |  | 14 | NOP command enabled = No |
|  |  | 13 | READ BUFFER command enabled = Yes |
|  |  | 12 | WRITE BUFFER command enabled = Yes |
|  |  | 10 | Host Protected Area feature set enabled = Yes |
|  |  | 9 | DEVICE RESET command enabled = No |
|  |  | 8 | SERVICE interrupt enabled = No |
|  |  | 7 | Release interrupt enabled = No |
|  |  | 6 | Look-ahead enabled = Yes |
|  |  | 5 | Write cache enabled = Yes |
|  |  | 3 | Power Management feature set enabled = Yes |
|  |  | 2 | Removable Media feature set enabled = No |
|  |  | 1 | Security Mode feature set enabled = No |
|  |  | 0 | SMART feature set enabled = Yes |
| 86 | 0A01h | - | \*\*\* COMMAND SET/FEATURE ENABLED \*\*\* |
|  |  | 13 | FLUSH CACHE EXT command supported = No |
|  |  | 12 | FLUSH CACHE command supported = No |
|  |  | 11 | Device Configuration Overlay supported = Yes |
|  |  | 10 | 48-bit Address features set supported = No |
|  |  | 9 | Automatic Acoustic Management feature set enabled = Yes |
|  |  | 8 | SET MAX security ext enabled by SET MAX SET PASSWORD = No |
|  |  | 6 | SET FEATURES subcommand required to spin-up after power-up = No |
|  |  | 5 | Power-Up In Standby feature set enabled = No |
|  |  | 4 | Removable Media Status Notification feature set enabled = No |
|  |  | 3 | Advanced Power Management feature set enabled = No |
|  |  | 2 | CFA feature set enabled = No |
|  |  | 1 | READ/WRITE DMA QUEUED command supported = No |
|  |  | 0 | DOWNLOAD MICROCODE command supported = Yes |
| 87 | 4000h | - | \*\*\* COMMAND SET/FEATURE DEFAULT \*\*\* |
|  |  | 12 | Time-limited R/W feature set R/W continuous enabled = No |
|  |  | 11 | Time-limited R/W feature set enabled = No |
|  |  | 10 | URG bit supported for WRITE STREAM DMA/PIO = No |
|  |  | 9 | URG bit supported for READ STREAM DMA/PIO = No |
|  |  | 8 | World wide name supported = No |
|  |  | 7 | WRITE DMA QUEUED FUA EXT command supported = No |
|  |  | 6 | WRITE DMA FUA EXT and WRITE MULTIPLE FUA EXT supported = No |
|  |  | 5 | General Purpose Logging feature set supported = No |
|  |  | 4 | Valid CONFIGURE STREAM command has been executed = No |
|  |  | 3 | Media Card Pass Through Command feature set enabled = No |
|  |  | 2 | Media serial number is valid = No |
|  |  | 1 | SMART self-test supported = No |
|  |  | 0 | SMART error logging supported = No |
| 88 | 203Fh | - | \*\*\* ULTRA DMA SETTINGS \*\*\* |
|  |  | 14 | Ultra DMA Mode 6 Selected = Yes |
|  |  | 13 | Ultra DMA Mode 5 Selected = Yes |
|  |  | 12 | Ultra DMA Mode 4 Selected = No |
|  |  | 11 | Ultra DMA Mode 3 Selected = No |
|  |  | 10 | Ultra DMA Mode 2 Selected = No |
|  |  | 9 | Ultra DMA Mode 1 Selected = No |
|  |  | 8 | Ultra DMA Mode 0 Selected = No |
|  |  | 6 | Ultra DMA Mode 6 and below supported = No |
|  |  | 5 | Ultra DMA Mode 5 and below supported = Yes |
|  |  | 4 | Ultra DMA Mode 4 and below supported = Yes |
|  |  | 3 | Ultra DMA Mode 3 and below supported = Yes |
|  |  | 2 | Ultra DMA Mode 2 and below supported = Yes |
|  |  | 1 | Ultra DMA Mode 1 and below supported = Yes |
|  |  | 0 | Ultra DMA Mode 0 and below supported = Yes |
| 94 | 8080h | 15-8 | Vendor's recommended acoustic management value = Minimum acoustic emanation level (80h) |
|  |  | 7-0 | Current automatic acoustic management value = Minimum acoustic emanation level (80h) |
| 95 | 0000h | - | Stream Minimum Request Size = 0h |
| 96 | 0000h | -- | Streaming Transfer Time - DMA = 0h |
| 97 | 0000h | - | Streaming Access Latency - DMA and PIO = 0h |
| 98-99 |  | - | Streaming Performance Granularity = 0h |
| 100-103 |  | - | Maximum user LBA for 48-bit address = 0000000000000000h |
| 104 | 0000h | - | Streaming Transfer Time - PIO = 0h |
| 127 | 0000h | 1-0 | Removable Media Status Notification = Not supported |
| 176-205 |  | - | Current media serial number = |
| 255 | 4FA5h | 15-8 | Checksum = 4Fh |

The raw hex IDENTIFY data is also available within busTRACE 6.0. Using our above example, the return hex data is:

|  |  |
| --- | --- |
|  | 00000000 - 427A 3FFF 0000 0010 E100 0258 003F 0010 - zB.?......X.?... 00000010 - 0000 000E 5744 2D57 4D41 3845 3338 3635 - ....DWW-AME88356 00000020 - 3734 3200 0000 0000 0003 1000 0028 3136 - 47.2........(.61 00000030 - 2E30 3656 3136 5744 4320 5744 3830 3042 - 0.V661DW CDW08B0 00000040 - 422D 3735 4341 4130 2020 2020 2020 2020 - -B57AC0A  00000050 - 2020 2020 2020 2020 2020 2020 2020 8010 - .. 00000060 - 0000 2F00 4001 0280 0000 0007 3FFF 0010 - .../.@.......?.. 00000070 - 003F FC10 00FB 0110 2F90 0950 0000 0007 - ?......../P..... 00000080 - 0003 0078 0078 0078 0078 0000 0000 0000 - ..x.x.x.x....... 00000090 - 0000 0000 0000 0000 0000 0000 0000 0000 - ................ 000000A0 - 003E 0000 3469 4B01 4000 3469 0A01 4000 - >...i4.K.@i4...@ 000000B0 - 203F 0000 0000 0000 0000 604D 8080 0000 - ? ........M`.... 000000C0 - 0000 0000 0000 0000 0000 0000 0000 0000 - ................ 000000D0 - 0000 0000 0000 0000 0000 0000 0000 0000 - ................ 000000E0 - 0000 0000 0000 0000 0000 0000 0000 0000 - ................ 000000F0 - 0000 0000 0000 0000 0000 0000 0000 0000 - ................ 00000100 - 0000 0000 0000 0000 0000 002D 0000 0000 - ..........-..... 00000110 - 0000 0000 0000 0000 0000 0000 0000 0001 - ................ 00000120 - 0000 0000 0000 0000 0000 0000 0000 0000 - ................ 00000130 - 0000 0000 0000 0000 0000 0000 0000 001E - ................ 00000140 - 0000 0000 0000 0000 0000 0000 0000 0000 - ................ 00000150 - 0000 0000 0000 0000 0000 0000 0000 0000 - ................ 00000160 - 0000 0000 0000 0000 0000 0000 0000 0000 - ................ 00000170 - 0000 0000 0000 0000 0000 0000 0000 0000 - ................ 00000180 - 0000 0000 0000 0000 0000 0000 0000 0000 - ................ 00000190 - 0000 0000 0000 0000 0000 0000 0000 0000 - ................ 000001A0 - 0000 0000 0000 0000 0000 0000 0000 0000 - ................ 000001B0 - 0000 0000 0000 0000 0000 0000 0000 0000 - ................ 000001C0 - 0000 0000 0000 0000 0000 0000 0000 0000 - ................ 000001D0 - 0000 0000 0000 0000 0000 0000 0000 0000 - ................ 000001E0 - 0000 0000 0000 0000 0000 0000 0000 0000 - ................ 000001F0 - 0000 0000 0000 0000 0000 0000 0000 4FA5 - ...............O |

Multitasking

http://www.osdever.net/tutorials/multitasking.php

Video

Standard VGA address: 0xb8000 (using standard segmenting 0xb800:0)

Real Mode Interrupt Functions

My walkthrough building Tranix!

Tips/Problems

=============

Prob:

undefined reference to '\_\_stack\_chk\_fail'

Desc:

This problem occurs when the gcc compiler thinks you have a possible buffer

overflow, or the code cannot be statically proven to be safe to such a bug.

Soln:

add -fno-stack-protector to the CFLAGS as an argument to gcc

<http://hackinglinux.blogspot.com/2006/11/resolving-stackchkfail-error.html>

-------------

Prob:

Desc:

Soln: