Build own operating system

Second stage :

Build kernel :

\* What is the kernel ?

**Kernel** is the core part of an operating system which manages system resources. It also acts like a bridge between application and hardware of the computer. It is one of the first programs loaded on start-up (after the Bootloader).

**And its the heart of the operating system.**

\* Resources

# ( What is a Kernel in OS? What are the types of Kernel? )

<https://www.thewindowsclub.com/what-is-a-kernel-in-os-what-are-the-types-of-kernel>

# ( Microkernel in Operating Systems )

<https://www.geeksforgeeks.org/microkernel-in-operating-systems/>

( How to Write a Simple OS Kernel )

<https://www.youtube.com/watch?v=wmR-_rxWvYc>

# (How-to customize compile and install the Linux Kernel on Arch

# Linux [HD )

# <https://www.youtube.com/watch?v=nsTc45wRaIM&t=193s>

# (Yocto Linux #1 - Hello World Kernel Module)

<https://www.youtube.com/watch?v=U5G_rT-GwPE&t=196s>

# ( How to compile the Linux kernel from source)

<https://www.youtube.com/watch?v=plfls9MqAHM>

# (Building a Linux Kernel )

<https://www.youtube.com/watch?v=NrEz2XuP57E>

# ( Creating a first hello world module for Linux kernel | Linux kernel programming )

<https://www.youtube.com/watch?v=ZoYn2UcErrs>

Tutorial] Building Your First Kernel

<https://forum.xda-developers.com/showthread.php?t=1748297>

Source approved for implementation:

# Create Your Own Operating System (OS)

# - Kernel

<https://www.youtube.com/watch?v=4hJDOvwbTZs&t=2s>

Tools :

GNU/Linux :- using GNU/Kali Linux 2017 i386 distribution

. Assembler :- using GNU Assembler(gas) to instruct the bootloader for loading the starting point of our kernel

. GCC :- GNU Compiler Collection a cross compiler.

. Xorriso :- A package that creates, loads, manipulates ISO 9660 filesystem images.(man xorriso)

grub-mkrescue :- Make a GRUB rescue image, this package internally calls the xorriso functionality to build an iso image.

QEMU :- Quick EMUlator to boot our kernel in virtual machine without rebooting the main system