linux/arch

- Subdirectories for each current port.
- Each contains **kernel**, **lib**, **mm**, **boot** and other directories whose contents override code stubs in architecture independent code.
- **lib** contains highly-optimized common utility routines such as memcpy, checksums, etc.
- **arch** as of 2.4:
  - alpha, arm, i386, ia64, m68k, mips, mips64.
  - ppc, s390, sh, sparc, sparc64.

```
quan@quan-ubuntu:~/linux-4.9.10/arch/x86$ ls
Kbuild          Makefile         configs  ia32     lguest    net       power      tools
Kconfig         Makefile.um      crypto   include  lib       oprofile  purgatory  um
Kconfig.cpu     Makefile_32.cpu  entry    kernel   math-emu  pci       ras        video
Kconfig.debug   boot             events   kvm      mm        platform  realmode   xen
```

```
quan@quan-ubuntu:~/linux-4.9.10/drivers$ ls
Kconfig         cpufreq     hsi         mailbox     oprofile    remoteproc  tty
Makefile        cpuidle     hv          mcb         parisc      reset       uio
accessibility   crypto      hwmon       md          parport     rpmsg       usb
acpi            dax         hwspinlock  media       pci         rtc         uwb
amba            dca         hwtracing   memory      pcmcia      s390        vfio
android         devfreq     i2c         memstick    perf        sbus        vhost
ata             dio         ide         message     phy         scsi        video
atm             dma         idle        mfd         pinctrl     sfi         virt
auxdisplay      dma-buf     iio         misc        platform    sh          virtio
base            edac        infiniband  mmc         pnp         sn          vlynq
bcma            eisa        input       mtd         power       soc         vme
block           extcon      iommu       net         powercap    spi         w1
bluetooth       firewire    ipack       nfc         pps         spmi        watchdog
bus             firmware    irqchip     ntb         ps3         ssb         xen
cdrom           fmc         isdn        nubus       ptp         staging     zorro
char            fpga        leds        nvdimm      pwm         target
clk             gpio        lguest      nvme        rapidio     tc
clocksource     gpu         lightnvm    nvmem       ras         thermal
connector       hid         macintosh   of          regulator   thunderbolt
```

linux/drivers

- **Largest amount of code in the kernel tree (~1.5M).**
- device, bus, platform and general directories.
- drivers/char – n_tty.c is the default line discipline.
- drivers/block – elevator.c, genhd.c, linear.c, ll_rw_blk.c, raidN.c.
- drivers/net –specific drivers and general routines Space.c and net_init.c.
- drivers/scsi – scsi_*.c files are generic; sd.c (disk), sr.c (CD-ROM), st.c (tape), sg.c (generic).
- General:
  - cdrom, ide, isdn, parport, pcmcia, pnp, sound, telephony, video.
- Buses – fc4, i2c, nubus, pci, sbus, tc, usb.
- Platforms – acorn, macintosh, s390, sgi.

linux/fs

```
quan@quan-ubuntu:~/linux-4.9.10/fs$ ls
9p                  compat.c              file.c              minix               qnx6
Kconfig             compat_binfmt_elf.c   file_table.c        mount.h             quota
Kconfig.binfmt      compat_ioctl.c        filesystems.c       mpage.c             ramfs
Makefile            configfs              freevxfs            namei.c             read_write.c
adfs                coredump.c            fs-writeback.c      namespace.c         readdir.c
affs                cramfs                fs_pin.c            ncpfs               reiserfs
afs                 crypto                fs_struct.c         nfs                 romfs
aio.c               dax.c                 fscache             nfs_common          select.c
anon_inodes.c       dcache.c              fuse                nfsd                seq_file.c
attr.c              dcookies.c            gfs2                nilfs2              signalfd.c
autofs4             debugfs               hfs                 nls                 splice.c
bad_inode.c         devpts                hfsplus             no-block.c          squashfs
befs                direct-io.c           hostfs              notify              stack.c
bfs                 dlm                   hpfs                nsfs.c              stat.c
binfmt_aout.c       drop_caches.c         hugetlbfs           ntfs                statfs.c
binfmt_elf.c        ecryptfs              inode.c             ocfs2               super.c
binfmt_elf_fdpic.c  efivarfs              internal.h          omfs                sync.c
binfmt_em86.c       efs                   ioctl.c             open.c              sysfs
binfmt_flat.c       eventfd.c             iomap.c             openpromfs          sysv
binfmt_misc.c       eventpoll.c           isofs               orangefs            timerfd.c
binfmt_script.c     exec.c                jbd2                overlayfs           tracefs
block_dev.c         exofs                 jffs2               pipe.c              ubifs
btrfs               exportfs              jfs                 pnode.c             udf
buffer.c            ext2                  kernfs              pnode.h             ufs
cachefiles          ext4                  libfs.c             posix_acl.c         userfaultfd.c
ceph                f2fs                  lockd               proc                utimes.c
char_dev.c          fat                   locks.c             proc_namespace.c    xattr.c
cifs                fcntl.c               logfs               pstore              xfs
coda                fhandle.c             mbcache.c           qnx4
```

- Contains:
  - virtual filesystem (VFS) framework.
  - subdirectories for actual filesystems.
- vfs-related files:
  - exec.c, binfmt_*.c - files for mapping new process images.
  - devices.c, blk_dev.c – device registration, block device support.
  - super.c, filesystems.c.
  - inode.c, dcache.c, namei.c, buffer.c, file_table.c.
  - open.c, read_write.c, select.c, pipe.c, fifo.c.
  - fcntl.c, ioctl.c, locks.c, dquot.c, stat.c.

linux/include

- include/asm-*:
  - Architecture-dependent include subdirectories.
- include/linux:
  - Header info needed both by the kernel and user apps.
  - Usually linked to /usr/include/linux.
  - Kernel-only portions guarded by #ifdefs
    - #ifdef __KERNEL__
    - /* kernel stuff */
    - #endif
- Other directories:
  - math-emu, net, pcmcia, scsi, video, …

```
quan@quan-ubuntu:~/linux-4.9.10/include$ ls
Kbuild       clocksource  dt-bindings  linux     memory  pcmcia  rxrpc  sound   uapi
acpi         crypto       keys         math-emu  misc    ras     scsi   target  video
asm-generic  drm          kvm          media     net     rdma    soc    trace   xen
```

linux/init

- Just two files: version.c, main.c.
- version.c – contains the version banner that prints at boot.
- main.c – architecture-independent boot code.
- start_kernel (in main.c) is the primary entry point.

More files in the latest version

```
quan@quan-ubuntu:~/linux-4.9.10/init$ ls
Kconfig      do_mounts.c         do_mounts_md.c  initramfs.c    version.c
Makefile     do_mounts.h         do_mounts_rd.c  main.c
calibrate.c  do_mounts_initrd.c  init_task.c     noinitramfs.c
```

linux/ipc

- System V IPC facilities.
- If disabled at compile-time, util.c exports stubs that simply return –ENOSYS.
- One file for each facility:
  - sem.c – semaphores.
  - shm.c – shared memory.
  - msg.c – message queues.

```
quan@quan-ubuntu:~/linux-4.9.10/ipc$ ls
Makefile  compat_mq.c   mq_sysctl.c  msg.c      namespace.c  shm.c      util.c
compat.c  ipc_sysctl.c  mqueue.c     msgutil.c  sem.c        syscall.c  util.h
```

IPC (Inter-Procedure Communication，进程间通信)

linux/kernel

```
quan@quan-ubuntu:~/linux-4.9.10/kernel$ ls
Kconfig.freezer     cred.c              kthread.c           smpboot.c
Kconfig.hz          debug               latencytop.c        smpboot.h
Kconfig.locks       delayacct.c         livepatch           softirq.c
Kconfig.preempt     dma.c               locking             stacktrace.c
Makefile            elfcore.c           membarrier.c        stop_machine.c
acct.c              events              memremap.c          sys.c
async.c             exec_domain.c       module-internal.h   sys_ni.c
audit.c             exit.c              module.c            sysctl.c
audit.h             extable.c           module_signing.c    sysctl_binary.c
audit_fsnotify.c    fork.c              notifier.c          task_work.c
audit_tree.c        freezer.c           nsproxy.c           taskstats.c
audit_watch.c       futex.c             padata.c            test_kprobes.c
auditfilter.c       futex_compat.c      panic.c             time
auditsc.c           gcov                params.c            torture.c
backtracetest.c     groups.c            pid.c               trace
bounds.c            hung_task.c         pid_namespace.c     tracepoint.c
bpf                 irq                 power               tsacct.c
capability.c        irq_work.c          printk              ucount.c
cgroup.c            jump_label.c        profile.c           uid16.c
cgroup_freezer.c    kallsyms.c          ptrace.c            up.c
cgroup_pids.c       kcmp.c              range.c             user-return-notifier.c
compat.c            kcov.c              rcu                 user.c
configs             kexec.c             reboot.c            user_namespace.c
configs.c           kexec_core.c        relay.c             utsname.c
context_tracking.c  kexec_file.c        resource.c          utsname_sysctl.c
cpu.c               kexec_internal.h    sched               watchdog.c
cpu_pm.c            kmod.c              seccomp.c           workqueue.c
cpuset.c            kprobes.c           signal.c            workqueue_internal.h
crash_dump.c        ksysfs.c            smp.c
```

- The core kernel code.
- sched.c – “the main kernel file”:
  - scheduler, wait queues, timers, alarms, task queues.
- Process control:
  - fork.c, exec.c, signal.c, exit.c etc…
- Kernel module support:
  - kmod.c, ksyms.c, module.c.
- Other operations:
  - time.c, resource.c, dma.c, softirq.c, itimer.c.
  - printk.c, info.c, panic.c, sysctl.c, sys.c.

上海交通大學

```
quan@quan-ubuntu:~/linux-4.9.10/lib$ ls
842                             kstrtox.h
Kconfig                         lcm.c
Kconfig.debug                   libcrc32c.c
Kconfig.kasan                   list_debug.c
Kconfig.kgdb                    list_sort.c
Kconfig.kmemcheck               llist.c
Kconfig.ubsan                   locking-selftest-hardirq.h
Makefile                        locking-selftest-mutex.h
argv_split.c                    locking-selftest-rlock-hardirq.h
asn1_decoder.c                  locking-selftest-rlock-softirq.h
assoc_array.c                   locking-selftest-rlock.h
atomic64.c                      locking-selftest-rsem.h
atomic64_test.c                 locking-selftest-softirq.h
audit.c                         locking-selftest-spin-hardirq.h
bcd.c                           locking-selftest-spin-softirq.h
```

linux/lib

- kernel code cannot call standard C library routines.
- Files:
  - brlock.c – “Big Reader” spinlocks.
  - cmdline.c – kernel command line parsing routines.
  - errno.c – global definition of errno.
  - inflate.c – “gunzip” part of gzip.c used during boot.
  - string.c – portable string code.
    - Usually replaced by optimized, architecture-dependent routines.
  - vsprintf.c – libc replacement.

```
quan@quan-ubuntu:~/linux-4.9.10/mm$ ls
Kconfig                 internal.h          mprotect.c              slab_common.c
Kconfig.debug           interval_tree.c     mremap.c                slob.c
Makefile                kasan               msync.c                 slub.c
backing-dev.c           khugepaged.c        nobootmem.c             sparse-vmemmap.c
balloon_compaction.c    kmemcheck.c         nommu.c                 sparse.c
bootmem.c               kmemleak-test.c     oom_kill.c              swap.c
cleancache.c            kmemleak.c          page-writeback.c        swap_cgroup.c
cma.c                   ksm.c               page_alloc.c            swap_state.c
cma.h                   list_lru.c          page_counter.c          swapfile.c
cma_debug.c             maccess.c           page_ext.c              truncate.c
compaction.c            madvise.c           page_idle.c             usercopy.c
debug.c                 memblock.c          page_io.c               userfaultfd.c
debug_page_ref.c        memcontrol.c        page_isolation.c        util.c
dmapool.c               memory-failure.c    page_owner.c            vmacache.c
early_ioremap.c         memory.c            page_poison.c           vmalloc.c
fadvise.c               memory_hotplug.c    pagewalk.c              vmpressure.c
failslab.c              mempolicy.c         percpu-km.c             vmscan.c
```

linux/mm

- Paging and swapping:
  - swap.c, swapfile.c (paging devices), swap_state.c (cache).
  - vmscan.c – paging policies, kswapd.
  - page_io.c – low-level page transfer.
- Allocation and deallocation:
  - slab.c – slab allocator.
  - page_alloc.c – page-based allocator.
  - vmalloc.c – kernel virtual-memory allocator.
- Memory mapping:
  - memory.c – paging, fault-handling, page table code.
  - filemap.c – file mapping.
  - mmap.c, mremap.c, mlock.c, mprotect.c.

linux/scripts

- Scripts for:
  - Menu-based kernel configuration.
  - Kernel patching.
  - Generating kernel documentation.

```
quan@quan-ubuntu:~/linux-4.9.10/scripts$ ls
Kbuild.include          config                      kernel-doc-xml-ref
Lindent                 conmakehash.c               ksymoops
Makefile                const_structs.checkpatch    ld-version.sh
Makefile.asm-generic    decode_stacktrace.sh        link-vmlinux.sh
Makefile.build          decodecode                  makelst
Makefile.clean          depmod.sh                   markup_oops.pl
Makefile.dtbinst        diffconfig                  mkcompile_h
Makefile.extrawarn      docproc.c                   mkmakefile
Makefile.fwinst         dtc                         mksysmap
Makefile.gcc-plugins    export_report.pl            mkuboot.sh
```

上海交通大學