About Daxinlang

Company Profile

Qualification honor

Talent area

news

Industry Information
company news

Free sample application

contact us

Industry Information

Ingenic T30 development guide, Ingenic original factory developmen

Source: Yijie Chip Agent Author: Xiaoxin Release time: 2020-04-21 10:42:43

Ingenic T30 development guide, Ingenic original factory development materials, Yijie editor con introduce. ISVP SDK, the software development kit, including API library, open source source code Samples, etc. Developers can quickly develop product functions through the SDK.

The following is an overview of the contents of the ISVP SDK:

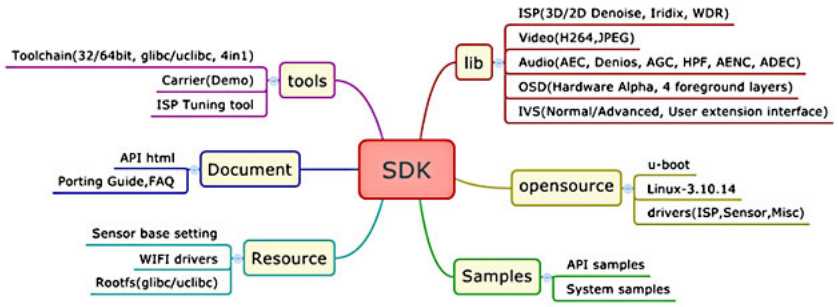

图 1-1 SDK 组成结构

SDK hierarchy

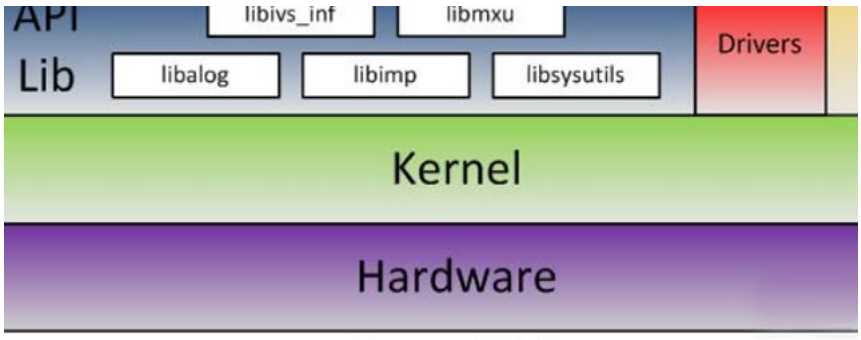

图 1-2 SDK 层次机构

● Hardware: The hardware layer, completes specific hardware functions such as I/O.
● linux kernel: the kernel layer. Complete basic system functions and define hardware resources
● drivers: ko module driver, hardware operations can be performed through the driver

● API lib: Interface library, which realizes the abstraction of hardware functions and facilitates the developr application layer. The API library has five main parts:

◆ libimp: Multimedia function library. Such as H264 encoding, JPEG encoding, IVS and audio, etc.
◆ libsysutils: system function library. Such as restarting, setting the system time and battery functi

◆ libalog: ISVP-SDK log implementation library
◆ libivs_inf: IVS algorithm library, including cross-line detection, perimeter prevention, etc.
◆ libmxu: 128-bit mxu accelerated instruction operator library
● Application: Application layer. Realize functional logic, etc.

● Application recommends using the API provided by the SDK library and cooperating with drivers For some special functional requirements, you can also directly call the kernel interface for develop

2. T30 must read

2.1 Uboot compilation
● Uboot compilation process:
u-boot can be compiled separately without relying on other codes. The board configuration file of T30 u-b located in include/configs/isvp_t30.h. The default compilation configuration file is described in Table 3-1.

表 3-1 T30 芯片对应 uboot 编译文件

芯片型号	编译命令	说明
T30N	make isvp_t30_sfcnor	表示编译 norflash 启动的 uboot，针对 T30N 芯片
T30L	make isvp_t30_sfcnor_lite	表示编译 norflash 启动的 uboot，针对 T30L 芯片
T30X	make isvp_t30_sfcnor_ddr128M	表示编译 norflash 启动的 uboot，针对 T30X 芯片
T30A	make isvp_t30a1_sfcnor_ddr128M	表示编译 norflash 启动的 uboot，针对 T30A 芯片
T30N	make isvp_t30_msc0	表示编译 SD 卡启动的 uboot，针对 T30N 芯片
T30L	make isvp_t30_msc0_lite	表示编译 SD 卡启动的 uboot，针对 T30L 芯片
T30X	make isvp_t30_msc0_ddr128M	表示编译 SD 卡启动的 uboot，针对 T30X 芯片
T30A	make isvp_t30a1_msc0_ddr128M	表示编译 SD 卡启动的 uboot，针对 T30A 芯片

First of all : $ MAKE distclean to clear the old configuration
Step 2: $ make isvp_t30_xxx compiles the corresponding uboot according to the corresponding chip, and g
the corresponding u boot-with-spl.bin
● Common modifications in the Uboot configuration file:
1) CONFIG_BOOTARGS, the main modification point is the memory after the kernel is started Configuratio
size configuration. (Note: mem means the memory reserved after the kernel is started, rmem means the m
reserved for the SDK (including the memory of the ISP module); the sum of the two is the real memory siz
chip; the specific size can refer to the code).
2) CONFIG_BOOTCOMMAND, configure the command executed by uboot. For example: add the commanc
the SD card in norflash boot mode, "sf probe;sf read 0x80600000 0x40000 0x280000; bootm 0x80600000"
"mmc read 0x80600000 0x1800 0X3000; bootm 0x80600000".
3) CONFIG_BOOTDELAY, configure the waiting time of uboot.
4) Need to add new norflash chip support.
5) Add password function in uboot:

Modify the configuration file and add the following content in
isvp_t30.h : #define CONFIG_AUTOBOOT_KEYED // Mandatory
#define CONFIG_AUTOBOOT_STOP_STR "123456" // Mandatory, the password set by uboot.
#define CONFIG_AUTOBOOT_PromPT "Press xxx in %d second", bootdelay //Optional, uboot pr
#define CONFIG_AUTOBOOT_DELAY_STR "linux" //Optional,
the specific implementation of uboot prompt information code is abortboot_keyed(int bootdelay)
common/main.c; you can change it according to your needs.

6) SD card upgrade problem
Add #define CONFIG_AUTO_UPDATE definition in isvp_t30.h. The specific code is implemented in
common/cmd_sdupdate.c. Points to note: LOAD_ADDR means to
load the corresponding memory on the SD card to the memory location. The default setting in the
0x82000000. Since this address is located on the heap of uboot, the heap size of common uboot is set in t
configuration of the CONFIG_SYS_MALLOC_LEN macro in isvp_t30.h; so
the size that this address can be used will be limited by the heap size, and The limitation of malloc space i
code. (Note: When you need to read larger files, you can increase CONFIG_SYS_MALLOC_LEN appropriatel

7) The size of the compiled uboot is larger than the limit
. The default limit of the uboot code is 26Kbytes for the spl part and 214Kbytes for the uboot part; a total
240Kbytes is limited. If the u-boot-with-spl.bin file generated by uboot is larger than 240Kbytes, it cannot
Solution 1:
Increase the limit of uboot; modify the definition of the CONFIG_SYS_MONITOR_LEN macro in isvp_t30.h; r
size of the boot partition in the CONFIG_BOOTARGS variable and the offset address of the subsequent par
the same time.
Solution 2:
If the generated u-boot-with-spl.bin exceeds 240Kbytes and less, you can compress uboot. Modify #undef
CONFIG_SPL_LZOP in isvp_t30.h to #define CONFIG_SPL_LZOP; then
recompile and burn the file name u-boot-lzo-with-spl.bin

8) uboot network problem The
default isvp configuration includes the code of the Ethernet part. If the product does not require TFTP dow
NFS mount during the uboot stage, the Ethernet part of the code can be cut out to reduce uboot.
Specific operation: Open the isvp_T30.h configuration file and annotate the #define CONFIG_CMD_NET ma

Previous: Digital Voice Recording and Playback Chip Winbond ISD15000 Multilingual Application Solution
Next: Winbond Electronics and Karamba New Solutions to Strengthen Supply Chain Security Protection