



# 龙芯大赛支持辅导培训



# 目录

- 01 龙芯平台介绍
- 02 龙芯平台系统架构
- 03 龙芯大赛开发支持

## 推荐平台

本赛题只针对本科组及研究生组，要求参赛队基于龙芯 2K1000-LA 芯片平台，设计并实现一个有创意的嵌入式系统作品。



## 序号

## 赛题

1

- 基于龙芯的可信嵌入式平台

2

- 基于龙芯的嵌入式工业互联网监控系统

3

- 工业物联网网关

4

- 国产数字示波器

5

- 工业无线物联网传感系统

## 龙芯派LA-Q2

龙芯2K1000LA处理器在引脚和接口兼容的基础上，处理器更新为基于LoongArch（简称龙架构）的 LA264 处理器核，也标志着龙芯工控全面转向LoongArch，可助力参赛队伍在龙芯平台上实现产学研融合贯通的创新科技成果。



## 2K1000LA主控芯片：

龙芯2K1000LA在实现与原有版本2K1000引脚和接口兼容的基础上，处理器核更新为基于龙芯自主指令系统架构LoongArch®（简称龙架构）的LA264处理器核。龙芯2K1000LA还对原龙芯2K1000的部分外围接口进行了功能优化，全面提升了可靠性，可直接在原主板上进行替换升级。

芯片外围接口包括两路PCIE2.0、一路SATA2.0、4路USB2.0、两路DVO、64位DDR2/3及其它多种接口。龙芯2K1000LA在硬件接口完全兼容2K1000的基础上，替换了处理器核并通过调整设计进行了性能和功耗优化，可广泛应用于电力、轨交、石油石化、新能源、智能制造、工业互联网、工业网络安全等行业和领域。

功能	描述
处理器核	2 个 LA264 处理器核
主频	1GHz
功耗	1~5W（支持动态降频降压）
浮点单元	128位向量单元
峰值运算速度	8GFlops
高速I/O	PCIE2.0*2、SATA2.0
其他接口	SPI、UART、GPIO、NAND、SDIO、DVO、I2S、HDA、I2C、USB2.0、GMAC

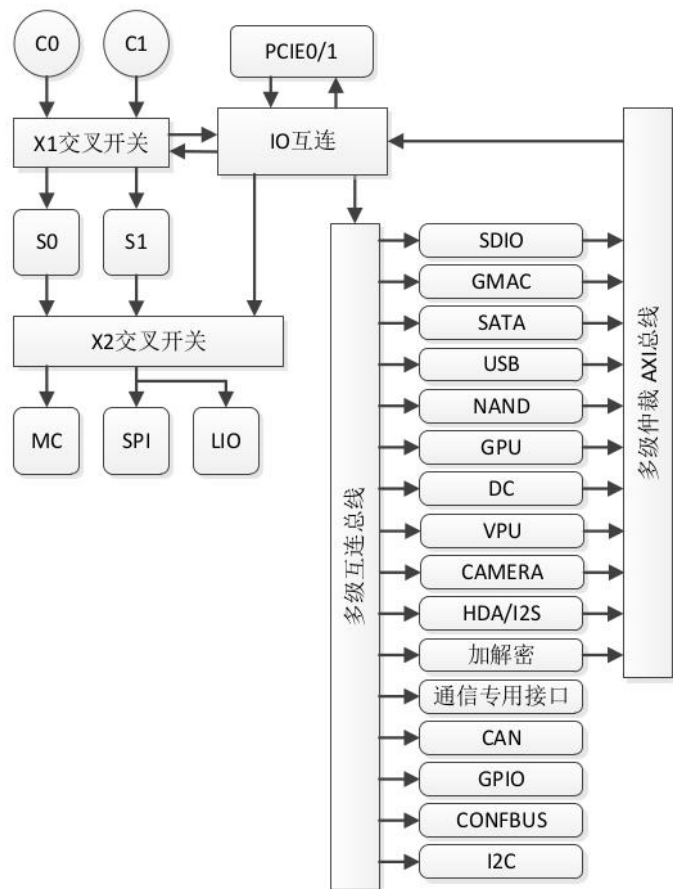


## LoongArch龙架构：

LoongArch是由龙芯团队2020年推出的新型 RISC指令系统。该指令系统摒弃了传统指令系统中部分不适应当前软硬件设计技术发展趋势的陈旧内容，吸纳了近年来指令系统设计领域诸多先进的技术发展成果。

在工程上，有助于硬件实现高性能低功耗的设计，也有利于软件的编译优化以及操作系统、虚拟机的开发；

在教学中，其简洁、灵活、模块化等特点，更能清晰阐释指令系统与底层微结构之间、上层软件与指令系统之间的密切关系，提供给读者针对不同架构进行相互参考对比的多样性选择。



## 2K1000LA主控芯片：

一级交叉开关连接两个处理器核、两个二级Cache一级IO子网络（Cache访问路径）。二级交叉开关连接二级Cache、内存控制器、启动模块(SPI或者LIO)以及IO自网络（Uncache访问路径）。

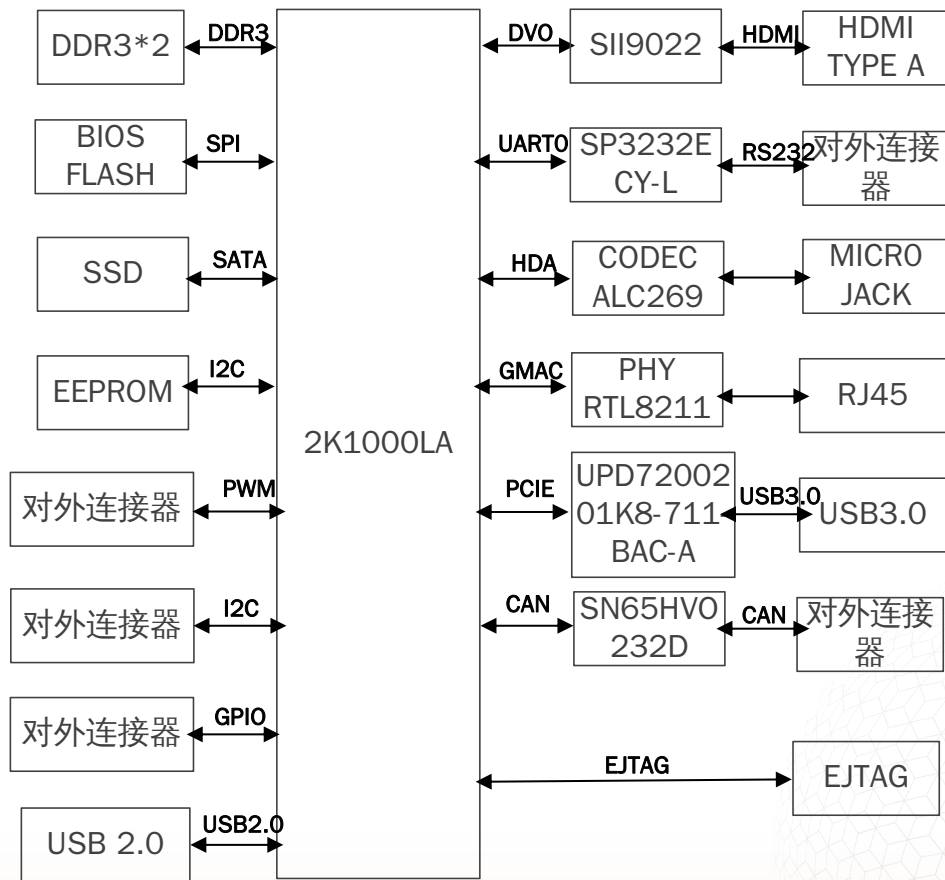
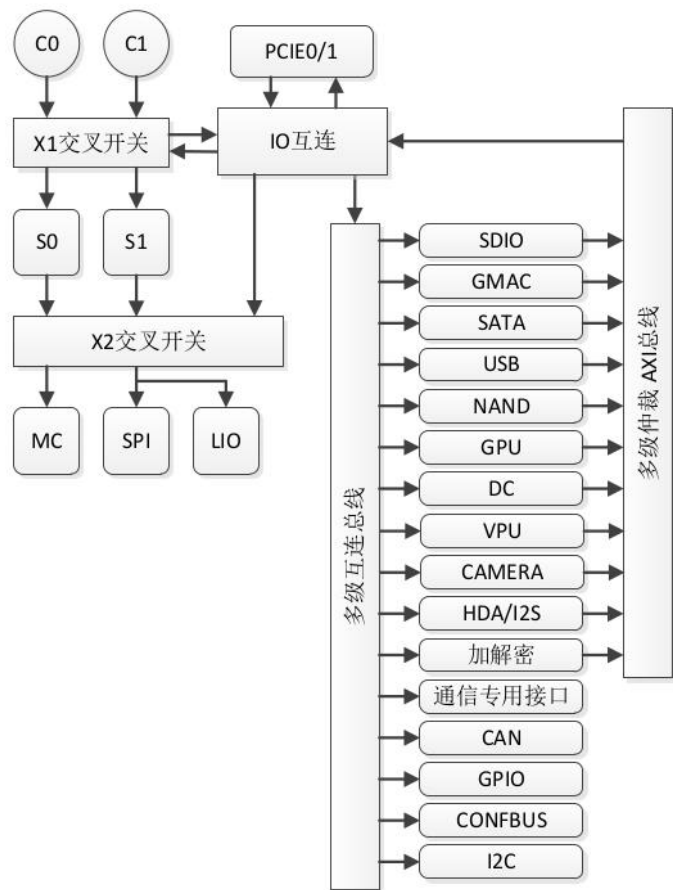
**IO子网络连接一级交叉开关，以减少处理器访问延迟。** IO子网络中包括需要DMA的模块（PCIE、GMAC、SATA、USB、HAD/I2S、NAND、SDIO、DC、GPU和加解密模块）和不需要DMA的模块，需要DMA的模块可以通过Cache或者Uncache方式访问内存。

- 详细内容可以查阅 《龙芯2K1000LA处理器用户手册》





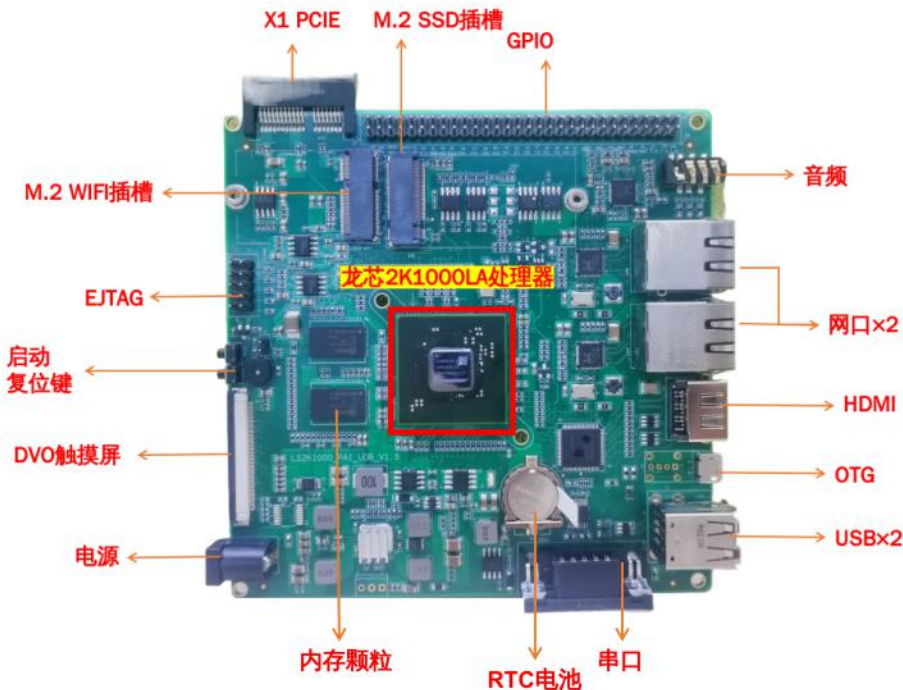
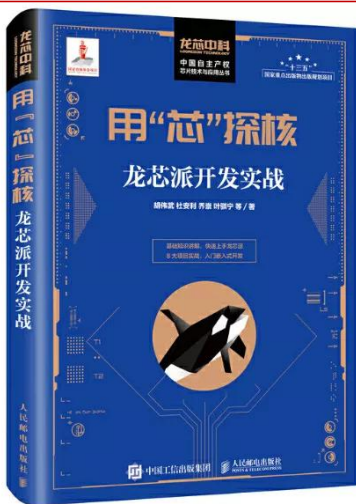
# 硬件资源和布局





## 龙芯派LA-Q2

龙芯派二代开发平台是一款非常迷你的产品，主要面向电子发烧友、极客、程序员或者计算机编程教育，可以更好的激励国产芯片爱好者的开发兴趣。

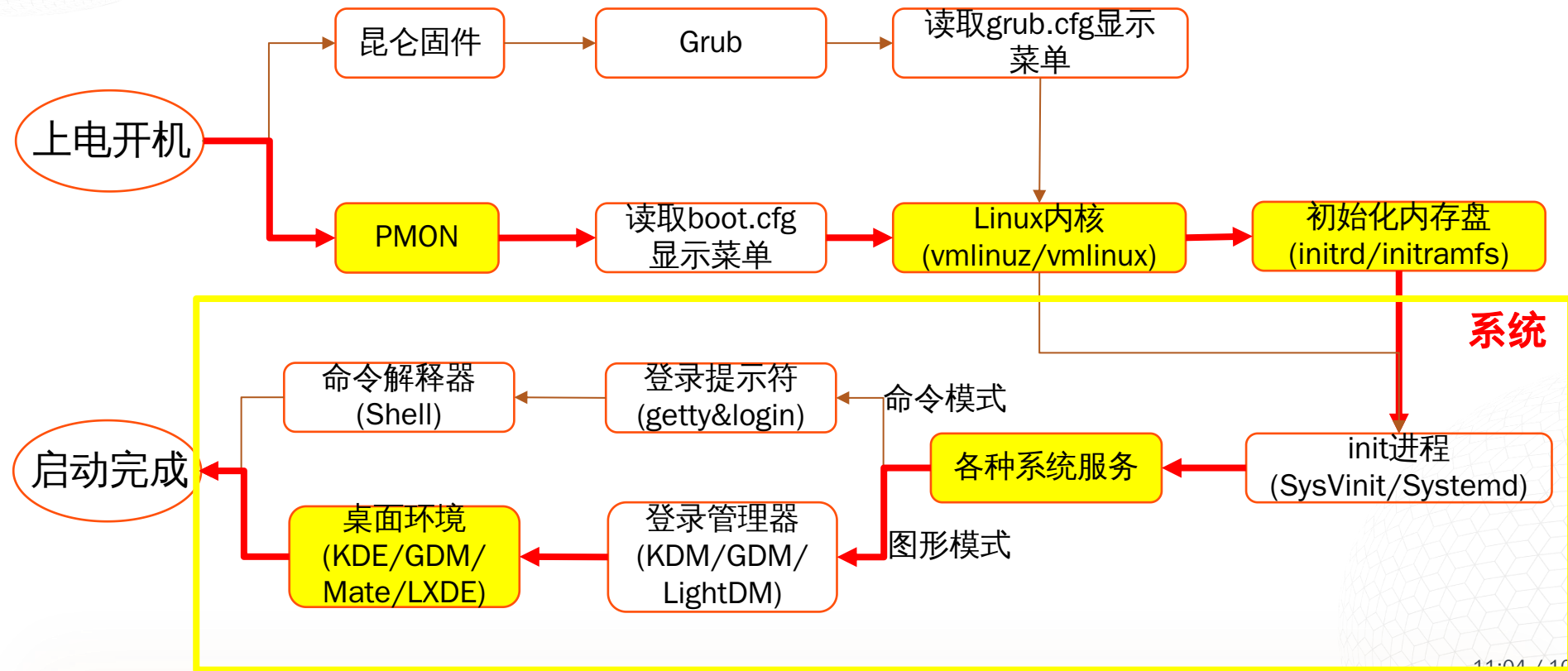


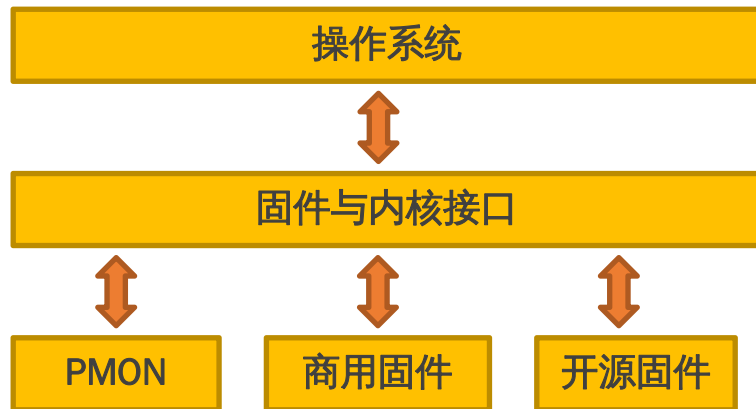
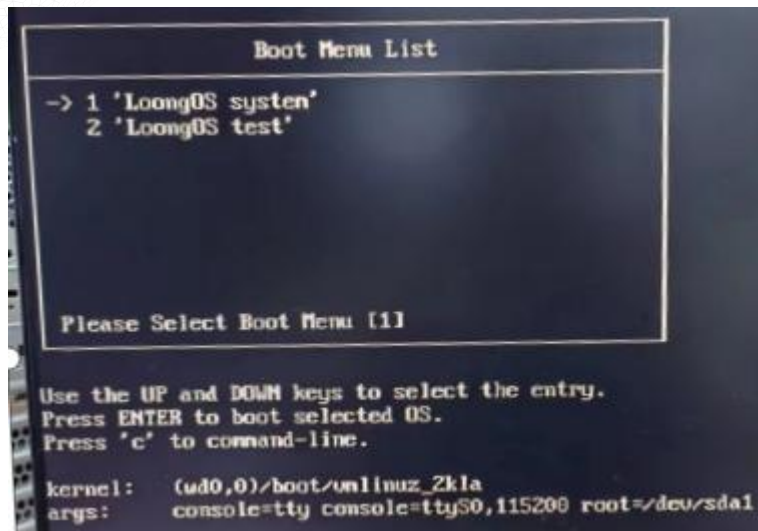




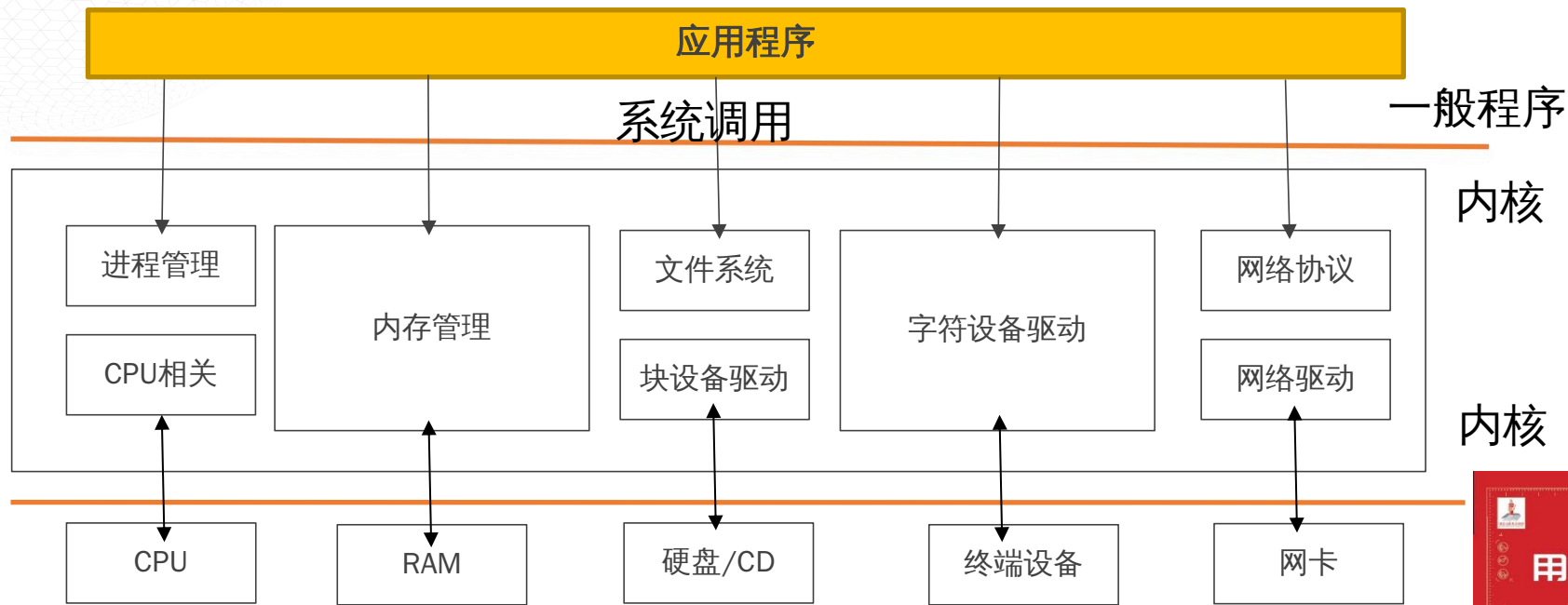
# 目录

- 01 龙芯平台介绍
- 02 龙芯平台系统架构
- 03 龙芯大赛开发支持

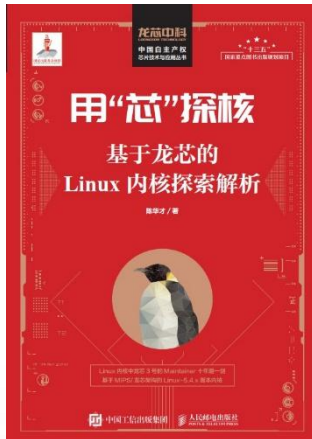




Pmon在龙芯平台上开源，具有强大丰富的功能，除基本的I/O功能外，还包括CPU初始化、板级外设初始化与检测、操作系统引导和调试等功能。



Linux操作系统的kernel内核是操作系统的核心，控制硬件设备，提供应用程序使用。

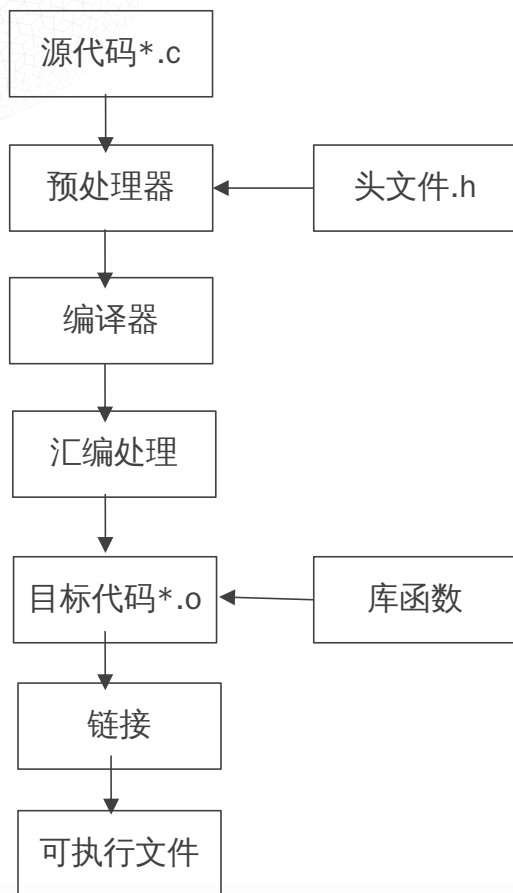




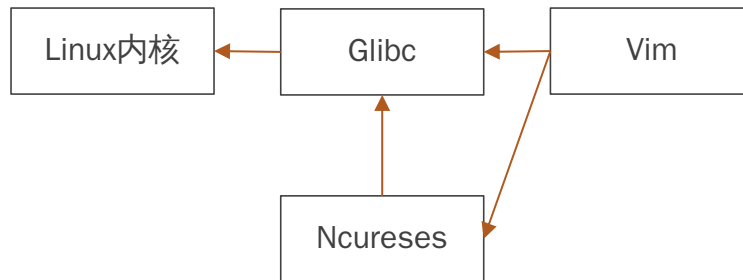
LoongOS是具备精简、高效、实时特征的工控类操作系统，基于通用Linux内核，利用RT-Linux技术实现实时性。

## 系统优势

- LoongOS简化了复杂的传统X个人计算机图形系统
- 支持FrameBuffer、EGL、Wayland三种图形应用模式，其中FrameBuffer（二维）和EGL（三维）都是单窗口应用全屏模式；
- 具备系统轻量、启动迅速、开发便捷等特点；
- Wayland模式可以支持多窗口应用，实现了全功能的三维OpenGL驱动和编程接口，具备简洁的个人计算机图形环境。
- LoongOS三种图形模式都支持Qt编程开发和应用环境，支持兼容VxWorks的RTAPI编程接口。
- 针对嵌入式系统的需求，实现了文件系统加固、掉电保护、分区保护、安全隔离等特性，还可以使用配套的集成开发环境LDK根据需求灵活定制专用系统。



## Vim的运行剖析



## GCC的编译流程分为四个步骤：

预处理(Pre-Processing)、编译(Compiling)、汇编(Assembling)、链接(Linking)





应用程序(包括开发套件、浏览器、办公软件等)

操作界面(命令用户界面GUI/图形用户界面GUI)

Linux操作系统(UserSpace)

系统服务

管理软件

基础运行时库(如Glibc等)

系统调用(System Call)

Linux内核(kernel)

网络协议

文件系统

设备驱动

内存管理

异常/中断处理

进程管理

BIOS(如PMON固件)

硬件(处理器2K1000LA、内存、外设、硬盘等)



# 目录

- 01 龙芯平台介绍
- 02 龙芯平台系统架构
- 03 龙芯大赛开发支持



## GPIO引脚:

龙芯 2K1000 共有 60 个 GPIO 引脚, 4 个为专用 GPIO, 其余 56 个与其他功能复用。其中GPIO0~GPIO3 为专用 GPIO 管脚, 所以不需要设置复用, 如下图所示。注意! 如果用的不是专用 GPIO 引脚, 则需要设置复用关系!

3	无复用	专用 GPIO 引脚
2	无复用	专用 GPIO 引脚
1	无复用	专用 GPIO 引脚
0	无复用	专用 GPIO 引脚

地址	名称	描述
0x1fe10500	GPIO0_OEN	GPIO 输出使能, 低有效
0x1fe10508	GPIO1_OEN	保留
0x1fe10510	GPIO0_O	GPIO 输出值
0x1fe10518	GPIO1_O	保留
0x1fe10520	GPIO0_I	GPIO 输入值
0x1fe10528	GPIO1_I	保留
0x1fe10530	GPIO0_INTEN	GPIO 的低 64 位中断使能
0x1fe10538	GPIO1_INTEN	保留

GPIO 的方向寄存器的地址  
为 0x1fe10500  
数据寄存器的地址为  
0x1fe10510



## Pmon下编写C程序

在Pmon源代码的Targets/LS2K/dev目录下使用命令

vim topeet\_led.c

创建一个 c 程序, 并输入代码:

操作的地址是 0xbfe10500 和

0xbfe10510 而不是我们在上一小节所说

0x1fe10500和 0x1fe10510 呢?

因为系统上电启动后, CPU 是从系统的虚拟地址

0xbfc00000 启动的,其对应的物理地址为 0x1FC00000, 所以这里我们操作的是 0x1fe10500 和 0x1fe10510 对应的虚拟地址0xbfe10500 和 0xbfe10510。

```
#include <stdio.h>
/*
 * 初始化 led 管脚
 */
void led_init(void)
{
    *(volatile int *)0xbfe10500 &= ~(0x01); //设置方向为输出
    *(volatile int *)0xbfe10510 &= ~(0x01); //默认输出低电平
    printf("led init!\n");
}
/*
 * 点亮 led
 */
void led_on(void)
{
    *(volatile int *)0xbfe10510 |= 0x01; //输出高电平
    printf("led3 on!\n");
}
/*
 * 熄灭 led
 */
void led_off(void)
{
    *(volatile int *)0xbfe10510 &= ~(0x01); //输出低电平
    printf("led off!\n");
}
```



## 配置编译规则

进到 Pmon 源码下的 Targets/LS2K/conf 文件夹

然后使用命令 vim files.LS2K 打开配置文件

并在配置文件的最下面添加以下代码

file Targets/LS2K/dev/topeet\_led.c

```
Terminal 终端 - loongson@loongson-pc: ~/pmon_dir/pmon-loongson3/Targets/LS2K/conf
文件(F) 编辑(E) 视图(V) 终端(T) 标签(A) 帮助(H)

# $Id: files.Bonito
#
# Bonito Target specific files
#
file Targets/LS2K/pci/pci_machdep.c
file Targets/LS2K/ls2k/tgt_machdep.c
file Targets/LS2K/ls2k/dc.c
file Targets/LS2K/pci/ls2k_pci.c
file Targets/LS2K/dev/topeet_led.c

define localbus { [base = -1 ] }
device localbus
attach localbus at mainbus
file Targets/LS2K/dev/localbus.c          localbus
file Targets/LS2K/dev/spi_w.c
file Targets/LS2K/dev/nand_opt.c
file Targets/LS2K/dev/set_cpu_ddr_freq.c
file Targets/LS2K/dev/set_vol.c
file Targets/LS2K/dev/rtc.c
file Targets/LS2K/dev/eeprom.c
file Targets/LS2K/dev/load_dtb.c          cmd_dtb
-- 插入 --
```



## 控制GPIO引脚

根据pmon 的启动流程, 最终会在  
Targets/LS2K/ls2k/tgt\_machdep.c中。  
里面的 initmips 函数初始化外设。

## 编译验证

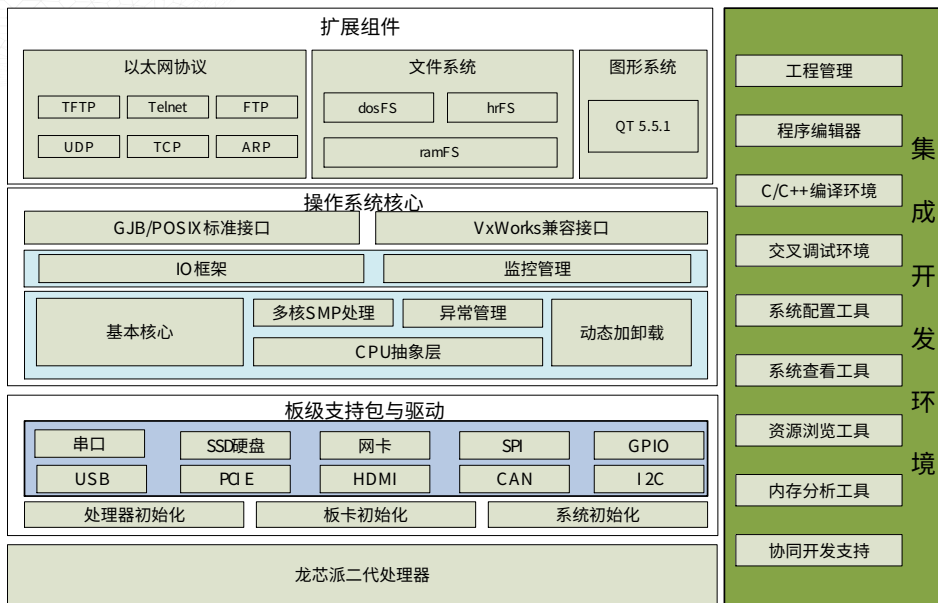
编译的 Pmon 烧写到开发板,如果没  
有成功点亮  
可以使用 Ejtag 单步调试 pmon。

```
Terminal 终端 - loongson@loongson-pc: ~/pmon_dir/pmon-loongson3/Targets/LS2K/ls2k
文件(F) 编辑(E) 视图(V) 终端(T) 标签(A) 帮助(H)

{
FREQSCALE1_CORE1EN = 2,
};

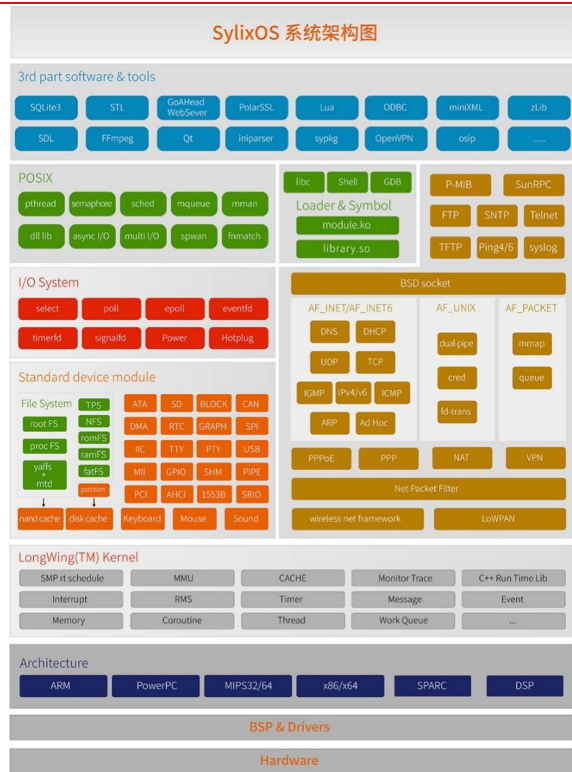
void initmips(unsigned long long raw_memsz)
{
    unsigned int hi;
    unsigned long long memsz;
    unsigned short i;
    //core1 run wait_for_smp_call function in ram
    asm volatile(".set mips64;sd %1,(%0);.set mips0;:::\"r\"(0xbfe11120),\"r\"(&wait_for_smp_call));");
#ifdef CONFIG_UART0_SPLIT
    *(volatile int *)0xbfe10428 |= 0xe;
#endif
#ifdef SATA_TX_REVERT
    *(volatile int *)0xbfe10450 |= (1<<9);
#endif
#ifdef SEL_HDA
    *(volatile int *)0xbfe10420 &= ~(7<<4);
    *(volatile int *)0xbfe10420 |= (1<<4);
#endif
#ifdef LCD_EN
```





ReWorks产品软件的功能构成

锐华嵌入式实时操作系统试用：  
<https://www.wjx.top/vm/Y0lKB7Q.aspx>



SylixOS系统

翼辉SylixOS实时操作系统试用：  
<https://wenjuan.feishu.cn/m?t=sCyIFO2emfKi-x8zc>

## 嵌入式信息系统LoongOS

LoongOS是具备精简、高效、实时特征的工控类操作系统。

LoongOS基于通用Linux内核，利用RT-Linux技术实现实时性。

LoongOS简化了复杂的传统X个人计算机图形系统，支持FrameBuffer、EGL、Wayland三种图形应用模式。



LoongOS显示环境

模式	支持显卡环境	二三维支持特性
Wayland模式	龙芯自主GPU、国产与商业GPU	基于OpenGL2.0、OpenGL/ES2.0以上实现
EGL模式	龙芯自主GPU、国产与商业GPU	基于OpenGL2.0、OpenGL/ES2.0以上实现
FrameBuffer模式	龙芯自主GPU、国产与商业GPU	支持2D加速接口



LoongOS具有丰富的API编程接口，并有配套的集成开发环境LDK。LDK支持可视化安装、远程调试，支持所见即所得的开发方式。

API

支持

备注

Qt2D

支持

支持linuxfb、wayland、eglfs、VNC等

QtOpenGL

支持

独立显卡环境支持eglfs模式和wayland模式，7A和2K环境仅支持eglfs模式；基于OpenGL或OpenGL ES封装

QtQuick

支持

独立显卡环境支持eglfs模式和wayland模式，7A和2K环境仅支持eglfs模式

QtWebengine

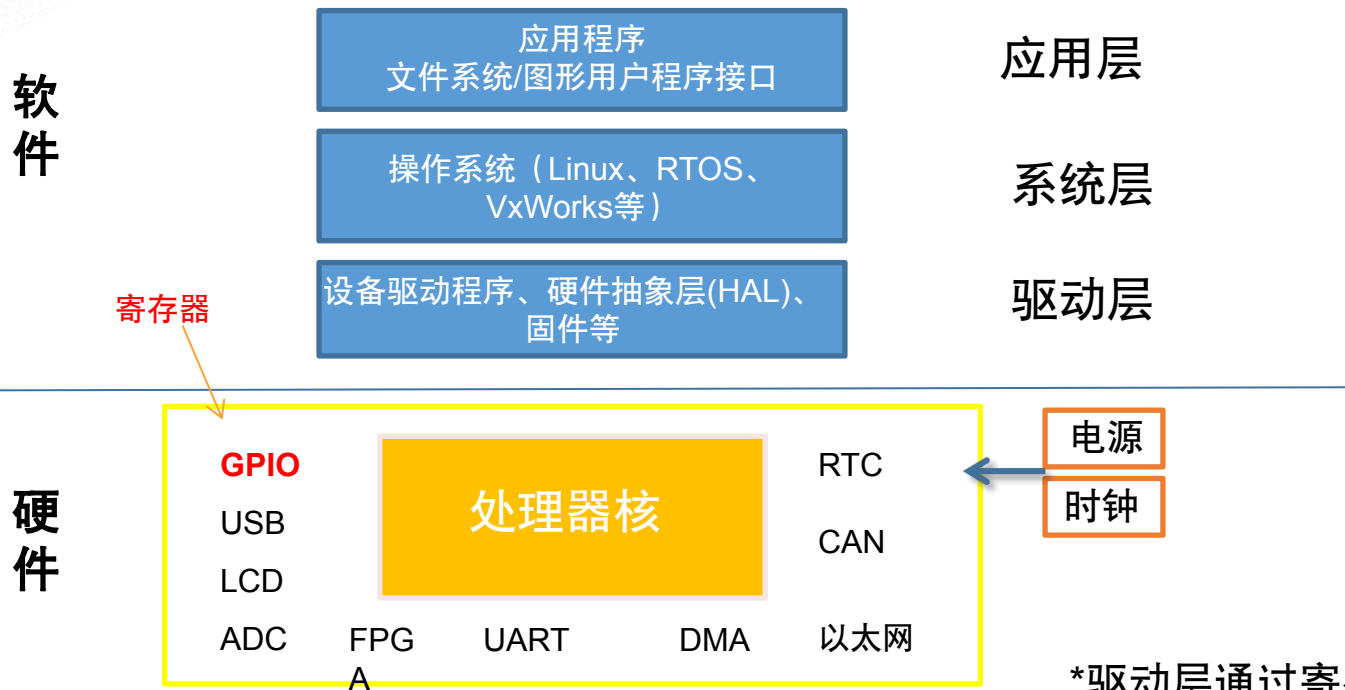
支持

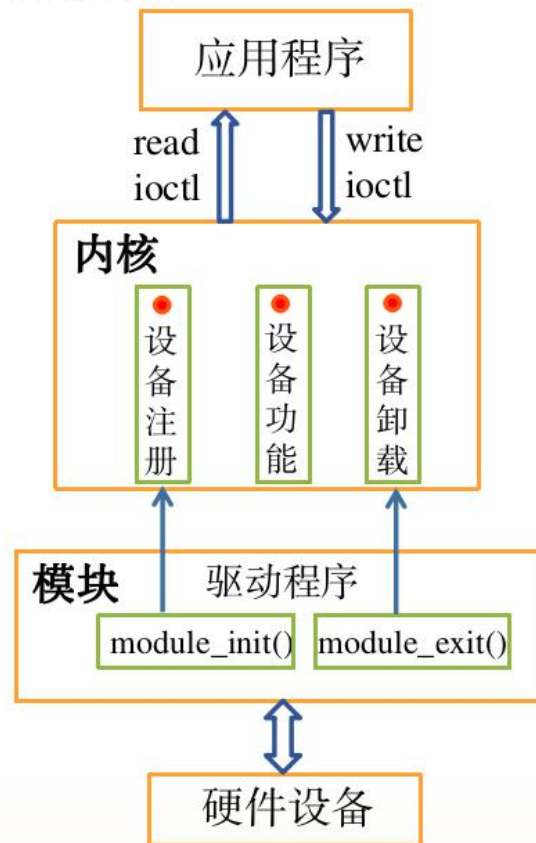
基于最新的chromium社区版本

NodeJS

支持

Vue、Echarts、react等开发组件支持





## 访问过程:

应用程序访问设备,将设备当作文件处理,在系统上层通过`read`、`write`、`ioctl`、`poll`等函数,访问设备文件,与内核进行交互.当内核进行设备注册时,会调用驱动模块里的函数,完成设备注册.

应用程序、系统、内核和驱动相互配合实现设备功能,若内核需要卸载设备,则调用驱动模块里的函数,卸载驱动模块.

## 开发示例:

ls2k\_hello\_world驱动程序模块的加载,会调用`module_init()`中的函数,通过访问GPIO寄存器,读取60号引脚的值,输出到系统中显示.



## ls2k\_hello\_world.c

```
#include<linux/kernel.h>
#include<linux/init.h>
#include<linux/module.h>
#include <loongson-2k.h>
// GPIO寄存器的操作使用可以查看龙芯2K1000处理器用户手册
#define GPIO_EN 0x1fe10500 // GPIO配置寄存器，输出使能，低有效
#define GPIO_I 0x1fe10520 // GPIO输入采样，反映引脚的值
// Module Init function
static int __init ls2k_hello_world_init(void)
{
    int gpio_num=60;
    int value;
    pr_info("ls2k_readq(GPIO_EN:%llx):%llx\n",GPIO_EN,ls2k_readq(GPIO_EN));
    // 位运算
    // 将GPIO_EN寄存器(0x1fe10500)的值，右数第(gpio_num+1)位变为1,也就是控制60号引
    // 脚为输入
    ls2k_writq(ls2k_readq(GPIO_EN) | (1 << gpio_num), GPIO_EN);
    // 将GPIO_I寄存器的值，取右数第（gpio_num+1）位的值，读取60号引脚的值
    value = ls2k_readq(GPIO_I) >> gpio_num & 1;
    pr_info("gpio_num is %d,value is %d\n", gpio_num,value);
    pr_info("Hello World.\n");
    pr_info("Kernel Module Inserted Successfully...\n");
    return 0;
}
// Module Exit function
static void __exit ls2k_hello_world_exit(void)
{
    pr_info("Kernel Module Removed Successfully...\n");
}
module_init(ls2k_hello_world_init);
module_exit(ls2k_hello_world_exit);
```

## 13 GPIO

龙芯 2K1000 共有 60 个 GPIO 引脚，4 个为专用 GPIO，其余 56 个与其他功能复用。下面具体介绍与 GPIO 相关的配置寄存器。

表 13-1 GPIO 配置寄存器

地址	名称	描述
0x1fe10500	GPIO0_OEN	GPIO 输出使能，低有效
0x1fe10508	GPIO1_OEN	保留
0x1fe10510	GPIO0_O	GPIO 输出值
0x1fe10518	GPIO1_O	保留
0x1fe10520	GPIO0_I	GPIO 输入值
0x1fe10528	GPIO1_I	保留
0x1fe10530	GPIO0_INTEN	GPIO 的低 64 位中断使能
0x1fe10538	GPIO1_INTEN	保留

### 13.1 GPIO 方向控制

地址：0x1fe10500

表 13-2 GPIO 方向控制

位域	名称	访问	初值	描述
63:0	GPIO0_OEN	R/W	64'hFFFFFF_FFFF_FFFF	0 为输出，1 为输入

### 13.2 GPIO 输出设置

地址：0x1fe10510

表 13-3 GPIO 输出设置

位域	名称	访问	初值	描述
63:0	GPIO0_O	R/W	0	0 输出低电平，1 输出高电平

### 13.3 GPIO 输入采样

地址：0x1fe10520

表 13-4 GPIO 输入采样

位域	名称	访问	初值	描述
63:0	GPIO0_I	R	-	反映 GPIO 引脚的值





## ls2k\_hello\_world.c

```
#include<linux/kernel.h>
#include<linux/init.h>
#include<linux/module.h>
#include <loongson-2k.h>
// GPIO寄存器的操作使用可以查看龙芯2K1000处理器用户手册
#define GPIO_EN 0x1fe10500 // GPIO配置寄存器，输出使能，低有效
#define GPIO_I 0x1fe10520 // GPIO输入采样，反映引脚的值
// Module Init function
static int __init ls2k_hello_world_init(void)
{
    int gpio_num=60;
    int value;
    pr_info("ls2k_readq(GPIO_EN:%llx):%llx\n",GPIO_EN,ls2k_readq(GPIO_EN));
    // 位运算
    // 将GPIO_EN寄存器(0x1fe10500)的值，右数第(gpio_num+1)位变为1,也就是控制60号引脚为输入
    ls2k_writq(ls2k_readq(GPIO_EN) | (1 << gpio_num), GPIO_EN);
    // 将GPIO_I寄存器的值，取右数第（gpio_num+1）位的值，读取60号引脚的值
    value = ls2k_readq(GPIO_I) >> gpio_num & 1;
    pr_info("gpio_num is %d,value is %d\n", gpio_num,value);
    pr_info("Hello World.\n");
    pr_info("Kernel Module Inserted Successfully...\n");
    return 0;
}
// Module Exit function
static void __exit ls2k_hello_world_exit(void)
{
    pr_info("Kernel Module Removed Successfully...\n");
}
module_init(ls2k_hello_world_init);
module_exit(ls2k_hello_world_exit);
```

## 功能描述：

ls2k\_hello\_world是上手开发驱动的入门示例。驱动程序模块的加载,会调用module\_init()中的函数,通过访问GPIO寄存器,读取60号引脚的值,输出到系统中显示。

在系统终端中,输入sudo dmesg命令,可以查看pr\_info的输出信息.驱动程序模块卸载时,会执行module\_exit()函数,这里ls2k\_hello\_world\_exit是使用pr\_info函数,输出模块卸载成功的信息



## 操作步骤:

- 1、编译驱动模块,将Makefile和ls2k\_hello\_world.c放在同一目录下,执行make命令编译.
- 2、执行insmod命令加载模块,例如sudo insmod ls2k\_hello\_world.ko,加载驱动函数功能.
- 3、卸载驱动模块,使用结束后,执行rmmod命令即可,例如sudo rmmod ls2k\_hello\_world.

## Makefile:

```
obj-m += ls2k_hello_world.o
KDIR = /lib/modules/$(shell uname -r)/build #
内核头文件存放目录
all:
    make -C $(KDIR) M=$(shell pwd) modules
clean:
    make -C $(KDIR) M=$(shell pwd) clean
```

## 内容详解:

首先make 命令先根据Makefile里定义的规则,obj-m += ls2k\_hello\_world.o,在该目录下找到ls2k\_hello\_world.c文件;  
然后在内核头文件存放目录下导入ls2k\_hello\_world.c指定需要的函数头文件,如linux/kernel.h、loongson-2k.h等;  
最后执行Makefile中的make编译规则,调用系统中的编译工具链gcc将ls2k\_hello\_world.c源文件编译成模块ko等系列文件.

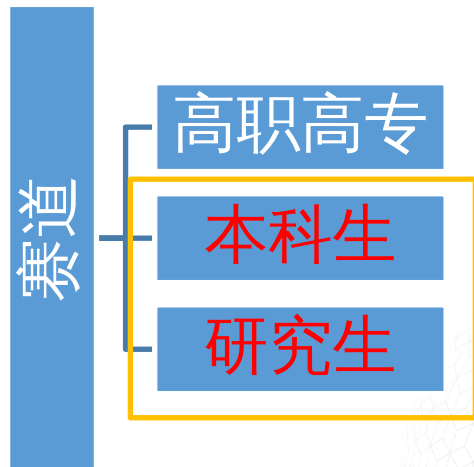


## 参赛对象:

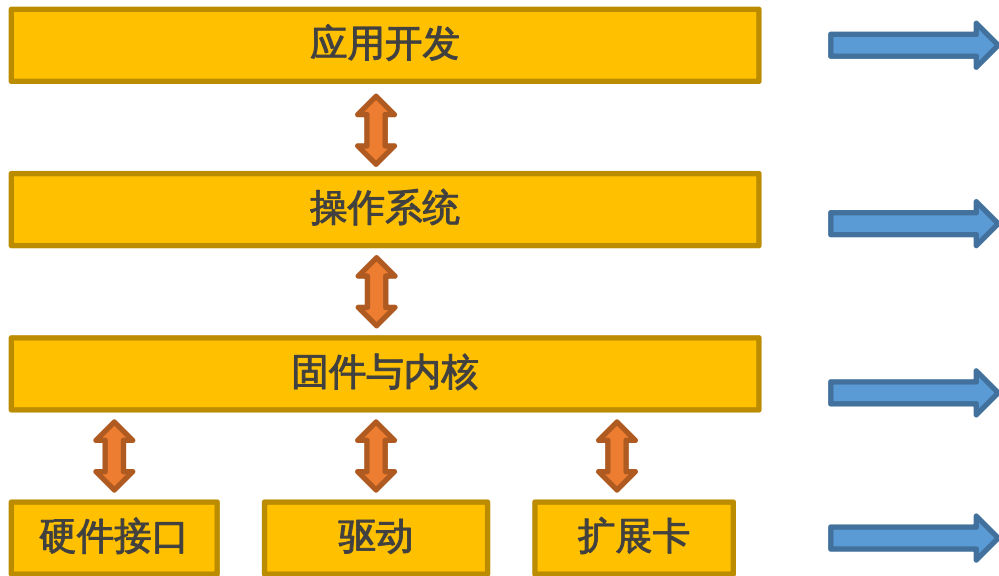
大赛参赛对象包括（但不限于）国内外高校电子电气类相关专业（电子、信息、计算机、自动化、电气、仪科等）学生；同时欢迎高水平高职高专学校学生参赛。本科生、研究生、高职高专分别组队，参赛队中有研究生成员的即为研究生参赛队。每个参赛队由不多于 3 名学生组成，可有不超过 2 名指导老师。



- QQ群里支持的对象包括带队的指导老师



- 能力水平存在差异



- 开发环境：C/C++语言开发、Python语言开发、Qt开发等等
- 这类问题基本与平台无关，可以沿用开发经验
- 实时操作系统：ReWorks、SylixOS
- LoongOS系统：命令操作、功能调试、
- 这类问题属于系统层，绝大部分可转化成普遍的技术问题，网上搜索答案和解决思路。
- 编译内核、更新固件(板卡容易变砖)
- 这类问题属于底层，可以参考手册和资料解决
- 硬件适配情况，驱动代码适配
- 这类问题属于硬件，龙芯在手册中提供相关硬件接口参考，同时要根据参赛队伍的作品硬件的情况处理

软件层面（多）：使用群里资源，远程协助处理。

硬件层面（少），参赛队伍可以向购买平台申请售后，比如iCEasy商城、淘宝等。



## 医药配送机器人：

当本项目是基于龙芯教育派及多源融合SLAM的医药配送机器人，将利用自动定位导航，机器视觉识别与机械臂抓取实现配送医药物资的需求。它能够根据医护人员所需求的药物种类，将药品、处方等物品自药房配送至对应患者的病房或取药柜台。因而主要指标为场景地图的构建速度与效果、小车位置的定位情况，路径规划情况、小车的自主调速能力、避障能力，以及识别药物准确率、识别药品的速度，机械臂的抓取与配送情况。



## 电动车禁入电梯管控系统：

当前大部分物业已经明确禁止电动车违规充电，然而一些高层用户为了方便，直接将电动车推进电梯，在家中或楼道充电。这种行为极有可能引起火灾，极大危害整栋楼房居民的人身和财产安全。

针对以上问题，本项目采用计算机视觉和深度学习技术，依靠龙芯教育派边缘计算处理器实时完成对输入图像的识别，结合 TPU 神经网络加速棒开发了具有图像全屏周界防护功能的电动车禁入系统。





参赛队伍不熟悉：需提供信息、资料、渠道等等

- 学生不熟悉赛道，咨询相关信息。选择赛题、开发平台获取方式、支持群等
- 学生不熟悉平台，需要快速学习上手。开发板资料、教材、指导视频等

开始  
(解答参赛疑惑)

发展

(协助处理问题)

- 参赛队伍熟悉平台后，开始动手开发作品，需要获取支持：比如工具链、系统重装。
- 在制作作品时，遇见各类技术问题。比如网络不通、系统权限、定制系统、编译等等。

参赛队伍提交作品：产生创新成果，提供相应指导。

- 根据作品的要求，提供优化建议，可参考以前龙芯赛道的优秀作品。
- 将作品进一步转化为产品，并产生创新成果。比如专利、补丁、论文等等。

结尾  
(注意比赛公平)

参赛队伍开发作品：提供技术支持，需协助处理技术问题





## 为人民造龙芯

自主研发CPU核心技术、构建自主创新的产业体系，**这是最艰难的道路，但前景最光明**，走通后对国家和人民的好处最大。只有符合国家利益的企业才能长远发展！

“**市场换技术**”的道路走不通

通过合资式把中国市场给予国外企业，通过引进技术发展CPU产品，只是将一副锁链换成另外一副锁链。

只能走“**市场带技术**”的道路

通过体制内市场引导，带动技术进步，逐步，**打造自主生态体系**，我们失去的只有锁链，得到的将是完整世界。

The logo consists of a red circular ring with a textured, brush-stroke-like appearance. Inside the ring, the Chinese characters "龙芯中科" are written in a bold, red, sans-serif font. Below the Chinese characters, the English text "LOONGSON TECHNOLOGY" is written in a smaller, red, sans-serif font.

龙芯中科  
LOONGSON TECHNOLOGY

为人民做龙芯