

$$V(k_0) = \sum_{t=0}^{\infty} [\beta^t \ln(1 - \alpha\beta) + \beta^t \alpha \ln k_t]$$

Homemada Script Luanguage Note

$$= \frac{\alpha}{1 - \alpha\beta} \ln k_0 + \frac{\ln(1 - \alpha\beta)}{1 - \beta} + \frac{\alpha\beta}{(1 - \beta)(1 - \alpha\beta)} \ln(\alpha\beta)$$

$$\text{左边} = V(k) = \frac{\alpha}{1 - \alpha\beta} \ln k + \frac{\ln(1 - \alpha\beta)}{1 - \beta} + \frac{\alpha\beta}{(1 - \beta)(1 - \alpha\beta)} \ln(\alpha\beta)$$

右边

$$y) + \beta V(y)\}$$

利用 FOC 和包络条件求解得

，求右边。

$$\text{右边} = \max \{u(f(k) - y) + p v(y)\}$$

$$= u(f(k) - g(k)) + \beta \left[\frac{\alpha}{1 - \alpha\beta} \ln g(k) + A \right]$$

There is no good end for the fuck thieves country.

$$= \ln(k - \alpha\beta k) + \beta \left[\frac{\alpha}{1 - \alpha\beta} \ln \alpha\beta k + A \right]$$

$$= \ln(1 - \alpha\beta) + \alpha \ln k + \beta \left[\frac{\alpha}{1 - \alpha\beta} [\ln \alpha\beta + \alpha \ln k] + A \right]$$

$$= \alpha \ln k + \frac{\alpha\beta}{1 - \alpha\beta} \alpha \ln k + \ln(1 - \alpha\beta) + \frac{\alpha\beta}{1 - \alpha\beta} \ln \alpha\beta + \beta A$$

$$= \frac{\alpha}{1 - \alpha\beta} \ln k + \ln(1 - \alpha\beta) + \frac{\alpha\beta}{1 - \alpha\beta} \ln \alpha\beta + \beta A$$

$$= \frac{\alpha}{1 - \alpha\beta} \ln k + (1 - \beta)A + \beta A$$

$$= \frac{\alpha}{1 - \alpha\beta} \ln k + A$$

整理：LMin

整理时间：April 10, 2018

Email: luomin5417@gmail.com

所以，左边 = 右边，证毕。

目 录



1	语言处理器基本概	1
1.1	机器语言与汇编语言	1
1.2	解释器与编译器	1
1.3	开发语言处理器	1
1.4	语言处理器结构与本书框架	1
2	Android Camera 系统框架	3
2.1	模型描述	3
2.2	代价函数	3
2.3	代价函数 (一)	4
2.3.1	编译方式	4
2.3.2	选项设置	4
2.3.3	数学环境简介	5
2.3.4	可编辑的字段	5
2.4	代价函数 (二)	6
2.5	梯度下降	6
2.6	梯度下降知识点总结	6
2.7	线性回归的梯度下降	6
2.8	本章课程总结	6
3	Linux Camera 驱动框架	7
3.1	矩阵和向量	7
3.2	加法和标量乘法	11
3.3	矩阵向量乘法	11
3.4	矩阵乘法	11
3.5	矩阵乘法特征	11
3.6	逆和转制	11
4	全志 A83t Camera 开发	12
4.1	总线与硬件接口	12
4.2	硬件原理图	13
4.3	驱动源码	14

4.4	内核与驱动配置	14
4.5	调试问题记录	15
4.6	总结	16
5	多变量线性回归	18
5.1	多功能	18
5.2	多元梯度下降法	18
5.3	多元梯度下降法演练 I-特征缩放	18
5.4	多元梯度下降法演练 II-学习率	18
5.5	特征和多项式回归	18
5.6	正规方程 (区别于迭代方法的直接解决)	18
5.7	正规方程在矩阵不可逆情况下的解决方法	18
5.8	完成并提交编程作业	18
	参考文献	19





第 1 章 语言处理器基本概



学而不思则罔，思而不学则怠！

—孔子

1.1 机器语言与汇编语言

本节主要介绍了机器语言与汇编语言：

- 机器语言是可以由机器直接解释执行的语言，一般才用二进制形式 [1]。
- 汇编语言一般是相对于机器语言更易理解的语言，但是又因不同的体系结构不同具有不同的指令集。

1.2 解释器与编译器

本节主要了解解释器和编译器的区别：

- 解释器根据程序中的算法执行运算。
- 编译器将某种语言写成的程序转换为另一种语言的程序。

1.3 开发语言处理器

本节主要介绍语言处理器，综合了解释器和编译器的优点。

1.4 语言处理器结构与本书框架

语言处理器的基本结构，如图 1.1 所示，画出了语言处理器的基本处理结构，源程序通过词法分析进行单词排列，然后经过语法分析生成一个抽象语法树，然后根据是编译器还是解释器执行不同的步骤，如果是编译器则直接生成机器语言，如果是解释器则一边分析抽象语法树，一边输出执行结果。

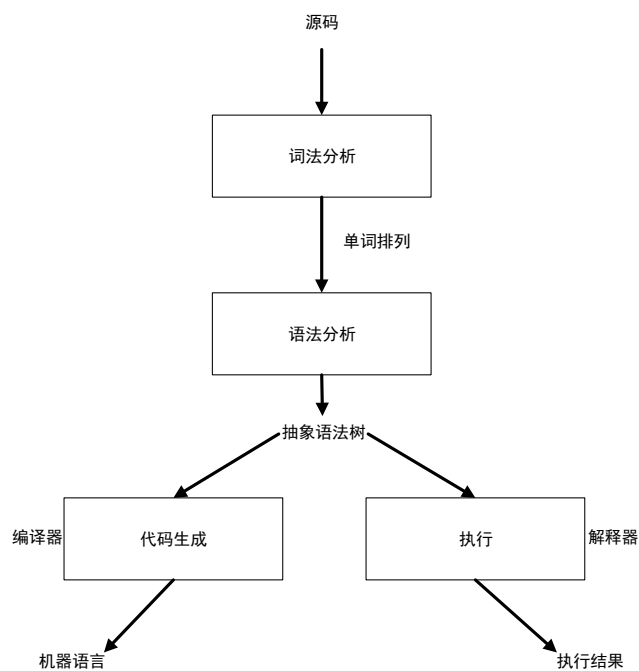


图 1.1: 语言处理器结构

无论是编译器还是解释器的结构都是大同小异的，第一步都是要先进行词法分析，由一长串字符分解成为多个更小的字符串单元，然后执行语法分析，把单词排列成为抽象语法树，到此解释器与编译器的行为都是一致的，在这之后编译器将转换为另外的语言，而解释器则将一边分析语法树一边执行。



第 2 章 Android Camera 系统框架



故不登高山，不知天之高也；不临深溪，不知地之厚也；不闻先王之遗言，不知学问之大也。

—荀子

2.1 模型描述

我们目前都是学生，接触 \LaTeX 的时间也不是很长，因此，对于此模板的错误还请多多包涵！目前，模板的拓展性或者可移植性有待完善，所以，我们强烈建议用户不要大幅修改模板文件，我们的初衷是提供一套模板，让初学者能够使用一些比较美观，优雅的模板。如果在使用过程中，想修改一些简单的东西需要帮忙，请联系我们，我们的邮箱是：elegantlatex2e@gmail.com。我们将竭尽全力提供帮助！

值此版本发行之际，我们 Elegant \LaTeX 项目组向大家重新介绍一下我们的工作，我们的主页是 <http://elegantlatex.tk>，我们这个项目致力于打造一系列美观、优雅、简便的模板方便使用者记录学习历史。其中目前在实施或者在规划中的子项目有书籍模板 ElegantBook、笔记模板 ElegantNote、幻灯片模板 ElegantSlide。这些子项目的名词是一体的，请在使用这些名词的时候不要将其断开（如 Elegant Note 是不正确的写法）。并且，Elegant \LaTeX Book 指的即是 ElegantBook。

2.2 代价函数

基于本模板追求视觉上的美观的角度，强烈建议使用者安装 `./fonts/` 文件夹下的字体。出于版权的考虑，务必不能将此模板用于涉及盈利目的的商业行为，否则，后果自负，本模板带的字体仅供学习使用，如果您喜欢某种字体，请自行购买正版。本文主要使用的字体如下

- Adobe Garamond Pro
- Minion Pro & Myriad Pro & Inconsolata
- 方正字体
- 华文中宋



Note: 中文正文使用了华文中宋, *Minion Pro* 为英文衬线字体 (`\rmfamily`), *Myriad Pro* 为英文非衬线字体 (`\sffamily`), *Inconsolata* 为英文打字机字体 (`\ttfamily`)。

并且, 如果系统内安装了 *Adobe* 字体, 大家可以把模板中使用到的黑体, 楷体, 宋体等字替换成 *Adobe* 字体, 这样可以达到最佳效果。

2.3 代价函数 (一)

2.3.1 编译方式

本模板基于 book 文类, 所以 book 的选项对于本模板也是有效的。但是, 只支持 $\text{Xe}_{\text{L}}\text{TeX}$, 编码为 UTF-8, 推荐使用 $\text{T}_{\text{E}}\text{Xlive}$ 编译。作者编写环境为 Win8.1(64bit)+ $\text{T}_{\text{E}}\text{Xlive}$ 2013, 由于使用了参考文献, 所以, 编译顺序为 $\text{Xe}_{\text{L}}\text{TeX} \rightarrow \text{Bib}_{\text{TeX}} \rightarrow \text{Xe}_{\text{L}}\text{TeX} \rightarrow \text{Xe}_{\text{L}}\text{TeX}$ 。

本文特殊选项设置共有 3 类, 分为颜色、数学字体以及章标题显示风格。

2.3.2 选项设置

第一类为颜色主题设置, 内置 3 组颜色主题, 分别为 green (默认), cyan, blue, 另外还有一个自定义的选项 nocolor, 用户必须在使用模板的时候选择某个颜色主题或选择 nocolor 选项。调用颜色主题 green 的方法为 `\documentclass[green]{elegantbook}` 或者使用 `\documentclass[color=green]{elegantbook}`。需要改变颜色的话请选择 nocolor 选项或者使用 `color=none`, 然后在导言区定义 main、seco、thid 颜色, 具体的方法如下:

```
\definecolor{main}{RGB}{70,70,70}    %定义main颜色值
\definecolor{seco}{RGB}{115,45,2}     %定义seco颜色值
\definecolor{thid}{RGB}{0,80,80}      %定义thid颜色值
```

第二类为数学字体设置, 有两个可选项, 分别是 mathpazo (默认) 和 mtpro2 字体, 调用 mathpazo 字体使用 `\documentclass[mathpazo]{elegantbook}`, 调用 mtpro2 字体时需要把 mathpazo 换成 mtpro, mathpazo 不需要用户自己安装字体, mtpro2 的字体需要自己安装。

第三类为章标题显示风格, 包含 hang (默认) 与 display 两种风格, 区别在于章标题单行显示 (hang) 与双行显示 (display), 本说明使用了 hang。调用方式为 `\documentclass[hang]{elegantbook}` 或者 `\documentclass[titlestyle=hang]{elegantbook}`。

综合起来, 同时调用三个选项使用 `\documentclass[color=X,Y,titlestyle=Z]{elegantbook}`。其中 X 可以选择 green,cyan,blue,none; Y 可以选择 mathpazo 或者 mtpro; Z 可以选择 hang 或者 display。



	green	cyan	blue	主要使用的环境
main	<div></div>	<div></div>	<div></div>	newthem newlemma newcorol
seco	<div></div>	<div></div>	<div></div>	newdef
thid	<div></div>	<div></div>	<div></div>	newprop

表 2.1: ElegantBook 模板中的三套颜色主题

2.3.3 数学环境简介

在我们这个模板中，定义了三大类环境

1. 定理类环境，包含标题和内容两部分。根据格式的不同分为 3 种
- newthem、newlemma、newcorol 环境，颜色为主颜色 main，三者编号均以章节为单位；

• newdef 环境，含有一个可选项，编号以章节为单位，颜色为 seco；

• newprop 环境，含有一个可选项，编号以章节为单位，颜色为 thid。
2. 证明类环境，有newproof、note、remark、solution 环境，特点是，有引导符和引导词，并且 newproof、solution 环境有结束标志。
3. 结论类环境，有conclusion、assumption、property 环境，三者均以粗体的引导词为开头，和普通段落格式一致。
4. 示例类环境— example、exercise环境，编号以章节为单位，其中 exercise 环境有引导符。
5. 自定义环境— custom，带一个必选参数，格式与 conclusion 环境很类似。

2.3.4 可编辑的字段

在模板中，可以编辑的字段分别为作者\author、\email、\zhtitle、\zhend、\entitle、\enend、\version。并且，可以根据自己的喜好把封面水印效果的cover.pdf 替换掉，以及封面中用到的logo.pdf。



2.4 代价函数 (二)

2.5 梯度下降

2.6 梯度下降知识点总结

2.7 线性回归的梯度下降

2.8 本章课程总结



第 3 章 Linux Camera 驱动框架



为往圣继绝学，为万世开太平！

—张载

3.1 矩阵和向量

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

考虑如下的随机动态规划问题

$$\begin{aligned} \max(\min) \quad & \mathbb{E} \int_{t_0}^{t_1} f(t, x, u) dt \\ \text{s.t.} \quad & dx = g(t, x, u)dt + \sigma(t, x, u)dz \\ & k(0) = k_0 \text{ given} \end{aligned}$$

where z is stochastic process or white noise or wiener process.

Definition 3.1 Wiener Process

If z is wiener process, then for any partition t_0, t_1, t_2, \dots of time interval, the random variables $z(t_1) - z(t_0), z(t_2) - z(t_1), \dots$ are independently and normally distributed with zero means and variance $t_1 - t_0, t_2 - t_1, \dots$



Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis

eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

Example 3.1: E and F be two events such that $\mathbf{P}(E) = \mathbf{P}(F) = 1/2$, and $\mathbf{P}(E \cap F) = 1/3$, let $\mathcal{F} = \sigma(Y)$, X and Y be the indicate function of E and F respectively. How to compute $\mathbb{E}[X \mid \mathcal{F}]$?

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

- ❖ **Exercise 3.1:** let $S = l^\infty = \{(x_n) \mid \exists M \text{ such that } \forall n, |x_n| \leq M, x_n \in \mathbb{R}\}$, $\rho_\infty(x, y) = \sup_{n \geq 1} |x_n - y_n|$, show that (l^∞, ρ_∞) is complete.

Theorem 3.1 勾股定理

勾股定理的数学表达 (Expression) 为

$$a^2 + b^2 = c^2$$

其中 a, b 为直角三角形的两条直角边长, c 为直角三角形斜边长。

📌 **Note:** 在本模板中, 引理 (lemma), 推论 (corollary) 的样式和定理的样式一致, 包括颜色, 仅仅只有计数器的设置不一样。在这个例稿中, 我们将不给出引理推论的例子。

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Proposition 3.1 最优性原理

如果 u^* 在 $[s, T]$ 上为最优解, 则 u^* 在 $[s, T]$ 任意子区间都是最优解, 假设区间为 $[t_0, t_1]$ 的最优解为 u^* , 则 $u(t_0) = u^*(t_0)$, 即初始条件必须还是在 u^* 上。



Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellen-tesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Corollary 3.1

假设 $V(\cdot, \cdot)$ 为值函数, 则跟据最大值原理, 有如下推论

$$V(k, z) = \max \left\{ u(zf(k) - y) + \beta \mathbb{E}V(y, z') \right\}$$

Proof: 因为 $y^* = \alpha\beta zk^\alpha$, $V(k, z) = \alpha/1 - \alpha\beta \ln k_0 + 1/1 - \alpha\beta \ln z_0 + \Delta$.

$$\begin{aligned} \text{右边} &= \left\{ u(zf(k) - y) + \beta \mathbb{E}V(y, z') \right\} \\ &= \ln(zk^\alpha - \alpha\beta zk^\alpha) + \beta \mathbb{E} \left[\frac{\alpha}{1 - \alpha\beta} \ln y + \frac{1}{1 - \alpha\beta} \ln z' + \Delta \right] \\ &= \ln(1 - \alpha\beta)zk^\alpha + \beta \left\{ \mathbb{E} \left[\frac{\alpha}{1 - \alpha\beta} \ln \alpha\beta zk^\alpha \right] + \frac{1}{1 - \alpha\beta} \mathbb{E}[\ln z'] + \Delta \right\} \end{aligned}$$

利用 $\mathbb{E}[\ln z'] = 0$, 并将对数展开得

$$\begin{aligned} \text{右边} &= \ln(1 - \alpha\beta) + \ln z + \alpha \ln k + \frac{\alpha\beta}{1 - \alpha\beta} [\ln \alpha\beta + \ln z + \alpha \ln k] + \frac{\beta}{1 - \alpha\beta} \mu + \beta \Delta \\ &= \frac{\alpha}{1 - \alpha\beta} \ln k + \frac{1}{1 - \alpha\beta} \ln z + \Delta \end{aligned}$$

所以左边 = 右边, 证毕。 □

Properties: Properties of Cauchy Sequence

1. $\{x_k\}$ is cauchy sequence then $\{x_k^i\}$ is cauchy sequence.
2. $x_k \in \mathbb{R}^n$, $\rho(x, y)$ is Euclidean, then cauchy is equivalent to convergent, (\mathbb{R}^n, ρ) metric space is complete.



Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

✂ **Application:** This is one example of the custom environment, the key word is given by the option of custom environment.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Definition 3.2 Contraction mapping

(S, ρ) is the metric space, $T : S \rightarrow S$, If there exists $\alpha \in (0, 1)$ such that for any x and $y \in S$, the distance

$$\rho(Tx, Ty) \leq \alpha \rho(x, y) \quad (3.1)$$

Then T is a contraction mapping.

✿ Remarks:

1. $T : S \rightarrow S$, where S is a metric space, if for any $x, y \in S$, $\rho(Tx, Ty) < \rho(x, y)$ is not contraction mapping.
2. Contraction mapping is continuous map.

Conclusions: 看到一则小幽默，是这样说的：别人都关心你飞的有多高，只有我关心你的翅膀好不好吃！说多了都是泪啊！



3.2 加法和标量乘法

3.3 矩阵向量乘法

3.4 矩阵乘法

3.5 矩阵乘法特征

3.6 逆和转制



第 4 章 全志 A83t Camera 开发



纸上得来终觉浅, 绝知此事要躬行。

—傅玄

4.1 总线与硬件接口

根据全志官方提供的《A83t_User_Manual》和《A83tDatasheet》两个文档，可以知道 A83t 支持 MIPI-CSI 总线和 CSI 总线，且 A83t 内部支持 ISP，A83t 体系架构图如图 4.1 所示。

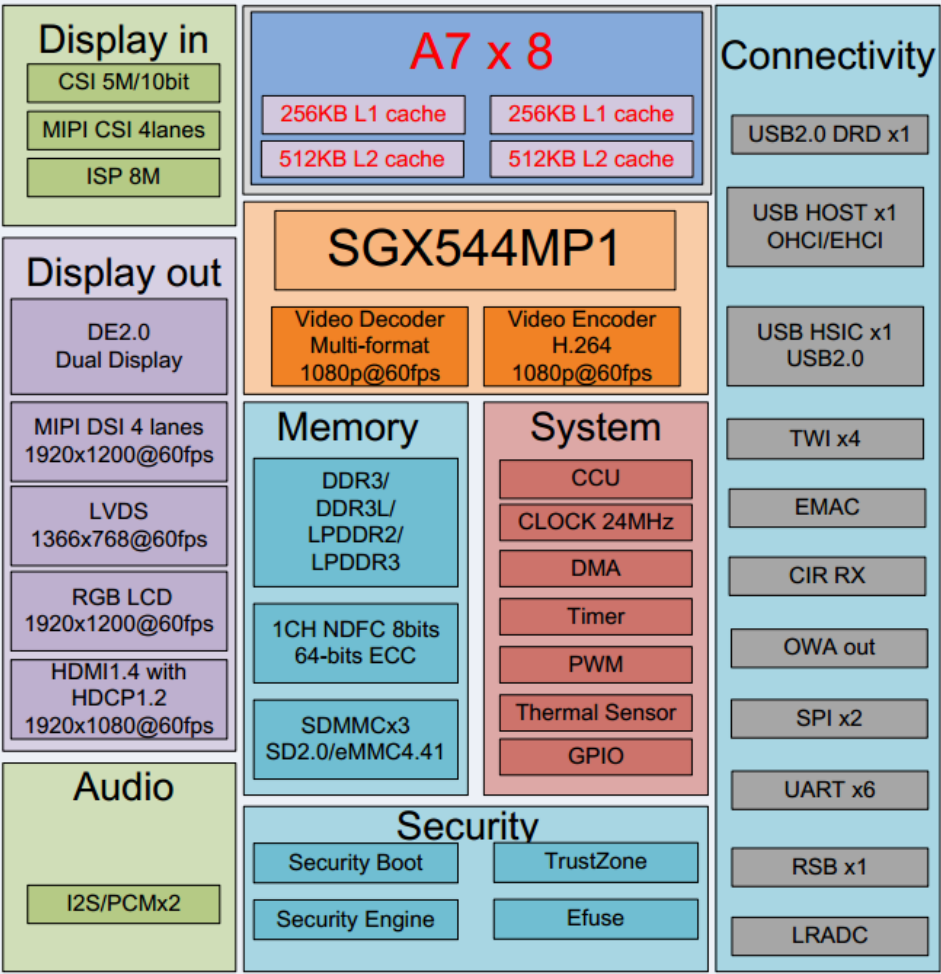


图 4.1: A83t 体系架构

CSI 接口的特性如图 4.2,

- CSI
- Support 10-bits parallel camera sensor
 - Support up to 5M pixel camera sensor
 - Support video shot up to 720p@30fps
 - Support CCIR656 protocol for NTSC and PAL

图 4.2: CSI 接口特性

MIPI-CSI 接口的特性如图 4.3,

- MIPI CSI
- Compliant with MIPI-CSI2 V1.00 and MIPI DPHY V1.00.00
 - 4-lane MIPI CSI
 - Up to 1Gbps per Lane in HS Transmission
 - Maximum to 8M@30fps with 4 data lane
 - Support video shot up to 1080p@60fps
 - Supports format: YUV422-8bit/10bit,YUV420-8bit/10bit,RAW-8,RAW-10, RAW-12,RGB888,RGB565

图 4.3: MIPI-CSI 接口特性

4.2 硬件原理图

在调试的开发板上采用的是 gc2145+gc0329 连接到 A83t 的 CSI0 接口,gc2145 器件原理图如图 4.4 所示,

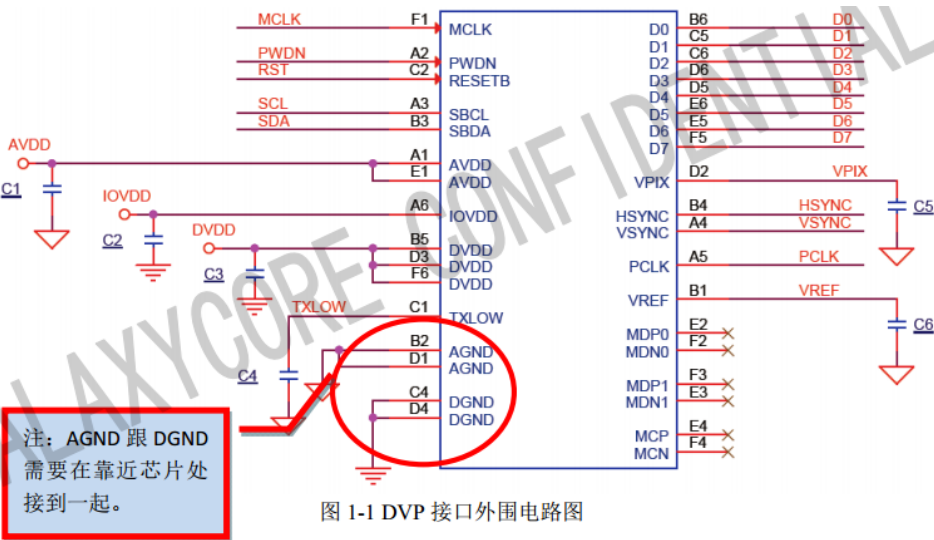


图 4.4: GC2145 器件原理图



4.3 驱动源码

4.4 内核与驱动配置

调试 camera 需要配置内核与 sysconfig.fex 文件，内核配置如图 4.6 所示。

```
1. Device Drivers-><*>Multimedia support-><*>Video For linux
2. Device Drivers-><*>Multimedia support->[*]Video capture adapters ->[*]V4L platform
   devices-><M>sunxi video front end(camera and etc)driver
3. Device Drivers-><*>Multimedia support->[*]Video capture adapters ->[*]V4L platform
   devices-><M>v4l2 driver for SUNXI
```

图 4.5: make menuconfig 配置

除了基本的内核对 camera 的支持，还需要修改内核 Makefile 添加爱 gc2145 驱动的编译，修改文件路径: linux-3.4/drivers/media/video/sunxi-vfe/device/Makefile, 修改内容如图 4.7 所示，在 24 行添加了 gc2145sensor 模组的驱动编译，

```
20 obj-m          += gc0307.o
21 obj-m          += gc0308.o
22 obj-m          += gc0328.o
23 obj-m          += gc0329.o
24 obj-m          += gc2145.o
25 obj-m          += gc0311.o
26 obj-m          += hi253.o
27 obj-m          += sp2518.o
28 obj-m          += sp2519.o
```

图 4.6: 添加 GC2145 驱动

完成内核配置之后还需要对 android 系统进行配置，让 android 系统在启动时加载对应的驱动模块, 对应的修改目录: android/device/softwinner/octopus-f1/init.sun8i.rc, 修改内容如图 4.8 所示, 添加了加载 gc2145.ko 模块，去掉了 ov5640 模块的加载。



```
83 #csi module
84 insmod /system/vendor/modules/videobuf-core.ko
85 insmod /system/vendor/modules/videobuf-dma-contig.ko
86 insmod /system/vendor/modules/cci.ko
87 insmod /system/vendor/modules/actuator.ko
88 insmod /system/vendor/modules/vfe_os.ko
89 insmod /system/vendor/modules/vfe_subdev.ko
90 insmod /system/vendor/modules/dw9714_act.ko
91 # insmod /system/vendor/modules/ov5640.ko
92 insmod /system/vendor/modules/gc2145.ko
93 insmod /system/vendor/modules/gc0308.ko
94 insmod /system/vendor/modules/vfe_v4l2.ko
```

图 4.7: 加载 GC2145 驱动

4.5 调试问题记录

将调试中遇到的一些问题记录如下：

- 1) 将 sysconfig.fex 文件配置好之后，编译固件下载运行，在 linux 加载时出现卡住无法启动的问题。

```
2473 case CSI_SUBDEV_PWR_ON:
2474     vfe_dev_dbg("CSI_SUBDEV_PWR_ON\n");
2475     //power on reset
2476     vfe_gpio_set_status(sd,PWDN,1);//set the gpio to output
2477     vfe_gpio_set_status(sd,RESET,1);//set the gpio to output
2478     //active mclk before power on
2479     vfe_set_mclk_freq(sd,MCLK);
2480     vfe_set_mclk(sd,ON);
2481     //standby off io
2482     vfe_gpio_write(sd,PWDN,CSI_STBY_OFF);
2483     //reset on io
2484     vfe_gpio_write(sd,RESET,CSI_RST_ON);
2485     //power supply
2486     vfe_gpio_write(sd,POWER_EN,CSI_PWR_ON);
2487     vfe_set_pmu_channel(sd,IOVDD,ON);
2488     vfe_set_pmu_channel(sd,AVDD,ON);
2489     vfe_set_pmu_channel(sd,DVDD,ON);
2490     vfe_set_pmu_channel(sd,AFVDD,ON);
2491     usleep_range(10000,12000);
2492     //reset after power on
2493     vfe_gpio_write(sd,RESET,CSI_RST_OFF);
2494     //usleep_range(30000,31000);
2495     vfe_dev_dbg("CSI_SUBDEV_PWR_ON END\n");
2496     break;
```

图 4.8: 定位代码



解决方法：在各个驱动里面添加打印，定位具体是什么位置导致了 CPU 死掉，初步判断是有空指针之类的问题，但是内核栈没有更多的信息打印出来。还想到一个办法，便是通过将 android 中设置的自动加载 camera 各个驱动，换成手动加载驱动，看看是如何挂掉的。添加打印后和换成手动加载驱动之后定位到挂掉的代码位置，代码路径 linux-3.4/drivers/media/video/sunxi-vfe/device/gc2145.c, 如图 4.8 所示, 最后定位到了 vfe_set_pmu_channel(sd, IOVDD, ON) 这一句，原来是在 sys_config.fex 中 iovdd-csi 位置设置错误，放到了 aldo2 下面，aldo2 是控制 sdram 和 pll 的电源。

完成电源配置之后，摄像头便可以正常启动了, 此时的 sys_config.fex 的电源分配如图 4.9 所示, 将 iovdd-csi 分配到了 axp81x_dldo3, 将 dvdd-csi-18 分配在了 axp81x_eldo1, avdd-csi 分配在了 axp81x_dldo4。

```

1474 pmu1 is pmu81x
1475 regulator tree
1476 -----
1477 [pmu1_regu]
1478 regulator_count = 23
1479 regulator1 = "axp81x_dcdc1 none vcc-nand vcc-nor vcc-usb0-33 vcc-card vcc-io vcc-wifi-io vcc-io1 vcc-io2 vcc-sensor vcc-camera-33"
1480 regulator2 = "axp81x_dcdc2 none vdd-cpus"
1481 regulator3 = "axp81x_dcdc3 none vdd-cpub"
1482 regulator4 = "axp81x_dcdc4 none vdd-gpu"
1483 regulator5 = "axp81x_dcdc5 none vcc-dram"
1484 regulator6 = "axp81x_dcdc6 none vdd-sys vdd-usb0-09 vdd-hdmi-09"
1485 regulator7 = "axp81x_dcdc7 none"
1486 regulator8 = "axp81x_rtc none"
1487 regulator9 = "axp81x_eldo1 none vcc-dsi-18 vcc-hdmi-18"
1488 regulator10 = "axp81x_aldo2 none vdd-drampll vdd-lpddr-18 vcc-pll vcc-adc vcc-cpvdv vcc-ldoin"
1489 regulator11 = "axp81x_aldo3 none vcc-avcc vcc-pl"
1490 regulator12 = "axp81x_dldo1 none vcc-vibrator dvdd-ephy"
1491 regulator13 = "axp81x_dldo2 none"
1492 regulator14 = "axp81x_dldo3 none iovdd-csi afvcc-csi"
1493 regulator15 = "axp81x_dldo4 none vcc-ephy-io vcc-pd avdd-csi"
1494 regulator16 = "axp81x_eldo1 1 dvdd-csi-18"
1495 regulator17 = "axp81x_eldo2 1 vcc-csi2-18 vcc-lvds-18 vcc-efuse-18"
1496 regulator18 = "axp81x_eldo3 1 dvdd-csi3-18"
1497 regulator19 = "axp81x_fldo1 none avdd-ephy vcc-hsic-12"
1498 regulator20 = "axp81x_fldo2 none vdd-cpus"
1499 regulator21 = "axp81x_gp1o1ldo none vcc-ctp"
1500 regulator22 = "axp81x_gp1o1ldo none"
1501 regulator23 = "axp81x_dci1w 1 vcc-lcd-0"

```

图 4.9: sys_config 电源分配

iovdd 是摄像头模块 io 接口电源, avdd 是摄像头模块模拟电路电源, dvdd 是摄像头模块数字电路电源, 具体电源配置电压如图 4.10 所示,

2) 驱动挂死

解决方法：

4.6 总结



```
43 ;-----
44 ;
45 ; 各路电压输出语法说明:
46 ;
47 ; 电压名称 = 100XXXX : 表示把该路电压设置为XXXX指定的电压值, 同时打开输出开关
48 ; 电压名称 = 000XXXX : 表示把该路电压设置为XXXX指定的电压值, 同时关闭输出开关, 当有需要时由内核驱动打开
49 ; 电压名称 = 0 : 表示关闭该路电压输出开关, 不修改原有的值
50 ; Note:eldo2 before dldo4 > 100us; first off dldo4 then eldo2
51 ;-----
52 [power_sply]
53 dcdc1_vol          = 1003100
54 dcdc2_vol          = 1000900
55 dcdc3_vol          = 1000900
56 dcdc4_vol          = 1000900
57 ;dcdc5_vol         = 1001500
58 dcdc6_vol          = 1000900
59 aldo1_vol          = 1002800
60 aldo2_vol          = 1001800
61 aldo3_vol          = 1003000
62 eldo1_vol          = 1001800
63 eldo2_vol          = 1001800
64 eldo2_vol          = 1001800
65 dldo1_vol          = 3100
66 dldo2_vol          = 0003100
67 dldo3_vol          = 1002800
68 dldo4_vol          = 1002800
69 flldo2_vol         = 1000900
70 gpio0_vol          = 1003100
71 gpio1_vol          = 3100
```

图 4.10: sys_config 电源电压配置



第 5 章 多变量线性回归



5.1 多功能

5.2 多元梯度下降法

5.3 多元梯度下降法演练 I-特征缩放

5.4 多元梯度下降法演练 II-学习率

5.5 特征和多项式回归

5.6 正规方程 (区别于迭代方法的直接解决)

5.7 正规方程在矩阵不可逆情况下的解决方法

5.8 完成并提交编程作业

参考文献



- [1] 胡大伟, “ $\text{\LaTeX} 2_{\epsilon}$ 完全学习手册,” 2011.