



ETWS-1810 雷达信处

软件设计讨论

部 门： 研发部

编 制： 罗敏

版 本： 1.0

使用 L<sup>A</sup>T<sub>E</sub>X 撰写于 2018 年 12 月 25 日

## 摘 要

本文主要是关于雷达系统自研信号处理分系统软件设计讨论，目的是确定信号处理分系统软件和算法处理部分的详细工作。因对于信号处理部分均是从零起步，一个好的实现方案和计划更显的尤为重要，本文从信号处理分系统输出目标出发，逐步详细阐述各个设计目标的计算和实现过程。

**关键字：** 雷达系统 信号处理 算法 实现

# 目录

1 设计目标 .....	1
1.1 输出指标 .....	1
1.2 硬件架构 .....	1
2 设计原理 .....	3
2.1 数据模型 .....	3
2.1.1 三维数据模型 .....	3
2.1.2 数据模型理解 .....	4
2.2 指标算法 .....	5
2.2.1 反射率强度 .....	5
2.2.2 径向速度 .....	8
2.2.3 速度谱宽 .....	11
3 设计实现 .....	12
3.1 数据获取 .....	12
3.1.1 数据接收 .....	12
3.1.2 数据存储 .....	12
3.2 数据处理 .....	15
4 详细设计 .....	17
4.1 协议设计 .....	17
4.2 软件框架 .....	18
5 总结 .....	20
参考文献 .....	20

# 第一章 设计目标

## 1.1 输出指标

本节主要是对雷达信号处理分系统的设计指标进行确认，确定设计输出的指标之后，才能够更好的针对设计指标进行软件设计和规划。在图 1.1 中根据《X 波段双偏振天气雷达改

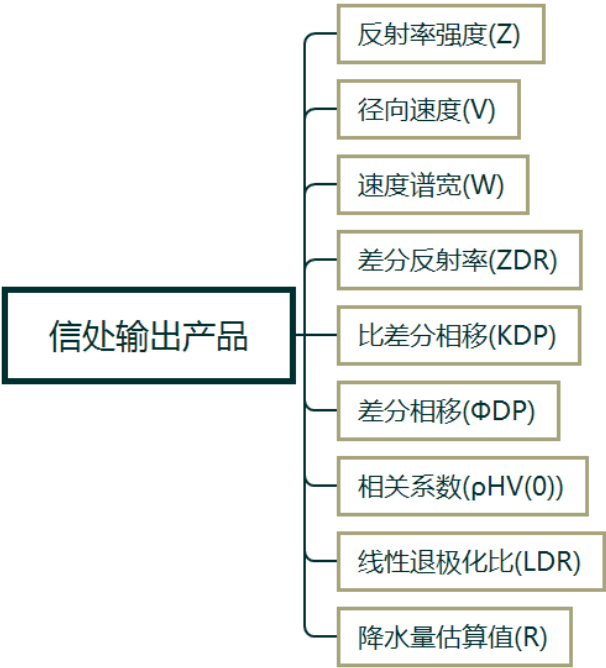


图 1.1 信处输出原始产品

造升级项目立项报告》列了一下信处分系统应该要输出的原始产品。

## 1.2 硬件架构

目前硬件架构已经基本确认，具体如图 1.2 所示主要采用 AD 板 + 接口板 +X86 板的方案。AD 板与接口板之间采用 SIRO 接口连接，接口板与 X86 板之间采用 PCIe3.0 接口连接。如图所示，AD 板主要负责下变频、抽取、滤波等工作，接口板则负责接收 AD 板采集的数据，然后对数据进行 DBF、脉冲压缩等处理，同时需要控制伺服、波控等组件,X86 板则接收接口板处理后的数据做后半段的信号数据处理，最终输出上一节中提到的输出指标。

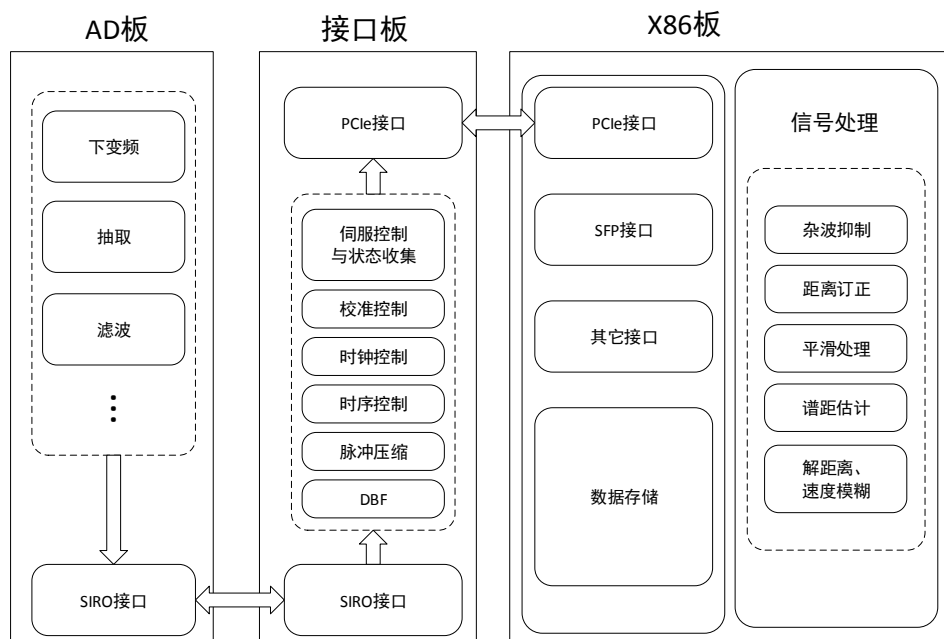


图 1.2 硬件设计

因本文主要讨论软件的设计实现，主要会聚焦于 X86 板的信号处理部分，目前接口板与 X86 板的数据通道已经是对接过，可以从接口板读取相关数据，后续的工作便是两个主要的工作：

- 设计接口板与 X86 板接收数据的协议
- 处理接收到的数据

## 第二章 设计原理

### 2.1 数据模型

#### 2.1.1 三维数据模型

需要对雷达回波进行数据处理，首先要将回波数据抽象成一个数据处理模型，在这里采用了三维数据块作为一个数据处理模型<sup>[1]</sup>，这一处理方法引用自雷达信号处理基础第三章，基本处理模型如图 2.1 所示：

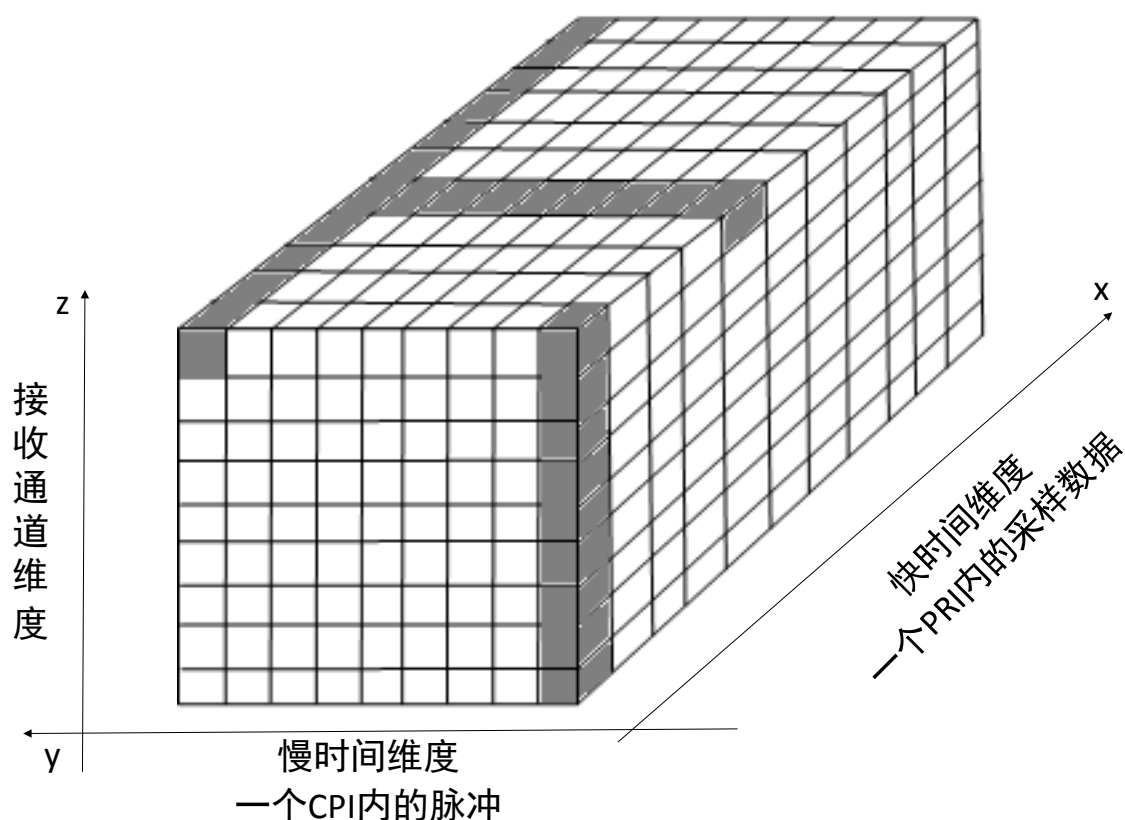


图 2.1 信号处理数据块

可以看到，在上图中的  $x$  轴方向标注为快时间维度，本质就是一个 PRI(脉冲重复间隔) 内的回波采样数据。 $y$  轴标注为慢时间维度，本质是一个 CPI(相干处理间隔) 内的多组回波数

据。 $z$  轴标注的是接收通道维度，本质就是对应天线的接收通道数目。

### 2.1.2 数据模型理解

上一节是介绍了基本的数据模型，在本节中还需要深入的理解一下设计的数据模型，在图 2.2 中，针对数据模型的三个维度进行了进一步的标注。

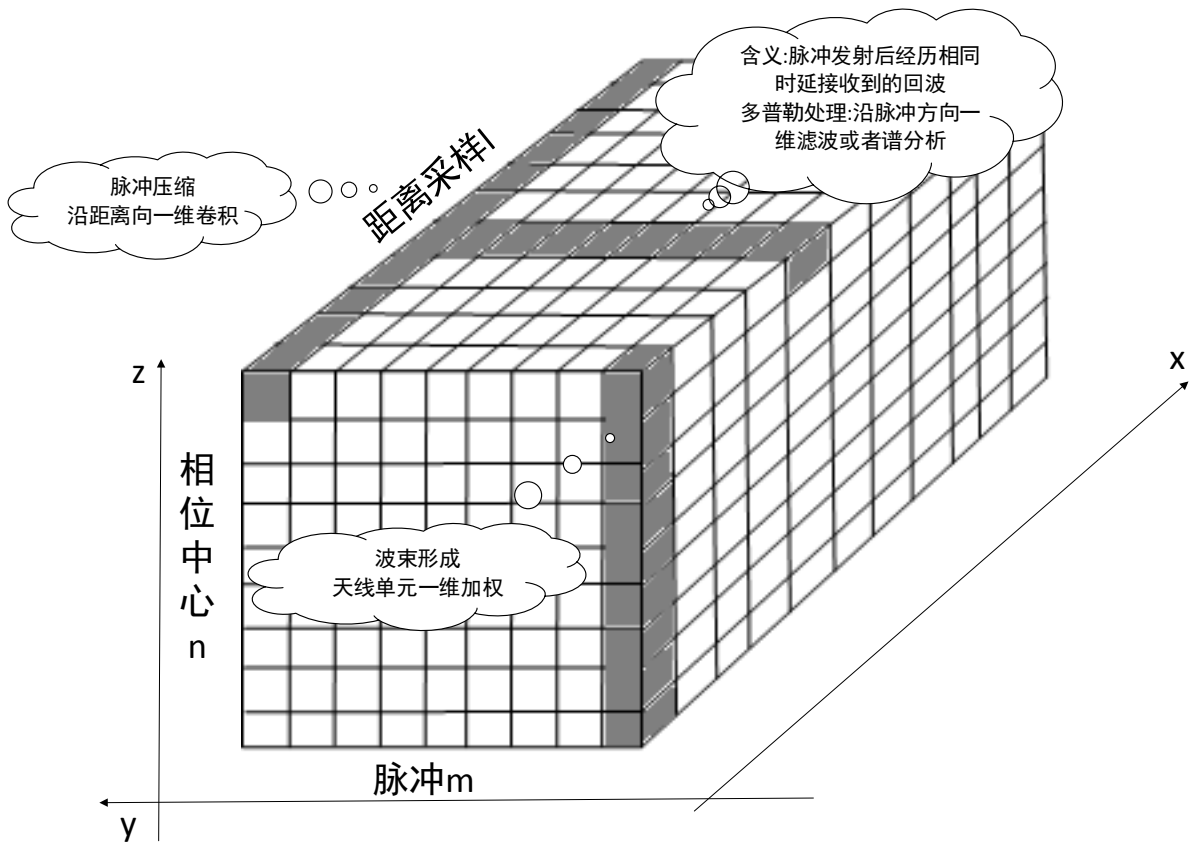


图 2.2 信号处理数据块

$x$  轴方向标注为距离采样 1，脉冲压缩处理便是在这一维度做处理，而多普勒处理则是沿  $y$  轴方向， $y$  轴方向还有一个很重要的含义，便是脉冲发射后经历相同时延接收到的回波， $z$  轴代表不同的接收通道，不同的接收通道具有不同的相位，波束赋形便是在  $z$  轴方向来处理的。

在这里我们将一个接收通道的采样数据拿出来详细研究一下，如图 2.3 所示，将  $x$  轴和  $y$  轴方向的数据转换到时间轴，一个距离库对应一个小方块，同时对应的就是一个采样周期内的采样数据，一系列数据便是一个 PRT 内采集到的数据，而一个 CPI 内的数据，则是多个 PRT 内的数据采集到一起进行数据处理。

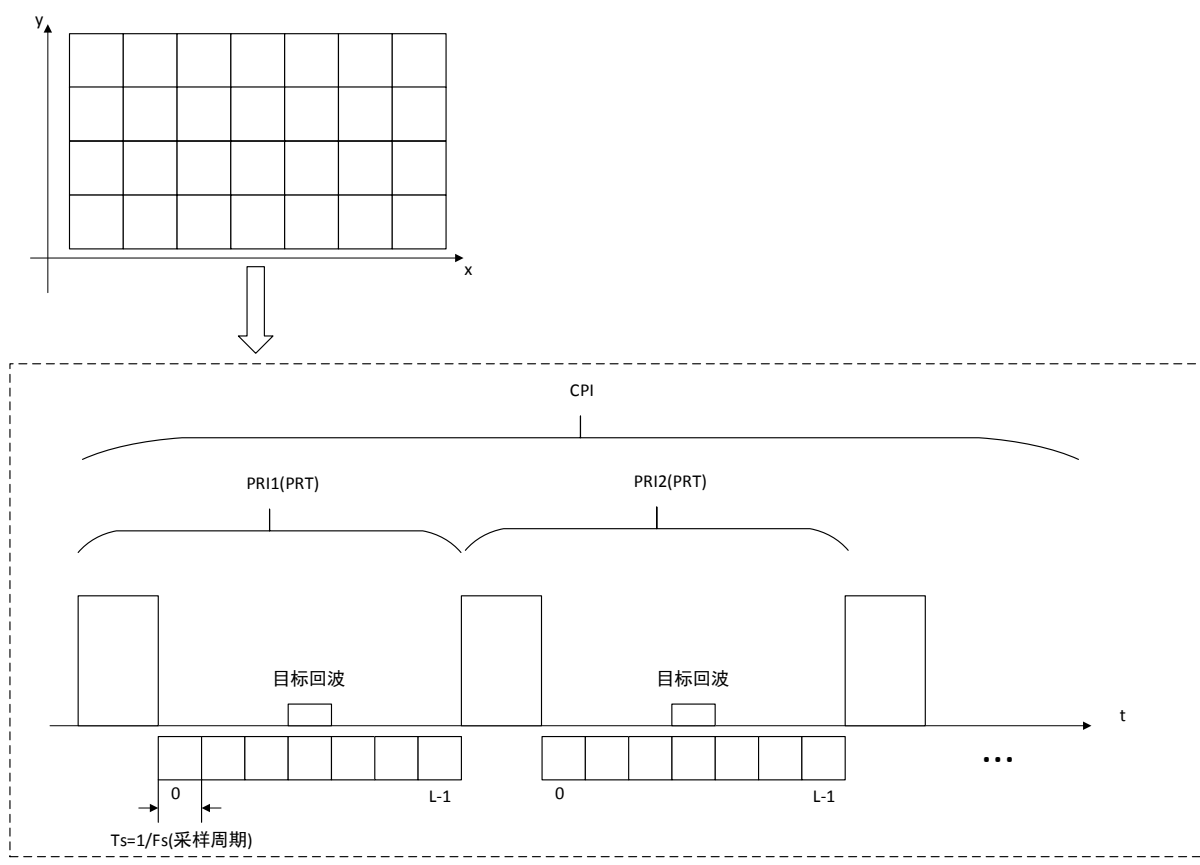


图 2.3 CPI 数据

根据上面两节的叙述基本建立了一个数据处理的模型，后面的计算和处理均会在这个三维模型来进行展开。

## 2.2 指标算法

有了数据处理模型，还需要知道如何处理这个数据模型，现在就像做菜，菜的原材料已经有了，但是还需要食谱，然后根据食谱做出想吃的菜，本节就是描述如何处理这些原材料。首先需要面对的就是 3 个最基本的指标，反射率强度 (Z)，径向速度 (V)，速度谱宽 (W)，下面就逐个指标来进行计算，其它指标后续逐步添加进入方案。

### 2.2.1 反射率强度

反射率强度 (Z) 是雷达回波的最基本指标，在这里主要引用《雷达信号处理基础 (第二版)》<sup>[1]</sup> 中 2.2.4 节中的计算模型，雷达气象学通常采用称为反射率的归一化因子来表示气象



目标的散射特性，通常用  $Z$  表示，体反射率 (RCS) 公式如下：

公式 1 (体反射率)

$$\eta = \frac{\pi^5 |K|^2}{\lambda} Z \quad (2.1)$$

$|K|^2$ : 对于由水构成的散射体其值近似为 0.93，由冰构成的散射体近似为 0.197

$\lambda$ : 雷达波长

当给出一个回波功率测量值，就可以用雷达方程估计出  $\eta$ ，然后利用公式 2.1 换算成  $Z$ ，再对取  $10\lg Z$  即得到 dBZ。

《雷达信号处理基础 (第二版)》<sup>[1]</sup> 中 2.2.2 节给出的体散射目标的雷达方程如下：

公式 2 (体散射雷达方程)

$$p_r(t_0) = \frac{P_t G^2 \lambda^2 \eta \Delta R \theta_3 \phi_3}{(4\pi)^3 R_0^2 L_s L_a(R_0)} \quad (2.2)$$

$p_r(t_0)$ :  $t_0$  时刻回波接收功率

$P_t$ : 雷达辐射功率

$G$ : 天线增益

$\lambda$ : 雷达波长

$\Delta R$ : 一个距离分辨单元的长度

$\theta_3$ : 方位 3dB 波束宽度

$\phi_3$ : 俯仰 3dB 波束宽度

$R_0$ : 分辨单元中心距离

$L_s$ : 系统损耗因子

$L_a(R_0)$ :  $R_0$  处的大气衰减

根据公式 2.1 和 2.2 便可以解出  $Z$ ，综合后的公式如下：

公式 3 (体散射雷达方程)

$$Z = \frac{64}{\pi^2} * \frac{L_s L_a(R_0)}{P_t G^2 \lambda |K|^2 \Delta R \theta_3 \phi_3} * R_0^2 P_r(t_0) \quad (2.3)$$

根据公式 2.3 可以看出只要从上一节中的距离库获取  $R_0$  和  $P_r(t_0)$  便可以解出  $Z$  的值，公式中的其它部分则可以根据相关的系统属性计算为一个常数。

$R_0$  的值是比较好计算的，直接使用下面的公式：

公式 4 (目标距离计算)

$$R_0 = \frac{c * t_0}{2} \quad (2.4)$$

$c$ : 光速

$t_0$ : 接收机接收回波的时间

下面重点介绍一下  $P_r(t_0)$  的计算，为了得到更好的信噪比，回波功率需要在一个 CPI 上来计算信号功率，相干积累可以增加信噪比主要源于下面三个公式。

公式 5 (相干积累)

$$\chi_1 = \frac{A^2}{\sigma_w^2} \quad (2.5)$$

$$\sum_{n=0}^{N-1} \{Ae^{j\phi} + w[n]\} = NAe^{j\phi} + \sum_{n=0}^{N-1} w[n] \quad (2.6)$$

$$\chi_N = \frac{(NA)^2}{N\sigma_w^2} = N\chi_1 \quad (2.7)$$

$\chi_1$ : 单脉冲的 SNR

$A$ : 复回波  $Ae^{j\phi}$  的幅度值

$\Sigma_w^2$ : 高斯白噪声的信号功率

$w[n]$ : 噪声信号

$\chi_N$ :  $N$  个相干信号累计的 SNR

根据相干积累公式中的原理，采样得到的总功率等于有效回波信号的功率加上噪声功率，而每个距离库采样信号的值表示为  $I+jQ$  两路的数据，总功率可以通过  $\sqrt{I^2 + Q^2}$  来计算。又根据相干积累中信噪比可以提升  $N$  倍，这样有利于判断是否接收到的信号为有效的回波，当然为了计算接收回波的功率，还需要计算噪声的功率。

噪声功率的计算方法: 取最大俯仰角的接收通道, 最远距离的 256 个距离库计算平均功率。

有了总功率和噪声功率, 很自然便可以得到回波信号的功率  $P_r(t_0)$ 。

### 2.2.2 径向速度

径向速度的计算主要在慢时间维度 (即数据块的 y 轴方向) 进行处理, 处理的原理主要是基于多普勒定律, 针对我司的气象雷达信号处理中多普勒处理主要有以下两个方法:

#### 1) MTI(运动目标指示)

MTI 处理的目的主要是设计一个高通滤波器滤除杂波, 如图 2.4 所示,

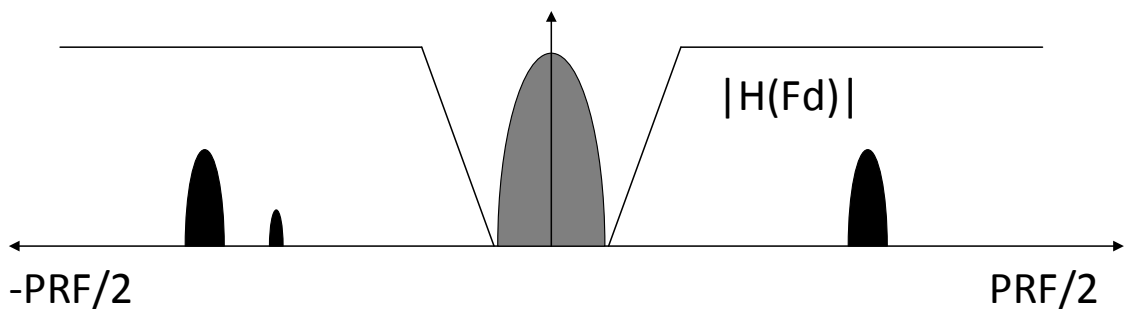
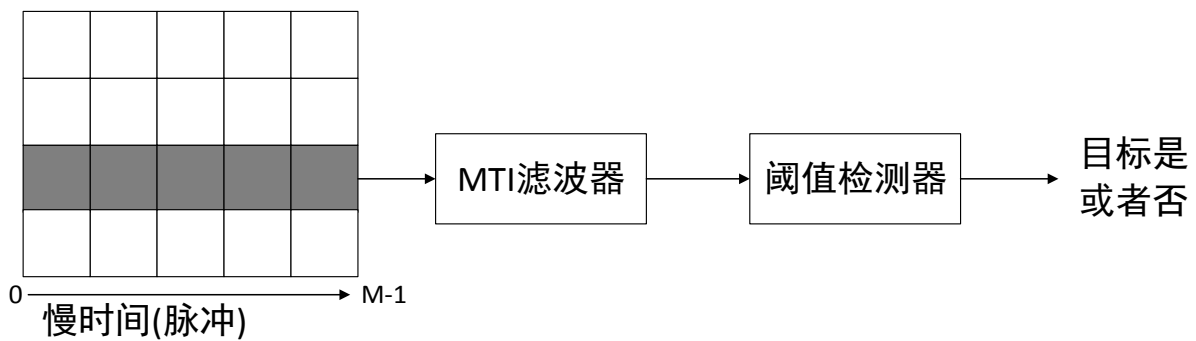


图 2.4 MIT 处理

#### 2) 脉冲对处理 (PPP)<sup>[1][2] [4]</sup>

在气象雷达中脉冲对处理主要是为了计算目标谱峰的参数, 速度计算如公式 6 所示, 处理原理主要是依据目标速度比较慢, 频谱偏移比较小, 采用计算相位的办法更为合适,

具体的阐述可以看引用网页<sup>[3]</sup>, 在上面做了比较清晰的解释。

具体的推导原理就是根据图 2.5。

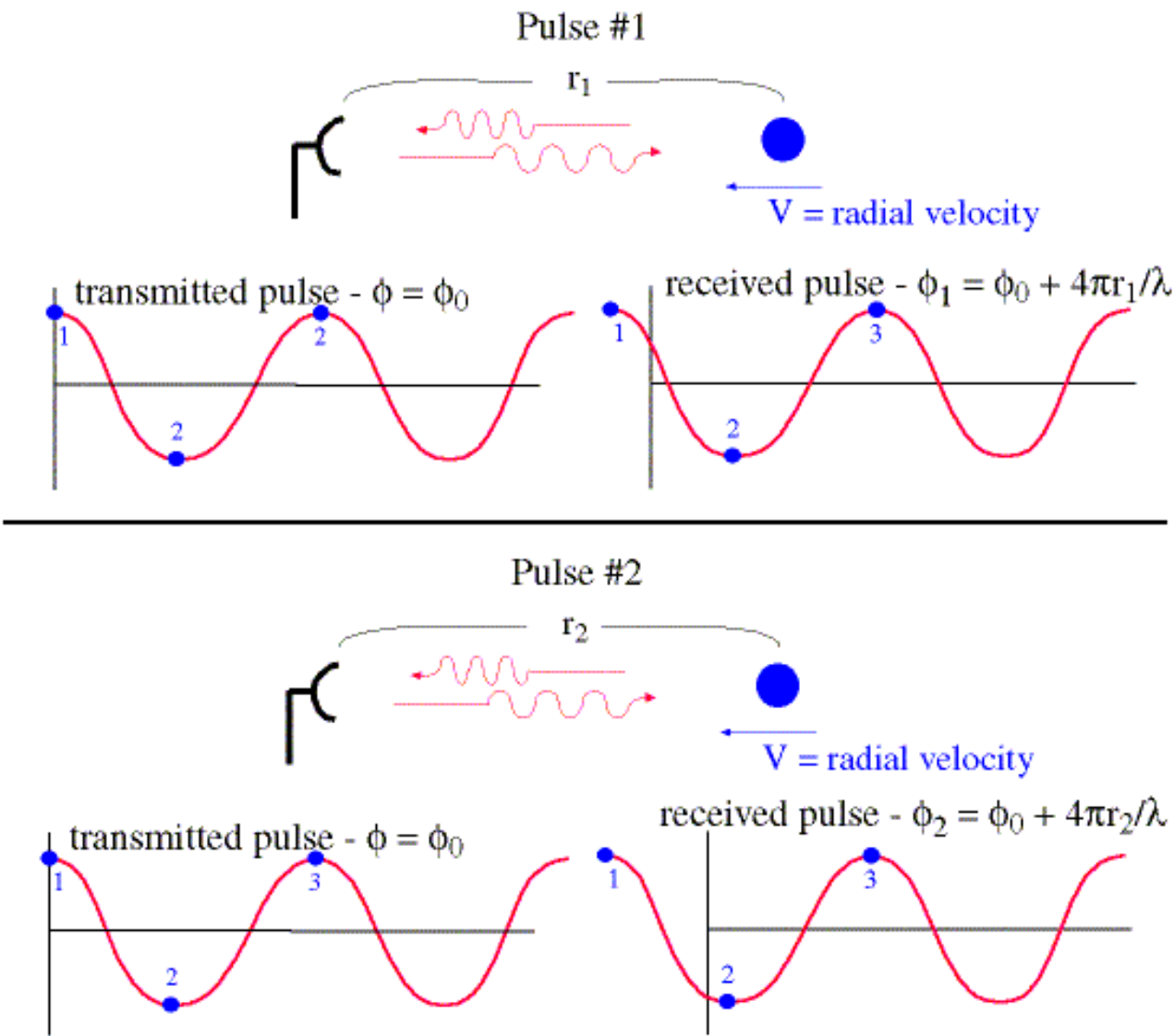


图 2.5 相位计算

根据图中所示，做如下推导：

## 公式 6 (相位计算推导公式)

$$\phi = \phi_0 + \frac{4\pi r}{\lambda} \quad (2.8)$$

$\phi_0$ : 发射信号的相位

$\phi$ : 接收信号的相位

等式两边做微分可得:

$$\frac{d\phi}{dt} = \frac{4\pi}{\lambda} \frac{dr}{dt} \quad (2.9)$$

$dt$ : 就是两个采样之间的时间间隔  $T$

$\frac{dr}{dt}$ : 就是速度  $V$

$$V = \frac{\lambda}{4\pi T} d\phi \quad (2.10)$$

在这里主要希望计算的是相位的变化。

公式 7 (速度计算公式<sup>[4]</sup>)

$$R[T] = \frac{1}{N} \sum_{k=0}^{N-1} s[k+1]s^*[k] \quad (2.11)$$

$$V_0 = \frac{\lambda}{4\pi T} \arg \{R(T)\} \quad (2.12)$$

$R(T)$ : 相邻两个脉冲的自相关函数

$s_k$ : 复通向 (I) 和正交 (Q) 信号样本

$V_0$ : 径向速度

$P_n$ : 噪声功率

$T$ : 连续两次取样的时间间隔

$\lambda$ : 波长

### 2.2.3 速度谱宽

计算出径向速度之后，也可以使用自相关函数计算出谱宽，谱宽计算在这里直接采用雷达手册中给出的公式，如公式 7 所示。

公式 8 (谱宽计算公式<sup>[4]</sup>)

$$\sigma^2 \approx \frac{\lambda}{8\pi^2 T^2} \arg \left\{ 1 - \frac{R(T)}{R(0) - P_n} \right\} \quad (2.13)$$

$R(T)$ : 相邻两个脉冲的自相关函数

$R(0)$ : 相当于将自己与自己的复共轭相乘之后除以  $N$

$s_k$ : 复通向 (I) 和正交 (Q) 信号样本

$V_0$ : 径向速度

$T$ : 连续两次取样的时间间隔

$\lambda$ : 波长

## 第三章 设计实现

上一章主要讨论了雷达回波几个关键参数的处理原理，这一章主要讲的是结合硬件设备的软件实现。首先需要考虑的问题便是数据的获取，其次便是如何处理数据。

### 3.1 数据获取

#### 3.1.1 数据接收

在第一章的硬件框图中可以看到，x86 板主要通过 PCIe 接口从 FPGA 接收数据，接收数据的过程中，主要需要考虑的就是接受的数据的数据量大小，而接收数据量大小又和发射脉冲、接收通道等有关，下面列出了一个单通道一个 CPI 数据量评估公式：

公式 9 (数据量算公式)

$$Data = 4L_{dist} * C_{prt} + C_{prt} * H \quad (3.1)$$

4: 表示需要 4 个字节用来存放 IQ 两路采样数据

$L_{dist}$ : 距离库数目，根据探测距离决定

$C_{prt}$ : PRT 个数，在我司研发的系统中采用的是 64 个 PRT

$H$ : 每个距离库的头部信息字节数

这里可以大概根据公式评估数据量，假设  $L_{dist}$  值为 1500， $H$  为 64 字节，则大致可以计算出一个 CPI 内单通道数据为 379KByte。

目前 PCIe 驱动已经完成了调试，接收数据通道目前基本已经打通。

#### 3.1.2 数据存储

完成数据的接收之后，主要考虑的是数据要按照计算要求进行存储，以便于进行数据处理。讨论到计算处理，需要先来看一下计算部分的物理结构。根据之前的计划，是采用 GPU 卡来加速信号处理计算，目前所使用的 GPU 卡内部结构基本如图 3.1<sup>[5]</sup> 所示：

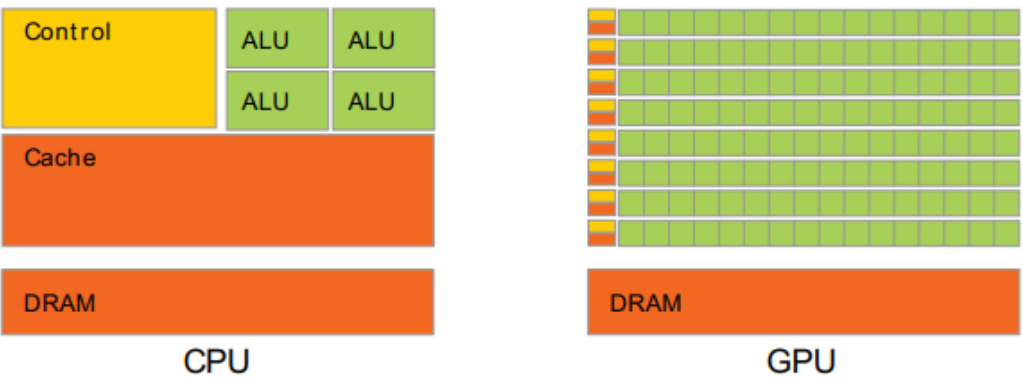


图 3.1 GPU 结构

GPU 有独立的内存和网状结构的个内部处理单元可用于并行处理，基本的结构如图 3.2 所示，一个小方块就是代表一个处理单元叫做 Streaming Multiprocessors(SMs)。

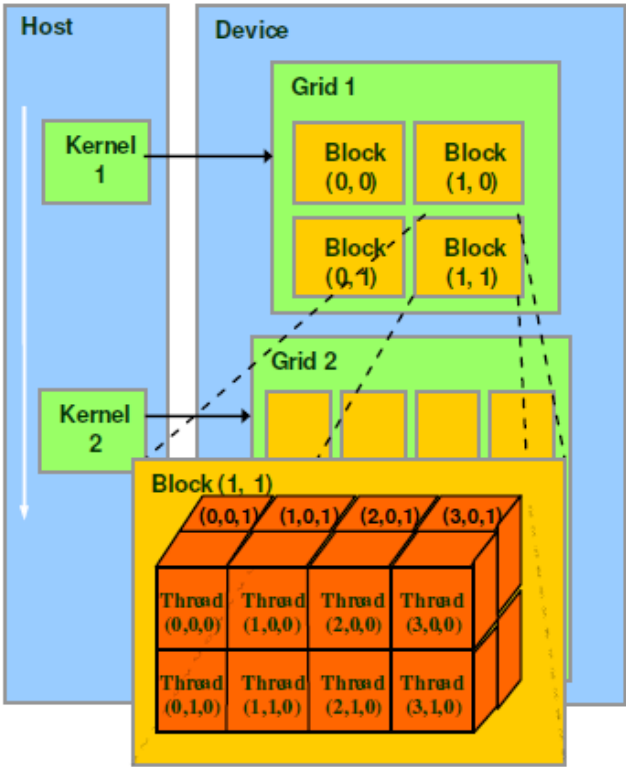


图 3.2 GPU 并行处理单元

为控制 GPU 的并行计算处理，Nvidia 推出的 Cuda 对底层物理硬件做了封装，从软件角度来看，并行计算单元主要划分为 block 和 Thread，每个 block 会分配到不同的 SMs 上执行，并行处理的一个主要的方面就是分配 Block 和 Thread 的索引。

关于 GPU 另外一个需要了解的部分就是 GPU 的内存体系结构，GPU 内部内存支持多种的模式，具体如图 3.3 所示，包括 local memory 是一个 Thread 的本地内存，shared



memory 是一个 Block 共享的内存，而 Global memory 是全局共享的内存。

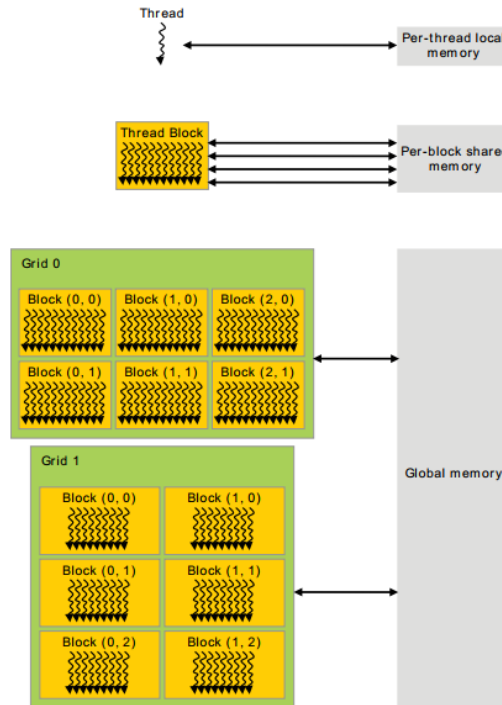


图 3.3 GPU 内存体系结构

上面基本介绍了 GPU 的体系结构之后，结合雷达信号处理的特点，使用 GPU 来计算处理雷达数据还是比较合适的。在上一章的雷达数据模型的计算中已经提到，针对数据模型的处理主要是按照距离库划分，在慢时间维度的数据处理可以分配到每一 block 进行处理。

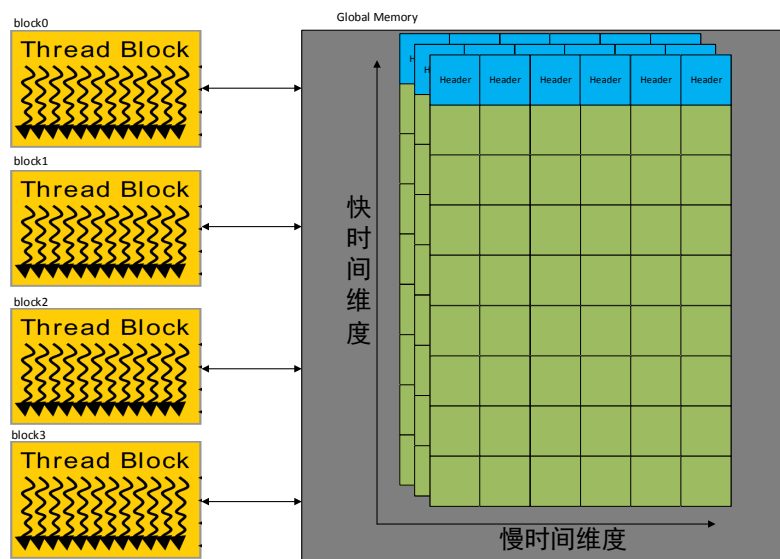


图 3.4 数据存储

根据 GPU 的内存模型，最好的方法便是在 GPU 的 Global Memory 建立一个全局的数据矩阵，数据矩阵的格式便是按照雷达数据处理模型来存储，具体的存储模型如图 3.4 所示进行划分。

## 3.2 数据处理

数据处理这一节主要是根据上一章整理的雷达信号处理的算法进行设计实现。上一章描述了主要的计算目标就是雷达回波强度、径向速度和谱宽的计算，根据算法的描述基本的计算流程可以如下图 3.5 所示。

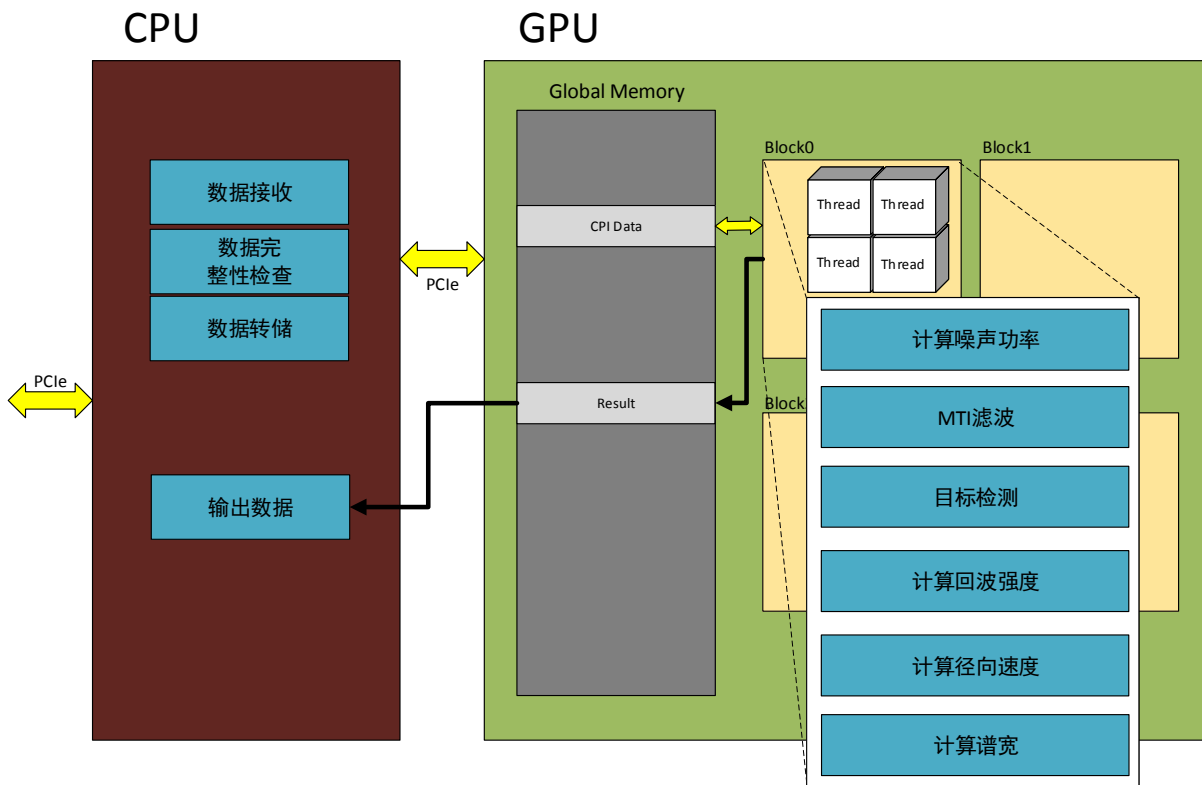


图 3.5 处理流程

在 CPU 一侧主要需要做的处理为数据接收，数据接收处理负责通过 PCIe 驱动从 FPGA 接收数据包，数据完整性检查处理负责检查每一包数据各项参数是否完整正确，数据转储处理负责将数据从 CPU 转储到 GPU 全局内存区，同时如果出现 GPU 内存区数据溢出的情况下，可以在数据转储处理中做一个数据缓冲。

GPU 一侧的计算则主要负责算法的并行计算，首先做的便是提取回波噪声功率用于后续

计算的使用，然后是进行 MTI 滤波，滤除杂波的处理可以让后续的计算结果更加精确可靠。计算回波强度、计算径向速度和计算谱宽的处理则可以并行进行。

# 第四章 详细设计

## 4.1 协议设计

本节主要设计了 FPGA 与上层 x86 平台的数据包格式，数据包中包含计算处理过程中需要的数据和相关参数，具体格式如图 4.1 所示，使用 64 个字节作为报文头，数据区则根据距离库的大小进行设计。

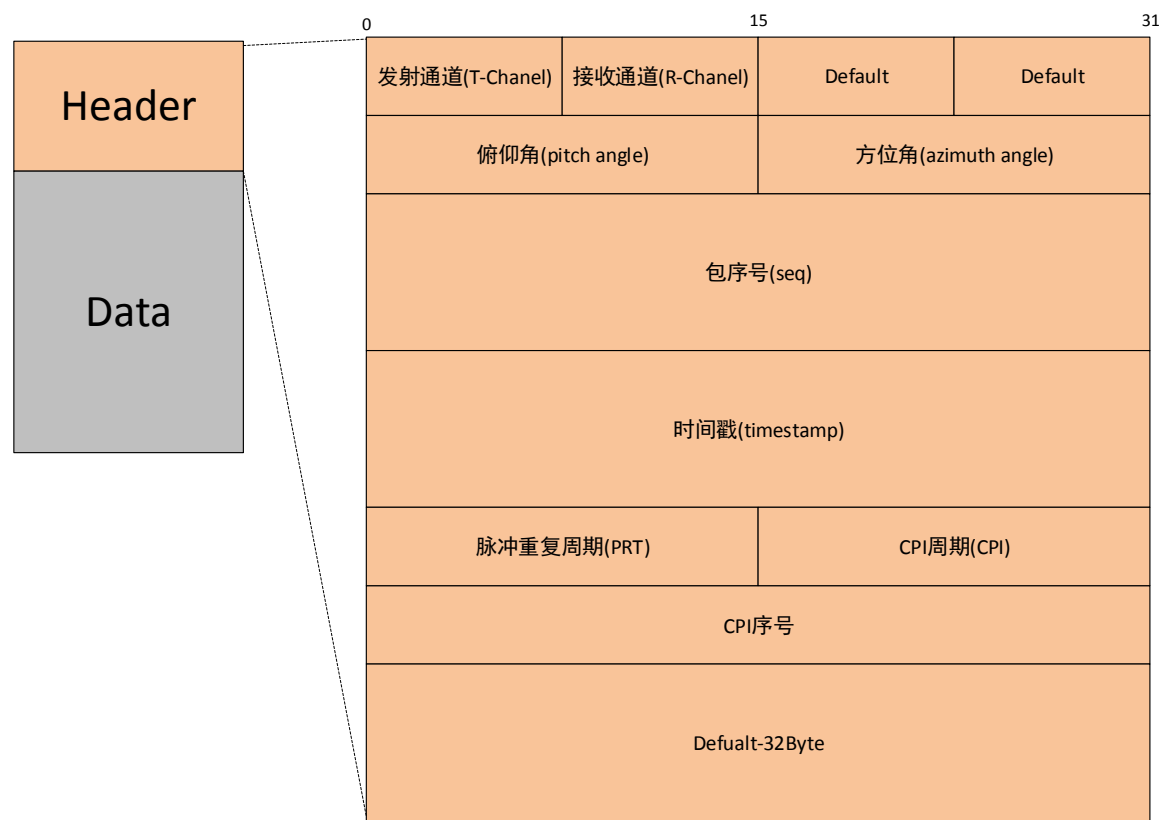


图 4.1 协议包格式

目前格式只是暂定，具体的格式还需要根据结合 FPGA 进行详细讨论。

## 4.2 软件框架

本节主要根据 GPU 软件设计模型，设计了基本的软件处理框架用于指导后续的代码编写过程。

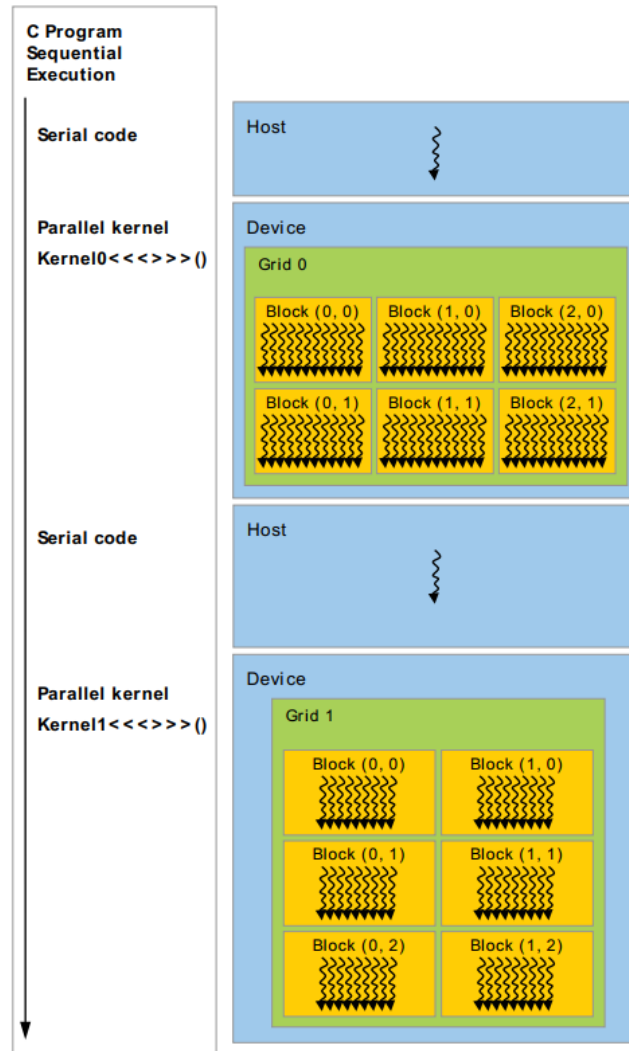


图 4.2 软件设计模型

在 Cuda 官方提供的编程指导<sup>[5]</sup> 中提供了一个基本的编程模式，具体如图 4.2 所示，在雷达信号处理的代码中也将遵循这个基本编程模型。

按照基本编程模型设计的软件程序框架如图 4.3 所示，程序启动之后需要完成对 FPGA 的初始化，如果初始化失败则退出程序。

完成 FPGA 初始化之后，开始进行数据接收，完成一个数据块的接收之后进行数据有效性检查，确定数据为有效数据，然后开始数据的转储，由 CPU 内存转储到 GPU 内存，如果 GPU 内存已经接近溢出，则转储处理还需要进行数据缓存，同时完成数据转储之后还会同步

需要进行数据接收。同时程序会检查 GPU 内存中的数据是否已经存储完成，一旦完成存储立即开始启动并行计算，计算相应的回波噪声功率，对同一个距离库的程序进行滤波处理，完成滤波处理之后进行目标检测，根据目标检测结果决定是否进行相关的参数计算。

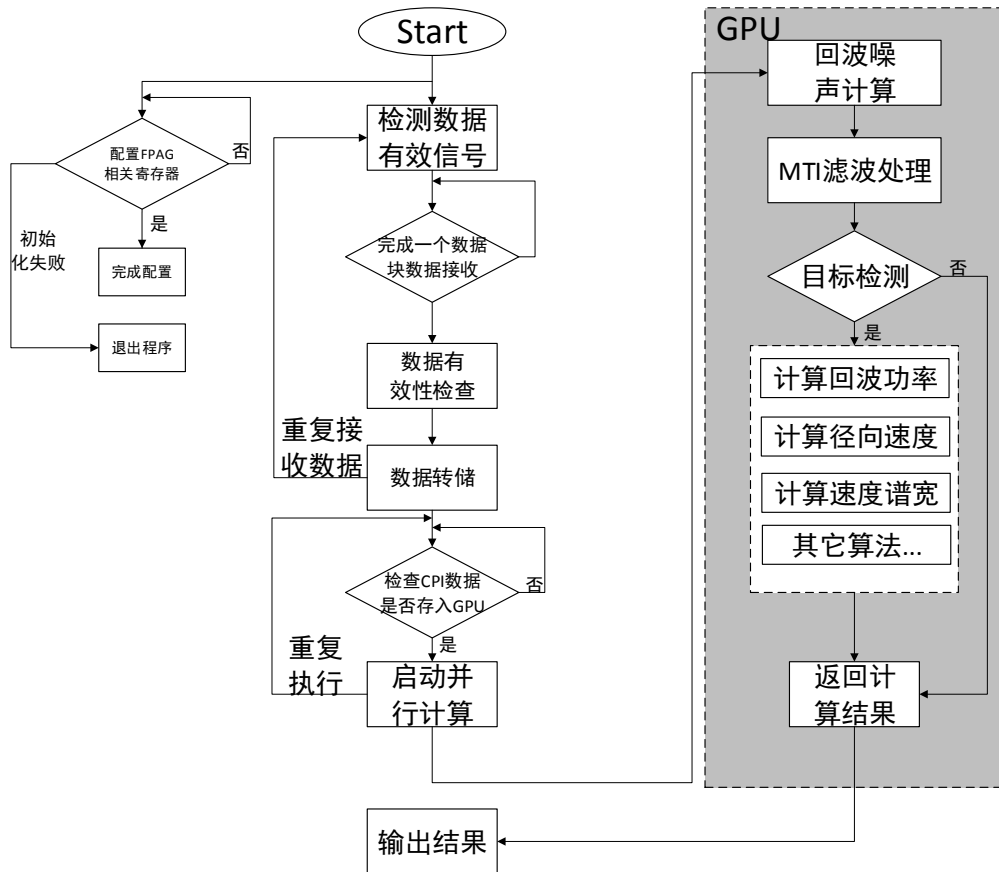


图 4.3 软件设计框架

如果检测到有目标，则计算回波功率、计算径向速度、计算谱宽，这三个值可以并行计算，算法部分的设计将会设计成模块化的模式，每一种算法均可方便进行添加删减，这样便于后续添加更加高级的算法处理模块，在完成计算之后，GPU 一侧的程序将会将计算结果输出到 CPU 主机一侧。

## 第五章 总结

雷达信号处理部分相关内容比较多且复杂，我司又缺少雷达信处的设计实现经验，并且从理论算法到工业实现，整个过程只能在摸索逐步实现，本方案也只是一个初步的讨论方案，但是只要一步一步讨论清楚，集思广益，则定能从无到有，从有到优。

## 参 考 文 献

- [1] Mark A.Richards . 雷达信号处理基础 (第二版) [M]. 2008.
- [2] 张培昌. 雷达气象学 [M]. 气象出版社, 1988.
- [3] [http://apollo.lsc.vsc.edu/classes/remote/lecture\\_notes/radar/doppler/two\\_pulses.html](http://apollo.lsc.vsc.edu/classes/remote/lecture_notes/radar/doppler/two_pulses.html)
- [4] 斯科尼克 (Skolnik,M.I.) 雷达手册: 第 3 版/(美) 南京电子研究所, 2010.
- [5] CUDA C PROGRAMMING GUIDE, October 2018.