

# 双偏振雷达 IQ 数据格式说明

版本 5

2014-03-24

北京敏视达雷达有限公司

版权 © 2014北京敏视达雷达有限公司版权所有。未经北京敏视达雷达有限公司  
允许不得以任何形式出版发行，或以数据库形式进行检索。

# 1 格式介绍

天气雷达的 IQ 数据以一次扫描为单位按文件存储。如 PPI、扇扫和体扫以一个仰角层为单位保存成一个文件，RHI 扫描以一个方位的上升或下降扫描为单位保存成一个文件。

IQ 数据文件名格式如下：CJ\_20130613\_061658\_01\_CDX.IQ，不同字段用‘\_’分隔。第一个字段为站点名称（可在 RDA 软件界面设置）或者特定字符(SOT 代表 RDASOT 控制雷达生成的文件，CAL 是 maina 标定过程中生成的文件)。第二、三字段是日期和时间，均为 UTC 时间。第四字段为该扫描在体扫中的序号，最后一个字段为扫描模式。

每个文件包括一个 TSHheader（128 字节），接着是预留的 256 字节保留空间，此后开始脉冲采样数据，每个脉冲包括一个 TSSweepHeader（128 字节），后面是该脉冲的 I/Q 数据，依次为水平通道 I/Q 数据，垂直通道 I/Q 数据及可能的 burst 信号 I/Q 数据，这些 I/Q 数据的距离库及其长度可由 TSSweepHeader 中的参数计算获得。接着是下一个脉冲的 TSSweepHeader，然后是它的 I/Q 数据，如此直到扫描文件结束。

如图 1 表述 I/Q 数据文件的总体结构。



图 1 I/Q 数据文件结构示意图

## 2 IQ 读取工具

系统发布时，RDA 计算机上有安装好的命令行 tsdump 用于将 IQ 文件输出为文本，tsdump 命令行的参数见该命令行的帮助信息。

Options:

- file filename. ts file to dump
- power . show iq data in (power(DB),angle) pair
- iq . show iq data in (I,Q) pair
- noheader . the file dont have TSHeader.
- onlyheader . just dump TSHeader.
- swpcnt . sweep number to print.
- swpseq . sweep seq number begin to print .
- binindex . binindex
- bincnt . bin number to print
- linewidth . data pair per line
- filter threshold .dump only power above threshold (db)
- triple dump only power min,avg,max per file (db)
- bin show data in bin order(for one radial),default time order
- hori show data only h channel,default all data
- vert show data only v channel,default all data
- verbose . show TSHeader and TSSweepHeader
- quiet . no other message
- version .show program version
- help .show this help message

命令行样例: `tsdump --file /usr/iris_data/cut_bat.s.iq --iq --swpcnt 1 --swpseq 75666046 --binindex 1 --bincnt 100 --quiet`

## 3 结构体描述

本节以 C/C++语言为例描述 I/Q 数据的结构体。

**重要说明:** 本文档声明的结构体要求必须按 1 字节对齐, 以 C/C++语言为例, 结构体定义部分需要用`#pragma pack(1)`声明对齐。

### 3.1 TSHeader 结构

```
int const SITE_NAME_LEN=16;
int const TS_HEADER_SIZE=128;
struct TSHeader{
    char version;//1, 2 为单极化版本, 3为双偏振版本号, 4为加burst信号版本。
    char sitename[SITE_NAME_LEN];
    int spared1;
    char spared2;
    char pol;// h =0, v=1 , hv=3
```

```

float pulsewidth;// pulse width in microseconds.
float calibration;// hori calibration reflectivity (dbz at 1km)
float noise; //noise level of the system in dbm
float freq;// tx freq in MHz
short firstbin;// first bin range in meter
char phasecode;//phase code of TX pulse,0 for fixed 1 for random
float vnoise;//noise level of v channel
float vcalib;// vert calibration reflectivity (dbz at 1km)
char pad[78];
};

```

## 3.2 TSSweepHeader 结构定义如下

```

int const TS_SWP_HEADER_SIZE=128;
struct TSSweepHeader{
    struct timeval time;// time of data,refer to linux man page(gettimeofday)
    int clock ;// clock tagged by Card
    int seqnum;// squence number of this sweep
    int spared[3];
    short az;//azimuth in scale of 1/100 degree
    short el;//elevation in scale of 1/100 degree
    short prf;//pulse repetition frequency in hertz
    short samples;// samples number in current radial
    short binnum;// range bin number in this radial
    short reso;// range bin resolution
    char mode;//surv/dopper. used only by batch mode
    int state;//radial state,0 cut_start,1 cut_mid,2 cut_end,3 vol_start,4 vol_end
    char spotblank;//true if the radial is sector blanking
    short nextprf;//PRF of next sweep;
    float burstmag;//magnitude of burst
    float burstang;//angle of burst
    short swpidx;// index of this sweep in the radial
    short anglereso;//angle (azimuth for ppi ) resolution
    char chan;//channnel number in this pulse 1 for h only and v only,2 for h+v
    unsigned short length;//just for rawiq, internal use
    short burstbinnum;//bin number of burst signal
    char pad[63];
};

```

## 3.3 IQ 数据长度计算

根据I/Q数据结构定义，I/Q数据区长度计算公式为

$$I/Q\text{数据区长度} = (\text{通道数} * \text{binnum} + \text{burstbinnum}) * 2 * 2$$

其中**通道数**由结构体TSSweepHeader中的chan字段决定；  
**binnum**为结构体TSSweepHeader中的字段，表示天气信号I/Q数据距离库的个数；  
**burstbinnum**为结构体TSSweepHeader中的字段，表示burst信号I/Q数据的距离库个数；  
**2**为复数I+Q；  
**2**为每个I/Q数据使用2个字节的编码。

- 注：
1. 在版本五之前，IQ数据。使用32位标准浮点存储的，I/Q数据区长度的公式为：(通道数\*binnum+burstbinnum)\*2\*4。
  2. 在版本三之前通道数可能为0，这时应将通道数作为1。

## 4 版本修订历史

I/Q 数据格式定义在开发过程中，因需求变化会有局部调整和改进，I/Q 数据文件用版本号来保持向下兼容。

I/Q 数据文件的版本号见 TSHader 结构中字段 version，不同版本的修订如表 1。

表 1 不同版本 I/Q 数据格式变化

序号	I/Q 格式版本	修订内容
1	版本 1	初始格式
2	版本 2	TSSweepHeader 结构中 State 字段定义有变化
3	版本 3	TSSweepHeader 中 az 和 el 编码公式由 $360/2^{**13}$ 修改为 $1/100$ 度
4	版本 4	格式中增加 burst 数据。 1、TSSweepHeader 中增加 burstbinnum 字段定义 burst 信号距离库的个数 2、burst 信号 I/Q 数据在天气信号 I/Q 的数据的后面。
5	版本 5	IQ 数据有 32 位浮点改为 16 位浮点编码

## 附录 A 16 比特编码 IQ 数据解码公式

16 比特 IQ 数据编码方式如下图所示， 0-10 位为尾数， 11 位为符号位， 12-15 位为指数位。

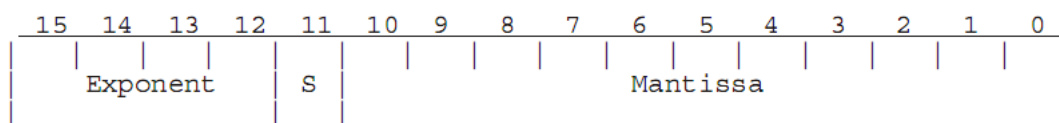


图 2 16 比特 IQ 编码格式

解码流程如下：

1. 如果指数位不为零
  - 1.1. 创建一个有符号的 32 位整数。
  - 1.2. 从编码中拷贝 0-11 位编码的尾数到 32 位整数的 0-11 位。
  - 1.3. 如果编码 11 位为 0
    - 1.3.1. 32 位整数的 11, 12 位为 01否则
    - 1.3.2. 32 位整数的 11, 12 位为 10 (13-31 位也为 1, 表示负数)
  - 1.4. 将指数作为无符号整数计算  $2^{(指数-25)}$ ，将其与 32 位整数相乘得到的浮点作为解码的结果。
2. 否则
  - 2.1. 将尾数视为 12 比特有符号整数乘以  $2^{(-24)}$ 得到的浮点作为解码结果