

SHSC软件开发设计要求说明

|  |  |
| --- | --- |
| 项目名称 | SHSC软件开发 |
| 编写人员 | 颜祺淞、陈剑超 |
| 编写日期 | 2024.9.30 |

目录

[一．项目概要 4](#_Toc178513241)

[二．开发内容1 5](#_Toc178513242)

[2.1 开发平台 5](#_Toc178513243)

[2.2 工作流程 5](#_Toc178513244)

[2.3 文件命名方式 6](#_Toc178513245)

[2.4 系统异常处理 7](#_Toc178513246)

[2.5 系统数据上传nas方案说明（双线程） 7](#_Toc178513247)

[2.5.1 数据量计算 7](#_Toc178513248)

[2.5.2 方案对比 8](#_Toc178513249)

[2.6 数据文件读出功能 11](#_Toc178513250)

[三．开发内容2 14](#_Toc178513251)

[3.1 开发平台 14](#_Toc178513252)

[3.2 工作流程 14](#_Toc178513253)

[3.3 文件命名方式 15](#_Toc178513254)

[3.4 文件数据量计算 15](#_Toc178513255)

[3.5 数据文件读出功能（同2.6节） 15](#_Toc178513256)

[四．附录 19](#_Toc178513257)

[4.1 起始命令协议 19](#_Toc178513258)

[4.2 惯导协议 23](#_Toc178513259)

[4.3 磁力仪协议 27](#_Toc178513260)

[4.4 原始数据上传协议 28](#_Toc178513261)

# 一．项目概要

SHSC软件开发项目是基于全志A40i核心板的硬件平台进行的，该平台的处理器架构采用了FPGA与ARM的组合设计。

在这一架构中，FPGA主要负责数据的采集和传输，能够高效地处理实时数据流并确保数据的准确性和完整性。同时，ARM处理器承担了与上位机的交互任务，负责接收FPGA、串口等发送的数据并储存到板载硬盘；同时板载硬盘中的数据上传给nas进行备份；在nas失效时，可以通过SSH远程访问板载硬盘来手动下载数据。

通过这种分工，FPGA与ARM之间的协作可以最大化系统性能，数据处理的效率和稳定性也得到了显著提升。

# 二．开发内容1

## 2.1 开发平台

本次软件开发均基于全志A40i核心板的硬件平台，该硬件平台处理器为FPGA+ARM，FPGA负责数据采集和传输，ARM负责上位机交互、数据接收、存储和读出功能。核心板示意图如图2-1所示。

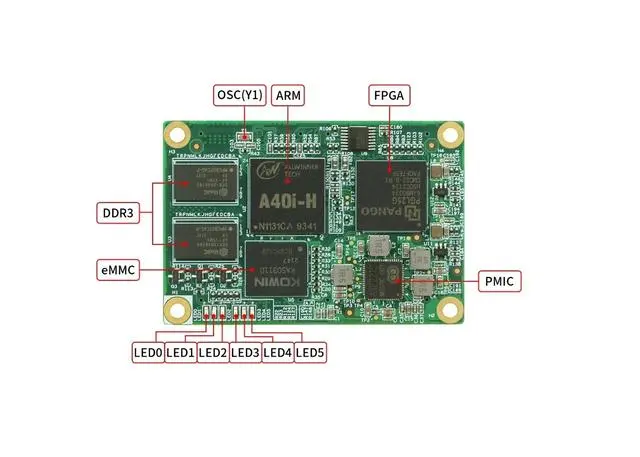


图2-1核心板示意图

## 2.2 工作流程

ARM端只保留一个千兆网接口（设备最终交付时百兆网PHY为断电状态），上位机的命令、向NAS发送数据、数据读出均采用该接口。

声学数据：

ARM接收上位机下发的起始命令，协议如文末附1所示，接收到起始命令后，先建立一个声学文件，并先写入32字节的起始命令内容，然后将该起始命令转发给FPGA（发送命令的接口可采用ARM的UART6，PI12、PI13）。FPGA收到后开始发射和采集过程，最大的数据量5MB/s。ARM接收FPGA发送的采集数据和串口的惯导数据（惯导接口为UART3）合并保存，其中声学数据全部保存，惯导数据只需保存时间和位置信息（惯导数据协议详见附2，需保存信息已标黄）。

ARM每收到FPGA发送的512KB声学采样数据，在512KB数据后加22字节惯导时间位置数据。数据存入硬盘中，形成一个声学+惯导时间位置的数据文件。当前存储的声学文件大于1GB（也可以2GB）时，结束当前文件并保存，再重新建立一个新文件，同样写入32字节的起始命令内容，然后重复上述过程并保存。

磁学数据：

ARM端通过串口接收惯导和磁力仪数据（磁力仪接口为UART4），数据的频率均为10Hz。ARM接收上位机下发的起始命令，协议如文末附1所示，接收到起始命令后，建立一个磁学文件，并先写入32字节的起始命令内容，然后将接收的惯导和磁力仪数据保存，惯导数据只保存时间和位置信息，磁力仪数据全部保存，形成一个磁学+惯导时间位置的数据文件。按照协议，每秒的数据量为（22+60）\*10=820B/s，正常情况磁学文件大小不会超过1GB。惯导协议如附2，磁力仪协议如附3。

## 2.3 文件命名方式

为了确保文件名不重复并提高文件管理的效率，采用统一的命名方式至关重要。这不仅有助于团队成员快速识别文件内容，还能避免因命名不当而导致的数据混淆。因此，每个文件名应该都是唯一且具有描述性的。

声学文件命名方法为：SX\_xxx.dat，磁学文件命名方法为：CX\_xxx.dat，其中xxx采用RTC时间。虽然采用顺序号1、2、3…表面上可以避免重复地命名，但是若出现系统重启的情况，程序会重新从1、2、3进行命名，导致文件重复。因此，文件的命名最好采用时间信息，时间信息不仅不会出现重复，而且代表了数据的先后顺序。在Linux系统中，RTC（实时时钟）是一种硬件时钟，通常用于跟踪时间，即使系统关闭或断电，RTC仍能保持时间。

## 2.4 系统异常处理

由于存在异常复位、掉电重启等可能性，设备需自主恢复运行。上位机只在初始阶段配置一次起始命令，ARM需保存该命令内容，当发生上述异常情况，ARM将读取命令内容并恢复2.2节所述的采集接收存储过程。文件名可按照2.3节所述进行命名。

## 2.5 系统数据上传nas方案说明（双线程）

### 2.5.1 数据量计算

第一部分来自于FPGA中的AD采集数据。高频采样芯片型号为AD9650BCPZ-25，采集速率为2Byte@5MSPS；低频采样芯片型号为AD7760，采集速率为3Byte@600KSPS。目前最大的数据采集量对应的配置参数为：每秒采集10次，每次采集时长为40ms。FPGA以每80us的数据采集量（944B）和冗余数据（80B）进行组合形成1024B，每次40ms的采集将产生500KB数据，则每秒钟的数据量为5000KB，8小时的工作时间内，总共能够形成137.33GB的数据。

第二部分来自于惯导数据，数据包为110B，实际存储22\*2B, 接收频率为10HZ，8小时数据量约为12.1MB。

第三部分来自于低速串口，数据包为60B，接收频率为10HZ，8小时数据量约为16.5MB。

因此，系统8小时采集总数据量约为137.36GB。

### 2.5.2 方案对比

方案一：采集同时上传，如图2-2所示

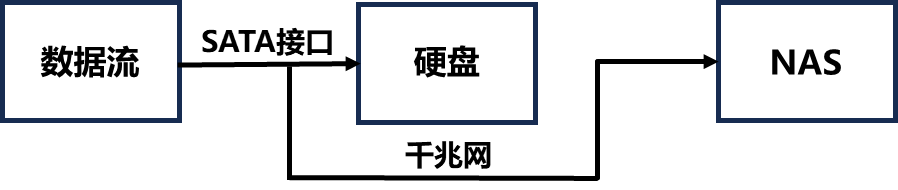


图2-2采集同时上传方案示意图

虽然千兆网口的理论传输速度能够达到100MB/s，但是在大多数情况下只能达到40MB/s，而工作8小时总数据量为137.36GB，而且采集并同时上传的方案受限于CPU性能、内存大小等，并在进行上传工作时，无法接收新的数据流。因此，此方案一不能应用于实际。

方案二：采集后上传，如图2-3所示



图2-3采集后上传方案示意图

在8小时的工作时间内，采集数据总共137.36GB，nas与开发板连接采用千兆网口，理论传输速度可达100MB/s，但经过实验室测试，很多时候只能达到40MB/s的速度。所以，在8小时的采集完成后，上传需要约1小时。虽然AUV需要2小时的上浮，上传工作可以在上浮过程中完成，但是在这种方案下，若采集设备出现问题，已采集但未上传的数据面临丢失的风险。这与“及时备份”的要求不符。因此，方案二在实际工作环境内的表现欠佳。

方案三：边采集边上传（双线程），如图2-4所示

方案三采取边采集采上传的双线程工作办法。主线程负责持续接收数据并保存到板载硬盘，子线程负责将写入结束的文件不断上传到nas，形成流水线的工作形式。

由于芯片位数限制（32位），单个文件写入最大不能超过2GB，因此在测试时，当文件写满2GB时，便会重新命名一个新文件。已写满的文件将进入子线程的上传程序，通过这样的工作方式，相比旧方案，不仅节省了时间，数据也能及时得到备份。图2-5是程序总体流程图。

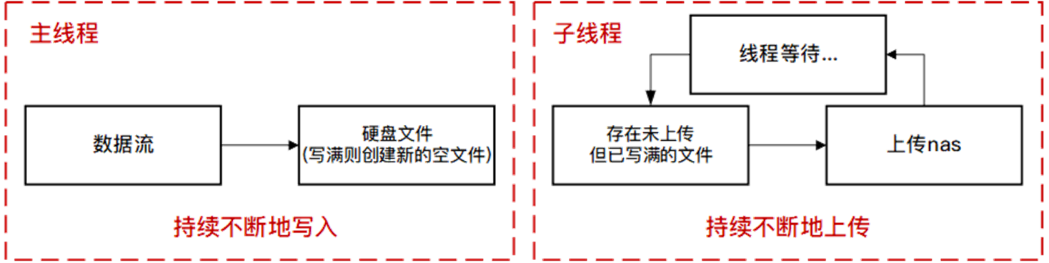


图2-4边采集边上传（双线程）方案示意图

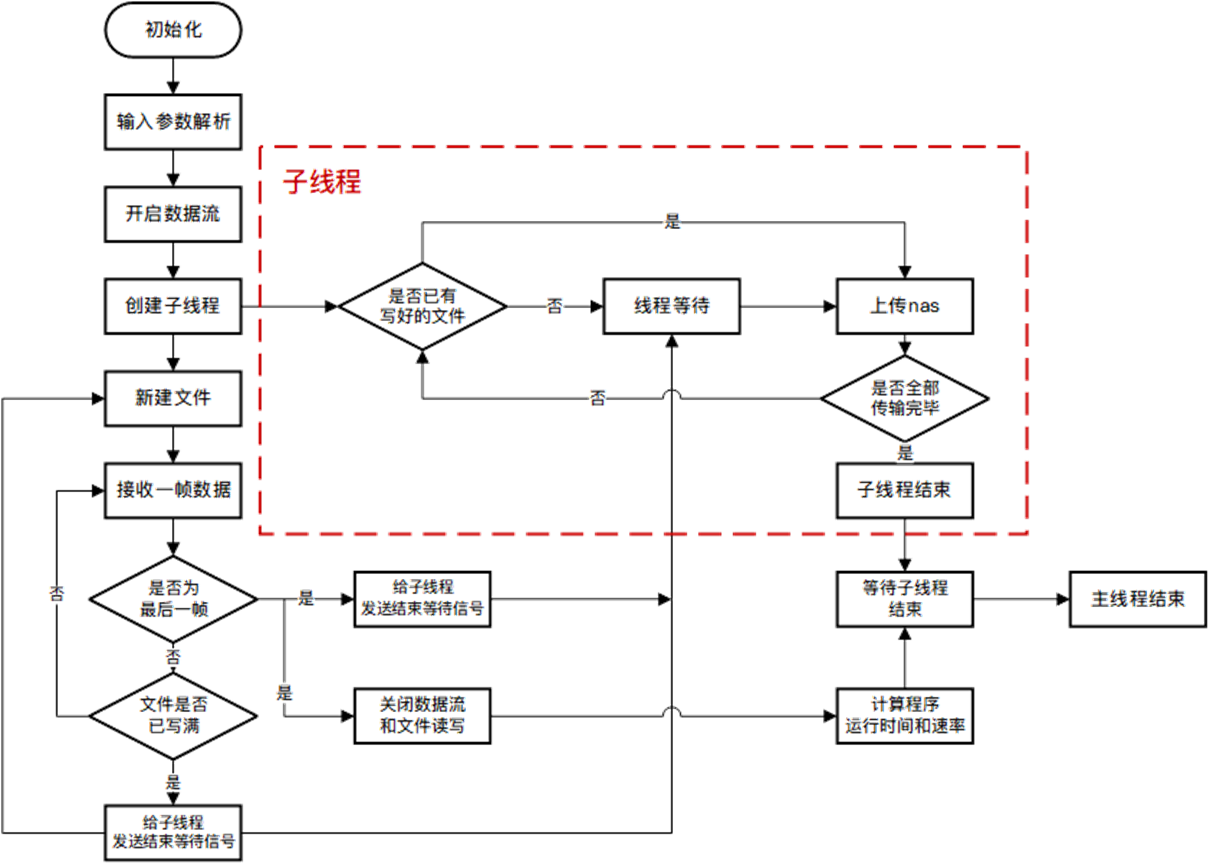


图2-5程序总体流程图

## 2.6 数据文件读出功能

Xshell软件是一款功能强大的终端模拟软件，它支持包括SSH、SFTP等多种协议来连接远程设备。Xftp是一款专门用于文件传输的SFTP和FTP客户端软件，能够帮助用户在本地计算机和远程服务器之间进行文件上传、下载和管理。它与Xshell无缝集成，用户可以通过Xshell直接启动Xftp进行文件传输，无需手动设置。

主控板的ARM系统是基于Linux5.10的Ubuntu20.04系统，支持SSH协议，在开发时我们会将主控板、nas、PC都连接到同一个路由器的局域网下，在Xshell软件中通过IP地址进行远程访问主控板系统，通过Xftp进行PC与硬盘、nas的文件传输。同样地，在采集数据完毕后，只要PC与主控板在同一个局域网下，PC就能远程访问主控板的板载硬盘和nas，通过手动下载的方式来获取本次工作的数据，这样便达到了甲方所要求的数据文件读出功能。以下是具体流程实现：

设备要求：windows系统的PC

软件要求：Xshell7及以上（家庭/学校免费版）、Xftp7及以上（家庭/学校免费版），两者结合使用，版本应对应

环境要求：PC、嵌入式开发板、nas应在一个局域网中，如图2-6所示。其中名称为Yan的设备是PC，其IP地址为192.168.1.207；名称为T3-A40i-Tronlong的设备是嵌入式开发板，其IP地址为192.168.1.61；名称为T2-NTOZ的设备是极空间nas，其IP地址为192.168.1.197。他们通过有线和无线连接的方式连接在同一个路由器中，组成了一个小型的局域网。

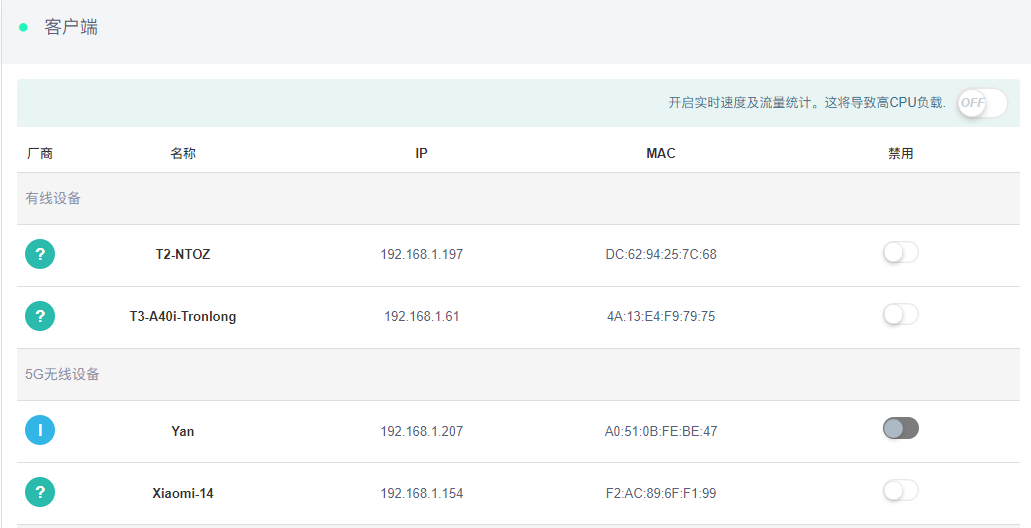
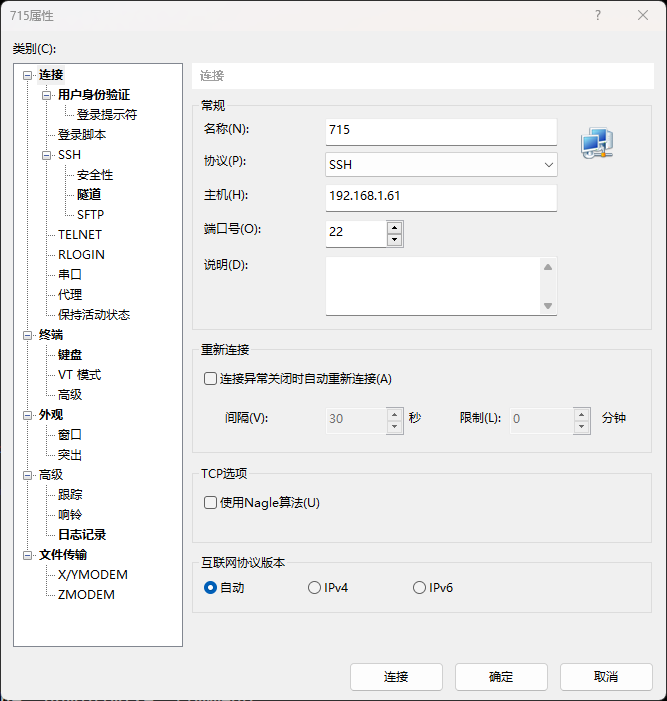


图2-6 同一个局域网下IP地址

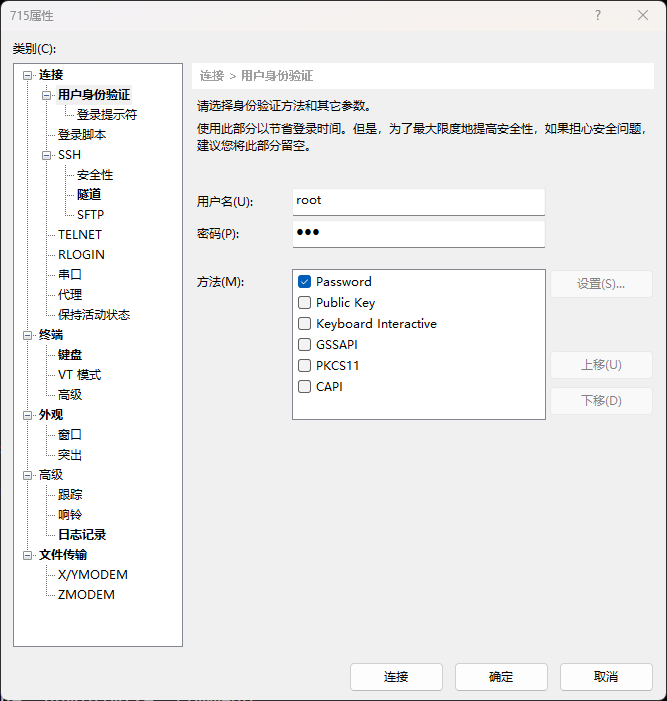
连接流程：

第一步，建立连接。

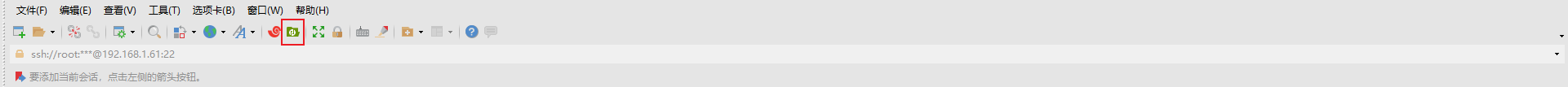
在Xshell的新建会话窗口的连接选项卡，选择协议为SSH，主机为主控板的IP地址，端口号为22保持不变，名称可以自己任取。



接着在用户身份验证选项卡，用户名为主控板Linux系统的用户名（举例：root），密码为对应的用户密码（举例：123）。接着点击连接并保存密钥，完成连接。



第二步，打开Xftp。



第三步，在打开的Xftp软件窗口输入文件所在地址，举例：/mnt/data（板载硬盘），便能够找到对应的文件，使用拖拽等方式将文件下载到本地计算机。

# 三．开发内容2

## 3.1 开发平台

本次软件开发均基于全志A40i核心板的硬件平台，该硬件平台处理器为ARM，负责上位机交互、数据接收、存储和读出功能。核心板示意图如图3-1所示。

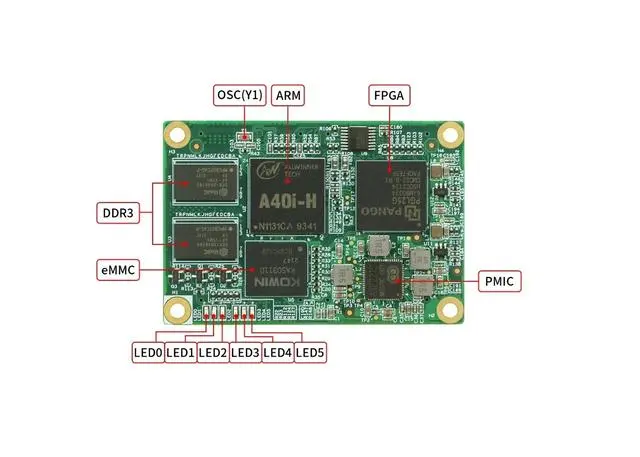


图3-1核心板示意图

## 3.2 工作流程

该硬件平台处理器为ARM，引出1路千兆网、1路百兆网（不使用）、1路SATA及调试串口。

该数据存储板与数据源板、及其他设备或上位机均通过交换机连接于同一个本地网络中。

数据源板的传输协议为UDP，数据率最大不大于2MB/s，数据协议见附录4。因采用UDP传输，收包顺序可能为乱序，不需要整理包的顺序，可后期读出由上位机软件进行数据包顺序的整理。

存储板配置1路SATA硬盘和1路千兆网。存储板通过千兆网接收来自数据源板的数据后，立即保存至SATA硬盘中。待工作结束，可通过千兆网读出硬盘中的数据文件。

## 3.3 文件命名方式

由于数据源板与存储板上均不含RTC，无法通过时间信息来进行命名，目前关于文件的命名，暂时使用如：xx\_1、xx\_2等按顺序进行命名。

## 3.4 文件数据量计算

目前数据源板的发送频次为10Hz，每次1044B\*100，整个工作时长为7天，计算数据量1044\*100\*10\*3600\*24\*7≈588GB。单个文件大小不超过2GB，因没有板载RTC芯片，文件名暂时按顺序号命名。

## 3.5 数据文件读出功能（同2.6节）

Xshell软件是一款功能强大的终端模拟软件，它支持包括SSH、SFTP等多种协议来连接远程设备。Xftp是一款专门用于文件传输的SFTP和FTP客户端软件，能够帮助用户在本地计算机和远程服务器之间进行文件上传、下载和管理。它与Xshell无缝集成，用户可以通过Xshell直接启动Xftp进行文件传输，无需手动设置。

主控板的ARM系统是基于Linux5.10的Ubuntu20.04系统，支持SSH协议，在开发时我们会将主控板、PC都连接到同一个路由器的局域网下，在Xshell软件中通过IP地址进行远程访问主控板系统，通过Xftp进行PC与硬盘的文件传输。同样地，在接收数据完毕后，只要PC与主控板在同一个局域网下，PC就能远程访问主控板的板载硬盘，通过手动下载的方式来获取数据，这样便达到了甲方所要求的数据文件读出功能。以下是具体流程实现：

设备要求：windows系统的PC

软件要求：Xshell7及以上（家庭/学校免费版）、Xftp7及以上（家庭/学校免费版），两者结合使用，版本应对应

环境要求：PC、嵌入式开发板、nas应在一个局域网中，如图3-2所示。其中名称为Yan的设备是PC，其IP地址为192.168.1.207；名称为T3-A40i-Tronlong的设备是嵌入式开发板，其IP地址为192.168.1.61；名称为T2-NTOZ的设备是极空间nas，其IP地址为192.168.1.197。他们通过有线和无线连接的方式连接在同一个路由器中，组成了一个小型的局域网。

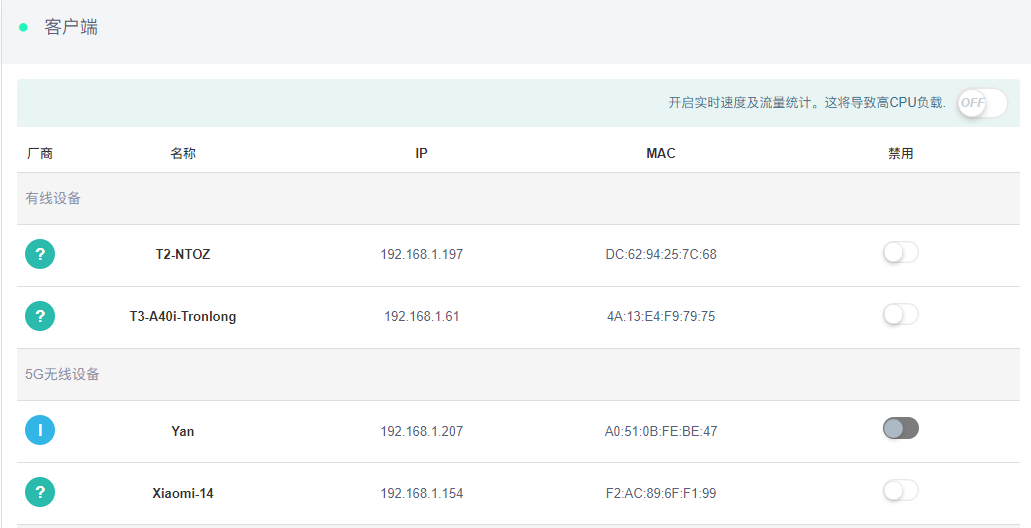
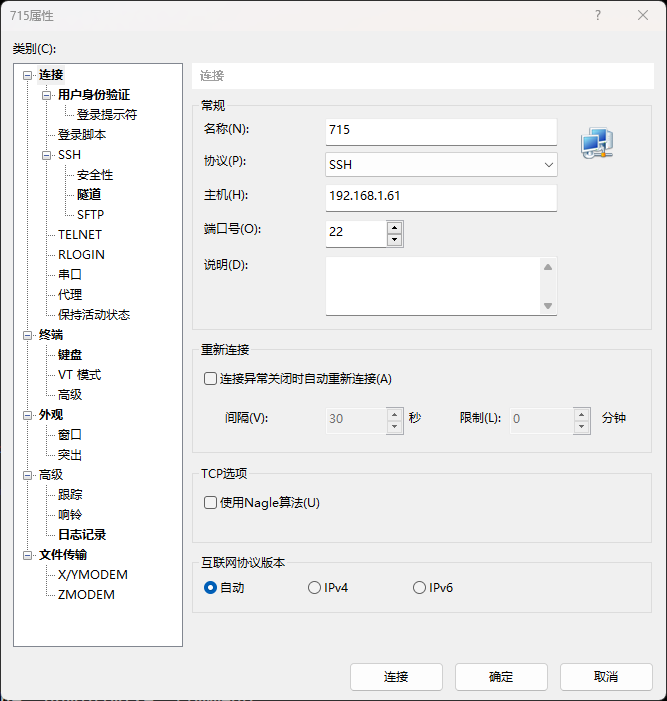


图3-2 同一个局域网下IP地址

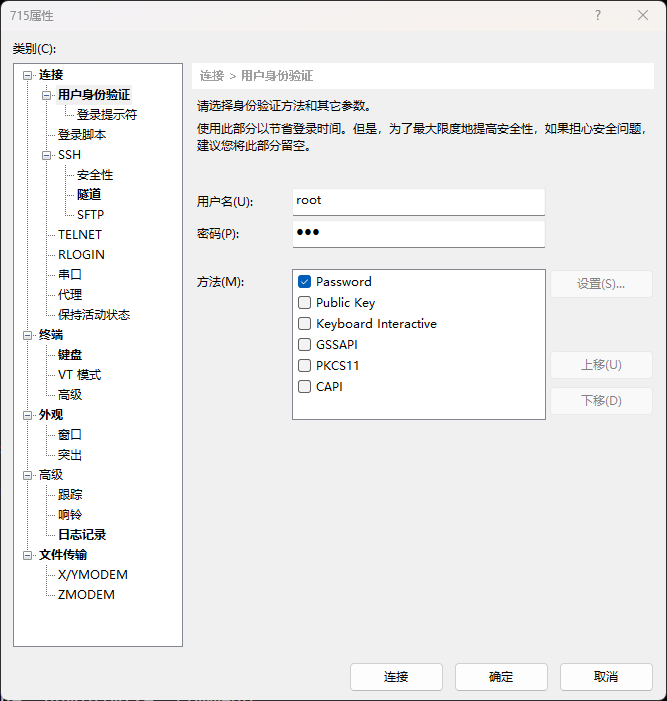
连接流程：

第一步，建立连接。

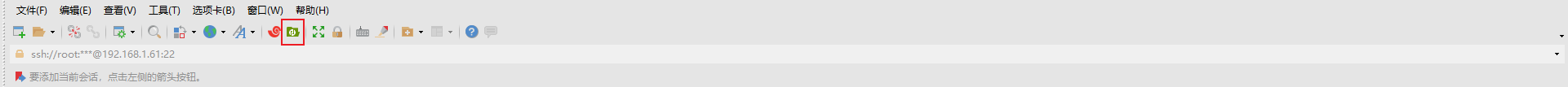
在Xshell的新建会话窗口的连接选项卡，选择协议为SSH，主机为主控板的IP地址，端口号为22保持不变，名称可以自己任取。



接着在用户身份验证选项卡，用户名为主控板Linux系统的用户名（举例：root），密码为对应的用户密码（举例：123）。接着点击连接并保存密钥，完成连接。



第二步，打开Xftp。



第三步，在打开的Xftp软件窗口输入文件所在地址，举例：/mnt/data（板载硬盘），便能够找到对应的文件，使用拖拽等方式将文件下载到本地计算机。



# 四．附录

## 4.1 起始命令协议

附1

起始命令，9600bps，32字节，数据为小端格式，低字节在前。

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 内容 | 字节数 | 说明 |
| 1 | 配置参数数据帧头 | 2 | Uint16，0xF0,0x00 |
| 2 | 指令长度 | 2 | Uint16，0x20,0x00 |
| 3 | 设备状态 | 1 | Uint8  0x0A: 启动  0x0B: 关闭 |
| 4 | 发射功率1 | 1 | Uint8，  0x0A: 1档  0x0B:2档  0x0C:3档  0x0D: 4档 |
| 5 | 工作周期 | 1 | Uint8,  0x0A:0.1s  0x0B:0.2s  0x0C:0.5s  0x0D: 1s |
| 6 | 发射脉宽 | 1 | Uint8,  0x0A: 0.1ms  0x0B:0.2ms  0x0C:0.5ms  0x0D:1ms |
| 7 | 发射功率2 | 1 | Int8，0-45dB可设，3dB步进,  0x00: 0dB  0xFD: -3dB  0xFA: -6dB  0xF7: -9dB  0xF4: -12dB  0xF1: -15dB  0xFF: -18dB  0xEB: -21dB  0xE8: -24dB  0xE5: -27dB  0xE2: -30dB  0xDF: -33dB  0xDC: -36dB  0xD9: -39dB  0xD6: -42dB  0xD3: -45dB |
| 8 | 差频信号 | 1 | Uint8  0x0A:单频  0x0B:宽带 |
| 9 | 差频信号宽带形式频率 | 1 | Uint8，宽带形式有效，  0x0A:(90-120)KHz  0x0B:(80-120)kHz  0x0C:(70-130)kHz  0x0D:(60-140)kHz. |
| 10 | 海底跟踪模式 | 1 | Uint8，  0x0A: 手动  0x0B:自动 |
| 11 | 接收增益 | 1 | Int8，0-45dB可设，3dB步进  0x00: 0dB  0xFD:-3dB  0xFA: -6dB  0xF7:-9dB  0xF4: -12dB  0xF1:-15dB  0xEE: -18dB  0xEB: -21dB  0xE8: -24dB  0xE5: -27dB  0xE2: -30dB  0xDF: -33dB  0xDC: -36dB  0xD9: -39dB  0xD6: -42dB  0xD3: -45dB |
| 12 | TVG GO | 1 | Uint8，0-255可设(对应0x00-0xFF) |
| 13 | TVGα | 1 | Uint8，0-255可设(对应0x00-0xFF) |
| 14 | TVGβ | 1 | Uint8，0-255可设(对应0x00-0xFF) |
| 15 | 深度范围 | 1 | Uint8，  0x0A:(0-10m)  0x0B:(0-20m)  0x0C:(0-30m)  0x0D:(0-40m)  0x0E:(0-50m)  0x0F:(0-60m) |
| 16 | 输出点数 | 1 | Uint8，  0x0A: 2048  0x0B:4096 |
| 17 | 发射状态 | 1 | Uint8，  0x0A: 打开发射  0x0B: 关闭发射 |
| 18 | 高压控制状态 | 1 | Uint8，  0x0A: 打开高压  0x0B: 关闭高压 |
| 19 | 保留字节 | 10 | Uint8，全部填充0xFF |
| 20 | 帧尾 | 2 | Uint8，0x0D，0x0A |

## 4.2 惯导协议

附2：

惯导协议，9600bps，10Hz，110个字节，记录22字节已标黄

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte 0~3 | 0xAA 55 5A A5 |  | 帧头 | |
| Byte 4 | 0xXX |  | 设备ID，跟型号一致  0x2B=43表示4X水面(3) | |
| Byte 5 | 0x6E |  | 报文长度110 | |
| Bytes 6 to 9 | Latitude (纬度) | Signed 32 bits integer | in °(+/-231 = +/- 180°)  Sign “+” North of equator | |
| Bytes 10 to 13 | Longitude (经度) | Signed 32 bits integer | in °(+/-231 = +/- 180°)  Sign “+” East of Greenwich | |
| Bytes 14 to 17 | Altitude (高度) | Signed 32 bits integer | in mm | |
| Bytes 18 to 19 | Heave (升沉) | Signed 16 bits short | in mm  Sign “+” in down direction | |
| Bytes 20 to 21 | VN | Signed 16 bits short | in mm/s | |
| Bytes 22 to 23 | VE | Signed 16 bits short |
| Bytes 24 to 25 | VD | Signed 16 bits short |
| Bytes 26 to 27 | Roll (横滚) | Signed 16 bits short | in ° (+/- 215= +/- 180°),Sign“+” when port side up | “北-东-地”  坐标系 |
| Bytes 28 to 29 | Pitch (俯仰) | Signed 16 bits short | in ° (+/- 215= +/- 180°),Sign “+” when bow up |
| Bytes 30 to 31 | Heading (航向) | Unsigned 16 bits short | in ° (216= 360°) |
| Bytes 32 to 33 | INS Status | Unsigned 16 bits short | 见表1.4 | |
| Bytes 34 to 37 | GPS Latitude | Signed 32 bits integer | in °(+/-231 = +/- 180°)  Sign “+” North of equator | |
| Bytes 38 to 41 | GPS Longitude | Signed 32 bits integer | in °(+/-231 = +/- 180°))  Sign “+” East of Greenwich | |
| Bytes 42 to 45 | GPS Altitude | Signed 32 bits integer | in mm | |
| Bytes 46 to 47 | GPS VEL | Signed 16 bits short | in mm/s | |
| Bytes 48 to 49 | GPS Heading | Unsigned 16 bits short | in ° (216= 360°) | |
| Byte 50 | GPS Status | Unsigned 8 bits | GPS状态：0初始化, 1单点定位, 2码差分, 3无效PPS, 4固定解， 5浮点解, 6正在估算, 7人工输入固定值, 8模拟模式, 9WAAS差分, 10有数据无效, 11无数据,15配置GPS | |
| Bytes 51 to 52 | DVL Vx | Signed 16 bits short | in mm/s  对地速度 | |
| Bytes 53 to 54 | DVL Vy | Signed 16 bits short |
| Bytes 55 to 56 | DVL Vz | Signed 16 bits short |
| Bytes 57 to 60 | DVL Altitude | Signed 32 bits integer | in mm，对水底高度 | |
| Byte 61 | DVL Status | Unsigned 8 bits | 0 无数据  1 有数据,但无效  2 有数据,对地有效  3 有数据,对水有效  4有数据,对地、对水都有效  15配置DVL | |
| Bytes 62 to 65 | Gx角速度 | Signed 32 bits integer | in 0.000001°/s  （10-6） | “前-右-下”  坐标系 |
| Bytes 66 to 69 | Gy角速度 | Signed 32 bits integer |
| Bytes 70 to 73 | Gz角速度 | Signed 32 bits integer |
| Bytes 74 to 77 | Ax加速度 | Signed 32 bits integer | in 0.0000001m/s2  （10-7） |
| Bytes 78 to 81 | Ay加速度 | Signed 32 bits integer |
| Bytes 82 to 85 | Az加速度 | Signed 32 bits integer |
| Bytes 86 to 87 | Tempreture | Signed 16 bits short | in 0.01°C | |
| Bytes 88 to 89 | IMU Status | Unsigned 16 bits | 见表1.5 | |
| Bytes 90 to 92 | YYMMDD | 24 bits unsigned integer | 年月日 | |
| Byte 93 to 95 | HHMMSS | 24 bits unsigned integer | 时分秒 | |
| Byte 96 to 97 | ms | 16 bits unsigned integer | 毫秒 | |
| Bytes 98 to 99 | DVL Vx | Signed 16 bits short | in mm/s, 对水速度 | |
| Bytes 100 to 101 | DVL Vy | Signed 16 bits short |
| Bytes 102 to 103 | DVL Vz | Signed 16 bits short |
| Bytes 104 to 106 | Depth | Signed 24 bits float | 压力计测量的深度, in mm；  如果没有深度计，Bytes 104-105做为发送计数器（低字节在前），Bytes 106为0 | |
| Bytes 107 | 显示GPS/DVL信息更新状态 |  | Bit 0: GPS位置更新  Bit 1: GPS速度更新  Bit 2: 0  Bit 3: DVL对底速度更新  Bit 4: DVL对水速度更新  Bit5:手动更新位置  Bit6:手动更新速度  Bit 7: 备用0 | |
| Bytes 108 to 109 | Checksum | Unsigned 16 bits | 求和 (Bytes 0 to 107) | |

## 4.3 磁力仪协议

附3：

磁力仪协议，9600bps，10Hz，60字节,数据为小端格式，低字节在前

|  |  |  |  |
| --- | --- | --- | --- |
|  | 长度 | 字段内容/数值范围 | 说明 |
| 帧头 | 2 | 0x4C\_0x57 |  |
| 帧长度 | 2 | 0x3C\_0x00 |  |
| 控制字段 | 4 | 0x01\_0x00\_0x01\_0x01 |  |
| 数据段  48字节 | 4 | -1000000～1000000 | X1轴磁场数据（单位分量为0.01nT） |
| 4 | -1000000～1000000 | Y1轴磁场数据（单位分量为0.01nT） |
| 4 | -1000000～1000000 | Z1轴磁场数据（单位分量为0.01nT） |
| 4 | 0～100000 | 总场值1 |
| 4 | -1000000～1000000 | X2轴磁场数据（单位分量为0.01nT） |
| 4 | -1000000～1000000 | Y2轴磁场数据（单位分量为0.01nT） |
| 4 | -1000000～1000000 | Z2轴磁场数据（单位分量为0.01nT） |
| 4 | 0～100000 | 总场值2 |
| 4 | -1000000～1000000 | X3轴磁场数据（单位分量为0.01nT） |
| 4 | -1000000～1000000 | Y3轴磁场数据（单位分量为0.01nT） |
| 4 | -1000000～1000000 | Z3轴磁场数据（单位分量为0.01nT） |
| 4 | 0～100000 | 总场值3 |
| 帧尾 | 4 | 0x00\_0x00\_0x0D\_0x0A |  |

## 4.4 原始数据上传协议

附录4

原始数据上传协议。每次上行若干数据包，每包大小为（20+1024）字节。

每包的前20字节为

**typedefstruct**

{

/\*000\*/int16u pack\_hdr; //---数据包头 0x6162

/\*002\*/int32u pack\_cnt; //---TP分包计数

/\*006\*/int32u pack\_sum; //---TP总包数

/\*010\*/int16u ping\_hdr; //---发射周期头 5557有发射头，5555无

/\*012\*/int32u ping\_cnt; //---周期计数

/\*016\*/int32u valid; //---本包有效声学数据

} tPing;

后1024字节为数据内容。

关于结构体tPing的说明：

我们将设备的每次发射信号和数据采集处理和传输称之为1个Ping，每个Ping，存储板将收到若干个包的数据。举例：假设每Ping，数据源板发送1044B\*100，则pack\_cnt取值为0~99，pack\_sum为100。ping\_cnt从1开始计数，若以10Hz的频次，7天的计数值最大为10\*3600\*24\*7=6048000。Pack\_hdr,ping\_hdr,valid为固定值。