

抚州市住房和城乡建设局文件

抚住建监〔2021〕18号

抚州市建设工程“智慧工地”管理实施办法

为规范建筑市场秩序，加强建筑工人实名制管理，维护建筑工人和建筑企业合法权益，促进建筑业高质量发展，贯彻落实住建部印发《2016-2020年建筑业信息化发展纲要》、《江西省城市功能与品质提升三年行动方案》、《关于进一步加强城市建筑工地扬尘治理工作的通知》，依据《建筑法》《保障农民工工资支付条例》《关于印发建筑工人实名制管理办法（试行）的通知》等法律法规、文件规定，结合我市实际，市住建局决定在全市范围内，全面推行建筑领域建筑监管信息化建设，现制定以下管理办法：

一、总体目标

在2021年11月31日前建成实现全市建筑行业从业人员及劳务用工实名制常态化监管、建筑工地扬尘治理“六个

百分百”常态化监管、2021年内建成特种设备运行常态化监管等涉及建筑行业安全质量监管的监管信息化一体平台（抚州市建筑工地监管信息化一体平台 <http://223.82.47.237:8081/>）全面实现建设工程智慧工地管理。

（一）全面落实工地施工作业“六个百分之百”

落实工地施工作业“六个百分之百”，实现“无带泥上路、无扬尘污染、无噪声扰民”。加强施工工地监管。实现工地围挡、物料堆放覆盖、土方开挖湿法作业、路面硬化、出入车辆清洗、渣土车辆密闭运输、“六个百分之百”。全市各区、县建筑工地全部达到扬尘治理“六个百分之百”要求。全市各区、县在建项目建成远程视频监控、远程环境监测，将建筑工程施工现场视频监控信息、环境监测数据通过网络实时传输至抚州市建筑工地监管信息化一体平台。

（二）全面推行实名制管理制度

全市范围内所有在建、新建工程项目的工程总承包、施工总承包企业、直接承包建设单位发包工程的专业承包企业和专业分包企业自有以及临时招用的建筑从业人员，全部纳入实名制管理范围，规范实名制管理信息采集、规范关键岗位人员考勤管理，同时关键岗位人员考勤信息纳入建筑市场诚信管理。全市所有在建工程项目要在施工现场按照国务院实名制管理要求设置门禁系统、考勤设备，全面展开建筑从业人员信息录入工作；建筑从业人员信息以第二代身份证为

基础，要采用人体生物特征（虹膜/人脸特征等）上传至抚州市建筑工地监管信息化一体平台。实名制管理基本信息主要包括姓名、年龄、身份证号码、籍贯、家庭地址、文化程度、培训信息、从业经历、技能水平、劳动合同、不良及良好行为记录等。实时上传施工现场人员流动信息。所有进出施工现场的建筑从业人员都应通过考勤通道进行考勤，考勤数据通过市平台实时传输至工程项目属区、县级主管部门平台，实现抚州地区范围内数据与省、国家实名制管理平台范围内数据互联共享，并确保数据的安全、真实、准确、完整、及时。

二、建设模式及取费标准

各县、区建设主管部门统一使用抚州市建筑工地监管信息化一体平台（以下简称“市平台”）县、区监管账号登录使用，不再重复建设。施工总承包企业递交账号申请表由上级建设主管部门免费开通项目端账号，施工总承包企业要通过自行筹建、已建、自主确立本单位现场相关硬件设备，同时确保硬件设备与上级主管部门平台系统可靠对接。

加强建筑施工扬尘治理过程控制，凡在建项目工地开展扬尘治理所采用的各类专业设备运行所需的费用均应依据《关于明确建筑工地扬尘治理相关费用计取事项的通知》（赣建价〔2019〕7号）按实签证计价，安装的扬尘视频监控系统使用费、维护费计入工程造价。

实施建筑工人实名制管理所需费用参考《关于印发建筑工人实名制管理办法（试行）的通知》第十三条可列入安全

文明施工费和管理费。

三、实施细则

（一）扬尘监测、视频监控管理

全市新开工建设、已开工建设的房屋建筑工程和市政建设工程，百分百必须安装视频监控系统，安装规格按建设部《建筑工程施工现场视频监控技术规范》（JGJ/T292-2012）执行、监控点应覆盖（大门、生活区、起重设备、作业面、钢筋加工区、木材加工区、材料堆放区）。开工后有扬尘的场所必须按照相关标准安装符合要求的扬尘噪音监控设备，设备需联动如雾炮机、喷淋等降尘设备，形成闭环的降尘系统，并将建筑工程施工现场视频监控数据、环境监测数据通过网络实时传输至抚州市建筑工地监管信息化一体平台。（技术标准详见附件）

（二）建筑从业人员实名制管理

全市所有新开工建设、已开工建设的房屋建筑工程和市政建设工程统一采用上级主管部门免费开通的项目端账号，采用符合要求的注册设备对全体从业人员进行基础信息（人员信息、资质信息、合同信息、实名制管理有关必要信息）录入，考勤数据实时上传市平台，由市平台统一上传至全国劳务实名制管理平台，关键岗位人员考勤信息纳入建筑市场诚信管理。（技术标准详见附件）

四、工作安排

（一）第一阶段

抚州市建筑工地监管信息化一体平台试运行。采取边运行、边调试、边完善的模式。尽快投入使用，在试运行期间完善工程建设项目信息以及建设单位施工单位监管单位等信息。各相关科室按照职责和工作实际做好全市所有在建工地通过抚州市建筑工地监管信息化一体平台工地端项目账号完成基础信息（人员信息、资质信息、合同信息、实名制管理有关必要信息）数据采集录入工作。（完成时限：2021年6月30日）

（二）第二阶段

抚州市建筑工地监管信息化一体平台正式运行在试运行基础上。经对功能模块的修改完善，实现系统正式运行。系统正式运行前，对相关工作人员进行操作培训，全面掌握操作流程，熟练操作程序。由传统监管向信息化监管过度。全市所有在建工地实现扬尘监测、视频监控、劳务实名制数据实时上传至抚州市建筑工地监管信息化一体平台，全面实现三大数据远程实时监管要求，11月31日开始全面检查并对各县、区上线率进行考核。（完成时限：2021年11月31日）

（三）第三阶段

进一步扩展工地信息化建设的其他功能，如：特种设备

安全监管、保障农民工工资支付监管等，真正实现建筑工地人员管理、环境管理、设备管理、农民工工资支付管理全方位的智慧化监管，从而达到建筑工地监管信息化智慧化的监管要求。

五、明确管理责任

各县、区住建局和有关职能部门要加强落实各项保障措施，多形式、多渠道加强建筑信息化平台的宣传推广，为建筑信息化建设营造良好的舆论环境，保质保量完成工作目标，稳步有序推进建筑信息化建设工作。并将抚州市建筑工地监管信息化一体平台，用于全市在建工地的数据共享。提高政府监管信息化水平。

施工总承包企业要加强现场信息化建设管理，根据工程类型、规模按标准规范编制信息化建设整体方案。并配备相应的管理人员，查看本单位建筑工地相关监控监测设备运转情况。确保施工现场的硬件配套设施符合平台介入要求。

各建设单位对建筑工地设备安装和使用负总责。按照设备基本技术标准和数据接口协议要求，自主配备满足功能规范和接入标准的设备。并保证设备正常运转，各施工单位要做好设备的日常维护，不得私自变更设备位置和方向，同时指派专人负责设备管理，每天均要检查并做好记录，各监理单位要督促建设单位，按时完成设备接入，监督施工单位的设备正常使用。

施工总承包企业对工程项目所招用的全部建筑从业人员实名制管理负总责，组织建立健全实名制管理组织，按工

工程项目配备专职管理人员，对所招用的全部从业人员实施实名制管理；工程项目经理负责具体承建工程项目所有从业人员的实名制信息录入、现场管理和考勤记录等；分包企业对自主招用的建筑从业人员实名制管理负具体责任，配备专职工管理人員，认真采集并及时向总承包企业报送建筑从业实名制管理所需的信息，确保信息实时更新。

六、加强监督管理

加强监管，注重实效。各县区建设主管部门要加强推进“六个百分百”工作的监督检查，促进企业形成制度并不断加以完善。对于推进“六个百分百”工作开展不力的单位和项目部，要约谈单位主要负责人和项目负责人，责令限期整改。

建立健全长效管理机制。按照“示范引路，以点带面，全面推开”的要求，建立实名制管理、全员劳动合同等长效管理机制，对在建工程项目逐一进行监督检查，定期进行通报。对限期整改不到位或未开展工作的企业进行通报，记入不良行为档案，同时年内不得参加各类评先评优活动。

附件：视频监控系统基本技术标准

视频监控数据接入通信协议

环境监测设备基本技术标准

环境监测设备数据接入通讯协议

实名制考勤设备基本技术标准

实名制考勤设备数据接入通讯协议

智慧工地设备安装及拆除告知表
账号申请表



(此件主动公开)

附件 1:

视频监控系统基本技术标准

一、高清红外枪式网络摄像机

1. 红外照射距离 ≥ 30 米;
2. 不低于 200 万像素 CMOS;
3. 视频本地数据回放分辨率不应低于 1920x1080;
4. 防护等级不低于 IP66;
5. 支持视频压缩标准: H. 264/H. 265/MJPEG/JPEG、;

二、高清红外球型网络摄像机

6. 红外照射距离 ≥ 100 米;
7. 视频本地数据回放分辨率不应低于 1920x1080;
8. 防护等级不低于 IP66;
9. 支持视频压缩标准: H. 264/H. 265/MJPEG/JPEG;
10. 不低于 20 倍光学变焦;
11. 提供云台控制功能, 可实现调节摄像头的旋转角度、镜头景深远近等;
12. 在制高点布置不少于 1 台高清球形摄像机, 水平支持 360° 旋转, 支持巡航、守望位、线扫、画面冻结功能。

附件 2:

视频监控数据接入通信协议

V1.0

版本	说明	时间
V1.0	/	2021.01.20

1. 设备上报 (设备 -> 平台)

设备端主动链接平台转发服务的设备上报端口（一般默认是 2009），链接成功后发送 64 字节的校验信息，具体如下：

```
typedef struct _net_device_type_info
{
    int headFlag; //取值DEVICE_TYPE_HEADFLAG
    int productSeries; //产系列，填0
    int producttype; //产品型号
    int netprotocolver; //协议版本
    int configver; //配置版本
    int devID; //分配的设备ID(平台中添加的主动报设备时的ID)
    int ulReserve[40]; //保留信息
}NETDEVICE_TYPE_INFO
```

2. 命令数据格式组成 (平台 <-> 设备)

命令交互数据采用二进制结构体组合：NET_DATAHEAD + PACKCMD + 具体命令附带数据结构体，比如登陆数据举例是：NET_DATAHEAD + PACKCMD + NET_LOGIN_INFO

```
typedef struct _tagNET_DATAHEAD
{
    char debug_head[4]; //调试头“1111”
    int data_len; //数据包长度，不含本结构后面长度。
```

心跳包填 0，其他数据填实际长度

```
    }NET_DATAHEAD;

    typedef struct _pack_cmd
    {
        unsigned int cmdType;//命令号，如登陆命令
        CMD_REQUEST_LOGIN
        unsigned int cmdID;//指令序号，指令发起端设
        定的一个数值，对应的指令返回头中的指令序号与它完全相
        同
        unsigned int cmdVer;//命令版本
        unsigned int dataLen;//数据长度，不含本结构
        后面长度。
    }PACKCMD;
```

3. 心跳包（平台 <-> 设备）

心跳包只发送 NET_DATAHEAD 结构体数据，8 字节，其
中 data_len 填 0

4. 登陆（平台 -> 设备）

采取平台登陆设备的方式，平台发送登陆命令数据给设
备，设备返回是否成功。

命令号：CMD_REQUEST_LOGIN 0x101

附带数据结构体：

//登录时用的结构体

```

typedef struct _net_login_info
{
    unsigned int connectType; //连接类型,5
    unsigned int IP;         //IP 地址
    char username[36];       //设备用户名
    char password[36];       //设备密码
    char computerName[28];   //计算机名称
    unsigned char MAC[6];    //网卡 Mac 地址
    unsigned charresv[2];    //保留字节, 对齐用
    unsigned int netprotocolver; //目前使用的网
网络协议版本
}NET_LOGIN_INFO, *PNET_LOGIN_INFO;

```

设备返回登陆成功时命令号

CMD_REPLY_LONGIN_SUCC 0x10001 //成功

附带数据结构体:

//用户登录成功时, 回复用的结构体

```

typedef struct _net_login_success_reply
{
    unsigned int authority; //用户网络权限
    long long authLiveCH; //手动录像控制
    long long authRecordCH; //手动录像控制
    long long authPlaybackCH; //检索和回放
    long long authBackupCH; //备份

```

```

        long long    authPTZCtrlCH; //云台控制
        long long    authRemoteViewCH; //远程预览
        PRODUCT_INFO productInfo; //产品信息
    }NET_LOGIN_SUCCESS_REPLY;
    typedef struct _product_info
    {
        unsigned char localVideoInputNum; // 视频输入通道数目
        unsigned char audioInputNum; // 音频输入通道数目
        unsigned char sensorInputNum; // 传感器输入通道数目
        unsigned char relayOutputNum; // 继电器输出数目
        tuint32 displayResolutionMask; // 可选择的分辨率
        unsigned char videoOutputNum; // 视频输出数目（及支持回放最大通道数）
        unsigned char netVideoOutputNum; // 网络回放最大通道数目
        unsigned char netVideoInputNum; // 数字信号接入通道数目
        unsigned char IVSNum; // 智能分析通道数目
    }

```

```
    unsigned char presetNumOneCH;        //每个通道预  
置点数目  
    unsigned char cruiseNumOneCH;       //每个通道巡  
航线数目  
    unsigned char presetNumOneCruise;   //每个巡  
航线的预置点数目  
    unsigned char trackNumOneCH;        //每个通道轨  
迹数目  
  
    unsigned char userNum;               //用户数目  
    unsigned char netClientNum;         //最多客户端数  
目  
    unsigned char netFirstStreamNum;    //主码流传输的  
通道最大数目，即同时可以有几个客户端查看主码流  
    unsigned char deviceType;           //设备类型  
  
    unsigned char doBlueStream;         //是否有提供双码  
流  
    unsigned char audioStream;          //是否有提供音频  
流  
    unsigned char talkAudio;            //是否有对讲功能：  
1 表示有对讲功能；0 表示没有  
    unsigned char bPasswordCheck;       //操作是否要输  
入密码
```

```

unsigned chardefBrightness;    //缺省亮度
unsigned chardefContrast;      //缺省对比度
unsigned chardefSaturation;    //缺省饱和度
unsigned chardefHue;          //缺省色调

    tuint16 videoInputNum;      //视频输入通道数
目（本地加网络）
    tuint16 deviceId;          //设备 ID 号
    tuint32  videoFormat;      //系统当前
制式
    tuint32 function[8];      //功能描述

    tuint32 deviceIP;          //设备网络地址
    unsigned chardeviceMAC[8]; //设备物理地址

    tuint32 buildDate;         //创建日期:year<<16
+ month<<8 + mday
    tuint32 buildTime;         //创建时间:hour<<16
+ min<<8 + sec

    char    deviceName[36];    //设备名称

    char    firmwareVersion[36]; //固件版本

```



```

char kernelVersion[64];    //内核版本
char hardwareVersion[36]; //硬件版本
char MCUVersion[36];      //单片机版本
}PRODUCT_INFO;
登陆返回失败时命令号
    CMD_REPLY_LONGIN_FAIL 0x10002 //拒绝
附带数据结构体:
typedef struct _net_reply
{
    unsigned int devReplyCode; //状态码
}NET_REPLY_INFO;

```

5. 请求预览视频 (平台 -> 设备)

平台向设备发送请求预览

命令号: CMD_REQUEST_STREAM_START 0x201

附带结构体:

```

typedef struct _net_livestream_ctrl_info
{
    unsigned int streamID; //0x0
    long long    fastVideoCH; //用位表示各通道
的状态_主码流
    long long    secondVideoCH; //用位表示各通
道的状态_子码流
    long long    threeVideoCH; //用位表示各通道

```

的状态_三码流

```
long long audioCH; //用位表示各通道
```

的状态_音频流

```
}NET_LIVESTREAM_CTRL_INFO;
```

```
设备回复命令号: CMD_REPLY_LAST_STREAM_INFO
```

0x50003

附带数据结构体:

```
typedef struct _net_reply
```

```
{
```

```
unsigned int devReplyCode; //状态码
```

```
}NET_REPLY_INFO;
```

```
成功状态码填: NETERR_STREAM_OK = 1000
```

6. 预览流数据 (设备 -> 平台)

在平台向设备请求预览, 设备返回成功之后, 设备需要把通道预览数据流发送到平台

```
命令号: CMD_REPLY_DATA_STREAM 0xa000001
```

附带数据结构:

```
FRAME_INFO_EX + 实际预览视频裸数据
```

```
typedef struct _frame_info_ex
```

```
{
```

```
tuint32 keyFrame; //0 不是关键帧
```

```
tuint32 frameType; //帧类型, 1 视频, 2 音频, 3
```

视频格式, 4 音频格式

```
tuint32 length; //长度
```

```

    tuint32      width; //宽
    tuint32      height; //高
        unsigned char Data[4]; //忽略
    tuint32      channel; //通道号
    tuint32      bufIndex; //忽略
    tuint32      frameIndex; //帧索引
    tuint32      frameAttrib;
    tuint32      streamID; //0x0
        long long time; //在帧绝对时间, 在改变设备时
间时会变化
        long long relativeTime; //相对时间, 在改变设
备时间时不会变化, 是个连续的时间
    }FRAME_INFO_EX;

```

7. 停止预览视频 (平台 -> 设备)

命令号: CMD_REQUEST_STREAM_STOP 0x203

附带数据结构:

```

typedef struct _net_livestream_ctrl_info
{
    unsigned int streamID; //0x0
    long long fastVideoCH; //用位表示各通道
的状态_主码流
    long long secondVideoCH; //用位表示各通
道的状态_子码流
    long long threeVideoCH; //用位表示各通道

```

的状态_三码流

```
long long audioCH; //用位表示各通道
```

的状态_音频流

```
}NET_LIVESTREAM_CTRL_INFO;
```

8. 录像事件搜索 (平台 -> 设备)

平台向设备发送搜索录像事件的请求

命令号: CMD_REQUEST_DATA_SEARCH 0x501

附带数据结构体:

```
typedef struct _net_data_search
```

```
{
```

```
    unsigned int searchType; //搜索类型, DATA_SEARCH_TYPE_EVENT (0x02)
```

```
    unsigned int streamID; //要求发送的 stream ID
```

```
    long long videoCH; //视频通道条件
```

```
    long long audioCH; //音频通道条件
```

```
    unsigned int startTime; //开始时间
```

```
    unsigned int endTime; //结束时间
```

```
    unsigned int dataType; //数据类型回放 0 备份 1
```

```
    unsigned int RecordTypeMASK; //录像类型组合, 所有类型可以填 0xFF
```

```
    unsigned int dataOwnerby; //0x0
```

```
}NET_DATA_SEARCH;
```

设备回复命令号:

如果没有数据命令号: CMD_REPLY_DATA_NULL 0x050002

如果有数据命令号: CMD_REPLY_DATA_INFO 0x050001

附带结构体:

NET_DATA_INFO + RECLOG_INFO_EX*n (n 为事件的个数)

```
typedef struct _net_data_info
{
    unsigned int dataType; // 数据类型 ,
    DATA_SEARCH_TYPE_EVENT=0x02
    unsigned int end; //0x0
    unsigned int index; //0x0
    unsigned int dataNum; // 数据个数, 接着的
    RECLOG_INFO_EX 的个数
}NET_DATA_INFO;
typedef struct _reclog_info_ex
{
    unsigned int[2] resv; //0x0
    unsigned int type; //类型组合, RECORD_TYPE
    unsigned int chnn; //通道号
    unsigned int startTime; //开始时间
    unsigned int endTime; //结束时间
}RECLOG_INFO_EX;
//录像数据类型
typedef enum _record_type
```

```

{
    RECORD_TYPE_NULL    = 0x0000,
    RECORD_TYPE_MANUAL  = 0x0001,    //手动录像
    RECORD_TYPE_SCHEDULE= 0x0002,    //排程录像
    RECORD_TYPE_MOTION  = 0x0004,    //移动侦测录像
    RECORD_TYPE_SENSOR  = 0x0008,    //传感器报警录
像
    RECORD_TYPE_END     = 0x0010,    //表示结束,必须在最后
    RECORD_TYPE_ALL     = 0xffff
}RECORD_TYPE;

```

9. 请求录像回放 (平台 -> 设备)

平台向设备请求回放录像

命令号: CMD_REQUEST_DATA_READ= 0x502

附带结构体:

```

typedef struct _net_data_search
{
    unsigned int searchType; // 类型 ,
    DATA_SEARCH_TYPE_TIME=0x01)
    unsigned int streamID; //要求发送的 stream
ID
    long long      videoCH; //视频通道条件
    long long      audioCH; //音频通道条件
    unsigned int startTime; //开始时间

```

```

        unsigned int endTime; //结束时间
        unsigned int dataType; //数据类型回放 0 备份
1
        unsigned int RecordTypeMASK; //录像类型组合,
        所有类型可以填 0xFF
        unsigned int dataOwnerby; //0x0
    }NET_DATA_SEARCH;
    设备回复命令号: CMD_REPLY_LAST_STREAM_INFO
0x50003

```

附带数据结构体:

```

typedef struct _net_reply
{
    unsigned int devReplyCode; //状态码
}NET_REPLY_INFO;
    成功状态码填: NETERR_STREAM_OK = 1000

```

10. 录像回放数据流 (设备 -> 平台)

当平台请求录像回放, 设备返回成功之后, 设备需要把录像回放数据流发送到平台

命令号: CMD_REPLY_DATA_STREAM 0xa000001

FRAME_INFO_EX + 实际回放视频裸数据

```

typedef struct _frame_info_ex
{
    tuint32 keyFrame; //1 关键帧, 0 非关键帧
    tuint32 frameType; //帧类型, 1 视频, 2 音频, 3

```

视频格式，4 音频格式

```
    tuint32    length; //长度
    tuint32    width;  //宽
    tuint32    height; //高
    unsigned char Data[4]; //忽略
    tuint32    channel; //通道号
    tuint32    bufIndex; //忽略
    tuint32    frameIndex; //帧索引
    tuint32    frameAttrib;
    tuint32    streamID; //流 ID, 回传请求时的流 ID
    long long time; //在帧绝对时间, 在改变设备时间时会变化
    long long relativeTime; //相对时间, 在改变设备时间时不会变化, 是个连续的时间
}FRAME_INFO_EX;
```

11. 停止录像回放 (平台 -> 设备)

平台向设备发送停止回放命令

命令号: CMD_REQUEST_DATA_STOP= 0x0507

附带数据结构:

```
typedef struct _net_data_search
{
    unsigned int    searchType; //0x0
    unsigned int    streamID; //stream ID, 请求时的 ID
```



```
long long      videoCH; //0x0
long long      audioCH; //0x0
unsigned int    startTime; //0x0
unsigned int    endTime; //0x0
    unsigned int    dataType; //0x0
unsigned int    RecordTypeMASK; // 0x0
unsigned int    dataOwnerby; //0x0
}NET_DATA_SEARCH;
```

主要参数是 streamID 为当初回放时的 streamID。

附件 3:

环境监测设备技术标准

- 1、对噪声、PM2.5、PM10、温度、湿度、风速、风向、气压、PST 等指标实时监测；
- 2、具备实时传输监测数据能力；
- 3、GPRS/4G 方式数据上传平台。
- 4、环保监测指标超限值后自动预警并开启喷淋和雾炮；
- 5、必须具有国家 CPA 证书。

附件 4:

环境监测设备数据接入通讯协议

V1.0

版本	说明	时间
V1.0	/	2021.01.20

1. 通讯方式

1.1、与上位机通讯方式

安装在现场的环境监测设备通过传输网络将数据与上位机进行通讯

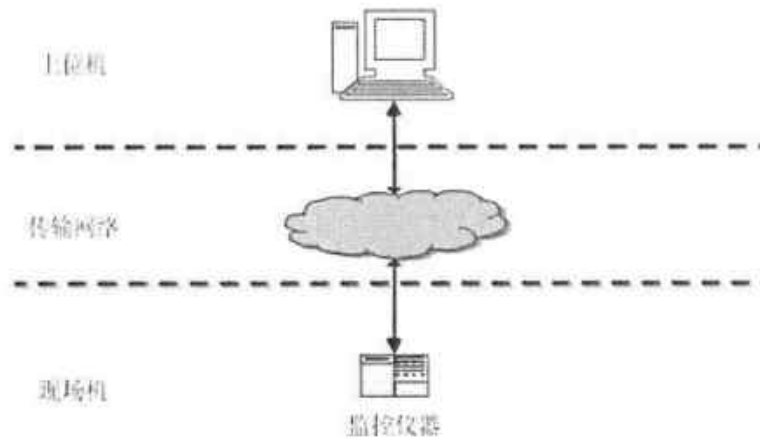


图1 数据上传示意图

1.2、通讯数据结构

所有的通讯包都是由 ASCII 码（汉字除外，采用 UTF-8 码，8 位，1 字节）字符组成。通讯协议数据结构如图 4 所示。

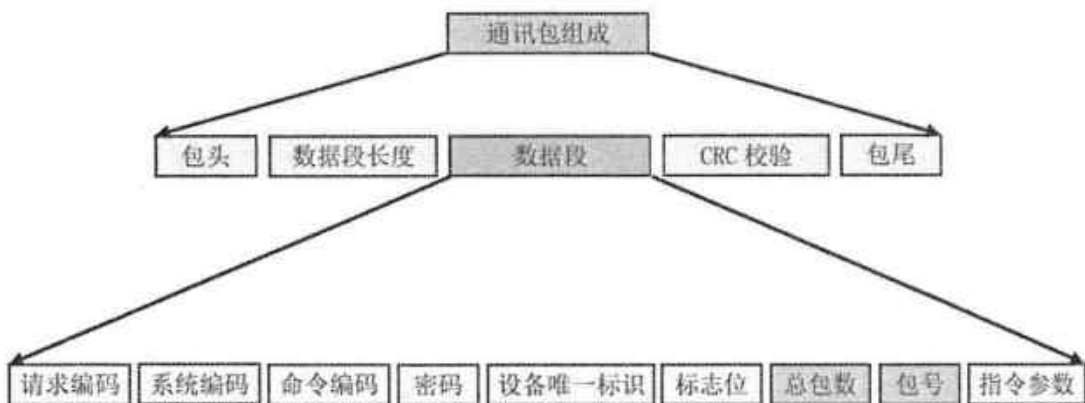


图 2 通讯协议数据结

1.3、报文格式

具体报文格式参考《(HJ_T 212-2017) 污染物在线监控(监测)系统数据传输标准》标准格式。其中 TSP, PM2.5, PM10 单位为微克每立方米($\mu\text{g}/\text{m}^3$)。ST=22; CN=2011; 每分钟上传一次实时数据,每五分钟上传一次五分钟平均值;每小时上传一次小时平均值。

1.4、服务器地址

设备使用 TCP 与服务器交互,服务器 IP 地址:由平台提供。

1.5、MN 编码规则

MN 使用 14 位编码,前 9 位由平台指定以区别厂家,后 5 位由设备厂家自己维护,确保唯一。

1.6、传输要求

设备直传平台,切勿经过第三方平台进行数据转发。

2. 参数定义

名称	类型	长度	描述
包头	字符	2	固定为##
数据段长度	十进制整数	4	数据段的 ASCII 字符数,例如:长 255,则写为“0255”
数据段	字符	$0 \leq n \leq 1024$	变长的数据 参照 2-2 数据段数据结构表

CRC 校验	十六进制整数	4	数据段的校验结果 参照附录
包尾	字符	2	固定为<CR><LF> (回车、换行)

2.1 通讯包结构组成表:

2.2 数据段结构组成表:

名称	类型	长度	描述
请求编码 QN	字符	20	精确到毫秒的时间戳:QN=YYYYMMDDhhmmsszzz, 用来唯一标识一次命令交互
设备唯一标识 MN	字符	22	MN=平台下发的设备编码。
访问密码 PW	字符	9	PW=访问密码
指令参数 CP	字符	$0 \leq n \leq 950$	CP=数据区&&, 参照 2-3 数据区数据结构

2.3 数据段描述符

名称	类型	长度	描述
a01001-xxx	浮点数	Rtd 代表实时数据 -Avg 代表平均数据	温度

a01002-xxx	浮点数	Rtd 代表实时数据 -Avg 代表平均数据	湿度
a01006-xxx	浮点数	Rtd 代表实时数据 -Avg 代表平均数据	气压
a01007-xxx	浮点数	Rtd 代表实时数据 -Avg 代表平均数据	风速
a01008-xxx	浮点数	Rtd 代表实时数据 -Avg 代表平均数据	风向
a34001-xxx	浮点数	Rtd 代表实时数据 -Avg 代表平均数据	总悬浮颗粒物 TSP
a34002-xxx	浮点数	Rtd 代表实时数据 -Avg 代表平均数据	可吸入颗粒物 PM10
a34004-xxx	浮点数	Rtd 代表实时数据 -Avg 代表平均数据	细微颗粒物 PM2.5
LA-xxx	浮点数	Rtd 代表实时数据 -Avg 代表平均数据	噪声

3. 协议格式示例

```
##0211ST=22 ; CN=2011 ; QN=20171201141917519 ; PW=123
456 ; MN=00000
0000000000017031024 ; F l a g=5 ; C P=&&D a t a T i m e=201
```

```

71201141917; a01
001-Rtd=5. 0; a01002-Rtd=54. 0; a34002-Rtd=4
6; a34004-Rtd=67; a01
007-Rtd=1. 1; a01008-Rtd=320; LA-Rtd=58. 0&&B
E01

```

注：双斜杠后面的仅仅为简单参数说明，具体参数说明可参照 3-1、3-2、3-3 表中的说明实时数据包，1 分钟一次

```

## // 包头
0211 // 数据段字节长度
//=====数据段内容=====
ST=22; // 污染物类别 31 气体污染物
CN=2011; // 命令编码 2011 实时数据
QN=20171201141917519; // 发送数据包的时间
PW=123456;
// PW 码 默认 123456 MN=00000 0000000000017031024;
// 设备唯一标识码
Flag=5; // 要求平台回复指令
CP=&&DataTime=20171201141917; // 数据采集时间 a34004-Rtd=67; // PM2.5 实时数据
a34002-Rtd=46; //
PM10 实时数据

```



```

a01001-Rtd=5.0; // 温度 实时数据

a01002-Rtd=54.0; // 湿度
实时数据

LA-Rtd=58.0; //
噪声 实时数据

a01007-Rtd=1.1; // 风
速 实时数据

a01008-Rtd=320 // 风向
实时数据 数据排列不分先后

&&

//=====数据段内容=====

1E00

```

4. CRC 校验代码 c 语言

CRC 校验算法示例:

```

/*****
*****

```

函数: CRC16_Checkout

描述: CRC16 循环冗余校验算法。

参数一: *puchMsg:

需要校验的字符串指

针 参 数 二 :

usDataLen: 要校验的

字符串长度返回 值:

返回 CRC16 校验码

```
*****
```

```
*****/
```

```
unsigned int CRC16_Checkout ( unsigned char *puchMsg, unsigned int  
usDataLen )
```

```
{
```

```
    unsigned int i,j,crc_reg,check;
```

```
    crc_reg = 0xFFFF;
```

```
    for(i=0;i<usDataLen;i++)
```

```
    {
```

```
        crc_reg = (crc_reg>>8) ^ puchMsg[i];
```

```
        for(j=0;j<8;j++)
```

```
        {
```

```
            check = crc_reg & 0x0001;
```

```
            crc_reg >>= 1;
```

```
            if(check==0x0001)
```

```
            {
```

```
                crc_reg ^= 0xA001;
```

```
            }
```

```
        }
```

```
    }
```

```

return crc_reg;
}

```

5. ST 码编码方法

系统名称	系统编码	描述
地表水质量监测	21	
空气质量监测	22	
声环境质量监测	23	
地下水质量监测	24	
土壤质量监测	25	
海水质量监测	26	
挥发性有机物监测	27	
大气环境污染源	31	

6. CN 码编码方法

命令编码表

命令名称	命令编码	命令类型	描述
提取现场机时间	1011	请求命令	用于提取现场机的系统时间
上传现场机时间	1011	上传命令	用于上传现场机时间
设置现场机时间	1012	请求命令	用于设置现场机的系统时间

现场机时间校准请求	1013	通知命令	用于向上位机发送现场机时间校准请求
提取实时数据间隔	1061	请求命令	提取实时数据间隔
上传实时数据间隔	1061	上传命令	上传实时数据间隔
设置实时数据间隔	1062	请求命令	指定实时数据间隔
提取分钟数据间隔	1063	请求命令	提取分钟数据间隔
上传分钟数据间隔	1063	上传命令	上传分钟数据间隔
取污染物实时数据	2011	请求命令	用于启动现场机上传实时数据
上传污染物实时数据	2011	上传命令	用于现场机上传污染物实时数据
停止察看污染物实时数据	2012	通知命令	用于停止现场机实时数据上传
取设备运行状态数据	2021	请求命令	用于启动现场机上传污染治理设施运行 状态
上传设备运行状态数据	2021	上传命令	用于现场机上传污染治理设施运行状态

附件 5:

实名制考勤设备技术标准

1. 具有可见光及红外双镜头的生物识别设备;
2. 识别距离 0.3m~1.5m;
3. 具有活体检测, 能够识别相片、视频、面具等伪装素材;
4. 人脸库容量: 2.4 万条以上, 识别记录: 6 万条以上, 考勤数据本地存储 30 天以上;
5. 识别精准快速, 识别速度 $\leq 1S$, 通过率高于 99%;
6. 工作温度: $-35^{\circ}C-60^{\circ}C$;
7. 满足户外使用环境, 防水防尘, 防水等级: $\geq IP66$;
8. 支持以太网通讯模块;
9. 支持断网本地存储和联网续传;
10. 考勤数据远程上传, 上传延时不超过 10 秒钟;

附件 6:

实名制考勤设备数据接入通讯协议

V1.0

版本	说明	时间
V1.0	/	2021.01.20

一、 接口说明

- 1) 通信过程中 PC 为服务器，设备为客户端
- 2) 使用 TCP 协议短链接方式进行通信
- 3) 为了保证传输过程中的数据安全，接口中的 JSON 格式的包体数据设定为使用 DES 进行加密，key 为 12345678
- 4) 字符编码格式 UTF-8
- 5) 单包数据(包头+包体+包尾)小于 10K
- 6) 在连续发送多包数据时，第一包包体为数据，后续包中为二进制数据，不需要加密处理

二、 接口数据包格式

1) 包头

请求包包头

内容	长度	备注	其它
包头标记	2 byte	0xDDCC	
数据包长度	2 byte	0x0000~0x2800	最大 0x2800 (包头+包体+包尾)
命令字	1 byte	0x00~0xFF	最大 0xFF
分包序号	2 byte	0x0000~0xFFFF	0x0000
分包总数	2 byte	0x0000~0xFFFF	0x0000
设备状态	1 byte	0x00~0xFF	默认 0x01
设备 SN 编号	16byte		

应答包包头

内容	长度	备注	其它
----	----	----	----

包头标记	2 byte	0xCCDD	
数据包长度	2 byte	0x0000~0x2800	最大 0x2800 (包头+包体+包尾)
命令字	1 byte	0x00~0xFF	最大 0xFF
分包序号	2 byte	0x0000~0xFFFF	0x0000
分包总数	2 byte	0x0000~0xFFFF	0x0000
执行结果返回编码或后续命令字	1 byte	0x00~0xFF	0x00: 服务器空闲(主要是心跳包使用) 0x01: 成功 0x02: 失败 0x10~0xFF: 为后续命令字
设备 SN 编号	16byte		

2) 包体

JSON 格式数据或二进制数据

3) 包尾

内容	长度	备注	其它
数据校验	1 byte	0x00~0xFF	包头+包体 Xor 校验运算 U8 calcChecksum(unsigned char *pkg, int len) { U8 BBC = 0; for(int i = 0; i < len; i++) {

			<pre> BBC = pkg[i]; } return BBC; } </pre>
--	--	--	---

三、接口详情

3.1 心跳包

设备主动发送请求到服务器，心跳间隔为 2-5 秒，设备在执行服务器下发的指令的时候不能再向服务器发送心跳包

(一) 请求(设备->服务器):

a) 请求命令: 0x10

b) 请求实例: 0xDDCC 包长

0x100x00000x00000x01SN16Byte 二进制数据校验

c) 请求包体数据格式: {"sn":"","date_time":""}

包体数据详细说明:

序号	参数	类型	说明
1	sn	string	设备序列号
2	date_time	datetime	设备请求时间, 格式为: yyyy-mm-dd hh:mm:ss

(二) 返回示例(服务器->设备):

0: 空闲(无后续命令)

返回数据: 0xCCDD 包长

0x100x00000x00000x00SN16Byte 校验

无包体

1: 成功 (查看包体中的详细信息-有后续命令)

返回数据: 0xCCDD 包长

0x100x00000x00000x01SN16Byte 二进制数据校验

返回包体数据格式:

```
{"cmd": "", "params": [{"date_time": ""}]}
```

包体详细说明:

序号	参数	说明
1	cmd	服务器端发送给需要设备执行的命令
2	params	服务器端发送给需要设备执行的命令的参数

2: 失败 (查看包体中的详细信息-无后续命令)

返回数据: 0xCCDD 包长

0x100x00000x00000x02SN16Byte 二进制数据校验

包体数据格式: {"result_code": "",
"result_desc": ""}

包体详细说明:

序号	参数	类型	说明
1	result_code	int	错误编码, 请参考错误编码说明
2	result_desc	string	描述

3.2 人员批量下发

根据心跳包返回数据到设备

(一) 返回结果 (服务器->设备):

1. 返回命令: 0x25

2. 返回数据 : 0xCCDD 包长

0x250x00000x00000x01SN16Byte 二进制数据校验

3. 返回数据包体格式:

```
{
  "count": "",
  "users": [{
    "user_id": "",
    "name": "",
    "work_sn": "",
    "iris_template": "",
    "ic": "",
    "id": "",
    "id_card": "",
    "phone_card": "",
    "iris_count": "",
    "enroll_type": "",
    "face_type": "",
    "face_template": "",
    "photo": "",
    "weigen": ""
  }]
}
```

包体数据详细说明:

序号	属性	类型	名称	说明
----	----	----	----	----

1	count	string	人员数	
2	user_id	string	用户 ID	用户唯一 ID
3	name	string	姓名	
4	work_sn	string	用户工号	和 user_id 一样
5	iris_template	string	虹膜模板	十六进制
6	ic	string	IC 卡号	
7	id_card	string	身份证号	
8	phone_card	string	手机卡号	
9	iris_count	int	虹膜数量	默认:1
10	enroll_type	string	注册类型	<p>默认值: 1000000000000000</p> <p>说明: 注册类型, 共 16 位, 每一位代表一种类型, 分别是, "虹膜", "人脸", "指纹", "IC 卡", "指静脉", "密码", "身份证", "一代手机卡", "二代手机卡", "ID 卡", "红外人脸" 后面剩余的暂时保留</p> <p>如果为 0: 未注册, 1: 已经注册</p>
11	face_type	string	人脸模板下发类型	<p>0: 特征模板 (16 进制字符串)</p> <p>1: url (不支持)</p> <p>2: 图片数据, Base64</p> <p>为空时默认 0</p>
12	face_template	string	可见光人脸模	1. 如果 face_type 为 0, 则 face_template

			板	就是照片模板信息，格式为：十六进制字符串 2. 如果 face_type 为 2 则 face_template 就是图片 base64 数据
13	facel_template	string	红外人脸模板	1. 如果 face_type 为 0 ， 则 facel_template 就是红外照片模板信息，如果没有红外照片，这边字段信息和 face_template 保持一致即可，格式为：十六进制字符串 2. 如果 face_type 为 2 ， 则 facel_templat 为空即可
14	picdata	string	人员图像	可以为空，采用 Base64 编码 1. 如果 face_type 为 0, picdata 是头像图片数据，设备识别成功后会显示该图片，如果不传，识别成功不显示照片 2. 如果 face_type 为 2, 这个 picdata 不用传，识别成功显示的照片就是 face_template 字段的图片信息
15	weigen	string	韦根	

(二) 下发结果返回(设备->服务器):

1. 返回命令: 0x25

2. 返回数据: 0xCCDD 包头

0x250x00000x00000x01SN16Byte 加密后的二进制数

数据校验

3. 返回数据格式：`{"result_code":"","result_desc":""}`

包体数据详细说明：

序号	参数	类型	说明
1	result_code	string	错误编码，请参考错误编码说明
2	result_desc	string	描述

3.3 删除人员信息

根据心跳包返回数据到设备

(一) 返回结果（服务器->设备）：

1. 返回命令：0x14

2. 返回数据：`0xCCDD` 包长
`0x140x00000x00000x01` SN16Byte 加密后的二进制数据
数据校验

3. 返回包体数据格式：
`{"type":"","user_id":"","time":""}`，type 默认为：0。

包体规范数据详细说明：

序号	属性	类型	名称	说明
1	type	int	删除类型	0: 默认值 4: 清空设备人员信息
2	user_id	string	用户工号	

3	time	datetime	时间	默认为空
---	------	----------	----	------

(二) 删除结果返回(设备->服务器):

1. 返回命令: 0x14

2. 返回数据: 0xCCDD 包长
0x140x00000x00000x01SN16Byte 加密后的二进制数据
校验

3. 返回包体数据格式: {"result_code": "",
"result_desc": ""}

包体数据详细说明:

序号	参数	类型	说明
1	result_code	int	错误编码, 请参考错误编码说明
2	result_desc	string	描述

3.4 上传识别结果

设备识别后把识别结果上传到服务器 (单包数据数量小于 5 条)

(一) 请求 (设备->服务器):

1. 请求命令: 0x19

2. 请求数据: 0xDDCC 包长
0x190x00000x00000x01SN16Byte 加密后的二进制数据
校验

3. 请求数据格式:

{

```

"sn": "",
"count": "",
"logs": [{
    "user_id": "",
    "recog_time": "",
    "recog_type": "",
    "recog_flag": "",
    "photo": ""
}, {
    "user_id": "",
    "recog_time": "",
    "recog_type": "",
    "recog_flag": "",
    "photo": ""
}]
}

```

包体数据详细说明:

序号	属性	类型	名称	说明
1	count	int	识别记录数量	
2	sn	string	设备序列号	
3	user_id	string	用户 ID	用户唯一 ID
4	recog_time	datetime	识别时间	
5	recog_type	string	识别类型	人脸: face 身份证: ID

				IC卡: IC
6	recog_flag	string	上下班标记	1: 上班(进门), 2: 下班(出门)
7	photo	string	识别照片	可以为空, 采用 Base64 编码

(二) 上传结果返回 (服务器->设备):

1. 返回命令: 0x19

2. 返回数据: 0xCCDD 包长
0x190x00000x00000x01SN16Byte 加密后的二进制数据
CRC校验

3. 返回包体数据格式: {"result_code": "",
"result_desc": ""}

包体数据详细说明:

序号	参数	类型	说明
1	result_code	string	错误编码, 请参考错误编码说明
2	result_desc	string	描述

四、 其它

1、 错误编码说明

序号	编码值	说明
1	0000	成功
2	9999	失败 (无效的数据)
3	9998	未知错误, 请和开发人员联系
4	9001	数据格式错误 [json 解析失败]

5	9021	设置时间失败
6	9002	数据格式错误[json 数据错误]
7	9003	数据文件长度不对
8	9005	数据接收有误,MD5 校验失败
9	9006	操作数据库错误
10	9007	删除设备上的数据失败
11	9008	添加人员信息失败

附件 7:

施工现场设备安装、使用和拆（移）除程序

智慧工地设备的安装、使用、拆（移）除按下列程序办理：

一、制定安装方案。由建设单位、施工单位、监理单位共同确定设备点位的数量、安装位置、安装计划等；施工单位完成智慧工地设备安装方案和点位平面布置图的制作，并填写《抚州市智慧工地设备安装告知表》，报市建设局备案。

二、安装与使用。施工工地同步安装视频监控设备和扬尘污染物在线检测设备，分别与建设主管部门，并保证设备正常运行。其它设备根据施工进度按要求同步安装。

三、安装方案修改。任何单位和个人不得擅自修改经审查的智慧工地设备安装方案和智慧工地设备点平面布置图，如确需变更的，应由施工单位提出申请，经监理单位、建设单位审核，报市建设局同意后方可进行修改。

四、拆除与移机。未经市建设局同意，施工单位不得擅自拆除或移动智慧工地设备。需要拆除或移动设备的，施工单位须提前 3 个工作日内到市建设局办理手续，填写《抚州市智慧工地设备拆除（移机）告知表》，经同意后方可拆除或移机。市建设局在收到申请后，应在 3 个工作日内进行审查并回复申请人。

抚州市智慧工地设备安装告知表

项目名称						
项目地址						
建设单位				项目负责人		
				联系电话		
施工单位				项目负责人		
				联系电话		
监理单位				项目负责人		
				联系电话		
建筑面积 (平方米)		总造价 (万元)		占地面积 (平方米)		
开竣工日期	年 月 日至 年 月 日					
项目	设备类型	品牌型号	数量(个)	计划安装 时间	计划拆除时间	安装点位位置
视频监控	球 枪					
	枪 机					
环境监测	项					
实名制通道	进出					

塔吊监测						
升降机监测						
施工单位 意见	联系人： 电 话： (盖章) 年 月 日			建设单位 意见	联系人： 电 话： (盖章) 年 月 日	
监理单位 意见	联系人： 电 话： (盖章) 年 月 日					
建设主管 部门意见	处室意见：	主管领导意见：		抚州市建设行政主管部门 (盖章) 年 月 日		

1. 本表一式肆份，建设单位、施工单位、监理单位 and 建设主管部门各一份。
2. 本表由施工单位填写，需附智慧工地设备安装方案和平面布置图一份。

抚州市智慧工地设备拆除（移机）告知表

项目名称			
项目地址			
建设单位		项目负责人 联系电话	
施工单位		项目负责人 联系电话	
监理单位		项目负责人 联系电话	
视频监控、环境监控、实名制通道等设备申请拆除（移机）情况	<p style="text-align: center;">（拆除和移机的设备名称、时间、原因、数量、位置等相关情况）</p> 施工单位： 联系人： 电 话： <p style="text-align: right;">（盖章） 年 月 日</p>		
监理单位 意 见	联系人： 电 话： （盖章） <p style="text-align: right;">年 月 日</p>	建设单位 意 见 （盖章） <p style="text-align: right;">年 月 日</p>	联系人： 电 话： （盖章） <p style="text-align: right;">年 月 日</p>
建设主管 部 门 意 见	处室意见：	主管领导意见：	抚州市建设行政主管部门 （盖章） <p style="text-align: right;">年 月 日</p>

1. 本表一式肆份，建设单位、施工单位、监理单位 and 建设主管部门各一份。

2. 本表由施工单位填写，需附移机后的智慧工地设备安装点平面布置图一份。

附件 8:

项目账号申请表

施工总承包单位信息			
单位名称			
单位性质 (多选一)	<input type="checkbox"/> 有限责任公司	<input type="checkbox"/> 股份有限公司	<input type="checkbox"/> 合资公司
	<input type="checkbox"/> 外资企业	<input type="checkbox"/> 国资独立	<input type="checkbox"/> 私营企业
统一社会 信用代码			
联系人		联系电话	
办公电话		企业经营地址	
注册资金		注册时间	
注册地址			
项目信息			
项目名称			
项目活动 类型 (多选一)	<input type="checkbox"/> 施工总承包	<input type="checkbox"/> 工程总承包	<input type="checkbox"/> 设计单位
	<input type="checkbox"/> 监理单位	<input type="checkbox"/> 施工劳务	
项目分类	<input type="checkbox"/> 市政工程	<input type="checkbox"/> 房屋工程	<input type="checkbox"/> 其他
登记日期		开工日期	
计划完工 日期		承包合同额 (万元)	
工程进度	<input type="checkbox"/> 基础施工	<input type="checkbox"/> 主体施工	<input type="checkbox"/> 装饰装修
施工许可			

证			
项目经理		项目经理电话	
详细工程 地址			

联系人：

电话：

填报时间：

***以上信息必须真实、准确。如施工总承包单位名称，统一社会信用代码、项目名称等必须准确，以上全部为必填选项。**