TC\_B Communication Protocol V2.1

Part I Communication Protocol Structure

1> Command format

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH（device code） | CMD（command） | LEN（data length） | DATA | CRC16 |
| 0xA5 | 4Bytes | 1Byte | 2Bytes | 0-400Bytes | 2Bytes |

2> Response format

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH (device code） | ACK（response） | RET（return value） | LEN（data length） | DATA | CRC16 |
| 0xA5 | 4Bytes | 1Byte（command+0x80） | 1Byte | 2Bytes | 0-400Bytes | 2Bytes |

Description：

1. Order of four byte CH：IDHH，IDHL，IDLH，IDLL；
2. CRC16 check means all data CRC16，order of two byte CRC16：CRCL CRCH；
3. When CH is 0，all devices connected will response to this command.
4. RET define as ：

|  |  |  |  |
| --- | --- | --- | --- |
| #define ACK\_SUCCESS | 0x00 |  | // operation successful |
| #define ACK\_FAIL |  | 0x01 | // operation failed |
| #define ACK\_FULL | 0x04 | // user full | |
| #define ACK\_EMPTY | 0x05 | // user empty | |
| #define ACK\_NO\_USER | 0x06 | // user not exist | |
| #define ACK\_TIME\_OUT | 0x08 | //capture timeout | |
| #define ACK\_USER\_OCCUPIED | 0x0A | //user already exists | |
| #define ACK\_FINGER\_OCCUPIED | 0x0B | //fingerprint already exists | |

1. When the RET != ACK\_SUCCESS, the DATA and LEN in the response data are always 0.

Part II Command instruction

# Get the information of T&A device 1 CMD：0x30

Function：Get the firmware version, communication password, sleep time, volume, language, date and time format, attendance state, language setting flag, command version.

Command：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x30 | 0x00 0x00 | CRCL CRCH |

Response：（29Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xB0 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x12 | 18Byte | CRCL CRCH |

Data format：（18Byte）

|  |  |  |
| --- | --- | --- |
| Byte | Data | Description |
| 1-8 | Firmware version | Firmware version is ASC |
| 9-11 | Communication  password and its length | Byte(9) bits 7-4 = password length  Byte (9) bits 3-0 + Byte(10-11) = password |
| 12 | Sleep time | 0-250 minutes, never sleep when set as 0 |
| 13 | Volume | Level 0-5, mute if set as 0 |

|  |  |  |
| --- | --- | --- |
| 14 | Language | Device language, 0-simplified Chinese, 1-Traditional Chinese  2-English, 3-French, 4-Spanish, 5-Portuguese |
| 15 | Date / Time format | Bit 7-4：date format, 0-Chinese, 1-America, 2-English  Bit 3-0：time format, 0-24 hours, 1-12 hours(AM/PM) |
| 16 | Attendance state | 0-15, user edit in software and upload to device |
| 17 | Language setting flag | =0x10, enable language setting, user could modify menu  language, other value would disable this function. (for C2 C3 C5) |
| 18 | Command version | =0x01, would response to 0x22 0x23(for C2 C3 C5)  =0x02, would response to 0x24 0x25 |

# Set the configure information of T&A 1 CMD：0x31

Function：Set the communication password, sleep time, volume, language, date format, attendance state, and language setting flag.

Notice：If you just modify some of the items, for the rest, you may set them as 0xFF. Command：（20Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16  CRCL CRCH |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x31 | 0x00 0x0A | 12Byte |

Data format: (10Byte)

|  |  |  |
| --- | --- | --- |
| Byte | Data | Description |
| 1-3 | Communication  password and its length | Byte(1) bits 7-4 = password length  Byte(1) bits 3-0 + Byte(10-11) = password |
| 4 | Sleep time | 0-250 minutes, never sleep when set as 0 |
| 5 | Volume | Level 0-5, mute if set as 0 |
| 6 | Language | Device language, 0-simplified Chinese, 1-Traditional Chinese  2-English, 3-French, 4-Spanish, 5-Portuguese |
| 7 | Date / Time format | Bit 7-4：date format, 0-Chinese, 1-America, 2-English  Bit 3-0：time format, 0-24 hours, 1-12 hours(AM/PM) |
| 8 | Attendance state | 0-15, user edit in software and upload to device |
| 9 | Language setting flag | =0x10, enable language setting, user could modify menu language, other value would disable this function. (for  C2 C3 C5) |
| 10 | reserved |  |

# Get the information of T&A device 2 CMD：0x32

Function：Get the T&A device Compare Precision, Fixed Wiegand Head Code, Wiegand Option, Work code permission, real-time mode setting, FP auto update setting, relay mode, Lock delay, Memory full alarm, Repeat attendance delay, door sensor delay, scheduled bell delay.

Command：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x32 | 0x00 0x00 | CRCL CRCH |

Response：（26Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xB2 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x0F | 15 Byte | CRCL CRCH |

Data format: （15Byte）

|  |  |  |
| --- | --- | --- |
| Byte | Data | Description |
| 1 | Fingerprint comparison  precision | Level 0-2, 0-low, 1-medium, 2-high |
| 2 | Fixed Wiegand head code | 1-254 |
| 3 | Wiegand option | 0-Wiegand26, 1-Anviz Wiegand, 2-fixed Wiegand  3- if user punch card, output low 26 bits of card ID, otherwise output 00000000+user id(16digit) |
| 4 | Work code permission | 0-disable, 1-enable |
| 5 | Real-time mode setting | 0-disable, 1-enable |
| 6 | FP auto update setting | 0-disable, 1-enable |
| 7 | Relay mode | 0-control lock, 1-scheduled bell |
| 8 | Lock delay | 0-15 seconds, never open lock if set as 0 |
| 9-11 | Memory full alarm | 0-5000, balance record space less than specified value,  device would give warning message |
| 12 | Repeat attendance delay, | 0-250 minutes, within the specified time range, only the  first record would be take as valid record |
| 13 | Door sensor delay | 0-250 seconds, won’t alarm if set as 0 |
| 14 | Scheduled bell delay | 0-15 seconds, won’t ring bell if set as 0 |
| 15 | reserved |  |

# Set the configure information of T&A 2 CMD：0x33

function：Set the T&A device Compare Precision, Fixed Wiegand Head Code, Wiegand Option, Work code permission, real-time mode setting, FP auto update setting, relay mode, Lock delay, Memory full alarm, Repeat attendance delay, door sensor delay, scheduled bell delay.

notice：If you just modify some of the items, for the rest, you may set them as 0xFF. Command：（25Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | Data | CRC16 |
| 0xA5 | IDHH IDHL IDLH  IDLL | 0x33 | 0x00 0x0F | 15Byte | CRCL CRCH |

Data format：（15Byte）

|  |  |  |
| --- | --- | --- |
| Byte | Data | Description |
| 1 | Fingerprint comparison  precision | Level 0-2, 0-low, 1-medium, 2-high |
| 2 | Fixed Wiegand head code | 1-254 |
| 3 | Wiegand option | 0-Wiegand26, 1-Anviz Wiegand, 2-fixed Wiegand 3- if user punch card, output low 26 bits of card ID,  otherwise output 00000000+user id(16digit) |
| 4 | Work code permission | 0-disable, 1-enable |

|  |  |  |
| --- | --- | --- |
| 5 | Real-time mode setting | 0-disable, 1-enable |
| 6 | FP auto update setting | 0-disable, 1-enable |
| 7 | Relay mode | 0-control lock, 1-scheduled bell |
| 8 | Lock delay | 0-15 seconds, never open lock if set as 0 |
| 9-11 | Memory full alarm | 0-5000, balance record space less than specified value,  device would give warning message |
| 12 | Repeat attendance delay, | 0-250 minutes, within the specified time range, only the  first record would be take as valid record |
| 13 | Door sensor delay | 0-250 seconds, won’t alarm if set as 0 |
| 14 | Scheduled bell delay | 0-15 seconds, won’t ring bell if set as 0 |
| 15 | reserved |  |

Response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH  IDLL | 0xB3 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Get the date and time of T&A** CMD：0x38 function：Get the date and time of T&A Command：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH  IDLL | 0x38 | 0x00 0x00 | CRCL CRCH |

Response：（17Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH  IDLL | 0xB8 | ACK\_SUCCESS  ACK\_FAIL | 0x00  0x06 | 6Byte | CRCL CRCH |

Data format：（6Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| DATA | year | month | day | hour | minute | second |
| Byte | 1 | 2 | 3 | 4 | 5 | 6 |

1. **Set the date and time of T&A** CMD：0x39 Function：Set the date and time of T&A Command：（16Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x39 | 0x00 0x06 | 6Byte | CRCL CRCH |

Data format：（6Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| DATA | year | month | day | hour | minute | second |
| Byte | 1 | 2 | 3 | 4 | 5 | 6 |

Response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xB9 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Get TCP/IP parameters** CMD：0x3A

Function：Get the IP address, subnet Mask, MAC address, Default gateway, Server IP address,

Far limit, Com port NO., TCP/IP mode, DHCP limit.

Command：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x3A | 0x00 0x00 | CRCL CRCH |

Response：（38Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xBA | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x1B | 27Byte | CRCL CRCH |

Data format：（27Byte）

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | IP  address | subnet  Mask | MAC  address | Default  gateway | Server IP  address | Far limit | Com port  NO. | TCP/IP  mode | DHCP  limit |
| Byte | 1-4 | 5-8 | 9-14 | 15-18 | 19-22 | 23 | 24-25 | 26 | 27 |

TCP/IP Mode defined as: 0 - sever mode, 1 -client mode.

1. **Set TCP/IP parameters** CMD：0x3B

Function：Get the IP address, subnet Mask, MAC address, Default gateway, Server IP address, Far limit, Com port NO., TCP/IP mode, DHCP limit.

Command：（37Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x3B | 0x00 0x1B | 27Byte | CRCL CRCH |

Data：（27Byte）

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | IP  address | subnet  Mask | MAC  address | Default  gateway | Server IP  address | Far  limit | Com  port NO. | TCP/IP  mode | DHCP  limit |
| Byte | 1-4 | 5-8 | 9-14 | 15-18 | 19-22 | 23 | 24-25 | 26 | 27 |

Response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xBB | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Get record information** CMD：0x3C

Function：Get record information, including the amount of Used User, Used FP, Used Password, Used Card, All Attendance Record, and New Record.

Command：（10 Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x3C | 0x00 0x00 | CRCL CRCH |

Response：（29 Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xBC | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x22 | 18Byte | CRCL CRCH |

Data format：（18Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| DATA | User Amount | FP Amount | Password  Amount | Card  Amount | All Record  Amount | New Record  Amount |
| Byte | 1-3 | 4-6 | 7-9 | 10-12 | 13-15 | 16-18 |

1. **Download T&A records** CMD：0x40

Function：download record, the downloading max number is 25 each time.（record data length: 25\*14 = 350Byte）

Command：（12 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x40 | 0x00 0x02 | 2Byte | CRCL CRCH |

Data format：

|  |  |  |
| --- | --- | --- |
| DATA | parameter | Record Amount |
| Byte | 1 | 2 |

Parameter define as below ：

*= 0*：*Normally downloading*

*= 1*：*Restart; retrieve all the records (The first data packet must send this data when*

*retrieving all the records)*

*= 2: Restart; retrieve new records (The first data packet must send this data when retrieving*

*the new records)*

*= 0x10*：*Send the last packet again*

Record amount *<=25*

Response：（12 + N \* 14Byte – *N is the valid records*）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xC0 | ACK\_SUCCESS  ACK\_FAIL | （1 + N \* 13） | （1 + N \* 14）Byte | CRCL CRCH |

Data format：（1 + N \* 14Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | *Valid records* N | *Record* 1 | *Record* 2 | … |
| Byte | 1 | 2-14 | 15-27 | … |

Record format ：（14Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| DATA | User code | Date&time | Backup code | Record type | Work types |
| Byte | 1-5 | 6-9 | 10 | 11 | 12-14 |

*Date&Time: how many seconds is it since the year 2000.*

*For instance, if record is made at 2012.12.31 24:00, then = (2012-1000)\*365\*24\*3600 Backup code: data 3—Card data2—Password data1—FP2 data 0—FP1*

*If Record Type bit 7(seventh bit) is 1,it means this record can open door;if 0,can’t open door;the low 4 bits is attendance state.*

1. **Upload T&A records** CMD：0x41

Function：Upload the T&A records, 1 record each time. Command：（24 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 041 | 0x00 0x0D | 14Byte | CRCL CRCH |

Data format：（14Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| DATA | User code | Date&time | Backup code | Record type | Work code |
| Byte | 1-5 | 6-9 | 10 | 11 | 12-14 |

*It counts the date and time from the year 2000. (It shows how many seconds is it from the year 2000.)*

Response：（11 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xC1 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Download staff info** CMD：0x42

Function：Download staff info, *<=*12 records each time (info data length: 12\*27= 324 Byte) Command：（12 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x42 | 0x00 0x02 | 2Byte | CRCL CRCH |

Data format：

|  |  |  |
| --- | --- | --- |
| DATA | Parameter | Info amount |
| Byte | 1 | 2 |

*Parameter defined as below:*

*= 0*：*Normally downloading*

*= 1*：*Restart the downloading*（*You must send this data when downloading the first data*

*packet*）

*= 0x10*：*Send the last packet again Info amount <=12*

Response：（12 + N \* 27 Byte -- *N is the valid records*）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xC2 | ACK\_SUCCESS  ACK\_FAIL | （1 + N \* 27） | （1 + N \* 27）Byte | CRCL CRCH |

Data format：（1 + N \* 27 Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | *Valid records* N | staff info 1 | staff info 2 | … |
| Byte | 1 | 2-28 | 29-55 | … |

staff info format：（27 Byte）

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DAT  A | User code | Number of  pwd+pwd | Card  code | Name | Departme  nt | Group  NO. | Attendance  mode | Registered  FP | Special  info |
| Byte | 1-5 | 6-8 | 9-11 | 12-21 | 22 | 23 | 24 | 25-26 | 27 |

*Number of pwd = Byte(6) >> 4*

*Registered FP Define : Byte (0 )=1 means already registered FP 1; Byte (1 )=1 means already registered FP 2. Special info Byte (7-6)*：*permission: 1-normal user , 3-administrator.*

*If all the byte (6-8) return 0xFF, it means the password does not exist. If all the byte (9-11) return 0xFF, it means the card ID doesn’t exist.*

1. **upload staff info** CMD：0x43

Function：download staff info, *<=*12 records each time（info data length：12\*27= 324 Byte） Command：（11+ N \* 27 Byte – *N is info amount*）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | Command | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x43 | 1 + N \* 27 | （1 + N \* 27）Byte | CRCL CRCH |

Data format：（1 + N \* 25 Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | *info amount* N | staff info 1 | staff info 2 | … |
| Byte | 1 | 2-28 | 29-55 | … |

*info amount <=12*

*If some item has no data, the setting of it is 0xFF.*

*The Registered FP item can not be set, it is 0 constantly.*

Response：（11 Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xC3 | ACK\_SUCCESS  ACK\_FAIL | 0x00  0x02 | 2Byte | CRCL CRCH |

Data content: 2 byte data , bits 15-0, low 12 bits indicate whether 1-12 employee upload successfully or not (1-succesful, 0-fail). For instance, 0000000010101110 means the second, third, Fourth, sixth, eighth user upload successfully, others failed.

1. **Download FP Template** CMD：0x44

Function：Download FP Template from T&A device Command：（16 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x44 | 0x00 0x06 | 6Byte | CRCL CRCH |

Data format：（6 Byte）

|  |  |  |
| --- | --- | --- |
| DATA | User code | Backup code |
| Byte | 1-5 | 6 |

*Backup code: 1- FP1, 2 –FP2.*

Response：（349 Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xC4 | ACK\_SUCCESS ACK\_FAIL  ACK\_NO\_USER | 0x01 0x52 | 338Byte | CRCL CRCH |

Data format：（338Byte）

|  |  |
| --- | --- |
| DATA | Fingerprint  template feature |
| Byte | 338 |

Device belongs to iris,Response：（1291Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xC4 | ACK\_SUCCESS  ACK\_FAIL ACK\_NO\_USER | 0x05 0x00 | 1280Byte | CRCL CRCH |

Data format：（1280Byte）

|  |  |
| --- | --- |
| DATA | 特征值 |
| Byte | 1280 |

1. **Upload FP Template** CMD：0x45 Function：Upload fingerprint template to the T&A device Command：（354 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x45 | 0x01 0x58 | 344Byte | CRCL CRCH |

Data format：（344 Byte）

|  |  |  |  |
| --- | --- | --- | --- |
| DATA | User code | Backup code | eigenvalue |
| Byte | 1-5 | 6 | 7-344 |

Device belongs to iris,Command：（1296Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x45 | 0x05 0x06 | 1286Byte | CRCL CRCH |

Data format：（1286Byte）

|  |  |  |  |
| --- | --- | --- | --- |
| DATA | 用户号 | 备份  号 | 特征值 |
| Byte | 1-5 | 6 | 7-1286 |

Response：（11 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xC5 | ACK\_SUCCESS  ACK\_FAIL ACK\_NO\_USER | 0x00 0x00 | CRCL CRCH |

1. **Get device S/N** CMD：0x46 Function：Get device ID which we set in device. Command：（10 Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x46 | 0x00 0x00 | CRCL CRCH |

Response：（15 Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xC6 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x04 | 4Byte | CRCL CRCH |

Data format：（4 Byte）

|  |  |
| --- | --- |
| DATA | Device ID |
| Byte | 1-4 |

1. **Modify device S/N** CMD：0x47 Function：Modify device ID in device menu Command：（14 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x47 | 0x00 0x04 | 4Byte | CRCL CRCH |

Data format：（4Byte）

|  |  |
| --- | --- |
| DATA | Device ID |
| Byte | 1-4 |

Response：（11 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xC7 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Get device type code** CMD：0x48 Function：Read device type code info Command：（10 Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x48 | 0x00 0x00 | CRCH CRCL |

Response：（19 Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xC8 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x08 | 8 Byte | CRCL CRCH |

Data format：（8 Byte）

|  |  |
| --- | --- |
| DATA | type code |
| Byte | 1-8 |

For instance (HEX)：02 00 00 00 01 C8 00 00 05 “TC400”000 CRCL CRCH

1. **Modify device type code** CMD：0x49 Function：Modify device type code info Command：（18 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x49 | 0x00 0x05 | 8 Byte | CRCL CRCH |

Data format：（8 Byte）

|  |  |
| --- | --- |
| DATA | type code |
| Byte | 1-8 |

Response：（11 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xC9 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Get the factory info code** CMD：0x4A Function：Read the device type info.

A）ANSI version Command：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x4A | 0x00 0x00 | CRCL CRCH |

Response：（21 Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xCA | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x0A | 10Byte | CRCL CRCH |

Data format：（10Byte）

|  |  |
| --- | --- |
| DATA | Type code |
| Byte | 1-10 |

B) UNICDE version Command: (10 byte) Same as ANSI version Response (31 byte)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xCA | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x14 | 20Byte | CRCL CRCH |

Data form：（20Byte）

|  |  |
| --- | --- |
| DATA | Message code |
| Byte | 1-20 |

1. **Modify the factory info code** CMD：0x4B

Function：Modify the device type info.

* 1. ANSI version Command：（20 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x4B | 0x00 0x0A | 10Byte | CRCL CRCH |

Data format：（10Byte）

|  |  |
| --- | --- |
| DATA | Type code |
| Byte | 1-10 |

Response：（11 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xCB | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

* 1. UNICODE Version Command: (30 byte)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x4B | 0x00 0x14 | 20Byte | CRCL CRCH |

Data form (20 byte)

|  |  |
| --- | --- |
| DATA | Message code |
| Byte | 1-20 |

Response: (11Byte) Same as ANSI version

# Delete the designated user data CMD：0x4C

Function：Delete all the data of designated user. Command：（16 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x4C | 0x00 0x06 | 6Byte | CRCL CRCH |

Data format：（6 Byte）

|  |  |  |
| --- | --- | --- |
| DATA | user code | Backup code |
| Byte | 1-5 | 6 |

*Backup code define : Byte(3) - card , Byte(2) – password , Byte(1) – FP2 , Byte(0) – FP1 . (can select the function, it does not cancel the staff info)*

*Backup code= 0xFF cancel all the data of the user (including the staff info)*

Response：（11 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xCC | ACK\_SUCCESS  ACK\_NO\_USER | 0x00 0x00 | CRCL CRCH |

1. **Initialize the user area** CMD：0x4D

Function：Initialize all the user data area, clear all the staff info, FP data, password/card data Command：（10 Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x4D | 0x00 0x00 | CRCL CRCH |

Response：（11 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xCD | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Clear up Records /Clear new records sign** CMD：0x4E Function：Cancel all records, or cancel all/part new records sign. Command：（13 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x4E | 0x00 0x04 | 4Byte | CRCL CRCH |

Data format：（3 Byte）

|  |  |  |
| --- | --- | --- |
| DATA | clear type | New record amount |
| Byte | 1 | 2-4 |

*Clear type definition: 0 - Clear up Records.*

1. *- Clear all the new Records sign.*
2. *- Clear the designated amount new records sign, new record amount is decided by Byte(2-4).*

Response：（11 Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xCE | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x03 | 3Byte | CRCL CRCH |

Data format：（3 Byte）

|  |  |
| --- | --- |
| DATA | Delete records/clear new record amount |
| Byte | 1-3 |

*If the delete type is 0, return the amount of cancelling all records;*

*If the delete type is 1, return the amount of cancelling all the new records; If the delete type is 2, return the amount of cancelling new records.*

1. **Initialize System** CMD：0x4F

Function：Initialize the device system to recover the factory settings , but the language/

date display format /communication setting /SN /factory info code /device type code is not changed.

Command：（10 Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x4F | 0x00 0x00 | CRCL CRCH |

Response：（11 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xCF | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Get the time zone info** CMD：0x50

Function：Read the time zone info. The total time zone amount is 30. Command：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x50 | 0x00 0x01 | 1Byte | CRCL CRCH |

Data format：（1Byte）

|  |  |
| --- | --- |
| DATA | NO. |
| Byte | 1 |

Response：（39 Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xD0 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x1D | 28Byte | CRCL CRCH |

Data format：（28Byte）

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | Monday subsidiary time zone | Tuesday subsidiary time zone | Wednesday subsidiary time zone | Thursday subsidiary time zone | Friday subsidiary time zone | Saturday subsidiary time zone | Sunday subsidiary time zone |
| Byte | 1-4 | 5-8 | 9-12 | 13-16 | 17-20 | 21-24 | 25-28 |

Subsidiary time zone format：（4Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | Begin hour | Begin minute | End hour | End minute |
| Byte | 1 | 2 | 3 | 4 |

1. **Set time zone info** CMD：0x51

Function：Set time zone info, the total time zone amount is 32. Command：（39 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x51 | 0x00 0x00 | 29Byte | CRCL CRCH |

Data format：（29Byte）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | NO. | Monday subsidiary  time zone | Tuesday  subsidiary time zone | Wednesday  subsidiary time zone | Thursday  subsidiary time zone | Friday  subsidiary time zone | Saturday  subsidiary time zone | Sunday  subsidiary time zone |
| Byte | 1 | 2-5 | 6-9 | 10-13 | 14-17 | 18-21 | 22-25 | 26-29 |

Response：（11 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xD1 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Get the group info** CMD：0x52

Function：Read some group info. Group NO. is 0-16 and Group 0/1 is the fixed normal close/ open group. We can just read group 2-16 info.

Command：（11 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x52 | 0x00 0x01 | 1Byte | CRCL CRCH |

Data format：（1Byte）

|  |  |
| --- | --- |
| DATA | Group NO. |
| Byte | 1 |

Response：（15 Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xD2 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x04 | 4Byte | CRCL CRCH |

Data format：（4Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | time zone 1 NO. | time zone 2 NO. | time zone 3 NO. | time zone 4 NO. |
| Byte | 1 | 2 | 3 | 4 |

1. **Set the group info** CMD：0x53

Function：Set some group info. Group NO. is 0-16 and Group 0/1 is the fixed normal close/ open group. We can just set group 2-16 info.

Command：（15 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x53 | 0x00 0x05 | 5 Byte | CRCL CRCH |

Data format：（5 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| DATA | Group NO. | time zone 1 NO. | time zone 2 NO. | time zone 3 NO. | time zone 4 NO. |
| Byte | 1 | 1 | 2 | 3 | 4 |

Response：（11 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xD3 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Get the scheduled bell info** CMD：0x54 Function：Read the scheduled ring time, the total amount is 30. Command：（10 Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x54 | 0x00 0x00 | CRCL CRCH |

Response：（101 Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xD4 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x3C | 90Byte | CRCL CRCH |

Data format：（90Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | Time point 1 | Time point 2 | … | Time point 30 |
| Byte | 1-3 | 4-6 | … | 88-90 |

Time format：（2Byte）

|  |  |  |  |
| --- | --- | --- | --- |
| DATA | Hour | Minute | Weekday |
| Byte | 1 | 2 | 3 |

*For instance, if weekday=00111110, means from Monday to Friday the bell would ring at specified time. Bits 6-1 stand for Saturday to Monday, 1 means ring, 0 means not ring.*

1. **Set ring info** CMD：0x55

Function：Set bell schedule Command：（14 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x55 | 0x00 0x04 | 4Byte | CRCL CRCH |

Data format：（4Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | NO. | Hour | Minute | Weekday |
| Byte | 1 | 2 | 3 | 4 |

Response：（11 Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xD5 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Retrieve specified short message** CMD: 0x56

Function: Retrieve the start date, end date and content of specified short message. There are 50 Short message at most, index 0-49, message data is 48 bytes.

* 1. ANSI version Command: (11Byte)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x56 | 0x00 0x01 | 1Byte | CRCL CRCH |

Data: short message index, 1 byte.

|  |  |
| --- | --- |
| DATA | Message index |
| Byte | 1 |

Response: (70Bytes)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xD6 | ACK\_SUCCESS  ACK\_FAIL ACK\_NO\_USER | 0x00 0x3B | 59 Byte | CRCL CRCH |

Data format: (59Bytes)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | User id | Start date | | | End date | | | Message  content |
| year | month | day | year | month | day |
| Byte | 1-5 | 6 | 7 | 8 | 9 | 10 | 11 | 12-59 |

* 1. UNICODE version Command: (11bytes) Same as ANSI version Response: (118Bytes)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xD6 | ACK\_SUCCESS  ACK\_FAIL ACK\_NO\_USER | 0x00 0x6B | 107 Byte | CRCL CRCH |

Data format: (107Bytes)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | User id | Start date | | | End date | | | Message  content |
| year | month | day | year | month | day |
| Byte | 1-5 | 6 | 7 | 8 | 9 | 10 | 11 | 12-107 |

1. **Add short message** CMD：0x57 Function: add one short message
   1. ANSI version Command：（69Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x57 | 0x00 0x3B | 59Byte | CRCL CRCH |

Data format：（59Byte）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | User id | Start date | | | End date | | | Message  content |
| year | month | day | year | month | day |
| Byte | 1-5 | 6 | 7 | 8 | 9 | 10 | 11 | 12-59 |

Response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xD7 | ACK\_SUCCESS  ACK\_FULL | 0x00 0x00 | CRCL CRCH |

* 1. UNICODE version Command：（117Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x57 | 0x00 0x6B | 107Byte | CRCL CRCH |

Data format：（107Byte）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | User id | Start date | | | End date | | | Message  content |
| year | month | day | year | month | day |
| Byte | 1-5 | 6 | 7 | 8 | 9 | 10 | 11 | 12-107 |

Response：（11Byte） Same as ANSI version

1. **Read all info head of all short message** CMD：0x58 Function: read all info head of all short message

Command：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x58 | 0x00 0x00 | CRCL CRCH |

Response：（561Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xD8 | ACK\_SUCCESS  ACK\_FAIL | 0x02 0x26 | 550Byte | CRCL CRCH |

Data format：（550Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | Info head of  message 0 | Info head of  message 1 |  | Info head of message 49 |
| Byte | 1-11 | 12-22 |  | 540-550 |

Info head format：（11Byte）

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | User id | Start date | | | End date | | |
| year | month | day | year | month | day |
| Byte | 1-5 | 6 | 7 | 8 | 9 | 10 | 11 |

*If short message doesn’t exist, all 11 bytes set as 0xFF*

1. **Delete specified index short message** CMD：0x59 Function: delete specified index short message Command：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x59 | 0x00 0x01 | 1Byte | CRCL CRCH |

Data format：（1Byte）

|  |  |
| --- | --- |
| DATA | Message index |
| Byte | 1 |

Response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xD9 | ACK\_SUCCESS  ACK\_FAIL ACK\_EMPTY | 0x00 0x00 | CRCL CRCH |

If index is 0xFF, delete all short messages.

1. **Get T&A state message** CMD：0x5A Function：Get T&A State message

Response：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x5A | 0x00 0x00 | CRCL CRCH |

Response：（27Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xDA | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x10 | 16Byte | CRCL CRCH |

Data format：（16Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | T&A State 0 | T&A state1 | ..… | T&A State 15 |
| Byte | 1 | 2 | …. | 16 |

*If actual T&A state less than 16, empty state byte set as 0xFF Default T&A state*（*index range 0-254*）：

*Index 0*：*IN*

*Index 1*：*OUT Index 2*：*BREAK*

1. **Set T&A State parameter table** CMD：0x5B Function：Set T&A State message

Command：（26Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x5B | 0x00 0x10 | 16Byte | CRCL CRCH |

Data format：（16Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | T&A State 0 | T&A State 1 | 。。。 | T&A State 15 |
| Byte | 1 | 2 | 。。。 | 16 |

*If actual T&A state less than 16, empty state byte set as 0xFF*

Response：（11Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xDB | | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Enroll user FP online** CMD：0x5C Function：enroll user FP online，verify double times Command：（17Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x5C | 0x00 0x07 | 7Byte | CRCL CRCH |

Data format：（7Byte）

|  |  |  |  |
| --- | --- | --- | --- |
| DATA | ID | Backup  ID | Enroll  times |

|  |  |  |  |
| --- | --- | --- | --- |
| Byte | 1-5 | 6 | 7 |

*Enroll times define as below*：*0-first 1-second*

Response：（12Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xDC | ACK\_SUCCESS ACK\_FAIL ACK\_TIME\_OUT  ACK\_FULL(only when enroll time==1) ST\_USER\_OCCUPIED(only when enroll time==1)  ST\_FINGER\_OCCUPIED(only when enroll  time==1) | 0x00 0x00 | CRCL CRCH |

1. **Get device capacity parameter** CMD：0x5D

Function：get device capacity parameter，including employee amount，fingerprints amount，support record amount

Command：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x5D | 0x00 0x00 | CRCL CRCH |

Response：（20Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xDD | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x09 | 9Byte | CRCL CRCH |

Data format：（9Byte）

|  |  |  |  |
| --- | --- | --- | --- |
| DATA | Support employee  amount | Support fingerprint  amount | Support record  amount |
| Byte | 1-3 | 4-6 | 7-9 |

1. **Output signal to open lock without verifying user** CMD：0x5E Function：Force T&A device output signal to open door1

Command：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x5E | 0x00 0x00 | CRCL CRCH |

Response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xDE | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Sent T&A record in real time** CMD：0x5F

Function：send T&A records after verify OK, only response message：（25Byte） Response: (25bytes)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xDF | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x0E | 14Byte | CRCL CRCH |

Data format：（14Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| DATA | User ID | *Date&Time* | Backup ID | Record type | Work code |
| Byte | 1-5 | 6-9 | 10 | 11 | 12-14 |

*Date&Time: how many seconds is it since the year 2000.*

*For instance, if record is made at 2012.12.31 24:00, then = (2012-1000)\*365\*24\*3600*

1. **Get customized T&A state table** CMD：0x70 Function：Read customize attendance state message
   1. ANSI version Command：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x70 | 0x00 0x00 | CRCL CRCH |

response：（172Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xF0 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0xA1 | 161Byte | CRCL CRCH |

Data format：（161Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| DATA | T&A state NUM | T&A state 0 char | T&A state 1 char | 。。。 | T&A state 15 char |
| Byte | 1 | 2-11 | 12-21 | 。。。 | 152-161 |

Attendance state largest Number is 16

* 1. UNICODE Version Command：（10Byte）

Same as ANSI version Response：（332Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xF0 | ACK\_SUCCESS  ACK\_FAIL | 0x01 0x41 | 321Byte | CRCL CRCH |

Data format：（321Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| DATA | attendance  state NUM | attendance state 0  char | attendance state 1  char | 。。。 | attendance state 15  char |
| Byte | 1 | 2-21 | 22-41 | 。。。 | 302-321 |

Attendance state largest Number is 16

1. **Set attendance state table** CMD：0x71 Function：Set customized attendance message
   1. ANSI Version Command：（171Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x71 | 0x00 0xA1 | 161Byte | CRCL CRCH |

Data format：（161Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| DATA | attendance  state NUM | attendance state 0  char | attendance state 1  char | 。。。 | attendance state 15  char |
| Byte | 1 | 2-11 | 12-21 | 。。。 | 152-161 |

Attendance state largest Number is 16

*although the largest string length is 10, ,because vendor code(as 0x4A command)and attendance state display on LCD are in same row*，*(string length + vendor code length) should <= 15*。*For*

*example, if vendor code length is 10,and every attendance state string length should <=5*

response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xF1 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

*Customized attendance state is one of attendance state mode, another is supplied by 0x5B command ,make following rules in order to distinct*：*default state is 0x5B*，*when 0x5B/0x71 is sent attendance device will be switch to0x5B/0x71 mode and keeping this state*

* 1. UNICODE version command：（331Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x71 | 0x01 0x41 | 321Byte | CRCL CRCH |

Date format：（321Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| DATA | attendance  state NUM | attendance state 0  char | attendance state 1  char | 。。。 | attendance state 15  char |
| Byte | 1 | 2-21 | 22-41 | 。。。 | 302-321 |

Response：（11Byte） Same as ANSI version

# Download employees data (extended) CMD：0x72

Function：download staff information，12 records at most at one time （data length：12\*30= 360Byte）

* 1. ANSI Version Command：（12Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x72 | 0x00 0x02 | 2Byte | CRCL CRCH |

Data format：

|  |  |  |
| --- | --- | --- |
| DATA | parameter | Data  amount |
| Byte | 1 | 2 |

*Parameter defined as below*：

*= 0*：*downloading*

*= 1*：*start downloading*（*must send this to receive first pack*）

*= 0x10*：*resend previous information Information amount<=12*

Response：（12 + N \* 30Byte – *N is valid message amount*）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xF2 | ACK\_SUCCESS  ACK\_FAIL | （1 + N \* 30） | （1 + N \* 30）Byte | CRCL CRCH |

Data format：（1 + N \* 30Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | Valid message numbers  N | Staff information 1 | Staff 2 | … |
| Byte | 1 | 2-31 | 32-61 | … |

Staff information format：（30Byte）

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | User ID | PWD NUM+PWD | CARD ID | NAME | Department | Group | Attendance mode | FP  enroll state | PWD  8 digit | keep | Speci  al Info |
| Byte | 1-5 | 6-8 | 9-12 | 13-22 | 23 | 24 | 25 | 26-27 | 28 | 29 | 30 |

*Password length = Byte(6) >> 4*

*The low 20bits of password is saved in Byte 6-8, high 8 bits saved in Byte28 FP enroll state define*：*digit 0 = 1 FP1 enrolled*，*digit 1 = 1 FP 2 enrolled Special message digit 7-6*：*Authority 1-normal user 3-admin*

*Digit 4*：*Length of card id 1 – 32 digit 0 – 24digit*

*If byte 6-8 return0xFF means password not exist If byte 9-12 return0xFF means card ID not exist*

* 1. UNICODE version

Could download 8 records at most each time(ANSI version is 12）data length ：8\*40= 320Byte Command ：（12Byte）

Same as ANSI version

Response：（12 + N \* 40Byte – *N is valid message numbers*）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xF2 | ACK\_SUCCESS  ACK\_FAIL | （1 + N \* 40） | （1 + N \* 40）Byte | CRCL CRCH |

Data format：（1 + N \* 40Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | Valid data item as N | Staff message 1 | Staff message 2 | … |
| Byte | 1 | 2-41 | 42-81 | … |

Staff information format：（30Byte）

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | User ID | PWD length+ PWD | card | name | dpt | group | Attendance mode | Enroll FP state | Pwd  high 8 digit | keep | Special message |
| Byte | 1-5 | 6-8 | 9-12 | 13-32 | 33 | 34 | 35 | 36-37 | 38 | 39 | 40 |

# Upload staff information(extended) CMD：0x73

Function：upload staff information, 12 records at most each time（data length：12\*30= 360Byte）

* 1. ANSI version

Command：（11 + N \* 30Byte – *N is data amount*）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | command | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x73 | 1 + N \* 30 | （1 + N \* 30）Byte | CRCL CRCH |

data：（1 + N \* 30Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | Information Num N | Staff information 1 | Staff information 2 | … |
| Byte | 1 | 2-31 | 32-61 | … |

*Data amount<=12*

*If user data is empty, set it as 0xFF. For instance, card Id set as 0xFF if user don’t enroll card. FP enroll state can not set, this value is 0*

Response：（13Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xF3 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x02 | 2Byte | CRCL CRCH |

Data format：（2Byte）

|  |  |
| --- | --- |
| DATA | flag |
| Byte | 2 |

*Flag bit 0-11*：*NO.1-12 staff enroll successfully or not*（*1*：*successful*；*0*：*fail*）

* 1. UNICODE Version

Upload 8 user date at most each time（ANSI version 12）data length：8\*40= 320Byte Command：（11 + N \* 40Byte – *N is data amount*）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | Command | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x73 | 1 + N \* 40 | （1 + N \* 40）Byte | CRCL CRCH |

Data format：（1 + N \* 40Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | Information numbers N | Staff information 1 | Staff information 2 | … |
| Byte | 1 | 2-41 | 42-81 | … |

Response：（13Byte） Same as ANSI version

1. **Get communication device ID** CMD：0x74 Function：Read communicate device id Command：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x74 | 0x00 0x00 | CRCL CRCH |

Response：（15Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xF4 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x04 | 4Byte | CRCL CRCH |

Data format：（4Byte）

|  |  |
| --- | --- |
| DATA | Device ID |
| Byte | 1-4 |

1. **Modify communication device ID** CMD：0x75 Function：Modify communication device ID Command：（14Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x75 | 0x00 0x04 | 4Byte | CRCL CRCH |

Data format：（4Byte）

|  |  |
| --- | --- |
| DATA | Device ID |
| Byte | 1-4 |

Response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xF5 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Clear administrator flag** CMD：0x3D Function：Clear all administrator flag Command：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x3D | 0x00 0x00 | CRCL CRCH |

Response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xBD | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

# Read employees enrollment timestamp CMD：0x3E

Function：Read specified staff enrollment timestamp, timestamp= how many seconds elapse since 2000-01-01 00:00

Command ：（15Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | command | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x3E | 0x00 0x04 | 5Byte | CRCL CRCH |

Data format：（4Byte）

|  |  |
| --- | --- |
| DATA | User ID |
| Byte | 1-5 |

Response：（15Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xBE | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x04 | 4Byte | CRCL CRCH |

Data format：（4Byte）

|  |  |
| --- | --- |
| DATA | Time stamp(seconds) |
| Byte | 1-4 |

1. **Set time stamp** CMD：0x3F

Function：Set specified staff enrollment timestamp, timestamp= how many seconds elapse since 2000-01-01 00:00

Command：（14Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | command | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x3F | 0x00 0x04 | 4Byte | CRCL CRCH |

Data format：（4Byte）

|  |  |
| --- | --- |
| DATA | timestamp(seconds) |
| Byte | 1-4 |

Response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xBF | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Read random number** CMD：0x76 Function：Read random number Command：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | Command | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x76 | 0x00 0x00 | CRCL CRCH |

Response：（15Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xF6 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x04 | 4Byte | CRCL CRCH |

Data format：（4Byte）

|  |  |
| --- | --- |
| DATA | random |

|  |  |
| --- | --- |
| Byte | 1-4 |

1. **Encrypt device type and language with random number**CMD：0x77 Function：Encrypt device type and language with random number generated by command 0x76 Command：（19Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | Command | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x77 | 0x00 0x09 | 9Byte | CRCL CRCH |

Data form：（4Byte）

|  |  |  |
| --- | --- | --- |
| DATA | Encrypt model | Encrypt  language |
| Byte | 1-8 | 9 |

Response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xF7 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Get specified index message**CMD：0x26 only for OA3000

Function：Read the start date and time, end date and time, content of specified index message. 200 message in total, index 0-199，message content is 450 byte in total.

Command：（12Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x26 | 0x00 0x02 | 2Byte | CRCL CRCH |

Data form：（2Byte）

|  |  |
| --- | --- |
| DATA | index |
| Byte | 2 |

Response：（472Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xA6 | ACK\_SUCCESS ACK\_FAIL  ACK\_NO\_USER | 0x01 0xCD | 461 Byte | CRCL CRCH |

Data format：（461Byte）

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | User ID | Start date | | | End date | | | Message title | Message content |
| Y | M | D | Y | M | date |
| Byte | 1-5 | 6 | 7 | 8 | 9 | 10 | 11 | 12-61 Byte | 62-461 Byte |

1. **Add new message** CMD：0x27 only for OA3000 Function：Add a new message

Command：（471Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x27 | 0x01 0xCD | 461Byte | CRCL CRCH |

Data form：（461Byte）

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | User ID | Start date | | | End date | | | Message title | Message content |
| Y | M | D | Y | M | D |
| Byte | 1-5 | 6 | 7 | 8 | 9 | 10 | 11 | 12-61 byte | 62-461byte |

*User ID is 0 means it’s a public message*

Response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xA7 | ACK\_SUCCESS  ACK\_FULL | 0x00 0x00 | CRCL CRCH |

1. **Read message head of assigned section message** CMD：0x28 only for OA3000 Function：Read message head of all short message

Command：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x28 | 0x00 0x01 | 1Byte | CRCL CRCH |

Data format：（1Byte）

|  |  |
| --- | --- |
| DATA | Section number (0-3) |
| Byte | 1 |

Data format：（561Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xA8 | ACK\_SUCCESS  ACK\_FAIL | 0x02 0x26 | 550Byte | CRCL CRCH |

Data format：（550Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | Message 50\*section number message  head | Message 50\*section number +1message head |  | Message 50\*section number +49message head |
| Byte | 1-11 | 12-22 |  | 540-550 |

Message head format：（11Byte）

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | User ID | Start date | | | End date | | |
| Y | M | D | Y | M | date |
| Byte | 1-5 | 6 | 7 | 8 | 9 | 10 | 11 |

*If this index message does not exist, 11 bytes all set as 0xFF*

1. **Delete appointed index message** CMD：0x29 only for OA3000 Function：Delete appointed index message content。

Command：（12Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x29 | 0x00 0x02 | 2Byte | CRCL CRCH |

Data format：（2Byte）

|  |  |
| --- | --- |
| DATA | index |
| Byte | 2 |

*If index is 0xFFFF delete all information*

Response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xA9 | ACK\_SUCCESS ACK\_FAIL  ACK\_EMPTY | 0x00 0x00 | CRCL CRCH |

1. **Get T&A state auto switch setting** CMD：0x20 only for OA3000/OA1000

Function：read T&A state auto switch setting，T&A state amount is 16 Command：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x20 | 0x00 0x01 | 1Byte | CRCL CRCH |

Data format：（1Byte）

|  |  |
| --- | --- |
| DATA | State index |
| Byte | 1 |

Response：（40Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xA0 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x1D | 29Byte | CRCL CRCH |

Data format：（29Byte）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | Monday  Sub-period | Tuesday  Sub-period | Wednesday  Sub-period | Thursday  Sub-period | Friday  Sub-period | Saturday  Sub-period | Sunday  Sub-period | State  Number |
| Byte | 1-4 | 5-8 | 9-12 | 13-16 | 17-20 | 21-24 | 25-28 | 29 |

Sub-period format：（4Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | Start hour | Start min | End hour | End Min |
| Byte | 1 | 2 | 3 | 4 |

1. **Set T&A state auto switch setting** CMD：0x21 only for OA3000/OA1000 Function：Set T&A state auto switch setting, 16 T&A state in total.

Command：（40Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x21 | 0x00 0x1E | 30Byte | CRCL CRCH |

Data form：（30Byte）

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | Series NO | Monday Sub-period | Tuesday Sub-period | Wednesda y  Sub-period | Thursday Sub-period | Friday Sub-period | Saturday Sub-period | Sunday Sub-period | State Number |
| Byte | 1 | 2-5 | 6-9 | 10-13 | 14-17 | 18-21 | 22-25 | 26-29 | 30 |

Response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xA1 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Download staff information (extended)** CMD：0x22 761 platform use only Function ：download staff information, download 6 staff information at most each time（data length：6\*84= 504Byte）

Command ：（12Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x22 | 0x00 0x02 | 2Byte | CRCL CRCH |

Data format：

|  |  |  |
| --- | --- | --- |
| DATA | parameter | Data items |
| Byte | 1 | 2 |

*Parameter items define as follow*：

*= 0*：*downloading*

*= 1*：*start downloading*（*must send this message to receive first package*）

*= 0x10*：*resend last package Data amount <=12*

Response:（12 + N \* 84Byte – *N is valid data numbers*）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xA2 | ACK\_SUCCESS  ACK\_FAIL | （1 + N \* 84） | （1 + N \* 84）Byte | CRCL CRCH |

Data format：（1 + N \* 84Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | Valid data N | Staff information 1 | Staff information 2 | … |
| Byte | 1 | 2-85 | 86-169 | … |

Staff information format：（84Byte）

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | user | Pwd  numbers+pwd | Card | name | dpt | group | Attendance  mode | FP enroll  state | keep | keep | Special  information |
| Byte | 1-5 | 6-8 | 9-12 | 13-76 | 77 | 78 | 79 | 80-81 | 82 | 83 | 84 |

*Password digit = Byte(6) >> 4*

*FP enroll state define*：*digit 0 = 1 means enrolled 1*，*digit 1 = 1 means enrolled 2 Special information digit 7-6*：*authority 1-normal user 3-administrator*

*Digit 4*：*length of card number 1 – 32bit 0 – 24 bit If byte 6-8 return 0xFF means password not exist*

*If byte 9-12 return 0xFF means card not exist*

1. **Upload staff information (extend)** CMD：0x23 761plate use only function：Upload staff information，upload 6 is maximum（data length：6\*84= 504Byte） command ：（11 + N \* 84Byte – *N is data numbers*）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | command | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x23 | 1 + N \* 84 | （1 + N \* 84）Byte | CRCL CRCH |

Data form：（1 + N \* 84Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | Information numbers N | Staff information1 | Staff information 2 | … |
| Byte | 1 | 2-85 | 86-169 | … |

*Information numbers<=6*

*If no data this value is 0xFF*

*FP enroll state can not set, this value is 0*

Response ：（13Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xA3 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x02 | 2Byte | CRCL CRCH |

Data format：（2Byte）

|  |  |
| --- | --- |
| DATA | mark |
| Byte | 2 |

*Mark digit 0-5*：*NO.1-6 staff upload successful or not*（*1*：*successful*；*0*：*fail*）

1. **Get device serial number** CMD：0x24 Function：Get device serial number command：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x24 | 0x00 0x00 | CRCL CRCH |

Response ：（27Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xA4 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x10 | 16Byte | CRCL CRCH |

Data format：（16Byte）

|  |  |
| --- | --- |
| DATA | Serial number |
| Byte | 1-16 |

1. **Modify device serial number** CMD：0x25 Function ：Modify device serial number

Command ：（26Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x25 | 0x00 0x10 | 16Byte | CRCL CRCH |

Data form：（16Byte）

|  |  |
| --- | --- |
| DATA | Serial number |
| Byte | 1-16 |

Response ：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xA5 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Get special state** CMD：0x2F VF30/VP30/T60+use only Function：Get special state in current

Command ：（12Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x2F | 0x00 0x00 | CRCL CRCH |

Response ：（19Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xAF | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x08 | 8Byte | CRCL CRCH |

Data format：（8Byte）

|  |  |  |
| --- | --- | --- |
| DATA | state | keep |
| Byte | 1 | 2-8 |

*State defined as below:*

*digit 1*：*door sensor state 0-normal 1-warning*

1. **Get photo amount** CMD：0x2A OA1000/OA3000/761platform use only Function：Get photo amount

Command ：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x2A | 0x00 0x00 | CRCL CRCH |

Response ：（14Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xAA | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x03 | 3 Byte | CRCL CRCH |

Data form ：（3Byte）

|  |  |
| --- | --- |
| DATA | Photo amount |
| Byte | 1-3 |

1. **Get photo head information** CMD：0x2B OA1000/OA3000/761 platform only use Function:Get photo head information，the maximum is 50 file head information in every times Command:（12Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x2B | 0x00 0x02 | 2Byte | CRCL CRCH |

Data format:

|  |  |  |
| --- | --- | --- |
| DATA | Parameter | Information  numbers |
| Byte | 1 | 2 |

*Parameter item define as follow*：

*= 0*：*downloading*

*= 1*：*start downloading*

*= 0x10*：*resend last package Information amount<=50*

Response:（12 + N \* 9Byte – *N is valid information* ）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xAB | ACK\_SUCCESS  ACK\_FAIL | （1 + N \* 9） | （1 + N \* 9）Byte | CRCL CRCH |

Data format:（1 + N \* 9Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | Valid information N | Photo file head 1 | Photo file head 2 | … |
| Byte | 1 | 2-10 | 11-19 | … |

Photo file head form：（9Byte）

|  |  |  |
| --- | --- | --- |
| DATA | user | Date time |
| Byte | 1-5 | 6-9 |

Date time = how many seconds elapse since 2000-01-01 00:00

1. **Read specified photo file** CMD：0x2C OA1000/OA3000/761platform use only Function：Read specified photo file

command：（20Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x2C | 0x00 0x0A | 10Byte | CRCL CRCH |

Data format：（10Byte）

|  |  |  |
| --- | --- | --- |
| DATA | Parameter | photo head |
| Byte | 1 | 9 |

*Parameter item define as follow*：

*= 0*：*downloading*

*= 1*：*start downloading*

*= 0x10*：*resend last package Information number<=50*

Photo file head form：（9Byte）

|  |  |  |
| --- | --- | --- |
| DATA | User | Date time |
| Byte | 1-5 | 6-9 |

Date time = how many seconds elapse since 2000-01-01 00:00

Response ：（12+NByte N is real capacity send file package N<=512）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xAC | ACK\_SUCCESS  ACK\_FAIL | (1+N) | (1+N)Byte | CRCL CRCH |

Data format：（1+NByte N<=512）

|  |  |  |
| --- | --- | --- |
| DATA | Parameter | File content |
| Byte | 1 | N |

*Parameter define as follow*：

*= 0*：*downloading*

*= 1*：*download done*

1. **Delete specified photo** CMD：0x2D OA1000/OA3000/761plat form use only Function ：Delete specified photo information

Command ：（19Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x2D | 0x00 0x09 | 9Byte | CRCL CRCH |

Data format ：（9Byte）

|  |  |
| --- | --- |
| DATA | Photo head file |
| Byte | 9 |

Photo head file format：（9Byte）

|  |  |  |
| --- | --- | --- |
| DATA | User | Date time |
| Byte | 1-5 | 6-9 |

Data time = how many seconds elapse since 2000-01-01 00:00

*If photo file head is 0xFF delete all*

Response ：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xAD | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Update firmware, photo, voice** CMD：0x10 761platform use only

Function ：Update firmware ,photo ,voice，must be upload 512 byte every times except package end

command ：（Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x10 | 16+len | 16+len Byte | CRCL CRCH |

Data format：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| DATA | Parameter | Type | Index | firmware ,photo ,voice,  name | Actual  byte |
| Byte | 1 | 1 | 2 | 12 | len |

*Parameter define as following* ：

*= 0*：*uploading*

*= 1*：*start uploading*

*= 2*：*end uploading Type defined as below:*

*= 0 firmware*，*= 1photo* ，*= 2 voice* ，*= 3 language files Index item define as follow*：

*Start from 0, increase by 1 each time Firmware type define as below*：

*= 0*：*firmware*，*= 1 booter, = 2 character library*

Response ：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| DLE STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x90 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **directory file operation**CMD：0x12 761 platforms

Function ：Retrieve file directory and file name, delete file, read file content Command ：（10+4+len Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x12 | 4+len | 4+len Byte | CRCL CRCH |

Data format：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | parameter | type | Index | File name/directory/file content  actual byte |
| Byte | 1 | 1 | 2 | len |

*Parameter define as following* ：

*= 0*：*upload normal*

*= 1*：*start upload*

*= 2*：*end upload Type define as below*：

*=0*：*get directory and file name from specified directory*（*must specify directory name*）

*=1*：*get specified file content*（*must specify file name*）

*=2*：*delete specified file*（*must specify file name*）

*=3*：*upload firmware*（*not specify file name*）

*=4*：*upload booter*（*not specify file name*）

*=5*：*upload character library* （*not specify file name*）

*=6*：*upload photo, voice , configuration file*（*must specify file name*） *Index item define as below*：

*0 increase form start*

Response ：（11+4+len Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| DLE STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x92 | ACK\_SUCCESS  ACK\_FAIL | 4+len | 4+len Byte | CRCL CRCH |

Data format：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | Parameter | type | index | File name/catalogue/file content  Real byte |
| Byte | 1 | 1 | 2 | len |

*Parameter define as following* ：

*= 0*：*upload normal*

*= 1*：*start upload*

*= 2*：*end upload Type define as below*：

*=0*：*get catalogue and file name from appoint catalogue*（*must be have catalogue name*）

*=1*：*get file name from appoint file*（*must be have file name*）

*=2*：*delete appoint file*（*must be have file name*）

*=3*：*upload firmware*（*not follow file name*）

*=4*：*upload boot*（*not follow file name*）

*=5*：*upload word store* （*not follow file name*）

*=6*：*upload photo*，*voice ,file*（*must be have file name*） *Index item define as below*：

*0 increase form start*

Notice ：1. The first pack doesn’t include any data, it just send transmission request or indicate It’s ready

2．directory marked by 0xFF，file marked by 0xFE，multiple directory or file name marked by 0x00

1. **Download log record** CMD：0x13 761 platform use only

Function ：download log record, 8 records at most each time(record data length ：8\*73 = 584Byte）

Command ：（12Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x40 | 0x00 0x02 | 2Byte | CRCL CRCH |

Data format ：

|  |  |  |
| --- | --- | --- |
| DATA | Parameter | Record  number |
| Byte | 1 | 2 |

*Parameter item define as follow:*

*= 0*：*downloading normal*

*= 1*：*download start* ，*all record*

*= 0x10*：*resend last data package Record items <=8*

Response ：（12 + N \* 73Byte – *N is valid record item*）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xC0 | ACK\_SUCCESS  FAIL | （1 + N \* 73） | （1 + N \* 73）Byte | CRCL CRCH |

Data form ：（1 + N \* 73Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | Valid record item N | Attendance record 1 | Attendance record 2 | … |
| Byte | 1 | 2-74 | 75-147 | … |

Log record form：（73Byte）

|  |  |  |  |
| --- | --- | --- | --- |
| DATA | User id | Date time | Record content |
| Byte | 1-5 | 6-9 | 10-73 |

Data time = how many seconds elapse since 2000-01-01 00:00

1. **Read admin card number/admin password** CMD：0x1C only for T5 Function ：Get T5A admin card number /T50 admin password

Command ：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x1C | 0x00 0x00 | CRCL CRCH |

Response ：（24Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x9C | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x0D | 13Byte | CRCL CRCH |

Data format: (13Byte)

1. if model is T5A，then

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | Add card ID | Delete card id | Keep | Special  information |
| Byte | 1-4 | 5-8 | 9-12 | 13 |

*Special information defined as below:*

*bit 0*：*Add card length 1 – 32 bit 0 - 24 bit*

*bit 1*：*Delete card length 1 - 32 bit 0 - 24 bit if device model is T5B*，*RET code return ACK\_FAIL*

1. *if device model is* T50，

|  |  |  |
| --- | --- | --- |
| DATA | Manage pwd length+  Manage length | keep |
| Byte | 1-3 | 4-13 |

Manage password length = Byte(1) >> 4

1. **Set admin card number/admin password** CMD：0x1D only for T5 Function：Set T50 admin card number/admin password

Command: (23Byte)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x1D | 0x00 0x0D | 13Byte | CRCL CRCH |

Data format：（13Byte）

1. If device model is T5A，

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | Add card ID | Delete card id | Keep | Special  information |
| Byte | 1-4 | 5-8 | 9-12 | 13 |

*Special information defined as below:*

*Digit 0*：*Add card length 1 – 32digit 0 - 24 digit digit 1*：*Delete card length 1 - 32 bit 0 - 24 bit*

1. *if device model is T50*，*RET code return ACK\_FAIL*

|  |  |  |
| --- | --- | --- |
| DATA | Manage PWD length+  Manage PWD | reserved |
| Byte | 1-3 | 4-13 |

*Manage password length = Byte(1) >> 4*

Response ：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x9D | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

*if device model is T50*，*RET code return ACK\_FAIL*

1. **Read daylight saving parameter** CMD：0x1A Function：Get daylight saving flag and time zone Command：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x1A | 0x00 0x00 | CRCL CRCH |

Response：（27Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x9A | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x10 | 16Byte | CRCL CRCH |

Data format：（16Byte）

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | enable  /disable | date/week option | Start time | | | | | | | Special information | | | | | | |
| M | D | Week  of month | Day  of week | Hour | Minute | Second | M | D | Week  of month | Day  of week | Hour | Minute | Second |
| Byte | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |

*Enable/disable*：*0-disable 1-enable*；

*date/week option*：*1-date format 2-week format*； *weeks of month defined as below*：

*0x01-0x04*：*former 1-4week 0x81-0x82*：*latter 1-2 week*

*Days of week defined as below*：

*0-6*：*Sunday /Monday/Tue/Wed/Thu/Fri/Sat*

1. **Set daylight saving time parameter** CMD：0x1B Function：Set daylight saving flag and time zone Command：（26Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x1B | 0x00 0x10 | 16Byte | CRCL CRCH |

Data format：（16Byte）

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | enable  /disable | date/week option | Start time | | | | | | | Special information | | | | | | |
| M | D | Week of  month | Day of  week | Hour | Minute | Second | M | D | Week of  month | Day of  week | Hour | Minute | Second |
| Byte | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |

Response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x9B | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Read optional language combination** CMD：0x18 Function：Read optional language combination

Command ：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x18 | 0x00 0x00 | CRCL CRCH |

Response：（15Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x98 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x04 | 4Byte | CRCL CRCH |

Data format：（4Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | optional language 1 | optional language 2 | optional language 3 | optional language4 |
| Byte | 1 | 2 | 3 | 4 |

*We could set 4 optional languages, can only switch language among this 4 language once set. Optional languages defined as below*：*0xFF not select*

*0- simplified Chinese*

*1- Chinese Traditional 2-english*；

*3-Frech*；

*4-German*； *5-Spain*；

*6-Portugal*； *7-Italian*；

*8- Bulgarian*；

*9- Slovak*； *10-hungary*； *11-slovene*； *12-Turklish*； *13-Poland*； *14-Bahasa*；

*15- Romanian*；

*16-Russian* ；

1. **Set optional language combination** CMD：0x19 Function：Set optional language combination Command：（14Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x19 | 0x00 0x04 | 4Byte | CRCL CRCH |

Data format：（14Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | optional language 1 | optional language 2 | optional language 3 | optional language4 |
| Byte | 1 | 2 | 3 | 4 |

Response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x99 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **Receive feature value /card ID to execute following operation** CMD：0x78 Function：Device receive feature value/card ID from communication port, then register or match, no response.
2. If it is feature value Command：（189Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | ACK | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x78 | 0x00 0xB3 | CRCL CRCH |

Data format：（179Byte）

|  |  |  |  |
| --- | --- | --- | --- |
| DATA | Type | keep | Feature value  data |
| Byte | 1 | 2-10 | 11-179 |

Type is 1

1. If it is card ID Command：（24Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | ACK | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x78 | 0x00 0x0E | CRCL CRCH |

Data format：（14Byte）

|  |  |  |  |
| --- | --- | --- | --- |
| DATA | Type | keep | card |
| Byte | 1 | 2-10 | 11-114 |

Type is 2

1. **Get GPRS parameter** CMD：0x16

Function：get GGSN name，GPRS server/local IP address、Port number、User name and Password。

# Basic version

Command：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x16 | 0x00 0x00 | CRCL CRCH |

Response：（119Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x96 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x6C | 108Byte | CRCL CRCH |

Data format：（108Byte）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | GGSN  name | server IP  address | Port | local IP address | User name | Password | Enable/Disable | keep |
| Byte | 1-16 | 17-20 | 21-22 | 23-26 | 27-66 | 67-106 | 107 | 108 |

*If GGSN name length less than 16 byte, add 0 If local IPaddress is dynamic,23-26 byte is 0*

*If User name length less than 40 byte, add 0;if name is null,and not set User name If Password length less than 40,add 0*

*Enable/disable*：*0-disable 1-enable*

# Improved version

Command：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x16 | 0x00 0x00 | CRCL CRCH |

Response：（91Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x96 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x50 | 80Byte | CRCL CRCH |

Data format：（80Byte）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | GGSN  name | server  IPaddress | Port | local IP address | User  name | password | Enable/Disnable | Keep |
| Byte | 1-32 | 33-36 | 37-38 | 39-42 | 43-60 | 61-78 | 79 | 80 |

*If GGSN name length less than 32 byte, add 0 If local IPaddress is dynamic,33-36 byte is 0*

*If User name length less than 18 byte, add 0;if name is null,and not set User name If Password length less than 18 byte,add 0*

*Enable/disable*：*0-disable 1-enable*

1. **Set GPRS parameter** CMD：0x17

Function：set GGSN name，GPRS server/local IP address、Port number、User name and Password。

# Basic version

Command：（118Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x17 | 0x00 0x6C | 108Byte | CRCL CRCH |

Data format：（108Byte）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | GGSN  name | server  IPaddress | Port | local IP address | User  name | password | Enable/Disnable | Keep |
| Byte | 1-16 | 17-20 | 21-22 | 23-26 | 27-66 | 67-106 | 107 | 108 |

Response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x97 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

# Improved version

Command：（90Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x17 | 0x00 0x50 | 80Byte | CRCL CRCH |

Data format：（80Byte）

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DATA | GGSN  name | server  IPaddress | Port | local IP address | User  name | password | Enable/Disnable | Keep |
| Byte | 1-32 | 33-36 | 37-38 | 39-42 | 43-60 | 61-78 | 79 | 80 |

Response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x97 | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

# Get device extended information code CMD：0x7A

Function：Read vendor name/tax code/address Command：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x7A | 0x00 0x00 | CRCL CRCH |

Response：（331Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xFA | ACK\_SUCCESS  ACK\_FAIL | 0x01 0x40 | 320Byte | CRCL CRCH |

Data format：（320Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | vendor name  (UNIODE) | vendor address  (UNICODE) | vendor tax code (digit  ASCII code) | reserved |
| Byte | 1-50 | 51-150 | 151-165 | 166-320 |

1. **Modify device extend message code** CMD：0x7B Function：Modify vendor name/tax code/address Command ：（330Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x7B | 0x01 0x40 | 320Byte | CRCL CRCH |

Data format：（320Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATA | vendor name  (UNIODE) | vendor address  (UNICODE) | vendor tax code (digit  ASCII code) | reserved |
| Byte | 1-50 | 51-150 | 151-165 | 166-320 |

Response：（11Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xFB | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x00 | CRCL CRCH |

1. **inquire information of card number** CMD：0x7E T5S use only Function：inquire information of punched card on T5S

Command：（10Byte）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x7E | 0x00 0x00 | CRCL CRCH |

Response：（15Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xFE | ACK\_SUCCESS  ACK\_FAIL | 0x00 0x04 | 4Byte | CRCL CRCH |

Data format：（4Byte）

|  |  |
| --- | --- |
| DATA | Card Number |
| Byte | 1-4 |

*If T5S doesn’t get card number,card number is 0.*

1. **Sending Email** CMD：0x7F only for C5 Function：setting for sending email

Command：（11+N Byte）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | CH | CMD | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0x7F | 1+N | (1+N)Byte | CRCL CRCH |

Data format：（1+N Byte）

|  |  |  |
| --- | --- | --- |
| DATA | Parameter | Data |
| Byte | 1 | N |

Illustration for Data：

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter  value | Parameter  illustration | Data  length | Data content and format |
| 0x00 | Setting mail  server | N | SMTP address+Ending flag+User name+ Ending flag +Password+ Ending flag |
| 0x01 | Setting reading record | N | （Enable flag+Starting Hour+ Starting minute+Ending Hour+ Ending minute）\*5+Email address  Note:it can be set 5 time slots,email addresses are separated with “；” |
| 0x02 | Setting reading access record | N | （Enable flag+Starting Hour+ Starting minute+Ending Hour+ Ending minute）\*2+ Email address  Note:it can be set 2 time slots,email addresses are separated with “；” |
| 0x03 | Setting abnormal-access  record | 2 | Flag of sending abnormal-access record Note:0-not sending 1-sending |
| 0x10 | Getting mail  server | 0 |  |
| 0x11 | Getting reading  record | 0 |  |
| 0x12 | Getting reading  access record | 0 |  |
| 0x13 | Getting abnormal-access  record | 0 |  |

Response：（11+N Byte）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | CH | ACK | RET | LEN | DATA | CRC16 |
| 0xA5 | IDHH IDHL IDLH IDLL | 0xFF | ACK\_SUCCESS  ACK\_FAIL | N | (N)Byte | CRCL CRCH |

Note：Parameter value< 0x10,length of Response is 11

Parameter value >=0x10,length of Response is 11+N