**Communication protocol of aROK5510 V0.4**

|  |  |
| --- | --- |
| **版本** | **內容** |
| **V0.1** | **Base On nROK5510 V0.6** |
| **V0.2** | 1. **Modified the command content of SYS\_Get\_MCU\_Version, SYS\_Get\_DTP\_Version, SYS\_Get\_BL\_Version, BIOS\_Get\_MCU\_Version** |
| **V0.3** | 1. **Modified the command content of SYS\_Event\_Alarm** |
| **V0.4** | 1. **Add new command of BIOS\_Get\_Watchdog\_Control, BIOS\_Get\_Watchdog\_Timer** |
| **V0.5** |  |
| **V0.6** |  |
| **V0.7** |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

1. **Protocol**

Use RS232 half-duplex, Baud rate is 115200 bps, non-parity，stop bit is 1.

* 1. **Frame Based Protocol**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Start Code | Device ID | Command | Address | Data | Checksum | End code |

|  |  |  |  |
| --- | --- | --- | --- |
| **Packet** | | | |
| Item | Size | Content | Range |
| Start code | 1 Byte | 0xF1 | N/A |
| Device | 1 Byte | N/A | 0x00 ~ 0x0A |
| Command | 1 Byte | N/A | 0x00 ~ 0x03 |
| Address | 1 Byte | N/A | 0x00 ~ 0xFF |
| Data | N Byte | N/A | N/A |
| Check sum | 1 Byte | CRC8 | N/A |
| End code | 1 Byte | 0xF2 | N/A |

* 1. **Device ID List**

|  |  |  |
| --- | --- | --- |
| **裝置(Device)** | | |
| ID | Content | Support |
| 0x00 | System | Yes |
| 0x01 | Reserved | No |
| 0x02 | Reserved | No |
| 0x03 | IAP | Yes |
| 0x04 | DEBUG | No |
| 0x05 | BIOS | Yes |
| 0x06 | CAN1 | No |
| 0x07 | CAN2 | No |
| 0x08 | IOT | No |
| 0x09 | NVR | No |
| 0x0A | BKP(Backup Battery) | No |
| 0x0B | POE | No |
| 0x0C | VMD | No |
| 0x11 | ODM | No |
| 0x80 | VTK62B | No |
| 0xFF | Error | Yes |

* 1. **Command List**

|  |  |
| --- | --- |
| **Command** | |
| Index | Content |
| 0x00 | Read |
| 0x01 | Write |
| 0x02 | Get |
| 0x03 | Set |
| 0x04 | Event (Slave mode) |
| 0x05 | Reserved for ODM |
| 0x06 | ByPass |

**Read/Write Command:**

The MCU waits for the Host to issue a command and transmits or receives data according to the received specified read/write start address and length. It is designed to reduce the host's frequent transmission of commands to the MCU. It is suitable for fixed-format command tables, such as command tables between the BIOS and the MCU.

* **Read (Host >> Mcu)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Start Code | Device ID | Command | Address | Data | Checksum | End code |
| 0xF1 | ID | 0x00 | Address | Length | CRC8 | 0xF2 |

* **Read Response (Mcu >> Host)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Start Code | Device ID | Command | Address | Length | Data | Checksum | End code |
| 0xF1 | ID | 0x00 | Address | Length | data | CRC8 | 0xF2 |

* **Write (Host >> Mcu)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Start Code | Device ID | Command | Address | Length | Data | Checksum | End code |
| 0xF1 | ID | 0x01 | Address | Length | data | CRC8 | 0xF2 |

* **Write Response (Mcu >> Host)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Start Code | | Device ID | Command | Address | Data | Checksum | End code |
| 0xF1 | ID | | 0x01 | Address | 0: N / 1: Y | CRC8 | 0xF2 |

**Get/Set Command:**

The MCU waits for the Host to issue a command, and according to the received specified address, performs the function specified by the address (such as: read/write data/IO control), and is applicable to a non-fixed format command table, such as between the System and the MCU. Command table.

* **Get (Host >> Mcu)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Start Code | | Device ID | Command | Address | Checksum | End code |
| 0xF1 | ID | | 0x02 | Address | CRC8 | 0xF2 |

* **Get Response (Mcu >> Host)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Start Code | Device ID | Command | Address | Data | Checksum | End code |
| 0xF1 | ID | 0x02 | Address | data | CRC8 | 0xF2 |

* **Set (Host >> Mcu)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Start Code | Device ID | Command | Address | Data | Checksum | End code |
| 0xF1 | ID | 0x03 | Address | data | CRC8 | 0xF2 |

* **Set Response (Mcu >> Host)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Start Code | Device ID | Command | Address | Data | Checksum | End code |
| 0xF1 | ID | 0x03 | Address | 0: N / 1: Y | CRC8 | 0xF2 |

**Event Command:**

When the MCU receives some event-driven or special-condition triggers (such as Interrupt/Alarm), it actively sends an event command to the Host to notify the Host that an event has occurred.

* **Event (Mcu >> Host)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Start Code | Device ID | Command | Address | Data | Checksum | End code |
| 0xF1 | ID | 0x04 | Address | data | CRC8 | 0xF2 |

* **Event Response(Host >> Mcu)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Start Code | Device ID | Command | Address | Data | Checksum | End code |
| 0xF1 | ID | 0x04 | Address | 0: N / 1: Y | CRC8 | 0xF2 |

**ByPass Command:**

The ByPass command allows the Host to forward commands to other devices connected to the MCU via the MCU.

* **ByPass (Host >> Mcu)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Start Code | Device ID | Command | Data Extend | Length | Checksum | End code |
| 0xF1 | ID | 0x05 | 0: N / 1: Y | Length | CRC8 | 0xF2 |

* **ByPass Response (Mcu >> Host)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Start Code | Device ID | Command | Length | Data | Checksum | End code |
| 0xF1 | ID | 0x05 | Length | Data | CRC8 | 0xF2 |

**※當資料延展設定為1，MCU會自動將欲轉傳的命令，依Nexcom通訊協定的資料延展格式，進行延展處理後才傳送。**

**※當資料延展設定為0，MCU將欲轉傳的命令原封不動直接傳送。**

**※Checksum (CRC8) = DEVICE ID + COMMAND + ADDRESS + DATA**

**※Data transmission mode is low-byte at first then high-byte follow。**

**※資料回覆時間，約需等待100毫秒。**

**Check sum**

const unsigned char CRC8\_Table[16] = { 0x00, 0x5A, 0xB4, 0xEE, 0x32, 0x68, 0x86, 0xDC, 0x64, 0x3E, 0xD0, 0x8A, 0x56, 0x0C, 0xE2, 0xB8};

unsigned char Get\_CRC8(Uchar \*ptr, Uint Length)

{

unsigned int CRC\_COUNT;

unsigned char HIGH\_NIBBLE, DATA\_TEMP, NEW\_CRC = 0;

CRC\_COUNT = Length;

while(CRC\_COUNT-- != 0)

{

DATA\_TEMP = \*ptr;

HIGH\_NIBBLE = NEW\_CRC/16; /\* The upper four bits of the temporarily CRC \*/

NEW\_CRC <<=4 /\* CRC right shift 4 bits = Take the lower 12 bits of the CRC \*/

/\* Adding the upper 4 bits of the CRC and the first half of the bit then look up the table to calculate the CRC,

Then add the remainder of the last CRC \*/

NEW\_CRC ^= CRC8\_Teble[HIGH\_NIBBLE ^ (DATA\_TEMP/16)];

HIGH\_NIBBLE = NEW\_CRC/16; /\* The upper four bits of the temporarily CRC \*/

NEW\_CRC <<= 4 /\* CRC right shift 4 bits = Take the lower 12 bits of the CRC \*/

/\* Adding the upper 4 bits of the CRC and the first half of the bit then look up the table to calculate the CRC,

Then add the remainder of the last CRC \*/

NEW\_CRC ^= CRC8\_Teble[HIGH\_NIBBLE ^ (DATA\_TEMP & 0x0F)];

ptr++;

}

return NEW\_CRC;

}

* 1. **Data extension**

Because we use 0xF1 and 0xF2 for “start” and “end” code of frame protocol. In order to distinguish data of 0xF0~0xF2, we followed previous experience to adapt data extension, to make start code(0xF1) and end code(0xF2) is unique. The rule of data

extension (0xF0~0xF3) shows as below.

When data is 0xF0~0xF2 in data frame, and they will be transferred to:

1) Data is 0XF0,it will be transferred to 0xF0, 0x00.

2) Data is 0xF1, it will be transferred to 0xF0, 0x01.

3) Data is 0xF2, it will be transferred to 0xF0, 0x02.

So we can use this rule to know data 0xF0~0xF3 format and start code (0xF1) and end code (0xF2).

1. **Command set**
   1. **All Command list Table**

**2.1.1 Get Command Support Table**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **System (ID:0x00)** | **IAP (ID:0x03)** | **BIOS (ID:0x05)** |
| **0x00** |  |  |  |
| **0x01** | MCU Version | Update Mode | MCU Version |
| **0x02** | HW Version | IAP Information | ~~Power Mode~~ |
| **0x03** | DTP Version | APP Version | Delay Mode |
| **0x04** | BL Version | Flash Data | ~~Power Range~~ |
| **0x05** | System Event Log | Program State | Module Control |
| **0x06** |  |  | Alarm Hour |
| **0x07** |  |  | Alarm Min |
| **0x08** |  |  | Alarm Sec |
| **0x09** |  |  | ~~Brightness~~ |
| **0x0A** | Ignition Status |  | RTC Year |
| **0x0B** | Supply Voltage |  | RTC Month |
| **0x0C** | System Temperature |  | RTC Day |
| **0x0D** |  |  | RTC Weekday |
| **0x0E** |  |  |  |
| **0x0F** |  |  |  |
| **0x10** | ~~Power Type~~ |  |  |
| **0x11** | Delay Time Option |  |  |
| **0x12** | ~~Startup Shutdown Option~~ |  |  |
| **0x13** | Alarm timer |  | Update Status |
| **0x14** |  |  | OS Status |
| **0x15** | WWAN Status |  |  |
| **0x16** | WIFI Status |  | Auto Reboot Status |
| **0x17** | BT Status |  | MB Version |
| **0x18** | GPS Status |  |  |
| **0x19** | USB Status |  |  |
| **0x1A** |  |  |  |
| **0x1B** | RTC Status |  |  |
| **0x1C** | Watchdog Configure |  |  |
| **0x1D** |  |  |  |
| **0x1E** |  |  |  |
| **0x1F** |  |  |  |
| **0x20** | Flash Status |  | Watchdog Control |
| **0x21** |  |  | Watchdog Timer |
| **0x22** |  |  |  |
| **0x23** | ~~G Sensor Data~~ |  |  |
| **0x24** |  |  |  |
| **0x25** |  |  |  |
| **0x26** |  |  |  |
| **0x27** |  |  |  |
| **0x28** |  |  |  |
| **0x29** |  |  |  |
| **0x2A** | ~~CAN Power Status~~ |  |  |
| **0x2B** |  |  |  |
| **0x2C** |  |  |  |
| **0x2D** | SIM Card Status |  | Poe Fan Value |
| **0x2E** |  |  | iAMT\_Low\_Power\_Wake\_Up |
| **0x2F** | GPO Status |  |  |
| **0x30** | GPI Status |  | Graphics Fan Value |
| **0x31** |  |  | PCIE Slot Mode |
| **0x32** |  |  |  |
| **0x33** | Program Led Status |  |  |
| **0x34** |  |  |  |
| **0x35** |  |  |  |
| **0x36** | Wake On Lan Status |  |  |
| **0x37** | ~~External 12V~~ |  |  |
| **0x38** |  |  |  |
| **0x39** |  |  |  |
| **0x3A** | GPI Active Mode |  |  |
| **0x3B** | GPO Pull Up Mode |  |  |
| **0x3C** |  |  |  |
| **0x3D** |  |  |  |
| **0x3E** |  |  |  |
| **0x3F** |  |  |  |
| **0x40** |  |  |  |
| **0x41** | Fan Value |  |  |
| **0x42** | Fan RPM |  |  |

* + 1. **Set Command Support Table**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **System(ID:0x01)** | **IAP(ID:0x03)** | **BIOS(ID:0x05)** |
| **0x00** |  |  |  |
| **0x01** |  | Update Request |  |
| **0x02** |  | Flash Unlock | ~~Power Mode~~ |
| **0x03** |  | Erase Flash | Delay Mode |
| **0x04** |  | Write Flash | ~~Power Range~~ |
| **0x05** | System Event Clear | Check Code | Module Control |
| **0x06** |  | IAP Reboot | Alarm Hour |
| **0x07** |  |  | Alarm Min |
| **0x08** | Load Default |  | Alarm Sec |
| **0x09** | RTC timer |  | ~~Brightness~~ |
| **0x0A** |  |  | RTC Year |
| **0x0B** |  |  | RTC Month |
| **0x0C** |  |  | RTC Day |
| **0x0D** |  |  | RTC Weekday |
| **0x0E** |  |  |  |
| **0x0F** |  |  |  |
| **0x10** | ~~Power Type~~ |  | RTC Hour |
| **0x11** | Delay Time Option |  | RTC Min |
| **0x12** | ~~Startup Shutdown Option~~ |  | RTC Sec |
| **0x13** | Alarm timer |  | Update Mode |
| **0x14** |  |  | OS Mode |
| **0x15** | WWAN Control |  |  |
| **0x16** | WIFI Control |  | Auto Reboot Control |
| **0x17** | BT Control |  | MB Version |
| **0x18** | GPS Control |  |  |
| **0x19** | USB Control |  |  |
| **0x1A** |  |  |  |
| **0x1B** | RTC Control |  |  |
| **0x1C** | Watchdog Control |  |  |
| **0x1D** | Watchdog Timer |  |  |
| **0x1E** |  |  |  |
| **0x1F** |  |  |  |
| **0x20** | Flash Update |  | Watchdog Control |
| **0x21** |  |  | Watchdog Timer |
| **0x22** |  |  |  |
| **0x23** | ~~G Sensor Data~~ |  |  |
| **0x24** |  |  |  |
| **0x25** |  |  |  |
| **0x26** |  |  |  |
| **0x27** |  |  |  |
| **0x28** |  |  |  |
| **0x29** |  |  |  |
| **0x2A** | ~~CAN Power Control~~ |  |  |
| **0x2B** |  |  |  |
| **0x2C** |  |  |  |
| **0x2D** | SIM Card Selected |  | Poe Fan Value |
| **0x2E** |  |  | iAMT\_Low\_Power\_Wake\_Up |
| **0x2F** | GPO Control |  |  |
| **0x30** |  |  | Graphics Fan Value |
| **0x31** |  |  | PCIE Slot Control |
| **0x32** |  |  |  |
| **0x33** | Program Led Control |  |  |
| **0x34** |  |  |  |
| **0x35** |  |  |  |
| **0x36** | Wake On Lan Control |  |  |
| **0x37** | ~~External 12V~~ |  |  |
| **0x38** |  |  |  |
| **0x39** |  |  |  |
| **0x3A** | GPI Active Level Control |  |  |
| **0x3B** | GPO Pull Up Control |  |  |
| **0x3C** |  |  |  |
| **0x3D** |  |  |  |
| **0x3E** |  |  |  |
| **0x3F** |  |  |  |
| **0x40** |  |  |  |
| **0x41** | Fan Value |  |  |
| **0x42** |  |  |  |

* 1. **System command set**

**2.2.1 ID:0x00(System) Get Command Address**

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **名稱(Name)** | **Default** | **Address** |
| Get | MCU Version |  | 0x01 |
| Get | HW Version |  | 0x02 |
| Get | DTP Version |  | 0x03 |
| Get | BL Version |  | 0x04 |
| Get | System Event Log |  | 0x05 |
| Get | Ignition Status |  | 0x0A |
| Get | Supply Voltage |  | 0x0B |
| Get | System Temperature |  | 0x0C |
| ~~Get~~ | ~~Power Type~~ | ~~0x00~~ | ~~0x10~~ |
| Get | Delay Time Option | 0x00 | 0x11 |
| ~~Get~~ | ~~Startup Shutdown Option~~ | ~~0x00~~ | ~~0x12~~ |
| Get | Alarm Timer |  | 0x13 |
| Get | WWAN Status |  | 0x15 |
| Get | WIFI Status |  | 0x16 |
| Get | BT Status |  | 0x17 |
| Get | GPS Status |  | 0x18 |
| Get | USB Status |  | 0x19 |
| Get | RTC Status |  | 0x1B |
| Get | Watchdog Configure |  | 0x1C |
| Get | Flash Status |  | 0x20 |
| ~~Get~~ | ~~G Sensor Data~~ |  | ~~0x23~~ |
| ~~Get~~ | ~~CAN Power Status~~ |  | ~~0x2A~~ |
| Get | SIM Card Status |  | 0x2D |
| Get | GPO Status |  | 0x2F |
| Get | GPI Status |  | 0x30 |
| Get | Program Led Status |  | 0x33 |
| Get | Wake On Lan Status |  | 0x36 |
| ~~Get~~ | ~~External 12V~~ |  | ~~0x37~~ |
| Get | GPI Active Mode |  | 0x3A |
| Get | GPO Pull Up Mode |  | 0x3B |
| Get | Fan Value |  | 0x41 |
| Get | Fan RPM |  | 0x42 |

* + 1. **ID:0x00(System) Set Command Address**

|  |  |  |
| --- | --- | --- |
| **Type** | **名稱(Name)** | **Address** |
| Set | System Event Clear | 0x05 |
| Set | Load Default | 0x08 |
| Set | RTC Timer | 0x09 |
| ~~Set~~ | ~~Power Type~~ | ~~0x10~~ |
| Set | Delay Time Option | 0x11 |
| ~~Set~~ | ~~Startup Shutdown Option~~ | ~~0x12~~ |
| Set | Alarm Timer | 0x13 |
| Set | WWAN Control | 0x15 |
| Set | WIFI Control | 0x16 |
| Set | BT Control | 0x17 |
| Set | GPS Control | 0x18 |
| Set | USB Control | 0x19 |
| Set | RTC Control | 0x1B |
| Set | Watchdog Control | 0x1C |
| Set | Watchdog Timer | 0x1D |
| Set | Flash Update | 0x20 |
| ~~Set~~ | ~~G Sensor Data~~ | ~~0x23~~ |
| ~~Set~~ | ~~CAN Power Control~~ | ~~0x2A~~ |
| Set | SIM Card Selected | 0x2D |
| Set | GPO Control | 0x2F |
| Set | Program Led Control | 0x33 |
| Set | Wake On Lan Control | 0x36 |
| ~~Set~~ | ~~External 12V~~ | ~~0x37~~ |
| Set | GPI Active Level Control | 0x3A |
| Set | GPO Pull Up Control | 0x3B |
| Set | Fan Value | 0x41 |
|  |  |  |

* + 1. **ID:0x00(System) Event Command Address**

|  |  |  |
| --- | --- | --- |
| **Type** | **名稱(Name)** | **Address** |
| Event | Reserved | 0x00 |
| Event | Reserved | 0x01 |
| Event | Reserved | 0x02 |
| Event | System Alarm | 0x03 |

* + 1. **Get Command List**
       1. **Command 0x01: GetMCUVersion**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x01 0x?? 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x01 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| MCU Version | 1 | 0x01~0x0FF |

Example: MCU return current MCU version as below.

->0xF1 0x00 0x02 0x01 0x01 CRC 0xF2

* + - 1. **Command 0x02: GetHWVersion**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x02 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x02 Data 0 CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| HW Version | 1 | 0x00~0x07(A~H) |

Example: MCU return current HW version as below.

->0xF1 0x00 0x02 0x02 0x01 CRC 0xF2

* + - 1. **Command 0x03: GetDTPVersion**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x03 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x03 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| DTP Version | 1 | 0x01~0x0FF |

Example: MCU return current DTP version as below.

->0xF1 0x00 0x02 0x03 0x01 CRC 0xF2

* + - 1. **Command 0x04: GetBLVersion**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x04 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x04 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| BL Version | 1 | 0x01~0x0FF |

Example: MCU return current DTP version as below.

->0xF1 0x00 0x02 0x04 0x01 CRC 0xF2

* + - 1. **Command 0x05: GetSystemEventLog**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x05 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x05 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| First Year | 1 | 0 ~ 99 |
| First Month | 1 | 1 ~ 12 |
| First Day | 1 | 1 ~ 31 per Month |
| First Hour | 1 | 0 ~ 23 |
| First Min | 1 | 0 ~ 59 |
| First Sec | 1 | 0 ~ 59 |
| Last Year | 1 | 0 ~ 99 |
| Last Month | 1 | 1 ~ 12 |
| Last Day | 1 | 1 ~ 31 per Month |
| Last Hour | 1 | 0 ~ 23 |
| Last Min | 1 | 0 ~ 59 |
| Last Sec | 1 | 0 ~ 59 |
| Startup Count | 4 | 0x00000000 ~ 0xFFFFFFFF |
| Shutdown Count | 4 | 0x00000000 ~ 0xFFFFFFFF |
| Over Volt Count | 4 | 0x00000000 ~ 0xFFFFFFFF |
| Low Volt Count | 4 | 0x00000000 ~ 0xFFFFFFFF |

Example: MCU return current system event log as below.

->0xF1 0x00 0x02 0x05 0x11 0x04 0x01 0x12 0x00 0x00

0x11 0x04 0x01 0x13 0x00 0x00 0x02 0x00 0x00 0x00

0x01 0x00 0x00 0x00 0x01 0x00 0x00 0x00 0x01 0x00

0x00 0x00 0x?? 0xF2

* + - 1. **Command 0x0A: GetIgnitionStatus**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x0A CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x0A Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Ignition Status | 1 | 0x00 : Ignition off  0x01 : Ignition on |

Example: MCU return current ignition status as below.

->0xF1 0x00 0x02 0x0A 0x01 CRC 0xF2

* + - 1. **Command 0x0B: GetSupplyVoltage**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x0B CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x0B Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Supply Voltage | 2 | 0 ~ 380 (unit: 100mV) |

Example: MCU return current supply voltage is 12.0V as below.

->0xF1 0x00 0x02 0x0B 0x78 0x00 CRC 0xF2

* + - 1. **Command 0x0C: GetSystemTemperature**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x0C CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x0C Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| System Temperature | 1 | -27 ℃ ~ 125 ℃ |

Example: MCU return current system temperature is 25 ℃ as below.

->0xF1 0x00 0x02 0x0C 0x19 CRC 0xF2

Example: MCU return current system temperature is -5 ℃ as below.

->0xF1 0x00 0x02 0x0C 0xFB CRC 0xF2

* + - 1. **~~Command 0x10: GetPowerType~~**

~~Direction: Host -> MCU~~

~~The package sent by host.~~

~~-> 0xF1 0x00 0x02 0x10 CRC 0xF2~~

~~Direction: MCU -> Host~~

~~The response package received from mcu.~~

~~-> 0xF1 0x00 0x02 0x10 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Power Type~~ | ~~1~~ | ~~0x03: 12V~~  ~~0x02: 24V~~  ~~0x01: Reserved(9~36)~~  ~~0x00: 9~36V(default)~~ |

~~Example: MCU return current power type as below.~~

~~->0xF1 0x00 0x02 0x10 0x03 CRC 0xF2~~

* + - 1. **Command 0x11: GetDelayTimeOption**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x11 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x11 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Delay Time | 1 | Bit 0~2: Power off  000: 20 Sec  001: 1 Min  010: 5 Mins  011: 10 Mins  100: 30 Mins  101: 1 Hour  110: 6 Hours  111: 18 Hours  Bit 3~5: Power on  000: 10 Sec  001: 30 Sec  010: 1Min  011: 5 Mins  100: 10 Mins  101: 15 Mins  110: 30 Mins  111: 1 Hour  Bit 6: Delay off control  0: Disable, 1: Enable (default Disable)  Bit7: Delay on control  0: Disable, 1: Enable(default Disable) |

Example: MCU return current delay time option as below.

->0xF1 0x00 0x02 0x11 0x41 CRC 0xF2

* + - 1. **~~Command 0x12: GetStartupShutdownOption~~**

~~Direction: Host -> MCU~~

~~The package sent by host.~~

~~-> 0xF1 0x00 0x02 0x12 CRC 0xF2~~

~~Direction: MCU -> Host~~

~~The response package received from mcu.~~

~~-> 0xF1 0x00 0x02 0x12 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Startup / Shutdown~~ | ~~1~~ | ~~12V / 24V~~  ~~Startup/Shutdown Startup/Shutdown~~  ~~0x00: 11.5V 10.5V 23.0V 21.0V~~  ~~0x01: 12.0V 11.0V 24.0V 22.0V~~  ~~0x02: 12.5V 11.0V 25.0V 22.0V~~  ~~0x03: 12.5V 11.5V 25.0V 23.0V~~ |

~~Example: MCU return current startup shutdown option as below.~~

~~->0xF1 0x00 0x02 0x12 0x03 CRC 0xF2~~

* + - 1. **Command 0x13: GetAlarmTimer**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x13 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x13 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Alarm Hour | 1 | 0 ~ 23 |
| Alarm Min | 1 | 0 ~ 59 |
| Alarm Sec | 1 | 0 ~ 59 |

Example: Mcu return current alarm timer as below.

->0xF1 0x00 0x02 0x13 0x01 0x09 0x0F CRC? 0xF2

* + - 1. **Command 0x15: GetWWANStatus**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x15 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x15 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| WWAN Status | 1 | Bit 0: (CN7)  Reserved  Bit 1: (CN7)  0: Disable WWANA  1: Enable WWANA(default)  Bit 2: (CN6)  Reserved  Bit 3: (CN6)  0: Disable WWANB  1: Enable WWANB(default)  Bit 4: (CN5)  Reserved  Bit 5: (CN5)  0: Disable WWANC  1: Enable WWANC(default)  Bit 6: (CN3)  0: Disable WWAND wakeup mode(default)  1: Enable WWAND wakeup mode  Bit 7: (CN3)  0: Disable WWAND  1: Enable WWAND(default) |
| WWAN Status1 | 1 | Bit 0: (CN20)  Reserved  Bit 1: (CN20)  0: Disable WWANE  1: Enable WWANE(default)  Bit 2~7: Reserved |

Example: MCU return current WWAN status as below.

->0xF1 0x00 0x02 0x15 0x10 0x00 CRC 0xF2

* + - 1. **Command 0x16: GetWifiStatus**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x16 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x16 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Wifi Status | 1 | Bit 0: (CN20)  0: WifiA Enable Off  1: WifiA Enable On(default)  Bit 1: (CN24)  0: WifiB Enable Off  1: WifiB Enable On(default)  Bit 2:  Always 0  Bit 3: (CN20)  0: WifiA Power Off  1: WifiA Power On(default)  Bit 4: (CN24)  0: WifiB Power Off  1: WifiB Power On(default)  Bit 5: (CN22)  0: WifiC Power Off  1: WifiC Power On(default)  Bit 6~7: Reserved |

Example: MCU return current wifi status as below.

->0xF1 0x00 0x02 0x16 0x00 CRC 0xF2

* + - 1. **Command 0x17: GetBTStatus**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x17 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x17 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| BT Status | 1 | Bit 0: (CN20)  0: BTA Off  1: BTA On(default)  Bit 1: (CN24)  0: BTB Off  1: BTB On(default)  Bit 2~7: Reserved |

Example: MCU return current BT status as below.

->0xF1 0x00 0x02 0x17 0x00 CRC 0xF2

* + - 1. **Command 0x18: GetGPSStatus**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x18 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x18 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| GPS Status | 1 | 0: GPS Off  1: GPS On(default) |

Example: MCU return current GPS status as below.

->0xF1 0x00 0x02 0x18 0x01 CRC 0xF2

* + - 1. **Command 0x19: GetUSBStatus**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x19 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x19 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| USB Status | 1 | Bit 0:  0: USB 0 Off  1: USB 0 On(default)  Bit 1:  0: USB 1 Off  1: USB 1 On(default)  Bit 2:  0: USB 2 Off  1: USB 2 On(default)  Bit 3:  0: USB 3 Off  1: USB 3 On(default)  Bit 4:  0: USBR Off  1: USBR On(default)  Bit5~7: Always 0 |

Example: MCU return current USB status as below.

->0xF1 0x00 0x02 0x19 0x11 CRC 0xF2

* + - 1. **Command 0x1B: GetRTCStatus**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x1B CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x1B Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| RTC Status | 1 | Bit 0:  0: RTC Off(default)  1: RTC On |

Example: MCU return current RTC status as below.

->0xF1 0x00 0x02 0x1B 0x01 CRC 0xF2

* + - 1. **Command 0x1C: GetWatchdogConfig**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x1C CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x1C Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Control | 1 | Bit 6~0: always 0  Bit 7: WDT enable  0: Disable(default)  1: Enable |
| Timeout | 1 | 3 ~ 255 (Sec) |

Example: MCU return current watchdog configure as below.

->0xF1 0x00 0x02 0x1C 0x80 0x03 CRC 0xF2

* + - 1. **Command 0x20: GetFlashStatus**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x20 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x20 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Flash Status | 1 | 0: Flash Store Complete  1: Flash Storing |

Example: MCU return current flash status as below.

->0xF1 0x00 0x02 0x20 0x00 CRC 0xF2

* + - 1. **~~Command 0x23: GetGsenserData~~**

~~Direction: Host -> MCU~~

~~The package sent by host.~~

~~-> 0xF1 0x00 0x02 0x23 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Register Index~~ | ~~1~~ | ~~0x00 ~ 0x39~~ |
| ~~Length~~ | ~~1~~ | ~~0x01 ~ 0x7F~~ |

~~Example: Host get g-sensor data as below.~~

~~->0xF1 0x00 0x02 0x23 0x00 0x01 CRC 0xF2~~

~~Direction: MCU -> Host~~

~~The response package received from mcu.~~

~~-> 0xF1 0x00 0x02 0x23 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Register Index~~ | ~~1~~ | ~~0x00 ~ 0x39~~ |
| ~~Data~~ | ~~N~~ | ~~0x00 ~ 0xFF~~ |

~~Example: MCU return current g-sensor data as below.~~

~~->0xF1 0x00 0x02 0x23 0x00 0xE5 CRC 0xF2~~

* + - 1. **~~Command 0x2A: GetCANPowerStatus~~**

~~Direction: Host -> MCU~~

~~The package sent by host.~~

~~-> 0xF1 0x00 0x02 0x2A CRC 0xF2~~

~~Direction: MCU -> Host~~

~~The response package received from mcu.~~

~~-> 0xF1 0x00 0x02 0x2A Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~CAN Power Status~~ | ~~1~~ | ~~0: CAN Power Off~~  ~~1: CAN Power On(default)~~ |

~~Example: MCU return current frequency as below.~~

~~->0xF1 0x00 0x02 0x2A 0x01 CRC 0xF2~~

* + - 1. **Command 0x2D: GetSIMCardStatus**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x2D CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x2D Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Status | 1 | Bit 0:  0: Select SIMA Card1(default)  1: Select SIMA Card2  Bit 1:  0: Select SIMB Card1(default)  1: Select SIMB Card2  Bit 2:  0: Select SIMC Card1(default)  1: Select SIMC Card2  Bit 3:  0: Select SIMD Card1(default)  1: Select SIMD Card2  Bit 4:  0: Select SIME Card1(default)  1: Select SIME Card2  Bit 5~7: Reserved |

Example: MCU return current SIM card status as below.

->0xF1 0x00 0x02 0x2D 0x01 CRC 0xF2

* + - 1. **Command 0x2F: GetGPOStatus**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x2F CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x2F Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Status | 1 | Bit 0: GPO1 Pin State  Bit 1: GPO2 Pin State  Bit 2: GPO3 Pin State  Bit 3: GPO4 Pin State  Bit4~7: Reserved |

Example: MCU return current gpo status as below.

->0xF1 0x00 0x02 0x2F 0x01 CRC 0xF2

* + - 1. **Command 0x30: GetGPIStatus**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x30 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x30 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Status | 1 | Bit 0: GPI1 Pin State  Bit 1: GPI2 Pin State  Bit 2: GPI3 Pin State  Bit 3: GPI4 Pin State  Bit4~7: Reserved |

Example: MCU return current gpi status as below.

->0xF1 0x00 0x02 0x30 0x01 CRC 0xF2

* + - 1. **Command 0x33: GetProgramLEDStatus**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x33 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x33 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| LED Status | 1 | Bit 0~1: P1 LED  0: OFF(default)  ~~1: Orange ON~~  ~~2: Green ON~~  1: Red ON  2: Yellow ON  Bit 2~3: P2 LED  0: OFF(default)  ~~1: Orange ON~~  ~~2: Green ON~~  1: Red ON  2: Yellow ON  Bit 4~7: Reserved |

Example: MCU return current program led status as below.

->0xF1 0x00 0x02 0x33 0x01 CRC 0xF2

* + - 1. **Command 0x36: GetWakeOnLanStatus**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x36 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x36 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Wake On Lan Status | 1 | 0: Wake on lan function disable (default)  1: Wake on lan function enable |

Example: MCU return current wake on lan status as below.

->0xF1 0x00 0x02 0x36 0x01 CRC 0xF2

* + - 1. **~~Command 0x37: GetExternal12V~~**

~~Direction: Host -> MCU~~

~~The package sent by host.~~

~~-> 0xF1 0x00 0x02 0x37 CRC 0xF2~~

~~Direction: MCU -> Host~~

~~The response package received from mcu.~~

~~-> 0xF1 0x00 0x02 0x37 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Status~~ | ~~1~~ | ~~0: power off~~  ~~1: power on~~ |

~~Example: MCU return current external 12v status as below.~~

~~->0xF1 0x00 0x02 0x37 0x01 CRC 0xF2~~

**2.2.4.29 Command 0x3A: GetGPIActiveMode**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x3B 0xF2

Example: Host get general purpose input active mode as below.

-> 0xF1 0x00 0x02 0x3A CRC 0xF2。

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x3A Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| GPI Level | 1 | 0: Active High, 1: Active Low (Default)  Bit 0: GPI12  Bit 1: GPI34  Bit 2~7: Always 0. |

Example: Mcu Return package as below

-> 0xF1 0x00 0x02 0x3A 0x00 CRC 0xF2。

* + - 1. **Command 0x3B: GetGPOPullUpMode**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x3B CRC 0xF2

Example: Host gets general purpose output pull up mode as below.

-> 0xF1 0x00 0x02 0x3B CRC 0xF2。

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x02 0x3B Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| GPO Pull up | 1 | 0: Internal pull high (default), 1: External pull high  Bit 0: GPO12  Bit 1: GPO34  Bit 2~7: Always 0. |

Example: Mcu Return package as below

-> 0xF1 0x00 0x02 0x3B 0x00 CRC 0xF2。

* + - 1. **Command 0x41: GetFANVALUE**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x41 0x62 0xF2

Direction: Host -> MCU

The response package received from mcu.

-> 0xF1 0x00 0x02 0x41 Data Data1 Data2 CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| SYS\_FAN4(Data) | 1 | Always 0. |
| POE\_FAN(Data1) | 1 | Range: 0~100  OFF: 0  ON: 1~100 |
| GRAPHICS\_FAN(Data2) | 1 | Range: 0~100  OFF: 0  ON: 1~100 |

Example: Host set main board version as below.

->0xF1 0x00 0x02 0x41 0x01 0x01 CRC 0xF2

* + - 1. **Command 0x42: GET\_FAN\_RPM**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x02 0x42 0x8C 0xF2

Direction: Host -> MCU

The response package received from mcu.

-> 0xF1 0x00 0x02 0x42 Data Data1 Data2 .. Data5 CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| SYS FAN1 RPM | 2 | 0 ~ 65535 |
| SYS FAN2 RPM | 2 | 0 ~ 65535 |
| SYS FAN3 RPM | 2 | 0 ~ 65535 |
| SYS FAN4 RPM | 2 | 0 ~ 65535 |
| POE FAN RPM | 2 | 0 ~ 65535 |
| GRAPHICS FAN RPM | 2 | 0 ~ 65535 |

Example: Host set main board version as below.

->0xF1 0x00 0x02 0x42 0x01 0x01 0x01 0x01 0x01 0x01 0x01 0x01

0x01 0x01 0x01 0x01 CRC 0xF2

* + 1. **Set Command List**
       1. **Command 0x05: SetSystemEventClear**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x05 Data CRC 0xF2

Example: Host set system event log initial as below.

->0xF1 0x00 0x03 0x05 00 CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| System Event Clear | 1 | 0x00: Clear All Log |

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x05 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x00 0x03 0x05 0x01 CRC 0xF2

* + - 1. **Command 0x08: SetLoadDefault**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x08 CRC 0xF2

Example: Host set mcu load default as below.

->0xF1 0x00 0x03 0x05 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x08 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x00 0x03 0x08 0x01 CRC 0xF2

* + - 1. **Command 0x09: SetRTCTimer**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x09 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| RTC Hour | 1 | 0 ~ 23 |
| RTC Min | 1 | 0 ~ 59 |
| RTC Sec | 1 | 0 ~ 59 |

Example: Host set RTC timer as below.

->0xF1 0x00 0x03 0x09 0x00 0x00 0x00 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x09 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x00 0x03 0x09 0x01 CRC 0xF2

* + - 1. **~~Command 0x10: SetPowerType~~**

~~Direction: Host -> MCU~~

~~The package sent by host.~~

~~-> 0xF1 0x00 0x03 0x10 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Power Type~~ | ~~1~~ | ~~0x05: 36V~~  ~~0x04: 48V~~  ~~0x03: 12V~~  ~~0x02: 24V~~  ~~0x01: Reserved (9~60)~~  ~~0x00: 9 ~ 60V (default)~~ |

~~Example: Host set power type as below.~~

~~->0xF1 0x00 0x03 0x10 0x00 CRC 0xF2~~

~~Direction: MCU -> Host~~

~~The response package received from mcu.~~

~~-> 0xF1 0x00 0x03 0x10 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Command Status~~ | ~~1~~ | ~~0x00: Setting Fail~~  ~~0x01: Setting Success~~ |

~~Example: MCU return setting result as below.~~

~~->0xF1 0x00 0x03 0x10 0x01 CRC 0xF2~~

* + - 1. **Command 0x11: SetDelayTimeOption**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x11 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Delay Time | 1 | Bit 0~2: Power off  000: 20 Sec  001: 1 Min  010: 5 Mins  011: 10 Mins  100: 30 Mins  101: 1 Hour  110: 6 Hours  111: 18 Hours  Bit 3~5: Power on  000: 10 Sec  001: 30 Sec  010: 1Min  011: 5 Mins  100: 10 Mins  101: 15 Mins  110: 30 Mins  111: 1 Hour  Bit 6: Delay off control  0: Disable, 1: Enable (default Disable)  Bit7: Delay on control  0: Disable, 1: Enable(default Disable) |

Example: Host set delay time option as below.

->0xF1 0x00 0x03 0x11 0x00 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x11 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x00 0x03 0x11 0x01 CRC 0xF2

* + - 1. **~~Command 0x12: SetStartupShutdownOption~~**

~~Direction: Host -> MCU~~

~~The package sent by host.~~

~~-> 0xF1 0x00 0x03 0x12 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Startup / Shutdown~~ | ~~1~~ | ~~12V / 24V~~  ~~Startup/Shutdown Startup/Shutdown~~  ~~0x00: 11.5V 10.5V 23.0V 21.0V~~  ~~0x01: 12.0V 11.0V 24.0V 22.0V~~  ~~0x02: 12.5V 11.0V 25.0V 22.0V~~  ~~0x03: 12.5V 11.5V 25.0V 23.0V~~ |

~~Example: Host set startup shutdown option as below.~~

~~->0xF1 0x00 0x03 0x12 0x00 CRC 0xF2~~

~~Direction: MCU -> Host~~

~~The response package received from mcu.~~

~~-> 0xF1 0x00 0x03 0x12 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Command Status~~ | ~~1~~ | ~~0x00: Setting Fail~~  ~~0x01: Setting Success~~ |

~~Example: MCU return setting result as below.~~

~~->0xF1 0x00 0x03 0x12 0x01 CRC 0xF2~~

* + - 1. **Command 0x13: SetAlarmTimer**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x13 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Alarm Hour | 1 | 0 ~ 23 |
| Alarm Min | 1 | 0 ~ 59 |
| Alarm Sec | 1 | 0 ~ 59 |

Example: Host set alarm timer as below.

->0xF1 0x00 0x03 0x13 0x00 0x00 0x00 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x13 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x00 0x03 0x13 0x01 CRC 0xF2

* + - 1. **Command 0x15: SetWWANControl**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x15 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| WWAN Control | 1 | Bit 0: (CN7)  Reserved  Bit 1: (CN7)  0: Disable WWANA  1: Enable WWANA(default)  Bit 2: (CN6)  Reserved  Bit 3: (CN6)  0: Disable WWANB  1: Enable WWANB(default)  Bit 4: (CN5)  Reserved  Bit 5: (CN5)  0: Disable WWANC  1: Enable WWANC(default)  Bit 6: (CN3)  0: Disable WWAND wakeup mode(default)  1: Enable WWAND wakeup mode  Bit 7: (CN3)  0: Disable WWAND  1: Enable WWAND(default) |
| WWAN Control1 | 1 | Bit 0: (CN20)  Reserved  Bit 1: (CN20)  0: Disable WWANE  1: Enable WWANE(default)  Bit 2~7: Reserved |

Example: Host set wwan control as below.

->0xF1 0x00 0x03 0x15 0x01 0x00 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x15 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x00 0x03 0x15 0x01 CRC 0xF2

* + - 1. **Command 0x16: SetWifiControl**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x16 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Wifi Control | 1 | Bit 0: (CN20)  0: Disable WifiA  1: Enable WifiA(default)  Bit 1: (CN24)  0: Disable WifiB  1: Enable WifiB(default)  Bit 2:  Always 0.  Bit 3: (CN20)  0: Disable WifiA Power  1: Enable WifiA Power (default)  Bit 4: (CN24)  0: Disable WifiB Power  1: Enable WifiB Power (default)  Bit 5: (CN22)  0: Disable WifiC Power  1: Enable WifiC Power (default)  Bit 6~7: Reserved |

Example: Host set wifi control as below.

->0xF1 0x00 0x03 0x16 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x16 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x00 0x03 0x16 0x01 CRC 0xF2

* + - 1. **Command 0x17: SetBTControl**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x17 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| BT Control | 1 | Bit 0:  0: Disable BTA  1: Enable BTA(default)  Bit 1:  0: Disable BTB  1: Enable BTB(default)  Bit 2~7: Reserved |

Example: Host set BT control as below.

->0xF1 0x00 0x03 0x17 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x17 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x00 0x03 0x17 0x01 CRC 0xF2

* + - 1. **Command 0x18: SetGPSControl**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x18 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| GPS Control | 1 | 0: Disable GPS  1: Enable GPS(default) |

Example: Host set GPS control as below.

->0xF1 0x00 0x03 0x18 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x18 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x00 0x03 0x18 0x01 CRC 0xF2

* + - 1. **Command 0x19: SetUSBControl**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x19 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| USB Control | 1 | Bit 0:  0: Disable USB 0  1: Enable USB 0(default)  Bit 1:  0: Disable USB 1  1: Enable USB 1(default)  Bit 2:  0: USB 2 Off  1: USB 2 On(default)  Bit 3:  0: USB 3 Off  1: USB 3 On(default)  Bit 4:  0: USBR Off  1: USBR On(default)  Bit5~7: Always 0 |

Example: Host set USB control as below.

->0xF1 0x00 0x03 0x19 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x19 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x00 0x03 0x19 0x01 CRC 0xF2

* + - 1. **Command 0x1B: SetRTCControl**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x1B Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| RTC Control | 1 | 0: Disable RTC wakeup(default)  1: Enable RTC wakeup |

Example: Host set SDCard control as below.

->0xF1 0x00 0x03 0x1B 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x1B Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x00 0x03 0x1B 0x01 CRC 0xF2

* + - 1. **Command 0x1C: SetWatchdogControl**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x1C Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Control | 1 | Bit 6~0: always 0  Bit 7:  0: Disable(default)  1: Enable |

Example: Host set watchdog control as below.

->0xF1 0x00 0x03 0x1C 0x80 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x1C Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x00 0x03 0x1C 0x01 CRC 0xF2

* + - 1. **Command 0x1D: SetWatchdogTimer**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x1D Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Timeout | 1 | 3 ~ 255 (Sec) |

Example: Host set watchdog timeout as below.

->0xF1 0x00 0x03 0x1D 0x03 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x1D Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x00 0x03 0x1D 0x01 CRC 0xF2

* + - 1. **Command 0x20: SetFlashUpdate**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x20 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Flash Update | 1 | 0x01: Update |

Example: Host set flash update as below.

->0xF1 0x00 0x03 0x20 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x20 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x00 0x03 0x20 0x01 CRC 0xF2

* + - 1. **~~Command 0x23: SetGsensorData~~**

~~Direction: Host -> MCU~~

~~The package sent by host.~~

~~-> 0xF1 0x00 0x03 0x23 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Register Index~~ | ~~1~~ | ~~0x01 ~ 0x39~~ |
| ~~Length~~ | ~~1~~ | ~~0x01 ~ 0x40~~ |
| ~~Data~~ | ~~N~~ | ~~Write data content~~ |

~~Example: Host set g-sensor data as below.~~

~~->0xF1 0x00 0x03 0x23 0x21 0x01 0x0A CRC 0xF2~~

~~Direction: MCU -> Host~~

~~The response package received from mcu.~~

~~-> 0xF1 0x00 0x03 0x23 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Command Status~~ | ~~1~~ | ~~0x00: Setting Fail~~  ~~0x01: Setting Success~~ |

~~Example: MCU return setting result as below.~~

~~->0xF1 0x00 0x03 0x22 0x01 CRC 0xF2~~

* + - 1. **~~Command 0x2A: SetCANPowerControl~~**

~~Direction: Host -> MCU~~

~~The package sent by host.~~

~~-> 0xF1 0x00 0x03 0x2A Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~CAN Power Control~~ | ~~1~~ | ~~0: CAN Power OFF~~  ~~1: CAN Power ON(default)~~ |

~~Example: Host set GPS control as below.~~

~~->0xF1 0x00 0x03 0x2A 0x01 CRC 0xF2~~

~~Direction: MCU -> Host~~

~~The response package received from mcu.~~

~~-> 0xF1 0x00 0x03 0x2A Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Command Status~~ | ~~1~~ | ~~0x00: Setting Fail~~  ~~0x01: Setting Success~~ |

~~Example: MCU return setting result as below.~~

~~->0xF1 0x00 0x03 0x2A 0x01 CRC 0xF2~~

* + - 1. **Command 0x2D: SetSIMCardSelect**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x2D Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| SIM Card | 1 | Bit 0:  0: Select SIMA Card1(default)  1: Select SIMA Card2  Bit 1:  0: Select SIMB Card1(default)  1: Select SIMB Card2  Bit 2:  0: Select SIMC Card1(default)  1: Select SIMC Card2  Bit 3:  0: Select SIMD Card1(default)  1: Select SIMD Card2  Bit 4:  0: Select SIME Card1(default)  1: Select SIME Card2  Bit 5~7: Reserved |

Example: Host set sim card select as below.

->0xF1 0x00 0x03 0x2D 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x2D Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x00 0x03 0x2D 0x01 CRC 0xF2

* + - 1. **Command 0x2F: SetGPOControl**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x2F Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| GPO Control | 1 | Bit 0:  0: GPO0 Pin State Low  1: GPO0 Pin State High  Bit 1:  0: GPO1 Pin State Low  1: GPO1 Pin State High  Bit 2:  0: GPO2 Pin State Low  1: GPO2 Pin State High  Bit 3:  0: GPO3 Pin State Low  1: GPO3 Pin State High  Bit 4~7: Reserved |

Example: Host set gpo control as below.

->0xF1 0x00 0x03 0x2F 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x2F Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x00 0x03 0x2F 0x01 CRC 0xF2

* + - 1. **Command 0x33: SetProgramLEDControl**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x33 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| LED Control | 1 | Bit 0~1: P1 LED  0: OFF(default)  ~~1: Orange ON~~  ~~2: Green ON~~  1: Red ON  2: Yellow ON  Bit 2~3: P2 LED  0: OFF(default)  ~~1: Orange ON~~  ~~2: Green ON~~  1: Red ON  2: Yellow ON  Bit 4~7: Reserved |

Example: Host set program led as below.

->0xF1 0x00 0x03 0x33 0x03 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x33 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x00 0x03 0x33 0x01 CRC 0xF2

* + - 1. **Command 0x36: SetWakeOnLanControl**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x36 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Wake On Lan Control | 1 | 0: Disable wake on lan(default)  1: Enable wake on lan |

Example: Host set wake on lan control as below.

->0xF1 0x00 0x03 0x36 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x36 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x00 0x03 0x36 0x01 CRC 0xF2

* + - 1. **~~Command 0x37: SetExteranl12V~~**

~~Direction: Host -> MCU~~

~~The package sent by host.~~

~~-> 0xF1 0x00 0x03 0x37 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~External 12V Control~~ | ~~1~~ | ~~0: power off~~  ~~1: power on~~ |

~~Example: Host set external 12v control as below.~~

~~->0xF1 0x00 0x03 0x37 0x01 CRC 0xF2~~

~~Direction: MCU -> Host~~

~~The response package received from mcu.~~

~~-> 0xF1 0x00 0x03 0x37 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Command Status~~ | ~~1~~ | ~~0x00: Setting Fail~~  ~~0x01: Setting Success~~ |

~~Example: MCU return setting result as below.~~

~~->0xF1 0x00 0x03 0x37 0x01 CRC 0xF2~~

* + - 1. **Command 0x3A: SetGPIActiveLevelControl**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x3A Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| GPI Level Control | 1 | 0: Active Low (Default), 1: Active High  Bit 0: GPI12  Bit 1: GPI34  Bit 2~7: Always 0. |

Example: Host set general purpose input active level control as below

-> 0xF1 0x00 0x03 0x3A 0x03 CRC 0xF2。

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x3A Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Set Ext. GPI level is Fail  0x01: Set Ext. GPI level is OK |

Example: Mcu Return package as below

-> 0xF1 0x00 0x03 0x3A 0x00 CRC 0xF2。

**2.2.5.26 Command 0x3B: SetGPOPullUpControl**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x3B Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| GPO Pull up Control | 1 | 0: Internal pull high (default), 1: External pull high  Bit 0: GPO12  Bit 1: GPO34  Bit 2~7: Always 0. |

Example: Host set general purpose output pull up control as below.

-> 0xF1 0x00 0x03 0x3B 0x03 CRC 0xF2。

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x3BData CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Set GPO pull up is Fail  0x01: Set GPO pull up is OK |

Example: Mcu Return package as below

-> 0xF1 0x00 0x03 0x3B 0x01 CRC 0xF2。

* + - 1. **Command 0x41: SetFANVALUE**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x00 0x03 0x41 Data Data1 Data2 CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | Comment |
| SYS\_FAN(Data) | 1 | Always 0. |
| POE\_FAN(Data1) | 1 | Range: 0~100  OFF: 0  ON: 1~100 |
| GRAPHICS\_FAN(Data2) | 1 | Range: 0~100  OFF: 0  ON: 1~100 |

Example: Host set main board version as below.

->0xF1 0x00 0x03 0x41 0x01 0x01 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x00 0x03 0x41 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x00 0x03 0x41 0x01 CRC 0xF2

* + 1. **Event Command List**
       1. **Command 0x03: Alarm**

Direction: MCU -> Host

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Scan code | 1 | Bit0: over voltage alarm  Bit1: lower voltage alarm  ~~Bit2: over temperature alarm~~  ~~Bit3: lower temperature alarm~~  Bit4~7: Reserved |

Example: MCU return event as below.

->0xF1 0x00 0x04 0x03 0x01 CRC 0xF2

**※請注意此Command為MCU主動往Host所以不須回應。**

**PS. 由於不修改MCU CODE因此高低溫event照發只是SDK**

**收到不予處理。**

* 1. **保留**
  2. **保留**
  3. **IAP Command Set**
     1. **ID: 0x03(IAP) Get Command Address**

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **名稱(Name)** | **Default** | **Address** |
| Get | Update Mode |  | 0x01 |
| Get | IAP Information |  | 0x02 |
| Get | APP Version |  | 0x03 |
| Get | Flash Data |  | 0x04 |
| Get | Program State | 0x00 | 0x05 |

* + 1. **ID: 0x03(IAP) Set Command Address**

|  |  |  |
| --- | --- | --- |
| **Type** | **名稱(Name)** | **Address** |
| Set | Update Request | 0x01 |
| Set | Flash Unlock | 0X02 |
| Set | Erase Flash | 0x03 |
| Set | Write Flash | 0x04 |
| Set | Check Code | 0x05 |
| Set | IAP Reboot | 0x06 |

* + 1. **Get Command List**
       1. **Command 0x01: GetUpdateMode**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x03 0x02 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x03 0x02 0x01 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Update Mode | 1 | 0x00: Application Mode  0x01: IAP Mode |

Example: MCU return current update mode as below.

->0xF1 0x03 0x02 0x01 0x01 CRC 0xF2

* + - 1. **Command 0x02: GetIAPInfomation**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x03 0x02 0x02 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x03 0x02 0x02 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Company Name | 8 | “NEXCOM” |
| Business Group | 8 | “MCS” |
| Product Name | 16 | “nROK5510” |
| MCU Type | 16 | “STM32F101VCT6” |
| Boot Loader Version | 15 | “VT551BXX” |
| Boot Method | 1 | 0x04 (New) |
| Nvram Block Start Address | 4 | 0x08008000 |
| Nvram Block Size(unit: byte) | 4 | 0x8000 |
| App Block Start Address | 4 | 0x08010000 |
| App Block Size(unit: byte) | 4 | 0x10000 |

Example: MCU return current iap information as below.

->0xF1 0x03 0x02 0x02 company business product mcu version

Method NVRAM start Addr NVRAM Size App Start Addr

App Size CRC 0xF2

* + - 1. **Command 0x03: GetAPPVersion**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x03 0x02 0x03 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x03 0x02 0x03 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| APP Version | 16 | Application Version |

Example: MCU return current app version as below.

->0xF1 0x03 0x02 0x03 version CRC 0xF2

* + - 1. **Command 0x04: GetFlashData**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x03 0x02 0x04 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Address | 4 | The start address will be read |
| Length | 4 | The length of data be read |

Example: Host set flash data as below.

->0xF1 0x03 0x02 0x04 address length CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x03 0x02 0x04 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Address | 4 | The start address be read |
| Flash | N | The data read from MCU flash |

Example: MCU return current flash data as below.

->0xF1 0x03 0x02 0x04 address flash CRC 0xF2

* + - 1. **Command 0x05: GetProgramState**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x03 0x02 0x05 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x03 0x02 0x05 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Program state | 1 | 0x00: AP have not been updated yet.  0x01: AP have been updated  0xFF: AP block data are mistake |

Example: MCU return current program state as below.

->0xF1 0x03 0x02 0x05 0x01 CRC 0xF2

* + 1. **Set Command List**
       1. **Command 0x01: SetUpdateRequest**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x03 0x03 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x03 0x03 0x01 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x03 0x03 0x01 0x01 CRC 0xF2

* + - 1. **Command 0x02: SetFlashUnLock**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x03 0x03 0x02 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Password | 6 | Fix data as below “NEXCOM” |

Example: Host set flash unlock as below.

->0xF1 0x03 0x03 0x02 password CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x03 0x03 0x02 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x03 0x03 0x02 0x01 CRC 0xF2

* + - 1. **Command 0x03: SetEraseFlash**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x03 0x03 0x03 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x03 0x03 0x03 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success  0x02: Erase Lock |

Example: MCU return command status as below.

->0xF1 0x03 0x03 0x03 0x01 CRC 0xF2

* + - 1. **Command 0x04: SetWriteFlash**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x03 0x03 0x04 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Address | 4 | The start address be write |
| Length | 4 | The length of data be read |
| Program data | N | The data write from host |

Example: Host set write flash as below.

->0xF1 0x03 0x03 0x04 address length program CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x03 0x03 0x04 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success  0x02: Write Lock  0x03: Data Over Range  0x04: Data Compare Fail |

Example: MCU return command status as below.

->0xF1 0x03 0x03 0x04 0x01 CRC 0xF2

* + - 1. **Command 0x05: SetCheckCode**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x03 0x03 0x05 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Check Code | 16 | 0x55 0xAA 0x5A 0xA5  0x13 0x57 0x24 0x68  0x03 0xFC 0x30 0xCF  0x00 0xFF 0x0F 0xF0 |

Example: Host set check code as below.

->0xF1 0x03 0x03 0x05 checkcode CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x03 0x03 0x05 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x03 0x03 0x04 0x01 CRC 0xF2

* + - 1. **Command 0x06: SetIAPReboot**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x03 0x03 0x06 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x03 0x03 0x06 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x03 0x03 0x06 0x01 CRC 0xF2

* + 1. **MCU Update Flash Map**



* + 1. **MCU Update Control Flow**



* 1. **除錯指令**
  2. **BIOS指令**
     1. **ID:0x05(BIOS) Get Command Address**

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **名稱(Name)** | **Default** | **Address** |
| Get | MCU Version | 0x01 | 0x01 |
| ~~Get~~ | ~~Power Mode~~ | ~~0x00~~ | ~~0X02~~ |
| Get | Delay Mode | 0x00 | 0x03 |
| ~~Get~~ | ~~Power Range~~ | ~~0x00~~ | ~~0x04~~ |
| Get | Module Control | 0x03 | 0x05 |
| Get | Alarm Hour | 0x00 | 0x06 |
| Get | Alarm Min | 0x00 | 0x07 |
| Get | Alarm Sec | 0x00 | 0x08 |
| ~~Get~~ | ~~Brightness~~ | ~~0x08~~ | ~~0x09~~ |
| Get | RTC Year | 0x00 | 0x0A |
| Get | RTC Month | 0x00 | 0x0B |
| Get | RTC Day | 0x00 | 0x0C |
| Get | RTC Weekday | 0x00 | 0x0D |
| Get | Update Status | 0x00 | 0x13 |
| Get | OS Status | 0x00 | 0x14 |
| Get | Auto Reboot Function | 0x00 | 0x16 |
| Get | MB Version | 0x00 | 0x17 |
| Get | Watchdog Control | 0x00 | 0x20 |
| Get | Watchdog Time | 0x03 | 0x21 |
| Get | Poe Fan Value | 0x64 | 0x2D |
| Get | iAMT Low Power Wake Up | 0x00 | 0x2E |
| Get | Graphics Fan Value | 0x64 | 0x30 |
| Get | PCIE Slot Mode | 0x00 | 0x31 |

* + 1. **ID:0x05(BIOS) Set Command Address**

|  |  |  |
| --- | --- | --- |
| **Type** | **名稱(Name)** | **Address** |
| ~~Set~~ | ~~Power Mode~~ | ~~0x02~~ |
| Set | Delay Mode | 0x03 |
| ~~Set~~ | ~~Power Range~~ | ~~0x04~~ |
| Set | Module Control | 0x05 |
| Set | Alarm Hour | 0x06 |
| Set | Alarm Min | 0x07 |
| Set | Alarm Sec | 0x08 |
| ~~Set~~ | ~~Brightness~~ | ~~0x09~~ |
| Set | RTC Year | 0x0A |
| Set | RTC Month | 0x0B |
| Set | RTC Day | 0x0C |
| Set | RTC Weekday | 0x0D |
| Set | RTC Hour | 0x10 |
| Set | RTC Min | 0x11 |
| Set | RTC Sec | 0x12 |
| Set | Update Mode | 0x13 |
| Set | OS Mode | 0x14 |
| Set | Auto Reboot Function | 0x16 |
| Set | MB Version | 0x17 |
| Set | Watchdog Control | 0x20 |
| Set | Watchdog Time | 0x21 |
| Set | Poe Fan Value | 0x2D |
| Set | iAMT Low Power Wake Up | 0x2E |
| Set | Graphics Fan Value | 0x30 |
| Set | PCIE Slot Control | 0x31 |

* + 1. **Read Command List**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x00 Address Length CRC 0xF2

Example: Host read BIOS parameter from address 0x02 ~ 0x0C as below.

->0xF1 0x05 0x00 0x02 0x0C CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x00 Address Length Data0 … DataN CRC 0xF2

Example: MCU return BIOS parameter from address 0x01 ~ 0x03 as below.

->0xF1 0x05 0x00 0x02 0x0C 0x00 0x00 0x00 CRC 0xF2

* + 1. **Write Command List**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x01 Address Length Data0 … DataN CRC 0xF2

Example: Host write BIOS parameter from address 0x02 ~ 0x04 as below.

->0xF1 0x05 0x01 0x02 0x03 0x00 0x00 0x00 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x01 Address data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Write Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU response write status as below.

->0xF1 0x05 0x01 0x02 0x01 CRC 0xF2

* + 1. **Get Command List**
       1. **Command 0x01: GetMCUVersion**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x01 Data CRC? 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| MCU Version | 1 | 0x01~0x0FF |

Example: MCU return current MCU version as below.

->0xF1 0x05 0x02 0x01 0x01 CRC 0xF2

* + - 1. **~~Command 0x02: GetPowerMode~~**

~~Direction: Host -> MCU~~

~~The package sent by host.~~

~~-> 0xF1 0x05 0x02 0x02 CRC 0xF2~~

~~Direction: MCU -> Host~~

~~The response package received from mcu.~~

~~-> 0xF1 0x05 0x02 0x02 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Power Type~~ | ~~1~~ | ~~0x03: 12V~~  ~~0x02: 24V~~  ~~0x01: Reserved(9~60)~~  ~~0x00: 9~36V(default)~~ |

~~Example: MCU return current power mode as below.~~

~~->0xF1 0x05 0x02 0x02 0x03 CRC 0xF2~~

* + - 1. **Command 0x03: GetDelayMode**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x03 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x03 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Delay Time | 1 | Bit 0~2: Power off  000: 20 Sec  001: 1 Min  010: 5 Mins  011: 10 Mins  100: 30 Mins  101: 1 Hour  110: 6 Hours  111: 18 Hours  Bit 3~5: Power on  000: 10 Sec  001: 30 Sec  010: 1Min  011: 5 Mins  100: 10 Mins  101: 15 Mins  110: 30 Mins  111: 1 Hour  Bit 6: Delay off control  0: Disable, 1: Enable (default Disable)  Bit7: Delay on control  0: Disable, 1: Enable(default Disable) |

Example: MCU return current delay time as below.

->0xF1 0x05 0x02 0x03 0x41 CRC 0xF2

* + - 1. **~~Command 0x04: GetPowerRange~~**

~~Direction: Host -> MCU~~

~~The package sent by host.~~

~~-> 0xF1 0x05 0x02 0x04 CRC 0xF2~~

~~Direction: MCU -> Host~~

~~The response package received from mcu.~~

~~-> 0xF1 0x05 0x02 0x04 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Startup / Shutdown~~ | ~~1~~ | ~~12V / 24V~~  ~~Startup/Shutdown Startup/Shutdown~~  ~~0x00: 11.5V 10.5V 23.0V 21.0V (default)~~  ~~0x01: 12.0V 11.0V 24.0V 22.0V~~  ~~0x02: 12.5V 11.0V 25.0V 22.0V~~  ~~0x03: 12.5V 11.5V 25.0V 23.0V~~ |

~~Example: MCU return current power rnage as below.~~

~~->0xF1 0x05 0x02 0x04 0x03 CRC 0xF2~~

* + - 1. **Command 0x05: GetModuleControl**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x05 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x05 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Module Control | 1 | Bit 0: (CN3)  0: Disable WWAND wakeup mode(default)  1: Enable WWAND wakeup mode  Bit 1:  0: Disable WWAND  1: Enable WWAND(default)  Bit 2:  0: Disable RTC wakeup(default)  1: Enable RTC wakeup  Bit 3:  0: Disable wake on lan(default)  1: Enable wake on lan  Bit 4: 0 (CN7)  0: Disable WWANA  1: Enable WWANA(default)  Bit 5: (CN6)  0: Disable WWANB  1: Enable WWANB(default)  Bit 6: (CN5)  0: Disable WWANC  1: Enable WWANC(default)  Bit 7: (CN20)  0: Disable WWANE  1: Enable WWANE(default) |

Example: MCU return current module control as below.

->0xF1 0x05 0x02 0x05 0xF2 CRC 0xF2

* + - 1. **Command 0x06: GetAlarmHour**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x06 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x06 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Alarm Hour | 1 | 0 ~ 23 |

Example: MCU return current alarm hour as below.

->0xF1 0x05 0x02 0x06 0x03 CRC 0xF2

* + - 1. **Command 0x07: GetAlarmMin**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x07 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x07 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Alarm Min | 1 | 0 ~ 59 |

Example: MCU return current alarm min as below.

->0xF1 0x05 0x02 0x07 0x03 CRC 0xF2

* + - 1. **Command 0x08: GetAlarmSec**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x08 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x08 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Alarm Sec | 1 | 0 ~ 59 |

Example: MCU return current alarm sec as below.

->0xF1 0x05 0x02 0x08 0x03 CRC 0xF2

* + - 1. **~~Command 0x09: GetBrightness~~**

~~Direction: Host -> MCU~~

~~The package sent by host.~~

~~-> 0xF1 0x05 0x02 0x09 CRC 0xF2~~

~~Direction: MCU -> Host~~

~~The response package received from mcu.~~

~~-> 0xF1 0x05 0x02 0x09 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Brightness~~ | ~~1~~ | ~~0x01 ~ 0x0A(default:8)~~ |

~~Example: MCU return current brightness as below.~~

~~->0xF1 0x05 0x02 0x09 0x03 CRC 0xF2~~

* + - 1. **Command 0x0A: GetRTCYear**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x0A CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x0A Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| RTC Year | 1 | 0 ~ 99 |

Example: MCU return current RTC year as below.

->0xF1 0x05 0x02 0x0A 0x03 CRC 0xF2

* + - 1. **Command 0x0B: GetRTCMonth**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x0B CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x0B Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| RTC Month | 1 | 1 ~ 12 |

Example: MCU return current RTC month as below.

->0xF1 0x05 0x02 0x0B 0x03 CRC 0xF2

* + - 1. **Command 0x0C: GetRTCDay**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x0C CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x0C Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| RTC day | 1 | 1 ~ 31 |

Example: MCU return current RTC day as below.

->0xF1 0x05 0x02 0x0C 0x03 CRC 0xF2

* + - 1. **Command 0x0D: GetRTCWeekday**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x0D CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x0D Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| RTC weekday | 1 | 0x01: Monday 0x02: Tuesday  0x03: Wednesday 0x04: Thursday  0x05: Friday 0x06: Saturday  0x07: Sunday |

Example: MCU return current RTC weekday as below.

->0xF1 0x05 0x02 0x0D 0x03 CRC 0xF2

* + - 1. **Command 0x13: GetUpdateStatus**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x13 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x13 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Update Mode | 1 | 0x00: MCU store data ok.  0x01: MCU is storing data now. |

Example: MCU return current update status as below.

->0xF1 0x05 0x02 0x13 0x01 CRC 0xF2

* + - 1. **Command 0x14: GetOSStatus**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x14 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x14 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| OS Mode | 1 | 0x00: none  0x01: BIOS  0x02: OS |

Example: MCU return current OS status as below.

->0xF1 0x05 0x02 0x14 0x01 CRC 0xF2

* + - 1. **Command 0x16: GetAutoRebootFunction**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x16 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x16 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Auto Reboot Function | 1 | Let OS use software shutdown when ignition status is on, VTC5510 will auto reboot or not.  0x00: Disable  0x01: Enable |

Example: MCU return current auto reboot function as below.

->0xF1 0x05 0x02 0x16 0x01 CRC 0xF2

* + - 1. **Command 0x17: GetMBVersion**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x17 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x17 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| MB Version | 1 | 0x00 ~ 0xFF |

Example: MCU return current main board version as below.

->0xF1 0x05 0x02 0x17 0x01 CRC 0xF2

**2.7.5.18 Command 0x20: GetWatchdogControl**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x20 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x20 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Control | 1 | Bit 6~0: always 0  Bit 7: WDT enable  0: Disable(default)  1: Enable |

Example: MCU return setting result as below.

->0xF1 0x05 0x02 0x20 0x80 CRC 0xF2

**2.7.5.19 Command 0x21: GetWatchdogTimer**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x21 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x21 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Timeout | 1 | 3 ~ 255 (Sec) |

Example: MCU return setting result as below.

->0xF1 0x05 0x02 0x21 0x0A CRC 0xF2

* + - 1. **Command 0x2D: GetPoeFanValue**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x2D CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x2D Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| POE Fan Value | 1 | Range: 0~100  OFF: 0  ON: 1~100 |

Example: Host set main board version as below.

->0xF1 0x05 0x02 0x2D 0x01 CRC 0xF2

* + - 1. **Command 0x2E:** **Get\_iAMT\_Low\_Power\_Wake\_Up**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x2E CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x2E Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| iAMT Low Power Wake Up | 1 | Off: 0 (default)  On: 1 |

Example: Host set main board version as below.

->0xF1 0x05 0x02 0x2E 0x01 CRC 0xF2

* + - 1. **Command 0x30: GetGraphicsFanValue**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x30 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x30 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Graphics Fan Value | 1 | Range: 0~100  OFF: 0  ON: 1~100 |

Example: Host set main board version as below.

->0xF1 0x05 0x02 0x30 0x01 CRC 0xF2

* + - 1. **Command 0x31: GetPCIESlotMode**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x02 0x31 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x02 0x31 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| PCIE Slot Mode | 1 | Bit 0: Slot1 (CN7)  0: WWANA(always 0)  1: N/A  Bit 1: Slot2 (CN6)  0: WWANB(always 0)  1: N/A  Bit 2: Slot3 (CN5)  0: WWANC(always 0)  1: N/A  Bit 3: Slot4 (CN3)  0: WWAND(always 0)  1: N/A  Bit 4: Slot5 (CN20)  0: WWANE  1: WIFIA(default)  Bit5~7:0(Reserved) |

Example: MCU return current PCIE slot mode as below.

->0xF1 0x05 0x02 0x31 0x00 CRC 0xF2

* + 1. **Set Command List**
       1. **~~Command 0x02: SetPowerMode~~**

~~Direction: Host -> MCU~~

~~The package sent by host.~~

~~-> 0xF1 0x05 0x03 0x02 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Power Type~~ | ~~1~~ | ~~0x05: 36V~~  ~~0x04: 48V~~  ~~0x03: 12V~~  ~~0x02: 24V~~  ~~0x01: Reserved(9~60)~~  ~~0x00: 9~60V(default)~~ |

~~Example: Host set power mode as below.~~

~~->0xF1 0x05 0x03 0x02 0x00 CRC 0xF2~~

~~Direction: MCU -> Host~~

~~The response package received from mcu.~~

~~-> 0xF1 0x05 0x03 0x02 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Command Status~~ | ~~1~~ | ~~0x00: Setting Fail~~  ~~0x01: Setting Success~~ |

~~Example: MCU return command status as below.~~

~~->0xF1 0x05 0x03 0x02 0x01 CRC 0xF2~~

* + - 1. **Command 0x03: SetDelayMode**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x03 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Delay Time | 1 | Bit 0~2: Power off  000: 20 Sec  001: 1 Min  010: 5 Mins  011: 10 Mins  100: 30 Mins  101: 1 Hour  110: 6 Hours  111: 18 Hours  Bit 3~5: Power on  000: 10 Sec  001: 30 Sec  010: 1Min  011: 5 Mins  100: 10 Mins  101: 15 Mins  110: 30 Mins  111: 1 Hour  Bit 6: Delay off control  0: Disable, 1: Enable (default Disable)  Bit7: Delay on control  0: Disable, 1: Enable(default Disable) |

Example: Host set delay time as below.

->0xF1 0x05 0x03 0x03 0x41 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x03 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x05 0x03 0x03 0x01 CRC 0xF2

* + - 1. **~~Command 0x04: SetPowerRange~~**

~~Direction: Host -> MCU~~

~~The package sent by host.~~

~~-> 0xF1 0x05 0x03 0x04 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Startup / Shutdown~~ | ~~1~~ | ~~12V / 24V~~  ~~Startup/Shutdown Startup/Shutdown~~  ~~0x00: 11.5V 10.5V 23.0V 21.0V (default)~~  ~~0x01: 12.0V 11.0V 24.0V 22.0V~~  ~~0x02: 12.5V 11.0V 25.0V 22.0V~~  ~~0x03: 12.5V 11.5V 25.0V 23.0V~~ |

~~Example: Host set current power range as below.~~

~~->0xF1 0x05 0x03 0x04 0x03 CRC 0xF2~~

~~Direction: MCU -> Host~~

~~The response package received from mcu.~~

~~-> 0xF1 0x05 0x03 0x04 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Command Status~~ | ~~1~~ | ~~0x00: Setting Fail~~  ~~0x01: Setting Success~~ |

~~Example: MCU return command status as below.~~

~~->0xF1 0x05 0x03 0x04 0x01 CRC 0xF2~~

* + - 1. **Command 0x05: SetModuleControl**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x05 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Module Control | 1 | Bit 0: (CN3)  0: Disable WWAND wakeup mode(default)  1: Enable WWAND wakeup mode  Bit 1: (CN3)  0: Disable WWAND  1: Enable WWAND(default)  Bit 2:  0: Disable RTC wakeup(default)  1: Enable RTC wakeup  Bit 3:  0: Disable wake on lan(default)  1: Enable wake on lan  Bit 4: (CN7)  0: Disable WWANA  1: Enable WWANA(default)  Bit 5: (CN6)  0: Disable WWANB  1: Enable WWANB(default)  Bit 6: (CN5)  0: Disable WWANC  1: Enable WWANC(default)  Bit 7: (CN20)  0: Disable WWANE  1: Enable WWANE(default) |

Example: Host set module control as below.

->0xF1 0x05 0x03 0x05 0x03 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x05 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x05 0x03 0x05 0x01 CRC 0xF2

* + - 1. **Command 0x06: SetAlarmHour**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x06 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Alarm Hour | 1 | 0 ~ 23 |

Example: Host set alarm hour as below.

->0xF1 0x05 0x03 0x06 0x03 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x06 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x05 0x03 0x06 0x01 CRC 0xF2

* + - 1. **Command 0x07: SetAlarmMin**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x07 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Alarm Min | 1 | 0 ~ 59 |

Example: Host set alarm min as below.

->0xF1 0x05 0x03 0x07 0x03 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x07 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x05 0x03 0x07 0x01 CRC 0xF2

* + - 1. **Command 0x08: SetAlarmSec**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x08 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Alarm Sec | 1 | 0 ~ 59 |

Example: Host set alarm sec as below.

->0xF1 0x05 0x03 0x08 0x03 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x08 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x05 0x03 0x08 0x01 CRC 0xF2

* + - 1. **~~Command 0x09: SetBrightness~~**

~~Direction: Host -> MCU~~

~~The package sent by host.~~

~~-> 0xF1 0x05 0x03 0x09 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Brightness~~ | ~~1~~ | ~~0x01 ~ 0x0A~~ |

~~Example: Host set brightness as below.~~

~~->0xF1 0x05 0x03 0x09 0x03 CRC 0xF2~~

~~Direction: MCU -> Host~~

~~The response package received from mcu.~~

~~-> 0xF1 0x05 0x03 0x09 Data CRC 0xF2~~

|  |  |  |
| --- | --- | --- |
| **~~Data Structure~~** | **~~Length~~** | **~~Comment~~** |
| ~~Command Status~~ | ~~1~~ | ~~0x00: Setting Fail~~  ~~0x01: Setting Success~~ |

~~Example: MCU return command status as below.~~

~~->0xF1 0x05 0x03 0x09 0x01 CRC 0xF2~~

* + - 1. **Command 0x0A: SetRTCYear**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x0A Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| RTC Year | 1 | 0 ~ 99 |

Example: Host set RTC year as below.

->0xF1 0x05 0x03 0x0A 0x03 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x0A Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x05 0x03 0x0A 0x01 CRC 0xF2

* + - 1. **Command 0x0B: SetRTCMonth**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x0B Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| RTC Month | 1 | 1 ~ 12 |

Example: Host set RTC month as below.

->0xF1 0x05 0x03 0x0B 0x03 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x0B Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x05 0x03 0x0B 0x01 CRC 0xF2

* + - 1. **Command 0x0C: SetRTCDay**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x0C Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| RTC day | 1 | 1 ~ 31 |

Example: Host set RTC day as below.

->0xF1 0x05 0x03 0x0C 0x03 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x0C Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x05 0x03 0x0C 0x01 CRC 0xF2

* + - 1. **Command 0x0D: SetRTCWeekday**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x0D Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| RTC weekday | 1 | 0x01: Monday 0x02: Tuesday  0x03: Wednesday 0x04: Thursday  0x05: Friday 0x06: Saturday  0x07: Sunday |

Example: Host set RTC weekday as below.

->0xF1 0x05 0x03 0x0D 0x03 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x0D Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x05 0x03 0x0D 0x01 CRC 0xF2

* + - 1. **Command 0x13: SetUpdateMode**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x13 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Update Mode | 1 | 0x00: MCU store data ok.  0x01: MCU is storing data now. |

Example: Host set update mode as below.

->0xF1 0x05 0x03 0x13 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x13 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x05 0x03 0x13 0x01 CRC 0xF2

* + - 1. **Command 0x14: SetOSMode**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x14 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| OS Mode | 1 | 0x00: none  0x01: BIOS  0x02: OS |

Example: Host set OS Mode as below.

->0xF1 0x05 0x03 0x14 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x14 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x05 0x03 0x14 0x01 CRC 0xF2

* + - 1. **Command 0x16: SetAutoRebootFunction**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x16 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Auto Reboot Function | 1 | Let OS use software shutdown when ignition status is on, nROK5510 will auto reboot or not.  0x00: Disable  0x01: Enable |

Example: Host set auto reboot function as below.

->0xF1 0x05 0x03 0x16 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x16 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x05 0x03 0x16 0x01 CRC 0xF2

* + - 1. **Command 0x17: SetMBVersion**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x17 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| MB Version | 1 | 0x00 ~ 0xFF |

Example: Host set main board version as below.

->0xF1 0x05 0x03 0x17 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x17 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x05 0x03 0x17 0x01 CRC 0xF2

**2.7.6.17 Command 0x20: SetWatchdogControl**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x20 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Control | 1 | Bit 6~0: always 0  Bit 7:  0: Disable(default)  1: Enable |

Example: Host set watchdog control as below.

->0xF1 0x05 0x03 0x20 0x80 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x20 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x05 0x03 0x20 0x01 CRC 0xF2

* + - 1. **Command 0x21: SetWatchdogTimer**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x21 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Timeout | 1 | 3 ~ 255 (Sec) |

Example: Host set watchdog timeout as below.

->0xF1 0x05 0x03 0x21 0x03 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x21 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x05 0x03 0x21 0x01 CRC 0xF2

* + - 1. **Command 0x2D: SetPoeFanValue**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x2D Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | Comment |
| Poe Fan Value | 1 | Range: 0~100  OFF: 0  ON: 1~100 |

Example: Host set poe fan value as below.

->0xF1 0x05 0x03 0x2D 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x2D Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x05 0x03 0x2D 0x01 CRC 0xF2

* + - 1. **Command 0x2E: Set\_iAMT\_Low\_Power\_Wake\_Up**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x2E Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | Comment |
| iAMT Low Power Wake Up | 1 | Off: 0 (default)  when use this item:  1 .the power status in s5, the stand by power turn off  2. when Ignition off , the stand by power is off  On: 1  when use this item:  1 .the power status in s5, the stand by power retain on  2. when Ignition off , the stand by power is on |

Example: Host set iAMT low power Wake-Up as below.

->0xF1 0x05 0x03 0x2E 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x2E Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x00 0x03 0x41 0x01 CRC 0xF2

* + - 1. **Command 0x30: SetGraphicsFanValue**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x30 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | Comment |
| Graphics Fan Value | 1 | Range: 0~100  OFF: 0  ON: 1~100 |

Example: Host set graphics fan value as below.

->0xF1 0x05 0x03 0x30 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x30 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return command status as below.

->0xF1 0x05 0x03 0x30 0x01 CRC 0xF2

* + - 1. **Command 0x31: SetPCIESlotControl**

Direction: Host -> MCU

The package sent by host.

-> 0xF1 0x05 0x03 0x31 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| PCIE Slot Control | 1 | Bit 0: Slot1 (CN7)  0: WWANA(always 0)  1: N/A  Bit 1: Slot2 (CN6)  0: WWANB(always 0)  1: N/A  Bit 2: Slot3 (CN5)  0: WWANC(always 0)  1: N/A  Bit 3: Slot4 (CN3)  0: WWAND(always 0)  1: N/A  Bit 4: Slot5 (CN20)  0: WWANE  1: WIFIA(default)  Bit5~7:0(Reserved) |

Example: Host set PCIE slot control as below.

->0xF1 0x05 0x03 0x31 0x01 CRC 0xF2

Direction: MCU -> Host

The response package received from mcu.

-> 0xF1 0x05 0x03 0x31 Data CRC 0xF2

|  |  |  |
| --- | --- | --- |
| **Data Structure** | **Length** | **Comment** |
| Command Status | 1 | 0x00: Setting Fail  0x01: Setting Success |

Example: MCU return setting result as below.

->0xF1 0x05 0x03 0x31 0x01 CRC 0xF2

* 1. **CAN Command Set**
  2. **IOT Command Set**
  3. **BKP Command Set**
  4. **POE Command Set**
  5. **VTK62B Command Set**
  6. **Error CommandSet**

To use it as an error command package is received. When the Host receives this

response, it needs to resend the command with the previous error to let processes

continue.

* + 1. **ID:0xFF(Error) Error Command Response List**

|  |  |
| --- | --- |
| **錯誤碼(Error)** | |
| Index | Content |
| 0x00 | Device |
| 0x01 | Command |
| 0x02 | Address |
| 0x03 | Data |
| 0x04 | Length |
| 0x05 | Type |
| 0x06 | Other |

**2.13.2 ID:0xFF(Error) Error Command Package**

* **Error Response (Mcu >> Host)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Start Code | Device ID | Error Code | CRC | End code |
| 0xF1 | 0xFF | 0x00~0x06 | CRC8 | 0xF2 |