This document outlines the communication protocol between the TDU and the Fleet Tracker application that resides in AWS. The Asset Tracker passes the data between the TDU and Fleet Tracker using 26 byte data packets. The data Asset tracker uses only 2 types of commands, Maintenance and Telemetry binary data strings. The command format below is also used via the Asset Tracker configuration program to accomplish OTA firmware upgrades for the TDU.

**Command Format**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Length**  **byte1** | **Length**  **byte2** | **Command** | **Parameter** | **Data byte(s)** | **Checksum** |

Length byte1: The overall command length % 256

Length byte2: The overall command length / 256 + 1

*\* note: this byte is offset by 1 because the Asset Tracker cannot accept a 0*

Command byte: Defines the type of command – SET, GET, STATUS, …

Parameter byte: Defines the data type

Data byte(s): Start of the optional command data

Checksum byte: Byte wise addition of previous command bytes

**Overall Command Length**

The overall command length is calculated by the 2 length bytes at the beginning of a command

The overall command length = length byte 1 + (length byte 2 – 1) \* 256

**Commands**

The command byte defines the type of operation being performed

#define COMMAND\_GET 1

#define COMMAND\_SET 2

#define COMMAND\_HISTORY 3

#define COMMAND\_SOFTWATE\_UPDATE 4

#define COMMAND\_STATUS 5

#define COMMAND\_DATA 6

**Parameters**

The parameter byte specifies the data type

#define PARAM\_COMPASS\_BEARING 1

#define PARAM\_COMPASS\_ORIENTATION 2

#define PARAM\_BATTERY\_VOLTAGE 3

#define PARAM\_BATTERY\_CURRENT 4

#define PARAM\_SOLAR\_VOLTAGE 5

#define PARAM\_SOLAR\_CURRENT 6

#define PARAM\_CHARGING\_INDICATOR 7

#define PARAM\_PATTERN\_SELECTION 8

#define PARAM\_PHOTO\_CELL\_READING 9

#define PARAM\_ALARM\_BITMASK 10

#define PARAM\_HARDWARE\_VERSION 11

#define PARAM\_SOFTWARE\_VERSION 12

#define PARAM\_GET\_MODEL\_TYPE 13

#define PARAM\_GET\_DATE\_TIME 14

#define PARAM\_GET\_TELEMETRY\_DATA 15

#define PARAM\_GET\_MAINTENANCE\_DATA 16

#define PARAM\_START\_UPGRADE 17

#define PARAM\_ERASE\_UPGRADE\_SECTORS 18

#define PARAM\_WRITE\_BLOCK 19

#define PARAM\_END\_UPGRADE 20

**Data**

The data byte(s) are optional and can be of variable number of bytes

**Checksum**

The checksum byte is a byte-wise addition of all of the previous command bytes

**Command synchronization**

Any byte received after a gap that exceeds 50 milliseconds is considered to be the start of a new command sequence

**Status Return Values**

GET commands sent to the device will return with the data specified by the command parameter. Other commands, such as SET, will respond with a STATUS command to acknowledge the received command.

#define STATUS\_OKAY 0

#define STATUS\_ERROR\_CHECKSUM 1

#define STATUS\_ERROR\_UNKNOWN\_CMD 2

#define STATUS\_ERROR\_UNKOWN\_PARAM 3

#define STATUS\_ERROR\_ILLEGAL\_SETTING 4

#define STATUS\_ERROR\_SECTOR\_NOT\_BLANK 5

#define STATUS\_ERROR\_WRITE\_BLOCK 6

#define STATUS\_CODE\_CS\_ERROR 7

#define STATUS\_ERROR\_WRITE\_FLAG 8

#define STATUS\_ERROR\_PRODUCT\_ID 9

**Examples**

Example: Set (TDU) arrow pattern

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Length1** | **Length2** | **Command** | **Parameter** | **Data** | **Checksum** |
| **6** | **1** | **2** | **8** | **7** | **22** |

Overall command length = 6 + (1-1) \* 256

Command: COMMAND\_SET

Parameter: PARAM\_PATTERN\_SELECTION

Data: eDISPLAY\_TYPE\_LEFT\_STEM\_ARROW 2

Response

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Length1 | Length2 | Command | Parameter | Checksum |
| 4 | 1 | 5 | 0 | 10 |

Overall command length = 4 + (1-1) \* 256

Command: COMMAND\_STATUS

Parameter: STATUS\_OKAY

*TDU acknowledges SET command, the arrow patter has been set to left stem arrow*

Example: Request command battery current

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Length1** | **Length2** | **Command** | **Parameter** | **Checksum** |
| **1** | **1** | **1** | **4** | **10** |

overall command length = 4 + 256 (1-1)

command: CMD\_GET

parameter: PARAM\_BATTERY\_CURRENT

Response

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Length1** | **Length2** | **Command** | **Parameter** | **Data1** | **Data2** | **Checksum** |
| **6** | **1** | **6** | **4** | **0** | **60** | **77** |

overall command length = 6 + (1-1) \* 256

command: CMD\_GET

parameter: PARAM\_BATTERY\_CURRENT

data = (data1 << 8) + data2 = 60 (current = 60/100 = .6A)

Example: Request (TDU) arrow pattern

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Length1** | **Length2** | **Command** | **Parameter** | **Checksum** |
| **4** | **1** | **1** | **8** | **14** |

Overall command length = 4 + (1-1) \* 256

Command: CMD\_GET

Parameter: PARAM\_PATTERN\_SELECTION

Response

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Length1** | **Length2** | **Command** | **Parameter** | **Data** | **Checksum** |
| **5** | **1** | **1** | **8** | **11** | **26** |

**Asset Tracker Interface**

The Asset Tracker interface is ASCII based and has a limit of 26 characters. The Asset Tracker does not parse any of the data, it is directly passed up to the FM Tracker application. The FM Tracker application parses the data and converts it to a JSON format, which is then passed on to Fleet Manager.

There are 2 types of data sets, Telemetry data and Maintenance data. Both types of data sets use the command structure described above.

Example formats of TDU to Asset Tracker

Telemetry data – prefaced with “D11”

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Byte | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|  | ‘D’ | ‘1’ | ‘1’ | D1 | D2 | D3 | D4 | D5 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| D6 | D7 | D8 | D9 | D10 | D11 | D12 | D13 | D14 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 17 | 18 | 19 | 20 | 21 | 22 | 23 | 25 | 25 |
| D15 | D16 | D17 | D18 | D19 | D20 | D21 | D22 | D23 |

D1-D2 battery voltage

D3-D4 battery current

D5-D6 compass bearing

D7 compass orientation x-axis

D8 compass orientation y-axis

D9 compass orientation z-axis

D10-D11 alarm bitmask

D12 averaged photocell brightness

D13 arrow board pattern

D14-D15 solar voltage

D16-D17 solar current

D18 3 position switch ON/OFF/REMOTE

D19 temperature (on board accelerometer)

D20 modem token count

D21 averaged photocell reading

D22 solar charge OFF/ON

D23 checksum

Maintenance Data – prefaced with “D12”

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Byte | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|  | ‘D’ | ‘1’ | ‘2’ | D1 | D2 | D3 | D4 | 0 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | D5 |

D1-D2 software version

D3 hardware version

D4 fleet manager model

D5 checksum

**Rate Limiter**

A rate limiter has been implemented to prevent over usage of cellular data. The TDU uses a token scheme where a token is used to limit the amount of telemetry data that is sent up to Fleet Manager. The TDU is initially given token count of 60 and a token is added every 60 seconds. The TDU will be prevented from sending any data once the tokens are exhausted. It will then resume after 5 minutes, when 5 tokes have been accumulated.

The maintenance data can be requested by fleet manager but generally is only sent 30 seconds after the TDU powers up.

The telemetry data is sent without regard to the token count:

1 On request by Fleet Manager

2 After a pattern selection change (delayed 5 seconds)

3 On/Off/Remote switch change (delayed 30 seconds)

4 Alarm LVD active or Alarm Data Rate (activated or cleared)

The telemetry data is sent under token control:

1 Alarm Lamps Disabled

2 Alarm Board Tipped

3 Alarm Error Magnetometer

4 Alarm Remote Control

5 Alarm Board Stowed