UPMSAT-2

Public Telemetry Decoding

Jorge Garrido Balaguer

Version 1.3
07 Sep 2020

UNIVERSIDAD POLITÉCNICA DE MADRID

GRUPO DE SISTEMAS DE TIEMPO REAL
Y ARQUITECTURA DE SERVICIOS TELEMÁTICOS

Copyright © DIT/UPM 2019, 2020. Some rights reserved.

This work is licensed under a *Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported* license. See http://creativecommons.org/licenses/by-nc-nd/3.0/.

Status: Public

Written by: Jorge Garrido Balaguer

Reviewed by:

Modificaciones

Version	Date	Purpose	Author
1.0	17-07-2019	Initial Version.	J. Garrido
1.1	22-07-2019	Minor text revision and updated conversion equations.	J. Garrido
1.2	23-07-2019	Added operating mode conversion table.	J. Garrido
1.3	07-09-2020	Fixed analog data table thanks to M.Rupprecht and D. Estevez.	J. Garrido

Groups and organisations taking part in the UPMSat2 project

IDR	Instituto Universitario de Microgravedad “Ignacio da Riva” (UPM)
STRAST	Sistemas de Tiempo Real y Arquitectura de Servicios Telemáticos (UPM)

1 Introduction

In this document the required information to receive the public Telemetry (TM) messages sent by the UPMSat-2 mission flight segment is presented. In Section 2 the necessary radio frequency details are provided. In Section 3 a brief manual of the software provided for TM decodification is included, and finally the detailed frame structure and other required information for building you own decoding software is presented in Section 4 .

2 General structure of the messages

UPMSat-2 public telemetry messages are sent in AX.25 Unnumbered Information Frames at 437.403.500 Hz at 1200 bauds, USB mode. The satellite callsign is UPMST2. General structure of frames is shown in Figure 1.

Flag	AX.25 Frame Header (184 bits)					Information Field	Frame-Check Sequence	Flag
	Destination Address	Source Address	Repeater Address	Control Bits	Protocol Identifier			
8	56	56	56	8	8	0-1840	16	8

Figure 1: UI Frame.

- Flag field: constant value: 0x7E.
- Addresses: flight segment callsign is UPMST2.
- Control bits: constant value 0x03.
- Protocol identifier: constant value 0xF0.
- Information Field: application data carried by the frame. UPMSat-2 public TM always carry 102 bytes.
- Frame Check Sequence: cyclic redundancy code.

3 Using pre-built application

Public TM messages can be decoded by the application provided by UPM team. This software shows in a terminal window the last received telemetry message information, while a log of received messages is stored in a .csv file for convenient post-processing at user will.

The application provided connects with a AGWPE server application using port 8000 (server application should be already running and accepting connections when starting the upmsat2 executable). Soundmodem by UZ7HO[1] has been used for developing and

[1] http://uz7.ho.ua/packetradio.htm

debugging procedures, and is known to work properly. Other similar applications can be used at user will.

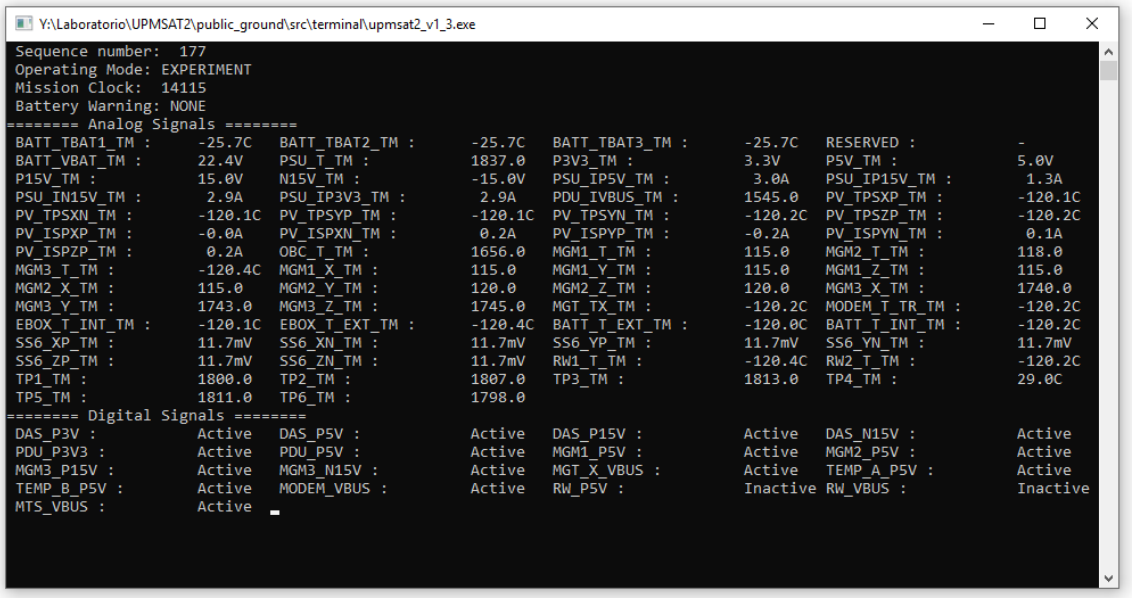

Figure 2: UPMSat-2 decoding appliaction.

The distributed executable file has been built for Windows 64 bits operating system, but is also known to work in Ubuntu Linux systems using wine64.

4 Develop your own decoder

In this section the required information is provided to build your own decoder for the UMPSat-2 public telemetry messages, i.e., the required information to interpret the Information Field of the data frames shown in Figure 1.

As mentioned before, the overall length of the information field is 102 bytes, and its internal structure is divided in two main sections: an application-defined header, and a data field, as shown in Figure 4.

The application-defined header is composed of 3 bytes carrying the following information:

- Command ID: 1 byte specifying the type of TM message. Public TM messages are named "Hello" messages with a binary value of 0x20.
- Sequence Number: 1 byte representing the number of sequence of this TM message. Range: 0-255.
- Length: 1 byte specifying the length of the TM Data field. For public TM this is always 99 bytes.

The TM Data field contains the application data, as shown in Figure 4. It is composed of the mission time at which the TM message was sent, and the last satellite housekeeping

Header			Data	
Command_ID	Seq_Number	Length	Sent_time	Housekeeping Data
1 byte	1 byte	1 byte	4 bytes	95 bytes

data snapshot taken. This data includes the satellite operating mode (see Table **??**), the battery status warning (if any), mission time at which the snapshot was taken, as well as the last reading from analog and digital signals.

Housekeeping Data			
Operating Mode	Snapshot Time	Analog Data	Digital Data
1 byte	4 bytes	87 bytes	3 bytes

Code	Mode
Off	0x00
Test	0x01
Await_Launch	0x02
Launch	0x03
Latency	0x04
Initialization	0x05
Commissioning	0x06
Safe	0x07
Beacon	0x08
Nominal	0x09
Experiment	0x0A

Table 1: Operating mode codes

Analog signal values are transmitted as read from the Analog to Digital Converter (ADC), spanning 12 bits. As such, values range from 0 to 4095. For each kind of signal, a transfer function is referenced in Table 2, please note that some such functions are not to be disclosed for the moment. Bytes are numbered from 1 to 87 and bits inside a byte are numbered from 1 to 8.

Table 2: List of analog signals, with their corresponding starting byte and bit inside the Analog Data structure, with corresponding transfer function.

Signal	Starting byte	Starting bit	Transfer Function
BATT_TBAT1_TM	1	1	
BATT_TBAT2_TM	2	5	Eq. 2
BATT_TBAT3_TM	4	1	
Reserved	5	5	
BATT_VBAT_TM	7	1	Eq. 3
PSU_T_TM	8	5	
p3V3_TM	10	1	3.3V
p5V_TM	11	5	5V
p15V_TM	13	1	+15V
n15V_TM	14	5	-15V
PSU_Ip5V_TM	16	1	
PSU_Ip15V_TM	17	5	Eq. 4
PSU_In15V_TM	19	1	
PSU_Ip3V3_TM	20	5	
PDU_IVBUS_TM	22	1	
PV_TPSXp_TM	23	5	
PV_TPSXn_TM	25	1	
PV_TPSYp_TM	26	5	Eq. 1
PV_TPSYn_TM	28	1	
PV_TPSZp_TM	29	5	
PV_ISPXp_TM	31	1	Eq. 5
PV_ISPXn_TM	32	5	Eq. 6
PV_ISPYp_TM	34	1	Eq. 7
PV_ISPYn_TM	35	5	Eq. 8
PV_ISPZp_TM	37	1	Eq. 9
OBC_T_TM	38	5	
MGM1_T_TM	40	1	
MGM2_T_TM	41	5	
MGM3_T_TM	43	1	Eq. 1
MGM1_x_TM	44	5	
MGM1_y_TM	46	1	
MGM1_z_TM	47	5	
MGM2_x_TM	49	1	
MGM2_y_TM	50	5	
MGM2_z_TM	52	1	
MGM3_x_TM	53	5	
MGM3_y_TM	55	1	
MGM3_z_TM	56	5	
MGT_TX_TM	58	1	
MODEM_T_TR_TM	59	5	
EBOX_T_INT_TM	61	1	Eq. 1

Table 2: List of analog signals, with their corresponding starting byte and bit inside the Analog Data structure, with corresponding transfer function.

Signal	**Starting byte**	**Starting bit**	**Transfer Function**
EBOX_T_EXT_TM	62	5	
BATT_T_EXT_TM	64	1	
BATT_T_INT_TM	65	5	
SS6_Xp_TM	67	1	Eq. 10
SS6_Xn_TM	68	5	
SS6_Yp_TM	70	1	
SS6_Yn_TM	71	5	
SS6_Zp_TM	73	1	
SS6_Zn_TM	74	5	
RW1_T_TM	76	1	Eq. 1
RW2_T_TM	77	5	
TP1_TM	79	1	
TP2_TM	80	5	
TP3_TM	82	1	
TP4_TM	83	5	Eq. 1
TP5_TM	85	1	
TP6_TM	86	5	

Digital signals are composed of just one bit, exception made for the Battery_Warning signal that can take up to 4 values and is thus represented with 2 bits, as shown in Table 3.

Signal	Byte	Bit	Notes
Battery_Warning	1	1..2	0x00 = None, 0x01 = Low, 0x02 = Critical, 0x03 = High
DAS_p3V	1	3	
DAS_p5V	1	4	
DAS_p15V	1	5	
DAS_n15V	1	6	
PDU_p3V3	1	7	
PDU_p5V	1	8	
MGM1_p5V	2	1	
MGM2_p5V	2	2	
MGM3_p15V	2	3	
MGM3_n15V	2	4	
MGT_X_VBUS	2	5	
TEMP_A_p5V	2	6	
TEMP_B_p5V	2	7	
MODEM_VBUS	2	8	
RW_p5V	3	1	
RW_VBUS	3	2	
MTS_VBUS	3	3	

Table 3: List of digital signals, with their byte and bit position inside the Digital Data structure.

4.1 Analog data transfer functions

For all equations, n is the value received for the given signal.

4.1.1 Temperatures

$$T(^{\circ}C) = \begin{cases} 0,336 * (n - 1708,1) & \text{when } n >= 1707,0 \\ 6,41 * [4,15 - sqrt(17,24 - 0,31 * (n - 1712,2))] & \text{otherwise} \end{cases} \quad (1)$$

4.1.2 Battery temperatures

$$T(^{\circ}C) = 1,2 * [60 - sqrt(3600 - 1,72 * (2333 - n))] \quad (2)$$

4.1.3 Battery voltage

$$V(V) = (1/264,1) * (n + 4039,2) \quad (3)$$

4.1.4 PSU currents

$$I(A) = (1/232,6) * (UPC - 0,42) \quad (4)$$

4.1.5 Current from solar panel Xp

$$I(A) = (1/810,64) * (n - 1688,3) \tag{5}$$

4.1.6 Current from solar panel Xn

$$I(A) = (1/656,02) * (n - 1622,3) \tag{6}$$

4.1.7 Current from solar panel Yp

$$I(A) = (1/853,8) * (n - 1798,4) \tag{7}$$

4.1.8 Current from solar panel Yn

$$I(A) = (1/810,64) * (n - 1688,3) \tag{8}$$

4.1.9 Current from solar panel Zp

$$I(A) = (1/638,81) * (n - 1571,8) \tag{9}$$

4.1.10 Solar sensor voltages

$$V(mV) = (1/17,7) * (n + 201,4) \tag{10}$$