LC-4102a
Terminal Node Controller
User Manual

Document change log

Rev.	Date	Author	Pages	Description
0	2014-10-17	Artur Scholz	All	Initial release

Table of Contents

1 Introduction

The LC-4102a Terminal Node Controller is used at the ground station for interfacing between a computer system (running a mission control system) and a radio system. The device is designed for the use of ECSS/CCSDS telecommand and telemetry frames. It utilizes a simple protocol for interfacing to a standard PC via serial port.

Features:

- ECSS/CCSDS frame protocol for telecommands
- Modified KISS protocol for TNC to PC communication
- Fixed FSK mode operation at 1200 bps

2 Applicable and reference documents

Reference	Title	Issue
RD01	KISS protocol, http://www.ax25.net/kiss.aspx	-
RD02	ECSS-E-ST-50-01C, Space data links – Telemetry synchronization and channel coding	-
RD03	ECSS-E-ST-50-04C, Space data links – Telecommand protocols synchronization and channel coding	-

3 Definitions and abbreviations

Abbreviation	Definition
ASM	Attached sync marker
CADU	Channel Access Data Unit
CCSDS	Consultative Committee for Space Data Systems
CLTU	Communications Link Transmission Unit
ECSS	European Cooperation on Space Standardization
FEND	Frame end
FESC	Frame escape
KISS	Keep it simple, stupid
MSB	Most significant bit
PTT	Push to talk
RF	Radio frequency
RSSI	Received signal strength indication
TC	Telecommand
TFESC	Transposed frames escape
TM	Telemetry

4 Specification

4.1 Absolute maximum ratings

Parameter	Min	Max	Unit
Supply voltage		26	V

4.2 Physical characteristics

Parameter	Min	Nom	Max	Unit
Mass		225		g
Height		46		mm
Width		110		mm
Depth		140		mm

4.3 Electrical characteristics

Parameter	Min	Nom	Max	Unit
Supply voltage		9		V
Power consumption			4.5	W
Microcontroller operating frequency		24.5		MHz
UART baud rate		115200		bps
DATAOUT output level	0		1.65	V
DATAIN bias level	0		1.65	V

4.4 RF and protocol characteristics

Parameter	Min	Nom	Max	Unit
FSK baud rate		1200		bps
Size of CLTU			210	Byte
Fixed size of CADU	16	64	2048	Byte

5 Interfaces

5.1 Mechanical interfaces

The mechanical layout of the device and its dimensions are shown in Illustration 1. A short description of all items is provided in Table 1.

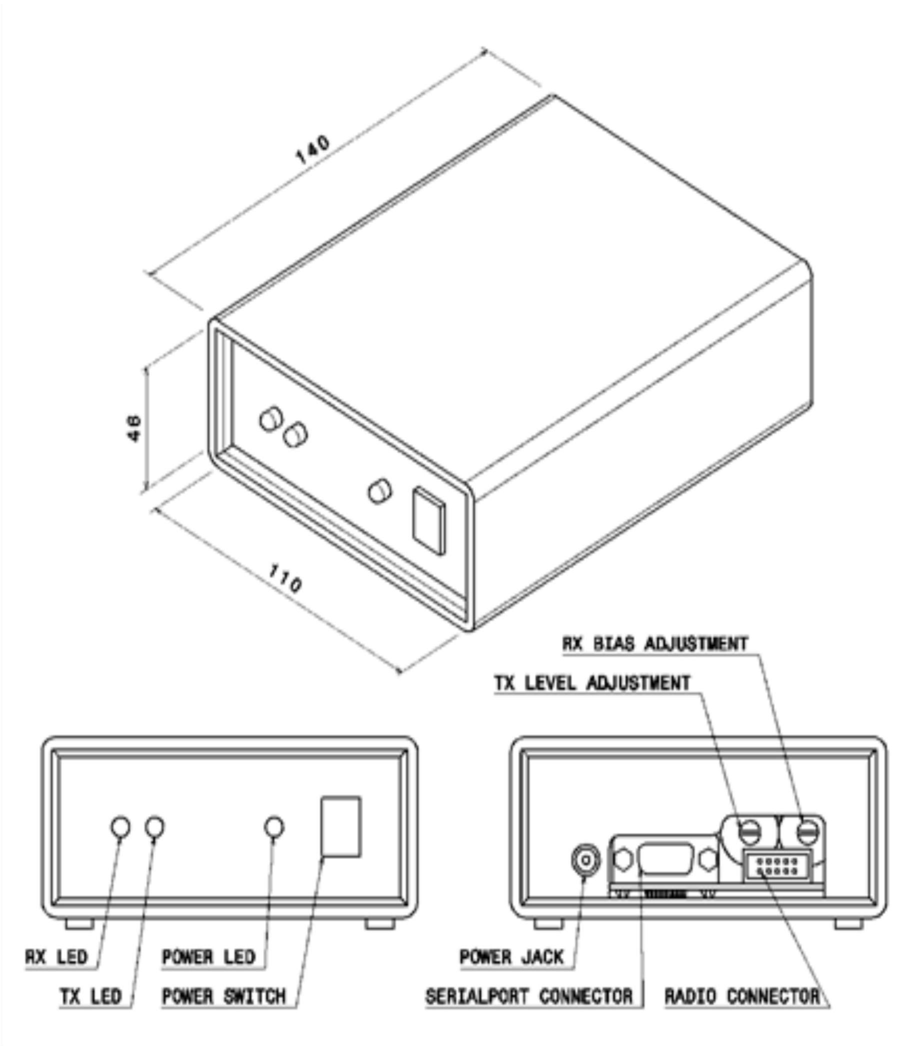

Illustration 1: Mechanical layout

Table 1: Description of front and back panel elements

Element	**Function**
RX LED	Indicates that frames are being received
TX LED	Indicates that transmission is active
POWER LED	Indicates that device is powered
POWER SWITCH	For switching the device on and off
POWER JACK	9 Volt power supply input, ø2.1 mm, center positive
SERIALPORT CONNECTOR	Serial port for connection to PC
RADIO CONNECTOR	Connects to the radio
TX LEVEL ADJUSTMENT	Adjust output level for DATAOUT (TX)
RX BIAS ADJUSTMENT	Adjust bias level for DATAIN (RX)

5.2 Electrical interfaces

The electrical interfaces to the device are shown in Illustration 2 and described in Table 2, Table 3, and Table 4.

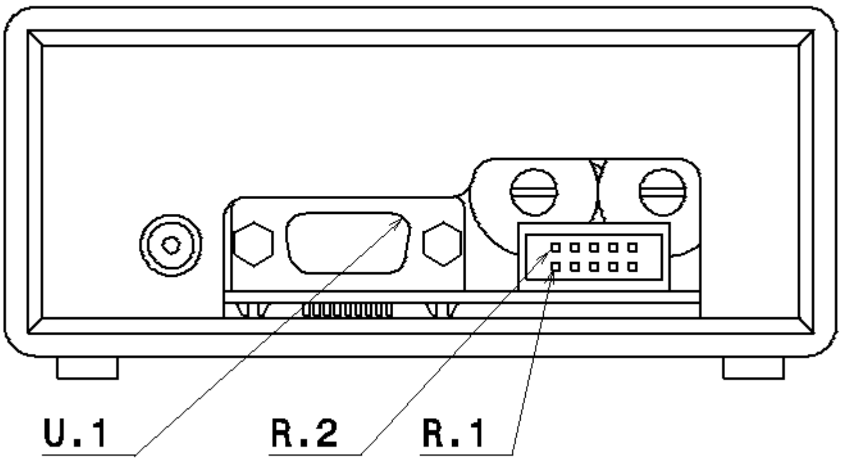

Illustration 2: Connector layout

Table 2: Radio connector

Pin	Signal	Description
R.1	-	Do not connect
R.2	+3.3V	+3.3 Volt output
R.3	ADJ0	For adjustment
R.4	**DATAIN**	**Input for data (from external radio)**
R.5	ADJ1	For adjustment
R.6	**-PTT**	**Push-to-talk input. Active low**
R.7	ADJ2	For adjustment
R.8	**DATAOUT**	**Output of data (to external radio)**
R.9	GND	Ground
R.10	**GND**	**Ground**

* Only the bold printed connections need to be wired to the radio

Table 3: Serial port connector

Pin	Signal	Description
U.2	RXD_TTL	RXD (TTL level)
U.3	TXD_TTL	TXD (TTL level)
U.5	GND	Ground

Table 4: Programming interface connector

Pin	Signal	Description
PI.1	+3.3V	+3.3V power supply
PI.2	C2D	C2 programming interface data
PI.3	C2CK	C2 programming interface clock
PI.4	GND	Ground.

* This connector is located on the PCB.

The programming interface is used for updating the firmware.

5.3 Software interfaces

Interfacing between the TNC and a computer for transfer of beacon/telemetry and telecommand data is accomplished via the UART bus.

The protocol is kept very simple, in close analogy to the KISS protocol (RD01). Data is transferred in frames between master and slave, one frame after the other. Frames are delimited by a special FEND (Frame End) character at the start and end. The first byte of a frame following the delimiter indicates the type of data in the frame.

FEND	CMD	Data	Data	Data	Data	...	Data	FEND

The values of the special characters are:

FEND	Frame End	0xC0
FESC	Frame Escape	0xDB
TFEND	Transposed Frame End	0xDC
TFESC	Transposed Frame Escape	0xDD

In case that a FEND character appears in the data, it is translated into the two byte sequence FESC TFEND (Frame Escape, Transposed Frame End). Likewise, if the FESC character ever appears in the user data, it is replaced with the two character sequence FESC TFESC (Frame Escape, Transposed Frame Escape).

As characters arrive at the receiver side, they are appended to a buffer containing the current frame. Receiving a FEND marks the end of the current frame. Receipt of a FESC puts the receiver into "escaped mode", causing the receiver to translate a following TFESC or TFEND back to FESC or FEND, respectively, before adding it to the receive buffer and leaving escaped mode. Receipt of any character other than TFESC or TFEND while in escaped mode is an error; no action is taken and frame assembly continues. A TFEND or TESC received while not in escaped mode is treated as an ordinary data character.

The CMD byte can have take following values:

UPLINK_CLTU	0x01	CLTU to be sent from computer to TNC
DOWNLINK_CADU	0x02	CADU received from TNC to computer
HANDSHAKE	0x0A	Handshake (answer)
TEST_CONNECTION	0x0B	Test port connection
TX_STATE	0x0C	Transmitter state (on/off)
IDLE_SEQUENCE	0x0D	Send idle pattern (on/off)
CADU_LENGTH	0x0E	Set the fixed CADU length

See Section Operations on how to use those commands.

6 Design description

6.1 Hardware description

The TNC consists of a power conditioner, a UART interface module, and microcontroller. The microcontroller handles the interfacer to the computer via the UART interface module. It further provides the interface to an external radio, and conditions its input and output.

For the output of data to the external radio (i.e. DATAOUT line), it provides the capability to adjust the signal level for a binary '1' from 0V to 1.65V (binary '0' is always at ground).

For the input of data from the external radio (i.e. DATAIN line) it first removes the DC offset from the signal through a high pass filter. It then compares the incoming signal level with an adjustable reference signal level, to discriminate between binary '1'and '0'.

6.2 Software description

The TNC provides is the node between computer and radio system for sending telecommand frames (CLTUs) and receiving telemetry frames (CADUs).

When not sending, the TNC continuously reads from the radio output, shifts these bits into a small buffer, and checks if the synch pattern (ASM) has been detected. It then receives the CADU (i.e. telemetry) frame bit for bit, shifts it into an octet buffer (i.e. byte buffer) and sends appends each completed octet to the frame being send to the computer via the KISS protocol. Since CADU frames have fixed length, the TNC knows when a full CADU has been received and then terminates the transfer to the computer.

For sending of CLTUs (i.e. telecommands) the computer first switches on the transmitter, then activates the sending of acquisition sequence (which is composed of idle pattern) and then transfers the CLTU via the KISS protocol to the TNC. Once the whole CLTU has been transferred, the TNC starts to transmit the CLTU bit stream to the radio, provided that the transmitter is left enabled. The computer finishes the transmit session by switching of the transmitter.

7 Installation

7.1 Making the connections

Plug a serial cable to the device and connect it with a PC (for example by using a serial to USB cable). Make sure that the device power switch is off and then plug a 9V power supply to the device. Finally, connect the bold written pins in Table 2 from the device's radio connector to the radio's data connector. For proper operation, the device must be adjusted to the radio's signal levels. This is described in the next sections.

7.2 Transmit adjustments

Refer to your radio for the specification on the recommended and allowable input levels on its DATA IN pin. Commonly this is about 0.4 Vpeak-peak.

To adjust the output level of the DATAOUT signal, temporarily disconnect the DATAOUT signal line from the device's radio connector and measure its voltage level. Rotate the TX LEVEL ADJUSTMENT potentiometer clockwise to increase and counter-clockwise to decrease the output level to the desired value.

Note: Same commercial radios output an inverted bit stream. In such case, one needs to set the bit invert flag in the TNC.

7.3 Receive adjustments

The device requires that the radio recovers the FSK modulated baseband signal from the RF signal fairly accurately. Internally, the device decouples and filters the signal provided by the radio and makes this signal available at the ADJ1 pin. An example of this signal when the radio receives only noise is shown in Illustration 3. Make sure that your radio is tuned properly to produce similar results. Determine the mean value of this signal, as shown as the red line.

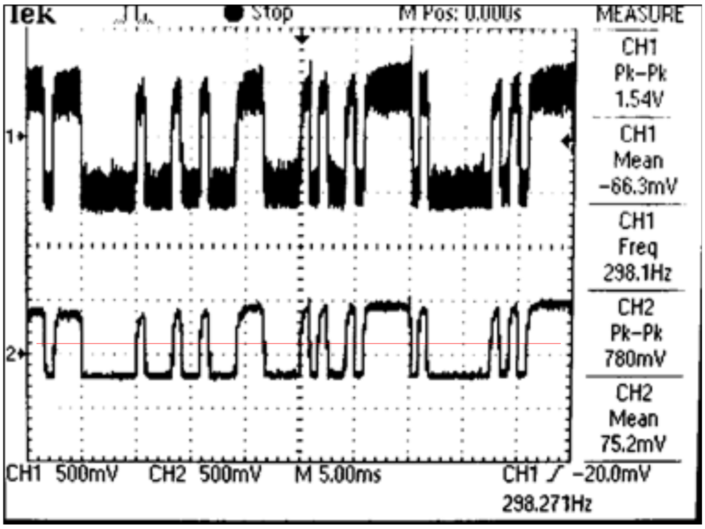

Illustration 3: Raw signal from radio (top) and filtered signal at ADJ1 (bottom)

Next, adjust the signal at the ADJ2 pin to this previously determined mean value, by using the RX BIAS ADJUSTMENT potentiometer clockwise for decrease or counter-clockwise for increase. The meaning of this bias voltage is that every signal level of the filtered signal in Illustration 4 that is above this threshold will be converted to 3.3 V whereas all signals below this threshold will be converted to 0V (CMOS level) for use by the TNC. As a last step, measure the signal at ADJ0 to see the recovered binary sequence in CMOS level.

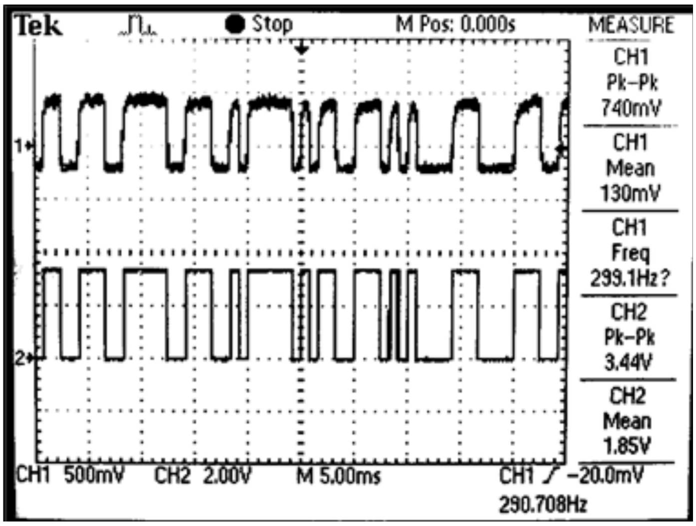

Illustration 4: The filtered radio signal (top) correctly converted into CMOS level (bottom).

8 Operations

8.1 Testing the connection

To ensure that the device is connected correctly to the computer and the KISS frames exchange works properly, the TEST_CONNECTION command can be used.

The computer sends an TEST_CONNECTION frame via the serial port, with following structure:

FEND	CMD	FEND

Where CMD = TEST_CONNECTION. The TNC then responds with an acknowledge frame:

FEND	CMD	FEND

Where CMD = HANDSHAKE.

8.2 Adjusting TNC settings

So far, the only setting that can be adjusted via software is the length of the CADU frame. CADU frames have (mission) fixed length. This is so because they only have markers for the start of a frame (i.e. the attached synch marker) and not for the frame end. Therefore they must be fixed length, to let the TNC know when the frame is complete.

8.3 Receiving

When not transmitting, the TNC is constantly in receive mode. It searches the incoming bit stream from the radio for the synch pattern (ASM), which marks the begin of a CADU, which is used to transport telemetry from the spacecraft. The structure of a CADU is as follows:

CADU – Channel Access Data Unit	
Attached Synch Marker 0x1ACFFC1D	Data
4 byte	Fixed, max. 2048 byte

When the TNC has detected an ASM, it will start a KISS frame transfer (sending a FEND) and then transfer the received CADU while it is being received, as follows:

FEND	CMD	CADU	FEND

Where CMD = DOWNLINK_CADU. Note again that the CADU frame is transferred at the same time while it is being received, meaning that there is no buffering of the whole CADU at the TNC. This is possible because the UART connection always to introduce delays in the transfer.

8.4 Transmitting

For transmitting CLTUs (i.e. telecommands), the computer shall operate the TNC in the following manner.

First, the transmitter must be turned on. This is accomplished via the following KISS frame:

FEND	CMD	PAR	FEND

Where CMD = TX_STATE, and PAR = ON. This mode is equivalent to CMM-1 "Unmodulated carrier only", as defined in RD03.

Then the sending of the acquisition sequence shall be activated, which is an idle pattern, i.e. a sequence of alternating "ones" and "zeros" starting with either a "one" or a "zero". This is a preamble that provides for initial symbol synchronization for the received on board the satellite. It should be sending at least 128 bits before switching to CMM-3. The command is as follows:

FEND	CMD	PAR	FEND

Where CMD = IDLE_SEQUENCE, and PAR = ON. This mode is equivalent to CMM-2 "Carrier modulated with acquisition sequence", as defined in RD03.

Then the sending of a CLTU can be initiated as follows:

FEND	CMD	CLTU	FEND

Where CMD = UPLINK_CLTU and the CLTU is structured as follows:

CLTU – Command Link Transmission Unit					
Start Sequence 0xEB90	1st Codeblock	2nd Codeblock	...	n-th Codeblock	Tail sequence
2 byte	8 byte	8 byte		8 byte	8 byte

A CLTU is of variable length and terminated by a tail sequence, which can be either:
"0xC5 0xC5 0xC5 0xC5 0xC5 0xC5 0xC5 0x79" or "0x55 0x55 0x55 0x55 0x55 0x55 0x55 0x55"

The sending of a CLTU is described as mode CMM-3 "Carrier modulated with telecommand data", as defined in RD03. Several CLTUs can be send during one transmit session (i.e. while the transmitter is on), provided that there is a small amount of idle bits being sent in between (which is then defined as mode CMM-4 "Carrier modulated with idle sequence"). The length of the idle sequence shall be at least 8 bits. This is achieved easily by waiting for short delay between the sending of CLTUs, while sending of acquisition sequence is left enabled. The transmit session is ended by stopping the transmission of the acquisition sequence and switching of the transmitter (see IDLE_SEQUENCE and TX_STATE commands above, both with parameter OFF).