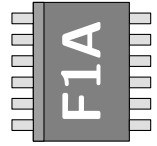


Michal Dvořák
www.F1A.info



Altimeter

Altimeter for Free Flight Models

User Manual for Firmware Version 1.00

1 Device Overview

The pressure altimeter works on the principle that the pressure of air varies in a known way with height. After power on, it measures the pressure value for zero level. Then, it measures pressure values and calculates a height difference from the zero level.

Measured altitude samples are stored into EEPROM memory. Every 10th sample is a temperature value. Altimeter has an UART interface with defined commands. So, it can work in stand alone mode or in UART operation mode. The USB cable with USB-UART converter is used for communication with PC software, which enables to download records and config the altimeter.

General Features

- Very miniature dimensions – 15x18 mm 1.5 g, with solid heatsink and glue 2.5 g.
- Wide range of supply voltage from 2.8 V to 16 V. Contains 3.0 V voltage regulator.
- Altitude resolution 0.1 m. Temperature resolution 0.1 °C.
- Large EEPROM memory (256 kb) for measured samples. The capacity is for 16000 samples (16-bit).
- **Bootloader** – enables to upgrade firmware using USB cable.
- **Stand Alone Mode** – Just connect the battery and altimeter starts to measure and store samples. Adjustable parameters by PC software – time step, altitude trigger, max. record length.
- **UART Operation Mode** – Communication interface with master device (Timer). Fixed time step 0.5 s.
- **PC Mode** – Enables to download captured records and set module parameters.

2 Device Connection

The Altimeter module has 4 pin interface. Two pins for power supply and two for UART serial link. It contains 3 V voltage regulator. The TX pin is an open collector output. The RX pin is an input with pull-up resistor. So, the master device can operate at different voltage level and has to use open collector output.

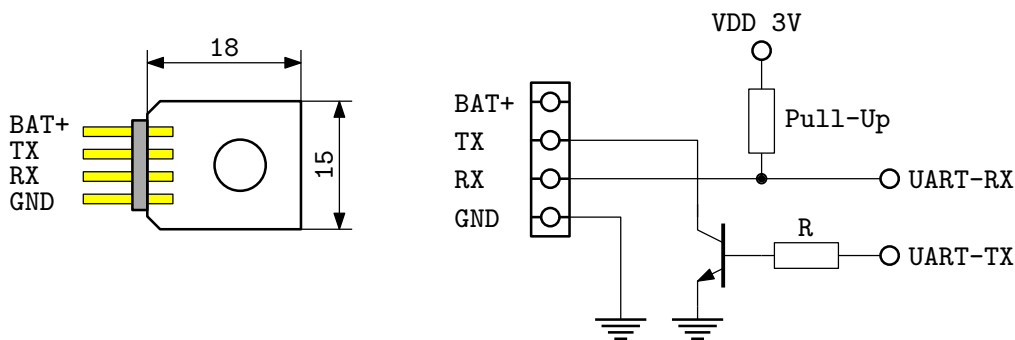


Figure 1: Altimeter module interface.

3 Device Operation

After power on, the module waits 200 ms for a command from PC bootloader software. Then it waits for a message to enter PC mode or Timer mode. Concurrently it is measuring a value of reference pressure for zero altitude. If no message is received on UART interface in first 5 s, it continues in stand alone mode.

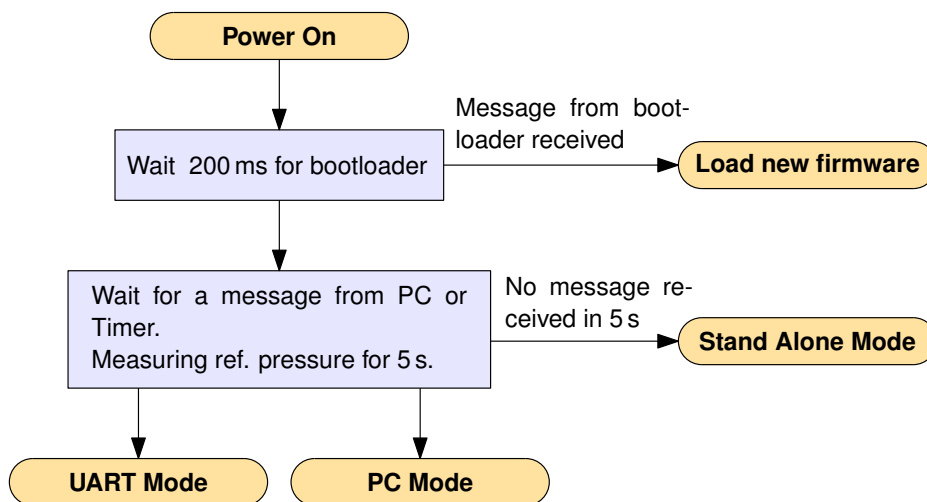


Figure 2: Device operation flowchart.

3.1 Stand Alone Mode

Altimeter module can works as a stand alone device and just store samples of altitude. There are some adjustable parameters relevant to this mode.

- **Time Step** – This is the time step for storing samples.
- **Altitude Trigger** – This is the trigger value to start storing samples. You can set about 5 m to start record after model start. Or set about 55 m to start the record after glider is released from towline.
- **Record Length** – This is the maximal length of record. Set 0 for infinity.

Time Step (s)	Record Length (hr)
0.5	2
1.0	4
2.0	8
5.0	20
10.0	40
30.0	120
60.0	240

Table 1: Record length in dependence on time step. Altimeter has 256 kb EEPROM Memory – it means 16000 samples (16-bit).

3.2 PC Mode

It is possible to connect the Altimeter module with PC software using USB-UART converter cable. Before first use, it is necessary to download and install the driver for USB cable.

At first, connect the USB cable and open the port. Then, connect the Altimeter module. Otherwise, if a message from PC software is not received after power on (in 5 s), the module continues in stand alone mode and begins new record.

The procedure of connecting Altimeter module:

1. Connect USB cable excluding Altimeter to PC.
2. Select port and click Open Port on Connect tab.
3. Connect Altimeter module to the USB cable and wait for response.

4 Communication Protocol

4.1 UART Operation Mode

This mode is intended for communication with Timer as a master device. It utilizes 9-bit multiprocessor/addressable communication mode. It enables to connect more devices to one UART link. Each device has its own address. Data packets contains a header with slave address and 8 bit CRC checksum.

- Multiprocessor mode – 9 data bits, 1 start bit, 1 stop bit, bit 8=1 for address byte, bit 8=0 for data byte.
- Baud rate 9600 b/s.
- LSB first, idle state is high.
- Device slave address: 'A' (0x41).

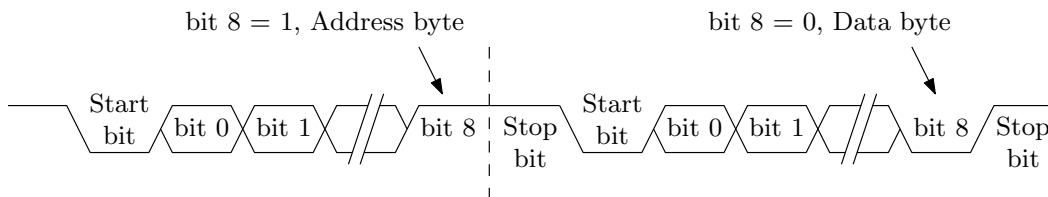


Figure 3: UART operation in 9-bit multiprocessor/addressable mode.

Recipient address	Message length	Message	CRC-8
[1B, b8=1]	[1B, b8=0]	[1-255B, b8=0]	[1B, b8=0]

Table 2: Data packets from master (Timer).

Header 0xFF	Slave address	Message length	Message	CRC-8
[1B, b8=0]	[1B, b8=0]	[1B, b8=0]	[1-255B, b8=0]	[1B, b8=0]

Table 3: Data packets from Altimeter module.

- To enter UART mode, Altimeter must receive **RCx** with model code.
- To begin store samples, Altimeter must receive one of these modes STORE, LAUNCH, GLIDE, FAST-GLIDE, PREGLIDE.
- To enter sleep mode, Altimeter must receive IDLE or BEACON mode.
- Altimeter confirms valid packet by acknowledge char **ACK(0x06)**. Not valid packet by **NAK(0x15)**.
- Master device does not confirm packet reception.

MSG	Description
RCx	Model designation number x
RMI	Report IDLE mode
RMS	Report STORE mode
RMA	Report ACCELERATE mode
RML	Report LAUNCH mode
RMG	Report GLIDE mode
RMF	Report FASTGLIDE mode
RMP	Report PREGLIDE mode
RMD	Report DT mode
RMB	Report BEACON mode
GAA	Get Actual Altitude [0.1 m]
GAH	Get Actual Altitude [1 m]
GAP	Get Actual Pressure [10 Pa]
GAT	Get Actual Temperature [0.01 °C]

Table 4: Messages from Timer.

MSG	Description
Axx	Return altitude [0.1 m], signed int <LSB:MSB>
Hxx	Return altitude [1 m], signed int <LSB:MSB>
Pxx	Return altitude [10 Pa], signed int <LSB:MSB>
Txx	Return altitude [0.01 °C], signed int <LSB:MSB>

Table 5: Messages from Altimeter.

4.2 PC Mode

This mode is intended for communication with PC Altimeter software. It utilizes 8-bit UART communication mode. Data packets contains 8 bit CRC checksum.

- UART mode – 8 data bits, 1 start bit, 2 stop bits.
- Baud rate 9600 b/s for begin communication, then 62500 b/s.
- LSB first, idle state is high.

Header 0xFF or 0xFE	Message length	Message	CRC-8
[1 B]	[1 B]	[1-255 B]	[1 B]

Table 6: Data packets from PC and Altimeter.

- To enter PC mode, Altimeter must receive **GDI** message after power on.
- After reception of **BDR xx** message, PC must switch to given baudrate.
- Messages from PC consists from 3 chars. Numbers are 16-bit integers **<LSB:MSB>**.
- Messages from Altimeter consists from strings. Numbers are ASCII strings.
- PC confirms valid packet by acknowledge char **ACK(0x06)**. Not valid packet by **NAK(0x15)**.
- Altimeter module does not confirm packet reception.

MSG	Description
GDI	Get device info
GCG	Get config
GAR	Get all records
DIS	Discard old records
RES	Reset config and memory
ERR	Get error message
Txx	Set time step [1 ms], <LSB:MSB>
Axx	Set altitude trigger [1 m], <LSB:MSB>
Lxx	Set record length [1 min], <LSB:MSB>
GAA	Get altitude
GTT	Get temperature
GPP	Get pressure
GRP	Get ref. pressure
SRP	Set ref. pressure to actual value

Table 7: Messages from PC software.

MSG	Description
"DEV Altimeter\n"	Return device name
"VER 1.00\n"	Return device version
"MEM 256 kb\n"	Return memory size
"BDR 62500\n"	Return new baudrate
"TSP 0.5 s\n"	Return time step
"TGR 55 m\n"	Return altitude trigger
"LEN 10 min\n"	Return record length
"A 10.2\n"	Return altitude [0.1 m]
"P 1020.45\n"	Return pressure [0.01 hPa]
"T 20.50\n"	Return temperature [0.01 °C]
"R 1013.25\n"	Return ref. pressure [0.01 hPa]

Table 8: Messages from Altimeter. Numbers are ASCII strings.

Message with Altitude Records

Altimeter module splits the ASCII stream into smaller data packets. It waits for **ACK** acknowledge from PC after each packet. If **NAK** or other char is received, it will repeat the packet with error header **0xFE**. It also reports the progress of download process in percentages e.g. **PGS45**. The record begins with a header which contains header length, record id, model code and time step. Values with prefix T are temperature.

```
STX ALTI HR4 ID5 MC10 TS500 RP952.8 ML T19.7 45.8 62.2 68.5 75.3 77.4 77.6 77.6 76.7
76.3 75.7 T19.5 75.5 75.6 75.8 75.8 75.7 75.3 74.8 74.4 74.0 73.7 T19.5 73.6 73.6 73.6
73.4 73.4 73.5 73.6 74.1 74.4 74.6 ETX
```

DONE

4.3 CRC Checksum

Data packets contains 8 bit CRC (Cyclic Redundancy Check) to detect invalid packets. The polynomial generator is written in reverse order (LSB first)

$$R(x) = x^8 + x^5 + x^4 + 1 \rightarrow 10001100(1) = 0x8C.$$

An example of CRC computation in ANSI C language is below.

```
unsigned char get_crc8(unsigned char *msg, unsigned char len)
{
    unsigned char i, j;
    unsigned char rem = 0;

    for(i=0; i<len; i++)
    {
        rem = rem ^ msg[i];
        for(j=0; j<8; j++)
        {
            if( rem & 0x01 )
                rem = (rem >> 1) ^ 0x8C;
            else
                rem = rem >> 1;
        }
    }
    return rem;
}
```

5 Specifications

Parameter	Symbol	Min	Typ	Max	Units
Supply Voltage	VBAT	2.8		16.0	V
Regulated Voltage	VDD		3.0		V
Supply Current in Measure Mode			3.8		mA
Supply Current in Sleep Mode			0.07		mA

6 Terms of Use

This is an experimental device manufactured in small numbers at home conditions. It is intended for competition use by experienced fliers, who really want it and understand the risk that comes with this sort of device.

Disclaimer

The manufacturer assumes no responsibility for any damage or loss resulting from the use of this device. The manufacturer reserves right to make changes and improvements to the device described in this document without prior notice. The manufacturer does not warrant that this document is error-free.

Warranty

The manufacturer warrants this product to be free of manufacturing defects for the term of two years from the date of purchase. This warranty does not apply to the damage resulting from ordinary wear, accident, misuse, neglect, electrical surges or reversed polarity on connectors.

This document was created using L^AT_EX.