EVALUATION KIT AVALIABLE

TIM-MC1 series

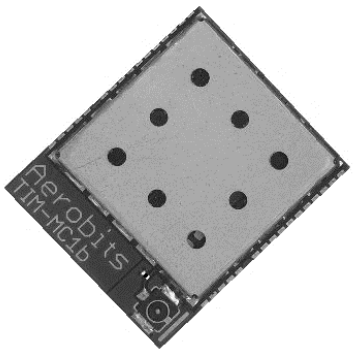
(TIM-MC1a, TIM-MC1b)

General Description

TIM-MC1 is a high-performance OEM receiver series operating at 1090MHz. It is based on the proven FPGA-In-The-Loop™ technology, which is a unique combination of a multi-core processor and FPGA. The patented solution allows high-speed RF data processing with significantly reduced number of electronic components. Simultaneous miniaturization of the module and its OEM nature open a wide range of possible applications. The basic version of TIM-MC1 offers the possibility of receiving and decoding ADS-B and Mode-A/C/S in different modes. The analysis of the power/quality of the RF signal and the use of time stamps facilitates the implementation of multilaterations, and the fast UART interface allows for the simple integration of the module with the user's system. In addition, large reserves of computing power open the way to customize the firmware and extend the module with non-standard functions. There are several communication interfaces, protocols and special functionalities available on request, such as ADS-B/Out (digital data). For more information please contact: support@aerobits.pl.

Applications

- SAA / DAA (Sense and Avoid / Detect and Avoid)
- UAS ground stations and high-density traffic surveillance
- UTM / U-Space construction (traffic surveillance network)
- Traffic-flow analysis and statistics
- Monitoring of 1090MHz band (signal integrity check)
- ADS-B/In/Out devices that meet the NextGen/SESAR philosophy
- Etc.



Features

- Fastest ADS-B implementation on a surface of <4cm^2
- Receiving of ADS-B, Mode-A/C/S with RF signal strength/quality analysis
- Time stamp (row data only) for multilateration
- High-resolution ADC with real-time signal processing; best-in-class aircraft tracking
- High sensitive front-end, jamming and ESD protection (only TIM-MC1b) with ranges over 300 km (open space, 1dBi antenna)
- Simple module integration via UART interface and AT commands
- Scalable OEM solution with enormous customization potential (additional functions or interfaces on request)
- Firmware update capability (uC and FPGA)
- Power consumption 5V/60mA (RF part), 3.3V/200mA (digital part)
- Small outline: 18.0 x 19.0 x 2.65mm, ver. (a); 22.5 x19.0 x 2.65mm, ver. (b), weight < 2g
- Designed to meet MOPS defined in TSO-C199

Ordering Information

Type	Symbol	Description
OEM module	TIM-MC1a	Without antenna connector
OEM module	TIM-MC1b	With antenna connector (U.FL)
Evaluation kit	EVAL-TIM-MC1	Based on TIM-MC1a

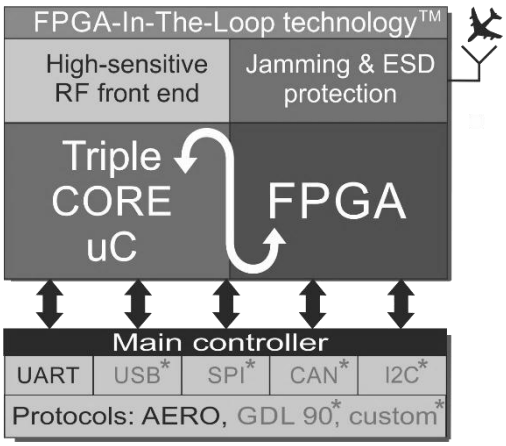


Fig. 1 Structure, interfaces and protocols of TIM-MC1 series.
*on request only (custom solutions)

This document is subject to change without notice. For technical questions, contact: support@aerobits.pl

1 ELECTRICAL CHARACTERISTICS 3
2 PIN DEFINITION 4
3 GENERAL PRINCIPLE OF OPERATION 6
3.1 STATES OF OPERATION 6
3.1.1 BOOTLOADERSTATE 6
3.1.2 RUNSTATE 6
3.1.3 CONFIGURATION STATE 6
3.2 TRANSITIONS BETWEEN STATES 6
3.2.1 BOOTLOADER TO RUN TRANSITION 6
3.2.2 RUN TO CONFIGURATION TRANSITION 7
3.2.3 CONFIGURATION TO RUN TRANSITION 7
3.2.4 CONFIGURATION TO BOOTLOADER TRANSITION 7
4 UART CONFIGURATION 8
5 SETTINGS 9
5.1 WRITE SETTINGS 9
5.2 READ SETTINGS 9
5.3 ERRORS 9
5.4 COMMANDS ENDING 9
5.5 UPPERCASE AND LOWERCASE 9
5.6 AVAILABLE SETTINGS 9
5.7 EXAMPLE 10
6 COMMANDS 11
6.1 COMMANDS IN BOOTLOADER AND CONFIGURATION STATE 11
6.1.1 AT+LOCK 11
6.1.2 AT+BOOT 11
6.2 COMMANDS IN CONFIGURATION STATE 11
6.2.1 AT+CONFIG 11
6.2.2 AT+SETTINGS? 11
6.2.3 AT+SETTINGS_DEFAULT 11
6.2.4 AT+SERIAL_NUMBER? 12
6.2.5 AT+FIRMWARE_VERSION? 12
6.2.6 AT+REBOOT 12
6.2.7 AT+REBOOT_BOOTLOADER 12
6.3 COMMANDS IN RUN STATE 12
7 AEROPROTOCOL 13
7.1 CRC 13
7.2 AIRCRAFT MESSAGE 13
7.3 MODE-S RAW FRAMES 14
7.4 MODE-AC RAW FRAMES 14
7.5 EXTENDED RAW FRAMES 15
7.6 STATISTICS MESSAGE 15
7.7 CALIBRATION OF RAW FRAMES 16
8 MINIMUMINTEGRATION RECOMMENDATION 17
9 MECHANICAL DRAWING 18
10 RECOMMENDED LAYOUT 19
11 REVISION HISTORY 20

1 ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings			
Parameter	Min.	Max.	Unit
Storage temperature	-40	+85	°C
Supply voltage (VCC1)	2.7	3.6	DCV
Supply voltage (VCC2)	4.0	5.5	DCV
Other pin voltage	VSS-0.4	VCC1+0.4	DCV
RF input power	-	+15	dBm
Recommended Operating Conditions			
Parameter	Min.	Max.	Unit
Temperature	-40	+70[1]	°C
Supply voltage VCC1	3.0	3.6	DCV
Supply voltage VCC2	4.5	5.5	DCV

Tab. 1 Absolute maximum ratings and recommended operation conditions.

General Electrical Specification					
Parameter	Description	Min.	Typ.	Max.	Unit
Carrier frequency		-	1090	-	MHz
RX sensitivity	For operation at 50Ω U.fl connector (TIM-MC1b)	-	-85	-	dBm
Input Low Voltage	RESET, UARTs, CAN, USB, SPI, I2C	-0.3	-	0.8	DCV
Input High Voltage	RESET, UARTs, CAN, USB, SPI, I2C, GPIO	0.7VCC1	-	VCC1+0.3	DCV
Output Low Voltage	UARTs, CAN, USB, I2C, SPI, GPIO	-	-	0.4	DCV
Output High Voltage	UARTs, CAN, USB, I2C, SPI, GPIO	VCC1-0.4	-	-	DCV
Current consumption (digital part)		-	200	-	mA
Current consumption (RF front end)		-	60	-	mA

Tab. 2 General electrical specification.

[1] Cooler required – please contact support@aerobits.pl

This document is subject to change without notice. For technical questions, contact: support@aerobits.pl

2 PIN DEFINITION

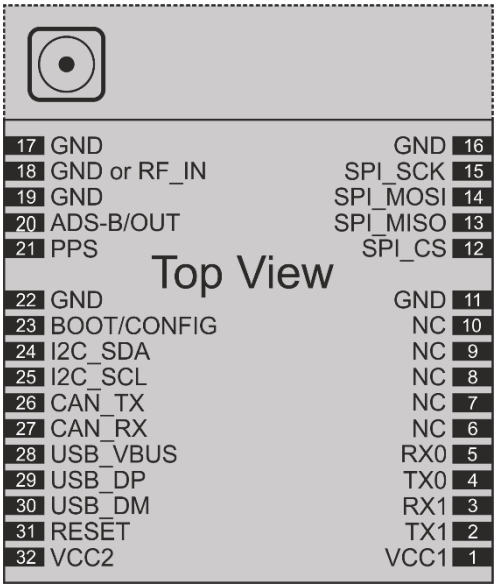


Fig. 2 Pin assignment for TIM-MC1a or TIM-MC1b.

Pin No.	Pin Name		Pin Type		Description	
1	VCC1		Power		3.3V (digital supply)	
2	TX1		CMOS Output		UART1 data output	
3	RX1		CMOS Input		UART1 data input	
4	TX0		CMOS Output		UART0 data output (standard interface)	
5	RX0		CMOS Input		UART0 data input (standard interface)	
6	NC		RES		Keep floating	
7	NC		RES		Keep floating	
8	NC		RES		Keep floating	
9	NC		RES		Keep floating	
10	NC		RES		Keep floating	
11	GND		GND		Common ground	
12	SPI_CS		CMOS Output		Serial Peripheral Interface / chip select	
13	SPI_MISO		CMOS Input		Serial Peripheral Interface / data input	
14	SPI_MOSI		CMOS Output		Serial Peripheral Interface / data output	
15	SPI_SCK		CMOS Output		Serial Peripheral Interface / clock	
16	GND		GND		Common ground	
17	GND		GND		Common ground	
18	TIM-MC1a	TIM-MC1b	TIM-MC1a	TIM-MC1b	TIM-MC1a	TIM-MC1b
	RF_IN	GND	RF Input	GND	RF Input (antenna)	Common ground
19	GND		GND		Common ground	
20	ADSB_OUT		CMOS Output		ADS-B Output (digital)	

21	PPS	CMOS Input	1PPS GNSS signal
22	GND	GND	Common ground
23	BOOT/CONFIG	CMOS Input	Bootloader / Configuration mode
24	I2C_SDA	Bi-directional	I2C Data line
25	I2C_SCL	Bi-directional	I2C Clock line
26	CAN_TX	CMOS Output	CAN transmit
27	CAN_RX	CMOS Input	CAN receive
28	USB_VBUS	Power	USB Power line
29	USB_DP	Bi-directional	USB+
30	USB_DM	Bi-directional	USB-
31	RESET	CMOS Input	Reset input / active low
32	VCC2	Power	5V (RF front end supply)
	Interfaces and I/Os not used in standard design (for custom implementation only)		

Tab. 3 Pinout.

3 GENERAL PRINCIPLE OF OPERATION

During work module goes through multiple states. In each state operation of the module is different. Each state and each transition is described in paragraphs below.

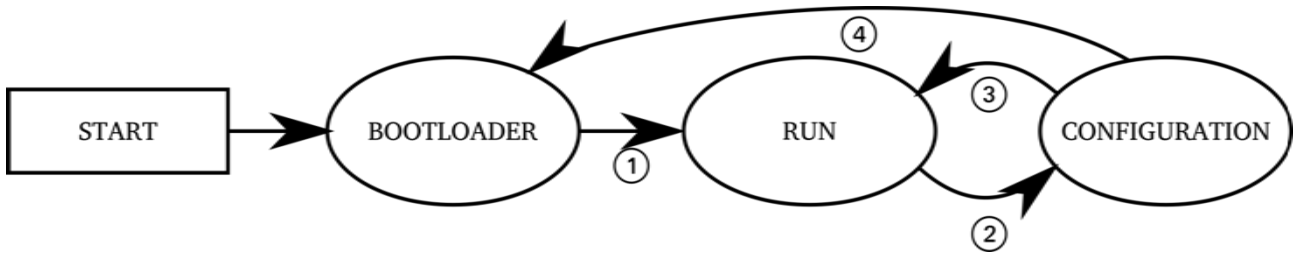


Fig 3. State machine of TIM-MC1 module

3.1 STATES OF OPERATION

3.1.1 BOOTLOADERSTATE

This is an initial state of TIM-MC1 module after restart. Firmware update is possible here. Typically module transits automatically to RUN state. It is possible to lock module in this state (prevent transition to RUN state) using one of BOOTLOADER triggers. UART baudrate is constant and is set to 115200bps. After powering up module, it stays in this state for 1s. If any BOOTLOADER trigger is not present, module will transit to RUN state. Firmware upgrade is possible with use of AEROBITS_updater.exe program. To acquirethis programpleasecontact: support@aerobits.pl.

3.1.2 RUNSTATE

In this state module is working and receiving the data from aircraft. It uses selected protocol to transmit received and decoded data to the host system. In this state of operation module settings are loaded from non-volatile internal memory, includingbaudrate.

3.1.3 CONFIGURATION STATE

In this mode change of stored settings is possible. Operation of the module is stopped and baudrate is set to fixed 115200bps. Change of settings is done by using AT-commands. Modifications are automatically stored in non-volatile memory on exiting state. Additional set of commands is also available in this state e.g. to reboot module into BOOTLOADER state, or to check serial number and firmware version. It is possible to lock module in this state (similarly to BOOTLOADER) using suitable command.

3.2 TRANSITIONS BETWEEN STATES

For each of state transitions, different conditions must be met, which are described below. Generally, the onlystable state is RUN. Module always tends to transit into this state. Moving to other states requires host to take some action.

3.2.1 BOOTLOADER TO RUN TRANSITION

BOOTLOADER state is semi-stable, that means the module requires additional action to stay in BOOTLOADER state. This transition will occur automatically if any of the required actions will not be taken. To stay in BOOTLOADER state (prevent this transition) one of following action must be processed:

- Pull BOOT/CONFIG pin low during start of module
- Send *AT+LOCK=1* command while device is in BOOTLOADER state (always after power on for approximately 1s)

- Send *AT+REBOOT_BOOTLOADER* command in CONFIGURATION state. This will move to BOOTLOADER state and will lock module in this state.

If none of preceding conditions are meet, the module will try to transit into RUN state. Firstly it will check firmware integrity. When firmware integrity is confirmed, module will transit into RUN state, if not, it will stay in BOOTLOADER state.

To transit into RUN state:

- Release or pull high BOOT/CONFIG pin
- If module is locked send,*AT+LOCK=0* command

When module enters RUN mode it will send *AT+RUN_START* command.

3.2.2 RUN TO CONFIGURATION TRANSITION

To transit from RUN into CONFIGURATION state, host should do one of the following:

- Pull BOOT/CONFIG pin low
- Send *AT+CONFIG=1* (using current baudrate). This method is not recommended, because module will support multiple protocols in future and Aerobits Sp. z o.o. cannot ensure that this command will be present in all protocols.

When moduleleaves RUN state it will send *AT+RUN_END* command and then *AT+CONFIG_START* command, on entering CONFIGURATION state. The former will be send using baudrate from settings, the latter will be send using 115200bps baudrate.

3.2.3 CONFIGURATION TO RUN TRANSITION

To transit from CONFIGURATION into RUN state, host should do one of the following:

- Release or pull high BOOT/CONFIG pin
- Send *AT+CONFIG=0* command.

When module leaves CONFIGURATION state it will send *AT+CONFIG_END* command and then *AT+RUN_START* command, on entering RUN state. The former will be send using 115200bps baudrate, the latter will be send using baudrate from settings.

3.2.4 CONFIGURATION TO BOOTLOADER TRANSITION

To transit from RUN into CONFIGURATION state, host should do one of the following:

- Send *AT+REBOOT_BOOTLOADER* command.
- Send *AT+REBOOT* and when module enters BOOTLOADER state, prevent transition to RUN state.

4 UART CONFIGURATION

Communication between module and host device is done using UART interface. This interface has following settings. In CONFIGURATION and BOOTLOADER state baudrate is fixed at 115200bps.

UART SETTINGS				
Parameter	Min.	Typ.	Max.	Unit
Baudrate	115200	115200	3000000	bps
Stop Bits Number	-	1	-	-
Flow Control	-	None	-	-
Parity Bit	-	None	-	-

Tab. 4 UART settings.

This document is subject to change without notice. For technical questions, contact: support@aerobits.pl

5 SETTINGS

In RUN state, operation of the module is determined based on stored settings. Settings can be changed in CONFIGURATION state using AT-commands. Settings can be written and read. Values are stored in non-volatile memory when leaving CONFIGURATION state, so they will be preserved during restart.

5.1 WRITE SETTINGS

AT+SETTING=VALUE

For example *AT+RAW_S=1*

Response: *AT+OK*

5.2 READ SETTINGS

AT+SETTING?

For example: *AT+RAW_S?*

Response: *AT+RAW_S=1*

5.3 ERRORS

Errors are reported using following structure:

AT+ERROR (DESCRIPTION)

DESCRIPTION is optional and contains information about error.

5.4 COMMANDS ENDING

Every command must be ended with one of the following characters: “\n”, “\r”, “\r\n”. Commandswithoutsuitableendingwill be ignored.

5.5 UPPERCASE AND LOWERCASE

All characters (except preceding AT+) used in command can be both uppercase and lowercase, so following commands are equal:

AT+RAW_S?

AT+raw_s?

NOTE: This statement is true in configuration state, not in bootloader state. in bootloader state all letters must be uppercase.

5.6 AVAILABLE SETTINGS

Setting	Min value	Max value	Default value	Comment
RAW_S	0	1	0	Showing raw MODE-S frames in AERO protocol: 0 - disable 1 - enable
RAW_AC	0	1	0	Showing raw MODE-AC frames in AERO protocol: 0 - disable

This document is subject to change without notice. For technical questions, contact: support@aerobits.pl

				1 - enable
RAW_S_EX	0	1	1	Should raw MODE-S frames have extended format: 0 - standard format 1 - extended format
RAW_AC_EX	0	1	1	Should raw MODE-AC frames have extended format: 0 - standard format 1 - extended format
AERO	0	1	1	Showing aircraft messages in AERO protocol: 0 - disable 1 - enable
STATS	0	1	1	Showing statistics messages in AERO protocol: 0 - disable 1 - enable
BAUDRATE	0	2	0	Baudrate in RUN state 0 - 115200bps 1 - 921600bps 2 - 3000000bps
PROTOCOL	0	0	0	Only AERO protocol currently available

Tab. 5 Settings.

5.7 EXAMPLE

If one wants to set the module to show only raw frames, MODE-S in standard format and MODE-AC in extended format, one should send following commands in CONFIGURATION state. "<<" means command send to module, ">>" means response.

```
<< AT+AERO=0\r\n
>> AT+OK\r\n
<< AT+STATS=0\r\n
>> AT+OK\r\n
<< AT+RAW_S=1\r\n
>> AT+OK\r\n
<< AT+RAW_AC=1\r\n
>> AT+OK\r\n
<< AT+RAW_S_EX=0\r\n
>> AT+OK\r\n
<< AT+RAW_AC_EX=1\r\n
>> AT+OK\r\n
```

To leave the CONFIGURATION state one should send:

```
<< AT+CONFIG=0\r\n
>> AT+OK\r\n
>> AT+CONFIG_END\r\n
>> AT+RUN_START\r\n
```

6 COMMANDS

Apart from settings, module supports set of additional commands. Format of this commands are similar to those used for settings, but they do not affect operation of module in RUN state.

6.1 COMMANDS IN BOOTLOADER AND CONFIGURATION STATE

6.1.1 AT+LOCK

AT+LOCK=1- set lock to enforce staying in BOOTLOADERor CONFIGURATION state

AT+LOCK=0 - remove lock

AT+LOCK? - check if lock is set

6.1.2 AT+BOOT

AT+BOOT? - check if module is in BOOTLOADER state

Response:

AT+BOOT=0 - module in CONFIGURATION state

AT+BOOT=1 - module in BOOTLOADER state

6.2 COMMANDS IN CONFIGURATION STATE

6.2.1 AT+CONFIG

AT+CONFIG=0 - transition to RUN state.

AT+CONFIG? - check if module is in CONFIGURATION state.

Response:

AT+CONFIG=0 - module in RUN state

AT+CONFIG=1 - module in CONFIGURATION state

6.2.2 AT+SETTINGS?

List all settings. Example output:

```
AT+RAW_S=0
AT+RAW_AC=0
AT+RAW_S_EX=1
AT+RAW_AC_EX=1
AT+AERO=1
AT+STATS=1
AT+BAUDRATE=0
AT+PROTOCOL=0
AT+CONFIG=1
AT+LOCK=1
```

NOTE: Apart from settings, this command also lists CONFIG and LOCK value.

6.2.3 AT+SETTINGS_DEFAULT

AT+SETTINGS_DEFAULT - set all settings to their default value.

6.2.4 AT+SERIAL_NUMBER?

AT+SERIAL_NUMBER? - Read serial number of module.

Response:

AT+SERIAL_NUMBER=0202041E43

6.2.5 AT+FIRMWARE_VERSION?

AT+FIRMWARE_VERSION? - Read firmware version of module.

Response:

AT+FIRMWARE_VERSION=10101017(May 11 2018)

6.2.6 AT+REBOOT

AT+REBOOT - Restart module.

6.2.7 AT+REBOOT_BOOTLOADER

AT+REBOOT_BOOTLOADER - Restart module to BOOTLOADER state.

NOTE: This command sets also lock.

6.3 COMMANDS IN RUN STATE

AT+CONFIG=1 - transition to CONFIGURATION state.

NOTE: This command also sets lock.

NOTE: This command is depreciated and could be not supported in all future protocols. It is better to use BOOT/CONFIG pin to make module transit to CONFIGURATION state.

7 AEROPROTOCOL

AERO protocol is simple text protocol, that allows fast integration and analysis of tracked aircrafts. AERO messages starts with "#" character and ends with "\r\n" characters. There are 2 types of messages. AIRCRAFT message and STATISTICS message.

7.1 CRC

Frames include CRC value for consistency check. CRC value is calculated using standard CRC16 algorithm and is based on every character in frame starting from '#' and including last comma ','. After calculation, value is appended to frame using hexadecimal coding. Function that calculates CRC is shown below.

```
uint16_t crc16(const uint8_t* data_p, uint32_t length){
uint8_t x;
   uint16_t crc = 0xFFFF;
   while (length--){
      x = crc>>8 ^ *data_p++;
      x ^= x>>4;
       crc = (crc<<8) ^ ((uint16_t)(x <<12)) ^ ((uint16_t)(x <<5)) ^ ((uint16_t)x);
   }
   return swap16(crc);
}
```

7.2 AIRCRAFT MESSAGE

This message describes state vector of aircraft and is send once per second.

#A:ICAO,FLAGS,CALL,SQ,LAT,LON,TRACK,HEA,VELH,VELV,SIGS,SIGQ,FPS,RES,CRC\r\n

#A	Aircraft message start indicator	E.g.
ICAO	ICAO number of aircraft (3 bytes)	3C65AC
FLAGS	Flags (see below)	1
CALL	Callsign of aircraft	N61ZP
SQ	SQUAWK of aircraft	7232
LAT	Latitude in degrees	57.57634
LON	Longitude in degrees	17.59554
ALT	Pressure altitude in feets	5000
TRACK	Track of aircraft in degrees <0,360>	35
VELH	Horizontal velocity of aircraft in knots	464
VELV	Vertical velocity of aircraft in ft/min	-1344
SIGS	Signal strength in mV	840
SIGQ	Signal quality in mV	72
FPS	Number of raw MODE-S frames received from aircraft during last second	5
RES	Reserved for future use	
CRC	CRC16 (described in CRC section)	2D3E

Tab. 6 Detailed message description.

Value	FLAG type	Description
0x0001	PLANE_ON_THE_GROUND	The aircraft is on the ground
0x0002	PLANE_IS_MILITARY	The aircraft is military object
0x0004	PLANE_ALTITUDE_BASED_ON_GNSS	The aircraft has no pressure altitude, but altitude based on GNSS
0x0100	PLANE_UPDATE_ALTITUDE	During last second altitude of this aircraft was updated
0x0200	PLANE_UPDATE_POSITION	During last second position (LAT & LON) of this aircraft was updated
0x0400	PLANE_UPDATE_TRACK	During last second track of this aircraft was updated
0x0800	PLANE_UPDATE_VELO_H	During last second horizontal velocity of this aircraft was updated
0x1000	PLANE_UPDATE_VELO_V	During last second vertical velocity of this aircraft was updated

Tab. 7 Flags description.

If data of any field of frame is not available, then this field is empty. For example:

#A:4CA948,300,,2122,52.99750,13.76526,37000,169,442,0,814,72,3,,6F1C\r\n

#A:424313,,,2362,52.43431,14.84535,37000,65,456,0,806,61,0,,6843\r\n

NOTE: SIGS and SIGQ fields are updated based on raw MODE-S frames. They are calculated from frames received in last second. If there were no receiver frames (FPS=0), those fields will not be updated.

NOTE: SIGS is measured based on analog RF signal. This signal has DC offset of about 700mV.

7.3 MODE-S RAW FRAMES

Raw MODE-S frames. Frames starts with '*' and ends with ';'. Frame contains 7 (short) or 14 (long) bytes of information encoded in hexadecimal format.

Examples of raw MODE-S frames.

Long frame	*8D4CA7E858B9838206BA422BBD7B;\r\n
Short frame	*5D4B18FFFC710B;\r\n

7.4 MODE-AC RAW FRAMES

Raw MODE-AC frames. Frames starts with '*' and ends with ';'. Frame contains 12 bits (3 nibbles) + 1 bit of information encoded in hexadecimal format.

Examples of raw MODE-AC frames.

*6610;\r\n
*E3E0;\r\n

First 3 nibbles contain information of pulses of MODE-AC frames in the following order:

C1, A1, C2, A2, C4, A4, B1, D1, B2, D2, B4, D4

Last nibble indicate presence of SPI pulse:

0 - SPI pulse not presented in frame

1 - SPI pulse present in frame

NOTE: It is impossible to reliably distinguish between MODE-A and MODE-C frames based only on received signal on 1090MHz.

7.5 EXTENDED RAW FRAMES

When extended raw mode is enabled, raw frames are outputted with additional data. Format of that frame is as follows:

*RAW_FRAME;(SIGS,SIGQ,TS)\r\n

Var.	Description	E.g.
SIGS	Signal strength in mV	840
SIGQ	Signal quality in mV	72
TS	Timestamp for multialteration. Time from last PPS pulse in hex format	20CB3

Tab. 8 Extended messages description.

NOTE: To use multilateration TS value must be calibrated using calibration value from statistics message.

NOTE: TS field is available when precise PPS signal from GNSS source is applied to module to 1PPS pin.

7.6 STATISTICS MESSAGE

This message contains some useful statistics about operation of module. Format of that frame is shown below:

#S:CPU,RES,RES,FPSS,FPSAC,TSCAL,CRC

#S	Statistics message start indicator	E.g.
CPU	CPU load in %	12.1
RES	Reserved for future use	
RES	Reserved for future use	
FPSS	Number of MODE-S frames received in last second	3
FPSAC	Number of MODE-A or MODE-C frames received in last second	35
TSCAL	Calibration value for TS field in raw frames	13999415
CRC	CRC16 (described in CRC section)	2D3E

Tab. 9 Statistics message description.

NOTE: TSCAL field is available when precise PPS signal from GNSS source is applied to module to 1PPS pin.

7.7 CALIBRATION OF RAW FRAMES

To get precise time, TS field from raw frames must be calibrated using TSCAL field from statistics message. This allow obtaining precise time which have passed between most recent PPS pulse and reception of that particular frame.

$$TS_{CALIB}[ns] = \frac{TS}{TSCAL}\frac{1s}{10^9}$$

8 MINIMUMINTEGRATION RECOMMENDATION

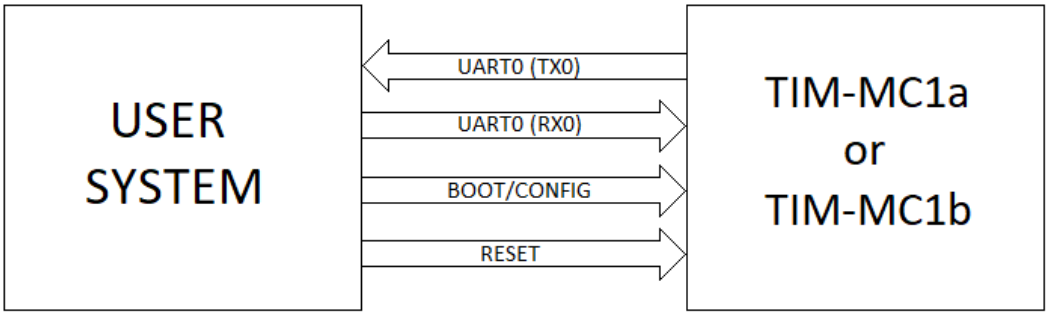


Fig. 4 Simplest module integration with user MCU.

This document is subject to change without notice. For technical questions, contact: support@aerobits.pl

9 MECHANICAL DRAWING

All dimensions in mm.

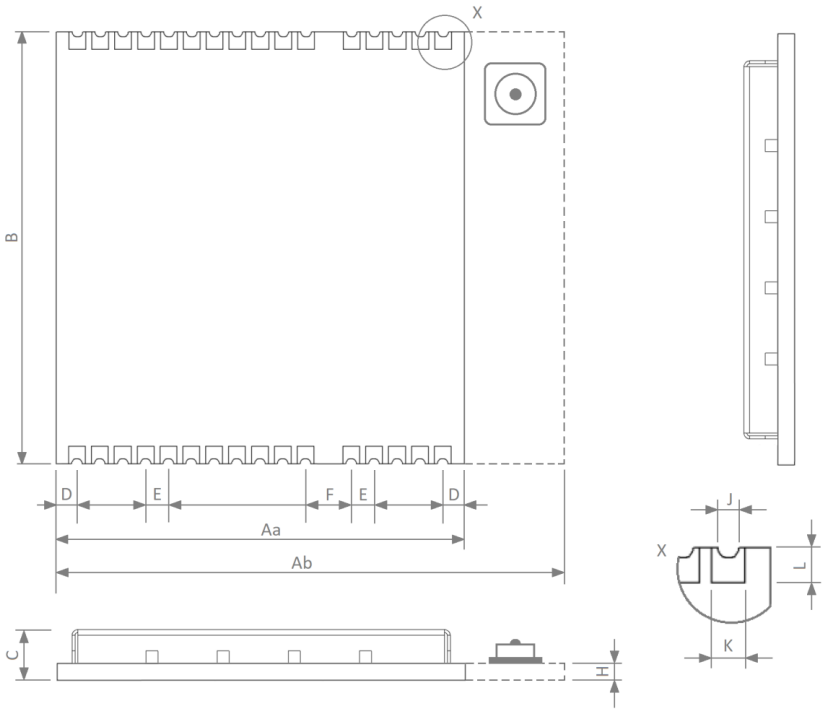


Fig. 5 Dimensions.

Symbol	Min.(mm)	Typ.(mm)	Max.(mm)
Aa (TIM-MC1a)	17.9	18.0	18.1
Ab (TIM-MC1b)	22.4	22.5	22.6
B	18.9	19.0	19.1
C	2.55	2.65	2.75
D	0.9	1.0	1.1
E	0.9	1.0	1.1
F	1.9	2.0	2.1
H	0.6	0.7	0.8
J	0.4	0.5	0.6
K	0.6	0.7	0.8
L	0.7	0.8	0.9

Tab. 10 Dimensions and tolerances.

10 RECOMMENDED LAYOUT

All dimensions in mm.

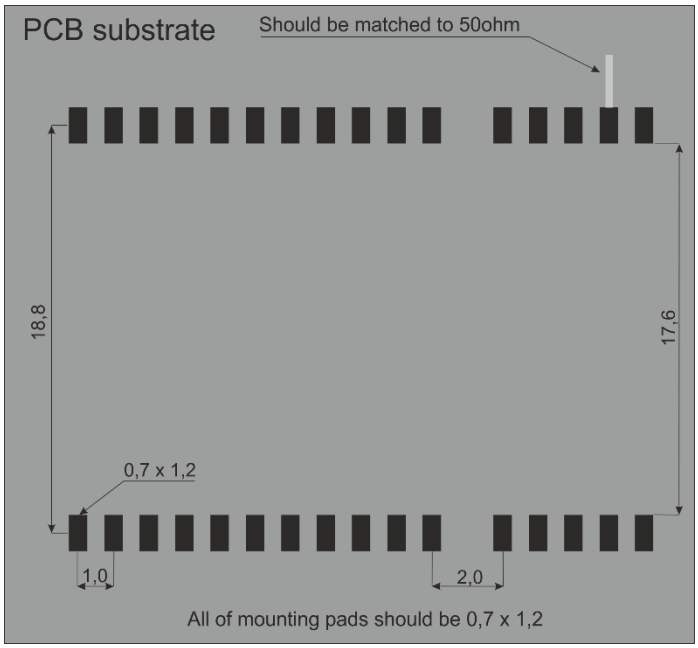


Fig. 6 Recommended footprint.

NOTE: In case of TIM-MC1a the RF input (pad 18) should be matched to 50ohm.

11 REVISION HISTORY

DATE	REVISION	CHANGES
10-May-2018	1	Initial release.
24-May-2018	2	Corrected inconsistencies

Tab. 11 Document revision history.

Please read carefully

Information contained in this document is provided solely in connection with **Aerobits** products. **Aerobits** reserves the right to make changes, corrections, modifications or improvements to this document, and to products and services described herein at any time, without notice. All **Aerobits** products are sold pursuant to our own terms and conditions of sale. Buyers are solely responsible for the choice, selection and use of the **Aerobits** products and services described herein, and **Aerobits** assumes no liability whatsoever, related to the choice, selection or use of **Aerobits** products and services described herein. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third party products or services, it shall not be deemed a license granted by **Aerobits** for use of such third party products or services, or any intellectual property contained therein or considered as a warranty covering use, in any manner whatsoever, of such third party products or services or any intellectual property contained therein.

UNLESS OTHERWISE SET FORTH IN AEROBITS TERMS AND CONDITIONS OF SALE, AEROBITS DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO USE AND/OR SALE OF AEROBITS PRODUCTS INCLUDING, WITHOUT LIMITATION, IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. UNLESS EXPRESSLY APPROVED IN WRITING BY AN AUTHORIZED AEROBITS REPRESENTATIVE, AEROBITS PRODUCTS ARE NOT RECOMMENDED, AUTHORIZED OR WARRANTED FOR USE IN LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS, NOR IN PRODUCTS OR SYSTEMS WHERE FAILURE OR MALFUNCTION MAY RESULT IN PERSONAL INJURY, DEATH, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE.

Information in this document supersedes and replaces all previously supplied information.
© 2018 Aerobits - All rights reserved

www.aerobits.pl