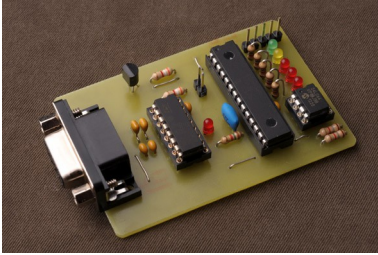# PICthermo (firmware v3.1)

PICthermo is a thermometer for the TinyTrak4 with APRS object generator. It connects to the TinyTrak4 and can act as a simple weather-station measuring the ambient temperature and/or as an object/item/message/bulletin generator for the APRS network.

Configuration of the TinyTrak4

PICthermo can be connected to any port of the TinyTrak4. Put the used port in TEXT mode (AMODE or BMODE) and use 9600 baud (ABAUD or BBAUD). The Tinytrak4 must be jumper configured to provide 5V DC to the serial ports.

Configuration of PICthermo

PICthermo can be configured with any terminal program via a COM-port. Use 9600 baud and no flow control. PICthermo must be connected in such a way to the COM-port that 5V DC is supplied to pin 4 (consider programming adapter on last page). When powering up PICthermo, *s for SETUP* should appear in the terminal window. Press s within a second to enter setup mode. If you are not fast enough, PICthermo will boot and you have to cycle the power in order to do a second attempt to enter setup mode. After entering setup mode, the valid commands (between parentheses) appear in the terminal window:

(**f**)lag, (**A.I.Q.Y**)obj, (**aiqy**)obj, (**t**)ime, t(**o**)ff, (**v**)er, (**w**)X, (**e**)nd

(**f**)lag. Push **f** to adjust the flags of PICthermo. The current value appears on the display, e.g. Fxx, where xx is a byte in hexadezimal format in the range of 00 .. FF. The value of the byte can be computed nibble-wise by checking which bits of the byte are set.

(**f**)lag	/	/	ADC ref. *	Temp sensor	/	/	No LED	Time unit
	8	4	2	1	8	4	2	1

HIGH nibble   LOW nibble

* PCB 3.0 and up

ADC reference :   **0** : Supply voltage   **1** : LM336 5-Volt-reference (needs 12V)

Temp sensor :   **0** : onboard sensor   **1** : external sensor

No LED :   **0** : LEDs are active   **1** : LEDs are inactive

Time unit :   **0** : 0,1 seconds   **1** : 10 seconds

If you want to use the external temperature sensor, no LEDS and 0,1 seconds time unit, then the corresponding value of the byte is **12**:

$0 \cdot 8 + 0 \cdot 4 + 0 \cdot 2 + 1 \cdot 1 = \mathbf{1}$	$0 \cdot 8 + 0 \cdot 4 + 1 \cdot 2 + 0 \cdot 1 = \mathbf{2}$
HIGH nibble	LOW nibble

(**A.I.Q.Y**)obj. Push a capital letter **A**, **B**, **C**, …, **Z** to view/change one of the 26 available memories for APRS objects, items, messages, bulletins, etc. Push ENTER to abort or type new text. If you type new text, the last chatacter must be a $ sign. If you enter a mistake, use backspace, finish with $.

(**aiqy**)obj. Whether an object is transmitted or not depends on the flags **a**, **i, q**, and **y**. The object is sent if the corresponding bit is set.

(**a**)obj	H	G	F	E	D	C	B	A
	8	4	2	1	8	4	2	1
(**i**)obj	P	O	N	M	L	K	J	I
	8	4	2	1	8	4	2	1
(**q**)obj	X	W	V	U	T	S	R	Q
	8	4	2	1	8	4	2	1
(**y**)obj	/	/	/	WX	/	/	Z	Y
	8	4	2	1	8	4	2	1

HIGH nibble   LOW nibble

Attention : Objects **Y** and **Z** are special objects. In order to be sent **pin RB5** on the microcontroller must additionally be grounded (useful as alarm function).

WX   :   **0** : temperature not sent   **1** : temperature is sent

If you want to send the objects H, F, E, C, B, A then the corresponding value of the byte stored in **a** is **D7**:

$1 \cdot 8 + 0 \cdot 4 + 1 \cdot 2 + 1 \cdot 1 =$ **D**	$0 \cdot 8 + 1 \cdot 4 + 1 \cdot 2 + 1 \cdot 1 =$ **7**
HIGH nibble	LOW nibble

Objects A to X are stored in the 24LC64 EEprom chip. Objects Y and Z are stored in the internal Eeprom of the PIC16F886.

(**t**)ime. Push **t** to adjust the time interval between objects. The current value appears on the display, e.g. txx, where xx is a byte in hexadezimal format in the range of 01 .. FF.

Together with time unit, it is possible to adjust the time interval in the following range :

| Time unit	Time interval			
0,1 seconds	t01 : 0,1 seconds	txx : xx · 0,1 s		tff : 25,5 seconds
10 seconds	t01 : 10 seconds	txx : xx · 10 s		tff : 2550 seconds

t(**o**)ff. Push **o** to adjust the offset of the temperature sensor. The current value appears on the display, e.g. oxx, where xx is a byte in hexadezimal format in the range of 01 .. FF. The default value is 0A. Two units up drops the temperature by approximatively 1 °C.

(**w**)X. Push **w** to display or modify the position of the weather station. The 18 character string must match the APRS protocol. Adjust as needed. E.g. : **4931.36N/00554.88E**. Enter a $ as last character.

(**v**)er. Push **v** to see the current firmware version.

(**e**)nd. Push **e** to exit setup and start the boot routine.

Schematic (PCB 3.0)
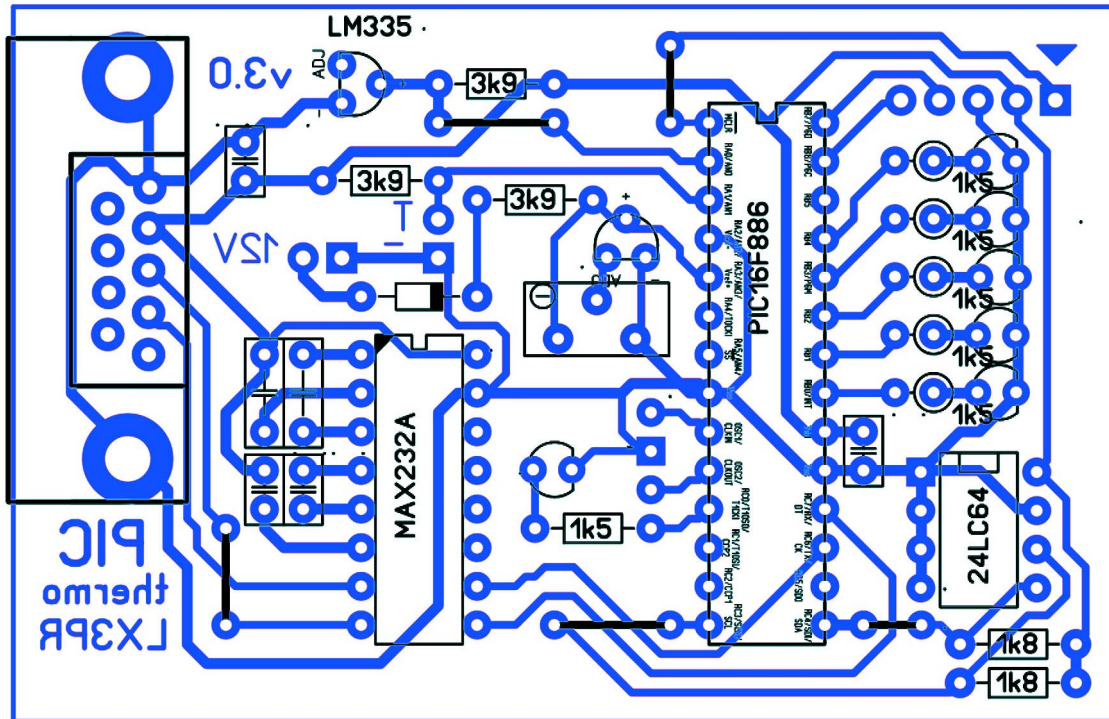
All capacitors are ceramic 100 nF. Use 1/4W carbon resistors. LEDs are low current type. Use a 4 MHz ceramic resonator on the 3 pins near the center LED. Five wire bridges have to be installed.

Adjustment of the optional ADC voltage reference: Supply 12V DC to the marked 2x1 header near the DB-9 connector. Adjust the potentiometer to obtain 5,000 V between GND and + of the LM336 (near the corresponding 3k9 resistor).
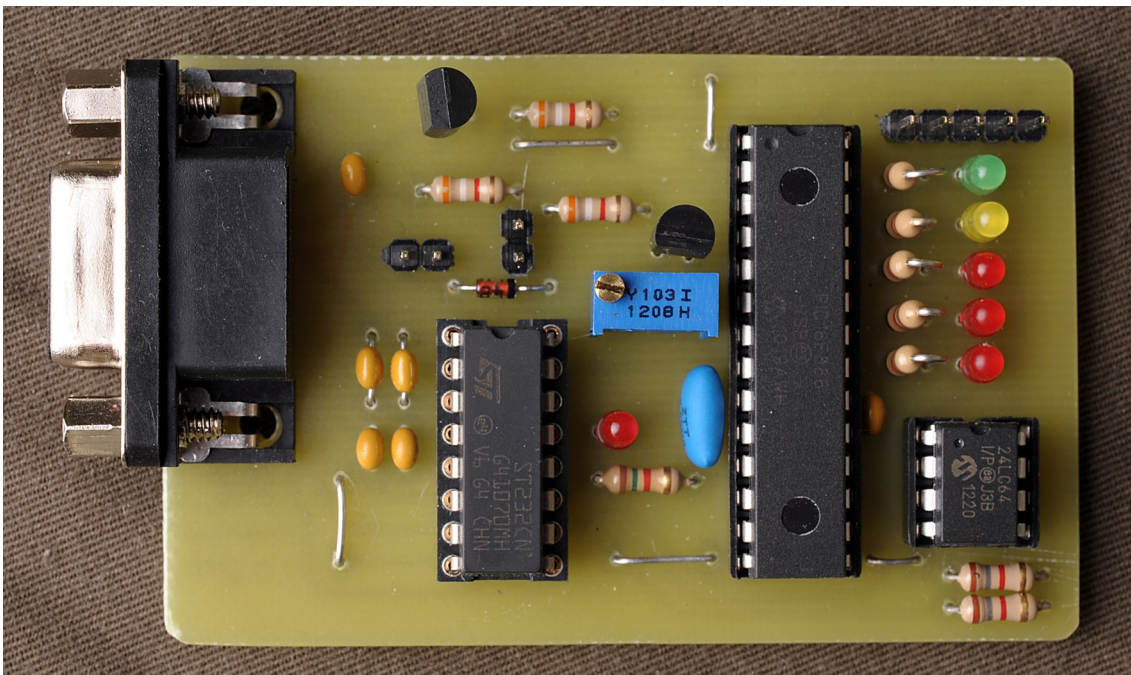

The 5-pin-header can be used for upgrading the firmware of the PIC16F886 via a Microchip PICKIT.

Part list (PCB 3.0)

1	Microchip PIC16F886	6	resistor 1,5 kΩ
1	MAX232 (100 nF type)	2	resistor 1,8 kΩ
1	Microchip 24 LC64 EEprom	3	resistor 3,9 kΩ
1	LM335 temperature sensor	6	ceramic capacitor 100 nF
1	LM336 5-Volt-reference	1	Potentiometer 10 kΩ
1	ceramic resonator 4 MHz	1	LED 3mm green
1	header 9x1	1	LED 3mm yellow
1	DB-9 connector female US-type	4	LED 3mm red
1	IC socket 28 pin small	1	IC socket 16 pin
1	IC socket 8 pin		

Programming adapter
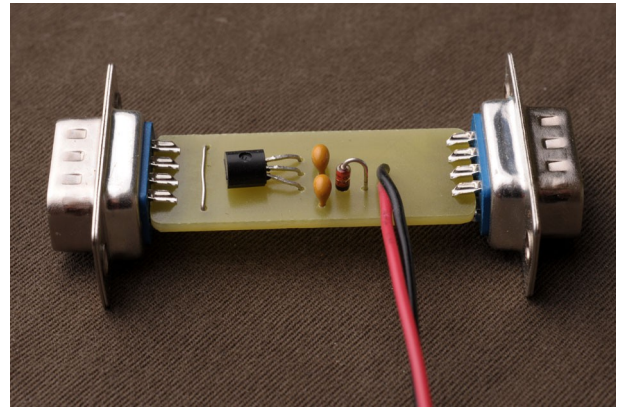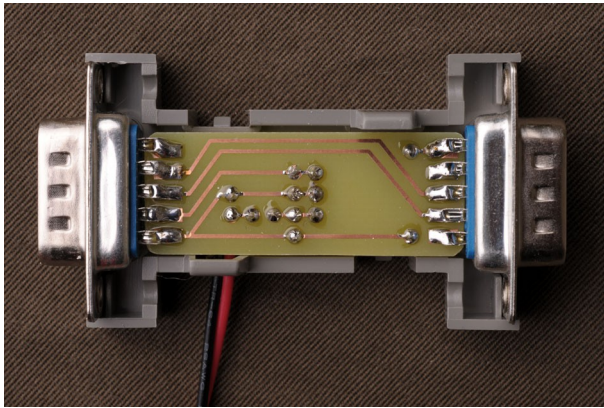
The pictures show a simple programming adapter for PICthermo, providing 5V DC to pin 4 of PICthermo via a 78L05 voltage regulator, two 100nF capacitors and a protection diode. PICthermo is connected to the DB-9 (male) connector near the power cables. The adapter can then be connected either via a null-modem cable to a PC or directly to a portable device (e.g. HP200LX) for programming in any terminal program set at 9600 baud.

Contact

LX3PR, email : lx3pr@physique.lu