

ATCC2707

Advanced Thermoelectric Cooler
Controller 27V/7A

DATASHEET

Rev2.1
July 2012



sysacom

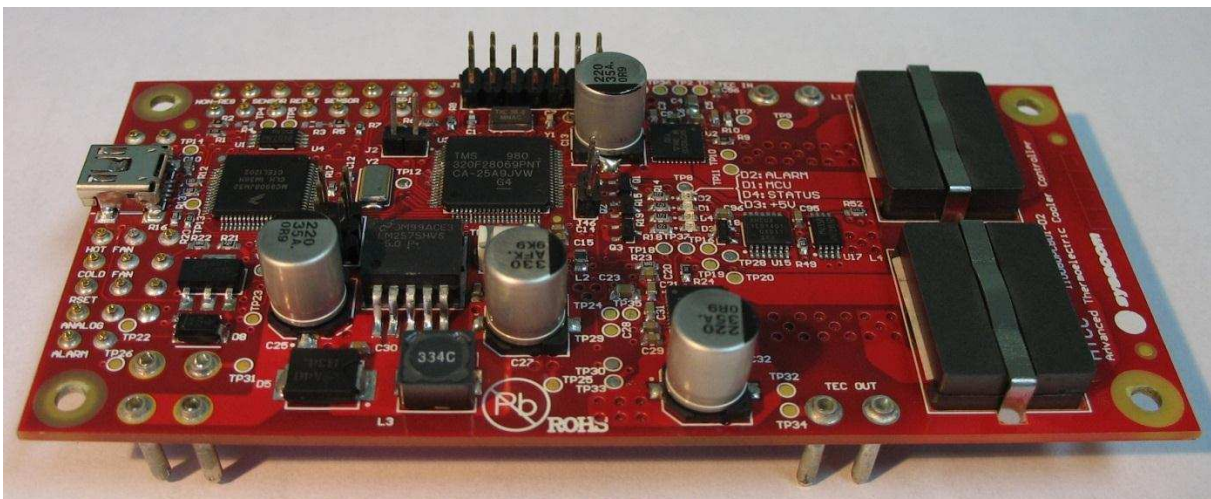
www.sysacom.com

Features

- Control of thermoelectric modules up to 7A, from 7 to 27V.
- Stand-alone module.
- Bi-directional control (Hot and Cold).
- Real-time data monitoring.
- Compatible with NTC sensors.
- Supports fans, with or without tachometer.
- PID controller.
- Alarms detections.
- Planar inductors.
- USB and SPI interfaces.
- Easy to configure via PC (USB) or external MCU (SPI).
- Protections circuits : Thermals protections, over-current, shoot-through.
- 600kHz - PWM control with ramp-up and saturation of command.
- Full brick size (2.2"x4.6")

Applications

- Air conditioner
- Liquid chillers
- Laser diode cooler
- CCD cooler



Overview

The ATCC is especially made to drive Peltier cells (thermoelectric cells). The Peltier cells are made to transfer thermal energy from a hot plate (not thermally regulated) to a cold plate (thermally regulated) or vice versa. The ATCC regulates the temperature of a plate by controlling the current level applies to Peltier cells. It measures the temperature of the thermally regulated plate, compares it to a set point and applies the regulation transfer function, which is a fully configurable PID.

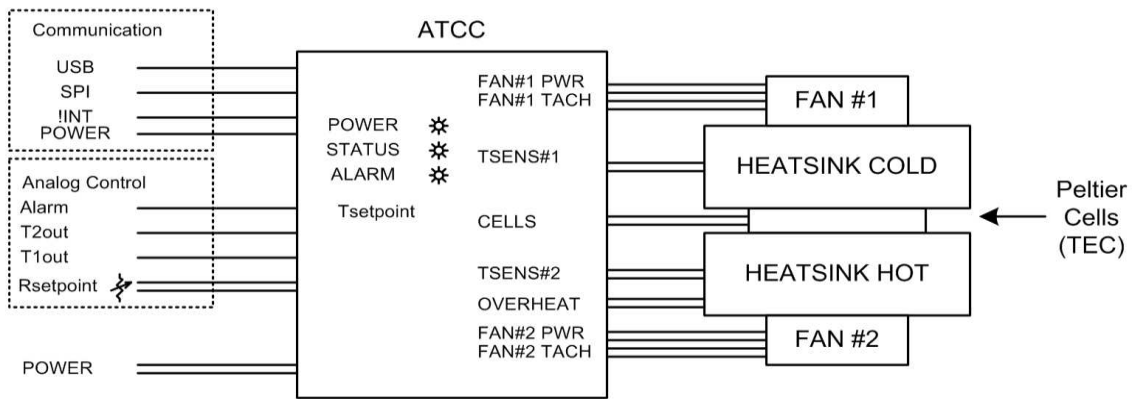


Figure 1 - Overview of ATCC external connections

Table of Contents

1. SPECIFICATIONS.....	8
2. DESCRIPTION.....	9
2.1) CONTROL OF THERMOELECTRIC MODULES UP TO 7A, FROM 7 TO 27V.....	9
2.2) STAND-ALONE MODULE.....	9
2.3) BI-DIRECTIONAL CONTROL (HOT AND COLD).....	9
2.4) REAL-TIME DATA MONITORING.....	10
i. Using HMI.....	10
a. Control of temperatures and command.....	10
b. Control of alarms.....	11
ii. In embedded mode.....	11
a. Control of temperatures.....	11
b. Control of alarms.....	11
c. Control of status of controller.....	11
2.5) COMPATIBLE WITH NTC SENSORS.....	11
2.6) SUPPORTS FANS, WITH OR WITHOUT TACHOMETER.....	12
2.7) PID CONTROLLER.....	12
2.8) ALARMS DETECTIONS.....	13
i. Alarms that stop system and latch fault.....	13
a. Overcurrent.....	13
b. Under voltage.....	13
c. Thermostat fault.....	13
d. Thermal protection of drives.....	13
ii. <i>Alarms that stop system but don't latch fault</i>	13
iii. <i>Alarms that don't stop system</i>	14
2.9) USB AND SPI INTERFACES.....	14
2.10) EASY TO CONFIGURE VIA PC (USB) OR EXTERNAL MCU (SPI).....	15
2.11) PROTECTIONS CIRCUITS: THERMALS PROTECTIONS, OVER-CURRENT, SHOOT-THROUGH.....	16
2.12) 600kHz - PWM CONTROL WITH RAMP-UP AND SATURATION OF COMMAND.....	17
3. TYPICAL PERFORMANCE CHARACTERISTICS.....	18
4. COMMUNICATION PROTOCOL.....	20
4.1) PROTOCOL PERSPECTIVE.....	20
4.2) PROTOCOL FUNCTIONS.....	20
4.3) CONSTRAINTS.....	21

4.4)	ASSUMPTIONS AND DEPENDENCIES	21
4.5)	PROTOCOL DEFINITION.....	21
iv.	Protocol.....	21
v.	Header.....	22
vi.	Data.....	22
vii.	Footer.....	23
4.6)	COMMANDS	23
i.	Configuration commands	24
a.	SetFanStatus command.....	24
b.	GetFanStatus command.....	24
c.	SetFanAlarmStatus command.....	25
d.	GetFanAlarmStatus command.....	25
e.	SetTachStatus command.....	26
f.	GetTachStatus command.....	26
g.	SetFanNominalSpeed command.....	27
h.	GetFanNominalSpeed command.....	27
i.	SetThermostatStatus command.....	28
j.	GetThermostatStatus command.....	28
k.	SetNonRegulatedLimitTemp command.....	28
l.	GetNonRegulatedLimitTemp command.....	29
m.	SetRegulatedLimitTemp command.....	29
n.	GetRegulatedLimitTemp command.....	30
o.	SetTECCurrent command.....	30
p.	GetTECCurrent command.....	30
q.	SetNTCRegParameters command.....	31
r.	GetNTCRegParameters command.....	31
s.	SetNTCNonRegParameters command.....	31
t.	GetNTCRegParameters command.....	32
ii.	Parameters setting commands.....	32
a.	SetTempSource command.....	32
b.	GetTempSource command.....	33
c.	SetPIDGainValues command.....	33
d.	GetPIDGainValues command.....	33
e.	UpdateParameters command.....	34
iii.	Data monitoring commands.....	34
a.	GetTemperatures command.....	34
b.	GetAlarmStatus command.....	35
c.	GetCommand command.....	35
iv.	Upgrade and device information commands.....	36

a.	GetSerial command	36
b.	SetSerial command	36
c.	GetDeviceInfos command	37
d.	HardwareReset command	37
e.	UpgradeInit command	37
f.	UpgradeSRec command	38
g.	UpgradeEnd command	38
h.	StopControl command	39
i.	RestartControl command	39
j.	StartConnect command.....	39
k.	GetControlState command.....	40
5.	MECHANICAL	41
6.	WARRANTY.....	44

Table of figures

Figure 1 - Overview of ATCC external connections	3
Figure 2 - a) Positive sense of current (MOSFET 1 and MOSFET 4 are closed).....	9
Figure 3 - Overview of the two complementary PWM @ 600kHz for a null current (Duty cycle = 50%)	10
Figure 4 - Overview of PID controller regulation loop.....	12
Figure 5 - Overview of plates temperatures process.....	14
Figure 6 - Interactions between ATCC and SPI connections	15
Figure 7 - Overview of HMI interface to configure ATCC and monitor temperatures in real time	16
Figure 8 - Protocol perspective	20
Figure 9 : Protocol sections.....	21
Figure 10 : Header definition figure.	22
Figure 11 : Footer definition figure.....	23
Figure 12 : Data definition detail figure.....	23

Revision History

Revision Number	Revision Date	Description of changes
01.00	10/05/2012	First release.
02.00	29/06/2012	New revision to respect Sysacom datasheet formatting. Incorporation of protocol part (section 4).
02.01	06/07/2012	Add precisions about description of features.

1. Specifications

ATCC2707 main board		
Electrical		
	Recommended settings	Absolute maximum ratings
Supply/Output Voltage (Vdc)	7 to 27	6.2 to 30
Supply/Output Current (A)	0 to +/-7	+/-8
Supply/Output Power	Up to 190W	Up to 240W
No-Load Supply Current (A)	0.4	1
Alarm output	Up to 40V/0.5A	60V/3A
Fans output current (A)	Up to 0.5	1
Performance		
PWM frequency (kHz)	600	
PWM precision (%)	1	
PWM variation speed (%/s)	1	
Temperature Error (°C)	+/- 0.1	
Temperature Resolution (°C)	0.03	
Fan speed (RPM)	Up to 20000	
Environment		
Operating Temperature (°C)	-20 to 55	
Physical		
Unit Size (in)	2.2 x 4.6 x 0.876	
(mm)	55.8 x 116.84 x 22.25	
Weight (g)	93	
(lb)	0.21	

2. Description

2.1) Control of thermoelectric modules up to 7A, from 7 to 27V

Purpose of the ATCC is its capability to be fully compatible to all Thermo Electric Controller (TEC) modules from 7 to 27V. Current that goes through the TEC can also be configured through an HMI in order to place any TEC assembly up to 7A. A current sensor controls the current going to the TEC, and gives a feedback to MCU to adjust max current available.

2.2) Stand-alone module

ATCC is configured through its HMI interface, which is used to set all parameters depending of the TEC assembly used. Once all parameters are set, they are stored in an external EEPROM, in order to make the circuit operate in stand-alone mode. User can choose to set the set point (i.e. temperature to reach) by an external potentiometer. Two analog outputs are furnished in order to watch temperatures of the both cells (Hot and Cold). Also, an open-drain connector enable user to plug an alarm output if a fault condition is detected (see section 2.8 for alarms description).

2.3) Bi-directional control (Hot and Cold)

TEC assemblies use Peltier Effect to transfer temperature between its plates. This means that if the current goes in a «positive» way, the thermal transfer will also be positive (from hot plate to cold plate). But, if the current goes in the other way, the thermal transfer will be in the other direction. Two opposite PWM control gates of bridge that drives the TEC cell. This controls the amount of current going through the cells.

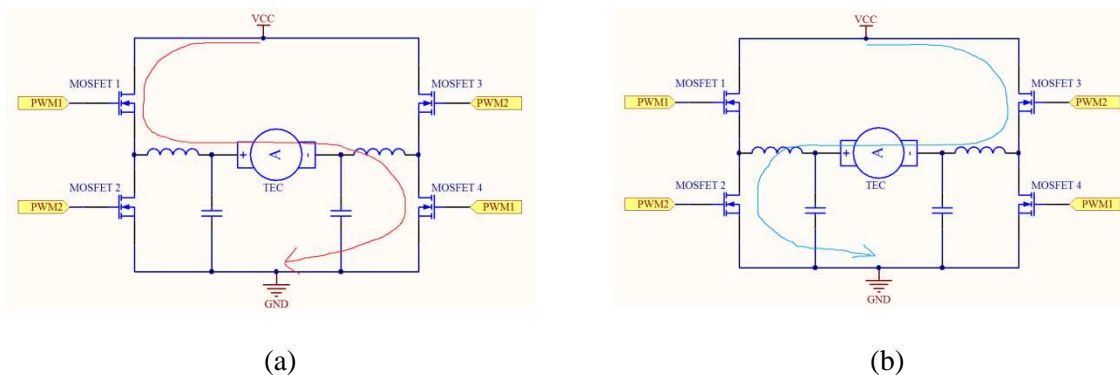


Figure 2 - a) Positive sense of current (MOSFET 1 and MOSFET 4 are closed)

b) Negative sense of current (MOSFET 2 and MOSFET 3 are closed)

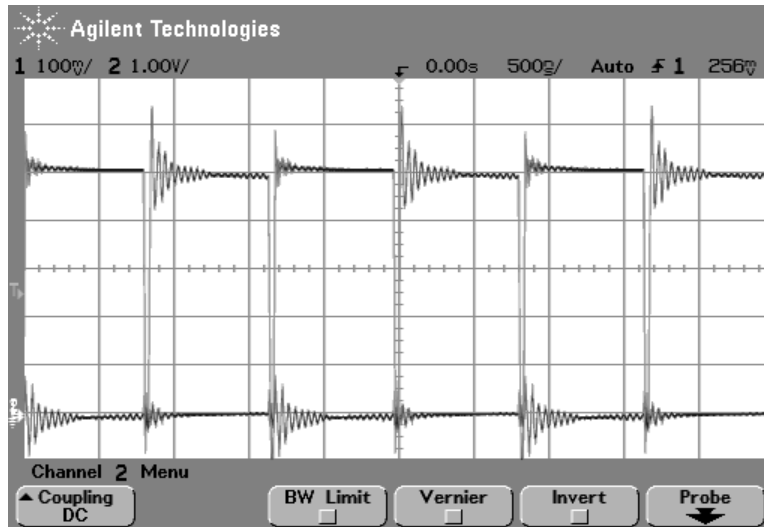


Figure 3 - Overview of the two complementary PWM @ 600kHz for a null current (Duty cycle = 50%)

2.4) Real-time data monitoring

Real-time data monitoring can be done by two ways, either by the way of Human Machine Interface (HMI), or with the outputs provided onto the board.

i. Using HMI

An HMI has been developed by Sysacom in order to set parameters, but also to view evolution of system in real-time.

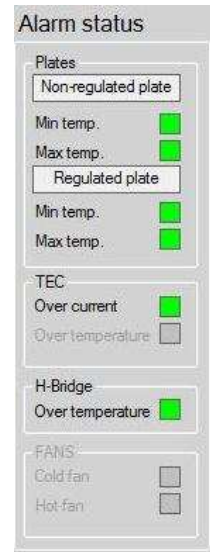
a. Control of temperatures and command

Data monitoring	
Non-regulated plate temperature	22.61 °C
Regulated plate temperature	05.64 °C
TEC control output	
TEC current	-2.182617 A
Command	33 %

Temperatures of both plates are monitored in real-time, but also command and current that go through the TEC cell. This allow user to view evolution of temperatures, depending of command.

b. Control of alarms

Many alarms are used in order not to damage TEC cells that are controlled by ATCC. Alarms can be sourced by temperatures of plates; min/max threshold can be set by user. Also, two alarms are provided at the TEC, overcurrent detection, and a trip of thermostat. An alarm indicates a fault of temperature of the H-Bridge; and finally two alarms indicate status of fans. See section 2.8 for more indications.



The screenshot shows an 'Alarm status' panel with the following indicators:

- Plates:**
 - Non-regulated plate: Min temp. Max temp.
 - Regulated plate: Min temp. Max temp.
- TEC:**
 - Over current:
 - Over temperature:
- H-Bridge:**
 - Over temperature:
- FANS:**
 - Cold fan:
 - Hot fan:

ii. In embedded mode

a. Control of temperatures

Temperatures of both TEC cell can be monitored in real-time using two DAC output that provide an analogic signal proportional to the temperature.

b. Control of alarms

An alarm led is an external indicator of one of the fault that can be triggered at the TEC. If this alarm LED activate, this signifies that one of the fault conditions has been encountered. Refer to X.Y for more precisions about alarms.

c. Control of status of controller

Two other LED, MCU and STATUS allows user to verify that controller is still active. MCU running LED indicates that ATCC is running. Status LED indicates that ATCC is acting, furnish some current to the TEC output.

2.5) Compatible with NTC sensors

Two outputs are provided in order for user to plug its own NTC sensors. They can then be configured through the HMI.

All kind of sensors will be supported in further revisions.

2.6) Supports fans, with or without tachometer

Two fans outputs are provided in order to cool down the cells. They can be either activated or not (settings through HMI). Also, they can be equipped of tachometers, in order to detect a failure of the fan(s), and an alarm can then be activated if the fan has a problem (if the RPM is below 90% of nominal RPM).

2.7) PID controller

The PID implemented is used to regulate control loop, in order to control temperature of cold side of TEC. The fact is that we would like to saturate command send when this one has a too high current for our circuit.

The way it is done is to verify current that goes out of the TEC, and when this current exceed max current; the command is saturated at this value of PWM. Also, the integral sum is saturated at the same value, which permits to continue to accumulate error, and to make a command consequently.

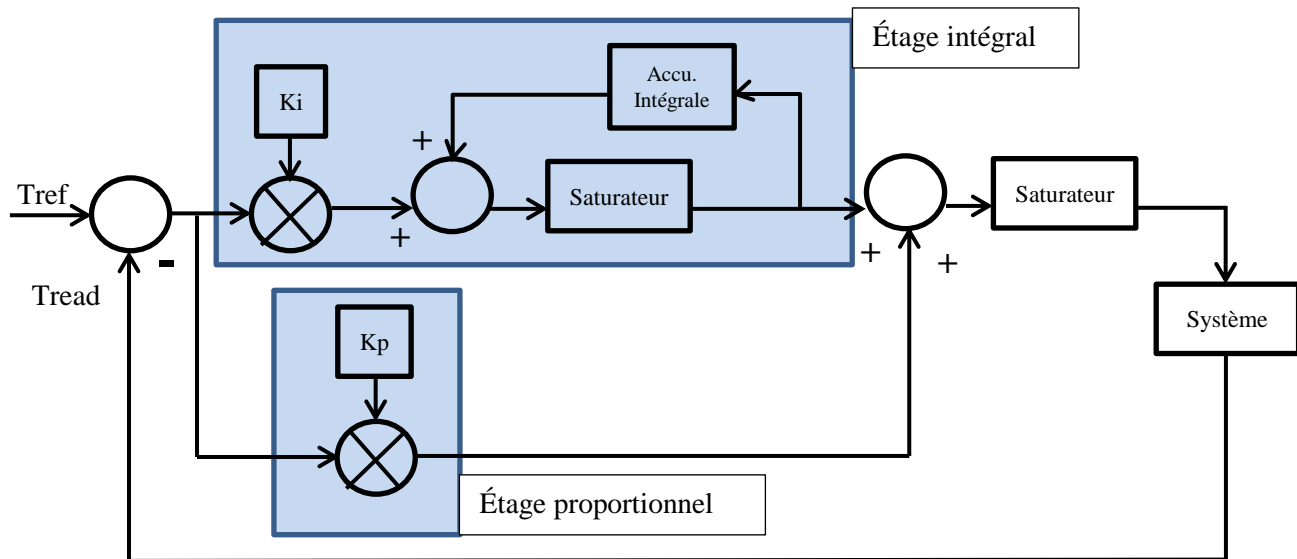


Figure 4 - Overview of PID controller regulation loop

2.8) Alarms detections

As mentioned in the previous section(s), multiple sources of alarms can be triggered by system, and depending of their criticism, they have different behavior.

i. Alarms that stop system and latch fault

These alarms are the most critic alarms. If one of the following alarms occurs, the system is stopped and the fault alarm is latched. This means that user needs to release latch by either HMI interface, or by stopping and restarting supply. If fault is still present, stop system and make verifications before restarting system.

a. Overcurrent

Overcurrent detection occurs at soon as the current reaches overcurrent threshold supported by circuit (8A). **If this fault occurs, verify that there is no short-circuit at the TEC before re-starting system.**

b. Under voltage

Under voltage lock out is a protection that verify if there is no fault in the +5V voltage. This permits to inhibit any trans-conductance of the drive due to a bad logical level coming from PWM. If this fault occurs, there may be a fault in the circuit (such as a short-circuit).

c. Thermostat fault

A thermostat Normally-Closed (NC) can be set onto one of the side of the TEC cell. This can prevent any damage of the cell due to a too high temperature.

d. Thermal protection of drives

Two flags (THDN) are set by the drive in order to indicate that an over temperature condition has occur. When this flag is set, system is stopped, in order to make the temperature drop down.

ii. Alarms that stop system but don't latch fault

These alarms are the alarms that can damage the system, but that can go back alone in safe conditions. These alarms are the plate temperature alarms. Here is an overview of temperature process.

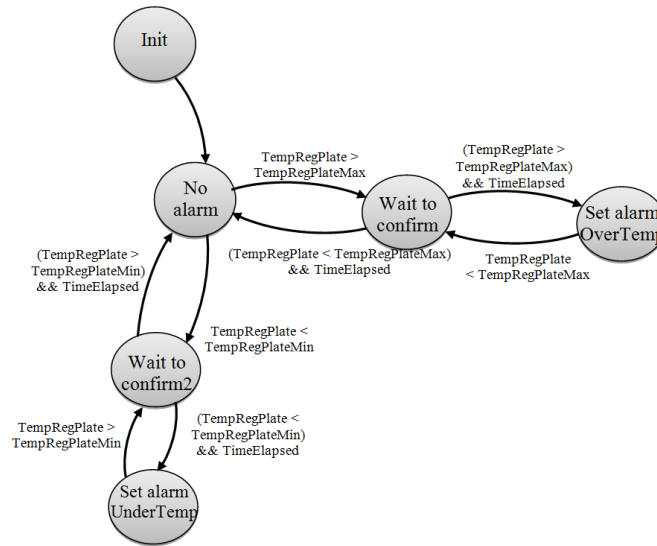


Figure 5 - Overview of plates temperatures process

When one of the critical threshold is reached (either cold or hot), the system falls in «confirmation» mode: if after a certain time (here its 2s), the temperature is still in a critical state, the alarm is trig and the system is stopped. But, as soon as the temperature goes back into the safe range, with confirmation of temperature, alarm is clear, and the system continues to operate.

iii. Alarms that don't stop system

The two fans alarms will trig alarm output, and stop fans but won't stop the system. These are not critical because they will cause another alarm to trig later (such as temperature alarm).

2.9) USB and SPI interfaces.

ATCC has two interfaces that can be used in order to set parameters, but also provide data monitoring. One of them is USB, which can be used directly via a Micro-B cable in order to interface ATCC through HMI. Also, an external MCU can be connected to ATCC via an SPI communication bus. See section 4 for specifications of protocol used.

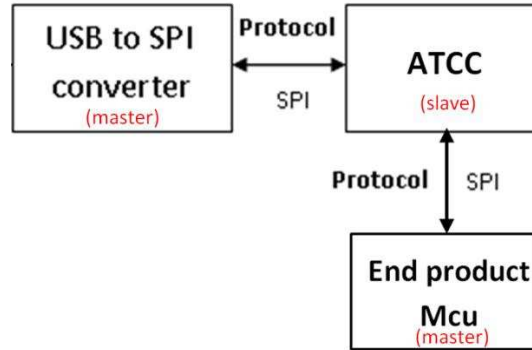


Figure 6 - Interactions between ATCC and SPI connections

2.10) Easy to configure via PC (USB) or external MCU (SPI)

An HMI has been developed by Sysacom in order to easily configure ATCC depending of TEC parameters. There are three types of commands:

- Configurations commands, such as TEC nominal current, presence of fans in the system, tachometers, temperatures limits, sensor parameters; that you set only one time.
- Parameters commands, which are the set point, and settings of PID controller. These are used to be adjusted at the beginning of system.
- Data monitoring, are the data sent in real-time, such as temperature of plates, current and alarms.
- Upgrade, these commands allows an update of ATCC with the latest firmware.

Refer to User Guide for more description.

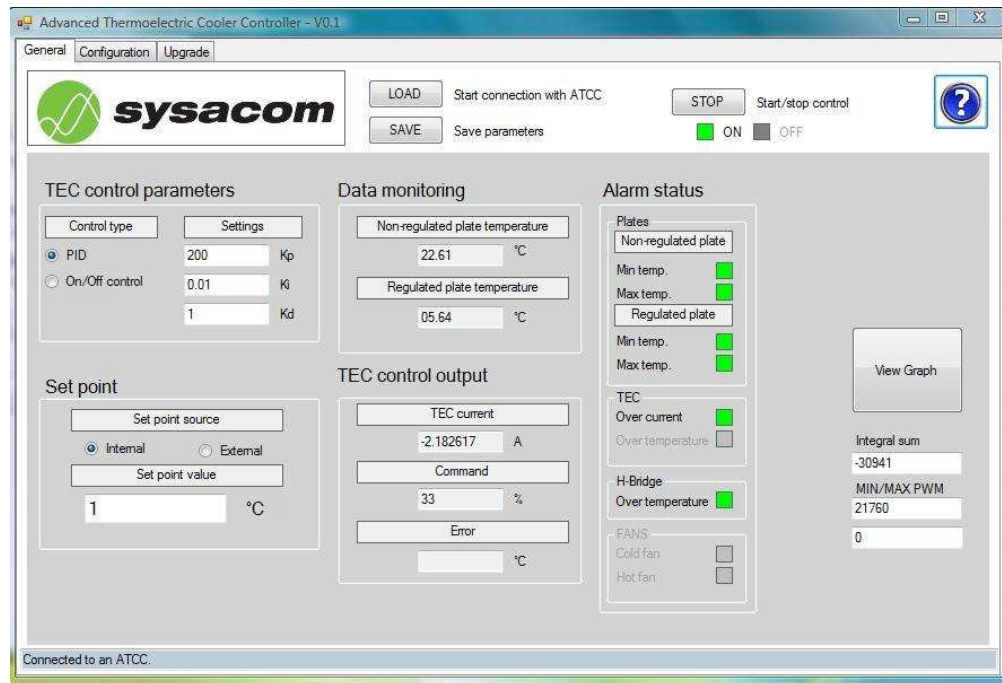


Figure 7 - Overview of HMI interface to configure ATCC and monitor temperatures in real time

2.11) Protections circuits: Thermals protections, over-current, shoot-through

Some of the ATCC sensors are on-board. These sensors are present in order to avoid any damage of the system. These are:

- Thermal flag of drives, which can stop the system if it is set
- Current sensor, which continually monitors current going to the TEC in order to control it.
 - Thresholds are set in order to stop the system in case of over or under current. This means that even if there is a short-circuit at the TEC, the system will stop immediately.
- Voltage supervisor is present in order to detect any bad logical level of +5V. This protection helps to prevent from any leakage that can be produced into the 5V. This is the logical level of PWM, and then can be a cause of shoot-through of the power bridge.

2.12) 600kHz - PWM control with ramp-up and saturation of command

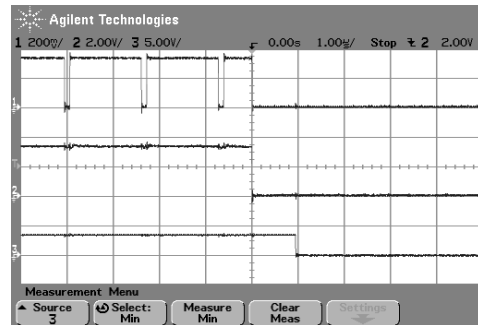
Control of the full bridge is made by a 600 kHz PWM generated by MCU. Precision of increment is 1% of PWM, varying from 5 to 95%. Command is limited in order not to stress too much components. Maximum increment is 1% by ten control process loop, which runs at 1/10s. This means that max increment is 1% by second. Max current is then reach is 50 seconds (50% of PWM). Also, integral gain of PID is saturate in order not to increase too much command, and then fall in instability of system.

3. Typical performance characteristics

Protection circuits

In order to protect circuit and TEC module, different protections are available:

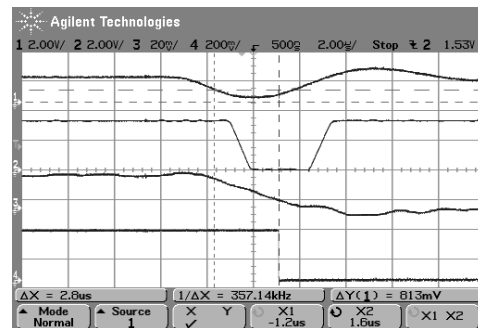
- **TEC current limitation:** Depending of your TEC nominal current, the send command will be saturate as soon as the current is reached.
- **Thermal protections:** Thermal protections are necessary to avoid temperature to reach critical points.
 - The TEC temperatures (non-regulated plate/regulated plate) limits can be software set.
 - The drive possesses a thermal flag indicating over-heating. This indicator immediately stops PWM, and deactivates drive.



We can see here on Ch1: PWM, Ch2: THDN signal, Ch3: Drive enable.

- **Over/Under current:** The min/max (+/-8A) currents are set to stop the system as soon as they are reached. This latches fault in order to board under alarm state.

Here are Ch1: Output of current sensor, Ch2: Output of current supervisor (logical and), Ch3: TEC current measurement (Current probe 100mV/1A), Ch4: Disable drive (Software).

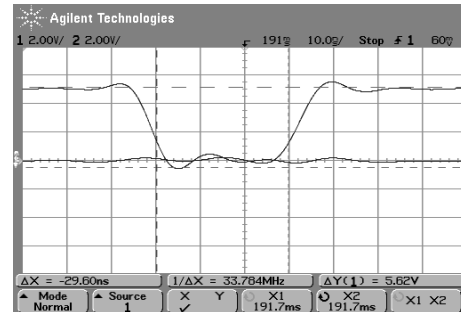


- **Voltage supervisor:** As soon as all circuit is supply by a 5V regulator, we need to be sure that this voltage is always respected. This is done by a 5V on-board voltage regulator.

Drive circuit

- **Pre-programmed dead-bands:** The drive circuit is controlled by two high-speed 600 kHz PWM. Their dead-band is software programmed in order to avoid any trans-conduction between the 2 sides of the bridge.

We can see here the dead-band between the both PWM.

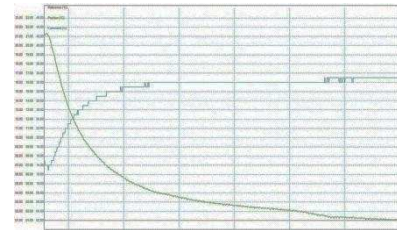


- **Limitation of command:**

In order to avoid a too-quick command, it has been software limited to increase by 1% max. PWM output every second.

Control loop

Precise control can be achieved using a PID. This type of controller can avoid any external perturbation, and can lead to temperature stabilization to 0.1°C from set point.



4. Communication protocol

Purpose of this section is to describe communication protocol used between controller and computer, but also between controller and external end point MCU. This protocol will be used for USB and SPI communication ports.

4.1) Protocol perspective

The protocol is used between computer and controller, by different communications links, as shown in Figure1.

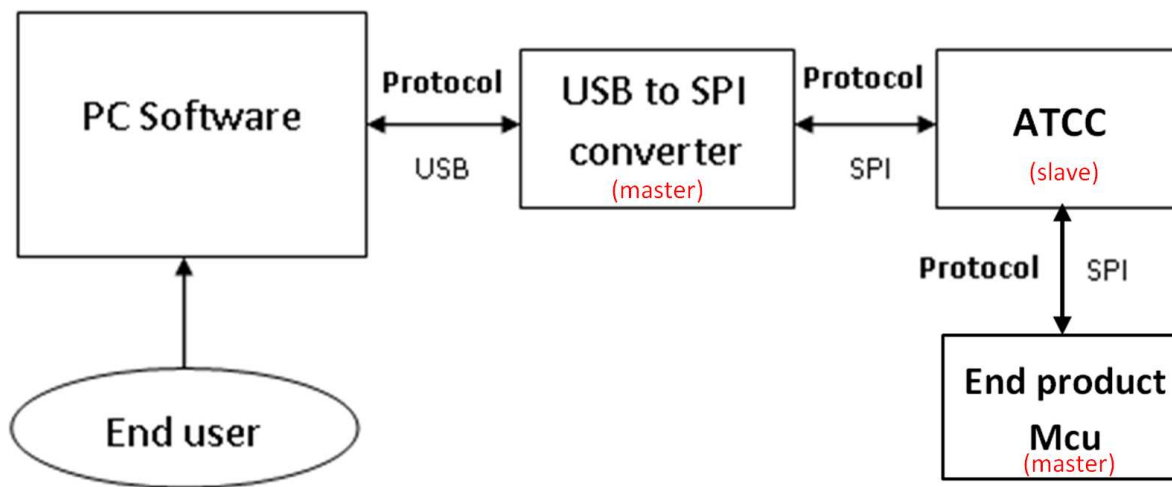


Figure 8 - Protocol perspective

As shown in figure 2, ATCC has two links that use protocol. Both use SPI and peripheral are describe as master. SPI communication is triggered by the master, so all commands are coming from PC software (passing by USB-SPI converter) or from end product MCU. All answers come from ATCC.

4.2) Protocol functions

The protocol is a software layer that bound information according to a set of predetermined command between controller and pc software or end product MCU. The protocol must allow addition of new commands easily. It also has to limit the data overhead in order not to slow down communication. Packet CRC is also present in order to ensure data integrity.

SPI communication is used for communication between controller and its local peripherals.

USB is used to interface a computer.

Notice also that the upgrade of controller firmware is done also by SPI link to computer. A boot loader must be specified in controller

4.3) Constraints

The only constraint is that a process will need to be called iteratively to parse the input buffer and look for incoming command. Also, the protocol uses byte stuffing so buffer shall reflect worst-case situation; more details in following sections.

4.4) Assumptions and dependencies

We assumed that all buffers sizes are calculated so that the larger packet, including byte stuffing, doesn't overflow. We also assume that either a process or a task is called fast enough to manage incoming command and answer efficiently. Code reference for binary protocol in C++ is already made for pc platform.

4.5) Protocol definition

iv. Protocol

It is a binary protocol made up of three major sections, a header, a data section and a footer section.

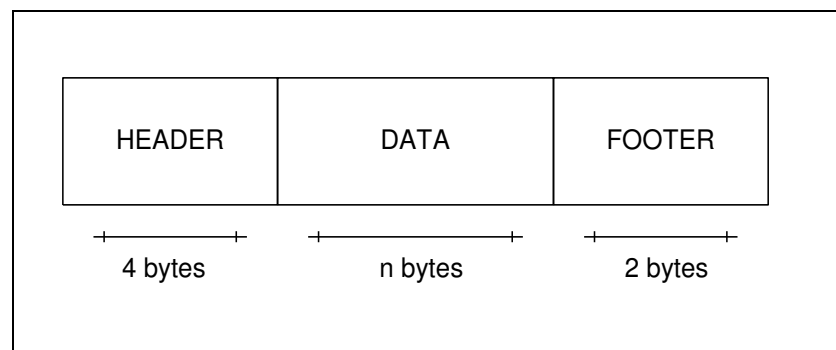


Figure 9 : Protocol sections.

The minimum packet, a simple command or an acknowledge answer, is 8 bytes, because minimum data is just a command, of 2 bytes.

v. Header

The header contains 4 bytes made of the following pattern.

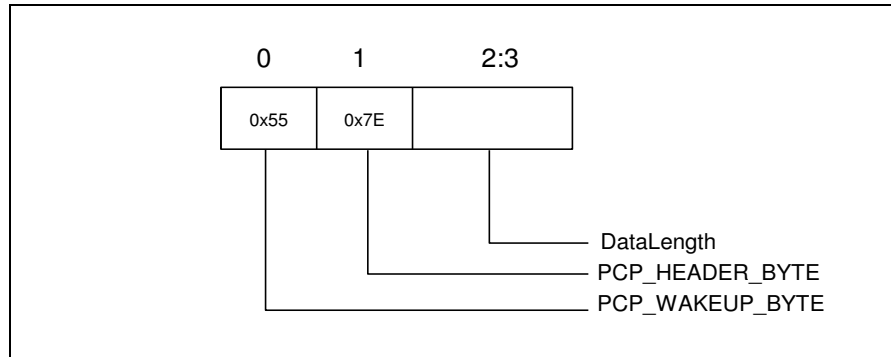


Figure 10 : Header definition figure.

# BYTE	Name	Description
1	PROTO_WAKEUP	Byte uses to bind a protocol packet. Always 0x55.
1	PROTO_HEADER	Byte use to detect beginning of packet. Always 0x7E.
2	DataLength	Length, in bytes, of the data in the data section.

Table 1 : Header definition table

The protocol uses some byte stuffing because it is binary based. If wakeup bytes are used in the data field, they must be replace by “0x55 - 0x00”. To simplify the code, a 0x00 is inserted after all 0x55 found in the data length bytes, data bytes and CRC bytes. They will be removed at reception.

vi. Data

Data contains any type of information according to the command sent. Definitions of the different possibilities are described in section 3.2.

Since DataLength holds on 16-bit, data can be between 1 and 65535 bytes.

# BYTE	Name	Description
n	Data	Data to transmit depending on command.

Table 2: Data definition table.

vii. Footer

The footer contains 2 bytes made of the following pattern.

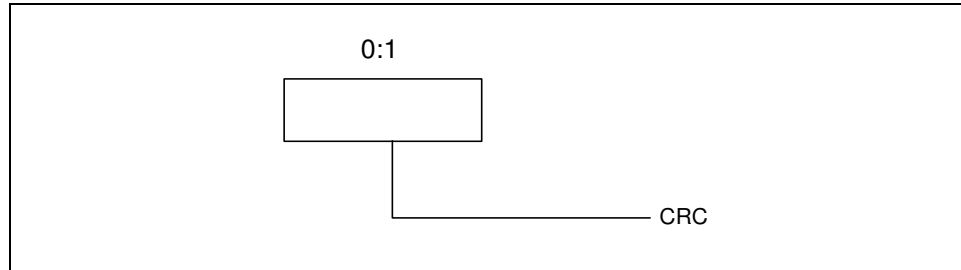


Figure 11 : Footer definition figure.

# BYTE	Name	Description
2	CRC	Cyclical Redundancy Checking uses to verify packet integrity.

Table 3 : Footer definition table.

The CRC computed is CRC-16 CCITT with polynomial ($X^{16} + X^{12} + X^5 + 1$).

4.6) Commands

All command must have the same pattern. First byte defines the command, leaving plenty of space to add more command later. A mask bit is used to represents an answer to a command (0x80 mask on the command value). Following it is all the rest of the data require for managing a specific command.

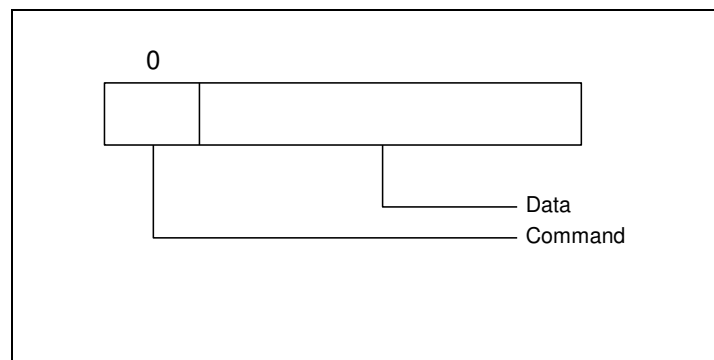


Figure 12 : Data definition detail figure

# BYTE	Name	Description
2	Command	Command identification number.
DataLength-1	ParamData	Rest of the command data. Varies depending of command.

Table 4 : Data definition global detail table

i. Configuration commands

Configuration commands are the static parameters of the system, which must be defined as soon as the ATCC is used.

a. SetFanStatus command

This command permit to enable/disable fans.

# BYTE	Name	Description
2	Command	0x0001
1	Status	The new fans status: Bits 3-7 : Reserved Bit 2: Hot fan status : Enabled (1) Disabled (0) Bit 1: Reserved. Bit 0: Cold fan status : Enabled (1) Disabled (0)

Table 5 : SetFanStatus command

# BYTE	Name	Description
2	Command	0x8001

Table 6 : SetFanStatus answer

b. GetFanStatus command

This command permit to get status of fans.

# BYTE	Name	Description
2	Command	0x0002

Table 7 : GetFanStatus command

# BYTE	Name	Description
2	Command	0x8002
1	Status	The fans status: Bits 3-7 : Reserved Bit 2: Hot fan status : Enabled (1) Disabled (0) Bit 1: Reserved. Bit 0: Cold fan status : Enabled (1) Disabled (0)

Table 8 : GetFanStatus answer

c. SetFanAlarmStatus command

This command permit to enable/disable fan alarms.

# BYTE	Name	Description
2	Command	0x002C
1	Status	The new fans status: Bits 3-7 : Reserved Bit 2: Hot fan status : Enabled (1) Disabled (0) Bit 1: Reserved. Bit 0: Cold fan status : Enabled (1) Disabled (0)

Table 9 : SetFanAlarmStatus command

# BYTE	Name	Description
2	Command	0x802C

Table 10 : SetFanAlarmStatus answer

d. GetFanAlarmStatus command

This command permit to get status of fan alarm(enable/disable).

# BYTE	Name	Description
2	Command	0x002D

Table 11 : GetFanAlarmStatus command

# BYTE	Name	Description
2	Command	0x802D
1	Status	The fans status: Bits 3-7 : Reserved Bit 2: Hot fan status : Enabled (1) Disabled (0) Bit 1: Reserved. Bit 0: Cold fan status : Enabled (1) Disabled (0)

Table 12 : GetFanAlarmStatus answer

e. SetTachStatus command

This command permit to enable/disable tachometers for fans.

# BYTE	Name	Description
2	Command	0x0003
1	Status	The new tach fans status: Bits 3-7 : Reserved Bit 2: Hot fan tachometer : Enabled (1) Disabled (0) Bit 1: Reserved. Bit 0: Cold fan tachometers: Enabled (1) Disabled (0)

Table 13 : SetTachStatus command

# BYTE	Name	Description
2	Command	0x8003

Table 14 : SetTachStatus answer

f. GetTachStatus command

This command permit to get status of fans tachometers.

# BYTE	Name	Description
2	Command	0x0004

Table 15 : GetTachStatus command

# BYTE	Name	Description
2	Command	0x8004
1	Status	The fans status: Bits 3-7 : Reserved Bit 2: Hot fan tachometer: Enabled (1) Disabled (0) Bit 1: Reserved. Bit 0: Cold fan tachometer: Enabled (1) Disabled (0)

Table 16 : GetTachStatus answer

g. SetFanNominalSpeed command

This command permit to set nominal speed RPM parameter of fans.

# BYTE	Name	Description
2	Command	0x0005
2	Nominal RPM cold fan	The new cold fan RPM: Bits 0-15: Value of nominal speed of cold fan (1LSB=1RPM)
2	Nominal RPM hot fan	The new hot fan RPM: Bits 0-15: Value of nominal speed of hot fan (1LSB=1RPM)

Table 17 : SetFanNominalSpeed command

# BYTE	Name	Description
2	Command	0x8005

Table 18 : SetFanNominalSpeed answer

h. GetFanNominalSpeed command

This command permit to get fans nominal speed.

# BYTE	Name	Description
2	Command	0x0006

Table 19 : GetFanNominalSpeed command

# BYTE	Name	Description
2	Command	0x8006
2	Nominal RPM cold fan	Cold fan RPM: Bits 0-15: Value of nominal speed of cold fan (1LSB=1RPM)
2	Nominal RPM hot fan	Hot fan RPM: Bits 0-15: Value of nominal speed of hot fan (1LSB=1RPM)

Table 20 : GetFanNominalSpeed answer

i. SetThermostatStatus command

This command permit to enable/disable thermostat on non-regulated plate.

# BYTE	Name	Description
2	Command	0x0007
1	Status	The new thermostat status: Bit 1-7: Reserved. Bit 0: Thermostat status : Enabled (1) Disabled (0)

Table 21 : SetThermostatStatus command

# BYTE	Name	Description
2	Command	0x8007

Table 22 : SetThermostatStatus answer

j. GetThermostatStatus command

This command permit to get status of thermostat on non-regulated plate.

# BYTE	Name	Description
2	Command	0x0008

Table 23 : GetThermostatStatus command

# BYTE	Name	Description
2	Command	0x8008
1	Status	Thermostat status: Bit 1-7: Reserved. Bit 0: Thermostat status : Enabled (1) Disabled (0)

Table 24 : GetThermostatStatus answer

k. SetNonRegulatedLimitTemp command

This command set limits high and low of temperature for non-regulated plate.

# BYTE	Name	Description
2	Command	0x0009
2	Minimum temperature	Bit 15 : Reserved Bits 0-14: Value of minimum temperature (1LSB=0.01°C)
2	Maximum temperature	Bit 15 : Reserved Bits 0-14: Value of maximum temperature (1LSB=0.01°C)

Table 25: SetNonRegulatedLimitTemp command

# BYTE	Name	Description
2	Command	0x8009

Table 26: SetNonRegulatedLimitTemp answer

1. GetNonRegulatedLimitTemp command

This command gets limits high and low of temperature for non-regulated plate.

# BYTE	Name	Description
2	Command	0x000A

Table 27: GetNonRegulatedLimitTemp command

# BYTE	Name	Description
2	Command	0x800A
2	Minimum temperature	Bit 15 : Reserved Bits 0-14: Value of minimum temperature (1LSB=0.01°C)
2	Maximum temperature	Bit 15 : Reserved Bits 0-14: Value of maximum temperature (1LSB=0.01°C)

Table 28: GetNonRegulatedLimitTemp answer

m. SetRegulatedLimitTemp command

This command set limits high and low of temperature for regulated plate.

# BYTE	Name	Description
2	Command	0x000B
2	Minimum temperature	Bit 15 : Reserved Bits 0-14: Value of minimum temperature (1LSB=0.01°C)
2	Maximum temperature	Bit 15 : Reserved Bits 0-14: Value of maximum temperature (1LSB=0.01°C)

Table 29: SetRegulatedLimitTemp command

# BYTE	Name	Description
2	Command	0x800B

Table 30: SetRegulatedLimitTemp answer

n. GetRegulatedLimitTemp command

This command gets limits high and low of temperature for regulated plate.

# BYTE	Name	Description
2	Command	0x000C

Table 31: GetRegulatedLimitTemp command

# BYTE	Name	Description
2	Command	0x800C
2	Minimum temperature	Bit 15 : Reserved Bits 0-14: Value of minimum temperature (1LSB=0.01°C)
2	Maximum temperature	Bit 15 : Reserved Bits 0-14: Value of maximum temperature (1LSB=0.01°C)

Table 32: GetRegulatedLimitTemp answer

o. SetTECCurrent command

This command set value of nominal current for TEC.

# BYTE	Name	Description
2	Command	0x000D
2	Value of nominal current	Bits 0-15 : TEC current value (1LSB = 1mA)

Table 33: SetTECCurrent command

# BYTE	Name	Description
2	Command	0x800D

Table 34: SetTECCurrent answer

p. GetTECCurrent command

This command gets last value of nominal TEC current.

# BYTE	Name	Description
2	Command	0x000E

Table 35: GetTECCurrent command

# BYTE	Name	Description
2	Command	0x800E
2	Current value	Bit 15 : Reserved Bits 0-14: TEC current value (1LSB = 0.5mA)

Table 36: GetTECCurrent answer

q. SetNTCRegParameters command

This command set parameters for regulated plate for temperature sensor of type NTC.

# BYTE	Name	Description
2	Command	0x0031
2	B value	Bits 0-15 : Value of B (°K)
2	R0	Bits 0-15 : Value of Res R0 (Ohm)
1	T0	Bits 0-7 : Value of temperature for R0 (127 to -128)

Table 37: SetNTCRegParameters command

# BYTE	Name	Description
2	Command	0x8031

Table 38: SetNTCRegParameters answer

r. GetNTCRegParameters command

This command gets parameters for temperature sensor of type NTC for regulated plate.

# BYTE	Name	Description
2	Command	0x0032

Table 39: GetNTCRegParameters command

# BYTE	Name	Description
2	Command	0x8032
2	B value	Bits 0-15 : Value of B (°K)
2	R0	Bits 0-15 : Value of Res R0 (Ohm)
1	T0	Bits 0-7 : Value of temperature for R0 (127 to -128)

Table 59: GetNTCRegParameters answer

s. SetNTCNonRegParameters command

This command set parameters for non-regulated plate for temperature sensor of type NTC.

# BYTE	Name	Description
2	Command	0x0033
2	B value	Bits 0-15 : Value of B (°K)
2	R0	Bits 0-15 : Value of Res R0 (Ohm)
1	T0	Bits 0-7 : Value of temperature for R0 (127 to -128)

Table 40: SetNTCNonRegParameters command

# BYTE	Name	Description
2	Command	0x8033

Table 41: SetNTCNonRegParameters answer

t. GetNTCRegParameters command

This command gets parameters for temperature sensor of type NTC for non-regulated plate.

# BYTE	Name	Description
2	Command	0x0034

Table 42: GetNTCNonRegParameters command

# BYTE	Name	Description
2	Command	0x8034
2	B value	Bits 0-15 : Value of B (°K)
2	R0	Bits 0-15 : Value of Res R0 (Ohm)
1	T0	Bits 0-7 : Value of temperature for R0 (127 to -128)

Table 62: GetNTCNonRegParameters answer

ii. Parameters setting commands

This section describes parameters commands for ATCC. These parameters are statics parameters which permit to configure control law of the system. Since characteristics of output may vary, these parameters can be easily adjusted.

a. SetTempSource command

This command set source of temperature (external or internal), and internal temperature reference for set point.

# BYTE	Name	Description
2	Command	0x0019
1	Temperature Source	Bit 0: Temperature source : Internal (0), External (1)
2	Temperature value	Bits 0-15 : Value of internal reference temperature (1LSB=1°C)

Table 43: SetTempSource command

# BYTE	Name	Description
2	Command	0x8019

Table 44: SetTempSource answer

b. GetTempSource command

This command gets the type of control law used in system.

# BYTE	Name	Description
2	Command	0x001A

Table 45: GetTempSource command

# BYTE	Name	Description
2	Command	0x801A
1	Temperature Source	Bit 0: Temperature source : Internal (0), External (1)
2	Temperature value	Bits 0-15 : Value of internal reference temperature (1LSB=1°C)

Table 46: GetTempSource answer

c. SetPIDGainValues command

This command set value of gains for PID controller.

# BYTE	Name	Description
2	Command	0x001F
2	Proportional gain	Value of K_P : Bits 0-15: Value of gain (1LSB=1)
2	Integral gain	Value of K_I : Bits 0-15: Value of gain (1LSB=0.01)
2	Derivate gain	Value of K_D : Bits 0-15: Value of gain (1LSB=100)

Table 47: SetPIDGainValues command

# BYTE	Name	Description
2	Command	0x801F

Table 48: SetPIDGainValues answer

d. GetPIDGainValues command

This command gets value of gains for PID controller.

# BYTE	Name	Description
2	Command	0x0020

Table 49: GetPIDGainValues command

# BYTE	Name	Description
2	Command	0x8020
2	Proportional gain	Value of K_P : Bits 0-15: Value of gain (1LSB=1)
2	Integral gain	Value of K_I : Bits 0-15: Value of gain (1LSB=0.01)
2	Derivate gain	Value of K_D : Bits 0-15: Value of gain (1LSB=100)

Table 50: GetPIDGainValues answer

e. UpdateParameters command

This command indicates an update for parameters (time to save parameters in EEprom).

# BYTE	Name	Description
2	Command	0x0021

Table 51: UpdateParameters command

# BYTE	Name	Description
2	Command	0x8021

Table 52: UpdateParameters answer

iii. Data monitoring commands

a. GetTemperatures command

This command gets all values of temperature sensors on a controller.

# BYTE	Name	Description
2	Command	0x0022

Table 53: GetTemperatures command

# BYTE	Name	Description
2	Command	0x8022
2	Non-regulated plate temperature	Bits 12-15: Reserved Bits 0-11: Sensor value
2	Regulated plate temperature	Bits 12-15: Reserved Bits 0-11: Sensor value
2	External reference temperature	Bits 12-15: Reserved Bits 0-11: Sensor value

Table 54: GetTemperatures answer

b. GetAlarmStatus command

This command gets status of all alarms.

# BYTE	Name	Description
2	Command	0x0023

Table 55 : GetAlarmStatus command

# BYTE	Name	Description
2	Command	0x8023
2	States of alarms	Bit 8: H-Bridge over-temperature = Active (1), Inactive(0) Bit 7: TEC current alarm = Active (1), Inactive(0) Bit 6: Hot fan alarm = Active (1), Inactive(0) Bit 5: Cold fan alarm = Active (1), Inactive(0) Bit 4 : Thermostat alarm = Active (1), Inactive(0) Bit 3: RPA = If active, Over Temperature (1), or Under Temperature(0) Bit 2 : Reg plate alarm (RPA) = Active (1), Inactive(0) Bit 1: NRPA = If active, Over Temperature (1), or Under Temperature(0) Bit 0 : Non-reg plate alarm (NRPA) = Active (1), Inactive(0)

Table 56: GetAlarmStatus answer

c. GetCommand command

This command gets all values of command and current.

# BYTE	Name	Description
2	Command	0x002F

Table 57: GetCommand command

# BYTE	Name	Description
2	Command	0x802F
2	Current (mA)	Bits 0-15: Current value
2	Command (0-100)	Bits 7-15: Reserved Bits 0-6: Command value

Table 58: GetCommand answer

iv. Upgrade and device information commands

a. GetSerial command

This command gets the controller serial number.

# BYTE	Name	Description
2	Command	0x0025

Table 59 : GetSerial command

# BYTE	Name	Description
2	Command	0x8025
3	Serial Number	Bits 14-23 : Number of ATCC (0 -> 999) Bits 8-13: Week (max 52) Bits 0-7: Year modulo 2000

Table 60: Get Serial answer

b. SetSerial command

This command set the serial number (called once).

# BYTE	Name	Description
2	Command	0x0030
3	Serial Number	Bits 14-23 : Number of ATCC (0 -> 999) Bits 8-13: Week (max 52) Bits 0-7: Year modulo 2000

Table 61 : SetSerial command

# BYTE	Name	Description
2	Command	0x8030

Table 62 : SetSerial answer

c. GetDeviceInfos command

This command gets the controller code revision and serial number.

# BYTE	Name	Description
2	Command	0x0024

Table 63 : GetDeviceInfos command

# BYTE	Name	Description
2	Command	0x8024
1	Minor revision	Bits 0-7: Minor revision
1	Major revision	Bits 0-7: Major revision
9	Revision Date	BYTES 5-8: Year BYTES 2-4: Month BYTES 0-1: Day
6	Revision Time	BYTES 4-5: Hour BYTES 2-3: Minute BYTES 0-1: Seconde

Table 64 : GetDeviceInfos answer

d. HardwareReset command

This command make a hardware reset of controller in order to enter update mode.

# BYTE	Name	Description
2	Command	0x0026

Table 65: HardwareReset command

# BYTE	Name	Description
2	Command	0x8026

Table 66: HardwareReset answer

e. UpgradeInit command

This command begins the firmware upgrade operation.

# BYTE	Name	Description
2	Command	0x0027
6	InitData	Initialization data.

Table 67 : UpgradeInit command

# BYTE	Name	Description
2	Command	0x8027
2	UpgradeAnswer	Upgrade answer.

Table 68 : UpgradeInit answer

f. UpgradeSRec command

This command sends data for the firmware upgrade operation.

# BYTE	Name	Description
2	Command	0x0028
variable	SRecord	S Record

Table 69 : UpgradeSRec command

# BYTE	Name	Description
2	Command	0x8028
2	UpgradeAnswer	Upgrade answer.

Table 70 : UpgradeSRec answer

g. UpgradeEnd command

This command ends the firmware upgrade operation.

# BYTE	Name	Description
2	Command	0x0029

Table 71 : UpgradeEnd command

# BYTE	Name	Description
2	Command	0x8029
2	UpgradeAnswer	Upgrade answer.

Table 72 : UpgradeEnd answer

h. StopControl command

This command stops the control loop into controller.

# BYTE	Name	Description
2	Command	0x002A

Table 73: StopControl command

# BYTE	Name	Description
2	Command	0x802A

Table 74: StopControl answer

i. RestartControl command

This command restarts the control loop into controller.

# BYTE	Name	Description
2	Command	0x002B

Table 75: RestartControl command

# BYTE	Name	Description
2	Command	0x802B

Table 76: RestartControl answer

j. StartConnect command

This command gets the state of ATCC (no data saved yet, or normal) to start connection.

# BYTE	Name	Description
2	Command	0x002E

Table 77: StartConnect command

# BYTE	Name	Description
2	Command	0x802E
1	State of control	Bits1-15 : Reserved Bit 0: Data (0: No data, 1:Data saved)

Table 78: StartConnect answer

k. GetControlState command

This command gets the state of control (ON/OFF).

# BYTE	Name	Description
2	Command	0x0035

Table 79: GetControlState command

# BYTE	Name	Description
2	Command	0x8035
1	State of control	Bits1-15 : Reserved Bit 0: State (0: disable, 1:enable)

Table 80: GetControlState answer

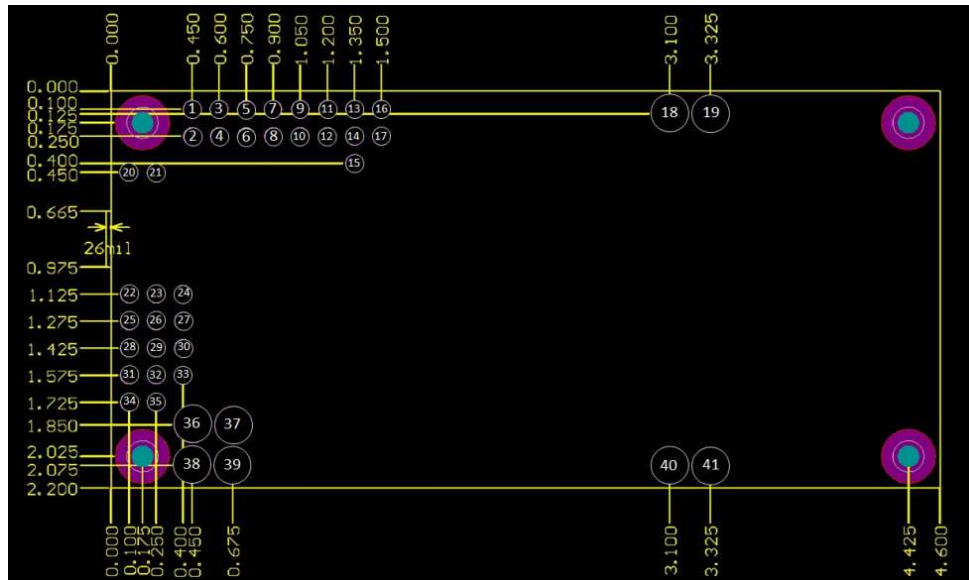
5. Mechanical

ATCC has been designed to interact with all thermoelectric assemblies within its range. An expansion board permits to directly connect all signals through terminal blocks. Also, PC pins can be directly soldered onto an expansion board, which footprint is furnished below. Here in an overview of connections and size of the board.

FUNCTION	DESCRIPTION
FANS	
Output	Two fans outputs (for both plates) are open-drain collector connected to Vin and control driven by MCU. System can operate with or without fan.
Input	Two inputs are provided to read thermostat in order to stop fans if fault is detected (e.g. fan has slow down). Alarm can be set or not, independently.
SENSORS	
Plate temperature sensors	Two NTC sensors must be connected on each plate of TEC in order to detect a temperature fault. Ranges of temperature are fully configurable.
External Set Point	The Set Point can be either external or internal. When using external, temperature is set via a potentiometer.
Current Sensor	A current sensor permit to measure current that goes to the TEC. It can avoid over/under current faults (hardware circuit), but is also an indicator indicating nominal current that goes to the drive, and can limit command.
Thermostat	An input is provided in order to branch a Normally-Closed thermostat on non-regulated plate. As soon as fault is detected, system stops.
Drive thermal protection	Drives possess a thermal flag that can cause the system to stop.
COMMUNICATION	
USB	A mini-B type USB connector can be used to set up parameters of TEC, and to select features of product. An interface permits to easily configure ATCC.
SPI	An external MCU can be connected to ATCC via an external SPI.
OUTPUTS	
Alarm output	An open-drain alarm output is also provided in order to branch an external signal (ex: LED, buzzer).
Status LEDS	Three LEDS indicates states of ATCC (Run, Work, Alarm).
DAC output	Two DAC outputs are used in order to supervise plate temperatures as analog signals.
MECHANICAL	
PCB Pins	All signals and supply of the board are connected to PCB pins. These pins can then be interfaced to terminal blocks via an expansion board. Sysacom provides an expansion board, and furnish all footprints in order to make a custom electrical interface.

Here is a list of all connections on ATCC that can be accessed through Expansion Board

PCB pins connections			
Pin	Function	Pin	Function
1	RTC non-reg GND	21	Thermostat GND
2	RTC non-reg OUT	22	Hot fan VIN
3	RTD non-reg GND	23	Hot fan CTRL
4	RTD non-reg OUT	24	Hot fan tach.
5	RTD non-reg +3.3V	25	Cold fan VIN
6	RTD reg +3.3V	26	Cold fan CTRL
7	RTD reg OUT	27	Cold fan tach.
8	RTD reg GND	28	Set point GND
9	RTC reg GND	29	Set point +3.3V
10	RTC reg OUT	30	Set point OUT
11	Spi GND	31	Output temp. GND
12	Spi +3.3V	32	Non-reg temp. OUT
13	Spi MOSI	33	Reg temp. OUT
14	Spi N_INT	34	Alarm output GND
15	Spi N_CS	35	Alarm output CTRL
16	Spi MISO	36, 37	VIN
17	Spi CLK	38, 39	GND
18, 19	TEC entry	40,41	TEC output
20	Thermostat OUT		



Overview of footprint

of ATCC.

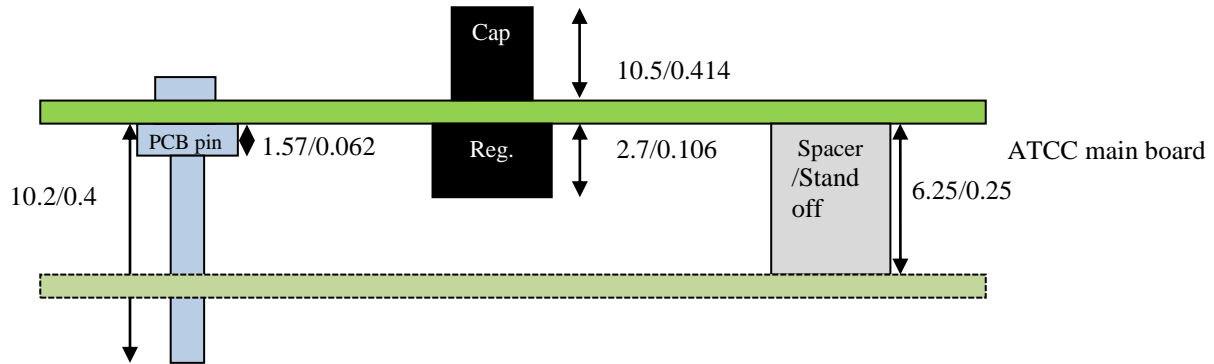
Dimensions are in

inches (in).

Size of drill / pad in inches (in).

Drill/pad sizes		
	With receptacle	No receptacle
Small pins	54/100	38/62
Big pins	162/200	93/150
Mounting holes	125/300	

Overview of the highest / lowest components top and bottom:



(Expansion board)

Drawing not to scale. Dimensions are mm/in. Expansion board is here for representation only.

Ordering information

Part Number	Purpose	Temperature range
ATCC2707	Main Board	-20°C to +55°C
ATCC2707_EVB	Expansion board	-40°C to +55°C

All products mentioned above are Lead-free/RoHs compliant.

6. Warranty

Sysacom R&D plus Inc warrants the Advanced Thermoelectric Cooler Controller 27V/7A (ATCC2707) to be free from malfunctions and defects in both materials and workmanship for one year from the date of purchase.

If the equipment does not function properly during the warranty period due to defects in either materials or workmanship, Sysacom R&D plus Inc will, at its option, either repair or replace the equipment without charge, subject to limitations stated herein. Such repair service will include all labour, as well as any necessary adjustments and / or replacement parts.

LIMITATIONS

The warranty becomes null and void if you fail to pack your ATCC2707 in a manner consistent with the original product packaging and damage occurs during shipment.

Sysacom R&D plus Inc makes no other warranties, express, implied, or of merchantability or fitness for a particular purpose for this equipment or software. Repair or replacements without charge are Sysacom R&D plus Inc only obligation under this warranty. Sysacom R&D plus Inc will not be responsible for any special, consequential or incidental damages resulting from the purchase, use, or improper functioning of this equipment regardless of the cause.

Depending on your geographical location, some limitations may not apply.

Sysacom R&D plus inc.
275A Pierre-Le Gardeur Blvd
Repentigny, Quebec
Canada, J5Z 3A7

www.sysacom.com

(450)585-6396