GC.TC Serial Protocol

NOTA BENE:
Any software issuing multibyte commands must ensure that btf is not equal to any of the single-byte commands. Inserting bytes with a value of zero at the end of the data section is one way to accomplish this for commands that happen to have such a btf value.

Single-Byte Commands

- u: increase temperature setpoint (by 1.0 degree)
- d: decrease temperature setpoint (by 1.0 degree)
- s: start/stop control

Single-byte commands will send no response, other than performing the requested action.

Multibyte Commands

Commands issued from software should take use the following format. All commands of this type will trigger a response from the software regardless of outcome, unless commands are being sent too fast for the software to respond (which is unlikely but altogether possible).

Transmission: btf, xbtf, command, data, checksum, end byte
Response: btf, xbtf, command, data, ack/nack, checksum, end byte

- btf
  Single byte, indicating how many bytes are in the message following xbtf
- xbtf
  Single byte, equal to the one's complement of btf (i.e. btf + xbtf = 0xFF). Use this property for error checking to ensure the btf byte is not corrupt.
- command
  Three ASCII characters, with specific behavior defined later in this document.
- data
  Variable-length data section, with behavior defined depending on command.
- ack/nack
  Single byte indicating failure (0x00) or success (0x01) in executing the command. Will fail if the embedded software could not execute the command for any reason, including but not limited to a bad checksum, buffer overflow, corrupt btf, unknown command or bad data.
- checksum
  Two byte, big-endian value that is the sum of all previous bytes in the message.
- end byte
  Single ASCII '>' to indicate end of message in case btf becomes corrupted.

If btf or xbtf is corrupt, embedded software will ignore all received bytes until it gets the end byte, when it will send a nacked response to an "OS" (out of sync) command (i.e. the response will be the byte sequence: 0x06, 0xF9, 'O', 'S', 0, 0x01, 0xA1, '>') and then wait for a new message.

Specific Commands

Following are detailed descriptions of the currently used command set available.

- **GVT: Get the current temperature in °C**
  - Outgoing Data:
    - None.
  - Response Data:
    - ASCII representation of the current temperature in °C
  - Notes:
    - All response values are terminated by an 0x0D byte.
    - Response data is preceded by an 0x0D byte.
- **GVS: Get the system setpoint temperature in °C**
  - Outgoing Data:
    - None.
  - Response Data:
    - ASCII representation of the current setpoint temperature in °C
  - Notes:
    - All response values are terminated by an 0x0D byte.
    - Response data is preceded by an 0x0D byte.
- **SVS: Set the system temperature setpoint in °C**
  - Outgoing Data:
    - The ASCII representation of the temperature to set in °C
  - Response Data:
    - 'S' command acknowledgement response.
  - Notes:
    - Values are decimal value in ASCII form, terminated with a non-numeric character, unless specified otherwise.