# RTD SCANNER, EIGHT CHANNEL
## With CANopen®

# USER MANUAL

**P/N: AXRTD8CO**

**VERSION HISTORY**

| Version | Date | Author | Modifications |
|---------|------|--------|---------------|
| 1.0.0 | August 13, 2010 | Anna Murray | Initial Draft |
| 1.0.1 | November 10, 2010 | Anna Murray | Changed fixed width of RS-232 data stream from 5 to 6 characters. Updated screen capture |
| - | November 19, 2010 | Amanda Wilkins | Changed optical isolation to digital isolation |
| 1.1.0 | January 25, 2011 | Anna Murray | Added new object 5555h |
| 2.0.0 | December 19, 2011 | Amanda Wilkins | New hardware – updated with certifications |
| 2.1.0 | April 26, 2012 | Anna Murray | Updated section 4.1.5 for Load New Software to reflect the new bootloader. |
| 2.1.1 | October 9, 2012 | Amanda Wilkins | Added input resistance range and clarified isolation. |
| 2.1.1A | April 2, 2014 | Olek Bogush | Added "Short Circuit" and "Open Circuit" conditions to the Technical Specification. |
| 2.1.1A | October 2, 2015 | Amanda Wilkins | Updated IP rating to IP67 based on test results |

**ACRONYMS**

| | |
|---|---|
| AVG | Average |
| CAN | Controller Area Network |
| CANopen® | CANopen® is a registered community trademark of CAN in Automation e.V. |
| CAN-ID | CAN 11-bit Identifier |
| CJ | Cold Junction |
| COB | Communication Object |
| EDS | Electronic Data Sheet |
| EMCY | Emergency |
| LSB | Least Significant Byte (or Bit) |
| LSS | Layer Settling Service |
| MSB | Most Significant Byte (or Bit) |
| MEMS | Micro-electromechanical system |
| NMT | Network Management |
| RO | Read Only Object |
| RPDO | Received Process Data Object |
| RTD | Resistive Thermal Device |
| RW | Read/Write Object |
| SDO | Service Data Object |
| TPDO | Transmitted Process Data Object |
| WO | Write Only Object |

**REFERENCES**

[DS-301]      CiA DS-301 V4.1 – CANopen Application Layer and Communication Profile. CAN in Automation 2005

[DS-305]      CiA DS-305 V2.0 – Layer Setting Service (LSS) and Protocols. CAN in Automation 2006

[DS-404]      CiA DS-404 V1.2 – CANopen profile for Measurement Devices and Closed Loop Controllers. CAN in Automation 2002

These documents are available from the CAN in Automation e.V. website http://www.can-cia.org/.

**TABLE OF CONTENTS**

# 1. OVERVIEW OF RTD SCANNER

## 1.1. Description of RTD Scanner

The following User Manual describes the architecture and functionality of a eight channel CANopen ® RTD scanner.

There are eight channels on the AXRTD8, each with four pins at the connector for 2, 3 or 4 wire connections, as well as a fifth pin for a shield. The RTD scanner will source current on pin A for all types of RTD sensors.

In the case of a 2-wire device, it will read the voltage between Pin A and GND, with no compensation for any resistance added by the wires. For 3-wire devices, it reads the voltage at Pin A, as well as that at Pin C. The Pin C reading will allow the device to calculate the approximate resistance in one wire, and will subtract twice that value from the calculated resistance based on the voltage measure at A. Lastly, for 4-wire devices, it will read the voltage at Pin B which already takes into account the resistance of the wire from A. It will also measure the voltage at Pin C to calculate the return wire resistance, and subtract that from the measured value at B.
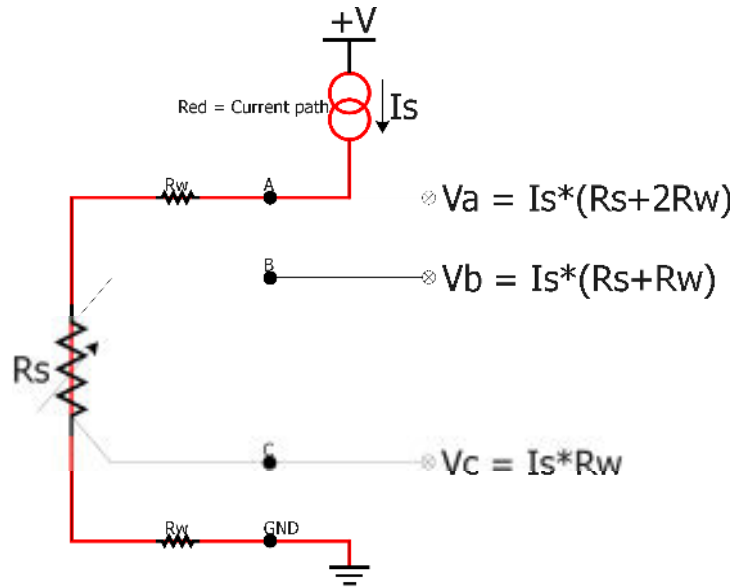


**Figure 1 – Resistive Input Wiring and Measurement**

In the case of a 4 wire RTD, if the wire to A is broken, the unit will report the load a short circuited because there will be no voltage on Pin B.

All channels are fully isolated from the CAN lines, and from the power supply. The power supply was designed for a wide range of nominal inputs of 12V, 24V or 48V and will provide proper operation from 9 to 60Vdc.

If desired, the average temperature of all the active channels, or all channels from a block of 4, can be broadcasted to the network using the Average Input function block. This feature is described in detail in section 1.3.

On power-up, the AXRTD8CO will immediately send the bootup message to the network. However, in ordered to prevent erroneous readings before all the data from all channels have been read correctly, the unit will only start broadcasting diagnostic data after 5 seconds have elapsed, and will not enter "Operational" mode during this period.

To measure voltages, the AXRTD8CO uses a very precise (24bit) dual channel analog-to-digital converter with a programmable gain. The RTD inputs and the current source (common to all 8 inputs) are multiplexed to the ADC chip. It has a programmable filtering for either 50Hz or 60Hz. The ADC provides a minimum 100dB normal mode rejection of the line frequency and its harmonics.

Active channels are scanned sequentially (1 to 8) with approximately 100ms between readings. For 2-wire types, there is one reading per channel, whereas 3 or 4 wire types require two readings per channel. On every read-thru of the channels, there is also a measurement taken of the common current source used to generate the voltage on each channel.

For 3 or 4 wire type channels, the wire resistance is only checked after 600 reads from the AtoD, since it does not change that frequently. This means the wire resistance is read and update once a minute.

If all 8 channels are active, it takes approximately 900ms to read through all the channels, and the current source. Therefore, any individual channel's reading is updated at least once per second, less if not all channels are active.

Temperature is measured in ºC, with a 0.1ºC resolution. When installed properly, as described in section 2.2, the scanner will send temperatures with +/- 1ºC accuracy typical at ambient temperature.

The scanner can be used to flag low temperature warnings, high temperature warnings, or high temperature shutdowns. It will also detect and flag open or short circuits on the sensor wires.

## 1.2. RTD Measurements
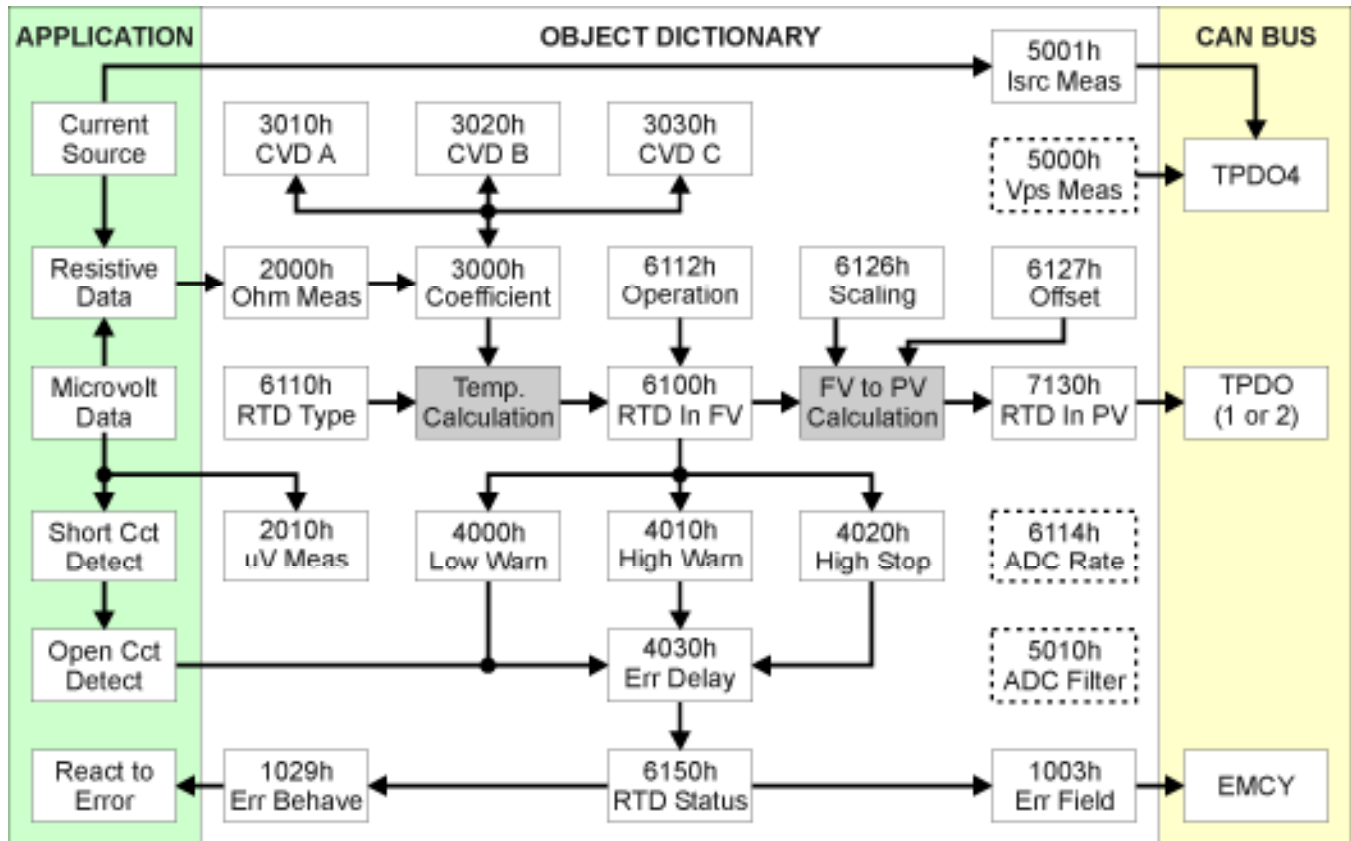


**Figure 2 – RTD Input Block Diagram**

The block diagram shown in Figure 2 capture the objects associated with each RTD channel. Each channel, 1 through 8, operates in the same fashion as described below.

In order to generate a measurable voltage across each resistive sensor, the scanner multiplexes an ~4mA current source to each input and reads the voltage generated. (See Figure 1 for more information.) In order to accurately calculate the resistance from the voltage, the scanner also multiplexes this source across a fixed reference resistor in order to know exactly what current is sourced to each channel. The actual current source value is available on read-only object $5001 **Current Source Measured**. By default, this object, along with read-only object $5000 **Power Supply Measured**, are mapped to TPD04.

Objects $3000h **RTD Coefficient**, $6110 **RTD Sensor Type** and $6112 **RTD Operating Mode** determine how the scanner processes the raw microvolt reading and converts it into a temperature value in degrees Celsius, which is written to read-only object $6100 **RTD Input Field Value**.

The resistance of the sensor is calculated based on the sensor type selected (2-wire, 3-wire or 4-wire) as per the formulas shown in Figure 1. The RTD Scanner then calculates the temperature from the measured resistance using the Callendar-Van Dusen constants.

According to IEC751, the non-linearity of the platinum thermometer can be expressed as:

$$R_t = R_o[1+At+Bt^2+C(t-100)t^3] \text{ in which C is only applicable when } t < 0 \ °C.$$

Depending on the value in object $3000, the objects $3010, $3020 and $3030 **Callendar-Van Dusen Constant A, B and C** are automatically updated as necessary. The constants A, B, and C for a standard sensor are stated in IEC751, and the values used by the scanner are listed below.

| Sensor Coefficient | Constant A [E-03] | Constant B [E-07] | Constant C [E-12] |
|---|---|---|---|
| IEC 00385 | 3.908300 | -5.77500 | -4.18301 |
| JIS 003916 | 3.974673 | -5.89730 | -4.35300 |
| US 003902 | 3.960000 | -5.93000 | -4.30000 |
| Legacy US 003920 | 3.984800 | -5.87000 | -4.00000 |
| SAMA 003923 | 3.981531 | -5.853116 | -4.35453 |

**Table 1 – Callendar-Van Dusen Constants for Standard RTD Coefficients**

Generally speaking, the Callendar-Van Dusen objects are treated as read-only variables. However, should a "User Defined" coefficient be selected, these objects would become write-able in order to allow for RTD sensors not listed in the above table to be connected.

Objects $6126 **RTD Scaling Factor** and $6127 **RTD Scaling Offset** are used to convert the field value to read-only object $7130 **RTD Input Process Value**, which is mapped either to TPDO1 (1 to 4) or TPDO2 (5 to 8) by default.

The formula to convert the field value (FV) to process value (PV) is:

**Process Value = (Field Value * Scaling Factor) + Scaling Offset**

While the FV is a real number, containing the temperature in °C, the PV is a 16-bit integer value. The default scaling has been selected such that the PV will send the temperature with a resolution of 0.0625 °C/bit and a offset of -273°C. [Scaling Factor = 16, Offset = 4368] Since the maximum temperature the scanner can measure for a RTD is 1735°C, this means the range of the PV data will be 0 to 32123 (-273°C to 1735°C.)

Alternatively, it may be desired to send the temperature in Fahrenheit with a 0.1°F resolution per bit. In this case, the Scaling Factor would be set to 18, and the Offset to 320. Other scaling can be selected as desired by the user.

In all cases, certain values will be 'plugged' into the PV object to indicate various conditions. Should the associated RTD be disabled by object $6112, then the value in the PV will always be -1 (0xFFFF).

Alternatively, should the scanner detect an open circuit on the sensor, then the PV value will be set to -512 (0xFE00). A short circuit on the sensor returns a PV value of -448 (0xFE40). Lastly, in the unlikely case that the processor detects that the ADC converter has stopped working (i.e. no longer sending updated data on every scan), then the controller will not continue to broadcast the 'frozen' data, but rather update the PV value to -384 (0xFE80) to indicate that there is a problem with the measurement.

In both error conditions mentioned above, open circuit or frozen data, the associated object $6150 **RTD Status** will also be updated to reflect the problem. Other faults that the scanner can detect and flag are determined by the values in objects $4000 **Low Temperature Warning Threshold**, $4010 **High Temperature Warning Threshold** and $4020 **High Temperature Shutdown Threshold**.

The threshold values are always measured with respect to the field value, and have a fixed resolution of 0.1°C/bit. For example, a value of 1250 in object $4010 means that any measured temperature greater than or equal to 125°C will flag a high temperature warning. The fault detection thresholds also have a fixed 1°C built-in hysteresis to clear them. In the example above, the temperature would have to drop below 124°C to clear the fault once it has been set.

Finally, in order to prevent flooding the network with emergency messages when the temperature hovers around a warning threshold, the object $4030 **Error React Delay** allows the user to select how long the fault condition must be present before the status object is updated and the error reaction is triggered.

Once the status object shows that the FV value is not longer valid, the object $1003 **Pre-Defined Error Field** is updated to reflect the appropriate emergency error code and additional information. Since both a high temperature warning and shutdown could be active at the same time, object $1003 could have up to 16 entries at any given time. Also, when a sensor error is activated, the controller will react as specified in object $1029 **Error Behaviour**.

The error values loaded in the status object $6150 are described in Table 6, while the associated emergency fields that are loaded into object $1003 are outlined in Table 4.

A couple of other miscellaneous objects associated with the RTD channels are three read-only objects $2000 **RTD Resistance**, $2010 **RTD Microvolts** and $6114h **ADC Sampling Rate**. These objects are associated directly with the ADC chip used to measure the RTDs and source current voltages. As channels are disabled, object $6114 is automatically updated by the controller to reflect approximately how many milliseconds will elapse between each scan of a particular channel. Objects $2000 and $2010 are available for debugging purposes.

Lastly, object $5010 **ADC Filter Frequency** is a single value (non-array) that sets the rejection filter frequency used by the analog-to-digital converter. The only permissible values in this case are either 50Hz (i.e. Europe) or 60Hz (i.e. North America).

## 1.3. Average Measurements
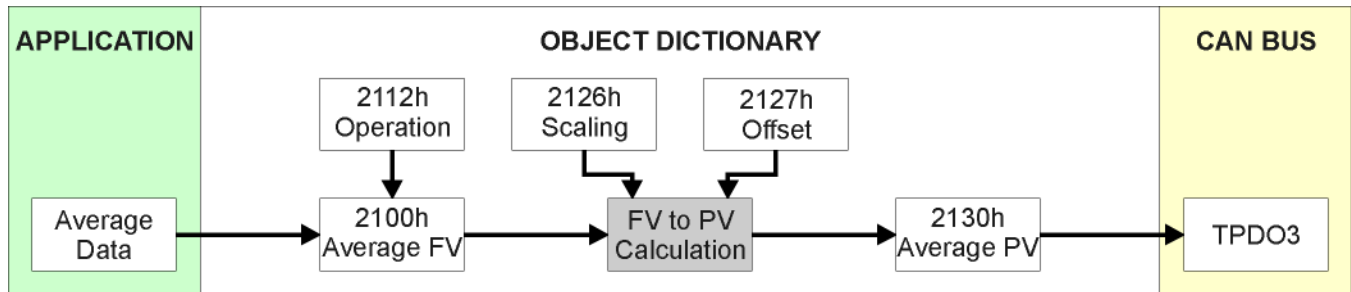


**Figure 3 – Average Measurement Block Diagram**

There are three types of average values that can be measured and broadcasted on a TPDO.
    a) Average of Bank 1 sensors (RTDs 1 to 4, active only)
    b) Average of Bank 2 sensors (RTDs 5 to 8, active only)
    c) Average of all sensors (active only)

Object $2112 **Average Operating Mode** determines if the average value of any of the above will be enabled. When enabled by selecting "Normal Operation", the average of all active channels is calculated and written to read-only object $2100 **Average Input Field Value** in degrees Celsius. If a RTD channel is disabled, open or short circuited or 'frozen', then the value in the FV object is not counted in the average calculation.

As with the RTD inputs, the average FV can be converted to a process value using scaling objects $2126 **Average Scaling Factor** and $2127 **Average Scaling Offset**. The formula to convert to read-only object $2130 **Average Input Process Value** is the same as describe in section 1.2.

By default, all averages are enabled and the calculated PVs are sent on TPDO3.

# 2.  INSTALLATION INSTRUCTIONS

## 2.1.    Dimensions and Pinout



1 = PWR+      5 = SHIELD
2 = CAN-H     6 = RS-232 GND
3 = CAN-L     7 = RS-232 TXD
4 = PWR-      8 = RS-232 RXD

**FRONT  VIEW
MODULE  MOUNTED  CONNECTOR
DEUTSCH P/N:    DT13-08PA**
(Mating  plug  is  DT06-08SA  with  wedge  W8S  and  sockets  0462-201-16141)
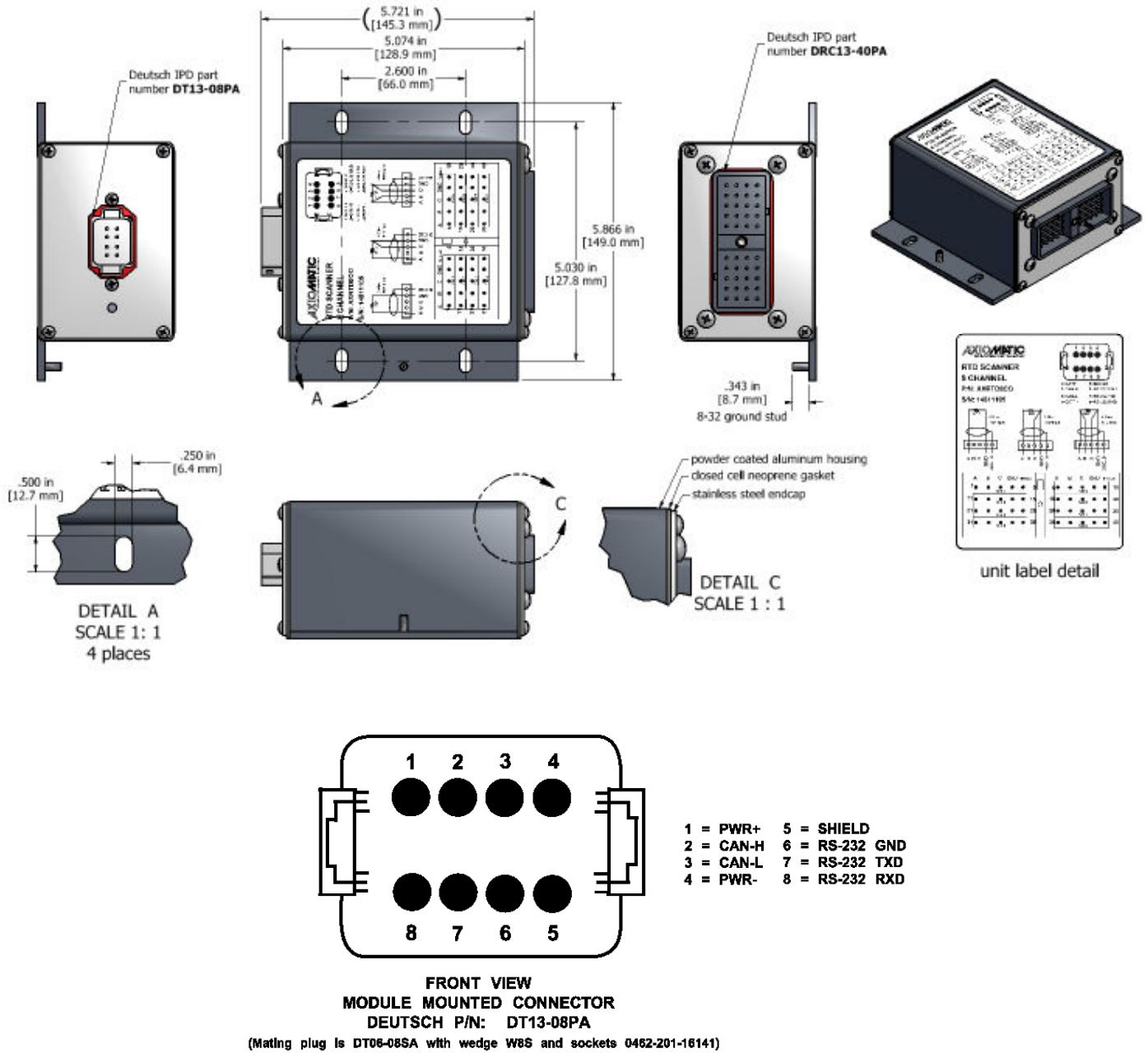
## Typical Connections – RTD Module:
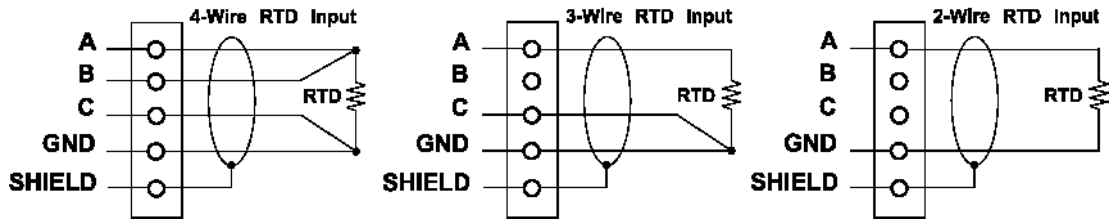


RTD MODULE - PIN OUT
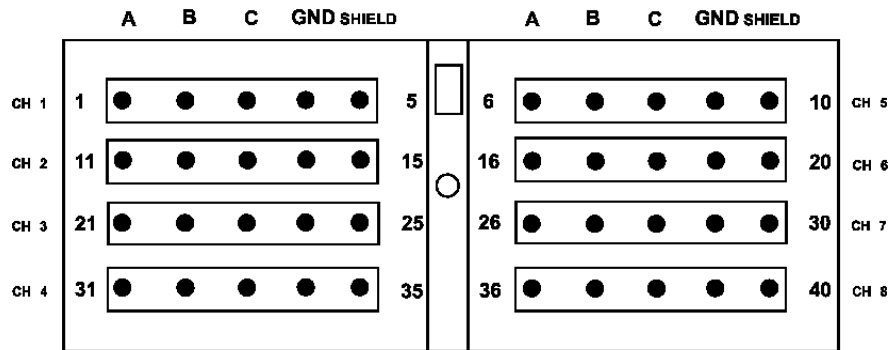


FRONT VIEW OF
MODULE MOUNTED CONNECTOR

Nov. 3/03 AJW

**(Mating plug is Deutsch IPD p/n DRC16-40SA with sockets 0462-201-16141)**

## 2.2.    Installation Instructions

**NOTES & WARNINGS**
- Do not install near high-voltage or high-current devices.
- Ground the chassis for safety purposes and proper EMI shielding.
- Note the operating temperature range.  All field wiring must be suitable for that temperature range.
- Install the unit with appropriate space available for servicing and for adequate wire harness access (15 cm) and strain relief (30 cm).
- Do not connect or disconnect the unit while the circuit is live, unless the area is known to be non-hazardous.

**MOUNTING**

The module is designed for mounting on the engine.  If it is mounted without an enclosure, the RTD Scanner should be mounted vertically with connectors facing left and right to reduce likelihood of moisture entry.

The RTD wires and CAN communication cable are considered intrinsically safe.  The power wires are not considered intrinsically safe.

Mask all labels if the unit is to be repainted, so label information remains visible.

Mounting ledges include holes sized for M6 or ¼ inch bolts.  The bolt length will be determined by the end-user's mounting plate thickness.  Typically 20 mm (3/4 inch) is adequate.

If the module is mounted off-engine, no wire or cable in the harness should exceed 30 meters in length.  The power input wiring should be limited to 10 meters.

**CONNECTIONS**

Use the following Deutsch IPD mating plugs to connect to the integral receptacles.  Wiring to these mating plugs must be in accordance with all applicable local codes.  Suitable field wiring for the rated voltage and current must be used.  The rating of the connecting cables must be at least 85°C.  For ambient temperatures below –10°C and above +70°C, use field wiring suitable for both minimum and maximum ambient temperature.

| Receptacle | Mating Socket (Refer to www.laddinc.com for more information on the wedgelock and contacts for this mating plug.) |
| --- | --- |
| Power and CAN bus:  DT13-08PA | DT06-08SA |
| RTD Interface Receptacle:  DRC13-40PA | DRC16-40SA <br> DRC18-40SA |

## NOISE – ELECTRICAL CONNECTIONS

To reduce noise, separate all RTD wires from power wires.  Shielded RTD wires will protect against ignition and injector noise.

## GROUNDING

Protective Earth (PE) must be connected to the module's grounding lug to reduce the risk of electric shock. The conductor providing the connection must have a ring lug and wire larger than or equal to 4 mm$^2$ (12 AWG).  The ring lug should be placed between the nut and a star washer.

All chassis grounding should go to a single ground point designated for the engine and all related equipment.

The ground strap that provides a low impedance path for EMI should be a ½ inch wide, flat, hollow braid, no more than 12 inches long with a suitable sized ring lug for the module's grounding lug.  It may be used in place of the PE grounding conductor and would then perform both PE and EMI grounding functions.

## SHIELDING

The RTD and CAN wiring should be shielded using a twisted conductor pair. All RTD wire shields should be terminated on the shield wire available on the 40-pin connector. The RTD wires should not be exposed for more than 50 mm (2 inches) without shielding.  The shield may be cut off at the RTD end as it does not require termination at that end.

Shields can be AC grounded at one end and hard grounded at the opposite end to improve shielding effectiveness.

If the module is installed in a cabinet, shielded wiring can be terminated at the cabinet (earth ground), at the entry to the cabinet or at the RTD Scanner.

## INPUT POWER

The main input to the power supply must be of low-impedance type for proper operation.  If batteries are used, an alternator or other battery-charging device is necessary to maintain a stable supply voltage.

Central suppression of any surge events should be provided at the system level.

The installation of the equipment must include overcurrent protection between the power source and the RTD Scanner by means of a series connection of properly rated fuses or circuit breakers. Input power switches must be arranged external to the RTD Scanner.

The power input wiring should be limited to 10 meters.

Note the operating temperature range.  All field wiring must be suitable for that temperature range.

**RTD INPUT WIRING**

Wiring for the RTD input must be shielded cable, 16 or 18 AWG. Cable lengths should be less than 30 meters. Shielding should be unbroken.

**CAN WIRING**

The CAN port is electrically isolated from all other circuits. The isolation is SELV rated with respect to product safety requirements. Refer to the CAN 2.0B specification for more information.

Shielded CAN cable is required. The RTD Scanner provides the CAN port shield connection ac coupled to chassis ground. The chassis ground stud located on the mounting foot must be tied directly to Earth Ground.

**NETWORK CONSTRUCTION**

Axiomatic recommends that multi-drop networks be constructed using a "daisy chain" or "backbone" configuration with short drop lines.

**TERMINATION**

It is necessary to terminate the network; therefore an external CAN termination is required. No more than two network terminators should be used on any one single network. A terminator is a 121Ω, 0.25 W, 1% metal film resistor placed between CAN_H and CAN_L terminals at the end two nodes on a network.

# 3. CANOPEN ® OBJECT DICTIONARY

The CANopen object dictionary of the RTD Scanner is based on CiA device profile DS-404 V1.2 (device profile for RTD Scanners). The object dictionary includes Communication Objects beyond the minimum requirements in the profile, as well as several manufacturer-specific objects for extended functionality.

## 3.1. NODE ID and BAUDRATE

By default, the RTD Scanner ships factory programmed with a Node ID = 127 (0x7F) and with Baudrate = 125 kbps.

### 3.1.1. LSS Protocol to Update

The only means by which the Node-ID and Baudrate can be changed is to use Layer Settling Services (LSS) and protocols as defined by CANopen ® standard DS-305.

Follow the steps below to configure either variable using LSS protocol. If required, please refer to the standard for more detailed information about how to use the protocol.

#### 3.1.1.1. Setting Node-ID

- Set the module state to LSS-configuration by **sending** the following message:

| Item | Value | |
|------|-------|--|
| COB-ID | 0x7E5 | |
| Length | 2 | |
| Data 0 | 0x04 | (cs=4 for switch state global) |
| Data 1 | 0x01 | (switches to configuration state) |

- Set the Node-ID by **sending** the following message:

| Item | Value | |
|------|-------|--|
| COB-ID | 0x7E5 | |
| Length | 2 | |
| Data 0 | 0x11 | (cs=17 for configure node-id) |
| Data 1 | Node-ID | (set new Node-ID as a hexadecimal number) |

- The module will send the following response (any other response is a failure):

| Item | Value | |
|------|-------|--|
| COB-ID | 0x7E4 | |
| Length | 3 | |
| Data 0 | 0x11 | (cs=17 for configure node-id) |
| Data 1 | 0x00 | |
| Data 2 | 0x00 | |

- Save the configuration by **sending** the following message:

| Item | Value | |
| --- | --- | --- |
| COB-ID | 0x7E5 | |
| Length | 1 | |
| Data 0 | 0x17 | (cs=23 for store configuration) |

- The module will send the following response (any other response is a failure):

| Item | Value | |
| --- | --- | --- |
| COB-ID | 0x7E4 | |
| Length | 3 | |
| Data 0 | 0x17 | (cs=23 for store configuration) |
| Data 1 | 0x00 | |
| Data 2 | 0x00 | |

- Set the module state to LSS-operation by **sending** the following message:
  (Note, the module will reset itself back to the pre-operational state)

| Item | Value | |
| --- | --- | --- |
| COB-ID | 0x7E5 | |
| Length | 2 | |
| Data 0 | 0x04 | (cs=4 for switch state global) |
| Data 1 | 0x00 | (switches to waiting state) |

### 3.1.1.2.    Setting Baudrate

- Set the module state to LSS-configuration by **sending** the following message:

| Item | Value | |
| --- | --- | --- |
| COB-ID | 0x7E5 | |
| Length | 2 | |
| Data 0 | 0x04 | (cs=4 for switch state global) |
| Data 1 | 0x01 | (switches to configuration state) |

- Set the baudrate by **sending** the following message:

| Item | Value | |
| --- | --- | --- |
| COB-ID | 0x7E5 | |
| Length | 3 | |
| Data 0 | 0x13 | (cs=19 for configure bit timing parameters) |
| Data 1 | 0x00 | (switches to waiting state) |
| Data 2 | Index | (select baudrate index per table 4) |

| Index | Bit Rate |
|-------|----------|
| 0 | 1 Mbit/s |
| 1 | 800 kbit/s |
| 2 | 500 kbit/s |
| 3 | 250 kbit/s |
| 4 | 125 kbit/s    (default) |
| 5 | reserved    (100 kbit/s) |
| 6 | 50 kbit/s |
| 7 | 20 kbit/s |
| 8 | 10 kbit/s |

**Table 2 – LSS Baudrate Indexes**

- The module will send the following response (any other response is a failure):

| Item | Value |
|------|-------|
| COB-ID | 0x7E4 |
| Length | 3 |
| Data 0 | 0x13          (cs=19 for configure bit timing parameters) |
| Data 1 | 0x00 |
| Data 2 | 0x00 |

- Activate bit timing parameters by **sending** the following message:

| Item | Value |
|------|-------|
| COB-ID | 0x7E5 |
| Length | 3 |
| Data 0 | 0x15          (cs=19 for activate bit timing parameters) |
| Data 1 | <delay_lsb> |
| Data 2 | <delay_msb> |

The delay individually defines the duration of the two periods of time to wait until the bit timing parameters switch is done (first period) and before transmitting any CAN message with the new bit timing parameters after performing the switch (second period). The time unit of switch delay is 1 ms.

- Save the configuration by **sending** the following message (on the NEW baudrate):

| Item | Value |
|------|-------|
| COB-ID | 0x7E5 |
| Length | 1 |
| Data 0 | 0x17          (cs=23 for store configuration) |

- The module will send the following response (any other response is a failure):

| Item | Value | |
|---|---|---|
| COB-ID | 0x7E4 | |
| Length | 3 | |
| Data 0 | 0x17 | (cs=23 for store configuration) |
| Data 1 | 0x00 | |
| Data 2 | 0x00 | |

- Set the module state to LSS-operation by **sending** the following message:
  (Note, the module will reset itself back to the pre-operational state)

| Item | Value | |
|---|---|---|
| COB-ID | 0x7E5 | |
| Length | 2 | |
| Data 0 | 0x04 | (cs=4 for switch state global) |
| Data 1 | 0x00 | (switches to waiting state) |

The following screen capture (left) shows the CAN data was sent (7E5h) and received (7E4h) by the tool when the baudrate was changed to 250 kbps using the LSS protocol. The other image (right) shows what was printed on the debug RS-232 menu while the operation took place.

Between CAN Frame 98 and 99, the baudrate on the CAN Scope tool was changed from 125 to 250 kbps.

## 3.2. COMMUNICATION OBJECTS (DS-301 and DS-404)

The communication objects supported by the RTD Scanner are listed in the following table. A more detailed description of some of the objects is given in the following subchapters. Only those objects that have device-profile specific information are described. For more information on the other objects, refer to the generic CANopen protocol specification DS-301.

| Index (hex) | Object | Object Type | Data Type | Access | PDO Mapping |
|---|---|---|---|---|---|
| 1000 | Device Type | VAR | UNSIGNED32 | RO | No |
| 1001 | Error Register | VAR | UNSIGNED8 | RO | No |
| 1002 | Manufacturer Status Register | VAR | UNSIGNED32 | RO | No |
| 1003 | Pre-Defined Error Field | ARRAY | UNSIGNED32 | RO | No |
| 100C | Guard Time | VAR | UNSIGNED16 | RW | No |
| 100D | Life Time Factor | VAR | UNSIGNED8 | RW | No |
| 1010 | Store Parameters | ARRAY | UNSIGNED32 | RW | No |
| 1011 | Restore Default Parameters | ARRAY | UNSIGNED32 | RW | No |
| 1016 | Consumer Heartbeat Time | ARRAY | UNSIGNED32 | RW | No |
| 1017 | Producer Heartbeat Time | VAR | UNSIGNED16 | RW | No |
| 1018 | Identity Object | RECORD | | RO | No |
| 1020 | Verify Configuration | ARRAY | UNSIGNED32 | RW | No |
| 1029 | Error Behaviour | ARRAY | UNSIGNED8 | RW | No |
| 1400 | RPDO1 Communication Parameter | RECORD | | RW | No |
| 1401 | RPDO2 Communication Parameter | RECORD | | RW | No |
| 1402 | RPDO3 Communication Parameter | RECORD | | RW | No |
| 1403 | RPDO4 Communication Parameter | RECORD | | RW | No |
| 1600 | RPDO1 Mapping Parameter | RECORD | | RO | No |
| 1601 | RPDO2 Mapping Parameter | RECORD | | RO | No |
| 1602 | RPDO3 Mapping Parameter | RECORD | | RO | No |
| 1603 | RPDO4 Mapping Parameter | RECORD | | RO | No |
| 1800 | TPDO1 Communication Parameter | RECORD | | RW | No |
| 1801 | TPDO2 Communication Parameter | RECORD | | RW | No |
| 1802 | TPDO3 Communication Parameter | RECORD | | RW | No |
| 1803 | TPDO4 Communication Parameter | RECORD | | RW | No |
| 1A00 | TPDO1 Mapping Parameter | RECORD | | RW | No |
| 1A01 | TPDO2 Mapping Parameter | RECORD | | RW | No |
| 1A02 | TPDO3 Mapping Parameter | RECORD | | RW | No |
| 1A03 | TPDO4 Mapping Parameter | RECORD | | RW | No |

Per the CANopen ® standard DS-301, the following procedure shall be used for re-mapping, and is the same for both RPDOs and TPDOs.

a) Destroy the PDO by setting bit **exists** (most significant bit) of sub-index 01h of the according PDO communication parameter to 1b

b) Disable mapping by setting sub-index 00h of the corresponding mapping object to 0

c) Modify the mapping by changing the values of the corresponding sub-indices

d) Enable mapping by setting sub-index 00h to the number of mapped objects

e) Create the PDO by setting bit **exists** (most significant bit) of sub-index 01h of the according PDO communication parameter to 0b

### 3.2.1.  Object 1000h: Device Type

This object contains information about the device type as per device profile DS-404. The 32-bit parameter is divided into two 16-bit values, showing General and Additional information as shown below.

MSB                                                                                                                      LSB

| **Additional Information** = 0x1002 | **General Information** = 0x0194 (404) |
|---|---|

DS-404 defines the Additional Information field in the following manner:
0000h = reserved
0001h = digital input block
0002h = analog input block
0004h = digital output block
0008h = analog output block
0010h = controller block
0020h = alarm block
0040h … 0800h = reserved
1000h ... 8000h = manufacturer-specific

The 0x10 in the MSB of the additional information indicates that this is the first CANopen ® RTD Scanner that has been manufactured by Axiomatic. The 0x02 in the LSB indicates that this module supports analog input blocks.

*Object Description*

| Index | 1000h |
|---|---|
| Name | Device Type |
| Object Type | VAR |
| Data Type | UNSIGNED32 |

*Entry Description*

| Access | RO |
|---|---|
| PDO Mapping | No |
| Value Range | 0x10020194 |
| Default Value | 0x10020194 |

### 3.2.2. Object 1001h: Error Register

This object is an error register for the device. Any time there is an error detected by the RTD Scanner, the Generic Error Bit (bit 0) is set. Only if there is no errors in the module will this bit will be cleared. No other bits in this register are used by the RTD Scanner.

*Object Description*

| Index | 1001h |
|---|---|
| Name | Error Register |
| Object Type | VAR |
| Data Type | UNSIGNED8 |

*Entry Description*

| Access | RO |
|---|---|
| PDO Mapping | No |
| Value Range | 00h or 01h |
| Default Value | 0 |

### 3.2.3. Object 1002h: Manufacturer Status Register

This object is used for manufacturer debug purposes.

### 3.2.4. Object 1003h: Pre-Defined Error Field

This object provides an error history by listing the errors in the order that they have occurred. An error is added to the top of the list when it occurs, and is immediately removed when the error condition has been cleared. The latest error is always at sub-index 1, with sub-index 0 containing the number of errors currently in the list. When the device is in an error-free state, the value of sub-index 0 is zero.

The error list may be cleared by writing a zero to sub-index 0, which will clear all errors from the list, regardless of whether or not they are still present. Clearing the list does NOT mean that the module will return to the error-free behaviour state if at least one error is still active.

The RTD Scanner has a limitation of a maximum of 17 errors in the list. If the device registers more errors, the list will be truncated, and the oldest entries will be lost.

The error codes stored in the list are 32-bit unsigned numbers, consisting of two 16-bit fields. The lower 16-bit field is the EMCY error code, and the higher 16-bit field is a manufacturer-specific code. The manufacturer-specific code is divided into two 8-bit fields, with the higher byte indicating the error description, and the lower byte indicating the channel on which the error occurred.

MSB                                                                LSB

| Error Description | Channel-ID | EMCY Error Code |
|---|---|---|

If node-guarding is used (not recommended per the latest standard) and a lifeguard event occurs, the manufacturer-specific field will be set to 0x1000. On the other hand, if a heartbeat consumer fails to be received within the expected timeframe, the Error Description will be set to 0x80 and the Channel-ID (nn) will reflect the Node-ID of the consumer channel that was not producing. In this case, the manufacturer-specific field will therefore be 0x80nn. In both cases, the corresponding EMCY Error Code will be the Guard Error 0x8130.

When a sensor fault is detected (i.e. open or short circuit) then the Error Description will reflect what kind of error is present using the following table. In these cases, the corresponding EMCY Error Code that will be used is the Input Overload 0xF001.

When a system fault is detected using the threshold objects ($4000, $4010 or $4020), then the Error Description will reflect which threshold was breached using the following table. In these cases, the corresponding EMCY Error Code that will be used is the Limit Exceed 0xF011.

In the unlikely event that a scanner fault is detected such as frozen data from the ADC chip or a math error occurred when calculating the temperature, then the Error Description will reflect the fault using the following table. In these cases, the corresponding EMCY Error Code that will be used is the Device Specific 0xFF00.

| Flag | Meaning |
|---|---|
| 0x02 | Positive Overload, i.e. Open Circuit |
| 0x04 | Negative Overload, i.e. Short Circuit |
| 0x10 | Low Warning Threshold Exceeded |
| 0x20 | High Warning Threshold Exceeded |
| 0x40 | High Shutdown Threshold Exceeded |
| 0x08 | Math Error Occurred |
| 0x80 | Frozen Data Detected from ADC Chip |

**Table 3 – Error Descriptions**

When a fault is detected, the corresponding Channel-ID will be 0x01 for RTD Input 1, 0x02 for RTD Input 2 … to 0x08 for RTD Input 8.

The EMCY Error Codes supported by this module are reflected in Table 4 shown below.

| Code | Meaning |
|---|---|
| 0x0000 | Error Reset (no errors) |
| 0x8110 | CAN Overrun |
| 0x8130 | Guard Error |
| 0xF001 | Input Overload – RTD Open or Short Circuit |
| 0xF011 | Limit Exceeded – See Table 3 above |
| 0xFF00 | Device Specific – See Table 3 above |

**Table 4 – EMCY Error Codes**

### Object Description

| Index | 1003h |
|---|---|
| Name | Pre-Defined Error Field |
| Object Type | ARRAY |
| Data Type | UNSIGNED32 |

### Entry Description

| Sub-Index | 0h |
|---|---|
| Description | Number of entries |
| Access | RW |
| PDO Mapping | No |
| Value Range | 0 to 17 |
| Default Value | 0 |

| Sub-Index | 1h to 17h |
|---|---|
| Description | Standard error field |
| Access | RO |
| PDO Mapping | No |
| Value Range | UNSIGNED32 |
| Default Value | 0 |

### 3.2.5.  Object 100Ch: Guard Time

The objects at index 100Ch and 100Dh shall indicate the configured guard time respective to the life time factor. The life time factor multiplied with the guard time gives the life time for the life guarding protocol described in DS-301. The Guard Time value shall be given in multiples of ms, and a value of 0000h shall disable the life guarding.

It should be noted that this object, and that of 100Dh are only supported for backwards compatibility. The standard recommends that newer networks do not use the life guarding protocol, but rather heartbeat monitoring instead. Both life guarding and heartbeats can NOT be active simultaneously.

*Object Description*

| Index | 100Ch |
|---|---|
| Name | Guard Time |
| Object Type | VAR |
| Data Type | UNSIGNED16 |

*Entry Description*

| Sub-Index | 0h |
|---|---|
| Access | RW |
| PDO Mapping | No |
| Value Range | 0 to 65535 |
| Default Value | 0 |

### 3.2.6.  Object 100Dh: Lifetime Factor

The life time factor multiplied with the guard time gives the life time for the life guarding protocol. A value of 00h shall disable life guarding.

*Object Description*

| Index | 100Dh |
|---|---|
| Name | Life time factor |
| Object Type | VAR |
| Data Type | UNSIGNED8 |

*Entry Description*

| Sub-Index | 0h |
|---|---|
| Access | RW |
| PDO Mapping | No |
| Value Range | 0 to 255 |
| Default Value | 0 |

### 3.2.7. Object 1010h: Store Parameters

This object supports the saving of parameters in non-volatile memory. In order to avoid storage of parameters by mistake, storage is only executed when a specific signature is written to the appropriate sub-index. The signature is "save".

The signature is a 32-bit unsigned number, composed of the ASCII codes of the signature characters, according to the following table:

| MSB | | | LSB |
|-----|-----|-----|-----|
| e | v | a | s |
| 65h | 76h | 61h | 73h |

On reception of the correct signature to an appropriate sub-index, the RTD Scanner will store the parameters in non-volatile memory, and then confirm the SDO transmission.

By read access, the object provides information about the module's saving capabilities. For all sub-indexes, this value is 1h, indicating that the RTD Scanner saves parameters on command. **This means that if power is removed before the Store object is written, changes to the Object Dictionary will NOT have been saved in the non-volatile memory, and will be lost on the next power cycle.**

### *Object Description*

| Index | 1010h |
|-------------|------------------|
| Name | Store Parameters |
| Object Type | ARRAY |
| Data Type | UNSIGNED32 |

### *Entry Description*

| Sub-Index | 0h |
|---------------|------------------------------|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 4 |
| Default Value | 4 |

| Sub-Index | 1h |
|---------------|-------------------------------|
| Description | Save all parameters |
| Access | RW |
| PDO Mapping | No |
| Value Range | 0x65766173 (write access) |
| | 1h            (read access) |
| Default Value | 1h |

| Sub-Index | 2h |
|---|---|
| Description | Save communication parameters |
| Access | RW |
| PDO Mapping | No |
| Value Range | 0x65766173 (write access) |
| | 1h           (read access) |
| Default Value | 1h |

| Sub-Index | 3h |
|---|---|
| Description | Save application parameters |
| Access | RW |
| PDO Mapping | No |
| Value Range | 0x65766173 (write access) |
| | 1h           (read access) |
| Default Value | 1h |

| Sub-Index | 4h |
|---|---|
| Description | Save manufacturer parameters |
| Access | RW |
| PDO Mapping | No |
| Value Range | 0x65766173 (write access) |
| | 1h           (read access) |
| Default Value | 1h |

### 3.2.8.    Object 1011h: Restore Parameters

This object supports the restoring of the default values for the object dictionary in non-volatile memory. In order to avoid restoring of parameters by mistake, the device restores the defaults only when a specific signature is written to the appropriate sub-index. The signature is "load".

The signature is a 32-bit unsigned number, composed of the ASCII codes of the signature characters, according to the following table:

| MSB | | | LSB |
|---|---|---|---|
| d | a | o | l |
| 64h | 61h | 6Fh | 6Ch |

On reception of the correct signature to an appropriate sub-index, the RTD Scanner will restore the defaults in non-volatile memory, and then confirm the SDO transmission. **The default values are set valid only after the device is reset or power-cycled.** This means that the RTD Scanner will NOT start using the default values right away, but rather continue to run from whatever values were in the Object Dictionary prior to the restore operation.

By read access, the object provides information about the module's default parameter restoring capabilities. For all sub-indexes, this value is 1h, indicating that the RTD Scanner restores defaults on command.

## Object Description

| | |
|---|---|
| Index | 1011h |
| Name | Restore Default Parameters |
| Object Type | ARRAY |
| Data Type | UNSIGNED32 |

## Entry Description

| | |
|---|---|
| Sub-Index | 0h |
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 4 |
| Default Value | 4 |

| | |
|---|---|
| Sub-Index | 1h |
| Description | Restore all default parameters |
| Access | RW |
| PDO Mapping | No |
| Value Range | 0x64616F6C (write access), 1h (read access) |
| Default Value | 1h |

| | |
|---|---|
| Sub-Index | 2h |
| Description | Restore default communication parameters |
| Access | RW |
| PDO Mapping | No |
| Value Range | 0x64616F6C (write access), 1h (read access) |
| Default Value | 1h |

| | |
|---|---|
| Sub-Index | 3h |
| Description | Restore default application parameters |
| Access | RW |
| PDO Mapping | No |
| Value Range | 0x64616F6C (write access), 1h (read access) |
| Default Value | 1h |

| | |
|---|---|
| Sub-Index | 4h |
| Description | Restore default manufacturer parameters |
| Access | RW |
| PDO Mapping | No |
| Value Range | 0x64616F6C (write access), 1h (read access) |
| Default Value | 1h |

### 3.2.9. Object 1016h: Consumer Heartbeat Time

The RTD Scanner can be a consumer of heartbeat objects for up to four modules. This object defines the expected heartbeat cycle time for those modules, and if set to zero, it is not used. When non-zero, the time is a multiple of 1ms, and monitoring will start after the reception of the first heartbeat from the module. If the RTD Scanner fails to receive a heartbeat from a node in the expected timeframe, it will indicate a communication error, and respond as per object 1029h.

| Bits | 31-24 | 23-16 | 15-0 |
|---|---|---|---|
| Value | Reserved 00h | Node-ID | Heartbeat time |
| Encoded as | | UNSIGNED8 | UNSIGNED16 |

*Object Description*

| Index | 1016h |
|---|---|
| Name | Consumer heartbeat time |
| Object Type | ARRAY |
| Data Type | UNSIGNED32 |

*Entry Description*

| Sub-Index | 0h |
|---|---|
| Description | Number of entries |
| Access | RO |
| PDO Mapping | No |
| Value Range | 4 |
| Default Value | 4 |

| Sub-Index | 1h to 4h |
|---|---|
| Description | Consumer heartbeat time |
| Access | RW |
| PDO Mapping | No |
| Value Range | UNSIGNED32 |
| Default Value | 0 |

### 3.2.10. Object 1017h: Producer Heartbeat Time

The RTD Scanner could be configured to produce a cyclical heartbeat by writing a non-zero value to this object. The value will be given in multiples of 1ms, and a value of 0 shall disable the heartbeat.

*Object Description*

| Index | 1017h |
|---|---|
| Name | Producer heartbeat time |
| Object Type | VAR |
| Data Type | UNSIGNED16 |

*Entry Description*

| Sub-Index | 0h |
|---|---|
| Access | RW |
| PDO Mapping | No |
| Value Range | 10 to 65535 |
| Default Value | 0 |

### 3.2.11. Object 1018h: Identity Object

The identity object indicates the data of the RTD Scanner, including vendor id, device id, software and hardware version numbers, and the serial number.

In the Revision Number entry at sub-index 3, the format of the data is as shown below

MSB                                                          LSB

| Major revision number (object dictionary) | Hardware Revision | Software Version |
|---|---|---|

*Object Description*

| Index | 1018h |
|---|---|
| Name | Identity Object |
| Object Type | RECORD |
| Data Type | Identity Record |

*Entry Description*

| Sub-Index | 0h |
|---|---|
| Description | Number of entries |
| Access | RO |
| PDO Mapping | No |
| Value Range | 4 |
| Default Value | 4 |

| Sub-Index | 1h |
|---|---|
| Description | Vendor ID |
| Access | RO |
| PDO Mapping | No |
| Value Range | 0x00000055 |
| Default Value | 0x00000055 (Axiomatic) |

| Sub-Index | 2h |
|---|---|
| Description | Product Code |
| Access | RO |
| PDO Mapping | No |
| Value Range | 0x00010106 |
| Default Value | 0x00010106 |

| Sub-Index | 3h |
|---|---|
| Description | Revision Number |
| Access | RO |
| PDO Mapping | No |
| Value Range | UNSIGNED32 |
| Default Value | No |

| Sub-Index | 4h |
|---|---|
| Description | Serial Number |
| Access | RO |
| PDO Mapping | No |
| Value Range | UNSIGNED32 |
| Default Value | No |

### 3.2.12. Object 1020h: Verify Configuration

This object can be read to see what date the software (version identified in object 1018h) was compiled. The date is represented as a hexadecimal value showing day/month/year as per the format below. The time value at sub-index 2 is a hexadecimal value showing the time in a 24 hour clock

MSB LSB

| Day (in 1-Byte Hex) | Month (in 1-Byte Hex) | Year (in 2-Byte Hex) |
|---|---|---|
| 00 | 00 | Time (in 2-Byte Hex) |

For example, a value of 0x10082010 would indicate that the software was compiled on August 10th, 2010. A time value of 0x00001620 would indicate it was compiled at 4:20pm.

*Object Description*

| Index | 1020h |
|---|---|
| Name | Verify configuration |
| Object Type | ARRAY |
| Data Type | UNSIGNED32 |

*Entry Description*

| Sub-Index | 0h |
|---|---|
| Description | Number of entries |
| Access | RO |
| PDO Mapping | No |
| Value Range | 2 |
| Default Value | 2 |

| | |
|---|---|
| Sub-Index | 1h |
| Description | Configuration date |
| Access | RO |
| PDO Mapping | No |
| Value Range | UNSIGNED32 |
| Default Value | No |

| | |
|---|---|
| Sub-Index | 2h |
| Description | Configuration time |
| Access | RO |
| PDO Mapping | No |
| Value Range | UNSIGNED32 |
| Default Value | No |

### 3.2.13. Object 1029h: Error Behaviour

This object controls the state that the RTD Scanner will be set into in case of an error of the type associated with the sub-index.

Communication errors are anything associate with the CAN network including life guard or heartbeat events, buffer overruns, busoff, etc.

Sensor errors are those associated with the sensor itself such as an open or short circuit.

System errors are those associate with the fault detection thresholds set by the used in objects $4010, $4020 and $4030.

Scanner errors are those associate with the performance of the device itself, such as detecting frozen data from the ADC chip, or returning a math error when converting resistance to temperature

#### *Object Description*
| | |
|---|---|
| Index | 1029h |
| Name | Error Behaviour |
| Object Type | ARRAY |
| Data Type | UNSIGNED8 |

#### *Entry Description*
| | |
|---|---|
| Sub-Index | 0h |
| Description | Number of entries |
| Access | RO |
| PDO Mapping | No |
| Value Range | 2 |
| Default Value | 2 |

| Sub-Index | 1h |
|---|---|
| Description | Communication Error |
| Access | RW |
| PDO Mapping | No |
| Value Range | 0 = Pre-Operational<br>1 = No State Change<br>2 = Stopped |
| Default Value | 0 (Pre-Operational) |

| Sub-Index | 2h |
|---|---|
| Description | Sensor Error |
| Access | RW |
| PDO Mapping | No |
| Value Range | 0 = Pre-Operational<br>1 = No State Change<br>2 = Stopped |
| Default Value | 1 (No State Change) |

| Sub-Index | 3h |
|---|---|
| Description | System Error |
| Access | RW |
| PDO Mapping | No |
| Value Range | 0 = Pre-Operational<br>1 = No State Change<br>2 = Stopped |
| Default Value | 1 (No State Change) |

| Sub-Index | 4h |
|---|---|
| Description | Scanner Error |
| Access | RW |
| PDO Mapping | No |
| Value Range | 0 = Pre-Operational<br>1 = No State Change<br>2 = Stopped |
| Default Value | 0 (Pre-Operational) |

### 3.2.14. RPDO Behaviour

The RTD Scanner can support up to four RPDO messages, but in reality, it does not used them. The other RPDO objects are provided simply for compliance with the standard CANopen ® Object Dictionary, but are disabled on this module (mapping objects are read-only)

All RPDOs on the RTD Scanner use the same default communication parameters, with the PDO IDs set according to the pre-defined connection set described in DS-301. All RPDOs do not exist, there is no RTR allowed, they use 11-bit CAN-IDs (base frame valid) and are event-driven.

### Object Description

| | |
|---|---|
| Index | 1400h to 1403h |
| Name | RPDO communication parameter |
| Object Type | RECORD |
| Data Type | PDO Communication Record |

### Entry Description

| | |
|---|---|
| Sub-Index | 0h |
| Description | Number of entries |
| Access | RO |
| PDO Mapping | No |
| Value Range | 5 |
| Default Value | 5 |

| | | | X | RPDOx ID |
|---|---|---|---|---|
| Sub-Index | 1h | | | |
| Description | COB-ID used by RPDO | | | |
| Access | RW | | 1 | 0200h |
| PDO Mapping | No | | 2 | 0300h |
| Value Range | See value definition in DS-301 | | 3 | 0400h |
| Default Value | C0000000h + RPDOx + Node-ID | | 4 | 0500h |

Node-ID = Node-ID of the module.
C0000000h in the COB-ID indicates that the PDO does not exist and no RTR is allowed

| | |
|---|---|
| Sub-Index | 2h |
| Description | Transmission type |
| Access | RO |
| PDO Mapping | No |
| Value Range | See value definition in DS-301 |
| Default Value | 255 (FFh) = Event Driven |

| | |
|---|---|
| Sub-Index | 3h |
| Description | Inhibit Time |
| Access | RW |
| PDO Mapping | No |
| Value Range | See value definition in DS-301 |
| Default Value | 0 |

| | |
|---|---|
| Sub-Index | 4h |
| Description | Compatibility entry |
| Access | RW |
| PDO Mapping | No |
| Value Range | UNSIGNED8 |
| Default Value | 0 |

| | |
|---|---|
| Sub-Index | 5 |
| Description | Event-timer |
| Access | RW |
| PDO Mapping | No |
| Value Range | See value definition in DS-301 |
| Default Value | 0 |

### 3.2.15. TPDO Behaviour

The RTD Scanner can support up to four TPDO messages, and all are enabled by default.

**TPDO1 Mapping at Object 1A00h: Default ID 0x180 + Node ID**

| Sub-Index | Value | Object |
|---|---|---|
| 0 | 4 | Number of mapped application objects in PDO |
| 1 | 0x71300110 | RTD Input 1 Process Value |
| 2 | 0x71300210 | RTD Input 2 Process Value |
| 3 | 0x71300310 | RTD Input 3 Process Value |
| 4 | 0x71300410 | RTD Input 4 Process Value |

**TPDO2 Mapping at Object 1A01h: Default ID 0x280 + Node ID**

| Sub-Index | Value | Object |
|---|---|---|
| 0 | 4 | Number of mapped application objects in PDO |
| 1 | 0x71300510 | RTD Input 5 Process Value |
| 2 | 0x71300610 | RTD Input 6 Process Value |
| 3 | 0x71300710 | RTD Input 7 Process Value |
| 4 | 0x71300810 | RTD Input 8 Process Value |

**TPDO3 Mapping at Object 1A02h: Default ID 0x380 + Node ID**

| Sub-Index | Value | Object |
|---|---|---|
| 0 | 3 | Number of mapped application objects in PDO |
| 1 | 0x21300110 | Average Value of Bank 1 RTDs (1 to 4) |
| 2 | 0x21300210 | Average Value of Bank 2 RTDs (5 to 8) |
| 3 | 0x21300310 | Average Value of All RTDs |
| 4 | 0 | Not used by default |

**TPDO4 Mapping at Object 1A03h: Default ID 0x480 + Node ID**

| Sub-Index | Value | Object |
|---|---|---|
| 0 | 2 | Number of mapped application objects in PDO |
| 3 | 0x50000020 | Power Supply Measured |
| 3 | 0x50010020 | Current Source Measured |
| 3 | 0 | Not used by default |
| 4 | 0 | Not used by default |

*(Note: This TPDO is not transmitted by default, as the event-timer in object $1803 is set to 0)*

All TPDOs on the RTD Scanner use the same default communication parameters, with the PDO IDs set according to the pre-defined connection set described in DS-301. All TPDOs are event-driven, there is no RTR allowed, and they use 11-bit CAN-IDs (base frame valid).

**Object Description**

| Index | 1800h to 1807h |
|---|---|
| Name | TPDO communication parameter |
| Object Type | RECORD |
| Data Type | PDO Communication Record |

### *Entry Description*

| Sub-Index | 0h |
|---|---|
| Description | Number of entries |
| Access | RO |
| PDO Mapping | No |
| Value Range | 5 |
| Default Value | 5 |

| Sub-Index | 1h |
|---|---|
| Description | COB-ID used by TPDO |
| Access | RW |
| PDO Mapping | No |
| Value Range | See value definition in DS-301 |
| Default Value | 40000000h + TPDOx + Node-ID |

| X | TPDOx ID |
|---|---|
| 1 | 0180h |
| 2 | 0280h |
| 3 | 0380h |
| 4 | 0480h |

Node-ID = Node-ID of the module. The TPDO COB-IDs are automatically updated if the Node-ID is changed by LSS protocol.
80000000h in the COB-ID would indicates that the PDO does not exist (destroyed)
40000000h in the COB-ID indicates that there is no RTR allowed on the PDO

| Sub-Index | 2h |
|---|---|
| Description | Transmission type |
| Access | RO |
| PDO Mapping | No |
| Value Range | See value definition in DS-301 |
| Default Value | 254 (FEh) = Event Driven |

| Sub-Index | 3h |
|---|---|
| Description | Inhibit Time |
| Access | RW |
| PDO Mapping | No |
| Value Range | See value definition in DS-301 |
| Default Value | 0 |

| Sub-Index | 4h |
|---|---|
| Description | Compatibility entry |
| Access | RW |
| PDO Mapping | No |
| Value Range | UNSIGNED8 |
| Default Value | 0 |

| Sub-Index | 5 |
|---|---|
| Description | Event-timer |
| Access | RW |
| PDO Mapping | No |
| Value Range | See value definition in DS-301 |
| Default Value | 1000ms (on objects 1800h to 1802h) 0ms     (on object 1803h) |

## 3.3. APPLICATION OBJECTS (DS-404)

| Index (hex) | Object | Object Type | Data Type | Access | PDO Mapping |
|---|---|---|---|---|---|
| 6100 | RTD Input Field Value | ARRAY | FLOAT32 | RO | Yes |
| 6110 | RTD Sensor Type | ARRAY | UNSIGNED16 | RW | No |
| 6112 | RTD Operating Mode | ARRAY | UNSIGNED8 | RW | No |
| 6114 | ADC Sampling Rate | ARRAY | UNSIGNED32 | RO | No |
| 6126 | RTD Scaling Factor | ARRAY | FLOAT32 | RW | No |
| 6127 | RTD Scaling Offset | ARRAY | FLOAT32 | RW | No |
| 7130 | RTD Input Process Value | ARRAY | INTEGER16 | RO | Yes |
| 6150 | RTD Status | ARRAY | UNSIGNED8 | RO | Yes |

### 3.3.1. Object 6100h: RTD Input Field Value

This read-only object represents the measured temperature of the associated RTD input in Degrees Celsius. When the corresponding sub-index in object $2000 is set to TRUE, the calculated temperature is automatically adjusted for the cold junction temperature compensation.

### *Object Description*

| | |
|---|---|
| Index | 6100h |
| Name | RTD Input Field Value |
| Object Type | ARRAY |
| Data Type | FLOAT32 |

### *Entry Description*

| | |
|---|---|
| Sub-Index | 0h |
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 8 |
| Default Value | 8 |

| | |
|---|---|
| Sub-Index | 1h to 8h (X = 1 to 8) |
| Description | RTD Input X Temperature |
| Access | RO |
| PDO Mapping | Yes |
| Value Range | -273°C to 1735°C |
| Default Value | No |

### 3.3.2. Object 6110h: RTD Sensor Type

This object determines what kind of RTD is connected to the input. The formulas used to calculate the resistance of the sensor based on the voltage readings taken on the various input pins are shown in Figure 1. The following sensor types are supported by the scanner.

| Value | Meaning |
|-------|---------|
| 10000 | Two wire sensor |
| 10001 | Three wire sensor |
| 10002 | Four wire sensor |

**Table 5 – Supported RTD Types**

*Object Description*

| Index | 6110h |
|-------|-------|
| Name | RTD Sensor Type |
| Object Type | ARRAY |
| Data Type | UNSIGNED16 |

*Entry Description*

| Sub-Index | 0h |
|-----------|-----|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 8 |
| Default Value | 8 |

| Sub-Index | 1h to 8h (X = 1 to 8) |
|-----------|------------------------|
| Description | RTD X Sensor Type |
| Access | RW |
| PDO Mapping | No |
| Value Range | See Table 5 |
| Default Value | 10001 (3-wire) |

### 3.3.3. Object 6112h: RTD Operating Mode

This object establishes whether a RTD input is active. When a sub-index is set to zero (0) "Channel off," the associate input is disabled. As described in section 1.2, the FV for the input is loaded with 0xFFFFFFFF (not a number) and the PV with 0xFFFF (-1). All error flags associated with the channel are cleared. This channel is also ignored in any averaging.

By default, all eight channels have this object set to one (1) "Normal Operation."

*Object Description*

| Index | 6112h |
|-------|-------|
| Name | RTD Operating Mode |
| Object Type | ARRAY |
| Data Type | UNSIGNED8 |

### Entry Description

| | |
|---|---|
| Sub-Index | 0h |
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 8 |
| Default Value | 8 |

| | |
|---|---|
| Sub-Index | 1h to 8h (X = 1 to 8) |
| Description | RTD X Operating Mode |
| Access | RW |
| PDO Mapping | No |
| Value Range | 0 or 1 |
| Default Value | 1 (Normal Operation) |

## 3.3.4.    Object 6114h: ADC Sample Rate

This read-only object is automatically updated when an input channel is disabled by object $6112. When read, it reflects the approximate time between scan updates for any given channel, in milliseconds.

### Object Description

| | |
|---|---|
| Index | 6114h |
| Name | ADC Sample Rate |
| Object Type | ARRAY |
| Data Type | UNSIGNED32 |

### Entry Description

| | |
|---|---|
| Sub-Index | 0h |
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 8 |
| Default Value | 8 |

| | |
|---|---|
| Sub-Index | 1h to 8h (X = 1 to 8) |
| Description | RTD X Sample Rate |
| Access | RO |
| PDO Mapping | No |
| Value Range | 100ms to 900ms |
| Default Value | 900ms |

### 3.3.5. Object 6126h: RTD Scaling Factor

This object represents the scaling factor by which the RTD field value is multiplied to get the process value.

**Object Description**

| Index | 6126h |
|---|---|
| Name | RTD Scaling Factor |
| Object Type | ARRAY |
| Data Type | FLOAT32 |

**Entry Description**

| Sub-Index | 0h |
|---|---|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 8 |
| Default Value | 8 |

| Sub-Index | 1h to 8h (X = 1 to 8) |
|---|---|
| Description | RTD X Scaling Factor |
| Access | RW |
| PDO Mapping | No |
| Value Range | -1000.0 to 1000.0 |
| Default Value | 16.0   [0.0625°C/bit] |

### 3.3.6. Object 6127h: RTD Scaling Offset

This object represents the scaling offset which is added to the scaled RTD field value to get the process value.

$$Process\ Value = (Field\ Value * Scaling\ Factor) + Scaling\ Offset$$

**Object Description**

| Index | 6127h |
|---|---|
| Name | RTD Scaling Offset |
| Object Type | ARRAY |
| Data Type | FLOAT32 |

**Entry Description**

| Sub-Index | 0h |
|---|---|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 8 |
| Default Value | 8 |

| Sub-Index | 1h to 8h (X = 1 to 8) |
|---|---|
| Description | RTD X Scaling Offset |
| Access | RW |
| PDO Mapping | No |
| Value Range | -10000.0 to 10000.0 |
| Default Value | 4368.0   (-273°C * 16) |

### 3.3.7. Object 7130h: RTD Input Process Value

This read-only object represents the scaled value of the measured RTD temperature.

*Object Description*

| Index | 7130h |
|---|---|
| Name | RTD Input Process Value |
| Object Type | ARRAY |
| Data Type | INTEGER16 |

*Entry Description*

| Sub-Index | 0h |
|---|---|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 8 |
| Default Value | 8 |

| Sub-Index | 1h to 8h (X = 1 to 8) |
|---|---|
| Description | RTD X Process Value |
| Access | RO |
| PDO Mapping | Yes |
| Value Range | -32768 to 32767 |
| Default Value | No |

### 3.3.8. Object 6150h: RTD Status

This read-only object reflects the status of the associated RTD input field value. When set to zero, it indicates that the data measured is within normal operating limits. When the least significant bit in the byte is set to 1, it indicates that the data is outside of the normal operating range. Other bits in the byte are set as per the following bit combinations.

| Frozen Data | High Shutdown | High Warn | Low Warn | Math Error | Negative Overload | Positive Overload | Data Not Valid |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| Value | Meaning |
|---|---|
| 00h | Data Valid and within normal operating range |
| 03h | Positive Overload, i.e. Open Circuit |
| 05h | Negative Overload, i.e. Short Circuit |
| 09h | Math Error, Data not valid |
| 10h | Data Valid, but outside of normal range, low warning |
| 20h | Data Valid, but outside of normal range, high warning |
| 41h | Data too high, shutdown |
| 81h | ADC not responding, 'frozen' data |

**Table 6 – RTD Status Values**

## *Object Description*

| Index | 6150h |
|---|---|
| Name | RTD Status |
| Object Type | ARRAY |
| Data Type | UNSIGNED8 |

## *Entry Description*

| Sub-Index | 0h |
|---|---|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 8 |
| Default Value | 8 |

| Sub-Index | 1h to 8h (X = 1 to 8) |
|---|---|
| Description | RTD X Status |
| Access | RO |
| PDO Mapping | Yes |
| Value Range | See Table 6 |
| Default Value | 0 (Data Valid) |

## 3.4. MANUFACTURER OBJECTS

| Index (hex) | Object | Object Type | Data Type | Access | PDO Mapping |
|---|---|---|---|---|---|
| 2000 | RTD Resistance | ARRAY | FLOAT32 | RO | No |
| 2010 | RTD Microvolts | ARRAY | INTEGER32 | RO | No |
| 2100 | Average Input Field Value | ARRAY | FLOAT32 | RO | Yes |
| 2112 | Average Operating Mode | ARRAY | UNSIGNED8 | RW | No |
| 2126 | Average Scaling Factor | ARRAY | FLOAT32 | RW | No |
| 2127 | Average Scaling Offset | ARRAY | FLOAT32 | RW | No |
| 2130 | Average Input Process Value | ARRAY | INTEGER16 | RO | Yes |
| 3000 | RTD Coefficient | ARRAY | UNSIGNED16 | RW | No |
| 3010 | Callendar-Van Dusen Constant A | ARRAY | FLOAT32 | RW | No |
| 3020 | Callendar-Van Dusen Constant B | ARRAY | FLOAT32 | RW | No |
| 3030 | Callendar-Van Dusen Constant C | ARRAY | FLOAT32 | RW | No |
| 4000 | Low Temperature Warning Threshold | ARRAY | INTEGER16 | RW | No |
| 4010 | High Temperature Warning Threshold | ARRAY | INTEGER16 | RW | No |
| 4020 | High Temperature Shutdown Threshold | ARRAY | INTEGER16 | RW | No |
| 4030 | Error React Delay | ARRAY | UNSIGNED16 | RW | No |
| 5000 | Power Supply Measured | VAR | FLOAT32 | RO | Yes |
| 5001 | Current Source Measured | VAR | FLOAT32 | RO | Yes |
| 5010 | ADC Filter Frequency | VAR | UNSIGNED8 | RW | No |
| 5555 | Start in Operational Mode | VAR | BOOLEAN | RW | No |

### 3.4.1. Object 2000h: RTD Resistance

This read-only object is available for diagnostic purposes. It reflects the measured resistance of the sensor input .

### Object Description

| Index | 2000h |
|---|---|
| Name | RTD Resistance |
| Object Type | ARRAY |
| Data Type | FLOAT32 |

### Entry Description

| Sub-Index | 0h |
|---|---|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 8 |
| Default Value | 8 |

| Sub-Index | 1h to 8h (X = 1 to 8) |
|---|---|
| Description | RTD X Resistance Measured |
| Access | RO |
| PDO Mapping | No |
| Value Range | Dependent on Sensor Type |
| Default Value | No |

### 3.4.2. Object 2010h: RTD Microvolts

This read-only object is available for diagnostic purposes. It reflects the raw microvolt reading measured by the ADC chip directly.

*Object Description*

| Index | 2010h |
|---|---|
| Name | RTD Microvolts |
| Object Type | ARRAY |
| Data Type | INTEGER32 |

*Entry Description*

| Sub-Index | 0h |
|---|---|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 8 |
| Default Value | 8 |

| Sub-Index | 1h to 8h (X = 1 to 8) |
|---|---|
| Description | RTD X Microvolt Reading |
| Access | RO |
| PDO Mapping | No |
| Value Range | Dependent on Sensor Type |
| Default Value | No |

### 3.4.3. Object 2100h: Average Input Field Value

This read-only object reflects the calculated average of a given subset of RTD inputs. Channels with invalid input data (i.e. disabled, open/short circuited or frozen) are not used in the average calculations. The average input field values are calculated by adding the temperature of all valid channels, then dividing by the number of valid inputs.

*Object Description*

| Index | 2100h |
|---|---|
| Name | Average Input Field Value |
| Object Type | ARRAY |
| Data Type | FLOAT32 |

*Entry Description*

| Sub-Index | 0h |
|---|---|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 3 |
| Default Value | 3 |

| Sub-Index | 1h |
|---|---|
| Description | Bank 1 Average FV (RTD1 to RTD4) |
| Access | RO |
| PDO Mapping | Yes |
| Value Range | -273°C to 1735°C |
| Default Value | No |

| Sub-Index | 2h |
|---|---|
| Description | Bank 2 Average FV (RTD5 to RTD8) |
| Access | RO |
| PDO Mapping | Yes |
| Value Range | -273°C to 1735°C |
| Default Value | No |

| Sub-Index | 3h |
|---|---|
| Description | Total Average FV |
| Access | RO |
| PDO Mapping | Yes |
| Value Range | -273°C to 1735°C |
| Default Value | No |

### 3.4.4.    Object 2112h: Average Operating Mode

This object determines if the averaging calculation is performed for the given sub-index. When set to one (1) "Normal Operation", averaging is performed. When set to zero (0) "Channel off", the average for the bank(s) is not available.

#### *Object Description*

| Index | 2112h |
|---|---|
| Name | Average Operating Mode |
| Object Type | ARRAY |
| Data Type | UNSIGNED8 |

#### *Entry Description*

| Sub-Index | 0h |
|---|---|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 3 |
| Default Value | 3 |

| Sub-Index | 1h |
|---|---|
| Description | Bank 1 Average Operation |
| Access | RW |
| PDO Mapping | No |
| Value Range | 0 or 1 |
| Default Value | 1 (TRUE) |

| Sub-Index | 2h |
|---|---|
| Description | Bank 2 Average Operation |
| Access | RW |
| PDO Mapping | No |
| Value Range | 0 or 1 |
| Default Value | 1 (TRUE) |

| Sub-Index | 3h |
|---|---|
| Description | Total Average Operation |
| Access | RW |
| PDO Mapping | No |
| Value Range | 0 or 1 |
| Default Value | 1 (TRUE) |

### 3.4.5. Object 2126h: Average Scaling Factor

This object represents the scaling factor by which the average field value is multiplied to get the process value.

*Object Description*

| Index | 2126h |
|---|---|
| Name | Average Scaling Factor |
| Object Type | ARRAY |
| Data Type | FLOAT32 |

*Entry Description*

| Sub-Index | 0h |
|---|---|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 3 |
| Default Value | 3 |

| Sub-Index | 1h |
|---|---|
| Description | Bank 1 Average Scaling Factor |
| Access | RW |
| PDO Mapping | No |
| Value Range | -1000.0 to 1000.0 |
| Default Value | 16.0   [0.0625°C/bit] |

| Sub-Index | 2h |
|---|---|
| Description | Bank 2 Average Scaling Factor |
| Access | RW |
| PDO Mapping | No |
| Value Range | -1000.0 to 1000.0 |
| Default Value | 16.0   [0.0625°C/bit] |

| | |
|---|---|
| Sub-Index | 3h |
| Description | Total Average Scaling Factor |
| Access | RW |
| PDO Mapping | No |
| Value Range | -1000.0 to 1000.0 |
| Default Value | 16.0   [0.0625°C/bit] |

### 3.4.6.   Object 2126h: Average Scaling Offset

This object represents the scaling offset which is added to the scaled average field value to get the process value.

$$Process\ Value = (Field\ Value * Scaling\ Factor) + Scaling\ Offset$$

*Object Description*

| | |
|---|---|
| Index | 2127h |
| Name | Average Scaling Offset |
| Object Type | ARRAY |
| Data Type | FLOAT32 |

*Entry Description*

| | |
|---|---|
| Sub-Index | 0h |
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 3 |
| Default Value | 3 |

| | |
|---|---|
| Sub-Index | 1h |
| Description | Bank 1 Average Scaling Offset |
| Access | RW |
| PDO Mapping | No |
| Value Range | -10000.0 to 10000.0 |
| Default Value | 4368.0   (-273°C * 16) |

| | |
|---|---|
| Sub-Index | 2h |
| Description | Bank 2 Average Scaling Offset |
| Access | RW |
| PDO Mapping | No |
| Value Range | -10000.0 to 10000.0 |
| Default Value | 4368.0   (-273°C * 16) |

| | |
|---|---|
| Sub-Index | 3h |
| Description | Total Average Scaling Offset |
| Access | RW |
| PDO Mapping | No |
| Value Range | -10000.0 to 10000.0 |
| Default Value | 4368.0   (-273°C * 16) |

### 3.4.7. Object 2130h: Average Input Process Value

This read-only object represents the scaled value of the measured average temperature.

*Object Description*

| Index | 2130h |
|---|---|
| Name | Average Input Process Value |
| Object Type | ARRAY |
| Data Type | FLOAT32 |

*Entry Description*

| Sub-Index | 0h |
|---|---|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 3 |
| Default Value | 3 |

| Sub-Index | 1h |
|---|---|
| Description | Bank 1 Average PV (RTD1 to RTD4) |
| Access | RO |
| PDO Mapping | Yes |
| Value Range | -32768 to 32767 |
| Default Value | No |

| Sub-Index | 2h |
|---|---|
| Description | Bank 2 Average PV (RTD5 to RTD8) |
| Access | RO |
| PDO Mapping | Yes |
| Value Range | -32768 to 32767 |
| Default Value | No |

| Sub-Index | 3h |
|---|---|
| Description | Total Average PV |
| Access | RO |
| PDO Mapping | Yes |
| Value Range | -32768 to 32767 |
| Default Value | No |

### 3.4.8. Object 3000h: RTD Coefficient

This object determines the Callendar-Van Dusen constants that will be used in the resistance to temperature calculation. See Table 1 in section 1.2 for the relationship between the supported RTD Coefficients, and the Callendar-Van Dusen constants that are used. Whenever this object is written with an non-zero value, the data in objects $3010, $3020 and $3030 are automatically updated per Table 1.

| Value | Meaning |
|---|---|
| 0 | USER DEFINED |
| 1 | IEC 00385 |
| 2 | JIS 003916 |
| 3 | US 003902 |
| 4 | Legacy US 003920 |
| 5 | SAMA 003923 |

**Table 7 - Supported RTD Coefficients**

*Object Description*

| Index | 3000h |
|---|---|
| Name | RTD Coefficient |
| Object Type | ARRAY |
| Data Type | UNSIGNED16 |

*Entry Description*

| Sub-Index | 0h |
|---|---|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 8 |
| Default Value | 8 |

| Sub-Index | 1h to 8h (X = 1 to 8) |
|---|---|
| Description | RTD X Coefficient |
| Access | RW |
| PDO Mapping | No |
| Value Range | See Table 7 |
| Default Value | 1 (IEC 00385) |

### 3.4.9. Object 3010h: Callendar-Van Dusen Constant A

This object is used in the resistance to temperature conversion function per the following relationship. The data in this object is multiplied by $10^{-3}$ such that it will represent the variable A in the formula.

$$R_t = R_o[1+At+Bt^2+C(t-100)t^3]$$ **in which C is only applicable when t < 0 °C.**

When the RTD coefficient selected in object $3000 is a non-zero value (i.e. not user defined), this object is treated as a read-only parameter. When object $3000 changes, this object is automatically updated to the appropriate value as shown in column 2 of Table 1.

*Object Description*

| Index | 3010h |
|---|---|
| Name | Callendar-Van Dusen Constant A |
| Object Type | ARRAY |
| Data Type | FLOAT32 |

*Entry Description*

| Sub-Index | 0h |
|---|---|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 8 |
| Default Value | 8 |

| Sub-Index | 1h to 8h (X = 1 to 8) |
|---|---|
| Description | RTD X Callendar-Van Dusen A |
| Access | RW (only when corresponding sub-index in object 3000 is zero) |
| PDO Mapping | No |
| Value Range | -9.9 to 9.9 |
| Default Value | 3.90830 |

### 3.4.10. Object 3020h: Callendar-Van Dusen Constant B

This object is used in the resistance to temperature conversion function per the following relationship. The data in this object is multiplied by $10^{-7}$ such that it will represent the variable B in the formula.

$$R_t = R_o[1+At+Bt^2+C(t-100)t^3] \text{ in which C is only applicable when t < 0 °C.}$$

When the RTD coefficient selected in object \$3000 is a non-zero value (i.e. not user defined), this object is treated as a read-only parameter. When object \$3000 changes, this object is automatically updated to the appropriate value as shown in column 3 of Table 1.

*Object Description*

| Index | 3020h |
|---|---|
| Name | Callendar-Van Dusen Constant B |
| Object Type | ARRAY |
| Data Type | FLOAT32 |

*Entry Description*

| Sub-Index | 0h |
|---|---|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 8 |
| Default Value | 8 |

| Sub-Index | 1h to 8h (X = 1 to 8) |
|---|---|
| Description | RTD X Callendar-Van Dusen B |
| Access | RW (only when corresponding sub-index in object 3000 is zero) |
| PDO Mapping | No |
| Value Range | -9.9 to 9.9 |
| Default Value | -5.77500 |

### 3.4.11. Object 3030h: Callendar-Van Dusen Constant C

This object is used in the resistance to temperature conversion function per the following relationship. The data in this object is multiplied by $10^{-12}$ such that it will represent the variable C in the formula.

$$R_t = R_o[1+At+Bt^2+C(t-100)t^3] \text{ in which C is only applicable when t < 0 °C.}$$

When the RTD coefficient selected in object \$3000 is a non-zero value (i.e. not user defined), this object is treated as a read-only parameter. When object \$3000 changes, this object is automatically updated to the appropriate value as shown in column 4 of Table 1.

*Object Description*

| Index | 3030h |
|---|---|
| Name | Callendar-Van Dusen Constant B |
| Object Type | ARRAY |
| Data Type | FLOAT32 |

*Entry Description*

| Sub-Index | 0h |
|---|---|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 8 |
| Default Value | 8 |

| Sub-Index | 1h to 8h (X = 1 to 8) |
|---|---|
| Description | RTD X Callendar-Van Dusen C |
| Access | RW (only when corresponding sub-index in object 3000 is zero) |
| PDO Mapping | No |
| Value Range | -9.9 to 9.9 |
| Default Value | -4.18301 |

### 3.4.12. Object 4000h: Low Temperature Warning Threshold

This object sets the low threshold of the normal operating temperature range for each RTD's input field value ($6100.) Any FV value below this limit flags a low warning in the corresponding RTD's status byte ($6150), as well as setting an EMCY code in the pre-defined error field ($1003). The data in sub-indexes 1 to 8 is interpreted with a fixed resolution of 0.1°C/bit.

*Object Description*

| Index | 4000h |
|---|---|
| Name | Low Temperature Warning Threshold |
| Object Type | ARRAY |
| Data Type | INTEGER16 |

*Entry Description*

| Sub-Index | 0h |
|---|---|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 8 |
| Default Value | 8 |

| Sub-Index | 1h to 8h (X = 1 to 8) |
|---|---|
| Description | RTD X Low Warning Temp. |
| Access | RW |
| PDO Mapping | No |
| Value Range | -2730 to (RTD X High Warning Temp – 100) |
| Default Value | -200 [-20°C] |

### 3.4.13. Object 4010h: High Temperature Warning Threshold

This object sets the high threshold of the normal operating temperature range for each RTD's input field value ($6100.) Any FV value above this limit flags a high warning in the corresponding RTD's status byte ($6150), as well as setting an EMCY code in the pre-defined error field ($1003). The data in sub-indexes 1 to 8 is interpreted with a fixed resolution of 0.1°C/bit.

*Object Description*

| Index | 4010h |
|---|---|
| Name | High Temperature Warning Threshold |
| Object Type | ARRAY |
| Data Type | INTEGER16 |

*Entry Description*

| Sub-Index | 0h |
|---|---|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 8 |
| Default Value | 8 |

| Sub-Index | 1h to 8h (X = 1 to 8) |
|---|---|
| Description | RTD X High Warning Temp. |
| Access | RW |
| PDO Mapping | No |
| Value Range | (RTD X Low Warning Temp + 100) to (RTD X High Shutdown Temp – 100) |
| Default Value | 1250 [125°C] |

### 3.4.14.  Object 4020h: High Temperature Shutdown Threshold

This object sets the high threshold of the shutdown operating temperature range for each RTD's input field value ($6100.) Any FV value above this limit flags a high shutdown in the corresponding RTD's status byte ($6150), as well as setting an EMCY code in the pre-defined error field ($1003). The data in sub-indexes 1 to 8 is interpreted with a fixed resolution of 0.1°C/bit.

*Object Description*

| Index | 4020h |
|---|---|
| Name | High Temperature Shutdown Threshold |
| Object Type | ARRAY |
| Data Type | INTEGER16 |

*Entry Description*

| Sub-Index | 0h |
|---|---|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 8 |
| Default Value | 8 |

| Sub-Index | 1h to 8h (X = 1 to 8) |
|---|---|
| Description | RTD X High Shutdown Temp. |
| Access | RW |
| PDO Mapping | No |
| Value Range | (RTD X High Warning Temp + 100) to 17350 |
| Default Value | 2500 [250°C] |

### 3.4.15.  Object 4030h: Error React Delay

This object defines the length of time during which a RTD's input field value must remain outside of the operating ranges as defined in objects $4000, $4010, and $4020. Should the temperature go back within the acceptable range during this period, the fault will not be flagged in the status byte ($6150), nor will an EMCY code be generated in the pre-defined error field ($1003).

*Object Description*

| Index | 4030h |
|---|---|
| Name | Error React Delay |
| Object Type | ARRAY |
| Data Type | UNSIGNED16 |

### Entry Description

| Sub-Index | 0h |
|---|---|
| Description | Largest sub-index supported |
| Access | RO |
| PDO Mapping | No |
| Value Range | 8 |
| Default Value | 8 |

| Sub-Index | 1h to 8h (X = 1 to 8) |
|---|---|
| Description | RTD X Error React Delay |
| Access | RW |
| PDO Mapping | No |
| Value Range | 0ms to 60000ms |
| Default Value | 5000ms |

## 3.4.16. Object 5000h: Power Supply Measured

This read-only object reflects the value, in volts, of the supply powering the scanner

### Object Description

| Index | 5000h |
|---|---|
| Name | Power Supply Measured |
| Object Type | VAR |
| Data Type | FLOAT32 |

### Entry Description

| Sub-Index | 0h |
|---|---|
| Access | RO |
| PDO Mapping | Yes |
| Value Range | 8.0V to 60.0V |
| Default Value | No |

## 3.4.17. Object 5001h: Current Source Measured

This read-only object reflects the value, in mA, of the current source multiplexed to each resistive input in order to generate a measurable voltage across the sensor.

### Object Description

| Index | 5001h |
|---|---|
| Name | Current Source Measured |
| Object Type | VAR |
| Data Type | FLOAT32 |

### Entry Description

| Sub-Index | 0h |
|---|---|
| Access | RO |
| PDO Mapping | Yes |
| Value Range | 3.8mA to 4.2mA |
| Default Value | No |

### 3.4.18. Object 5010h: ADC Filter Frequency

This object defined the filter cutoff frequency used by the 24-bit analog-to-digital converter.

**Object Description**

| Index | 5010h |
|---|---|
| Name | ADC Filter Frequency |
| Object Type | VAR |
| Data Type | UNSIGNED8 |

**Entry Description**

| Sub-Index | 0h |
|---|---|
| Access | RW |
| PDO Mapping | No |
| Value Range | 50Hz or 60Hz |
| Default Value | 60Hz |

### 3.4.19. Object 5555h: Start in Operational

This manufacturer specific object allows the unit to start in Operational mode without requiring the presence of a CANopen ® Master on the network. It is intended to be used only when running the controller as a stand-alone module. This should always be set FALSE whenever it is connected to a standard master/slave network.

When set to TRUE, the unit will still power up in BOOT mode and send the pre-operation message. Five seconds later, the unit will automatically switch to OPERATIONAL mode, and start broadcasting the measured temperatures on the relevant TPDOs. The 5 second delay is to make sure that all data from all channels is read correctly and prevents sending erroneous data to the network after a power cycle.

**Object Description**

| Index | 5555h |
|---|---|
| Name | Start in Operational Mode |
| Object Type | VARIABLE |
| Data Type | BOOLEAN |

**Entry Description**

| Sub-Index | 0h |
|---|---|
| Access | RW |
| PDO Mapping | No |
| Value Range | 0 (FALSE) or 1 (TRUE) |
| Default Value | 0 [FALSE] |

## 4. USING RS-232 WITH TERA TERM

Additional information for diagnostics or testing is available through RS-232.

- Connect the DB-9 to a COM port on a PC or laptop. Use the following RS-232 connection.

| 8-Pin Connector | | DB-9 Female | |
|---|---|---|---|
| Pin # | Controller Function | Pin # | PC Function |
| 7 | RS-232 Transmit | 2 | RS-232 Receive |
| 8 | RS-232 Receive | 3 | RS-232 Transmit |
| 6 | GND Reference | 5 | RS-232 GND |

- Open Tera Term Pro, and set it up as shown in the steps below.
  (Free downloadable from http://logmett.com/index.php?/products/teraterm.html)

- Select **Serial** with the appropriate COM port for your PC or laptop
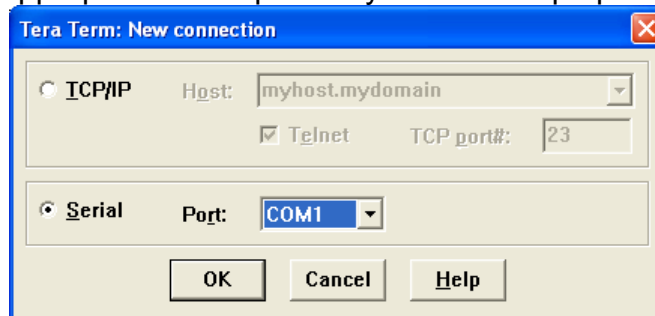
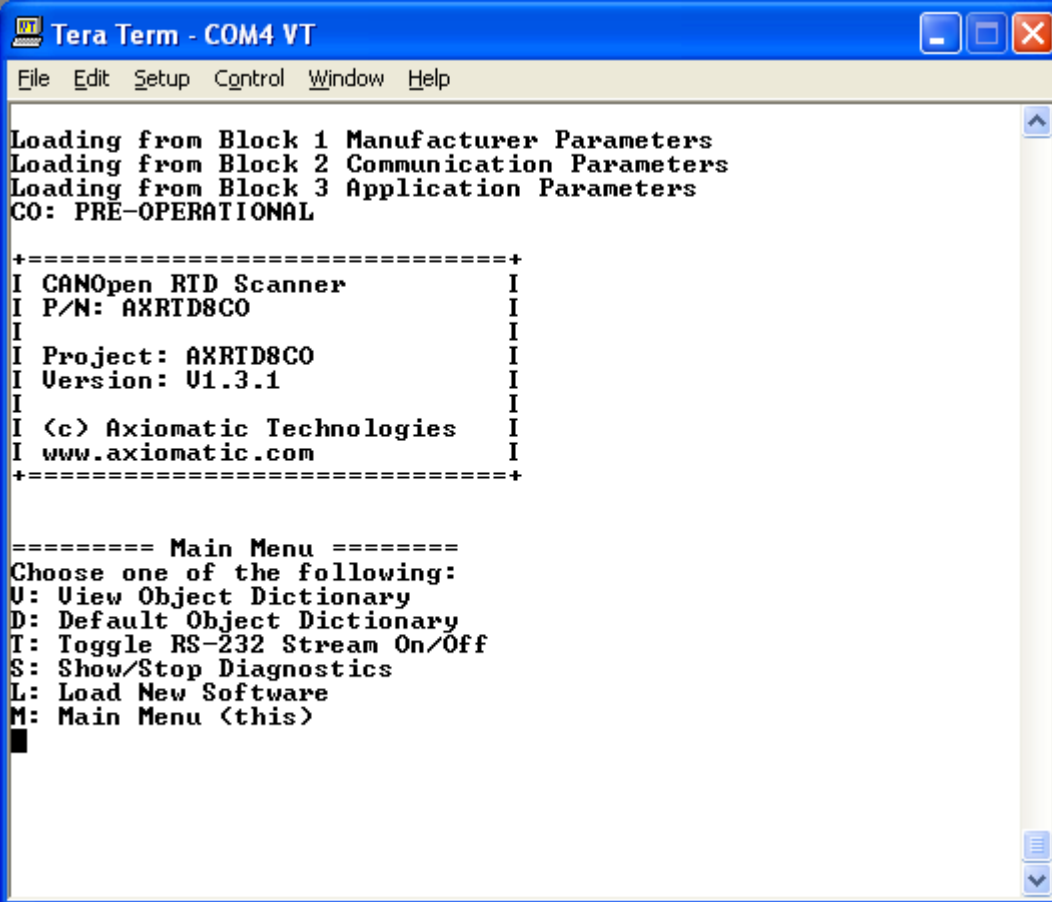- Go to **Setup/Serial Port** and change the settings to exactly as shown below (other than Port)
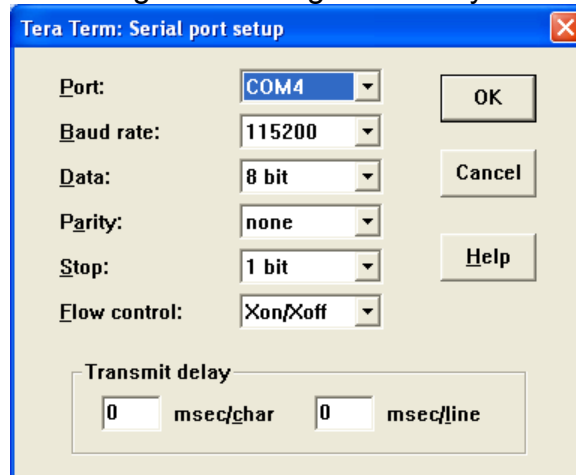
- Go to **Setup/Terminal** and verify that New-line Transmit and Receive are CR. The window size can be adjusted as desired by checking 'Term size = win size'
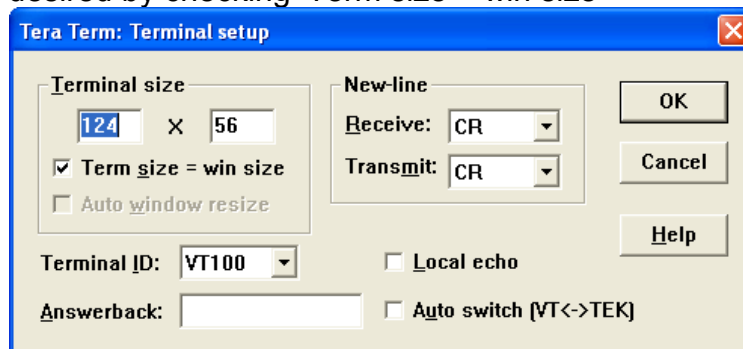
## 4.1. Main Menu Options

At power up, the Main Menu will be displayed, after the power up banner has been printed. If at any time you wish to see the menu again, simply hit 'm' or 'M' and it will be reprinted, along with the basic information about the CANopen ® network variables.
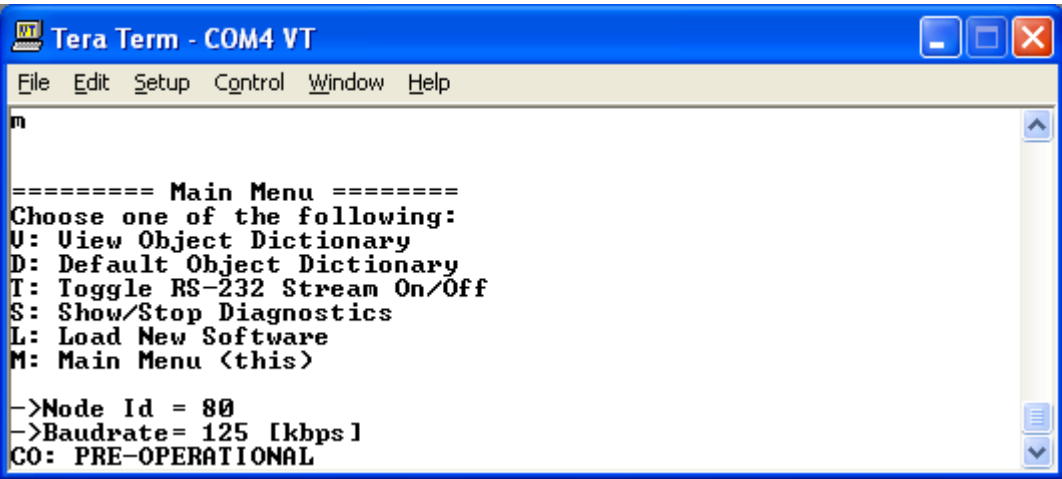
### 4.1.1. V – View Object Dictionary

Entering 'v' or 'V' while the node is in the pre-operational state starts the display of the active Object Dictionary for the AXRTD8CO. Since it is too large to print in one shot, it will show the PDO communication and mapping parameters, then display each object one by one. To view the next object, simply hit the 'Enter' key. If at any time you would like to exit the loop, simply hit 'x' or 'X' and normal operation will resume.

*WARNING: While in the view sub-menu, all other operations including temperature scanning and CAN networking are halted. Do not use this feature when regular operating conditions are required.*

```
Tera Term - COM4 VT

File  Edit  Setup  Control  Window  Help

v

RPDO Comm and Mapping Objects
=============================
0x1400   0xC0000250   255   0   0   0
0x1401   0xC0000350   255   0   0   0
0x1402   0xC0000450   255   0   0   0
0x1403   0xC0000550   255   0   0   0

0x1600
0x1601
0x1602
0x1603

TPDO Comm and Mapping Objects
=============================
0x1800   0x400001D0   254   0   0   1000
0x1801   0x400002D0   254   0   0   1000
0x1802   0x400003D0   254   0   0   1000
0x1803   0x400004D0   254   0   0   0

0x1A00   0x71300110   0x71300210   0x71300310   0x71300410
0x1A01   0x71300510   0x71300610   0x71300710   0x71300810
0x1A02   0x21300110   0x21300210   0x21300310
0x1A03   0x50000020   0x50010020

Object Dictionary
=============================
0x1000        0x10020194
0x1001        1
0x1002        0
0x1003    0   8
          1   0x0208F001
          2   0x0207F001
          3   0x0206F001
          4   0x0205F001
          5   0x0204F001
          6   0x0203F001
          7   0x0202F001
          8   0x0201F001
          9   0x00000000
          10  0x00000000
          11  0x00000000
          12  0x00000000
          13  0x00000000
          14  0x00000000
          15  0x00000000
          16  0x00000000
          17  0x00000000
0x100C        0
0x100D        0
0x1010    0   4
```
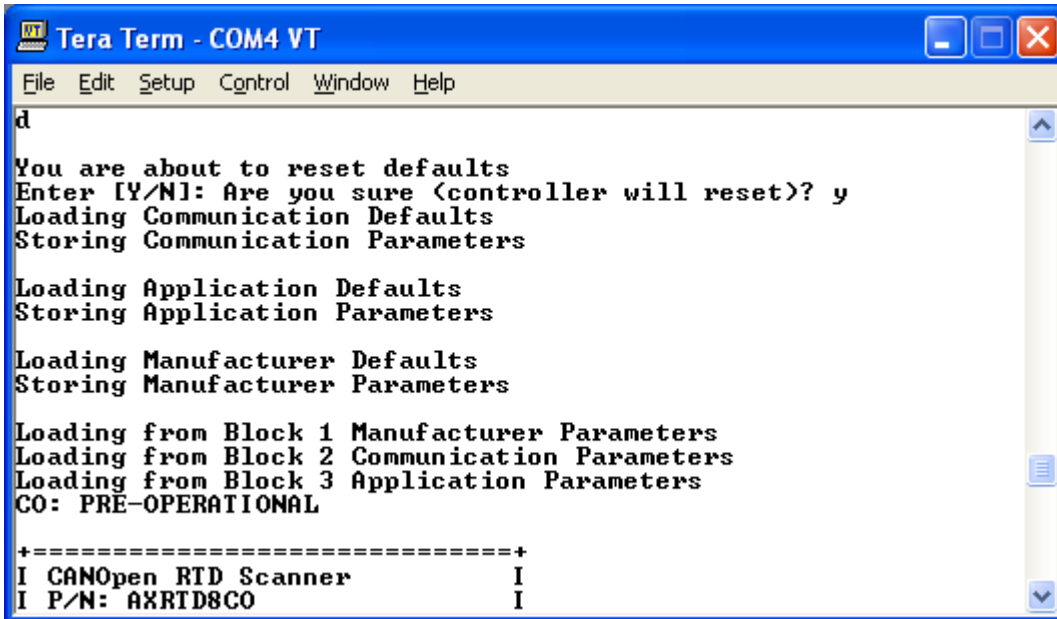
### 4.1.2. D – Default Object Dictionary

To reset the default Object Dictionary, enter 'd' or 'D'. At the prompt, enter 'Yes'. This function mirrors that of writing 'load' to Object 1011h, Restore Defaults. Consequently, it does not reset the Node-ID or Baudrate if they have been changed using LSS protocol. However, unlike the Restore Defaults object, these defaults will be immediately applied, as the controller automatically resets itself.



### 4.1.3. T – Toggle RS-232 Stream On/Off

In some applications, it may be desirable to be able to read or log data using RS-232. In these cases, this menu option enables the user to configure some basic variables associated with the RS-232 data stream. All configurations are automatically stored in non-volatile memory, and will still take effect after every power cycle. (i.e. once on, the data will always be sent until turned off).

The two variables that the user can change when selecting to toggle the data stream ON are:
a) Whether the data sent will be the field value (object $6100) or the Process Value (object $7130)
b) The repetition rate at which the data stream will be sent, in milliseconds.

The field width is fixed in all cases to 6 characters, and each temperature is delimited by a semi-colon. The order of the data is always sent as shown below.
RTD1;RTD2;…RTD7;RTD8

If a channel is not used (disabled by object $6112), the entry will read "Null ". Alternatively, if the sensor is open circuited, the entry will read "Open  ", and a short would read "Short".

An example of the RS-232 data stream that would be sent is shown on the next page.

```
t


Do you want to send the field values (DegC)? [Y/N]
No = Send Process Value (Objects $6126 and $6127): y

What repetition rate (in [ms]) will the stream be sent? 500


Field Value will be sent every 500 [ms]
RTD1;RTD2;...;RTD7;RTD8
Storing Factory Parameters


53.600;53.300;54.200;50.400;53.100;53.700;53.100;53.500
53.600;53.300;53.500;50.400;53.100;53.700;53.100;53.500
53.600;53.300;53.500;53.300;53.100;53.700;53.100;53.500
53.600;53.300;53.500;53.400;53.100;53.700;53.100;53.500
53.600;53.300;53.500;53.400;53.100;53.700;53.100;53.500
53.600;53.300;53.500;53.300;53.100;53.700;53.100;53.500
53.600;53.300;53.500;53.300;53.100;53.700;53.100;53.600
53.600;53.300;53.500;53.400;53.100;53.700;53.100;53.600
```
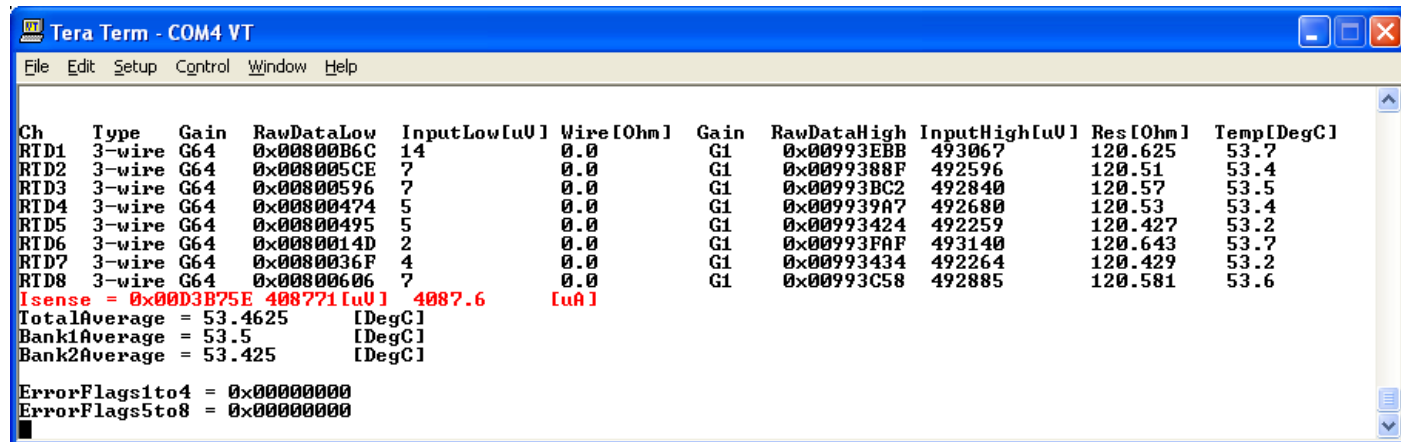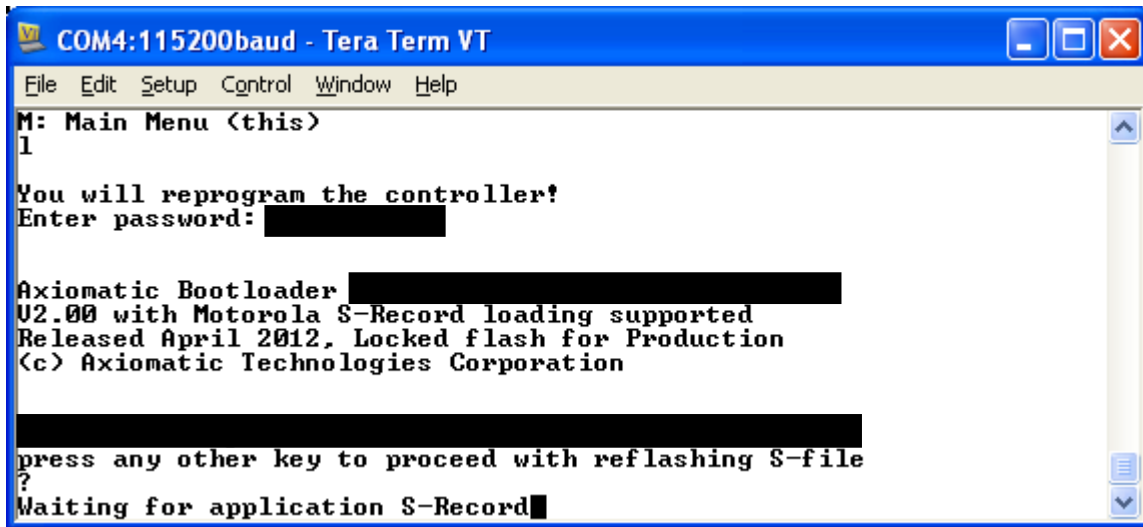
### 4.1.4. S – Show/Stop Diagnostics

Another diagnostic option is available using the 's' or 'S' option. Once started, the diagnostic screen will be refreshed every 1 second until stopped. Please note, this option is only for manual diagnostics, and is not saved in non-volatile memory (i.e. the diagnostic screen is never displayed after a power cycle without the 's' entry)



```
Ch     Type    Gain  RawDataLow   InputLow[uV] Wire[Ohm]  Gain  RawDataHigh  InputHigh[uV] Res[Ohm]  Temp[DegC]
RTD1   3-wire  G64   0x00800B6C   14           0.0        G1    0x009993EBB  493067        120.625   53.7
RTD2   3-wire  G64   0x008005CE   7            0.0        G1    0x0099388F   492596        120.51    53.4
RTD3   3-wire  G64   0x00800596   7            0.0        G1    0x00993BC2   492840        120.57    53.5
RTD4   3-wire  G64   0x00800474   5            0.0        G1    0x009939A7   492680        120.53    53.4
RTD5   3-wire  G64   0x00800495   5            0.0        G1    0x00993424   492259        120.427   53.2
RTD6   3-wire  G64   0x0080014D   2            0.0        G1    0x00993FAF   493140        120.643   53.7
RTD7   3-wire  G64   0x0080036F   4            0.0        G1    0x00993434   492264        120.429   53.2
RTD8   3-wire  G64   0x00800606   7            0.0        G1    0x00993C58   492885        120.581   53.6
Isense = 0x00D3B75E 408771[uV]   4087.6       [uA]
TotalAverage = 53.4625        [DegC]
Bank1Average = 53.5           [DegC]
Bank2Average = 53.425         [DegC]

ErrorFlags1to4 = 0x00000000
ErrorFlags5to8 = 0x00000000
```
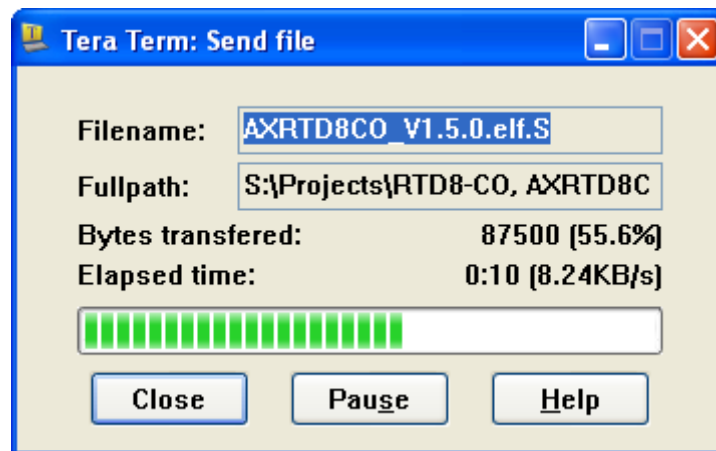
### 4.1.5. L – Load New Software

Should a software upgrade of the application software be required, the units can be reprogrammed by selecting this option. If Axiomatic has sent new software, select 'l' or 'L', and at the prompt enter the password that was provided by your Axiomatic contact.

Once the correct password has been entered (case-sensitive), the controller will automatically reset itself the bootloader header will be printed. At the prompt, hit any key (i.e. Enter) and wait for the message "**Waiting for application S-Record**" to be displayed. Go to **File/Send File** and send the AXRTD8CO_Vx.y.z.elf.S file sent by Axiomatic.

```
 COM4:115200baud - Tera Term VT
File  Edit  Setup  Control  Window  Help
M: Main Menu (this)
l

You will reprogram the controller!
Enter password: ████████████

Axiomatic Bootloader ████████████████████████
V2.00 with Motorola S-Record loading supported
Released April 2012, Locked flash for Production
(c) Axiomatic Technologies Corporation

████████████████████████████████████████

press any other key to proceed with reflashing S-file
?
Waiting for application S-Record█
```

As the file uploads, a pop-up progress message will be shown, and the controller will print dots on the display. After the file has finished loading, the message "S-File has been loaded successfully" will be displayed, and the normal power-up messages will be printed. Verify the version number in the power-up banner matches that of the latest software.



```
 Tera Term: Send file
Filename:        AXRTD8CO_V1.5.0.elf.S
Fullpath:        S:\Projects\RTD8-CO, AXRTD8C
Bytes transfered:              87500 (55.6%)
Elapsed time:                  0:10 (8.24KB/s)
[████████████████████                        ]
      Close          Pause          Help
```

# APPENDIX A – Technical Specifications

## Inputs

| | |
|---|---|
| Power Supply Input | 12Vdc or 24Vdc nominal (9…32 VDC power supply range) |
| Supply Current | 170 mA at 12 V Typical, 90 mA at 24 V Typical<br>Inrush does not exceed 800 mA. |
| Protection | Reverse polarity protection is provided.<br>Power supply input section protects against transient surges and short circuits and is isolated from RTD inputs |
| RTD Types | Up to 8 channels, independently configurable for 2, 3 or 4 wire RTDs. Each channel independently supports specific sensors IEC 0.00385, JIS 0.003916, US 0.003902, Legacy 0.003920, SAMA 0.003923.<br>A user defined coefficient would enable custom Callendar-Van Dusen constants to be set for sensors not listed above. |
| RTD Inputs | The device accepts inputs within the following range of 10 - 350 Ohms.<br>"Short Circuit" condition is triggered when the sensor resistance is below 5 Ohm.<br>"Open Circuit" condition is reported when the sensor resistance exceeds approximately 600 Ohm.<br><br>Accuracy: +/- 1°C typical at ambient temperature<br>Resolution: 0.1°C<br>Isolation voltage is 1500 Vac (rms) or 2550V for 1 sec. |
| Scan Rate | 100ms per channel, total sweep time maximum 900 ms |
| Common Mode Readings | Input range +/- 4V maximum<br>Rejection is 100db at 5Vp-p (50-60Hz) |
| Thermal Drift | Overall drift with temperature is 15mOhm/°C (maximum) |
| Isolation | Digital isolation is 500VDC from input to ground. Three way isolation is provided for the CAN line, inputs and power supply |
| Averaging | Available on Bank 1 (RTD1-RTD4), Bank 2 (RTD5-RTD8) and Total (All) |
| Protection | Open or short circuit detection, Frozen data detection<br>Over or under temperature detection, High temperature shutdown detection |

## Communication

| | |
|---|---|
| CAN | 1 CAN 2.0Bport, protocol CiA CANopen ®<br>By default, the RTD Scanner transmits the process value (object $7130) according to the device profile in CiA Standard DS-404 |
| RS-232 | 1 RS-232 port available, ASCII Text Format, 115200 Baud Rate<br>Data – 8 bit, Parity – None, Stop – 1 bit. Flow Control – Xon/Xoff.<br>Short circuit protection to ground. |

## General Specifications

| | |
|---|---|
| Microprocessor | 16-bit, 512 KByte flash program memory |
| Control Logic | User programmable functionality using SDO object access, per CiA DS-301 |
| User Interface | .EDS provided to interface to standard CANopen ® tools<br>Go to the Log-in tab on www.axiomatic.com. Request to password from sales@axiomatic.com. |
| UL and cUL Compliance | UL508 (April 2010) (FTPM2) – Controls for Stationary Engine Driven Assemblies<br>cUL C22.2 No. 14-10 (2010) |
| CE Compliance | 2004/108/EC (EMC Directive)<br>2011/65/EU (RoHS Directive) |
| Vibration | 11.48 G for a device rigidly mounted to a generator housing<br>*The marine type approval process tested to 4,0 G per IEC 60068-2-6, Test Fc.* |
| Marine Type Approval | Lloyd's Register, DNV, ABS, RINA, GL, BV, CCS, IRS, RS<br>*The AXRTD8CO meets the environmental, EMC and vibration requirements of generator set applications in marine installations.* |
| Operating Conditions | -40 to 85 °C (-40 to 185 °F) |
| Storage | -50 to 120 °C (-58 to 248 °F) |
| Humidity | Protected against 95% humidity non-condensing, 30°C to 60°C. |
| Enclosure | Rugged aluminum housing, stainless steel end plates, neoprene gaskets<br>145.30 x 149.00 x 73.00 mm (5.72 x 5.86 x 2.87") L x W x H<br>Connectors, Deutsch IPD P/N: 1 8-pin DT13-08PA, 1 40-pin DRC13-40PA |
| Protection | IP67 rating, Pollution Degree 3 per UL508<br>*The marine type approval process tested to IP56.* |
| Weight | 2.2 lbs. (1.00 kg) |