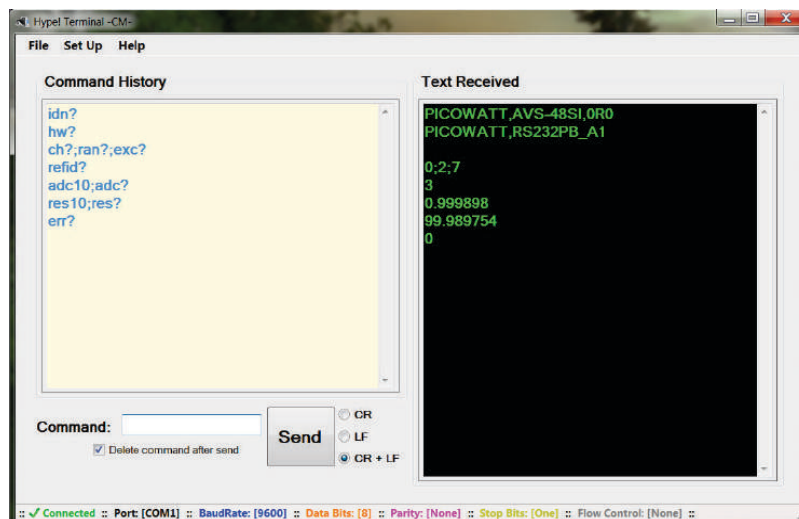
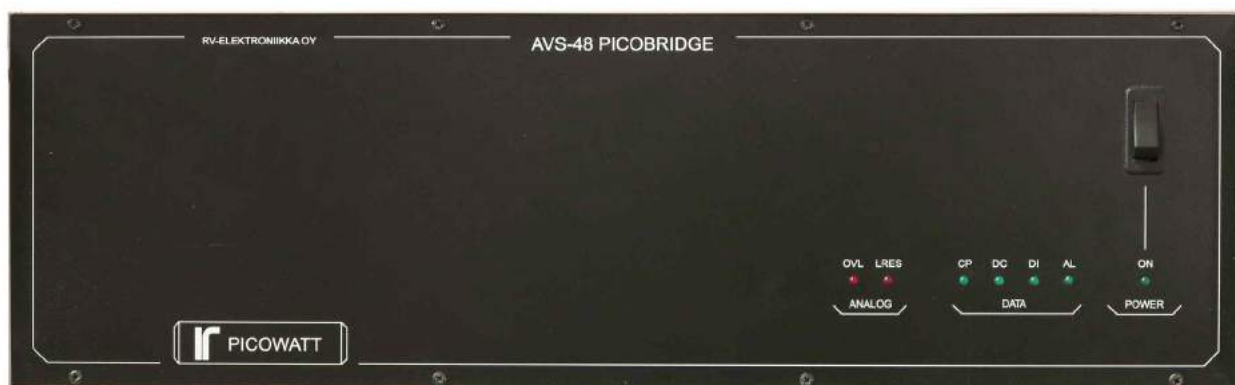


# AVS-48SI PICOBRIDGE®

High-performance  
Cryogenic AC Resistance Bridge and Temperature Controller

## User Guide for Operation via USB/RS232 Serial Interface or LabView



**PICOWATT**

Veromiehentie 14  
FI-01510 VANTAA, Finland  
phone 358 50 3375192  
Internet: [www.picowatt.fi](http://www.picowatt.fi)  
e-mail: [reijo.voutilainen@picowatt.fi](mailto:reijo.voutilainen@picowatt.fi)



## **CONTENTS**

<b>WARRANTY</b>	<b>3</b>
<b>SPECIFICATIONS</b>	<b>4</b>
<b>STARTING TO USE THE AVS-48SI</b>	<b>7</b>
<b>CONNECTIONS</b>	
W version with wire connection	7
F version with optical fibre link	8
<b>SENSOR WIRING</b>	<b>9</b>
<b>PHYSICAL FRONT PANEL</b>	<b>9</b>
<b>PHYSICAL REAR PANEL</b>	<b>10</b>

## **PART I - SERIAL OPERATION**

<b>AVS-48SI SERIAL COMMAND SUMMARY</b>	<b>15</b>
<b>INTRODUCTION</b>	<b>18</b>
<b>SHORT BRIDGE TUTORIAL</b>	<b>19</b>
<b>SERIAL COMMANDS AND QUERIES</b>	<b>20</b>
<b>CALIBRATION</b>	<b>29</b>
<b>OTHER COMMANDS</b>	<b>33</b>
<b>CABLE SPECIFICATIONS</b>	<b>35</b>
<b>RE-PROGRAMMING ARDUINO</b>	<b>36</b>
<b>TROUBLE SHOOTING</b>	<b>37</b>

## **PART II - LABVIEW OPERATION**

<b>INTRODUCTION</b>	<b>41</b>
<b>PROGRAM INSTALLATION</b>	<b>42</b>
<b>SHORT TUTORIAL</b>	<b>43</b>
<b>THE AVS48SI.VI PROGRAM</b>	
Front tab	45
Settings tab	53
Service tab	59
<b>CALIBRATION</b>	<b>61</b>
<b>TEMPERATURE READOUT</b>	
R/T conversion files	65
Assigning to sensors	66
<b>OPENING THE INSTRUMENT</b>	<b>68</b>
<b>INDEX</b>	<b>69</b>



## WARRANTY

**Picowatt** warrants the **AVS-48SI hardware** to be free from defects in materials and workmanship. Our liability under this warranty is limited to repairing or replacing any instrument or part thereof which, within three (3) years after the shipment to the original purchaser, proves defective. This warranty is void if the instrument has not been used according to the instruction manual, or if it has been used under exceptional environmental conditions (see below).

In need of warranty repair, the instrument must be returned to **Picowatt**, prepaid, and with a detailed description of the fault or malfunction following the instrument.

The instrument must always be returned complete with the preamplifier, preamplifier cable and with the microprocessor unit. This is because then the calibration and performance of the AVS-48SI can be checked and guaranteed.

The name, address and e-mail address of a person who is able to give supplementary information should be included whenever possible. If the repair was covered by warranty, **Picowatt** will return the instrument on our cost using an economical shipping method.

If no fault is found, or if there is a strong indication that the warranty is void, the purchaser is charged for the return freight and costs in addition to the repair. It is recommended that **Picowatt** be contacted prior to shipment. We can possibly give instructions for additional tests or simple component replacements so that unnecessary shipments may be avoided.

LabView is a trade mark of National Instruments, USA.

Picobridge is a registered trade mark of RV-Elektroniikka Oy

**Caution:** The **AVS-48SI Picobridge** is a delicate laboratory instrument. It has been designed only for the purpose of measuring resistances of passive resistive networks. Using this instrument for any other purpose will void the warranty and may cause permanent damage to the instrument. By no means should the **AVS-48SI** be used to measure internal resistances of power supplies, batteries, capacitors or any other devices which can supply energy to the bridge input. The **AVS-48SI** has been designed to operate in a laboratory environment, which means normal living room atmosphere, temperature, humidity and purity of air. The unit does not tolerate continuous vibration or hard shocks.

RV-Elektroniikka Oy Picowatt  
Veromiehentie 14  
FI-01510 VANTAA  
FINLAND  
telephone +358 50 337 5192  
email: reijo.voutilainen@picowatt.fi

## SPECIFICATIONS

(Specifications marked with \* are applicable only when the optional temperature controller card is installed).

RANGE 0-3Ω...0-30MΩ 8 ranges

EXCITATION 3μV...10mV RMS  
8 excitation ranges, square wave current,  
12.5/13.64/15Hz. Excitation current = excita-  
tion voltage divided by 1/3 of the range.

CHANNELS 4-wire connected channels #1-7.  
2-wire mode for any sensor is selectable by  
software. Difference between 2- and 4-wire  
readings can be used for diagnostics. Channel  
#0 is reserved for measuring internal reference  
resistors.

INPUT CONNECTOR DC37S (female).  
Mating (user's) connector type DC37P. All 7  
channels are pin-to-pin compatible with the  
wiring of the AVS-47x sensors 0-6.

SENSOR GROUNDING Floating inside the  
cryostat (recommended default) or I- lead  
grounded to the cryostat.

MAXIMUM CURRENT LEAD RESISTANCE  
3Ω range and 10mV excitation: 2\*10Ω  
3Ω range and 1mV excitation: 2\*100Ω  
30Ω range and 10mV excitation: 2\*100Ω  
Higher ranges and any excitation: 2\*1kΩ

INPUT CAPACITANCE Max. 1nF in each or  
any lead. Input capacitance is cancelled by a  
shape-based capacitance compensation circuit.

INPUT NOISE VOLTAGE Typically 5nV/√Hz  
at 13.64Hz.

SPEED Time needed for the signal to reach a  
good final level when step is from 0 to 2/3 of  
the range:  
- About 6 seconds for 10mV excitation  
- About 12 seconds for 3μV excitation

ANALOG FILTER Residues of the carrier are  
removed from the analog output by a 3rd or-  
der Bessel filter with a 3dB roll-off frequency

of about 1.2Hz. Filtering is not applied to the  
control error  $V_{Cerr}$  for avoiding delay in the  
control loop.

## Signal Inputs and Outputs

(rear panel BNC connectors)

ANALOG OUTPUT Calibrated voltage output  
0...+3V from the analog circuitry.

$$V_O = (R_{sensor} / \text{Range}) * 3V$$

$$(R_{OUT} = 100\Omega, \text{Range} = 3\Omega, 30\Omega \text{ ..etc.})$$

Analog output can also be set to be the control  
error  $V_{Cerr}$ , filtered by the analog filter.

CONTROL SET POINT DAC voltage output cali-  
brated to ±100μV. Available also when the TC  
option is not installed.

$$V_{Setp} = 0.005...+2.99V \text{ . Step } 50\mu V.$$

CONTROL ERROR OUTPUT Voltage output  
 $V_{Cerr} = -3...+3V$  from the analog circuitry.

$$V_{Cerr} = (V_O - V_{setp}) \text{ or } (V_{setp} - V_O)$$

Not filtered in order to minimize delays in  
the temperature control loop. Control polarity  
can be changed from software. Available also  
when temperature controller is not installed.

AC MONITOR Amplified AC signal from the  
sensor. For trouble-shooting and for checking  
low-frequency interference and noise levels  
with an oscilloscope. BNC with grounded  
jacket.

\*HEATER OUTPUT Temperature controller's  
heater output is a negative current source.  
Positive terminal is grounded to instrument  
case and mains earth. Maximum compli-  
ance 12-13 Volts and maximum current about  
250mA. Connection to heater using one or  
two coaxial cables.

\*EXTERNAL HEATER DRIVE The con-  
troller's heater can be driven by an external  
voltage 0...+3V. Selection between internal

drive (analog PID circuitry) and external drive is made by a short-circuit piece.

**EXTERNAL CLOCK I/O** This I/O can be used to operate the bridge at a non-standard frequency. The external signal must be 0...+5V square wave signal

$$F_{PSD} = \frac{F_{EXT}}{600Hz} * 13.64Hz$$

(Instead of 13.64, use the currently selected excitation frequency 12.5/13.64/15.0Hz)

If not loaded, this I/O outputs a frequency of 600Hz (0...+5V square).

**USER DAC** DAC voltage output from 0.005V to 2.99V calibrated to  $\pm 100\mu V$ . Rear panel BNC connector. Settling resolution 50 $\mu V$ .

### Analog PID Temperature Controller (option)

\*HEATER RESISTANCE No limits, but 100 $\Omega$  is the recommended default. Maximum power is obtained when  $R_H = 50\Omega$ .

\*HEATER POWER 18 logarithmically spaced power ranges from 1 $\mu W$  to 1.8W into a 100 $\Omega$  heater and from 0.5 $\mu W$  to about 3W into a 50 $\Omega$  heater. Ratio between power of adjacent ranges is  $\sqrt[3]{10}$

#### \*PID PARAMETERS

14 logarithmically spaced proportional gains  $G_P$  from 1 to 10000

12 logarithmically spaced integrator gains  $G_I$

11 logarithmically spaced derivator gains  $G_D$  (modified response)

#### \*TRANSFER FUNCTION

$$I_{heater} = \sqrt{G_P * \left( V_{Cerr} + G_I \int V_{Cerr} dt + G_D * dV_{Cerr} / dt \right) / 3R_{sense}}$$

$$P_{heater} = I_{heater}^2 * R_{heater}$$

The heater power - not current - is a linear function of the control error. This can improve stability when the power level varies.

### Other

**POWER INPUT** The AVS-48SI Picobridge features three options for supplying power:

- 1) 90-250V 50-60Hz 0.1A mains voltage (safety earth lead is mandatory)
- 2) 18V 50-60Hz 1A safe low voltage (allows your own power wiring and filtering).
- 3) 12V 1.5A DC battery input e.g. for hunting interference due to ground loops or mains power line.

### RESOLUTION AND CALIBRATION ACCURACY

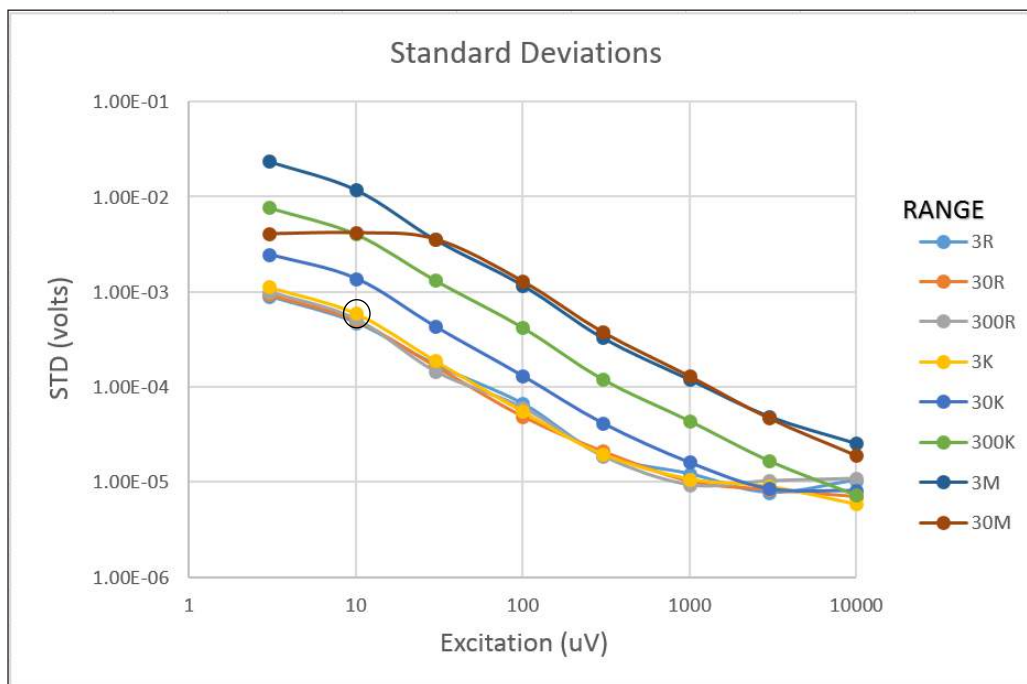
The RMS resolution is here defined as the standard deviation of non-filtered A/D conversions. *Digital filtering reduces these figures approximately by a factor of  $\sqrt{N}$ , where  $N$  is the filter length in samples.*

The standard deviations are measured using room temperature sensors, which means that resistor noise is dominating when resistance is high.

Peak-to-peak resolution is about 5 times worse than the STD resolution.

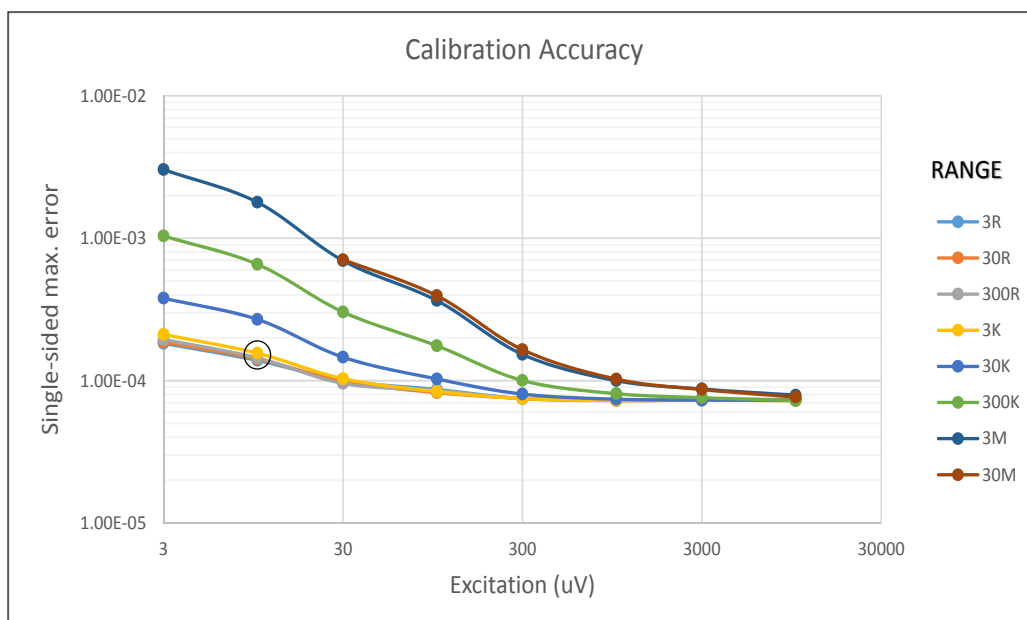
One volt corresponds to 1/3 of the measuring range (e.g. 1 volt on 300k $\Omega$  range corresponds to 100k $\Omega$ ).

When excitation is high and S/N ratio is good, accuracy of the calibration is determined by the ohmmeter used for measuring the reference resistors. When excitation is low and S/N ratio is poor, calibration accuracy becomes mainly dependent on the peak-to-peak noise of the averaged signal. On the lowest 3 $\mu V$  excitation range, as many as 800 readings are averaged by the calibration program.



Single-conversion STD resolutions. Room-temperature decade resistors were measured on their respective ranges and all excitations. The low-excitation behaviour of the 30MΩ range is due to reduced gain.

Read the graph as follows: 1kΩ sensor is measured on 3kΩ range and 10μV excitation. The graph gives a single-conversion STD of about  $6E-4V \cdot 3k\Omega / 3V = 0.6\Omega$ . An average of 100 conversions (about 20 seconds) reduces noise to about 60mΩ.



At high excitations, calibration accuracy is worse than the resolution because of the performance limits of the calibrating ohmmeter. Accuracy at low excitations is better than resolution, because very long averages are used for suppressing noise.

Values of the above example give accuracy of  $\pm 1.5E-4V \cdot 3k\Omega / 3V = \pm 0.15\Omega$ .



## STARTING TO USE THE AVS-48SI

### Unpack the shipment

The shipment should contain the following (if no options):

1. AVS-48SI Main Unit (19")
2. AVS-48SI Preamplifier Unit
3. AVS48SI-Serial/USB-W or -F CPU unit (second alternative, if optical link was ordered)
4. 5-meter Preamplifier Cable 25-way (PR25P25P25W5M)
5. 5-meter cable AVS-48SI <=> CPU unit (PB9P15P6W5M)
6. 1.5-meter cable CPU unit <=> Computer (RS9P9S7W1.5M)
7. Power mains cord
8. 3-way male DIN plug (PREH 71430-030) for the alternative 18V AC power supply
9. 4-way male DIN plug (PREH 71430-040) for the alternative 12V DC power supply
10. 1A slow blow spare fuse (rear panel) 5x20mm
11. Pre-wired Molex connector (22-01-2045) for measuring reference resistors
12. Two male DC37P plugs with metal shells for sensor connections
13. 37-way short-circuit plug for protecting the pre-amplifier input (is plugged into the input).
14. +12V wall adapter for powering the CPU unit

### Only if included in the order:

15. Temperature Controller Option, installed
16. National Instrument's USB-232 converter with its installation software (NI 778472-01)
17. AVS48SI-Picolink optical Receiver/transmitter unit if optical link was ordered.
18. 5 or 10-meter Optical cable
19. 1.5 meter cable between AVS-48SI Main Unit and item 17. This replaces item 5, which is omitted when AVS48SI-Picolink is ordered (PB9P15P6W1.5M).

### Install the USB-232 Converter

Before obtaining a USB-232 converter, verify its compatibility with your platform. Our CPU box with Arduino Mega2560 uses RS232 in its simplest and most common configuration: only RXD, TXD and Ground wires, no handshaking, 9600 bauds, 8 data bits, no parity and one stop bit.

Follow the manufacturer's instructions on how to install their converter and its driver. Use Windows Control Panel's Device Manager or National Instruments' Measurement and Automation Explorer for finding the serial port name that you will use, either directly or via the converter.

## CONNECTING THE AVS-48SI

### WIRE CONNECTION, NO OPTICAL LINK

Connect the preamplifier unit to the main unit using the 5 meter long 25-way cable with DB25P connectors at both ends (cable type PR25P25P25W5M). Tighten the connectors properly, a good ground connection is important.

Connect the AVS48-USB/Serial-W CPU unit to the AVS-48SI main unit by the 5 meter long cable that has 9-pin DE9P and 15-pin DA15P connectors (cable type PB9P15P6W5M).

Connect the CPU unit to your computer or to the USB-232 converter by the 1.5 meter long cable that has male and female 9-pin DE9 connectors (cable type RS9P9S7W1.5M ).

Verify that there is the supplied 37-way short-circuit connector plugged into the AVS-48SI preamplifier input.

Connect the AVS-48SI main unit to mains power. The instrument has a universal power supply for 90-250V. Switch on the main unit. Green power-on light will turn on.

Connect the CPU unit to mains using the 12V mains adapter. Some activity on the AVS front panel LEDs shows that the CPU has booted.

Check the voltage on the rear panel ANALOG OUT connector. It should be near to 1V.

## OPTICAL FIBRE CONNECTION

(If the AVS-48SI was shipped with the optional AVS48SI-Picolink).

Connect the preamplifier unit to the main unit using the 5 meter long 25-way cable with DB25P connectors at both ends (cable type PR25P25P25W5M). Tighten the connectors properly, a good ground connection is important.

Connect the AVS48SI-Picolink Box to the AVS-48SI main unit using the 1.5 meter long cable with 9-way DE9P and 15-way DE15P connectors (cable type PB9P15P6W1.5M). This box has only optical isolators, no digital intelligence. Therefore it can be placed inside the Faraday shield. This box gets its power from the bridge.

Connect the Picolink Box to the AVS48-USB/Serial-F CPU unit by using the 5- or 10 meter long optical fibre cable. The all-plastic optical cable consists of four fibres, which have been marked with colors black, red, yellow and blue. The optical transmitters (white) and receivers (black) are marked with numbers 1..4. Connect the fibres to corresponding numbers on the boxes. Recommended use of the colors: 1=black, 2=red, 3=yellow and 4=blue.

Push the ends of the fibres into the receivers/transmitters as deep as they go and tighten the nuts. Be extremely careful not to tighten them too much: these components are mechanically weak and they break easily. Use only fingers. However, the fibres must stay fixed, and they do, if you avoid tension and sharp bending of the cable.

The fibre ends can be renewed by cutting a millimeter or two using sharp full-flash end cutters, if they do not seem to be in good condition after years.

Connect the CPU unit to your computer or to the USB-232 converter by the 1.5 meter long cable that has male and female 9-pin DE9 connectors (cable type RS9P9S7W1.5M).

Verify that there is the supplied 37-way short-circuit connector plugged into the AVS-48SI preamplifier input.

Connect the AVS-48SI main unit to mains power. The instrument has a universal power supply for 90-250V. Switch on the main unit. Green light will turn on.

Connect the CPU unit to mains using the 12V mains adapter. Some activity on the AVS48SI front panel LEDs shows that the CPU has booted.

Check the voltage on the rear panel ANALOG OUT connector. It should be near to 1V.



The optional optical link for the AVS-48SI consists of two boxes connected by the fibre optic cable. Box on the left is connected to the AVS-48SI and contains only the opto card. It gets power from the bridge. The rightmost box contains both the opto card and an Arduino Mega2560 processor unit (CPU), which contains the firmware of the AVS-48SI. It is powered by a wall adapter.



## SENSOR WIRING

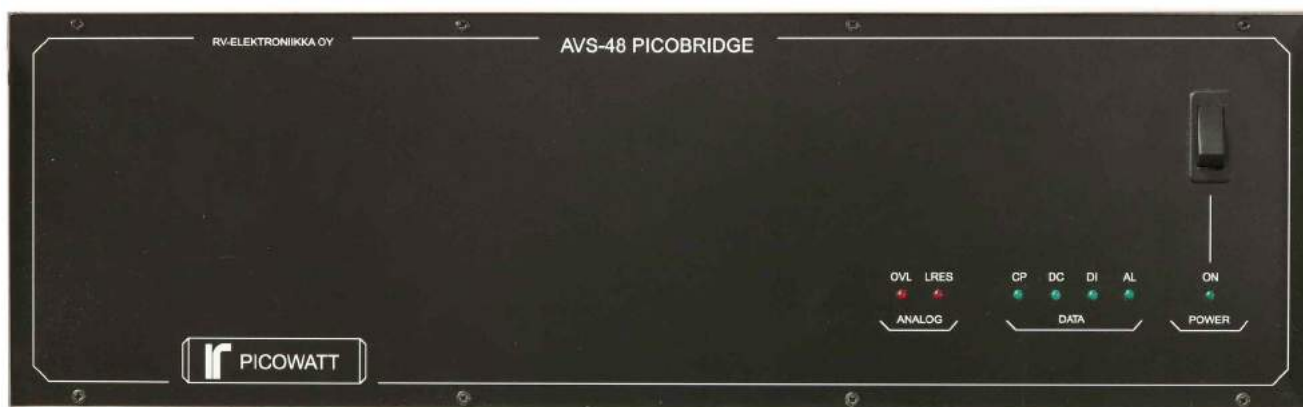
Wiring of the AVS-48SI sensors is pin-to-pin compatible with the 7 first sensors of the AVS-47x series of bridges. The ground pins are for cable shields

Pin name	CH0	CH1	CH2	CH3	CH4	CH5	CH6
I+	1	3	5	7	9	11	13
V+	20	22	24	26	28	30	32
V-	2	4	6	8	10	12	14
I-	21	23	25	27	29	31	33

GROUND: 15, 34, 16, 35, 17, 36, 18, 37, 19 and the shielding braid. (GND is connected to the enclosures of the preamplifier and main units). The braid is an important part of the ground-connection, so please tighten the connectors well.



Front panel of the AVS-48SI Picobridge  
Preamplifier Unit



## AVS-48SI PHYSICAL FRONT PANEL

The physical front panel of the AVS-48SI Picobridge contains only the power switch and seven indicator lights.

**Power switch and Power indicator.** This switch does not break the mains power but an internal 15V voltage. While it breaks all power from the instrument's circuitry, mains power continues to flow into the main AC/DC converter. If you have to open the cover of the instrument, verify that the mains power cord has been pulled off. A red light on the power supply board warns, if mains power is connected. The green power indicator on the front panel lights when low DC voltages are connected to AVS-48SI circuitry.

Power can be switched separately off from the bridge, or from the microprocessor box, or both. In any case, you need to re-boot the CPU in order

to restore correct operation. Re-starting brings the bridge in the reset state. A completely new session has to be started.

**Analog signal error lights.** The two red lights show the most common *analog* errors.

- **OVL light.** Analog overload occurs if the signal amplitude before phase-sensitive detection exceeds  $\pm 2$  Volts. This is usually due to an external interference (poor shielding of sensor leads, high resistance and low excitation together with mechanical vibration, or a severe ground current in the sensor leads).
- **LRES light.** Although a combination of excessive lead resistance measured on a low range and high excitation (refer to specifications) can greatly reduce the measurement accuracy, a warning about high lead resistance is in practice caused only by a broken current path to the sensor. Check the lead

resistances by measuring a sensor first in two-wire mode and then in 4-wire mode. The difference is the total resistance of the current leads.

**Picobus status lights.** The following four lights show the line states of the low-noise synchronous, serial Picobus interface between the external microprocessor box and the bridge.

- CP light. Clock Pulses from the box to the bridge.

- DC light. Synchronous Data from the box to the bridge.
- DI light. Synchronous Data from the bridge (Instrument) to the box.
- AL light. ALarm tells about an analog error (OVL and/or LRES) to the program.

The CP, DC and DI lights blink when the AVS-48SI exchanges data with the microprocessor box.



Physical rear panel as seen from the back side. Inserting cables etc., when the AVS-48SI is mounted in a rack, may be easier if one “looks the panel from the front side”.

## AVS-48SI PHYSICAL REAR PANEL

**Mains input.** The mains input voltage range is 90-250V 50-60Hz. The power cord must contain a ground lead connected to the safety ground, it is forbidden to cut it. If you need to isolate the AVS-48SI from mains ground, use an approved 1:1 isolation transformer. The AVS-48SI offers also some low-voltage power inputs, which are described below.

**Fuse.** The primary fuse size is 1A slow blow, 5x20mm

**18V AC input.** The bridge can be powered by a 50-50Hz AC voltage of about 18V rms. Current drain is less than 1A. This low-voltage input does not require a ground lead. The voltage is connected between pins 1 and 3 of the 3-way DIN socket. Mating plug for the wires is 71430-030/0800,

manufactured by Prehkeytec GmbH.

**12V DC input.** The third alternative to power the bridge is a 12V DC source - a battery or a laboratory power supply, for example. Current drain is as much as 1.5A. Negative terminal of the battery is connected to pins 1&2 and positive terminal to pins 3&4 or the 4-way DIN socket. Mating plug is 71430-040/0800, manufactured by Prehkeytec GmbH. A series diode protects the input from reverse polarity.

The mains input can remain connected. If the mains voltage is within specifications, the DC source will act as a backup source with no drain. There is no charging circuitry for the battery, however.

**Picobus Serial Input.** The AVS-48SI is connected to the microprocessor box via a synchronous, serial **Picobus interface**. This is a proprietary standard. It uses the four signal lines described above. If you



The AVS-48SI physical rear panel. This photo has been flipped horizontally and texts have flipped once again in order to make them more legible. This view can be useful when plugging cables when the bridge is fixed in a rack.

need to remove the 9-pin DE9P connector plug from the cable, e.g. for taking the cable via a small hole, refer to the cable specifications when reconnecting wires. Cable type: PB9P15P6W5M.

**Heater out.** The heater output is a negative current source. Maximum current is -250mA on the highest power range and the open-circuit voltage cannot exceed about -13V. The heater, which must be isolated from ground, can be connected to the AVS-48SI in two ways.

- 1) Use one coaxial cable connected to the grounded BNC terminal on the rear panel. The shielding braid is in contact with the AVS-48SI enclosure. This simple solution requires, that the connector in the cryostat should be isolated from ground, and also the heater must be floating. Then there is no path for a low-frequency ground current. A problem may be, that the cable can take high frequency interference into the cryostat. It may be possible to reduce this danger by connecting a capacitor from the braid to the cryostat.
- 2) Use two coaxial cables connected to the left (negative) and right (ground) terminals. In this solution, both shielding braids are isolated from the AVS-48SI enclosure but they must be connected to the body of the cryostat. The center wires of the cables carry the heating current. As the braids are now connected to the cryostat, they provide a better shield against high frequencies than solution #1. The price is that two connectors are needed.

**Analog out.** ANALOG OUT is the PSD integrator's voltage from the self-balancing loop. The residues of the 12.5/13.64/15.0 Hz square-wave carrier are suppressed by an analog 3<sup>RD</sup> order Bessel filter having a 3dB roll-off frequency of about 1.2Hz. The analog output is stepless and it provides a resolution that is limited only by noise, not by the digitizing step. Analog output extends up to 3 Volts, and with possibly reduced accuracy to some hundred millivolts above.

The ANALOG OUT signal can be switched to show the difference (analog out - set point voltage). This deviation signal is also filtered by the analog filter.

**Control Error Out.** The CONTROL ERROR OUT is also formed by subtracting the Set Point DAC's output voltage from the Analog Output voltage. Instead of a normal differential error amplifier, the Linear Technology's LT1043 building block creates the difference with practically no offset or gain error.

CONTROL ERROR goes from -3V to +3V. If temperature control is enabled, this stepless voltage serves as input to the controller. In order to prevent unnecessary phase-shift in the closed control loop, the CONTROL ERROR is not filtered as is the ANALOG OUTPUT.

**Control Set Point.** If the user has an external controller and wants to use it instead of the AVS-48SI's controller option, there are two possibilities that depend on the controller's structure: either take the CONTROL ERROR OUT -signal or use the CONTROL SET POINT. The set point, which is programmed by command SDACV, goes from 0.005V to +2.99 Volts and has 50μV settling resolution. This voltage comes directly from the LTC2600 analog-to-digital converter. It is not buffered nor protected by the AVS-48SI.

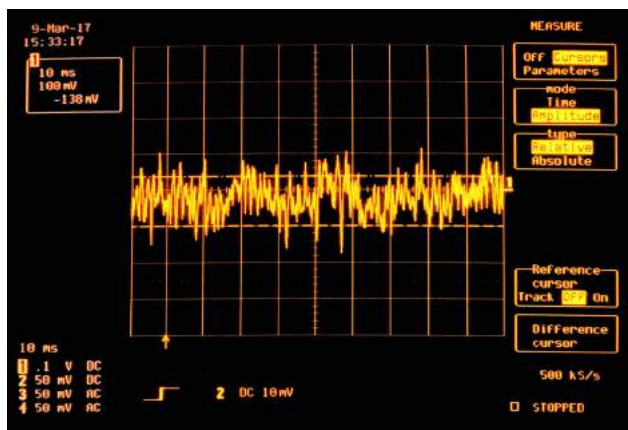
**AC Monitor Out.** This is the preamplifier's output amplified by the main amplifier's variable gain.

EXCITATION	GAIN
10mV	1000
3mV	1700
1mV	4200
300μV	12000
100μV	38000
30μV	120000
10μV	340000
3μV	340000

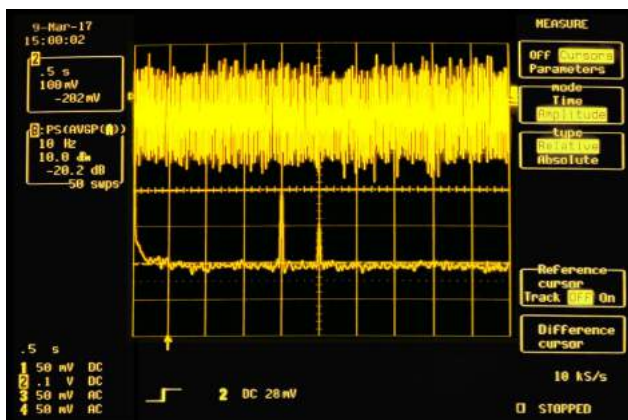
The above gains are (analog out)/(voltage drop across sensor). A 1V<sub>pp</sub> 50Hz signal on scope screen when excitation is 3μV, means 3μV<sub>pp</sub> mains frequency interference at the input.

If the bridge is in balance, the AC amplitude is very small. It contains noise whose source is mainly the voltage noise of the preamplifier input. When a high resistance is measured at low excitation level, the noise is increased. This output is very useful in checking that there is nothing unwanted in the signal that is received from the sensor.

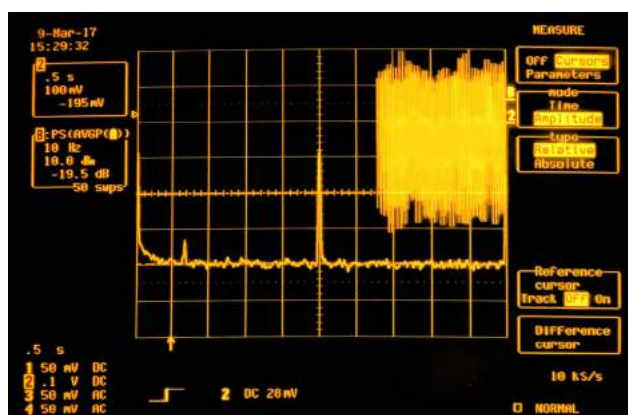




A sound trace looks like this: Excitation is  $3\mu\text{V}$  and the measured resistance is zero. The preamplifier was set on material that damped vibrations. The place was free from changing magnetic fields.



Excitation is  $3\mu\text{V}$  and bridge was measuring  $100\Omega$  on  $3\text{M}\Omega$  range. A  $100\text{nVpp}$   $40\text{Hz}$  signal was injected to the sensor. The FFT trace (average of 50 sweeps) shows also a little  $50\text{Hz}$ , but no other unwanted interferences. This is an ideal case!



This FFT was taken from an internal wirewound reference resistor when the preamp was placed as such on the table. Vibration from the scope's fan increased the  $50\text{Hz}$  peak tenfold. The many turns of wirewound resistors make them sensitive to vibration and changing magnetic fields.

Connect a digital oscilloscope with FFT capability to this output and measure the FFT spectrum. It should be quite flat with the exception of a small peak at the operating frequency and a small peak at your mains frequency. Make the same measurement from one of the AVS-48SI's internal references when the preamp is on the table, not connected to anything else than to the main unit. An ideal measurement of a real cooled sensor should look as good.

If excitation is reduced by one step, the noise level will increase by a factor of 3. However, on the two lowest levels, noise does not increase so much because the amplifier gain is reduced.

**User Dac.** This is another output from the LTC2600 converter that is available for you, if you need an accurately adjustable voltage. The User DAC is available also while the Set Point DAC is needed for temperature control. User DAC goes from  $0.005\text{V}$  to  $+2.99\text{V}$ . The accuracy is better than  $100\mu\text{V}$ . This 16-bit converter has  $50\mu\text{V}$  settling resolution.

The exceptionally good DAC accuracy has been achieved by measuring the DAC voltage by the LTC2400 A/D converter. The reading is then used for generating a correcting software feedback. Some commands and operations are slow because of the required A/D conversion.

**Ext Heater Drive In.** The analog "External heater drive" -signal is very similar to the digital "Direct Heater Drive". External drive is a voltage from 0 up to  $+3\text{V}$ . It must be supplied to this rear panel terminal by some external voltage source, whereas direct heater drive is controlled by software (HTRDIR, HDACV).

Before using External Heater Drive, the top cover of the AVS-48SI must be opened and the short-circuit piece in jumper JP1, near to the left rear corner of the "B" controller board, must be moved from INT to EXT. The heater output stage is now a voltage-controlled current source. Its heater ranges and power displays work, but the PID circuit and set point are bypassed. 0 input results in no output and  $+3\text{V}$  gives the maximum output of the selected range. .



Move the short circuit piece from INT to EXT for driving the heater output stage by an external voltage.

**Ext Clock In.** You can change the AVS-48SI to operate also on other frequencies than 12.5/13.64/15.0 Hz. This requires an external 0/+5V square-wave generator to be connected to the external clock input. If the generator has a low output impedance (typically 50Ω), it will override the bridge's internal clock.

The equation for setting a new frequency  $F_{PSD}$  is

$$F_{PSD} = \frac{F_{EXT}}{600Hz} * F_{SOFTWARE}$$

Where  $F_{SOFTWARE}$  is the operating frequency currently selected by software. If "Ext Clock In" is not loaded, it outputs a 600Hz square signal.

**To AVS-48SI Preamplifier.** The cable consists of 25 wires within a shielding braid. The braid is very important, as it is responsible for the low impedance of the 5-meter ground connection between preamplifier and main units. This 1:1 cable has two DB25P plugs, and the signals are:

#### Preamplifier cable and connector

1	Ground
14	+15V
2	-15V
15	Relay +15V
3	Reference square 1
16	Reference square 2
4	Excitation preamplifier output 1
17	Excitation preamplifier output 2
5	Excitation feedback 1
18	Excitation feedback 2
6	Signal preamplifier output 1
19	Signal preamplifier output 2
7	Signal feedback 1
20	Signal feedback 2
8	Sensor bias voltage measure output
21	External sensor bias input
9	Capacitance compensation square-wave
22	Synchronous serial data to preamplifier unit
10	Synchronous serial clock to preamplifier unit
23	Synchronous serial strobe to preamplifier unit
11	Inverted power-on signal
24	Return current path for relays (see pin 15)
12	-15V
25	+15V
13	Ground



## **AVS-48SI SERIAL COMMAND SUMMARY**

### **INITIAL COMMANDS**

IDN? and \*IDN? Identification query (includes firmware version)  
HW? Hardware query (only CPU hardware version)  
LINETERM[0..3 | ?] Line terminator command/query (1 for LabView, 2 or 3 for hyperterminal)

### **CONFIGURATION OF MEASUREMENT**

PSDF[0..2 | ?] Operating frequency command/query (12.5, 13.64, 15.0 Hz)  
CH[0..7 | ?] Sensor channel command/query (0=calibration references)  
RAN[0..7 | ?] Range command/query  $3\Omega$ ,  $30\Omega$ ,  $300\Omega$ ... $30M\Omega$   
EXC[0..7 | ?] Excitation command/query  $3\mu V$ ,  $10\mu V$ ,  $30\mu V$ ... $10mV$   
GNDS[0..1 | ?] Sensor grounding command/query (0=floating, default. 1=I- lead grounded)  
TW[0..1 | ?] 2- or 4-wire measurement command/query (0=4-wire, default)  
ARN[0..60 | ?] Autorange command/query (argument= autorange delay in sec., 0=disabled)

### **CONFIGURATION OF TEMPERATURE CONTROL**

HTRRAN[0..18 | ?] Heater range command/query (0= heater disabled, all params. are reset to 0)  
PROPG[0..13 | ?] Proportional gain command/query (0 = contr. error does not affect output)  
INTG[0..10 | ?] Integrator gain command/query (0=integrator is held at reset)  
DERG[0..10 | ?] Derivator gain command/query  
INTHEATER[0..1 | ?] Internal/external heater command/query (0=external, normal heater)  
HTRDIR[0..1 | ?] Heater driver source command/query (0=PID circuitry drives the output)  
HDACV[0.005..2.99 | ?] Heater DAC voltage command/query (DAC drives heater when HTRDIR=1)  
SDACV[0.005...2.99 | ?] Set Point DAC voltage command/query  
SETPOINT[0..30,000,000] Temperature control set point command (Set point in ohms)  
HTRI? Heater current query (current is measured, result in Amperes)  
HTRV? Heater voltage query (voltage is measured, result in Volts)  
HTRP? Heater power query (= HTRI? \* HTRV? result in Watts)  
ERRSIGNAL? Error signal query (result in Volts)  
DRDT[0..3 | ?] Control polarity command/query (0 for sensor with  $dR/dT > 0$ . 1 for negative)  
HOLDMODE[0..1 | ?] Hold mode command/query. 1=enter hold, 0=release hold, query returns the current holding status (0=normal, 1=holding).

### **COMMANDS FOR SAVING INTO EEPROM**

SAVEBRD Saves the currently effective bridge settings as properties of the currently selected channel.  
SAVETCR Saves the currently effective PID parameters as properties of the currently selected heater range.  
SAVECAL Saves calibration coefficients for all range-excitation pairs.



**SAVEADC** Saves calibration coefficients for the A/D converter. Affects also analog outputs.

**SAVEREF** Saves stated or measured “true” values of the calibrator resistors.

**SAVELINETERM** Saves the selected line terminator to be the default

**PRESETMODE** Enables entering properties of channels and heater ranges and saving them with **SAVEBRD** and **SAVETCR** without making the settings effective.

## COMMANDS FOR RECALLING SAVED SETTINGS

**RECALLBR[0..7]** Recalls from the EEPROM a channel’s properties and makes them effective.  
Argument=channel to be recalled.

**RECALLTC[1..18]** Recalls from the EEPROM a heater range’s PID parameters and makes them effective.  
Argument = heater range to be recalled.

## COMMANDS FOR REAR PANEL OUTPUTS

**BNC[0..1 | ?]** Analog Out Command/query. Selects one of two analog voltages to the output. 0 = voltage proportional to sensor resistance, 1 = voltage proportional to  $(V_{\text{SENSOR}} - V_{\text{SETPOINT}}) * 3V$  (Reversed by setting **DRDT=1**).

**UDACV[0.005..2.99 | ?]** User DAC voltage command/query. Command programs the DAC, query measures the DAC output voltage.

**SDACV[0.005..2.99 | ?]** Set point DAC voltage command/query. Command programs the DAC, query measures the DAC output voltage.

## CALIBRATION

**CALIBRATE[0..2]** Main Calibration command. Starts one of three automatic calibration procedures:

- 0: Slow full calibration determines corrections for all range-excitation pairs. Long averages and settling times for maximum accuracy. Takes about 4 hours to complete.
- 1: Fast full calibration. Short averages and waiting times. Faster but less accurate, especially on low excitations. Takes about 1 hour.
- 2: Partial calibration. Calibrates only the currently selected range-excitation pair, but uses long averages and settling times. Required time depends on excitation.

The above three calibration procedures can be terminated by sending anything via the RS232 interface.

**CALADC** Calibrates the A/D and D/A converters so that they comply with an external voltmeter that is used for this calibration. Required only, if such a compliance is desired. See commands **DVMHI** and **DVMLO**.

**CALREF[0..1]** Enables entering or measuring and entering values of the calibrator resistors. The accuracy of the AVS-48SI depends on how accurately values of these 7 resistors are known. 1 starts the calibration session, 0 ends it. Requires a high-end, calibrated 4-wire ohmmeter. See commands **REFID** and **REFVALUE**.

## OTHER COMMANDS

**DLY[0..30]** Delay command in seconds for time separation between commands on a single line. It can be used for letting the bridge to stabilize before **RESx** and **ADCx** commands.

**REPEAT** Repeats commands/queries on a single line in an endless loop until terminated by sending any-



thing via the RS232 interface. REPEAT must be the last item on the line.

TIME and TIME? Interval measurement command and query. They are for measuring the time taken by commands and queries or groups of them. Multiline groups are possible, but then the result includes also delays that are external to the AVS-48SI. Result is in milliseconds.

TIME is placed before the sequence to be measured. TIME? is placed after the sequence.

RESTART Takes bridge to start-up state. Does not alter nor save any properties or calibration.

DEFAULTS Sets all properties to safe initial values and saves them into EEPROM. Does not change calibration in any way.

RESETALL Resets all properties and calibration factors and saves them into EEPROM. Everything must be re-calibrated and re-programmed.

EPRCAL Reads the offset and scale correction DAC voltages into RAM

EPRADC Reads the offset and scale correction factors for the ADC into RAM

EPRREF Reads the “true” stated values of the references from EEPROM into RAM

ERR? Error query. Place at the end of a line. Response is text (string) , except that “0” means no error. Several error messages may have been concatenated into a long string. The error register is emptied by the ERR? query.

MAX?, MIN? Maximum and minimum readings among the group of averaged data points.

STD? Standard deviation of the averaged A/D conversion data points.

QRATIO? Quality ratio is  $(\text{max}-\text{min})/\text{std}$ . Is about 5 for a healthy measurement. Unreliable on the highest excitations as the digitizing step overrides random noise.

CP[0..1] Turns off or on the CP indicator light (service)

DC[0..1] Turns off or on the DC indicator light (service)

## INTRODUCTION - Part I

This Part I is for operating **AVS-48SI Picobridge®** via its versatile **USB** or **RS232** serial interface (the letters “SI” denote “Serial Interface”). The optional factory-installable **Temperature Controller** is included in these instructions.

The bridge alone is **AVS-48SI**. Together with a **AVS48-Serial/USB** CPU box, connected by a wire cable, the bridge becomes **AVS-48SI-W**. If the wire cable is replaced by a 5-meter **AVS48SI-Picolink** optical link, the bridge becomes **AVS-48SI-F5**. Optional increase of the fibre length to 10 meters makes an **AVS-48SI-F10**.

The AVS-48SI is a very versatile and perhaps world’s most accurate cryogenic resistance bridge.

The AVS-48SI bridge is divided in three parts:

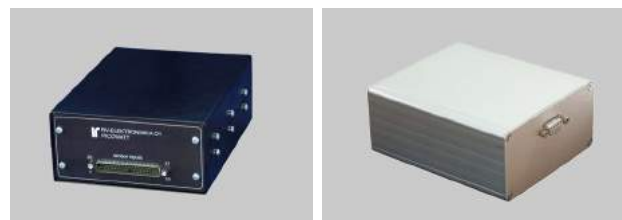
- 1) a small preamplifier box, which can be mounted near to the cryostat in order to keep the sensitive cables to the sensors short
- 2) a 19-inch wide main unit for rack mounting inside the shielded room. The preamplifier and the main unit are connected together by a 5-meter cable.
- 3) a small external microprocessor unit. This CPU box should be *located outside the shielded room* in order to minimize the electromagnetic interference that a processor unit can generate. It is connected to the main unit by a 5-meter shielded cable that is galvanically isolated in order to prevent ground currents. The box provides a simple RS232 interface and a wide command set for full control of the hardware. This new version of the AVS-48 can be used with any computer, operat-

ing system and application software, including LabView, which support RS232 communication. If the computer does not offer a physical COM: port, a virtual port can be created by using a USB-232 converter.

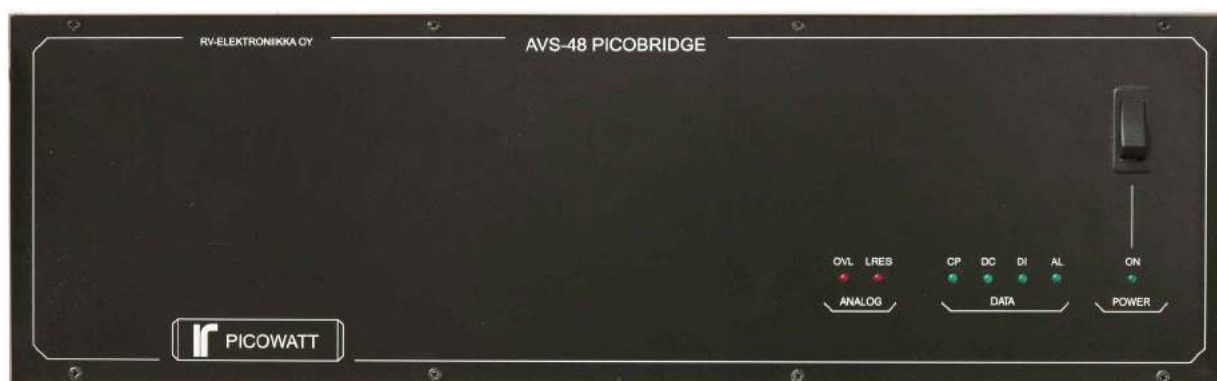
The AVS-48SI can be also equipped with a fibre-optical connection between the microprocessor unit and the main unit. In a case where the shielding braid of the wire cable cannot be grounded to the conducting wall of the cryostat room, the optical cable provides the EMI-safest solution. It is also easy to take the optical cable through small holes, as it needs no connectors.

The sensor input of the Picobridge is pin-to-pin compatible with that of our popular AVS-47x series bridges, making upgrading easy. The analog instrument itself is a “black box” without a display, keyboard or digital intelligence of its own. The necessary firmware is provided by an external microprocessor unit.

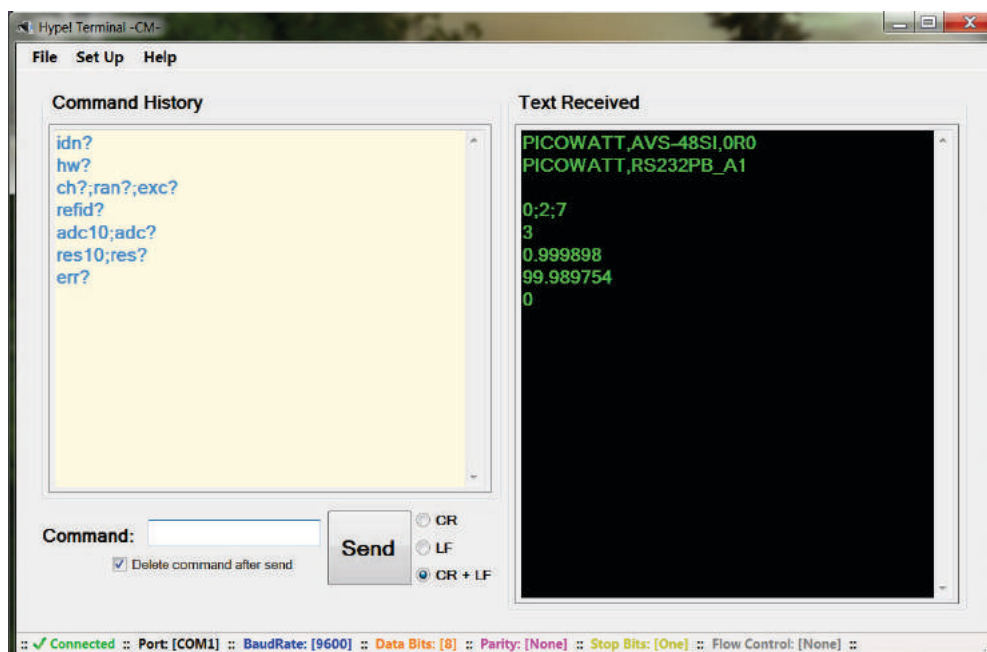
A factory-installable **temperature controller option** offers many improvements compared with our old TS-530A, such as more tightly spaced power ranges, smooth changing of heater power range, and a hold mode for visiting other sensors temporarily.



The 170x130x65mm preamplifier can be located near to the cryostat, whereas the 103x123x55mm microprocessor unit should be located as far from the cryostat as possible



The 19” wide main unit of the AVS-48SI Picobridge



The “Hype! Terminal” (see text for link) is easy to use: Just specify the Com port number (SetUp) and it is ready. In the above picture: identification and hardware queries tell the firmware version and the CPU hardware version. Then the currently active channel, range, excitation and also the ordinal number of the calibration reference resistor are queried. The result of ten ADC conversions is a voltage, whereas the RES? query gives the answer in ohms. Response to ERR? is zero, which means no error.

## BEGIN WITH A HYPERTERMINAL

A very good way to get acquainted with the serial commands is to use a hyperterminal program, which lets you send commands to the AVS-48SI and read the response. For Windows users, we can suggest a program called “Hype!Terminal” written by Carlos Montoya. It is downloadable presently from

<https://sourceforge.net/projects/hypeterminal/>

In the following text, we will use examples that have been tested with Hype!Terminal. When writing your own computer application, keep a hyperterminal program at hand so that you can test your command sequences and measure the time they need.

A hyperterminal program is also useful for calibrating the instrument, where a simple command starts a process that needs a long time to complete but does not produce output.

## SHORT BRIDGE TUTORIAL

These instructions assume that you have a hyperterminal program running on your computer. If your computer does not have a physical Com: serial port (which is also called an RS232 port), but you have a free USB port, you can convert it to a Com: port by using a USB-232 converter. A converter made by National Instruments is available from us, but many other -and perhaps cheaper- converters are also commercially available.

Make the connections between parts of the system as instructed in the previous paragraph. Connect also a digital voltmeter to the ANALOG OUT rear panel BNC. Check that the short-circuit plug is inserted in the preamplifier’s 37-way input connector.

Check the serial port number that the hyperterminal program will use. It must correspond to the physical COM port or the port name that the USB-232 converter is using.

Switch on the AVS-48SI. A green light in its front panel shows that power is on.





Switch on the CPU box by plugging its wall adapter into a mains socket. The universal power supply will work from 90 to 250V, 50-60Hz. Some activity on the AVS-48SI's front panel lights will show that the CPU has booted and that the firmware program is running.

Your DVM on the analog output should now show 1.000 Volts to a few  $\pm 100$  microvolts accuracy.

Start the hyperterminal program and check the serial port number - correct if necessary. Send the query IDN?. The response will be "PICOWATT,AVS-48SI,1R0" or some newer firmware version. Send query HW?. The response "PICOWATT,RS232PB\_A1", is the hardware version of the CPU box. You may need to change the line terminator character(s) depending on your hyperterminal program, if the response did not show up or if there are no line breaks. This is made by the LINETERM(argument). The arguments are

- 0: no termination
- 1: LF (newline, 1\n, ASCII 10)
- 2: CR (carriage return, 1\r, ASCII 13)
- 3: CRLF (initial factory default)

Check the present state of the bridge by writing a line of queries: "CH?;RAN?;EXC?" (quotes are for clarity, do not include them in the messages). The response will be "0;2;7" indicating channel 0 (calibration references), 300 $\Omega$  range and 10mV excitation.

Measure the calibrator as "RES;RES?". The response is in ohms. If you issue "ADC;ADC?", the response is in Volts. Compare the response with your DVM's display.

Reduce the excitation to the 3 $\mu$ V minimum by sending "EXC0". Now the DVM display will be quite noisy, as so is also the response, if you repeat the RES;RES? sequence several times. Reduce the fluctuation by using an average: send "RES100;RES?". This will take about 20 seconds to complete, but the noise will be much lower, which you see by repeating this a few times.

The AVS-48SI offers some statistics queries: MAX?, MIN?, STD? and QRATIO?. The peak-to-peak noise (max-min) of those 100 conversions could be something like 0.0035 and the standard deviation 0.0007. These readings are in Volts. If the noise is pure random, the ratio (max-min)/std (=qratio) is about 5. A much higher reading suggests

an interfering spike. QRATIO? is not very reliable on high excitations because of digitizing steps.

Measure the short-circuit input plug by sending "CH1;RAN0;EXC5". Channel, range and excitation will now change, and after a reasonable settling time, RES10;RES? will yield about zero.

It is not necessary to write everything each time a channel is measured. Save a channel's settings and enable autoranging for it:

```
CH0;RAN2;EXC7;ARN10;SAVEBRD  
CH1;RAN0;EXC5;ARN10;SAVEBRD
```

Now the channels are measured by

```
RECALLBR0;DLY10;RES5;RES?  
RECALLBR1;DLY10;RES5;RES?
```

The ten seconds delay (DLY) allows the bridge to settle before starting to measure. If autorange needs to change the range, the 10 second autorange delay (10 after ARN) is added to DLY.

## **AVS-48SI SERIAL INTERFACE COMMANDS AND QUERIES**

The serial commands that will be described below cover the necessary functions for creating computer controlled applications. If you feel that some further commands would be necessary or useful, please let us know.

The second part of this manual describes, how to use the AVS-48SI hardware together with the "avs48si.vi" virtual instrument LabView program. The hardware specifications, connection of the instruments, cabling and physical front and rear panels are the same for both applications. The LabView program offers more software functionality and is ready to use without need to write a line of code, whereas use of the serial interface does not require LabView. Automatic calibration is supported in both cases.

All calibration coefficients are stored in the Arduino Mega2560 CPU, and therefore:

- calibrations made using LabView are valid and available for serial operation
- calibrations made via the serial interface are valid and available for LabView operation.



## COMMANDS, QUERIES AND RESPONSES

Communication between a computer and the AVS-48SI consists of sending messages, i.e. commands and queries, to the bridge and reading responses from it. The AVS-48SI does not send anything by itself, it can only respond to queries. Because of this strict rule one does not need to poll the computer's serial input after commands, but only after queries. Commands are performed in the order they were received and responses are received in the same order as they were queried. Responses have no headers, which makes them easier to read into a program. Some operations, like changing range or excitation, need a lot of time, and one must be prepared to that.

All commands and queries consist of two parts: a) two or more letters and b) an argument.

- The letter part is case-insensitive (RAN1 = ran1)
- There can, but needs not, be one or more spaces between the two parts (RAN1 = RAN 1 = ran 1)
- In most cases, commands and queries have identical letter parts (command EXC7, query EXC?)
- Not all commands have corresponding queries and not all queries have corresponding commands
- The argument of a query is a single “?” character (e.g. LINETERM?). A query without the “?” is assumed to be a command whose argument is 0.
- The argument of a command must be numeric. Some commands do not have an argument, which the program then assumes to be zero (e.g. RE-START). Similarly, if only a letter part of a command or query is given, it is assumed to be a command whose argument is zero (RAN = RAN0, REFVALUE = REFVALUE 0).
- Most numeric arguments are integers. Values of analog voltages and set point must be given as floating point numbers (e.g. SETPOINT 100.123). Do not give numbers in exponential form, they are not understood by the Arduino CPU.
- Arguments have their own minimum and maximum values. If a given argument exceeds its maximum or falls below its minimum, it is coerced to the maximum or minimum and the command is performed without any error message (CH9 will be coerced to CH7).

## COMMAND DELIMITER

Several commands and/or queries can be sent on one message line. The items must be separated from each other by a command delimiter, which is “;” (semicolon, ASCII 59). This delimiter is also used in IEEE-488.2 standard. A delimiter must not be used in the end of a message line.

The message line must be shorter than 255 characters, including spaces and delimiters.

## MESSAGE LINE TERMINATOR

A message line must be terminated by a line terminator character or characters. The AVS-48SI accepts three commonly used terminators (see LINETERM). Instead, your computer program or the platform may expect only one of them, and therefore the line terminator can be specified and saved. When the AVS-48SI responds to queries, it uses only the specified terminator. Try the terminators and save the working one.

## TIMING

The simple interface without handshaking and the firmware program that does not use interrupts require a lot of attention to timing.

The first thing is to make use of the fact that the AVS-48SI does not send anything by itself. It needs a query to which it can respond. However, commands take very different times to complete, and if you use timeouts in your program, make sure that enough time is reserved for waiting for the response to come. It is best to write only such message lines that have all queries at the end, so that when responding starts, responses to all queries come in quick succession.

You do not need to worry about timing between commands within one message line: they are performed in the order that they came and they just take the time that they need. Responses are in the same order as the queries were on the message line.

However, **NOTHING** must be sent to the AVS-48SI while it is processing previously given commands or queries. This is difficult to remember, but there are five ways to make sure that the next message line is delayed until the CPU is ready to accept a new message.

1. Add the “Operation Complete” query OPC? as the last item on a line, if it contains only commands. Poll the computer’s serial input until character “1” (without the quotation marks) is received. The AVS-48SI is then ready to receive a new message line.
2. Add the “Error Query” ERR? as the last item on a line that contains only commands. This works like OPC?, except that you must be prepared to handle a text string response. If there is no error on the line, response is 0. In case of an error, the response is a message like “analog overload”.
3. If the line has queries or both commands and queries, place at least one -but preferably all- queries in the end. You can then trust that all responses come in, say, 100 milliseconds after the first character of the response string was detected.
4. Poll the response for the specified line terminator. Your program can stop reading the response as soon as the line terminator has been received. Then a new message line can be sent..
5. Use delays. This simple method has the serious drawback, that commands need different times to complete. Response to a simple query like RAN? comes in twenty milliseconds, whereas changing range or excitation requires about two seconds. A complete automatic calibration takes about four hours. This must be taken into account, if your program contains timeouts.

For simplicity, these synchronisation methods are not shown explicitly in this manual’s examples, but use some of them in your programs. When testing commands with a hyperterminal program, the need for synchronisation is difficult to see, as one cannot send message lines in too quick succession. Use the hyperterminal program and the TIME/TIME? command/query for measuring how much time your command sequences need. Most commands/queries need less than 60ms. In the following, there are some estimates in curly brackets: {time1 | time2}, where time1 is for command and time2 for query.

Check the timing, if your program gives strange results or if it does not work as expected!

## GENERAL INSTRUMENT COMMANDS

In the following text, [CH] in the end of a command description means, that this value is saved as a property of the currently selected channel. [BRD] means that the value is saved as a general setting common to all channels and is read from memory at start-up. [HTR] means that the setting is saved as a property of a heater range. Otherwise, the setting or parameter must be entered again for a new session.

IDN? and \*IDN? Identification query (Response e.g. “PICOWATT, AVS-48SI,1R0”)

HW? Hardware query (only the CPU hardware, e.g. “PICOWATT,RS232PB\_A1”)

LINETERM[0..3 | ?] Response line terminator command/query.

0 = no terminator

1 = LF (linefeed) ASCII 10

2 = CR (carriage return) ASCII 13

3 = CRLF

A message line must be terminated by a line terminator. The AVS-48SI accepts alternatives 1-3, whereas an application program may accept only one of them. For example, LabView requires that serially received strings are terminated by LF. The selected termination should be saved using SAVELINETERM. [BRD]

## RESTART DEFAULTS RESETALL

Refer to the descriptions of these commands. The RESTART command does not destroy any properties that have been pre-programmed, it only takes the AVS-48SI to its start-up state. It does not save anything.

The DEFAULTS command sets all properties and other settings to safe initial values and also saves them into the EEPROM. Use with care.

The RESETALL command resets all calibration factors and all properties of all channels and heater ranges. Use with extreme care! Everything must be re-calibrated and re-programmed.

**RESTART** Restart command. Re-starts the CPU and sets it to default state: Reference channel #0, 300R range, 10mV excitation, 100R reference resistor, 4-wire and floating sensor, no autoranging and previously selected line terminator. If the program has hanged, this command will not work. In such a case, plug off and on the mains power adapter in order to get a real power-on boot.

## CONFIGURING THE BRIDGE

**PSDF[0..2 | ?]** Excitation frequency of the bridge (command/query)  
0 = 12.5Hz, 1=13.64Hz, 2 = 15.0Hz  
[BRD]

**CH[0..7 | ?]** Input channel command/query.  
Channel 0 is connected to one of seven internal ultra-stable wire-wound reference resistors. Channels 1-7 are available for sensors.  
[CH]

**RAN[0..7 | ?]** Range command/query. Ranges from 0 to 7 are: 3Ω, 30Ω, 300Ω, 3kΩ, 30kΩ, 300kΩ, 3MΩ and 30MΩ. Two DACs are re-programmed, which takes time. [CH] {1.4s | 20ms}

**EXC[0..7 | ?]** Excitation command/query. Excitations from 0 to 7 are: 3uV, 10uV, 30uV, 100uV, 300uV, 1mV, 3mV and 10mVrms. Excitation current is the nominal excitation voltage divided by one third of the selected range. Two DACs are re-programmed, which takes time. [CH] {1.4s | 20ms}

**GNDS[0..1 | ?]** Sensor grounding command/query.  
0 = floating sensor. None of a floating sensor's leads must be in contact with the cryostat ground. This preferred connection prevents ground currents.  
1 = grounded sensor. A grounded sensor's current return lead (-I lead) is grounded to the cryostat, e.g. for better thermal contact. The software setting must correspond to the wiring of the sensor. Each sensor can have its own grounding setting. [CH]

**TW[0..1 | ?]** 2- or 4-wire measurement command/query.  
0 = 4-wire connection with separate two wires for excitation current and two non-loaded wires for the voltage drop (default),  
1 = only two leads for both voltage and current. Measured value includes lead resistances. [CH]

**ARN[0..60 | ?]** Autorange command/query.  
0 = autoranging is not used.  
1 - 60 = autoranging is enabled for the currently selected channel.

The argument is "autoranging delay" in seconds. Its purpose is to allow a settling time after an autoranging operation. For example, ARN5 means that there is a 5 seconds delay after an automatic change of range before any further operation can take place (usually one or more A/D conversions). The same delay is used between subsequent autoranging operations in a case, where range must be changed several times.

If autorange has to change range, the new range will be automatically saved into EEPROM. If this channel is later recalled, the AVS-48SI will try the new range. [CH].

## CONFIGURING THE TEMPERATURE CONTROLLER

Any sensor can be used as the control sensor. Set it for measurement and select the sensor's DRDT polarity. The control set point can be given only in ohms, because the Arduino Mega2560 CPU does not offer enough storage space for R/T conversion tables. The "avs48si.vi" LabView program features temperature readout.

A special feature of the AVS-48SI temperature controller is, that the output power, not the voltage or current, is linearly dependent on error signal. Output from the PID circuit is a linearly varying voltage, but then an approximate square root is taken before sending the drive to the output stage. The desired result of this is, that the closed-loop gain does not change with power level. Without the square root, loop gain at 1W level would be 10 times higher than at 0.1W level.

Connection to the heater is described in the

paragraph “Physical Rear Panel” under title “Heater Output”.

A typical “standard” resistance is 100Ω, but the current source heater output of the AVS-48SI accepts any value. The maximum power of 3W is achieved when heater is 50Ω. If heater resistance is higher than 50Ω, the output voltage compliance limits the achievable maximum power, if it is lower, the maximum output current limits the power.

**HTRRAN[0..18 | ?]** Heater range command/query. HTRRAN0 disables the heater so, that no current flows into the heater. Unlike with the LabView program, HTRRAN0 does not automatically reset other settings. If you need to do it: HTRRAN0;PROPG0;INTG0;DERG0 (ITG0 keeps the PID integrator’s voltage at zero).

A special feature of the AVS-48SI is that changing heater range during control has no practical effect on the output power if the system was in equilibrium before the change. The long time, about 1s, that HTRRAN takes is due to this feature. Please observe, that there must be enough power available also on the new range after a change downwards. The proportional gain may need to be increased in order to maintain response speed (see SAVETCR and RECALLTC).

Heater range can always be changed upwards, but the P-gain may need to be reduced in order to avoid instability. Controlling at a very low percentage of the maximum range power should be avoided.

Maximum power with a 100Ω heater:

0. control off	10. 3.81 mW
1. 1 μW	11. 10.0 mW
2. 2.5 μW	12. 24.9 mW
3. 6.2 μW	13. 61.9 mW
4. 15.4 μW	14. 154 mW
5. 38.1 μW	15. 381 mW
6. 100 μW	16. 1.00 W (0.5W@50Ω)
7. 249 μW	17. 1.53 W (1.24W@50Ω)
8. 619 μW	18. 1.53 W (3.06W @50Ω)
9. 1.54 mW	

**PROPG[0..13 | ?]** Proportional gain command/query. Proportional amplifier’s output is held at zero when PROPG=0. If PROPG is made zero during temperature control, the heater output will be solely dependent on the PID integrator’s voltage. [HTR]

**INTG[0..10 | ?]** Integrator gain command/query. INTG0 resets the integrator and keeps its voltage at zero. A higher gain means stronger integration action. [HTR]

**DERG[0..10 | ?]** Derivator gain command/query. DERG0 means no derivation. A higher gain means stronger derivation action. The derivator has a modified response, which concentrates its effect to low frequencies and suppresses derivation at high frequencies.[HTR]

**INTHEATER[0..1 | ?]** Internal heater command/query. The AVS-48SI has a 100Ω 2W internal heater mainly for test purposes. It is in thermal contact with a forward-biased silicon diode. They can be used for testing the controller (see DRDT).

0 = Normal external heater (default),  
1 = Ext. htr. replaced by internal 100Ω heater.

**HTRDIR[0..1 | ?]** Heater Driver Source command/query. The temperature controller’s output stage can be driven either by the PID circuit or by an internal D/A converter. 0 = PID, 1 = DAC. The converter is programmed as below.

**HDACV[0.005..2.99 | ?]** Heater Direct DAC voltage command/query. This is the D/A converter that can be used for driving the heater output stage instead of the analog PID circuitry. The DAC does not go to exact zero. Use one of these two methods for nulling the heater output: 1) set the heater range to HTRRAN0, 2) direct the heater current to the internal heater by INTHEATER1. {0.7s|0.3s}  
The query HDACV? measures the DAC output – it is not the programmed digital value.

**SDACV[0.005...2.99 | ?]** Set Point DAC voltage command/query. The set point can be given in addition to resistance (SETPOINT), also in voltage. A voltage set point is independent on



the control sensor's measuring range. Query measures the DAC output voltage. {0.7s | 0.3s}

**SETPOINT[0...]** Set point command. This command is for giving the temperature control set point in resistance (ohms). The argument must be a floating point number, no exponential representation. The entered resistance is converted to voltage on the basis of the currently selected control sensor's measuring range. The resulting voltage can be queried by **SDACV?**. *If the measuring range is changed, the SETPOINT command must be repeated,* because the resistance-to-voltage conversion takes place only after this command. {0.1s}

**HTRI?** Heater output current query. Response is a floating point number (Amperes). Current measurement works well even on the lowest heater ranges. Use the  $I_{\text{HEATER}}^2 * R_{\text{HEATER}}$  equation for calculating heating power more reliably. The heater output is a current source, and therefore  $I_{\text{HEATER}}$  does not reveal short circuits between heater leads. {0.5s}

**HTRV?** Heater output voltage query. Response is a floating point number (Volts). The measured voltage is low on low heater ranges and may contain a significant offset. The full Main Calibration procedure measures this offset and saves the result, which is automatically subtracted from subsequent HTRV? responses. The offset can also be determined by measuring HTRV? when HTRRAN=0.

HTRV? measures the voltage at the output of the AVS-48SI. It does not take into account the possible voltage drop in the heater leads. HTRV? does react to shorts between heater leads. {0.5s}

**HTRP?** Heater output power query. It returns the product  $HTRI? * HTRV?$ . It is better to calculate the power as  $I_{\text{HEATER}}^2 * R_{\text{HEATER}}$  on the lowest power ranges if more accuracy is needed. On the other hand, HTRP? has the benefit, that it reacts to both shorts and open

circuits in the heater connection. {0.5s}

**ERRSIGNAL?** Error signal query. This is the measured difference between the control sensor's value and the set point, both as voltages. Error signal's polarity depends on DRDT as below. {0.5s}

**DRDT[0..3 | ?]** Control polarity command/query. 0 = for sensor that has a positive dR/dT, 1 = for negative dR/dT. DRDT determines also the behaviour of the rear panel ERROR BNC output:

- 0: signal - setpoint voltage
- 1 : setpoint voltage - signal
- 2 : testvoltage - setpoint voltage
- 3 : setpoint voltage – testvoltage

Testvoltage is an internally generated voltage of about 0.6-0.65 Volts (forward biased silicon diode in thermal contact with the internal 100Ω heater resistor "INTHEATER").

A simple test to the temperature controller can be done as follows: Connect a DVM to the rear panel CONTROL ERROR output. Give commands "INTHEATER1;DRDT3;SDACV0.64;PROPG10;HTRRAN16". The error signal should now start to approach zero. When approaching becomes slow or stops, issue "INTG5". Approaching should start again as integrator resets the remaining control error. The final error should fluctuate around zero.

Query the output:

HTRI? => 70-80mA  
HTRV? => 7-8V  
HTRP? => ~ 0.5W

**HOLDMODE[0..1 | ?]** Hold mode command/query.

Command: System goes into hold mode by issuing HOLDMODE1. System comes off from the hold mode by issuing HOLDMODE0. Both processes take some time.

Query: 0 = normal operation, 1 = holding.

In hold mode, the PID integrator's output is replaced by a D/A converter. It is programmed to output the voltage that is measured by the Go-to-Hold process. You can visit and measure other sensors in this mode, but you must

return back to the control sensor before exiting hold. Some drift during hold is unavoidable, as there is no active control. The amount of drift depends on how near the system was initially to equilibrium and how stable is the heat load. The procedure itself causes almost no change in control temperature. {2.3s for 0=>1, 60ms for 1=>0}.

## A/D CONVERSION

ADC[1..1000 | ?] Analog-to-Digital Conversion command/query

**Command:** Makes one or more conversions, whose mean value is calculated. 100 conversions are made in 20 seconds.

If any of the readings exceeds 2.8V or falls below 0.2V, autorange will change the range if autorange is enabled. Refer to the ARN command.

ADC0 or without argument is ADC1.

**Query:** Outputs the calculated average. The result is a floating point number in Volts. The A/D conversion function checks for some errors, so the ERR? query after ADC? can be useful. {216ms}

RES[1...1000 | ?] Resistance Command and Query.

**Command:** Same as ADC with another name.

**Query:** The voltage is converted to resistance on the basis of the currently effective measuring range. Otherwise same as ADC?. {216ms}

## STATISTICS

MAX?, MIN? These queries return the maximum and minimum ADC values (in Volts) of conversions averaged for one ADCx command.

STD Standard deviation of the averaged ADC conversions. Use a long average for a reliable STD. If noise is purely random, STD behaves as noise: two times higher STD means two times higher voltage noise at the input. If the measured data has a trend, STD is of little use.

QRATIO This is simply (MAX-MIN)/STD. For a purely random noise, QRATIO is about 5. A much higher figure may tell about an interfering spike in the data, whereas a much lower figure can tell about insufficient number of samples. Qratio is also affected by the digitising step when excitation is high. It is best used at low excitations.

## SAVING SETTINGS FOR LATER USE

Settings and parameters are saved into an EEPROM, whose write count is guaranteed up to 100,000 times. A worn out memory cannot be replaced. In order to prevent the write count from growing unnecessarily, the firmware does not save anything automatically. Instead, the user must explicitly save items that need to be in memory. This is the list of the save commands, and what they do:

**SAVEREF** True values of the 7 reference resistors are saved by this command. Refer to “Calibrating the References” for a description on how these values are entered and saved or measured and saved. This calibration is needed only in few cases.

**SAVEADC** This saves the scale factor and the zero offset of the A/D converter. Refer to “Calibrating the A/D and D/A converters” for how to calibrate the converters so, that the analog outputs comply with your own DVM. This calibration should not be done, unless such a compliance to a high accuracy is needed.

**SAVECAL** 128 correcting factors for the scale and offset of the resistance bridge are saved by this command, which is to be used after the main calibration procedure. Each combination of range and excitation has its own pair of numbers, which correct the analog and digital outputs of the self-balancing bridge circuit. Refer to “Calibrating the Bridge”.

**SAVEBRD** Saves the following properties of a channel for later RECALLing of that channel: Range, Excitation, 2/4-wire sensor connection, floating or grounded sensor and autoranging delay (autoranging is enabled if delay is >0).

- If you want to save the currently effective



settings for a channel, give the command  
SAVEBRD.

- If you are not sure about the current settings, you can first query them.
- For programming all settings for channel No. x, construct a line like this:  
CHx;EXC2;RAN3;TW0;ARN5;OPC?

Poll the response for character “1”. When it shows that the whole line has been handled, issue SAVEBRD. The drawback of this method is, that the bridge actually goes to the programmed settings, which can lead to unwanted measuring of some sensors at a high excitation current. Refer to description of “PRESETMODE” on how to program the channels quickly without changing bridge state.

**SAVETCR** Each of the 18 heater ranges of the temperature controller can have its own PID parameters (proportional, integration and derivation gains). The idea is, that different control temperatures often need different heating power and that thermal parameters of the system can change a lot at very low temperatures. Making the PID parameters dependent on power level is therefore somewhat equivalent to making them dependent directly on temperature. Just as with the bridge, the PID parameters can be saved and recalled as properties of a heater power range.

Saving properties of the currently selected heater range is made by issuing the SAVETCR command after suitable parameters have been made effective. Refer to “PRESETMODE” for how to program many heater ranges quickly.

Example: HTRRAN5;PROPG10;ITG5;DERG0;SAVETCR;OPC? {1.2s}

**SAVELINETERM** Saves the line terminator, which has been previously set by command LINE-TERM[0..3].

**PRESETMODE[0..1]** Preset mode is for saving properties into the EEPROM without making them effective, in other words, without affecting the states of the bridge or the temperature controller. Programming the channels and

heater ranges is much faster in the preset mode and there is no danger of accidental excess heating of sensors or the heater. For example, pre-program range and excitation of channels 1 and 2:

```
PRESETMODE1    (starts the mode)
CH1;RAN4;EXC3
CH2;RAN5;EXC2
SAVEBRD        (saves both channels' properties)
PRESETMODE0    (ends the mode and sets
the bridge to the last programmed channel)
```

Recall the pre-programmed Channel #1 by  
RECALLBR1.

Properties of a heater range are pre-programmed in the same way:

```
PRESETMODE1
HTRRAN5;PROPG7;INTG5;DERG0
HTRRAN6;PROPG8;INTG6;DERG0
SAVETCR
PRESETMODE0    (controller is left to the last
programmed heater range using its new PID
parameters).
```

For example, take heater range #5 in use:  
RECALLTC5.

The PRESETMODE1 command needs 20ms whereas return to normal with PRESETMODE0 needs 1.3 seconds.

## RECALLING SAVED SETTINGS

**RECALLBR[0..7]** The saved properties of a channel can be made effective by recalling. Argument is the number of the channel to be recalled. The bridge goes immediately to the recalled state. {1.6s}

If you simply change the channel from CHx to CHy by sending only CHy, this new channel will be measured using settings that were effective for CHx, not those that are in the non-volatile memory for CHy. The idea is, that you can use the same settings for any channel by just selecting the channel. You can alternatively use individual settings for each channel by first saving and then recalling them.

Recalling is very handy and, with autoranging, it requires a minimum number of commands in a scanning application. If autoranging changes range, the new range is automatically saved into the EEPROM. When this channel is recalled again, the bridge tries the new range and no further autoranging operations are needed.

A simple scanning application of pre-programmed channels 1-3 would need only

```
RECALLBR1;DLY10;ADC20;RES?
RECALLBR2;DLY10;ADC20;RES?
RECALLBR3;DLY10;ADC20;RES?
```

Should one need to change the excitation of e.g. CH2:

```
RECALLBR2;EXC[new exc];SAVEBR;OPC?
```

RECALLBR2 makes CH2 active and so ensures that the new excitation is saved for CH2.

**RECALLTC[1..18]** Recalls the heater range (= the argument) with its saved PID parameters. The recalled settings are made effective immediately. RECALL0 is not included, because it disables controls, sets all parameters to zero and resets the integrator. This could disrupt the control session. {0.3s}

If you select a new heater range without recalling it, the currently effective PID parameters, instead of the saved ones, are used for the new range.

## REAR PANEL ANALOG OUTPUTS

### ANALOG OUT

**BNC[0..1 | ?]** Rear panel Analog Output command/query. The Analog Out BNC can output two quantities:

- 0: Voltage proportional to  $(R_{\text{SENSOR}}/\text{Range}) * 3V$
- 1: Voltage proportional to  $(V_{\text{SENSOR}} - V_{\text{SETPOINT}}) * 3V$   
(Reversed, if DRDT=1)

BNC0 is the voltage measured normally by the ADC or RES commands.

### USER DAC

**UDACV[0.005..2.99 | ?]** User DAC voltage command/query. The User DAC is always available for you, if you need a very precise programmable voltage in your laboratory. The settling resolution of the 16-bit DAC is 50µV and absolute calibration accuracy ±100µV. The output is not protected, do not supply energy into it. {0.7s}

UDACV? query causes the output voltage to be measured by the A/D converter. It is not the digital programmed value and therefore it can be slightly different. {1.1s}

By building an external scaling/level shifting circuit, arbitrary unipolar or bipolar DAC outputs can be realized. It is also possible to place such a circuit inside the AVS-48SI unit. A +/-15V power connector is available inside for such purposes. Contact factory.

### CONTROL SET POINT

**SDACV[0.005..2.99 | ?]** Set Point Voltage command/query. The Set Point DAC is also available for your own purposes, if temperature control is not in use. It has the same settling resolution and calibration accuracy as the User DAC. The output is not protected. {0.7s | 1.1s}.

The SDACV command is used for setting the set point voltage in applications, where the analog CONTROL SET POINT voltage is used for some external purpose. It does not depend on measuring range and it does not need to be re-entered if range is changed.

On the contrary, the SETPOINT command is used in temperature control applications. Then set point is given in ohms, and the firmware converts it to DAC voltage, which depends on the current measuring range. This conversion takes place only after the SETPOINT command. Therefore, this command with its previous argument must be repeated after the measuring range was changed.

## CALIBRATION

The AVS-48SI has three self-calibration processes for maintaining its accuracy, which is exceptionally good among low temperature resistance bridges.

- The accuracy of the instrument's digital output relies on **Main Calibration**. Although this procedure takes a lot of time, it can be made in situ without moving the instrument, it is completely automatic and no external instruments are needed. The Main Calibration cancels most of the effects of temperature changes and aging of the instrument. The more often Main Calibration is run, the better.
- **Calibration of the A/D and D/A converters** changes the offset and scale factors of both the analog-to-digital and digital-to-analog converters. The idea of this calibration is not to achieve an absolute accuracy but to make the AVS-48SI to comply with your own laboratory instruments. This calibration is not necessary unless accuracy of such a compliance is required. An external DVM is necessary.
- The Main Calibration relies on seven wire-wound ultra-stable resistors. Their "true" values have been measured before delivery and saved into a non-volatile memory. In some rare cases, e.g. if the preamplifier has been replaced, those values must be either entered or first measured and then entered and saved again. This is made with the help of the **Reference Resistor Calibration**. A very accurate high-end 4-wire ohmmeter is required.

## MAIN CALIBRATION

During a full Main Calibration, seven wire-wound ultra-stable reference resistors are measured, each using its best corresponding measuring range and all 8 excitations. In addition, each range-excitation pair is used for measuring a zero resistance. This makes 128 measurements altogether. On the lowest excitations, averages consist of 800 A/D conversions. Therefore this command takes a long time to complete. The full calibration can be run in the nighttime.

## CALIBRATION PRINCIPLE

The idea of this procedure is to compare readings from the measured references with the stated "true"

values of them and to calculate correction factors that make them equal. But unlike with most other instruments, the correction factors do not affect the digital output alone. Instead, they fine-adjust the offset of the phase-sensitive detector and the magnitude of the excitation current so that measuring the analog output results in readings that correspond to the references. That is why the main calibration corrects also the analog output.

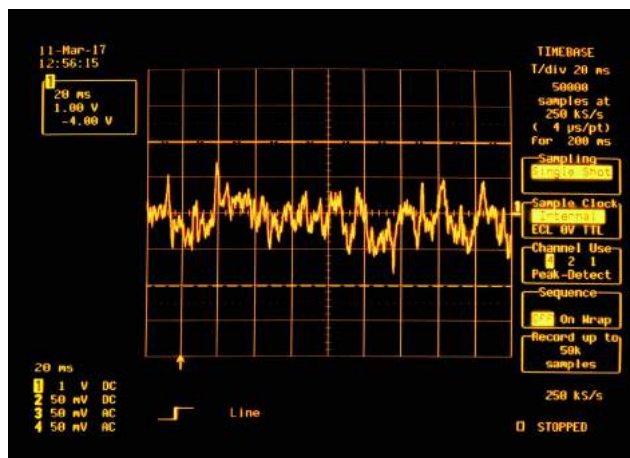
## AFTER CALIBRATION

Completion of the long calibration procedure is indicated by lighting the CP and DC indicators on the AVS-48SI's physical front panel. Completion can also be seen by adding the ERR? query after the CALIBRATE command. If there has been no error (response = 0), the calibration should be tested by measuring most relevant reference resistors at the most relevant excitations. Compare the readings with the stated "true" values of the references. Those values are obtained by REFIDx;REFVALUE? (x=0..7). If the values agree, save the calibration into EEPROM by SAVECAL. After that, the old calibration cannot be restored. If you do not save the new calibration, it will be in effect until the CPU is booted, the CPU is RESTARTed, or the old calibration is read from the EEPROM by EPRCAL.

## PRECAUTIONS

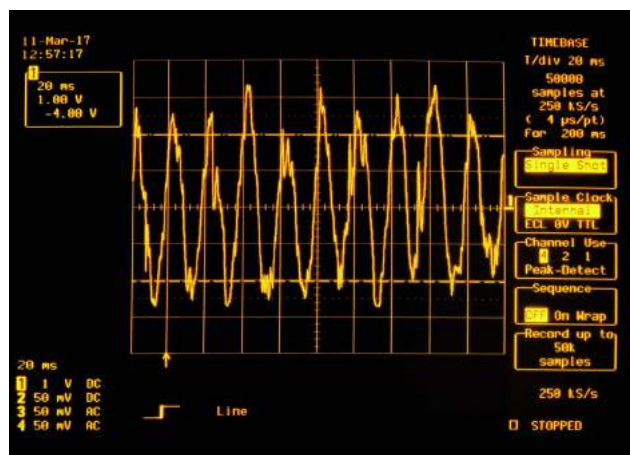
Although calibration of low ranges and high excitations is likely to be successful without any precautions, the situation is different when the highest ranges (100k $\Omega$ ..30M $\Omega$ ) are calibrated using the lowest excitations (3 $\mu$ V..100 $\mu$ V).

The wire-wound resistors are -because of their construction- sensitive to vibration in a magnetic field (like that of the earth!). Also the circuit boards can generate tiny voltages if they vibrate. A successful calibration of the lowest ranges/excitations requires that the preamplifier is isolated from vibration and rapidly changing magnetic fields. In addition, if anything is connected to the preamplifier input, it must be well shielded. Check before starting to calibrate that the signal is OK. Select 3M $\Omega$  range, 3 $\mu$ V excitation and 1M $\Omega$  reference ("REFID7"). Watch the red OVL and green AL lights on the AVS-48SI front panel. If you see either of them blinking, there is probably too much mains hum in the signal



One can try to calculate the input noise from this sound trace as follows: peak-to-peak noise is about 2 volts, as seen from the photo, and rms noise is  $1/5$  of it = 0.4V. The amplifier gain (see “AC Monitor Out”) is 340000, which makes  $1.2\mu\text{V}_{\text{rms}}$  at the input. This must still be divided by  $\sqrt{\Delta f}$ , where  $\Delta f=100$  is the approximate noise bandwidth from an FFT measurement. This calculation yields therefore  $120\text{nV}/\sqrt{\text{Hz}}$ . The noise equation gives  $130\text{nV}/\sqrt{\text{Hz}}$  for  $1\text{M}\Omega$  at  $300\text{K}$ .

For comparison, the input noise of the bridge itself is  $5\text{nV}/\sqrt{\text{Hz}}$ .



One 10cm piece of wire feeds 50Hz hum into the preamplifier. Because of the high measured resistance ( $1\text{M}\Omega$  reference), the hum couples capacitively to the input and generates an interference, which causes overload. Depending on the frequency and severity of clipping, such a signal would lead to a bad or extremely bad calibration of this range/excitation pair. Scales of the oscilloscope are same in both photos.

Vibration or fast changing magnetic fields can also disturb calibration. An actual measurement, in turn, is sensitive to vibration of the leads to the sensors.

for successful calibration of the highest range(s) and lowest excitation(s). These two photos show, how the signal behaves. For getting more information about the signal:

Connect an oscilloscope to the AC OUT monitor output. Because of our “ground everything everywhere”- principle, the oscilloscope can be connected normally with one coax with grounded braid. The first photo shows a sound signal. A short-circuit plug is connected to the input, and the trace never exceeds the  $\pm 2\text{V}$  limits of signal overload. Calibration of this range/excitation pair should succeed unlike in the second photo, where a small amount of mains hum is fed into the box.

If your sensors or the sensor wires are not perfectly shielded, you may need to detach the input connector before starting to calibrate. Insert the short-circuit plug into the input rather than letting it open.

Even if one cannot get a sound signal for the worst ran/exc combinations, the higher excitation ranges or lower resistance ranges may have no problems and then they calibrate successfully.

**CALIBRATE[0..2]** Starts the automatic Main Calibration Procedure. All 8 excitations are used for measuring long averages on all 8 measuring ranges.

**0:** Full calibration of all range-excitation pairs. Used for maintaining maximum accuracy of the bridge. Takes about 3.5 hours to complete, and is therefore intended to be run in the nighttime. When calibration is complete, the green CP and DC indicators on the AVS-48SI physical front panel light until the CPU receives any command via the serial interface. The calibration can be tested before saving it into the non-volatile memory. Save it by command **SAVECAL**.

This calibration calibrates the digital output and eliminates most of the drift of the analog output.

Full calibration must be made, if the ADC and DAC calibration has been run, or if the values of the reference resistors have been changed.



**1:** Same as above, but with much shorter averages and shorter settling times. CALIBRATE1 needs less time, about 30 minutes, but at the cost of reduced accuracy. It is mainly intended for production. Especially on low excitation ranges, the short averages do not guarantee good calibration.

**2:** CALIBRATE2 starts a slow partial calibration. It calibrates only the currently selected range and excitation using long averages just like the full calibration of alternative 0. Use this mode before an accurate measurement that does not need many ranges or excitations. Time required is dependent on excitation, the maximum being about 5 minutes at  $3\mu\text{V}$ .

This procedure affects only the correction factors of the calibrated range-excitation pair. SAVECAL after Calibrate2 updates factors for the calibrated properties, but it retains the calibration of other ranges/excitations.

If the main calibration fails and it has not yet been saved, the old calibration can be restored by command EPRCAL (EePromReadCALibration).

## CALIBRATION OF THE A/D AND D/A CONVERTERS

The offset and gain errors of the analog-to-digital converter are corrected by software, and the correction factors are stored in an EEPROM. They were initially measured as a part of the manufacturing process. It is possible to recalibrate the ADC so, that it corresponds to your own external DVM's reading. Good compliance with your instruments, if that is required, is the main reason to run this procedure. The ADC itself is very stable and it will not need recalibration otherwise.

The outputs of the digital-to-analog converters are corrected, in turn, by measuring them with the ADC and then applying software feedback. Calibration of the ADC adjusts all the rear panel outputs so, that they agree with your DVM. Note that the absolute accuracy of this calibration cannot be better than your DVM.

### PROCEDURE:

- 1) Connect your DVM to the CONTROL SET POINT rear panel output
- 2) Send command CALADC. This starts the mode.
- 3) The DVM should show a voltage near to  $0.1000 \text{ Volts} = V_{\text{LOW}}$
- 4) Enter this value by sending command  $\text{DVMLO } V_{\text{LOW}}$
- 5) The program proceeds, and your DVM should be near to  $2.8000 \text{ V} = V_{\text{HI}}$
- 6) Enter this value by sending command  $\text{DVMHI } V_{\text{HI}}$

The procedure has now all the data it needs, and the mode ends by itself. ADC calibration is now complete, and your DVM should show 2.8000 V. You can test the new calibration by the SDACV[0.005...2.99] command.

Save the new calibration into EEPROM by "SAVEADC". If the new calibration is not saved, it remains in effect only until the CPU is restarted or rebooted. Then old coefficients are read from the EEPROM. Alternatively, the old ADC coefficients can be read back into RAM by issuing command "EPRADC".

Full calibration must be made, if the ADC and DAC calibration has been run.

## ENTERING VALUES OF CALIBRATION REFERENCES

Values of the 7 calibration references are saved in the non-volatile EEPROM memory of the CPU. The accuracy of the bridge depends on these ultra-stable resistors. If the preamplifier unit is changed, values of the new unit's references must be entered the firmware and saved. The values, originally measured by Picowatt, are written on a sticker inside the preamplifier's bottom cover. The reference resistors are measured by the bridge as any other sensor (select CH0 and REFID[0..7]). If the AVS-48SI has been just calibrated, the measured values correspond to the saved values to a high accuracy.

The “true” values of the references can be entered into the program by first giving the ordinal number of each resistor (REFID) and then its stated value (REFVALUE[argument]). When all or any of the 7 values have been entered, the result can be checked against spelling errors by giving REFIDx and then querying REFVALUE? The new, checked values are saved by sending command SAVEREF.

REFID[0..7 | ?] Reference order number command/query. The references have the following nominal values from number 0 to 7:

Zero, 1, 10 and 100Ω, 1, 10 and 100 kΩ  
1MΩ. REFID[x] selects a reference and REFID? query returns the ordinal number of the reference that is currently selected.

REFVALUE[0..1100000 | ?] Reference value command/query. This command is used for giving a value to reference, whose number was previously specified by the REFID command. Use floating point representation, exponential format is not understood by the Arduino2560 CPU.

The REFVALUE? query returns the value of the reference, whose ordinal number was specified by REFID[x]. Use this command carefully: If you by mistake type REFVALUE without the question mark, the program interprets this as a command with argument = 0 and sets this resistor’s value to 0 ohms.

Values given by the REFVALUE command remain in RAM memory until the CPU is booted (at power-on). Then the old values are read from the EEPROM. The old values can also be recalled by command EPRREF.

The new values are saved permanently by sending SAVEREF command (no argument).

## MEASURING VALUES OF CALIBRATION REFERENCE RESISTORS

This procedure requires an external high-end ohmmeter and a special cable that must be built by the user.

The ohmmeter must be capable of measuring resistances from 1Ω to 1MΩ with a resolution of 1E-5. Because of the lowest reference values, 4-wire measurement is necessary and so is also offset

compensation (the DVM alternates the polarity of the measuring current). *Recalibration of the 7 reference resistors should be done only, if one is sure that this does not degrade the calibration.* Picowatt has records of the values at the time of shipping. They have also been written on a sticker inside the bottom cover of the preamplifier.

1. Prepare the measuring cable. The shielded cable must have 4 wires for +I, +V, -V and -I. Make the connections to your ohmmeter as required.

The AVS-48SI accessories include a pre-wired Molex 22-01-2045 connector:

pin	function	wire color
1	+I	brown
2	+V	red
3	-V	orange
4	-I	yellow

Solder this cable to your prepared DVM cable. Photos on the next page show, how the cable is connected.

Measurement and entering the values is made using the CALREF procedure. After this command the bridge is in a mode, where the reference resistors are isolated from other circuitry so, that they can be measured. During the bridge’s normal operation, this would not be possible.

CALREF[0..1] Command for starting/ending calibration mode for the references with the aid of an external ohmmeter. Before starting to measure the reference values, query and write down the previous values (REFIDx;REFVALUE?). They cannot be restored after the new values have been saved. This procedure is best used from a hyperterminal program.

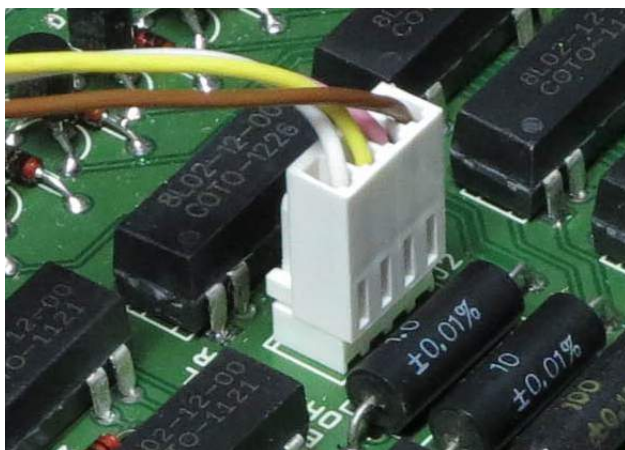
- 1) Send CALREF 1 for starting the mode
- 2) Send REFID 1.
- 3) Read the resistance R1 from your ohmmeter. Write it down as a floating point number in ohms.
- 4) Send REFVALUE[ R1 ]
- 5) go to 2 and command REFID 2 and so on.

When all 7 resistors has been measured and





A 4-way cable is connected to the multiplexer board. The shield is grounded



Wires from right to left: +I, +V, -V, -I. The white current return lead (-I) is in contact with circuit ground and the box.



The bottom cover is shielding the multiplexer board against low-frequency interference. It should be in electrical contact with the chassis in order to be useful.

entered, remove the external ohmmeter and send CALREF0 for ending this mode. Check the new values for typing errors by using REFID[x] commands and REFVALUE? queries. The entered values remain in effect until the CPU box is re-started. Command SAVEREF (without argument) saves them in the EEPROM. If the new values are not saved, old values are read at start-up. They can also be read from the EEPROM to RAM any time by sending command EPRREF (no argument). This command does not restore anything else.

If you see a typing error or otherwise a suspect value, you can start this mode again. It is not necessary to go through all values, you can measure and enter only the wrong value and then save again.

Full main calibration must be run after having changed any of the reference resistor values.

## OTHER COMMANDS

**DLY[0..30]** This delay function inserts a delay between two commands or queries on the command line. Argument is seconds. Do not use DLY as the last item on a row, because it does not prevent sending a new line too early. Use OPC?, ERR? or any other query as the last item and poll for the response.

**REPEAT** This command repeats one command line continuously. The commands and queries behave exactly similarly as if only one the line was sent. If the line includes queries, the CPU sends responses and your program can read them as they come. Repeating is stopped by sending anything via the RS232 line. Examples with a hyperterminal program:

ADC5;ADC?;REPEAT

response:

1.2345

1.2347

1.2343 etc.

UDACV1;DLY2;UDACV2;DLY2;REPEAT

Response: The user DAC output toggles between 1 and 2 volts. The DLY2 gives the DAC voltage time to stabilize so that you can see both final values.

**TIME** and **TIME?** These are used for measuring the time taken by a command, query or a complete command line. The **TIME** command is placed before the sequence to be tested, and the **TIME?** is set after it. The results are in milliseconds. Examples:

TIME;TIME? => 10

10ms general overhead

TIME;ADC;TIME? => 215ms

sigma-delta ADC alone needs 180ms

TIME;ADC100;TIME? => 19537ms

100 conversions at max. speed, 19.5s

TIME;RAN3;TIME? => 1371ms

Command requires that two DACs are reprogrammed. Needs 1.37s.

TIME;UDACV2.5;TIME? => 671ms

(the UDACV and SDACV commands need only 20ms if argument was not changed).

**RESTART** Restart command. Re-starts the CPU and sets it to default state: Reference channel #0, 300R range, 10mV excitation, 100R reference resistor, 4-wire and floating sensor, no autoranging and previously selected line terminator. Does not change pre-programmed properties nor save anything. Plug off and on the mains power adapter in order to boot also the operating system of the CPU.

**DEFAULTS** Sets all properties of the bridge and temperature controller to safe initial values and saves them into EEPROM:

Selected: Channel 0, Ref. resistor 100Ω, Range 300Ω, excitation 10mV, no autorange.

Other channels: Range 30MΩ, excitations 3μV, sensors floating, 4-wire, no autoranging, lineterminator CRLF, all PID parameters zero.

**RESETALL** This command resets all calibration coefficients:

- Offset and scale correction DACs are both set to 1.5 Volts, which is in the middle of their 0.005..2.99V output range.
- The offset voltage correction factor of the ADC is set to 0.00 and scale correction factor to 1.00.
- The “true” values of the reference resistors are set to 1Ω, 10Ω...1MΩ.
- The reset values are saved into EEPROM.
- The old values, settings or properties cannot be restored.

Use this command with great care! After saving the reset values, everything must be calibrated again.

## ERROR MESSAGES

**ERR?** Error query should be placed in the end of a message line. Output is a text string, where a plain “0” means no error. Several error messages may be concatenated into one string.

**Commands:** Because a command never produces a response, there is no way for the bridge to inform about a command error. One must use the **ERR?** query routinely or if an error is suspected.

**Queries:** Because a query always produces a response, the CPU sends a question mark, if there was a query error or if the ADC? query reports about an error. In order to know more about the error, issue the **ERR?** query.

Error register is emptied by the **ERR?** query. Many commands and autoranging also reset the register, so it must be queried as close to the error source as possible.

Error message, if a command or query is unknown or misspelled

Command [command] not recognized

Query [query] not recognized

Error messages from A/D conversion:

adc overrange V > 3V

adc underrange  $V < -3V$   
analog error (AL=1 during A/D conversion)  
AC signal overload OVL  
High lead resistance LRES

#### Error messages from calibration

Any error message from A/D conversion  
calibration aborted by user  
offset correction  $> 2.9V$   
offset correction  $< 0.1V$   
scale correction  $> 2.9V$   
scale correction  $< 0.1V$

## CABLE SPECIFICATIONS

The AVS-48SI-W comes with two cables, Picobus Cable and Serial Cable. The 5 meter Picobus cable connects the converter to the AVS-48SI and the 1.5 meter Serial cable connects the box to the computer directly or via an USB-232 converter. The -F version has an additional 5- or 10- meter optical cable, whereas the 5-meter Picobus cable is shortened to 1.5m.

**Picobus Cable** (PB9P15P6W5M) for -W version  
**Picobus Cable** (PB9P15P6W1.5M) for -F version

DE9P pin	DA15P	color	Picobus name
1	9	brown	+5V (floating)
2	--	--	
3	--	--	
4	15	grey	DC
5	7	white	GND (floating)
6	6	lila	AL
7	4	yellow	CP
8	5	green	DI
9	--	--	

The male DE9P and male DA15P are connected by a braided (shielded) cable with 6 conductors (e.g. Tasker C6015). Length: 5 meters for -W, 1.5 meters for -F.

Note that the shielding braid must remain isolated from the 15-pin connector shell of the -W type cable. A 100nF ceramic capacitor with shortest possible leads connects the braid to the connector shell. The

braid **is** connected to the shell of the 9-way connector at the AVS-48SI end.

This arrangement reduces the antenna effect at high frequencies while not providing a path for ground currents at the mains frequency. Inside the AVS-48SI, Picobus signals and its power supply are galvanically isolated from the bridge ground. We recommend strongly that the braid is connected firmly to the conducting wall of the shielded room at the entering point. This completely prevents the cable from taking high frequencies inside the Faraday shield.

With the 1.5M long cable (-F version), braid is connected directly to both connector shells. No capacitor is needed. This is because the AVS48SI-Picobus Box1 is located inside the shielded room near to the AVS-48SI main unit and it shares the common ground with the bridge

#### Serial Cable (RS9P9S7W1.5M)

The male DE9P and female DE9S are connected by a braided 1:1 cable of 6 conductors (e.g. Tasker C6015). This cable is for RS232 communication between the CPU box and the computer (or USB-232 converter). The DSR, RTS, CTS and DTR leads are reserved for other applications using Picobus, they are not used for RS232 communication. Length: 1.5 meters. Same for both the -W and -F versions.

DE9S DE9P	RS232	Description
1 -	-	-
2 brown	RXD	RS232 output box=>PC
3 lila	TXD	RS232 input PC=>box
4 yellow	DTR	DC (for Picobus appl.)
6 green	DSR	AL (for Picobus appl.)
5 braid	Computer ground	= shielding braid
7 white	RTS	CP (for Picobus appl.)
8 gray	CTS	DI (for Picobus appl.)
9 -	-	-

Shielding braid is divided in two parts at both ends. One is connected to pin 5, and the other part to the connector's metal shell.

## RE-PROGRAMMING THE ARDUINO MEGA2560 CPU (Windows)

The firmware can be updated by reprogramming the Arduino Mega2560 board. In order to do this, you need

- a USB cable (type A/B)
  - Arduino development software for Mega2560. It is best to download this from Arduino WEB site.
  - The new firmware version. It is available from us.
1. Follow Arduino's instructions to download and install their programming environment software.
  2. Open the four screws holding the "front" panel of the CPU box (the panel with the DA15S connector or opto connectors). The Picobus cable from this connector to the AVS-48SI must have been removed. Plug the "A" type connector into the USB connector of your computer and the "B" connector into the "hidden" USB connector inside the CPU box (refer to the photo). The box will now start, as it gets power from the computer's USB.
  3. Start the Arduino programming environment. Select the Arduino Mega2560 board type and check the serial port number.
  4. Open the downloaded new version of the "avs48232\_xry.ino" software ("sketch", as they call it). From the Sketch menu, select UP-LOAD. If you do not get any error messages, updating has been done in a few seconds. You can now detach the USB cable, fix the rear panel and connect the 25/15 Picobus cable to the resistance bridge. Then test the new firmware using an RS232 terminal program or your own software.

Before asking us to email an updated version, please check and tell us your old firmware version so, that we can send to you also the old version for backup. The version number is the last item in the response to IDN?

Re-programming does not alter any of the calibration factors nor saved channel or heater range properties.



The CPU unit is opened by removing the front plate of the DA15 connector side. Then the cover plate can be pulled off. If it does not come out easily, unscrew also the two topmost screws on the other side.





## SERVICE MEASUREMENTS

The A/D converter has access to many internal nodes of the AVS-48SI circuitry. Many of them are measured by various commands listed in this manual. There are also nodes whose voltage levels can provide valuable information in case of troubles. All points are listed below without more explanations. Contact factory for more information.

The voltage to be measured is selected by the ADCINP[0..25 | ?] command (service tab). All responses are in volts.

- 0: Ground 0V
- 1: ADC reference voltage ( 3.00V)
- 2: DC level of signal preamp. output
- 3: DC level of signal main amp. output
- 4: **BRIDGE OUTPUT** (normally used, ADC?)
- 5: DC level of exc. preamp. output
- 6: DC level of exc. main amp. output
- 7: Exc. channel PSD integrator output
- 8: Sensor DC bias voltage (not in use)
- 9: Integrator volt. of capacit. compens. circuit
- 10: PID control error
- 11: PID proportional ampl. output
- 12: PID integrator voltage
- 13: PID derivator output
- 14: PID Square-root circuit's output voltage
- 15: PID heater drive voltage
- 16: PID heater output voltage (HTRV?)
- 17: PID heater current (HTRI?)
- 18: Excitation adjust DAC voltage
- 19: Offset adjust DAC voltage
- 20: Excitation fast settle DAC voltage
- 21: Signal fast settle DAC voltage
- 22: Set Point DAC voltage (SDACV?)
- 23: PID integrator fast settle DAC voltage
- 24: User DAC voltage (UDACV?)
- 25: PID heater direct DAC voltage (HDACV?)

## TROUBLE SHOOTING IDEAS FOR THE SERIAL COMMUNICATION

If you have difficulties in establishing serial communication between your computer and the AVS-48SI, you can try the following trouble-shooting procedure. The problem might not be found this way, but it at least gives us valuable knowledge of where in the system the problem possibly is.

For making these tests, you need a hyperterminal program, or any other program that allows you to send serial commands to the CPU box. Typically, your modern computer has only USB ports and then you need a USB-232 adapter between the computer and the CPU.

Communication between the AVS-48SI bridge and the CPU box uses four signals, two from the box to the bridge and two from the bridge to the box. They are

AL: "alarm" line indicating a possible analog error (OVL or LRES). Bridge => Box

DI: data bits Bridge => Box

DC: data bits Box => Bridge

CP: clock pulses Box => Bridge

Both the sent and received data bits are synchronized to the same clock. Communication *between the CPU and the bridge* is based on our proprietary synchronous, serial "Picobus" protocol. Although this communication is serial, it is NOT RS232. Please do not mix synchronous and asynchronous serial formats. In synchronous Picobus communication, long strings, that are sent from the CPU to the bridge, contain the *state* of the bridge. They are not commands. Some data, like the A/D conversion result, can be read from the bridge to the CPU. Asynchronous serial communication *between the CPU and the computer*, on the other hand, uses short mnemonic commands and queries for controlling individual bridge settings and for reading conversion results and currently effective settings.

The command interpreter recognizes and interprets the messages that it has received. Various functions (subroutines) build the required Picobus strings and send them to the bridge, which does not have any digital intelligence of its own. The RF emissions from the AVS-48SI are therefore exceptionally low.

Because of the slow speed of the AVS-48SI



and tiny amounts of data, the simplest possible serial protocol can be used (9600 bauds, 8 data bits, no parity and one stop bit. No handshaking). It is based on signals called TxD, RxD and Ground. TxD transfers data from the external computer to the box, and RxD from the box back to the computer.

## PROCEDURE FOR THE -W VERSION

- 1) Remove the four cross-head screws that hold the front panel of the CPU box (the panel with the 15-way connector). Then pull off the top cover lid. Place the box, so that the front panel is to the left. You can see a row of several green and two red LEDs. They are for trouble shooting. LEDs on the left (“bridge side”) show signals to an from the primary interface of the AVS-48SI bridge. They are marked with AL, DI, CP and DC. At right are the two asynchronous signals, TxD and RxD. The DI, AL, CP and DC signals on the right have no meaning in this application. They can light or not.
- 2) Connect a DVM to the ANALOG OUT rear panel BNC of the bridge.
- 3) Connect the serial cable (RS9P9S7W1.5M) from the box to the USB-232 converter or to your computer, if it has a physical COM: port (RS232 port). Connect the DE9/DA15 cable (PB9P15P6W1.5M) from the AVS-48SI bridge to the CPU box. Switch on the bridge.
- 4) Connect the +12V DC power plug. The green “ON” light, and one yellow LED on the Arduino CPU board should turn on after some blinking. This tells that
  - the CPU gets power and that
  - the CPU has booted
- 5) Reduce the baudrate of the hyperterminal to 300 or 600. Send the command RESTART and at the same time watch the TxD light. It should blink. This shows that the computer sends via TxD and that the CPU box is receiving the transmission. The CPU does not react to this command as it uses 9600 bauds. Increase baud rate back to 9600.
- 6) Send commands CP1 and DC1. The corresponding LEDs in the CPU box should light on and stay lighted. The same with the LEDs on the

bridge front panel. This indicates that

- the commands have been interpreted correctly, the firmware program is running, and
- the CPU sends Picobus data to the bridge, which has received the data successfully.

- 7) Send command RESTART again. The DI, DC and CP lights both in the CPU box and on the front panel should blink, which shows that the bridge is sending data. The bridge should go to its default state: CH0, 300Ω range and 10mV excitation. The DVM should show about 1 Volt.
  - Three of the four Picobus lines are OK.
  - This is a big macro command. Correct result shows that the bridge can be remotely controlled.
- 8) Send command RAN1. This will cause overload as the bridge measures 100Ω on the 30Ω range. The green AL lights both on the front panel and in the CPU box turn on and the red OVL front panel light starts to blink. This indicates that also the remaining AL Picobus line is OK. The DVM will show about 4.2 Volts.
- 9) Send query IDN? The response “PICOWATT,AVS-48SI,1R1” ends with the version number (1R1 or a later version) of the firmware. A correct response tells that serial data sent by the CPU to the computer has been received successfully.

If the test stops to a failure, please let us know your results so that we can try to help.

## PROCEDURE FOR THE -F VERSION

- 1) Connect the CPU box to the computer or to the USB-232 converter using the 1.5m long DE9P/DE9S cable (RS9P9S7W1.5M). Connect the AVS48SI-Picolink Box1 to the AVS-48SI bridge by the 1.5m DB9P/DA5P cable (PB9P15P6W1.5M).
- 2) Connect a DVM to the ANALOG OUT rear panel BNC of the bridge.
- 3) The three most common reasons, why the system with opto link does not start are: the fibres are in

wrong order or some of the fibres has not been inserted properly or an end of a fibre is not in good condition.

- the fibres are marked with colors. The order in “Box1” (near to the bridge) is 1,2,3,4 from left to right. The order in the CPU box (the one with the mains adapter) is 4,3,2,1. Use 1=black, 2=red, 3=yellow and 4=blue.

- Check that the ends of all fibres are cut sharply. If suspect, use sharp full-flush end cutters to remove a millimeter or two from the fibres.

- the fibres must be pushed as deep as they go. Insert depth is 17mm for the black and 16mm for the white connectors. Sometimes a fibre wants to stop earlier.

- The receivers/transmitters and their threads are mechanically weak and vulnerable. Tighten only by fingers in order not to break the things.

Insert first only fibres 1 and 2 and only to the CPU box, leave other fibres intact. As you now start the CPU by plugging in the +12V wall adapter, the black and red fibres should blink several times with the same brightness. If this happens, connect these two fibres also to the Picolink Box1.

- 4) Give command CP1 from the hyperterminal. The front panel CP (Clock Pulses) led should light on the bridge’s physical front panel. Give command DC1 and the DC (Data from “Computer”) led should also light. This shows that the bridge receives both signals.
- 5) Insert fibres 3 and 4 to AVS48SI-Picolink Box1, but not yet to the CPU box. Give command ADC;ADC? Fibre 3 should blink two times when the A/D converter sends the response from the bridge to the CPU using the DI (Data from Instrument) line. Fibre 4 can be lighted or not, it depends on the state of the AL (Alarm) signal, which reacts to analog errors. The bridge has not yet been initialized, so the state can be random.
- 6) Insert the remaining fibre ends. Disconnect and reconnect the +12V power plug or the wall adapter. The bridge should enter into state, where it measures an internal 100Ω resistor using 300Ω range and 10mV excitation. The DVM connected to the Analog Out BNC should show 1. If you

give command SDACV1.2 (Setpoint DAC Voltage = 1.2V), the set point BNC should be at 1.2 Volts.

7. Select the 30Ω range. This causes signal overload, which should be shown by the red OVL warning light and the green AL light on the physical front panel as well as the AL light inside the CPU unit.

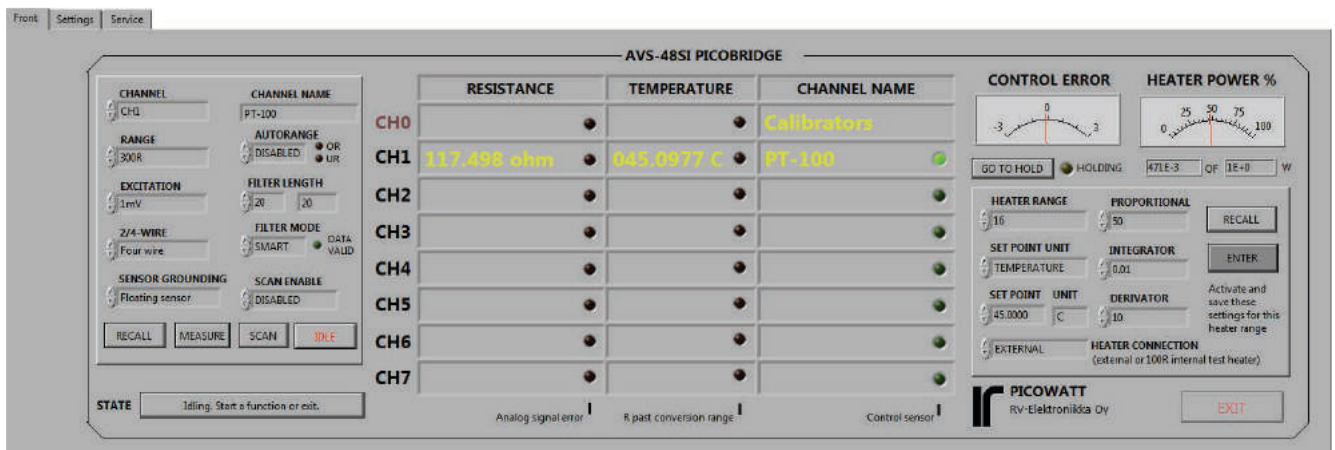
If this test fails somewhere, please contact us so that we can together try to find out, why.

RV-Elektroniikka Oy Picowatt  
Veromiehentie 14  
FI-01510 VANTAA  
Finland

e-mail: [reijo.voutilainen@picowatt.fi](mailto:reijo.voutilainen@picowatt.fi)  
WEB: [www.picowatt.fi](http://www.picowatt.fi)

## Part II

### LabView Programs





## INTRODUCTION

Part II of this manual is for how to operate the **AVS-48SI Picobridge®** and its optional analog Temperature Controller from the “**avs48si.vi**” **LabView™** program. Also calibration of the instrument via LabView is discussed.

The LabView programs use the same serial interface between the computer and the Arduino CPU box as the serial operation described in Part I. These programs also make use of the same commands. Everything has not been written two times. Something that is not here may be found in Part I and vice versa.

All settings and calibration factors are stored in the *EEPROM memory of the CPU*. The LabView programs read data from there and also save changed data into the CPU. This means, that one can change over from using LabView to serial operation and vice versa without losing anything. There are two exceptions: The optional names of the seven channels and all Resistance-to-Temperature conversion files are held in the computer's memory. Sensor names and direct temperature readout are offered only by the “**avs48si.vi**”.

Direct temperature readout, smart filtering, and very informative front panel are just examples of features and functionality that are impossible with the CPU with its limited program and storing space. Important is also the fact that the “**avs48si.vi**” is a complete virtual instrument, ready for use without need to write a line of program code. The measurement results are available via a disk file for use by other programs. They are also available via a functional global variable for other simultaneously running LabView programs.

The exceptionally informative front panel of the “**avs48si.vi**” is divided in three sections: The leftmost area shows all settings that are currently used for measuring the selected sensor. During automatic scanning, the displays change showing the settings of the currently scanned channel.

The area in the center shows the resistance, temperature and the name of the measured or scanned channel. If only one channel is measured, only this channel is lighted. If some channels are scanned,

all scanned channels are lighted so, that one can see results from all sensors at a single glance.

The rightmost side is for the optional temperature controller. There you choose from 18 logarithmically spaced heater ranges, and either set or pre-program its PID parameters. Magnitude of the control error is indicated by a strongly non-linearized animated panel meter, which shows the trend even when the error is large but becomes very sensitive near to balance. In addition to showing the heating power in Watts, an animated analog meter tells the percentage of the delivered power compared with the selected range's maximum.

Just like with serial operation, the AVS-48SI can be equipped with the optional **AVS48SI-Picolink** Optical Fibre Link.

The **AVS-48LVA** version was the predecessor of this “SI” version. There are several important changes between them:

- LVA used our proprietary synchronized “Picobus” protocol for communication between the bridge and the computer, whereas SI uses the well-known asynchronous RS232 protocol.
- LVA required a Windows PC running LabView. The SI version requires only an RS232 or USB port and capability to use RS232 for communication. The LVA could be used only via LabView, whereas the SI can be controlled either using its serial commands or via LabView.
- The LVA saved everything on PC's disk. The SI keeps all calibration and pre-programmed settings stored in the CPU so, that the same stored data can be used both in serial or LabView operation..
- The LVA did not have any digital intelligence of its own but relied totally on the computer. The SI has an external small CPU unit. In order to minimize RF interference, this box should be located outside the shielded cryostat room.
- Optical Link was not available for the LVA, but it is available for the SI version.
- The virtual front panel of the “**avs48si.vi**” works more logically than that of the old “**avs48lva.vi**”.

## INSTALLATION OF THE AVS48SI.VI

Copy the “avs48si\_1r0” (or any later version number) directory with all its contents from the supplied memory stick to your hard disk. You can rename this directory, but include the version number in the name: Updated versions in the future will use many, but not all, same sub-VIs with unchanged names. Keeping new and old versions in separate directories is almost necessary. If you have an older version on your disk, rename the old directory before starting the new version for the first time. Then LabView is sure not to use automatically any of those older files or VIs.

Do not run programs from the memory stick, but keep it as a read-only backup.

There is nothing more to install the programs. Just navigate into the new directory and click avs48si.vi.

## SHORT TUTORIAL

### MEASURING AND SCANNING

These instructions assume that you have at least the base version of LabView2012 or later running on your computer. If your computer does not have a physical Com: serial port (which is also called an RS232 port), but you have a free USB port, you can get a Com: port by using a USB-232 converter. A converter made by National Instruments is available from us (NI part 778472-01), but many other converters are also commercially available. Connect your computer to the CPU box by the 1.5-meter cable RS9P9S7W1.5M. Alternatively, connect the CPU box to your USB-232 converter by this cable. The CPU box is intended to be located far from the cryostat, preferably outside the shielded cryostat room.

Open the measurement program by clicking “avs48si.vi”.

The front panel is divided in three parts: The most often needed controls for the AVS-48SI resistance bridge are on the left, while the measurement results of the 1+7 channels are in the middle. The rightmost side is for the optional temperature controller.

On the bridge side, the visible values are automatically saved for the currently selected *sensor*

*channel* when a resistance measurement or scanning is started. They can -but they need not- be recalled from memory when this channel is selected for measurement next time.

On the controller side, the visible values can be saved for the currently selected *heater range* by clicking “ENTER”.

In addition to the front tab, the “avs48si.vi” has two other tabs for less frequently needed controls and indicators: Settings tab and Service tab.

Check that the short-circuit plug is inserted in the preamplifier’s 37-way input connector.

Before running the program, go to the Settings Tab (make LabView hand tool active) and check the serial port number. It must correspond to the physical COM port or to the port name that the USB-232 converter is using.

If this port number is correct, you need not do anything. If it is not, select the correct port, make it default (Data Operations => Make Current Value Default) and then save the “avs48si.vi” so that you need not check the port every time in the future.

Start the VI from LabView. The “STATE” box should show “Initialising” and the bridge side texts go disabled and grayed out for a moment. During initialisation, the program reads settings and other necessary data from the CPU box. Once initialisation is ready, the bridge ends up to the “IDLE” state. You can now start to measure.

A measurement is stopped by selecting the IDLE state. The EXIT button is active only in the IDLE state.

If the bridge has not yet been used, or if the CPU has just been booted, the default start-up state is channel #0, where a 100Ω reference is measured on the 300Ω range and 10mV excitation. If the bridge has already been used, the last measured channel with its settings is automatically recalled. Select now the following settings

Channel: 0 “Calibrators”  
Range: 300Ω





Excitation: 10mV  
2/4-wire: Four wire  
Sensor grounding: Floating sensor  
Autorange: disabled  
Filter length: 20  
Filter mode: SMART  
Scan enable: disabled

The middle screen display should now show a resistance that is very near to 100Ω. A green light to the right of the FILTER MODE box is blanked until the filter has received 20 readings. Soon after that it starts to blink as an indication of a valid result and that readings are taken. Stop the measurement by clicking "IDLE".

Select CH1 and re-start the measurement. This input is shorted by the protecting input plug. It should show only few milliohms. Stop the measurement.

While still on channel CH1, include it in a scan cycle by selecting "SCAN ENABLE" = enabled. Enable also autoranging. Change range to 3kΩ and excitation to 1mV. Start the measurement in order to save these new selections for CH1.

The Picobridge tries to measure the short circuit using the 3kΩ range. It gives a very low reading, and therefore the enabled autoranging changes range one step down. It is not enough, and autoranging continues until the lowest 3Ω range is reached. Then the program will happily measure the short. You can stop the measurement now with IDLE.

Select channel CH0 and click RECALL for recalling its previous settings. Include also it in the scan cycle. Start and stop a measurement. This saves the changed settings for CH0.

Now you have enabled scanning and saved good properties of two channels, which will use some different settings. Start scanning by clicking "SCAN". The scan-enabled channels are shown on the display by a slightly different color of the channel number.

The middle display will show results for both channels. A channel's reading is frozen on the display as soon as the filter has been filled with A/D conversions and the smart filter has found that all data that form the average are valid. The currently scanned channel's settings are visible on the grayed panel and they change when the channel changes.

## TEMPERATURE READOUT

Just for demonstrating temperature display, a very simple resistance-to-temperature control file (low\_cost\_pt100.txt) has been assigned to channel CH0. Of course, as there is a fixed reference resistor in CH0 instead of a real PT-100, the temperature will show about 0.0 °C. You probably wish to get rid of this non-existent sensor. It is described a little later.

Exit the "avs48si.vi", but leave it in memory. From the operating directory, find and open program "avs48si\_create\_r\_t\_information\_file.vi". Before running this multi-purpose VI, set "Processing Mode = Read and Exit", and "Save Mode = Save .cnf File". You can now start the program.

On the left side, select Input Channel = 0. You should see "low\_cost\_pt100.txt" as the R/T file's name, that the file has three columns (breakpoint number, resistance, temperature), the sensor has positive dR/dT, the file uses resistance (instead of Log(R)), and that temperature is expressed in degrees Celsius.

The breakpoints of the assigned R/T files are seen highest on the right side. Select CH0 and view the six breakpoints.

In the middle, the file information is visible in the form it was read from the file.

The lowest field shows comments that may have been written on the 9 first lines of the R/T file. The conversion data begins on the tenth line (line 9).

The program is still running. In order to **get rid** of this demo assignment, change the file name from "low\_cost\_pt100.txt" to something non-existent like "noname.txt" or just blank. Then click "ACCEPT EDITS". The assignment has been removed and the program ends.

In order to later assign a real R/T file, follow the same procedure. Place the file in the operating directory. There is no search facility in this program, so either type the file name or copy it from the computer's directory and paste it to the program. You can run the VI once again for checking that the file was found and understood correctly. See "Resistance-Temperature Conversion Files".

Start the "avs48si.vi". Temperature display for CH0 should have been disappeared.

## CONTROLLING TEMPERATURE

(only if the -T option has been installed)

The AVS-48SI temperature controller's heating current can be directed to a 100 $\Omega$  internal resistor instead of the heater in the cryostat. This resistor is in thermal contact with a forward-biased silicon diode. The diode's voltage can be used to replace the output from the actual bridge. This assembly is for superficial and fast testing of the controller.

Insert the short-circuit plug in the preamplifier input for preventing signal overload that an open input will cause. Connect a DVM to the "Control Error" BNC output. Use the settings suggested below for this test.

With the "avs48si.vi" running, go to the "Settings" tab. Select CH1 as the "Control Channel" (any other channel than 0 can be named as the control channel), and set DRDT to "SETPT-TESTV". Verify that the heater resistance has been set to 100 $\Omega$ . Return to the front tab. Select CH1 and the 300 $\Omega$  range, 10mV excitation and autoranging disabled. Start a measurement, which also makes these changes effective.

A green light should show that CH1 is the control sensor. In this test, it could be any channel, as the test does not use the bridge at all. The controller display is still disabled and grayed out, but as soon as you select heater range #1, the controller becomes enabled. In order to operate, the controller requires that

- A channel has been named as control channel
- This channel is selected for measurement
- Heater range is other than zero

Select the following parameters:

INTERNAL HEATER

SETPOINT UNIT = RESISTANCE

SETPOINT = 64 $\Omega$  (0.64V across diode)

PROPG = 1000

HTRRAN = #16

The ENTER button has turned dark grey in order to remind that there are pending changes, which need to be saved. Click ENTER. Click ENTER after any changes, but wait until it has turned dark. If it turns dark again, just click it again.

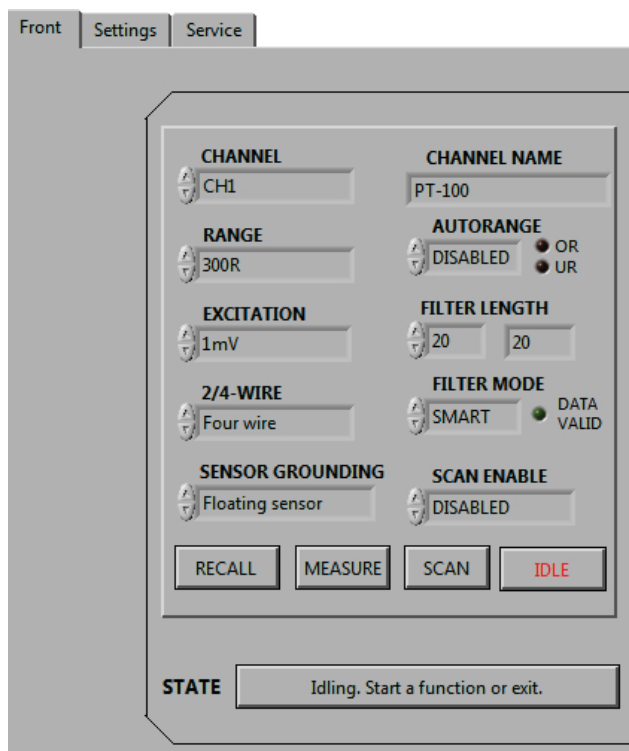
Heating starts. The ERROR panel meter should slowly approach zero without reaching it because of the control error of a P-controller.

As soon as approaching has become very slow, select INTG=0.02 and click ENTER when it has turned dark. Heating continues as the PID integrator works to reset the remaining control error. The error should end up to zero, or very near to it. The DVM connected to the Control Error BNC should fluctuate around zero and the Analog Out BNC should show 0.64V.

Stop control by nulling the PID parameters, set the heater range to zero, change back to external heater and click ENTER once prompted by its changed color.

Heating is stopped by setting Heater Range to zero. This also resets all PID parameters and keeps them at zero.

Turn off the "Control sensor" indicator light by setting the control channel to "NO CONTROL" (Settings tab).



Resistance Bridge side of the avs48si.vi -program's virtual front panel. Front tab.

## FRONT TAB

### BRIDGE SIDE

**Input channel.** Channels 1..7 are wired to the DC37S input connector and they are pin-to-pin compatible with channels 0..6 of the old AVS-47, AVS-47A and AVS-47B bridges. Channel No. 0 is reserved for measuring any one of the 7 ultra-stable calibration reference resistors. It is selected on the Settings tab.

If you suspect that you are getting incorrect readings from a sensor, select the most suitable reference while keeping the range and excitation unchanged, and switch to channel 0. If the reading is near to the true resistance of the reference, the bridge is most likely to work normally.

**Channel names.** You can give short descriptive channel names to the channels, and these names are shown on the main display. The big and bright letters make the results more convenient to read from a distance. 1) Select a channel 2) Recall its parameters, if you have already used this channel,

3) enter or edit the name 4) Start a measurement for saving the edited name together with all other settings visible on the left side display. It is not relevant, whether this channel is currently connected to a sensor or not. 5) Stop the measurement by clicking "IDLE". The new name for the selected sensor comes in effect. Channel names are saved on computer disk in file "avs48chnames.cfg".

**Range.** Ranges from 0-3Ω to 0-30 MΩ are available. Digital resolution of the display is 6 digits, which means 10μΩ on the lowest range and 100Ω on the highest range. Resolution of the readings themselves depends on range, excitation, sensor temperature, filter length and possible external disturbances. Starting a measurement saves the range setting for this channel automatically, if automatic saving of properties is enabled (default).

**Autorange.** Autorange can be enabled or disabled individually for each sensor channel. It changes range upwards, if the unfiltered reading exceeds 2.8 / 3 of the range, and downwards, if the reading falls below 0.2 / 3 of the range. Autoranging action re-starts the filter. It guarantees that each channel is measured on a valid range. Autoranging should be used routinely with the exception of the possible temperature control channel. Autoranging is extremely useful for unattended scanning on the background. Starting to measure saves autoranging setting for the selected channel automatically, if automatic saving of properties is enabled (default).

**Excitation:** The AVS-48SI features nominal excitations from 3μV to 10mV rms. Excitation is a square AC current, whose magnitude is the nominal excitation divided by one third of the range. For example, 1mV excitation on a 3 kΩ range creates a current of  $1\text{mV}/(3000\text{ ohm}/3) = 1\text{ }\mu\text{A}$ . This causes a voltage drop of 1nV across a 1 kΩ sensor. Excitation is in form of symmetrical square-wave.

The exceptional insensitivity to input capacitance is a unique feature of the AVS-48SI Picobridge. It is the result of a shape-based compensation circuit, which makes use of the square waveform. Even small passive filters are allowed in the sensor leads for preventing high frequencies from entering the cryostat. Starting to measure saves the setting automatically for the selected channel, if automatic

saving of properties is enabled (default).

**Filter length.** Excitation power must be extremely low in order not to heat a cryogenic sensor. This makes the signal-to-noise ratio poor, which is seen as large random noise in the readings. The price that normal averaging filters have to pay for their smoothing action, is their slow response. A 20-second filter takes 20 seconds to settle after a change in channel or range.

The AVS-48SI software addresses this problem with its “smart filter”, that outputs single raw readings until the filter has been filled up to its length. Then it starts to output a running average. One can get a rough estimate of the final value in seconds even if the filter is very long. The smart filter has also an algorithm for eliminating readings that are not valid because of a settling transient, or if values in the filter contents have a significant interference spike. Eight lengths from 5 to 1000 A/D conversions are available (the longest filter takes about 300 seconds to become full).

*Zero filter length:* Raw readings are output at maximum speed, which is about 3 /second. The light remains blanked. Scanning requires a non-zero filter length, because it proceeds to the next enabled channel only after a valid average has been found. See also “Saving of properties”

**Filter mode.** In addition to the default “smart” mode, the filter has an “always” mode for exceptionally noisy situations, where the data always contains some readings that render the contents invalid. The smart filter would then be continuously reset without giving any output. In the “always” mode, the filter does not make any validity checks.

*Smart filter:* The green light turns on when the average is valid and blinks at half the speed of A/D conversions. It turns blanked, if a spike or increased noise make the average invalid again and remains blanked until the disturbance has travelled out from the running average. Scanning proceeds to the next channel when the light turns on.

*Always mode:* Validity of the average is not tested. The light turns on when the filter is filled up to its length and then it starts to blink. Scanning works also in the “always” mode if the filter length

is >0. See also “Saving of properties”.

**2/4-wire.** Any sensor can be connected using either two current and two voltage leads, or using only two wires that carry both current and voltage

*Four wire* connection is the common standard and it should be preferred whenever the resistances of the sensor wires represent any significant part compared with the sensor itself.

*Two-wire* connection can be used with very high resistivity sensors or when one is desperately in short of connector space. The two wires are soldered to the appropriate current terminals of the input connector. Select “TWO WIRE” configuration for a 2-wire connected channel.

Also a 4-wire connected sensor can be measured using only the current leads. Just switch to “TWO WIRE” mode. The difference between 2- and 4-wire readings is the total resistance of the current leads. Short circuits between sensor leads inside the cryostat can change this difference. Keeping track of such differences provides a way to detect some faults without affecting the experiment. See also “Saving of properties”.

**Sensor grounding.** Any sensor can be either isolated from the cryostat (“Floating sensor”), or its current return lead (“I- lead”) may be grounded to the cryostat for improving thermal contact or for using ground as a common return, which saves wiring and connector space. The front panel control must be set in accordance with how the sensor has been wired. Floating sensor is the preferred default, because then no ground current can flow in the I-lead. See also “Saving of properties”.

**Scan enable.** Any channel can be included in or excluded from the scan sequence. Scan-enabled channels are marked by a differently coloured “CH1” etc. If one wants to save results from only one channel, it must be scan-enabled alone, so that it can be measured in scan mode. Results are saved to a disk file when an average is considered valid by the filter algorithm. A filtered average with time stamp is available after each A/D conversion from the Global Functional Variable “avs48si\_





data\_out\_array\_fg.vi” See also “Saving of properties”.

**Recall.** The “avs48si.vi” remembers the bridge settings that were used when the selected channel was measured last time. You must be in the idle state for recalling settings. Select a channel and click “RECALL”. The previously saved settings become visible, but they are not yet sent to the bridge. In other words, the bridge state is not changed by recalling a new channel with its settings from Arduino’s memory. You can now check these settings and modify them, if desired. The new settings are sent to the bridge as soon as you click “MEASURE”, and this also saves the possibly edited settings for the selected channel.

Even if you are going to scan the channel, start a measurement first for testing the settings and for saving them. Scanning will then read the already saved values. Controls on the bridge side are disabled and grayed out during recall.

**Measure.** You must be in the idle state before you can start a measurement of the selected channel. In order to prevent unnecessary heating of the sensor, the program sets excitation first to its lowest value, then all other new settings are sent to the bridge hardware, and the process is ended by setting excitation to its requested value.

The bridge makes about 3 A/D conversions per second. During measurement, you cannot change the settings. All controls on the left side display are disabled and grayed out. Stop the measurement by clicking “IDLE”. The controls become enabled. Stopping the measurement or scanning does not stop the analog bridge from working. Also the analog outputs work as before. This remains true even if you stop the VI until it is re-started.

Results cannot be saved on disk in the measure mode. However, each reading is available in a functional global variable “avs48si\_data\_out\_array\_fg.vi” for other simultaneously running Lab-View programs. Refer to “Saving the results” for more about the FG variable.

If you want results from one single channel to be saved on disk, enable this channel alone for scanning. Enable also saving and select the file type.

An on-going measurement is indicated by the blinking DATA VALID light. Blinking rate is one half of the A/D conversion rate. Also the CP, DC and DI lights on the AVS-48SI’s physical front panel blink.

**Scan.** In the idle state, click “SCAN” for starting continuous scanning. The scan-enabled and previously saved channels are measured in numerical order, each using its personal settings. Cycles start from the currently selected channel. The scan-marked rows on the main display are filled or updated as soon measurements become ready. This happens when the filter, whose length must be 5 or longer, is filled with valid data. The AVS-48SI shows resistances (and temperatures) of all scanned channels conveniently at the same time.

Enable autoranging for all scan-enabled channels with the exception of the possible temperature control sensor. Use a long filter for best results and especially for channels that require a low excitation.

Scanning speed is determined by the sum of all filter lengths plus some overhead needed to change the channel. You can make scanning slower by using longer filters. You can also include channel 0, the reference channel, in the scan sequence and select a long filter for it.

Scan results can be saved into a disk file and they are also available in the FG (functional global) variable “avs48si\_data\_out\_array\_fg.vi”. The scan sequence can include any number of channels starting from one. They need not form a contiguous set.

Continuous scanning is stopped by clicking “IDLE”. No changes are possible during scanning

**Idle.** Click “IDLE” for stopping to measure or to scan. All controls become enabled and you can change the channels and their properties.

A measurement is stopped by selecting the IDLE state. The EXIT button that stops the whole VI is active only in the IDLE state.



AVS-48SI PICOBRIDGE				
	RESISTANCE	TEMPERATURE	CHANNEL NAME	
CH0			Calibrators	
CH1	117.498 ohm	045.0977 C	PT-100	
CH2				
CH3				
CH4				
CH5				
CH6				
CH7				
<div> <div>Analog signal error</div> <div>R past conversion range</div> <div>Control sensor</div> </div>				

The bright and informative main display shows resistances, temperatures and sensor names of all eight channels conveniently at the same time. Shown are also scan-enabled sensors, the possible control sensor and some errors.

The idle mode of the software does not stop analog temperature controller from working.

## MAIN DISPLAY

**Resistance display.** The measured channel or the scanned channel(s) are shown each on its own row. Scan-enabled channels are marked by a differently coloured channel number.

Each row has also a warning light for analog signal errors. They are: signal overload and high lead resistance. The AVS-48SI combines them, but the physical front panel of the instrument shows them separately.

Signal overload might be caused by a broken V+ sensor lead or a large mains frequency interference, for example. High lead resistance might be caused by a broken sensor current lead. If a low resistance is measured using a low range and high excitation, lead resistance exceeding the specifications can reduce accuracy severely without necessarily turning on the warning light.

Please compare your wire resistance with the specifications. The total current lead resistance can be determined by setting the “2/4 WIRE” first to “Two Wire” and then to “Four Wire” and calculating the difference.

**Temperature display.** Temperature is shown only for channels that have associated R/T conversion files. The association process using “avs48si\_create\_r\_t\_information\_file.vi” and the R/T file structure are described elsewhere in this document.

Each row has a red warning light. It turns on, if the measured resistance is below the lowest or above the highest breakpoint in the R/T file. Then the minimum or maximum temperature of the calibration range is displayed and the temperature reading is meaningless. Use the resistance reading in such a case.

**Channel name.** Short descriptive channel names make the display easier to read from a distance. You give or edit the names in the “CHANNEL NAME” control box. The new name becomes visible once LabView concludes that you are done. It saved when you start a measurement.

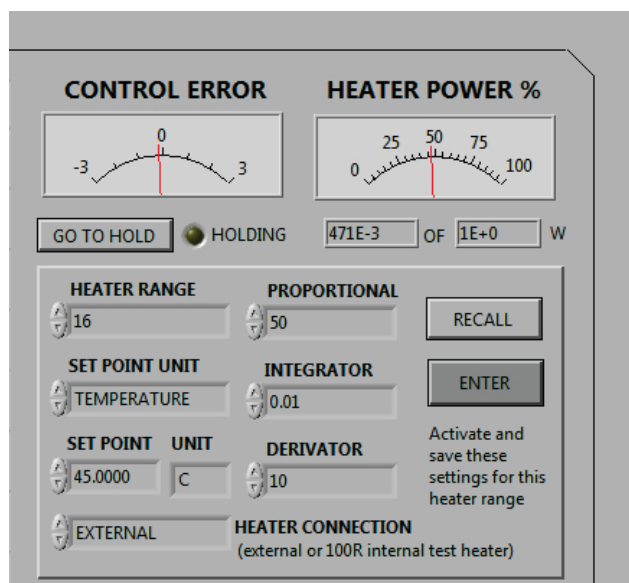
A green light shows the channel that has possibly been defined as the temperature control channel.

## TEMPERATURE CONTROLLER

The heater output *power* of the AVS-48SI, instead of *current*, is a linear function of the error signal.

**Control error meter.** The scale of this animated analog meter has been nonlinearized strongly. It shows the trend of the error signal even if it is large, but the meter is very sensitive near to balance. Sensitivity near zero is about 100µV while the error signal’s maximum value is +/-3V (≈ 90dB dynamic range).

Positive deflection means that more power is needed in order to achieve the set point temperature. In a way, this corresponds to the nature of the exponentially settling control temperature.



Display of the avs48si.vi program's temperature controller side. Front tab.

**Heater power % meter.** This animated panel meter shows heating power as a percentage of the selected range's maximum. The maximum is calculated on basis of the given heater resistance, which can be set on the "Settings" tab. Edit it, if needed, and click "ENTER" on the controller side of the front panel in order to save the new value. The heater resistor's value  $R_H$  can be saved only when temperature control is enabled by clicking ENTER when temperature control is enabled.

**Heater power display (digital).** The rightmost field under the power meter shows the selected heater range's calculated maximum power. It assumes 12.4V compliance voltage for  $R_H$  higher than 100 $\Omega$  and calculated maximum currents for  $R_H$  lower than 100 $\Omega$ . The maximum heating power that the AVS-48SI can deliver is about 3W into a 50 $\Omega$  heater.

The leftmost indicator box shows the actual heating power in Watts. It is the product of measured true heater voltage and current. The voltage is very low on the lowest heater ranges, which makes the product then somewhat unstable.

The true heater power indicates both shorts and open circuits in the heater wires: The  $V \cdot I$  product will be zero if a short nulls the voltage drop, or if an open circuit prevents current from flowing.

**Heater range.** The AVS-48SI has 18 power ranges,

the power ratio between ranges is  $10^{(1/5)}=1.58$ . Closely spaced ranges make it easier to keep heating in the middle region of the range.

The AVS-48SI features changing the heater range without appreciable effect on temperature provided, that the system was in equilibrium before the change. For reducing the range:

- 1) Change the heater range downwards
- 2) Watch the power display or the %-meter. If they show that the intended lower range can give enough power for maintaining the control temperature, then
- 3) click "ENTER", otherwise return back to the original range and quit this experiment by clicking ENTER.

You can always change upwards.

If you increase power range, the loop gain is increased, and you may need to reduce the proportional gain in order to maintain system stability. If you reduce heater range, the gain of the control loop is reduced and you may need to increase proportional gain in order to maintain settling speed.

Temperature control is enabled, if the following three provisions are fulfilled:

- 1) Measurement channel is greater than zero,
- 2) the control channel is the same as the measurement channel and
- 3) heater range is greater than zero. Then the controls become visible and enabled. The power range and all its personal settings become in effect by clicking "ENTER".

In a sensitive application, when accidental high power can cause damage, the maximum power can be limited by disabling the highest ranges. *EXIT the program before the following procedure:*

- 1) click the "HEATER RANGE" control box with the pointing tool,
- 2) from the context menu, select "Edit Items..",
- 3) select range #18 and click "Disable Item",
- 4) continue disabling all ranges that you consider too high,
- 5) save the "avs48si.vi". The disabled ranges can be enabled back by a reverse procedure.

Heating can be stopped quickly by setting heater range to "CONTROL OFF". This makes all settings zero and disables control and the heater. As

an exception to the general rule, ENTER is not needed in this case. See also “Heater connection”.

**Proportional gain.** There are 13 logarithmically spaced proportional gains in 1:2:5 etc. sequence from 1 to 10000. If proportional gain is set to 0 when the system is in equilibrium, the heating power is solely determined by the voltage that has accumulated in the PID controller’s integrator.

In many cases, you can set the proportional gain to a useful value as follows:

- 1) Set the integrator and derivator gains to zero,
- 2) set the proportional gain to maximum value that does not yet cause overshoot when the set point is changed up and down.

Different temperatures are best controlled on different power ranges. Therefore each power range can have its own PID parameters which can be RECALLED for this range. If parameters are not recalled, they remain the same for all ranges, just as on the resistance bridge side. See also “Saving of properties”.

**Integrator gain.** The AVS-48SI uses the expression “*integrator gain*” instead of the more common “*integration time constant*”. The concept of gain is more convenient to use: When you increase integrator gain, you increase the integrating effect, i.e. you make integration faster. There are ten selectable logarithmically spaced gains from 0.001 to 1 in a 1:2:5 etc. sequence.

Too little integration makes the system slow, whereas too much integration leads to unacceptable overshoot or instability of the control system. A suitable integrator gain can add a little overshoot to the proportional-only mode. Zero integrator gain (“None”) shorts the integrator to zero.

Do not reset the integrator by mistake. You may be able to speed up balancing after a change in set point by setting the integrator gain to 0.001 until the P (or PD) controller is almost settled. Then enable the original integrator gain. See also “Saving of properties”.

**Derivator gain.** There are 10 selectable derivator gains in a 1:2:5 etc. sequence. The higher is the gain, the more significant is the derivation action. The derivator has a modified response, which prevents it from amplifying higher frequencies. Because of this modified response, the derivator gain has a maximum, which shifts lower when INTG is increased.

Derivation is the most difficult of the three PID parameters to adjust. A suitable amount of derivation can make control faster so, that it can better react to temperature changes or to a slope, whereas too much derivation leads to excessive overshoot, or it makes the system unstable. Too weak derivation is of no use at all. A lot of temperature-dependent trial-and-error work may be needed, which means that derivator is most useful for saving time in a repeated operation, where the extra setup work is justified. See also “Saving of properties”

**Set point unit.** Set point is given in Ohms by default.

Set point can also be given in temperature, if an R/T conversion file has been associated with the control sensor. Then the main display shows both the sensor’s resistance and temperature. Depending on the conversion file, temperatures are given in Kelvins or degrees Celsius. Select the appropriate SET POINT UNIT, enter the desired resistance or temperature. For making the new set point effective and click “ENTER” when it turns darker.

If the sensor has an R/T conversion file, you are likely to give the set point in temperature. However, it may happen that a desired set point is outside the calibration range. Then the SET POINT control box turns red and the resulting set point defaults to the minimum or maximum of the calibration range.

In order to reach the desired far control point, change over from temperature to resistance. Change the SET POINT UNIT from temperature to resistance and the SET POINT indicator box starts to show the set point in ohms. Click ENTER when it turns dark gray. Now also the SCALE box shows Ohms. The control set point has not changed so far. After this, you can edit the set point, if desired, and click ENTER again. You can change the set point from resistance to temperature in the same way.

**Set point.** Set point is given as a floating point number like 115.05 or in exponential form like 1.1505E2 or 1.1502e2. It is internally converted to a voltage, based on the selected resistance measuring range. If the set point is given in temperature, the conversion accuracy depends on the number and locations of the breakpoints. The R/T and T/R conversions (for set point) are based on linear interpolation.

Exponential form is allowed only when using LabView. Give set points in floating point form when using serial commands.

**Scale.** This indicator shows the temperature scale (“K” or “C”) that is used by the control sensor’s R/T conversion file. If no conversion file exists for the control sensor, Scale defaults to “Ohm”. Refer also to discussion about the R/T files.

**Heater connection.** The AVS-48SI has an internal 100Ω heater, which is rated for 2 Watts. This resistor is in thermal contact with a silicon diode, whose forward voltage changes about -2.5mV/C. The “INTERNAL 100R” position can be used for simplified operational test of the temperature controller. The default position of this control is “EXTERNAL”.

Previously was described a method for stopping control quickly: Set the heater range to “CONTROL OFF” (refer to “Heater range”). This will null all control settings. Restoring control after that requires recalling or re-programming the original settings and waiting for the system to reach balance.

Output power can also be disconnected temporarily from the heater without nulling any of the control settings. This results in faster return to balance. Because there is no active control, the following method can be useful, if only a short pause in control is sufficient. The system should be at equilibrium for best results. Do as follows:

- 1) Select “INTERNAL 100R” heater connection,
- 2) click “GO TO HOLD”,
- 3) once the yellow light shows that the system is in hold state, do whatever you need the pause for,
- 4) select “EXTERNAL” heater connection

5) click “EXIT HOLD”.

*This operation must be done without using “ENTER”*

If you do not want to null the heating power, but need to e.g. disconnect the sensor cable for a moment, go to the HOLD mode for a while (see “Hold”).

**Recall.** Each heater range has its own set of the three PID parameters: proportional gain, integrator gain and derivator gain. They are made active but also saved automatically for the currently selected heater range when you click ENTER. Settings other than these three on the controller side are also saved, but they are common to all heater ranges.

Select a new heater range and click RECALL. This brings visible the settings that were previously used with this new range and which were saved for it. You have now the opportunity to recall another heater range or modify the settings before you make them active by clicking ENTER.

The philosophy is very similar to that on the bridge side. On both sides one is able to see the recalled settings, possibly change them, recall another range, or recall the original heater range before activating the settings. On the bridge side, the MEASURE and SCAN modes are started from the IDLE state, whereas on the controller side, new settings are made active by clicking “ENTER”.

**Enter.** Changes made to the controller settings are activated by clicking ENTER. Many changes take time because of required two-step D/A conversions, and you should not click ENTER until a command has been fully handled. In order to avoid that, ENTER button turns dark grey when it is ready to be clicked. The new colour also reminds you, if you have made changes or just looked at the alternatives, that you need to activate them. If you did not do changes but returned back to old settings, clicking ENTER does not alter anything, it only changes its color back. Sometimes a change is left pending after your first press of ENTER and the button turns back to dark grey. Just click ENTER again.

Even if ENTER has turned dark grey, you can continue making changes, it is not necessary to activate all changes separately as they are done. All



changes can be entered together.

There are two exceptions to the above rule: zero HEATER RANGE and GO TO HOLD:

- If heater range is set to CONTROL OFF, control is stopped immediately, all settings are nulled and the heater is disabled. This is for enabling rapid shutdown.
- GO TO HOLD and EXIT HOLD also *work* without entering. However, entering will be needed for restoring the light gray color of “ENTER” (Hold has required internal changes).

**Hold.** The hold state is mainly for measuring sensors other than the control sensor without completely stopping temperature control. Once “GO TO HOLD” has been clicked, the program measures the PID integrator’s output, which is then replaced by the output of a digital-to-analog converter. The heating output remains constant in the hold state but there is no active control. Holding works best, when the PID system is in good equilibrium. The magnitude of the unavoidable temperature drift depends on initial balance and on the stability of the heat load.

While holding, you can visit any sensor and you can even scan the enabled sensors. The scan results are saved on disk, if saving is enabled, and they are always available in the global functional variable for other LabView programs. After your measurements or scanning, BE SURE to select the control channel again and start measurement with MEASURE. After the bridge has settled, click EXIT HOLD.

You can change the *measuring range* or *excitation* of the *control channel* during temperature control. Do as follows:

- 1) Switch the bridge side to IDLE,
- 2) select GO TO HOLD,
- 3) once holding, select the new measuring range or excitation,
- 4) click MEASURE
- 5) when the measured reading has been reasonably settled, click EXIT HOLD.

This works regardless of whether the setpoint is in resistance or temperature. If the system was in equilibrium, the effect in controlled temperature would be minimal. Please note, that lowering

measuring range by one step will increase the loop gain by a decade. Increasing range will reduce the gain by a decade. You may need to decrease or increase the proportional gain for maintaining system stability or for maintaining its response speed, respectively. Control accuracy will be better, if you do not control very near to the low end of the measuring range. A special feature of the AVS-48SI is that the heating power, not current, is a linear function of the control error. This reduces the need to adjust proportional gain when power level changes.

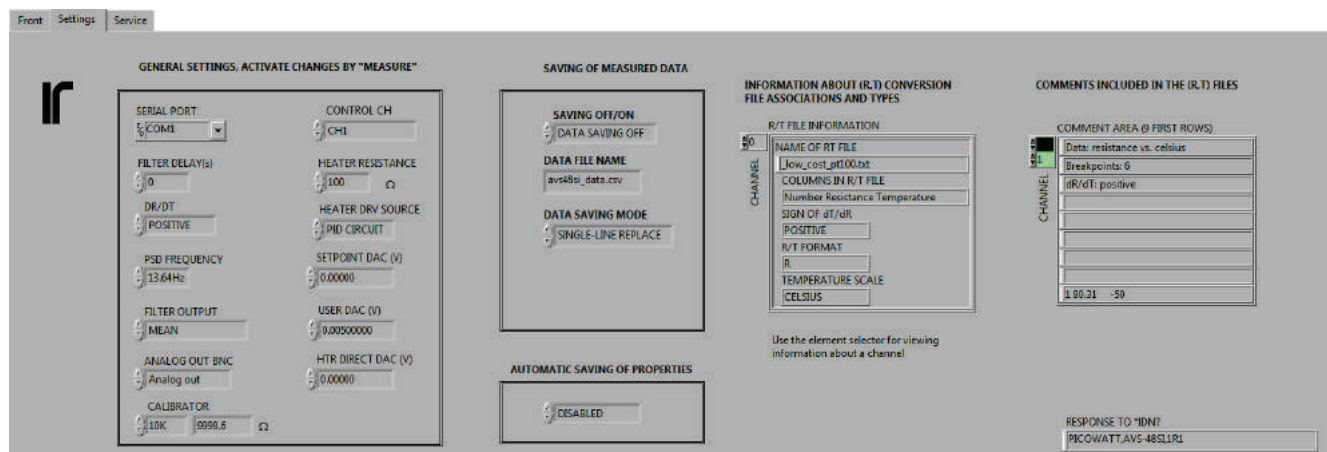
All controls on the right controller side are dimmed and disabled when the system is holding.

**Exit.** Use EXIT for stopping the “avs48si.vi” program, do not abort it from LabView if not necessary. The “legal” stopping closes the serial port so that it can be used by other programs.

For security, the program can be exited only when it is doing nothing. It is only then that the EXIT button is enabled, and this happens only if:

- The bridge has been set into IDLE state
- The controller has been disabled by setting Heater Range to CONTROL OFF for stopping the controller.

If the “avs48si.vi” is aborted, or if remote control stops in some other way, the analog bridge and controller continue working. Re-starting the VI does not alter operation of the bridge, whereas the controller is reset and it must be started from its default no-control state. This is for safety.



## SETTINGS TAB

The Settings tab can be accessed regardless of the selected LabView tool when the “avs48si.vi” program is running. When it is not running, use the hand tool for selecting a tab. The same applies to the Service tab.

**Serial port.** This control shows the serial ports that are available. Select the one that is or will be connected to the external CPU box of the AVS-48SI. Serial port is one of the few settings that require saving the “avs48si.vi”. Once saved, you do not need to adjust it every time the program is started.

If your computer does not offer a physical RS232 (or Com: ) port, you need a USB-232 converter for creating such a port. If the converter is compatible with your platform, the name of the virtual Com: port that it has created should be listed in the “SERIAL PORT” control box. Picowatt offers a converter from National Instruments (part number 778472-01), but working converters of other brands may be available from elsewhere at a lower cost. The Arduino box expects two-wire RS232 communication without handshaking with serial parameters 9600,8,N,1 (baud rate 9600, 8 data bits, no parity and 1 stop bit). This is believed to provide the best possible compatibility, because it is the most common default of various programs and platforms.

**Filter delay.** If the filter is used in “ALWAYS” mode, it calculates a running average without checking the validity of the readings in any way. The average can therefore contain transients after a change in range or channel. “FILTER DELAY” in-

serts a fixed delay between such changes and starting of the filter. On the highest excitation ranges, some 5 seconds is enough, whereas on the lowest excitation a delay of 15 seconds could be suitable. Filter delay is not used in the SMART mode.

The filter does not react to sudden changes in the *sensor value* in the ALWAYS mode, whereas in the SMART mode it does.

**DR/DT.** The instrument’s control polarity must correspond to the sign of the control sensor’s slope dR/dT. Most cryogenic sensors have negative slopes, while sensors for higher temperatures, like Platinum wire resistors, have positive temperature coefficients. Clicking “MEASURE” saves the selected polarity.

If temperature control is NOT needed, you can use the set point voltage (see below) as a reference for the deviation output. Use DR/DT to determine the polarity of the difference between measured signal and the set point DAC voltage: Positive polarity means (signal - set point) and negative polarity means (set point - signal). This deviation is available at a rear panel “CONTROL ERROR” connector and it can be displayed on the front panel as an alternative to the normal resistance display. See below.

**Analog out BNC.** Normally this rear panel BNC connector outputs a voltage proportional to the measured resistance as

$$V_{\text{OUT}} = R_{\text{SENSOR}} / \text{RANGE} * 3V.$$

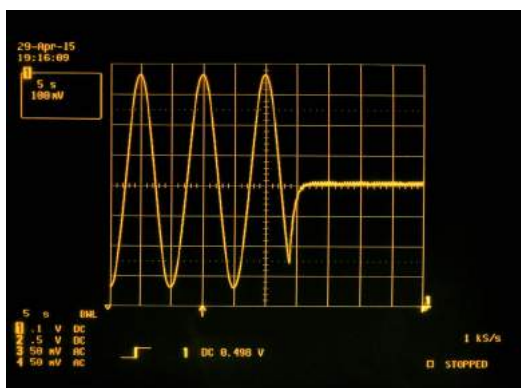
The calibration system of the AVS-48SI considers

this output as the primary quantity to calibrate and therefore the accuracy of the analog output is very good. In order to suppress residues of the carrier frequency, the analog voltage from the phase-sensitive detector (PSD) is smoothed by a 3<sup>RD</sup> order Bessel filter of about 1.2Hz 3dB frequency before outputting.

If the “ANALOG OUTPUT BNC” rear panel control is changed from “Analog out” to “Error out”, the rear panel connector shows the difference between the analog output and the “SETPOINT DAC (V)” control. This selector is available on the settings tab but do not use it if temperature control is active. It will overwrite the setpoint coming from the controller section. Deviation voltage is seen also on the front panel, where the display title changes to “DEVIATION (dR)”.

**PSD Frequency.** Select one of three crystal-controlled operating frequencies 12.5Hz, 13.64Hz (default) or 15.0Hz. The default frequency is usually suitable both in 50 and 60Hz countries.

Electric motors can generate mechanical or acoustic vibrations containing frequency components that are near to but not exact (sub-)multiples of the mains frequency. Such components, if they produce voltages in the input circuit, can cause very slow beating especially if the bridge uses 12.5 or 15Hz excitation. Then selecting another operating frequency can reduce beating and make it faster and easier to filter off. Even an arbitrary frequency is possible by feeding an external signal to the



12.4Hz interfering signal of 200μVpp was induced to a 10Ω sensor measured with  $V_{EXC}=300\mu V$  and  $f_{BRIDGE}=12.5Hz$ . The interference beats with the excitation at  $f_{BRIDGE}-f_{EXC}=0.1Hz$ . The beating amplitude of 700mVpp (analog output) went down to 12mVpp when the bridge frequency was changed to 15Hz. A 60-fold improvement!

“EXT CLOCK” input.

**Filter output.** The smart filter maintains a running average of the indicated number (= filter length) of A/D conversion results. In addition, the filter

- fits a least squares line to the data points in the filter
- calculates the average, and standard error of data in the filter
- calculates the Mean Squared Error (MSE)
- outputs the last point of the fitted line

MSE is the sum of squared distances of the data points from the least squares line, normalized by dividing the sum by the number of data points. Because of its quadratic nature, MSE is very sensitive to the curvature of the initial transient part of data. In addition, it grows rapidly, if the data in the filter contains a peak due to some interference or if the data changes rapidly. This behaviour is used for excluding invalid data from the results.

The filter is re-started by many changes in the system in addition to signal curvature or peaks. After re-start, new conversion results are collected into the filter. As long as the filter is not full, it outputs raw non-filtered readings as such, not their average. They approach the final value much sooner than the average, but of course they contain noise that is later to be smoothed by the filter. The purpose is, that the user gets a rapid estimate of the final value.

As soon as the filter is full, fitting a line to the data starts and MSE is calculated after each new A/D conversion. The MSE is compared with limits that have been tabulated for range and excitation, and if it is smaller than the limit, the average is considered valid. This is shown by a green indicator light, and the display changes to show average. If the raw data has been noisy, you see a sudden decrease in noise level.

“MEAN” is the normal output. In addition to the mean, you can see the standard deviation of the data (on Service tab). It becomes visible once data in the filter has been considered valid. You can compare this STD with our room-temperature specifications. If the STD is much larger even though your sensor is cooled, you have an interference problem. Use a long filter in order to make

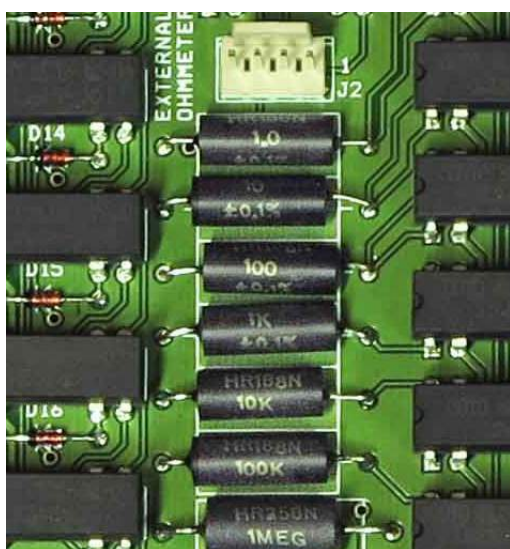


STD reliable.

“LAST POINT” means the most recent point of the fitted line. In a case where temperature is drifting rapidly, a normal average will lag in time. For example, a filter with  $L=100$  (about 30 seconds), the average will lag by 15 seconds. The “last point” output will be closer to the instantaneous temperature than the average. As far as noise is considered, this output is inferior to the normal average. Use this mode only for drifting data.

**Calibrator.** Seven ultra-stable wire-wound resistors located in the preamplifier unit serve as calibration standards of the AVS-48SI. They have excellent temperature ( $3\text{-}5\text{ppm}/^\circ\text{C}$ ) and long-term stability ( $\approx 30\text{ppm}/\text{year}$  during first years, later usually even less), but their initial tolerance is only 0.1%. Therefore the true values of these resistors are measured and saved into the non-volatile EEPROM memory of the Arduino CPU card. This is a part of the initial calibration. New calibrator values are easily entered into memory if the preamplifier is changed, or after some repairs to the preamplifier or main unit. The original calibrator values are written on a sticker inside the bottom cover of the preamplifier. This is a great improvement compared with the older AVS-47x bridges, which required external standards for restoring its calibration.

Sensor channel #0 is devoted to measuring these



The seven wirewound calibrator references are located on the multiplexer board in the preamplifier unit. The 4-wire connector makes it possible to measure these resistors using an external high-end ohmmeter.

standards. In the “CALIBRATOR” control box, select one of the resistances 0,  $1\Omega$ ,  $10\Omega$ ,  $100\Omega$ ,  $1k\Omega$ ,  $10k\Omega$ ,  $100k\Omega$ , or  $1M\Omega$ . Its true value is shown immediately. Return to front panel, select channel CH0, and the interesting range and excitation. Start a measurement for activating these changes. If the display differs from the indicated true value by more than few hundredths of a percent, you should consider running the automatic Full Main Calibration program (see Main Calibration, Calibrating the A/D and D/A Converters and Reference Resistor Calibration).

Despite its special nature, CH0 can be included in a scan cycle. This provides some degree of quality control. It can also be used for increasing the total scan cycle time, if slower scanning is desired.

**Control CH.** Temperature control requires that one of channels 1-7 is assigned as the control channel. For clarity, the selected channel is shown also on the front panel by a green LED. The controller section becomes enabled when 1) the control channel has been selected, 2) the measuring channel is the same as the control channel and 3) heater range is not zero. Autoranging should be disabled for the control channel.

If temperature control is not active, the control sensor can be measured or scanned like any other sensors. If you change the control channel during control (Settings tab), control will be stopped and all parameters will be nulled.

**Heater resistance.** Numeric value of the heater resistance is needed only for calculating the maximum power on each heater range. A typical resistance “standard” resistance is  $100\Omega$ , but the current source heater output of the AVS-48SI accepts any value. The maximum power of 3W is achieved when heater is  $50\Omega$ . If  $R_H$  is lower than  $50\Omega$ , the available current will limit the maximum power. If  $R_H$  is higher than  $50\Omega$ , the maximum output compliance voltage will limit the power.

**Setpoint DAC (V) and User DAC (V).** These two digital-to-analog converter outputs (from LTC2600) give very accurately calibrated voltages in the region  $0.005\text{...}2.99$  Volts. The outputs can supply over 10mA, but it is recommended to buffer them externally, if more than a few milliamps are needed. The outputs are not protected, so please be



careful not to supply energy into them.

The user DAC is always available for your own applications, whereas the setpoint DAC is needed for temperature control when it is active. Voltages of these two DACs can be changed without stopping the possibly on-going measurement. Just enter the new value and click an empty place on the panel or hit enter of your computer keyboard.

The set point should be given in resistance or temperature from the SET POINT box when temperature control is active. If not controlling, or if no TC option is installed, give the set point in voltage from the Setpoint DAC (V) box on the Settings tab. The user DAC voltage is always given as a voltage on the Settings tab.

Do not use the Setpoint DAC(V) input while controlling. This input will override the set point coming from the controller.

#### **Heater Drv Source and Heater Direct DAC (V).**

In addition to normal use, where the PID circuit drives the heater output stage, the AVS-48SI features alternative direct numeric control of the output current. Under the “HEATER DRV SOURCE” select “DIRECT DAC”, goto the front panel and click “ENTER” after it has turned dark. Now the heater output can be considered a voltage-controlled current source. Use the “HTR DIRECT DAC (V)” control box for setting the output current.

This voltage determines the output current, not the power. Three volts corresponds to the maximum current of the selected power range. The DAC does not go to exact zero, there will be a small fraction of the maximum current left even if the DAC is set to zero. Above 0.005 Volts, response is linear. Select the “CONTROL OFF” heater range for disabling power output completely.

“ENTER” is required after changing the driver source, but not for changing the DAC voltage.

**Saving off/on.** The “avs48si.vi” can save measurement data on a disk file. Saving is available only in scanning mode, but because even one single

channel can be measured in scan mode, this is not a limitation. Saving is always off by default when this program is started. The purpose is to prevent the data file from growing infinitely without anybody noticing.

Regardless of whether saving is enabled or not, measurement data is always available to other running LabView programs via a functional global variable (FG).

**Data file name.** Data is saved as comma-separated values (CSV). The default name is “avs48si\_data.csv”, which is always offered as default when this program is started. You can modify the default name only temporarily for your current session.

**Data saving mode.** In addition to offering data to other simultaneously running VIs, the “avs48si.vi” can save data to disk in one of two selectable modes.

*To a disk file:*

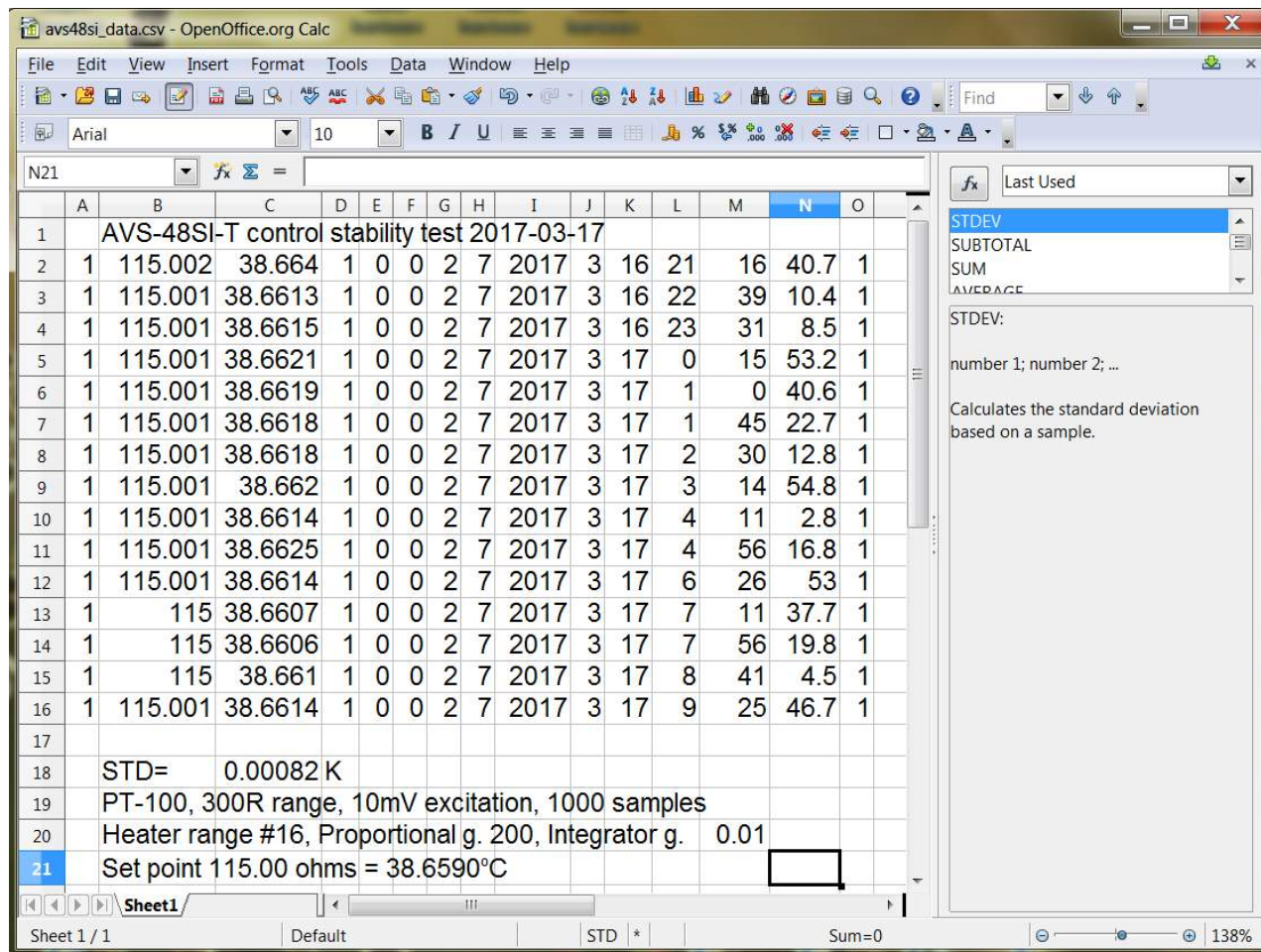
If “Single-line replace” is selected, the file will consist of only one row that contains data from the latest scanned channel. After a channel has been successfully scanned and new data is available, the existing file is re-written. This file type is intended to be used by some external program, which continuously polls the existence of “avs48si\_data.csv” and once it appears, reads the single line and then deletes the file.

If “Multi-line append” is selected, every completed scan measurement of a channel adds a new line to the file, which is never deleted by the “avs48si.vi”. This mode is mainly intended for collecting data that is to be analyzed later.

*For other LabView VIs:*

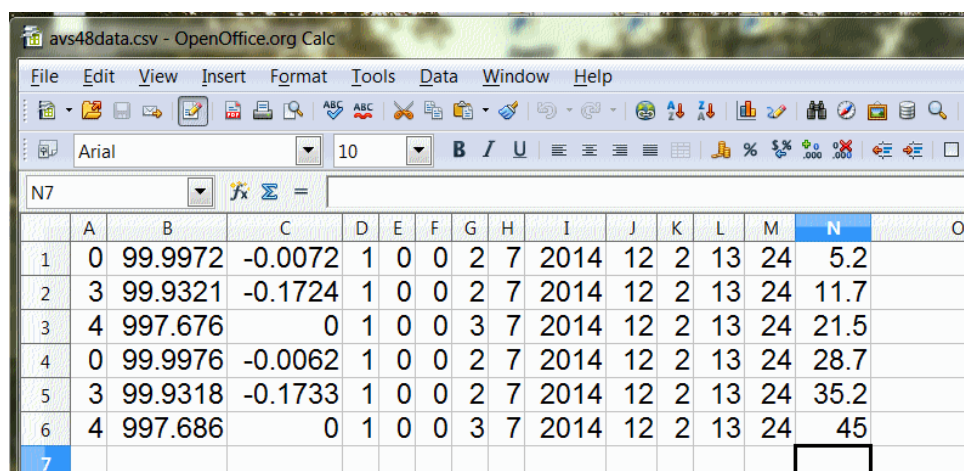
In addition to saving data to disk, the “avs48si.vi” offers data to other simultaneously running VIs by means of a “Functional Global” variable “avs48si\_data\_out\_array\_fg.vi”. This variable is a VI that consists basically from two non-initialized shift registers. They have fixed locations in the computer memory and they can be accessed for writing and reading by any running VI.

Data can be read from this FG in two forms: 1) a CSV string containing the result of the last com-



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1		AVS-48SI-T control stability test 2017-03-17													
2	1	115.002	38.664	1	0	0	2	7	2017	3	16	21	16	40.7	1
3	1	115.001	38.6613	1	0	0	2	7	2017	3	16	22	39	10.4	1
4	1	115.001	38.6615	1	0	0	2	7	2017	3	16	23	31	8.5	1
5	1	115.001	38.6621	1	0	0	2	7	2017	3	17	0	15	53.2	1
6	1	115.001	38.6619	1	0	0	2	7	2017	3	17	1	0	40.6	1
7	1	115.001	38.6618	1	0	0	2	7	2017	3	17	1	45	22.7	1
8	1	115.001	38.6618	1	0	0	2	7	2017	3	17	2	30	12.8	1
9	1	115.001	38.662	1	0	0	2	7	2017	3	17	3	14	54.8	1
10	1	115.001	38.6614	1	0	0	2	7	2017	3	17	4	11	2.8	1
11	1	115.001	38.6625	1	0	0	2	7	2017	3	17	4	56	16.8	1
12	1	115.001	38.6614	1	0	0	2	7	2017	3	17	6	26	53	1
13	1	115	38.6607	1	0	0	2	7	2017	3	17	7	11	37.7	1
14	1	115	38.6606	1	0	0	2	7	2017	3	17	7	56	19.8	1
15	1	115	38.661	1	0	0	2	7	2017	3	17	8	41	4.5	1
16	1	115.001	38.6614	1	0	0	2	7	2017	3	17	9	25	46.7	1
17															
18		STD= 0.00082 K													
19		PT-100, 300R range, 10mV excitation, 1000 samples													
20		Heater range #16, Proportional g. 200, Integrator g. 0.01													
21		Set point 115.00 ohms = 38.6590°C													

CSV data from a temperature control test. An assembly of a 100Ω heater and a PT-100 sensor was inserted in an empty bottle for eliminating air flows at the room temperature. Channel #1 was measured in the SCAN mode, saving was enabled and save mode was multi-line append. Each reading is an average of 1000 A/D conversions. Standard deviation of temperatures was 0.001°C. The columns are from left to right: Channel No., Resistance, Temperature, Temperature Unit, Signal Error, R past calibr. range, Range, Excitation, Year, Month, Day, Hour, Minute, Fractional Seconds, Data Valid.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	0	99.9972	-0.0072	1	0	0	2	7	2014	12	2	13	24	5.2	
2	3	99.9321	-0.1724	1	0	0	2	7	2014	12	2	13	24	11.7	
3	4	997.676	0	1	0	0	3	7	2014	12	2	13	24	21.5	
4	0	99.9976	-0.0062	1	0	0	2	7	2014	12	2	13	24	28.7	
5	3	99.9318	-0.1733	1	0	0	2	7	2014	12	2	13	24	35.2	
6	4	997.686	0	1	0	0	3	7	2014	12	2	13	24	45	
7															

Here channels 0, 3 and 4 are scanned. All channels have fixed 100Ω resistors, so the temperature reading has no meaning in this photo. This is an older picture where the Data Valid column was still missing.

pleted measurement of scan measurement, or 2) an 8-element array of clusters. In the latter case, each element corresponds to one of the 8 channels. When sensors are measured or scanned, only the corresponding elements contains data, other elements have empty clusters.

Often it is easiest to use the cluster output. One needs only to “Unbundle by Name” the desired items. If the AVS-48SI is scanning and one reads the FG only occasionally, this arrangement guarantees, that the information is as up-to-date as it can be. The FG can be re-initiaized after it has been read. This eliminates the possibility that some data is read multiple times.

Reading every complete scan cycle as soon as it has been completed is more complicated. One solution is to poll the CSV string's first character (=channel number). When it equals to the last channel in the cycle, it is time to read the output array and re-initialize the FG.

Data is saved on disk only if saving is enabled, whereas data is written to the FG always.

FG STRINGS, ONE ELEMENT		
CHANNEL	Integer 0..7	
RESISTANCE	Real	
TEMPERATURE	Real	
TEMP UNIT K OR C	Integer 0=K, 1=C	
SIGNAL ERROR	Integer 0=no error, 1=OVL or LRES	
R PAST CAL RANGE	Integer 0=In range, 1=R or T past cal range	
RANGE	Integer 0..7 (3R..30M)	
EXCITATION	Integer 0..8 (3uV..10mV)	
YEAR	Integer	
MONTH	Integer	
DAY	Integer	
HOUR	Integer	
MINUTE	Integer	
FRAC SECOND	Real	
DATA VALID	Integer 0=not valid,, 1=data valid	

This is one of the eight elements in the output array of the “avs48si\_data\_out\_array\_fg.vi”. Every element is a cluster of 15 items. In this figure, the complete cluster has been unbundled.

**R/T file information.** You can inspect the types of the R/T conversion files that have been associated with various input channels. Change the array index selector (=channel number) for seeing the following data about the conversion file that was

associated with this channel:

- The file name. Conversion files are non-formatted plain text files with a .txt extension
- The file may have two or three columns: a) resistance and temperature, or b) number of break-point, resistance and temperature.
- Sign of dR/dT
- R/T format. The resistance may be expressed as resistance or log(R). The latter is common with cryogenic sensors that are highly nonlinear within their operating range.
- Temperature scale in the file can be either Celsius or Kelvins.

Refer to description on how to create and assign R/T files.

**Comment area (9 first rows).** Nine first rows of each R/T conversion file are available for free-form comments, like sensor type, date, serial number and number of breakpoints etc. The actual conversion data begins from the tenth row.

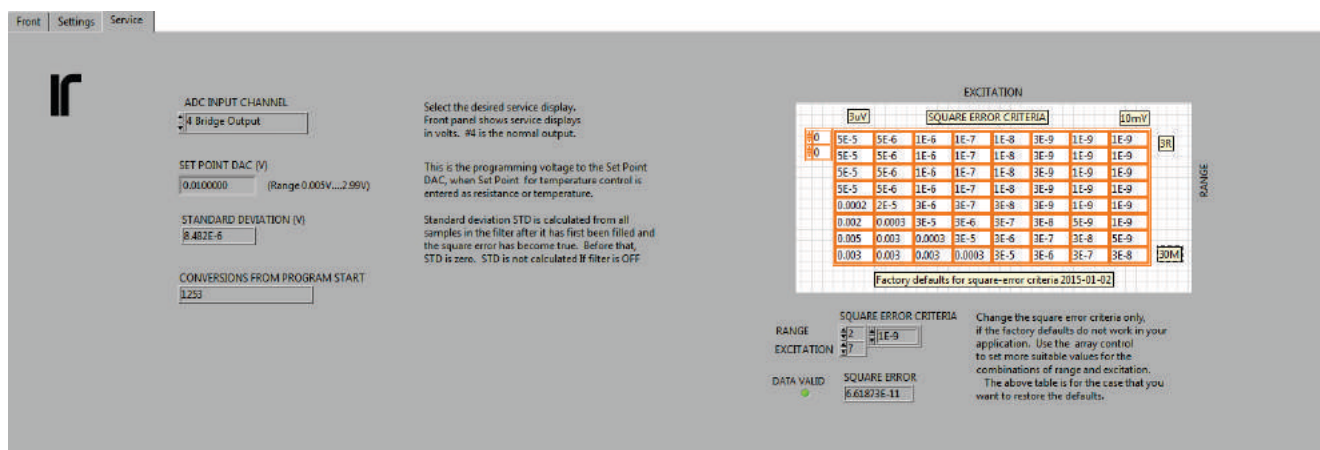
Refer to description on how to create and assign R/T files.

**Response to \*IDN?.** Here you can check the firmware version of the microprocessor program that runs on the Arduino Mega2560 microprocessor module. The firmware version of the “avs48si.vi” LabView program is also in the name of the directory that contains the necessary VIs.

**Automatic Saving of Properties.** If enabled (default), the avs48si.vi will save the currently visible properties of the selected channel each time “MEASURE” is clicked. When this same channel is measured next time, its previous settings can be conveniently recalled. The same applies to PID parameters of the temperature controller - they are saved for each heater power range for later recalling when “ENTER” is clicked.

Automatic saving can also be disabled. Then no properties or PID parameters are saved by pressing “MEASURE” or “ENTER”. Settings that were saved before disabling automatic saving are available for recalling. This can eliminate the possibility of altering the properties by mistake.





The “avs48si.vi” service panel provides access to various service measurements and to modifying the smart filter’s Mean Square Error criteria.

## SERVICE TAB

The service tab contains some information that you will not need often.

**Adc input channel.** The analog-to-digital converter of the AVS-48SI can be connected to 26 various internal voltages, and most of them are for production testing. They also help in remote diagnosis, as many internal voltages can be measured without opening the instrument.

Input #4 is the default - it is the bridge output. All other inputs change the “RESISTANCE” title on the front panel to “SERVICE” and the unit changes to Volts. Some other interesting inputs are listed below. The channel, where service voltages are shown, depends on channel that happens to be selected. Filtering is usually not needed with service displays.

- 0 = Ground. The display should be very near to 0, only few tens of  $\mu\text{V}$ .
- 1 = ADC reference. The voltage should be about 3 Volts. Its exact magnitude is not very important.
- 16 = PID heater voltage and #17 PID heater current. These are not shown on the temperature controller's front panel. However, they are measured and the true heating power is calculated as their product.
- 22 = Set point DAC or #24 User DAC. These voltages are measured from the DAC outputs by the A/D converter. Their values should be same as the programmed values  $\pm 100\mu\text{V}$ .

After having selected a display item, stop and restart measurement in order to make the selection effective.

**Standard deviation (V).** Standard deviation is calculated from readings in the filter, which must be filled up to its length by valid data. There is a green light on the service tab that shows when the filter contents is valid.

If filter length is short, like tens of readings, STD will fluctuate a lot. Use a long filter for getting stable readings. STD may help in finding some problems. Take readings both from your sensor and from the appropriate reference resistor. Compare these with the AVS-48SI specifications. If your cooled sensor gives much higher STD readings than those from a room-temperature reference, you may have an interference problem. But if the STD from the reference is much higher than specified, something with the bridge may be out of order.

**Square error table.** The smart filter compares the calculated and normalized mean squared error (MSE) with tabulated criteria. The table shows the factory default criteria for all combinations of range and excitation. They have been determined so, that a valid average can be found even when measuring a 10M $\Omega$  room-temperature resistor (except on the lowest excitations). This table serves as a reference if you have changed the criteria but then want to restore the original values.



**Square error array.** This two-dimensional array lets you change the MSE criteria. Do as follows:

1. Use the array index controls so that the “Square Error Criteria” box shows the MSE criterion for the currently selected range-excitation pair.
2. If the criterion is something else than what is shown on the orange table, write it down as a back-up.
3. Measure your sensor with a long filter.
4. Write down the MSE reading the shown in “Square Error” box when it has reached its final value (it may fluctuate a lot) .
5. Replace the element by a new MSE value.
6. Follow the green light for some time. If it turns off even seldom, then the new criterion is too low and you have to increase it.

The new value should be ten to hundred times greater than the measured MSE. If it is too small, the filter may never become valid, if there happens to be more noise in the measurement.

This is the tradeoff: The higher is the limit, the faster and more surely validity is found, but the distance to the final value will be longer and the first valid averages will contain more readings that belong to the settling transient. The closer the limit is to the measured MSE, the longer it takes to find validity, but the first valid averages are closer to the final value.

## CALIBRATION

The LabView program package contains three VIs for calibration: Main Calibration, ADC and DAC Calibration and Reference Resistor Calibration. The bridge can be calibrated using these programs, or it can be calibrated by using the serial commands as instructed in Part I of this manual (please refer also to those instructions). Both yield equally good results, and because the calibration data is stored in the CPU, both the serial operation and LabView operation use the same stored data.

Using LabView is perhaps more convenient, it gives some more information and -this is very comfortable- a screen capture program can be used for documenting the calibration. The benefit of serial operation is of course, that it does not require LabView.

### MAIN CALIBRATION

The purpose of the Main Calibration procedure is to find correction factors so, that when the bridge measures its internal calibration reference, the display shows the “true” value of that reference. These “true” values have previously been measured at the factory and stored in the CPU’s non-volatile memory. This measurement can also be repeated, which can be needed in some rare cases.

In addition to the digital display and digital output, Main Calibration affects also the analog voltage output.

For each pair of range and excitation, two correction factors are determined: one for zero offset and one for the scale factor. Both are determined using averages of several A/D conversions, as many as 800, when 1M $\Omega$  reference is measured at 3 $\mu$ V excitation. There are 8x8 combinations, each requiring two averages, so 128 averages are needed. This will take time, and the full Main Calibration needs about 3.5 hours, but it is fully automatic and it can be left to run in the nighttime.

Main Calibration can be run as often or seldom as one wants. However, the more often, the better.

### BEFORE STARTING

Main Calibration can, if it should, be run in situ when the AVS-48SI is in the place and at the temperature where it is used. However, the combina-

tion of a high calibrator resistance and a low excitation is quite sensitive to vibration and changing magnetic fields. If these or other interference cause clipping of the signal from the preamplifier, calibration results on these range-excitation pairs will be more or less erroneous. It is therefore best to verify the signal quality.

Select 3M $\Omega$  range and 3 $\mu$ V excitation and start to measure. If you see the red OVL and/or the green AL light on the physical front panel blinking, try the following:

- Replace the connector to the sensor(s) by the short-circuit plug. Never calibrate when there is nothing inserted in the input plug, because even then interference can enter the preamplifier box.
- Try to decouple the preamplifier box from mechanical vibration by placing it on a piece of some soft material.
- Move the preamplifier farther away from pumps etc. and try if it should stand sideways.

Even if these measures do not help and the highest ranges/lowest excitations cannot be calibrated properly, other ranges will not suffer from this. Refer also to “Main Calibration/Precautions” in Part I.

### FULL CALIBRATION

Full main calibration calibrates all range-excitation pairs using long averages (serial command “CALIBRATE 0”). Load the “avs48si\_main\_calibration.vi” and proceed as follows:

- Check that your computer does not turn off automatically after calibration when it is not used.
- Select the correct serial port before starting the program.
- Start the program. CP, DC and DI lights on the physical front panel blink as readings are taken.
- Select method as “all ranges slow”. Because this is a full calibration, range or excitation are not selected manually. The “ADC OUT” shows approximate readings from a 100 $\Omega$  reference and 10mV excitation.
- If this was OK, click “PRESS TO CONTINUE”. After some initial operations, the calibration begins. You can now go home and have a good night’s sleep.

AVS48SI Main Calibration.vi  
2017-03-06

PORT NUMBER  
COM1

SELECT METHOD  
ALL RANGES SLOW


IF USING METHOD 2: SELECT RAN AND EXC

RANGE	EXCITATION	ADC OUT
300R	10mV	0.999907

EXIT W/O SAVING

SAVE AND EXIT

ERROR MSG  
0

 RV-Elektronikka Oy  
Picowatt

DIFFERENCES BETWEEN OLD AND NEW SCALE CORRECTION VOLTAGES

	3uV	10uV	30uV	100uV	300u	1mV	3mV	10mV	
-0.009	-0.006	-0.001	-0.000	-0.000	-0.001	-0.000	-0.000	3R	
0.0341	-0.007	-0.000	-0.001	-0.000	-0.000	-0.000	-0.000	30R	
0.0278	-0.003	-0.002	-0.000	0.0000	-0.000	-0.000	-0.000	300R	
-0.029	0.0045	0.0036	-0.000	-0.000	-0.000	-0.000	-0.000	3K	
-0.002	-0.004	-0.001	-0.000	-0.000	-0.000	-0.000	-0.000	30K	
-0.013	0.0243	0.0000	-0.004	-0.001	-0.000	-0.000	-0.000	300K	
-0.049	0.0403	-0.020	-0.001	0.0055	-0.000	-0.000	-0.000	3M	
0.3020	-0.037	0.2342	0.0667	-0.017	-0.006	-0.004	0.0012	30M	

NEW SCALE CORRECTION DAC VOLTAGES

	3uV	10uV	30uV	100uV	300u	1mV	3mV	10mV	
1.3350	1.5397	1.3372	1.2575	1.1695	1.1730	1.1346	1.1409	3R	
1.3391	1.5301	1.2801	1.1979	1.2080	1.2115	1.1398	1.1457	30R	
1.3179	1.4919	1.2722	1.1916	1.1493	1.1529	1.1798	1.1856	300R	
1.3027	1.4867	1.2668	1.1852	1.1427	1.1466	1.1209	1.1268	3K	
1.2842	1.4824	1.2548	1.1475	1.1380	1.1418	1.1149	1.1209	30K	
1.1691	1.4832	1.2363	1.1282	1.1206	1.1232	1.1097	1.1154	300K	
1.1093	1.0860	1.1023	1.1282	1.1044	1.0825	1.0900	1.0952	3M	
1.0827	1.0396	1.1595	0.7006	0.9579	1.0749	1.0673	1.0486	30M	

NEW OFFSET CORRECTION DAC VOLTAGES

	3uV	10uV	30uV	100uV	300u	1mV	3mV	10mV	
1.5172	1.5199	1.4491	1.4506	1.4497	1.4473	1.4478	1.4500	3R	
1.5148	1.5184	1.4609	1.4511	1.4465	1.4471	1.4477	1.4490	30R	
1.4316	1.4547	1.4467	1.4515	1.4461	1.4472	1.4472	1.4492	300R	
1.4115	1.4554	1.4548	1.4506	1.4471	1.4466	1.4473	1.4492	3K	
1.4772	1.4114	1.4469	1.4530	1.4487	1.4469	1.4474	1.4492	30K	
1.5149	1.4408	1.4582	1.4385	1.4476	1.4468	1.4474	1.4490	300K	
1.4521	1.4306	1.4436	1.4437	1.4457	1.4470	1.4477	1.4494	3M	
1.4405	1.4516	1.4483	1.4452	1.4459	1.4469	1.4475	1.4495	30M	

Successful Main Calibration (full) has been completed.

Full Main Calibration is best aborted by powering the CPU unit off and on.

In the morning, you should see a screen like this screen capture. If there were no overloads during calibration, the ERROR MSG field should show "0". An overload could have caused a message like "6 0 scale correction >2.9V". This means, that calibration of the 3MΩ range (=RAN6) using 3μV excitation (=EXC0) has failed and the calibration can be more or less erroneous.

There are also three useful tables: The new offset correction voltages are usually all almost in the middle of the 0.1...2.9V adjustment range. The scale corrections have more spread, but also they should be within this range. The differences between old and

new scale correction voltages tell, how much these voltages will be changed from the previous calibration (usually some thousands at high excitations and some tenths when excitation is low). Changes in correction voltages have the following effects on the ADC readings:

- offset corr.: +1V increase => 0.01V increase
- scale corr.: +1V increase => 0.1V decrease

If everything looks good, select "SAVE AND EXIT". In case of errors, you must decide what to do. You can check the signal quality, as was discussed earlier, and try a new calibration at a suitable time. You can also save the calibration, if you do not need those failed ran-exc combinations (but somebody else may need them in the future!!).

Taking a photograph or a screen capture of the result is a good way to remember, when the instrument

DIFFERENCES BETWEEN OLD AND NEW SCALE CORRECTION VOLTAGES								
3uV	10uV	30uV	100uV	300u	1mV	3mV	10mV	
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3R
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	30R
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	300R
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3K
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	30K
0.0000	-0.011	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	300K
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3M
0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	30M

Partial Main Calibration of the 300kΩ range and 10μV excitation. This calibration, if saved, would change the scaling of this range-excitation pair by  $0.0011 \times 100k\Omega = 110\Omega$  (or 0.1%) upwards.

was calibrated and how the values compare.

Full main calibration is necessary after the ADC and DAC Calibration and/or after the Reference Resistor Calibration programs.

### PARTIAL SLOW CALIBRATION

If only one or few ranges or excitations will be needed and you have no time, you can calibrate only those combinations in a much less time without compromising accuracy. After having started the program as before, proceed as follows:

- Select method #2, “SELECTED RAN & EXC SLOW” (serial command “CALIBRATE 2”).
- Select the interesting first range-excitation pair.
- Check the “ADC OUT” voltage.
- Click “PRESS TO CONTINUE”. Calibration starts. The time needed depends on selected excitation. 3μV requires less than 5 minutes. Then the screen is updated.

Both correction voltage- fields seem just like in the case of full calibration, but the difference field has only one non-zero entry corresponding to the just calibrated range-excitation pair. If it seems OK, select “SAVE AND EXIT”. The remaining part of the calibration will not be altered.

### FAST CALIBRATION

This method is a first step when starting a new instrument after assembly. It uses short averages and settling times, and it skips some phases that

are required for a final calibration (serial command “CALIBRATE 1”).

### AFTER CALIBRATION

Load the “avs48si.vi” and measure some of the internal reference resistors using the interesting range-exc pairs.

### CALIBRATION OF THE A/D AND D/A CONVERTERS

The LTC2400 analog-to-digital converter itself is very stable and this calibration will not be needed unless:

- A major part of the AVS-48SI has been changed or replaced,
- You want the analog outputs (analog or deviation out, set point DAC and user DAC) to comply with your own DVM or some other instrument.

The LTC2600 8-channel digital-to-analog converter has only moderate performance, which is not comparable with the accuracy of the AVS-48SI. Therefore we have placed the LTC2600 in a correcting software feedback loop, which works as follows: Suppose that one of the eight DAC output channels is instructed to output e.g. 1.0000 Volt. The actual output voltage is measured by the LTC2400 resulting in only 0.9972 Volts (some offset, some scaling error). The firmware calculates the difference and sends a new instruction that demands now 1.0028 Volts. This corrects the output to 1.0000V. The sigma-delta A/D conversion takes time and therefore the A/D converters in the AVS-48SI are slow but exceptionally accurate. Their settling step is 50μV.

In order to make the analog outputs of the AVS-48SI compatible with your DVM, do as follows:

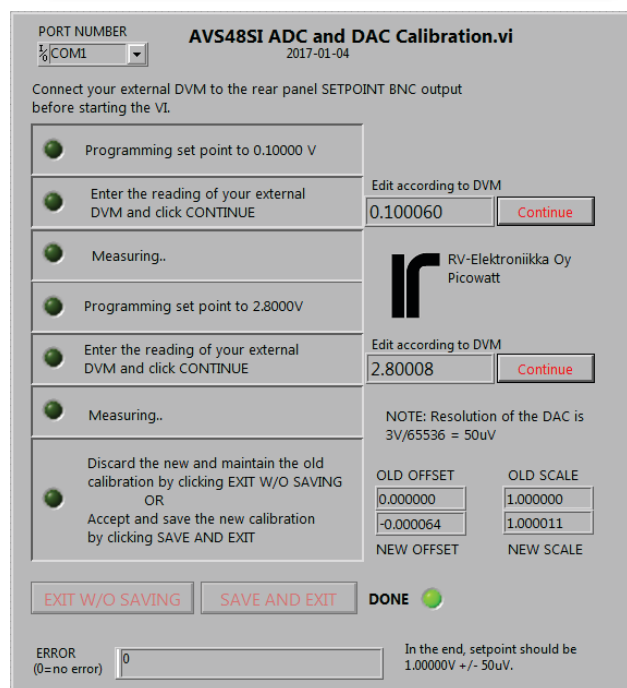
- Open “avs48si\_adc\_and\_dac\_calibration.vi”. Before starting it, verify that the serial port is correct.
- Connect your DVM to the “CONTROL SET POINT” output.
- Start the VI. The DVM should about 0.1 Volts. When the program asks, enter the exact reading of your DVM to the first box. Press CONTINUE.
- The DVM shows then ≈2.8 Volts. Enter the exact reading to the second box. Press CONTINUE.



- If the new offset and scale factors seem plausible, select SAVE AND EXIT. If you made a typing error or the results seem strange, select EXIT W/O SAVING and try again.

Please refer also to similar instructions in Part I.

Full Main Calibration must be run after having calibrated the A/D converter.



The ADC and DAC calibration program makes it easy to align the analog outputs of the AVS-48SI with your own DVM.

## REFERENCE RESISTOR CALIBRATION

This calibration will be needed only if the preamplifier unit, main unit or the CPU box has been replaced. There are two alternate levels:

- Only enter and save known values of the references,
- Measure, enter and save the values

The originally measured “true” values of the reference resistors are written on a sticker inside the bottom cover of the preamplifier. They are also available from us. If there is a strong reason to suspect, that these values are not correct, those values can be measured, and this calibration program makes it easy.

One needs a 4-wire ohmmeter that can measure resistances with offset compensation from  $1\Omega$  to  $1M\Omega$  with a resolution of  $1E-5$  ohms and has an accuracy better than 0.01%. Do as follows for *measuring and entering* the references:

- Prepare a 4-wire measuring cable for your ohmmeter. The AVS-48SI accessory package includes a Molex 22-01-2045 connector pre-wired as follows:

pin	function	wire color
1	+I	brown
2	+V	red
3	-V	orange
4	-I	yellow

Solder this cable to your prepared ohmmeter cable. The following photo shows, how the cable is connected.



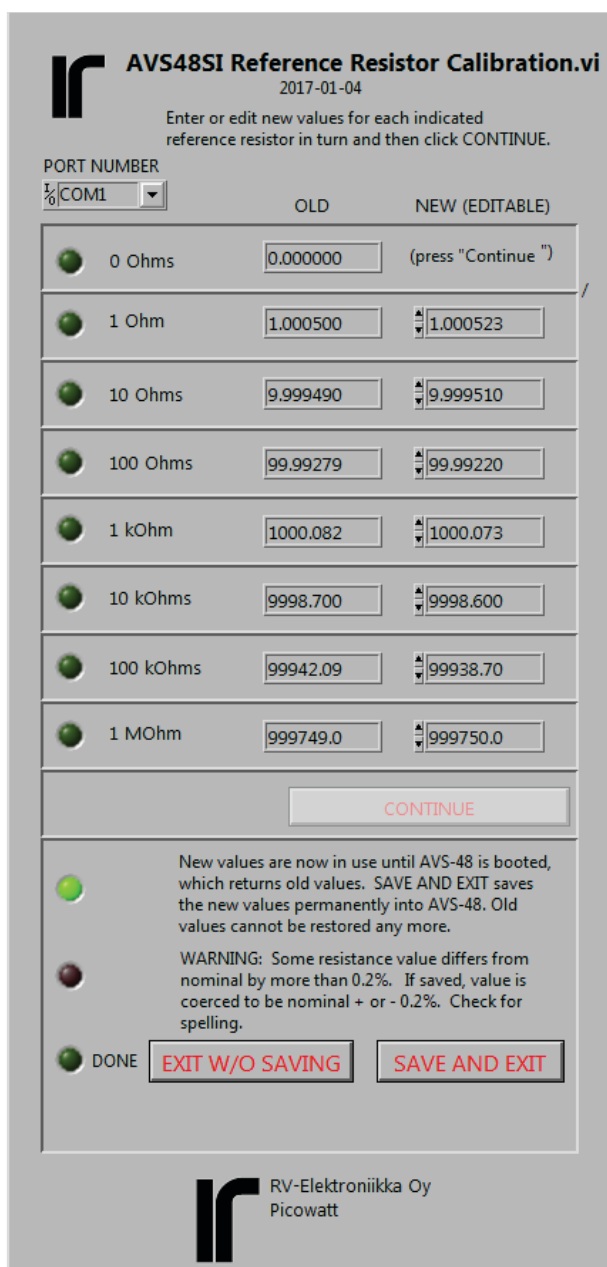
Wires from right to left: +I, +V, -V, -I. The current return lead (-I) is in contact with circuit ground and the box. See the text for actual colors.

- Load “avs48si\_reference\_resistor\_calibration.vi”. Before starting the program, check that the serial port is correct.
- Start the VI. Your ohmmeter should now read zero ohms to better than  $\pm 10\mu\Omega$ . The zero reading cannot be changed. Click “CONTINUE”.
- The  $1\Omega$  reference is now measured. When asked, write your ohmmeter’s reading into the corresponding box and click Continue.
- Repeat the above up to  $1M\Omega$ . The program will then warn, if it has seen an unexpected value. Check then for possible typing errors. If everything looks good, select “SAVE AND EXIT”,

otherwise, do not save but try to find the cause of the error.

When this VI is running, the AVS-48SI is in a special mode, where the reference resistors are isolated from other circuitry but connected to the Molex header by reed relays. During normal operation, the references cannot be measured.

Full Main Calibration must be run after having changed the reference resistor values



**AVS48SI Reference Resistor Calibration.vi**  
2017-01-04

Enter or edit new values for each indicated reference resistor in turn and then click CONTINUE.

PORT NUMBER  
COM1

	OLD	NEW (EDITABLE)
0 Ohms	0.000000	(press "Continue")
1 Ohm	1.000500	1.000523
10 Ohms	9.999490	9.999510
100 Ohms	99.99279	99.99220
1 kOhm	1000.082	1000.073
10 kOhms	9998.700	9998.600
100 kOhms	99942.09	99938.70
1 MOhm	999749.0	999750.0

CONTINUE

New values are now in use until AVS-48 is booted, which returns old values. SAVE AND EXIT saves the new values permanently into AVS-48. Old values cannot be restored any more.

WARNING: Some resistance value differs from nominal by more than 0.2%. If saved, value is coerced to be nominal + or - 0.2%. Check for spelling.

DONE EXIT W/O SAVING SAVE AND EXIT

RV-Elektroniikka Oy  
Picowatt

## TEMPERATURE READOUT

### TEMPERATURE CONVERSION FILES

The AVS-48SI uses simple conversion files for converting resistances to temperatures and vice versa. R/T conversion files are plain text files with two or three columns. Conversion is made by simple linear interpolation between breakpoints. When measuring, interpolation is made from resistance to temperature. When giving a temperature control set point in degrees, interpolation is from temperature to resistance.

If the resistance exceeds the highest breakpoint value, the interpolation algorithm gives the temperature corresponding to the highest breakpoint. If it falls below the lowest breakpoint, the temperature of the lowest breakpoint is the result. Logic is similar when converting the set point from temperature to resistance. An error is indicated if the argument is outside the interpolation range.

The columns are:

- Ordinal number of the breakpoint | resistance | temperature
- OR
- Resistance | temperature

Nine first rows 0..8 are reserved for your free-form comments, like name and type of the sensor. This useful information is visible on the Settings Tab of the "avs48si.vi" program.

- Numbers in adjacent columns must be separated by one or more spaces or tab characters.
- The first breakpoint must be on line #9 (the tenth line).
- Resistances must be in ascending order, from low to high regardless of the polarity of dR/dT.
- A special "avs48si\_create\_r\_t\_information\_file.vi" program exists for writing information about the R/T conversion files and about their types into the "avs48si\_rt\_file\_info.cnf" configuration file. This VI also associates the R/T files with the input channels.

```
Anturityyppi: PT-100 kaupall
Data: vastus vs. celsius
Pisteita: 6
L-tilakerroin: positiivinen
```

1	80.31	-50
2	100.00	0
3	119.4	50
4	138.5	100
5	157.31	150
6	175.84	200

If the sensor covers a large temperature range and is nonlinear, it may be advantageous to use logarithms of the resistance as argument:

```
Sensor Model: RU-1000-BF0.007
Serial Number: U02889
Data Format: 4 (Log Ohms/Kelvin)
SetPoint Limit: 100.0 (Kelvin)
Temperature coefficient: 1 (Negative)
Number of Breakpoints: 198
```

No.	Units	Temperature (K)
1	3.02771	102
2	3.02845	99
3	3.2913	96.5
4	3.02985	94
5	3.03062	91.5
6	3.03144	89
7	3.03232	86.5
8	3.03325	84
9	3.03424	81.5
ETC...		

All conversion files must be located in the avs48si.vi's operating directory.

A very simple conversion file for a room-temperature PT-100 resistor could look like follows (this file is included in the software package for use with the tutorial):

R/T conversion files can be created using a spreadsheet program and by saving the file as plain tab or space limited text (.TXT) without any formatting. Text editors like Notepad can also be used. The sensor manufacturer may provide conversion files, which need no or little editing.

Number of the breakpoints is limited only by capabilities of LabView and its running environment.

## ASSIGNING TO SENSORS

The existing R/T conversion files are assigned to sensors by means of "avs48si\_create\_r\_t\_information\_file.vi". Many similar sensors can use the same R/T file and any sensor can have more than one R/T file, although only one of them can be assigned at a time.

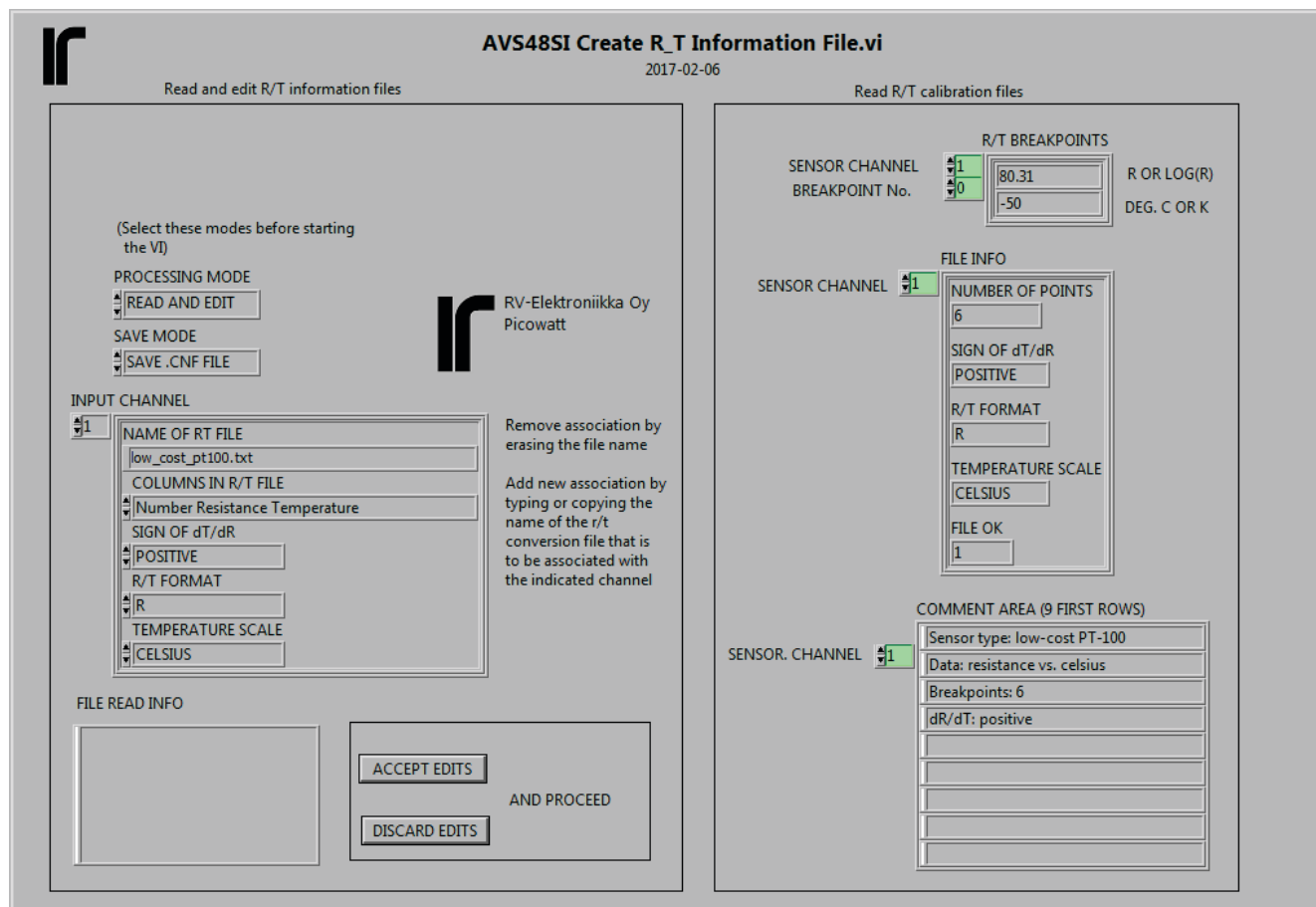
This VI creates an information file, which the "avs48si.vi" needs for interpreting the conversion files properly. Items that must be told to the program for each file to be assigned are:

- The names of the conversion text files. Spelling errors can be avoided by finding the file in the operating directory. The copy and paste its name to the program.
- Number of breakpoints in the file
- Number of columns: the conversion file can have either two columns (resistance and temperature) or three columns (breakpoint number, resistance and temperature).
- Polarity of the sensor's dR/dT (monotonicity is assumed).
- Temperature unit (Kelvin or Celsius)

With this information, the program creates an information file "avs48si\_rt\_file\_info.cnf". It contains records for all 8 channels, but records are empty for channels that have not been assigned. This is a binary file, which can be read only using the "avs48si\_create\_r\_t\_information\_file.vi".

The assignments are made as follows:

- Open the "avs48si\_create\_r\_t\_information\_file.vi" but do not run it yet.



A simple R/T conversion file for a room-temperature PT-100 has been assigned to channel 1. This file has 6 breakpoints (resistance), three columns, positive dR/dT and temperatures are in Celsius. At top right, one can see the breakpoints, at bottom right are shown the possible 9 comment rows.

- Select “READ AND EDIT” as the processing mode and “SAVE .CNF FILE” as the save mode. This is a multi-purpose VI, which has also other modes for use as a sub-VI.
- Start the program. If the “avs48\_rt\_file\_info.cnf” file is not found, the “file read info” field tells that a default file without any prior associations will be created. If a file is found, go through all channels and verify that there are no unwanted prior associations.
- If such unnecessary associations are found, you can remove them by changing the filename to blank or to something non-existent. Removing an association does not remove the R/T file.
- Select the desired sensor channel and enter the required information. You can repeat this now for other sensors and then click “ACCEPT EDITS”. A new information file is created and the VI exits. The possibly existing associations, if they were not removed or changed, remain intact.
- If you just want to see the existing associations and do not want to edit, add or remove anything, select “DISCARD EDITS” and the present file will not be altered.
- Finally, start the VI again using the same modes. On the right side, you can now view the information stored in the file, and correct any possible errors.

Use this method for removing the demo association of a PT-100 to the calibration channel CH0.

Later, when you run the “avs48si.vi” measuring program, the Settings Tab shows this information.



## OPENING THE INSTRUMENT

The main unit of the AVS-48SI (ELMA type 15) has been designed for good shielding against received and transmitted RF interference. Its top and bottom covers have special spring contacts for grounding the covers to the case at many places. These contacts are quite tight if the covers have not been opened many times already. These photographs show, how to open the covers without bending them permanently.



Remove the two side panels.



Lift the cover slightly from the side. Then insert a flat screwdriver or some other flat tool like in this photo. Move the tool to the right and lift the cover from the spring contacts. Do not let the plate to bend sharply - you can lightly press the lifted side downwards for preventing bending.



Start from the rear side. Both sides are similar, but traces on the black front panel are more visible than any damage on the rear. Open all spring contacts with the tool before lifting the cover by hands.



Lift the cover from the back and also the front contacts open.



Check that the 3+3 contacts are in good condition before closing the unit. Press the cover firmly at the positions of the contacts in order to close them properly. You may hear a click when this happens.

# Index

## Symbols

2/4-wire **46**  
12V DC battery input **10**  
18V AC low-voltage input **10**

## A

Accuracy, curve **5**  
AC monitor output **4, 30**  
ADC a/d conversion c/q **26**  
ADC and DAC Calibration **63**  
Adc input channel **59**  
A/D conversion speed, LabView **47**  
ADC reference voltage **59**  
AL alarm signal  
    indicator light **10, 61**  
All ranges slow **61**  
Analog filter **4, 54**  
Analog out BNC **53**  
Analog output **4, 7, 28**  
    rear panel BNC **12**  
Analog overload **9**  
Arduino Mega2560 **7, 20, 23, 36**  
ARN autorange c/q **23**  
Array of clusters **58**  
ASSIGNING R/T FILES TO SENSORS **66**  
Automatic saving of properties **45, 58**  
Autorange **45**  
Autoranging  
    enabling **43, 47**  
AVS-47x compatibility  
    Compatibility w. AVS-47x **18**  
avs48chnames.cfg channel names file **45**  
AVS-48LVA **41**  
AVS-48 main unit **18**  
AVS-48 preamplifier **18**  
Avs48si\_adc\_and\_dac\_calibration.vi **63**  
Avs48si\_create\_r\_t\_information\_file.vi **43, 66**  
avs48si\_data.csv file **56**  
avs48si\_data\_out\_array\_fg.vi **47, 56**  
AVS-48SI-F5 **18**  
AVS-48SI-F10 **18**  
Avs48si\_main\_calibration **61**  
AVS48SI-Picolink **8, 18, 35**  
Avs48si\_reference\_resistor\_calibration.vi **64**  
Avs48si\_rt\_file\_info.cnf **65, 66**  
avs48si.vi (LabView) **41**  
    installing the program **42**  
AVS-48SI-W **18**

AVS48-USB/Serial-W CPU **7**

## B

Booting the CPU **23**  
Breakpoint number **43**

## C

Cables, included cable types **7**  
Cable specifications **35**  
CALADC calibrate ADC and DAC command **31**  
CALIBRATE 1 **63**  
CALIBRATE 2 **63**  
CALIBRATE main calibration command **30**  
Calibration **29, 61**  
    coefficients **20**  
    precautions **29**  
    Reference Resistor Calibration.vi **32**  
CALIBRATION OF THE A/D AND D/A CONVERTERS  
    **63**  
Calibrator **55**  
    stability of references **55**  
CALREF ref. resistor calibration command **32**  
Capacitance, input **4**  
Channel **4**  
Channel name **48**  
Channel names **45**  
    avs48chnames.cfg file **45**  
CH input channel c/q **23**  
Clock input, external **5**  
Coercing command's argument **21**  
Command arguments **21**  
Command delimiter **21**  
Command interpreter **37**  
Command order **21**  
Commands and queries **21**  
Comment area (9 first rows) **58**  
Compatibility w. AVS-47x **9, 18**  
Compliance voltage, heater output **49**  
COM: port **18**  
Connections betw. instruments **7**  
Connector for measuring ref. resistors **32**  
Control CH **55**  
Control channel **44**  
Control error meter **48**  
Control error output  
    rear panel BNC **12, 53**  
Control sensor c/q **23**  
Control set point **28**  
    rear panel BNC **12**  
Correction factors **29, 61**  
CP Clock Pulse  
    indicator light **10**  
CPU box, opening **38**



CPU unit 18  
CSV string 56  
CTS 35  
Current leads, resistance 4

## D

DAC outputs 5  
Data file name **56**  
Data saving mode **56**  
DC data from computer  
    indicator light 10  
DEFAULTS command 22, **34**  
Delays 22  
DERG derivator gain c/q **24**  
Derivator gain **50**  
Deviation output, polarity 53  
DI data from instrument  
    indicator light 10  
DIRECT DAC 56  
    Direct DAC(V) control box 56  
Disabling automatic saving of properties 58  
DLY delay command **33**  
DR/DT **53**  
    deviation polarity 53  
DRDT control polarity c/q 23, **25**  
DTR 35  
DVMHI command 31  
DVMLO command 31

## E

EEPROM 41  
EEPROM write count 26  
EMI 18  
Enter **51**  
ENTER button 44  
    turns dark 51  
EPRADC, read ADC cal. coefficients 31  
EPRCAL read Main cal. coefficients from EEPROM 31  
EPRREF read ref. resistor values from EEPROM 32, 33  
ERR error query **34**  
Error indicators lights **9**  
Error messages 34  
Error output, temperature control 4  
ERRSIGNAL error signal query **25**  
EXC excitation c/q **23**  
Excitation 4, **45**  
    magnitude of current 45  
Exit **52**  
External clock input 5  
    frequency equation 14  
    rear panel BNC **14**  
External heater 4

External heater drive  
    INT/EXT jumper JP1 13  
    rear panel BNC input **13**

## F

FAST CALIBRATION **63**  
FFT fast Fourier transform 13  
fibre-optical connection 18  
Filter  
    indicator light 43  
    settling time 46  
    zero length 46  
Filter, analog 4  
Filter delay **53**  
Filter length **46**  
Filter mode **46**  
    always 46  
    smart 46  
Filter mode control 43  
Filter output **54**  
Four wire connection 46  
Front tab 42  
Full calibration 30  
FULL CALIBRATION **61**  
Functiona global (FG) variable 47  
Fuse **10**

## G

Galvanic isolation 18  
GNDS sensor grounding c/q **23**  
Ground currents 18  
Grounding, sensor 4

## H

HDACV heater drive voltage c/q **24**  
Heater connection **51**  
Heater Direct DAC **56**  
HEATER DRV SOURCE **56**  
Heater, external drive 4  
Heater output 4  
    maximum power 24  
    rear panel BNC **12**  
Heater power 5  
Heater power display (digital) **49**  
Heater power % meter **49**  
Heater range **49**  
    changing downwards 49  
    table of ranges 24  
Heater resistance 5, 24, **55**  
Hold **52**  
HOLDMODE command/query **25**



HTRDIR heater drive by voltage c/q **24**  
HTRI heater output current query **25**  
HTRP heater output power query **25**  
HTRRAN heater range c/q **24**  
HTRV heater output voltage query **25**  
HW? hardware query **20, 22**  
Hyperterminal program **19**

## I

Idle **47**  
IDN? **58**  
IDN? identification query **20, 22**  
Initializing the FG **58**  
Input capacitance **4**  
Input channel **45**  
Input connector **4**  
    wiring **9**  
Installing the avs48si.vi **42**  
Integrator gain **50**  
Interference **61**  
Internal resistances **3**  
Interpolation, R/T files **65**  
    limits **65**  
INTG integrator gain c/q **24**  
INTHEATER external/internal heater c/q **24, 25**  
Introduction **18, 41**

## L

Laboratory environment **3**  
Lead resistances  
    checking **9**  
    measuring **46, 48**  
    warning light **9**  
Lead resistances, maximum **4, 48**  
Least squares line **54**  
LINETERM c/q **22**  
Line terminator **20, 21**  
Log(R) in an R/T conversion file **43**  
Loop gain, temperature control **49, 52**  
Low\_cost\_pt100.txt demo R/T file **43**

## M

Magnetic fields **61**  
Main calibration **29**  
MAIN CALIBRATION **61**  
    time taken **61**  
Mains adapter **7**  
Mains hum **29, 30**  
Mains input **10**  
MAX? maximum query **20, 26**  
Mean Squared Error (MSE) **54**  
Measure **47**

Message line terminator **21**  
MIN? minimum query **20, 26**  
Multi-line append **56**

## N

National Instruments USB-232 converter **7, 19, 42, 53**  
Noise  
    oscilloscope view **13**  
    STD curves **6**  
Noise voltage **4**  
Notepad **66**

## O

Opening the CPU box **38**  
Opening the instrument **9**  
OPENING THE INSTRUMENT **68**  
Optical cable **8**  
    renewing ends **8**  
Optical link option **8**  
Oscilloscope, connection **30**  
Outputs, analog **4**  
Overload **48**  
    warning light **9, 61**  
Overshoot, temperature control **50**

## P

Partial calibration **31**  
PARTIAL CALIBRATION **63**  
Peak-to-peak resolution **5**  
Physical front panel **9**  
Physical rear panel **10**  
    viewed from front side **11**  
Picobridge **3, 18**  
Picobus cable **35**  
Picobus, proprietary protocol **41**  
Picobus Serial Input **10**  
Picobus status lights **10**  
Picolink **35**  
PID parameters **5**  
Power input **5, 7**  
Power switch **9**  
Preamplifier input connector **14**  
PRESETMODE command/query **27**  
Primary interface, picobus **10**  
PROPG proportional gain c/q **24**  
Proportional gain **50**  
PSDF operating frequency c/q **23**  
PSD Frequency **54**

## Q

QRATIO? quality ratio query **20, 26**  
Quick calibration **31**





## R

Range 4, **45**  
    changing r. of control sensor 52  
RAN measuring range c/q **23**  
Raw readings, filter output 54  
Recall **47, 51**  
RECALLBR recall channel properties **27**  
Recalling  
    channel properties 43  
RECALLTC recall heater range properties **28**  
REFERENCE RESISTOR CALIBRATION **64**  
Reference Resistor Calibration.vi 32  
Reference resistors 31  
REFID **32**  
REFID ordinal number of calib.reference c/q 29, 32  
REFVALUE **32**  
REFVALUE “true” value of calib.reference c/q 29, 32,  
    64  
Remote diagnosis 59  
Removing demo R/T file association 67  
REPEAT command for repeating mode **33**  
Reprogramming the CPU 36  
RESETALL command 22, **34**  
Resistance display **48**  
Resolution, curve 5  
Response to \*IDN? **58**  
RES resistance c/q **26**  
RESTART command 22, **34**  
Re-starting the CPU 9  
RS232 18  
R/T conversion files 65  
    assigning to sensors 43  
R/T file information **58**  
RTS 35  
RXD 35, 38

## S

SAVEADC save adc offset and scale factors **26, 31**  
SAVEBRD save channel properties **26**  
SAVECAL save calibration coefficients **26, 29, 30, 31**  
SAVELINETERM save line terminator **27**  
SAVEREF save ref. values command **26, 32**  
SAVETCR save heater range properties **27**  
Saving data from one sensor 46, 47  
Saving measured data  
    for other LabView VIs 56  
    to a disk file 56  
Saving off/on **56**  
Saving of properties 45, 58  
Scale **51**  
Scan **47**

Scan enable **46**  
Scanning  
    enabling for a CH 43  
    filter length 47  
    order of channels 47  
    SCAN command 43  
    speed 47  
SDACV **28**  
SDACV set point DAC c/q **24, 31**  
Sensor grounding **46**  
Sensor, grounding 4  
Sensor heating, starting to measure 47  
Sensor wiring **9**  
Serial cable 35  
Serial format 7  
Serial port **53**  
    serial port number 19, 42  
Service measurements 37  
Service tab 42  
Set point **51**  
Setpoint DAC (V) **55**  
SETPOINT resistance set point cmd **25**  
Set point, temperature control 4  
Set point unit **50**  
    changing between R and T 50  
    exceeding limits 50  
Settings tab 42  
Shielded room 18  
Shielding braid 35  
Short-circuit plug 7, 8, 30, 42  
Signal overload 48  
Single-line replace 56  
Smart filter 41, 46, 54  
    LAST POINT output mode 55  
    MEAN output mode 54  
Specifications  
    Other 5  
    Temperature controller 5  
Speed of balance 4  
Spreadsheet program 66  
Square error array **60**  
Square error table **59**  
Standard deviation STD 5  
Standard deviation (V) **59**  
Start-up state 42  
Statistics queries 20  
STD? standard deviation query 20, **26**

## T

Temperature controller 18  
    disabling high ranges 49  
    enabling control 49, 55  
    internal test assembly 51



- resetting integrator by mistake 50
- stopping control 49, 51, 52
- testing with an internal load and sensor 44
- Temperature conversion files 65
  - examples 66
- TEMPERATURE CONVERSION FILES 65**
- Temperature display **48**
  - warning lights 48
- Temperature readout 41
- Testing temp.ctrl with internal heater 25
- TIME command for measuring time interval **34**
- Timeout 21
- Timing issues 21
- Trade marks 3
- Transfer function, controller 5
- Transient part of settling data 54
- Trouble-shooting
  - serial operation 37
- TS-530A temperature controller 18
- Tutorial
  - LabView operation 42
  - serial operation 19
- TW 2/4-wire connection c/q **23**
- Two-wire connection 46
- TXD 7, 38

## U

- UDACV 28**
- Unbundle by Name 58
- Unpacking the shipment 7
- USB 18
- USB-232 converter 7, 35, 53
- USB-232 converter, installing 7
- USB cable 36
- User DAC 5
  - accuracy 13
  - rear panel BNC **13**
- User DAC output 13, **28**
- User DAC (V) **55**

## V

- Vibration 3
  - sensitivity to 61
- Virtual instrument 41

## W

- Wall adapter 7
- Warranty 3
- Write count, EEPROM 26

## DECLARATION OF CONFORMITY



Manufacturer: **RV-Elektroniikka Oy Picowatt**

Address: Veromiehentie 14  
01510 VANTAA  
Finland  
Telephone: +358 50 337 5192  
E-Mail: reijo.voutilainen@picowatt.fi

**declares that under our sole responsibility**

Product Name: AVS-48 Picobridge

Product Description: Cryogenic AC Resistance Bridge including 19" main unit, small preamplifier unit and an interconnection cable

**is in conformity with the following Directives:**

2004/108/EC: Electromagnetic Compatibility  
2006/95/EC: Low-voltage Directive (Electrical Safety)  
2011/65/EU: ROHS Directive

**and that the following harmonized standards have been applied:**

EN 50 081-1: Generic emission standard, Part 1: Residential, commercial and light industry  
EN 50 082-1: Generic immunity standard, Part 1: Residential, commercial and light industry  
EN 61 010-1: Safety requirements for electrical equipment for measurement, control and laboratory use  
EN 50 581: ROHS

**Additional information:** This product has been designed for measuring low and ultralow temperatures in cryostats using resistive temperature sensors. In order to prevent the measuring current from heating the sensor, currents down to 1pA can be used. The environment is therefore expected to be electromagnetically benign, free from changing strong electric and magnetic fields and free from significant changes or interference in the mains supply voltage.

Vantaa, 25 June 2013

RV-Elektroniikka Oy Picowatt

Reijo Voutilainen  
President