2006

AVS47- IB

TWO-STAGE COMPUTER INTERFACE
for the **AVS-47B**

INSTRUCTION MANUAL

Serial Number. 68147A63R2 - 70047A63R2

RV-Elektroniikka Oy PICOWATT
Veromiehentie 14
FI-01510 VANTAA, Finland
phone +358 9 822087
fax +358 9 822184
Internet: www.picowatt.fi

Contents

(use this manual only with the AVS-47 "B" version and AVS47-IB having serial number within denoted range)

1. COMMAND SUMMARY (firmware version 3R2)

NOTE 1: All commands listed below are sequential.
NOTE 2: All responses are single-line.

BASIC AVS-47B COMMANDS

REM [0..1 | ?]	Local/Remote Mode Command / Query. (The interface does not recognize the IEEE-488.1 REN command).
INP [0..2 | ?]	Input Command / Query (Zero..Measure..Calibrate).
MUX [0..7 | ?]	Multiplexer Channel Command / Query.
RAN [0..7 | ?]	Range Command / Query. 0=no range...7=2MΩ.
EXC [0..7 | ?]	Excitation Command / Query. 0=no excitation...7=3mV.
DIS [0..7 | ?]	Display Selector Command / Query. 0=R...7=TS-530 Set Point.
REF [0..20000]	Set the Difference Reference Command. (No query).
RFS ?	Reference Source Query. 0=memory...1=direct potentiometer.
MAG ?	Difference Magnifier Query. 0=dRx1...1=dRx10.
ARN [0..1 | ?]	Manual/Autorange Command / Query. (AVS-47B must be in manual when ARN=1).
SDY [1..100 | ?]	Stabilisation Delay Command / Query (seconds).
ZDR	Zero Difference Command (macro). Takes the reference from the resistance reading.
ADC	Make A/D Conversion Command (<400 milliseconds).
ADC ?	A/D Conversion Result Query. Range -19999 to +19999. Overload indicated by 20001. This query must be preceded by the ADC Command.
RES ?	Resistance Query. Gives resistance in Ohms as a real number. Range -1.9999E+6 to +1.9999E6, overload indicated by 2.0001E+6. Must be preceded by the ADC Command.
AVE [1..1000]	Take Average Command (needs n x 0.4 seconds).
└─ STP	Stop (in the middle).
	Results are asked using the following four queries:
AVE ?	Average
MIN ?	Minimum
MAX ?	Maximum
STD ?	Standard Deviation.
OVL ?	Overload Query. OVL=1 if the last A/D conversion was overload, or if overload occurred during averaging in manual mode. OVL works on all ranges in manual mode, in ARN1 mode only if R>2MΩ.

TS-530 COMMANDS

SPT [1..42000 | ?]	Set Point Command / query (microvolts x 100).
PRO [0..11 | ?]	Proportional Gain Command / query
PRO 15	Force Error Signal to Zero Command
ITC [0..11 | ?]	Integrator Time Constant Command / query
DTC [0..7 | ?]	Derivator Time Constant Command / query
BIA [0..5 | ?]	Power Bias Command / query
POW [0..7 | ?]	Heater Output Power Range Command / query
SPV, SPV ?	Measure Set Point Voltage Command, Set Point Voltage Query.
HTV, HTV ?	Measure Heater Output Voltage Command (macro), Heater Output Voltage Query.
HTI, HTI ?	Measure Heater Output Current Command (macro), Heater Output Current Query.
HTP, HTP ?	Measure Heater Output Power Command (macro), Heater Output Power Query.
	Use always as pairs. Allow for some seconds delay after the command.

DIGITAL FILTER COMMANDS

DFL [1..1000 | ?]	Filter Length Command / Query.
DFS	Start Digital Filtering Command.
├─ STP	Stop.
└─ DFR ?	Get Filtered Reading Query. Reading cannot be printed nor buffered.

SCANNING COMMANDS

FCH [0..7 | ?]	First Scan Channel Command / Query.
LCH [0..7 | ?]	Last Scan Channel Command / Query. (FCH ≤ LCH).
SCI [0..10000 | ?]	Scan Interval Command / Query (seconds). Only in continuous scan mode.
ETC [0..1 | ?]	Disable/Enable Temperature Control Command / Query.
TCP [0..1000 | ?]	Temporary Temperature Control Period Command / Query (seconds).

SCP [0..7 | ?]	Set Channel Parameters Command / Query. SCP n must be the first command in a new line. SCP specifies the channel for which one or all of the following commands or queries, on the same single line, are given:
├─ SDY [0..100 | ?]	Stabilisation Delay after autoranging or change of any AVS-47B setting.
├─ CNT [1..1000 | ?]	Average Count.
├─ EXC [0..7 | ?]	Excitation.
└─ RAN [0..7 | ?]	Range. When ARN=0, this range is used for channel n. If ARN=1, then measurement starts from this range, but autoranging may change it.
SCN [0..2]	Start to Scan Command (macro). SCN 0 starts a single cycle. SCN 1 begins the Front-Panel-Started scan mode. SCN 2 starts continuous scanning.
└─ STP	Stop.
SCR [0..7 | ?]	Scan Results Command / Query. SCR n must be the first command in a new line. It specifies the channel for which one or all of the following queries, on the same single line, are given:
├─ AVE ?	Average.
├─ MIN ?	Minimum.
├─ MAX ?	Maximum.
└─ STD ?	Standard Deviation.

GENERAL AVS47-IB COMMANDS

TIM [hour,minute,second | ?]	Real Time Clock Command / Query.
DAY [year, month,day | ?]	Real Time Date Command / Query.
PRN [0..1 | ?]	Printer off/on Command / Query.
DSK [0..1 | ?]	Buffer off/on Command / Query (RAM disk).
PBF	Print Buffer File Command. Must have PRN=1 and printer on-line.
└─ STP	Stop.
RBF	Read Buffer File Mode Command.
├─ NXT ?	Proceed to and Read the next data line Query.
└─ STP	Stop.
BUF ?	Read size of the buffer file and free buffer space.
KIL	Delete the Buffer File Command.
DLY [1..1000]	Insert Delay Command (seconds).
PONRST	Power-on Reset Command (simulates power-on, does not destroy the buffer).

IEEE-488.2 COMMON COMMANDS AND QUERIES

*IDN ?	Identification Query.
*STB ?	Read Status Byte Query.
*SRE [0..255 | ?]	Service Request Enable Command / Query.
*ESE [0..255 | ?]	Standard Event Status Enable Command / Query.
*ESR ?	Standard Event Status Register Query.
*CLS	Clear Status Command.
*OPC	Operation Complete Command.
*OPC ?	Operation Complete Query.
*RST	Device Reset Command (resets both the AVS-47B and the TS-530).
*TST	Self-test Query.
*WAI	Wait-to-Continue Command is bypassed (all commands are sequential).

OTHER IEEE-488 RELATED COMMANDS

IBA [1..30 | ?]	IEEE-488 Device Address Command / Query.
HDR [0..1 | ?]	Include Response Message Headers Command / Query.

PICOBUS RELATED COMMANDS

PBA [1..15 | ?]	Picobus Device Address Command / Query.
PBD [200..1000 | ?]	Picobus Delay Factor Command / Query.

MISCELLANEOUS COMMANDS

LRS, LRS ?	Measure Lead Resistance Command (macro) and Query.
SCAL [0..1]	Self-Calibration Command. SCAL 0 resets the calibration (offsets=0 and scale factors=1. SCAL 1 needs about 15 minutes to complete. Calibration data is **not** protected against power failure.
SCAL ?	Self-Calibration Factors Query.
FSM [0..1 | ?]	Disable/Enable Front Panel Started Measurement Command / Query. Disable if you have more than one instrument in the same Picobus line.
EWC?	Eeprom Write Count query.

2. INTRODUCTION

One of the most difficult problems in resistive low temperature thermometry is the unwanted extra heating of sensors. This heating is usually caused by high frequencies that are emitted by radio stations, electrical appliances, instruments involving fast digital electronics, and unfortunately also by computers and the IEEE-488 instrumentation bus.

The electromagnetic interference (EMI) is usually divided into two categories: emitted and conducted interference. Guarding against the emitted interference is made by shielding whereas conducted EMI requires filtering. But this is only the beautiful principle. One can realize, by just looking at the jungle of cables around a cryostat, that is not possible to separate the heating mechanisms - the reality is a mixture of conduction, capacitive and inductive coupling plus transmission and reception of HF waves.

A cryostat is usually a very well shielded environment. Therefore the most obvious thing to do is to filter the lead wires to the resistive sensors. However, this may be impractical if there are many 4-wire sensors so that the space taken by the filters is too large. Filtering can also turn out to be insufficient, if the interference enters the cryostat via other cables that cannot be filtered, like those for a magnet, NMR pick-up coil etc.

Eventually, one may be forced to use the trial-and-error method by
- filtering the mains power separately for each instrument or by using isolation transformers,
- shielding the cables as well as possible,
- making them as short as possible,
- trying to find the best ways of grounding,
- RF-filtering those lines to the cryostat that can be filtered,
- using optical fibres for digital signals
- purchasing only EMI silent instruments.
- abandoning the idea of a modern computerised automatic system, and returning back to “good old times”.

The **AVS47-IB** is a two-stage interface that consists of a silent primary unit inside the **AVS-47B** Resistance Bridge and of a secondary IEEE-488 unit in a separate shielded box. It has been designed to make the AVS-47B as quiet as possible while still allowing it to be interfaced with the popular IEEE-488 bus and to be used together with a computer. However, the **AVS47-IB** cannot prevent any other instruments from generating EMI.

The primary section is based on our proprietary **Picobus** interfacing technique. The **Picobus** is a slow synchronous serial transfer protocol, which can be implemented without neither a microprocessor nor other high-speed digital circuits. It does not even have a clock oscillator, so that it is extremely silent except during a transaction. The average energy of the high-frequency signals that it can emit is minimal.

The secondary section is based on a CM420 PC/104 Module from AMPRO Computers Inc., USA, accompanied by an IEEE-488 interface card . This unit, which has its own separate metal enclosure, communicates with the external world through its IEEE-488 port, and with the AVS-47B via its primary interface. The primary and secondary interface stages are connected together by a **Picobus** cable. The length of this cable is not limited. If the waveforms are deteriorated because of the cable length, the speed of the **Picobus** can be reduced by a simple command. The same feature makes it possible to filter the **Picobus** line effectively if desired.

The **AVS47-IB** has a comprehensive and extremely versatile command set, consisting of something like 70 commands. In fact, calling the **AVS47-IB** a “Computer Interface” does not make justice to this product, because it turns the basically analog AVS-47B into a highly sophisticated and intelligent instrument without sacrifying anything from its low-temperature performance.

We have tried to follow the IEEE-488.2 standard where possible. Deviations should be minor and they should not affect the normal operation too much. Except for very simple applications, some prior knowledge on IEEE-488 status reporting, service requests and so on, would be useful. On the contrary, when using the LabView Driver, these skills are not so important any more, as the low-level programming has already been done by us for you.

This manual incorporates a command summary, which you may want to copy and keep at hand while writing the application program. And before starting to do that, read the Getting Started section which tells what you necessarily must do and know in order to get the interface working. Even if you decide to use the LabView, it might be a good idea to look through this manual, because the driver relies on the commands described here.

Picolink

In cases where the best possible isolation between the AVS-47B and the AVS47-IB is necessary, the standard Picobus can be replaced by the optional **Picolink** optical fibre connection.

Picolink consists of transmitter/receiver circuit boards inside both the AVS-47B and the AVS47-IB. The two boards are interconnected by a link that is made of four low-cost plastic fibres. The length of the optical cable is 5 metres. The Picolink option can be installed to new instruments only at the factory. Later upgrading is not possible.

LabView Driver

Graphical user interfaces have made complex programs easy to use, and only few of us need to write low-level code any more. This applies also to programs that are needed for connecting laboratory instruments together, for controlling their operation and for collecting the data.

Perhaps the most important one of such programs is the **LabView** software from National Instruments. LabView is based on blocks that look similar to those used in electronic block diagrams. The blocks, which are called “virtual instruments” or VI:s, have inputs and outputs which are connected together by drawing lines between them. LabView checks automatically that connections are made properly. These blocks contain the program code on varying levels of hierarchy. The blocks are actually graphical representations of procedures and functions, the inputs and outputs representing parameters that appear in the headings used for calling the procedures in a text-based language.

Programming in LabView is mostly based on connecting graphical symbols together. As soon as data from a measuring instrument has been received, it can be analysed in almost any imaginable way.

A **LabView Instrument Driver** is a set of low-level VI:s, which take care of the communications with the instrument. With a good and detailed documentation about a computer interface, an experienced user can write such VI:s, but usually it is the manufacturer that makes this work and offers the VI:s to customers free of charge. This is also the best way to guarantee that the VI:s work as expected. An Instrument Driver hides device- or bus-specific peculiarities, such like timing issues, making remote control easy. A driver contains graphical block-diagram symbols for groups of instrument-specific commands plus whatever commands or actions are necessary to communicate with the instrument and to initialize it for remote use.

A **LabView application** consist typically of two types of virtual instruments: VI:s of the first type are used for configuring an instrument for the task, making the required measurements and for obtaining the data from it. VI:s of the second type are used to present, save or analyze the data. The first type VI:s typically belong to instrument drivers, whereas VI:s of the second type are usually written by the user.

The LabView Driver that we have written for the **AVS47-IB** contains practically all the commands described in this manual. It provides tools for controlling almost all features of both the **AVS-47B** and **TS-530A**. In addition, many powerful programmable macros of the AVS47-IB interface, like digital calibration or scanning, are included in the driver so that their use is now much easier than before.

In order to program with LabView, you must purchase the software from National Instruments. You must also have an GPIB controller card in your computer. It is a safe decision to buy the GPIB card also from National, as this guarantees that the software and the GPIB card work together without problems.

National Instruments have published general guidelines regarding the structure and appearance of LabView Instrument Drivers. We have tried to follow these instructions as much as possible. The driver and its written manual are available for downloading from our WEB site www.picowatt.fi.

Who is a “serviceman” ?

Some operations to the instrument, like changing the mains voltage setting, may require that it be opened. There are hazardous voltages inside the unit, which one can touch. Therefore, such operations are allowed only for a "qualified serviceman": “A person having appropriate technical training and experience necessary to be aware of hazards to which he is exposed in performing a task, and of measures to minimize the danger to him or other persons”.

3. WARRANTY

Picowatt warrants this product to be free from defects in material and workmanship. Our liability under this warranty is limited to repairing or replacing any instrument or part thereof which, within the warranty period, proves defective. This warranty is void if the instrument has not been used according to the instruction manual, or if it has been used or stored under environmental conditions which significantly differ from those of a normal laboratory environment. This warranty is also void if the instrument has been modified without our authorization.

The warranty period is three (3) years from shipment to the original purchaser, except one (1) year for the Core Module made by Ampro Computers Inc., USA.

In need of warranty repair, a description of the fault must first be given to the local distributor or to us directly. The name, address and other contact data of a person who can give supplementary information shall be included. After our permission the instrument can be shipped to us by Air Freight or by Courier. **Picowatt** will take responsibility of the freight costs in repair cases which are covered by warranty.

If no fault is found, or if there is a strong indication that the repair is not covered by warranty, the purchaser shall be responsible for all shipment costs.

Repairs, which are not covered by warranty, are made at nominal or reasonable cost. Estimates of the repair costs are given when possible. Often it is, however, not possible. After the fault has been found, an estimate could be given, but most of the work has already been done by successfully locating the fault.

CAUTION:

THE AVS47-IB IS A DELICATE LABORATORY INSTRUMENT. IT HAS BEEN DESIGNED ONLY TO BE INTERFACED WITH THE PICOBUS INSTRUMENTS AND THE IEEE-488 BUS. USING THE AVS47-IB FOR ANY OTHER PURPOSE WILL MAKE THIS WARRANTY VOID. ALSO PERMANENT DAMAGE TO THE INSTRUMENT MAY RESULT FROM INAPPROPRIATE CONNECTIONS.

THE AVS47-IB HAS BEEN DESIGNED TO OPERATE IN A LABORATORY ENVIRONMENT, WHICH MEANS NORMAL LIVING ROOM ATMOSPHERE, TEMPERATURE, HUMIDITY AND PURITY OF AIR. THE UNIT MUST BE ESPECIALLY PROTECTED AGAINST VIBRATION OR SHOCKS.

RV-Elektroniikka Oy PICOWATT
Veromiehentie 14
FI-01510 VANTAA, Finland
phone 358 9 822087 or +358 50 3375192 (mobile)
fax 358 9 822184
email reijo.voutilainen@picowatt.fi
Internet: www.picowatt.fi

4. PICOBUS CABLE

The cable between the AVS-47B and the AVS47-IB interface box (or alternatively, a PC computer) is called **Picobus Cable**. This shielded cable has 6 inner wires, a female DB25S and a male DB25P in the ends, and it is connected as follows:

AVS47-IB end (DB25S)		**AVS-47B end (DB25P)**	
Pin	RS232 signal	Pin	Picobus signal
1	SHIELD, SHELL	1	SHIELD[2]
4	RTS[1]	4	CP Clock pulse
5	CTS	5	DI Data from Instrument
6	DSR	6	AL Alarm Status Line
7	GND	7	GND Signal Ground
20	DTR	20	DC Data from Computer

NOTES:
[1] The RS232 names are here only for help, if you want to monitor the bus operation with an RS232 tester.
[2] By default, shield has been disconnected from AVS-47B ground by detaching the ground lead from its header J201. This enables full optical isolation of the Picobus cable.

The current Picobus Cables can be used with both the AVS-47B and the older bridge revisions. The cable supplied with AVS-47B has connector pin 1 soldered to the shield at **both ends** of the cable, whereas the metal shell of the connector is connected to the shield only at the female end. This end is for the AVS47-IB box or a computer.

Inside the AVS47-IB box, the cable shield is permanently connected to ground, whereas inside the AVS-47B bridge, it is the grounding wire going to either J201 or J202 that determines, whether the shield is connected to bridge ground. The purpose of this arrangement is to offer means for finding the best grounding scheme for the devices in many experimental situations. **Please remember that the shield must always be grounded at least at one point.**

• If the shield is grounded at both ends by inserting the grounding wire into J202 (GND), immunity against electrostatic discharges (ESD) is good, but optical isolation does not work and ground currents can flow via the shield from the AVS47-IB to the AVS-47B
• If the shield is grounded only to the AVS47-IB by inserting the cable is into J201 (FLOAT), galvanic isolation works and there is no path for ground current. On the other hand, the electrical circuits of the instruments are more vulnerable to ESD.

Default with the AVS-47B is that J201 is occupied and full optical isolation is enabled (FLOAT). Make sure that the shield is grounded somewhere. As a final resort, insert J202. A cable that is not grounded is likely to cause troubles by radiating RF.

5. MAINS POWER VOLTAGE

The AVS47-IB has a metal enclosure. In order to maintain its electrical safety, it must be connected to a grounded power outlet.

The mains voltage range is 90..260V for AVS47-IB units with serial number greater than #680. Older units with a linear power supply have two fixed ranges 90V-125V, and 180-250V. The voltage range or setting is marked on the rear panel.

If the fixed range of an older unit does not correspond to your local voltage or in case of **any uncertainty**, please let a serviceman open the top cover of the **AVS47-IB** and check the location of the 8-way jumper. This jumper can be lifted with pliers (do not pull from the leads) and plugged into either of the two male header connectors marked “115V” or “230V”.

Always disconnect the AVS47-IB from mains before opening it. Because of a possibility to touch a hazardous voltage, opening the AVS47-IB is allowed only for a qualified serviceman.

Although the power supply is protected by a fuse, there exists a potential danger of a serious and expensive damage in the case of connecting a 115V unit to 230V mains.

5.1. THE MAINS FUSE

The **AVS47-IB** has a primary and possibly also a secondary fuse. The primary fuse is located on the rear panel and the secondary fuse is on the power supply circuit board. Its rating is independent of the mains voltage.

Primary fuse:
Size: 5 x 20 mm
1A-T (slow action)

Older AVS47-IB units have either 160mA-T fuse for 115V or 80mA-T fuse for 230V. These fuses are too small to stand the inrush current of the new switching power supply, whereas the 1A fuse is too big to protect the old linear supply. Please use a fuse of the correct size.

Older power supplies have also a secondary fuse, either a 5x20mm 1A-T fuse or a resettable semiconductor fuse. Its purpose is to protect the power supply from load that is excessive for the supply but not large enough to break the primary fuse. A typical condition that can lead to shutdown of the secondary fuse is, if one **forgets to remove** the two power jumpers in the AVS-47 or AVS-47A. If the AVS47-IB is then powered and the AVS-47[A] is not, excessive power may be drawn from the secondary unit and the fuse should break the current. This warning does not apply to the current “B” revision, which has an extra voltage supply for Picobus.

5.2. THE COOLING FAN

The new CM420 PC/104 Core Module consumes four times more power than the older 386SXi module. Because of this, the linear power supply was replaced by a small switching unit with a universal 90..260V input voltage range. In order to enhance the free convection cooling, a small fan had to be added. The fan is specified to have an estimated lifetime of 50000 hours. In spite of the noise that it generates, the fan must not be disabled.

Please take care that the fan operates and that the air holes on top and bottom of the box are not blocked. If the fan stops or air flow is stopped, the internal temperature of the AVS47-IB box can rise up to 70 degrees Centigrade, and correct operation of the circuits cannot be guaranteed. Excessive heat can also result in permanent damage to semiconductors or firmware stored in the CF memory.

6. THE AVS-47B DEFAULT SETUP

The default setup of the AVS-47B **Primary Interface** is suitable for connecting the bridge either to an AVS47-IB or directly to a PC computer using the Picobus Cable “B” version. If you must use an older cable version, modify it to version “B” according to what was said on page 11.

If your system is equipped with the **Picolink Optical Fibre Link option**, the DB25 connectors on the rear panels of the AVS47-IB and AVS-47B are covered and they **must not be used** as long as Picolink is enabled. Picolink can be disabled by opening the bridge and pulling off the connector with 8 coloured wires from its socket on the Picolink transmitter/receiver board. Do the same with the AVS47-IB. Then the plastic covers of the connectors can be removed and the wire-Picobus can be used normally.

7. GETTING STARTED

> Use an older AVS47-IB manual together with resistance bridge versions AVS-47 and AVS-47A. They may require checking and changing Picobus power jumper positions. Older manuals can be downloaded from our WEB site www.picowatt.fi/downloads.

7.1. THE START-UP PROCEDURE

First, verify the mains voltage setting of the AVS47-IB, and change if necessary as described above. Connect the AVS47-IB to the AVS-47B by using the supplied **Picobus** cable version “B”.

If you have one, connect also a parallel printer (LPT) to the printer output of the AVS47-IB.

Switch on the AVS47-IB by connecting the power cord. There is no other mains switch, as the interface is designed to be powered always. The two green front panel LEDs will light up immediately, indicating that the unit is getting its +5V operating power. After a few seconds, the leftmost LED will blink three times. This indicates that the CPU has booted up succesfully, and that the unit is most probably OK.

If you cannot see the “READY” led blinking, pull off the power cord and try again. Push it in decisively. If you fail to maintain the power after the first contact, the computer may not succeed in booting properly.
The printer will print the following message:

```
AVS47-IB IEEE-488 Converter Program version 3R1
Copyright (c) by RV-Elektroniikka Oy Picowatt, 1993
Revised 2008-02-19

IB address: 20
IB status:  ADMC=0   MODE=I    CLR=0
Picobus address: 1
Picobus delay: 200
Eeprom write count: 11
```
(or something else)

The fourth line tells that you must use IEEE-488 device address 20, at least in the beginning, in order to communicate with the AVS47-IB. You can then change the address later, if desired.

The fifth line tells that the IEEE-488 card has been initialized correctly.

The sixth line specifies the Picobus Device Address, which is **always** 1 at power-on. Change it later if needed.

The seventh line specifies the default value of the Picobus Delay Factor. It needs no changing unless the Picobus cable is very long, or if RF filters with long time constants are used.

The eight line is the EEPROM write count. This figure is incremented each time when either the PBD or IBA command (not query) is issued. **Exceeding a write count of 10000 may lead to an unrepairable failure of the CPU.**

7.2. HYPHOTETICAL CONTROLLER LANGUAGE

As the variety of available controllers and their different language interfaces prevent us from using any syntax that would be applicable in all cases, we use a hyphotetical controller language, similar to that described in Appendix B2.3. of the IEEE-488.2 standard. The language contains following keywords:

RESET a — Interface Clear (IFC) is performed. A Device Clear (DCL) is sent. *RST common command is sent to device No. a.

SEND a; "data" Data enclosed in quotes is sent to device No. a. The controller is assumed to terminate the message automatically with NL^END (or LF^END).

RECEIVE a; var A <response message> is read from device No. a. The result is placed in the variable "var".

READ STATUS BYTE a; **spr** The IEEE-488.2 status byte is read from device No. a using serial poll. The result is placed in the numeric variable "spr" (serial poll response).

WAIT_SRQ The controller waits for a SRQ on the IEEE-488.1 bus. The program proceeds when SRQ is detected.

```
REPEAT
    BEGIN
    END
    UNTIL condition
```

The statements between BEGIN and END will be repeated until "condition" becomes true.

```
IF condition THEN
    BEGIN
    END
```

The statements beween BEGIN and END will be processed only if "condition" is true.

In order to avoid timeout errors etc., a controller shoud never try to address an GPIB device to talk if its output queue is empty. As will be discussed later, many AVS47-IB features, which are actually queries, have been divided into a command and a query. By waiting until the command has been executed, and making the query only after that, one can reduce the possibility of hanging up the whole bus. Many queries themselves also need up to tens of milliseconds time until the response is ready.

A simple Boolean function will be used later to point out the importance of waiting until the result is ready.

```
Function MAV:boolean
    BEGIN
        REPEAT
         BEGIN
              READ STATUS BYTE  a, spr
          END
              UNTIL (spr AND 16)=16 or timeout
          If timeout then MAV=false else MAV=true
    END
```

7.3. WRITING THE FIRST COMMANDS

Before proceeding, let us make the self-test first. Switch the AVS-47B off and on again so that it is surely reset. If you plan to use the TS-530, connect also it to the system (see the *TST ? description). Start the test:

```
SEND a;"*TST ?"
```

Look at the AVS-47B as its settings change during the test. In a couple of minutes the test is over, which is shown by the bridge when it goes to the reset state (REM led is blanked, range 2 MΩ and excitation 3 μV.

Read the response by

```
RECEIVE a; response
```

If everything is OK, the response should be 0. A missing TS-530A results in a minimum value of 96. The most important thing is that the response must not contain values 1 or 2.

Let us first set the bridge parameters and then take a reading. If desired, you can first reset the bridge by using the powerful "RESET a" equivalent of your own controller language, or just by

```
SEND a;"*RST"
```

Note that the default state of the AVS-47B after *RST differs from the power-on reset state. Range is now 2MΩ and excitation 3 μV. With these settings the bridge is fully operative. Try the following:

```
SEND a; "REM 1;INP 2;RAN 3;EXC 7;DIS 0"
SEND a;"ADC"
```

You must wait after an ADC command for half a second before trying to read the result. There are many possibilities to do that, and these will be discussed later. After the delay, issue

```
SEND a; "ADC ?"
IF MAV THEN RECEIVE a; adcresult
```

You will notice that the result is an integer. It could be converted to the sensor resistance by calculating $R = adcresult*10^{(range-5)}$. But there is an easier method, and it is to use the RES ? query. Instead of using ADC ?, issue

```
SEND a; "RES ?"
IF MAV THEN RECEIVE a; resistance
```

Instead of taking just one A/D conversion, you can increase the resolution by taking an average as follows:

```
SEND a; "AVE 20"
```

Wait for at least 20*0.4 seconds before proceeding.

```
SEND a; "AVE ?"
IF MAV THEN RECEIVE a; averesistance
```

7.4. THE ADVANTAGE OF HIGH-LEVEL COMMANDS

The purpose of the following example is to show, how the high-level commands make your programming easier. Suppose that you wish to measure three sensors in the cryostat. The sensors are connected to channels 3,4 and 5. Their predicted values are 37Ω, 1.95kΩ and 31kΩ, respectively. In order to prevent self-heating, the sensors must be measured using 100μV, 30μV and 10μV excitations. The reduction of accuracy on the low excitation ranges can be overcome by counting an average of e.g. 50 measurements. It is simplest to use the same average count for all sensors, even though it would not be necessary to have as many samples on 100μV as on the 10μV excitation.

```
EXAMPLE USING ONLY LOW-LEVEL COMMANDS

MUX[3]:=“MUX 3”;RAN[3]:=“RAN 3”; EXC[3]:=“EXC 4”
MUX[4]:=“MUX 4”;RAN[4]:=“RAN 4”; EXC[4]:=“EXC 3”
MUX[5]:=“MUX 5”;RAN[5]:=“RAN 5”;
EXC[5]:=“EXC 2”

SEND 20; “REM 1;INP 1;DIS 0;ARN 1”
J:=0
REPEAT
   BEGIN
   J:=J+1
   SEND 20;MUX[j]+“;”+RAN[j]+“;”+EXC[j]
   DELAY(15 sec)
   I:=0
   REPEAT
      BEGIN
         I:=I+1
         SEND 20;“ADC”
         DELAY(500 msec)
         SEND 20;“RES ?”
         IF MAV THEN RECEIVE 20; r
               r[j]:=r[j]+ r
      END
   UNTIL  i=50
   r[j]:=r[j]/50
UNTIL  j=3

results: r[1], r[2], r[3]
```

```
EXAMPLE USING HIGH-LEVEL COMMANDS

SEND 20; “REM 1;INP 1;DIS 0;ARN 1”
SEND 20; “FCH 3;LCH 5”
SEND 20; “SCP 3;RAN 3;EXC 4;CNT 10;SDY 10”
SEND 20; “SCP 4;RAN 4;EXC 3;CNT 30;SDY 10”
SEND 20; “SCP 5;RAN 5;EXC 5;CNT 50;SDY 15”
SEND 20; “*ESE 1;*SRE 32;*CLS”
SEND 20; “SCN 0;*OPC”
WAIT_SRQ
SEND 20;“SCR 3;AVE ?”
IF MAV THEN RECEIVE 20; r3
SEND 20;“SCR 4;AVE ?”
IF MAV THEN RECEIVE 20; r4
SEND 20;“SCR 5;AVE ?”
IF MAV THEN RECEIVE 20; r5

results: r3, r4, r5.
```

Line 2 specifies the first and last channels to be scanned

Lines 3-5 give the scan parameters separately for each sensor. Note that the average counts are different, because now it is easy to take into account the fact that there is also a tenfold difference between the excitations. Also stabilization delays are different, which reflects the fact that the AVS-47B is faster on higher excitations.

The *ESE 1 command enables bit 0 (operation complete) of the ESR register to be reflected by the ESB Event Status Summary bit of the Master Status Byte (MSB) register. The *SRE 32 command, in turn, enables the ESB bit to generate a Service Request. The *CLS command clears the event registers before the measurement.

A single scan cycle is started by SCN 0. The *OPC common command forces the AVS47-IB to assert bit 0 of the ESR register as soon as the cycle has ended.

After the service request has been detected, the results can be asked separately for each channel.

Comparison of these two programs shows how the high-level commands help one to write shorter and more readable programs. They also utilize the AVS-47B more effectively by encouraging one to use channel-specific average counts and stabilization delays.

And if you are smart enough to write interrupt-driven programs, you can use the time that the AVS47-IB needs for scanning for making something else with the computer.

Get acquainted with the high-level commands by reading the command descriptions and chapters on “Miscellaneous Topics”. Excercise by writing short programs that use those commands.

8. CONTROLLING THE INSTRUMENTS

8.1. COMMAND SYNTAX

All commands and queries are case-insensitive and they can be given using any mix of lower and upper case letters.

All <program data> for commands is numeric and integer. The data element denoting a query is a question mark (“?”). Almost all commands have corresponding queries. Data elements shall (but this is not necessary) be separated from the message headers by blank spaces.

Almost all program message headers consist of three letters, and they try to be mnemonic and easy to memorize. Those commands/queries that have been defined by the IEEE-488.2 standard as “IEEE-488.2 Common Commands and Queries” have a leading “*” character and they should behave as defined in the standard.

Response to the *IDN ? query is a string, but all other response data are numeric; either integers or floating point reals (like -1.2345E+03). By default, a response consists of a <response header> followed by the numeric data. The headers can be omitted by issuing the HDR 0 command. Then the response consists of plain numbers and separators (“;”) between them.

Two commands (TIM and DAY) need three <program data elements> and the BUF ? query returns two. In compliance to the IEEE-488.2, these elements are separated from each other by commas (DAY 1993,12,31).

One <program message> can contain more than one <program message unit>. In other words, one can give several commands in the same line. The <program message units> must be separated by semicolons (;). The last unit is followed by the <program message terminator>. As required by the IEEE-488.2, the AVS47-IB recognizes three terminators: A NL (newline, linefeed) character (ASCII 10 decimal), an ^END message on the last data byte, or both together. The response is always terminated by the combination of NL and ^END.

In most cases it is possible to use <program messages> that consist of only one command. An exception is scanning. Parameters for various channels can only be given or results asked if the SCP n or SCR n command is immediately (within the same message) followed by the desired commands/queries.

The length of the output queue is 255 bytes maximum. However, it can hold only one <response message> at a time. Read the buffer after every <program message> that contains one or more queries. Otherwise the output queue will be overwritten by the later response.

All commands are sequential in nature. This means that the previous operation is always fully completed before the next command or query is processed. If the command is, for example, AVE 50;INP 0 , all 50 samples will be taken before grounding the input. For this reason, the *WAI common command has no effect with the AVS47-IB.

You must note, however, that the sequential nature of the commands does not imply that the data is otherwise valid. If you give commands RAN 5;ADC;RES ? without any delay between the first two <program message units>, the A/D converter will take the reading before the self-balancing loop has stabilized. The length of the required delay depends on excitation, but is always several seconds. Similarly, DIS 1;ADC;ADC ? will record the difference ΔR before the A/D converter has settled. At least one second delay should be inserted as follows: DIS 1;DLY 1;ADC;ADC ?. Note also that neither the *OPC command nor the *OPC ? query can guarantee that the bridge is in balance. Use them only for telling that a sequence of operations, like a scan cycle, has been completed. And if time is needed for the bridge to stabilize or for the A/D converter to settle, insert a suitable delay (DLY x) before the *OPC / *OPC ?.

The following section is used to describe all the commands of the AVS47-IB. Some of the commands will also be discussed in the “Miscellaneous Topics” section. The notation REM [0..1 | ?] means the following: 0..1 is the allowed range of the argument (or <program data> for the command. “|” denotes an alternative, which is usually a question mark that indicates a query.

Those operations that require a long time have been divided in two parts: a command and a query. For example AVE 100 needs 40 seconds to complete. You can use this time for other things and tell with a Service Request to the controller when the AVS47-IB is ready. The you can use the AVE ? query to get an immediate result, without hanging the IEEE-488 bus for tens of seconds.

Each command name is followed by examples on a command (c), query (q) and a typical response (r).

8.2. BASIC AVS-47B COMMANDS

REM [0..1 | ?]
Local/Remote Mode Command / Query
c:REM 1 q:REM ? r: REM 1

In the **local** mode you can control the AVS-47B using its front panel switches, just as if there were no computer interface. You can read data and all settings via the interface, but you cannot change them. Many macro commands are available also in the local mode: they change the bridge automatically into remote, returning it back to local after the macro has been completed. Especially, you cannot use the ARN 1 autoranging command. If, for some reason, you need autoranging while in local, use the AVS-47B's own AUTO mode.

The front panel controls are disabled in the **remote** (REM 1) mode. This does not apply to the hard-wired switches: Reference Source, Difference Magnifier and the Reference Potentiometer. The RFS, MAG and DIS commands provide means to read the positions of these switches.

INP [0..2 | ?]
Input Source Command / Query
c: INP 2 q: INP ? r: INP 2

INP 0 connects the bridge input to ground. Use this position for determining any possible offset.

INP 1 enables the actual measurement. The sensor channel is determined by the MUX setting.

INP 2 connects the bridge to an internal 100Ω precision reference. Use this position for calibrating the scale factor.

It is a recommended practice to set INP 0 before changing other bridge settings. This prevents the bridge from supplying excessive current to the sensor, and it prevents unintentional overload if the range happens to be momentarily too low.

MUX [0..7 | ?]
Multiplexer Channel Command / Query
c: MUX 3 q: MUX ? r: MUX 3

Unlike with the old model AVS-46, all input channels of the AVS-47B are multiplexed and exactly similar.

Excercise some care when changing the input channel in order to avoid extra heating of your sensors. First connect the input to ground (INP 0). Then select the new range and excitation, and after a couple of seconds enable the input again. One way to do this is to use the SCAN macro commands.

RAN [0..7 | ?]
Range Command / Query
c: RAN 5 q: RAN ? r: RAN 5

Do not use range 0, because this prevents the AVS-47B from stabilizing. (This setting means that no range is connected. The AVS-47B powers on with RAN 0, so that it could not heat the sensor before the proper range and excitation have been selected).

EXC [0..7 | ?]
Excitation Command / Query
c: EXC 1 q: EXC ? r: EXC 1

Excitation 0 can be safely used, but often it is more convenient to use EXC 1 instead of EXC 0. Then the bridge will work properly and show reasonable, although noisy, output.

DIS [0..7 | ?]
Display Selector Command / Query
c: DIS 0 q: DIS ? r: DIS 0

The display items are in the same order as on the AVS-47B front panel (DIS 0 = R etc). Insert a delay of about 1 second after having selected a new display item, before taking the reading.

REF [0..20000]
Set the Reference Command
c: REF 12345 q: ----- r: ------

This command has exactly the same effect as dialling the potentiometer to the desired reading and then activating the SET REF switch.
The argument value is internally rounded to the nearest number divisible by 5.

There is no direct query for reading the digital reference value. Instead, use DIS 3;DLY 1; ADC;ADC ? for reading the actual reference voltage.

If the Reference Source switch is in the REF POT position, the REF command has no effect at all.

RFS ?
Reference Source Query
c: - - -   q: RFS ?   r: RFS 0

Response RFS=0 means that the reference for the difference output is taken via the digital reference memory. There are two ways to program a value into this memory: The REF [] command, and by activating the SET REF switch. The latter method saves the current display into the memory, regardless of what was the item being displayed.

Response RFS=1 means that the reference is taken directly from the potentiometer, bypassing the memory.

MAG ?
Magnifier Query
c: - - -   q: MAG ?   r: MAG 0

The magnifier has effect only in the deviation display mode. **0** means ΔRx1 and **1** means ΔRx10.

You do not need this command, if you use the RES ? query to read the difference, because RES ? takes the magnifier automatically into account, whereas ADC ? does not.

The AVE and SCN commands also read the magnifier switch automatically.

Note that the AVS-47B decimal point is moved, besides in the difference mode, also when reading the resistance, although the display is then not magnified. This is a limitation of the hard-wired electronics of the bridge. Readings recorded via the computer interface are correct in any case.

ARN [0..1 | ?]
Autorange Command / Query
c: ARN 1   q: ARN ?   r: ARN 1

Autoranging by software can be used only when the hardware autoranging of the AVS-47B is disabled and the bridge is in the Remote mode.

ARN 1 enables autoranging. Whenever the ADC result falls below 1800, range is changed downwards until either a valid reading is obtained or range 1 is reached. Then the reading is taken anyway. Whenever the ADC result exceeds 19999, range is changed upwards until either a valid reading is obtained or range 7 is reached. Then the reading is taken anyway.

A stabilization delay SDY is automatically inserted between the successive autoranging operations. The program also waits for SDY seconds before taking a reading after having changed the range. You can control this delay by using the SDY command.

In the autorange mode, the reading will tell about overload only if the resistance exceeds 1.9999MΩ. The ADC ? query will then return 20001 and the RES ? query returns 2.0001E+06. The OVL ? query returns OVL 1.

If such a range is found that a non-overloaded reading can be taken, the OVL bit is cleared. You may want to use the RAN ? query to tell the range that was finally used.

ARN 1 works for the ADC, AVE and SCN commands. It is especially recommended for the latter two, because it is the easiest way to guarantee that the results are valid and non-distorted.

ARN 0 disables autoranging. The ADC command is easy to use also without autoranging, as you need only to compare the result with 20001 or 2.0001E+6.

If you use AVE without autoranging, include the OVL ? query in your program. If any one of the samples to be averaged results in overload, the OVL bit will be set. Do not use such averages, they are distorted and not reliable.

Scanning without autoranging is possible, but dangerous. When ARN=0, the firmware program uses those ranges that were specified using the SCP x; RAN y commands separately for each channel. As there is only one OVL bit, there will be no way to detect a past overload after having proceeded to a new channel.

Use the SDY command to set an overall autoranging delay. The program inserts this delay between two successive autoranging operations when you use either the ADC or AVE command. It is also the delay after the range change, before taking the reading.

Because various sensors may be **scanned** using very different excitations, SDY can be specified separately for each sensor (SCP x; SDY y) to be scanned.

SDY [1..100 | ?]
Stabilization Delay Command / Query
c: SDY 20   q: SDY ?   r: SDY 20

Stabilization delay allows the AVS-47B to find balance between two successive autoranging operations, or between autoranging and taking the reading.

Default delay after power-on is 15 seconds.

Stabilization delays that are used when **scanning** can be specified separately for each channel.

Note the difference between the SDY and DLY commands. SDY is used to set the length of a delay which is used automatically in connection with autoranging. DLY is a delay that you insert whenever desired between any two commands or queries.

ZDR
Zero Difference Command
c: ZDR q: ---- r: ----

This command sets the AVS-47B in Remote mode and switches the Display Selector to 0. After a delay it takes a reading. This reading is programmed into the Reference Memory. The original display item is selected, and finally the AVS-47B returns back to local (if it was in local in the beginning).

The resulting ΔR will not be ecactly zero for two reasons: The reference D/A converter has only 12 bits and therefore the reading must be rounded. This explains a deviation of max. 3 digits from zero. The difference is formed by an analog amplifier, which has only moderate stability. This is an important factor especially in the 10xΔR display mode.

ADC, ADC ?
A/D Conversion Command / Query
c: ADC q: ADC ? r: ADC 9876

The AVS-47B has a free-running A/D converter. In order to guarantee that the reading is up-to-date, this command uses the AL (Alarm) flag which is provided by the **Picobus.** Normally, generation of the AL signal is disabled. The ADC command enables the flag and waits until the completion of the next A/D conversion sets it. Then the reading is read from the converter. As the converter takes 2.5 readings in a second, the ADC command needs a time that can be 400 milliseconds at most.

(An exception to this rule is the reading 0000. When overloaded, the A/D converter outputs an exact zero together with a blinking overload signal. In order to separate the overload-zero from a true zero input voltage, the AVS47-IB records several successive readings and inspects whether the overload signal is blinking or not. Therefore the time needed for converting a exact zero is much longer than for other input voltages.)

The ADC ? query should be used after the ADC command. If it follows immediately, it will hang the bus for <400 milliseconds. There are many ways to prevent this. The AVS47-IB provides the DLY delay, serial polling and generation of a Service Request by using the *OPC command or *OPC ? query.

The ADC ? result is a signed integer from -19999 to 19999. Overload is coded to 20001. You can also detect overload by using the OVL ? query. The lowest digit means 100 μV, except in the 10xΔR display mode, when it means 10 μV (referred to the AVS-47B analog output).

RES ?
Resistance Query
c: ----- q: RES ? r: RES 1.2345E+04

Like the ADC ?, this query is used once the ADC command has first been executed. It calculates the sensor’s resistance by taking the measurement range into account. The result is a real number with a 5-digit mantissa. The smallest digit is always zero for the RES ? query (it exists for the more accurate readings obtained by averaging).

You can also use the RES ? query when in the 10xΔR display mode. The program takes the higher magnification into account.

Do not, however, use RES ? for reading display items like TS-530 set point or heater voltage, where the resistance range should have no effect.

AVE [1..1000]
Measure Average Command
c: AVE 20 q: ----- r: -------

This command calculates an average (or mean) of n successive A/D conversions. Each conversion requires 0.4 seconds. Sampling can be stopped by issuing the **STP** command.

Completion of the averaging process can be signalled by enabling either the *OPC command or the *OPC ? query to generate a service request. Another possibility is to ask the master status byte by serial polling (see later). The serial poll response tells about the state of the AVS47-IB.

Four queries are available for asking the results after the samples have been taken.

AVE ? returns the calculated average (AVE 1.23456E+02),

MIN ? returns the smallest observed value,

MAX ? returns the largest observed value and

STD ? returns the standard deviation of the samples. STD is equal to the rms noise in the output.

You can take averages as well in the local as in the remote control mode, and also with or without autoranging. In the last case, verify the validity of the result using the OVL ? query. If ARN=1 and REM=1, then the AVS47-IB monitors the readings and changes range if necessary. The average count is nulled and sampling starts again from the beginning.

OVL ?
Overload Query
c: ------ q: OVL ? r: OVL 0

The overload flag is reset when starting to execute an ADC command , and it will be set if the command encounters overload.

The AVE command uses the ADC command repeatedly. The OVL flag is OR'ed with its prceding values. Therefore, if at least one sample is overload, the query will return 1, otherwise 0. See also description of the SCAL command.

Use OVL ? routinely if you take averages without autoranging. Use it also together with **digital calibration.**

In the **autorange** mode, the overload flag is reset after each autoranging operation. You will never see OVL 1 in this mode, unless the resistance exceeds 2 MΩ.

Do not rely on OVL ? when scanning. Scan only with ARN 1.

8.3. TS-530A COMMANDS

The TS-530A Temperature Controller must be connected to the AVS-47B using an one-to-one 37-way ribbon cable having a DC37P connector at each end. Remote control is possible only when also the AVS-47B is in the Remote mode (REM 1).

The design of the TS-530A does not permit reading of the PID parameters to the AVS47-IB. Some commands, however, require that the interface knows the state of the temperature controller. This problem has been solved so that all PID parameters are updated even though only one parameter is being changed. This principle has an important consequence:
The AVS47-IB remembers all the parameter settings given for the TS-530A. Even if it is not possible to ask these settings from the controller, they can be asked from the interface. These queries may be useful if the computer or the control program has been off and one wants to restart remote control without introducing any changes in the system state.

If you want to control any one aspect of the TS-530A remotely, you must control them all!

SPT [1..42000 | ?]
Set Point Command
c: SPT 11500 q: SPT ? r: SPT 11500

This command specifies the TS-530A Set Point Voltage. The unit is 100 μV.

Read the **analog set point** using the SPV command and the SPV ? query.

PRO [0..11 | ?]
Proportional Gain Command
c: PRO 5 q: PRO ? r: PRO 5

Refer to TS-530A manual for the actual gain values.

PRO 15
Force Error Signal to Zero Command
c: PRO 15 q: --- r: ---

Forcing the error signal to zero prevents the controller from supplying excessive power when the bridge settings are being altered or the sensor channel is changed. The SCN procedure uses this command extensively.

ITC [0..11 | ?]
Integrator Time Constant Command
c: ITC 8 q:ITC ? r: ITC 8

Refer to TS-530A manual for the actual time constant values.

DTC [0..7 | ?]
Derivator Time Constant Command
c: DTC 3 q:DTC ? r: DTC 3

Refer to TS-530A manual for the actual time constant values.

Scanning is best made without the derivator, because the derivator will amplify all spikes resulting from switching the error signal.

BIA [0..5 | ?]
Power Bias Command
c: BIA 0 q: BIA ? r: BIA 0

Refer to TS-530A manual for the actual bias power values.

POW [0..7 | ?]
Power Range Command
c: POW 7 q: POW ? r: POW 7

Refer to TS-530A manual for the actual power range values.

SPV, SPV ?
Measure Set Point Voltage Command / Query
c: SPV q: SPV ? r: SPV 1.1503E+00

The command sets the display item to 7, inserts a delay and then takes a reading. Then the bridge is returned to its original state.

The response to SPV ? is the set point voltage in Volts, given as a floating point number.

HTV, HTV ?
Measure Heater Output Voltage Command / Query
c: HTV q: HTV ? r:5.3018E+00

The command sets the display item to 5, inserts a delay and takes a reading. Then the bridge is returned to its original state.

The response to HTV ? is the voltage across the heater in Volts. The number is only an estimate: the voltage is amplified strongly on the lower POW ranges using a coarse analog amplifier.

HTI, HTI ?
Measure Heater Output Current Command and Query
c: HTI q: HTI ? r:5.5028E-02

The command sets the display item to 6, inserts a delay and takes a reading. Then the bridge is returned to its original state.

The response to HTI ? is the heating current in Amperes. Use this figure as an estimate only.

HTP, HTP ?
Measure Heater Output Power Command / Query
c: HTP q: HTP ?r:2.9108E-01

The command combines the two preceding commands for calculating the true heating power.

Note that execution takes two times longer than the HTI and HTV commands alone.

The result to HTP ? is the output power in Watts.

8.4. DIGITAL FILTER COMMANDS

DFL [1..1000 | ?]
Filter Length Command and Query
c: DFL 100 q: DFL ?r: DFL 100

The digital filter maintains an average of n last measurement results. Number n is set by the DFL command. Using a small n makes the filter faster but less effective. A filter with large n rejects noise more effectively, but its response is slow.
If the noise could be assumed to be “white”, making the filter four times longer should improve the resolution by two. Unfortunately, this is not the case with the AVS-47B. Its noise is somewhat “red” (noise spectral density increases when the frequency decreases). Therefore the improvement is smaller than expected.

The default filter length is 50.

DFS
Start Digital Filter Command
c: DFS q: ----- r: -----

The digital filter is limited to maintaining results of only one input channel in one circular queu. After having started the filter with DFS, only two commands are available:

- **DFR ?** Get Filtered Reading Query. This corresponds to the RES ? query.
- **STP** Stop the filter.

The filtered reading cannot be buffered nor printed. If those features are needed, use AVE or SCN. The digital filter allows you to read up-to-date results to the computer quickly without need to wait for an average.

Read more about the digital filter in the “Miscellaneous Topics” section.

8.5. SCANNING COMMANDS

Scanning is possible with or without using a TS-530 Temperature Controller for simultaneous control. Read more about scanning in the “Miscellaneous Topics” section.

The commands fall in four categories: setting overall parameters like scanning interval, setting measuring parameters separately for each channel, starting to scan and asking the results.

8.5.1. SETTING THE OVERALL PARAMETERS

FCH [0..7 | ?]
First Scan Channel Command / Query
c: FCH 1 q: FCH ? r: FCH 1

If temperature control is enabled, the control channel **shall not be included in the scan sequence**. For example, if you use channel 3 for control, the range FCH..LCH must not include 3.

LCH [0..7 | ?]
Last Scan Channel Command / Query
c: LCH 3 q: LCH ? r: LCH 3

The last channel must not be lower than the first one. The sensors are measured in order from FCH to LCH. Do not leave blank channels between them.

Setting FCH=LCH makes it possible to program a scheduled measurement of one sensor.

SCI [0..10000 | ?]
Scan Interval Command
c: SCI 60 q: SCI ? r: SCI 60

Scan Interval is the time from the beginning of one complete scan cycle to the beginning of the next cycle. Default is 600 (seconds).

SCI must be longer than the time needed for the measurements. Otherwise the next cycle will start immediately after the first one.

You can measure the minimum time needed for a scan cycle as follows:

```
*ESE 1;*SRE 32;*CLS
TIM ?
IF MAV THEN RECEIVE 20; time1
SCN 0;*OPC              {single  cycle}
WAIT_SRQ
TIM ?
IF MAV THEN RECEIVE 20;time2
scantime=time2-time1
```

The scan interval is a meaningful quantity only in the continuous-scan mode (SCN 2).

ETC [0..1 | ?]
Disable/Enable Temperature Control Command / Query
c: ETC 1 q: ETC ? r: ETC 1

ETC 1 tells to the AVS47-IB that the channel, which was selected before starting to scan, is used for temperature control, and it will be handled in a special way. It also tells that the program must turn to this original sensor in between all channels to be scanned and also after the scan cycle has been completed. The scan cycles are described in detail in 8.5.3. and in the “Miscellaneous Topics” section.

ETC 0 is the default. Then all channels are equal and scanning proceeds from one channel to the next without breaks.

TCP [0..1000 | ?]
Temporary Control Period Command / Query
c: TCP 120 q: TCP ? r: TCP 120

TCP determines the time, in seconds, that is used for temporary control between various channels. It does not affect in any way to the controlling time between two scan cycles, as this time depends only on the difference between SCI and the actual time needed for scanning.

For best results, use a reasonably long TCP so that the control system is in good equilibrium when the next channel is measured. Default is 60 seconds.

8.5.2. SETTING THE CHANNEL PARAMETERS

Parameters for each scan channel are set and queried using a multi-command structure: the first command specifies the channel number in question, and the succeeding commands set the parameter values for that channel.

SCP [0..7 | ?]
Set Channel Parameters Command / Query
c: SCP 3 q: SCP ? r: SCP 3

This command must start a new line. It specifies the channel number for which the following parameter values shall apply.

The purpose of the SCP ? query is to make documentation easier. Suppose that you want to print settings for channel No. 5. Your command would then be

SCP 5;SCP ?;RAN ?;EXC ?;SDY ?;CNT ?

The SCP 5 command tells to the program, about which channel you are asking. As a command, it does not produce any output. But with the SCP ? query, the output line would be e.g.

SCP 5;RAN 4;EXC 2;SDY 20;CNT 30

This is exactly the same form as used also for setting those parameters. Before the next <program message terminator> after SCP n you can give one or more of the following commands:

- **RAN [0..7]** for range.
- **EXC [0..7]** for excitation.
- **SDY [0..100]** for stabilization delay after autoranging and
- **CNT [1..1000]** for the average count.

In case of **manual ranging** (ARN 0), the specified range will be used for a given channel regardless of a possible overload. In case of **autoranging** (ARN 1), the AVS47-IB starts to take readings on the specified range, but changes it in case of overload or underrange. The final range that gave valid readings replaces the original RAN, and will be used as the starting point for the next scan cycle. Default range for all channels is 7. In ARN 1 mode, it is not necessary to specify RAN at all, if you have time to wait for some autorangings in the beginning.

Excitations are not changed automatically, because there is no common rule how this should be made. The default excitation for all channels is 1 (3 μV).

Stabilization delay is the same as described earlier. Default delay is 15 seconds. It should be long enough for all but the lowest two excitations.

The default average count is 10. In case of an autorange action, the average count is nulled and after an SDY sampling is started again from the beginning.

The channel parameters are queried in a similar way. The SCP [n] command must start a new <program message>, and it can be followed by one or more of the following queries: SCP ?, RAN ?, EXC ?, SDY ? and CNT ?.

Note that while all other queries return those values that were given to the interface earlier, the RAN ? returns the latest range setting for a given channel. This may differ from the original range if there has been need to change range and ARN=1.

Note also that if ETC=1 (temperature control is enabled), the channel that was selected before starting to scan, will be measured using manual ranging regardless of the ARN setting.

8.5.3. STARTING TO SCAN

There are three different ways to scan: a single cycle, continuous scanning and front-panel started scanning.

SCN [0..2]
Start to SCaN Command
c: SCN 0 q:-- r:--

SCN 0 command starts a single scan cycle. Upon completion of the cycle, the interface returns to the idle state, and you can ask the results immediately (See below: "Results in the SCN0 mode"). This mode is recommended when the interface is used under the supervision of a controlling computer. The completion of a cycle can be indicated with the aid of the *OPC command or *OPC? query. One can also use serial polling in order to detect the return to the idle state.

If temperature control is enabled (ETC 1), the original sensor is used for control as described earlier. Scan cycle starts from channel FCH. Control is disabled during scanning, and it is enabled again once the cycle has been completed. Temperature is **also** controlled between each channel. This temporary control period is determinded by the TCP parameter. If you do not want control between the channels, set TCP=0.

Use the **STP** command to stop scanning, if you have to do so before the cycle has ended. In such a case, do not use the results - they are unpredictable.

SCN 1 command starts the front-panel effected scan mode. The first scan cycle is made immediately after the command. But then, instead of returning to the idle state, the interface starts to wait until the front-panel START/PRINT switch is activated. As soon as that happens, a new cycle begins.

Also this mode is exited by issuing the **STP** command. There may be some phases in the program when the STP command does not work, and is forgotten. Therefore it is a good practice to serial poll the status byte in order to verify that the system has really entered the idle state, and if not, just repeat the STP command. It is best to exit this mode while waiting for the button (status=5). Then all results are sure to belong to the previous completed cycle. If you exit while actually scanning, the results can be unpredictable.

Front-panel started scanning is useful when the operator knows better than the computer, when to take readings. It is also possible to use the computer for other purposes after setting the scan parameters and starting the mode. The results can be read later from the RAM buffer. Use the front-panel started scan mode e.g. to save paper in applications where the readings are printed on-line after each scan cycle.

Temperature control (if effective) is disabled when the switch is activated. In the end of the scan cycle, the system returns to measure the original sensor and enables control again. This phase continues until the switch is activated again. Temperature is controlled **also** between the various channels. The length of this temporary control period is determined by the TCP parameter, which can be set to 0 if you do not want temporary control.

SCN 2 starts the continuous-scan mode. The scan cycles are automatically repeated at a fixed scan interval SCI. This mode is useful for scheduling unattended measurements during night etc.

The first scan cycle is made immediately after the SCN 2 command. Upon completion, the system waits until time equal to SCI has elapsed from the beginning of the first cycle, before the second cycle is started. And so on.

Exit the mode with **STP** before reading the results. It is best to exit while waiting for the next cycle to begin (you can verify the phase by serial polling, state=3). STP can also be issued during actual scanning, but then you must be careful when reading the results: the last cycle can be documented only partially.

The results can be stored in the RAM buffer (if DSK=1), and/or they can be printed on-line (if PRN=1). These are the only two possibilities to save the readings. After having programmed the scan parameters, the host computer is no longer needed. However, you must remember to read the buffered results before the power goes off or before the buffer overflows (reset the buffer, too).

Temperature control, if it was effective before the SCN 2 command, is disabled in the beginning. After channel No. FCH it is enabled (having first selected the original control sensor) for time period equal to TCP. Then control is disabled again and channel No. FCH+1 is measured. And so on. After the last scan channel, the original sensor is selected and control is enabled. This final active-control phase lasts until time equal to the SCI parameter (scan interval), counted from the beginning of the previous cycle, has elapsed.

Note that SCI must be rather long: it must be longer than the total time needed for all measurements, temporary controls, stabilisation delays and some other shorter delays needed when the channels and excitations are switched. It is best to make first some SCN 0 cycles and measure the minimum required time (See 8.5.1. for an example).

NOTE: We strongly recommend that you use the autoranging ARN 1 mode while scanning. The program has only one overload flag which is common for all channels. There is no way to detect a past overload after having measured a new channel. The autorange mode guarantees that the results are always valid. An important exception is original channel, if temperature control is enabled (ETC 1). Then autoranging is not applied for this channel. You are responsible for selecting initially a safe range for the control channel that will not be exceeded during the experiment.

8.5.4. RESULTS IN THE SCN 0 MODE

If you want to ask the scanning results via the bus immediately after the cycle, then use the SCN 0 mode. The SCN1 and SCN2 mode are intended for applications where results are either printed on-line, or stored in the buffer for later reading.

Before proceeding, verify by serial polling or SRQ generation that the SCN 0 cycle has ended.

SCR [0..7 | ?]
SCan Results Command / Query
c: SCR 4 q: SCR ? r: SCR 4

SCR [0..7] must be the first command in a new <program message> line. This **SC**an**R**esults command specifies the number of the channel for which you are going to ask the results. Being a command, SCR n does not produce any output.

SCR ? query helps documentation. Use this query immediately after the SCR n command so that you get the <response message unit> “SCR n” in the beginning of the response message. Then it is easy to realize later that the line means **SC**an **R**esults for channel **n**.

After having specified the channel number, you can query the same items as for the AVE command. All queries for channel **n** must follow within the same program message, before the next message terminator.

AVE ? returns the calculated resistance as a real number (AVE 1.23456E+04).

MIN ? returns the smallest observed resistance,

MAX ? returns the highest observed resistance and

STD ? returns the calculated standard deviation of the samples.

If you want also the range, exctitation or some other parameter to appear in the output message, you can include the corresponding queries. For example, query:

SCR 2;SCR ?;TIM ?;AVE ?; RAN ?
could return
SCR 2;TIM 11,05,56;AVE 1.23456E+04;RAN 5

If you have enabled temperature control by ETC 1, then the AVS47-IB measures the control channel before starting to scan (see “Miscellaneous Topics: Scanning”);

8.5.5. RESULTS IN THE SCN1 AND SCN2 MODES

It is possible to print and/or buffer the results of the individual scan cycles for later reading or printing. While this is true for all three scan modes, buffering and printing are useful mainly in the SCN1 and SCN2 modes, where the results cannot be queried before the mode has been exited.

The following two commands control saving of the results:

DSK [0..1 | ?] Turns the RAM disk buffer off/on..
PRN [0..1 | ?] Turns on-line printing off/on.

The printed and buffered output lines are identical to those of the Front-Panel Started measurement. You cannot change the output format. If you wish to include more items, or if you want the items to appear in a different order, then use the SCN0 mode and ask the results after each cycle as was described above. **Do not turn printing on unless you have a printer connected and ready.**

Scan results for the first two channels could look as follows when printed on-line:

```
1993-12-06   4:37:24   1.0006E+02   1 0 3 7 0
1993-12-06   4:38:02   1.3450E+02   1 1 3 7 0
```

The printed format deviates from the IEEE-488.2 conventions for better readability. The last five digits mean the input switch position, channel, range, excitation and display item, respectively. Note that these parameters are in the same order as on the AVS-47B front panel. In case of an overload, the data is followed by a remark “dvm overload”.

Data can be stored also in the buffer when DSK=1. Then the IEEE-488.2 conventions are followed. If the previous lines were read via the bus from the buffer (after first exiting the scan mode), they would look as follows:

DAY 1993,12,6;TIM 4,37,24; RES 1.0006E+02;INP 1; MUX 0; RAN 3; EXC 7;DIS 0;OVL 0

DAY 1993,12,6;TIM 4,38,02; RES 1.3450E+02;INP 1; MUX 1; RAN 3; EXC 7;DIS 0;OVL 0

Data is read from the buffer using the RBF, NXT and STP commands, as will be described later.

You can also print the buffer contents using the PRT command. See “Printer” and “Buffer”.

9. SETTING THE TIME AND DATE

The AVS47-IB has a real-time clock. However, this clock is not battery-backed. Therefore you must first set the time and date in order to include useful time stamps in the results.

DAY [year,month,day | ?]
Set the date command / query
c: DAY 2001,2,26 q: DAY ? r:DAY 2001,2,26

Note that this command needs three arguments, separated by commas. The program makes some simple checks regarding the validity of the date. If the date is invalid, the program does not complain, but the date may remain unset. It is a good practice to check the new date using the DAY ? query.

Power-on default date is DAY 0,0,0. This can be used to show experiment time. The clock can be reset to this value using command DAY 0,0,0. The AVS47-IB’s own clock is not adjusted, but an offset is added to the current date to result in zeros. You can restore the original date by using an argument that is invalid, for example: DAY 1 works because there are no three data.

Note that if you use the above restoring after power-on, the date is meaningless because the clock is not battery-backed.

TIM [hour,minute,second | ?]
Set the time command / query
c:TIM 16,5,12 q: TIM ? r:TIM 16,05,20

Just like above, this command needs three integer arguments (<program data elements>). The program makes some simple checks regarding the validity of the time, but it does not complain if the time is invalid. In such a case, the clock may remain unset. It is a good practice to check the clock using the TIM ? query.

The power-on default setting of the clock is TIM 0,0,0.

10. THE BUFFER

The AVS47-IB has a 64 kB RAM disk buffer for temporary storage of measurement results. The buffer is not battery-backed, so it will lose its contents in case of a power failure. The buffer can be used in several ways, but the last line is always appended to it, i.e. added after the last previous data line.

For simplicity, the buffer can be read or printed only sequentially starting from the beginning. Therefore you may want to clear the buffer from unnecessary data before a new experiment.

The AVS47-IB creates separate buffers for each Picobus Address PBA that you select. So if you have two resistance bridges, both use their own RAM disk files. The current value of the PBA parameter identifies the buffer so that you need not care about it. In other words, if PBA=2, all results are appended to the second file, and also the reading and printing commands apply to the second file. Use the PBA ? query with HDR1 to insert an identification in the buffer file or printout.

NOTE: After power-on, the front-panel started measurement uses the default buffer for PBA 1, even though your bridge might have been set for a different address.

The RAM disk capacity is sufficient for appr. 20 A4-size papers filled with data, but a possible overflow is not detected in this program version 3R1. Therefore, avoid overflow. The interface may hang up and you lose all your data. Use the BUF ? query for investigating the used and remaining RAM disk space.

DSK [0..1 | ?]
Enable RAM DiSK storage command / query
c: DSK 1 q:DSK ? r: DSK 1

The DSK command is used to disable or enable buffering. Neither of these operations clear the previous data, you need to use a separate command for that.

The buffer is disabled by default after power-on. However, results from a front-panel started measurement are always buffered, regardless of the DSK setting. Separate buffers are created for different Picobus addresses.

If the buffer is enabled, all responses to your queries are saved. One <response message> makes one line in the buffer. If your message to the AVS47-IB is, for example:

MUX ?;ADC;ADC ?;RAN ?;EXC ?;TIM ?

the buffered line could be:

MUX 4;1234;RAN 4;EXC 3;TIM 11,02,56

Note that the response consists of only 5 <response message units>, because the ADC command does not produce output.

Scanning produces lines of predetermined format, which you cannot alter (see Scanning commands above).

RBF
Read Buffer File command
c: RBF q:-- r:--

Before reading the results from the buffer, you must set the interface into the Read Buffer File mode using the RBF command. If necessary, check by serial polling whether the AVS47-IB is in the RBF mode (the serial poll response is 8).

You can exit this mode in three ways: read all records in the buffer, issue the STP command, or issue the *RST common command (which will reset the AVS-47B and the optional TS-530, too).

After the RBF command, the records can be read.

NXT ?
NeXT Record Query

You must issue the NXT ? query even for the first record. The response to this query is the next <response message> line in the buffer. You can verify by serial polling that a message is in the output queue before trying to read it via the bus (bit 4 of the serial poll response is 1)

After having read the first response line, issue the NXT ? query again. Check the serial poll response (spr) in order to verify that a new message is available, and read the line. Continue this until bit 4 of the spr stays at zero, indicating that the buffer has no more records. The interface returns automatically to the idle state.

The following example shows how to read the buffer:

```
SEND 20;"*CLS"                          clear status
SEND 20;"RBF"                           enter read buffer mode
REPEAT
    BEGIN
        SEND 20;"NXT ?"                 proceed to the next line
        READ STATUS BYTE 20; spr        serial-poll the status byte
        IF (spr AND 16)=16 THEN         bit 4=16 tells if message is
            RECEIVE 20; dataline        available, if so, read the
line.

        do something useful with the dataline

        READ STATUS BYTE 20; spr        spr goes to 0 when the last
    END                                 record is read
UNTIL spr=0
```

KIL
Delete the Buffer File command
c:KIL q:-- r:--

The KIL command clears the RAM disk buffer of the currently selected PBA. Use this command with caution; there is no way to recover a deleted buffer.

NOTE: Be careful not to activate the START/PRINT switch if you are storing other results to the buffer. If you do so, the buffer will contain records of different number and order of fields and it will be difficult to read.

If you encounter this problem, disable the front-panel started measurements by using the FSM 0 command.

BUF ?
Buffer Size Query
c:-- q:BUF ? r: BUF filesize,diskfree

This query returns information on the RAM disk buffer. The BUF ? query applies to the currently active Picobus Device Address PBA.

The response consists of two <data elements>. The first is the size of the buffer file in bytes. The second is the free space that remains from the initial 64 kB RAM disk. One scan-result line occupies 70-80 bytes depending on time and date.

Use the BUF ? query to prevent buffer overflow. The program does not generate an SRQ to warn about overflow.

Size of the empty RAM disk is over 62 kB.. If no buffer exists , the response is 0,0. (E.g. because no buffer has been created for the currently active PBA, or because nothing has been stored to the buffer since power-on or after the last KIL command).

11. PRINTING

The AVS47-IB offers a parallel (Centronics) printer port. An optional printer can be used basically in seven ways: 1) It can print the response to all queries on-line, 2) It can print results from scanning on-line, 3) It can print the contents of the buffer, 4) It can document the front-panel started measurements, 5) It can print detailed results from a self-test, 6) It can print initial messages after power-on and 7) it can print some error messages.

PRN [0..1 | ?]
Enable PRiNter Command / Query
c:PRN 1 q:PRN ? r:PRN 1

Do not enable printing unless you have a printer connected and ready. The printer is off by default after power-on, but the program should not hang up even if printing is tried without a printer.

As an exception, the AVS47-IB tries to print the results from a front-panel started measurement regardless of whether the PRN is 1 or 0.

If the printer is enabled, it will print all responses to queries. You can determine the items that appear in the printout by the arrangement of the queries. For example, the following sequence can be used to produce only printed output:

```
SEND 20;"PRN 1"
SEND 20;"RAN ?;EXC ?"
SEND 20;"PRN 0"
```

The print format and items list is the same for scanning and front-panel started measurements, and they cannot be changed. See 8.5.5.

PBF
Print the Buffer File command
c:PBF q:-- r:--

This command initiates printing of the buffer file, line by line and starting from the beginning. It requires that printing has been enabled (PRN 1;PBF).

You can stop printing by the STP command. Note, however, that all data may have been already transferred to the printer's own buffer, and then the STP command has no effect.

Printing the results from scanning does not need any commands. The only requirement is that the printer is enabled (PRN 1). Printing results from the Front-Panel Started measurement does not need even that. The format of the printout is described in the appropriate sections.

You can also print the contents of the buffer by keeping the START/PRINT switch lifted for more than 3 seconds. See 16.1. for more information.

12. STATUS REPORTING

12.1. COMMANDS

All status reporting conforms to the IEEE-488.2 standard, except the use of the status byte. Four lowest bits of this byte, whose purposes have not been predetermined by the standard, are used to specify the system's status in the serial poll response.

The register structure consist of one dynamic register (ESR, Standard Event Status Register), enabling mask for this (ESE, Standard Event Status Enable Register), Service Request Enable Register SRE and the Status Byte STB. The Status Byte is maintained in two registers, which are otherwise identical, except that the first one is read by serial polling (we will call it the "serial poll response", **spr** register), and the second is read by the IEEE-488.2 defined common query *STB ? (STB register). Bit 6, which indicates request for service, is reset when the spr register is read by serial polling, whereas bit 6 of the STB register is not reset when it is read by the *STB ? query. Although complicated, this seems to be required by IEEE-488.2.

***ESR ?**
Standard Event Status Register Common Query
c:-- q:*ESR ? r: ESR 32

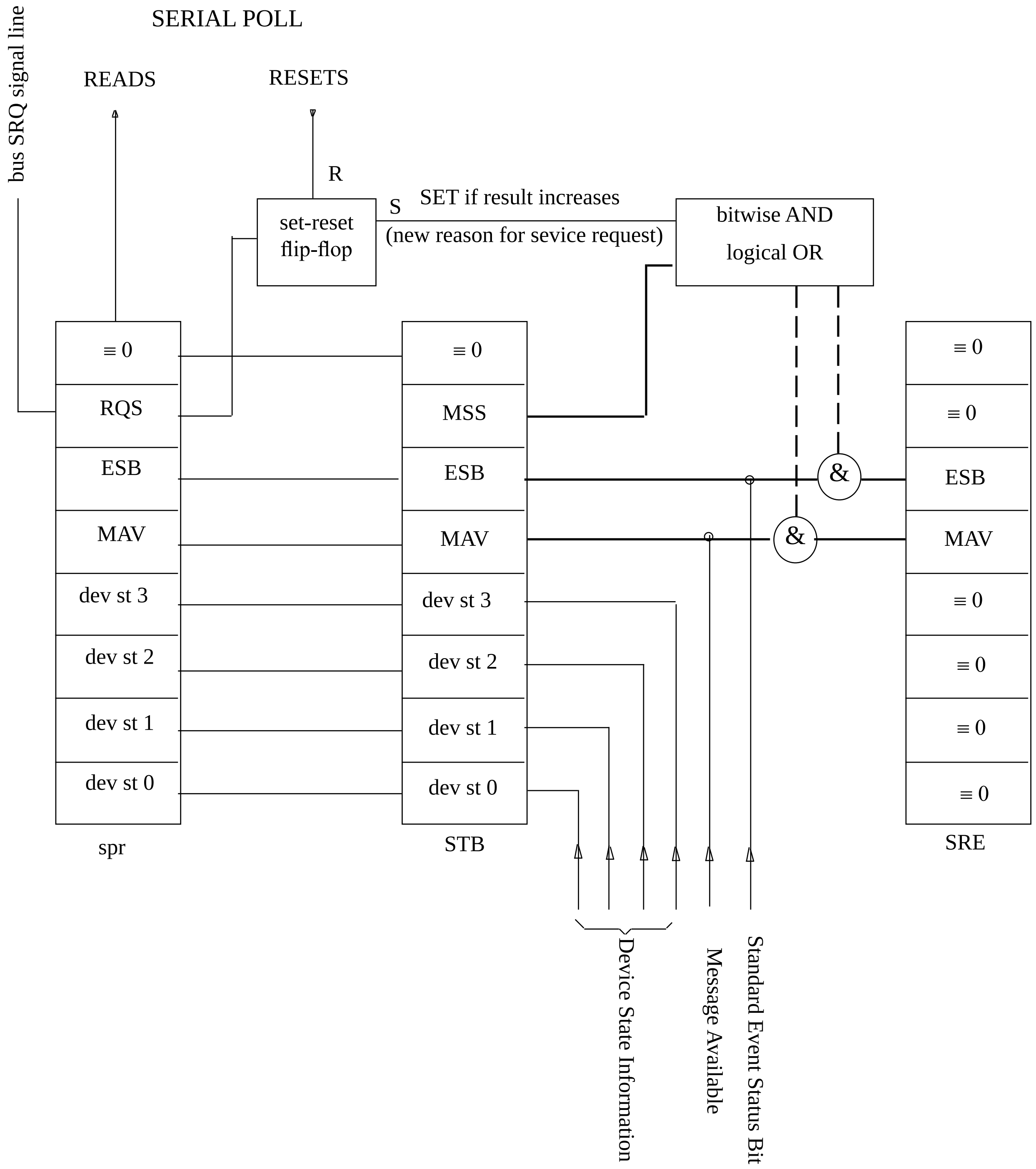

Fig. 12.1. State reporting and generation of the Service Request

The ESR register contains the following information:

128 Power-on.. Also the PONRST command sets this bit.

64 User Request. When FSM=0, tells that the front-panel START/PRINT switch has been activated. If FSM=1 (default), indicates that the Front-panel started measurement has been completed.

32 Command Error. Tells that the command mnemonic was not known by the device (usually misspelled).

16 Execution Error. Not implemented in this firmware version.

8 Device Dependent Error. Only ADC overload can set this bit.

4 Query Error. Not implemented in this firmware version.

2 Request for Control. A peripheral device like the AVS47-IB cannot request control.

1 Operation Complete. Asserted by the *OPC command.

The ESR register is dynamic. It tells about events that have happened, not about conditions that are prevailing. Reading this register by the *ESR ? query clears it. Also *CLS clears the ESR.
Except clearing, you cannot change the contents of the ESR register.

The highest **bit 7** is self-explanatory.

Bit 6 (User Request) indicates that the front panel START/PRINT switch has been activated. You can make use of this feature if you want to program some other routines than the existing Front-panel Started Scanning and Front-panel Started Measurement.

Bit 5 indicates a Command Error. It means that the command was either misspelled or otherwise unknown. **Note that also a misspelled or unknown query results in a Command Error.** This is due to the fact that commands and queries have the one single interpreter, and that queries are separated from commands only by the existence of the question mark.

Bit 4 should report about an Execution Error. The IEEE-488.2 standard defines two types of execution errors: either the argument is outside the acceptable limits, or a command cannot be succesfully executed because of some other device condition.

The AVS47-IB compares a command argument with minimum and maximum values allowed for that specific command. If the argument exceeds the limit, it is forced to the maximum value. If the argument falls below the minimum, it is forced to the smallest acceptable value. Then the command is executed normally. For example, if you command “RAN 8”, the bridge will be set to range No. 7, and no execution error takes place.

This firmware version 2.3 does not detect any such occasions where a device condition could prevent a command to be executed succesfully. There are such possibilities, but it is questionable whether they are important enough to be handled as errors. (Examples: Command REM 1 when hardware autoranging is on. Program the Difference Reference when the Reference Source is in the REF POT position.)

As a conclusion, Execution Error is never reported in version 2.3.

Bit 3 is reserved for Device Specific Errors. In the case that there are many possible such errors, this bit is a summary (logical OR) of them all. The AVS47-IB version 2.3. knows only one Device Specific Error; the A/D converter overload, which is fed directly to bit 4.

Bit 2 reports a Query Error. This instrument handles Query Errors as Command Errors, therefore **bit 3 is always 0**. Consider, for example, that you misspell “RAN ?” as “RAN +”. How could the program know, that you did not intend to write “RAN 5”.

Bit 1 is defined as Request for Control. Only an IEEE-488 controller device can request for control. The AVS47-IB is a peripheral device, and therefore bit 1 is always 0.

Bit 0 is reserved for Operation Complete -information. This bit is not set automatically. You must insert the *OPC common command in the <program message>. When the command interpreter encounters *OPC, it sets bit 0 of the ESR register. Because all AVS47-IB commands are sequential in nature, this means that all preceding commands have been executed.

Please note that completing the **execution** of a command does not mean that the AVS-47B is ready for taking a reading. Assume, for example, that the bridge is on range No. 3. Your command is

*CLS;RAN 4;ADC;*OPC

In other words, you reset the error registers, change the range, take a reading and generate the Operation Complete signal. However, the bridge may need 10 seconds to stabilize and the reading that you got is useless.

A correct program message would have been

*ClS;RAN 4;DLY 10;ADC;*OPC

where a delay of 10 seconds gives the bridge time to settle.

***ESE [0..255 | ?]**
Standard Event Status Enable Command / Query
c:*ESE 32 q:*ESE ? r:*ESE 32

ESE is a user-programmable 8-bit register. By setting some of it bits to 1 you can control, which of the ESR-specified events are reflected in the ESB summary bit of the Status Byte STB. The program takes a bitwise AND function of the ESR and ESE registers, and if the result is greater than zero, the ESB bit is set (OR function).

Suppose that you would like to reflect pressing the START/PRINT switch and Operation Complete (bits 6 and 0) to the ESB summary. Suppose further that your program makes some measurements after it has detected that the user has requested them and that *OPC is the last command. Then you could further enable this ESB summary bit to generate a service request; first when the switch is activated and next time when all the measurements have been made. The ESR and ESE registers would look like following:

Bit	ESR	ESE	ESR#AND#ESE
7	x	0	0
6	x	1	x
5	x	0	0
4	0	0	0
3	x	0	0
2	0	0	0
1	0	0	0
0	x	1	x

The ESE register is cleared by writing 0 to it. Zero is also the power-on value, to which ESE defaults after the PONRST command. According to the IEEE-488.2 the *RST Common Reset Command does not clear ESE.

***STB ?**
Master Status Byte Common Query
c:-- q:*STB ? r:*STB 16

The *STB ? query returns the IEEE-488.2 Master Status Byte. Reading does not clear nor alter this register in any way.

The STB register holds the following information:

128 presently not used
64 MSS, Master Summary Status
32 ESB, Event Status Bit Summary
16 MAV, Message Available
8 device state 3
4 device state 2
2 device state 1
1 device state 0

Figure 12.1. illustrates how the Master Status Byte and the serial poll response byte are constructed.

BIT 6, the Master Status Bit MSS is one if the bitwise AND function between the STB and SRE registers (see below) returns a value greater than zero. In other words, there is at least one **enabled** reason to request service.

Bit 5, the Event Status Bit Summary ESB is one if, and only if ESR#AND#ESE is greater than zero. In other words, bit 5 is set if any one of the specified events occurs, **and** if it has been enabled to set this bit.

Bit 4, Message Available, MAV, indicates that the output queue is non-empty. According to the IEEE-488 standards, the AVS47-IB will never produce output by itself. Only a query can result in an output message. It is good practice to verify always that a message is available before trying to read anything from the interface. Failure to do so may result in timeout errors in the case that the output queue is empty. But never use the *STB ? query for this purpose! Instead use serial polling.

NOTE 1: All queries produce output, and so does also the *STB ? query. The existence of a response message is indicated by bit 4. Therefore the answer to the *STB ? query can never be less than 16.

> **NOTE 2**: If you try to query with *STB ? whether the response to an earlier query is available, the response to *STB ? will overwrite that earlier message in the output queue. The length of the output queue is 255 characters. It holds only one <response message> at a time, but this message can consist of several <response message units>.

Bits 3, 2, 1 and 0 have been used for describing the state of the instrument. This lower portion of the STB byte has been used in a non-standard way. Please refer to a more detailed description below.

***SRE [0..255 | ?]**
Service Request Enable Common Command / Query
c: *SRE 16 q:*SRE ? r:*SRE 16

The SRE register is used to enable some of the bits appearing in the STB master status byte to generate a service request. The next paragraph describes generation of the service request in closer detail.

The firmware version 2.3 offers the possibility to enable only two bits: Set *SRE 16 if you want to generate an SRQ whenever a message is available, and *SRE 32 if you want to enable some of the possible **events** (or both, then SRE=48).

***CLS**
Clear Status Common Command
c:*CLS q:-- r:--

The *CLS command clears the ESR event register, the STB status register and the **spr** serial poll response byte. Generating a service request requires a **new reason** for that. Therefore you should routinely clear the status before giving commands that should later result in an SRQ.

This command can be used only when the system is in idle state (STB=16, serial poll returns 0).

***OPC**
Operation Complete Command
c:*OPC q:-- r:--

In order to operate properly, the *OPC should be the last command in a <program message>. When the command interpreter encounters this command, it sets the MAV bit in the Master Status Byte. If MAV has been enabled by setting bit 4 of the SRE register to 1, also the RQS bit in the serial poll response byte will be set and the physical service request signal line of the IEEE-488 bus will be asserted.

*OPC is the recommended way to tell to the controlling computer that a measuring sequence is ready. Example:

```
*ESE 1;*SRE 32
*CLS;ARN 1;MUX 3;EXC 2;DLY 15;ADC;RES ?;*OPC
```

In this example, *ESE 1 enables the Operation Complete bit in the ESR register to be reflected in the ESB summary bit in the master status byte, and *SRE 32 in turn enables the ESB bit to generate a service request.

***OPC ?**
Operation Complete Common Query
c:-- q:*OPC ? r:1

When the command interpreter encounters the *OPC ? query, it places the plain number “1” in the output queue.

Instead of using the *OPC command to initiate a service request, you can also use the *OPC ? query and enable the MAV bit of the status byte.

You can also mark some important points in your data by inserting the character “1” in it.

12.2. THE DEVICE STATES

The IEEE-488.2 standard requires that all bits in the status registers should be used so that a bit , if set, indicates that a condition is true and otherwise false.

Bits 0-3 of the Master Status Byte are not reserved by the standard, so that they are free to be used for device specific state information. However , there are only 4 free bits and many more interesting device states. By interpreting these bits as a 4-bit binary number instead of just individual bits, it was possible to provide useful state information.

Some of the AVS47-IB commands need a long time for execution, and there are also permanent modes where the interface can stay forever if not exited. Because the firmware is not interrupt-driven, it would be difficult to use ordinary queries (like *STB ?) for asking “what is it doing now”. The 7210 IEEE-488 controller is, however, able to handle serial polling quite independently from the software. So serial polling provides a convenient answer to asking the state, but at the cost of violating the standard. This violation should not be harmful in any way.

The states are decoded as follows:

0 idle state, accepts all commands
1 averaging, only STP or *RST allowed
2 scanning Only STP of *RST allowed
3 scan delay (waits for a new cycle to begin in the continuous-scan mode). Only STP or *RST allowed
4 temporary temperature control during scanning. Only STP or *RST allowed.
5 waiting for the START/PRINT switch in the Front-panel Started Scan Mode. Only STP or *RST allowed.
6 digital filter in progress. DFR ? query, STP and *RST allowed.
7 printing the buffer. STP and *RST allowed.
8 Reading the buffer (RBF mode). NXT ? query, STP and *RST allowed.
9 Self calibration in progress. STP and *RST allowed.
10 Sef-test in progress. Cannot be terminated.

There may be some short program phases where neither the STP nor *RST command works. Please verify by serial polling that the system has gone into the idle state, and if not, repeat the command.

Ask the state by serial polling. Separate the four lowest bits from the serial poll response **spr** using bitwise AND :
STATE = **spr** #AND# 15 15=(00001111)
Then compare STATE with the table above.

13. RESET COMMANDS

Three reset commands are the *RST Common Command, the device specific PONRST and the selective or non-selective Device Clear bus Command. The *CLS Clear Status, which was described earlier, represents also a sort of reset command.

***RST**
Device Reset Common Command
c:*RST q:-- r.--

This command sets the AVS-47B bridge to the following state:

INP 0
MUX 0
RAN 7
EXC 1
DIS 0

This reset state has been selected to prevent unintentional heating of the sensor connected to channel 0.

All PID parameters of the TS-530 Temperature Controller, if connected, are set to 0, except that the Set Point Voltage is set to its minimum allowed value 0001 (100 μV) . The bridge is left in the manual control (REM 0) mode.

The *RST command does not destroy the self-calibration data. Also all other parameters and data are kept intact.

PONRST
Power-On Reset Command
c:PONRST q:-- r:--

This command causes the interface to first perform the *RST command and then start the sorftware program just like when the power is switched on.

PONRST will return all parameters to their default values, and reads both the IBA and PBD from the non-volatile memory. It destroys all calibration data, scan parameters etc. So use this powerful command very cautiously.

> PONRST differs from a real power-on in that it does not destroy the RAM buffer.

The **Device Clear** bus commands work like the *RST command.

14. OTHER IEEE-488 RELATED COMMANDS

***IDN ?**
Identification Common Query
c:-- q:*IDN ? r:see below

The format of the response to this query has been defined precisely by the IEEE-488.2 standard. The response is :

*IDN PICOWATT,AVS47-IB,0,3.1

It consists of a response header, name of the manufacturer, name of the device, no serial number (indicated by 0), and the firmware version number.

This is presently the only query that produces a string response. All other responses consist of numeric elements.

IBA [1..30 | ?]
IEEE-488 Device Address Command / Query
c:IBA 20 q:IBA ? r:IBA 20

This device address is stored permanently in an EEPROM. Writing to this memory is limited to some 10000 times, so please do not change the address too frequently. Connect a printer to the AVS47-IB for the time of booting, and the current EEPROM write count is printed.

The factory default address is 20.

> The new address does not become into effect immediately. You must first either switch the power off and on, or use the PONRST command.

As long as you have not re-started the AVS47-IB, you must use the old device address. If a printer is connected to the AVS47-IB, it reports the current device address at power-on.

HDR [0..1 | ?]
Response Header Command / Query
c:HDR 0 q:HDR ? r:HDR 0

The IEEE-488.2 standard defines two formats for response messages: a “precise” version where the response data are preceded by descriptive message headers and a “headerless” version.

The HDR 0 command enables you to omit all response headers. The headers are included by default (HDR 1). Example:

```
HDR 1
DAY ?;TIM ?  =>  DAY 1994,3,1;TIM 10,12,13
HDR 0
DAY ?;TIM ? => 1994,3,1;10,12,13
```

The precise form is much easier for a human being to read, whereas the concise form might be better for a computer.

***TST ?**
Self-Test Common Query
c:-- q:*TST ? r:*TST error byte

This IEEE-488.2 defined query starts a self-test procedure. This procedure checks several functions of the AVS-47B+TS-530A combination and it may provide useful information in case of troubles.

Before issuing this query do the following: Connect the TS-530A to the AVS-47B using the 37-way ribbon cable. The signal cable is not necessary for this test. Reset both the TS-530A and the AVS-47B by **switching them off and on, and set the AVS-47B for CAL input, 200Ω range**. After that, do not touch the switches. Both instruments must be in their normal operating temperature (they should have been switched on for at least one hour). Make sure that the **Reference Source Switch is in the REF MEM position and the magnifier switch in the ΔR position**. Do not conduct self-test during an experiment, the test may interfere with your temperature measurements. Before starting the test, **enable the Alarm Line of the Picobus by using the DAL 0 command**.

Issue the *TST ? query. You can naturally enable a service request based on Operation Complete and use the *OPC command after *TST ?. The test takes some minutes and after that both the AVS-47B and the TS-530A are left to their reset states (see *RST). The response to the query is an 8-bit byte whose bits have the following meanings:

Bit 0: The AL signal fails. Possible causes: The Picobus cable is defective. 0 indicates that the interface is getting power and that it can assert this signal line.

Bit 1: Communication with the AVS-47B fails. The program could not set the bridge in remote mode and control its settings successfully. Possible causes: Wrong Picobus Device Address DIP switch setting, Defective Picobus cable. 0 indicates that the AVS-47B obeys remote control.

Bit 2: Analog Self-calibration fails. The test procedure checks the self-calibration only on the 3 mV excitation and 200 Ω resistance ranges. This check fails if offset is greater than ± 2 digits or if the middle-scale value deviates from 10000 by more than ± 2 digits.0 indicates that the check was successful.

Bit 3: Difference Reference fails. Either the offset is greater than ± 2 digits or the middle-scale value deviates from 10000 by more than ± 2 digits.

Bit 4: Difference mode fails. This phase of the *TST ? query checks the calibration of the ΔR difference amplifier. Failure can be due to an excessive offset, or changes either in the signal or reference gain. Errors are greater than ± 3 digits. Check both the ΔR and 10xΔR modes manually. 0 means the test was passed.

Bit 5: Communication with the TS-530A fails. Possible cause: The ribbon cable is defective.

Bit 6: The TS-530A Set Point fails. Either the offset is greater than ±3 digits or the middle-scale value deviates from 10000 by more than ± 3 digits. Adjust as described in the TS-530A manual. 0 means success.

The order of various checks follows the above bits, starting from the lowest one. Failure in one check may cause also the subsequent checks to fail. If, for example, the Picobus address is wrong, all but the first check fail. And if the ΔR reference is miscalibrated, also the ΔR mode check will fail. In such a case the *TST ? query would return *TST 24 (Bit3 + Bit 4 = 8 + 16 = 24). You shall therefore correct the lower-order errors and then repeat the test until it returns 0.

It is not necessary to have the TS-530A attached, but then the query will return at least the minimum value of Bit6+Bit5=64+32=96.

If a printer is connected, the results are interpreted and printed.

15. OTHER COMMANDS

DLY [1..1000]
Insert Delay Command
c:DLY 10 q:-- r:--

This command inserts a delay (in seconds) in the program message. The main usage is to give the bridge time to stabilize before taking a reading. Example:

EXC 2;DLY 15;ADC;RES ?

Be careful not to address the AVS47-IB to talk before all commands of the above line have been completed. Failure to wait enough results in a partial response or no response at all! The failsafe way to use DLY is to put it on its own command line. Then the firmware will read the next command only after the delay has passed.:

```
EXC 2
DLY 15
ADC;RES ?
```

You can also use the *OPC command and use the service request to indicate completion of all commands.

```
EXC 2;DLY 15;ADC;RES ?;*OPC
```

PBA [1..15 | ?]
Picobus Device Address Command / Query
c:PBA 2 q:PBA ? r:PBA 2

The default PBA used by the program is 1. You can change it by the PBA command temporarily. The AVS47-IB defaults to 1 again at power-on.

There is no need to change PBA unless there are more than one Picobus-equipped instruments connected to the same Picobus interface line. Refer to “several Picobus Instruments” under the “Miscellaneous Topics” title.

The Picobus instrument must have its address DIP switch set to the corresponding value. The factory default setting of all instruments is 1.

PBD [200..1000 | ?]
Picobus Delay Factor Command / Query.
c:PBD 10 q:PBD ? r:PBD 10

You can adjust the speed of Picobus with the PBD dommand. There is no need to change the default value of 200 as long as the supplied 5m Picobus cable is used. If you have a very, very long cable, or if you have inserted RF filters with long time constants in the cable, then you may need to make the bus slower.

It is also easier to see the bus activity (e.g. for trouble-shooting purposes) when PBD is made long. Experiment with PBD=1000. You can see how the front panel “READY” light indicates that data is being transferred. On the other hand, the inteface will become very slow.

The PBD is, like the IBA, written to the EE-PROM memory having a limited write count. As opposed to the IBA, the new delay factor comes into effect immediately.

LRS, LRS ?
Lead Resistance Measurement Command / Query.
c:LRS q:LRS ? r:1.2345E+01

This macro command measures both the sensor’s value and the excitation compliance voltage (DIS 4). Then it calculates an estimate for the overall current path lead resistance.

The result is only a rough estimate, because the compliance voltage is often very small compared with the resolution of the A/D converter and with the offsets etc. in the measuring circuit.

The compliance voltage is the higher the lower is the measuring range and the higher is the excitation. Therefore, select the lowest possible range and the highest excitation you dare to use (perhaps you can afford the excitation to heat the sensor temporarily).

Use the LRS ? query to read the response when it is available.The lead resistance is given in Ohms.

SCAL [0..1 | ?]
Self-Calibration Command / Query
c: SCAL q:SCAL ? r: see below

Digital Self-Calibration means that the offsets and scale factors of the AVS-47B are measured separately on each excitation range, using long averages, and that the results are then automatically used to correct all subsequent readings.

The correction applies to the RES ?, AVE ?, MIN ?, MAX ? and STD ? queries, but it does not affect the lower level ADC ? query. Also measurements of lead resistance and heater output etc. remain unaffected.

Digital self-calibration is useful when the AVS-47B stays in a rack and it cannot be opened for adjusting the trimmers, which is necessary for calibrating the analog circuits.

Another situation, where the digital calibration is very useful, is when the ΔRx10 display mode is used. Because the analog scale factor and offset (determined by the front panel potentiometer) are very difficult to adjust precisely, SCAL can greatly improve the measurement accuracy.

Digital self-calibration provides also a convenient way to increase the measuring range of the AVS-47B near to 40000 instead of the normal 20000 (with slightly decreased accuracy). See “Miscellaneous Topics”.

SCAL 0 sets all seven offsets (for the seven excitation ranges) to zero and all seven scale factors to unity. This is the default condition.

SCAL 1 command starts a self-calibration procedure. Before issuing SCAL 1, you must first select the 200 Ω range (RAN 3) and normal resistance display (DIS 0). If you are going to calibrate the ΔR-mode, then select RAN 3 and DIS 1, and if you wish to calibrate the ΔRx10-mode, then select RAN 4, DIS 1 and set the magnifier switch to ΔRx10. Check also that the Difference Reference is what you want it to be (often zero).

SCAL 1 needs a long time, over 15 minutes to complete. Use the *OPC command, for example, to generate a service request when the macro is ready.

Calibration factors are stored in normal RAM, and they are lost in case of a power failure.

Neither the *RST nor *CLS commands destroy the calibration data, whereas PONRST does that.

SCAL ? query returns seven offsets (for the seven excitations from 3 mV **down to** 3 μV), and seven scale factors in the same order. These real numbers are separated by commas. After SCAL 0 these will be only 7 zeros and 7 ones.

> Digital self-calibration does not calibrate the analog outputs of the bridge. Therefore, using this feature together with the TS-530A Temperature Controller may lead up to confusion.

> NOTE: The out-of-range values of the RES, AVE etc. readings can be used to indicate overload only when the self-calibration factors are reset (after power-on and SCAL 0), but not after SCAL 1. Instead, use the OVL ? query together with digital calibration. Or check bit 3 of the ESR register (see “Status Reporting” and “Using SRQ’s”).

FSM [0..1 | ?]
Front-Panel Started Measurement Command / Query
c: FSM 0 q:FSM ? r: FSM 0

The FSM 0 command gives you the option of disabling the Front-panel Started Measurement. This operation is necessary if there are many Picobus instruments connected to a single Picobus line. If any one of them asserts the AL signal line of the bus, then the AVS-47B would behave as if its START/PRINT switch had been activated (this feature relies on the AL signal).

Another situation where you might need FSM 0 is when you are storing query responses to the buffer and you do not want records of quite different format to appear among your data. The Front-Panel Started Measurement stores its results automatically in the buffer if DSK=1, and it uses its own number and order of items.

FSM 0 disables also the front-panel started printing of the buffer contents.

16. MISCELLANEOUS TOPICS

16.1. FRONT-PANEL STARTED MEASUREMENT

The AVS47-IB can be used as a printer output and/or data buffer even without a computer.

When the AVS47-IB is in idle state (which happens always after power-on), the software program monitors the state of the front panel START/PRINT switch. Lift this switch **momentarily** (less than 3 seconds), and the interface counts an average of several A/D conversions without changing any of the current front panel settings. The average count is 300 for 3µV excitation, 100 for 10µV, 30 for 30 µV and 10 for all higher excitations.

The result is automatically stored in the buffer, even if buffering had not been enabled. The purpose of bypassing the DSK parameter in this case is to make the computer unnecessary for front-panel started measurements. One measurement would look like follows in the buffer:

DAY 1993,12,6;TIM 4,37,24; RES 1.0006E+02;INP 1; MUX 0; RAN 3; EXC 7;DIS 0;OVL 0

The result is automatically also printed. Just connect a parallel printer to the AVS47-IB:s printer port. The PRN parameter is bypassed just like DSK. Following is an example of a printed line

0-0-0 4:37:24 1.0006E+02 1 0 3 7 0

Compare these with the scan output, and you see that the formats are the same. Because the AVS47-IB:s real time clock is not battery-backed, it is set to zero date and time at power-on. These can be changed only via the GPIB interface.

The printed items are: year-month-day hour:minute: second resistance inp mux ran exc dis.

In case of overload, a remark “dvm overload” is added to the end of the line.

Keeping the START/PRINT switch lifted for more than 3 seconds (but less than 10) enables printing the buffer contents. This feature is also independent on the PRN setting, so that the buffer can be printed even without a computer. Use a printer having at least 80 character line length.

You can disable the Front-Panel Started Measurement using **FSM 0**, if you e.g. wish to prevent the user from initiating it by accident and thereby interfering with your other buffered data. Also front-panel started printing of the buffer is then disabled. See the “FSM command”.

16.2. DIGITAL FILTER

The digital filter makes it possible for you to program the AVS47-IB to collect data from one sensor, using a selected range, excitation and display item (in practice only R, ΔR or ΔRx10) without further intervention. You can then, any time, ask for a current reading, which is as stable as an equally long average but which is available immediately.

The filter has few commands which were described in 8.4. Let us denote the filter's output DFR (Digital Filter Reading; this is also the mnemonic command for reading the filter). After having set the filter length DFL and started the filter mode by DFS, the filter operates as follows: The first DFR is the first A/D conversion divided by one. The second DFR is the sum of the two first A/D conversions divided by two and so on. This continues until the divider has grown to DFL, the filter's full length. Then the filter starts to maintain a circular queue of DFL last measurements and keeps track of their average.

This algorithm guarantees fast settling even when the programmed DFL is large. Because the first DFRs are based on only few samples, they contain noise more than later readings.

In most applications it is a valid assumption that large jumps in the A/D conversion values do not come from the sensor but are due to switching the input channel. Therefore, if a new A/D conversion differs more than ±1000 counts from the last calculated DFR, the filter is automatically re-started from the beginning. The filter is also re-started in case of an autorange operation.

Because the AVS47-IB accepts only the STP command and DFR ? query in the filtering mode, it is not possible to detect overload by the OVL ? query. Therefore overload is handled as follows: As long as there is at least one overloaded ADC result inside the circular queue of length DFL, the DFR ? query returns 2.0001E+06. But as soon as overload has disappeared and the queue has been filled with valid fresh numbers, DFR ? starts to return useful readings.

This way you cannot obtain a distorted reading by mistake.

16.3. USING SRQ's

Service requests are very important when using the AVS47-IB, because the AVS-47B bridge is a slow instrument and because the firmware program offers many macro commands that need an exceptionally long time to complete. Suppose a query "ABC ?" which causes a device to make a sequence of operations that needs 15 minutes and then places the result in the output queue to be read by the controller.

How can you know, when to read it? If you try to read it immediately, the IEEE-488 bus either hangs up for 15 minutes, preventing any other action, or you get a timeout error. Then you must try again until reading succeeds. Neither is a good alternative.

The second way to proceed is to serial poll the device (to read its IEEE-488.1 Status Byte; we call it the "serial poll response" **spr** to separate it from the IEEE-488.2 defined Master Status Byte STB which is read by the *STB ? query). There are many ways, how the **spr** can then tell that the query has been completed.

Serial polling is a good method and often it is the only real alternative. But it is difficult to write a program that could serially poll one or more instruments and still use that time, when nothing happens, for something useful. One needs a hardware interrupt in order to write such programs.

The IEEE-488 bus includes a Service Request signal line. This line, when asserted, can be used to initiate a hardware interrupt in the controlling computer. The interrupt service routine could then find out, which device is requesting service and what kind of service.

The SRQ bus signal follows bit 6 of the serial poll response register **spr** (RQS bit). Now return back to Figure 12.1. The **spr** is a direct copy of the STB register, with the exception of bit 6. While bit 6 of the STB register is a logical OR of STB#AND#SRE, bit 6 of the **spr** is set each time the value of STB#AND#SRE increases.

The MSS bit indicates that the device has at least one enabled reason to request service. This register is not altered by reading it (*STB ? query). **The RQS bit is reset by a serial poll**, but it is set again if a **new reason** to request service appears.

Service requests fall in two categories: those resulting from a message becoming available (MAV) and those resulting from any enabled device event(s).

The MAV bit is set whenever the output queue is non-empty.

The reason that caused the ESB bit to become true is found by inspecting all those registers that contribute to this bit, more specifically the enabled events in those registers. Let us take some practical examples.

We want to calculate an average of 100 A/D conversions, and get a service request when the result is ready. The commands would be:

*SRE 16	enable the MAV bit
*CLS	clear status
AVE 100	make the measurement
AVE ?	query the result

As soon as the AVE ? query is completed, the MAV bit will be the "new reason to request service", setting RQS and asserting thereby also the physical SRQ bus signal.

Suppose that your use the digital filter, and that you want to control autoranging yourself. Then a possible overload should initiate a service request.

Refer to the description of the *ESR ? query and *ESE command (12.1). Setting ESE=8 carries the overload error to the ESB bit. Setting SRE=32 in turn enables the ESB to generate an SRQ.

```
*ESE 8;*SRE 32;*CLS
DFS
```

Now the first occurrence of an overload will cause a service request and your program can change the range. Even if you serial poll the device (thus resetting the RQS bit), ESB remains set, and the second overload will not be a "new reason for service request" unless you remember to clear the registers by *CLS.

A single scan cycle may take a long time but it does not produce any output. Here the *OPC command comes in handy. The Operation Complete is bit 0 in the ESR register and it is enabled by setting ESE=1.

```
*ESE 1;*ESR 32;*CLS
SCN 0
```

Often it is not necessary to write an interrupt-driven program. This can be the case if the controller serves only few instruments that do not need to be used simultaneously. Then serial polling provides a much easier way to detect a service request than using a hardware interrupt.

Serial polling could be used as follows:

```
WAIT_SRQ

i:=0
REPEAT
   BEGIN
       i=i+1
     READ STATUS BYTE spr(i)
   END
UNTIL i=number of instruments

IF  (spr(1) AND 64)=64 THEN
    BEGIN
        Device No. 1 requested for service. Take care
        of it. Section is specific to the type of device 1.
    END

IF (spr(2) AND 64)=64 THEN     etc.
        Section for device type 2.
```

16.4. INCREASING THE MEASURING RANGE WITH SCAL

Digital calibration provides a convenient way to increase the measuring range from the standard 19999 to almost 40000 with a slight decrease in accuracy. Proceed as follows:

Let the unit warm up until its temperature is as stable as possible. Set the Difference Reference to 19900. You shall do this using command REF 19900, because the front panel potentiometer does not reach the full scale. Select ΔRx1 magnification and difference display mode (DIS 1). Then start the digital self-calibration procedure by SCAL 1.

The idea is to offset the input voltage to the A/D converter by -1.9900 Volts and to use digital calibration to provide the opposite offset so that 0 Ω input results in a reading of 0.

It is necessary to use an offset which is slightly smaller than -2V (we used 19900), because otherwise there is the danger that the DVM sees a negative overload (V_{in} < -2V) especially at lower excitations when the zero readings are noisy.

After calibration, the range extends from 0 to 39900, thanks to the fact that the range of the A/D converter goes from -19999 to +19999.

The "promised" reduce in accuracy is due to the limited performance of the difference amplifier. For best results, verify, and if necessary re-adjust the amplifier.

16.5. COMPATIBILITY WITH THE DC900

The AVS47-IB provides some degree of compatibility with the old DC900 type IEEE-488+RS232 interface of the AVS-46 Resistance Bridge. The command interpreter recognizes the following commands and queries. With them you may be able to include the new AVS-47B in a system originally designed for the AVS-46 with few or no changes, and use these two models from the same program.

Because the DC900 does not use response headers, you must first issue the HDR 0 command.

/ IBA [1..30 | ?] Sets the GPIB device address. The highest address is 30, whereas DC900 offers 31.

P This command resets the bridge. However, the default states of the AVS-46 and AVS-47B are different.

C [0..1 | ?] Sets the bridge to local or remote mode (same as REM).

R [0..7 | ?] Selects the range. Note that the old AVS-46 does not have R 0, whereas the AVS-47B has (the "open" range). Other numbers have the same meanings.

X [0..7 | ?] Selects the excitation. The AVS-46 has no excitation No. 0. And although the excitation ranges start from 1 in both cases, the numbers have different meanings: In AVS-46 ranges start from 10 µV, whereas in the AVS-47B they start from 3 µV.

M [0..7 | ?] Selects the multiplexer channel. In the AVS-46, MUX 0 means the non-multiplexed front-panel input. There are 7 multiplexed channels available. In the AVS-47B all 8 channels are multiplexed and there is no front-panel input.

D ? Query returns the next A/D conversion. The result is expressed in Ohms, as a fixed point decimal value. Results obtained on the 20 kΩ or higher ranges are integers, results obtained on lower ranges have a decimal point.

H ? Query is the same as D ?. The AVS47-IB does not offer the possibility to read the last A/D conversion, but always the next.

The indirect readout mode is not supported.
The Echo command is not supported.

Overload is indicated by returning 9999900 for the D ? and H ? queries.

The status reporting systems of the DC900 and AVS47-IB differ greatly. The least common denominator is the Status Byte STB, more specifically bits 4,5 and 6 (MAV, ESB and RQS/MSS). Also the deeper registers are different.

100% compatibility is provided only by the MAV bit, **when it is read by serial polling**.

16.6. THE AVS47-IB AND SEVERAL BRIDGES

It is possible to connect several instruments to the Picobus by just extending the Picobus cable from one instrument to another. This cable is a 1:1 extension of the cable coming from the computer (see chapter 4).

Picowatt can offer extension cables having convenient “piggyback” connectors in both ends. The standard length is 1 meter, but the cables are made on order, so that the length can be changed.

Consider a situation where you have two AVS-47B bridges. There may or may not be TS-530 Temperature Controllers connected to them. You also have one single AVS47-IB secondary interface.

The first bridge is connected to the secondary interface by the “standard” Picobus cable, and the second bridge is connected to the first one using an extension cable.

Now you can operate both bridges from the same secondary interface. Just give the command PBA 2 and all subsequent conversations are made only with the second bridge. Give the command PBA 1 and you can control the first bridge.

But be very careful! The program uses the same settings and parameters (scan parameters, for example) for both bridges. Write you program so that you can easily reprogram the AVS47-IB completely after having changed PBA. The output data buffers will be separate for the two bridges, so that you need not read nor clear before moving to a new bridge.

Before changing PBA, always leave the bridges in the REM 0 mode. It is also good practice to use the REM 0 as the first command to a new bridge. After these precautions you have the maximum security that the bridges do not change their state by mistake.

Disable the Front-Panel Started Meassurements using FSM 0. This measurement relies on the AL signal line, which is common for all instruments. Therefore the bridge that you are **not** using could command the AVS47-IB to start an FSM measurement with the bridge that you have selected.

16.7. TROUBLE SHOOTING

There is not very much one can do with the **AVS47-IB** in the case of problems. Something can still be done, and because it is much easier to ship the secondary interface unit for repair alone than together with the whole bridge, it is important to try to isolate the problem.

THE RIGHTMOST LED DOES NOT LIGHT

The secondary fuse has blown. Disconnect the **AVS47-IB** from mains, open it and check the fuse. If it has blown, replace by a 5x20mm 1A slow fuse.

POWER SEEMS TO BE OK BUT THE "READY" LIGHT DOES NOT COME ON

The unit is wired for 230V and you are using 115V mains. The power LED lights although the secondary voltage is insufficient for the electronics. Open the unit and check the voltage setting.

If the secondary unit has been subject to vibration or hard shocks, any of the connectors may have loosened. Check that the connectors are firmly positioned.
If these measures do not help, the secondary unit is apparently defective.

BOTH LED'S ARE OK BUT THE UNIT DOES NOT RESPOND VIA GPIB

It is possible that the GPIB connector to the upper of the two circuit boards is not in its place.

Check that you are using the correct IEEE-488 device address. The default address is 20 when the interface is shipped. You or somebody else may have changed the address, which is stored in the **AVS47-IB**'s nonvolatile memory. And the funny thing is that you cannot ask the address if you do not know the address. If you do not want to serial poll the **AVS47-IB** using all the possible 30 addresses, do the following:
Connect a parallel printer (Centronics type) to **AVS47-IB**'s printer port. Then switch the **AVS47-IB** off and on again. After 25 seconds , the printer should print a few lines of text, including the current GPIB address.

THE UNIT COMMUNICATES AND ANSWERS TO QUERIES LIKE ESR?, BUT DOES NOT COMMUNICATE WITH THE AVS-47B.

Conduct the Self-Test procedure (see description of the *TST ? common query). Failure in communications can be due to a wrong Picobus Device Address setting or defective Picobus Cable (use the originally supplied 5 m cable version "B" for this test if it is available). Check that the short-circuit piece JP204 is in its position but JP201.. JP203 are not jumpered (AVS47E circuit board).

THE UNIT COMMUNICATES AND CONTROLS THE AVS-47B, BUT NOT THE TS-530A.

Conduct the self-test procedure. Check whether the ΔR reference test passes OK. If it does not, verify manually that the reference offset and scale factors are properly calibrated. Repeat the test. It the ΔR reference test still fails, there may be a hardware fault in the AVS-47B.
If the self-test is now OK, but TS-530A still fails, check the 37-pole ribbon cable rigorously, or test with another cable.

16.8. SCANNING AND TEMPERATURE CONTROL

16.8.1. THE PRINCIPLE IN GENERAL

Sometimes an experiment requires, besides temperature control, also information from a few other resistive sensors. A high-performance, though expensive, solution would be two resistance bridges and one controller. But for "less-than-critical" applications, the AVS47-IB offers a cost-effective way to use one single resistance bridge for both scanning and control.

The principle is based on the idea, that **if** a PID control system is in balance, the controller's heating power is determined solely by the charge accumulated in the integrator. In an equilibrium, the error signal is zero by definition, and so is also the rate of change of temperature. The proportional and differential output terms are thus also zero.

Let us then inhibit the active control by forcing the error signal to stay at zero. Nothing should happen to the heater output power, because the integral term remains intact. If the thermal load and all other factors were ideally stable, the system would maintain its temperature forever. In practice, however, the system starts to drift immediately.

Assume that we switch the resistance bridge quickly to another sensor, measure it as soon as possible, return back to the control sensor, let the bridge to stabilize and enable the active control again by re-opening the error signal. If the drift is slow enough, the system would not have gone far from equilibrium. Give the system some time for active control and switch to measure the next sensor. And so on.

16.8.2. SCANNING WITHOUT TEMPERATURE CONTROL

Let us first discuss the more simple application, where a few sensors should be measured from time to time and where no temperature control is involved.

There are several parameters that must be set before the scanning commands can be used. See also the detailed descriptions of each parameter.

FCH Specifies the first channel to be measured. E.g. FCH 3.

LCH Specifies the last channel to be measured. E.g. LCH 5. So the channels to be scanned are 3,4 and 5. LCH must be ≤ 8.

SCI Specifies the scan interval in seconds. This time determines, how often the three channels are measured. E.g. SCI 600 means that the scan cycle is repeated every 5 minutes. The SCI parameter is important only in the continous mode, whereas it has no meaning in the single-cycle nor in the Front-Panel Started modes.

It is possible that FCH=LCH. Then only one sensor is measured. But LCH must not be smaller than FCH. The channels are measured in numerical order, and it is not possible to skip channels.Please wire all sensors to adjacent inputs.

The next step is to specify how to measure these sensors. The measuring conditions are given separately to each channel. All these parameters have default values, which are mentioned in the detailed descriptions of the appropriate commands.

The AVS47-IB requires answer to the following questions (for each sensor): a) Which range to use, b) which excitation to use, c) how many samples to measure for an average and d) how long time to reserve for the bridge to stabilize.

SCP n This SCan Parameters -command sets the channel for which you are going to specify those parameters that follow on the same single line.

RAN Specifies the range

EXC Sets the excitation

SDY Defines the stabilization delay and

CNT Is the average count.

For example, parameters for channels No. 3 and 4 could be as follows:

SCP 3;RAN 5;EXC 3;SDY 10; CNT 30
SCP 4;RAN 2;EXC 6;SDY 5;CNT 10

It is strongly recommended that you always scan in autoranging mode (ARN 1), which guarantees that the results do not suffer from overload nor underrange. Then the RAN parameter is optional. It specifies the initial range for measuring a sensor. If autoranging has to change this setting, the AVS47-IB remembers the new range and uses it for the next scan cycle. But if ARN=0, then the range specified by RAN is used regardless of what happens.

Excitations cannot be changed automatically.

Scanning without temperature control is the default, but you can also specify this by command ETC 0.

16.8.3. THE SCAN MODES

A single scan cycle is started by command SCN 0. Please refer to Figure 16.1 The simplified flow diagram may help you to follow this discussion.

After the SCN 0 -command, the AVS47-IB saves the original settings. It is possible, for example, that the channels to be scanned are 3-5, but that the bridge is currently in the following state: Input=Calibrate, Mux=0, Range=3, Excitation=7, Local control. The “save original settings” means saving all the above information.

The block diagram tells that (when ETC=0) the system is set in Remote Mode, and the bridge is prepared to measure the first channel FCH using the appropriate settings. After a stabilization delay, an averaging measurement is started, using the average count CNT that was specified for this channel. If autoranging is on (ARN 1) and if the range had to be changed, the AVS47-IB saves the new final range to be used the next time.

Now you have the options of printing and/or buffering the result from the first channel. Set either PRN 1, DSK 1 or both. The output has a fixed format that cannot be changed. You can also ask the results later via the IEEE-488 bus. Refer to 8.5.4, “Results in the SCN 0 mode”.

If the LCH is greater than the channel that was just measured, the channel number is incremented and the next sensor is measured.

After having measured the last channel LCH, the original settings are restored, the original REMote control mode is restored and the system returns to the idle state.

As soon as you gave the SCN 0 command the system entered state No 2. The state can be queried by serial polling. After completion, the system returns to the idle state No. 0.

The second scan mode, the "Front-Panel Started" mode is effected by command SCN 1.

The beginning of this mode is exactly the same, but instead of returning to the idle state, the system starts to wait for the front-panel START/PRINT switch to be lifted momentarily. This action begins a new cycle.

The system state during scanning is 2, but the waiting-phase is indicated by state 5.

The AVS47-IB would stay in this mode forever unless you exit from the mode by the STP command. Be sure to verify that state is 5 before exiting the mode.

Use either printing or buffering to save the results. After having exited the scan mode, you can also ask for the data via the bus, but only for the last cycle. Chapter 10 discussed how to read the buffer.

The third scan mode is the "Continuous Scanning". It is started by SCN 2. Instead of exiting the mode after the first scan cycle, the system waits until the Scan Interval, as specified by the SCI parameter, has elapsed.

The scan interval is calculated from the start. Therefore SCI must be longer than the total time needed for one SCN 0 cycle.

The waiting phase in the continuous mode is indicated by state No. 3. Be sure to check that state is 3 before exiting the continuous mode by the STP command. Then the system will return to the initial situation.

The results must be saved by printing and/or buffering. Only the last cycle can be queried after having exited the mode.

If you watch the AVS-47B while it conducts a scan cycle, you can notice several delays of varying lenghts. These delays, which reserve the bridge time to stabilize, have not been shown in the flow diagram for simplicity.

16.8.4. DISABLING CONTROL TEMPORARILY

Suppose then that the main task of the AVS-47+TS-530A combination is temperature control, but that you would also like to have information from a few other resistive sensors from time to time.

Let us further suppose that channel No. 6 is used for control (the control channel can be anything outside the range FCH...LCH) and that the system is in currently good balance. Temperature control is best made without autoranging (ARN 0). All PID parameters have been given to the TS-530A via the interface, not manually.

The AVS47-IB is enabled to handle both temperature control and scanning by the ETC 1 command (ETC 0 is the default). Believing that the temperature remains very stable during scanning, we disable temporary control by issuing TCP 0 (TCP 60 is the default).

The flow diagram tells that, after having saved the control channel settings, this channel is measured. The default average count for any channel is 10, but you could have changed this default e.g. to 30 using the commands SCP 6;CNT 30. Also the control channel's stabilization delay can be changed in the same way: SCP 6;SDY 20. This SDY is used later in the cycle when control is enabled again. The results can be printed and/or buffered depending on the PRN and DSK settings. Note that the original channel is not measured if ETC=0.

The system is set into Remote and temperature control is disabled. This means that the TS-530A's error signal is forced to stay at zero and that the integrator time constant is set to infinite (for minimizing drift).

The bridge is prepared to measured the first channel FCH using this channel's parameters. After the measurement, the final range is saved for the next cycle. Results can now be printed and/or buffered. The "ERROR" meter of the TS-530A displays some deviations, but this meter shows the error signal before it is short-circuited. The "POWER" meter should indicate that the output power remains constant.

All the original settings are restored. But because TCP=0, the bridge is not switched to the control sensor. Instead, the channel number is incremented, new parameters are selected and the next channel is measured. This continues until the channel number is equal to LCH.

The original settings are restored and temperature control is enabled. Enabling the control involves that the AVS-47B is given time to settle (SDY for the control channel) before the error signal and the integrator are opened.

Depending on the scan mode, the system will now return to the idle state (SCN 0), start to wait for the switch (SCN 1), or start to wait until the scan interval has elapsed (SCN 2). Temperature control is active during the waiting time. The system stays in the remote mode regardless of what was the mode before starting to scan.

You can exit from the wait states using the STP command. Then the system returns to the original remote mode. You can safely exit when the state is either 3 or 5, but if it is not, the system may end up to some peculiar state which is difficult to correct without disturbing the control temperature.

16.8.5. USING TEMPORARY CONTROL

The above method works the better the less channels you have to scan and the more stable are the heat loads etc. But it can turn out that the temperature drifts more than what is acceptable during scanning. Some improvement can be achieved by using temporary temperature control also in between various channels.

The flow diagram shows how the temporary control period is inserted in the loop after having measured each sensor. The length of this control period (in seconds) is determined by the TCP parameter (which must now be greater than 0). A control period of one minute may be all that the system needs to approach equilibrium so well that a new cycle can be started. As a drawback, scanning is now much slower than before.

It is not possible to predict the performance obtained this way. Only an experiment in a real situation can show that.

17. THE PICOLINK OPTION

17.1. DESCRIPTION

The Picolink Option is an optical fibre link that replaces the standard Picobus wire connection between the AVS-47B and AVS47-IB. The Picolink consists of two circuit boards and a fibre cable.

The transmitter/receiver boards are installed in both the AVS-47B and the AVS47-IB, right behind the rear panels. Picolink can be installed only into new instruments, when both an AVS-47B and an AVS47-IB are purchased together.

Each circuit board contains two optical transmitters and two optical receivers, corresponding to four Picobus signal lines. The fibre cable consists of four low-cost, 1 millimeter plastic fibres. Four fibres are needed because Picobus uses four hardware handshake lines of the embedded computer’s COM port.

The optical receivers/transmitters use visible light, which makes it easier to verify the operation of the interface. No fibre connectors are needed, but the 1mm fibres are inserted directly in the receivers/transmitters and tightened by hand.

17.2. CONNECTING THE PICOLINK

The four fibres of the Picolink cable have colour codes, whereas the rear panel optical connectors in both the AVS-47B and AVS47-IB have numbers. Connect the fibres between corresponding numbers. One suggestion to use the colour codes is:

1 = Black
2 = Red
3 = Yellow
4 = Blue

Be sure to push the fibres fully in before tightening the connectors by fingers. **Do not use any tools for tightening, it would lead to damage of the plastic threads.**

Check the operation as follows: Switch the power of the AVS-47B on first. Select CAL mode, 200 kohm range and 3mV excitation. Then plug the power cord into the AVS47-IB. After the “ready” light has come on, lift the “START/PRINT” button of the AVS-47B momentarily. After a second or so, the “ready” light should be dimmed 10 or 11 times, telling about transactions between the instruments.

If you cannot see any sign of transactions, verify once again that the fibres are well inserted in the connectors and that corresponding numbers are connected. The connectors are not in the same physical order in the two instruments!

If the interface still does not operate, ask a qualified serviceman to open the top cover of the AVS-47B and verify that

- jumpers JP201 and JP202 are absent
- the 8-pin connector going to the Picolink circuit board is in place.
- the TRACO DC-DC converter is firmly in its socket

Similarly, ask the serviceman to open the top cover of the AVS47-IB and verify that the connector to the Picolink board is in place.

If the link still does not work, contact Picowatt.

17.2. REMOVING PICOLINK

Removing the Picolink requires opening the AVS47-IB and therefore it is allowed only of a qualified serviceman.

The Picolink is removed as follows:

1) Open the top cover of the AVS47-IB. Pull out the connector that goes to the Picolink circuit board. Close the cover.

2) Open the top cover of the AVS-47B. Pull out the connector that goes to the Picolink circuit board.

Close the top covers.

The Picobus cable must not be connected between the two instruments until the Picolink has been disabled. In such a case, the communications do not work. None of the circuit boards should suffer from this, however.

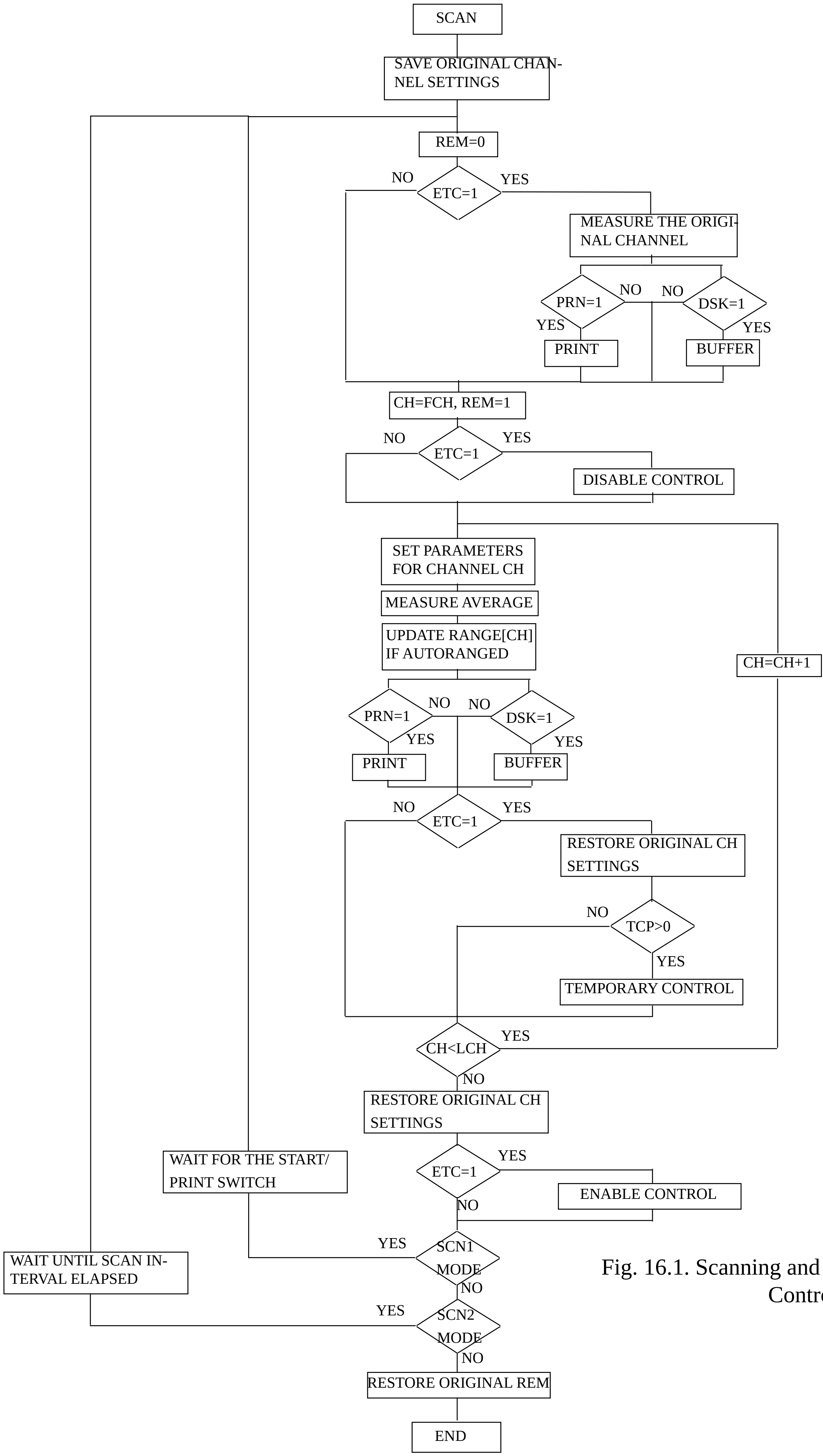

Fig. 16.1. Scanning and Temperature Control

INDEX

REVISION HISTORY

CHANGES BETWEEN VERSIONS 1.1 AND 1.2

The power-on default date is no longer 2000-1-1 but 0-0-0 (year-month-day). The new default date indicates experiment time better.

The buffer contents can now be printed by holding the Start/Print swicth lifted for more than 3 seconds. Like the Front-panel started measurement, this printing feature does not need a computer.

The Picobus device address range extends from 1 to15.

Each AVS-47B that has its own Picobus Device Address, has now also its own buffer. A new buffer becomes effective when a new PBA is selected. At power-on, the program defaults to PBA 1. All operations apply to the currently selected PBA buffer.

A new query BUF ? returns the buffer size and free buffer space.

Overload indication of the digital filter has been corrected.

The IEEE-488.2 defined self-test query *TST ? has been implemented.

During scanning, the state no longer toggles between scanning (st=2) and averaging (st=1) but is continuously 2.

CHANGES BETWEEN VERSIONS 1.2 AND 1.3

The computer inside the secondary unit has been changed from AMPRO CM/XT to a newer type CM/XT PLUS. This is a lot faster, and therefore the default Picobus delay (PBD) is now 30 instead of the previous value of 1.

There are no other changes between the two versions.

CHANGES BETWEEN VERSIONS 1.3 AND 1.4

When the ADC and ADC? commands were repeated at maximum speed, the interface was able to read only every second or even every third A/D conversion. This was due to an excessively long delay loop between some commands. This has been fixed by shortening the delay, so that the maximum speed of 2.5 conversions per second can be reached. (Previously, only the AVE command was able to use the maximum speed).

CHANGES BETWEEN VERSIONS 1.4 AND 1.5

If a printer is connected when booting, the unit will report the cumulative Eeprom write count. Writes to the Eeprom are limited to about 10000 times, so that no application should use this operation frequently. Checking the write count provides a way to ensure that there is e.g. no initialisation block in the software that is used often and every time sets either the PBD or IBA routinely.

If a printer is connected, the results of the *TST? self-test query are printed. The printed report contains more information than the integer response to the query.

CHANGES BETWEEN VERSIONS 1.5 AND 1.6

A bug, causing down-going spikes in one of the Picobus signals (DC), was fixed in the A47.TPU unit. This change does not affect operation in any way, but makes this bus signal cleaner.

CHANGES BETWEEN VERSIONS 1.6 AND 2.0

The computer is changed to 386SXi. Corrected bug that could hang the program if *TST? was used without a printer. Added queries for the EEPROM write count and for the PID parameters (asked from the IB, not from the TS-530A). This version was used for writing the LabView driver.

CHANGES BETWEEN VERSIONS 2.0 AND 2.1

Version 2.0 required that the AVS-47B be turned on first and the AVS47-IB only after that. This sequence guaranteed that the interface could initialize the bridge correctly for remote operation. Sometimes users have forgotten this sequence with the result that the instruments seemed to be defective. Version 2.1. eliminates the need for this sequence, so that the system should always become operative soon after having booted.

CHANGES BETWEEN VERSIONS 2.1 AND 2.2

One combination was still found, where the above improvement did not work. This was corrected.

CHANGES RELATED TO RESISTANCE BRIDGE REVISION AVS-47B

This manual is to be used with the AVS-47B. It takes into account that the new bridge revision has an isolated power supply for the output side of the Picobus primary computer interface, and that a new version “B” of the Picobus Cable is supplied with the resistance bridge. As a consequence, the default setup of new bridges is suitable for all interfacing options without changes in Picobus power jumpering.

The shield is no longer grounded at the AVS47-IB end of the Picobus Cable version “B”, which enables full galvanic isolation between the AVS-47B and AVS47-IB.

CHANGES BETWEEN VERSIONS 2.2 AND 2.3

A program bug prevented Picobus Delay factors (PBD) greater than 255 to be saved and restored correctly. This was fixed (2006-07-26).

CHANGES BETWEEN VERSIONS 2.3 AND 3.1

The embedded computer module was changed to CM420. The minimum PBD is now 30 (commands for setting a lower value are forced to value 30). Because of higher power consumption, the linear power supply was replaced by a switching unit and the box was equipped with a small cooling fan. Succesful booting and starting of the application program are indicated by blinking the “READY” light (previously this LED was initially blank and lighted when the program started). The EEPROM has been replaced by a CompactFlash (CF) memory unit. For compatibility, the name of the EWC command was retained, but now it keeps track of writes to the CF. The fuse is 1A slow for all mains voltages 90..250V. No setting of the mains voltage is required because of the wide input voltage range of the power supply.

CHANGES BETWEEN VERSIONS 3.1 AND 3.2

The the default PBD value of 45 turned out to be too small, and signals were too fast for the slow optical transmitters/receivers that are used in the PICOLINK option. The new default is 200. This is now also the minimum value for PBD. Any PBD command that uses a smaller argument is converted to PBD200. This should prevent problems, where PBD is inadvertently made too small and communication between AVS-47B and AVS47-IB fails.