# USER'S HANDBOOK

# 4920M

## Alternating Voltage Measurement Standard

datron
WAVETEK

Datron Instruments Ltd. 52 Hurricane Way, Norwich Airport, Norwich, Norfolk NR6 6JB, England.
Documentation Part No. 850261-1

MOD LEVEL 3

# USER'S HANDBOOK

for

# THE DATRON 4920M ALTERNATING VOLTAGE MEASUREMENT STANDARD

(for maintenance procedures
refer to the Servicing Handbook)

850261 Issue 1 (May 1990)

For any assistance contact your nearest Datron Sales and Service center.
Addresses can be found at the back of this handbook.

# DANGER

## HIGH VOLTAGE

**THIS INSTRUMENT IS CAPABLE OF DELIVERING**

**A LETHAL ELECTRIC SHOCK !**

**when connected to a high voltage source**

Input plugs carry the Full Input Voltage

**THIS CAN KILL !**

**Overvoltage can damage your instrument !**

Unless **you** are **sure** that it is **safe** to do so,

**DO NOT TOUCH**

any **terminals** connected to the **inputs**

# DANGER

# CONTENTS

Page

## SECTION 4 USING THE 4920M



**Section 4 Appendices**:



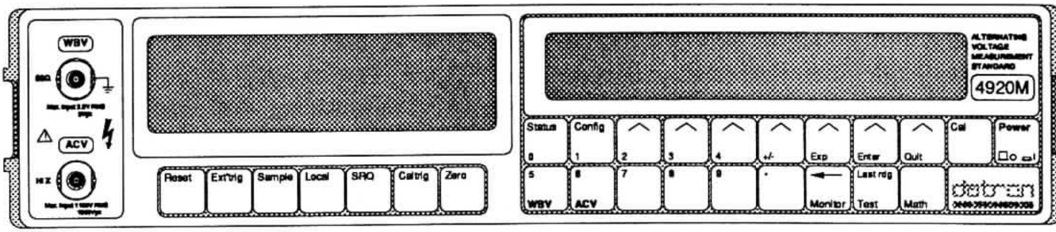
## SECTION 5 SYSTEMS APPLICATION VIA THE IEEE 488 INTERFACE



**Section 5 Appendices**:

# CONTENTS *(Continued)*

# SECTION 1
# Introduction and General Description




Designed with Standards and Calibration laboratories in mind, the 4920M provides extremely high performance in AC Voltage measurement, combined with ease of use.

## Standard Measurement Facilities

### Basic Configuration

The 4920M is a high-quality Alternating Voltage Measurement Standard. Its basic configuration offers the following measurement capabilities:

- **Display Resolution** - selectable 5 to 7 digit display resolution.
- **High Accuracy mode** - RMS AC Voltage measurement in eight ranges from 100mV to 1100V; 10Hz to 1MHz; 1-year specifications to ±30ppm.
  Input Impedance > 200Ω/V.
- **Wide Band mode** - RMS AC Voltage in two ranges from 100mV to 3.5V; 10Hz to 20MHz; 1-year specifications to ±1000ppm.
  Input VSWR < 1.02 : 1 at 50Ω.
- **Menu Control** - flexible and easy to use.
- **Calibration** - Autocal external calibration.
- **Remote Control** - Fully IEEE-488.2 programmable.

## 'Hard' and 'Soft' Keys - Menus

The use of hard keys (labels printed on the keys themselves) and soft keys (labels appear on the separate menu display) allows programming of the instrument into a wide range of configurations. Pressing the hard key of one of the main functions (WBV or ACV) alters the instrument circuitry to the selected function, at the same time displaying its own menu. Each soft key, marked with an arrowhead (∧), is labelled by the legend above it on the display. Whenever a main function key is pressed, the soft keys in its menu select only its ranges or autorange.

Once a main function is active, the Status hard key allows a check of configured parameters. Alternatively, the Config hard key can be used to alter the configuration. The Monitor key permits access to such information as the signal frequency of an AC input signal being measured; and to calculations of the difference or deviation between two measurements.

The menus are arranged in tree structures, leading the user through their branches to an end node, at which the physical circuitry or software selections of the instrument can be changed to suit the required parameters. For ease of use, each track from main function to end node involves the minimum number of user selections possible.

When the instrument power is switched on, all functions are forced into a safety default state. Once a function is configured to a required state it remains in that state, regardless of subsequent configurations in other functions, until either the state is changed or the instrument power is switched off.

As an easy introduction to the main function keys and their associated menus, users can follow a guided tour through the tree structures, sequenced in Section 3. The full range of facilities, together with access information, is detailed in Section 4.

## System Use

The 4920M is designed as standard to form part of a remote control system, conforming to IEEE 488.2 Standard Digital Interface. Remote control information is given in Section 5. This fulfils the Device Documentation Requirements of the standard (summary in Section 5 Appendix A).

## Self Test

The Test key displays a menu which provides access to a comprehensive series of self-tests. Among these are:

- An operational selftest;
- A diagnostic selftest;
- A test of the displays;
- A test of the front panel keys;

Details of these selftests can be found in Section 4.

## Calibration

### Autocal

The 4920M is an 'Autocal' instrument, providing full external calibration of all ranges and functions from the front panel; thus making the removal of covers unnecessary.

Periodically, the 4920M is electronically calibrated against traceable external standards, where any differences in the instrument's readings compared to the value of the external calibration sources can be used to derive calibration constants, which are stored by the instrument in non-volatile memory. These calibration corrections later serve to correct all readings taken by the instrument.

Calibration can also be carried out under remote control via the IEEE 488 interface.

### Calibration Security

A recessed Calibration Enable/Disable switch on the rear panel prevents accidental use of Autocal.

### Calibration Routines

The Routine Calibration procedures are described in Section 8 of this handbook.

## Message Readout

Generally, the selections offered in the menus reflect the availability of facilities, incompatible combinations being excluded. Nevertheless, the menu display doubles as a message screen, giving a clear readout of information to the user such as unsuitable attempts at configuration, test failures and some other conditions which would need to be reported to a Datron service center.

## Processor

The instrument is internally controlled by a 68000 series microprocessor. It ultimately translates all information, either from the front panel keys or from remote control, according to its program in firmware, into control signals which determine the instrument's operation.

## Computing

Measurements can be compared with manually-input data (or the most-recent measurement). Some of the keys under the Menu display double as a keyboard for setting the bus address, math constants etc.

Full details of the computing facilities are given in Section 4.

## Accessories

The instrument is supplied with the following accessories:

| Description | Part Number |
|---|---|
| Power Connector Cable MIL-C-28777 II | 920233 |
| IEEE 488 Connector Cable, 2m | 630366 |
| Power Fuse (230V use) 500mA SLO-BLO | 920084 |
| Power Fuse (115V use) 1.0A (L2080/1) SLO-BLO | 920116 |
| Hex Key 1.5mm AF (Handle removal) | 630284 |
| User's Handbook | 850261 |
| Calibration and Servicing Handbook (2 Volumes): | |
| Volume 1 | 850262 |
| Volume 2 | 850263 |

## Additional Documentation

The Calibration and Servicing Handbook contains information required to adjust and service the 4920M, in two volumes:

**Volume 1**:
Diagnostic data, maintenance information and circuit descriptions.

**Volume 2**:
Parts lists, layout drawings and circuit diagrams.

# Principles of Operation

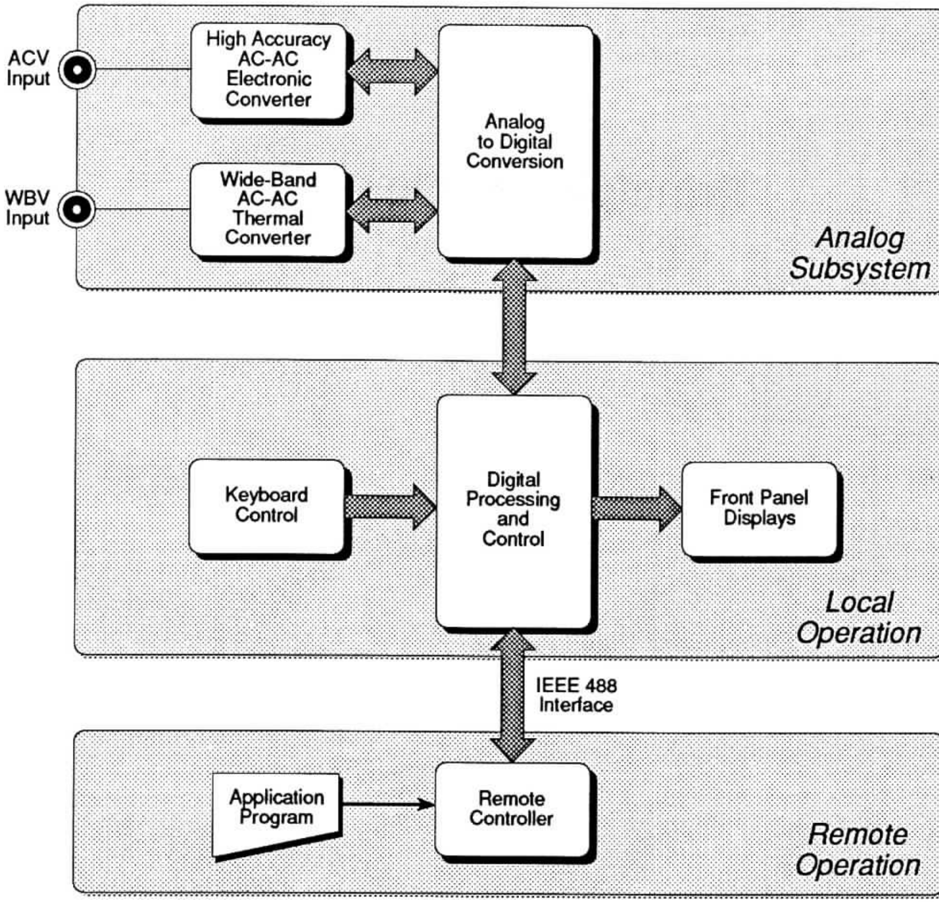



**FIG. 1.1 4920M BLOCK DIAGRAM**

# Principles of Operation

Figure 1.1 (opposite) shows the instrument's basic measurement functions.

## Precision DMM Design

The 4920M Alternating Voltage Measurement Standard is designed for calibration and standards laboratory applications, taking full advantage of the inherent qualities of critical accuracy-defining components to achieve its high performance. It also employs a method of calibration which is designed to maintain performance across the entire range of its functions.

## Basics

### Three-Reading Transfer

In both High Accuracy and Wide Band modes, each measurement consists of three separate readings in which the input RMS value is detected and processed via the A-D and microprocessor system, then fed back to calibrate the measurement process. The final result is transferred to the instrument's microprocessor for calibration and display.

### High Accuracy (ACV) 10Hz - 1MHz

AC voltages input via the front panel ACV socket are conditioned by a DC-coupled AC preamp, and presented to an electronic RMS detector. The DC 'RMS' level acts as reference for the generation of a 'Calculable AC Reference'. This represents the processed RMS value of the applied signal in an AC-AC transfer, during the three measurements.

### Wide Band (WBV) 10Hz - 20MHz

AC voltages are injected via the front panel WBV socket into a 50Ω input circuit, then applied to a vacuum thermocouple whose amplitude response has been characterized by a prior calibration. The three measurements recycle the readings to perform an AC-AC transfer based on an accurate DC reference level.

# High Accuracy (ACV) Operation

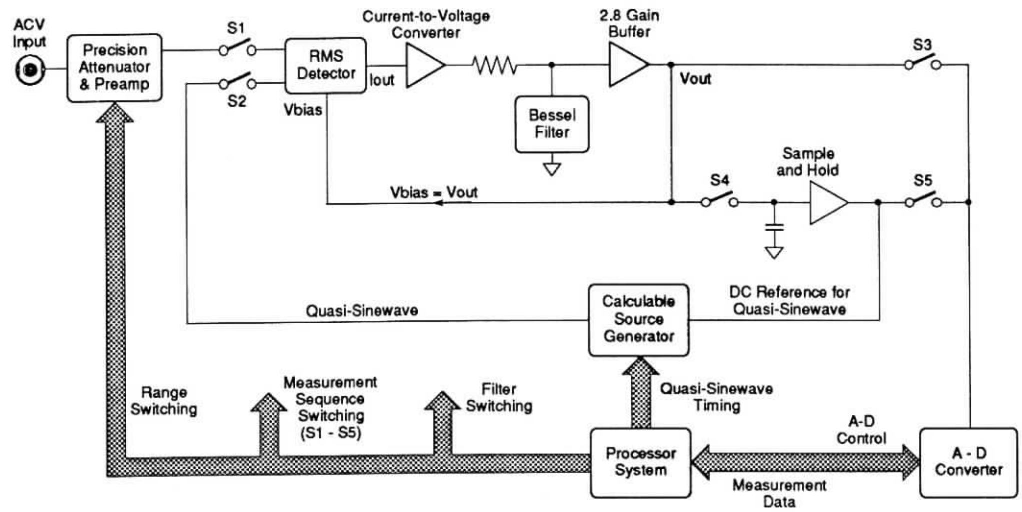



*FIG. 1.2 4920M HIGH ACCURACY (ACV) OPERATION*

## Attenuators and Preamp

**Basic Design** (Fig. 1.2)
The basic design incorporates a DC-coupled preamp, and has relatively high input resistance (400kΩ on the 1V and 1kV ranges; 124kΩ on the 10V and 100V ranges). The gain-defining resistor chains are guarded, and the remaining time constants are set above 1MHz so that hardware trims are not required.

### Range Selection

There are three main attenuators: 1kV/300V permanently connected to sense overload; 100V/30V/10V/3V switched in parallel. 1V/300mV in unity gain buffer in series with the 1kV/300V attenuator.

### Attenuator Elements

The attenuators, which configure the amplifier gain to define the high-accuracy AC Voltage ranges, consist of extremely stable metal foil resistors, packaged in large hermetic-seal cases. Similar types are used in the 1V range protection chain.

To ensure that no spurious leakage currents cause linearity, temperature-coefficient or drift problems in the attenuator chains, the sealed cases form a guard at HF, driven by a capacitor chain. Within the cases of the attenuator elements, the resistor values are split and the junctions guarded to effect a high degree of frequency flatness.

## RMS Converter

### DC Coupled Preamp

In order to minimize the input capacitance to the preamp, reed relays are used to select the signal paths for the various ranges. The preamp bootstraps extensive pcb tracking to guard the attenuators, input tracking and reed relays; it also drives a bootstrap buffer which forces the preamp power supplies to follow the input signal level by reference to bootstrap. The preamp thus sees no change in input signal relative to its supplies, again minimizing its input capacitance and achieving very high common mode rejection.

### Separate PCB

The whole attenuator and preamp circuit is mounted on a separate sub-assembly, fabricated with PTFE board, and mounted above the main High Accuracy ACV printed circuit assembly.

The preamp is run at unity or X3 gain depending on range selection, with corresponding compensation switching. Its output voltage is fed to the RMS Converter on the High Accuracy ACV printed circuit assembly, from which the compensation current for the preamp signal current into the signal ground (0V-4) is derived.

### Protection

The instrument can measure up to 1100VRMS and can withstand a continuous overload of 1100VRMS or 1556V peak. Overload is sensed by a resistor at the low end of the 1000V/300V attenuator; and protection is activated from the HIACC_AC pcb. Two series resistors, referred to two zener diodes, absorb the overload energy for up to 1 second, by which time the protection system will have disconnected the instrument input from both the 100V/30V/10V attenuator and the preamp input.

### Conversion Process

RMS conversion is based on a modified analog multiplier, consisting of a 'squaring' logarithmic amplifier whose output is buffered into a balanced exponential amplifier.

The DC current output of the squaring circuit 'Iout' is proportional to '$Vin^2$' if a fixed source-current bias is applied to the logging and antilogging elements. The transfer function is

$$\mathbf{Iout \propto Vin^2 / (R^2 \times Ibias)}$$

where **R** is the common value of balanced source resistors and **Ibias** is the common source-bias current for the logging and antilogging circuits.

Iout is converted into a DC voltage **Vout** which ultimately becomes the RMS Converter output. However, in order to apply the 'square-root' element of the computation, this voltage is fed back to provide a current bias whose amplitude follows Iout, and this forces Iout to a value proportional to $\sqrt{(\mathbf{Vin}^2)}$.

The transfer function now becomes

$$\mathbf{Iout \propto Vin^2 / (R^2 \times Iout)}$$

as R is constant this leads to $\mathbf{Iout^2 \propto Vin^2}$, and to $\mathbf{Iout \propto \sqrt{Vin^2}}$. After current-to-voltage conversion and filtering, this gives $\mathbf{Vout \propto \sqrt{Vin^2}}$ as the DC output voltage fed to the A/D, subject only to calibration.

# Transfer Loop

## Transfer Process

Three measurements are made to determine the precise RMS value of the signal. The first is an estimate (to about 1%) which is a function of the gain of the RMS Converter, and the other two are used to determine that gain and then apply corrections to the first measurement.

## Calculable Source Generator

The Sample-and-Hold circuit provides a memory of Vout, to be used as reference for the Calculable Source Generator. This in turn constructs a 'quasi-sinewave' whose amplitude and form factor are known, to ensure that the transfer is a true AC to AC process.

## Transfer Sequence Switching

Figure 1.2 shows the arrangement of the elements in the loop; the positions of switches S1-S5 are altered by firmware to generate the sequence of the three measurements:

**First Measurement (M1):**

The uncorrected RMS value of the input signal (Vsig) is measured directly by the A-D and processor. The DC analog value of Vout (Vref) is memorized in the Sample-and-Hold circuit:

| | |
|---|---|
| **S1 closed** | drives the RMS Converter from Vsig; |
| **S2 open** | prevents the quasi-sinewave from interfering with the measurement; |
| **S3 closed** | the A-D and processor evaluate the first estimate; |
| **S4 closed** | Vout is sampled; |
| **S5 open** | removes Vref from the direct measurement of the input. |

**Second Measurement (M2):**

Vref from the previous measurement (M1) is measured by the A-D and processor. Meanwhile, the quasi-sinewave (resulting from Vref) is applied to the RMS converter to allow settling for the next measurement (M3).

| | |
|---|---|
| **S1 open** | Vsig is removed; |
| **S2 closed** | drives the RMS Converter from the the quasi-sinewave (settling only); |
| **S3 open** | removes Vout from the measurement; |
| **S4 open** | freezes Vref; |
| **S5 closed** | the A-D and processor measure Vref. |

**Third Measurement (M3) :**
The quasi-sinewave is measured by the RMS converter, A-D and processor.

| | |
|---|---|
| **S1 open** | Vsig is not applied; |
| **S2 closed** | drives the RMS Converter from the the quasi-sinewave for the measurement; |
| **S3 closed** | applies Vout for the quasi-sinewave to the A-D and processor for measurement; |
| **S4 open** | Vref remains frozen; |
| **S5 open** | prevents Vref from interfering with the RMS measurement of the quasi-sinewave; |

## Final RMS Computation

If the gain of the RMS Detector, Filter and Buffer is G (not precisely known), then:

M1 = G.Vsig(RMS);
so Vsig(RMS) = M1 / G.

M2 and M3 are combined to determine G very precisely:

G = Transfer Measurement / Transfer Reference = M3 / M2.

This is used to eliminate G:

**Vsig(RMS) = M1.M2 / M3.**

# Wide Band (WBV) Operation

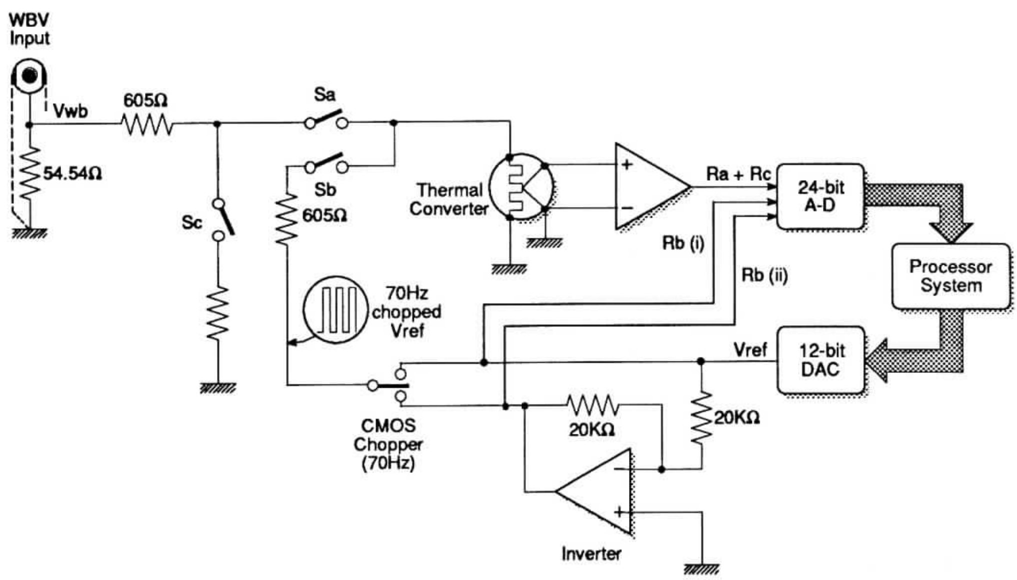



***FIG. 1.3 4920M WIDE BAND (WBV) OPERATION***

## Basic Design (Fig 1.3)

The design is based on a three-part transfer sequence which first registers the raw reading in digital form, then internally calibrates the gain of the conversion circuit, using a known value close to the registered raw reading. This calibration also accounts for ageing of the Thermal Converter. An accurate correction factor is derived, in digital form, and applied to the digital raw reading. The final result is displayed.

### WBV Analog Input Elements

The input is applied via a 50Ω coax. to the WBV N-type input socket, which itself is matched to 50Ω by internal resistors. The signal is applied via a limiting resistor directly to a precision vacuum thermocouple, whose junction DC voltage is taken as an analog of the input RMS voltage value. This is amplified and buffered, then its value is converted into a digital word by the 24-bit A-D during the first measurement in the sequence. This word represents the raw value in digital form.

### Digital Processing

The processor derives a second digital value close to the raw value, which it outputs to the 12-bit D-A during the second measurement of the transfer sequence. This and its inverted version are passed directly to the 24-bit A-D, which records its value as the two inputs to the chopper circuit. On the third measurement the chopper output is passed through the thermal converter and is also digitized in the A-D. The results from all three transfer measurements are combined in calculations which include correction factors based on the known performance of the thermal converter. The final result is passed to the digital display.

### Transfer Analog Elements

The measured value of the D-A output is applied to a chopping circuit, comprising an inverter and an electronic analog switch. The chopped signal is applied to the thermal converter during the third measurement to establish the gain of the input circuit. The chopped Vref has a crest factor of 1, which because it was derived from the RMS estimate of Vwb, represents the RMS value of Vwb during the third measurement.

## Transfer Sequence (Refer to Fig. 1.3)

Three readings are made to determine the precise RMS value of the signal. The first is an estimate (to about 1%), which is a function of the gain of the Thermal Converter, and the other two are used to determine that gain and then apply corrections to the first reading. Figure 1.3 shows the arrangement of the elements in the loop; the positions of switches Sa and Sb are altered by firmware to generate the sequence of the three readings. Switch Sc closes as Sb closes and Sa opens, providing a matched termination at the WBV input when the thermal converter is disconnected by Sa.

**First Reading (Ra):**
The uncorrected RMS value of the input signal (Vwb) is allowed to settle for 10-12 seconds, and then measured directly by the A-D and processor (average of 4 readings).

**Sa closed** drives the Thermal Converter from Vwb;
**Sb/Sc open** prevents the chopper output from interfering with the reading.

**Second Reading (Rb):**
Vref, derived from (and within 1% of) the first reading Ra, is output via the 12-bit D-A and applied to the inverter input. The thermal converter input is sourced from the chopper output so that it can settle to 1% of the change of input value.

**i.** Vref is measured by the A-D and processor (average of 4 readings);

**ii.** The inverter output is also measured (average of 4 readings) so that any offsets from the inverter can be removed digitally in the final calculation.

**Sa open** Vwb is removed;
**Sb/Sc closed** the chopped Vref is applied to the thermal converter (approx. 7 seconds settling).

**Third Reading:**
**(Rc)** The chopped Vref is measured by the thermal converter, A-D and processor (average of 4 readings).

**Sa open** Vwb is not applied;
**Sb/Sc closed** drives the thermal converter from the chopped Vref for the reading;

## WBV Transfer Sequence (Contd.)

### Final RMS Computation

If the gain of the system is G (not precisely known), then:

Ra = G.Vwb(RMS);

so Vwb(RMS) = Ra / G.

Rb and Rc are combined to determine G very precisely:

G = Transfer Reading / Transfer Reference
= Rc / Rb.

This is used to eliminate G:

**Vwb(RMS) = Ra.Rb / Rc.**

### Linearity Calibration

For the above sequence to work, the two thermal element measurements Ra and Rc must be at almost the same value so that non-linearity does not give rise to differential gain errors. The *shape* of the thermal converter gain response remains virtually constant over time, but its *gain* changes with age. A special internal calibration is performed at manufacture to about 0.1%, which steps through a number of DAC values with Sb closed, in order to characterize the thermal element for subsequent linearity correction.

Thus each reading which involves the thermal converter is corrected for linearity during the calculations. Any gain drift due to ageing is corrected for *every* transfer sequence, by measuring the gain of the thermal element as part of the system gain during readings Rb and Rc.

# Analog to Digital Converter

### Introduction

The instrument's analog-to-digital converter (A-D) takes the form of a highly linear, low noise, fast and flexible multislope integrator. Timing, counting and control are executed by a custom 'Application-Specific Integrated Circuit' (ASIC), resulting in a design which offers both variable integration times and user-selectable resolutions. A simplified schematic is given as Fig. 1.4. The A-D is involved in every measurement in ACV and WBV operation.

### Multislope Operation (Figs. 1.4 and 1.5)

Multislope operation permits the integration capacitor value to be smaller than normally required for a more conventional circuit, greatly reducing problems due to dielectric absorption. Reference switching errors are reduced to a constant value, which are subtracted from the reading by the instrument's microprocessor. A further benefit is that both the signal and the reference may be applied to the integrator simultaneously, greatly reducing the conversion time. A digital autozero system avoids the need for the more common sample-and-hold type of autozero circuit.

The timing and counting considerations with this design of A-D are quite complex. Programmable delay timers, a ramp timer and a counter for the number of completed ramps exercise great control flexibility over its performance. All of these timers and counters are integrated into a custom ASIC which has a 32 bit control register, programmed by the instrument's microprocessor via a special serial interface. The same serial loop is used to transmit the reading from the ASIC to the processor for calibration and display.

## Features

The result is a compact A-D with the following features:

- Excellent rejection of normal-mode power-line interference (Integration time is fixed at 200ms, an exact multiple of 60Hz and 50Hz line supply periods).
- Excellent linearity of 0.2ppm of full scale.
- Low noise of < 0.05ppm of full scale.

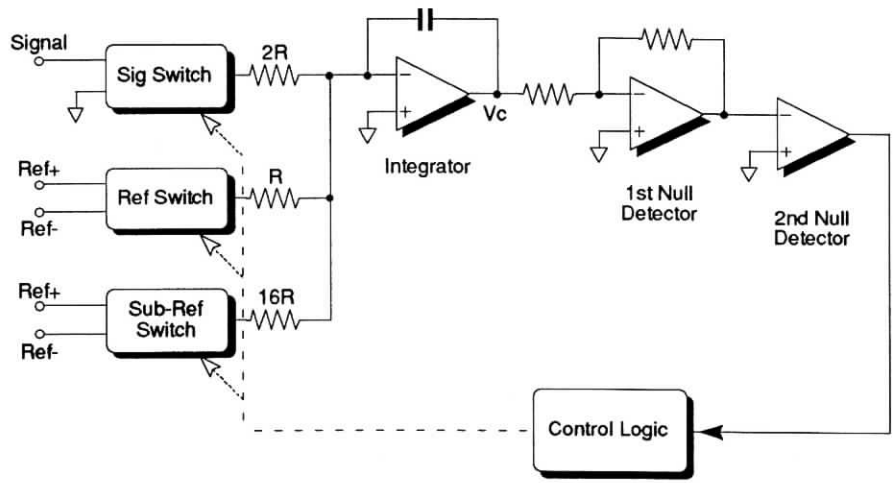



*FIG. 1.4 MULTISLOPE OPERATION - SIMPLIFIED A-D SCHEMATIC*

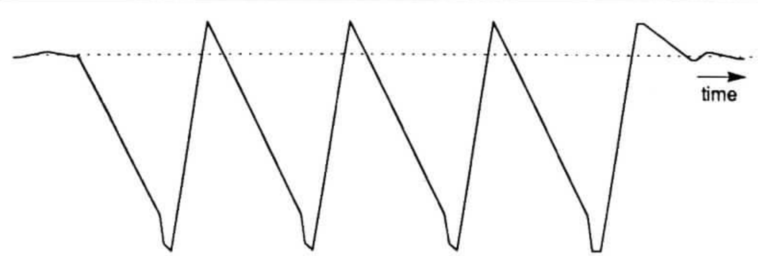



*FIG. 1.5 MULTISLOPE OPERATION - SIMPLIFIED A-D WAVEFORM*

# A-D Master Reference

### Reference Module

The reference used in the analog to digital conversion is derived from a specially conditioned zener reference module. It contains the reference device and its associated buffer circuits, which are all hermetically encapsulated together in order to ensure constant temperature across the module. The module is stable to within ±4ppm per year, produces pk-pk noise less than 0.1ppm, and has a temperature coefficient better than 0.15ppm/°C. This temperature coefficient is held over a temperature span of 0°C to 70°C, and the reference exhibits negligible temperature shock hysteresis.

### Module History

Extensive evaluation of successive reference modules has resulted in a burn-in process which equates to an ageing of 1 year, reducing infant mortalities and improving stability. Following this process, all reference modules are checked over a temperature span of 0°C to 70°C for temperature performance, and then monitored for long term drift over a period of three months minimum.

# Processor

The instrument is internally controlled by a 68000-series microprocessor. It ultimately translates all information from the front panel keys, according to its program in firmware, into control signals which determine the instrument's operation.

# SECTION 2
# Installation and Operating Controls

This section contains information and instructions for unpacking and installing the Datron 4920M Selfcal Digital Multimeter. It also introduces the layout of controls on the instrument.

## Unpacking and Inspection

Every care is taken in the choice of packing material to ensure that your equipment will reach you in perfect condition.

If the equipment has been subject to excessive handling in transit, the fact will probably be visible as external damage to the shipping carton.

In the event of damage, the shipping container and cushioning material should be kept for the carrier's inspection.

Unpack the equipment and check for external damage to the case, sockets, keys etc. If damage is found notify the carrier and your sales representative immediately.

Standard accessories supplied with the instrument should be as described in Section 1.

## Calibration Enable Switch S2

### CAUTION

This two-position rear-panel switch protects the instrument calibration memory.

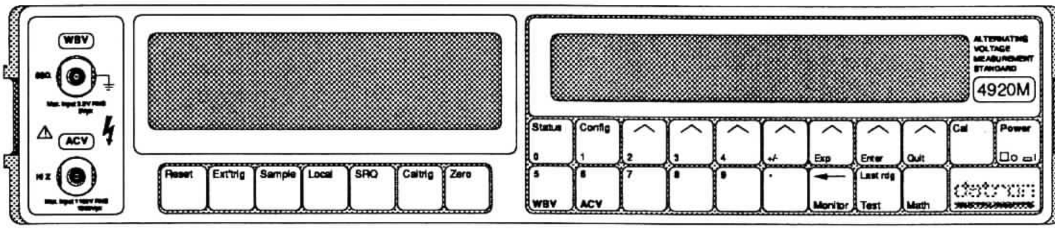
The instrument was initially calibrated at the factory, so the switch should always remain set to DISABLE, until immediate recalibration is intended.

### For Recalibration:

If the external calibration menu is selected while the key is not in the enabling position, the menu is replaced by the warning message:

**1002: calibration disabled**

# Introduction to the Front Panel




The two displays on the front panel deal with different aspects of operation. We set up the instrument's configuration using menus shown in the right-hand (dot-matrix 'menu') display, then readings appear in the left-hand ('main') display.

Beneath the dot matrix display, all keys other than the Power key are associated with menus. The keys beneath the main display are direct action keys, associated with triggers, remote control, and instrument reset.

## Menu Keys

There are two classes of front panel menu keys, those that lead to an immediate change of instrument state (i.e the major function keys WBV, ACV), and those that do not (Status, Config, Cal, Monitor, Test, Math).

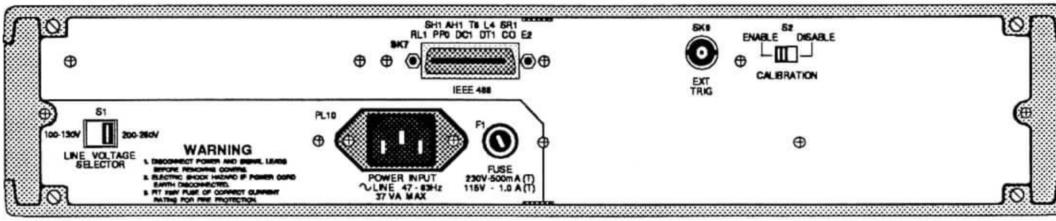
## Numeric Keyboard

Seventeen of the menu and soft function keys also act as a keyboard for entry of parameters such as math constants, bus address, etc. The data entered is purely numeric, and can consist of either a keyboard-entered value or the value of the most recent reading.

## Major Function Keys: WBV, ACV

Each of these keys defines a separate measurement state and activates its corresponding menu on the dot matrix display. Changing a selection alters the measurement state.

# Introduction to the Rear Panel




## Mechanical Access

The top or bottom cover is released for removal by undoing two screws visible at the rear. A single screw retains the corner block which covers the handle mechanism on each side panel.

## Labels

The rear panel displays the identification label for the instrument, and a modification label.

## External Connections

Apart from the front input sockets, connections to the internal circuitry enter via the rear panel.

SK7 is the standard IEEE 488 connector. A list of interface function subsets is printed next to the connector.

SK9 provides a coaxial BNC trigger input.

## Fuses

The fuse adjacent to the power input plug protects the power input line.

## Voltage Selector

The recessed power line voltage selector adapts the instrument to either 115V or 230V line inputs.

## Calibration Switch

To calibrate the instrument, special menus are available from the front panel. But to enter these menus it is necessary to set the calibration Switch on the rear panel to ENABLE.

# Preparation for Operation

**DANGER**
**THIS INSTRUMENT IS CAPABLE OF DELIVERING A LETHAL ELECTRIC SHOCK IF IT IS CONNECTED TO A HIGH VOLTAGE SOURCE. THE WBV AND ACV CONNECTORS ARE MARKED WITH THE SYMBOL TO WARN USERS OF THIS DANGER.**

**UNDER NO CIRCUMSTANCES SHOULD USERS TOUCH ANY LEADS CONNECTED TO THE INPUT CONNECTORS UNLESS THEY ARE FIRST SATISFIED THAT NO DANGEROUS VOLTAGE IS PRESENT.**

## Power Cable

The detachable power supply cable, provided with the instrument, comprises eight feet of 3-core PVC sheath cable; at one end moulded to a Type II, 3-pin equipment connector, and at the other to a 15A, 125V connector. It fits into a plug receptacle (PL10 - incorporates a filter) at the rear of the instrument and should be pushed firmly home.

Alternatively, the receptacle PL10 can be used to receive a Type II 250V supply connector, providing that the voltage selector and power fuse have been changed for the higher voltage.

The supply cable should be connected to a grounded outlet ensuring that the Ground lead is connected. Internally, the Brown lead is connected to Line, Blue lead to Neutral, and Green/Yellow lead to Ground.

## IEEE 488 Bus Connector

This detachable connector comprises two metres of multicore-core PVC sheath cable with standard IEEE-488.1 fittings. It fits into a socket (SK7) at the rear of the instrument. Refer to Section 5 for full details of IEEE-488 operation.

## Fuses

### Power Fuse:

The power fuse F1 is situated next to the power input plug on the rear panel. It should be of the anti-surge type. Its rating is dependent on the supply voltage:

for 100V to 130V - 1.0A SLO-BLO,
for 200V to 260V - 500mA SLO-BLO.

MAKE SURE THAT ONLY A FUSE WITH THE REQUIRED RATED CURRENT AND OF THE SPECIFIED TYPE IS INSERTED AS REPLACEMENT.

AVOID THE USE OF MENDED FUSES AND DO NOT SHORT-CIRCUIT THE FUSE HOLDERS. SUCH PRACTICES ARE DANGEROUS; AND WILL RENDER THE WARRANTY VOID.

## Line Voltage

### Voltage Selector and Line Fuses - 115V

When shipped, the instrument is packed ready for use with 100V to 130V, 60Hz supplies. The legend '115V' will be visible in the window of the line voltage selector switch (S1) on the rear panel, and the fuse F1 will be rated at 1.0A.

### Changing Supply Voltage to 230V

For 200V to 260V supply; the voltage selector switch S1 must be moved so that '230' is visible in the window, and the fuse rating must be reduced to 500mA.

### Automatic 50-60Hz Operation

The 4920M is capable of operation at any line frequency from 47Hz to 63Hz, so no means of line frequency selection is necessary, or provided.

## Mounting

### Bench Use:

The instrument is fitted with rubber-soled plastic feet and tilt stand. It can be placed flat on a shelf or tilted upwards for ease of viewing.

### Rack Mounting:

The 4920M is basically a bench-mounted instrument, but a kit is available which adapts the instrument to standard 19 inch rack fixings.

# Connectors and Pin Designations

## Inputs

Two N-type coaxial plugs are fitted on the left of the front panel. Their functions are as follows:

**WBV** 50Ω input to the wideband WBV analog circuitry.

**ACV** High impedance input to the high-accuracy ACV analog circuitry.

## SK9 - External Trigger Input

This co-axial BNC socket on the rear panel can be used to trigger a measurement when external triggers are enabled. The single pin is pulled up internally to +5V, and requires a negative-going TTL edge to initiate the reading.

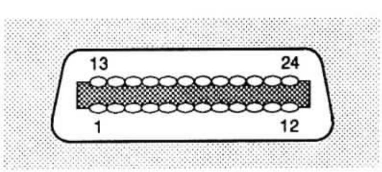
## SK7 - IEEE 488 Input/Output

### Compatibility

The IEEE input/output is a 24-way Amphenol connector which is directly compatible with the IEEE 488 interface and the IEC 625 Bus.

Note that the Bus Address is set from the front panel (refer to Section 5).

### Pin Layout




### SK7 Pin Designations

| Pin No. | Name | Description |
|---|---|---|
| 1 | DIO 1 | Data Input/Output Line 1 |
| 2 | DIO 2 | Data Input/Output Line 2 |
| 3 | DIO 3 | Data Input/Output Line 3 |
| 4 | DIO 4 | Data Input/Output Line 4 |
| 5 | EOI | End or Identify |
| 6 | DAV | Data Valid |
| 7 | NRFD | Not Ready For Data |
| 8 | NDAC | Not Data Accepted |
| 9 | IFC | Interface Clear |
| 10 | SRQ | Service Request |
| 11 | ATN | Attention |
| 12 | SHIELD | Screening on cable (connected to 4920M safety ground) |
| 13 | DIO 5 | Data Input/Output Line 5 |
| 14 | DIO 6 | Data Input/Output Line 6 |
| 15 | DIO 7 | Data Input/Output Line 7 |
| 16 | DIO 8 | Data Input/Output Line 8 |
| 17 | REN | Remote Enable |
| 18 | GND 6 | Gnd wire of DAV twisted pair |
| 19 | GND 7 | Gnd wire of NRFD twisted pair |
| 20 | GND 8 | Gnd wire of NDAC twisted pair |
| 21 | GND 9 | Gnd wire of IFC twisted pair |
| 22 | GND 10 | Gnd wire of SRQ twisted pair |
| 23 | GND 11 | Gnd wire of ATN twisted pair |
| 24 | GND | 4920M Logic Ground (internally connected to Safety Ground) |

# SECTION 3 Basic Measurements

This section introduces the basic 'User Interface' of the 4920M, describing how to make straightforward measurements without recourse to the more advanced features of the instrument. Descriptions of these other features can be found in Part 2, Section 4.

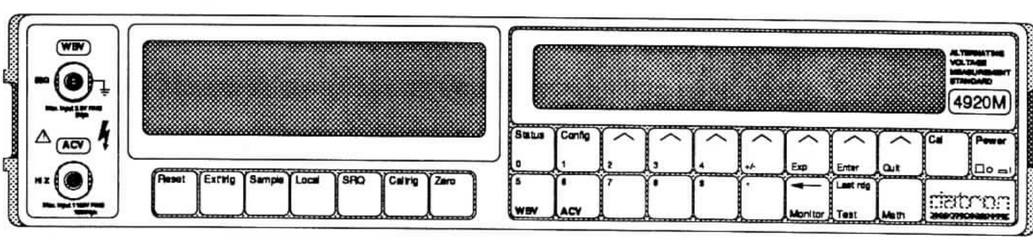
## The Measurement Task

With the external circuit properly connected, any measurement requires us to take two actions:

1. Configure the instrument;
2. Trigger the measurement and read the result.

The 4920M allows us to choose from many actions to control these processes. As an introduction, we shall concentrate on the selections for taking basic measurements. These are not complicated - all we need to do is to work through the instrument's selection menus.

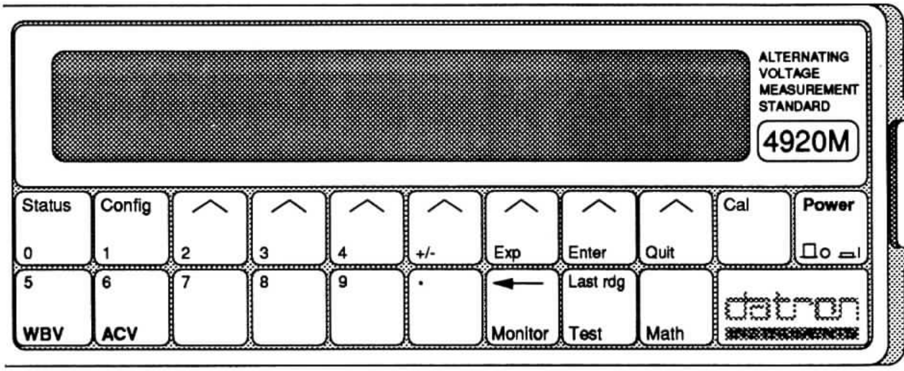
## Introduction to the Front Panel




The two displays on the front panel deal with different aspects of operation. We set up the instrument's configuration using menus shown in the right-hand (dot-matrix) display, then readings appear in the left-hand (main) seven-segment display.

Beneath the dot matrix display, all keys other than the Power key are associated with menus. The keys beneath the main display are direct action keys, associated with triggers, remote control, and instrument reset.

# Menu Keys




There are two classes of front panel menu keys; those that lead to an immediate change of instrument state (i.e the major function keys **WBV**, **ACV**), and those that do not (Status, Config, Cal, Monitor, Test, Math).

As well as the menu selection keys, there are seven soft function selection keys which have different actions depending on the selected menu. An arrowhead printed on each soft key lines up with a label which defines the action of the key.

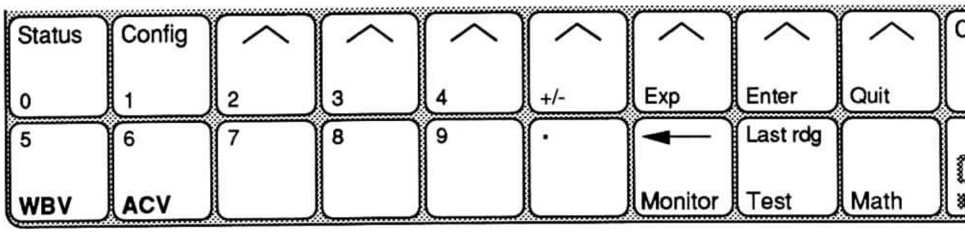
Also, system messages (all in capitals) may appear, these assist to clarify operation.

The labelled soft keys have actions which fall into the following classes:

- Select another menu.
- Enable or disable a facility (e.g. Range selection). When enabled, the soft key label is underlined by a cursor.
- Trigger a direct action (e.g. 'Oper' in the TEST menu activates an operational selftest).

An error message appears if a selection cannot be executed.

## Numeric Keyboard



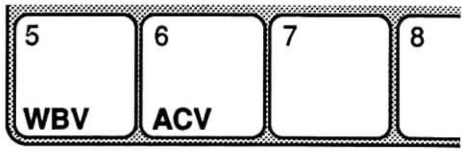

Some menu and soft function keys, shown above, also act as a keyboard for entry of parameters such as math constants, bus address, etc. The data entered is purely numeric, and can consist of either a keyboard-entered value or the value of the most recent reading.

## Exit from Menus

We can generally exit from any menu by selecting another menu key. For those menus where the keyboard is active, we can exit by pressing either Enter or Quit. For some menus, a special soft key permits exit by a single keystroke.

# Major Function Keys:

**WBV (Wide Band Voltage) -** 10Hz to 20MHz, max. input 3.5V RMS @ 50Ω
**ACV (AC Voltage)** - 10Hz to 1MHz, max. input 1000V RMS @ high impedance




Each of these function keys defines a separate measurement state and activates its corresponding menu on the display. Changing a selection therefore commands a change of measurement state.

Each function has its associated CONFIG (Configuration) menu, which we can use to set up 'function-dependent' parameters such as resolution and filter settings. Once set up, the instrument remembers the pattern of parameter conditions in that function, so that when we reselect it on a later occasion, it remains set up as before until we change it or turn off the instrument power.

## Initial State at Power On

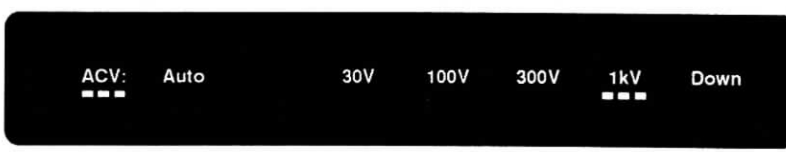
To see this condition, ensure that the instrument has been correctly installed in accordance with Section 2.

Operate the **Power** switch on the front panel.

The 4920M performs an operational self-test and then forces the following state:

| | |
|---|---|
| ***Function*** | ACV |
| ***Range*** | 1kV |
| ***Resolution*** | 1ppm (Res7) |
| ***Filter*** | 100Hz |
| ***Monitor*** | Off |
| ***Math*** | Off |

Observe the **ACV Menu**:
1kV is underlined, showing the active selection. It can be cancelled by any other range selection, or by pressing the Auto key. Ranges themselves cross-cancel.




Leave the power switched on.

We have next to distinguish between three main types of action built into the operation of the soft keys. These are defined overleaf, together with the shorthand conventions we use in the quick tour to refer to them.

## Soft Key Conventions

Now look at the soft keys (the ones with the arrowheads) to make some distinctions in a little more detail. Each soft key's action is defined by the legend presented above it on the display. The legends usually define three different types of soft key:

*Choice* key — Chooses one of several possible states. Deselection is by cross-cancelling, i.e. by selecting another state.
(The ranges on the ACV menu are *Choice* keys).
**cursor underline** indicates 'active',
**no cursor** indicates 'not active'.

*Toggle* key — Activates a particular facility - a second press when its state is active will cancel it.
('Filt' on the WBV CONFIG menu is a *Toggle* key).
**cursor underline** indicates 'active',
**no cursor** indicates 'not active'.

*Menu* key — Activates another menu - cursor not used. The whole aim of branching via a menu is to gain access to further grouped state keys at an end of the branch.
('Filt' on the ACV CONFIG menu is a *Menu* key).

**N.B.** When introducing soft keys in this text we shall differentiate between the three types (to avoid lengthy paragraphs) as follows:

| | | |
|---|---|---|
| *Choice* key | Underlined | e.g. <u>300mV</u> |
| *Toggle* key | Underlined italic | e.g. <u>*Filt*</u> |
| *Menu* key | Not underlined | e.g. Filt |

Note that this is purely a short method of identifying the type, and bears no relation to its physical appearance on the instrument. Some displays do not relate to soft keys.

## Quick Tour of the Major Function Menus

The following introduction takes the form of a quick tour of the main functions, starting from Power On. To relate the descriptions to the physical appearance, process through the sequence as indicated by the pointer ( ☞ ).

### ACV Menu (See the figures on pages *3-2* and *3-5*)

This menu defines the following *choice* and menu keys.

Auto — The range it makes active is also underlined. As well as being cancelled by any range selection, it can also be cancelled by re-pressing the Auto key itself (in this case the instrument reverts to the auto-selected range).

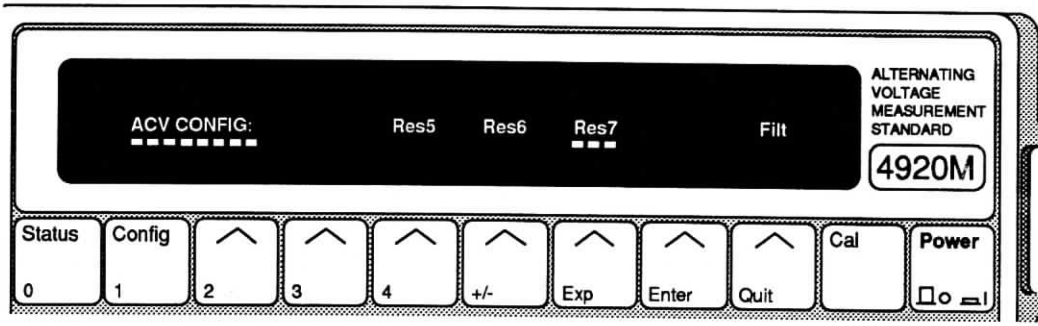
| | | | | | |
|---|---|---|---|---|---|
| Low Voltage Ranges: | 0.3V | 1V | 3V | 10V | Up |
| High Voltage Ranges: | 30V | 100V | 300V | 1kV | Down |

Up — on the low voltage menu. When pressed selects the high voltage menu.
Down — on the high voltage menu. When pressed selects the low voltage menu.

## ACV Configuration

*(Resolution and Filtering)*

☞ Press the Config key to see the ACV CONFIG menu:




Res5: Accommodates 100ppm resolution for all readings on the range.

Res6: Accommodates 10ppm resolution for all readings on the range.

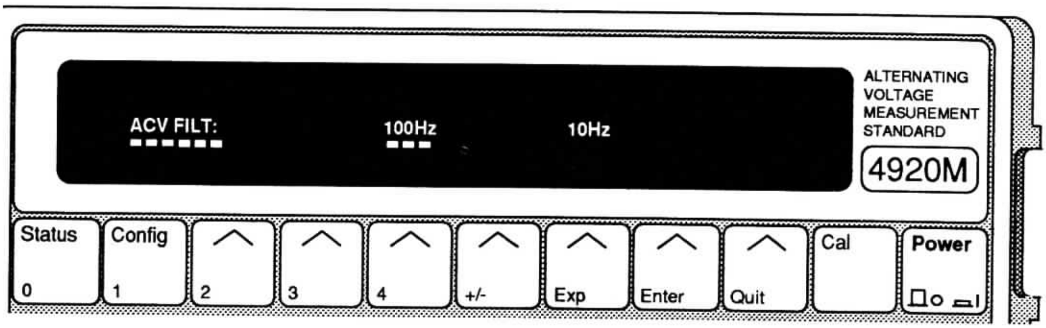
Res7: Accommodates 1ppm resolution for all readings on the range.

At Power On, Res7 is active.

Filt: Selects the ACV FILT menu.

## ACV Filtering

☞ Press the Filt soft key to see the ACV FILT menu:



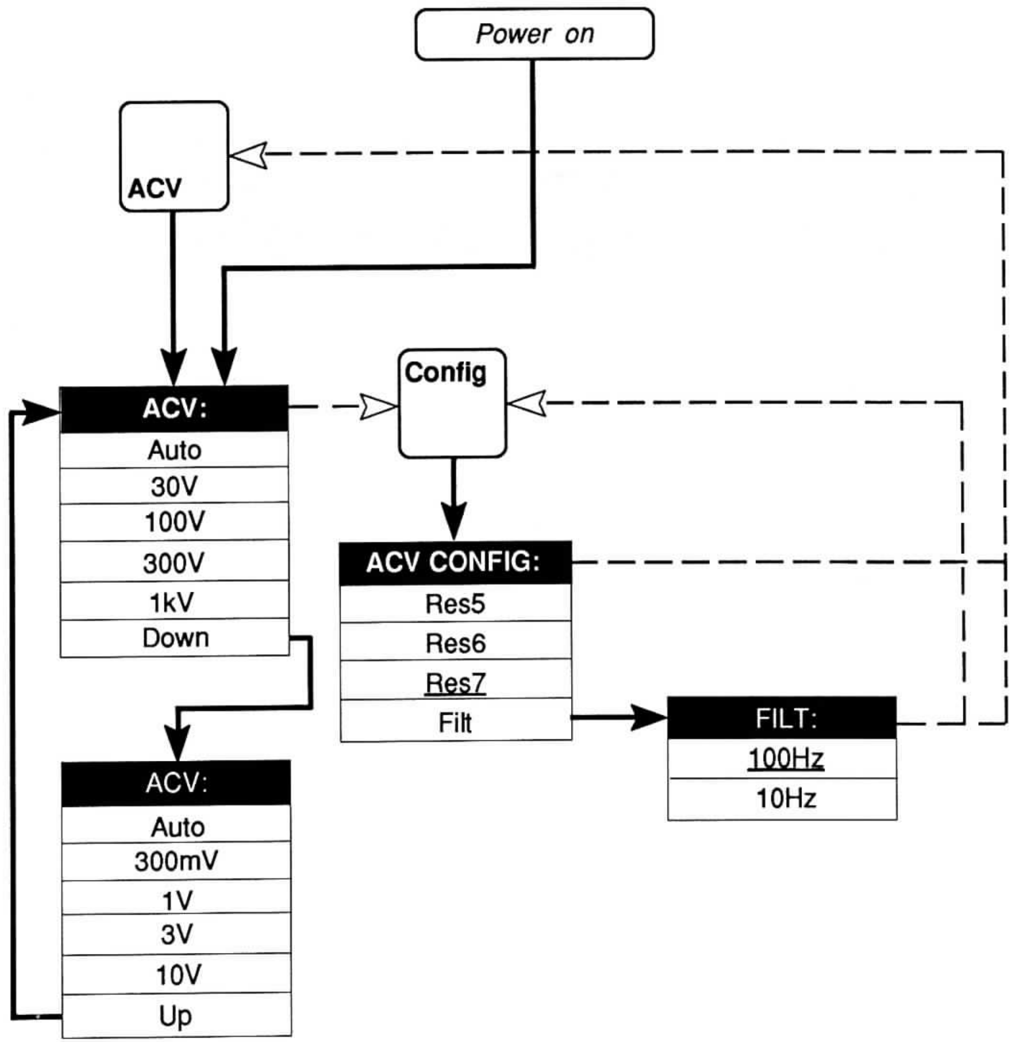

<u>100Hz</u>: Includes a 100Hz low-pass analog RMS integration filter.

<u>10Hz</u>: Includes a 10Hz low-pass analog RMS integration filter.

<u>100Hz</u> is selected at Power On. One filter is always included.

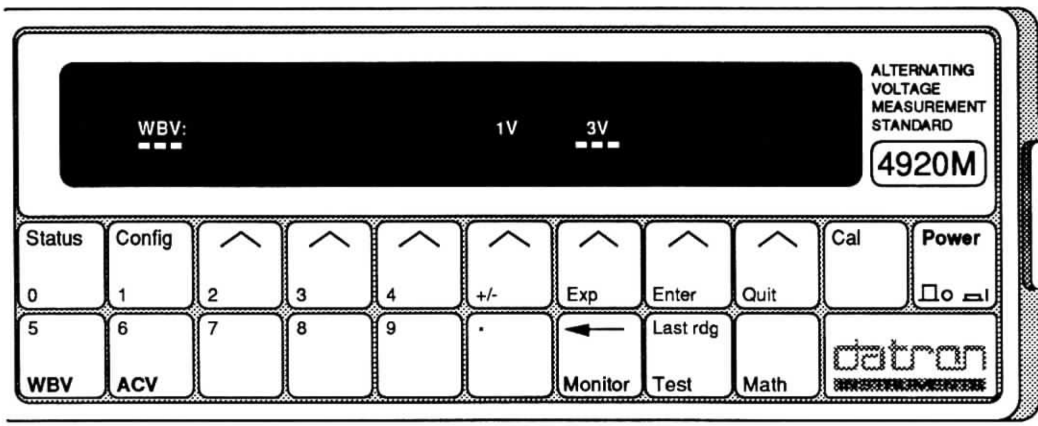
The menu can be exited by selecting either of the main function keys.

## ACV (High Accuracy AC Voltage) - Movement between Menus



# WBV Menu

☞ Press the **WBV** key to see the **WBV** menu:




This menu defines the following *choice* keys.

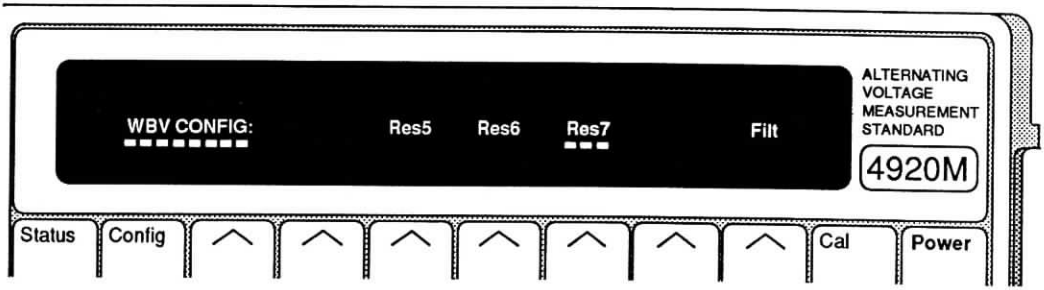
Ranges: 1V 3V

At Power On, 3V is selected, but not active.

## WBV Configuration

*(Resolution)*

☞ Press the Config key to see the WBV CONFIG menu:




This menu defines the following *choice* keys.

Res5: Accommodates 100ppm resolution for all readings on the range.

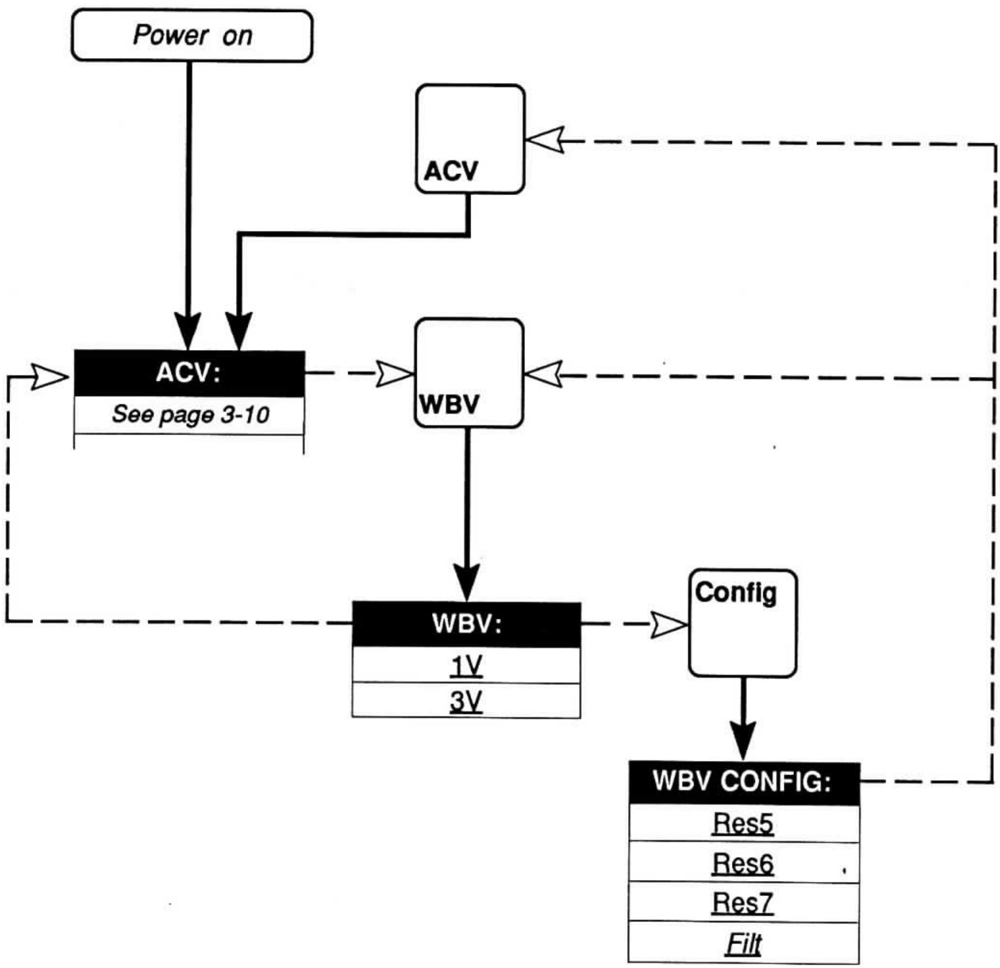
Res6: Accommodates 10ppm resolution for all readings on the range.

Res7: Accommodates 1ppm resolution for all readings on the range.

At Power On, Res7 is active.

*Filt* When underlined, introduces an analog low-pass filter to extend the frequency band down to 10Hz. A second press removes the filter.

# WBV (Wide Band Voltage) - Movement between Menus



# 'Status' Key

So far in this section, we have concentrated on the menus of the keys which select the type of physical quantity to be measured - we call them the Main Function keys. With these, we can configure the functions so that basic measurements conform to our requirements. Obviously the instrument is capable of more sophisticated operation than just taking straightforward measurements.

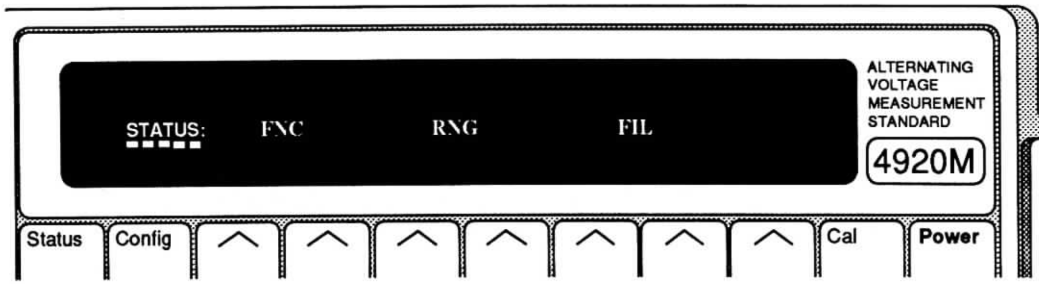
These are discussed in subsequent sections, but the Status key is relevant to basic measurements.

Using the Status key, we can review the instrument parameters which are currently set up, over and above those indicated by the annunciators on the main display.

In addition, the IEEE 488 bus address can be displayed and changed if required.

# Instrument Status Reporting

☞ Press the Status key to see the STATUS report:




Status is a complete report of the most recent selections made using any of the various menus. It can be used at any time as a fast means of checking that the DMM selections are suitable for the measurement being made.

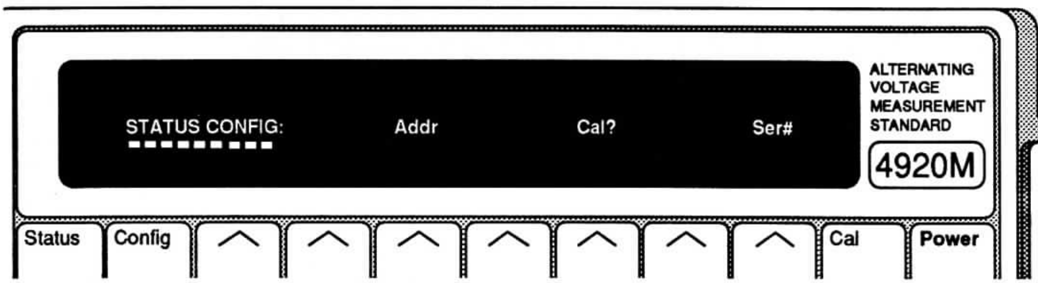
The legends shown in the above diagram do **not** actually appear, they only mark the approximate positions for legends which can appear. Each is an abbreviation which merely acts as a key to the list below. The meaning and possible parameters which appear in each position are given in the list:

| Abbr. | Meaning | Possible Parameters |
|---|---|---|
| **FNC** | Function | ACV, WBV. |
| **RNG** | Range (ACV) | Auto; 0.3V, 1V, 3V, 10V, 30V, 100V, 300V, 1kV. |
| | Range (BWV) | 1V, 3V. |
| **FIL** | Filter | 100Hz, 10Hz. |

## Status Configuration

*(IEEE 488 Bus Address, Power Line Frequency, and Serial Number/Software Issue)*

☞ Press the Config key to see the STATUS CONFIG menu:




This is a menu, defining the following *menu* keys.

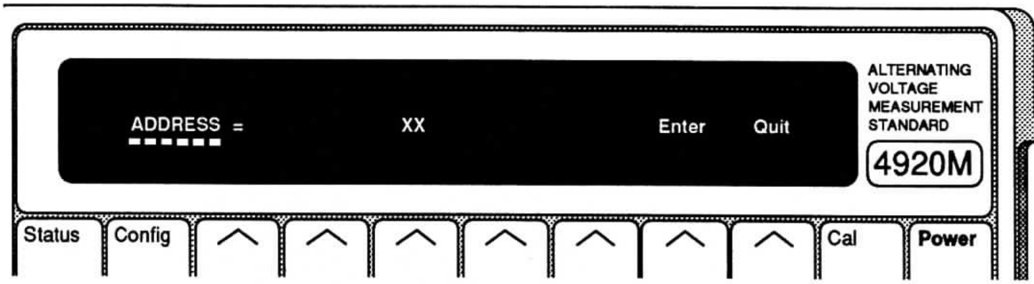
Addr: displays the ADDRESS menu, to review and change the IEEE-488 bus address of the instrument.

Cal?: displays the LAST CAL menu, to review the date of the most recent calibration of the instrument and the due date of the next.

Ser#: displays the SER# menu, to review the serial number and software issue of the instrument.

## IEEE 488 ADDRESS

Press the Addr key to see the IEEE 488 ADDRESS:




This menu permits entry of a value to be used as an IEEE-488 bus address.

Initially, the menu displays the present address value, and the numeric-keyboard keys are activated. Any valid numeric value (0-30) may be entered.

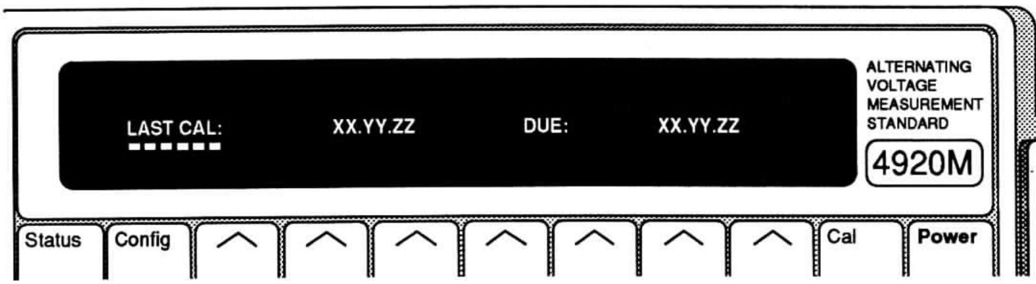
Pressing Enter stores the new value (or restores the old value if unchanged), but pressing Quit leaves the old value intact.

Either Enter or Quit causes exit back to the STATUS CONFIG menu.

Transfer from the ADDRESS menu back to the STATUS CONFIG menu by pressing the Config key.

## Calibration Dates

Press the Cal? key to see the LAST CAL and DUE displays:



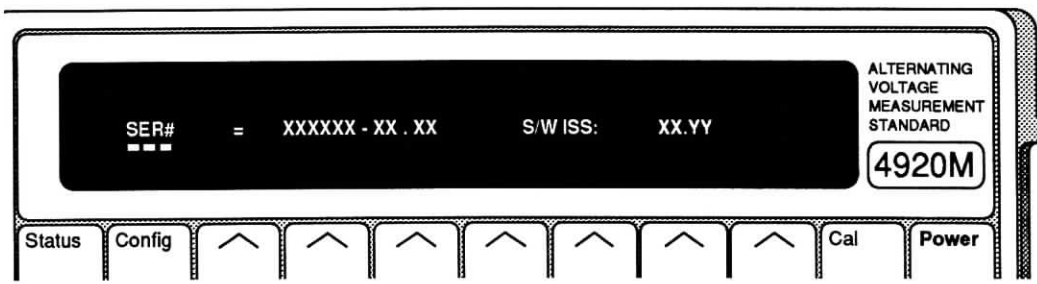

The meanings of these displays are self-evident. Leading zeros are used, and the dates are presented numerically as follows: XX = month; YY = day; ZZ = year.

The dates cannot be altered except in one of the calibration menus. The dates are not lost when the instrument power is turned off.

Transfer to the STATUS CONFIG menu by pressing the Config key.

## Serial Number and Software Issue

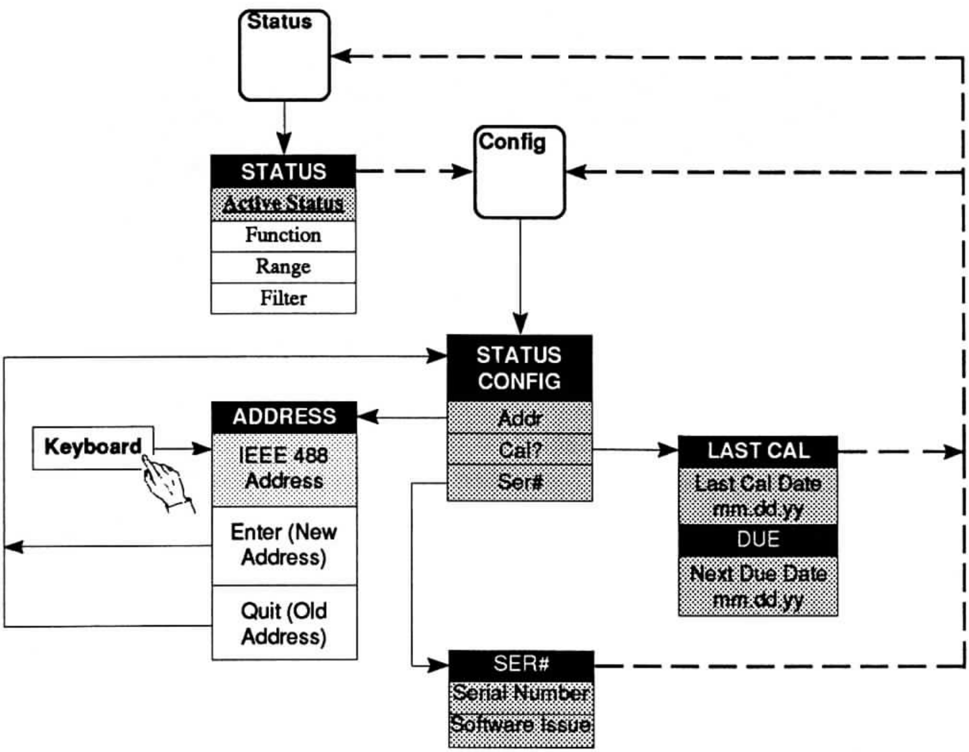
☞ Press the SER# key to see the SER# and S/W ISS displays:




Inspect the instrument serial number and software issue number.

This display is for information only. The serial number cannot be altered except in one of the calibration menus, and this facility is only provided for use during manufacture. Once changed, the number is not lost when the instrument power is turned off. The software issue number is embedded in the software itself, and is not user-alterable.

## Status Reporting - Movement between Menus



## Conclusion

We have now come to the end of our introductory tour of the main menu keys. This is, however, far from the end of the instrument's facilities. Now you are more conversant with the operation of the front panel, it is not necessary to continue in the same sort of programmed way.

You will find that the information in Section 4 is presented in a more concise and accessible form than here in Section 3. Your familiarity with the instrument will allow you to progress rapidly to the facilities you wish to investigate.

Section 4 deals with the manual selection of the facilities not covered here;
Section 5 is devoted to the operation of the instrument via the IEEE 488 Interface.

# SECTION 4 Using the 4920M

## Preliminaries

This section details the methods of using the instrument, divided so as to provide an easy reference for particular functions and facilities. The divisions are as follows:

| Functions | Facilities |
|---|---|
| AC Voltage, | Status Reporting, |
| WB Voltage. | Monitoring, |
| | Math (average), |
| | Test, |
| | Calibration. |

The descriptions include: methods of connection, input limits, types of configurations, methods of access to facilities, and calculations available.

Where appropriate, examples of procedures are given in a format similar to that used in Section 3. Although the menus for calibration are shown, all routine calibration should be referred to Section 8.

## Installation

Before using the instrument, it is important that it has been correctly installed as detailed in Section 2.

## Limiting Characteristics

Maximum inputs are detailed in Section 6.

## Safety

The 4920M meets the safety requirements of UL 1244, ANSI C39.5 (Draft 5) and BSI 4743. Protection is provided by a direct connection via the power cable from ground to exposed metal parts and internal ground screens. The power cable line connection must only be inserted in a socket outlet provided with a protective ground contact, and continuity of the ground conductor must be assured between the socket and the instrument.

**WARNING:**

**Any interruption of the protective ground conductor inside or outside the instrument, or disconnection of the protective ground terminal may make the apparatus dangerous. Intentional interruption is prohibited. The terminals marked with the ⚡ symbol carry the input to the 4920M. These terminals and any other connections to the source under test could carry lethal voltages. Under no circumstance should users touch any of the front or rear panel terminals unless they are first satisfied that no dangerous voltage is present.**

**CAUTION:**

**The ⚠ symbol is used to remind users of special precautions detailed in this handbook, and is placed next to terminals that are sensitive to overvoltage conditions.**

# Interconnections - General Guidelines

### Importance of Correct Connections

When calibrated, the 4920M is capable of giving highly accurate traceable measurements. To attain this, it is necessary to make connection to any external circuitry or load, correctly. A few general guidelines for correct external connection are given in the following paragraphs.

## Sources of Error

### Thermal EMFs

These can give rise to series (Normal) mode interference, particularly where large currents have a heating effect at junctions. In otherwise thermoelectrically-balanced measuring circuits, cooling caused by draughts can upset the balance.

### E-M Interference

Noisy or intense electric, magnetic and electromagnetic effects in the vicinity can disturb the measurement circuit. Some typical sources are:

- Proximity of large static electric fields.
- Fluorescent lighting.
- Inadequate screening, filtering or grounding of power lines.
- Transients from local switching.
- Induction and radiation fields of local E-M transmitters.
- Excessive common mode voltages between source and load.

The disturbances can be magnified by the user's hand capacitance. Separation of leads and creation of loops in the circuit can intensify the disturbances.

### Lead Resistance

The resistance of the connecting leads can drop significant voltages between the source and load, especially at high load currents.

### Lead Insulation Leakage

This can cause significant errors in measurement circuits at high voltages. Some insulating materials suffer greater losses than others, e.g. PVC has more leakage than PTFE.

## Avoidance Tactics

### Thermal EMFs:

Thermal EMFs are not a great source of error in AC measurements, but it is as well to remember the countermeasures:

- Screen thermal junctions from draughts.
- Allow time for thermal equilibrium to be reached before taking readings.
- Use conductors, joints and terminals with a good margin of current-carrying capacity.
- Avoid thermoelectric junctions where possible:
  - Use untinned single-strand copper wire of high purity.
  - Avoid making connections through Nickel, Tin, Brass and Aluminium. If oxidation is a problem use gold-plated copper terminals, and replace the terminals before the plating wears off.
  - If joints must be soldered, low-thermal solders are available, but crimped joints are preferred.
  - Use low-thermal switches and relays where they form part of the measuring circuit.
  - Balance one thermal EMF against another in opposition, where possible. (Switch and relay contacts, terminals etc.)

### E-M Interference:

- Choose as "quiet" a site as possible (a screened cage may be necessary if interference is heavy). Suppress as many sources as possible.
- Always keep interconnecting leads as short as possible, especially unscreened lengths.
- Run leads together as twisted pairs in a common screen (or use coax) to reduce loop pick-up area, but beware of leakage problems and excessive capacitance.
- Where both source and load are floating, connect Lo to ground at the source to reduce common mode voltages.

### Lead Resistance:

- Keep all leads as short as possible.
- Use conductors with a good margin of current-carrying capacity.
- Use Remote Guard or 4-wire connections where necessary.

### Lead Insulation Leakage:

Choose low loss insulated leads - PTFE is preferred to PVC. When running leads together in screened pairs, avoid large voltages between leads in the same screen, especially if using PVC insulation.

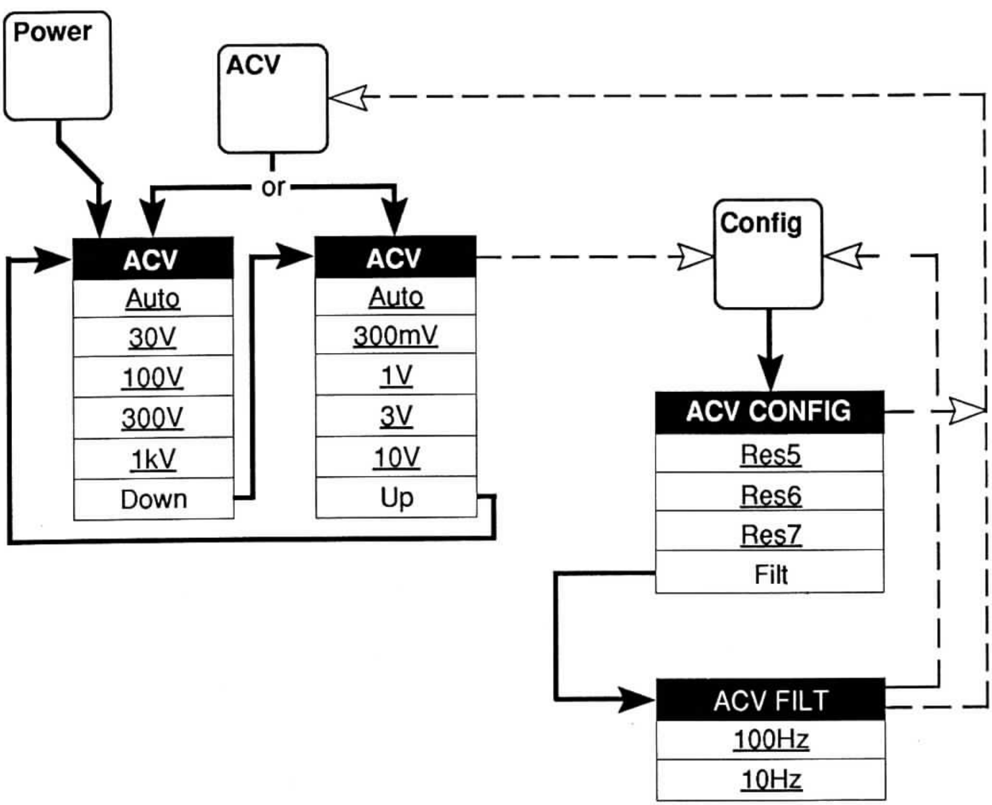
# Functions

## Measurement of AC Voltage

### Generalized Procedure

#### ACV Key and Menus

A description of the User Interface is given in Section 3 for the main functions. If you are unfamiliar with the front panel controls, you should complete the quick tour which starts on Page 3-1. Specific reference to AC Voltage measurement appears on Pages 3-7 to 10. If you are familiar with the controls, but need a reminder of the way a particular facility can be selected; movement among the ACV group of menus is described by the following diagram:



## ACV Input Connections

The ACV input is switched to the High Accuracy AC pre-amp by a relay which isolates the ACV Hi terminal (center contact of the front-panel ACV coaxial connector) whenever the ACV function is deselected.

**Note:** When the ACV function is deselected, the ACV Lo terminal (outer contact of the ACV coaxial connector) remains connected to instrument ground.

## Setup Sequence

The following sequence of operations is arranged so as to configure an AC voltage measurement rapidly from the power on default state. In general, it is quicker to use toggle or choice soft keys on one menu before selecting another menu key.

Obviously, once the instrument has been set up to one configuration, that is the starting point.

- After power on, the power-on default function is ACV, and the default range state (1kV) is shown automatically on the ACV menu.


- Choose a range or Auto, as required, or press the Down key for a lower range.


- Choose a range or Auto, as required, or press the Up key for a higher range.

### Config Key

#### To Alter the Main Display Resolution, or Filter Frequency:

- Press the Config key.


#### Main Display Resolution:

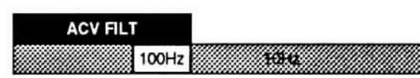
If it has not already been changed, the Power-on default of 7 digit resolution (Res7) is selected.

- Choose from 5, 6 or 7 digits resolution by pressing the appropriate Res choice key.

  Any Main display reading alters to the new resolution when the next measurement result is displayed.

#### Integration Filter Frequency:

- Press the Filt menu key.




- The ACV FILT Menu is displayed.

If the state has not already been changed, the Power-on selection (100Hz) will be indicated.

- Choose either the 100Hz or 10Hz integration filter by pressing its key.

  A change to the10Hz filter will light the FILT annunciator on the Main display, and the read-rate will slow perceptibly.

Exit from the ACV FILT menu by pressing any hard menu key.

# Functions (Contd.)

## Measurement of WB (Wide Band) Voltage

### WBV Input Connection

#### Lead Impedance and Length

For alternating voltage measurement, particularly at the higher WBV frequencies, the impedance and length of the connecting coax can give rise to standing waves and significant loading effects. In some cases the connecting cable can degrade the signal to a greater extent than the full specification tolerance of the 4920M. It is important to ensure that the specified voltage to be measured is as delivered at the electrical (4920M) end of the actual connecting cable to be used (including the precision-N termination), and that the precision-N connector is screwed fully home into the WBV socket.

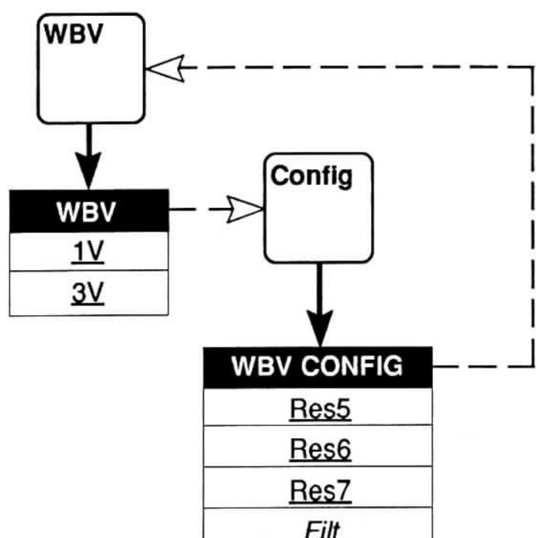
#### Substitute Termination

When WBV is deselected, and during transfers when the thermal converter is not connected to the input, the WBV socket circuit is switched to provide an identical 50Ω termination.

### Generalized Procedure

#### WBV Key and Menus

A description of the User Interface is given in Section 3 for the main functions. If you are unfamiliar with the front panel controls, you should complete the quick tour which starts on Page 3-1. Specific reference to WB Voltage measurement appears on Pages 3-11 to 3-13. If you are familiar with the controls, but need a reminder of the way a particular facility can be selected; movement among the WBV group of menus is described by the following diagram:



## Setup Sequence

The following sequence of operations is arranged so as to configure a WBV measurement rapidly from the power on default state. In general, it is quicker to use toggle or choice soft keys on one menu before selecting another menu key.

Obviously, once the instrument has been set up to one configuration, that is the starting point.

- After power on, the power-on default function is ACV, and the default range state (1kV) is shown automatically on the ACV menu.


- Press the WBV key.
- The power-on default WBV range is 3V, shown automatically on the WBV menu.


- Choose the 3V or 1V range, as required.

**Config Key**

### To Alter the Main Display Resolution, or introduce the WBV Filter:

- Press the Config key.


### Main Display Resolution:

If it has not already been changed, the Power-on default of 7 digit resolution (Res7) is selected. The WBV filter is not selected.

- Choose from 5, 6 or 7 digits resolution by pressing the appropriate Res choice key.

  Any Main display reading alters to the new resolution when the next measurement result is displayed.

### WBV Filter:

- To select 'Filter In', press the *Filt* key.


- The 'Filter In' selection will be indicated.

  Introducing the filter will light the FILT annunciator on the Main display, and the read-rate will be reduced perceptibly.

  Pressing the FILT key when it is underlined will select 'Filter Out', and extinguish the Main display FILT annunciator light.

Exit from the WBV CONFIG menu is by pressing any hard menu key.

# Facilities

## Status Reporting Facilities

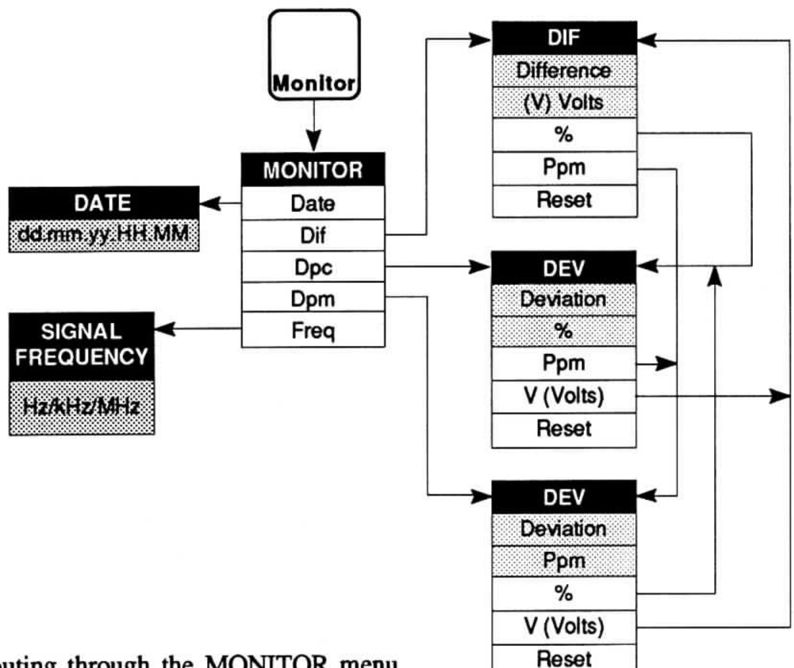
This subject is fully described in Section 3, pages 3-14 to 3-20.

## Monitoring Facilities

### Monitor Menus

The User Interface is described in Section 3. If you are unfamiliar with the front panel controls, complete the quick tour which starts on Page 3-5.

To give an overall view of the monitoring facilities, movement among the MONITOR group of menus is described by the following diagram:

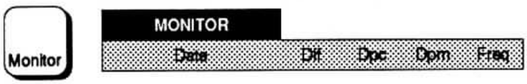



**Note:**

To avoid routing through the MONITOR menu when changing from one form of calculation to another, access to the two other DIF and DEV results is provided from within each of the three menus. In the diagram, these 'internal' access routes are drawn on the right of the menu blocks.

## Monitor Key

Pressing the Monitor front panel key causes the MONITOR menu to be displayed:




This menu defines five *menu* keys:

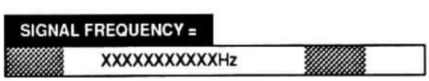
**Date** opens the **DATE** menu:
The DATE menu presents a readout of the present date in the format:

mm.dd.yy.HH.MM

where the pairs of digits have the meanings:

mm = month
dd = day
yy = year
hh = hour
MM = minute

**Freq** displays the **SIGNAL FREQUENCY**
This gives the frequency corresponding to the RMS value shown on the main display.




There are no selections to be made, exit is by pressing a hard key.

### Dif, Dpc and Dpm Keys

These keys enable a particular reading to be established as a 'reference reading', and then display the difference or deviation of subsequent readings from this reference. After performing the calculations, the instrument enters the result on the Menu Display.

**Dif**
The Dif key opens the DIF menu. Pressing the Reset soft key sets the most-recent reading as the reference. The instrument calculates the difference between each subsequent reading and the reference, showing the result on the DIF menu display.

To set a new reference reading, simply press the Reset soft key, and the most-recent reading becomes the new reference.

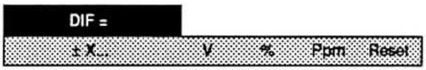
**Dpc**
The Dpc key opens the percentage DEV menu. Operation is similar to that for the DIF menu, except that the calculation is for percentage deviation.

**Dpm**
The Dpm key opens the parts-per-million DEV menu. Operation is similar to that for the %DEV menu, except that the calculation is for ppm deviation.

**Menu Transfer**
In each of the three menus, soft keys permit direct transfer from one to the other without recourse to the main MONITOR menu.

**Menu Details**
These are given overleaf.

**Dif** opens the **DIF** menu:
After each reading, the DIF menu updates the difference between the RMS value of the most-recent reading and that of a reference reading (refer to '**Reset**' below).




DIF = + X... : Larger than reference.

DIF = - X... : Smaller than reference.

X... : Most-Recent Reading - Reference Reading

Normal notation with exponent.

'NOT VALID' means that *either:* no reference is set; *or:* the input voltage is outside the bounds of the selected range.

'**Reset**': Stores the most-recent reading as reference. The 'Difference' area remains blank until the next reading is available for comparison.

'**%**': Transfers to the DEV % menu.

'**Ppm**': Transfers to the DEV ppm menu.

**Dpc** opens the **DEV** % menu:
After each reading, the DEV menu updates the percentage deviation of the RMS value of the most-recent reading from that of a reference reading (refer to '**Deviation**' and '**Reset**' below).


DEV = + X... : Larger than reference.

DEV = - X... : Smaller than reference.

X... : $\frac{100 \text{ x } (\text{Most-Recent Rdg - Reference Rdg})}{\text{Reference Rdg}}$

Normal notation with exponent.

'NOT VALID' means that *either*: no reference is set; *or*: the input voltage is outside the bounds of the selected range.

'**Reset**': Stores the most-recent reading as reference. The 'Deviation' area remains blank until the next reading is available for comparison.

'**V**': Transfers to the DIF menu.

'**Ppm**': Transfers to the DEV ppm menu.

**Dpm** opens the **DEV** ppm menu:
After each reading, the DEV menu updates the ppm deviation of the RMS value of the most-recent reading from that of a reference reading (refer to '**Deviation**' and '**Reset**' below).


DEV = + X... : Larger than reference.

DEV = - X... : Smaller than reference.

X... : $\frac{10^6 \times (\text{Most-Recent Rdg - Reference Rdg})}{\text{Reference Rdg}}$

Normal notation with exponent.

'NOT VALID' means that *either*: no reference is set; *or*: the input voltage is outside the bounds of the selected range.

'**Reset**': Stores the most-recent reading as reference. The 'Deviation' area remains blank until the next reading is available for comparison.

'%': Transfers to the DEV % menu.

'V': Transfers to the DIF menu.

**Reading Consistency**

Once a reference reading is established, it is not destroyed by changing between functions or ranges. It is memorized in a standard format so that it is directly comparable with any subsequent reading, regardless of range or function.

**Transfer Standard**

By connecting a Standard AC Voltage Source to one of the inputs, and using the external trigger facility to sample its value, it can be set as reference. The difference and deviation of other sources connected to either input can then be measured with respect to the standard source.

# Direct Action Keys

These seven keys are located beneath the main display. They allow the operator to act as follows:

## Reset

Provides a quick means of resetting the instrument to the power-up state, when in local operation.

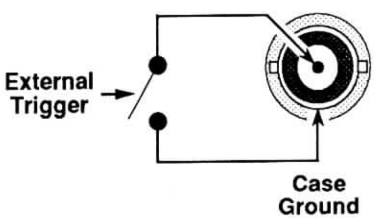
The instrument default states for Power On are given in Appendix B to Section 5. Pressing **Reset** provides the same result, except that any settings directly concerned with remote operation are not altered.

## Ext'trig

Disables internal triggers, and enables the external trigger sources: 'Sample' and SK9. The 'Ext' annunciator on the main display is lit.

Ext'trig can be self-cancelled by a second press, to enable internal triggers. The Ext annunciator is turned off when internal triggers are enabled.

### External Trigger Socket SK9 (Rear Panel)




## Sample Key

Triggers a single-shot measurement if the DMM is in Ext'trig mode. All 'Sample' measurements are subject to the standard acquisition times. These are listed in Appendix B at the end of Section 4.

During the measurement the 'Busy' annunciator on the main display is lit.

## Local

Returns the DMM to front panel control when operating on the IEEE-488 bus, provided that it is not disabled by remote command. It will cause the Rem annunciator on the main display to turn off.

Local can be disabled by a controller using the LLO (Local Lockout) function.

## SRQ

If set to remote in IEEE 488 system operation, with 'URQ' and 'ESB' bits enabled; this key generates a Service Request (SRQ) on the IEEE 488 bus. It causes the SRQ annunciator on the main display to light, and remain lit until the request is serviced.

SRQ can be disabled via the IEEE 488 bus using the 'Event Status Enable' or 'Service Request Enable' register commands.

For further information refer to Section 5.

## Caltrig

This key is only active when the Cal annunciator is lit in the main display. It is used to trigger all operations selected in the calibration menus, except those concerned with entry of numeric data (in which case the menu contains an 'enter command).

## Zero

As the 4920M measures only alternating voltage, the Zero key has no function on this instrument.

Its action is disabled in firmware and so it does not respond.

# 'Numeric Keyboard' keys

## Keyboard Facility

### Numeric Function

Seventeen of the menu keys double as numeric keyboard keys when certain menus appear on the dot-matix display, and in most cases all other keys are locked out. As well as the numbers 0 to 9, the decimal point and the polarity changeover (+/-) keys, five other functions are represented.

### Exp

The number appearing on the numeric display to the right of 'E' is a power of ten, by which the number to the left of the E is multiplied. The Exp key is used to enter E into the expression.

Exp can be preceded by a minus sign to indicate fractional powers of ten: to do this press the +/- key before pressing Exp.

### Enter

After assembling the number within a menu, the Enter key is pressed to confirm that it is to be used. Usually the word Enter also appears in the menu. In some cases the Enter command enables another key, or presents another menu.

### Quit

For a few menus (associated with 'Math' and 'Cal') the Quit key is provided for convenient exit, without activating any process.

### ← ('Monitor' key)

Deletes the previous numerical character.

### Last rdg

When a reading from the main display is required to be incorporated into a process, the Last rdg key can be used to enter the value of the most-recent measurement on to the dot-matrix menu.

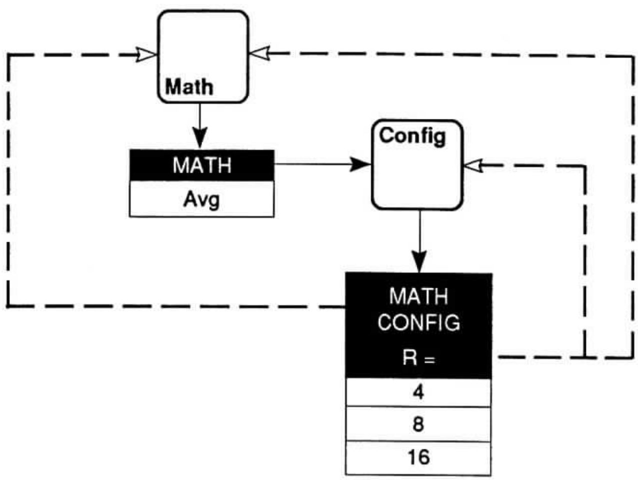
# Mathematical Facility

### Math Menus

A description of the User Interface is given in Section 3 for the main functions.

If you are unfamiliar with the front panel controls, you should complete the quick tour which starts on Page 3-5.

To give an overall view, movement among the MATH group of menus is described by the following diagram:



## Math Key

The Math front panel key causes the MATH menu to be displayed. This menu activates a rolling average mode, whose window length is entered via the MATH CONFIG menu.

## MATH Menu


This menu defines one *toggle* key, which is not selected at Power On. The constant 'R' is defined via the MATH CONFIG menu.

Avg — Causes a rolling average of **R** readings to be made.

Activation of rolling average causes the Math annunciator on the main display to be lit.

**'Rolling Average' Process**

The average is not displayed until R readings have been taken.

As each subsequent reading becomes available, it is added to the others in the window, the earliest of which is discarded. Then the mean of the readings in the window is calculated and displayed.

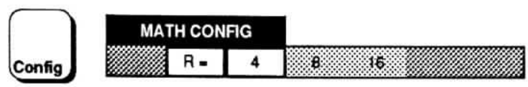
Thus once the 'Rolling Average is displayed, it is updated after each reading.

## CONFIG Key

Selection of the Config key in MATH will cause the MATH CONFIG menu to be displayed.

## MATH CONFIG Menu

This menu allows the user to select the number of readings for the 'moving window' used in rolling average. The most-recently selected value is underlined with a cursor.




This menu defines three *choice* keys:

4 — Selects a rolling average of 4 readings. 4 is selected On at Power On.

8 — Selects a rolling average of 8 readings. 8 is not selected at Power On.

16 — Selects a rolling average of 16 readings. 16 is not selected at Power On.

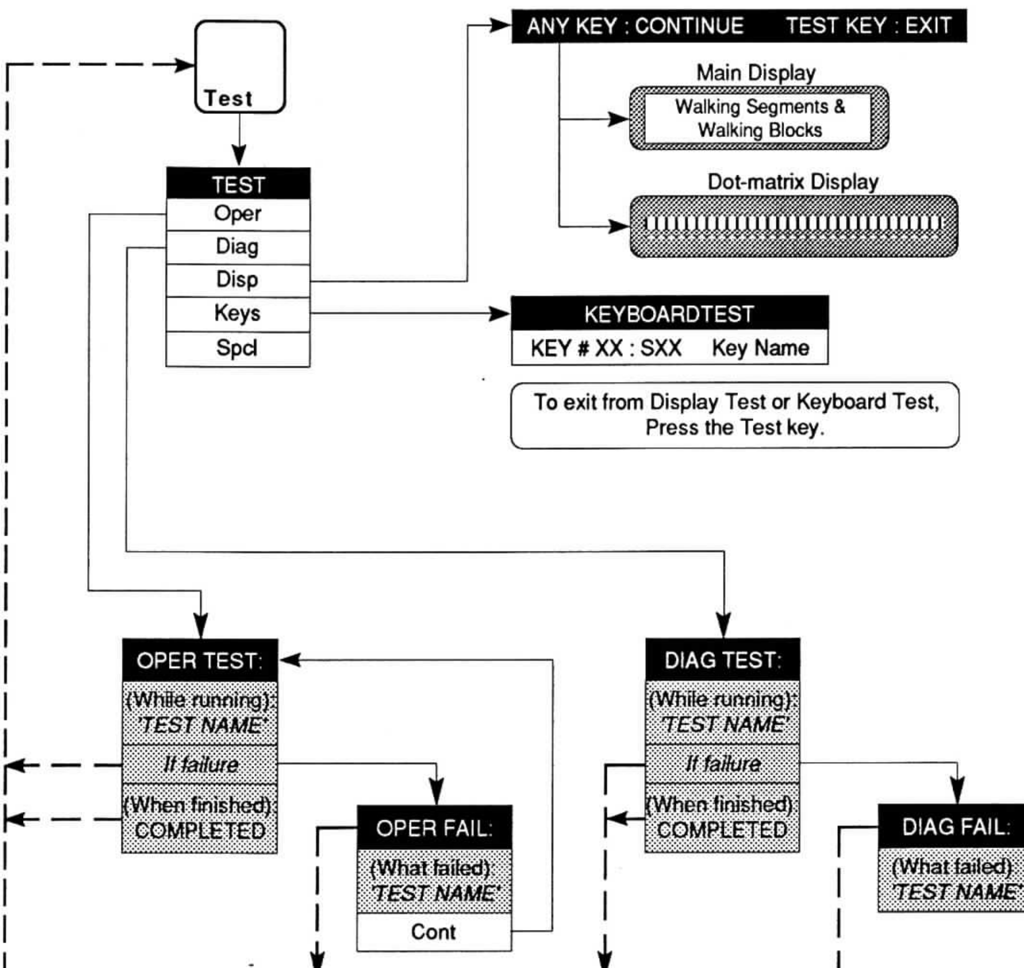
# Test Facilities

### Test Menus

A description of the User Interface is given in Section 3 for the main functions.

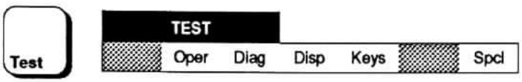
If you are unfamiliar with the front panel controls, you should complete the quick tour which starts on Page 3-5.

To give an overall view, movement among the TEST group of menus is described by the following diagram:



## Test Key

The front panel Test key causes the TEST menu to be displayed. Different types of selftest can be chosen from this menu.




### Caution

The success of the Operational and Diagnostic Tests can be inhibited by:

- temperature not in the range: 13°C to 33°C;
- more than 1 year since the most-recent external calibration; or
- presence of excessive RFI or power-line noise.

### Oper Key

Oper transfers to the OPER TEST menu, starts an operational test, disabling all menu and direct action keys, signal inputs and normal trigger sources. This test includes a calibration memory check.

### OPER TEST Menu


While operational test is running, the display shows the name of the test currently being performed. Once a failure is noted, the test is halted and transfers to the OPER FAIL menu, showing the test which failed and a key to permit a user to continue the operational test.

### OPER FAIL Menu

When an operational test has failed, the test sequence is stopped, and the OPER FAIL menu is opened.


The name of the failed test appears, and a soft key (Cont) is allocated to allow a user to continue the test after noting the failure.

### Disp Key

A reminder menu appears first, noting the actions of the keys.


Repeatedly pressing any key **other** than Test increments both displays through a sequence of 'walking strobes', which allow a user to inspect individual segments and complete blocks.

### Keys Key

The Keys soft key presents the KEYBOARD TEST menu.


All keys **other** than the Test key can be tested by pressing. For each key pressed, the 'Key #' is followed by the key's hexadecimal matrix positon, then after a colon an 'S' is followed by the key's switch ident number. The name of the key is given on the right of the display.

### Exit

During 'Disp' or 'Keys' checks, pressing the Test key **terminates the sequence.**

**Diag Key**

Diag transfers to the DIAG TEST menu, starts an diagnostic test, disabling all menu and direct action keys, signal inputs and normal trigger sources. This test includes a calibration memory check.

**DIAG TEST Menu**


While diagnostic test is running, the display shows the name of the test currently being performed. Once a failure is noted, the test is halted and transfers to the DIAG FAIL menu, showing the test which failed.

**DIAG FAIL Menu**

When an diagnostic test has failed, the test sequence is stopped, and the DIAG FAIL menu is opened.


The name of the failed test appears. No means of continuing the test is provided, as the failure could affect subsequent tests.

**Exit**

To exit from Diagnostic Test after a test failure, press any major function key. Pressing the Test key then choosing Diag again will lead to the same failure condition and report.

# Calibration Operations

## Caution

**The descriptions in the following pages are intended only as a guide to the operations available to calibrate the instrument. They contain neither examples nor calibration routines, and should NOT be used directly as a basis for calibrating any part of the instrument. Some of the commands, if used unwisely, will obliterate an expensive calibration or recalibration.**

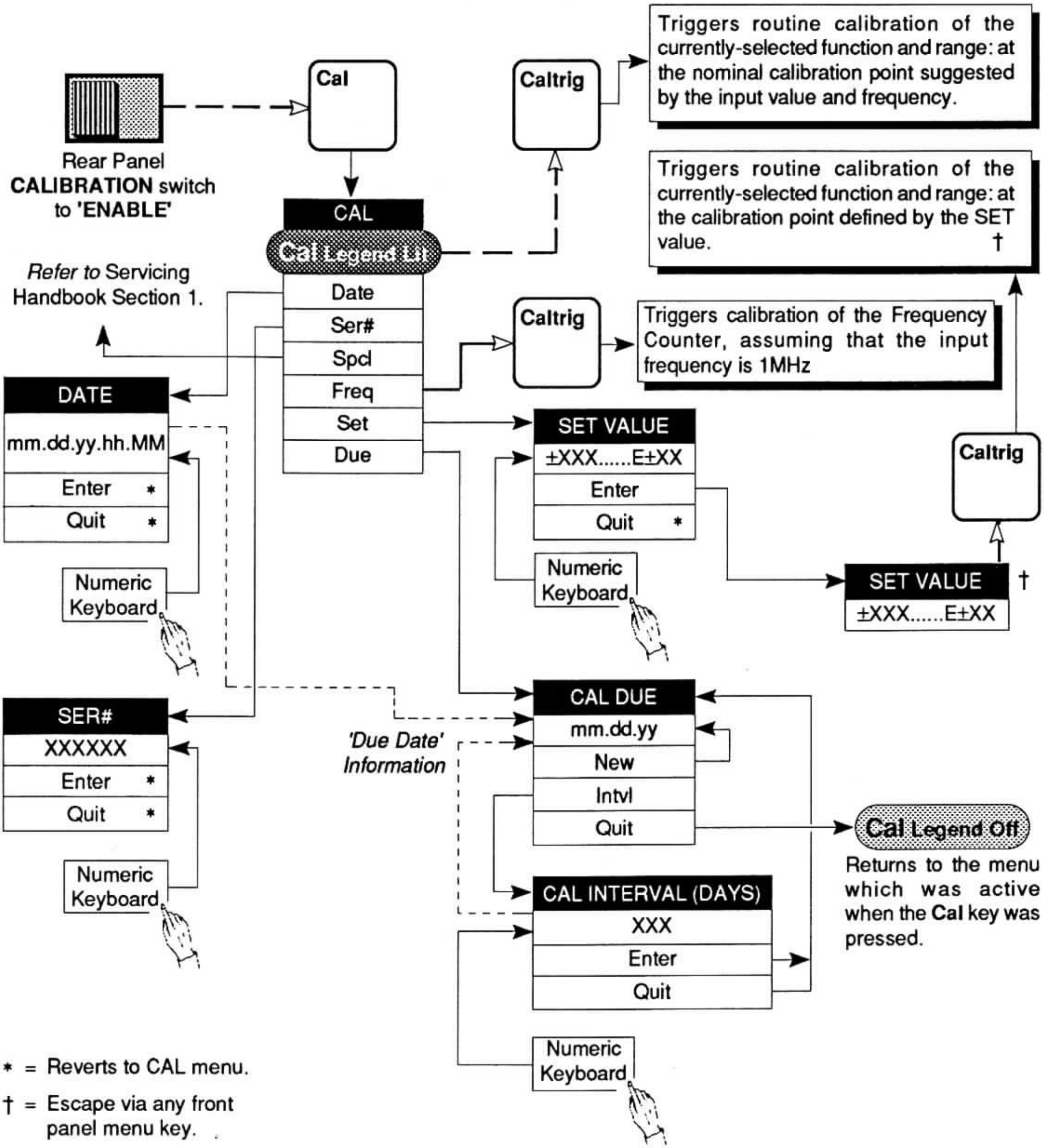
**For a guide to calibration routines refer to Section 8.**

### General Outline of Calibration Operations

The calibration process generally conforms to a set sequence of operation:

1. The rear-panel switch must be set to ENABLE, then calibration is enabled by pressing the Cal key (this may need further parameters to be specified). An optional parameter is available for use when the calibration is to be performed at a non-nominal value.
2. With the appropriate analog input applied, the calibration operation is triggered. The relevant corrections are calculated and stored in non-volatile memory. Subsequently, in normal use; gain and flatness calibrations are applied to correct the pre-selected function and range. Filter and linearity calibrations are carried out on one range and applied to correct all ranges of the pre-selected function.
3. Other operations can be carried out, such as setting the calendar/clock or the calibration interval.
4. When calibration is complete, calibration is finally disabled by setting the rear panel switch to DISABLE.

## CAL Group of Menus - Overall View




* = Reverts to CAL menu.

† = Escape via any front panel menu key.

# Calibration Menus

## Front Panel Cal Key

The Cal key on the front panel causes the CAL menu to be displayed in the dot matrix display, so long as the instrument is not already in Cal mode. This menu provides access to the calibration menus, also permitting some other non-volatile memory data to be accessed and changed.

## CAL Group of Menus

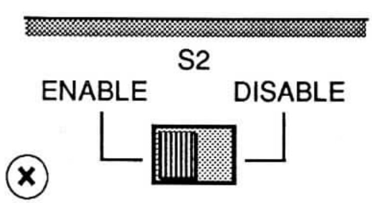
A description of the User Interface is given in Section 3 for the main functions.

If you are unfamiliar with the front panel controls, you should complete the quick tour which starts on Page 3-5.

To give an overall view, movement among the CAL group of menus is described by the diagram on the opposite page.

## Access Conditions

Rear panel switch S2 provides access to the CAL menu, and to the non-volatile calibration memory. S2 must be set to 'ENABLE' for calibration.

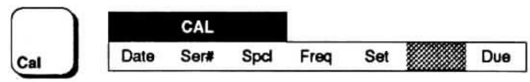



CALIBRATION

S2 is recessed to avoid inadvertent operation. A paper seal can be placed over the switch to protect calibration.

## CAL Menu

Pressing the Cal key opens the CAL menu. The main group of calibration menus is available.




**Caution:**

In this and other calibration menus the **Caltrig** key is enabled, and when pressed alters the calibration memory. To reduce the possibility of inadvertently obliterating the previous calibration, the menu should only be used during a genuine recalibration. Refer to Section 8.

Once the 'Cal' legend is lit, the major function hard keys can be selected and the various ranges calibrated at lower and upper cardinal points, using the **Caltrig** direct action key.

If the values are not exactly at the cardinal points, then Set in the CAL menu can be used to inform the instrument of the exact value.

This menu defines five *menu* keys, all of which are *not selected* at Power On. They are described overleaf.

## Calibration Menus (Contd.)

**Spcl** This key displays the SPCL menu, which accesses five 'special' calibration operations. They can be regarded as 'presets'; being intended for use at manufacture and at times when the basic instrument specification is to be redefined. **They are not to be used for routine calibration purposes.**

Quit transfers back to the CAL menu.

**Freq** Selects Frequency Calibration. With an accurate 1MHz signal applied on the 0.3V, 1V, 3V or 10V range, the action of pressing the Caltrig key performs calibration of the internal Frequency Counter.

**Date** Opens the DATE menu. This shows the present date in the format:

mm.dd.yy.HH.MM

where the pairs of digits have the meanings:

mm = month
dd = day
yy = year
hh = hour
MM = minute

| DATE | | | |
|---|---|---|---|
| mm.dd.yy.hh.MM | | Enter | Quit |

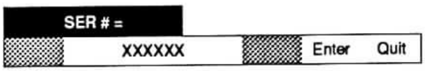
The date and time can be changed in this menu, by operating the numeric keyboard keys. Pressing any numeric key will cancel the whole date, permitting the present date to be written, in the same format.

Enter causes the internal calendar/clock to be reset to the date/time just written.

Quit aborts the attempt to reset the calendar/clock, which continues uninterrupted running.

Both Enter and Quit transfer back to the CAL menu.

**Ser#** Opens the SER# = menu. This shows the assigned instrument serial number, and the numeric keyboard is activated. A numeric value can be entered.

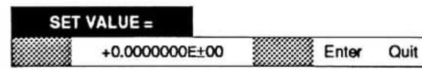



**Enter** writes the new serial number into non-volatile RAM, overwriting the old number.

**Quit** aborts the attempt to reset the serial number. The existing serial number remains stored in non-volatile memory.

Both Enter and Quit transfer back to the CAL menu.

**Set** Opens the SET VALUE = menu. This shows a value of zero, and the numeric keyboard is activated.



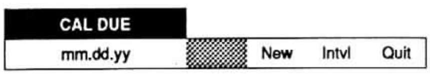

A particular cal source value can be entered as target value for non-nominal calibration.

**Enter** writes the new target value into non-volatile RAM, and generates a new SET VALUE = display. Pressing the Caltrig key performs the 'SET' calibration.

**Quit** aborts the attempt to reset the target value and transfers back to the CAL menu. The newly-written value is destroyed.

**Due** This is intended to be used following a calibration, in order to note the date when the next calibration is due. The new date is calculated from the current date registered by the internal calendar/clock, and calibration-interval information entered via the CAL DUE and CAL INTERVAL menus.

The Due key opens the CAL DUE menu.

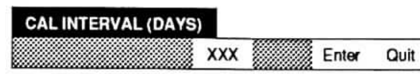



This shows the old due date for the next calibration, calculated at the previous calibration.

**New** causes the instrument to calculate the date when the next calibration is due, from the date currently registered by the internal calendar/clock, and calibration-interval information entered via the CAL INTERVAL menu.

The new 'Cal Due' date is presented on the DUE DATE menu instead of the old due date.

**Intvl** Opens the CAL INTERVAL (DAYS) = menu. this displays the currently-registered calibration interval, and the numeric keyboard is activated. A new interval can be written using the numeric keyboard.




**Enter** causes the new interval to be written into non-volatile memory, in place of the old interval.

Enter transfers back to the CAL DUE menu, where the Cal Due date changes to show the effect of the new interval.

**Quit** aborts the attempt to reset the calibration interval. The old interval remains stored in non-volatile memory.

Quit transfers back to the CAL DUE menu. The Cal Due date does not change.

**CAL DUE Menu - Exit from Calibration mode.**

**Quit** from the CAL DUE menu. The Cal legend on the main display is extinguished. Quit returns either to the menu which was open when the Cal key was pressed; or to a 'neutral' menu, whereupon a main menu key can be pressed to select the next required menu.

*Appendix A to*
*Section 4 of the*
*User's Handbook for*
*Datron Model 4920M*

***Note to users:*** *For the sake of completeness, this appendix collects together the error codes which might be generated either on the instrument front panel, or via the IEEE 488 system bus.*

## Error Detection

All errors which cannot be recovered without the user's knowledge, result in some system action to inform the user via a message, and where possible restore the system to an operational condition. Errors are classified by the method with which they are handled. Recoverable errors report the error and then continue. System errors which cannot be recovered cause the system to halt with a message displayed. Restarting the instrument from Power On may clear the error, but generally such messages are caused by hardware or software faults, which require user action.

## Error Messages

### Fatal System Errors

For all fatal system errors, the error condition is reported only via the front panel. The processor stops after displaying the message. A user must respond by retrying operation from power on, and initiate repair if the fault persists. The following is a list of error numbers displayed, with their associated fault descriptions:

9000 - System Kernel Fault
9001 - Run Time System Error
9002 - Unexpected Exception
9005 - Serial Interface Fault
9099 - Undefined Fatal Error

## Recoverable Errors

These consist of **Command** Errors, **Execution** Errors and **Device-Dependent** Errors. Command Errors can only be generated due to incorrect remote programming. Some Execution Errors and all Device-Dependent Errors can all be generated by manual operation as well. Each of the reportable Execution and Device-Dependent Errors are identified by a code number.

**Command Errors (CME)**
(Remote operation only)
Command Errors are generated when the remote command does not conform, either to the device command syntax, or to the IEEE 488.2 generic syntax. The CME bit (5) is set *true* in the Standard-defined Event Status Byte, but there is no associated queue.

The error is reported by the mechanisms described in the sub-section of Section 5 which deals with status reporting.

Errors generated due to incorrect front panel manipulation are not reported to the bus; and vice versa.

**Execution Errors (EXE)**
An Execution Error is generated if a received command cannot be executed because it is incompatible with the current device state, or because it attempts to command parameters which are out-of-limits.

In remote operation, the EXE bit (4) is set *true* in the Standard-defined Event Status Byte, and the error code number is appended to the Execution Error queue.

The error is reported by the mechanisms described in the sub-section of Section 5 which deals with status reporting, and the queue entries can be read destructively as LIFO by the Common query command *EXQ?.

There is no queue when execution errors are generated during manual operation, the description of the error being presented directly on the Menu display.

The Execution Error numbers are given below, with their associated descriptions.

### List of Execution Errors

1000 - No Execution Error
1001 - No Test in Cal Mode
1002 - Resume test not allowed
1003 - Cal SWITCH disabled
1004 - Cal Mode not enabled
1005 - Set nominal not allowed
1006 - Invalid range/function
1007 - Invalid numeric data

## Recoverable Errors (contd)

**Device-Dependent Errors (DDE)**

A Device-Dependent Error is generated if the device detects an internal operating fault (eg. during self-test). The DDE bit (3) is set *true* in the Standard-defined Event Status Byte, and the error code number is appended to the Device-Dependent Error queue. The code number and description appear on the right-hand display, remaining visible until the next key-press or remote command.

In Remote, the error is reported by the mechanisms described in the sub-section of Section 5 which deals with status reporting, and the queue entries can be read destructively as LIFO by the Common query command *DDQ?.

Note that error codes beginning 2... can be caused by incorrect operation or instrument failure; error codes beginning 3... are almost certainly due to instrument failure.

**Device-Dependent Error Lists**

Device-dependent errors are associated mainly with test and calibration operations. The error numbers in the following pages are therefore listed in these categories. There is some overlap.

The error list for calibration operations, with their associated descriptions, commences overleaf, followed by the selftest error list.

## Error Messages - Calibration Operations

**Calibration Invalidities**

2000 - No Device Error in List
2001 - Invalid Gain Cal
2002 - Invalid HF Trim Cal
2003 - Invalid Flatness Cal
2004 - Invalid Linearity Cal
2005 - Invalid Frequency Cal
2006 - Invalid Calstore Read
2007 - Invalid Calstore Write
2008 - Invalid Cal Arithmetic
2009 - Invalid Serial Data
2010 - Corrupt Datarec Value

**Calibration Illegalities**

3001 - Illegal Cal Exercise
3002 - Illegal Special Cal Step
3003 - Illegal Cal Update
3004 - Illegal Calstore Access
3005 - Illegal Calstore Clear
3006 - Illegal Calstore Func
3007 - Illegal Calstore Range
3008 - Illegal Calstore Length
3009 - Illegal Clock Destinatn
3010 - Illegal Clock Reading
3011 - Illegal Clock Setting
3012 - Illegal Clock Access
3013 - Illegal Measure Phase
3014 - Illegal Scale Index
3015 - Illegal Flatness Value
3016 - Illegal Linearity Value
3017 - Illegal HF Correction
3018 - Illegal Gain Correction
3019 - Illegal WBLIN Parameter
3020 - Illegal WB RMS Value
3021 - Illegal WB DAC Setting
3022 - Illegal WB Autorange Func
3023 - Illegal WB Autorange Range
3024 - Illegal WB Test Stage
3025 - Illegal WB Test Store
3026 - Illegal WB Test Polarity
3027 - Illegal Engine Op-Code
3028 - Chip Test AC Pre-amp
3029 - Chip Test AC RMS Control
3030 - Chip Test A-to-D DAC
3031 - Chip Test A-to-D Switch
3032 - Chip Test A-to-D Tester
3033 - Serial Loop Failure
3034 - Illegal Operation
3035 - Illegal Parameter

## Selftest Operations - Error Messages

Note that errors from 4015 to 4052 may occur if a signal or low impedance is connected to either instrument input when the test is taking place. If this is not the case, then a self-test error implies instrument failure.

| Step | Error Message |
|---|---|
| **Preliminary** | |
| 4000 | Stuck Key. |
| 4001 | Dig Power Supplies |
| 4002 | Battery Voltage |
| **Memory** | |
| 4003 | ROM Bitsums. |
| 4004 | RAM Read/Write. |
| 4005 | CAL Integrity. |
| **Digital Communication** | |
| 4006 | Serial Loop Config. |
| **Analog-to-Digital Conversion** | |
| 4007 | A-to-D Zero. |
| 4008 | A-to-D +15V Supply. |
| 4009 | A-to-D -15V Supply. |
| 4010 | A-to-D Gain. |
| 4011 | A-to-D Linearity +. |
| 4012 | A-to-D Linearity -. |
| 4013 | A-to-D +11V Supply. |
| 4014 | A-to-D -19V Supply. |
| **High Accuracy ACV Conditioning** | |
| 4015 | AC Preamp X1 Zero. |
| 4016 | AC Preamp X3 Zero. |
| 4017 | AC 1V Gain, DC Pos. |
| 4018 | AC 1V Gain, DC Neg. |
| 4019 | AC Preamp X3 Gain. |
| 4020 | AC 10V Attenuator. |
| 4021 | AC 100V Attenuator. |
| 4022 | AC 1kV Attenuator. |
| **Frequency and Overload** | |
| 4023 | AC Freq Counter. |
| 4024 | AC Overload Detect. |
| 4025 | AC Overload Switch. |

| Step | Error Message |
|---|---|
| **High Accuracy RMS Conversion** | |
| 4026 | AC HF Path Detect. |
| 4027 | AC RMS Converter. |
| 4028 | AC RMS Track. |
| 4029 | AC RMS Hold. |
| 4030 | AC Quasi-Sine. |
| 4031 | AC Measurement. |
| 4032 | Not Used. |
| 4033 | AC 10Hz Filter. |
| 4034 | Not Used. |
| 4035 | AC RMS Linearity. |
| **Wide Band D-A Conversion** | |
| 4036 | WB DAC +ve Zero. |
| 4037 | WB DAC -ve Zero. |
| 4038 | WB DAC +ve Full Scl. |
| 4039 | WB DAC -ve Full Scl. |
| 4040 | WB DAC +3.5V. |
| 4041 | WB DAC -3.5V. |
| **Wide Band Chopper Action** | |
| 4042 | WB DC Switch +ve. |
| 4043 | WB DC Switch -ve. |
| **Wide Band Thermal Conversion** | |
| 4044 | WB TTU DC 3.5V. |
| 4045 | WB TTU DC 1.0V. |
| 4046 | WB TTU DC 0V. |
| 4047 | WB TTU AC 0V. |
| 4048 | Not Used. |
| **Wide Band Measurement Paths** | |
| 4049 | A-to-D WB Path. |
| 4050 | WB Input Path. |
| 4051 | WB Input Z, DC Chan. |
| 4052 | WB 100mV Path. |
| 4053 | WB Cal Validity. |
| **Other** | |
| 4054 | Bus Address. |

*Appendix B to*
*Section 4 of the*
*User's Handbook for*
*Datron Model 4920M*

# 4920M Standard Signal-Acquisition Times

The 4920M uses either a thermal converter (WBV 3V Range) or an electronic converter (High Accuracy ACV all Ranges and WBV 1V Range), to measure the RMS value of the input signal.

After the reading trigger, with the conditioned input applied to the converter, an optimum settling delay is programmed to allow the converter output to stabilize before the start of the transfer sequence. Users cannot alter this settling time. It is fixed in firmware for each significant filter and range state.

After settling, the signal must remain present during the first (sampling) phase of the thermal transfer cycle, or for the whole of the electronic transfer cycle.

The total signal acquisition period (which includes both settling and sampling times) is given for Function, Range and Filter in the tables below. The signal must remain connected and activated for the periods shown.

**ACV All Ranges**

| Filter | Acquisition Time |
|---|---|
| 100Hz | 2.3 sec |
| 10Hz | 5.9 secs |

**WBV**

| Range | Filt. | Acquisition Time |
|---|---|---|
| 1V | Out | 2.3 sec |
| | In | 5.9 secs |
| 3V | Out | 12 secs |
| | In | 12 secs |

*Appendix C to Section 4 of the User's Handbook for Datron Model 4920M*

# 4920M ACV and WBV Significant Range Points

(including Range and Resolution selection features)

| ACV Range | ACV *Nrf* selection limits | Range Point | Res5 RESL5 | Res6 RESL6 | Res7 RESL7 |
|---|---|---|---|---|---|
| **1000V** | 300 < *Nrf* | Upper Limit | 1199.50V | 1199.500V | 1199.5000V |
| | | Nominal | 1000.00V | 1000.000V | 1000.0000V |
| | | Auto Down | 299.99V | 299.999V | 299.9999V |
| | | Lower Limit | 90.00V | 90.000V | 90.0000V |
| **300V** | 100.0 < *Nrf* ≤ 300.0 | Upper Limit/Auto Up | 349.95V | 349.950V | 349.9500V |
| | | Nominal | 300.00V | 300.000V | 300.0000V |
| | | Auto Down | 99.99V | 99.999V | 99.9999V |
| | | Lower Limit | 27.00V | 27.000V | 27.0000V |
| **100V** | 30.0 < *Nrf* ≤ 100.0 | Upper Limit/Auto Up | 119.950V | 119.9500V | 119.95000V |
| | | Nominal | 100.000V | 100.0000V | 100.00000V |
| | | Auto Down | 29.999V | 29.9999V | 29.99999V |
| | | Lower Limit | 9.000V | 9.0000V | 9.00000V |
| **30V** | 10.0 < *Nrf* ≤ 30.0 | Upper Limit/Auto Up | 34.995V | 34.9950V | 34.99500V |
| | | Nominal | 30.000V | 30.0000V | 30.00000V |
| | | Auto Down | 9.999V | 9.9999V | 9.99999V |
| | | Lower Limit | 2.700V | 2.7000V | 2.70000V |
| **10V** | 3.0 < *Nrf* ≤ 10.0 | Upper Limit/Auto Up | 11.9950V | 11.99500V | 11.995000V |
| | | Nominal | 10.0000V | 10.00000V | 10.000000V |
| | | Auto Down | 2.9999V | 2.99999V | 2.999999V |
| | | Lower Limit | 0.9000V | 0.90000V | 0.900000V |
| **3V** | 1.0 < *Nrf* ≤ 3.0 | Upper Limit/Auto Up | 3.4995V | 3.49950V | 3.499500V |
| | | Nominal | 3.0000V | 3.00000V | 3.000000V |
| | | Auto Down | 0.9999V | 0.99999V | 0.999999V |
| | | Lower Limit | 0.2700V | 0.27000V | 0.270000V |
| **1V** | 0.3 < *Nrf* ≤ 1.0 | Upper Limit/Auto Up | 1.19950V | 1.199500V | 1.1995000V |
| | | Nominal | 1.00000V | 1.000000V | 1.0000000V |
| | | Auto Down | .29999V | .299999V | .2999999V |
| | | Lower Limit | .09000V | .090000V | .0900000V |
| **300mV** | 0 < *Nrf* ≤ 0.3 | Upper Limit/Auto Up | .34995V | .349950V | .3499500V |
| | | Nominal | .30000V | .300000V | .3000000V |
| | | Lower Limit | .02700V | .027000V | .0270000V |

(WBV Range Points - *overleaf*)

# 4920M ACV and WBV Significant Range Points (Contd.)

| WBV Range | WBV *Nrf* selection limits | Range Point | Res5 RESL5 | Res6 RESL6 | Res7 RESL7 |
|---|---|---|---|---|---|
| 3V | *Nrf* > 1.0 | Upper Limit | 3.5995V | 3.59950V | 3.599500V |
| | | Nominal | 3.0000V | 3.00000V | 3.000000V |
| | | Lower Limit | 0.3000V | 0.30000V | 0.300000V |
| 1V | *Nrf* ≤ 1.0 | Upper Limit | 1.19950V | 1.199500V | 1.1995000V |
| | | Nominal | 1.00000V | 1.000000V | 1.0000000V |
| | | Lower Limit | .07950V | .079500V | .0795000V |

# SECTION 5 SYSTEMS APPLICATION via the IEEE 488 INTERFACE

## Contents List

# Alphabetical Index of IEEE 488.2 Codes used in the 4920M



## Normal Operations

## Calibration Operations

| Device-Dependent Command/Query Header | Description | Page 5- |
|---|---|---|
| CAL? | Triggers calibration operation | 65 |
| CALINT | Sets calibration interval | 71 |
| CALSEAL | Copies current date to 'Last Cal' date | 68 |
| CLRMEM | Clears calibration memory | 75 |
| DATE | Sets calendar and clock | 69 |
| DUMP? | Returns values from calibration memory | 72 |
| ENBCAL | Enables calibration | 64 |
| EXITCAL | Disables calibration | 68 |
| SERIAL (12 ASCII chars) | Allows access to change the serial number | 70 |
| *PUD | Allows entry of user data to protected store | 74 |

## Other Mnemonics

| | | Page 5- |
|---|---|---|
| ALL | Parameter of CLRMEM command (Selects all non-volatile calibration memory). | 75 |
| AUTO | Parameter of ACV and WBV commands (Automatic range selection). | 28/30 |
| EXT | Parameter of TRG_SRCE command (Selects external triggers). | 32 |
| FILT100HZ | Parameter of ACV command (Selects 100Hz filter). | 28 |
| FILT10HZ | Parameter of ACV and ENBCAL commands (Selects 10Hz filter). | 28/64 |
| FILT_OFF | Parameter of WBV command (Selects 100Hz filter for 1V range only). | 30 |
| FILT_ON | Parameter of WBV command (Selects 10Hz filter for 1V range only). | 30 |
| FLAT | Parameter of ENBCAL/CLRMEM commands (Selects 'Flatness' calibration mode). | 64/67/75 |
| FREQ | Parameter of ENBCAL command (Selects frequency calibration of counter). | 64 |
| GAIN | Parameter of ENBCAL/CLRMEM commands (Selects 'Gain' calibration mode). | 64/67/75 |
| INT | Parameter of TRG_SRCE command (Selects Internal Triggers). | 32 |
| LINA | Parameter of ENBCAL/CLRMEM commands (Selects 'Linearity' calibration of ACV function). | 64/67/75 |
| LINW | Parameter of ENBCAL command (Selects 'Linearity' calibration of WBV function). | 64/67 |
| RESL5 | Parameter of ACV and WBV commands (Selects 'better than 100ppm' resolution). | 28/30 |
| RESL6 | Parameter of ACV and WBV commands (Selects 'better than 10ppm' resolution). | 28/30 |
| RESL7 | Parameter of ACV and WBV commands (Selects 'better than 1ppm' resolution). | 28/30 |

# SECTION 5 SYSTEMS APPLICATION VIA THE IEEE 488 INTERFACE

## Introduction

This first part of Section 5 gives the information necessary to put the 4920M into operation on the IEEE 488 bus. As some operators will be first-time users of the bus, the text is pitched at an introductory level. For more detailed information, refer to the standard specification, which appears in the publications ANSI/IEEE Std. 488.1-1987 and IEEE Std. 488.2-1988.

## Section Contents

The section is divided so as to group certain types of information together. These divisions are:

**Interface Capability** - IEEE 488.1 subsets which are implemented in the model 4920M, satisfying IEEE 488.2.
**Interconnections** - the rear panel IEEE 488 connector and its pin designations.
**Typical System** - a brief view of a typical process using the 4920M to measure the output from a programmable AC voltage source.
**Using the 4920M in a System** - addressing, remote operation and programming guidance - introduction to syntax diagrams.
**Message Exchange** - a simplified model showing how the 4920M deals with incoming and outgoing messages.
**Service Request** - why the 4920M needs the controller's attention and how it gets it.
**Retrieval of Device Status Information** - how the IEEE 488.2 model is adapted to the 4920M.
**Programming Messages** - detailed descriptions of both common and device-specific commands and queries.

## Interface Capability

### IEEE Standards 488.1 and 488.2

The 4920M conforms to the Standard Specification IEEE 488.1-1987: 'IEEE Standard Digital Interface for Programmable Instrumentation', and to IEEE 488.2-1988: 'Codes, Formats, Protocols and Common Commands'.

### The 4920M in IEEE 488.2 Terminology

In IEEE 488.2 terminology the 4920M is a **device** containing a **system interface.** It can be connected to a **system** via its **system bus** and set into programmed communication with other bus-connected **devices** under the direction of a system **controller.**

### Programming Options

The instrument can be programmed via the IEEE Interface, to:

- Change its operating state (Function, Range, Resolution etc).
- Transmit results of measurements, and its own status data, over the bus.
- Request service from the system controller.

## Capability Codes

To conform to the IEEE 488.1 standard specification, it is not essential for a device to encompass the full range of bus capabilities.
But for IEEE 488.2, the device must conform exactly to a specific subset of IEEE 488.1, with a minimal choice of optional capabilities.

The IEEE 488.1 document describes and codes the standard bus features, for manufacturers to give brief coded descriptions of their own interfaces' overall capability. For IEEE 488.2, this description is required to be part of the device documentation. A code string is often printed on the product itself.

The codes which apply to the 4920M are given in table 5.1, together with short descriptions. They also appear on the rear of the instrument next to the interface connector. These codes conform to the capabilities required by IEEE 488.2.

Appendix C of the IEEE 488.1 document contains a fuller description of each code.

| IEEE 488.1 Subset | Interface Function |
|---|---|
| SH1 | Source Handshake Capability |
| AH1 | Acceptor Handshake Capability |
| T6 | Talker (basic talker, serial poll, unaddressed to talk if addressed to listen) |
| L4 | Listener (basic listener, unaddressed to listen if addressed to talk) |
| SR1 | Service Request Capability |
| RL1 | Remote/Local Capability (including Local Lockout) |
| PP0 | No Parallel Poll Capability |
| DC1 | Device Clear Capability |
| DT1 | Device Trigger Capability |
| C0 | No Controller Capability |
| E2 | Open-Collector and Three-State Drivers |

***Table 5.1 IEEE Interface Capability***

## Bus Addresses

When an IEEE 488 system comprises several instruments, a unique 'Address' is assigned to each to enable the controller to communicate with them individually.

Only one address is required for the 4920M, as the controller adds information to it to define either 'talk' or 'listen'. The method of setting the address, and the point at which the user-initiated address is recognized by the 4920M, is detailed on page 5-6.

The 4920M has a single primary address, which can be set by the user to any value within the range from 0 to 30 inclusive. It cannot be made to respond to any address outside this range.
Secondary addressing is not programmed.

## Interconnections

Instruments fitted with an IEEE 488 interface communicate with each other through a standard set of interconnecting cables, as specified in the IEEE 488.1 Standard document.

The interface socket, SK7, is fitted on the rear panel. It accommodates the specified connector, whose pin designations are also standardized as shown in Fig. 5.1 and Table 5.2

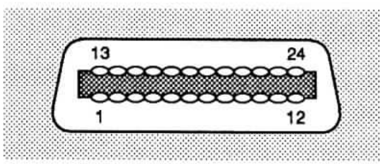



***Fig 5.1 Connector SK7 - Pin Layout***

| Pin No. | Name | Description |
|---|---|---|
| 1 | DIO 1 | Data Input/Output Line 1 |
| 2 | DIO 2 | Data Input/Output Line 2 |
| 3 | DIO 3 | Data Input/Output Line 3 |
| 4 | DIO 4 | Data Input/Output Line 4 |
| 5 | EOI | End or Identify |
| 6 | DAV | Data Valid |
| 7 | NRFD | Not Ready For Data |
| 8 | NDAC | Not Data Accepted |
| 9 | IFC | Interface Clear |
| 10 | SRQ | Service Request |
| 11 | ATN | Attention |
| 12 | SHIELD | Screening on cable (connected to 4920M safety ground) |
| 13 | DIO 5 | Data Input/Output Line 5 |
| 14 | DIO 6 | Data Input/Output Line 6 |
| 15 | DIO 7 | Data Input/Output Line 7 |
| 16 | DIO 8 | Data Input/Output Line 8 |
| 17 | REN | Remote Enable |
| 18 | GND 6 | Gnd wire of DAV twisted pair |
| 19 | GND 7 | Gnd wire of NRFD twisted pair |
| 20 | GND 8 | Gnd wire of NDAC twisted pair |
| 21 | GND 9 | Gnd wire of IFC twisted pair |
| 22 | GND 10 | Gnd wire of SRQ twisted pair |
| 23 | GND 11 | Gnd wire of ATN twisted pair |
| 24 | GND | 4920M Logic Ground (internally connected to Safety Ground) |

***Table 5.2 Socket SK7 - Pin Designations***

# Using the 4920M in a System

## Addressing the 4920M

### Address Recognition

With an address selected in the range 0 to 30; control may be manual, or remote as part of a system on the Bus. The address must be the same as that used in the controller program to activate the 4920M. The 4920M is always aware of its stored address, responding to Talk or Listen commands from the controller at that address. When the address is changed by the user, the 4920M recognizes its new address and ignores its old address, as soon as it is stored by the user pressing the **Enter** key in the **ADDRESS** menu.

### Setting the Bus Address

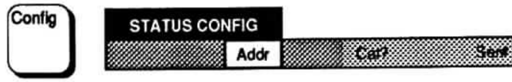
The instrument address can only be set manually; using the ADDRESS menu, which is accessed via the STATUS and STATUS CONFIG menus.

To change the address, proceed as follows:

- Press the **Status** key to see the **STATUS** menu:


This menu defines three positions on the dot-matrix display (refer to Section 3 for details). The soft keys are deactivated, and play no part in setting the address.

- Press the **Config** key to see the **STATUS CONFIG** menu:




This menu defines three soft *menu* keys; at present we are interested only in the Addr key.

Addr: displays the ADDRESS menu, to review or change the IEEE-488 bus address of the instrument.

### ADDRESS Menu

- Press the **Addr** soft key to see the **ADDRESS** menu:


This menu permits entry of a value to be used as an IEEE-488 bus address. Initially, the menu displays the present address value (in the position shown above by XX), and the numeric-keyboard keys are activated. Any valid numeric value (0-30) may be entered, an invalid address resulting in the display message '1007: data entry error'.

Pressing Enter stores the new value (or restores the old value if unchanged), but pressing Quit leaves the old value intact. Either Enter or Quit causes exit back to the STATUS CONFIG menu, then press any required function key to escape.

# Remote Operation

## General

When the 4920M is operating under the direction of the controller, the legend rem appears on the Main display, and all front panel controls are disabled except Power and Local.

The power-up sequence is performed as for manual operation. The 4920M can be programmed to generate an SRQ at power-up, also preparing a status response for transmission to the controller when interrogated by a subsequent serial poll.

## Transfer to Local Operation (GTL)

The 4920M can be switched into 'Local' operation (by Command GTL), permitting a user to take manual control from the front panel. The system controller regains 'Remote' control by sending the overriding command:

*Listen Address* with *REN* true

The controller addresses the 4920M as a listener with the Remote Enable management line true (Low). This returns the 4920M from local to remote control.

## 'Device Clear'

Either of the commands *DCL* or *SDC* will force the following instrument states:

- all IEEE 488 input and output buffers cleared;
- parser reset to the beginning of a message;
- any device-dependent message bus holdoffs cleared.

These commands **will not**:

- change any settings or stored data within the device except as listed above;
- interrupt analog input;
- interrupt or affect any functions of the device not associated with the IEEE 488 system;
- change the status byte.

## Levels of Reset

Three levels of reset are defined for IEEE 488.2 controllers, a complete system reset being accomplished by resetting at all three levels, in order, to every device. In other circumstances they may be used individually or in combination:

**IFC** Bus initialization;
**DCL** Message exchange initialization;
***RST** Device initialization.

The effects of the *RST command are described on page 5-38.

# Programming Guidance

## Programming Elements

The 4920M operates within the syntax demands of IEEE 488.2. To alter its configuration, elicit information etc., it requires to be sent an address code followed by commands or queries.

The smallest program element capable of activation is called a 'program message unit' (*pmu*), which must conform to a standard structure (detailed later in this section). One *pmu* can be sent on its own, followed by a recognized 'terminator'; in which case it is known as a 'program message'. Several formally-complete *Pmus* can be concatenated together, using semi-colons as separators, to form a program message. All program messages must be correctly terminated.

## Activation

The 4920M cannot activate any commands or queries until it receives a message unit separator (;) or a correct message terminator.

The message terminator for the 4920M is the Hex number ØA, characterized in IEEE 488.2 as 'NL'. Alternatively, the 'End or Identify' (EOI) line can be set true with the last byte to be sent; this is represented on the syntax diagram by /^END/.

To assist in eliminating incorrect commands or queries, the 4920M checks for errors in the message, and can generate a service request (SRQ) if a syntax error occurs. To ensure that a prohibited state is not set up, it also checks each program message unit for validity. If it finds any errors in this phase, the message unit is ignored.

For Example:
The invalid *pmu*: WBV FILT100HZ will generate an execution error and the whole message unit is discarded.

## Conformance to IEEE 488.2

IEEE 488.2 defines sets of Mandatory and Optional Common Commands/Queries along with a standard method of Status Reporting. The 4920M conforms with all Mandatory Commands/Queries but not all the Options, and uses the defined Status Reporting method.
Note: Commands prefaced by asterisk, eg *TRG, are standard-defined 'Common' commands.

In addition to these Common Commands, the 4920M has a set of Device-Dependent Commands, defined by Datron to program the instrument into its various functions and ranges. IEEE 488.2 defines how these commands should be linked or separated (ie the syntax is defined). These device-dependent commands have been designed to be self-explanatory, while conforming to the standard-defined syntax.

The IEEE 488.2 also requires certain 'Device Documentation' to be supplied by its manufacturer. This data is included within the text of this section, and is indexed by Appendix A at the back of the section.

## Command Formation

The following paragraphs describe the commands that are used to program the 4920M.

A command (or 'Program Message Unit') can merely comprise a simple alphabetic code. But if there are alternative ways of programming within a command, this is signified by using a 'Command Program Header', followed by the appropriate 'Program Message Elements'.

An example of a simple command is the query 'FREQ?', which recalls the frequency of the most-recent reading.

An example of a more complex command is:

'ACV 10,RESL6,FILT10HZ'

which will program the instrument to ACV function, 10V range, 10ppm resolution and 10Hz filter selected. In this example, ACV is the Command Program Header, while 10,RESL6, and FILT_10 are all Program Message Elements.

### 'Forgiving Listening'

IEEE 488.2 closely specifies the format of messages to be transmitted by a conforming 'talker' device. This imposes consistencies which make clear the exact meaning of any message received from the device.

Conversely, a conforming 'listener' device should be able to interpret a variety of received messages, which may have been originated by different types of controller. By limiting the language to the commonly-used ASCII character set, this condition is partly met. But within the ASCII set, the IEEE 488.2 also insists that certain characters be used only as command symbols to separate and specify the elements of concatenated strings, and for no other purpose:

- A succession of Message Units making up a complete Program Message must each be separated by a semi-colon (;).
- A Program Header is separated from its Message Elements by 'white space' - (i.e one or more non-printing ASCII characters in the ranges Hex ØØ to Ø9 and ØB to 2Ø) - denoted here by {phs} - Program Header Separator.
- Successive Message Elements of a Message Unit are each separated by a comma (,).
- In numeric data strings, all variants of a particular family of numeric representations (Nrf) are acceptable.
- No differentiation is made between upper and lower case characters.
- Program Messages may be terminated by a Line Feed - (ie the ASCII character at Hex ØA) - denoted by {NL} (Newline), or by EOI true with the last byte, or both.

An example of a complete Program Message could be:

ACV{phs}10,ResL6,fiLT10hz;*TrG{NL}

Unfortunately, most of this message appears visually as a cramped string of characters. This can be confusing to read because of the lack of spaces between the various elements. So IEEE 488.2 also permits optional 'white space' to be inserted within a string to improve its visual intelligibility, providing certain rules are followed:

- The 'white space' characters must be as defined earlier for the Program Header Separator.
- White space can be inserted *before* a Program Message Separator (;).
- White space can be inserted *before and/or after* a Program Message Element Separator (,).
- White space can be inserted *before and/or after* an Exponent Symbol (E) in a numeric string.

These rules are programmed into the 4920M, to ensure that it does not reject received commands or data which obey these rules, thus giving it the nature of a 'Forgiving Listener'.

## IEEE 488.2 Syntax Diagrams

To standardize the approach to programming, the IEEE 488.2 Standard has introduced a form of 'Syntax Diagram', in which the possible command formation for particular messages can be given. The IEEE 488.2 syntax has been adhered to, so in the following descriptions of device-dependent commands, we have adopted the standard syntax diagram, with modified style to fit this handbook. A word of explanation about the notation is needed, and the diagrams are defined, although they are virtually self-explanatory.

### Notation

- Syntactic elements are connected by lines with directional symbols to indicate the flow, which generally proceeds from left to right.
- Repeatable elements have a right-to-left reverse path shown above and around them, which can also contain a separator such as a comma.
- When it is possible to bypass elements, a left-to-right path is shown below and around them.
- When there is a choice of elements, the path branches to the choices.

The example program message:

'ACV{phs}10,RESL6,FILT_10;*TRG{NL}',

mentioned earlier, is a syntactic string derived from the ACV function and *TRG diagrams, which appear in the range of diagrams described below. Note that 'phs' means 'program header separator', one or more white-space characters as mentioned earlier.

## Syntax Diagrams in this Handbook

The following paragraphs describe the syntax diagrams used in this handbook. Some repetition of earlier matter is inevitable, but at this point we are more concerned with the syntax diagrams.

### Hierarchy of Syntactic Elements

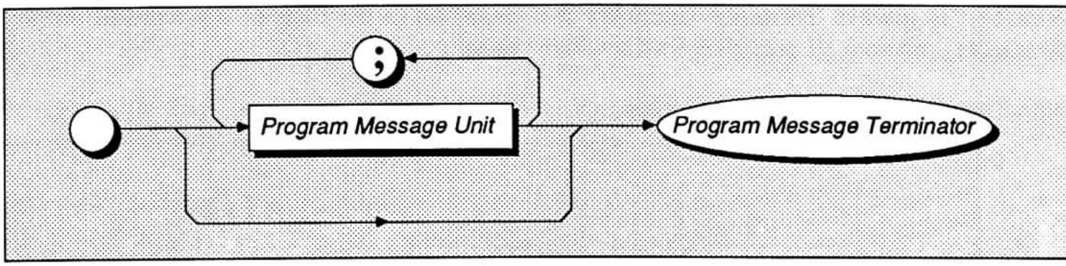
All messages are subject to the protocols of addressing and handshake defined in the IEEE 488.1 Standard document. Within these protocols, messages are characterized by the presence of terminators, each of which seals the set of syntactic elements sent since the previous terminator to form a 'Program Message'.

### The Program Message

Each Program Message may consist of only one syntactic element plus its terminator, or may be subdivided into many 'Program Message Units', separated by semi-colons (;) which are known as 'Program Message Unit Separators'. Thus the semi-colon cannot be used for any other purpose.

As you can see from the diagram, multiple Program Message Units can be sent if they are separated using semi-colons (shown in the repeat path). The block named 'Program Message Unit' therefore represents **either** repeats of the same unit, **or** a set of different units, **or** a mixture of both. The starting circle is a device used only for the diagram; there is no requirement to use a special character to start a message, providing the previous message was correctly terminated. It is possible to send only the terminator as a complete Program Message (as shown by the forward bypass path), but this feature has little use when programming the 4920M.

## Syntax Diagram of a Simple Program Message




**Character Usage**
Notice that the names of some elements are shown here in italics. This agrees with the convention used on the syntax diagrams in this handbook, which sets 'non-literal' text (names given to particular elements) in italics, whereas 'literal' text (the actual characters to be sent, such as the semi-colon in the diagram) is shown in plain-text capitals.

**Upper/Lower Case Equivalence**
The plain-text capitals are not demanded by the standard, and the 4920M will not differentiate between upper and lower case characters in literal program text. Either or both can be used, mixed upper and lower case if this conveys an advantage.

**Numeric Representation**
Several commands and queries used for the 4920M require transmission and reception of numbers. Decimal formats are generally used.

The IEEE 488.2 document specifies formats which ensure that a device is 'forgiving' when receiving program or query commands, but 'precise' when transmitting responses to queries.

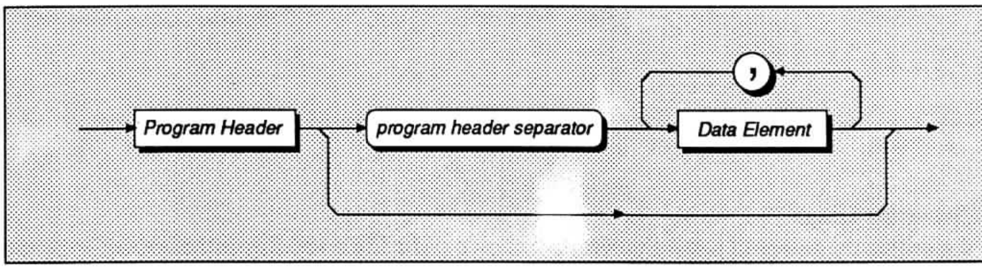
For program data it insists that a device must accept the decimal 'Flexible Numeric Representation (*Nrf*)', which is a flexible version of three numeric representations (Nr1, Nr2 and Nr3) defined by ANSI X3.42-1975 [2]. The 4920M complies.

Decimal numeric response data from the 4920M employs either Nr1 or Nr3 format, usage depending on the particular response. In this handbook, all syntax diagrams for query messages are accompanied by a paragraph which indicates the response format. Users are left in no doubt as to the construction of the response.

**The Program Message Unit**

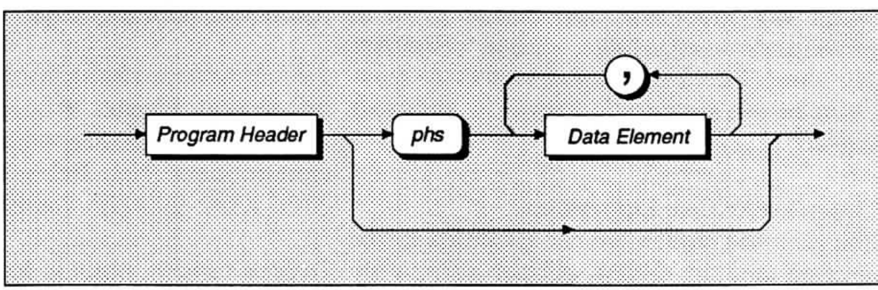
Program Message Units (PMUs) can be 'Terminal' or 'Non-terminal'. The final PMU in any Program Message is always Terminal (includes the terminator), whereas all preceding PMUs within the Program Message are obviously Non-terminal. Most of the commands in this handbook are described in the form of non-terminal message units:

Non-Terminal Program Message Unit




To save space, the name 'program header separator' is abbreviated to 'phs'.

Use of *phs*

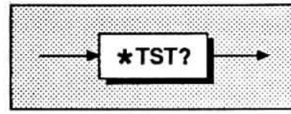



**The Command Program Header**

Several versions are defined by the IEEE 488.2 Standard document. The 'Simple', 'Common' and 'Query' headers are designed into the 4920M, but not 'Compound' headers.

The asterisk (Common) and question mark (Query) are defined separately by the standard document, but as they are inseparable from the command, they are shown on the 4920M syntax diagrams in the same block as the program mnemonic (abbreviated Format). For example: the command for Operational Selftest (*TST?) is shown in abbreviated, rather than full format.

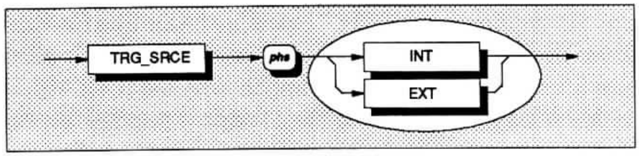
Common Query - Abbreviated Format



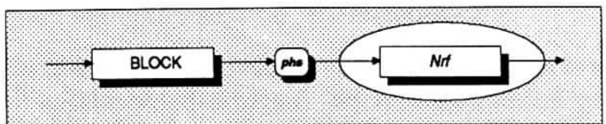
**Program Data Elements**

Four versions of the defined program data elements are employed. They are emphasized in the following syntax diagrams, which are examples from the list of commands available for the 4920M:

Character

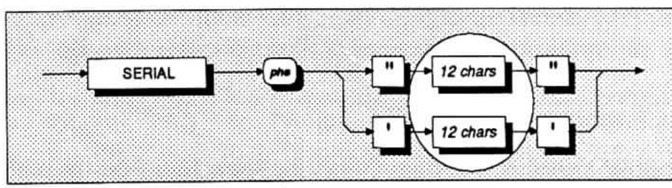



Decimal Numeric




(***Nrf*** can be expressed in any of the ways defined by the Standard document)

String



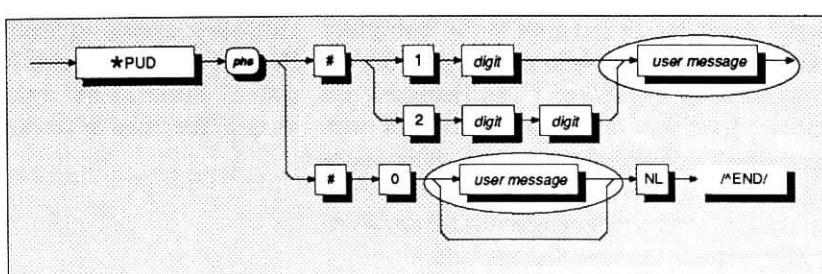

(The string size is defined)

Arbitrary Block Data Elements

Both the 'Definite' and 'Indefinite' forms specified in the Standard document are used, as shown in the Syntax diagram below. The *user message* must be limited to a maximum of 63 bytes.

The definite form can be fitted into a string of message units, but the indefinite form (lower path) has no exit to further message units. In this case the program message must be terminated to inform the instrument that the block is complete.

Note that the slash-delimited /^END/ box is not outlined. This is to draw attention to the fact that it is not a data element, but represents the EOI line being set true with the last byte 'NL' to terminate the program message.



# Message Exchange

## IEEE 488.1 Model

The 4920M conforms to the requirements of the IEEE 488.1 Standard, in respect of the interactions between its device system interface and the system bus. Its conformance is described by the interface capability codes listed in Table 5.1 on page 5-2. In addition, the 4920M is adapted to the protocols described by the IEEE 488.2 model, as defined in that standard's specification.

## IEEE 488.2 Model

The IEEE 488.2 Standard document illustrates its Message Exchange Control Interface model at the detail level required by the device designer. Much of the information at this level of interpretation (such as the details of the internal signal paths etc.) is transparent to the application programmer. However, because each of the types of errors flagged in the Event Status Register are related to a particular stage in the process, a simplified 4920M interface model can provide helpful background. This is illustrated in Fig. 5.2, together with brief descriptions of the actions of its functional blocks.

## 4920M Message Exchange Model

**Input/Output Control** transfers messages from the 4920M output queue to the system bus; and conversely from the bus to either the input buffer, or other predetermined destinations within the device interface. Its interaction with the controller, via the system bus, is subject to the IEEE 488.1 management and handshake protocol. It receives the Status Byte from the status reporting system, as well as the state of the request service bit which it imposes on bit 6 of the Status Byte (ultimately on bus line DIO 7) in the event of a serial poll. Bit 6 reflects the 'Request Service state *true*' condition of the interface.

## Incoming Commands and Queries

The **Input Buffer** is a first in - first out queue, which has a maximum capacity of 128 bytes (characters). Each incoming character in the I/O Control generates an interrupt to the instrument processor which places it in the Input Buffer for examination by the Parser. The characters are removed from the buffer and translated with appropriate levels of syntax checking. If the rate of programming is too fast for the Parser or Execution Control, the buffer will progressively fill up. When the buffer is full, the handshake is held.

The **Parser** checks each incoming character and its message context for correct Standard-defined generic syntax, and correct device-defined syntax. Offending syntax is reported as a **Command Error**, by setting *true* bit 5 (CME) of the Standard-defined Event Status register (refer to the sub-section 'Retrieval of Device Status Information').

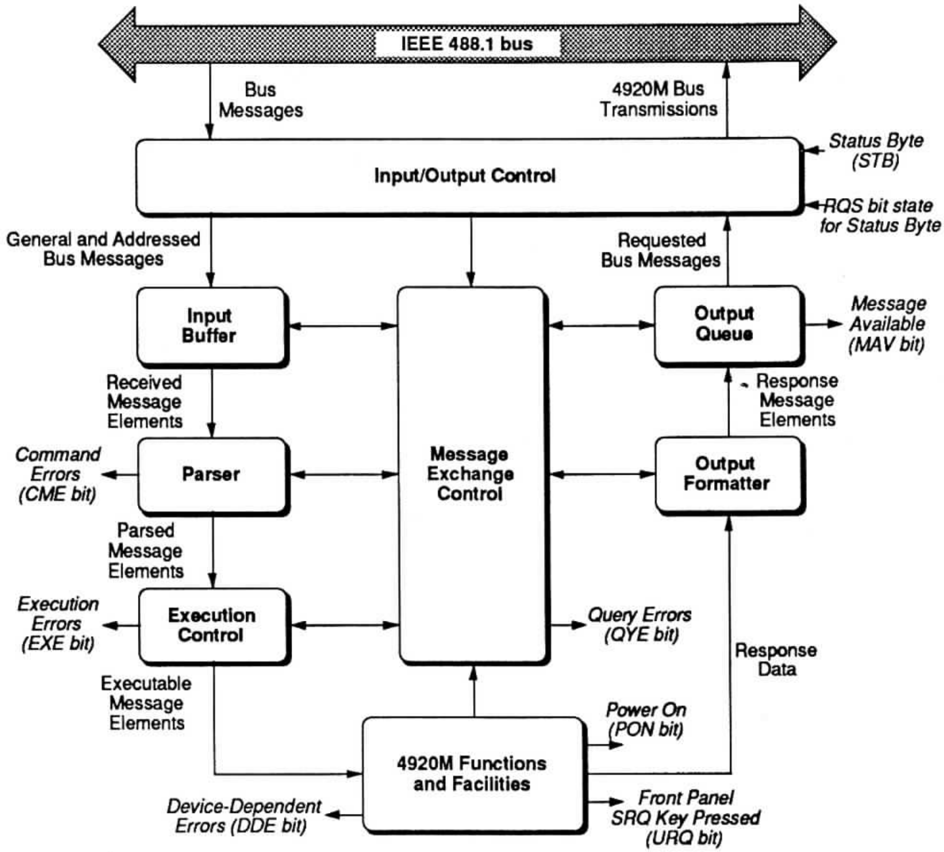



**Fig. 5.2 - Simplified 4920M Message Exchange Model**

**Execution Control** receives successfully parsed messages, and assesses whether they can be executed, given the currently-programmed state of the 4920M functions and facilities. If a message is not viable (eg the selftest common query: *TST? when calibration is successfully enabled); then an Execution Error is reported, by setting *true* bit 4 (EXE) of the Standard-defined Event Status register, and placing an error description number in a queue associated with the EXE bit.

Viable messages are executed in order, altering the 4920M functions, facilities etc. Execution does not 'overlap' commands; instead, the 4920M Execution Control processes all commands 'Sequentially' (ie. waits for actions resulting from the previous command to complete before executing the next).

## 4920M Functions and Facilities

The 4920M Functions and Facilities block contains all the device-specific functions and features of the 4920M, accepting Executable Message Elements from Execution Control and performing the associated operations. It responds to any of the elements which are valid Query Requests (both IEEE 488.2 Common Query Commands and 4920M Device-specific Command Queries) by sending any required Response Data to the Response Formatter (after carrying out the assigned internal operations).

**Device-dependent** errors are detected in this block. Bit 3 (DDE) of the Standard-defined Event Status register is set true when an internal operating fault is detected, for instance during a self test. Each reportable error has a listed number, which is appended to an associated queue as the error occurs.

This block also originates a local power-on message by the action of the 4920M line power being applied. Bit 7 (PON) of the Standard-defined Event Status register is set true when the instrument power transits from off to on (refer to the sub-section 'Retrieval of Device Status Information').

The front-panel **SRQ** key allows users to initiate an SRQ (providing the appropriate status register bits are enabled). Bit 6 (URQ) of the Standard-defined Event Status register is set true when the key is pressed, and set to false by reading the Event Status register or if the registers are cleared by *CLS.

## Trigger Control

Two types of message are used to trigger the 4920M A-D into taking a measurement:

**GET** (IEEE 488.1-defined)
***TRG** (IEEE 488.2-defined)

In the 4920M both GET and *TRG messages are passed through the Input Buffer, receiving the same treatment as program message units, being parsed and executed as normal.

## Outgoing Responses

The **Response Formatter** derives its information from Response Data (being supplied by the Functions and Facilities block) and valid Query Requests. From these it builds Response Message Elements, which are placed as a Response Message into the Output Queue.

The **Output Queue** acts as a store for outgoing messages until they are read over the system bus by the Controller. For as long as the output queue holds one or more bytes, it reports the fact by setting *true* bit 4 (Message Available - MAV) of the Status Byte register. Bit 4 is set *false* when the output queue is empty (refer to the sub-section 'Retrieval of Device Status Information').

## 'Query Error'

This is an indication that the controller is following an inappropriate message exchange protocol, resulting in the following situations:

- ***Interrupted Action.*** When the 4920M has not finished outputting its **Response Message** to a **Program Query**, and is interrupted by a new **Program Message.**

- ***Unterminated Action.*** When the controller attempts to read a **Response Message** from the 4920M without having first sent the complete **Query Message** (including the **Program Message Terminator**) to the instrument.

The Standard document defines the 4920M's response, part of which is to set *true* bit 2 (QYE) of the Standard-defined Event Status register.

# Service Request (SRQ)

## IEEE 488.1 Model

The IEEE 488.1 model provides for a separate line (SRQ line) on the system bus, to be set true (Low) by the device to request service of the controller. The model defines the subsequent action by the controller, and in the 4920M the serial poll facility has been incorporated.

The controller polls each device on the system bus in sequence, reading a 'Status Byte' onto DIO lines 8-1, whereby the bit on the DIO 7 line (Request Service bit) indicates whether that device was the originator of the request for service.

### Reasons for Requesting Service

There are two main reasons for the 4920M to request service from the controller:

- When the 4920M message exchange interface discovers a system programming error;
- When the 4920M is programmed to report significant events by SRQ.

The significant events vary between types of devices; thus there is a class of events which are known as 'Device-Specific'. These are determined by the device designer and included in the device operating program.

## IEEE 488.2 Model

The application programmer can enable or disable the event(s) which are required to originate an SRQ at particular stages of the application program. The IEEE 488.2 model incorporates a flexible extended status reporting structure in which the requirements of the device deigner and application programmer are both met.

This structure is described in the next sub-section, dealing with 'Retrieval of Device Status Information'. As SRQ provision is integral to the structure, the description of the implementation of SRQ features is covered in that sub-section rather than in this.

# Retrieval of Device Status Information

## Introduction

For any remotely-operated system, the provision of up-to-date information about the performance of the system is of major importance. This is particularly so in the case of systems which operate under automatic control, as the controller requires the necessary information feedback to enable it to progress the programmed task, and any break in the continuity of the process can have serious results.

When developing an application program, the programmer needs to test and revise it, knowing its effects. Confidence that the program elements are couched in the correct grammar and syntax (and that the program commands and queries are thus being accepted and acted upon), helps to reduce the number of iterations needed to confirm and develop the viability of the whole program. So any assistance which can be given in closing the information loop must benefit both program compilation and subsequent use.

## The 4920M Status Reporting Structure

In a closely-specified Standard such as the IEEE 488.2, we should expect to find a well-defined and comprehensive status reporting facility, and this is indeed the case. Not only does the Standard establish regular methods of retrieving the information, but it also provides the means for the device designer to build a status-reporting structure which is pertinent to the nature of the device. Within this structure the application programmer is then given a wide choice to decide on the sort of information required at each stage in the program.

***Note:*** *The registers have binary weightings - the numbers in the boxes are bit numbers, not weighted values.*

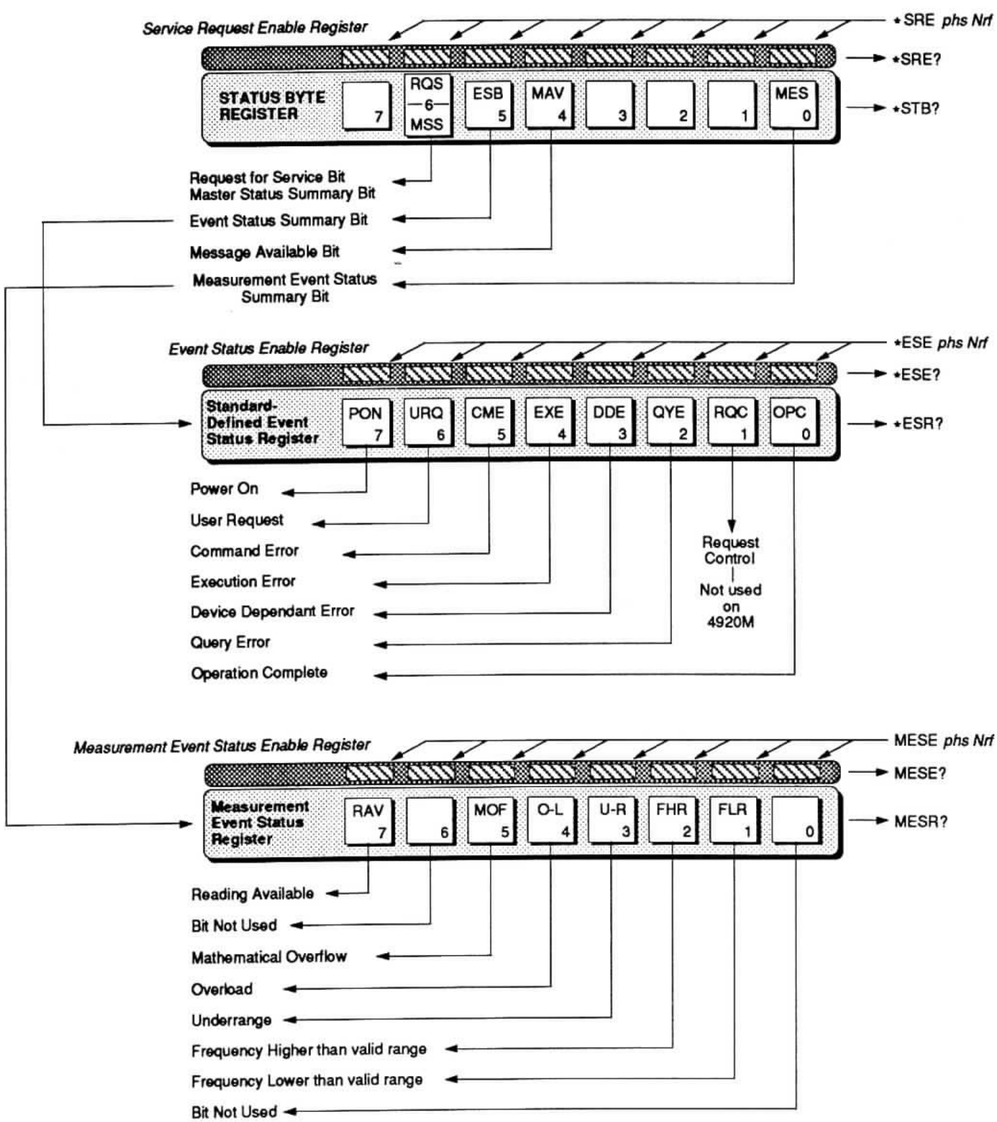



***Fig. 5.3 - 4920M Status Reporting Structure***

**Standard-Defined and Device-Specific Features**

In the 4920M, the structure has been developed into three main registers, as follows:

- **The 'Status Byte Register'**
  contains the **'Status Byte'**, which summarizes the remainder of the structure. Bits 6-4 are Standard-defined, but bits 3-0 and 7 are provided for the device designer to define.
- **The 'Event Status Register'**
  Defined by the standard, contains the **'Event Status Byte'**, whose component bits report Standard-defined types of events. This register is summarized by the 'ESB' bit 5 in the Status Byte.
- **The 'Measurement Event Status Register'**
  Up to five Device-Specific Event Status Registers or queues can be defined by the device designer; in this case only one register is defined, for the **'Measurement Event Status Byte'**, whose component bits are device-specific (ie. to the 4920M). It is summarized by the 'MES' bit 0 in the Status Byte.

Although the Event Status Byte bits are defined by the Standard, they are permitted to summarize device-specific events (eg. EXE is associated with a list of execution errors related to the 4920M programmed condition, and DDE is associated with a list of device-dependent errors related to 4920M internal faults). These extensions, with the structures based on bits 3-0 and 7 of the Status Byte, allow the device designer a wide latitude to match status reporting to the requirements of the device.

**Access via the Application Program**

The application designer has access to three enable registers (one for each main register - Fig. 5.3). The application program can enable or disable any individual bit in these registers.

Each bit in the two event status bytes remains in *false* condition unless its assigned event occurs, when its condition changes to *true*. If an event is to be reported, its corresponding bit in the enable byte will already have been set *true* by the application program, using the number *Nrf* (defined as a decimal numeric from 0 to 255 in any common format). Then when this event occurs and changes its event bit from *false* to *true*, the appropriate summary bit in the Status Byte (ESB or MES) is also set true. If this summary bit is also enabled, then the 4920M will generate an SRQ by causing the SRQ line on the system bus to be set *true* (low).

Thus the application programmer can decide which assigned events will generate an SRQ, by enabling their event bits and then enabling the appropriate summary bit(s) in the Status Byte. The controller can be programmed to read the Status Byte during a resulting serial poll, and be directed to the appropriate Event Register to discover which event was responsible for originating the SRQ.

The Status Byte is the only one of the six which can be read *bitwise* on to the DIO lines of the system bus, and then only by a serial poll to which special conditions are attached. Each byte can be read by a suitable query (*STB?, *ESR? or MESR? for the status and event bytes; *SRE?, *ESE?, or MESE? for the enable bytes) and will be presented as an ASCII decimal numeric, which when rounded and expressed in binary, represents the bit pattern in the register. This numeric form is also used to set the enabling registers to the required bit-patterns. The detail of the data in each byte is explained in the following paragraphs, and in the command descriptions.

## Types of Status Information Available

Three main categories of information are provided for the controller:

### Status Summary Information

Contained within the 'Status Register', the 'Status Byte' (STB) consists of flag bits which direct the controller's attention to the type of event which has occurred. Four bits are employed in the 4920M; these are described in detail later, but two ('ESB' and 'MES') appear in the following paragraphs.

**Standard-defined events:**

- Power On - the instrument line power has been switched on and the associated operational selftest has been completed successfully.
- User Request - the 'SRQ' key on the front panel has been pressed.
- Command Error - a received bus command does not satisfy the syntax rules programmed into the instrument interface's parser, and so is not recognized as a valid command.
- Execution Error - a received command has been successfully parsed, but it cannot be executed owing to the currently-programmed condition of the instrument.
- Device-Dependent Error - a reportable internal operating fault has been detected. This may be failure of a self-test.
- Query Error - the controller is following an inappropriate message exchange protocol, in attempting to read data from the output queue.
- Request Control - provided for devices which are able to assume the role of controller. This capability is not available in the 4920M.
- Operation Complete - initiated by a message from the controller, indicates that the 4920M has completed all selected pending operations.

These events are flagged in the 8-bit latched 'Event Status Register' (ESR), read-accessible to the controller. The user's application program can also access its associated enabling register, to program the events which will be eligible to activate the ESB summary bit in the Status Byte.

**Measurement events:**

- When the instrument has been commanded to take a measurement; the measurement has been taken and is available to be read.
- Mathematical Overflow
- Overload
- Underrange
- Input frequency higher than the valid range
- Input frequency lower than the valid range

These events are flagged in another 8-bit latched register, called the 'Measurement Event Status Register' (MESR), which is read-accessible to the controller. The user's application program can also access its associated enabling register, to program the events which will be eligible to activate the MES summary bit in the Status Byte.

### A Note about Queues

Someof the event bits are summaries of queues of events. These are 'historical' (Last-in - Last Out) stacks to aid diagnosis of errors, and when the queue stack is full the eldest entries are discarded.

It is good practice to program the application to read the queue as soon as its summary bit is set true, particularly the error bits, otherwise the original cause of the error can be discarded as subsequent dependent errors fill up the stack.

# 4920M Status Reporting - Detail

## IEEE 488.1 Model

Provides for two major forms of status reporting:

- Device-specific commands from the controller, to generate status responses which have been previously programmed into the device to represent specific device conditions.
- Serial-polling of devices on the bus following a Service Request (the device pulling the SRQ line *true*).

  The device anticipates that the controller will conduct a serial poll of devices on the bus as a result of the SRQ. So as it issues the SRQ, it sets up a 'Status Byte' for the controller to read.

  If the controller is programmed to conduct a serial poll to identify the source of the SRQ, and subsequently read the 'Status Byte', the number represented by the byte can be interpreted as an event message. Such numbers are previously coded into the device's firmware to represent specific device conditions, and application programmers are thus able to program alarms or other actions to occur when such messages are received by the controller.

## IEEE 488.2 Model

This incorporates the two aspects of the IEEE 488.1 model into an extended structure with more definite rules. These rules invoke the use of standard 'Common' messages and provide for device-dependent messages. A feature of the structure is the use of 'Event' registers, each with its own enabling register as illustrated in Fig. 5.3.

## 4920M Model Structure

The IEEE 488.2 Standard provides for a more extensive hierarchical structure with the Status Byte at the apex, defining its bits 4, 5 and 6 and their use as summaries of a *Standard*-defined event structure which **must** be included, if the device is to claim conformance with the Standard. The 4920M employs these bits as defined in the Standard.

Bits 0, 1, 2 and 3 and 7 of the Staus Byte are made available to the device designer, to act as summaries of *device*-specific events. In the 4920M, only bit 0 is necessary in order to summarize its device-specific events.

It must be recognized by the application programmer that whenever the controller reads the Status Byte, it can only receive summaries of types of events, and further query messages are necessary to dig deeper into the detailed information relating to the events themselves.

Thus two further bytes are used to expand on the summaries at bits 0 and 5 of the Status Byte.

## Status Byte Register

In this structure the Status Byte is held in the 'Status Byte Register'; the bits being allocated as follows:

**Bit 0** (DIO1) Device-specific Measurement Event Summary Bit (MES)

This summarizes the byte held in a Device-defined 'Measurement Event Status Register' (MESR), whose bits represent reportable conditions in the device. In the 4920M these are overload, math overflow, reading available, underrange, and input frequency outside of the valid range. The MES bit is *true* when the byte in the MESR contains one or more enabled bits which are *true*; or *false* when all the enabled bits in the byte are *false*. The Measurement Event Status Register, its enabling register and byte are described later.

**Bits 1** (DIO2), **2** (DIO3) and **3** (DIO4) are not used in the 4920M status byte. They are always *false*.

**Bit 4** (DIO5) IEEE 488.2-defined Message Available Bit (MAV)

The MAV bit helps to synchronize information exchange with the controller. It is *true* when the 4920M message exchange interface is ready to accept a request from the controller to start outputting bytes from the Output Queue; or *false* when the Output Queue is empty.

The common command *CLS can clear the Output Queue, and the MAV bit 4 of the Status Byte Register; providing it is sent immediately following a 'Program Message Terminator'.

**Bit 5** (DIO6) IEEE 488.2-defined Standard Event Summary Bit (ESB)

This summarizes the state of the 'Event Status byte', held in the 'Event Status register' (ESR), whose bits represent IEEE 488.2-defined conditions in the device. The ESB bit is *true* when the byte in the ESR contains one or more enabled bits which are *true*; or *false* when all the enabled bits in the byte are *false*. The byte, the Event Status Register and its enabling register are defined by the IEEE 488.1 Standard; they are described later.

**Bit 6** (DIO7) This bit has a dual purpose:

When the controller is conducting a serial poll (as a result of receiving a Service Request via the SRQ line), the 4920M is placed into 'serial poll active state' and bit 6 is the Request Service Message (RQS bit). If the 4920M had been the device which originated the SRQ, its output control will set DIO 7 (bit 6's channel) *true*, but if not, then DIO 7 is set *false*. By reading the Status Byte *bitwise*, the controller identifies the device which originated the SRQ; and in the case of it being the 4920M, also receives any enabled summary bits to allow further investigation of the originating event.

If the controller reads the Status Byte using the common query *STB?, then bit 6 is the Master Status Summary Message (MSS bit), and is set *true* if one of the bits 0 to 4 or bit 5 is *true* (bits 1 to 3 are always *false* in the 4920M).

**Bit 7** (DIO8) is not used in the 4920M status byte. It is always *false*.

### Reading the Status Byte Register

There are two ways of reading the Status Byte register: by serial poll or by common query *STB?

#### Serial Poll

When the controller conducts a serial poll, the 4920M is placed into 'serial poll active state' by the IEEE 488.1 command SPE, and is addressed as a talker. The enabled contents of the Status Byte register are transferred in binary form into the 4920M I/O control, which sets the RQS bit 6 *true* if the 4920M had originated the preceding SRQ, or *false* if it had not. The binary values of bits 1, 2, 3 and 7 are always zero. The resulting byte is placed in binary onto the system bus on the corresponding DIO 8-1 lines. When the serial poll is disabled by the command SPD, the 4920M enters 'serial poll inactive state', and the I/O control relinquishes control of RQS bit 6 on the DIO 7 line.

#### *STB?

The common query: *STB? reads the binary number in the Status Byte register. The response is in the form of a decimal number which is the sum of the binary weighted values in the enabled bits of the register. In the 4920M, the binary-weighted values of bits 1, 2, 3 and 7 are always zero. The query *STB? is provided mainly for controllers with no serial poll capability, and for those users who are using the device interface for RS232-type communication.

## Service Request Enable Register

The SRE register is a means for the application program to select those types of events which are to cause the 4920M to originate an SRQ, by enabling individual Status Byte summary bits. The register contains a user-modifiable image of the Status Byte, whereby each *true* bit acts to enable its corresponding bit in the Status Byte.

### Bit Selector: *SRE *phs Nrf*

The program command: *SRE *phs Nrf* performs the selection, where *Nrf* is a decimal numeric, which when decoded into binary produces the required bit-pattern in the enabling byte.

For example:

> If an SRQ is required only when a Standard-defined event occurs and when a message is available in the output queue, then *Nrf* should be set to 48. The binary decode is 00110000 so bit 4 or bit 5, when *true*, will generate an SRQ; but even when bit 0 or bit 6 is *true*, no SRQ will result. The 4920M always sets the Status Byte bits 1, 2, 3 and 7 *false*, so they can never originate an SRQ whether enabled or not.

### Reading the Service Request Enable Register

The common query: *SRE? reads the binary number in the SRE register. The response is in the form of a decimal number which is the sum of the binary-weighted values in the register. The binary-weighted values of bits 1, 2, 3 and 7 are always zero.

## IEEE 488.2-defined Event Status Register

The 'Event Status Register' holds the Event Status Byte, consisting of event bits, each of which directs attention to particular information. All bits are 'sticky'; ie. once *true*, cannot return to *false* until the register is cleared. This occurs automatically when it is read by the query: *ESR?. The common command *CLS clears the Event Status Register and associated error queues, but not the Event Status Enable Register. The bits are named in mnemonic form as follows:

**Bit 0** Operation Complete (OPC)
This bit is *true* only if *OPC has been programmed *and* all selected pending operations are complete. As the 4920M operates in serial mode, its usefulness is limited to registering the completion of long operations, such as self-test.

**Bit 1** Request Control (RQC)
This bit would be *true* if the device were able to assume the role of controller, *and* is requesting that control be transferred to it from the current controller. This capability is not available in the 4920M, so bit 1 is always *false*.

**Bit 2** Query Error (QYE)
QYE *true* indicates that an attempt is being made to read data from the output queue when no output is present or pending, or data in the output queue has been lost. The Standard document defines the conditions under which a query error is generated, as a result of the controller failing to follow the message exchange protocol.

**Bit 3** Device Dependent Error (DDE)
DDE is set *true* when an internal operating fault is detected, for instance during a self test. Each reportable error has been given a listed number, which is appended to an associated queue as the error occurs. The queue is read destructively as a First In Last Out stack, using the query command DDQ? to obtain a code number. The DDE bit is not a summary of the contents of the queue, but is set or confirmed *true* concurrent with each error as it occurs; and once cleared by *ESR? will remain *false* until another error occurs. The query DDQ? can be used to read all the errors in the queue until it is empty, when the code number zero will be returned.

The common command *CLS clears the queue.

**Bit 4** Execution Error (EXE)
An execution error is generated if the received command cannot be executed, owing to the device state or the command parameter being out of bounds. Each reportable execution error has been given a listed number, which is appended to an associated queue as the error occurs. The queue is read destructively as a First In Last Out stack, using the query command EXQ?. The EXE bit is not a summary of the contents of the queue, but is asserted *true* as each error occurs; and once cleared by *ESR? will remain *false* until another error occurs. The query EXQ? can be used to read all the errors in the queue until it is empty, when the code number zero will be returned.

The common command *CLS clears the queue.

**Bit 5** Command Error (CME)
CME occurs when a received bus command does not satisfy the IEEE 488.2 generic syntax or the device command syntax programmed into the instrument interface's parser, and so is not recognized as a valid command. Command errors do not have an associated queue.

**Bit 6** User Request (URQ)
This bit is set *true* by the action of pressing the front panel SRQ key. If the URQ bit and the ESB bit are enabled, an SRQ is generated and the SRQ legend on the main display lights. During a subsequent serial poll the controller reads the Status Byte, the RQS bit in the I/O control is destroyed, and the front panel legend is extinguished. The ESB and URQ bits remain *true*, returning to *false* when the controller destructively reads the Event Status register by *ESR?, or clears status by *CLS.

**Bit 7** 4920M Power Supply On (PON)
This bit is set *true* only when the Line Power has just been switched on to the 4920M, and the subsequent Operational Selftest has been completed successfully (if unsuccessful, the DDE bit 3 is set true, generating an SRQ if its enable bit is also set true). Whether this generates an SRQ or not is dependent on the decimal numeric value previously programmed as part of the 'Power On Status Clear' message *PSC *phs Nrf*. If *Nrf* was zero, the Event Status Enable register would have been cleared at power on, so PON would not generate the ESB bit in the Status Byte register, and no SRQ would occur at power on. For an *Nrf* of 1, **and** the Event Status Enabling register bit 7 *true*, **and** the Service Request Enabling register bit 5 *true*; a change from Power Off to Power On generates an SRQ. This is only possible because the enabling register conditions are held in non-volatile memory, and restored at power on.

This facility is included to allow the application program to set up conditions so that a momentary Power Off followed by reversion to Power On (which could upset the 4920M programming) will be reported by SRQ. To achieve this, the Event Status register bit 7 must be permanently *true* (by *ESE *phs Nrf*, where $Nrf \geq 128$); the Status Byte Enable register bit 5 must be set permanently *true* (by command *SRE *phs Nrf*, where $Nrf \geq 32$); Power On Status Clear must be disabled (by *PSC *phs Nrf*, where $Nrf = 0$); and the Event Status register must be read destructively immediately following the Power On SRQ (by the common query *ESR?).

## Standard Event Status Enable Register

The ESE register is a means for the application program to select, from the positions of the bits in the standard-defined Event Status Byte, those events which when *true* will set the ESB bit *true* in the Status Byte. It contains a user-modifiable image of the standard Event Status Byte, whereby each *true* bit acts to enable its corresponding bit in the standard Event Status Byte.

**Bit Selector: *ESE *phs Nrf***
The program command: *ESE *phs Nrf* performs the selection, where *Nrf* is a decimal numeric, which when decoded into binary, produces the required bit-pattern in the enabling byte.

For example:

If the ESB bit is required to be set *true* only when an execution or device-dependent error occurs, then *Nrf* should be set to 24. The binary decode is 00011000 so bit 3 or bit 4, when *true*, will set the ESB bit *true*; but when bits 0-2, or 5-7 are *true*, the ESB bit will remain *false*.

**Reading the Standard Event Enable Register**
The common query: *ESE? reads the binary number in the ESE register. The response is in the form of a decimal number which is the sum of the binary-weighted values in the register.

## Measurement Event Status Register

In this structure a 'Measurement Event Status Register' holds the Measurement Event Status Byte, consisting of event bits, specific to the 4920M. All bits are 'sticky'; ie. once *true*, can only return to *false* when the register is cleared. This register is automatically cleared when it is read by the query: MESR?. The common command *CLS clears the Measurement Event Status Register but not the Measurement Event Status Enable Register. Each of the bits is named in mnemonic form; they are described below.

**Bit 0** Bit 0 is not allocated

**Bit 1** Frequency Lower than Valid Range (FLR)
Bit 1 is asserted *true* whenever a signal, applied to the either input for any ACV or WBV measurement, has a frequency lower than 10Hz. The value recalled by the query: RDG? is ±200.0000E+33.

**Bit 2** Frequency Higher than Valid Range (FHR)
Bit 1 is asserted *true* whenever a signal, applied to the ACV input for any ACV measurement, has a frequency higher than 1.1MHz; or if applied to the WBV input for any WBV measurement, has a frequency higher than 22MHz. The value recalled by the query: RDG? is ±200.0000E+33.

**Bit 3** Underrange (U-R)
Bit 3 is asserted *true* whenever a signal, applied to the analog input for any measurement, has a value less than 2% of the selected range; or if on Auto, has a value less than the lowest autorange. The value recalled by the query: RDG? is ±200.0000E+33.

**Bit 4** Overload (O-L)
Bit 4 is asserted *true* whenever a signal, applied to either input for any measurement, has exceeded the selected range; or if on Auto in ACV function, has exceeded the highest autorange. The value recalled by the query: RDG? is ±200.0000E+33.

**Bit 5** Mathematical Overflow (MOF)
Bit 5 is asserted *true* whenever the modulus of the result of an internal math calculation has a value which is too large to be represented.

A divide-by-zero command will automatically be rejected as an execution error, but a very large number could result from trying to divide by (say) a reading which is very close to zero.

**Bit 6** Bit 6 is not allocated.

**Bit 7** Reading Available (RAV).
Bit 7 is asserted *true* whenever the result of a reading is available (when the A-D cycle is completed). If command RDG? is sent, the result will be placed in the output queue.

## Measurement Event Status Enable Register

The application program uses the MESE register to select, from the positions of the bits in the Measurement Event Status Byte, those events which when *true* will assert the MES bit *true* in the Status Byte. It contains a user-modifiable image of the Measurement Event Status Byte, whereby each *true* bit acts to enable its corresponding bit in the Measurement Event Status Byte.

### Bit Selector: MESE *phs Nrf*

The program command: MESE *phs Nrf* performs the selection, where *Nrf* is a decimal numeric, which when decoded into binary, produces the required bit-pattern in the enabling byte.

For example:

If the MES bit is required to be asserted *true* only when the frequency is higher or lower than its valid range, then the value of *Nrf* should be set to 6. The binary decode is 00000110 so bit 1 or bit 2, when *true*, will assert the MES bit *true*; but when bits 0 or 3-6 are *true*, the MES bit will not be asserted.

### Reading the Measurement Event Status Enable Register

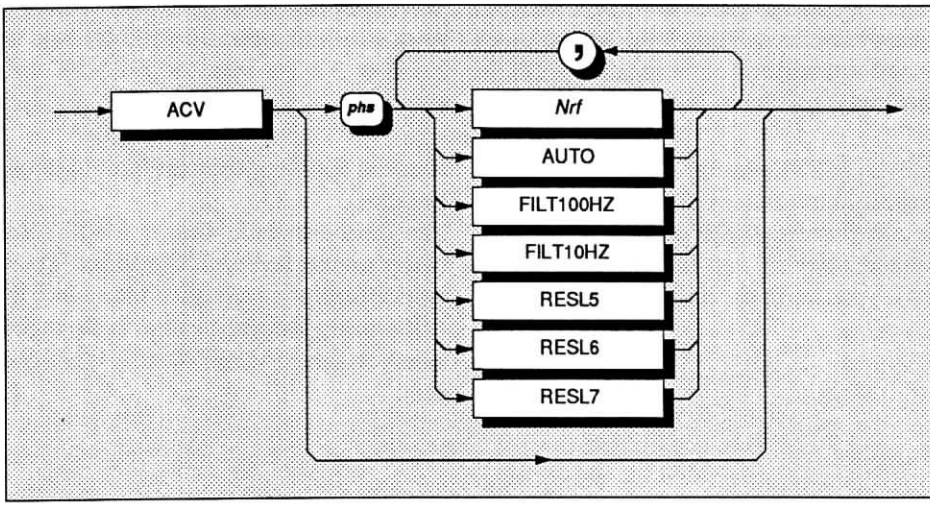
The device-specific query: MESE? reads the binary number in the MESE register. The response is in the form of a decimal number which is the sum of the binary-weighted values in the register. The binary-weighted values of bits 0 and 6 are always zero.

# 4920M COMMANDS AND QUERIES - Syntax Diagrams

## MAJOR FUNCTIONS

### High Accuracy AC Voltage

The following commands are used to select ACV function along with its associated configuration.




***Nrf*** is a decimal numeric value.
It is meant to represent the expected signal amplitude, so that the instrument will go to the most relevant range. For example, an Nrf of 6, 10, or even 9.8765, will select the 10V range. Any valid numeric value cancels autorange.

Note that numbers exceeding the defined data element resolution of 7 digits will be rounded to that resolution.

For a full list of significant ACV range points, see Appendix C to Section 4. The Nrf limits for nominal range selection are shown in the following list, together with the range input limits:

| | |
|---|---|
| 0 < *Nrf* ≤ 0.3 | selects the 0.3V range. (0.027V to 0.34995V) |
| 0.3 < *Nrf* ≤ 1.0 | selects the 1V range. (0.09V to 1.1995V) |
| 1.0 < *Nrf* ≤ 3.0 | selects the 3V range. (0.27V to 3.4995V) |
| 3.0 < *Nrf* ≤ 10.0 | selects the 10V range. (0.9V to 11.995V) |
| 10.0 < *Nrf* ≤ 30.0 | selects the 30V range. (2.7V to 34.995V) |
| 30.0 < *Nrf* ≤ 100.0 | selects the 100V range (9V to 119.95V) |
| 100.0 < *Nrf* ≤ 300.0 | selects the 300V range (27V to 349.95V) |
| 300.0 < *Nrf* | selects the 1000V range. (90V to 1,199.5V) |

**AUTO** selects the autorange facility:

***Autoranging up***

A measured signal whose value exceeds the auto-selected range upper limit (see list on page 5-28) will activate the next higher range and trigger a new measurement. If it exceeds the new range upper limit, the process continues until the signal value is in range.

If the signal exceeds the upper limit on the highest (1000V) range then 'error overload' appears on the front panel. The relevant query command invokes the 'invalid number response', and the appropriate bit is set in the device status registers.

***Autoranging down***

For signals equal to or smaller than the full range of the next lower range, the measured value determines the new range, which is selected, then a new measurement is triggered.

**FILT_100HZ or FILT_10HZ**

inserts the appropriateanalog filter into the signal path. One of these filters is always in circuit.

**RESLX** where X is in the range 5 to 7:

sets the resolution of the measurement in the corresponding range 5 to 7 digits, where for the whole of each range:

RESL5 guarantees 100ppm resolution;
RESL6 guarantees 10ppm resolution;
RESL7 guarantees 1ppm resolution.

Example:

ACV 3,FILT10,RESL7 would program the instrument to the ACV 3V range with 10Hz filter selected and a resolution of 1ppm.

**Recall of RMS Value and Frequency**

For each RMS reading trigger, a measurement of signal frequency (with selectable frequency resolution) is also triggered. For recall of these two parameters refer to RDG? and FREQ? commands.

**Execution Errors**

None.

**Reversion from Remote to Local**

No Change.

**Exit from ACV Function**

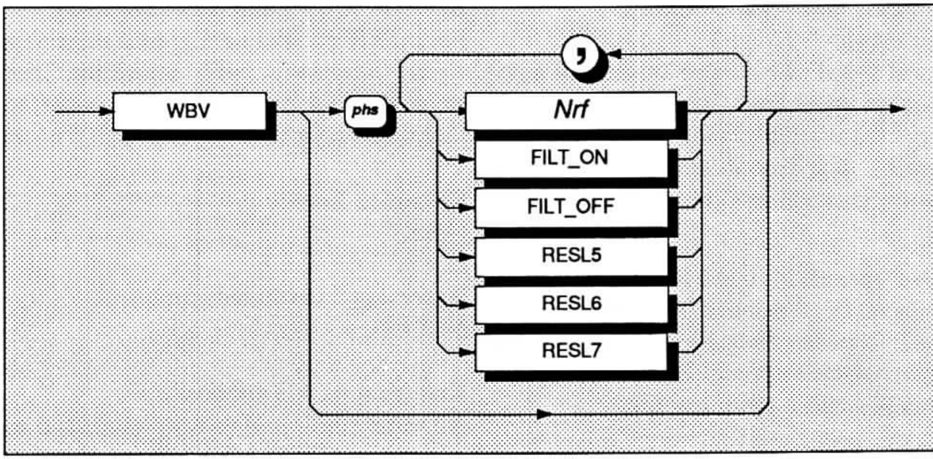
All parameters saved on exit; restored on re-entry.

**Power On and Reset Conditions**

| | |
|---|---|
| ACV | Selected active. |
| Autorange | Off |
| Range | 1000 |
| Analog Filter | FILT100HZ |
| Resolution | RESL7 (max) |

## Wide Band Voltage

The following commands are used to select WBV function along with its associated configuration.




***Nrf*** is a decimal numeric value which is meant to represent the expected signal amplitude, so that the instrument will go to the most relevant range.

If Nrf is 0.3, 1, or even 0.156789, then the 1V range is automatically selected.

| | |
|---|---|
| *Nrf* ≤ 1.0 | selects the 1V range. (0.0795V to 1.1995V) |
| *Nrf* >1.0 | selects the 3V range. (0.3000V to 3.5995V) |

Note that numbers exceeding a resolution of 7 digits will be rounded to that resolution.

**Autorange** is **not** provided for WBV function.

For a full list of significant WBV range points, see Appendix C to Section 4.

**FILT_ON:**
inserts a 10Hz analog integration filter into the signal path, on the 1V range only.

**FILT_OFF:**
removes the analog high-pass filter from the signal path, on the 1V range only.

**3V Range:**
The filter is not needed on the 3V range (because of the wide frequency span of the detector) and it cannot be inserted. As can be seen from the syntax diagram, FILT_ON and FILT_OFF can be *commanded* for the 3V range. However, insertion of the filter is automatically excluded by the analog configuration of the range.

**RESLX** where X is in the range 5 to 7:
sets the resolution of the measurement in the corresponding range 5 to 7 digits, where for the whole of each range:

RESL5 guarantees 100ppm resolution;
RESL6 guarantees 10ppm resolution;
RESL7 guarantees 1ppm resolution.

Example:
WBV 1,FILT_ON,RESL7 would program the instrument to the WBV 1V range with 10Hz filter selected and a resolution of 1ppm.

**Recall of RMS Value and Frequency**
For each RMS reading trigger, a measurement of signal frequency (with selectable frequency resolution) is also triggered. For recall of these two parameters refer to RDG? and FREQ? commands.

**Execution Errors**
None.

**Reversion from Remote to Local**
No Change.

**Exit from ACV Function**
All parameters saved on exit; restored on re-entry.

**Power On and Reset Conditions**

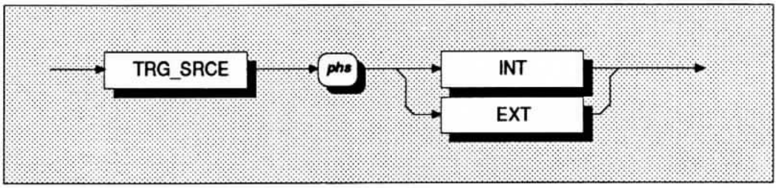
| | |
|---|---|
| Range | 3V |
| Analog Filter | FILT_OFF |
| Resolution | RESL7 (max.) |

WBV not active (ACV active).

# Triggers and Readings Operations

## Trigger Source Selector

**Caution:**
The use of internal triggers can produce unexpected results, due to the time required for the A-D conversion, and the A-D triggers being unsynchronized with the IEEE 488 bus operations. Such triggers should be avoided unless they form an essential ingredient of the required measurement.




**TRG_SRCE INT**
generates internal triggers within 0.5 second of completion of the previous reading. External trigger sources are disabled.

**TRG_SRCE EXT**
disables internal triggers and enables controller-generated external trigger sources. These are:

- IEEE 488.1: GET command.
- IEEE 488.2: *TRG command.

Both selections are mutually exclusive.

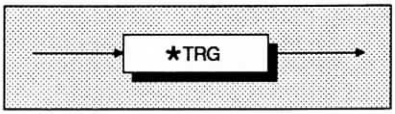
**Execution Errors:**
None.

**Reversion from Remote to Local**
No Change.

**Power On and Reset Conditions**
The default condition is TRG_SRCE INT.

## Execute Trigger

This command conforms to the IEEE 488.2 standard requirements.




***TRG**
is equivalent to a Group Execute Trigger (GET), and will cause a single reading to be taken.

**Execution Errors:**
None

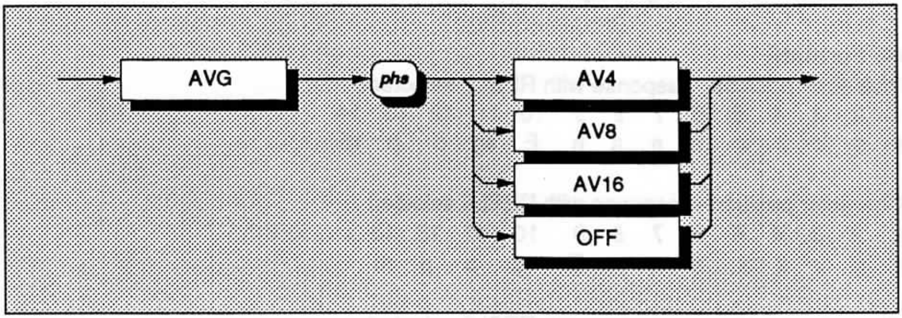
**Reversion from Remote to Local**
Not applicable.

**Power On and Reset Conditions**
Not applicable.

Standard signal acquisition times are given in Section 4, Appendix B; for Function and Filter selections.

## Averaging

**Rolling Average:** processes successive readings to provide a measurement which is the arithmetic mean of the most recent 'R' (4, 8 or 16) readings, Once the window is full with the selected number of readings, the earliest reading is discarded as each new reading is added. The mean is updated with every new reading.




**AVG OFF** cancels averaging.
**AVG AV4** selects 4 readings.
**AVG AV8** selects 8 readings.
**AVG AV16** selects 16 readings.

Note: From a cleared average store the average is the mean of the number of readings to date, until the selection number window is reached. The average stores are cleared on each command update.

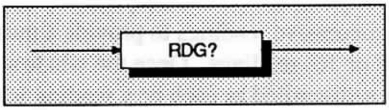
**Execution Errors:**
None.

**Reversion from Remote to Local**
No Change.

**Power On and Reset Conditions**
The default condition is AVG OFF.

# Reading Recall

## Voltage Readings




**RDG?** recalls the most recently completed measurement made by the instrument.

**Response Format:**

Character position - response with RESL7 selected

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
| s | n | x | x | x | n | n | n | n | E | sg | p | p | nl |

Character position - response with RESL6 selected

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |
| s | n | x | x | x | n | n | n | E | sg | p | p | nl |

Character position - response with RESL5 selected

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| s | n | x | x | x | n | n | E | sg | p | p | nl |

**Where:**

s = + or - or space
n = 0 to 9
x = either n or decimal point (.)
E = ASCII character identifying the exponent
sg = + or -
p = 0 to 9 (exponent is in engineering units)
nl = newline with EOI

**Response Decode:**
If no signal has been received to generate a conversion of the input signal, then the response to this command will represent the most-recent measurement. If no triggers are available, the invalid response is given. If a trigger has already been received, this query will wait for the completion of the measurement and place its result in the output queue.

**Returned Value:**
The value represents the applied signal together with any mathematical modifications selected with the Math facility. Overload or underscale is represented by a value of +200E+33 along with a set flag bit in the measurement qualifying byte of the status data.

**Execution Errors:**
None

**Power On and Reset Conditions**
All previous results are cleared at Power On and Reset, thus an overload response is given until after the first trigger.

## Frequency Readings


**FREQ?** recalls the frequency associated with the most-recently completed measurement.

**Response Format:**

Character position

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| s | n | x | x | x | n | n | n | n | E | sg | p | p | nl |

**Where:**

s = + or - or space
n = 0 to 9
x = either n or decimal point (.)
E = ASCII character identifying the exponent
sg = + or -
p = 0 to 9 (exponent is in engineering units)
nl = newline with EOI

A value of +200E+33 is returned if the measurement circuits cannot produce a result.

If no signal has been received to generate a conversion of the input signal, then the response to this command will be the frequency of the most-recent measurement. If no triggers are available, the invalid response is given. If a trigger has already been received, this query will wait for the completion of the measurement and place its result in the output queue.

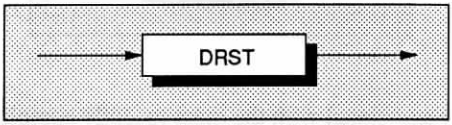
**Execution Errors:**
None

**Power On and Reset Conditions**
All previous results are cleared at Power On and Reset, thus an invalid response is given until after the first trigger.

## Difference and Deviation Calculations

The 4920M is able to make a variety of comparisons between a current reading and one which was previously stored as a reference. The following codes are used to command the processes:

### Reference Reading




**DRST** stores the most recently triggered reading as reference.

**NB.** No response is given to this command; but if the reading is overload, underscale, or does not exist when the DRST command is sent, then any subsequent DIF?/DPC?/DPM? query will return +200E+33.

**Execution Errors:**
None

**Power On and Reset Conditions**
All previous results are cleared at Power On and Reset, thus no reading will exist until after the first trigger.

### Difference between the Most-Recent Reading and the Reference Reading


**DIF?** returns the voltage difference between the most-recently triggered reading and the reference reading, in the following sense:

(Most-Recent Reading - Reference Reading)

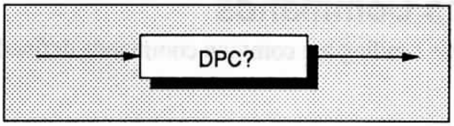
**Response**
The response is given in the same format as for RDG?.

**N.B.** If either reading is overload, underscale, or does not exist; then DIF? returns +200E+33.

**Execution Errors:**
None

**Power On and Reset Conditions**
All previous results are cleared at Power On and Reset, thus no reading will exist until after the first trigger.

## Deviation of the Most-Recent Reading from the Reference Reading (%)




**DPC?** returns the voltage deviation, in percentage, of the most-recently triggered reading from the reference reading, in the following sense:

$$\frac{100 \times (\text{Most-Recent Reading} - \text{Reference Reading})}{\text{Reference Reading}}$$

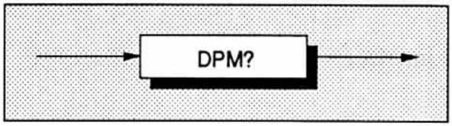
**Response**
The response is given in the same format as for RDG?.

**N.B.** If either reading is overload, underscale, or does not exist; then DIF? returns +200E+33.

**Execution Errors:**
None

**Power On and Reset Conditions**
All previous results are cleared at Power On and Reset, thus no reading will exist until after the first trigger.

## Deviation of the Most-Recent Reading from the Reference Reading (ppm)




**DPM?** returns the voltage deviation, in parts-per-million, of the most-recently triggered reading from the reference reading, in the following sense:

$$\frac{10^6 \times (\text{Most-Recent Reading} - \text{Reference Reading})}{\text{Reference Reading}}$$

**Response**
The response is given in the same format as for RDG?.

**N.B.** If either reading is overload, underscale, or does not exist; then DIF? returns +200E+33.

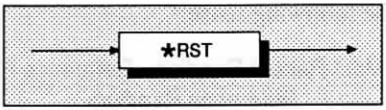
**Execution Errors:**
None

**Power On and Reset Conditions**
All previous results are cleared at Power On and Reset, thus no reading will exist until after the first trigger.

# Internal Operations Commands

All of the commands under this heading are common commands defined in the IEEE-488.2 standard.

### Reset




***RST**
will reset the instrument to a defined condition, stated for each applicable command with the command's description, and listed in Appendix B to this section.

The reset condition is independent of past-use history of the instrument except as noted below:

*RST does not affect the following:

- the selected address of the instrument;
- calibration data that affect specifications;
- SRQ mask conditions;
- the state of the IEEE 488.1 interface;
- stored math constants.

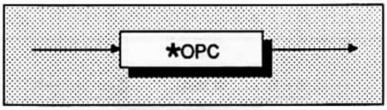
The action of the front panel Reset key is **not** equivalent to *RST, but is a subset of it.

**Execution Errors:**
None.

**Power On and Reset Conditions**
Not applicable.

## Operation Complete

This command conforms to the IEEE 488.2 standard requirements.




***OPC**

is a synchronization command which will generate an operation complete message in the standard Event Status Register when all pending operations are complete.

**Execution Errors:**

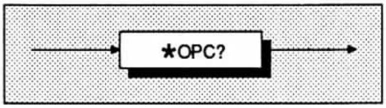
None.

**Power On and Reset Conditions**

Not applicable.

## Operation Complete?

This command conforms to the IEEE 488.2 standard requirements.




**Response Format:**

Character position
1 2
n nl

**Where:**

n = 1
nl = newline with EOI

**Response Decode:**

The value returned is always 1, which is placed in the output queue when all pending operations are complete.

**Wait**

This command conforms to the IEEE 488.2 standard requirements.


***WAI**

prevents the instrument from executing any further commands or queries until the *No Pending Operations Flag* is set true. This is a mandatory command for IEEE-488.2 but has no relevance to this instrument as there are no parallel processes requiring Pending Operation Flags.

**Execution Errors:**

None.

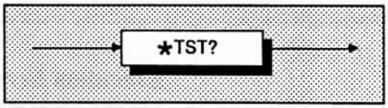
**Power On and Reset Conditions**

Not applicable.

# Test Operations

## Operational Selftest

This command conforms to the IEEE 488.2 standard requirements.




***TST?**
executes an operational selftest. A response is generated after the test is completed.

**N.B.** Operational selftest is valid only at temperatures: 23°C±10°C.

**Response Format:**

Character position
1 2
n nl

**Where:**

n = 0 or 1
nl = newline with EOI

**Response Decode:**
The value returned identifies pass or failure of the operational selftest:

**ZERO** indicates operational selftest complete with no errors detected.

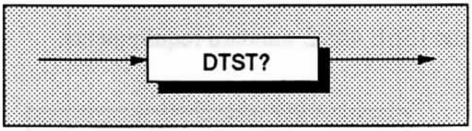
**ONE** indicates operational selftest has failed. The errors can be found in the device dependent error queue.

**Execution Errors:**
Operational selftest is not permitted when calibration is successfully enabled.

**Reversion from Remote to Local**
Not applicable.

**Power On and Reset Conditions**
Not applicable.

## Diagnostic Selftest




**DTST?**
initiates a diagnostic selftest. A response is generated after the test is completed.

**N.B.** Operational selftest is valid only at temperatures: 23°C±10°C.

**Response Format:**

Character position

1 2

n nl

**Where:**

n = 0 or 1

nl = newline with EOI

**Response Decode:**
The value returned identifies pass or failure of the diagnostic test:

**ZERO** indicates test complete with no errors detected.

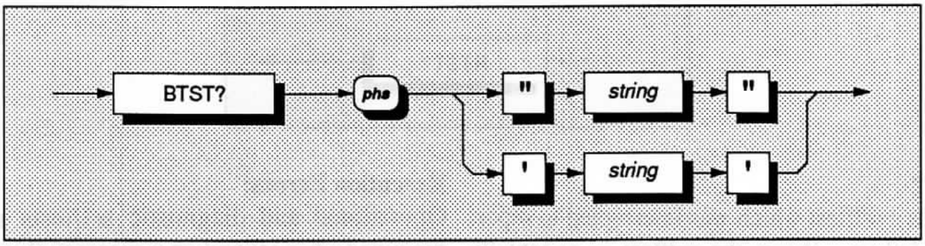
**ONE** indicates diagnostic selftest has failed. The errors can be found in the device dependent error queue.

**Execution Errors:**
Diagnostic test is not permitted when calibration is successfully enabled.

**Reversion from Remote to Local**
Not applicable.

**Power On and Reset Conditions**
Not applicable.

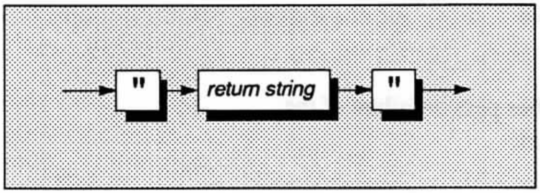
## Bus Test




**BTST?**
is used to test the bus drivers. It is a <STRING PROGRAM DATA> element as defined in the standard document. A response is generated when the 4920M is addressed to talk following this command.

*string* is a <string data item>, consisting of up to 128 ASCII characters which must be enclosed in single or double quotes as shown in the syntax diagram. Any embedded quote character in the string, of the same type as the enclosing quotes, must be doubled (repeated) to distinguish it from them. All extra quotes will be stripped from the string upon receipt by the 4920M.

**Response Syntax:**




The response is a <STRING REPONSE DATA> element as defined in the standard document.

*return string* is the same as the transmitted *string* data item, stripped of any extra quotes; but as it is to be returned enclosed in double quotes, any embedded double-quote character will be repeated to distinguish it from the enclosing quotes.

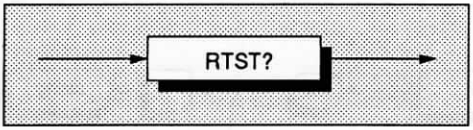
**Response Decode:**
After correctly processing embedded quote characters, the returned data can be compared with the transmitted data. The two should be identical.

**Execution Errors:**
None

**Reversion from Remote to Local**
Not applicable.

**Power On and Reset Conditions**
Not applicable.

## Resume Selftest




**RTST?**
resumes a selftest which was interrupted to report a test failure. When the selftest is eventually completed, a response is given.

**Response Format:**

Character position

| 1 | 2 |
|---|---|
| n | nl |

**Where:**

n = 0 or 1
nl = newline with EOI

**Response Decode:**
The value returned identifies pass or failure of the selftest which was resumed:

**ZERO** indicates test complete with no errors detected.

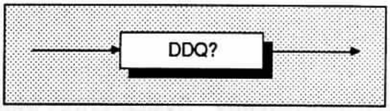
**ONE** indicates the resumed selftest has failed. The errors can be found in the device dependent error queue.

**Execution Errors:**
Operational and diagnostic selftests are not permitted when calibration is successfully enabled.

**Reversion from Remote to Local**
Not applicable.

**Power On and Reset Conditions**
Not applicable.

## Recall Device Errors




**DDQ?**
recalls the last error from the queue of device dependent errors (e.g errors recorded during a failed operational or diagnosticselftest). The queue is organized as a last-in - first-out stack, its individual entries being destructively read. If there are no entries in the queue, then use of this command generates a result of Ø.

**Read the Queue until Empty**
It is good practice to read the queue until empty on each occurrence of device-dependent error, to prevent unrelated history of errors being retained.

**Response Format:**
Character position
1 2 3 4 5
n n n n nl

**Where:**
n = 0 to 9
nl = newline with EOI

**Response Decode:**
The value returned is a specified integer value indicating the fault. Refer to 'Error Detection' overleaf, and for the meanings of specific codes to Appendix A of Section 4.

**Execution Errors:**
None.

**Reversion from Remote to Local**
Not applicable.

**Power On and Reset Conditions**
Not applicable.

## Error Detection

All errors which cannot be recovered transparently result in some system action to inform the user via a message, and where possible restore the system to an operational condition. Errors are classified by the method with which they are handled.

Recoverable errors report the error and continue.

System errors which cannot be recovered cause the system to halt with a message displayed.

In this case, restarting the system from power on may clear the error, but generally such messages are caused by hardware or software faults.

### Device-Dependent Errors (DDE)

A Device-Dependent Error is generated if the device detects an internal operating fault (eg. during self-test). The DDE bit (3) is set *true* in the Standard-defined Event Status Byte, and the error code number is appended to the Device-Dependent Error queue.

In both Remote and Local operation, a device-dependent error is reported by the mechanisms described in the sub-section which deals with status reporting, and the queue entries can be read destructively as LIFO by the query DDQ?. An error message appears on the Menu display.

The Remote user can ignore the queue, but it is good practice to read the errors as they occur.

The queue cannot be read in local operation, but the local user can continue by pressing any primary menu key. The error is, however, added to the queue so that it can be retrieved in remote operation using DDQ? When entering remote operation (unless it is intended to investigate errors generated in local) it is therefore advisable to empty old errors from the queue using the common command *CLS.

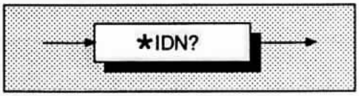
The code numbers for device dependent errors, with their associated descriptions, are given in Appendix A to Section 4. Each code number refers to a particular stage in a selftest, indicating failure of an individual test. Section 2 in the Servicing Handbook provides diagnostic information related to the test whose code number has been generated.

# Routine Operational Queries

N.B. Other program queries are listed with their companion program commands (see index)

## I/D (Instrument Identification)

This command conforms to the IEEE 488.2 standard requirements.




***IDN?**

will recall the instrument's manufacturer, model number, serial number and firmware level.

**Response Format:**

Character position

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| D | A | T | R | O | N | | W | A | V | E | T | E | K | , |

| | | | | | |
|---|---|---|---|---|---|
| 16 | 17 | 18 | 19 | 20 | 21 |
| 4 | 9 | 2 | 0 | M | , |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 |
| 4 | 5 | 6 | 7 | 8 | 9 | - | 0 | 1 | . | 0 | 9 | , |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 |
| 4 | 0 | 0 | 9 | 3 | 5 | / | X | X | . | X | X |

**Where:**

The data contained in the response consists of four comma-separated fields, the last two of which are instrument-dependent. The data element type is defined in the IEEE 488.2 standard specification.

A single query sent as a terminated program message will elicit a single response terminated by:

nl = newline with EOI

If multiple queries are sent as a string of program message units (separated by semi-colons with the string followed by a permitted terminator), then the responses will be returned as a similar string whose sequence corresponds to the sequence of the program queries. The final response in the string will be followed by the terminator:

nl = newline with EOI

**Response Decode:**

The data contained in the four fields is organized as follows:

- First field - manufacturer
- Second field - model
- Third field - serial number - can be altered via a calibration operation - see page 5-70.
- Fourth field - firmware level (will possibly vary from one instrument to another).

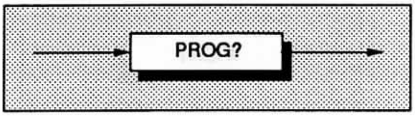
**Execution Errors:**

None.

**Power On and Reset Conditions**

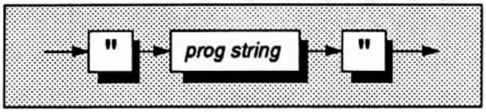
Not applicable.

## Recall Instrument Settings




**PROG?**
**will recall the instrument's function and range; resolution, filter setting, averaging mode and trigger source.**

**IEEE 488.2 String Response Data Syntax:**




Where *prog string* describes the presently-programmed state of the instrument. It can be defined as a series of character groups representing the program headers and program data elements which would normally be used to set the instrument into its present state.

These groups convey information in the order:

Function (ACV or WBV),
Range,
Resolution(5, 6 or 7),
Filter (100Hz, 10Hz, On or Off),
Averaging (On or Off),
Trigger Source (Int or Ext)'.

This is illustrated by the following examples:

"ACV 1000,RESL6,FILT100HZ, OFF,EXT"

"ACV 10,RESL7,FILT10HZ, AV8,INT"

"WBV 1,RESL6,FILT_OFF, OFF,EXT"

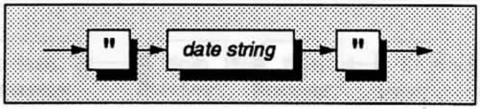
**Execution Errors:**
None.

**Power On and Reset Conditions**
Not applicable. This query recalls the instrument state at the time of asking.

## Recall Today's Date


**DATE?**
will return the present date and the time that the query was processed.

**Response Syntax:**




**Response Format:**

Character position

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| " | m | m | d | d | y | y | h | h | M | M | " |

**Where:**

m m = two digits representing *month*
d d = two digits representing *day of month*
y y = two digits representing *year*
h h = two digits representing *hour*
M M = two digits representing *minute*

A single query sent as a terminated program message will elicit a single response terminated by:

nl = newline with EOI

If multiple queries are sent as a string of program message units (separated by semi-colons with the string followed by a permitted terminator), then the responses will be sent as a similar string whose sequence corresponds to the sequence of the program queries. The final response in the string will be followed by the terminator:

nl = newline with EOI

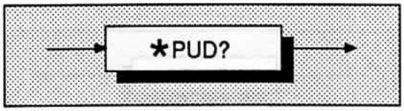
**Response Sources and Decode**
The returned date is derived from the date most-recently entered when in calibration mode: either as a parameter of the DATE command (page 5-69), or from the front panel. This is modified by the internal clock to give today's date and time.

**Execution Errors:**
None

**Power On and Reset Conditions**
No Change. Today's date is calculated from data saved in non-volatile memory.

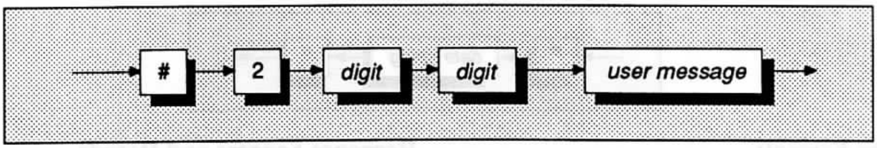
## Recall of User Data

This common command conforms to the IEEE 488.2 standard requirements.




***PUD?** recalls previously entered user data. Refer to program command ***PUD**, page 5-74.

**Response Syntax:**




**where:**

*digit* = one of the ASCII-coded numerals previously entered,

*user message* = the saved user message.

**Response Decode:**

The previously-saved message is recalled.

If no message is available, the value of the two digits is 00. The data area contains 63 bytes of data.

A single query sent as a terminated program message will elicit a single response terminated by:

nl = newline with EOI

If multiple queries are sent as a string of program message units (separated by semi-colons with the string followed by a permitted terminator), then the responses will be sent as a similar string whose sequence corresponds to the sequence of the program queries. The final response in the string will be followed by the terminator:

nl = newline with EOI

**Execution Errors:**

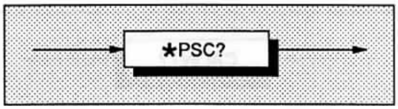
None.

**Power On and Reset Conditions**

Data area remains unchanged.

## Recall Power On Status Clear Flag

This common command conforms to the IEEE 488.2 standard requirements. The flag condition is determined by the *PSC command (page 5-62).




***PSC?**
will recall the Power On Status condition.

**Response Format:**
A single ASCII character is returned.

A single query sent as a terminated program message will elicit a single response terminated by:

nl = newline with EOI

If multiple queries are sent as a string of program message units (separated by semi-colons with the string followed by a permitted terminator), then the responses will be sent as a similar string whose sequence corresponds to the sequence of the program queries. The final response in the string will be followed by the terminator:

nl = newline with EOI

**Response Decode:**
The value returned identifies the state of the saved flag:

**Zero** indicates **false**. The instrument **is not** programmed to clear the Standard Event Status Enable Register and Service Request Enable Register at Power On, so the instrument will generate a 'Power On' SRQ.

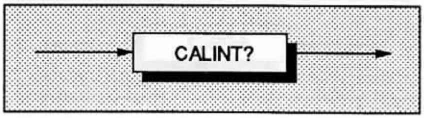
**One** indicates **true**. The instrument **is** programmed to clear the Standard Event Status Enable Register and Service Request Enable Register at Power On, so the instrument cannot generate a 'Power On' SRQ.

**Execution Errors:**
None

**Power On and Reset Conditions**
No Change. This data is saved in non-volatile memory at Power Off, for use at Power On.

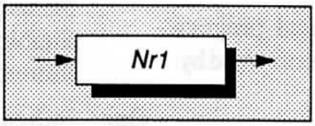
## Query Calibration Interval

This facility returns the user-entered time interval between recalibrations of the instrument.




CALINT?
returns the instrument's calibration interval; entered previously from the front panel, or remotely as a parameter of **CALINT**.

**Response Syntax**




**Response Format:**
Nr1 is an integer.

**Response Source**
The value of Nr1 returned is the current number of days entered from the front panel, or remotely as a parameter of CALINT.

A single query sent as a terminated program message will elicit a single response terminated by:

nl = newline with EOI

If multiple queries are sent as a string of program message units (separated by semi-colons with the string followed by a permitted terminator), then the responses will be sent as a similar string whose sequence corresponds to the sequence of the program queries. The final response in the string will be followed by the terminator:

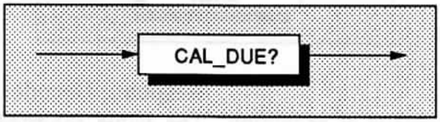
nl = newline with EOI

**Execution Errors:**
None

**Power On and Reset Conditions**
No Change. The Calibration Interval is saved in non-volatile memory.

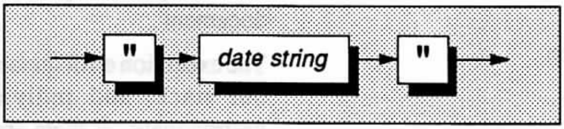
## Query Calibration Due Date

This facility returns the user-recommended date for the next recalibration of the instrument.




CAL_DUE?
returns the relevant due date, calculated from Last Calibration Date and Calibration Interval.

**Response Syntax**




**Response Format:**

Character position

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|
| " | m | m | d | d | y | y | " |

**Where:**

m m = two digits representing *month*
d d = two digits representing *day*
y y = two digits representing *year*

A single query sent as a terminated program message will elicit a single response terminated by:

nl = newline with EOI

If multiple queries are sent as a string of program message units (separated by semi-colons with the string followed by a permitted terminator), then the responses will be sent as a similar string whose sequence corresponds to the sequence of the program queries. The final response in the string will be followed by the terminator:

nl = newline with EOI

**Response Sources**

The Last Calibration Date is the date most-recently entered either by use of the command CALSEAL (page 5-68), or when calibration mode was exited from the front panel. The Calibration Interval is the current number of days entered from the front panel, or remotely as a parameter of CALINT (page 5-71).

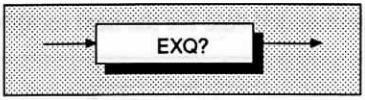
**Execution Errors:**

None

**Power On and Reset Conditions**

No Change. The due date is calculated from data saved in non-volatile memory.

## Recall Execution Errors




**EXQ?**
recalls the last error from the queue of execution errors. An execution error occurs when a command cannot be complied with (e.g calling up FILT10HZ {ACV only} when in WBV function).

**Response Format:**

Character position

| 1 | 2 | 3 | 4 |
|---|---|---|---|
| n | n | n | n |

**Where:**

n = 0 to 9

A single query sent as a terminated program message will elicit a single response terminated by:

nl = newline with EOI

If multiple queries are sent as a string of program message units (separated by semi-colons with the string followed by a permitted terminator), then the responses will be sent as a similar string whose sequence corresponds to the sequence of the program queries. The final response in the string will be followed by the terminator:

nl = newline with EOI

**Response Decode:**
The value returned is a specified integer value indicating the fault. For details of the number/fault relationship refer to Appendix A to Section 4 of this handbook. Execution Errors are reported in the form required by the IEEE 488.2 standard document .

The execution error queue operates as a last in - first out stack, and individual entries are read destructively. If there are no entries in the queue, then use of this command produces a result of zero.

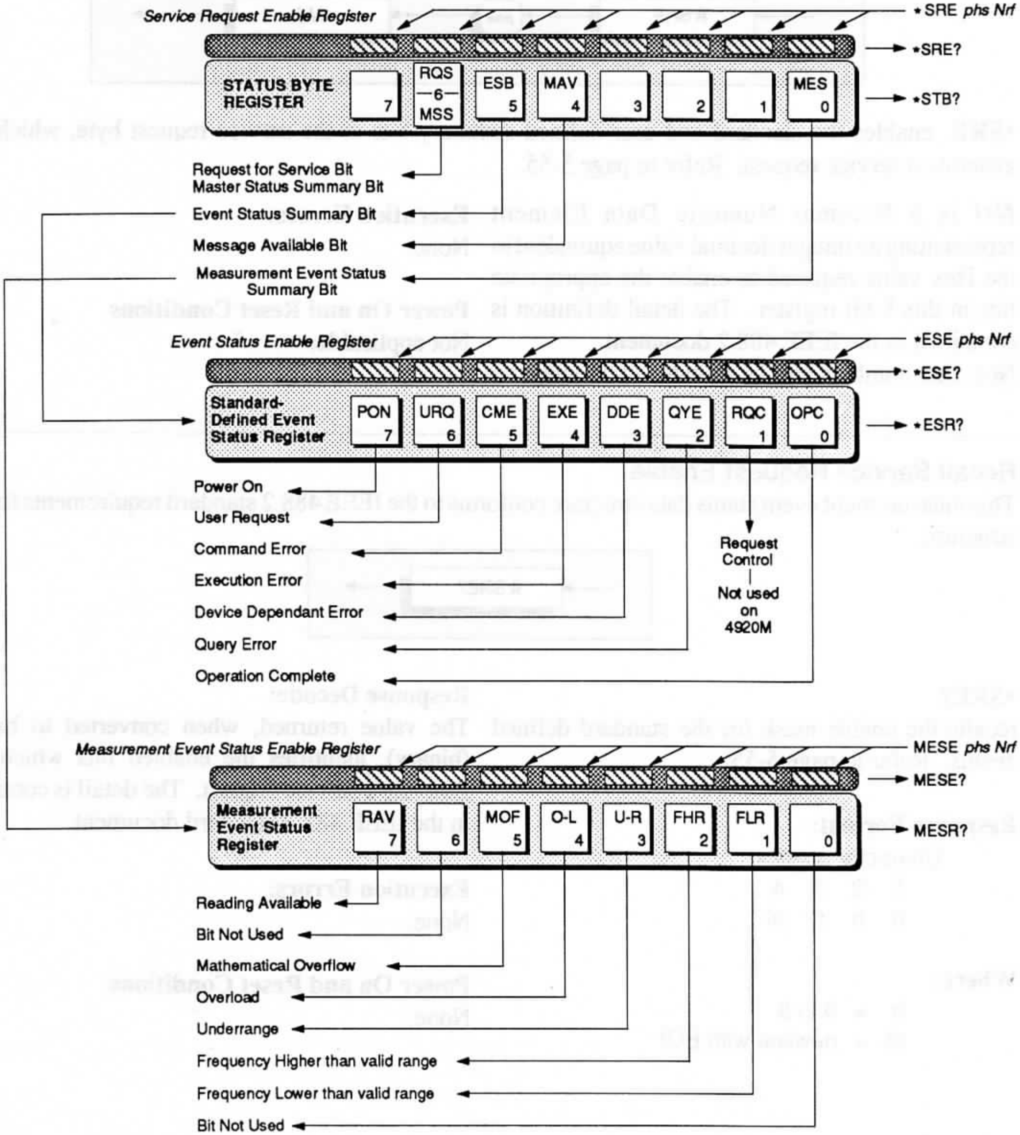
**Read the Queue until Empty**
It is good practice to read the queue until empty on each occurrence of execution error, to prevent retention of an unrelated history of errors.

**Execution Errors:**
None

**Power On and Reset Conditions**
The queue is cleared.

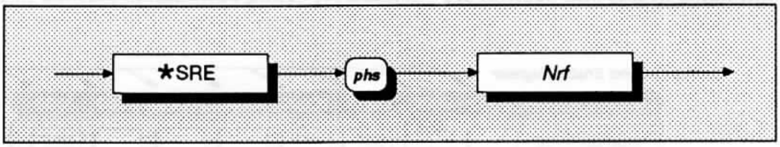
# Status Reporting

Most of the commands in this sub-section are standard reporting commands defined in the IEEE-488.2 standard.



## Service Request Enable

This measurement event status data structure conforms to the IEEE 488.2 standard requirements for this structure.




***SRE** enables the standard and user-defined summary bits in the service request byte, which will generate a service request. Refer to page 5-55.

***Nrf*** is a Decimal Numeric Data Element representing an integer decimal value equivalent to the Hex value required to enable the appropriate bits in this 8-bit register. The detail definition is contained in the IEEE 488.2 document.
Note that numbers will be rounded to an integer.

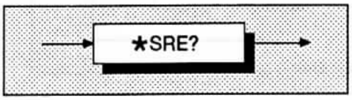
**Execution Errors:**
None.

**Power On and Reset Conditions**
Not applicable.

---

## Recall Service Request Enable

This measurement event status data structure conforms to the IEEE 488.2 standard requirements for this structure.




***SRE?**
recalls the enable mask for the standard defined events. Refer to page 5-55.

**Response Format:**

Character position

| | | | |
|---|---|---|---|
| 1 | 2 | 3 | 4 |
| n | n | n | nl |

**Where:**

n = 0 to 9
nl = newline with EOI

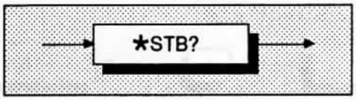
**Response Decode:**
The value returned, when converted to base 2 (binary), identifies the enabled bits which will generate a service request. The detail is contained in the IEEE 488.2 standard document.

**Execution Errors:**
None.

**Power On and Reset Conditions**
None.

## Read Service Request Register

This measurement event status data structure conforms to the IEEE 488.2 standard requirements for this structure.




***STB?**

recalls the service request register for summary bits. Refer to page 5-55.

**Response Format:**

Character position

| | | | |
|---|---|---|---|
| 1 | 2 | 3 | 4 |
| n | n | n | nl |

**Where:**

n = 0 to 9

nl = newline with EOI

**Response Decode:**

The value returned, when converted to base 2 (binary), identifies the summary bits for the current status of the data structures involved. For the detail definition see the IEEE 488.2 standard document. There is no method of clearing this byte directly. Its condition relies on the clearing of the overlying status data structure.

**Execution Errors:**

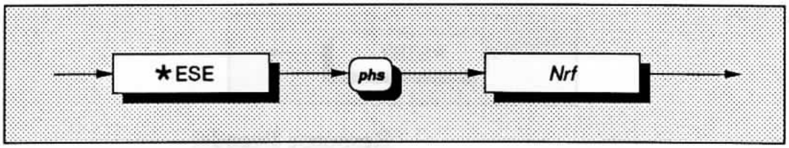
None.

**Power On and Reset Conditions**

Not applicable.

## Event Status Enable

This event status data structure conforms to the IEEE 488.2 standard requirements for this structure.




***ESE** enables the standard defined event bits which will generate a summary message in the status byte. Refer to page 5-55.

***Nrf*** is a Decimal Numeric Data Element representing an integer decimal value equivalent to the Hex value required to enable the appropriate bits in this 8-bit register. The detail definition is contained in the IEEE 488.2 standard document. Note that numbers will be rounded to an integer.

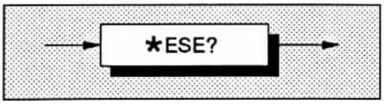
**Execution Errors:**
None.

**Power On and Reset Conditions**
Not applicable.

---

## Recall Event Status Enable

This event status data structure conforms to the IEEE 488.2 standard requirements for this structure.




***ESE?**
recalls the enable mask for the standard defined events. Refer to page 5-55.

**Response Format:**

Character position

| 1 | 2 | 3 | 4 |
|---|---|---|---|
| n | n | n | nl |

**Where:**

n = 0 to 9
nl = newline with EOI

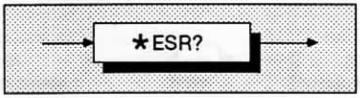
**Response Decode:**
The value returned, when converted to base 2 (binary), identifies the enabled bits which will generate a summary message in the service request byte, for this data structure. The detail definition is contained in the IEEE 488.2 document.

**Execution Errors:**
None

**Power On and Reset Conditions**
The Power On condition depends on the condition stored by the common *PSC command - if 0 then it is not cleared; if 1 then the register is cleared. Reset has no effect.

## Read Event Status Register

This event status data structure conforms to the IEEE 488.2 standard requirements for this structure.




***ESR?**

recalls the standard defined events.
Refer to page 5-55.

**Response Format:**

Character position

| 1 | 2 | 3 | 4 |
|---|---|---|---|
| n | n | n | nl |

**Where:**

n = 0 to 9

nl = newline with EOI

**Response Decode:**

The value returned, when converted to base 2 (binary), identifies the bits as defined in the IEEE 488.2 standard.

**Execution Errors:**

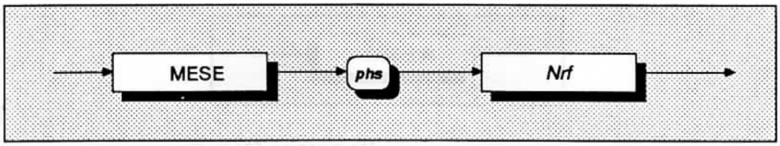
None

**Power On and Reset Conditions**

The Power On condition depends on the condition stored by the common *PSC command - if 0 then it is not cleared; if 1 then the register is cleared. Reset has no effect.

## Measurement Event Enable

This measurement event status data structure conforms to the IEEE 488.2 standard requirements for this structure.




**MESE**
enables the measurement event bits which will generate a summary message in the standard defined service request byte. Refer to page 5-55.

***Nrf*** is a Decimal Numeric Data Element representing a value which, when rounded to an integer and expressed in base 2 (binary), enables the appropriate bits in this event enable register. For example:

32 will enable the MOF bit;
24 will enable Overload and Underrange.

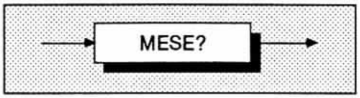
Note that numbers will be rounded to an integer.

**Execution Errors:**
None.

**Power On and Reset Conditions**
Not applicable.

## Recall Measurement Event Enable

This measurement event status data structure conforms to the IEEE 488.2 standard requirements for this structure.




**MESE?**
recalls the measurement status register enable mask. Refer to page 5-55.

**Response Format:**

Character position
1 2 3 4
n n n nl

**Where:**

n = 0 to 9
nl = newline with EOI

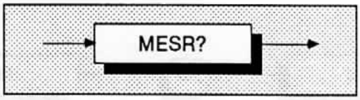
**Response Decode:**
The value returned, when converted to base 2 (binary), identifies the enabled bits which will generate a summary message in the service request byte, for this data structure. See the device status reporting model for detail.

**Execution Errors:**
None

**Power On and Reset Conditions**
Cleared (ie. nothing enabled).

## Read Measurement Event Register

This measurement event status data structure conforms to the IEEE 488.2 standard requirements for this structure.




**MESR?**
reads the event register for measurement qualifiers destructively. Refer to page 5-55. The register is also cleared by the common command *CLS.

**Response Format:**

Character position

| 1 | 2 | 3 | 4 |
|---|---|---|---|
| n | n | n | nl |

**Where:**

n = 0 to 9
nl = newline with EOI

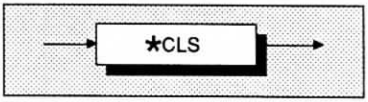
**Response Decode:**
The value returned, when converted to base 2 (binary), identifies the events that have occurred since the most-recent read or general clear of this register. The detail is contained in the status data structure description.

**Execution Errors:**
None.

**Power On and Reset Conditions**
The register is cleared.

## Clear Status

This measurement event status data structure conforms to the IEEE 488.2 standard requirements for this structure.



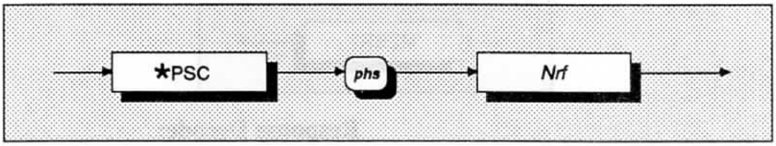

***CLS**
clears all the event registers and queues except the output queue. The output queue and MAV bit will be cleared if *CLS immediately follows a 'Program Message Terminator'; refer to the IEEE 488.2 standard document.

**Execution Errors:**
None.

**Power On and Reset Conditions**
Not applicable.

## Power On Status Clear

This common command conforms to the IEEE 488.2 standard requirements.




***PSC**
sets the flag controlling the clearing of defined registers at Power On.

***Nrf*** is a decimal numeric value which, when rounded to an integer value of zero, sets the *power on clear flag* **false**. This allows the instrument to assert SRQ at power on.
When the value rounds to an integer value other than zero it sets the *power on clear flag* **true**, which clears the standard *event status enable* and *service request enable* registers so that the instrument will not assert an SRQ on power up.

Examples:
*PSC 0 or *PSC 0.173 sets the instrument to **assert** an SRQ at Power On.
*PSC 1 or *PSC 0.773 sets the instrument to **not assert** an SRQ on Power On.

**Execution Errors:**
None.

**Power On and Reset Conditions**
Not applicable.

## Recall Status Clear Flag (*PSC?)

This common command conforms to the IEEE 488.2 standard requirements.
Refer to page 5-51.

# Calibration Commands and Messages

## Caution

**The descriptions in the following pages are intended only as a guide to the messages available to calibrate the instrument. They contain neither examples nor calibration routines, and should NOT be used directly as a basis for calibrating any part of the instrument. Some of the commands, if used unwisely, will obliterate an expensive calibration or recalibration.**

**For a guide to calibration routines refer to Section 8.**

## Calibration Sequences

Remote calibration via the IEEE 488 system bus generally follows similar sequences (and is subject to similar constraints) as for local calibration. But because the remote method does not require a human operator to gain access to a sequence of commands via a single menu screen, it is possible to group commands together within bus message units.

For this reason we should not always expect to find a one-to-one correspondence between the local and remote calibration commands.

### General Outline of Calibration Operations

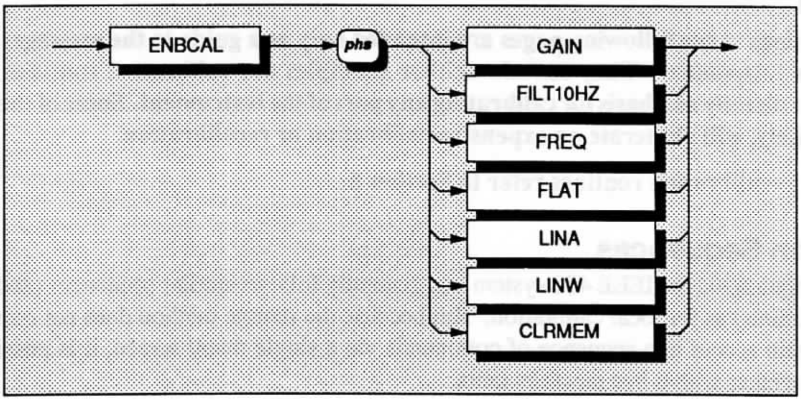
The calibration process generally conforms to a set sequence of operation:

1. The rear-panel switch must be set to ENABLE, then calibration is enabled using the ENBCAL command, which also requires the type of calibration to be selected (this may need further parameters to be specified).
2. With the appropriate analog input applied, the calibration operation is triggered (usually using the CAL? query). An optional parameter is available for use when the calibration is to be performed at a non-nominal value. Subsequently, in normal use; gain and flatness calibrations are applied to correct the pre-selected function and range. Filter and linearity calibrations are carried out on one range and applied to correct all ranges of the pre-selected function.
3. Other operations can be carried out, such as setting the calendar/clock or the calibration interval.
4. When all calibration operations are complete, the CALSEAL command can be used to record the date of the current calibration. Calibration is then disabled using the EXITCAL command and switching the rear panel switch to DISABLE.

Successive calibration program message units can be sent within a single program message; but as it is normal for calibration operations to be interspersed with other actions to change ranges, inputs etc., it is more usual for each calibration operation to use a terminated program message.

## Enable Calibration

The ENBCAL command allows access to the calibration operations, provided the calibration switch on the instrument rear panel is set to 'ENABLE'. It also permits a choice between six types of calibration process, and permits selected areas of non-volatile calibration memory to be cleared.




## Effects of Data Elements

**GAIN**
A subsequent CAL? will trigger calibration of the gain of the selected function/range.

**FILT10HZ**
A subsequent CAL? will trigger calibration of the 10Hz filter if the 10V ACV range is selected.

**FREQ**
A subsequent CAL? will trigger calibration of the frequency counter if the 300mV, 1V, 3V or 10V ACV range is selected.

**FLAT**
A subsequent CAL? will trigger calibration of the flatness of the selected function/range.

**LINA**
A subsequent CAL? on the 10V ACV range will trigger calibration of the linearity of all ACV ranges, and of the 1V WBV range.

**LINW**
A subsequent CAL? will trigger calibration of the linearity of the 3V WBV range if selected.

**CLRMEM (Caution!)**
Enables the use of the CLRMEM program command, which in turn will clear the section of non-volatile calibration memory specified as its parameter. Refer to page 5-75.

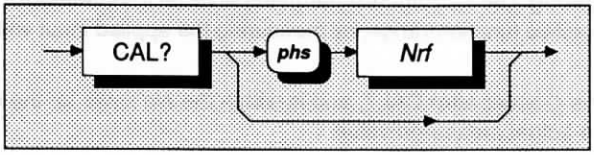
**Protected Commands**
ENBCAL with any parameter enables the protected commands CALINT, CALSEAL, SERIAL, *PUD and EXITCAL, provided the rear panel switch is in the enable position.

**Execution Errors:**
An execution error is generated if the rear panel switch is not in the ENABLE position.

**Power On and Reset Conditions**
Calibration disabled.

## Trigger 'Calibration'

The CAL? command triggers a calibration event. Optional selection of a user-specific non-nominal target value is also available.




### Calibration Point Targeting

The instrument determines whether it is intended to target an upper or lower calibration point on the currently active function/range, by measuring the applied input voltage and evaluating the unprocessed reading.

### *Nrf*

is a decimal numeric data element representing the user-specific value which is to be assigned as target for the actual measured value. The difference between these two values is used to determine the correction factors.
The *Nrf* value is rounded to 7 digits resolution.

If the *Nrf* data element is included then *phs* is required. The target value must conform to the limits required for the intended calibration point.

### *Nrf* omitted

If the program header separator (*phs*) and *Nrf* are omitted, the instrument assumes that the nominal value is the target for the actual measured value.

### Response Format:

A single ASCII character is returned.

A single query sent as a terminated program message will elicit a single response terminated by:

nl = newline with EOI

If multiple queries are sent as a string of program message units (each separated by semi-colons, with the string followed by a permitted terminator), then the responses will be sent as a similar string whose sequence corresponds to the sequence of the program queries. The final response in the string will be followed by the terminator:

nl = newline with EOI

### Response Decode:

The value returned identifies the success or failure of the calibration operation:

**Zero** indicates **complete** with **no error** detected.

**One** indicates **error detected**. The error can be found in the device-dependent error queue.

### Execution Errors

occur if calibration is not enabled, or if the number used is incompatible with the setting being calibrated.

### Power On and Reset Conditions

Not applicable.

## ENBCAL Parameters - GAIN, FREQ, FLAT, LINA, LINW - Further Details

### GAIN

**Calibration Point Targeting** The instrument determines whether it is intended to target an upper or lower calibration point on the active function/range, by measuring the applied input voltage and evaluating the unprocessed reading.

***Nrf***: If the optional *Nrf* is present after CAL?, and is within the *Nrf* limits for the calibration point, then it is used as target for the actual measured value.

***Nrf* omitted**: The instrument assumes that nominal is the target for the actual measured value.

**ACV Calibration**

| Range | Lower Calibration Point | | Upper Calibration Point | |
|---|---|---|---|---|
| | **Nominal** | **Nrf Limits** | **Nominal** | **Nrf Limits** |
| **0.3V** | 0.1V | 0.0099V ≤ *Nrf* < 0.15V | 0.3V | 0.15V ≤ *Nrf* < 0.36V |
| **1V** | 0.3V | 0.03V ≤ *Nrf* < 0.5V | 1.0V | 0.5V ≤ *Nrf* < 1.2V |
| **3V** | 1.0V | 0.099V ≤ *Nrf* < 1.5V | 3.0V | 1.5V ≤ *Nrf* < 3.6V |
| **10V** | 3.0V | 0.3V ≤ *Nrf* < 5.0V | 10V | 5.0V ≤ *Nrf* < 12V |
| **30V** | 10V | 0.99V ≤ *Nrf* < 15.0V | 30V | 15.0V ≤ *Nrf* < 36V |
| **100V** | 30V | 3.0V ≤ *Nrf* < 50.0V | 100V | 50.0V ≤ *Nrf* < 120V |
| **300V** | 100V | 9.9V ≤ *Nrf* < 150.0V | 300V | 150.0V ≤ *Nrf* < 360V |
| **1000V** | 300V | 30.0V ≤ *Nrf* < 500.0V | 1000V | 500.0V ≤ *Nrf* < 1200V |
| **Frequency** | All Ranges: | Nominal Fc = 1kHz; | Frequency Limits are 300Hz < Fc < 3kHz | |

**WBV Calibration**

| Range | Lower Calibration Point | | Upper Calibration Point | |
|---|---|---|---|---|
| | **Nominal** | **Nrf Limits** | **Nominal** | **Nrf Limits** |
| **1V** | 0.1V | 0.01V ≤ *Nrf* < 0.05V | 3.0V | 0.05V ≤ *Nrf* < 1.2V |
| **3V** | 0.3V | 0.099V ≤ *Nrf* < 0.15V | 1.0V | 0.15V ≤ *Nrf* < 3.6V |
| **Frequency** | All Ranges: | Nominal Fc = 1kHz; | Frequency Limits: | 300Hz < Fc < 3kHz |
| Alternative Frequency Limits for 1V Range Upper Cal Point: | | | | 0.95MHz < Fc < 1.02MHz |

## FREQ

The frequency counter can be calibrated using the ACV 300mV, 1V, 3V or 10V range.

**Suitable Input Signal Level**: The Lower **GAIN** Calibration Point for the range or greater.

***Nrf***: If the optional *Nrf* is present after CAL?, and is within the *Nrf* frequency limits, then it is used as target calibration frequency.

***Nrf* Omitted**: If *Nrf* is not present, then the nominal calibration frequency of 1MHz is assumed.

***Nrf* and Signal Frequency Limits:**

30kHz ≤ Fc ≤ 1.2MHz.

## LINA

The ACV Linearity can be calibrated using the ACV 1V, 3V or 10V range.

**Suitable Input Signal Level**: The Lower **GAIN** Calibration Point for the range or greater.

***Nrf***: If the optional *Nrf* is present after CAL?, and is within the *Nrf* limits for the upper calibration point, then it is used as target for the actual measured value.

***Nrf* omitted**: The instrument assumes that nominal is the target for the actual measured value.

**Frequency Limits**: 950kHz ≤ Fc ≤ 1MHz.

## LINW

The WBV Linearity can be calibrated using the WBV 3V range.

**Suitable Input Signal Level**: Immaterial - the calibration is performed using internal resources.

**Input Impedance**: Unspecified during this operation.

## FLAT

All ranges can be calibrated for flatness.

**Calibration Point Targeting**

The upper calibration point only is used.

***Nrf***: If the optional *Nrf* is present after CAL?, and is within the **GAIN** *Nrf* limits for the upper calibration point, then it is used as target calibration value.

***Nrf* Omitted**: If *Nrf* is not present, then the nominal **GAIN** upper calibration point and input limits are assumed.

**Frequency Limits**: Signal frequency Fs is checked against the following bands of limits before use:

**ACV: 0.3V; 1V; 3V; 10V; 30V & 100V Ranges**
**WBV: 1V Range**

| | |
|---|---|
| Flatness A: | 9.8kHz ≤ Fs ≤ 51kHz |
| Flatness B: | 98kHz ≤ Fs ≤ 408kHz |
| Flatness C: | 490kHz ≤ Fs ≤ 765kHz |
| Flatness D: | 882kHz ≤ Fs ≤ 1020kHz |

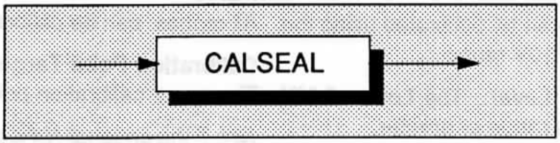
**ACV: 300V & 1000V Ranges**

| | |
|---|---|
| Flatness A: | 4.9kHz ≤ Fs ≤ 15.3kHz |
| Flatness B: | 20.4kHz ≤ Fs ≤ 40.8kHz |
| Flatness C: | 49kHz ≤ Fs ≤ 76.5kHz |
| Flatness D: | 88.2kHz ≤ Fs ≤ 102kHz |

**WBV: 3V Range**

| | |
|---|---|
| Flatness | 2MHz ≤ Fs |

## Set 'Date of Last Calibration'

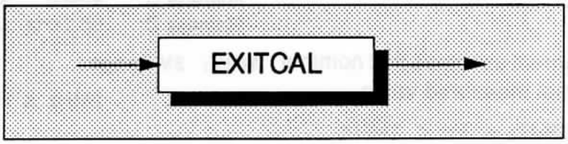



**CALSEAL**
copies today's date to 'Last Cal Date'. This then becomes one of the elements from which the response to the query DUE_DATE is calculated.

**Execution Errors**
CALSEAL is executable only when the rear panel calibration switch is in the enabled position *and* calibration is enabled by the command ENBCAL. Otherwise an Execution Error is returned.

**Power On and Reset Conditions**
The 'Last Cal Date' is saved in non-volatile memory, so is not destroyed at Power Off.

## Exit from Calibration



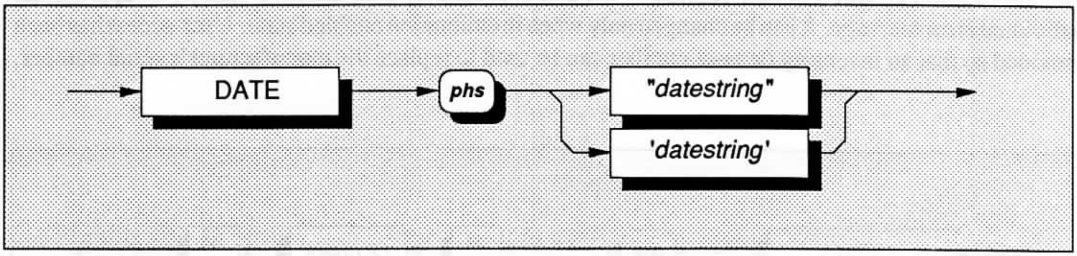

**EXITCAL**
takes the instrument out of calibration mode, thereby disabling the calibration operations which were enabled by the command ENBCAL.

**Protected Commands**
EXITCAL also disables the protected commands CALINT, CALSEAL, SERIAL, *PUD, and EXITCAL itself.

**Execution Errors**
EXITCAL is executable only when the rear panel calibration switch is in the enabled position *and* calibration is enabled by the command ENBCAL. Otherwise an Execution Error is returned.

**Power On and Reset Conditions**
Not applicable, as calibration is not enabled.

## Set Internal Clock/Calendar




**DATE**
allows access to re-set the internal Calendar/Clock to 'Today's Date and Time'.

***datestring*** This is a string of ASCII printing characters int he form:

*mmddyyhhMM*

Where:

- *m m* = two digits representing *month* *(01-12)*
- *d d* = two digits representing *day of month* *(valid day of specified month and year)*
- *y y* = two digits representing *year* *(00-99)*
- *h h* = two digits representing *hour* *(00-23)*
- *M M* = two digits representing *minute* *(00-59)*

Unspecified values of *hhMM* or *MM* will remain unchanged. The datestring is set in quotes so that the specified format can be used for the number itself.

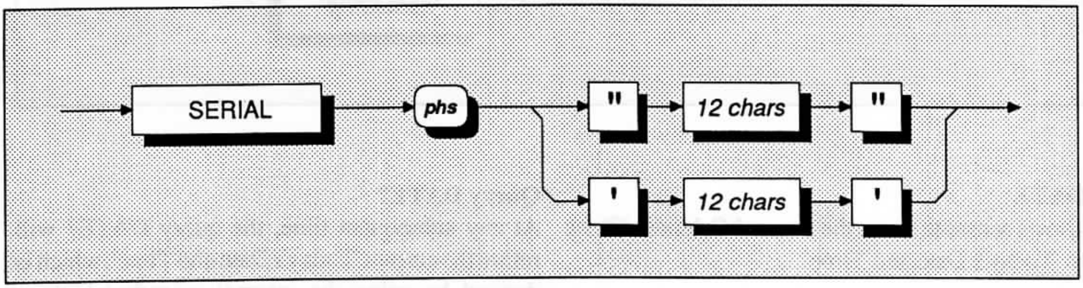
**Query DATE?**
At any subsequent time, the query DATE? will return the current 'Today's Date and Time', which is derived by using the internal calendar/clock to project forward from the most-recent registered date: entered either as a parameter of the above command DATE, or entered from the front panel when in calibration mode. Refer to page 5-49.

**Execution Errors**
DATE is executable only when the rear panel calibration switch is in the enabled position *and* calibration has been enabled by the command ENBCAL. Otherwise an Execution Error is returned.

**Power On and Reset Conditions**
The date is saved in non-voiatile memory, so is not destroyed at Power Off.

## Set Instrument Serial Number

This number is originally set at manufacture to match the serial number on the rear panel plate. The information is stored in non-volatile RAM and is separately sum-checked against an appropriate individual error message. It can be changed only when in calibration enabled state. User-access has been provided so that an inventory or asset number can be used to replace the manufacturer's serial number.




**SERIAL** allows access to change the serial number.
***chars*** are ASCII printing characters.

The serial number is set in quotes so that a free format can be used for the number itself.

It can be recalled together with the manufacturer's name, model number and firmware level, using the standard IEEE 488.2 identification query *IDN? (refer to page 5-47).

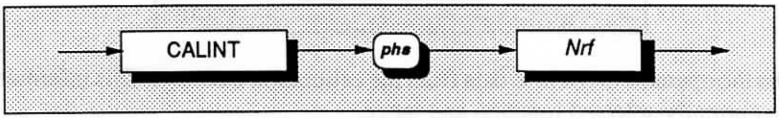
**Execution Errors**

SERIAL is executable only when the rear panel calibration switch is in the enabled position *and* calibration has been enabled by the command ENBCAL. Otherwise an Execution Error is returned.

**Power On and Reset Conditions**

The serial number is saved in non-volatile memory, so is not destroyed at Power Off.

## To Change the Instrument's Calibration Interval




**CALINT** allows the operator to enter the calibration periodicity in days.

***Nrf***

will be rounded to an integer representing the number of days between calibrations. After rounding it must satisfy: 0 < *Nrf* < 1000.

Theinteger can be returned using the program query CALINT?. It can also be displayed by a front panel user, who can enter a new date only via the (protected) calibration mode menu.

**Calibration Due Date**

The due date can be returned using the program query DUE_DATE? (page 5-53). It is calculated internally; by projecting the calibration interval forward, starting at the date which was most-recently copied to 'Last Cal Date' by the command CALSEAL (page 5-68).

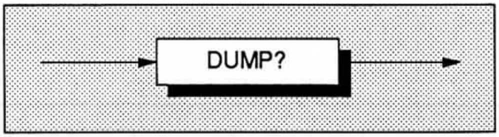
**Execution Errors**

CALINT is executable only when the rear panel calibration switch is in the enabled position *and* calibration has been enabled by the command ENBCAL. Otherwise an Execution Error is returned.

**Power On and Reset Conditions**

The integer is saved in non-volatile memory, so is not destroyed at Power Off.

## To Report the Contents of Calibration Memory




**DUMP?** allows the operator to retrieve data from the calibration memory. This information is related to the currently-selected function and range.

### Response

Each response consists of an three digit index number followed by two signed floating point numbers. The index number defines the source of the data in the other two numbers, according to the list opposite.

The index number is reset to zero whenever function is changed remotely, or whenever either function or range is changed locally.

In response to successive DUMP? queries without index reset; the index number is incremented by 1 in each response, and the source of the information is changed as shown in the list opposite.

To return data from a particular source, the DUMP? query must be repeated until the index for that source is reached.

---

**Response Format:**

Character position

| 1 | 2 | 3 | 4 |
|---|---|---|---|
| n | n | n | , |

| 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| sg | n | . | n | n | n | n | n | n | n | E | sg | n | n | , |

| 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| sg | n | . | n | n | n | n | n | n | n | E | sg | n | n |

**Where:**

n = 0 to 9
E = ASCII character identifying the exponent
sg = + or -

---

A single query sent as a terminated program message will elicit a single response terminated by:

nl = newline with EOI

If multiple queries are sent as a string of program message units (separated by semi-colons with the string followed by a permitted terminator), then the responses will be returned as a similar string whose sequence corresponds to the sequence of the program queries. The final response in the string will be followed by the terminator:

nl = newline with EOI

**Response Decode:**
The data contained in the three fields is organized as in the following list:

**Cal Data for all ACV Ranges and WBV 1V Range**

| | | | |
|---|---|---|---|
| 001 , Lower LF Cal Nominal , | Lower LF Cal Reading | (Volts) | (Volts) |
| 002 , Upper LF Cal Nominal , | Upper LF Cal Reading | (Volts) | (Volts) |
| 003 , HF Cal frequency , | Flatness Error | (Hertz) | (Fraction) |
| 004 , Band A Flatness Cal Frequency , | Flatness Error | (Hertz) | (Fraction) |
| 005 , Band B Flatness Cal Frequency , | Flatness Error | (Hertz) | (Fraction) |
| 006 , Band C Flatness Cal Frequency , | Flatness Error | (Hertz) | (Fraction) |
| 007 , Band D Flatness Cal Frequency , | Flatness Error | (Hertz) | (Fraction) |
| 008 , Frequency Gain Factor , | 10Hz Filter Error | (Fraction) | (Fraction) |
| 009 , Lower AC Lin Nominal , | Lower AC Lin Reading | (Volts) | (Volts) |
| 010 , Upper AC Lin Nominal , | Upper AC Lin Reading | (Volts) | (Volts) |

Note that these last three pairs should be range-independent.

**Cal Data for WBV 3V Range**

| | | | |
|---|---|---|---|
| 001 , Lower Cal Nominal , | Lower Cal Reading | (Volts) | (Volts) |
| 002 , Upper Cal Nominal , | Upper Cal Reading | (Volts) | (Volts) |
| 003 , HF Cal frequency , | Flatness Error | (Hertz) | (Fraction) |
| 004 , Lower DAC Cal Setting , | A-D Response | (Fraction) | (Fraction) |
| 005 , Upper DAC Cal Setting , | A-D Response | (Fraction) | (Fraction) |
| 006-101,Linearity Cal RMS Stimulus, | TTU Response | (Fraction) | (Fraction) |

**Errors and Corruptions**
If the index is forced outside its range by excessive repetitions of DUMP?, then the number +2.0000000E+32 is always returned.

If the calstore information is corrupt (i.e. it is a mismatch with its stored complement) then the number -2.0000000E+32 is always returned.

Because DUMP? is mostly range related, it is inadvisable to be in Autorange when DUMP? is received - the effects will be unpredictable if the instrument is triggered.

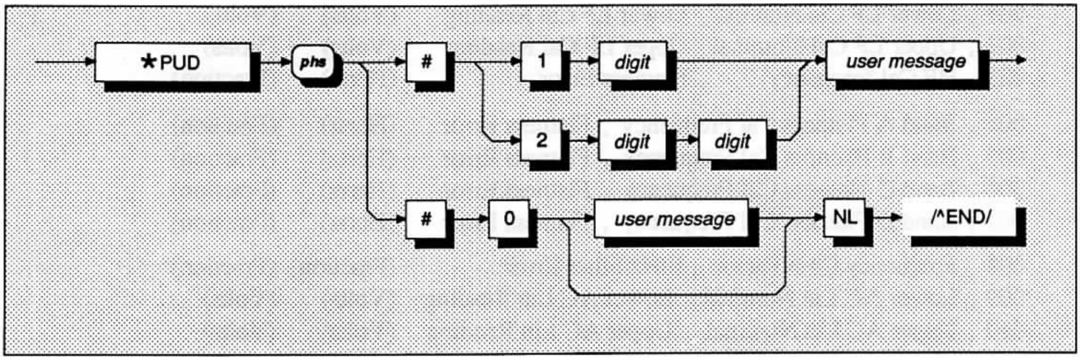
**Execution Errors:**
None.

**Power On and Reset Conditions**
Not applicable.

# Protected User Data

## Entry of User Data

This command conforms to the IEEE 488.2 standard requirements.




**where:**

*phs* = Program Header Separator,
*digit* = one of the ASCII-coded numerals,
*user message* = any message up to 63 bytes maximum.

***PUD**

allows a user to enter up to 63 bytes of data into a protected area to identify or characterize the instrument. The two representations above are allowed depending on the message length and the number of 'digits' required to identify this. The instrument must be in calibration mode for this command to execute.

The data can be recalled using the *PUD? query. Refer to page 5-50.

**Execution Errors**

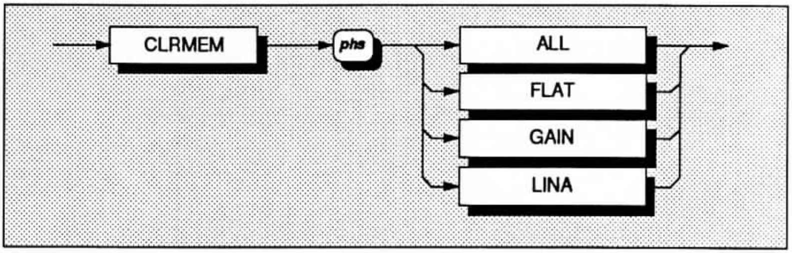
*PUD is executable only when the rear panel calibration switch is in the enabled position *and* calibration has been enabled by the command ENBCAL. Otherwise an Execution Error is returned.

**Power On and Reset Conditions**

Data area remains unchanged.

## Clear Calibration Stores

To allow the calibration correction memories to be cleared.




**CLRMEM**

**Caution!**

This command can **obliterate** the results of an **expensive** original calibration or recalibration!

**Extent of Clear**

The extent of clear is defined by programming the following options:

ALL clears all non-volatile calibration memory *except* the data entered using the program headers: SERIAL and *PUD.

CALINT: The calibration interval is reset to 90 days,
DATE: The internal calendar/clock is reset to 121281.

FLAT Clears the section of non-volatile calibration memory which contains corrections to flatness.

GAIN Clears the section of non-volatile calibration memory which contains corrections to range gain.

LINA Clears the section of non-volatile calibration memory which contains corrections to ACV linearity.

**ROM Checksums**

ROM checksums are recalculated and written to calibration memory when any CLRMEM command is executed.

**Execution Errors**

CLRMEM is executable only when the rear panel calibration switch is in the enabled position *and* calibration has been enabled by the command ENBCAL. Otherwise an Execution Error is returned.

**Power On and Reset Conditions**

Not applicable.

*Appendix A to Section 5 of the User's Handbook for Datron Model 4920M*

# IEEE 488.2 Device Documentation Requirements

IEEE 488.2 requires that certain information be supplied to the user about how the device has implemented the standard. The Device Documentation Requirements are detailed in Section 4.9 of the Standard document, on page 32. In this handbook, the required information is already contained within the descriptions of the system, and this appendix provides cross-references to those descriptions in which it is presented. The following paragraphs have the same numbers as the paragraphs of Section 4.9 in the Standard document to which they refer.

1. The list of IEEE 488.1 Interface Functions subsets implemented is given as Table 5.1 (page 5-2). The list is also printed close to the IEEE 488 connector on the rear of the instrument.

2. The instrument address is set manually, and the instrument firmware refuses to set any address outside the range 0-30. It responds instead with a Data Entry Error, displayed on the front panel.

3. The (manual only) method of setting the address is described on page 5-4, including the point in time when the 4920M recognizes a user-initiated address change.

4. Appendix B to Section 5 describes the active and non-active settings at power-on.

5. Message Exchange Options:

a. The Input Buffer is a first in - first out queue, which has a maximum capacity of 128 bytes (characters). Each character generates an interrupt to the instrument processor which places it in the Input Buffer for examination by the Parser. The characters are removed from the buffer and translated with appropriate levels of syntax checking. If the rate of programming is too fast for the Parser or Execution Control, the buffer will progressively fill up.
When the buffer is full, the handshake is held.

b. No query returns more than one <RESPONSE MESSAGE UNIT>.

c. All queries generate a response when parsed.

d. No query generates a response when read.

e. No commands are coupled.

6. The following functional elements are used in constructing the device-specific commands:

- Command Program Header
- Query Program Header
- Character Program Data
- Decimal Numeric Program Data.
- String Program Data (DATE, SERIAL, BTST? and PROG?)
- Arbitrary Block Program Data (*PUD)

Compound Command Program Headers are not used

7. *PUD blocks are limited to 63 bytes.

8. Expression Program Data elements are not used.

9. The syntax for each command is described in the general list of commands on pages 5-28 to 5-75. This list includes all queries, for which the response syntax is also described.

10. None. All device-to-device message transfer traffic follows the rules for <RESPONSE MESSAGE> elements.

11. The only command which elicits a Block Data response is the query *PUD?.
Its response consists of #, 2, two digits and a data area of 63 bytes; 67 bytes in all.

12. A separate list of every implemented Common Command and Query is given in the alphabetical index at the start of Section 5.

They are also described in the general list on pages 5-28 to 5-75.

13. *CAL? is not implemented.

14. *DDT is not implemented.

15. Macro commands are not implemented.

16. *IDN? is described on page 5-47.

17. *PUD blocks are limited to 63 bytes.

18. Neither *RDT nor *RDT? are implemented.

19. The states affected by *RST are described for each command in the list of commands and queries on pages 5-28 to 5-75.

Query Command *LRN? is not implemented; neither are Commands *RCL and *SAV.

20. *TST? invokes the Operational Selftest. The response to *TST? is described on page 5-41, with a list of possible errors detailed in Appendix A to Section 4 of this handbook.

The Servicing Handbook Section 2 describes the nature of the tests applied and the resulting error codes for both the Operational Selftest, and the Diagnostic Selftest DTST?.

21. The additional status data structures used in the instrument's status reporting are fully described on pages 5-17 to 5-27.

Operating instructions for the status reporting facilities are given on pages 5-55 to 5-62.

22. All commands are sequential - overlapped commands are not used.

23. As all commands are sequential, there are no pending parallel operations. The functional criterion which is met, therefore, is merely that the associated operation has been completed.

*Appendix B to Section 5 of the User's Handbook for Datron Model 4920M*

# 4920M Device Settings at Power On

**Active Function:**

| Funct. | Range | Filter | Resol. |
|---|---|---|---|
| ACV | 1kV | FILT_100Hz | RESL7 |

**Inactive Function:**

| Funct. | Range | Filter | Resol. |
|---|---|---|---|
| WBV | 3V | FILT_OFF | RESL7 |

**Analog Connections**

| | |
|---|---|
| Input | ACV |

**Analog Processes and Conditioning**

| | |
|---|---|
| Trigger Source | Internal |
| Voltage Reading | 'Overload' given until first trigger processed. |
| Frequency Reading | 'Invalid' given until first trigger processed. |

**Post A-D Processes**

| | |
|---|---|
| Difference and Deviation Readings | Cleared at Power On and Reset, so no reading exists until after the first trigger. |

**Math**

| | | |
|---|---|---|
| AVG | OFF | N as previously entered |

**Calibration Processes**

| | |
|---|---|
| Calibration | Disabled |
| Calibration Corrections | Applied |
| Calibration Interval | Previous interval preserved |
| Calibration Due Date | Previous date preserved |

**Device Monitoring**

| | |
|---|---|
| Last Reading Value Recall | Invalid until after first trigger |
| Last Reading Frequency Recall | Invalid until after first trigger |
| Device I/D (Serial Number) | Previous entry preserved |
| Protected User Data | Previous entry preserved |

| | |
|---|---|
| **Status Reporting Conditions** | |
| **Status Byte Register** | Depends on state of *PSC |
| **Event Status Register** | Depends on state of *PSC |
| **Event Summary Register** | Depends on state of *PSC |
| ***PSC Condition** | Previous state preserved |
| **Output Queue** | Empty until after first trigger or unless error detected |

# SECTION 6 SPECIFICATIONS

## MECHANICAL

| | |
|---|---|
| HEIGHT | 88mm (3.46ins). |
| WIDTH | 427mm (16.81ins). |
| DEPTH | 482 max. (18.98ins) which includes 13mm (0.51ins) of front-panel terminals. |
| WEIGHT | 11.5kg (25 lbs) approx. |
| RACK MOUNTING | Rack mounting ears to fit standard 19inch rack (ANSI-EIA-310-C). Conversion to accept 0.5inch wide slides, including MATE standard (Drg No. 2806701, Sperry). |
| RACK MOUNTING DEPTH | 467mm (18.39ins) excluding rear-panel connectors. |

## ENVIRONMENTAL

| | |
|---|---|
| TEMPERATURE | Non-Operating: -40°C to 71°C.<br>Operating: 10°C to 40°C. |
| HUMIDITY | Operating (non-condensing): 0°C to 30°C : < 95% ± 5% RH. 30°C to 40°C : <75% ± 5% RH. 40°C to 50°C : < 45% ± 5% RH. |
| ALTITUDE | Non-Operating: 0-4570m (15,000 feet).<br>Operating: 0-3050m (10,000 feet). |
| ELECTROMAGNETIC COMPATIBILITY | Meets the requirements of MIL-T-28800D for Type III, Class 5, Style E equipment. |
| SHOCK AND VIBRATION | Meets the requirements of MIL-T-28800D for Type III, Class 5, Style E equipment. |
| FUNGUS RESISTANCE | Meets the requirements of MIL-T-28800D for Type III, Class 5, Style E equipment. |

# ELECTRICAL

| | |
|---|---|
| POWER SUPPLY | Voltage: single-phase 100V-130V or 200V-260V selectable from rear panel.<br>Line Frequency: 47Hz to 63Hz. |
| POWER CONSUMPTION | 37 VA maximum. |
| INPUT PROTECTION | ACV input: 1.1kV rms all ranges.<br>WBV input: 3.5V rms all ranges. |
| INPUT IMPEDANCE | ACV input: >200Ω/V<br>WBV input: 50Ω |
| INPUT VSWR | WBV input: ≤1.02:1 from 10Hz to 20MHz |
| INPUT SENSING | True rms, periodic waveforms. |
| INPUT VOLT. HERTZ | $7.5x10^7$ maximum. |
| WARM UP TIME | 30 minutes to full accuracy after power on. |
| MEASUREMENT ACCURACY | See accuracy specification tables. |
| LONG-TERM STABILITY | Better than ±15ppm for 6 months for voltages between 100mV and 1000V at frequencies between 40Hz and 20kHz. |
| ACQUISITION TIME | Normal Band: >100Hz: 3 seconds.<br><100Hz: 6 seconds.<br>Extended Band: 20 seconds. |
| AUTORANGING | ACV input: Range up at approximately 118% of range. Range down at approximately 28% of range.<br>WBV input: No autoranging capability. |
| SAFETY | Meets the requirements of MIL-T-28800D for Type III, Class 5, Style E equipment. |

# ACCURACY SPECIFICATIONS

## ACV Accuracy

| Range and Frequency | Absolute Accuracy [1][2][3][4][5] ±ppm of reading | |
|---|---|---|
| | 18°C to 28°C | 10°C to 18°C and 28°C to 40°C |
| **300.0000mV to 100.0000V** | | |
| 10Hz - 40Hz | 75 | 300 |
| 40Hz - 20kHz | 30 | 120 |
| 20kHz - 50kHz | 70 | 280 |
| 50kHz - 100kHz | 150 | 600 |
| 100kHz - 500kHz | 300 | 1200 |
| 500kHz - 1MHz | 1000 | 4000 |
| **300.0000V to 1000.000V** | | |
| 10Hz - 40Hz | 80 | 320 |
| 40Hz - 20kHz | 35 | 140 |
| 20kHz - 50kHz | 75 | 300 |
| 50kHz - 100kHz | 150 | 600 |

## Notes to ACV Accuracy Specifications

[1] Traceable to National Standards, and inclusive of National Standards uncertainties.

[2] Valid for a period of 6 months from date of last calibration.

[3] Valid only after a 30 minute warm-up period.

[4] Specifications apply for max resolution.

[5] Specifications apply for inputs >30% of range.

## WBV Accuracy

| Range and Frequency | Absolute Accuracy [1][2][3][4] ±[ppm of reading + μV] | |
|---|---|---|
| | 18°C to 28°C | 10°C to 18°C and 28°C to 40°C |
| 1.000000V [5]<br>10Hz - 500kHz | 1250 + 75 | 5000 + 300 |
| 3.000000V [6]<br>10Hz - 500kHz<br>500kHz - 1MHz<br>1MHz - 10MHz<br>10MHz - 20MHz | 1000 + 0<br>1250 + 0<br>1500 + 0<br>2000 + 0 | 4000 + 0<br>5000 + 0<br>6000 + 0<br>8000 + 0 |

## WBV Frequency Response

| Range and Frequency | Maximum Deviation from 1kHz Reference Level [1][2][3][4] ±ppm of reading | |
|---|---|---|
| | 18°C to 28°C | 10°C to 18°C and 28°C to 40°C |
| 3.000000V [6]<br>10Hz - 40Hz<br>40Hz - 1MHz<br>1MHz - 10MHz<br>10MHz - 20MHz | 750<br>250<br>500<br>1000 | 3000<br>1000<br>2000<br>4000 |

## Notes to WBV Accuracy Specifications

[1] Traceable to National Standards, and inclusive of National Standards uncertainties.

[2] Valid for a period of 6 months from date of last calibration.

[3] Valid only after a 30 minute warm-up period.

[4] Specifications apply for max resolution.

[5] Specifications apply for inputs >10% of range.

[6] Specifications apply for inputs >33% of range.

# Maximum RMS Inputs

## ACV Input [1][2][3]

| Hi | | | |
|---|---|---|---|
| 1100V | Lo | | |
| 1100V | 0V | Safety Ground | |
| 1100V | 0V | 0V | Logic Ground |

## WBV Input [1][2][3]

| Hi | | | |
|---|---|---|---|
| 3.5V | Lo | | |
| 3.5V | 0V | Safety Ground | |
| 3.5V | 0V | 0V | Logic Ground |

## Notes to Maximum Input Tables

[1] Maximum RMS inputs specified assume a peak of ≤RMS x 1.414

[2] Logic Ground and Input Signal Lo (ACV and WBV inputs) are internally connected to Safety Ground.

[3] Excessive voltage at any input during Self Test may affect performance.

# SECTION 7 SPECIFICATION VERIFICATION

## Introduction

This section details the procedures required to check that the 4920M is performing to its accuracy specification. It should be used to verify correct instrument performance when the instrument is first received from the manufacturer, and as a normal part of routine 4920M recalibration.

## Equipment Requirements

| Equipment Type | Range | Accuracy [1][2] |
|---|---|---|
| **DC Voltage Calibrator** [3]<br>e.g. Datron Model 4708 with Option 10 | Voltage: 100mV to 1000V<br>Resolution: 1ppm | ±4ppm |
| **AC Voltage Calibrator/ High Resolution AC Voltage Source**[3]<br>e.g. Datron Model 4708 with Option 20 | Voltage: 300mV to 1000V<br>Frequency: 10Hz to 1MHz<br>Resolution: 1ppm | 1V to 3V: ±210ppm at 10Hz<br>±100ppm at 1kHz<br>±5ppm short-term stability |
| **Wideband AC Voltage Source**<br>e.g. Wavetek Model 178 | Voltage: 1V to 3V<br>Frequency: 10Hz to 20MHz | ±50ppm short-term stability |
| **Wideband Amplifier** | Voltage: 1V to 3V<br>Frequency: 10Hz to 20MHz<br>Gain: 1 approx. | ±25ppm short-term stability |
| **Thermal Voltage Converters**<br>e.g. Holt Model 11, and Ballantine Model 1396 | Voltage: 300mV to 1000V<br>Frequency: DC to 20MHz | To best NIST uncertainties |
| **DC Nanovoltmeter**<br>e.g. EM Model N2a or Keithley Model 181 | ±100nV to ±10mV | Better than 2% of range |
| **DC nV Source**<br>e.g. EM Model S6 | 10mV | ±1ppm short-term stability |

[1] Absolute accuracy (traceable to National Standards, and inclusive of National Standards uncertainties).

[2] Provides a 95% confidence level of achieving a 4:1 calibration ratio except where 'state-of-the-art' limitations apply.

[3] This equipment can be part of a single AC/DC unit. e.g. a Datron 4708 multifunction calibrator with Options 10 and 20.

# Preparation

1. Before operating any of the performance verification equipment, familiarize yourself with the equipment by reading the appropriate operating manuals. In particular, note the electrical and mechanical handling procedures required to maintain the calibration of the thermal voltage converters.
2. **Note any safety precautions which are necessary to prevent electrical shock from the equipment.**
3. Set up the equipment in a stable environment at 23°C±1°C, power it on and allow it to stabilize for an appropriate period of time. The 4920M requires a minimum of 30 minutes (preferably 2 hours) to warm up after it is powered on.
4. Select the **TEST** menu by pressing the front-panel **Test** key and select the **Oper** (Operational Test) menu option. Allow the operational self-test to run to completion, at which point the 4920M should display the word **COMPLETED.** If at any point during the operational self-test the 4920M displays the words **OPER FAIL** (Operational Test Failure) the unit probably has a fault, in which case refer to the 4920M Servicing Handbook before proceeding.

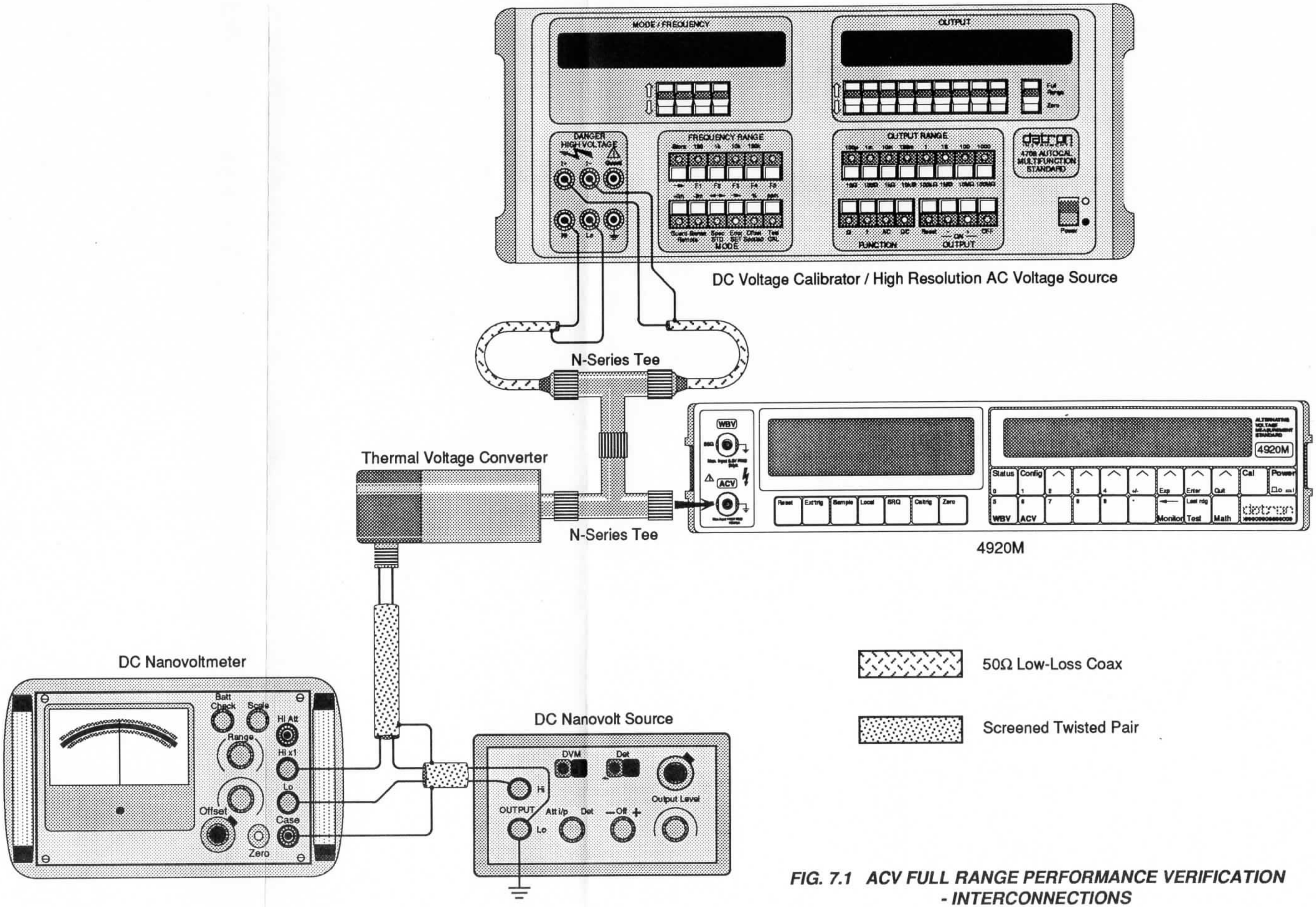



*FIG. 7.1 ACV FULL RANGE PERFORMANCE VERIFICATION - INTERCONNECTIONS*

# ACV (Normal Band) Performance Verification

For each of the verification point voltages detailed in the ACV Full Range Checks section of the 4920M Verification Report Sheet, carry out the following procedure:-

1. Configure a thermal voltage converter for the required verification point voltage.
2. Connect the DC voltage calibrator to the 4920M and thermal voltage converter as shown in Figure 7.1 and select the 4920M's ACV function.
3. Set the output of the DC voltage calibrator to the required verification point voltage and turn the output of the calibrator on.
4. Allow the thermal voltage converter to stabilize.
5. Adjust the DC nV source to achieve a null on the DC nanovoltmeter.
6. Reverse the polarity of the DC voltage calibrator output, allow the thermal voltage converter to settle, and note the DC reversal error on the nanovoltmeter. Adjust the nV source until the nanovoltmeter shows half of the DC reversal error.
7. Turn the output of the calibrator off and disconnect it by splitting the two N-series tees.
8. Connect the high resolution AC voltage source in place of the DC voltage calibrator. (Note: if the DC calibrator and AC voltage source are combined into a single unit, the output of this unit can simply be switched from DC to AC.)
9. Set the AC voltage source to the selected verification point voltage and frequency and turn its output on.
10. Increment or decrement the AC source output voltage to achieve a null on the DC nanovoltmeter. Allow the thermal voltage converter to settle, and check and adjust the null. Note the 4920M reading in the '4920M Reading' column of the Verification Report Sheet.
11. Calculate the true value of the AC voltage source from the data recorded on the 4920M Verification Report Sheet, taking into account the AC-DC transfer error of the thermal voltage converter at the selected voltage and frequency.
12. Calculate the Validity Tolerance Limits as indicated on the Verification Report Sheet and check that the value entered in the '4920M Reading' column is within these limits.

# WBV (Extended Band) Performance Verification

Because of the limited accuracy and setting resolution of most wideband AC voltage sources, the WBV (Extended Band) performance verification procedure differs from that for ACV (Normal Band) performance verification. For WBV performance verification, the absolute accuracy of the 4920M at frequencies up to 1kHz is determined by comparing the 4920M reading to the output of a high accuracy AC calibrator. For frequencies above 1kHz, WBV absolute accuracy is calculated from the 1kHz absolute accuracy and the frequency response flatness of the 4920M — this flatness being determined by performing AC-AC transfers between the verification point frequency and 1kHz, using a wideband thermal voltage converter.

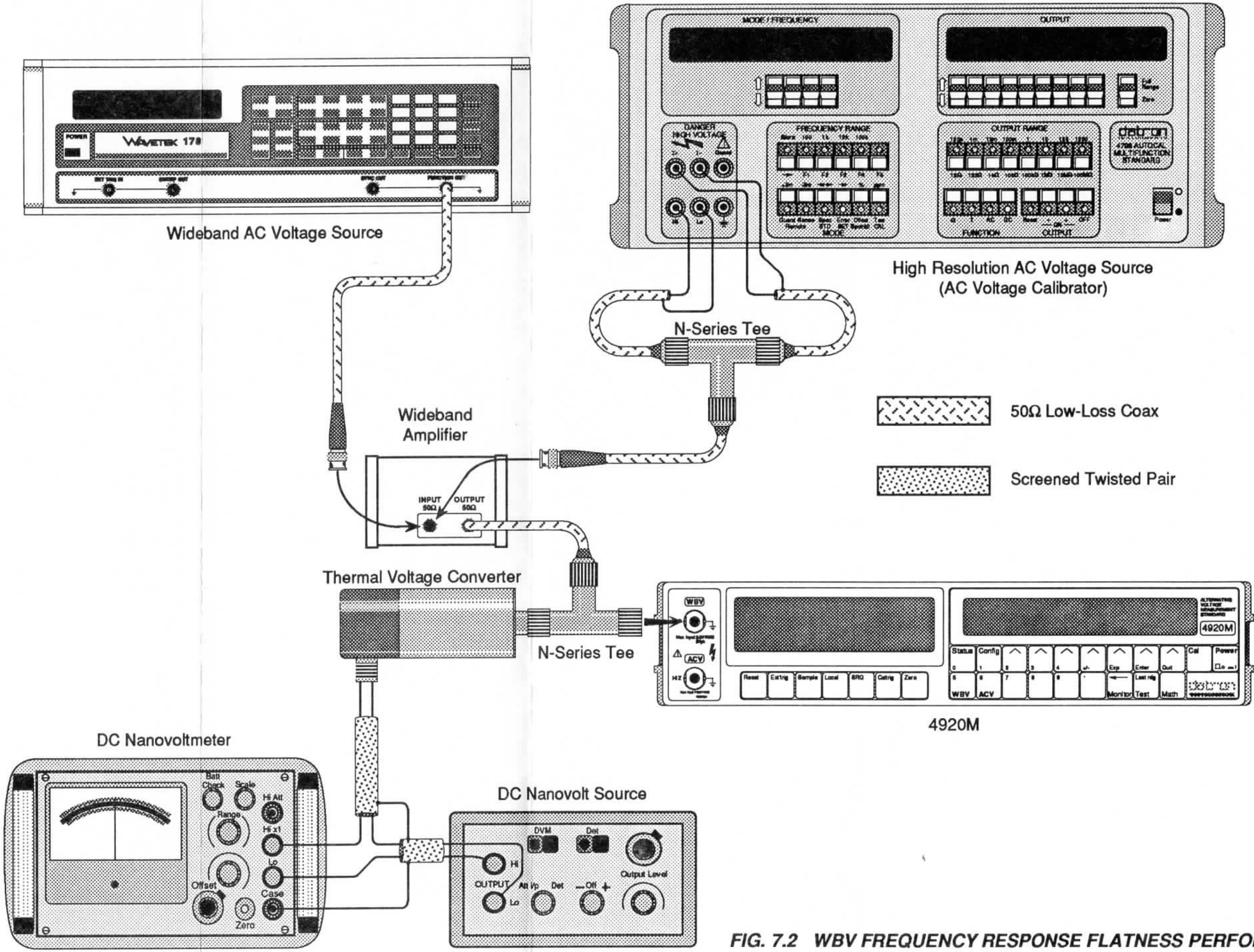



*FIG. 7.2 WBV FREQUENCY RESPONSE FLATNESS PERFORMANCE VERIFICATION - INTERCONNECTIONS*

For each of the verification point voltages and frequencies detailed in the WBV Frequency Response Flatness Checks section of the 4920M Verification Report Sheet carry out the following procedure:-

1. Configure a thermal voltage converter for the required verification point voltage.
2. Connect the wideband AC voltage source and wideband amplifer to the 4920M and thermal voltage converter as shown in Figure 7.2, and select the 4920M's WBV function.
3. Set the wideband source to the selected verification point voltage and frequency.
4. Turn the output of the wideband amplifier on and increment or decrement the output voltage of the wideband source until the 4920M reads as close as possible to the selected verification point voltage. Note the 4920M reading in the '4920M Verification Point Reading' column of the 4920M Verification Report Sheet.
5. Allow the thermal voltage converter to settle.
6. Adjust the DC nV source to achieve a null on the DC nanovoltmeter.
7. Turn the output of the wideband AC voltage source off and disconnect it from the input to the wideband amplifier.
8. Connect the high resolution AC voltage source (AC calibrator) to the input of the wideband amplifier as shown in Figure 7.2.
9. Set the output of the high resolution AC voltage source to the required verification point voltage at a frequency of 1kHz, and turn its output on.
10. Increment or decrement the output voltage of the high resolution AC source to achieve a null on the DC nanovoltmeter. Allow the thermal voltage converter to settle and, if necessary, check and adjust the null.
11. Note the 4920M reading in the '4920M 1kHz Reading' column of the 4920M Verification Report Sheet.
12. Calculate the 4920M's frequency response flatness error from the data recorded on the 4920M Verification Report Sheet, taking into account the AC-AC transfer error of the thermal voltage converter between the verification point frequency and 1kHz.
13. Calculate the Flatness Validity Tolerance and the 4920M Flatness Error as indicated on the Verification Report Sheet and check that the 4920M Flatness Error is within the Flatness Validity Tolerance limits.

For each of the verification point voltages and frequencies detailed in the WBV Low Frequency Full Range Checks section of the 4920M Verification Report Sheet, carry out the following procedure:-

**1.** Connect the AC voltage calibrator to the 4920M as shown in Figure 7.3 and select the 4920M's WBV function.

**2.** Set the output of the calibrator to the required calibration voltage and frequency and turn the output of the calibrator on.

**3.** Note the 4920M reading in the '4920M Reading' column of the Verification Report Sheet.

**4.** Calculate the Accuracy Validity Tolerance Limits as indicated on the Verification Report Sheet and check that the 4920M Reading is within these limits.

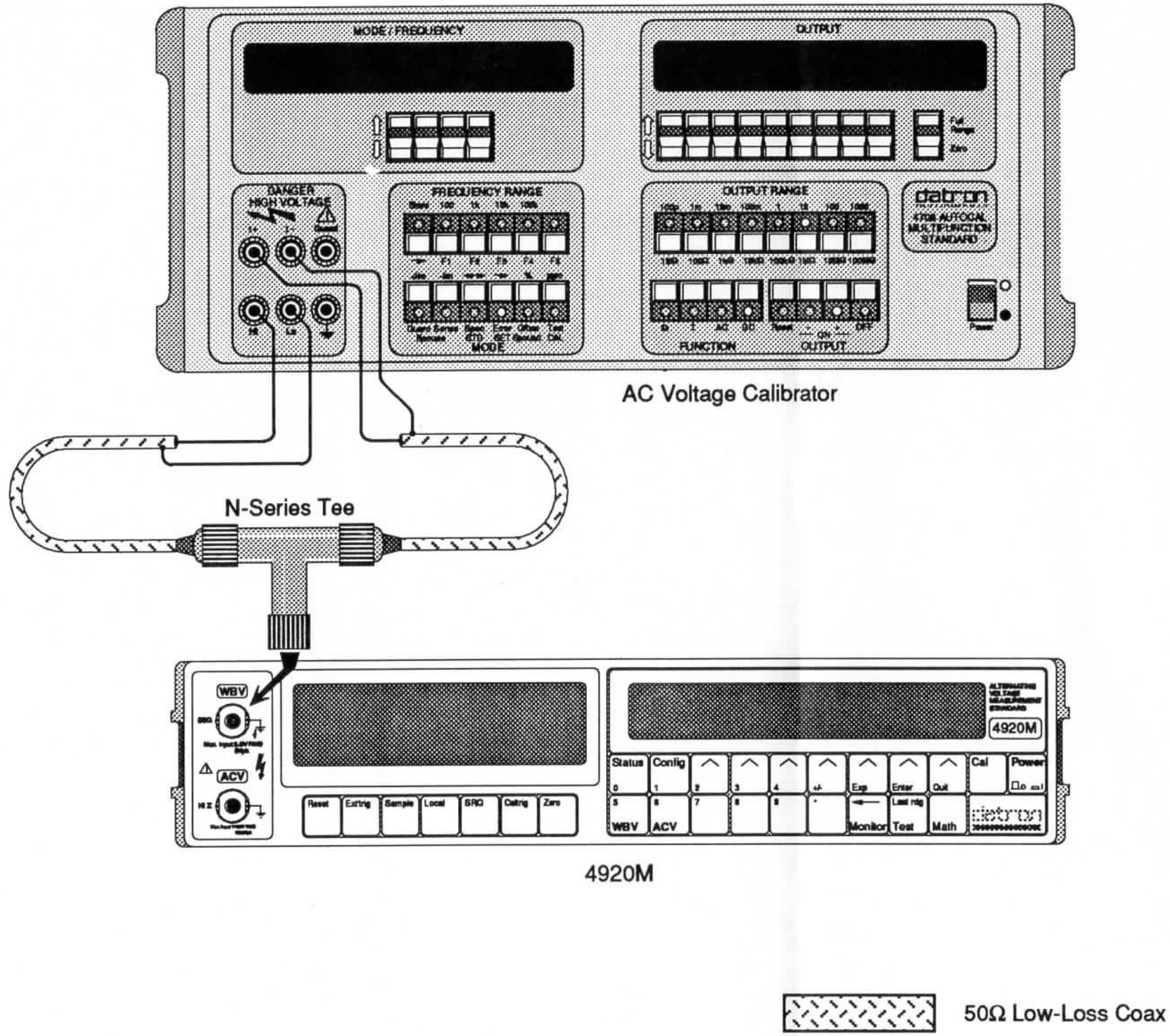



***FIG. 7.3 WBV LOW FREQUENCY FULL RANGE PERFORMANCE VERIFICATION - INTERCONNECTIONS***

## User's Uncertainty Calculations

The accuracy and traceability of a user's standards affect the manner in which the performance of the 4920M can be verified. Because users will need to evaluate the effects of the uncertainties associated with their own equipment in conjunction with the 4920M's accuracy specifications, calculations for total tolerance limits (Validity Tolerances) are required.

### The 'Validity Tolerance'

It is impossible to verify the specification of an instrument with absolute certainty, even using the original calibration equipment to make the measurements. All measurements carry a degree of uncertainty in themselves, these uncertainties being quantified by the traceability of the measuring equipment to National Standards and the repeatability of the transfers involved.

The performance verification measurements which follow are intended to establish that the instrument performs within its specifications - i.e. that it operates within the tolerance of its accumulated uncertainties. As the measurements taken to verify the instrument's performance have their own accumulated uncertainties, these must be added to those of the instrument in order to set a 'Validity Tolerance'.

The Validity Tolerance for the 4920M is obtained by adding all the intervening uncertainties (from the point at which the performance verification measurement is made, back to and including National Standards uncertainties) to the worst case instrument specification at the verification point voltage and frequency. For the standards equipment used, worst-case tolerances must be assumed. Complete the following tables and calculate the validity tolerance limits using the formulae provided. If any range fails to verify and the instrument is to be returned, please be certain to include a copy of the Verification Report Sheet and give as much detail as possible about the performance verification setup.

## User's Uncertainty Calculations

The accuracy and traceability of a user's standards affect the manner in which the performance of the 4920M can be verified. Because users will need to evaluate the effects of the uncertainties associated with their own equipment in conjunction with the 4920M's accuracy specifications, calculations for total tolerance limits (Validity Tolerances) are required.

### The 'Validity Tolerance'

It is impossible to verify the specification of an instrument with absolute certainty, even using the original calibration equipment to make the measurements. All measurements carry a degree of uncertainty in themselves, these uncertainties being quantified by the traceability of the measuring equipment to National Standards and the repeatability of the transfers involved.

The performance verification measurements which follow are intended to establish that the instrument performs within its specifications - i.e. that it operates within the tolerance of its accumulated uncertainties. As the measurements taken to verify the instrument's performance have their own accumulated uncertainties, these must be added to those of the instrument in order to set a 'Validity Tolerance'.

The Validity Tolerance for the 4920M is obtained by adding all the intervening uncertainties (from the point at which the performance verification measurement is made, back to and including National Standards uncertainties) to the worst case instrument specification at the verification point voltage and frequency. For the standards equipment used, worst-case tolerances must be assumed. Complete the following tables and calculate the validity tolerance limits using the formulae provided. If any range fails to verify and the instrument is to be returned, please be certain to include a copy of the Verification Report Sheet and give as much detail as possible about the performance verification setup.

# Verification Report Sheet

**Model 4920M** Serial Number.................... Last Calibration Date...................

Current Date.................... Checked by.................. Company/Dept............................

**Note:** It is advisable to make duplicate copies of this report sheets for future use. Check at the values shown in the tables. Contact your Datron Service Centre if the instrument fails to verify and please include a copy of the completed Verification Report Sheet if the instrument is returned.

### IMPORTANT:

The 4920M accuracy specifications listed in the following tables are absolute accuracies which assume that the 4920M has been traceably calibrated as detailed in Section 8 of the 4920M User's Handbook. If the 4920M has been calibrated against calibration standards which do not have the absolute accuracies detailed in Section 8 of the 4920M User's Handbook then the 4920M accuracy specifications listed in the following tables will not apply.

## Calculation of Validity Tolerances

Intermediate quantities and final Validity Tolerances should be calculated using the equations shown beneath the column headings in the tables. It should be noted that the references used in these equations — for example (a), (b), (c), (d) etc. — refer only to values within the **same** table as indicated.

### ACV Full Range Checks

| ACV Range | Verification Point | | DC Calibrator Output | TVC AC-DC Transfer Error | Calculated AC Voltage | Verification Point Uncertainty | 4920M Accuracy | Accuracy Validity Tolerance Limits | | 4920M Reading |
|---|---|---|---|---|---|---|---|---|---|---|
| | Voltage | Frequency | (V) (a) | (ppm) (b) | (V) (c) = (a) x [1 + (b)x10$^{-6}$] | (±ppm) (d) | (±ppm) (e) | Lower (V) (f) = (c) x {1 - [(d) + (e)]x10$^{-6}$} | Higher (V) (g) = (c) x {1 + [(d) + (e)]x10$^{-6}$} | (V) |
| 0.3V | 300.0000mV | 10Hz | | | | | 75 | | | |
| 0.3V | 300.0000mV | 40Hz | | | | | 30 | | | |
| 0.3V | 300.0000mV | 20kHz | | | | | 30 | | | |
| 0.3V | 300.0000mV | 50kHz | | | | | 70 | | | |
| 0.3V | 300.0000mV | 100kHz | | | | | 150 | | | |
| 0.3V | 300.0000mV | 500kHz | | | | | 300 | | | |
| 0.3V | 300.0000mV | 1MHz | | | | | 1000 | | | |
| 1V | 1.000000V | 10Hz | | | | | 75 | | | |
| 1V | 1.000000V | 40Hz | | | | | 30 | | | |
| 1V | 1.000000V | 20kHz | | | | | 30 | | | |
| 1V | 1.000000V | 50kHz | | | | | 70 | | | |
| 1V | 1.000000V | 100kHz | | | | | 150 | | | |
| 1V | 1.000000V | 500kHz | | | | | 300 | | | |
| 1V | 1.000000V | 1MHz | | | | | 1000 | | | |
| 10V | 10.00000V | 10Hz | | | | | 75 | | | |
| 10V | 10.00000V | 40Hz | | | | | 30 | | | |
| 10V | 10.00000V | 20kHz | | | | | 30 | | | |
| 10V | 10.00000V | 50kHz | | | | | 70 | | | |
| 10V | 10.00000V | 100kHz | | | | | 150 | | | |
| 10V | 10.00000V | 500kHz | | | | | 300 | | | |
| 10V | 10.00000V | 1MHz | | | | | 1000 | | | |
| 100V | 100.0000V | 10Hz | | | | | 75 | | | |
| 100V | 100.0000V | 40Hz | | | | | 30 | | | |
| 100V | 100.0000V | 20kHz | | | | | 30 | | | |
| 100V | 100.0000V | 50kHz | | | | | 70 | | | |
| 100V | 100.0000V | 100kHz | | | | | 150 | | | |
| 100V | 100.0000V | 200kHz | | | | | 300 | | | |
| 1000V | 1000.000V | 45Hz | | | | | 35 | | | |
| 1000V | 1000.000V | 20kHz | | | | | 35 | | | |
| 1000V | 1000.000V | 33kHz | | | | | 75 | | | |

## WBV Frequency Response Flatness Checks

| WBV Range | Verification Point | | 4920M Verification Point Reading (V) (a) | 4920M 1kHz Reading (V) (b) | TVC Flatness Error (ppm) (c) | 4920M Flatness Specification ±(ppm) (d) | Verification Point Uncertainty ±(ppm) (e) | Flatness Validity Tolerance ±(ppm) (f) = (d) + (e) | 4920M Flatness Error (ppm) (g) = $\left[\frac{(a)-(b)}{(b)}\right] \times 10^6 - (c)$ |
|---|---|---|---|---|---|---|---|---|---|
| | Voltage | Frequency | | | | | | | |
| 1V | 1.000000V | 500kHz | | | | | ⑧ | | ⑳ |
| 3V | 3.000000V | 10Hz | | | | 750 | | | |
| 3V | 3.000000V | 40Hz | | | | 250 | | | |
| 3V | 3.000000V | 500kHz | | | | 250 | ⑨ | | ㉑ |
| 3V | 3.000000V | 1MHz | | | | 250 | ⑩ | | ㉒ |
| 3V | 3.000000V | 10MHz | | | | 500 | ⑪ | | ㉓ |
| 3V | 3.000000V | 20MHz | | | | 1000 | ⑫ | | ㉔ |

② = ① + ⑧
④ = ③ + ⑨
⑤ = ③ + ⑩
⑥ = ③ + ⑪
⑦ = ③ + ⑫

⑭ = ⑬ + ( 1 x ⑳ x $10^{-6}$ )
⑯ = ⑮ + ( 3 x ㉑ x $10^{-6}$ )
⑰ = ⑮ + ( 3 x ㉒ x $10^{-6}$ )
⑱ = ⑮ + ( 3 x ㉓ x $10^{-6}$ )
⑲ = ⑮ + ( 3 x ㉔ x $10^{-6}$ )

## WBV Low Frequency Full Range Checks

| WBV Range | Verification Point | | 4920M Accuracy Specification (±ppm) (b) | Verification Point Uncertainty (±ppm) (c) | Accuracy Validity Tolerance Limits | | 4920M Reading (V) |
|---|---|---|---|---|---|---|---|
| | Voltage (a) | Frequency | | | Lower (V) (d) = (a) x { 1 - [(b) + (c)]x$10^{-6}$} | Higher (V) (e) = (a) x { 1 + [(b) + (c)]x$10^{-6}$} | |
| 1V | 1.000000V | 10Hz | 1325 | | | | |
| 1V | 1.000000V | 1kHz | | ① | | | ⑬ |
| 1V | 1.000000V | 500kHz | 1325 | ② | | | ⑭ |
| 3V | 3.000000V | 10Hz | 1000 | | | | |
| 3V | 3.000000V | 1kHz | | ③ | | | ⑮ |
| 3V | 3.000000V | 500kHz | 1000 | ④ | | | ⑯ |
| 3V | 3.000000V | 1MHz | 1250 | ⑤ | | | ⑰ |
| 3V | 3.000000V | 10MHz | 1500 | ⑥ | | | ⑱ |
| 3V | 3.000000V | 20MHz | 2000 | ⑦ | | | ⑲ |

# SECTION 8 ROUTINE CALIBRATION

## Introduction

**IMPORTANT**: The following procedures allow the 4920M to be recalibrated to external calibration standards. To verify the 4920M's performance to specification without affecting the instrument's calibration memory, refer to Section 7 of this manual.

In addition to the routine calibration procedures detailed in this section, the 4920M requires several special calibration operations to be performed after it has undergone repair. For details of these special calibration operations refer to Section 1 of the 4920M Servicing Handbook.

## Autocal

The 4920M includes Datron's autocal feature, which allows full control of external calibration either from the unit's front panel controls or via its integral IEEE-488.2 bus interface. The autocal feature eliminates the thermal disturbance which would occur if the instrument's covers had to be removed during calibration, and it greatly speeds up the calibration process.

To perform autocalibration, appropriate calibration voltages are applied to the 4920M for each of its measurement ranges. As each of these calibration voltages is applied to the instrument, a single depression of the front-panel 'Caltrig' key or an appropriate IEEE-488.2 calibration command causes the 4920M to calculate an appropriate digital calibration constant, which is stored in non-volatile memory within the instrument. These calibration constants are subsequently used to correct the output of the 4920M's internal A/D converter so that the instrument provides accurate measurements of the input signal.

The 4920M automatically determines whether the applied calibration voltage is at a bottom-of-range or top-of-range calibration point, and whether it is a low-frequency or a high-frequency input. Provision is also made for the user to enter the true value of the calibration source where this differs from the nominal bottom-of-range or top-of-range value.

Access to the autocal function is protected by a rear-panel slide switch. This switch must be set to the 'Enable' position before the autocal feature can be operated.

It should be noted that it is not always necessary to perform all the calibration procedures detailed in this section. Because each of the 4920M's measurement ranges can be calibrated separately, without affecting the measurement performance of other ranges, partial recalibration of the 4920M can be carried out if required.

The ACV input calibration procedures detailed in this section use AC-DC Thermal Transfer Techniques to traceably refer the 4920M measurement performance to DC calibration sources. If it is required to automate the calibration process, or to calibrate large numbers of 4920Ms, it may be better to characterize the performance of a suitable AC voltage calibrator and to then use this calibrator as a direct calibration source for the 4920M.

# The CAL Menu

Pressing the front-panel **Cal** key, with the rear-panel **CALIBRATION** switch set to the **ENABLE** position, opens the **CAL** menu as illustrated opposite.

The **CAL** menu provides the following functions:-

### Date

The **Date** menu option allows display and modification of the current date.

### Ser#

The **Ser#** (Serial Number) menu option allows the instrument serial number to be displayed and, if necessary, modified.

### Spcl

The **Spcl** (Special Calibration) menu option allows various special calibration operations to be performed. These calibration operations need only be performed when the instrument is set up immediately after manufacture or repair. For more information on special calibration operations refer to the 4920M Servicing Manual.

### Set

The **Set** menu option allows the operator to enter the known value of a calibration input where this differs from the nominal bottom-of-range or top-of-range value. The set function must be used immediately prior to use of the **Caltrig** key.

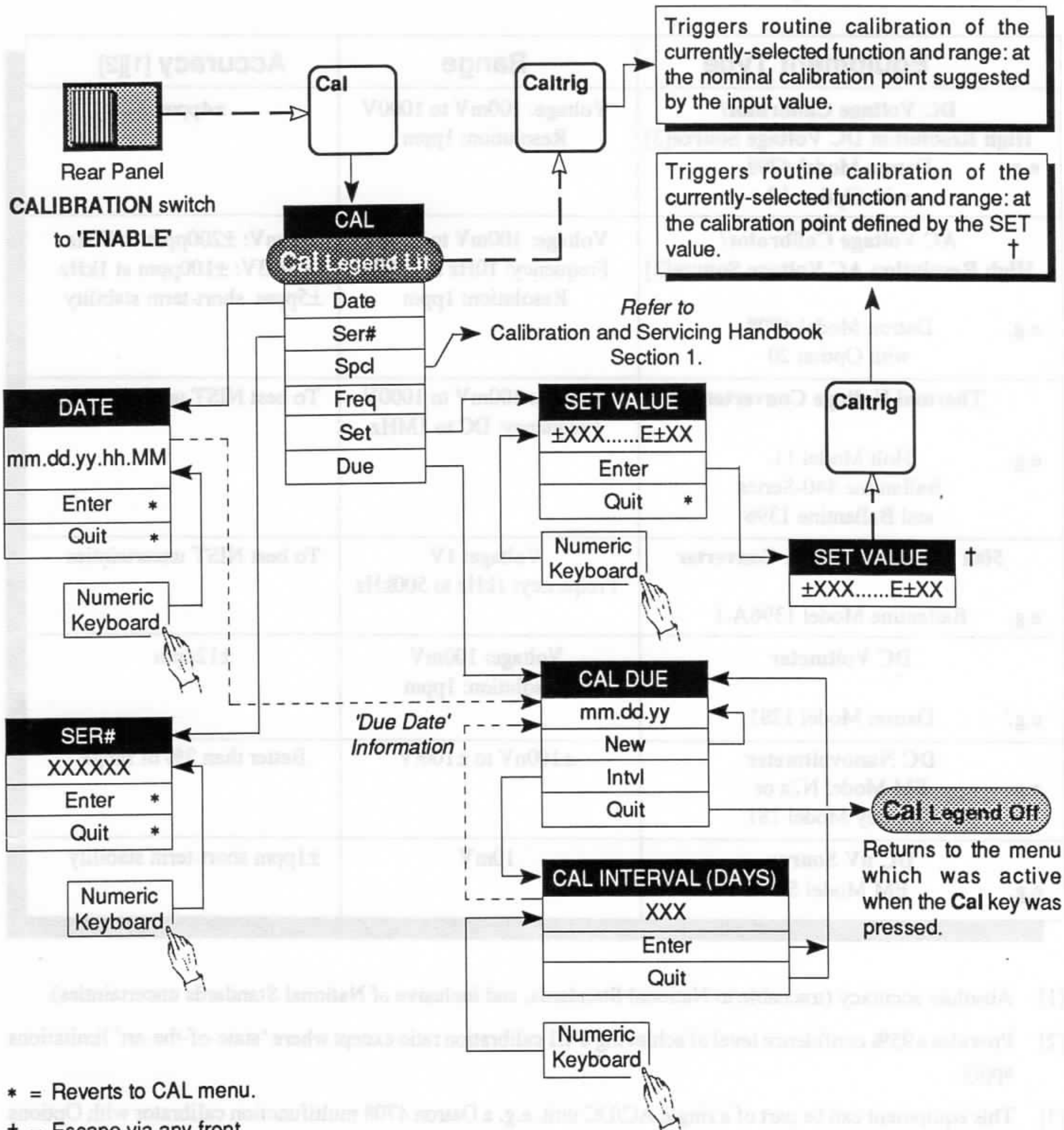
### Due

The **Due** menu option allows the due date for instrument recalibration to be displayed and, if necessary, modified.

For more information on the use of the **CAL** menu options refer to Section 4 of this manual.

## CAUTION:

**While the CAL menu or any of its menu options is displayed, the Caltrig key is enabled, and when pressed alters the contents of the calibration memory. To avoid the risk of inadvertently overwriting previous calibration data, the Caltrig key should only be used for genuine calibration operations.**




* = Reverts to CAL menu.

† = Escape via any front panel menu key.

# Equipment Requirements

| Equipment Type | Range | Accuracy [1][2] |
|---|---|---|
| **DC Voltage Calibrator/ High Resolution DC Voltage Source**[3]<br>e.g. Datron Model 4708 with Option 10 | Voltage: 100mV to 1000V<br>Resolution: 1ppm | ±4ppm |
| **AC Voltage Calibrator/ High Resolution AC Voltage Source**[3]<br>e.g. Datron Model 4708 with Option 20 | Voltage: 100mV to 1000V<br>Frequency: 10Hz to 1MHz<br>Resolution: 1ppm | 100mV: ±200ppm at 1kHz<br>1V to 3V: ±100ppm at 1kHz<br>±5ppm short-term stability |
| **Thermal Voltage Converters**<br>e.g. Holt Model 11, Ballantine 440-Series and Ballantine 1396 | Voltage: 100mV to 1000V<br>Frequency: DC to 1MHz | To best NIST uncertainties |
| **50Ω Thermal Voltage Converter**<br>e.g. Ballantine Model 1396A-1 | Voltage: 1V<br>Frequency: 1kHz to 500kHz | To best NIST uncertainties |
| **DC Voltmeter**<br>e.g. Datron Model 1281 | Voltage: 100mV<br>Resolution: 1ppm | ±12ppm |
| **DC Nanovoltmeter**<br>e.g. EM Model N2a or Keithley Model 181 | ±100nV to ±10mV | Better than 2% of range |
| **DC nV Source**<br>e.g. EM Model S6 | 10mV | ±1ppm short-term stability |

[1] Absolute accuracy (traceable to National Standards, and inclusive of National Standards uncertainties).

[2] Provides a 95% confidence level of achieving a 4:1 calibration ratio except where 'state-of-the-art' limitations apply.

[3] This equipment can be part of a single AC/DC unit. e.g. a Datron 4708 multifunction calibrator with Options 10 and 20.

# Preparation

1. Before operating any of the calibration equipment, familiarize yourself with the equipment by reading the appropriate operating manuals. In particular, note the electrical and mechanical handling procedures required to maintain the calibration of the thermal voltage converters.

2. **Note any safety precautions which are necessary to prevent electrical shock from the equipment.**

3. Set up the equipment in a stable environment at 23°C±1°C, power it on and allow it to stabilize for an appropriate period of time. The 4920M should be allowed to stabilize for a minimum of 2 hours after it is powered on.

4. Select the **TEST** menu by pressing the front-panel **Test** key and select the **Oper** (Operational Test) menu option. Allow the operational self-test to run to completion, at which point the 4920M should display the word **COMPLETED.** If at any point during the operational self-test the 4920M displays the words **OPER FAIL** (Operational Test Failure) the unit probably has a fault, in which case refer to the 4920M Servicing Handbook before proceeding.

5. Set the rear-panel **CALIBRATION** switch to the **ENABLE** position.

6. Press the front-panel **Cal** key to display the **CAL** menu.

# ACV (Normal Band) Calibration

For the ACV 100mV calibration point carry out the following procedure:-

1. Configure a Selby micropotentiometer for 100mV.
2. Connect the high resolution DC voltage source (DC calibrator) into the calibration setup as shown in Figure 8.1 and select the 4920M's ACV 0.3V range.
3. Turn the output of the DC voltage source on and increment its output from zero until the DC voltmeter indicates a reading of 100.0000mV. Allow the Selby micropotentiometer to settle and, if necessary, readjust the DC source to achieve a DC voltmeter reading of 100.0000mV.
4. Adjust the DC nV source to achieve a null on the DC nanovoltmeter.
5. Reverse the output polarity of the DC voltage source and allow the Selby micropotentiometer to settle.
6. Note the new reading on the DC voltmeter, the new reading on the DC nanovoltmeter, and the output voltage of the DC voltage source.
7. Turn the output of the DC voltage source off and disconnect the DC voltage source and the DC voltmeter from the calibration setup.
8. Connect the high resolution AC voltage source (AC calibrator) in place of the DC voltage source. (Note: if the DC voltage source and the AC voltage source are combined into a single unit, the output of this unit can simply be switched from DC to AC.)
9. Set the output of the AC voltage source to the same value as the DC source output noted in step 6, and a frequency of 1kHz. Turn its output on and increment or decrement the output to achieve a DC nanovoltmeter reading of half the DC nanovoltmeter reading noted in step 6. Allow the Selby micropotentiometer to settle and, if necessary, adjust the output of the AC voltage source to achieve the required reading on the DC nanovoltmeter.
10. Calculate the arithmetic mean of 100mV and the DC voltmeter reading noted in step 6, and then increment or decrement this mean as appropriate by the AC-DC difference correction of the Selby micropotentiometer at 100mV and 1kHz. Select the **Set** option from the 4920M's **CAL** menu and enter a value which is equal to this incremented or decremented mean.
11. Execute the calibration of the 4920M by pressing the **Caltrig** front panel key.

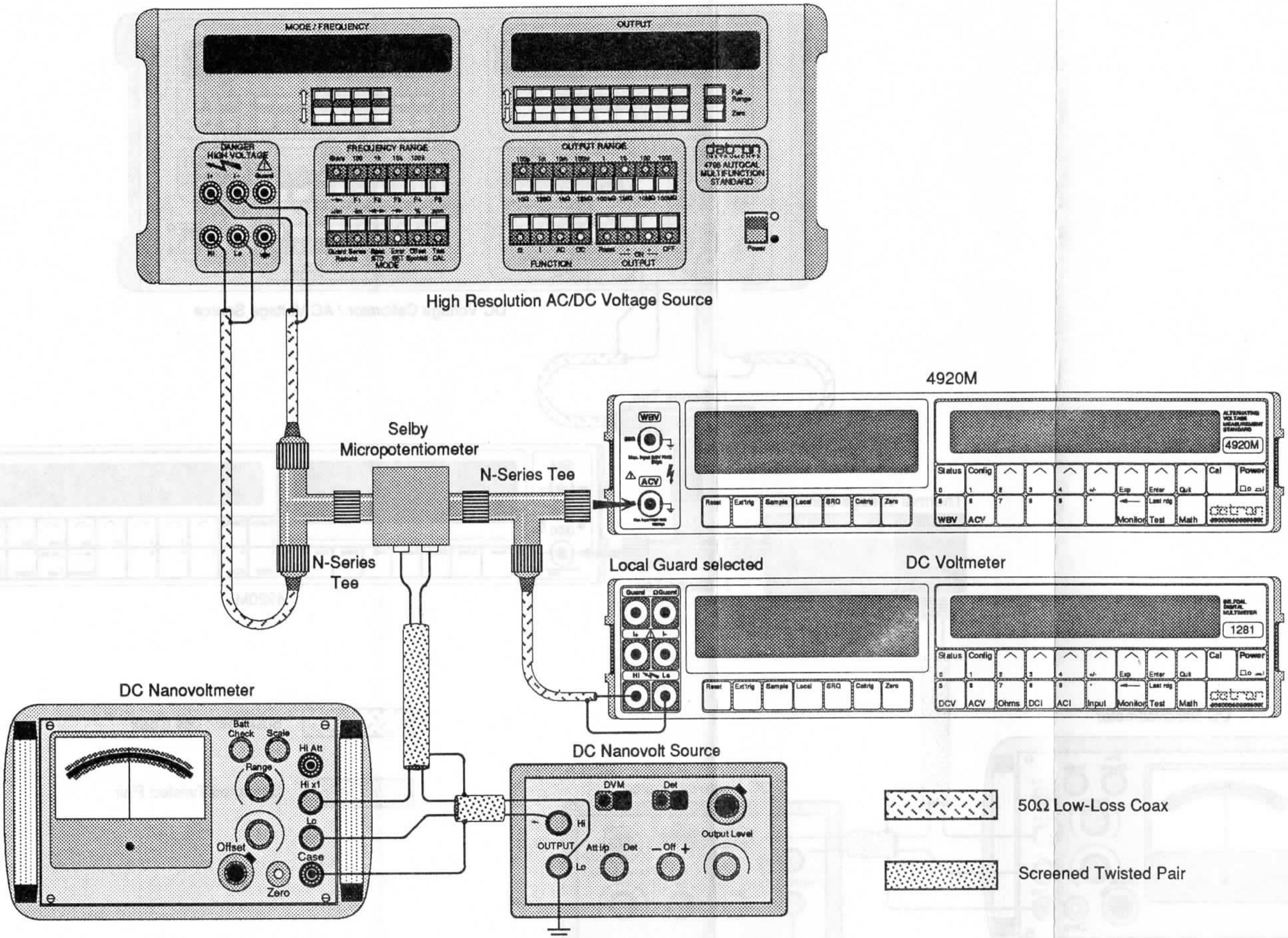



*FIG. 8.1 ACV 100mV CALIBRATION - INTERCONNECTIONS*

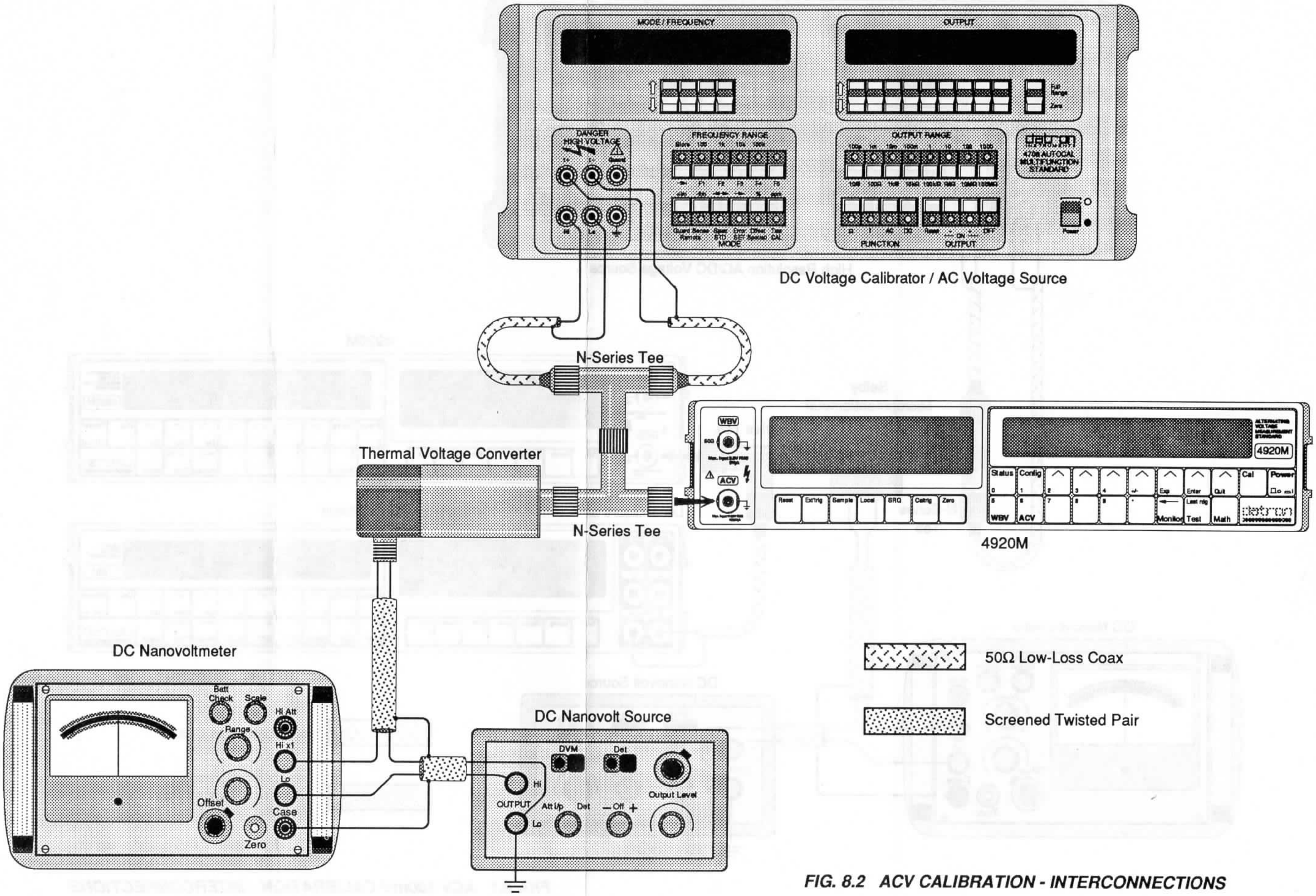



*FIG. 8.2 ACV CALIBRATION - INTERCONNECTIONS*

For each of the calibration voltages and frequencies detailed in Table 8.1, carry out the following procedure:-

1. Configure a thermal voltage converter for the required calibration voltage.
2. Connect the DC voltage calibrator into the calibration setup as shown in Figure 8.2 and select the 4920M's ACV function.
3. Set the output of the DC voltage calibrator to the required calibration voltage and turn the output of the calibrator on.
4. Allow the thermal voltage converter to settle.
5. Adjust the DC nV source to achieve a null on the DC nanovoltmeter.
6. Reverse the polarity of the DC voltage calibrator output, allow the thermal voltage converter to settle, and note the DC reversal error on the nanovoltmeter. Adjust the nV source until the nanovoltmeter shows half of the DC reversal error.
7. Turn the output of the calibrator off and disconnect it by splitting the two precision N-series tees.
8. Connect the AC voltage source in place of the DC voltage calibrator. (Note: if the DC calibrator and AC voltage source are combined into a single unit, the output of this unit can simply be switched from DC to AC.)
9. Set the AC voltage source to the required calibration voltage and frequency and turn its output on.
10. Increment or decrement the AC source output voltage to achieve a null on the DC nanovoltmeter. Allow the thermal voltage converter to settle, and check and adjust the null.
11. Select the **Set** option from the 4920M's **CAL** menu and enter a value which is equal to the calibration voltage selected from Table 8.1, incremented or decremented as appropriate by the AC-DC difference correction of the thermal voltage converter at the calibration voltage and frequency.
12. Execute the calibration of the 4920M by pressing the **Caltrig** front panel key.

## TABLE 8.1

| 4920M ACV Range | CALIBRATION Voltage | CALIBRATION Frequency |
|---|---|---|
| 0.3V | 300.0000mV | 1kHz |
| 0.3V | 300.0000mV | 1MHz |
| 1V | 0.300000V | 1kHz |
| 1V | 1.000000V | 1kHz |
| 1V | 1.000000V | 1MHz |
| 3V | 1.000000V | 1kHz |
| 3V | 3.000000V | 1kHz |
| 3V | 3.000000V | 1MHz |
| 10V | 3.00000V | 1kHz |
| 10V | 10.00000V | 1kHz |
| 10V | 10.00000V | 1MHz |
| 30V | 10.00000V | 1kHz |
| 30V | 30.00000V | 1kHz |
| 30V | 20.00000V | 1MHz |
| 100V | 30.0000V | 1kHz |
| 100V | 100.0000V | 1kHz |
| 100V | 20.0000V | 1MHz |
| 300V | 100.0000V | 1kHz |
| 300V | 300.0000V | 1kHz |
| 300V | 300.0000V | 100kHz |
| 1kV | 300.000V | 1kHz |
| 1kV | 1000.000V | 1kHz |
| 1kV | 700.000V | 100kHz |

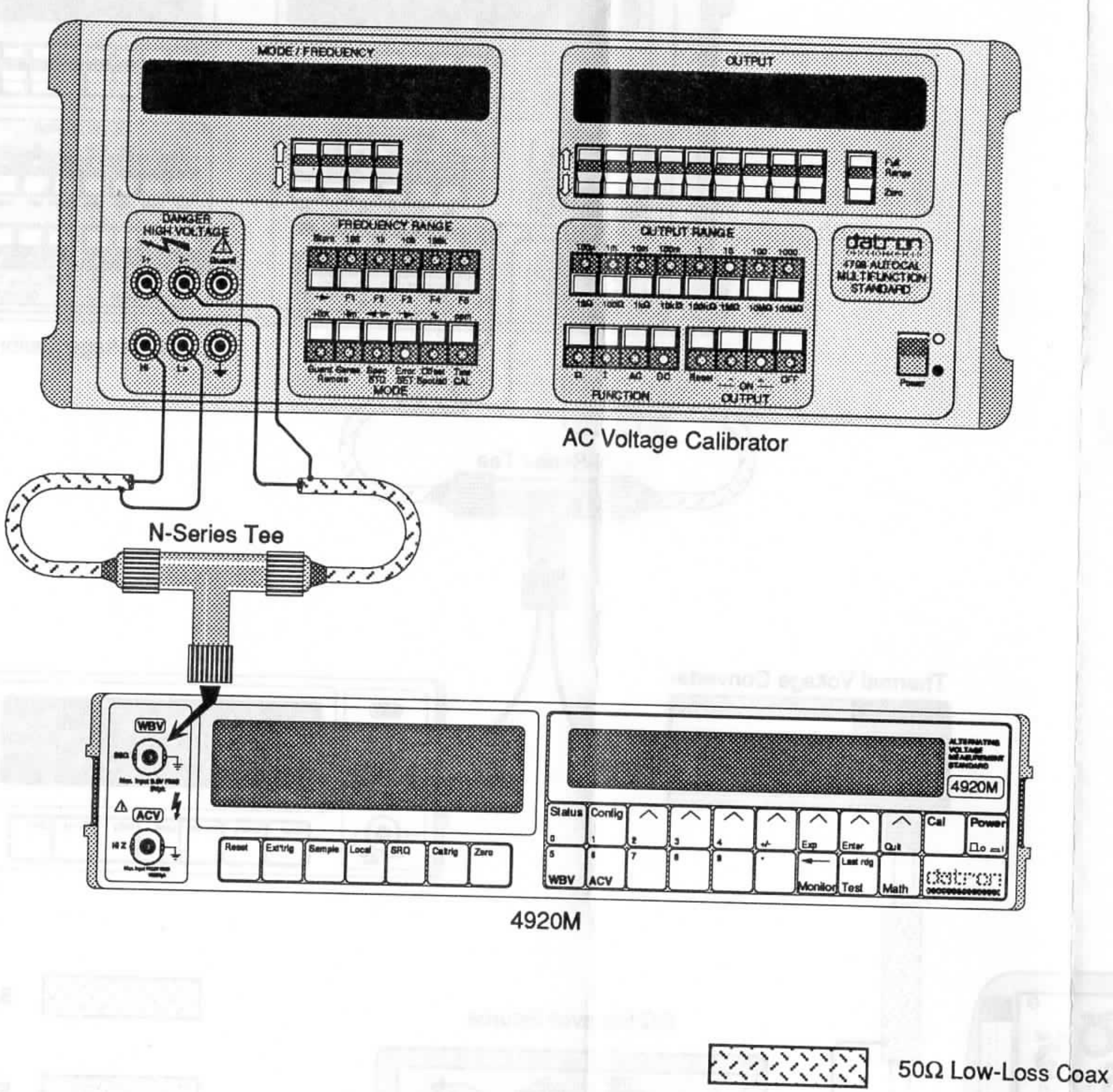



*FIG. 8.3 WBV LF CALIBRATION - INTERCONNECTIONS*

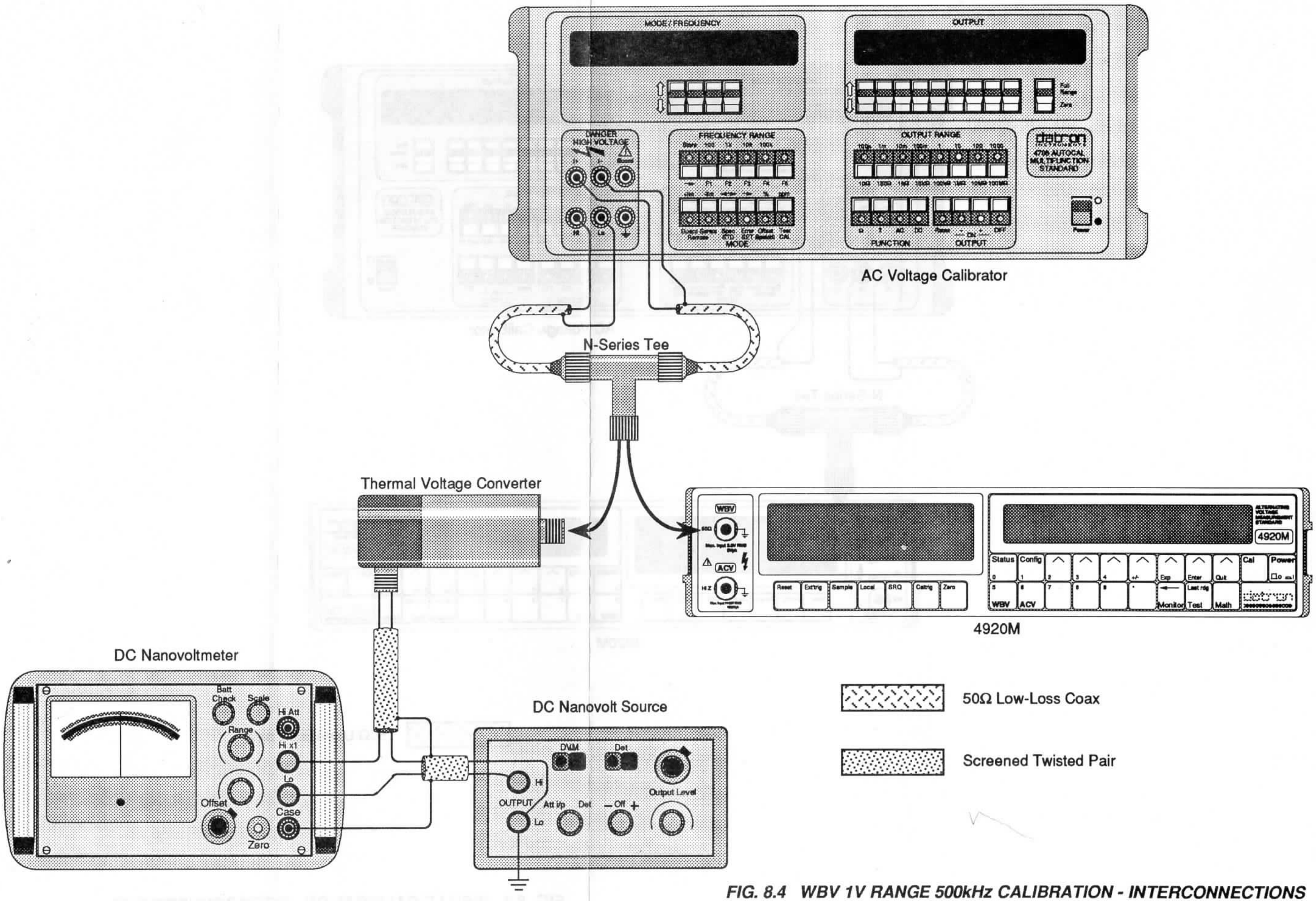



*FIG. 8.4 WBV 1V RANGE 500kHz CALIBRATION - INTERCONNECTIONS*

# WBV (Extended Band) Calibration

For each of the calibration voltages and frequencies detailed in Table 8.2, carry out the following procedure:-

1. Connect the AC voltage calibrator to the 4920M as shown in Figure 8.3 and select the 4920M's WBV function.
2. Set the output of the AC voltage calibrator to the required calibration voltage and frequency and turn the output of the calibrator on.
3. Execute the calibration of the 4920M by pressing the **Caltrig** front panel key.

To perform the high-frequency calibration point for the WBV 1V range, carry out the following procedure:-

1. Configure a 50Ω thermal voltage converter for 1V operation.
2. Connect the thermal voltage converter to the AC voltage calibrator as shown in Figure 8.4.
3. Set the calibrator output to 1.000000V at 1kHz and turn its output on.
4. Allow the thermal voltage converter to settle and adjust the nV source to achieve a null on the DC nanovoltmeter.
5. Set the calibrator output to 1.000000V at 500kHz and turn its output on.
6. Increment or decrement the output of the AC voltage calibrator to achieve a null on the DC nanovoltmeter. Allow the thermal voltage converter to settle and, if necessary, check and adjust the null.
7. Turn the output of the AC voltage calibrator off and transfer its output from the thermal voltage converter to the WBV input of the 4920M. Select the 4920M's WBV function and 1V range, and turn the output of the calibrator back on.
8. Select the **Set** option from the 4920M's **CAL** menu and enter a value which is equal to 1.000000V plus or minus (as appropriate) the AC-AC flatness error of the thermal voltage converter between 1kHz and 500kHz.
9. Execute the calibration of the 4920M by pressing the **Caltrig** front panel key.

**TABLE 8.2**

| 4920M | CALIBRATION | |
|---|---|---|
| WBV Range | Voltage | Frequency |
| 1V | 0.100000V* | 1kHz |
| 1V | 1.000000V | 1kHz |
| 3V | 1.000000V | 1kHz |
| 3V | 3.000000V | 1kHz |

* If a Datron Model 4708 calibrator is used, this calibration voltage should be generated by setting the calibrator to 0.100000V on its 1V range.

## Calibration at other than Nominal Values

Calibration can be carried out at voltages close to (but not exactly at) the nominal values detailed in the calibration procedures provided that the **Set** option is selected from the 4920M's **CAL** menu, and the known calibration point voltage is input via the 4920M's numeric keys and entered with the **Enter** key immediately before pressing the **Caltrig** key.

## Setting a New Calibration Due Date

After completing calibration of the 4920M carry out the following procedure to set the new calibration date:-

1. Select the **Date** menu option from the **CAL** menu. Enter the current date and time using the front-panel numeric keys and the **Enter** key. (Use the **Quit** menu option if the display already shows the correct date and time.)
2. Select the **Due** menu option from the **CAL** menu and then select the **Intvl** (Interval) menu option from the **CAL DUE** menu. Enter the required calibration interval in days using the front-panel numeric keys and the **Enter** key. (Use the **Quit** menu option if the display already shows the correct calibration interval.)
3. Select the **New** menu option from the **CAL DUE** menu. The 4920M should now display a date which is one calibration interval ahead of the current date.
4. Return the rear-panel **CALIBRATION** switch to the **DISABLE** position and carry out the performance verification procedure detailed in section 7 of this manual.

# Subject Index



*An Index of IEEE 488.2 Syntax Diagrams is given in Section 5, page 5-0-2.*

*An Index of IEEE 488.2 Syntax Diagrams is given in Section 5, page 5-0-2.*

*An Index of IEEE 488.2 Syntax Diagrams is given in Section 5, page 5-0-2.*

*An Index of IEEE 488.2 Syntax Diagrams is given in Section 5, page 5-0-2.*