Filter488

Programmable Analog Filter

For the Filter488/4 and Filter488/8

Note: Filter488 is no longer a stocked item.

USER'S MANUAL

Part #178-0920
Revision 1.2
November 1992

IOtech, Inc. • 25971 Cannon Road • Cleveland, Ohio 44146 (440) 439-4091

Table of Contents

Table of Contents

Introduction

1.1 General Description

The Filter488 is a programmable analog filter that offers Butterworth, Bessel, elliptic (Cauer), and Chebyshev filter response options. The Filter488/4 has four differential input channels. The Filter488/8 has eight differential input channels. Each channel can be configured for a different filter response.

The low-pass characteristic, with a rolloff exceeding 72 dB/octave, makes Filter488 ideal as an anti-aliasing filter in front of analog-to-digital converters. It is useful in situations where analog input signals with frequency content greater than one half the sampling rate of the A/D converter can cause significant errors (due to aliasing). The cut-off frequency of each of the channels is digitally controlled via IEEE 488 or RS-232C, providing easy, industry-standard instrument/controller interfacing. Differential analog inputs up to ±10 volts are accommodated on each channel via BNC style connectors located on the rear of the enclosure. Single-ended outputs are provided on a DB-25 connector, also on the rear panel.

Filter488 provides multiple-pole, lowpass filter functions with cut-off frequencies between 1Hz and 50kHz. In addition to filter cut-off frequency, each channel may be programmed for full scale input ranges between 10mV and 10V in a "1, 2, 5" sequence and for output ranges of 1, 2, 5, and 10V. Each channel input is true differential and may be AC or DC coupled also under program control.

The filter function in each channel may be bypassed so that individual channels can be used for amplification purposes only. Filter488 is packaged in a standard 2U (3.5") high, 19-inch rack enclosure. An IEEE 488 standard connector is provided for bus communication and an RS-232C interface is available through a DB-9S connector. Analog channel inputs are through BNC connector pairs and analog outputs are through a DB-25S connector.

Filter488's lowpass filter options eliminate errors due to aliasing in data acquisition applications. The types of errors eliminated include noise in the form of high frequency signal components. These random signals are the source of significant errors. The lowpass filter options limit the input signal bandwidth below one-half of the A/D sample rate.

All Filter488 functions, including input and output voltage range, AC/DC coupling and cut-off frequency, can be programmed from its standard IEEE 488 or RS-232C interface. The Filter488's IEEE 488 base address is specified from 0 to 30 via rear panel switches. The RS-232C rear-panel connector is a 9-pin D-sub male. Baud rates, parity, and data flow are selected via rear panel switches.

A DB-25 connector provides convenient access to all Filter488 (single-ended) analog outputs. Filter488 outputs may be connected to the ADC488/16A and the ADC488/8SA with the optional CA-96 cable.

All analog circuitry is optically isolated from the digital control logic, the communication buses, and the AC power supply line by up to 50 V common mode. This isolation reduces the risk of damage to both the Filter488 and the host controller by signals with ground differences. It also isolates the analog circuitry from the digital noise created by the host controller and communication buses.

Filter488 has circuitry and firmware commands to perform offset nulling under software control. When Filter488 is used with an A/D converter, such as IOtech's ADC488/16A, offset nulling can be performed as a closed loop operation using the A/D converter to measure and report the amount of offset at the channel outputs of Filter488 to the host controller.

Filter488 can be internally configured to support either 90-125 V ac or 210-250 V ac power. The current Filter488 configuration may be saved in non-volatile memory so that the filter will be properly configured at power up.

1.2 Filter488 Modules

Filter488's elliptic, Butterworth, and Chebyshev filters are ideally suited for frequency-domain, anti-aliasing data acquisition applications that are amplitude-critical. Filter488 also features a lowpass Bessel filter that is best suited for time-domain applications where phase and delay are critical factors.

The Bessel filter module provides constant time delay with excellent phase linearity, but has poor amplitude response and the slowest roll-off in the passband. It is best used for time-domain applications requiring faithful but delayed pulse reproduction and good transient response applications with complex signals with multiple frequencies.

The Butterworth filter module has a wide and attenuation-free passband (maximally flat) with monatomic roll-off in the stopband that is more gradual than the elliptic filter. It has a non-linear phase response and step response overshoot. It is intended for frequency-domain, anti-aliasing applications requiring excellent frequency response and amplitude accuracy. The Butterworth filter is used when there is significant difference between the Nyquist frequency and the highest frequency of the desired signal.

The Chebyshev filter module is similar to Butterworth, but has faster roll-off closer to the cut-off frequency and a square amplitude response. It has poor phase and time delay with some passband ripple. It can be used in applications similar to the Butterworth's with a need for increased uniformity around the passband.

The elliptic filter module has the steepest stopband roll-off and the widest passband. It has non-uniform phase response and some degree of passband and stopband ripple. The elliptic filter is best used for frequency-domain, anti-aliasing applications requiring excellent frequency response and amplitude accuracy. It is generally used when there is little difference between the Nyquist frequency and the highest frequency of the desired signal.

1.3 Ordering Information

Additional accessories that can be ordered for the Filter488 include:

CA-7-1	1.5 foot IEEE 488 cable
CA-7-2	6 foot IEEE 488 cable
CA-7-3	6 foot shielded IEEE 488 cable
CA-7-4	6 foot reverse entry IEEE 488 cable
CA-96	Filter488 to ADC488/16A or ADC488/8SA cable
Filter488-901	Additional user's manual
FL1	Four channel 8-pole Butterworth lowpass filter module
FL2	Four channel 8-pole elliptic lowpass filter module
FL3	Four channel 8-pole Bessel lowpass filter module
FL4	Four channel 8-pole Chebyshev lowpass filter module

1.4 Specifications

GENERAL

Line Voltage: 100-125/210-240vac 50/60Hz
Current Draw: 0.5 A at 110VAC typ.
Number of Channels: 4 or 8
Input Type: Differential >60 dB common mode rejection.
Total Harmonic Distortion: <0.1%
Passband Gain: 0 dB +/- 0.1 dB (1Hz-20KHz) +/- 0.5 dB (20KHz-40KHz)
Signal/Noise: >60 dB FS
Output DC offset: Programmable
Thermal Drift: 100 uv/C
Phase difference between channels: +/- 2 deg. for channels with same cut-off frequency
IEEE 488 Implementation: TI 9914
AC/DC coupling: Software selectable
Gain: Software selectable X.1 to X1000

Cut-off Frequency: Software selectable

Operating Frequency: dc to 50 KHz

System Grounds: Digital and chassis

Ambient Operating Temperature: 0°C to 50°C

Power Requirements: 90-125 or 210-250 VAC, 50/60 Hz; 20 VA max

Dimensions: 425 mm wide x 305 mm deep x 90 mm high (16.75" x 12" x 3.5")

Supplied Accessories: Mating 9-pin D-sub female connector, manual, and rack-mount box

FILTER CHARACTERISTICS

Functions: lowpass

Response: Butterworth, elliptic, Chebyshev, or Bessel

Tunable Cut-off Frequency Range: 1 Hz-50 kHz

Frequency Accuracy: ±1% @ 3 dB

Stopband Attenuation: 130 dB (elliptic), 48 dB (Butterworth), 80 dB (Chebyshev), 40 dB (Bessel)

Passband Attenuation: 0 dB±0.1 dB

Harmonic Distortion: 0.15% max.

Phase Match: ±2° (filter to filter) @ 1 kHz

INPUT CHARACTERISTICS

Input Channels: 4 or 8 differential

Range: ±10mV to ±10V

Input Gain: x1, x10, x100

Coupling: AC or DC

Mode: Filter or bypass

Impedance: >1mΩ (single-ended)

CMRR: 60 dB to 1 kHz; 50 dB to 50 kHz

Maximum Signal: ±25V

Maximum DC Component: 100 VDC in AC coupled mode

AC Coupling: -3 dB @ 3 Hz

Frequency Ranges and Steps:

Range (Hz)	Step (Hz)
25,000-50,000	100
10,000-25,000	50
5,000-10,000	20
2,500-5,000	10
1,000-2,500	5
500-1,000	2
250-500	1
100-250	0.5
50-100	0.20
25-50	0.10
10-25	0.05
5-10	0.02
2.5-5	0.01
1-2.5	0.005

OUTPUT CHARACTERISTICS

Output: 4 or 8 single ended
Range: ±1, ±2, ±5, ±10V
Impedance: 100 Ω
Maximum Signal Voltage: ±10V
Maximum Current: 10 mA
DC Offset: Programmable
DC Drift: 100µV/°C

IEEE 488 SPECIFICATIONS

Interface Subsets: SH1, AH1, T4, TE0, L4, LE0, SR1, PP0, RL0, DC1, DT0, C0, E1
Connector: Standard IEEE 488 connector with metric studs

RS-232C SPECIFICATIONS

Electrical Characteristics: Supports RS-232C

Character Set: Asynchronous bit serial

Duplex: Full

Word Length: 7 or 8 data bits

Stop Bits: One

Parity: Odd, even, none

Baud Rates: 300, 1200, 2400, 4800, 9600

Terminator: CR, LF or both

Control: Supports Data Terminal Ready (DTR), Data Set Ready (DSR) or XON/XOFF

Output Voltage: ±5V min (RS-232C)

Input Voltage: ±3.0V min; ±15V max

Connector: 9-pin D-sub male

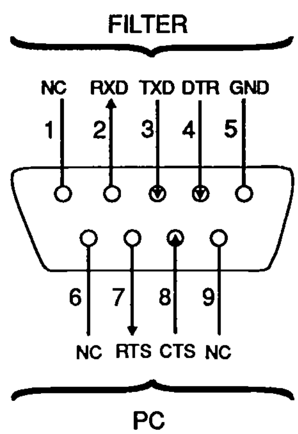


Figure 1.1 RS-232 Connector Pinouts

1.4.1 Filter Specifications for Butterworth

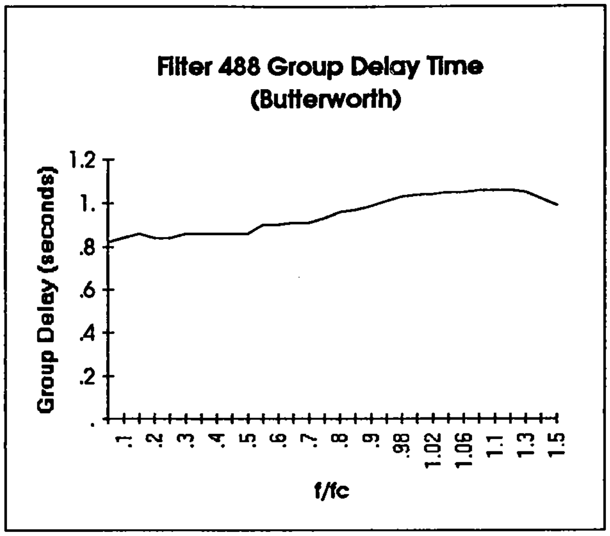
The following data has been collected for the analog performance of Filter488 with the Butterworth (LTC1064-2) filter installed:

FREQUENCY (f/fc)	DELAY (mSec)	PHASE (deg)
.05	.82	-14.76
.1	.84	-30.24
.15	.86	-46.44
.2	.84	-60.48
.25	.84	-75.6
.3	.86	-92.88
.35	.86	-108.36
.4	.86	-123.84
.45	.86	-139.32
.5	.86	-154.8
.55	.9	-178.2
.6	.9	-194.4
.65	.91	-212.94
.7	.91	-229.32
.75	.93	-251.1
.8	.958	-275.904
.85	.968	-296.208
.9	.986	-319.464
.96	1.008	-348.3648
.98	1.028	-362.6784
1.	1.038	-373.68
1.02	1.042	-382.6224
1.04	1.049	-392.7456
1.06	1.05	-400.68
1.08	1.06	-412.128
1.1	1.06	-419.76
1.2	1.06	-457.92
1.3	1.05	-491.4
1.4	1.02	-514.08
1.5	.988	-533.52

Where f/fc is the test frequency (f) divided by the programmed cut off frequency (fc).

Butterworth Filter, fc = 1000Hz

The collected data is used in the following chart:



Signal to noise ratio and distortion are as follows:

INPUT RANGE V (full scale)	OUTPUT RANGE V (full scale)	DISTORTION (%)	SIGNAL TO NOISE (db)
10	10	0.0470	78.5
5	5	0.0490	82.6
2	2	0.0462	82.3
1	1	0.0447	79.8

With a cut-off frequency of 1000 Hz the stopband attenuation rate is 48 db per octave. The noise floor is about 1 mV at 10 V full scale or about 80 db below the input signal at full scale.

1.4.2 Filter Specifications for Chebyshev

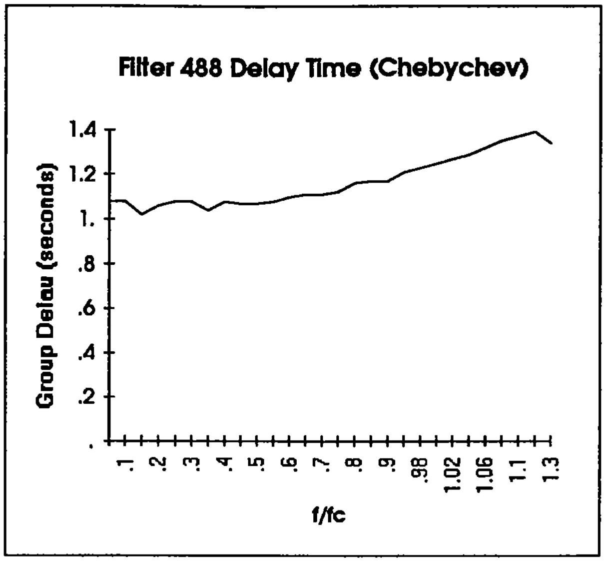
The following data has been collected for the analog performance of Filter488 with the Chebyshev (LTC1064) filter installed:

FREQUENCY (f/fc)	DELAY (mSec)	PHASE (deg)
.05	1.08	-19.44
.1	1.08	-38.88
.15	1.02	-55.08
.2	1.06	-76.32
.25	1.08	-97.2
.3	1.08	-116.64
.35	1.04	-131.04
.4	1.08	-155.52
.45	1.07	-173.34
.5	1.07	-192.6
.55	1.08	-213.84
.6	1.1	-237.6
.65	1.11	-259.74
.7	1.11	-279.72
.75	1.12	-302.4
.8	1.16	-334.08
.85	1.17	-358.02
.9	1.17	-379.08
.96	1.21	-418.176
.98	1.23	-433.944
1.	1.25	-450
1.02	1.27	-466.344
1.04	1.29	-482.976
1.06	1.32	-503.712
1.08	1.35	-524.88
1.1	1.37	-542.52
1.2	1.39	-600.48
1.3	1.34	-627.12

Where f/fc is the test frequency (f) divided by the programmed cut-off frequency (fc).

Chebychev Filter, fc = 1000Hz

The collected data is used in the following chart:



Signal to noise ratio and distortion are as follows:

INPUT RANGE V (full scale)	OUTPUT RANGE V (full scale)	DISTORTION (%)	SIGNAL TO NOISE (db)
10	10	0.0285	75.50
5	5	0.0632	80.03
2	2	0.0640	79.25
1	1	0.0640	77.43

With a cut-off frequency of 1000 Hz the stopband attenuation rate is 80 db per octave. The noise floor is about 1 mV at 10 V full scale or about 80 db below the input signal at full scale.

1.4.3 Filter Specifications for Elliptic

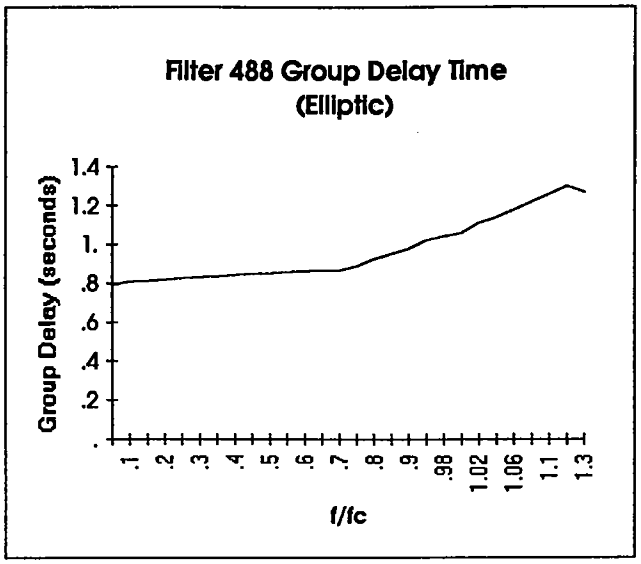
The following data has been collected for the analog performance of Filter488 with the elliptic (LTC1064-1) filter installed:

FREQUENCY (f/fc)	DELAY (mSec)	PHASE (deg)
.05	.79	-14.22
.1	.81	-29.16
.15	.814	-43.956
.2	.82	-59.04
.25	.825	-74.25
.3	.831	-89.749
.35	.836	-105.336
.4	.842	-121.248
.45	.847	-137.214
.5	.85	-153
.55	.853	-168.894
.6	.858	-185.328
.65	.86	-201.24
.7	.862	-217.224
.75	.884	-238.68
.8	.92	-264.96
.85	.948	-290.088
.9	.976	-316.224
.96	1.02	-352.512
.98	1.04	-366.912
1.	1.06	-381.6
1.02	1.11	-407.592
1.04	1.14	-426.816
1.06	1.18	-450.288
1.08	1.22	-474.336
1.1	1.26	-498.96
1.2	1.3	-561.6
1.3	1.27	-594.36

Where f/fc is the test frequency (f) divided by the programmed cut-off frequency (fc).

Elliptic filter, fc = 1000 Hz.

The collected data is used in the following chart:



Signal to noise ratio and distortion are as follows:

INPUT RANGE V (full scale)	OUTPUT RANGE V (full scale)	DISTORTION (%)	SIGNAL TO NOISE (db)
10	10	0.07	78
5	5	0.06	82
2	2	0.05	82
1	1	0.05	80

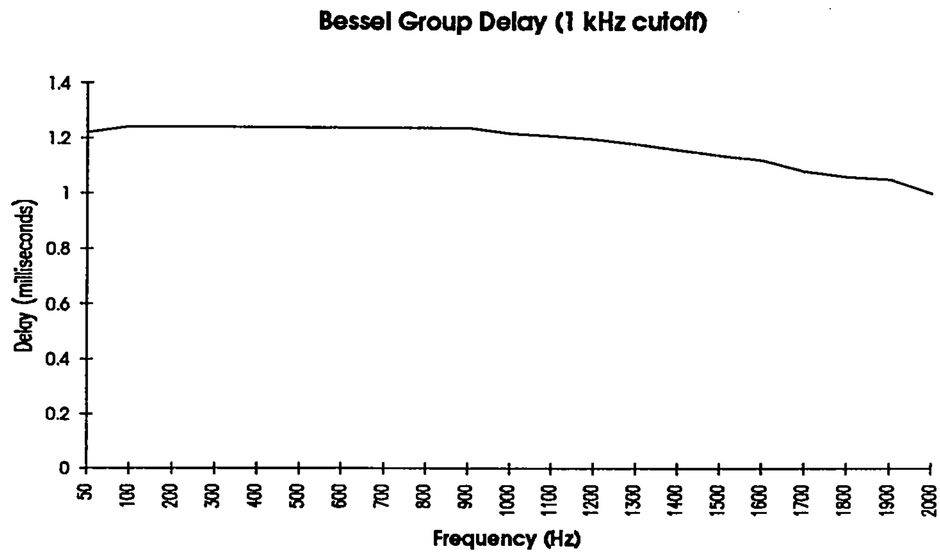
With a cut-off frequency of 1000 Hz the stopband attenuation rate is 140 db per octave. The noise floor is about 1 mV at 10 V full scale or about 80 db below the input signal at full scale.

1.4.4 Filter Specifications for Bessel

The following data has been collected for the analog performance of Filter488 with the Bessel (LTC1064-7) filter installed. In order to more accurately report this filter's response, test data was gathered for both a 1kHz cut-off and a 2kHz cut-off.

GROUP DELAY (1kHz cut-off)		
FREQUENCY (Hz)	DELAY (mSec)	PHASE (deg)
50	1.22	22
100	1.24	44.6
200	1.24	89.3
300	1.24	133.9
400	1.24	178.6
500	1.24	223.2
600	1.24	267.8
700	1.24	312.5
800	1.24	357.1
900	1.24	401.8
1000	1.22	439.2
1100	1.21	479.2
1200	1.2	518.4
1300	1.18	552.2
1400	1.16	584.6
1500	1.14	615.6
1600	1.12	645.1
1700	1.08	661
1800	1.06	686.9
1900	1.05	718.2
2000	1	720

The collected data is used in the following chart:



Signal to noise ratio and distortion are as follows:

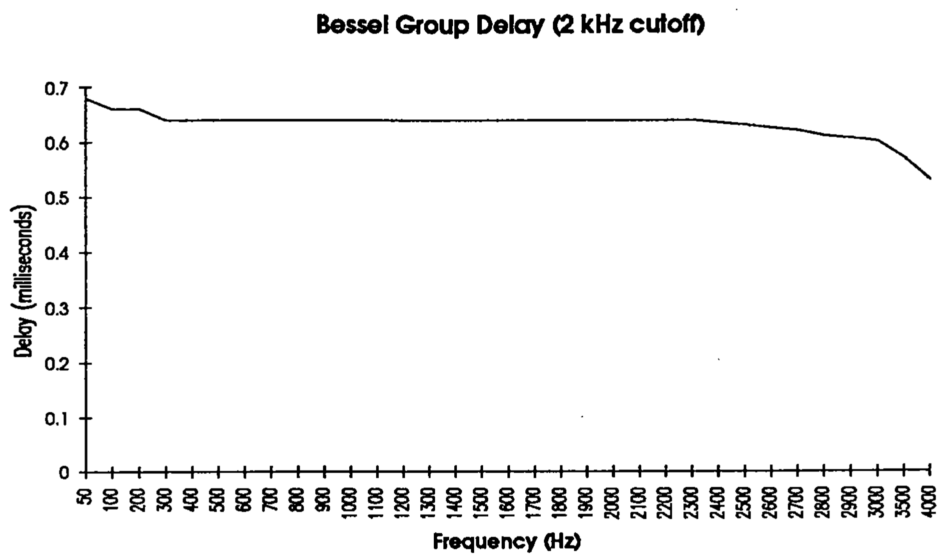
INPUT RANGE V (full scale)	OUTPUT RANGE V (full scale)	DISTORTION (%)	SIGNAL TO NOISE (db)
10	10	0.0149	78.9
5	5	0.0134	83.3
2	2	0.0142	82.19
1	1	0.0163	78.52

With a cut-off frequency of 1000Hz the stopband attenuation rate is 35 dB per octave. The noise floor is about 1 mV at 10V full scale or about 70 dB below the input signal at full scale.

The following data has been collected for the analog performance of Filter488 with the Bessel (LTC1064-7) filter installed with a 2kHz cut-off.

GROUP DELAY (2kHz cut-off)		
FREQUENCY (Hz)	**DELAY (mSec)**	**PHASE (deg)**
50	0.68	12.2
100	0.66	23.8
200	0.66	47.5
300	0.64	69.1
400	0.64	92.2
500	0.64	115.2
600	0.64	138.2
700	0.64	161.3
800	0.64	184.3
900	0.64	207.4
1000	0.64	230.4
1100	0.64	253.4
1200	0.64	276.5
1300	0.64	299.5
1400	0.64	322.6
1500	0.64	345.6
1600	0.64	368.6
1700	0.64	391.7
1800	0.64	414.7
1900	0.64	437.8
2000	0.64	460.8
2100	0.66	483.8
2200	0.66	506.9
2300	0.64	529.9
2400	0.635	548.6
2500	0.63	567
2600	0.625	585
2700	0.62	602.6
2800	0.61	614.9
2900	0.605	631.6
3000	0.6	648
3500	0.57	718.2
4000	0.53	763.2

The collected Bessel data is used in the following chart:



Signal to noise ratio and distortion are as follows:

INPUT RANGE V (full scale)	OUTPUT RANGE V (full scale)	DISTORTION (%)	SIGNAL TO NOISE (db)
10	10	0.0149	78.9
5	5	0.0134	83.3
2	2	0.0142	82.19
1	1	0.0163	78.52

With a cut-off frequency of 1000Hz the stopband attenuation rate is 35 dB per octave. The noise floor is about 1 mV at 10V full scale or about 70 dB below the input signal at full scale.

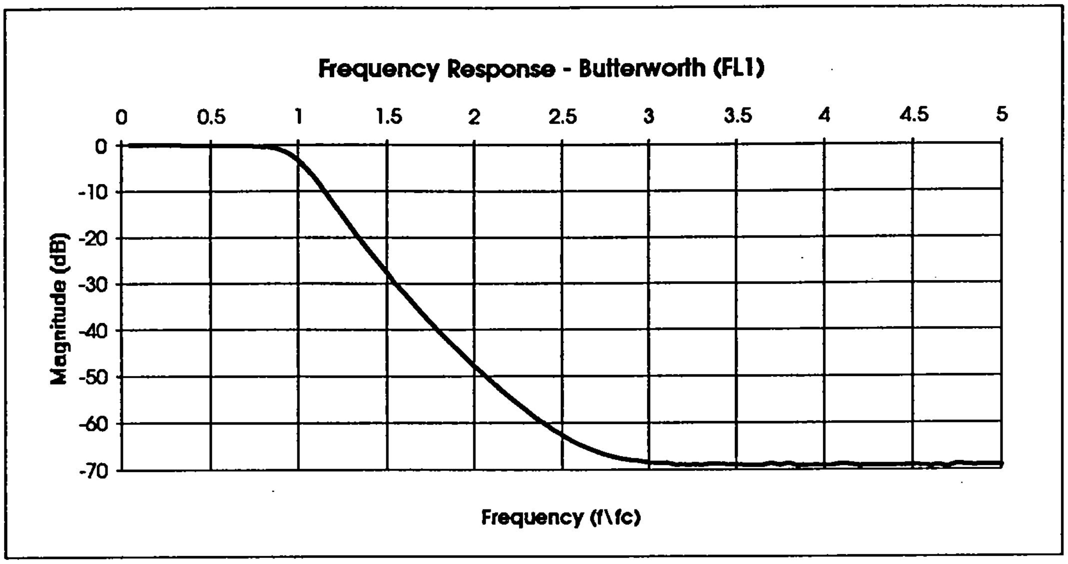
1.4.5 Filter Characterization

The following information has been collected for the Butterworth (FL1) Filter:

FREQUENCY (f/fc)	MAGNITUDE (dB)
.05	0
.1	0
.15	0
.2	-0.01449
.25	-0.01449
.3	-2.90E-02
.35	-4.35E-02
.4	-5.81E-02
.45	-0.07269
.5	-8.73E-02
.55	-0.10193
.6	-0.11659
.65	-0.14598
.7	-0.17548
.75	-0.21991
.8	-0.33951
.85	-0.5683
.9	-1.01219
.95	-1.86611
1	-3.244
1.05	-5.19271
1.1	-7.47657
1.15	-10.0062
1.2	-12.6466
1.25	-15.2978
1.3	-17.9695
1.35	-20.4912
1.4	-22.934
1.45	-25.2985
1.5	-27.6633

Where f/fc is the test frequency (f) divided by the programmed cut-off frequency (fc). Butterworth filter, fc = 2000 Hz.

For more information on additional values which were not included in the above chart, contact IOtech.



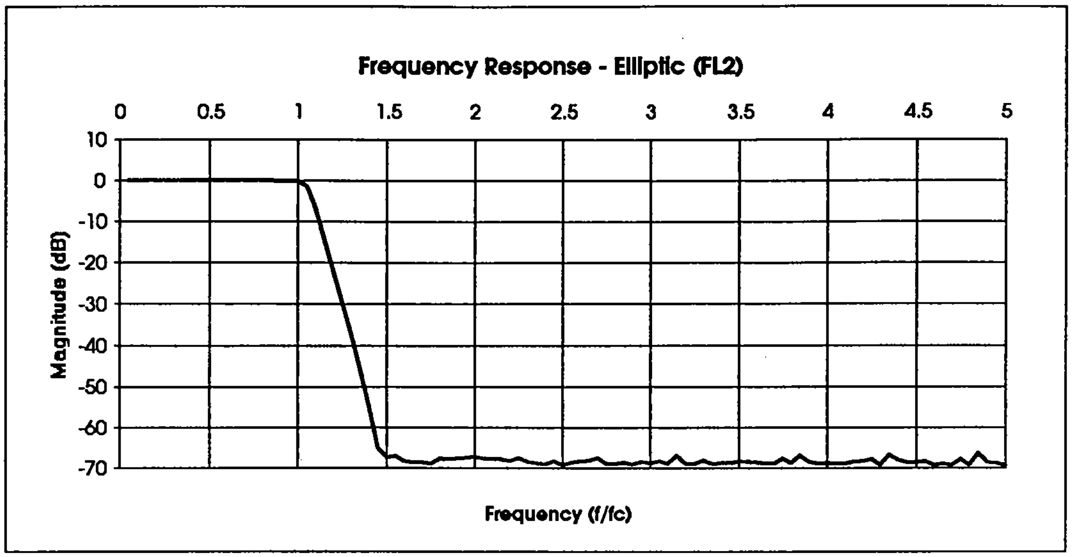
The 8-pole low-pass Butterworth filter (FL1) transfer function is maximally flat in the passband with a monotonic roll-off rate of 48 dB/octave to the stopband.

The following data has been collected for the Elliptic (FL2) Filter:

FREQUENCY (f/fc)	MAGNITUDE (dB)
.05	0.028905
.1	1.45E-02
.15	1.45E-02
.2	1.45E-02
.25	1.45E-02
.3	1.45E-02
.35	1.45E-02
.4	0.028905
.45	4.33E-02
.5	5.77E-02
.55	8.64E-02
.6	0.100748
.65	0.100748
.7	8.64E-02
.75	4.33E-02
.8	0
.85	-4.35E-02
.9	-0.07269
.95	-8.73E-02
1	-0.13127
1.05	-1.36067
1.1	-6.558
1.15	-14.2813
1.2	-22.0657
1.25	-29.7044
1.3	-37.5336
1.35	-45.5142
1.4	-54.6484
1.45	-65.1661
1.5	-67.3876

Where f/fc is the test frequency (f) divided by the programmed cut-off frequency (fc). Elliptic filter, fc = 2000 Hz.

For more information on additional values which were not included in the above chart, contact IOtech.



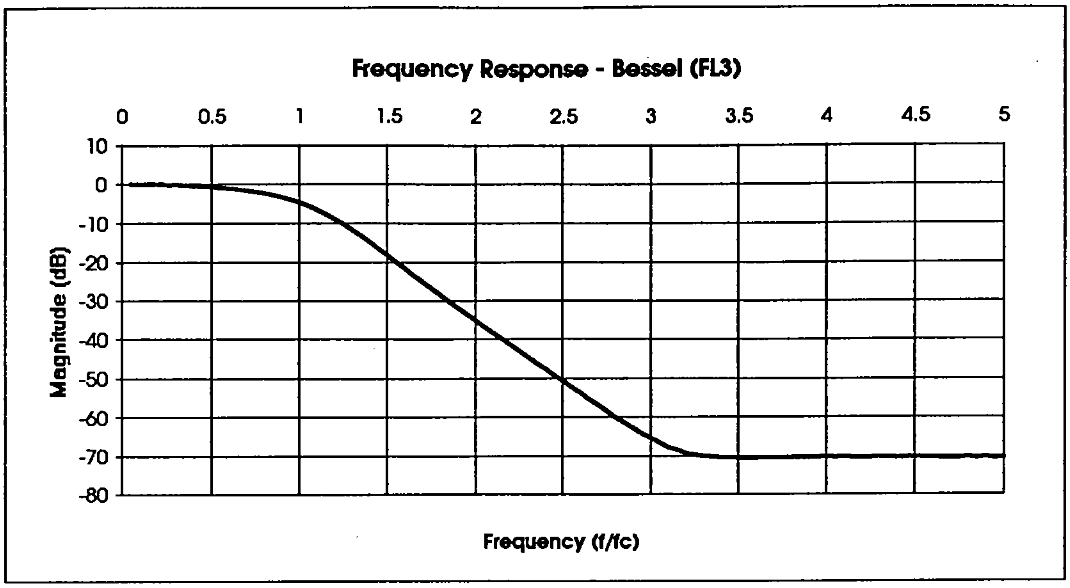
The 8-pole low-pass Elliptic filter (FL2) transfer function has a wide passband with a passband ripple of <.1 dB and a steep (140 dB/octave) roll-off to the stopband.

The following data has been collected for the Bessel (FL3) Filter:

FREQUENCY (f/fc)	MAGNITUDE (dB)
.05	0.028905
.1	-0.01449
.15	-4.35E-02
.2	-0.10193
.25	-0.17548
.3	-0.24965
.35	-0.35457
.4	-0.47606
.45	-0.59926
.5	-0.75577
.55	-0.93124
.6	-1.12681
.65	-1.34375
.7	-1.61843
.75	-1.92011
.8	-2.30785
.85	-2.75338
.9	-3.26506
.95	-3.92173
1	-4.65685
1.05	-5.5418
1.1	-6.59504
1.15	-7.67745
1.2	-8.93037
1.25	-10.2808
1.3	-11.7339
1.35	-13.2441
1.4	-14.8623
1.45	-16.4781
1.5	-18.1548

Where f/fc is the test frequency (f) divided by the programmed cut-off frequency (fc).
Bessel filter, fc = 2000 Hz.

For more information on additional values which were not included in the above chart, contact IOtech.



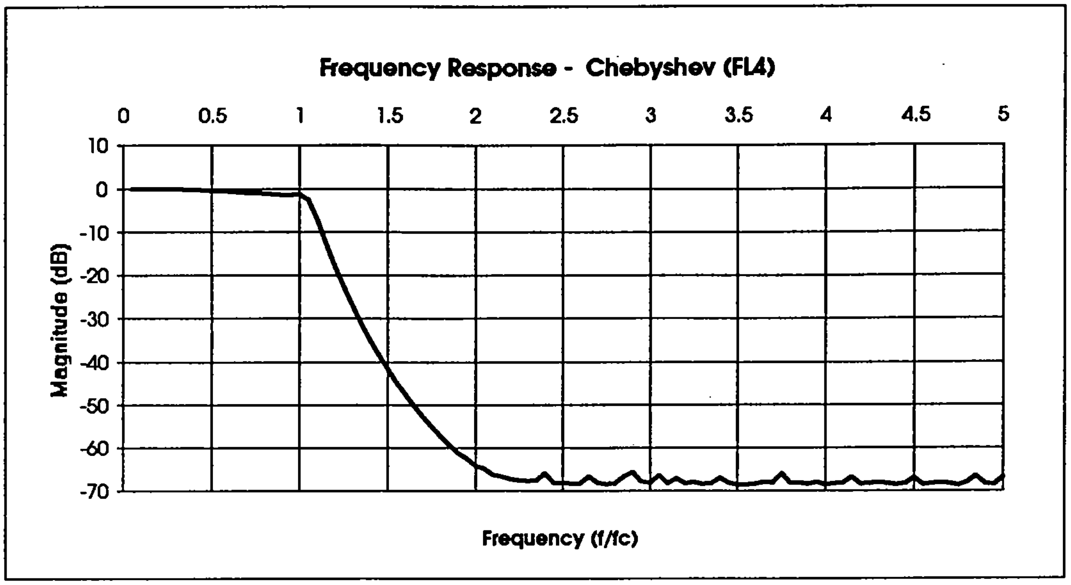
The 8-pole low-pass Bessel filter (FL3) transfer function has a monotonic passband with a slow (35 dB/octave) roll-off to the stopband.

The following data has been collected for the Chebyshev (FL4) Filter:

FREQUENCY (f/fc)	MAGNITUDE (dB)
.05	0.028905
.1	0.028905
.15	5.77E-02
.2	5.77E-02
.25	0.028905
.3	-2.90E-02
.35	-0.11659
.4	-0.20507
.45	-0.29446
.5	-0.38479
.55	-0.47606
.6	-0.58376
.65	-0.72424
.7	-0.85104
.75	-0.96353
.8	-1.09391
.85	-1.20961
.9	-1.29321
.95	-1.29321
1	-1.20961
1.05	-2.44105
1.1	-7.10462
1.15	-12.5971
1.2	-18.0155
1.25	-22.8131
1.3	-27.1836
1.35	-31.1189
1.4	-34.8542
1.45	-38.2017
1.5	-41.4455

Where f/fc is the test frequency (f) divided by the programmed cut-off frequency (fc). Chebyshev filter, fc = 2000 Hz.

For more information on additional values which were not included in the above chart, contact IOtech.



The 8-pole low-pass Chebyshev filter (FL4) transfer function has a <.1 dB ripple in the passband with a moderately steep roll-off (80 dB/octave) to the stopband.

Getting Started

The Filter488 manual includes information about both the Filter488/8 and the Filter488/4. "Filter488" refers to either unit. Where information is specific to the Filter488/8 or the Filter488/4, the full name is used.

2.1 Inspection

The Filter488 was carefully inspected mechanically and electrically prior to shipment. When you receive the interface, carefully unpack all items from the shipping carton and check for any obvious signs of physical damage that may have occurred during shipment. Report any such damage found to the shipping agent immediately. Retain all shipping materials in the event that shipment back to the factory is necessary.

Every Filter488 is shipped with the following items:

Filter488/8	Eight channel programmable filter with RS-232C and IEEE 488 interfaces.
or	
Filter488/4	Four channel programmable filter with RS-232C and IEEE 488 interfaces.
FL1, FL2, FL3, and/or FL4	One FLx module for each channel.
Filter488/8-901	User's Manual.
000-0720	Sample program disk in IBM Format.
000-0801	Accessory kit, including:

CA-1	Power cable
FE-1	Rubber Feet (4)
HA-70	Rack Ears (2)
HA-41-6	Screws (4)
FU-1-.5	1/2 Amp Replacement Fuse
FU-1-.25	1/4 Amp Replacement Fuse

WARNING

Filter488 is intended for INDOOR USE ONLY. Failure to observe this warning could result in equipment failure or personal injury.

2.2 Internal Configuration

The internal configuration of a Filter488 consists of setting the line voltage and replacing fuses. Line voltage must be set for 110 or 220 V ac to match the power being supplied to the Filter488. If the line voltage is changed, the fuse must also be changed. See Figure 2.1 for line voltage switch and fuse locations.

WARNING

Disconnect the power cord from the power line and from the Filter488 prior to disassembly.

Never open the Filter488 case while it is connected to the power line. Internal voltage potentials exist that could cause personal injury.

To avoid injury, de-energize attached circuits, then disconnect the Filter488 from other equipment and the power line before changing the internal configuration of the Filter488.

To open the unit, place the Filter488 on a flat surface. Remove the two screws on the top rear of the case. Remove the top cover by sliding it out toward the back of the unit. Reverse this procedure to reassemble the unit.

2.2.1 Line Voltage Selection

The Filter488 can be operated from 90-125 or 210-250 V ac, 50-60 Hz. The interface was shipped from the factory set for the operating voltage marked on the label placed over the rear panel line cord jack. If this setting is not appropriate for the power to be supplied to the unit, the setting of the internal voltage switch (S3) and the power fuse must be changed to avoid damage to the unit. The locations of S3 and the fuse are shown in Figure 2.1.

CAUTION

A fuse with a rating higher than that specified may cause damage to the instrument and may pose a fire hazard. If the instrument repeatedly blows fuses, locate and correct the cause of the trouble before replacing the fuse.

1.

The line voltage selection switch (S3) is located next to the power supply transformer on the right rear of the bottom circuit board. Insert the tip of a small screwdriver into the slot of the switch and slide the switch until it clicks into place with the desired line voltage visible.

2.

Install a power line fuse appropriate for the line voltage. The fuse is located next to the internal line voltage switch (S3). Pull upward on the plastic fuse housing to remove the entire housing with the fuse inside. Select a fuse with the proper rating (see following table).

Line Voltage	Fuse Type
90-125V	½ A 250V, Slo Blo, 3AG
210-250V	¼ A 250V, Slo Blo, 3AG

3. Open the fuse housing by pushing up on the tab on the bottom of the housing.

4. Replace the fuse and close the housing. Replace the fuse housing into the fuse holder. Make sure the fuse snaps into place.

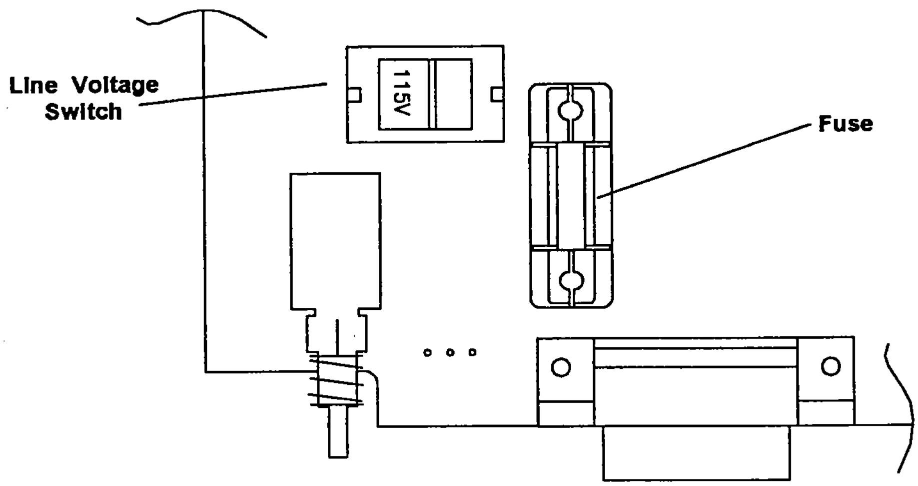


Figure 2.1: Line Voltage Switch and Fuse Location

5. Make note of the new voltage setting for later reference and carefully re-assemble the unit.

2.3 External Switch Settings

The Filter488 system configuration is set by switches accessible from its rear panel. Filter488 has two eight position switches that determine its configuration. The switches select which command set is to be used, RS-232C or IEEE 488, and the operation parameters for both. The switch labeled X is not used and should be set down (0) for future compatibility.

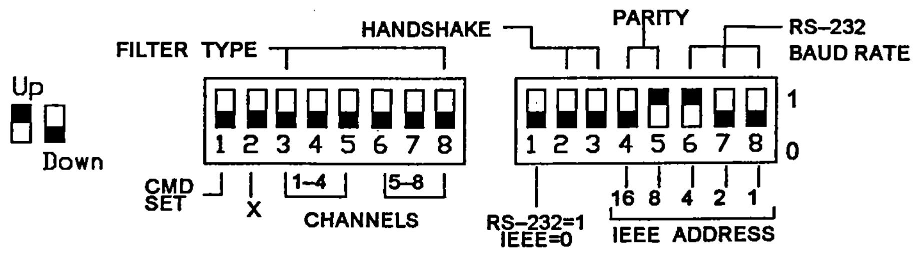


Figure 2.2: Filter488 Default Switch Settings

The rear panel switches are read only during power on or reset and should be set before applying power. Figure 2.2 shows the factory default settings for the rear panel switches. The tables following the diagrams show the options for the switches. Additional switch settings are shown in later sections describing IEEE 488 bus or RS-232C configuration.

Left Filter488 Rear Panel Switches		
Switch #	**Label**	**Settings**
1	CMD SET	1 (up): Reserved for future enhancements 0 (down): Register-based Command Set (default)
2	X	Not Used; Set to 0 (default).
3, 4, 5, 6, 7, 8	FILTER TYPE CHANNELS 1-4, 5-8	(5-8 ignored on Filter488/4) 000000: IOtech standard Filter488 module types (FL1, FL2, FL3 or FL4 (default)).

Right Filter488 Rear Panel Switches		
Switch #	Label	Settings
1	RS-232=1 IEEE=0	0: IEEE 488 Operation (default) 1: RS-232C Operation
4, 5, 6, 7, 8 IEEE 488 Operation	IEEE ADDRESS	IEEE 488 bus address. Default is 12.
2, 3 RS-232C Operation	HANDSHAKE	00: No handshaking 01: XON/XOFF handshaking 10: DTR/CTS handshaking 11: XON/XOFF and DTR/CTS handshaking
4, 5 RS-232C Operation	PARITY	00: No parity 01: Odd parity 10: Even parity
6, 7, 8 RS-232C Operation	BAUD RATE	000: 300 baud 100: 4800 baud 001: 600 baud 101: 9600 baud 010: 1200 baud 110: 19200 baud 011: 2400 baud 111: 38400 baud

2.3.1 IEEE 488 Configuration

This section describes the configuration for Filter488s that will be commanded through the IEEE 488 bus. When switch setup is completed, refer to page 2.11 for wiring instructions.

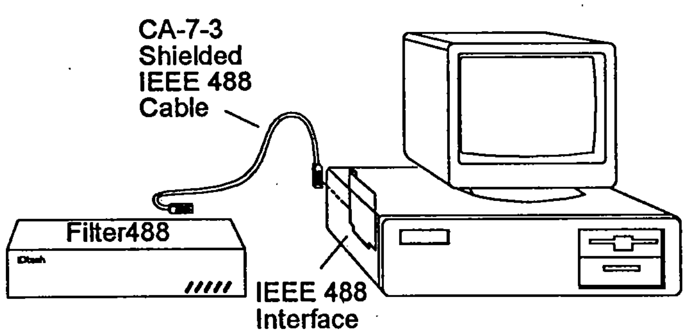


Figure 2.3: IEEE 488 System Configuration

The switch settings required for IEEE 488 operation are as follows.

The switch labeled **CMD SET** must be down (0). The up (1) position is reserved for future enhancements. This sets the unit for the register-based command set.

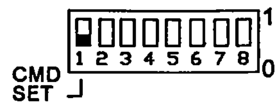


Figure 2.4: Set for Register-Based Command Set

The **FILTER TYPE** switches select the type of personality module that is used. For standard IOtech modules FL1, FL2, FL3 and FL4, set these switches down, as shown in Figure 2.5.

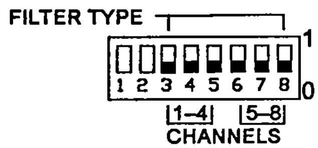


Figure 2.5: Set for Standard IOtech Filter Modules

To operate the Filter488 through its IEEE 488 interface, the switch labeled **RS-232=1/IEEE=0** must be set for the IEEE 488 bus, as shown in Figure 2.6.

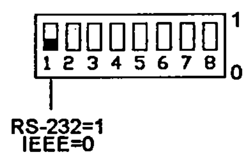


Figure 2.6: Filter Set for IEEE 488 Operation

All IEEE 488 bus devices, including the Filter488, must have an IEEE 488 bus address. The switches labeled **IEEE ADDRESS** are used for this purpose. The bus address can be set from 0 through 30 and is read only at power on or reset. The address is selected by simple binary weighting. The switch labeled **1** is the least significant bit; **16** is the most significant bit. The factory default is address 12, as shown in Figure 2.7. If the Filter488's bus address switches are set to 31, a bus address of 30 is used.

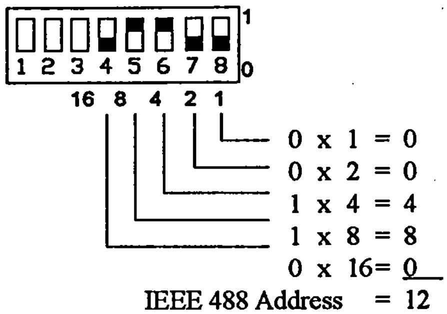


Figure 2.7: IEEE 488 Bus Address Setting (default)

2.3.2 RS-232C Configuration

This section describes the configuration for Filter488s that will be commanded through the RS-232C port. When setup is completed, refer to page 2.11 for wiring instructions.

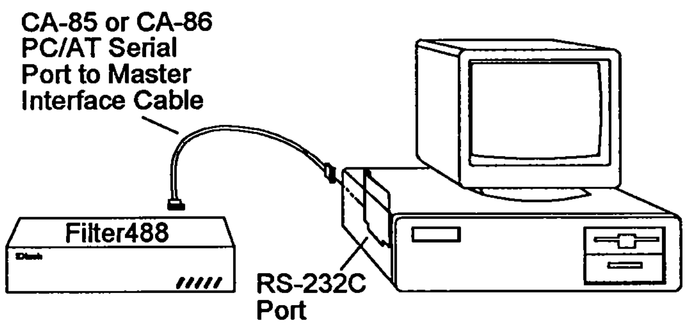


Figure 2.8: RS-232C System Configuration

The switch settings required for RS-232C operation are as follows:

The switch labeled CMD SET must be down (0). The up (1) position is reserved for future enhancements. This sets the unit for the register-based command set.

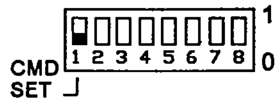


Figure 2.9: Set for Register-Based Command Set

The FILTER TYPE switches select the type of personality module that is used. For standard IOtech modules FL1, FL2, FL3 and FL4, set these switches down, as shown in Figure 2.10.

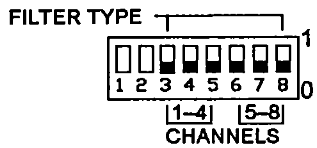


Figure 2.10: Set for Standard IOtech Filter Modules

To operate the Filter488 through its RS-232C interface, the switch labeled RS-232=1/IEEE=0 must be set for RS-232, as shown in Figure 2.11.

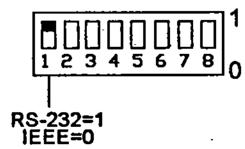


Figure 2.11: Set for RS-232C Operation

When the RS-232C port is used, the type of handshaking must be selected by the switches labeled HANDSHAKE. Filter488 offers the options of no handshaking, XON/XOFF, DTR/CTS or both XON/XOFF and DTR/CTS handshaking. Figure 2.12 shows the switch settings for each option.

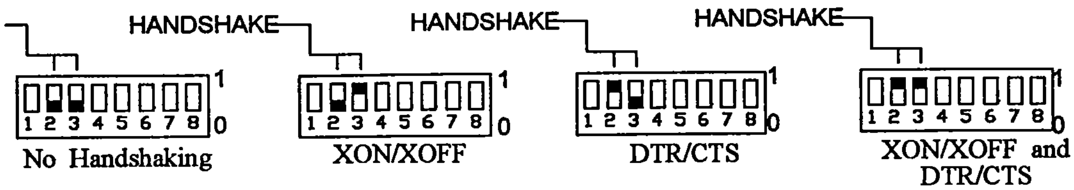


Figure 2.12: RS-232C Handshake Settings

The parity must be selected using the switches labeled PARITY. Filter488 provides for odd, even or no parity. Figure 2.13 shows the switch settings for each option.

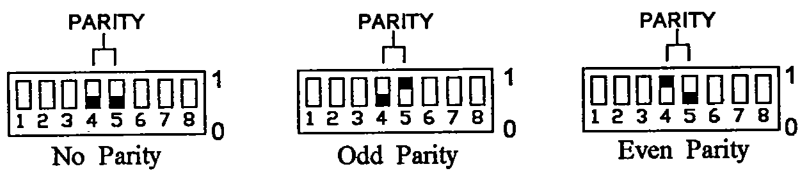


Figure 2.13: RS-232C Parity Settings

The baud rate is selected using the switches labeled BAUD RATE. The available baud rates are 300, 600, 1200, 2400, 4800, 9600, 19200 and 38400. The settings for each are shown in Figure 2.14. Baud rates of 19200 and 38400 are not supported by Filter488.

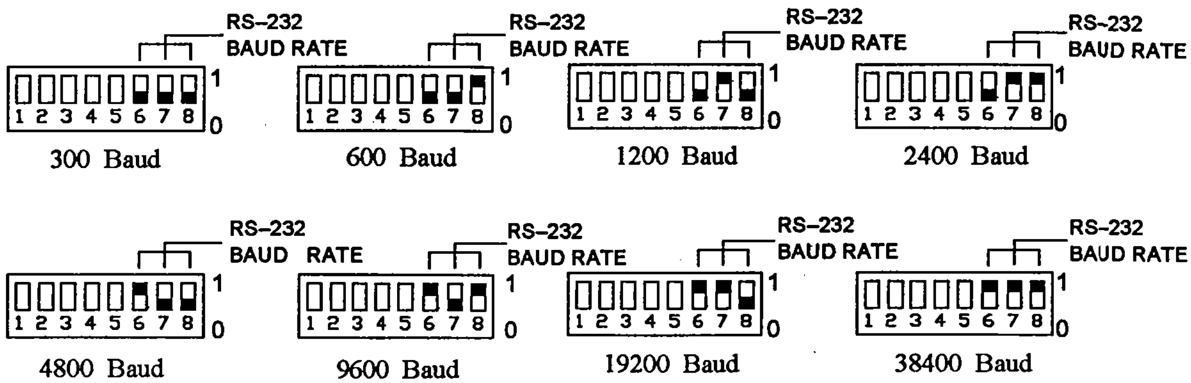


Figure 2.14: RS-232C Baud Rate Settings

2.4 Mounting

The Filter488 includes accessories for rack or bench use.

2.4.1 Rack Mount

If rack mount installation is required, remove the two plastic screws from the predrilled holes on each side of the unit. Only remove the screws from the set of holes that will be toward the front of the rack (the unit can be mounted with the front or rear panel facing the front of the rack fixture).

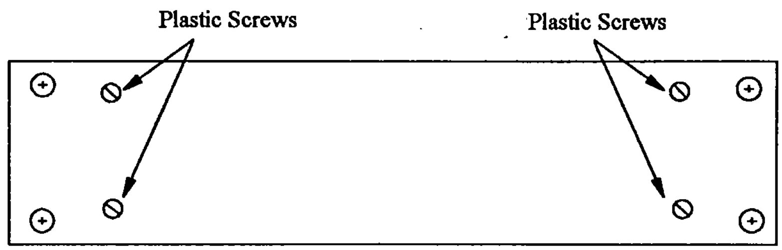


Figure 2.15: Rack Installation (Side View)

Install the two rack ears using the enclosed screws as shown in Figure 2.16.

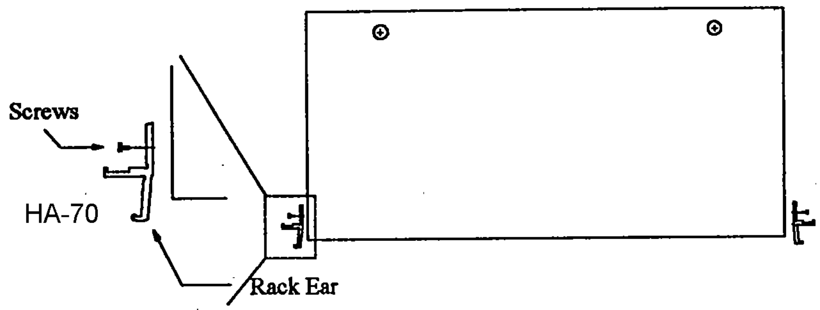


Figure 2.16: Installing Rack Ears (Top View)

2.4.2 Bench Top

For bench top use, place the self-adhesive rubber feet on the bottom of the unit approximately one inch from each corner.

2.5 Wiring

Wiring of the Filter488 for input consists of connecting the BNC connectors to the BNC terminals, as shown in Figure 2.17.

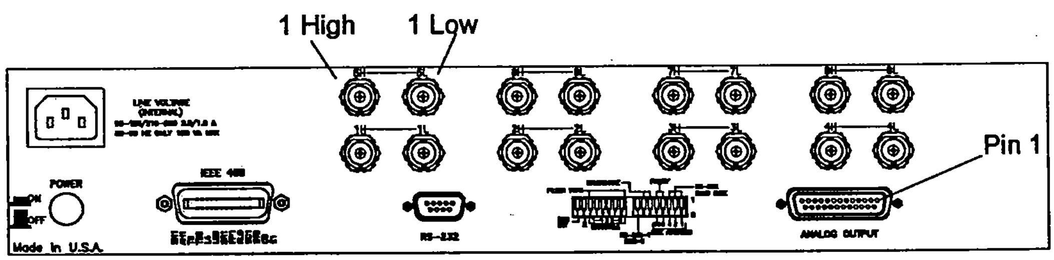


Figure 2.17: Filter488 Rear Panel BNC Connectors

2.5.1 Outputs

The location of pin one of the output connector is shown in Figure 2.17. The pinout of DB-25 is given in the following table.

Pin	Filter488/8 Connection	Filter488/4 Connection
1	Channel 1	Channel 1
2	Signal Ground	Signal Ground
3	Signal Ground	Signal Ground
4	Channel 3	Channel 3
5	Signal Ground	Signal Ground
6	Signal Ground	Signal Ground
7	Channel 5	No Connection
8	Signal Ground	Signal Ground
9	Signal Ground	Signal Ground
10	Channel 7	No Connection
11	Signal Ground	Signal Ground
12	Signal Ground	Signal Ground
13	No Connection	No Connection
14	Signal Ground	Signal Ground
15	Channel 2	Channel 2
16	Signal Ground	Signal Ground
17	Signal Ground	Signal Ground
18	Channel 4	Channel 4
19	Signal Ground	Signal Ground
20	Signal Ground	Signal Ground
21	Channel 6	No Connection
22	Signal Ground	Signal Ground
23	Signal Ground	Signal Ground
24	Channel 8	No Connection
25	Signal Ground	Signal Ground

2.6 Front Panel Indicators

Indicator lights on the Filter488 front panel display the status of the interface. The function of each indicator is described below.

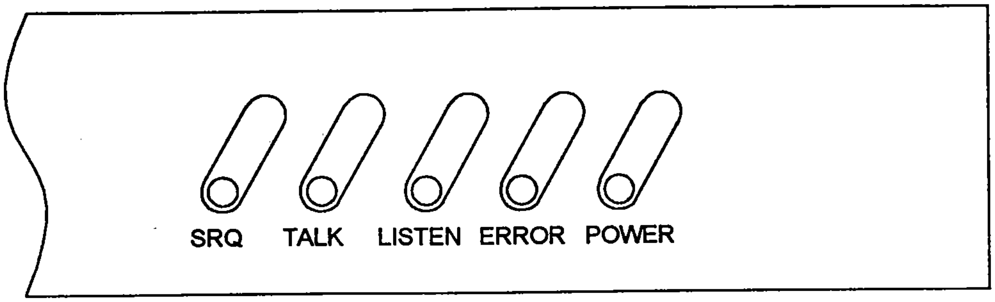


Figure 2.18: Filter488 Front Panel Indicator Lights

SRQ — SRQ (Service Request) is lit when the Filter488 requires the attention of the RS-232C or IEEE 488 controller. The SRQ light follows the internal Master Summary Status (MSS), which is maintained by the status reporting functions of the interface firmware (see Section 4.6.4 for more discussion).

TALK — If the unit is set for IEEE 488 communication, TALK is lit when Filter488 has been addressed to talk. It does not reflect actual data transmission. If the unit is set for RS-232C communication, TALK is lit while characters are being transmitted out the RS-232C port.

LISTEN — If Filter488 is set for IEEE 488 communication, LISTEN is lit when Filter488 has been addressed to listen. It does not reflect actual data transmission. If Filter488 is set for RS-232C communication, LISTEN is lit when characters are being received from the RS-232C port.

ERROR — On when an error has occurred, off when no error condition exists. ERROR is also used, in combination with other indicators, to display various self test results as described in Section 2.7.

POWER — On when power is applied to the Filter488 and the power switch on the rear panel is in the on position (depressed). Off if power is not present. POWER indicates that the 5 volt digital power supply is operating.

2.7 Power-Up

At initial power-up or on Reset (*R command), the Filter488 performs several automatic self-tests to ensure that it is fully functional. The indicator lights on the Filter488 front panel show any errors if they occur. Possible error conditions and their corresponding indicator light patterns are shown in the following table. Any pattern not shown is an internal error that is not field-serviceable – contact the factory. Check the error using the Error Query (E?) command (see Section 4.7 for more information on the E? command).

Failure	SRQ	Talk	Listen	Error	Power
General Hardware Failure	○	○	○	○	○
Communication or NVRAM Error	●	●	●	○	○
RAM Error	●	○	●	☼	○
ROM Checksum Error	●	●	○	☼	○

● Indicator light off. ○ Indicator light on. ☼ Indicator light flashing.

Filter488 Operation

3.1 Filter488 Commands

Operation of the Filter488 is accomplished using a set of character-based commands that configure the entire unit as well as each analog output port. The Filter488 commands are divided into two groups: Unit commands and Channel commands. Unit commands affect the operation of the entire unit and are not specific to a given channel. Channel commands only affect the operation of the selected channel.

3.1.1 The Register-Based Command Set

IOtech peripheral products for the IEEE 488 bus use a register-based command set. This model is based upon the concept of "registers." Just as most peripheral chips have one or more registers that may be set to different values to control their operation, each command letter in this command set corresponds to an internal register. In general, each of these registers holds a single numeric value that is maintained at the last value set.

Filter488 is controlled by modifying the contents of these registers. The relationship between the contents of the registers and the actions taken by the Filter488 are described in the following sections.

3.2 Analog Filter Channel Theory of Operation

Differential analog inputs are AC or DC coupled into a differential amplifier. This amplifier has a programmable gain of 1 or 10 as part of selecting the input range. The output feeds a programmable gain amplifier with a selectable gain of 1 or 10. The output of the programmable amplifier is then fed through a prefilter and further amplified by a range amplifier to give the desired span level for the filter. The switched capacitor filter (which can be bypassed under program control) filters the signal, after which it is post-filtered to remove clock feed-through noise. The switched capacitor filter cut-off frequency is set by either one of the two digitally synthesized clocks. The filtered signal is then amplified to restore its output range to ±1, 2, 5 or 10 V, and is either AC or DC coupled to the single-ended DB-25 output. The final gain stage includes a DC offset adjustment DAC to reduce the net DC offset error. The operation of Filter488 is illustrated by Figure 3.1.

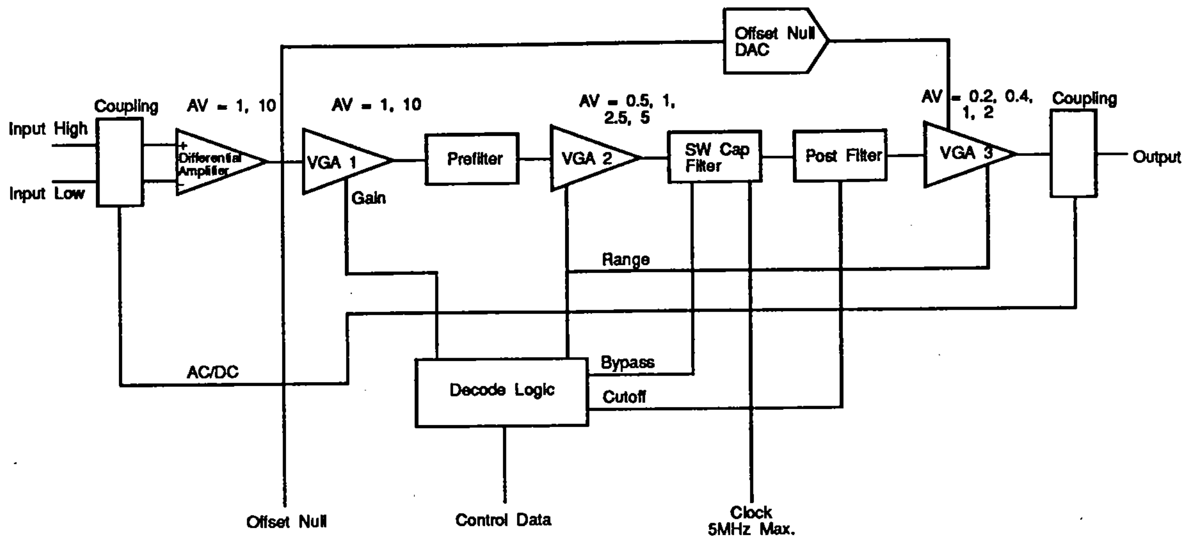


Figure 3.1 Single Channel Block Diagram

3.2.1 Coupling

The differential signal is AC or DC coupled into a differential amplifier. The input to the amplifier may be set to ground reference, without affecting the input signal, to remove the signal from the output. The output of this stage is single-ended and is AC or DC coupled. Programming the coupling affects both the input and output. This stage is set with the `An` command.

3.2.2 Input Range Setting and Prefilter

The next two amplifiers are set to the input range. The first amplifier puts the signal into a suitable voltage range to feed the prefilter. The output of the first amplifier feeds a prefilter to eliminate the possibility of aliasing the switched capacitor filter in the following stage. The prefilter has a roll-off of 6 db per octave.

The last preamplifier puts the signal into a suitable operating range for the switch cap filter. The input range is set by the `Ri,o` command.

3.2.3 Switched Capacitor Filter

The switched capacitor filter is a module that provides personality to each channel. The personality module may be ordered from the factory with Butterworth, Cauer (elliptic), Bessel or Chebyshev eighth order filter characteristics. Each channel may have a different module characteristic as described above. The module is frequency programmable to the desired cut-off using the Gn command. The filter design is based on Linear Technology's LTC1064 series switched capacitor filter. Filter specifications for Filter488 are located in the specifications section of this manual; Section 1.4. For more information on the LTC1064 parts contact Linear Technology.

3.2.4 Programmable Clock

There are two programmable clocks in Filter488. Each clock may be programmed for the desired cut-off frequency. The clocks use phase lock loops to control the frequency. This allows the clocks to be easily programmed over the usable range of Filter488. Because the phase lock loops are digital, they do not produce continuous frequencies over the entire frequency span. Section 1.4 charts the frequency steps for each range at which Filter488 will operate. The cut-off frequency is rounded off to the nearest programmed frequency, not to exceed the programmed frequency, that Filter488 can produce. Section 4.7 describes how to query Filter488 for the corrected cut-off frequency. The clock for the switched capacitor filter runs at 100 times the programmed cut-off frequency. Each clock may be programmed to any number and combination of channels.

3.2.5 Post Filter

The post filter is similar to the prefilter and is used for removing unwanted noise from the output of the switched capacitor filter.

3.2.6 Output Amplifier

The output amplifier is a single-ended programmable amplifier. DC offset nulling is achieved by a programmable 12 bit DAC. This allows the user to adjust for offset of the output.

3.2.7 Microprocessor Interface

Filter488 uses a 64180 microprocessor. The information from the IEEE 488 bus or RS-232 port is interpreted and sent to decode logic via opto-isolators. This provides good isolation between the controlling signals and the desired filtered signals.

3.3 Sample Programs

The following BASIC program uses a personal computer functioning as an IEEE 488 bus controller, using the IOtech Personal488 PC/IEEE 488 board and associated software. The Filter488 bus address is set to 12. Although we are using the IOtech IEEE interface in this example, the commands sent to the Filter488 apply to it's serial port as well.

As with all programs that use IOtech's Driver488/DRV, a short preamble is required to open the driver. The following program assumes that the IEEE 488 driver has been opened.

To make certain that the filter is in its original power up state, we first send it the `'*R'` commands. During the reset process, the Filter is unable to communicate on the IEEE 488 bus, so a line is placed in our program to wait 5 seconds.

```
PRINT #1, "output 12;*Rx"
SLEEP (5)
```

As a matter of good practice, an IEEE 488 Selected Device Clear can also be sent to the unit.

```
PRINT #1, "clear 12"
```

The Filter488 is capable of asserting the IEEE 488 interrupt line, SRQ, during its operation. To trap the SRQ, we will enable SRQ detecting in our IEEE 488 driver. SRQ trapping is a feature on the driver, not the Filter488. For more information on SRQ trapping, consult the Driver488 user manual. The following lines of Basic code prepare the driver to vector our program execution to a subroutine called 'srqhandler' when the driver detects an SRQ from the Filter488.

```
ON PEN GOSUB srqhandler
PEN ON
PRINT #1, "arm srq"
```

To allow the Filter488 to issue an SRQ for a specific internal event, the `M` command will be used. We will allow the Filter488 to issue the SRQ to the controller when an Event takes place.

```
PRINT #1, "output 12;M32x"
```

Although not necessary, we will next check for errors then query the Filter488 for is name and revision number. Sending a `E?` command will yield a 3 digit code from the Filter488 representing its error state. After receiving a `U9` command the Filter488 will respond with a string containing its name and revision number.

```
PRINT #1, "output 12;E?x"
PRINT #1, "enter 12"
INPUT #2, b$
PRINT #1, "output 12;U9x"
PRINT #1, "enter 12"
LINE INPUT #2, a$
```

```
CLS
PRINT "The error code of the "; a$; " is : "; b$
PRINT #1, "output 12;U2x"
PRINT #1, "enter 12"
LINE INPUT #2, c$
PRINT "The present settings of the Filter488 are :"; c$
```

The following lines of Basic code setup the cutoff frequency, the range, and the coupling for channel 1. Each line is commented to describe its function.

```
PRINT #1, "output 12;P1 x"          ' select channel 1
PRINT #1, "output 12;A0 x           ' DC coupling
PRINT #1, "output 12;F1, 100 x"     ' set cutoff #1 to frequency 100
PRINT #1, "output 12;G1 x"          ' set channel 1 to use cutoff #1
PRINT #1, "output 12;H50 x"         ' set offset constant to 50
PRINT #1, "output 12;R0,0 x"        ' input and output ranges set to 10V
```

Next channel two will be configured.

```
PRINT #1, "output 12;P2 x"          ' select channel 2
PRINT #1, "output 12;A1 x           ' AC coupling
PRINT #1,"output 12;F2,1000 x"      ' set cutoff #2 to frequency 1000
PRINT #1, "output12;G2 x"           ' set channel 2 to use cutoff #2
PRINT #1, "output 12;H150 x"        ' set offset constant to 150
PRINT #1, "output 12;R1,1x"         ' input and output ranges set to 5V
```

The following END statement marks the end of the main program. The SRQ handler used to service the Filter488 after its SRQ comes after the end of the main program.

```
END
```

Our SRQ handler is very simple. The Basic program is automatically vectored to this handler when Driver488 detects the SRQ from the Filter488. The lines of Basic code below Serial Poll the Filter488, which also serves to clear the SRQ, then prints the string "error occurred...".

```
srqhandler:
CLS
PRINT #1, "spoll 12"
INPUT #2, sp
PRINT "error occurred..."
RETURN
```

This second example shows Filter488 used with a Hewlett Packard HP8903 Audio Analyzer.

```
DECLARE SUB GetFilterString ()
DECLARE SUB ReadHPData (address!)
FilterAdr = 12                          'address of Filter488
HP8903Adr = 28                          'address of HP8903 Audio Analyzer
Channel = 1                             'select filter channel

OPEN "\DEV\IEEEOUT" FOR OUTPUT AS #1         'open driver
IOCTL #1, "BREAK"
PRINT #1, "RESET"
OPEN "\DEV\IEEEIN" FOR INPUT AS #2
PRINT #1, "FILL ERROR"

'reset filter defaults
PRINT #1, "OUTPUT"; FilterAdr; ";S2X"
'set filter for 10V input, 10 V output, AC coupling, clock1, 2000 Hz
PRINT #1, "OUTPUT"; FilterAdr; ";P"; Channel; "X"
PRINT #1, "OUTPUT"; FilterAdr; ";A1R0,0X"<R>PRINT #1, "OUTPUT";
FilterAdr; ";F1,2000X"
'turn off HP8903 filters
PRINT #1, "output"; HP8903Adr; ";H0L0"
'set HP8903 to read AC level
PRINT #1, "output"; HP8903Adr; ";M1"
'set HP8903 source for 100 Hz, 6V rms
PRINT #1, "output"; HP8903Adr; ";AP6.0VLFR100HZ"

'get Filter488 status information with "U" commands
PRINT #1, "OUTPUT"; FilterAdr; ";U9X"
PRINT #1, "enter"; FilterAdr
PRINT "Product Code:   ";
CALL GetFilterString
PRINT
PRINT #1, "OUTPUT"; FilterAdr; ";U3X"
PRINT #1, "enter"; FilterAdr
PRINT "Channel settings: ";
CALL GetFilterString
PRINT
PRINT #1, "OUTPUT"; FilterAdr; ";U2X"
```

```
PRINT #1, "enter"; FilterAdr
PRINT "System settings: ";
CALL GetFilterString
PRINT
PRINT #1, "OUTPUT"; FilterAdr; ";U1X"
PRINT #1, "enter"; FilterAdr
PRINT "Status Byte Reg: ";
CALL GetFilterString
PRINT
PRINT #1, "OUTPUT"; FilterAdr; ";U0X"
PRINT #1, "enter"; FilterAdr
PRINT "Event Status Reg: ";
CALL GetFilterString
PRINT

'perform magnitude response test
PRINT " Channel"; Channel; "Magnitude Response Test"
PRINT " Freq   Mag   dB"
fc = 2000
'set filter for 10V input, 10 V output, AC coupling, clock1, fc Hz
PRINT #1, "OUTPUT"; FilterAdr; ";P"; Channel; "X"
PRINT #1, "OUTPUT"; FilterAdr; ";A1G1R0,0X"
PRINT #1, "OUTPUT"; FilterAdr; ";F1,"; fc; "X"
'set HP8903 to read AC level
PRINT #1, "output"; HP8903Adr; ";M1"
'set HP8903 for 100 Hz, 6V
PRINT #1, "output"; HP8903Adr; ";AP6.0VLFR100HZ"
f3db1 = 0
FOR f = (fc  300) TO (fc + 300) STEP 50
   'set input frequency
   PRINT #1, "output"; HP8903Adr; ";FR"; f; "HZ"
   'read output level
   CALL ReadHPData (HP8903Adr)
   Mag = HPDataIn
   dbmag = (20 / 2.3026) * LOG (Mag / 6!)
   PRINT USING "#####"; f;
   PRINT USING "#####.##"; Mag; dbmag
   IF (f3db1 = 0) AND (dbmag  3!) THEN f3db1 = f
```

```
NEXT f
LOCATE 10, 30: PRINT " 3 dB frequency ="; f3db1; "Hz "

'manipulate output offset
PRINT " Channel"; Channel; "DAC Offset Test ": COLOR 7, 0: PRINT
'set HP8903 for 0V, DC measurement
PRINT #1, "output"; HP8903Adr; ";AP0.0VLS1"
'set filter for DC coupling, 10V scale, bypass mode
PRINT #1, "OUTPUT"; FilterAdr; ";A0P"; Channel; "G0R0,0X"
'set maximum negative offset
PRINT #1, "OUTPUT"; FilterAdr; ";P"; Channel; "H0X"
CALL ReadHPData (HP8903Adr)
bypoffset0 = HPDataIn
PRINT "Negative Bypass Offset: ";
PRINT USING "###.###"; bypoffset0; : PRINT "  ";
'set mid range offset
PRINT #1, "OUTPUT"; FilterAdr; ";P"; Channel; "H128X"
CALL ReadHPData (HP8903Adr)
bypoffset128 = HPDataIn
PRINT "Mid range  Bypass Offset: ";
PRINT USING "###.###"; bypoffset128; : PRINT "  ";
'set maximum offset
PRINT #1, "OUTPUT"; FilterAdr; ";P"; Channel; "H255X"
CALL ReadHPData (HP8903Adr)
bypoffset255 = HPDataIn
PRINT "Positive Bypass Offset: ";
PRINT USING "###.###"; bypoffset255; : PRINT "  ";

'manipulate filter gain ranges
'set HP8903 for 100 Hz, 1 volt, AC measurement
PRINT #1, "OUTPUT"; HP8903Adr; ";AP1.0VLFR100HZM1"
'set filter for AC coupling, 10/10 scale, filter
PRINT #1, "OUTPUT"; FilterAdr; ";A1P"; Channel; "G1R0,0X"
CALL ReadHPData (HP8903Adr)
vftr00 = HPDataIn
PRINT "10/10 gain ="; : PRINT USING "#.###"; vftr00;
'set filter for AC coupling, 10/5 scale, filter
PRINT #1, "OUTPUT"; FilterAdr; ";A1P"; Channel; "G1R0,1X"
```

```
CALL ReadHPData (HP8903Adr)
vftr01 = HPDataIn
PRINT "10/5 gain ="; : PRINT USING "#.###"; vftr01;
'set filter for AC coupling, 5/1 scale, filter mode
PRINT #1, "OUTPUT"; FilterAdr; ";A1P"; Channel; "G1R1,3X"
CALL ReadHPData (HP8903Adr)
vftr13 = HPDataIn
PRINT "5/1  gain ="; : PRINT USING "#.###"; vftr13;
'set filter for AC coupling, 1/1 scale, filter
PRINT #1, "OUTPUT"; FilterAdr; ";A1P"; Channel; "G1R3,3X"
CALL ReadHPData (HP8903Adr)
vftr33 = HPDataIn
PRINT "1/1  gain ="; : PRINT USING "#.###"; vftr33;

CLOSE
PRINT "Filter488 test terminated."
STOP

SUB GetFilterString
   WHILE X$ CHR$ (10)
         X$ = INPUT$ (1, 2)
         IF X$ CHR$ (10) THEN PRINT X$;
   WEND
   COLOR 7, 0
   X$ = INPUT$ (2, 2)
   PRINT
END SUB

SUB ReadHPData (address)
   SHARED HPDataIn
   'read output level
   PRINT #1, "enter"; address
   INPUT #2, HPDataIn
 END SUB
```

Notes:

Command Descriptions

4.1 Overview

The Filter488 is controlled by modifying the contents of its internal registers through commands. The relationship between the contents of the registers and the actions taken by Filter488 are described in the command descriptions that follow in this section.

There are two types of register-based commands. Unit commands affect the entire Filter488 unit. Channel commands affect only the channel specified by the Select Channel (Pn) command.

The Unit commands are:

Reset (*R)
Response Terminator (Dn)
Error Query (E?)
Frequency Cutoff (Fn,xxxxx)
SRQ Mask (Mn)
Event Mask (Nn)
Select Channel (Pn)
Save/Restore (Sn)
Status (Un)
Execute (X)

The channel is selected using the Pn command. The Channel commands are:

Analog Coupling (An)
System Clock Select (Gn)
Offset Calibration (Hn)
Range (Ri,o)

Most commands consist of one alphabetic character followed by one or more numbers. The alphabetic character is the command and the number(s) are the command parameters.

The examples in this section use a personal computer functioning as an IEEE 488 bus controller, using the IOtech Personal488 PC/IEEE 488 board and associated driver software. All examples are given using BASIC. The Filter488 bus address is set to 12 for all examples.

In order to establish communication with Driver488, IOtech's IEEE 488 interface driver, from BASIC, the following sequence must be used:

```
OPEN "\DEV\IEEEOUT" FOR OUTPUT AS #1
IOCTL#1,"BREAK"
PRINT#1,"RESET"
OPEN "\DEV\IEEEIN" FOR INPUT AS #2
PRINT#1,"TERM IN LF EOI"
```

All of the command examples assume the driver has been properly opened and reset by the above sequence.

4.2 Terminators

Responses from a Filter488's IEEE 488 port are terminated by default with a line-feed with EOI (End Or Identify) asserted. Responses from a Filter488's RS-232C port are, by default, terminated with a carriage-return followed by a line-feed. The Response Terminator (`Dn`) command is used to change the terminator(s). Commands to a Filter488's IEEE 488 port must be terminated with the `X` command. Filter488 generally treats control characters, such as carriage-return and line-feed, as white space and ignores them unless they are within a command or command option.

4.3 Command Interpretation

As commands are received by Filter488, they are interpreted in the order in which they are received. Some commands are immediate, which means they immediately take effect. Other commands are deferred, and have no effect on device operation until the execute command (`X`) is interpreted.

An example of an immediate command is Select Channel (`Pn`), which immediately chooses which channel is being programmed or addressed. The immediate commands are `Dn`, `E?`, `Pn`, and all queries including `Un`. Immediate commands must be followed by an `X` command to terminate the command string for correct operation. For example: `P1 X`.

An example of a deferred command is Range (`Ri,o`), which sets the input and output range for a channel when `X` is interpreted. As deferred commands are interpreted, their desired effects are recorded in internal temporary registers. As additional deferred commands are interpreted, their effects are added to these registers, possibly overwriting earlier commands' effects. Finally, when `X` is interpreted, the temporary registers are examined according to their sequence in the command line. If two deferred commands are sent that affect the same function, the earlier command is overridden.

If an error is detected during command processing, commands are ignored up through and including the next execute (`X`) command. Thus any immediate commands after the error, as well as all deferred commands, are ignored until the `X` is received.

Deferred commands help reduce the effects of errors. The primary advantage of deferred commands is that they are executed as a group – either all or none. If any errors occur, deferred commands have no effect and the device is left in a consistent state instead of a partially modified, inconsistent state.

The deferred commands are `An`, `Fn,xxxxx`, `Gn`, `Hn`, `Mn`, `Rn,m`, `Sn` and `*R`.

4.4 Syntax Rules

Most commands are identified by a single letter (A through Z) or an asterisk (*) followed by an single letter.

4.4.1 Case Sensitivity

Commands can be entered in upper or lower case. For example, the command `A1 X` acts the same as `a1 x`.

4.4.2 Spaces

White space, which consists of all ASCII values of 32 and below and includes the space, tab, new-line (line-feed) and carriage-return characters, is generally allowed anywhere between commands and command arguments. White space is not allowed in the middle of command options (for instance, `1  2  3` is not the same as `123`).

4.4.3 Multiple parameters

If more than one parameter is used for a command, they must be separated by a comma and can be separated by white space.

Examples:

```
PRINT#1,"OUTPUT 12;R1, 2"
```

is a command with two parameters: 1 and 2.

4.4.4 Command Strings

Commands can be sent individually or in a string with other commands.
For example, these three commands:

```
PRINT#1,"OUTPUT 12;P1X"
PRINT#1,"OUTPUT 12;R1,2 X"
PRINT#1,"OUTPUT 12;A1 X"
```

have the same effect as the single command:

```
PRINT#1,"OUTPUT 12;P1 R1,2 X A1 X"
```

4.4.5 Query Option

Most commands have a corresponding query command formed by appending a question mark (?) to the command letter. Query commands respond with the present configuration or mode of a previously executed command. When appropriate, the response from a query command is in the form of a command string that, if it were executed, would put the unit into the configuration it was in when the query was executed. For example, the response to a G? X is a command of the form Gnnnnn, where nnnnn is the present frequency setting. Query responses are always fixed-length strings in a pre-defined format.

Any number of query commands can be combined into one string to create a specialized status command that responds with only the information of interest for a given application. For example, P? G? X responds with the current channel number followed immediately by its current setting, such as P3G012345. No spaces separate the responses from consecutive queries.

Query commands are immediate. Their responses are generated as soon as they are interpreted, before any other commands, including X. For example:

`PRINT#1,"OUTPUT 12;A0 X A? X"`	Response is A0.
`PRINT#1,"OUTPUT 12;A1 X A? X"`	Response is A1.
`PRINT#1,"OUTPUT 12;A2 A? X"`	Response is still A1, because the A? is an immediate command. The A2 is a deferred command that takes effect at the X.
`PRINT#1,"OUTPUT 12;A? X`	Response is now A2.

Even though query commands generate their response as soon as they are interpreted, they must still be followed by the Execute (X) command for proper termination.

4.5 Default Configuration

The factory default configuration, which is restored by the execution of the restore defaults (S2) command, is as follows:

Command	Effect
A0	DC coupling.
D4	LF with EOI
F1	10,000
F2	10,000
G0	Set the selected channel to filter bypass mode.
Hn	128
M0	Clear Service Request Enable Register.
N0	Clear Event Mask
R0,0	Input and output ranges are set for 10 V input and 10 V output.

4.6 Status Reporting

The Filter488 includes several registers whose bits indicate various status conditions within the unit. These registers include:

ESC	Error Source Register
ESR	Event Status Register
ESE	Event Status Enable Register
STB	Status Byte Register
SRE	Service Request Enable Register

and are shown in Figures 4.1A and 4.1B.

4.6.1 Error Source Register

The error source register indicates which errors, if any have occurred. The individual errors are described in the E? command.

When an error occurs, it sets the appropriate bit in the error source register. This in turn sets a bit in the event status register as shown in Figure 4.1B.

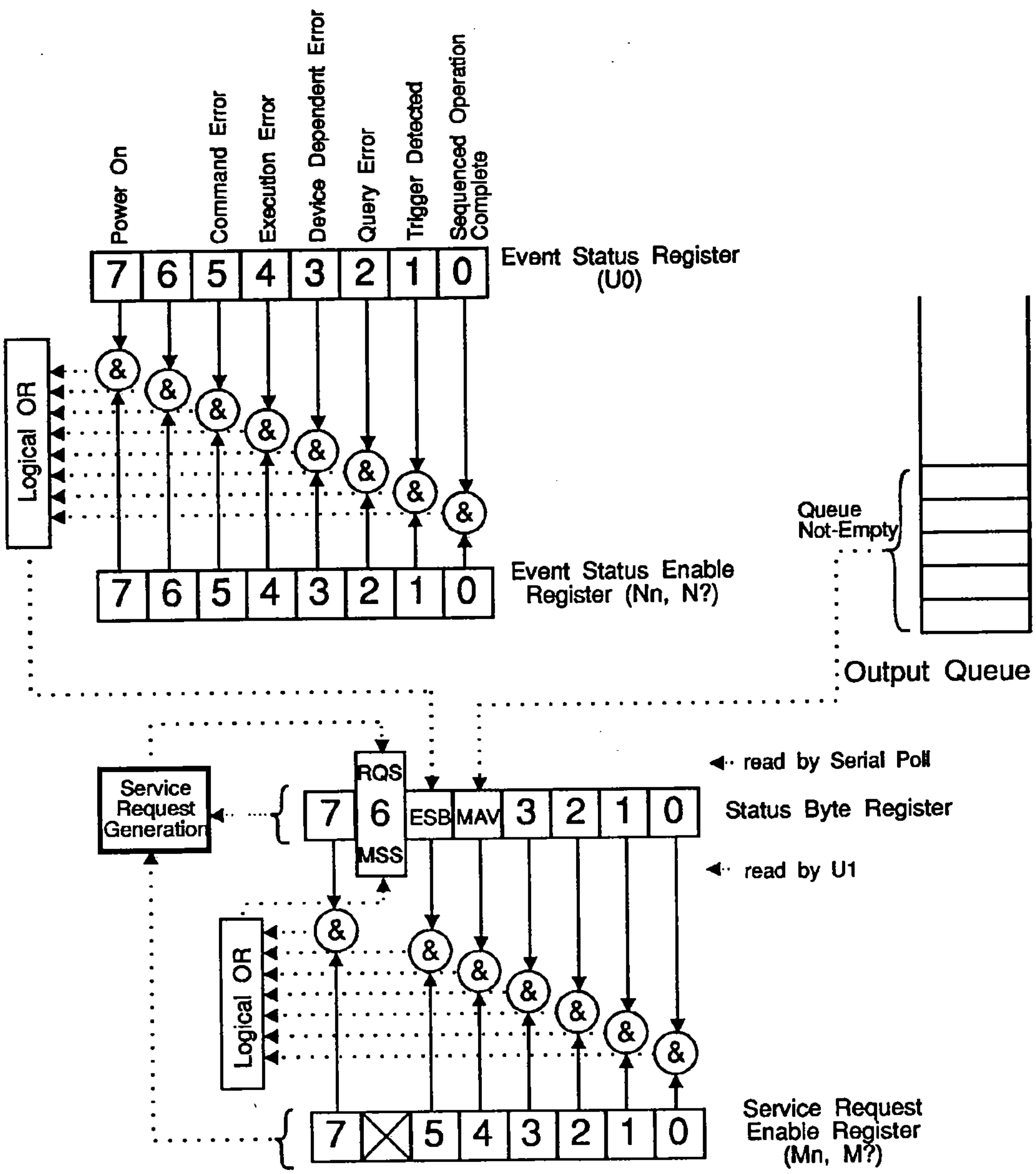


Figure 4.1A: Filter488 Status Reporting Registers

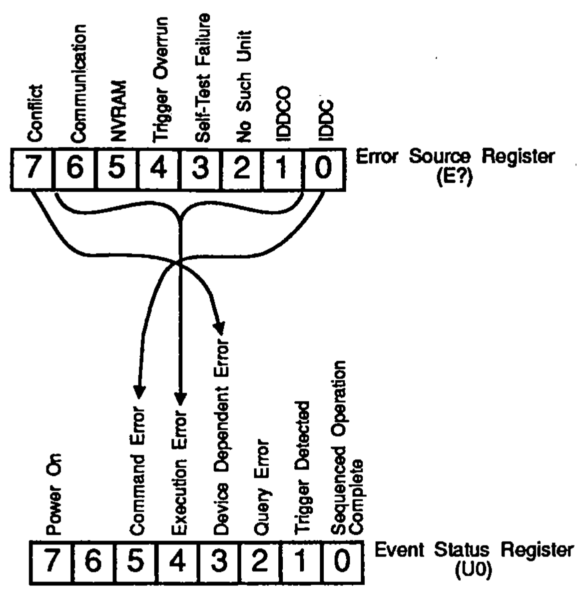


Figure 4.1B: Filter488 Error Source Registers

4.6.2 Event Status Register

The event status register indicates which events, if any, have occurred. Its bits, and the event that set them, are as follows:

`Bit 0`	Sequenced Operation Complete	Set when sequenced operation has been stopped for any reason, including completion of the sequence, execution of an appropriate command (such as `L0`), or an error such as a trigger overrun.
`Bit 1`	Trigger Detected	Set when the Filter488 has been configured to accept triggers and the specified trigger has occurred.
`Bit 2`	Query Error	Set when the controller has attempted to read from the Filter488 when no response is present or pending, or when a response has been lost because the controller has sent a new query before reading the response to a prior query.
`Bit 3`	Device Dependent Error	Set when a conflict error has occurred. A conflict error is generated when a command cannot execute correctly because it would interfere with other commands or settings.
`Bit 4`	Execution Error	Set when one of several errors have occurred during the execution of a command.

`Bit 5`	Command Error	Set when a command syntax error is detected.
`Bit 6`	Unused	Reserved for future expansion. Always 0.
`Bit 7`	Power On	Set on power-on or system reset (`*R`).

The event status register is read with the `U0` command, which clears the register after responding with its contents.

4.6.3 Event Status Enable Register

The event status enable register controls which events, if any, are to be reflected in the status byte register. As shown in Figure 4.1A, the bits of the event status register are logically ANDed with the corresponding bits of the event status enable register, and the resulting bits are logically ORed together to form the event status bit (ESB) in the status byte register. The event status enable register does not affect the event status register; it only affects the ESB bit of the status byte register. The event status enable register is set and interrogated with the `Nn` command.

4.6.4 Status Byte Register

The Filter488 status byte register has only three active bits:

`Bit 4`	Message Available (MAV)	Set when a response is waiting to be read by the controller. Unread responses are stored in an output queue which sets the message available bit when it is not empty.
`Bit 5`	Event Status	Set when the corresponding bits in both the event status register and the event status enable register are set.
`Bit 6`	Request For Service (RQS) Master Summary Status (MSS)	Request for service is returned with the status byte when it is read by the controller using a serial poll. If it is set, then it indicates that the Filter488 is requesting service and asserting the IEEE 488 bus Service Request (SRQ) signal. Master summary status is returned with the status byte when it is read with the Read Status Byte `U1` command. If it is set, then it indicates that corresponding bits of the status byte and service request enable registers are set. RQS is cleared when the Filter488 is serial polled.

All other bits of the status byte register are unused and are always 0.

4.6.5 Service Request Enable Register

The service request enable register controls which bits of the status byte register are to be reflected in the request for service and master summary status bits of the status byte register. As shown in Figure 4.1A, the bits of the status byte register are logically ANDed with the corresponding bits of the service request enable register, and the resulting bits are logically ORed together to form the master summary event status bit (MSS) in the status byte register and to control the request for service (RQS) bit in that register. The service request enable register does not affect the status byte register; it only affects the MSS and RQS bits of the status byte register. The event status enable register is set and interrogated with the Mn command.

4.7 Command Set

The Command Set for Filter488 is described on the following pages.

Reset *R

Type: Unit Command, Deferred

`*R` Restores the Filter488 to its initial power-up state.

This command has the same effect as removing and re-applying power. Filter488 is returned to its power up default settings as described in Section 4.5.

The reset process takes several seconds, during which the Filter488 is unable to receive or process commands. No commands should be addressed to the unit for at least five seconds after issuing the reset command.

The IEEE 488 bus Device Clear commands (DCL and SDC) do not have the same effect as `*R`. They do not perform a reset. They do clear any pending commands and responses and prepare the Filter488 to receive new commands.

Example:

`PRINT#1,"OUTPUT 12;*RX"`	Reset unit.
`SLEEP(5)`	Wait for reset to finish.

Analog Coupling An

Type: Channel Command, Deferred

`A0`	DC coupling.
`A1`	AC coupling.
`A2`	Ground inputs; DC output coupling.
`A?`	Response is present coupling setting for channel selected.

Analog Coupling sets the coupling for the Filter488 channel selected by the `Pn` command.

Examples:

`PRINT#1,"OUTPUT 12;P1 X"`	Specify channel 1.
`PRINT#1,"OUTPUT 12;A1 X"`	Set channel 1 to AC coupling.
`PRINT#1,"OUTPUT 12;P1 A2 P2 A1 X"`	Set channel 1 to ground inputs with DC output coupling; set channel 2 to AC coupling.

Response Terminator Dn

Type: Unit Command, Immediate

Dn	Set output terminator used by RS-232C and IEEE 488.
D0	LF
D1	CR
D2	CR-LF (RS-232C default)
D3	LF-CR
D4	IEEE 488: LF with EOI; RS-232C: LF (IEEE 488 default)
D5	IEEE 488: CR with EOI; RS-232C: CR
D6	IEEE 488: CR-LF with EOI; RS-232C: CR-LF
D7	IEEE 488: LF-CR with EOI; RS-232C: LF-CR
D?	Response is Dn, where n is the present termination setting, a one digit number from 0 to 7.

The Response Terminator command sets the character(s) that will be appended to the end of responses from the Filter488. The power-on default terminator is LF with EOI (D4) for the IEEE 488 interface and CR-LF (D2) for the RS-232C interface. The choice of the correct response terminator can make responses easier to receive and interpret. For example, if CR-LF with EOI was preferred for an IEEE 488 system, the following command could be used:

`PRINT#1,"OUTPUT 12;D6 X"` Set response terminator to CR-LF with EOI.

Error Query E?

Type: Unit Command, Immediate

`E?`	Reports and clears the current contents of the error source register. After execution of the Error Query command, Filter488 responds with one of the following error codes:
`E000`	No error has occurred.
`E001`	Invalid device dependent command (IDDC). Due to a command syntax error.
`E002`	Invalid device dependent command option (IDDCO). A command parameter was out of range or missing.
`E004`	Command conflict.
`E008`	Self-test failure. Reserved for non-critical internal errors.
`E032`	Non-volatile RAM failure. Access to the non-volatile RAM was impossible, or the data retrieved were corrupt.
`Ennn`	If two or more errors occurred, nnn is the sum of the corresponding error codes.

The Error Query command responds with the contents of the error source register and then clears that register. When an error occurs, the appropriate bits are set in the error source and standard event registers, and possibly in the status byte register. The ERROR indicator light on the front panel of the Filter488 illuminates.

The error query clears the error source register, the corresponding bits in the standard event register, and possibly bits in the status byte register. Clearing the error source register allows the ERROR indicator light to turn off.

Error query responds with Ennn, where nnn is a three-digit decimal number equal to the sum of the error codes.

Example:

`PRINT#1,"OUTPUT 12;E?X"`	Query error source.
`PRINT#1,"ENTER 12"`	Retrieve response.
`INPUT#2,E`	
`PRINT E`	Displays 000 (no errors).
`PRINT#1,"OUTPUT12;P9 X"`	Try to select an invalid channel.
`PRINT#1,"OUTPUT 12;E?X"`	Query error source.
`PRINT#1,"ENTER 12"`	Retrieve response.
`INPUT#2,E`	
`PRINT E`	Displays 002 (IDDCO).

If more than one type of error has occurred, the response is the sum of the corresponding error codes.

Frequency Cutoff Fn,xxxxx

Type: Unit Command, Deferred

`Fn,xxxxx.x`
: Set the cut-off for the selected clock. Where n is system clock 1 or 2 and frequency range, `xxxxx.x` is 1-50,000.0 Hz

`F?`
: Responds with both actual cutoff frequencies.

The Frequency Cutoff command sets the clocks used by each channel to set cut-off frequency. There are two independent clocks. Each clock may be programmed to result in the desired cut-off frequency between 1 to 50000.0 Hz. The frequency is entered as the desired cut-off frequency and is then rounded off to the nearest obtainable frequency of the clock generators. Leading zeros are not required and the decimal point can be used or omitted. F 1, 100 is the same as F 1, 100.0. The actual cut-off frequency may be read back with the F?. Each channel is programmed to use either clock by the use of the Gn command.

Example:

`PRINT#1, "OUTPUT 12; F1,100 X"`	Set clock #1 for a cutoff of 100Hz.
`PRINT#1, OUTPUT 12; F2,200 X"`	Set clock #2 for a cutoff of 200Hz.
`PRINT#1, "OUTPUT 12;F? X"` `PRINT#1, "ENTER 12"`	Query the system clock settings.
`INPUT#2, FREQUENCY$`	Read the response from Filter488.
`PRINT FREQUENCY$`	Response is F00100.1F00200.1.

Note that a wait of about a half a second is needed whenever the sequence `F1, xxxxx F2, xxxxx x` is sent to the filter.

System Clock Select Gn

Type: Channel Command, Deferred

`G0`	Set the selected channel to filter bypass mode. In bypass mode the 8-pole switched capacitor filter is not used. The signal is routed through the gain stages selected by the `Ri, o` command, as well as the 2-pole low pass prefilter. The prefilter cutoff frequency is approximately two times the channel cutoff frequency selected by the `Fn, xxxxx` command.
`G1`	Sets system clock #1 to use as the cutoff frequency for selected channel.
`G2`	Sets system clock #2 to use as the cutoff frequency for selected channel.
`G?`	Responds with system clock # programmed for selected channel.

The Frequency Cutoff command (`Fn,xxxxx.x`) sets the clocks for the desired cutoff frequencies. The `Gn` command then selects which of the two system clocks to use for the selected channel. To program a channel for a desired cutoff frequency, the system clock must be set to the correct cutoff frequency and then the channel must be programmed to the correct system clock. The cutoff frequency for a selected channel may then be changed by reprogramming the system clock with the `Fn,xxxxx.x` command or by reprogramming the selected channel's system clock to the opposite system clock.

Example:

`PRINT#1, "OUTPUT 12; F1,100 X"`	Set clock #1 for a cutoff of 100Hz.
`PRINT#1, OUTPUT 12; F2,200 X"`	Set clock #2 for a cutoff of 200Hz.
`PRINT#1, "OUTPUT 12;P1 G1 X"`	Set channel #1 to system clock 1. Channel #1 now has a cutoff of 100Hz.
`PRINT#1,"OUTPUT 12; G2 X"`	The selected channel is still #1 and is now set to system clock #2. Channel #1 is now set for a cutoff of 200Hz.

Offset Constant Hn

Type: Channel Command, Deferred

Hn	Sets offset constant to n (0–255) for the selected channel (factory default is 128).
H?	Responds with offset constant for the selected channel.

Offset Constant sets the output offset of the channel selected by the Pn command. This command allows the user to null any DC offset in the output circuitry. The output offset correction circuit of Filter488 allows +/- 6.25 V correction of the total system offset. System offset may vary due to factors such as the offset of the input signal, the range-dependent output offset of the analog components, and the offset due to the individual switched-capacitor filter modules.

Zero correction is obtained with H128 (default) command; note that this is not the same as zero offset. The H129 command provides a positive correction of approximately 50 millivolts, while the H126 command provides a negative 100 mV correction.

Example:

`PRINT#1,"OUTPUT 12;P1 X"`	Select channel 1.
`PRINT#1,"OUTPUT 12;H255 X"`	Set offset constant to 255. Increases the DC output offset level.
`PRINT#1,"OUTPUT 12;H? X"`	Query Offset constant setting. Response is H255.

SRQ Mask Mn

Type: Unit Command, Deferred

`M0`	Clear Service Request Enable Register (default)
`M4`	SRQ on ready.
`M16`	SRQ on Message Available.
`M32`	SRQ on event.
`M?`	Response is `Mnnn` where `nnn` is the service request enable mask.

The SRQ Mask command sets or clears the service request enable register, which controls the generation of Master Summary Status, Request for Service, and the IEEE 488 bus SRQ signal.

The SRQ Mask command enables SRQs on one or more of the conditions listed above. Multiple SRQ Mask conditions can be enabled simultaneously by issuing them separately or by combining them in one command string. The programmed SRQ modes remain enabled until the `M0` (clear SRQ mask) command is sent, or the controller sends a Reset (`*R`) command. This command acts directly on the Service Request Enable Register (see Status Reporting, Section 4.6).

Example:

`PRINT#1,"OUTPUT 12;M16 M32 X"`	Enable SRQ on Message Available or Event Detected.
`PRINT#1,"OUTPUT 12;M48 X"`	Same effect as previous command.
`PRINT#1,"OUTPUT 12;M? X"`	Read present `Mn` setting. Response is `M048` (32+16).

Event Mask Nn

Type: Unit Command

`N000`	Clear event mask (default).
`N004`	Enable query error.
`N016`	Enable execution error.
`N032`	Enable command error.
`N128`	Power-on.
`N?`	Read standard event status enable register.

This command directly sets the Event Status Enable Register (ESE). ESE determines which conditions in the Event Status Register (ESR) are enabled to generate Event Status register Bit (ESB) in the Serial Poll Status Register. See the section on the Serial Poll Model (Section 4.6) for complete details.

Multiple ESR bits can be enabled simultaneously by issuing them separately or by combining them in one command string. The programmed event enables remain set until a Clear Event Mask (`N000`) command is sent or the controller sends a `*R` command.

A query error (`N004`) is set when an attempt is made to read data from the output queue when no data are present or data in the output queue were lost. Data may be lost when too many data are requested to be buffered in the queue (for example, issuing multiple commands to return data without ever reading them).

A device dependent error (`N008`) is set when a conflict in programmed parameters is detected. This is also referred to as a conflict error.

An execution error (`N016`) is set when a parameter exceeds valid limits for a particular command. This is also referred to as Invalid Device Dependent Command Option (IDDCO) error.

A command error (`N032`) is set when an illegal command is sent to the Filter488. This is also referred to as Invalid Device Dependent Command (IDDC) error.

A power on (`N128`) is set whenever Filter488 is first powered up.

Example:

`PRINT#1,"OUTPUT12;N0 X"`	Clear ESE.
`PRINT#1,"OUTPUT12;N? X"`	Read ESE back, computer screen shows `N000`.
`PRINT#1,"ENTER12"`	Read Filter488 for the query response.
`INPUT#2,A$`	Read the response from Driver488.
`PRINT A$`	Computer screen shows `N000`.
`PRINT#1,"OUTPUT12;N4 X N8 X"`	Set enable on query error, enable on Device Dependent (conflict) error.
`PRINT#1,"OUTPUT12;N? X"`	Query event mask setting.
`PRINT#1,"ENTER12"`	Read Filter488 for the query response.
`INPUT#2,A$`	Read the response from Driver488.
`PRINT A$`	Computer screen shows `N012`, the logical OR of these two conditions.

Select Channel Pn

Type: Unit Command, Immediate

`Pn` Selects the channel subsequent commands affect. n is 1-8 for the Filter488/8, 1-4 for the Filter488/4.

`P?` Responds with `Pn`, where n is the presently selected channel or 0 if all channels are selected.

Select Channel chooses the channel that subsequent commands affect.

Select Channel (`Pn`) immediately affects all subsequent commands, including subsequent deferred commands such as `An`. This allows several channels to be referred to in a single command string (as in `P1 A1 P2 A1 X`).

Example:

`PRINT#1,"OUTPUT 12;P1 X"`	Specify channel 1.
`PRINT#1,"OUTPUT 12;A1 X"`	Select AC coupling.
`PRINT#1,"OUTPUT 12;P4 A1 X"`	Select AC coupling on channel 4.
`PRINT#1,"OUTPUT 12;P1 A1 P2 A0 X"`	Select AC coupling on channel 1, DC coupling on channel 2.

Range Ri,o

Type: Channel Command, Deferred

`Ri,o` Sets input and output range for the channel specified by `Pn`.

`R?` Responds with the present input and output range settings for the channel specified by the `Pn` command.

INPUT RANGES	
Setting for i	**Voltage Value**
0	10
1	5
2	2
3	1
4	0.5
5	0.2
6	0.1
7	0.05
8	0.02
9	0.01

OUTPUT RANGES	
Setting for o	**Voltage Value**
0	10
1	5
2	2
3	1

GAIN FACTORS											
Input Ranges (i)										Output Ranges (o)	
0	1	2	3	4	5	6	7	8	9		
1	2	5	10	20	50	100	200	500	1000		0
0.5	1	2.5	5	10	25	50	100	250	500		1
0.2	0.4	1	2	4	10	20	40	100	200		2
0.1	0.2	0.5	1	2	5	10	20	50	100		3

Input and output ranges do not have to be the same. Ranges 0-3 are proportional to, with respect to, each range. A full scale input causes the output to be full scale, regardless of the output range setting. Therefore, a 1 V input signal when the range is set to 1 V input and 10 V output (`R3,0`) results in a 10 V output.

Example:

`PRINT#1,"OUTPUT12;P1 R1,0 X"` Set channel 1 to 5 V input, 10 V output.

1P1 R1,0X

Save/Restore Sn

Type: Unit Command, Deferred

`S0`	Restore setup stored in NVRAM.
`S1`	Save the existing command settings as the power-on default setup in NVRAM.
`S2`	Restores the factory default settings.
`S?`	Responds with the previous `Sn` command.

Save/Restore allows the user to specify the initial configuration of the unit on power-up or Reset (`*R`). This information is stored in non-volatile RAM (NVRAM). The NVRAM contents are preserved even if power is off.

`S2` does not affect NVRAM. To save the restored factory defaults to NVRAM, the sequence `S2 X S1 X` must be entered to return the power-on default settings to the factory defaults.

Example:

`PRINT#1,"OUTPUT 12;P1 G1 F1, 500 R2,2 X"`	Sets channel 1 to a frequency cutoff of 500 Hz with a 2 V input and 2 V output.
`PRINT#1,"OUTPUT 12; S1 X"`	Save existing settings as the power-on default.
`PRINT#1,"OUTPUT 12;F1, 2000 R0,0 X"`	Change frequency to 2000 Hz and input and output ranges to 10 V.
`PRINT#1,"OUTPUT 12;S0 X"`	Restore saved settings. Frequency returns to 500 Hz and input and output ranges return to 2 V.

Status Un

Type: Unit Command, Immediate

U0	Query and clear Event Status Register (ESR)
U1	Query the status byte register (STB)
U2	Query the system settings.
U3	Query all channel settings.
U9	Query product name and revision.
U?	Responds with the last Un command executed.

U0 reads and clears the Event Status Register (ESR) (see Section 4.6 for details on status reporting). The event status register is a read-only register whose bits correspond to those of the event enable register and indicates which events have occurred since the event status register has last been read. It is reset immediately after being read.

It responds with:

Return	Error	Description
004	Query Error	Set when an attempt is made to read data from the output queue when no data are present or data in the output queue were lost. Data may be lost when the output queue buffer overflows (for example, issuing multiple commands to return data without ever reading them).
008	Device Dependent Error	Set when a conflict in programmed parameters is detected. This is also referred to as a conflict error.
016	Execution Error	Set when a parameter exceeds valid limits for a particular command. This is also referred to as Invalid Device Dependent Command Option (IDDCO) error.
032	Command Error	Set when an illegal command is sent to the Filter488. This is also referred to as Invalid Device Dependent Command (IDDC) error.
064	Unused	This bit always returns 0 (reserved for future use).
128	Power On	This bit is set on initial power-up of the Filter488.
nnn		A three-digit number equaling the sum of some or all of the above responses.

U1 responds with the Status Byte Register. This is a copy of the same byte returned in response to a serial poll from the IEEE 488 bus except that bit 6 carries the Master Summary Status rather than the Request for Service. The status byte register is a read-only register whose bits correspond to those of the service request enable register with the addition of bit value 64, which responds with the Master Summary Status (MSS). The MSS indicates whether or not this device needs service.

004	Ready	Set when the Filter488 is ready to process another command. It is cleared when the Filter488 is processing a command line. This bit should be examined with a serial poll prior to issuing a new command line. This allows any detected errors to be traced to the specific command line containing the error. If all the setup information for a specific Filter488 operation is included in one line, this bit also indicates when all processing is done and the X command is completed. This ensures that the Filter488 is done processing all state changes before initiating any further activity.
016	Message Available	Set when the output queue is not empty. It is cleared when the output queue is empty. This bit reflects whether any command responses are still in the output queue.
032	Event Status Register Bit (ESB)	Reflects the logical OR of all the bits in the Event Status Register (ESR) ANDed with their equivalent enable bits in the Event Status Enable (ESE) register. If this bit is set, at least one bit in the ESR is set and has its corresponding enable bit in the ESE set. The status command U0 can be issued to read the ESR. See the following for more information on ESR and ESE.
064	Service Request Bit (SRQ)	Set when the Filter488 is requesting service. It is cleared when an SPoll is performed.
nnn		A three-digit number equaling the sum of some or all of the above responses.

`U2` responds with the present system settings. This is all the information necessary to reconfigure the Filter488 system commands to the same state as the existing state when this command was executed. Its response is in the form:

`DxMxxxNxxxFxxxxx.xFxxxxx.x`

`U3` responds with present channel settings for each channel. Its response is in the form:

`PnAnRn,nGnHnnn`

for each channel.

The `U9` response is an ASCII string identifying the product and the revision and version of the firmware installed in the unit. The response is `"IOtech,Filter488/8,0,v.r"` or `"IOtech,Filter488/4,0,v.r"` where `v` is the version and `r` is the revision of the firmware.

`U?` responds with the last `Un` command executed. The response is `Unn` where `nn` is the number of the most recent query command from 00 to 09.

Execute X

Type: Unit Command

X Execute preceding command string.

The Execute command executes all deferred commands in a command string, takes care of enabled actions such as non-volatile RAM storage, and adds output terminators to any query responses. Deferred commands are interpreted and processed when they are received, but are not executed until an X is received.

Immediate commands do not require an Execute command to be processed. For more detail, see the full description for each command.

If multiple deferred commands that refer to the same setting are used in the same string, each use of the command must be followed by an X. Any number of Execute commands can be inserted into the same command string. If a deferred command is repeated without the X between them, only the second command takes effect. For example, A1 A0 X is the equivalent of A0 X. However, P1 A1 P2 A2 X both take effect, because they are setting different channels.

If errors occur while processing the command string, X has no effect. For example:

If A1 X A0 GGGGGGG X A? X is sent, A1 is the only command that takes effect.

Examples:

`PRINT#1,"OUTPUT 12;P1 R1,1 X"`	Set range of channel 1 to 5V input and output.
`PRINT#1,"OUTPUT 12;P1 R1,1 A1 R0,0X"`	Set range of channel 1 to 5V input and output, and then set range of channel 1 to 10V input and output.

Notes:

Command Summary

Command	Code	Description
Reset	*R	Restores the Filter488 to its initial power-up state.
Analog Coupling	A0	DC coupling (default)
	A1	AC coupling
	A2	Ground inputs; DC output coupling
	A?	Response is present coupling setting for channel selected.
Response Terminator	Dn	Set output terminator used by RS-232C and IEEE 488.
	D0	LF
	D1	CR
	D2	CR-LF (RS-232C default)
	D3	LF-CR
	D4	IEEE 488: LF with EOI; RS-232C: LF (IEEE 488 default).
	D5	IEEE 488: CR with EOI; RS-232C: CR-LF
	D6	IEEE 488: LF-CR with EOI; RS-232C: LF-CR
	D?	Response is Dn, where n is the present termination setting, a one digit number from 0-7.
Error Query	E?	Reports and clears the current contents of the error source register. After execution of the Error Query command, Filter488 responds with one of the following error codes:
	E000	No error has occurred.

Command	Code	Description
	E001	Invalid device dependent command (IDDC). Due to a syntax error.
	E002	Invalid device dependent command option (IDDCO). A command parameter was out of range or missing.
	E004	Command confilct.
	E008	Self-test failure. Reserved for non-critical internal errors.
	E032	Non-volatile RAM failure. Access to the non-volatile RAM was impossible, or the data retrieved were corrupt.
	Ennn	If two or more errors occurred, nnn is the sum of the corresponding error codes.
Frequency Cutoff	Fn,xxxxx.x	Set the cutoff for the selected clock. Where n is system clock 1 or 2 and frequency range, xxxxx.x is 1-50,000.0 Hz (10k Hz default).
	F?	Responds with both actual cutoff frequencies.
System Clock Select	G0	Sets the selected channel to filter bypass mode. All frequencies passed (default).
	G1	Sets the system clock #1 to use as the cutoff frequency for selected channel.
	G2	Sets system clock #2 to use as the cutoff frequency for selected channel.
	G?	Responds with system clock # programmed for selected channel.
Offset Constant	Hn	Sets offset constant to n (0-255) for the selected channel (factory default is 128).
	H?	Responds with offset constant for the selected channel.
SRQ Mask	M0	Clear Service Request Enable Register (default).
	M4	SRQ on ready.

Command	Code	Description
	M16	SRQ on message available.
	M32	SRQ on event.
	M?	Response is Mnnn, where nnn is the service request enable mask.
Event Mask	N000	Clear event mask (default).
	N004	Enable query error.
	N016	Enable execution error.
	N032	Enable command error.
	N128	Power-on.
	N?	Read standard event status enable register.
Select Channel	Pn	Selects the channel subsequent commands affect. n is 1-8 for the Filter488/8, 1-4 for the Filter488/4.
	P?	Responds with Pn, where n is the presently selected channel or 0 if all channels are selected.
Range	Ri,o	Sets input and output range for the channel specified by Pn (R0,0 default).
	R?	Responds with the present input and output range settings for the channel specified by the Pn command.
Save/Restore	S0	Restore setup stored in NVRAM.
	S1	Save the existing command settings as the power-on default setup in NVRAM.
	S2	Restores the factory default settings.
	S?	Responds with the previous Sn command.
Status	U0	Query and clear Event Status Register (ESR).
	U1	Query the Status Byte Register (STB).
	U2	Query the system settings.

Command	Code	Description
	U3	Query all channel settings.
	U9	Query product name and revision.
	U?	Responds with the last Un command executed.
Execute	X	Execute preceding command string.

Character Codes and IEEE Multiline Messages

$00 NUL 0	$10 DLE 16	$20 SP 32 00	$30 0 48 16	$40 @ 64 00	$50 P 80 16	$60 ` 96 SCG	$70 p 112 SCG
$01 SOH 1 GTL	$11 DC1 17 LLO	$21 ! 33 01	$31 1 49 17	$41 A 65 01	$51 Q 81 17	$61 a 97 SCG	$71 q 113 SCG
$02 STX 2	$12 DC2 18	$22 " 34 02	$32 2 50 18	$42 B 66 02	$52 R 82 18	$62 b 98 SCG	$72 r 114 SCG
$03 ETX 3	$13 DC3 19	$23 # 35 03	$33 3 51 19	$43 C 67 03	$53 S 83 19	$63 c 99 SCG	$73 s 115 SCG
$04 EOT 4 SDC	$14 DC4 20 DCL	$24 $ 36 04	$34 4 52 20	$44 D 68 04	$54 T 84 20	$64 d 100 SCG	$74 t 116 SCG
$05 ENQ 5 PPC	$15 NAK 21 PPU	$25 % 37 05	$35 5 53 21	$45 E 69 05	$55 U 85 21	$65 e 101 SCG	$75 u 117 SCG
$06 ACK 6	$16 SYN 22	$26 & 38 06	$36 6 54 22	$46 F 70 06	$56 V 86 22	$66 f 102 SCG	$76 v 118 SCG
$07 BEL 7	$17 ETB 23	$27 ' 39 07	$37 7 55 23	$47 G 71 07	$57 W 87 23	$67 g 103 SCG	$77 w 119 SCG
$08 BS 8 GET	$18 CAN 24 SPE	$28 (40 08	$38 8 56 24	$48 H 72 08	$58 X 88 24	$68 h 104 SCG	$78 x 120 SCG
$09 HT 9 TCT	$19 EM 25 SPD	$29) 41 09	$39 9 57 25	$49 I 73 09	$59 Y 89 25	$69 i 105 SCG	$79 y 121 SCG
$0A LF 10	$1A SUB 26	$2A * 42 10	$3A : 58 26	$4A J 74 10	$5A Z 90 26	$6A j 106 SCG	$7A z 122 SCG
$0B VT 11	$1B ESC 27	$2B + 43 11	$3B ; 59 27	$4B K 75 11	$5B [91 27	$6B k 107 SCG	$7B { 123 SCG
$0C FF 12	$1C FS 28	$2C , 44 12	$3C < 60 28	$4C L 76 12	$5C \ 92 28	$6C l 108 SCG	$7C | 124 SCG
$0D CR 13	$1D GS 29	$2D – 45 13	$3D = 61 29	$4D M 77 13	$5D] 93 29	$6D m 109 SCG	$7D } 125 SCG
$0E SO 14	$1E RS 30	$2E . 46 14	$3E > 62 30	$4E N 78 14	$5E ^ 94 30	$6E n 110 SCG	$7E ~ 126 SCG
$0F SI 15	$1F US 31	$2F / 47 15	$3F ? 63 UNL	$4F O 79 15	$5F _ 95 UNT	$6F o 111 SCG	$7F DEL 127 SCG
ACG	UCG	LAG		TAG		SCG	

ACG = Addressed Command Group
UCG = Universal Command Group
LAG = Listen Address Group
TAG = Talk Address Group
SCG = Secondary Command Group

Notes: