



ACCES I/O PRODUCTS, INC.  
10623 Roselle St. San Diego, CA 92121-1506  
Tel: (858) 550-9559 FAX: (858) 550-7322

---

**REMOTE DAS POD**  
**Model RA1216**

**USER MANUAL**

## NOTICES

The information in this document is provided for reference only. ACCES I/O Products Inc does not assume any liability arising out of the application or use of the information or products described herein. This document may contain or reference information and products protected by copyrights or patents and does not convey any license under the patent rights of ACCES, nor the rights of others.

Printed in USA. Copyright 1999 by ACCES I/O Products Inc, 10623 Roselle Street, San Diego, CA 92121. All rights reserved.

# TABLE OF CONTENTS

INSTALLATION .....	1-1
CD INSTALLATION.....	1-1
3.5-INCH DISKETTE INSTALLATION.....	1-1
DIRECTORIES CREATED ON THE HARD DISK.....	1-2
CALIBRATION.....	1-4
HARDWARE INSTALLATION.....	1-5
INPUT/OUTPUT CONNECTIONS.....	1-6
50-Pin Connector.....	1-6
GETTING STARTED.....	1-8
FUNCTIONAL DESCRIPTION.....	2-1
ANALOG INPUTS.....	2-2
PROGRAMMABLE OFFSET.....	2-2
ANALOG OUTPUTS.....	2-2
DIGITAL I/O.....	2-2
WATCHDOG TIMER.....	2-3
BLOCK DIAGRAM.....	2-3
SOFTWARE.....	3-1
GENERAL.....	3-1
COMMAND STRUCTURE.....	3-1
Addressed Mode.....	3-1
Concepts: Onboard Digital I/O.....	3-1
Concepts: The Point List Buffer.....	3-2
RA1216 COMMAND LIST.....	3-3
COMMAND FUNCTIONS.....	3-5
Configure Point List.....	3-5
Read Point List.....	3-5
Write Point List to EEPROM.....	3-5
Restore Point List.....	3-5
Write Calibration Parameters.....	3-5
Return Calibration Parameters.....	3-6
Set Sample Rate.....	3-6
Read Sample Rate.....	3-7
Configure Bits as Input or Output.....	3-7
Read Digital Inputs.....	3-8
Write Digital Outputs.....	3-8
Read Firmware Revision Number.....	3-9
Resend Last Response.....	3-9
Configure Baud Rate.....	3-10
Configure Pod Address.....	3-11
Address Select.....	3-11
Enter a New Program.....	3-11
SPECIFICATIONS.....	4-1
SERIAL COMMUNICATIONS INTERFACE.....	4-1
ANALOG INPUTS.....	4-1
DIGITAL I/O.....	4-2
POWER REQUIRED.....	4-2
WARRANTY.....	5-1
APPLICATION CONSIDERATIONS.....	A-1

Files contained on the 3.5 inch diskette are as follows:

**SETUP & CALIBRATION PROGRAM:** A menu-driven program that assists you in configuring and calibrating your RA1216. The **SETUP.EXE** program instructs you on the use of the serial opto-isolation and DAC Offset jumpers and allows you to specify the Pod Address and the serial baud rate. This program can also auto-detect the address and/or baud rate of an attached card in the host computer. Finally, a calibration procedure tests the unit and determines the calibration correction values and stores them into internal memory. A sample program is provided to demonstrate the use of the calibration feature.

The Setup program should be run with only a single Pod attached to the RS-485 port in your computer. You will need to know the setup information (base address, IRQ) of the port so that the Setup program can access it. The Setup program will create a batch file in its directory, called **RISC.BAT**, that can be used to run RISCterm.

**RISCterm:** This program provides a simple, non-automated interface to RS-485 devices and can be used to configure or test RA1216 setup and operations. Context sensitive help on using RISCterm is provided by pressing **h**.

**DRIVERS:** COMMDRV.PAS, COMMDRV.H, and COMMDRV.C are communication shortcuts or drivers provided in C and Pascal. Three functions are provided: "initComCard", "writePod", and "readPod". The functions are polling based, no IRQ necessary, and deal with strings rather than character-by-character. Consult the source code files for more information.

**SAMPLE:** This program demonstrates reading and displaying A/D channels and polling the Pod for general information. The sample also demonstrates use of the driver and the software driven calibration procedure.

## CALIBRATION

The setup software provided with the RA1216 supports the ability to check calibration and to write correction values into EEPROM so they are available automatically on power-up. Calibration checks need only be performed periodically, not every time power is cycled. The SETUP.EXE software calibration procedure requires the ability to ground an analog channel and provide a voltage source set at 5.000V and another for 0.250V.

The SAMPLE1 program illustrates the procedure of recalling these values and adjusting the readings. The command descriptions of CALn? CALn=xxxx,xxxx and CAL=BACKUP provide instructions on adding the calibration functions to your own software. See the Software section for a description on calibrating the A/D input and DAC output using the values from the calibration procedure.

## HARDWARE INSTALLATION

The RA1216 enclosure is a sealed, die-cast, aluminum-alloy, NEMA-4 enclosure that is easily mounted using two diagonal holes accessible when the cover is removed. Outside dimensions of the enclosure are: 8.74" long by 5.75" wide by 2.17" high. The cover incorporates a recessed neoprene gasket and the cover is secured to the body by six recessed M-4, stainless steel, captive screws. The mounting holes and cover-attaching screws are outside the sealed area to prevent ingress of moisture and dust. Four threaded bosses inside the enclosure provide for mounting the printed circuit board assemblies.

### Micro-controller board setup:

JP2, JP3, and JP4:

**ISL** position TXD, RXD and TXEN are opto-isolated from the host computer. In the **/ISL** position the opto-isolators are by-passed. This position allows for a single power supply and higher baud rates.

### Personality board setup:

OFFSET DAC jumper:

**OFST** position, analog inputs are offset by a value set by the user.

**GND** position, the OFFSET DAC has no effect on the analog input readings.

## INPUT/OUTPUT CONNECTIONS

Electrical connections to the RA1216 are through watertight glands that seal the wires terminated inside to a Euro-style, screw-terminal block that plugs into a 50-pin connector. To ensure that there is minimum susceptibility to EMI and minimum radiation, it is important that there be a positive chassis ground. Also, proper EMI cabling techniques (cable connected to chassis ground, twisted pair wiring, and, in extreme cases, ferrite-level of EMI protection) may be needed for input/output wiring.

### 50-Pin Connector

Connector pin assignments for the 50-pin connector are listed below.

Pin	Signal	Pin	Signal
1	RS485 Comm Line (+)	2	/RS485 Comm Line (-)
3	DIO1 Digital I/O Bit 1	4	DIO0 Digital I/O Bit 0
5	DIO3 Digital I/O Bit 3	6	DIO2 Digital I/O Bit 2
7	DIO5 Digital I/O Bit 5	8	DIO4 Digital I/O Bit 4
9	APPLV+ Ext. Dig. Out Power	10	DIO6 Digital I/O Bit 6
11	/PBRST Ext. Reset for $\mu$ Ctrlr	12	ISOV+ Isol. Pwr +
13	ISOGND Isol. Pwr. Ground	14	PT0 Input to Counter 0
15	V <sub>CC</sub> 10 10V Pwr. (Ext.Sens.)	16	PINT1 Interrupt 1
17	PWR+ Local Pwr (+)	18	GND Digital Pwr Ground
19	AOGND Analog Output Ground	20	GND Local Pwr. Ground
21	AOUT0 Analog Output 0 (Volt)	22	IOOUT0 Analog Output 0 (Current)
23	AOUT1 Analog Output 1 (Volt)	24	IOOUT1 Analog Output 1 (Current)
25	AOUT2 Analog Output 2 (Volt)	26	IOOUT2 Analog Output 2 (Current)
27	AGND Analog Input Ground	28	AGND Analog Output Ground
29	CH00(+) Analog Input Chl 0 (+)	30	CH04(+) Analog Input Chl 4 (+)
31	CH00(-) Diff. Analog Input Chl 0(-) Or S.E. Analog Input Chl 8	32	CH04(-) Diff. Analog Input Chl 4(-) Or S.E. Analog Input Chl 12
33	AGND Analog Input Ground	34	AGND Analog Input Ground
35	CH01(+) Analog Input Chl 1 (+)	36	CH05(+) Analog Input Chl 5 (+)
37	CH01(-) Diff. Analog Input Chl 1(-) Or S.E. Analog Input Chl 9	38	CH05(-) Diff. Analog Input Chl 5(-) Or S.E. Analog Input Chl 13
39	AGND Analog Input Ground	40	AGND Analog Input Ground
41	CH02(+) Analog Input Chl 2 (+)	42	CH06(+) Analog Input Chl 6 (+)
43	CH02(-) Diff. Analog Input Chl 2(-) Or S.E. Analog Input Chl 10	44	CH06(-) Diff. Analog Input Chl 6(-) Or S.E. Analog Input Chl 14
45	AGND Analog Input Ground	46	AGND Analog Input Ground
47	CH03(+) Analog Input Chl 3 (+)	48	CH07(+) Analog Input Chl 7 (+)
49	CH03(-) Diff. Analog Input Chl 3(-) Or S.E. Analog Input Chl 11	50	CH07(-) Diff. Analog Input Chl 7(-) Or S.E. Analog Input Chl 15

**Pin Description:**

**PWR+ and PWR GND:** These terminals are used to apply power to the pod from a local power supply. The voltage can be anywhere in the range of 12 VDC to 16 VDC. Higher voltage can be used, 24 VDC for example, if an external Zener diode is used to reduce the voltage applied to the RA1216. (See the Specification section of this manual to determine the Zener diode power rating required.)

**ISOV+ and ISOGND:** This is the power connection for the isolator section that may be supplied from the computer's +12VDC supply via a pair of wires on the RS-485 network *or* from a central power supply. This power is independent of "local power". The voltage level can be from 7.5 VDC to 35 VDC. (An on-board voltage regulator regulates the power to +5 VDC.) The RA1216 requires only 5mA of current when idling and ~33mA current when data is being transmitted so any loading effects on the computer power will be minimal.

Note: If separate power is not available, ISOV+ and ISOGND must be jumpered to the "local power" terminals, which defeats the optical isolation.

**RS485+ and RS485-:** These are the terminals for RS-485 serial communications.

**CH0-7(+/-) and AGND:** Analog voltage input terminals.

**AOUT0-2 and AOGND:** Analog voltage output terminals.

**IOUT0-2:** 4 to 20mA current output terminals.

**DIO0-6:** Digital input or output terminals.

**APPLV+:** This terminal is for the "application power" or the user- provided voltage level to which digital outputs are connected through the loads. Open-collector Darlington amplifiers are used at the outputs. Inductive suppression diodes are included in the APPLV circuit. If inductive loads are being driven, this must be connected. The application power level (APPLV) can be as high as 50 VDC.

# FUNCTIONAL DESCRIPTION

## FEATURES

- ! Remote Intelligent Analog and Digital I/O Unit with Opto-Isolated RS-485 Serial Interface to Host Computer.
- ! Eight Differential or Sixteen Single-ended Analog Inputs, Programmable Gain and Offset on a Channel-by-Channel Basis, 12-Bit Resolution A/D Converter.
- ! Optional Signal Conditioning for Use with Thermocouples, RTDs, 4-20 mA Current Inputs, and Resistors for Strain Gage Bridge Completion.
- ! Three 12-Bit D/A Converters with Both Voltage and Current Outputs.
- ! Seven Bits of Digital I/O Configured on a Bit-by-Bit Basis as Either Inputs or High-Current Outputs.
- ! Onboard 16-Bit, 8031 Compatible, Microcontroller with 32K x 8 bits RAM and 32K x 8 bits EEPROM.
- ! All Programming and Calibration in Software, No Switches to Set. Jumpers available to By-Pass Opto-Isolators if Desired and for Calibration Purposes.
- ! Protective NEMA-4 Enclosure for Harsh Atmospheric and Marine Environments.

## COMPUTER COMMUNICATIONS

A type DS80C310 microcontroller (with 32K x 8 bits SRAM, 32K x 8 bits non-volatile EEPROM, and a watchdog timer circuit) gives RA1216 the capability and versatility expected from a modern distributed control system. RA1216 contains CMOS low-power circuitry, an optically-isolated receiver/transmitter, and power conditioners for local and external isolated power. The pod can operate at baud rates up to 57.6 K baud and distances up to 4000 feet with low-attenuation twisted-pair cabling, such as Belden #9841 or equivalent. Data collected by the pod can be stored in local SRAM and accessed later through the computer's serial port. This facilitates a stand-alone mode of operation.

All programming of RA1216 is in ASCII-based software. ASCII-based programming is flexible because it permits you to write applications in any high-level language that supports ASCII string functions. You can use REMOTE ACCES Series pods with virtually any computer that has an RS-485 port.

The module, or pod, address is programmable from 00 to FF hex. Whatever address is assigned is stored in EEPROM and used as the default address at the next Power-ON. Similarly, the baud rate is programmable, is stored in EEPROM, and is used as the default at the next Power-ON.



## ANALOG INPUTS

RA1216 uses a 12-bit successive approximation analog-to-digital converter (A/D) capable of up to 50,000 conversions per second. A/D conversions may be initiated either by software command or by an external input.

The RA1216 accepts up to sixteen single-ended analog inputs (or eight differential). Solid-state multiplexers are used and, to prevent crosstalk and increase stability between channels, the input of the multiplexer is shorted to ground before each channel selection for a time period sufficient to discharge the multiplexer's output capacitance.

The RA1216 can be configured to accept inputs from a variety of sensor types. The standard configuration is for voltage inputs. For thermocouple inputs, an on-board temperature sensor is used to provide input for cold junction compensation, and break detect resistors are available (specify RA1216T/C). The pod can accept inputs from three wire RTDs or 4-20 mA transmitter circuits (RA1216R, or RA1216I respectively). It can also be configured to provide bridge completion in quarter bridge Strain Gage applications (RA1216B/C).

Eight analog voltage spans and a 4-20 mA current input range are available. All inputs are internally protected up to 200 VDC. The Programmable Gain Amplifier consists of a programmable gain instrumentation amplifier and a programmable gain operational amplifier. This increases the signal conditioning flexibility of the pod.

## PROGRAMMABLE OFFSET

Each input can be programmed with an offset of  $\pm 2.5V$  generated by a digital-to-analog converter. This allows easy scaling and shifting of an input signal to take advantage of the pod's full input ranges.

## ANALOG OUTPUTS

Three 12-bit digital-to-analog converters provide simultaneous voltage and current outputs. Voltage outputs are independently programmable in two ranges: 0-5V, and 0-10V. Simultaneously, each D/A can provide 4-20 mA outputs. These current outputs are derived for user-supplied compliance voltages from 5.5V to 30V.

## DIGITAL I/O

Seven bits of digital I/O are provided. These can be software configured on a bit-by-bit basis as either inputs or outputs. Digital inputs are sensed at TTL levels. There is one interrupt input.

Each digital output provides compliance with user-supplied voltages up to 50 VDC and the maximum sink current per output bit is 350 mA. The maximum total sink current is limited to 650 mA on all seven bits.

All bits have a factory installed 10K pull up resistor to 5Volts.

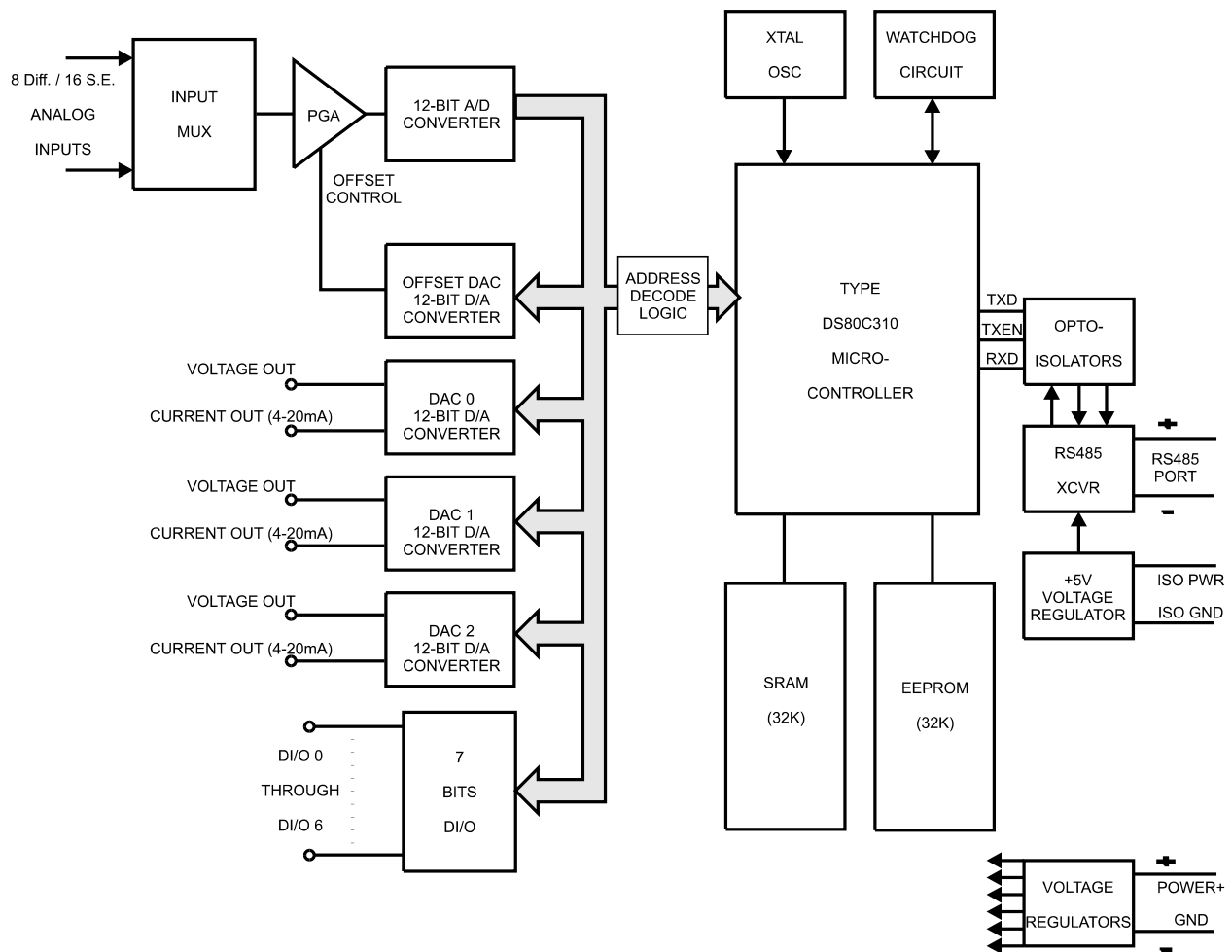
## WATCHDOG TIMER

The built-in watchdog timer resets the pod if the microcontroller “hangs up” or the power supply voltage drops below 7.5 VDC. The microcontroller can also be reset by an external manual pushbutton input available at the interface connector.

**POWER SUPPLIES**

RA1216 requires two power sources to maintain opto-isolation. A small amount of +7.5V to +24V power (50 mA) is required by the opto-isolated RS485 circuitry, this can be supplied by the host computer through the cable carrying the RS485 serial data. Alternatively, you can use an external power supply. The remainder of the RA1216 circuitry is powered by an external power supply of +12V to +16V (500mA).

**BLOCK DIAGRAM**



# SOFTWARE

## GENERAL

The RA1216 comes with ASCII-based software provided on CD or diskette. ASCII programming permits you to write applications in any high-level language that supports ASCII text string functions. This allows REMOTE ACCES series modules to be used with virtually any computer that has an RS-485 port.

The communication protocol has two forms: addressed and non-addressed. Non-addressed protocol is used when there is only one REMOTE ACCES pod. Addressed protocol must be used when there is more than one REMOTE ACCES pod. The difference is that an address command is sent to enable the specific pod. The address command is only sent once during communication between the specific pod and the host computer. It enables communication with that specific pod and disables all other REMOTE ACCES devices from communicating on the network.

## COMMAND STRUCTURE

All communication must be 7 data bits, even parity, 1 stop bit. All numbers sent to and received from the pod are in hexadecimal form. The factory default baud rate is 9600 Baud. The pod is considered to be in addressed mode any time its pod address is not 00. The factory default pod address is 00 (non-addressed mode).

### Addressed Mode

The address select command must be issued before any other command to the addressed pod. The address command is as follows:

“!xx[CR]”      where xx is the pod address from 01 to FF hex, and [CR] is Carriage Return, ASCII character 13.

The pod responds with “[CR]”. Once the address select command has been issued, all further commands (other than a new address select) will be executed by the selected pod. *The addressed mode is required when using more than one pod.* When there’s only one pod connected, no address select command is needed.

### Command Structure

Pod commands are listed in the following table. Terminology used is as follows:

- a. The single lower case letter ‘x’ designates any valid hex digit (0-F).
- b. The symbol ‘±’ designates either a ‘+’ or a ‘-’.
- c. All commands are terminated with [CR], the ASCII character 13.
- d. All commands are not case-sensitive, i.e., upper or lower case may be used.
- e. The symbol ‘\*’ means zero or more valid characters (total msg length < 255 decimal)

**Concepts: Onboard Digital I/O**

The RA1216 has 7 digital inputs/outputs. The outputs are on the same pins as the inputs. They comprise the DI/O port.

**Concepts: The Offset DAC**

The Analog inputs include an analog offset capability. This allows you to create bipolar ranges, or ranges offset above ground, to maximize the effective number of bits for your sensor output.

The amount of offset is controlled by DAC3, the Offset DAC. When the OFST jumper is in the OFST position the output of this DAC is fed to the negative-input of the A/D operational amplifier, effectively shifting the zero-point of the A/D by a proportional amount.

0x800 counts is “no offset”, or zero volts. The offset is not affected by the analog input gain code. 0xC00 counts is -2.5V offset, useful to achieve a ±2.5V input range when using Gain Code 0. 0xFFF produces a -5V offset, producing a -5V to +0V unipolar input range when using Gain Code 0.

Note: The Offset DAC can be measured with a DVM at the OFST jumper for calibration purposes.

**Concepts: The Point**

A Point, as used in the RA1216, is a channel number and its associated gain, Diff/SE, and offset settings. The Point defines the A/D channel number, what gain should be applied, whether it is single-ended or differential, and what setting is desired on the Offset DAC. The format of the 3-byte Point is as follows:

D23	D22	D21	D20	D19	D18	D17	D16	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
use 0	GAIN			CHANNEL				Diff/SE	unused use 0s		Offset DAC count value												

A typical Point might be 308800, which would translate to gain code 3, channel 0, Differential, with zero volts on the Offset DAC (see the Offset DAC section for more info).

GAIN CODE	GAIN	BASE A/D RANGE Volts	½ Offset DAC counts	Bipolar Range V
0	1	0 - 5	C00	±2.5
1	2	0 - 2.5	A00	±1.25
2	5	0 - 1.0	8CC	±0.5
3	10	0 - 0.5	866	±0.25
4	20	0 - 0.25	833	±0.125
5	40	0 - 0.125	819	±0.0625
6	100	0 - 0.05	80A	±0.025
7	200	0 - 0.025	805	±0.0125

The “BASE A/D RANGE” column indicates the input span associated with each gain code, assuming the offset DAC voltage is zero V. The offset DAC will shift this base range by the amount shown.

The Diff/SE bit will configure the point for differential or single-ended (1=differential, 0=single-ended) use. Please note: points configured for differential should come before points configured for single-ended, if both types of points will be acquired with the same command. (Differential takes a moment to engage.)

**Concepts: The Point List Buffer**

Because each channel can be acquired at a different gain and Offset DAC setting, some setup is necessary to improve usability. The setup consists of telling the Pod the input range and Diff/SE for each channel and the order in which to acquire the channels. This is done using the Point List (PL) series of commands. The Point List is a list of the channels to be acquired, along with all of the configuration information for each channel. All of the commands to set PL entries take a point number (or the keyphrase “ALL”), and a 3-byte (6 hex digit) value describing the configuration for that point (or the keyphrase “DEFAULT”).

ALL numbers passed to and from the Pod are in hexadecimal.

**RA1216 COMMAND LIST**

Command	Description	Returns
PROGRAM=	<b>Use with caution.</b> Loads new firmware into the Pod's EEPROM. Use ESC (ASCII 27) to abort.	n/a
(Vertical Bar)	Same as PROGRAM=, except used only after a reset to gain access to Pod.	n/a
BAUD=nnn	Sets Baud rate	=:Baud:0n
POD=xx	Sets the Pod address to xx	=:Pod#xx
S=xxxx	Sets the Sample Rate of the A/D. Valid ranges are 00A2 through FFFF. Value passed is the desired divisor of the rate clock. 00A2 is 6666Hz, or 150us/sample. 039A is 1000Hz, or 1ms/sample. (11,059,200Hz / 12 / timebase = Hz sample rate.	[CR]
S?	Returns the Clock Divisor from EEPROM on the Pod, as above.	xxxx[CR]
!xx	Address Select Command, Selects the Pod at address 'xx' for communication.	[CR]
H*	“Hello” command. H followed by anything returns Pod name, revision, and copyright message.	See Desc.
N	Resend last message. Causes the Pod to re-transmit last message. Should be used upon parity-error, for example.	See Desc.
V	Reads the version number from the firmware.	x.xx[CR]

PLnn=xxxxxx	Set Point List Entry for point number nn to xxxxxx. See <b>Concepts: The Point</b> for more information on the point configuration setup required to use the PL commands.	[CR]
PLnn?	Reads the Point List entry for point number nn.	XXXXXX[CR]
PLALL?	Reads the Point Configuration for all points (0-40 hex). Data is returned as XXXXXX XXXXXX ... XXXXXX XXXXXX[CR] with a total of 64 entries.	See Desc.
PLnn=DEFAULT	Sets point nn to default configuration (5V span, no offset, S.E.)	[CR]
PLALL=DEFAULT	Sets all points to default configuration (5V span, no offset, entries 0-F = A/D channel 0-F S.E., all other entries= AD channel 0)	[CR]
BACKUP=PL	Stores the point list into the onboard EEPROM, for power-off safekeeping. The point list is restored automatically every time the Pod is reset, or using the PLALL=BACKUP command.	[CR]
PLALL=BACKUP	Restores the EEPROM point list into the current point list. This command is executed automatically upon Pod reset.	[CR]
CALn=xxxx,xxxx	Writes offset and scale calibration constants #n used for $Y=MX+B$ calibration. Used by setup/calibration procedures.	[CR]
CALn?	Returns stored calibration constant # n. See sample program on disk for example usage.	xxxx,xxxx [CR]
CAL=BACKUP	Use this command to restore the factory calibration, overwriting the current user calibration. The factory cal is stored permanently in EEPROM during the initial calibration.	[CR]
Mxx	Set bits as Inputs or Outputs. Each bit of xx controls a single bit of the DIO port. One's set the corresponding port as output, zeroes set the port as input. Writing to a port marked as input will fail. Power-on default is ALL INPUTS.	[CR]
Mx+	Turns bit x to output.	[CR]
Mx-	Turns bit x to input.	[CR]
Oxx	Writes xx to the DI/O port.	[CR]
Ox+	Sets output bit x (0-6) to ONE.	[CR]
Ox-	Sets output bit x to ZERO.	[CR]
I	This command reads the seven digital inputs	xx[CR]
AS	Set the unit into Single Ended (S.E.) mode	S[CR]
AD	Set the unit into Differential (Diff) mode	D[CR]
AA	Acquire and report all channels at Gain x1 using the last configured offset DAC and Diff / S.E values. The data is returned immediately, as it is acquired, as "XXXX XXXX ... XXXX [CR]"	See desc.
In	Single-bit read from bit n, $0 \leq n \leq 6$ .	0[CR] or 1[CR]

Axxxxxx	Acquires a single channel and returns the data immediately via the serial port, bypassing the point list. This command takes as its sole argument the same 3-byte parameter passed to the PL series of commands. The channel specified in the parameter is acquired using the range, polarity, and MUX gain specified. The data is returned immediately, unlike the AC command.	XXXX[CR]
ACnn <sub>1</sub> -nn <sub>2</sub> ,xxxx	Acquires point list entries <i>nn<sub>1</sub></i> through <i>nn<sub>2</sub></i> , with a total of <i>xxxx</i> conversions performed. <i>xxxx</i> must be equal to or less than 2710hex (which is 10000 decimal--all numbers must be presented to the Pod in hex). 10000 data samples is the maximum internal storage capacity of the standard Pod. (Optional firmware can increase this capacity. Contact the factory for further information.) Please note that this function does not return any data. Data is stored in internal SRAM storage for later retrieval using the R command.	[CR]
Ann <sub>1</sub> -nn <sub>2</sub> ,xxxx	The A command works very similarly to the AC command. All parameters are the same. However, the A command runs in the foreground, occupying all of the attention of the Pod's microcontroller. This yields higher throughput, but sacrifices flexibility. The Pod acquires data at its fastest possible rate. The data is returned immediately as if the "R" command had been issued. Avoid issuing commands to the device until the data has been received.	See R
R*	This command reads the last buffer of data acquired using the AC or A command. The data is returned in CCXXXX CCXXXX ... CCXXXX CCXXXX[CR] format, where CC is the point number that was acquired, and XXXX is the analog conversion data. The total number of return values is identical to the XXXX parameter passed to the most recently issued AC command.	see desc.
An=mxxx	Set analog output <i>n</i> to <i>xxxx</i> , with 0-5V range if <i>m</i> is 0, or 0-10V range if <i>m</i> is 1. <i>xxx</i> is hexadecimal counts; 000 is 0V, FFF is full scale(5V or 10V), 800 is half scale(2.5V or 5V), etc. Note, <i>n</i> =3 will control the Offset DAC, but the range is fixed at ±5V	[CR]
AA=mxxx	Similar to the An= command, but sets all three analog outputs to the same value.	[CR]

NOTE: Pod reset occurs upon power-up, programming process, or watchdog time-out.

### COMMAND FUNCTIONS

The following paragraphs give details of the command functions, describe what the commands cause, and give examples. Please note that all commands have an acknowledgment response. You must wait for a response from a command before sending another command.

#### Configure Point List

- PLnn=xxxxxx      Sets point list entry for point number *nn* to *xxxxxx*.
- PLnn=DEFAULT    Sets point *nn* to default configuration.

PLALL=DEFAULT Sets all points to default configuration.

NOTE: Default configuration is 0-5V, Single Ended, 0V Offset DAC.

### Read Point List

PLnn? Reads the point configuration for point number nn.

PLALL? Reads the point configuration for all points (0-127).

NOTE: Data is returned as xxxxxx xxxxxx... xxxxxx[CR] with a total of 64 entries.

### Write Point List to EEPROM

BACKUP=PL Stores the point list configuration in the onboard EEPROM, for power-off safekeeping.

NOTE: The point list configuration is automatically restored every time the Pod is reset, or when PLALL=BACKUP command is issued.

### Restore Point List

PLALL=BACKUP Restores point list configuration in EEPROM to the current point list.

NOTE: This command is executed automatically upon Pod reset.

### Write Calibration Parameters

CALn= bbbb,aaaa Write offset and scale calibration values for  $Y=mX+b$  calibration in two's-complement hex as two four-digit numbers. The full equation for calibrated input to the A/D is  $Y=(4095/(4095-a-b))X-b$ . The equation for calibrated DAC output is  $Y=((4095-a-b)/4095)X+b$ . In both equations, b is the number of counts below zero and a is the number of counts over full scale. CAL entries 0-7 use the A/D equation. Entries 9-14 use the DAC equation.



The Pod stores 16 calibration M/b pairs, one each for the 8 gain codes, one for the Offset DAC, one each for the 3 DACs at both Gains (6 total) and one for the onboard Reference. The following table shows the values for n for each calibration constant:

Entry number (n)	Contains calibration constants for:
0	AD Gain 0
1	AD Gain 1
2	AD Gain 2
3	AD Gain 3
4	AD Gain 4
5	AD Gain 5
6	AD Gain 6
7	AD Gain 7
8	Offset DAC
9	DAC 0 Gain 0 (0-5V)
10	DAC 1 Gain 0
11	DAC 2 Gain 0
12	DAC 0 Gain 1 (0-10V)
13	DAC 1 Gain 1
14	DAC 2 Gain 1
15	Onboard Reference (factory use)

This function stores the values required to adjust the measurement readings to agree with the last calibration. The SETUP program will measure and write these calibration parameters. The SAMPLE1 program illustrates using the CALn? command with the results of this function.

### Return Calibration Parameters

CALn?      Recalls the scale and offset calibration constants for  $Y=mX+b$  calibration stored using the CALn=xxxx,xxxx command.

This function recalls the stored calibration constants that may be used by your software to adjust the measurement readings to agree with the last calibration. The SAMPLE1 program illustrates the use of this command.

### Set Sample Rate

S=xxxx      Set Sample Rate of the A/D Converter

This function sets the sample rate of the A/D converter. Valid values range from 00A2 to FFFF. The value passed is the desired divisor of the rate clock (11.0592 MHz). The equation to use in calculating the divisor is:

$$\text{Divisor} = [(1/\text{Rate}) - 22\mu\text{Sec}] * [\text{Clock}/12]$$

Examples:

Program the RA1216 for 1K samples per second.

```
SEND:            S=0385
RECEIVE:        [CR]
```

NOTE: The sample rate configured is stored in EEPROM on the Pod and will be used as the default (power-on) sample rate. The factory default sample rate (100Hz) can be restored by sending "S0000" to the Pod.

## Read Sample Rate

S? Returns the clock Divisor from EEPROM on the Pod.

## Special Acquisition Commands

ACnn<sub>1</sub>-nn<sub>2</sub>,xxxx: Acquires Point List entries nn<sub>1</sub> through nn<sub>2</sub>, with a total of xxxx conversions performed.

xxxx must be equal to or less than 2710hex (which is 10000 decimal--all numbers must be presented to the Pod in hex). 10000 data samples is the maximum internal storage capacity of the standard Pod. (Optional firmware can increase this capacity. Contact the factory for further information.) Please note that this function does not return any data. Data is stored in the internal SRAM storage for later retrieval using the R command.

R: This command reads the last buffer of data acquired using the AC command.

The data is returned in CCXXXX CCXXXX ... CCXXXX CCXXXX[CR] format, where CC is the point number that was acquired, and XXXX is the analog conversion data. The total number of return values is identical to the XXXX parameter passed to the most recently issued AC command.

Axxxxxx: Acquires a single channel and returns the data immediately via the serial port, bypassing the point list.

This command takes as its sole argument the same 3-byte parameter passed to the PL series of commands. The channel specified in the parameter is acquired using the range, polarity, and MUX gain specified. The data is returned immediately, unlike the AC command.

Ann<sub>1</sub>-nn<sub>2</sub>,xxxx The A command works very similarly to the AC command. All parameters are the same. However, the A command runs in the foreground, occupying all of the attention of the Pod's microcontroller. This yields higher throughput, but sacrifices flexibility. The Pod acquires data at its fastest possible rate and returns data as if an "R" command had been issued. Avoid issuing commands to the pod until the data is received.

## Configure Bits as Input or Output

Mxx Configures digital bits as inputs or outputs.  
 Mx+ Configures digital bit 'x' as output.  
 Mx- Configures digital bit 'x' as input.

These commands program the digital bits, on a bit-by-bit basis, as input or output. A "zero"

in any bit position of the xx control byte designates the corresponding bit to be configured as an input. Conversely, a “one” designates a bit to be configured as an output. (Note: Any bit configured as an output can still be read as an input if the current value output is a “one”.

Example:

Program even bits as outputs, and odd bits as inputs.

```
SEND:      MAA
RECEIVE:   [CR]
```

## Read Digital Inputs

```
I      Read 7 bits.
In     Read bit number n.
```

These commands read the digital input bits from the Pod. All byte responses are sent most-significant nybble first.

Examples:

Read ALL 7 bits.

```
SEND:      I
RECEIVE:   FF[CR]
```

Read only bit 2.

```
SEND:      I2
RECEIVE:   1[CR]
```

## Write Digital Outputs

```
Oxx      Write to all 7 digital output bits.
Ox±     Set bit x hi or low.
```

These commands write outputs to digital bits. Any attempt to write to a bit configured as an input will fail. Writing to a byte or word wherein some bits are input and some are output will cause the output latches to change to the new value, but the bits that are inputs will not output the value until/unless they are placed in output mode.

Single bit commands will return an error (4) if an attempt is made to write to a bit configured as an input.

Writing a “one” (+) to a bit asserts the pull-down for that bit. Writing a “zero” (-) de-asserts the pull-down. Therefore, if the factory default +5V pull-up is installed, writing a one will cause zero volts to be at the connector, and writing a zero will cause +5 volts to be asserted.

Examples:

Write a one to bit 6 (set output to zero volts, assert the pull-down).

```
SEND:      O6+
RECEIVE:   [CR]
```

Write a zero to bit 2 (set output to +5V or user pull-up).

SEND: O2-

or

SEND: O02-  
RECEIVE: [CR]

Write zeros to bits 0-6.

SEND: O00  
RECEIVE: [CR]

Write zeros to every odd bit.

SEND: OAA  
RECEIVE: [CR]

### Read Firmware Revision Number

V Read the firmware revision number.

This command is used to read the version of firmware installed in the Pod. It returns "X.XX[CR]".

Example:

Read the RA1216 version number.

SEND: V  
RECEIVE: 0.50[CR]

NOTE: The "H" command returns the version number along with other information. See "Hello Message", below.

### Resend Last Response

n Resend last response

This command will cause the Pod to return the same thing it just sent. This command works for all responses less than 255 characters in length. Normally this command is used if the host detected a parity or other line fault while receiving data and needs the data to be sent a second time.

The "n" command may be repeated.

Example:

Assuming the last command was "I", ask Pod to resend last response.

SEND: n  
RECEIVE: FFFFFFFF[CR] ;or whatever the data was

### Hello Message

H\* Hello message

Any string of characters starting with "H" will be interpreted as this command. ("H[CR]")

alone is also acceptable.) The return from this command takes the form (without the quotes):

"=Pod aa, RA1216 Rev rr Firmware Ver:x.xx ACCES I/O Products, Inc."

aa is the Pod address

rr is the hardware revision, such as "B1"

x.xx is the software revision, such as "0.50"

Example:

Read the greeting message.

SEND: Hello?

RECEIVE: = Pod 00, RA1216 Rev B1 Firmware Ver:1.00 ACCES I/O Products, Inc.[CR]

**Configure Baud Rate** (When initially shipped, the baud rate is set at 9600.)

BAUD=nnn Program the Pod with a new baud rate

This command sets the Pod to communicate at a new baud rate. The parameter passed, nnn, is slightly unusual. Each n is the same digit from the following table:

CODE	BAUD RATE
0	1200
1	2400
2	4800
3	9600
4	14400
5	19200
6	28800
7	57600

Therefore, valid values for the command's "nnn" are 000, 111, 222, 333, 444, 555, 666, or 777. The pod returns a message indicating it will comply. The message is sent in the old baud rate, not the new one. Once the message is transmitted, the Pod changes to the new baud rate. The new baud rate is stored in EEPROM and will be used even after power-reset, until the next "BAUD=nnn" command is issued.

Examples:

Set the Pod to 19200 baud.

SEND: BAUD=555

RECEIVE: =:Baud:05[CR]

Set the Pod to 9600 baud.

SEND: BAUD=333

RECEIVE: =:Baud:03[CR]

## Configure Pod Address

**POD=xx** Program the currently selected Pod to respond at address xx.

This command changes the Pod's address to xx. If the new address is 00, the Pod will be placed into non-addressed mode. If the new address is not 00, the Pod will not respond to further communications until a valid address command is issued. Hex numbers 00-FF are considered valid addresses. The RS-485 specification allows only 32 drops on the line, so some addresses may be unused.

The new Pod address is saved in EEPROM and will be used even after power-down until the next "Pod=xx" command is issued. Note that, if the new address is not 00 (i.e., the Pod is configured to be in addressed mode), it is necessary to issue an address command to the Pod at the new address before it will respond.

The Pod returns a message containing the Pod number as confirmation.

Examples:

Set the Pod address to 01.

```
SEND:          A=01
RECEIVE:       =:Pod#01[CR]
```

Set the Pod address to F3.

```
SEND:          A=F3
RECEIVE:       =:Pod#F3[CR]
```

Take the Pod out of addressed mode.

```
SEND:          A=00
RECEIVE:       =:Pod#00[CR]
```

## Address Select

**!xx** Selects the Pod addressed 'xx'

**NOTE:** When using more than one Pod in a system, each Pod is configured with a unique address. This command must be issued prior to any other commands to that particular Pod. This command needs to be issued only once prior to executing any other commands. Once the address select command has been issued, that Pod will respond to all other commands until a new address select command is issued.

## Enter a New Program

**PROGRAM=** This command initiates transfer of a new program to the RA1216.

This command should be used carefully. If you accidentally issue a "PROGRAM=" command, ESC (ASCII 27) will restart the Pod as if power had been reset. This feature is designed to allow ACCES to provide field-upgrades to the RA1216 firmware, and, for advanced users, the opportunity to customize the firmware in the Pod. Documentation on using this command is provided with the upgrade diskette, or is available separately for a small fee.

| (Vertical Bar)= Same as PROGRAM= except it is used only after a reset has occurred. Allows access to Pod to download new firmware.

## ERROR CODES

The following error codes can be returned from the Pod:

- 1: Invalid channel number (too large, or not a number. All channel numbers must be between 00 and 0F, in hex).
- 3: Improper Syntax. (Not enough parameters is the usual culprit).
- 4: Channel number is invalid for this task (For example if you try to output to a bit that is set as an input bit, that will cause this error).
- 9: Parity error. (This occurs when some part of the received data contains a parity or framing error).

Additionally, several full-text error codes are returned. All begin with "Error," and are useful when using a terminal to program the Pod.

Error, Unrecognized Command: {command received}[CR]

This occurs if the command is not recognized.

Error, Command not fully recognized: {Command received}[CR]

This occurs if the first letter of the command is valid, but the remaining letters are not.

Error, Address command must be CR terminated[CR]

This occurs if the address command (!xx[CR]) has extra characters between the Pod number and the [CR].

# SPECIFICATIONS

## SERIAL COMMUNICATIONS INTERFACE

Serial Port: Opto-isolated Matlabs type LTC485 Transmitter/Receiver. Compatible with RS-485 specification. Up to 32 drivers and receivers allowed on the line. I/O bus programmable from 00 to FF hex (0 to 255 decimal). Whatever address is assigned is stored in EEPROM and used as default at next Power-On.

Asynchronous Data Format: 7 data bits, even parity, one stop bit

Input Common Mode Voltage: 300V minimum (opto-isolated). If opto-isolators are bypassed: -7V to +12V.

Receiver Input Sensitivity:  $\pm 200$  mV, differential input.

Receiver Input Impedance: 12K minimum

Transmitter Output Drive: 60 mA, 100 mA short circuit current capability.

Serial Data Rates: Programmable for 1200, 2400, 4800, 9600, 14400, 19200, 28800, and 57600 baud. Crystal oscillator provided.

## ANALOG INPUTS

Channels: Sixteen single-ended with common ground or eight differential.

Voltage Ranges: Software controllable.

Unipolar: 0-5V, 0-2.5V, 0-1V, 0-500 mV, 0-250 mV, 0-125 mV, 0-50 mV, and 0-25 mV.

Bipolar:  $\pm 2.5$ V,  $\pm 1.25$ V,  $\pm 500$  mV,  $\pm 250$  mV,  $\pm 62.5$  mV,  $\pm 25$  mV, and  $\pm 12.5$  mV.

Current Range: 4-20 mA across 200 $\Omega$  (0.8 - 4V).

Input Protection: 200 VDC.

Input Filter: Low-pass (10K $\Omega$ /0.1 $\mu$ F).

Input Impedance: 10 Megohms.

Resolution: 12 binary bits.

Accuracy: 0.025%  $\pm 1$  LSB.

Gain Drift:  $\pm 5$  PPM/ $^{\circ}$ C

Coding: True binary for unipolar inputs and offset binary for bipolar inputs.

Throughput: 50,000 conversions per second in foreground mode, 6.67K conversions per second in background mode.

A/D Type: Successive Approximation

Trigger Source: Software command or on-board programmable timer.

## ANALOG OUTPUTS

Channels: Three independent.

Type: 12-bit, double-buffered

Non-Linearity:  $\pm 0.9$  LSB maximum.

Monotonicity:  $\pm 1/2$  bit.

Output Range: 0-5V, 0-10V.

Current Output: 4-20 mA (User supplied excitation of 5.5V-30V).

Output Drive: 5 mA.

Output Resistance: 0.5  $\Omega$ .

Settling Time: 15  $\mu$ sec to  $\pm 1/2$  LSB.



## DIGITAL I/O

Seven bits configured as input or output

Digital Inputs	Logic High: +2.0V to +50V at 20 $\mu$ A max. (5mA max at 50V in) Protected to 200 VDC
	Logic Low: -30V to +0.8V at -0.4 mA max. Protected to -140 VDC.
Digital Outputs	Logic-Low Sink Current: 350 mA maximum. (See note below.) Inductive kick suppression diode included in the each circuit.

### NOTE

Maximum allowable current per output bit is 350 mA. When all seven bits are used, there is a maximum total current of 650 mA.

High-Level Output Voltage: Open Collector, compliance with up to 50VDC user-supplied voltage. If no user supplied voltage exists, outputs pulled up to +5VDC via 10 k $\Omega$  resistors.

## INTERRUPT INPUT (For use with development kit)

Input Low: -0.3V to +0.8V.  
Input Low Current at 0.45V: -55  $\mu$ A.  
Input High: 2.0V to 5.0V

## POWER REQUIRED

Power can be applied from the computer's +12VDC power supply for the opto-isolated section via the serial communication cable and from a local power supply for the rest of the unit. If you do not wish to use power from the computer, a central power supply may be used for the opto-isolated section.

Local Power: 12 to 16 VDC @ 500 mA. (See box that follows.)

Opto-Isolated Section: 7.5 to 24 VDC @ 50 mA. (Note: Due to the small amount of current required, voltage drop in long cables is not significant.)

If the local power supply has an output voltage greater than 16VDC, you can install a Zener diode in series with the supply voltage. The voltage rating of the Zener diode ( $V_z$ ) should be equal to  $V_1 - 16$  where  $V_1$  is the power supply voltage. The power rating of the Zener diode should be  $\geq V_z \times 0.35$  (watts). Thus, for example, a 24VDC power supply would require using an 8.2V Zener diode with a power rating of  $8.2 \times 0.35 \approx 3$  watt.

# WARRANTY

Prior to shipment, ACCES products are thoroughly inspected and tested to applicable specifications. However, should equipment failure occur, ACCES assures its customers that prompt service and support will be available. All equipment originally manufactured by ACCES which is found to be defective will be repaired or replaced subject to the following considerations.

## TERMS AND CONDITIONS

If a unit is suspected of failure, contact ACCES' Customer Service department. Be prepared to give the model number, serial number, and a description of the failure symptom(s). We may suggest some simple tests to confirm the failure. We will assign a Return Material Authorization (RMA) number which must appear on the outer label of the return package. All units/components should be properly packed for handling and returned, freight prepaid, to the ACCES designated Service Center, and will be returned to the customer's/user's site freight prepaid and invoiced.

## COVERAGE

**First Three Years:** Returned unit/part will be repaired and/or replaced at ACCES option with no charge for labor or parts not excluded by warranty. Warranty commences with equipment shipment.

**Following Years:** Throughout your equipment's lifetime, ACCES stands ready to provide on-site or in-plant service at reasonable rates similar to those of other manufacturers in the industry.

## EQUIPMENT NOT MANUFACTURED BY ACCES

Equipment provided but not manufactured by ACCES is warranted and will be repaired according to the terms and conditions of the respective equipment manufacturer's warranty.

## GENERAL

Under this Warranty, liability of ACCES is limited to replacing, repairing or issuing credit (at ACCES discretion) for any products which are proved to be defective during the warranty period. In no case is ACCES liable for consequential or special damage arising from use or misuse of our product. The customer is responsible for all charges caused by modifications or additions to ACCES equipment not approved in writing by ACCES or, if in ACCES opinion the equipment has been subjected to abnormal use. "Abnormal use" for purposes of this warranty is defined as any use to which the equipment is exposed other than that use specified or intended as evidenced by purchase or sales representation. Other than the above, no other warranty, expressed or implied, shall apply to any and all such equipment sold or furnished by ACCES.

## APPENDIX A

### APPLICATION CONSIDERATIONS

#### INTRODUCTION

Working with RS-422 and RS-485 devices is not much different from working with standard RS-232 serial devices and these two standards overcome deficiencies in the RS-232 standard. First, the cable length between two RS-232 devices must be short; less than 50 feet at 9600 baud. Second, many RS-232 errors are the result of noise induced on the cables. The RS-422 standard permits cable lengths up to 4000 feet and, because it operates in the differential mode, it is more immune to induced noise.

Connections between two RS-422 devices (with CTS ignored) should be as follows:

Device #1		Device #2	
Signal	Pin No.	Signal	Pin No.
Gnd	5	Gnd	5
TX <sup>+</sup>	2	RX <sup>+</sup>	9
TX <sup>-</sup>	3	RX <sup>-</sup>	1
RX <sup>+</sup>	9	TX <sup>+</sup>	2
RX <sup>-</sup>	1	TX <sup>-</sup>	3

A third deficiency of RS-232 is that more than two devices cannot share the same cable. This is also true for RS-422 *but RS-485 offers all the benefits of RS-422 plus allows up to 32 devices to share the same twisted pairs*. An exception to the foregoing is that multiple RS-422 devices can share a single cable if only one will talk and the others will all receive.

#### BALANCED DIFFERENTIAL SIGNALS

The reason that RS-422 and RS-485 devices can drive longer lines with more noise immunity than RS-232 devices is that a balanced differential drive method is used. In a balanced differential system, the voltage produced by the driver appears across a pair of wires. A balanced line driver will produce a differential voltage from  $\pm 2$  to  $\pm 6$  volts across its output terminals. A balanced line driver can also have an input “enable” signal that connects the driver to its output terminals. If the “enable signal is OFF, the driver is disconnected from the transmission line. This disconnected or disabled condition is usually referred to as the “tristate” condition and represents a high impedance. RS-485 drivers must have this control capability. RS-422 drivers may have this control but it is not always required.

A balanced differential line receiver senses the voltage state of the transmission line across

the two signal input lines. If the differential input voltage is greater than +200 mV, the receiver will provide a specific logic state on its output. If the differential voltage input is less than -200 mV, the receiver will provide the opposite logic state on its output. A maximum operating voltage range is from +6V to -6V allows for voltage attenuation that can occur on long transmission cables.

A maximum common mode voltage rating of  $\pm 7V$  provides good noise immunity from voltages induced on the twisted pair lines. The signal ground line connection is necessary in order to keep the common mode voltage within that range. The circuit may operate without the ground connection but may not be reliable.

**RS-422 SPECIFICATION SUMMARY**

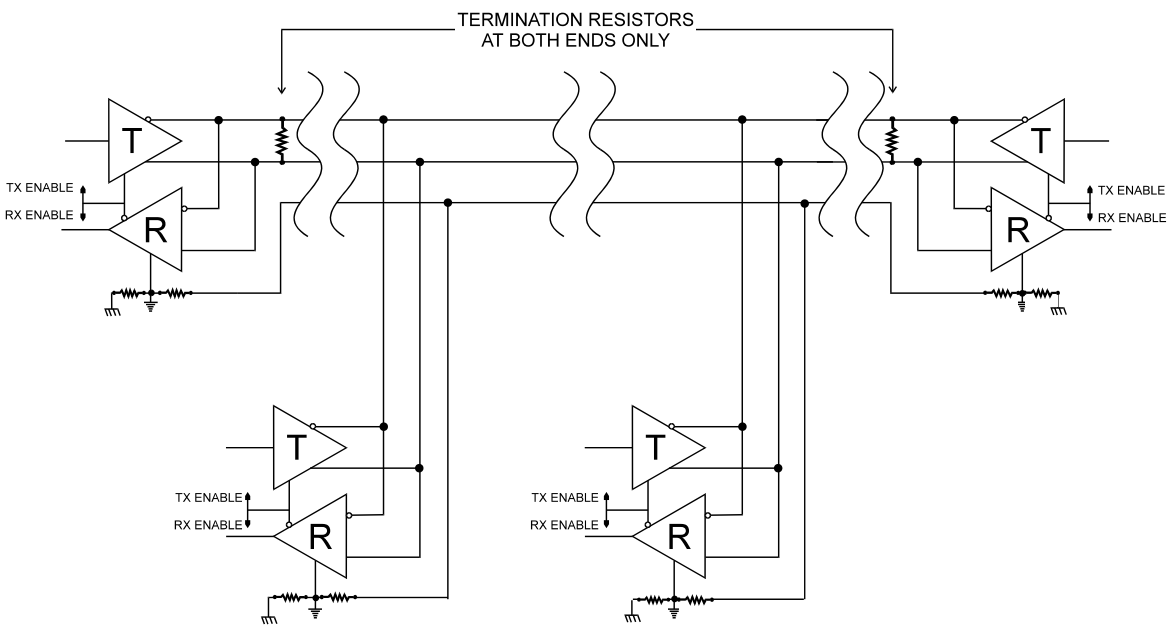
Parameter	Conditions	Min.	Max.
Driver Output Voltage (unloaded)		4V -4V	6V -6V
Driver Output Voltage (loaded)	LD and LDGND jumpers in	2V -2V	
Driver Output Resistance			50
Driver Output Short-Circuit Current			$\pm 150$ mA
Driver Output Rise Time			10% unit interval
Receiver Sensitivity			$\pm 200$ mV
Receiver Common Mode Voltage Range			$\pm 7V$
Receiver Input Resistance			4K

To prevent signal reflections in the cable and to improve noise rejection in both the RS-422 and RS-485 mode, the receiver end of the cable should be terminated with a resistance equal to the characteristic impedance of the cable. (An exception to this is the case where the line is driven by an RS-422 driver that is never “tri-stated” or disconnected from the line. In this case, the driver provides a low internal impedance that terminates the line at that end.)

## RS-485 DATA TRANSMISSION

The RS-485 Standard allows a balanced transmission line to be shared in a party-line mode. As many as 32 driver/receiver pairs can share a two-wire party line network. Many characteristics of the drivers and receivers are the same as in the RS-422 Standard. One difference is that the common mode voltage limit is extended and is +12V to -7V. Since any driver can be disconnected (or tri-stated) from the line, it must withstand this common mode voltage range while in the tristate condition.

The following illustration shows a typical multidrop or party line network. Note that the transmission line is terminated on both ends of the line but not at drop points in the middle of the line.



Typical **RS-485** Two-Wire Multidrop Network