invensys
Eurotherm

KD485
User Guide

KD485 Multi-mode interface converter
Firmware versions1.03 and later

HA030692/4
March 2010

CE

This product is compliant with the EC EMC Directive when installation procedures are followed.

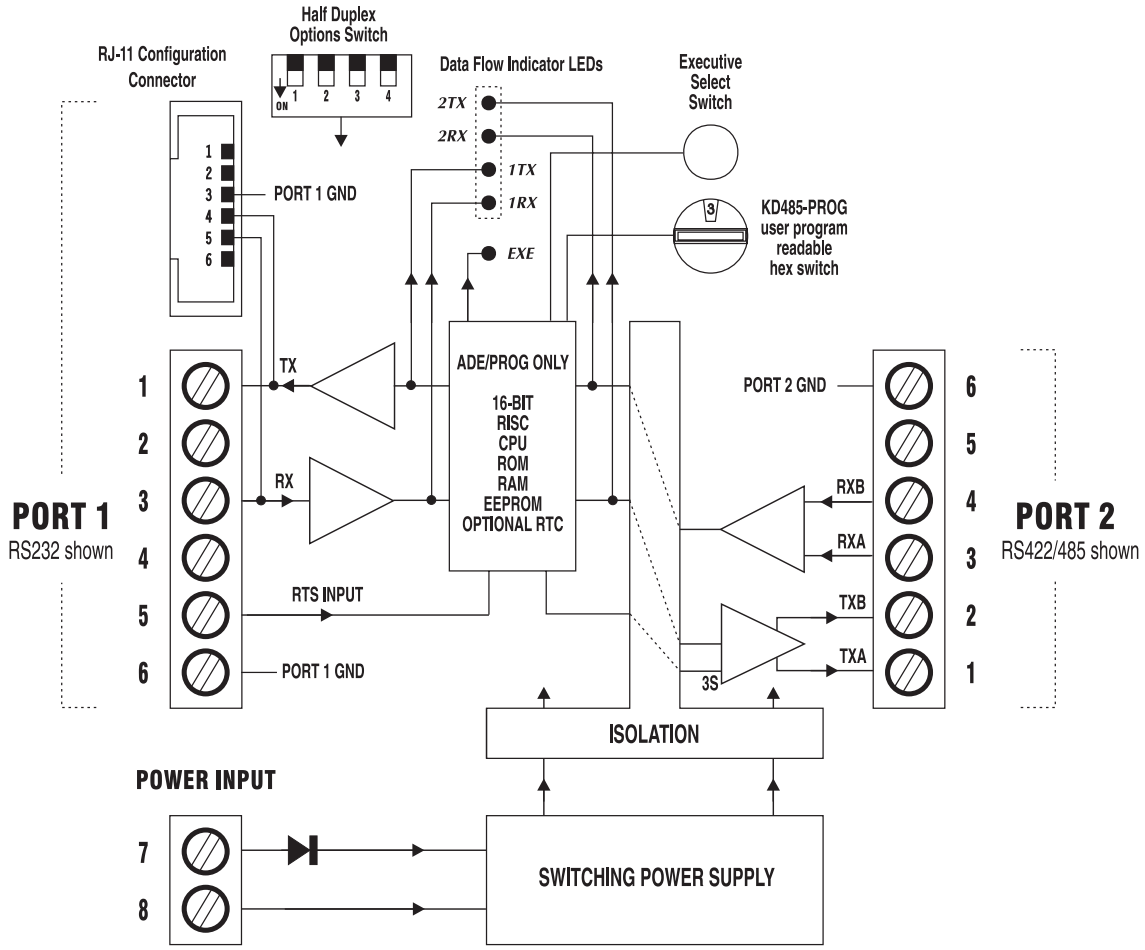
Product versions
This manual covers the following versions:
KD485 firmware v1.03 dated 13/1/95 or later

HA030692/4 (CN26268).

Table of Contents

KD485 Overview

The KD485 is a multi-mode interface converter, with two serial asynchronous ports which are available factory-fitted in any combination of RS232, RS422/485 or 20mA loop:



The default port configuration (shown above) is **RS232 on Port 1** and **RS422/485 on Port 2**. If the port types are not specified when ordering this is what is supplied. All other configurations are specified with a suffix, e.g.

KD485-ADE-422-20MA

specifies Port 1 as RS422/485 and Port 2 as 20mA loop.

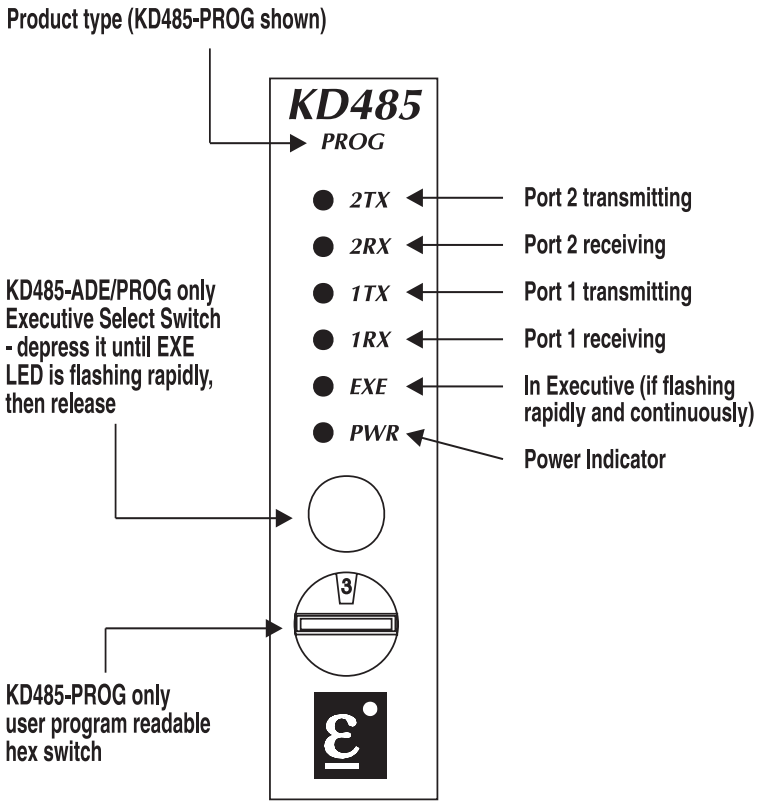
Port 1, power input, and Port 2 are on removable screw terminals. For additional convenience during configuration of KD485-ADE and KD485-PROG, Port 1 is electrically duplicated on an RJ-11 connector. The RJ-11 connector can also be useful on the KD485-STD as it provides a quicker connection to Port 1.

The KD485 is three-way isolated. This means that the two ports and the power supply input are all isolated from each other. This offers great installation flexibility while avoiding ground loops.

The KD485 is powered by a supply in the range +7V to +35V, 1-2 watts (typical). This is a high efficiency switching power supply which draws nearly constant power over the input voltage range and ensures very low power dissipation in the KD485.

☞ The constant power property of the KD485 may require a larger power supply to be used. Please see the **Installation** chapter for details.

Front Panel Description

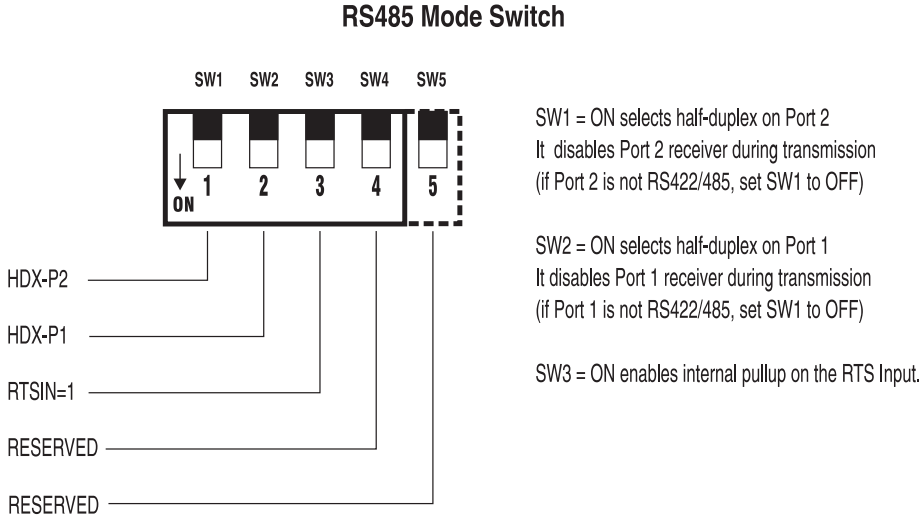


All KD485 versions feature four LEDs which indicate data flow on the two ports.

The KD485-ADE has an additional push-button switch for selecting the Executive (configuration) mode, and an “EXE” LED to indicate this mode, and other conditions.

The KD485-PROG has an additional 16-position rotary hex switch which is user program readable.

Half Duplex Options Switch



This switch is fitted on all KD485 versions; however, its function is presently limited to versions which have at least one RS422/485 port.

SW1 Controls if Port 2 receiver is enabled during transmission out of Port 2. This switch is relevant only if Port 2 is RS422/485 and is externally wired for 2-wire RS485 operation. Then, reception of own transmission is usually not desired (i.e. half-duplex operation is required) and therefore SW1=ON. There are special cases where one may want to receive everything in which case SW1=OFF. In general, if Port 2 is not RS485, set SW1 to OFF.
SW1 replaces the "HDX" jumper in the previous (grey box) KD485 model.

SW2 This is as SW1 but operates on Port 1. Since only the KD485-ADE/-PROG can control the Port 1 RS422/485 driver, this switch is relevant only if the KD485-ADE/-PROG is used as a converter from one RS485 bus to another RS485 bus. The most common such application is attaching 2-wire devices to an existing 4-wire bus, in which case reception of own transmission is not desired on either port, and SW1=SW2=ON.

SW3 This merely enables an internal pullup on the RTS Input. The RTS Input is normally used only if Port 2 is RS422/485, to control its driver. Typically, with the KD485-STD converting from RS232 to RS422 when a permanently enabled Port 2 driver is desired, then SW3=ON.
The RTS Input is also user program readable in the KD485-PROG.
SW3 replaces the "RTS=1" jumper in the previous KD485 model.

SW4,5 Currently unused. Separate documentation will be provided if used.

Standard KD485 Versions

The KD485 is available in three standard versions:

1. KD485-STD Isolated Converter; standard model

This entry level product has no CPU. It is an interface converter/isolator only and is baud rate and character format independent.

In the default RS232-RS422/485 version, it enables RS232 devices to communicate with RS422 or 2/4-wire RS485 devices. When tri-state operation is required on Port 2, the RS232 host must provide RTS control.

The KD485-STD can also be used as a Master on a 4-wire RS485 system.

2. KD485-ADE Isolated Converter; with Auto Driver Enable

This inserts a CPU (with two UARTs) into the data path. The primary purpose of this version is for RS232 to RS485 conversion; it can however be used for character format / baud rate conversion also.

The two ports are individually configurable 30-115200 baud. Port 1 can be placed into an "Executive" configuration mode and the KD485-ADE can be configured using any ASCII terminal ranging from small hand-held devices (e.g. a Psion Organiser XP) to IBM PCs. A Windows-based pulldown-menu program can also be used. The configuration is stored in an EEPROM. The configuration software is included on a diskette.

The KD485-ADE can be user-configured to operate in one of the following modes:

0 Applicable to all port types. **Character format conversion** (baud rate, bits/word etc) only. The two ports can be individually configured with different settings. No auto driver enable for RS485.

1 Applicable only if Port 2 is RS422/485. Operation as for Mode 0, plus **auto driver enable**. This controls the Port 2 RS485 driver according to the presence of data arriving at Port 1, and eliminates the need for the Port 1 system to provide driver control. This mode can also be configured to work the other way, i.e. to control the Port 1 RS485 driver according to data arriving at Port 2, to form a bidirectional RS485-RS485 converter with auto driver enable at both ends.

2 Applicable only if Port 2 is RS422/485. This is an RS485 **addressable adapter**. This supports applications where multiple non-RS485 (i.e. RS232 or 20mA Loop) devices need to be multi-dropped on a 2-wire or 4-wire RS485 bus, but the devices themselves do not support address recognition, and/or do not have tri-state drivers. Devices which emit data only in response to a poll and devices which emit data continuously are both supported.

3. KD485-PROG Isolated Converter; user programmable in ANSI C

This does everything that KD485-ADE does and adds a large EEPROM, plus a means of uploading user-written programs into the EEPROM. User programs can be written in ANSI C, assembler, or other languages, and are uploaded (in Intel Hex) to the KD485-PROG using supplied software. The KD485-PROG is a very powerful high-performance programmable datacomms product suitable for highly complex projects.

A high quality ANSI C compiler is available separately and its multi-stream I/O and other features are supported with dedicated code in the KD485-PROG on-chip ROM. The result is a well-integrated system which greatly simplifies the creation of datacomms programs.

A MODBUS RTU SLAVE library is available which facilitates the rapid development of a MODBUS front end for a non-MODBUS instrument.

Custom Versions

Other KD485 versions are available. These include any combination of RS232, 422, 485 and 20mA loop interfaces, and custom software. Please contact Factory with your requirements.

Completely customised products, based on the KD485 or the 4-port PPC Programmable Protocol Converter, can also be supplied. Custom labelling and different physical layout (e.g. Eurocard) can be supplied.

Specification

Ports:	Two asynchronous ports, TX & RX signals only. XON/XOFF selectable.
Port parameters:	KD485-STD: 0 to 115200 baud, all character formats. KD485-ADE/PROG: 30-115200 baud, none/even/odd parity, 7/8 data bits, 1/2 stop bits. 20mA Loop ports: 30-19200 baud.
Interface combinations:	Standard product: port 1 is RS232; port 2 is RS422/485. Any combination of RS232, RS422/485 or 20mA loop can be supplied.
RS232 interface:	Receiver threshold +1.5V typ. Hysteresis 500mV typ. Receiver input impedance 5kΩ typ. Transmitter output swing $\pm$8V typ into 3k load.
RS422/485 interface:	Receiver threshold 200mV typ (differential). Hysteresis 70mV typ. Receiver input impedance 12kΩ min. Transmitter output swing 0 to +5V (no DC load); +2 to +3V (120Ω ohm load).
20mA loop interface:	Input: LED, nominal drop 2V Output: open collector transistor, Vce(sat) < 2V 20mA current source: accuracy $\pm$20%; no-load voltage approximately equal to KD485 supply voltage.
I/O delay:	STD: <100μs. ADE/PROG: approx 2ms (@9600 baud, Mode 1).
Power supply:	+7V to +35V DC. +12V DC recommended. +7V to +26V DC if two 20mA Loop ports are fitted. Input power is approximately constant at 1-2 watts (startup current 300-600mA) depending on model. At startup, the supply voltage must reach 7V within 1 sec otherwise the power supply will not function.
Isolation:	64V PK, tested at >1000V AC RMS, 1 second.
Environmental:	Operating temperature 0 to +50C. Storage temperature -25C to +70C. Relative humidity (operating and storage) 0 to 90%, non-condensing.
Ventilation:	Rail-mounted KD485 must have a 50mm gap above and below.
EMC compliance:	Emissions EN50081-2 (94), immunity EN50082-2 (95).
Dimensions:	29mm (W) x 114mm (H) x 97mm (L) approx. in rail-mounted position, including screw terminals.

Changes from previous (1994-1998) KD485 version

A number of improvements have been made:.

- New enclosure with removable screw terminals.
- Three-way isolation; previous model shared a ground between the power supply and Port 1.
- Wide input range switching power supply. Much reduced heat dissipation, especially at 24V input.
- Front-accessible push button switch and LED for KD485-ADE/-PROG configuration; on previous model these were side accessible.
- Front panel visible data flow indicator LEDs.
- Externally accessible DIP switch replaces the internal jumpers.
- Port 1 can be switch configured to disable its receiver during transmission (for RS485). On previous model the Port 1 receiver was always enabled.
- Port 1 is duplicated on an RJ-11 connector. This makes KD485-ADE/-PROG configuration more convenient and also brings out the KD485-PROG TTL-level DEBUG output port.
- 16-position user program readable switch on KD485-PROG.
- 20mA Loop option, with active output (20mA current source) available on either or both ports.
- Factory-fittable plug-in I/O cards enable unusual versions, e.g. KD485-ADE-20MA-422 (for conversion between 20mA loop and 2-wire RS485) to be available on a short lead time.
- 14V (bidirectional) varistor protection on RS232 ports; previous model was varistor protected on RS422/485 ports (Port 2) only.
- +5V low power output on Port 2 screw terminal; can be used for powering external RS485 pullup resistors.
- On the KD485-STD, the polarity of the RTS Input can be reversed (as a factory option) to support users whose RTS source is HIGH to receive and LOW to transmit.
- The KD485-PROG has a Real Time Clock option.
- KD485-PROG has a factory-option optocoupler (open-collector) user-addressable output, available if Port 1 is not RS422/485.
- No need for installer to open the KD485 enclosure

☞ The following features should be examined carefully when replacing the previous model with the new model:

- Due to the constant power characteristics of the new KD485 power supply, together with the low startup voltage (typically 4V) the unit draws a current which can reach 300-600mA (depending on model and port types) momentarily as the supply voltage rises through 4V. This may necessitate the use of a larger power supply than the previous KD485 required.
- The KD485-STD no longer has a default pullup on the RTS Input. If you are not driving RTS externally, you must set SW3=ON to obtain a HIGH level on the RTS Input.
- Only 35mm DIN rails are supported. Asymmetric rails are not.

KD485-ADE/PROG Mode 2 Detail Description

The KD485 monitors the RS485 bus for an (optional) **lead-in byte**, followed by a matching **address byte**. Upon address match, any subsequent data received from the RS485 bus is passed to the attached RS232/422 device on port 1; this continues until a user-specified **timeout** has passed since the last data byte transferred. These three parameters are configured with the **AD** command (see the **Executive** chapter), or with KDCFG under "Mode 1 Configuration".

When all RS485 data has been passed on to the attached device, and following the expiration of the timeout, the KD485 assumes that there is some data in the attached device's (port 1) input queue. In other words, it assumes that the attached device *has already started* to generate its response. The timeout must therefore be long enough to ensure this!

At the expiration of the timeout, the KD485 transfers any data from the attached device's queue back to the RS485 bus. This continues until there is no more data to transfer. If the timeout is too short, there will be no data *to start with*, and nothing will be transferred.

Mode 2 is also designed to be suitable for multi-dropping RS232 devices which generate data *continuously*, i.e. without being prompted. The KD485 simply returns everything which the device has generated since the last time the KD485 was polled. The size of the KD485 RX queue is usually sufficient, except when the KD485 is polled very infrequently.

☞ When Mode 2 is used with a 2-wire RS485 bus, ensure that SW1=ON. This inhibits the reception of own data back from Port 1.

Mode 2 supports two special features: an address of **255** is treated as a "broadcast" address and will always match. This is useful if one needs to send a message to all devices simultaneously. Of course, they must not respond to this message, otherwise bus contention will occur!

An address of **254** instructs the KD485 to return the number of bytes in the KD485's Port 1 RX queue, as a 7-character ASCII string in the form ":ddddd<CR>". The initial ":" and the final <CR> characters are useful delimiters. All five digits are always returned, with leading zeroes if necessary. This can be useful if you have a device which emits unsolicited data, and you need to periodically check how much data it has emitted. If there is more than one Mode 2 KD485 on the RS485 bus, *you need to use the lead-in byte* as an address byte otherwise bus contention will occur when all the KD485s respond!

Choosing Address and Lead-In bytes: If these values accidentally appear in the data on the RS485 bus (data flowing in *either* direction) this can cause false address recognition. This is a standard problem with RS485 and, apart from specialised solutions, it cannot be avoided. There are however two things you can do:

If possible, choose the message content and/or the addressing so the address byte (or the sequential combination of lead-in and address byte, if a lead-in byte is used) cannot appear within it.

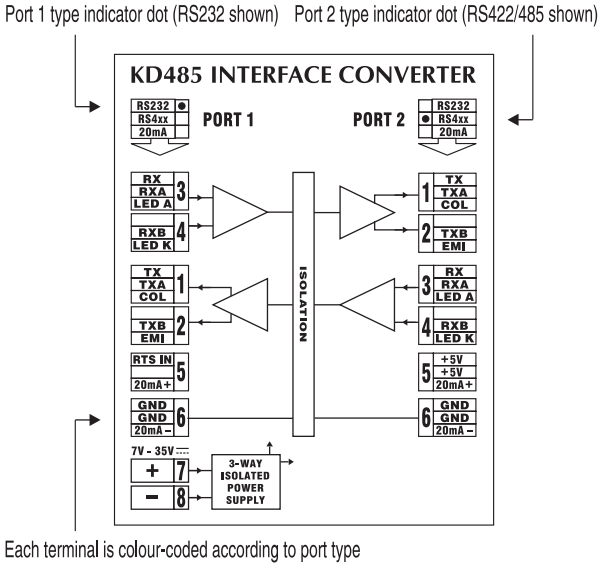
Use **4-wire** RS485 in preference to 2-wire RS485, for the multidrop bus. With a 4-wire system, the KD485 slaves do not see each others' responses and this eliminates the possibility of false addressing caused by the data being *returned*. However, you still need to ensure that the (optional) data *transmitted by the master* after the addressing byte(s) cannot cause false addressing of other slaves.

Installation

For detail information on the various port types (RS232, RS422/485, 20mA loop) and DIP switch configuration please also refer to the **Ports and Connections** section.

Product and Interface Type identification

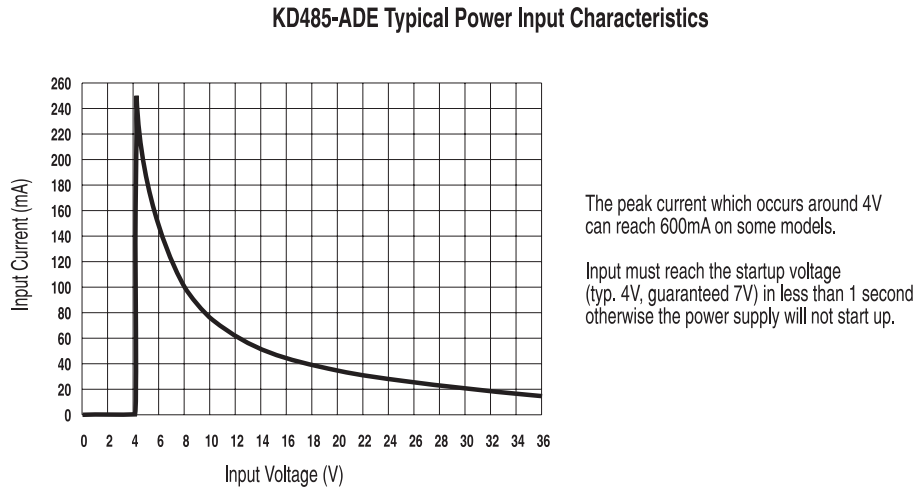
The KD485 type, i.e. -STD, -ADE, -PROG is marked on the front label. The type of each of the two ports is marked on the side label and these are colour-coded to show which terminals are applicable to that port type:



DC Power Requirements

The KD485 requires a regulated or unregulated DC power supply whose voltage must at all times be in the range +7V to +35V.

For efficiency, the KD485 uses a switching power supply. Therefore, **the current consumption varies with the supply voltage**, such that the power (in watts) remains fairly constant over the supply voltage range. The following graph shows the relationship for a typical KD485-ADE:



It can be seen that to ensure reliable startup the power supply must be capable of delivering up to 600mA. If you have a choice, a +9V or +12V regulated supply is recommended. There is no advantage in using

higher voltages; the power supply needs to deliver (during startup) up to 600mA so if a higher voltage power supply is used, it needs to be a larger-wattage unit.

KD485-STD Configuration

The only potential configuration is on the DIP switch. The most common configurations are:

KD485-STD (RS232 to RS422): SW1,SW2=OFF, SW3=ON
KD485-STD (RS232 to 2-wire RS485, RTS control provided by RS232 device): SW1=ON, SW2,SW3=OFF

With 2-wire RS485 (on Port 2) applications, remember that the RS232 host must control the RTS signal to control the KD485 tri-state driver. Port 1 must therefore be RS232.

If the KD485-STD is a Master on a 4-wire RS485 system, or is used in a simple RS422 point-to-point link, leave RTS unconnected and permanently enable the Port 2 RS485 driver with SW3=ON.

Finally, with the power disconnected, connect the Port 1 and Port 2 signals and the power supply.

KD485-ADE Configuration

In addition to the DIP switch settings, **this version must be electronically configured before it can be used,** via Port 1. This involves mainly the port parameters, but you must also specify which of the several built-in programs ("Modes") you wish to run. The configuration can be done in one of two ways:

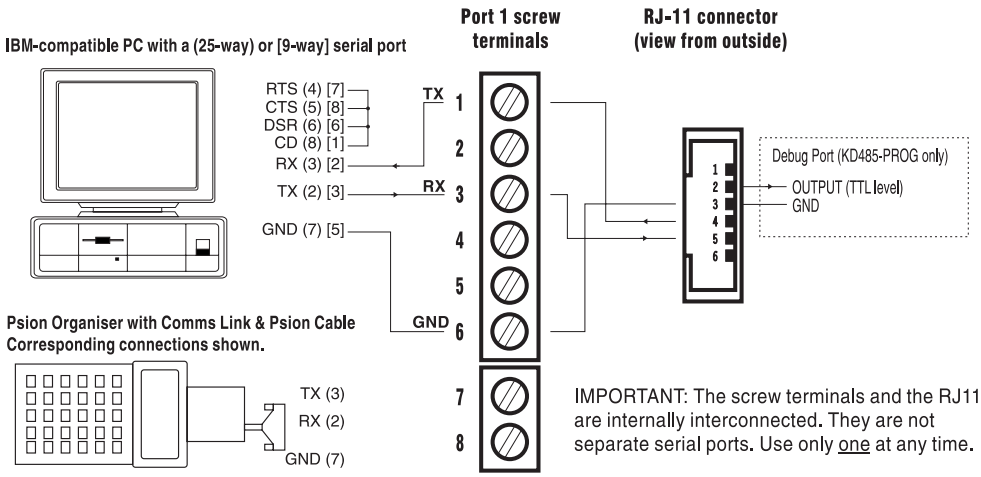
1 With a "dumb" terminal, set to 9600 baud, 8 bits/word, no parity, 1 stop bit. This can be a PC-based terminal emulator; a simple utility called TERM.EXE is provided on the diskette. This uses a simple command-line interface where the KD485 provides a ">" prompt. Refer to the **Executive** chapter for details.

2 With a Windows-based program, called KDCFG.EXE. This is provided on the diskette. This method offers access to some additional (although very rarely used) KD485-ADE features.

Configuration is done via Port 1 which must be set to the Executive Mode, indicated by the "EXE" LED flashing rapidly and continuously. To enter Executive Mode, press the front panel pushbutton switch until the "EXE" LED starts flashing, then release it. A few seconds later, the LED will resume continuous flashing and you can start the configuration software.

Port 1 can be accessed either via the Port 1 screw terminals (numbered 1-6 on the 8-way terminal block), or via the RJ-11 connector. Do not connect to both simultaneously.

Configuration Connections for Port 1 = RS232



The supplied KDCFG.EXE and TERM.EXE programs require only the TX,RX,GND connections. The other connections shows at the PC end are sometimes needed with other terminal emulation programs.

The Psion Organiser pin numbers are those at the end of the Psion cable which comes as a part of the Psion Comms Link accessory. To use the Organiser as a dumb terminal, select COMMS, then select TERM. Configure to 9600, 8 bits/word, no parity, 1 stop bit.

Next, set up the DIP switches as applicable. The most common configuration is:

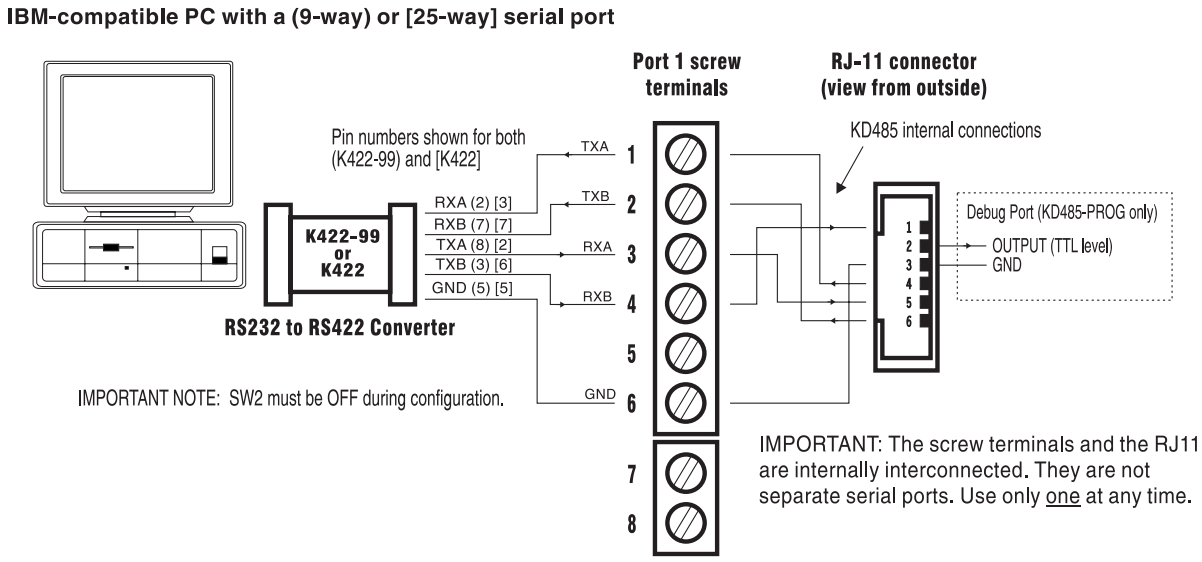
KD485-ADE (RS232 to 2-wire RS485): SW1=ON, SW2=OFF, SW3=ON)

Finally, with the power disconnected, connect the Port 1 and Port 2 signals and the power supply.

If using the Mode 0 program and Port 2 is RS422/485, RTS input must be externally driven, or be tied to a RS232 HIGH level with SW3=ON.

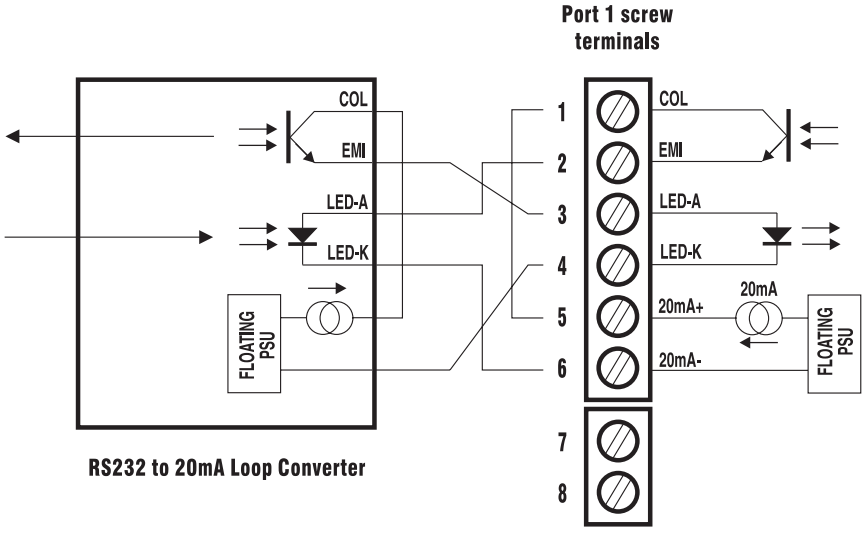
If Port 1 is RS422/485 then a suitable RS232-RS422 interface converter must be used:

Configuration Connections for Port 1 = RS422/485



If Port 1 is 20mA Loop then a RS232 to 20mA converter must be used:

Configuration Connections for Port 1 = 20mA Loop

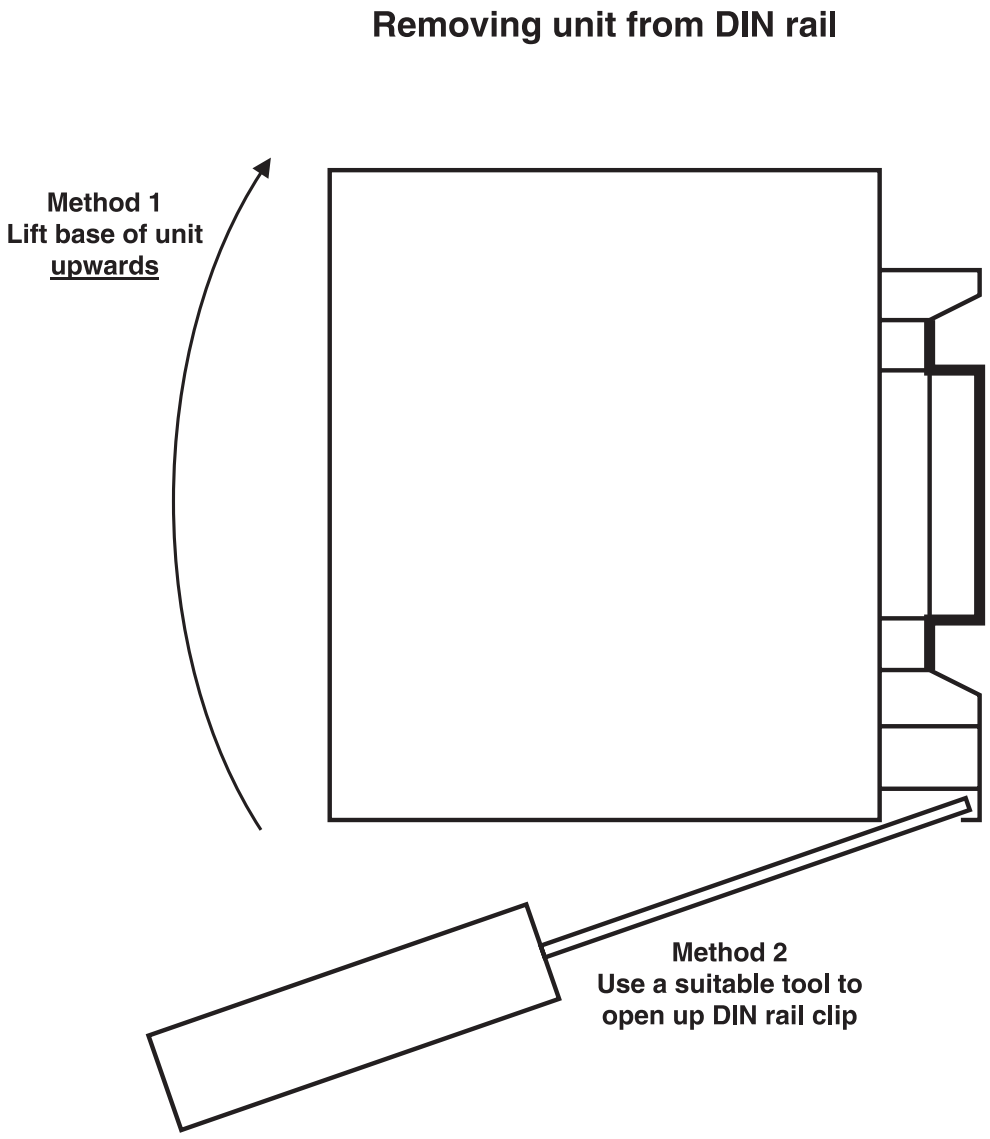


The RJ-11 connector cannot be used for the 20mA configuration since, due to insufficient pins, not all of the 20mA signals come out on it.

Fitting KD485 on DIN Rail

To fit the KD485 to the rail, simply clip it on.

To remove the KD485 from the rail, lift the front end **upwards**. However, this may not be successful, particularly with the taller and thicker 35mm rails, in which case use a suitable tool (e.g. a flat screwdriver) to gently (very little force is required) open up the DIN rail clip as shown below.



KD485-PROG Configuration

If a KD485-PROG is used to run one of the -ADE built-in programs, the installation is as per the -ADE product. If the KD485-PROG is used to run a user program, this program must first be uploaded to the KD485-PROG.

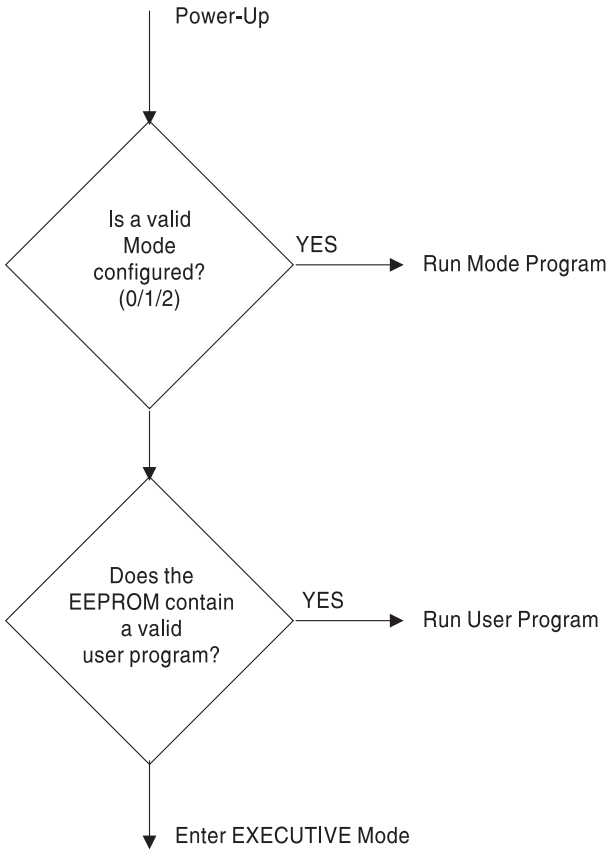
For information on how to produce such custom programs, refer to the **C Introduction** and **C Reference** chapters. The supplied hello.c and test.c are good examples to get you started.

User programs can be uploaded to the KD485-PROG in one of two ways:

1 Using TERM.EXE and the UF (Upload File) command. Refer to the **Executive** chapter. Any other terminal emulator which has an ASCII file upload facility can be used; note however that most of the modem-oriented comms programs (e.g. Procomm) will abort a file transfer if the PC's CD (carrier detect) input goes LOW and you may therefore need to wire this input to a HIGH level.

2 Using the Windows-based KDCFG.EXE program. This has a self-explanatory function for file upload.

The KD485-PROG executes programs in the following order:



Therefore, to execute your program, you need to set Mode to something invalid. With TERM, set Mode to e.g. 100 (md=100). With KDCFG, there is a specific MODE=OFF option.

Using the KD485 with Eurotherm Controllers

Series 2000

Communications to this series uses RS485, 2-wire mode. On the KD485 Port 2 connect:

1 Terminals TXA and RXA

2 Terminals TXB and RXB

The above places the KD485 RS485 port into a 2-wire mode.

To connect to the instrument(s) connect:

3	TXA+RXA	to	HE
4	TXB+RXB	to	HF
5	GND	to	HD

818, 815, 808, all 900 series units, Slave communications only

These instruments all use 4-wire RS485 mode. The connections required are shown in the following table. Note that the terminal labelling varies between different instrument types and the table reflects this.

Function in KD485	808	815	818	902/4	900EPC	Function in instrument
TXA	14	13	13	F3	X3	RX+ note 1
TXB	15	14	14	F4	X4	RX– note 1
RXA	12	15	15	F1	X1	TX+
RXB	13	16	16	F2	X2	TX–
GND	11	17	17	F5	X5	COM note 2

Note 1 If there are several instruments connected to the line they must be wired as a single point-to-point link (i.e. without any spur connections). A 100Ω resistor should be connected across the RX+ and RX– terminals of the instrument which is at the opposite end of the link from the KD485.

Note 2 The COM link is usually not required. In cases where significant interference with the communications is encountered, the use of this link may reduce the problem. The wire used for this link should be carried within the screening of the communications cable. The screen should not be used for this purpose.

PC3000

In order to connect between an LCM or ICM port and the KD485 the D-type connector will have to be removed from the cable supplied by Eurotherm and the wires connected directly to the terminals of the KD485. If Standard Eurotherm cables are not used for this purpose then the special connector will have to be obtained from BERG. It is recommended that a standard cable is modified.

Function in KD485	PC3000 Cable Colour	Connector Pin #	Function in PC3000
TXA	Blue	6	RX+ note 3
TXB	Green	5	RX– note 3
RXA	Black	4	TX+
RXB	Red	3	TX–
GND	White, Grey, Screen	7,8	0V

Note 3: A terminating resistor should be fitted across these terminals. Dependent on the port and module it may be necessary to fit a PCB mounted link to achieve this. Please refer to the PC3000 documentation.

Other Instrument Types

Refer to the table for the 818. Using the “Function in Instrument” column, identify the terminal with the equivalent functionality on the instrument to be connected. Connect accordingly. There are some generally applicable rules and exceptions to this:

Equipment which follows the above rule:

Eurotherm Drives
Eurotherm Thyristor Stacks
Eurotherm Process Automation Controllers

Equipment which does not follow the above rule:

Chessell Recorders

For these instrument types you should invert the polarity of both the RX and TX channels, i.e. if the instrument terminal is labelled “TX+” you should wire it as if it were called “TX–” and connect it to terminal RXB on the KD485.

Please remember that it is not possible, using the *standard* KD485, to mix instruments which use 4-wire communications on the same link as instruments which use 2-wire communications, even if they are using the same communications protocol.

The following two pages contain extracts from the Eurotherm Communications Handbook.

WIRING 2-WIRE485

To use RS485, buffer the RS232 port of the PC with a suitable RS232/RS485 converter. The Eurotherm Controls KD485 Communications Adapter unit is recommended for this purpose. Eurotherm does not recommend the use of a RS485 board built into the computer since this board is unlikely to be isolated, which may cause noise problems, and the Rx terminals are unlikely to be biased correctly for this application.

To construct a cable for RS485 operation use a screened cable with one (RS485) twisted pair plus a separate core for common. Although common or screen connections are not necessary, their use will significantly improve noise immunity.

The terminals used for RS485 digital communications are listed in the table below.

Standard Cable Colour	PC socket pin no. 25 way	PC Function *	Instrument Terminal	Instrument Function
White	3	Receive (RX+)	HF (B) or (B+)	Transmit (TX)
Black	16	Receive (RX−)		
Red	12	Transmit (TX+)	HE (A) or (A+)	Receive (RX)
Black	13	Transmit (TX−)		
Green	7	Common	HD	Common
Screen	1	Earth		

** These are the functions normally assigned to socket pins. Please check your PC manual to confirm .*

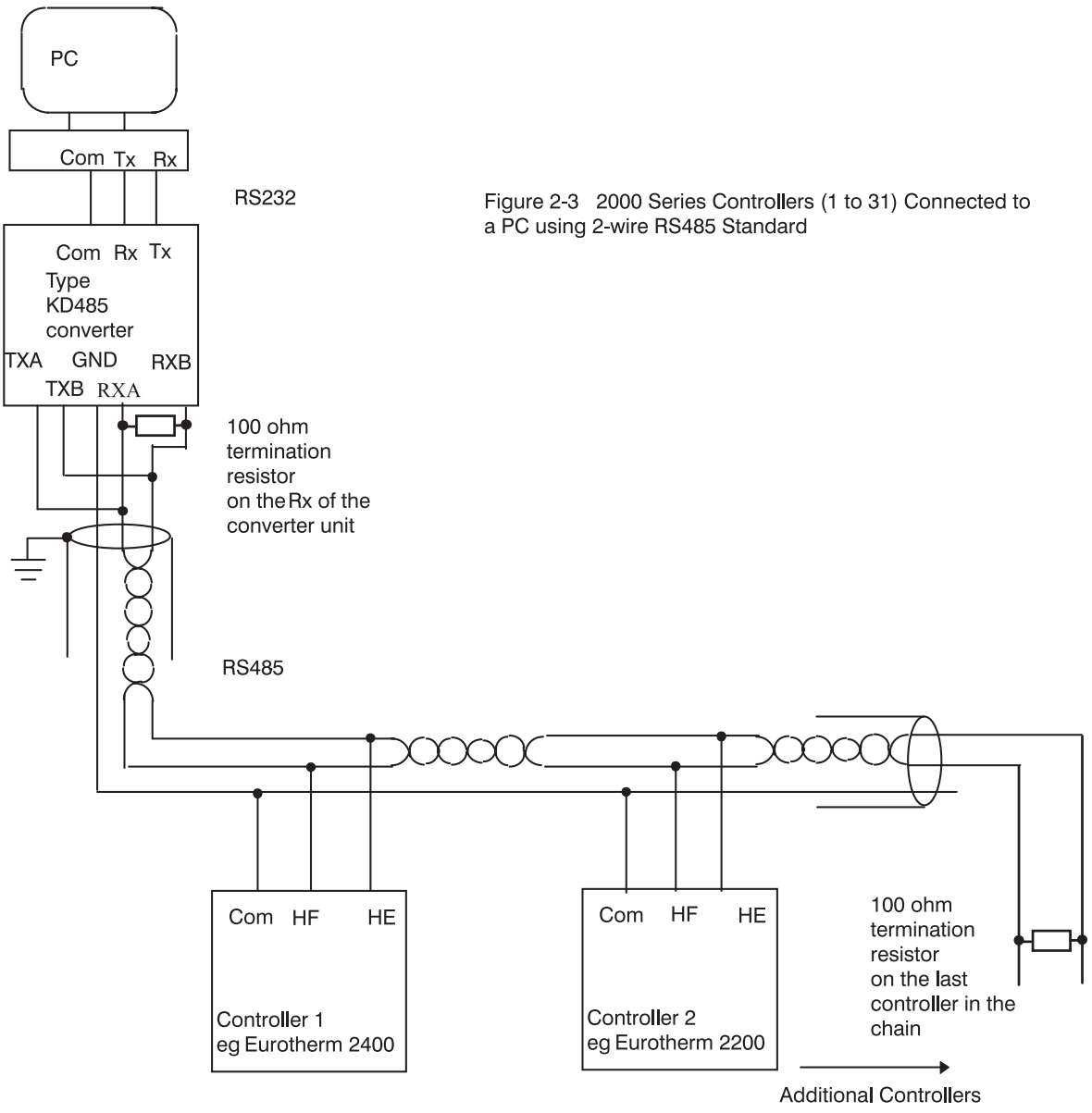


Figure 2-3 2000 Series Controllers (1 to 31) Connected to a PC using 2-wire RS485 Standard

WIRING RS422 AND RS485 CONTROLLERS

It is generally not possible to connect controllers using a 2-wire standard to controllers on a 4-wire standard. This may be required, for example, if 2000 series controllers are to be added to an existing installation.

It is, however, possible to modify the existing communications link by adding a special version of the KD485 converter unit, Supplied as KD485-ADE 422-422. This is shown in the diagram below.

The first KD485-ADE unit below is the standard RS232-RS422/485 type. It converts from 232 to 4-wire 485 and this link is used to communicate to the existing Eurotherm controllers. The second KD485-ADE is the special -422-422 version which converts from 4-wire to 2-wire 485 communications. It's input side behaves to the 4-wire link as another controller would on an existing system, whilst at the same time the communications messages from the computer are passed onto the output side of this unit. This is connected to the 2-wire communications link, that will contain the series 2000 controllers. Any responses from controllers on this link will cause data to be placed on to the 4-wire link and thence will be passed back to the computer.

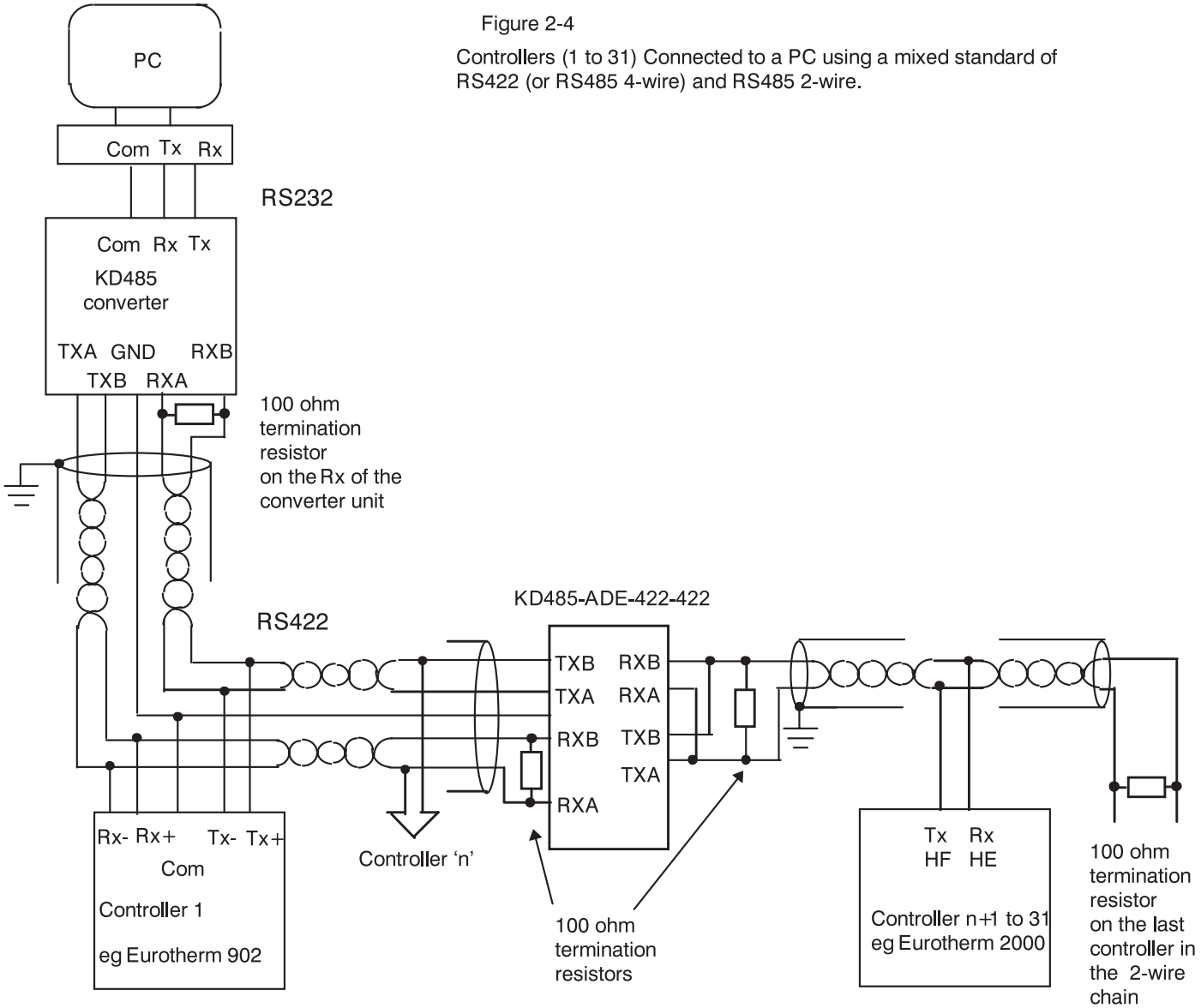
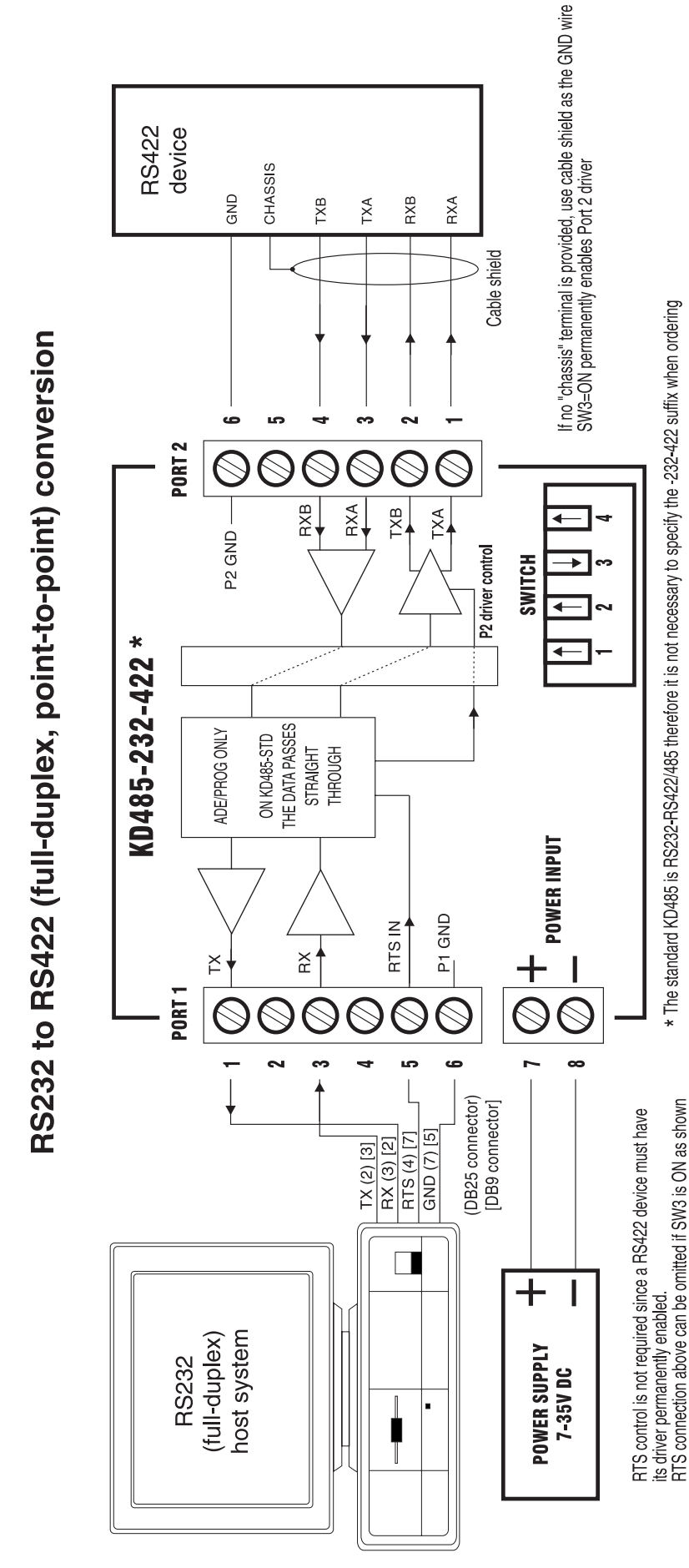


Figure 2-4

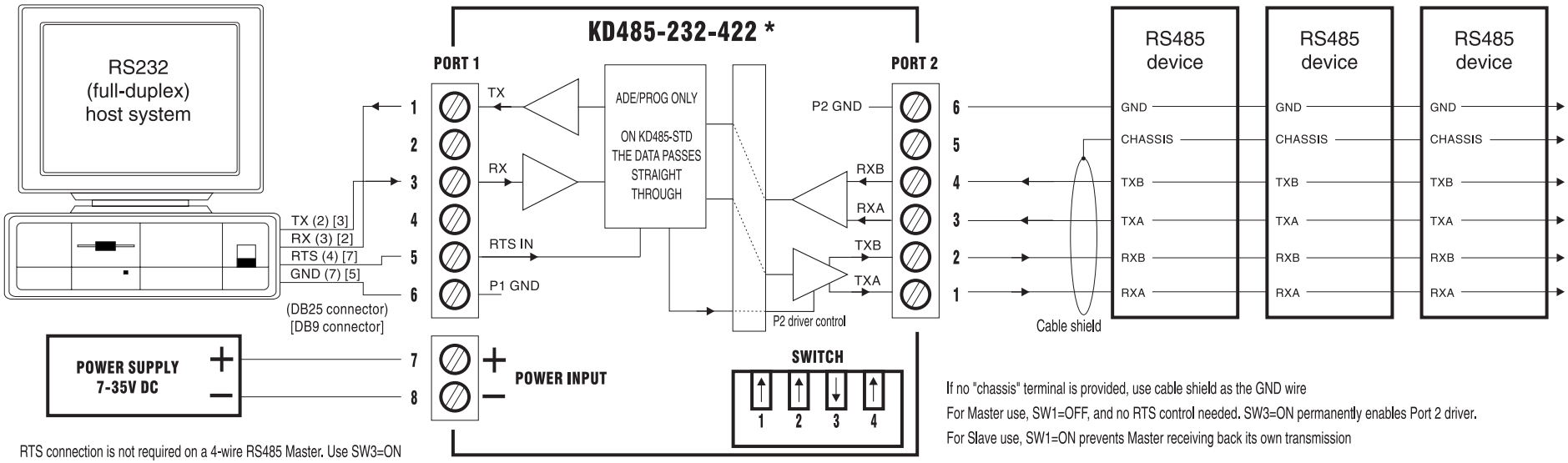
Controllers (1 to 31) Connected to a PC using a mixed standard of RS422 (or RS485 4-wire) and RS485 2-wire.

Wiring Examples

The following pages contain examples of typical applications.



RS232 to 4-wire RS485 (half-duplex, multidrop) conversion



RTS connection is not required on a 4-wire RS485 Master. Use SW3=ON

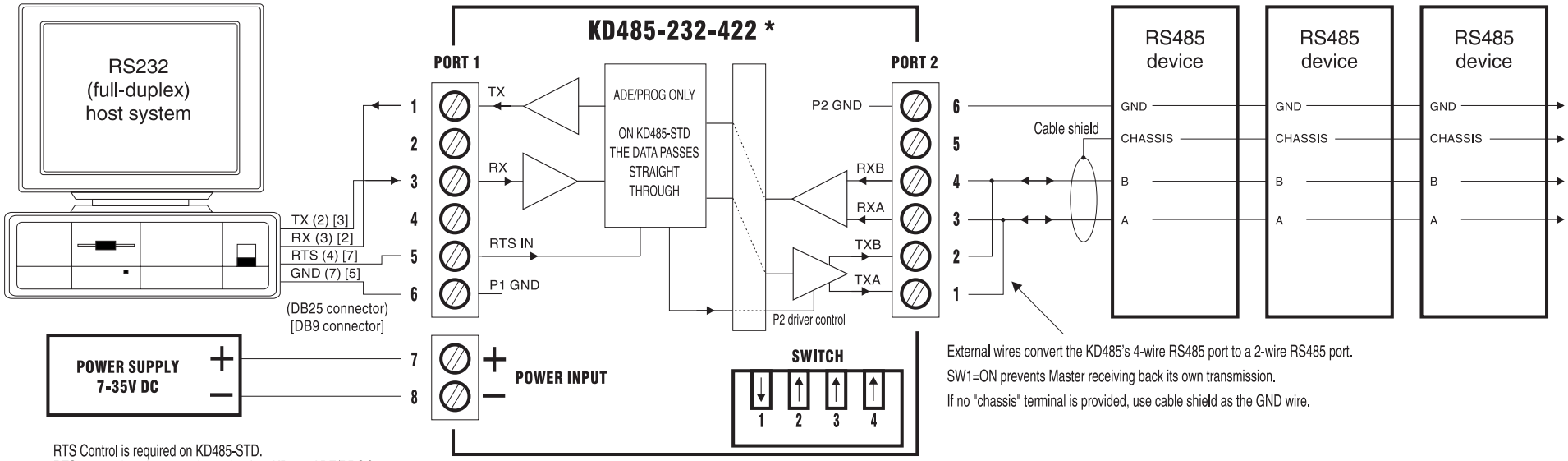
If no "chassis" terminal is provided, use cable shield as the GND wire

For Master use, SW1=OFF, and no RTS control needed. SW3=ON permanently enables Port 2 driver.

For Slave use, SW1=ON prevents Master receiving back its own transmission

* The standard KD485 is RS232-RS422/485 therefore it is not necessary to specify the -232-422 suffix when ordering

RS232 to 2-wire RS485 (half-duplex, multidrop) conversion

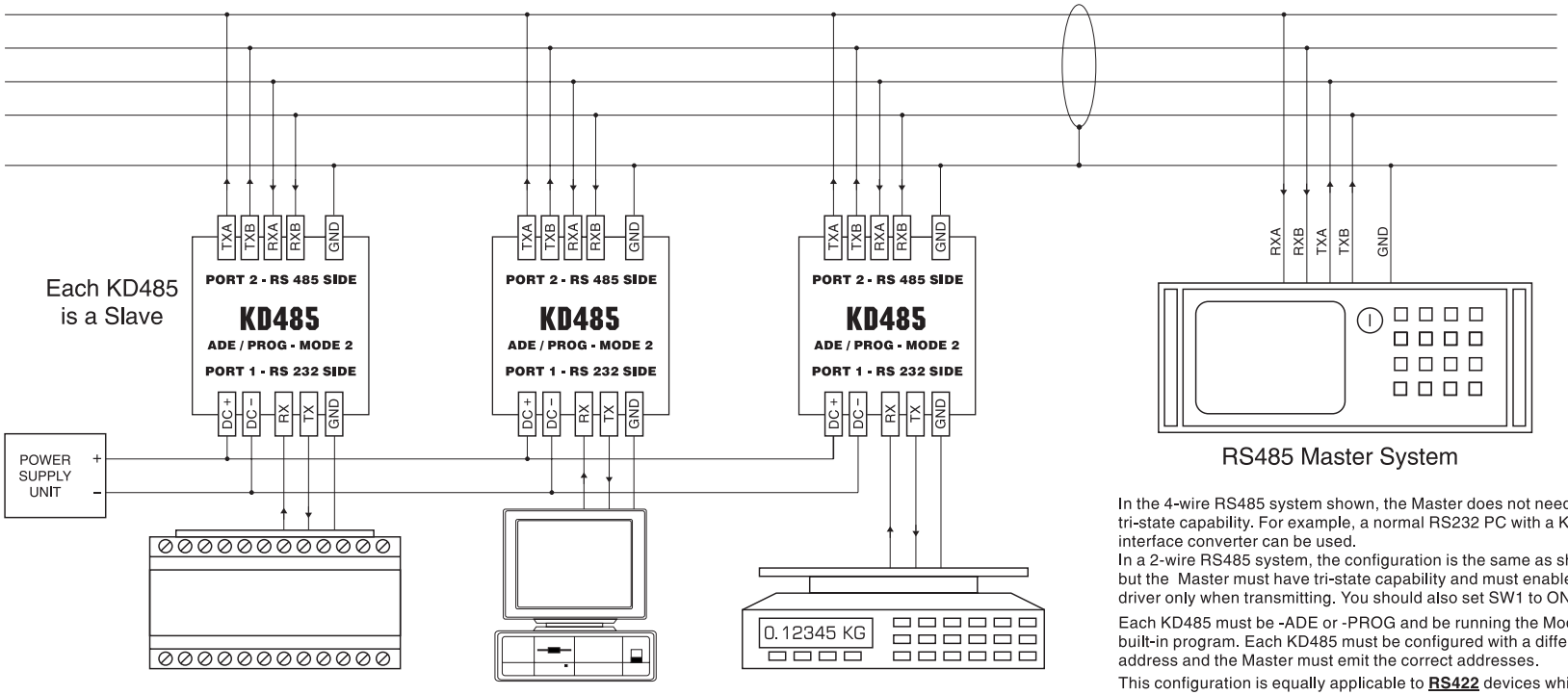


External wires convert the KD485's 4-wire RS485 port to a 2-wire RS485 port.
SW1=ON prevents Master receiving back its own transmission.
If no "chassis" terminal is provided, use cable shield as the GND wire.

RTS Control is required on KD485-STD.
RTS connection above can be omitted on KD485-ADE/PROG

* The standard KD485 is RS232-RS422/485 therefore it is not necessary to specify the -232-422 suffix when ordering

KD485-ADE/PROG as an Addressable Adapter (Mode 2 program)



RS232 devices with no built-in addressing or multi-drop support

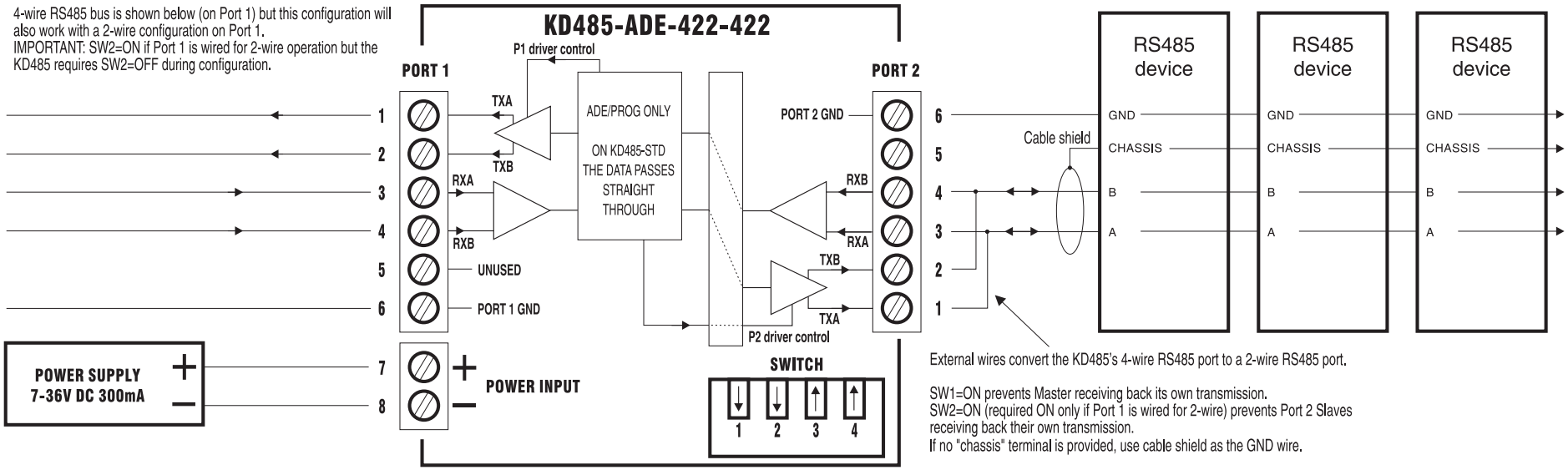
RS485 Master System

In the 4-wire RS485 system shown, the Master does not need tri-state capability. For example, a normal RS232 PC with a K422 interface converter can be used.
In a 2-wire RS485 system, the configuration is the same as shown but the Master must have tri-state capability and must enable its driver only when transmitting. You should also set SW1 to ON.

Each KD485 must be -ADE or -PROG and be running the Mode 2 built-in program. Each KD485 must be configured with a different address and the Master must emit the correct addresses.

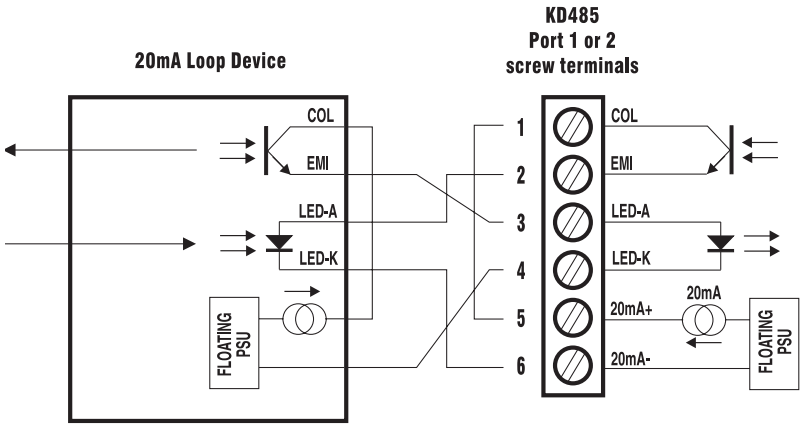
This configuration is equally applicable to **RS422** devices which do not support multi-drop operation, using KD485-ADE-422-422.

4-wire (or 2-wire) RS485 to 2-wire RS485 (half-duplex, multidrop) bus converter / repeater



IMPORTANT NOTE: If operating as 2-wire to 2-wire repeater, the KD485-ADE-422-422 requires a minimum gap of 4 character periods between messages, to ensure reliable line turn-around.

20mA Loop - General Connections



For full-duplex 20mA operation, each device needs a floating 20mA current source. This is used to supply power to the collector of the "output" transistor, whose emitter drives the "input" LED of the other device.

See KD485 Specification for maximum data rates.

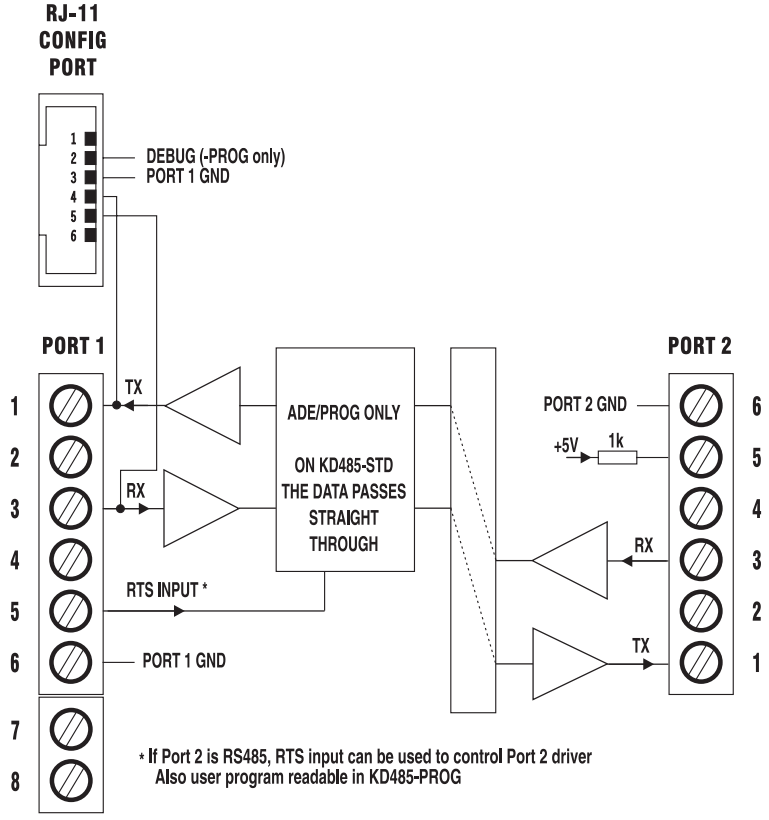
Ports & Connections

The KD485 has two ports, called Port 1 and Port 2. Each of these can be factory configured as RS232, RS422/485, or 20mA loop. The port types fitted to a particular KD485 are marked on its side label.

Port 1 is duplicated on an RJ-11 connector. The RJ-11 is not a third serial port; however it does carry a DEBUG output on the KD485-PROG.

The following text describes each of the three possible port types:

RS232 Ports



The RS232 port supports TX, RX signals only.

On Port 1 only, there is an additional input signal, RTS IN, which can be used to control an RS485 driver on Port 2. This operating mode is applicable only to the KD485-STD although the KD485-ADE Mode 0 program offers a similar function. On the KD485-PROG this input is user program readable.

RS232 Ports - Detail Description of Terminals

The numbers in brackets are the markings on the KD485 terminals.

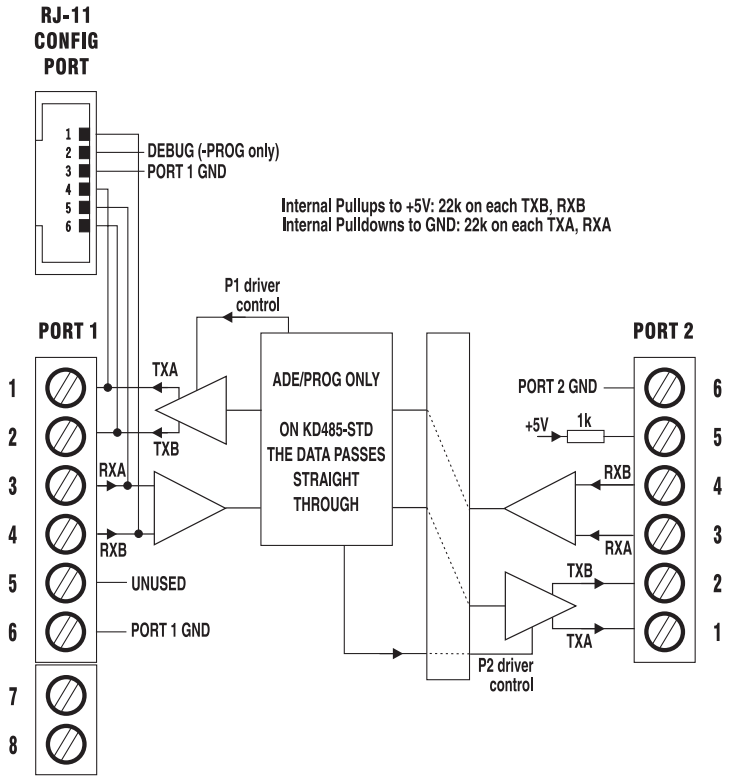
TX (1) — "Transmit Data" output.
ADE/PROG: when XON/XOFF is enabled as an RX handshake, XON/XOFF characters are automatically transmitted from the TX terminal to control the data flow into the RX input.

RX (3) — "Receive Data" input.
ADE/PROG: when XON/XOFF is enabled as a TX handshake, any XON/XOFF characters arriving at this pin will be stripped by the KD485 and will control the data flow from the TX output.

RTS IN (5) — "RTS" input. On the KD485-STD, and on KD485-ADE Mode 0, this signal directly controls the tri-state state of the Port 2 driver if Port 2 is RS485. In other cases this signal is ignored. On the KD485-PROG this signal is software-readable.

GND (6) This signal must always be connected. This combines both Protective Ground and Signal Ground functions.

RS422/485 Ports



This is a multi-purpose port which supports the following operating modes:

- **RS422.** This is a four-wire (plus ground) interface whose driver is permanently enabled.
- **4-wire RS485**. This is a four-wire (plus ground) interface whose driver can be enabled or disabled; this is called tri-state capability. This interface is often incorrectly called “RS422”.
- **2-wire RS485**. This is a two-wire (plus ground) interface whose driver must have tri-state capability.

The operation of the ports depends on whether it is Port 1 or Port 2, and on the KD485 version. The following notes cover the configuration of the most common product versions:

	STD Port 1	STD Port 2	ADE/PROG Port 1	ADE/PROG Port 2
RS422	SW2=OFF	SW1=OFF SW3=ON	SW2=OFF	Select Mode 0 program. SW1=OFF
4-wire RS485 **	Not supported	Requires RTS Control. Port 1 must be RS232. SW1=ON.	Select Mode 1 program & enable Port 1 driver to be controlled. * SW2=ON.	Select Mode 1 program. SW1=ON.
2-wire RS485	Not supported	Requires RTS Control. Port 1 must be RS232. SW1=ON. Interconnect TXA-RXA & TXB-RXB.	Select Mode 1 program & enable Port 1 driver to be “controlled”. * SW2=ON. Interconnect TXA-RXA & TXB-RXB.	Select Mode 1 program. SW1=ON. Interconnect TXA-RXA & TXB-RXB.

* This configuration option is available only via KDCFG.EXE. It cannot be done with a dumb terminal.
** The Master on a 4-wire RS485 bus can be an RS422 device because driver can be permanently enabled.

RS422/485 Ports - difference between RS422 and RS485?

RS422 and RS485 drivers and receivers have identical electrical characteristics. Both systems transmit each signal with two wires. Each signal is driven with a differential driver, and received with a differential receiver. Both of the driver outputs swing between 0V and +5V.

The main difference between the two systems is that while RS422 is a four-wire system suitable only for point-to-point use, the driver in a RS485 system has tri-state capability (its output can be disabled) which allows multiple transmitters to be connected to the same bus. RS485 thus supports "multi-drop" operation. In multi-drop systems there is always one device which is permanently a "master" and which periodically polls (sends messages to, requests data from) the "slaves". A slave never initiates a communication.

There can be more than one Master (for fault tolerance, or other reasons) but separate precautions must exist to ensure only one can be driving the bus at any one time.

Furthermore, RS485 exists in two versions: 4-wire and 2-wire. The sole advantage of a 2-wire system is that it uses only two wires, (plus the ground connection, or the cable shield) and is thus cheaper to install.

The main advantage of a 4-wire system is more subtle: because the master is driving a pair of wires which no other device may drive, the master's driver does not need tri-state capability. Many systems do not have tri-state capability, usually because their software was not written for multi-drop operation. The other advantage of a 4-wire system is that it is theoretically possible for the slave to respond before the master has finished transmitting its poll, without risk of bus contention, but no properly designed system relies on this.

Note that 4-wire RS485 with the driver left permanently enabled is the same as RS422.

RS422/485 Ports - Grounding

With an isolated device such as the KD485, an extra conductor is required for interconnecting the grounds between the devices connected to the RS485 bus. The cable shield can be used for this purpose.

☞ Although it is not unusual to find products which have the ground connection missing, you can get away with this only where the common mode voltage between all the devices attached to the *same* cable will never exceed the RS422/485 common mode voltage range which is typically -7V to +12V. This is difficult to guarantee in practice; this is why the KD485 is an isolated converter with a GND connection!

RS422/485 Ports - A/B Terminal Markings

The standard specifies that the terminals should be labelled "A" and "B". It also specifies their polarity: during the inter-character space (i.e. during the one or more stop bits which follow each character in asynchronous data) A is at ground potential and B is at +5V. Many manufacturers mis-label the terminals; some are reversed while others are marked + and – .

RS422/485 Ports - Detail Description of Terminals

The numbers in brackets are the markings on the KD485 terminals.

TXA (1), TXB (2) "Transmit Data" output.
ADE/PROG: if port 2 is used as a RS422 port, and when XON/XOFF is enabled as an RX handshake, XON/XOFF characters are automatically transmitted from TX to control the data flow into the RX input.

RXA (3), RXB (4) "Receive Data" input.
ADE/PROG: when XON/XOFF is enabled as a TX handshake, any XON/XOFF characters arriving at this pin will be stripped by port 2 and will control the data transmitted by the TX output.

TEST OUT (5) This pin is internally connected, via a 1k resistor, to an internal DC +5V supply referenced to pin 6. It is provided mainly for test purposes, although current (limited only by the resistor) may be drawn from it for RS485 bus pullup purposes.

GND (6) This is an isolated ground signal which should always be connected to the ground terminal(s) of the other RS422/485 equipment.

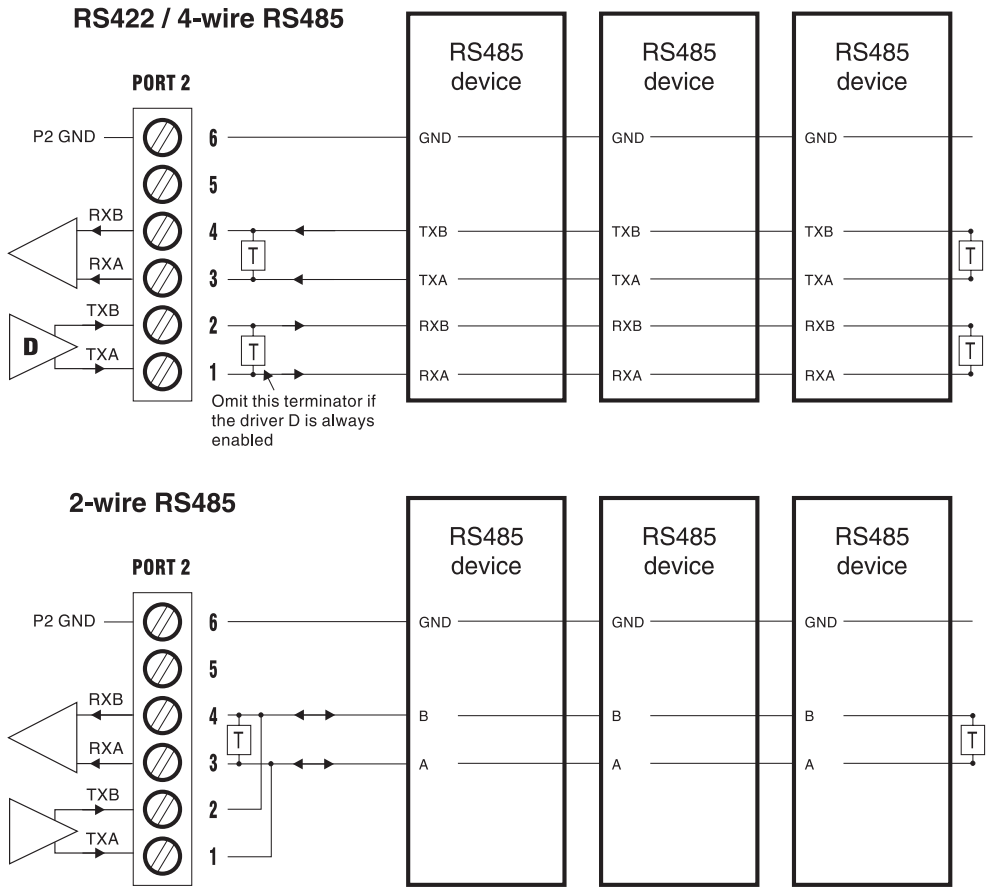
☞ Do not enable XON/XOFF on an RS485 port. Due to the half-duplex data flow, the handshake is unlikely to work as expected.

RS422/485 Ports - Terminators

Termination resistors are not normally required, because the KD485 uses controlled-slew-rate RS422/485 drivers and on cables below around 300m in length the reflections decay within the risetime of the signal.

When driving cables beyond 300m, you can connect a resistor equal to the cable impedance (typically 100-220 ohms) at each end of the cable (if RS485) or at the end farthest from the driver (if RS422).

Terminator Placement



To minimise DC loading on the driver(s) and to maximise the effectiveness of pullup/pulldown resistors (see below), a capacitor (e.g. 10nF) should be connected in series with any termination resistor.

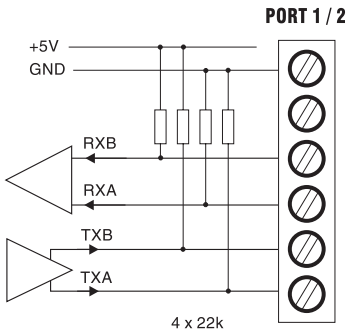
RS422/485 Ports - Bus Pullups

With multi-drop systems, it is important that when *none* of the connected devices is driving the bus, the level on the bus represents the inter-character space, i.e. B must be at a higher potential than A.

However, particularly where a number of devices are multi-dropped, driver leakage and other factors do sometimes cause A to float *above* B during the tri-state condition, in which case any attached receiver will see a permanent “start bit”; this is also known as a “break level”. The effect of this depends on how well the firmware in the various devices is designed, and it can manifest itself as comms errors which are consistently cleared by a re-transmission of the data.

The KD485 contains internal pullup and pulldown resistors whose purpose is to ensure that when the bus is not being driven, B is above A:

Internal Pullup Resistors



However, these resistors may fail to work in the following cases:

1 Where another instrument also contains its own pullups/pulldowns and they are connected to the wrong (opposite) voltages. This rather fundamental error is not uncommon.

2 Where there is a terminator resistor (e.g. 100-200 ohms) connected directly across the 485 bus. The 22k resistors in the KD485 cannot develop a voltage across 100R which is sufficient to be a defined RS485 "1" or "0" level (200mV or more is needed). A solution might have been to use much lower value resistors but this would present an excessive load if many KD485 units were present on the bus.

To obtain a well defined level on the bus, one solution is to connect a capacitor (e.g. 10nF) in series with each of the terminator(s).

Another is to use external resistors to supplement those inside the KD485; this is where the +5V supply (available on pin 5, Port 2 only) is useful. Such external resistors are needed only in one place on a 485 bus; anywhere will do.

It is easy to verify, with a voltmeter, that B is above A when no communications are taking place. The voltages must be measured *between* A and B, or relative to the isolated ground.

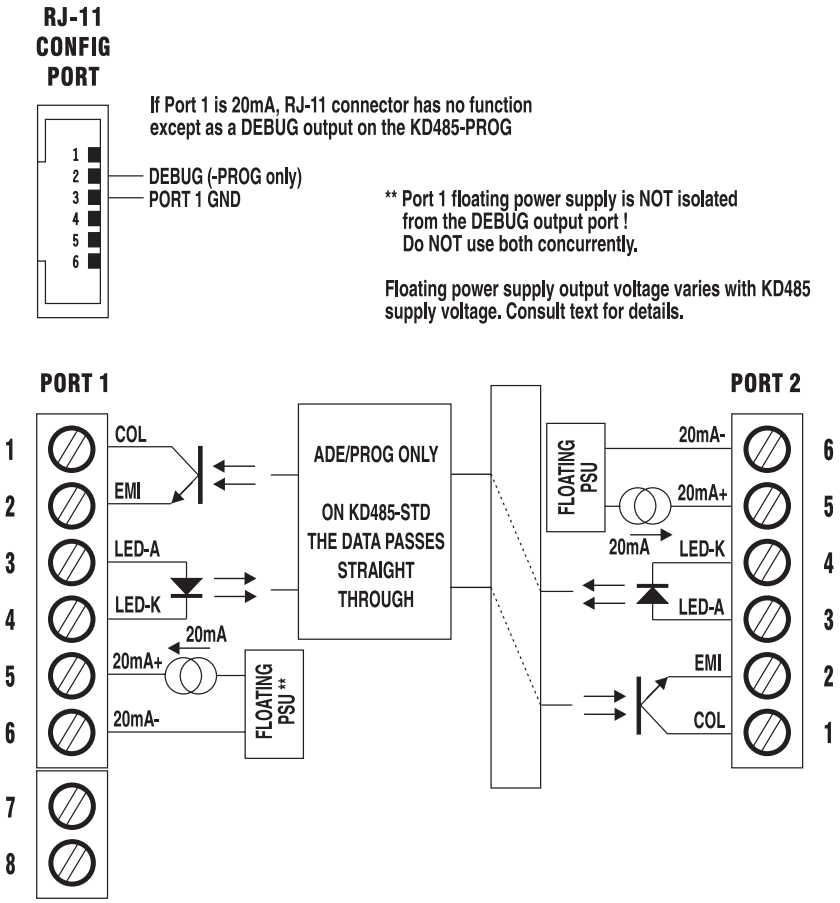
20mA Loop Ports

The 20mA port supports TX, RX signals only.

20mA Loop Ports - Detail Description of Terminals

The numbers in brackets are the markings on the KD485 terminals.

COL (1), EMI (2)	"Transmit Data" output. These are the collector and emitter terminals of the output transistor which is used to switch the externally supplied 20mA loop current. ADE/PROG: when XON/XOFF is enabled as an RX handshake, XON/XOFF characters are automatically transmitted from the TX terminal to control the data flow into the RX input.
LED-A (3), LED-K (4)	"Receive Data" input. These are the anode and cathode terminals of the LED which is used to receive the externally switched 20mA loop current. ADE/PROG: when XON/XOFF is enabled as a TX handshake, any XON/XOFF characters arriving at this pin will be stripped by the KD485 and will control the data flow from the TX output.
20mA+ (5), 20mA– (6)	This is the output of the floating 20mA constant current generator.



☞ KD485-ADE/PROG only: The DEBUG output port is NOT isolated from the Port 1 20mA constant current generator. Therefore, if you are using a KD485-PROG whose Port 1 is a 20mA port, do not make simultaneous connections to Port 1 terminals 1,2 and to the DEBUG port (RJ-11 connector pins 2,3). The converter may otherwise be damaged. This requirement also applies to the KD485-ADE in so far as simultaneous connections to Port 1 terminals 1,2 and RJ-11 pins 2,3 are prohibited, even though the DEBUG port is not functional on the KD485-ADE.

Both the output transistor and the input LED are reverse polarity protected, with parallel diodes.

The internal supply voltage to the 20mA generator (i.e. its open circuit voltage) is not regulated. It is approximately equal to the KD485 supply voltage.

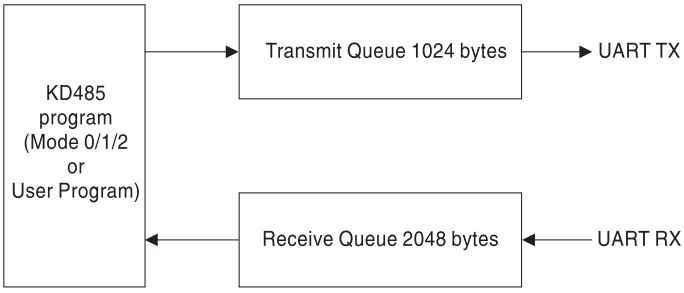
For example, if the KD485 is powered from +24V, the compliance of the 20mA current source is about 24V.

☞ If both KD485 ports are 20mA type, KD485 supply voltage is limited to **26V**, to limit the maximum possible internal heat dissipation.

KD485-ADE and KD485-PROG - General Data Flow Details

I/O Queues

Each KD485 port contains two queues: a transmit (TX) queue and a receive (RX) queue:



A user program in the KD485-PROG does not access the port hardware directly. Any I/O functions communicate only with the queues. The transfer of data between a queue and its port is performed as a background process (i.e. under interrupts) by the system firmware and is transparent to the user program.

Transmit Queues

A TX queue greatly enhances performance in both computational and datacomms areas, by allowing a user program to perform computations while previously-generated data is being transmitted under interrupts.

With 1024-byte TX queue, a user program could theoretically generate up to 1024 bytes of data even if the output device was BUSY throughout that time. However, to allow for insertion of XON/XOFF characters into the data stream and other reasons, the KD485 limits the filling-up of its TX queues to 1024–4 bytes.

Receive Queues

An RX queue is necessary to avoid data loss when handshaking the incoming data because most systems continue to transmit data for some time after being told to stop. An RX queue also permits the program to continue operations while data is arriving, and extract it from the queue when required.

When the KD485-ADE is used as an Addressable Adapter, the RX queue buffers any data received from the attached RS232 device, until it is retrieved by the RS485 bus Master.

Handshaking

The KD485 supports XON/XOFF handshakes only, and they are applicable to all port types except RS485. RS485 systems are usually half-duplex and do not have handshakes.

There are two types of handshakes: transmit (TX) handshakes and receive (RX) handshakes. A TX handshake signal is *received by* the KD485 and controls the transmit data flow *from the KD485*. An RX handshake signal is *transmitted by* the KD485 and controls the receive data flow *into the KD485*.

When the KD485 port is receiving data, if the free space in its RX queue falls below 512 bytes, it transmits an XOFF. When the space in the RX queue increases above 900 bytes the KD485 transmits an XON.

☞ From the above it is clear that when a device which is sending data to the KD485 is told by the KD485 to stop transmitting, it must not transmit more than 512 additional bytes after that. A failure to meet this requirement will usually result in loss of data.

When transferring binary data, XON/XOFF characters could appear accidentally within the data. In such cases, an XON/XOFF handshake in which the XON/XOFFs flow in the same direction as the data must be disabled. Failure to observe this will result in a locked-up system, and/or in the XON/XOFF codes (11 and 13 hex) disappearing from your data. This may not be immediately apparent, and will certainly not be apparent if you are testing your system with printable data only, or only with short messages!

The Executive

The Executive is a special mode used to configure the KD485, using a serially-connected terminal or a PC.

Only the "intelligent" KD485 versions (KD485-ADE and KD485-PROG) have the Executive. The KD485-STD requires no such configuration and does not have this feature.

The Executive is normally accessed using the supplied TERM.EXE terminal emulation program. A Windows-based configuration program, KDCFG.EXE, is also supplied and can be used as a more user-friendly alternative to the "dumb terminal" commands described here.

Basic Principles

The Executive is a command-line-driven user interface, with a **>** prompt appearing after each line typed-in. Backspace editing is supported, but no cursor commands. This simple interface is deliberate, to allow field configuration with a small hand-held terminal such as the 2-line Psion Organiser.

The KD485 can enter the Executive at power-up, or following any attempt to run a program, if there is no program to run. This happens under the following conditions:

KD485-ADE: no Mode has been configured.
KD485-PROG: no Mode has been configured, and no valid user program is found.

Each configuration command consists of a two-letter command code, optionally followed by parameters. Unless specified otherwise, nothing is case-sensitive. No spaces are allowed as parameter separators; only commas may be used where shown. Optional items are shown in square [] brackets.

Each new setting is immediately stored in an EEPROM (non-volatile memory) and there is therefore no "save" command. To exit the Executive mode (i.e. to run whichever program has been selected or uploaded) you issue the RU (run) command, or interrupt the KD485 power.

The Executive always runs via Port 1 and always uses 9600,n,8,1. The terminal does not need to support any handshakes, although XON/XOFF should be enabled if possible.

☞ The port 2 (RS485) driver is enabled while in the Executive. If port 2 is connected to a multidrop bus, ensure that no other device is trying to drive that bus while in the Executive.

The following section lists each command in the form of a definition, followed by an example.

Pn – Port Configuration

```
port=baudrate[,parity,bits/word,stopbits[,RXxon/xoff,TXxon/xoff]]
p2=9600,n,8,1,n,n
```

The following table lists valid values:

baud rates	30 37.5 50 75 100 110 134.5 150 300 600 1200 2000 2400 3600 4800 7200 9600 19200 38400 57600 115200
bits/word	7 8
parity	n e o (none/even/odd)
stop bits	1 2
RX XON/XOFF h/shake	y n (on/off)
TX XON/XOFF h/shake	y n (on/off)

Both ports have equal capabilities as far as this configuration is concerned. However, the handshakes should never be enabled on a port which is half-duplex.

Specifying the baud rate only (e.g. "p2=9600") selects "n,8,1,n,n" for the other parameters. Specifying everything but the handshakes selects "n,n" for those.

Port 2 is *immediately* initialised to the new parameters, but port 1 remains at 9600,n,8,1 while in the Executive, otherwise the terminal would have to be reconfigured, too!

MD – KD485 Mode

```
mode=modenumber (0,1,2)
md=1
```

The above command sets the operating Mode. This simply selects one of several standard commonly used application programs which are already provided in the on-chip ROM. Both the -ADE and the -PROG products have these programs.

When one of these programs is enabled, it is executed at power-up, or following the RU command.

☞ On the -PROG, the Mode program is executed in preference to any user program. To run a user program instead, you must set the Mode to an invalid value, e.g. 100. If using KDCFG, set it to "OFF".

Port types: All the Mode programs currently provided assume that port 1 is never tri-state, i.e. is RS232 or RS422. Also, a tri-stateable (i.e. RS485) port 2 is implicit in modes 1 and 2.

Valid Modes are 0, 1 or 2. Please see the **Overview** chapter for their details.

AD – RS485 address – applies to Mode 2 only

```
address=address,lead-in,timeout
ad=\65,\101,25
```

The above command specifies the RS485 address, the optional lead-in byte, and the timeout. All three values must always be specified.

The first two values are entered in decimal, 1-3 digits, each with a "\" prefix. Allowed range is 0-255 for both. Note that address values of 254 and 255 have special functions - see above. To disable the lead-in byte, enter "\0" (zero) for its value.

The timeout (in ms) is entered without a "\" and can be any value 0-65535. In 9600 baud systems, a suggested initial value for this timeout is 20ms plus the worst-case response time of the device attached to port 1. The "response time" here is the delay between the device receiving the end of the poll and and the first byte of the response. In most devices, this response time is only a few ms.

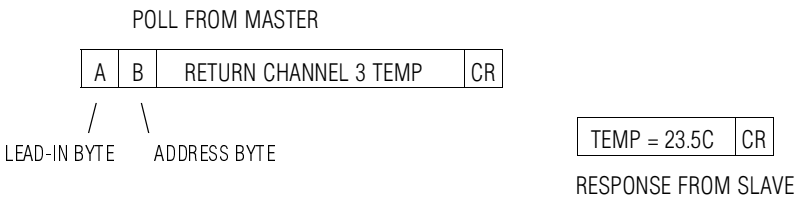
☞ RS485 systems do not have handshakes. Many RS232/422 systems do not have them either (although the KD485 can support XON/XOFF). If one KD485 port is set to e.g. 1200 baud and the other is set to e.g. 115200 baud, then you must ensure that the messages involved are short enough and/or infrequent enough to not overflow the KD485 RX buffers.

The value of the address byte must match the value emitted by your RS485 host system, and each KD485 on the RS485 bus must be configured to a *different* value.

The purpose of the lead-in byte is simply that it makes communications more robust. The receiving device (the KD485 in this case) can discard all data until this byte is received; only then does it look for a valid address byte. This scheme vastly reduces the probability of an accidental address match in a noisy system.

A lead-in byte also enables the use of an address value which may appear within the data.

The following illustrates one possible RS485 data exchange, demonstrating the use of the lead-in byte "A" and an RS485 address byte "B":



UF – Upload a "user program" file – KD485-PROG only

```
uf
```

The above command prompts for and requires a "Y" confirmation. Any other key terminates the upload.

A "Y" confirmation invalidates any existing program in the EEPROM (by overwriting its checksum) and places the KD485 in a receive mode where it expects data in the standard Intel Hex format.

At this point (within 60 seconds) you must initiate file upload in your terminal emulation program (with **ctrl-u** if using the supplied **term.exe** program), or press ESC to terminate the upload mode. Almost any terminal emulation program capable of uploading an ASCII file can be used. No handshake is required because, at 9600 baud, the KD485 can write each hex record to the EEPROM faster than the data can arrive.

When reading each hex line, the KD485 discards everything preceeding the initial ":" character. The line must end with an LF, optionally preceeded by a CR. A CR-only ending is not permitted. Maximum line length is 128 bytes, including the CRLF. Only type 00 (data record) and type 01 (end record) record types are used; all other types are ignored. Gaps in the data exceeding 5 seconds will terminate the upload mode.

Each hex record is written into the EEPROM as soon as it is received and its address and checksum are verified. When the end record is received, the program size and checksum fields in the EEPROM are updated. The program is now ready to run, either with the RU command, or by interrupting KD485 power.

The "end" record is mandatory, as are correct checksums on each line of the data. Any checksum errors, or address fields which are outside the EEPROM address range, terminate the upload mode and place the KD485 into a "data dump" mode where all data is discarded until there is a gap of a few seconds. This allows you to terminate the upload in your terminal emulator program in an orderly manner.

If you need to prevent the user program executing at power-up (e.g. to get to the Executive instead, without using the front panel pushbutton switch to do it) type "UF" followed by a "Y", then press <enter>, wait for 5 seconds, and do not upload a program. This permanently invalidates the user program.

☞ A "UF" command with a "Y" confirmation irreversibly invalidates any existing program. Use it with care!

☞ The uploaded program will not execute if a valid Mode program is selected. The KD485-PROG executes a valid Mode program in preference to a User Program. Use e.g. md=100 to select an invalid Mode program.

RU – Run program

```
ru
```

The above command reboots the KD485. Also see the UF command above.

TE – Test ports

```
te
```

The above command runs a continuous loopback test on Port 2. The test stops when any key is pressed on the terminal. The test runs at the currently configured Port 2 baud rate. It uses the string "A"..."Z" repeated. This string is *not* output to the terminal; this function is intended for oscilloscope monitoring of data flow.

This test requires a loopback connection on Port 2: if Port 2 is RS422/485, interconnect TXA–RXA and TXB–RXB, and a 220 ohm load resistor may be connected TXA–TXB for additional confidence. Ensure that nothing else is connected to port 2.

TS – Test Slave device

```
ts[=h]
```

The above command is useful for testing slave devices attached to Port 2. A ":" prompt is displayed. All characters entered (except the final <enter>) are transmitted to the slave device. Non-printing characters are entered as three decimal digits with a "\" prefix, e.g. \013 is a CR. *Three* digits must always be entered. Up to 80 characters may be entered after the ":" prompt.

Any data returned from the slave device is displayed as received. If it contains non-printing characters, it can be displayed in hex instead, using the command form "ts=h".

Port 2 = **RS485:** The port 2 driver is permanently enabled while in the Executive. To enable testing of 2-wire RS485 devices, the driver goes tri-state for 10 seconds following the transmission of the last character of the enquiry string and this allows more than ample time for the slave device to drive the bus and return its response. The Executive **>** prompt does not re-appear until this time has expired.

The timeout is extended, indefinitely if necessary, by any data returned from the slave device. This enables the KD485 to be used to monitor data on Port 2; you simply enter a null string (<enter> by itself) to the ":" prompt.

The TS function is not suitable for port 2 baud rates below 75 baud.

HE – Help

he

Presently this returns the firmware date and version only.

SOFTWARE UTILITIES

TERM.EXE

This is a basic terminal emulation program, with file upload and hex dump capability. It is a DOS application which works under DOS, and in a Windows 3.1, 95 or NT4 DOS box.

For usage details, type

```
term
```

with no parameters.

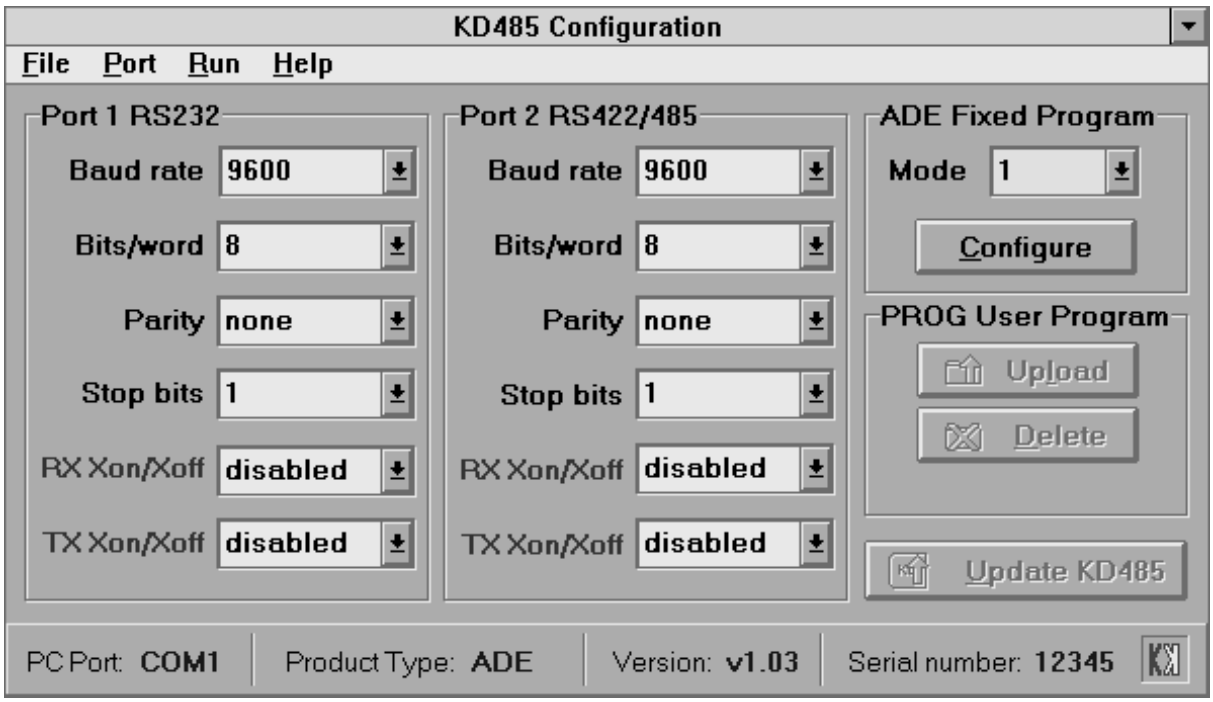
By default, term sets the COM port to 9600,n,8,1 and leaves it so configured upon exit. This is correct for the KD485 but there is an override switch to stop this.

For non-KD485 applications, term internally translates cursor keys to ANSI ESCape sequences, so if you have ansi.sys loaded, term.exe provides a simple "ANSI" terminal.

KDCFG.EXE

This is a Windows-based configuration program for both the KD485-ADE and the KD485-PROG. For installation instructions, see the readme.txt file on its diskette.

KDCFG performs the main functions of a dumb terminal as described on the preceeding pages, but it uses a proprietary protocol to interface to the KD485. The configuration is displayed using pull-down menus:



The port types (RS232 or RS422/485) are displayed at the top, and the product type, firmware version and serial number are reported in the lower part of the display.

☞ 20mA ports are displayed as "RS232".

☞ KDCFG requires that the KD485 is in the Executive mode (LED flashing rapidly) when it is started.

KDCFG cannot be used to run the TE and TS commands. It does however contain additional configuration items; for example, for "difficult" multi-drop systems containing poorly-designed instruments, customised RS485 driver turn-off delays can be specified for each of the baud rates above 4800 baud.

Troubleshooting

KD485 power-up problems

With a valid DC power input (7V to 35V), the voltage on the TEST OUT terminal (Port 2, pin 5) must be within the range +4.5V to +5.5V, measured relative to Port 2 pin 6.

KD485-ADE/-PROG only: When the KD485 is powered-up, the LED should blink twice, regardless of its configuration. If it does not illuminate at all, check DC power input. It must be in the range +7V to +35V and be connected solely to terminals 7 and 8 (on the Port 1 terminal block).

The KD485's power supply will also fail to start if the supply voltage rises from zero to the switching power supply startup voltage (typically 4V) very slowly. See the Specification section in the **Overview** chapter for details.

If after power-up the LED enters a repeating pattern of n rapid blinks followed by a longer pause, this indicates that a hardware fault has been detected by the KD485 power-up test firmware. The number of blinks "n" indicates the fault, but any such fault requires a return to the factory.

RS232 Communications Problems

1 Make sure TX,RX are correctly connected. RS232 is not a "bus" system; the TX terminal of one device feeds the RX terminal of the other device - unless one device is "DTE" and the other is "DCE". If in doubt, consult the device documentation.

2 KD485-ADE/PROG: check you have configured the KD485 to the correct baud rate and other parameters.

RS422/485 Communications Problems

1 Many RS422/485 equipment manufacturers have incorrectly labelled their A/B signals and you may need to swap them. Measured at the KD485 RS422/485 interface relative to the isolated ground M, a HIGH level (e.g. a start bit) is represented by A=+5V and B=0V, as per the published standard. Conversely, with no data being sent, A=0V and B=+5V. With significant loading (e.g. with cable terminators fitted) these voltages will be shifted towards the centre (+2.5V) but the *polarity of the difference* between A and B remains.

   One way (other than swapping the wires) of determining how the manufacturer has labelled their terminals is to read the product's manual and look for a detailed description of the terminal functions.

   Another is to measure the DC voltage on each of the product's terminals, with nothing connected and no data flowing, relative to ground. The terminal with the higher potential is our "B". This test is reliable only if the product has internal pullup/pulldowns and they are connected the right way!

2 Some RS485 systems use only the signal wires (2 or 4) and *no ground connection*. While this should work where the ground connections of the equipment involved are interconnected via another route, it is not a recommended practice because the finite common mode range (typically -7V to +12V) can easily be exceeded, resulting in comms errors or even equipment damage.

   The isolation property of the KD485 allows the grounds to be properly connected to their respective equipment without the danger of creating ground loops.

3 On KD485-STD in 2-wire RS485 mode (Port 2) only: check that your RS232 host system properly controls its RTS signal: HIGH when transmitting and LOW when listening. Note that RTS must remain HIGH until the very last bit has shifted out of the host's UART.

5 On KD485-ADE/-PROG running the **Mode 0** program with the RTS input unconnected: set SW3=ON.

6 Check the DIP switch settings.

Cannot enter the KD485 Executive

If unable to communicate when using the KDCFG.EXE Windows configuration program, this problem is normally the result of the KD485-ADE/PROG not being in the Executive mode *when KDCFG is started*. Exit KDCFG, ensure the "EXE" LED is flashing rapidly (by pressing the front panel button - see **Installation**), then re-start KDCFG.

If, upon power-up, the "EXE" LED is not flashing this indicates that the KD485 is running a program. To force entry into the Executive mode, hold down the switch (this takes a number of seconds) until the LED is flashing rapidly, then immediately release it. Then wait (several more seconds) until it again starts flashing rapidly. If you have a terminal connected to Port 1, you should now see a PRESS ENTER message repeated on the terminal.

1 Under Windows 95/98 or Windows NT, there can be a conflict between KDCFG.EXE and another serial port driver. Download the latest version of KDCFG.EXE from www.kksystems.com.

2 If the KD485 has a suffix like -422-422 or -422-20MA, etc, Port 1 is not RS232 and a suitable interface converter is required.

The following applies to configuration using the command line mode:

3 If the "EXE" LED is flashing rapidly and continuously then the KD485 is already in the Executive. Pressing ENTER should produce the KD485 **>** prompt. If this does not appear, it is most likely due to a communications problem in the connection to the terminal. Try the following:

- Reset the KD485 by interrupting power to it, and repeat 1 above if necessary.
- Press ENTER on the terminal several times until the **>** prompt appears.
- If using a non-TERM terminal, check it is configured for NO handshake (or XON/XOFF only, no hardware handshakes) and is set to **9600,N,8,1**.
- If using TERM, exit it (with ctrl-c) and re-enter.
- If using TERM, try a different serial port on the PC (e.g. COM2).

Intermittent Comms Errors - General

The usual cause is bad wiring or grounding, in an electrically hostile environment. Make sure the wiring is shielded, with the shield connected to a good ground. If interconnecting two devices, one of which is isolated (i.e. has a floating ground terminal) then you must interconnect the grounds of the two devices.

Intermittent Comms Errors - RS422/485

Check that your signal wiring is **properly shielded** and that the KD485 Port 1 / Port 2 **isolated ground** is connected to the ground of the other RS485 devices. While RS422/485 is known for its good noise immunity, this is true only if any induced common mode voltages are within the capability of the receiver. The DC common mode limits on RS485 are typically -7 to +12V and these can be exceeded under transient conditions if the isolated ground is not connected. Also, even if the common mode voltage is always within this range but includes a very high frequency component, the receiver may again not function properly. This is true for all RS485 devices, particularly when the ground terminal of an RS485 device is left unconnected.

Another cause of persistent comms problems may be that the bus is floating into an invalid state in between messages. See the discussion of Pullups in the **Ports and Connections** section.

In some systems, problems can occur if the value of the RS485 address byte occurs *within* a message.

Intermittent Comms Errors - KD485-PROG

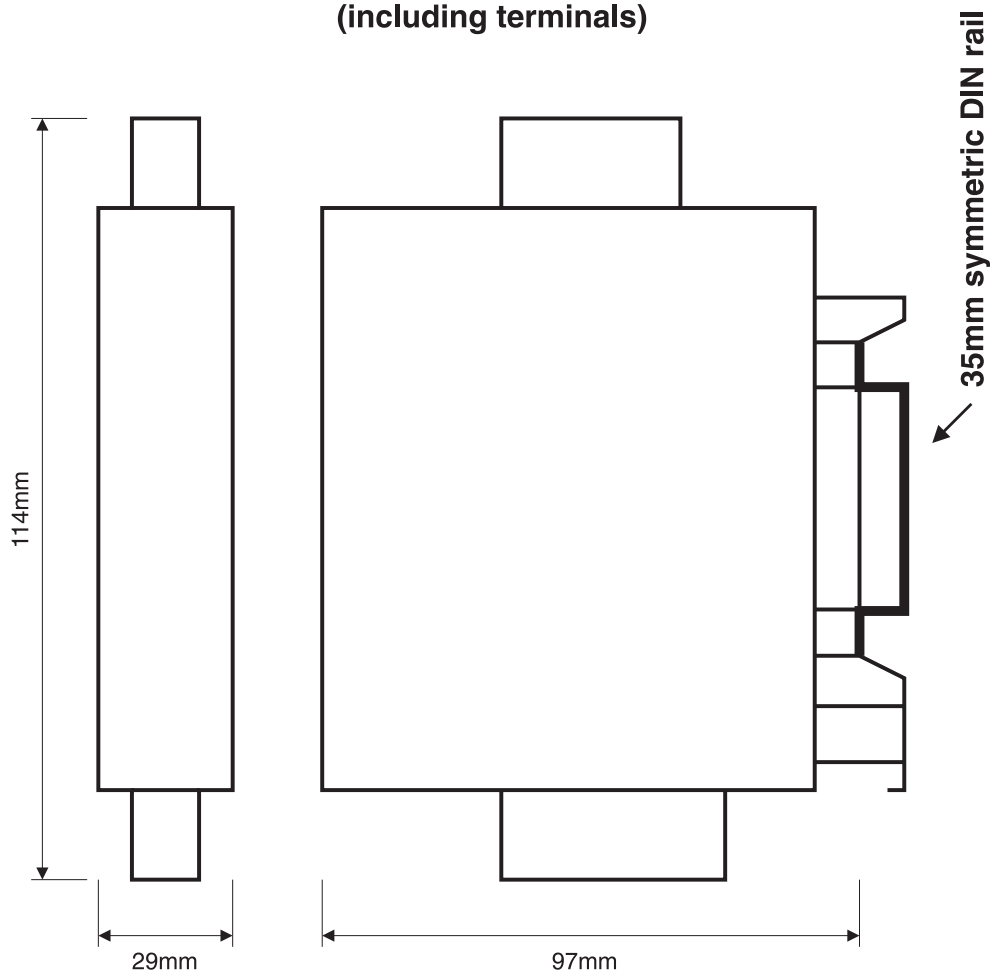
In the KD485-PROG you can experience runtime errors in your programs, if you have made programming errors. These may or may not cause the watchdog timer to trip. They are more likely to cause it to trip if your program has taken over the watchdog triggering. There are many possible runtime errors, ranging from endless loops to stack overflow. Some runtime errors can be triggered by invalid input data; your programs must be written to handle this.

ASCII Character Codes

hex code	decimal code	ASCII value	common function (if any)	hex code	decimal code	ASCII value	common function (if any)
#00	0		null	#40	64	'@'	–
#01	1	ctrl-A	–	#41	65	'A'	–
#02	2	ctrl-B	STX	#42	66	'B'	–
#03	3	ctrl-C	ETX	#43	67	'C'	–
#04	4	ctrl-D	EOT	#44	68	'D'	–
#05	5	ctrl-E	ENQ	#45	69	'E'	–
#06	6	ctrl-F	ACK	#46	70	'F'	–
#07	7	ctrl-G	bell	#47	71	'G'	–
#08	8	ctrl-H	backspace	#48	72	'H'	–
#09	9	ctrl-I	tab	#49	73	'I'	–
#0A	10	ctrl-J	line feed	#4A	74	'J'	–
#0B	11	ctrl-K	–	#4B	75	'K'	–
#0C	12	ctrl-L	form feed	#4C	76	'L'	–
#0D	13	ctrl-M	carriage return	#4D	77	'M'	–
#0E	14	ctrl-N	–	#4E	78	'N'	–
#0F	15	ctrl-O	NAK	#4F	79	'O'	–
#10	16	ctrl-P	–	#50	80	'P'	–
#11	17	ctrl-Q	XON	#51	81	'Q'	–
#12	18	ctrl-R	–	#52	82	'R'	–
#13	19	ctrl-S	XOFF	#53	83	'S'	–
#14	20	ctrl-T	–	#54	84	'T'	–
#15	21	ctrl-U	NAK	#55	85	'U'	–
#16	22	ctrl-V	SYNC	#56	86	'V'	–
#17	23	ctrl-W	–	#57	87	'W'	–
#18	24	ctrl-X	–	#58	88	'X'	–
#19	25	ctrl-Y	–	#59	89	'Y'	–
#1A	26	ctrl-Z	–	#5A	90	'Z'	–
#1B	27	ESC	escape	#5B	91	'['	–
#1C	28		–	#5C	92	'\'	–
#1D	29		–	#5D	93	']'	–
#1E	30		–	#5E	94	'^'	–
#1F	31		–	#5F	95	'_'	(underscore)
#20	32	' '	space	#60	96		–
#21	33	'!'	–	#61	97	'a'	–
#22	34	'"'	(double quote)	#62	98	'b'	–
#23	35	'#'	–	#63	99	'c'	–
#24	36	'$'	–	#64	100	'd'	–
#25	37	'%'	–	#65	101	'e'	–
#26	38	'&'	–	#66	102	'f'	–
#27	39	'	(single quote)	#67	103	'g'	–
#28	40	'('	–	#68	104	'h'	–
#29	41	')'	–	#69	105	'i'	–
#2A	42	'*'	–	#6A	106	'j'	–
#2B	43	'+'	–	#6B	107	'k'	–
#2C	44	','	(comma)	#6C	108	'l'	–
#2D	45	'–'	(minus sign)	#6D	109	'm'	–
#2E	46	'.'	–	#6E	110	'n'	–
#2F	47	'/'	–	#6F	111	'o'	–
#30	48	'0'	–	#70	112	'p'	–
#31	49	'1'	–	#71	113	'q'	–
#32	50	'2'	–	#72	114	'r'	–
#33	51	'3'	–	#73	115	's'	–
#34	52	'4'	–	#74	116	't'	–
#35	53	'5'	–	#75	117	'u'	–
#36	54	'6'	–	#76	118	'v'	–
#37	55	'7'	–	#77	119	'w'	–
#38	56	'8'	–	#78	120	'x'	–
#39	57	'9'	–	#79	121	'y'	–
#3A	58	':'	–	#7A	122	'z'	–
#3B	59	';'	–	#7B	123	'{'	–
#3C	60	'<'	–	#7C	124	'|'	(vert bar)
#3D	61	'='	–	#7D	125	'}'	–
#3E	62	'>'	–	#7E	126	'~'	–
#3F	63	'?'	–	Codes 126-255 vary.			

Overall Dimensions

KD485 Dimensions
(including terminals)



Eurotherm: International sales and service

AUSTRALIA Sydney
Eurotherm Pty. Ltd.
Telephone (+61 2) 9838 0099
Fax (+61 2) 9838 9288
E-mail info.au@eurotherm.com

AUSTRIA Vienna
Eurotherm GmbH
Telephone (+43 1) 798 7601
Fax (+43 1) 798 7605
E-mail info.at@eurotherm.com

BELGIUM & LUXEMBOURG Moha
Eurotherm S.A/N.V.
Telephone (+32) 85 274080
Fax (+32) 85 274081
E-mail info.be@eurotherm.com

BRAZIL Campinas-SP
Eurotherm Ltda.
Telephone (+5519) 3707 5333
Fax (+5519) 3707 5345
E-mail info.br@eurotherm.com

CHINA
Eurotherm China
Shanghai Office
Telephone (+86 21) 6145 1188
Fax (+86 21) 6145 2602
E-mail info.cn@eurotherm.com
Beijing Office
Telephone (+86 10) 8458 5757
Fax (+86 10) 8458 4521
E-mail info.cn@eurotherm.com

DENMARK Copenhagen
Eurotherm Danmark AS
Telephone (+45 70) 234670
Fax (+45 70) 234660
E-mail info.dk@eurotherm.com

FINLAND Abo
Eurotherm Finland
Telephone (+358) 2250 6030
Fax (+358) 2250 3201
E-mail info.fi@eurotherm.com

FRANCE Lyon
Eurotherm Automation SA
Telephone (+33 478) 664500
Fax (+33 478) 352490
E-mail info.fr@eurotherm.com

GERMANY Limburg
Eurotherm Deutschland GmbH
Telephone (+49 6431) 2980
Fax (+49 6431) 298119
E-mail info.de@eurotherm.com

INDIA Chennai
Eurotherm India Limited
Telephone (+91 44) 2496 1129
Fax (+91 44) 2496 1831
E-mail info.in@eurotherm.com

IRELAND Dublin
Eurotherm Ireland Limited
Telephone (+353 1) 469 1800
Fax (+353 1) 469 1300
E-mail info.ie@eurotherm.com

ITALY Como
Eurotherm S.r.l
Telephone (+39 031) 975111
Fax (+39 031) 977512
E-mail info.it@eurotherm.com

KOREA Seoul
Eurotherm Korea Limited
Telephone (+82 31) 2738507
Fax (+82 31) 2738508
E-mail info.kr@eurotherm.com

NETHERLANDS Alphen a/d Rijn
Eurotherm B.V.
Telephone (+31 172) 411752
Fax (+31 172) 417260
E-mail info.nl@eurotherm.com

NORWAY Oslo
Eurotherm A/S
Telephone (+47 67) 592170
Fax (+47 67) 118301
E-mail info.no@eurotherm.com

POLAND Katowice
Invensys Eurotherm Sp z o.o
Telephone (+48 32) 218 5100
Fax (+48 32) 218 5108
E-mail info.pl@eurotherm.com

SPAIN Madrid
Eurotherm España SA
Telephone (+34 91) 661 6001
Fax (+34 91) 661 9093
E-mail info.es@eurotherm.com

SWEDEN Malmo
Eurotherm AB
Telephone (+46 40) 384500
Fax (+46 40) 384545
E-mail info.se@eurotherm.com

SWITZERLAND Wollerau
Eurotherm Produkte (Schweiz) AG
Telephone (+41 44) 787 1040
Fax (+41 44) 787 1044
E-mail info.ch@eurotherm.com

UNITED KINGDOM Worthing
Invensys Eurotherm Limited
Telephone (+44 1903) 268500
Fax (+44 1903) 265982
E-mail info.uk@invensys.com
Web www.eurotherm.co.uk

U.S.A Ashburn VA
Eurotherm Inc.
Telephone (+1 703) 443 0000
Fax (+1 703) 669 1300
E-mail info.us@eurotherm.com
Web www.eurotherm.com

HA030692/4 (CN26268)

invensys
Eurotherm