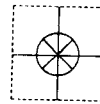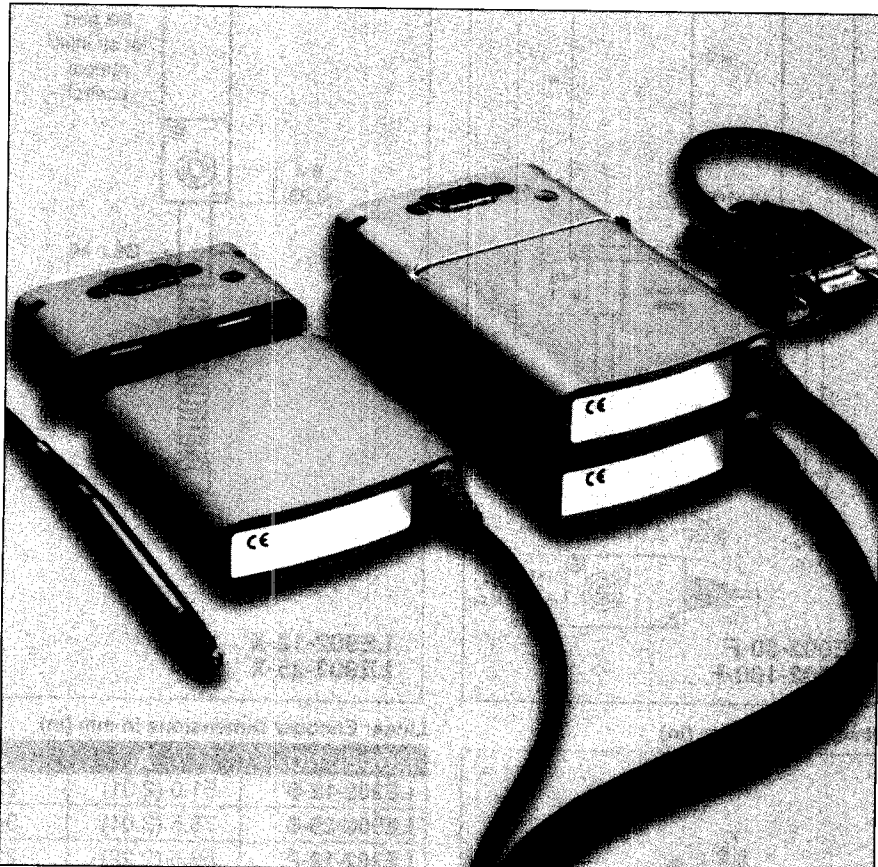$C\:E$

⊕ # User's Guide

Shown smaller than actual size

Shop on line at

omega.com™
ΩEOMEGA®

www.omega.com
e-mail: info@omega.com

LDN 232
RS232 Interface Module

omega.com™
ΩOMEGA®

OMEGAnet® On-Line Service www.omega.com	Internet e-mail info@omega.com

Servicing North America:

USA:
ISO 9001 Certified

One Omega Drive, P.O. Box 4047
Stamford CT 06907-0047
TEL: (203) 359-1660
e-mail: info@omega.com
FAX: (203) 359-7700

Canada:

976 Bergar
Laval (Quebec) H7L 5A1
TEL: (514) 856-6928
e-mail: info@omega.ca
FAX: (514) 856-6886

For immediate technical or application assistance:

USA and Canada: Sales Service: 1-800-826-6342 / 1-800-TC-OMEGA®
Customer Service: 1-800-622-2378 / 1-800-622-BEST®
Engineering Service: 1-800-872-9436 / 1-800-USA-WHEN®
TELEX: 996404 EASYLINK: 62968934 CABLE: OMEGA

Mexico:

En Español: (001) 203-359-7803
FAX: (001) 203-359-7807
e-mail: espanol@omega.com
info@omega.com.mx

Servicing Europe:

Benelux:

Postbus 8034, 1180 LA Amstelveen, The Netherlands
TEL: +31 (0)20 6418405
FAX: +31 (0)20 6434643
Toll Free in Benelux: 0800 0993344
e-mail: nl@omega.com

Czech Republic:

Rudé armády 1868, 733 01 Karviná 8
TEL: +420 (0)69 6311899
Toll Free: 0800-1-66342
FAX: +420 (0)69 6311114
e-mail: czech@omega.com

France:

9, rue Denis Papin, 78190 Trappes
TEL: +33 (0)130 621 400
Toll Free in France: 0800-4-06342
e-mail: france@omega.com
FAX: +33 (0)130 699 120

Germany/Austria:

Daimlerstrasse 26, D-75392 Deckenpfronn, Germany
TEL: +49 (0)7059 9398-0
FAX: +49 (0)7056 9398-29
Toll Free in Germany: 0800 639 7678
e-mail: germany@omega.com

United Kingdom:
ISO 9002 Certified

One Omega Drive, River Bend Technology Centre
Northbank, Irlam, Manchester
M44 5EX United Kingdom
TEL: +44 (0)161 777 6611
FAX: +44 (0)161 777 6622
Toll Free in United Kingdom: 0800-488-488
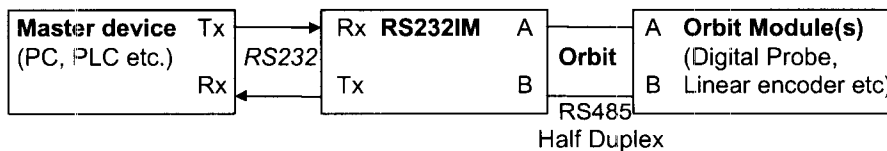e-mail: sales@omega.co.uk

---

RS232 Interface Module (RS232IM) Product Description.

The RS232IM is designed to act as a bridge between any device with a full duplex RS232 serial port and the Orbit Network.

The RS232IM receives a command header string and an Orbit command string from the 'Master' device it then re-transmits the Orbit command string to the Orbit network @ 187.5 kBaud.

The Orbit module processes the command and returns a reply (if appropriate) to the RS232IM which then adds a reply header and transmits both header and reply the to the 'Master'.

The production of the break character and the handling of the Orbit message turn around are all performed by the RS232IM.

Master device Tx		Rx **RS232IM** A		A **Orbit Module(s)**
(PC, PLC etc.)	*RS232*		**Orbit**	(Digital Probe,
Rx		Tx B		B Linear encoder etc)

RS485
Half Duplex

The command and reply header strings contain information which is used by the RS232IM to control and report on the Orbit Network.

RS232IM Commands.

The RS232IM command byte must always be the first byte sent to the RS232IM, it determines what actions are to be performed, it also dictates the meaning of the remaining bytes in the command header.

The Information below describes each type of RS232IM command and the meaning of all the bytes in the command and reply headers.
Examples are written in Quick Basic, #1 is an open Com port set to 9600 Baud, 8 data bits, no parity.

Command Type 1 - Send Orbit message, no reply

String received by RS232IM on RS232 Rx (9 way D skt pin 3).

To RS232IM:	Byte 1	Byte 2	Byte 3 - 257(max)
	Cmd byte - **0**	Cmd Length	Cmd String
	RS232IM command byte	Length of Orbit Cmd string	Orbit command string

String transmitted from RS232IM on RS232 Tx (9 way D skt pin 2).

From RS232IM:	NO REPLY

Command Type 1 - Example - Send OrbitRst command.

OrbitString$="R"+CHR$(0)
msg$ = CHR$(0)+CHR$(LEN(OrbitString$))+ OrbitString$
PRINT #1, msg$;

Command Type 2 - Send Orbit message, wait for reply

String received by RS232IM on RS232 Rx (9 way D skt pin 3).

To RS232IM:	Byte 1	Byte 2	Byte 3	Byte 4- 258(max)
	Cmd byte - **2**	Reply Length	Cmd Length	Cmd String
	RS232IM command byte	Length of Orbit Reply string	Length of Orbit Cmd string	Orbit Cmd string

String transmitted from RS232IM on RS232 Tx (9 way D skt pin 2).

From RS232IM:	Byte 1	Byte 2	Byte 3- 257(max)
	Status byte	Byte count	Reply string
	RS232IM / Orbit Status	Number of bytes in Orbit reply	Orbit reply string

Command Type 2 - Example   Send OrbitSetaddr command to Orbit module M892780 - 36 (set module to address 1)

```
oaddr% = 1
id$ = "M892780-36"
msg$ = CHR$(2) + CHR$(2) + CHR$(13) + "S" + CHR$(oaddr%) + id$ + CHR$(0)
PRINT #1, msg$;
status$ = INPUT$(1,#1)
IF status$ = chr$(0) THEN
        count = asc(INPUT$(1,#1))
        Reply$ = INPUT$(count,#1)
END IF
```

Command Type 6 - Setup interface (RS232 and Orbit)

String received by RS232IM on RS232 Rx (9 way D skt pin 3).

To RS232IM:	Byte 1	Byte 2	Byte 3
	Cmd byte - **10**	RS232 settings	Orbit speed
	RS232IM command byte	RS232 Baud rate / handshake	Orbit Baud rate

String transmitted from RS232IM on RS232 Tx (9 way D skt pin 2).

From RS232IM:	Byte 1	Byte 2
	Status byte	Byte count, always = **0**
	RS232IM status	Number of bytes in Orbit reply

Notes:
1] If the 'RS232 settings' and 'Orbit speed' codes are valid the RS232IM will be set to the new values after the status byte and byte count have been transmitted.
2] The status byte and byte count are transmitted at the same Baud rate as the received command.

Setup Codes.

RS232 settings. (byte 2 in String received by RS232IM).

This byte is used to set the RS232 Baud rate and handshaking, the Baud rate code selects the rate at which the RS232IM will operate after the command has been acknowledged.
Hardware handshaking (CTS/RTS) can be turned on/off by setting/clearing the MS bit of the in RS232 Settings byte, hence:

If handshaking is required,   RS232 Settings byte = Baud rate code + 128
If no handshaking is required,   RS232 Settings byte = Baud rate code

<u>Command Type 6</u> - Continued.

RS232 settings. (byte 2 in String received by RS232IM).

Baud rate codes

0	RS232_DefltBaud - 9600, RS232IM is set to this rate and no handshaking at power on.
1	RS232_9600
2	RS232_19200
3	RS232_28800
4	RS232_38400
5	RS232_57600
6	RS232_115200

Orbit speed (byte 3 in String received by RS232IM).

This byte is used to set the Orbit Baud rate. Currently only 9600Baud and 187.5kBaud are available, the other codes are reserved for future expansion.

0	RS485_DefltBaud - currently 187.5k, RS232IM is set to this rate at power on
1	RS485_187500
2	RS485_9600
3	reserved for future expansion

RS232 Setup interface - status codes. (byte 1 in String transmitted from RS232IM).

0	OK - RS232 Settings byte and Orbit speed byte both valid
7	bad RS232 Settings byte
8	bad Orbit speed byte

Command Type 6 - Example Set RS232IM to RS232 9600 with handshaking and Orbit speed 187.5kB
RS232Settings$ = CHR$(1 + 128)
OrbitSpeed$ = CHR$(1)
msg$ = CHR$(10) + RS232Settings$ + OrbitSpeed$
PRINT #1, msg$;
Reply$ = INPUT$(2,#1)

<u>**Command Type 8 - Send Orbit message (wait for variable length reply)**</u>

String received by RS232IM on RS232 Rx (9 way D skt pin 3).

To RS232IM:	Byte 1	Byte 2	Byte 3 - 257
	Cmd byte - **14**	Cmd Length	Cmd String
	RS232IM command byte	Length of Orbit Cmd string	Orbit Cmd string

String transmitted from RS232IM on RS232 Tx (9 way D skt pin 2).

From RS232IM:	Byte 1	Byte 2	Byte 3- 257(max)
	Status byte	Byte count	Reply string
	RS232IM / Orbit Status	Number of bytes in Orbit reply	Orbit reply string

Command Type 8 - Example - Send OrbitReadbuffer1 command to Orbit module
oaddr% = 1
msg$ = CHR$(14) + CHR$(2) + "2" + CHR$(oaddr%)
PRINT #1, msg$;
status$ = INPUT$(1,#1)
IF status$ = chr$(0) THEN
        count = asc(INPUT$(1,#1))
        Reply$ = INPUT$(count,#1)
END IF
<u>**RS232IM Error Codes**</u>

For commands 2, 6 and 8 the RS232IM returns a status byte (byte 1 in String transmitted from RS232IM) to indicate the RS232IM / Orbit status.

0	status OK
3	RS232Rx receive timeout (RS232Rx command string too short)
7	bad RS232Settings byte
8	bad Orbit speed byte
255 (0FFh)	Orbit (RS485) receive timeout (module reply too short)
254 (0FEh)	Orbit (RS485) receive parity error
253 (0FDh)	bad checksum - RS232IM calculated checksum not equal to receivec Orbit module checksum

RS232IM Strings required for each Orbit Command
In the list below $ - represents a string of bytes and CHR$(n) - represents a received / transmitted byte of binary value n.

[OrbitSetaddr]
To RS232IM: CHR$(2),CHR$(2),CHR$(13),**"S",CHR$(oaddr%),id$,CHR$(0)**
From RS232IM:CHR$(status),CHR$(2),**"S",CHR$(prevoaddr%)**
[Total reply length = 4 bytes]

[OrbitNotify]
To RS232IM: CHR$(2),CHR$(11),CHR$(2),**"N",CHR$(0)**
From RS232IM:CHR$(status),CHR$(11),**"N",id$**[10 bytes] - if moved
[Total reply length = 13 bytes]

[OrbitIdentify]
To RS232IM: CHR$(2),CHR$(30),CHR$(2),**"I",CHR$(oaddr%)**
From RS232IM:CHR$(status),CHR$(30),**"I",id$**[10 bytes],**devtype$**[12 bytes],**ver$**[5 bytes],**stroke%**[2 bytes]
[Total reply length = 32 bytes]

[OrbitGetinfo]
To RS232IM: CHR$(2),CHR$(41),CHR$(2),**"B", CHR$(oaddr%)**
From RS232IM:CHR$(status), CHR$(41), **"B",moduletype$**[4 bytes],**hwtype$**[2 bytes],**reso%**[2 bytes],
               moduleinfo$[32 bytes]
[Total reply length = 43 bytes]

[OrbitGetstatus]
To RS232IM: CHR$(2),CHR$(4),CHR$(2),**"G", CHR$(oaddr%)**
From RS232IM:CHR$(status),CHR$(4),**"G",CHR$(errcode%),status%**[2 bytes]
[Total reply length = 6 bytes]

[OrbitStartdiff]
To RS232IM: CHR$(0),CHR$(2),**"O", CHR$(0)**
From RS232IM:nothing

[OrbitStopdiff]
To RS232IM: CHR$(0),CHR$(2),**"H", CHR$(0)**
From RS232IM:nothing

[OrbitTrigger]
To RS232IM: CHR$(0),CHR$(2),**"T", CHR$(0)**
From RS232IM:nothing

[OrbitRst]
To RS232IM: CHR$(0),CHR$(2),**"R", CHR$(0)**
From RS232IM:nothing
RS232IM Strings required for each Orbit Command - Continued

[OrbitAcquire]

To RS232IM: CHR$(2),CHR$(2),CHR$(5),**"A",CHR$(oaddr%),CHR$(rdgs%),dly%**[2 bytes]
From RS232IM:CHR$(status),CHR$(2),**"A",CHR$(oaddr%)**
[Total reply length = 4 bytes]

[OrbitClr]
To RS232IM: CHR$(2),CHR$(2),CHR$(2),**"C",CHR$(oaddr%)**
From RS232IM:CHR$(status),CHR$(2),**"C",CHR$(oaddr%)**
[Total reply length = 4 bytes]

[OrbitDifference]
To RS232IM: CHR$(2),CHR$(2),CHR$(2),**"F",CHR$(oaddr%)**
From RS232IM:CHR$(status),CHR$(2),**"F",CHR$(oaddr%)**
[Total reply length = 4 bytes]

[OrbitDirection]
To RS232IM: CHR$(2),CHR$(2),CHR$(2),**"U",CHR$(oaddr%)**
From RS232IM:CHR$(status),CHR$(2),**"U",CHR$(oaddr%)**
[Total reply length = 4 bytes]

[OrbitPreset]
To RS232IM: CHR$(2),CHR$(2),CHR$(6),**"P",CHR$(oaddr%),pstlong&**[4 bytes]
From RS232IM:CHR$(status),CHR$(2),**"P",CHR$(oaddr%)**
[Total reply length = 4 bytes]

[OrbitRead1]
To RS232IM: CHR$(2),CHR$(3),CHR$(2),**"1",CHR$(oaddr%)**
From RS232IM:CHR$(status),CHR$(3),**"1",rd%**[2 bytes]
[Total reply length = 5 bytes]

[OrbitRead2]
To RS232IM: CHR$(2),CHR$(5),CHR$(2),**"L",CHR$(oaddr%)**
From RS232IM:CHR$(status),CHR$(5),**"L",rdLong&**[4 bytes]
[Total reply length = 7 bytes]

[OrbitReaddiff1]
To RS232IM: CHR$(2),CHR$(13),CHR$(2),**"D",CHR$(oaddr%)**
From RS232IM:CHR$(status),CHR$(13),**"D",min%**[2 bytes],**max%**[2 bytes] **,sum#**[5 bytes],**num!**[3 bytes]
[Total reply length = 15 bytes]

[OrbitReaddiff2]
To RS232IM: CHR$(2),CHR$(9),CHR$(2),**"X",CHR$(oaddr%)**
From RS232IM:CHR$(status),CHR$(9),**"X",minlong&**[4 bytes],**maxlong&%**[4 bytes]
[Total reply length = 11 bytes]

[OrbitReadia]
To RS232IM: CHR$(2),CHR$(51),CHR$(2),**"E",CHR$(oaddr%)**
From RS232IM:CHR$(status),CHR$(51),**"E", 25 x 2 byte values** [50 bytes]
[Total reply length = 53 bytes]

[OrbitRefmark]
To RS232IM: CHR$(2),CHR$(2),CHR$(2),**"K",CHR$(oaddr%)**
From RS232IM:CHR$(status),CHR$(2),**"K",CHR$(oaddr%)**
[Total reply length = 4 bytes]

MADE IN USA

WARRANTY/DISCLAIMER

OMEGA ENGINEERING, INC. warrants this unit to be free of defects in materials and workmanship for a period of **13 months** from date of purchase. OMEGA's WARRANTY adds an additional one (1) month grace period to the normal **one (1) year product warranty** to cover handling and shipping time. This ensures that OMEGA's customers receive maximum coverage on each product.

If the unit malfunctions, it must be returned to the factory for evaluation. OMEGA's Customer Service Department will issue an Authorized Return (AR) number immediately upon phone or written request. Upon examination by OMEGA, if the unit is found to be defective, it will be repaired or replaced at no charge. OMEGA's WARRANTY does not apply to defects resulting from any action of the purchaser, including but not limited to mishandling, improper interfacing, operation outside of design limits, improper repair, or unauthorized modification. This WARRANTY is VOID if the unit shows evidence of having been tampered with or shows evidence of having been damaged as a result of excessive corrosion; or current, heat, moisture or vibration; improper specification; misapplication; misuse or other operating conditions outside of OMEGA's control. Components which wear are not warranted, including but not limited to contact points, fuses, and triacs.

OMEGA is pleased to offer suggestions on the use of its various products. However, OMEGA neither assumes responsibility for any omissions or errors nor assumes liability for any damages that result from the use of its products in accordance with information provided by OMEGA, either verbal or written. OMEGA warrants only that the parts manufactured by it will be as specified and free of defects. OMEGA MAKES NO OTHER WARRANTIES OR REPRESENTATIONS OF ANY KIND WHATSOEVER, EXPRESS OR IMPLIED, EXCEPT THAT OF TITLE, AND ALL IMPLIED WARRANTIES INCLUDING ANY WARRANTY OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE HEREBY DISCLAIMED. LIMITATION OF LIABILITY: The remedies of purchaser set forth herein are exclusive, and the total liability of OMEGA with respect to this order, whether based on contract, warranty, negligence, indemnification, strict liability or otherwise, shall not exceed the purchase price of the component upon which liability is based. In no event shall OMEGA be liable for consequential, incidental or special damages.

CONDITIONS: Equipment sold by OMEGA is not intended to be used, nor shall it be used: (1) as a "Basic Component" under 10 CFR 21 (NRC), used in or with any nuclear installation or activity; or (2) in medical applications or used on humans. Should any Product(s) be used in or with any nuclear installation or activity, medical application, used on humans, or misused in any way, OMEGA assumes no responsibility as set forth in our basic WARRANTY/DISCLAIMER language, and, additionally, purchaser will indemnify OMEGA and hold OMEGA harmless from any liability or damage whatsoever arising out of the use of the Product(s) in such a manner.

RETURN REQUESTS/INQUIRIES

Direct all warranty and repair requests/inquiries to the OMEGA Customer Service Department. BEFORE RETURNING ANY PRODUCT(S) TO OMEGA, PURCHASER MUST OBTAIN AN AUTHORIZED RETURN (AR) NUMBER FROM OMEGA'S CUSTOMER SERVICE DEPARTMENT (IN ORDER TO AVOID PROCESSING DELAYS). The assigned AR number should then be marked on the outside of the return package and on any correspondence.

The purchaser is responsible for shipping charges, freight, insurance and proper packaging to prevent breakage in transit.

FOR **WARRANTY** RETURNS, please have the following information available BEFORE contacting OMEGA:

1. Purchase Order number under which the product was PURCHASED,
2. Model and serial number of the product under warranty, and
3. Repair instructions and/or specific problems relative to the product.

FOR **NON-WARRANTY** REPAIRS, consult OMEGA for current repair charges. Have the following information available BEFORE contacting OMEGA:

1. Purchase Order number to cover the COST of the repair,
2. Model and serial number of the product, and
3. Repair instructions and/or specific problems relative to the product.

OMEGA's policy is to make running changes, not model changes, whenever an improvement is possible. This affords our customers the latest in technology and engineering.

OMEGA is a registered trademark of OMEGA ENGINEERING, INC.