INSTRUCTION MANUAL

700171N

485 OPTIONS

MICROPROCESSOR BASED
Remote Terminal with RS485 (RT4, RT4jr.), RS485 Option,
RS232 to RS485 Isolated Converter (S485),
USB to RS485 Isolated Converter (U485),
and ENBUS Protocol

WIRING DIAGRAMS
See Section 7.4

ENTRON Controls, LLC.
1402 S. Batesville Road
Greer, South Carolina 29650
(864) 416-0190
FAX: (864) 416-0195
www.entroncontrols.com

RWMA® MEMBER

THIS PAGE IS INTENTIONALLY BLANK

ENTRON Controls, LLC.

INSTALLATION AND OPERATION MANUAL FOR:
Model Series RT4, RT4jr., RS485, U485, S485,
& ENBUS PROTOCOL

MICROPROCESSOR BASED REMOTE TERMINAL RT4/RT4jr.

! CAUTION !
READ THIS MANUAL COMPLETELY BEFORE ATTEMPTING TO INSTALL OR OPERATE THIS CONTROL

ENTRON Controls, LLC., reserves the right to alter the contents of this manual without previous notice.

ENTRON Controls, LLC.
Greer, South Carolina 29650

TABLE OF CONTENTS

TABLE OF CONTENTS (cont.)

FIGURES

TABLE OF CONTENTS (cont.)

FIGURES

TABLES

1.0 INTRODUCTION

All EN1000 Series Controls with the RS485 option can communicate with Remote Terminal RT4, RT4jr. or PC or other devices with the RS485 interface, over two-wire RS485 network (ENBUS network) as shown in Figure 1-1. This manual covers the RS485 Option, RT4, RT4jr., ENLINK and ENBUS operation. See appropriate section for the information required.

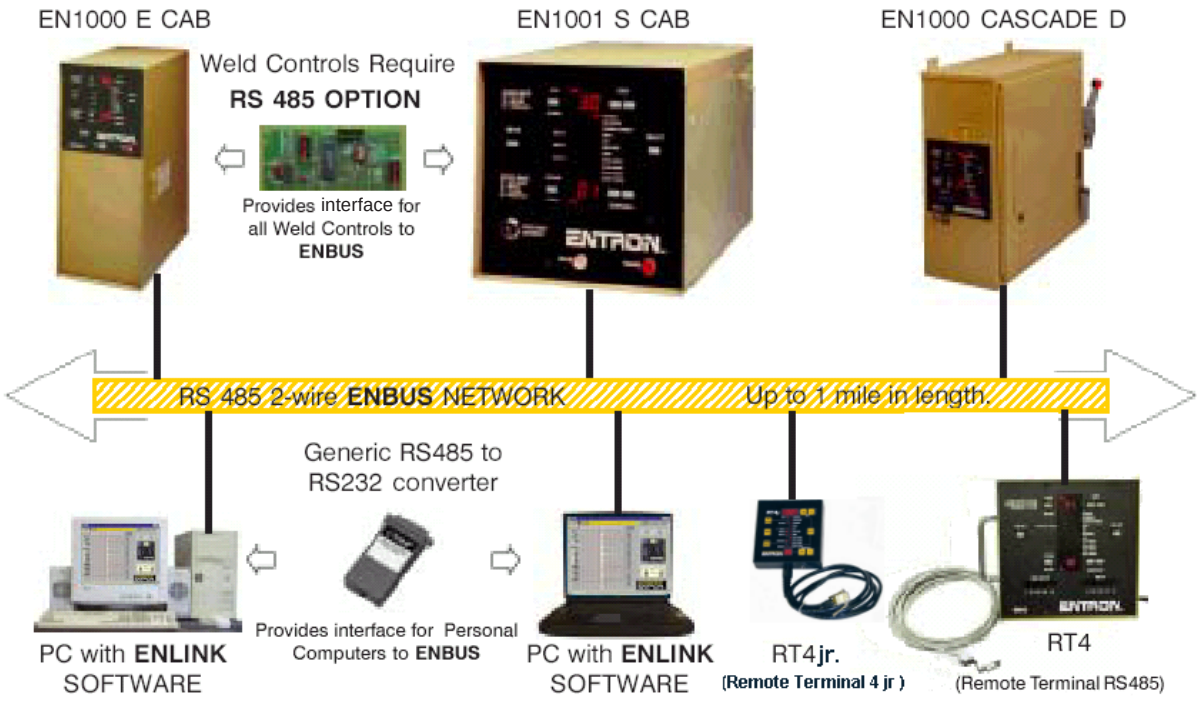


Figure 1-1. *RT4 and PC network interface*

1.1 EQUIPMENT AND SOFTWARE

1. Any EN1000 Series Control with RS485 option.
2. A 9 Pin D-Subminiature serial cable, provided by ENTRON and constructed using Appendix 7.2.
3. RT4/RT4jr. Remote Terminal.
4. PC with S485/U485 converter and communication software like ENLINK for Windows or similar (***optional*** – not required).

2.0 REMOTE TERMINAL – RT4/RT4jr.

The Remote Terminal with RS485 Option (RT4 or RT4jr.) is designed as a master device for communication with ENTRON Controls, through an RS485 two-wire network (see Figure 1-1). The Front Panel of both the RT4 and RT4jr. is similar to ENTRON EN1000 Series Welding Controls (see Section 2.1 Front Panel Layouts). Programming the RT4/RT4jr. is similar to programming any EN1000, EN1001, EN1000 Cascade, or EN1001 Cascade Series Control.

The RT4 is a more ruggedized version with a 14 Ga. steel cabinet and a steel Dial Plate with mechanical switches, the same size as any standard ENTRON Control. The RT4jr. is built in a smaller, plastic cabinet with an aluminum Dial Plate and membrane-style switch pad.

2.0 REMOTE TERMINAL – RT4/RT4jr. (cont.)

RT4 – OBSOLETE – SEE RT4jr.

The RT4 has a 9 pin female connector on the rear of the cabinet (opposite the Dial Plate). This connects, via the J485-J485 Harness Assembly, to any control provided with the RS485 Option (see Section 7.2 Field Construction of J485-J485 Harness Assembly). There is a 120 VAC line cord permanently attached to the cabinet (see Wiring Diagram 421406).

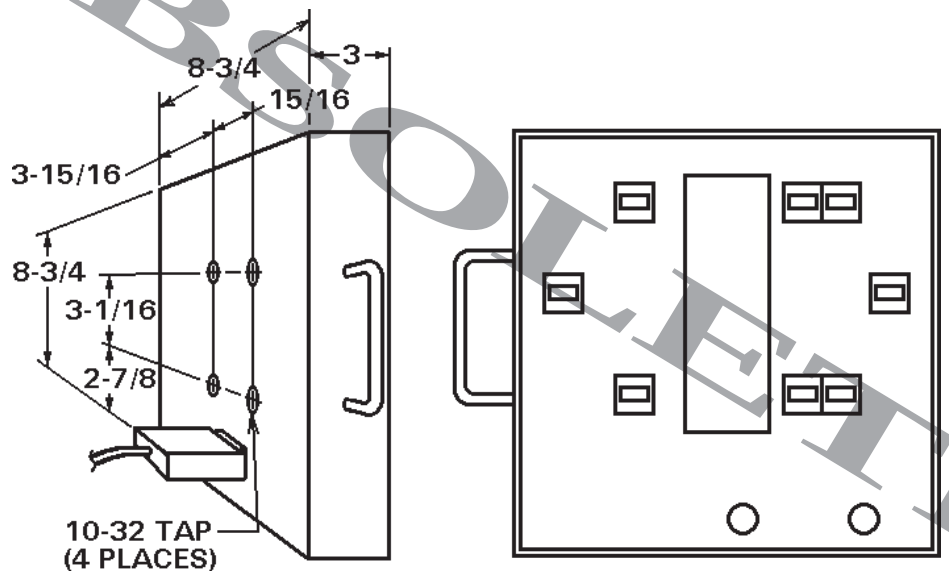


Figure 2-1. *RT4 dimensions*

RT4jr.

The RT4jr. has a six foot long cord with a female 9 pin connector which is used to connect the RT4jr., via the J485-J485 Harness Assembly, to any control equipped with an RS485 Option (see Section 7.2 Field Construction of J485-J485 Harness Assembly). The RT4jr. is powered by a wall mounted 120 VAC transformer which connects to the RT4jr. through a 3 pin female miniature connector (see Wiring Diagram 421472).

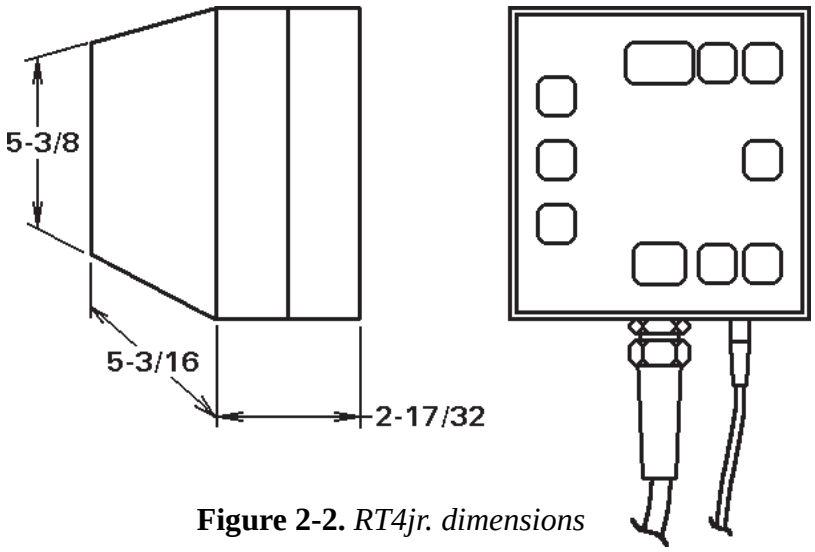


Figure 2-2. *RT4jr. dimensions*

2.1 FRONT PANEL LAYOUTS

2.1.1 REMOTE TERMINAL RT4 FRONT PANEL LAYOUT – OBS - SEE RT4jr.

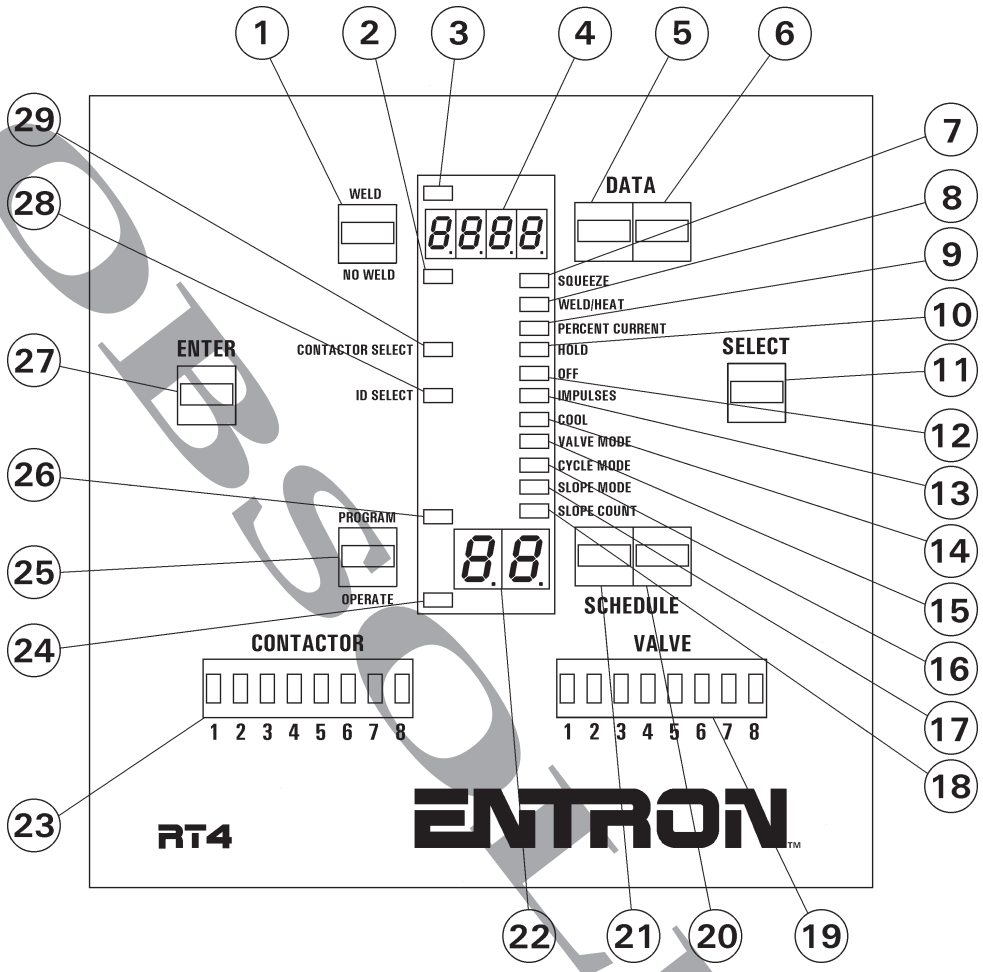


Figure 2-3. *RT4 Front Panel layout*

1 – WELD/NO WELD push button
2 – NO WELD mode indicator LED
3 – WELD mode indicator LED
4 – DATA display
5 – DATA 10s push button
6 – DATA 1s push button
7 – SQUEEZE indicator LED
8 – WELD/HEAT indicator LED
9 – PERCENT CURRENT indicator LED
10 – HOLD indicator LED
11 – SELECT push button
12 – OFF indicator LED
13 – IMPULSES indicator LED
14 – COOL indicator LED
15 – VALVE MODE indicator LED
16 – CYCLE MODE indicator LED
17 – SLOPE MODE indicator LED
18 – SLOPE COUNT indicator LED
19 – VALVE indicator display
20 – SCHEDULE 1s push button
21 – SCHEDULE 10s push button
22 – SCHEDULE display
23 – CONTACTOR indicator display
24 – OPERATE mode indicator LED
25 – PROGRAM/OPERATE push button
26 – PROGRAM mode indicator LED
27 – ENTER push button
28 – ID SELECT indicator LED *for Error Scanning*
29 – CONTACTOR SELECT indicator LED (CASCADE ONLY)

2.1 FRONT PANEL LAYOUTS (cont.)

2.1.2 REMOTE TERMINAL RT4jr. FRONT PANEL LAYOUT

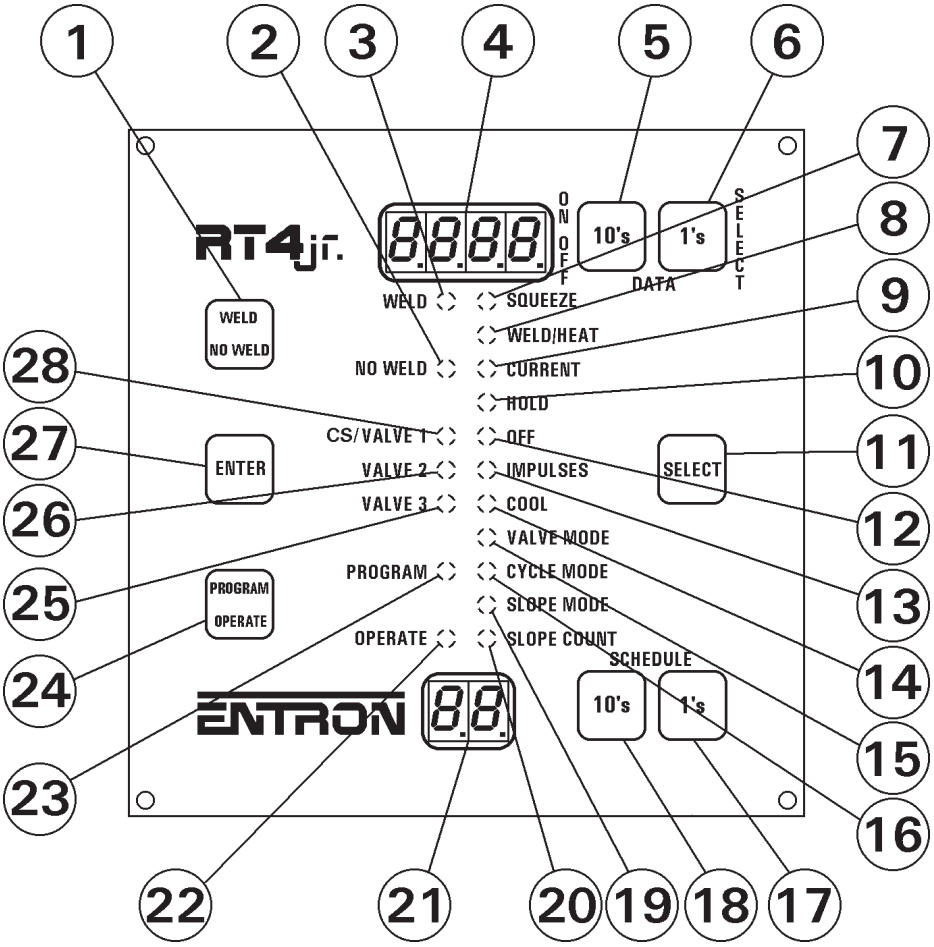


Figure 2-4. *RT4jr. Front Panel layout*

1 – WELD/NO WELD push button
2 – NO WELD mode indicator LED
3 – WELD mode indicator LED
4 – DATA display
5 – DATA 10s push button
6 – DATA 1s push button
7 – SQUEEZE indicator LED
8 – WELD/HEAT indicator LED
9 – PERCENT CURRENT indicator LED
10 – HOLD indicator LED
11 – SELECT push button
12 – OFF indicator LED
13 – IMPULSES indicator LED
14 – COOL indicator LED
15 – VALVE MODE indicator LED
16 – CYCLE MODE indicator LED
17 – SCHEDULE 1s push button
18 – SCHEDULE 10s push button
19 – SLOPE MODE indicator LED
20 – SLOPE COUNT indicator LED
21 – SCHEDULE display
22 – OPERATE mode indicator LED
23 – PROGRAM mode indicator LED
24 – PROGRAM/OPERATE push button
25 – VALVE 3 indicator LED
26 – VALVE 2 indicator LED
27 – ENTER push button
28 – CONTACTOR SELECT/VALVE 1 indicator LED (CASCADE ONLY)

2.2 CONNECTION

To connect an RT4 as a Remote Terminal to any EN1000 Series Control with RS485 Option, a J485-J485 cable is necessary. This cable may be constructed using instructions given in Appendix 7.2. The RT4jr. has an integral 6 foot long cable.

The optimal configuration for the RS485 two-wire network is the daisy-chain connection from node to node, as shown in Figure 2-5. Stubs will be as short as possible with this network topology. Minimizing the stub length minimizes transmission-line problems. Furthermore, termination on only two end points prevents overloading RS485 drivers and reduces transmission-line problems due to unterminated ends. All network wires should travel as far as possible from other high voltage wires connected to solenoid valves, welding transformers, or the AC line.

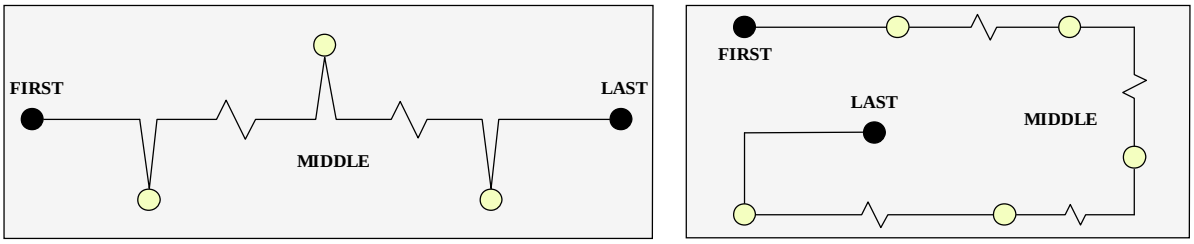


Figure 2-5. *Examples of daisy-chain topology*

2.3 SET TERMINATION OPTIONS ON THE RS485 DRIVERS

The RS485 two-wire network requires termination because of fast transitions, high data rates, or long cables. The purpose of termination is to prevent reflections or similar transmission-line problems. The RS485 bus can be terminated using a dip switch on the RS485 Driver Boards. The dip switch on RS485 Driver Board of the RT4/RT4jr. is already set at the factory to default as a FIRST location, as shown on RT4 Wiring Diagram 421406, RT4jr. Wiring Diagram 421472, or Interconnect Wiring Diagram 421407 for RS485 Drivers. The dip switch on the RS485 Driver Board of the weld controls is already set at the factory to default as LAST location. When weld controls are used in other (first or middle) positions, the dip switches must be changed from factory settings. The termination options are explained in Appendix 7.1.

2.4 PROGRAMMING EN1000 SERIES CONTROLS USING RT4/RT4jr.

The RT4 and RT4jr. may be used for programming or for error monitoring of EN1000 Series Weld Controls.

2.4.1 PROGRAM SLAVE AND HOST IDENTIFICATION NUMBER

1. CONTROL – Program the IDENTIFICATION NUMBER of the control. Use the **I.d.** parameter in the EXTENDED FUNCTIONS. This **I.d.** number could be any number from **01** to **64**. IDENTIFICATION NUMBER **I.d.**=**00** is reserved for RS232 serial communication between a PC and a welding control. More than one control with the same ID NUMBER on the same network is not allowed as the network will not operate properly.

2.4.1 PROGRAM SLAVE AND HOST IDENTIFICATION NUMBER (cont.)

2. REMOTE TERMINAL RT4/RT4jr. – Address the weld control by programming the control ID NUMBER on the RT4/RT4jr., using **I.d.** parameter in EXTENDED FUNCTIONS. This **I.d.** must match the **I.d.** on the welding control the RT4/RT4jr. needs to address. The range is the same, from **01** to **64**. ID NUMBER **I.d.**=**00** is reserved for broadcasting from the RT4 or RT4jr. to all controls.

3. MULTIPLE TERMINALS – If more than one RT4/RT4jr. exists on the same RS485 two-wire network, program the host ID NUMBER of RT4/RT4jr. by using the **H.o.** parameter in the EXTENDED FUNCTIONS. It may be any number from **65** to **95**. Number **96** is reserved for RS232 serial communications between PC and RT4/RT4jr. More than one RT4/RT4jr. with the same **H.o.** on the same network is not allowed.

2.4.2 SHOW I.d. ON ALL CONTROLS FROM REMOTE TERMINAL RT4/RT4jr.

To display the ID NUMBER **I.d.** on all controls connected to the Remote Terminal RT4/RT4jr., press the ENTER push button while the RT4/RT4jr. is in OPERATE mode. All controls on the RS485 network will show the **I.d.** number on their displays, without sending any response back to the RT4/RT4jr.

NOTICE
The RT4/RT4jr. at this time may have ENBUS conflicts with other master devices such as PCs running ENLINK. Future revisions of RT4/RT4jr. software are planned to better this combination; this command may be removed.

2.4.3 START COMMUNICATION BETWEEN RT4/RT4jr. AND CONTROL WITH RS485 OPTION

1. Select a target control by programming the same **I.d.** number on the RT4/RT4jr. and target control, as described in Section 2.4.1.
2. Press the PROGRAM/OPERATE push button on the RT4/RT4jr. to switch from OPERATE to PROGRAM mode. At that time the RT4/RT4jr. will try to communicate with the control. If that control exists on the network, the RT4/RT4jr. will display SCHEDULE Sequence or EXTENDED FUNCTION data from desired control.

If the control with the desired **I.d.** does not exist on the network, or if the control is turned off or busy, the RT4/RT4jr. will display [—] on the DATA display. By selecting EXTENDED FUNCTIONS **EF**, the **I.d.** and **H.o.** parameters can be accessed and programmed. Any time the mode is changed from PROGRAM to OPERATE, the RT4/RT4jr. will try to find that control. If communication is lost for any reason, or if the RT4/RT4jr. cannot find that control, ERROR CODE **E.r.**=**40** will be shown on the display. To clear ERROR CODE, press any push button.

NOTICE
Weld controls will not communicate during weld sequences.

2.4.4 COMMUNICATION BETWEEN RT4/RT4jr. AND MULTIPLE CONTROLS WITH RS485 OPTION

To change the target control, program the new **I.d.** number on the RT4/RT4jr. to match the **I.d.** number of the new target control, as mentioned in Section 2.4.1. At the same time, the RT4/RT4jr. will try to communicate with the control. If that control exists on the network, the RT4/RT4jr. will display the SCHEDULE sequence or EXTENDED FUNCTION data from the desired control; if not, ERROR CODE **E.r.**=**40** will be shown on display.

2.4.5 TOGGLE WELD/NO WELD

Press the WELD/NO WELD push button on the RT4/RT4jr. and the target control will toggle from WELD to NO WELD mode.

2.4.6 CHANGE THE SCHEDULE SEQUENCE

Press or hold the SELECT push button on the RT4/RT4jr. The target control will display the corresponding control SCHEDULE parameters on the DATA displays, and the corresponding LED will illuminate.

2.4.7 CHANGE THE SCHEDULE

1. Press the right SCHEDULE push button on the RT4/RT4jr. and the schedule number will be incremented by 1, or hold the same push button to decrement by 1.
2. Press the left SCHEDULE push button on the RT4/RT4jr. and the schedule number will be incremented by 10, or hold the same push button to decrement by 10.

2.4.8 CHANGE AND SAVE DATA IN PROGRAM MODE

1. In PROGRAM mode, press the right DATA push button on the RT4/RT4jr. and the parameter on the DATA display will be incremented by 1, or hold the same push button to decrement by 1 (or increment by 10 for 4-digit data).
2. In PROGRAM mode, press the left DATA push button on the RT4/RT4jr. and the parameter on the DATA display will be incremented by 10 (or increment by 100 for 4-digit data), or hold the same push button to decrement by 10 (or increment by 1000 for 4-digit data).
3. Press the ENTER push button to write the new data to the EEPROM. Previous data will be overwritten, only if ENTER is pressed.

2.4.9 OTHER FRONT PANEL SHORTCUTS IN PROGRAM MODE

1. In the PROGRAM mode, press and hold ENTER and press the PROGRAM push button on the RT4/RT4jr. to ***clear*** current schedule data.
2. In the PROGRAM mode, press and hold ENTER and press the right SCHEDULE push button on the RT4/RT4jr. to ***copy*** the current schedule to the next schedule up.
3. In the PROGRAM mode, press and hold ENTER and press the left SCHEDULE push button on the RT4/RT4jr. to ***copy*** the current schedule to that schedule number plus 10.
4. In the PROGRAM mode, press and hold ENTER and press both DATA push buttons on the RT4/RT4jr. to **reset** the control.

2.5 ERROR MONITORING USING THE RT4/RT4jr.

The two-wire ENBUS may be constructed with up to 64 different controls, up to 32 Remote Terminals RT4/RT4jr., and up to 32 other devices with an RS485 interface. Working groups can be used to break up ENBUS structures into smaller sub-sets.

The working groups are determined by programming an additional set of parameters, **IdEn** on the RT4/RT4jr. This set of parameters, ID ENABLE FOR ERROR MONITORING, is a two-dimensional array of 2x65 elements. The first array (index) are control ID NUMBERS from **00.** to **64.** shown on the SCHEDULE display with a dot after the last digit. The second array (data) is shown on the DATA display. Program **00** to disable or **01** to enable monitoring the control corresponding to the indicated ID NUMBER. Each RT4/RT4jr. can monitor any control from **I.d.**=**01** to **I.d.**=**64**, up to 64 controls.

Error monitoring, described in this section, is available in RT4/RT4jr. with PROM firmware version 619037-001**D** and can be used with EN1001 Weld Controls with PROM firmware version 619016-002**E** or later. (For all previous software revisions, working groups were defined by programming **I.d.** and **H.o.** numbers.)

2.5.1 PROGRAMMING WORKING GROUPS ON RT4/RT4jr.

To enable or disable a desired control with programmed ID NUMBER for Error Monitoring (Scanning) or to program the **IdEn** on RT4/RT4jr.:

1. Put the RT4/RT4jr. in PROGRAM mode.
2. Click SELECT until the DATA display shows **EF**.
3. Click SELECT one more time until the DATA display shows **IdEn**. At this point ID SELECT indicator LED will illuminate.
4. Click the right SCHEDULE push button to increase by 1 (push and hold to decrease by 1) or left SCHEDULE push button to increase by 10 (push and hold to decrease by 10) the ID INDEX NUMBER.

2.5.1 PROGRAMMING WORKING GROUPS ON RT4/RT4jr. (cont.)

5. Use the DATA push buttons to program the desired values, where:
   01.=**00** – Control with **I.d.**=**01** is ***disabled*** for Error Monitoring on this RT4/RT4jr.
   01.=**01** – Control with **I.d.**=**01** is ***enabled*** for Error Monitoring on this RT4/RT4jr.
6. Press ENTER.
7. Repeat steps 5 and 6 for all controls with ID NUMBER from **I.d.**=**01** to **I.d.**=**64**.
8. Put the RT4/RT4jr. back in OPERATE mode.

2.5.2 START AND STOP ERROR MONITORING ON RT4/RT4jr.

Error Scanning or Monitoring is possible only if the RT4/RT4jr. is in OPERATE mode.
START – To begin Error Scanning immediately, press both DATA push buttons on RT4/RT4jr.
STOP – To end scanning process, press and hold any other single push button. Auto-scanning will begin 30 seconds after last push button is pressed on the RT4/RT4jr.

2.5.3 ERROR SCANNING STATUS

After Error Scanning begins, the RT4/RT4jr. will scan for errors on all enabled controls. The SCHEDULE display will show the ***I.d. number*** of the scanned control, the DATA display will display the ***status*** of the same scanned control. Status may be one of the following messages:

1. **noE.r.** – Scanned control is operating normally without any errors.
2. **E.r.XX** – Scanned control has an error, where ***xx*** is ERROR CODE.
   For example: **E.r.01** – Scanned control has ERROR CODE **E.r.**=**01** which means TLS open or over temperature error.
3. **E.r...** – Error on scanned control is cleared.
4. [——] – Scanned control does not respond.

2.5.4 MONITORING CURRENT AT THE END OF WELD

If any of the scanned controls has PROCESS OUTPUT **P.O.** =**22** or **23** enabled, see Section 6.2 for more details. During Error Scanning, the RT4/RT4jr. will show the current on the DATA display and one of the following messages: **C.U...** or **H. ı.** or **L.o.**, depending on current value at the end of completed weld sequence on that control. The RT4/RT4jr. will stop scanning and continue to display the current and corresponding message until any push button is pressed. After that, scanning will continue.

3.0 EN1000 SERIES WELD CONTROLS WITH RS485 OPTION

As mentioned in the first section, EN1000 Series Controls with RS485 Option are slaves on the two-wire RS485 network. Only the RT4/RT4jr. can be a host or master. A PC with RS232/ RS485 converter could also be a host. Control Board connection to the RS485 driver board is shown on Figure 3-1.

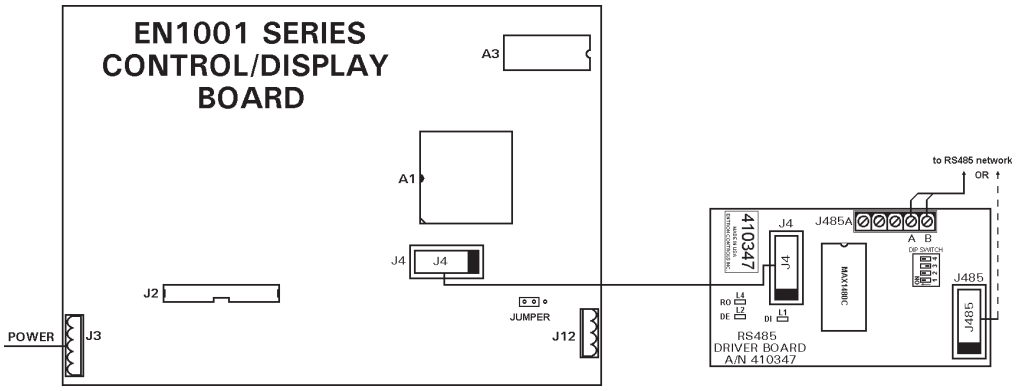


Figure 3-1. *Control Board connection to RS485 driver board*

3.1 PROGRAMMING OF THE CONTROL WITH RS485 OPTION

To allow more than one EN1000 Series Control with RS485 Option on the same two-wire RS485 network, a new ***I.d.*** must be assigned to any new control on the network. To program the ID NUMBER of the control, use the ***I.d.*** parameter in the EXTENDED FUNCTIONS. This ***I.d.*** could be any number from ***01*** to ***64***. ID NUMBER ***I.d.=00*** is reserved for RS232 serial communication between PC and welding control. More than one control with the same ***I.d.*** on the same network is not allowed. Since up to 64 different controls might be on the network at the same time, there is the option of assigning different working groups with different RT4s. More information about Error Scanning and working groups can be found in Section 2.5.

3.2 RS485 DRIVER BOARD

An ENTRON RS485 interface (Figure 3-2) is a half duplex device allowing both transmission and receiving, but not at the same time.

Figure 3-2. *RS485 driver board*

All three LED diodes L1, L2, and L4 will illuminate only during communication on the RS485 bus, otherwise they should be off. If any of the three LEDs are steadily ON, this can indicate a bad connection or wrong termination or bias switches are set wrong.

A simplified RS485 interface component layout is shown in Figure 3-3.

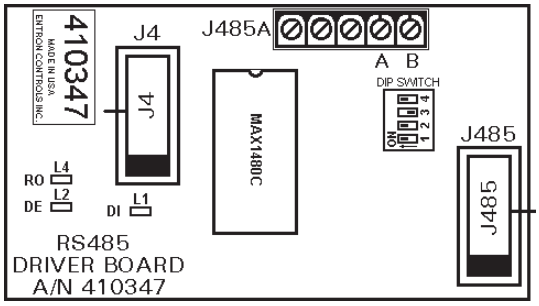


Figure 3-3. *RS485 driver board layout*

The termination and sense options on RS485 driver boards must be selected properly using the dip switch, as described in Appendix 7.1.

4.0 PERSONAL COMPUTER WITH RS485 OPTION AS HOST

Besides using an RT4/RT4jr. as a host (i.e., master) on the two-wire RS485 network, personal computers with an RS485/RS232 converter and communication software such as ENLINK or similar may be used.

If ENLINK is not used, the user must provide communication software for communication with all EN1000 Series ENTRON Controls with the RS485 Option. Such software must be based on ENBUS protocol. This protocol is described in Section 5.0.

A PC or laptop connection to the ENBUS network is shown on Figure 4-1.

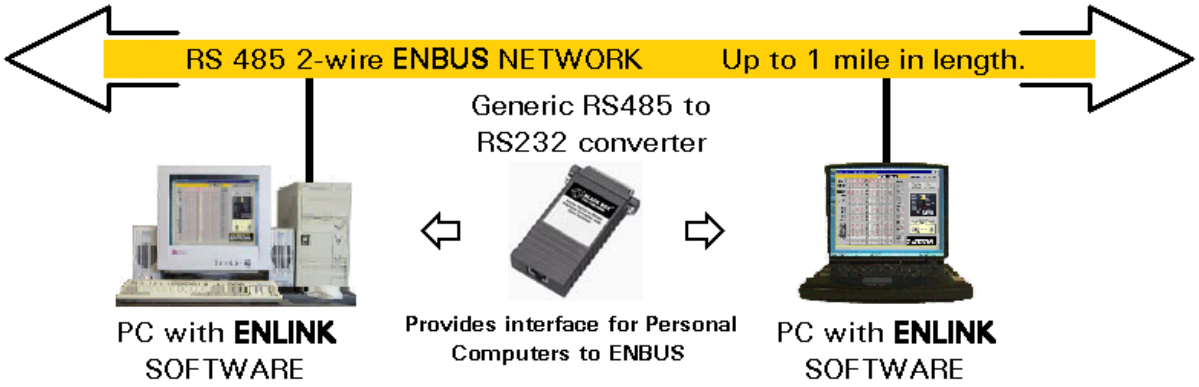


Figure 4-1. *PC or laptop connection to the ENBUS network*

Since up to 32 different computers or similar devices with RS485 Option might be on the same two-wire RS485 network with EN1000 Series Controls, they must have different addresses (i.e., Host Identification Numbers). The termination options on the RS485 interface must be selected properly (see Appendix 7.1).

For customer convenience, two isolated RS485 converters are available from ENTRON.

The two devices are produced by B&B Electronics. The manuals for these devices are included in Sections 4.1 and 4.2. They were written by B&B Electronics. Updated copies may be found on their web page at www.bb-elec.com.

The ENTRON Model S485 incorporates the B&B model 485OTLED. The ENTRON Model U485 incorporates the B&B model USOPTL4.

When these converters are purchased from ENTRON, a male 9 pin D-Subminiature cable assembly is connected to the RS485 outputs that connects directly to the J485 connector on weld controls equipped with RS485 Options.

ENTRON configures all jumpers and dip switches to work with ENBUS and ENLINK.

The U485 ships complete with a 6 foot USB cable.

The S485 ships complete with 2 cables to adapt to either 9 or 25 pin RS232 serial connectors.

4.1 S485 – RS232 TO RS485 OPTICALLY ISOLATED CONVERTER CE

Model 485OTLED

Introduction

The Model 485OTLED is our most feature-packed RS-232 to RS-422/485 converter. It converts unbalanced, full or half-duplex RS-232 signals to balanced, full or half-duplex RS-422 or RS-485 signals at baud rates up to 115.2 kbps. In addition the unit optically isolates and surge suppresses the RS-422/485 lines. The driver uses automatic SD (send data) or RTS (handshake) control, or can be configured as always enabled for use in RS-422 systems.

A dipswitch selects communication features on the 485OTLED. Two LEDs show data traffic in either direction. Only one power supply is necessary, located on the RS-232 side. The isolated power and ground on the RS-422/485 side are generated internally.

Connections are made through a DB25 female connector on the RS-232 side and terminal blocks on the RS-422/485 side. All terminal blocks, dipswitches, and jumpers are located inside the hood and are reachable through the access panel, which slides closed to protect the connections.

Description

The RS-232 port has a female DB25 connector with pins 2 (TD), 3 (RD), and 7 (Signal Ground) supported, pins 4 (RTS) and 5 (CTS) are tied together, and pins 6 (DSR), 8 (CD), and 20 (DTR), are also tied together. The 485OTLED has two LEDs: A Transmit Data LED to show when data goes from the RS-232 side to the RS-485 side, and a Receive Data LED showing data going from the RS-485 side to the RS-232 side. These are very useful for determining if data is getting through the converter. The RS-485 terminal blocks support Transmit Data (A) and (B), Receive Data (A) and (B), and Signal Ground. A single supply voltage of 10 to 30 VDC on the RS-232 side powers the unit. An 8 position dipswitch allows the selection of baud rate, 2-wire or 4-wire mode, echo on or off, and termination in or out.

RS-422/485 Driver Control

The 485OTLED uses either RTS Control or Send Data Control to enable the RS-485 driver. This option is user selectable by setting push-on jumpers located next to the terminal blocks. RTS controls the driver using toggling Request to Send (pin 4) of the RS-232 side. Raising RTS enables the driver and lowering RTS disables it. Automatic Send Data Control recognizes the first bit of data from the RS-232 side, enables the transmitter and disables the receiver. After the last bit of data is sent from the RS-232 side, the timeout waits one character length then disables the transmitter and enables the receiver. The timeout can be selected with dipswitches or by changing the value of R21 (see Table 4-1). The timeout is preset at the factory for 9600 baud. Removing both sets of jumpers completely will constantly enable the RS-485 driver. This makes the 485OTLED behave like an RS-422 converter. See Table 4-2 for standard communications settings.

Termination and Echo Options

Termination resistance can be selected with switch 8 for high baud rates and long cable distances. See B&B Electronics RS-422/485 Application Note for more information (available on the web site). The 4-wire/2-wire switches (6 and 7) are turned off for 4-wire mode and on for 2-wire mode. When they are set to ON, they connect TD(A) to RD(A) and TD(B) to RD(B) internally. Switch 5 is controls Echo on and off. When switch 5 is in the off position the receiver is constantly enabled. Placing the switch in the on position allows the driver **or** the receiver to be enabled at any time. In 2-wire mode it is recommended that switch 5 be turned on to prevent data being sent out from being "echoed" back through the receiver. Up to 32 receivers can be driven by any one RS-485 driver, allowing you to put together large systems with many drop points. Using an RS-422/485 repeater manufactured by B&B Electronics can increase the number of receivers.

Table 4-1. *Baud Rate Selection*

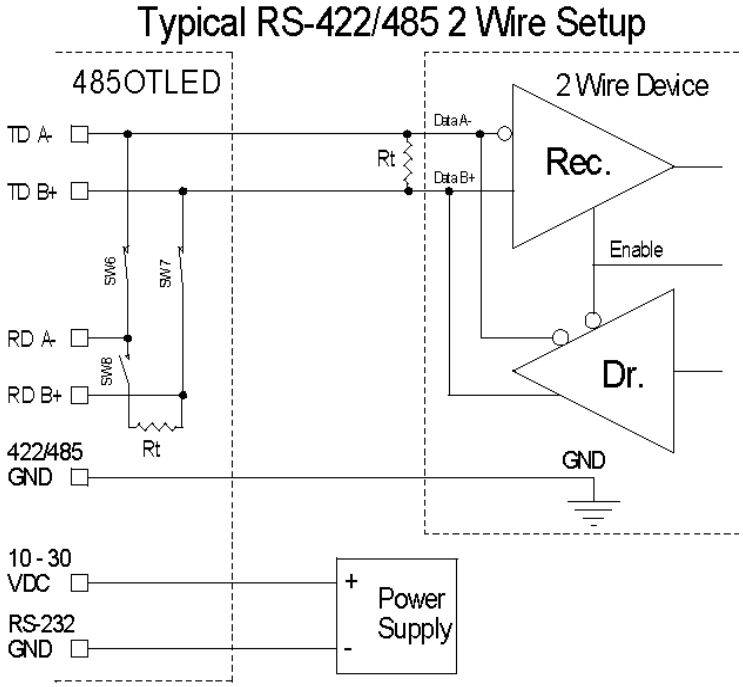
Baud	Switch 1	Switch 2	Switch 3	Switch 4	R21	Time out (ms)	
1200	OFF	OFF	OFF	OFF	820kΩ	9.0	
2400	OFF	OFF	OFF	OFF	410kΩ	4.5	
4800	ON	OFF	OFF	OFF	Not Used	2.2	← **FOR USE WITH ENLINK**
9600	OFF	ON	OFF	OFF	Not Used	1.1	
19200	OFF	OFF	ON	OFF	Not Used	0.6	
38400	OFF	OFF	OFF	ON	Not Used	0.3	
57600	OFF	OFF	ON	ON	Not Used	0.2	
115200	OFF	OFF	OFF	OFF	8.2kΩ	0.1	

4.1 S485 (cont.)

Table 4-2. *Standard Communications Settings*

Communication Mode	JP1	Switch 5 Echo	Switch 6 4W/2W	Switch 7 4W/2W
RS-422 Mode (full duplex)	Neither	OFF	OFF	OFF
RS-485 4-Wire Mode (full-duplex)	RTS or SD	OFF	OFF	OFF
RS-485 2-Wire Mode (half-duplex)	RTS or SD	ON	ON	ON

← FOR USE WITH ENLINK



Specifications

Dimensions:	2.7 X 5.1 X 0.9 in. (7.0 X 13.0 X 24.0 cm)
Temperature Range:	0°C to 70°C
Supply Voltage:	+10 to 30 VDC @ 95 mA maximum
Data Rates:	up to 115.2 kbs
Connectors:	DB25 female for RS-232 Terminal blocks for RS-422/ 485 and power
Isolation:	2000 VAC Optical Isolation of Data Signals and Ground
Surge Suppression:	7.5V, bi-directional avalanche breakdown device. 500W peak power dissipation Clamping time <1 picosecond (theoretical)

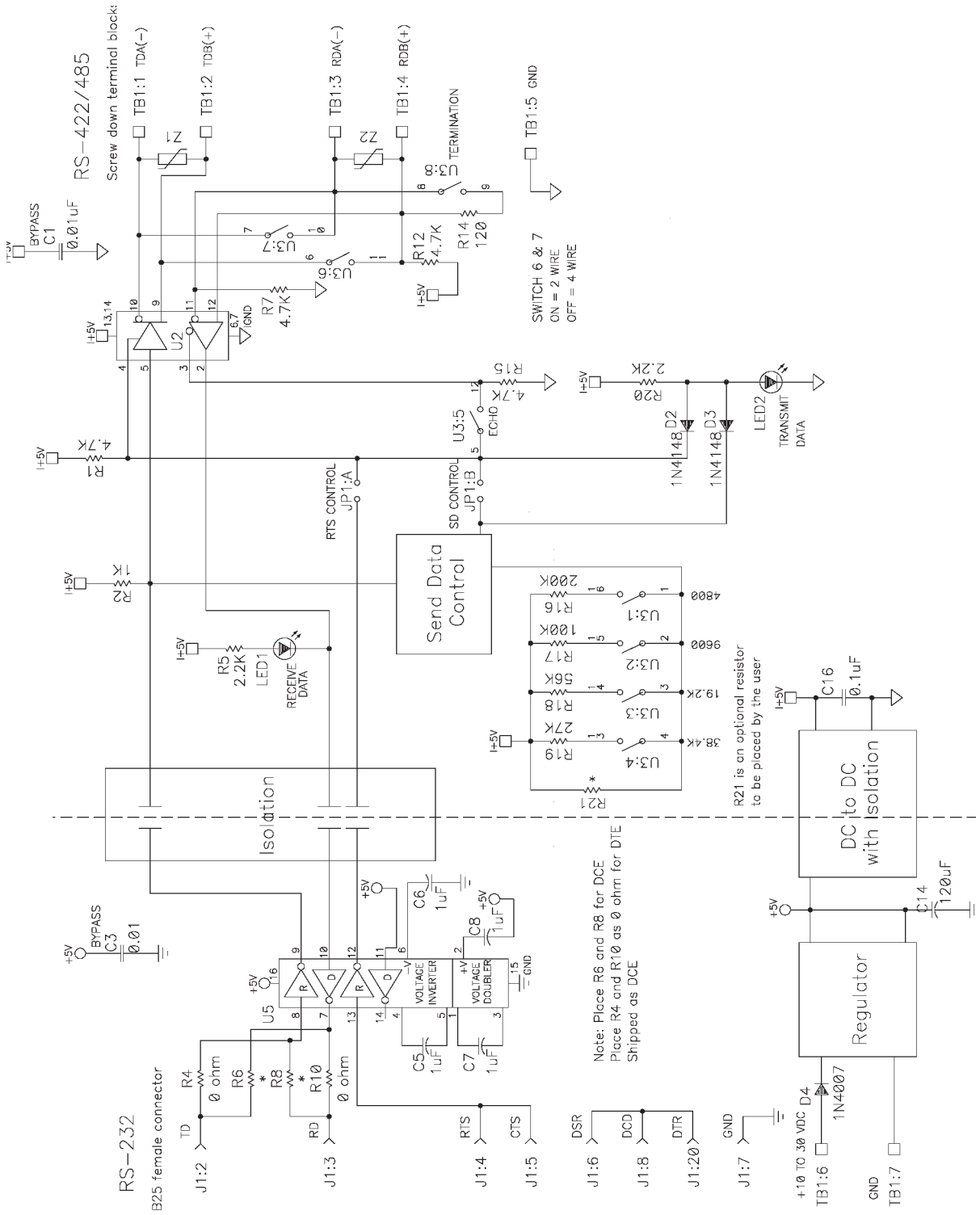
DECLARATION OF CONFORMITY

Manufacturer's Name:	B&B Electronics Manufacturing Company
Manufacturer's Address:	P.O. Box 1040 707 Dayton Road Ottawa, IL 61350 USA
Model Numbers:	485OTLED
Description:	RS-232/485 Optically Isolated Converter
Type:	Light industrial ITE equipment
Application of Council Directive:	89/336/EEC
Standards:	EN 50082-1 (IEC 801-2, IEC 801-3, IEC 801-4) EN 50081-1 (EN 55022, IEC 1000-4-2) EN 61000 (-4-2, -4-3, -4-4, -4-5, -4-6, -4-8, -4-11) ENV 50204 EN 55024

CE

William H. Franklin III, Director of Engineering

4.1 S485 (cont.)


RS-422/485
Screw down terminal block:
TB1:1 TDA(-)
TB1:2 TDB(+)
TB1:3 RDA(-)
TB1:4 RDB(+)
TB1:5 GND
TERMINATION
SWITCH 6 & 7
ON = 2 WIRE
OFF = 4 WIRE
Send Data Control
RTS CONTROL
SD CONTROL
ECHO
RECEIVE DATA
TRANSMIT DATA
Isolation
DC to DC with Isolation
Regulator
R21 is an optional resistor to be placed by the user
Note: Place R6 and R8 for DCE
Place R4 and R10 as 0 ohm for DTE
Shipped as DCE
RS-232
B25 female connector
+10 TO 30 VDC
1N4007

4.2 U485 – USB TO RS485 OPTICALLY ISOLATED CONVERTER CE

Model USOPTL4

The **USOPTL4** is a USB (Universal Serial Bus) to one port RS-422/485 converter. Supporting 2-wire RS-485 or a 4-wire RS422/485 communications, this device is great for any application that requires long range or multi-drop capabilities. This model uses pluggable terminal blocks on the RS-422/485 side and has a pair of LEDs that indicate data being transmitted or received. Model USOPTL4 includes special circuitry that adds 2000 volts isolation protection against ground loops and voltage spikes. This product draws power from the USB port so no power supply is required.

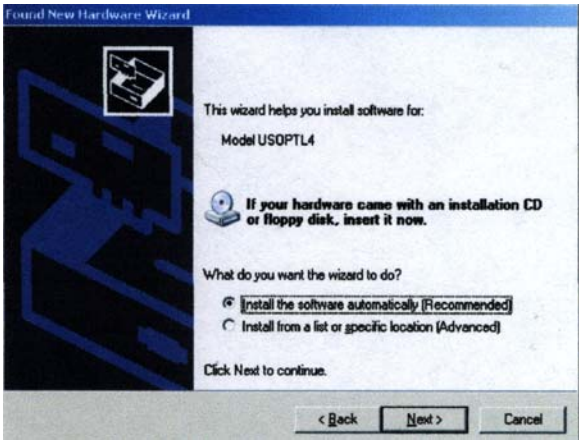
Simply plug the converter into an available USB port on your computer or USB hub and the device will show up as an additional COM port in the Windows Device Manager. Configured as an additional COM port, the converter is now compatible with your Windows applications.

NOTICE

ENTRON will attempt to keep the most current Installation Drivers for the B&B USOPTL4 on the ENLINK CD. The most current Drivers can also be found on the B&B web site at www.bb-elec.com.

Installation for Windows

#1. Plug the USOPTL4 into an available USB port on your computer or connected hub. The screen above appears, telling you that there is a new device plugged into the USB bus. Click on *No, not this time*, then the **Next>** button.



#2. The screen above appears. Make sure *Install the software automatically* is selected. Then select the **Next>** button.

#3. The screen above will appear. Click the **Finish** button to complete the installation.

#4. You will also need to continue to install the serial port the same as installing the converter. Click the **Next>** button followed by the **Finish** button. It will take a couple seconds for the serial port to be installed.

4.2 U485 – MODEL USOPTL4 (cont.)

RS-485 Control

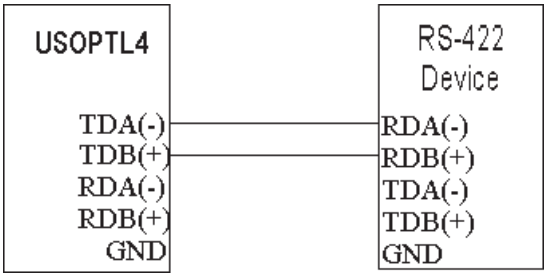
No special software is required to control the RS-485 receiver or transmit line driver. The driver is automatically enabled during each byte transmitted in RS485 mode. The transmitter is always enabled in RS-422 mode. The receiver is tri-stated during each byte transmitted in the echo off mode. The receiver is always enabled in the echo on mode. There are 4.7k ohm pull up/pull down resistors on the RDA and RDB lines. A termination resistor may be added to R16 if needed. See B&B's RS-422/RS-485 Application Note (available at www.bb-elec.com) for more information on termination and DC biasing of an RS-485 network.

Dip Switch Setup

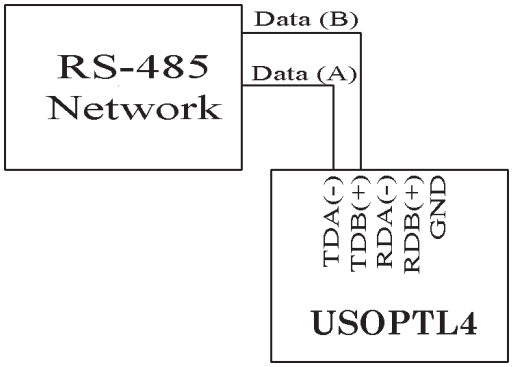
The dip switches allow the module to be configured for two-wire or four-wire, RS-422 or RS-485 modes. In two-wire mode the TDA (-) and RDA (-) are tied together and so are TDB (+) and RDB (+), making multi-dropping this converter into an existing network easy.

Switch	OFF	ON
One	TD always enabled **(TD 422)**	TD only enabled during data transmission **(TD 485)**
Two	RD always enabled **(ECHO ON)**	RD disabled during data transmission **(OFF)**
Three	Four-wire mode **(4-Wire)**	Two-wire mode **(2-Wire)**
Four	Four-wire mode **(4-Wire)**	Two-wire mode **(2-Wire)**

↑ FOR USE WITH ENLINK



USOPTL4 in RS-422 two-wire set-up with all switches in the ON position.



USOPTL4 in RS-485 two-wire set-up with all switches in the ON position.

Specifications

Dimensions:	3.5 x 1.7 x 0.8 in (8.9 x 4.3 x 2.1 cm)
Temperature Range:	0°C to 70°C (32°F to 158°F)
RS-485/422 Baud Rate:	Up to 460.8 kbps
RS-485/422 Connector:	Pluggable (removable) terminal blocks
USB Baud Rate:	12 Mbps
USB Connector:	USB type-B female
USB Power:	Low power device (draws < 100mA)
Operating System:	Windows 98/SE, 2000, ME, XP, Vista
Accessories:	CD Driver, USB Cable
Isolation:	2000 V RMS
Surge Protection:	15KV ESD

NOTICE

Changing the COM Port Number

The latest software revision allows the user to change the COM port number of a USB serial port. From ***DeviceManager***, select "***View devices by type***", then "***Ports (COM & LPT)***". Select the USB serial port and click ***Properties***. Select the "***Port Settings***" tab, then click ***Advanced***. Choose the required COM port number from the list and click ***OK***.

4.3 U485 – USB TO RS485 OPTICALLY ISOLATED CONVERTER CE

Model USOTL4 (**OBSOLETE as of 1/1/08**)

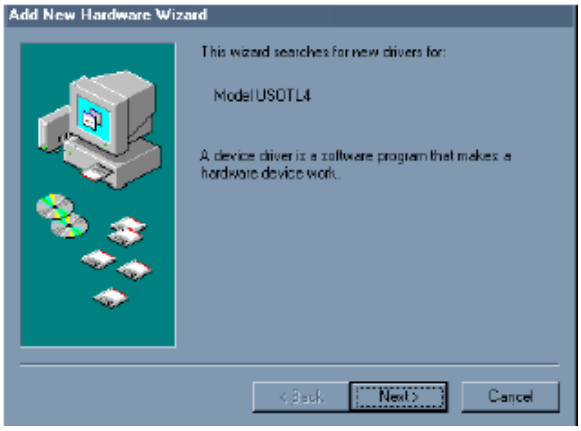
The **USOTL4** is a USB (Universal Serial Bus) port to 2 or 4 wire isolated RS-485/422 converter. This converter requires no PCI/ISA slots or IRQs. Simply plug the converter into an available USB port on your computer or hub. Windows will configure the converter as an additional COM port, compatible with your Windows applications.

The serial port side can be set up for an RS-422 or RS-485 network. A pair of LEDs shows when RS-485/422 data is being received or transmitted. The USB side permits quick setup. Just plug in the USOTL4 and Windows will install the drivers and set up the converter. The USB bus supplies power so no separate power supply is needed. The converter has 2000 volt RMS of isolation.

NOTICE

ENTRON will attempt to keep the most current Installation Drivers for the B&B USOTL4 on the ENLINK CD. The most current Drivers can also be found on the B&B web site at www.bb-elec.com.

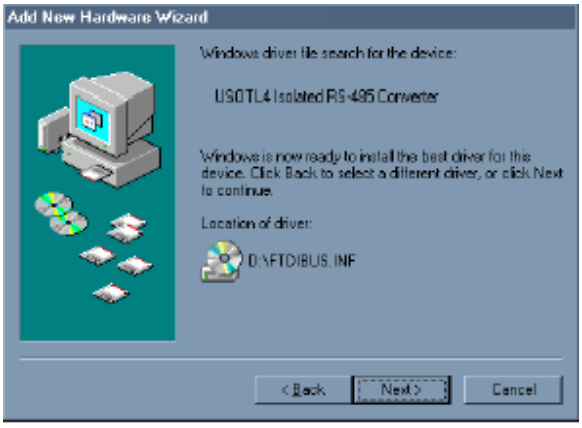
Installation for Windows (Installation for Windows 98 second editions pictured.)



#1. Plug the USOTL4 into an available USB port on your computer or connected hub. The screen above appears, telling you that there is a new device plugged into the USB bus. Click on the **Next>** button.

#2. The screen above appears. Make sure *Search for the best driver for your device* is selected. Then select the **Next>** button.

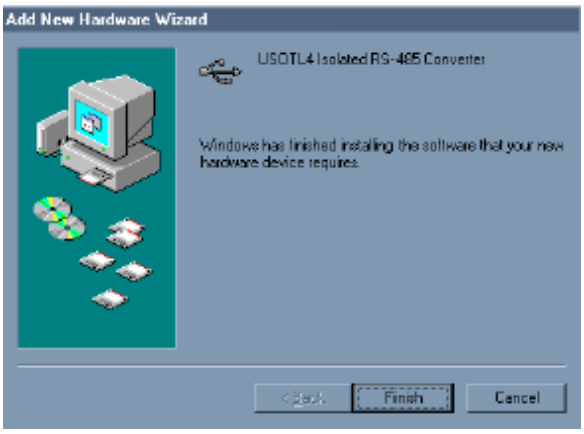
#3. The screen above appears. Make sure *CD-ROM drive* is selected. Insert the *USOTL4 Driver* CD into the CD-ROM drive. Then select the **Next>** button.



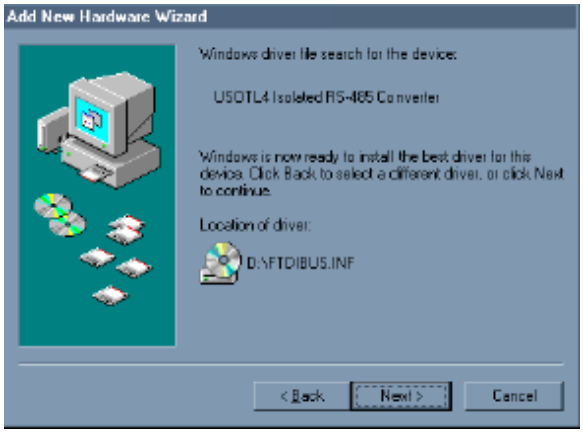
#4. The screen above appears. Make sure *Model USOTL4 (B&B's USB to Isolated RS-485)* is listed as the device. Then select the **Next>** button.

OBSOLETE

4.3 U485 – MODEL USOTL4 (cont.)



#5. The screen above will appear. Click the **Finish** button to complete the installation.



#6. You will also need to continue to install the serial port the same as installing the converter. Click the **Next** button followed by the **Finish** button. It will take a couple seconds for the serial port to be installed.

RS-485 Control

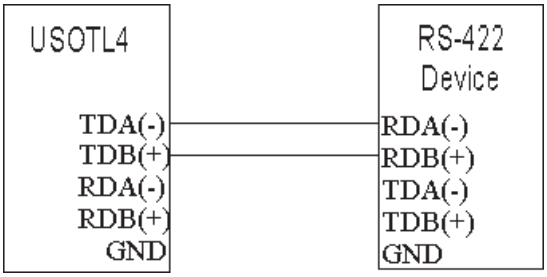
No special software is required to control the RS-485 receiver or transmit line driver. The driver is automatically enabled during each byte transmitted in RS485 mode. The transmitter is always enabled in RS-422 mode. The receiver is tri-stated during each byte transmitted in the echo off mode. The receiver is always enabled in the echo on mode. There are 4.7k ohm pull up/pull down resistors on the RDA and RDB lines. A termination resistor may be added to R16 if needed. See B&B's RS-422/RS-485 Application Note (available at www.bb-elec.com) for more information on termination and DC biasing of an RS-485 network.

Dip Switch Setup

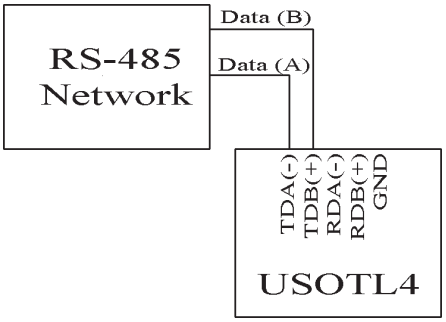
The dip switches allow the module to be configured for two-wire or four-wire, RS-422 or RS-485 modes. In two-wire mode the TDA (-) and RDA (-) are tied together and so are TDB (+) and RDB (+), making multi-dropping this converter into an existing network easy.

Switch	OFF	ON
One	TD always enabled **(TD 422)**	TD only enabled during data transmission **(TD 485)**
Two	RD always enabled **(ECHO ON)**	RD disabled during data transmission **(OFF)**
Three	Four-wire mode **(4-Wire)**	Two-wire mode **(2-Wire)**
Four	Four-wire mode **(4-Wire)**	Two-wire mode **(2-Wire)**

↑ FOR USE WITH ENLINK



USOTL4 in RS-422 two-wire set-up with all switches in the ON position.



USOTL4 in RS-485 two-wire set-up with all switches in the ON position.

Specifications

Dimensions:	3.0 x 0.9 x 1.7 in (7.7 x 2.3 x 4.3 cm)
Temperature Range:	0ºC to 70ºC
RS-485/422 Baud Rate:	Up to 115.2 kbps
USB Baud Rate:	High speed device
USB Power:	Low power device (draws < 100mA)
Operating System:	Windows 98, 2000
Accessories:	Driver disks on 3.5 in. media
Isolation:	2000 V RMS

NOTICE

Changing the COM Port Number

The latest software revision allows the user to change the COM port number of a USB serial port. From ***DeviceManager***, select "***View devices by type***", then "***Ports (COM & LPT)***". Select the USB serial port and click ***Properties***. Select the "***Port Settings***" tab, then click ***Advanced***. Choose the required COM port number from the list and click ***OK***.

5.0 ENBUS PROTOCOL

The ENBUS protocol defines a message format that controls will recognize and obey whether the host is an RT4/RT4jr., a laptop, or some other device using this protocol. It describes the process which a host uses to request access to different controls, how the host will respond to a request from the controls, and how errors will be detected and reported. This protocol provides a standard which hosts and controls use for parsing messages. During communication on an ENBUS network, the protocol determines how each host and control will know its address, or ID Number, recognize a message addressed to it, determine which action will be taken, and extract any data or other information contained in the message.

If a reply is required, depending on the type of action, the hosts or controls will send a reply message according to the ENBUS protocol.

5.1 COMMUNICATION MODE

EN1000 Series Controls and Remote Terminal RT4/RT4jr. communicate on a standard two-wire ENBUS network using binary mode.

COMMUNICATION PARAMETERS

Serial parameters on ENBUS networks are:

- 4800 baud
- 8 data bits
- No parity
- 1 stop bit
- No Flow control

BITS PER BYTE

- 1 **start** bit
- 8 **data** bits
- 1 **stop** bit

5.2 COMMAND/RESPONSE MESSAGE FORMAT

The ENTRON RS485 interface is half duplex and operates in binary mode with specified ENBUS protocol. Using binary (or RTU – Remote Terminal Unit) mode, each eight-bit byte in a message contains two hexadecimal characters. The main advantage of this mode is faster communication than ASCII mode with the same baud rate since binary mode has greater character density. The Command Format for the ENBUS protocol is shown in Table 5-1, and the Response Format is shown in Table 5-2.

Table 5-1. *ENBUS Protocol – Command Message Format*

Ho	Id	FUNCTION	DATA	CheckSum	END
1 byte (8 bits)	1 byte	1 byte	0 to 10 bytes	1 byte	1 byte
41 to FF hex	01 to 40 hex	00 to FF hex	—	00 to FF hex	0D hex

Table 5-2. *ENBUS Protocol – Response Message Format*

Ho	Id	FUNCTION	DATA	CheckSum	END
1 byte	1 byte	1 byte	0 to 10 bytes	1 byte	1 byte
41 to FF hex	01 to 40 hex	00 to FF hex	—	00 to FF hex	0D hex

5.2.1 H.o. ADDRESS FIELD

H.o. is the HOST NUMBER address of a host device such as an RT4/RT4jr., PC, or other device with an RS485 interface. The HOST NUMBER of an RT4/RT4jr. – **H.o.** parameter in EXTENDED FUNCTIONS – could be any number from **65** to **95** (41 to 5F hex). Number **96** is reserved only for serial communication between the RT4/RT4jr. and PC. All other numbers from **97** to **127** might be the address of a PC or other device with an RS485 interface.

5.2.2 I.d. ADDRESS FIELD

I.d. is the IDENTIFICATION NUMBER, or address, of the control which may be programmed using EXTENDED FUNCTIONS, either on the control or RT4/RT4jr. More than one control with the same ID on the same network is not allowed. ID NUMBER **I.d.**=**00** on the RT4/RT4jr. is reserved for broadcasting to all controls on the network, and in this case a response is not expected from the control.

5.2.3 FUNCTION FIELD

FUNCTION is a Command (FnC) or Response (FnR) one byte field. The upper nibble of this byte, in some commands, holds a number of data bytes which follows the FnC or FnR byte, the lower nibble holds the type of action. A full description of the FUNCTION field is found in Section 5.3 ENBUS Protocol Messages.

5.2.4 DATA FIELD

The **DATA** field contains all necessary data required by corresponding FnC or FnR codes. The number of DATA bytes is determined by the upper nibble value in FnC or FnR byte and may vary from 0 up to 10 bytes in one message.

5.2.5 CHECKSUM FIELD

The **CHECKSUM** field is an error checking field and contains a single byte value. The CHECKSUM is the result of a summing calculation performed on the message contents, including the **I.d.** field, FUNCTION field, and DATA field. **H.o.** and END fields are not used for the error CHECKSUM calculation.

5.2.6 END FIELD

The **END** field is at the end of each message and contains 0D hex (13 decimal).

5.3 ENBUS PROTOCOL MESSAGES

ENBUS protocol is a multi-point protocol. All EN1000 Series Controls are **only** slaves and are always in listening mode, waiting for a master to start communication. The only exception to this rule is when the control is actually in a welding sequence, during this time the control cannot be accessed. A host, such as an RT4/RT4jr. or a PC or some other device with an RS485 interface, can be the only master on an ENBUS network. For better understanding of the whole protocol, RAM memory and EEPROM memory maps are necessary; these are found in Appendix 7.5 and 7.6. Table 5-3 illustrates the ENBUS protocol.

Table 5-3. *Commands and Responses*

FUNCTION	FnC hex code	FnR hex code	DESCRIPTION
1. FnId	00	00	Display ID NUMBER on the displays.
2. FnPM	01	01	Put Control in PROGRAM mode.
3. FnOM	02	02	Put Control in OPERATE mode.
4. FnWM	03	03	Put Control in WELD mode.
5. FnNoWM	04	04	Put Control in NO WELD mode.
6. FnCSC1	05	05	Copy current schedule to next one.
7. FnCSC10	06	06	Copy current schedule to schedule plus ten.
8. FnClrSC	07	07	Clear current schedule.
9. FnDATE	0A	AA	Read date when the latest software revision was made.
10. FnEPROM	0B	BB	Read last software (firmware) version number.
11. FnReSet	0F	0F	Restart the Control.
12. FnR1	11	11	Reload one byte from the Control.
13. FnR2	12	22	Reload two data bytes from RAM of the Control.
14. FnR8	18	88	Reload eight data bytes from RAM of the Control.
15. FnR16	1E	1E	Reload sixteen data bytes from RAM of the Control.
16. FnUSWS	1A	0A	Update push buttons status.
17. FnUSC	1B	0B	Send dialed Schedule Number.
18. FnErr	—	1F	Send Error Code to Host, from Control.
19. FnScan	1F	2F	Send Scan Code to Control: Error Monitoring, status, read current...
Note: **FnScan** is always followed by one DATA byte which represents **ScanCode**. Respond **FnscanR** is always followed by two or four DATA bytes, depending on ScanCode value. More details about FnScan and ScanCode values are found Section 5.4 Examples.			
20. FnUD	20	00	Update Sequence and EF display pointers.
21. FnREE1	21	11	Read one byte from EEPROM of the Control.
22. FnREE2	22	22	Read two bytes from EEPROM of the Control.
23. FnREE8	28	88	Read eight bytes from EEPROM of the Control.
24. FnREE16	2E	2E	Read sixteen bytes from EEPROM of the Control.
25. FnUDATA	30	00	Update data on DATA displays.
26. FnWEE1	31	01	Write one byte to EEPROM of the Control.
27. FnWEE2	42	02	Write two bytes to EEPROM of the Control.
28. FnWEE8	A8	08	Write eight bytes to EEPROM of the Control.
29. FnWEE16	AF	AF	Write sixteen bytes to EEPROM of the Control.

5.4 EXAMPLES

In these examples, the following addresses are used: **H.o.**=41 hex (65 decimal), and **I.d.**=01 hex (01 decimal) or **I.d.**=**00** for broadcasting to all controls. All numbers are in hex code.

Ho	Id	FUNCTION	DATA	CheckSum	END
41	01	—	—	—	0D

	COMMAND	RESPONSE	DESCRIPTION
1a.	41 00 00 00 0D	41 00 00 00 0D	Display ID NUMBERS on all Controls (Broadcasting).
1b.	41 01 00 01 0D	41 01 00 01 0D	Display ID NUMBER on Control with **I.d.**=**01**.

The same example in different form

Ho	Id	FUNCTION	DATA	CheckSum	END
41	01	00	—	01	0D

2.	41 01 01 02 0D	41 01 01 02 0D	Put Control with **I.d.**=**01** in PROGRAM mode.
3.	41 01 02 03 0D	41 01 02 03 0D	Put Control with **I.d.**=**01** in OPERATE mode.
4.	41 01 03 04 0D	41 01 03 04 0D	Put Control with **I.d.**=**01** in WELD mode.
5.	41 01 04 05 0D	41 01 04 05 0D	Put Control with **I.d.**=**01** in NO WELD mode.
6.	41 01 05 06 0D	41 01 05 06 0D	Copy current schedule to next one.
7.	41 01 06 07 0D	41 01 06 07 0D	Copy current schedule to schedule plus ten.
8.	41 01 07 08 0D	41 01 07 08 0D	Clear current schedule.
9.	41 01 0A 0B 0D	41 01 AA <10/15/2001.> <CheckSum> 0D	Read last software revision DATE.
10.	41 01 0B 0C 0D	41 01 BB <619016-002D> <CheckSum> 0D	Read last software version number.
11.	41 01 0F 10 0D	——	Restart Control, no response.
12.	41 01 11 23 35 0D	41 01 11 55 67 0D	Reload one byte from Control: from RAM address 23 hex, reload value 55 hex.
13.	41 01 12 23 36 0D	41 01 22 55 55 CD 0D	Reload two bytes from Control, two successive data bytes, first is on RAM address 23 hex.
14.	41 01 18 40 59 0D	41 01 88 0A 04 3C 00 00 01 00 00 D4 0D	Reload eight RAM data bytes.
15.	41 01 1E 40 5F 0D	41 01 1E 0A 04 3C 00 00 01 00 00 00 00 00 00 00 00 00 6A 0D	Reload sixteen successive RAM data bytes, first on address 40 hex.
16.	41 01 1A FD 18 0D	41 01 0A 0B 0D	Update push buttons status; i.e., simulation push buttons pressing from Remote Terminal.

Hex values:
FE hex - SELECT
FB hex - WELD/NO WELD
FD hex - ENTER
FE hex - PROGRAM/OPERATE
7F hex - SCHEDULE10
BF hex - SCHEDULE1
DF hex - DATA10
EF hex - DATA1

17.	41 01 1B 14 30 0D	41 01 0B 0C 0D	Send dialed Schedule Number 20.
18.	——	41 01 1F 01 21 0D	Send Error Code to the Host – **E.r.**=**01**.

5.4 EXAMPLES (cont.)

19a.	41 01 1F F1 11 0D	41 01 2F 00 00 30 0D	ScanCodeR: Error Monitoring. Response: No errors.
19b.	41 01 1F F1 11 0D	41 01 2F F1 01 22 0D	ScanCodeR: Error Monitoring. Response: **E.r.=01**.
19c.	41 01 1F F1 11 0D	41 01 2F FF 01 30 0D	ScanCodeR: Error Monitoring. Still error: **E.r.=01**.
19d.	41 01 1F F1 11 0D	41 01 2F F0 01 21 0D	ScanCodeR: Error Monitoring **E.r.=01** is cleared.
19e.	41 01 1F F1 11 0D	41 01 2F E1 00 60 3C AD 0D	ScanCodeR: Current was **0060** and 60% phase shift.
19f.	41 01 1F F1 11 0D	41 01 2F E3 00 85 56 EE 0D	ScanCodeR: **H. .**–current was **0085** & 86% phase shift.
19g.	41 01 1F E1 01 0D	41 01 2F E2 00 50 31 93 0D	ScanCodeR: **L.o.**–current was **0050** & 49% phase shift.

ScanCode Commands and Responses:
ScanCode = F1 hex => Scan Control and check for errors.
ScanCodeR = 00 and ErrNo = 00 => no errors, Control is ready to operate.
ScanCodeR = F1 hex and ErrNo = 01 to 37 => one of errors from **E.r.=01** to **E.r.=37**.
ScanCodeR = FF hex and ErrNo = 01 to 37 => Control still has the same error.
ScanCodeR = F0 hex and ErrNo = 00 to 37 => Error is cleared.

ScanCode = F1 hex => Scan Control and check for errors or Read current after end of weld (must be used with **P.O.=22** or **23**);
ScanCode = E1 hex => Scan Control and read current for last weld, anytime before next weld initation.
ScanCodeR = E1 hex and three more data bytes => measured current after end of weld.
ScanCodeR = E2 hex and three more data bytes => **L.o.** occurred and current after end of weld.
ScanCodeR = E3 hex and three more data bytes => **H. .** occurred and current after end of weld.

NOTE: First two bytes are measured current after weld in % or kA –First byte is high bcd byte, two LEFT digits
–Second byte is low bcd byte, two RIGHT digits
Third byte is phase shift of last weld cycle in %.

20.	41 01 20 02 00 23 0D	41 01 00 01 0D	Update Sequence and EF display pointers.
21.	41 01 21 A0 00 C2 0D	41 01 11 55 67 0D	Read one byte from EEPROM: Page A0, address 00; Control responds with one data byte 55.
22.	41 01 22 A0 00 C3 0D	41 01 22 55 55 CD 0D	Read two successive bytes from EEPROM.
23.	41 01 28 A0 00 C9 0D	41 01 22 00 00 00 00 00 01 00 00 24 0D	Read 8 bytes from EEPROM.
24.	41 01 2E A0 00 CF 0D	41 01 2E 0A 04 3C 00 00 01 00 00 00 00 00 00 00 00 00 7A 0D	Reload sixteen bytes from EEPROM: Page A0 hex, address 00 hex.
25.	41 01 30 10 01 00 42 0D	41 01 00 01 0D	Update DATA displays with new values: **10** or **0100**.
26.	41 01 31 A0 00 0A DC 0D	41 01 01 02 0D	Write one byte to EEPROM: Page A0, address 00; data 0A.
27.	41 01 42 A0 00 0A 0A F7 0D	41 01 02 03 0D	Write two bytes to EEPROM: Page A0, address 00.

The same example in different form

Ho	Id	FUNCTION	DATA	CheckSum	END
41	01	42	A0 00 0A 0A	F7	0D

28a.	41 01 A8 A0 00 0A 04 3C 00 00 01 00 00 94 0D	41 01 A8 A9 0D	Write eight bytes to EEPROM: Page A0 hex, address 00 hex.
28b.	41 01 A8 A0 00 0A FF 3C 00 00 01 00 00 8F 0D	41 01 2F 20 <data pointer> <CheckSum> 0D	Error Code Response: Data out of range.
28c.	41 01 A8 FF 00 0A 04 3C 00 00 01 00 00 E9 0D	41 01 2F 20 20 70 0D	Error Code Response: Wrong EEPROM page number.
29.	41 01 AF A0 00 0A 04 3C 00 00 01 00 00 00 00 00 00 00 00 00 00 9B 0D	41 01 AF B0 0D	Write sixteen bytes to EEPROM: Page A0 hex, address 00 hex.

6.0 ERROR CODES AND PROCESS OUTPUTS

6.1 ERROR CODES

Beside standard error codes described in Application Note 700158, some of the following errors may occur only on Remote Terminal RT4/RT4jr.

E.r.=40 No active control with the requested **I.d.** is on the bus. This error will occur if the control with the desired **I.d.** is turned off or does not respond. After clearing this error, [—] will be displayed on the RT4/RT4jr. DATA display, which means no data is being received from the control.

E.r.=41 Message is not received. This error will occur if the control does not respond on request from RT4/RT4jr.

E.r.=42 Error in **FUNCTION** byte. This error will occur if the Command or Respond FUNCTION byte is not valid; i.e., if the FUNCTION byte does not match the values from the ENBUS protocol.

E.r.=43 **CHECKSUM** error. This error will occur if the data is corrupt during communication between RT4/RT4jr. and corresponding controls.

Each of the errors may be cleared by pressing any push button on RT4/RT4jr. Front Panel.

NOTICE
Weld Controls will not communicate during weld sequences.

6.2 PROCESS OUTPUTS

Two PROCESS OUTPUTS are directly correlated to ENBUS protocol and Error and Status Monitoring.

P.O.=22 Send HI/LO and CURRENT after out of limit window weld or last weld in sequence with RS485 Option (*EN1001 only*).

Will send data for out of limit window weld in sequence or last weld only in completed sequence to master on ENBUS or to serial printer. Display will flash **H. ı.** or **L.o.** and measured CURRENT if out of limit window.

P.O.=23 Send HI/LO and CURRENT if out of limit window with RS485 Option (*EN1001 only*).

Will send data after weld if the CURRENT is out of limit window to master on ENBUS or to serial printer. Display will flash **H. ı.** or **L.o.** and measured CURRENT.

If **I.d.=00**, these PROCESS OUTPUTS can be used to send data to a serial printer at the end of completed weld sequence. If **I.d.** is between **01** and **64**, data will be prepared to be sent to a master PC with ENLINK software or to a master device. With these two PROCESS OUTPUTS, the weld sequence will be stopped and the control will wait for a master to scan or read all necessary prepared data. At the same time, the control will show CURRENT on the display and will flash **L.o.** or **H. ı.** if they exist until any push button is pressed on the Front Panel. In other words, in order to clear the display of errors, any push button on the display must be pressed and the master must read the prepared message from the control. Only after an error is cleared will the control allow re-initiation of a new weld sequence.

7.0 APPENDICES

7.1 TERMINATION OPTIONS ON RS485 DRIVERS

There are several options for terminating the RS485 bus. Using a dip switch, ENTRON RS485 drivers can be terminated with fail-safe biasing (Figure 7-4), unterminated (Figure 7-3), and parallel terminated (Figure 7-2). For more information, refer to interconnect wiring of RS485 driver boards (A/N 421407).

NOTICE

For proper operation of the RS485 bus, the control with fail-safe biasing configuration (last position) should remain powered on.

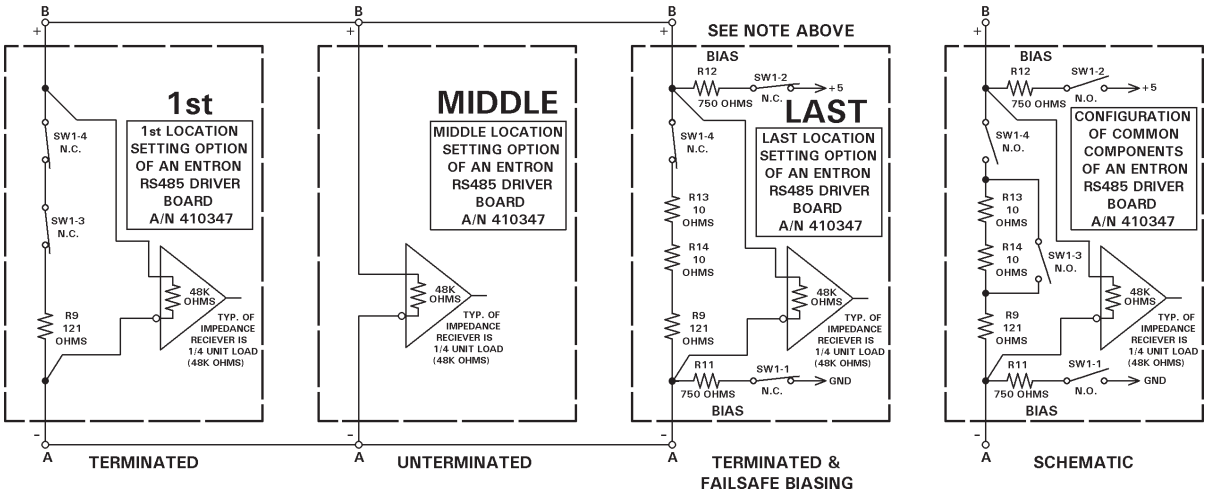


Figure 7-1. *Termination options*

FIRST RS485 driver
- parallel termination
- dip switch:**OFF-OFF-ON-ON**
- default settings for RS485 drivers of **RT4**

All MIDDLE RS485 drivers
- unterminated
- dip switch: **OFF-OFF-OFF-OFF**

LAST RS485 driver
- fail-safe termination
- dip switch: **ON-ON-OFF-ON**
- default settings for RS485 drivers of **welding controls**

See NOTE above

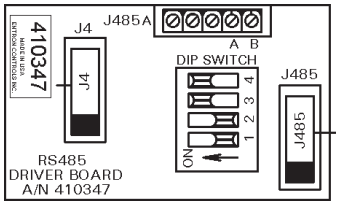


Figure 7-2.
First RS485 driver

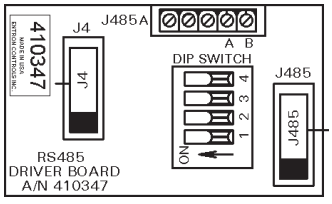


Figure 7-3.
Middle RS485 drivers

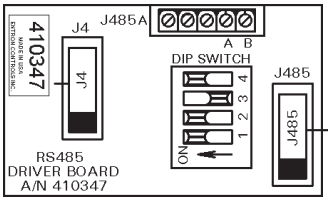


Figure 7-4.
Last RS485 driver

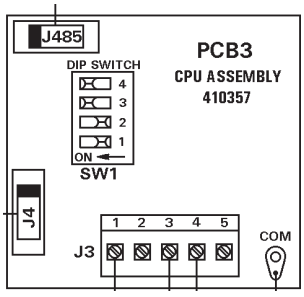


Figure 7-5.
RT4jr. layout

Figure 7-5 shows layout of the RT4jr. drive board with dip switch set for parallel termination.

7.2 FIELD CONSTRUCTION OF J485-J485 HARNESS ASSEMBLY

Weld controls and RT4jr. are provided with components to build cable assemblies to interconnect RS485-equipped controls (pre-fabricated J485-J485 cable assemblies are not available). Cable assemblies will need to be constructed onsite after routing through conduits, holes, troughs, etc. All cables should be separated as much as possible from other high voltage wires connecting to solenoid valves, welding transformers, and the AC line. Connectors are set-screw type, requiring no soldering. If provided cable (25') is not adequate, longer cable length can be ordered.

To create a J485-J485 Harness Assembly, the following parts are required (supplied with weld controls equipped with RS485 Option):

2	331136	Connector, 9 Pin, Screw Terminal, “D” Style, Plug
25'	900258	Cable, 4 Conductor, 24 Ga. Stranded w/Shield
2	460228	Label, J485

Additional Connectors, Cable and Labels are available at additional cost

Reference Drawing: 421407 Interconnect Wiring of RS485 Driver Boards
(see Section 7.1 if this drawing is not available)

ASSEMBLY INSTRUCTIONS:

1. Cut cable to length or route from source to destination. DO NOT route cable with or place cable in same conduit with wires carrying 120VAC or higher.
2. Strip outer insulation and foil shield at each end 1-1/2" and wire both ends as shown in Figure 7-6.

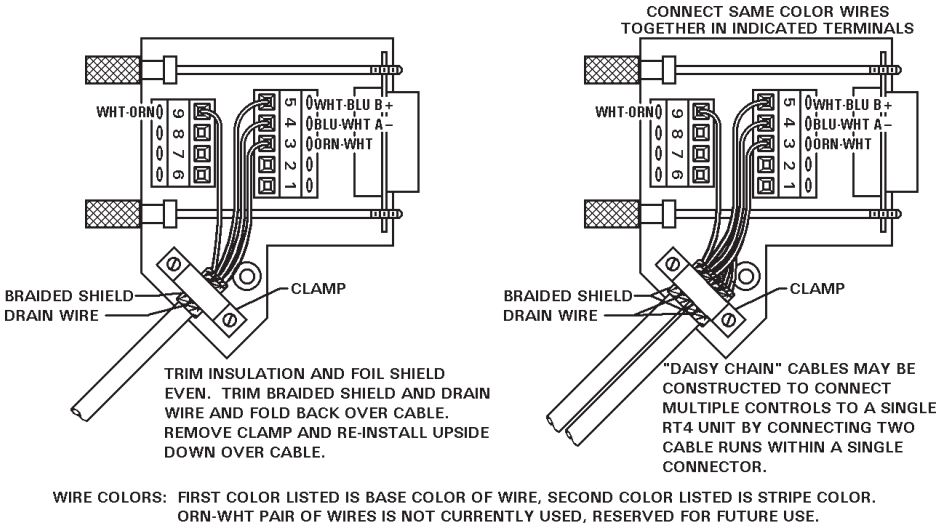


Figure 7-6. *J485 Connector wiring*

3. Inspect connections before proceeding to next step. Be aware that, in multiple connection installations, one wrong connection will stop ALL communication.
4. Assemble cover to plug assembly at both ends by snapping cover in place and using provided screw to secure assembly.

NOTICE
Connector can be assembled with cable exiting to either side.

5. Complete assembly by installing the provided labels on all connectors as shown in Figure 7-7.

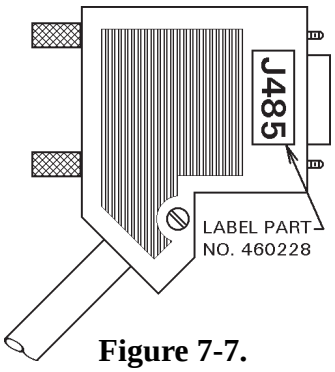


Figure 7-7.
J485 Connector labeling

7.3 RS485 RETROFIT OPTION ASSEMBLIES

RS485 RETROFIT OPTION ASSEMBLIES – WHERE USED CHART

(Determine the correct combination of Control Style and Cabinet Style)

OPTION KITS for	B,S,E	C# & T/D	NEMA 12 ENCL.
EN1000	600656-001	600656-002	600656-003
EN1001	600657-001	600657-002	600657-003
EN1000/EN1001 Cascade	N/A	600658-002*	600658-003
EN1200 – **OBSOLETE CONTROL**	N/A	600703-002**	N/A

RS485 RETROFIT OPTION KIT BILL OF MATERIAL

600656-001	600656-002	600656-003	600657-001	600657-002	600657-003	600658-002	600658-003	600703-002	PART NO.	DESCRIPTION
1	1	1	1	1	1	1	1	1	410347	PCB Assembly, RS485 Option
1			1						322415	Harness, J4-J4, 22-1/2"
	1					1			322416	Harness, J4-J4, 26-1/2"
				1					322422	Harness, J4-J4, 24"
		1			1		1		322417	Harness, J4-J4, 34"
								1	322482	Harness, J4-J4-J4, 54"
1		1	1		1		1	1	322369-010	Harness, J485-J485, 34"
	1			1		1			322369-011	Harness, J485-J485, 13-3/4"
1	1	1							619010-002†	EN1000 PROM firmware version **E** (or later)
			1	1	1				619016-001	EN1001 PROM firmware version **N** (or later)
						1	1		619044-002	EN1000/1001 Cascade PROM version **ORIG** (or later)
								1	619024-002	EN1200 PROM version **E** (or later)
	1	1		1	1	1	1		525035	Bracket, PCB Mounting
4	4	4	4	4	4	4	4	4	331124	Connector, Jack Screw, for .090 Panel
2	2	2	4	2	2	2	2	4	557003	6-32 x 1/4 PHSMS, Phil, Brite
	2	2		2	2		2		557006	6-32 x 3/8 PHSMS, Phil, Brite
	2	2		2	2	2	2	2	557017	#6 Med. Split Lockwasher, Brite
	2			2		2			557018	6-32 x 1/4 AF Hex Nut, Brite
2			2					2	555031	Spacer, Hex Threaded Brass, 6-32 x 1/2
2	2	2	2	2	2	2	2	2	342012	Cable Clamp, 5/32 Flat, Adhesive Back
1	1	1	1	1	1	1	1	1	342020	Cable Clamp, 2 x .17 Thk, Adhesive Back
2	2	2	2	2	2			3	460129	Label, "J4"
						2	2		460129	Label, "J4B"
1	1	1	1	1	1	1	1	1	460231	Label, RS485
2	2	2	2	2	2	2	2	2	331136^	Connector, 9 Pin, Screw Terminal, "D" Style, Plug
25'									900258^	Cable, 4 Conductor, 24 Ga. Stranded w/Shield
2	2	2	2	2	2	2	2	2	460228^	Label, J485
1	1	1	1	1	1	1	1	1	421407	Interconnect Wiring of RS485 Driver Boards
1	1	1	1	1	1	1	1	1	600689	ENLINK 1000/1001 for Windows
1	1	1	1	1	1	1	1	1	700171	Manual, 485 Options

The Harness Assemblies included with kit may not match the Harness Assemblies listed on Wiring Diagram. Either Assembly will work; the harness supplied with kit may be overly long.

See Section 7.3.2 for cabinet modification and mounting instructions for EN1000/EN1001 in "C" Cabinet.

* See Section 7.3.10 for cabinet modification and mounting instructions for EN1000/EN1001 Cascade Control in "T" Cabinet.

** See Section 7.3.9 for cabinet modification and mounting instructions for EN1200 in "T" Cabinet.

† EN1000 Controls with PROM firmware version 619010-**001** require a new Control Board.

^ These parts comprise J485-J485 Harness Assembly which is field-assembled. See Section 7.2 for instructions.

See Sections 7.3.3, 7.3.4, and 7.3.5 for information on Retrofit Kits for Existing Multiple EN1001 Controls.

7.3.1 FIELD INSTALLATION OF RETROFIT RS485 OPTION

See Section 7.3.2 for 485 Option in EN1000/EN1001 “C” Cabinet.
See Section 7.3.9 for 485 Option in an EN1200 “T” Cabinet.
See Section 7.3.10 for 485 Option in EN1000/EN1001 Cascade “T” Cabinet.

1. Remove **ALL** power to control and open door.
2. In EN1000, EN1001, EN1000 Cascade and EN1001 Cascade Controls, if the version letter on the PROM (on the Control Board) is earlier than the new PROM included with this kit, remove and replace the old PROM with the new PROM.

NOTICE
In EN1000 Controls, if the PROM Part No. is 619010-**001** (Original release through version **S**), a new Control Board is required.

3. In “T/D” Cabinets, mount bracket (P/N 525035) on the two 6-32 X 3/8 studs on the left side of cabinet using the #6 lockwashers and 6-32 hex nuts (P/Ns 557017 and 557018) supplied. In NEMA 12 Enclosures, mount the bracket using 6-32 X 3/8 Phillips Pan Head Screws and #6 lockwashers in the holes provided on the rear panel.
4. In “T/D” and NEMA 12 Enclosures, mount the RS485 PCB Assembly (410347) to bracket using the 6-32 X 1/4 Phillips Pan Head Screws (P/N 557003) supplied. In “B”, “S”, and “E” Cabinets, mount the PCB to the existing standoffs as indicated in Figure 7-8.
5. In “G”, “H”, and “U” NEMA Cabinets, drill 7/16" diameter hole and punch 9 Pin D-Subminiature Connector mounting using Greenlee No. 229 Punch at positions shown in Figure 7-9. In “B”, “S”, “E”, “T/D” and “L” Cabinets purchased after December 1999, mounting has been provided.
6. Mount the dual 9 Pin, D-Subminiature Connector end of J485-J485 Harness to the cabinet using the Jack Screws (P/N 331124) supplied. Plug the other end of the harness into J485 jack on RS485 PCB Assembly. The red stripe on harness **MUST** be oriented as shown on Wiring Diagram.
7. Plug J4-J4 (or J4B-J4B) Harness Assembly into J4 jack on RS485 PCB Assembly and J4 (or J4B) jack on Indicator PCB Assembly. The red stripe on harness **MUST** be oriented as shown on Wiring Diagram. Add supplied harness identification Labels (J4 or J4B) to the connectors.

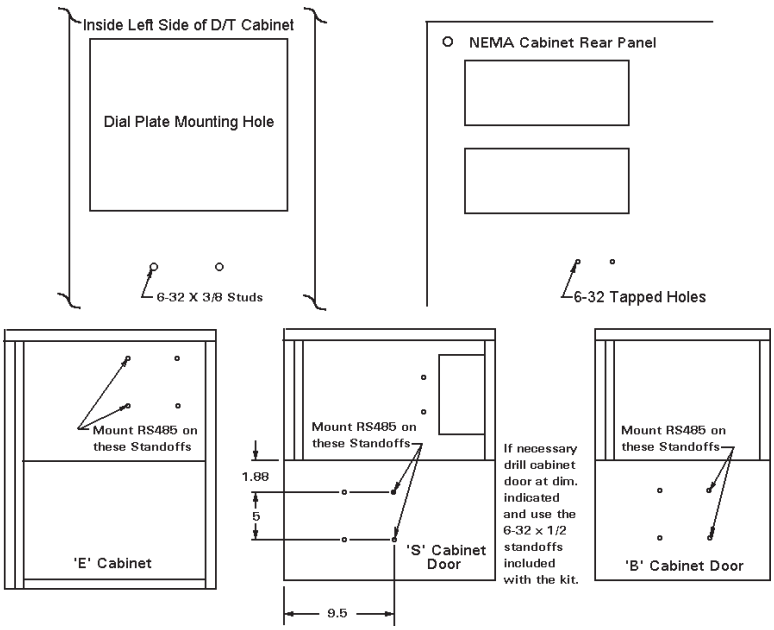


Figure 7-8. *RS485 mounting diagrams*

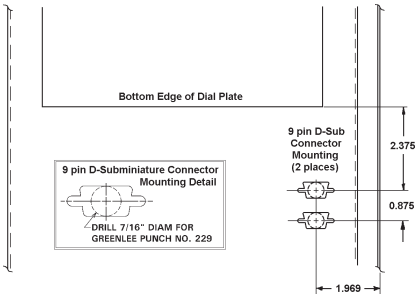


Figure 7-9.
9 pin D-Subminiature connector mounting for “G”, “H”, and “U” Cabinets

7.3.2 MODIFICATION TO ADD RS485 TO EXISTING EN1000/EN1001 CONTROL IN "C" CABINET

See Section 7.3 for Parts List for Kit Assemblies 600656-002 and 600657-002.

MODIFICATION NOTES:

1. Remove **ALL** power to control. Open door.

2. Remove existing cable clamps and associated hardware. Move harnesses away from area to be punched, drilled and tapped.

3. Drill and tap per Cabinet modification detail (see Figure 7-10), be sure to vacuum or otherwise remove **ALL** metal chips before assembling Mounting Bracket (P/N 525035) to left side of cabinet. Use 2 ea. 6-32 x 3/8 PHSMS, Phil Brite, from the outside and 2 ea. #6 Med. Split Lockwashers and 2 ea. 6-32 x 1/4 A/F Hex Nuts on the inside.

4. Mount the RS485 PC Board Assembly (P/N 410347) to the Mounting Brackets using 2 ea. 6-32 x 1/4 PHSMS, Phil, Brite (see Figure 7-10).

5. Replace existing PROM 619016-001 in Control Board 600572-002 with new PROM firmware version 619016-001**N** (or later) supplied with this kit.

6. Connect J4 Harnesses and complete point-to-point wiring of J485A terminals per Wiring Diagram 421499-001 included with this set of instructions. Twist point-to-point wires approximately one twist per inch. Install 9 Pin D-Subminiature Connectors and associated harnesses per Wiring Diagram.

7. Remove existing 4 ea. #6-32 Screws and 2 ea. cover plates and discard. Secure harnesses to cabinet and Mounting Brackets using supplied cable clamps and wire ties and using existing 9 Pin D-Subminiature Connector holes.

8. Close cabinet, reapply power to control.

NOTICE

On the Wiring Diagram, the dark band on connectors indicates stripe on ribbon harness. Harness **MUST** be installed with ribbon harness stripe oriented correctly.

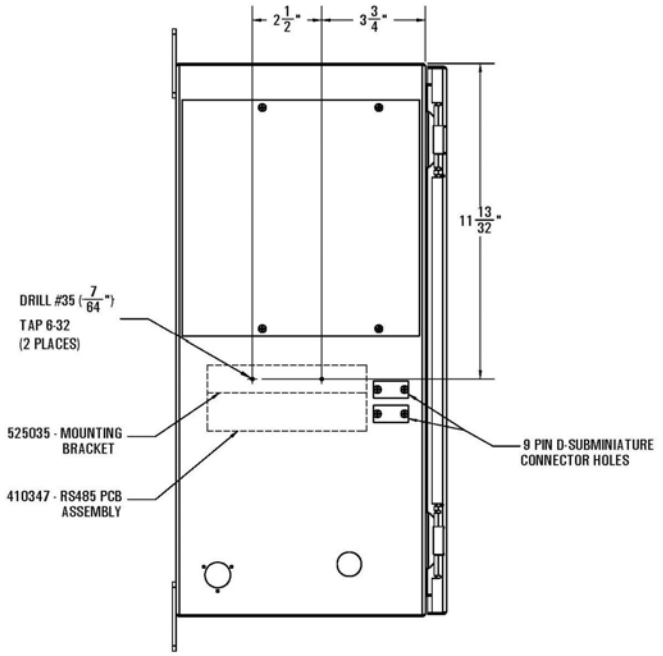


Figure 7-10.
Cabinet modification detail for EN1000/EN1001-C

7.3.3 KIT ASSEMBLY NO. 600657-006

MODIFICATION TO ADD RS485 TO EXISTING EN(6)1001-UF

PARTS LIST

6	619016-001	PROM, Firmware Version **N**, Programmed for EN1001
6	410347	PCB Assembly, RS485 Driver Bd.
6	525035	Bracket, Contactor & Firing Bd. Mounting
6	331069	Connector, Plug, 5 Pin, Screw Terminal
1	322441	Harness Assem., J4-J4 Ctrl. Bd. 1 to RS485 1
1	322442	Harness Assem., J4-J4 Ctrl. Bd. 2 to RS485 2
1	322443	Harness Assem., J4-J4 Ctrl. Bd. 3 to RS485 3
1	322463	Harness Assem., J4-J4 Ctrl. Bd. 4 to RS485 4
1	322464	Harness Assem., J4-J4 Ctrl. Bd. 5 to RS485 5
1	322465	Harness Assem., J4-J4 Ctrl. Bd. 6 to RS485 6
2	322369-012	Harness Assem., RS485 to Cabinet Connector
4	331124	Connector, Jackscrew, for .090 Panel
25'	900258	Cable, 4 Conductor, 24AWG w/Shield
2	331136	Connector, Plug, 9 Pin, Screw Terminal
12	557003	6-32 X 1/4 PHSMS, Phil, Brite
12	557006	6-32 X 3/8 PHSMS, Phil, Brite
12	557017	#6 Med. Split Lockwasher, Brite
12	557018	6-32 X 1/4 A/F Hex Nut, Brite
5'	900022	Wire, Stranded, 22AWG, Blu
5'	900213	Wire, Stranded, 22AWG, Wht/Blu
8	342009	Cable Tie, 1-1/4' Dia. Wrap
6	342012	Cable Clamp, Flat Cable Mt.

MODIFICATION NOTES:

1. Remove **ALL** power to control. Open door.

2. Remove existing cable clamps and associated hardware. Move harnesses away from area to be punched, drilled and tapped.

3. Drill 7/16" diameter hole and punch 9 Pin D-Subminiature Connector mounting using Greenlee No. 229 Punch at positions per 9 Pin D-Subminiature Connector mounting detail (see Figures 7-11 and 7-12).

4. Drill and tap per Cabinet modification detail (see Figure 7-11), be sure to vacuum or otherwise remove **ALL** metal chips before assembling Mounting Brackets (P/N 525035) to left side of cabinet. Use 2 ea. 6-32 x 3/8 PHSMS, Phil Brite, from the outside and 2 ea. #6 Med. Split Lockwashers and 2 ea. 6-32 x 1/4 A/F Hex Nuts on the inside.

5. Mount the RS485 PC Board Assembly (P/N 410347) to the Mounting Brackets using 2 ea. 6-32 x 1/4 PHSMS, Phil, Brite (see Figure 7-11).

7.3.3 KIT ASSEMBLY NO. 600657-006 (cont.)

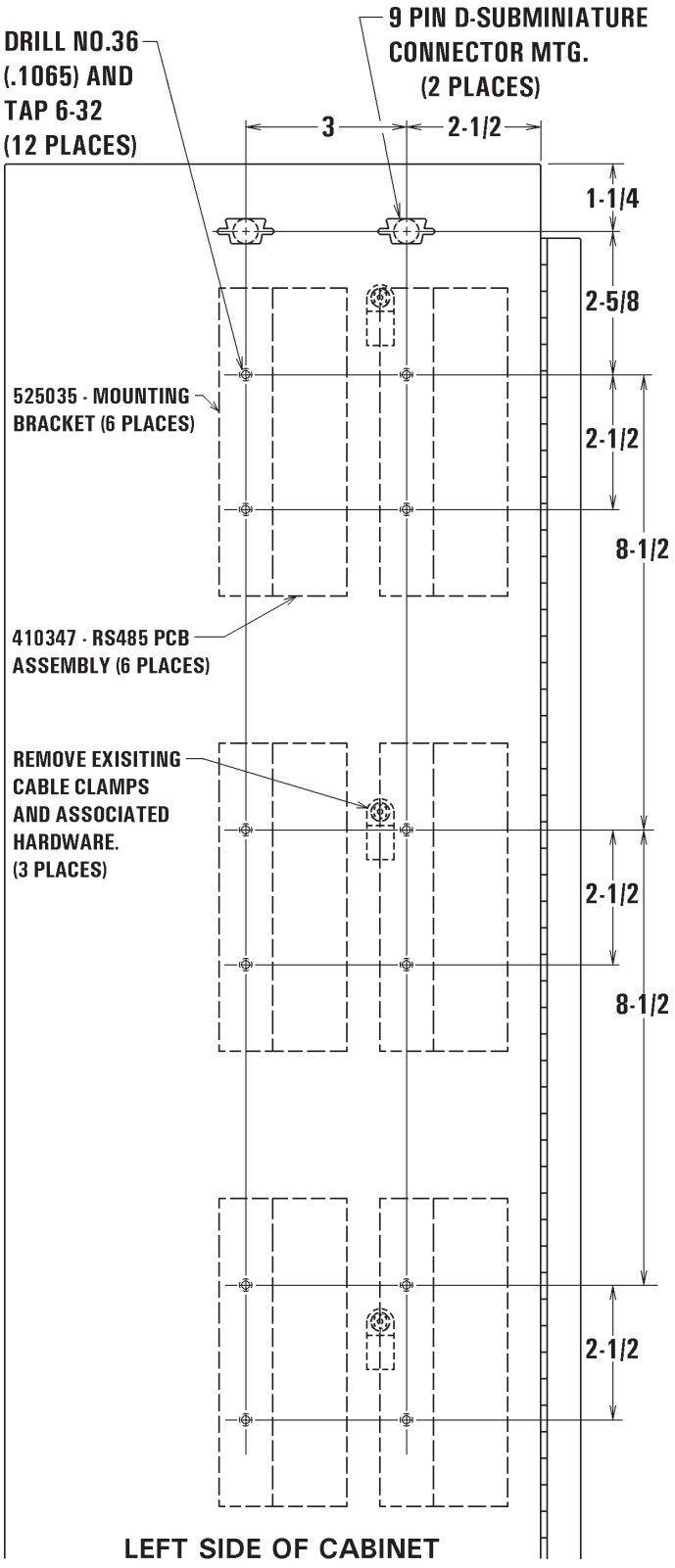


Figure 7-11.
Cabinet modification detail for EN(6)1001-UF

6. Replace existing PROM 619016-001 in Control Board 600572-002 with new PROM firmware version 619016-001**N** (or later) supplied with this kit.

7. Connect J4 Harnesses and complete point-to-point wiring of J485A terminals per Wiring Diagram 421413 included with this set of instructions. Twist point-to-point wires approximately one twist per inch. Install 9 Pin D-Subminiature Connectors and associated harnesses per Wiring Diagram.

8. Secure harnesses to cabinet and Mounting Brackets using supplied cable clamps and wire ties.

9. Close cabinet, reapply power to control.

NOTICE

On the Wiring Diagram, the dark band on connectors indicates stripe on ribbon harness. Harness **MUST** be installed with ribbon harness stripe oriented correctly.

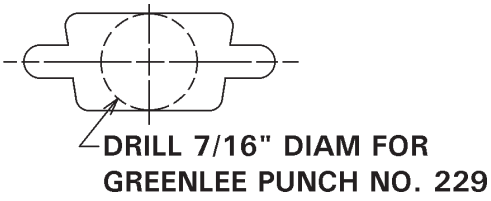


Figure 7-12.
9 Pin D-Subminiature Connector mounting detail (2 places)

7.3.4 KIT ASSEMBLY NO. 600657-005

MODIFICATION TO ADD RS485 TO EXISTING EN(4)1001-HF

PARTS LIST

4	619016-001	PROM, Firmware Version **N**, Programmed for EN1001
4	410347	PCB Assembly, RS485 Driver Bd.
4	525035	Bracket, Contactor & Firing Bd. Mounting
4	331069	Connector, Plug, 5 Pin, Screw Terminal
1	322441	Harness Assem., J4-J4 Ctrl. Bd. 1 to RS485 1
1	322442	Harness Assem., J4-J4 Ctrl. Bd. 2 to RS485 2
1	322443	Harness Assem., J4-J4 Ctrl. Bd. 3 to RS485 3
1	322463	Harness Assem., J4-J4 Ctrl. Bd. 4 to RS485 4
2	322369-012	Harness Assem., RS485 to Cabinet Connector
4	331124	Connector, Jackscrew, for .090 Panel
25'	900258	Cable, 4 Conductor, 24AWG. w/Shield
2	331136	Connector, Plug, 9 Pin, Screw Terminal
8	557003	6-32 X 1/4 PHSMS, Phil, Brite
8	557006	6-32 X 3/8 PHSMS, Phil, Brite
8	557017	#6 Med. Split Lockwasher, Brite
8	557018	6-32 X 1/4 A/F Hex Nut, Brite
30"	900022	Wire, Stranded, 22AWG, Blu
30"	900213	Wire, Stranded, 22AWG, Wht/Blu
6	342009	Cable Tie, 1-1/4' Dia. Wrap
6	342012	Cable Clamp, Flat Cable Mt.

MODIFICATION NOTES:

1. Remove **ALL** power to control. Open door.

2. Remove existing cable clamps and associated hardware. Move harnesses away from area to be punched, drilled and tapped.

3. Drill 7/16" diameter hole and punch 9 Pin D-Subminiature Connector mounting using Greenlee No. 229 Punch at positions per 9 Pin D-Subminiature Connector mounting detail (see Figures 7-13 and 7-14).

4. Drill and tap per Cabinet modification detail (see Figure 7-13), be sure to vacuum or otherwise remove **ALL** metal chips before assembling Mounting Brackets (P/N 525035) to left side of cabinet. Use 2 ea. 6-32 x 3/8 PHSMS, Phil Brite, from the outside and 2 ea. #6 Med. Split Lockwashers and 2 ea. 6-32 x 1/4 A/F Hex Nuts on the inside.

5. Mount the RS485 PC Board Assembly (P/N 410347) to the Mounting Brackets using 2 ea. 6-32 x 1/4 PHSMS, Phil, Brite (see Figure 7-13).

6. Replace existing PROM 619016-001 in Control Board 600572-002 with new PROM firmware version 619016-001**N** (or later) supplied with this kit.

7.3.4 KIT ASSEMBLY NO. 600657-005 (cont.)

7. Connect J4 Harnesses and complete point-to-point wiring of J485A terminals per Wiring Diagram 421412 included with this set of instructions. Twist point-to-point wires approximately one twist per inch. Install 9 Pin D-Subminiature Connectors and associated harnesses per Wiring Diagram.

8. Secure harnesses to cabinet and Mounting Brackets using supplied cable clamps and wire ties.

9. Close cabinet, reapply power to control

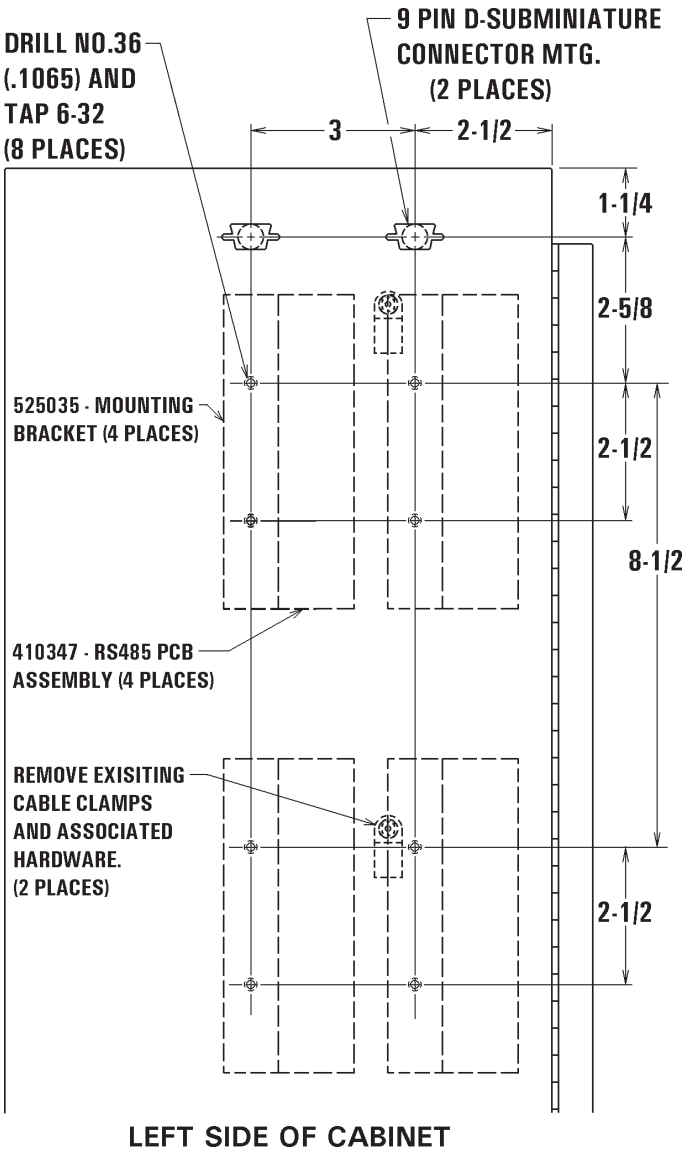


Figure 7-13.
Cabinet modification detail for EN(4)1001-HF

NOTICE

On the Wiring Diagram, the dark band on connectors indicates stripe on ribbon harness. Harness **MUST** be installed with ribbon harness stripe oriented correctly.

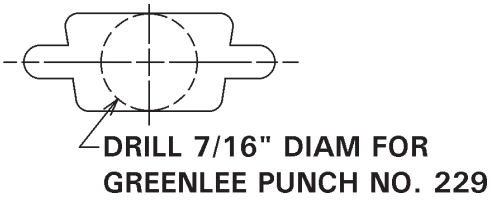


Figure 7-14.
9 Pin D-Subminiature Connector mounting detail (2 places)

7.3.5 KIT ASSEMBLY NO. 600657-004

MODIFICATION TO ADD RS485 TO EXISTING EN(2)1001-LF

PARTS LIST

2	619016-001	PROM, Firmware Version **N**, Programmed for EN1001
2	410347	PCB Assembly, RS485 Driver Bd.
2	525035	Bracket, Contactor & Firing Bd. Mounting
2	331069	Connector, Plug, 5 Pin, Screw Terminal
2	322398	Harness Assem., J4-J4 Ctrl. Bd. 1&2 to RS485 1&2
2	322369-012	Harness Assem., RS485 to Cabinet Connector
4	331124	Connector, Jackscrew, for .090 Panel
25'	900258	Cable, 4 Conductor, 24AWG. w/Shield
2	331136	Connector, Plug, 9 Pin, Screw Terminal
4	557003	6-32 X 1/4 PHSMS, Phil, Brite
4	557006	6-32 X 3/8 PHSMS, Phil, Brite
4	557017	#6 Med. Split Lockwasher, Brite
4	557018	6-32 X 1/4 A/F Hex Nut, Brite
9"	900022	Wire, Stranded, 22AWG, Blu
9"	900213	Wire, Stranded, 22AWG, Wht/Blu
4	342009	Cable Tie, 1-1/4' Dia. Wrap
4	342012	Cable Clamp, Flat Cable Mt.

MODIFICATION NOTES:

1. Remove **ALL** power to control. Open door.

2. Remove existing cable clamps and associated hardware. Move harnesses away from area to be punched, drilled and tapped.

3. Drill 7/16" diameter hole and punch 9 Pin D-Subminiature Connector mounting using Greenlee No. 229 Punch at positions per 9 Pin D-Subminiature Connector mounting detail (see Figures 7-15 and 7-16).

4. Drill and tap per Cabinet modification detail (see Figure 7-15), be sure to vacuum or otherwise remove **ALL** metal chips before assembling Mounting Brackets (P/N 525035) to left side of cabinet. Use 2 ea. 6-32 x 3/8 PHSMS, Phil Brite, from the outside and 2 ea. #6 Med. Split Lockwashers and 2 ea. 6-32 x 1/4 A/F Hex Nuts on the inside.

5. Mount the RS485 PC Board Assembly (P/N 410347) to the Mounting Brackets using 2 ea. 6-32 x 1/4 PHSMS, Phil, Brite (see Figure 7-15).

6. Replace existing PROM 619016-001 in Control Board 600572-002 with new PROM firmware version 619016-001**N** (or later) supplied with this kit.

7.3.5 KIT ASSEMBLY NO. 600657-004 (cont.)

7. Connect J4 Harnesses and complete point-to-point wiring of J485A terminals per Wiring Diagram 421411 included with this set of instructions. Twist point-to-point wires approximately one twist per inch. Install 9 Pin D-Subminiature Connectors and associated harnesses per Wiring Diagram.

8. Secure harnesses to cabinet and Mounting Brackets using supplied cable clamps and wire ties.

9. Close cabinet, reapply power to control.

NOTICE

On the Wiring Diagram, the dark band on connectors indicates stripe on ribbon harness. Harness **MUST** be installed with ribbon harness stripe oriented correctly.

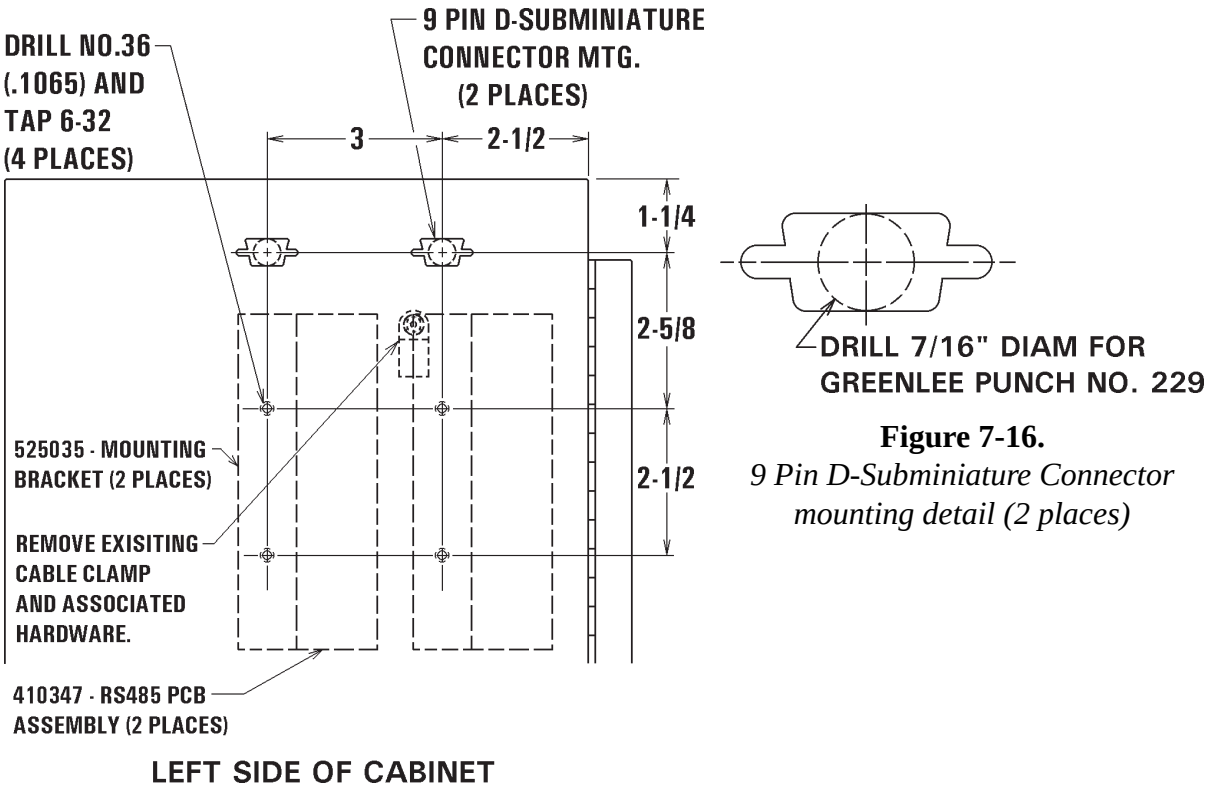


Figure 7-15.
Cabinet modification detail for EN(2)1001-LF

Figure 7-16.
9 Pin D-Subminiature Connector mounting detail (2 places)

7.3.6 ASSEMBLY NO. 600691 Original Revision – U485 CONVERTER FOR USE WITH COMPUTERS WITH USB JACKS (OBSOLETE as of 1/1/08)

PARTS LIST

1	380005	Converter, Isolated, USB to RS485/422
1	331136	Conn., 9 Pin, Screw Terminal, “D” Style Plug
6'	900258	Cable, 4 Cond. 24 Ga. Stranded w/Shield
1	342002	Cable Tie, ½" Diam. Wrap
5/8"	900087	Tubing, Shrink, 3/8"
1	211214	Resistor, Metal Film, 121 Ohm (Desig. R16)
1	460275	Label, Converter, U485
1	460237	Label, “Tested By:”
1	460228	Label, J485
1	326049	Cable, USB-A Male To USB-B Male, 2 Meter
1	460272	Label, Cable, 326049
1	421407	Wiring Diagram, Interconnect Wiring of RS485 Driver Boards

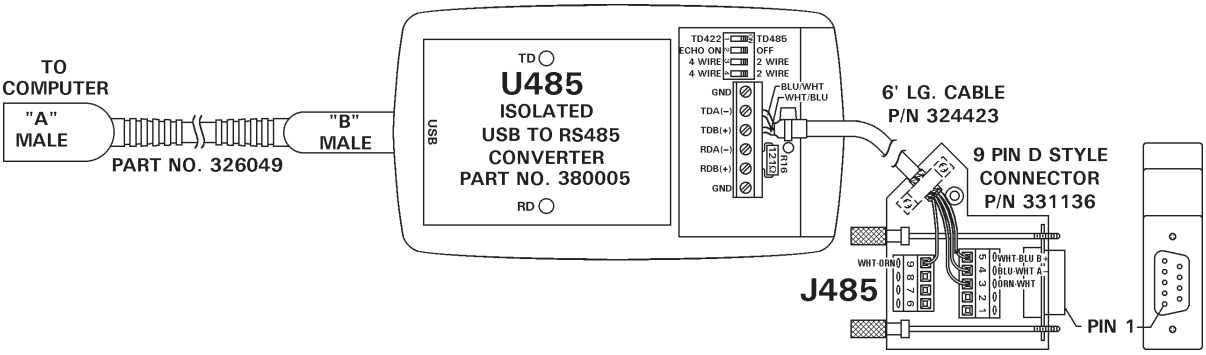


Figure 7-17. *U485 Assembly No. 600691*

INSTALLATION INSTRUCTIONS:

1. Remove **ALL** power to control and computer.

2. Plug 9 Pin D-Sub Connector into RS485 jack on control.

3. Plug USB Type “A” Male Plug into corresponding jack on computer and the USB Type “B” Male Connector into the U485 Converter.

4. Reapply power to computer and install software supplied with USB Converter per instructions in Section 4.3.

5. Install ENLINK or other communication software.

6. Reapply power to control.

OBSOLETE

7.3.7 ASSEMBLY NO. 600691 Revision A – U485 CONVERTER FOR USE WITH COMPUTERS WITH USB JACKS

PARTS LIST

1	380005	Converter; Isolated; USB to RS485/422
1	331136	Connector; 9 Pin; Screw Terminal; “D” Style Plug
2	342018	Cable Tie; 1.75" Diam Wrap
6	345043	Lug; Ferrule; Insulated; 24-22AWG
1	211214	Resistor; Carbon; 121 Ohms; 1/4W; 10%
1"	900005	Tubing; Heat Shrink; .18"
6'	900258	Cable; 4 Cond; 24Ga. Stranded w/Shield
1	345045	Lug; Ferrule; Insulated Dual; 18AWG
1	460228	Label; J485
1	326049	Cable; USB-A Male to USB-B Male; 6 Ft
1	460272	Label; Cable; 326049
2"	900026	Wire; Stranded; 18Ga; Blk
1	342009	Cable Tie; 1.25" Diam Wrap
5/8"	900087	Tubing; Shrink; 3/8"
1	460275	Label; Converter; U485/600691
1"	900237	Tape; Transparent; .75" Wide
1	460237	Label; “Tested By”
1	421407	Wiring Diagram; Interconnect Wiring of RS485 Driver Boards

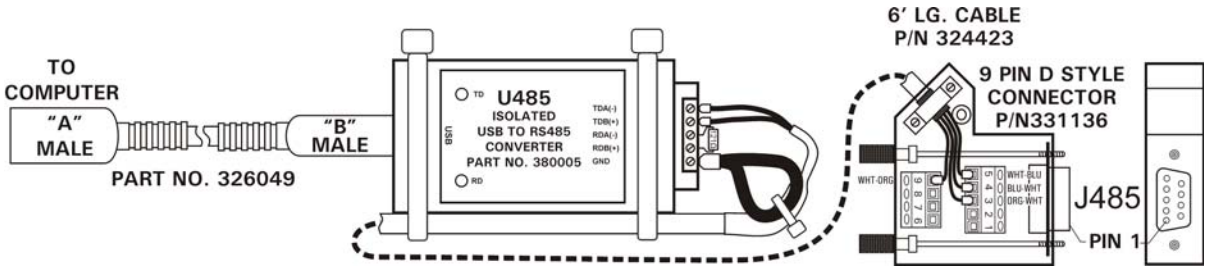


Figure 7-18. *U485 Assembly No. 600691A*

INSTALLATION INSTRUCTIONS:

1. Remove **ALL** power to control and computer.

2. Plug 9 Pin D-Sub Connector into RS485 jack on control.

3. Plug USB Type “A” Male Plug into corresponding jack on computer and the USB Type “B” Male Connector into the U485 Converter.

4. Reapply power to computer and install software supplied with USB Converter per instructions in Section 4.2.

5. Install ENLINK or other communication software.

6. Reapply power to control.

7.3.8 ASSEMBLY NO. 600692 – S485 CONVERTER FOR USE WITH COMPUTERS WITH 9 OR 25 PIN SERIAL BUS

PARTS LIST

1	380006	Converter, Isolated, RS232 to RS485/422
1	331136	Conn., 9 Pin, Screw Terminal, “D” Style Plug
1	380007	Transformer, Power Supply, 110VAC to 12VDC, Wall Plug
6'	900258	Cable, 4 Cond. 24 Ga. Stranded w/Shield
1	342002	Cable Tie, ½” Diam. Wrap
1	460276	Label, Converter, S485
1	460237	Label, “Tested By:”
1	460228	Label, J485
1	326050	Cable, DB25 Male To DB25 Female, 6' Lg.
1	460273	Label, Cable, 326050
1	326051	Cable, DB25 Male To DB9 Female, 6" Lg.
1	460274	Label, Cable, 326051
5/8"	900087	Tubing, Shrink, 3/8"
8	345043	Lug, Ferrule, Insulated, 22-24 Ga.
1	421407	Wiring Diagram, Interconnect Wiring of RS485 Driver Boards

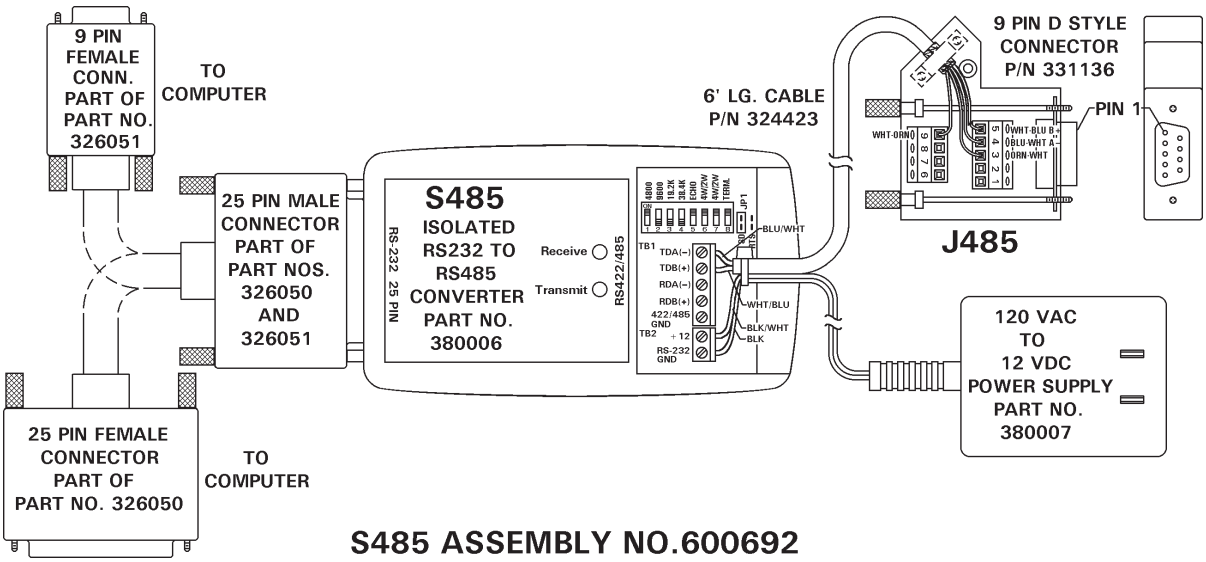


Figure 7-19. *S485 Assembly No. 600692*

INSTALLATION INSTRUCTIONS:

1. Remove **ALL** Power to control and computer.

2. Plug 9 Pin D-Sub Connector into RS485 jack on control.

3. Plug the 25 Pin Female or 9 Pin Female Plug into the corresponding serial jack on computer, plug the opposing 25 Pin Male Plug into the RS485 Converter.

4. Reapply power to computer and install ENLINK or other communication software.

5. Reapply power to control.

7.3.9 MODIFICATION TO ADD RS485 TO EXISTING EN1200 CONTROL IN "T" CABINET – OBSOLETE

See Section 7.3 for Parts List for Kit Assembly 600703-002.

MODIFICATION NOTES:

1. Remove **ALL** power to control. Open door.

2. Drill and tap door per modification detail (see Figure 7-20), be sure to vacuum or otherwise remove **ALL** metal chips before proceeding. Use 2 ea. 6-32 x 1/4 PHSMS, Phil Brite, from the outside and 2 ea. #6 Med. Split Lockwashers and 2 ea. 6-32 x 1/2 hex threaded Brass Spacers on the inside.

3. Mount the RS485 PC Board Assembly (P/N 410347) to the hex threaded Brass Spacers using 2 ea. 6-32 x 1/4 PHSMS, Phil, Brite. See Option mounting detail (Figure 7-21).

4. Replace existing PROM 619024-002 in Control Board 600606 with new PROM firmware version 619024-002**E** (or later) supplied with this kit.

5. Connect J4 Harness and complete point-to-point wiring of J485A terminals per Wiring Diagram 421435-001 included with this set of instructions. Twist point-to-point wires approximately one twist per inch. Install 9 Pin D-subminiature Connectors and associated harnesses per Wiring Diagram.

6. Secure harnesses to cabinet and Mounting Brackets using supplied cable clamps and wire ties.

7. Close cabinet, reapply power to control.

NOTICE

On the Wiring Diagram, the dark band on connectors indicates stripe on ribbon harness. Harness **MUST** be installed with ribbon harness stripe oriented correctly.

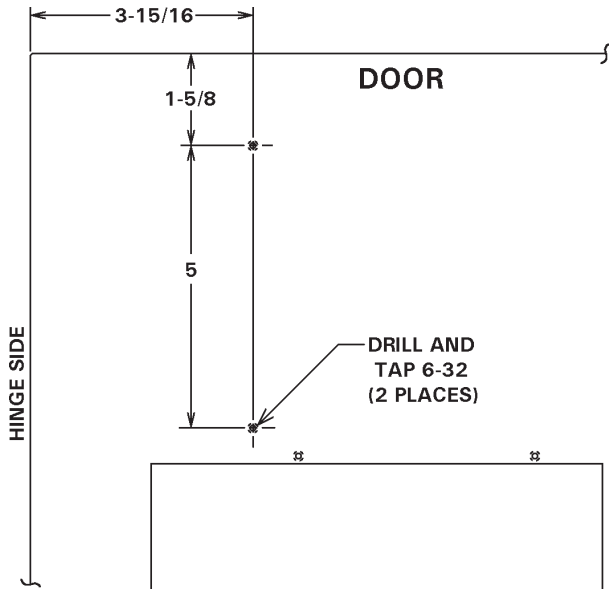


Figure 7-20. *Door modification detail*

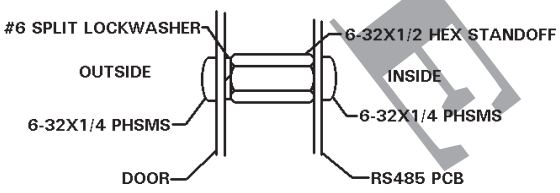


Figure 7-21. *Option mounting detail*

7.3.10 MODIFICATION TO ADD RS485 TO EXISTING EN1000/EN1001 CASCADE CONTROL IN "T" CABINET

See Section 7.3 for Parts List for Kit Assemblies 600658-002.

MODIFICATION NOTES:

1. Remove **ALL** power to control. Open door.

2. Remove existing cable clamps and associated hardware. Move harnesses away from area to be punched, drilled and tapped.

3. On left side of cabinet, assemble Mounting Bracket (P/N 525035) to existing standoffs shown in Figure 7-22. Use 2 ea. 6-32 x 3/8 PHSMS, Phil Brite, from the outside and 2 ea. #6 Med. Split Lockwashers and 2 ea. 6-32 x 1/4 A/F Hex Nuts on the inside.

4. Mount the RS485 PC Board Assembly (P/N 410347) to the Mounting Brackets using 2 ea. 6-32 x 1/4 PHSMS, Phil, Brite (see Figure 7-22).

5. Replace existing PROM in Control Board 410321 with new PROM firmware version 619044-002 (**ORIG** rev or later) supplied with this kit.

6. Connect J4 Harnesses and complete point-to-point wiring of J485A terminals per Wiring Diagram 421499-001 included with this set of instructions. Twist point-to-point wires approximately one twist per inch. Install 9 Pin D-Subminiature Connectors and associated harnesses per Wiring Diagram.

7. Remove existing 4 ea. #6-32 Screws and 2 ea. cover plates and discard. Secure harnesses to cabinet and Mounting Brackets using supplied cable clamps and wire ties and using existing 9 Pin D-Subminiature Connector holes.

8. Close cabinet, reapply power to control.

NOTICE

On the Wiring Diagram, the dark band on connectors indicates stripe on ribbon harness. Harness **MUST** be installed with ribbon harness stripe oriented correctly.

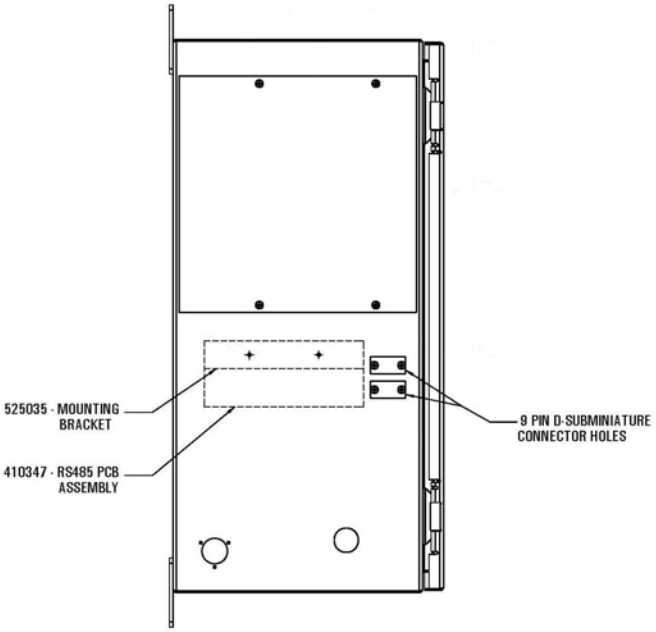


Figure 7-22.
Modification detail for primary location of RS485 Option for EN1000/1001 Cascade in "T" Cabinet

7.3.10 MODIFICATION TO ADD RS485 TO EXISTING EN1000/EN1001 CASCADE CONTROL IN "T" CABINET (cont.)

If primary location of RS485 Option is being used by another option, use secondary mounting location on bottom of cabinet as shown in Figure 7-23.

Figure 7-23.
Secondary location of RS485 Option for EN1000/1001 Cascade in "T" Cabinet

7.4 LIST OF WIRING DIAGRAMS (available as of 5-14)

421180-008 W/D, EN1000/485-Series “S” Cabinet

421210-027 W/D, EN1000/485-Series “D/T” Cabinet
421210-028 W/D, EN1000-FP(SCR)/RDH/485

421214-043 W/D, EN1000/S99/485-(1-8)Series Cascade
421214-047 W/D, EN1000-(1-8)(SCR)/485 Cascade
421214-048 W/D, EN1000/VE/485-(1-8)Series Cascade
421214-050 W/D, EN1000/485-(1-8)Series Cascade

421268-010 W/D, EN1001/S49/485-Series “E” Cabinet
421268-011 W/D, EN1001/485-Series “E” Cabinet

421269-014 W/D, EN1001/485-Series “D/T” Cabinet
421269-019 W/D, EN1001-FP(SCR)/485
421269-031 W/D, EN1001-Series/485/SP “D/T” Cabinet, SP=24VDC Valves/Power Supply
421269-035 W/D, EN1001-Series/485/SP “D/T/LS/LF” Cabinet, SP=24VDC Valves/Power Supply; 380 VAC

421270-007 W/D, EN1001/485-Series “S” Cabinet

421377-001 W/D, EN1000(TGA)/GF/485-Series “D/T” Cabinet
421377-006 W/D, EN1001/GF/485-Series “D/T” Cabinet

421406 W/D, Remote Terminal 4 (RT4) – OBSOLETE

421407 W/D, Interconnect Wiring of RS485 Driver Boards

421410 W/D, EN1001 RDH Demo Control with RS485

421411 W/D, EN(2)1001/485-600/1200 “LF” Cabinet

421412 W/D, EN(4)1001/485-600/1200 “HF” Cabinet

421413 W/D, EN(6)1001/485-600/1200 “UF” Cabinet

421423-002 W/D, EN1001-IMU(SCR)/485

421433-002 W/D, EN1200-200/485 “A” Cabinet – OBSOLETE

421435-001 W/D, EN1200/485-Series “T” Cabinet – OBSOLETE
421435-002 W/D, EN1200/S49/485-Series “T” Cabinet – OBSOLETE
421435-004 W/D, EN1200/485-Series “T” Cabinet w/24VDC Valves – OBSOLETE

421438-008 W/D, EN1001-(1-8)Series/485 Cascade
421438-013 W/D, EN1001/VE/485-(1-8)Series/SP Cascade, SP=J4B for MM8

421447-002 W/D, EN1001/485-Series “B” Cabinet

421464-001 W/D, EN1201/485-Series “T” Cabinet – OBSOLETE

421472 W/D, RT4jr.

421477 W/D, EN(2)1001-1200(2)/485 “GF” Cabinet

421499-001 W/D, EN1001/485, “C” Cabinet

7.5 RAM MEMORY MAP

In EN1000 Series Weld Controls, EEPROM Memory is used for non-volatile storage all SCHEDULE and EXTENDED FUNCTIONS data. Before executing any weld sequence, all SCHEDULE and EXTENDED FUNCTIONS data are loaded into RAM memory. The RAM memory map may be useful when an RS232 or RS485 option is used to read SCHEDULE data, EXTENDED FUNCTIONS data or other useful flags or data from PC or PLC.

To read data from RAM memory, data location (i.e., address of the data) must be known.

7.5.1 FLAGS

The bytes from **0x20** to **0x25** are bit addressable and contain different status flags.

==========

```
at 0x20 bdata byte flags0;          //DATA @ 20h - Status flags
sbit flag_wnw = flags0 ^ 0;         //WELD/NoWELD push button status
sbit flag_program = flags0 ^ 1;     //PROGRAM/OPERATE push button status
sbit flag_PIN_lock = flags0 ^ 2;    //PIN lock enable/disable flag
sbit flag_sw_lock = flags0 ^ 3;     //PLK switch flag
sbit flag_timer = flags0 ^ 4;
sbit flag_valve = flags0 ^ 5;
sbit flag_sync = flags0 ^ 6;
sbit flag_slope = flags0 ^ 7;
```

```
at 0x21 bdata byte flags1;          //DATA @ 21h - Status flags
sbit flag_estop = flags1 ^ 0;       //Emergency STOP status
sbit flag_i2c_busy = flags1 ^ 1;    //i2c bus busy flag
sbit first_i2c_error = flags1 ^ 2;  //i2c error flag
sbit first_weld = flags1 ^ 3;       //first weld flag
sbit flag_PO_07 = flags1 ^ 4;
sbit pf_zero = flags1 ^ 5;
sbit wnw_pressed = flags1 ^ 6;
sbit flag_av = flags1 ^ 7;
```

7.5.1 FLAGS (cont.)

```
at 0x22 bdata byte flags2;          //DATA @ 22h - Status flags
sbit flag_IPSEr = flags2 ^ 0;       //the same functionality as PSW1: flag_IPSEr=1->PCerror
sbit flag_avc=flags2^1;             //
sbit flag_adc = flags2 ^ 2;         //
sbit flag_short = flags2 ^ 3;       //
sbit flag_RS232 = flags2 ^ 4;       //RS232 active flag
sbit flag_tx_active=flags2 ^ 5;     //UART TxD active
sbit flag_msg_ack = flags2 ^ 6;     //master UART has red message if this flag is set
sbit flag_msg_recv=flags2 ^ 7;      //Message (or byte) received on RxD
_____________________________

at 0x23 bdata byte flags3;          //DATA @ 23h -> STATUS byte: type of the Control
sbit flag_IPC = flags3 ^ 0;         //IPC=Pressure Control option
sbit flag_IPS = flags3 ^ 1;         //IPS=Pressure Sense option
sbit          flags3 ^ 2;           //reserved
sbit          flags3 ^ 3;           //reserved
sbit flag_EN1000C = flags3^4;       //EN1000Cascade
sbit flag_EN1200=flags3 ^ 5;        //EN1200
sbit flag_EN1001=flags3 ^ 6;        //if set->EN1001, if cleared->EN1000
sbit flag_ENRT = flags3 ^ 7;
_____________________________

at 0x24 bdata byte sws;             //DATA @ 24h - Status for all pushbuttons
sbit sw_program = sws ^ 0;          //PROGRAM/OPERATE button pressed if this flag is zero
sbit sw_enter = sws ^ 1;            //ENTER push button pressed if this flag is zero
sbit sw_weld = sws ^ 2;             //WELD/No WELD push button pressed if this flag is zero
sbit sw_select = sws ^ 3;           //SELECT push button pressed if this flag is zero
sbit sw_data1 = sws ^ 4;            //Right DATA push button pressed if this flag is zero
sbit sw_data10 = sws ^ 5;           //Left DATA push button pressed if this flag is zero
sbit sw_schedule1 = sws ^ 6;        //Right SCHEDULE push button pressed if this flag is zero
sbit sw_schedule10 = sws ^ 7;       //Left SCHEDULE push button pressed if this flag is zero
_____________________________

at 0x25 data byte flags4;           //DATA @ 25h
sbit flag_overcur = flags4 ^ 0;     //reserved
sbit flag_S49 = flags4 ^ 1;         //S49 option status: =1 =>S49 board exists
sbit flag_welddone = flags4^2;      //=1 if Weld mode, WNW swich closed and WELD time>00
sbit flag_PLK_en = flags4 ^ 3;      //enable/disable PLK switch: =1 PLK enabled
sbit flag_sekvencer = flags4^4;     //reserved
sbit flag_reset_range = flags4 ^ 5; //temporary bit
=====================================================================
```

7.5.2 DATA from 0x26 to 0x3F

The bytes from **0x26** to **0x3F** are different status bytes.

```
at 0x26 data byte ;                     //reserved
at 0x27 data byte first_sched;          //temporary byte
at 0x28 data byte z_low;                //temporary counter
at 0x29 data byte z_high;               //temporary counter
at 0x2a data byte SEQptr;               //Schedule Pointer
at 0x2b data byte EFptr;                //Extended Function pointer
at 0x2c data byte led_data;             //Data (two digit DATA display)
at 0x2d data byte led_sched;            //Schedule number (SCHEDULE Display)
at 0x2e data byte *i2c_xptr;            //pointer to IIC MasterTransmitData
at 0x2f data byte *i2c_rptr;            //pointer to IIC MasterReceiveData
at 0x30 data struct ByteLed_a6 Led;     //Structure for DATA and SCHEDULE displays
    //at 0x30 data byte a6_write;       //Led.inst
    //at 0x31 data byte a6_write1;      //Led.ctrl
    //at 0x32 data byte a6_write2;      //Led.DATA10
    //at 0x33 data byte a6_write3;      //Led.DATA1
    //at 0x34 data byte a6_write4;      //Led.SC10
    //at 0x35 data byte a6_write5;      //Led.SC1
at 0x36 data struct ByteLed_a19 Leds;   //Structure for two bytes LED display
    //at 0x36 data byte a19_write;      //Leds.inst
    //at 0x37 data byte a19_write1;     //Leds.ctrl
    //at 0x38 data byte a19_write2;     //Leds.LED1
    //at 0x39 data byte a19_write3;     //Leds.DATA10L
    //at 0x3a data byte a19_write4;     //Leds.LED2
    //at 0x3b data byte a19_write5;     //Leds.DATA1R
at 0x3c data byte Timer;                //0.5 second Timer
at 0x3d data byte a7_read;              //temporary byte for EEPROM reading
at 0x3e data byte a7_write;             //temporary byte for EEPROM writing
at 0x3f data byte a7_write1;            //temporary byte for EEPROM writing
=================================================================
```

7.5.3 SCHEDULE DATA

The bytes from **0x40** to **0x4F** are SCHEDULE data bytes. Each schedule has 16 bytes (in CONSTANT CURRENT mode, there are three bytes more), with sequence pointer from 00 to 15 (0x0F) as follows:

```
at 0x40 data byte cur_squeeze;          //00. SQ - SQUEEZE
at 0x41 data byte cur_weld;             //01. We - WELD
at 0x42 data byte cur_percent;          //02. CU - PERCENT CURRENT
at 0x43 data byte cur_hold;             //03. HO - HOLD
at 0x44 data byte cur_off;              //04. OF - OFF
at 0x45 data byte cur_impulses;         //05. IM - IMPULSES
at 0x46 data byte cur_cool;             //06. CL - COOL
at 0x47 data byte cur_valve;            //07. VM - VALVE MODE
at 0x48 data byte cur_cycle;            //08. CM - CYCLE MODE
at 0x49 data byte cur_slope;            //09. SM - SLOPE MODE
at 0x4a data byte cur_slcount;          //0A. SC - SLOPE COUNT
at 0x4b data byte pressure_ref;         //0B. Pr - Pressure reference
at 0x4c data byte pressure_trg;         //0C. Pt - Pressure trigger
at 0x4d data union IntData cur_stc;     //0D. StCnt - Stepper counter (one integer)
    //at 0x4d data byte cur_stc_h;      //0D cur_stc.b.hi \  cur_stc.w
    //at 0x4e data byte cur_stc_l;      //0E cur_stc.b.lo /
at 0x4f data byte cur_per_base;         //0F. Cb - Base value for Current Offset
=================================================================
```

7.5.4 EXTENDED FUNCTIONS DATA

The bytes from **0x50** to **0x64** and **0xea** are EXTENDED FUNCTIONS data as follows:

```
at 0x50 data byte cur_id;               //00. I.d. - ID number of the control
at 0x51 data byte cur_se;               //01. S.E. - Seam
at 0x52 data byte cur_ss;               //02. S.S. - Schedule Select
at 0x53 data byte cur_cc;               //03. C.C. - Automatic Voltage Compensation
at 0x54 data byte cur_ca;               //04. C.A. - Clear All / Setup Constant Current
at 0x55 data byte cur_bs;               //05. b.S. - Back Step
at 0x56 data byte cur_po;               //06. P.O. - Process Output
at 0x57 data byte cur_be;               //07. b.E. - Beat/Non Beat
at 0x58 data byte cur_87;               //08. 8.7. - 87 Degree Delay
at 0x59 data byte cur_pp;               //09. P.P. - Programmable Power Factor
at 0x5a data byte cur_pf;               //0A. P.F. - Power Factor Measuring
at 0x5b data byte cur_sd;               //0B. S.d. - Squeeze Delay
at 0x5c data byte cur_bl;               //0C. b.L. - Blocking Delay
at 0x5d data byte cur_cr;               //0D. C.r. - Current Regulation
at 0x5e data union IntData cur_ra;      //0E. r.A. - Range
    //at 0x5e data byte cur_ra_h;       //cur_ra.b.hi \ cur_ra.w;     0E. r.A. - First Byte
    //at 0x5f data byte cur_ra_l;       //cur_ra.b.lo /               0F. r.A. - Second Byte
at 0x60 data byte cur_co;               //10. C.O. - Current Offset
at 0x61 data byte cur_st;               //11. S.t. - Stepper
at 0x62 data byte cur_pc;               //12. P.C. - Pressure Control
at 0x63 data byte cur_bd;               //13. b.d. - Background(Return)Pressure
at 0x64 data byte cur_si;               //14. S.I. - Sensor Input(reading)
at 0xea idata byte turns_ratio;         //19. t.r. - Turns Ratio of transformer
=================================================================
```

7.5.5 DATA from 0x65 to 0xFF

```
at 0x65 data byte ;                 //reserved
at 0x66 data byte temp_percent;     //temporary byte
at 0x67 data union IntData cur_cu;  //cc mode: current in [%] or [kA*100]
    //at 0x67 data byte cur_cu.b.hi; //cur_cu high byte
    //at 0x68 data byte cur_cu.b.lo; //cur_cu low byte
at 0x69 data byte last_percent;     //temporary byte
at 0x6a data byte p1_shadow;        //temporary byte
at 0x6b data byte led_datab;        //temporary byte
at 0x6c data byte led_datab_next;   //temporary byte
at 0x6d data byte rms2dc_ref;       //desired current, scaled from 0x00 to 0xff
at 0x6e data byte rms2dc;           //measured current from A/D converter
at 0x6f data byte rms2dc_old;       //temporary byte
_____________________________

at 0x70 data byte t_low;            //temporary byte
at 0x71 data byte t_high;           //temporary byte
at 0x72 data byte temp;             //temporary byte
at 0x73 data byte temp1;            //temporary byte
at 0x74 data union IntData TmpRMS; //Temporary Integer
    //at 0x74 data byte TmpRMS_h; //TmpRMS.b.hi \  TmpRMS.w
    //at 0x75 data byte TmpRMS_l; //TmpRMS.b.lo /
at 0x76 data byte load_16byte;      //temporary byte
at 0x77 data byte load_16byte1;     //temporary byte
at 0x78 data byte acc32_3;          //temporary byte
at 0x79 data byte acc32_2;          //temporary byte
at 0x7a data byte acc32_1;          //temporary byte
at 0x7b data byte acc32_0;          //temporary byte
at 0x7c data byte tmp_3;            //temporary byte
at 0x7d data byte tmp_2;            //temporary byte
at 0x7e data byte tmp_1;            //temporary byte
at 0x7f data byte tmp_0;            //temporary byte
=================================================================

at 0x80 idata byte stack;           //IDATA from 80h to BFh - Stack
- - -
at 0xac idata byte integral_3;      //   \
at 0xad idata byte integral_2;      //    | Temporary bytes for  SPOT subroutines
at 0xae idata byte integral_1;      //    |
at 0xaf idata byte integral_0;      //   /
- - -
at 0xb0 idata byte OutBuf[OutBufLen]; //output buffer from B0-C4 = 21 bytes
_____________________________
```

7.5.5 DATA from 0x65 to 0xFF (cont.)

```
at 0xc5 idata byte ;                    //reserved
at 0xc6 idata byte chars4Tx;            //number of characters for Transmit
at 0xc7 idata byte ScanCode;            //Control status: =0->Control ready; =0xF1 error, ...
at 0xc8 idata byte ErrorNo;             //Error Number: =0->no errors; =01...99->error no.
#define InpBufLen 23                    //UART input RX buffer length
at 0xc9 idata byte InpBuf[InpBufLen];   //input buffer from C9-DF = 23 bytes
_____________________________

at 0xe0 idata byte setup_cu;            //temporary byte
at 0xe1 idata byte setup_ctr;           //temporary byte
at 0xe2 idata byte sample_ctr;          //auxiliary counter
at 0xe3 idata byte samp_ptr;            //to store number of samples acquired
at 0xe4 idata byte fa_adjust;           //temporary byte
at 0xe5 idata byte short_count;         //temporary counter
at 0xe6 idata byte last_duty;           //temporary byte
at 0xe7 idata byte adc_18V;             //A/D value of 18 volts power supply
at 0xe8 idata byte adc_18V_cnt;         //temporary byte
at 0xe9 idata byte trend_up;            //temporary byte
at 0xea idata byte turns_ratio;         //Turns Ratio of transformer

at 0xeb idata byte max_dc;              //full range voltage for A/D converter
at 0xec idata byte gain_1;              //gain setting of amplifier
at 0xed idata byte gain_0;              //gain setting of amplifier
at 0xee idata byte gain_ptr;            //temporary byte
at 0xef idata byte rms2dc_rA;           //temporary byte
_____________________________

at 0xf0 idata byte rms2dc_refLO;        //temporary byte for current monitor
at 0xf1 idata byte rms2dc_refHI;        //temporary byte for current monitor
at 0xf2 idata byte current_mask;        //temporary byte for current control
at 0xf3 idata byte last_ss;             //temporary byte for schedule selection
at 0xf4 idata byte avc_count;           //temporary byte for AVC
at 0xf5 idata byte avc_value;           //temporary byte for AVC
at 0xf6 idata byte avc_value_nom;       //temporary byte for AVC
at 0xf7 idata byte avc_slope;           //temporary byte for AVC
at 0xf8 idata byte OutPointer;          //pointer to RS485 Transmit Data
at 0xf9 idata byte InpPointer;          //pointer to RS485 Receive Data
at 0xfa idata union IntData TmpInt;     //temporary integer
    //at 0xfa idata byte TmpInt_h;      //TmpInt.b.hi \  TmpInt.w
    //at 0xfb idata byte TmpInt_l;      //TmpInt.b.lo /
at 0xfc idata union IntData led_data4;  //temporary integer
    //at 0xfc idata led_data4+1;        //led_data4.b.hi  \ led_data4.w
    //at 0xfd idata led_data4+1;        //led_data4.b.lo  /
at 0xfe idata byte a8_read;             //temporary byte
at 0xff idata byte a8_write;            //temporary byte
=================================================================
```

7.6 EEPROM MEMORY MAP

In EN1000 Series Weld Controls, EEPROM Memory is used for non-volatile storage all SCHEDULE and EXTENDED FUNCTIONS data. Memory Map may be useful when an RS232 or RS485 option is used to program SCHEDULE and EXTENDED FUNCTIONS data from PC or PLC.

To read data from EEPROM or write data to EEPROM memory, data location (i.e., EEPROM page and EEPROM word address of the data) must be known.

7.6.1 SCHEDULE DATA

The pages **A0** to **A6** contain SCHEDULE data. EEPROM **page** for any SCHEDULE data can be calculated by using following equation:

$$EEPROM\ page\ [hex] = A0 + 2*\text{int}(\frac{SN}{16})$$

EEPROM word **address** for any SCHEDULE data can be calculated by using following equation:

$$EEPROM\ address\ [hex] = SN * 16 + SP$$

where: *SN* is Schedule Number
SP is Sequence Pointer

Each schedule is contained in a line of 16 bytes, with *SP* from 0 to 15 (0x0F), as follows:

```
00. SQ    - Squeeze
01. We    - Weld
02. CU    - Percent Current
03. HO    - HOLD
04. OF    - OFF
05. IM    - IMPULSES
06. CL    - COOL
07. VM    - Valve Mode
08. CM    - Cycle Mode
09. SM    - Slope Mode
0A. SC    - Slope Count
0B. Pr    - Pressure (Squeeze Pressure)
0C. Pt    - Pressure trigger (Level of Pressure to begin welding)
0D. StCnt - Stepper counter (two bytes, one integer)
0F. Cb    - Base value for Current Offset use
```

7.6.1 SCHEDULE DATA (cont.)

PAGE A0 hex - SCHEDULE data for Schedule #00 to Schedule #15

SN	Add	SQ	WE	CU	HO	OF	IM	CL	VM	CM	SM	SC	Pr	Pt	StCnt	Cb
		0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
00.	**00**	0A	04	32	0A	00	01	00	01	00	00	00	4E	41	00	00	00
01.	**10**	0A	03	50	0A	00	01	00	01	03	01	0A	4E	41	00	00	00
02.	**20**	0A	03	50	0A	00	01	00	02	00	02	0A	4E	41	00	00	00
03.	**30**	0A	03	14	00	00	01	00	03	02	00	00	4E	41	00	00	00
04.	**40**	1E	14	1E	00	00	01	00	03	01	00	00	4E	41	00	00	00
05.	**50**	0A	03	3C	0A	00	01	00	01	00	00	00	55	4B	00	00	00
06.	**60**	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
07.	**70**	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
08.	**80**	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
09.	**90**	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
10.	**A0**	0A	01	46	00	00	03	02	04	02	00	00	4E	41	01	10	00
11.	**B0**	00	3C	14	0A	00	01	00	07	00	00	00	00	00	00	00	00
12.	**C0**	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
13.	**D0**	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
14.	**E0**	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
15.	**F0**	0A	03	3C	0A	00	01	00	01	00	00	00	4E	41	00	00	00

SN	Add	SQ	WE	CU	HO	OF	IM	CL	VM	CM	SM	SC	Pr	Pt	StCnt	Cb

PAGE A2 hex - SCHEDULE data for Schedule #16 to Schedule #31

SN	Add	SQ	WE	CU	HO	OF	IM	CL	VM	CM	SM	SC	Pr	Pt	StCnt	Cb
		0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
16.	**00**	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
17.	**10**	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
18.	**20**	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
19.	**30**	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
20.	**40**	0A	05	3C	0A	0A	01	00	01	01	00	00	4E	41	02	20	00
21.	**50**	14	0A	46	14	00	06	05	02	03	00	00	4E	41	00	00	00
22.	**60**	14	0A	14	00	00	01	00	04	02	00	00	4E	41	00	00	00
23.	**70**	00	00	0A	28	00	01	00	04	02	00	00	00	00	00	00	00
24.	**80**	00	00	0A	3C	00	01	00	04	02	00	00	00	00	00	00	00
25.	**90**	0A	03	3C	0A	00	01	00	01	00	00	00	32	28	00	00	00
26.	**A0**	14	01	5A	0A	00	1E	01	07	01	00	00	32	28	00	00	00
27.	**B0**	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
28.	**C0**	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
29.	**D0**	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
30.	**E0**	0A	03	3C	0A	00	01	00	01	00	00	00	00	00	03	30	00
31.	**F0**	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00

SN	Add	SQ	WE	CU	HO	OF	IM	CL	VM	CM	SM	SC	Pr	Pt	StCnt	Cb

7.6.1 SCHEDULE DATA (cont.)

```
============
```
PAGE A4 hex - SCHEDULE data for Schedule #32 to Schedule #47
```
============
SN  Add  SQ WE CU HO OF IM CL VM  CM SM SC Pr Pt StCnt Cb
          0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
32.  00  00 00 00 00 00 01 00 00  00 00 00 00 00 00 00 00
33.  10  00 00 00 00 00 01 00 00  00 00 00 00 00 00 00 00
34.  20  00 00 00 00 00 01 00 00  00 00 00 00 00 00 00 00
35.  30  0A 03 3C 0A 00 01 00 01  00 00 00 00 00 00 00 00
36.  40  00 00 00 00 00 01 00 00  00 00 00 00 00 00 00 00
37.  50  00 00 00 00 00 01 00 00  00 00 00 00 00 00 00 00
38.  60  00 00 00 00 00 01 00 00  00 00 00 00 00 00 00 00
39.  70  00 00 00 00 00 01 00 00  00 00 00 00 00 00 00 00
40.  80  0A 03 3C 0A 00 01 00 01  00 00 00 00 00 04 40 00
41.  90  00 00 00 00 00 01 00 00  00 00 00 00 00 00 00 00
42.  A0  00 00 00 00 00 01 00 00  00 00 00 00 00 00 00 00
43.  B0  00 00 00 00 00 01 00 00  00 00 00 00 00 00 00 00
44.  C0  00 00 00 00 00 01 00 00  00 00 00 00 00 00 00 00
45.  D0  0A 03 3C 0A 00 01 00 01  00 00 00 00 00 00 00 00
46.  E0  00 00 00 00 00 01 00 00  00 00 00 00 00 00 00 00
47.  F0  00 00 00 00 00 01 00 00  00 00 00 00 00 00 00 00
-------------------------------------------------------
SN  Add  SQ WE CU HO OF IM CL VM  CM SM SC Pr Pt StCnt Cb
-------------------------------------------------------

============
```
PAGE A6 hex - SCHEDULE data for Schedule #48 and #49; and EXTENDED FUNCTIONS data
```
============
SN  Add  SQ WE CU HO OF IM CL VM  CM SM SC Pr Pt StCnt Cb
          0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
48.  00  00 00 00 00 00 01 00 00  00 00 00 00 00 00 00 00
49.  10  1E 04 3C 0A 00 01 00 01  01 00 00 00 00 00 00 00
     20  FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
     30  FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
     40  FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
     50  FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
     60  FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
     70  FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
     80  FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
     90  FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
     A0  FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
     B0  FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
     C0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
     D0  00 00 00 0A 00 00 14 00  00 1E 00 00 28 00 00 00
EF   E0  01 00 00 00 00 00 00 00  01 00 00 00 00 00 00 63
EF   F0  00 00 00 00 FF FF FF FF  FF 30 FF FF FF FF 4D 00
-------------------------------------------------------
EXTENDED FUNCTIONS (EF) data (starting from address 0xE0):
         Id SE SS CC CA bS PO bE  87 PP PF Sd bL Cr rA rA
         CO St PC bd SI              tr             WN LS
-------------------------------------------------------
```

7.6.2 SCHEDULE DATA EXAMPLES

Schedule 20, EEPROM page is **0xA2**, has following EEPROM addresses and data:

	SQUEEZE	WELD	PERCENT	HOLD	OFF	IMPULS	COOL	ValveM	CycleM	SlopeM	SlopeC	Pr	Pt	StepperC	% base
addr [*hex*]	0x40	0x41	0x42	0x43	0x44	0x45	0x46	0x47	0x48	0x49	0x4A	0x4B	0x4C	0x4D	0x4F
data [*hex*]	0x0A	0x05	0x3C	0x0A	0x0A	0x01	0x00	0x01	0x01	0x00	0x00	0x4E	0x41	0x0220	0x00
data	10	05	60	10	10	01	00	01	01	00	00	78	65	544	00

7.6.3 SCHEDULE DATA RANGES

	SQUEEZE	WELD	PERCENT	HOLD	OFF	IMPULS	COOL	ValveM	CycleM	SlopeM	SlopeC	Pr	Pt	StepperC	% base
MIN [*hex*]	0x00	0x00	0x00	0x00	0x00	0x01	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
MAX [*hex*]	0x63	0x63	0x63	0x63	0x63	0x63	0x63	0x07	0x04	0x02	0x63	0xFF	0xFF	0x27FF	0x63
MAX	99	99	99	99	99	99	99	07	04	02	99	255	255	9999	99

7.6.4 EXTENDED FUNCTIONS DATA

The page **A6** also holds the EXTENDED FUNCTIONS data. EEPROM word **address** for any EXTENDED FUNCTIONS data can be calculated by using following equation:

$$EEPROM\ \ addressEF\ [hex] = 0xE0 + EP$$

where: *0xE0* is hex value of first address for EXTENDED FUNCTIONS data
EP is EXTENDED FUNCTIONS Pointer

These data start at Word address 0xE0, with *EP* from 0 to 20 (0x14), as follows:

```
00.  Id - ID number of the control
01.  SE - Seam
02.  SS - Schedule Select
03.  CC - Automatic Voltage Compensation
04.  CA - Clear All / Setup Constant Current
05.  bS - Back Step
06.  PO - Process Output
07.  bE - Beat/Non Beat
08.  87 - 87 Degree Delay
09.  PP - Programmable Power Factor
0A.  PF - Power Factor Measuring
0B.  Sd - Squeeze Delay
0C.  bL - Blocking Delay
0D.  Cr - Current Regulation
0E.  rA - Range (Current capacity) First Byte
0F.  rA - Range (Current capacity) Second Byte
```

7.6.4 EXTENDED FUNCTIONS DATA (cont.)

10. CO - Current Offset
11. St - Stepper
12. PC - Pressure Control
13. bd - Background(Return)Pressure
14. SI - Sensor Input(reading)
19. tr - Turns Ratio of transformer

7.6.5 EXTENDED FUNCTIONS DATA RANGES

	I.d.	S.E.	S.S.	C.C.	C.A.	b.S.	P.O.	b.E.	87	P.P.	P.F.	S.d.	b.L.
MIN [*hex*]	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
MAX [*hex*]	0x63	0x63	0x03	0x1F	0x0B	0x0A	0x27	0x03	0x01	0x99	–	0x63	0x63
MAX	99	99	03	31	11	10	39	03	01	99	–	99	99

	C.r.	r.A.	C.O.	S.t.	P.C.	b.d.	S.I.	t.r
MIN [*hex*]	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x0A
MAX [*hex*]	0x21	0x270F	0x27	0x01	0x06	0xFF	–	0xFF
MAX	33	9999	39	01	06	255	–	255

7.6.6 OTHER DATA

==========

PAGE A8 hex - Calibration data. Should not be accessed.

==========

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
00	40	06	05	0D	40	08	09	20	40	0C	0B	0E	40	0D	10	20
10	40	40	0A	0A	40	40	0E	0C	40	40	12	0E	40	40	26	1D
20	40	40	40	0A	40	40	40	0E	40	40	40	12	40	40	40	17
30	40	40	40	1C	40	40	40	22	40	40	40	29	40	40	40	30
40	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
50	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
60	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
70	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
80	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
90	E4	E5	EB	EB	EA	E9	EA	EB	E3	E4	E8	E7	E9	E9	E9	EB
A0	E4	E5	EB	EB	EA	E9	EA	EB	E3	E4	E8	E7	E9	E9	E9	EB
B0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
C0	40	06	05	0D	40	08	09	20	40	0C	0B	0E	40	0D	10	20
D0	40	40	0A	0A	40	40	0E	0C	40	40	12	0E	40	40	26	1D
E0	40	40	40	0A	40	40	40	0E	40	40	40	12	40	40	40	17
F0	40	40	40	1C	40	40	40	22	40	40	40	29	40	40	40	30

7.6.6 OTHER DATA (cont.)

```
============
PAGE AA hex - empty page
============
     0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
00   FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
10   FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
20   FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
30   FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
40   FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
50   FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
60   FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
70   FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
80   FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
90   FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
A0   FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
B0   FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
C0   FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
D0   FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
E0   FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
F0   FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
-------------------------------------------------
```

The last two pages **AC** and **AE** contain CONSTANT CURRENT data for EN1001 Weld Controls. These data are stored in groups of 9 bytes per schedule. Schedules from 00 to 24 are stored in EEPROM page 0xAC, and Schedules from 25 to 49 in page 0xAE. For example, for Schedule 00:

- Current is stored in first two locations; i.e., address 0x00 and 0x01; value is 60%;
- ***L.o.*** (Low limit) correspond to two bytes at addresses 0x03 and 0x04; value is 55%;
- ***H. i.*** (High limit) correspond to two bytes at addresses 0x06 and 0x07; value is 65%.

The first byte is the MSB and the second the LSB byte; i.e., they are stored as two bytes integer type. Third byte is calculated internally by the control after programming first two bytes.

```
============
PAGE AC hex - CU, Lo, Hi data for CONSTANT CURRENT mode
============
     0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
00   00 3C 00 00 37 00 00 41  00 00 00 00 00 00 00 00
10   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
20   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
30   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
40   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
50   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
60   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
70   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
80   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
90   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
A0   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
B0   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
C0   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
D0   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
E0   00 FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
F0   FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF
-------------------------------------------------
```

7.6.6 OTHER DATA (cont.)

PAGE AE hex - CU, Lo, Hi data for CONSTANT CURRENT mode

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
40	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
50	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
70	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
80	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
90	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
E0	00	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	F5	00	30	FF
F0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF

8.0 ENTRON LIMITED WARRANTY AND FACTORY SERVICE

ENTRON Controls, LLC., warrants that all ENTRON control panels, **EXCEPT** Mid-frequency Inverter controls, silicon controlled rectifiers (SCRs), insulated gate bipolar transistors (IGBTs), SCR and IGBT assemblies, circuit breakers, and electro-mechanical contactors, are free of manufacturing defects for a period of **TWO YEARS** from the date of original purchase and, in the event of a manufacturing defect, ENTRON will repair or replace, at its discretion, the defective part without any cost for parts or labor.

All silicon controlled rectifiers, SCR and IGBT assemblies, circuit breakers, and electro-mechanical contactors in ENTRON control panels are covered by **a limited warranty from the original manufacturer**. If these parts fail because of a manufacturing defect, they will not be repaired or replaced by ENTRON, but will be returned by ENTRON to the original manufacturer in accordance with said manufacturer's warranty.

ENTRON Controls, LLC., warrants that all Mid-frequency Inverter controls are free of manufacturing defects for a period of **ONE YEAR** from the date of original purchase and, in the event of a manufacturing defect, ENTRON will repair or replace, at its discretion, the defective part without any cost for parts or labor.

To obtain repairs or replacement parts under this warranty, the defective part must be returned, prepaid, to ENTRON Controls, LLC., 1402 S. Batesville Road, Greer, SC 29650. Please send your repair to the attention of "Service" with a description of the problem you are experiencing, contact person and phone number.

EXCLUSIONS: This warranty does not cover damage by accident or misuse, unauthorized repair or modification to any control assembly by the customer.

IMPORTANT NOTE: The warranty period is considered from the date of shipment and is tracked by a serial number code.

USE OF OUT OF WARRANTY REPAIR SERVICE:
To obtain service for any printed circuit board assembly or welding control after the warranty period, send assembly or control, prepaid, to ENTRON Controls, LLC., and ENTRON will repair the printed circuit board assembly or control and return it to you without further warranty on its part. Additional service charges will be invoiced at time of shipment.

Your ENTRON Controls, LLC., Original Equipment Manufacturers (OEM), Dealers and Distributors are your first response contact to secure technical assistance on control or welding problems. Should they be unable to assist you, please contact your ENTRON sales representative or the factory directly. Contact the factory at 864-416-0190.