Mercury 2+

Mercury 2+ Operator Terminal User Manual

This page is intentionally left blank

Special Conditions

The Mercury 2+ has special conditions for safe use (denoted by X after the certificate number) applicable to it. There are 6 conditions applied, which are shown verbatim below, followed by an explanation written by Mercury HMI Ltd.

Sira Certification Services, special conditions for safe use:-

1. Parts of the enclosure are non-conducting and may generate an ignition-capable level of electrostatic charge under certain extreme conditions. The user should ensure that the equipment is not installed or used in a location where it may be subjected to external conditions (such as high-pressure steam) which might cause a build-up of electrostatic charge on non-conducting surfaces. Additionally, cleaning of the equipment should be done only with a damp cloth.
2. As aluminium is used at the accessible surface of this equipment, in the event of rare incidents, ignition sources due to impact and friction sparks could occur. This shall be considered when the Mercury 2 is being installed in locations that specifically require group II, category 1G equipment.
3. The Mercury 2 may be used in conjunction with a bar code wand, which may be unplugged. Only the type bearing the certificate number Sira 02ATEX2297X may be used.
4. The bar code wand does not meet the 500V insulation requirements. This shall be taken into account when the Mercury 2 is being installed.
5 The user shall take account of the fact that the equipment has been assessed assuming a low risk of impact in service.
6. The bar-code wand shall not be used in the presence of combustible dust.

Mercury HMI Ltd explanation of the special conditions for safe use:-

1. The Mercury 2+ has areas of the enclosure (for example, the display window) that could generate static under some circumstances, and if the static were to build up then a spark could result. To avoid this risk, do not clean or polish the device with a dry cloth or position it where jets of high-pressure steam or other gases could blast over its surface.
2. This is a warning that the enclosure is made of LM24 grade Aluminium, and there is the potential for sparking to occur by impact or friction from objects hitting it. Hence position/protect the unit so that it is unlikely to be struck.
3. This is not relevant to the Mercury 2+, since there is not a bar code wand available and the socket is not fitted.
4. Not relevant, see 3 above.
5. Please position the Mercury 2+ terminal to avoid it being physically struck in normal operation.
6. Not relevant, see 3 above.

This page is intentionally left blank

Contact

Mercury HMI Ltd
George House
Derwent Road
Malton
North Yorkshire
YO17 6YB
UK

Tel: +44 (0) 1653 697 200
Fax: +44 (0) 8700 667 325
Email: sales@mercuryHMI.co.uk
Web www.mercuryHMI.co.uk

This page is intentionally left blank

Revision History

Rev	Date	Changes
080804	13 Aug 08	Mercury IMC Version
090302	2 Mar 09	Various minor updates
090713	13 Jul 09	Various minor updates
110317	17 Mar 11	Changed to Mercury HMI
		Added Special conditions of use
2014r0	11 Nov 14	Re-written in Scribus
		Fix various typos
		Updated "C" Multidrop ID
		Added extra Setup options
		Added extra information regarding twisted pair cable usage
		Updated paint type
		Removed cabling parameters - see R507 manual
		Added Firmware Upgrade

This page is intentionally left blank

Contents

Mercury 2+ Terminal With Proximity Card Reader

Introduction

Overview

The Mercury 2+ is a robust operator interface for use within a remote computer or microcomputer system. The terminal is ideal for use in Oil & Gas, Petrochemical, Pharmaceutical, Paint and other industries, where low-cost, high performance Intrinsically Safe operation is mandatory. The Mercury 2+ is simple to install and incorporates a set-up facility for system configuration.

Weather proof to IP66 standard, the Mercury 2+ is approved for Intrinsically Safe operation for group IIC gases within a zone 0 hazardous area, as well as dust hazards. Power and communication are connected via an Intrinsically Safe interface module mounted in a safe area. The interface module provides RS232, RS422 or RS485 communications for point-to-point or multi-drop systems.

The Mercury 2+ comprises a high contrast back lit liquid crystal display, a tactile, alphanumeric keypad, dedicated socket for an optional reader and external digital inputs and outputs. Communications use standard 7 or 8 bit ASCII characters and transmit inputs from the keypad or data from a security card reader.

The Mercury 2+ terminal and IS interface module may also be used in any non-hazardous industrial location, where the advantages of galvanic isolation are required.

An optional non-Intrinsically Safe version of the Mercury 2+, which does not require an IS interface module, may be used in a safe area. This version can be supplied with either RS232 or RS422/485 communications, and is powered by an external 24 Volt supply.

Serial Numbers

It should be noted that each unit will have two serial numbers:-

1. ATEX serial number, as shown on the label affixed to the front of the device.
2. Product serial number, as shown in the default startup message when the unit is powered on.

It is highly likely that these two serial numbers will be different.

Nomenclature & Conventions

In this manual, ASCII single characters which are either control or non-visible codes (Hexadecimal 00 - 1F, 20 and 7F) are indicated by enclosure in < >, for example, <ESC>.

Character strings which are indivisible sequences are shown between quotation marks, for example, "<ESC> [2 J".

In the ASCII 7 and 8 bit code sets used by Mercury 2+, a character is represented by two digits, each in the range hexadecimal 0 to F. For example, <SP>, the space character is defined (20H).

Mercury 2+ System

Mercury 2+ Terminals are approved for operation within a hazardous area when connected to a R007 or R507 Intrinsically Safe interface module. They may also be used within a safe area using an RS232 or RS485/422 interface.

The Intrinsically Safe Interface Module provides galvanic isolation between the I.S. hazard area connection and the 24 V d.c. power & host communication ports. The connections to the Mercury 2+ Terminal allow a cable length of up to 1 kilometre within the hazardous area. The R007 or R507 Intrinsically Safe interface module communication ports provide for RS232 and differential transmit and receive terminals for RS422/485, with tri-state control.

Card Reader

The security card reader is factory fitted in place of the removable front panel and is secured by four M5 stainless Steel cap-head bolts. The binary number contained on the customer card is read when the card is presented to the card reader and the information is made available to the host computer for a system response.

Dust Hazards

The user shall take account of the fact that the equipment has been assessed assuming a low risk of impact in service.

Locating the Mercury Terminal

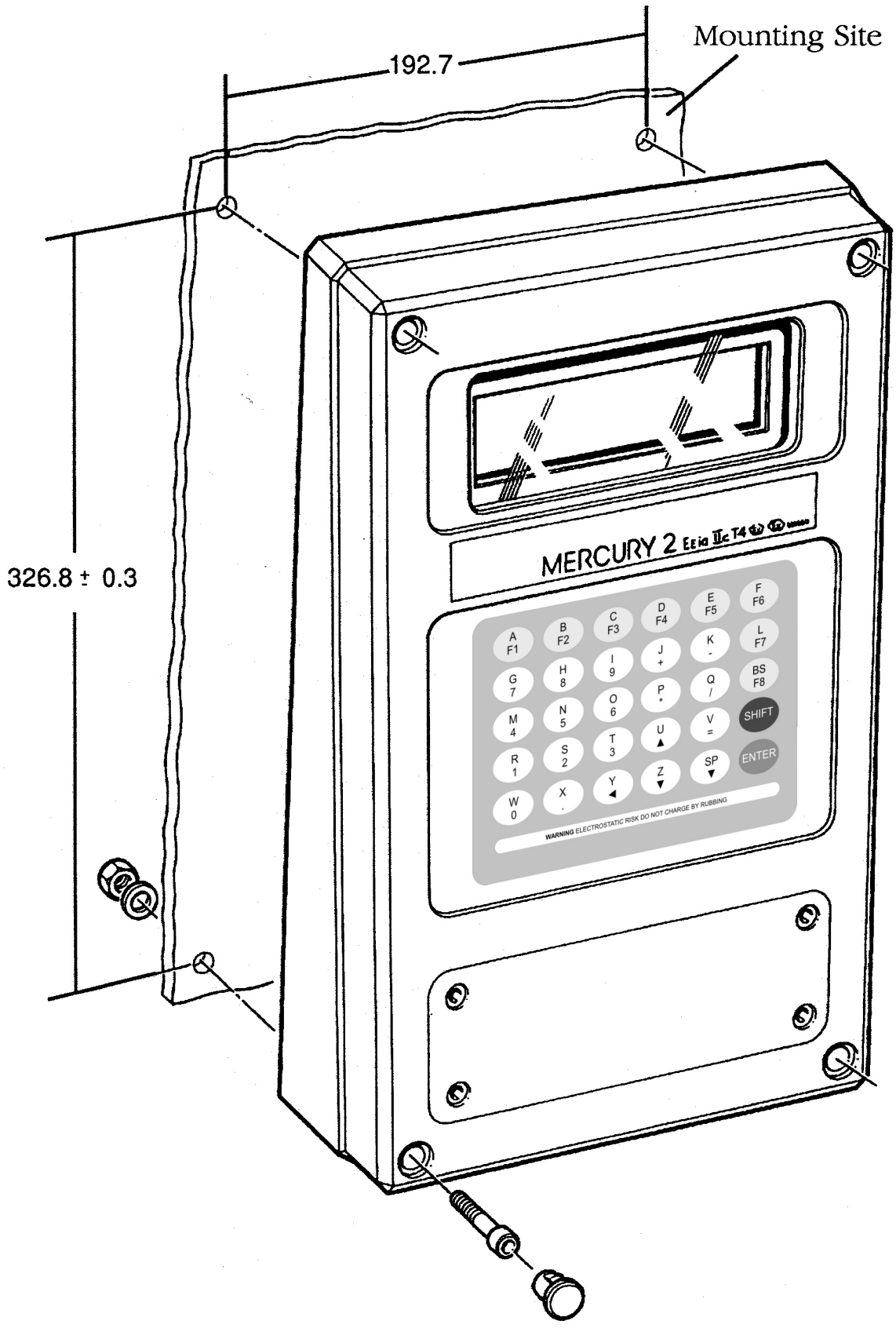
For your safety remember to implement all relevant precautions and procedures. In the United Kingdom installations must comply with BS5345:Part 4.

The Mercury terminal is weather-proof to IP66, so it can be installed outside as well as under shelter or indoors.

Mount the terminal in a vertical position on an even surface, strong enough to support its weight of 5.5 kg.

Position the unit so that the LCD and Keypad are convenient for the operator, usually at eye level. Note that, in strong direct sunlight, display clarity and service life may be reduced.

Mounting the Mercury 2+ Terminal



Mounting Procedure

1. Check that the connection panel cover or security card reader is secured to the face of the Terminal; this prevents any dust or water from entering the unit.

2. Remove the plastic protective plugs covering the four corner mounting holes by pushing the plugs from behind. Retain the plugs in a safe place. It is not necessary to remove the back of the terminal.

3. Place the Terminal against the surface on which it is to be mounted at the correct position and height for operator use and mark the position of the four mounting holes using the dimensions given (see Fig 2). Drill, and plug if necessary, the mounting holes on the mounting surface.

4. Place the Terminal against the mounting surface and secure the Terminal using 4 off M5 cap head screws (26 mm shank) or similar.

5. To avoid electro-chemical corrosion of the aluminium case, the fixing bolts and nuts should be thoroughly greased.

6. Check that the Terminal is securely fastened to the mounting surface and re-insert the plastic protective plugs into the four corner holes.

This page is intentionally left blank

Hazardous Area Connections

Power and communication connections to the Mercury 2+ Terminal are made via the R507 Intrinsically Safe interface module. The IS interface module should be installed in a safe area. The cable screen should be connected to J5 pin 5 within the Mercury 2+.

▲ WARNING **Power must be disconnected before connecting or inspecting the IS interface module.**

Connecting the I.S. Interface Module

No Intrinsically Safe earth is required as the unit is Galvanically isolated.

Mount the IS interface module on to a DIN standard (DIN 46277), 35mm transverse symmetrical rail.

For the complete inter-connection diagram see Appendix F.

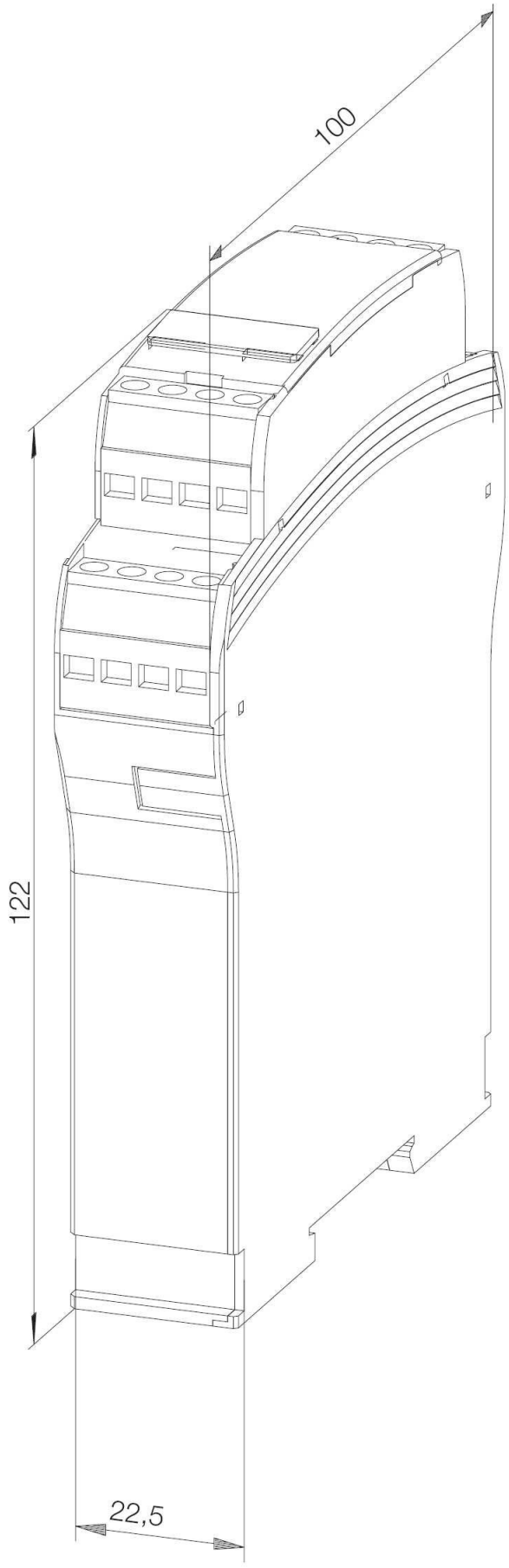
The inter-connection cable between the safe area IS Interface Module and the hazard area Mercury 2+ Terminal requires four cores, which may be either two twisted pairs or a quad.

Where a quad is used, diagonally opposite cores should be paired to reduce any communications cross-talk.

Where twisted pairs are used, one pair should be used for TX1+/TX2+, and the other pair for RX1-/RX2-.

The wiring schedule for the IS interface module is listed in Appendix D.

Fig. 3
I.S. Interface Module



Connectioning the Mercury Terminal

After installing the Intrinsically Safe interface module, you can proceed to connect power and communication wiring to the Mercury Terminal.

For the cable entry into the Mercury Terminal, fit a suitable insulated cable gland (M20) on the four core or twin twisted pair cable.

Connecting to the Terminal Blocks

The terminal connection blocks are protected by a cover, or the card reader if fitted. Remove the cover by unscrewing the four retaining screws.

Power and communication connections are made to terminal block J5, pins 1 to 4. The cable screen is connected to pin 5.

The Mercury 2+ has a "Simple Equipment Interface" to both Digital Inputs and Digital Outputs.

Digital Inputs

The Digital inputs on J2 are whetted and can supply 2.1mA to the Simple Apparatus Switch. This could be a reed-relay or a NAMUR style proximity Switch.

J2 Pins 1, 2, 3, 4, 5 & 6

Uo	=	29.4 V	Li	=	0
Io	=	66.4 mA	Ci	=	0
Po	=	262 mW	Lo/Ro	=	135 µH / Ω
Co	=	71 nF			
Li	=	0			

The connections J2 pins 2, 4 and 6 are at a logic 0V. Only volt free contacts are permitted for use with these digital inputs.

Digital Outputs

The Digital Outputs on J1 are isolated, voltage-free open-collectors.

J1 Pins 1, 2, 3 & 4

Uo	=	29.4 V	Li	=	0
Po	=	262 mW	Ci	=	0

Safe Area Connection

For non Intrinsically Safe connections, specify the Mercury 2+ General Purpose variant of the Mercury 2+. There are two choices, an RS232 version and the RS485 & RS422 version. Please note that operation in Multi-drop mode is not possible if the RS232 option is specified.

In RS232 mode, the Signal Ground needs to be commoned with the incoming power 0V. This allows the same 4 core cable to be utilised as for IS operation.

The General Purpose Mercury 2+ requires a nominal 24V supply (15V to 30V DC) to be fed into External 24V power connections on the Mercury terminal located on terminal block J4, pins 1 and 2. The current is typically 45mA.

The serial data connections utilise J5, as follows:-

J5 Pin No	IS Use	RS232	RS422
1	TX1+	TX	TX+
2	TX2+		TX-
3	RX1-	RX	RX+
4	RX2-		RX-
5	Screen	Screen	Screen

When the Mercury 2+ is used in RS485 mode, the TX+ and RX+ should be connected together and the TX- and RX- should be connected together. Ensure that the Mercury 2+ is set to Multi-drop mode.

Connecting the Wiegand Card Reader

Unscrew the cap-head bolts retaining the cover below the Terminal keypad.

Plug in the 4-pin plug to J3.

Fit and screw the card reader into place, ensuring that the O ring is correctly seated.

Connecting Other Card Readers

Unscrew the cap-head bolts retaining the cover below the Terminal keypad.

Plug in the 4-pin plug to J6.

Fit and screw the card reader into place, ensuring that the O ring is correctly seated.

Cleaning the Mercury Terminal

The body of the terminal is finished in epoxy paint. The display window is polycarbonate and the keypad surface is polyester. These may be cleaned with soapy water. Difficult grease deposits may be treated with most solvents.

In a hazardous area, avoid rubbing dry plastic surfaces with cleaning cloths, as there is a small spark hazard by triboelectric charge generation. This risk can be overcome by using moist cleaning processes. The keypad carries a reminder of this risk.

This page is intentionally left blank

Applying Power

With no local switch, the Mercury terminal receives power when the IS interface module in the safe area is connected. On connection, a beep sounds and the initialisation routine starts with a logo scrolling up the screen. The model number and software version are then detailed on the LCD, the cursor appears and the keyboard mode is indicated in the bottom right corner.

Automatic Message Recall

A feature of the Mercury 2+ terminal is the automatic Message Recall. Immediately after powering up and the version message has been displayed, the Terminal automatically recalls Message No. 1.

For example, this may be a string of text or an escape sequence to set keyboard mode.

Operational Modes

Point-to-Point Mode

The single Terminal "dumb" mode is the default mode of operation, with simple transmission of characters to the host computer on key press, and display of received characterson the LCD.

Multi-drop Mode

Up to fifteen Mercury terminals can be installed on a single multi-drop operation.

Modbus Mode

The Mercury terminal supports the industry standard Modbus protocol. Up to fifteen Mercury terminals can be installed on a single Modbus operation. The Modbus system uses the IS Interface Module as the tri-state controlled communication port.

Firmware Update Mode

The Mercury 2+ product is capable of receiving firmware updates, which may be issued by Mercury HMI Ltd as and when new features are added.

See appendix H for further details.

Entering Set-up Mode

In Set-up mode, the Mercury 2+ is configurable for system operation and communication with the host computer. Set-up is selected from the keypad only and, while in set-up mode, no characters are transmitted by the terminal. Parameters and options are listed on the Set-up menu.

Access Set-up by pressing **SHIFT** four times, then **ENTER**. (On the S600 keyboard, the **SHIFT** key is invisible and is located below the ‘F4’ key and to the left of the '4' key).

If the Security Code has been enabled, the prompt on the LCD asks for a six figure security code. If no code is entered, the options menu can be viewed but not configured.

The first parameter and variable of the Set-up menu is displayed on the bottom line of the LCD.

Setting options

Use the ▲ and ▼ keys to move the cursor to the required parameter, and the keys ◄► to select an option. (On the Terminal Automation keyboard, the ▲▼◄► keys are not indicated. The user must use **H**, **T**, **M**, **O** instead).

Continue selecting parameters and options until the configuration of the terminal is complete. Press **ENTER**, and the selected options are entered into non-volatile memory.

Cold Start

Parameters and options available are listed in Table 1 and described below. Default settings on cold start appear in bold.

Performing a cold start will restore all default settings and erase any stored messages. A Cold Start can be performed in one of three ways:-

1. Power on holding down the two leftmost and two rightmost keys on the bottom row.

2. In Set-up mode, press the bottom left key (i.e. '**O**', **STOP**, **VIEW ALARM**). This will display the message "Cold Start? NO". Pressing the same key will toggle between "Cold Start? NO" and "Cold Start? YES". If "YES" is shown, pressing **ENTER** will cause the machine to pause and then perform a cold start. To return to set-up mode, press **ENTER** when "NO" is displayed.

Should these methods fail for some reason, there is a third method that is guaranteed to work unless the Mercury 2+ is damaged. However, this requires that the Mercury 2+ is partially dismantled, which breaks the rear seal, and incorrect reassembly can lead to water ingress, so please contact the factory before attempting this method.

3. Power on with link LK10 set to right hand position (i.e. ‘1’) then replace in left hand position for normal operation.

Table 1
Set-up Mode Menu

Parameter	Options
Baud Rate	1200, 2400, 4800, **9600**, 19200, 38400, 56700, 115200
Serial Setup	**8N1**, 8N2, 8O1, 8E1, 8M1, 8S1, 7N1, 7N2, 7O1, 7O2, 7E1, 7E2, 7M1, 7M2, 7S1, 7S2
Rx Xon/Xoff	**Enabled**, Disabled
Tx Xon/Xoff	**Enabled**, Disabled
BEL Length	**0.05s**, 0.1s, 0.2s, 0.4s, 0.8s, 1.0s
Beep Loudness	1 to **3** Stars * to ***
Key Click	Off, **70ms**
Security Code	Enabled, **Disabled**
Telemetry Address	**1** to 15
Telemetry Mode	**Point-to-Point**, Multi-drop, Modbus
Multi-drop Acknowledge	Enabled, **Disabled**
Full Stop Key Transmit	**Full Stop**, Comma
Software Version	Indicates Version Number
F8/BS Key Transmit	**<BS>**, {F8}
Rubout Key Transmit	**<BS>**, <DEL>
Keyboard Modes	**N U L**, N U
Block Structure	**Enabled**, Disabled
Contrast	1 to 32, default is **16**
Backlight	Auto only
Cardreader Bits	1 to 64, default is **34**
Language	**English**, French
Clear Text Compose Line	**No**, Yes
DIGIN1 Card Present	**No**, Yes
M2e Compatible	Enabled, **Disabled**
Proximity Byte Count	**00** to 32
Proximity Truncate	**Lower**, Upper
Barcode/Proximity Data	**Normal**, Reversed

Note the **BOLD** text indicates the default settings.

Parameters & Options Explained

Rx Xon/Xoff (Point-to-Point mode only)
When enabled, the Mercury 2+ sends an Xoff and Xon to prevent the receive buffer being over-filled. If disabled, overrunning the terminal's receive buffer may result in characters being lost.

Tx Xon/Xoff (Point-to-Point mode only)
When enabled, the Mercury 2+ stops or restarts sending data from its transmit buffer in response to Xoffs and Xons. If disabled, then the host computer may over-run its input.

Block Structure
When disabled, data normally sent in Blocks (see page 37) is sent as raw data (i.e. with no <STX>, Addr, Func, ID, DMY, CSUM or <ETX> characters as shown on page 38).

Security Code
The default security code is 000000. The six digit security code is programmed into the terminal from the host computer. If the security code option is enabled, the operator has to enter a matching code at the keypad to access Set-up configuration.

Telemetry Modes
In multi-drop or Modbus mode the host computer (the master device) transmits strings and commands to its population of Mercury 2+ Terminals (slave devices) with an address, data and message terminator structure.

Telemetry Address
When Multi-drop or Modbus is enabled, a unique Telemetry Address must be set for each Mercury 2+ Terminal. Fifteen unique addresses, 1 to 15, are available. Address "0" is reserved for broadcast operation when the same message is sent to all slave devices simultaneously.

Multi-drop Acknowledge
When Enabled, an acknowledgement reply is sent in response to every valid received message of matching address except a broadcast.

Cardreader Bits
This specifies the expected card bitcount when using a Wiegand swipe card reader module. This option has no effect when a Proximity RFID reader module is in use.

Clear Text Compose Line
When Enabled, this will automatically clear the text compose line when the ENTER key is pressed. Note that this option is only valid in Multi-drop and Modbus modes.

DIGIN1 Card Present
When Enabled, this allows DIGIN1 to correctly display the card present status of the Proximity card reader module.

M2e Compatible
When Enabled, this allows the unit to match the behaviour of a Mercury 2e terminal.

Proximity Byte Count
This option allows the number of data bytes read from a proximity card to be adjusted to suit the user's requirements. When set to '00' (the default), all data from the card is used. When set to any other value, that number of bytes will be sent back to the host whenever a card is read. The card data will be truncated (or padded with zeros) to accomodate the requuired setting.

Proximity Truncate
Used in conjunction with the "Proximity Byte Count", this option specifies whether data will be truncated (or padded) at the upper or lower end of the string of card data).

Barcode/Proximity Data
When set to 'Reversed', the byte order of card data is swapped.

Backlight

Due to the limited amount of power available when using I.S. equipment, the backlight control is automatic. The adaptive control attempts to keep the backlight as bright as possible, and will purposefully turn the backlight off and then back on again whenever the beeper sounds, or a card is swiped.

This option also acts as a clear visual indicator that a key has been pressed or a card has been swiped, and may be useful in noisy environments when the beep itself may not be audible

Local Echo
(Point-to-point mode only)

Local Echo mode is a facility to help in checking that the keyboard, barcode or card reader is functioning correctly. When set, all key presses, readings or swipes echo the transmitted characters to the screen. The characters are displayed in current screen mode, and at current cursor coordinates, so the screen display should be set to the appropriate mode before Local Echo is set. Local Echo toggles on and off by pressing **F1** (or **START BATCH 1** on the terminal automation keyboard), when in Setup. In Local Echo mode, the terminal continues to communicate with the host. Press **ENTER** to return to normal set-up mode.

Digital Inputs

Pressing **F6** (or **START BATCH 6** on the terminal automation Keyboard), while in the set-up mode gives a single line display showing the state of the digital inputs in real time. Press **ENTER** to return to normal set-up mode.

Display Test

Pressing **F3** (or **START BATCH 3** on the terminal automation Keyboard), while in setup mode, performs a display test. The screen will turn all pixels on, then off, and the terminal will automatically exit setup mode and return to normal operation.

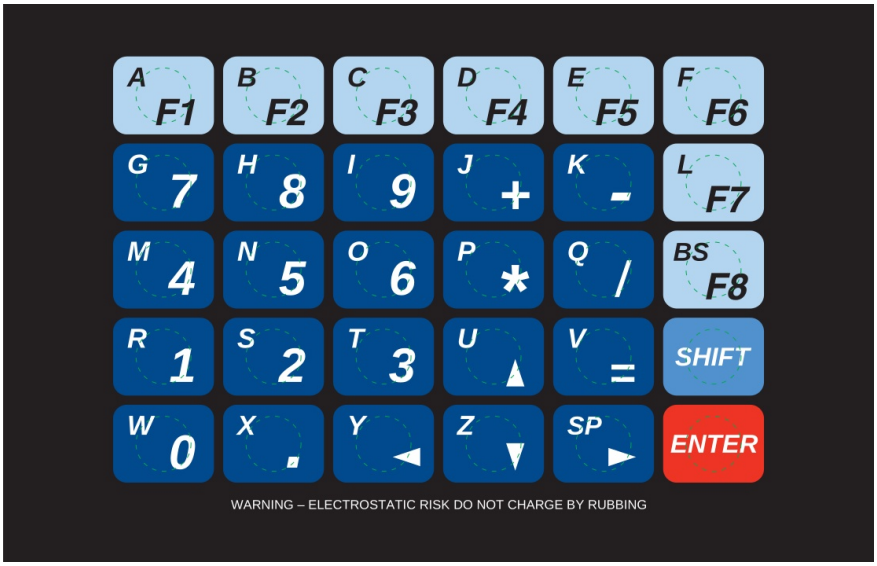
Liquid Crystal Display

In character display or text mode, the LCD offers a display of eight lines high by 40 characters long. Using standard size characters, line wrap is automatic and operates in page format. So after a character has been placed at the end of line eight, in the bottom right corner of the LCD, the cursor goes to the top of the screen again, overwriting line 1. Graphics mode display is detailed on page 41.

Keypad

The keypad is programmed for standard or customised key set. A click is audible each time a key is pressed (this option can be disabled), and confirms operation. Standard keypad layout is illustrated in Figure 4, showing numeric / command symbols in larger type and alphabet keys in smaller type.

Fig. 4
Standard Keypad Layout



Keyboard Modes

There are three keyboard modes - Uppercase characters (U), Lowercase characters (L) and Numeric/Command (N). The current keyboard mode, U, L or N, is shown in the bottom right corner of the LCD.

Note that certain keypad types may not support all keyboard modes. For exampe, the TA keypad supports only Uppercase and Numeric modes.

Press **SHIFT** to display the next keyboard mode. Press **SHIFT** again for the next keyboard mode. A third **SHIFT** key press returns the display to the original keyboard mode.

The capability to change keyboard mode using the **SHIFT** key can be disabled/enabled from the master device (see page 35).

A full list of standard characters for each keyboard mode, together with their hexadecimal values, is shown in Table 2.

Table 2 Keystroke Characters

Key	Alpha Upper-Case	Lower Case	Numeric
A F1	A (41_H)	a (61_H)	F1 (1B,4F,50_H)
B F2	B (42_H)	b (62_H)	F2 (1B,4F,51_H)
C F3	C (43_H)	c (63_H)	F3 (1B,4F,52_H)
D F4	D (44_H)	d (64_H)	F4 (1B,4F,53_H)
E F5	E (45_H)	e (65_H)	F5 (1B,5B,31,37,$7E_H$)
F F6	F (46_H)	f (66_H)	F6 (1B,5B,31,38,$7E_H$)
G 7	G (47_H)	g (67_H)	7 (37_H)
H 8	H (48_H)	h (68_H)	8 (38_H)
I 9	I (49_H)	i (69_H)	9 (39_H)
J +	J ($4A_H$)	j ($6A_H$)	+ ($2B_H$)
K -	K ($4B_H$)	k ($6B_H$)	- ($2D_H$)
L F7	L ($4C_H$)	1 ($6C_H$)	F7 (1B,5B,31,39,$7E_H$)
M 4	M ($4D_H$)	m ($6D_H$)	4 (34_H)
N 5	N ($4E_H$)	n ($6E_H$)	5 (35_H)
O 6	0 ($4F_H$)	o ($6F_H$)	6 (36_H)
P *	P (50_H)	P (70_H)	* ($2A_H$)
Q /	Q (51_H)	q (71_H)	/ ($2F_H$)
BS F8	<BS> (08_H)	<BS> (08_H)	F8 (1B,5B,32,30,$7E_H$) **or** <BS> (08_H) *
R 1	R (52_H)	r (72_H)	1 (31_H)
S 2	S (53_H)	s (73_H)	2 (32_H)
T 3	T (54_H)	t (74_H)	3 (33_H)
U ▲	U (55_H)	u (75_H)	▲ (1B,5B,41_H)
V =	V (56_H)	v (76_H)	= ($3D_H$)
SHIFT	SHIFT	SHIFT	SHIFT
W 0	W (57_H)	w (77_H)	0 (30_H)
X .	X (58_H)	x (78_H)	. ($2E_H$) or , ($2C_H$) *
Y ◄	Y (59_H)	y (79_H)	◄ (1B,5B,44_H)
Z ▼	Z ($5A_H$)	2 ($7A_H$)	▼ (1B,5B,42_H)
SP ►	SP (20_H)	SP (20_H)	► (1B,5B,43_H)
ENTER	ENTER ($0D_H$)	ENTER ($0D_H$)	ENTER ($0D_H$)

Notes

* if selected in Setup menu

F keys in Numeric mode, and ENTER in all modes, can be programed to send user-defined strings.

Receiving Data

The Mercury 2+ can operate in text or graphics display mode or a combination of text and graphics 'screens'. Receiving data in text or Character Display Mode is described below. Graphics mode is described on page 41.

Character Display Mode

The Mercury 2+ receives and transmits commands in 7 or 8 bit ASCII characters (see page 25). The enhanced character set is only available when both communication and operation of Mercury 2+ are set to 8 bit mode. Note that in 7 bit transmission, only codes 00_H to $7F_H$ are used.

In 8 bit mode, Mercury can receive 256 unique data codes. The first 32 codes control the operation of the terminal and so are not displayed. The next 128 codes are displayable on the LCD, except for <DEL> ($7F_H$). The codes used in 8 bit transmission, 00_H to $9F_H$, with their respective Hexadecimal values and standard ASCII names, are given in Table 3.

Table 3 Standard ASCII Names of Control Codes

Hex	$0(x)_H$	$1(x)_H$	$2(x)_H$	$3(x)_H$	$4(x)_H$	$5(x)_H$	$6(x)_H$	$7(x)_H$	$8(x)_H$	$9(x)_H$
0	<NUL>	<DLE>	<SP>	0	@	P	`	p	Ç	É
1	<SOH>	<DC1>	!	1	A	Q	a	q	ü	æ
2	<STX>	<DC2>	“	2	B	R	b	r	é	Æ
3	<ETX>	<DC3>	#	3	C	S	c	s	â	ô
4	<EOT>	<DC4>	$	4	D	T	d	t	ä	ö
5	<ENQ>	<NAK>	%	5	E	U	e	u	à	ò
6	<ACK>	<SYN>	&	6	F	V	f	v	å	û
7	<BEL>	<ETB>	‘	7	G	W	g	w	ę	ù
8	<BS>	<CAN>	(	8	H	X	h	x	ê	ÿ
9	<HT>	<EM>	)	9	I	Y	i	y	ë	ö
A	<LF>	<SUB>	*	:	J	Z	j	z	è	ü
B	<VT>	<ESC>	+	,	K	[	k	{	ï	¢
C	<FF>	<FS>	,	<	L	\	l	|	î	£
D	<CR>	<GS>	-	=	M	]	m	}	ì	¥
E	<SO>	<RS>	.	>	N	^	n	~	Ä	Pt
F	<SI>	<US>	/	?	O	_	o	<DEL>	Å	₣

In 8 bit mode, Hexadecimal $A0_H$ to FF_H are available for users to download as customised characters.

The effect of each standard ASCII control character is given in Table 4.

Table 4 Control Character Actions

Character String	Action
<NUL> <SOH> <EOT> <ENQ> <ACK> <HT> <SO> <SI> <DLE> <DC2> <DC4> <NAK> <SYN> <ETB> <EM> <SUB> <FS> <GS> <RS> <US>	These characters terminate escape sequences and are not displayed
<STX> <ETX>	Codes to start and stop blocks.
<BEL>	Sounds Beeper.
<BS>	Back Space.
<LF> <VT> <FF>	Line feed.
<CR>	Carriage Return.
<DC1>	Xon. Causes Terminal to continue transmitting.
<DC3>	Xoff. Causes Terminal to stop transmitting.
<CAN>	Cancels Escape Sequence.
<ESC>	Introduces Escape Sequence.
<EM>	Special Terminator for messages
<IND>	Processed as "down arrow" key.
<DEL> ($7F_H$)	Terminates any escape sequence which is in progress.

Escape Sequence

Each character can act as a standalone control code as above, or can be combined in strings to create escape sequences that offer further control and data transmission opportunities.

To combine characters into strings, the <ESC> control code must begin all sequences. The <ESC> ($1B_H$) code indicates to the unit receiving data that the following string should be considered together.

Sequences transmitted by Mercury 2+ cannot be interrupted and other blocks or keyboard inputs will be buffered until the first sequence is transmitted in its entirety.

The Mercury 2+ is configured to respond to defined escape sequences transmitted by the host computer, and these are listed below with their actions.

Table 5 lists escape sequences that control the cursor and screen.

Table 6 gives escape sequences that create commands.

Table 5 Cursor/Screen Control Strings in Text Mode

Action	Character String
Clear Display	"<ESC> [2 J" (on enabled screens)
Clear To End Of Line	"<ESC> [K"
Set Graphic Display Mode	"<ESC> [? 2 z"
Set Character Display Mode	"<ESC> [? 3 z"
Make Cursor Visible	"<ESC> [? 25 h" (Text mode only)
Make Cursor Invisible	"<ESC> [? 25 1" (Text mode only)
Move Cursor *	"<ESC> [**Pr** ; **Pc** H"
	"<ESC> [**Pr** ; **Pc** f" (F/W 4v47+)
New Line	"<ESC> E"
Cursor Down	"<ESC> D"
Cursor Up	"<ESC> M"
Save Cursor Position	"<ESC> 7" (Text mode only)
Restore Cursor Position	"<ESC> 8" (Text mode only)
Home Cursor (1,1) Top Left	"<ESC> [? 6 1"

* **Pr** and **Pc** define the location to which the cursor moves, being respectively row and column numerals in ASCII code. With Row 1 as the top of the screen and row 8 at the bottom, the number is interpreted as modulo 8 thus if **Pr** equals 16 then the cursor moves to row 8. Similarly, with Column 1 as the left side of the screen and Column 40 on the right, the number is interpreted at modulo 40. Thus if **Pc** equals 84 then the cursor moves to column 4.

Table 6
Command Escape

Action	Character String
Set Security Code (where **Pa**, **Pb**, **Pc** are two digit ASCII coded numerals to create 6 digit security code - see page 26)	"<ESC> [? 1 ; **Pa** ; **Pb** ; **Pc** z"
Select 7 bit ASCII mode	"<ESC> [= 1 l"
Select 8 bit ASCII mode	"<ESC> [= 1 h"
Sample Digital Inputs	"<ESC> [? 4 z"
Write Digital Outputs	"<ESC> [? 25 ; **n** z"
Set Up Message For Later Recall	"<ESC> [? 7 ; **Pn** z **data** <EM>"
Recall Message (where **Pn** is a two digit ASCII numeral defining message identifier, value 1 to 99)	"<ESC> [? 8 ; **Pn** z"
Send Next Queued Block	"<ESC> [? 9 ; 1 z"
Re-Send Last Sent Block	"<ESC> [? 9 ; 2 z"
Delete all stored Messages	"<ESC> [? 10 z"
Store To NVRAM	"<ESC> [? 11 z"
Set Keypad To Numeric Mode	"<ESC> (<"
Set Keypad To Upper-Case Mode	"<ESC>) <"
Set Keypad to Lower-Case Mode	"<ESC> * <"
Enable bar code in one shot mode	"<ESC> [? 15 ; 1 z"
Disable bar code	"<ESC> [? 15 ; 2 z"
Enable bar code	"<ESC> [? 15 ; 3 z"
Enable card reader in one shot mode	"<ESC> [? 15 ; 4 z"
Disable card reader	"<ESC> [? 15 ; 5 z"
Enable card reader	"<ESC> [? 15 ; 6 z"
Redefine <STX> character	"<ESC> [? 16 ; 1 z **p** <EM>"
Redefine <ETX> character (where p is a single character to be defined)	"<ESC> [? 16 ; 2 z **p** <EM>"
Restrict keyboard to U and N modes	"<ESC> [? 17 ; 1 z"
Allow keyboard U, N and L modes	"<ESC> [? 17 ; 2 z"
Revert function of <FF> character to line feed	"<ESC> [? 21 ; 0 z"
Change function of <FF> character to Clear Screen	"<ESC> [? 21 ; 1 z"
Select text page number (where **num** is 0 or 1)	"<ESC> [1 ; **num** }"

Sequences Specifying Screen Mode

Up to three separate screen pages can be controlled at a time from the host computer, to display two text pages and a graphics image. The screen pages can be individually controlled to display text only, graphics only or a combination of text with graphics. The following specify screen mode:-

Text and graphics off	"<ESC> [? 20 ; 0 z"
Text on, graphics off	"<ESC> [? 20 ; 1 z"
Graphics on, text off	"<ESC> [? 20 ; 2 z"
Text and graphics on, with the screens logically ORed	"<ESC> [? 20 ; 3 z"
Text and graphics on, with the screens logically XORed	"<ESC> [? 20 ; 4 z"
Text and graphics on, with the screens logically ANDed	"<ESC> [? 20 ; 5 z"

Disabling Keyboard or Keyboard Mode Selection

Keyboard entry is disabled / re-enabled from the host computer by transmitting the escape sequence:-

Disable Keyboard	"<ESC> [? 2 h"
Enable Keyboard	"<ESC> [? 2 l"

Using **SHIFT** key to change the keyboard mode can be disabled / re-enabled from the host computer by transmitting the escape sequence:-

Disable Shift Key	"<ESC> [? 13 z"
Enable Shift Key	"<ESC> [? 12 z"

With the standard keyboard, the set-up menu, however, can be entered as before by pressing **SHIFT** four times, then **ENTER**. For other keyboards see page 24.

NOTE: When the keyboard or **SHIFT** is disabled, the ULN mode indicator is not displayed on the screen.

Pin Number

When enabled, any text entered in the Composed Text Input will be shown as a dot character '.', but will be transmitted to the host as the actual text that was entered. Enabling 'PIN Mode' has no effect in point-to-point mode.

'PIN Mode' disabled (i.e. text visible)	"<ESC> [? 24 ; 0 z"
'PIN Mode' enabled (i.e. text shown as dot characters '.')	"<ESC> [? 24 ; 1 z"

e.g.

'PIN Mode' status	**Text entered via keyboard**	**Text shown on display**	**Text sent to host**
Disabled	1234567890	1234567890	1234567890
Enabled	1234567890	…….	1234567890

User-Definable Characters

There are 96 ASCII two-digit codes available for users to assign to escape sequences that are frequently required. These are hexadecimal $A0_H$ to FF_H (see Table 3) and available in 8 bit mode only. The sequence to define a character is as follows:-

"<ESC> [? 19 ; **num** z **data**"

num is ASCII number of the character being defined (between $A0_H$ and FF_H). **data** gives 8 bytes of data which must be supplied as ASCII characters. For Pixel to ASCII character conversion table, see page 41.

Note that some fonts may use these characters for their own purposes. Changing fonts may overwrite some or all of the user-defined character data.

User-Definable Keys

Redefine a Function key or **ENTER** with up to 8 ASCII characters, placed in the following sequence at **data**.

"<ESC> [? 23; **Pn** z **data** <EM>"

Pn is the number of the key being redefined. Numbers 1 to 8 redefine **F1** to **F8** respectively, and 9 redefines **ENTER**.

Digital Outputs

The digital outputs can be individually controlled. Each output can be turned on or off, or set to toggle its current setting for a period of time (so as to produce a single pulse). The following sequence is used:-

"<ESC> [? 25; **num** z"

Values for **num** are as follows:

num	**Action**
0	Turn Digout 1 off
1	Turn Digout 1 on
2	Turn Digout 2 off
3	Turn Digout 2 on

The outputs can also be set to toggle their current setting for a period of time, so as to produce a single pulse, using the following sequence:-

"<ESC> [? 25; **num** ; **count** z"

num	**Action**
4	Pulse Digout 1
5	Pulse Digout 2

count denotes the pulse length in $1/10^{th}$ second steps. A value of 0 is ignored and the maximum value is 255.

Font selection

Use the following sequence to select the required font when displaying text on the screen:-

"<ESC> [? 26 ; **num** z"

Values for **num** are as follows:-

num	**Font**
0	English
1	Cyrillic
2	Thai

Messages

Messages, in the form of characters or escape sequences that are sent to the Mercury, can be recalled for display many times (see Table 6).

A total of 7,000 characters may be stored as messages in RAM. Each message is assigned an identifying value between 1 and 99. A message should not be allocated to an identifier already in use.

The codes in the message can be any visible or control character, except <STX>, <ETX> and <EM>, or the escape sequence for "Set Up Message for Later Recall". It is possible to create a message string to recall other messages.

Storing Messages

On transmission to the Mercury, messages are stored temporarily in Static Random Access Memory (SRAM). A message may be recalled as often as required, but it is lost if the power is removed from the Mercury 2+ Terminal. If the command "Store to NVRAM" is sent after the message, then the contents of the SRAM are copied to the Non-Volatile RAM (NVRAM) to ensure preservation in case of power failure.

If the Mercury 2+ Terminal is turned off and subsequently turned on, it copies the contents of NVRAM into SRAM, restoring any messages down-loaded up to the time the last "Store to NVRAM" command was sent. If the set-up mode is entered and left, the contents of SRAM are copied to NVRAM automatically.

Blocks

Blocks are strings of data in a fixed format which cannot be interrupted, obtained from a read of digital inputs, or from a bar code read or a card swipe. The format distinguishes the data string from keypad-entered data.

Point to Point Transmission

Transmission of messages in point-to-point configuration is immediate, i.e. at every key stroke, or promptly after a card-read. The control codes and escape sequences described in this section are available, with the exceptions noted.

Transmitted messages follow the format:

<STX> ADDR FUNC ID DATA DMY CSUM <ETX>

<STX>	Start Transmission (02_H)
ADDR	A two byte address field, as set up in Telemetry Address with value 1 to 15. For multi-drop mode see page 40. This is always "01" in point to point mode.
FUNC	Determines type of information being transmitted with a single byte character. For messages transmitted from the Mercury 2+ terminal this is always "D" (44_H), and "R" (52_H) for messages received by the Mercury 2+ terminal.
ID	Identifies source of data from a single byte:- "B" (42_H) data from Wiegand Security Card swipe "C" (43_H) data from Proximity Card reading "E" (45_H) data from Digital Input read. Other ID codes are available in Multi-drop mode (see page 40)
DATA	Information being sent from the identified source: data from Security Card, encoded as hex ASCII, or data for Digital Input reader as a single byte.
DMY	Single character, usually 00_H but if this would result in the following CSUM byte being a control character, the DMY is set to 20_H.
CSUM	A single byte checksum character, which is the 7 bit negated algebraic sum of all the characters in the string from <STX> to DMY inclusive.
<ETX>	End Transmission character (03_H) (see Table 4)

Examples of messages are given in Appendix B.

Multi-Drop Telemetry

In multi-drop mode the master device polls the slave Mercury 2+ Terminals. Up to 15 Terminal systems can transmit to a single host computer via a single twisted pair cable utilising RS485 levels (two pairs if RS422 is used). The Multi-drop master initiates all communications and the slaves can only reply when requested.

Multi-drop operation is supported over the RS485 and tri-state controlled RS422 communications link between I.S Interface Modules. So these must be installed even if operation is in non-hazardous areas.

Multi-Drop Mode Screen

The Mercury 2+ LCD screen in multi-drop mode operates in the standard page format, with the exception that the eighth line is used as the editing area for composing blocks. To avoid deletion of data being composed on Line 8, host operators should use lines 1 to 7.

In the case of the host computer transmitting a Clear Screen escape sequence, in Multi-drop mode, lines 1 to 7 of the Screen are cleared. Line 8 is not affected.

On Line 8, an alpha-numeric key press causes the character to appear on the bottom line of the screen at column 6. Up to 30 characters can be composed into a message. To edit use the backspace key, **BS** (**F8** in Numeric mode). When complete, press **ENTER** to queue the message for transmission.

If a function key is activated while a message is being composed, the function code is added to the block buffer ahead of the message, without affecting the composition of the message.

Messages are block-based and therefore indivisible. So the software handshaking facility Xon/Xoff is automatically disabled if Multi-drop Operation is selected in the Setup menu (see Table 1).

The Mercury terminal does not inhibit received characters being displayed on the 8th line of the screen. So a message being composed may be overwritten on the screen, although it will still be composed correctly into a block.

Block Format

The block format for messages in multi-drop are similar to that for Point-to-Point mode, and are received as well as transmitted:-

<STX> ADDR FUNC ID DATA DMY CSUM <ETX>

These fields are described on page 38. The following fields have additional options in multi-drop mode as follows:-

ADDR — Mercury 2+ Terminal can be set to respond to any address in range 1 to 15 in the Setup menu. Address 00 is reserved for broadcast to all slave terminals.

ID — In addition to the codes B, C and E described in Point-to-Point Mode, a single byte defines the block source as follows:-

"A" (41_H) block from operator keyboard entry
"D" (44_H) block from keyboard function key (Fl to F8)

DATA — Format differs slightly for sending or receiving data blocks. Block data received by the Mercury 2+ can be up to 128 bytes long, containing visible characters and escape sequences.

Block data transmitted by the Mercury 2+ includes one of the above ID bytes or the "Multi-drop Acknowledge" block (see Setup menu on page 25). No acknowledgement is sent if the received block was a broadcast.

Sending Block

A key or series of keys pressed on the terminal are stored into an output buffer on the key press **ENTER**. The master unit reads the output buffer on a FIRST-IN, FIRST-OUT basis. The escape sequence to send the next buffer is defined in Table 6 on page 34.

Examples of typical sequences and how they are composed are given in Appendix B.

Graphics Display Mode

To select Graphics with Text or Graphics alone, see page 35. Note that on Mercury terminals with text only, the enhanced software described below is not available.

Graphics Display

In Graphics mode, the display is made up of 64 rows of 40 columns. Each column position is a 'tile' made up of 6 pixels across by 1 pixel in height.

No cursor is visible in graphics mode. As a tile is written, a virtual cursor is incremented. The virtual cursor has its own set of coordinates allowing access to graphics without corrupting text currently displayed.

Coordinates are defined by the byte number and row number, calculated from the top left corner of the display. With one byte equating to 6 pixels, simple or bitmap images are positioned horizontally from the 6 pixel boundaries.

Graphics may be downloaded as a bit-map image from off-line Graphics Converter Software, available from Mercury HMI Ltd, or designed directly on the display.

Each tile is individually accessible, and each pixel pattern is uniquely mapped to an ASCII character, defined in Table 7 below. In the table a ○ represents a clear, OFF state pixel, and a ● represents a pixel in the energised, ON state. The pixel pattern is read left to right.

Table 7 Graphics Mode Pixel Data

○○○○○○	SP	○●○○○○	0	●○○○○○	@	●●○○○○	P
○○○○○●	!	○●○○○●	1	●○○○○●	A	●●○○○●	Q
○○○○●○	“	○●○○●○	2	●○○○●○	B	●●○○●○	R
○○○○●●	#	○●○○●●	3	●○○○●●	C	●●○○●●	S
○○○●○○	$	○●○●○○	4	●○○●○○	D	●●○●○○	T
○○○●○●	%	○●○●○●	5	●○○●○●	E	●●○●○●	U
○○○●●○	&	○●○●●○	6	●○○●●○	F	●●○●●○	V
○○○●●●	‘	○●○●●●	7	●○○●●●	G	●●○●●●	W
○○●○○○	(	○●●○○○	8	●○●○○○	H	●●●○○○	X
○○●○○●	)	○●●○○●	9	●○●○○●	I	●●●○○●	Y
○○●○●○	*	○●●○●○	:	●○●○●○	J	●●●○●○	Z
○○●○●●	+	○●●○●●	;	●○●○●●	K	●●●○●●	[
○○●●○○	,	○●●●○○	<	●○●●○○	L	●●●●○○	\
○○●●○●	-	○●●●○●	=	●○●●○●	M	●●●●○●	]
○○●●●○	.	○●●●●○	>	●○●●●○	N	●●●●●○	^
○○●●●●	/	○●●●●●	?	●○●●●●	O	●●●●●●	_

Cursor and Screen Control Codes

In Graphics Mode, Cursor/Screen Control codes differ slightly from those in Character Display Mode. Table 8 lists the escape sequences and actions that result in when Graphics Mode.

Table 8 Cursor/Screen Control in Graphics Mode

Action	Character String
Clear Display	"<ESC> [2 J"
Set Graphic Display Mode	"<ESC> [? 2 z"
Set Character Display Mode	"<ESC> [? 3 z"
Move Virtual Cursor *	"<ESC> [**Pr** ; **Pc** H"
	"<ESC> [**Pr** ; **Pc** f" (F/W 4v47+)
Virtual New Line	"<ESC> E"
Virtual Cursor Down	"<ESC> D"
Virtual Cursor Up	"<ESC> M"
Select standard character size	"<ESC> [? 3 z"
Select larger character size	"<ESC> [? 3 Z"

The LARGER character mode uses a triple height, extra wide character needed to satisfy certain metrological regulations.

Note : Selecting either standard or larger character size automatically places the Mercury into character display mode.

* Pr and Pc define the tile to which the cursor should move, being respectively row and column numerals in ASCII code. Tile row 1 is the top of the screen and tile row 64 is the bottom, so **Pr** refers to the row location of the tile to where the virtual graphics cursor will move. Tile column 1 is at the left side of the screen and tile column 40 at the right, so **Pc** refers to the column location of the tile.

Control Codes

The remaining Control character strings listed in Table 6 on page 34 for Character Display Mode apply equally in Graphics Mode.

Draw Line

A line can be drawn at any angle using the following sequence:-

<ESC> [? 18 ; 4 ; **x1** ; **y1** ; **x2** ; **y2** z

where **x1** (pixel number) and **y1** (row number) give the pixel coordinates of the start of the line, and **x2**, **y2** give the pixel coordinates of the end of the line.

Note that all coordinates start at zero, i.e. the top left pixel is located at (0, 0).

Draw a Solid Box

To draw a solid box, use one of the following sequences.

Solid black box	<ESC> [? 18 ; 2 ; **x1** ; **y1** ; **x2** ; **y2** z
Solid white box	<ESC> [? 18 ; 3 ; **x1** ; **y1** ; **x2** ; **y2** z

where **x1** (pixel number) and **y1** (row number) give the pixel coordinates of the start of the line, and **x2**, **y2** give the pixel coordinates of the end of the box.

Downloading a Graphic Bitmap Image to Screen

When downloading a graphics bitmap image, enter the position and size of the image in the following sequence.

<ESC> [? 18 ; 1 ; **x1** ; **y1**; **width** ; **height** z **data**

where **x1** (byte number) and **y1** (row number) are the coordinates of the top left corner, and **width**, **height** give the width (in bytes) and height (in rows) of the image. **data** is the pixel data as shown in table 7.

Optional Equipment

Card Reader

Operated by presenting a security card through the reader, a successful read is indicated by a beep. In point-to-point mode, transmission to the host computer is immediate. In multi-drop mode the data is stored in the output buffer. The message format is determined by the byte structure setting, selected during set-up, and comprises an escape prefix, function code, data and return. For sequences controlling the card reader, see page 34.

Technical Specification

Mercury 2+ Terminal

Physical Data

Height	370 mm	
Width	227 mm	
Depth	(with card reader)	97 mm
	(without card reader)	67 mm
Weight	5.5Kg	
Colour	Ultramarine Blue	
Paint type	Epoxy primer DuPont Alesta® AP	

Performance Data

Display Type	Supertwist, Liquid Crystal. Reflective filters, LED backlight
Display Colour	Black on silver
Character Mode	40 characters, 8 lines 6 x 8 dot character cell; 128 pre-defined and 96 downloadable ASCII characters in 8 bit mode; 96 ASCII character set in 7 bit mode. Character height 4.2 mm.
Graphics Mode	240 x 64 graphic pixels, forming 2560 tiles. Character cell height 4.2 mm or 12.7 mm.
Keypad	Sealed membrane switches. Polyester outer layer, with tactile response. Damp wipeable for clean/sterile environments. 30 keys including 8 functions. Option for Audible sounder via membrane selected in Setup.
Keyboard modes	Upper Case (U), Lower case (L), Numeric/Command (N).
Digital Inputs	3 whetted inputs for external contacts or I.S. "simple apparatus".
Digital Outputs	2 Optically Isolated Open-collector outputs. For Safe Area use, can be used with 24V DC up to 100mA maximum.
Protocol	Based on VT 100.
Communication	To and from the host computer in full or half duplex, using standard 7 bit or 8 bit ASCII characters. 1200 to 115,200 Baud.

Buffers	Incoming buffer 384 characters long. Xon-Xoff control selected in Setup. Xoff sent to host computer when buffer is within 10 bytes of being full. Xon sent when Terminal buffer is within 5 characters of being empty. In multi-drop or Modbus mode, Xon-Xoff control is disabled.
User Memory	8K NVRAM for rapid recall of user graphics or characters.
Digital Inputs	Data is the binary value of the 3 readable inputs biased by 20_H

D7	Always a zero.
D6	Always a zero.
D5	Always a one.
D4	Same as Digital Input 1
D3	Always a zero.
D2	Digital input 3- This bit is a "1" if a contact is closed.
D1	Digital input 2. This bit is a "1" if a contact is closed.
D0	Digital input 1. This bit is a "1" if a contact is closed.

	If the Digital Input is wired to a proximity switch; “1” = No Metal Present “0” = Metal Present
Inputs identified by ID	"A" (41_H) Operator key press "B" (42_H) Security Swipe Card "C" (43_H) Security Proximity Card "D" (44_H) Function key "E" (45_H) Digital Input "F" (46_H) Field Serial Input
Multi-Drop Mode	Maximum of 15 terminals may be multi-dropped via IS interface modules.
Multi-Drop Protocol	Proprietary, based on ANSI-X3.

Environmental Conditions

Operating Temperature	-20°C to 60°C
Storage	-20°C to 70°C
Protection	IP66 Standard
Certification	EEx ia IIC T4
Sira Safety Services Ltd.	SIRA 02ATEX2297X
Quality Assurance	ISO9000:2000

Non-I.S. Mercury 2+ Terminal

The non-IS terminal has similar specifications to IS Terminal. The unit may be used as a stand alone safe area Terminal powered by a 24 V, 1 Watt supply, with an RS232 or RS485/422 port.

HID Proximity Card Reader

Performance Data

Principle of Operation	125KHz RFID Proximity card reader
Card Code	26 bit binary plus 2 parity bits is standard, but the Mercury 2+ supports all other HID 125KHz card standards
Intrinsic Safety	Card reader is certified for use with Mercury 2+ Terminal
Reading Distance	Minimum of 10mm at reader Centre

Environmental Conditions

Operating Temperature	-20°C to +60°C
Storage	-20°C to +70°C
Protection	IP66 Standard

EM4001 Proximity Card Reader

Performance Data

Principle of Operation	125KHz RFID Proximity card reader
Card Code	26 bit binary plus 2 parity bits is standard
Intrinsic Safety	Card reader is certified for use with Mercury 2+ Terminal
Reading Distance	Minimum of 10mm at reader Centre

Environmental Conditions

Operating Temperature	-20°C to +60°C
Storage	-20°C to +70°C
Protection	IP66 Standard

MIFARE Proximity Card Reader

Performance Data

Principle of Operation	13.56 Mhz Contactless Smartcard (originality from Philips)
Card Code	MIFARE Classic cards supported. Only the 32 bit Unique Serial Number (USN) is read
Card Writing	This is not permitted with the IS Card Reader
Intrinsic Safety	Card reader is certified for use with Mercury 2+ Terminal
Reading Distance	Minimum of 10mm at reader Centre

Environmental Conditions

Operating Temperature	-20°C to +60°C
Storage	-20°C to +70°C
Protection	IP66 Standard

R507 I.S. Interface Module

Physical Data

Height	1220mm
Width	118 mm
Depth	33 mm
Weight	250 grams (approx.)
Material	Thermoplastic UL 94 V-0
Colour	Black
IS Terminals	20V, 100Ω nominal, galvanically isolated
IS Cabling	See page 17
Mounting	35mm transverse rail DIN 46277
Power	20-32 V d.c. 2W maximum. Nominal 80mA at 24V
IS Earth	Not Required
Isolation	Incoming power and Comms are galvanically isolated from hazardous area
Communications	RS232, RS422 and RS485 (see Appendix E)
Intrinsic Safety	Mounted in safe area Epsilon 06ATEX2107

Environmental Conditions

Operating Temperature	-20°C to +60°C
Storage	-40°C to +70°C
Humidity	95% non-condensing
Protection	IP40
Location	Safe Area

Appendix A

References

1. Hand Book of Reliability Data (4), British Telecom.

2. BS EN 60079-25:2004 Equipment for potentially explosive atmospheres.

This page is intentionally left blank

Appendix B

Using Mercury 2+ in Multi-drop Mode

Examples of how to use the Mercury 2+ escape codes to complete actions are given below. Take a few minutes to familiarise yourself with the method for designing blocks and messages, or graphic images.

Text Transmission

This example runs through the method to send a text message to a Mercury's display. The text used in this example is the word "TEST".

Using the format of a block, we can examine how it is composed:-

"<STX> ADDR FUNC DATA DMY CSUM <ETX>"

<STX> character always starts a block and is sometimes known as the Control B character. The terminal needs to receive the character Hex value 02_H.

ADDR
Before any transmission is made, determine the destination terminal address. This address is set manually in the Set-up mode on the terminal and is a number between 1 and 15. The ADDR field is a two byte field. So if the destination terminal address is 1, then the field must contain 01.

FUNC
The FUNC byte is a "D" (44_H) for messages transmitted from the Mercury 2+ terminal and "R" (52_H) for messages received by the Mercury 2+ terminal. In this example this byte must be "R".

DATA
A field of variable length, this contains the "message" to be transmitted whether it is text or a control character string. In this example the word "TEST" is being transmitted.

<ETX>
The <ETX> character always ends a block and is sometimes known as the Control-C character. The terminal needs to receive the <ETX> character that has the value 03_H.

Checksum Calculation

To calculate the CSUM for the above example:-

Take the Hex value of the Characters and add them all together. The codes are listed on page 31.

<STX>		(02_H)
ADDR	01	(30, 31_H)
FUNC	R	(52_H)
DATA	TEST	(54,45,53,54_H)
DMY	<NUL>	(00_H)

Sum of above codes:-

$02+30+31+52+54+45+53+54+00 = 01F5_H$

01F5 in Hex = 111110101 in Binary
'AND' the above with ($7F_H$) = 1110101
Negate the above (two's complement) by inverting all bits and add 1 = 0001011 = $0B_H$

This value is less than 20_H, which is a control code and so the process needs to be repeated with DMY = 20_H. This results in a new checksum having the value $6B_H$, which translates to the character "k".

To transmit the word "TEST" to the terminal at address 01, the following string needs to be sent:-

"<STX> 0 1 R T E S T <SP> k <ETX>"

Reading the Block Buffer

Using the format described above, the Master sends a block to read the output buffer on a Mercury 2+ terminal with Multi-drop address 01 as follows:-

<STX>		(02_H)
ADDR	01	($30,31_H$)
FUNC	R	(52_H)
DATA	"<ESC> [? 9 ; 1 z"	($1B,5B,3F,39,3B,31,7A_H$)
DMY	<NUL>	(00_H)

The checksum can be calculated to be 77_H, which is the ASCII code for the letter "w".

The block to be sent to the terminal takes the form:-

"<STX> 0 1 R <ESC> [? 9 ; 1 z <NUL> w <ETX>"

The Re-send Last Block command (see page 34) allows the buffer to be read repeatedly.

Reading the Digital Inputs

The Control Character String to read the Digital Inputs in Multi-drop mode is:-

"<ESC> [? 4 z"

Using the method demonstrated above, the block to be sent in multi-drop mode becomes:-

"<STX> 0 1 R <ESC> [? 4 z <NUL> h <ETX>"

This action causes the status of the external contacts to be copied into the output buffer. It is possible to incorporate a Block Read instruction into a single block command by putting into the DATA field both the Control String to read the digital inputs and the Control String to read the block buffer.

Multi-Drop Acknowledge Mode

If the destination terminal is set up with Multi-drop Acknowledge Enabled then the Mercury 2+ terminal sends an Acknowledge Block back to the master. It does this on receipt of a valid Block of matching address (i.e. the Multi-drop Protocol is correct) and takes the form:-

"<STX> 0 1 D <NUL> Y <ETX>"

This string acknowledges that a message with the correct protocol, ie FUNC, CSUM, etc. has been received, not that the message data is recognised. So if a message is constructed with a non-existent Escape sequence in DATA yet with the correct protocol, then a Multi-drop Acknowledge is transmitted but the Mercury 2+ ignores the instruction.

When the Digital Inputs are read to the block, no information is relayed back to the Master. If Multi-drop Acknowledge is enabled, however, then the acknowledge would confirm that the message was received correctly.

This page is intentionally left blank

Appendix C

Mercury 2+ Terminal Wiring Schedule

Terminal	Pin No.	Description	Application
J1 Digital Output I.S. interface			
J1	1	DO1+	Dig Out 1 +ve
J1	2	DO1-	Dig Out 1 -ve
J1	3	DO2+	Dig Out 2 +ve
J1	4	DO2-	Dig Out 2 -ve
J2 Digital Input, Simple Equipment I.S. interface			
J2	1	DI1+	Dig In 1 +ve
J2	2	DI1-	Dig In 1 -ve
J2	3	DI2+	Dig In 2 +ve
J2	4	DI2-	Dig In 2 -ve
J2	5	DI3+	Dig In 3 +ve
J2	6	DI3-	Dig In 3 -ve
J3 Wiegand Swipe Card Reader			
J3	1	Coil (Black)	Wiegand Reader
J3	2	Coil (Red)	Wiegand Reader
J3	3	+5V	Wiegand Reader
J3	4	Screen	Wiegand Reader
J4 Non-I.S. External Power			
J4	1	+24V	Non I.S. Power
J4	2	0V	Non I.S. Power
J5 I.S. interface module			
J5	1	Txl+	To R507 Pin No. 16
J5	2	Tx2+	To R507 Pin No. 15
J5	3	Rxl-	To R507 Pin No. 14
J5	4	Rx2-	To R507 Pin No. 13
J5	5	Screen	To IS Cable Screen, if applicable.
J6 Proximity Card Reader Interface			
(Pin 1 is at the bottom of the connector)			
J6	1	Tx	Proximity Reader
J6	2	Rx	Proximity Reader
J6	3	GND	Proximity Reader
J6	4	+5V	Proximity Reader

This page is intentionally left blank

Appendix D

R507 I.S. Interface Module Wiring Schedule

Safe Area Connections

Pin No	Area Usage	Description
1	Safe Area	RS422 RX+ from Host
2	Safe Area	RS422 RX- from Host
3	Safe Area	RS422 TX+ to Host
4	Safe Area	RS422 TX+ to Host
5	Safe Area	+24 V Nom. Supply
6	Safe Area	0V/GND Nom. Supply
7	Safe Area	RS232 RX from Host
8	Safe Area	RS232 TX to Host

Field connections to Mercury 2+

Pin No	Area Usage	Description
9	None	Not Used
10	None	Not Used
11	None	Not Used
12	None	Not Used
13	Hazard Area	M2+ J5, Pin No. 4 (RX2-)
14	Hazard Area	M2+ J5, Pin No. 3 (RX1-)
15	Hazard Area	M2+ J5, Pin No. 2 (TX2+)
16	Hazard Area	M2+ J5, Pin No. 1 (TX1+)

This page is intentionally left blank

Appendix E

Communications with the R507 Interface Module

RS422/485 and RS232 ports share a common 0V rail which is commoned with the incoming power 0V connection and galvanically isolated from the Hazardous Area.

RS422 Tx drivers are tri-state devices which remain in a state of high impedance until data transmission, when they are asserted; this allows multi-drop operation. In point-to-point mode, RS422 drivers are constantly active, i.e. never in Hi-Z state.

Multi-drop communications are provided via RS422 or RS485. To connect RS485 to the Interface Module, the Tx+ and the Tx- terminals should be paralleled with the Rx+ and Rx- terminals respectively. See Appendix D.

The transmitters require a high impedance state and a protocol to ensure that only one transmitter is allowed to drive the 'bus' at any one time. To avoid contention, the transmitter is enabled in the marking state (see Fig. 5).

Once a message has been completely transmitted, there is a period of time before the transmitter is tri-stated (high impedance state). This period depends on the baud rate (see the table below). The master device must wait for this period before enabling its transmitter once more.

Tri-state disable times at various baud rates

Baud Rate	Time (µs)
1200	2000
2400	1000
4800	500
9600	250
19,200	150
38,400	100
57,600	75
115,200	150

RS485 Signal Levels

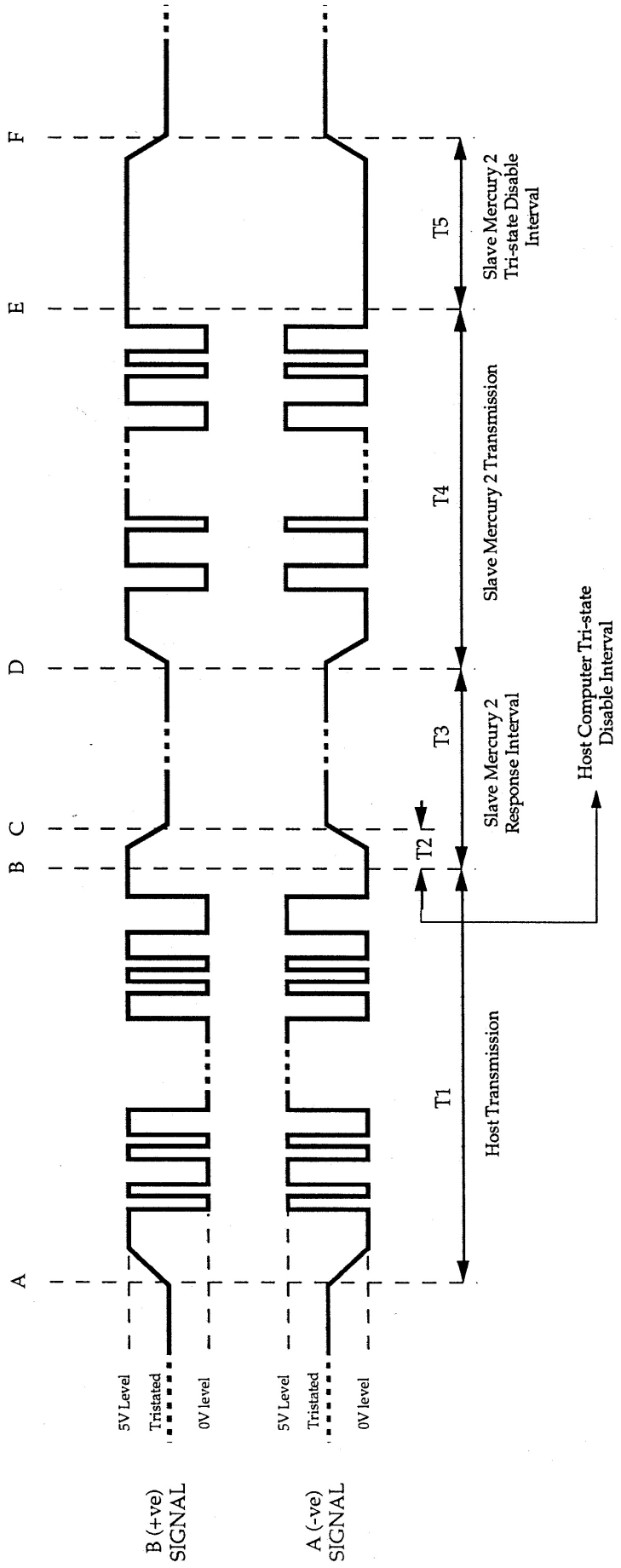
Fig. 5 illustrates signal levels on Multi-drop RS 485 lines.

At time A, the Host computer has determined that the RS485 link is not being used (previous polled message has been received) and enables its transmitter. The B signal then moves from its Hi-Z state to the marking level, a high level. At the same time the A signal moves from its Hi-Z state to the marking level, a low level. The Host then sends out its message. This occupies the time interval T1. At time B, the Host has sent the stop bit for the final character, <ETX>. The Host can now disable its transmitter and at time C the line returns to its Hi-Z state. The interval T2 is determined by the Host, but it should be short enough to ensure that the bus is tri-stated before the slave replies.

The interval T3 is the response time of the slave Mercury 2+, which has a typical time of 20ms, an absolute maximum time of 50ms and a minimum time of 10ms. At time D, the slave Mercury 2+ has enabled its transmitter and started to transmit the message requested by the Host. Interval T4 depends upon the message length and baud rate.

At time E the slave Mercury 2+ has transmitted the stop bit of the final character, <ETX>. At time F the slave disables its transmitter to the Hi-Z state. The interval T5 is determined by the Mercury 2+ system, and its maximum time is shown in the table on page 59.

Fig 5.
Signal Levels on Multi-Drop RS485 Lines

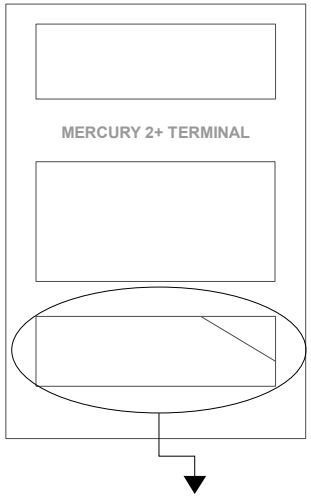


This page is intentionally left blank

Appendix F

Wiring diagrams

R054'018	M2+ Connector Layout
R054'019 Rev1	M2+ Hook-up example with Digital I/O
R054'020 Rev1	M2+ with R507 using RS232 in Safe Area
R054'021 Rev1	M2+ with R507 using RS422 in Safe Area
R054'022 Rev1	M2+ with R507 using RS485 in Safe Area
R054'023	M2+ hook-up with R007 barrier
R054'024	Safe Area M2+ using RS232
R054'025	Safe Area M2+ using RS422
R054'026	Safe Area M2+ using RS485



1	2	3	4
DO1 +	DO1 -	DO2 +	DO2 -

J1 DIGOUT

1	2	3	4	5	6
DI1 +	DI1 -	DI2 +	DI2 -	DI3 +	DI3 -

J2 DIGIN

1	2	3	4
BLK	RED	+5V	SCN

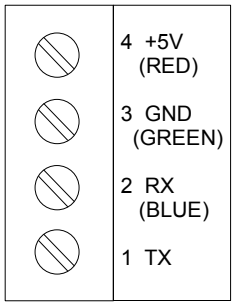
J3 WIEGAND CARDREADER

1	2
+24V	GND

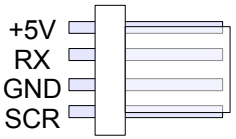
J4 NON-I.S.

1	2	3	4	5
TX1 +	TX2 +	RX1 -	RX2 -	SCN

J5 COMMS / PSU



J6 SMART CARDREADER (Orange Connector)



J7 SERIAL BARCODE WAND

Related Product Document
The QAN Responsible Person must Approve any Changes or ECNs

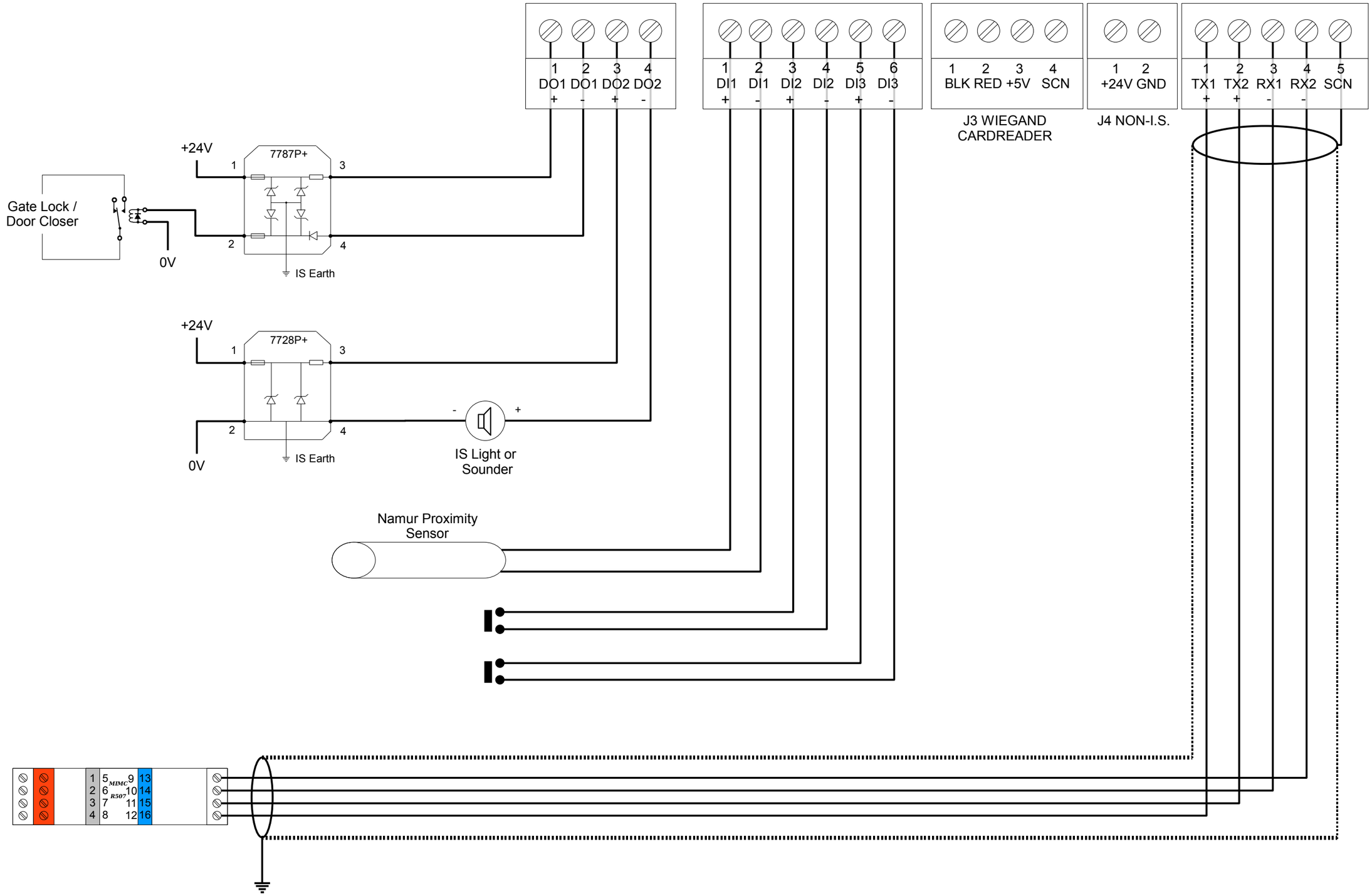
Mercury IMC

Revision	Date	Drawn	Chk'd	App'd	Description
4					
3					
2					
1					
0	04-Aug-08	GPL	MOB	MSC	Initial Drawing

Title: Mercury 2+ Connector Layout

Sheet 1 **of** 1

Drg No. R054'018



Related Product Document
The QAN Responsible Person must Approve any Changes or ECNs

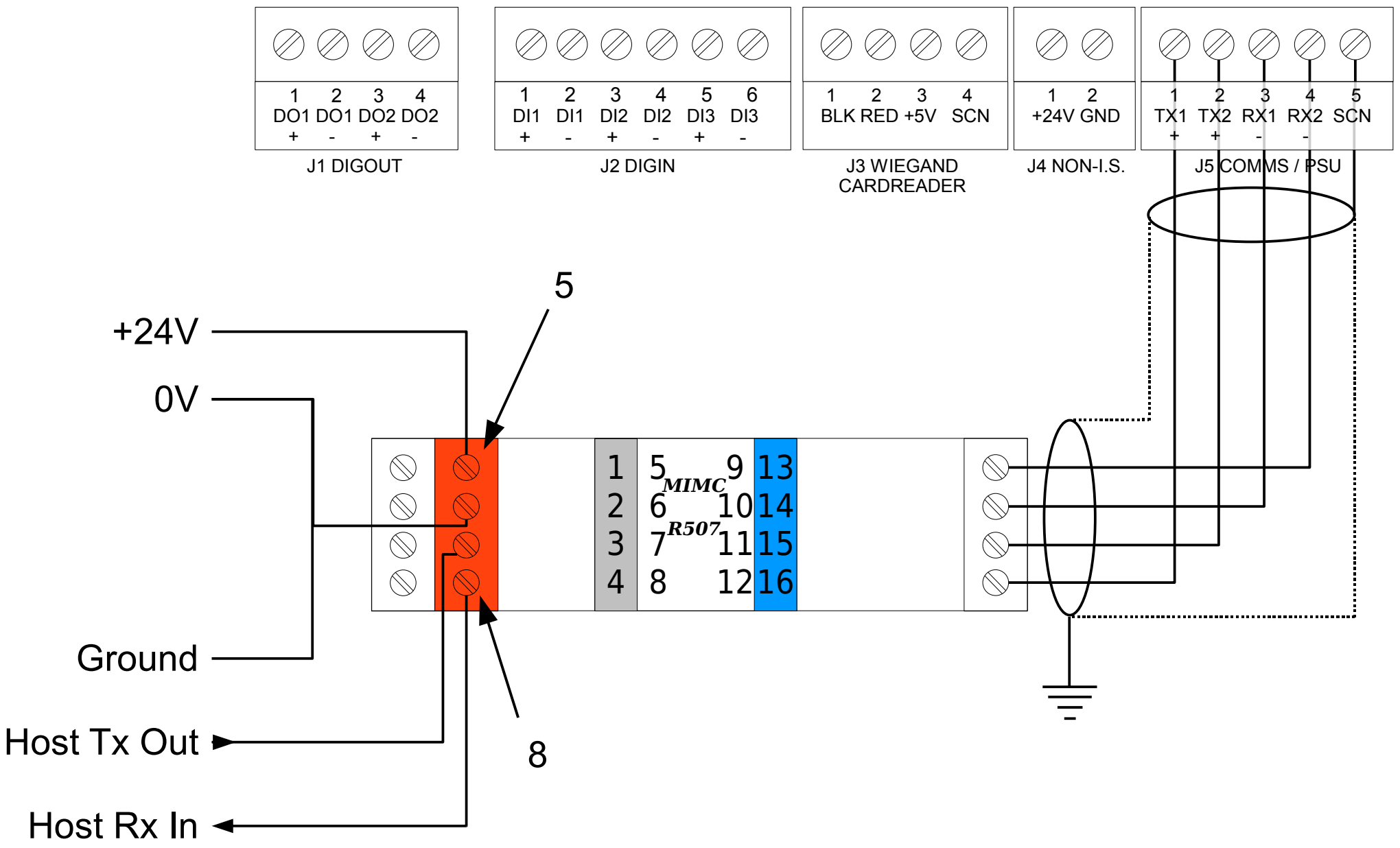
Mercury IMC

Revision	Date	Drawn	Chk'd	App'd	Description
4					
3					
2					
1	02-Mar-09	GPL	MOB	MSC	Added Orange Power Connector
0	04-Aug-08	GPL	MOB	MSC	Initial Drawing

Title: Mercury 2+ Hook-up Example with Digital I/O

Sheet 1 **of** 1

Drg No. R054'019



Related Product Document
The QAN Responsible Person must Approve any Changes or ECNs

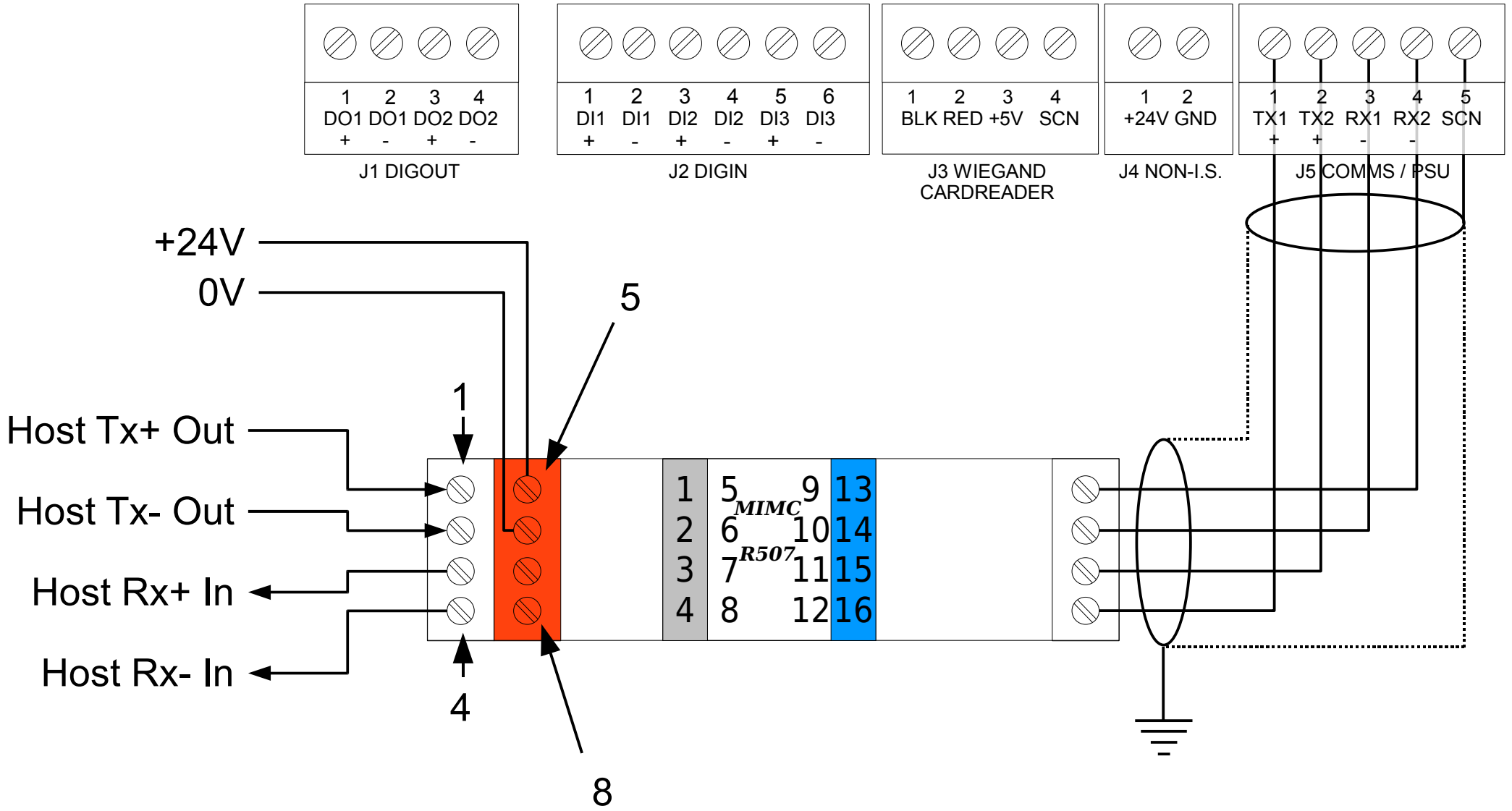
Mercury IMC

Revision	Date	Drawn	Chk'd	App'd	Description
4					
3					
2					
1	02-Mar-09	GPL	MOB	MSC	Added Orange Power Connector
0	04-Aug-08	GPL	MOB	MSC	Initial Drawing

Title: Mercury 2+ with R507 using RS232

Sheet 1 **of** 1

Drg No. R054'020



Related Product Document
The QAN Responsible Person must Approve any Changes or ECNs

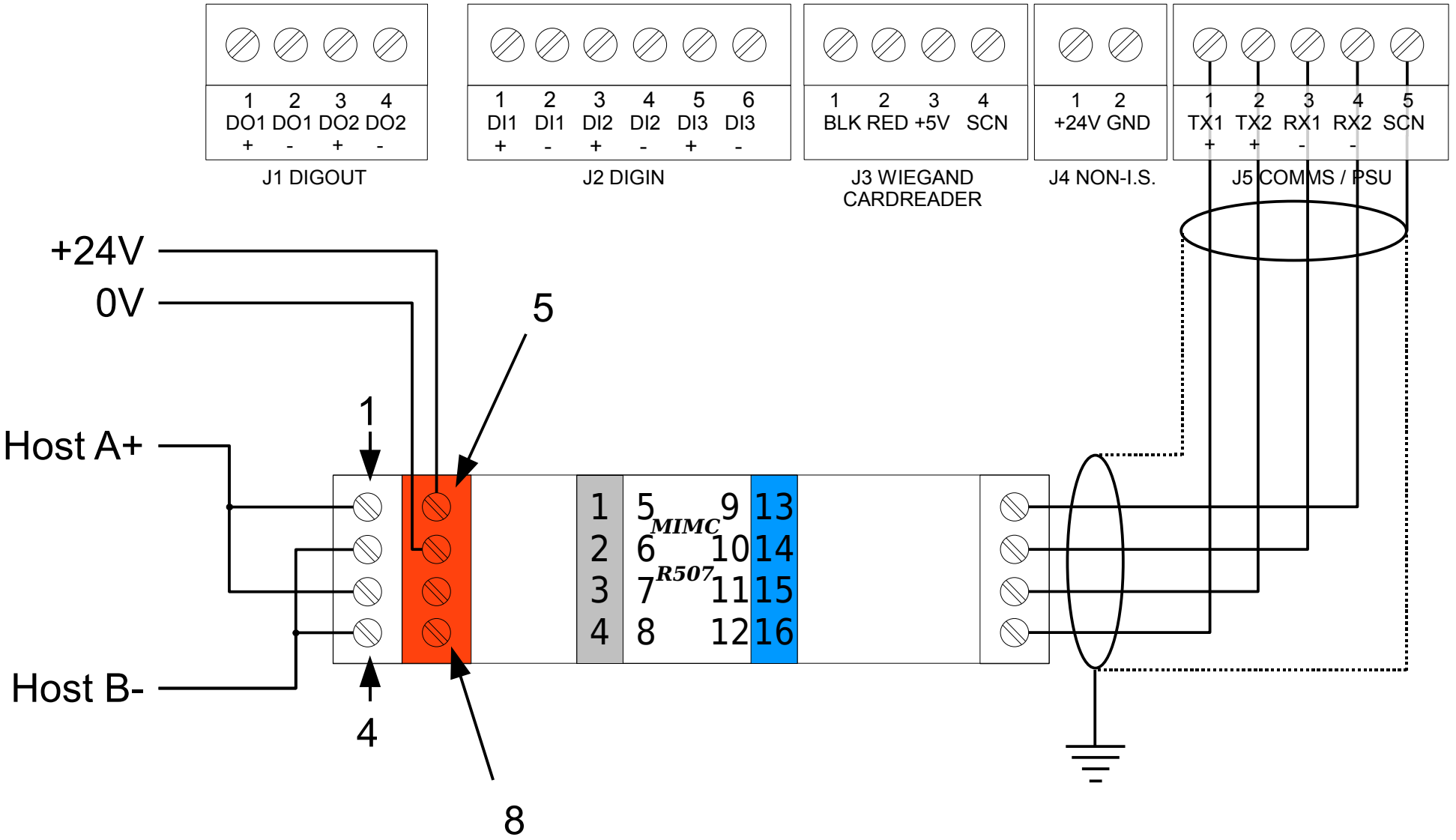
Mercury IMC

Revision	Date	Drawn	Chk'd	App'd	Description
4					
3					
2					
1	02-Mar-09	GPL	MOB	MSC	Added Orange Power Connector
0	04-Aug-08	GPL	MOB	MSC	Initial Drawing

Title: Mercury 2+ with R507 using RS422/RS485

Sheet 1 **of** 1

Drg No. R054'021



Related Product Document
The QAN Responsible Person must Approve any Changes or ECNs

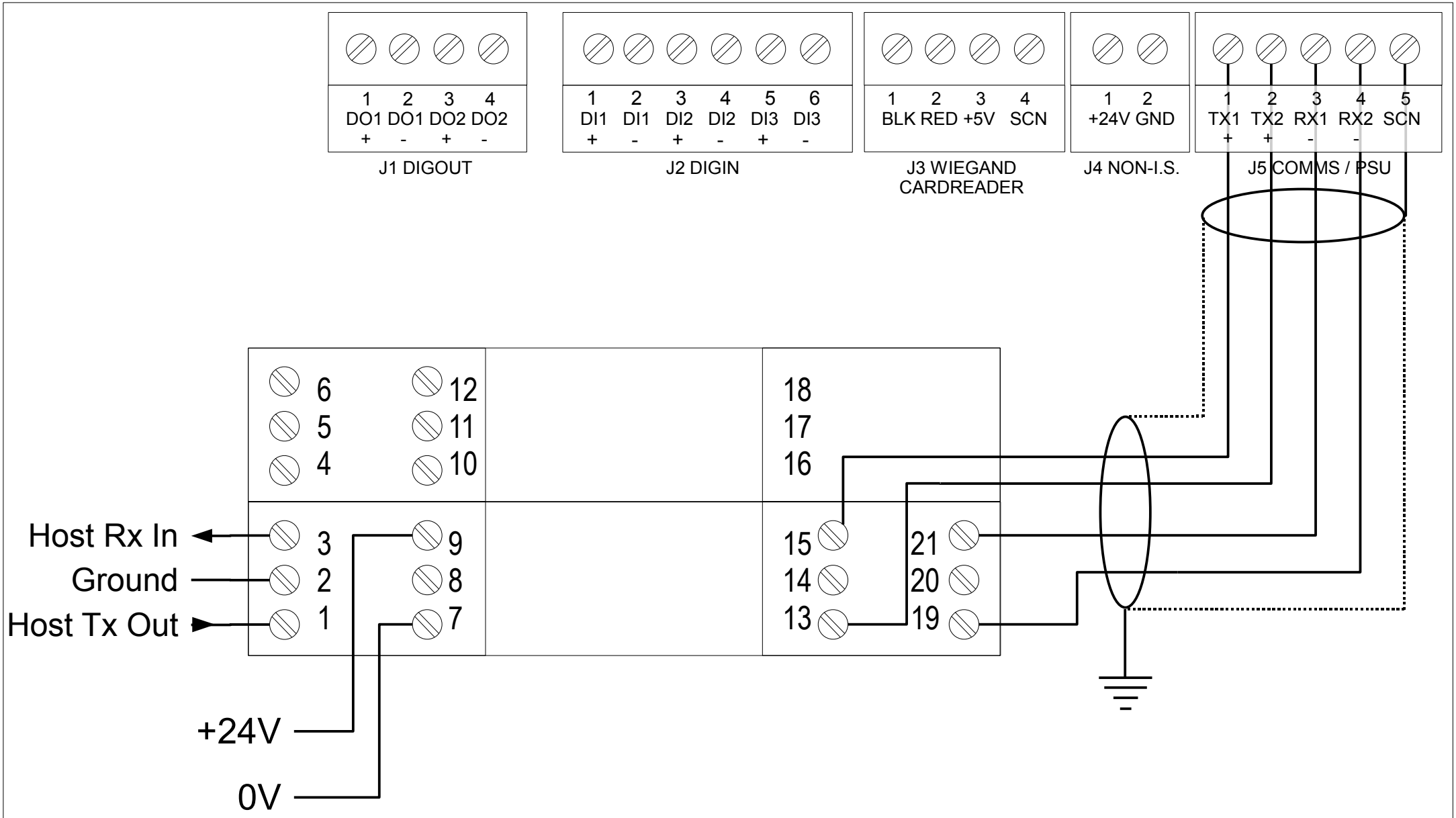
Mercury IMC

Revision	Date	Drawn	Chk'd	App'd	Description
4					
3					
2					
1	02-Mar-09	GPL	MOB	MSC	Added Orange Power Connector
0	04-Aug-08	GPL	MOB	MSC	Initial Drawing

Title: Mercury 2+ with R507 using 2 Wire RS485

Sheet 1 **of** 1

Drg No. R054'022



Related Product Document
The QAN Responsible Person must Approve any Changes or ECNs

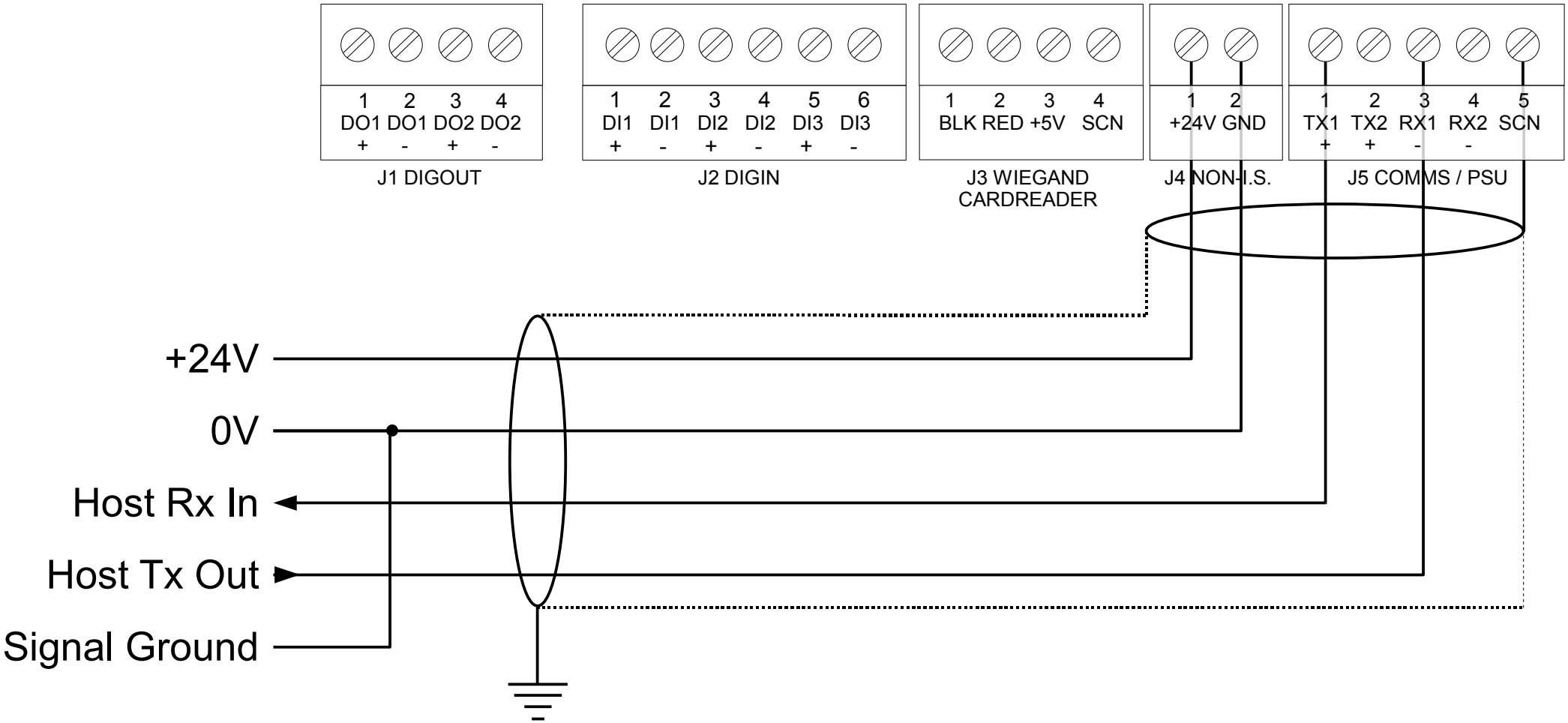
Mercury IMC

Revision	Date	Drawn	Chk'd	App'd	Description
4					
3					
2					
1					
0	04-Aug-08	GPL	MOB	MSC	Initial Drawing

Title: Mercury 2+ with R007 using RS232

Sheet 1 **of** 1

Drg No. R054'023



Only valid for
Non-IS builds of the
Mercury 2+

Related Product Document
The QAN Responsible Person must Approve any Changes or ECNs

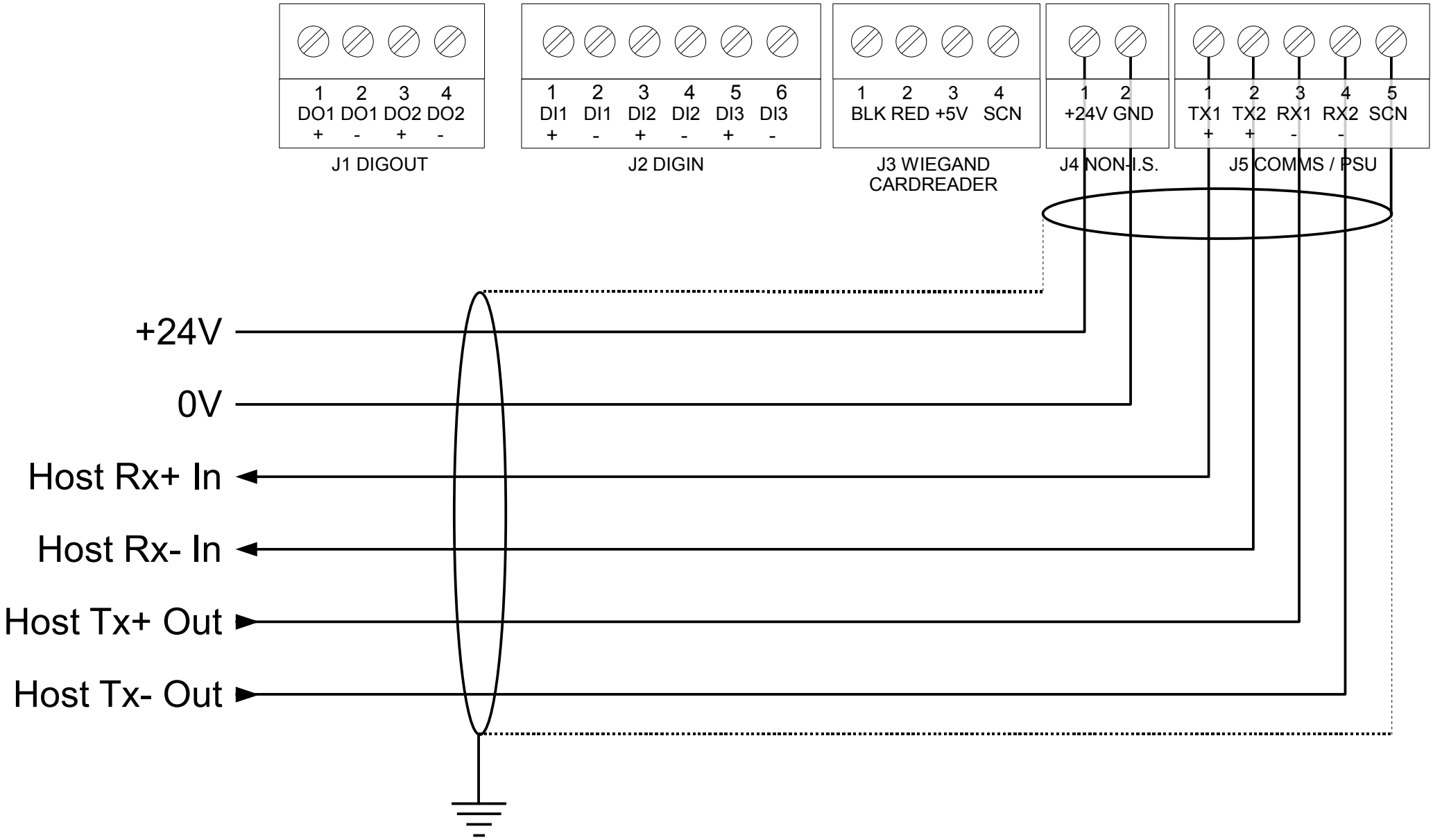
Mercury IMC

Revision	Date	Drawn	Chk'd	App'd	Description
4					
3					
2					
1					
0	04-Aug-08	GPL	MOB	MSC	Initial Drawing

Title: Safe Area Mercury 2+ using RS232

Sheet 1 **of** 1

Drg No. R054'024



Only valid for Non-IS builds of the Mercury 2+

Related Product Document
The QAN Responsible Person must Approve any Changes or ECNs

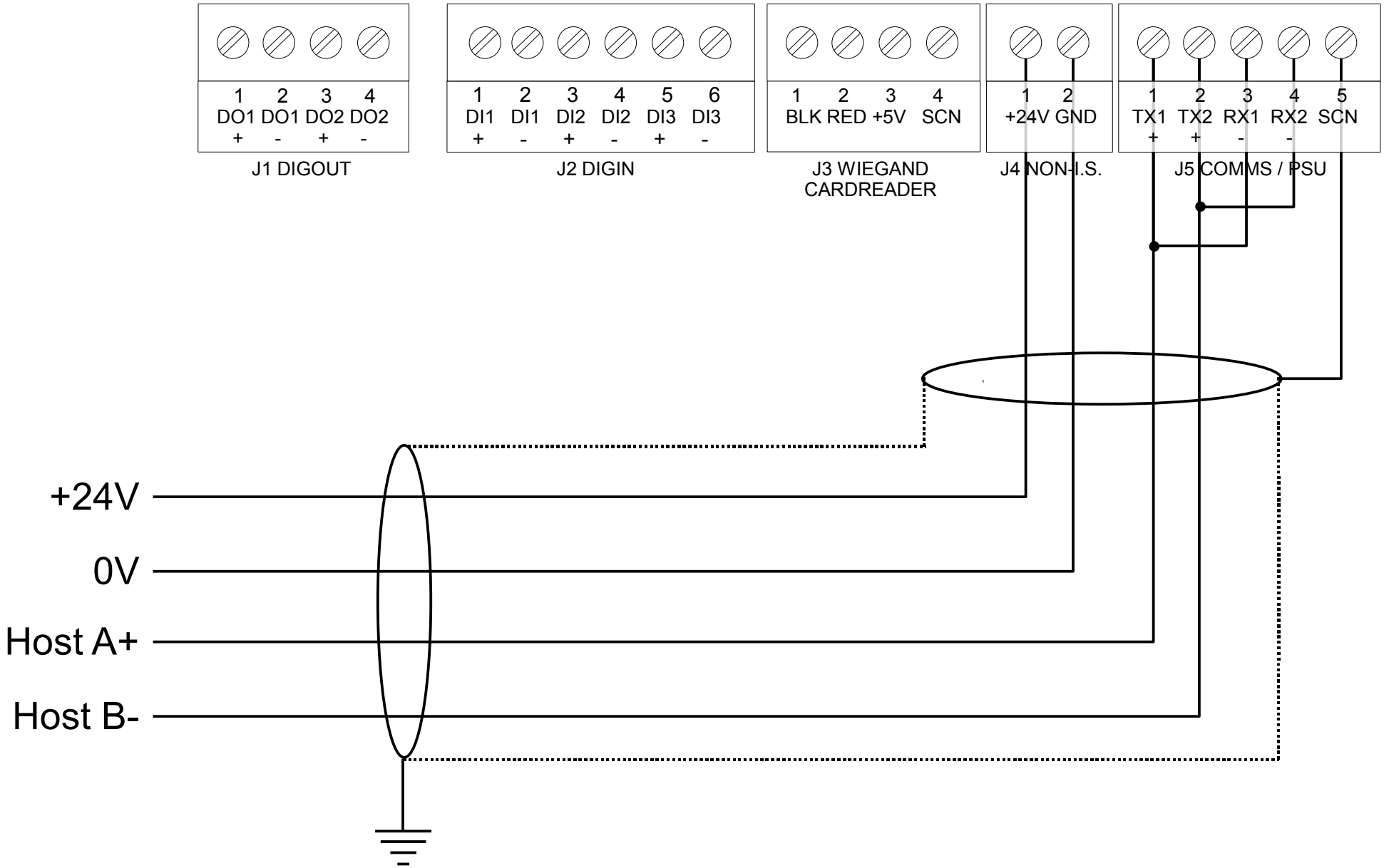
Mercury IMC

Revision	Date	Drawn	Chk'd	App'd	Description
4					
3					
2					
1					
0	04-Aug-08	GPL	MOB	MSC	Initial Drawing

Title: Safe Area Mercury 2+ using RS422

Sheet 1 **of** 1

Drg No. R054'025



Only valid for
Non-IS builds of the
Mercury 2+

Related Product Document
The QAN Responsible Person must Approve any Changes or ECNs

Mercury IMC

Revision	Date	Drawn	Chk'd	App'd	Description
4					
3					
2					
1					
0	04-Aug-08	GPL	MOB	MSC	Initial Drawing

Title: Safe Area Mercury 2+ using 2 Wire RS485

Sheet 1 **of** 1

Drg No. R054'026

Appendix G

Modbus Operation

The Mercury 2+ Modbus interface will work in the following manner:-

- The Mercury 2+ Modbus Terminal will be the Modbus slave device.
- The Mercury 2+ Modbus Terminal will respond in Modbus RTU mode only.
- The Mercury 2+ Modbus Terminal will respond only to a fairly basic set of commands, such as those that a PLC operating as a Modbus master may produce.
- Complex commands, such as graphics etc. will be preloaded into the Mercury 2+ Terminal whilst in ANSI (VT100) mode and stored as messages in the Non-Volatile memory, and simply recalled when in Modbus mode.

Screen Control Registers

Register 1 : Display Mode
A single Modbus register will be allocated to set the character and graphics screen modes and interactions.

0	Neither screen visible
1	Text on, Graphics off (Same coding as in ANSI Mode)
2	Graphics on, Text off
3	Text and Graphics on, screens logically ORed
4	Text and Graphics on, screens logically ANDed
5	Text and Graphics on, screens logically XORed

Register 2 : Set Keyboard Mode

1	Uppercase
2	Lowercase
3	Numerical

Register 3 : Reserved

Register 4 : Set Card Reader Mode

1	One shot mode
2	Disabled
3	Enabled

Screen Control Coils

1...99	Recall the stored message
100	Clear Screen
101	Cursor On/Off
102	Cursor Up
103	Cursor Down
104	Cursor Left
105	Cursor Right

106 Home Cursor
107 Reserved
108 Clear Wiegand register input buffers
109 Clear Composed Text register input buffers
110 Enable/Disable Latching Function keys
111 Clear Latched Function key register
112 Enable/Disable Latching Buffers
113 Clear Latched Buffers registers

NOTE on coils 110 to 113 ...

Each “latching” function has two coils, one to enable/disable the latching mode, and the other to clear any latched values. By default, the relevant function operates exactly as before, i.e. the register(s) is automatically cleared when read. If the latching mode is enabled (by turning on coil 110 or 112), the register(s) is not cleared automatically.

To clear the register, you must turn on coil 111 or 113 (which both clears the register and resets coil 111 or 113 back to the OFF state).

For example, the latching function keys performs as follows:-
Coil 110 OFF = Latching mode disabled
ON = Latching mode enabled
Coil 111 ON = Clear latched register, set coil 111 to OFF

Example with latching mode disabled

Action	**Comment**
Power-on	Coil 110 = OFF (Latching mode disabled)
Read register 2 » 0x0000	No function keys pressed
Press F1, F2, F3	
Read register 2 » 0x0007	Register automatically cleared
Read register 2 » 0x0000	
Press F5, F6	
Read register 2 » 0x0030	Register automatically cleared
Read register 2 » 0x0000	

Example with latching mode enabled

Action	**Comment**
Power-on	
Turn on coil 110	Latching mode enabled
Read register 2 » 0x0000	No function keys pressed
Press F1, F2, F3	
Read register 2 » 0x0007	Register not cleared by read operation
Read register 2 » 0x0007	
Press F5, F6	Extra function keys added to register
Read register 2 » 0x0037	Register not cleared by read operation
Read register 2 » 0x0037	
Turn on coil 111	Clear latched register
Read register 2 » 0x0000	

Registers For Writing Data To The Screen

There will be several ways of achieving this aim as any one method may be simpler for some users than other methods.

Registers 10...169 : Blanket coverage of the screen
The Mercury 2+ screen supports 8 rows of 40 characters, 320 character positions in total. Each pair of character positions will be assigned a single Modbus register (160 registers in total). Of the Modbus register, D0-D7 will represent the right hand character ASCII value and D8-D15 will represent the left hand character of the pair.

Register 170 : Cursor Positioning
A single Modbus register will be allocated to the cursor positioning function. D8-D15 will be the X coordinate and D0-D7 will be the Y coordinate.

Register 200 : 16 bit Unsigned Integer
A single Modbus register will be allocated which, when written to, will display, at the current cursor position, the value written as an unsigned number in the range 0 to 65535.

Register 201 : 16 bit Signed Integer
A single Modbus register will be allocated which, when written to, will display, at the current cursor position, the value written as a signed number in the range -32768 to 32767. Note that positive numbers will have no leading plus sign, whereas negative numbers will have a preceding minus sign.

Registers 202...207 : 96 bit Packed Data Float
6 registers will be allocated which, when written to, will display, at the current cursor position, the value written as a Packed Data FP. The lowest numbered Modbus address field of the pair will contain the bits <95:80> and the highest numbered register will hold the bits <15:0>. The conversion occurs when the higher numbered Modbus register is written to.

The format of the number displayed will be as follows:-

-X.XXXXXXE-XX

i.e. optional minus, compulsory 1 digit, optional decimal point, up to 6 optional digits, and an optional 2 digit exponent part (with optional minus sign).

e.g.	0.05	appears as	5.000000E-2
	145.667	appears as	1.456670E2
	-1.000	appears as	-1.000000
	0.0	appears as	0.0

Note : positive overrange, if number > 9.999999E99
‘+overrange’ displayed

positive under range, if number < 1.000000E-99
‘+underrange’ displayed

negative overrange, if number > -9.999999E99
‘-overrange’ displayed

negative under range, if number < -1.000000E-99
‘-underrange’ displayed

The FP format is as follows:-

			Word 5		Word 4	Words 3...0
	15	14	13...12	11...0	15...0	
	SM	SE		3 Digit Exp	1 Digit Integer	16 Digit Fraction
ZERO	0/1	0/1	xx	000-999	xxx0	00...00
+INRANGE	0	0/1	xx	000-999	xxx0-xxx9	00...01-99...99
-INRANGE	1	0/1	xx	000-999	xxx0-xxx9	00...01-99...99

SM = Mantissa Sign, SE = Exponent Sign

Register 208 : 16 bit Unsigned Integer (10mm)
As register 200, but 10mm high text characters (displayed on the graphics screen) will be used.

Register 209 : 16 Bit signed integer (10mm)
As register 201, but 10mm high text characters (displayed on the graphics screen) will be used.

Registers 210...215 : 96 bit Packed Data Float (10mm)
As registers 202...207, but 10mm high text characters (displayed on the graphics screen) will be used.

NOTE for registers 200 to 215 : the characters sent will overwrite the current screen data, all control characters will be ignored, and after a control character all subsequent data in the Modbus registers will also be ignored. This applies to all characters, and thus any string may be shortened by putting (for example) a null character after the last character to be displayed. The screen will wrap if the character string exceeds column 40 of the display. Valid character data will be sent to the display when data is written to bits D0-D7 of the highest relevant Modbus register.

Register 171 : 2 character string (no auto increment)
A single Modbus register will be used which, when written to, will put two characters on the screen at the current cursor position. The cursor position will not, however, be moved. In the Modbus register, the high order (D8-D15) will represent the left hand character ASCII value and low order (D0-D7) will represent the right hand character of the pair.

Registers 172...175 : 8 character string (no auto increment)
A quad set of Modbus registers will put 8 characters on the screen at the current cursor position when the last of the 4 register quads is written. The cursor position will not be moved. In each Modbus register, the high order (D8-D15) will represent the left hand character ASCII value and the low order (D0-D7) will represent the right hand character of the pair. The next Modbus register will represent the next pair of character positions.

Registers 176...195 : 40 character string (no auto increment)
This is similar to registers 172...175, but, by reserving 20 registers, allows a complete line of 40 characters to be written in one go.

Register 196 : 2 character string (with auto increment)
A single Modbus register will be used which, when written to, will put two characters on the screen at the current cursor position, and then move the cursor position along two character positions. In the Modbus register, the high order (D8-D15) will represent the left hand character ASCII value and the low order (D0-D7) will represent the right hand character of the pair.

Registers For Receiving Data from the Mercury 2+

Register 1 : Data Pending
A single resister can be read to determine if there is any valid data in the Card Reader or Text input registers. This single register contains 3 separate numbers, each corresponding to the number of relevant readings that are currently buffered in the Mercury 2+, waiting to be read.

D0-D3	Number of buffered text messages
D4-D7	Number of buffered card readings
D8-D15	Reserved

Register 2 : Function Keys
D0-D7 1 means Function Keys F1 to F8 have been pressed

Registers 3...18 : Proximity Card Reading
A set of 16 Modbus registers are used to hold the Proximity card reading.
Up to 5 Proximity card readings may be buffered, waiting to be read. If further Proximity card readings are made without the buffer being read, the subsequent data is lost.
For each Modbus register, the high order (D8-D15) will represent the left hand character ASCII value and the low order (D0-D7) will represent the right hand character of the pair.
If the Proximity card reading is less than 32 characters long, then the Modbus register "half" following the last valid character will have the value 00_H, as will all the other Modbus registers in the rest of the register set.
Note: only when register 18 is read will the next buffered reading be made available.

Registers 19...20 : Wiegand Security Card Reading
A pair of Modbus registers is used to hold the Wiegand card reading.
Up to 5 Wiegand card readings may be buffered waiting to be read. If further Wiegand card readings are made without the buffer being read, the subsequent data is lost.
The Modbus input registers are used together to hold the 32 bit value read from the Wiegand card. The lower Modbus register of the pair will contain the bits <31:16> and the higher order register will hold the bits <15:0>.

Registers 21...35 : Composed Text Modbus Input
A set of 15 Modbus registers are used to hold the Composed Text readings.
Up to 5 Composed Text readings may be buffered waiting to be read. If further text is entered without the buffer being read, the subsequent data is lost.
For each Modbus register, the high order (D8-D15) will hold the left hand character ASCII value and the low order (D0-D7) will hold the left hand character ASCII value. If the Composed Text is less than 30 characters long, then the Modbus register "half" following the last valid character will have the value 00_H, as will all the other Modbus registers in the rest of the register set.

Coils 1 to 5 : Digital Inputs (single bit reads)
5 off single bit Modbus inputs that reflect the digital input status of the Mercury 2+.

The bits are as follows:-

Coil 1	Digital Input 1
Coil 2	Digital Input 2
Coil 3	Digital Input 3
Coil 4	Unused – always reads '0'
Coil 5	Same as Digital Input 1 (this is to emulate the old Wiegand Card Present coil)

Mercury 2+ Example Modbus Messages

The following examples are given as a guide in helping set up the Modbus option of Mercury 2+.

To recap:-

- The Mercury terminal will be the Modbus slave device.
- Only standard Modicon RTU Modbus is supported.
- The default serial port settings are 9600,8,N,1.
- Complex commands, such as graphics images and backdrops, should be preloaded into the Mercury and stored as messages. These can be recalled when in Modbus mode.

The Mercury 2+ will respond to the following Modbus functions:-

1	READ OUTPUT STATUS (1)	1-109
2	READ INPUT STATUS	1-5
3	READ OUTPUT REGISTERS (1)	1-4, 10-196, 200-215
4	READ INPUT REGISTERS	1-35
5	FORCE SINGLE COIL	1-109
6	PRESET SINGLE REGISTER	1-4, 10-196, 200-215
7	READ EXCEPTION STATUS	
8	LOOPBACK TEST (2)	
15	FORCE MULTIPLE COILS	1-109
16	PRESET MULTIPLE REGISTERS	1-4, 10-196, 200-215

Notes

1. Since there is no way of reading back any of the output registers or output coils, functions codes 1 and 3 always return zeros.

2. Function code 8 currently only supports Diagnostic Code 0 (Return Query Data).

Example 1 - Read Digital Inputs (uses Function 2)

Read the status of digital inputs 1 to 5 (i.e. input coils 1 to 5) from slave device number 1.

Query Message	`01 02 00 00 00 05 B8 09`
Reply Message	`01 02 01 10 A0 44`

Example 2 - Read Data Pending Inputs (uses Function 4)

Read the Data Pending input register (input register 1) from slave device number 1.

Query Message	`01 04 00 00 00 01 31 CA`
Reply Message	`01 04 02 00 00 B9 30`

Example 3 - Clear The Screen (uses Function 5)

Clear the screen (i.e. force output coil 100) on slave device number 1.

Query Message	`01 05 00 63 FF 00 7C 24`
Reply Message	`01 05 00 63 FF 00 7C 24`

Example 4 - Display 16bit Unsigned Integer In 10mm Font (uses Function 6)

Display the 16 bit value 12345, in 10mm high characters (output register 208), on slave device number 1.

Query Message	`01 06 00 CF 30 39 6D E7`
Reply Message	`01 06 00 CF 30 39 6D E7`

Example 5 - Loopback Test (uses Function 8, Diagnostic Code 0)

Perform a simple loopback test (with data 0xFACE) on slave device number 1.

Query Message	`01 08 00 00 FA CE 23 3F`
Reply Message	`01 08 00 00 FA CE 23 3F`

Example 6 - Preset Multiple Registers

Place the message “HELLO” at the top, left corner of the display (output registers 10 to 12) of slave device number 1.

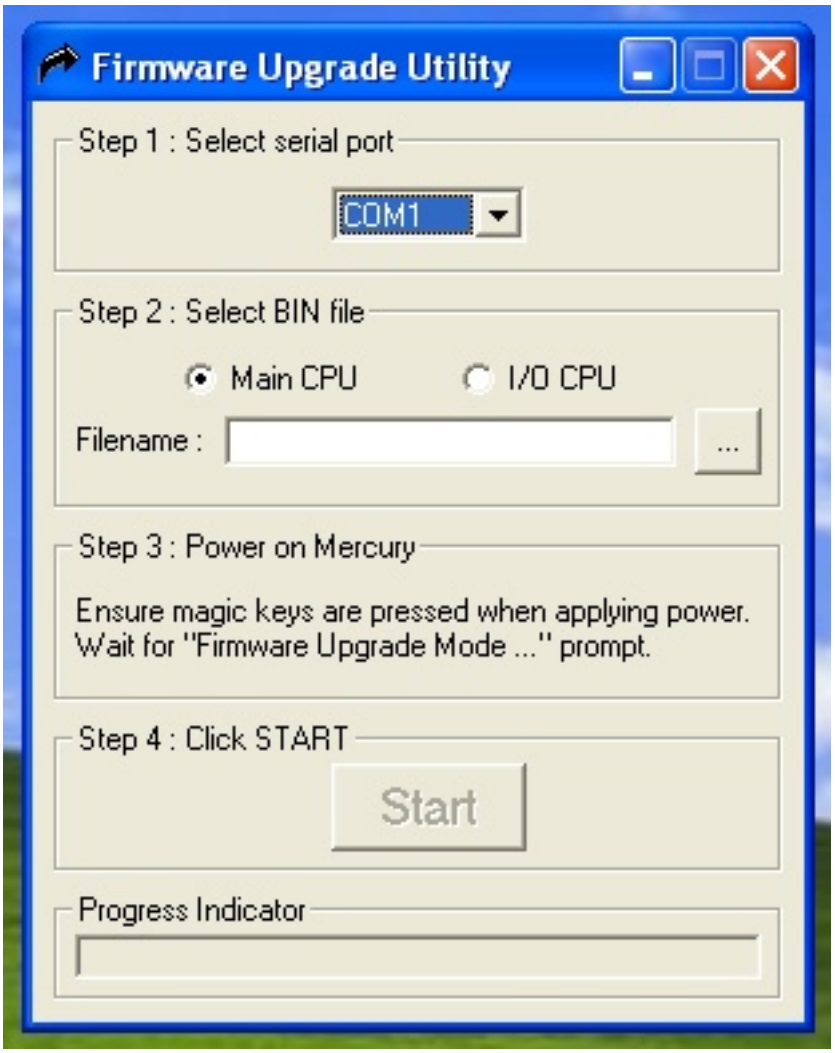
Query Message	`01 10 00 09 00 03 06 48 45 4C 4C 4F 00 17 9F`
Reply Message	`01 10 00 09 00 03 50 0A`

Appendix H

Firmware Upgrades

The Mercury 2+ unit supports in-situ upgrades to its firmware, giving the ability to add new features and fix software issues without having to return units to the factory.

To support this, Mercury HMI Ltd provide a Windows based software package allowing end users to update the firmware.



Connection

The Mercury 2+ unit should be connected to the R507 barrier as per normal operation.

The Windows PC must also be connected to the Safe Area connections (either RS232 or RS422/485) via a suitable cable.

Step 1 : Select serial port

After running up the software, please select the required comms port.

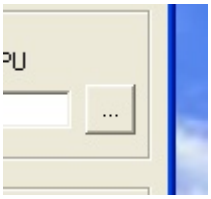
Step 2 : Select BIN file

The user must now select the firmware file to send.

The Mercury 2+ contains two separate processors (one for handle the main functions and another to handle several I/O functions).

Select the option relating to the type of firmware to be uploaded (normally files for the “main” processor will be called main_XvXX.bin, and files for the “I/O” processor will be called io_XvXX.bin ... where XvXX is the version number).

Now select the file itself using the file chooser button ...



Step 3 : Power on Mercury

Now the user must power on the Mercury 2+ unit and place it in the “Upgrade Firmware” mode.

There are a pair of “magic keys” which must be held down as the power is turned on.

On all units these keys are the two right-most on the 2nd to bottom row:-

- On a standard keyboard (with F1 ... F6), the magic keys are “=” and “SHIFT”.
- On a TA keyboard (with Start Batch 1 ... Start Batch 6), the magic keys are “8” and “9”.

Once the unit has powered up in the correct mode, the Mercury display with show “Firmware Update Mode ...”.

Step 4 : Click START

Now the user can click the Start button.

Provided the comms link is connected correctly, the progress bar along the bottom will begin to fill, and the Mercury display will show “Progress : xxxx / yyyy”, where xxxx shows the increasing upgrade block count and yyyy shows the total number of blocks to be sent.

Once complete the Mercury display will show “Complete. Please reboot.”.

The user can now reboot the Mercury 2+ unit.

Please note that a factory reset is performed upon a firmware upgrade, so all settings within the Mercury 2+ unit will need to be re-configured.